Estimation of a Bivariate Distribution under Univariate Censoring

Jiang Mei

Thesis submitted to the Faculty of Graduate and Postdoctoral Studies
in partial fulfillment of the requirements for the degree of Master of Science in Mathematics [1]

Department of Mathematics and Statistics
Faculty of Science
University of Ottawa

[1]The M.Sc. program is a joint program with Carleton University, administered by the Ottawa-Carleton Institute of Mathematics and Statistics

Abstract

We compare four estimators of a bivariate distribution function H on $\Re^2_+$ when both components of the data point (X, Y) are subject to censoring by the same (univariate) random variable C with distribution G. We use the same simulated data to calculate each of the four estimators for 5 different FGM copulas. Finally we find the best of the four estimators, that is the adapted path dependent estimator. We will show that it is not necessarily a good idea to use all the available information about C and that our estimator of H can be improved by using a worse estimator of G.

Acknowledgements

First and foremost I am heartily thankful to my supervisor Professor Gail Ivanoff, whose encouragement, guidance and support from the initial to the final level enabled me to develop an understanding of the subject. Her way of removing misunderstanding gave me the opportunity to pursue this work in the best conditions. One simply could not wish for a better or friendlier supervisor. Thank you deeply.

I also offer my regards to all of those who supported me in any respect throughout the process of writing this thesis, such as Mr. Farid Elktaibi, Mr. Jesse Collingwood, and so on.

Lastly, I wish to express my love and gratitude to my beloved family for their understanding and endless love throughout the duration of my studies.

Dedication

This work is dedicated to my son Di Mei, to my lovely wife Manqiong Wang.

Contents

Chapter 1

Introduction

The goal of this thesis is to compare four estimators of a bivariate distribution function H on $\Re_+^2$ when both components of the data point (X, Y) are subject to censoring by the same (univariate) random variable C with distribution G. We will show that it is not necessarily a good idea to use all the available information about C and that our estimator of H can be improved by using a worse estimator of G.

Most of the literature on multivariate survival analysis deals with estimation of the survival function S, where $S(x, y) = P(X > x, Y > y)$. In dimensions great than 1, it is not true that $S = 1-H$; also the survival function depends on both the hazard function (dH/S) and the marginal distributions (see [4], equation (14.4)). Various hazard based estimators have been proposed, but these estimators are complex and can be non-monotonic (in other words, negative mass can be put on some points). See [4] for details. A second approach to estimation of the survival function is the path-dependent approach in which the estimator of $S(x, y)$ depends on a specified path from $(0, 0)$ to (x, y). A good description of various path-dependent estimators can be found in [3] where it is pointed out that these estimators can also be non-monotonic.

Because of the problems with the estimation of the bivariate survival function, two approaches to the estimation of a bivariate distribution function H under various types of censoring mechanisms were studied in [2] and [3] . The general model is that there is a random set ξ on which (X, Y) can be observed, but if $(X, Y) \in \xi^c$, the point is either completely or partially obscured. The first approach is the reverse probability method (cf. [2]). The idea behind the reverse probability estimator of [2] is to weight each uncensored observation (x, y) by a suitable estimator of $1/P((x, y) \in \xi)$. This method can be used in two situations: either the random set ξ is always fully observed, or ξ takes its values in a totally ordered class of sets (the order is set-inclusion). The second approach is the path-dependent estimator of [3] . The idea behind this estimator of $H(x, y)$ is to project observations of (X, Y) and ξ onto a path from $(0, 0)$ to (x, y) to reduce the data points and censoring sets to one dimension; now the Kaplan-Meier estimator of the survival function of the one dimensional data can be used to estimate $1 - H(x, y)$. This method has the advantage of producing an adapted estimator: only information available in $[0, x] \times [0, y]$ is required to estimate $H(x, y)$. However, for certain censoring models there can be problems with observability and monotonicity of the path-dependent estimator; these issues are examined in detail in [3] .

In the case of univariate censoring, we have $\xi = [0, C] \times [0, C]$ in the model above and both approaches can be used, yielding three different estimators. For the reverse probability method, we have $P((x, y) \in \xi) = 1 - G(x \vee y)$, so if G is known we have a fourth potential estimator. It is important to note that in this case all four estimators of H are monotone(there is no negative mass), unlike estimators of the survival function.

Outline of the thesis:

Chapter 2 - In this chapter, we present a mathematical description of the model and formal definitions of the four estimators. The asymptotic behavior of the estimators is considered, including a comparison of the asymptotic variances, where possible.

Chapter 3 - Here we present the algorithms and R code required for an empirical comparison of our estimators. First, we describe how to simulate a sample from a bivariate distribution given the copula and marginals. This is illustrated for a distribution with a Farlie-Gumbel-Morgenstern(FGM) copula and exponential marginals. Next, we present the R code used to calculate each of the four estimators of $H(x, y)$ using the same simulated samples of (X, Y) and C.

Chapter 4 - We present a detailed comparison of the results of using the same simulated data to calculate each of the four estimators for five different FGM copulas.

Chapter 5 - Conclusion and recommendations: Our study leads us to a somewhat surprising conclusion: we should use the worst estimate of G in order to obtain the best estimate of H. Even if G or the empirical distribution $\hat{G}_n$ is known, neither one should be used.

Appendix A - We present the results of our simulations, including tables of average values and MSE for all four estimators, based on 1000 samples of size 50 for 5 different values of the FGM copula (see Tables 1—60), tables giving bias (see Tables 61—64) and tables comparing MSE of the estimators (see Tables 65—70).

Appendix B - This appendix contains the overall R code for producing the simulations and Tables of Appendix A.

Chapter 2

The Model and Definition of the Estimators

We have n test subjects and we have two values $(X, Y) \in \Re_+^2$ of interest for each subject (for example, age of onset of two different diseases, or age of onset of cataracts in each eye). Each of X and Y are censored by the same variable $C \in \Re_+$, the time at which the subject leaves the study for any reason. We observe $(X \wedge C, Y \wedge C)$, $I(X \leq C)$, $I(Y \leq C)$. We assume that (X, Y) and C are independent. Let H denote the (bivariate) distribution of (X, Y) and let G denote the (univariate) distribution of C. Assume that both H and G are continuous, with densities h and g, respectively. Given the data $(X_i \wedge C_i, Y_i \wedge C_i)$, $i = 1, ..., n$, we propose four different estimators of H. The first three are reverse probability estimators and the fourth is an adapted estimator. We begin by describing the reverse probability method.

Method 1 : Reverse Probability Method.(cf. [2])

Let $(X, Y) \sim H$ and given a sample $(X_i, Y_i), i = 1, ..., n$, the empirical distribution function is

$$H_n(x,y) = \frac{1}{n} \cdot \sum_{i=1}^{n} I(X_i \le x, Y_i \le y).$$

We observe (X, Y) only if $(X \le C, Y \le C)$, and so the observable sub-distribution of uncensored observations is

$$\tilde{H}(x,y) = P(X \le x, Y \le y, X \le C, Y \le C).$$

We can write $\tilde{H}(x,y) = E[P(X \le x, Y \le y, X \le C, Y \le C|(X,Y))]$.
Now,

$$\begin{aligned} & P(X \le x, Y \le y, X \le C, Y \le C|(X = s, Y = t)) \\ = \;& P(s \le C, t \le C) \cdot I(s \le x, t \le y) \\ = \;& P(C \ge s \vee t) \cdot I(s \le x, t \le y) \\ = \;& (1 - G(s \vee t)) \cdot I(s \le x, t \le y). \end{aligned} \tag{2.0.1}$$

Using (2.0.1), we see that

$$\tilde{H}(x,y) = \int_0^y \int_0^x (1 - G(s \vee t)) dH(s,t)$$

$$d\tilde{H}(x,y) = (1 - G(x \vee y)) dH(x,y)$$

$$dH(x,y) = (1/(1 - G(x \vee y))) d\tilde{H}(x,y)$$

$$H(x,y) = \int_0^y \int_0^x (1/(1 - G(s \vee t))) d\tilde{H}(s,t). \tag{2.0.2}$$

The idea is to find suitable estimators for quantities on the right hand side of (2.0.2). $\tilde{H}$ can be estimated by its empirical counterpart:

$$\tilde{H}_n(x,y) = \frac{1}{n} \cdot \sum_{i=1}^{n} I(X_i \le x, Y_i \le y) \cdot I((X_i \vee Y_i) \le C_i). \tag{2.0.3}$$

If (X_i, Y_i) is observed, $\tilde{H}_n(x,y)$ gives this point mass $1/n$. Next, we must weight each observed value of (X_i, Y_i) by $1/W_i$, where W_i is an appropriate estimate of $(1 - G(X_i \vee Y_i))$ and the estimate of $H(x,y)$ will be the form

$$\hat{H}_n(x,y) = \frac{1}{n} \sum_{i=1}^{n} \frac{I(X_i \le x, Y_i \le y) I(X_i \vee Y_i \le C_i)}{W_i}. \tag{2.0.4}$$

We now consider three possible situations:

Case A: We know the distribution G of C.
Therefore, let $W_i = 1 - G(X_i \vee Y_i)$ in (2.0.4) to obtain the estimator

$$\hat{H}_n^{(1)}(x,y) = \frac{1}{n} \sum_{i=1}^{n} \frac{I(X_i \le x, Y_i \le y) I(X_i \vee Y_i \le C_i)}{1 - G(X_i \vee Y_i)}. \tag{2.0.5}$$

Case B: G is unknown, but $C_1, C_2 \ldots$ are all observed. We estimate W_i using the empirical distribution: $G_n(c) = \frac{1}{n} \sum_{i=1}^{n} I(C_i \le c)$.
Now, $W_i = 1 - G_n(X_i \vee Y_i) = \frac{1}{n} \sum_{j=1}^{n} I(X_i \vee Y_i < C_j)$ in (2.0.4). In this case, the estimator becomes

$$\hat{H}_n^{(2)}(x,y) = \sum_{i=1}^{n} \frac{I(X_i \le x, Y_i \le y) I(X_i \vee Y_i \le C_i)}{\sum_{j=1}^{n} I(C_j > X_i \vee Y_i)}. \tag{2.0.6}$$

Case C: G is unknown and C_i is observed only if (X_i, Y_i) is censored: i.e., we observe (X_i, Y_i) only when $X_i \leq C_i$ and $Y_i \leq C_i$, and C_i is observed only if either $X_i > C_i$ or $Y_i > C_i$.

Reduce the problem to one dimension. Let $V_i = X_i \vee Y_i$; we observe V_i if $V_i \leq C_i$, otherwise we observe C_i. Let $Z_i = V_i \wedge C_i$.

Reverse the roles of V and C: We treat C as the variable of interest and V as the censoring variable. We estimate G using the usual Kaplan Meier (K-M) estimator of $(1 - G) = S_G$. (See [1] pg.256, for example.)

$$\hat{S}_G(c) = \Pi_{i:Z_i \leq c} \left(1 - \frac{I(Z_i = C_i)}{\sum_{j=1}^{n} I(Z_j \geq Z_i)} \right).$$

Now, $W_i = \hat{S}_G(X_i \vee Y_i)$, and the estimator is

$$\hat{H}_n^{(3)}(x, y) = \frac{1}{n} \sum_{i=1}^{n} \frac{I(X_i \leq x, Y_i \leq y) I(X_i \vee Y_i \leq C_i)}{\Pi_{j:Z_j \leq X_i \vee Y_i} \left(1 - \frac{I(Z_j = C_j)}{\sum_{h=1}^{n} I(Z_h \geq Z_j)} \right)}. \tag{2.0.7}$$

Method 2 : Adapted path-dependent estimator (cf. [3])

In Case C, the estimator of $H(x, y)$ requires data from $[0, x \vee y] \times [0, x \vee y]$. The adapted estimator uses only data available in $[0, x] \times [0, y]$, and is defined by projecting observations of (x, y) onto a path from $(0, 0)$ to (x, y). The general idea behind the path-dependent approach is described in detail in [3]. To construct an adapted estimator of $H(x, y)$, a continuous map (or path) $\gamma = (\gamma_1, \gamma_2)$: $\Re_+ \longrightarrow \Re_+^2$ is defined such that γ_1 and γ_2 are continuous and strictly increasing, $\gamma_1(0) = \gamma_2(0) = 0$, and for some $t \in \Re$, $\gamma_1(t) = x$ and $\gamma_2(t) = y$. The projection of (X, Y) onto the path is defined by

$$\vee^{\gamma} := \inf\{s : X \leq \gamma_1(s), Y \leq \gamma_2(s)\}. \tag{2.0.8}$$

Then if $\gamma(t) = (x, y)$,

$$\begin{aligned} P(\vee^{\gamma} \leq t) &= P(X \leq \gamma_1(t), Y \leq \gamma_2(t)) \\ &= P(X \leq x, Y \leq y) \\ &= H(x, y), \end{aligned} \tag{2.0.9}$$

and it is enough to estimate $P(\vee^{\gamma} \leq t)$. Since only observations of $H(x, y)$ such that $X \leq x$, and $Y \leq y$ are used, the resulting estimator will be adapted. For the univariate censoring model, as shown in [3] the appropriate path to (x, y) is defined by

$$\gamma(s) = (s \wedge x, s \wedge y)$$

for

$$0 \leq s \leq x \vee y,$$

and in any convenient way for $s > x \vee y$. We have $\vee^{\gamma} = X \vee Y$ if $X \leq x$, and $Y \leq y$. Therefore, if $\vee^{\gamma} \leq x \vee y$, then $\vee^{\gamma} \leq C$ if and only if $X \leq C$ and $Y \leq C$, and so $\vee^{\gamma}$ is censored by C if and only if (X, Y) is censored. We can now estimate $1 - H(x, y) = P(\vee^{\gamma} > t)$ using one-dimensional survival analysis. Suppressing dependence on γ in our notation, we have

Case D: In $[0, x] \times [0, y]$ we will have observed:

1. (X_i, Y_i) if $X_i \leq x$, $Y_i \leq y$ and $V_i = (X_i \vee Y_i) \leq C_i$

2. C_i if $C_i \leq x \vee y$, and $C_i \leq (X_i \vee Y_i) = V_i$

Any test subject not satisfying either of the criteria above will be considered still at risk at (x, y).

We use those observations of V_i and C_i in $[0, x] \times [0, y]$ to calculate a one-dimensional K-M estimator of $1 - H$, which leads to the following adapted estimator of H:

$$\hat{H}_n^{(4)}(x, y) = 1 - \Pi_{i:(X_i \leq x \wedge C_i, Y_i \leq y \wedge C_i)} \left(1 - \frac{1}{R_i}\right) \tag{2.0.10}$$

where

$R_i = n + 1 - \Sigma_{j=1}^n I(C_j \leq (V_i \wedge V_j)) - \Sigma_{j=1}^n I(X_j \leq (V_i \wedge C_j \wedge x, Y_j \leq V_i \wedge C_j \wedge y)$.

It is proven in [3] that $\hat{H}_n^{(4)}(x, y)$ is a measure: there is positive mass on each observed value (X_i, Y_i) and non-negative mass on $(X_i \vee X_j, Y_i \vee Y_j)$ when both (X_i, Y_i) and (X_j, Y_j) are fully observed.

Next we consider the asymptotic behavior of our four estimators.

Mean and Variance of Estimators of $H(x, y)$.

Case A. From (2.0.5), we have $\hat{H}_n^{(1)}(x, y) = \frac{1}{n} \sum_{i=1}^n \frac{I(X_i \leq x, Y_i \leq y) I(X_i \vee Y_i \leq C_i)}{1 - G(X_i \vee Y_i)}$.

$$\begin{aligned}
E(\hat{H}_n^{(1)}(x, y)) &= \frac{1}{n} \sum_{i=1}^n E\left(E[\frac{I(X_i \leq x, Y_i \leq y) I(X_i \vee Y_i \leq C_i)}{1 - G(X_i \vee Y_i)} \mid X_i, Y_i]\right) \\
&= \frac{1}{n} \sum_{i=1}^n E\left(I(X_i \leq x, Y_i \leq y) \frac{E[I(X_i \vee Y_i \leq C_i) \mid X_i, Y_i]}{1 - G(X_i \vee Y_i)}\right) \\
&= \frac{1}{n} \sum_{i=1}^n E\left(I(X_i \leq x, Y_i \leq y)\right) \\
&= \frac{1}{n} \cdot \sum_{i=1}^n H(x, y) \\
&= H(x, y).
\end{aligned}$$

$$Var(\hat{H}_n^{(1)}(x,y)) = \frac{1}{n} Var\left(\frac{I(X \le x, Y \le y) I(X \vee Y \le C)}{1 - G(X \vee Y)}\right) = \frac{1}{n}\sigma_1^2(x,y),$$

where

$\sigma_1^2(x,y) = \left(E[Var\left(\frac{I(X\le x, Y\le y)I(X\vee Y\le C)}{1-G(X\vee Y)} \mid X,Y\right)] + Var[E\left(\frac{I(X\le x, Y\le y)I(X\vee Y\le C)}{1-G(X\vee Y)} \mid X,Y\right)]\right).$

Because

$$Var\left(\tfrac{I(X\le x, Y\le y)I(X\vee Y\le C)}{1-G(X\vee Y)} \mid X,Y\right)$$

$$= E\left(\tfrac{I^2(X\le x, Y\le y)I^2(X\vee Y\le C)}{(1-G(X\vee Y))^2} \mid X,Y\right) - \left(E\left(\tfrac{I(X\le x, Y\le y)I(X\vee Y\le C)}{1-G(X\vee Y)} \mid X,Y\right)\right)^2$$

$$= \tfrac{I(X\le x, Y\le y)}{1-G(X\vee Y)} - I(X \le x, Y \le y)$$

so,

$$\begin{aligned} & E\left[Var\left(\frac{I(X \le x, Y \le y) I(X \vee Y \le C)}{1 - G(X \vee Y)} \mid X, Y\right)\right] \\ &= E\left[\frac{I(X \le x, Y \le y)}{1 - G(X \vee Y)} - I(X \le x, Y \le y)\right] \\ &= \int_0^y \int_0^x \frac{h(s,t)}{1 - G(s \vee t)} ds dt - H(x,y). \end{aligned}$$

Also,

$$\begin{aligned} Var\left[E\left(\frac{I(X \le x, Y \le y) I(X \vee Y \le C)}{1 - G(X \vee Y)} \mid X, Y\right)\right] &= Var[I(X \le x, Y \le y)] \\ &= H(x,y)(1 - H(x,y)). \end{aligned}$$

Finally,

$$\sigma_1^2(x,y) = \int_0^y \int_0^x \frac{h(s,t)}{1 - G(s \vee t)} ds dt - H^2(x,y). \tag{2.0.11}$$

Case B. From (2.0.6), $\hat{H}_n^{(2)}(x,y) = \sum_{i=1}^{n} \frac{I(X_i \le x, Y_i \le y) I(X_i \vee Y_i \le C_i)}{\sum_{j=1}^{n} I(C_j > X_i \vee Y_i)}$.

It is shown in [2] that $E[\hat{H}_n^{(2)}(x,y)] \longrightarrow H(x,y)$ and

$Var(\hat{H}_n^{(2)}(x,y)) = \frac{1}{n}\sigma_2^2(x,y) + \circ(n^{-1})$

where,

$\sigma_2^2(x,y) = \int_0^y \int_0^x \frac{h(s,t)}{1-G(s\vee t)} dsdt - \int_0^y \int_0^x \int_0^y \int_0^x \frac{(1-G(s\vee t\vee u\vee v))}{(1-G(s\vee t))(1-G(u\vee v))} h(s,t)h(u,v)dsdtdudv.$

Since

$$\sigma_2^2(x,y) = \left(\int_0^y \int_0^x \frac{h(s,t)}{1-G(s\vee t)} dsdt - H^2(x,y) \right)$$

$$- \left(\int_0^y \int_0^x \int_0^y \int_0^x \frac{(1-G(s\vee t\vee u\vee v))}{(1-G(s\vee t))(1-G(u\vee v))} h(s,t)h(u,v)dsdtdudv - H^2(x,y) \right),$$

it is straightforward to show that

$$\sigma_2^2(x,y) = \sigma_1^2(x,y) - Var\left(\int_0^y \int_0^x \frac{h(s,t)I(s\vee t \le C)}{1-G(s\vee t)} dsdt \right). \tag{2.0.12}$$

Case C. From (2.0.7), $\hat{H}_n^{(3)}(x,y) = \frac{1}{n}\sum_{i=1}^{n} \frac{I(X_i \le x, Y_i \le y) I(X_i \vee Y_i \le C_i)}{\Pi_{j: Z_j \le X_i \vee Y_i} \left(1 - \frac{I(Z_j = C_j)}{\sum_{h=1}^{n} I(Z_h \ge Z_j)}\right)}$.

It is shown in [2] that $E[\hat{H}_n^{(3)}(x,y)] \longrightarrow H(x,y)$ and
$Var(\hat{H}_n^{(3)}(x,y)) = \frac{1}{n}\sigma_3^2(x,y) + \circ(n^{-1})$

where,

$\sigma_3^2(x,y) = \int_0^y \int_0^x \frac{h(s,t)}{1-G(s\vee t)} dsdt - H^2(x,y) - \int_0^{x\vee y} \frac{(H(x,y)-H(x\wedge s, y\wedge s))^2}{(1-G(s))^2(1-H(s,s))} g(s)ds.$
We observe that

$$\sigma_3^2(x,y) = \sigma_1^2(x,y) - \int_0^{x\vee y} \frac{(H(x,y)-H(x\wedge s, y\wedge s))^2}{(1-G(s))^2(1-H(s,s))} g(s)ds. \tag{2.0.13}$$

Case D. From (2.0.10), $\hat{H}_n^{(4)}(x,y) = 1 - \Pi_{i:(X_i \le x\wedge C_i, Y_i \le y \wedge C_i)} \left(1 - \frac{1}{R_i}\right).$
It is shown in [3] that $E[\hat{H}_n^{(4)}(x,y)] \longrightarrow H(x,y)$ and
$Var(\hat{H}_n^{(4)}(x,y)) = \frac{1}{n}\sigma_4^2(x,y) + \circ(n^{-1})$
where,

$$\sigma_4^2(x,y) = (1-H(x,y))^2 \int_0^{x\vee y} \frac{dH(x\wedge u, y\wedge u)}{(1-G(u))(1-H(x\wedge u, y\wedge u))^2}. \tag{2.0.14}$$

We have $\sigma_1^2(x,y) > \sigma_2^2(x,y)$ and $\sigma_1^2(x,y) > \sigma_3^2(x,y)$. We are not able to compare $\sigma_2^2(x,y)$, $\sigma_3^2(x,y)$, and $\sigma_4^2(x,y)$ analytically and so we will do these empirically.

Central Limit Theorems:

The CLT for Case A follows from the usual CLT since the summands are i.i.d. Case B, and C are proven in [2] and case D is proven in [3].

Case A. $n^{1/2}(\hat{H}_n^{(1)}(x,y) - H(x,y)) \longrightarrow_{\mathcal{D}} N(0, \sigma_1^2(x,y)).$
Case B. $n^{1/2}(\hat{H}_n^{(2)}(x,y) - H(x,y)) \longrightarrow_{\mathcal{D}} N(0, \sigma_2^2(x,y)).$
Case C. $n^{1/2}(\hat{H}_n^{(3)}(x,y) - H(x,y)) \longrightarrow_{\mathcal{D}} N(0, \sigma_3^2(x,y)).$
Case D. $n^{1/2}(\hat{H}_n^{(4)}(x,y) - H(x,y)) \longrightarrow_{\mathcal{D}} N(0, \sigma_4^2(x,y)).$

Chapter 3

Simulations of (X, Y) and C

In this section we compare the four estimators defined above via simulations. First, we show how to simulate $(X, Y) \sim H$ using the copula and marginal distributions of H. The censoring variable $C \sim G$ is simulated by letting $C = G^{-1}(U)$, where $U \sim U[0, 1]$. We then describe the algorithms used to calculate each of the estimators and present the R-code. Finally, in Appendix A, we present results of 1000 simulations of censored samples of size 50 from a bivariate distribution with exponential marginals and a FGM (Farlie-Gumbel-Morgenstern) copula, censored by an independent exponential random variable. In each case, we use the same simulated data to calculate the estimators of the distribution. The overall code is given in Appendix B.

First, we demonstrate how to simulate observations from a continuous bivariate distribution H. We use the representation

$$H(x, y) = C_H(H_X(x), H_Y(y))$$

where C_H is a copula, and H_X and H_Y are the marginal distributions. A copula is any distribution on $[0, 1]^2$ with uniform marginal distributions. If $(X, Y) \sim H$, then $(U, V) \sim C_H$, where $U = H_X(X)$ and $V = H_Y(Y)$. Conversely, if $(U, V) \sim C_H$, then

$(X, Y) \sim H$, where $X = H_X^{-1}(U)$ and $Y = H_Y^{-1}(V)$. (See [5] for details.)

We assume that C_H is a FGM copula with parameter q, $-1 \leq q \leq 1$:

$$C_H(u, v) = C_{U,V}(u, v) = uv + quv(1-u)(1-v), 0 \leq u, v \leq 1.$$

The corresponding pdf of $C_H(u, v)$ is $c_H(u, v)$:

$$c_H(u, v) = c_{U,V}(u, v) = 1 + q(1-2u)(1-2v), 0 \leq u, v \leq 1.$$

To simulate a pair $(X, Y) \sim H$, first we simulate $(U, V) \sim C_H$, and then let $X = H_X^{-1}(U)$ and $Y = H_Y^{-1}(V)$. To simulate U and V, let $U \sim U[0, 1]$. Therefore, the conditional pdf of V given $U = u$ is c_H, and this density is used to simulate an observation of V.

To demonstrate, we assume that the marginal distributions are exponential distributions with parameters a and b, $a, b > 0$.
We have

$$\begin{aligned} H_{X,Y}(x, y) &= C_H(H_X(x), H_Y(y)) \\ &= (1 - \exp(-ax))(1 - \exp(-by))(1 + q \times \exp(-ax - by)). \end{aligned}$$

Let $U = H_X(X) = 1 - \exp(-aX); V = H_Y(Y) = 1 - \exp(-bY)$.
The inverse functions are therefore

$$X = -\ln(1-U)/a; Y = -\ln(1-V)/b$$

.

In what follows, we assume that $X \sim \exp(1)$, $Y \sim \exp(2)$ and $C \sim \exp(0.5)$. The value of q will vary.

The procedure to simulate a sample of size 50 is as follows:

a.) Generate 50-dimensional vector $[U_1, ..., U_{50}]$, where the $U_i's$ are i.i.d $U[0, 1]$.

R-codes:

q=0.5; # given q:relation between U and V

n=50; # sample size

$U \leftarrow$ runif(n); # generate n-dimension of vector $U \sim U(0, 1)$

b). Use the rejection method to create 50-dimensional vector $V \sim f_{V|U}(v|u) = c_H(u, v)$ using the formula for every given u to get the corresponding value of v. Since $1 \geq u \geq 0$ and $1 \geq v \geq 0$, so $1 \geq |1 - 2u|$ and $1 \geq |1 - 2v|$ and $1 + |q| \geq |1 + q(1 - 2u)(1 - 2v)|$. Therefore, we have $|c_H(u, v)/(1 + |q|)| \leq 1$.

The rejection algorithm runs as follows:

1. Sample $W \sim U[0, 1]$.
2. Sample $U \sim U[0, 1]$.
3. If $U < f(W)/K * g(W)$, let $K = 1 + |q|, g = 1, f(W) = c_H(u, v)$.

then set $V = W$. Otherwise, return to 1.

R-codes:

```
# Given U ~ U(0-1),generate #n of V by f(u,v)=1+q(1-2u)*(1-2v)

RejectionSampling <- function(w1,q,n1) { RN <- NULL   #a2
for(i in 1:n1)
```

```
{ OK <- 0
while(OK<1)
{x1 <- runif(1)
U1 <- runif(1)
r1=(1+q*(1-2*w1)*(1-2*x1))/(1+abs(q))
if(U1<r1)
{
OK <- 1; RN <- c(RN,x1)
}
}
}

RN }      #a2
```

c). Use the Inverse Transform Algorithm to get X and Y, assuming a=1, b=2.

R-codes:

a=1

b=2

X=-(log(1-U))/a

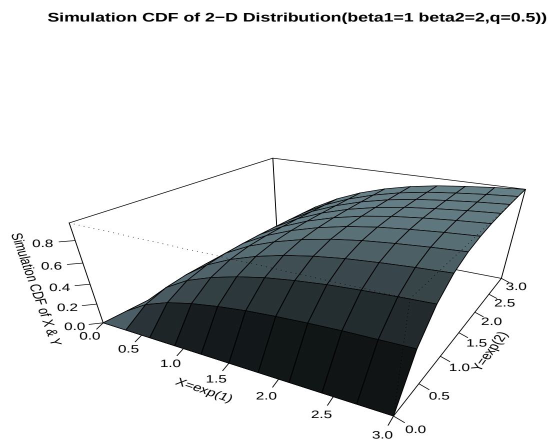
Y=-(log(1-V))/b

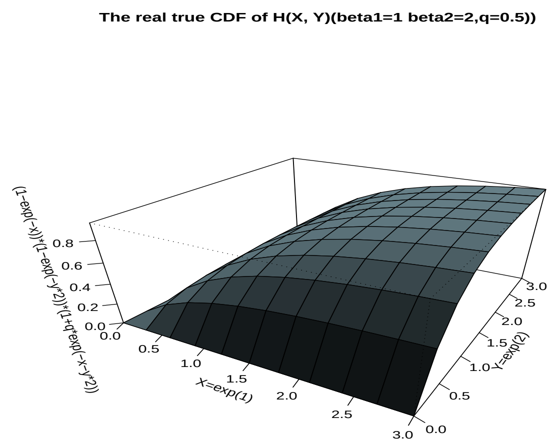
d) Calculation of estimators of $H(x, y)$:

The goal is to create an array of estimated values of $H(x, y)$ for each of our four cases. Since $a = 1, b = 2$, an upper bound of $x, y = 3$ is adequate; and for k=1,2,3,4 the array will contain values of $\hat{H}_n^{(k)}(x, y)$ for $(x, y) = (0.5(i-1), 0.5(j-1)), 1 \leq i, j \leq 7$.

The arrays will be denoted by M_A, M_B, M_C, M_D respectively.

e) To illustrate that the estimators work well, the estimator obtained by Case A with n=1000 is compared below with the true value of H. The other three cases yield similar results.





Method 1: reverse probability estimator:

Case A. Calculate $M_A = M2 = \hat{H}_n^{(1)}$: ($M_A$ is denoted by $M2$ in the R-code below.) The value of row i and column j of M_A is $\hat{H}_n^{(1)}((i-1)*0.5; (j-1)*0.5)$, where $1 \leq i, j \leq 7$. For example, the value of row 5 and column 6 of M_A is
$\hat{H}_n^{(1)}(0.5*(5-1); 0.5*(6-1)) = \hat{H}_n^{(1)}(2.0, 2.5)$.
We calculate $\hat{H}_n^{(1)}$ according to equation (2.0.5):

R-codes:

```
q=-1;  # given  q:relation between U and V
q7=q;
n=50;  # sample size
U <- runif(n);  # generate n-dimension of vector  U ~ U(0-1)
V=rep(0,n);     # generate  n-dimension of vector  V=0

# generate #n of X~exp(1) and Y~exp(2) by U and V

for (i in 1:n){V[i]=RejectionSampling(U[i],q,1)}
a=1
b=2
X=-(log(1-U))/a
Y=-(log(1-V))/b

par(mfrow=c(1,2))
plot(X,Y,type="p",main="Simulation Plot of  X(exp(1) and Y(exp(2)(q=0.5))")
ca=0.5;
C=rexp(n,ca);# generate n-dimension of vector  C ~ exp(0.5)
plot(C,C,type="p",col = "blue",main="Simulation Plot of  C(exp(0.5))")



# computing  CDF of simulations using M1
m=7

M1=matrix(0.0,m,m)
M2=matrix(0.0,m,m)
M3=matrix(0.0,m,m)
M4=matrix(0.0,m,m)
```

```
M5=matrix(0.0,m,m)
M6=matrix(0.0,m,m)
M7=matrix(0.0,m,m)
  for (i in 1:m){      #a3
     for (j in 1:m){   #a4
x11=5*(i-1)/10;
y11=5*(j-1)/10;
M6[i,j]=(1-exp(-x11*a))*(1-exp(-y11*b))*(1+q*exp(-x11*a-y11*b));
          for (k in 1:n){
if (X[k]<=(5*(i-1)/10) && Y[k]<=(5*(j-1)/10))
 {M1[i,j]=M1[i,j]+1}

                        }
 for (k in 1:n){
if (X[k]<=5*(i-1)/10 && Y[k]<=5*(j-1)/10){
if (X[k]<=C[k] && Y[k]<=C[k]) {
hh=max(X[k],Y[k])
M2[i,j]=M2[i,j]+1/exp(-0.5*hh)}
                                         }
               }

                    }   #a3
               }   #a4
M1=M1/n;
M2=M2/n;
```

Case B. Calculate $M_B = M3 = \hat{H}_n^{(2)}$: ($M_B$ is denoted by $M3$ in the R-code below.) The value of row i and column j of M_B is $\hat{H}_n^{(2)}((i-1)*0.5; (j-1)*0.5)$, where $1 \leq i, j \leq 7$. For example, the value of row 5 and column 6 of M_B is
$\hat{H}_n^{(2)}(0.5*(5-1); 0.5*(6-1)) = \hat{H}_n^{(2)}(2.0, 2.5)$.
We calculate $\hat{H}_n^{(2)}$ according to equation (2.0.6):

R-codes:

```
C1=rep(0,n);   # generate  n-dimension of vector  C1=0

# computing  #n of weights C1
for (i in 1:n){
if  (X[i]<C[i] & Y[i]<C[i]){
for (j in 1:n){
if (X[i]<C[j] & Y[i]<C[j]){ C1[i]=C1[i]+1}
             }
                             }
             }
  mean(C1)
 C1
# computing  CDF of simulations using M1
m=7

M3=matrix(0.0,m,m)
```

```
  for (i in 1:m){
     for (j in 1:m){
          for (k in 1:n){
if (X[k]<=5*(i-1)/10 && Y[k]<=5*(j-1)/10){
if (X[k]<=C[k] && Y[k]<=C[k]) {M3[i,j]=M3[i,j]+1/C1[k]}
                                     }
                         }


                    }
               }
```

Case C. Calculate $M_C = M4 = \hat{H}_n^{(3)}$: ($M_C$ is denoted by $M4$ in the R-code below.) The value of row i and column j of M_C is $\hat{H}_n^{(3)}((i-1)*0.5; (j-1)*0.5)$, where $1 \le i, j \le 7$. For example, the value of row 5 and column 6 of M_C is
$\hat{H}_n^{(3)}(0.5*(5-1); 0.5*(6-1)) = \hat{H}_n^{(3)}(2.0, 2.5)$.
We calculate $\hat{H}_n^{(3)}$ according to equation (2.0.7):
R-codes:

```
w=rexp(n,1)
for (i in 1:n){w[i]=max(X[i],Y[i])}

z=rep(0,n);
for (i in 1:n){z[i]=min(w[i],C[i])}
```

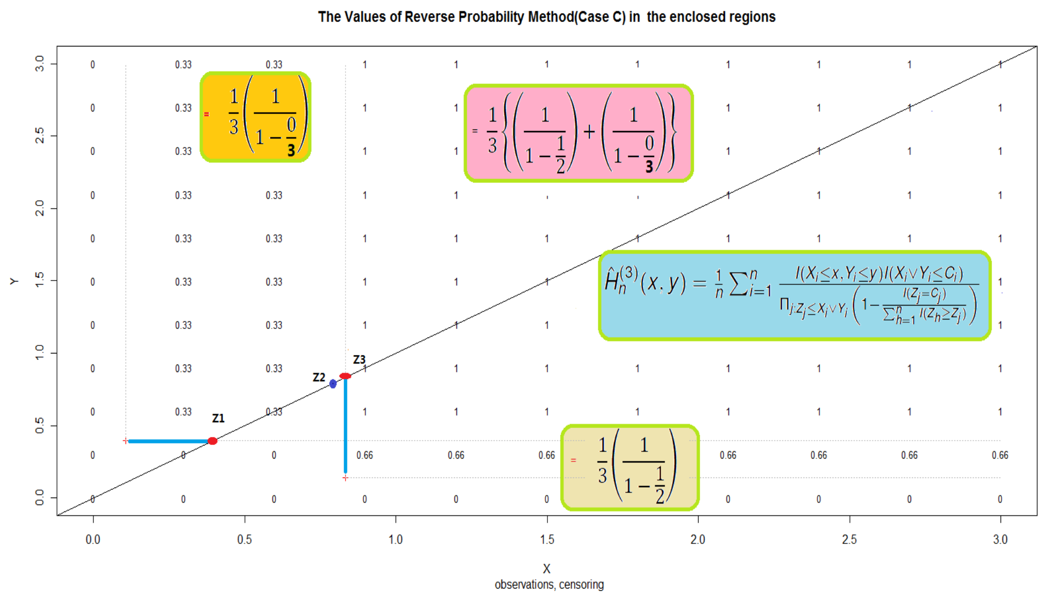
```
t=w;
Sc=rep(1,n);
nc=rep(0,n);
for (j in 1:n){
for (i in 1:n)   {if (z[i]>=C[j]){nc[j]=nc[j]+1}
                 }
              }
for (j in 1:n)
{
for (i in 1:n){if (z[i]<=t[j] && C[i]<=w[i]){Sc[j]=Sc[j]*(1-1/nc[i]);}
              }
}


 for (i in 1:m){
     for (j in 1:m){
          for (k in 1:n){
if (X[k]<=5*(i-1)/10 && Y[k]<=5*(j-1)/10){
if (X[k]<=C[k] && Y[k]<=C[k]) {M4[i,j]=M4[i,j]+1/(n*Sc[k]);}
                                         }
                         }
                    }


               }
```

To demonstrate how the estimator is calculated, we simulate three data points, one of which has been censored to obtain: $(X_1, Y_1) = (0.1080016, 0.3964248)$, $C_2 =$

(0.7969372), $(X_3, Y_3) = (0.8324562, 0.1400063)$. (X_1, Y_1) and (X_3, Y_3) are indicated by crosses and their projections Z_1 and Z_3 by dots, $C_2 = Z_2$ is also indicated by a dot. The estimated values of H are illustrated on the graph below:



Method 2: Adapted estimator:

Case D. Calculate $M_D = M5 = \hat{H}_n^{(4)}$: ($M_D$ is denoted by $M5$ in the R-code below.) The value of row i and column j of M_D is $\hat{H}_n^{(4)}((i-1)*0.5; (j-1)*0.5)$, where $1 \le i, j \le 7$. For example, the value of row 5 and column 6 of M_D is
$\hat{H}_n^{(4)}(0.5*(5-1); 0.5*(6-1)) = \hat{H}_n^{(4)}(2.0, 2.5)$.
We calculate $\hat{H}_n^{(4)}$ according to equation (2.0.10):

R-codes:

```
 #"Projections of n observations,one censoring"
par(mfrow=c(1,1))
```

```
 m6=7;
n8=n;
 X1 <- runif(n);  # generate n-dimension of vector  X ~ U(0-1)
 X2 <- runif(n);  # generate n-dimension of vector  Y ~ U(0-1)
X1=X;
X2=Y;
h5=rep(0,n);
 for (i in 1:n){h5[i]=max(X1[i],X2[i]);}

 Xa1=rep(0,n);   # generate  n-dimension of vector  0
 Xa2=rep(0,n);   # generate  n-dimension of vector  0
S=matrix(1,m6,m6)
 F=matrix(0,m6+1,m6+1)
U=matrix(0,2,n)

 X=rep(0,m6);
 Y=rep(0,m6);
for (i in 1:m6){
  X[i]=5*(i-1)/10;
 Y[i]=5*(i-1)/10;
}
 for (i in 1:m6){    # c1
    for (j in 1:m6){    # c2
        Xa1=rep(0,n);   # generate  n-dimension of vector  0
        Xa2=rep(0,n);   # generate  n-dimension of vector  0
 # "when the v is  over the line y=x"
      if (X[i]<=Y[j]) {         # c3
```

```
for (k in 1:n)  {      # c4


  # "when the observations are  over the line y=x"
  if (X1[k]<=X2[k] ){      # c5
    if(X1[k]<=X[i] && X2[k]<=Y[j])
        {                            # c6
         {if (X2[k]<=X[i])             # c7
                {Xa1[k]=X2[k];Xa2[k]=X2[k]}
                else{Xa1[k]=X[i];Xa2[k]=X2[k]}
         }                               # c7
        }                                 # c6
# "when the observations are  under the line y=x"
                  }else                   # c5
       { if(X1[k]<=X[i] && X2[k]<=Y[j])      # c8
          {Xa1[k]=X1[k];Xa2[k]=X1[k]}
       }                                  # c8


                }                         # c4
   ##"Projections of n observations and C"
     for (l in 1:n)  {                 # c9
      if (h5[l]<=C[l]){    #9a
        if (Xa1[l]+Xa2[l]>0){          # c10
            ac=0;
             for (v in 1:n8){if(h5[v]>C[v]){
             if (C[v]<=Y[j] && C[v]<=Xa2[l]){ac=ac+1;}
                                            }
                            }
for (m in 1:n)  { if(Xa1[m]+Xa2[m]>0 && h5[m]<=C[m])  # c11
```

```
            {if (Xa1[m]<=Xa1[l] && Xa2[m]<=Xa2[l])     # c12
                       {ac=ac+1;}
             }                                         # c12


                                 }                     # c11
    S[i,j]=(1-1/(n-ac+1))*S[i,j]; # computing the probabilities.
                       }                                # c10



                        }                     #9a


                     }                        # c9


                  }                           # c3
  # "when the v is under the line y=x"
     else      {                       # c13
      for (k in 1:n)  {                       # c14
        if (X1[k]<=X2[k]){                    # c15
          if(X1[k]<=X[i] && X2[k]<=Y[j])
           {Xa1[k]=X2[k];Xa2[k]=X2[k]; }
                      }else                   # c15
             { if(X1[k]<=X[i] && X2[k]<=Y[j])    # c16
                {if (X1[k]<=Y[j])                  # c17
                     {Xa1[k]=X1[k];Xa2[k]=X1[k]}
                      else{Xa1[k]=X1[k];Xa2[k]=Y[j]}
               }                                     # c17


             }                                   # c16
```

```
                        }                       # c14
            for (l in 1:n)  {                   # c18
                if (h5[l]<=C[l]){    #9a
                if (Xa1[l]+Xa2[l]>0){           # c19
                    ac=0;
                       for (v in 1:n8)  {if(h5[v]>C[v]){

                      if (C[v]<=X[i] && C[v]<=Xa1[l]){ac=ac+1;}
                                      }
                                                  }
    for (m in 1:n)  { if(Xa1[m]+Xa2[m]>0 && h5[m]<=C[m])  # c20
                  {if (Xa1[m]<=Xa1[l] && Xa2[m]<=Xa2[l])   # c21
                            {ac=ac+1;}
                   }                                       # c21
                               }                       # c20
             S[i,j]=(1-1/(n-ac+1))*S[i,j];


                                   }           # c19
                             }    #9a
                            }                  # c18
                          }    # c13
                   }     #c2
                 }    #c1
 F1=1-S
 F2=F1
F3=F1
F2=t(F1)
for (p in 1:m6){
```

```
    for (q in 1:m6){
   F[m6-p+1,q+1]=F2[p,q];
}
}
for (p in 1:m6){
F[m6+1-p,1]=Y[p];
F[m6+1,p+1]=X[p];


}
X3=X1;
X4=X2;
C3=C;
for (i in 1:n){
if (h5[i] <=C[i]){C3[i]=100;}
else{X3[i]=100;X4[i]=100}
              }
F[m6+1,p+1]=X[p];




plot(X3, X4,type='p',cex =0.8, xlim=range(0,3), ylim=range(0, 3),
col="red",sub="  observations, censoring",
main="Values of the path-dependent estimators in each of
the enclosed regions",pch=3, xlab='X',  ylab='Y')

abline(0,1)
 for (v in 1:n){
 points(C3[v], C3[v], type='p', col="blue", xlab='x', ylab='y')
```

```
                }
for (p in 1:m6){
    for (q in 1:m6){
F1[p,q]=as.integer((F1[p,q]+0.0001)*100)/100;
text(X[p], Y[q], F1[p,q],cex =.8)
}
}
for (p in 1:n){
segments(X3[p], X4[p],X3[p],3,col = "grey", lty =3);
segments(X3[p], X4[p],3,X4[p],col = "grey", lty =3);
}
n  # The number of observations
F   # The matrix of  values of the path-dependent estimators
q=q7;

M5=F3
```

Using the same three data points as in Case C, we illustrate the paths and related projections of (X_1, Y_1), C_2 and (X_3, Y_3) to various points (x, y) on the plane, with the corresponding estimators(a, b, and c).

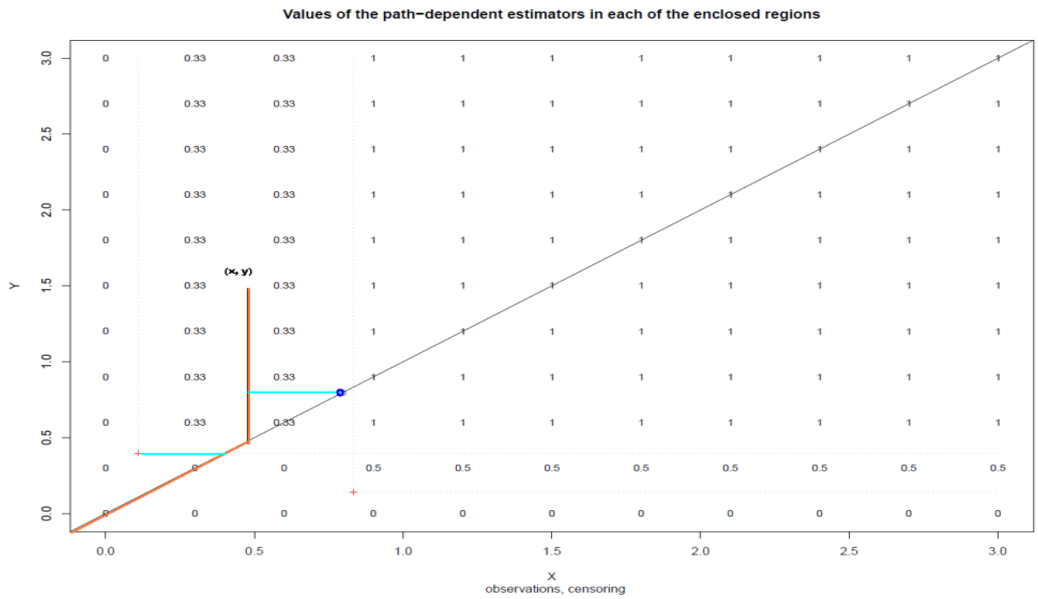
Adapted path-dependent estimator formula(2.0.10):

$$\hat{H}_n^{(4)}(x, y) = 1 - \Pi_{i:(X_i \leq x \wedge C_i, Y_i \leq y \wedge C_i)} \left(1 - \frac{1}{R_i}\right) \tag{3.0.1}$$

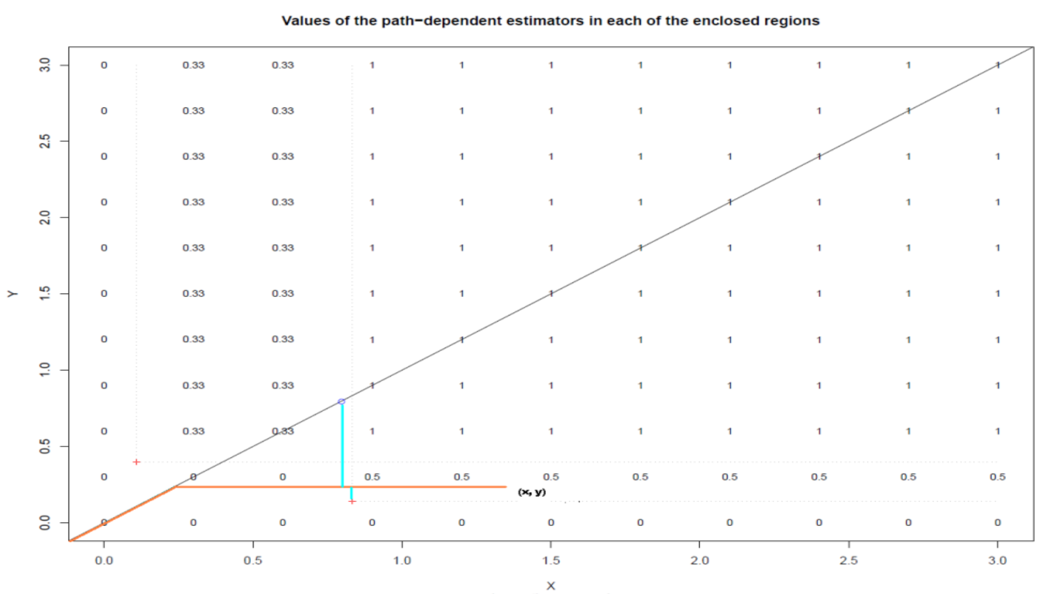
where

$R_i = n + 1 - \Sigma_{j=1}^{n} I(C_j \leq (V_i \wedge V_j)) - \Sigma_{j=1}^{n} I(X_j \leq (V_i \wedge C_j \wedge x, Y_j \leq V_i \wedge C_j \wedge y)$.

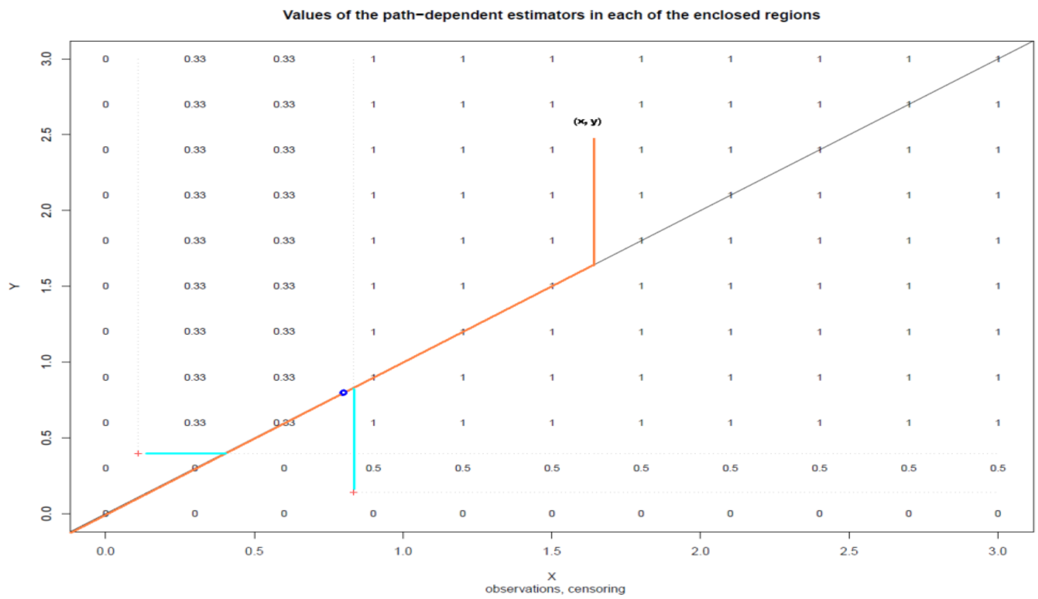
a.) Adapted path-dependent estimator $\hat{H}_n^{(4)}(x,y) = 1 - (1 - \frac{1}{3+1-1}) = 0.33.$



b.) Adapted path-dependent estimator $\hat{H}_n^{(4)}(x,y) = 1 - (1 - \frac{1}{3+1-2}) = 0.5.$



c.) Adapted path-dependent estimator $\hat{H}_n^{(4)}(x, y) = 1-(1-\frac{1}{3+1-1})(1-\frac{1}{3+1-3}) = 1.$



e.) Comparison of the Estimators

The preceding steps were repeated 1000 times and the average and the mean square error were calculated for each of the estimators.

As noted before, the marginal distributions are exp(1) and exp(2) and we consider five values of the FGM copula: q=1, 0.5, 0, -0.5, -1. In each case, we display the following tables in the Appendix A: The first gives the real value of $H(x, y)$; the next gives average values of $\hat{H}_n^{(i)}(x, y)$ and the final gives the MSE of the estimators, so on every page, we have 3 tables. These are Tables 1 — 60. Bias of the estimators is illustrated in Tables 61—64 and the MSE of the estimators is compared in Tables 65—70.

Chapter 4

Comparison of the Estimators

1. First of all, we know that the asymptotic variance in $\hat{H}_n^{(1)}(x,y)$ is larger than $\hat{H}_n^{(2)}(x,y)$ and $\hat{H}_n^{(3)}(x,y)$. Our simulations demonstrate that this is also true for MSE (which includes any bias), see Tables 65 and 66 of Appendix A. We were not able to do an analytic comparison between Var $\hat{H}_n^{(1)}(x,y)$ and Var $\hat{H}_n^{(4)}(x,y)$, but the simulations indicate that $\hat{H}_n^{(4)}(x,y)$ has a smaller MSE, see Table 67.

2. $\hat{H}_n^{(4)}(x,y)$ is biased downward (this is since the KM estimator is biased upward, see [1], pg. 257), and in fact it is shown in reference [3] that $\hat{H}_n^{(4)}(x,y)$ is always less than or equal to $\hat{H}_n^{(3)}(x,y)$. Tables 62 and 63 indicate that $\hat{H}_n^{(2)}(x,y)$ and $\hat{H}_n^{(3)}(x,y)$ don't seem to have any particular bias. The bias in $\hat{H}_n^{(4)}(x,y)$ is most noticeable for small values of x and y, and decreases for larger values. See Table 64.

3. Although there are exceptions, from Tables 68—70 it seems that most of the time MSE $\hat{H}_n^{(4)}(x,y) <$ MSE $\hat{H}_n^{(3)}(x,y) <$ MSE $\hat{H}_n^{(2)}(x,y)$. So even with the bias, $\hat{H}_n^{(4)}(x,y)$ seems to perform quite well regardless of whether the correlation between X and Y is positive or negative.

Therefore, we draw the following conclusions:

1. $\hat{H}_n^{(1)}(x,y)$ requires the most information (G must be known) and has the largest MSE. Therefore, this method is not recommended.

2. $\hat{H}_n^{(2)}(x,y)$ requires all of the censoring values and $\hat{H}_n^{(2)}(x,y)$ seems to have a larger MSE than $\hat{H}_n^{(3)}(x,y)$ and $\hat{H}_n^{(4)}(x,y)$. If bias is important, $\hat{H}_n^{(2)}(x,y)$ might be preferred, provided that all of the censoring values are available. However, if we focus on MSE, then we should use $\hat{H}_n^{(3)}(x,y)$ or $\hat{H}_n^{(4)}(x,y)$.

3. $\hat{H}_n^{(3)}(x,y)$ and $\hat{H}_n^{(4)}(x,y)$ require only the censoring values when (X,Y) is censored. $\hat{H}_n^{(3)}(x,y)$ is not adapted, but has a smaller bias than $\hat{H}_n^{(4)}(x,y)$. On the other hand, the MSE of the two estimators are very similar and $\hat{H}_n^{(4)}(x,y)$ is adapted, and so uses the smaller amount of information.

Chapter 5

Conclusion

The major contribution of this thesis is a systematic comparison of four estimators of a bivariate distribution under univariate censoring. In particular, we have

- compared the asymptotic variances of the estimators analytically (where possible) and empirically,
- produced the R code to calculate each of the four estimators,
- developed an algorithm and R code to simulate data from a bivariate distribution given the copula and marginals.

Our study leads us to a somewhat surprising conclusion: we should use the worst estimate of G in order to obtain the best estimate of H. Even if G or the empirical distribution $\hat{G}_n$ is known, neither one should be used. Regardless of the amount of information available, we recommend using $\hat{H}_n^{(3)}(x, y)$ or $\hat{H}_n^{(4)}(x, y)$. $\hat{H}_n^{(4)}(x, y)$ should be used if an adapted estimator is required. $\hat{H}_n^{(3)}(x, y)$ should be used if bias near the axes is important.

Appendix A

Tables

Simulations were carried out for $(X, Y) \sim H$ where $X \sim \exp(1)$, $Y \sim \exp(2)$, C_H is an FGM copula with q=-1, -0.5, 0, 0.5 or 1. The censoring variable satisfies $C \sim \exp(0.5)$. Tables 1-60 give the average values and MSE of all four estimators based on 1000 samples of size 50. Bias is summarized in Tables 61-64 and MSE of the estimators is compared in Tables 65-70.

(x,y)	0.5	1.0	1.5	2.0	2.5	3.0
0.5	0.304217	0.368146	0.385170	0.390554	0.392415	0.393084
1.0	0.453653	0.573785	0.611650	0.624724	0.629418	0.631129
1.5	0.531385	0.692016	0.746392	0.765758	0.772795	0.775373
2.0	0.573785	0.761339	0.827152	0.850932	0.859622	0.862811
2.5	0.597754	0.802506	0.875779	0.902458	0.912234	0.915826
3.0	0.611650	0.827152	0.905143	0.933660	0.944127	0.947975

Table A.1: Table for real values of $H(x,y)$ when q=1.0, a=1, b=2, c=0.5,n=50, p=1000

(x,y)	0.5	1.0	1.5	2.0	2.5	3.0
0.5	0.304206	0.368263	0.385729	0.391238	0.393431	0.394112
1.0	0.450900	0.572054	0.610682	0.624101	0.629421	0.631121
1.5	0.527143	0.691702	0.746018	0.765472	0.773687	0.776152
2.0	0.570428	0.761889	0.827051	0.850309	0.859976	0.863286
2.5	0.593552	0.802437	0.875565	0.901434	0.912315	0.915922
3.0	0.606752	0.826496	0.904578	0.932715	0.944392	0.948412

Table A.2: Table for Case A: average values of $\hat{H}_n^{(1)}(x,y)$ q=1.0, a=1, b=2, c=0.5,n=50, p=1000

(x,y)	0.5	1.0	1.5	2.0	2.5	3.0
0.5	0.005301	0.006418	0.006820	0.006903	0.006939	0.006985
1.0	0.007264	0.008296	0.008850	0.009071	0.009287	0.009430
1.5	0.008370	0.009454	0.010244	0.010474	0.011026	0.011279
2.0	0.009530	0.010756	0.011594	0.011631	0.012116	0.012366
2.5	0.010480	0.012162	0.012960	0.013137	0.013483	0.013578
3.0	0.010887	0.013192	0.013915	0.014500	0.014914	0.014998

Table A.3: Table for Case A:MSE q=1.0, a=1, b=2, c=0.5,n=50, p=1000

(x,y)	0.5	1.0	1.5	2.0	2.5	3.0
0.5	0.304217	0.368146	0.385170	0.390554	0.392415	0.393084
1.0	0.453653	0.573785	0.611650	0.624724	0.629418	0.631129
1.5	0.531385	0.692016	0.746392	0.765758	0.772795	0.775373
2.0	0.573785	0.761339	0.827152	0.850932	0.859622	0.862811
2.5	0.597754	0.802506	0.875779	0.902458	0.912234	0.915826
3.0	0.611650	0.827152	0.905143	0.933660	0.944127	0.947975

Table A.4: Table for real values of $H(x,y)$ when q=1.0, a=1, b=2, c=0.5,n=50, p=1000

(x,y)	0.5	1.0	1.5	2.0	2.5	3.0
0.5	0.304229	0.368360	0.385555	0.391124	0.393445	0.394057
1.0	0.451181	0.572618	0.610702	0.624059	0.629380	0.630955
1.5	0.527305	0.692390	0.746118	0.765552	0.773709	0.776118
2.0	0.570261	0.762232	0.826736	0.849873	0.859430	0.862757
2.5	0.593235	0.803147	0.875938	0.901653	0.912530	0.916222
3.0	0.606537	0.827253	0.905273	0.933096	0.944791	0.949015

Table A.5: Table for Case B: average values of $\hat{H}_n^{(2)}(x,y)$ q=1.0, a=1, b=2, c=0.5,n=50, p=1000

(x,y)	0.5	1.0	1.5	2.0	2.5	3.0
0.5	0.005130	0.006200	0.006478	0.006612	0.006657	0.006696
1.0	0.006747	0.007247	0.007402	0.007523	0.007643	0.007734
1.5	0.007394	0.007455	0.007299	0.007348	0.007701	0.007905
2.0	0.008100	0.007727	0.007168	0.006785	0.007005	0.007142
2.5	0.008775	0.008637	0.007936	0.007448	0.007496	0.007511
3.0	0.009251	0.009539	0.008775	0.008425	0.008475	0.008565

Table A.6: Table for Case B:MSE q=1.0, a=1, b=2, c=0.5,n=50, p=1000

(x, y)	0.5	1.0	1.5	2.0	2.5	3.0
0.5	0.304217	0.368146	0.385170	0.390554	0.392415	0.393084
1.0	0.453653	0.573785	0.611650	0.624724	0.629418	0.631129
1.5	0.531385	0.692016	0.746392	0.765758	0.772795	0.775373
2.0	0.573785	0.761339	0.827152	0.850932	0.859622	0.862811
2.5	0.597754	0.802506	0.875779	0.902458	0.912234	0.915826
3.0	0.611650	0.827152	0.905143	0.933660	0.944127	0.947975

Table A.7: Table for real values of $H(x, y)$ when q=1.0, a=1, b=2, c=0.5,n=50, p=1000

(x, y)	0.5	1.0	1.5	2.0	2.5	3.0
0.5	0.304435	0.368761	0.386013	0.391582	0.393823	0.394308
1.0	0.451372	0.572922	0.611102	0.624425	0.629395	0.630760
1.5	0.527785	0.693000	0.746825	0.766369	0.773967	0.776224
2.0	0.571419	0.762932	0.827731	0.850971	0.860134	0.863327
2.5	0.593791	0.802478	0.875174	0.900918	0.911338	0.914975
3.0	0.604947	0.823226	0.900603	0.928124	0.939143	0.943060

Table A.8: Table for Case C: average values of $\hat{H}_n^{(3)}(x, y)$ q=1.0, a=1, b=2, c=0.5,n=50, p=1000

(x, y)	0.5	1.0	1.5	2.0	2.5	3.0
0.5	0.005148	0.006239	0.006504	0.006644	0.006730	0.006751
1.0	0.006727	0.007123	0.007194	0.007335	0.007443	0.007474
1.5	0.007276	0.007039	0.006535	0.006528	0.006733	0.006884
2.0	0.008170	0.006948	0.005823	0.005240	0.005299	0.005368
2.5	0.008751	0.007216	0.005577	0.004604	0.004353	0.004302
3.0	0.008679	0.007013	0.004971	0.004025	0.003596	0.003430

Table A.9: Table for Case C:MSE q=1.0, a=1, b=2, c=0.5,n=50, p=1000

(x, y)	0.5	1.0	1.5	2.0	2.5	3.0
0.5	0.304217	0.368146	0.385170	0.390554	0.392415	0.393084
1.0	0.453653	0.573785	0.611650	0.624724	0.629418	0.631129
1.5	0.531385	0.692016	0.746392	0.765758	0.772795	0.775373
2.0	0.573785	0.761339	0.827152	0.850932	0.859622	0.862811
2.5	0.597754	0.802506	0.875779	0.902458	0.912234	0.915826
3.0	0.611650	0.827152	0.905143	0.933660	0.944127	0.947975

Table A.10: Table for real values of $H(x, y)$ when q=1.0, a=1, b=2, c=0.5,n=50, p=1000

(x, y)	0.5	1.0	1.5	2.0	2.5	3.0
0.5	0.304435	0.368159	0.383980	0.388320	0.389711	0.390046
1.0	0.450274	0.572922	0.610728	0.622775	0.626731	0.627717
1.5	0.521757	0.692255	0.746825	0.766172	0.773289	0.775194
2.0	0.557628	0.758847	0.827428	0.850971	0.860081	0.863155
2.5	0.573251	0.793320	0.873432	0.900871	0.911338	0.914945
3.0	0.580456	0.810166	0.897121	0.927796	0.939143	0.943060

Table A.11: Table for Case D: average values of $\hat{H}_n^{(4)}(x, y)$ q=1.0, a=1, b=2, c=0.5,n=50, p=1000

(x, y)	0.5	1.0	1.5	2.0	2.5	3.0
0.5	0.005148	0.006204	0.006420	0.006484	0.006494	0.006505
1.0	0.006681	0.007123	0.007191	0.007247	0.007293	0.007314
1.5	0.007128	0.006988	0.006535	0.006510	0.006706	0.006841
2.0	0.007718	0.006812	0.005825	0.005240	0.005296	0.005366
2.5	0.008122	0.007068	0.005594	0.004609	0.004353	0.004304
3.0	0.008389	0.007111	0.005091	0.004038	0.003596	0.003430

Table A.12: Table for Case D:MSE q=1.0, a=1, b=2, c=0.5,n=50, p=1000

(x,y)	0.5	1.0	1.5	2.0	2.5	3.0
0.5	0.276469	0.354182	0.379525	0.388408	0.391617	0.392789
1.0	0.426615	0.560178	0.606150	0.622633	0.628640	0.630841
1.5	0.511230	0.681874	0.742292	0.764199	0.772215	0.775158
2.0	0.560178	0.754492	0.824384	0.849880	0.859230	0.862666
2.5	0.588994	0.798097	0.873997	0.901780	0.911982	0.915733
3.0	0.606150	0.824384	0.904024	0.933234	0.943969	0.947916

Table A.13: Table for real values of $H(x,y)$ when q=0.5, a=1, b=2, c=0.5,n=50, p=1000

(x,y)	0.5	1.0	1.5	2.0	2.5	3.0
0.5	0.274250	0.353685	0.379354	0.387169	0.390722	0.392316
1.0	0.422852	0.559544	0.605409	0.620286	0.626560	0.628929
1.5	0.505355	0.679690	0.740709	0.760203	0.768947	0.771833
2.0	0.553837	0.751051	0.821835	0.844810	0.854661	0.857949
2.5	0.582531	0.796444	0.872924	0.897943	0.908606	0.912053
3.0	0.599784	0.822074	0.903585	0.929947	0.941308	0.944830

Table A.14: Table for Case A: average values of $\hat{H}_n^{(1)}(x,y)$ q=0.5, a=1, b=2, c=0.5,n=50, p=1000

(x,y)	0.5	1.0	1.5	2.0	2.5	3.0
0.5	0.004631	0.006011	0.006577	0.006693	0.006895	0.006994
1.0	0.006858	0.008464	0.008856	0.008901	0.009207	0.009295
1.5	0.008820	0.010344	0.010649	0.010665	0.010986	0.011212
2.0	0.009853	0.011539	0.011812	0.011738	0.012075	0.012299
2.5	0.010997	0.012867	0.013067	0.013124	0.013418	0.013633
3.0	0.012265	0.014249	0.014896	0.015140	0.015434	0.015675

Table A.15: Table for Case A:MSE q=0.5, a=1, b=2, c=0.5,n=50, p=1000

(x, y)	0.5	1.0	1.5	2.0	2.5	3.0
0.5	0.276469	0.354182	0.379525	0.388408	0.391617	0.392789
1.0	0.426615	0.560178	0.606150	0.622633	0.628640	0.630841
1.5	0.511230	0.681874	0.742292	0.764199	0.772215	0.775158
2.0	0.560178	0.754492	0.824384	0.849880	0.859230	0.862666
2.5	0.588994	0.798097	0.873997	0.901780	0.911982	0.915733
3.0	0.606150	0.824384	0.904024	0.933234	0.943969	0.947916

Table A.16: Table for real values of $H(x, y)$ when q=0.5, a=1, b=2, c=0.5,n=50, p=1000

(x, y)	0.5	1.0	1.5	2.0	2.5	3.0
0.5	0.274121	0.353744	0.379553	0.387273	0.390693	0.392119
1.0	0.423459	0.560767	0.606690	0.621320	0.627431	0.629647
1.5	0.506068	0.681192	0.742185	0.761357	0.769950	0.772680
2.0	0.554088	0.752142	0.823002	0.845716	0.855381	0.858456
2.5	0.582673	0.797028	0.873722	0.898343	0.908852	0.912044
3.0	0.599499	0.822326	0.903515	0.929413	0.940579	0.943827

Table A.17: Table for Case B: average values of $\hat{H}_n^{(2)}(x, y)$ q=0.5, a=1, b=2, c=0.5,n=50, p=1000

(x, y)	0.5	1.0	1.5	2.0	2.5	3.0
0.5	0.004339	0.005464	0.006034	0.006141	0.006279	0.006345
1.0	0.006172	0.007061	0.007126	0.006996	0.007173	0.007133
1.5	0.007749	0.008008	0.007716	0.007398	0.007580	0.007637
2.0	0.008323	0.008242	0.007574	0.007100	0.007308	0.007307
2.5	0.009072	0.008778	0.007911	0.007402	0.007514	0.007477
3.0	0.010119	0.009658	0.008758	0.008244	0.008208	0.008192

Table A.18: Table for Case B:MSE q=0.5, a=1, b=2, c=0.5,n=50, p=1000

(x,y)	0.5	1.0	1.5	2.0	2.5	3.0
0.5	0.276469	0.354182	0.379525	0.388408	0.391617	0.392789
1.0	0.426615	0.560178	0.606150	0.622633	0.628640	0.630841
1.5	0.511230	0.681874	0.742292	0.764199	0.772215	0.775158
2.0	0.560178	0.754492	0.824384	0.849880	0.859230	0.862666
2.5	0.588994	0.798097	0.873997	0.901780	0.911982	0.915733
3.0	0.606150	0.824384	0.904024	0.933234	0.943969	0.947916

Table A.19: Table for real values of $H(x,y)$ when q=0.5, a=1, b=2, c=0.5,n=50, p=1000

(x,y)	0.5	1.0	1.5	2.0	2.5	3.0
0.5	0.274211	0.353774	0.379553	0.387378	0.390746	0.392148
1.0	0.423503	0.560654	0.606682	0.621469	0.627593	0.629763
1.5	0.506200	0.681086	0.742151	0.761491	0.769827	0.772428
2.0	0.555072	0.753036	0.823954	0.846893	0.856340	0.859183
2.5	0.583720	0.798000	0.874190	0.898828	0.909179	0.912128
3.0	0.599337	0.820864	0.901187	0.926728	0.937689	0.940686

Table A.20: Table for Case C: average values of $\hat{H}_n^{(3)}(x,y)$ q=0.5, a=1, b=2, c=0.5,n=50, p=1000

(x,y)	0.5	1.0	1.5	2.0	2.5	3.0
0.5	0.004335	0.005424	0.005980	0.006137	0.006292	0.006384
1.0	0.006111	0.006867	0.006890	0.006800	0.006978	0.006925
1.5	0.007521	0.007360	0.006840	0.006410	0.006499	0.006517
2.0	0.007947	0.007271	0.006003	0.005244	0.005237	0.005201
2.5	0.008652	0.007410	0.005381	0.004361	0.004079	0.003955
3.0	0.009328	0.007425	0.005097	0.003981	0.003459	0.003322

Table A.21: Table for Case C:MSE q=0.5, a=1, b=2, c=0.5,n=50, p=1000

(x,y)	0.5	1.0	1.5	2.0	2.5	3.0
0.5	0.276469	0.354182	0.379525	0.388408	0.391617	0.392789
1.0	0.426615	0.560178	0.606150	0.622633	0.628640	0.630841
1.5	0.511230	0.681874	0.742292	0.764199	0.772215	0.775158
2.0	0.560178	0.754492	0.824384	0.849880	0.859230	0.862666
2.5	0.588994	0.798097	0.873997	0.901780	0.911982	0.915733
3.0	0.606150	0.824384	0.904024	0.933234	0.943969	0.947916

Table A.22: Table for real values of $H(x,y)$ when q=0.5, a=1, b=2, c=0.5,n=50, p=1000

(x,y)	0.5	1.0	1.5	2.0	2.5	3.0
0.5	0.274211	0.353034	0.376704	0.382942	0.385168	0.385966
1.0	0.422289	0.560654	0.606226	0.619838	0.624643	0.626059
1.5	0.499420	0.680213	0.742151	0.761234	0.768984	0.771090
2.0	0.539048	0.748232	0.823592	0.846893	0.856269	0.858954
2.5	0.558882	0.787235	0.872493	0.898772	0.909179	0.912128
3.0	0.568285	0.805311	0.898152	0.926597	0.937689	0.940686

Table A.23: Table for Case D: average values of $\hat{H}_n^{(4)}(x,y)$ q=0.5, a=1, b=2, c=0.5,n=50, p=1000

(x,y)	0.5	1.0	1.5	2.0	2.5	3.0
0.5	0.004335	0.005395	0.005814	0.005878	0.005929	0.005947
1.0	0.006068	0.006867	0.006866	0.006741	0.006813	0.006769
1.5	0.007378	0.007321	0.006840	0.006388	0.006448	0.006446
2.0	0.007792	0.007215	0.005999	0.005244	0.005235	0.005202
2.5	0.008443	0.007329	0.005434	0.004364	0.004079	0.003955
3.0	0.009255	0.007602	0.005244	0.003991	0.003459	0.003322

Table A.24: Table for Case D:MSE q=0.5, a=1, b=2, c=0.5,n=50, p=1000

(x, y)	0.5	1.0	1.5	2.0	2.5	3.0
0.5	0.248720	0.340219	0.373880	0.386263	0.390818	0.392494
1.0	0.399576	0.546572	0.600649	0.620543	0.627861	0.630554
1.5	0.491075	0.671732	0.738192	0.762641	0.771635	0.774944
2.0	0.546572	0.747645	0.821616	0.848828	0.858839	0.862521
2.5	0.580233	0.793689	0.872215	0.901103	0.911730	0.915640
3.0	0.600649	0.821616	0.902905	0.932809	0.943810	0.947858

Table A.25: Table for real values of $H(x, y)$ when q=0, a=1, b=2, c=0.5,n=50, p=1000

(x, y)	0.5	1.0	1.5	2.0	2.5	3.0
0.5	0.247170	0.339522	0.373561	0.386112	0.390280	0.391848
1.0	0.398332	0.546026	0.599564	0.619714	0.626147	0.628510
1.5	0.490970	0.671650	0.736901	0.761154	0.770012	0.772915
2.0	0.547882	0.748159	0.820950	0.848312	0.857820	0.860877
2.5	0.581814	0.794060	0.871469	0.900183	0.909874	0.912930
3.0	0.601583	0.820732	0.901074	0.931482	0.941793	0.945095

Table A.26: Table for Case A: average values of $\hat{H}_n^{(1)}(x, y)$ q=0, a=1, b=2, c=0.5,n=50, p=1000

(x, y)	0.5	1.0	1.5	2.0	2.5	3.0
0.5	0.004718	0.006634	0.007255	0.007694	0.007847	0.007980
1.0	0.006985	0.008810	0.009456	0.009896	0.010090	0.010205
1.5	0.008828	0.010157	0.010763	0.011304	0.011244	0.011290
2.0	0.009889	0.011467	0.011883	0.012412	0.012277	0.012343
2.5	0.011095	0.012536	0.012735	0.013369	0.013149	0.013222
3.0	0.012290	0.014031	0.014404	0.015062	0.014840	0.015088

Table A.27: Table for Case A:MSE q=0, a=1, b=2, c=0.5,n=50, p=1000

(x, y)	0.5	1.0	1.5	2.0	2.5	3.0
0.5	0.248720	0.340219	0.373880	0.386263	0.390818	0.392494
1.0	0.399576	0.546572	0.600649	0.620543	0.627861	0.630554
1.5	0.491075	0.671732	0.738192	0.762641	0.771635	0.774944
2.0	0.546572	0.747645	0.821616	0.848828	0.858839	0.862521
2.5	0.580233	0.793689	0.872215	0.901103	0.911730	0.915640
3.0	0.600649	0.821616	0.902905	0.932809	0.943810	0.947858

Table A.28: Table for real values of $H(x, y)$ when q=0, a=1, b=2, c=0.5,n=50, p=1000

(x, y)	0.5	1.0	1.5	2.0	2.5	3.0
0.5	0.247225	0.339523	0.373551	0.386073	0.390319	0.391747
1.0	0.398671	0.546466	0.599943	0.620048	0.626682	0.628964
1.5	0.491590	0.672415	0.737562	0.761774	0.770891	0.773685
2.0	0.548309	0.748698	0.821451	0.848741	0.858462	0.861416
2.5	0.582402	0.794393	0.871786	0.900379	0.910287	0.913240
3.0	0.602028	0.820958	0.901449	0.931711	0.942224	0.945383

Table A.29: Table for Case B: average values of $\hat{H}_n^{(2)}(x, y)$ q=0, a=1, b=2, c=0.5,n=50, p=1000

(x, y)	0.5	1.0	1.5	2.0	2.5	3.0
0.5	0.004555	0.006247	0.006777	0.007179	0.007307	0.007422
1.0	0.006500	0.007536	0.007752	0.007967	0.008136	0.008239
1.5	0.007962	0.008012	0.007729	0.007908	0.007871	0.007904
2.0	0.008604	0.008331	0.007477	0.007495	0.007410	0.007449
2.5	0.009690	0.008896	0.007520	0.007543	0.007386	0.007364
3.0	0.010422	0.009776	0.008502	0.008449	0.008246	0.008316

Table A.30: Table for Case B:MSE q=0, a=1, b=2, c=0.5,n=50, p=1000

(x, y)	0.5	1.0	1.5	2.0	2.5	3.0
0.5	0.248720	0.340219	0.373880	0.386263	0.390818	0.392494
1.0	0.399576	0.546572	0.600649	0.620543	0.627861	0.630554
1.5	0.491075	0.671732	0.738192	0.762641	0.771635	0.774944
2.0	0.546572	0.747645	0.821616	0.848828	0.858839	0.862521
2.5	0.580233	0.793689	0.872215	0.901103	0.911730	0.915640
3.0	0.600649	0.821616	0.902905	0.932809	0.943810	0.947858

Table A.31: Table for real values of $H(x, y)$ when q=0, a=1, b=2, c=0.5,n=50, p=1000

(x, y)	0.5	1.0	1.5	2.0	2.5	3.0
0.5	0.247079	0.339563	0.373555	0.385841	0.390043	0.391468
1.0	0.398598	0.546560	0.600074	0.619680	0.625992	0.628037
1.5	0.491236	0.672154	0.737318	0.761065	0.769631	0.772115
2.0	0.547757	0.748643	0.821123	0.847900	0.856994	0.859794
2.5	0.580399	0.792495	0.869416	0.897306	0.906620	0.909420
3.0	0.597487	0.815269	0.894743	0.924280	0.934157	0.937175

Table A.32: Table for Case C: average values of $\hat{H}_n^{(3)}(x, y)$ q=0, a=1, b=2, c=0.5,n=50, p=1000

(x, y)	0.5	1.0	1.5	2.0	2.5	3.0
0.5	0.004532	0.006268	0.006782	0.006987	0.007081	0.007207
1.0	0.006422	0.007501	0.007636	0.007560	0.007624	0.007679
1.5	0.007672	0.007611	0.007074	0.006854	0.006682	0.006662
2.0	0.008096	0.007429	0.006184	0.005533	0.005271	0.005295
2.5	0.008861	0.007285	0.005425	0.004639	0.004181	0.004078
3.0	0.009011	0.007057	0.005215	0.004277	0.003730	0.003677

Table A.33: Table for Case C:MSE q=0, a=1, b=2, c=0.5,n=50, p=1000

(x, y)	0.5	1.0	1.5	2.0	2.5	3.0
0.5	0.248720	0.340219	0.373880	0.386263	0.390818	0.392494
1.0	0.399576	0.546572	0.600649	0.620543	0.627861	0.630554
1.5	0.491075	0.671732	0.738192	0.762641	0.771635	0.774944
2.0	0.546572	0.747645	0.821616	0.848828	0.858839	0.862521
2.5	0.580233	0.793689	0.872215	0.901103	0.911730	0.915640
3.0	0.600649	0.821616	0.902905	0.932809	0.943810	0.947858

Table A.34: Table for real values of $H(x, y)$ when q=0, a=1, b=2, c=0.5,n=50, p=1000

(x, y)	0.5	1.0	1.5	2.0	2.5	3.0
0.5	0.247079	0.338625	0.369793	0.379572	0.382308	0.383100
1.0	0.397248	0.546560	0.599449	0.617417	0.622562	0.624031
1.5	0.483352	0.671123	0.737318	0.760838	0.768835	0.770964
2.0	0.529530	0.743107	0.820687	0.847900	0.856976	0.859738
2.5	0.552441	0.781072	0.867271	0.897185	0.906620	0.909393
3.0	0.563075	0.798943	0.891114	0.924035	0.934140	0.937175

Table A.35: Table for Case D: average values of $\hat{H}_n^{(4)}(x, y)$ q=0, a=1, b=2, c=0.5,n=50, p=1000

(x, y)	0.5	1.0	1.5	2.0	2.5	3.0
0.5	0.004532	0.006221	0.006606	0.006764	0.006801	0.006848
1.0	0.006371	0.007501	0.007619	0.007505	0.007549	0.007576
1.5	0.007439	0.007559	0.007074	0.006841	0.006677	0.006644
2.0	0.007678	0.007219	0.006173	0.005533	0.005271	0.005291
2.5	0.008472	0.007228	0.005437	0.004642	0.004181	0.004077
3.0	0.009224	0.007439	0.005352	0.004291	0.003730	0.003677

Table A.36: Table for Case D:MSE q=0, a=1, b=2, c=0.5,n=50, p=1000

(x,y)	0.5	1.0	1.5	2.0	2.5	3.0
0.5	0.220972	0.326256	0.368235	0.384117	0.390020	0.392199
1.0	0.372538	0.532966	0.595148	0.618452	0.627083	0.630266
1.5	0.470920	0.661590	0.734091	0.761083	0.771055	0.774730
2.0	0.532966	0.740798	0.818848	0.847776	0.858447	0.862377
2.5	0.571472	0.789280	0.870432	0.900425	0.911478	0.915547
3.0	0.595148	0.818848	0.901786	0.932384	0.943652	0.947799

Table A.37: Table for real values of $H(x,y)$ when q=-0.5, a=1, b=2, c=0.5,n=50, p=1000

(x,y)	0.5	1.0	1.5	2.0	2.5	3.0
0.5	0.217932	0.321642	0.363052	0.377664	0.383758	0.386212
1.0	0.370718	0.531360	0.594241	0.616681	0.625259	0.628315
1.5	0.470646	0.660251	0.734888	0.762051	0.772306	0.775592
2.0	0.532426	0.737902	0.818161	0.846930	0.857798	0.861460
2.5	0.572074	0.787587	0.871453	0.901199	0.912503	0.916400
3.0	0.597076	0.819751	0.905605	0.935909	0.947283	0.951330

Table A.38: Table for Case A: average values of $\hat{H}_n^{(1)}(x,y)$ q=-0.5, a=1, b=2, c=0.5,n=50, p=1000

(x,y)	0.5	1.0	1.5	2.0	2.5	3.0
0.5	0.003908	0.005931	0.006787	0.007247	0.007739	0.007860
1.0	0.006456	0.008300	0.009456	0.009874	0.010272	0.010395
1.5	0.008531	0.010182	0.011157	0.011365	0.011911	0.012117
2.0	0.010482	0.011921	0.012491	0.012578	0.013034	0.013160
2.5	0.011920	0.013206	0.013836	0.013911	0.014351	0.014591
3.0	0.013545	0.015164	0.015776	0.015918	0.016255	0.016503

Table A.39: Table for Case A:MSE q=-0.5, a=1, b=2, c=0.5,n=50, p=1000

(x, y)	0.5	1.0	1.5	2.0	2.5	3.0
0.5	0.220972	0.326256	0.368235	0.384117	0.390020	0.392199
1.0	0.372538	0.532966	0.595148	0.618452	0.627083	0.630266
1.5	0.470920	0.661590	0.734091	0.761083	0.771055	0.774730
2.0	0.532966	0.740798	0.818848	0.847776	0.858447	0.862377
2.5	0.571472	0.789280	0.870432	0.900425	0.911478	0.915547
3.0	0.595148	0.818848	0.901786	0.932384	0.943652	0.947799

Table A.40: Table for real values of $H(x, y)$ when q=-0.5, a=1, b=2, c=0.5,n=50, p=1000

(x, y)	0.5	1.0	1.5	2.0	2.5	3.0
0.5	0.217730	0.321350	0.362856	0.377505	0.383572	0.385982
1.0	0.370651	0.531249	0.594411	0.617095	0.625795	0.628821
1.5	0.470502	0.660276	0.735265	0.762627	0.772898	0.776124
2.0	0.531883	0.737628	0.818378	0.847274	0.858174	0.861866
2.5	0.571431	0.787238	0.871493	0.901317	0.912630	0.916600
3.0	0.596657	0.819772	0.906030	0.936398	0.947763	0.951877

Table A.41: Table for Case B: average values of $\hat{H}_n^{(2)}(x, y)$ q=-0.5, a=1, b=2, c=0.5,n=50, p=1000

(x, y)	0.5	1.0	1.5	2.0	2.5	3.0
0.5	0.003817	0.005567	0.006262	0.006689	0.007201	0.007351
1.0	0.006079	0.006982	0.007780	0.008097	0.008437	0.008543
1.5	0.007738	0.007799	0.008120	0.008097	0.008484	0.008612
2.0	0.009140	0.008380	0.007881	0.007478	0.007785	0.007795
2.5	0.010087	0.008847	0.008181	0.007568	0.007751	0.007800
3.0	0.011379	0.010160	0.009276	0.008605	0.008671	0.008699

Table A.42: Table for Case B:MSE q=-0.5, a=1, b=2, c=0.5,n=50, p=1000

(x, y)	0.5	1.0	1.5	2.0	2.5	3.0
0.5	0.220972	0.326256	0.368235	0.384117	0.390020	0.392199
1.0	0.372538	0.532966	0.595148	0.618452	0.627083	0.630266
1.5	0.470920	0.661590	0.734091	0.761083	0.771055	0.774730
2.0	0.532966	0.740798	0.818848	0.847776	0.858447	0.862377
2.5	0.571472	0.789280	0.870432	0.900425	0.911478	0.915547
3.0	0.595148	0.818848	0.901786	0.932384	0.943652	0.947799

Table A.43: Table for real values of $H(x, y)$ when q=-0.5, a=1, b=2, c=0.5,n=50, p=1000

(x, y)	0.5	1.0	1.5	2.0	2.5	3.0
0.5	0.217654	0.321256	0.362601	0.377328	0.382820	0.385170
1.0	0.370489	0.530962	0.593826	0.616538	0.624427	0.627259
1.5	0.470182	0.659549	0.734234	0.761410	0.770748	0.773736
2.0	0.531080	0.736446	0.816781	0.845467	0.855437	0.858939
2.5	0.569473	0.784455	0.868237	0.897921	0.908315	0.912006
3.0	0.591893	0.813264	0.898746	0.928849	0.939285	0.943083

Table A.44: Table for Case C: average values of $\hat{H}_n^{(3)}(x, y)$ q=-0.5, a=1, b=2, c=0.5,n=50, p=1000

(x, y)	0.5	1.0	1.5	2.0	2.5	3.0
0.5	0.003817	0.005548	0.006172	0.006637	0.007063	0.007212
1.0	0.006054	0.006852	0.007515	0.007829	0.008061	0.008079
1.5	0.007583	0.007222	0.007183	0.007009	0.007212	0.007286
2.0	0.008694	0.007265	0.006208	0.005380	0.005441	0.005395
2.5	0.009196	0.007127	0.005727	0.004629	0.004418	0.004334
3.0	0.009676	0.007010	0.005182	0.003951	0.003604	0.003454

Table A.45: Table for Case C:MSE q=-0.5, a=1, b=2, c=0.5,n=50, p=1000

(x, y)	0.5	1.0	1.5	2.0	2.5	3.0
0.5	0.220972	0.326256	0.368235	0.384117	0.390020	0.392199
1.0	0.372538	0.532966	0.595148	0.618452	0.627083	0.630266
1.5	0.470920	0.661590	0.734091	0.761083	0.771055	0.774730
2.0	0.532966	0.740798	0.818848	0.847776	0.858447	0.862377
2.5	0.571472	0.789280	0.870432	0.900425	0.911478	0.915547
3.0	0.595148	0.818848	0.901786	0.932384	0.943652	0.947799

Table A.46: Table for real values of $H(x, y)$ when q=-0.5, a=1, b=2, c=0.5,n=50, p=1000

(x, y)	0.5	1.0	1.5	2.0	2.5	3.0
0.5	0.217654	0.320209	0.358154	0.369700	0.373606	0.374849
1.0	0.369085	0.530962	0.593082	0.613661	0.620146	0.621969
1.5	0.461782	0.658484	0.734234	0.761192	0.769913	0.772331
2.0	0.511480	0.730837	0.816349	0.845467	0.855414	0.858632
2.5	0.537978	0.771694	0.866140	0.897863	0.908315	0.912003
3.0	0.551469	0.794062	0.894436	0.928398	0.939285	0.943083

Table A.47: Table for Case D: average values of $\hat{H}_n^{(4)}(x, y)$ q=-0.5, a=1, b=2, c=0.5,n=50, p=1000

(x, y)	0.5	1.0	1.5	2.0	2.5	3.0
0.5	0.003817	0.005504	0.006026	0.006331	0.006627	0.006684
1.0	0.006007	0.006852	0.007459	0.007608	0.007744	0.007726
1.5	0.007316	0.007186	0.007183	0.007005	0.007148	0.007194
2.0	0.008316	0.007191	0.006191	0.005380	0.005441	0.005378
2.5	0.009084	0.007126	0.005727	0.004631	0.004418	0.004335
3.0	0.010096	0.007403	0.005325	0.003978	0.003604	0.003454

Table A.48: Table for Case D:MSE q=-0.5, a=1, b=2, c=0.5,n=50, p=1000

(x, y)	0.5	1.0	1.5	2.0	2.5	3.0
0.5	0.193223	0.312292	0.362589	0.381972	0.389221	0.391904
1.0	0.345500	0.519360	0.589648	0.616362	0.626305	0.629979
1.5	0.450765	0.651447	0.729991	0.759524	0.770475	0.774516
2.0	0.519360	0.733951	0.816080	0.846724	0.858055	0.862232
2.5	0.562711	0.784872	0.868650	0.899748	0.911226	0.915453
3.0	0.589648	0.816080	0.900667	0.931959	0.943494	0.947741

Table A.49: Table for real values of $H(x, y)$ when q=-1, a=1, b=2, c=0.5,n=50, p=1000

(x, y)	0.5	1.0	1.5	2.0	2.5	3.0
0.5	0.192487	0.312168	0.362219	0.381266	0.388358	0.390965
1.0	0.344212	0.519133	0.588788	0.615518	0.625646	0.629416
1.5	0.451444	0.654351	0.732333	0.761759	0.772602	0.776763
2.0	0.520772	0.737011	0.819486	0.850026	0.861170	0.865332
2.5	0.564776	0.788798	0.872813	0.903586	0.914922	0.919083
3.0	0.590130	0.819182	0.903904	0.934891	0.946227	0.950388

Table A.50: Table for Case A: average values of $\hat{H}_n^{(1)}(x, y)$ q=-1, a=1, b=2, c=0.5,n=50, p=1000

(x, y)	0.5	1.0	1.5	2.0	2.5	3.0
0.5	0.003788	0.005929	0.007149	0.007540	0.007953	0.008042
1.0	0.006551	0.009061	0.009804	0.010008	0.010393	0.010490
1.5	0.009162	0.011079	0.011196	0.011275	0.011714	0.011971
2.0	0.011035	0.013381	0.012811	0.012777	0.013128	0.013401
2.5	0.013171	0.015158	0.014726	0.014743	0.015073	0.015277
3.0	0.014393	0.016470	0.015938	0.016035	0.016250	0.016531

Table A.51: Table for Case A:MSE q=-1, a=1, b=2, c=0.5,n=50, p=1000

(x,y)	0.5	1.0	1.5	2.0	2.5	3.0
0.5	0.193223	0.312292	0.362589	0.381972	0.389221	0.391904
1.0	0.345500	0.519360	0.589648	0.616362	0.626305	0.629979
1.5	0.450765	0.651447	0.729991	0.759524	0.770475	0.774516
2.0	0.519360	0.733951	0.816080	0.846724	0.858055	0.862232
2.5	0.562711	0.784872	0.868650	0.899748	0.911226	0.915453
3.0	0.589648	0.816080	0.900667	0.931959	0.943494	0.947741

Table A.52: Table for real values of $H(x,y)$ when q=-1, a=1, b=2, c=0.5,n=50, p=1000

(x,y)	0.5	1.0	1.5	2.0	2.5	3.0
0.5	0.192610	0.312202	0.362237	0.381424	0.388463	0.391068
1.0	0.344161	0.518904	0.588452	0.615246	0.625268	0.629069
1.5	0.451322	0.653975	0.731755	0.761452	0.772222	0.776352
2.0	0.520571	0.736479	0.818730	0.849485	0.860514	0.864645
2.5	0.564692	0.788502	0.872230	0.903226	0.914405	0.918536
3.0	0.590199	0.819052	0.903602	0.934807	0.945987	0.950118

Table A.53: Table for Case B: average values of $\hat{H}_n^{(2)}(x,y)$ q=-1, a=1, b=2, c=0.5,n=50, p=1000

(x,y)	0.5	1.0	1.5	2.0	2.5	3.0
0.5	0.003719	0.005648	0.006736	0.007044	0.007434	0.007493
1.0	0.006036	0.007670	0.007946	0.007973	0.008248	0.008322
1.5	0.007939	0.008181	0.007303	0.007128	0.007414	0.007635
2.0	0.009101	0.009170	0.007361	0.006894	0.006966	0.007218
2.5	0.010902	0.010258	0.008341	0.007852	0.007789	0.007976
3.0	0.012036	0.011294	0.009226	0.008782	0.008552	0.008795

Table A.54: Table for Case B:MSE q=-1, a=1, b=2, c=0.5,n=50, p=1000

(x, y)	0.5	1.0	1.5	2.0	2.5	3.0
0.5	0.193223	0.312292	0.362589	0.381972	0.389221	0.391904
1.0	0.345500	0.519360	0.589648	0.616362	0.626305	0.629979
1.5	0.450765	0.651447	0.729991	0.759524	0.770475	0.774516
2.0	0.519360	0.733951	0.816080	0.846724	0.858055	0.862232
2.5	0.562711	0.784872	0.868650	0.899748	0.911226	0.915453
3.0	0.589648	0.816080	0.900667	0.931959	0.943494	0.947741

Table A.55: Table for real values of $H(x, y)$ when q=-1, a=1, b=2, c=0.5,n=50, p=1000

(x, y)	0.5	1.0	1.5	2.0	2.5	3.0
0.5	0.192633	0.312338	0.362387	0.381490	0.388355	0.390537
1.0	0.344390	0.519327	0.588877	0.615604	0.625463	0.628609
1.5	0.451628	0.654304	0.732083	0.761797	0.772472	0.775911
2.0	0.519948	0.735679	0.817904	0.848610	0.859515	0.862954
2.5	0.562223	0.785585	0.869280	0.900306	0.911394	0.914833
3.0	0.585694	0.813401	0.897818	0.928988	0.940076	0.943514

Table A.56: Table for Case C: average values of $\hat{H}_n^{(3)}(x, y)$ q=-1, a=1, b=2, c=0.5,n=50, p=1000

(x, y)	0.5	1.0	1.5	2.0	2.5	3.0
0.5	0.003730	0.005606	0.006669	0.006938	0.007325	0.007389
1.0	0.006010	0.007463	0.007675	0.007692	0.007831	0.007858
1.5	0.007852	0.007842	0.006767	0.006430	0.006593	0.006695
2.0	0.008425	0.007954	0.005766	0.004852	0.004726	0.004784
2.5	0.009347	0.008018	0.005536	0.004540	0.004207	0.004128
3.0	0.009916	0.007994	0.005133	0.004086	0.003567	0.003476

Table A.57: Table for Case C:MSE q=-1, a=1, b=2, c=0.5,n=50, p=1000

(x, y)	0.5	1.0	1.5	2.0	2.5	3.0
0.5	0.193223	0.312292	0.362589	0.381972	0.389221	0.391904
1.0	0.345500	0.519360	0.589648	0.616362	0.626305	0.629979
1.5	0.450765	0.651447	0.729991	0.759524	0.770475	0.774516
2.0	0.519360	0.733951	0.816080	0.846724	0.858055	0.862232
2.5	0.562711	0.784872	0.868650	0.899748	0.911226	0.915453
3.0	0.589648	0.816080	0.900667	0.931959	0.943494	0.947741

Table A.58: Table for real values of $H(x, y)$ when q=-1, a=1, b=2, c=0.5,n=50, p=1000

(x, y)	0.5	1.0	1.5	2.0	2.5	3.0
0.5	0.192633	0.311203	0.357391	0.372520	0.377113	0.378444
1.0	0.342870	0.519327	0.588031	0.612490	0.620257	0.622504
1.5	0.441977	0.653151	0.732083	0.761467	0.771347	0.774322
2.0	0.497101	0.729495	0.817366	0.848610	0.859423	0.862765
2.5	0.526059	0.771586	0.866530	0.900160	0.911394	0.914833
3.0	0.539758	0.793126	0.892817	0.928515	0.940068	0.943514

Table A.59: Table for Case D: average values of $\hat{H}_n^{(4)}(x, y)$ q=-1, a=1, b=2, c=0.5,n=50, p=1000

(x, y)	0.5	1.0	1.5	2.0	2.5	3.0
0.5	0.003730	0.005570	0.006476	0.006619	0.006830	0.006860
1.0	0.005945	0.007463	0.007645	0.007547	0.007618	0.007582
1.5	0.007518	0.007799	0.006767	0.006405	0.006525	0.006574
2.0	0.008167	0.007874	0.005748	0.004852	0.004720	0.004775
2.5	0.009534	0.008025	0.005505	0.004537	0.004207	0.004128
3.0	0.010849	0.008520	0.005290	0.004105	0.003567	0.003476

Table A.60: Table for Case D:MSE q=-1, a=1, b=2, c=0.5,n=50, p=1000

Table A.61: Table for (average values of $\hat{H}_n^{(1)}(x,y)$ - real values of $H(x,y)$)

q	(x,y)	0.5	1.0	1.5	2.0	2.5	3.0
1	0.5	-1.1E-05	0.000117	0.000559	0.000684	0.001016	0.001028
	1	-0.002753	-0.001731	-0.000968	-0.000623	3E-06	-8E-06
	1.5	-0.004242	-0.000314	-0.000374	-0.000286	0.000892	0.000779
	2	-0.003357	0.00055	-0.000101	-0.000623	0.000354	0.000475
	2.5	-0.004202	-6.9E-05	-0.000214	-0.001024	8.1E-05	9.6E-05
	3	-0.004898	-0.000656	-0.000565	-0.000945	0.000265	0.000437
0.5	0.5	-0.002219	-0.000497	-0.000171	-0.001239	-0.000895	-0.000473
	1	-0.003763	-0.000634	-0.000741	-0.002347	-0.00208	-0.001912
	1.5	-0.005875	-0.002184	-0.001583	-0.003996	-0.003268	-0.003325
	2	-0.006341	-0.003441	-0.002549	-0.00507	-0.004569	-0.004717
	2.5	-0.006463	-0.001653	-0.001073	-0.003837	-0.003376	-0.00368
	3	-0.006366	-0.00231	-0.000439	-0.003287	-0.002661	-0.003086
0	0.5	-0.00155	-0.000697	-0.000319	-0.000151	-0.000538	-0.000646
	1	-0.001244	-0.000546	-0.001085	-0.000829	-0.001714	-0.002044
	1.5	-0.000105	-8.2E-05	-0.001291	-0.001487	-0.001623	-0.002029
	2	0.00131	0.000514	-0.000666	-0.000516	-0.001019	-0.001644
	2.5	0.001581	0.000371	-0.000746	-0.00092	-0.001856	-0.00271
	3	0.000934	-0.000884	-0.001831	-0.001327	-0.002017	-0.002763
-0.5	0.5	-0.00304	-0.004614	-0.005183	-0.006453	-0.006262	-0.005987
	1	-0.00182	-0.001606	-0.000907	-0.001771	-0.001824	-0.001951
	1.5	-0.000274	-0.001339	0.000797	0.000968	0.001251	0.000862
	2	-0.00054	-0.002896	-0.000687	-0.000846	-0.000649	-0.000917
	2.5	0.000602	-0.001693	0.001021	0.000774	0.001025	0.000853
	3	0.001928	0.000903	0.003819	0.003525	0.003631	0.003531
-1	0.5	-0.000736	-0.000124	-0.00037	-0.000706	-0.000863	-0.000939
	1	-0.001288	-0.000227	-0.00086	-0.000844	-0.000659	-0.000563
	1.5	0.000679	0.002904	0.002342	0.002235	0.002127	0.002247
	2	0.001412	0.00306	0.003406	0.003302	0.003115	0.0031
	2.5	0.002065	0.003926	0.004163	0.003838	0.003696	0.00363
	3	0.000482	0.003102	0.003237	0.002932	0.002733	0.002647

Table A.62: Table for (average values of $\hat{H}_n^{(2)}(x,y)$ - real values of $H(x,y)$)

q	(x,y)	0.5	1.0	1.5	2.0	2.5	3.0
1	0.5	1.2E-05	0.000214	0.000385	0.00057	0.00103	0.000973
	1	-0.002472	-0.001167	-0.000948	-0.000665	-3.8E-05	-0.000174
	1.5	-0.00408	0.000374	-0.000274	-0.000206	0.000914	0.000745
	2	-0.003524	0.000893	-0.000416	-0.001059	-0.000192	-5.4E-05
	2.5	-0.004519	0.000641	0.000159	-0.000805	0.000296	0.000396
	3	-0.005113	0.000101	0.00013	-0.000564	0.000664	0.00104
0.5	0.5	-0.002348	-0.000438	2.8E-05	-0.001135	-0.000924	-0.00067
	1	-0.003156	0.000589	0.00054	-0.001313	-0.001209	-0.001194
	1.5	-0.005162	-0.000682	-0.000107	-0.002842	-0.002265	-0.002478
	2	-0.00609	-0.00235	-0.001382	-0.004164	-0.003849	-0.00421
	2.5	-0.006321	-0.001069	-0.000275	-0.003437	-0.00313	-0.003689
	3	-0.006651	-0.002058	-0.000509	-0.003821	-0.00339	-0.004089
0	0.5	-0.001495	-0.000696	-0.000329	-0.00019	-0.000499	-0.000747
	1	-0.000905	-0.000106	-0.000706	-0.000495	-0.001179	-0.00159
	1.5	0.000515	0.000683	-0.00063	-0.000867	-0.000744	-0.001259
	2	0.001737	0.001053	-0.000165	-8.7E-05	-0.000377	-0.001105
	2.5	0.002169	0.000704	-0.000429	-0.000724	-0.001443	-0.0024
	3	0.001379	-0.000658	-0.001456	-0.001098	-0.001586	-0.002475
-0.5	0.5	-0.003242	-0.004906	-0.005379	-0.006612	-0.006448	-0.006217
	1	-0.001887	-0.001717	-0.000737	-0.001357	-0.001288	-0.001445
	1.5	-0.000418	-0.001314	0.001174	0.001544	0.001843	0.001394
	2	-0.001083	-0.00317	-0.00047	-0.000502	-0.000273	-0.000511
	2.5	-4.1E-05	-0.002042	0.001061	0.000892	0.001152	0.001053
	3	0.001509	0.000924	0.004244	0.004014	0.004111	0.004078
-1	0.5	-0.000613	-9E-05	-0.000352	-0.000548	-0.000758	-0.000836
	1	-0.001339	-0.000456	-0.001196	-0.001116	-0.001037	-0.00091
	1.5	0.000557	0.002528	0.001764	0.001928	0.001747	0.001836
	2	0.001211	0.002528	0.00265	0.002761	0.002459	0.002413
	2.5	0.001981	0.00363	0.00358	0.003478	0.003179	0.003083
	3	0.000551	0.002972	0.002935	0.002848	0.002493	0.002377

Table A.63: Table for (average values of $\hat{H}_n^{(3)}(x,y)$ - real values of $H(x,y)$)

q	(x,y)	0.5	1.0	1.5	2.0	2.5	3.0
1	0.5	0.000218	0.000615	0.000843	0.001028	0.001408	0.001224
	1	-0.002281	-0.000863	-0.000548	-0.000299	-2.3E-05	-0.000369
	1.5	-0.0036	0.000984	0.000433	0.000611	0.001172	0.000851
	2	-0.002366	0.001593	0.000579	3.9E-05	0.000512	0.000516
	2.5	-0.003963	-2.8E-05	-0.000605	-0.00154	-0.000896	-0.000851
	3	-0.006703	-0.003926	-0.00454	-0.005536	-0.004984	-0.004915
0.5	0.5	-0.002258	-0.000408	2.8E-05	-0.00103	-0.000871	-0.000641
	1	-0.003112	0.000476	0.000532	-0.001164	-0.001047	-0.001078
	1.5	-0.00503	-0.000788	-0.000141	-0.002708	-0.002388	-0.00273
	2	-0.005106	-0.001456	-0.00043	-0.002987	-0.00289	-0.003483
	2.5	-0.005274	-9.7E-05	0.000193	-0.002952	-0.002803	-0.003605
	3	-0.006813	-0.00352	-0.002837	-0.006506	-0.00628	-0.00723
0	0.5	-0.001641	-0.000656	-0.000325	-0.000422	-0.000775	-0.001026
	1	-0.000978	-1.2E-05	-0.000575	-0.000863	-0.001869	-0.002517
	1.5	0.000161	0.000422	-0.000874	-0.001576	-0.002004	-0.002829
	2	0.001185	0.000998	-0.000493	-0.000928	-0.001845	-0.002727
	2.5	0.000166	-0.001194	-0.002799	-0.003797	-0.00511	-0.00622
	3	-0.003162	-0.006347	-0.008162	-0.008529	-0.009653	-0.010683
-0.5	0.5	-0.003318	-0.005	-0.005634	-0.006789	-0.0072	-0.007029
	1	-0.002049	-0.002004	-0.001322	-0.001914	-0.002656	-0.003007
	1.5	-0.000738	-0.002041	0.000143	0.000327	-0.000307	-0.000994
	2	-0.001886	-0.004352	-0.002067	-0.002309	-0.00301	-0.003438
	2.5	-0.001999	-0.004825	-0.002195	-0.002504	-0.003163	-0.003541
	3	-0.003255	-0.005584	-0.00304	-0.003535	-0.004367	-0.004716
-1	0.5	-0.00059	4.6E-05	-0.000202	-0.000482	-0.000866	-0.001367
	1	-0.00111	-3.3E-05	-0.000771	-0.000758	-0.000842	-0.00137
	1.5	0.000863	0.002857	0.002092	0.002273	0.001997	0.001395
	2	0.000588	0.001728	0.001824	0.001886	0.00146	0.000722
	2.5	-0.000488	0.000713	0.00063	0.000558	0.000168	-0.00062
	3	-0.003954	-0.002679	-0.002849	-0.002971	-0.003418	-0.004227

Table A.64: Table for (average values of $\hat{H}_n^{(4)}(x,y)$ - real values of $H(x,y)$)

q	(x,y)	0.5	1.0	1.5	2.0	2.5	3.0
1	0.5	0.000218	1.3E-05	-0.00119	-0.002234	-0.002704	-0.003038
	1	-0.003379	-0.000863	-0.000922	-0.001949	-0.002687	-0.003412
	1.5	-0.009628	0.000239	0.000433	0.000414	0.000494	-0.000179
	2	-0.016157	-0.002492	0.000276	3.9E-05	0.000459	0.000344
	2.5	-0.024503	-0.009186	-0.002347	-0.001587	-0.000896	-0.000881
	3	-0.031194	-0.016986	-0.008022	-0.005864	-0.004984	-0.004915
0.5	0.5	-0.002258	-0.001148	-0.002821	-0.005466	-0.006449	-0.006823
	1	-0.004326	0.000476	7.6E-05	-0.002795	-0.003997	-0.004782
	1.5	-0.01181	-0.001661	-0.000141	-0.002965	-0.003231	-0.004068
	2	-0.02113	-0.00626	-0.000792	-0.002987	-0.002961	-0.003712
	2.5	-0.030112	-0.010862	-0.001504	-0.003008	-0.002803	-0.003605
	3	-0.037865	-0.019073	-0.005872	-0.006637	-0.00628	-0.00723
0	0.5	-0.001641	-0.001594	-0.004087	-0.006691	-0.00851	-0.009394
	1	-0.002328	-1.2E-05	-0.0012	-0.003126	-0.005299	-0.006523
	1.5	-0.007723	-0.000609	-0.000874	-0.001803	-0.0028	-0.00398
	2	-0.017042	-0.004538	-0.000929	-0.000928	-0.001863	-0.002783
	2.5	-0.027792	-0.012617	-0.004944	-0.003918	-0.00511	-0.006247
	3	-0.037574	-0.022673	-0.011791	-0.008774	-0.00967	-0.010683
-0.5	0.5	-0.003318	-0.006047	-0.010081	-0.014417	-0.016414	-0.01735
	1	-0.003453	-0.002004	-0.002066	-0.004791	-0.006937	-0.008297
	1.5	-0.009138	-0.003106	0.000143	0.000109	-0.001142	-0.002399
	2	-0.021486	-0.009961	-0.002499	-0.002309	-0.003033	-0.003745
	2.5	-0.033494	-0.017586	-0.004292	-0.002562	-0.003163	-0.003544
	3	-0.043679	-0.024786	-0.00735	-0.003986	-0.004367	-0.004716
-1	0.5	-0.00059	-0.001089	-0.005198	-0.009452	-0.012108	-0.01346
	1	-0.00263	-3.3E-05	-0.001617	-0.003872	-0.006048	-0.007475
	1.5	-0.008788	0.001704	0.002092	0.001943	0.000872	-0.000194
	2	-0.022259	-0.004456	0.001286	0.001886	0.001368	0.000533
	2.5	-0.036652	-0.013286	-0.00212	0.000412	0.000168	-0.00062
	3	-0.04989	-0.022954	-0.00785	-0.003444	-0.003426	-0.004227

Table A.65: Table for the values of (MSE of Case B - MSE of Case A)

q	(x, y)	0.5	1.0	1.5	2.0	2.5	3.0
1	0.5	-0.000171	-0.000218	-0.000342	-0.000291	-0.000282	-0.000289
	1	-0.000517	-0.001049	-0.001448	-0.001548	-0.001644	-0.001696
	1.5	-0.000976	-0.001999	-0.002945	-0.003126	-0.003325	-0.003374
	2	-0.00143	-0.003029	-0.004426	-0.004846	-0.005111	-0.005224
	2.5	-0.001705	-0.003525	-0.005024	-0.005689	-0.005987	-0.006067
	3	-0.001636	-0.003653	-0.00514	-0.006075	-0.006439	-0.006433
0.5	0.5	-0.000292	-0.000547	-0.000543	-0.000552	-0.000616	-0.000649
	1	-0.000686	-0.001403	-0.00173	-0.001905	-0.002034	-0.002162
	1.5	-0.001071	-0.002336	-0.002933	-0.003267	-0.003406	-0.003575
	2	-0.00153	-0.003297	-0.004238	-0.004638	-0.004767	-0.004992
	2.5	-0.001925	-0.004089	-0.005156	-0.005722	-0.005904	-0.006156
	3	-0.002146	-0.004591	-0.006138	-0.006896	-0.007226	-0.007483
0	0.5	-0.000163	-0.000387	-0.000478	-0.000515	-0.00054	-0.000558
	1	-0.000485	-0.001274	-0.001704	-0.001929	-0.001954	-0.001966
	1.5	-0.000866	-0.002145	-0.003034	-0.003396	-0.003373	-0.003386
	2	-0.001285	-0.003136	-0.004406	-0.004917	-0.004867	-0.004894
	2.5	-0.001405	-0.00364	-0.005215	-0.005826	-0.005763	-0.005858
	3	-0.001868	-0.004255	-0.005902	-0.006613	-0.006594	-0.006772
-0.5	0.5	-9.1E-05	-0.000364	-0.000525	-0.000558	-0.000538	-0.000509
	1	-0.000377	-0.001318	-0.001676	-0.001777	-0.001835	-0.001852
	1.5	-0.000793	-0.002383	-0.003037	-0.003268	-0.003427	-0.003505
	2	-0.001342	-0.003541	-0.00461	-0.0051	-0.005249	-0.005365
	2.5	-0.001833	-0.004359	-0.005655	-0.006343	-0.0066	-0.006791
	3	-0.002166	-0.005004	-0.0065	-0.007313	-0.007584	-0.007804
-1	0.5	-6.9E-05	-0.000281	-0.000413	-0.000496	-0.000519	-0.000549
	1	-0.000515	-0.001391	-0.001858	-0.002035	-0.002145	-0.002168
	1.5	-0.001223	-0.002898	-0.003893	-0.004147	-0.0043	-0.004336
	2	-0.001934	-0.004211	-0.00545	-0.005883	-0.006162	-0.006183
	2.5	-0.002269	-0.0049	-0.006385	-0.006891	-0.007284	-0.007301
	3	-0.002357	-0.005176	-0.006712	-0.007253	-0.007698	-0.007736

Table A.66: Table for the values of (MSE of Case C - MSE of Case A)

q	(x, y)	0.5	1.0	1.5	2.0	2.5	3.0
1	0.5	-.000153	-.000179	-.000316	-.000259	-.000209	-.000234
	1	-.000537	-.001173	-.001656	-.001736	-.001844	-.001956
	1.5	-.001094	-.002415	-.003709	-.003946	-.004293	-.004395
	2	-.00136	-.003808	-.005771	-.006391	-.006817	-.006998
	2.5	-.001729	-.004946	-.007383	-.008533	-.00913	-.009276
	3	-.002208	-.006179	-.008944	-.010475	-.011318	-.011568
0.5	0.5	-.000296	-.000587	-.000597	-.000556	-.000603	-.00061
	1	-.000747	-.001597	-.001966	-.002101	-.002229	-.00237
	1.5	-.001299	-.002984	-.003809	-.004255	-.004487	-.004695
	2	-.001906	-.004268	-.005809	-.006494	-.006838	-.007098
	2.5	-.002345	-.005457	-.007686	-.008763	-.009339	-.009678
	3	-.002937	-.006824	-.009799	-.011159	-.011975	-.012353
0	0.5	-.000186	-.000366	-.000473	-.000707	-.000766	-.000773
	1	-.000563	-.001309	-.00182	-.002336	-.002466	-.002526
	1.5	-.001156	-.002546	-.003689	-.00445	-.004562	-.004628
	2	-.001793	-.004038	-.005699	-.006879	-.007006	-.007048
	2.5	-.002234	-.005251	-.00731	-.00873	-.008968	-.009144
	3	-.003279	-.006974	-.009189	-.010785	-.01111	-.011411
-0.5	0.5	-9.1E-05	-.000383	-.000615	-.00061	-.000676	-.000648
	1	-.000402	-.001448	-.001941	-.002045	-.002211	-.002316
	1.5	-.000948	-.00296	-.003974	-.004356	-.004699	-.004831
	2	-.001788	-.004656	-.006283	-.007198	-.007593	-.007765
	2.5	-.002724	-.006079	-.008109	-.009282	-.009933	-.010257
	3	-.003869	-.008154	-.010594	-.011967	-.012651	-.013049
-1	0.5	-5.8E-05	-.000323	-.00048	-.000602	-.000628	-.000653
	1	-.000541	-.001598	-.002129	-.002316	-.002562	-.002632
	1.5	-.00131	-.003237	-.004429	-.004845	-.005121	-.005276
	2	-.00261	-.005427	-.007045	-.007925	-.008402	-.008617
	2.5	-.003824	-.00714	-.00919	-.010203	-.010866	-.011149
	3	-.004477	-.008476	-.010805	-.011949	-.012683	-.013055

Table A.67: Table for the values of (MSE of Case D - MSE of Case A)

q	(x, y)	0.5	1.0	1.5	2.0	2.5	3.0
1	0.5	-0.000153	-0.000214	-0.0004	-0.000419	-0.000445	-0.00048
	1.0	-0.000583	-0.001173	-0.001659	-0.001824	-0.001994	-0.002116
	1.5	-0.001242	-0.002466	-0.003709	-0.003964	-0.00432	-0.004438
	2.0	-0.001812	-0.003944	-0.005769	-0.006391	-0.00682	-0.007
	2.5	-0.002358	-0.005094	-0.007366	-0.008528	-0.00913	-0.009274
	3.0	-0.002498	-0.006081	-0.008824	-0.010462	-0.011318	-0.011568
0.5	0.5	-0.000296	-0.000616	-0.000763	-0.000815	-0.000966	-0.001047
	1.0	-0.00079	-0.001597	-0.00199	-0.00216	-0.002394	-0.002526
	1.5	-0.001442	-0.003023	-0.003809	-0.004277	-0.004538	-0.004766
	2.0	-0.002061	-0.004324	-0.005813	-0.006494	-0.00684	-0.007097
	2.5	-0.002554	-0.005538	-0.007633	-0.00876	-0.009339	-0.009678
	3.0	-0.00301	-0.006647	-0.009652	-0.011149	-0.011975	-0.012353
0	0.5	-0.000186	-0.000413	-0.000649	-0.00093	-0.001046	-0.001132
	1.0	-0.000614	-0.001309	-0.001837	-0.002391	-0.002541	-0.002629
	1.5	-0.001389	-0.002598	-0.003689	-0.004463	-0.004567	-0.004646
	2.0	-0.002211	-0.004248	-0.00571	-0.006879	-0.007006	-0.007052
	2.5	-0.002623	-0.005308	-0.007298	-0.008727	-0.008968	-0.009145
	3.0	-0.003066	-0.006592	-0.009052	-0.010771	-0.01111	-0.011411
-0.5	0.5	-9.1E-05	-0.000427	-0.000761	-0.000916	-0.001112	-0.001176
	1.0	-0.000449	-0.001448	-0.001997	-0.002266	-0.002528	-0.002669
	1.5	-0.001215	-0.002996	-0.003974	-0.00436	-0.004763	-0.004923
	2.0	-0.002166	-0.00473	-0.0063	-0.007198	-0.007593	-0.007782
	2.5	-0.002836	-0.00608	-0.008109	-0.00928	-0.009933	-0.010256
	3.0	-0.003449	-0.007761	-0.010451	-0.01194	-0.012651	-0.013049
-1	0.5	-5.8E-05	-0.000359	-0.000673	-0.000921	-0.001123	-0.001182
	1.0	-0.000606	-0.001598	-0.002159	-0.002461	-0.002775	-0.002908
	1.5	-0.001644	-0.00328	-0.004429	-0.00487	-0.005189	-0.005397
	2.0	-0.002868	-0.005507	-0.007063	-0.007925	-0.008408	-0.008626
	2.5	-0.003637	-0.007133	-0.009221	-0.010206	-0.010866	-0.011149
	3.0	-0.003544	-0.00795	-0.010648	-0.01193	-0.012683	-0.013055

Table A.68: Table for the values of (MSE of Case C- MSE of Case B)

q	(x, y)	0.5	1.0	1.5	2.0	2.5	3.0
1	0.5	1.8E-05	0.000039	0.000026	3.2E-05	7.3E-05	5.5E-05
	1	-2E-05	-0.000124	-0.000208	-0.000188	-0.0002	-0.00026
	1.5	-0.000118	-0.000416	-0.000764	-0.00082	-0.000968	-0.001021
	2	7E-05	-0.000779	-0.001345	-0.001545	-0.001706	-0.001774
	2.5	-2.4E-05	-0.001421	-0.002359	-0.002844	-0.003143	-0.003209
	3	-0.000572	-0.002526	-0.003804	-0.0044	-0.004879	-0.005135
0.5	0.5	-4E-06	-4E-05	-5.4E-05	-4E-06	0.000013	0.000039
	1	-6.1E-05	-0.000194	-0.000236	-0.000196	-0.000195	-0.000208
	1.5	-0.000228	-0.000648	-0.000876	-0.000988	-0.001081	-0.00112
	2	-0.000376	-0.000971	-0.001571	-0.001856	-0.002071	-0.002106
	2.5	-0.00042	-0.001368	-0.00253	-0.003041	-0.003435	-0.003522
	3	-0.000791	-0.002233	-0.003661	-0.004263	-0.004749	-0.00487
0	0.5	-2.3E-05	2.1E-05	5E-06	-0.000192	-0.000226	-0.000215
	1	-7.8E-05	-3.5E-05	-0.000116	-0.000407	-0.000512	-0.00056
	1.5	-0.00029	-0.000401	-0.000655	-0.001054	-0.001189	-0.001242
	2	-0.000508	-0.000902	-0.001293	-0.001962	-0.002139	-0.002154
	2.5	-0.000829	-0.001611	-0.002095	-0.002904	-0.003205	-0.003286
	3	-0.001411	-0.002719	-0.003287	-0.004172	-0.004516	-0.004639
-0.5	0.5	0	-1.9E-05	-9E-05	-0.000052	-0.000138	-0.000139
	1	-2.5E-05	-0.00013	-0.000265	-0.000268	-0.000376	-0.000464
	1.5	-0.000155	-0.000577	-0.000937	-0.001088	-0.001272	-0.001326
	2	-0.000446	-0.001115	-0.001673	-0.002098	-0.002344	-0.0024
	2.5	-0.000891	-0.00172	-0.002454	-0.002939	-0.003333	-0.003466
	3	-0.001703	-0.00315	-0.004094	-0.004654	-0.005067	-0.005245
-1	0.5	1.1E-05	-4.2E-05	-6.7E-05	-0.000106	-0.000109	-0.000104
	1	-0.000026	-0.000207	-0.000271	-0.000281	-0.000417	-0.000464
	1.5	-8.7E-05	-0.000339	-0.000536	-0.000698	-0.000821	-0.00094
	2	-0.000676	-0.001216	-0.001595	-0.002042	-0.00224	-0.002434
	2.5	-0.001555	-0.00224	-0.002805	-0.003312	-0.003582	-0.003848
	3	-0.00212	-0.0033	-0.004093	-0.004696	-0.004985	-0.005319

Table A.69: Table for the values of (MSE of Case D - MSE of Case B)

q	(x, y)	0.5	1.0	1.5	2.0	2.5	3.0
1	0.5	1.8E-05	4E-06	-5.8E-05	-0.000128	-0.000163	-0.000191
	1	-6.6E-05	-0.000124	-0.000211	-0.000276	-0.00035	-0.00042
	1.5	-0.000266	-0.000467	-0.000764	-0.000838	-0.000995	-0.001064
	2	-0.000382	-0.000915	-0.001343	-0.001545	-0.001709	-0.001776
	2.5	-0.000653	-0.001569	-0.002342	-0.002839	-0.003143	-0.003207
	3	-0.000862	-0.002428	-0.003684	-0.004387	-0.004879	-0.005135
0.5	0.5	-4E-06	-6.9E-05	-0.00022	-0.000263	-0.00035	-0.000398
	1	-0.000104	-0.000194	-0.00026	-0.000255	-0.00036	-0.000364
	1.5	-0.000371	-0.000687	-0.000876	-0.00101	-0.001132	-0.001191
	2	-0.000531	-0.001027	-0.001575	-0.001856	-0.002073	-0.002105
	2.5	-0.000629	-0.001449	-0.002477	-0.003038	-0.003435	-0.003522
	3	-0.000864	-0.002056	-0.003514	-0.004253	-0.004749	-0.00487
0	0.5	-2.3E-05	-0.000026	-0.000171	-0.000415	-0.000506	-0.000574
	1	-0.000129	-3.5E-05	-0.000133	-0.000462	-0.000587	-0.000663
	1.5	-0.000523	-0.000453	-0.000655	-0.001067	-0.001194	-0.00126
	2	-0.000926	-0.001112	-0.001304	-0.001962	-0.002139	-0.002158
	2.5	-0.001218	-0.001668	-0.002083	-0.002901	-0.003205	-0.003287
	3	-0.001198	-0.002337	-0.00315	-0.004158	-0.004516	-0.004639
-0.5	0.5	0	-6.3E-05	-0.000236	-0.000358	-0.000574	-0.000667
	1	-0.000072	-0.00013	-0.000321	-0.000489	-0.000693	-0.000817
	1.5	-0.000422	-0.000613	-0.000937	-0.001092	-0.001336	-0.001418
	2	-0.000824	-0.001189	-0.00169	-0.002098	-0.002344	-0.002417
	2.5	-0.001003	-0.001721	-0.002454	-0.002937	-0.003333	-0.003465
	3	-0.001283	-0.002757	-0.003951	-0.004627	-0.005067	-0.005245
-1	0.5	1.1E-05	-7.8E-05	-0.00026	-0.000425	-0.000604	-0.000633
	1	-9.1E-05	-0.000207	-0.000301	-0.000426	-0.00063	-0.00074
	1.5	-0.000421	-0.000382	-0.000536	-0.000723	-0.000889	-0.001061
	2	-0.000934	-0.001296	-0.001613	-0.002042	-0.002246	-0.002443
	2.5	-0.001368	-0.002233	-0.002836	-0.003315	-0.003582	-0.003848
	3	-0.001187	-0.002774	-0.003936	-0.004677	-0.004985	-0.005319

Table A.70: Table for the values of (MSE of Case C - MSE of Case D)

q	(x, y)	0.5	1.0	1.5	2.0	2.5	3.0
1	0.5	0	3.5E-05	8.4E-05	0.00016	0.000236	0.000246
	1	0.000046	0	3E-06	8.8E-05	0.00015	0.00016
	1.5	0.000148	5.1E-05	0	1.8E-05	2.7E-05	4.3E-05
	2	0.000452	0.000136	-2E-06	0	3E-06	2E-06
	2.5	0.000629	0.000148	-1.7E-05	-5E-06	0	-2E-06
	3	0.00029	-0.000098	-0.00012	-0.000013	0	0
0.5	0.5	0	2.9E-05	0.000166	0.000259	0.000363	0.000437
	1	4.3E-05	0	2.4E-05	0.000059	0.000165	0.000156
	1.5	0.000143	0.000039	0	2.2E-05	5.1E-05	7.1E-05
	2	0.000155	5.6E-05	4E-06	0	2E-06	-1E-06
	2.5	0.000209	8.1E-05	-5.3E-05	-3E-06	0	0
	3	7.3E-05	-0.000177	-0.000147	-1E-05	0	0
0	0.5	0	4.7E-05	0.000176	0.000223	0.00028	0.000359
	1	5.1E-05	0	1.7E-05	5.5E-05	7.5E-05	0.000103
	1.5	0.000233	0.000052	0	0.000013	5E-06	1.8E-05
	2	0.000418	0.00021	1.1E-05	0	0	4E-06
	2.5	0.000389	5.7E-05	-1.2E-05	-3E-06	0	1E-06
	3	-0.000213	-0.000382	-0.000137	-1.4E-05	0	0
-0.5	0.5	0	4.4E-05	0.000146	0.000306	0.000436	0.000528
	1	4.7E-05	0	5.6E-05	0.000221	0.000317	0.000353
	1.5	0.000267	3.6E-05	0	4E-06	6.4E-05	9.2E-05
	2	0.000378	7.4E-05	1.7E-05	0	0	1.7E-05
	2.5	0.000112	1E-06	0	-2E-06	0	-1E-06
	3	-0.00042	-0.000393	-0.000143	-2.7E-05	0	0
-1	0.5	0	3.6E-05	0.000193	0.000319	0.000495	0.000529
	1	0.000065	0	3E-05	0.000145	0.000213	0.000276
	1.5	0.000334	4.3E-05	0	2.5E-05	6.8E-05	0.000121
	2	0.000258	8E-05	1.8E-05	0	6E-06	9E-06
	2.5	-0.000187	-7E-06	3.1E-05	3E-06	0	0
	3	-0.000933	-0.000526	-0.000157	-1.9E-05	0	0

Appendix B

R-programming codes

Overall R code for producing the simulations and Tables of Appendix A.

```
pa=1000;
p=pa;
m=7;
Mp1<- array(0,c(m,m,p))
Mp2<- array(0,c(m,m,p))
Mp3<- array(0,c(m,m,p))
Mp4<- array(0,c(m,m,p))
Mp5<- array(0,c(m,m,p))
Ap1=matrix(0.0,m,m)
Ap2=matrix(0.0,m,m)
Ap3=matrix(0.0,m,m)
Ap4=matrix(0.0,m,m)
Ap5=matrix(0.0,m,m)
```

```
Sp1=matrix(0.0,m,m)
Sp2=matrix(0.0,m,m)
Sp3=matrix(0.0,m,m)
Sp4=matrix(0.0,m,m)
Sp5=matrix(0.0,m,m)

Bp1=matrix(0.0,m,m)
Bp2=matrix(0.0,m,m)
Bp3=matrix(0.0,m,m)
Bp4=matrix(0.0,m,m)
Bp5=matrix(0.0,m,m)

B1p1=matrix(0.0,m,m)
B1p2=matrix(0.0,m,m)
B1p3=matrix(0.0,m,m)
B1p4=matrix(0.0,m,m)
B1p5=matrix(0.0,m,m)

#c<- array(1,c(10,10,10))
for(ip in 1:pa)
{                     #a1
######How to simulate the H(x, y) distribution

# Given U ~ U(0-1),generate #n of V by f(u,v)=1+q(1-2u)*(1-2v)

RejectionSampling <- function(w1,q,n1) { RN <- NULL   #a2
for(i in 1:n1)
```

```
{ OK <- 0
while(OK<1)
{x1 <- runif(1)
U1 <- runif(1)
r1=(1+q*(1-2*w1)*(1-2*x1))/(1+abs(q))
if(U1<r1)
{
OK <- 1; RN <- c(RN,x1)
}
}
}

RN }      #a2

q=-1;  # given  q:relation between U and V
q7=q;
n=50;  # sample size
U <- runif(n);  # generate n-dimension of vector  U ~ U(0-1)
V=rep(0,n);     # generate  n-dimension of vector  V=0

# generate #n of X~exp(1) and Y~exp(2) by U and V

for (i in 1:n){V[i]=RejectionSampling(U[i],q,1)}
a=1
b=2
X=-(log(1-U))/a
Y=-(log(1-V))/b
```

```
par(mfrow=c(1,2))
plot(X,Y,type="p",main="Simulation Plot of  X(exp(1) and
Y(exp(2)(q=0.5))")
ca=0.5;
C=rexp(n,ca);# generate n-dimension of vector  C ~ exp(0.5)
plot(C,C,type="p",col = "blue",main="Simulation Plot of
C(exp(0.5))")


# computing  CDF of simulations using M1
m=7

M1=matrix(0.0,m,m)
M2=matrix(0.0,m,m)
M3=matrix(0.0,m,m)
M4=matrix(0.0,m,m)
M5=matrix(0.0,m,m)
M6=matrix(0.0,m,m)
M7=matrix(0.0,m,m)
  for (i in 1:m){      #a3
     for (j in 1:m){   #a4
x11=5*(i-1)/10;
y11=5*(j-1)/10;
M6[i,j]=(1-exp(-x11*a))*(1-exp(-y11*b))*(1+q*exp(-x11*a-y11*b));
          for (k in 1:n){
if (X[k]<=(5*(i-1)/10) && Y[k]<=(5*(j-1)/10))
 {M1[i,j]=M1[i,j]+1}
```

```
                        }
 for (k in 1:n){
if (X[k]<=5*(i-1)/10 && Y[k]<=5*(j-1)/10){
if (X[k]<=C[k] && Y[k]<=C[k]) {
hh=max(X[k],Y[k])
M2[i,j]=M2[i,j]+1/exp(-0.5*hh)}
                                         }
               }


                    }   #a3
               }   #a4
M1=M1/n;
M2=M2/n;

C1=rep(0,n);   # generate  n-dimension of vector  C1=0

# computing  #n of weights C1
for (i in 1:n){
if  (X[i]<C[i] & Y[i]<C[i]){
for (j in 1:n){
if (X[i]<C[j] & Y[i]<C[j]){ C1[i]=C1[i]+1}
             }
                              }
             }
  mean(C1)
 C1
# computing  CDF of simulations using M1
m=7
```

```
M3=matrix(0.0,m,m)



  for (i in 1:m){
     for (j in 1:m){
          for (k in 1:n){
if (X[k]<=5*(i-1)/10 && Y[k]<=5*(j-1)/10){
if (X[k]<=C[k] && Y[k]<=C[k]) {M3[i,j]=M3[i,j]+1/C1[k]}
                                     }
                         }

                    }
               }



w=rexp(n,1)
for (i in 1:n){w[i]=max(X[i],Y[i])}

z=rep(0,n);
for (i in 1:n){z[i]=min(w[i],C[i])}

t=w;
Sc=rep(1,n);
nc=rep(0,n);
for (j in 1:n){
for (i in 1:n)   {if (z[i]>=C[j]){nc[j]=nc[j]+1}
                 }
```

```
              }
for (j in 1:n)
{
for (i in 1:n){if (z[i]<=t[j] && C[i]<=w[i]){Sc[j]=Sc[j]*(1-1/nc[i]);}
              }
}

 for (i in 1:m){
     for (j in 1:m){
          for (k in 1:n){
if (X[k]<=5*(i-1)/10 && Y[k]<=5*(j-1)/10){
if (X[k]<=C[k] && Y[k]<=C[k]) {M4[i,j]=M4[i,j]+1/(n*Sc[k]);}
                                         }
                         }

                    }
               }

 #"Projections of n observations,one censoring"
par(mfrow=c(1,1))
 m6=7;
n8=n;
 X1 <- runif(n);  # generate n-dimension of vector  X ~ U(0-1)
 X2 <- runif(n);  # generate n-dimension of vector  Y ~ U(0-1)
X1=X;
X2=Y;
h5=rep(0,n);
 for (i in 1:n){h5[i]=max(X1[i],X2[i]);}
```

```
 Xa1=rep(0,n);   # generate  n-dimension of vector  0
 Xa2=rep(0,n);   # generate  n-dimension of vector  0
S=matrix(1,m6,m6)
 F=matrix(0,m6+1,m6+1)
U=matrix(0,2,n)

 X=rep(0,m6);
 Y=rep(0,m6);
for (i in 1:m6){
  X[i]=5*(i-1)/10;
 Y[i]=5*(i-1)/10;
}
 for (i in 1:m6){    # c1
    for (j in 1:m6){    # c2
        Xa1=rep(0,n);   # generate  n-dimension of vector  0
        Xa2=rep(0,n);   # generate  n-dimension of vector  0
 # "when the v is  over the line y=x"
      if (X[i]<=Y[j]) {         # c3

        for (k in 1:n)  {      # c4

          # "when the observations are  over the line y=x"
          if (X1[k]<=X2[k] ){      # c5
            if(X1[k]<=X[i] && X2[k]<=Y[j])
                {                              # c6
                 {if (X2[k]<=X[i])              # c7
                        {Xa1[k]=X2[k];Xa2[k]=X2[k]}
```

```
                else{Xa1[k]=X[i];Xa2[k]=X2[k]}
         }                                    # c7
        }                                      # c6
# "when the observations are  under the line y=x"
                   }else                       # c5
       { if(X1[k]<=X[i] && X2[k]<=Y[j])        # c8
          {Xa1[k]=X1[k];Xa2[k]=X1[k]}
       }                                       # c8


                 }                             # c4
   ##"Projections of n observations and C"
     for (l in 1:n)  {                    # c9
      if (h5[l]<=C[l]){    #9a
        if (Xa1[l]+Xa2[l]>0){          # c10
            ac=0;
             for (v in 1:n8){if(h5[v]>C[v]){
             if (C[v]<=Y[j] && C[v]<=Xa2[l]){ac=ac+1;}
                                           }
                             }
         for (m in 1:n)  { if(Xa1[m]+Xa2[m]>0 && h5[m]<=C[m])  # c11
         {if (Xa1[m]<=Xa1[l] && Xa2[m]<=Xa2[l])     # c12
                    {ac=ac+1;}
           }                                        # c12


                            }                       # c11
      S[i,j]=(1-1/(n-ac+1))*S[i,j]; # computing the probabilities.
                   }                                 # c10
```

```
                              }                    #9a


                          }                        # c9


                     }                             # c3
    # "when the v is under the line y=x"
       else      {                          # c13
        for (k in 1:n)  {                        # c14
          if  (X1[k]<=X2[k]){                    # c15
            if(X1[k]<=X[i] && X2[k]<=Y[j])
             {Xa1[k]=X2[k];Xa2[k]=X2[k]; }
                          }else                # c15
               { if(X1[k]<=X[i] && X2[k]<=Y[j])    # c16
                  {if  (X1[k]<=Y[j])                 # c17
                       {Xa1[k]=X1[k];Xa2[k]=X1[k]}
                        else{Xa1[k]=X1[k];Xa2[k]=Y[j]}
                 }                                     # c17


               }                                   # c16
                         }                        # c14
            for (l in 1:n)  {                    # c18
                if  (h5[l]<=C[l]){    #9a
                if  (Xa1[l]+Xa2[l]>0){          # c19
                    ac=0;
                       for (v in 1:n8)  {if(h5[v]>C[v]){


                      if  (C[v]<=X[i] && C[v]<=Xa1[l]){ac=ac+1;}
```

```
                                        }
                                                    }
         for (m in 1:n)  { if(Xa1[m]+Xa2[m]>0 && h5[m]<=C[m])  # c20
                  {if (Xa1[m]<=Xa1[l] && Xa2[m]<=Xa2[l])   # c21
                            {ac=ac+1;}
                   }                                         # c21
                                    }                        # c20
                      S[i,j]=(1-1/(n-ac+1))*S[i,j];

                                    }            # c19
                              }    #9a
                             }                   # c18
                          }    # c13
                   }     #c2
                 }    #c1
 F1=1-S
 F2=F1
F3=F1
F2=t(F1)
for (p in 1:m6){
    for (q in 1:m6){
   F[m6-p+1,q+1]=F2[p,q];
}
}
for (p in 1:m6){
F[m6+1-p,1]=Y[p];
F[m6+1,p+1]=X[p];
```

```
}
X3=X1;
X4=X2;
C3=C;
for (i in 1:n){
if (h5[i] <=C[i]){C3[i]=100;}
else{X3[i]=100;X4[i]=100}
              }
F[m6+1,p+1]=X[p];



plot(X3, X4,type='p',cex =0.8, xlim=range(0,3), ylim=range(0, 3),
col="red",sub="  observations, censoring",
main="Values of the path-dependent estimators in each of the enclosed
regions",pch=3, xlab='X',  ylab='Y')

abline(0,1)
 for (v in 1:n){
 points(C3[v], C3[v], type='p', col="blue", xlab='x', ylab='y')
                }
for (p in 1:m6){
    for (q in 1:m6){
F1[p,q]=as.integer((F1[p,q]+0.0001)*100)/100;
text(X[p], Y[q], F1[p,q],cex =.8)
}
}
for (p in 1:n){
```

```
segments(X3[p], X4[p],X3[p],3,col = "grey", lty =3);
segments(X3[p], X4[p],3,X4[p],col = "grey", lty =3);
}
n  # The number of observations
F   # The matrix of  values of the path-dependent estimators
q=q7;

M5=F3
M0=M6;



q=q7;
M=M6;
 for (i in 1:m6){
     for (j in 1:m6){
Mp1[i,j,ip]=M1[i,j];
Mp2[i,j,ip]=M2[i,j];
Mp3[i,j,ip]=M3[i,j];
Mp4[i,j,ip]=M4[i,j];
Mp5[i,j,ip]=M5[i,j];

Ap1[i,j]=M1[i,j]+Ap1[i,j];
Ap2[i,j]=M2[i,j]+Ap2[i,j];
Ap3[i,j]=M3[i,j]+Ap3[i,j];
Ap4[i,j]=M4[i,j]+Ap4[i,j];
Ap5[i,j]=M5[i,j]+Ap5[i,j];
```

```
}
}
M1;
M2;
M3;
M4;
M5;
M6;

}

p=pa;
for (ip in 1:p){
   for (i in 1:m6){
     for (j in 1:m6){
Sp1[i,j]=(Mp1[i,j,ip]-M[i,j])^2+Sp1[i,j];
Sp2[i,j]=(Mp2[i,j,ip]-M[i,j])^2+Sp2[i,j];
Sp3[i,j]=(Mp3[i,j,ip]-M[i,j])^2+Sp3[i,j];
Sp4[i,j]=(Mp4[i,j,ip]-M[i,j])^2+Sp4[i,j];
Sp5[i,j]=(Mp5[i,j,ip]-M[i,j])^2+Sp5[i,j];
}
}
}

for (i in 1:m6){
     for (j in 1:m6){
```

```
Sp1[i,j]=Sp1[i,j]/p;
Sp2[i,j]=Sp2[i,j]/p;
Sp3[i,j]=Sp3[i,j]/p;
Sp4[i,j]=Sp4[i,j]/p;
Sp5[i,j]=Sp5[i,j]/p;

Ap1[i,j]=Ap1[i,j]/p;
Ap2[i,j]=Ap2[i,j]/p;
Ap3[i,j]=Ap3[i,j]/p;
Ap4[i,j]=Ap4[i,j]/p;
Ap5[i,j]=Ap5[i,j]/p;

Bp1[i,j]=(Ap1[i,j]-M[i,j])^2;
Bp2[i,j]=(Ap2[i,j]-M[i,j])^2;
Bp3[i,j]=(Ap3[i,j]-M[i,j])^2;
Bp4[i,j]=(Ap4[i,j]-M[i,j])^2;
Bp5[i,j]=(Ap5[i,j]-M[i,j])^2;



B1p1[i,j]=(Ap1[i,j]-M[i,j])^1;
B1p2[i,j]=(Ap2[i,j]-M[i,j])^1;
B1p3[i,j]=(Ap3[i,j]-M[i,j])^1;
B1p4[i,j]=(Ap4[i,j]-M[i,j])^1;
B1p5[i,j]=(Ap5[i,j]-M[i,j])^1;

}
}
p;  ## The number of repetition
```

```
n;    ## sample size
q;   ## relation between X and Y
a;   ## X's exp. parameter=a
b;    ## Y's exp. parameter=b
ca;   ## C's exp. parameter=ca
M; ## Real true matrix of H(X,Y)
M1; ## Method 1:   no censoring
M2; ## Method 2:   know censoring distribution
M3; ## Method 3:   all Censoring's observed
M4; ## Method 4:   K-M estimator of Fc
M5; ## Method 5:   path-dependent estimator
M6;  ## Real true matrix of H(X,Y)

Ap1;## Method 1:average M1: no censoring
Ap2;## Method 2:average M2: know censoring distribution
Ap3;## Method 3:average M3: all Censoring's observed
Ap4;## Method 4:average M4: K-M estimator of Fc
Ap5;## Method 5:average M5: path-dependent estimator

M6;
Sp1  #### MSE 1
Sp2  #### MSE 2
Sp3  #### MSE 3
Sp4  #### MSE 4
Sp5  #### MSE 5
Bp1; #### Bias1^2
Bp2; #### Bias2^2
Bp3; #### Bias3^2
```

```
Bp4; #### Bias4^2

Bp5; #### Bias5^2

B1p1; #### Bias1^1

B1p2; #### Bias2^1

B1p3; #### Bias3^1

B1p4; #### Bias4^1

B1p5; #### Bias5^1


Sp1-Bp1;

Sp2-Bp2;

Sp3-Bp3;

Sp4-Bp4;

Sp5-Bp5;


MO=round(M,digit=6); ## Real true matrix of H(X,Y)

MA=round(Ap2,digit=6);## Method 2:average M2: know censoring distribution

MB=round(Ap3,digit=6);## Method 3:average M3: all Censoring's observed

MC=round(Ap4,digit=6);## Method 4:average M4: K-M estimator of Fc

MD=round(Ap5,digit=6);## Method 5:average M5: path-dependent estimator


p;  ## The number of repetition

n;    ## sample size

q;   ## relation between X and Y

a;   ## X's exp. parameter=a

b;    ## Y's exp. parameter=b

ca;   ## C's exp. parameter=ca
```

```
MA_MSE=round(Sp2,digit=6);  #### MSE 2
MB_MSE=round(Sp3,digit=6);  #### MSE 3
MC_MSE=round(Sp4,digit=6);  #### MSE 4
MD_MSE=round(Sp5,digit=6);  #### MSE 5
MO;
MA;
MA_MSE;

MB;
MB_MSE;

MC;
MC_MSE;

MD;
MD_MSE;
```

Bibliography

[1] P.K. Andersen. O. Borgan. R. D. Gill, N. Keiding, *Statistical Models Based on Counting Processes*, Springer Series in Statistics, Springer-Verlag, New York, 1993.

[2] Carabarin Aguirre, A. and Ivanoff, B. G., Estimation of a Distribution under Generalized Censoring, *Journal of Multivariate Analysis*,Vol. 101, 2010, 1501-1519.

[3] Carabarin Aguirre, A. and Ivanoff, B. G., Path-dependent Estimation of a Distribution under Generalized Censoring, preprint, 2010.

[4] Hougaard, Philip, *Analysis of Multivariate Survival Data*, Springer, New York, 2000.

[5] Nelsen, R. B. *An Introduction to Copulas*, Lecture Notes in Statistics 139, Springer-Verlag, New York, 1999.