8-1-2013

Bayesian estimation of the fractal dimension index of fractional Brownian motion

Chen-Yueh Chen

UNIVERSITY OF NORTHERN COLORADO

Greeley, CO

The Graduate School

THE BAYESIAN ESTIMATION OF THE FRACTAL DIMENSION INDEX OF FRACTIONAL BROWNIAN MOTION

A Dissertation Submitted in Partial Fulfillment
of the Requirements of the Degree of
Doctor of Philosophy

Chen-Yueh Chen

College of Education and Behavioral Sciences
School of Applied Statistics and Research Methods

August 2013

This Dissertation by: Chen-Yueh Chen

Entitled: *The Bayesian Estimation of the Fractal Dimension Index of Fractional Brownian Motion*

has been approved as meeting the requirement for the Degree of Doctor of Philosophy in College of Education and Behavioral Sciences in School of Applied Statistics and Research Methods, Program of Applied Statistics and Research Methods

Accepted by the Doctoral Committee

_____
Khalil Shafie, Ph.D., Research Advisor

_____
Jay Schaffer, Ph.D., Committee Member

_____
Daniel Mundfrom, Ph.D., Committee Member

_____
Robert Heiny, Ph.D. Faculty Representative

Date of Dissertation Defense _____.

Accepted by the Graduate School

_____
Linda L. Black, Ed.D., LPC
Acting Dean of the Graduate School and International Admissions

ABSTRACT

Chen, Chen-Yueh. *The Bayesian Estimation of the Fractal Dimension Index of Fractional Brownian Motion.* Published Doctor of Philosophy dissertation, University of Northern Colorado, 2013.

The primary purpose of this study was to find Bayesian estimates for the Hurst dimension of a fBm with a Beta prior when the process is observed at both discrete and continuous times. Additionally, this study sought to examine how sensitive is the Bayesian analysis with Beta prior to the choice of parameters of Beta prior. Finally, this study attempted to develop R codes for the research questions.

Using Metropolis-Hastings algorithm of MCMC as well as the assumed proposal distribution of Beta distribution, the Bayesian estimate for the Hurst dimension of a fBm with a Beta prior when the process is observed at discrete times was obtained. For the continuous case, however, the probability measures generated by two different Hurst dimension processes are singular with respect to each other, so it follows that there is no likelihood function for the continuous case.

Overall, the estimated H appears to be greater than the real H. Overestimation is observed though the overestimation is less severe as real H goes up. In addition, the estimated H decreases as Beta parameters go up given an Alpha value. In contrast, the estimated H increases as Alpha parameters go up given a Beta value. For the real-world data, the 2011 daily Taiwan Stock Index was used and the estimated Hurst index was 0.21. Finally, the R codes were successfully developed to implement the simulation in this study using a variety of packages such as “dvfBm,” “mnormt,” and “mcmcse.”

ACKNOWLEDGEMENTS

I would like to thank my research advisor, Dr. Khalil Shafie, for supporting me throughout my Ph.D. study. I appreciate your encouragement, inspiration, and flexibility for everything regarding my academic study. I also would like to thank Dr. Schaffer, Dr. Mundfrom, and Dr. Heiny for your support and guidance on my dissertation. Furthermore, I would like to thank Keyleigh and Dr. Hutchinson for your unlimited support.

I would like to thank many friends in Greeley. I want to thank my friend family, Alan and Cathy, for your support and friendship. I can not accomplish my study without Yi-Hsiu and my family's unlimited support and encouragement. I appreciate your support when I felt discouraged. Finally, I dedicate my dissertation to Yi-Hsiu and my family.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

Figure

Figure

Figure

Figure

CHAPTER I

INTRODUCTION

Dimension is not an easy concept to understand. In the early 1900's, it was one of the major problems in mathematics to determine its properties and meanings, resulting in various forms of dimensions (Peitgen, Jurgens, & Saupe, 1992). Among the various dimensions, fractal dimension has been widely used for real-world problems. In geography, fractal dimension is used to facilitate the estimation of the length of coast of Britain; in physiology and biology, fractal dimension is utilized to discuss the metabolic rates of various animals (e.g., rats, dogs, and horses) and relates them to their respective body masses; in financial engineering, fractal dimension is employed to estimate the S&P index; in image analysis, fractal dimension is exploited to quantify texture. The following will briefly describe some of the applications of fractal dimension in the real-world contexts.

One of the salient examples regarding fractal dimension is to measure the length of the coast of Britain in which the question of length is ill-posed. This causes ordinary measurements to become meaningless due to its complexity. Fractal dimension is one way to measure the degree of complexity by evaluating how fast length increases if we measure with respect to smaller and smaller scale. Box-counting dimension (one form of fractal dimension) reveals that the fractal dimension for measuring the length of the coast of Britain is 1.31 (Peitgen et al., 1992).

Fractal dimension has been widely used in financial engineering as well. Much of financial theory relies on the assumption that markets adjust prices rapidly to exclude arbitrage opportunities, i.e., buying an asset at a low price then immediately selling it for a higher price. However, it is well-known that models based on fractional Brownian motion (fBm) allow arbitrage opportunities (Cheridito, 2003; Rogers, 1997). Nevertheless, realization of arbitrage based on this kind of model can be hindered by transaction costs and the minimal amount of time between two consecutive transactions. Even though practical arbitrage application of the fBm in the financial market seems plausible, its possible applications in financial context have been proposed.

In fractal geometry, the fractal dimension, D, is a quantity that gives an indication of how completely a fractal appears to fill space, as one zooms down to finer and finer scales ("Fractal Dimension", 2010). The following examples can help elaborate the idea of fractal dimension. Let us consider the Koch curve. We begin with a straight line of length 1, called the initiator. We then remove the middle third of the line, and replace it with two lines that each have the same length (1/3) as the remaining lines on each side. This new form is called the generator, because it specifies a rule that is used to generate a new form ("Fractals and the Fractal Dimension", 2010). Iterations of such rules constructs the so-called Koch curve (see Figure 1) with fractal dimension $D = \log(N)/\log(r) = \log(4)/\log(3) = 1.26$. Another example is the Sierpinski triangle. One starts with an equilateral triangle, then connect the mid-points of the three sides and remove the resulting inner triangle. Such iterations construct the Sierpinski triangle (see Figure 1) with fractal dimension $D = \log(N)/\log(r) = \log(3)/\log(2) = 1.585$ ("Fractal Dimension", 2010).

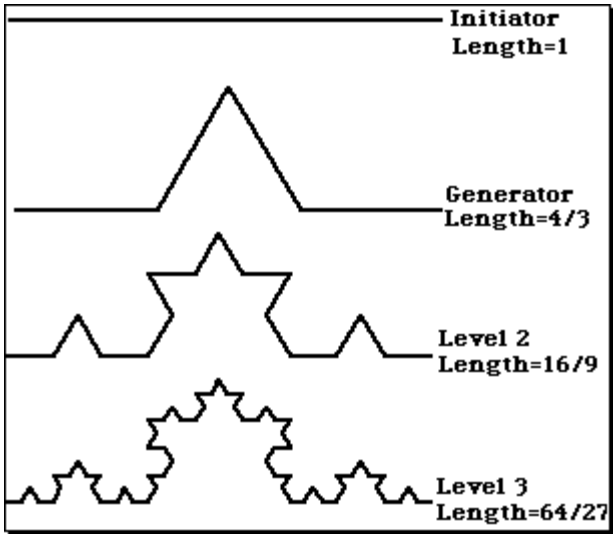

Figure 1 Koch Curve. Source: http://www.vanderbilt.edu/AnS/psychology/cogsci/chaos/workshop/Fractals.html

Figure 2. Sierpinski Triangle. Source: http://en.wikipedia.org/wiki/Fractal_dimension

The primary task of Bayesian inference is to develop the model $p(\theta, x)$ and perform the necessary computations to summarize the posterior distribution $p(\theta|x)$ in appropriate ways. To begin with, we must start with a model providing a joint probability distribution for θ and y. The joint probability density function can be written as a product of two densities that are referred to as the prior distribution $p(\theta)$ and the sampling distribution $p(x|\theta)$ respectively: $p(\theta,x)=p(\theta)p(x|\theta)$. The posterior density:

$$p(\theta|x) = \frac{p(\theta, x)}{p(x)} = \frac{p(\theta)p(x|\theta)}{p(x)}$$

where $p(x)=\int p(\theta)p(x|\theta)d\theta$. Since p(x) does not depend on θ, it yields the unnormalized posterior density: p(θ|x) is proportional to p(θ)p(x|θ) (Gelman, Carlin, Stern, & Rubin, 2004). The primary difference between classical statistical theory and the Bayesian approach is that the latter consider the parameters as random variables that are characterized by a prior distribution (Ntzoufras, 2009).

Although fractal dimension is not easy to understand and estimate, various domains have benefited from studying fractal dimension. Among previous studies regarding fractal dimension, most of them are highly theoretical. Therefore, the current study attempts to estimate fractal dimension from an applied perspective.

Purpose of the Study

The primary purpose of this study is to find Bayesian estimates for the Hurst dimension (the parameter to be estimated) of a fractional Brownian motion (fBm) with a Beta distribution as a prior distribution when the process is observed at both continuous and discrete times. Since there are two parameters (alpha and beta) associated with the Beta distribution, this study will investigate the impact of the parameters on the Hurst dimension estimation. Furthermore, the study attempts to develop a R code for computing Bayesian estimates for the Hurst dimension of a fBm with a Beta prior when the process is observed at both continuous and discrete times.

Research Questions

Q1 Can we find a Bayesian estimate for the Hurst dimension of a fBm with a Beta prior when the process is observed at discrete times?

Q2 Can we find a Bayesian estimate for the Hurst dimension of a fBm with a Beta prior when the process is observed at continuous times?

Q3 Will the Bayesian estimate for the Hurst dimension of a fBm vary when the parameters of the Beta prior change?

Q4 Can we develop a R code for questions 1 through 3?

Rationale for the Study

The fBm has become widely popular in a theoretical context as well as in a practical context for modeling self-similar process since the pioneering work of Mandelbrot and Van Ness in 1968 (Achard & Coeurjolly, 2010). A number of previous studies have been dedicated to the estimation of fractal dimension using various approaches such as the maximum likelihood method (Bishwal, 2003; Breton, 1998; Dahlhaus, 1989; Es-Sebaiy, Ouassou, & Ouknine, 2009; Lundahl, Ohley, Kay, & Siffert, 1986; Praskasa Rao, 2004), wavelet analysis (Bayraktar, Poor, & Sicar, 2004), and discrete variations (Achard & Coeurjolly, 2010). Although much work has been done to the estimation of the Hurst dimension of the fBm, limited studies of this kind focus on the Bayesian estimation. Therefore, the current study is to estimate the Hurst dimension from a perspective of the Bayesian framework.

Rossi, Allenby, and McCulloch (2006) argued that there are really no other approaches (except the Bayesian approach) which can provide a unified treatment of inference and decision as well as properly accounting for parameter and model uncertainty. Namely, flexibility and the generality of the Bayesian approach allow researchers to cope with complex problems. However, somewhat controversial is the view that Bayesian approach delivers the answer to the question in the sense that the Bayesian inference provides answers conditional on the observed data rather than based on distribution of test statistics over imaginary samples not observed. Even though the Bayesian approach has decent benefits, it has some trivial costs including formulation of

prior, requirement of a likelihood function, and computation of various integrals required in Bayesian paradigm (Rossi et al.). Development of various simulation-based methods in recent years has dramatically alleviated the computational costs of the Bayesian approach, leading to the increased adoption of Bayesian models in marketing and other fields. The increased adoption of the Bayesian approach implies that the benefits outweigh the costs for many problems of interest (Rossi et al.). The advancement of computation make complicated integrals become possible. However, choosing an appropriate or objective prior has been an issue in the Bayesian approach (Gelman et al., 2004). Ross and his colleagues argued that investigators are facing a practical problem with little information in the real-world situations and should not neglect sources of information outside of the current data set.

According to the definition of fBm (please see the definition in page 8) proposed by Mandelbrot and Van Ness (1968), the domain of Hurst parameter ranges between 0 and 1. From the non-Bayesian perspective, the Hurst parameter is viewed as a fixed quantity; from the Bayesian perspective, however, the Hurst parameter is regarded as a random variable. With the definition of fBm with $0<H<1$, it is reasonable to use Beta distribution as the prior distribution in order to estimate the Hurst parameter by means of Bayesian approach. As a result, Beta distribution was utilized as the prior distribution in the present study.

Most of the previous studies regarding the estimation of the fractal dimension of fBm involved highly theoretical derivation of the estimator. Limited studies provided practical programming for such problems. Furthermore, Bayesian approach involves complex computation. Winbugs, R, C, and other software are the common tools for

Bayesian computation. Therefore, the current study is to develop R code to the Bayesian estimation of the fractal dimension of fBm using Beta distribution as the prior distribution.

Delimitations of the Study

In these questions I assume there is no other parameter. Namely, the Hurst parameter is the only parameter to be estimated. Additionally, Beta prior for Hurst dimension of a fBm is the only prior distribution for estimation of the Hurst parameter due to the fact that Hurst dimension ranges from 0 to 1, which coincides with the domain of Beta distribution.

Definition of Terms

Bayesian data analysis: involves setting up a joint probability distribution (full probability model) for all observable and unobservable quantities in a problem and calculating and interpreting the conditional probability distribution (posterior distribution) of the unobserved quantities of ultimate interest, given the observed data (Gelman et al., 2004).

Fractal dimension: it is a statistical quantity that gives an indication of how completely a fractal appears to fill space, as one zooms down to finer and finer scales ("Fractal Dimension," 2010).

Fractional Brownian motion: the definition of fractional Brownian motion was proposed by Mendelbrot and Van Ness (1968). Let H be a constant belonging to (0,1). A fractional Brownian motion $(B^{(H)}(t))_{t>0}$ of Hurst index H is a continuous and centered Gaussian process with covariance function $E[B^{(H)}(t)\, B^{(H)}(s)]=1/2(t^{2H}+s^{2H}-|t-s|^{2H})$. For

H=1/2, the fBm is a standard Brownian motion. A standard fbm $B^{(H)}$ has the following properties:

1. $B^{(H)}(0)=0$ and $E[B^{(H)}(t)]=0$ for all $t>0$
2. $B^{(H)}$ has homogeneous increments, i.e., $B^{(H)}(t+s)- B^{(H)}(s)$ has the same law of $B^{(H)}(t)$ for $s,t>0$
3. $B^{(H)}$ is a Gaussian process and $E[B^{(H)}(t)^2]=t^{2H}$, $t>0$, for all H(0,1)

The fBm is divided into three different families corresponding to $0<H<1/2$ (the process is negatively correlated), $H=1/2$ (the process is a Brownian motion), $1/2<H<1$ (the process is positively correlated), respectively.

*Gaussian process: A stochastic process is a collection of time indexed random variables w*here the set of time points could be continuous or discrete. A stochastic process is called strictly stationary if the joint distributions of $X(t_1), \dots, X(t_k)$, only depend on the intervals between $t_1, t_2 \dots t_k$ and are not affected by the shift of the time origin. i.e., the joint distribution of $X(t_1), \dots, X(t_k)$ and the joint distribution of $X(t_{1+\tau}), \dots, X(t_{k+\tau})$ are the same. To illustrate, when $k = 1$, then for all t, the distribution of $X(t)$ is the same with $\mu(t) = \mu$ and $\sigma^2(t) = \sigma^2$. When k=2, the joint distribution of $X(t_1)$ and $X(t_2)$ depends only on the interval between $t_2 - t_1$, which is often called the lag τ. As a result, the autocovariance function between t_1 and t_2 depends on τ only. i.e., $\gamma(t_1, t_2) = \gamma(\tau) = \mathrm{Cov}[X(t), X(t+\tau)]$. In contrast, a less restricted stationarity is called weakly stationarity when there is no specification on moments higher than second order. That is, if a process' mean is constant, i.e., $E[X(t)] = \mu$ and its autocovariance function depends only on the lag, it is said to be second-order stationary. The weakly stationarity is particularly useful in practice since it is less cumbersome to just check with the first

two moments. More specifically, when the joint distribution of $X(t_1), \ldots, X(t_k)$ is distributed as multivariate normal, the process is called the Gaussian process (Chatfield, 2004; Wei, 2006). Thus, Gaussian process can be conceptualized as a generalization of the Normal distribution. It should be noted that since the multivariate normality is uniquely characterized by its first and second moments, a Gaussian process is also strictly stationary. That is, a Gaussian process is possessed of both the strictly stationary and weakly stationary. With these desired properties, a Gaussian process is often used in Bayesian modeling due to its characteristics in the ease of computational tasks. (Chatfield, 2004; Wei, 2006).

Hurst index: it was Mendelbrot that named the parameter H of B(H) after the name of the hydrologist Hurst (Biagini, Hu, Øksendal, & Zhang, 2010).

Posterior distribution: refers to the conditional probability distribution of the unobserved quantities of ultimate interest, given the observed data (Gelman et al., 2004, p. 3).

Prior distribution: refers to a probability distribution that treats parameter as a random variable, which may reflect prior information or belief as to what the true value of the parameter may be (Bain & Engelhardt, 1992).

Summary

Fractional Brownian motion has been applied in various fields since Mendelbrot and Van Ness (1968). Estimation of the Hurst parameter has been of interest for researchers. Various estimation methods have been proposed in order to solve practical problems and enrich the theoretical bases as well. However, Bayesian estimation of the Hurst parameter is limited. Bayesian estimation warrants applied statisticians' efforts due

to its flexibility of incorporating prior information of parameters of interest. Hurst parameter ranges from 0 to 1, making Beta distribution be a reasonable prior. Therefore, the above reasoning motivates the current study.

CHAPTER II

REVIEW OF LITERATURE

Chapter II is organized around five major topics. The first section includes Bayesian data analysis. The second section contains fractal dimension. The third section describes Brownian motion. The fourth section discusses fractional Brownian motion. Additionally, estimations of fractal dimension in fractional Brownian motion are provided, followed by the summary of this chapter.

Bayesian Data Analysis

The main difference between classical statistical theory and Bayesian thinking is that the latter views parameters as random variables that are characterized by a prior distribution (Ntzoufras, 2009). The prior distribution represents the information available to the researcher before any data are involved in the statistical analysis. Three steps are summarized for Bayesian data analysis: the first step is to set up a full probability model; that is, consistent with knowledge about the underlying scientific problem, a joint probability distribution for all quantities is set up. The second step is to condition on observed data; by doing this, the appropriate posterior distribution is calculated. The posterior distribution refers to the conditional probability distribution of the unobserved quantities of interest, given the observed data. The third step is to evaluate the model fit; more specifically, researchers would like to know if the model fits the data (Gelman et al., 2004).

Researchers usually are more interested in calculation of the posterior distribution of the parameters given the observed data. The posterior distribution contains both prior and observed data information. The primary reason for believing Bayesian thinking is that it provides a common-sense interpretation of statistical conclusions. More specifically, a Bayesian interval for an unknown quantity of interest can be directly regarded as having a high probability of containing the unknown quantity, in contrast to confidence interval, which may strictly be interpreted in relation to a sequence of similar inferences that might be made in repeated practice (Gelman et al., 2004). Additionally, flexibility and generality of Bayesian approach allow researchers to cope with complex problems.

In Bayesian data analysis, various numerical summaries of the posterior distribution are desirable. For example, summaries of location such as mean, median, and mode are commonly used; summaries of variation including standard deviation, the interquartile range and other quartiles are often used. Besides, point summaries, it is important to report posterior uncertainty. Probability is particularly used as a fundamental measure for uncertainty in Bayesian data analysis.

Bayesian data analysis depends heavily on simulation because of its relative ease with which samples can often be generated from a probability distribution, even when the density function cannot be explicitly integrated (Gelman et al., 2004). Among the computation techniques, Markov chain Monte Carlo (MCMC) methods are the most helpful. Using MCMC, researchers can set up and estimate complicated models that solve problems that could not be solved with traditional methods (Ntzoufras, 2009).

General guidelines for computation strategy in Bayesian data analysis begin with fitting many models and increase the complexity gradually. We prefer to fit each model relatively quickly, using inferences from the previously-fitted simpler models as starting values, and displaying inferences and comparing to data before continuing (Gelman et al., 2004).

In summary, posterior distribution is the desired product in Bayesian data analysis derived from the specified model and appropriate prior information. Additionally, more common-sense interpretation of statistical conclusions is the primary reason for considering the Bayesian approach. However, Bayesian data analysis relies heavily on computations.

Fractal Dimension

Idea of Fractal Dimension

Fractals, derived from the Latin word frangere meaning to break, are unusual imperfectly defined, mathematical objects that observe self-similarity, that the parts are somehow self-similar to the whole (Peitgen et al., 1992). A structure is said to be strictly self-similar if it can be broken down into arbitrarily small pieces, each of which is a small replica of the entire structure (Peitgen et al.). The self-similarity process implies that fractals are scale-invariant, meaning that you cannot distinguish a small part from the larger structure (see Figure 3, Scrumerati, 2009).

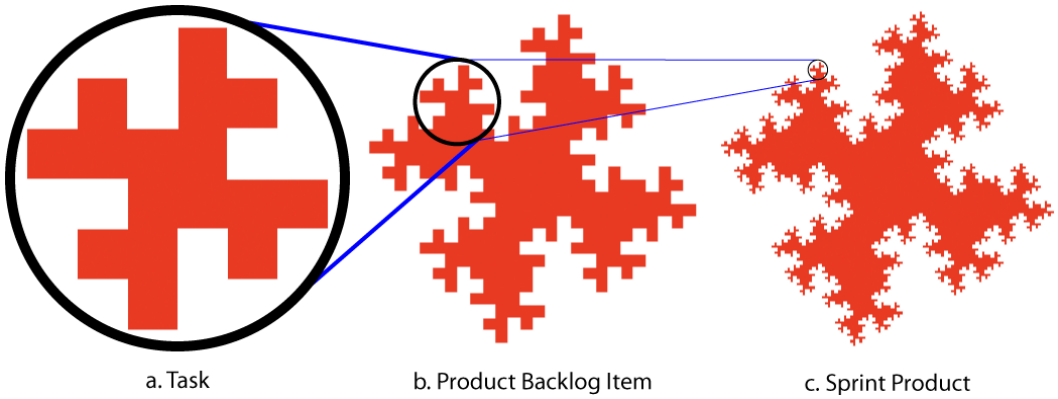

Figure 3. Illustration of Self-Similarity (Excerpted from http://scrumerati.com/2009/05/scrum-fractals.html#tp)

The fractal dimension measures cannot be derived exactly but must be estimated ("Fractal Dimension", 2010). One of the most common examples for fractal dimension is the estimation of the length of the coastline of Britain or in other cases in which area or volume may be ill-posed (see Figure 4). The unit of interest (curve, surface, area or volume) can be so complex that ordinary measurements become meaningless. However, scientists came up with an alternative way to measure the complexity by evaluating how fast the unit of interest increases if we measure with respect to smaller and smaller scales. The idea is that the unit of interest and the scale do not vary arbitrarily, but instead are regulated by a power law. The power law allows researchers to compute one quantity from the other (Peitgen et al., 1992).

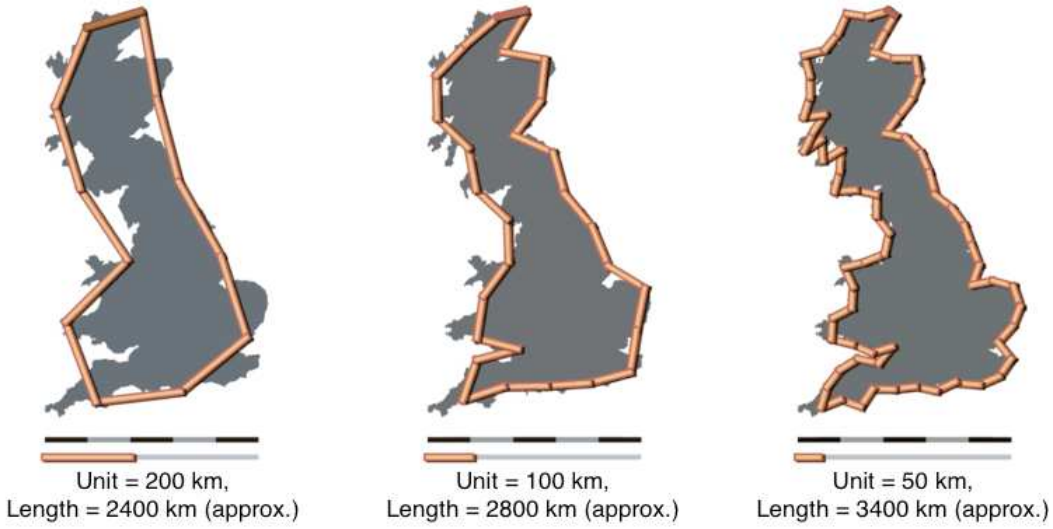

Figure 4. Example of Measurement of Coastline Length of Britain. (Excerpted from http://computationallegalstudies.com/page/3/)

> Consider the following case for further understanding of fractal dimension.
>
> For instance, the length of a coastline may be determined by placing a 1 km ruler end-to-end along the shore line. If a 0.5 km ruler is used for the same coast, then the measured length will be longer. If the increase in length follows a consistent rule over a range of elemental rules, then it may be called a measure of the coastline's geometrical properties. The functional relationship between ruler size and length is $L = \lambda \varepsilon^{1-D}$, where L refers to total length, ε means elemental ruler length, D stands for fractal dimension, and λ indicates scaling constant. In practice, D has been shown to be correlated with the function's intuitive roughness. For D = 1.0, the curve is a smooth line, while for D = 1.99, the line is extremely rough. (Lundahl et al., 1986, p. 152)

Type of Fractal Dimension

Mathematicians have dedicated themselves to come up with various notions of dimension. They include topological dimension, fractal dimension, self-similarity dimension, box-counting dimension, Hausdorff dimension, capacity dimension, information dimension, compass dimension, Euclidean and more (Peitgen et al., 1992). Among the various dimensions, I will briefly introduce the following dimensions for

clarification including self-similarity dimension, compass dimension, and box-counting dimension, each of which is a special form of Mandelbrot's fractal dimension.

Self-similarity dimension. Given a self-similar structure, a relation exists between the reduction factor *s* and the number of pieces *a* into which the structure can be divided. The relation is shown as follows: $a=1/s^D$, or equivalently, D=(log a)/(log(1/s)), where D is called the self-similarity dimension (Peitgen et al., 1992). Let us take the Koch curve (refer to Figure 1) as an example, assuming a=4, s=1/3 and a=16, s=1/9, respectively. By looking at the both conditions, an identical self-similar dimension of the Koch curve D was derived, where D=(log4)/(log3)=(log16)/(log9)=1.2619. This example implies that the power law relation between the number of pieces and the reduction factor results in identical self-similarity dimension, regardless of the scale used for evaluation (Peitgen et al.).

Compass dimension. Compass dimension (also called divider or ruler dimension) is defined as D=1+d, where d is the slope in the log/log-diagram of the measured length *u* versus precision 1/*s* (Peitgen et al., 1992). For example, d for the coast of Britain is approximately 0.36, meaning that the coast has a compass (fractal) dimension of around 1.36. Another example is the 3/2 curve in which d=0.5 makes the compass dimension D=1+0.5=1.5 (Peitgen et al.).

Box-counting dimension. When structures have certain special properties such as self-similar or structures like a coastline, self-similarity dimension and compass dimension can be used to deal with such problems. However, what can be done if the structures are not of special properties? The box-counting dimension has been developed to deal with structures of no special properties. The idea of box-counting dimension is

related to coastline measurements (Peitgen et al., 1992). More specifically, the structure is put onto a regular mesh with mesh size s and then the number of grid boxes that contain some of the structure, say N are counted. The relationship between s and N is demonstrated to be N(s) due to the dependence of N on the choice of s. Then, s is changed progressively to smaller and smaller sizes and the corresponding number N(s) is counted. Finally, the measurements in a log(N(s))/log(1/s) diagram is made to fit a straight line to the plotted points of the diagram and calculate its slope D, which is the box-counting dimension (Peitgen et al). The box-counting dimension is one of the most used dimensions in measurements and in all the sciences due to the following reasons. First, the box-counting dimension proposes a systematic way to measure any structure in the plane or even in the space. Additionally, it is straightforward to calculate the dimension by counting boxes and maintain statistics. Finally, the structure can be the form of either self-similar or wild (Peitgen et al).

In summary, fractal dimension may not be an easy concept to understand. Various types of fractal dimension have been proposed in a variety of research areas. Fractal dimension can not be derived exactly but has to be estimated. Due to its potential for applications, many studies have been dedicated to the estimation of the fractal dimension.

Brownian Motion

Brownian motion is named after Robert Brown, who observed the motion in 1827 (Mazo, 2008). The definition of Brownian motion is as follows (Chiang, 2006; Mörters & Peres, 2008). A real-valued stochastic process $\{B(t): t\geq 0\}$ is called a linear Brownian motion with start in x belongs to R if the following holds:

1. B(t) is Gaussian,

2. $B(0) = x$,

3. B(t) has independent increments, i.e., for all times $0 \leq t_1 \leq t_2 \leq ... \leq t_n$, the increments $B(t_n) - B(t_{n-1})$, $B(t_{n-1}) - B(t_{n-2})$, …, $B(t_2) - B(t_1)$ are independent variables;

4. for all $t \geq 0$ and $h>0$, the increments $B(t + h) - B(t)$ are normally distributed with expectation zero and variance h;

5. $E[B(t)-B(s)]=0$, and

6. $Var[B(t)-B(s)]= \sigma^2(t-s)$ for $s < t$.

Brownian motion has been well established in finance. However, classical mathematical models of financial assets are far from perfect. There are two problems associated with classical mathematical models of financial assets; namely, financial processes are not wholly Gaussian and Markovian in distribution. For decades, researchers have argued that it is reasonable to assume all information contained within current asset price to be Markovian process. Nevertheless, technical traders have consistently beaten the market. Similarly, Mandelbrot (1997) pointed out a list of discrepancies between Brownian motion and the facts. They include non-stationarity of the underlying rules, repeated instances of discontinuous change, and long-term dependence and so on. This resulted in academic efforts in purporting the existence of non-Markovian market. Fractional Brownian motion deals with the long-ranged dependence problem while still assuming a Gaussian process. It has the advantage of giving simple and tractable solutions as opposed to the stochastic volatility models (Daye, 2003). The following section deals with fractional Brownian motion.

Fractional Brownian Motion

The fractional Brownian motion was named as fractional Brownian motion by Mandelbrot and Van Ness. The fractional Brownian motion, which provides a suitable generalization of the Brownian motion, is one of the simplest stochastic processes exhibiting long-range dependence (Bishwal, 2003; Breton, 1998). It has been used as a modeling tool. The following demonstrates the stochastic integral representation of fractional Brownian motion (Biagini et al., 2010, p.6). The process

$$Z(t) = \frac{1}{\Gamma\left(H + {}^{1}/_{2}\right)} \int \left((t-s)^{H-1/2} - (-s)^{H-{}^{1}/_{2}}\right) dB(s)$$

$$= \frac{1}{\Gamma\left(H + {}^{1}/_{2}\right)} \left(\int_{-\infty}^{0} \left((t-s)^{H-1/2} - (-s)^{H-{}^{1}/_{2}}\right) dB(s) + \int_{0}^{t} (t-s)^{H-1/2}\, dB(s) \right)$$

where B(t) is a standard Brownian motion and Γ refers to the gamma function, is a fBm with 0< Hurst index <1. The constant $1/\Gamma(H+1/2)$ in the following computation is dropped for the sake of simplicity. Let , $ds = t\,du$ $\frac{s}{t} = u$

$$E[Z^2(t)] = \int \left[(t-s)_{+}^{H-1/2} - (-s)_{+}^{H-{}^{1}/_{2}}\right]^2 ds = \int [t^{H-\frac{1}{2}} \left(1 - \frac{s}{t}\right)^{H-\frac{1}{2}} - t^{H-\frac{1}{2}} (-\frac{s}{t})^{H-\frac{1}{2}}]^2 \, ds$$

$$= \int t^{2H-1} \left[\left(1 - \frac{s}{t}\right)^{H-\frac{1}{2}} - \left(-\frac{s}{t}\right)^{H-\frac{1}{2}} \right]^2 ds = \int t^{2H-1} \left[(1-u)^{H-\frac{1}{2}} - (-u)^{H-\frac{1}{2}} \right]^2 t\,du$$

$$= t^{2H} \int \left[(1-u)_{+}^{H-1/2} - (-u)_{+}^{H-{}^{1}/_{2}}\right]^2 du \; = C(H) t^{2H}$$

Where $C(H) = \int \left[(1-u)_{+}^{H-1/2} - (-u)_{+}^{H-{}^{1}/_{2}}\right]^2 du.$

Analogously, we have that

$$E[|Z(t) - Z(s)|^2] = \int \left[(t-u)_{+}^{H-1/2} - (s-u)_{+}^{H-{}^{1}/_{2}}\right]^2 ds$$

$$= t^{2H} \int \left[(t - s - u)_+^{H-1/2} - (-u)_+^{H-1/2} \right]^2 du \; = C(H)|t - s|^{2H}$$

Now

$$E[Z(t)Z(s)] = -\frac{1}{2}\{E[|Z(t) - Z(s)|]^2 - E[Z(t)^2] - E[Z(s)^2]\}$$

$$= \frac{1}{2}(t^{2H} + s^{2H} - |t - s|^{2H})$$

According to the definition of fractional Brownian motion fBm proposed by Mendelbrot and Van Ness (1968), a fractional Brownian motion $(B^{(H)}(t))_{t>0}$ of Hurst index H is a continuous and centered Gaussian process with covariance function $E[B^{(H)}(t) B^{(H)}(s)]=1/2(t^{2H}+s^{2H}-|t-s|^{2H})$. Therefore, Z(t) is a fBm of Hurst index H.

The fBm is divided into three different families corresponding to $0<H<1/2$, $H=1/2$, $1/2<H<1$, respectively. It was Mandelbrot that named the parameter H of $B^{(H)}$ after the name of the hydrologist Hurst, who made a statistical study of yearly water run-offs of the Nile river. Hurst found that the water run-offs in Nile River could not be modeled by using a process with independent increments, but rather the increments could be viewed as the increments of a fBm. Due to Hurst' study, Mandelbrot introduced the name Hurst index. The basic feature of fBm is that the span of independence between their increments can be infinite (Mandelbrot & Van Ness, 1968). As the Hurst parameter H governs the fractal dimension of the fractional Brownian motion, its regularity and the long-memory behavior of its increments, the estimation of H is an important but difficult task which has led to very vast literature (Achard & Coeurjolly, 2010). The following section describes estimation of fractal dimension in fractional Brownian motion.

Estimation of Fractal Dimension for the Fractional Brownian Motion

Maximum Likelihood Estimation

Maximum likelihood estimation (MLE) is often considered to be the best obtainable estimator. In addition, the estimate is asymptotically unbiased (unbiased as the sample size becomes large), asymptotically efficient (it obtains the Cramer-Rao bound) and is asymptotically Normally distributed (Lundahl et al., 1986). In most cases, it is impossible to obtain an explicit form for the estimate as a function of the data. Instead, numerical methods are used to find the maximum of the likelihood function (Lundahl et al.).

The Bishwal (2003) studies show the properties of the MLE of a parameter appearing linearly in drift coefficient of a nonlinear stochastic differential equation driven by fBm when the signal process is a nonlinear diffusion process. They proved the strong consistency and asymptotic normality of the MLE, and verified that the MLE can be explicitly calculated. This satisfies the asymptotic properties mentioned in their work.

Lundahl and his colleagues (1986) extended the basic theory of fractional Brownian motion to the discrete case. More specifically, an asymptotic Cramer-Rao bound is derived for the variance of an estimate of H; a maximum likelihood estimator is developed to estimate H. Results reveal that the variance of the estimator nearly achieves the minimum bound. A generation algorithm for discrete fractional motion is presented and used to demonstrate the capabilities of the MLE when the discrete fractional Brownian process is contaminated with additive Gaussian noise. The results indicate that it has strong potential for quantifying texture. Furthermore, Dahlhaus (1989) proved the

asymptotic normality of the maximum likelihood estimator for the parameters of a long range dependent Gaussian process.

In spite of the fact that maximum likelihood estimators are consistent and asymptotically normal and also asymptotically efficient in general, they have the following shortcomings. First, the calculations are often cumbersome since the expression for the MLE involves stochastic integrals which need appropriate approximations for computational purposes. Additionally, the MLE are not robust in the sense that a slight perturbation in the noise component will change the properties of the MLE substantially. Therefore, other estimation methods were proposed in order to circumvent such problems (Rao, 2004).

Other Estimation Methods of Fractal Dimension in fBm

Bayraktar et al. (2004) used Wavelet analysis to estimate the fractal dimension of the S&P 500 index. They sampled and analyzed S&P 500 data at one-minute intervals over the course of 11.5 years (January 1989-May 2000). Specifically, they developed a method to investigate long range dependence, quantified by the Hurst parameter in a high frequency financial time series. They found the Hurst parameter to be around the 0.6 level for most of the 1990s, but dropped to the level of 0.5 in the period 1997-2000, which coincides with growth in Internet trading among small investors.

Achard and Coeurjolly (2010) reviewed different estimation procedures of Hurst parameter of fractional Brownian motion and tried to provide estimators that were quickly computable. They described four methods of estimation of the Hurst parameter: (a) the standard procedure based on the log-linearity of the variogram of dilated time series (ST), (b) robust alternatives to outliers using sample quantiles (Q) or trimmed

means (TM), (c) robust alternative Gaussian white noise or to additive Brownian motion (methods B0 and B1), and (d) robust alternatives to outliers and additive noise by combining these methods. Additionally, three different models of contamination were used in this study: (a) a model of additive outlier (AO), (b) a model of additive Gaussian white noise to the fBm (B0), and (c) a model of additive Gaussian white noise to the fGn (B1). They further recommended, from a practical perspective, that one should first observe the data for the presence of outliers, and the estimator based on trimmed means (TM) should be used. Moreover, they recommended use the standard method if the differences $|H^{ST}-H^{B0-ST}|$and $|H^{ST}-H^{B1-ST}|$ are close to zero.

Chiang (2006) utilized the method of Embedded Branching Process (EBP) proposed by Jones and Shen (2004) to estimate the Hurst parameter, which builds a tree of crossings that encodes the sample path. The estimator is as follows: $\hat{H}=\frac{\log 2}{\log \mu}$ where $\mu = \frac{\sum_{k=1}^{N(m)} Z_k^m+\sum_{k=1}^{N(m+1)} Z_k^{m+1}+\cdots+\sum_{k=1}^{N(n)} Z_k^n}{N(m)+N(m+1)+\cdots+N(n)}\frac{\log 2}{\log \mu}$, N(.)is the total number of crossings, and Z is the number of subcrossings.

Berzin and Leon (2007) consider the second order increments of a fBm using variation techniques. Based on an almost-sure convergence theorem for general functions, they construct certain regression models for the parameter H. The regression based estimator for H turns out to be asymptotically unbiased, consistent and that it satisfies the Central Limit Theorem.

Constantine and Hall (1994) proposed simple methods for estimating fractal dimension based on variogram along with log linear regression. The estimator for fractal dimension presented in their work as $D= 2 - \frac{1}{2}\alpha$, where

$\alpha = \{\sum_{j=1}^{m}(x_j - \bar{x})w_j \log(g_j)\}\{\sum_{j=1}^{m}(x_j - \bar{x})w_j\}^{-1}$. w_j denotes the positive weight; the integer m plays the role of a smoothing parameter, describing the distance away from the origin; log (g_j) refers to the heteroscedasticity of the variables.

Moreover, Dieker (2004) proposed an aggregate variance method, which is based on the self-similarity of the sample. The idea is that the sequence $\{x_k\}$ is divided into blocks of size m, and then the aggregated process is defined as $x_k^{(m)}=m^{-1}(x_{km}+\ldots+x_{(k+1)m-1})$, and $Var(x_k^{(m)})=m^{2H-2}N^{-2H}$. An estimator for $Var(x_k^{(m)})$ is $M^{-1}\sum_{i=1}^{M-1}(x_i^{(m)} - \bar{x}^{(m)})^2$, where M=integer part of N/m and $\bar{x}^{(m)} = M^{-1}\sum_{i=1}^{M-1} x_i^{(m)}$. The estimator of Hurst dimension is obtained by plotting the estimated variance of $x_k^{(m)}$ against m on a log-log scale.

Higuchi (1988) estimated Hurst dimension by plotting L(m) in a log-log plot versus m and adding 2 to the fitted line, where $L(m)= \frac{N-1}{m^3}\sum_{i=1}^{m}\frac{1}{M_i}\sum_{k=1}^{M_i}\left|\sum_{j=i+(k-1)m+1}^{i+km} x_j\right|$, and M_i is the integer part of (N-i)/m.

Summary

The estimation of fractal dimension in fractional Brownian motion has attracted much attention for decades. Many efforts were made in using maximum likelihood estimators. However, cumbersome calculations involving stochastic integrals which need appropriate approximations for computational purposes and the robustness issue associated with the MLE led to other estimation methods. Bayesian estimation for fractal dimension in fractional Brownian motion along with the prior of Beta distribution is limited. Therefore, the current study attempts to estimate the fractal dimension in fractional Brownian motion using the Bayesian approach.

CHAPTER III

METHODOLOGY

Various methods of the estimation of the Hurst parameter have been proposed. There are, however, limited studies focusing on Bayesian data analysis along with Beta distribution as the prior information for the estimation of the Hurst parameter for the fractional Brownian motion. Thus, we will focus on the Bayesian approach to estimate the Hurst parameter for the fractional Brownian motion. In the following sections, the Markov Chain Monte Carlo (MCMC) method will be specified in order to obtain the Bayesian estimators. Additionally, model descriptions along with the likelihood, prior and posterior functions will be discussed. The final section describes information associated with data simulations including different choices of the parameters of the Beta prior in order to examine if the Bayesian estimators are sensitive to the parameters of the Beta prior as well as number of replications.

Markov Chain Monte Carlo Method

Direct simulation cannot be applied in all cases when posterior distributions of interest involve multidimensional integrals. Simulation techniques based on Markov chains overcome such problem due to their generality and flexibility. Markov Chain Monte Carlo (MCMC) techniques have become popular since the early 1990s due to the massive development of computing facilities. The MCMC techniques enable quantitative researchers to use highly complicated models and estimate the corresponding posterior distributions with accuracy. These MCMC techniques are based on the construction of a

Markov chain that eventually converges to the target distribution or so-called posterior distribution f(θ|x). This characteristic distinguishes MCMC algorithms from direct simulation methods, which provide samples directly from the posterior distribution. Furthermore, the MCMC output is a dependent sample because it is generated from a Markov chain, in contrast to the output of direct methods, which is an independent sample. Finally, MCMC methods, frequently called iterative methods, involve the notion of an iterative procedure since in every step they produce values based on the previous one (Ntzoufras, 2009). The following sections include (a) definitions and terminology associated with the MCMC method, (b) Markov chain--the algorithm of MCMC, (c) how to describe the target distribution using MCMC output, (d) Monte Carlo error, and (e) two popular MCMC algorithms, Metropolis-Hastings algorithm and the Gibbs sampling.

Definitions and Terminology

In this section, definitions and terminology associated with MCMC are presented as follows. They include equilibrium distribution, convergence of the algorithm, iteration and total number of iterations T, initial values of the chain $\theta^{(0)}$, burnin period, thinning interval or sampling lag, iterations kept T', MCMC output, and output analysis (Ntzoufras, 2009).

Equilibrium distribution. This is also known as the stationary or target distribution of the MCMC algorithm. The notion of the equilibrium distribution is related to the Markov chain used to construct the MCMC algorithm, such that the chain stabilizes to the equilibrium/stationary distribution after a number of time sequences $t > B$. Therefore, in a Markov chain, the distribution of $\theta^{(t)}$ and $\theta^{(t+1)}$ will be identical and equal to the equilibrium/stationary distribution. Equivalently, once it reaches its

equilibrium (distribution), an MCMC scheme generates dependent random values from the corresponding stationary distribution (Ntzoufras, 2009, p. 38).

Convergence of the algorithm. An MCMC algorithm converges when the algorithm has reached its equilibrium and generates values from the desired target distribution. Generally it is unclear how much we must run an algorithm to obtain samples from the correct target distributions. Several diagnostic tests have been developed to monitor the convergence of the algorithm, including monitoring the Monte Carlo error, monitoring the trace plots (the plots of iterations versus generated values), examining the ergodic mean (the mean value until the current iterations), and other statistical diagnostics. Small values of Monte Carlo error indicate quantity of interest with precision. Moreover, convergence is ensured if all values are within a zone without strong periodicities. Additionally, an indication of the convergence of the algorithm is achieved if the ergodic mean stabilizes after some iterations. However, it is recommended that all diagnostics must be applied to ensure that convergence has been reached (Ntzoufras, 2009, p. 38, 41).

Iteration. Iteration refers to a cycle of the algorithm that generates a full set of parameter values from the posterior distribution. For example, $\theta^{(7)}$ and $\theta^{(t)}$, respectively, denote the values of random vector θ generated at the 7th and *t* th iterations of the algorithm. Additionally, total number of iterations T refers to the total number of the iterations of the MCMC algorithm (Ntzoufras, 2009).

Initial values of the chain $\theta^{(0)}$. The initial values refer to the starting values used to initialize the chain. These initial values may influence the posterior summaries if they are far away from the highest posterior probability areas. Solutions to mitigating or

avoiding the influence of the initial values include removing the first iterations of the algorithm or letting the algorithm run for a large number of iterations or obtain different samples with different starting points (Ntzoufras, 2009).

Burnin period. In the burnin period the first B iterations are eliminated from the sample in order to avoid the influence of the initial values. If the generated sample is large enough, the effect of this period on the calculation of posterior summaries is minimal (Ntzoufras, 2009, p. 38).

Thinning interval or sampling lag. As has already been mentioned, the final MCMC generated sample is not independent. For this reason, we need to monitor the autocorrelations of the generated values and select a sampling lag L> 1 after which the corresponding autocorrelation are low. Then, we can produce an independent sample by keeping the first generated values in every batch of L iterations. Hence, if we consider a lag (or thin interval) of three iterations, then we keep the first of every three iterations (i.e., we keep observations 1, 4, 7, etc.). This tactic is also followed to save storage space or computational speed in high-dimensional problems (Ntzoufras, 2009, p. 38).

Iterations kept T'. These are the number of the iterations retained after discarding the initial burnin iterations (i.e., T'=T-B). If we also consider a sampling lag L>1, then the total number of iterations kept refers to the final independent sample used for posterior analysis (Ntzoufras, 2009, p. 38).

Markov Chain Monte Carlo (MCMC) output. This refers to the MCMC generated sample. We often refer to the MCMC output as the sample after removing the initial iterations (produced during the burnin period) and considering the appropriate lag (Ntzoufras, 2009, p. 38).

Output analysis. This refers to analysis of the MCMC output sample. It includes both the monitoring procedure of the algorithm's convergence and analysis of the sample used for the description of the posterior distribution and inference about the parameters of interest (Ntzoufras, 2009, p. 38).

Markov Chain. The algorithm of MCMC is primarily based on the Markov chain. A Markov chain is a stochastic process $\{\theta^{(1)},\theta^{(2)},\ldots,\theta^{(T)}\}$ such that

$$f(\theta^{(t+1)}|\theta^{(t)},\ldots,\theta^{(1)}) = f(\theta^{(t+1)}|\ \theta^{(t)}) \qquad (3.1);$$

From equation 3.1, we know that the distribution of θ at time t+1 given all the preceding θ values (from times t, t-1, …, 1) depends only on the value $\theta^{(t)}$ of the previous step t. Moreover, $f(\theta^{(t+1)}|\ \theta^{(t)})$ is independent of time t and the initial values of the chain $\theta^{(0)}$. In order to generate a sample from, $f(\theta|x)$, we must construct a Markov chain with two desired properties: (a) $f(\theta^{(t+1)}|\ \theta^{(t)})$ should be easy to generate from, and (b) the equilibrium distribution of the selected Markov chain must be the posterior distribution of interest $f(\theta|\ x)$ (Ntzoufras, 2009). The following steps are recommended to construct a Markov chain.

1. Select an initial value $\theta^{(0)}$.
2. Generate T values until the equilibrium distribution is reached.
3. Monitor the convergence of the algorithm using convergence diagnostics. We generate more observations if convergence diagnostics fail.
4. Cut off the first B observations.
5. Consider $\{\theta^{(B+1)},\theta^{(B+2)},\ldots,\theta^{(T)}\}$ as the sample for the posterior analysis.
6. Plot the posterior distribution; specifically the univariate marginal distributions
7. Finally, obtain summaries of the posterior distribution including mean, median, standard deviation, quantiles, and correlations etc. (Ntzoufras, 2009, pp. 36-37)

Describing the Target Distribution Using MCMC Output

The MCMC output provides us with a random sample of the type

$$\theta^{(1)},\ \theta^{(2)},\ldots,\ \theta^{(t)},\ldots\theta^{(T')}.$$

From this sample, for any function $G(\theta)$ of the parameters of interest θ, Ntzoufras (2009, p. 39) suggested the following procedures to describe the target distribution using MCMC output.

1. Obtain a sample of the desired parameter $G(\theta)$ by simply considering

$$G\left(\theta^{(1)}\right), G\left(\theta^{(2)}\right), \ldots, G\left(\theta^{(t)}\right), \ldots, G\left(\theta^{(T')}\right).$$

2. Obtain any posterior summary of $G(\theta)$ from the sample using traditional sample estimates. For example, we can estimate the posterior mean by

$$\widehat{E}(G(\theta)|y) = \overline{G(\theta)} = \frac{1}{T'}\sum_{t=1}^{T'} G\left(\theta^{(t)}\right) \qquad (3.2)$$

and the posterior standard deviation by

$$\widehat{SD}\left(G(\theta|x)\right) = \frac{1}{T'-1}\sum_{t=1}^{T'}\left[G\left(\theta^{(t)}\right) - \widehat{E}(G(\theta)|x)\right]^2 \qquad (3.3)$$

3. Calculate and monitor correlations between parameters.

4. Produce plots of the marginal posterior distributions.

Monte Carlo Error

The Monte Carlo error (MC error), an important measure that must be reported and monitored in the analysis of the MCMC output, measures the variability of each estimate due to the simulation. Low MC error indicates the parameter of interest with increased precision. It is proportional to the inverse of the generated sample size that can be controlled by the user. Therefore, for a sufficient number of iterations T, the quantity of interest can be estimated with increased precision. Batch mean method and the

window estimator method are the two most common ways to estimate MC error (Ntzoufras, 2009, p. 39).

In order to calculate the MC error using the batch means method, we simply partition the resulting output sample in K batches (usually K = 30 or K = 50). Both the number of batches K and the sample size of each batch $\nu = T'/K$ must be sufficiently large in order to enable us to estimate the variance consistently and also eliminate autocorrelations (Ntzoufras, 2009, p. 39).

The following procedures deal with the calculation of the Monte Carlo error of the posterior mean of $G(\theta)$ (Ntzoufras, 2009, p. 40). First, calculate each batch mean $\overline{G(\theta)_b}$ by:

$$\overline{G(\theta)_b} = \frac{1}{\nu}\sum_{t=(b-1)\nu+1}^{b\nu} G\left(\theta^{(t)}\right)$$

for each batch $b = 1, \ldots, K$, and the overall sample mean by

$$\overline{G(\theta)} = \frac{1}{T'}\sum_{t=1}^{T'} G\left(\theta^{(t)}\right) = \frac{1}{K}\sum_{b=1}^{K}\overline{G(\theta)_b},$$

assuming that we keep $\theta^{(1)}, \ldots, \theta^{(T')}$ observations. Then an estimate of the MC error is simply given by the standard deviation of the batch means estimates $\overline{G(\theta)_b}$

$$\mathrm{MCE}\left[G(\theta)\right] = \widehat{\mathrm{SE}}\left[\overline{G(\theta)}\right] = \sqrt{\frac{1}{K}}\,\widehat{\mathrm{SD}}\left[\overline{G(\theta)_b}\right]$$

$$= \sqrt{\frac{1}{K(K-1)}\sum_{b=1}^{K}\left(\overline{G(\theta)_b} - \overline{G(\theta)}\right)^2} \qquad (3.4)$$

The procedure for calculating the MC error for any other posterior quantity of interest $\widehat{U} = U\left(\theta^{(1)}, \ldots, \theta^{(T')}\right)$ is equivalent. To estimate the corresponding Monte

Carlo error, we calculate $\widehat{U}_b = U\left(\theta^{((b-1)v+1)}, \ldots, \theta^{(bv)}\right)$ from each batch $b = 1,\ldots, K$ and then the MC error by

$$\text{MCE}\left(\widehat{U}\right) = \sqrt{\frac{1}{K(K-1)}\sum_{b=1}^{K}\left(\widehat{U}_b - \widehat{U}\right)^2} \qquad (3.5)$$

The second method (window estimator) is based on the expression of the variance in auto correlated samples given by Ntzoufras (2009)

$$\text{MCE}\left[G(\theta)\right] = \frac{\widehat{SD}\left[G(\theta)\right]}{\sqrt{T'}}\sqrt{1 + 2\sum_{k=1}^{\infty}\hat{\rho}_{k\left[G(\theta)\right]}},$$

where $\hat{\rho}_{k\left[G(\theta)\right]}$ is the estimated autocorrelation of lag k, that is, the correlation between parameters $G\left(\theta^{(t)}\right)$ and $G\left(\theta^{(t+k)}\right)$. Thus, it is obvious that for large k the autocorrelations will not be estimated reliably from the sample because of the small number of remaining observations. Moreover, in practice the autocorrelation will be close to zero for a sufficiently large k. For this reason, we identify a window w after which autocorrelations are considerably low $\left[\text{say, } < 0.1 \text{ (Carlin and Louis, 2000)}\right]$ and discard $\hat{\rho}_k$ with $k > w$ from the preceding MC error estimate. Hence, this window based modified MC error estimate is given by

$$\text{MCE}\left[G(\theta)\right] = \frac{\widehat{SD}\left[G(\theta)\right]}{\sqrt{T'}}\sqrt{1 + 2\sum_{k=1}^{w}\hat{\rho}_{k\left[G(\theta)\right]}} \qquad (3.6)$$

Popular MCMC Algorithm

f(θ)The following section briefly introduces the concept of the two most popular MCMC methods: the Metropolis-Hastings algorithm and the Gibbs sampling. Since we have just one parameter, H, to estimate, we will not focus on the Gibbs sampling.

$\theta^{(0)}\theta^{(t)}$The Metropolis-Hastings algorithm. Assume a target distribution from which we wish to generate a sample of size T. The Metropolis-Hastings algorithm can be described by the following iterative steps (Ntzoufras, 2009, p. 43); where is the vector of generated values in t iteration of the algorithm:

1. Set initial values $\theta^{(0)}$.
2. For t= 1,…, T repeat the following steps
   a. Set $\theta = \theta^{(t-1)}$
   b. Generate new candidate values θ' from a proposal distribution q $(\theta \to \theta')$
   $= q(\theta'|\theta))$.
   c. Calculate
   $$\alpha = \min\left(1, \frac{f(\theta')q(\theta|\theta')}{f(\theta)q(\theta'|\theta)}\right).$$
   d. Update $\theta^{(t)} = \theta'$ with probability α and $\theta^{(t)} = \theta = \theta^{(t-1)}$ with probability 1-α

The Metropolis-Hastings algorithm will converge to its equilibrium distribution regardless of whatever proposal distribution q is selected. Nevertheless, in practice, the choice of the proposal is important since poor choices will considerably delay convergence towards the equilibrium distribution.

The algorithm outlined above can be directly implemented in Bayesian framework by substituting the target distribution $f(\theta)$ by the posterior distribution $f(\theta|x)$. Thus, in Bayesian inference, the algorithm is summarized as follows:

1. Set initial values $\theta^{(0)} f(\theta)$.
2. For t = 1,…, T repeat the following steps
   a. Set $\theta = \theta^{(t-1)}$
   b. Generate new candidate parameter values θ' from a proposal distribution q $(\theta'|\theta)$
   c. Calculate

$$\alpha = \min\left(1, \frac{f(\theta'|x)q(\theta|\theta')}{f(\theta|x)q(\theta'|\theta)}\right) \qquad (3.7)$$

   d. Update $\theta^{(t)} = \theta'$ with probability α; otherwise set $\theta^{(t)} = \theta$.

Model Descriptions

The likelihood function for fractional Brownian motion, the distribution of Beta prior, and the posterior distribution are as follows.

The likelihood function for fractional Brownian motion for the discrete case is as follows (Lundahl et al., 1986, p. 155):

$$f(\mathbf{x}|H) = \frac{1}{(2\pi)^{\frac{n}{2}}|\mathbf{R}|^{\frac{1}{2}}} \exp\{-\frac{1}{2}\mathbf{x}^{T}\mathbf{R}^{-1}\mathbf{x}\} \quad (3.8)$$

where H is the Hurst parameter, $\mathbf{x}=(x_1, x_2,\ldots, x_n)$ follows a standard fBm and $\mathbf{R}_{ij}=1/2(i^{2h}+j^{2h}-|i-j|^{2h})$.

The Beta prior distribution for H is as follows:

$$P(h) = \frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)} h^{\alpha-1}(1-h)^{\beta-1} \qquad (3.9)$$

The full probability model is derived from the product of (3-5) and (3-6):

$$g(\mathbf{x}; h) = \frac{1}{(2\pi)^{\frac{n}{2}}|\mathbf{R}|^{\frac{1}{2}}} \exp\{-\frac{1}{2}\mathbf{x}^T\mathbf{R}^{-1}\mathbf{x}\} \cdot \frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)} h^{\alpha-1}(1-h)^{\beta-1} \qquad (3.10)$$

Then we get the posterior distribution $k(h|\, x_1, x_2, \ldots, x_n)$ for Q1 as follows:

$$k(h|\, x_1, x_2, \ldots, x_n) = \frac{g(x;h)}{m(\mathbf{x})} \qquad (3\text{-}8)$$

$$k(h|x_1 \ldots x_n) = \frac{g(x;h)}{m(x)} = \frac{\frac{1}{(2\pi)^{\frac{n}{2}}|\mathbf{R}|^{\frac{1}{2}}}\exp\{-\frac{1}{2}\mathbf{x}^T\mathbf{R}^{-1}\mathbf{x}\}\frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)}h^{\alpha-1}(1-h)^{\beta-1}}{\int_0^1 \frac{1}{(2\pi)^{\frac{n}{2}}|\mathbf{R}|^{\frac{1}{2}}}\exp\{-\frac{1}{2}\mathbf{x}^T\mathbf{R}^{-1}\mathbf{x}\}\frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)}h^{\alpha-1}(1-h)^{\beta-1}dh} \qquad (3.11)$$

For the continuous case, however, the probability measures generated by two different Hurst dimension processes are singular with respect to each other (Praskasa Rao, 2008), so it follows that there is no likelihood function for the continuous case. The theorem proposed by Praskasa Rao (2008, p. 23) is as follows.

Theorem: Let $\{W_{Hi}(t), t \geq 0\}$, $i = 1, 2$, be two standard fBms with Hurst indices $H1 \neq H2$. Let *Pi* be the probability measure generated by the process $\{W_{Hi}(t, t \geq 0\}$ for $i = 1, 2$. Then the probability measures *P1* and *P2* are singular with respect to each other.

Before moving on to the proof of theorem regarding singularity of fBms for different Hurst indices, the theorem proposed by Kurchenko (2003) needs to be addressed to facilitate the proof of theorem regarding singularity of fBms for different Hurst indices. Let $\{W_H(t), t \geq 0\}$ be standard fBm with Hurst index $H \in (0, 1)$. Then, with probability one, $\lim_{n\to\infty} \frac{1}{n}\sum_{m=0}^{n-1}[W_H(m) - 2W_H\left(m + \frac{1}{2}\right) + W_H(m+1)]^2 = V_2(0, H)$ for any standard fBm with Hurst index $H \in (0, 1)$.

Proof: Using the theorem proposed by Kurchenko (2003),

$\lim_{n\to\infty} \frac{1}{n}\sum_{m=0}^{n-1}[W_{Hi}(m) - 2W_{Hi}\left(m + \frac{1}{2}\right) + W_{Hi}(m+1)]^2 = V_2(0, Hi)$, i=1, 2.

Since $V_2(0,H_1) \neq V_2(0,H_2)$ if $H_1 \neq H_2$, and since the convergence stated above is convergence under the corresponding probability measures, it follows that the measures P_1 and P_2 are singular with respect to each other. Due to the theorem of singularity of fBms for different Hurst indices proposed by Praskasa Rao (2008), the continuous case was not included in this study.

Data Analysis

In this research, both estimation of the Hurst dimension using simulations and real data will be implemented. Descriptions of both parts are presented as follows.

Estimation of the Hurst Dimension Using Simulations

In this research, estimation of the Hurst dimension using simulations was implemented. The details with respect to computations were described. In order to answer the Research Question 1, we obtained the posterior distribution up to a normalizing constant and then using MCMC method we obtained the Bayesian estimate (posterior mean). Since there is no likelihood function for the continuous case, therefore, it is not possible to obtain the Hurst estimate for Research Question 2 in this research. For Research Question 3, different beta priors were utilized to investigate how sensitive the Hurst estimates would be. As for setting parameters of the Beta prior, the scenarios considered in this research were presented in Table 1. Furthermore, the Hurst dimension in the simulation program will include 0.1 up to 0.9 by the increment of 0.1.

Table 1

Parameters of Beta Distribution α and β in the Current Study

Scenario	α	β	Scenario	α	β	Scenario	α	β
1	0.1	0.1	11	1	0.1	21	3	0.1
2	0.1	0.5	12	1	0.5	22	3	0.5
3	0.1	1	13	1	1	22	3	1
4	0.1	2	14	1	2	24	3	2
5	0.1	3	15	1	3	25	3	3
6	0.5	0.1	16	2	0.1			
7	0.5	0.5	17	2	0.5			
8	0.5	1	18	2	1			
9	0.5	2	19	2	2			
10	0.5	3	20	2	3			

Since the Metropolis-Hastings algorithm of MCMC was used for computation in this study, formula (3.7) was utilized. It is assumed that the proposal distribution is the Beta distribution. Applying formula (3.7) in this research,

$$\left(\frac{f(h'|x)q(h|h')}{f(h|x)q(h'|h)}\right)$$

$$= \frac{\frac{1}{(2\pi)^{\frac{n}{2}}|\mathbf{R}|^{\frac{1}{2}}}\exp\{-\frac{1}{2}\mathbf{x}^{T}\mathbf{R}^{-1}\mathbf{x}\}\frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)}(h')^{\alpha-1}(1-h')^{\beta-1}}{\int_0^1 \frac{1}{(2\pi)^{\frac{n}{2}}|\mathbf{R}|^{\frac{1}{2}}}\exp\{-\frac{1}{2}\mathbf{x}^{T}\mathbf{R}^{-1}\mathbf{x}\}\frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)}h^{\alpha-1}(1-h)^{\beta-1}dh} \cdot \frac{\frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)}h^{\alpha-1}(1-h)^{\beta-1}}{\frac{1}{(2\pi)^{\frac{n}{2}}|\mathbf{R}|^{\frac{1}{2}}}\exp\{-\frac{1}{2}\mathbf{x}^{T}\mathbf{R}^{-1}\mathbf{x}\}\frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)}(h')^{\alpha-1}(1-h')^{\beta-1}}}{\int_0^1 \frac{1}{(2\pi)^{\frac{n}{2}}|\mathbf{R}|^{\frac{1}{2}}}\exp\{-\frac{1}{2}\mathbf{x}^{T}\mathbf{R}^{-1}\mathbf{x}\}\frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)}h^{\alpha-1}(1-h)^{\beta-1}dh}$$

Where $\mathbf{R}_{ij}=1/2(i^{2h}+j^{2h}-|i-j|^{2h})$.

Therefore, the formula was used for computation in the simulation for this research. Metropolis-Hasting algorithm was used for computation with a starting value of 0.1. The number of iterations was set to be 10,000. Monte Carlo Error was used to examine the convergence.

Estimation of the Hurst Dimension
Using Real Data

The data used in this research include the daily Taiwan Stock Index in the year of 2011. The dataset comprises of 247 data points. It is reasonable to use this dataset since only the discrete case will be considered in this research. For the estimation of the Hurst dimension using real data, it is necessary to estimate the mean and the variance. According to Lundahl et al. (1986), the variance is estimated by $\hat{\sigma}2 = \frac{x^T R^{-1} x}{n}$. Besides, the mean is estimated by $\hat{\mu} = \frac{t' R^{-1} X}{t' R^{-1} t}$ (Hu, Nualart, Xiao, & Zhang, 2011). Metropolis-Hasting algorithm was used for computation with a starting value of 0.1. The number of iterations was set to be 10,000. Monte Carlo Error was used to examine the convergence.

CHAPTER IV

RESULTS AND DISCUSSION

The purpose of this study was to estimate the Hurst dimension of a fBm using the Bayesian approach with Beta distribution as a prior. Additionally, this study was to examine how the choice of parameters of a Beta prior would affect the estimation. In order to achieve the goals, the following research questions were studied:

Q1 Can we find a Bayesian estimate for the Hurst dimension of a fBm with a Beta prior when the process is observed at discrete times?

Q2 Can we find a Bayesian estimate for the Hurst dimension of a fBm with a Beta prior when the process is observed at continuous times?

Q3 Will the Bayesian estimate for the Hurst dimension of a fBm vary when the parameters of the Beta prior change?

Q4 Can we develop a R code for questions 1 through 3?

To answer these research questions, this chapter is organized in the following manners. First, a Bayesian estimate for the Hurst dimension of a fBm with a Beta prior when the process is observed at discrete times is discussed. Second, the difficulty in finding a Bayesian estimate for the Hurst dimension of a fBm with a Beta prior when the process is observed at continuous times is suggested. Third, the Bayesian analysis with different parameters of Beta priors is illustrated. Finally, the R codes for Research Questions 1 and 3 are presented.

The likelihood function for fractional Brownian motion for the discrete case is as follows (Lundahl et al., 1986, p.155):

$$f(\mathbf{x}|\mathrm{H}) = \frac{1}{(2\pi)^{\frac{n}{2}}|\mathbf{R}|^{\frac{1}{2}}}\exp\{-\frac{1}{2}\mathbf{x}^{\mathrm{T}}\mathbf{R}^{-1}\mathbf{x}\}$$

where H is the Hurst parameter, $\mathbf{x}=(x_1, x_2, \ldots, x_n)$ and $\mathbf{R}_{ij}=1/2(i^{2h}+j^{2h}-|i-j|^{2h})$.

The distribution of Beta prior:

$$\mathrm{P(h)} = \frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)}\mathrm{h}^{\alpha-1}(1-\mathrm{h})^{\beta-1}$$

The full probability model is derived from the product of (3-5) and (3-6):

$$\mathrm{g}(\mathbf{x};\mathrm{h}) = \frac{1}{(2\pi)^{\frac{n}{2}}|\mathbf{R}|^{\frac{1}{2}}}\exp\{-\frac{1}{2}\mathbf{x}^{\mathrm{T}}\mathbf{R}^{-1}\mathbf{x}\}\cdot\frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)}\mathrm{h}^{\alpha-1}(1-\mathrm{h})^{\beta-1}$$

Therefore, the posterior distribution $\mathrm{k}(\mathrm{h}|\, x_1, x_2, \ldots, x_n)$ for research question 1 is shown as follows:

$$\mathrm{k}(\mathrm{h}|\, x_1, x_2, \ldots, x_n) = \frac{\mathrm{g}(\mathbf{x};\mathrm{h})}{\mathrm{m}(\mathbf{x})}$$

$$\mathrm{k}(\mathrm{h}|\mathrm{x}_1 \ldots \mathrm{x}_n) = \frac{\mathrm{g(x;h)}}{\mathrm{m(x)}} = \frac{\frac{1}{(2\pi)^{\frac{n}{2}}|\mathbf{R}|^{\frac{1}{2}}}\exp\{-\frac{1}{2}\mathbf{x}^{\mathrm{T}}\mathbf{R}^{-1}\mathbf{x}\}\frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)}\mathrm{h}^{\alpha-1}(1-\mathrm{h})^{\beta-1}}{\int_0^1\frac{1}{(2\pi)^{\frac{n}{2}}|\mathbf{R}|^{\frac{1}{2}}}\exp\{-\frac{1}{2}\mathbf{x}^{\mathrm{T}}\mathbf{R}^{-1}\mathbf{x}\}\frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)}\mathrm{h}^{\alpha-1}(1-\mathrm{h})^{\beta-1}\mathrm{dh}}\ .$$

Using Metropolis-Hastings algorithm of MCMC as well as the assumed proposal distribution of Beta distribution, the following equation was used for estimating the Hurst dimension of a fBm in this study.

$$\left(\frac{\mathrm{f(h'|x)q(h|h')}}{\mathrm{f(h|x)q(h'|h)}}\right)$$

$$= \frac{\frac{1}{(2\pi)^{\frac{n}{2}}|\mathbf{R}|^{\frac{1}{2}}}\exp\{-\frac{1}{2}\mathbf{x}^{\mathrm{T}}\mathbf{R}^{-1}\mathbf{x}\}\frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)}(\mathrm{h}')^{\alpha-1}(1-\mathrm{h}')^{\beta-1}}{\int_0^1\frac{1}{(2\pi)^{\frac{n}{2}}|\mathbf{R}|^{\frac{1}{2}}}\exp\{-\frac{1}{2}\mathbf{x}^{\mathrm{T}}\mathbf{R}^{-1}\mathbf{x}\}\frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)}\mathrm{h}^{\alpha-1}(1-\mathrm{h})^{\beta-1}\mathrm{dh}}}{\frac{\frac{1}{(2\pi)^{\frac{n}{2}}|\mathbf{R}|^{\frac{1}{2}}}\exp\{-\frac{1}{2}\mathbf{x}^{\mathrm{T}}\mathbf{R}^{-1}\mathbf{x}\}\frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)}\mathrm{h}^{\alpha-1}(1-\mathrm{h})^{\beta-1}}{\int_0^1\frac{1}{(2\pi)^{\frac{n}{2}}|\mathbf{R}|^{\frac{1}{2}}}\exp\{-\frac{1}{2}\mathbf{x}^{\mathrm{T}}\mathbf{R}^{-1}\mathbf{x}\}\frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)}\mathrm{h}^{\alpha-1}(1-\mathrm{h})^{\beta-1}\mathrm{dh}}}\cdot\frac{\frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)}\mathrm{h}^{\alpha-1}(1-\mathrm{h})^{\beta-1}}{\frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)}(\mathrm{h}')^{\alpha-1}(1-\mathrm{h}')^{\beta-1}}$$

For the continuous case, however, the probability measures generated by two different Hurst dimension processes are singular with respect to each other (Praskasa Rao, 2008), so it follows that there is no likelihood function for the continuous case. Using the theorem proposed by Kurchenko (2003),

$$\lim_{n\to\infty} \frac{1}{n}\sum_{m=0}^{n-1}[W_{Hi}(m) - 2W_{Hi}\left(m+\frac{1}{2}\right) + W_{Hi}(m+1)]^2 = V_2(0, Hi),\ i=1, 2.$$

Since $V_2(0,H_1) \neq V_2(0,H_2)$ if $H_1 \neq H_2$, and since the convergence stated above is convergence under the corresponding probability measures, it follows that the measures P_1 and P_2 are singular with respect to each other. As a result, the continuous case was not discussed in this study.

Simulation results are presented in the appendix (see Appendix A). Overall, the estimated H appears to be greater than the real H. Take Table 2 example, the real H is 0.1 while the estimated H appears to be greater than 0.1. Let us take Table 3 with real H of 0.5 and Table 4 with real H of 0.9 for instance, overestimation is observed though the overestimation is less severe as real H goes up. Due to the ease of readability, please see the rest of the scenarios in Appendix A.

Table 2

Simulation Results: True H=0.1

Real H	Alpha	Beta	Estimated H	Monte Carlo Error
0.1	0.1	0.1	0.834	0.0038
0.1	0.1	0.5	0.536	0.0069
0.1	0.1	1	0.403	0.0074
0.1	0.1	2	0.218	0.0052
0.1	0.1	3	0.248	0.0055
0.1	0.5	0.1	0.879	0.0024
0.1	0.5	0.5	0.622	0.0031
0.1	0.5	1	0.470	0.0033
0.1	0.5	2	0.333	0.0029
0.1	0.5	3	0.272	0.0024
0.1	1	0.1	0.918	0.0018
0.1	1	0.5	0.700	0.0027
0.1	1	1	0.546	0.0029
0.1	1	2	0.399	0.0023
0.1	1	3	0.321	0.0021
0.1	2	0.1	0.954	0.0011
0.1	2	0.5	0.800	0.0021
0.1	2	1	0.670	0.0023
0.1	2	2	0.511	0.0021
0.1	2	3	0.417	0.0020
0.1	3	0.1	0.966	0.0008
0.1	3	0.5	0.856	0.0017
0.1	3	1	0.751	0.0018
0.1	3	2	0.600	0.0020
0.1	3	3	0.504	0.0016

Table 3

Simulation Results: True H=0.5

Real H	Alpha	Beta	Estimated H	Monte Carlo Error
0.5	0.1	0.1	0.938	0.0015
0.5	0.1	0.5	0.786	0.0059
0.5	0.1	1	0.716	0.0077
0.5	0.1	2	0.638	0.0085
0.5	0.1	3	0.575	0.0088
0.5	0.5	0.1	0.944	0.0012
0.5	0.5	0.5	0.810	0.0025
0.5	0.5	1	0.720	0.0030
0.5	0.5	2	0.640	0.0045
0.5	0.5	3	0.600	0.0057
0.5	1	0.1	0.950	0.0010
0.5	1	0.5	0.830	0.0020
0.5	1	1	0.743	0.0027
0.5	1	2	0.651	0.0030
0.5	1	3	0.610	0.0040
0.5	2	0.1	0.964	0.0009
0.5	2	0.5	0.857	0.0016
0.5	2	1	0.769	0.0017
0.5	2	2	0.677	0.0021
0.5	2	3	0.630	0.0025
0.5	3	0.1	0.972	0.0006
0.5	3	0.5	0.880	0.0014
0.5	3	1	0.800	0.0015
0.5	3	2	0.699	0.0016
0.5	3	3	0.646	0.0019

Table 4

Simulation Results: True H=0.9

Real H	Alpha	Beta	Estimated H	Monte Carlo Error
0.9	0.1	0.1	0.800	0.0044
0.9	0.1	0.5	0.938	0.0027
0.9	0.1	1	0.909	0.0061
0.9	0.1	2	0.855	0.0130
0.9	0.1	3	0.822	0.0151
0.9	0.5	0.1	0.984	0.0004
0.9	0.5	0.5	0.943	0.0012
0.9	0.5	1	0.909	0.0033
0.9	0.5	2	0.858	0.0091
0.9	0.5	3	0.762	0.0202
0.9	1	0.1	0.985	0.0004
0.9	1	0.5	0.943	0.0010
0.9	1	1	0.913	0.0021
0.9	1	2	0.851	0.0118
0.9	1	3	0.816	0.0115
0.9	2	0.1	0.986	0.0003
0.9	2	0.5	0.947	0.0080
0.9	2	1	0.913	0.0013
0.9	2	2	0.876	0.0032
0.9	2	3	0.836	0.0083
0.9	3	0.1	0.980	0.0030
0.9	3	0.5	0.951	0.0007
0.9	3	1	0.919	0.0014
0.9	3	2	0.880	0.0033
0.9	3	3	0.830	0.0070

In addition, the estimated H decreases as Beta parameters go up given an Alpha value (see Appendix B). Take Figure 5 for example, given the true Hurst dimension of 0.1 and the assumed alpha parameter of the Beta prior distribution of 0.1, the estimated values of the Hurst dimension were 0.834, 0.536, 0.403, 0.218, and 0.248 when the values of beta parameter were assumed to be 0.1, 0.5,1, 2, and 3, respectively. This finding revealed that the estimated value of H decreases as Beta parameters increase given a fixed Alpha value. Similar patterns were found in Figure 6 through Figure 13.

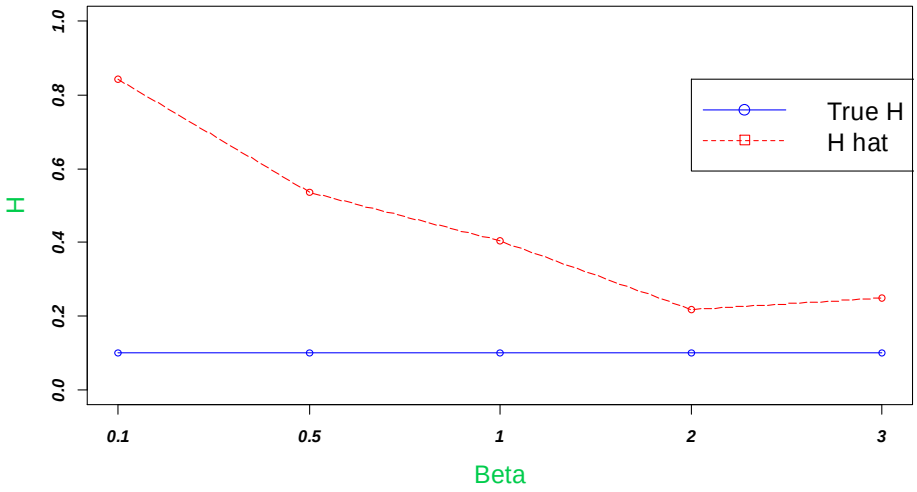

Figure 5. Changes in H Hat as β Increases Given True H=0.1, α = 0.1

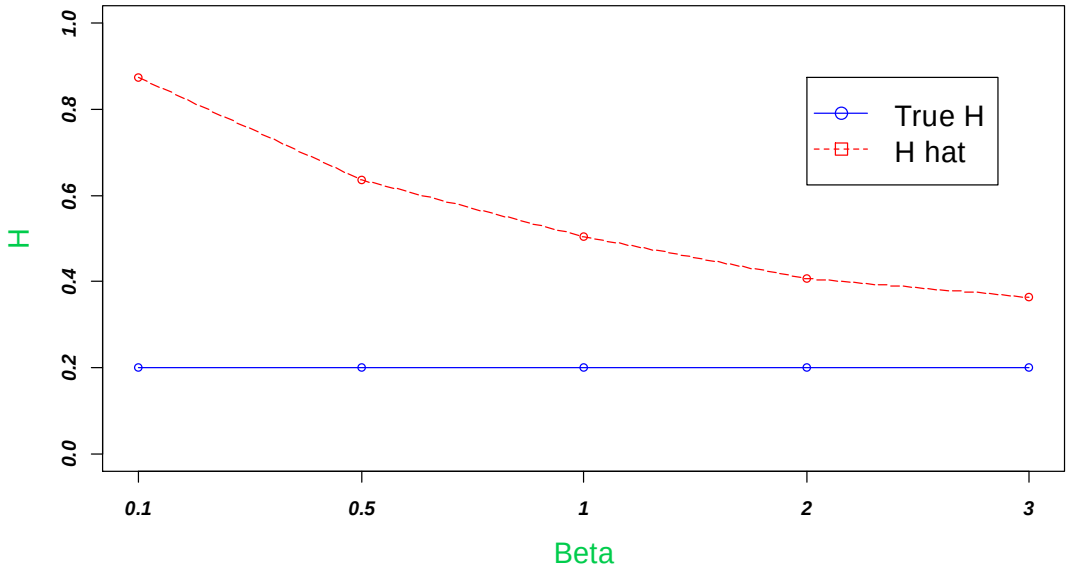

Figure 6. Changes in H Hat as β Increases Given True H=0.2, α = 0.1

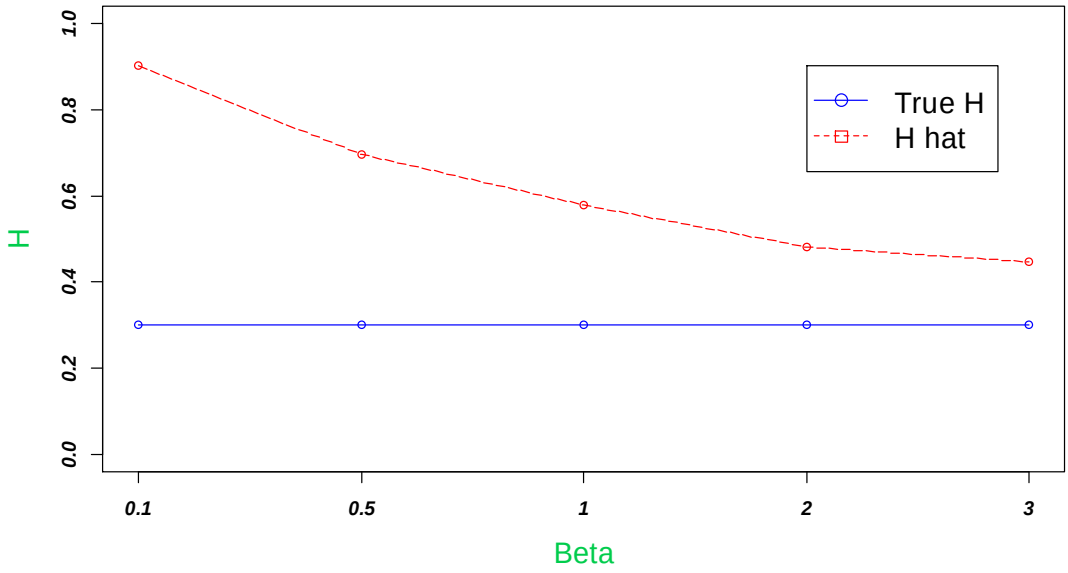

Figure 7. Changes in H Hat as β Increases Given True H=0.3, α = 0.1

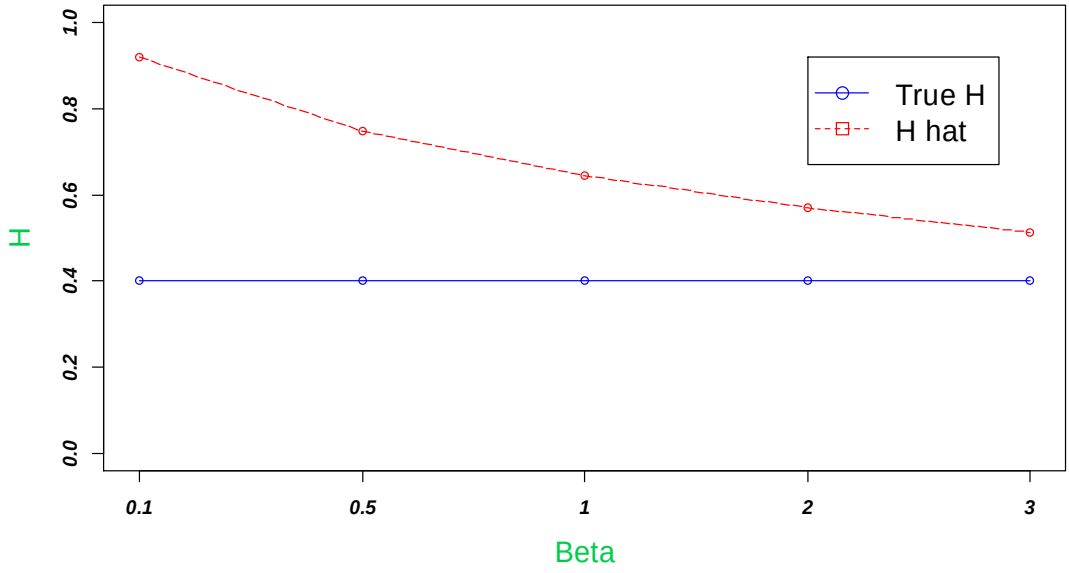

Figure 8. Changes in H Hat as β Increases Given True H=0.4, α = 0.1

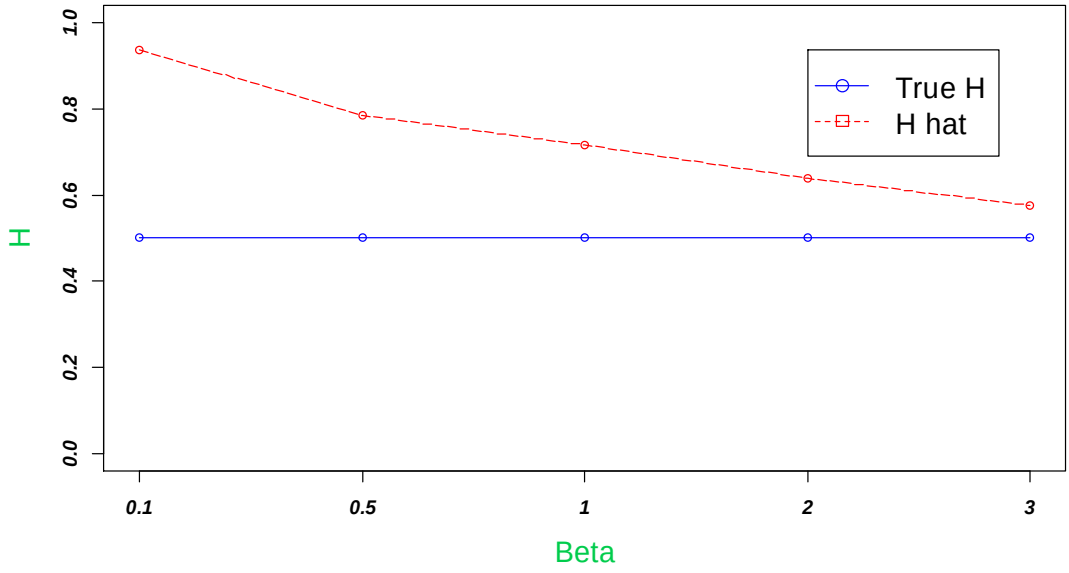

Figure 9. Changes in H Hat as β Increases Given True H=0.5, α = 0.1

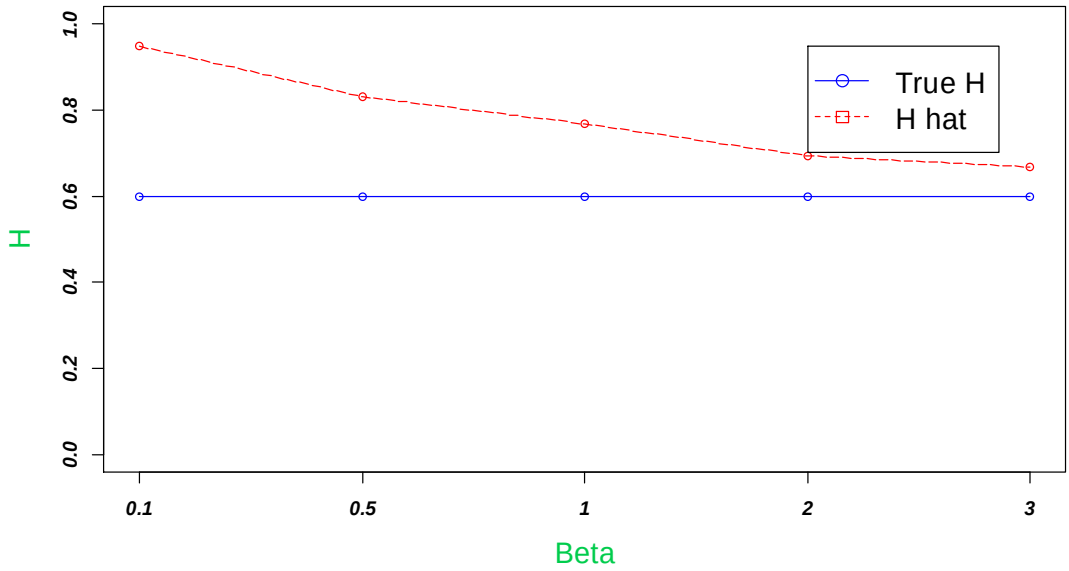

Figure 10. Changes in H Hat as β Increases Given True H=0.6, α = 0.1

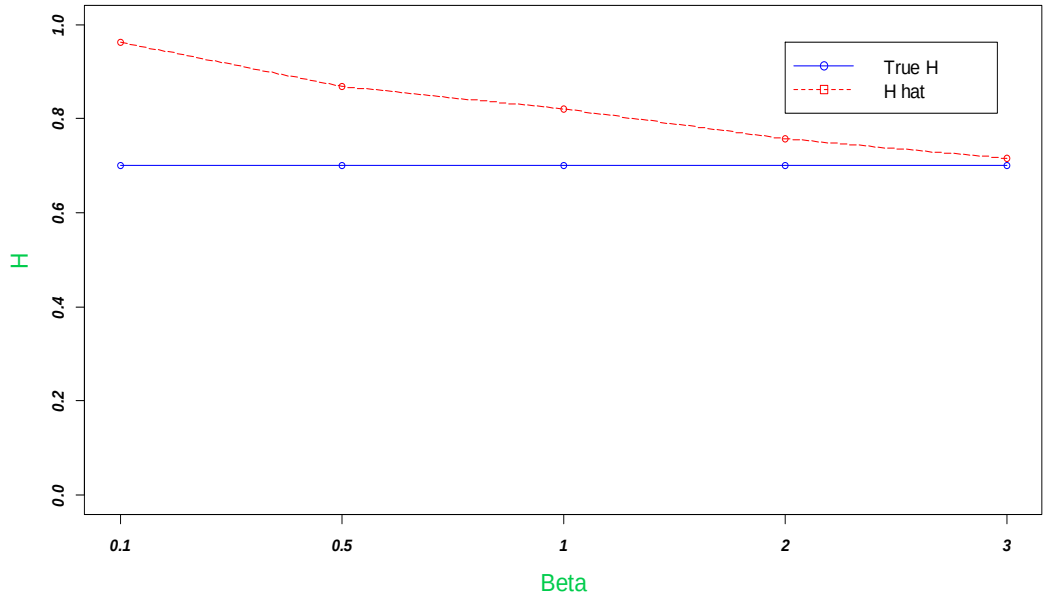

Figure 11. Changes in H Hat as β Increases Given True H=0.7, α = 0.1

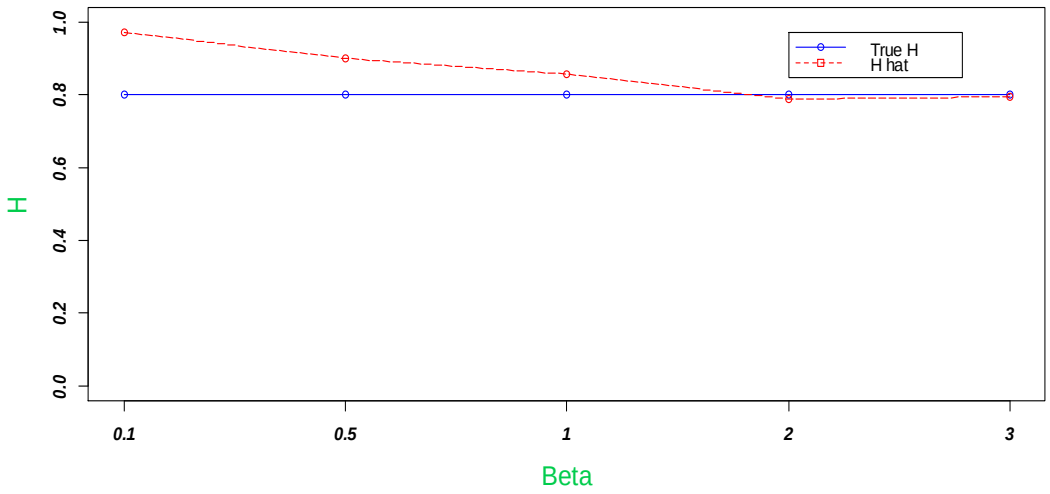

Figure 12. Changes in H Hat as β Increases Given True H=0.8, α = 0.1

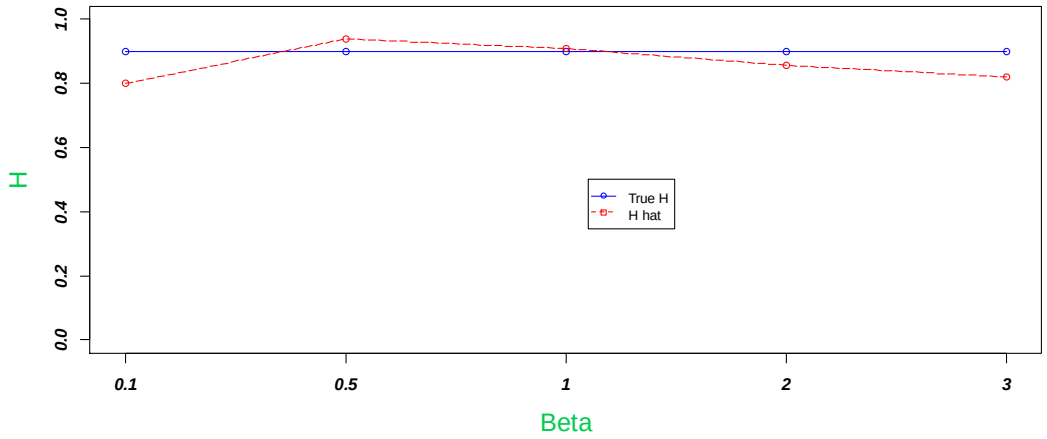

Figure 13. Changes in H Hat as β Increases Given True H=0.9, α = 0.1

In contrast, the estimated H increases as Alpha parameters go up given a Beta value (see Appendix B). Take Figure 14 for example, given the true Hurst dimension of 0.1 and the assumed beta parameter of the Beta prior distribution of 0.1, the estimated values of the Hurst dimension were 0.834, 0.879, 0.918, 0.954, and 0.966 when the values of alpha parameter were assumed to be 0.1, 0.5,1, 2, and 3, respectively. This finding revealed that the estimated value of H increases when alpha parameters increase given a fixed beta value. Similar patterns were found in Figure 14 through Figure 22. Research Question 3 was answered since the estimation results vary as Alpha and Beta parameters change. All the Monte Carlo errors appear to be close to 0, implying that results are precise.

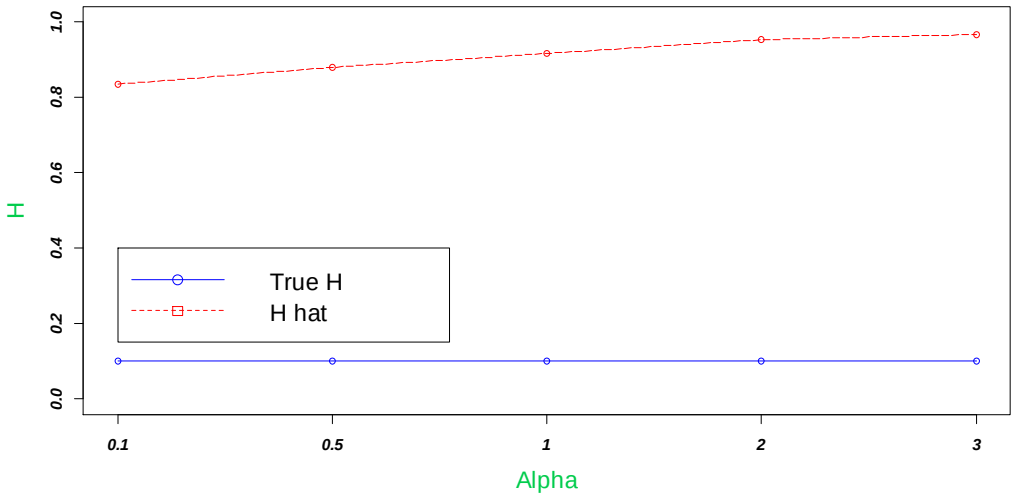

Figure 14. Change in H hat as α Increases Given True H= 0.1, β=0.1

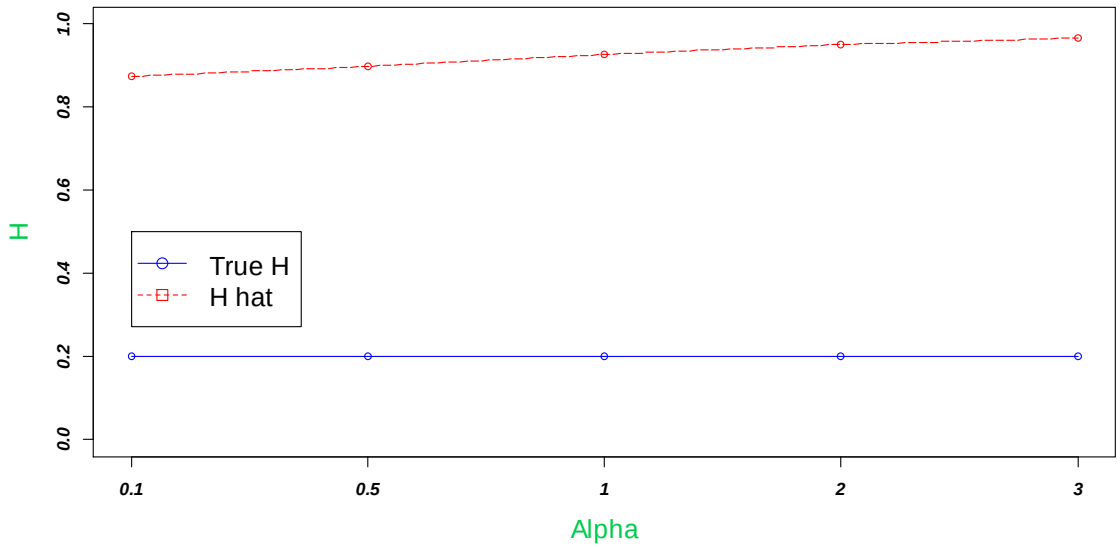

Figure 15. Change in H hat as α Increases Given True H=0.2, β=0.1

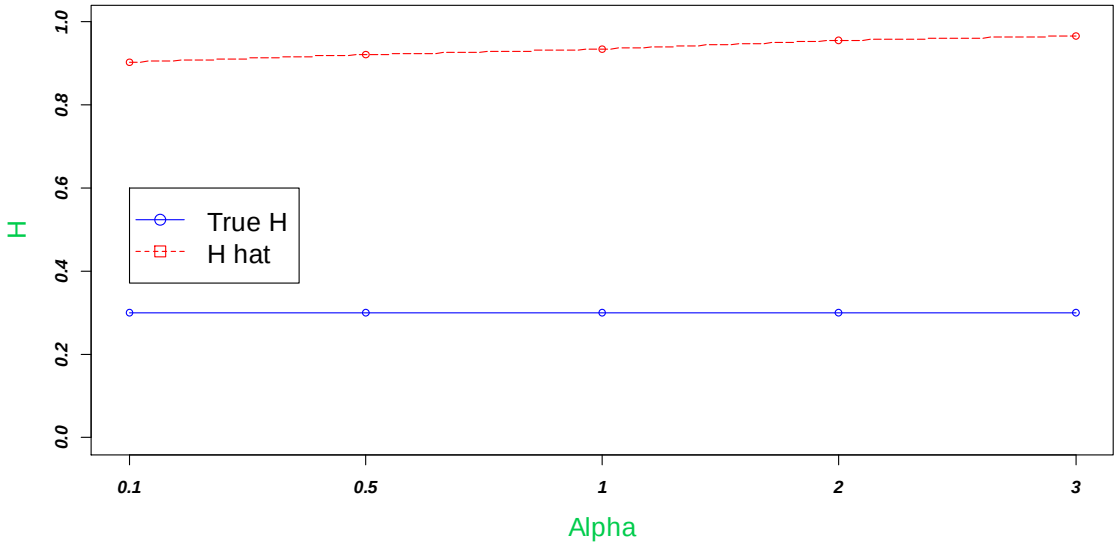

Figure 16. Change in H hat as α Increases Given True H=0.3, β=0.1

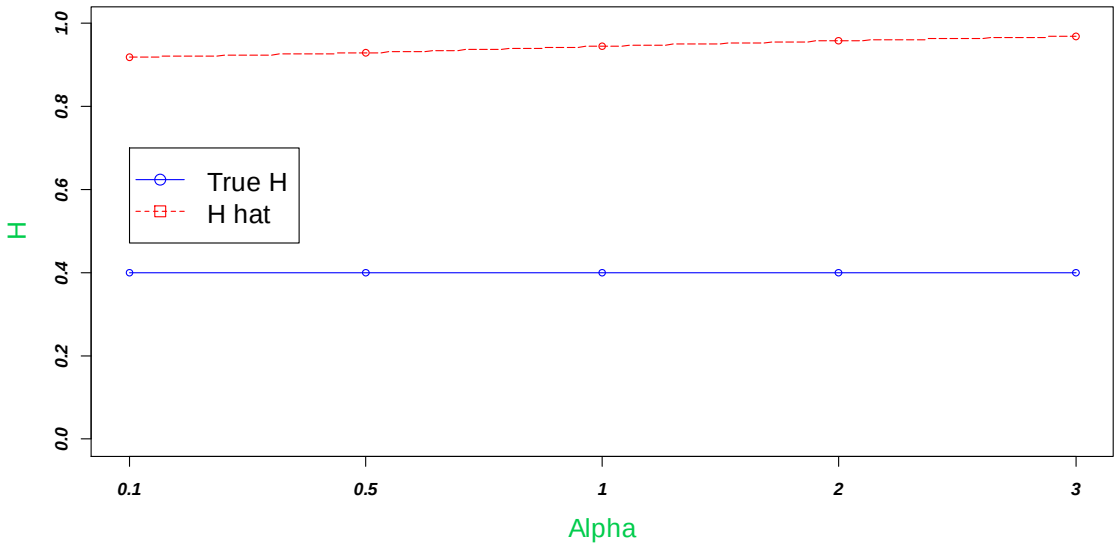

Figure 17. Change in H hat as α Increases Given True H=0.4, β=0.1

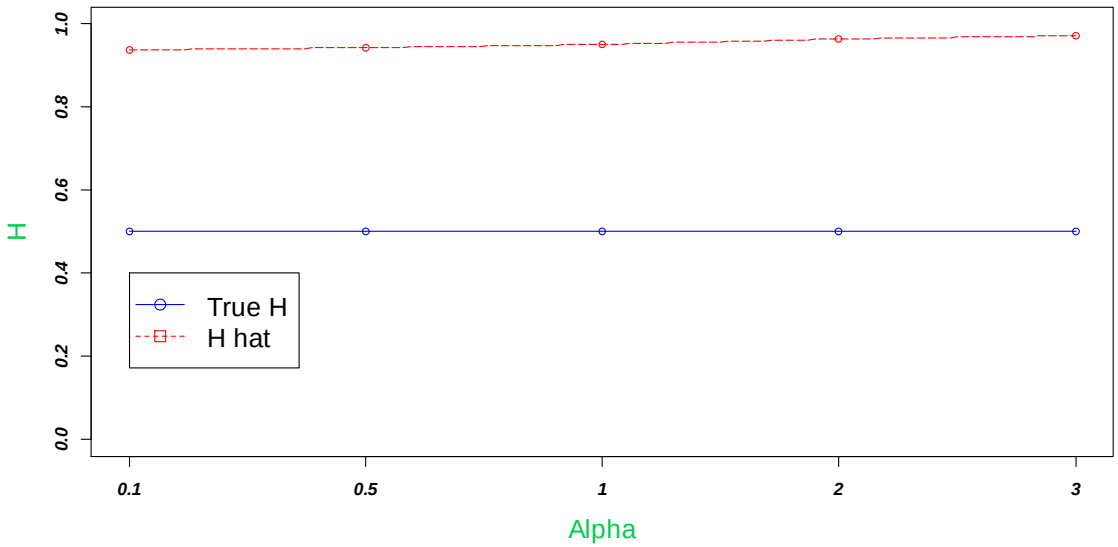

Figure 18. Change in H hat as α Increases Given True H=0.5, β=0.1

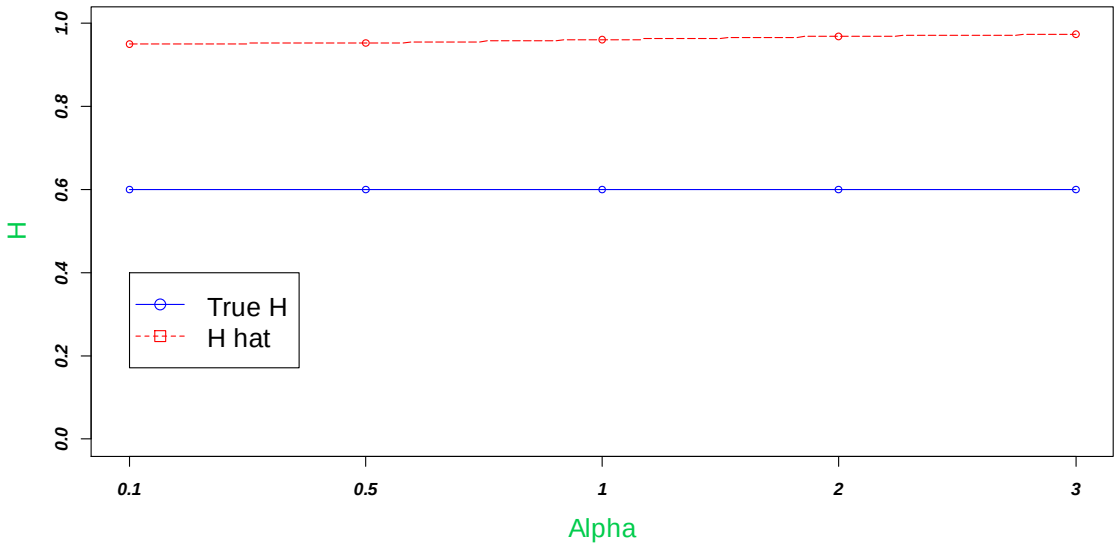

Figure 19. Change in H hat as α Increases Given True H=0.6, β=0.1

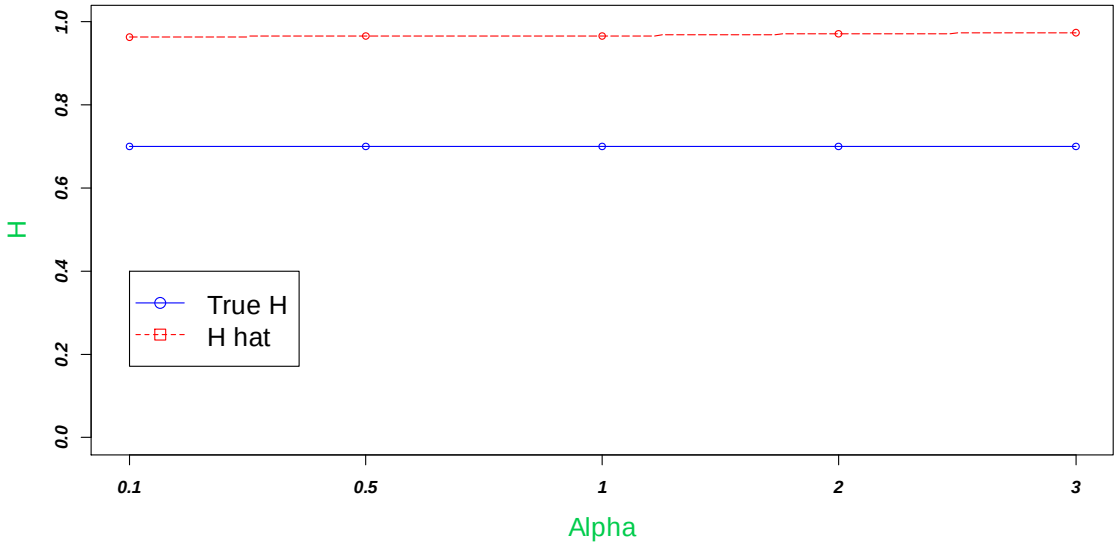

Figure 20. Change in H hat as α Increases Given True H=0.7, β=0.1

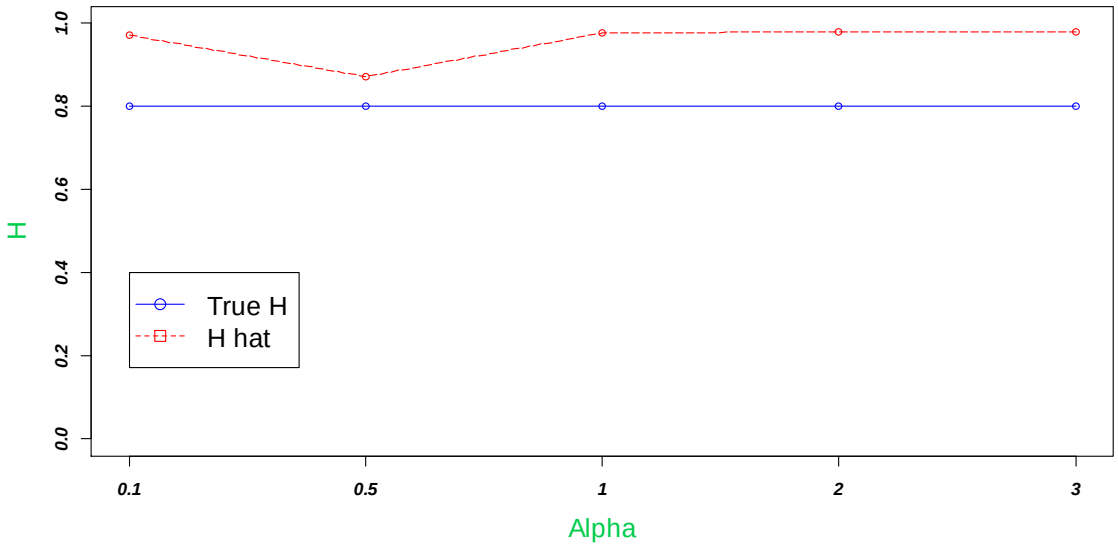

Figure 21. Change in H hat as α Increases Given True H=0.8, β=0.1

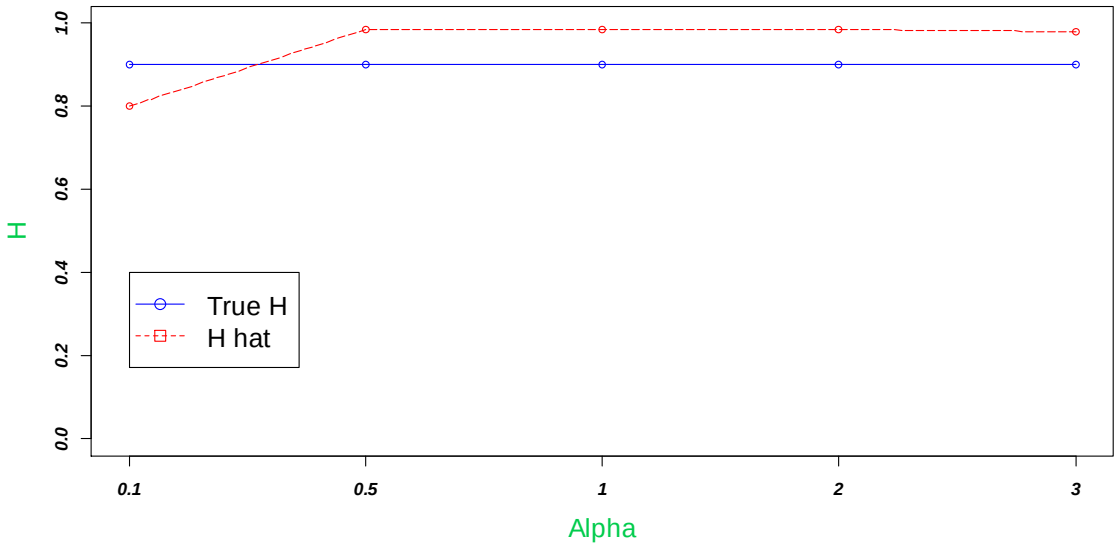

Figure 22. Change in H hat as α Increases Given True H=0.9, β=0.1

This study also sought to estimate the Hurst index for the real-world dataset obtained from the daily Taiwan Stock Index in the year of 2011. For the real-data case, the estimations of the variance and the mean of the process were necessary, which requires computation of the $\mathbf{R}_{ij}$. In order to compute $\mathbf{R}_{ij}$, the value of less than 0.5 for the Hurst index was assumed due to its down trend (see Figure B73). Specifically, 0.3 of the Hurst index was assumed for this study. Additionally, the values of Alpha and Beta of the Beta prior distribution were assumed to be 0.1 and 3 respectively since the estimated Hurst index from the simulation appeared to be close to the true Hurst index. The results revealed that the estimated Hurst index was 0.21 with a Monte Carlo error of 0.0025.

The R codes were successfully developed to implement the simulation in this study using a variety of packages such as “dvfBm,” “mnormt,” and “mcmcse.” The package of “dvfBm” referring to discrete variations of a fractional Brownian motion was developed by Coeurjolly (2009) to deal with the estimation of Hurst dimension of a fractional Brownian motion by using discrete variations methods in presence of outliers and/or an additive noise. In the package of “dvfBm,” simulation of a fractional Brownian motion by using the circulant matrix method known as “circFBM” was utilized to generate a discretized sample path of a fBm with Hurst parameter H in (0,1) by using the circulant matrix method.

The package of ‘mnormt’ standing for the multivariate normal and t distributions was developed by Azzalini (2012) to provide functions for computing the density and the distribution function of multivariate normal and multivariate ‘‘t’’ variates, and for generating random vectors sampled from these distributions. In the package of ‘mnormt,’

multivariate normal distribution known as “dmnorm” was used to generate observations from the multivariate normal (Gaussian) probability distribution.

The package of ‘mcmcse’ representing Monte Carlo Standard Errors for MCMC was developed by Flegal and Hughes (2012) to provide tools for computing Monte Carlo standard errors (MCSE) in Markov chain Monte Carlo (MCMC) settings. In the package of ‘mcmcse,’ the function of “mcse” was utilized to compute Monte Carlo standard errors for expectations. The whole R code programming is listed in Appendix C and Appendix D.

CHAPTER V

CONCLUSIONS AND RECOMMENDATIONS

Conclusions

The primary focus of this study was the estimation of the Hurst dimension of a fBm with a Beta prior when the process is observed at both discrete and continuous times. Furthermore, this study sought to investigate how sensitive the estimation would be to the choice of the parameters in the Beta distribution. Finally, developing the R codes for these research questions was another purpose. The conclusions of this study can be summarized as follows. The Bayesian estimate of Hurst dimension of a fBm with a Beta prior when the process is observed at discrete times was successfully developed. For the continuous case, however, the probability measures generated by two different Hurst dimension processes are singular with respect to each other (Praskasa Rao, 2008), so it follows that there is no likelihood function for the continuous case. It implies that we are unable to develop a Bayesian estimate of a fBm with a Beta prior when the process is observed at continuous times unless we find another dominating probability measure. As in Chapter IV, the estimated H appears to be greater than the real H. Overestimation is observed though the overestimation is less severe as real H increases. In addition, the estimated H decreases as Beta parameters increase given an Alpha value. In contrast, the estimated H increases as Alpha parameters increase given a Beta value. All the Monte Carlo errors appear to be close to 0, implying that results are precise. Moreover, the estimate of the Hurst index for the real-world dataset obtained from the daily Taiwan

Stock Index in the year of 2011 appeared to be 0.21. Meanwhile, the R codes were successfully developed to implement the simulation in this study using a variety of packages such as “dvfBm,” “mnormt,” and “mcmcse.”

Recommendations

In this study, the researcher has assumed that the proposal distribution is Beta distribution. In such a case, the assumption of the Beta proposal distribution led to the following formula

$$\left(\frac{f(h'|x)q(h|h')}{f(h|x)q(h'|h)}\right) = \frac{\frac{1}{(2\pi)^{\frac{n}{2}}|\mathbf{R}|^{\frac{1}{2}}}\exp\{-\frac{1}{2}\mathbf{x}^{\mathrm{T}}\mathbf{R}^{-1}\mathbf{x}\}\frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)}(h\prime)^{\alpha-1}(1-h\prime)^{\beta-1}}{\int_0^1 \frac{1}{(2\pi)^{\frac{n}{2}}|\mathbf{R}|^{\frac{1}{2}}}\exp\{-\frac{1}{2}\mathbf{x}^{\mathrm{T}}\mathbf{R}^{-1}\mathbf{x}\}\frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)}h^{\alpha-1}(1-h)^{\beta-1}dh} \cdot \frac{\frac{1}{(2\pi)^{\frac{n}{2}}|\mathbf{R}|^{\frac{1}{2}}}\exp\{-\frac{1}{2}\mathbf{x}^{\mathrm{T}}\mathbf{R}^{-1}\mathbf{x}\}\frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)}h^{\alpha-1}(1-h)^{\beta-1}}{\int_0^1 \frac{1}{(2\pi)^{\frac{n}{2}}|\mathbf{R}|^{\frac{1}{2}}}\exp\{-\frac{1}{2}\mathbf{x}^{\mathrm{T}}\mathbf{R}^{-1}\mathbf{x}\}\frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)}h^{\alpha-1}(1-h)^{\beta-1}dh} \cdot \frac{\frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)}h^{\alpha-1}(1-h)^{\beta-1}}{\frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)}(h\prime)^{\alpha-1}(1-h\prime)^{\beta-1}}.$$

Future studies may consider a different type of proposal distribution. Moreover, the researcher utilized Beta distribution to be the prior distribution. Future studies may choose a different prior distribution such as uniform distribution to see if interesting estimation outcomes occur. One thing needs to be noted is that the estimation of the Hurst dimension tends to be over estimated. Future studies may consider different algorithm in computing the MCMC process. Overall, all the research questions have been successfully answered except for the continuous case. Future studies may consider overcoming such an issue. Finally, this study focused on the simulated data. Future studies may consider estimating the Hurst dimension using different types of real-world data.

REFERENCES

Achard, S., & Coeurjolly, J. F. (2010). Discrete variations of the fractional Brownian motion in the presence of outliers and an additive noise. *Statistical Surveys, 4*, 117-147.

Azzalini, A. (2012). *Package 'mnormt.'* Retrieved February 1, 2013 from http://cran.r-project.org/web/packages/mnormt/mnormt.pdf

Bain, L. J., & Engelhardt, M. (1992). *Introduction to probability and mathematical statistics* (2nd ed.). Pacific Grove, CA: Thomson Learning.

Bayraktar, E., Poor, H. V., & Sicar, K. R. (2004). Estimating the fractal dimension of the S&P 500 index using wavelet analysis. *International Journal of Theoretical and Applied Finance, 7*(5), 1-26.

Berzin, C., & Leon, J. (2007). Estimating the Hurst parameter. *Statistical Inference for Stochastic Processes, 10*, 49-73.

Biagini, F., Hu, Y., Øksendal, B., & Zhang, T. (2010). *Stochastic calculus for fractional Brownian motion and applications*. Berlin: Springer.

Bishwal, J. P. N. (2003). Maximum likelihood estimation in partially observed stochastic differential system driven by a fractional Brownian motion. *Stochastic Analysis and Applications, 21*(5), 995-1007.

Breton, A. L. (1998). Filtering and parameter estimation in a simple linear system driven by a fractional Brownian motion. *Statistics and Probability Letters, 38*, 263-274.

Chatfield, C. (2004). *The analysis of time series: An introduction* (6th ed.). London, England: Chapman and Hall.

Cheridito, P. (2003). Arbitrage in Fractional Brownian Motion Models. *Finance and Stochastics, 7*, 533-553.

Chiang, P. J. (2006). *A study on the estimation of the parameter and goodness of fit test for the self-similar process.* Unpublished Master Thesis, National Sun Yat-sen University, Kaohsiung, Taiwan.

Coeurjolly, J. (2009). *Package 'dvfBm.'* Retrieved February 1, 2013 from http://cran.r-project.org/web/packages/dvfBm/dvfBm.pdf

Constantine, A. G., & Hall, P. (1994). Characterizing surface smoothness via estimation of effective fractal dimension. *Journal of the Royal Statistical Society, 56*, 97-113.

Dahlhaus, R. (1989). Efficient estimation for self-similar process. *The Annals of Statistics, 17*(4), 1749-1766.

Daye, Z. J. (2003). *Introduction to fractional Brownian motion in finance*. Retrived January 15, 2010 from mpra.ub.uni-muenchen.de/9146/.

Dieker, T. (2004). *Simulation of fractional Brownian motion*, University of Twente, Enschede.

Es-Sebaiy, K., Ouassou, I., & Ouknine, Y. (2009). Estimation of the drift of fractional Brownian motion. *Statistics and Probability Letters*, *79*, 1647-1653.

Flegal, J. M., & Hughes, J. (2012). *Package 'mcmcse.'* Retrieved February 1, 2013 from http://cran.r-project.org/web/packages/mcmcse/mcmcse.pdf

"Fractals and the Fractal Dimension" (2010). Wikipedia. Retrieved October 11, 2010 from http://www.vanderbilt.edu/AnS/psychology/cogsci/chaos/workshop/Fractals.html

"Fractal Dimension" (2010). Wikipedia. Retrieved October 11, 2010 from http://en.wikipedia.org/wiki/Fractal_dimension

Gelman, A., Carlin, J. B., Stern, H. S., & Rubin, D. B. (2004). *Bayesian data analysis* (2nd ed.). New York, NY: Chapman & Hall/CRC.

Higuchi, T. (1988). Approach to an irregular time series on the basis of fractal theory. *Physica D*, *31*, 277-283.

Hu, Y., Nualart, D., Xiao, W., & Zhang, W. (2011). Exact maximum likelihood estimator for drift fractional Brownian motion at discrete observation. *Acta Mathematica Scientia 2011, 31B*(5), 1851-1859.

Jones, O. D., & Shen, Y. (2004). Estimating the Hurst index of a self-similar process via the crossing tree. *IEEE Signal Processing Letters, 11*(4), 416-419.

Kurchenko, O. O. (2003) A consistent estimator of the Hurst parameter for a fractional Brownian motion, *Theory of Probability and Mathematical Statistics*, 67, 97-106.

Lundahl, T., Ohley, W. J., Kay, S. M., & Siffert, R. (1986). Fractional Brownian motion: A maximum likelihood estimator and its application to image texture. *IEEE Transaction on Medical Image, 5*(3), 152-161.

Mandelbrot, B. B. (1997). *Fractals and scaling in finance: Discontinuity, concentration, risk.* New York,NY: Springer-Verlag.

Mandelbrot, B. B., & Van Ness, J. W. (1968). Fractional Brownian motions, fractional noises and applications. *SIAM Review, 10*, 422-437.

Mazo, R. (2008). *Brownian motion: Fluctuation, dynamics, and applications.* Oxford: Oxford University Press.

Mörters, P., & Peres, Y. (2008). *Brownian motion.* Retrived January 15, 2010 from www.stat.berkeley.edu/~peres/bmbook.pdf

Ntzoufras, I. (2009). *Bayesian modeling using WinBUGS*. New Jersey: Wiley.

Peitgen, H. O., Jurgens, H., & Saupe, D. (1992). *Chaos and fractals: New frontiers of science.* New York, NY: Springer-Verlag.

Praskasa Rao, B. L. S. (2004). Self-similar process, fractional Brownian motion and statistical inference. *Lecture Notes-Monograph Series, 45,* 98-125.

Praskasa Rao, B. L. S. (2008). Singularity of fractional Brownian motions with different Hurst indices. *Stochastic Analysis and Applications, 26*(2), 33B337.

Rao, P. (2004). Self-similar process, fractional Brownian motion and statistical inference. *Lecture Notes-Monograph Series, 45*, 98-125.

Rogers, C. (1997). Arbitrage with fractional Brownian motion. *Mathematical Finance, 7,* 95-105.

Rossi, P. E., Allenby G. M., & McCulloch, R. (2006). *Bayesian statistics and marketing.* Hoboken, NJ: John Wiley & Sons.

Scrumerati, (2009). Scrum self-similarity: Creating organizational fractals. Retrived March 12, 2011 from http://scrumerati.com/2009/05/scrum-fractals.html#tp

Wei, W. W. S. (2006). *Time series analysis: Univariate and multivariate methods.* Boston, MD: Addison-Wesley.

APPENDIX A

TABLES

Table 5

Simulation Results: True H=0.2

Real H	Alpha	Beta	Estimated H	Monte Carlo Error
0.2	0.1	0.1	0.874	0.0040
0.2	0.1	0.5	0.635	0.0079
0.2	0.1	1	0.504	0.0790
0.2	0.1	2	0.407	0.0086
0.2	0.1	3	0.363	0.0077
0.2	0.5	0.1	0.898	0.0019
0.2	0.5	0.5	0.678	0.0038
0.2	0.5	1	0.543	0.0037
0.2	0.5	2	0.428	0.0032
0.2	0.5	3	0.371	0.0037
0.2	1	0.1	0.927	0.0017
0.2	1	0.5	0.733	0.0025
0.2	1	1	0.598	0.0026
0.2	1	2	0.455	0.0026
0.2	1	3	0.399	0.0023
0.2	2	0.1	0.952	0.0011
0.2	2	0.5	0.811	0.0021
0.2	2	1	0.686	0.0022
0.2	2	2	0.539	0.0019
0.2	2	3	0.455	0.0022
0.2	3	0.1	0.968	0.0008
0.2	3	0.5	0.859	0.0016
0.2	3	1	0.756	0.0019
0.2	3	2	0.610	0.0019
0.2	3	3	0.520	0.0019

Table 6

Simulation Results: True H=0.3

Real H	Alpha	Beta	Estimated H	Monte Carlo Error
0.3	0.1	0.1	0.904	0.0030
0.3	0.1	0.5	0.697	0.0058
0.3	0.1	1	0.580	0.0076
0.3	0.1	2	0.481	0.0091
0.3	0.1	3	0.448	0.0080
0.3	0.5	0.1	0.922	0.0018
0.3	0.5	0.5	0.728	0.0034
0.3	0.5	1	0.611	0.0032
0.3	0.5	2	0.502	0.0039
0.3	0.5	3	0.459	0.0040
0.3	1	0.1	0.936	0.0013
0.3	1	0.5	0.767	0.0022
0.3	1	1	0.646	0.0027
0.3	1	2	0.529	0.0027
0.3	1	3	0.468	0.0028
0.3	2	0.1	0.957	0.0011
0.3	2	0.5	0.824	0.0017
0.3	2	1	0.711	0.0018
0.3	2	2	0.584	0.0023
0.3	2	3	0.507	0.0020
0.3	3	0.1	0.968	0.0007
0.3	3	0.5	0.861	0.0014
0.3	3	1	0.765	0.0017
0.3	3	2	0.637	0.0019
0.3	3	3	0.550	0.0017

Table 7

Simulation Results: True H=0.4

Real H	Alpha	Beta	Estimated H	Monte Carlo Error
0.4	0.1	0.1	0.920	0.0029
0.4	0.1	0.5	0.749	0.0067
0.4	0.1	1	0.645	0.0076
0.4	0.1	2	0.570	0.0083
0.4	0.1	3	0.512	0.0090
0.4	0.5	0.1	0.933	0.0014
0.4	0.5	0.5	0.772	0.0026
0.4	0.5	1	0.669	0.0037
0.4	0.5	2	0.577	0.0041
0.4	0.5	3	0.534	0.0040
0.4	1	0.1	0.945	0.0014
0.4	1	0.5	0.791	0.0024
0.4	1	1	0.692	0.0026
0.4	1	2	0.592	0.0028
0.4	1	3	0.533	0.0031
0.4	2	0.1	0.960	0.0009
0.4	2	0.5	0.838	0.0015
0.4	2	1	0.736	0.0020
0.4	2	2	0.625	0.0018
0.4	2	3	0.567	0.0020
0.4	3	0.1	0.970	0.0008
0.4	3	0.5	0.867	0.0014
0.4	3	1	0.777	0.0015
0.4	3	2	0.664	0.0018
0.4	3	3	0.594	0.0018

Table 8

Simulation Results: True H=0.6

Real H	Alpha	Beta	Estimated H	Monte Carlo Error
0.6	0.1	0.1	0.950	0.0020
0.6	0.1	0.5	0.831	0.0052
0.6	0.1	1	0.767	0.0077
0.6	0.1	2	0.694	0.0101
0.6	0.1	3	0.668	0.0093
0.6	0.5	0.1	0.955	0.0009
0.6	0.5	0.5	0.847	0.0022
0.6	0.5	1	0.769	0.0029
0.6	0.5	2	0.702	0.0049
0.6	0.5	3	0.665	0.0063
0.6	1	0.1	0.961	0.0008
0.6	1	0.5	0.857	0.0019
0.6	1	1	0.784	0.0021
0.6	1	2	0.712	0.0031
0.6	1	3	0.680	0.0041
0.6	2	0.1	0.969	0.0007
0.6	2	0.5	0.876	0.0014
0.6	2	1	0.804	0.0019
0.6	2	2	0.729	0.0018
0.6	2	3	0.683	0.0025
0.6	3	0.1	0.974	0.0006
0.6	3	0.5	0.893	0.0014
0.6	3	1	0.821	0.0015
0.6	3	2	0.741	0.0018
0.6	3	3	0.695	0.0018

Table 9

Simulation Results: True H=0.7

Real H	Alpha	Beta	Estimated H	Monte Carlo Error
0.7	0.1	0.1	0.963	0.0013
0.7	0.1	0.5	0.869	0.0045
0.7	0.1	1	0.820	0.0060
0.7	0.1	2	0.757	0.0080
0.7	0.1	3	0.716	0.0168
0.7	0.5	0.1	0.967	0.0007
0.7	0.5	0.5	0.878	0.0020
0.7	0.5	1	0.822	0.0035
0.7	0.5	2	0.772	0.0052
0.7	0.5	3	0.724	0.0047
0.7	1	0.1	0.967	0.0008
0.7	1	0.5	0.887	0.0015
0.7	1	1	0.825	0.0024
0.7	1	2	0.768	0.0032
0.7	1	3	0.739	0.0056
0.7	2	0.1	0.972	0.0006
0.7	2	0.5	0.898	0.0009
0.7	2	1	0.841	0.0017
0.7	2	2	0.780	0.0021
0.7	2	3	0.743	0.0027
0.7	3	0.1	0.976	0.0006
0.7	3	0.5	0.911	0.0010
0.7	3	1	0.852	0.0012
0.7	3	2	0.786	0.0016
0.7	3	3	0.754	0.0021

Table 10

Simulation Results: True H=0.8

Real H	Alpha	Beta	Estimated H	Monte Carlo Error
0.8	0.1	0.1	0.973	0.0010
0.8	0.1	0.5	0.900	0.0032
0.8	0.1	1	0.857	0.0061
0.8	0.1	2	0.789	0.0119
0.8	0.1	3	0.794	0.0095
0.8	0.5	0.1	0.871	0.0027
0.8	0.5	0.5	0.908	0.0019
0.8	0.5	1	0.866	0.0031
0.8	0.5	2	0.812	0.0056
0.8	0.5	3	0.784	0.0103
0.8	1	0.1	0.977	0.0005
0.8	1	0.5	0.916	0.0013
0.8	1	1	0.873	0.0018
0.8	1	2	0.815	0.0036
0.8	1	3	0.780	0.0084
0.8	2	0.1	0.979	0.0004
0.8	2	0.5	0.920	0.0010
0.8	2	1	0.878	0.0014
0.8	2	2	0.826	0.0028
0.8	2	3	0.787	0.0041
0.8	3	0.1	0.981	0.0004
0.8	3	0.5	0.928	0.0010
0.8	3	1	0.883	0.0012
0.8	3	2	0.833	0.0018
0.8	3	3	0.805	0.0028

APPENDIX B

FIGURES

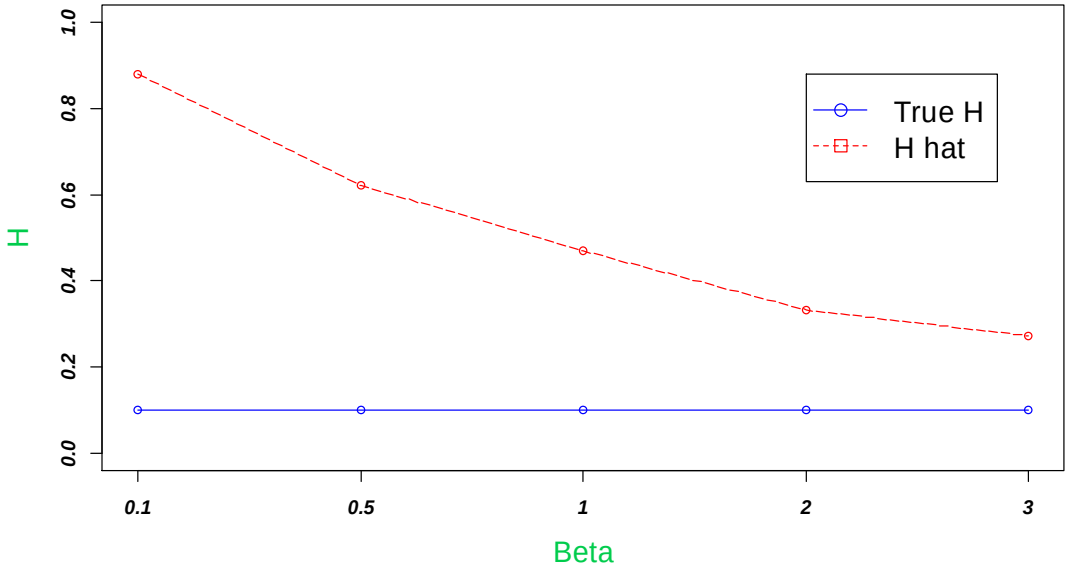

Figure 23. Changes in H Hat as β Increases Given True H=0.1, α = 0.5

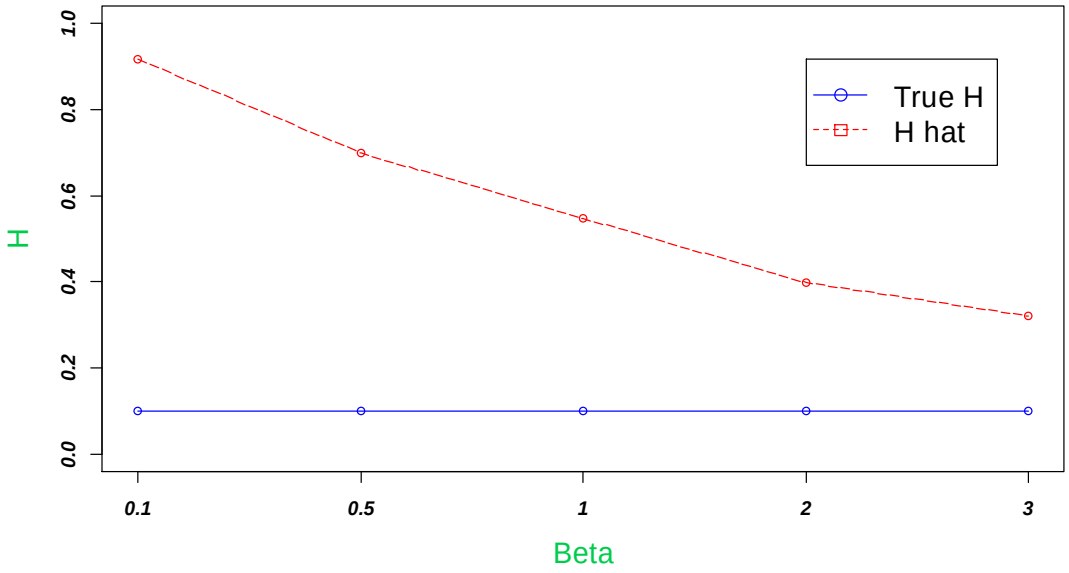

Figure24. Changes in H Hat as β Increases Given True H=0.1, α = 1

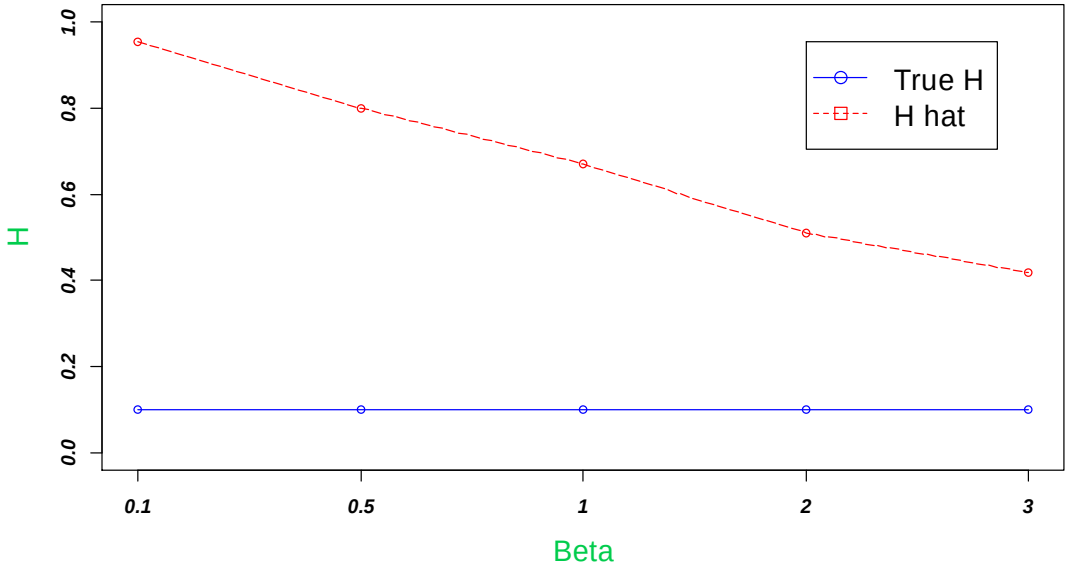

Figure 25. Changes in H Hat as β Increases Given True H=0.1, α = 2

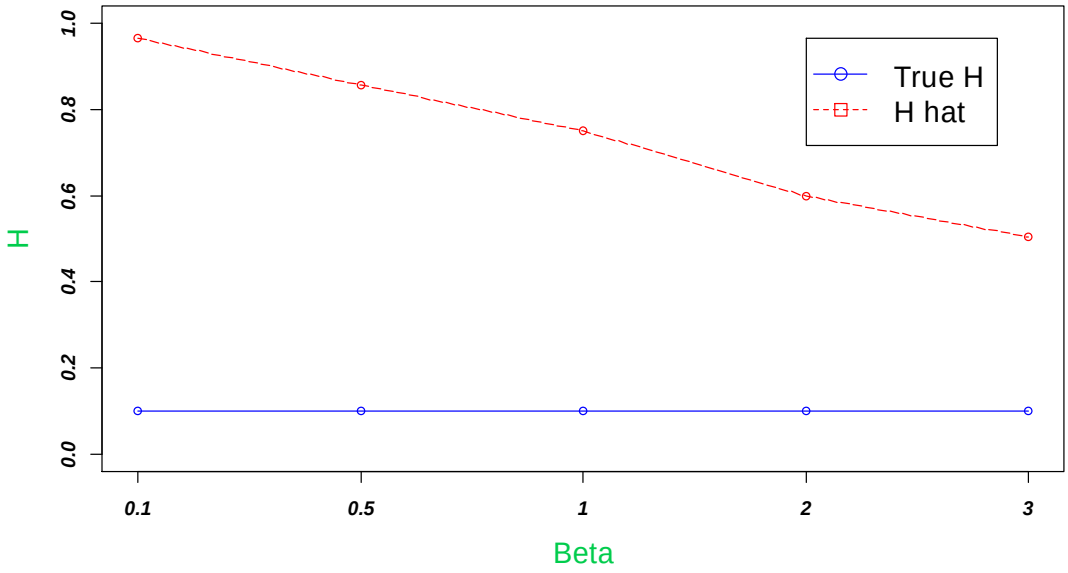

Figure 26. Changes in H Hat as β Increases Given True H=0.1, α = 3

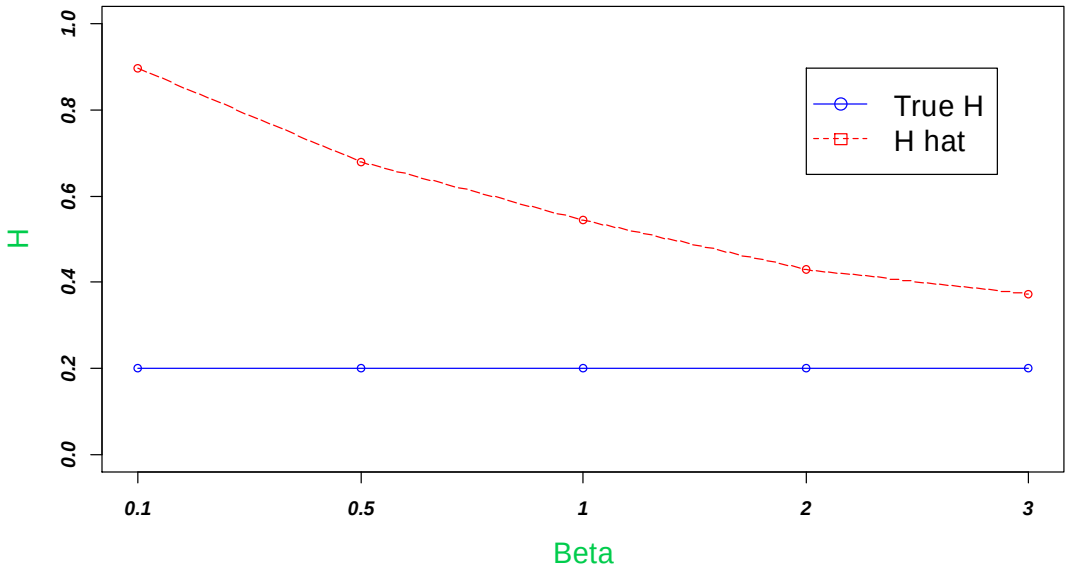

Figure 27. Changes in H Hat as β Increases Given True H=0.2, α = 0.5

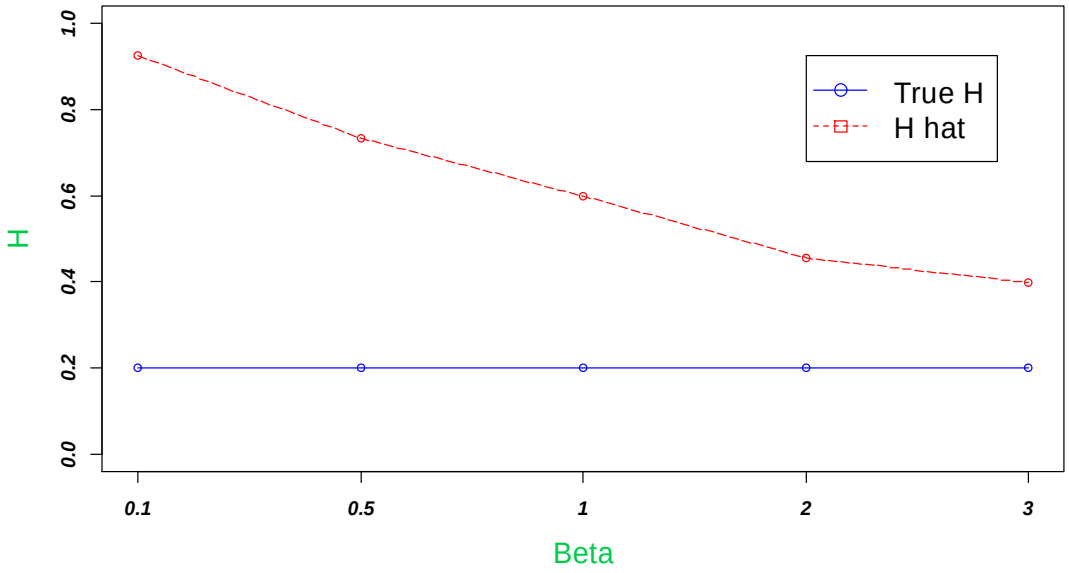

Figure 28. Changes in H Hat as β Increases Given True H=0.2, α = 1

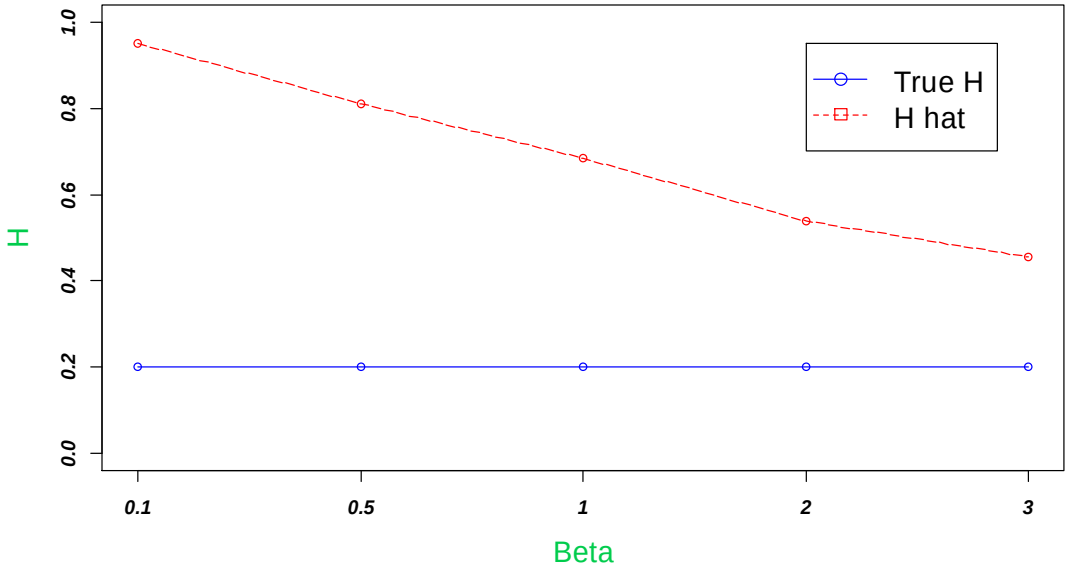

Figure 29. Changes in H Hat as β Increases Given True H=0.2, α = 2

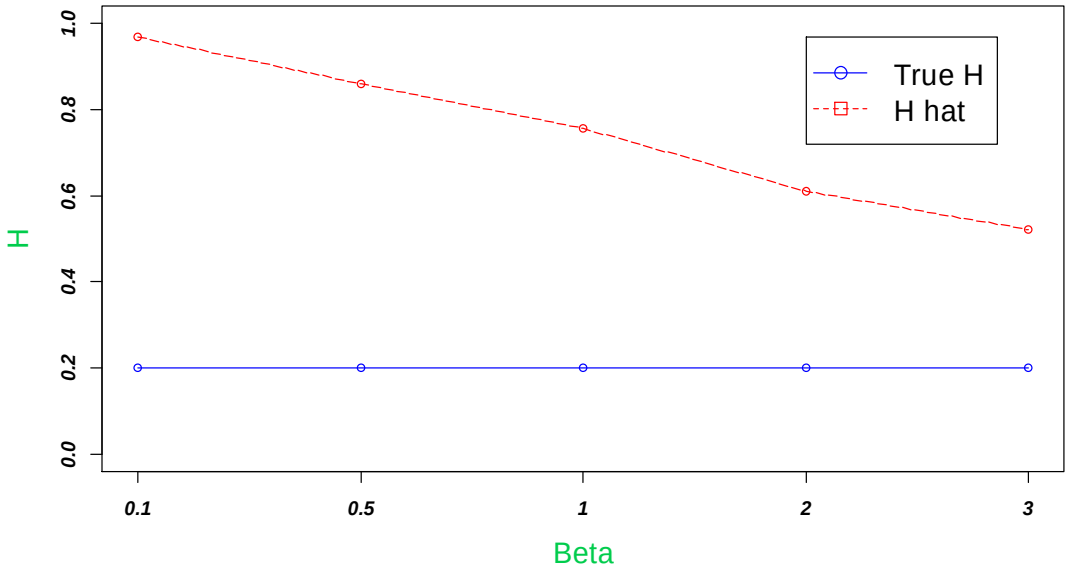

Figure 30. Changes in H Hat as β Increases Given True H=0.2, α = 3

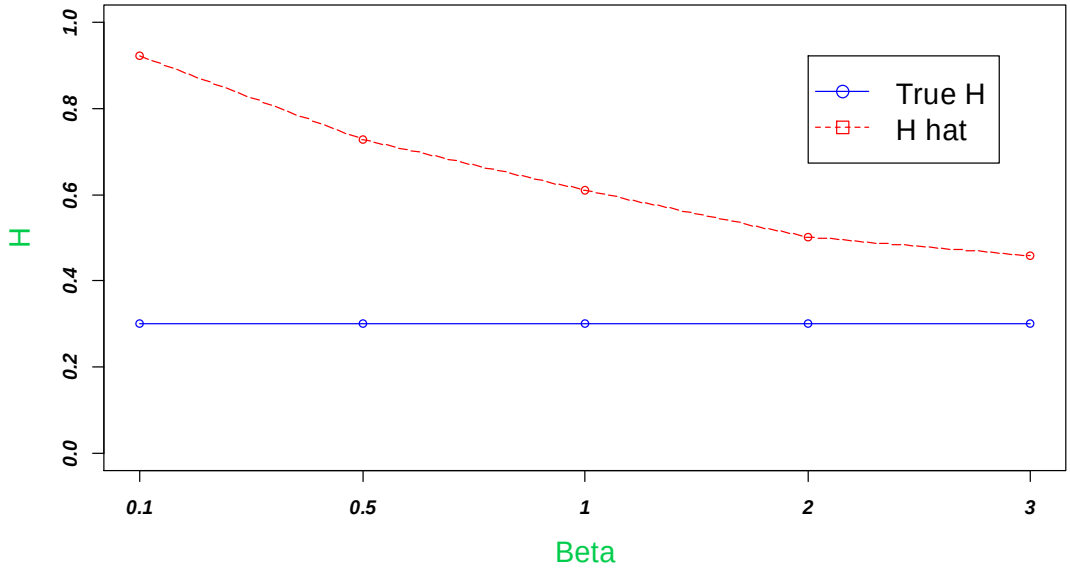

Figure 31. Changes in H Hat as β Increases Given True H=0.3, α = 0.5

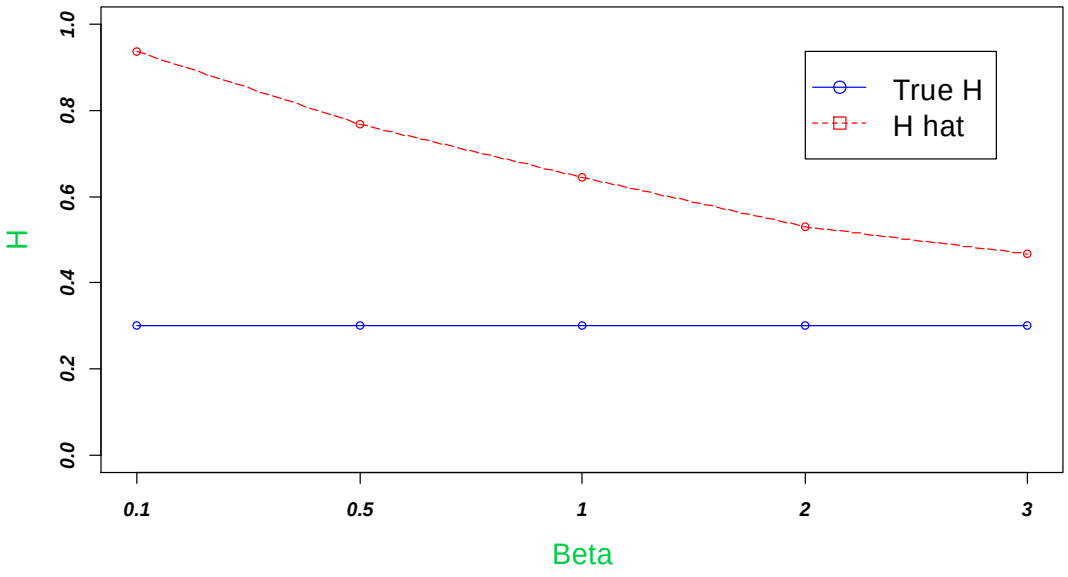

Figure 32. Changes in H Hat as β Increases Given True H=0.3, α = 1

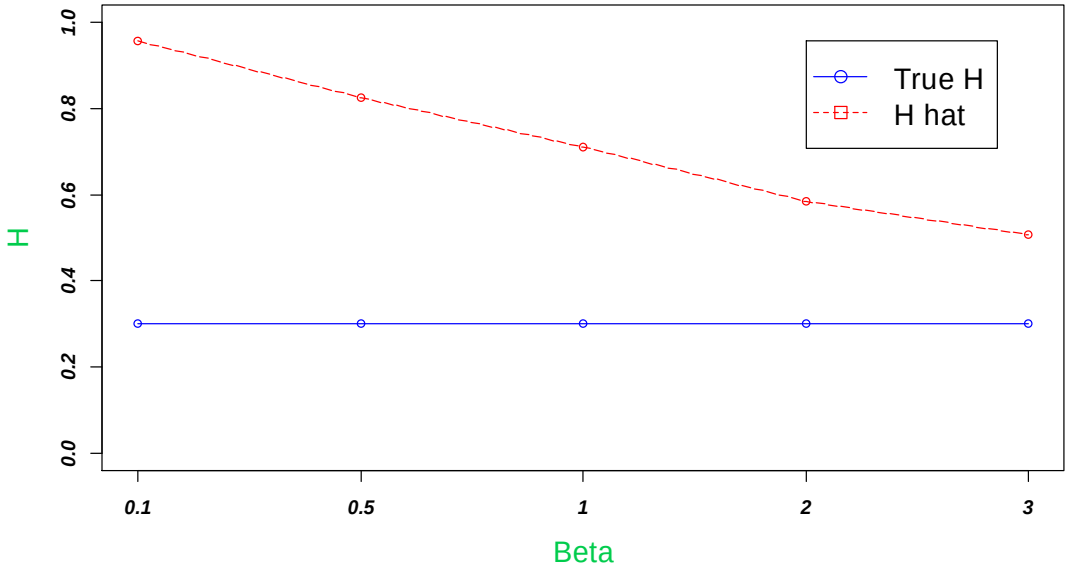

Figure 33. Changes in H Hat as β Increases Given True H=0.3, α = 2

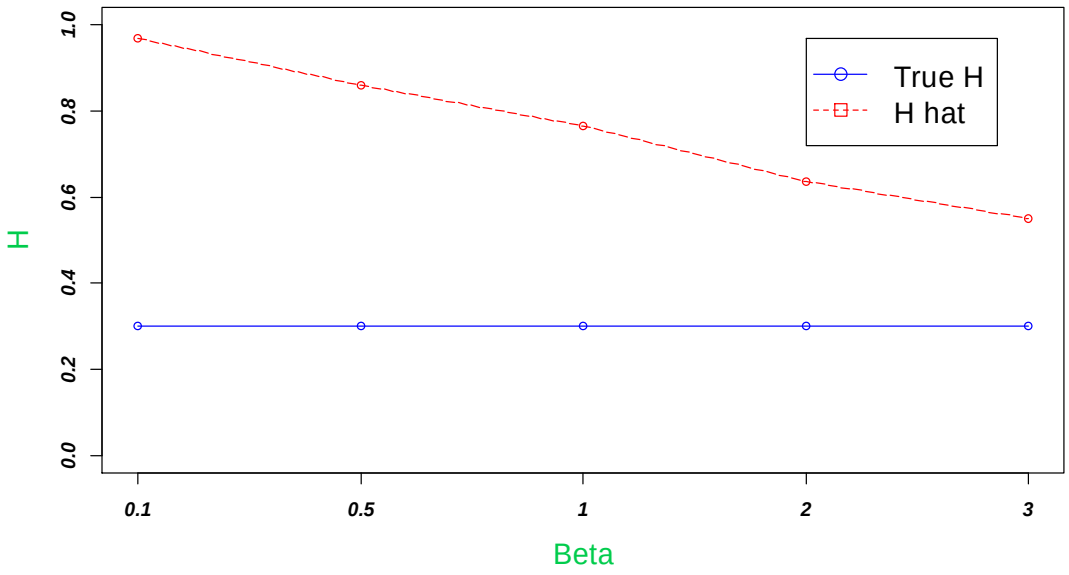

Figure 34. Changes in H Hat as β Increases Given True H=0.3, α = 3

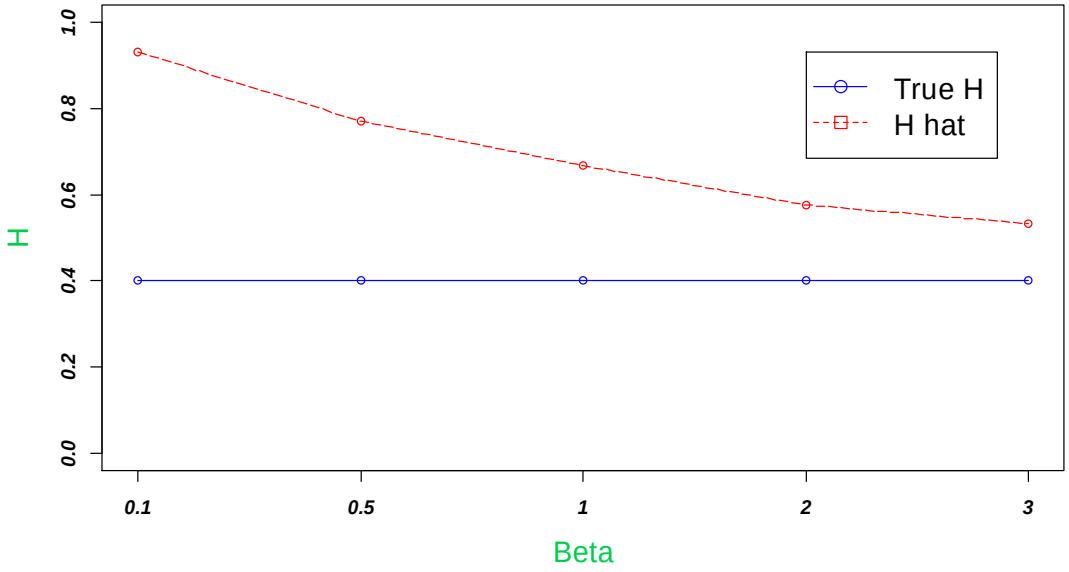

Figure 35. Changes in H Hat as β Increases Given True H=0.4, α = 0.5

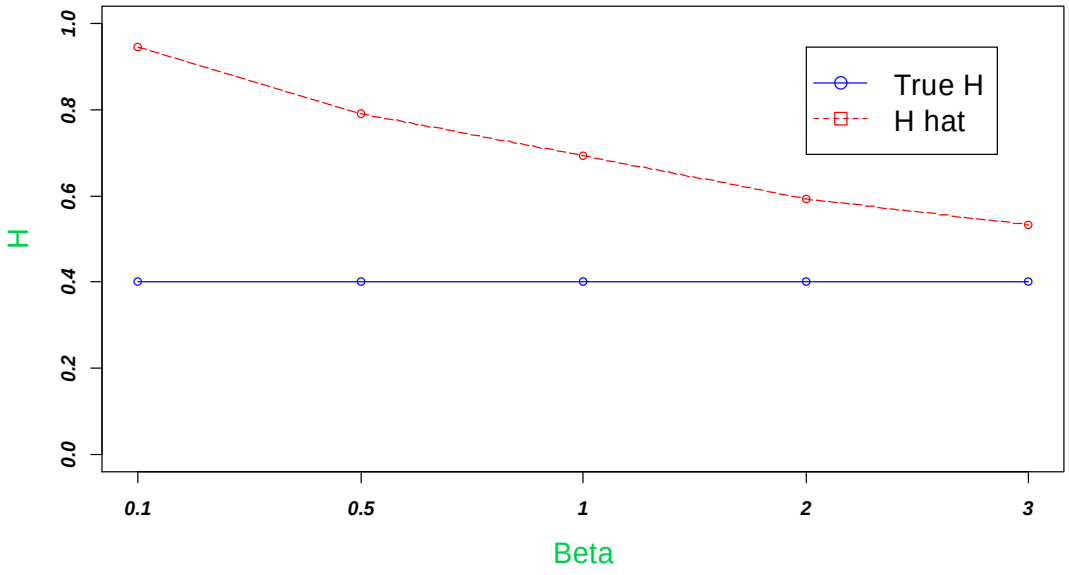

Figure 36. Changes in H Hat as β Increases Given True H=0.4, α = 1

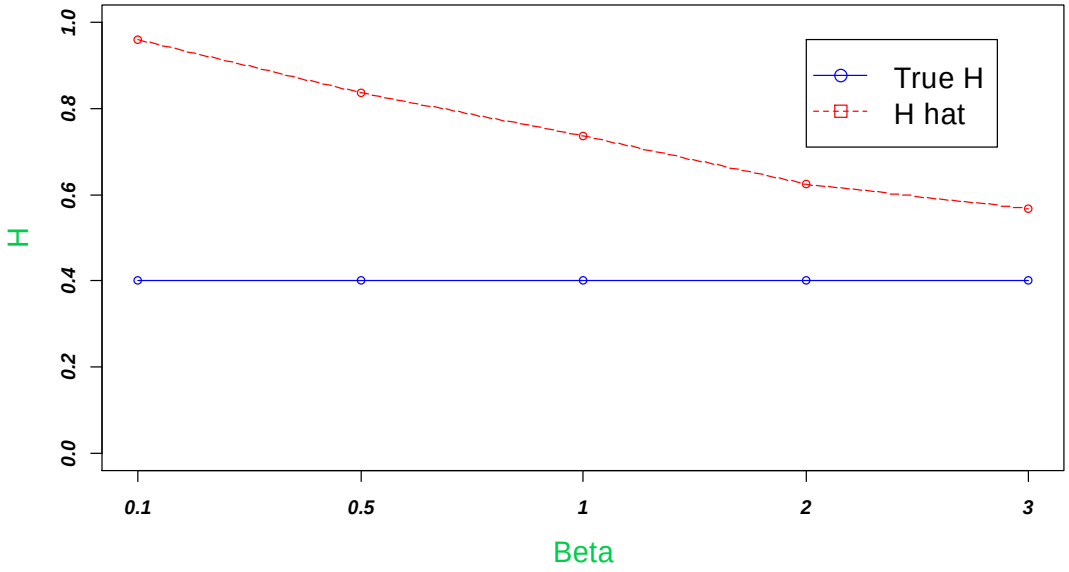

Figure 37. Changes in H Hat as β Increases Given True H=0.4, α = 2

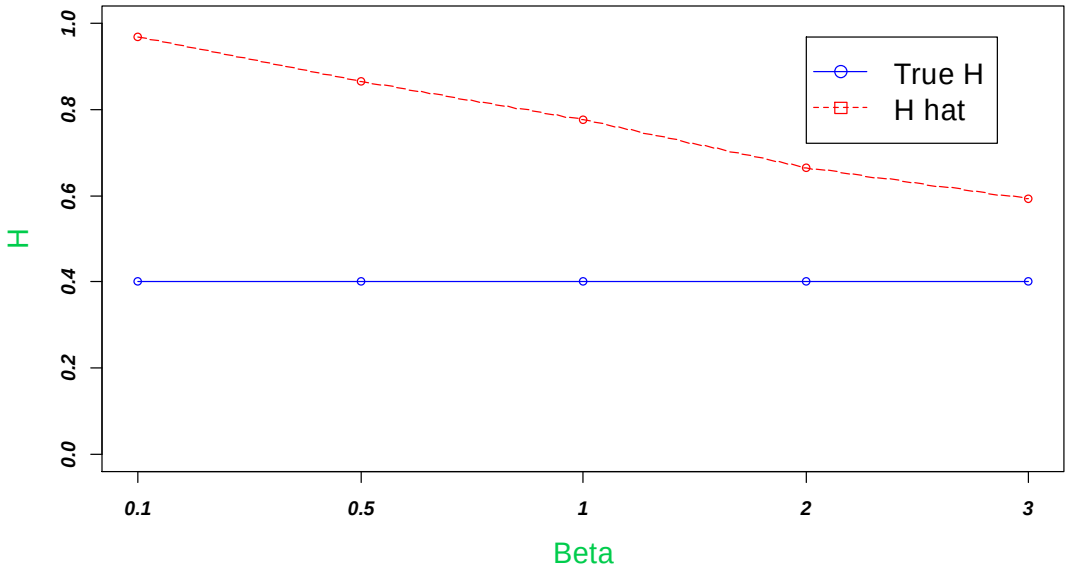

Figure 38. Changes in H Hat as β Increases Given True H=0.4, α = 3

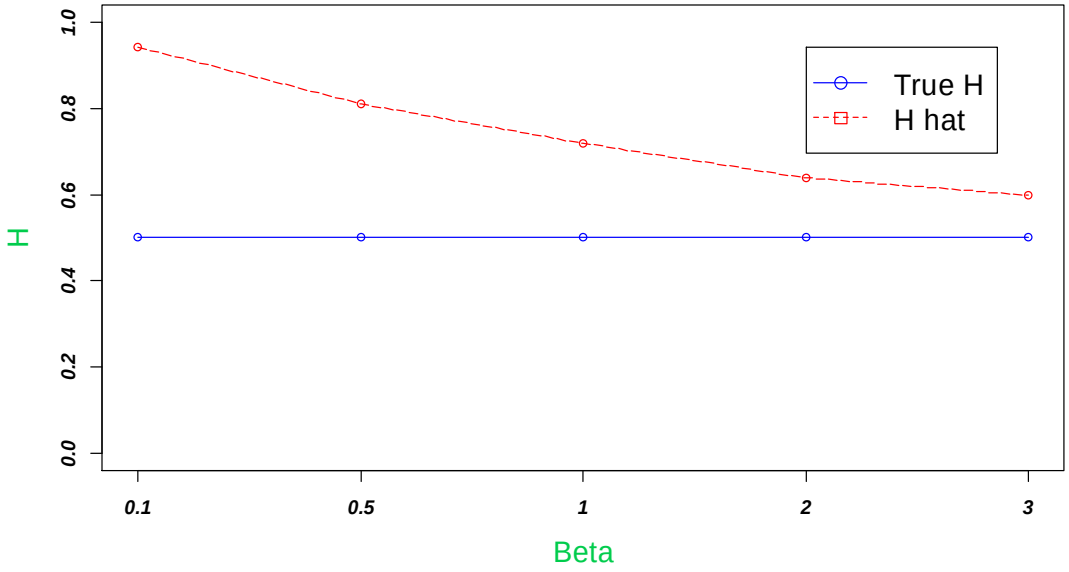

Figure 39. Changes in H Hat as β Increases Given True H=0.5, α = 0.5

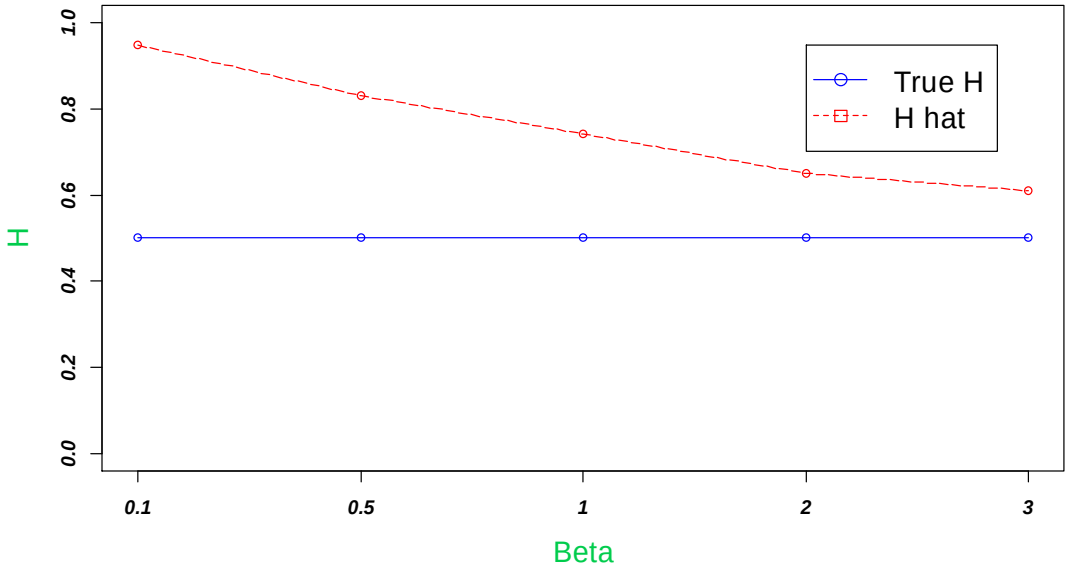

Figure 40. Changes in H Hat as β Increases Given True H=0.5, α = 1

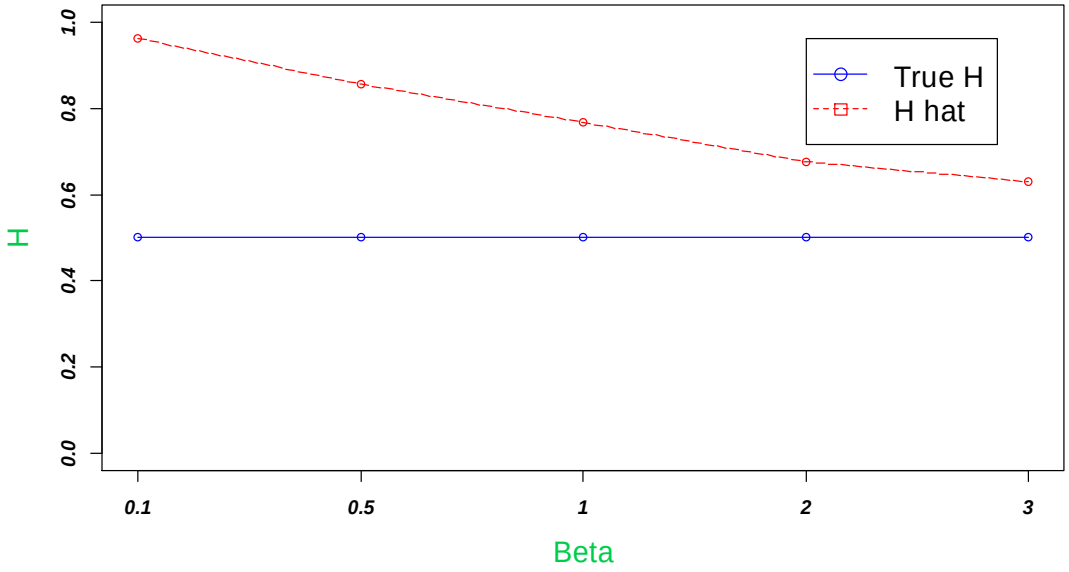

Figure 41. Changes in H Hat as β Increases Given True H=0.5, α = 2

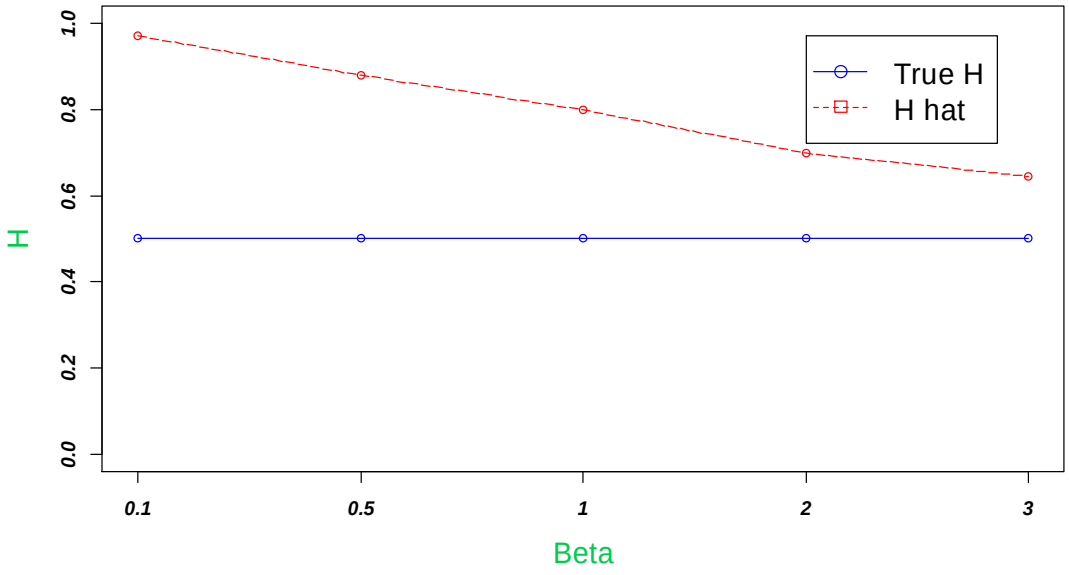

Figure 42. Changes in H Hat as β Increases Given True H=0.5, α = 3

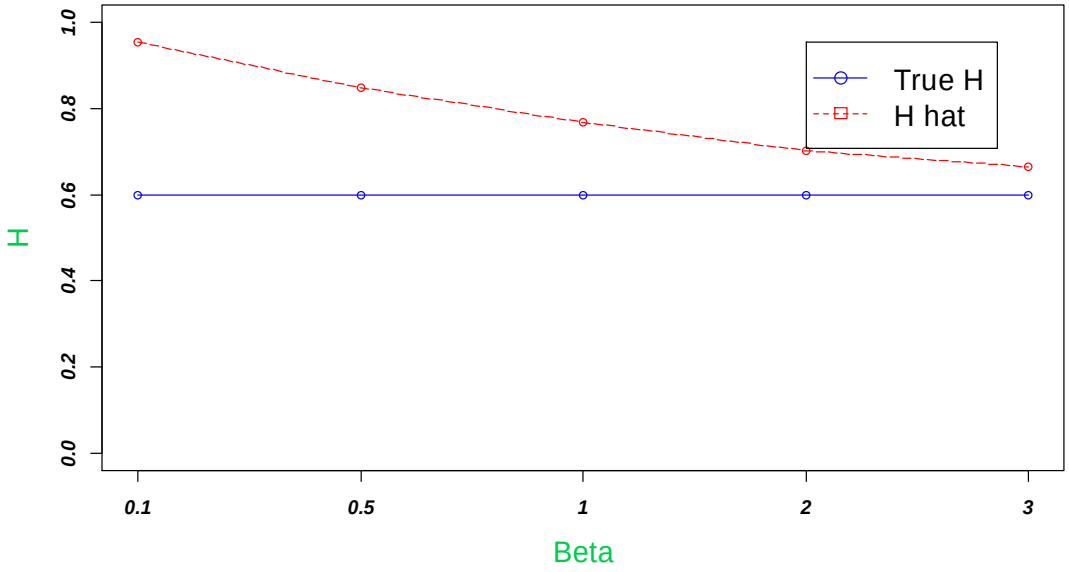

Figure 43. Changes in H Hat as β Increases Given True H=0.6, α = 0.5

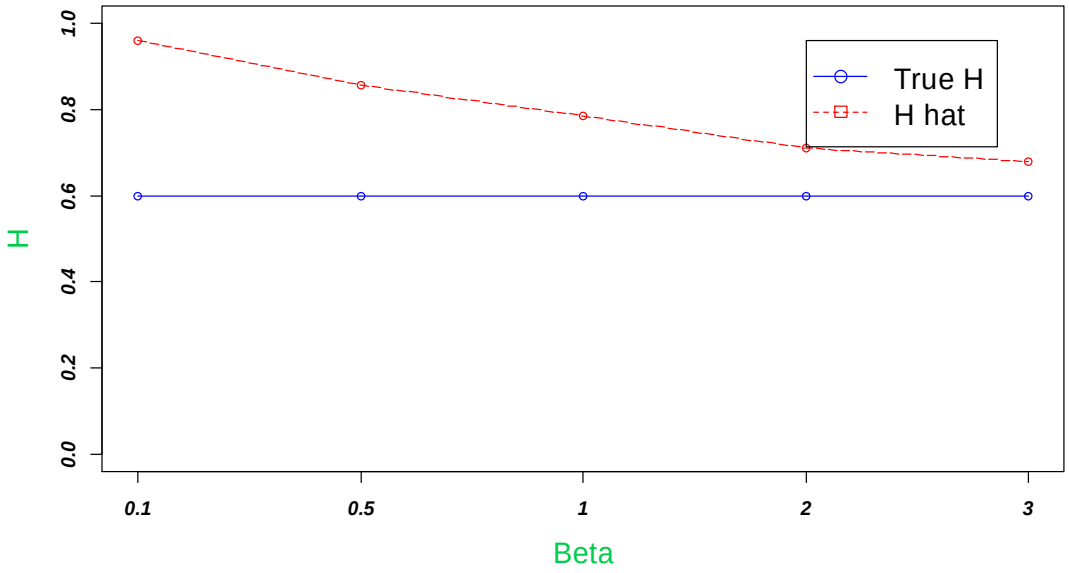

Figure 44. Changes in H Hat as β Increases Given True H=0.6, α = 1

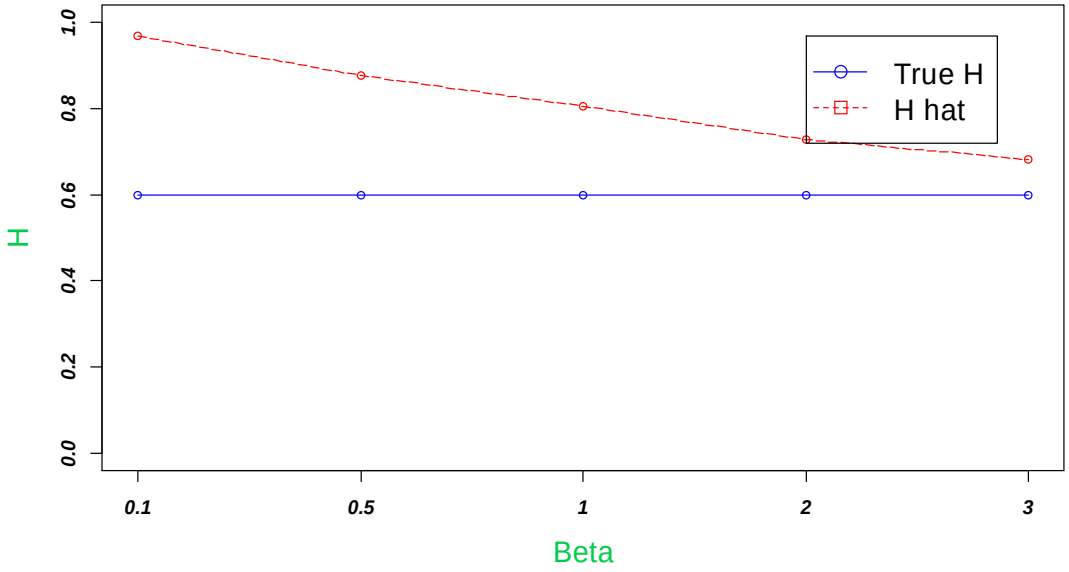

Figure 45. Changes in H Hat as β Increases Given True H=0.6, α = 2

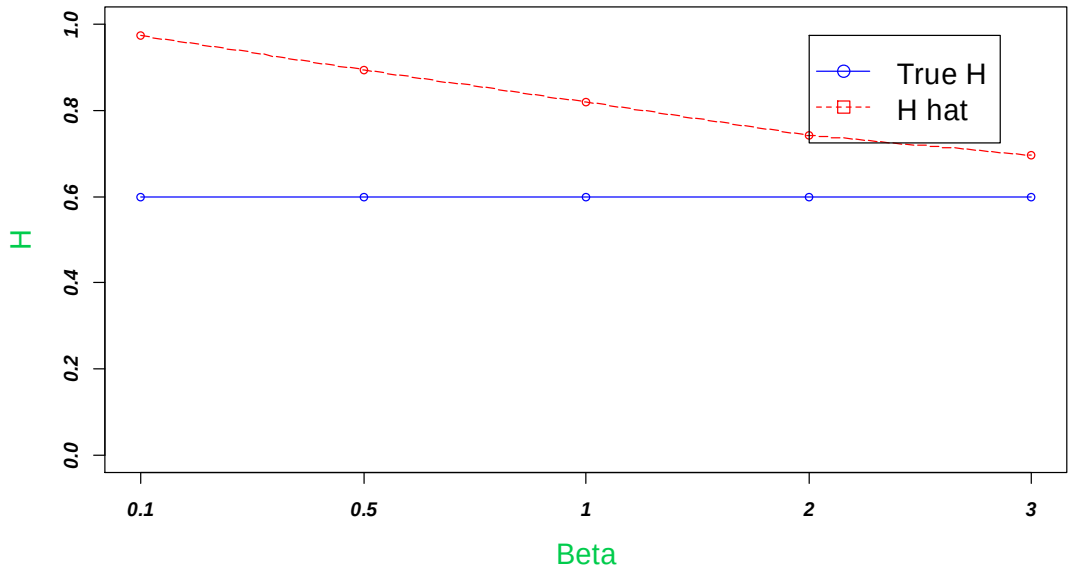

Figure 46. Changes in H Hat as β Increases Given True H=0.6, α = 3

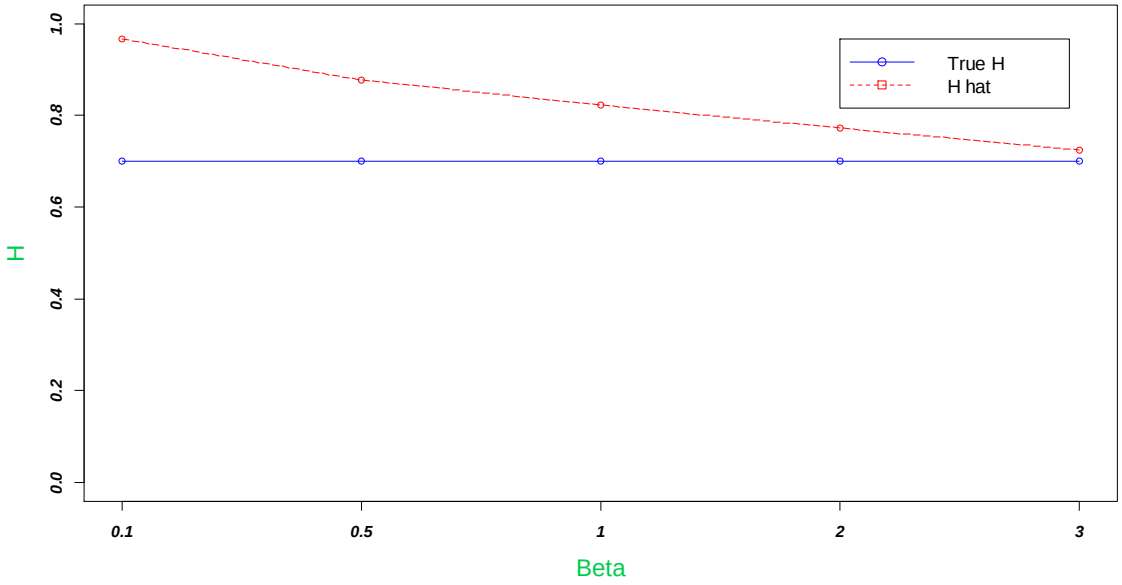

Figure 47. Changes in H Hat as β Increases Given True H=0.7, α = 0.5

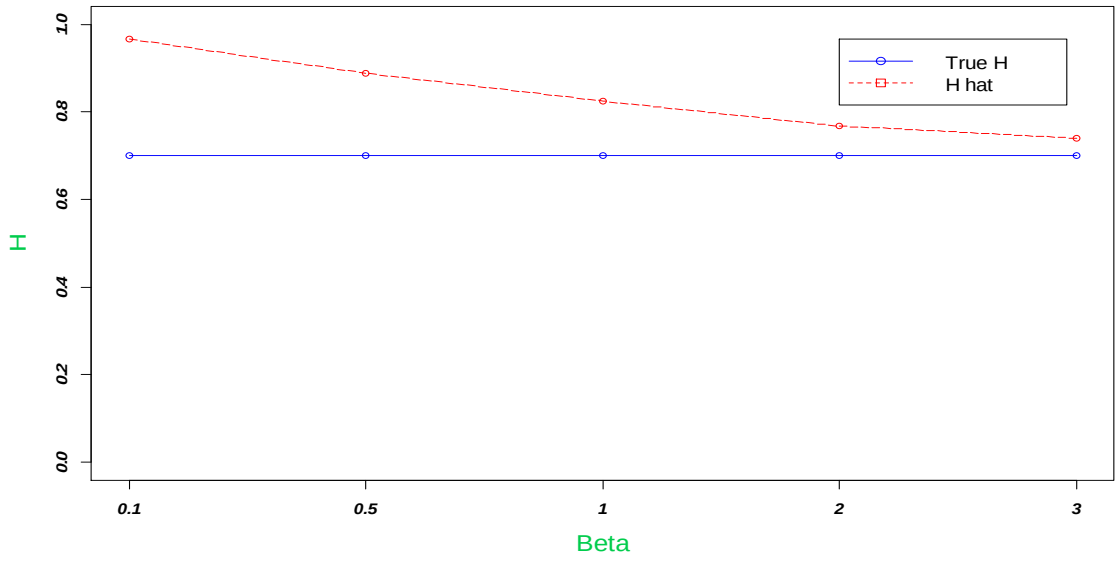

Figure 48. Changes in H Hat as β Increases Given True H=0.7, α = 1

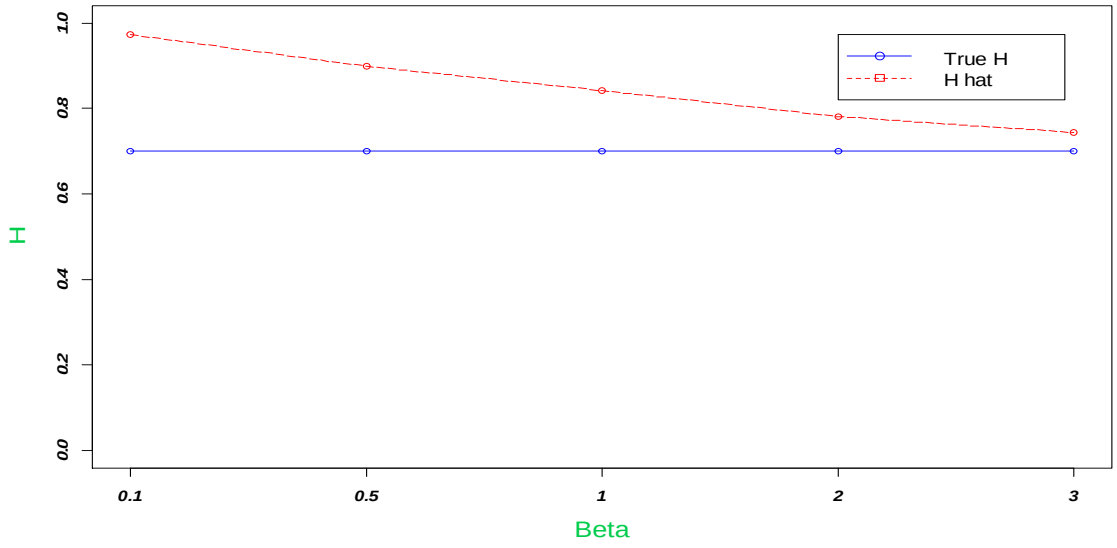

Figure 49. Changes in H Hat as β Increases Given True H=0.7, α = 2

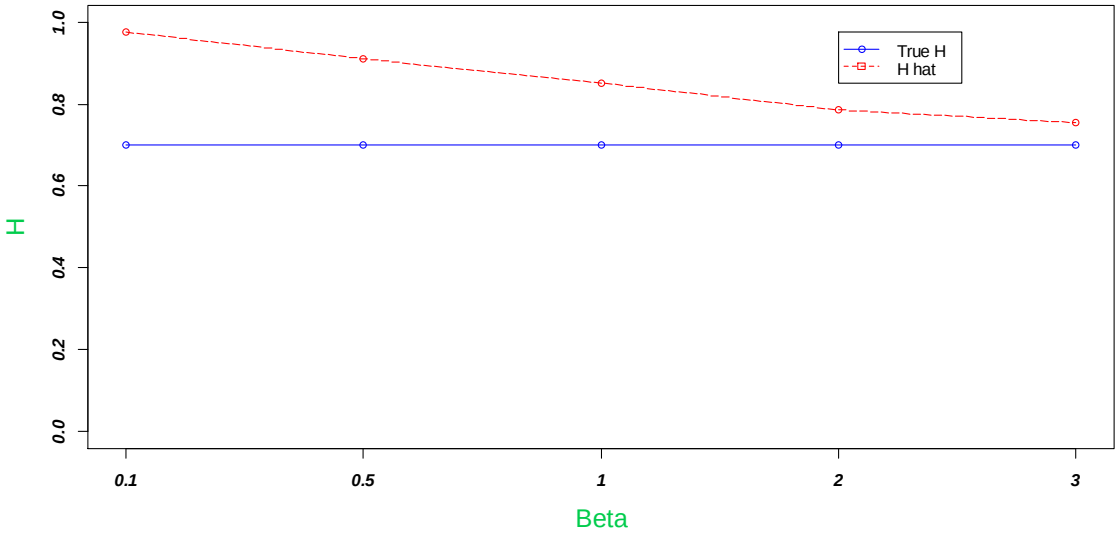

Figure 50. Changes in H Hat as β Increases Given True H=0.7, α = 3

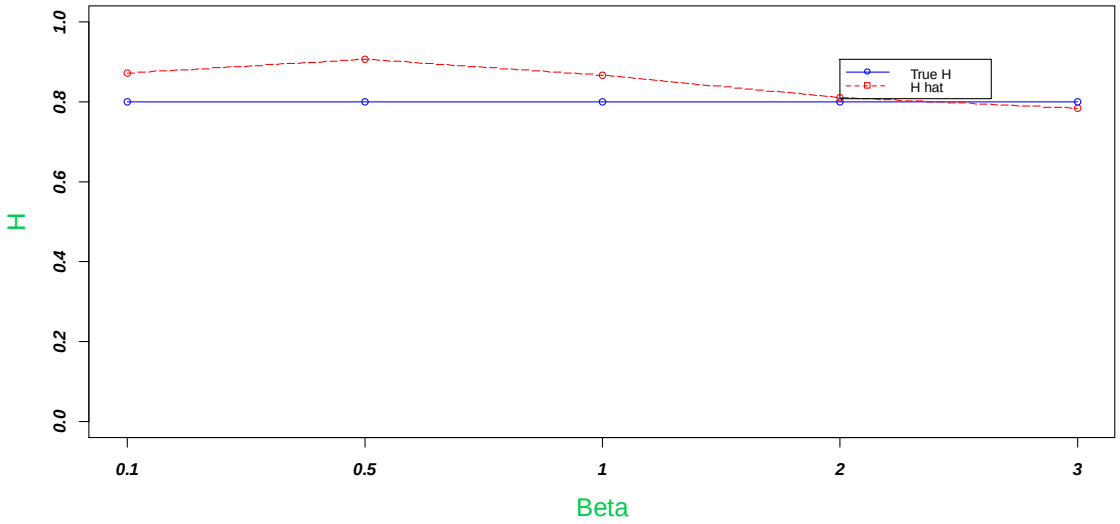

Figure 51. Changes in H Hat as β Increases Given True H=0.8, α = 0.5

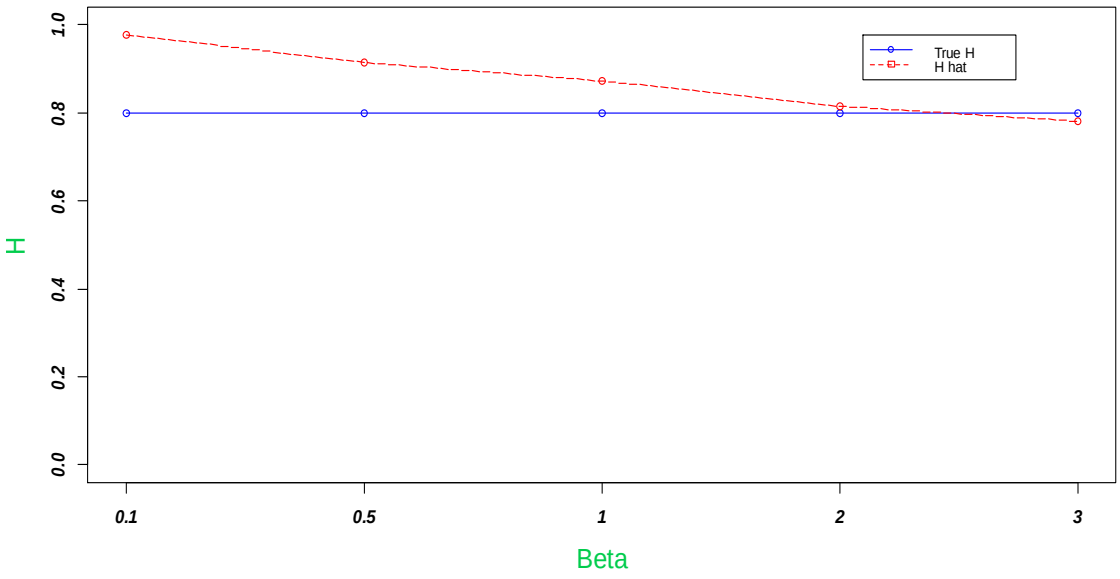

Figure 52. Changes in H Hat as β Increases Given True H=0.8, α = 1

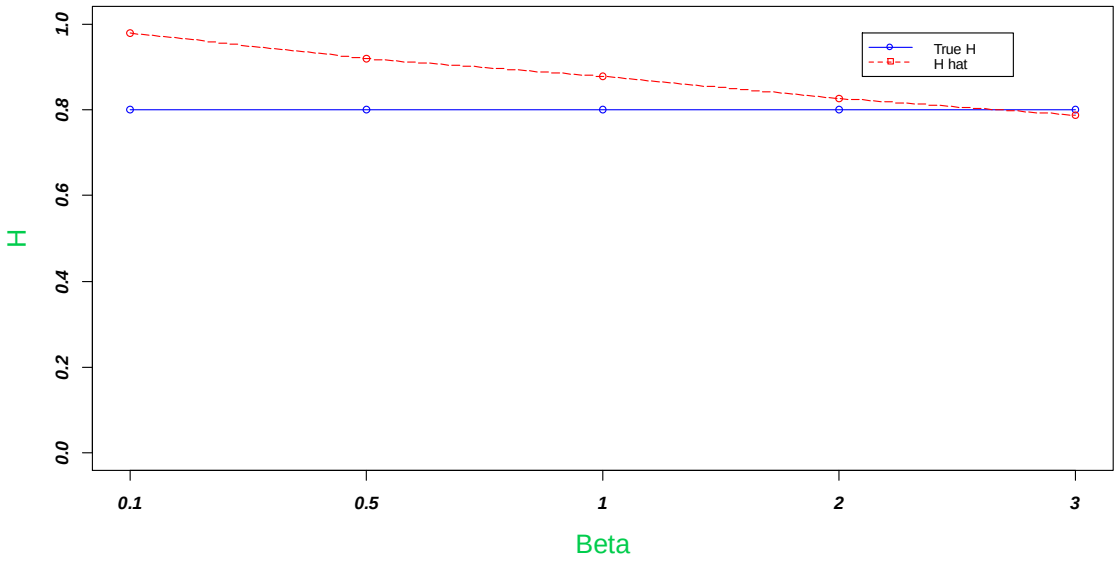

Figure 53. Changes in H Hat as β Increases Given True H=0.8, α = 2

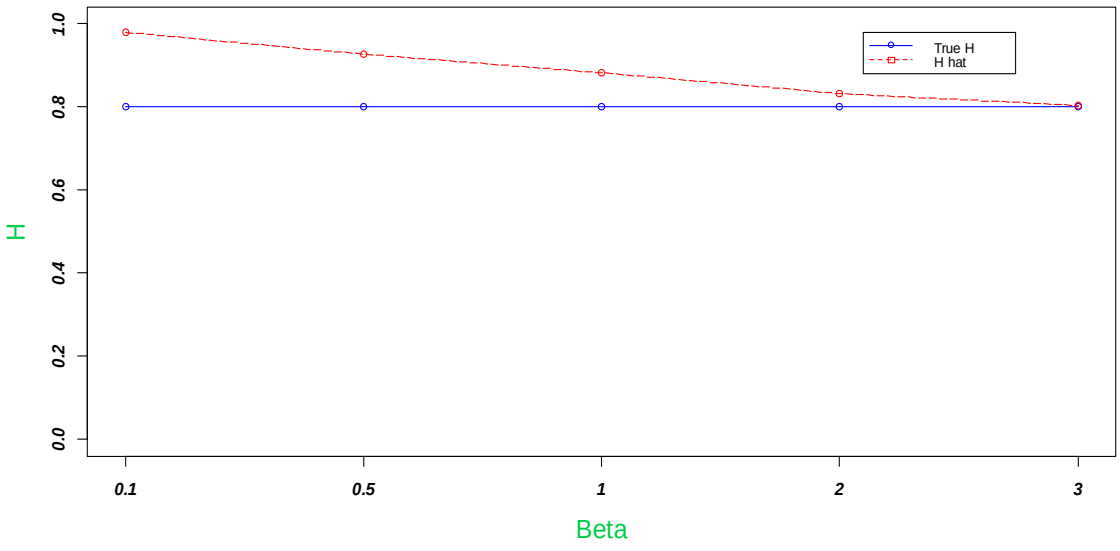

Figure 54. Changes in H Hat as β Increases Given True H=0.8, α = 3

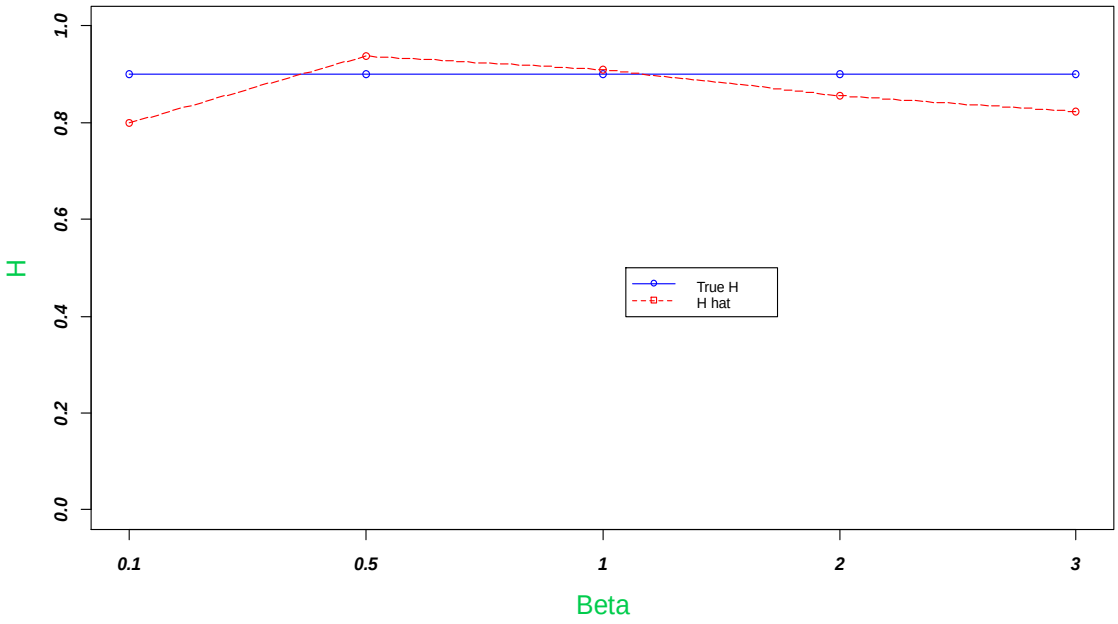

Figure 55. Changes in H Hat as β Increases Given True H=0.9, α = 0.5

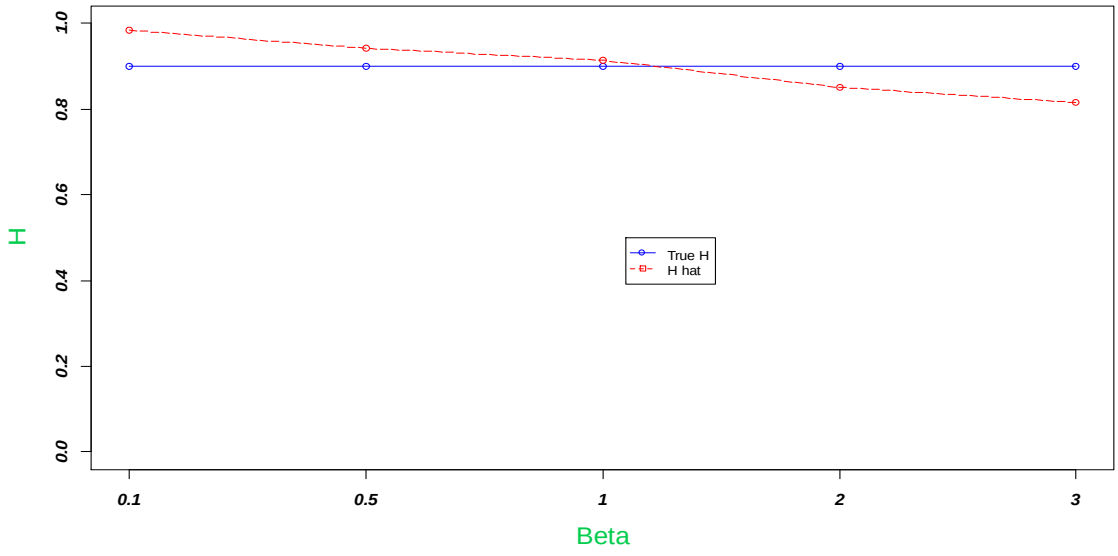

Figure 56. Changes in H Hat as β Increases Given True H=0.9, α = 1.

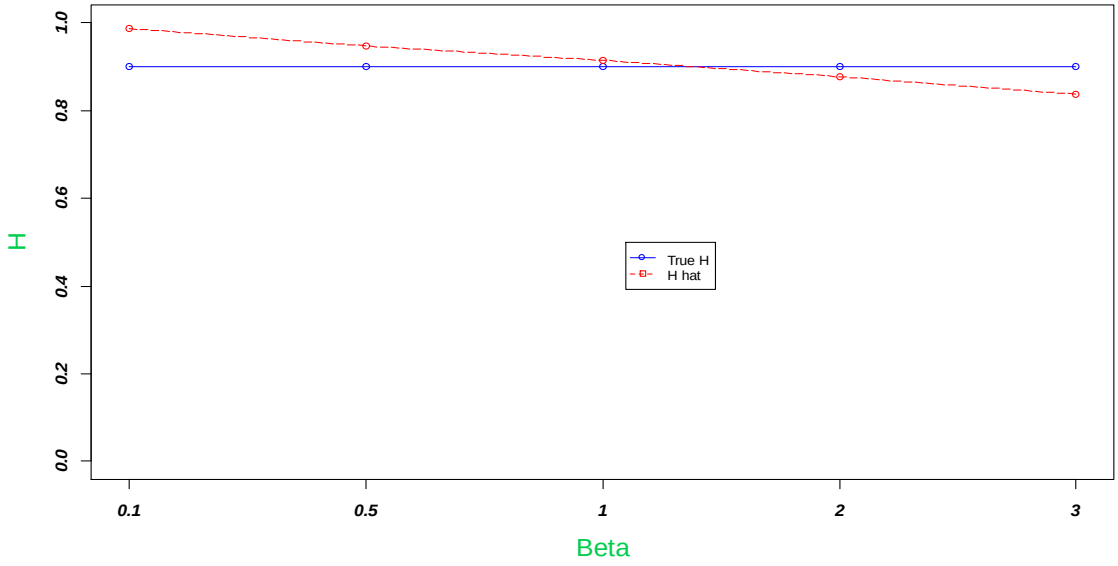

Figure 57. Changes in H Hat as β Increases Given True H=0.9, α = 2

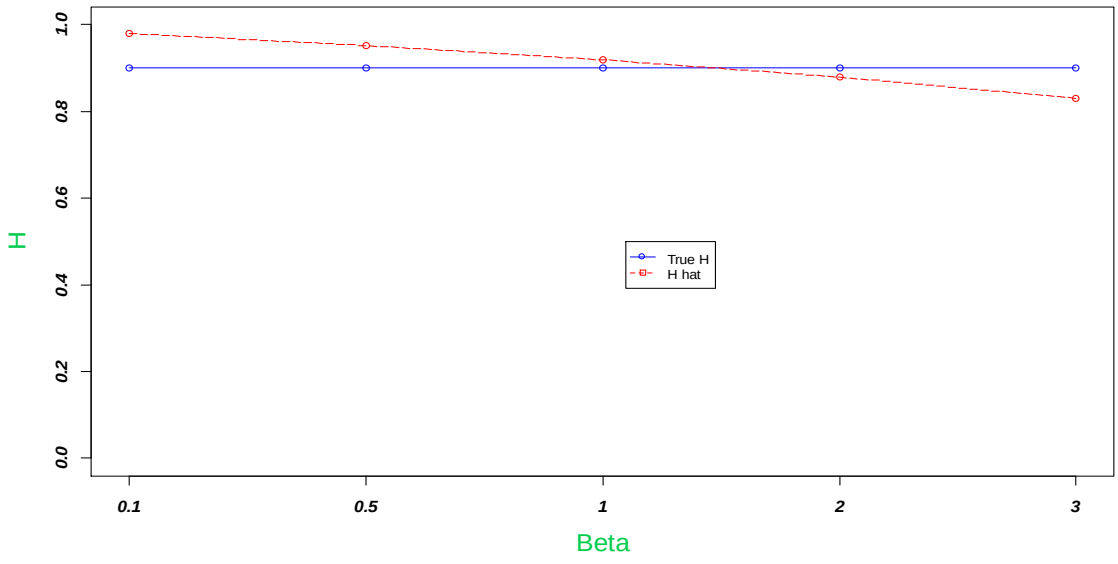

Figure 58. Changes in H Hat as β Increases Given True H=0.9, α = 3

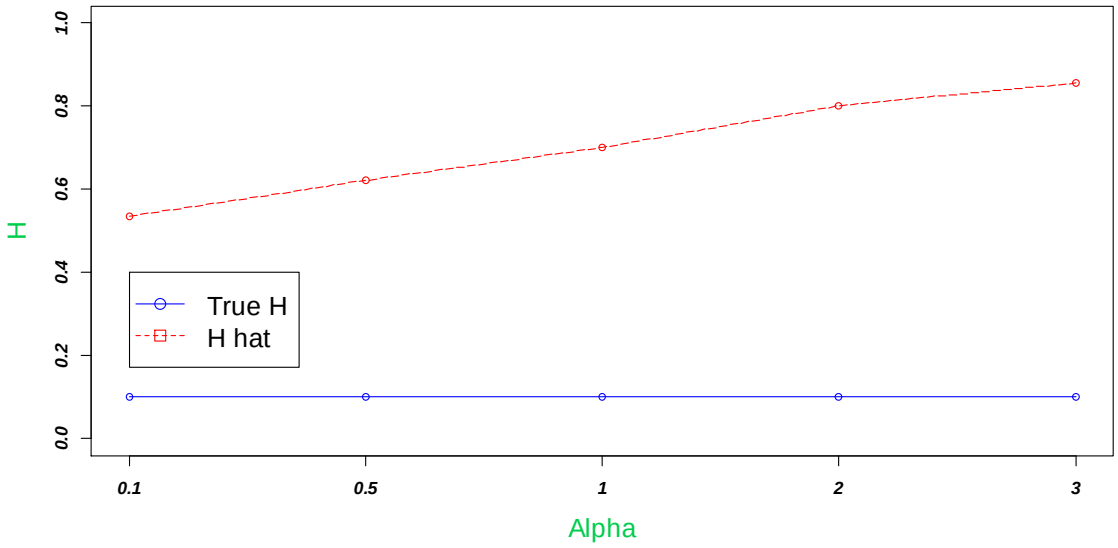

Figure 59. Change in H hat as α Increases Given True H=0.1, β=0.5

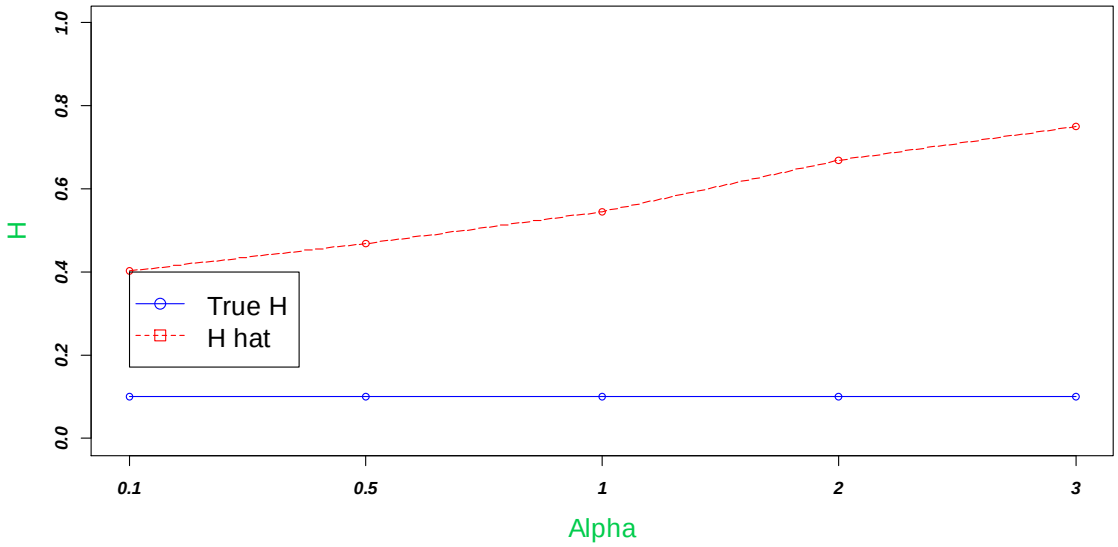

Figure 60. Change in H hat as α Increases Given True H=0.1, β=1

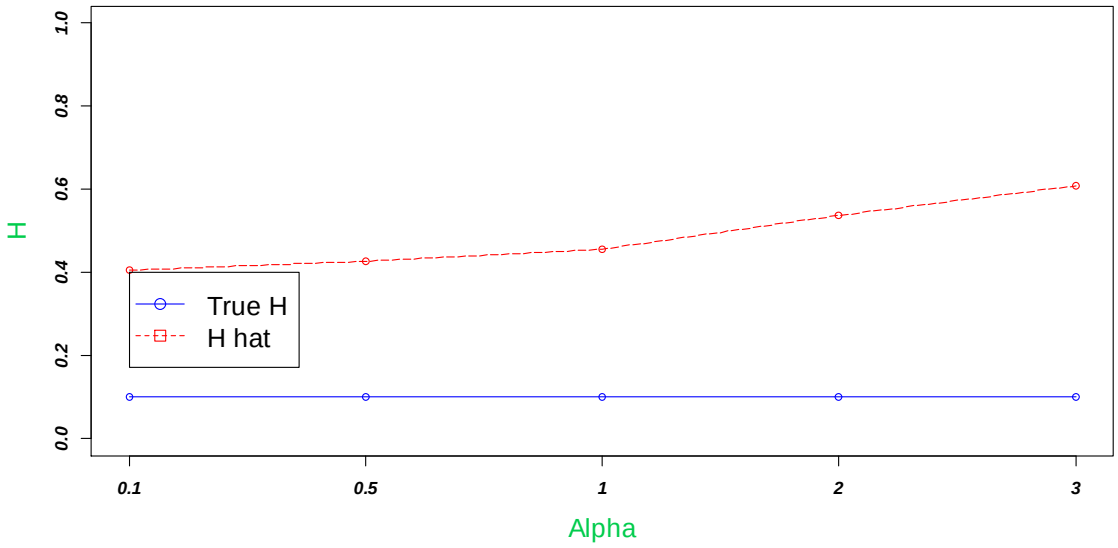

Figure 61. Change in H hat as α Increases Given True H=0.1, β=2

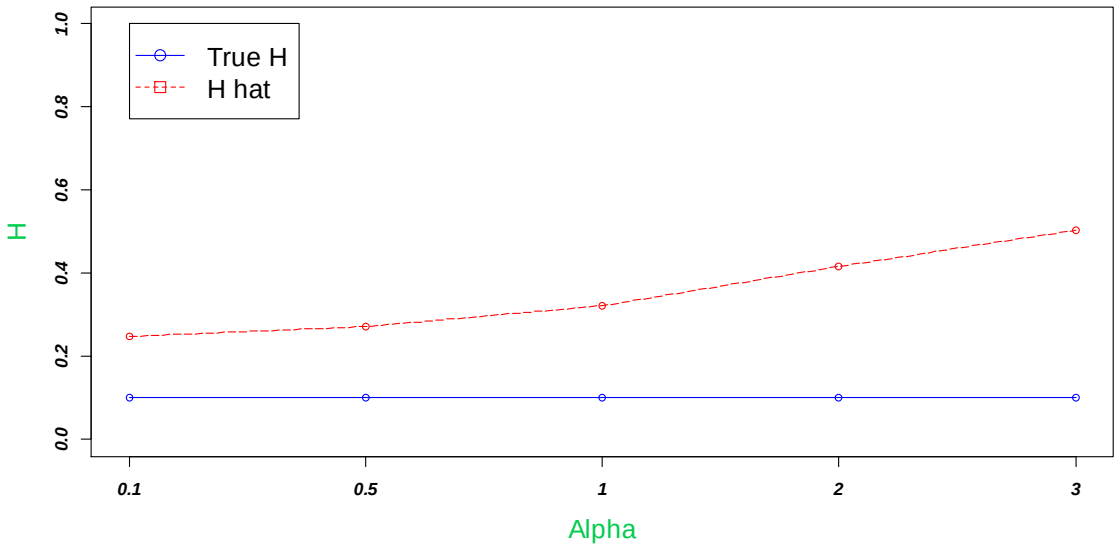

Figure 62. Change in H hat as α Increases Given True H=0.1, β=3.

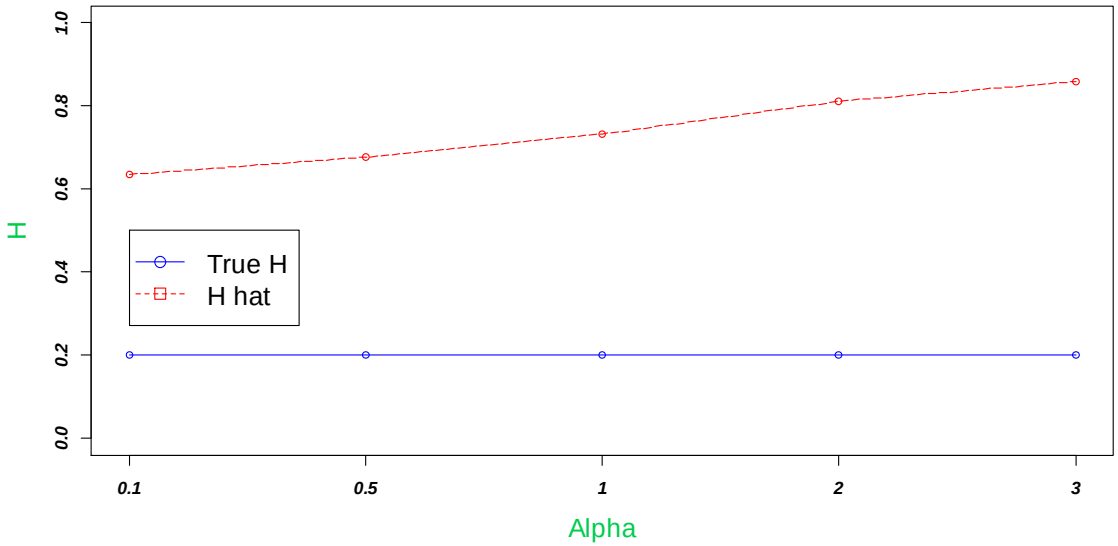

Figure 63. Change in H hat as α Increases Given True H=0.2, β=0.5

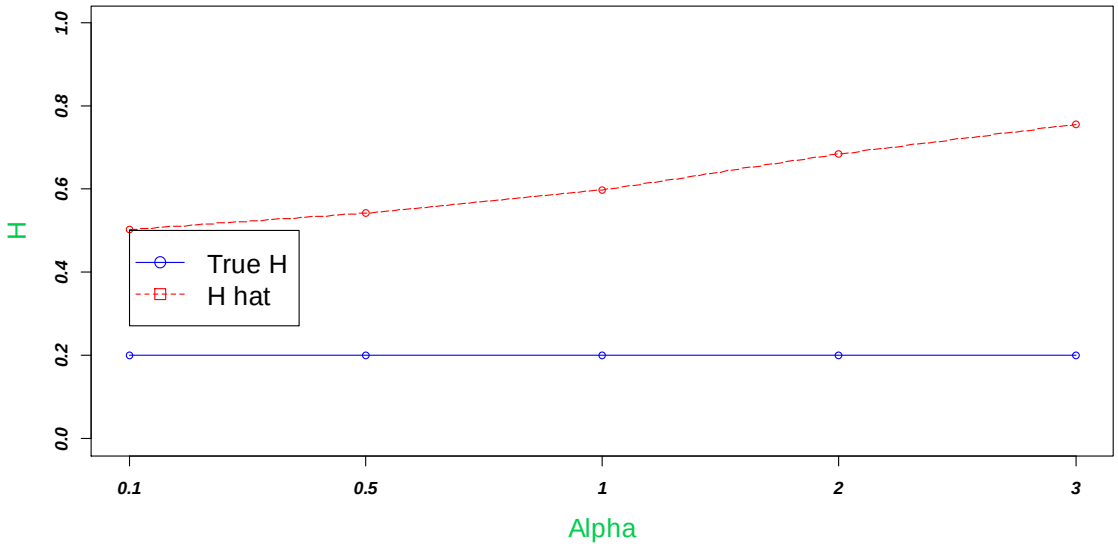

Figure 64. Change in H hat as α Increases Given True H=0.2, β=1

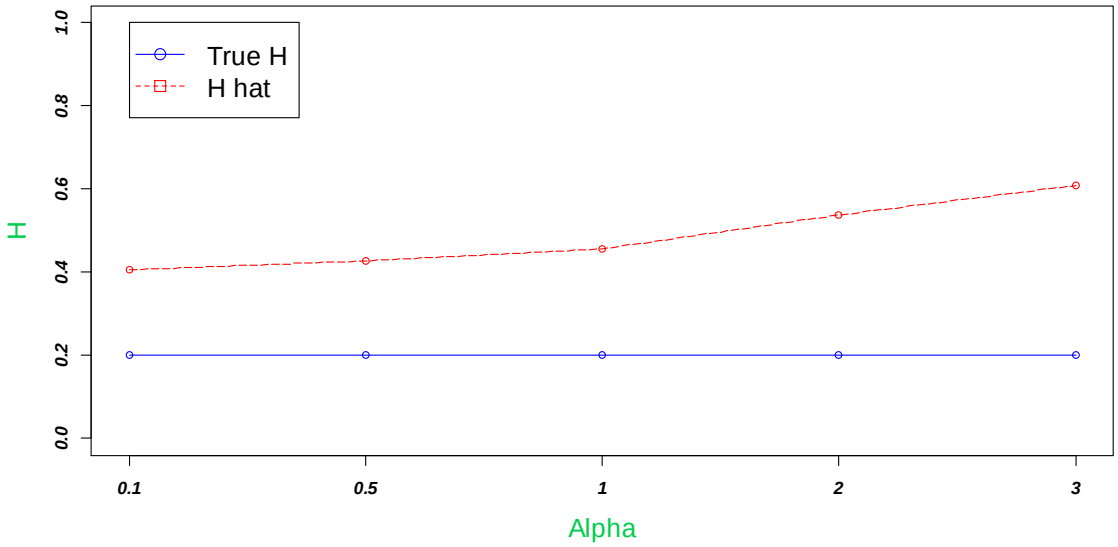

Figure 65. Change in H hat as α Increases Given True H=0.2, β=2

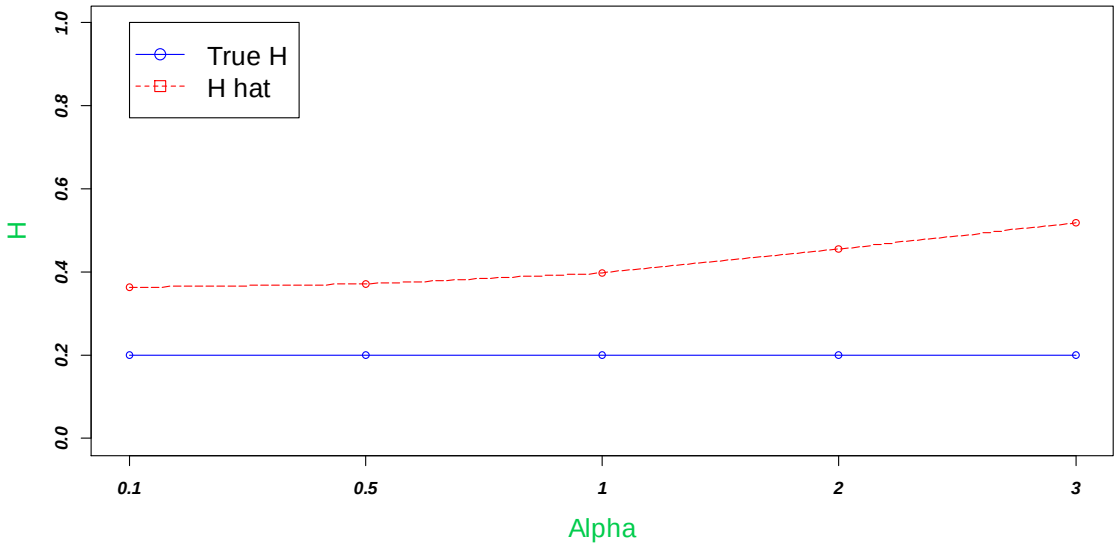

Figure 66. Change in H hat as α Increases Given True H=0.2, β=3

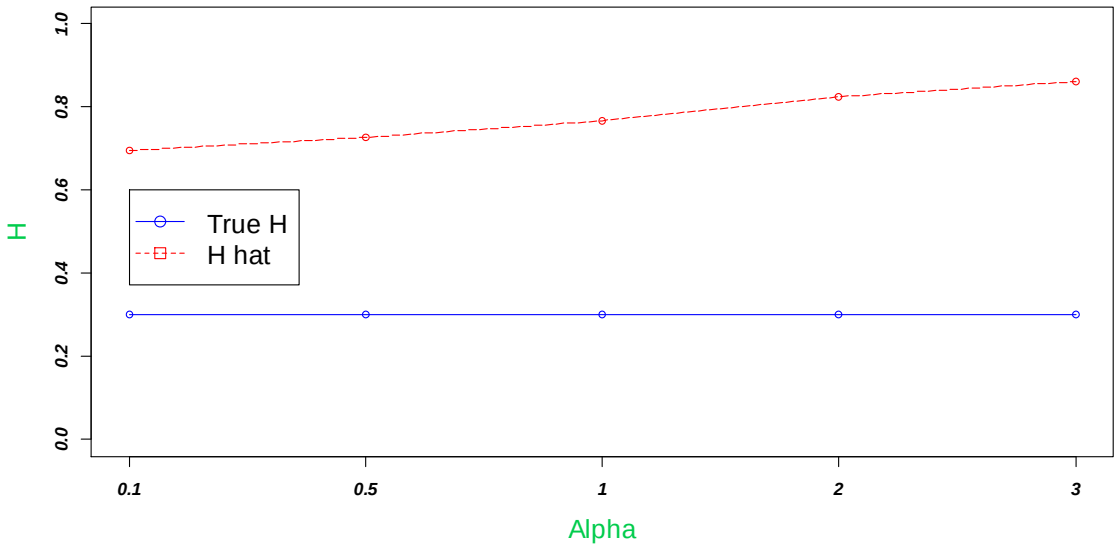

Figure 67. Change in H hat as α Increases Given True H=0.3, β=0.5

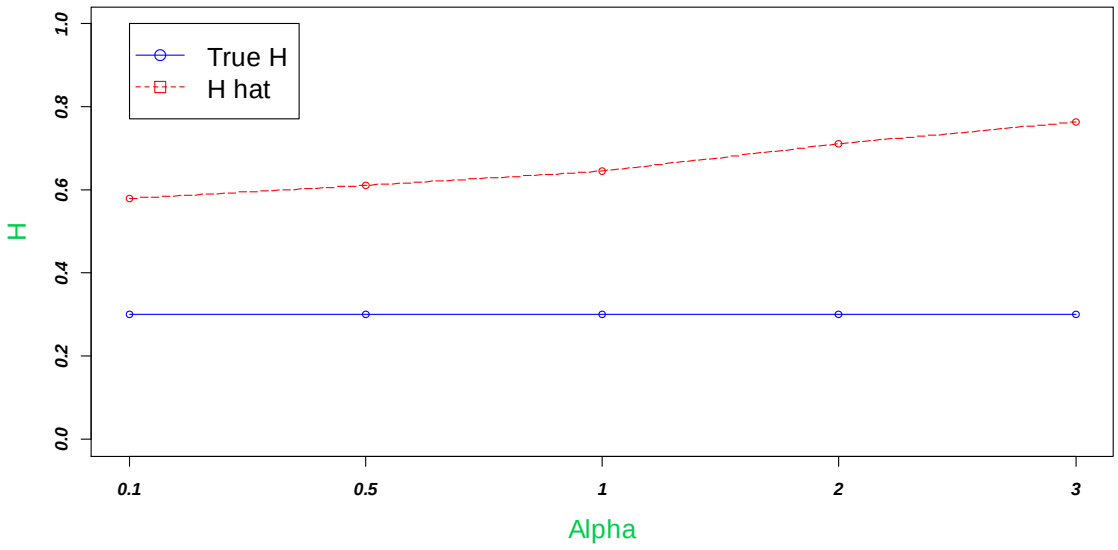

Figure 68. Change in H hat as α Increases Given True H=0.3, β=1

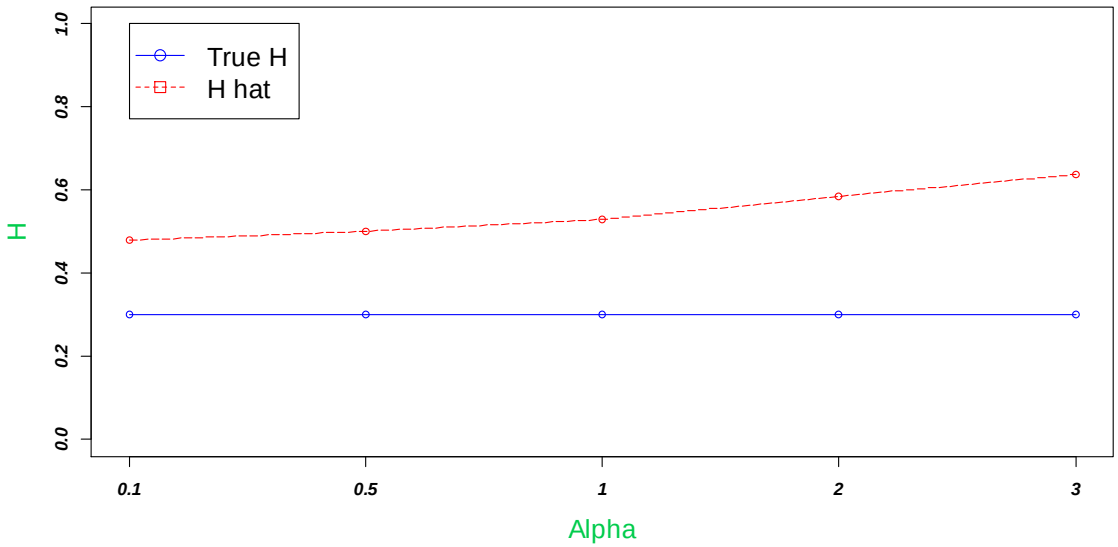

Figure 69. Change in H hat as α Increases Given True H=0.3, β=2

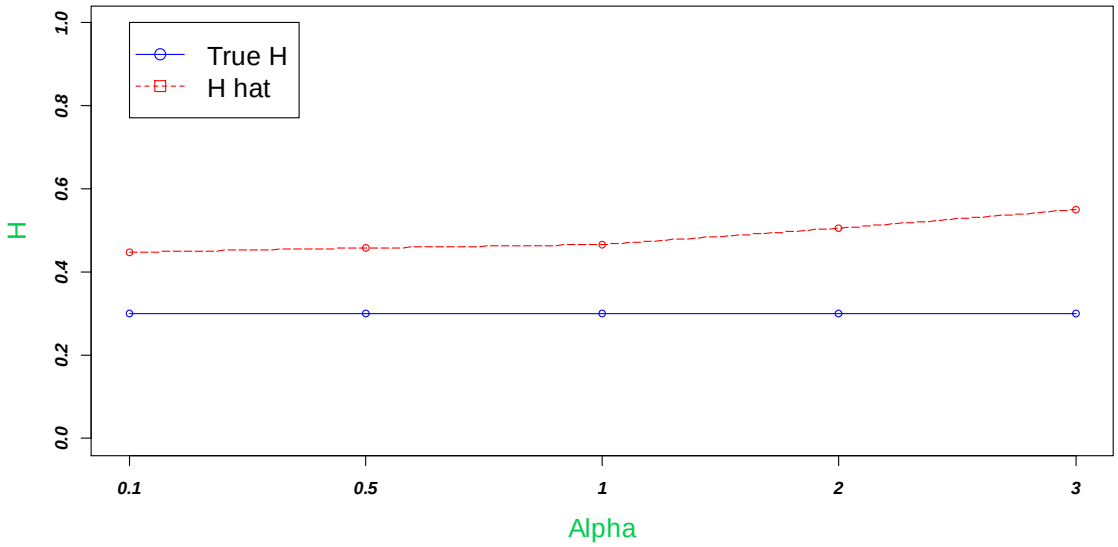

Figure 70. Change in H hat as α Increases Given True H=0.3, β=3

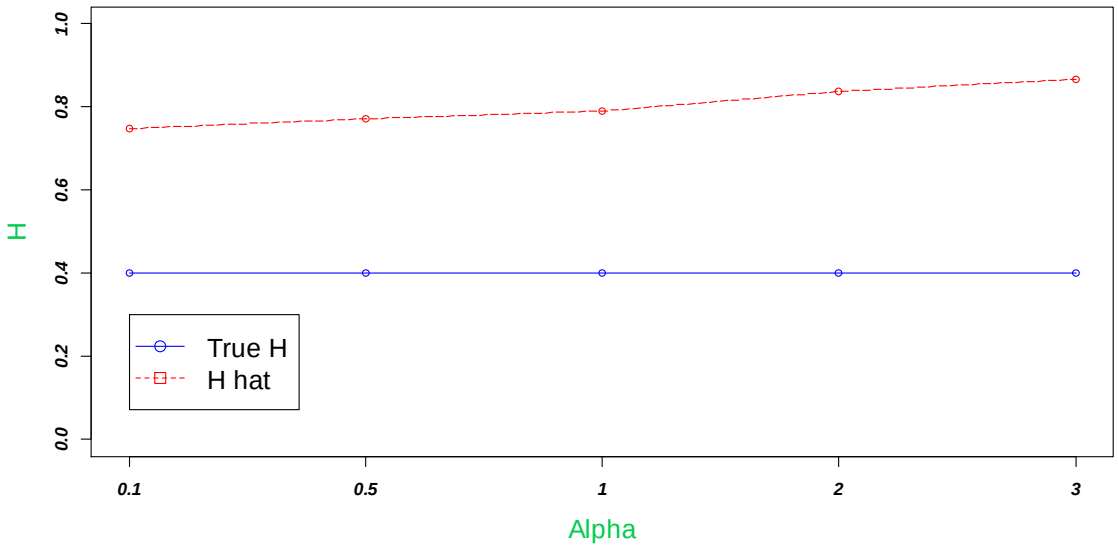

Figure 71. Change in H hat as α Increases Given True H=0.4, β=0.5

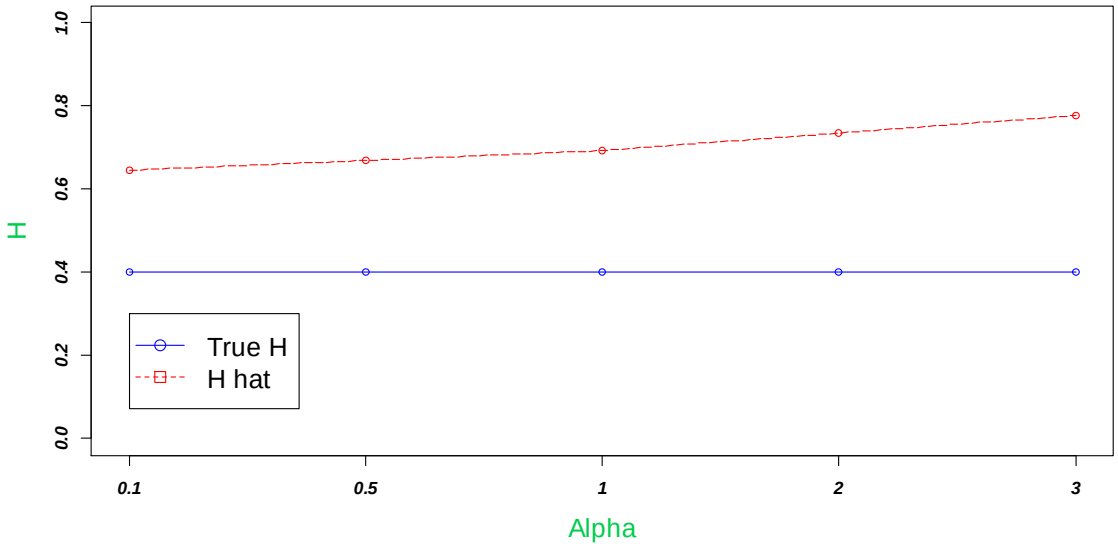

Figure 72. Change in H hat as α Increases Given True H=0.4, β=1

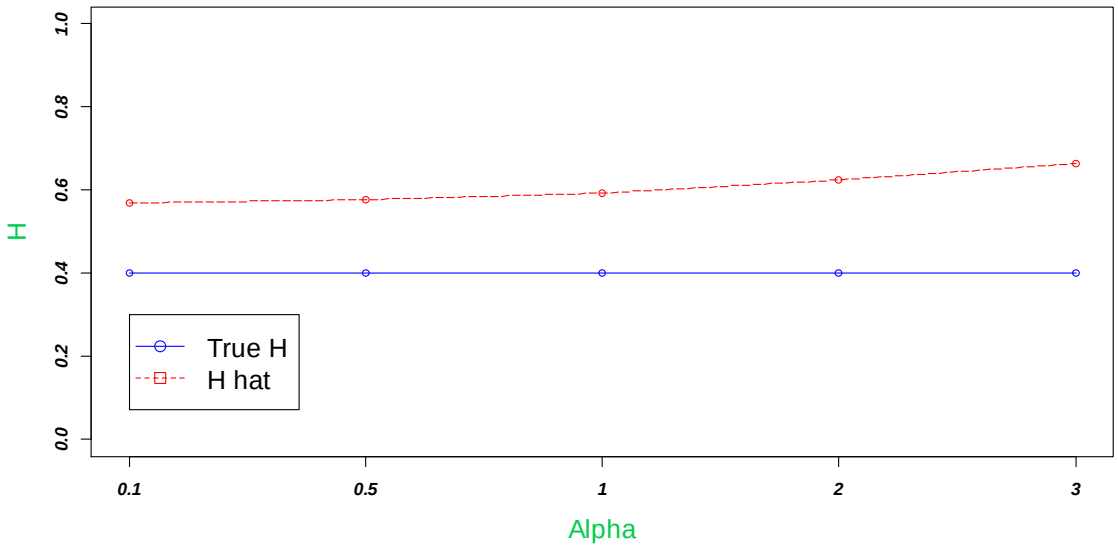

Figure 73. Change in H hat as α Increases Given True H=0.4, β=2

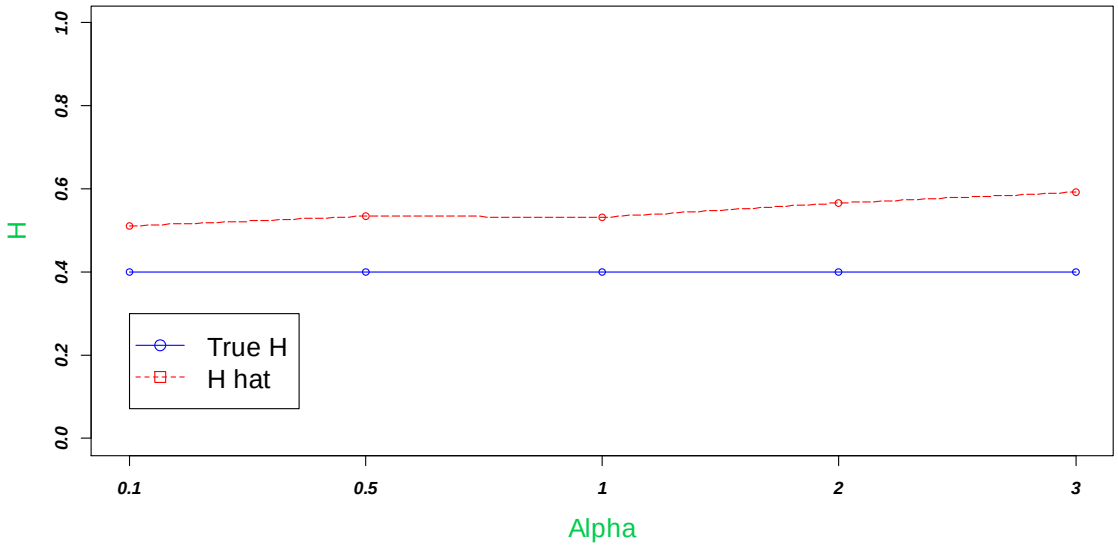

Figure 74. Change in H hat as α Increases Given True H=0.4, β=3

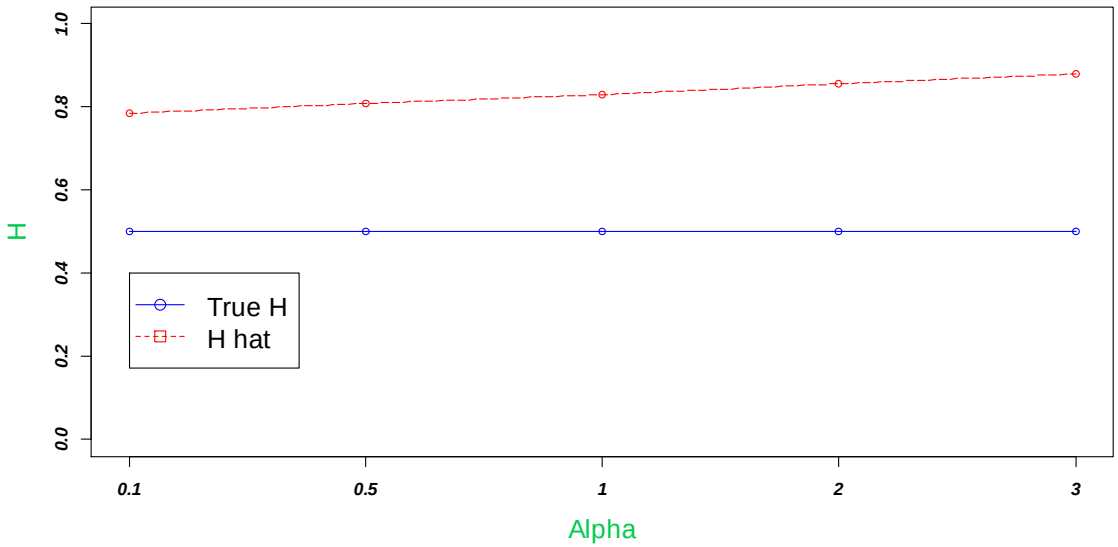

Figure 75. Change in H hat as α Increases Given True H=0.5, β=0.5

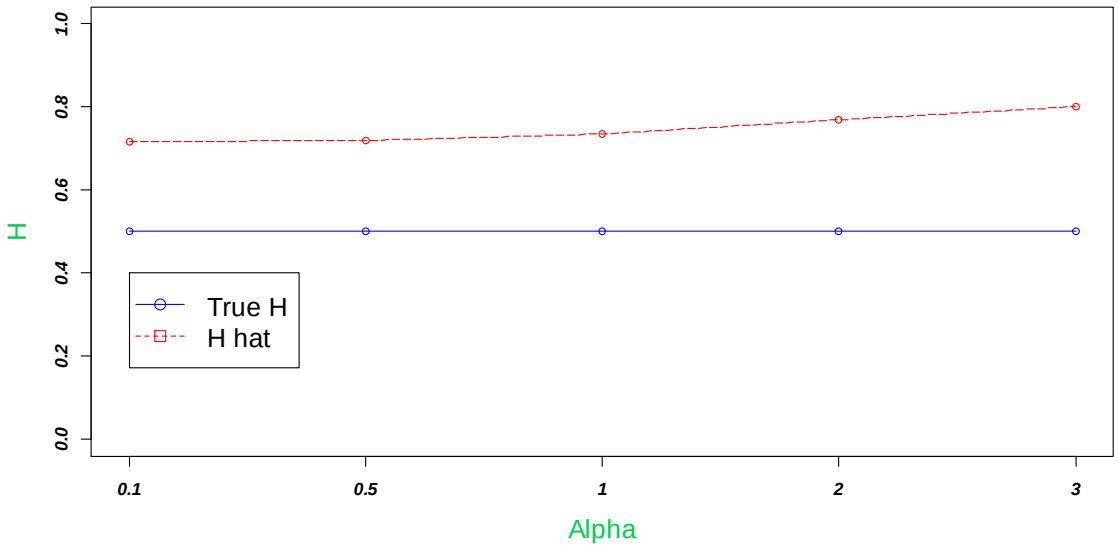

Figure 76. Change in H hat as α Increases Given True H=0.5, β=1

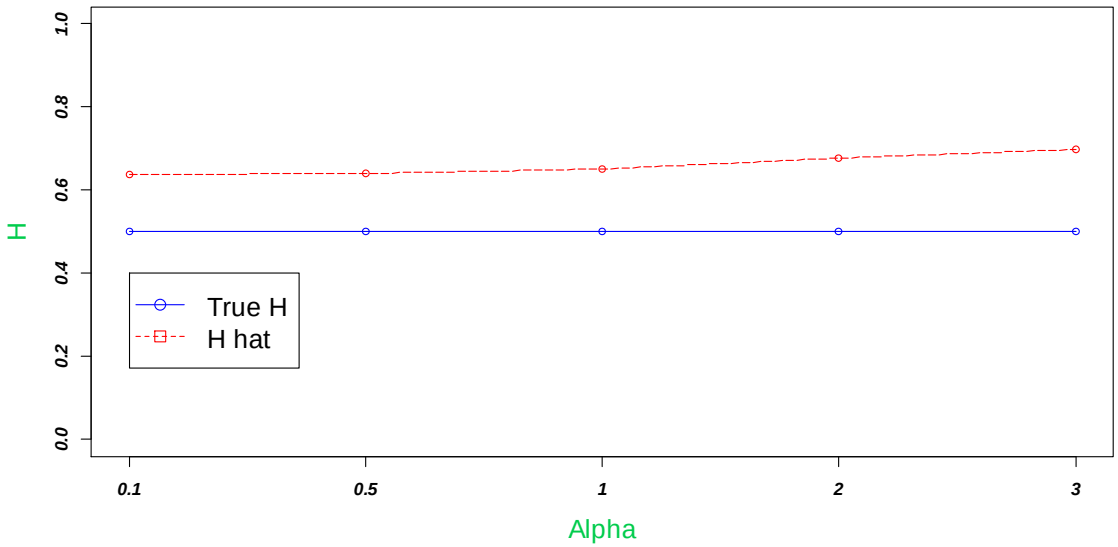

Figure 77. Change in H hat as α Increases Given True H=0.5, β=2

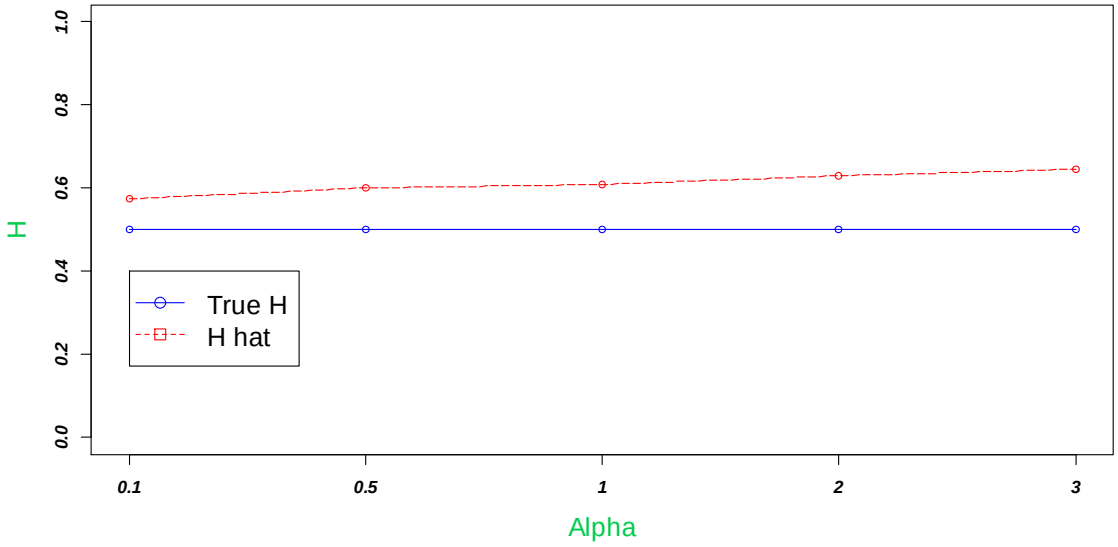

Figure 78. Change in H hat as α Increases Given True H=0.5, β=3

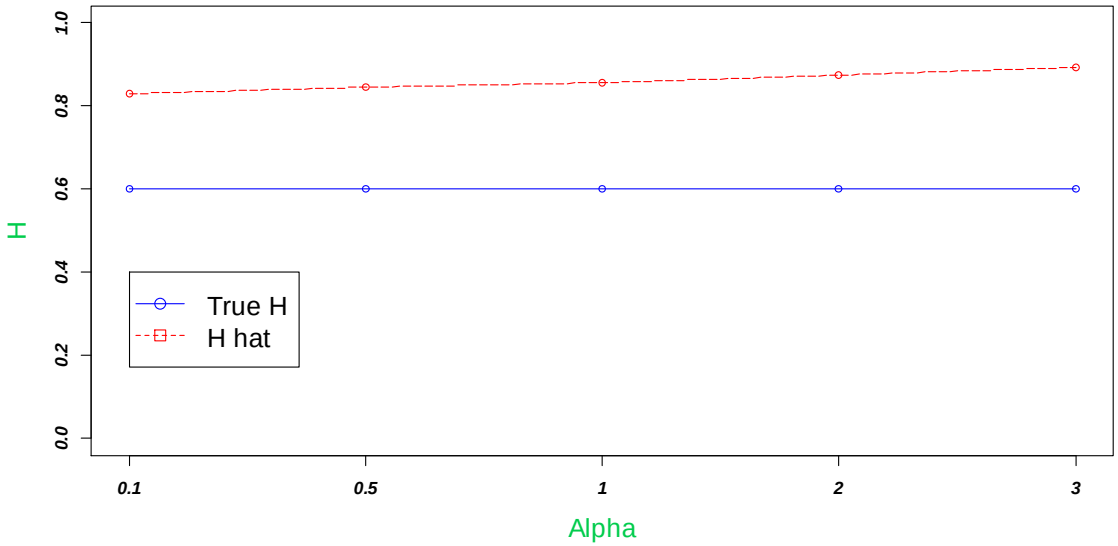

Figure 79. Change in H hat as α Increases Given True H=0.6 β=0.5

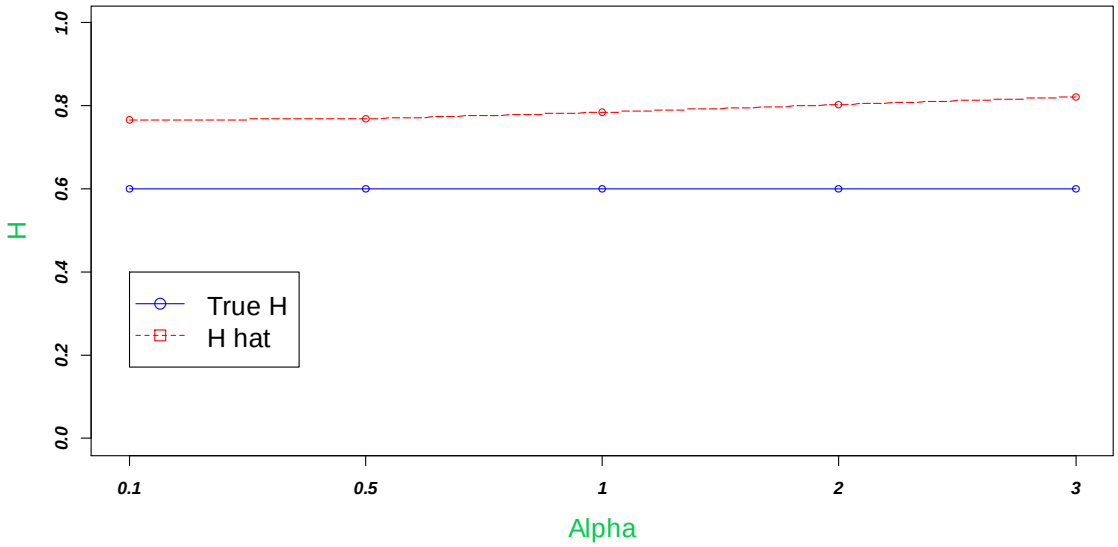

Figure 80. Change in H hat as α Increases Given True H=0.6, β=1

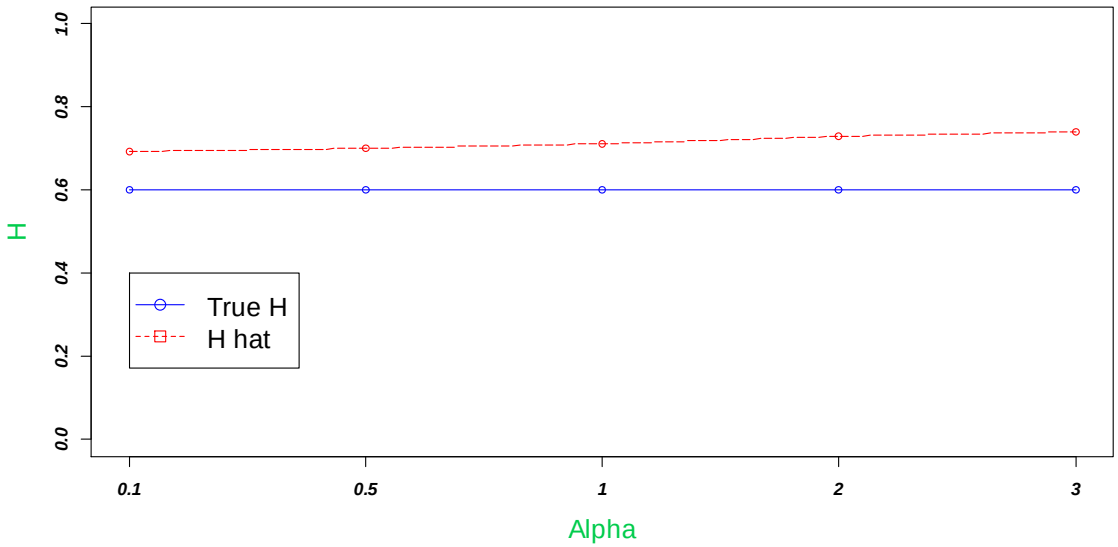

Figure 81. Change in H hat as α Increases Given True H=0.6, β=2

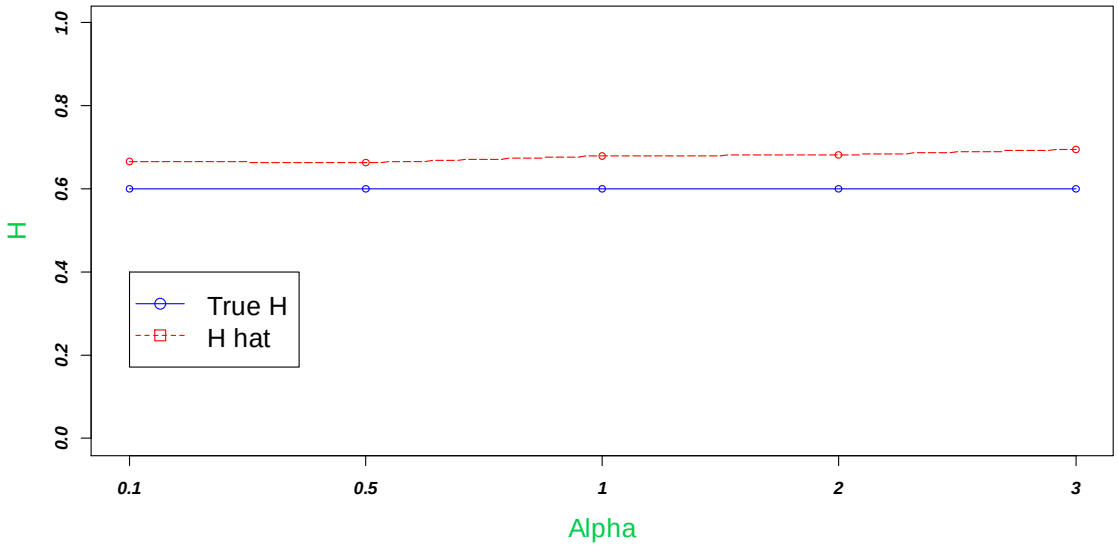

Figure 82. Change in H hat as α Increases Given True H=0.6, β=3

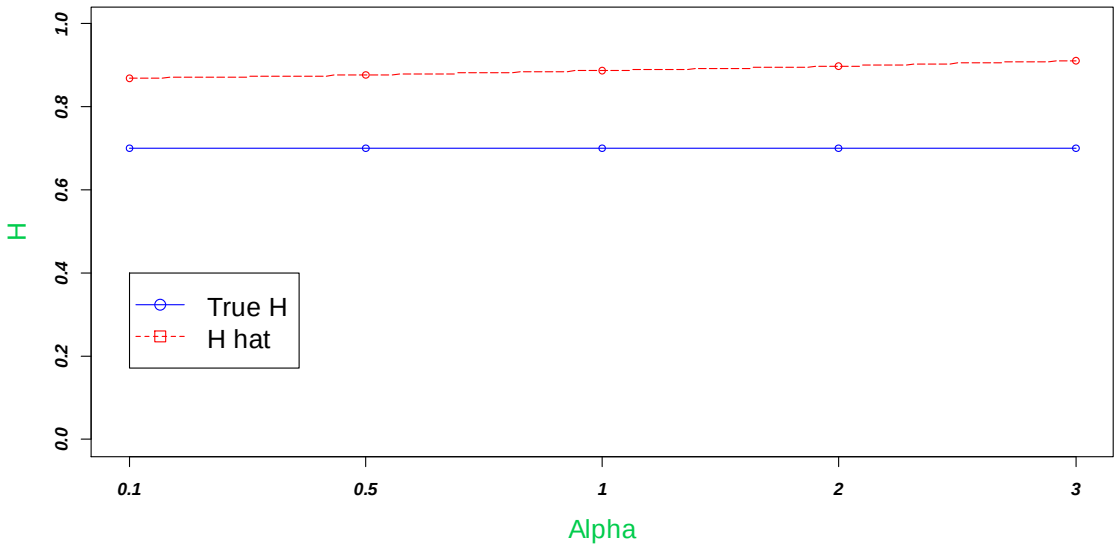

Figure 83. Change in H hat as α Increases Given True H=0.7, β=0.5

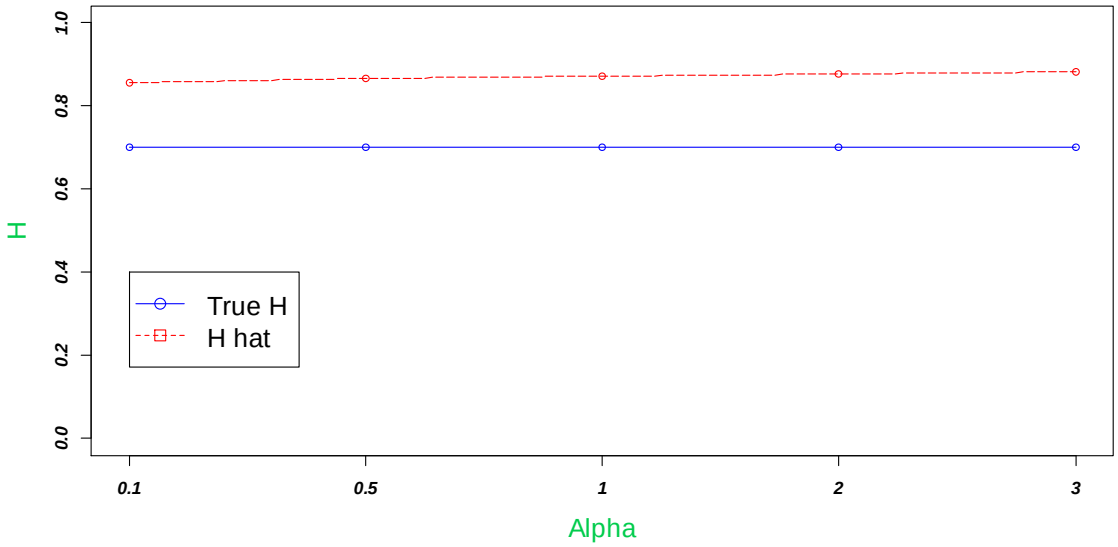

Figure 84. Change in H hat as α Increases Given True H=0.7, β=1

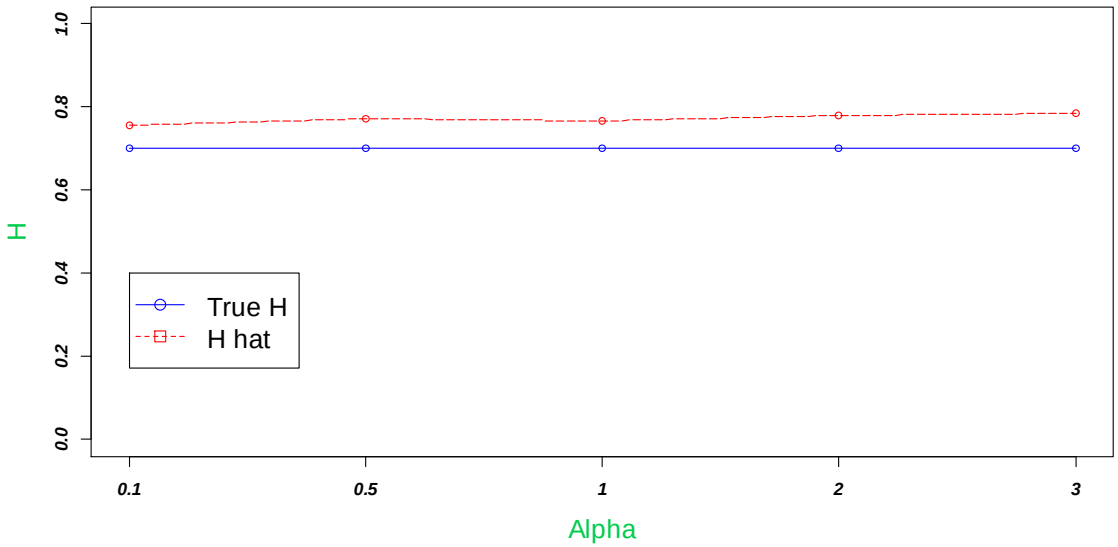

Figure 85. Change in H hat as α Increases Given True H=0.7, β=2

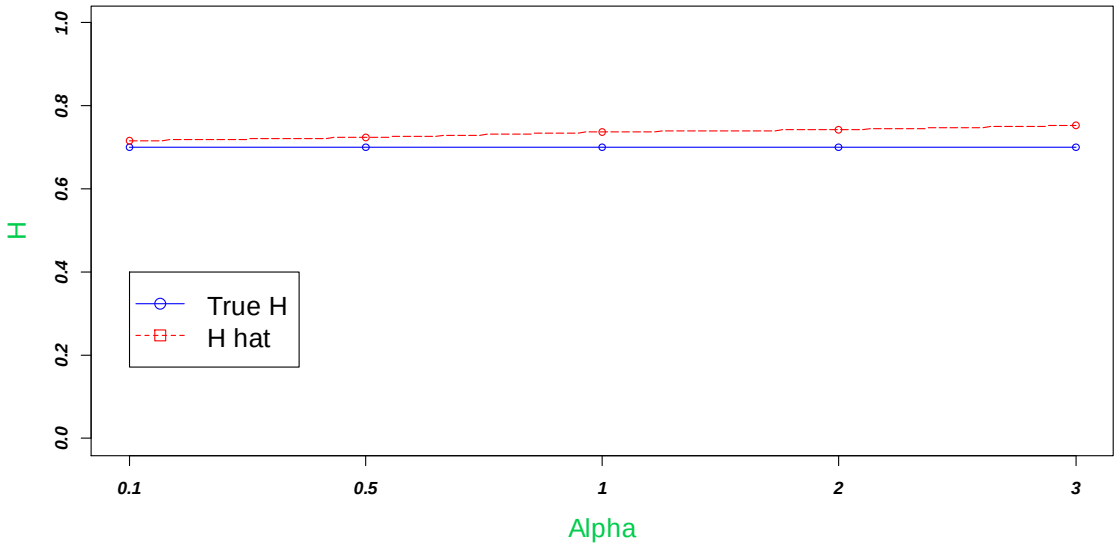

Figure 86. Change in H hat as α Increases Given True H=0.7, β=3

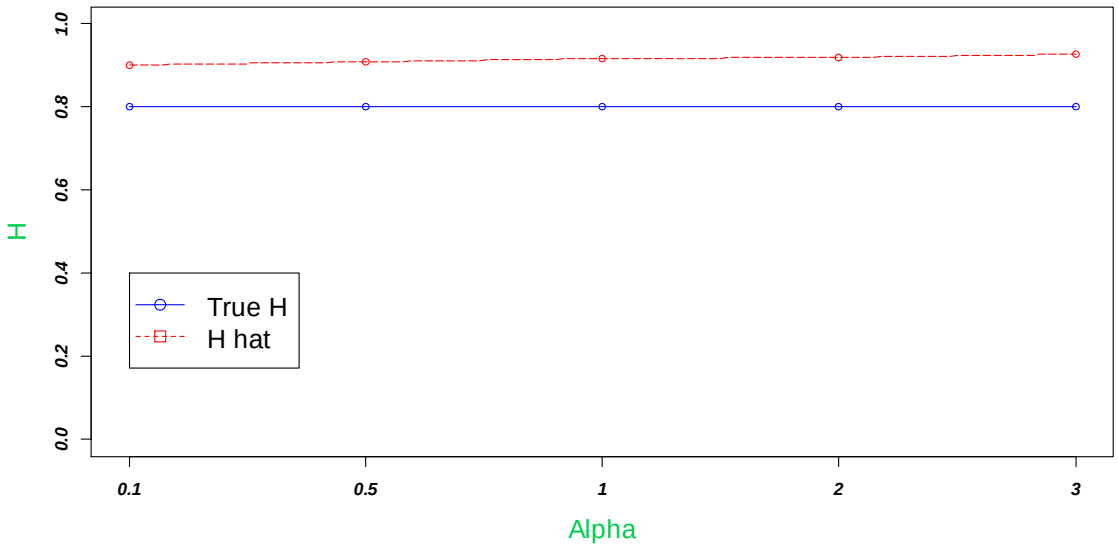

Figure 87. Change in H hat as α Increases Given True H=0.8, β=0.5

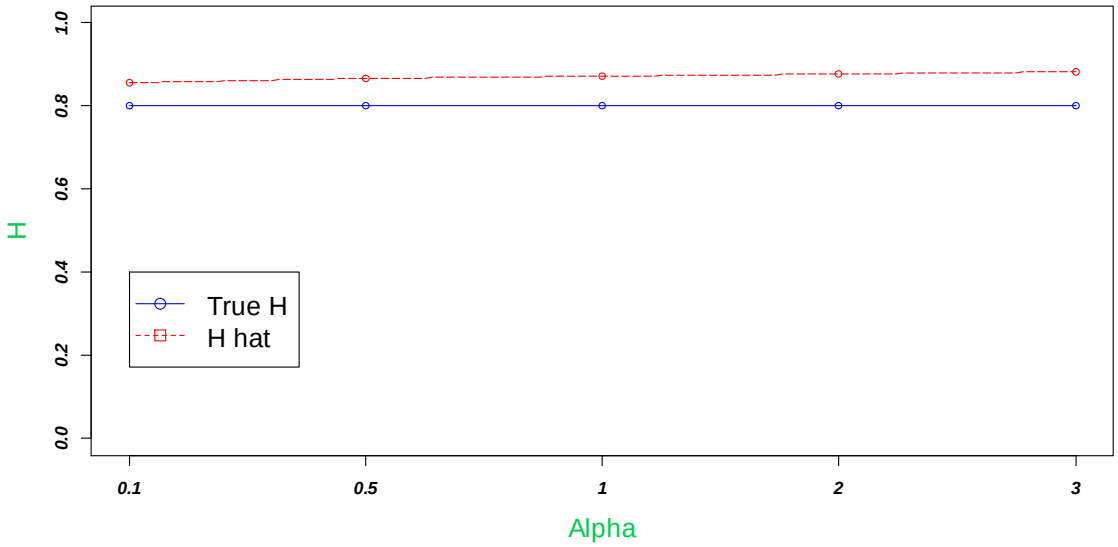

Figure 88. Change in H hat as α Increases Given True H=0.8, β=1

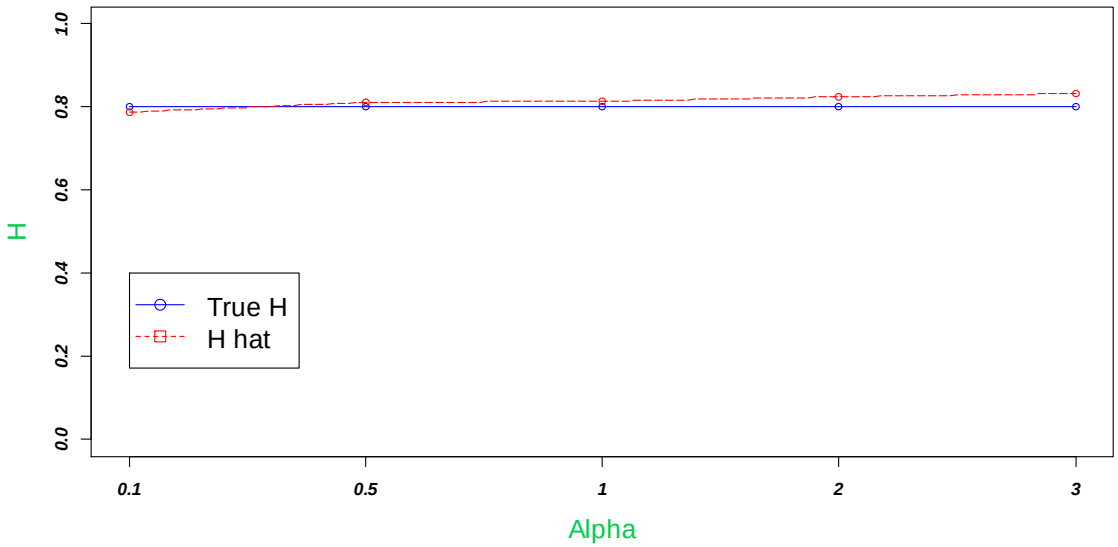

Figure 89. Change in H hat as α Increases Given True H=0.8, β=2

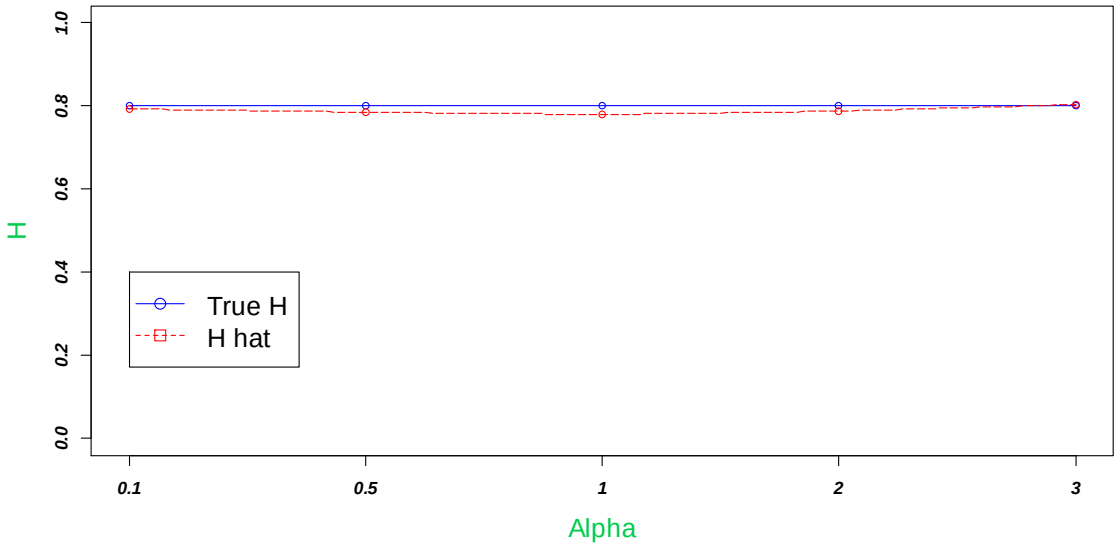

Figure 90. Change in H hat as α Increases Given True H=0.8, β=3

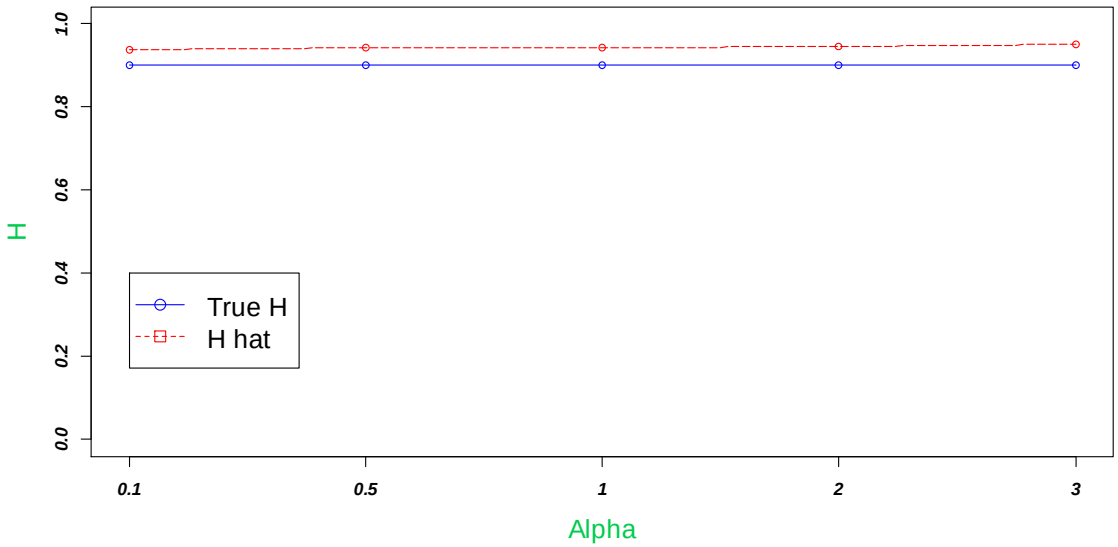

Figure 91. Change in H hat as α Increases Given True H=0.9, β=0.5

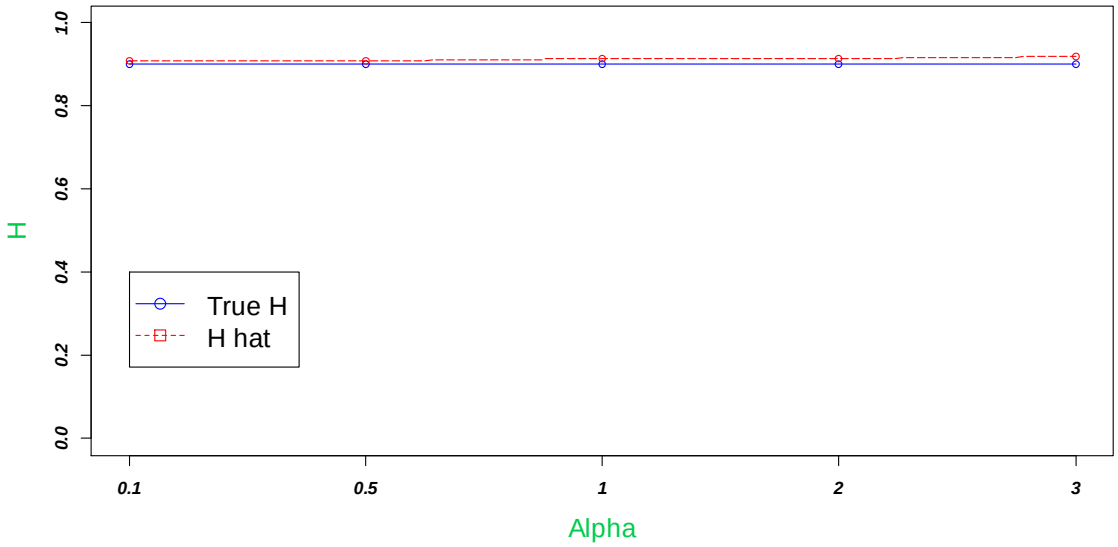

Figure 92. Change in H hat as α Increases Given True H=0.9, β=1

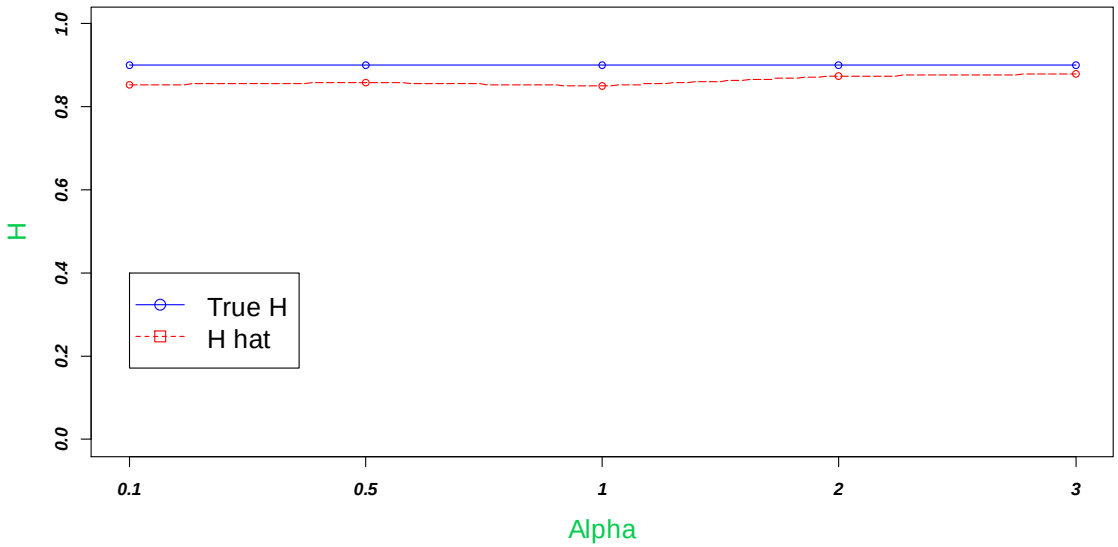

Figure 93. Change in H hat as α Increases Given True H=0.9, β=2

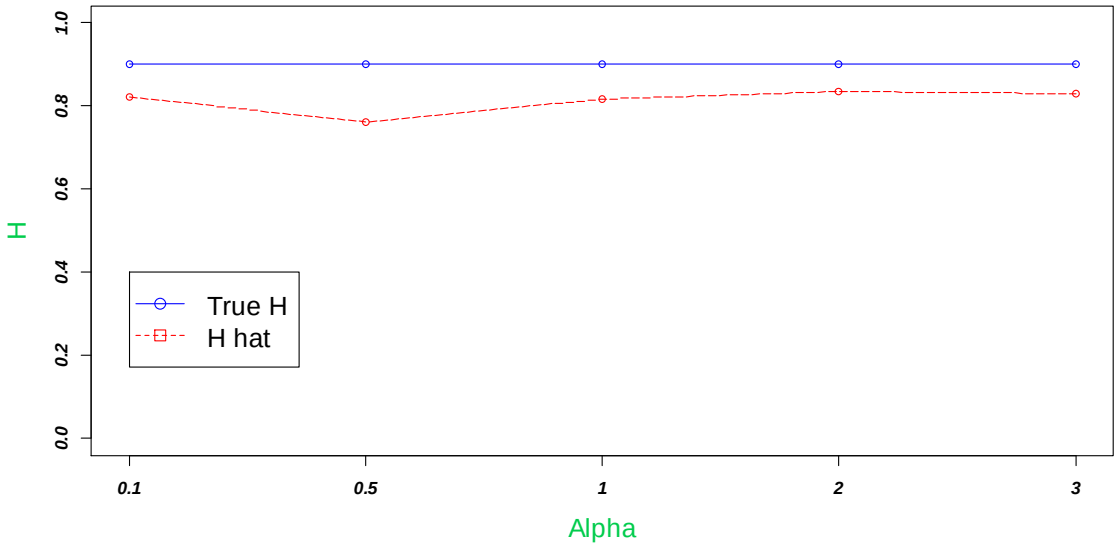

Figure 94. Change in H hat as α Increases Given True H=0.9, β=3

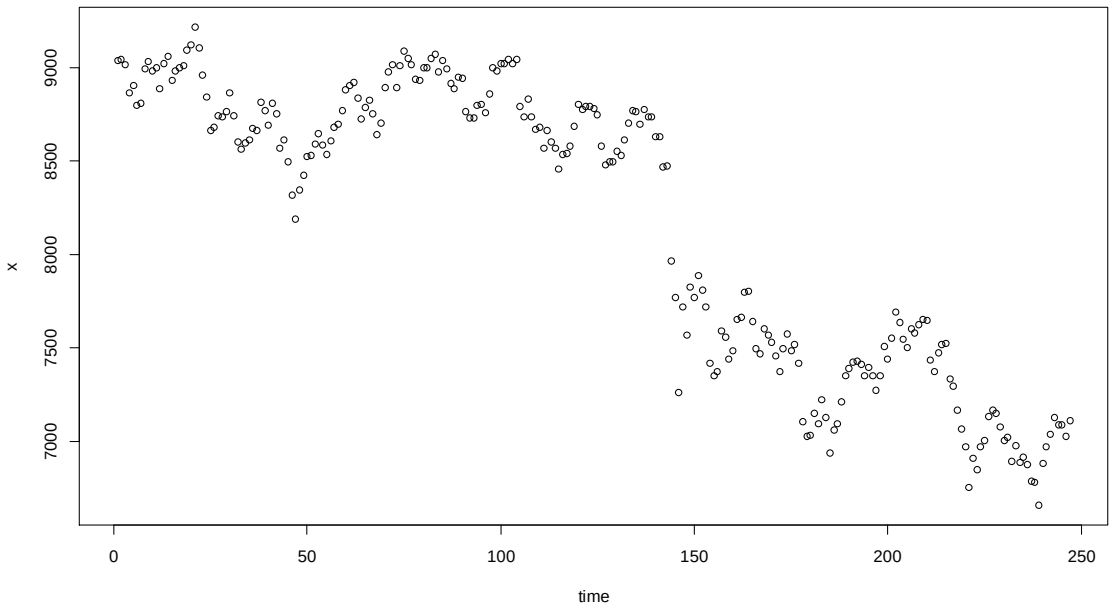

Figure 95. The Scatter Plot of 2011 Daily Taiwan Stock Index.

APPENDIX C

R CODE FOR SIMULATION

```
##### Simulation #####

library(mnormt)

library(dvfBm)

library(coda)

library(mcmcse)

        ############## Defining the Likelihood Function ##################

R=function(n,h){

R=matrix(0,n-1,n-1)

   for( i in 1:(n-1)){

   for( j in 1:(n-1)){

   R[i,j]=( i^(2*h)+j^(2*h)-abs(i-j)^(2*h))/(2*n^(2*h))

   } }

return(R)}

  L=function(x,h){

  n=length(x)

  m=rep(0,n-1)

  R=R(n,h)

x=x[2:n]
```

```
sigma2=as.numeric(t(x)%*%solve(R)%*%x/n)
  L=dmnorm(x/sqrt(sigma2),m ,R)
    L= ifelse(L==0, exp(-700), L)
return(L)  }
  ###########  Defining a Function for Metroplis-Hasting Algorithm   ############


  mh<-function(x, alpha, beta,h0, N)
##### Metroplis-Hasting Algorithm with Proposal Distribution of Beta Distribution
      #####
          {
            vec<- vector("numeric", N)
            vec[1]<- h0
            for (i in 2:N)
            {
              can<- rbeta(1, alpha, beta)
              aprob<- min(1, L(x, can)/L(x, h))
              u<- runif(1)
              vec[i]<- ifelse (u< aprob, can, vec[i-1])
```

```
        }

        vec

      }

########## Setting Number of Alpha, Beta, H and nsim, n, ###########

h0=.1

h1=.9

nh=9

nsim=10000

n=100

Alpha=c(.1,.5, 1, 2, 3)

Beta<-Alpha

for(i in 1:nh){

print(L(x,H[i])) }

H=seq(h0,h1,length.out=nh)

I=length(H)

J=length(Alpha)

K=length(Beta)

record = matrix(0,I*J*K,5)
```

```
for(i in 1:I)
{
  for(j in 1:J)
  {
     for(k in 1:K)
     {
print(c(i,j,k))

       h = H[i]            ##### Setting Hurst Dimesion ####
       alpha=Alpha[j]
       beta=Beta[k]
      x = circFBM(n, h,plotfBm = FALSE)

       vec<-mh (x, alpha , beta, h0, nsim)
##### Computing Hurst dimension using different alpha, beta, and number of iterations
      #####
       mean.hurst= mean(vec)
```

```
        hhh<-mcmc(data=vec, start=1, end=nsim, thin=1)
        remc = mcse(hhh, size="sqroot", g=NULL, method="bm", warn=FALSE)
      ##### Computing Monte Carlo Error #####
        record[J*K*(i-1)+J*(j-1)+k,1] = H[i]
        record[J*K*(i-1)+J*(j-1)+k,2] = alpha
        record[J*K*(i-1)+J*(j-1)+k,3] = beta
        record[J*K*(i-1)+J*(j-1)+k,4] = mean.hurst
        record[J*K*(i-1)+J*(j-1)+k,5] = remc$se
    }
  }
}
write.table(record,"temp.csv",append = TRUE,sep =" , ",row.name =FALSE,col.name
      =FALSE)
```

APPENDIX D

R CODE FOR REAL DATA

For Real Data

```
library(mnormt)

library(dvfBm)

library(coda)

library(mcmcse)

n=247

N=10000

h0=0.1

h=0.3

 x <-c(9039.63, 9045.11, 9014.32, 8866.23, 8905.25, 8798.05, 8810.77, 8992.01,
9034.17, 8985.34, 8998.89, 8889.84, 9023.30, 9059.80, 8935.29, 8983.59, 9001.74,
9010.41, 9093.54, 9122.16, 9220.69, 9107.96, 8962.07, 8842.41, 8666.61, 8679.12,
8740.32, 8738.71, 8763.69, 8867.16, 8744.81, 8604.32, 8561.16, 8597.87, 8611.79,
8675.37, 8661.63, 8815.52, 8769.87, 8692.10, 8810.35, 8754.41, 8566.91, 8616.29,
8493.83, 8317.38, 8190.50, 8343.79, 8421.66, 8523.27, 8529.93, 8589.71, 8645.37,
8587.99, 8538.44, 8605.83, 8683.69, 8697.84, 8772.13, 8884.84, 8904.44, 8922.75,
8839.22, 8727.95, 8787.75, 8828.63, 8752.56, 8643.40, 8704.48, 8893.07, 8980.43,
9015.91, 8894.01, 9012.28, 9088.97, 9050.49, 9014.33, 8937.30, 8930.24, 8997.73,
9001.80, 9048.25, 9073.88, 8977.77, 9039.55, 8996.70, 8917.25, 8885.40, 8952.25,
8944.38, 8767.30, 8730.23, 8732.30, 8797.29, 8806.29, 8757.66, 8860.58, 8999.84,
8981.71, 9024.54, 9025.21, 9044.85, 9020.60, 9042.35, 8791.67, 8737.24, 8833.86,
8739.84, 8671.45, 8679.15, 8571.93, 8663.86, 8601.15, 8567.46, 8456.36, 8536.13,
8540.35, 8582.08, 8684.39, 8801.51, 8777.22, 8790.94, 8795.00, 8782.72, 8749.55,
8581.43, 8480.26, 8495.51, 8494.70, 8550.16, 8528.20, 8612.53, 8705.84, 8770.95,
8764.68, 8700.59, 8775.18, 8739.83, 8737.72, 8629.97, 8629.09, 8466.35, 8472.37,
7962.87, 7769.60, 7261.54, 7717.58, 7566.39, 7825.80, 7769.17, 7886.93, 7807.67,
7721.19, 7414.36, 7348.71, 7369.43, 7592.51, 7559.47, 7438.52, 7482.92, 7654.37,
7665.22, 7799.18, 7801.28, 7637.90, 7496.24, 7468.29, 7601.29, 7568.22, 7530.00,
7455.50, 7374.58, 7494.29, 7572.83, 7485.60, 7515.66, 7419.49, 7101.34, 7026.49,
7031.13, 7148.38, 7090.12, 7222.16, 7128.66, 6937.24, 7061.17, 7094.61, 7210.92,
7348.86, 7390.06, 7423.18, 7430.50, 7412.88, 7352.12, 7393.73, 7347.93, 7272.51,
7352.43, 7506.08, 7437.77, 7548.36, 7693.24, 7637.48, 7547.24, 7500.54, 7603.45,
7581.61, 7623.16, 7652.47, 7646.12, 7431.26, 7369.73, 7472.16, 7515.50, 7520.95,
7333.57, 7296.09, 7166.26, 7064.61, 6967.66, 6748.92, 6906.34, 6844.30, 6966.58,
7001.59, 7132.41, 7164.00, 7149.58, 7075.84, 7004.18, 7018.64, 6891.47, 6975.62,
6883.08, 6912.54, 6874.18, 6783.02, 6780.64, 6654.67, 6878.63, 6968.22, 7035.10,
7125.04, 7085.50, 7086.10, 7026.86, 7109.85)
```

```
time=numeric(n)


 for( i in 1:n){
 {
 time[i]=i
 } }
 time

        ############## defining the likelihood function ##################


R=function(n,h){
R=matrix(0,n-1,n-1)


 for( i in 1:(n-1)){
   for( j in 1:(n-1)){


 R[i,j]=( i^(2*h)+j^(2*h)-abs(i-j)^(2*h))/(2*n^(2*h))


 } }


return(R)}


 L=function(x,h){
```

```
  n=length(x)
  R=R(n,h)
x=x[2:n]
time=time[2:n]
R1=R+diag(0.1, n-1)

sigma2=as.numeric(t(x)%*%solve(R1)%*%x/n)
est.mean=as.numeric((t(time)%*%solve(R1)%*%x)/(t(time)%*%solve(R1)%*%time))
m=rep(est.mean,n-1)



L=dmnorm(x/sqrt(sigma2),m ,R1)



L= ifelse(L==0, exp(-700), L)
return(L)  }

################  defining a function for Metroplis-Hasting algorithm   #############
##### Metroplis-Hasting algorithm with proposal distribution of beta distribution #####

  mh<-function(x, alpha, beta, N)
            {
```

```
            vec<- vector("numeric", N)
            vec[1]<- h0
            for (i in 2:N)
            {
              can<- rbeta(1, alpha, beta)
              aprob<- min(1, L(x, can)/L(x, h))
              u<- runif(1)
              vec[i]<- ifelse (u< aprob, can, vec[i-1])

            }
            vec
          }
 vec<-mh (x, 0.1, 3, 10000)
##### Computing Hurst dimension using different alpha, beta, and number of iterations
#####
 mean.hurst=mean(vec)

 library(coda)
 hhh<-mcmc(data=vec, start=1, end=10000, thin=1)

 summary(hhh)       ##### Give summary statistics #####
mcse(hhh, size="sqroot", g=NULL, method="bm", warn=FALSE)
##### Computing Monte Carlo Error #####
```