Fall 2012

Finding a Better Confidence Interval for a Single Regression Changepoint Using Different Bootstrap Confidence Interval Procedures

Bodhipaksha Thilakarathne

FINDING A BETTER CONFIDENCE INTERVAL FOR A SINGLE REGRESSION CHANGEPOINT USING DIFFERENT BOOTSTRAP CONFIDENCE INTERVAL PROCEDURES

by

BODHIPAKSHA B.P.THILAKARATHNE

(Under the Direction of Jonathan W. Duggins)

ABSTRACT

Recently a number of papers have been published in the area of regression changepoints but there is not much literature concerning confidence intervals for regression changepoints. The purpose of this paper is to find a better bootstrap confidence interval for a single regression changepoint. ("Better" confidence interval means having a minimum length and coverage probability which is close to a chosen significance level). Several methods will be used to find bootstrap confidence intervals. Among those methods a better confidence interval will be presented.

INDEX WORDS:Changepoint, Regression, Bootstrap,Confidence Interval

FINDING A BETTER CONFIDENCE INTERVAL FOR A SINGLE REGRESSION CHANGEPOINT USING DIFFERENT BOOTSTRAP CONFIDENCE INTERVAL PROCEDURES

by

BODHIPAKSHA B.P.THILAKARATHNE

B.S. in Mathematics

A Thesis Submitted to the Graduate Faculty of Georgia Southern University in Partial Fulfillment

of the Requirement for the Degree

MASTER OF SCIENCE

STATESBORO, GEORGIA

2012

FINDING A BETTER CONFIDENCE INTERVAL FOR A SINGLE REGRESSION CHANGEPOINT USING DIFFERENT BOOTSTRAP CONFIDENCE INTERVAL PROCEDURES

by

BODHIPAKSHA B.P.THILAKARATHNE

Major Professor: Jonathan W. Duggins

Committee: Patricia Humphrey

Arpita Chatterjee

ACKNOWLEDGMENTS

The first person I would like to thank is my direct supervisor Dr. Jonathan Duggins. Especially the helpful comments and many discussions and the interactions with Dr. Jonathan Duggins had a direct impact on the final form and quality of my thesis. A special thanks goes to my co-advisor, Dr. Patricia Humphrey for helping me to complete the writing of this work and her valuable comments. I would like to thank Dr. Arpita Chatterjee for dedicating her time to read my thesis and make valuable suggestions.

I feel a deep sense of gratitude for my father and my mother who formed part of my vision and taught me the good things that really matter in life.

I am very grateful for my loving wife Anuradha, for her great support and encouragement.

TABLE OF CONTENTS

Page

LIST OF TABLES

Table Page

LIST OF FIGURES

CHAPTER 1

LITERATURE REVIEW

1.1 Regression Analysis

Regression analysis is a statistical methodology that utilizes the relationship between two or more quantitative variables so that a response or outcome variable can be predicted from the other or others. This method is widely used in Economics, Management, Social and Behavioral Sciences, Biological Sciences, Political Sciences, Physical and Chemical Sciences and many other disciplines.

There are several ways to categorize regression analysis. Here I will discuss the category which is related my research.

One category of regression analysis is based on the number of explanatory variables in the analysis. If there is a single explanatory variable (also called a predictor variable) the analysis is called simple regression. If there are several explanatory variables it is called multiple regression.

My research is based on the simple linear regression model, a technique for analyzing bivariate data which can help us to understand the linear association between the two variables, to see how a change in the predictor variable is associated with a change in the response variable, and to estimate or predict the value of one of the variables knowing the value of the other variable.

A simple linear regression model is given by the equation

$$y_i = \beta_o + \beta_1 x_i + \epsilon_i \tag{1.1}$$

where x_i is the value of the predictor variable and Y_i is the value of the response variable in the i^{th} trial. ϵ_i is a random error. The simple linear regression model relies on a number of assumptions being satisfied in order for it to provide a reliable

approximation to a linear association between two variables. These assumptions describe the probability distributions of the random errors in the model. According to Iain Pardoe [4] there are four assumptions about these random errors, ϵ:

1. The probability distribution of ϵ at each x-value has a **mean of zero** (in other words, the data points in a scatter plot balance along both sides of the regression line so that the random errors average out to zero as we move across the plot from right to left)$(E\{\epsilon_i\} = 0)$

2. The probability distribution of ϵ at each x-value has **constant variance**, called *homoscedasticity* (in other words, the data points in a scatter plot spread out evenly around the regression line so that the variation of the random errors is similar as we move across the plot from left to right) (var $\{\epsilon_i\} = \sigma^2$),

3. The probability distribution of ϵ at each x-value is **normal** (in other words, the data points in a scatter plot are more likely to be closer to the regression line than farther away and have a gradually decreasing chance of being farther away).

4. The value of ϵ for one observation is **independent** of the value of ϵ for any other observation.

By using the above assumptions,

$E(y_i) = E(\beta_o + \beta_1 x_i + \epsilon_i) = \beta_o + \beta_1 x_i$

$Var(y_i) = Var(\beta_o + \beta_1 x_i + \epsilon_i) = Var(\epsilon_i) = \sigma^2.$

Therefore,

$$y_i \sim N(\beta_o + \beta_1 x_i, \sigma^2) \tag{1.2}$$

1.2 Least-square estimation of the parameter

Our simple linear regression model is

$$y_i = \beta_o + \beta_1 x_i + \epsilon_i.$$

The parameters β_o and β_1 are unknown and must be estimated using sample data, in this case n pairs of (x,y) values. If we can estimate a "best fit" regression line going through our sample (x,y) values, then we can use probability theory results to make inferences about the corresponding regression line for the population.

To estimate the parameters we use least-squares estimation. The best fit in the least-squares sense minimizes the sum of squared residuals (SSR):

$$SSR = \sum_{i=1}^{n} e_i^2 = \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 = \sum_{i=1}^{n} (y_i - \hat{\beta}_o - \hat{\beta}_1 x_i)^2 \tag{1.3}$$

a residual being the difference between an observed value and the fitted value provided by a model ($e_i = y_i - \hat{y}_i$). According to Iain Pardoe [4] the following values of $\hat{\beta}_o$ and $\hat{\beta}_1$ make SSR minimum.

$$\hat{\beta}_1 = \frac{\sum (y_i - \bar{y})(x_i - \bar{x})}{\sum (x_i - \bar{x})^2},\ \hat{\beta}_o = \bar{y} - \hat{\beta}_1 \bar{x}$$

Using this $\hat{\beta}_o$ and $\hat{\beta}_1$ we find the best-fitting line.

1.3 Changepoint Analysis

According to Dr. Duggins a changepoint is any point that marks off homogeneous subsets of data from a heterogeneous whole. If data is no longer considered to be homogeneous then a changepoint may have occurred in the data. Identifying these changes at unknown times and estimating the location of changepoints is referred to as changepoint analysis. Many applications of changepoint analysis are available in different fields, such as quality control, statistical control theory, estimation of the

current position of a time series, testing and estimation of change in the patterns of a regression model, comparison and matching of DNA sequences in microarray data analysis, and environmental threshold models. The changepoint problem also occurs frequently in medical research. For example, cancer incidence rates remain relatively stable for people at a younger age, but change drastically after a certain age. Another example arises from a study of the risk of heart attacks, which showed a sharp decrease in risk at low alcohol intakes and a dramatic increase after reaching a certain amount of daily alcohol consumption. According to Khodadadi and Asgharian [5] numerous methodologies have been implemented to examine changepoint models such as maximum likelihood estimation, piecewise regression, nonparametric regression, and grid searching.

The area of the changepoint analysis has been the subject of intensive research in the past half century. Changepoint detection methods can be classified into two categories: real-time detection (Adams and Mackay) [6] and retrospective detection (Basseville and Nikiforov) [1]. Real-time change-point detection targets applications that require immediate responses such as robot control. On the other hand, according to Basseville and Nikiforov [1] retrospective change-point detection requires longer reaction periods. Retrospective change-point detection accommodates various applications that allow certain delays, for example: climate change detection, genetic time-series analysis, signal segmentation, and intrusion detection in computer networks.

Did a change occur? Did more than one change occur? When did the changes occur? With what confidence did the changes occur? According to Taylor [2] all these questions and more can be answered by performing a changepoint analysis. For each change it provides detailed information including a confidence level indicating the likelihood that a change occurred and a confidence interval indicating when the

change occurred. The changepoint analysis procedure provided is extremely flexible. It can be performed on all types of time ordered data including attribute data, data from non-normal distributions, ill-behaved data such as particle counts and complaint data, as well as data with outliers.

Taylor [2] introduced new software to detect changes by using CUSUM charts and a bootstrap method. Taylor has proved that his method is more powerful than a control chart when detecting smaller sustained changes. The major difference between retrospective changepoint analysis and control charting is that control charts can be updated following the collection of each data point while a retrospective changepoint analysis can only be performed once all the data is collected. Control charts are generally better at detecting major change quickly (real-time detection) while retrospective changepoint analysis can detect changes missed by control charts (retrospective detection). Lai [3] gives a review of problems in sequential analysis, including a discussion about sequential changepoint detection in quality control. Bhattacharya [7] gives an overview of changepoint analysis as it had developed into the mid-1990s. He presents the asymptotic properties of the changepoint and regression coefficient estimators using a local log-likelihood process approach. Through this approach he shows the distinctive features of the asymptotic properties of the changepoint with and without the continuity constraint at the point of change.

1.4 Regression and changepoint analysis

Choosing an appropriate regression equation is the most important feature of the regression analysis. In many applications a smooth regression equation cannot describe the relationship between response and predictor variables and one has to fit different models in different sub-regions. The points at which the regression equation are

not smooth, often representing a change in the pattern of data, are called regression changepoints. More precisely, regression changepoint analysis is a regression problem in which the expected value of the response is assumed to have a different functional form in several neighborhoods of the explanatory variable space. In general, changepoint models can be divided into two groups: models with a discontinuous change at the changepoint (Figure 1.1) and models with a continuous change at the changepoint (Figure 1.2). Discontinuous changepoint models are models with no continuity constraints at the changepoints. In my research the regression function is assumed to

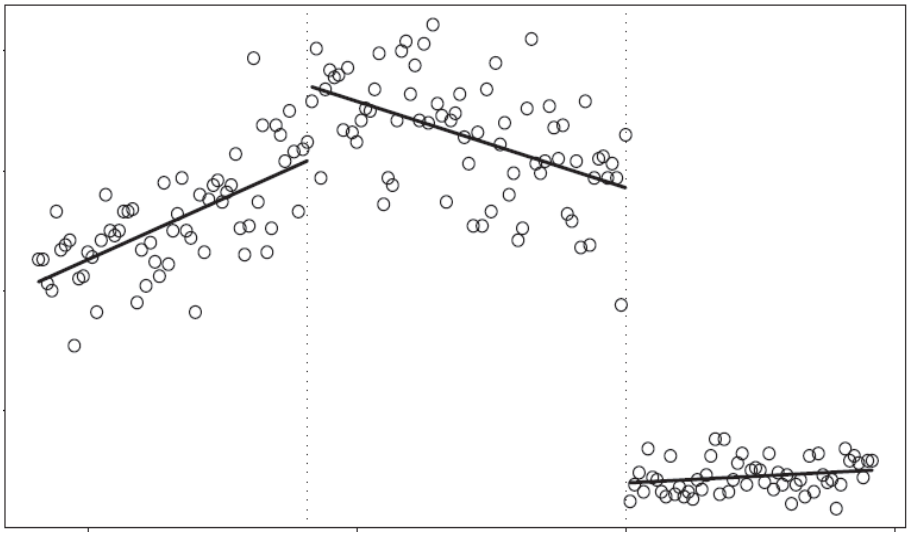

Figure 1.1: Changepoint regression model with two discontinuous changepoints

be continuous at the point of change and we only consider a simple linear regression model with one changepoint. The model of a linear regression with one continuous changepoint can then be stated as

$$Y_i = \begin{cases} \alpha_1 + \beta_1 x_i + \epsilon_i & x_i \leq \tau \\ \alpha_2 + \beta_2 x_i + \epsilon_i & x_i > \tau \end{cases} \tag{1.4}$$

with continuity constraint

$$\alpha_1 + \beta_1\tau = \alpha_2 + \beta_2\tau. \tag{1.5}$$

The changepoint is denoted by τ.

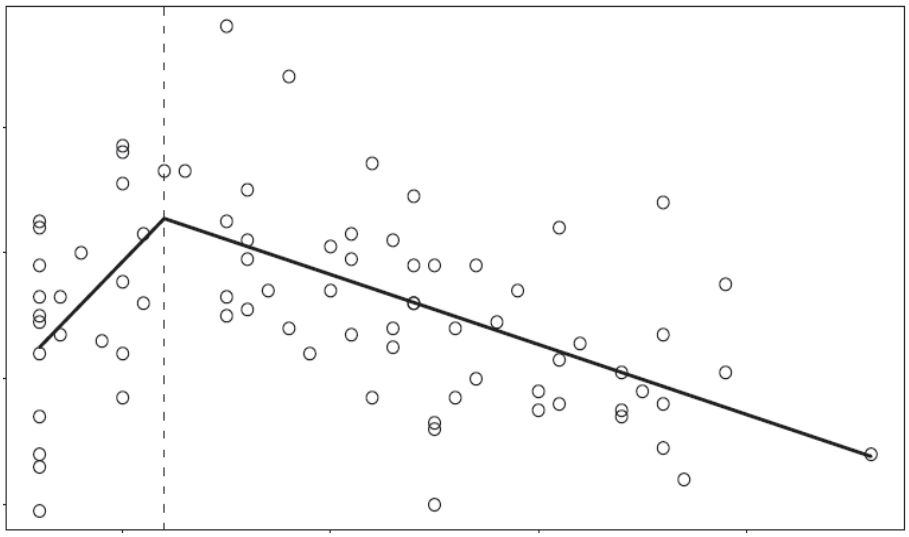

Figure 1.2: Continuous regression changepoint model

The standard changepoint problem in regression models consists of testing the null hypothesis that no change in regimes has taken place against the alternative that observations were generated by two (or possibly more) distinct regression equations.

There has been a surge in research over the past several decades on locating and making inferences about the changepoint, as well as the pattern of the data before and after the changepoint. The problem of a changepoint in the coefficients of a linear regression model has also been analyzed under the assumption of normality by several authors. Sprent [8] was among the first to discuss the estimation of piecewise linear models. His interest in this type of model is based on the observation that a biologist would often postulate a two-phase linear model over some alternatives such as the quadratic model largely on intuitive grounds. Feder [10] studied the model in

a more general framework and proved the consistency of the least-squares estimators of the regression coefficients and the changepoint. The estimators are asymptotically normal for some special cases including models with all linear segments. Hinkley [11] considered the same two-phase straight line model and derived the maximum likelihood estimator (MLE) of the changepoint by its marginal likelihood function and presented the asymptotic distribution of the estimator. Quandt [12] discussed estimates and hypothesis tests for a regression model in two phases. Brown, Durbin and Evans [13] used recursive residuals to detect a single changepoint in regression models. Judith and Lesperance [14] used piecewise regression as a statistical technique to model ecological thresholds. Two statistical methods are proposed by Qian, King and Richardson [15] for the detection of environmental thresholds.

1.5 The Bootstrap

1.5.1 Introduction

Bootstrapping is a computer-based technique that can be used to infer the sampling distribution of many statistics via repeated samples drawn from the sample itself. It is a recently developed technique, introduced by Efron (1979). According to Efron and Tibshirani [19], when using a bootstrap, the basic ideas of statistics haven't changed, but their implementation has. A great advantage of the bootstrap is its simplicity.

When making inferences, traditional parametric procedures are primarily based on several major assumptions about the population(s) from which our data come. For example, one may assume that the distribution of population values is truly normal, with unknown mean and variance, and that data sets are generated by simple random sampling. Applying these assumptions makes inference easier because we can easily draw conclusions about the underlying population.

Most of the time all these assumptions are violated. In the first place, it is not hard to create reasonable data that violate a normality assumption and have "true" answers that are quite different from the answer we would get by making a normality assumption. Second, there are many situations where even with normality we don't know enough about the statistic we are using to draw the appropriate inferences. For example, one of the first things students learn in statistics is that the estimated standard error of a mean $\bar{x}$ based on n independent data points $x_1, x_2, x_3...x_n$ is given by the formula

$$\sqrt{\frac{s^2}{n}} \tag{1.6}$$

where $s^2 = \sum_{i=1}^{n} \frac{(x_i - \bar{x})^2}{n-1}$,but what is the standard error of the median, or the standard error of the difference between medians? We need some other way to find that standard error. The bootstrap was introduced in 1979 as a computer-based method for estimating the standard error for any statistic.

The bootstrap algorithm from Efron and Tibshirani [19] for estimating standard error is given as follows. B bootstrap samples are generated from the original data points $x_1, x_2, ..., x_n$. Each bootstrap sample, $x^* = (x_1^*, x_2^*, ...x_n^*)$, has n elements generated by sampling with replacement n times from the original data set. Corresponding to each bootstrap sample there is a bootstrap replication of s, namely $s(x^*)$, the value of the statistic s evaluated for x^*. For instance, if $s(x)$ is the sample median, then $s(x^*)$ is the median of the bootstrap sample. Finally the standard deviation of the bootstrap replications $(s(x^{*1}), s(x^{*2}), ...s(x^{*B}))$ is our estimate of the standard error of $s(x)$.

$$\hat{se}_{boot} = \left\{ \sum_{b=1}^{B} [s(x^{*b}) - s(\cdot)]^2/(B-1) \right\}^{1/2} \tag{1.7}$$

where $s(\cdot) = \sum_{b=1}^{B} s(x^{*b})/B$ and $s(x^{*b})$ is the value of the statistic s evaluated for $x^{*b}th$ bootstrap sample.

According to Efron and Tibshirani, [19] Figure 1.3 clearly displays the bootstrap process for estimating the standard error of a statistic $s(x)$.

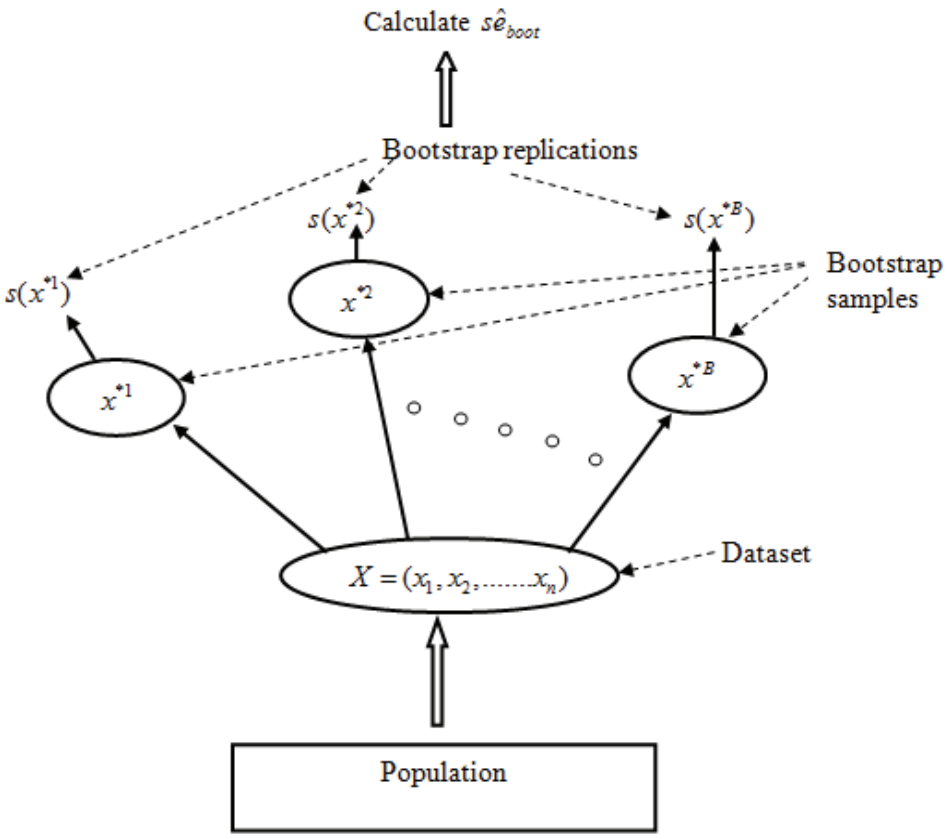

Figure 1.3: Bootstrap process for estimating the standard error of a statistic $s\,(x)$

The Empirical Distribution Function

The empirical distribution function (cdf), which we will call $\hat{F}$, is a simple estimate of the population distribution F. According to Efron and Tibshirani [19], the empirical distribution function $\hat{F}$ is defined to be the discrete cumulative distribution function (cdf) that puts probability $\frac{1}{n}$ on each value x_i, $i = 1, 2, ...n$ when there are no ties. If there are ties $\hat{F}$ is defined to be the discrete cumulative distribution function that puts

probability $\frac{f(i)}{n}$ on each value x_i, $i = 1, 2, ...n$ where $f(i)$ is the frequency of x_i. Many statistical procedures depend on the performance of the cdf. In particular, bootstrap methods rely heavily on the empirical distribution function. Let $(x_1, x_2, ..., x_n)$ be independent and identically distributed real-valued random variables with common cdf $F(t)$. Then the empirical distribution function is defined as

$$\hat{F}_n(t) = \frac{1}{n} \sum_{i=1}^{n} I \{x_i \leq t\} \tag{1.8}$$

Where $I = 1$ if $x_i \leq t$ and 0 otherwise.

1.5.2 Bootstrap Confidence Interval

In statistics a confidence interval is a measurement of how good, or how accurate, an estimate of a certain parameter is. Confidence intervals consist of a range of values that act as good estimates of the unknown population parameter. The reason that we need confidence intervals is that a point estimate, being a single value, cannot express the statistical variation, or random error, that the estimate has. A definition of a confidence interval is a range of values constructed from sample data so that the population parameter is likely to occur within that range at a specified probability. The specified probability before sampling is called the level of confidence.

Standard errors are often used to assign approximate confidence intervals to a parameter θ of interest. Given an estimate $\hat{\theta}$ and an estimated standard error $\hat{se}$, the $100(1-\alpha)\%$ confidence interval for θ is often given as

$$\hat{\theta} \pm critical\ value * \hat{se} \tag{1.9}$$

In bootstrapping, the bootstrap distribution of a parameter estimate has been used to calculate bootstrap confidence intervals for its population parameter. There is no advantage to calculate normal-theory bootstrap confidence intervals for statistics like

the mean, because in this case the ideal bootstrap standard deviation of the statistic and the standard error based directly on the sample coincide. If we don't know what the sampling distribution of our estimate is then we can compute a confidence interval nonparametrically. Here I will discuss some bootstrap confidence interval methods that I will use in my research.

Percentile Method

The percentile method is the most obvious way to construct a confidence interval for a bootstrap estimate. This method is based on percentiles of the bootstrap distribution of the statistic. Suppose that $\hat{\theta}_i^*$ is the bootstrap estimate from the i^{th} bootstrap sample where each bootstrap sample is of size n. If we ordered the bootstrap estimates from smallest to largest the central interval containing 90% of the $\hat{\theta}_i^*$ values would be a 90% percentile confidence interval for θ. A bootstrap confidence interval generated this way is called a percentile method confidence interval.

According to Efron and Tibshirani [19], let $\hat{G}$ be the cumulative distribution function of $\hat{\theta}_i^*$. The $1-2\alpha$ percentile interval defined by the α and $1-\alpha$ percentiles of $\hat{G}$ is given by

$$[\hat{\theta}_{\%,lo}, \hat{\theta}_{\%,up}] = [\hat{G}^{-1}(\alpha), \hat{G}^{-1}(1-\alpha)]. \tag{1.10}$$

This is a percentile interval when number of bootstrap replications is infinite. In practice we use some finite number of bootstrap replications to approximate the percentile method. If the number of bootstrap replications B is large, then the $100\alpha^{th}$ or lower limit of the empirical percentile interval is the $(B \cdot \alpha)^{th}$ value in the ordered list of the B replications. Likewise the upper limit is given by $(B(1-\alpha))^{th}$ value in the ordered list of the B replications. If the value of $(B \cdot \alpha)$ is not an integer then

we consider the floor value and if $(B(1-\alpha))$ is not an integer then we consider the celling value. We must choose a finite number of bootstrap samples, therefore it is an approximation to the percentile interval. To increase the accuracy we need to choose a wider interval if those values are not an integer. So that is why we choose floor and ceiling values.

According to Efron and Tibshirani [19] if the bootstrap distribution of $\hat{\theta}^*$ is approximately normal, then the standard normal and percentile intervals will nearly agree. In general the central limit theorem tells us that as $n \to \infty$ where n is the sample size, the bootstrap histogram (histogram of B bootstrap replications) will become normal shaped, but for small samples it may look very non-normal. According to Efron and Tibshirani [19] the percentile interval is better than standard normal interval for small samples.

The Jackknife

The jackknife is a technique for estimating the bias and standard error of an estimate. It is done by deleting one data point each time from the original data set and recalculating the estimator based on the rest of the data.

According to Efron and Tibshirani [19] the jackknife estimate of bias is defined by

$$\hat{bias}_{jack} = (n-1)(\hat{\theta}_{(\cdot)} - \hat{\theta}) \tag{1.11}$$

where $\hat{\theta}_{(\cdot)} = \sum_{i=1}^{n} \hat{\theta}_i/n$ and $\hat{\theta}_i$ is the ith jackknife replication of estimator.The jackknife estimate of standard error is defined by

$$\hat{se}_{jack} = [(n-1)/n \sum (\hat{\theta}_i - \hat{\theta}_{(\cdot)})^2]^{1/2} \tag{1.12}$$

The jackknife often provides a simple and good approximation to the bootstrap for estimation of standard errors and bias. However, according to Efron and Tibshi-

rani [19], the jackknife can fail miserably if the statistic $\hat{\theta}$ is not "smooth." Intuitively, the idea of smoothness is that small changes in the data set cause only small changes in the statistic. A simple example of a non-smooth statistic is the median. To see why the median is not smooth, consider following nine ordered values.

$$10, 27, 31, 40, 46, 50, 52, 104, 146$$

The median of these values is 46. Now suppose we start increasing the value of the 4^{th} largest value $x_{(4)} = 40$. The median does not change at all until $x_{(4)}$ becomes larger than 46, and then after that the median equal to $x_{(4)}$, until $x_{(4)}$ exceeds 50. This implies that the median is not a differentiable (or smooth) function of x.

1.6 Better bootstrap confidence intervals

One of the principal goals of bootstrap theory is to produce good confidence intervals automatically. According to Efron and Tibshirani [19] "good" means that the bootstrap intervals should closely match exact confidence intervals in those special situations where statistical theory yields an exact answer and should give dependably accurate coverage probabilities in all situations.

Bias-corrected method

Efron and Tibshirani [19] described a bias-corrected method for a constructing approximate confidence intervals for a parameter θ. This method is an adjustment to percentile intervals that improves their accuracy. According to them a bias corrected interval is given by

$$(\hat{\theta}_{lo}, \hat{\theta}_{up}) = (\hat{\theta}^{*(\alpha_1)}, \hat{\theta}^{*(\alpha_2)}) \tag{1.13}$$

where $\hat{\theta}^{*(\alpha_1)}$ indicates the $100\alpha_1^{th}$ percentile of B bootstrap replications and $\hat{\theta}^{*(\alpha_2)}$ indicates the $100\alpha_2^{th}$ percentile of B bootstrap replications, and $\alpha_1 = \phi\left(2\hat{z}_0 + z^{(\alpha)}\right)$

and $\alpha_2 = \phi\left(2\hat{z}_0 + z^{(1-\alpha)}\right)$.

Here $\phi(\cdot)$ is the standard normal cumulative distribution function and $z^{(\alpha)}$ is the $100\alpha^{th}$ percentile point of a standard normal distribution. $\hat{z}_0$ is called the bias correction. The value of the bias correction $\hat{z}_0$ is obtained directly from the proportion of bootstrap replications less than the original estimate, $\hat{\theta}$.

$$\hat{z}_0 = \phi^{-1}\left(\frac{\#\left\{\hat{\theta}^*(b) < \hat{\theta}\right\}}{B}\right)$$

$\phi^{-1}(\cdot)$ is the inverse function of a standard normal cumulative distribution function. Roughly speaking, $\hat{z}_0$ measures the median bias of $\hat{\theta}^*$, that is the discrepancy between the median of $\hat{\theta}^*$ and $\hat{\theta}$. If the bootstrap sampling distribution is symmetric, and if $\hat{\theta}$ is unbiased, then this proportion will be close to .5, and the correction factor will be close to zero.

Acceleration method

This is another way to improve the accuracy of percentile intervals. According to Efron and Tibshirani [19] the acceleration interval is given by,

$$(\hat{\theta}_{lo}, \hat{\theta}_{up}) = (\hat{\theta}^{*(\alpha_1)}, \hat{\theta}^{*(\alpha_2)})$$

where

$$\alpha_1 = \phi\left(\frac{z^{\alpha}}{1 - \hat{a} * z^{(\alpha)}}\right) \; \alpha_2 = \phi\left(\frac{z^{\alpha}}{1 - \hat{a} * z^{(1-\alpha)}}\right)$$

$\phi(\cdot)$ is the standard normal cumulative distribution function and $z^{(\alpha)}$ is the 100αth percentile point of a standard normal distribution. $\hat{a}$ is called the acceleration constant. To compute the acceleration constant Efron and Tibshirani used jackknife values of a statistic. According to Efron and Tibshirani [19] a simple expression for the acceleration is given as

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

where

$$\hat{\theta}_{(\cdot)}=\textstyle\sum_{i=1}^{n}\hat{\theta}_{(i)}/n$$

is called the average jackknife estimate and $\hat{\theta}_{(i)}$ is called the jackknife estimate with the i^{th} point deleted. The quantity $\hat{a}$ is called the acceleration because it measures the skewness of original estimate.

Bias corrected and Accelerated method (BC_a)

In this method Efron and Tibshirani used both acceleration and bias correction for defining α_1 and α_2. According to Efron and Tibshirani [19] the BC_a interval of coverage $1-2\alpha$ is given by

$$(\hat{\theta}_{lo},\hat{\theta}_{up})=(\hat{\theta}^{*(\alpha_1)},\hat{\theta}^{*(\alpha_2)})$$

where

$$\alpha_1=\phi\left(\hat{z}_0+\frac{\hat{z}_0+z^{(\alpha)}}{1-\hat{a}(\hat{z}_0+z^{(\alpha)})}\right)$$

$$\alpha_2=\phi\left(\hat{z}_0+\frac{\hat{z}_0+z^{(1-\alpha)}}{1-\hat{a}(\hat{z}_0+z^{(1-\alpha)}}\right)$$

If $\hat{a}$ and $\hat{z}_0$ equal zero , then $\alpha_1=\phi\left(z^{(\alpha)}\right)=\alpha$ and $\alpha_2=\phi\left(z^{(1-\alpha)}\right)=1-\alpha$ so the BC_a interval is the same as the percentile interval.

1.7 Bootstrap Regression

Since my research is based on simple linear regression, I only pay attention to bootstrap analysis for a simple linear regression model in this section. In Section 1.1

and Section 1.2, I discussed the least-square estimation and four assumptions related to the residuals. Under the assumptions I discussed in Section 1.1, the least-square procedure provides the best linear unbiased estimates of the regression parameters. However, if these assumptions are violated then inferential prediction done by least-square estimation may not be suitable. This where the bootstrap can help us.

The basic idea behind bootstrapping regression is to construct bootstrap standard errors and confidence intervals for the regression coefficients. Efron and Tibshirani [19] discussed bootstrapping regression in detail in their book. Here is the procedure to construct bootstrap standard errors and confidence intervals for the regression coefficients for a simple linear regression model. This procedure is called bootstrapping the residuals.

1. Estimate the regression coefficients ($\hat{\beta}_o$, $\hat{\beta}_1$) for the original sample, and calculate the fitted value and residual for each observation. $\hat{Y}_i = \hat{\beta}_o + \hat{\beta}_1 x_i$ and $\epsilon_i = Y_i - \hat{Y}_i$, $i = 1, 2, 3, ...n$

2. Keeping these residuals as the original sample, generate B bootstrap samples. Then calculate bootstrap $Y^*_{b,i}$ values for each observation in the bootstrap sample. ($Y^*_{b,i} = \hat{Y}_i + \epsilon_{b,i}$) where $b = 1, 2, 3, ..., B$

3. Regress the bootstrapped $Y^*_{b,i}$ values on the fixed x values to obtain estimated bootstrap regression coefficients. Estimates are calculated by least-square regression.These estimators can be used to construct the bootstrap standard error and confidence intervals for the regression coefficients.

This is one way to do bootstrapping regression. There is another way called bootstrap by pairs. In this method, bootstrap (x, y) pairs similarly to individual observation and generate B bootstrap samples. Since we have bootstrap (x, y) pairs,

we can find estimated regression coefficients for each bootstrap sample by using least-square method.

My research is based on bootstrapping residuals. I am finding a better bootstrap confidence interval for a regression changepoint based on a previously developed test statistic. So only the bootstrapping residuals method guaranteed my underlying structure is there. I will explain why the bootstrap pairs method does not work by an example. Suppose a bootstrap sample (x^*) is obtained by randomly sampling n times, with replacement, from the original pairs $(x_1, y_1), (x_2, y_2), ..., (x_n, y_n)$. For instance, with $n = 5$ we might obtain $x^* = (x_2, y_2), (x_2, y_2), (x_2, y_2), (x_2, y_2), (x_2, y_2)$. So in this case this bootstrap sample has one original pair (x_2, y_2) which is repeated 5 times. If this happened then it didn't give us our underlying structure because it has only one data point, but if we used bootstrapping residuals method even though it has repeated residuals for each bootstrap sample we can generate bootstrap $Y^*_{b,i}$ values on the fixed x values to obtain estimated bootstrap regression coefficients.

The most important thing in bootstrapping regression is not depending on error assumptions, which are discussed in Section 1.1. Normally we construct standard errors and confidence intervals for the regression coefficients by using these assumptions. In many scenarios one or more of these assumptions may be violated. For example, in some cases such as the error distribution is not particularly close to a normal distribution. Therefore bootstrapping regression is a very good method to construct standard errors and confidence intervals for the regression coefficients.

CHAPTER 2

METHODS

2.1 Narrowing the Field

Recently a number of papers have been published in the area of bootstrapping regression changepoint analysis. Most of these papers are concerned with estimating or detecting a changepoint in regression model. In 2000 Julious [20] proposed a test procedure to test whether the two-line model has a statistically better fit compared to a one line model when the changepoint is known. This leads to the following test statistic.

$$F = \frac{RSS_1 - RSS_2}{RSS_2/(n-3)} \tag{2.1}$$

Here the RSS_1 and RSS_2 are the sum of squared residuals for the one- and two-line models respectively and n is the number of observations. The statistic has an F distribution with 1 and $n-3$ degrees of freedom. When the location of changepoint is not known then this method does't work. According to Julious [20] test statistic for unknown changepoint is given by the following equation.

$$F = \frac{(RSS_1 - RSS_2)/2}{RSS_2/(n-4)} \tag{2.2}$$

This test statistic no longer has an exact F distribution. Therefore the only way to estimate a parameter is through numerical optimization. In my research an unknown changepoint was estimated by using following method.

1. Fit overall one line model for points $x_1, x_2....., x_k, ...x_n$
2. Compute RSS_1
3. Fit a two line model for points $x_1, x_2....., x_k$ and $x_{k+1}, ...x_n$
4. Compute RSS_2

5. Compute all possible two line models

6. Compute F for each two line model

7. The two line model with largest F gives the best changepoint model.

To test whether the two line model has a statistically better fit compared to a one line model we need to know the distribution of our test statistic. According to the Julious paper [20] Efron and Gong proposed nonparametric bootstrap method to estimate distribution of test statistic (bootstrap distribution for the F-test).

The methodology in applying bootstrap methods to the changepoint problem is given by

1. For a given set of data obtain the best fitting two-line and one-line models and calculate the F statistic.

2. Calculate the residuals for the two-line case.

3. Using the original $x$values, recalculate the new y values, by using the values from the best fitting one line model and adding an error term, sampled with replacement from the set of residuals from the best fitting two-line model.

4. To this new set of data, fit a two line and one line model and calculate Fstatistic.

5. Repeat steps 3 and 4 a large number of times, each time using the one line parameters and two line residuals from the original data.

A bootstrap distribution for the F test can be derived and a P-value can thus be calculated.

Another important issue in changepoint analysis is how to obtain a confidence interval for the changepoint. Huskova and Kirch(2008) [21] considered bootstrap

confidence intervals for the changepoint of the mean in a time series context. Dumbgen(1991) [22] proposed an asymptotically valid confidence interval for the changepoint by inverting bootstrap tests in a one-sample problem.

There is not much literature concerning confidence intervals for the regression changepoint. Seijo and Sen (2011)[23] used bootstrap procedures to construct confidence intervals for the unique jump discontinuity. Toms, Judith and Lesperance [24] have calculated three types of confidence intervals for the regression changepoint estimate: an interval based on the computed standard error of the estimate from the fitting procedure, an empirical bootstrap confidence interval, and a confidence interval derived from an inverted F test.

2.2 Extensions

In this research I focus on obtaining a better confidence interval for a regression changepoint among a set of bootstrap confidence interval methods compared to the standard percentile methods. I used confidence intervals based on bootstrap percentiles which are clearly described in Efron and Tibshirani [19]. The methods I used include the percentile method and its adjustments such as bias-corrected method, acceleration method and bias-corrected and acceleration method. By keeping these four methods as a base I constructed three more different bootstrap confidence intervals.

1. Bias-corrected and acceleration method with average jackknife estimator
2. Bias-corrected and acceleration method with adjustment for ties
3. Bias-corrected and acceleration method with average jackknife estimator and adjustment for ties

Including these three and the four basic methods we have seven bootstrap confidence intervals. Among these seven bootstrap confidence intervals a better confidence interval which gives coverage probability close to 95% and minimum length will be selected. The four main methods were described in Sections 1.6 and 1.7.

Bias-corrected and acceleration method with average jackknife estimator

As I discussed in Section 1.7, Efron and Tibshirani [19] used both acceleration and bias correction for defining α_1 and α_2. According to them, bias correction is defined as

$$\hat{z}_0 = \phi^{-1}\left(\frac{\#\left\{\hat{\theta}^*(b) < \hat{\theta}\right\}}{B}\right)$$

In this case the value of the bias correction $\hat{z}_0$ is obtained directly from the proportion of bootstrap replications less than the original estimate $\hat{\theta}$. In our new method we used the average jackknife estimator ($\hat{\theta}_{(\cdot)} = \sum_{i=1}^{n} \hat{\theta}_{(i)}/n$) instead of the original estimator to calculate bias correction.

The new formula for bias correction is given by

$$\hat{z}_0 = \phi^{-1}\left(\frac{\#\left\{\hat{\theta}^*(b) < \hat{\theta}_{(\cdot)}\right\}}{B}\right).$$

By using this new formula for bias correction we can compute α_1 and α_2

$$\alpha_1 = \phi\left(\hat{z}_0 + \frac{\hat{z}_0 + z^{(\alpha)}}{1 - \hat{a}(\hat{z}_0 + z^{(\alpha)})}\right)$$

$$\alpha_2 = \phi\left(\hat{z}_0 + \frac{\hat{z}_0 + z^{(1-\alpha)}}{1 - \hat{a}(\hat{z}_0 + z^{(1-\alpha)}}\right)$$

Bias-corrected and acceleration method with adjustment for ties

In this method we calculate bias correction and acceleration with an adjustment for ties. If we have ties in our original sample then $P(X < x^*) \neq P(X \leq x^*)$. So we need

an adjustment for ties otherwise we cannot calculate correct acceleration and bias correction for our original sample. Let x^* be a point where ties occurred in original sample. To make the adjustment for ties we take the number of cases where $X = x^*$ and divide them in half. We pool half of them with $X < x^*$ and other half with $X > x^*$.

The new formula for bias correction is given by

$$\hat{z}_0 = \phi^{-1}\left(\frac{\#\left\{\hat{\theta}^*(b) < \hat{\theta}\right\} + \frac{1}{2}\#\{\hat{\theta}^*(b) = \hat{\theta}\}}{B}\right)$$

By using this new formula for bias correction we can compute α_1 and α_2.

Bias-corrected and acceleration method with average jackknife estimator and adjustment for ties

As the name suggests we apply both the average jackknife estimator and adjustment for ties to calculate bias correction and acceleration in this method.

A summary of all bootstrap confidence intervals and notations that I used are given by Table 2.1.

Name	Notation
Bias-corrected method	*BC*
Acceleration method	*a*
Bias-corrected and acceleration method	*BCa*
Percentile method	*Pct*
Bias-corrected and acceleration method with average jackknife estimator	*BCajk*
Bias-corrected and acceleration method with adjustment for ties	*BCaat*
Bias-corrected and acceleration method with average jackknife estimator and adjustment for ties	*BCajkat*

Table 2.1: Summary of all bootstrap confidence intervals and notations that I used

2.3 Design Procedure

Bootstrap confidence intervals for the regression changepoint were constructed by the following procedure.

1. Generate random sample data (sample size is n) with a given changepoint according to

$$Y_i = \begin{cases} \alpha_1 + \beta_1 x_i + \epsilon_i & x_i \leq \tau \\ \alpha_2 + \beta_2 x_i + \epsilon_i & x_i > \tau \end{cases} \tag{2.3}$$

with continuity constraint

$$\alpha_1 + \beta_1 \tau = \alpha_2 + \beta_2 \tau. \tag{2.4}$$

The changepoint is denoted by τ.

To make our data denser, we only consider the x values within the interval [0,10]. Then subdivide this interval into n subintervals with equal length. Then we can say how many data points are less than the changepoint and how many data points are greater than the changepoint. For example let $n = 20$; the length of subinterval is $10/20 = 0.5$. So now we have a data point for each 0.5 x value. Normally distributed residuals are generated with mean 0 and variance 1 and added to the underlying model.

2. Estimate the changepoint ($\hat{\tau}$) from that original sample data, using SAS 9.3 code which was co-developed by Dr. Jonathan Duggins and Dr. James Blum (University of North Carolina Wilmington) and which is included in Appendix A.

3. Select 1000 independent bootstrap samples from that original sample data. Each bootstrap sample consists of n residuals drawn with replacement from original data.

4. Estimate the changepoint in each bootstrap sample. After ordering the bootstrap replicates from smallest to largest then compute the percentile confidence interval.

5. Create jackknife estimates for our original data by deleting one data point each time from the original data set. Then find the average jackknife estimate.

6. By using average jackknife estimate and jackknife estimates for our original data we can compute the acceleration constant and acceleration confidence interval.

7. Compute bias correction. Because of our method, we will have enough data to compute all seven confidence intervals.

So in this case we calculate all confidence intervals for a regression changepoint by generating only one sample data set. The entire process (steps 1-7) were repeated 10,000 times for each scenario to more effectively compare the resulting confidence intervals for estimated coverage probability and overall average length.

2.4 Comparisons of Methods

Simulations were conducted using SAS 9.3 to compare the different confidence interval methods shown in Table 2.1. In all cases without loss of generality the standard deviation of residuals was taken as one and slope of the null model (β_1) was assumed to have a common slope of one. The Y intercept of the first line model was taken as 0 and the Y intercept of the second line model was calculated according to the following equation to have a continuous change point model.

$$Y\text{intercept of second line model} = \tau^*(\beta_1\text{-}\beta_2)$$

where τ is the changepoint and β_1 is the slope of the first line model and β_2 is the slope of the second line model. According to the SAS code we can run different simulations by changing sample size, simulation size, bootstrap samples, β_2, and nominal coverage probability. The resulting simulation designs are shown below in Table 2.2.

According to this simulation design I considered three different sample sizes (n=20, 40, 60). For each sample size three different changepoints (3, 4, 5) were considered. Finally for every changepoint I considered four different β_2 levels (1.5, 2, 2.5 ,3). Therefore for each sample size we can consider 12 different cases. For each case 10,000 simulations were performed and each simulation has 1,000 bootstrap samples.

sample size	changepoint	β_2	effect size $\frac{\lvert\beta_2 - \beta_1\rvert}{\sigma}$
		1.5	0.5
		2	1
		2.5	1.5
	3	3	2
		1.5	0.5
		2	1
		2.5	1.5
	4	3	2
		1.5	0.5
		2	1
		2.5	1.5
20	5	3	2
		1.5	0.5
		2	1
		2.5	1.5
	3	3	2
		1.5	0.5
		2	1
		2.5	1.5
	4	3	2
		1.5	0.5
		2	1
		2.5	1.5
40	5	3	2
		1.5	0.5
		2	1
		2.5	1.5
	3	3	2
		1.5	0.5
		2	1
		2.5	1.5
	4	3	2
		1.5	0.5
		2	1
		2.5	1.5
60	5	3	2

Table 2.2: Simulation designs for sample sizes 20,40, and 60

CHAPTER 3

ANALYSIS

3.1 Simulation Results

In the previous chapter I have explained how simulations were conducted using SAS 9.3 to compare the different confidence interval methods shown in Table 2.1. In this chapter results of those simulations will be discussed.

The main purpose of this research is to choose a better confidence interval for a regression changepoint among a set of bootstrap confidence interval methods discussed in Table 2.1. I will choose a confidence interval method which gives a coverage probability close to 95% and the minimum length as a better confidence interval for the regression changepoint among seven confidence interval methods.

The estimated coverage probability was calculated over all simulations for each bootstrap confidence interval method. Then it was used to compare estimated coverage probabilities in all seven bootstrap confidence intervals for different β_2 levels to choose a better one. By setting the nominal coverage probability as 95% for our simulated data, I checked the estimated coverage probabilities of each confidence interval methods for different β_2 values.

The mean length was calculated over all simulations for each bootstrap confidence interval method. Then it was used to compare all other mean lengths in all seven bootstrap confidence intervals for different β_2 values to choose a better one. It is not clear to compare all seven bootstrap confidence intervals in one graph. Therefore, I split them into two categories to find a better bootstrap confidence interval for a regression changepoint. In the first category *BCa, a, BC* and *Pct* intervals are considered.

To make the results section more readable, I only present the graphs for sample

sizes (n) 20, 60 and changepoints (τ) 3, 5. The graphs of $n = 40$ and three other changepoints will not be discussed here because their estimated coverage probabilities and mean lengths are in between $n = 20$ and 60. Also, graphs of $n = 40$ follow the same pattern as graphs of $n = 60$. I am not going to discuss the graphs of changepoint 4 for different sample sizes here because their estimated coverage probabilities and mean lengths are in between changepoint 3 and 5 for different sample sizes. Graphs related to sample size 40 and changepoint 4 are in Appendix B.

Figure 3.1 displays estimated coverage probabilities of *BCa, a, Bc,* and *Pct* intervals for $n = 20$ and $\tau = 3$ for different β_2 (*Slope2*) values. According to this figure *Pct* and *a* intervals show good coverage probabilities close to 95%. Their coverage probabilities vary from 98% to 93% when the slope increases from 1.5 to 3. *BCa* and *BC* coverage probabilities are increased when β_2 increases. The estimated coverage probabilities of *BCa* and *BC* are considerably less than *Pct* and *a* estimated coverage probabilities, but *BC* interval always has a higher coverage probability compare to *BCa* .

The mean lengths related to *BCa, a, Bc,* and *Pct* intervals for $n = 20$ and $\tau = 3$ are shown in Figure 3.2. The mean length of all four methods decreases when β_2 increases. Comparing all four bootstrap confidence intervals, the mean length of *BCa* shows the lowest values when β_2 changes from 1.5 to 3. Its mean length decreases from 3.4 to 2.1 but mean length of *BC* interval is very close to *BCa* interval for large β_2 values.

Figure 3.3 displays estimated coverage probabilities of *BCa, a, Bc,* and *Pct* intervals for $n = 60$ and $\tau = 3$ for different β_2 values. When sample size increases, both *BCa* and *BC* intervals show an increase in their estimated coverage probabilities. *Pct* confidence interval shows the highest coverage probability compared to the other three intervals when β_2 changes from 1.5 to 3, but *BC* estimated coverage probability

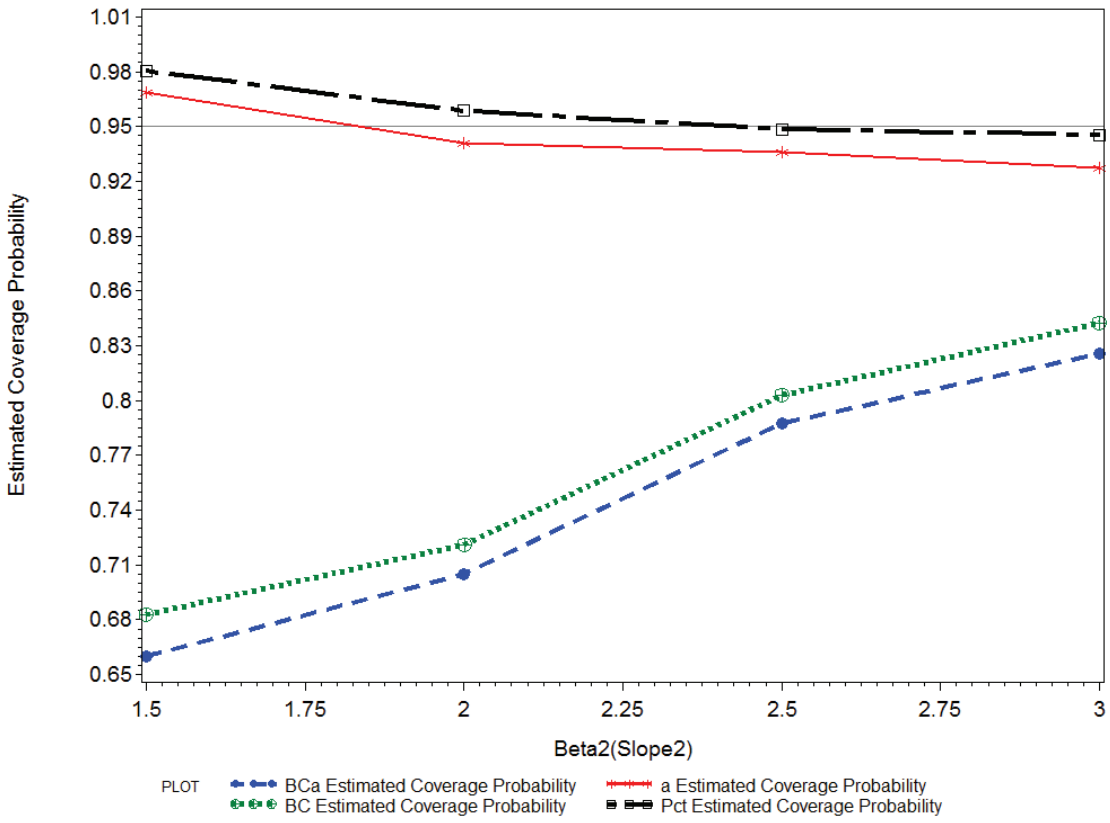

Figure 3.1: Estimated coverage probabilities of BCa,a,BC,and Pct for n=20 and $\tau = 3$

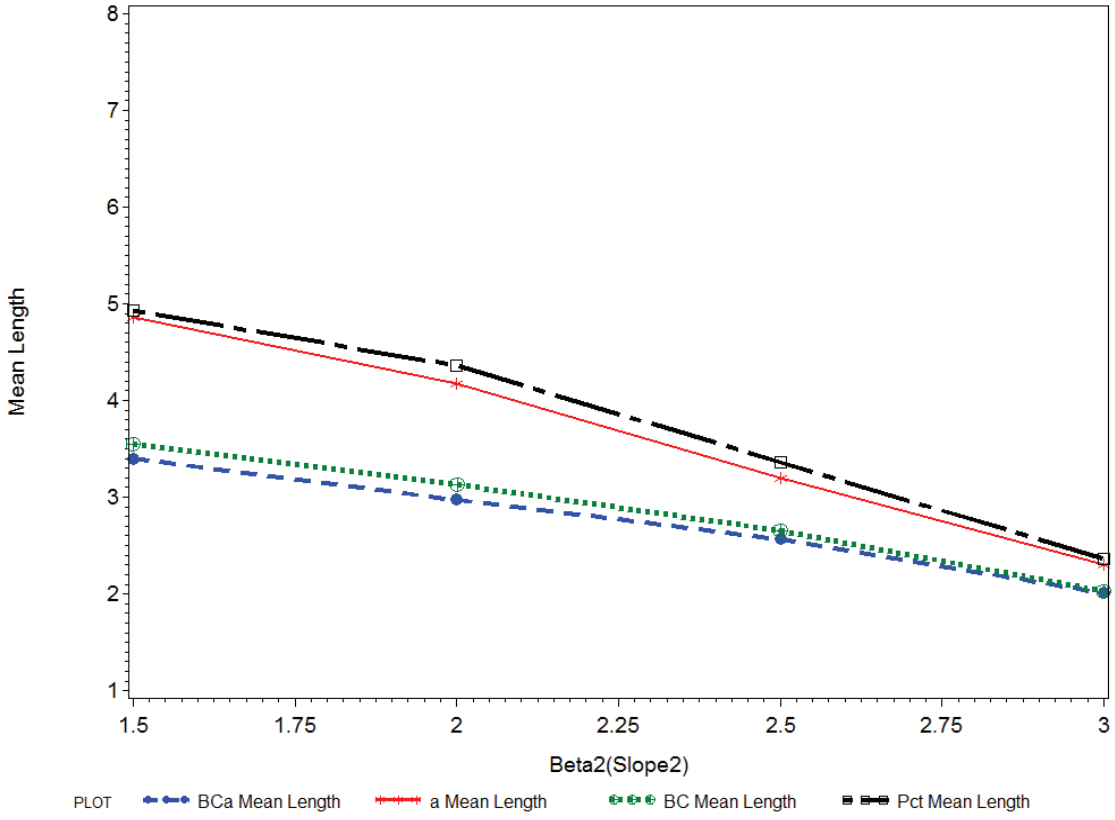

Figure 3.2: Mean length of BCa,a,BC,and Pct for n=20 and $\tau = 3$

is very close to *Pct* estimated coverage probability when $\beta_2 = 2.5$ and 3.

Figure 3.4 shows the mean length of *BCa, a, Bc,* and *Pct* for $n = 60$ and $\tau = 3$ for different β_2 values. The graph indicates that mean length of each interval declines when β_2 increases. When β_2 is 1.5 *Pct* interval shows the highest mean length, but when β_2 increases it is close to *BC* mean length. For higher β_2 values *Pct* and *BC* intervals have minimum mean lengths compared to the other two.

The results of comparison in estimated coverage probabilities of *BCa, a, Bc,* and *Pct* intervals for $n = 20$ and $\tau = 5$ are presented in Figure 3.5. Initially, the graph of *BC* interval has estimated coverage probability close to 82% and *Pct* interval has estimated coverage probability close to 99%. When $\beta_2 = 3$, the estimated coverage probabilities of both *Pct* and *BC* intervals are approximately 90%. On the other hand, the estimated coverage probability of *a* interval is close to 97% and *BCa* interval is close to 79%. When $\beta_2 = 3$, both estimated coverage probabilities are close to 87%.

According to Figure 3.6, the mean length of *BCa, a, Bc,* and *Pct* ($n = 20$ and $\tau = 5$) intervals decline when β_2 increases. At the beginning the mean length of *BC* and *BCa* intervals show low values compared to the other two intervals, but at the end *BC* and *Pct* intervals show low mean lengths compared to the other two intervals.

Figure 3.7 indicates estimated coverage probabilities of *BCa, a, Bc,* and *Pct* intervals for $n = 60$ and $\tau = 5$. When $\beta_2 = 1.5$ the estimated coverage probabilities of *a* interval is close to 95% and *Pct* interval is close to 96%. Both of them decline when β_2 changes from 1.5 to 2 and remain constant after that. Both *BCa* and *BC* intervals show a small increase from $\beta_2 = 1.5$ to $\beta_2 = 2$ and remain constant after that. The estimated coverage probability of *BC* interval is close to 93% at the beginning and it is close 94% at the end.

Figure 3.8 shows the mean length of *BCa, a, Bc,* and *Pct* intervals for $n = 60$

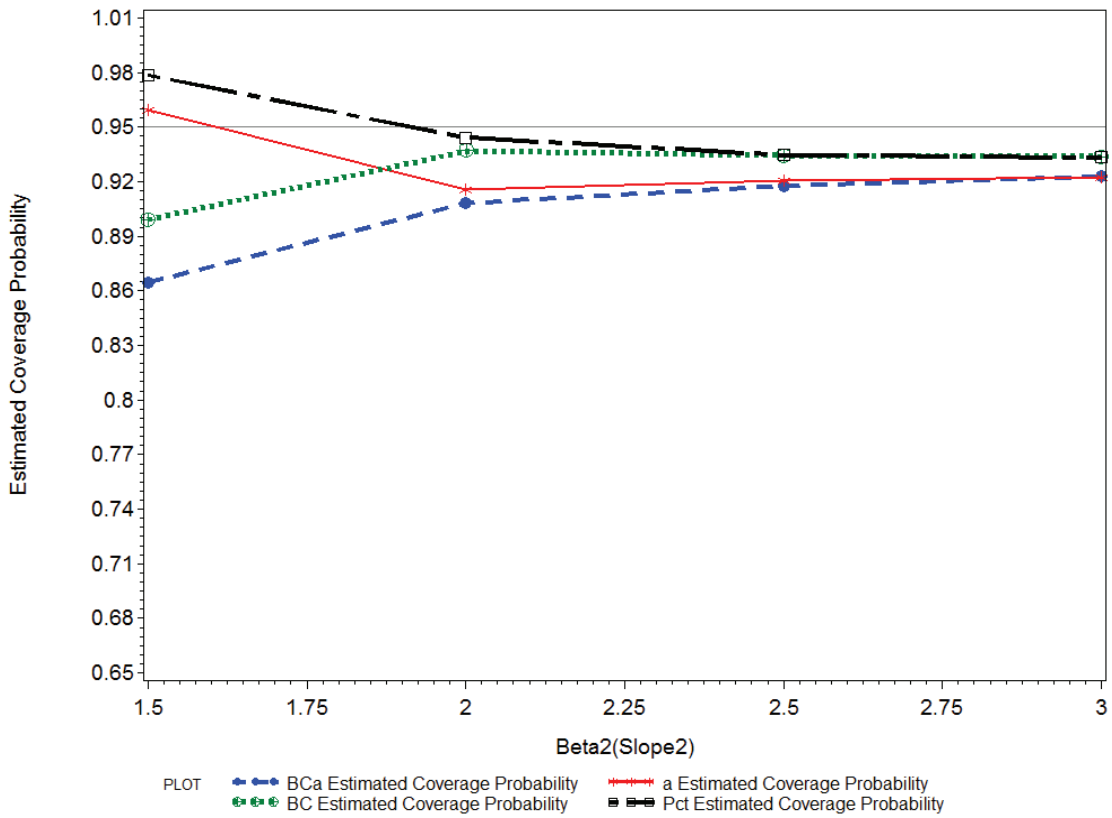

Figure 3.3: Estimated coverage probabilities of *BCa, a, BC*, and *Pct* for n=60 and $\tau = 3$

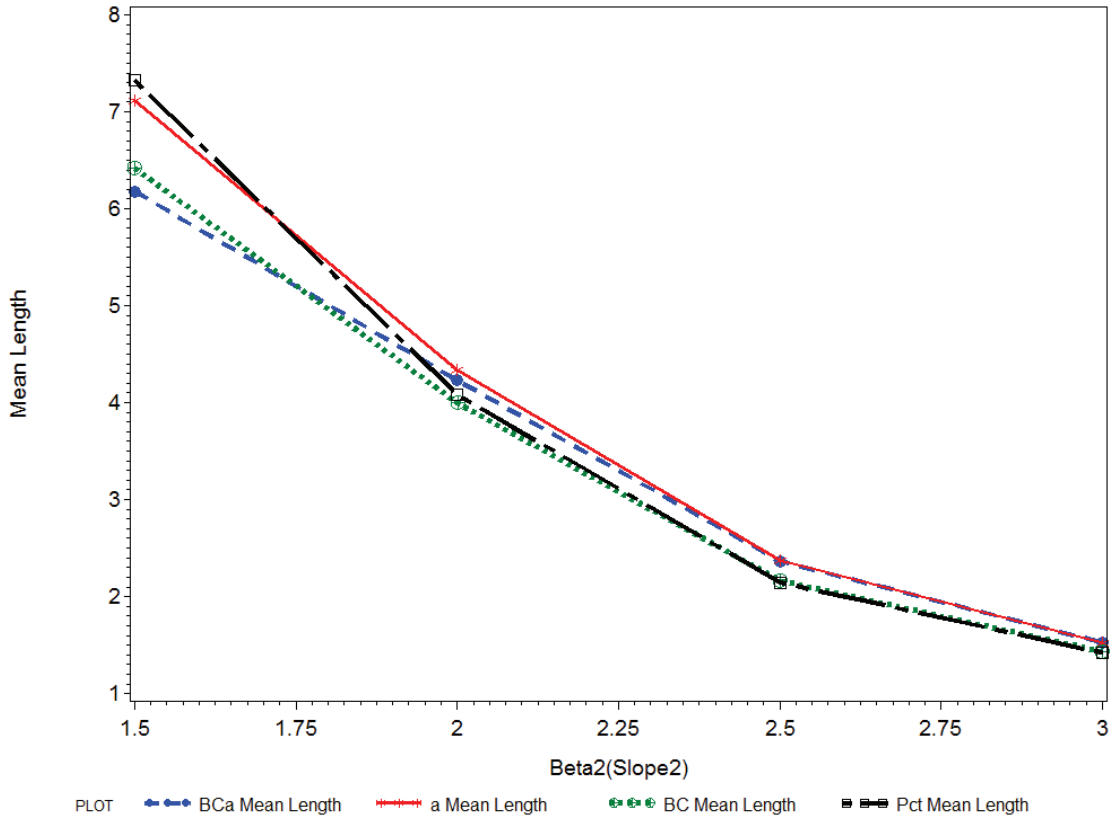

Figure 3.4: Mean length of *BCa, a, BC*, and *Pct* for n=60 and $\tau = 3$

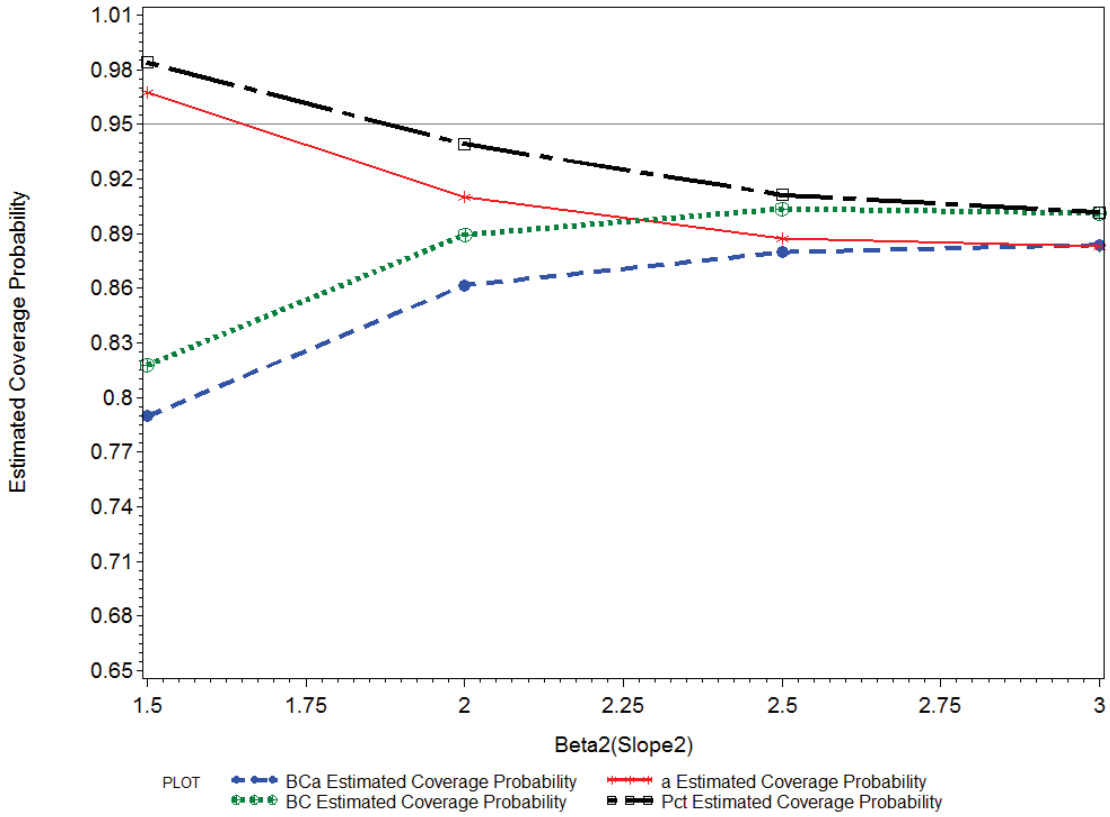

Figure 3.5: Estimated coverage probabilities of *BCa, a, BC*, and *Pct* for n=20 and $\tau = 5$

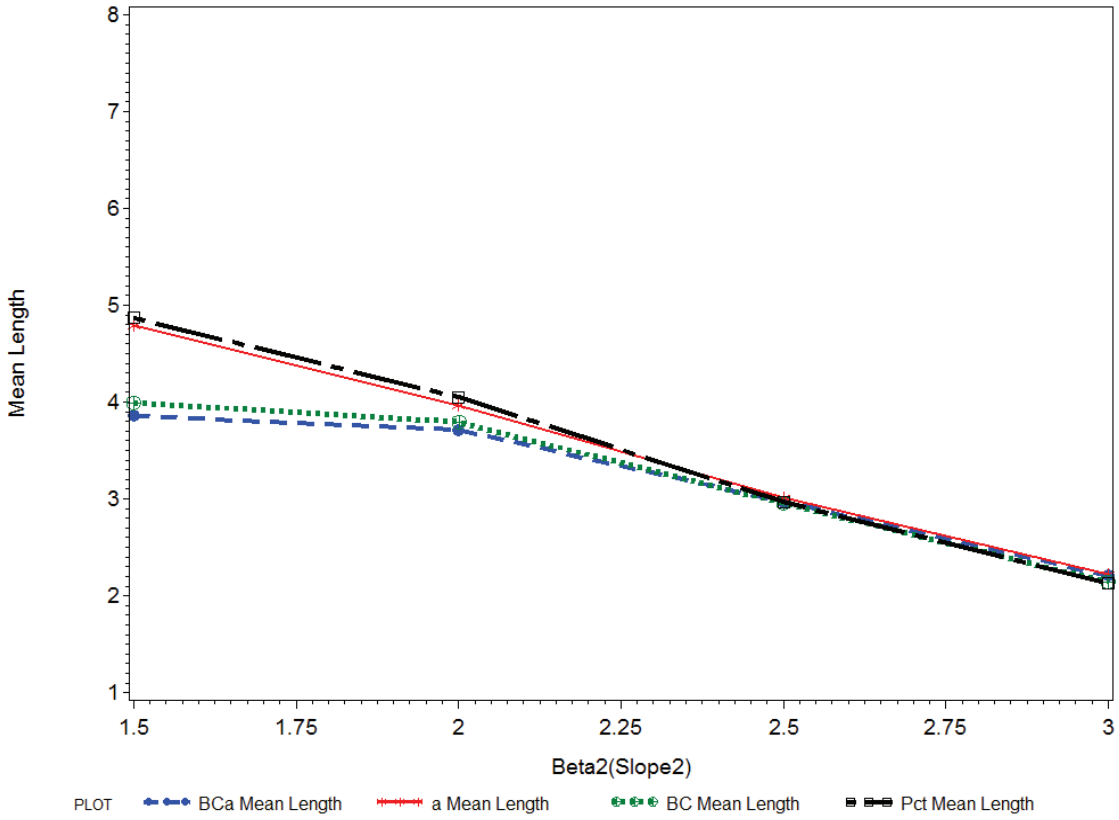

Figure 3.6: Mean length of *BCa, a, BC*, and *Pct* for n=20 and $\tau = 5$

and $\tau = 5$. According to the graph the mean length of four methods lies between 6 and 7 when $\beta_2 = 1.5$. Among all four methods, the *BCa* interval shows the lowest mean length. The mean length of intervals declines when β_2 increases. The mean length of both *BC* and *Pct* intervals are close to 1.3 and the other two are close to 1.4 when $\beta_2 = 3$.

Figures from 3.1 to 3.8 belong to the first category which is *BCa, a, BC* and *Pct* bootstrap confidence intervals. Next I will discuss graphs of the second category which is *BCaat*, *BCajk*, *BCajkat* bootstrap confidence intervals.

Figure 3.9 indicates estimated coverage probabilities of *BCaat*, *BCajk*, and *BCajkat* intervals for $n = 20$ and $\tau = 3$. *BCajkat* shows the highest estimated coverage probability and it remains constant for all β_2 values. *BCajkat* value lies between 89% and 90%. The estimated coverage probability of *BCajk* interval is close to 87% when β_2 =1.5 and then it slowly increase. When $\beta_2 = 3$ both *BCajkat* interval and *BCajk* interval follow the same pattern. *BCaat* interval has the lowest estimated coverage probability and it shows a considerable difference compared to other two.

Figure 3.10 shows the mean length of *BCaat*, *BCajk*,and *BCajkat* intervals for $n = 20$ and $\tau = 3$. In this graph, the mean length of all three intervals decline when β_2 increases. *BCaat* interval shows the lowest mean length and both *BCajk* and *BCajkat* mean lengths are very close for all β_2 values.

Figure 3.11 indicates estimated coverage probabilities of *BCaat*, *BCajk* and *BCajkat* intervals for $n = 60$ and $\tau = 3$. Both *BCajk*, and *BCajkat* intervals have same estimated coverage probabilities and they stay the same for all β_2 values. According to the graph *BCaat* interval shows the lowest estimated coverage probability and *BCajk*, and *BCajkat* intervals show approximately 93% estimated coverage probability.

Figure 3.12 shows the mean length of *BCaat*, *BCajk* and *BCajkat* intervals for $n = 60$ and $\tau = 3$. Both *BCajk*, and *BCajkat* intervals have same mean lengths for

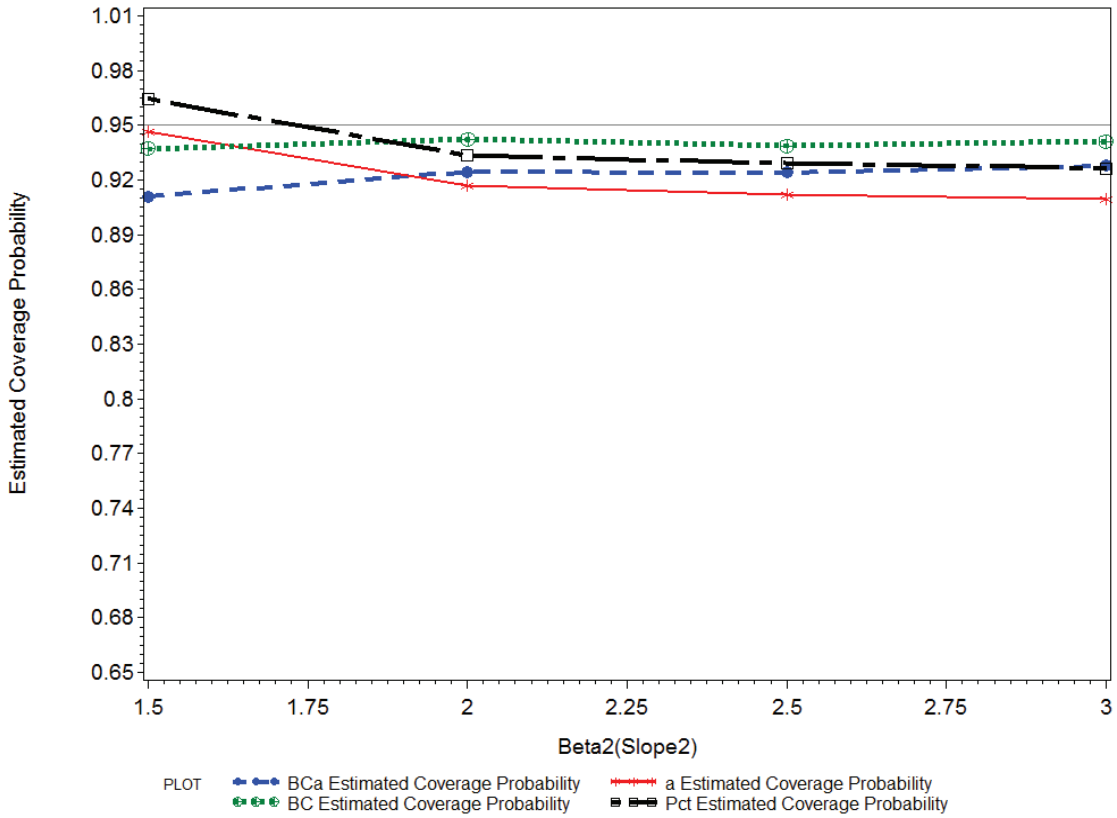

Figure 3.7: Estimated coverage probabilities of *BCa, a, BC*, and *Pct* for n=60 and $\tau = 5$

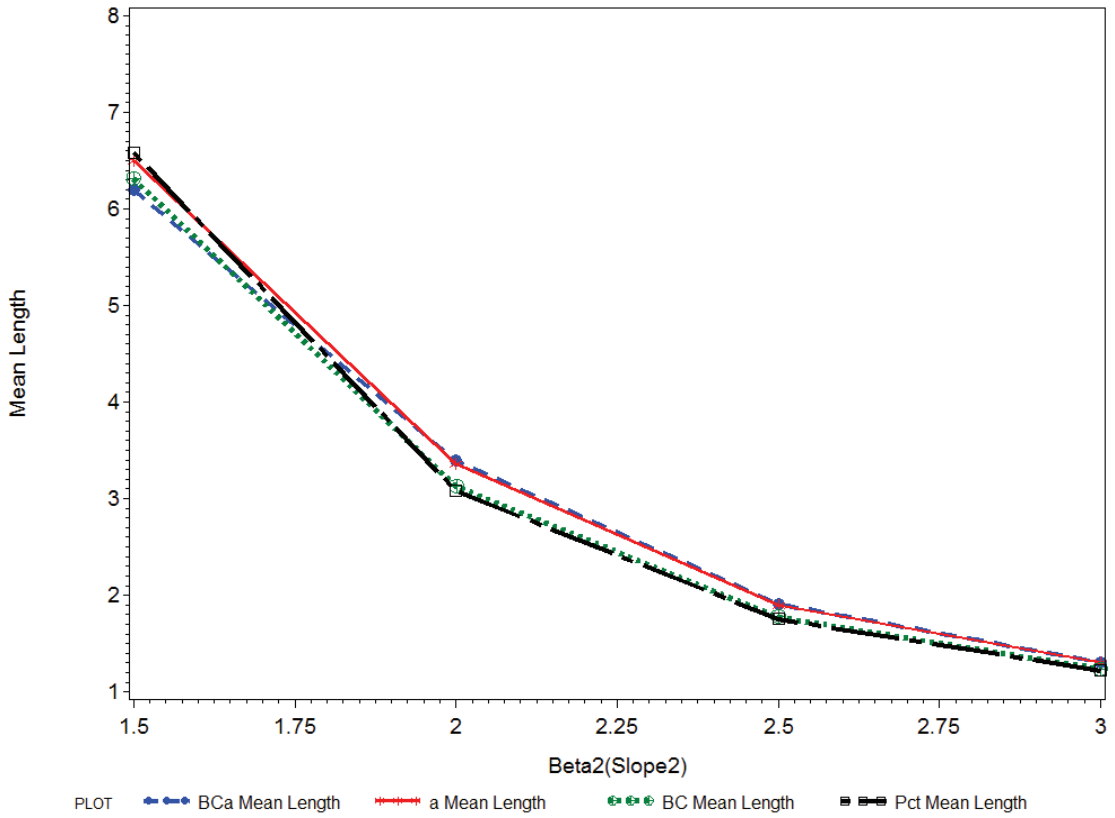

Figure 3.8: Mean length of *BCa, a, BC*, and *Pct* for n=60 and $\tau = 5$

all β_2 values. When β_2= 1.5 *BCaat* shows the lowest mean length and when β_2= 2 all three intervals have the same mean length. When $\beta_2 > 2$ both *BCajk*, *BCajkat* intervals show low mean length.

Figure 3.13 displays estimated coverage probabilities of *BCaat*, *BCajk*, and *BCajkat* intervals for $n = 20$ and $\tau = 5$. When $\beta_2 = 1.5$ the estimated coverage probabilities of the *BCajk* and *BCajkat* intervals are very close and both intervals show the highest estimated coverage probability. Both *BCajk* and *BCajkat* intervals have the same estimated coverage probabilities and remain constant when β_2 increases.

Figure 3.14 shows the mean length of *BCaat*, *BCajk*, and *BCajkat* intervals for $n = 20$ and $\tau = 5$. In this graph, it is hard to see the coordinates of *BCajk* interval because it is completely overlapping with *BCajkat* interval. When $\beta_2 = 1.5$ *BCaat* shows the lowest mean length and it is approximately close to 3.9. The mean length of both the *BCajk* and *BCajkat* intervals is approximately equal to 4.5 when $\beta_2 = 1.5$ and they show the lowest mean length when $\beta_2 = 3$.

The estimated coverage probabilities of *BCaat*, *BCajk*, and *BCajkat* intervals for $n = 60$ and $\tau = 5$ are shown in the Figure 3.15. The estimated coverage probabilities of *BCajk*, and *BCajkat* intervals are close to 94% when $\beta_2 = 1.5$. Similar to the Figure 3.13, the estimated values of both intervals are very close for all β_2 values. When β_2 increases, the estimated coverage probabilities of above two intervals show a small decline. *BCaat* interval shows the lowest coverage probability compared to other two intervals.

The mean length of *BCaat*, *BCajk*, and *BCajkat* intervals for $n = 60$ and $\tau = 5$ are shown in Figure 3.16. *BCaat* interval shows the minimum length when $\beta_2 = 1.5$. Both *BCajk* and *BCajkat* intervals show the minimum mean length for $\beta_2 \geq 2$.

In this chapter I discussed results of our simulations. In the next chapter I will present the conclusion and discussion based on these results.

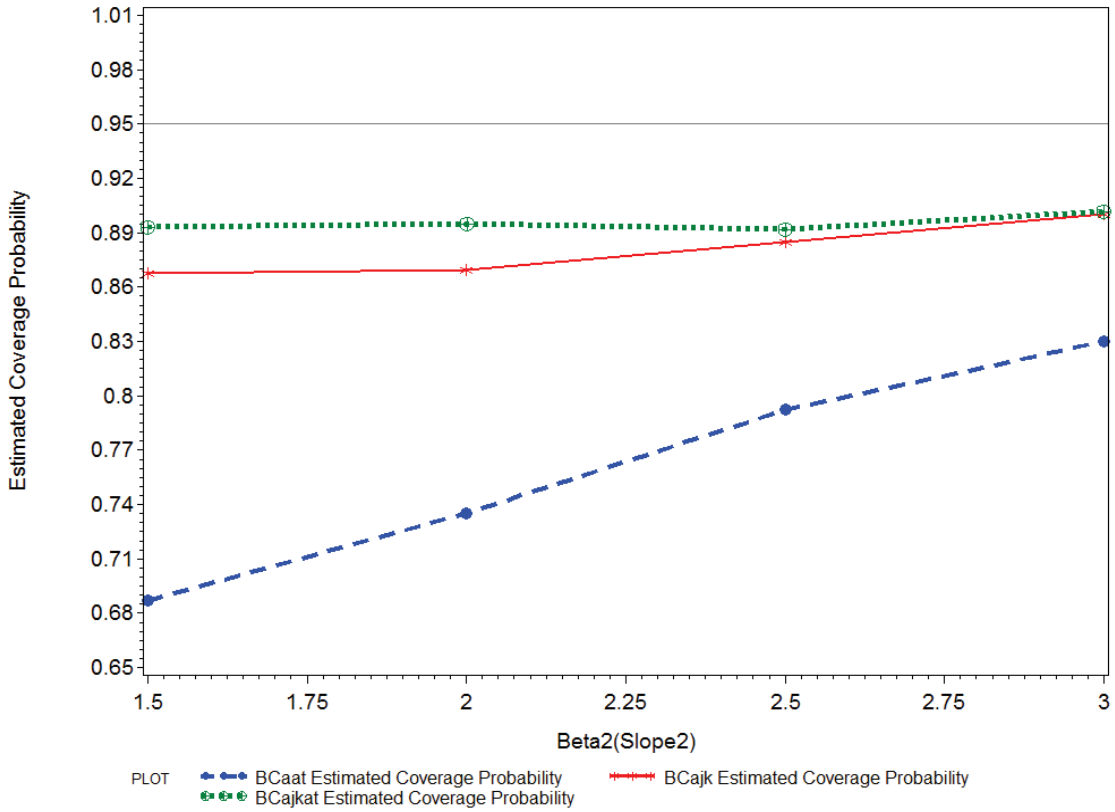

Figure 3.9: Estimated coverage probabilities of *BCaat, BCajk, BCajkat* for n=20 and $\tau = 3$

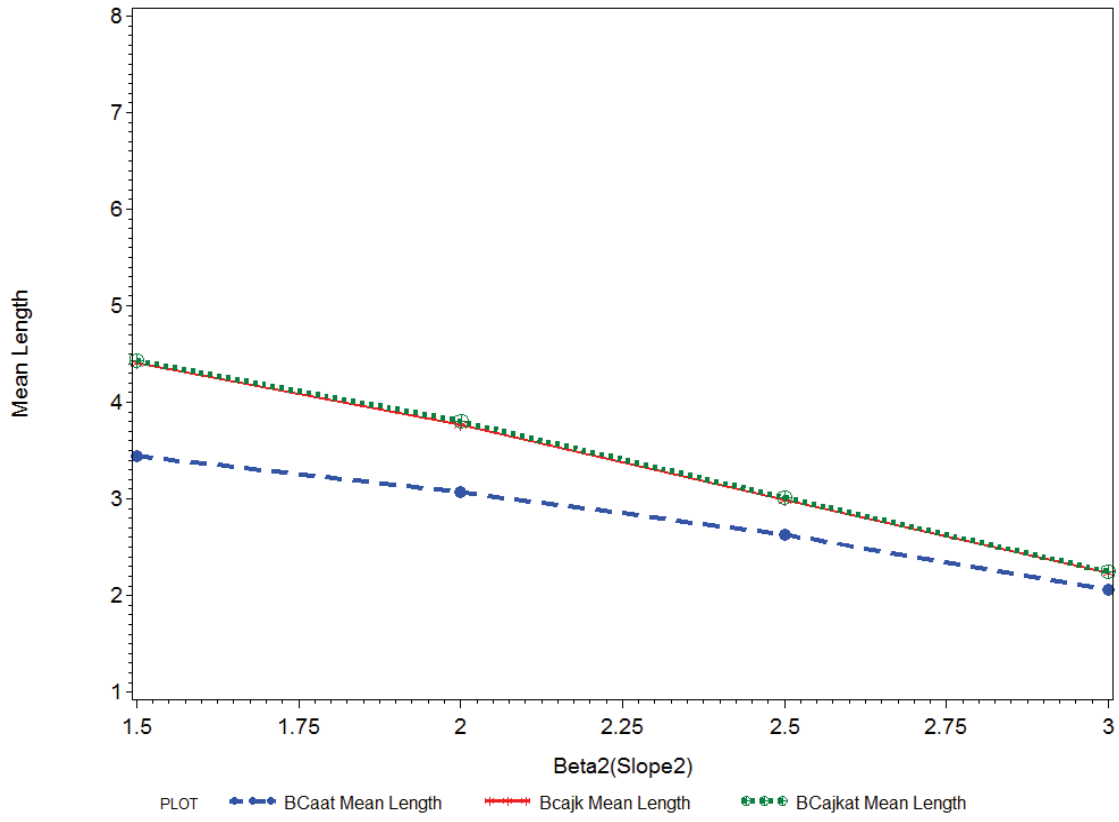

Figure 3.10: Mean length of *BCaat, BCajk, BCajkat* for n=20 and $\tau = 3$

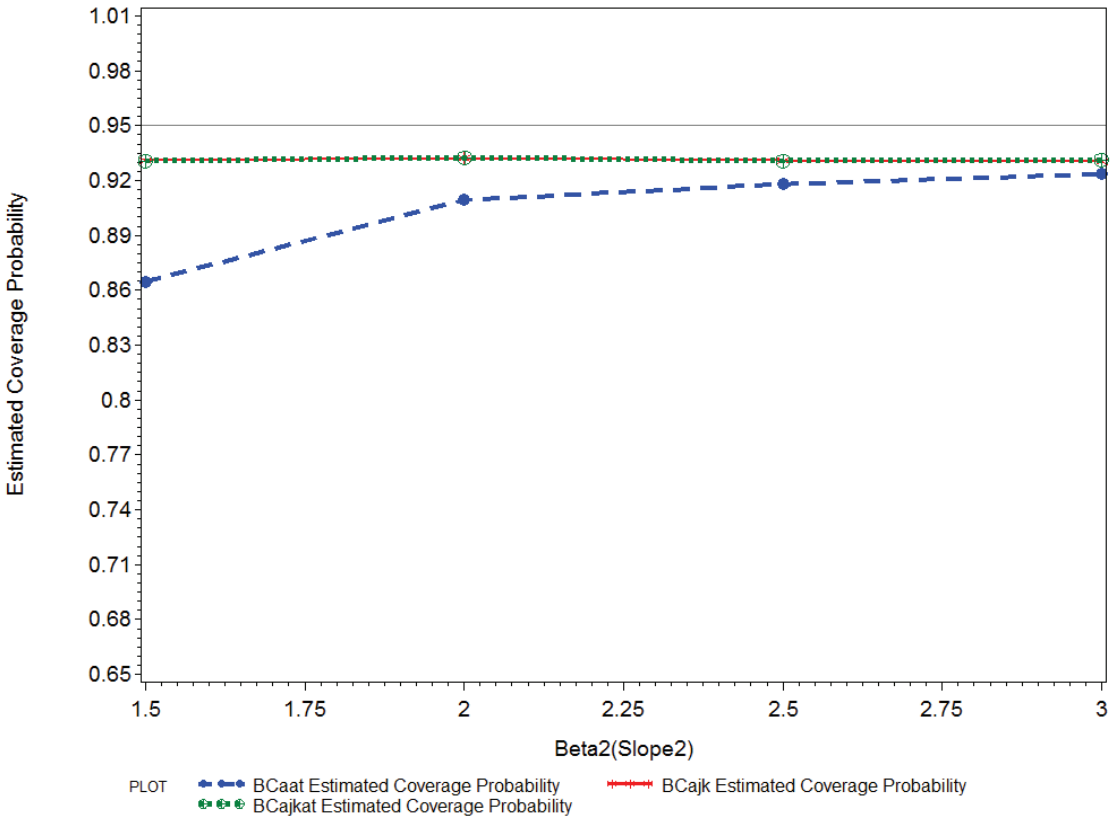

Figure 3.11: Estimated coverage probabilities of *BCaat, BCajk, BCajkat* for n=60 and $\tau = 3$

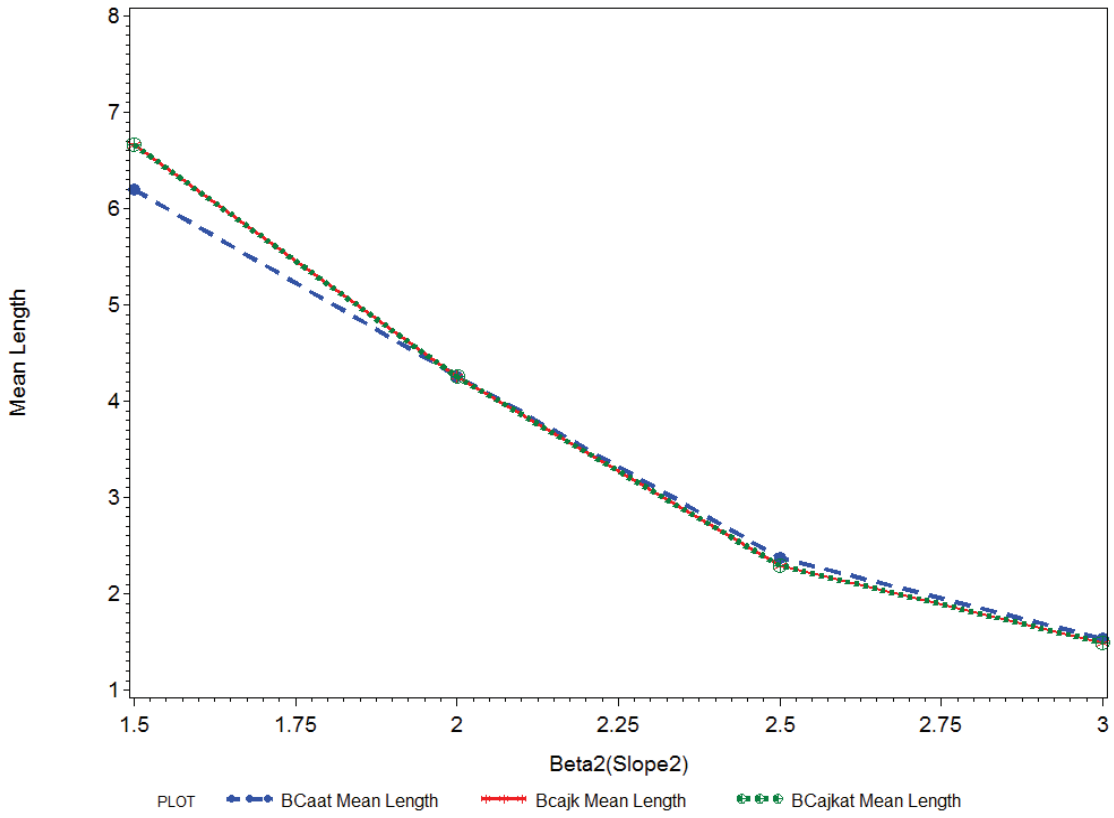

Figure 3.12: Mean length of *BCaat, BCajk, BCajkat* for n=60 and $\tau = 3$

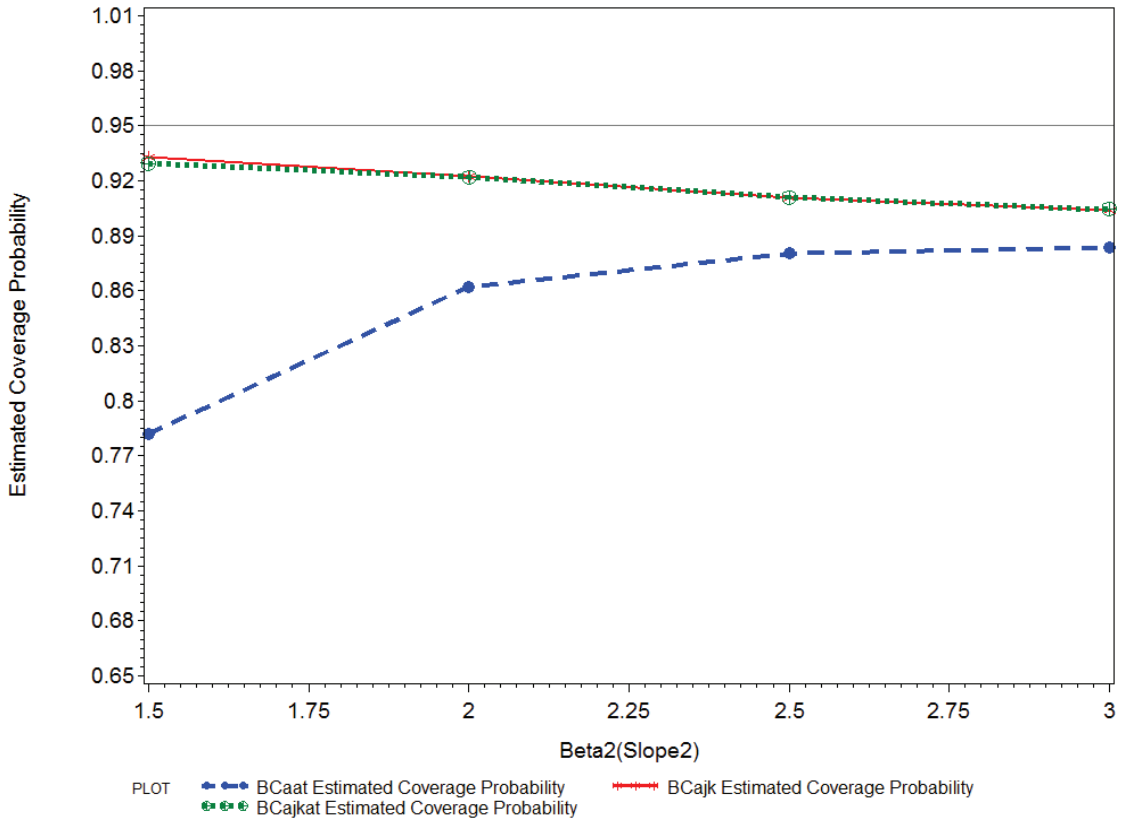

Figure 3.13: Estimated coverage probabilities of *BCaat, BCajk, BCajkat* for n=20 and $\tau = 5$

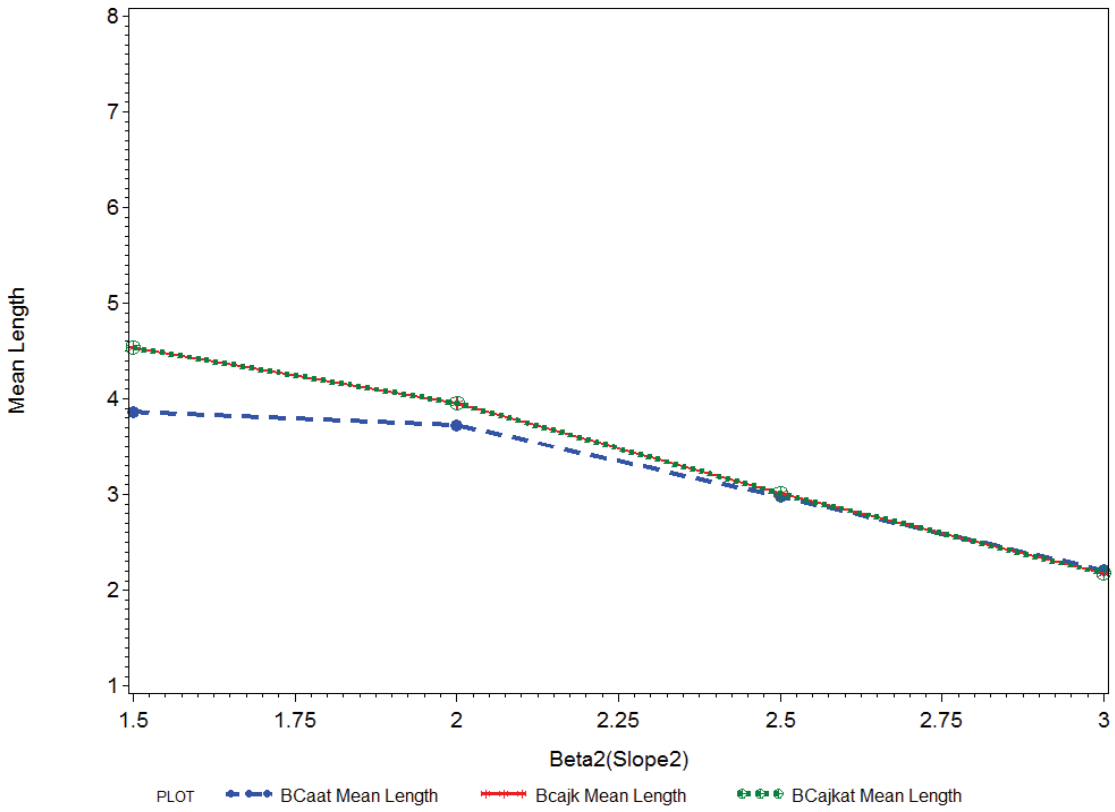

Figure 3.14: Mean length of *BCaat, BCajk, BCajkat* for n=20 and $\tau = 5$

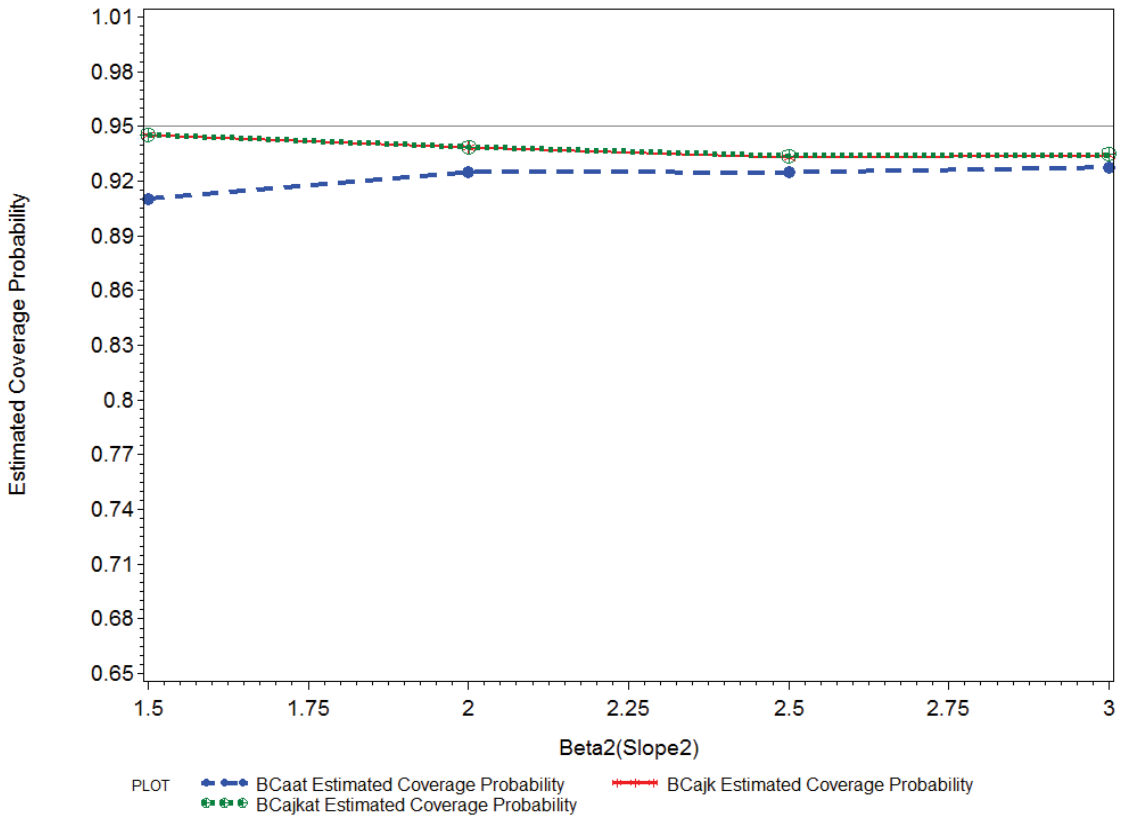

Figure 3.15: Estimated coverage probabilities of *BCaat, BCajk, BCajkat* for n=60 and $\tau = 5$

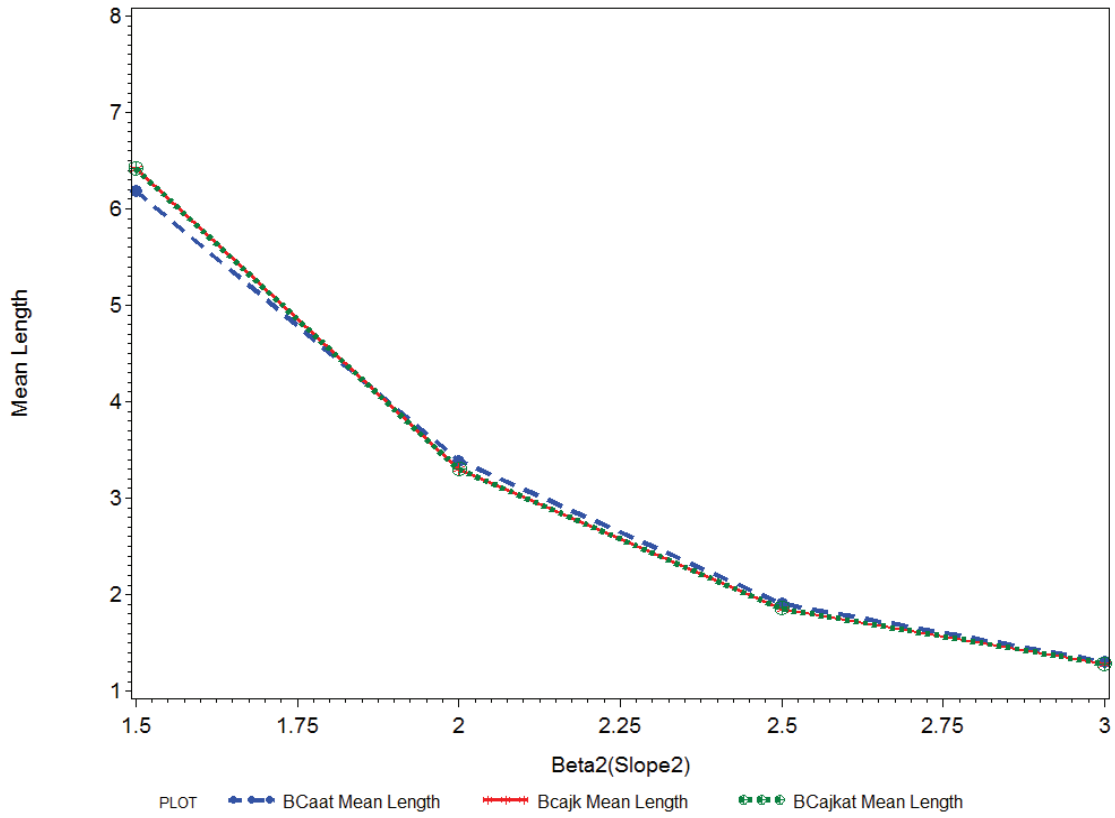

Figure 3.16: Mean length of *BCaat, BCajk, BCajkat* for n=60 and $\tau = 5$

CHAPTER 4

DISCUSSION AND CONCLUSION

4.1 Summary of Previous Chapters

As I mentioned in Chapters 2 and 3, the main purpose of this research is to choose a "better" (having minimum length and close to 95% expected coverage probability) confidence interval for a regression changepoint among seven different bootstrap confidence interval methods.

The model of a simple linear regression with one continuous changepoint can be stated as

$$Y_i = \begin{cases} \alpha_1 + \beta_1 x_i + \epsilon_i & x_i \leq \tau \\ \alpha_2 + \beta_2 x_i + \epsilon_i & x_i > \tau \end{cases} \tag{4.1}$$

with continuity constraint

$$\alpha_1 + \beta_1 \tau = \alpha_2 + \beta_2 \tau. \tag{4.2}$$

The changepoint is denoted by τ.

Seven different methods based on bootstrap samples were used to find a confidence interval for the above regression changepoint. The methods to construct seven different bootstrap confidence intervals and the way of doing my research were discussed in Chapter 2. Analysis of these methods was discussed in Chapter 3.

4.2 Summary of Findings

As I mentioned in the Chapter 3, I only considered n=20 and n=60 with τ=3 and τ=5 for detailed discussion. The estimated coverage probabilities and mean lengths of each bootstrap confidence intervals for $n = 40$ and different changepoints are in between corresponding results of $n = 20$ and $n = 60$. As an example, the estimated

coverage probability of *BCaat*, *BCajk*, *BCajkat* intervals when β_2 =1.5 for $n = 20$ and $\tau = 3$ is 89.5%, $n = 40$ and $\tau = 3$ is 90%, and $n = 60$ and $\tau = 3$ is 93.5% .

It is not clear to compare all seven bootstrap confidence intervals in one graph. Therefore, I split them into two categories. In the first category (Figures 3.1-3.8) *BCa, a, BC* and *Pct* intervals are considered and in the second category (Figures 3.9-3.16) *BCaat*, *BCajk*, and *BCajkat* intervals are considered.

It is not necessary to consider estimated coverage probabilities for the *BCa* and *BC* intervals in Figure 3.1 because they are considerably less than 95%. The estimated coverage probability of *a* interval is close to 95% when $\beta_2 = 1.5$. For other β_2 values *Pct* interval is close to 95%. *BCa* interval shows the minimum mean length for all β_2 values in Figure 3.2. So it can be concluded that *Pct* interval has a better estimated coverage probability and *BCa* has the minimum length for $n = 20$ and $\tau = 3$.

The estimated coverage probabilities of *Pct* and *BC* intervals are close to 95% when $\beta_2 \geq 2$ and *a* interval is close to 95% when β_2 =1.5 for $n = 60$ and $\tau = 3$. Both *Pct* and *BC* intervals have the minimum length when $\beta_2 \geq 2$ and *BCa* interval has the minimum length when β_2=1.5 for $n = 60$ and $\tau = 3$. So it can be concluded that both *Pct* and *BC* intervals have a better estimated coverage probability and minimum length when $\beta_2 \geq 2$ for $n = 60$ and $\tau = 3$.

According to Figure 3.5, it is clear that *Pct* interval shows a better estimated coverage probability and *BCa* interval shows the minimum mean length for $n = 20$ and $\tau = 5$. The estimated coverage probability of *BC* interval is close to 95% for all β_2 values and both *Pct* and *BC* intervals show minimum length when $\beta_2 \geq 2$ for $n = 60$ and $\tau = 5$.

The estimated coverage probability of *BCaat*, *BCajk*, and *BCajkat* intervals for $n = 20$ and $\tau = 3$ are shown in Figure 3.9. There is no interval which has estimated coverage probability close to 95%. *BCajkat* interval gives the highest estimated cov-

erage probability which is close to 89% for all β_2 values. The minimum mean length is given by *BCaat* interval for all β_2 values for $n = 20$ and $\tau = 3$. So it can be concluded that *BCajkat* interval has a better estimated coverage probability and *BCaat* interval shows minimum length for $n = 20$ and $\tau = 3$.

Both *BCajk*, and *BCajkat* intervals show a better estimated coverage probability for all β_2 values and minimum mean length when $\beta_2 \geq 2$ for $n = 60$ and $\tau = 3$. According to Figures 3.13 and 3.14 both *BCajk*, and *BCajkat* intervals have a better estimated coverage probability, but *BCaat* interval shows minimum mean length for all β_2 values for $n = 20$ and $\tau = 5$.

In Figure 3.15 both *BCajk*, and *BCajkat* intervals show a better estimated coverage probability. Its value is close to 95% when β_2=1.5 and slightly decrease when β_2 is increase. In Figure 3.16 both *BCajk*, and *BCajkat* intervals show minimum mean length when $\beta_2 \geq 2$. So it can be concluded that both *BCajk*, and *BCajkat* intervals have a better estimated coverage probability and minimum length for $n = 60$ and $\tau = 5$.

4.3 Discussion

In the previous section, I summarized confidence intervals which gives better estimated coverage probabilities and minimum lengths of each sample size and each changepoint. In this section I will summarize all these estimated coverage probabilities and mean lengths to find the a better bootstrap confidence interval for regression changepoint. I have summarized the results of first category which is *BCa, a, Bc,* and *Pct* intervals in Table 4.1. The results of second category which is *BCajk*, *BCaat*, and *BCajkat* intervals are summarized in Table 4.2.

According to Table 4.1, *Pct* interval shows a better estimated coverage proba-

n	τ	Better estimated Coverage Probability	Minimum Mean Length
20	3	*Pct* interval	*BCa* interval
60	3	*Pct* and *BC* intervals	*Pct* and *BC* intervals ($\beta_2 \geq 2$)
20	5	*Pct* interval	*BCa* interval
60	5	*BC* interval	*Pct* and *BC* intervals ($\beta_2 \geq 2$)

Table 4.1: First Category Best Results

n	τ	Better estimated Coverage Probability	Minimum Mean Length
20	3	*BCajkat* interval	*BCaat* interval
60	3	*BCajkat* and *BCajk* intervals	*BCajkat* and *BCajk* intervals ($\beta_2 \geq 2$)
20	5	*BCajkat* and *BCajk* intervals	*BCaat* interval
60	5	*BCajkat* and *BCajk* intervals	*BCajkat* and *BCajk* intervals ($\beta_2 \geq 2$)

Table 4.2: Second Category Best Results

bility and *BCa* interval gives the minimum mean length for $n = 20$ and $\tau = 3$ and 5. We can't make a conclusion on a better confidence interval for $n = 20$, because no interval gives a better estimated coverage probability with the minimum mean length. *BC* interval shows a better estimated coverage probability and the minimum mean length when $\beta_2 \geq 2$ for $n = 20$ and $\tau = 3$ and 5.

According to Table 4.2, a better estimated coverage probability (close to 95%) for $n = 20$ and $\tau = 3$ is given by *BCajkat* and the minimum length is given by *BCaat*. A better estimated coverage probability for other sample sizes and changepoints are given by *BCajkat* and *BCajk* intervals. For the $n = 60$ and $\tau = 3$ and 5, *BCajkat* and *BCajk* intervals give the minimum mean length when $\beta_2 \geq 2$, but for $n = 20$ and $\tau = 5$ *BCaat* interval gives the minimum length.

From Table 4.1, BC interval can be chosen as a better confidence interval (better estimated coverage probability and minimum mean length) when $\beta_2 \geq 2$ and $n = 60$. From Table 4.2, we can choose both *BCajkat* and *BCajk* intervals, which gives a better mean coverage probability and the minimum mean length when $\beta_2 \geq 2$ and $n = 60$.

So we can choose one method from the first category (BC interval) and two methods from the second category (*BCajkat* and *BCajk* intervals) as better bootstrap confidence intervals when $\beta_2 \geq 2$ and $n = 60$. Next we have to choose a better method among these three intervals which gives approximately 95% estimated coverage probability and minimum mean length. Therfore we need to compare the estimated coverage probabilities and mean lengths of above three methods.

Figure 4.1 displays the estimated coverage probability of BC, *BCajk*, and *BCajkat* intervals for $n = 60$ and $\tau = 3$. According to Figure 4.1, the estimated coverage probability of BC interval is close to 95% compare to *BCajkat* and *BCajk* intervals when $\beta_2 \geq 2$ for $n = 60$ and $\tau = 3$. Figure 4.2 displays the mean lengths of BC, *BCajk*, and *BCajkat* intervals for $n = 60$ and $\tau = 3$. BC interval shows the minimum mean length for all β_2 values.

According to Figure 4.3 the estimated coverage probability of BC interval is close to 95% compared to *BCajkat* and *BCajk* when $\beta_2 \geq 2$ for $n = 60$ and $\tau = 5$. Figure 4.4 displays, the mean lengths of BC, *BCajk* and *BCajkat* intervals for $n = 60$ and $\tau = 5$. Among these three methods BC interval shows the minimum mean length for all β_2 values.

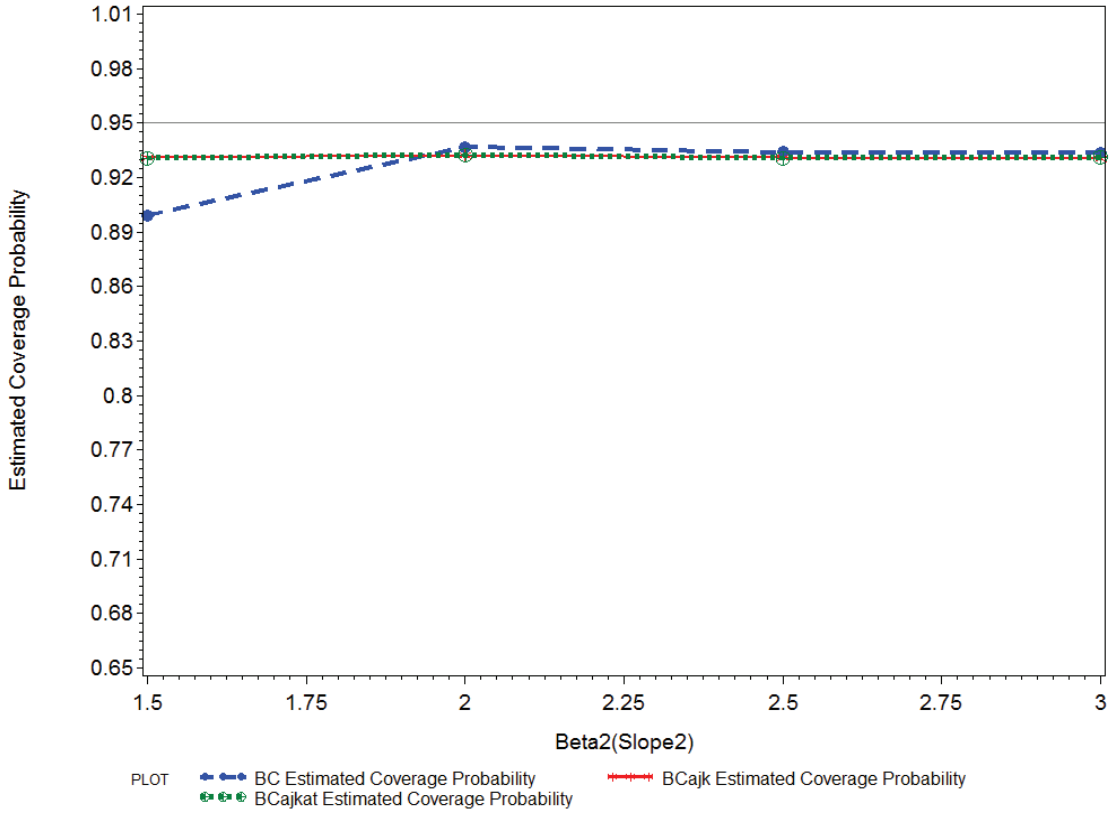

Figure 4.1: Estimated coverage probabilities of *BC, BCajk, BCajkat* for n=60 and τ=3

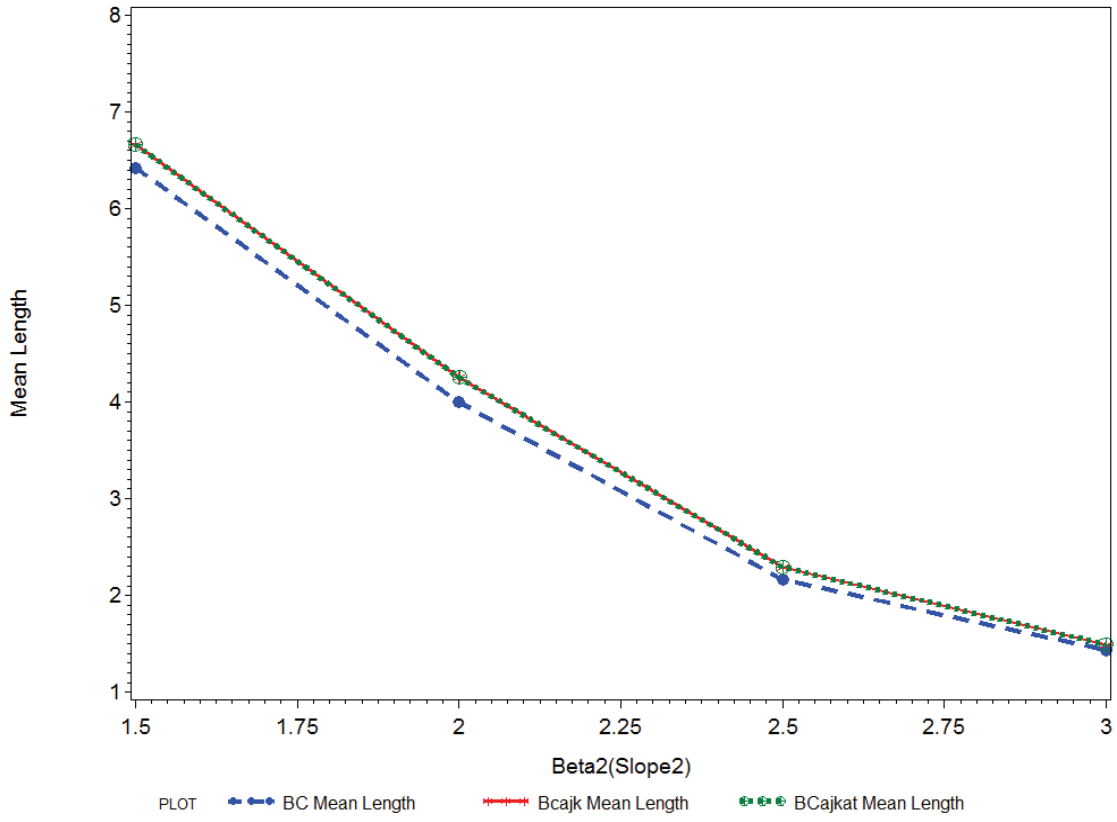

Figure 4.2: Mean length of *BC , BCajk , BCajkat* for n=60 and τ=3

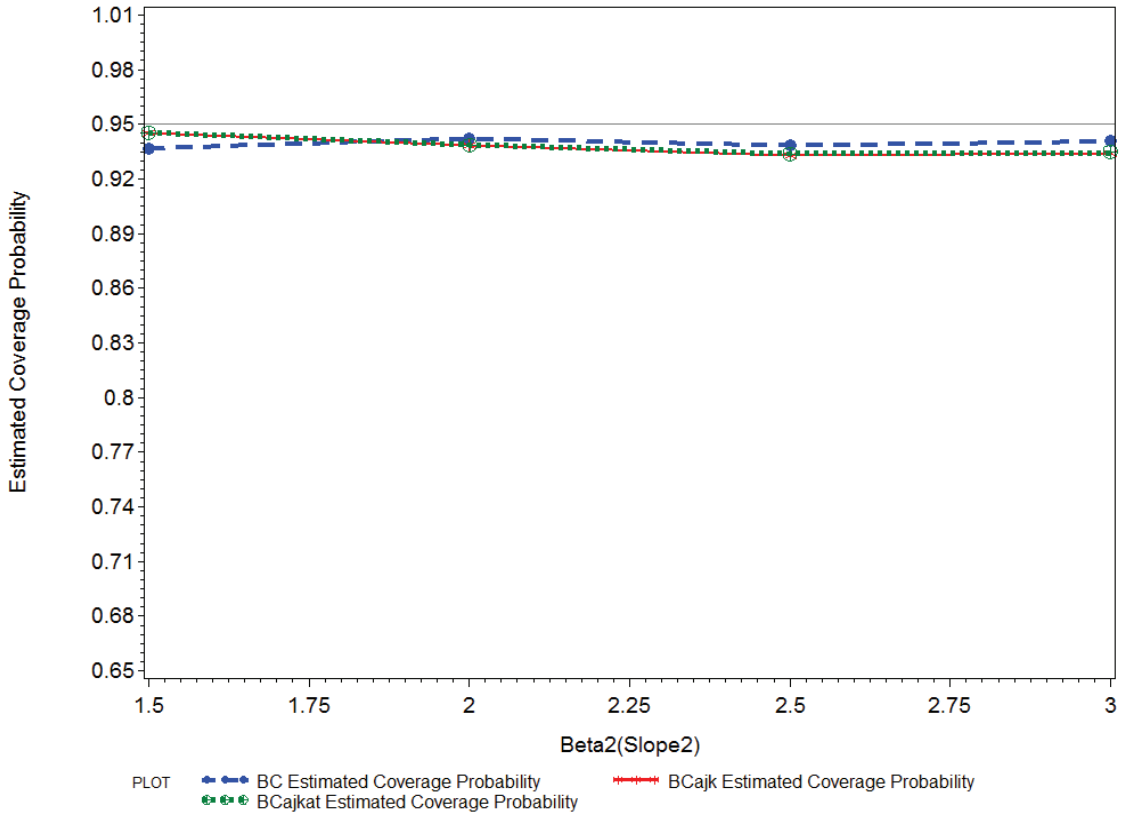

Figure 4.3: Estimated coverage probabilities of BC , $BCajk$, $BCajkat$ for n=60 and τ=5

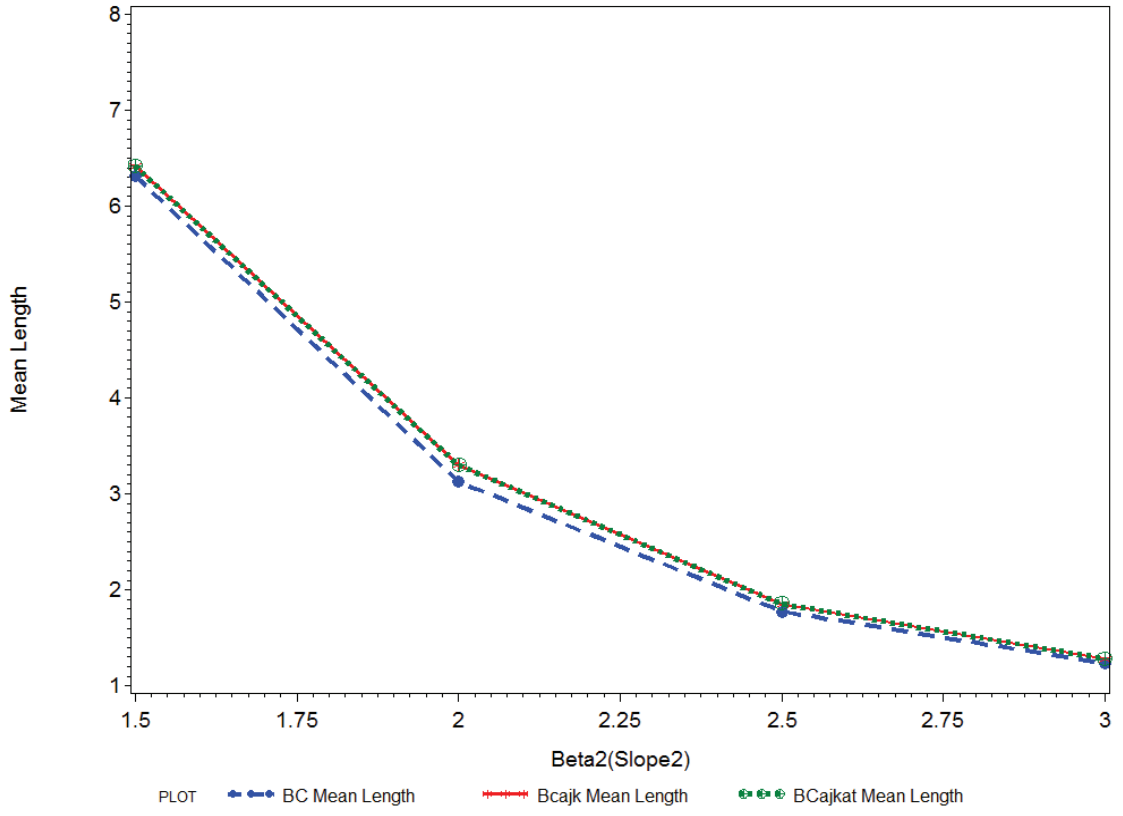

Figure 4.4: Mean length of BC , $BCajk$, $BCajkat$ for n=60 and τ=5

4.4 Conclusion and Remarks

By comparing these four figures, it can be concluded that BC interval gives a better estimated coverage probability which is close to 95% when $\beta_2 \geq 2$ and $n = 60$. Also BC interval gives minimum length for all β_2 values for $n = 60$. Since we couldn't find one bootstrap confidence interval which gives a better estimated coverage probability and minimum mean length, we cannot make a conclusion about $\beta_2 = 1.5$ for $n = 60$. If we look at graphs of BC interval for $n = 60$ (Figures 3.3, 3.4, 3.7, 3.8) it is clear that when β_2 increases estimated coverage probability of BC interval increases and mean length decreases. So we can conclude that for large β_2 values BC interval gives more accurate results. Also it can be concluded that when sample size increases BC interval clearly shows a better estimated coverage probability with minimum mean length.

Finally we can conclude that among all seven bootstrap confidence interval methods BC interval gives a better estimated coverage probability and minimum mean length for $n = 60$ when $\beta_2 \geq 2$ for our simulated data. This can be explained using the error distribution of the data and definition of bias correction. Bias correction is obtained directly from the proportion of bootstrap replications less than the original estimate. In my research errors are normally distributed with mean zero and variance one. That implies half of the bootstrap replicates values are less than or equal to original estimate. This means the bias correction is close to zero.

I have made conclusions based on my simulated data for a given changepoint and given nominal coverage probability. In practice one can come up with different scenarios and can ask which bootstrap confidence interval method gives a better confidence interval for each scenario. Based on my findings in general I can conclude that for large sample sizes ($n \geq 40$) BC interval gives a better coverage probability

and minimum mean length out of seven methods for higher β_2 values. No method works well for small sample sizes. Since errors are normally distributed with mean zero and variance one errors have no bias or skewness to correct for, so that may be a reason why those methods didn't perform well.

In my research the regression function is assumed to be continuous at the point of change and we only consider a simple linear regression model with one changepoint. Since there are some limitation in my research we cannot make a conclusion about discontinuous changepoint models and multiple linear regression models.

4.5 Future Work

In this research I used only three new bootstrap confidence interval methods, which are constructed from our four basic bootstrap confidence interval methods. In addition to these three, we can construct four more methods which are listed in below. To construct these methods, I could use a jackknife estimator and adjustment for ties.

1. Bias-corrected method with average jackknife estimator
2. Bias-corrected method with adjustment for ties
3. Bias-corrected method with average jackknife estimator and adjustment for ties
4. Acceleration method with adjustment for ties

In future we can develop our SAS code to run the above four methods. Then it can be used to check if there is any method which gives estimated coverage probability close to 95% and minimum length.

REFERENCES

[1] M.Basseville and I.V.Nikiforov, *Detection of Abrupt Changes*, Prentice-Hall, Inc,Eaglewood Cliffs, N.J (1993).

[2] Wayne A. Taylor, *Change-Point Analysis: A Powerful New Tool For Detecting Changes*, Statistica Sinica,(2000).

[3] Tze Leung Lai, *SEQUENTIAL ANALYSIS:SOME CLASSICAL PROBLEMS AND NEW CHALLENGES*, Statistica Sinica,303-408,(2001).

[4] Iain Pardoe, *Applied Regression Modeling*, Published by John Wiley and Sons, Inc., Hoboken, New Jersey,2nd ed, (1970).

[5] A. Khodadadi and M.Asgharian *Changepoint Problem and Regression:An Annotated Bibliography*, The Berkeley Electronic Press, (2008)

[6] P.Adams and J.C.Mackay, *Bayesian Online Changepoint Detection*, (2007).

[7] P. Bhattacharya, *Some Aspects of Change-Point Analysis*, IMS Lecture Notes - Monograph Series, Vol.23, 28-56(1994).

[8] P.Sprent, *Some hypothesis concerning two-phase regression lines*, Biometrics, Vol.7,634-645 (1961).

[9] Kenneth J.Rothman, *Epidemiology*, Published by Oxford University Press Inc, New York 10016 (2002).

[10] P.I.Feder, *On asymptotic distribution theory in segmented regression problems identified case*, The Annals of Statistics, Vol.3, No.1, 49-83, (1975).

[11] D.V.Hinkley, *Inference in two-phase regression*, Journal of the American Statistical Association,Vol.67, No.1, 306-310, (1971).

[12] R.E.Quandt, *Tests of the hypothesis that a linear regression system obeys two separate regimes*, Journal of the American Statistical Association, 324-330, (1960).

[13] R.L.Brown and J.Durbin and J.M.Evans *Techniques for Testing the Constancy of Regression Relationships over Time*, Journal of the Royal Statistical Society, Vol.37, No.2, 149-192, (1971).

[14] D.T Judith and M.L.Lesperance, *PIECEWISE REGRESSION: A TOOL FOR IDENTIFYING ECOLOGICAL THRESHOLDS*, Ecological Society of America, Vol.8, 2034-2041, (2003).

[15] Song S. Qian and Ryan King and Curtis J. Richardson, *Two Statistical Methods for the Detection of Environmental Thresholds*, (2001).

[16] M. Banerjee and I. McKeague, *Confidence Sets for Split Points in Decision Trees*, The Annals of Statistics, Vol.35, 543-574, (2007).

[17] N. Barrowman and R. Myers, *Still More Spawner-Recruitment Curves: The Hockey Stick and its Generalizations*, Canadian Journal of Fisheries and Aquatic Sciences, Vol.57, 665-676, (2000).

[18] D. Barry and J. Hartigan, *A Bayesian Analysis for Change Point Problems*, Vol.88, No.421, 309–319, (1993).

[19] B. Efron and R. Tibshirani, *An Introduction to the Bootstrap*,Chapman & Hall, New York, USA, (1993)

[20] S. A. Julious *Inference and Estimation in a Changepoint Regression Problem*,The Statistician,50,(2001)

[21] M. Huskova and C. Kirch *Bootstrapping confidence intervals for the change-point of time series*,Journal of Time Series Analysis,29:947-972, (2008)

[22] Dumbgen, L. *The asymptotic behavior of some nonparametric changepoint estimators* Ann. Statist, 19:1471-1495,(1991)

[23] E. Seijo and B. Sen *Change-point in stochastic design regression and the bootstrap*,Ann. Statist. Volume 39, Number 3 (2011), 1580-1607.

[24] Toms, Judith D. and Lesperance, Mary *Piecewise regression: A tool for identifying ecological thresholds*,Ecology, 84, 2034-2041,(2003)

[25] M.R.Chernick *Bootstrap Methods*,John Wiley and Sons,Inc.,Canada(1999)

Appendix A

SAS CODE

This SAS (version 9.3) code was co-developed by Dr. Jonathan Duggins and Dr.James Blum (University of North Carolina Wilmington). As I mentioned in Section 2.3, I used this SAS code to construct seven different confidence intervals for a regression changepoint.

1. Generate random sample data (sample size is n) with a given changepoint.
2. Estimate the changepoint ($\hat{\tau}$) from that original sample data
3. Select 1000 independent bootstrap samples from that original sample data.
4. Estimate the changepoint in each bootstrp sample.
5. Create jackknife estimates for our original data by deleting one data point each time from the original data set. Then find the average jackknife estimate.
6. Compute all seven confidence intervals.
7. Finally this code is used to run 10,000 simulations to effectively compare the confidence intervals.

```
libname local "C:\Users\buddhi\Desktop\Local";
options sasmstore=sasuser mstored nomlogic nomprint nosymbolgen nonotes nosource ls=80;
%let int=0;
%let slope1=1;
%let slope2=1;
%let slopename = 10;
%let cp=4.01;
%let cpname = 4;
%let conf=95;
%let int2=%sysevalf(&cp*(&slope1-&slope2));
%let stderr=1;
%let sampsz=40;
%let sims=1000;
%let bootsize = 1000;
%let start=%sysfunc(time()); %let st=%sysfunc(time());

proc datasets nolist nowarn library=work kill;
run;
quit;

data work.sim;
	do sim=1 to &sims;
		do x=10/&sampsz to 10 by 10/&sampsz;
			if x le &cp then do;
				y=&int+&slope1*x+&stderr*rannor(0);
			end;
			else do;
				y=&int2+&slope2*x+&stderr*rannor(0);
			end;
			j = x*&sampsz/10;
			output;
		end;
	end;
run;

proc surveyselect data=work.sim out=work.accel method=srs samprate=100 reps=&sampsz
noprint;
	strata sim;
run;

data work.accel2;
	set work.accel;
	if replicate = j then delete;
	type = 'Jacknife';
run;
```

```
data work.jn;
        length type $10;
        set work.accel2 work.sim(in=inorig);

        if inorig then do;
                type = 'Original';
                replicate = 1;
                end;

        rename replicate=jn;
run;

proc sort data=jn;
        by sim type jn;
run;
%macro twoline;


/******fit one line models (and get residuals)******/
proc reg data=work.jn noprint
        outest=work.result0(drop=_model_ _type_ _depvar_ _rmse_ y _in_ _p_ _edf_);
        by sim type jn;
        model y=x/sse;
        output out=work.residuals r=resid;
run;
quit;

/*******get replicate data sets*******/
proc surveyselect data=work.jn out=work.all
                samprate=1 method=srs reps=&sampsz noprint;
        strata sim type jn ;
run;

/********set up dummy variables for two line modelling******/
data work.all;
        set work.all;
        if x le replicate*10/&sampsz then do;
                dummy=0; xdummy=0;
                end;
                else do;
                dummy=1; xdummy=x;
                end;
run;

/*******fit two line, unconstrained******/
proc reg data=work.all noprint
```

```
        outest=work.result(drop=_model_ _type_ _depvar_ _rmse_ y _in_ _p_ _edf_);
        by sim type jn replicate;
        where replicate between 5 and %eval(&sampsz-5);
        model y=dummy x xdummy/sse;
run;
quit;


/*********check for meeting points********/
data work.result;
        set work.result;
        mt=-dummy/xdummy;
        if mt ge replicate*10/&sampsz-.01 and mt le (replicate+1)*10/&sampsz+.01 then do;
meet=1; uc=1; end;
                else do; meet=0; uc=0; end;
run;

/********clear constrained result data set (for appending)******/
proc datasets nolist nowarn library=work;
        delete resultc;
run;


/*******fit restriced models, compile results*****/
%macro restrict;
%do rep=5 %to %eval(&sampsz-5);
%let const=%sysevalf(&rep*10/&sampsz);
proc reg data=work.all noprint
        outest=work.out(drop=_model_ _type_ _depvar_ _rmse_ y _in_ _p_ _edf_) ;
        by sim type jn replicate;
        where replicate eq &rep;
        model y=dummy x xdummy/sse;
        restrict &const.*xdummy=-dummy;;
run;
quit;

proc append base=work.resultc data=work.out;
run;
%end;
%mend;
%restrict;

/******check meeting points for restricted models (unnecessary...)*****/
data work.resultc;
        set work.resultc;
        mt=-dummy/xdummy;
```

```
        uc=0;
        if mt ge replicate*10/&sampsz-.01 and mt le (replicate+1)*10/&sampsz+.01 then
meet=1;
                else meet=0;
run;

/******put together constrained and unconstrained results****/
data work.allresult;
        set work.result(where=(meet=1)) work.resultc;
run;

proc sort data=work.allresult;                          *NEW CODE BEGINS HERE 04.13.2012;
        by sim type jn descending _sse_;
run;

data work.maxsse;
        set work.allresult;
        by sim type jn;
        if first.jn;
        keep sim type jn _sse_;
        rename _sse_ = maxsse;
run;

proc sql;
        create table work.ratio as
        select allresult.*, _sse_/maxsse as ratio
        from work.allresult, work.maxsse
        where allresult.sim eq maxsse.sim and allresult.type eq maxsse.type and allresult.jn eq
maxsse.jn
        ;
quit;

data work.adjust;
        set work.ratio;
        c=0;
        s=0.75;
        if mt gt 5 then do;
                mingrp = (10-mt)/0.50;
                penalty = c /(s*sqrt(2*arcos(-1)))*exp(-1*(mt-8.5)**2/(2*s**2));
        end;
        else do;
                mingrp = mt/0.50;
                penalty = c / (s*sqrt(2*arcos(-1)))*exp(-1*(mt-2)**2/(2*s**2));
        end;
        pen_sse = ratio + penalty;
run;
```

```
/*****order results******/
proc sort data=work.adjust nodup;
        by sim type jn pen_sse; *changed to pen_sse;
run;

/*****and select best*****/
data work.best;
        set work.adjust;
        by sim type jn;
        if first.jn;
run;

proc sql; *changed all best._sse_ to best.pen_sse;
        create table work.final as
        select best.*, result0._sse_ as nullsse,
                result0._sse_-best.pen_sse as diffsse,
                0.5*(result0._sse_-best.pen_sse)/(result0._sse_/(&sampsz-4)) as ratio
        from work.best, work.result0
        where best.sim eq result0.sim and best.jn eq result0.jn         and best.type eq result0.type
        ;
quit;*NEW CODE ENDS HERE;

%mend twoline;
%twoline;

%let en=%sysfunc(time());
        data _null_;
                call symput('time',put(&en-&st,time8.2));
        run;
%put Time for twoline is: &time;

%let st=%sysfunc(time());
proc sql;
        create table work.residuals as
        select   final.sim as sim,
                        final.intercept as alt_int1,
                        final.dummy as alt_int2,
                        final.x as alt_slope1,
                        final.xdummy as alt_slope2,
                        final.mt,
                        case
                                when sim.x lt final.mt then y - (alt_int1 + alt_slope1*sim.x)
                                else  y - (alt_int1+alt_int2 + (alt_slope1+alt_slope2)*sim.x)
                        end as alt_resid
```

```
                    from work.final, work.sim
                    where final.type="Original" and final.sim eq sim.sim
                    order by sim, sim.x
        ;
quit;

proc sort data=work.residuals out=work.models nodupkey;
        by sim;
run;

%let en=%sysfunc(time());
        data _null_;
                call symput('time',put(&en-&st,time8.2));
        run;
%put Time for construction is: &time;

%let st=%sysfunc(time());

%let type=alt;
%macro boot;
proc datasets library=work;
        delete finalboot;
run;

%do s=1 %to &sims;
%let st=%sysfunc(time());
proc surveyselect data=work.residuals out=work.bootdata method=urs sampsize=&sampsz
reps=&bootsize
                noprint outhits;
        where sim=&s;
run;

proc sql;
        create table work.bootdata2 as
        select bootdata.*, ranuni(0) as rand
        from work.bootdata
        order by replicate, rand
        ;
quit;

data work.bootdata2;
        set work.bootdata2;
        by replicate;
        if first.replicate then x=0;
        x+10/&sampsz;
run;
```

```
data work.boot;
        merge work.models work.bootdata2;
        where sim=&s;
        drop numberhits rand;
        if x lt mt then
                alt_yboot = alt_int1+alt_slope1*x  + alt_resid;
                else  alt_yboot = alt_int1+alt_int2+(alt_slope1+alt_slope2)*x  + alt_resid;
        rename replicate=bootrep;
run;


/******fit one line models (and get residuals)******/
proc reg data=work.boot noprint
        outest=work.result0(drop=_model_ _type_ _depvar_ _rmse_ &type._yboot _in_ _p_
_edf_);
        by bootrep;
        model &type._yboot=x/sse;
run;
quit;

/*******get replicate data sets*******/
proc surveyselect data=work.boot out=work.all samprate=1 method=srs reps=&sampsz noprint;
        strata bootrep;
run;

/********set up dummy variables for two line modelling******/
data work.all;
        set work.all;
        if x le replicate*10/&sampsz then do;
                dummy=0; xdummy=0;
                end;
                else do;
                dummy=1; xdummy=x;
                end;
run;

/*******fit two line, unconstrained******/
proc reg data=work.all noprint
        outest=work.result(drop=_model_ _type_ _depvar_ _rmse_ &type._yboot _in_ _p_
_edf_);
        by bootrep replicate;
        where replicate between 5 and %eval(&sampsz-5);
        model &type._yboot=dummy x xdummy/sse;
run;
quit;
```

```
%let en=%sysfunc(time());
        data _null_;
                call symput('time',put(&en-&st,time8.2));
        run;
%put Time for unconstrained fits is: &time;



/*********check for meeting points********/
data work.result;
        set work.result;
        mt=-dummy/xdummy;
        if mt ge replicate*10/&sampsz-.01 and mt le (replicate+1)*10/&sampsz+.01 then do;
meet=1; uc=1; end;
                else do; meet=0; uc=0; end;
run;

/********clear constrained result data set (for appending)******/
proc datasets nolist nowarn library=work;
        delete resultc;
run;


/*******fit restriced models, compile results*****/
%macro restrict;
%do rep=5 %to %eval(&sampsz-5);
%let const=%sysevalf(&rep*10/&sampsz);
proc reg data=work.all noprint
        outest=work.out(drop=_model_ _type_ _depvar_ _rmse_ &type._yboot _in_ _p_ _edf_) ;
        by bootrep replicate;
        where replicate eq &rep;
        model &type._yboot=dummy x xdummy/sse;
        restrict &const.*xdummy=-dummy;;
run;
quit;

proc append base=work.resultc data=work.out;
run;
%end;
%mend;
%restrict;

%let en=%sysfunc(time());
        data _null_;
                call symput('time',put(&en-&st,time8.2));
        run;
```

```
%put Time for constrained fits is: &time;


/******check meeting points for restricted models (unnecessary...)*****/
data work.resultc;
	set work.resultc;
	mt=-dummy/xdummy;
	uc=0;
	if mt ge replicate*10/&sampsz-.01 and mt le (replicate+1)*10/&sampsz+.01 then
meet=1;
		else meet=0;
run;

/******put together constrained and unconstrained results****/
data work.allresult;
	set work.result(where=(meet=1)) work.resultc;
run;

proc sort data=work.allresult;				*NEW CODE BEGINS HERE 04.13.2012;
	by bootrep descending _sse_;
run;

data work.maxsse;
	set work.allresult;
	by bootrep;
	if first.bootrep;
	keep bootrep _sse_;
	rename _sse_ = maxsse;
run;

proc sql;
	create table work.ratio as
	select allresult.*, _sse_/maxsse as ratio
	from work.allresult, work.maxsse
	where allresult.bootrep eq maxsse.bootrep
	;
quit;

data work.adjust;
	set work.ratio;
	c=0;
	s=0.75;
	if mt gt 5 then do;
		mingrp = (10-mt)/0.50;
		penalty = c /(s*sqrt(2*arcos(-1)))*exp(-1*(mt-8.5)**2/(2*s**2));
	end;
```

```
        else do;
                mingrp = mt/0.50;
                penalty = c / (s*sqrt(2*arcos(-1)))*exp(-1*(mt-2)**2/(2*s**2));
        end;
        pen_sse = ratio + penalty;
run;

/*****order results******/
proc sort data=work.adjust nodup;
        by bootrep pen_sse;
run;                                                    *NEW CODE ENDS HERE;

/*****and select best*****/
data work.best;
        set work.adjust;
        by bootrep;
        sim=&s;
        if first.bootrep;
run;

/*proc sql;*/
/*      create table work.fin as*/
/*      select best.*, result0._sse_ as nullsse, */
/*              result0._sse_-best._sse_ as diffsse,*/
/*              0.5*(result0._sse_-best._sse_)/(result0._sse_/(&sampsz-4)) as ratio*/
/*      from work.best, work.result0*/
/*      where best.bootrep eq result0.bootrep*/
/*      ;*/
/*quit;*/

proc append base=work.finalboot data=work.best;
run;

%let en=%sysfunc(time());
        data _null_;
                call symput('time',put(&en-&st,time8.2));
        run;
%put Time for sim &s is:  &time;
%end;
%mend;

%boot;

ods listing;
options notes source;
proc sql;
```

```
        create table jnavg as
        select sim, sum(mt)/&sampsz as jnavg
        from final
        where type eq 'Jacknife'
        group by sim
        ;
        create table jn as
        select jnavg.*, final.*
        from final, jnavg
        where type eq 'Jacknife' and final.sim eq jnavg.sim
        ;
        create table accel as
        select sim, sum((jnavg-mt)**3)/(6*sum((jnavg-mt)**2)**(3/2)) as accel
        from jn
        group by sim
        ;
        create table boots as
        select finalboot.mt as bootmt, final.*, jnavg
        from final, finalboot, jnavg
        where finalboot.sim eq final.sim eq jnavg.sim and type ne 'Jacknife'
        order by sim, bootmt
        ;
        create table biascorr as
        select sim, probit(sum(bootmt lt mt)/&bootsize) as biascorr,
                probit(sum(bootmt lt jnavg)/&bootsize) as biascorr_jk,
                probit((sum(bootmt lt mt) + 0.5*sum(bootmt eq mt))/&bootsize) as biascorr_cc,
                probit((sum(bootmt lt jnavg) + 0.5*sum(bootmt eq jnavg))/&bootsize) as
biascorr_jkcc
        from boots
        group by sim
        ;

quit;

data work.coverage_&slopename._&cpname._&sampsz._&sims;
        merge accel biascorr boots;
        by sim;
        retain alpha1 alpha2 BCaLo BCaHi PctLo PctHi BCaLow BCaHigh PctLow PctHigh
                        alpha1a alpha2a aLo aHi aLow aHigh
                        alpha1bc alpha2bc BCLo BCHi BCLow BCHigh
                        alpha1bcajk alpha2bcajk BCajkLo BCajkHi BCajkLow BCajkHigh
                        alpha1bcacc alpha2bcacc BCaccLo BCaccHi BCaccLow BCaccHigh
                        alpha1bcajkcc alpha2bcajkcc BCajkccLo BCajkccHi BCajkccLow
BCajkccHigh;

        alpha=(1-(&conf/100))/2;
```

```
        if first.sim then do;
                posit=0;
                /*Compute the adjusted percentiles */
                alpha1 = cdf("normal", biascorr+(biascorr + probit(alpha))/(1-
accel*(biascorr+probit(alpha))));
                alpha2 = cdf("normal", biascorr+(biascorr + probit(1-alpha))/(1-
accel*(biascorr+probit(1-alpha))));
                *Get BCa conservative intervals;
                BCaLo = max(floor(alpha1*&bootsize),1);
                BCaHi = ceil(alpha2*&bootsize);
                /*Compute accleration only adjusted percentiles */
                alpha1a = cdf("normal", probit(alpha)/(1-accel*(probit(alpha))));
                alpha2a = cdf("normal",probit(1-alpha)/(1-accel*(probit(1-alpha))));
                *Get a conservative intervals;
                aLo = max(floor(alpha1a*&bootsize),1);
                aHi = ceil(alpha2a*&bootsize);
                /*Compute BC only adjusted percentiles */
                alpha1bc = cdf("normal", biascorr+(biascorr + probit(alpha)));
                alpha2bc = cdf("normal", biascorr+(biascorr + probit(1-alpha)));
                *Get a conservative intervals;
                BCLo = max(floor(alpha1bc*&bootsize),1);
                BCHi = ceil(alpha2bc*&bootsize);
                *Get Percentile conservative intervals;
                PctLo = max(floor(alpha*&bootsize),1);
                PctHi = ceil((1-alpha)*&bootsize);
        /*Compute the adjusted percentiles */
                alpha1bcajk = cdf("normal", biascorr_jk+(biascorr_jk + probit(alpha))/(1-
accel*(biascorr_jk+probit(alpha))));
                alpha2bcajk = cdf("normal", biascorr_jk+(biascorr_jk + probit(1-alpha))/(1-
accel*(biascorr_jk+probit(1-alpha))));
                *Get BCa conservative intervals;
                BCajkLo = max(floor(alpha1bcajk*&bootsize),1);
                BCajkHi = ceil(alpha2bcajk*&bootsize);
        /*Compute the adjusted percentiles */
                alpha1bcacc = cdf("normal", biascorr_cc+(biascorr_cc + probit(alpha))/(1-
accel*(biascorr_cc+probit(alpha))));
                alpha2bcacc = cdf("normal", biascorr_cc+(biascorr_cc + probit(1-alpha))/(1-
accel*(biascorr_cc+probit(1-alpha))));
                *Get BCa conservative intervals;
                BCaccLo = max(floor(alpha1bcacc*&bootsize),1);
                BCaccHi = ceil(alpha2bcacc*&bootsize);
        /*Compute the adjusted percentiles*/
                alpha1bcajkcc = cdf("normal", biascorr_jkcc+(biascorr_jkcc + probit(alpha))/(1-
accel*(biascorr_jkcc+probit(alpha))));
```

```
                alpha2bcajkcc = cdf("normal", biascorr_jkcc+(biascorr_jkcc + probit(1-
alpha))/(1-accel*(biascorr_jkcc+probit(1-alpha))));
                *Get BCa conservative intervals;
                BCajkccLo = max(floor(alpha1bcajkcc*&bootsize),1);
                BCajkccHi = ceil(alpha2bcajkcc*&bootsize);
        end;
        posit+1;
        if posit eq BCaLo then BCaLow=bootmt;
        if posit eq PctLo then PctLow=bootmt;
        if posit eq aLo then aLow=bootmt;
        if posit eq BCLo then BCLow=bootmt;
        if posit eq BCaHi then BCaHigh=bootmt;
        if posit eq PctHi then PctHigh=bootmt;
        if posit eq aHi then aHigh=bootmt;
        if posit eq BCHi then BCHigh=bootmt;

        if posit eq BCajkLo then BCajkLow=bootmt;
        if posit eq BCajkHi then BCajkHigh=bootmt;
        if posit eq BCaccLo then BCaccLow=bootmt;
        if posit eq BCaccHi then BCaccHigh=bootmt;
        if posit eq BCajkccLo then BCajkccLow=bootmt;
        if posit eq BCajkccHi then BCajkccHigh=bootmt;

        if last.sim then do;
                cp=&cp;
                if BCaLow le cp le BCaHigh then BCaCov=1; else BCaCov=0;
                if aLow le cp le aHigh then aCov=1; else aCov=0;
                if BCLow le cp le BCHigh then BCCov=1; else BCCov=0;
                if PctLow le cp le PctHigh then PctCov=1; else PctCov=0;

                if BCajkLow le cp le BCajkHigh then BCajkCov=1; else BCajkCov=0;
                if BCaccLow le cp le BCaccHigh then BCaccCov=1; else BCaccCov=0;
                if BCajkccLow le cp le BCajkccHigh then BCajkccCov=1; else BCajkccCov=0;
                BCaLength=BCaHigh-BCaLow;
                aLength=aHigh-aLow;
                BCLength=BCHigh-BCLow;
                PctLength=PctHigh-PctLow;

                BCajkLength=BCajkHigh-BCajkLow;
                BCaccLength=BCaccHigh-BCaccLow;
                BCajkccLength=BCajkccHigh-BCajkccLow;
                output;
        end;
        *keep sim cp mt alpha BCaLow BCaHigh PctLow PctHigh BCaCov PctCov BCaLength
PctLength;
run;
```

```
%let en=%sysfunc(time());
        data _null_;
                call symput('time',put(&en-&start,time8.2));
        run;
%put &time;
%put &sampsz;

proc means data=work.coverage_&slopename._&cpname._&sampsz._&sims;
        var bcacov -- BCajkccLength;
run;

proc print data=work.coverage_&slopename._&cpname._&sampsz._&sims;
        var bcacclo bcacchi bcacclow bcacchigh biascorr_cc bcacccov alpha1bcacc alpha2bcacc;
run;

quit;
```

Appendix B

GRAPHS NOT DISCUSSED IN ANALYSIS SECTION

Here I will show all graphs related to $n = 40$ with $\tau = 3$ and $\tau = 5$. Also $\tau = 4$ with $n = 20$ and $n = 60$.

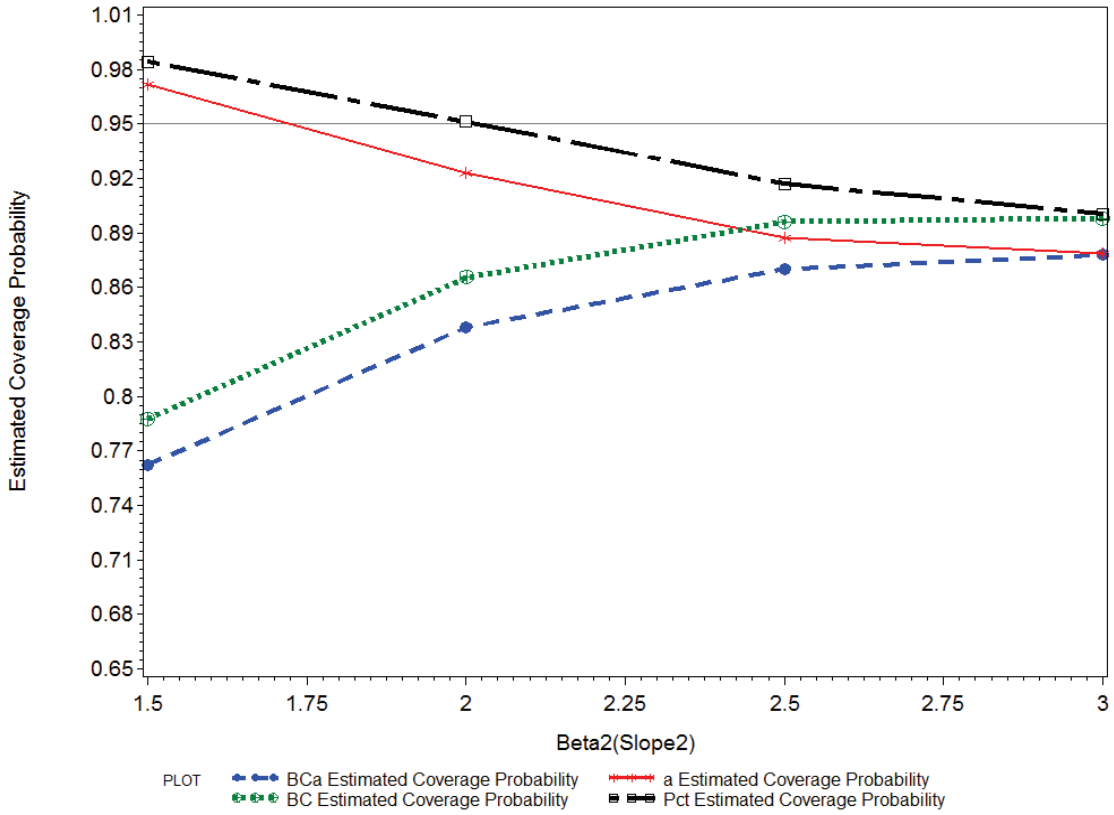

Figure B.1: Estimated coverage probabilities of *BCa, a, BC*, and *Pct* for n=20 and τ=4

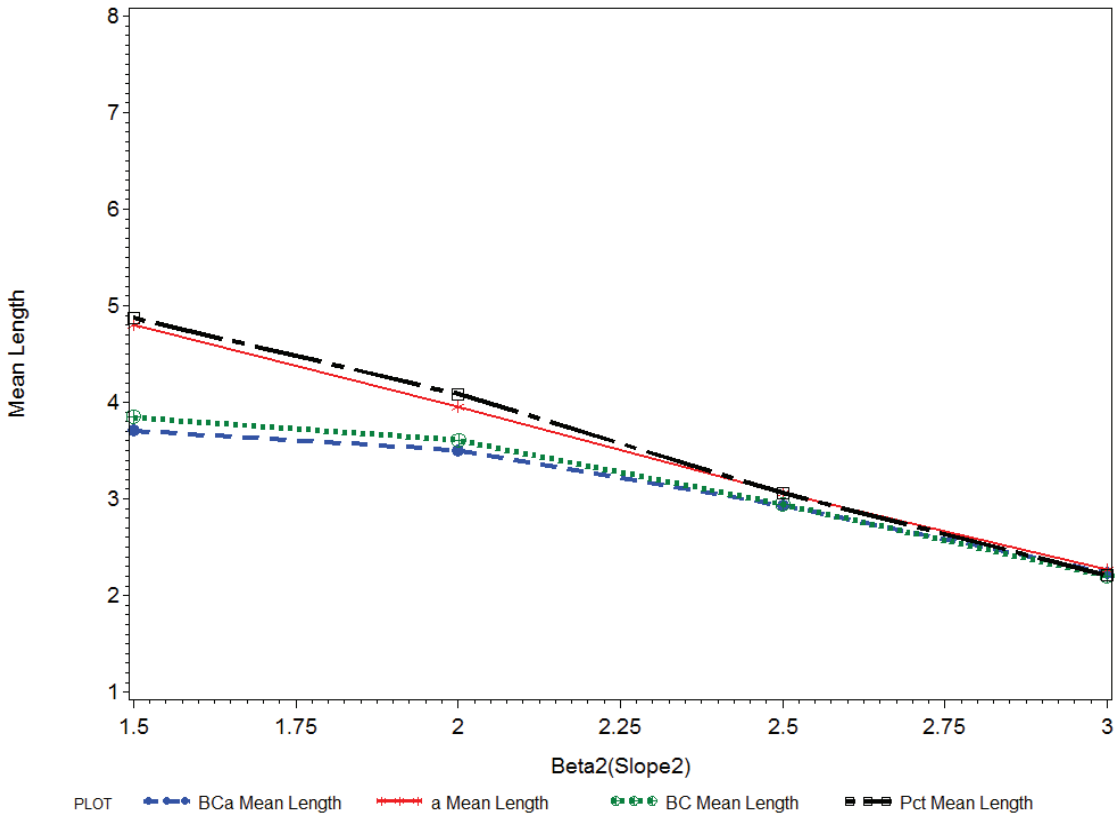

Figure B.2: Mean length of *BCa, a, BC*, and *Pct* for n=20 and τ=4

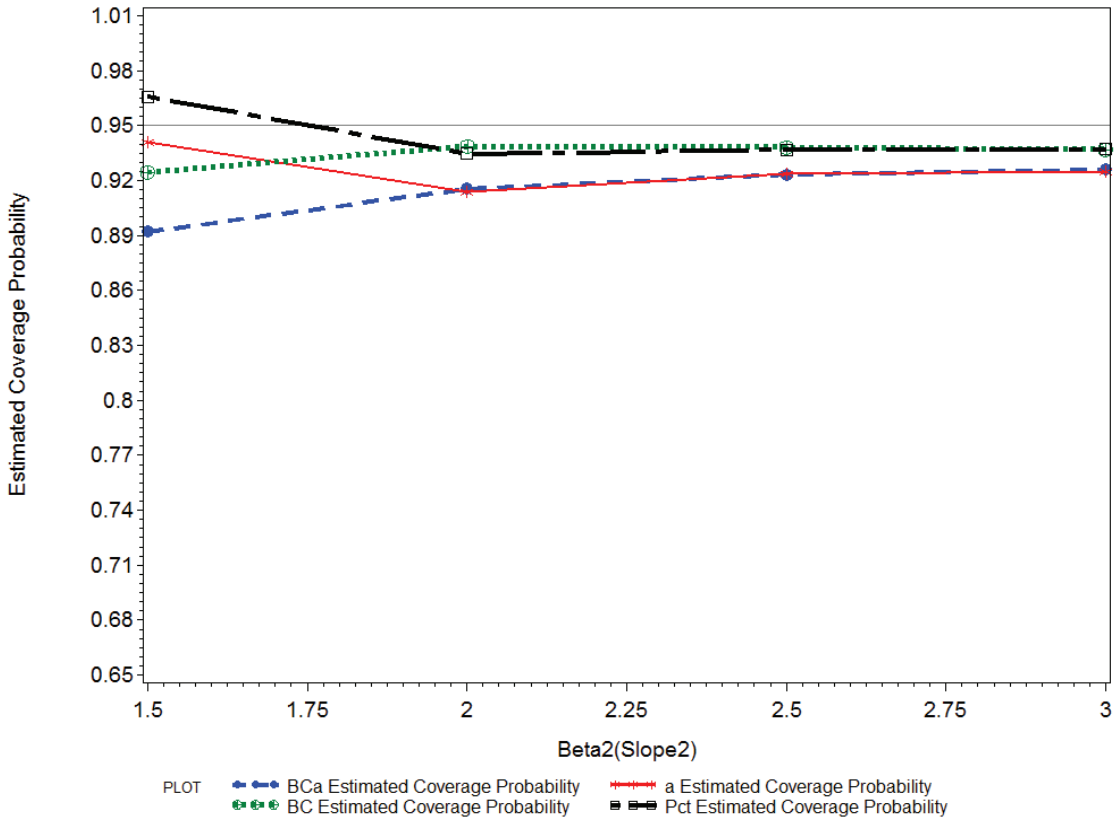

Figure B.3: Estimated coverage probabilities of *BCa, a, BC*, and *Pct* for n=60 and τ=4

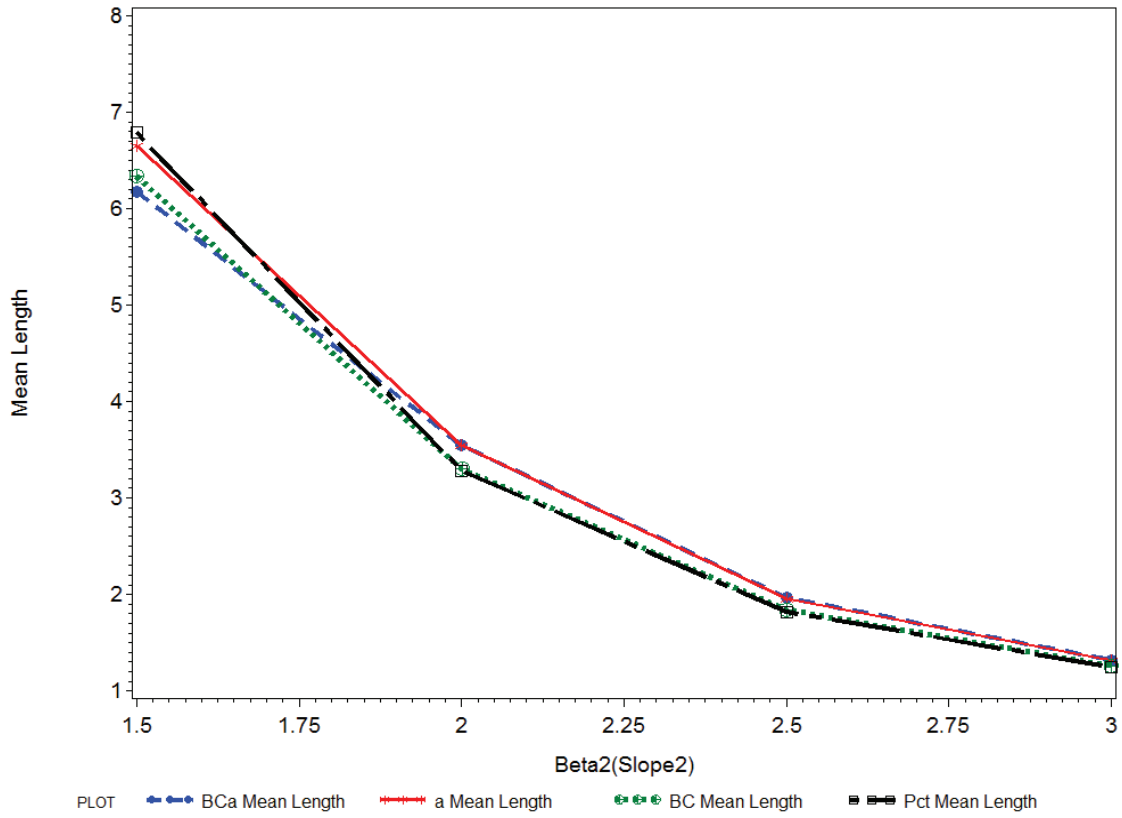

Figure B.4: Mean length of *BCa, a, BC*, and *Pct* for n=60 and τ=4

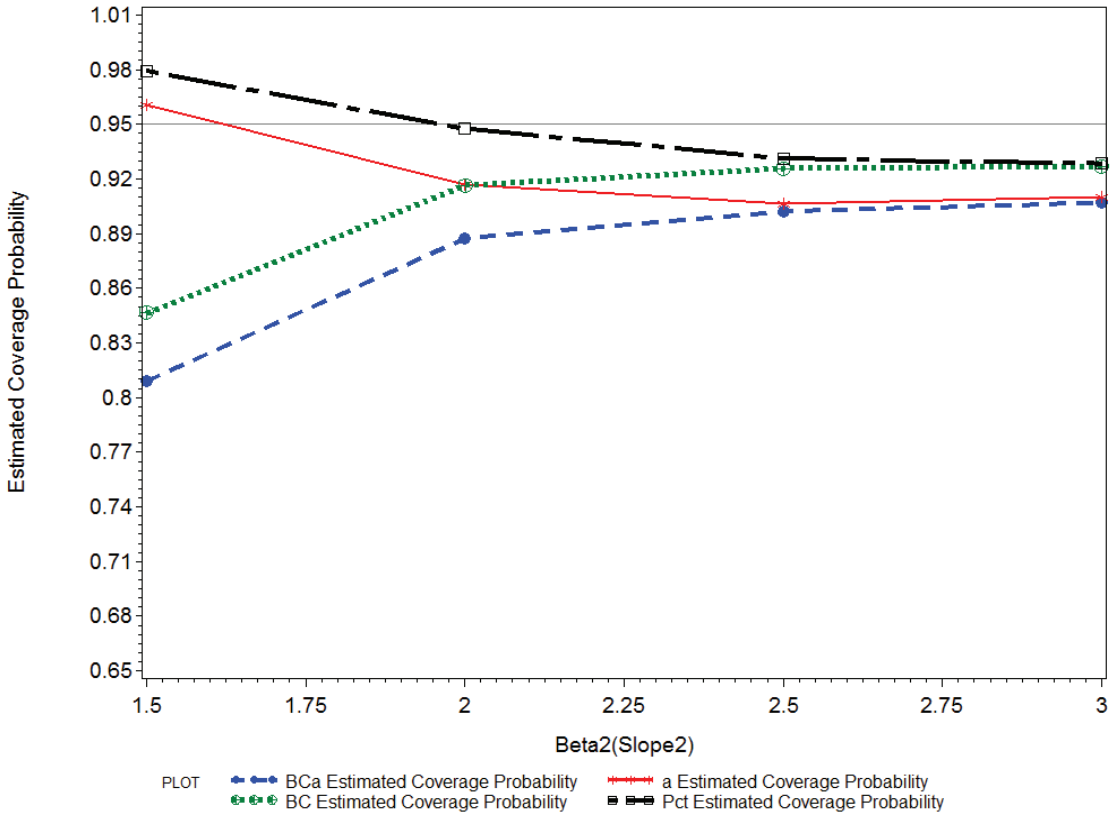

Figure B.5: Estimated coverage probabilities of *BCa, a, BC*, and *Pct* for n=40 and τ=3

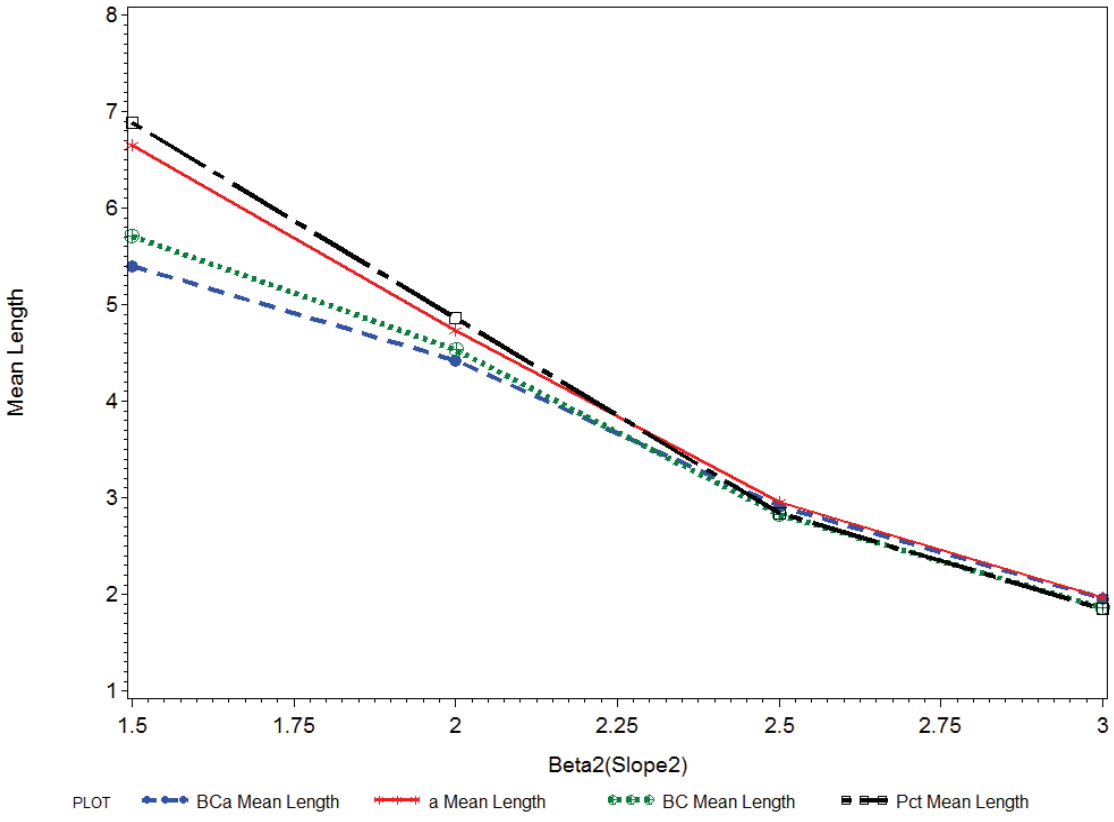

Figure B.6: Mean length of *BCa, a, BC*, and *Pct* for n=40 and τ=3

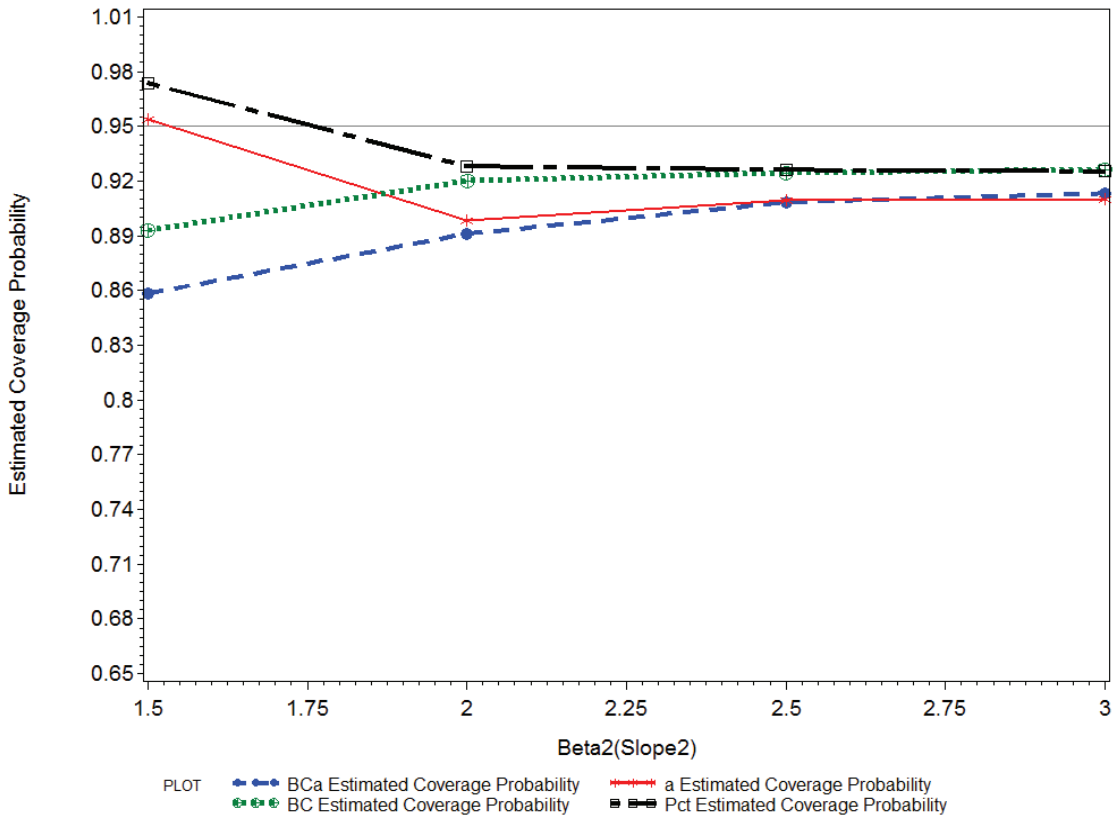

Figure B.7: Estimated coverage probabilities of *BCa, a, BC*, and *Pct* for n=40 and τ=4

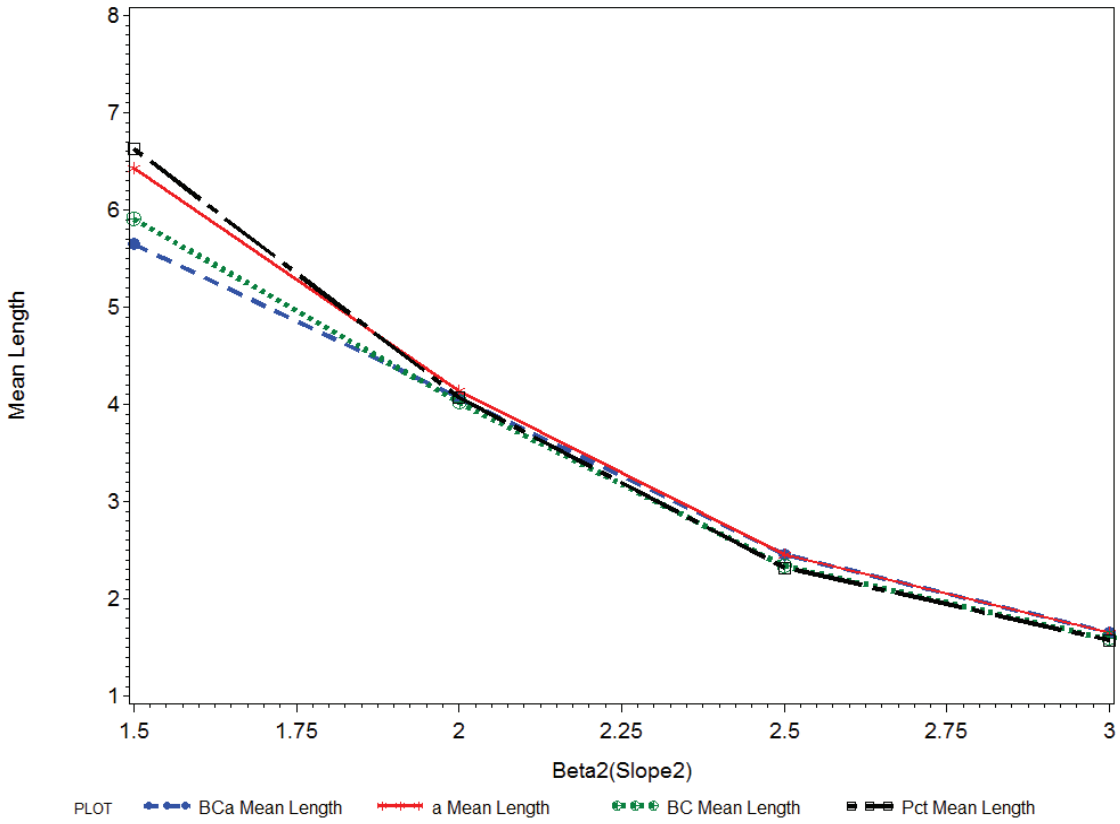

Figure B.8: Mean length of *BCa, a, BC*, and *Pct* for n=40 and τ=4

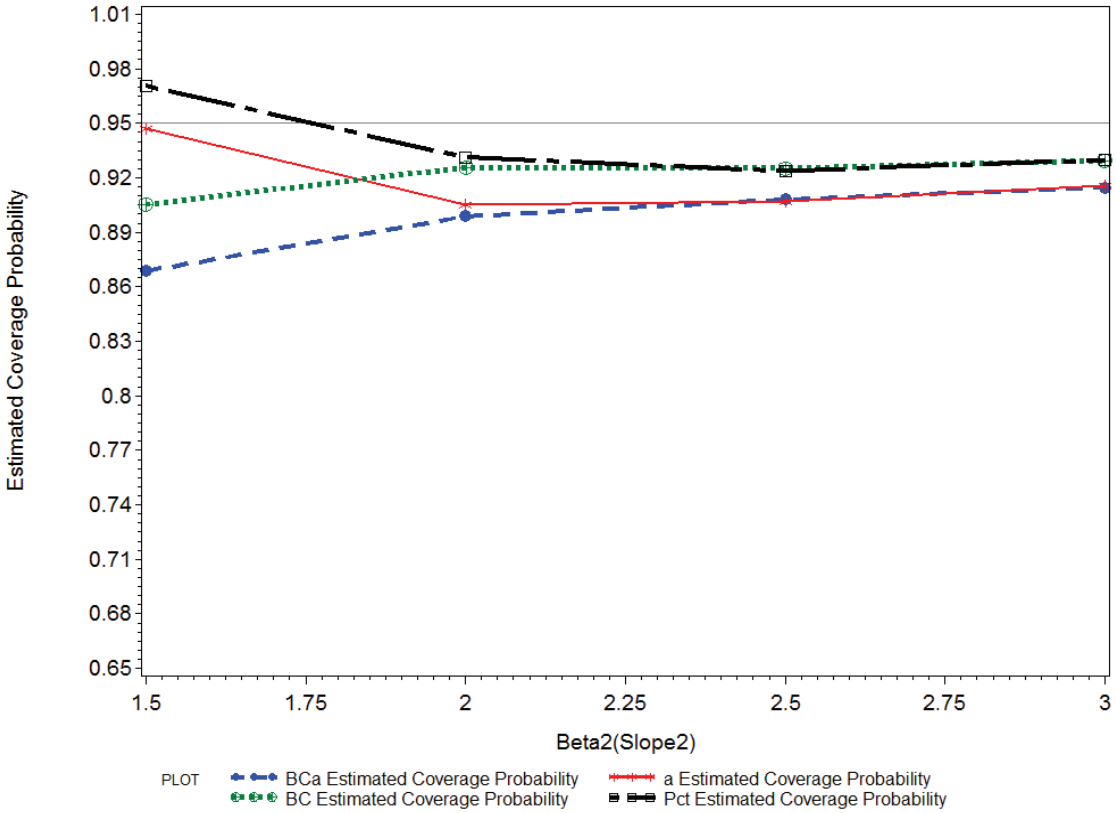

Figure B.9: Estimated coverage probabilities of *BCa, a, BC*, and *Pct* for n=40 and τ=5

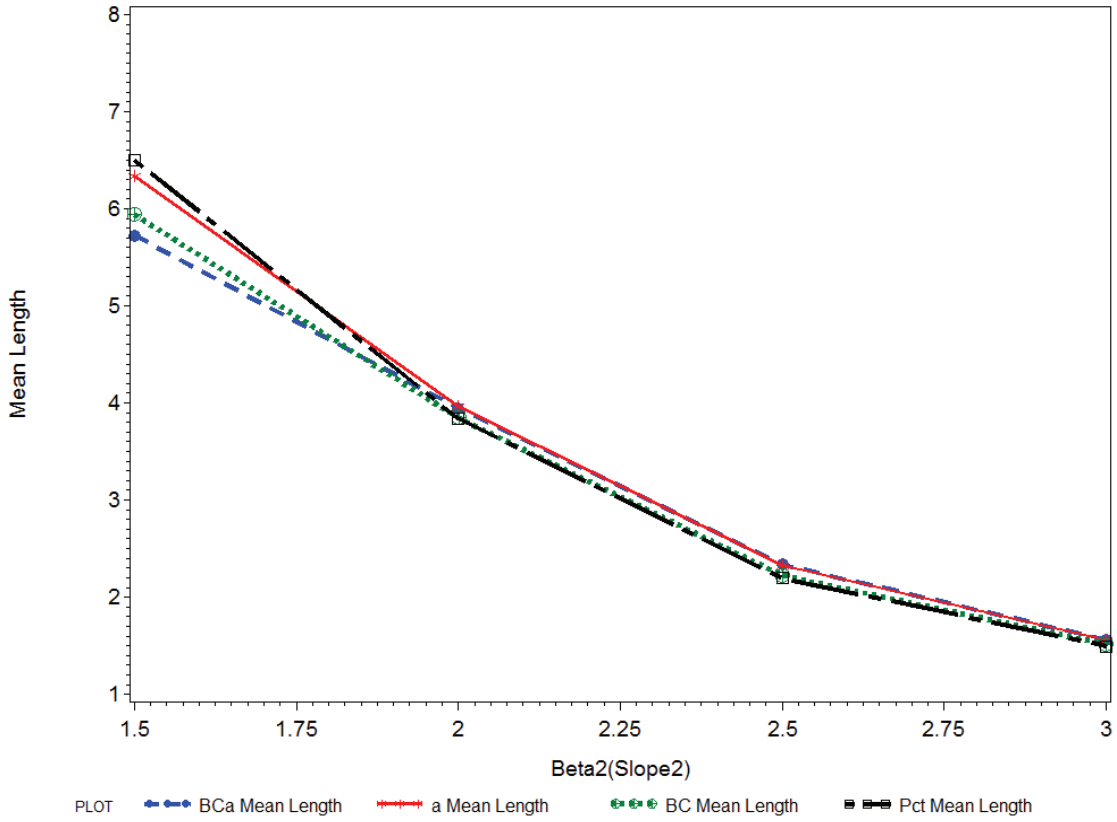

Figure B.10: Mean length of *BCa, a, BC*, and *Pct* for n=40 and τ=5

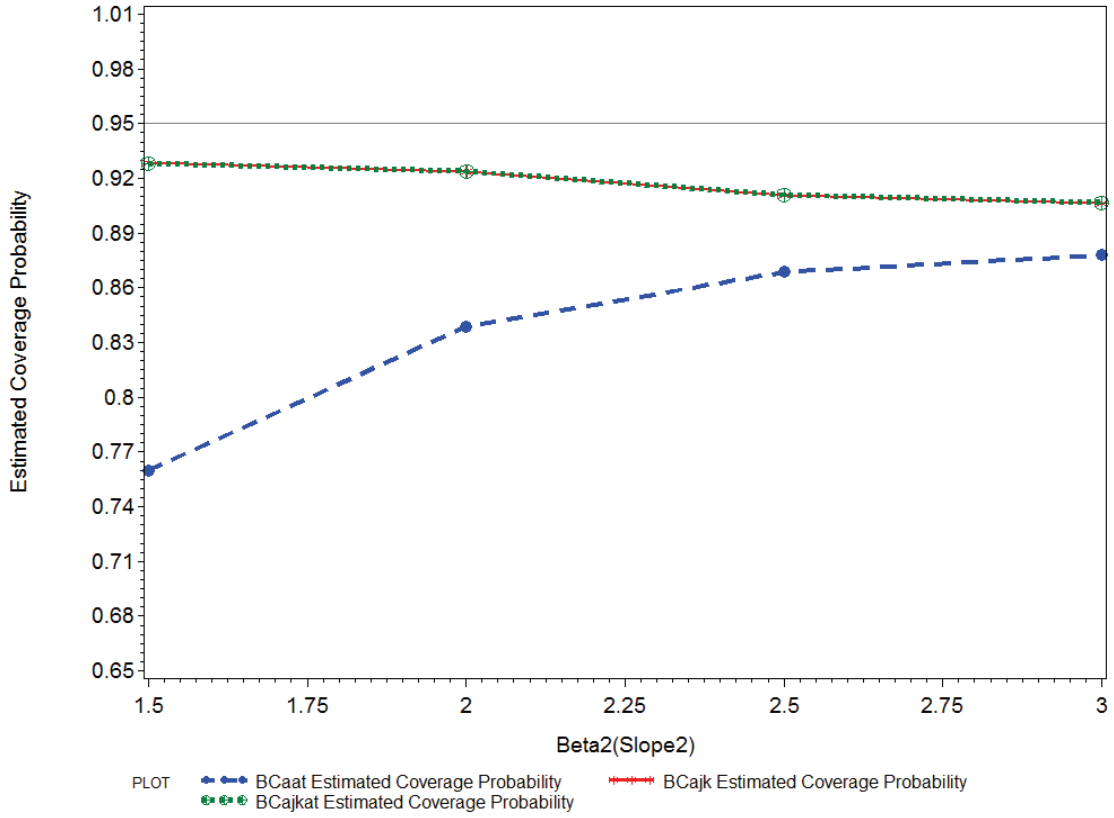

Figure B.11: Estimated coverage probabilities of *BCaat, BCajk*, and *BCajkat* for n=20 and τ=4

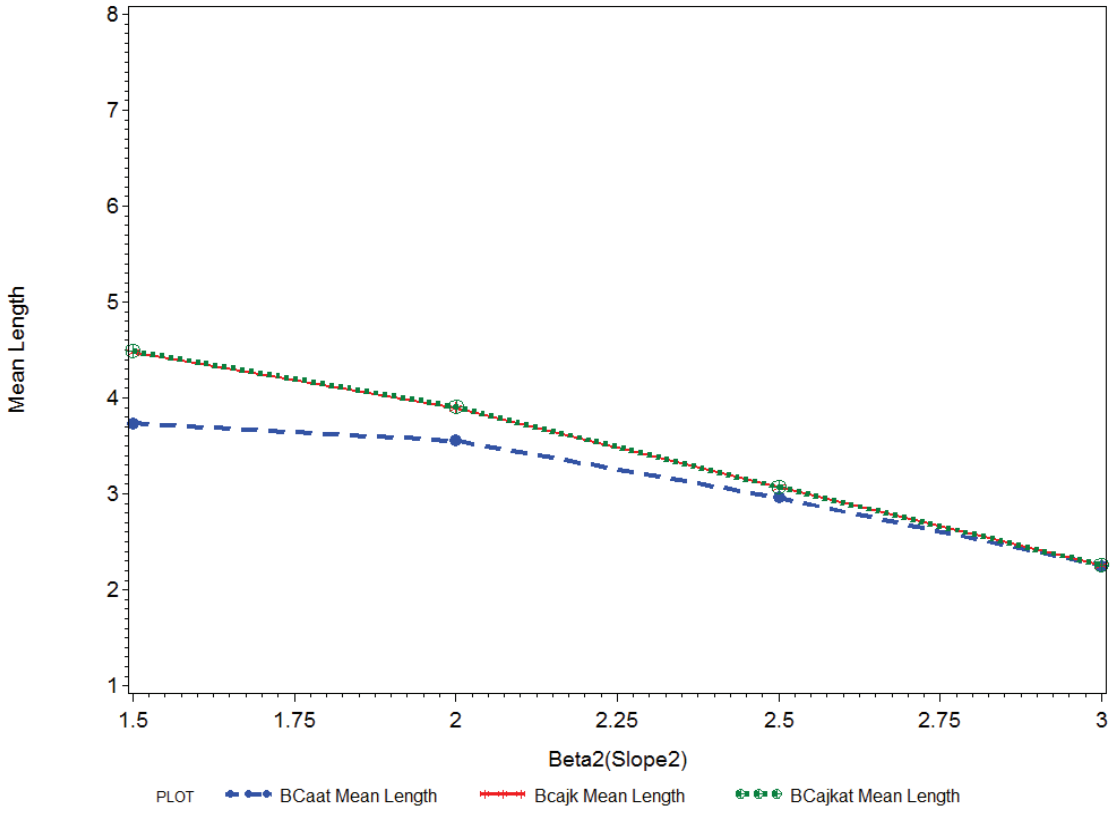

Figure B.12: Mean length of *BCaat, BCajk*, and *BCajkat* for n=20 and τ=4

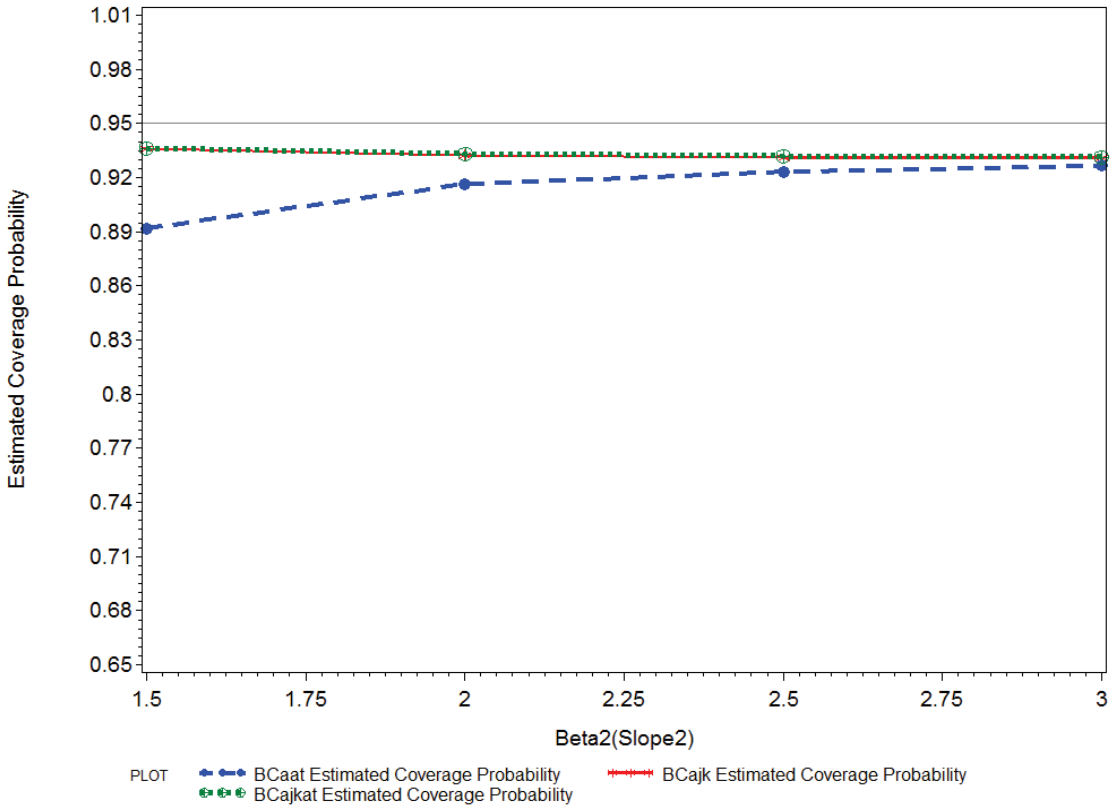

Figure B.13: Estimated coverage probabilities of *BCaat, BCajk*, and *BCajkat* for n=60 and τ=4

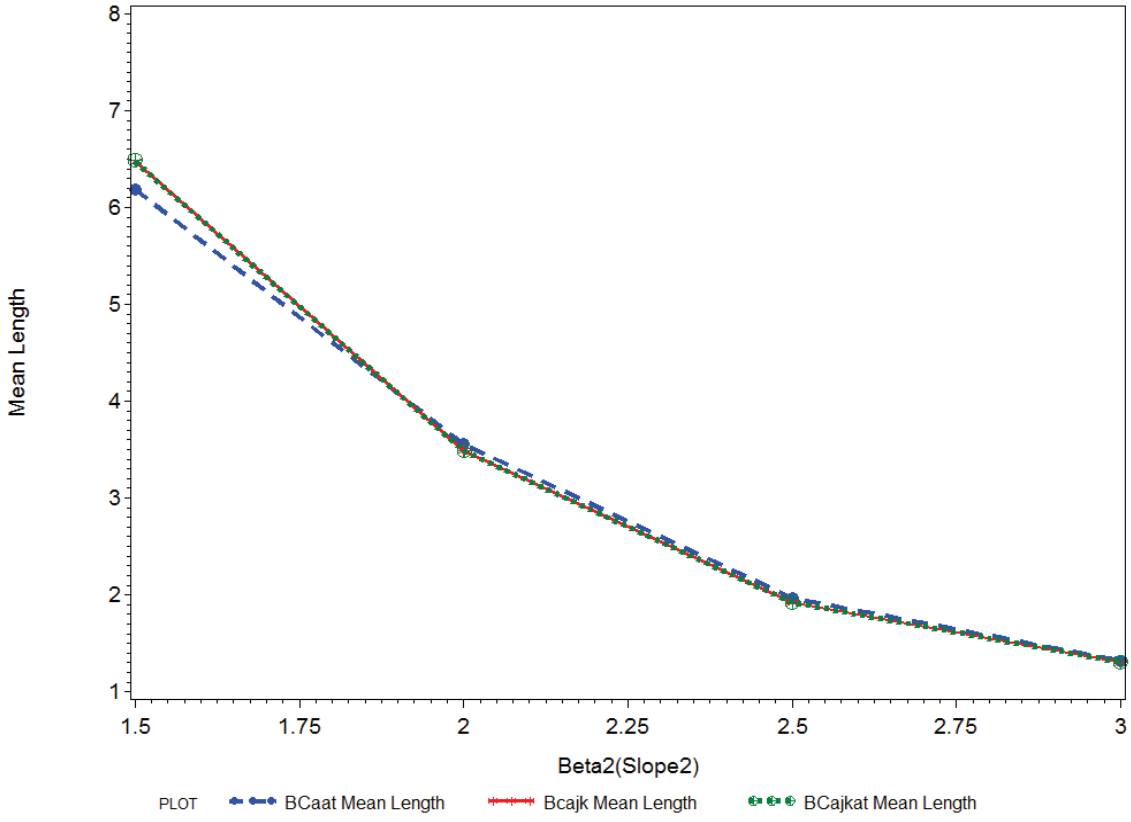

Figure B.14: Mean length of *BCaat, BCajk*, and *BCajkat* for n=60 and τ=4

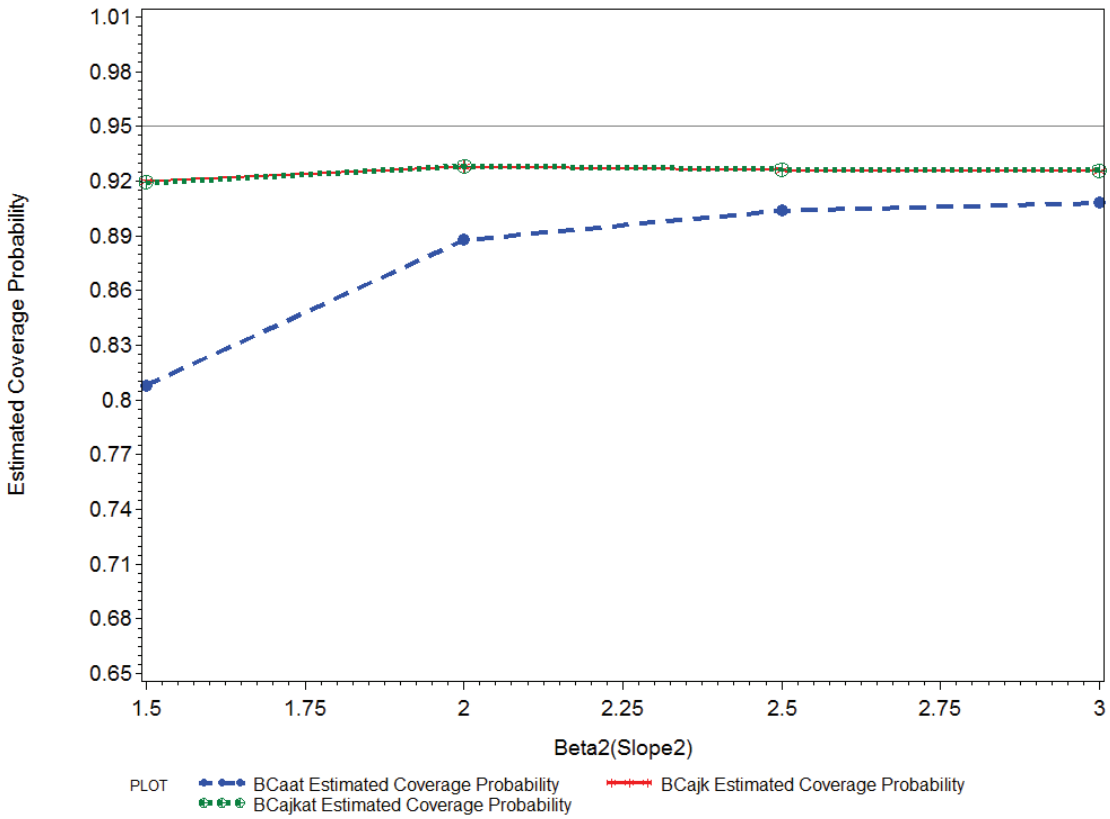

Figure B.15: Estimated coverage probabilities of *BCaat, BCajk*, and *BCajkat* for n=40 and τ=3

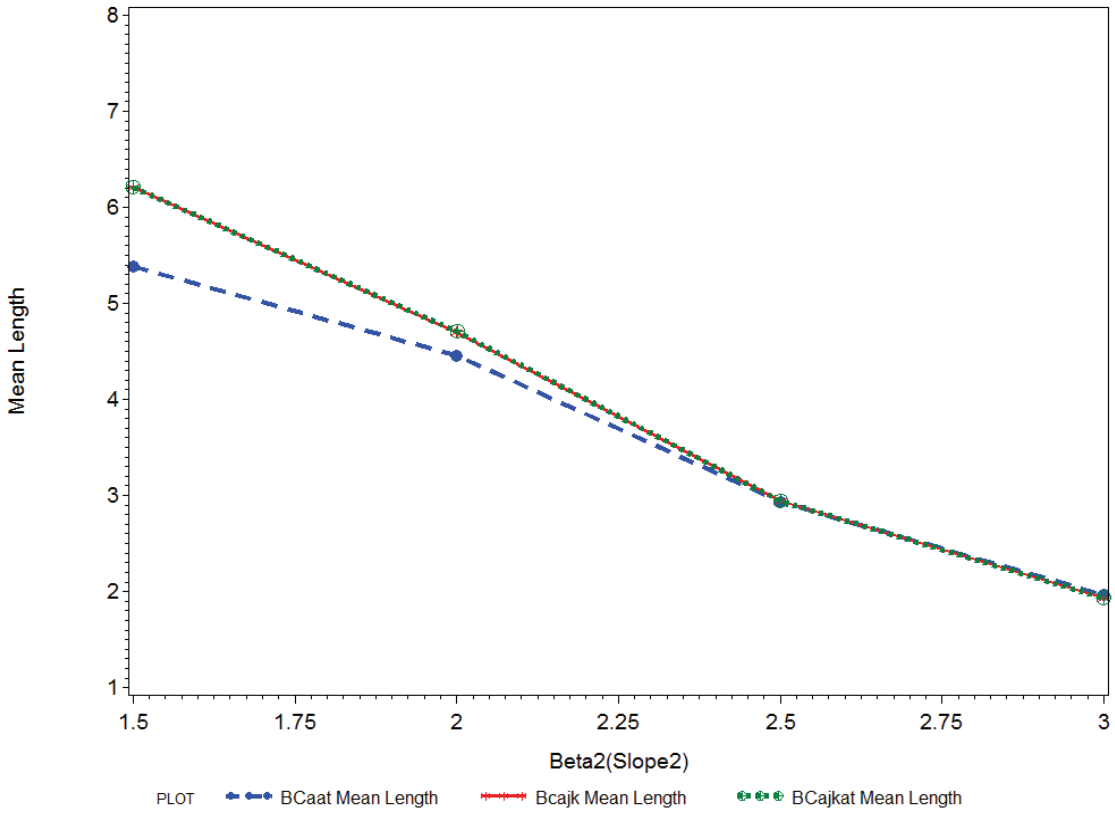

Figure B.16: Mean length of *BCaat, BCajk*, and *BCajkat* for n=40 and τ=3

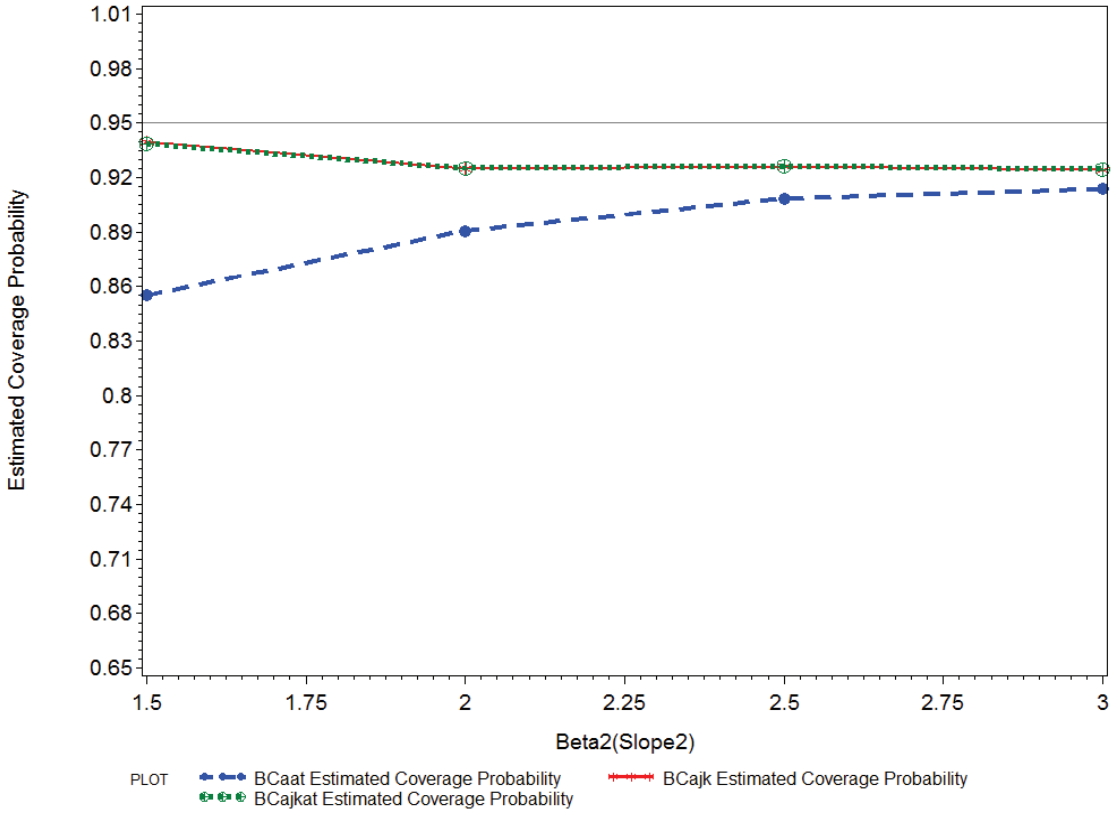

Figure B.17: Estimated coverage probabilities of *BCaat, BCajk*, and *BCajkat* for n=40 and τ=4

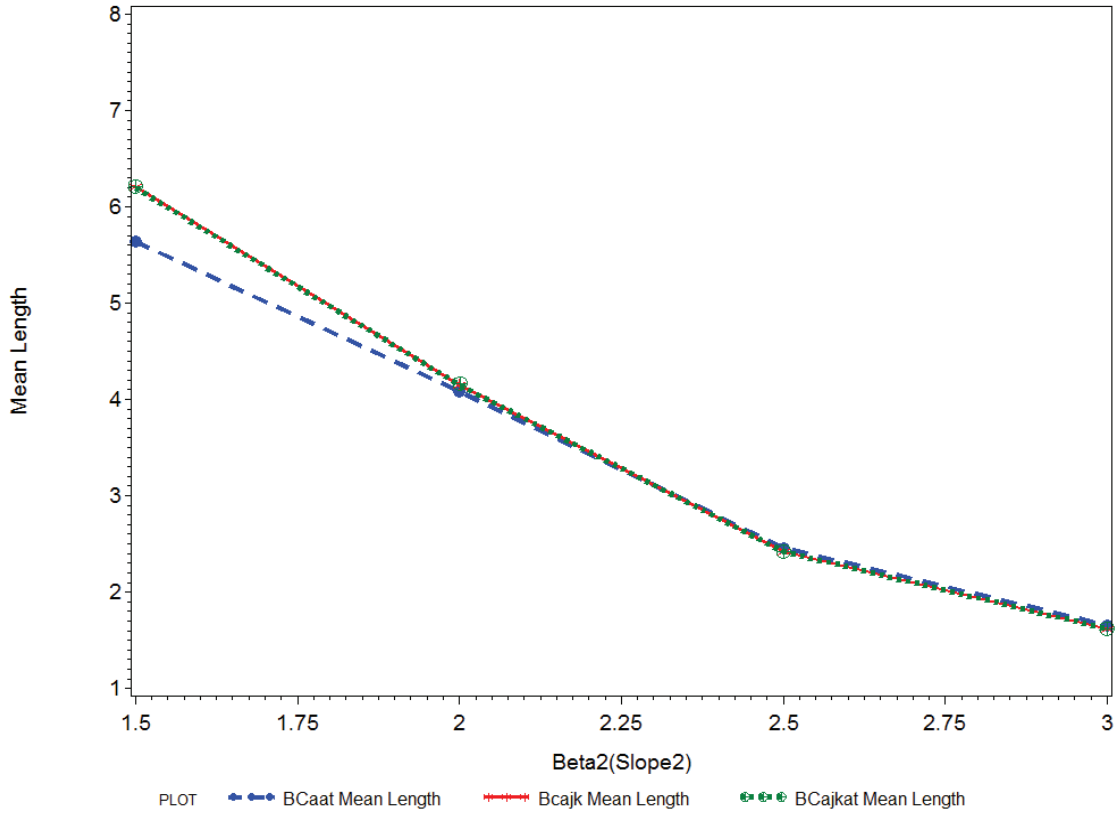

Figure B.18: Mean length of *BCaat, BCajk*, and *BCajkat* for n=40 and τ=4

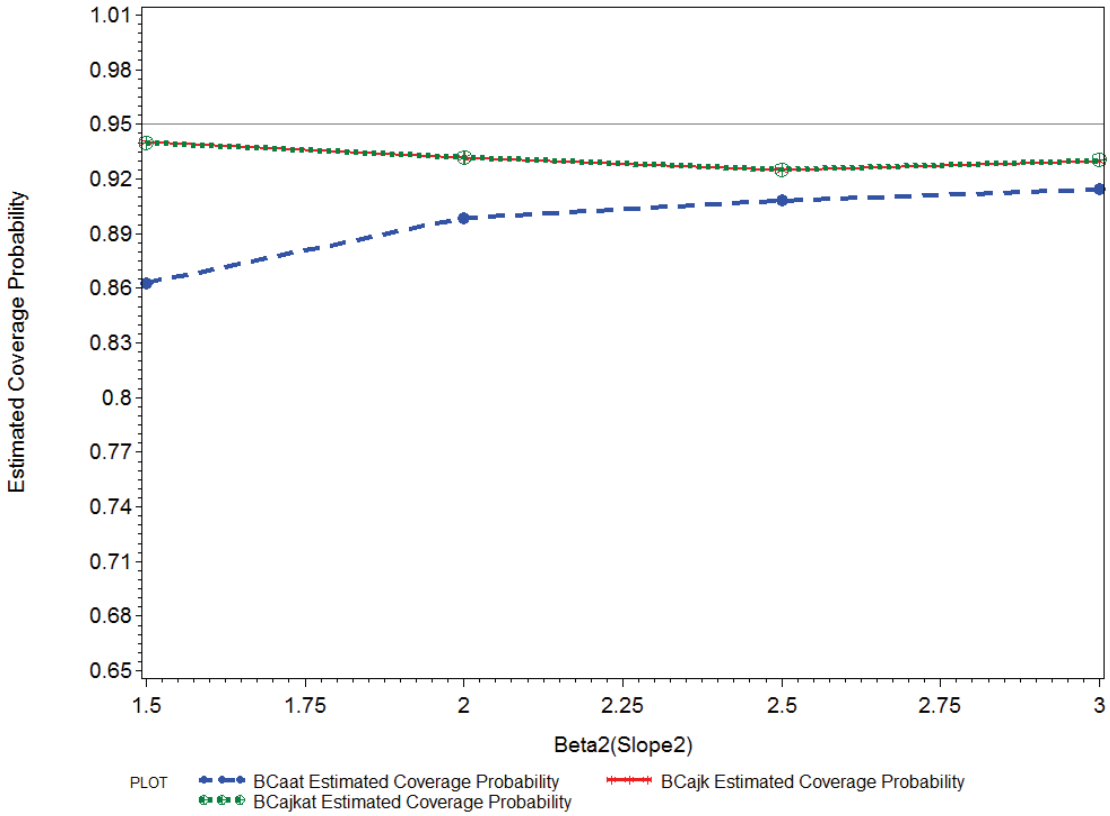

Figure B.19: Estimated coverage probabilities of *BCaat, BCajk*, and *BCajkat* for n=40 and τ=5

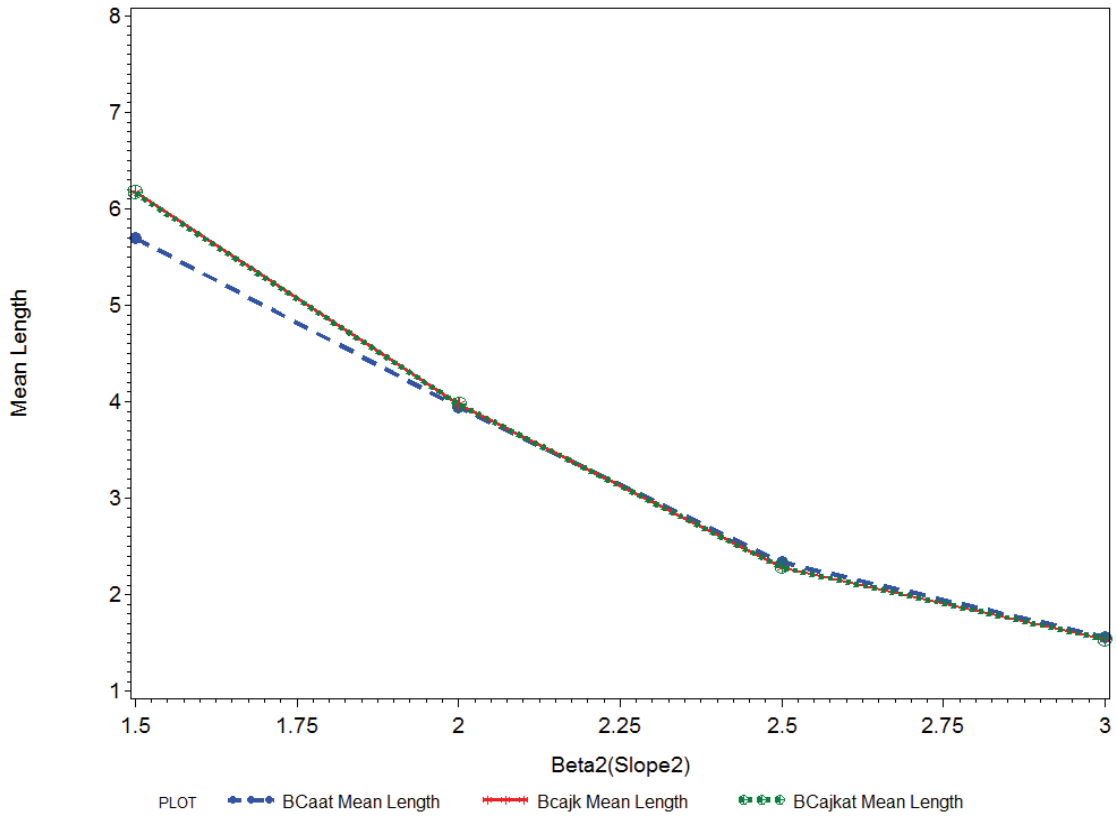

Figure B.20: Mean length of *BCaat, BCajk*, and *BCajkat* for n=40 and τ=5