DOKUZ EYLÜL UNIVERSITY
GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES

ROBUST BOOTSTRAP PROCEDURES IN THE PRESENCE OF OUTLIERS

by
Uğur BİNZAT

July, 2024
İZMİR

ROBUST BOOTSTRAP PROCEDURES IN THE PRESENCE OF OUTLIERS

A Thesis Submitted to the
Graduate School of Natural and Applied Sciences of Dokuz Eylül University
In Partial Fulfillment of the Requirements for the Degree of Doctor of Philosophy in Statistics

by
Uğur BİNZAT

July, 2024
İZMİR

Ph.D. THESIS EXAMINATION RESULT FORM

We have read the thesis entitled "**ROBUST BOOTSTRAP PROCEDURES IN THE PRESENCE OF OUTLIERS**" completed by **UĞUR BİNZAT** under supervision of **ASSOC. PROF. DR. ENGİN YILDIZTEPE** and we certify that in our opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Doctor of Philosophy.

.....

Assoc. Prof. Dr. Engin YILDIZTEPE

Supervisor

.....	.....
Prof. Dr. Abdullah Fırat ÖZDEMİR	Assoc. Prof. Dr. Hakan Savaş SAZAK
Thesis Committee Member	Thesis Committee Member
.....	.....
Examining Committee Member	Examining Committee Member

Prof. Dr. Okan FISTIKOĞLU
Director
Graduate School of Natural and Applied Sciences

ACKNOWLEDGEMENTS

Foremost, I would like to express my sincere gratitude to my advisor Assoc. Prof. Dr. Engin YILDIZTEPE for the continuous support of my study and research, for his patience, motivation, enthusiasm, and immense knowledge.

I would like to offer my special thanks to my dissertation committee members Prof. Dr. A. Fırat ÖZDEMİR and Assoc. Prof. Dr. Hakan Savaş SAZAK for their insightful comments and precious advices.

I am also greatly indebted to my friends Pervin BAYLAN, Eda Gizem KOÇYİĞİT, and Gözde NAVRUZ for their help and support.

Last but not least, I would like to express my deepest to my family for supporting me and encouraging me throughout not only writing this thesis but also all of my life with their best wishes.

Uğur BİNZAT

ROBUST BOOTSTRAP PROCEDURES IN THE PRESENCE OF OUTLIERS

ABSTRACT

Outliers are data points that deviate significantly from the majority of a dataset and show abnormal behavior. They can occur for a variety of reasons, including technical issues, changes in system behavior, fraudulent actions, human mistakes, or natural variances. Depending on the context, outliers can either offer valuable insights into underlying phenomena or represent noise and errors that necessitate careful management. When dealing with outliers, traditional approaches often prove inadequate when handling outliers, whereas robust methods are better suited for such cases.

The first contribution of this thesis is the development of a new histogram-based unsupervised outlier detection method. When a histogram contains large gaps, the dynamic bin-width approach is preferred over the static bin-width approach. These gaps in a histogram usually occur as a result of various distributions in real data. When working with a static bin-width histogram, gaps can be utilized to acquire better distinction between outliers and inliers. In this study, we propose an adjusted version of the histogram-based outlier score named adjusted histogram-based outlier score, which considers neighboring bins prior to density estimation. Results from a simulation study and real data application indicate that the adjusted histogram-based outlier score yields a better performance not only in the simulated data but also for various types of real-life data.

This thesis investigates robust bootstrap methods, especially in regression models. Robust bootstrap methods are suggested as an alternative to bootstrapping robust estimators. Bootstrapping robust estimators has gained popularity for solving various issues; however, challenges remain, particularly with outliers. Bootstrap samples can contain more outliers than the original sample, and calculating robust estimators can be costly due to their iterative processes. Influence Function Bootstrap (IFB) and Fast and Robust Bootstrap (FRB) have emerged to address these issues. In this study, we

introduce a new approach, Weighted Fast and Robust Bootstrap (WFRB), which combines IFB and FRB to increase the robustness of the bootstrap procedure.

The performance of the proposed method is determined through simulation studies and real-life datasets. Through the application of several simulation studies and analysis of real-life datasets, the results indicate that the proposed method achieves lower mean squared error and shorter confidence interval lengths compared to conventional methods in some cases. Furthermore, it consistently satisfies Bradley's liberal criterion across moderate sample sizes and various scenarios.

Keywords: Bootstrap, robust bootstrap, outlier, unsupervised outlier detection.

AYKIRI DEĞERLERİN VARLIĞINDA DAYANIKLI BOOTSTRAP YÖNTEMLERİ

ÖZ

Aykırı değerler, veri setinin büyük çoğunluğundan önemli ölçüde sapma gösteren ve anormal davranış sergileyen veri noktalarıdır. Bu tür anomaliler, teknik sorunlar, sistem davranışındaki değişiklikler, dolandırıcılık eylemleri, insan hataları veya doğal varyasyonlar gibi çeşitli nedenlerden kaynaklanabilir. Bağlama bağlı olarak, aykırı değerler ya altta yatan olguya ilişkin değerli bilgiler sunar ya da dikkatli şekilde yönetilmesi gereken gürültü ve hataları temsil eder. Aykırı değerlerle başa çıkarken, geleneksel yaklaşımlar genellikle yetersiz kalırken, yöntemler bu tür durumlar için daha uygundur.

Tezin ilk katkısı histogram tabanlı denetimsiz aykırı değer tespit yöntemi geliştirilmesidir. Histogram tabanlı aykırı değer puanı, statik veya dinamik kutu genişliğine sahip histogram kullanılarak hesaplanabilir. Histogramda kutular arası büyük boşluklar olduğunda, statik kutu genişliği yaklaşımı yerine dinamik kutu genişliği yaklaşımı daha iyi sonuçlar vermektedir. Histogramdaki kutular arası boşluklar gerçek verilerin geldiği çeşitli dağılımların bir sonucu olarak ortaya çıkabilmektedir. Statik histogram ile çalışırken, aykırı değerler ve olağan değerler arasında daha iyi bir ayrım elde etmek için bu boşluklardan yararlanılabilir. Bu çalışmada, yoğunluk kestirimi öncesinde komşu kutuları da dikkate alan düzeltilmiş histogram tabanlı aykırı değer puanı yöntemi önerilmiştir. Benzetim çalışması ve gerçek veri uygulamasından elde edilen sonuçlar, düzeltilmiş histogram tabanlı aykırı değer puanı yönteminin yalnızca yapay verilerde değil, aynı zamanda farklı türlerde gerçek verilerde de daha iyi bir performans sağladığını göstermektedir.

Bu tezde dayanıklı bootstrap yöntemleri incelenmiştir. Dayanıklı bootstrap yaklaşımları dayanıklı tahmin ediciler ile bootstrap uygulamasına alternatif olarak önerilmiştir. Dayanıklı tahmin ediciler ile bootstrap uygulamaları yaygın olsa da, bu uygulamalarda aykırı değerlerin varlığı sorun çıkarabilmektedir. Burada en önemli

sorunlardan biri bootstrap örneklemlerinin orjinal örneklemdeki aykırı değerlerden daha fazla aykırı değer içerebilmesidir. Ayrıca bazı dayanıklı kestiricilerin yinelemeli teknikler ile elde edilmesi bootstrap kullanımlarında süre ve hesaplama maliyetlerini artırmaktadır. Bu sebeple Etki Fonksiyonu Bootstrap (IFB) ve Hızlı ve Dayanıklı Bootstrap (FRB) gibi yöntemler önerilmiştir. Bu çalışmada, FRB ve IFB yöntemlerini birleştiren ve Ağırlıklı Hızlı ve Dayanıklı Bootstrap (WFRB) olarak adlandırılan yeni bir yaklaşım tanıtılmaktadır.

Önerilen yöntemin performansı, benzetim çalışmaları ve gerçek hayat verileri ile belirlenmiştir. Çeşitli benzetim çalışması ve gerçek hayat verilerinin analizi sonucunda, önerilen yöntemin geleneksel yöntemlere kıyasla, bazı durumlarda, daha düşük hata kareler ortalaması ve daha kısa güven aralığı genişliği sağladığı görülmüştür. Ayrıca, orta büyüklükteki örneklemlerde ve farklı senaryolarda tutarlı bir şekilde Bradley'in liberal kriterini karşılamaktadır.

Anahtar kelimeler: Bootstrap, dayanıklı boostrap, ayrıkı değer, denetimsiz aykırı değer tespiti.

CONTENTS

Page

Ph.D. THESIS EXAMINATION RESULT FORM ii
ACKNOWLEDGEMENTS iii
ABSTRACT iv
ÖZ vi
LIST OF FIGURES x
LIST OF TABLES xi

CHAPTER ONE – INTRODUCTION 1

CHAPTER TWO – OUTLIER DETECTION 4

2.1 Definition 4
2.2 Statistical Outlier Detection Methods 7
2.2.1 Parametric Outlier Detection 8
2.2.2 Non-Parametric Outlier Detection 12
2.3 Adjusted Histogram Based Outlier Detection 13
2.3.1 Static and Dynamic Bin-Width Histograms 14
2.3.2 Adjusted Histogram-Based Outlier Score 16
2.3.3 Numerical Studies 17
2.3.3.1 Simulation Study 18
2.3.3.2 Real-Life Data Study 21

CHAPTER THREE – THE BOOTSTRAP 24

3.1 Definition 24
3.2 Bootstrap Confidence Interval 25
3.2.1 Normal-Theory Intervals 26
3.2.2 Percentile Intervals 27
3.2.3 Bootstrap-t Intervals 27

3.2.4 Bias-corrected and Accelerated Bootstrap Intervals 28
3.3 Bootstrap in Regression Analysis 29
3.3.1 Fixed-x Resampling 30
3.3.2 Random-x Resampling 31

CHAPTER FOUR – ROBUST BOOTSTRAP METHODS 32

4.1 Effect of Outliers in Bootstrap 35
4.2 Robust Regression Methods 36
4.2.1 Least Median of Squares 37
4.2.2 Least Trimmed Squares 38
4.2.3 MM Regression 38
4.3 Robust Bootstrap Methods in Regression 39
4.3.1 Influence Function Bootstrap 39
4.3.2 Weighted Bootstrap with Probability 41
4.3.3 Fast and Robust Bootstrap 42
4.4 Weighted Fast and Robust Bootstrap 43
4.5 A New Approach to Use Good Leverage Points in Bootstrapping Regression Models 44

CHAPTER FIVE – SIMULATION AND REAL-LIFE DATA STUDY 47

5.1 Simulation Study 48
5.2 Real-Life Data Study 66

CHAPTER SIX – CONCLUSION 69

REFERENCES 72

LIST OF FIGURES

Page

Figure 2.1 Example for outliers 5
Figure 2.2 Example of boxplot 9
Figure 2.3 Example of outliers in two dimensions 11
Figure 2.4 Static and dynamic bin-width histograms in the presence of an outlier ... 15
Figure 2.5 3d scatter plot of the generated data 19
Figure 4.1 Scatter plot of outliers in simple linear regression 32
Figure 5.1 MSE, CIL, and CA results for simple linear regression (case I) 50
Figure 5.2 MSE, CIL, and CA results for simple linear regression (case II) 52
Figure 5.3 MSE, CIL, and CA results for simple linear regression (case III) 54
Figure 5.4 MSE, CIL, and CA results for simple linear regression (case IV) 56
Figure 5.5 Average MSE, CIL and CA results for multiple linear regression (case I]) 59
Figure 5.6 Average MSE, CIL and CA results for multiple linear regression (case II) 61
Figure 5.7 Average MSE, CIL and CA results for multiple linear regression (case III) 63
Figure 5.8 Average MSE, CIL and CA results for multiple linear regression (case IV) 65
Figure 5.9 Least squares and LMS fits on Belgium phone calls data 67

LIST OF TABLES

Page

Table 2.1 ROC-AUC, PR-AUC, and F1-Score results for the simulated data 20
Table 2.2 The Friedman and Nemenyi post-hoc test results for the simulated data... 21
Table 2.3 Description of the real datasets 22
Table 2.4 ROC-AUC, PR-AUC, and F1-Score results for real datasets..... 23
Table 4.1 MSE results for standard error estimation of the mean..... 35
Table 4.2 MSE results for slope coefficient estimation 35
Table 5.1 Comparison of robust bootstrap methods for case I: $\varepsilon \sim N(0,1)$ 49
Table 5.2 Comparison of robust bootstrap methods for case II: $\varepsilon \sim t_3$..... 51
Table 5.3 Comparison of robust bootstrap methods for case III: $\varepsilon \sim N(0,1)$ and %10 of error replaced by 5 53
Table 5.4 Comparison of robust bootstrap methods for case IV: $\varepsilon \sim$ Contaminated Normal $0.9N(0,1) + 0.1N(0,9^2)$ 55
Table 5.5 Comparison of robust bootstrap methods (multiple) for case I: $\varepsilon \sim N(0,1)$57
Table 5.6 Comparison of robust bootstrap methods (multiple) for case II: $\varepsilon \sim t_3$ 60
Table 5.7 Comparison of robust bootstrap methods (multiple) for case III: $\varepsilon \sim N(0,1)$ and %10 of error replaced by 10 62
Table 5.8 Comparison of robust bootstrap methods (multiple) for case IV: $\varepsilon \sim$ Contaminated Normal $0.9N(0,1) + 0.1N(0,9^2)$..... 64
Table 5.9 The confidence intervals and their lengths of the coefficients (Belgium phone calls)..... 67

CHAPTER ONE
INTRODUCTION

Outliers are data points that show abnormal behavior or deviate significantly from the bulk of dataset. They have been a primary focus in statistics and other fields for many years. Outliers can be referred to by different terms such as abnormalities, discordants, deviants, or anomalies (Aggarwal, 2017). Mechanical faults, changes in system behavior, fraudulent behavior, human error, instrument error, or just natural deviations in populations may be the reason of presence of outliers (Hodge & Austin, 2004).

Depending on the context, outliers can be either the focal point of interest or a nuisance that requires appropriate handling (Wilcox, 1998). In the first case, they can provide valuable insights for the underlying processes or phenomena being studied. In the latter case, outliers could be noise or errors in the data, requiring careful consideration and appropriate methodologies.

Outliers heavily violate the conventional assumptions that the traditional methods require to work well. One of the common mistakes when dealing with outliers is discarding them without prior investigation which causes valuable insights unnoticed. In this case, common practice is using robust methods over traditional ones since robust methods are designed to work well with outliers.

However, it's crucial to note that in some cases the asymptotic properties of robust methods, like the exact distribution, are valid only under certain mild conditions (Salibian-Barrera & Zamar, 2002). To address this, combining robust estimators with Efron's bootstrap, a popular resampling method based on resampling data with replacement, is a commonly employed technique. This combination of robust estimators with bootstrap methodology typically provides reliable results, even when dealing with limited sample sizes (Wilcox, 2017)

In some cases, two problems exist when using robust estimators with bootstrap.

First, some of the bootstrap resamples might have more outliers than the original sample which decreases the breakdown point of the whole bootstrap procedure. The second one is some of the robust estimators do not have explicit solutions. Solving these equations requires a considerable amount of time and when it is combined with bootstrap the problem gets even worse.

The above problems emerge new solutions in the bootstrap procedure which can be generalized under "Robust Bootstrap Methods". The Robust Bootstrap Methods are generally practiced in regression modeling in the literature. In order to protect the bootstrap procedure from outliers, Singh (1998) suggested winsorizing the data prior to bootstrapping to control the tail condition. The study was limited to certain kinds of *L* and *M* estimators. (Salibian-Barrera & Zamar, 2002) proposed a fast and robust bootstrap (FRB) algorithm to deal with problems associated with using robust estimators via bootstrap.

Amado & Pires (2004) proposed Influence Function Bootstrap (IFB), which is based on based on re-weighting the bootstrap selection probabilities. IFB aims to protect the bulk of the data and assign lower bootstrap selection probabilities to the outlying observations. Amado et al. (2014) published a more detailed study on IFB. Norazan et al. (2009) proposed a new Weighted Bootstrap with Probability (WBP) for linear regression models which is similar to IFB.

With the intention of making a contribution to Robust Bootstrap Methodology, a new method is proposed which combines FRB with IFB in order to further robustify the bootstrap procedure.

The thesis consists of six chapters. Chapter Two provides an in-depth look at outlier diagnosis and includes a comprehensive review of the methods and techniques used to identify anomalies in data sets. Additionally, a new histogram-based outlier detection approach was suggested. Detailed information about the Bootstrap method is presented in Chapter Three. In this section, the basic principles and application areas of the bootstrap method are discussed comprehensively. Chapter Four

introduces the second proposed method and investigates the Robust Bootstrap Methods. This section describes in detail the development and application of these methodologies. Chapters Five and Six include simulation and real-life data studies and the conclusion. Chapter Five shows the application of the proposed and robust methods to simulated data and demonstrates their usefulness and robustness in order to assess their performance. Real-life data examples are also presented to validate the effectiveness of these methods. Finally, Chapter Six concludes the thesis by summarizing the main findings and contributions of the research.

CHAPTER TWO
OUTLIER DETECTION

This section delves into the literature and statistical methods associated with outlier detection.

2.1 Definition

Outlier detection has gained significant research attention in recent years due to its extensive applications across various domains. It is widely used in areas such as intrusion detection, medical diagnostics, credit card fraud detection, fault detection in critical safety systems, and surveillance of enemy activities (Chandola et al., 2009). Over the years, the definition of what constitutes an anomaly has evolved. Initially, outliers were considered as irrelevant or deviating observations generated by different mechanisms (Anscombe, 1960, Grubbs, 1969, Hawkins, 1980). Breunig et al. (2000) and Chandola et al. (2009) expanded this definition to include neighborhood, pattern, and behavioral characteristics of data. Generally, outliers are defined as data points that exhibit behavior significantly different from the majority of the dataset. Outliers can occur due to mechanical faults, system behavior changes, fraudulent activities, human errors, instrument errors, or natural deviations in populations (Hodge & Austin, 2004).

In general, three types of outliers exist, global, contextual, and collective. Global outliers, also known as point outliers, deviate from the rest of the data and can be seen as the simplest form of outliers. For example, in a class where most test scores fall between 50 and 80, a score of 1 would be considered an outlier. Contextual outliers show significant deviations when considered within a specific context or behavior. For example, a temperature of 0 Celsius measured in winter may not be an outlier however, it can seen as an outlier in summer. Collective outliers show similar behavior within a group and may deviate from the entire dataset, however, they may not seen as outliers individually. Chandola et al. (2009) emphasizes that point outliers can arise in any data collection, but collective outliers can only emerge in data sets where data instances are

related. Assume that monthly electricity consumption in residential areas usually falls between 500 and 700 kWh per family. A consumption of more than 2000 kWh in a short period of a day may indicate an anomaly, probably brought on by an unexpected local energy problem or incident.

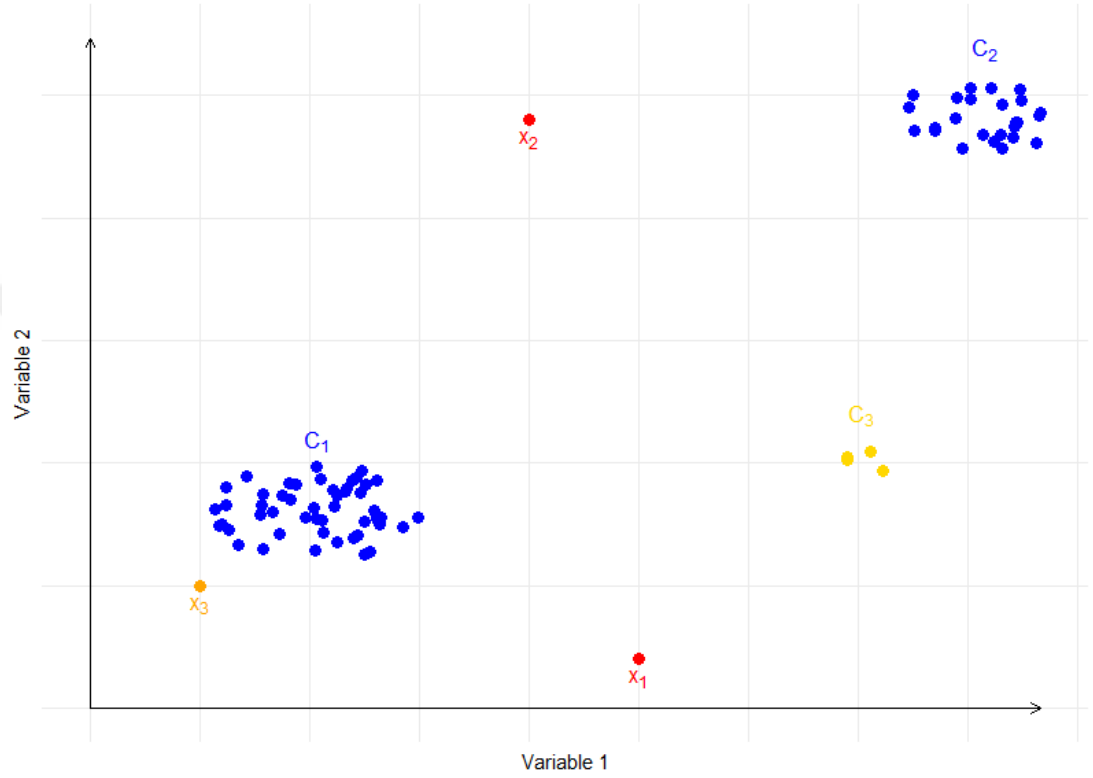


Figure 2.1 Example for outliers

An example of outliers is illustrated in Figure 2.1. The graph displays a two-dimensional scatter plot featuring normal data points (C_1,C_2) as well as outliers that deviate from the main clusters (x_1,x_2,x_3,C_3). The axes are designated as Variable 1 and Variable 2. The clusters are depicted as C_1, C_2, and C_3. The potential point outliers are identified x_1, x_2, x_3. Points x_1 and x_2 are classified as the most basic form of point outliers and can be categorized as global outliers due to their isolated nature and substantial distance from the remaining data points. At first sight, point x_3 may not be immediately recognized as an outlier because it appears to be part of the C_1. Nevertheless, when analyzing C_1 in isolation, it becomes apparent that point x_3 can be classified as a local outlier. Cluster C3 is a group of data points that deviate

significantly from the rest which are also outliers.

Anomaly detection approaches can be divided into three categories, which are supervised, semi-supervised, and unsupervised anomaly detection. Supervised methods require a fully labeled training dataset indicating normal and anomalous instances (Chandola et al., 2009, Hodge & Austin, 2004). However, it is often impractical to label all data as normal or anomalous in real-world applications. Semi-supervised approaches require a training dataset labeled only with normal instances and this approach is known as novelty detection or recognition (Hodge & Austin, 2004). On the other hand, unsupervised anomaly detection methods assume that anomalies are significantly rarer than normal instances in the data. These methods do not require pre-labeled training datasets, offering greater flexibility for real-world applications (Chandola et al., 2009, Hodge & Austin, 2004, Goldstein & Uchida, 2016). The output of an anomaly detection algorithm can be either a label or a score, with semi-supervised and unsupervised methods typically generating a score that can be converted into a label using a threshold.

There are many outlier detection techniques in the literature. These techniques can be roughly categorized into four groups: statistical, distance-based, density-based, and cluster-based outlier detection techniques.

In statistical techniques, data points are typically described using a stochastic distribution, and points are classified as outliers based on how far they deviate from the distributional model (Eskin, 2000). Commonly used methods for spotting outliers are the z-score and the IQR method. The z-score identifies points that are far away from the mean as outliers. Likewise, the IQR method determines outliers as points that fall below the first quartile or above the third quartile by a factor of 1.5 times the interquartile range. These techniques help ensure that the analysis accurately presents the central tendency and variability of the data set.

Distance-based techniques take into account the distance between data points. The general concept in distance-based methods includes local neighborhood and k-nearest

neighbor (k-NN) of the data. If a data point is significantly distant from its nearest neighbor, it is called an outlier. One of the straightforward ways to calculate distance is using the Manhattan distance or Euclidean distance metrics.

Density-based methods measure the outlyingness of a data point by inspecting its local density as well as the neighbor density. This measure is defined by the local outlier factor (LOF) which is a ratio between local densities (Breunig et al., 2000). The more the ratio between local and neighbor density increases, the more its outlyingness degree increases. A similar approach named connectivity-based outlier factor (COF) was introduced by Tang et al. (2002) to improve the drawbacks of LOF. Since distance and density-based techniques consider the distance and density of the k-NN data points, it is crucial to determine the k (Goldstein & Uchida, 2016).

Clustering-based approaches are devised to form clusters to characterize the behavior of data points. Relatively small clusters and data points that do not belong to any cluster are considered outliers. Generally, the emphasis is on clustering data points rather than detecting outliers.

In this thesis, we focused on the statistical outlier detection methods.

2.2 Statistical Outlier Detection Methods

The core principle underlying any statistical anomaly detection technique is briefly explained by Anscombe (1960): ”An anomaly is an observation which is suspected of being partially or wholly irrelevant because it is not generated by the stochastic model assumed.” Therefore, outliers are more likely to occur in the low-probability regions of a stochastic model, while normal instances are more likely to be found in the high-probability regions. These concepts form the foundation for the majority of the statistical methods that have been developed and applied to identify outliers.

Data points are often modeled and labeled as outliers using a stochastic distribution in statistical outlier detection methods. Statistical techniques involve fitting a model

and testing the new instances whether they conform to the model.

Statistical methods for outlier detection are divided into two main categories: parametric methods and non-parametric methods. Parametric approaches are based on the assumption of knowledge about the underlying distribution model, while non-parametric approaches do not require any assumptions (Eskin, 2000).

2.2.1 Parametric Outlier Detection

Parametric outlier detection methods are applicable when the data distribution is already known. In this section, we consider distribution-based and regression-based techniques as various parametric methods. The Gaussian-based outlier detection method assumes that the data distribution is Gaussian. Maximum Likelihood Estimates (MLE) are used for the parameter estimates. In general, a location and scale model is used to fit the estimates with the distribution function $G[(x-\mu)/\sigma]$, where G is usually the Gaussian distribution function. Let us say, x is independent and identically distributed (i.i.d.) with size n and the estimates are

$$\hat{\mu} = \bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i \tag{2.1}$$

$$\hat{\sigma} = s = \sqrt{\frac{1}{n-1}\sum_{i=1}^{n}\left(x_i - \bar{x}\right)^2} \tag{2.2}$$

One of the straightforward ways to detect outliers in a univariate model is using the z-score, where $z_i = (x_i - \bar{x})/s$. If $|z_i|$ exceeds the predetermined threshold, let us say 3, the point is flagged as an outlier. One of the weaknesses of this approach is that the estimates are sensitive to outliers and we might not correctly flag outliers. To avoid this scenario, one can use robust estimates of location and scale instead of MLE estimates.

Another way to detect univariate outliers is using Tukey's boxplot, which is one of the most well-known graphical methods to detect outliers (Tukeyet al. , 1977). A

boxplot uses two boundaries, also called fences, to separate the outliers and inliers, that is $[Q_1 - 1.5 \text{ IQR}, Q_3 - 1.5 \text{ IQR}]$. If an observation falls within the fences it is called an inlier and if it falls outside of the fences it is called an outlier. An example of a boxplot is illustrated in Figure 2.2 when there is an outlier. Here, Q_1 and Q_3 are the first and third quartiles, respectively. The interquartile range (IQR) is the difference between the third and first quartile, that is IQR $= Q_3 - Q_1$. However, Tukey's boxplot assumes symmetry because we move 1.5IQR from Q_1 and Q_3. When the underlying distribution is skewed regular boxplots may yield poor performance in separating outliers and inliers. To remedy this problem, Hubert & Vandervieren (2008) proposed the adjusted boxplot, which considers the asymmetry when constructing the fences.

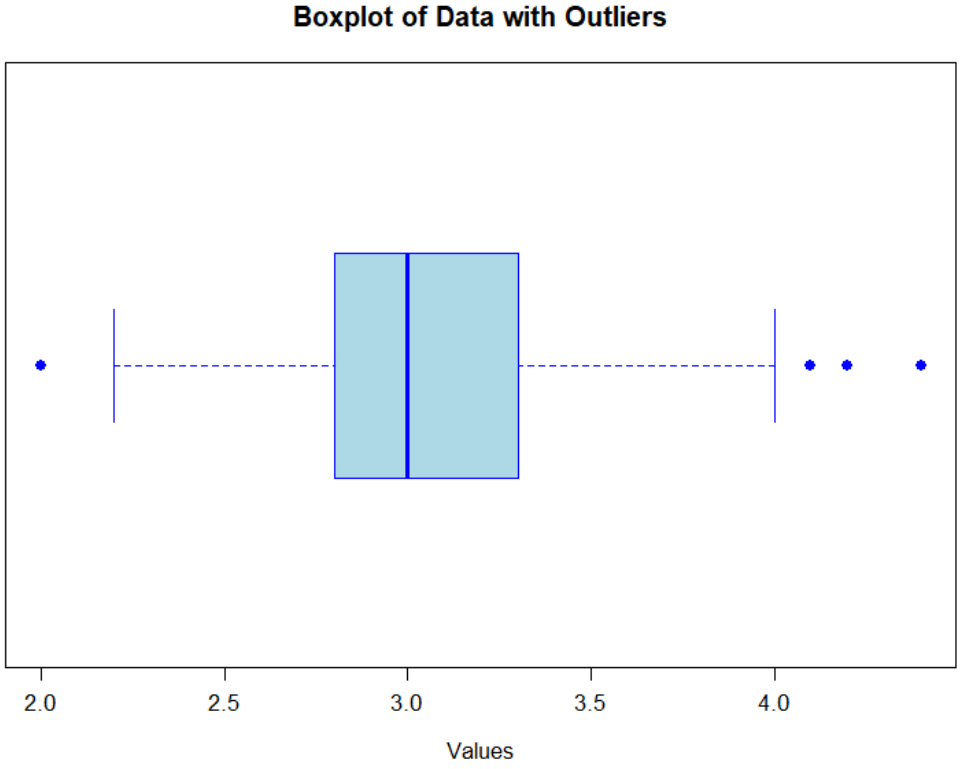


Figure 2.2 Example of boxplot

When data consists of more than one attribute or variable, it is considered multivariate data. As the number of dimensions increases, the complexity of the data also increases, leading to additional challenges. One of the primary issues that arise with high-dimensional data is the difficulty of inspection and visualization. Graphical

representations, such as scatter plots, become less effective because they can only display a limited number of dimensions at once.

One of the straightforward techniques is the use of Mahalanobis distance, which considers the correlations between variables and measures the distance of each point from the mean of the distribution in a multivariate space. Although it has a great advantage in complex data since calculations are made according to the variance and covariance of each variable in multivariate data sets, it should noted that it is vulnerable to missing values and noise.

Suppose that x_i is the data vector of interest. Then the calculation involves the following, $T(X)$ is the mean vector of the data, $C(X)$ is the covariance matrix of the data, and $C(X)^{-1}$ is the inverse of the covariance matrix. The term $(x_i - \bar{X})^T$ is the difference between the data point x_i and the mean vector $T(X)$.

Mahalanobis distance is computed as:

$$D_i = \sqrt{(x_i - T(X))C(X)^{-1}(x_i - T(X))^T} \tag{2.3}$$

where D^2 follows a chi-squared distribution with degrees of freedom equal to the number of dimensions. For a dataset with p variables, a critical value is calculated from the χ^2 (chi-squared) distribution using a certain confidence level (e.g., 95%). If D_i is greater than the calculated critical value, x_i is considered an outlier, as illustrated in Figure 2.3. This scatter plot displays that the majority of the data points are clustered within an ellipse, indicating the central tendency and normal range for these variables. The dots inside the ellipse represent normal inliers and the dots that fall outside the ellipse indicate that they deviate significantly from the central distribution of the data.

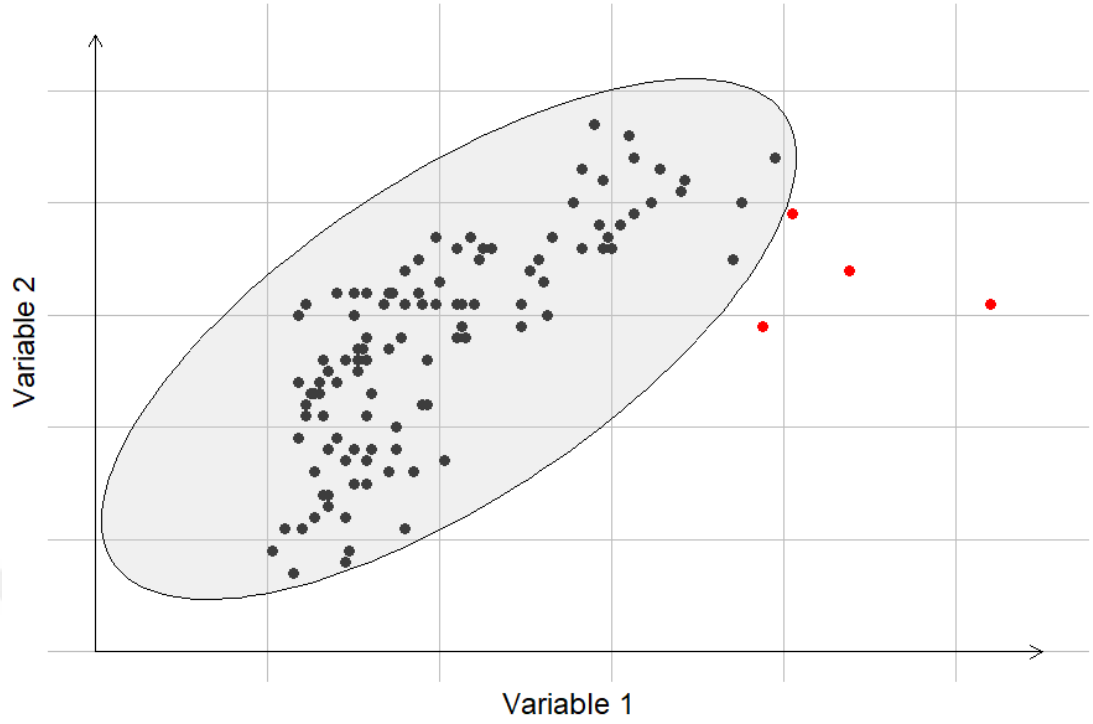


Figure 2.3 Example of outliers in two dimensions

Regression models are one of the most commonly used statistical processes to describe the relationship between dependent (Y) and explanatory variables (X). Outliers can occur in two ways: either in the Y-space (vertical) or in the X-space (horizontal). Detection of outliers begins by fitting the regression model. Then, the most straightforward way to detect outliers is by investigating the magnitude of the residuals for each instance. The least squares (LS) method is commonly used in regression analysis because it is a traditional and convenient approach. However, relying only on LS can be misleading, especially when there are small deviations from the underlying assumptions. Fitting a model with LS in the presence of outliers might give misleading results. Robust regression is a commonly employed method for handling outliers in regression modeling. (Rousseeuw & Leroy, 1987). Robust methods are not only resistant to outliers but also better tools when detecting outliers.

2.2.2 Non-Parametric Outlier Detection

Kernel and histogram-based methods are the most well-known non-parametric outlier detection methods. In general, both algorithms compare the densities of data instances in blocks. Kernel functions are commonly used in statistics to estimate the density of data instances. In non-parametric kernel density estimation, no assumption is needed for the underlying distribution. Latecki et al. (2007) was the one who came up with the idea of detecting outliers via kernel density estimation. The kernel estimation of given data points is described as:

$$\hat{q}(x) = \frac{1}{n} \sum_{i=1}^{n} \frac{1}{h(x_i)^{dim}} K\left(\frac{x - x_i}{h(x_i)}\right) \tag{2.4}$$

here, n represents the number of instances and dim gives the corresponding dimension, K is the kernel function and $h(x_i)$ is the bandwidth of each data instance x_i ($i = 1, 2, ..., n$). In general, bandwidth is usually a constant and depicted as h. However, in this method, $h(x_i)$ varies depending on the local sample density. Finally, K is chosen as multivariate Gaussian, that is:

$$K\left(x\right) = \frac{1}{(2\pi)^{dim}} \exp\left(-\frac{\|x\|^2}{2}\right) \tag{2.5}$$

To detect outliers, the density of the corresponding instance is compared with the average density of its neighbors. If a noticeable difference is seen in densities, the instance is called an outlier.

Histogram is one of the most common methods to demonstrate data distribution. Goldstein & Dengel (2012) proposed a histogram-based scoring method called Histogram Based Outlier Score (HBOS). HBOS is an unsupervised multivariate anomaly detection method that computes an outlier score for each instance in data.

HBOS combines outlier scores obtained from the density of each feature of

separate histograms. Histograms of these attributes are constructed using either a static bin- width or a dynamic bin-width approach, where the height of each bin represents a density estimate (Goldstein & Dengel, 2012).

The height of each bin in a histogram represents an estimate of density for each dimension *d*. A static or dynamic method can be used to determine the bin-width. The densities are normalized such that the maximum value is 1. The HBOS of every instance x is calculated using the corresponding density (p) of the bins where the instance is located (Goldstein & Dengel, 2012):

$$HBOS(x) = \sum_{i=1}^{d} log\left(\frac{1}{hist_i(p)}\right) \tag{2.6}$$

an instance is labeled as an anomaly if its score exceeds the predetermined threshold.

HBOS assumes independence of the features, enabling the algorithm to multiply scores derived from inverse densities. As the number of features increases, the drawback of this assumption becomes less critical (Goldstein & Uchida, 2016). Using the logarithm on inverted densities provides robustness against extreme data fluctuations. However, the outliers can still reduce the performance of HBOS (Binzat & Yıldıztepe, 2023).

2.3 Adjusted Histogram Based Outlier Detection

HBOS is an unsupervised multivariate outlier detection technique that calculates an outlier score for each data instance. Histograms can be constructed using either a static bin-width or a dynamic bin-width approach.

Data in anomaly detection tasks involves various distributions, and gaps can frequently occur in these datasets. Therefore, the static bin-width approach may produce poor density estimation when applied to datasets containing huge gaps

between data instances. If the number of bins is incorrectly calculated, almost all data may be collected in a few bins. Goldstein & Dengel (2012) have provided a dynamic bin-width approach applicable to this eventuality. Due to the mentioned reason, the dynamic bin-width approach is favored over the static bin-width approach. Even though the dynamic bin-width approach is designed to work well with these issues, it may suffer from pointwise data fluctuations. We proposed the Adjusted Histogram-Based Outlier Score (AHBOS) to overcome these problems and showed that AHBOS efficiently utilizes gaps to detect outliers while successfully labeling inliers (Binzat & Yıldıztepe, 2023).

2.3.1 Static and Dynamic Bin-Width Histograms

A static bin-width (traditional) histogram is one of the most common methods to demonstrate data distribution. The static bin-width histogram for a dataset with n instances is constructed using k equal-width bins. The number of k should be pre-specified. The data instances falling into the bins determine the bin heights. These frequencies (heights) can also be considered to be density estimates.

Dynamic bin-width histograms differ in terms of having non-static bin-widths where n/k data instances are distributed equally into k bins (Goldstein & Dengel, 2012). The area of each bin is equal since the number of samples that fall into the bins is equal. Every bin is rectangular, having a dynamic height and width. Data instances covering larger areas will eventually yield lower heights. In the dynamic bin-width approach, data instances are distributed into specified bins equally. However, as this is not applicable in some cases, the algorithm should allow more than n/k instances in the bins. In addition to the area, the height of these bins should also grow appropriately. The remaining instances can be allocated to the bin where the median is located. Likewise, the number of bins in the dynamic histogram cannot be larger than the unique values in the feature (Paulauskas & Baskys, 2019). In this case, the number of bins diminishes to the number of unique values in the feature.

The critical point in constructing a histogram is correctly determining the bin number (k). Inappropriate k can cause "under-smoothing" or "over-smoothing". Wand [20] discussed the choice of bin-width in detail and proposed a data-based choice. A common practice is to choose the square root of the *n*, where *n* is the number of instances (Goldstein & Dengel, 2012). Sturges' rule is the most well-known and oldest method to compute the number of bins. The *hist()* function in the R statistical programming language (R Core Team, 2023) has three built-in options for setting the "break" which are "Sturges" (Sturges, 1926) as default, "Scott" (Scott, 1979), and "FD" (Freedman & Diaconis, 1981). After computing bin-width with these rules, R uses a "pretty" function that ensures a good choice of bin number (Wand, 1997, Venables & Ripley, 2002). Many studies have been published on determining the optimal number of bins (Wand, 1997, Shimazaki & Shinomoto, 2007); however, this discussion is beyond the scope of this study.

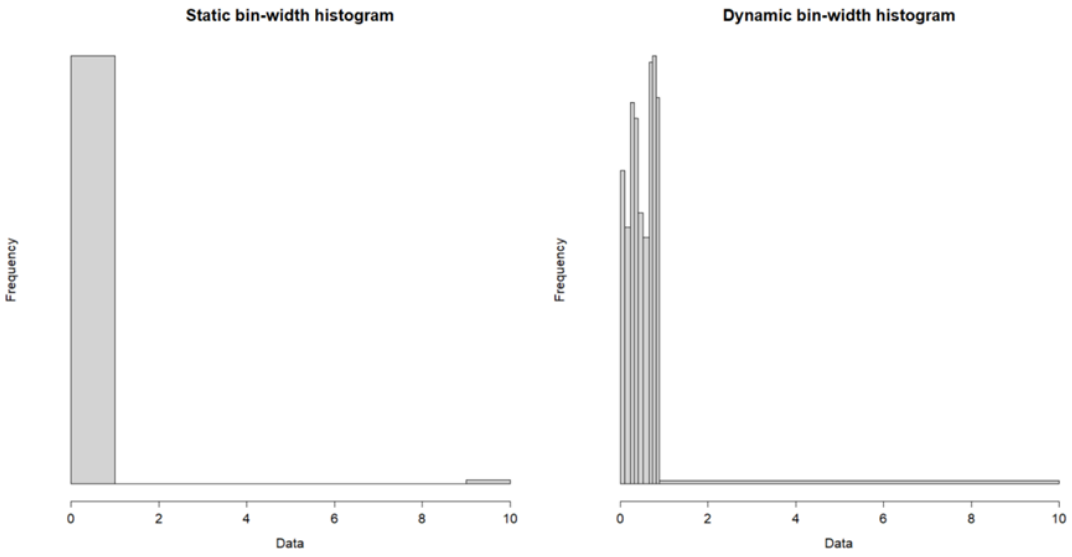


Figure 2.4 Static and dynamic bin-width histograms in the presence of an outlier

Wand (1997) stated that practitioners should be cautious about using the default rules of statistical packages in applications where important features in the dataset could be unnoticed. To illustrate, assume a mixed dataset with 99 data from a standard uniform distribution [0:1] and an extreme point valued at 10. For simplicity, we set the *k* to

10 for both approaches. Figure 2.4 illustrates both approaches in the presence of an extreme value.

Figure 2.4 shows that almost all data in the static bin-width histogram is accumulated in a single bin; hence, this does not represent a reasonable estimation of data. This example indicates why the static bin-width approach would not work well with anomalies. The over-smoothing problem is inevitable if a non-robust rule computes the number of bins in the presence of extreme value(s). Since anomalies are expected in datasets, there is merit in using a robust FD rule for a static bin-width approach. Implementing the FD rule also allows the static bin-width approach to work efficiently with heavy-tailed distributions.

The dynamic bin-width approach is recommended for unknown or heavy-tailed distributions. However, having only one extreme value would yield harmful effects on outlier scores of non-outlier observations. Since the number of data instances is distributed equally to the bins, non-outlier, and outliers might fall into the same bin. As evident in Figure 2.4, having only one extreme value lowered the densities of other instances in the same bin. Eventually, this would increase the false positive rate.

2.3.2 Adjusted Histogram-Based Outlier Score

When working with static bin-width histograms, outliers, and inliers may have the lowest densities, which is an additional concern. Since HBOS is a density-based approach, it might not be possible to distinguish these observations density-wise. When the population distribution is unknown, gaps between ordered values can help detect outliers (Wilkinson, 2017). When working with heavy-tailed distributions or in the presence of extreme values, gaps in the histograms tend to occur more frequently. The neighborhoods of the bins and gaps also contain information that could be applied for better density estimation.

With this motivation, we presented a new method based on adjusting bin heights

using neighbor bins; the adjustment was calculated by taking the average of the corresponding bin and its adjacent bins' heights. The method uses the FD rule as default to determine the number of bins, enabling the static bin-width histogram to work well with heavy-tailed distributions.

Assuming that a static bin-width histogram consists of k bins, the adjusted height of the j_{th} bin is computed below:

$$h'_j = \frac{h_{j-1} + h_j + h_{j+1}}{3}, \qquad j = 1, ..., k \tag{2.7}$$

where h_j is the height of the j_{th} bin of the histogram.

The prime notation indicates the adjusted version of the corresponding bin. As the first and last bins naturally do not have an adjacent bin, we have added dummy bins h_0 and $h_{(k+1)}$ with a height of 0. The adjusted representation of the bin heights (h'_j) yields the adjusted histogram and these adjusted heights (densities) are subsequently normalized so that the maximum value is 1. Consequently, the AHBOS of every instance *x* is calculated using the corresponding adjusted height of the bins where the instance is located:

$$AHBOS(x) = \sum_{i=1}^{d} log\left(\frac{1}{Ahist_i(p)}\right) \tag{2.8}$$

here, $Ahist$ represents the adjusted histogram, which has been normalized, and $Ahist_i(p)$ stands for the density of the corresponding instance for the ith feature. The sum of the logarithm of inverse densities yields the $AHBOS$ for each instance.

2.3.3 Numerical Studies

We conducted real and simulated data applications to evaluate the performance of the AHBOS. We used 17 simulated and 11 real data sets. Receiver operating characteristic

(ROC), Precision-Recall (PR) curves, and F-1 score were evaluated as the performance metrics.

The ROC curve is a commonly used metric in binary classification. Provost et al. (1998) suggested using the ROC curve over accuracy. The ROC curve represents how the True Positive Rate varies against the False Positive Rate. We used the area under the ROC curve (ROC-AUC) results which is equal to 1 for perfect classification. Naturally, anomalies rarely occur in the data, resulting in imbalanced classification. Davis & Goadrich (2006) argued that the performance of the ROC curve is questionable for imbalanced classification. They recommended using the PR curve alongside the ROC curve. The PR curve represents the Precision (P) versus Recall (R) for all thresholds. The area under the PR curve (PR-AUC) ranges from 0 (naive classification) and 1 (perfect classification). We also included the F-1 measure, which yields a balanced result of Precision and Recall: (2×P×R/(P+R)).

For all performance metrics, we applied the Friedman test (Friedman, 1937, 1940) and the Nemenyi post-hoc test (Nemenyi, 1963) to check whether the differences between methods were statistically significant (Demšar, 2006). In the study, the significance level of the tests was set to 0.05. All computations were performed using "R 4.2" (R Core Team, 2023). The "Cutpointr" and "MLmetrics" packages were used to compute ROC-AUC, F-1 score, and PR-AUC (Thiele & Hirschfeld, 2021, Yan, 2024). Three-dimensional plot was constructed via the "Scatterplot3d" package (Ligges & Mächler, 2002). Statistical tests were performed with "PMCMRplus" package (Pohlert, 2023).

2.3.3.1 Simulation Study

HBOS is based on the assumption of independence between features, and this assumption usually does not hold for real data. Therefore, we generated simulated datasets not only to control global outliers but also to provide gaps. For the sake of simplicity, the number of features was limited to three. We generated data

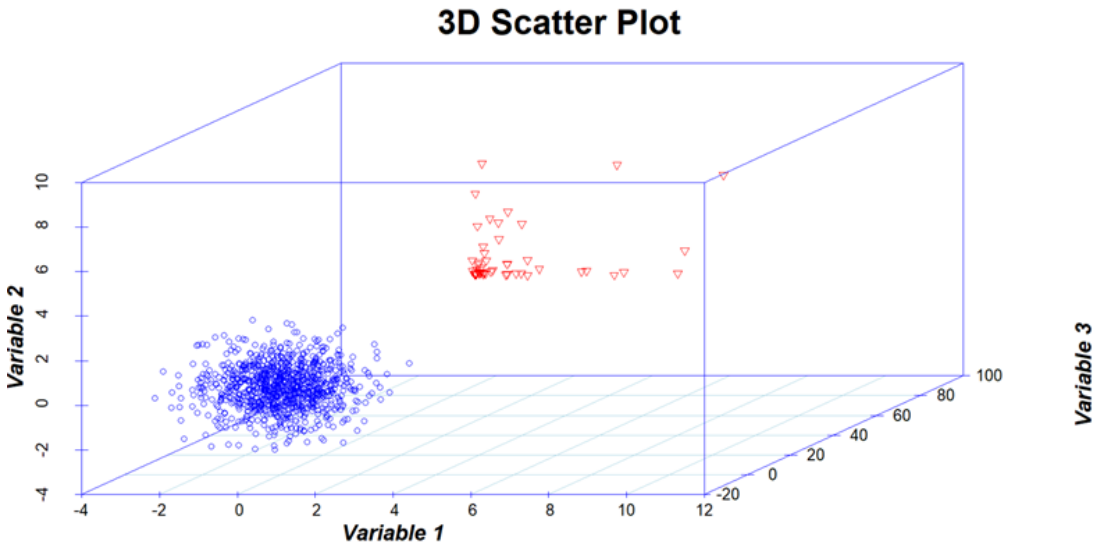


Figure 2.5 3d scatter plot of the generated data

independently from a standard normal distribution (N(0,1)), and a certain percentage of data was replaced with outliers. The first two variables contain outliers from shifted chi-square distribution ($\chi_1^2 + 5$) and the final variable only contains an extreme value of 100. Figure 2.5 provides the 3d visualization of one simulation setting when the number of instances is 1000 and the outlier percentage is 0.05. Inliers and outliers are represented as circles and triangles, respectively. Figure 2.5 clearly illustrates the successful generation of gaps of varying magnitudes between outliers, thereby fulfilling our primary objective in the simulation study. We used a broad range of sample sizes and outlier percentages to cover many possible settings. Also, consistency was acquired by repeating the data generation process ten times, and the results' averages are provided in Table 2. The number of best scores for all settings is indicated at the bottom of the table. Performance metrics were calculated for a predetermined threshold and optimized for the maximum F-1 score. The best results are highlighted in boldface.

ROC-AUC results in the simulation study demonstrated how all methods performed well, and the results were above 0.90. A closer inspection of the ROC-AUC results indicates that AHBOS achieved the maximum scores in 15 of 17 cases, and for the remaining two cases, the results were very close. When considering the PR-AUC results, AHBOS performed best in all cases. Similar to the ROC-AUC

results, AHBOS achieved a maximum of 14 of 17 F-1 scores. Dynamic bin-width HBOS yielded the highest F-1 scores in the remaining three cases. The overall results showed that AHBOS is more successful than the other approaches in terms of the performance metrics used in the simulation study. In other words, AHBOS performed best in the presence of gaps and global outliers.

Table 2.1 ROC-AUC, PR-AUC, and F1-Score results for the simulated data

			ROC-AUC			***PR-AUC***			***F-1 Score***		
n	***%***	***#***	***Sta***	***Dyn***	***AHBOS***	***Sta***	***Dyn***	***AHBOS***	***Sta***	***Dyn***	***AHBOS***
100	*5*	*5*	0.994	0.962	**1.000**	0.691	0.347	**0.796**	0.881	0.653	**0.991**
100	*10*	*10*	0.947	0.957	**0.993**	0.584	0.489	**0.828**	0.698	0.781	**0.924**
500	*1*	*5*	1.000	0.990	**1.000**	0.753	0.262	**0.800**	0.965	0.572	**1.000**
500	*5*	*25*	0.985	0.982	**0.998**	0.751	0.574	**0.915**	0.733	0.756	**0.955**
500	*10*	*50*	0.945	0.941	**0.988**	0.649	0.521	**0.848**	0.659	0.690	**0.878**
1000	*1*	*10*	1.000	0.996	**1.000**	0.867	0.512	**0.897**	0.955	0.767	**0.991**
1000	*5*	*50*	0.985	0.990	**0.996**	0.755	0.729	**0.889**	0.751	0.805	**0.899**
1000	*10*	*100*	0.952	0.935	**0.988**	0.685	0.551	**0.861**	0.656	0.645	**0.873**
5000	*1*	*50*	0.999	0.999	**1.000**	0.906	0.825	**0.946**	0.875	**0.952**	0.948
5000	*5*	*250*	0.982	0.986	**0.993**	0.756	0.697	**0.845**	0.702	0.746	**0.793**
5000	*10*	*500*	0.928	0.900	**0.968**	0.642	0.507	**0.736**	0.620	0.569	**0.698**
10000	*1*	*100*	0.999	**1.000**	1.000	0.899	0.893	**0.935**	0.840	**0.973**	0.885
10000	*5*	*500*	0.976	**0.988**	0.988	0.733	0.723	**0.789**	0.685	**0.779**	0.736
10000	*10*	*1000*	0.929	0.950	**0.967**	0.640	0.626	**0.730**	0.612	0.649	**0.686**
50000	*1*	*500*	0.999	0.998	**1.000**	0.907	0.836	**0.938**	0.840	0.786	**0.881**
50000	*5*	*2500*	0.976	0.968	**0.988**	0.733	0.624	**0.789**	0.681	0.640	**0.730**
50000	*10*	*5000*	0.915	0.907	**0.953**	0.629	0.577	**0.686**	0.605	0.617	**0.649**
Best score in			0/17	2/17	15/17	0/17	0/17	17/17	0/17	3/17	14/17

Note: N is the number of instances. % and # indicate the percentage and number of outliers, respectively. Sta, Dyn, and AHBOS refer to results for static bin-width, dynamic bin-width, and AHBOS, respectively.

We have used the Friedman test to test the null hypothesis that all methods' performance metrics are equal on the datasets. If the null hypothesis is rejected, the Nemenyi post-hoc test is conducted for all pairwise comparisons. The Friedman test and Nemenyi post-hoc test p-values are given in Table 3. According to the Friedman test results, p-values were lower than the significance level for all metrics. That is, at

Table 2.2 The Friedman and Nemenyi post-hoc test results for the simulated data

Metric	***Friedman test p-value***	***Nemenyi post-hoc test p-value***	
ROC-AUC	**3.97e-5**	Sta vs Dyn	0.9372
		AHBOS vs Sta	**0.0007**
		AHBOS vs Dyn	**0.0002**
PR-AUC	**4.14e-8**	Sta vs Dyn	**0.0010**
		AHBOS vs Sta	**0.0010**
		AHBOS vs Dyn	**1.7e-8**
F-1 Score	**1.03e-4**	Sta vs Dyn	0.5585
		AHBOS vs Sta	**0.0001**
		AHBOS vs Dyn	**0.0057**

Note: The *p*-values lower than the significance level of 0.05 are given in boldface.

least one method was different from the others for all metrics. Pairwise comparisons showed that AHBOS is statistically and significantly better than the other methods.

2.3.3.2 Real-Life Data Study

There is a vast number of datasets used for anomaly detection in numerous publications, but many have questionable validity, missing references, or are publicly unavailable [38]. For the sake of reliability and reproducibility, we considered some commonly used benchmark datasets. Our real data application comprises 11 benchmark datasets that cover many aspects of the application fields and vary in size, dimension, and outlier percentage. Five data sets were pre-processed datasets published by Goldstein for anomaly detection tasks [39]. The remaining six datasets, which are commonly used benchmark datasets, were obtained from the ODDS Library [40]. A description of the real datasets is provided in Table 4, and the performance metrics are provided in Table 5. In Table 2.3, *n* is the number of instances, *d*, #, and % indicate dimension, number of outliers, and percentage of outliers, respectively. Superscripts on data names give the sources, (1) are from Goldstein & Uchida (2016) and (2) are from ODDS Library [40].

Table 2.3 Description of the real datasets

Dataset	***n***	***d***	**#**	**%**
ionosphere[2]	351	33	126	35.90
b-cancer[1]	367	30	10	2.72
pima[2]	768	8	275	35.90
cardio[2]	1831	21	176	9.61
musk[2]	3062	166	97	3.17
satellite[1]	5100	36	75	1.49
satimage-2[2]	5803	36	71	1.22
pendigits[2]	6870	16	156	2.27
annthyroid[1]	6916	21	250	3.61
shuttle[1]	46464	9	878	1.89
kdd99[1]	620098	38	1052	0.17

According to ROC-AUC results, out of 11 results, AHBOS had the highest 7, and dynamic bin-width HBOS had the highest 4. Compared with static bin-width HBOS, AHBOS had the highest results or very close values. It can be seen from the PR-AUC and F1-score results that AHBOS performed better than the other methods. AHBOS had 8 best F-1 scores while it was the second in the remaining 3 data sets. The dynamic bin-width HBOS gave better results in all metrics for "annthyroid" dataset. Overall, AHBOS showed better and more competitive performance than other methods, especially for PR-AUC and F1-score.

For real datasets, the Freidman test p-value was found 0.075 for ROC-AUC results, and none of the methods were found significantly different from the others. The Friedman test p-values were found 0.02 and 0.019 for PR-AUC and F-1 score results, respectively. According to Nemenyi's post-hoc test results for PR-AUC, AHBOS was significantly different than dynamic bin-width HBOS (p-value 0.015).

Table 2.4 ROC-AUC, PR-AUC, and F1-Score results for real datasets

	ROC-AUC			***PR-AUC***			***F-1 Score***		
Dataset	***Sta***	***Dyn***	***AHBOS***	***Sta***	***Dyn***	***AHBOS***	***Sta***	***Dyn***	***AHBOS***
ionosphere[2]	0.522	**0.693**	0.691	0.314	0.258	**0.478**	0.577	0.597	**0.656**
b-cancer[1]	**0.983**	0.980	0.982	**0.552**	0.423	0.550	**0.625**	0.556	**0.625**
pima[2]	0.640	0.654	**0.674**	0.510	0.508	**0.524**	0.529	0.545	**0.558**
cardio[2]	0.851	0.840	**0.899**	0.474	0.390	**0.512**	0.485	0.470	**0.539**
musk[2]	1.000	**1.000**	**1.000**	0.987	**0.990**	0.990	0.985	**1.000**	0.995
satellite[1]	0.923	**0.929**	0.929	0.580	0.566	**0.618**	0.597	0.607	**0.614**
satimage-2[2]	0.982	0.980	**0.985**	0.787	0.799	**0.825**	0.774	0.800	**0.809**
pendigits[2]	0.936	0.788	**0.940**	0.267	0.157	**0.270**	0.349	0.258	**0.366**
annthyroid[1]	0.832	**0.916**	0.851	0.161	**0.281**	0.181	0.265	**0.331**	0.277
shuttle[1]	**0.997**	0.993	**0.997**	**0.878**	0.764	0.875	**0.857**	0.712	0.855
kdd99[1]	0.998	0.994	**0.998**	0.706	0.677	**0.719**	0.668	0.660	**0.681**
Best score in	2/11	4/11	7/11	2/11	2/11	7/11	2/11	2/11	8/11

Note: Sta, Dyn, and AHBOS refer to results for static bin-width, dynamic bin-width, and AHBOS, respectively. Superscripts on data names give the sources, (1) are from Goldstein (2015) and (2) are from Rayana (2016).

For the F-1 score, the AHBOS and static bin-width HBOS were found significantly different (p-value 0.028).

The performance metrics obtained from the real data study agree with the simulation results, although the difference was not as sharp. This can be attributed to the diverse characteristics of the real datasets, such as variations in sample size, number of variables, anomaly percentage, and the presence of gaps.

Both simulations and real-life data applications confirm that the proposed method substantially enhances performance. The simulation findings demonstrate that AHBOS surpasses the HBOS approach in all performance metrics, particularly outperforming in the PR-AUC measure. With few exceptions, the real-life data study supports the results obtained from the simulation. The new approach demonstrated that it could compete well with dynamic bin-width HBOS when dealing with heavy-tailed or unknown distributions.

CHAPTER THREE
THE BOOTSTRAP

This chapter includes basic principles for constructing confidence intervals with bootstrap.

3.1 Definition

The term ”bootstrap” originates from the famous phrase ”pull oneself up by one’s bootstraps” in the 19th century. It describes the situation of accomplishing something without relying on others. In statistics, bootstrap is one of the most popular resampling methods which was introduced by Efron (1979). A vast number of developments have been done by including Efron (1981, 1982) and a monograph published by Tibshirani & Efron (1993). Hall (2012) and Shao & Tu (1995) conducted comprehensive studies on bootstrap methods within a theoretical framework. An excellent handbook for practitioners published by Davison & Hinkley (1997) including various applications and guides. Another guidebook for practitioners including the historical background of resampling methods was published by Chernick (2011).

The popularity of bootstrap has increased in the last decades thanks to technological improvements. It is a great tool for estimating standard error, confidence intervals, etc (Tibshirani & Efron, 1993). Resampling the original data with replacement to estimate the population distribution is a common strategy in bootstrap methods. This procedure can be performed using either parametric or non-parametric bootstrap methods. However, it is often used when the distribution of the population is unknown, and the only information available comes from the sample.

The ordinary bootstrap method uses resampling with a replacement while giving equal selection probabilities to each observation. Constructing bootstrap samples is

straightforward. Suppose we have a univariate sample of *n* observations, $x = (x_1, x_2, ..., x_n)$. Resample with replacement of size n generates a bootstrap sample, $x^* = (x_1^*, x_2^*, ..., x_n^*)$. The "*" notation indicates that the sample is generated via bootstrap. This process is repeated *B* times to construct a confidence interval, estimate standard error, etc.

3.2 Bootstrap Confidence Interval

Bootstrap confidence intervals are a powerful and flexible method used to calculate confidence intervals in statistical analysis. Bootstrap approaches offer significant benefits, particularly in situations where the population distribution is unknown and in analyses with limited sample sizes. Let us suppose that x_i is a random sample from a population with an unknown probability distribution. Let θ represent the population parameter of interest, and $\hat{\theta}$ denote the estimate derived from the sample. The bootstrap approach, introduced by Efron (1979), is designed to estimate the standard error of $\hat{\theta}$ in situations when the standard error cannot be computed analytically or where the computation is too complex.

The statistic of interest $\hat{\theta}^*$ is calculated from each sample x^*. The distribution of these statistics forms the bootstrap distribution. The calculated bootstrap distribution is used to estimate the standard error and bias of the statistics and to determine the shape, scale, and location of the sampling distribution. The steps of this process are as follows (Yildiztepe & Özdemir, 2013):

1. Let $\{x_1, x_2, \ldots, x_n\}$ be a sample of size n selected from a population with an unknown probability distribution. From this sample, B random samples of size n are drawn with replacement.

2. The statistic of interest is calculated from each bootstrap sample to form the bootstrap distribution.

$$\hat{\theta}_1^*, \hat{\theta}_2^*, \ldots, \hat{\theta}_B^*$$

3. The standard error of $\hat{\theta}$ is estimated by

$$\hat{SE}_{\text{boot}} = \sqrt{\frac{1}{B-1}\sum_{b=1}^{B}\left(\hat{\theta}_b^* - \hat{\theta}^*(\cdot)\right)^2} \tag{3.1}$$

where $\hat{\theta}_b^*$ is the statistic calculated from the b-th bootstrap sample and

$$\hat{\theta}^*(\cdot) = \frac{1}{B}\sum_{b=1}^{B}\hat{\theta}_b^*$$

4. The bias of $\hat{\theta}$ is estimated by $\hat{B}^*_{\text{bias}} = \hat{\theta}^*(\cdot) - \hat{\theta}$.

This approach is called the non-parametric bootstrap. In the parametric bootstrap approach, bootstrap samples are drawn from the parametric model, not by resampling the original sample data with replacement. It has been stated that setting B to 200 is sufficient for bootstrap estimation of the standard error, but it is necessary to draw between 1000 and 2000 samples for bootstrap confidence intervals (Tibshirani & Efron, 1993). Nowadays, there are many methods proposed for estimating confidence intervals with bootstrap; the four most commonly used methods are included in this study.

3.2.1 Normal-Theory Intervals

The Normal-Theory intervals are used under the assumption that the standard error of the statistic of interest is unknown but the sampling distribution is normally distributed. In this approach, the bootstrap method provides an estimate of the standard error. Two-sided percent $(1-\alpha)$ bootstrap standard normal confidence interval for θ:

$$\left(\hat{\theta} - \hat{\text{bias}}^*\right) \pm z_{1-\frac{\alpha}{2}}\hat{se}^*_{\text{boot}} \tag{3.2}$$

where $z_{1-\frac{\alpha}{2}}$ is the z-value corresponding to the $1-\frac{\alpha}{2}$ area under the standard normal distribution. This method is not widely preferred due to the underlying assumptions, and in practice, the percentile approach is often used instead.

3.2.2 Percentile Intervals

In this method, the limits of the confidence interval are determined by the bootstrap distribution of $\hat{\theta}$. The steps to calculate the two-sided $(1-\alpha)$ percentile bootstrap confidence interval for $\hat{\theta}$ are as follows:

1. Arrange the estimated values $(\hat{\theta}_1^*, \hat{\theta}_2^*, \ldots, \hat{\theta}_B^*)$ from the bootstrap samples in ascending order, $\hat{\theta}_{(1)}^* < \hat{\theta}_{(2)}^* < \ldots < \hat{\theta}_{(B)}^*$.
2. For the $(1-\alpha)$ confidence interval, let $l = B(\alpha/2)$. Let l the smallest $\hat{\theta}^*$ value be taken as the lower limit, and the $u = (B-l)$ is the largest $\hat{\theta}^*$ value is taken as the upper limit. If l is not an integer, then l is rounded to the nearest integer, specifically the greatest integer (Wilcox, 2017).
3. The limits of the two-sided $(1-\alpha)$ percentile bootstrap confidence interval for θ are given by $(\hat{\theta}_{(l)}^*, \hat{\theta}_{(u)}^*)$.

3.2.3 Bootstrap-t Intervals

When assumptions are met, a z or t table is needed in parametric methods to find confidence intervals. In this method, a table appropriate to the data is created and used to determine the confidence interval. The steps are:

1. Calculate the t^* values for each bootstrap sample:

$$t_b^* = \frac{\hat{\theta}_b^* - \hat{\theta}}{\hat{se}_b^*} \quad b = 1, 2, \ldots, B \tag{3.3}$$

2. Arrange t^* in ascending order ($t^*_{(1)}$, $t^*_{(2)}$,...$t^*_{(B)}$) .

3. Let $l = \alpha\beta/2$ where l is rounded to the nearest integer, and let $u = B - l$

4. The limits of the two-sided $(1-\alpha)$ bootstrap-t confidence interval for θ are given by $\left(\hat{\theta} - \hat{t}^*_l \, \hat{se}^*_{boot}, \hat{\theta} - \hat{t}^*_u \, \hat{se}^*_{boot}\right)$

3.2.4 Bias-corrected and Accelerated Bootstrap Intervals

The bias-corrected accelerated (BCa) bootstrap method, which was developed to eliminate the deficiencies of the percentage bootstrap method seen both in theory and practice, was proposed by (Efron, 1987). This method used the correction factors z_0 and $\hat{a}$. It these factors are both 0, then BCa confidence interval is the same as that found by the percentile bootstrap. Otherwise, the two-sided $(1-\alpha)$ BCa confidence interval for θ is denoted by $(\hat{\theta}^*_{\alpha_1}, \hat{\theta}^*_{\alpha_2})$. Here, α_1 and α_2 are obtained as follows:

$$\alpha_1 = \Phi\left(\hat{z_0} + \frac{\hat{z_0} + z_{1-\frac{\alpha}{2}}}{1 - \hat{a}(\hat{z_0} + z_{1-\frac{\alpha}{2}})}\right) \tag{3.4}$$

$$\alpha_2 = \Phi\left(\hat{z_0} + \frac{\hat{z_0} + z_{\frac{\alpha}{2}}}{1 - \hat{a}(\hat{z_0} + z_{\frac{\alpha}{2}})}\right) \tag{3.5}$$

where $\Phi(.)$ is the standard normal cumulative distribution function, and $z_{1-\frac{\alpha}{2}}$ is the z-value corresponding to the $1 - \frac{\alpha}{2}$ area under the standard normal distribution. The bias correction value z_0 is found as follows:

$$\hat{z_0} = \Phi^{-1}\left(\frac{\#(\hat{\theta}^* < \hat{\theta})}{B}\right) \tag{3.6}$$

Here, $\Phi^{-1}(.)$ is the inverse of the standard normal cumulative distribution function. The acceleration value $\hat{a}$, which is a measure of skewness, is calculated using the jackknife estimate as follows:

$$\hat{a} = \frac{\sum_{i=1}^{n}(\hat{\theta}_{(.)} - \hat{\theta}_{(-i)})^3}{6\left(\sum_{i=1}^{n}(\hat{\theta}_{(.)} - \hat{\theta}_{(-i)})^2\right)^{3/2}} \tag{3.7}$$

Here, $\hat{\theta}_{(-i)}$ denotes the value of $\hat{\theta}$ calculated by excluding the i-th observation, and $\hat{\theta}_{(.)}$ denotes the mean of these $\hat{\theta}_{(-i)}$ values.

3.3 Bootstrap in Regression Analysis

Regression models play a vital role in applied statistics, acting as a fundamental tool for understanding the relationships between variables. The application of regression is extensive and may be observed in almost every domain of research. Least squares is a conventional approach for describing the relation between variables. Consider a linear model.

$$y = X\beta + \varepsilon \tag{3.8}$$

y is a $n \times 1$ vector containing the response, X is an $n \times p$ matrix containing predictor including the intercept, and β is a $p \times 1$ matrix containing unknown regression coefficients. Finally, ε is a $n \times 1$ vector of i.i.d. errors and ideally follows normal distribution with constant variance, $E(\varepsilon) = 0$ and $V(\varepsilon) = \sigma^2$. The general idea is to find least square estimators, $\hat{\beta}$, that minimize the sum of the squared residuals (ε^2):

$$min \sum_{i=1}^{n} \varepsilon_i^2 \tag{3.9}$$

and the least squares estimation of β is:

$$\hat{\beta} = (X^T X)^{-1} X^T y \tag{3.10}$$

after solving the Equation 3.9 the fitted values and residuals are computed as $\hat{y} = X\hat{\beta}$ and $\hat{\varepsilon} = y - \hat{y}$, respectively.

In regression, fixed-x resampling and random-x resampling approaches can be used to generate bootstrap samples. Fixed-x resampling is done by resampling residuals and random-x resampling is done by resampling the data pairs. Fox (2015) suggests using random-x resampling if an incorrect model is fitted. However, fixed-x resampling is more efficient if the correct model has been fitted.

3.3.1 Fixed-x Resampling

Fixed-x resampling is also known as residual resampling. Assume that we have a pair of observations $(y_i, x_{i1}, x_{i2}, ..., x_{ip})$, $(i = 1, 2, .., n)$ and p is the number of predictors. After fitting the model with LS, fixed-x resampling starts with obtaining the bootstrapped residuals ($\hat{\varepsilon}_i^*$). Then, bootstrapped residuals are attached to the fitted values ($\hat{y}_i^* = \hat{y}_i + \hat{\varepsilon}_i^*$) to get the new bootstrap response ($\hat{y}_i^*$). Steps of fixed-x resampling is

1. First, obtain fitted values, $\hat{y}_i$ and residuals, $\hat{\varepsilon}$.
2. Draw bootstrap residuals, ε^*.
3. Then obtain bootstrapped response values by attaching bootstrapped residuals to fitted values, $\hat{y}_i^* = \hat{y}_i + \hat{\varepsilon}_i^*$.
4. Compute $\hat{\beta}^*$ via regressing bootstrapped y_i^* on fixed x values.
5. Repeat $Step - 3$ and $Step - 4$ for B times to get $\hat{\beta}_1^*, ..., \hat{\beta}_B^*$.
6. Put $\hat{\beta}_1^*, ..., \hat{\beta}_B^*$ in ascending order yielding $\hat{\beta}_{(1)}^*, ..., \hat{\beta}_{(B)}^*$.
7. Set $l = \alpha B/2$, round l to the nearest integer, and let $u = B - l$. Approximate $(1 - \alpha)$ confidence interval for β is $(\hat{\beta}_{(l)}^*, \hat{\beta}_{(u)}^*)$.

3.3.2 Random-x Resampling

Random-x resampling is also known as pair resampling. Assume that we have a pair of observations $z_i = (y_i, x_{i1}, x_{i2}, ..., x_{ip})$, $(i = 1, 2, .., n)$ and p is the number of predictors. The algorithm is similar to the bootstrapping univariate sample except that we bootstrap the regression pairs. Steps of random-x resampling is

1. Draw bootstrap sample pairs $z_1^*, z_2^*, .., z_B^*$, from the original cases z_i.
2. Compute $\hat{\beta}^*$ using bootstrap pairs.
3. Repeat $Step - 1$ and $Step - 2$ for B times to get $\hat{\beta}_1^*, ..., \hat{\beta}_B^*$.
4. Put $\hat{\beta}_1^*, ..., \hat{\beta}_B^*$ in ascending order yielding $\hat{\beta}_{(1)}^*, ..., \hat{\beta}_{(B)}^*$.
5. Set $l = \alpha B/2$, round l to the nearest integer, and let $u = B - l$. Approximate $(1 - \alpha)$ confidence interval for β is $(\hat{\beta}_{(l)}^*, \hat{\beta}_{(u)}^*)$.

The Random-x approach is not based on the assumption of constant variance or linearity, which makes it robust to some problems but may result in lower efficiency. This approach employs random sampling of design values, which can cause little fluctuation but is typically insignificant in large datasets (Davison & Hinkley, 1997).

CHAPTER FOUR
ROBUST BOOTSTRAP METHODS

Outliers in regression can occur in x-space (horizontal) and y-space (vertical). Figure 4.1 gives an illustration of various outliers in simple linear regression. Normal data points are depicted within an ellipse. Point A is a vertical outlier that diverges from the bulk of points in the y-axis. Points B, C and D are horizontal outliers that deviate from the majority of the points in the x-space called leverage points. Also, points B and C are commonly known as bad leverage (BL) points. Although D is an outlier, it follows the general trend of the data points that are called good leverage points. Point E is a typical point in the y-space but also a residual outlier (Alma, 2011).

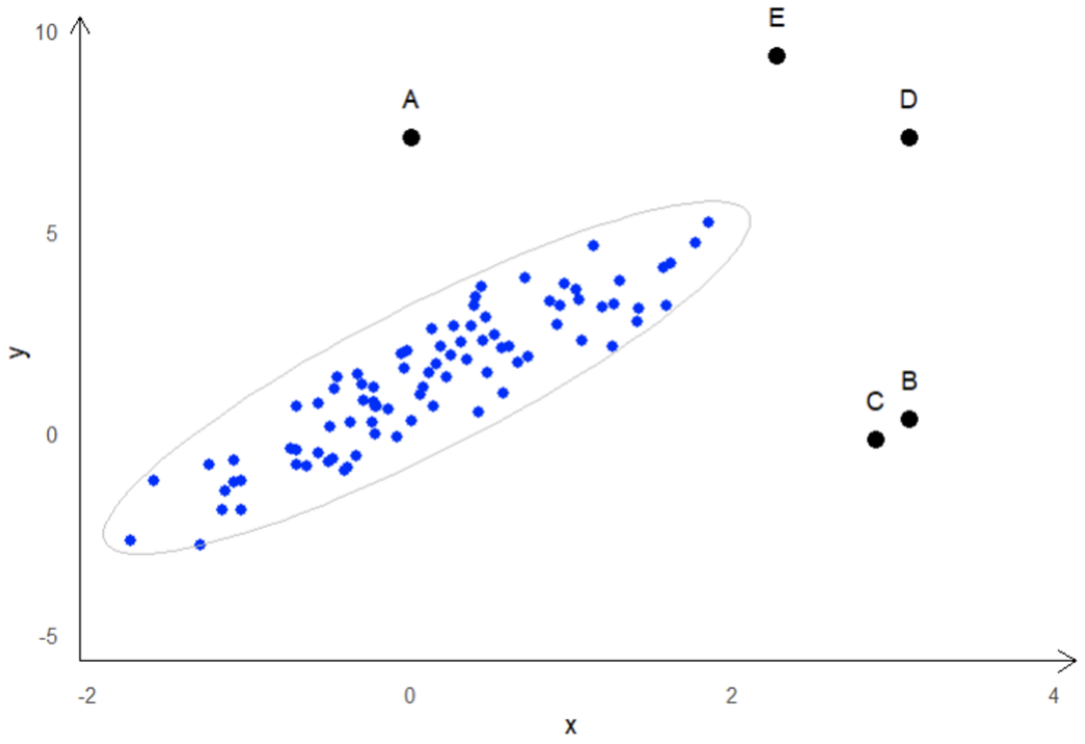


Figure 4.1 Scatter plot of outliers in simple linear regression

LS is frequently employed in regression analysis due to its conventional and easy nature when the assumptions are met. However, relying only on LS can be misleading, especially when there are small deviations from the underlying

assumptions. Outliers, defined as data points that deviate significantly from the majority of data, pose a difficulty in regression analysis and have the potential to impact the reliability of predictions.

In the presence of outliers, robust methods are preferred over classical approaches due to their ability to handle the effects of outliers. In regression modeling, robust estimators' asymptotic distribution is usually examined under specific conditions, such as when the data displays symmetry or elliptical distribution (Salibián-Barrera et al., 2008). For example, the errors are assumed to be symmetric, which does not usually hold because outliers do not need to appear symmetrical. Moreover, in some cases, it may not be feasible to derive the exact distribution of robust estimators and even if it is derivable, convergence may be slow (Amado et al., 2014).

Bootstrapping robust estimators become popular since it solves some of the problems associated with the problems mentioned. Also, bootstrapping robust estimators may have satisfying small sample properties (Wilcox, 2017). Despite providing practical solutions in many scenarios, additional issues may arise, particularly in the presence of outliers. Two main problems associated with bootstrapping robust estimators are:

- Outliers may aggregate in bootstrap samples.
- Calculating robust estimators for each bootstrap can be expensive, especially considering that some robust estimators require iterative processes.

Singh (1998) gives a simple example of how bootstrap with a robust estimator fails. Consider a sample with sample size 20 which includes 1 outlier on the upper side. The probability of bootstrap samples containing at most 1 outlier is $P(Bin(20, 0.05) \leq 1) = 0.736$. If one uses a 10 % trimmed mean, it will not be enough to deal with 26.4% of the bootstrap samples.

The studies related to robust bootstrap methods began in the early 1980s. Ghosh et al. (1984) showed that variance obtained by bootstrap sample median converges to

asymptotic variance when a tail condition is applied to the underlying distribution. Athreya (1987) implied that classical bootstrap may fail for heavy-tailed distributions. Shao (1990) also demonstrated the poor performance of bootstrap variance estimation under heavy-tailed distributions. Shao (1992) again pointed out the inconsistency of bootstrap in some cases and proposed a modified bootstrap based on truncation. Stromberg (1997) showed that the sample covariance matrix obtained via bootstrap may have a lower breakdown point than the estimator itself in the presence of outliers. In simpler terms, even if it is a small possibility, some bootstrap samples may contain a percentage of outliers that exceeds the breakdown point of an estimator. Stromberg (1997) suggested using either deleted-*d* jackknife or employing bootstrap with a more robust measure of variability as alternatives. Singh (1998) suggested bootstrapping from the winsorized sample instead of the original sample for certain *L* and *M* estimators in order to increase the robustness of bootstrap confidence intervals.

Salibian-Barrera & Zamar (2002) suggested a robust bootstrap method based on the re-weighted representation of the MM-regression estimates that is numerically stable and resistant to outliers. Also, the suggested approach drastically reduces the computation time since instead of computing a robust estimate in each bootstrap sample a weighted representation of an estimator is used. Then, this approach is generalized under the name of FRB Salibián-Barrera et al. (2008).

Most of the methods in the robust bootstrap concept are based on using robust estimators via bootstrap. Amado & Pires (2004) and Amado et al. (2014) used a different approach to robustify the bootstrap process. In order to protect bootstrap from the harmful effects of outliers they proposed IFB. IFB is based on re-weighting the bootstrap probabilities so that outlying observations receive lower probabilities while the bulk of the data remains the same. Norazan et al. (2009) proposed a similar method that is called WBP in particular for linear regression.

We conducted an experiment to show the difference between the computation time of bootstrapping robust estimators and robust bootstrap methods. The model was set as 5 explanatory variables with a sample size of 100 and 10000 bootstrap samples were

generated. The bootstrap estimates obtained by bootstrapping robust estimators took 2-3 minutes. However, a dramatic decrease in computation time was observed where robust bootstrap methods finished the task in 3-8 seconds.

4.1 Effect of Outliers in Bootstrap

The existence of outliers is a common problem in statistics. Using bootstrap with classical estimators does not solve the outlier problem by itself and the situation might get worse when using data with outliers.

A simulation study is conducted to show the outlier effect in bootstrap. The simulation study is conducted with R statistical software program. All simulations are done with 10000 Monte Carlo repetitions and 1500 bootstrap samples. The Mean Square Error (MSE) results are obtained with bootstrap and the performance of Bootstrap for parameter estimation is evaluated with clean data and data with outliers.

Table 4.1 MSE results for standard error estimation of the mean

	Sample Sizes				
Outlier Condition	**10**	**20**	**30**	**50**	**100**
Without Outlier	0.0055	0.0013	0.0006	0.0002	0.00005
With Outlier	0.45	0.23	0.15	0.09	0.046
With Outlier/Without Outlier	**82.32**	**173.10**	**254.05**	**428.5**	**820.73**

Table 4.1 shows MSE results for the estimation of the mean. The data is generated from Standard Normal Distribution (Without Outlier) and 10% of the data is replaced with 10. MSE values for the with outlier case are 82.32, 173.10, 254.05, 428.5, and 820.73 times the Without Outlier case for the respective sample sizes.

Table 4.2 MSE results for slope coefficient estimation

	Sample Sizes				
Outlier Condition	**10**	**20**	**30**	**50**	**100**
Without Outlier	0.146	0.058	0.037	0.021	0.0098
With Outlier	1.52	0.625	0.395	0.22	0.11
With Outlier/Without Outlier	**10.36**	**10.82**	**10.61**	**10.48**	**10.63**

Table 4.2 shows the MSE results for slope coefficient estimation in the linear regression case. Errors are generated from Standard Normal Distribution (Without Outlier) and 10% of the data is replaced with 10. MSE values for the with outlier case are nearly 10 times the without outlier case for all sample sizes.

4.2 Robust Regression Methods

Consider x_i and β the p dimensional vectors with coordinates $(x_{i1}, ..., x_{ip})$ and $(\beta_1, ..., \beta_p)$. the model can be written as

$$y_i = x_i^T \beta + \varepsilon_i \tag{4.1}$$

where, x^T is the transpose of x. The fitted values are $\hat{y}_i = x_i^T \beta$ and the residuals $\hat{\varepsilon}_i = y_i - \hat{y}_i$ $(i = 1, 2, ..., n)$. In the least squares, the aim is to minimize the sum of the squared residuals. However, LS might yield accurate estimates in the presence of outliers and one of the most straightforward ways to solve this problem is replacing the classical estimator with robust estimators. To assess the performance of robust estimators two common criteria are used that is efficiency and breakdown point (Donoho & Huber, 1983). The breakdown point refers to the minimum level of contamination that can cause an estimator to produce arbitrarily large values.

The highest breakdown point an estimator can achieve is 50%, indicating that the estimator can tolerate data sets where up to half of the observations are outliers. Conversely, the lowest breakdown point an estimator can have is 1/n or 0/n, meaning that even a single outlier can make the estimator unusable. The relative efficiency of a robust estimator compared to LS when the underlying assumptions are true is what defines its efficiency. This concept quantifies how well the robust estimator performs relative to LS under ideal conditions.

Donoho & Huber (1983) defines finite sample breakdown point as follows: Consider $z_i = (x_{i1}, x_{i2}..., x_{ip}, y_i)$ and let T(z) is the estimator of β, representing the vector of

regression coefficients in the model. Assume that z' is contaminated data such that m corrupted data is replaced from the original data. The maximum bias is defined as

$$b(m; z, T) = sup \, \|T(z') - T(z)\| \tag{4.2}$$

and the finite sample breakdown point is

$$\gamma^*(z, T) = min \left\{ \frac{m}{n}; bias(m; T, z) = \infty \right\} \tag{4.3}$$

In simple terms, the estimator T can be greatly affected by even a small fraction of corrupted data, leading to a substantial deviation from T(Z).

In this thesis, We only presented the robust regression estimators LMS, LTS and MM, which are used in the robust bootstrap procedure.

4.2.1 Least Median of Squares

Least median of squares (LMS) estimator was first proposed by Hampel (1975) and further improved by Rousseeuw (1984). Instead of minimizing the sum of the squared residuals, LMS is based on minimizing

$$\text{MED}(r_1^2, \ldots, r_n^2) \tag{4.4}$$

the median of the squared residuals. LMS estimate achieves a high breakdown point that is approximately 0.5. Despite that, its efficiency is relatively low compared to the least square estimate (Wilcox, 2017). Another drawback of the LMS estimator is that its converge rate $(n^{1/3})$ is relatively low (Rousseeuw, 1984). Although the LMS estimator has certain undesirable characteristics, it is frequently recommended as a diagnostic tool or a first fit for data.

4.2.2 Least Trimmed Squares

Another often used robust regression estimator is the least trimmed squares (LTS) estimate that is suggested by Rousseeuw (1984). LTS estimator minimizes

$$min \sum_{i=1}^{h} r_{(i)}^2 \tag{4.5}$$

where $r_(1)^2 \leq \cdots \leq r_(h)^2$ are the ordered squared residuals.

The value of h is frequently chosen as $h = [n/2] + [(p+1)/2]$, where n represents the total number of observations, and p denotes the number of predictor variables. For $h = [n/2] + 1$, the LTS estimate has the same breakdown point as LMS estimate which is 0.5. The converge rate of the LTS estimator is $n^{1/2}$ (Rousseeuw, 1984). One of the downsides of the LTS estimate is that it has a relatively poor efficiency, which is approximately 0.08 in comparison to LS estimates (Stromberg et al., 2000).

4.2.3 MM Regression

The MM estimator was proposed by Yohai (1987), aiming to achieve a high breakdown point and normal efficiency. This procedure can be explained by three in three steps.

First, compute the initial estimate $\hat{\beta}_0$ via a high breakdown point and low-efficiency estimate. Following the completion of the first estimate, the second stage involves the computation of an S-estimate of the errors scale by using the initial estimate. The third and final stage involves the computation of an MM-estimate of the regression parameters.

For the scale estimate ($\hat{\sigma}$), use the S-estimator

$$\frac{1}{n}\sum_{i=1}^{n}\rho_0\left(\frac{r_i(\hat{\beta})}{\hat{\sigma}}\right)=0.5 \tag{4.6}$$

and minimize the MM-estimate based on scale S-estimate $\hat{\sigma}_0$

$$\sum_{i=1}^{n}\rho\left(\frac{r_i(\hat{\beta}_{\text{MM}})}{\hat{\sigma}_0}\right) \tag{4.7}$$

One of the popular choices for the loss functions ρ and ρ_0 is that Tukey's bisquare

$$\rho(u)=\begin{cases}1-\left(1-\left(\frac{u}{c}\right)^2\right)^3 & \text{if } |u|\le c\\ 1 & \text{if } |u|>c\end{cases} \tag{4.8}$$

The constant c for S-estimator and the MM-estimator is set as 1.548 and 4.685 which provides a 50% breakdown point and 0.95 efficiency.

4.3 Robust Bootstrap Methods in Regression

This section comprises robust bootstrap methods in this study.

4.3.1 Influence Function Bootstrap

IFB might be the first method that re-weights bootstrap selecting probabilities to protect bootstrap procedure from the harmful effects of outliers (Amado & Pires, 2004). They aimed to give less selection probabilities for outlying observations meanwhile keeping these probabilities the same for the bulk of the data. Instead of omitting outlying observations directly, they re-arrange probabilities between 0 and 1. We can define this process in two steps. In the first step, bootstrap selection probabilities are calculated, followed by the usual bootstrap process. In IFB,

weighting functions are determined by the standardized influence function (SIF) (Amado et al., 2014).

We followed the notation used by Zoubir et al. (2018). Define $F(\Omega)$ as a parametric model and $\Omega = (\beta, \tau)$ is the parameter of estimate and nuisance, respectively. The standardized influence function is:

$$\text{SIF}\left(x, \hat{\beta}^{nr}, F(\Omega)\right) = \left[IF\left(x, \hat{\beta}^{nr}, F(\Omega)\right)^T V\left(\hat{\beta}^{nr}, F(\Omega)\right)^{-1} IF\left(x, \hat{\beta}^{nr}, F(\Omega)\right)\right]^{1/2} \quad (4.9)$$

Then, the robust standardized empirical influence function $RESIF(x, \hat{\beta}^{nr}, \Omega^r)$ is obtained by substituting robust estimates Ω^r of Ω (4.9).

Assume a simple linear regression model where the robust estimator is MM-estimator. The RESIF is defined as:

$$RESIF(\hat{r_i}/\hat{\sigma}) = |y_i - x_i\hat{\beta}|/\hat{\sigma}, \quad i = 1, 2, ..., n \quad (4.10)$$

where $\hat{\sigma}$ is the scale estimator and $r_i = y_i - x_i\beta, i = 1, 2, .., n$.

Finally, the resampling probabilities in bootstrap are computed as below,

1. Compute $RESIF(\hat{r_i}/\hat{\sigma})$ and then compute the weights, w_i, according to $w_i = I_{[0,c]}\left(|(RESIF(\hat{r_i}/\hat{\sigma})|\right) + \psi\left(c, |(RESIF(\hat{r_i}/\hat{\sigma})|\right) \times I_{[c,+\infty]}\left(|(RESIF(\hat{r_i}/\hat{\sigma})|\right)$, $i = 1, 2, ..., n$. The I is indicator function of set A, $c > 0$ is a tuning constant and ψ is a non-negative function that satisfies:
   Condition 1: $lim_{t\to+\infty}\, t^2\psi(c, t) = 0$ (for fixed c)
   Condition 2: $\frac{\partial}{\partial t}\psi(c, t)\big|_{t=c} = 0$
2. Draw bootstrap samples with probabilities $p = (p_1, p_2, ..., p_n)$, where $p_i = \frac{w_i}{\sum_{j=1}^{n} w_j}$

The bootstrap samples are drawn with probabilities specified at *Step-2*. The tuning constant c defines the balance between efficiency and robustness and ψ is the *t*-distribution and its limiting form (Amado & Pires, 2004).

$$\psi(x;c,d,\gamma) = \begin{cases} \left[1+\frac{(x-c)^2}{\gamma d^2}\right]^{-\frac{\gamma+1}{2}} & 0<\gamma<\infty \\ exp\left[\frac{(x-c)^2}{2d^2}\right] & \gamma=\infty \end{cases} \tag{4.11}$$

The parameters c, d and γ is defined as location, scale, and shape, respectively. Condition 1 is met if γ is greater than 1, and condition 2 is met when the location parameter is set as the tuning constant. To simplify the Equation (4.11), Amado & Pires (2004) set the parameters c and d as $\sqrt{\chi^2_{1;0.99}}$.

4.3.2 Weighted Bootstrap with Probability

A Similar approach to the IFB is suggested by Norazan et al. (2009). They aimed to protect the bootstrap procedure against outliers by re-assigning new bootstrap selection probabilities. WBP begins with fitting the model with the LMS to get residuals. Here, r_i represents the LMS residuals, $u_i = r_i/MAD(r_i)$ represents the standardized LMS residuals and MAD stands for median absolute deviation $(i = 1,\ldots,n)$. Afterwards, Hampel's ψ function (4.12) is used to obtain bootstrap probabilities, $p_i = \psi(u_i)/u_i$.

$$\psi_{hampel}(u) = \begin{cases} u & |u| \le a \\ asign(u) & a<|u|\le b \\ \frac{a(c-|u|)}{(c-b)sign(u)} & b<|u|\le c \\ 0 & c<|u| \end{cases} \tag{4.12}$$

Potential outliers $(p_i = 0)$ will get 0 weight and thus will not be included in the bootstrap samples. The tuning constants are set as, $a = 3, b = 4, c = 8$. The steps of the WBP approach are given below,

1. Apply LMS to obtain LMS residuals.
2. Use Hampel's ψ function on LMS residuals and obtain $p_i = (p_1, \ldots, p_n)$, $i = 1, 2, ..., n$.
3. Draw bootstrap sample pairs $z_1^*, z_2^*, .., z_B^*$, from the original cases z_i with probabilities computed at $Step - 2$. .
4. Compute $\hat{\beta}^*$ using bootstrap pairs,
5. Repeat $Step - 3$ and $Step - 4$ for B times to get $\hat{\beta}_1^*, ..., \hat{\beta}_B^*$,
6. Put $\hat{\beta}_1^*, ..., \hat{\beta}_B^*$ in ascending order yielding $\hat{\beta}_{(1)}^*, ..., \hat{\beta}_{(B)}^*$,
7. Set $l = \alpha B/2$, round l to the nearest integer, and let $u = B - l$. Approximate $(1 - \alpha)$ confidence interval for β is $(\hat{\beta}_{(l)}^*, \hat{\beta}_{(u)}^*)$.

4.3.3 Fast and Robust Bootstrap

Salibian-Barrera & Zamar (2002) proposed a fast and robust bootstrap algorithm to deal with outliers in the linear regression models. Consider any estimate $\hat{\beta}$ that can be represented as a solution of a fixed-point equation that includes a large class of estimators, such as the M, MM, or τ-estimators. A significant reduction in computational complexity can be achieved by using the FRB which does not require recomputing the estimates in each stage. It is applicable for estimators $\hat{\beta}$ that can be expressed as a solution to a system of smooth fixed-point estimation equations. Instead of computing $\hat{\beta}^*$ as previously, the following approximation is used:

$$\hat{\beta}^{1^*} = g(\hat{\beta}; x^*) \tag{4.13}$$

Because the distribution of $\hat{\beta}^{1^*}$ typically underestimates the sampling variability of $\hat{\beta}$ a linear correction term based on a Taylor-series approximation of the function g is applied as follows:

$$\hat{\beta}^{R^*} = \hat{\beta} + \left[I - \nabla g(\hat{\beta}; x)\right]^{-1} \left(\hat{\beta}^{1^*} - \hat{\beta}\right) \tag{4.14}$$

Where $\nabla g(\hat{\beta}; x) \in \mathbb{R}^{p \times p}$ is the matrix of partial derivatives $\hat{\beta}$. The FRB method has the following advantages: first, it is fast to compute because the initial estimate $\hat{\beta}$ is computed only once. The estimate $\hat{\beta}^{R^*}$ is not only much faster to compute than $\hat{\beta}^*$, it is also more robust because outlying points carry their small weights into the bootstrap recalculations. FRB uses MM-estimator (Subsection 4.2.3) with the same tuning parameters. Detailed definitions can be found in Salibian-Barrera (2000) and Salibian-Barrera & Zamar (2002).

4.4 Weighted Fast and Robust Bootstrap

The performance of robust estimators might differ based on the individual properties of the data, therefore it is impractical to expect one to always outperform the other. A number of factors can affect how well certain robust estimators perform, including the data distribution, the sample size, the existence and kind of outliers, and the presence or absence of outliers themselves. Rather than assuming that a single robust estimator will always perform better than the others, it is crucial to take into account the specific circumstances and context in which the estimators are being used.

Ensemble methods in machine learning combine several methods to improve accuracy instead of using a single technique. For example, one estimator can have a higher breakdown point than the other one however, its efficiency might be lower than the other. In a similar context, a new robust bootstrap method is introduced by combining IFB and FRB. The new approach uses two robust estimators, one in IFB and one in FRB, where the aim is to take advantage of these robust estimators instead of using one.

The basic idea of IFB is reweighting the bootstrap selection probabilities to protect the classical estimator from the harmful effects of outliers. In addition, FRB reduces the weights of outlying points in the bootstrap recalculations process. The new approach is based on reweighting the bootstrap selection probabilities prior to FRB to increase the robustness of the procedure.

The new approach, weighted fast and robust bootstrap (WFRB), uses a two-step procedure. In the first step, we apply IFB to find bootstrap selection probabilities (weights). Since FRB uses ordinary bootstrap procedure (uniform weighting), we implement the pre-found weights computed via IFB.

1. Find RESIF (Equation 4.10) and Use Hampel's ψ function (Equation 4.12) on scaled residuals then obtain $p_i = (p_1, \ldots, p_n)$, $i = 1, 2, ..., n$.
2. Draw bootstrap samples with probabilities computed at $Step - 1$.
3. Apply FRB procedure with bootstrap samples and compute $\hat{\beta}^*$.
4. Repeat $Step - 2$ and $Step - 3$ for B times to get $\hat{\beta}^*_1, ..., \hat{\beta}^*_B$.
5. Put $\hat{\beta}^*_1, ..., \hat{\beta}^*_B$ in ascending order yielding $\hat{\beta}^*_{(1)}, ..., \hat{\beta}^*_{(B)}$.
6. Set $l = \alpha B/2$, round l to the nearest integer, and let $u = B - l$. Approximate $(1 - \alpha)$ confidence interval for β is $(\hat{\beta}^*_{(l)}, \hat{\beta}^*_{(u\,)})$.

FRB is an asymptotically consistent method. By changing the bootstrap selection probabilities, we obtain more robust estimates of the parameter of interest. However, this comes at the cost of underestimating the variance, thus yielding shorter confidence intervals and reduced coverage. Since FRB uses MM-estimator in general, we used LMS and LTS to weigh bootstrap selection probabilities in this study. The steps of the WFRB procedure is:

4.5 A New Approach to Use Good Leverage Points in Bootstrapping Regression Models

The impact of good leverage points in regression has not been well recognized in the literature since they rarely occur. The aim was usually diagnosing and treating the effects of outliers. Good leverage points can improve the precision of estimated

regression coefficients and lower their standard error. The use of good leverage points of bootstrap has not been investigated as far as we know.

As we mentioned earlier, bootstrap resamples might have a higher contamination level than the original sample in the presence of outliers. Some of the splitting-based robust bootstrap techniques aimed to solve this problem by separating the outliers from the rest of the data according to the breakdown point of the estimator (Alamgir & Ali, 2013, Vlaski et al., 2014). This data-splitting approach is based on controlling the outlier percentage within bootstrap samples. The algorithm splits the dataset into normal data points and outliers, which can be called strata bootstrap, enabling the robust estimators' breakdown point to effectively handle them. Similarly, some of the bootstrap samples might not benefit from good leverage points. As a consequence, the number of good leverage points in bootstrap resamples may be fewer than those found in the original sample.

For example, consider a regression model with a sample size of 20 which includes a good leverage point, which is similar to the example given by Singh (1998) about outliers in bootstrap. The probability of bootstrap samples that do not contain a good leverage point is, P(Bin(20,0.05) < 1) = 0.358. Consequently, in such instances, the estimator may not benefit from these good leverage points. Our splitting approach can be used in conjunction with other robust bootstrap algorithms. If no good leverage points are found, the algorithm simplifies to the regular robust bootstrap techniques.

Let us say $z_i = (y_i, x_i)$ and we want to fit a simple linear regression model with response y_i and predictor x_i which forms a sample of *n* observations. Assume that we readily obtained the good leverage points and defined the pairs as z^{GL}. Then, we define the remaining observations as z^R. Here, The number of good leverage points is presented as n_{GL}, and the number of remaining points is presented as n_R where $n_{GL} + n_R = n$. Assuming that at least one good leverage point exists, the steps of the new approach are:

1. Separate the good leverage points and remaining observations, z^{GL} and z^R.

2. Preserve good leverage points, z_j^{GL} and $j = 1, ..., n_{GL}$.

3. Draw bootstrap samples from z^R, ($z_1^{R*}, z_2^{R*}, ..., z_{n_R}^{R*}$).

4. Merge the good leverage points and generated bootstrapped observations and obtain new bootstrap samples ($z_1^{GL}, ..., z_{n_{GL}}^{GL}, z_1^{R*}, z_2^{R*}, ..., z_{n_R}^{R*}$).

5. Use one of the robust regression methods on merged bootstrap samples to get $\hat{\beta}^*$.

6. Repeat $Steps\, 3-5$ and obtain $\hat{\beta}_1^*, ..., \hat{\beta}_B^*$.

7. Put $\hat{\beta}_1^*, ..., \hat{\beta}_B^*$ in ascending order yielding $\hat{\beta}_{(1)}^*, ..., \hat{\beta}_{(B)}^*$.

8. Set $l = \alpha B/2$, round l to the nearest integer, and let $u = B - l$. Approximate $(1-\alpha)$ confidence interval for β is $(\hat{\beta}_{(l)}^*, \hat{\beta}_{(u)}^*)$.

CHAPTER FIVE
SIMULATION AND REAL-LIFE DATA STUDY

In order to assess the performance of robust bootstrap methods on constructing confidence intervals, a Monte-Carlo simulation study and real data application are conducted for simple and multivariate linear regression. We compared ORB (Ordinary Bootstrap with Least Squares), FRB, IFB, WBP, and the proposed approach WFRB with the metrics explained below. Two variants of WFRB were used WFRB_{LMS} and WFRB_{LTS} with 20% trimming. The subscripts give the information about the prior estimator to compute weights before the FRB method. Bootstrap samples were drawn with random-x sampling and confidence intervals were created with the percentile bootstrap method. Given metrics assess the robustness and accuracy properties of bootstrap confidence intervals:

Bootstrap Mean Squared Error (MSE): Measures the dispersion of bootstrap estimates from the true parameter. MSE gives the average of mean squared errors obtained for the Monte Carlo loop.

Confidence Interval Length (CIL): Gives the average length of bootstrap confidence intervals. A lower CIL indicates better robustness.

Coverage Accuracy (CA): Proportion of the confidence intervals that contain the true parameter.

Various range of sample sample sizes (20,30,50,100) were used and all Monte Carlo simulations were done with 10000 replications and 1500 bootstrap samples. Nominal level for the confidence intervals was set to $\alpha = 0.05$. To assess the reliability of the methods, we employed Bradley's liberal criterion (Bradley, 1978). The criteria is calculated as $1 - \alpha \pm 0.5\alpha$. The coverage accuracies that fall within the range of (0.925,0.975) were considered to be effectively controlled for the nominal level 0.95 which were given in boldface in tables.

The whole work in the chapter is conducted via R statistical software (R Core Team, 2023). For MM and LTS estimates "Robustbase" (Maechler et al., 2021) package was

used and for LMS estimate ”MASS” package was used (Venables & Ripley, 2002).

5.1 Simulation Study

Simple and multiple linear regression models were considered in the simulation study. We followed a similar simulation plan that is used by Yu & Yao (2017). The first model is set as $Y = 1 + X + \varepsilon$, where X is generated as i.i.d from $N(0, 1)$ and ε having one of the error distributions listed below. The multiple linear regression the model is set as $Y = 1 + X_1 + X_2 + X_3 + X_4 + X_5 + \varepsilon$ where all predictors generated as i.i.d. $N(0, 1)$ and ε having one of the error distributions listed below. The ε distributions:

Case I: ε follows Standard Normal Distribution *N(0,1)*
Case II: ε follows t distribution with degrees of freedom 3
Case III: ε follows Standard Normal Distribution *N(0,1)* and %10 of error replaced by 5 for simple linear regression and 10 for multiple linear regression
Case IV: ε follows Contaminated Normal Distribution $0.9N(0, 1) + 0.1N(0, 9^2)$

In the simulation study, the cases were determined to cover various error distributions to assess the performance of robust bootstrap methods. Case I is given for the base model to measure how close the robust bootstrap estimates to the metrics were found with ORB since ORB gives the best results when the errors are normal. Case II includes the errors from a heavy-tailed distribution. Case III comprises the fixed percentage of point outliers that are aggregated on one side. Finally, the last case includes the fixed percentage of point outliers that are generated from a distribution with high variability relative to standard normal distribution.

Tables 5.1-4 and Figures 1.1-4 present the simple linear regression results. The results for multiple linear regression are given in Tables 5.5-8 and Figures 1.-5-8. Because the ORB is sensitive to outliers, it exaggerates the limits of the metrics in the figures and results in a lack of readability. Thus the results of ORB are discarded

from the figures.

Table 5.1 Comparison of robust bootstrap methods for case I: $\varepsilon \sim N(0,1)$

		ORB	**IFB**	**FRB**	**WBP**	**WFRB$_{LMS}$**	**WFRB$_{LTS}$**
n=20	MSE	0.121	0.121	0.137	0.124	0.143	0.137
	CIL	0.944	0.941	1.003	0.928	0.998	1.002
	CA	0.922	0.922	0.918	0.911	0.911	0.918
n=30	MSE	0.075	0.075	0.082	0.075	0.084	0.082
	CIL	0.744	0.742	0.782	0.735	0.779	0.782
	CA	**0.927**	**0.926**	**0.925**	0.921	0.920	**0.925**
n=50	MSE	0.043	0.043	0.046	0.043	0.047	0.046
	CIL	0.566	0.566	0.592	0.563	0.591	0.592
	CA	**0.932**	**0.934**	**0.938**	**0.933**	**0.937**	**0.938**
n=100	MSE	0.021	0.021	0.022	0.021	0.022	0.022
	CIL	0.395	0.395	0.409	0.394	0.409	0.409
	CA	**0.946**	**0.946**	**0.945**	**0.944**	**0.945**	**0.944**

MSE: Mean Squared Error, CIL: Confidence Interval Length, CA: Coverage Accuracy, ORB: Ordinary Bootstrap with Least Squares, IFB: Influence Function Bootstrap, FRB: Fast and Robust Bootstrap, WBP: Weighted Bootstrap with Probability, WFRB$_{LMS}$: Weighted Fast and Robust Bootstrap with Least Median Squares, WFRB$_{LMS}$: LMS: Least Median Squares and LTS: Weighted Fast and Robust Bootstrap with Least Trimmed Squares

Table 5.1 and Figure 5.1 compare robust bootstrap methods for estimating the distribution of slope coefficient in Case I, where the error term ε follows standard normal distribution *N(0,1)*. The results were provided for comparison with other error distributions. For n=20, none of the methods satisfied Bradley's liberal criterion. For n=30, all of the methods satisfied Bradley's liberal criterion except WBP and WFRB$_{LMS}$. ORB, IFB, and WBP again have the lowest MSE (0.075). FRB and WFRB$_{LTS}$ showed slightly higher MSE values (0.082), with WFRB$_{LMS}$ having the highest (0.084). When the sample sizes were 50 and 100 all of the methods satisfied Bradley's liberal criterion. ORB, IFB and WBP had the lowest MSE and CL for both settings. All of the methods performed well except when the sample size was 20.

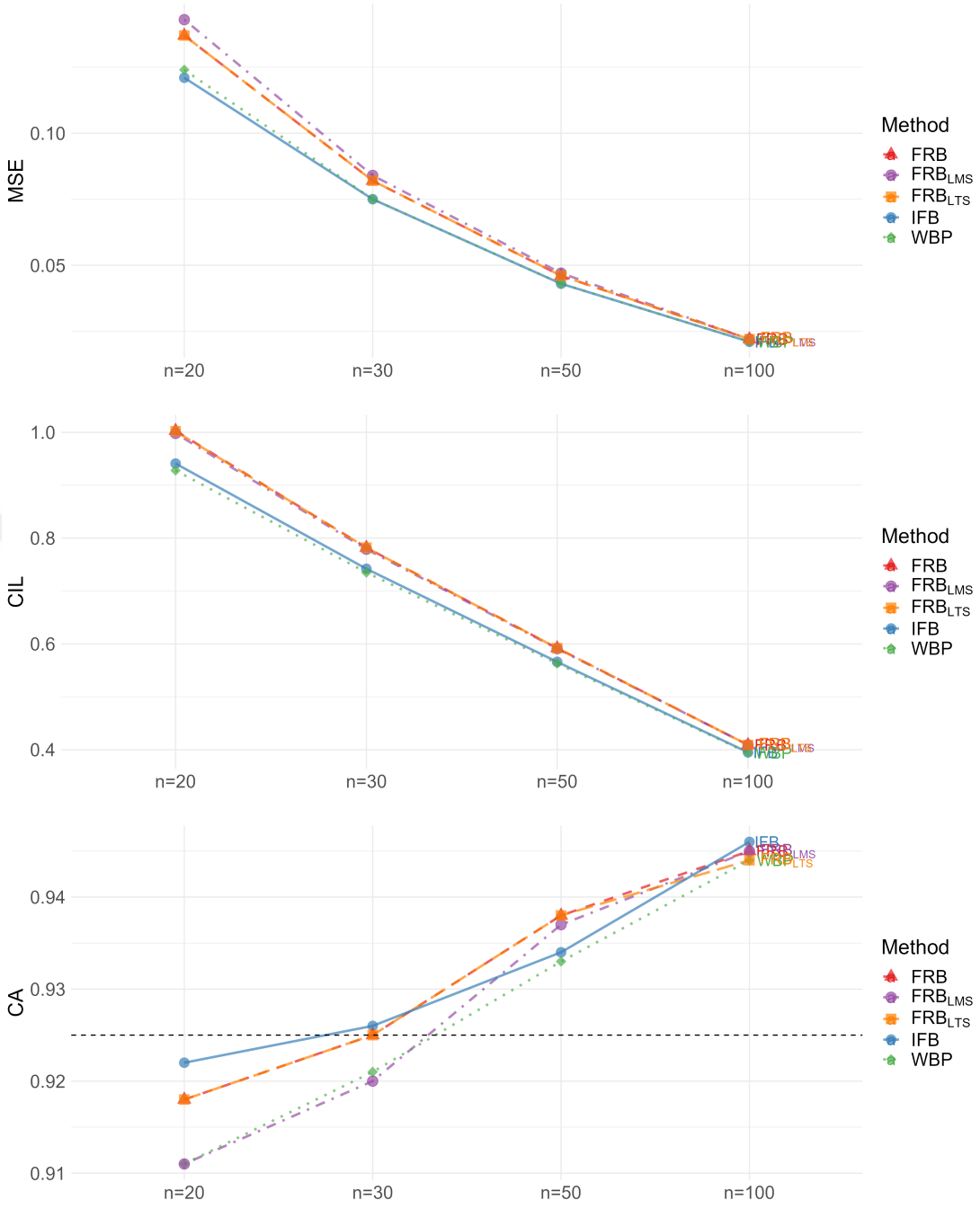


Figure 5.1 MSE, CIL, and CA results for simple linear regression (case I)

Table 5.2 and Figure 5.2 compare robust bootstrap methods for estimating the slope coefficient in Case II, where the error term ε follows a t-distribution with 3 degrees of freedom (t_3). For n=20, all methods have CA below 0.925, indicating that none of the methods effectively controlled for the nominal level of 0.95. For n=30, IFB, FRB, and WFRB_{LTS} have CA in the desired range (0.926, 0.928, and 0.925,

Table 5.2 Comparison of robust bootstrap methods for case II: $\varepsilon \sim t_3$

		ORB	**IFB**	**FRB**	**WBP**	**WFRB$_{LMS}$**	**WFRB$_{LTS}$**
n=20	MSE	0.342	0.251	0.240	0.227	0.240	0.239
	CIL	1.474	1.333	1.307	1.242	1.280	1.297
	CA	0.919	0.920	0.920	0.909	0.911	0.917
n=30	MSE	0.211	0.153	0.138	0.139	0.136	0.137
	CIL	1.169	1.049	1.003	0.991	0.986	0.996
	CA	0.922	**0.926**	**0.928**	0.922	0.922	**0.925**
n=20	MSE	0.123	0.087	0.073	0.079	0.073	0.073
	CIL	0.901	0.798	0.739	0.760	0.729	0.734
	CA	**0.926**	**0.931**	**0.938**	**0.930**	**0.933**	**0.935**
n=20	MSE	0.061	0.042	0.034	0.038	0.033	0.034
	CIL	0.645	0.561	0.507	0.536	0.501	0.504
	CA	**0.934**	**0.942**	**0.945**	**0.941**	**0.941**	**0.942**

MSE: Mean Squared Error, CIL: Confidence Interval Length, CA: Coverage Accuracy, ORB: Ordinary Bootstrap with Least Squares, IFB: Influence Function Bootstrap, FRB: Fast and Robust Bootstrap, WBP: Weighted Bootstrap with Probability, WFRB$_{LMS}$: Weighted Fast and Robust Bootstrap with Least Median Squares, WFRB$_{LMS}$: LMS: Least Median Squares and LTS: Weighted Fast and Robust Bootstrap with Least Trimmed Squares

respectively). FRB and WFRB$_{LMS}$ had the lowest MSE, indicating higher accuracy. WFRB$_{LTS}$ has the shortest confidence interval length (0.996), followed by FRB (1.003) and IFB (1.049). For the rest of the settings, all methods have CA in the desired range, indicating effective control for the nominal level of 0.95. WFRB and FRB have the lowest CIL and MSE. Overall, FRB and WFRB show a balance of accuracy and precision with effective control over coverage accuracy, making them reliable choices for estimating the slope coefficient under t_3 error distribution.

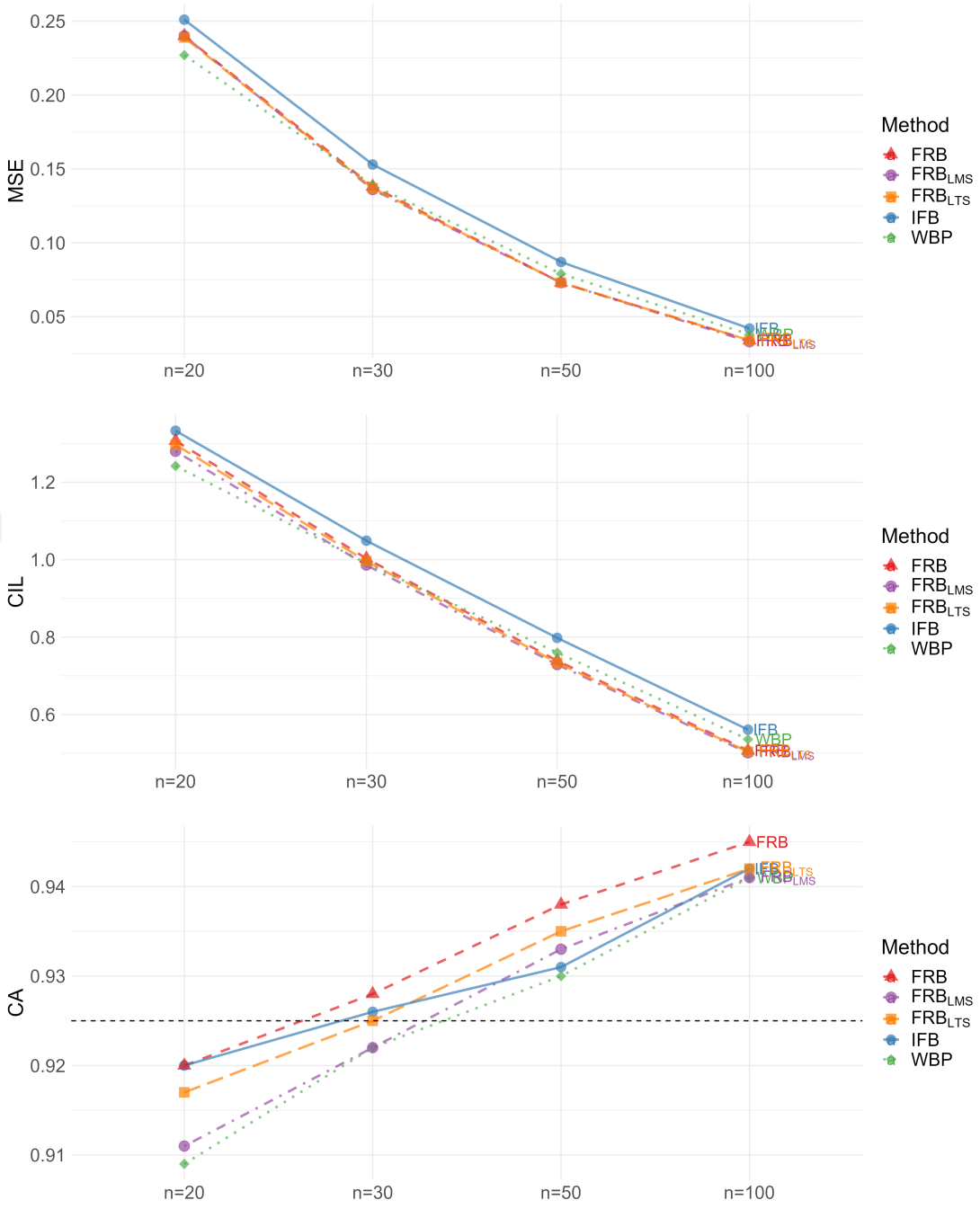


Figure 5.2 MSE, CIL, and CA results for simple linear regression (case II)

Table 5.3 and Figure 5.3 compare robust bootstrap techniques for estimating the slope coefficient in Case III, where the error term ε follows a normal distribution $N(0,1)$ with 10% of the errors replaced by 5. For $n = 20$, none of the approaches show effective control for the nominal level of 0.95, as the CA values are all below 0.925. FRB has the lowest MSE at 0.239 followed by WFRB$_{LMS}$ with an MSE of

Table 5.3 Comparison of robust bootstrap methods for case III: $\varepsilon \sim N(0,1)$ and %10 of error replaced by 5

		ORB	**IFB**	**FRB**	**WBP**	**WFRB$_{\text{LMS}}$**	**WFRB$_{\text{LTS}}$**
n=20	MSE	0.398	0.332	0.239	0.261	0.233	0.235
	CIL	1.701	1.556	1.289	1.347	1.240	1.265
	CA	0.917	0.923	**0.925**	0.911	0.910	0.918
n=30	MSE	0.239	0.204	0.130	0.165	0.126	0.128
	CIL	1.334	1.234	0.968	1.100	0.938	0.951
	CA	**0.922**	**0.929**	**0.935**	**0.929**	**0.926**	**0.931**
n=50	MSE	0.135	0.115	0.064	0.096	0.063	0.063
	CIL	1.008	0.938	0.694	0.856	0.676	0.683
	CA	**0.931**	**0.939**	**0.946**	**0.941**	**0.937**	**0.940**
n=100	MSE	0.065	0.056	0.029	0.048	0.028	0.028
	CIL	0.705	0.660	0.469	0.612	0.459	0.462
	CA	**0.939**	**0.947**	**0.949**	**0.949**	**0.943**	**0.945**

MSE: Mean Squared Error, CIL: Confidence Interval Length, CA: Coverage Accuracy, ORB: Ordinary Bootstrap with Least Squares, IFB: Influence Function Bootstrap, FRB: Fast and Robust Bootstrap, WBP: Weighted Bootstrap with Probability, WFRB$_{\text{LMS}}$: Weighted Fast and Robust Bootstrap with Least Median Squares, WFRB$_{\text{LMS}}$: LMS: Least Median Squares and LTS: Weighted Fast and Robust Bootstrap with Least Trimmed Squares

0.233. The CIL of WFRB$_{\text{LMS}}$ is the shortest at 1.240, suggesting better robustness in interval estimation. For $n = 30$, all techniques effectively control for the nominal level of 0.95. FRB, WFRB$_{\text{LMS}}$ and WFRB$_{\text{LTS}}$ again show the lowest MSE at 0.130, 0.126 and 128 respectively. The CIL of WFRB$_{\text{LMS}}$ is the shortest at 0.938, followed by FRB at 0.968 and WFRB$_{\text{LTS}}$ at 0.951. For $n = 50$ and $n = 100$ similar results were acquired. Overall, WFRB$_{\text{LMS}}$ consistently shows the shortest CILs and among the lowest MSEs across different sample sizes, indicating robust performance in terms of interval estimation and accuracy.

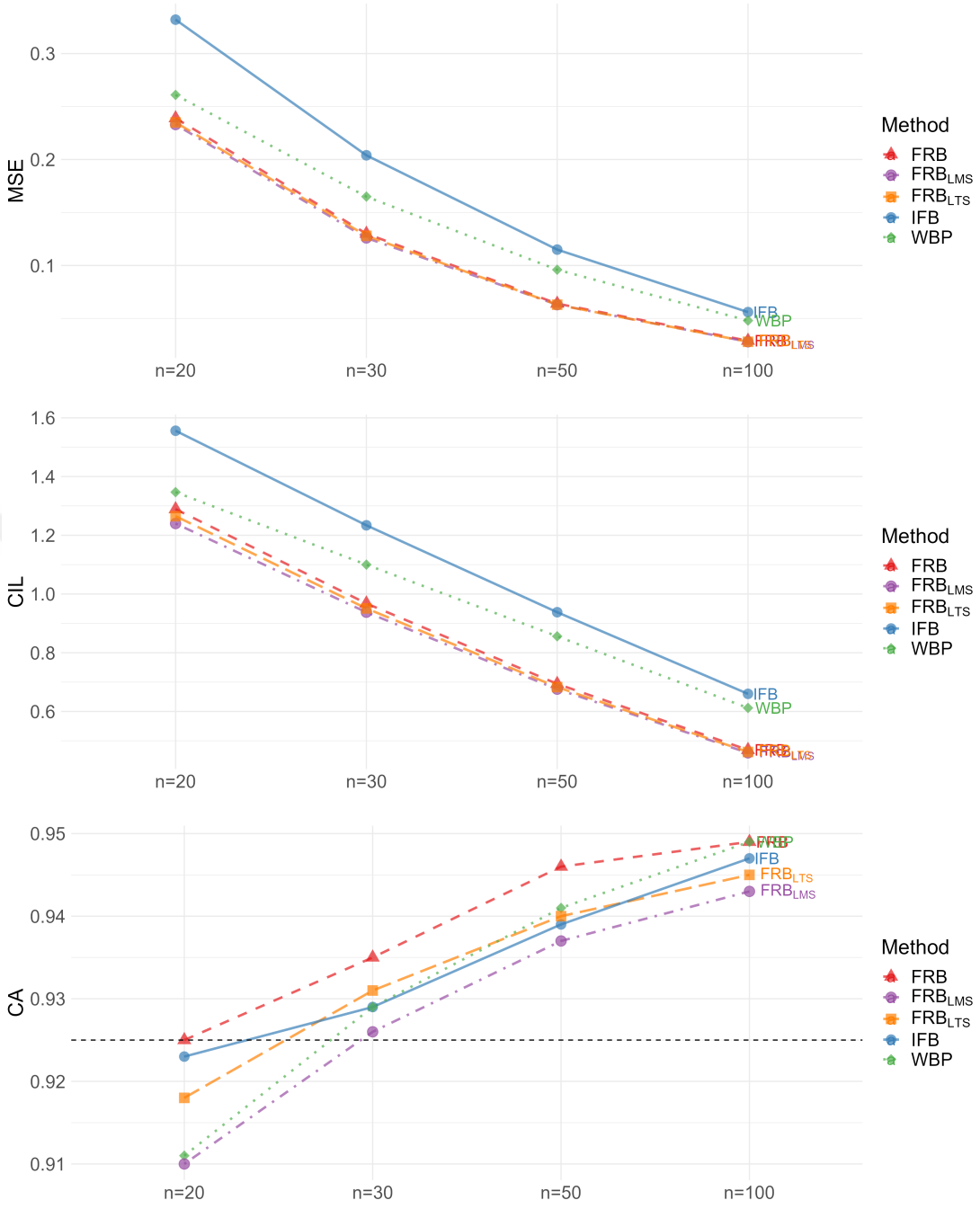


Figure 5.3 MSE, CIL, and CA results for simple linear regression (case III)

Table 5.4 and Figure 5.4 compare robust bootstrap techniques for estimating the slope coefficient in Case IV, where the error term ε follows a contaminated normal distribution $0.9N(0,1) + 0.1N(0,9^2)$. For $n = 20$, only IFB effectively controls for the nominal level of 0.95, as indicated by the boldface CA of 0.925 and FRB has the CA at the edge of the lower bound. FRB has lower MSE and CIL than The IFB suggesting

Table 5.4 Comparison of robust bootstrap methods for case IV: $\varepsilon \sim$ Contaminated Normal $0.9N(0,1) + 0.1N(0,9^2)$

		ORB	**IFB**	**FRB**	**WBP**	**WFRB$_{\text{LMS}}$**	**WFRB$_{\text{LTS}}$**
n=20	MSE	1.099	0.257	0.191	0.197	0.193	0.189
	CIL	2.375	1.328	1.157	1.141	1.101	1.115
	CA	0.922	**0.925**	0.924	0.910	0.906	0.913
n=30	MSE	0.673	0.148	0.105	0.118	0.104	0.104
	CIL	1.941	1.033	0.882	0.909	0.846	0.853
	CA	0.920	**0.931**	**0.930**	0.920	0.915	0.919
n=50	MSE	0.384	0.080	0.056	0.066	0.055	0.055
	CIL	1.532	0.775	0.649	0.693	0.627	0.630
	CA	0.920	**0.940**	**0.936**	**0.931**	**0.925**	**0.927**
n=100	MSE	0.186	0.037	0.026	0.031	0.025	0.025
	CIL	1.118	0.536	0.446	0.485	0.432	0.433
	CA	0.922	**0.947**	**0.943**	**0.943**	**0.935**	**0.936**

MSE: Mean Squared Error, CIL: Confidence Interval Length, CA: Coverage Accuracy, ORB: Ordinary Bootstrap with Least Squares, IFB: Influence Function Bootstrap, FRB: Fast and Robust Bootstrap, WBP: Weighted Bootstrap with Probability, WFRB$_{\text{LMS}}$: Weighted Fast and Robust Bootstrap with Least Median Squares, WFRB$_{\text{LMS}}$: LMS: Least Median Squares and LTS: Weighted Fast and Robust Bootstrap with Least Trimmed Squares

better robustness in interval estimation. For $n = 30$, IFB and FRB effectively control for the nominal level of 0.95 and similar results were acquired as $n = 20$. For $n = 50$, IFB, FRB, WBP, WFRB$_{\text{LMS}}$, and WFRB$_{\text{LTS}}$ effectively control for the nominal level of 0.95, as indicated by the boldface (0.940, 0.936, 0.931, 0.925, and 0.927, respectively). FRB, WFRB$_{\text{LMS}}$, and WFRB$_{\text{LTS}}$ have the lowest MSE at 0.056 and 0.055, respectively. The CIL of WFRB$_{\text{LMS}}$ is the shortest at 0.627, followed by FRB at 0.649 and WFRB$_{\text{LTS}}$ at 0.630. For $n = 100$ similar results were acquired as $n = 50$. One of the reasons that FRB shows good performance is that it is designed to work well when data has been contaminated in the neighborhood.

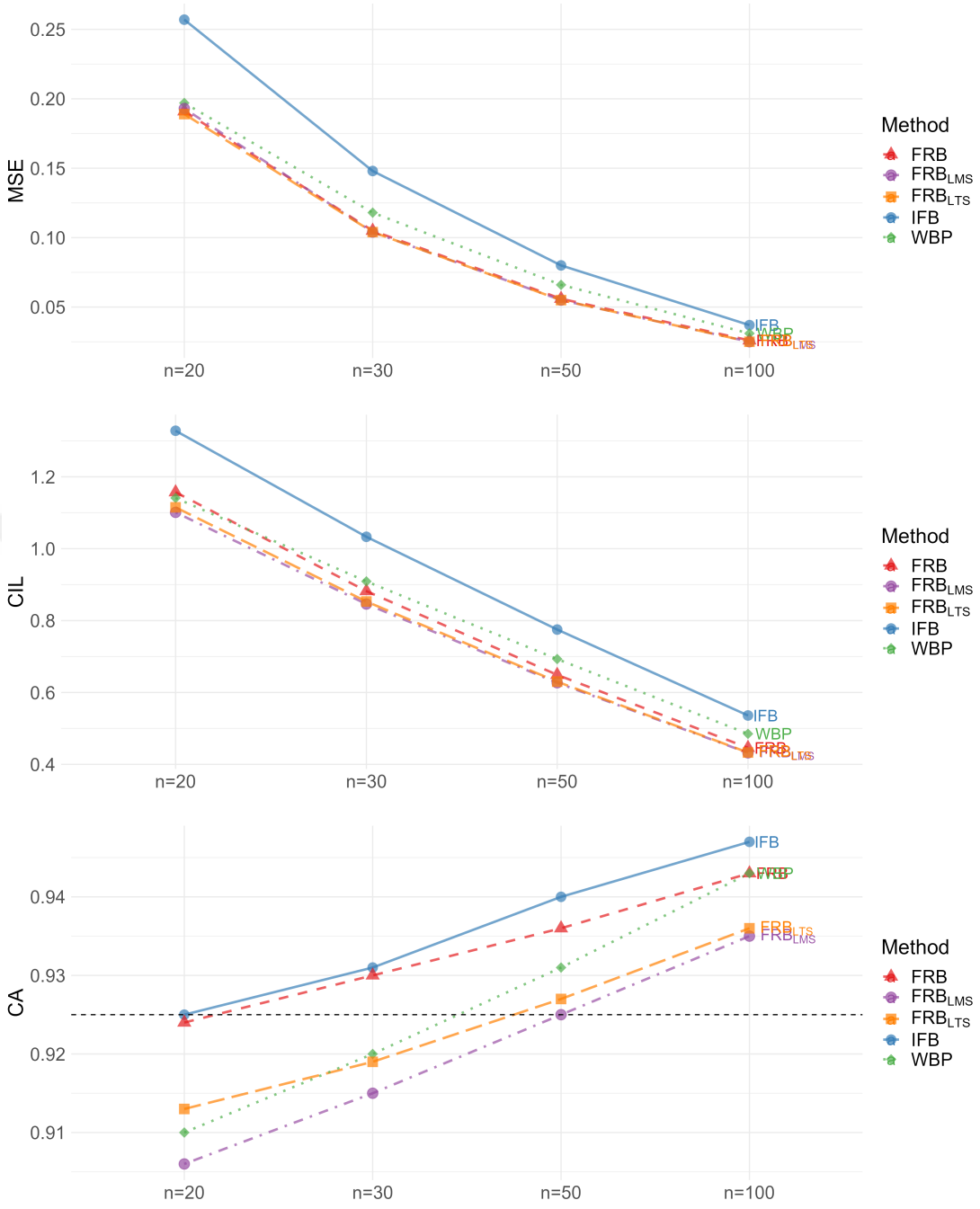


Figure 5.4 MSE, CIL, and CA results for simple linear regression (case IV)

Overall, the comparison of robust bootstrap methods for simple linear regression across various cases and sample sizes reveals that none of the methods performed best in all scenarios. As expected, there is a positive correlation between the MSE and CIL. This suggests that higher levels of error in the model's predictions result in increased CIL. ORB yielded the largest CIL and MSE when the error distribution was

not normal. Both IFB and FRB methods also performed well, consistently satisfying Bradley's liberal criterion. WBP and WFRB_{LMS} suffered from the small sample sizes 20 and 30 because the converge rate of LMS low ($n^{1/3}$). WFRB_{LTS} consistently showed shorter CIL than FRB while satisfying Bradley's liberal criterion.

For the multiple linear regression model ($p = 5$), the average results of all coefficients were provided for the sake of simplicity. The tables and figures include the average results of MSE, CIL, and CA for all sample sizes and one of each error setting. The dashed line in the plots of CA indicates the lower bound (0.925) for Bradley's liberal criterion.

Table 5.5 Comparison of robust bootstrap methods (multiple) for case I: $\varepsilon \sim N(0,1)$

		IFB	**FRB**	**WBP**	**WFRB_{LMS}**	**WFRB_{LTS}**
n=20	MSE	0.151	0.182	0.161	0.205	0.182
	CIL	1.299	1.464	0.957	1.153	1.458
	CA	**0.954**	**0.949**	0.664	0.693	**0.947**
n=30	MSE	0.073	0.087	0.073	0.090	0.086
	CIL	0.869	0.982	0.794	0.927	0.981
	CA	**0.945**	**0.949**	0.864	0.884	**0.949**
n=50	MSE	0.037	0.043	0.036	0.043	0.043
	CIL	0.606	0.663	0.590	0.653	0.663
	CA	**0.943**	**0.949**	**0.925**	**0.935**	**0.949**
n=100	MSE	0.016	0.018	0.016	0.018	0.018
	CIL	0.406	0.430	0.403	0.428	0.430
	CA	**0.945**	**0.949**	**0.943**	**0.947**	**0.949**

MSE: Mean Squared Error, CIL: Confidence Interval Length, CA: Coverage Accuracy, ORB: Ordinary Bootstrap with Least Squares, IFB: Influence Function Bootstrap, FRB: Fast and Robust Bootstrap, WBP: Weighted Bootstrap with Probability, WFRB_{LMS}: Weighted Fast and Robust Bootstrap with Least Median Squares, WFRB_{LMS}: LMS: Least Median Squares and LTS: Weighted Fast and Robust Bootstrap with Least Trimmed Squares

Table 5.5 and Figure 5.5 show robust bootstrap methods for estimating the distribution of the slope coefficient in Case I, where the error term ε follows a standard normal distribution $N(0,1)$. For $n = 20$, IFB, FRB, and WFRB_{LTS} satisfied Bradley's liberal criterion with CAs of 0.954, 0.949, and 0.947, respectively. Among these, IFB and WBP had the lowest MSE values of 0.151 and 0.161, respectively. For

$n = 30$, IFB, FRB, and WFRB_{LTS} again satisfied Bradley's liberal criterion with CAs of 0.945, 0.949, and 0.949, respectively. IFB and WBP exhibited the lowest MSE (0.073 each), and the CIL of WBP was the shortest at 0.794, further confirming its robustness. When the sample size increased to $n = 50$, all methods satisfied Bradley's liberal criterion, except WBP and WFRB_{LMS}, (0.925 and 0.935, respectively). WBP had the lowest MSE at 0.036, and its CIL was the shortest at 0.590. For $n = 100$, all approaches satisfied Bradley's liberal criterion, with CAs of 0.945, 0.949, 0.943, 0.947, and 0.949 for IFB, FRB, WBP, WFRB_{LMS}, and WFRB_{LTS}, respectively. Both IFB and WBP had the lowest MSE values of 0.016 individually, and the CIL of WBP was the shortest at 0.403, suggesting better robustness in interval estimation. Overall, WBP consistently demonstrated the shortest CIL and one of the lowest MSEs across various sample sizes. Both IFB and FRB also showed strong performance for MSE.

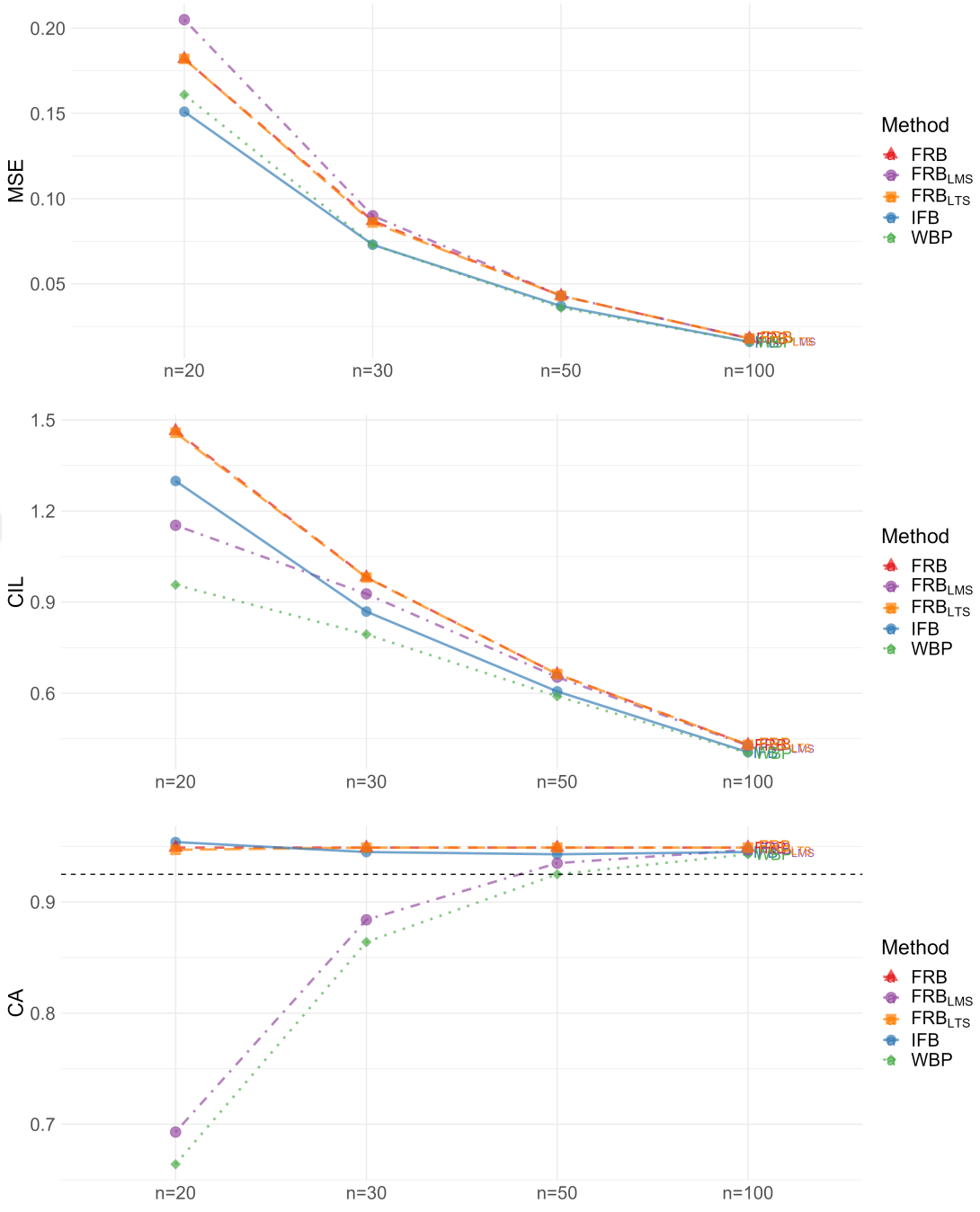


Figure 5.5 Average MSE, CIL and CA results for multiple linear regression (case I])

Table 5.6 and Figure 5.6 present a comparison of robust bootstrap methods for estimating the distribution of the slope coefficient in Case II, where the error term ε follows a t-distribution with 3 degrees of freedom (t_3). For $n = 20$, IFB, FRB, and WFRB$_{LTS}$ satisfied Bradley's liberal criterion with CAs of 0.958, 0.956, and 0.952, respectively. Out of each choice, WBP had the lowest MSE (0.243), and the CIL of

Table 5.6 Comparison of robust bootstrap methods (multiple) for case II: $\varepsilon \sim t_3$

		IFB	**FRB**	**WBP**	**WFRB$_{\text{LMS}}$**	**WFRB$_{\text{LTS}}$**
n=20	MSE	0.291	0.312	0.243	0.323	0.305
	CIL	1.846	2.021	1.257	1.582	1.993
	CA	**0.958**	**0.956**	0.692	0.723	**0.952**
n=30	MSE	0.144	0.148	0.119	0.142	0.147
	CIL	1.230	1.306	1.026	1.206	1.294
	CA	**0.946**	**0.954**	0.873	0.900	**0.951**
n=50	MSE	0.073	0.069	0.063	0.067	0.069
	CIL	0.861	0.858	0.785	0.832	0.852
	CA	**0.944**	**0.952**	**0.929**	**0.942**	**0.951**
n=100	MSE	0.033	0.029	0.030	0.028	0.029
	CIL	0.861	0.858	0.785	0.832	0.852
	CA	**0.944**	**0.952**	**0.929**	**0.942**	**0.951**

MSE: Mean Squared Error, CIL: Confidence Interval Length, CA: Coverage Accuracy, ORB: Ordinary Bootstrap with Least Squares, IFB: Influence Function Bootstrap, FRB: Fast and Robust Bootstrap, WBP: Weighted Bootstrap with Probability, WFRB$_{\text{LMS}}$: Weighted Fast and Robust Bootstrap with Least Median Squares, WFRB$_{\text{LMS}}$: LMS: Least Median Squares and LTS: Weighted Fast and Robust Bootstrap with Least Trimmed Squares

WBP was the shortest at 1.257. For $n = 30$, IFB, FRB, and WFRB$_{\text{LTS}}$ again satisfied Bradley's liberal criterion (0.946, 0.954, and 0.951, respectively). WBP exhibited the lowest MSE of 0.119, and the CIL of WBP was the shortest at 1.026, demonstrating its robustness. When the sample size increased to $n = 50$, all methods met Bradley's liberal criterion except for WBP and WFRB$_{\text{LMS}}$, which had CAs of 0.929 and 0.942, respectively. WBP achieved the lowest MSE at 0.063, and its CIL was the shortest at 0.785. For $n = 100$, all methods satisfied Bradley's liberal criterion, with CAs of 0.944, 0.952, 0.929, 0.942, and 0.951 for IFB, FRB, WBP, WFRB$_{\text{LMS}}$, and WFRB$_{\text{LTS}}$, respectively. WBP had the lowest MSE (0.030), and the CIL of WBP was the shortest at 0.785. Overall, WBP showed the shortest CILs and among the lowest MSEs across different sample sizes, and IFB and FRB also performed well in MSE.

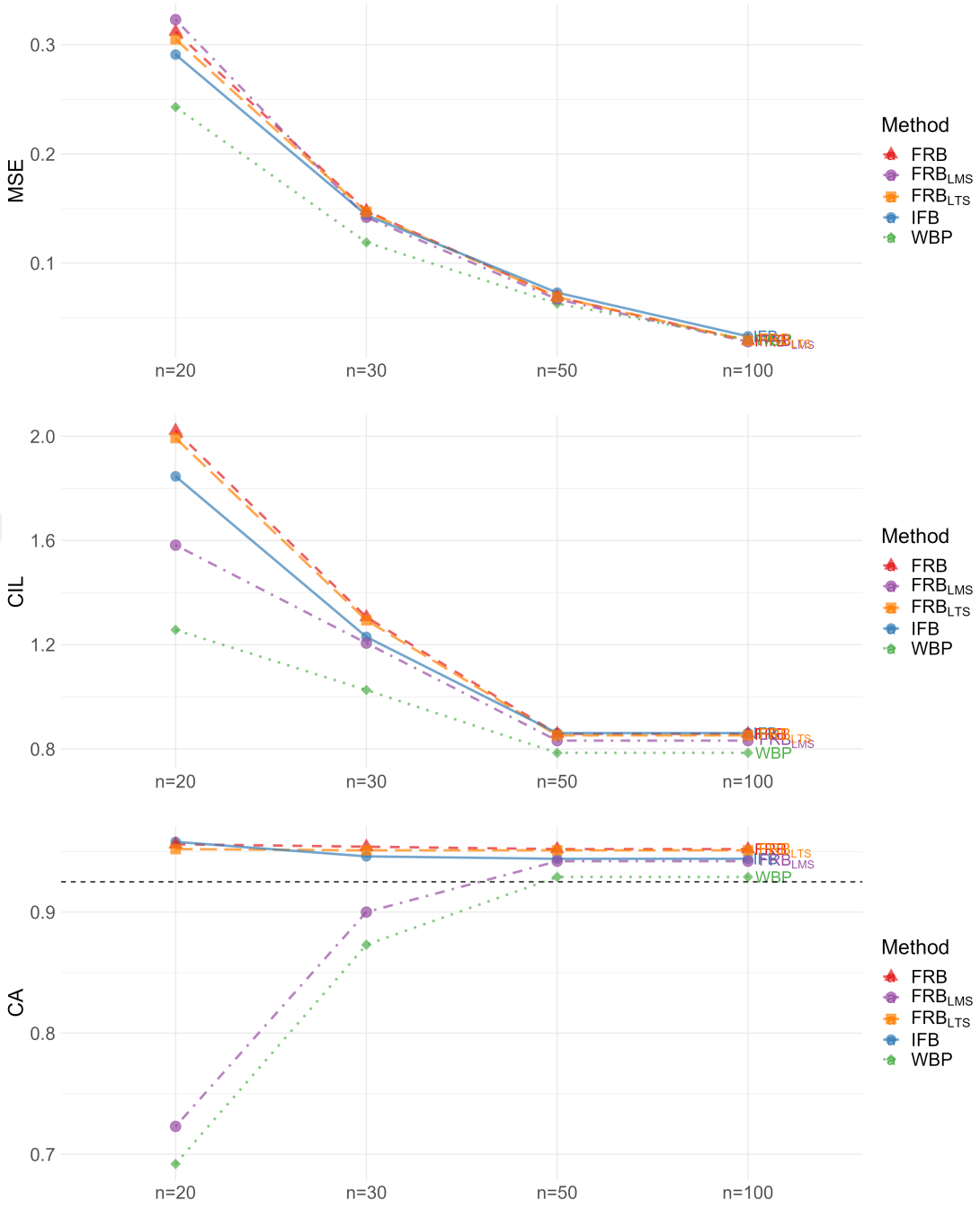


Figure 5.6 Average MSE, CIL and CA results for multiple linear regression (case II)

Table 5.7 and Figure 5.7 provides a comparison of robust bootstrap methods for estimating the distribution of the slope coefficient in Case III, where the error term ε follows $N(0,1)$ with 10% of the errors replaced by 10. When $n = 20$, IFB, FRB, and WFRB$_{LTS}$ all met Bradley's liberal criterion with CAs of 0.959, 0.969, and 0.947, respectively. WBP had the smallest MSE with a value of 0.181, and the CIL of WBP

Table 5.7 Comparison of robust bootstrap methods (multiple) for case III: $\varepsilon \sim N(0,1)$ and %10 of error replaced by 10

		IFB	**FRB**	**WBP**	**WFRB$_{\text{LMS}}$**	**WFRB$_{\text{LTS}}$**
n=20	MSE	0.312	0.234	0.181	0.196	0.188
	CIL	2.069	1.664	1.221	1.288	1.477
	CA	**0.959**	**0.969**	0.761	0.786	**0.947**
n=30	MSE	0.133	0.095	0.079	0.085	0.085
	CIL	1.239	1.154	0.937	1.041	1.106
	CA	**0.948**	**0.955**	0.881	0.898	**0.944**
n=50	MSE	0.063	0.046	0.039	0.041	0.042
	CIL	0.862	0.687	0.628	0.635	0.641
	CA	**0.955**	**0.948**	0.919	**0.928**	**0.933**
n=100	MSE	0.028	0.020	0.018	0.018	0.018
	CIL	0.578	0.542	0.548	0.534	0.539
	CA	**0.944**	**0.950**	**0.942**	**0.947**	**0.949**

MSE: Mean Squared Error, CIL: Confidence Interval Length, CA: Coverage Accuracy, ORB: Ordinary Bootstrap with Least Squares, IFB: Influence Function Bootstrap, FRB: Fast and Robust Bootstrap, WBP: Weighted Bootstrap with Probability, WFRB$_{\text{LMS}}$: Weighted Fast and Robust Bootstrap with Least Median Squares, WFRB$_{\text{LMS}}$: LMS: Least Median Squares and LTS: Weighted Fast and Robust Bootstrap with Least Trimmed Squares

was the shortest at 1.221. For $n = 30$, IFB, FRB, and WFRB$_{\text{LTS}}$ again satisfied Bradley's liberal criterion (0.948, 0.955, and 0.944, respectively). WBP has the lowest MSE of 0.079, and the CIL of WBP was the shortest at 0.937. When the sample size $n = 50$, all methods met Bradley's liberal criterion except for WBP and WFRB$_{\text{LMS}}$, which had CAs of 0.919 and 0.928, respectively. WBP achieved the lowest MSE at 0.039, and its CIL was the shortest at 0.628. For $n = 100$, all approaches satisfied Bradley's liberal criterion, with CAs of 0.944, 0.950, 0.942, 0.947, and 0.949 for IFB, FRB, WBP, WFRB$_{\text{LMS}}$, and WFRB$_{\text{LTS}}$, respectively. WBP had the lowest MSE (0.018), and the CIL of WBP was the shortest at 0.548. Overall, WBP showed the shortest CILs and among the lowest MSEs across different sample sizes, and IFB and FRB also performed well in MSE.

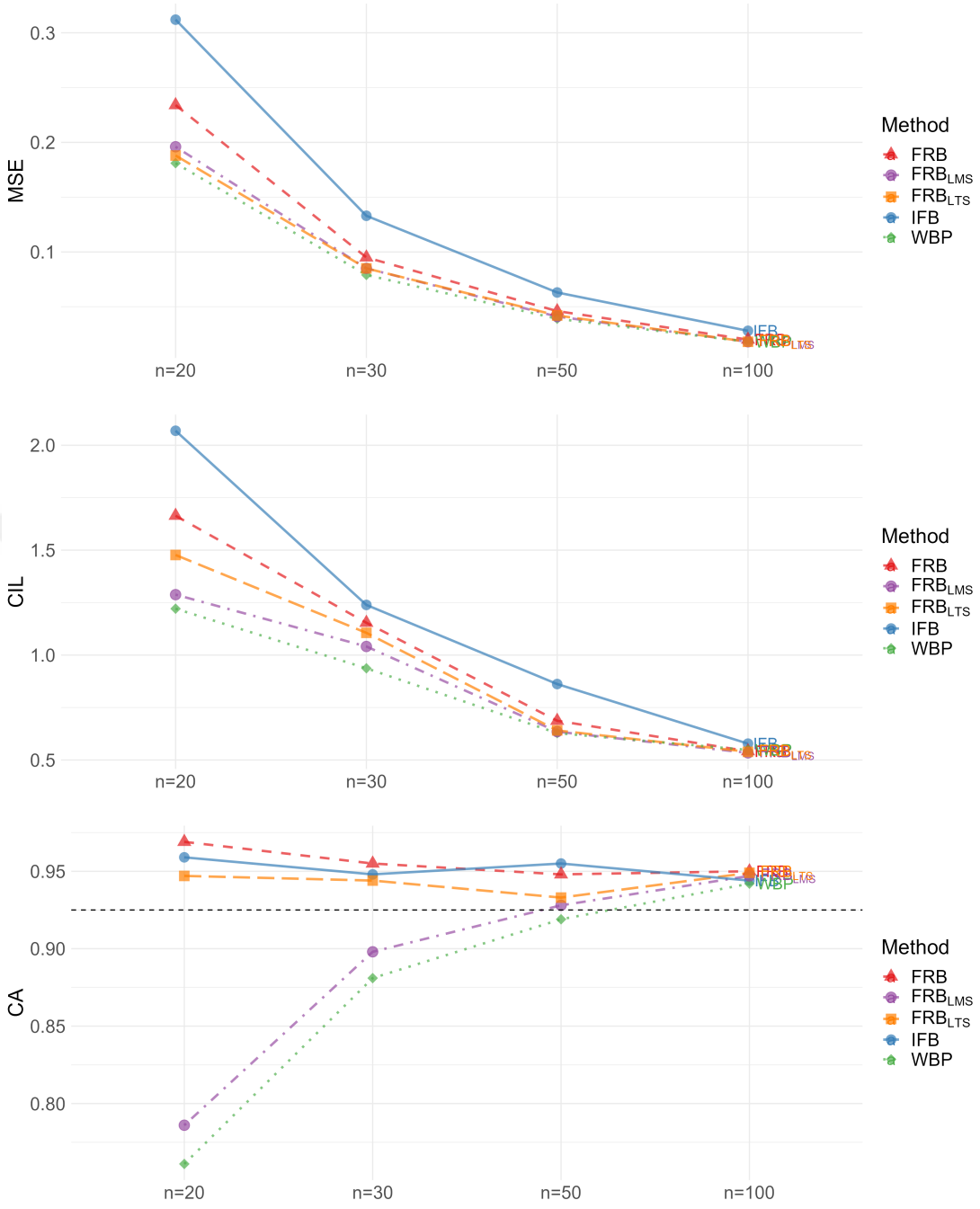


Figure 5.7 Average MSE, CIL and CA results for multiple linear regression (case III)

Table 5.8 and Figure 5.8 present a comparison of robust bootstrap methods for estimating the distribution of the slope coefficient in Case IV, where the error term ε follows a contaminated normal distribution $0.9N(0,1) + 0.1N(0,9^2)$. When the value of n is 20, IFB, FRB, and WFRB$_{\text{LTS}}$ satisfied Bradley's liberal criterion with CAs of 0.960, 0.962, and 0.951, respectively. Out of these, WBP had the lowest MSE

Table 5.8 Comparison of robust bootstrap methods (multiple) for case IV: $\varepsilon \sim$ Contaminated Normal $0.9N(0,1) + 0.1N(0,9^2)$

		IFB	**FRB**	**WBP**	**WFRB$_{\text{LMS}}$**	**WFRB$_{\text{LTS}}$**
n=20	MSE	0.278	0.261	0.205	0.258	0.238
	CIL	1.948	1.925	1.226	1.481	1.820
	CA	**0.960**	**0.962**	0.721	0.739	**0.951**
n=30	MSE	0.132	0.112	0.093	0.105	0.106
	CIL	1.234	1.148	0.938	1.038	1.102
	CA	**0.948**	**0.955**	0.883	0.899	**0.945**
n=50	MSE	0.064	0.052	0.049	0.049	0.049
	CIL	0.842	0.742	0.706	0.703	0.717
	CA	**0.946**	**0.950**	**0.927**	**0.933**	**0.941**
n=100	MSE	0.029	0.022	0.024	0.021	0.021
	CIL	0.555	0.448	0.425	0.421	0.422
	CA	**0.964**	**0.947**	**0.930**	**0.934**	**0.935**

MSE: Mean Squared Error, CIL: Confidence Interval Length, CA: Coverage Accuracy, ORB: Ordinary Bootstrap with Least Squares, IFB: Influence Function Bootstrap, FRB: Fast and Robust Bootstrap, WBP: Weighted Bootstrap with Probability, WFRB$_{\text{LMS}}$: Weighted Fast and Robust Bootstrap with Least Median Squares, WFRB$_{\text{LMS}}$: LMS: Least Median Squares and LTS: Weighted Fast and Robust Bootstrap with Least Trimmed Squares

(0.205), and the CIL of WBP was the shortest at 1.226. For $n = 30$, IFB, FRB, and WFRB$_{\text{LTS}}$ all met Bradley's liberal criterion with CAs of 0.948, 0.955, and 0.945, respectively. WBP showed the lowest MSE of 0.093, and the CIL of WBP was the shortest at 0.938. When the sample size increased to $n = 50$, all approaches satisfied Bradley's liberal criterion except for WBP and WFRB$_{\text{LMS}}$, (0.927 and 0.933, respectively). WBP achieved the lowest MSE at 0.049, and its CIL was the shortest at 0.706. For $n = 100$, all methods met Bradley's liberal criterion (0.964, 0.947, 0.930, 0.934, and 0.935, respectively). FRB had the lowest MSE (0.022), and the CIL of FRB was the shortest at 0.448. In general, WBP showed the shortest CILs and among the lowest MSEs across different sample sizes. Additionally, the IFB and FRB methods also shown strong performance in terms of MSE.

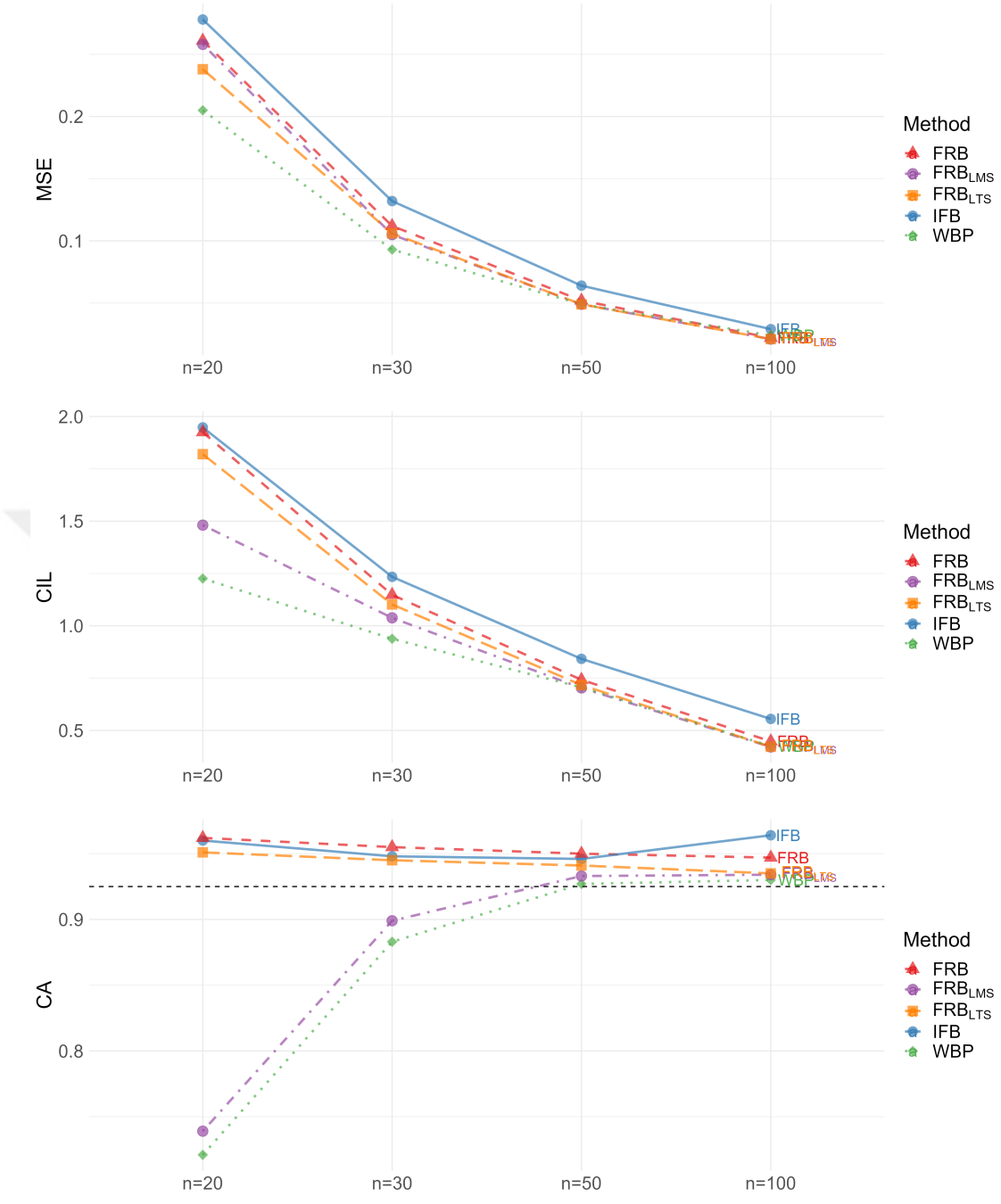


Figure 5.8 Average MSE, CIL and CA results for multiple linear regression (case IV)

The comparative analysis of robust bootstrap methods for multiple linear regression across different cases and sample sizes underscores the consistent performance and reliability of these methods in various challenging scenarios. IFB, FRB and WFRB$_{LTS}$ consistently satisfy Bradley's liberal criterion and achieve low MSE values, in both small and large sample sizes. However, WFRB$_{LTS}$ had the

smallest MSE and CIL among IFB and FRB. Similar to the simple linear regression results, WBP and $WFRB_{LMS}$ suffered from the low convergence rate in the small sample sizes, although had the shortest CIL and lowest MSE results. In larger sample sizes, WBP had the shortest CIL while satisfying Bradley's liberal criterion.

5.2 Real-Life Data Study

In this study, "Belgium Phone Calls 1950-1973" a real-life data set included in the R package MASS, was used (Rousseeuw & Leroy, 1987). This data set includes the last two digits of the year when the information was received, and the international phone calls (in tens of millions) made in that year. As seen in the scatter plot, the data set obtained between 1964 and 1970 differs from previous years. It was found that this discrepancy was due to the use of a different recording system from 1964 to 1969, which recorded the total number of minutes of these calls. The years 1963 and 1970 were also partially affected. Since the transition to a different recording system did not occur precisely at the beginning of the year, the number of calls in some months was added to the number of minutes recorded in the remaining months. This minor glitch resulted in a small but effective data set that can be used to test the performance of different methods. The scatter plot of the dataset is provided in Figure 5.9. We only added the fitted line of LMS as a robust estimate to avoid overlapping since robust estimates are found close.

The confidence intervals of the coefficients for the 0.05 significance level are as in Table 5.9. The values in the initial row indicate the confidence interval of the coefficient, while the subsequent row displays the length of the confidence interval for each method.

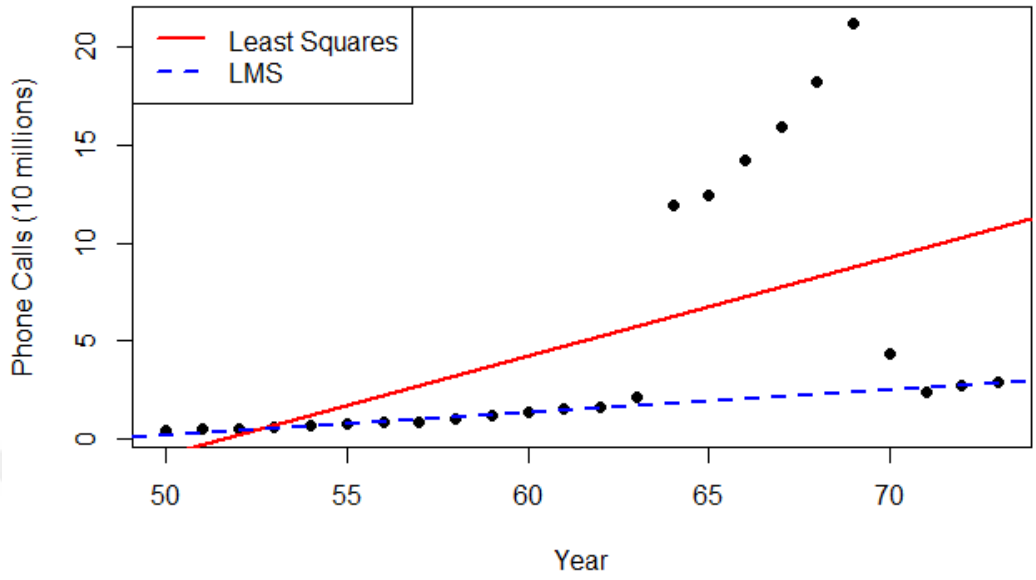


Figure 5.9 Least squares and LMS fits on Belgium phone calls data

Table 5.9 The confidence intervals and their lengths of the coefficients (Belgium phone calls)

	β_1
ORB	(0.2231, 0.930) 0.6926
IFB	(0.1032, 0.1304) 0.0272
FRB	(0.0996, 0.1206) 0.0210
WBP	(0.1033, 0.1193) 0.0160
$WFRB_{LMS}$ **$WFRB_{LTS}$**	(0.1019, 0.1177) 0.0158 (0.1019, 0.0.1182) 0.0163

Table 5.9 presents a comparison of robust bootstrap methods for estimating the distribution of the slope coefficient. The CIL of ORB is much more larger than that of other methods since the ORB is sensitive to outliers. Although the CILs of WBP, $WFRB_{LMS}$, and $WFRB_{LTS}$ are similar, $WFRB_{LMS}$ has the shortest length at 0.0158. Additionally, the CILs of IFB and FRB are 0.0210 and 0.0272, respectively. The lengths of these methods are greater than other robust bootstrap methods. The findings of the real-life data study support the results of the simulation.

CHAPTER SIX
CONCLUSION

In this thesis, the outlier observations in the data sets, their behaviors, and their effects were investigated. Data points that deviate significantly from the majority of a data set are called outliers. These observations may occur due to factors such as human errors, changes in system behavior, machine error, fraud activity, or natural and spontaneous variations within populations.

Outlier observations, which are seen in many fields, have been a focus of research for a long time and maintain their popularity among researchers, as they play a major role in revealing hidden information on the phenomenon under study, improving data quality before analysis, detecting anomalies, and affecting the performance of any findings or models obtained in the analysis results.

In the Internet era, it is vital to take prompt and immediate action in situations such as credit card fraud detection, intrusion detection, and enemy activity surveillance. The ever-increasing amount of data and the expectation of rapid response in modern applications has greatly increased the need for fast and accurate methods. HBOS meets the expectations since it is a quick unsupervised anomaly detection method and can be calculated using either a static or dynamic bin-width histogram. HBOS has gained substantial popularity and is widely used in many application areas and benchmark studies.

However, several issues directly affect the performance of the HBOS. The choice of an incorrect bin number causes an over-smoothing problem in the static bin-width approach in the presence of extreme values. This explains why the dynamic bin-width approach has gained popularity over the static bin-width approach. However, the dynamic bin-width approach tends to label inliers as anomalies in the presence of gaps. This issue can be fixed in the static bin-width approach by implementing a commonly used robust bin size rule FD. Since the FD rule is robust to outliers, the constructed histograms are not affected by them.

Another issue with the static bin-width approach is that inliers and outliers might get equal densities. In this case, histogram gaps could be used to remedy this issue. Gaps in the static bin-width histogram are more likely to occur in circumstances when the parent distribution has a heavy tail or anomalies exist. However, scores obtained from static bin-width histograms do not consider the gaps that could be used to increase the performance of HBOS. This study firstly presents the AHBOS, which modifies the static bin-width histogram's densities depending on both the bins and their neighbors. Simulation and real data application results supported the fact that the proposed method improves performance. Based on the simulation results, the AHBOS performed better than the HBOS method for all performance metrics, especially for the PR-AUC measure. Except for a few cases, the real data study also yielded a result supporting the simulation outcomes. The new approach demonstrated that it could compete well with dynamic bin-width HBOS when dealing with heavy-tailed or unknown distributions. The adjustment also performed well in the absence of gaps.

In this thesis, we investigated the effect of outliers in bootstrapping, specifically in linear regression models. In the presence of outliers, robust methods are preferred over classical approaches due to their ability to handle the effects of outliers. In regression modeling, robust estimators' asymptotic distribution is usually examined under specific conditions, such as when the data displays symmetry or an elliptical distribution (Salibián-Barrera et al., 2008). For example, the errors are assumed to be symmetric, which does not usually hold because outliers do not need to appear symmetrical. Moreover, in some cases, it may not be feasible to derive the exact distribution of robust estimators and even if it is derivable, convergence may be slow (Amado et al., 2014).

Bootstrapping robust estimators become popular since it solves some of the problems associated with the problems mentioned. Despite providing practical solutions in many scenarios, additional issues may arise, particularly in the presence of outliers. The first problem is that bootstrap samples may contain more outliers than the original sample. Calculating robust estimators for each bootstrap can be

expensive, especially considering that some robust estimators require iterative processes. Robust bootstrap methods have emerged to overcome these problems. In this study, we particularly investigated IFB, WBP, and FRB. To contribute to the theory of robust bootstrap procedure, we have provided a new approach called WFRB, which is a combination of IFB and FRB.

Simulation studies comparing robust bootstrap approaches for both simple and multiple linear regression across different situations and sample sizes demonstrate that no single method consistently outperformed all others. A positive correlation between MSE and CIL was observed, indicating higher prediction errors lead to larger CIL. For simple linear regression, ORB yielded the largest CIL and MSE with non-normal error distributions, while IFB and FRB consistently met Bradley's liberal criterion. WBP and $WFRB_{LMS}$ struggled with small sample sizes due to the low convergence rate of LMS. $WFRB_{LTS}$ consistently showed shorter CIL than FRB while satisfying Bradley's liberal criterion in some cases. For multiple linear regression, IFB, FRB, and $WFRB_{LTS}$ performed reliably, with $WFRB_{LTS}$ achieving the smallest MSE and CIL. Despite challenges with small sample sizes, WBP and $WFRB_{LMS}$ had the shortest CIL and lowest MSE, with WBP having the best results in larger sample sizes. Overall, FRB yielded a balanced result for three metrics. Depending on the initial estimator, WFRB methods gave the best results. IFB usually gave the longest CIL among the other robust bootstrap methods while satisfying Bradley's criteria in most cases. WBP yielded the shortest CIL in general, however, it struggled to give reliable CA results for small sample sizes. Real-life data study including the CIL estimates, supports the findings of the simulation study.

The impact of good leverage points in bootstrap has not been recognized well in the literature. Good leverage points can increase the accuracy of the estimates and decrease their standard error. We proposed a data splitting technique to utilize the effect of good leverage points in bootstrap that can be used in conjunction with the robust bootstrap methods. A simple simulation study showed that our approach significantly decreases MSE and CIL.

REFERENCES

Aggarwal, C. C. (2017). *An introduction to outlier analysis*. Springer.

Alamgir, S., & Ali, A. (2013). Split sample bootstrap method. *World Applied Sciences Journal*, *21*(7), 983–993.

Alma, Ö. G. (2011). Comparison of robust regression methods in linear regression. *Int. J. Contemp. Math. Sciences*, *6*(9), 409–421.

Amado, C., Bianco, A. M., Boente Boente, G. L., & Pires, A. M. (2014). Robust bootstrap: an alternative to bootstrapping robust estimators.

Amado, C., & Pires, A. M. (2004). Robust bootstrap with non random weights based on the influence function. *Communications in Statistics-Simulation and Computation*, *33*(2), 377–396.

Anscombe, F. J. (1960). Rejection of outliers. *Technometrics*, *2*(2), 123–146.

Athreya, K. (1987). Bootstrap of the mean in the infinite variance case. *The annals of statistics*, 724–731.

Binzat, U., & Yıldıztepe, E. (2023). The adjusted histogram-based outlier score-ahbos. *Mugla Journal of Science and Technology*, *9*(1), 92–100.

Bradley, J. V. (1978). Robustness? *British Journal of Mathematical and Statistical Psychology*, *31*(2), 144–152.

Breunig, M. M., Kriegel, H.-P., Ng, R. T., & Sander, J. (2000). Lof: identifying density-based local outliers. In *Proceedings of the 2000 ACM SIGMOD international conference on Management of data*, (93–104).

Chandola, V., Banerjee, A., & Kumar, V. (2009). Anomaly detection: A survey. *ACM computing surveys (CSUR)*, *41*(3), 1–58.

Chernick, M. R. (2011). *Bootstrap methods: A guide for practitioners and researchers*. John Wiley & Sons.

Davis, J., & Goadrich, M. (2006). The relationship between precision-recall and roc curves. In *Proceedings of the 23rd international conference on Machine learning*, (233–240).

Davison, A. C., & Hinkley, D. V. (1997). *Bootstrap methods and their application*. Number 1. Cambridge university press.

Demšar, J. (2006). Statistical comparisons of classifiers over multiple data sets. *The Journal of Machine learning research*, *7*, 1–30.

Donoho, D. L., & Huber, P. J. (1983). The notion of breakdown point. *A festschrift for Erich L. Lehmann*, *157184*.

Efron, B. (1979). Bootstrap Methods: Another Look at the Jackknife. *The Annals of Statistics*, *7*(1), 1 – 26.

Efron, B. (1981). Nonparametric standard errors and confidence intervals. *canadian Journal of Statistics*, *9*(2), 139–158.

Efron, B. (1982). *The jackknife, the bootstrap and other resampling plans*. SIAM.

Efron, B. (1987). Better bootstrap confidence intervals. *Journal of the American Statistical Association*, *82*(397), 171–185.

Eskin, E. (2000). Anomaly detection over noisy data using learned probability distributions, 255–262.

Fox, J. (2015). *Applied regression analysis and generalized linear models*. Sage publications.

Freedman, D., & Diaconis, P. (1981). On the histogram as a density estimator: L 2 theory. *Zeitschrift für Wahrscheinlichkeitstheorie und verwandte Gebiete*, *57*(4), 453–476.

Friedman, M. (1937). The use of ranks to avoid the assumption of normality implicit in the analysis of variance. *Journal of the american statistical association*, *32*(200), 675–701.

Friedman, M. (1940). A comparison of alternative tests of significance for the problem of m rankings. *The annals of mathematical statistics*, *11*(1), 86–92.

Ghosh, M., Parr, W. C., Singh, K., & Babu, G. J. (1984). A note on bootstrapping the sample median. *The Annals of Statistics*, *12*(3), 1130–1135.

Goldstein, M. (2015). Unsupervised anomaly detection benchmark. https://doi.org/10.7910/DVN/OPQMVF. Harvard Dataverse.

Goldstein, M., & Dengel, A. (2012). Histogram-based outlier score (hbos): A fast unsupervised anomaly detection algorithm. *KI-2012: poster and demo track*, *1*, 59–63.

Goldstein, M., & Uchida, S. (2016). A comparative evaluation of unsupervised anomaly detection algorithms for multivariate data. *PloS one*, *11*(4), e0152173.

Grubbs, F. E. (1969). Procedures for detecting outlying observations in samples. *Technometrics*, *11*(1), 1–21.

Hall, P. (2012). *The bootstrap and Edgeworth expansion*. Springer.

Hampel, F. R. (1975). *Beyond location parameters: Robust concepts and methods*. Eidgenössische Technische Hochschule Zürich.

Hawkins, D. M. (1980). *Identification of outliers*, volume 11. Springer.

Hodge, V., & Austin, J. (2004). A survey of outlier detection methodologies. *Artificial intelligence review*, *22*, 85–126.

Hubert, M., & Vandervieren, E. (2008). An adjusted boxplot for skewed distributions. *Computational statistics & data analysis*, *52*(12), 5186–5201.

Latecki, L. J., Lazarevic, A., & Pokrajac, D. (2007). Outlier detection with kernel density functions. In *International Workshop on Machine Learning and Data Mining in Pattern Recognition*, (61–75). Springer.

Ligges, U., & Mächler, M. (2002). Scatterplot3d-an r package for visualizing multivariate data. Technical report, Technical report.

Maechler, M., Rousseeuw, P., Croux, C., Todorov, V., Ruckstuhl, A., Salibian-Barrera, M., Verbeke, T., Koller, M., Conceicao, E. L. T., & Anna di Palma, M. (2021). robustbase: Basic robust statistics.

Nemenyi, P. B. (1963). *Distribution-free multiple comparisons.* PhD thesis, Princeton University.

Norazan, M., Midi, H., & Imon, A. (2009). Estimating regression coefficients using weighted bootstrap with probability. *WSEAS Transactions on Mathematics*, *9*(7).

Paulauskas, N., & Baskys, A. (2019). Application of histogram-based outlier scores to detect computer network anomalies. *Electronics*, *8*(11), 1251.

Pohlert, T. (2023). *PMCMRplus: Calculate Pairwise Multiple Comparisons of Mean Rank Sums Extended*. R package version 1.9.10.

Provost, F. J., Fawcett, T., Kohavi, R. et al. (1998). The case against accuracy estimation for comparing induction algorithms. In *ICML*, volume 98, (445–453).

R Core Team (2023). *R: A Language and Environment for Statistical Computing*. Vienna, Austria: R Foundation for Statistical Computing.

Rayana, S. (2016). Odds library. Stony Brook University, Department of Computer Sciences. Retrieved from https://odds.cs.stonybrook.edu.

Rousseeuw, P. J. (1984). Least median of squares regression. *Journal of the American statistical association*, *79*(388), 871–880.

Rousseeuw, P. J., & Leroy, A. M. (1987). *Robust regression and outlier detection*. John wiley & sons.

Salibian-Barrera, M. (2000). *Contributions to the theory of robust inference*. PhD thesis, University of British Columbia.

Salibián-Barrera, M., Van Aelst, S., & Willems, G. (2008). Fast and robust bootstrap. *Statistical Methods and Applications*, *17*, 41–71.

Salibian-Barrera, M., & Zamar, R. H. (2002). Bootstrapping robust estimates of regression. *Annals of Statistics*, 556–582.

Scott, D. W. (1979). On optimal and data-based histograms. *Biometrika*, *66*(3), 605–610.

Shao, J. (1990). Bootstrap estimation of the asymptotic variances of statistical functionals. *Annals of the Institute of Statistical Mathematics*, *42*, 737–752.

Shao, J. (1992). Bootstrap variance estimators with truncation. *Statistics & probability letters*, *15*(2), 95–101.

Shao, J., & Tu, D. (1995). *Theory for the Bootstrap*, (71–128). New York, NY: Springer New York.

Shimazaki, H., & Shinomoto, S. (2007). A method for selecting the bin size of a time histogram. *Neural computation*, *19*(6), 1503–1527.

Singh, K. (1998). Breakdown theory for bootstrap quantiles. *The annals of Statistics*, *26*(5), 1719–1732.

Stromberg, A. J. (1997). Robust covariance estimates based on resampling. *Journal of Statistical Planning and Inference*, *57*(2), 321–334.

Stromberg, A. J., Hössjer, O., & Hawkins, D. M. (2000). The least trimmed differences regression estimator and alternatives. *Journal of the American Statistical Association*, *95*(451), 853–864.

Sturges, H. A. (1926). The choice of a class interval. *Journal of the american statistical association*, *21*(153), 65–66.

Tang, J., Chen, Z., Fu, A. W.-C., & Cheung, D. W. (2002). Enhancing effectiveness of outlier detections for low density patterns. In *Advances in Knowledge Discovery and Data Mining: 6th Pacific-Asia Conference, PAKDD 2002 Taipei, Taiwan, May 6–8, 2002 Proceedings 6*, (535–548). Springer.

Thiele, C., & Hirschfeld, G. (2021). cutpointr: Improved estimation and validation of optimal cutpoints in r. *Journal of Statistical Software*, *98*(11), 1–27.

Tibshirani, R. J., & Efron, B. (1993). An introduction to the bootstrap. *Monographs on statistics and applied probability*, *57*(1), 1–436.

Tukey, J. W. et al. (1977). *Exploratory data analysis*, volume 2. Springer.

Venables, W. N., & Ripley, B. D. (2002). *Modern Applied Statistics with S* (Fourth ed.). New York: Springer. ISBN 0-387-95457-0.

Vlaski, S., Muma, M., & Zoubir, A. M. (2014). Robust bootstrap methods with an application to geolocation in harsh los/nlos environments. In *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, (7988–7992). IEEE.

Wand, M. (1997). Data-based choice of histogram bin width. *The American Statistician*, *51*(1), 59–64.

Wilcox, R. R. (1998). The goals and strategies of robust methods. *British Journal of Mathematical and Statistical Psychology*, *51*(1), 1–39.

Wilcox, R. R. (2017). *Introduction to robust estimation and hypothesis testing*. Academic press.

Wilkinson, L. (2017). Visualizing big data outliers through distributed aggregation. *IEEE transactions on visualization and computer graphics*, *24*(1), 256–266.

Yan, Y. (2024). *MLmetrics: Machine Learning Evaluation Metrics*. R package version 1.1.3.

Yildiztepe, E., & Özdemir, A. F. (2013). Asimetrik ve ağır kuyruklu dağılımların konum parametresinin bootstrap güven aralıkları için bir benzetim çalışması. *Anadolu Üniversitesi Bilim ve Teknoloji Dergisi: A-Uygulamalı Bilimler ve Mühendislik*, *14*(3), 213–229.

Yohai, V. J. (1987). High breakdown-point and high efficiency robust estimates for regression. *The Annals of statistics*, 642–656.

Yu, C., & Yao, W. (2017). Robust linear regression: A review and comparison. *Communications in Statistics-Simulation and Computation*, *46*(8), 6261–6282.

Zoubir, A. M., Koivunen, V., Ollila, E., & Muma, M. (2018). *Robust statistics for signal processing*. Cambridge University Press.