DOKUZ EYLÜL UNIVERSITY
GRADUATE SCHOOL OF NATURAL AND APPLIED
SCIENCES

JACKKNIFE-AFTER-BOOTSTRAP METHOD FOR DETECTION OF OUTLIERS AND INFLUENTIAL OBSERVATIONS IN LINEAR REGRESSION MODELS

by
Ufuk BEYAZTAŞ

June, 2012
İZMİR

JACKKNIFE-AFTER-BOOTSTRAP METHOD FOR DETECTION OF OUTLIERS AND INFLUENTIAL OBSERVATIONS IN LINEAR REGRESSION MODELS

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 Master of Science in Statistics

by

Ufuk BEYAZTAŞ

June, 2012
İZMİR

M.Sc THESIS EXAMINATION RESULT FORM

We have read the thesis entitled **"JACKKNIFE-AFTER-BOOTSTRAP METHOD FOR DETECTION OF OUTLIERS AND INFLUENTIAL OBSERVATIONS IN LINEAR REGRESSION MODELS"** completed by **UFUK BEYAZTAŞ** under supervision of **ASSOC. PROF AYLİN ALIN** and we certify that in our opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Master of Science.

Assoc. Prof. Aylin ALIN

Supervisor

Doç. Dr. Ali Kemal Şehirlioğlu

(Jury Member)

Asist. Prof. Dr. A. Fırat Özdemir

(Jury Member)

Prof.Dr. Mustafa SABUNCU
Director
Graduate School of Natural and Applied Sciences

ACKNOWLEDGMENTS

The words alone can not be express the thank to my supervisor, Assoc. Prof. Aylin ALIN, for her direction, assistance and guidance. In particular, her recommendations and suggestions have been invaluable for my thesis and my academic career, and this thesis would not have been possible without the support of her.

I would like to thank Prof. Dr. Serdar KURT for his helpful comments which improved the my thesis significantly, and I would like to thank to Assoc. Prof. Ali Kemal ŞEHİRLİOĞLU for his constructive comments as a member of my dissertation commite.

I also wish to thank to Dr. Sedat ÇAPAR, Dr. Engin YILDIZTEPE and Assist. Prof. A. Fırat ÖZDEMİR for their contributions and help for improving R coding in the simulation study.

Finally, special thanks also to Gülcan BEYAZTAŞ, my mother, to Müslüm BEYAZTAŞ, my father and to Hıdır BEYAZTAŞ, my brother for their support, understanding and endless love, through the duration of my studies.

Ufuk BEYAZTAŞ

JACKKNIFE-AFTER-BOOTSTRAP METHOD FOR DETECTION OF OUTLIERS AND INFLUENTIAL OBSERVATIONS IN LINEAR REGRESSION MODELS

ABSTRACT

In this thesis, the jackknife-after-bootstrap method which was proposed by Bradley Efron (1992) for estimating the standard errors and bias of a statistic, and proposed by Martin and Roberts in the context of influence diagnostics have been investigated. In addition, this method has been extended for several influence measures such as t-star, Likelihood Distance, Welsch' Distance and Modified Cook's Distance statistics. The therminology and algorithm of the method have been studied in detail. Performance of the proposed method has been evaluated with both real world examples and designed simulation studies. The results have been compared with the traditional version of the influence statistics. The simulations have been run by R 2.14.0. The sufficient bootstrap method proposed by Sing and Sedory (2011) has been combined with jackknife-after-bootstrap algorithm. We call this method as "sufficient jackknife-after-bootstrap" method. The same simulation studies and real-world examples have been carried out for this method, and the results were compared with conventional jackknife-after-bootstrap results.

Keywords: regression diagnostics, bootstrap, sufficient bootstrap, jackknife, influential observation.

DOĞRUSAL REGRESON MODELLERİNDE UÇ DEĞERLERİN VE ETKİN GÖZLEMLERİN BELİRLENMESİNDE BOOTSTRAPTEN-SONRA-JACKKNİFE YÖNTEMİ

ÖZ

Bu tezde, Bradley Efron (1992) tarafından istatistiğin standart hatasını ve yanlılığını tahmin etmek için önerilen ve ayrıca Martin ve Roberts (2006) tarafından etkin gözlemleri belirlemek için geliştirilen jackknife-after-bootstrap metodu incelenmiştir. Ek olarak, bu metot t-star, Likelihood Distance, Welsch's Distance ve Modified Cook's Distance gibi çeşitli etkinlik ölçümleri için genişletilmiştir. Bu metodun terminolojisi ve algoritması detaylı bir şekilde incelenmiştir. Önerilen metodun performansı gerçek dünya verileri ve simülasyon çalışmaları ile değerlendirilmiştir. Simülasyonlar R 2.14.0 programı kullanılarak çalıştırılmıştır. Sing ve Sedory (2011) tarafından önerilen sufficient bootstrap metodu jackknife-after-bootstrap algoritması ile birleştirilmiştir. Biz bu metodu sufficient jackknife-after-bootstrap metodu olarak adlandırıyoruz. Aynı simülasyon çalışmaları ve gerçek dünya örnekleri bu metot için çalıştırılmış ve sonuçları jackknife-after-bootstrap metodunun sonuçları ile karşılaştırılmıştır.

Anahtar sözcükler: regresyon tanı teşhisleri, bootstrap, yeterli bootstrap, jackknife, etkin gözlem.

CONTENTS

Page

CHAPTER ONE
INTRODUCTION

Detection and evaluation of influential observation/s is a critical part of data analysis in linear models. Since the computers were not as common or fast as they are now, and since the most of the calculations had to be performed by hand, it was very hard to make detailed examination of influential observations in the past. With the increased usage of computers, detecting influential observations has become an obligatory part of data analysis. The first studies were conducted by Cook (1977, 1979). Afterwards, they have been followed by Andrews and Pregibon (1978), Cook and Weisberg (1982), Belsley et al. (1980), Cook and Weisberg (1980), Welsch and Kuh (1977) and Welsh and Peters (1978). Most of the statistics developed by these authors, such as Cook’s Distance, DFFITS, DFBETAS, Andrew-Pregibon statistic, Likelihood Distance, Covariance Ratio, Cook-Weisberg Statistic, Welsch’s Distance and Modified Cook’s Distance, today have become an indispensable part of many statistical packages. Chattarjee and Hadi (1986) and Cook (1979) provide an excellent overview of research into regression diagnostics. The general idea of the all proposed measures is deleting the cases from the data one data point at a time. Then, the influence of each individual case is measured by comparing the full data analysis to the analysis with a case removed. Cases whose removal cause major changes in the analysis are called “*influential*”. Cut-off points are used to determine whether these changes are major or not.

Traditional methods have generally been used for identification and evaluation of influential observations and outliers. With the increased usage of computers, some methods which are better than traditional methods in general or in some situations were developed. One of the most important method among these is jackknife-after-bootstrap (JaB) method.

While the traditional usage of regression influence diagnostics is straightforward, the cut-offs suggested remain somewhat ad hoc (Martin and Roberts, 2010). Traditional methods work under the assumption of large sample theory and normal distribution, and therefore they work well when errors have a normal distribution and sample size is large enough. In the aforementioned cases, traditional methods work well for identification of influential observations. But, in case of non–normal error distributions or in case of small sample size, these methods may not be sufficient since they always use the same quantity as a cut-off point with the same sample sizes, irrespective of what might be known or suspected about the data generating process. In addition, the cut-offs calculated on the basis of large sample theory may not be accurate for small samples. To overcome these problems, Martin and Roberts (2010) proposed a variation of Efron (1979)'s well known bootstrap method.

Bootstrapping is a computer based method for assigning measures of accuracy to sample estimates (Efron and Tibshirani, 1993). This method is used to approximate the sampling distribution of a statistic. In the bootstrap method, bootstrap resamples of the data are obtained by random sampling with replacement from the original data set, and these resamples are assumed to be independent and identically distributed. Because of the construction method of bootstrap resamples, a point may appear multiple times in resamples. For example, the approximate proportion of resamples in which any given data point will appear j times is $(j!e)^{-1}$, meaning that a particular point fails to appear in about $(1-n^{-1})^n \to e^{-1} \approx$ 36.79% of resamples, appears only once in about e^{-1} of resamples, but appears multiple times in the remaining $1-2/e \approx$ 26.4% of resamples (Martin and Roberts 2010). Thus, if the original data set contains influential observations, these observations will potentially appear many times in the created sampling distributions and as a result, quantities calculated from those samples will not be satisfactory for comparison. In order to calculate the appropriate quantities, the cut-offs should be determined from the sampling distributions estimated using resamples not containing the point in question. Therefore, Martin and Roberts (2010) proposed jackknife-after-bootstrap (JaB) technique developed by Efron (1992). With this technique, which is fast and convenient for practitioners, the appropriate quantities can be calculated for both any individual data point and for all observations.

Bootstrap method has several advantages over the traditional methods. First, traditional methods are based on the large sample theory, and cut-offs calculated from these methods are affected by model size and the sample size. But, bootstrap method tries to approximate the sampling distribution and calculates the cut-off points, regardless of sample size. Second, while bootstrap allows for asymmetry in the sampling distributions of the diagnostic statistics, traditional methods assume that the distribution is symmetric. Traditional methods work well when the distribution of errors is normal and sample size is large enough, but when the distribution of errors is different from the normal distribution such as heavy tailed or skewed, this approximation may not be adequate to detect the actual influential observations. Since an influential observation arising from a certain underlying distribution does not have to be influential with respect to other underlying distributions, or a non-influential observation arising from a certain underlying distribution may be influential with respect to other underlying distributions, the observations detected as influential by the traditional methods in the different distribution cases may not be reasonable. This has been proven by both the study of Martin and Roberts (2010) and Beyaztas and Alin (2012). Of course, the bootstrap method automatically takes into account the features of the underlying distribution. In this thesis, several simulation studies and real-world examples were performed for Welsch’s Distance, Modified Cook’s distance, Likelihood Distance and t-star statistics, under normal, log-normal and t-distributions. The results which will be discussed in detail in Chapter 4 reveal that traditional methods flagged roughly the same number of influential observations in these three error distributions, while JaB method flagged fewer number of influential observations in skewed distribution case than normal distribution case. This result should not be surprising. A point flagged as influential in normal distribution does not have to be influential in skewed distribution, and logically fewer influential observations are expected for skewed distribution. Third advantage of the bootstrap method is that it combines the model information with the values of the diagnostic statistics to approximate the sampling distribution, while the traditional methods do not take into account the model information when the cut-offs are calculated.

Also in this thesis, we studied with sufficient jackknife-after-bootstrap method which is simply combination of the sufficient bootstrapping proposed by Sign and Sedory (2011) and jackknife-after-bootstrap algorithm. The methodology of the sufficient jackknife-after-bootstrap is the same as conventional JaB except the fact that only distinct units are used for sufficient version. Using sufficient bootstrapping into the jackknife-after-bootstrap algorithm provides important advantages for practitioners. This method and its advantages will be discussed in Chapter 3 and 4. Chapter 2 describes the influence measures used in the applications of this thesis, the methodology of these studies will be discussed in detailed in Chapter 3, and the simulation studies, real world examples, their results and discussions about these results are given in Chapter 4.

CHAPTER TWO
INFLUENTIAL OBSERVATION

In this chapter, we will investigate the linear regression model, influential observation and regression diagnostics used in this thesis which are Welsch's Distance, Modified Cook's Distance, Likelihood Distance and t-star statistics.

2.1 Linear Regression Model

The linear regression model used with influence measures throughout this thesis is

$$y_i = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \ldots + \beta_k x_{ik} + \varepsilon_i \text{, } i = 1,2,\ldots,n \tag{2.1}$$

This can be written in matrix form as

$$Y = X\beta + \varepsilon \tag{2.2}$$

where, Y is an $n \times 1$column vector for response variable, X is an $n \times p$ (p=k+1) fixed full-rank design matrix, β is an $p \times 1$ vector of unknown parameters including β_0, and $\boldsymbol{\varepsilon}$ is an $n \times 1$ error vector with zero mean and unknown variance σ^2. Using the method of least squares with the multiple linear regression model (2.1) we have;

$$\hat{\beta} = (X^T X)^{-1} X^T Y \tag{2.3}$$

$$Var(\hat{\beta}) = \sigma^2 (X^T X)^{-1} \tag{2.4}$$

$$\hat{Y} = X\hat{\beta} = PY \tag{2.5}$$

$$P = X(X^T X)^{-1} X^T \tag{2.6}$$

$$Var(\hat{Y}) = \sigma^2 P \tag{2.7}$$

$$e = Y - \hat{Y} = (I - P)Y \tag{2.8}$$

$$Var(e) = \sigma^2 (I - P) \tag{2.9}$$

$$\hat{\sigma}^2 = \frac{e^T e}{N - p} \tag{2.10}$$

These quantities can be influenced by one or a group of observations, but all observations do not have same impact over the least square regression outputs. For this reason, identification of influential observations is an important part of regression analysis, and this process is required to make a good inference. To identify influential observations, as mentioned above, several methods have been proposed.

Before examining these methods, we want to determine what is meant by influence. An influential observation is one which, either individually or together with several observations, has a demonstrably larger impact on the calculated values of various model features than is the case for other observations (Belsley et al. 1980). Existing diagnostic statistics explore the impact of the observations in various way. In general, the influence measures can be classified as follows;

- Measures based on the prediction matrix
- Measures based on the volume of confidence ellipsoids
- Measures based on influence functions, and
- Measures based on partial inference.

The rest of this chapter describes the influence measures used in this thesis. For more information about these measures and another measures, see Chatterjee and Hadi (1986).

2.2 t-star Statistic

Generally, the the least squared residual for the *ith* observation can be found as;

$$e_i = y_i - x_i \hat{\beta} \tag{2.11}$$

where x_i is the *ith* row of *X*. The standard error for this residual is

$$\sigma_{e_i} = \frac{e_i}{\sigma\sqrt{1-p_i}} \tag{2.12}$$

where p_i is the *ith* diagonal element of *P* given with (2.6). Two special cases of (2.12) are:

$$t_i = \sigma_{e_i} = \frac{e_i}{\hat{\sigma}\sqrt{1-p_i}} \tag{2.13}$$

where $\hat{\sigma}$ is defined in (2.10), and

$$t_i^* \equiv \frac{e_i}{\hat{\sigma}_{(i)}\sqrt{1-p_i}} \tag{2.14}$$

where

$$\hat{\sigma}_{(i)}^2 = \frac{Y_{(i)}^T(I-P_{(i)})Y_{(i)}}{(N-p-1)}$$

$$= \frac{(N-p)\hat{\sigma}^2}{(N-p-1)} - \frac{e_i^2}{(N-p-1)(1-p_i)} \tag{2.15}$$

So, equivalently the t_i^* statistic can be computed as follows;

$$t_i^* = t_i\sqrt{\frac{(N-p-1)}{(N-p-t_i^2)}} \tag{2.16}$$

This measure is based on residuals with and without *ith* observation, and is distributed approximately *t-distribution* with (*N*-*p*-*1*) degrees of freedom. That is the cut-off points for this measure approximately are $t_{\alpha/2,(N-p-1)}$.

2.3 The Likelihood Distance

Let $L(\hat{\beta})$ and $L(\hat{\beta}_{(i)})$ be the log likelihood functions at $\hat{\beta}$ and $\hat{\beta}_{(i)}$, respectively. A measure of the influence of the *ith* observation on $\hat{\beta}$ can be based on the distance between $L(\hat{\beta})$ and $L(\hat{\beta}_{(i)})$ (Cahtterjee and Hadi, 1986). The likelihood distance defined by Cook and Weisberg (1982) is

$$LD_i = 2\left|L(\hat{\beta}) - L(\hat{\beta}_{(i)}\right|$$

$$= N\log\left[\left(\frac{N}{N-1}\right)\frac{N-p-1}{t_i^{*2}+N-p-1}\right]+\frac{t_i^{*2}(N-1)}{(1-p_i)(N-p-1)}-1 \qquad (2.17)$$

This influence measure is based on the change in volume of confidence ellipsoids with and without the *ith* observation. The likelihood distance is related to the asymptotic confidence region, $\{\ \beta : 2\left[L(\hat{\beta})-L(\beta)\right] \le \chi^2_{\alpha,p+1}\ \}$ where $\chi^2_{\alpha,p+1}$ is the upper α point of the χ^2 distribution with (*p+1*) degrees of freedom (Chatterjee and Hadi, 1986). Hence, *LD*$_i$ is compared to χ^2_{p+1}.

2.4 Welsch's Distance

Welsch's Distance is based on the idea of influence function introduced by Hampel (1986, 1974) with and without ith observation,

$$IF_i(x_i; y_i; F; T)$$

$$\lim_{\varepsilon\to\infty}\frac{T\left[(1-\varepsilon)F+\varepsilon\delta_{x_i,y_i}\right]-T\left[F\right]}{\varepsilon} \qquad (2.18)$$

where $T(\cdot)$ is a vector-valued statistic, and is based on a random sample from the cumulative distribution function (cdf) of *F*, δ_{x_i,y_i} is the kronecher delta function which takes value of 1 at x_i, y_i and 0 otherwise. *IF*$_i$ measures the change in T caused by adding x_i, y_i to a very large sample. For a finite sample, several approximations exist including empirical influence curve, the sample influence curve and the sensitivity curve.

Let $\hat{F}$ be the empirical distribution function based on the full sample, and $\hat{F}_{(i)}$ be the empirical distribution function when the *ith* observation is omitted. The empirical influence curve (*EIC*) is

$$EIC_i = (N-1)(X_{(i)}^T X_{(i)})^{-1} x_i^T (y_i - x_i\hat{\beta}_{(i)})$$

$$= (N-1)(X^T X)^{-1} x_i^T \frac{e_i}{(1-p_i)^2} \qquad (2.19)$$

where

$$\hat{\beta}_{(i)} = (X_{(i)}^T X_{(i)})^{-1} X_{(i)}^T Y_{(i)} \tag{2.20}$$

is the estimate of β when the *ith* observation is omitted. Eg. (2.19) is obtained by replacing $\hat{F}_i$ by F and $T(\hat{F}_i)$. Omitting the limit in (2.18) and taking $F = \hat{F}$, $T(\hat{F}) = \hat{\beta},\ \ \varepsilon = -1/(N-1)$ gives the following formula for the sample influence curve.

$$SIC_i = (N-1)(X^T X)^{-1} x_i^T (y_i - x_i \hat{\beta}_{(i)})$$

$$= (N-1)(X^T X)^{-1} x_i^T \frac{e_i}{(1-p_i)} \tag{2.21}$$

On the other hand, setting $F = \hat{F}_{(i)}$, $T(\hat{F}_{(i)}) = \hat{\beta}_{(i)}$, and $\varepsilon = 1/N$ yields the sensitivity curve (*SC*).

$$SC_i = N(X^T X)^{-1} x_i^T \frac{e_i}{1-p_i} \tag{2.22}$$

To be able to order the observations in a meaningful way, *IF_i* vector must be normalized. The class of norms which are location/scale invariant is given by

$$D_i(M;c) = \frac{(IF_i)^T M (IF_i)}{c} \tag{2.23}$$

for any appropriate choice of *M* and *c* Chatterjee and Hadi (1986). If $D_i(M;c)$ is large it means that *ith* observation has strong influence on estimated coefficients relative to *M* and *c*. Using (2.19) to approximate (2.18) and setting $M = X_{(i)}^T X_{(i)}$ and $c = (N-1)\hat{\sigma}_{(i)}^2$, (2.23) becomes the Welsch Distance.

$$W_i^2 D_i (X_{(i)}^T X_{(i)}; (N-1)\hat{\sigma}_{(i)}^2)$$

$$= (N-1) t_i^{*2} \frac{p_i}{(1-p_i)^2} \tag{2.24}$$

Welsch (1982) suggested using W_i as a diagnostic tool and, $n > 15$, using $3\sqrt{p}$ as a cut-off point for W_i. Equivalently

$$W_i = WK_i \sqrt{\frac{N-1}{1-p_i}} \tag{2.25}$$

where $WK_i = \left| t_i^* \right| \sqrt{p_i /(1-p_i}$ also known as $DFFITS_i$.

2.5 Modified Cook's Distance

The measure is the modified version of the Cook's Distance proposed by Cook (1977).

$$C_i^* = \sqrt{D_i(X^T X; \frac{p(N-1)^2}{N-p}\hat{\sigma}_{(i)}^2}$$

$$= \left|t_i^*\right|\sqrt{\frac{N-p}{p}\frac{p_i}{1-p_i}} = WK_i\sqrt{\frac{N-p}{p}} \qquad (2.26)$$

The cut-off point for this measure is defined as $2\sqrt{\frac{N-p}{n}}$. A short summary of the influence measures used in this thesis is shown in Table 2.1.

Table 2.1 Influence measures

Influence measures	***Formulas***	***Cut-off points***
t-star (t^*)	$t_i\sqrt{\dfrac{n-p-1}{n-p-t_i^2}}$ *where* $t_i = \dfrac{y_i - x_i\hat{\beta}}{\hat{\sigma}\sqrt{1-p_i}}$ p_i =*ith* diagonal element of hat matrix $\boldsymbol{X(X'X)^{-1}X'}$	$\approx \pm t_{(n-p-1)}$
Welsch's Distance	$WK_i\sqrt{\dfrac{n-1}{1-p_i}}$ *where* $WK_i = t_i^*\sqrt{p_i \big/ (1-h_{ii})}$	$\pm 3\sqrt{p}$
Modified Cook's Distance	$WK_i\sqrt{\{(n-p)\}/p\}$	$\pm 2\sqrt{\sqrt{\{(n-p)\}/n\}}$
Likelihood Distance	$N\log\left\{\left(\dfrac{n}{n-1}\right)\left(\dfrac{n-p-1}{t_i^{*2}+n-p-1}\right)\right\}$ $+\dfrac{t_i^{*2}(n-1)}{(1-h_{ii})(n-p-1)}-1$	χ_p^2

CHAPTER THREE
METHODOLOGY

This chapter includes the history, methodology and algorithm of the methods used in this thesis.

3.1 The Bootstrap

The bootstrap, which was proposed by Bradley Efron (1979, 1981, 1982) and further developed by Efron and Tibshirani, is an one of the most important tool of modern statistical analysis. It establishes a general framework for simulation based statistical inference. There are two types of bootstrap methods: parametric and nonparametric. Our interest will be a nonparametric bootstrap. From now, we will simply call it as bootstrap. The main goal of bootstrap method is; to estimate the standart errors, bias and other measures of a statistic, and approximate the sampling distribution by re-sampling with replacement from the original sample. The most useful references about theory and applications of bootstrap are Efron and Tibshirani (1993), Davison and Hinkley (2005), and Hall (1995). In the bootstrap method, bootstrap re-samples of the data are obtained by random sampling with replacement from the original data set, and these re-samples are assumed to be independent and identically distributed (i.i.d.).

Let $Y_1,Y_2,...,Y_n$ be the i.i.d. random samples from unknown distribution F with parameter θ. The data $Y_1,Y_2,...,Y_n$ is used to estimate θ; $\hat{\theta}=\hat{\theta}(Y_1,Y_2,...,Y_n)$. Generally, we are interested in the distribution of $\hat{\theta}$ in order to provide standard errors, to construct confidence intervals, or to perform test of hypothesis. Using random samples taken from a population, we estimate the population parameter θ wheres in the bootstrap context, we try to estimate the parameter of the sampling distribution. That is, our population is now the original sample, and now we estimate the parameter of the sampling distribution $\hat{\theta}$. The general bootstrap idea is given step by step as follows;

- Let Y_1^*, Y_2^*,..., Y_n^* be the generated bootstrap re-samples with replacement from the original sample $Y_1,Y_2,...,Y_n$.
- Let $\hat{\theta}^*$ be the bootstrap estimates of $\hat{\theta}$.
- The first two steps are repeated for B times, say B = 1000, and B values of $\hat{\theta}_1^*, \hat{\theta}_2^*, ..., \hat{\theta}_B^*$ are obtained.

The empirical distribution of $\hat{\theta}^*$ is used to approximate the distribution of $\hat{\theta}$.

3.2 Sufficient Bootstrap

One of the most recent studies related to bootstrap is about sufficient bootstraping by Singh and Sedory (2011). The main idea underlying this is to use only distinct individual responses to estimate a statistic. Apart from the usage of only distinct individual responses, this technique is the same as conventional bootstrap. Singh and Sedory (2011) claim that the usage of the sufficient bootstrapping may help to reduce the amount of computation, and may results in better inference for certain cases than conventional bootstrapping. While the sufficient bootstrap uses only distinct observations, conventional bootstrap uses all of the observations in the re-samples. For this reason, the size of a sufficient bootstrap re-sample is smaller than the one obtained by conventional bootstrap. Every unit in a sample of size n has probability

$$1-(1-1/n)^n \tag{3.1}$$

to appear in a sufficient bootstrap resamples. So, the expected length of a sufficient bootstrap resample can be found as

$$[1-(1-1/n)^n]\times n \tag{3.2}$$

which causes sufficient bootstrap to be more advantageous than conventional bootstrap in terms of time and amount of computation. For example, for a sample with size 50, while the size of the conventional bootstrap re-samples are constantly 50, the size of the sufficient bootstrap re-samples are $[1-(1-1/50)^{50}]\times 50 \approx$ 31.80 in average. Therefore, the computing time is less than conventional bootstrap method. For more information about sufficient bootstrapping, see Singh and Sedory (2011).

3.3 The Jackknife

The jackknife technique is a cross-validation technique. First, Quenouille proposed the jackknife to estimate bias (1949), and Tukey named the technique the "jackknife" and used it to estimate standard errors (1958). Jackknife creates sample data sets from the data leaving out one or more observations at a time, and uses these samples to estimate bias and standard errors of a statistic. The jackknife procedure can be explained as follows. Let $\hat{\theta}_{(i)}$ be computed from the sample with the *ith* value deleted. Then the jackknife estimator calculated as

$$\hat{\theta}_J = \frac{1}{n}\sum_{i=1}^{n}\hat{\theta}_{(i)} \tag{3.3}$$

As mentioned above, the goal of the jackknife is to estimate a parameter of a population of interest from a random sample of data. More precisely, Let $X_1, X_2, ..., X_n$ be a data set of size *n*. Using jackknife we get *n* set of *n*-1 data. Let *T* be a function which is used to approximate the distribution of the data set $\hat{F}$. Let $\hat{F}_{(i)}$ be the emprical distribution of the data set where the *ith* observation deleted. That is, $\hat{\theta}_{(i)}$ = $T(\hat{F}_{(i)})$ which is the estimate of $T(\hat{F})$. The jackknife estimate which is the expected value of $T(\hat{F})$ is $\hat{T} = \frac{1}{n}\sum_{i=1}^{n} T(\hat{F}_{(i)})$.

3.4 Jackknife-after-Bootstrap

Jackknife-after-Bootstrap method was proposed by Bradley Efron (1992) for estimating the standard errors and bias of a statistic. This method was proposed by Martin and Roberts (2005) in the context of influence diagnostics.

Efron (1992) described the idea behind the JaB method as follows: a sample of size *n* from $y_1, y_2, ..., y_{i-1,} y_{i+1}, ..., y_n$ has the same distribution as a bootstrap sample from $y_1, y_2, ..., y_n$ in which none of the bootstrap values equals y_i. This method requires about *e* times more re-samples than regular bootstrap. For example, for any data set, if we want to determine whether an individual data point is influential or not, and to obtain 1000 re-samples without this individual data point, about 1000*e* $\approx$ 3000 re-samples are required. Then, these 1000 re-samples are used to construct the sampling distribution, and to determine the influence cut-offs. The algorithm of JaB method for detection of influential observations proposed by Martin and Roberts (2010) can be described as follows;

- Let θ_i be the diagnostic statistic that we study. The appropriate model is fitted for original data set, and the θ_i for *i*= 1, 2, …, *n* are calculated.
- Construct *B* re-samples with replacement from the original data set.
- For each data point within these *B* re-samples, get a subset of the samples which do not contain that data point, so there are *B/e* re-samples obtained for each data point. Calculate the *n* values of θ_i, *i* = 1, 2, …, *n*, for each of these resample, so *nB/e* values of θ_i are obtained. Collect all *nB/e* values of θ into a single vector.
- Suitable quantiles (say 2.5% and 97.5%) of this generated bootstrap distribution are determined. Percentiles of this distribution are then compared to the original θ_i, *i* = 1, 2, …, *n*, values to flag the points as influential or not.

The steps 1-4 are repeated *M* times. Then, the average and standard deviation for the number of flagged points for all these *M* simulations can be calculated. It should be noted that this algorithm runs only once for the real data.

As described by Martin and Roberts (2010), the rationale behind this approach is to generate a “null” bootstrap distribution of θ under the hypothesis that the *ith* data point is not influential. They propose that since the *ith* data point is not present in any of the re-samples from which this bootstrap distribution is generated, it cannot exert influence, and thus the distribution generated is free from the influence of this point.

3.5 Sufficient Jackknife-after-Bootstrap Method

The idea of sufficient bootstrapping is easily applicable in JaB method, and the all mentioned advantages apply to sufficient JaB method. As it is known, compared to traditional methods, JaB method requires too much computation. By implementing sufficient bootstrapping into the JaB method, similar results can be obtained with less calculation and less time, which is important for practitioners. The one of the purposes of this thesis is to study this hypothesis. The performance of JaB method and Sufficient JaB method as we call it were compared on both real world examples and simulated data sets for Welsch’s Distance, Modified Cook’s distance, Likelihood Distance and t-star statistics, under normal, log-normal and t-distributions. The results which will be discussed in detail in Chapter 4 reveal that the general behaviour of JaB does not change by adapting sufficient JaB. In addition, with the increase of the sample size the sufficient JaB performance gets better and for some cases sufficient JaB results are even better than conventional JaB results. The algorithm of the sufficient JaB is same as the conventional JaB. The only difference is that sufficient bootstrap is used rather than conventional bootstrap in the JaB method.

CHAPTER FOUR
RESULTS AND DISCUSSIONS

Two real-world examples and various designed simulation studies have been performed for traditional methods, conventional JaB and sufficient JaB methods, and the result have been compared. All calculations have been done by R 2.14.0.

4.1 Numerical Results for Conventional JaB

4.1.1 The life cycle savings data (Belsley et al., 1980, p.41)

The life cycle savings data for 50 countries are explained by per capita disposable income, the percentage rate of change in per-capita disposable income, and two demographic variables: the percentage of population less than 15 years old and the percentage of the population over 75 years old. The data were averaged over the decade 1960–1970 to remove the business cycle or other short-term fluctuations. The outliers in this example were already determined by traditional methods by Belsley et al (1980). For instance, Japan (23), Zambia (46) and Libya (49) flagged as outliers by using DFFITS, and Canada (6), Chile (7), South Rhodesia (37), United States (44), Zambia (46) and Libya (49) flagged as outliers by using COVRATIO. We use our proposed methods to flag influential observations. Influential observations in this data set were flagged by using both JaB and traditional methods, and the results are presented in Table 4.1. For this example, 3100 resamples were created from the original data set, so that roughly 1000 resamples without that point were produced for each data point.

JaB cutoffs are consistent with traditional cutoffs for Modified Cook's Distance and t-star statistics, but for Welsch's Distance and Likelihood Distance, JaB cutoffs are significantly different from traditional’s for all designs. JaB method flagged same points as influential as traditional method for Modified Cook’s Distance and t-star statistics. For Welsch’s Distance, JaB flagged Japan (23), Zambia (46) and Libya (49) as influential and traditional method flagged Japan (23) and Libya (49) but did not flag point 46. For Likelihood Distance, while JaB flagged Zambia (46) and Libya (49) as influential, traditional method did not flag any point as influential. The results of the proposed method in this study are consistent with the results in Belsley et al. (1980).

4.1.2 The Hertzsprung - Russell diagram of the star cluster data (Rousseauw and Leroy, 1987, p.27)

Data for the Hertzsprung - Russell diagram of the star cluster CYG OB1, which contains 47 stars in the direction of Cygnus from C. Doom. For this data-set, the explanatory variable (x) is the logarithm of the effective temperature at the surface of the star, and dependent variable (y) is the logarithm of its light intensity. Influential observations in this data set were flagged by using both JaB and traditional methods, and the results are presented in Table 4.2.

For Likelihood Distance and t-star statistics JaB results are better than traditional's. While traditional Likelihood Distance did not flag any point as influential, JaB method flagged point 34. In addition, while for t-star statistic, points 14, 17 and 34 were flagged as influential by JaB, traditional t-star flagged only points 14 and 17. For Welsch's and Modified Cook's Distances, it is difficult to discriminate the performances of JaB and traditional methods. For both distances, JaB flagged points 14 and 34. On the other hand, traditional Welsch's Distance only flagged point 14, and traditional Modified Cook's Distance flagged points 14, 20, 30 and 34 as influential. In this example, the results of JaB and traditional methods differ. These differences may be caused due to masking or swamping effects, but for this example, we are not interested in such of these situations. Using delete-d jackknife proposed by Martin et al. (2010) may be useful to find more reliable results and to eliminate the masking and swamping effects.

4.2 Numerical Results for Sufficient JaB

4.2.1 The life cycle savings data (Belsley et al., 1980, p.41)

In this example, sufficient JaB cutoffs are consistent with convential JaB cutoffs for Welsch's Distance, Modified Cook's Distance and t-star, but for Likelihood Distance, sufficient JaB cutoffs are significantly different from conventional JaB's for all designs. Sufficient JaB method flagged same points as influential as conventional JaB method for Modified Cook's Distance and t-star statistics. Welsch's Distance, conventional JaB flagged Japan (point 23), Zambia (46) and Libya (49) as influential, and sufficient JaB flagged Japan (23) and Libya (46) but did not flag point 46. For Likelihood Distance, while conventional JaB flagged Zambia (46) and Libya (49) as influential, sufficient JaB did not flag any point as influential as in the traditional case for Likelihood Distance. Our results reveal that the proposed method in this study is consistent with not only conventional JaB but also with traditional methods.

4.2.2 The Hertzsprung - Russell diagram of the star cluster data (Rousseauw and Leroy, 1987, p.27)

Apart from the Likelihood Distance conventional JaB and sufficient JaB results are the same. For Likelihood Distance, while conventional JaB method flagged point 34, sufficient JaB did not flag any point as influential. Note that, for this data set, it is difficult to identify actual influential observations because of the masking or swamping effects. Nevertheless, except for the Likelihood Distance, proposed method showed the same performance as conventional JaB.

4.3 Simulation Results for Conventional JaB

A simulation study was conducted to assess the performance of JaB and traditional methods for detection of influential observations under different sample sizes and various modeling scenarios based on the design of Martin and Roberts

Table 4.1 Conventional JaB Regression influence diagnostics for life cycle saving data, n=50, p=5

Method		***Welsch's Distance***	***Modified***	***Likelihood Distance***	***t-star***
			Cook's Distance		
Traditional					
	Low Cut-off	*-6.708*	*-1.897*		*-2.015*
	High Cut-off	*6.708*	*1.897*	*11.070*	*2.015*
	Points below	*49*	*49*		*7*
	Points above	*23*	*23, 46*	*None*	*46*
JaB					
	Low Cut-off	*-4.468*	*-1.769*		*-2.014*
	High Cut-off	*5.141*	*2.059*	*0.969*	*2.178*
	Points below	*49*	*49*		*7*
	Points above	*23, 46*	*23, 46*	*46, 49*	*46*

Table 4.2 Conventional JaB Regression influence diagnostics for Hertzsprung - Russell diagram of the star cluster data, n=47, p=2

Method		***Welsch's Distance***	***Modified***	***Likelihood Distance***	***t-star***
			Cook's Distance		
Traditional					
	Low Cut-off	*-4.242*	*-1.956*		*-2.015*
	High Cut-off	*4.242*	*1.956*	*5.991*	*2.015*
	Points below	*None*	*14,*		*14, 17*
	Points above	*30, 34*	*20, 30, 34*	*None*	*None*
JaB					
	Low Cut-off	*-2.391*	*-1.637*		*-1.906*
	High Cut-off	*5.381*	*3.387*	*0.589*	*1.667*
	Points below	*14*	*14*		*14, 17*
	Points above	*34*	*34*	*34*	*34*

Table 4.3 Sufficient JaB Regression influence diagnostics for life cycle saving data, n=50, p=5

Method		***Welsch's Distance***	***Modified Cook's Distance***	***Likelihood Distance***	***t-star***
Conventional JaB					
	Low Cut-off	*-4.470*	*-1.770*		*-2.013*
	High Cut-off	*5.134*	*2.062*	*0.971*	*2.179*
	Points below	*49*	*49*		*7*
	Points above	*23, 46*	*23, 46*	*46, 49*	*46*
Sufficient JaB					
	Low Cut-off	*-6.152*	*-1.966*		*-2.031*
	High Cut-off	*5.874*	*2.184*	*2.068*	*2.245*
	Points below	*49*	*49*		*7*
	Points above	*23*	*23, 46*	*None*	*46*

Table 4.4 Sufficient JaB Regression influence diagnostics for Hertzsprung - Russell diagram of the star cluster data, n=47, p=2

Method		***Welsch's Distance***	***Modified Cook's Distance***	***Likelihood Distance***	***t-star***
Conventional JaB					
	Low Cut-off	*-2.391*	*-1.637*		*-1.906*
	High Cut-off	*5.381*	*3.387*	*0.589*	*1.667*
	Points below	*14*	*14*		*14, 17*
	Points above	*34*	*34*	*34*	*34*
Sufficient JaB					
	Low Cut-off	*-2.530*	*-1.695*		*-1.966*
	High Cut-off	*6.902*	*4.008*	*1.318*	*1.727*
	Points below	*14*	*14*		*14, 17*
	Points above	*34*	*34*	*None*	*34*

(2010). We considered the cases (n, p) = (20, 2) for small sample, (n, p) = (50, 5) for large sample, and three error distributions: normal (N(0, 0.5625)), t(3) (heavy-tailed), and centered log-normal (1.5[exp{N(0, 0.5625)}– exp(1/2)]; skewed). The modeling scenarios are adapted such that no clear influential data points were deliberately generated, and a clearly influential data point was inserted into the data set. The regression model $Y = 1 + 2X + \varepsilon$ was used for small sample and $Y = 1 + 2X_1 + 4X_2 + 3X_3 + 2X_4 + \varepsilon$ for large sample. For each model X was generated as i.i.d N(2, 1) variates, and ε was generated with one of three error distributions mentioned above. The deliberately inserted influential point was at (x = 5, y = 2) for small sample and at (x_2 = 10, y = 10) for large sample. Simulation studies were carried out for four diagnostic statistics given in Table 2.1 as in real world examples. For each statistic, M = 500 simulations were performed, and for each case, a sample of size n was generated, and B = 3100 resamples were generated in each resampling operation, so that roughly 1000 resamples without that point were produced for each data point. The simulation study results are given with Tables 4.5-4.7. The average number of points flagged as influential for each simulation is recorded as “Average no. of points”. For deliberately inserted data point, the detection rate for all simulations recorded as “Percent of times point identified”. The standard deviations are given in brackets below.

In Table 4.5, the “influential point cut-off” values are the cut-off points belonging to sampling distributions which do not contain the deliberately inserted influential observation, and the “other cut-offs” values are the cut-off points belonging to sampling distributions containing the deliberately inserted influential observation. For normal errors, while influential point cut-offs are almost symmetric, the other cut-offs are not symmetric. The sampling distribution for other cut-offs contains the deliberately inserted data point. Hence, the percentiles of this distribution are affected by this point and become skewed in its direction. Since the skewness caused by the inserted data point is to the left, the other cut-offs become skewed to the left. In addition, because the JaB method takes into account the distribution structure, for log-normal errors, the cut-offs become asymmetric in the direction of error distribution. Since the influential point cut-offs calculated are free from the effect of inserted point, these values are affected only by the error distribution. That is, these values become skewed to the right which is the direction of the skewness of the log-normal distribution. The JaB method combines the skewness of the error distribution and skewness of the inserted data point. All of these changes are the result of internal scaling automatically performed by bootstrap distribution. The results of Tables 4.6 and 4.7 reveal that in general, traditional modified Cook’s distance and t-star measures do not seem to be heavily affected by the violation of normal error distribution. On the other hand, for n = 20 traditional Welch distance and likelihood distance detect more points as the distribution gets skewed. It is more obvious for likelihood distance. Tendency is same for JaB version of these two measures but with less increase. Their performance gets better as sample size increases. For n = 50, while traditional Welch distance and likelihood distance get affected by the asymmetry, their JaB versions flags less points which is logical since a point influential in normal error case may not actually influential in skewed error case. For both small and large samples, when no deliberate influential data point is inserted into the original data set, the generated JaB cut-offs are nearly symmetric and close to traditional cut-offs in normal error case. However, with inserted influential point, the generated JaB cut-offs are skewed in the direction of inserted point. But, traditional cut-offs remain the same. For the heavy-tailed distribution, the JaB distribution of the Modified Cook’s Distance and t-star tend to be heavier tailed than the traditional cut-offs. For the skewed error case, skewness of the JaB distribution is more clear for n = 20.

Even if there are no deliberately inserted influential points, some influential points may occur randomly. The results in Tables 4.6 and Table 4.7 show that traditional Modified Cook’s Distance and t-star measures successfully flag such points. The average number of points flagged by these measures is consistent with the results of DFBETAS given in Martin and Roberts (2010). Moreover, in general the deliberately inserted point did not have significant affect on the percentage of points flagged by JaB especially for n = 50, which is not surprising since randomly occurring points are likely to be less influential than deliberately inserted point and the bootstrap automatically scales the distribution for this.

Regardless of the error distributions, our method is promising for Welsch’s Distance and Likelihood Distance measures. For n = 20, when there is no deliberately inserted influential point, the traditional Welsch’s Distance and Likelihood Distance flagged small percentage of data points. However, the other traditional measures studied in this paper and the ones given in Martin and Ro
(2010) indicate more influential points. JaB method for the same measures gıve results consistent with other traditional measures including the ones in Martin and Roberts (2010). For n = 50 with no influential point present, traditional Likelihood Distance flagged no points for normal error case while the JaB Likelihood Distance flagged some points giving consistent results with other measures. Even if traditional Likelihood Distance flags some points for non-normal error cases, the percentage is still much less than JaB predicts. When an influential point is inserted into data set for n = 50, traditional Likelihood Distance flags only that point. However, even though the difference is not so significant, JaB Likelihood Distance flags more points, such points occurring at random. Apart from Welsch’s Distance and Likelihood Distance, traditional cut-offs are stringent especially for small samples and when the assumption for the normal-error distribution is not satisfied. On the other hand, with 5.991 traditional cut-off Likelihood Distance is too liberal compared to its JaB cut-off 1.560. For each of non-normal error distributions, JaB performed generally well by identifying the deliberately inserted influential point and even more points especially for Welsch’s Distance for n = 20.

For the real-world, it is hard to find data where the model assumptions are satisfied, so using traditional methods for these data may not be satisfactory for detecting influential observations. When model assumptions are not satisfied and the sample size is small, the results for traditional methods may be misleading and flag fewer points than is either desirable or prudent. To overcome this problem, we propose to use JaB method for detecting influential points. The simulation results in this study show that JaB is much more effective for Welch distance in terms of tendency for non-normal error cases. For likelihood distance, it adjusts the cut-off value to a value consistent with other measures. For both of these measures, the number of points flagged increase with JaB.

Table 4.5 Conventional JaB Low and High average cut-off points for influential point and other points – all simulations.

	Welsch's Distance		***t-star***		***Modified Cook's Distance***		***Likelihood Distance***	
	Influential point cut-off	***Other cut-offs***	***Influential point cut-off***	***Other cut-offs***	***Influential point cut-off***	***Other cut-offs***	***Influential point cut-off***	***Other cut-offs***
Normal errors, n=20, p=2								
	-3.385	*-8.253*	*-2.057*	*-2.561*	*-2.142*	*-4.809*		
	3.423	*3.082*	*2.068*	*1.920*	*2.150*	*1.951*	*1.589*	*4.323*
t(3) errors, n=20, p= 2								
	-3.164	*-8.101*	*-1.927*	*-2.474*	*-2.037*	*-4.680*		
	3.651	*3.063*	*2.252*	*2.003*	*2.269*	*1.964*	*1.734*	*4.385*
Log-normal errors, n=20, p=2								
	-2.686	*-7.216*	*-1.428*	*-2.070*	*-1.685*	*-4.145*		
	4.633	*3.642*	*2.961*	*2.567*	*2.962*	*2.373*	*3.011*	*5.896*
Normal errors, n=50, p=5								
	-5.270	*-7.011*	*-2.031*	*-2.174*	*-2.108*	*-2.703*		
	5.292	*4.854*	*2.007*	*1.872*	*2.103*	*1.944*	*1.050*	*1.527*
t(3) errors, n=50, p=5								
	-5.170	*-6.922*	*-1.959*	*-2.131*	*-2.055*	*-2.649*		
	5.451	*4.927*	*2.108*	*1.891*	*2.172*	*1.960*	*1.086*	*1.531*
Log-normal errors, n=50, p=5								
	-4.069	*-6.219*	*-1.402*	*-1.869*	*-1.584*	*-2.394*		
	7.013	*5.448*	*2.861*	*2.155*	*2.819*	*2.186*	*1.691*	*2.062*

Table 4.6 Conventional JaB Simulation results, n=20, p=2 for all distribution of errors.

Distribution of errors	***Normal***				***t(3)***				***Log-normal***			
Method	***Welsch's Distance***	***Modified Cook's Distance***	***Likelihood Distance***	***t-star***	***Welsch's Distance***	***Modified Cook's Distance***	***Likelihood Distance***	***t-star***	***Welsch's Distance***	***Modified Cook's Distance***	***Likelihood Distance***	***t-star***
Influential point not present												
Traditional												
Low cut-off	-4.242	-1.897		-2.100	-4.242	-1.897		-2.100	-4.242	-1.897		-2.100
High cut-off	4.242	1.897	5.991	2.100	4.242	1.897	5.991	2.100	4.242	1.897	5.991	2.100
Average no. of points (SD)	0.524 (0.643)	1.570 (0.937)	0.023 (0.152)	1.226 (0.734)	0.653 (0.704)	1.548 (0.912)	0.085 (0.279)	1.003 (0.659)	0.860 (0.658)	1.500 (0.746)	0.385 (0.491)	1.224 (0.519)
JaB												
Low cut-off	-3.387	-2.178		-2.062	-3.261	-1.990		-1.904	-2.654	-1.682		-1.466
High cut-off	3.391	2.107	1.560	2.088	3.682	2.313	1.828	2.270	4.620	2.863	2.827	2.904
Average no. of points (SD)	1.038 (0.696)	1.348 (0.714)	0.390 (0.487)	1.088 (0.699)	1.127 (0.687)	1.225 (0.664)	0.590 (0.497)	1.118 (0.663)	1.100 (0.651)	1.110 (0.628)	0.800 (0.415)	0.972 (0.509)
Influential point present												
Traditional												
Average no. of points (SD)	1.384 (0.520)	1.800 (0.665)	1.020 (0.120)	1.318 (0.483)	1.304 (0.472)	1.750 (0.616)	1.010 (0.104)	1.400 (0.529)	1.465 (0.563)	2.000 (0.682)	1.050 (0.343)	1.852 (0.647)
Percent of times point identified	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	0.998	1.000	0.914	0.980
JaB												
Low cut-off	-8.180	-4.770		-2.519	-8.001	-4.632		-2.432	-7.055	-4.057		-2.027
High cut-off	3.099	1.961	4.166	1.929	3.094	1.982	4.234	2.017	3.694	2.404	5.621	2.590
Average no. of points (SD)	1.443 (0.497)	1.650 (0.492)	1.000 (0.000)	1.256 (0.436)	1.346 (0.476)	1.500 (0.478)	1.000 (0.000)	1.226 (0.418)	1.474 (0.514)	1.800 (0.500)	1.025 (0.188)	1.492 (0.520)
Percent of times point identified	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	0.997	1.000	0.893	0.964

Table 4.7 Conventional JaB Simulation results, n=50, p=5 for all distribution of errors.

Distribution of errors	***Normal***				***t(3)***				***Log-normal***			
Method	***Welsch's Distance***	***Modified Cook's Distance***	***Likelihood Distance***	***t-star***	***Welsch's Distance***	***Modified Cook's Distance***	***Likelihood Distance***	***t-star***	***Welsch's Distance***	***Modified Cook's Distance***	***Likelihood Distance***	***t-star***
Influential point not present												
Traditional												
Low cut-off	-6.708	-1.897		-2.015	-6.708	-1.897		-2.015	-6.708	-1.897		-2.015
High cut-off	6.708	1.897	11.07	2.015	6.708	1.897	11.07	2.015	6.708	1.897	11.07	2.015
Average no. of points (SD)	1.106 (0.946)	3.572 (1.408)	None (0.000)	2.522 (1.062)	1.151 (0.895)	3.543 (1.380)	0.003 (0.051)	2.511 (1.041)	1.486 (0.821)	3.133 (1.231)	0.262 (0.440)	2.540 (0.885)
JaB												
Low cut-off	-5.299	-2.098		-2.018	-5.170	-2.056		-1.974	-4.060	-1.593		-1.397
High cut-off	5.318	2.086	1.043	2.020	5.484	2.146	1.055	2.085	6.951	2.812	1.774	2.876
Average no. of points (SD)	2.544 (0.856)	2.534 (0.854)	1.176 (0.686)	2.488 (0.805)	2.512 (0.896)	2.475 (0.896)	1.206 (0.635)	2.507 (0.829)	2.152 (0.851)	2.142 (0.833)	1.419 (0.548)	1.838 (0.796)
Influential point present												
Traditional												
Average no. of points (SD)	1.470 (0.614)	2.874 (1.032)	1.000 (0.000)	1.738 (0.713)	1.566 (0.671)	2.764 (1.038)	1.000 (0.000)	1.674 (0.675)	1.508 (0.631)	2.968 (1.123)	1.005 (0.072)	2.540 (0.885)
Percent of times point identified	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
JaB												
Low cut-off	-6.936	-2.678		-2.170	-6.847	-2.624		-2.126	-6.134	-2.363		-1.856
High cut-off	4.864	1.947	1.510	1.875	4.938	1.965	1.516	1.896	5.478	2.198	2.051	2.167
Average no. of points (SD)	2.038 (0.732)	2.070 (0.738)	1.134 (0.349)	1.745 (0.628)	2.084 (0.760)	1.930 (0.722)	1.127 (0.333)	1.674 (0.620)	1.832 (0.659)	1.890 (0.694)	1.109 (0.312)	1.776 (0.684)
Percent of times point identified	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000

4.4 Simulation Results for Sufficient JaB

The simulation design for conventional JaB was also applied for sufficient JaB method. For sufficient JaB, we considered the cases as (n, p) = (50, 5) and (n, p) = (100, 5). For this simulation study, conventional and sufficient JaB results are slightly different. This difference is because of sample size differences for conventional and sufficient JaB resamples. This difference gets much less as n becomes larger.

For the case (n, p) = (50, 5), when no deliberate influential data point is inserted into the original data set and for three error distributions, the average number of points flagged by sufficient JaB are close to the average number of points flagged by conventional JaB for Modified Cook's Distance and t-star statistics. For Welsch's Distance, there is a slight difference between conventional and sufficient JaB results, while difference is more significant for Likelihood Distance. With inserted influential point, sufficient JaB showed nearly the same performance as conventional JaB to flag influential points for Likelihood Distance. However, the other measures showed the same performance as in the first scenario. For the case (n, p) = (100, 5), sufficient JaB performed better than the first case (n, p) = (50, 5). Modified Cook's Distance and Likelihood Distance calculated based on sufficient bootstrap even flagged more points as influential than their counterparts based on conventional bootstrap under all three error distributions.

A question that comes to mind in a large samples, is whether the relative effects of unusual data points are diluted by the sheer number of "good" data points or not. But it is seen from the Table 4.12 that the deliberately inserted influential observation were flagged by both conventional and sufficient JaB (Percent of times point identified = 1.000 for all distribution of errors). In both scenarios, sufficient JaB showed almost the same performance with the smallest standard deviations as conventional JaB and traditional method. That is, the sufficient JaB results in this simulation ((n, p) = (250, 5)) are more efficient than both of the conventional JaB and traditional results. If n is sufficiently large, we expect that the bootstrap distribution of the statistics will approximately be the normal. Another notable point is in the results given in Table 4.12, in general, the sufficient JaB cut-offs are more symmetric than conventional JaB cut-offs.

Let $s_b = (k_1, k_2, ..., k_M)$ be a vector including the number of flagged influential observations in conventional bootstrap resamples and $s_{sb} = (l_1, l_2, ..., l_M)$ be a vector including the number of flagged influential observations in sufficient bootstrap resamples. The percent relative efficiency of the sufficient bootstrap estimator over the conventional bootstrap estimator is given by:

$$RE = \frac{V(s_b)}{V(s_{sb})} \times 100\% \tag{4.1}$$

The percent relative efficiency of the sufficient bootstrap over the conventional bootstrap are given in Table 4.13, 4.14 and 4.15 for sample sizes *n* =50, *n* = 100 and *n* = 250, respectively. Even though the size of sufficient JaB resamples are smaller than conventional JaB, in general, the percent relative efficiency $RE \geq 100$. Thus, the use of sufficient JaB may lead to more efficient results than conventional JaB.

As mentioned in Chapter 3, in general, since the number of observation in sufficient bootstrap resample is less than conventional bootstrap, the computing time is less than conventional bootstrap. R-software contains the R-function, *system.time* which calculates the computing time. To illustrate the time spent by conventional and sufficient JaB methods, computing times (in seconds) were recorded for a simulation where (*n*, *p*) = (100, 5) for all statistics. The results are given in Table 4.8. There is no doubt that, elapsed time for sufficient JaB is less than conventional JaB for all statistics.

Time spent by conventional bootstrap can be much more as the sample size gets larger. To see if it is true, we recorded the computing times both conventional and sufficient JaB simulations for Modified Cook's Distance where (*n*, *p*) = (250, 5) and M = 500. The computing times were recorded as roughly 93.54 hours for conventional JaB and 68.16 hours for sufficient JaB. As a conclusion, the computational burden of conventional JaB can be reduced by roughly %30 by using sufficient JaB.

Table 4.8 Elapsed time for all statistics, n=100, p=5

Method	***Welsch's Distance***	***Modified Cook's Distance***	***Likelihood Distance***	***t-star***
Conventional JaB	*114.18*	*112.06*	*114.43*	*109.40*
Sufficient JaB	*97.38*	*95.11*	*98.89*	*92.64*

For small sample sizes, sufficient JaB cut-offs are more liberal compared to conventional JaB cut-offs. For this reason, when the deliberately inserted data point appears in the original data set, conventional JaB flagged more points as influential than sufficient JaB in general. But, with the increase of the sample size, the results for sufficient JaB started to be the same as conventional JaB results with reduced computing times roughly by %30 and less standard deviation. To be brief, our study reveals that, sufficient JaB is a good competitor for conventional JaB with less amount of computation and time with more efficient results than conventional JaB.

Table 4.9 Sufficient JaB Low and High average cut-off points for influential point (point 1) and other points – all simulations.

Welsch's Distance		***t-star***		***Modified Cook's Distance***		***Likelihood Distance***	
Influence point Cut-off	***Other Cut-offs***	***Influence point Cut-off***	***Other Cut-offs***	***Influence point Cut-off***	***Other Cut-offs***	***Influence point Cut-off***	***Other Cut-offs***
Conventiona JaB Normal errors, n=50, p=5							
-5.270	*-7.011*	*-2.031*	*-2.174*	*-2.108*	*-2.703*		
5.292	*4.854*	*2.007*	*1.872*	*2.103*	*1.944*	*1.050*	*1.527*
Sufficient JaB Normal errors, n=50, p=5							
-6.092	*-8.428*	*-2.066*	*-2.505*	*-2.238*	*-3.006*		
6.010	*5.519*	*2.076*	*1.920*	*2.178*	*2.021*	*2.132*	*3.774*
Conventional JaB t(3) errors, n=50, p= 5							
-5.170	*-6.922*	*-1.959*	*-2.131*	*-2.055*	*-2.649*		
5.451	*4.927*	*2.108*	*1.891*	*2.172*	*1.960*	*1.086*	*1.531*
Sufficient JaB t(3) errors, n=50, p= 5							
-5.854	*-8.189*	*-1.995*	*-2.442*	*-2.123*	*-2.871*		
6.345	*5.629*	*2.177*	*1.957*	*2.307*	*2.051*	*2.145*	*3.861*
Conventional JaB Log-normal errors, n=50, p=5							
-4.069	*-6.219*	*-1.402*	*-1.869*	*-1.584*	*-2.394*		
7.013	*5.448*	*2.861*	*2.155*	*2.819*	*2.186*	*1.691*	*2.062*
Sufficient JaB Log-normal errors, n=50, p=5							
-4.524	*-7.489*	*-1.375*	*-2.084*	*-1.626*	*-2.583*		
8.508	*6.280*	*3.321*	*2.250*	*3.182*	*2.307*	*5.075*	*15.168*
Conventiona JaB Normal errors, n=100, p=5							
-4.841	*-4.808*	*-2.000*	*-1.875*	*-2.081*	*-2.611*		
4.907	*4.336*	*1.986*	*1.721*	*2.137*	*1.975*	*1.080*	*1.557*
Sufficient JaB Normal errors, n=100, p=5							
-5.231	*-5.208*	*-1.984*	*-1.869*	*-2.095*	*-2.098*		
5.238	*4.676*	*2.002*	*1.730*	*2.128*	*1.886*	*0.759*	*0.733*
Conventional JaB t(3) errors, n=100, p= 5							
-4.756	*-4.721*	*-1.961*	*-1.860*	*-2.101*	*-2.677*		
4.947	*4.409*	*2.023*	*1.735*	*2.120*	*1.949*	*1.069*	*1.524*
Sufficient JaB t(3) errors, n=100, p= 5							
-5.095	*-5.162*	*-1.985*	*-1.881*	*-2.088*	*-2.121*		
5.335	*4.628*	*2.055*	*1.768*	*2.147*	*1.887*	*0.7431*	*0.718*
Conventional JaB Log-normal errors, n=100, p=5							
-3.333	*-4.011*	*-1.230*	*-1.465*	*-1.609*	*-2.444*		
6.494	*4.818*	*2.813*	*2.027*	*2.833*	*2.179*	*1.794*	*2.158*
Sufficient JaB Log-normal errors, n=100, p=5							
-3.409	*-4.387*	*-1.190*	*-1.446*	*-1.423*	*-1.813*		
6.957	*5.327*	*2.985*	*2.097*	*2.877*	*2.113*	*1.342*	*1.055*

Table 4.10 Sufficient JaB Simulation results, n=50, p=5 for all distribution of errors.

Distribution of errors	***Normal***				***t(3)***				***Log-normal***			
Method	***Welsch's Distance***	***Modified Cook's Distance***	***Likelihood Distance***	***t-star***	***Welsch's Distance***	***Modified Cook's Distance***	***Likelihood Distance***	***t-star***	***Welsch's Distance***	***Modified Cook's Distance***	***Likelihood Distance***	***t-star***
<u>Influential point not present</u>												
Conventional JaB												
Low Cut-off	*-5.299*	*-2.098*		*-2.018*	*-5.170*	*-2.056*		*-1.974*	*-4.060*	*-1.593*		*1.397*
High Cut-off	*5.318*	*2.086*	*1.043*	*2.020*	*5.484*	*2.146*	*1.055*	*2.085*	*6.951*	*2.812*	*1.774*	*2.876*
Average no. of points	*2.544*	*2.534*	*1.176*	*2.488*	*2.512*	*2.475*	*1.206*	*2.507*	*2.152*	*2.142*	*1.419*	*1.838*
(SD)	*(0.856)*	*(0.854)*	*(0.686)*	*(0.805)*	*(0.896)*	*(0.896)*	*(0.635)*	*(0.829)*	*(0.851)*	*(0.833)*	*(0.548)*	*(0.796)*
Sufficient JaB												
Low Cut-off	*-6.207*	*-2.211*		*-2.079*	*-5.836*	*-2.130*		*-1.993*	*-4.540*	*-1.632*		*-1.371*
High Cut-off	*6.040*	*2.204*	*2.108*	*2.070*	*6.228*	*2.305*	*2.214*	*2.178*	*8.735*	*3.277*	*4.804*	*3.338*
Average no. of points	*1.516*	*2.145*	*0.127*	*2.197*	*1.436*	*2.156*	*0.147*	*2.254*	*1.344*	*1.835*	*0.596*	*1.583*
(SD)	*(0.628)*	*(0.710)*	*(0.334)*	*(0.642)*	*(0.697)*	*(0.675)*	*(0.355)*	*(0.627)*	*(0.615)*	*(0.650)*	*(0.491)*	*(0.675)*
<u>Influential point present</u>												
Conventional JaB												
Low Cut-off	*-6.936*	*-2.678*		*-2.170*	*-6.847*	*-2.624*		*-2.126*	*-6.134*	*-2.363*		*-1.856*
High Cut-off	*4.864*	*1.947*	*1.510*	*1.875*	*4.938*	*1.965*	*1.516*	*1.896*	*5.478*	*2.198*	*2.051*	*2.167*
Average no. of points	*2.038*	*2.070*	*1.134*	*1.745*	*2.084*	*1.930*	*1.127*	*1.674*	*1.832*	*1.890*	*1.109*	*1.776*
(SD)	*(0.732)*	*(0.738)*	*(0.349)*	*(0.628)*	*(0.760)*	*(0.722)*	*(0.333)*	*(0.620)*	*(0.659)*	*(0.694)*	*(0.312)*	*(0.684)*
Percent of times point identified	*1.000*	*1.000*	*1.000*	*1.000*	*1.000*	*1.000*	*1.000*	*1.000*	*1.000*	*1.000*	*1.000*	*1.000*
Sufficient JaB												
Low Cut-off	*-8.314*	*-2.970*		*-2.485*	*-8.077*	*-2.836*		*-2.422*	*-7.367*	*-2.545*		*-2.061*
High Cut-off	*5.529*	*2.025*	*2.132*	*1.923*	*5.643*	*2.056*	*2.145*	*1.962*	*6.316*	*2.321*	*5.075*	*2.265*
Average no. of points	*1.544*	*1.768*	*1.116*	*1.488*	*1.514*	*1.757*	*1.097*	*1.482*	*1.388*	*1.549*	*1.026*	*1.385*
(SD)	*(0.541)*	*(0.598)*	*(0.327)*	*(0.564)*	*(0.553)*	*(0.614)*	*(0.296)*	*(0.541)*	*(0.492)*	*(0.576)*	*(0.160)*	*(0.492)*
Percent of times point identified	*1.000*	*1.000*	*1.000*	*1.000*	*1.000*	*1.000*	*1.000*	*1.000*	*1.000*	*1.000*	*1.000*	*1.000*

Table 4.11 Sufficient JaB Simulation results, n=100, p=5 for all distribution of errors.

Distribution of errors	***Normal***				***t(3)***				***Log-normal***			
Method	***Welsch's Distance***	***Modified Cook's Distance***	***Likelihood Distance***	***t-star***	***Welsch's Distance***	***Modified Cook's Distance***	***Likelihood Distance***	***t-star***	***Welsch's Distance***	***Modified Cook's Distance***	***Likelihood Distance***	***t-star***
Influential point not present												
Conventional JaB												
Low Cut-off	*-4.911*	*-2.052*		*-1.978*	*-4.703*	*-2.005*		*-1.948*	*-3.530*	*-1.392*		*-1.233*
High Cut-off	*4.843*	*2.038*	*0.426*	*1.994*	*4.899*	*2.039*	*0.418*	*2.027*	*5.901*	*2.774*	*0.723*	*2.785*
Average no. of points	*5.320*	*5.130*	*2.590*	*5.050*	*5.160*	*5.010*	*2.630*	*5.000*	*4.230*	*4.230*	*2.480*	*3.480*
(SD)	*(1.043)*	*(0.812)*	*(0.753)*	*(1.038)*	*(1.051)*	*(1.058)*	*(0.824)*	*(0.994)*	*(0.851)*	*(1.135)*	*(0.673)*	*(1.049)*
Sufficient JaB												
Low Cut-off	*-5.081*	*-2.061*		*-2.007*	*-5.098*	*-2.088*		*-1.974*	*-3.404*	*-1.397*		*-1.205*
High Cut-off	*5.060*	*2.116*	*0.789*	*1.975*	*5.183*	*2.138*	*0.797*	*2.048*	*6.961*	*2.909*	*1.334*	*2.926*
Average no. of points	*3.900*	*4.790*	*0.510*	*4.720*	*4.020*	*4.730*	*0.630*	*4.780*	*3.460*	*4.080*	*1.330*	*3.390*
(SD)	*(0.745)*	*(0.807)*	*(0.627)*	*(0.877)*	*(0.852)*	*(0.789)*	*(0.525)*	*(0.773)*	*(0.914)*	*(0.981)*	*(0.603)*	*(0.993)*
Influential point present												
Conventional JaB												
Low Cut-off	*-4.809*	*-2.605*		*-1.877*	*-4.722*	*-2.671*		*-1.862*	*-4.001*	*-2.436*		*-1.462*
High Cut-off	*4.342*	*1.976*	*1.080*	*1.724*	*4.416*	*1.950*	*1.069*	*1.738*	*4.833*	*2.182*	*1.794*	*2.034*
Average no. of points	*2.990*	*2.400*	*1.010*	*1.930*	*3.150*	*2.190*	*1.040*	*1.960*	*3.540*	*2.510*	*1.090*	*2.980*
(SD)	*(0.926)*	*(0.953)*	*(0.100)*	*(0.807)*	*(1.028)*	*(0.950)*	*(0.196)*	*(0.777)*	*(0.957)*	*(0.858)*	*(0.287)*	*(1.053)*
Percent of times point identified	*1.000*	*1.000*	*1.000*	*1.000*	*1.000*	*1.000*	*1.000*	*1.000*	*1.000*	*1.000*	*1.000*	*1.000*
Sufficient JaB												
Low Cut-off	*-5.208*	*-2.098*		*-1.872*	*-5.160*	*-2.120*		*-1.883*	*-4.374*	*-1.808*		*-1.442*
High Cut-off	*4.683*	*1.889*	*0.759*	*1.733*	*4.636*	*1.890*	*0.743*	*1.772*	*5.341*	*2.120*	*1.342*	*2.104*
Average no. of points	*2.540*	*2.760*	*1.070*	*1.670*	*2.450*	*2.790*	*1.140*	*1.730*	*3.120*	*3.520*	*1.180*	*2.920*
(SD)	*(0.887)*	*(0.996)*	*(0.256)*	*(0.652)*	*(0.783)*	*(0.935)*	*(0.348)*	*(0.736)*	*(0.902)*	*(1.049)*	*(0.386)*	*(1.001)*
Percent of times point identified	*1.000*	*1.000*	*1.000*	*1.000*	*1.000*	*1.000*	*1.000*	*1.000*	*1.000*	*1.000*	*1.000*	*1.000*

Table 4.12 Sufficient JaB Simulation results for Modified Cook's Distance, n=250, p=5 for all distribution of errors

Method	***Traditional***			***Conventional JaB***			***Sufficient JaB***		
Distribution of errors	***Normal***	***t(3)***	***Log-normal***	***Normal***	***t(3)***	***Log-normal***	***Normal***	***t(3)***	***Log-normal***
Influential point not present									
Low Cut-off	*-1.979*	*-1.979*	*-1.979*	*-2.018*	*-2.017*	*-1.259*	*-2.048*	*-2.052*	*-1.243*
High Cut-off	*1.979*	*1.979*	*1.979*	*2.048*	*2.037*	*2.676*	*2.028*	*2.052*	*2.668*
Average no. of points (SD)	*13.860*	*13.760*	*11.220*	*12.820*	*12.650*	*10.970*	*12.280*	*12.135*	*11.173*
	(2.730)	*(2.404)*	*(2.634)*	*(1.351)*	*(1.666)*	*(1.696)*	*(1.064)*	*(1.069)*	*(1.378)*
Influential point present									
Low Cut-off	*-1.979*	*-1.979*	*-1.979*	*-1.725*	*-1,710*	*-1.310*	*-1.720*	*-1.736*	*-1.330*
High Cut-off	*1.979*	*1.979*	*1.979*	*1.675*	*1.698*	*2.031*	*1.736*	*1.732*	*2.073*
Average no. of points (SD)	*2.710*	*2.900*	*5.610*	*4.391*	*4.693*	*8.711*	*4.246*	*4.400*	*8.536*
	(1.200)	*(1.218)*	*(1.841)*	*(1.650)*	*(1.169)*	*(1.707)*	*(1.457)*	*(1.370)*	*(1.513)*
Percent of times point identified	*1.000*	*1.000*	*1.000*	*1.000*	*1.000*	*1.000*	*1.000*	*1.000*	*1.000*

Table 4.13 Sufficient JaB Relative efficiency, n=50, p=5 for all distribution of errors.

Distribution of errors	***Normal***				***t(3)***				***Log-normal***			
Method	***Welsch's Distance***	***Modified Cook's Distance***	***Likelihood Distance***	***t-star***	***Welsch's Distance***	***Modified Cook's Distance***	***Likelihood Distance***	***t-star***	***Welsch's Distance***	***Modified Cook's Distance***	***Likelihood Distance***	***t-star***
Influential point not present	*185.892*	*144.676*	*421.847*	*157.225*	*165.253*	*176.210*	*319.956*	*174.813*	*191.673*	*164.234*	*124.565*	*139.065*
Influential point present	*183.074*	*152.303*	*111.958*	*123.982*	*188.876*	*138.273*	*126.562*	*131.337*	*179.407*	*145.169*	*380.250*	*193.277*

Table 4.14 Sufficient JaB Relative efficiency, n=100, p=5 for all distribution of errors.

Distribution of errors	***Normal***				***t(3)***				***Log-normal***			
Method	***Welsch's Distance***	***Modified Cook's Distance***	***Likelihood Distance***	***t-star***	***Welsch's Distance***	***Modified Cook's Distance***	***Likelihood Distance***	***t-star***	***Welsch's Distance***	***Modified Cook's Distance***	***Likelihood Distance***	***t-star***
Influential point not present	*196.000*	*101.242*	*144.229*	*140.086*	*152.169*	*179.811*	*246.340*	*165.353*	*86.689*	*133.860*	*124.564*	*111.596*
Influential point present	*108.987*	*91.551*	*15.258*	*153.197*	*172.370*	*103.234*	*31.721*	*111.451*	*112.566*	*66.899*	*55.282*	*111.080*

Table 4.15 Sufficient JaB Relative efficiency for Modified Cook's Distance, n=100, p=5 for all distribution of errors.

Distribution of errors	***Normal***	***t(3)***	***Log-normal***
Influential point not present	*161.223*	*242.881*	*151.479*
Influential point present	*128.247*	*72.809*	*127.288*

CHAPTER FIVE
CONCLUSION

The main theme of this study is that using traditional methods in some situations mentioned in this thesis may not be sufficient since they always use the same quantity as a cut-off point irrespective of sample sizes and what might be known or suspected about the data generating process. The cut-offs calculated on the basis of large sample theory may not be accurate for small samples. To overcome this problem, based on the idea of Martin and Roberts (2006), we proposed the jackknife-after-bootstrap method for the process of identification of influential observations and outliers. Even though this method has a lot of advantages over the traditional methods, there is one main disadvantage which is the computational burden. To overcome this, we also proposed the sufficient jackknife-after-bootstrap method. To support our ideas, two real-world examples and various designed simulation studies were performed for traditional methods, conventional and sufficient JaB methods.

The results have showed that, when model assumptions are not satisfied and the sample size is small, the results for traditional methods may be misleading and flag fewer points than is either desirable or prudent. The simulation results in this study show that JaB is much more effective for Welch distance for non-normal error cases. For likelihood distance, it adjusts the cut-off value to a value consistent with other measures. For both of these measures, the number of points flagged increase with JaB. When no points inserted deliberately, having points flagged may seem confusing and as an error. However, the issue of flagging points is reasonable in a sense that some point will have the most extreme value of the measure and the solution would be to put tests on these points. (Martin, 2011 by personal contact). The same problem also holds for traditional methods but this time we have, at least, estimable error rate. Another solution would be using hybrid method which includes using the bootstrap cut-offs in general but then using traditional cut-offs as threshold. These solutions worth considering to be able to find some method to moderate the fact that the method being considered will always flag some points.

The results for sufficient JaB against conventional JaB showed that, for small sample sizes, sufficient JaB cut-offs are more liberal compared to conventional JaB cut-offs. For this reason, when the deliberately inserted data point appears in the original data set, conventional JaB flagged more points as influential than sufficient JaB in general. But, with the increase of the sample size, the results for sufficient JaB started to be the same as conventional JaB results with reduced computing times roughly by %30 and less standard deviation. To be brief, our study reveals that, sufficient JaB is a good competitor for conventional JaB with less amount of computation and time with more efficient results than conventional JaB.

REFERENCES

Andrews, D. F., & Pregibon, D. (1978). Finding the outliers that matter. *Journal of Royal Statistical Society*, *Series B*, *40* (1), 85-93.

Belsley, D. A., Kuh, E., & Welsch, R.E., (1980). *Regression diagnostics*. Ney Work: Wiley.

Beyaztas, U., & Alin, A. (2012). Jackknife-after-Bootstrap Method for Detection of Influential Observations in Linear Regression Models. *Communication in Statistics- Simulation and Computation*, (In press).

Chatterjee, S., & Hadi, A.S. (1986). Influential observations, high leverage points, and outliers in linear regression. *Statistical Science*, *1* (3), 379-416.

Chatterjee, S., & Hadi, A.S. (2006). *Regression analysis by example*. New Jersey: Wiley.

Cook, R.D. (1979). Influential observations in linear regression. *Journal of the American Statistical Association*, *74* (365), 169-174.

Cook, R. D., & Weisberg, S. (1980). Characterization of an empirical influence function for detecting influential cases in regression. *Technometrics*, *22* (4), 495-508.

Cook, R. D., & Weisberg, S. (1982). *Residuals and influence in regression*. New York: Chapman&Hall.

Davison, A.C. & Hinkley, D.V. (1997). *Bootstrap Methods and their Application*. Cambridge: Cambridge University Press.

Efron, B. (1979). Bootstrap methods: Another look at the Jackknife. *The Annals of Statistics*, *7* (1), 1-26.

Efron, B. (1992). Jackknife-after-bootstrap standard errors and influence functions. *Journal of Royal statistical Society*, *54* (1), 83-127.

Efron, B., & Tibshirani, R.J. (1993). *An introduction to bootstrap*. New York: Chapman & Hall

Hall, P. (1992). *The Bootstrap and Edgeworth Expansion*. Berlin: Springer.

Martin, M.A., & Roberts, S. (2006). An Evaluation of Bootstrap Methods for Outlier Detection in Least Squares Regression. *Journal of Applied Statistics, 33* (7), 703–720.

Martin, M.A., & Roberts, S. (2010). Jackknife-after-bootstrap regression influence diagnostics. *Journal of Nonparametric Statistics, 22* (2), 257-269.

Rousseeuw, P.J., & Leroy, A.M. (1987). *Robust regression and outlier detection*. New York: Wiley.

Sarjinder, S. & Stephen, A.S. (2011). Sufficient bootstrapping. *Computational Statistics and Data Analysis*, *55*, 1629-1637.

Welsch, R. E., & Kuh, E. (1977). Linear regression diagnostics. *Technical report*, 923-977. Sloan School of Management, MIT.

Welsch, R. E., & Peters, S. C. (1978). Finding influential subsets of data in regression models. *Proc. Eleventh Interface Symposium for Computer Science Statistics*, 240-244.

APPENDIX 1.

THE R CODES FOR SIMULATION STUDY

```
################### Conventional jackknife-after-bootstrap#################

system.time({                                        # to get computing time
output.percentile.general.low.Ci.star <- list ()
output.percentile.general.high.Ci.star <- list ()
output.percentile.1.low.Ci.star <- list ()           # these are required for getting
output.percentile.1.high.Ci.star <- list ()          # modified cook's distance
outputs
output.percentile.others.low.Ci.star <- list ()
output.percentile.others.high.Ci.star <- list ()
output.number.of.influential.observation.jab.Ci.star <- list ()
output.number.of.influential.observation.tra.Ci.star <- list ()
output.percentile.of.time.Ci.star <- list()
output.percentile.of.time.tra.Ci.star <- list()

output.percentile.general.low.Wi <- list ()
output.percentile.general.high.Wi <- list ()
output.percentile.1.low.Wi <- list ()                # these are required for getting
output.percentile.1.high.Wi <- list ()               # welsch's distance outputs
output.percentile.others.low.Wi <- list ()
output.percentile.others.high.Wi <- list ()
output.number.of.influential.observation.jab.Wi <- list ()
output.number.of.influential.observation.tra.Wi <- list ()
output.percentile.of.time.Wi <- list()
output.percentile.of.time.tra.Wi <- list()

output.percentile.general.high.LDi <- list ()
output.percentile.1.high.LDi <- list ()              # these are required for getting
output.percentile.others.high.LDi <- list ()         # likelihood distance outputs
output.number.of.influential.observation.jab.LDi <- list ()
output.number.of.influential.observation.tra.LDi <- list ()
output.percentile.of.time.LDi <- list()
output.percentile.of.time.tra.LDi <- list()

output.percentile.general.low.ti.star <- list ()
output.percentile.general.high.ti.star <- list ()
output.percentile.1.low.ti.star <- list ()           # these are required for getting
output.percentile.1.high.ti.star <- list ()          # t-star statistic outputs
output.percentile.others.low.ti.star <- list ()
output.percentile.others.high.ti.star <- list ()
output.number.of.influential.observation.jab.ti.star <- list ()
output.number.of.influential.observation.tra.ti.star <- list ()
output.percentile.of.time.ti.star <- list()
output.percentile.of.time.tra.ti.star <- list()
```

```
n = 50                          # number of observation in original data set
B = 3100                        # number of bootstrap

for (M in 1:500) {              # number of simulation

e <- rnorm(n=n, mean=0, sd=sqrt(0.5625))
x0 <- c(rep(1,n))
x1 <- rnorm(n=n,mean=2,sd=1)    # the creation of the original data set
x2 <- rnorm(n=n,mean=2,sd=1)    # step 1 begins from here
x3 <- rnorm(n=n,mean=2,sd=1)
x4 <- rnorm(n=n,mean=2,sd=1)
y <- 1+ 2*x1+4*x2+3*x3+2*x4+e
x2[1] = 10    #influential observarion
y[1] = 10     #influential observarion

X <- matrix(c(x0,x1,x2,x3,x4),ncol=5)
Y <- matrix(y,ncol=1)
Design.data <- cbind(X, Y)      # original data set

B.cap <- solve(crossprod(X)) %*% crossprod(X, Y)
P <- X %*% solve(crossprod(X)) %*% t(X)
Y.cap <- P %*% Y
e <- Y - Y.cap
dX <- nrow(X) - ncol(X)
var.cap <- crossprod(e) / (dX)              #
ei <- as.vector(Y - X %*% B.cap)            # matrix operations
pi <- diag(P)                               #
var.cap.i <- (((dX) * var.cap)/(dX - 1)) -
(ei^2/((dX - 1) * (1 - pi)))
ti <- ei / sqrt(var.cap * (1 - pi))
ti.star <- ei / sqrt(var.cap.i * (1 - pi))
pi.star <- pi + ei^2 / crossprod(e)
LDi <- nrow(X) *
log((nrow(X)/(nrow(X) - 1)) * ((dX - 1)/(ti.star^2 + dX - 1))) +
((ti.star^2 * (nrow(X) - 1)) / ((1 - pi) * (dX - 1))) - 1
WKi <- (ti.star)*sqrt(pi/(1-pi))
Wi <- WKi * sqrt((nrow(X)-1)/(1-pi))
Ci.star <- WKi* sqrt((dX)/(ncol(X)))

Comparing.table.Ci.star <- c(Ci.star)
Comparing.per.of.time.Ci.star <- Wi[1]
Comparing.table.Wi <- c(Wi)
Comparing.per.of.time.Wi <- Wi[1]
Comparing.table.LDi <- c(LDi)
Comparing.per.of.time.LDi <- LDi[1]
Comparing.table.ti.star <- c(ti.star)
Comparing.per.of.time.ti.star <- ti.star[1]     # end of step 1
```

```
for (j in 1:n) {                                # JaB steps start from here
result.Ci.star <- vector("list", )
result.ti.star <- vector("list", )
result.LDi <- vector("list", )
result.Wi <- vector("list", )

for( i in 1: B) {                               # number of bootstrap ( B )

data <- Design.data[sample(n,n,replace=TRUE),]   # bootstrap step, step 2 begins
dataX <- data[,1:ncol(X)]                        # from here
dataY <- data[,(ncol(X)+1)]

B.cap.simulation <- solve(crossprod(dataX)) %*% crossprod(dataX, dataY)
P.simulation <- dataX %*% solve(crossprod(dataX)) %*% t(dataX)
Y.cap.simulation <- P.simulation %*% dataY
e.simulation <- dataY - Y.cap.simulation
dX.simulation <- nrow(dataX) - ncol(dataX)
var.cap.simulation <- crossprod(e.simulation) / (dX.simulation)     # re-sample
ei.simulation <- as.vector(dataY - dataX %*% B.cap.simulation)   # operations
pi.simulation <- diag(P.simulation)
var.cap.i.simulation <- (((dX.simulation) * var.cap.simulation)/(dX.simulation - 1)) -
(ei.simulation^2/((dX.simulation - 1) * (1 - pi.simulation)))
ti.simulation <- ei.simulation / sqrt(var.cap.simulation * (1 - pi.simulation))
ti.star.simulation <- ei.simulation / sqrt(var.cap.i.simulation * (1 - pi.simulation))
pi.star.simulation <- pi.simulation + ei.simulation^2 / crossprod(e.simulation)
WKi.simulation <- (ti.star.simulation)*sqrt(pi.simulation/(1-pi.simulation))
Ci.star.simulation <- WKi.simulation* sqrt((dX.simulation)/(ncol(dataX)))
Wi.simulation <- WKi.simulation * sqrt((nrow(X)-1)/(1-pi.simulation))
LDi.simulation <- nrow(X) *
log((nrow(X)/(nrow(X) - 1)) * ((dX.simulation - 1)/(ti.star.simulation^2 +
dX.simulation - 1))) +
((ti.star.simulation^2 * (nrow(X) - 1)) / ((1 - pi.simulation) * (dX.simulation - 1))) - 1

result.Ci.star[[i]] <- list(outCi.star.simulation=(Ci.star.simulation),influ.obs = any
(dataY ==Y[j]))
result.ti.star[[i]] <- list(outti.star.simulation=(ti.star.simulation),influ.obs = any
(dataY ==Y[j]))
result.LDi[[i]] <- list(outLDi.simulation=(LDi.simulation),influ.obs = any (dataY
==Y[j]))
result.Wi[[i]] <- list(outWi.simulation=(Wi.simulation),influ.obs = any (dataY
==Y[j]))

}                                               # end of step 2

i.obs.Ci.star <- sapply(result.Ci.star,function(x) {x$influ.obs})      #
i.obs.ti.star <- sapply(result.ti.star,function(x) {x$influ.obs}) # step 3 and 4 begins
i.obs.LDi <- sapply(result.LDi,function(x) {x$influ.obs})          # from here
i.obs.Wi <- sapply(result.Wi,function(x) {x$influ.obs})            #
```

```
ni.result.Ci.star <- result.Ci.star[! i.obs.Ci.star]
ni.result.ti.star <- result.ti.star[! i.obs.ti.star]
ni.result.LDi <- result.LDi[! i.obs.LDi]
ni.result.Wi <- result.Wi[! i.obs.Wi]

ni.Ci.star.simulation <- sapply(ni.result.Ci.star,function(x)
{x$outCi.star.simulation})
if (j==1) {
ni.Ci.star.simulation1 <-  ni.Ci.star.simulation
}else if (j==2) {
ni.Ci.star.simulation49 <-  matrix(ni.Ci.star.simulation , nrow=1)

}else{
ni.i.star.simulation49 <-
cbind(ni.Ci.star.simulation49,matrix(ni.Ci.star.simulation,nrow=1))
}

ni.ti.star.simulation <- sapply(ni.result.ti.star,function(x) {x$outti.star.simulation})
if (j==1) {
ni.ti.star.simulation1 <-  ni.ti.star.simulation
}else if (j==2) {
ni.ti.star.simulation49 <-  matrix(ni.ti.star.simulation , nrow=1)

}else{
ni.ti.star.simulation49 <-
cbind(ni.ti.star.simulation49,matrix(ni.ti.star.simulation,nrow=1))
}

ni.LDi.simulation <- sapply(ni.result.LDi,function(x) {x$outLDi.simulation})
if (j==1) {
ni.LDi.simulation1 <-  ni.LDi.simulation
}else if (j==2) {
ni.LDi.simulation49 <-  matrix(ni.LDi.simulation , nrow=1)

}else{
ni.LDi.simulation49 <-
cbind(ni.LDi.simulation49,matrix(ni.LDi.simulation,nrow=1))
}

ni.Wi.simulation <- sapply(ni.result.Wi,function(x) {x$outWi.simulation})
if (j==1) {
ni.Wi.simulation1 <-  ni.Wi.simulation
}else if (j==2) {
ni.Wi.simulation49 <-  matrix(ni.Wi.simulation , nrow=1)
}else{
ni.Wi.simulation49 <-cbind(ni.Wi.simulation49,matrix(ni.Wi.simulation,nrow=1))
}

}                                   # end of step 3 and 4


# other calculations such as calculation of cut-off points are made with the rest of the
# code
full.data.Ci.star <- unlist(c(ni.Ci.star.simulation1,ni.Ci.star.simulation49))
full.data.ti.star <- unlist(c(ni.ti.star.simulation1,ni.ti.star.simulation49))
full.data.Wi <- unlist(c(ni.Wi.simulation1,ni.Wi.simulation49))
full.data.LDi <- unlist(c(ni.LDi.simulation1,ni.LDi.simulation49))

percentile.1.low.Ci.star <- quantile(unlist(ni.Ci.star.simulation1), 0.025)
percentile.1.high.Ci.star <- quantile(unlist(ni.Ci.star.simulation1), 0.975)
percentile.others.low.Ci.star <- quantile(unlist(ni.Ci.star.simulation49), 0.025)
percentile.others.high.Ci.star <- quantile(unlist(ni.Ci.star.simulation49), 0.975)
percentile.general.low.Ci.star <- quantile(full.data.Ci.star, 0.025)
percentile.general.high.Ci.star <- quantile(full.data.Ci.star, 0.975)
output.percentile.general.low.Ci.star <- c(output.percentile.general.low.Ci.star,
list(percentile.general.low.Ci.star))
output.percentile.general.high.Ci.star <- c(output.percentile.general.high.Ci.star,
list(percentile.general.high.Ci.star))
output.percentile.1.low.Ci.star <- c(output.percentile.1.low.Ci.star,
list(percentile.1.low.Ci.star))
output.percentile.1.high.Ci.star <- c(output.percentile.1.high.Ci.star,
list(percentile.1.high.Ci.star))
output.percentile.others.low.Ci.star <- c(output.percentile.others.low.Ci.star,
list(percentile.others.low.Ci.star))
output.percentile.others.high.Ci.star <- c(output.percentile.others.high.Ci.star,
list(percentile.others.high.Ci.star))

number.of.influential.observation.jab.Ci.star <- sum(sapply(Comparing.table.Ci.star,
function(x) (x < percentile.general.low.Ci.star||x > percentile.general.high.Ci.star)))
number.of.influential.observation.tra.Ci.star <- sum(sapply(Comparing.table.Ci.star,
function(x) (x < (-2) * sqrt((nrow(X)-ncol(X))/nrow(X))||x > 2 * sqrt((nrow(X)-
ncol(X))/nrow(X)))))

output.number.of.influential.observation.jab.Ci.star <-
c(output.number.of.influential.observation.jab.Ci.star,
list(number.of.influential.observation.jab.Ci.star))
output.number.of.influential.observation.tra.Ci.star <-
c(output.number.of.influential.observation.tra.Ci.star,
list(number.of.influential.observation.tra.Ci.star))
```

```
percentile.of.time.inf.Ci.star <- sum(sapply(Comparing.per.of.time.Ci.star,
function(x)              (x < percentile.general.low.Ci.star||x >
percentile.general.high.Ci.star)))
output.percentile.of.time.Ci.star <- c(output.percentile.of.time.Ci.star,
list(percentile.of.time.inf.Ci.star))
percentile.of.time.inf.tra.Ci.star <- sum(sapply(Comparing.per.of.time.Ci.star,
function(x)      (x < (-2) * sqrt((nrow(X)-ncol(X))/nrow(X))||x > 2 * sqrt((nrow(X)-
ncol(X))/nrow(X)))))
output.percentile.of.time.tra.Ci.star <- c(output.percentile.of.time.Ci.star,
list(percentile.of.time.inf.Ci.star))

percentile.1.low.ti.star <- quantile(unlist(ni.ti.star.simulation1), 0.025)
percentile.1.high.ti.star <- quantile(unlist(ni.ti.star.simulation1), 0.975)
percentile.others.low.ti.star <- quantile(unlist(ni.ti.star.simulation49), 0.025)
percentile.others.high.ti.star <- quantile(unlist(ni.ti.star.simulation49), 0.975)
percentile.general.low.ti.star <- quantile(full.data.ti.star, 0.025)
percentile.general.high.ti.star <- quantile(full.data.ti.star, 0.975)
output.percentile.general.low.ti.star <- c(output.percentile.general.low.ti.star,
list(percentile.general.low.ti.star))
output.percentile.general.high.ti.star <- c(output.percentile.general.high.ti.star,
list(percentile.general.high.ti.star))
output.percentile.1.low.ti.star <- c(output.percentile.1.low.ti.star,
list(percentile.1.low.ti.star))
output.percentile.1.high.ti.star <- c(output.percentile.1.high.ti.star,
list(percentile.1.high.ti.star))
output.percentile.others.low.ti.star <- c(output.percentile.others.low.ti.star,
list(percentile.others.low.ti.star))
output.percentile.others.high.ti.star <- c(output.percentile.others.high.ti.star,
list(percentile.others.high.ti.star))

number.of.influential.observation.jab.ti.star <- sum(sapply(Comparing.table.ti.star,
function(x)      (x < percentile.general.low.ti.star||x > percentile.general.high.ti.star)))
number.of.influential.observation.tra.ti.star <- sum(sapply(Comparing.table.ti.star,
function(x)      (x < -2.1||x > 2.1)))

output.number.of.influential.observation.jab.ti.star <-
c(output.number.of.influential.observation.jab.ti.star,
list(number.of.influential.observation.jab.ti.star))
output.number.of.influential.observation.tra.ti.star <-
c(output.number.of.influential.observation.tra.ti.star,
list(number.of.influential.observation.tra.ti.star))

percentile.of.time.inf.ti.star <- sum(sapply(Comparing.per.of.time.ti.star, function(x)
(x < percentile.general.low.ti.star||x > percentile.general.high.ti.star)))
output.percentile.of.time.ti.star <- c(output.percentile.of.time.ti.star,
list(percentile.of.time.inf.ti.star))
percentile.of.time.inf.tra.ti.star <- sum(sapply(Comparing.per.of.time.ti.star,
function(x) (x < -2.1||x > 2.1)))
```

```
output.percentile.of.time.tra.ti.star <- c(output.percentile.of.time.ti.star,
list(percentile.of.time.inf.ti.star))

percentile.1.high.LDi <- quantile(unlist(ni.LDi.simulation1), 0.950)
percentile.others.high.LDi <- quantile(unlist(ni.LDi.simulation49), 0.950)
percentile.general.high.LDi <- quantile(full.data.LDi, 0.950)
output.percentile.general.high.LDi <- c(output.percentile.general.high.LDi,
list(percentile.general.high.LDi))
output.percentile.1.high.LDi <- c(output.percentile.1.high.LDi,
list(percentile.1.high.LDi))
output.percentile.others.high.LDi <- c(output.percentile.others.high.LDi,
list(percentile.others.high.LDi))

number.of.influential.observation.jab.LDi <- sum(sapply(Comparing.table.LDi,
function(x)     (x > percentile.general.high.LDi)))
number.of.influential.observation.tra.LDi <- sum(sapply(Comparing.table.LDi,
function(x)     (x > qchisq(.95, ncol(X), lower.tail=T))))

output.number.of.influential.observation.jab.LDi <-
c(output.number.of.influential.observation.jab.LDi,
list(number.of.influential.observation.jab.LDi))
output.number.of.influential.observation.tra.LDi <-
c(output.number.of.influential.observation.tra.LDi,
list(number.of.influential.observation.tra.LDi))

percentile.of.time.inf.LDi <- sum(sapply(Comparing.per.of.time.LDi, function(x)
(x > percentile.general.high.LDi)))
output.percentile.of.time.LDi <- c(output.percentile.of.time.LDi,
list(percentile.of.time.inf.LDi))
percentile.of.time.inf.tra.LDi <- sum(sapply(Comparing.per.of.time.LDi, function(x)
(x > qchisq(.95, ncol(X), lower.tail=T))))
output.percentile.of.time.tra.LDi <- c(output.percentile.of.time.LDi,
list(percentile.of.time.inf.LDi))

percentile.1.low.Wi <- quantile(unlist(ni.Wi.simulation1), 0.025)
percentile.1.high.Wi <- quantile(unlist(ni.Wi.simulation1), 0.975)
percentile.others.low.Wi <- quantile(unlist(ni.Wi.simulation49), 0.025)
percentile.others.high.Wi <- quantile(unlist(ni.Wi.simulation49), 0.975)
percentile.general.low.Wi <- quantile(full.data.Wi, 0.025)
percentile.general.high.Wi <- quantile(full.data.Wi, 0.975)
output.percentile.general.low.Wi <- c(output.percentile.general.low.Wi,
list(percentile.general.low.Wi))
output.percentile.general.high.Wi <- c(output.percentile.general.high.Wi,
list(percentile.general.high.Wi))
output.percentile.1.low.Wi <- c(output.percentile.1.low.Wi, list(percentile.1.low.Wi))
output.percentile.1.high.Wi <- c(output.percentile.1.high.Wi,
list(percentile.1.high.Wi))
```

```
output.percentile.others.low.Wi <- c(output.percentile.others.low.Wi,
list(percentile.others.low.Wi))
output.percentile.others.high.Wi <- c(output.percentile.others.high.Wi,
list(percentile.others.high.Wi))

number.of.influential.observation.jab.Wi <- sum(sapply(Comparing.table.Wi,
function(x)      (x < percentile.general.low.Wi||x > percentile.general.high.Wi)))
number.of.influential.observation.tra.Wi <- sum(sapply(Comparing.table.Wi,
function(x)      (x < (-3) * sqrt(ncol(X))||x > 3 * sqrt(ncol(X)))))

output.number.of.influential.observation.jab.Wi <-
c(output.number.of.influential.observation.jab.Wi,
list(number.of.influential.observation.jab.Wi))
output.number.of.influential.observation.tra.Wi <-
c(output.number.of.influential.observation.tra.Wi,
list(number.of.influential.observation.tra.Wi))

percentile.of.time.inf.Wi <- sum(sapply(Comparing.per.of.time.Wi, function(x)
(x < percentile.general.low.Wi||x > percentile.general.high.Wi)))
output.percentile.of.time.Wi <- c(output.percentile.of.time.Wi,
list(percentile.of.time.inf.Wi))
percentile.of.time.inf.tra.Wi <- sum(sapply(Comparing.table.Wi, function(x) (x < (-
3) * sqrt(ncol(X))||x > 3 * sqrt(ncol(X)))))
output.percentile.of.time.tra.Wi <- c(output.percentile.of.time.Wi,
list(percentile.of.time.inf.Wi))

}
output.genr.per.low.Ci.star <- do.call(rbind.data.frame,
output.percentile.general.low.Ci.star)
names(output.genr.per.low.Ci.star) = c("per.general.low.Ci.star")
output.genr.per.high.Ci.star <- do.call(rbind.data.frame,
output.percentile.general.high.Ci.star)
names(output.genr.per.high.Ci.star) = c("per.general.high.Ci.star")
output.low.per.1.Ci.star <- do.call(rbind.data.frame, output.percentile.1.low.Ci.star)
names(output.low.per.1.Ci.star) = c("per.poi1.low.Ci.star")
output.upp.per.1.Ci.star <- do.call(rbind.data.frame, output.percentile.1.high.Ci.star)
names(output.upp.per.1.Ci.star) = c("per.poi1.high.Ci.star")
output.low.others.Ci.star <- do.call(rbind.data.frame,
output.percentile.others.low.Ci.star)
names(output.low.others.Ci.star) = c("per.others.low.Ci.star")
output.upp.others.Ci.star <- do.call(rbind.data.frame,
output.percentile.others.high.Ci.star)
names(output.upp.others.Ci.star) = c("per.others.high.Ci.star")
output.number.of.inf.obs.jab.Ci.star <- do.call(rbind.data.frame,
output.number.of.influential.observation.jab.Ci.star)
names(output.number.of.inf.obs.jab.Ci.star) = c("number.of.inf.jab.Ci.star")
output.number.of.inf.obs.tra.Ci.star <- do.call(rbind.data.frame,
output.number.of.influential.observation.tra.Ci.star)
```

```
names(output.number.of.inf.obs.tra.Ci.star) = c("number.of.inf.tra.Ci.star")
output.percent.of.time.Ci.star <- do.call(rbind.data.frame,
output.percentile.of.time.Ci.star)
names(output.percent.of.time.Ci.star) = c("percentile.of.time.Ci.star")
result.per.of.time.Ci.star <-  sum(output.percent.of.time.Ci.star) /
nrow(output.percent.of.time.Ci.star)
output.percent.of.time.tra.Ci.star <- do.call(rbind.data.frame,
output.percentile.of.time.tra.Ci.star)
names(output.percent.of.time.tra.Ci.star) = c("percentile.of.time.tra.Ci.star")
result.per.of.time.tra.Ci.star <-  sum(output.percent.of.time.tra.Ci.star) /
nrow(output.percent.of.time.tra.Ci.star)

output.genr.per.low.ti.star <- do.call(rbind.data.frame,
output.percentile.general.low.ti.star)
names(output.genr.per.low.ti.star) = c("per.general.low.ti.star")
output.genr.per.high.ti.star <- do.call(rbind.data.frame,
output.percentile.general.high.ti.star)
names(output.genr.per.high.ti.star) = c("per.general.high.ti.star")
output.low.per.1.ti.star <- do.call(rbind.data.frame, output.percentile.1.low.ti.star)
names(output.low.per.1.ti.star) = c("per.poi1.low.ti.star")
output.upp.per.1.ti.star <- do.call(rbind.data.frame, output.percentile.1.high.ti.star)
names(output.upp.per.1.ti.star) = c("per.poi1.high.ti.star")
output.low.others.ti.star <- do.call(rbind.data.frame,
output.percentile.others.low.ti.star)
names(output.low.others.ti.star) = c("per.others.low.ti.star")
output.upp.others.ti.star <- do.call(rbind.data.frame,
output.percentile.others.high.ti.star)
names(output.upp.others.ti.star) = c("per.others.high.ti.star")
output.number.of.inf.obs.jab.ti.star <- do.call(rbind.data.frame,
output.number.of.influential.observation.jab.ti.star)
names(output.number.of.inf.obs.jab.ti.star) = c("number.of.inf.jab.ti.star")
output.number.of.inf.obs.tra.ti.star <- do.call(rbind.data.frame,
output.number.of.influential.observation.tra.ti.star)
names(output.number.of.inf.obs.tra.ti.star) = c("number.of.inf.tra.ti.star")
output.percent.of.time.ti.star <- do.call(rbind.data.frame,
output.percentile.of.time.ti.star)
names(output.percent.of.time.ti.star) = c("percentile.of.time.ti.star")
result.per.of.time.ti.star <-  sum(output.percent.of.time.ti.star) /
nrow(output.percent.of.time.ti.star)
output.percent.of.time.tra.ti.star <- do.call(rbind.data.frame,
output.percentile.of.time.tra.ti.star)
names(output.percent.of.time.tra.ti.star) = c("percentile.of.time.tra.ti.star")
result.per.of.time.tra.ti.star <-  sum(output.percent.of.time.tra.ti.star) /
nrow(output.percent.of.time.tra.ti.star)

output.genr.per.high.LDi <- do.call(rbind.data.frame,
output.percentile.general.high.LDi)
names(output.genr.per.high.LDi) = c("per.general.high.LDi")
```

```
output.upp.per.1.LDi <- do.call(rbind.data.frame, output.percentile.1.high.LDi)
names(output.upp.per.1.LDi) = c("per.poi1.high.LDi")
output.upp.others.LDi <- do.call(rbind.data.frame, output.percentile.others.high.LDi)
names(output.upp.others.LDi) = c("per.others.high.LDi")
output.number.of.inf.obs.jab.LDi <- do.call(rbind.data.frame,
output.number.of.influential.observation.jab.LDi)
names(output.number.of.inf.obs.jab.LDi) = c("number.of.inf.jab.LDi")
output.number.of.inf.obs.tra.LDi <- do.call(rbind.data.frame,
output.number.of.influential.observation.tra.LDi)
names(output.number.of.inf.obs.tra.LDi) = c("number.of.inf.tra.LDi")
output.percent.of.time.LDi <- do.call(rbind.data.frame,
output.percentile.of.time.LDi)
names(output.percent.of.time.LDi) = c("percentile.of.time.LDi")
result.per.of.time.LDi <-  sum(output.percent.of.time.LDi) /
nrow(output.percent.of.time.LDi)
output.percent.of.time.tra.LDi <- do.call(rbind.data.frame,
output.percentile.of.time.tra.LDi)
names(output.percent.of.time.tra.LDi) = c("percentile.of.time.tra.LDi")
result.per.of.time.tra.LDi <-  sum(output.percent.of.time.tra.LDi) /
nrow(output.percent.of.time.tra.LDi)

output.genr.per.low.Wi <- do.call(rbind.data.frame,
output.percentile.general.low.Wi)
names(output.genr.per.low.Wi) = c("per.general.low.Wi")
output.genr.per.high.Wi <- do.call(rbind.data.frame,
output.percentile.general.high.Wi)
names(output.genr.per.high.Wi) = c("per.general.high.Wi")
output.low.per.1.Wi <- do.call(rbind.data.frame, output.percentile.1.low.Wi)
names(output.low.per.1.Wi) = c("per.poi1.low.Wi")
output.upp.per.1.Wi <- do.call(rbind.data.frame, output.percentile.1.high.Wi)
names(output.upp.per.1.Wi) = c("per.poi1.high.Wi")
output.low.others.Wi <- do.call(rbind.data.frame, output.percentile.others.low.Wi)
names(output.low.others.Wi) = c("per.others.low.Wi")
output.upp.others.Wi <- do.call(rbind.data.frame, output.percentile.others.high.Wi)
names(output.upp.others.Wi) = c("per.others.high.Wi")
output.number.of.inf.obs.jab.Wi <- do.call(rbind.data.frame,
output.number.of.influential.observation.jab.Wi)
names(output.number.of.inf.obs.jab.Wi) = c("number.of.inf.jab.Wi")
output.number.of.inf.obs.tra.Wi <- do.call(rbind.data.frame,
output.number.of.influential.observation.tra.Wi)
names(output.number.of.inf.obs.tra.Wi) = c("number.of.inf.tra.Wi")
output.percent.of.time.Wi <- do.call(rbind.data.frame, output.percentile.of.time.Wi)
names(output.percent.of.time.Wi) = c("percentile.of.time.Wi")
result.per.of.time.Wi <-  sum(output.percent.of.time.Wi) /
nrow(output.percent.of.time.Wi)
output.percent.of.time.tra.Wi <- do.call(rbind.data.frame,
output.percentile.of.time.tra.Wi)
names(output.percent.of.time.tra.Wi) = c("percentile.of.time.tra.Wi")
```

```
result.per.of.time.tra.Wi <-  sum(output.percent.of.time.tra.Wi) /
nrow(output.percent.of.time.tra.Wi)
})

colMeans(output.genr.per.low.Wi)   #average of all lower cut-offs of welsch's
distance
colMeans(output.genr.per.high.Wi)  #average of all upper cut-offs of welsch's
distance
colMeans(output.low.per.1.Wi)       #average of all lower cut-offs of re-samples
which do not ################################contain the deliberately inserted
data point
colMeans(output.upp.per.1.Wi)       #average of all upper cut-offs of re-samples
which do not ################################contain the deliberately inserted
data point
colMeans(output.low.others.Wi)      #average of all lower cut-offs of re-samples
which do not ################################contain each data point ( except
inserted point)
colMeans(output.upp.others.Wi)      #average of all upper cut-offs of re-samples
which do not ################################contain each data point ( except
inserted point)
colMeans(output.number.of.inf.obs.jab.Wi) #number of flagged influential
observations by ######################################jackknife-after-
bootstrap welsch's distance
sqrt(var(output.number.of.inf.obs.jab.Wi))   #stdev of flagged influential
observations
colMeans(output.number.of.inf.obs.tra.Wi)  #number of flagged influential
observations by ######################################traditional welsch's
distance
sqrt(var(output.number.of.inf.obs.tra.Wi))   # stdev of flagged influential
observations
result.per.of.time.Wi                 #Percent of times point identified of JaB method
sqrt(var(result.per.of.time.Wi))      #stdev of Percent of times point identified of
JaB
result.per.of.time.tra.Wi             #Percent of times point identified of traditional
method
sqrt(var(result.per.of.time.tra.Wi))  # stdev of Percent of times point identified of
traditional

colMeans(output.genr.per.low.Ci.star)
colMeans(output.genr.per.high.Ci.star)
colMeans(output.low.per.1.Ci.star)
colMeans(output.upp.per.1.Ci.star)
colMeans(output.low.others.Ci.star)          # results for modified cook's distance
colMeans(output.upp.others.Ci.star)
colMeans(output.number.of.inf.obs.jab.Ci.star)
sqrt(var(output.number.of.inf.obs.jab.Ci.star))
colMeans(output.number.of.inf.obs.tra.Ci.star)
sqrt(var(output.number.of.inf.obs.tra.Ci.star))
```

```
result.per.of.time.Ci.star
sqrt(var(result.per.of.time.Ci.star))
result.per.of.time.tra.Ci.star
sqrt(var(result.per.of.time.tra.Ci.star))
colMeans(output.genr.per.low.ti.star)
colMeans(output.genr.per.high.ti.star)
colMeans(output.low.per.1.ti.star)
colMeans(output.upp.per.1.ti.star)
colMeans(output.low.others.ti.star)          # results for t-star statistic
colMeans(output.upp.others.ti.star)
colMeans(output.number.of.inf.obs.jab.ti.star)
sqrt(var(output.number.of.inf.obs.jab.ti.star))
colMeans(output.number.of.inf.obs.tra.ti.star)
sqrt(var(output.number.of.inf.obs.tra.ti.star))
result.per.of.time.ti.star
sqrt(var(result.per.of.time.ti.star))
result.per.of.time.tra.ti.star
sqrt(var(result.per.of.time.tra.ti.star))

colMeans(output.genr.per.high.LDi)
colMeans(output.upp.per.1.LDi)
colMeans(output.upp.others.LDi)              # results for likelihood distance
colMeans(output.number.of.inf.obs.jab.LDi)
sqrt(var(output.number.of.inf.obs.jab.LDi))
colMeans(output.number.of.inf.obs.tra.LDi)
sqrt(var(output.number.of.inf.obs.tra.LDi))
result.per.of.time.LDi
sqrt(var(result.per.of.time.LDi))
result.per.of.time.tra.LDi
sqrt(var(result.per.of.time.tra.LDi))

########################################################################
```

There is no need to writing new codes for sufficient jackknife-after-bootstrap. Writing sufficient bootstrap codes instead of conventional bootstrap codes is enough. Sufficient bootstrap codes should be as follows.

```
########################## Sufficient bootstrap #########################

re.sample <- runif(n, 1, n)
re.sample <- as.integer(re.sample)
unique.sample <- unique(re.sample)
Sufficient.data <- Design.data[unique.sample,]
dataX <- Sufficient.data[,1:ncol(X)]
dataY <- Sufficient.data[,(ncol(X)+1)]

########################################################################
```