https://doi.org/10.26219/heal.aueb.5743

ATHENS UNIVERSITY OF ECONOMICS AND BUSINESS

DEPARTMENT OF STATISTICS

POSTGRADUATE PROGRAM

A SIMULATION APPROACH TO THE PROBLEM OF MOMENTS

By

Nikolaos G. Vargemidis

A THESIS
Submitted to the Department of Statistics
of the Athens University of Economics and Business
in partial fulfillment of the requirements for
the degree of Master of Science in Statistics

Athens
1998

ISBN : 960-7929-22-5

https://doi.org/10.26219/heal.aueb.5743

ATHENS UNIVERSITY OF ECONOMICS AND BUSINESS

DEPARTMENT OF STATISTICS

A SIMULATION APPROACH TO THE PROBLEM OF MOMENTS

By

Nikolaos G. Vargemidis

A THESIS

Submitted to the Department of Statistics
of the Athens University of Economics and Business
in partial fulfillment of the requirements for
the degree of Master of Science in Statistics

Athens
1998

https://doi.org/10.26219/heal.aueb.5743

ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

ΤΜΗΜΑ ΣΤΑΤΙΣΤΙΚΗΣ

ΠΡΟΣΕΓΓΙΣΗ ΤΟΥ ΠΡΟΒΛΗΜΑΤΟΣ ΤΩΝ ΡΟΠΩΝ ΜΕ ΜΕΘΟΔΟΥΣ ΠΡΟΣΟΜΟΙΩΣΗΣ

Νικόλαος Γ. Βαργεμίδης

ΔΙΑΤΡΙΒΗ

Που υποβλήθηκε στο Τμήμα Στατιστικής
του Οικονομικού Πανεπιστημίου Αθηνών
ως μέρος των απαιτήσεων για την απόκτηση
Μεταπτυχιακού Διπλώματος Ειδίκευσης στη Στατιστική

Αθήνα
1998

https://doi.org/10.26219/heal.aueb.5743

ATHENS UNIVERSITY
OF ECONOMICS AND BUSINESS
DEPARTMENT OF STATISTICS

A Thesis submitted in partial fulfillment of

the requirements for the degree of

Master of Science

A SIMULATION APPROACH TO THE PROBLEM OF MOMENTS

Nikolaos Vargemidis

Supervisor:
Dr. Petros Dellaportas

External examiner:
Professor L. Devroye
School of Computer
Science Office
McGill University
Montreal, Canada

Approved by the Graduate Committee

Professor J. Panaretos
Director of the Graduate Program
March 1999

https://doi.org/10.26219/heal.aueb.5743

DEDICATION

This thesis is dedicated to my family.

ACKNOWLEDGEMENTS

I would like to thank my supervisor, Dr. Petros Dellaportas and the Ph.D. student, D. Karlis for their comments on this work.

I would like also to thank the Associate Professor N.Frangos for providing the data, used in section 5.

https://doi.org/10.26219/heal.aueb.5743

VITA

Name: **Nikolaos Vargemidis**

Year of birth: **1973**

Studies

High School Certificate (1991)

Degree in Statistics - Athens University of Economic and Business (1996)

M.Sc. in Statistics - Athens University of Economics and Business (1998)

The second semester of the M.Sc. studies was elaborated at the postgraduate program of Statistics in the Katholieke Universiteit Leuven, Belgium.

https://doi.org/10.26219/heal.aueb.5743

ABSTRACT

Nikolaos Vargemidis

A SIMULATION APPROACH TO THE PROBLEM OF MOMENTS

1998

We present a sampling-based approach to one version of the problem of moments. In particular, we reconstruct a density function, when its first four moments are available, by simulating from it.

We provide a brief review of the problem of moments and then we examine in detail three algorithms, which provide samples from families of distributions when the first four moments are given. The first algorithm was provided by Devroye (1986) and is based on a mixture representation of a distribution. The second is based on the fact that simulation from the main types of the Pearson's system of distributions is readily available. The third is an attempt to use the Generalized Lambda distribution because generating samples from it is straightforward through the inversion method.

We perform extensive simulations to test the above algorithms and provide useful guidelines for the user who faces such a problem.

https://doi.org/10.26219/heal.aueb.5743

ΠΕΡΙΛΗΨΗ

ΝΙΚΟΛΑΟΣ Γ. ΒΑΡΓΕΜΙΔΗΣ

ΠΡΟΣΕΓΓΙΣΗ ΤΟΥ ΠΡΟΒΛΗΜΑΤΟΣ ΤΩΝ ΡΟΠΩΝ ΜΕ ΜΕΘΟΔΟΥΣ ΠΡΟΣΟΜΟΙΩΣΗΣ

1998

Μία από τις εκδοχές του προβλήματος των ροπών είναι αυτή της εκτίμησης μια συνάρτησης κατανομής δεδομένων των τεσσάρων πρώτων ροπών της. Σε αυτή τη διατριβή εφαρμόζουμε μεθόδους προσομοίωσης για την προσέγγιση του παραπάνω προβλήματος.

Δίνουμε μια σύντομη περιγραφή του προβλήματος των ροπών και έπειτα αναπτύσσουμε τρεις αλγόριθμους οι οποίοι παράγουν τυχαία δείγματα από κάποια κατανομή δεδομένων των τεσσάρων πρώτων ροπών της. Ο πρώτος αλγόριθμος, ο οποίος βασίζεται σε μίξη κατανομών, αναπτύχθηκε από τον Devroye (1986). Ο δεύτερος αλγόριθμος αφορά την παραγωγή τυχαίων δειγμάτων από την οικογένεια κατανομών του Pearson. Ο τρίτος αλγόριθμος αποτελεί μια προσπάθεια χρησιμοποίησης της Γενικευμένης Λάμδα κατανομής για τον παραπάνω σκοπό δεδομένου ότι η προσομοίωση τυχαίων μεταβλητών από αυτήν είναι πολύ εύκολη με την μέθοδο της αντιστροφής.

Τέλος εφαρμόζουμε εκτεταμένες προσομοιώσεις για να ελένξουμε τους παραπάνω αλγόριθμους και δίνουμε χρήσιμες συμβουλές για την εφαρμογή τους.

https://doi.org/10.26219/heal.aueb.5743

TABLE OF CONTENTS

1 Introduction 1
1.1 Definition of Moments 3

2 The Problem of Moments 6
2.1 Introduction to the Problem of Moments 6
2.2 Density Estimation through Sampling 9

3 Simulation From a Distribution Given the First Four Sample Moments 10
3.1 Devroye's Method 10
3.2 Pearson's System of Distributions 17
3.2.1 Introduction 17
3.2.2 Estimation of the Parameters 21
3.2.2.1 Type I 22
3.2.2.2 Type IV 23
3.2.2.3 Type VI 24
3.2.3 Simulation from the Pearson's System 30
3.2.3.1 Simulation from Type I 30
3.2.3.2 Simulation from Type IV 31
3.2.3.3 Simulation from Type VI 33
3.3 Matching the Moments 33
3.4 Simulation Study 35

4 Generating Random Variables Matching the Moments of the Generalized Lambda Distribution 62
4.1 Introduction to the GLD (λ_1, λ_2, λ_3, λ_4) 63
4.2 Estimating the Parameters of the GLD (λ_1, λ_2, λ_3, λ_4) 65
4.3 Simulation Comparison 69

5 A Real Data Application 77
5.1 The Data 77
5.2 Methodology 78
5.3 Results of the Application 82

6 Concluding Remarks 87

References 89

https://doi.org/10.26219/heal.aueb.5743

LIST OF TABLES

Table	Page
Table 3.1: Densities of the Pearson's Family	18
Table 3.2: Choosing the appropriate Member of the Pearson's Family	20
Table 3.3: Quantity to add to the generated random variables	34
Table 3.4: Given Sets of First Four Moments	36
Table 3.5.1: Simulation results with moments $\mu_1 = 0, \mu_2 = 1, \mu_3 = 3, \mu_4 = 17$.	38
Table 3.5.2: Simulation results with moments $\mu_1 = 0, \mu = 1, \mu_3 = 3, \mu_4 = 17$.	39
Table 3.5.3: Simulation results with moments $\mu_1 = 0, \mu = 1, \mu_3 = -3, \mu_4 = 17$.	40
Table 3.5.4: Simulation results with moments $\mu_1 = 0, \mu = 1, \mu_3 = -3, \mu_4 = 17$.	41
Table 3.5.5: Simulation results with moments $\mu_1 = 0, \mu = 1, \mu_3 = 2, \mu_4 = 6$.	42
Table 3.5.6: Simulation results with moments $\mu_1 = 0, \mu = 1, \mu_3 = 2, \mu_4 = 6$.	43
Table 3.5.7: Simulation results with moments $\mu_1 = 0, \mu = 1, \mu_3 = -2, \mu_4 = 6$.	44
Table 3.5.8: Simulation results with moments $\mu_1 = 0, \mu = 1, \mu_3 = -2, \mu_4 = 6$.	45
Table 3.5.9: Simulation results with moments $\mu_1 = 0, \mu = 1, \mu_3 = 1, \mu_4 = 8$.	46
Table 3.5.10: Simulation results with moments $\mu_1 = 0, \mu = 1, \mu_3 = 1, \mu_4 = 8$.	47
Table 3.5.11: Simulation results with moments $\mu_1 = 0, \mu = 1, \mu_3 = -1, \mu_4 = 8$.	48
Table 3.5.12: Simulation results with moments $\mu_1 = 0, \mu = 1, \mu_3 = -1, \mu_4 = 8$.	49
Table 4.1: Parameters and the corresponding moments of the Generalized Lambda distribution	70
Table 4.2.1: Simulation results with moments $\mu_1 = -0.017, \mu_2 = 0.79, \mu_3 = 0.25, \mu_4 = 1.43$	71
Table 4.2.2: Simulation results with moments $\mu_1 = -0.017, \mu_2 = 0.79, \mu_3 = 0.25, \mu_4 = 1.43$	72
Table 4.2.3: Simulation results with moments $\mu_1 = 0.04, \mu_2 = 1.98, \mu_3 = -0.6, \mu_4 = 7.95$	73
Table 4.2.4: Simulation results with moments $\mu_1 = 0.04, \mu_2 = 1.98, \mu_3 = -0.6, \mu_4 = 7.95$.	74
Table 4.2.5: Simulation results with moments $\mu_1 = 0.33, \mu_2 = 1.28, \mu_3 = -0.7, \mu_4 = 4.55$	75
Table 4.2.6: Simulation results with moments $\mu_1 = 0.33, \mu_2 = 1.28, \mu_3 = -0.7, \mu_4 = 4.55$	76
Table 5.1: Distribution of the Number of the Car Accidents	82
Table 5.2: Results of the Real Data Application	84

https://doi.org/10.26219/heal.aueb.5743

LIST OF FIGURES

Figure	Page
Figure 3.1.1: Density Curves of the Type I distribution	26
Figure 3.1.2: Density Curves of the Type I distribution	27
Figure 3.2: Density Curves of the Type IV distribution	28
Figure 3.3: Density Curves of the Type VI distribution	29
Figure 3.4: Illustration of the Rejection Algorithm	33
Figure 3.5.1: Plots of $\hat{\mu}_i - \mu_i, i = 1,2,3,4$, versus Sample Size Given set of moments $\mu_1 = 0, \mu = 1, \mu_3 = 3, \mu_4 = 17$.	50
Figure 3.5.2: Plots of $\hat{\mu}_i - \mu_i, i = 1,2,3,4$, versus Sample Size Given set of moments $\mu_1 = 0, \mu_2 = 1, \mu_3 = -3, \mu_4 = 17$.	51
Figure 3.5.3: Plots of $\hat{\mu}_i - \mu_i, i = 1,2,3,4$, versus Sample Size Given set of moments $\mu_1 = 0, \mu = 1, \mu_3 = 2, \mu_4 = 6$.	52
Figure 3.5.4: Plots of $\hat{\mu}_i - \mu_i, i = 1,2,3,4$, versus Sample Size Given set of moments $\mu_1 = 0, \mu = 1, \mu_3 = -2, \mu_4 = 6$.	53
Figure 3.5.5: Plots of $\hat{\mu}_i - \mu_i, i = 1,2,3,4$, versus Sample Size Given set of moments $\mu_1 = 0, \mu = 1, \mu_3 = 1, \mu_4 = 8$.	54
Figure 3.5.6: Plots of $\hat{\mu}_i - \mu_i, i = 1,2,3,4$, versus Sample Size Given set of moments $\mu_1 = 0, \mu = 1, \mu_3 = -1, \mu_4 = 8$.	55
Figure 3.6.1: Plots of moment's Standard Deviation versus Sample Size Given set of moments $\mu_1 = 0, \mu = 1, \mu_3 = 3, \mu_4 = 17$.	56
Figure 3.6.2: Plots of moment's Standard Deviation versus Sample Size Given set of moments $\mu_1 = 0, \mu = 1, \mu_3 = -3, \mu_4 = 17$.	57
Figure 3.6.3: Plots of moment's Standard Deviation versus Sample Size Given set of moments $\mu_1 = 0, \mu = 1, \mu_3 = 2, \mu_4 = 6$.	58
Figure 3.6.4: Plots of moment's Standard Deviation versus Sample Size Given set of moments $\mu_1 = 0, \mu = 1, \mu_3 = -2, \mu_4 = 6$.	59
Figure 3.6.5: Plots of moment's Standard Deviation versus Sample Size Given set of moments $\mu_1 = 0, \mu = 1, \mu_3 = 1, \mu_4 = 8$.	60
Figure 3.6.6: Plots of moment's Standard Deviation versus Sample Size Given set of moments $\mu_1 = 0, \mu = 1, \mu_3 = -1, \mu_4 = 8$.	61
Figure 5.1: Histogram of the generated values given the moments of the partial claims	85
Figure 5.2: Histogram of the actual partial claims	86

https://doi.org/10.26219/heal.aueb.5743

Chapter 1

Introduction

An aspect of the problem of moments is the reconstruction of a density function given a sequence of its moments. In this thesis we will be concerned with the case of a finite sequence of moments and specifically that of the first four moments. Unfortunately, the reconstruction of a density function is not so simple since it is known that many distributions share the same first few moments. Nevertheless, there are conditions that ensure the existence of at least one distribution matching the given moments of a distribution.

It is well known that there is a dual relationship between the sample and the density function from which it comes, (see Smith and Gelfand (1992)). A density can generate a sample drawn from it, whereas given a sample we can approximate the generating density via a histogram or via a kernel density estimate. Hence, a sensible way to approximate a density is via simulation from a distribution given its first four sample moments.

Here we deal only with the class of unimodal continuous densities. For random variable generation from discrete densities see Devroye (1991). In particular, Devroye developed algorithms for generating discrete random variables, given a probability generating function or a moment sequence. Note that apart from algorithms for generating random variables, given the moments of a density, there are algorithms for generating random variables given the characteristic function of a

https://doi.org/10.26219/heal.aueb.5743

distribution or given the Fourier coefficients (Devroye, 1986,1989). Another possible approach is that of generating random variables from a distribution matching the quantiles of a distribution.

The method of moments has been criticized for the high standard deviation of the moments up to the fourth (Pearson et al., 1979). For this reason, in this thesis, we focus on algorithms, which use only the first four moments.

The rest of the thesis is organized as follows. In chapter 2 we present the main aspects of the problem of moments and the sufficient conditions for the existence of a solution to the problem of moments. In chapter 3 we develop two algorithms for simulating random samples given the first four simple moments of a distribution. The first algorithm concerns the generation of random variables in the real line and is based on the algorithm given by Devroye (1986). The second generates random variables from the appropriate member of the Pearson's system of distributions. We apply both algorithms to given sets of moments and hence report simulation results and conclusions.

In chapter 4, we introduce the Generalized Lambda distribution and then we develop an algorithm for generating random variables matching its first four moments. Furthermore, we apply the algorithms developed in chapter 3 and chapter 4 to the same sets of moments and report the simulation results.

In the last chapter we apply the algorithms of chapter 3 to a real data set from the field of car insurance. Our objective is to approximate the distribution of the individual claims, given the total claims.

https://doi.org/10.26219/heal.aueb.5743

1.1 Definition of Moments

To facilitate the reader we give the definitions of the main terms used in this thesis.

The $n-th$, $n \geq 1$, simple moment of a random variable X, $X \in A$, with distribution function $F(X)$, is defined as

$$\mu_n = E(X^n) = \int_A x^n dF(x),$$

under the condition that $\int_A x^n dF(x) < \infty$.

It is clear that for $n=0$ we obtain $\mu_0=1$. The first moment is the mean or the expectation of a random variable X and is usually denoted by μ. However, in this thesis we denote the first moment by μ_1.

The $n-th, n \geq 1$, central moment of a random variable X, where $X \in A$, with distribution function $F(X)$, is defined as

$$\mu'_n = E(X-\mu_1)^n = \int_A (x-\mu_1)^n dF(x),$$

under the condition that $\int_A (x-\mu_1)^n dF(x) < \infty$.

It is obvious from the definition of the central moments that the first central moment of a random variable X is always zero. The variance of a random variable X is the second central moment

$$\sigma^2 = \mu'_2 = \int_A (x-\mu_1)^2 dF(x),$$

https://doi.org/10.26219/heal.aueb.5743

which in terms of the simple moments can be expressed as $\sigma^2 = \mu_2 - \mu_1^2$. The standard deviation of a random variable X, denoted by σ, is the square root of the variance, i.e. $\sigma = \sqrt{\sigma^2} = \sqrt{\mu_2 - \mu_1^2}$.

Let X be a random variable and r a positive integer. We define the $r-th$ factorial moment of the random variable X as:

$$\mu_{(r)} = E(X_{(r)}) = E\big(X(X-1)\cdots(X-r+1)\big). \qquad (1.1)$$

Relation (1.1) enables us, to compute the corresponding r simple moments, given the first r factorial moments of a random variable X. For the first factorial and simple moment that $\mu_1=\mu_{(1)}$.

The third central moment μ_3' and especially the quantity, $\frac{\mu_3'}{\sigma^3}$, known as coefficient of skewness, is a measure of the skewness of the distribution of a random variable X. When the third moment μ_3' or equivalently the coefficient of skewness is negative or positive then the distribution of the random variable X is negatively or positively skewed respectively. Additionally, when the third moment μ_3' or equivalently the coefficient of skewness is equal to zero then the distribution of the random variable X is symmetric with respect to the mean μ_1. Moreover, the fourth central moment μ_4' and especially the quantity, $\frac{\mu_4'}{\sigma^4}$, known as coefficient of kurtosis, is a measure of how steep is the distribution of a random variable X. However, in this thesis we define the coefficient of skewness and kurtosis as

https://doi.org/10.26219/heal.aueb.5743

$$\beta_1 = \frac{\mu_3'^2}{\mu_2'^3} \text{ and } \beta_2 = \frac{\mu_4'}{\mu_2'^2} \text{ respectively.} \tag{1.2}$$

Given the simple moments of a random variable X we can compute the corresponding central moments of the random variable X through the following recursion formulae

$$\mu_n' = \mu_n - \left(\frac{n!}{1!(n-1)!}\right)\mu_{n-1}\mu_1 + \left(\frac{n!}{2!(n-2)!}\right)\mu_{n-2}\mu_1^2 - \ldots + (-1)\mu_1^n. \tag{1.3}$$

Hence, from formulae (1.3) we obtain for the first four central moments

$$\begin{aligned}
\mu_1' &= 0 \\
\mu_2' &= \mu_2 - \mu_1^2 \\
\mu_3' &= \mu_3 - 3\mu_2\mu_1 + 2\mu_1^3 \\
\mu_4' &= \mu_4 - 4\mu_3\mu_1 + 6\mu_2\mu_1^2 - 3\mu_1^4.
\end{aligned}$$

Given a sample $X_1 \ldots X_N$, of sample size N, the corresponding $n-th$, $n \geq 1$, sample simple and central moment is defined as $\hat{\mu}_n = \frac{\sum_{i=1}^{N} X_i^n}{N}$ and $\hat{\mu}_n' = \frac{\sum_{i=1}^{N}(X_i - \hat{\mu}_1)^n}{N}$ respectively. The $n-th, n \geq 1$, sample simple and central moment is the estimate of the corresponding $n-th, n \geq 1$, simple and central moment of a random variable X. The first sample simple moment, denoted as $\hat{\mu}_1$ or $\overline{x}$, is the sample estimate of the mean of a random variable X while the second sample central moment $\hat{\mu}_2' = \frac{\sum_{i=1}^{N}(X_i - \overline{x})^2}{N}$, denoted as S^2, is the sample estimate of the variance of a random variable X.

https://doi.org/10.26219/heal.aueb.5743

Chapter 2

The Problem of Moments

In this chapter we give a brief review of the main aspects of the problem of moments, which was introduced by Stieltjes in 1894-95. In its simplest form the problem of moments can be formulated as follows: Given a collection of moments $\mu_1,\ldots,\mu_n$, $n \geq 0$, first determine whether there exists a distribution, which matches these moments, second try to construct such a distribution and determine whether it is unique or not. For more details about the problem of moments see Shohat and Tamarkin (1943).

2.1 Introduction to the Problem of Moments

Stieltjes proposed the following problem: Find a bounded non-decreasing function $F(x)$, where $F(x)$ could be seen as a distribution function, in the interval $[0,\infty)$ such that its "moments" $\int x^k dF(x)$, $k = 1,\ldots,n$, have a prescribed set of values $\mu_1,\ldots,\mu_n$, $n \geq 1$.

He further showed that a necessary and sufficient condition for the existence of a solution to the problem of moments, in terms of a given sequence of moments $\mu_1,\ldots,\mu_n$, $n \geq 1$, is the positiveness of the following determinants:

https://doi.org/10.26219/heal.aueb.5743

$$D_n = \begin{vmatrix} 1 & \mu_1 & \cdots & \mu_n \\ \mu_1 & \mu_2 & \cdots & \mu_{n+1} \\ \cdot & & & \\ \cdot & & & \\ \cdot & & & \\ \mu_n & \mu_{n+1} & \cdots & \mu_{2n} \end{vmatrix} \geq 0 \qquad (2.1)$$

$$D'_n = \begin{vmatrix} \mu_1 & \mu_2 & \cdots & \mu_{n+1} \\ \mu_2 & \mu_3 & \cdots & \mu_{n+2} \\ \cdot & & & \\ \cdot & & & \\ \cdot & & & \\ \mu_{n+1} & \mu_{n+2} & \cdots & \mu_{2n+1} \end{vmatrix} \geq 0 \qquad (2.2)$$

for all $n \geq 1$.

An extension of the Stieltjes's moment problem is the Hamburger moment problem, where the distribution function $F(x)$, of a random variable X, is defined in the real line R. In the case of the Hamburger problem of moments a necessary and sufficient condition for the existence of a solution is the positiveness of the determinant (2.1) for all $n \geq 1$. Thus, given a sequence of moments $\mu_1,...,\mu_n$, if condition (2.1) holds then there exists at least one distribution, which matches the given moments.

The next theorem (see Springer, 1979) gives the necessary condition for the uniqueness of the solution to the moment problem.

Theorem 1: A set of moments $\mu_1,...,\mu_n$, uniquely determines a distribution if and only if:

a) when the range of the distribution is $(-\infty,\infty)$, it holds that

https://doi.org/10.26219/heal.aueb.5743

$$\sum_{i=0}^{\infty} |\mu_{2i}|^{\frac{-1}{2i}} = \infty. \tag{2.3}$$

b) when the range of the distribution is $[0,\infty)$, it holds that

$$\sum_{i=0}^{\infty} (\mu_i)^{\frac{-1}{2i}} = \infty .$$

Relation (2.3) is known as the Carleman's condition.

Two other extensions of the problem of moments is the Hausdorff one-dimensional and the trigonometric problem of moments respectively. The first corresponds to the case where the distribution function $F(x)$, of a random variable X, is defined in the closed interval $[0,1]$. The Hausdorff problem of moments can be expanded in two dimensions, where the distribution function $F(u,\upsilon)$ is defined in a rectangle in the (u, υ)-plane, which without loss of generality may be taken as the unit square, i.e $0 \leq u \leq 1$, $0 \leq \upsilon \leq 1$. The trigonometric problem of moments corresponds to the case where the distribution function $F(x)$ is defined in the circumference of a circle which, without loss of generality, can be taken as the unit circle. For information the appropriate conditions for the solution of the Hausdorff and the trigonometric problem of moments see Shohat and Tamarkin (1943).

https://doi.org/10.26219/heal.aueb.5743

2.2 Density Estimation through Sampling

An aspect of the problem of moments is the random variable generation given a sequence of moments. The most important case is that of a finite sequence of moments. On one hand, researchers are interested in obtaining random samples which match the first few given moments since many distributions have only a finite number of moments. On the other hand, we could say that there are more than one distributions which share the first n moments (Devroye, 1986).

Our approach here is sample based in the sense that we are interested in generating random samples matching the given moments by making only broad assumptions about the underlying functional form of the distribution.

Let us assume that the given moments are $\mu_1 = 0, \mu_2 = 1, \mu_3, \mu_4$. Condition (2.1) implies that

$$\begin{vmatrix} 1 & 0 & 1 \\ 0 & 1 & \mu_3 \\ 1 & \mu_3 & \mu_4 \end{vmatrix} \geq 0 \Rightarrow \mu_4 \geq \mu_3^2 + 1. \qquad (2.4)$$

Inequality (2.4) is very important since it provides a sufficient condition for the moments $\mu_1 = 0, \mu_2 = 1, \mu_3, \mu_4$ to correspond to a distribution function. Hence, assuming the given moments satisfy (2.4) then there is at least one distribution having these moments. However, it is obvious that condition (2.4) is restrictive and usually does not hold for small sample sizes.

https://doi.org/10.26219/heal.aueb.5743

Chapter 3

Simulation from a Distribution Given the First Four Moments

In this chapter we describe two algorithms for generating random variables from a unimodal continuous distribution given the first four simple moments. Without loss of generality we deal with the standardized moments, i.e. $\mu_1 = 0, \mu_2 = 1, \mu_3, \mu_4$. The first algorithm is a modification of the Devroye's algorithm (Devroye, 1986) while the second algorithm generates random samples from the appropriate member of the Pearson's system of distributions. For each algorithm we present the theoretical background and its performance to prespecified sets of first four sample moments.

3.1 Devroye's Method

Let us assume that we want to simulate from a distribution with given its first four moments $\mu_1, \mu_2, \mu_3, \mu_4$. However, we know that if $\mu_1, \mu_2, \mu_3, \mu_4$ are the simple moments of a random variable X then the first four moments of the standardized random variable $\frac{X - \mu_1}{\sigma}$, where $\sigma = \sqrt{\mu_2 - \mu_1^2}$ is the standard deviation, are 0, 1, μ_3, μ_4. Hence, it is equivalent to simulate from a distribution with moments 0, 1, μ_3, μ_4. Moreover, we assume that the distribution, f, from which we want to

https://doi.org/10.26219/heal.aueb.5743

simulate, is a mixture of two distributions f_1 and f with mixing parameter p, i.e. $f(x) = pf_1(x) + (1-p)f_2(x)$. Then, the simple moments of the distribution f, will be

$$\mu_i = p\mu_{1i} + (1-p)\mu_{2i},\ i = 1,2...,\qquad (3.1)$$

where μ_{1i} and μ_{2i} are the simple moments of f_1 and f_2 respectively (Titterington et al, 1985).

If we set the moments of the distributions f_1 and f_2 respectively as

$$\mu_{11} = 0,\ \ \mu_{12} = 1,\ \ \mu_{13} = \sqrt{\mu_4 - 1},\ \ \mu_{14} = \mu_4 \qquad (3.2)$$

$$\mu_{21} = 0,\ \ \mu_{22} = 1,\ \ \mu_{23} = -\sqrt{\mu_4 - 1},\ \ \mu_{24} = \mu_4, \qquad (3.3)$$

then by solving the equation $\mu_3 = p\mu_{13} + (1-p)\mu_{23}$ with respect to p we find that the mixing parameter is given by the formula $p = \dfrac{1 + \dfrac{\mu_3}{\sqrt{\mu_4 - 1}}}{2}$.

Note that relations (3.2) and (3.3) imply that $\mu > 1$. However, if the condition $\mu_4 > \mu_3^2 + 1$ is satisfied then $\mu > 1$ always stands.

From relation (3.1) we can say that the moments of the distribution f are convex combinations of the moments, of distributions f_1 and f_2, given by relations (3.2) and (3.3).

The next step is to specify the distributions f_1 and f_2. If X is a Bernoulli random variable with parameter q, i.e. $X \sim$ Bernoulli (q), then the standardized Bernoulli random variable $\dfrac{X-q}{\sqrt{q(1-q)}}$ has moments

https://doi.org/10.26219/heal.aueb.5743

$\mu_1=0,\ \mu_2=1,\ \mu_3=\dfrac{1-2q}{\sqrt{q(1-q)}},\ \mu_4=\dfrac{3q^2-3q+1}{q(1-q)}$. Note that relations (3.2) and (3.3) imply the condition

$$\mu_4=\mu_3^{\ 2}+1. \tag{3.4}$$

The moments of the standardized Bernoulli random variable satisfy the condition (3.4) since $\left(\dfrac{1-2q}{\sqrt{q(1-q)}}\right)^2+1=\dfrac{3q^2-3q+1}{q(1-q)}$. Therefore, a good choice for f_1 and f_2 distributions is a Bernoulli (q) distribution.

The question that arises now is the following. What is the value of the probability q for the f_1 and f_2 Bernoulli distributions? By equating the third moment of the f_1 distribution, $\mu_{13}=\sqrt{\mu_4-1}$, with the third moment of the standardized Bernoulli random variable, $\dfrac{1-2q}{\sqrt{q(1-q)}}$, then for the f_1, f_2 distributions we respectivalely find that:

$$q=q_{f_1}=\frac{1+\sqrt{\dfrac{\mu_4-1}{\mu_4+3}}}{2} \quad \text{and} \quad q_{f_2}=\frac{1-\sqrt{\dfrac{\mu_4-1}{\mu_4+3}}}{2}=1-q_{f_1}=1-q \ .$$

Hence, the standardized moments of the variable X are a mixture of the moments of two standardized Bernoulli random variables, with Bernoulli distributions f_1, f_2 and parameters q, $(1-q)$ respectively.

https://doi.org/10.26219/heal.aueb.5743

Therefore, the algorithm is given as:

First Devroye's Algorithm

- Give the simple first four moments μ_1 , μ_2, μ_3 , μ_4
- Standardize the given moments by setting :

$$\mu_1 \to 0$$

$$\sigma \to \sqrt{\mu_2 - \mu_1^2},$$

$$\mu_3 \to \frac{\mu_3 - 3\mu_2\mu_1 + 2\mu_1^3}{\sigma^3},$$

$$\mu_4 \to \frac{\mu_4 - 4\mu_3\mu_1 + 6\mu_2\mu_1^2 - 3\mu_1^4}{\sigma^4}.$$

- Set $p = \dfrac{1 + \dfrac{\mu_3}{\sqrt{\mu_4 - 1}}}{2}$ and $q = \dfrac{1 + \sqrt{\dfrac{\mu_4 - 1}{\mu_4 + 3}}}{2}$
- Generate a random variable U from Uniform(0,1)

  if $U \le p$ then generate a r. v. X from Bernoulli($1-q$)

  else

  if $U > p$ then generate a r. v. X from Bernoulli(q)
- Set $X = X\sigma + \mu_1$.

The resulting random variable X takes the following four values:

- $X = \mu_1 + \sigma\dfrac{1-q}{\sqrt{q(1-q)}}$ with probability pq
- $X = \mu_1 - \sigma\dfrac{q}{\sqrt{q(1-q)}}$ with probability $p(1-q)$
- $X = \mu_1 + \sigma\dfrac{q}{\sqrt{q(1-q)}}$ with probability $(1-p)(1-q)$
- $X = \mu_1 - \sigma\dfrac{1-q}{\sqrt{q(1-q)}}$ with probability $(1-p)q$.

https://doi.org/10.26219/heal.aueb.5743

At this point, the moments of the generated random variable X are those of the standardized moments of the second step of the algorithm. However, in order to obtain a continuous random variable from a unimodal distribution one more step is needed.

By the term unimodal distribution on the real line, R, one usually refers to a density f which has a maximum at a unique point ν and the random variable X decreases as it goes away from ν in either direction. In this sense the Normal and the Cauchy distributions are unimodal. For the distributions whose support is only a part of the real line, like the Beta and Gamma distribution, the random variable X must only be just nonincreasing as it goes away from the mode ν (Dharmadhikari and Joag-dev, 1987). Below we give a formal definition for unimodality in terms of distribution functions (see Dharmadhikari and Joag-dev, 1987).

Definition: A real random variable X or its distribution function F is called unimodal about a mode ν, if F is convex on $(-\infty, \nu)$ and concave on $(\nu, +\infty)$.

The following theorem (see in Dharmadhikari and Joag-dev, 1987) enables us to generate random variables from a unimodal distribution.

Theorem 2: A distribution function F on R is unimodal if, and only if, there exist independent random variable U and Z such that U is Uniform on $(0,1)$ and the product UZ has distribution function F.

Therefore, by setting $Y = UZ$ then the random variable Y has moments given by

$$\mu_i = E(Y^i) = E\left[(UZ)^i\right] = E(U^i)E(Z^i) = \frac{1}{i+1}E(Z^i) = \frac{1}{i+1}\nu_i, \qquad (3.5)$$

https://doi.org/10.26219/heal.aueb.5743

where $E(U^i)=\frac{1}{i+1}$ is the $i-th$ simple moment of a Uniform random variable and v_i denotes the $i-th$ simple moment of the random variable Z.

Let us substitute the random variable Z with X in (3.5) and set the first four moments of the variable X as $v_i=(i+1)\mu_i, i=1,2,3,4$. Then from relation (3.5) we obtain for the first four simple moments of the product $Y=UX$

$$\mu_i=E(Y^i)=E\left[(UX)^i\right]=E(U^i)E(X^i)=\frac{1}{i+1}E(X^i)=\frac{1}{i+1}(i+1)\mu_i=\mu_i.$$

Hence, the product $Y=UX$ has the required moments $\mu_1, \mu_2, \mu_3, \mu_4$.

Summing up, if $\mu_1, \mu_2, \mu_3, \mu_4$ are the given moments of a unimodal continuous distribution from which we want to draw a random sample then

Second Devroye's Algorithm

- generate with the above algorithm, First Devroye's algorithm, a random variable X with moments $v_1=2\mu_1$, $v_2=3\mu_2$, $v_3=4\mu_3$, $v_4=5\mu_4$,
- generate a uniform on $(0,1)$ random variable U,
- set $Y=UX$,
- the resulting random variable Y has the required given moments $\mu_1, \mu_2, \mu_3, \mu_4$.

A disadvantage of the algorithm is that by setting $v_1=2\mu_1$, $v_2=3\mu_2$ we may obtain negative variance. We know that the variance of a random variable, $\mu_2-\mu_1^2$, is always positive. Unfortunately, this does not always

https://doi.org/10.26219/heal.aueb.5743

hold for the quantity $3\hat{\mu}_2 - (2\hat{\mu}_1)^2$, which may be negative. In such a case, it is obvious that the Devroye's algorithm can not be implemented.

Another disadvantage of the algorithm is that if we want to simulate a sample from a distribution in the range $[0,\infty)$ we may get negative values. According to the Devroye's algorithm, we take a mixture of two standardized Bernoulli variables, with mixing parameter p, such that the moments of the two standardized Bernoulli variables satisfy condition (3.4). In order to take values in the range $[0,\infty)$ one may suggest replacing the mixture of standardized Bernoulli random variables with the mixture of standardized Exponential random variables which are defined in the range $[0,\infty)$. However, in such a case the condition (3.4) is not satisfied by the moments of the standardized Exponential variables and the Devroye's algorithm can not be implemented.

We must clarify that the algorithm can not be implemented when condition (2.4) is violated by the given moments. For this reason, we are interested in an algorithm that does not take into account the inequality (2.4). In the next section we try to overcome restriction (2.4) by developing an algorithm for generating random variables from the appropriate member of the Pearson's system of distributions.

https://doi.org/10.26219/heal.aueb.5743

3.2 Pearson's System of Distributions

3.2.1 Introduction

Pearson originated this system at the end of the 19th century when it was observed that many data sets were not adequately represented by the Normal distribution. The system of the Pearson's densities encompasses many well-known distributions and it was widely used for fitting distribution curves to data sets. In particular, the Pearson's system consists of 12 different types. In table 3.1 we give a complete list of all the distributions of the system along with their parameters and their support (Devroye, 1986). For details about the Pearson's system of distributions see Johnson and Kotz (1970), Kendall and Stuart (1961), Elderton and Johnson (1969). The Pearson's system of distributions is suitable for moment matching since, as we will show below, it does not require more than the first four moments for the parameters to be estimated.

Each density of the system satisfies the following differential equation

$$\frac{df}{dx} = \frac{(x-a)f(x)}{c_0 + c_1 x + c_2 x^2}. \qquad (3.6)$$

The shape of the probability density function curve varies with respect to the values of a, c_0, c_1, c_2.

https://doi.org/10.26219/heal.aueb.5743

Table 3.1

Densities of the Pearson's Family

Type	$f(x)$	Parameters	Support
I	$K(1+\frac{x}{a})^b(1-\frac{x}{a})^d$	$b,d > -1$; $a,c > 0$	$[-a,c]$
II	$K(1-(\frac{x}{a})^2)^b$	$b > -1$; $a > 0$	$[-a,a]$
III	$K(1+\frac{x}{a})^{ba}\exp(-bx)$	$ba > -1$; $b > 0$	$[-a,\infty)$
IV	$K(1+(\frac{x}{a})^2)^{-b}\exp(-c\arctan(\frac{x}{a}))$	$a > 0 ; b > \frac{1}{2}$	$(-\infty,+\infty)$
V	$Kx^{-b}\exp(-\frac{c}{x})$	$b > 1$; $c > 0$	$[0,\infty)$
VI	$K(x-a)^b x^{-c}$	$c > b+1 > 0$; $a > 0$	$[a,\infty)$
VII	$K(1+(\frac{x}{a})^2)^{-b}$	$b > \frac{1}{2}$; $a > 0$	
VIII	$K(1+\frac{x}{a})^{-b}$	$0 \leq b \geq 1$; $a > 0$	$[-a,0]$
IX	$K(1+\frac{x}{a})^b$	$b > 0$; $a > 0$	$[-a,0]$
X	$\frac{1}{a}\exp(-\frac{x}{a})$	$a > 0$	$[0,\infty)$
XI	$K(\frac{a}{x})^b$	$a > 0$; $b > 1$	$[a,\infty)$
XII	$K(\frac{a+x}{b-x})^c$	$0 < b < a$; $0 \leq c < 1$	$[-a,b]$

Note: K is a normalizing constant.

Equation (3.6) can be written as, $(c_o + c_1 x + c_2 x^2)df = (x-a)f(x)dx$. If we multiply each part by x^n, we obtain,

$$x^n(c_o + c_1 x + c_2 x^2)\frac{df}{dx}dx = x^n(x-a)f(x)dx \ . \tag{3.7}$$

By integrating the left-hand side of (3.7) by parts over the range of the distribution and assuming that the integrals exist, we find

$$\left[x^n(c_0 + c_1 x + c_2 x^2)f(x)\right]_{-\infty}^{+\infty} - \int_{-\infty}^{+\infty}\left[nc_0 x^{n-1} + (n+1)c_1 x^n + (n+2)c_2 x^{n+1}\right]f(x)dx =$$

$$\int_{-\infty}^{+\infty} x^{n+1} f(x)dx - a\int_{-\infty}^{+\infty} x^n f(x)dx. \tag{3.8}$$

Let us assume that the expression $\left[x^n(c_0 + c_1 x + c_2 x^2)f(x)\right]_{-\infty}^{+\infty}$, vanishes at the extremities of the distribution, i.e. $\lim_{x\to\pm\infty} x^{n+2} f \to 0$. Then by substituting moments for integrals in relation (3.8) we obtain

$$-nc_0\mu_{n-1} - (n+1)c_1\mu_n - (n+2)c_2\mu_{n+1} = \mu_{n+1} - a\mu_n ,$$

or equivalently

$$nc_0\mu_{n-1} + \{(n+1)c_1 - a\}\mu_n + \{(n+2)c_2 + 1\}\mu_{n+1} = 0 \ . \tag{3.9}$$

It is clear from (3.9) that all moments can be expressed in terms of the parameters, a, c_0, c_1, c_2. Conversely, we can express the four constants, a, c_0, c_1, c_2, in terms of the moments. Note also that relation (3.9) can be expressed in terms of the central moments. Practically, relation (3.9) enables us, given the first four moments, to estimate the parameters of the appropriate member of the Pearson's system.

The choice of the appropriate member of the Pearson's family is of a particular interest. The criterion for choosing the appropriate member of

https://doi.org/10.26219/heal.aueb.5743

the family is based on the value of the κ - coefficient, which is a function of the coefficients of skewness and kurtosis respectively.

Hence, if μ'_n, n=1,2,3,4 is the $n-th$ central moment and β_1, β_2 is the coefficient of skewness and kurtosis respectively, defined as in formulae (1.2), we define the κ - coefficient as

$$\kappa = \frac{\beta_1(\beta_2+3)^2}{4(4\beta_2-3\beta_1)(2\beta_2-3\beta_1-6)}.$$

Table 3.2 gives the values of the κ - coefficient and the corresponding distribution of the family.

Table 3.2

Choosing the Appropriate Member of the Pearson's Family

Values of κ - coefficient	**Appropriate distribution**
$\kappa=-\infty$	Type III
$\kappa<0$	Type I
$\kappa=0$	Type II , when $\beta_1=0,\ \beta_2<3$ Type VII , when $\beta_1=0,\ \beta_2>3$ Normal Curve , when $\beta_1=0,\ \beta_2=3$
$0<\kappa<1$	Type IV
$\kappa=1$	Type V
$\kappa>1$	Type VI
$\kappa=\infty$	Type III

In this thesis we develop an algorithm for generating random variables from three main Types of the family which are, Type I, Type IV and Type VI. These three main types cover all the possible ranges to which a

https://doi.org/10.26219/heal.aueb.5743

distribution can practically be defined. Particularly, Type I is of limited range in both sides, Type IV is of unlimited range in both sides and finally Type VI is of limited range in one side. As we can see from Table 3.2, Type I corresponds to the negative values of κ - coefficient, Type IV corresponds to values of κ between zero and one while Type VI to values of $\kappa > 1$.

Moreover, in real data sets it is rather rare to obtain values, for the κ - coefficient, which correspond to remaining types of the family. For instance, with real data it is practically impossible with real data to obtain value, $\kappa = 0$, $\mu_3 = 0$, which lead to Type II or Type VII. Types VIII, IX, X, XI are very uncommon in practice and their density curves in the limits are of horizontal line or trapezoidal. Type VIII is a special case of Type I with $d = 0$. Finally, the criterion of κ - coefficient is not suitable for the above types VIII, IX, X, XI (see Elderton and Johnson, 1969).

3.2.2 Estimation of the Parameters

The first step is to identify the appropriate member of the Pearson's family, given the first four central moments. In the section 3.2.1 we have shown that we can express the constants of the differential equation in terms of the moments μ_1, μ_2, μ_3, μ_4. The next step is to estimate the parameters of the distributions.

In order to simplify the calculations we define the next two quantities:

$$r = \frac{6(\beta_2 - \beta_1 - 1)}{6 + 3\beta_1 - 2\beta_2},$$

$$t = \beta_1 (r+2)^2 + 16(r+1).$$

For details about the estimation of the parameters see Cooper et al (1965).

https://doi.org/10.26219/heal.aueb.5743

3.2.2.1 TYPE I

The probability density of the Type I distribution is given by the function:

$$f(x) = K\left(1+\frac{x}{a}\right)^{b}\left(1-\frac{x}{c}\right)^{d},$$

where $x \in [-a, c]$, $b,d > -1$, $a,c > 0$ are the parameters of the Type I distribution and K is the normalization constant.

The values of the parameters can be computed by the following equations

$$b = \frac{1}{2}\left[r - 2 + r(r+2)\sqrt{\frac{\beta_1}{t}}\right],$$

$$d = \frac{1}{2}\left[r - 2 - r(r+2)\sqrt{\frac{\beta_1}{t}}\right],$$

$$a + c = \frac{1}{2}\sqrt{\mu_2 t}, \qquad (3.10)$$

$$\frac{b}{a} = \frac{d}{c}. \qquad (3.11)$$

It is rather simple to solve the system of equations (3.10) and (3.11) for a and c respectively. Quantity t, defined in section 3.2.2, must always be positive for Type I distribution. When μ_3 is positive then the values of the parameters b,d must be interchanged. The density curve of Type I is usually bell shaped with various degrees of skewness but it can be J-shaped and U-shaped. If $b,d > 0$ then the curve is bell shaped. Particularly, if b,d are approximately equal then the density curve is nearly symmetrical; if b, d are not small it tails off at both ends, and if

https://doi.org/10.26219/heal.aueb.5743

b, d are small it rises abrubtly at both ends. If $b<0$ and $d>0$ then the density curve of the Type I is J-shaped; it starts at an infinite ordinate, falls rapidly and runs out at $x=c$. Conversely, if $b>0$ and $d<0$ we have a reversed J-shaped curve. Finally if both b and d are negative the density curve is U-shaped, starting and ending with infinite ordinates and having an anti-mode. In figures 3.1.1 and 3.1.2 we illustrate the possible density curve shapes of the Type I distribution.

3.2.2.2 TYPE IV

The probability density function of the TYPE IV distribution is given by the function:

$$f(x) = K\left[1+\left(\frac{x}{a}\right)^2\right]^{-b} \exp\left[-c\arctan\left(\frac{x}{a}\right)\right],$$

where $x \in (-\infty,+\infty)$, $a>0$, $b>\frac{1}{2}$ are the parameters of the distribution and K is the normalization constant.

The values of the parameters can be calculated by the following equations

$$b = \frac{2-r}{2},$$

$$c = \frac{-r(r+2)\sqrt{\beta_1}}{\sqrt{-t}},$$

$$a = \frac{1}{4}\sqrt{\mu_2}\sqrt{-t}.$$

Note that for this distribution the quantities r and t , defined in section 3.2.1, must be greater than 3 and negative respectively. The Type IV

https://doi.org/10.26219/heal.aueb.5743

distribution is unimodal and its first four moments exist if $b > 2.5$. Figure 3.2 illustrates the density curve shape of the Type IV distribution.

3.2.2.3 TYPE VI

The probability density function of the Type VI distribution is given by the function:

$$f(x) = K(x-a)^b x^{-c},$$

where $x \in [a, +\infty)$, $c > b+1>0$, $a>0$ are the parameters and K is the normalization constant. However, when the third sample moment is negative then $a<0$ and the range of the distribution is $(-\infty, -a]$.

The parameters of the distribution can be computed by the following equations

$$a = \frac{1}{2}\sqrt{\mu_2}\sqrt{t},$$

$$b = \frac{1}{2}\left[r - 2 + r(r+2)\sqrt{\frac{\beta_1}{t}}\right],$$

$$-c = \frac{1}{2}\left[r - 2 - r(r+2)\sqrt{\frac{\beta_1}{t}}\right]$$

The density curve of the Type VI distribution, is usually skewed and bell shaped or J-shaped when the parameter b is negative. Moreover, quantity r must always be negative. In Figure 3.3, we illustrate the two possible curve shapes of Type VI distribution.

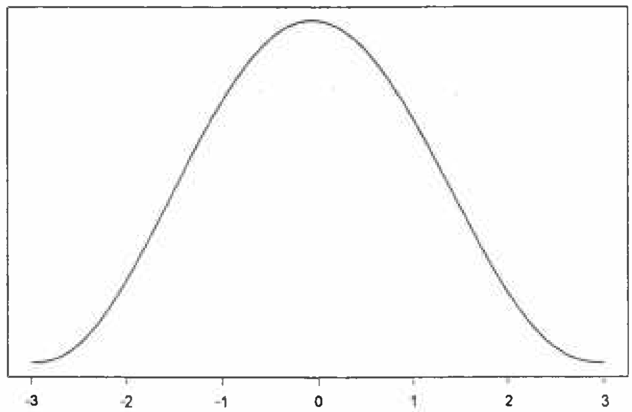
https://doi.org/10.26219/heal.aueb.5743

Moreover, we define the quantities $a_1 = \frac{a(c-1)}{(c-1)-(b+1)}$ and $a_2 = \frac{a(b+1)}{(c-1)-(b+1)}$, where $a = a_1 - a_2$. When a_2 is positive, our algorithm generates values from Type VI, in the interval $[-a_2, +\infty)$ whereas when a_2 is negative it generates values from Type VI in the interval $(-\infty, -a_2]$. Hence, by adding the quantity a_1, to the generated values the range becomes $[a, +\infty)$ and $(-\infty, -a]$ respectively.

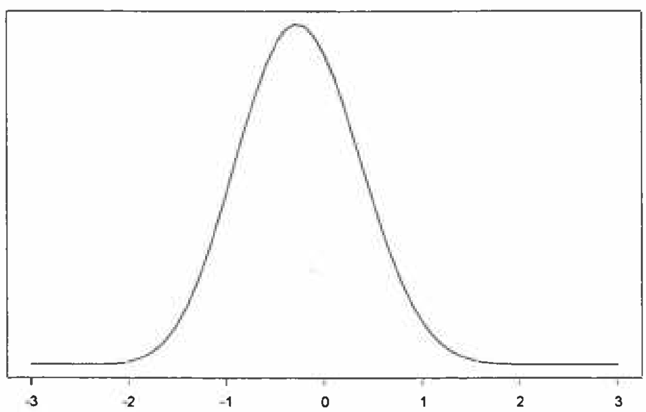
https://doi.org/10.26219/heal.aueb.5743

Figure 3.1.1
Density Curves of the Type I Distribution

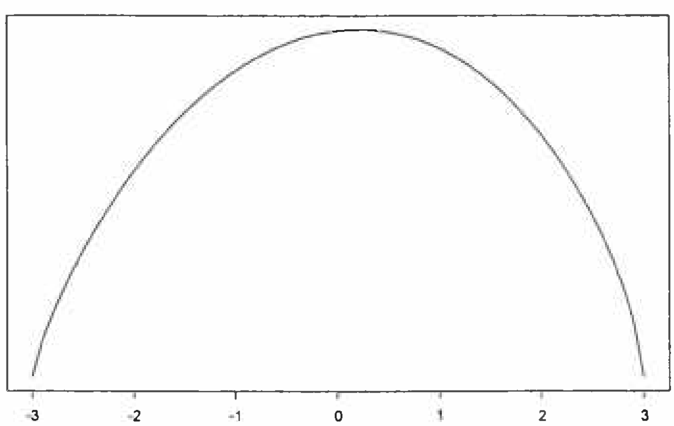
Density Curve of Type I Distribution:Bell-Shaped,b=2.5,d=2.6



Density Curve of Type I Distribution:Bell-Shaped,b=10.0,d=12.0



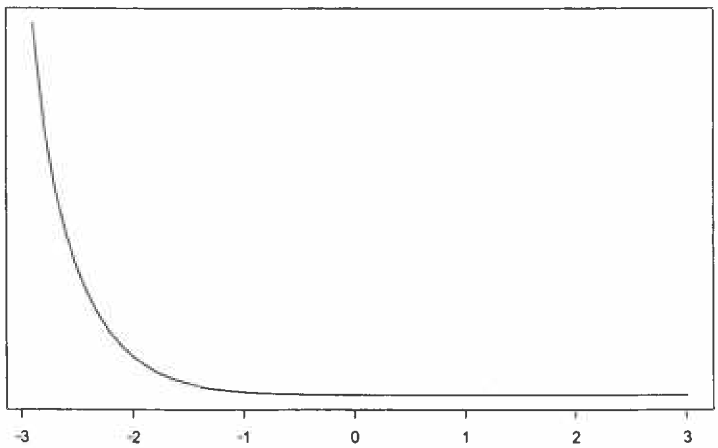
Density Curve of Type I Distribution:Bell-Shaped,b=0.8,d=0.7



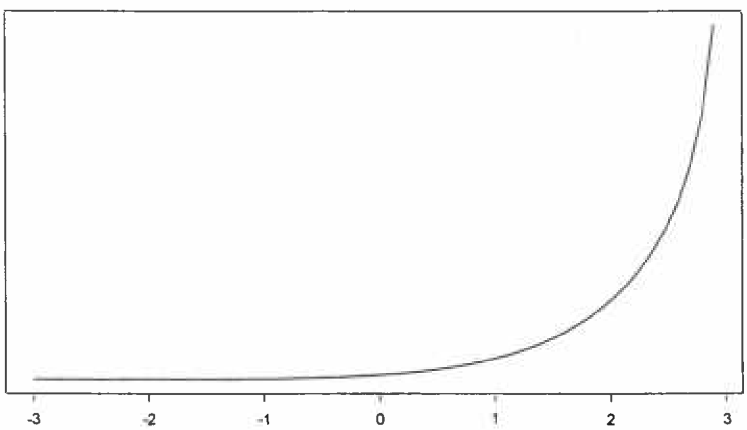
https://doi.org/10.26219/heal.aueb.5743

Figure 3.1.2
Density Curves of the Type I Distribution

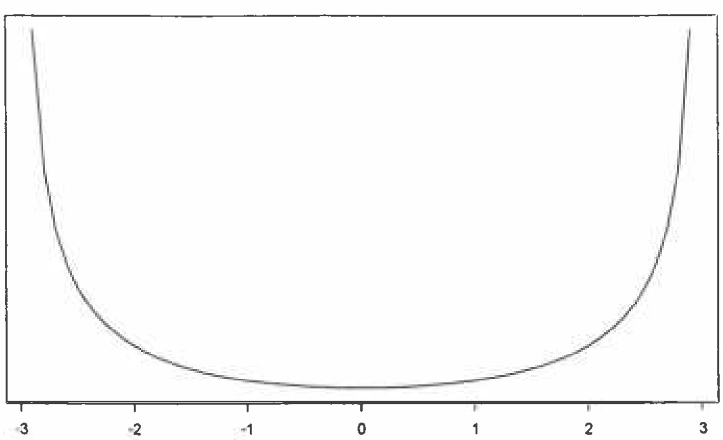
Density Curve of Type I Distribution:J-Shaped,b=-0.2,d=11.0



Density Curve of Type I Distribution:Reversed J-Shaped,b=5.0,d=-0.3



Density Curve of Type I Distribution:U-Shaped,b=-0.5,d=-0.5

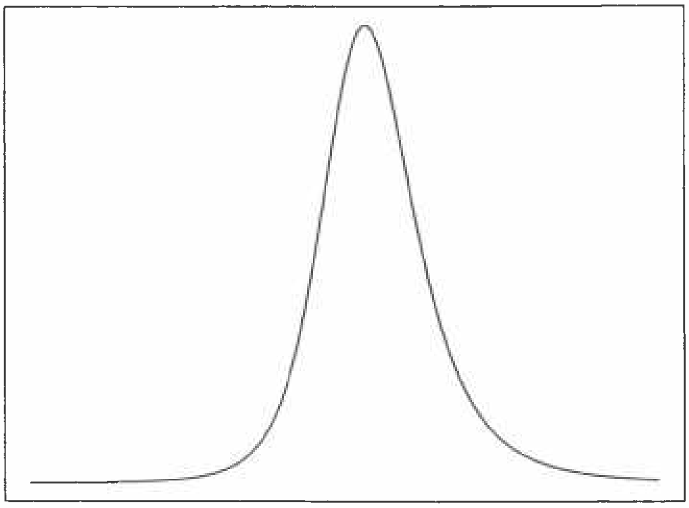


https://doi.org/10.26219/heal.aueb.5743

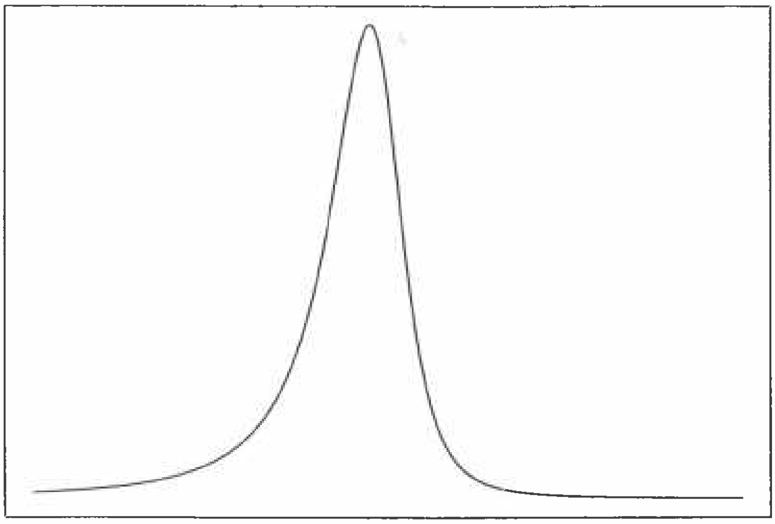
Figure 3.2

Density Curves of the Type IV Distribution

Density Curve of Type IV Distibution:a=1.5,b=2.5,c=-1.1



Density Curve of Type IV Distibution:a=0.75,b=1.5,c=1.0

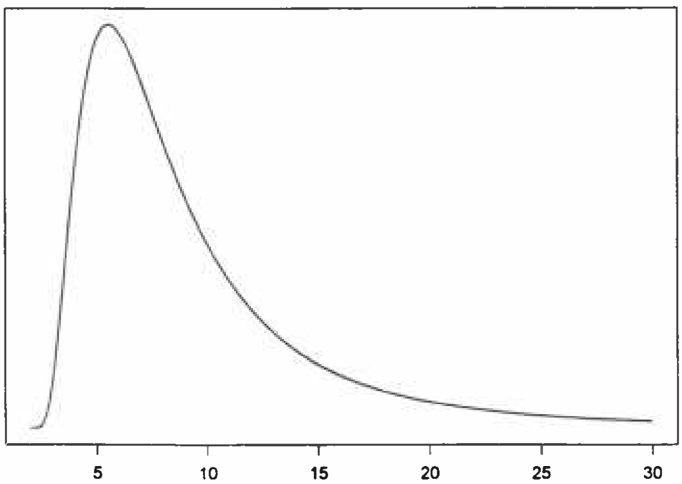


https://doi.org/10.26219/heal.aueb.5743

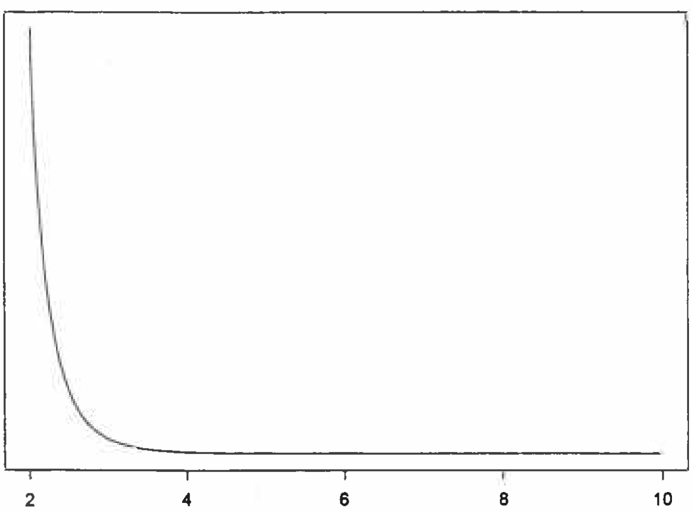
Figure 3.3

Density Curves of the Type VI Distribution

Density Curve of Type VI Distribution:Bell Shaped,b=7.0,c=11.0



Density Curve of Type VI Distribution:J-Shaped,b=-0.05,c=8.0



https://doi.org/10.26219/heal.aueb.5743

3.2.3 Simulation from Pearson's Family

3.2.3.1 Simulation from Type I

Let assume that the random variable X follows the Beta Type II distribution, denoted as BetaII (a,b), with probability density function given by

$$f(x)=\frac{x^{a-1}}{(1+x)^{a+b}\mathrm{B}(a,b)}, \quad x,a,b>0$$

where a,b are the parameters and $\mathrm{B}(a,b)$ is the Beta function. Then by applying a random variable transformation it holds that if the random variable X follows the BetaII $(b+1,d+1)$ distribution then the random variable $Y=\dfrac{cX-a}{1+X}$ follows the Pearson Type I distribution with parameters a,b,c,d.

Hence, the algorithm for simulating a random variable Y from the Type I distribution is the following:

- Generate a random variable X from BetaII $(b+1,d+1)$ distribution
- Set $Y=\dfrac{cX-a}{1+X}$.

Note that the generator proposed by Devroye (1986, p. 481) is misleading since it cannot generate values for $b,d<0$.

A simple algorithm for generating random variables from the BetaII $(b+1,d+1)$ distribution is the following:

https://doi.org/10.26219/heal.aueb.5743

- Generate random variables X_1 and X_2 from the Gamma($b+1,1$) and Gamma($d+1,1$) distribution respectively
- Set $X = \dfrac{X_1}{X_2}$, then the random variable X follows the Beta type II distribution.

Hence, the speed for generating random variables from the Type I distribution depends on the generator used for the Gamma distribution. Several algorithms for generating random variables from the Gamma distribution are discussed in Devroye (1986) and Ripley (1987).

3.2.3.2 Simulation from TYPE IV

The simulation of a random variable from Type IV is not very simple. However, by random variable transformation it is shown that if a random variable X follows the Type IV distribution then random variable $Y = \arctan\left(\dfrac{X}{\alpha}\right)$ follows a distribution with probability density function,

$$f(y) = K\alpha\cos^{2b\text{-}2} y \exp(-cy), \qquad (3.12)$$

where $y \in \left[-\dfrac{\pi}{2}, \dfrac{\pi}{2}\right]$ and K is the normalizing constant.

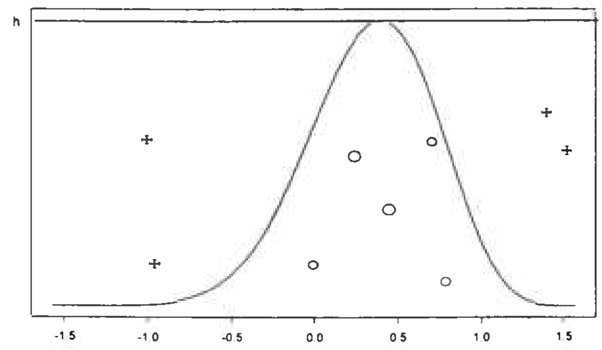
Hence, a simple algorithm for simulating a random variable X from the Type IV distribution is:

- Generate a random variable Y from the density function (3.12)
- Set $X = \tan(Y)$.

https://doi.org/10.26219/heal.aueb.5743

A simple method for generating a random variable Y from the distribution (3.12) is the rejection method (for details see Morgan, 1984, and Ripley, 1987). Since the density function (3.12) is non-zero over the range $\left[-\frac{\pi}{2},\frac{\pi}{2}\right]$, it is easy to box it in, as shown in figure 3.4. By using two Uniform(0,1) random variables, U_1, U_2, we randomly generate points over the rectangle with cartesian coordinates $\left((-\frac{\pi}{2}+\pi U_1), hU_2\right)$. Points above the density (3.12) are rejected, denoted by crosses in figure 3.4, while the points below the density (3.12) are accepted, denoted by circles in figure 3.4, and we take $\left((-\frac{\pi}{2}+\pi U_1), hU_2\right)$ as a realization of the random variable Y. In our case h is the mode of the density (3.12). The choice of h is very crucial since it affects the probability of acceptance. The probablity of accepting a point is $\frac{1}{h\left(\frac{\pi}{2}-(-\frac{\pi}{2})\right)}$. Thus, as h increases the probability of accepting a point gets smaller. In our case, the choice of h as the mode of density (3.12) is the most efficient. However, the disadvantage of the above algorithm is its slow speed. Another possibility, as pointed out to us by Prof. L. Devroye, is to use the generator in Devroye (1986) which is probably faster.

Figure 3.4
Illustration of the Rejection Algorithm



3.2.3.3 Simulation from TYPE VI

A simple algorithm for generating a random variable X from the Type VI distribution of the Pearson's family is the following (Devroye, 1986, p.481):

- Generate a random variable X_1 from a Gamma(c-b-1,1) distribution
- Generate a random variable X_2 from a Gamma(b+1,1) distribution
- Set $X = a\dfrac{X_1 + X_2}{X_1}$.

3.3 Matching the Moments

In order for the simulated values to match the given moments a quantity must be added to them. This is due to the fact that the simulated values have central moments 0, μ'_2, μ'_3, μ'_4 and thus a constant must be added in order to make them have the given simple moment. This quantity differs among the three main types of the Pearson's family. By adding the

https://doi.org/10.26219/heal.aueb.5743

appropriate quantity, given in table 3.3, to the generated values we obtain the simple moments of the corresponding distribution.

Table 3.3
Quantity to add to the generated random variables

Type	Quantity to add
Type I	$\mu_1 - \frac{1}{2}\frac{\mu_3}{\mu_2}(\frac{r+2}{r-2})$
Type IV	$\mu_1 - \frac{ac}{r}$
Type VI	$\mu_1 - \frac{1}{2}\frac{\mu_3}{\mu_2}(\frac{r+2}{r-2}) + \frac{ac}{b-c}$

To sum up, we give the basic steps of the algorithm for generating random variables from the three main types of the Pearson's family given the first four sample moments.

Pearson's Algorithm

- Calculate the central moments (only in the case we have the simple moments).
- Calculate the κ-coefficient and choose the appropriate member of the Pearson's family.
- Calculate the parameters of the chosen member.
- Simulate from this member.
- Add the corresponding quantity given in Table 3.3, in order to match the four simple moments.

https://doi.org/10.26219/heal.aueb.5743

3.4 Simulation Study

So far, we have developed two algorithms for generating random variables from a unimodal continuous distribution given the first four moments. In this section we apply both algorithms to the same given sets of first four moments. For each set of moments and for a specified number of replications we generate samples of size n from the corresponding set of moments.

Particularly, the procedure is as follows: let $\mu_1, \mu_2, \mu_3, \mu_4$ be the given set of four moments and B be the number of replications. At each replication $b, b=1,\ldots,B$, we first draw samples of size n and then we compute the sample moments $\mu_{1(b)}, \mu_{2(b)}\, \mu_{3(b)}, \mu_{4(b)}$ of the $b-$th generated sample. After that, we compute the estimate of the $i-$th moment as $\hat{\mu}_i = \frac{\sum_{b=1}^{B} \mu_{i(b)}}{B}$ for $i=1,\ldots 4$. When the above procedure is completed we expect the values $\hat{\mu}_i$, $i=1,\ldots 4$, to be close to the values of the given moments $\mu_i, i=1,\ldots 4$. Additionally, we compute the standard deviation of the estimates, $S_{\hat{\mu}_i}$.

In order to examine the behaviour of the algorithms with respect to the sample size n we drew samples of different sample size; namely n=50, 100, 150, 250, 500, 750 and 1000 respectively.

The combinations of the given moments used for the simulations are given in the table 3.4.

Table 3.4
Given Sets of First Four Moments used for the simulation

	Type	μ_1	μ	μ	μ
		0	1	3	17
Set 1	VI	0	1	-3	17
		0	1	2	6
Set 2	I	0	1	-2	6
		0	1	1	8
Set 3	IV	0	1	-1	8

The choice of the moments is arbitrary in the sense that they do not correspond to any known distribution function. The first set of the given moments corresponds to the Type VI distribution of the Pearson's system while the second and third set corresponds to the Type I and Type IV respectively.

In tables 3.5.1 to 3.5.12 we report the results of the simulation for 1000 and 5000 replications respectively. The entries are the matched moments and the standard deviations of the computed four moments at each replication b. First of all, from these tables we notice that both algorithms match the given sets of moments, as expected. Second, the standard deviation of the moments, as it is expected, decreases as the sample size increases, which was also expected. Moreover, for both sizes of replications, 1000 and 5000, the results are almost identical. In figures 3.5.1 to 3.5.6, for each moment, we plot the difference between the given moment and the matched moment, i.e. $\mu_i - \hat{\mu}_i$, versus the sample size. We notice that as the sample size increases the difference $\hat{\mu}_i - \mu_i, i = 1,2,3,4$, decreases and tends to become equal to zero.

Moreover, we can not see a systematic pattern in the sense that the matched moments do not always underestimate (underfit) or overestimate (overfit) the given moments. In figures 3.6.1 to 3.6.6 we plot for both the algorithms and for each moment, we plot its estimated standard deviation versus the sample size. From these figures, we notice that the standard deviation of the first two moments computed at each replication does not differ significantly between the two algorithms. However, there is a great difference for the fourth moment of the first and third pair of the given moments. Notice that the variability of the generated third and fourth moments of the Pearson's algorithm is always higher than the corresponding moments of the Devroye's algorithm. We can see that the standard deviation of Pearson's algorithm for the fourth moment is extremely large. We must state here that when the fourth given sample moment is large we expect its standard deviation to be high. Finally, we see, for both algorithms, that the standard deviations of the corresponding generated moments from the second pair of given moments are almost identical.

After experimenting with both algorithms we could conclude that both algorithms match the given moments. The matching is better when the sample size is large. The Pearson's algorithm gives us smoother samples. However, the algorithm for generating random variables from the Pearson family does not match the fourth given moment well, especially when it takes large values. Moreover, for large values of the fourth moments its variability is too high in comparison with the Devroye's algorithm.

https://doi.org/10.26219/heal.aueb.5743

Table 3.5.1

Simulation results with moments $\mu_1 = 0, \mu_2 = 1, \mu_3 = 3, \mu_4 = 17$.

Standard Deviations are given in brackets

Pearson's Algorithm

Replications	$\mathbf{n_i}$	$\hat{\mu}_1$	$\hat{\mu}_2$	$\hat{\mu}_3$	$\hat{\mu}_4$
	n_1=50	0.00047	1.01180	3.17521	19.87207
		(0.14057)	(0.60334)	(6.24837)	(98.24733)
	n_2=100	-0.00243	0.99441	2.95400	16.41238
		(0.09766)	(0.38028)	(2.91223)	(28.44912)
	n_3=150	0.00356	1.00891	3.01787	16.86989
		(0.08230)	(0.33042)	(2.60979)	(27.03633)
1000	n_4=250	-0.00107	0.99572	2.96998	16.63063
		(0.06355)	(0.25151)	(1.97529)	(20.86372)
	n_5=500	0.00004	0.99850	2.98577	16.75673
		(0.04438)	(0.17829)	(1.41075)	(14.79775)
	n_6=750	-0.00019	0.99874	3.00412	17.11801
		(0.03593)	(0.14283)	(1.14661)	(12.50226)
	n_7=1000	0.00035	1.00351	3.02243	17.16727
		(0.03107)	(0.12425)	(0.99003)	(10.9772)

Devroye's Algorithm

		$\hat{\mu}_1$	$\hat{\mu}$	$\hat{\mu}$	$\hat{\mu}$
	n_1=50	-0.00006	1.00605	3.03032	17.15993
		(0.14276)	(0.58868)	(2.69816)	(13.6609)
	n_2=100	-0.00102	0.99199	2.97501	16.77613
		(0.09743)	(0.38904)	(1.86919)	(9.14384)
	n_3=150	0.00202	0.99138	3.04865	16.75456
		(0.08066)	(0.32434)	(1.51807)	(7.55351)
1000	n_4=250	0.00052	1.00712	3.01975	17.18265
		(0.06383)	(0.26417)	(1.25865)	(6.15346)
	n_5=500	0.00290	1.00662	3.04051	17.14649
		(0.04365)	(0.17560)	(0.84345)	(4.19406)
	n_6=750	-0.00129	0.99061	2.97470	16.81854
		(0.03621)	(0.14401)	(0.67026)	(3.30310)
	n_7=1000	0.00018	0.99829	2.99882	16.98528
		(0.03112)	(0.12636)	(0.60094)	(2.99673)

https://doi.org/10.26219/heal.aueb.5743

Table 3.5.2

Simulation results with moments $\mu_1 = 0, \mu = 1, \mu_3 = 3, \mu_4 = 17$.

Standard Deviations are given in brackets

Pearson's Algorithm

Replications	n_i	$\hat{\mu}_1$	$\hat{\mu}$	$\hat{\mu}$	$\hat{\mu}$
	n_1=50	-0.00018	1.00024	2.99048	16.88967
		(0.14068)	(0.56129)	(4.67353)	(58.81671)
	n_2=100	0.00053	1.00565	3.04584	17.50828
		(0.10140)	(0.40388)	(3.39835)	(40.54278)
	n_3=150	-0.00106	1.00093	3.02912	17.42875
		(0.08160)	(0.32965)	(2.80129)	(36.11281)
5000	n_4=250	-0.00089	0.99756	2.98071	16.81344
		(0.06394)	(0.25338)	(2.02734)	(22.52211)
	n_5=500	-0.00086	0.99689	2.99237	17.06150
		(0.04390)	(0.17784)	(1.48972)	(18.14707)
	n_6=750	-0.00009	1.00191	3.01919	17.17712
		(0.03681)	(0.14801)	(1.18532)	(13.16714)
	n_7=1000	-0.00057	0.99981	3.00117	17.02983
		(0.03113)	(0.12587)	(1.02658)	(11.94197)

Devroye's Algorithm

Replications	n_i	$\hat{\mu}_1$	$\hat{\mu}$	$\hat{\mu}$	$\hat{\mu}$
	n_1=50	0.00349	1.01507	3.06750	17.41444
		(0.14040)	(0.56324)	(2.72791)	(13.32826)
	n_2=100	-0.00210	0.99300	2.97259	16.86634
		(0.09999)	(0.39916)	(1.92366)	(9.34488)
	n_3=150	0.00287	1.00660	3.05999	17.12390
		(0.08189)	(0.33132)	(1.56154)	(7.76832)
5000	n_4=250	-0.00008	1.00100	2.99820	17.01459
		(0.06341)	(0.25765)	(1.20907)	(6.00159)
	n_5=500	-0.00030	0.99797	2.99372	16.95375
		(0.04390)	(0.17764)	(0.84100)	(4.17744)
	n_6=750	0.00069	1.00096	3.00995	16.99719
		(0.03654)	(0.14505)	(0.69648)	(3.39699)
	n_7=1000	-0.00026	0.99965	2.99476	17.01551
		(0.03162)	(0.12599)	(0.60572)	(2.95825)

https://doi.org/10.26219/heal.aueb.5743

Table 3.5.3

Simulation results with moments $\mu_1 = 0, \mu = 1, \mu_3 = -3, \mu_4 = 17$.

Standard Deviations are given in brackets

Replications	n_i	Pearson's Algorithm $\hat{\mu}_1$	$\hat{\mu}$	$\hat{\mu}$	$\hat{\mu}$
	n_1=50	0.00755 (0.13518)	0.97453 (0.52188)	-2.78223 (3.44725)	14.51429 (27.12406)
	n_2=100	0.00176 (0.09869)	0.98740 (0.38943)	-2.88177 (2.92746)	15.72482 (28.96495)
	n_3=150	0.00250 (0.08266)	1.00068 (0.34087)	-3.09790 (2.84518)	18.38314 (31.22278)
1000	n_4=250	-0.00066 (0.06266)	1.00274 (0.25104)	-3.02471 (2.04367)	17.16003 (22.67638)
	n_5=500	-0.00039 (0.04407)	0.99795 (0.18170)	-2.95188 (1.41356)	16.38437 (14.97182)
	n_6=750	-0.00128 (0.03717)	1.00623 (0.14979)	-3.04685 (1.17698)	17.28679 (12.00897)
	n_7=1000	-0.00157 (0.03099)	1.00938 (0.12752)	-3.08352 (1.12011)	18.04241 (15.14824)

Replications	n_i	Devroye's Algorithm $\hat{\mu}_1$	$\hat{\mu}$	$\hat{\mu}$	$\hat{\mu}$
	n_1=50	0.00678 (0.13458)	0.99098 (0.54333)	-2.89726 (2.58345)	16.69699 (12.71367)
	n_2=100	-0.00419 (0.10156)	1.02566 (0.40880)	-3.10448 (1.99779)	17.59218 (9.76148)
	n_3=150	0.00022 (0.08356)	1.00040 (0.32249)	-3.00406 (1.58752)	16.92355 (7.49902)
1000	n_4=250	-0.00127 (0.06081)	1.00254 (0.24671)	-2.99884 (1.15448)	16.99809 (5.83759)
	n_5=500	-0.00010 (0.04377)	0.99458 (0.17444)	-3.00415 (0.84905)	16.93318 (4.13157)
	n_6=750	-0.00013 (0.03754)	1.00201 (0.14782)	-3.00785 (0.71988)	17.05719 (3.47412)
	n_7=1000	0.00115 (0.03189)	0.99604 (0.12842)	-2.97327 (0.62111)	16.90609 (3.02016)

Table 3.5.4

Simulation results with moments $\mu_1 = 0, \mu = 1, \mu_3 = 3, \mu_4 = 17$.

Standard Deviations are given in brackets

Pearson's Algorithm

Replications	n_i	$\hat{\mu}_1$	$\hat{\mu}$	$\hat{\mu}$	$\hat{\mu}$
	n_1=50	-0.00003 (0.14071)	1.00766 (0.59214)	-3.09207 (5.22152)	18.31128 (64.25341)
	n_2=100	-0.00057 (0.09813)	1.00457 (0.40896)	-3.09207 (3.41678)	18.31128 (38.91305)
	n_3=150	0.00034 (0.08133)	1.00248 (0.32631)	-3.01556 (2.72739)	17.18422 (34.07089)
5000	n_4=250	0.00129 (0.06310)	0.99557 (0.25512)	-2.97771 (2.03679)	16.83306 (22.29045)
	n_5=500	0.00008 (0.04521)	1.00022 (0.18037)	-3.00116 (1.46047)	17.02985 (16.71047)
	n_6=750	0.00028 (0.03599)	0.99972 (0.14412)	-2.99513 (1.18382)	17.00456 (14.07767)
	n_7=1000	-0.00055 (0.03077)	1.00074 (0.12303)	-3.00263 (1.03422)	17.05239 (12.63501)

Devroye's Algorithm

Replications	n_i	$\hat{\mu}_1$	$\hat{\mu}$	$\hat{\mu}$	$\hat{\mu}$
	n_1=50	-0.00143 (0.14238)	1.00863 (0.57533)	-3.03351 (2.74340)	17.22292 (13.44549)
	n_2=100	-0.00072 (0.09917)	0.99835 (0.40485)	-3.00912 (1.92761)	16.94905 (9.51872)
	n_3=150	0.00061 (0.08298)	1.00268 (0.33342)	-3.00264 (1.58362)	17.06073 (7.83073)
5000	n_4=250	-0.00082 (0.06308)	1.00411 (0.24991)	-3.00842 (1.18904)	17.07519 (5.84671)
	n_5=500	-0.00004 (0.04453)	0.99749 (0.17530)	-3.00082 (0.85290)	16.95092 (4.09138)
	n_6=750	0.00045 (0.03563)	0.99999 (0.14279)	-2.98468 (0.68850)	16.99684 (3.35518)
	n_7=1000	0.00000 (0.03135)	1.00058 (0.12706)	-3.00215 (0.59755)	17.03265 (2.97511)

https://doi.org/10.26219/heal.aueb.5743

Table 3.5.5

Simulation results with moments $\mu_1 = 0, \mu = 1, \mu_3 = 2, \mu_4 = 6$.

Standard Deviations are given in brackets

Pearson's Algorithm

Replications	n_i	$\hat{\mu}_1$	$\hat{\mu}$	$\hat{\mu}$	$\hat{\mu}$
	n_1=50	0.00714	1.00791	2.03132	6.11515
		(0.14072)	(0.32123)	(1.02714)	(3.29166)
	n_2=100	0.00245	1.00416	2.01066	6.03120
		(0.10087)	(0.21986)	(0.70149)	(2.04488)
	n_3=150	-0.00297	0.98770	1.96155	5.87688
		(0.08310)	(0.18509)	(0.58791)	(1.86672)
1000	n_4=250	-0.00071	0.99786	1.99545	5.98488
		(0.06476)	(0.14640)	(0.46361)	(1.46381)
	n_5=500	0.00026	0.99764	1.99384	5.98003
		(0.04458)	(0.09980)	(0.31586)	(1.00369)
	n_6=750	0.00140	1.00211	2.00852	6.02657
		(0.03465)	(0.08018)	(0.25384)	(0.80762)
	n_7=1000	0.00012	1.00030	2.00395	6.01700
		(0.03087)	(0.06924)	(0.21945)	(0.69816)

Devroye's Algorithm

Replications	n_i	$\hat{\mu}_1$	$\hat{\mu}$	$\hat{\mu}$	$\hat{\mu}$
	n_1=50	0.00252	1.00770	2.00178	6.05284
		(0.14518)	(0.32036)	(0.99566)	(3.08393)
	n_2=100	0.00698	1.01354	2.01954	6.08957
		(0.10262)	(0.22766)	(0.70735)	(2.17938)
	n_3=150	0.00341	1.00494	1.99902	6.03214
		(0.08358)	(0.18885)	(0.58287)	(1.79924)
1000	n_4=250	0.00129	1.00180	1.99374	6.03242
		(0.06358)	(0.14085)	(0.43603)	(1.34600)
	n_5=500	-0.00120	0.99740	1.97477	5.96869
		(0.04414)	(0.09668)	(0.29862)	(0.92338)
	n_6=750	0.00120	1.00135	1.99009	6.01662
		(0.03695)	(0.08402)	(0.25760)	(0.79788)
	n_7=1000	0.00036	1.00090	1.98711	6.00722
		(0.03271)	(0.07494)	(0.22998)	(0.70780)

https://doi.org/10.26219/heal.aueb.5743

Table 3.5.6

Simulation results with moments $\mu_1 = 0, \mu = 1, \mu_3 = 2, \mu_4 = 6$

Standard Deviations are given in brackets

Replications	n_i	Pearson's Algorithm			
		$\hat{\mu}_1$	$\hat{\mu}$	$\hat{\mu}$	$\hat{\mu}$
	n_1=50	-0.00176 (0.13886)	0.99353 (0.31071)	1.98254 (0.98505)	5.94887 (3.13446)
	n_2=100	-0.00140 (0.10065)	0.99711 (0.22530)	1.99340 (0.71086)	5.98157 (2.24521)
	n_3=150	0.00143 (0.08236)	1.00156 (0.18403)	2.00477 (0.57939)	6.01347 (1.83092)
5000	n_4=250	0.00004 (0.06276)	0.99933 (0.13963)	1.99845 (0.44224)	5.99181 (1.40227)
	n_5=500	0.00125 (0.04526)	1.00168 (0.10024)	2.00689 (0.31681)	6.02085 (1.00301)
	n_6=750	0.00034 (0.03651)	0.99998 (0.08181)	2.00120 (0.25915)	6.00271 (0.82270)
	n_7=1000	0.00050 (0.03134)	0.99896 (0.07040)	1.99790 (0.22270)	5.99159 (0.70607)

Replications	n_i	Devroye's Algorithm			
		$\hat{\mu}_1$	$\hat{\mu}$	$\hat{\mu}$	$\hat{\mu}$
	n_1=50	-0.00389 (0.14079)	0.99420 (0.31581)	1.96885 (0.97431)	5.96458 (3.01099)
	n_2=100	-0.00329 (0.09880)	0.99420 (0.22040)	1.96408 (0.68351)	5.94244 (2.11813)
	n_3=150	0.00198 (0.08064)	1.00245 (0.18098)	1.99367 (0.55973)	6.02531 (1.73023)
5000	n_4=250	-0.00054 (0.06239)	0.99984 (0.14013)	1.98344 (0.43426)	5.99904 (1.34705)
	n_5=500	0.00024 (0.04548)	0.99957 (0.10096)	1.98379 (0.31387)	5.99710 (0.97094)
	n_6=750	-0.00089 (0.03655)	0.99881 (0.08121)	1.98081 (0.25092)	5.99105 (0.77496)
	n_7=1000	-0.00027 (0.03157)	0.99884 (0.07018)	1.98006 (0.21713)	5.98542 (0.67008)

https://doi.org/10.26219/heal.aueb.5743

Table 3.5.7

Simulation results with moments $\mu_1 = 0, \mu = 1, \mu_3 = -2, \mu_4 = 6$.

Standard Deviations are given in brackets

Replications	**n**	**Pearson's Algorithm**			
		$\hat{\mu}_1$	$\hat{\mu}$	$\hat{\mu}$	$\hat{\mu}$
	n_1=50	0.00433 (0.14221)	1.01003 (0.33224)	-2.03985 (1.04859)	6.14092 (3.32026)
	n_2=100	-0.00386 (0.09901)	1.00754 (0.22805)	-2.02037 (0.72141)	6.14092 (2.29264)
	n_3=150	-0.00167 (0.07928)	0.99845 (0.18104)	-1.99070 (0.57567)	5.95753 (1.83635)
1000	n_4=250	0.00153 (0.06258)	0.99522 (0.13898)	-1.98700 (0.43908)	5.95840 (1.39105)
	n_5=500	0.00127 (0.04255)	0.99628 (0.09443)	-1.98933 (0.30011)	5.96607 (0.95450)
	n_6=750	-0.00150 (0.03601)	1.00276 (0.08040)	-2.01118 (0.25436)	6.03796 (0.80835)
	n_7=1000	-0.00088 (0.03183)	1.00055 (0.07205)	-2.00217 (0.22831)	6.00490 (0.72270)

		Devroye's Algorithm			
		$\hat{\mu}_1$	$\hat{\mu}$	$\hat{\mu}$	$\hat{\mu}$
	n_1=50	0.00120 (0.13672)	0.99361 (0.30620)	-1.95884 (0.93834)	5.90908 (2.89655)
	n_2=100	-0.00057 (0.10134)	1.00125 (0.22551)	-1.98516 (0.69188)	6.00329 (2.12475)
	n_3=150	-0.00217 (0.08064)	1.00294 (0.18179)	-1.99330 (0.56432)	6.02428 (1.74238)
1000	n_4=250	-0.00340 (0.06112)	1.00848 (0.13928)	-2.00684 (0.43565)	6.06301 (1.35801)
	n_5=500	-0.00066 (0.04586)	1.00165 (0.10200)	-1.98860 (0.31349)	6.01122 (0.96395)
	n_6=750	-0.00096 (0.03814)	1.00191 (0.08580)	-1.99024 (0.26522)	6.01494 (0.81859)
	n_7=1000	0.00084 (0.03226)	0.99525 (0.07211)	-1.96989 (0.22334)	5.95097 (0.68955)

https://doi.org/10.26219/heal.aueb.5743

Table 3.5.8

Simulation results with moments $\mu_1 = 0, \mu = 1, \mu_3 = -2, \mu_4 = 6$.

Standard Deviations are given in brackets

Pearson's Algorithm

Replications	**n_i**	$\hat{\mu}_1$	$\hat{\mu}$	$\hat{\mu}$	$\hat{\mu}$
	n_1=50	-0.00205	1.00073	-2.00664	6.02435
		(0.14346)	(0.31967)	(1.01108)	(3.20556)
	n_2=100	-0.00165	1.00404	-2.01426	6.04542
		(0.10084)	(0.22592)	(0.71481)	(2.26152)
	n_3=150	-0.00055	0.99873	-1.99771	5.99251
		(0.08189)	(0.18158)	(0.57174)	(1.80616)
5000	n_4=250	0.00128	0.99602	-1.98997	5.97093
		(0.06329)	(0.14076)	(0.44341)	(1.40041)
	n_5=500	-0.00121	1.00102	-2.00523	6.01609
		(0.04515)	(0.10180)	(0.32227)	(1.02233)
	n_6=750	-0.00093	1.00106	-2.00550	6.01772
		(0.03669)	(0.08148)	(0.25717)	(0.81448)
	n_7=1000	-0.00025	1.00035	-2.00354	6.01392
		(0.03149)	(0.06996)	(0.22087)	(0.70021)

Devroye's Algorithm

		$\hat{\mu}_1$	$\hat{\mu}$	$\hat{\mu}$	$\hat{\mu}$
	n_1=50	0.00156	0.99913	-1.98209	5.99953
		(0.14182)	(0.31637)	(0.98322)	(3.04231)
	n_2=100	-0.00081	1.00265	-1.99204	6.02397
		(0.10017)	(0.22234)	(0.68817)	(2.12435)
	n_3=150	-0.00055	1.00384	-1.99734	6.04508
		(0.08148)	(0.18303)	(0.56898)	(1.76377)
5000	n_4=250	0.00055	0.99703	-1.97365	5.96328
		(0.06271)	(0.14148)	(0.44029)	(1.36702)
	n_5=500	0.00017	0.99984	-1.98226	5.99201
		(0.04444)	(0.09916)	(0.30767)	(0.95367)
	n_6=750	-0.00043	1.00054	-1.98537	6.00166
		(0.03672)	(0.08237)	(0.25530)	(0.78994)
	n_7=1000	0.00031	0.99885	-1.98170	5.99254
		(0.03193)	(0.07119)	(0.22031)	(0.68064)

https://doi.org/10.26219/heal.aueb.5743

Table 3.5.9
Simulation results with moments $\mu_1 = 0, \mu = 1, \mu_3 = 1, \mu_4 = 8$.
Standard Deviations are given in brackets

Replications	n_i	Pearson's Algorithm $\hat{\mu}_1$	$\hat{\mu}$	$\hat{\mu}$	$\hat{\mu}$
	n_1=50	0.00521	0.99389	1.03329	7.49383
		(0.14191)	(0.35628)	(2.38482)	(21.99276)
	n_2=100	-0.00228	1.00187	1.01142	7.48489
		(0.10238)	(0.24991)	(1.70631)	(16.38019)
	n_3=150	0.00017	0.99639	1.01763	8.53125
		(0.08060)	(0.21974)	(2.43073)	(43.55561)
1000	n_4=250	-0.00213	1.00335	1.01839	7.79556
		(0.06235)	(0.16016)	(1.12043)	(11.0650)
	n_5=500	0.00036	1.00455	1.00545	7.64381
		(0.04246)	(0.11387)	(0.90511)	(11.88364)
	n_6=750	0.00022	0.99694	0.98474	7.63039
		(0.03676)	(0.09569)	(0.77890)	(10.72170)
	n_7=1000	0.00031	1.00292	1.02416	8.17015
		(0.03093)	(0.08141)	(0.81751)	(14.999)

Replications	n_i	Devroye's Algorithm $\hat{\mu}_1$	$\hat{\mu}$	$\hat{\mu}$	$\hat{\mu}$
	n_1=50	0.00527	1.02395	1.02607	8.24585
		(0.10070)	(0.27161)	(0.95533)	(3.34709)
	n_2=100	-0.00186	1.00082	0.97601	7.96824
		(0.08245)	(0.20838)	(0.77013)	(2.52170)
	n_3=150	-0.00397	0.99167	0.96601	7.89767
		(0.06123)	(0.16986)	(0.57647)	(2.08563)
1000	n_4=250	0.00029	0.99755	1.00478	7.97469
		(0.04402)	(0.11893)	(0.41474)	(1.42091)
	n_5=500	-0.00115	1.00164	0.99056	8.02490
		(0.03629)	(0.09981)	(0.34184)	(1.20473)
	n_6=750	0.00149	1.00626	1.00939	8.07225
		(0.03195)	(0.07890)	(0.30982)	(0.96768)
	n_7=1000	0.00415	0.98174	-0.96512	7.77408
		(0.14356)	(0.37186)	(1.32044)	(4.45256)

https://doi.org/10.26219/heal.aueb.5743

Table 3.5.10

Simulation results with moments $\mu_1 = 0, \mu = 1, \mu_3 = 1, \mu_4 = 8$.

Standard Deviations are given in brackets

Pearson's Algorithm

Replications	n_i	$\hat{\mu}_1$	$\hat{\mu}$	$\hat{\mu}$	$\hat{\mu}$
	n_1=50	0.00383 (0.14075)	1.01273 (0.37458)	1.06788 (2.86627)	7.92995 (35.25930)
	n_2=100	-0.00105 (0.10103)	0.99784 (0.27321)	1.02870 (3.43727)	8.58014 (82.50836)
	n_3=150	0.00193 (0.08180)	1.00333 (0.21645)	1.01631 (1.81355)	7.87412 (27.98008)
5000	n_4=250	-0.00302 (0.06288)	0.99992 (0.17290)	1.00201 (2.66835)	8.60186 (84.78524)
	n_5=500	-0.00072 (0.04455)	0.99881 (0.11802)	1.00201 (1.14650)	8.60186 (23.35891)
	n_6=750	-0.00009 (0.03678)	0.99828 (0.10179)	1.01200 (1.42613)	8.53537 (45.34676)
	n_7=1000	0.00036 (0.03134)	0.99932 (0.08184)	0.99683 (0.75222)	7.80366 (13.57320)

Devroye's Algorithm

Replications	n_i	$\hat{\mu}_1$	$\hat{\mu}$	$\hat{\mu}$	$\hat{\mu}$
	n_1=50	-0.00197 (0.14093)	0.99751 (0.37520)	0.98560 (1.31977)	7.96625 (4.56264)
	n_2=100	-0.00019 (0.10119)	1.00276 (0.26455)	1.00048 (0.95634)	8.03349 (3.18450)
	n_3=150	-0.00028 (0.08190)	0.99941 (0.21505)	1.00118 (0.77088)	7.98876 (2.60197)
5000	n_4=250	0.00149 (0.06398)	1.00479 (0.16859)	1.01621 (0.60001)	8.05590 (2.06682)
	n_5=500	0.00059 (0.04520)	1.00161 (0.11665)	1.00916 (0.42381)	8.01861 (1.42316)
	n_6=750	0.00049 (0.03666)	1.00069 (0.09660)	1.00455 (0.34670)	8.00721 (1.16545)
	n_7=1000	0.00074 (0.03206)	1.00169 (0.08386)	1.00567 (0.30133)	8.02710 (1.01813)

https://doi.org/10.26219/heal.aueb.5743

Table 3.5.11

Simulation results with moments $\mu_1 = 0, \mu = 1, \mu_3 = -1, \mu_4 = 8$.

Standard Deviations are given in brackets

Pearson's Algorithm

Replications	n_i	$\hat{\mu}_1$	$\hat{\mu}$	$\hat{\mu}$	$\hat{\mu}$
	n_1=50	0.00339	0.99223	-0.95115	7.95703
		(0.13935)	(0.36764)	(3.77114)	(61.02023)
	n_2=100	0.00837	0.99680	-1.03457	9.88441
		(0.10383)	(0.31307)	(4.68970)	(104.2768)
	n_3=150	0.00277	0.99700	-0.93566	7.27604
		(0.08238)	(0.20685)	(1.55287)	(19.50002)
1000	n_4=250	0.00069	0.99658	-0.94537	7.07690
		(0.06502)	(0.15276)	(0.99034)	(8.68342)
	n_5=500	0.00258	0.99832	-0.97916	7.33460
		(0.04477)	(0.11603)	(0.90868)	(12.42055)
	n_6=750	-0.00080	1.00057	-0.97916	7.33460
		(0.03602)	(0.08670)	(0.66165)	(8.51111)
	n_7=1000	-0.00213	1.00406	-1.08816	10.18666
		(0.03295)	(0.09871)	(1.69803)	(54.48437)

Devroye's Algorithm

Replications	n_i	$\hat{\mu}_1$	$\hat{\mu}$	$\hat{\mu}$	$\hat{\mu}$
	n_1=50	0.00415	0.98174	-0.96512	7.77408
		(0.14356)	(0.37186)	(1.32044)	(4.45256)
	n_2=100	0.00348	1.00429	-0.97284	8.01091
		(0.10072)	(0.27234)	(0.94204)	(3.32422)
	n_3=150	0.00127	0.99086	-0.98699	7.87885
		(0.07937)	(0.21380)	(0.74324)	(2.55679)
1000	n_4=250	-0.00143	0.99492	-1.01245	7.91400
		(0.06392)	(0.15969)	(0.58872)	(1.92863)
	n_5=500	0.00167	1.00028	-0.98333	8.02655
		(0.04307)	(0.11671)	(0.39265)	(1.43831)
	n_6=750	-0.00242	1.00203	-1.01972	8.00267
		(0.03655)	(0.09414)	(0.34274)	(1.14477)
	n_7=1000	-0.00090	1.00066	-1.00793	7.99865
		(0.03164)	(0.08482)	(0.29214)	(1.01420)

https://doi.org/10.26219/heal.aueb.5743

Table 3.5.12

Simulation results with moments $\mu_1 = 0, \mu = 1, \mu_3 = -1, \mu_4 = 8$.

Standard Deviations are given in brackets

Replications	n_i	Pearson's Algorithm $\hat{\mu}_1$	$\hat{\mu}$	$\hat{\mu}$	$\hat{\mu}$
	n_1=50	-0.00109	1.00366	-1.07328	8.86291
		(0.14068)	(0.39503)	(4.19222)	(71.96981)
	n_2=100	0.00076	1.00339	-1.02424	8.02287
		(0.09968)	(0.26685)	(2.17323)	(28.24342)
	n_3=150	-0.00047	1.00080	-1.01010	8.04700
		(0.08290)	(0.22000)	(2.20577)	(44.09722)
5000	n_4=250	0.00086	0.99723	-0.99864	7.78084
		(0.06279)	(0.16403)	(1.30665)	(16.36814)
	n_5=500	-0.00038	1.00042	-0.99251	7.78234
		(0.04498)	(0.11527)	(1.05830)	(19.08955)
	n_6=750	-0.00004	0.99880	-0.98239	7.50385
		(0.03686)	(0.09356)	(0.73750)	(10.28859)
	n_7=1000	-0.00010	0.99891	-0.99923	8.03914
		(0.03157)	(0.08533)	(0.96504)	(24.50956)

Replications	n_i	Devroye's Algorithm $\hat{\mu}_1$	$\hat{\mu}$	$\hat{\mu}$	$\hat{\mu}$
	n_1=50	0.00085	1.00277	-0.99844	7.96608
		(0.14182)	(0.37268)	(1.32980)	(4.53833)
	n_2=100	-0.00002	0.99769	-0.99312	7.96608
		(0.10100)	(0.26248)	(0.93715)	(3.18199)
	n_3=150	-0.00144	0.99953	-1.00883	7.99492
		(0.08215)	(0.21497)	(0.77138)	(2.60073)
5000	n_4=250	0.00057	1.00111	-0.99087	8.01351
		(0.06183)	(0.16495)	(0.59536)	(2.00461)
	n_5=500	-0.00025	1.00286	-1.00429	8.03521
		(0.04494)	(0.11874)	(0.42212)	(1.44641)
	n_6=750	0.00038	0.99978	-0.99925	7.99876
		(0.03671)	(0.09609)	(0.34084)	(1.16957)
	n_7=1000	-0.00063	1.00041	-1.00634	8.00000
		(0.03153)	(0.08372)	(0.29776)	(1.01393)

https://doi.org/10.26219/heal.aueb.5743

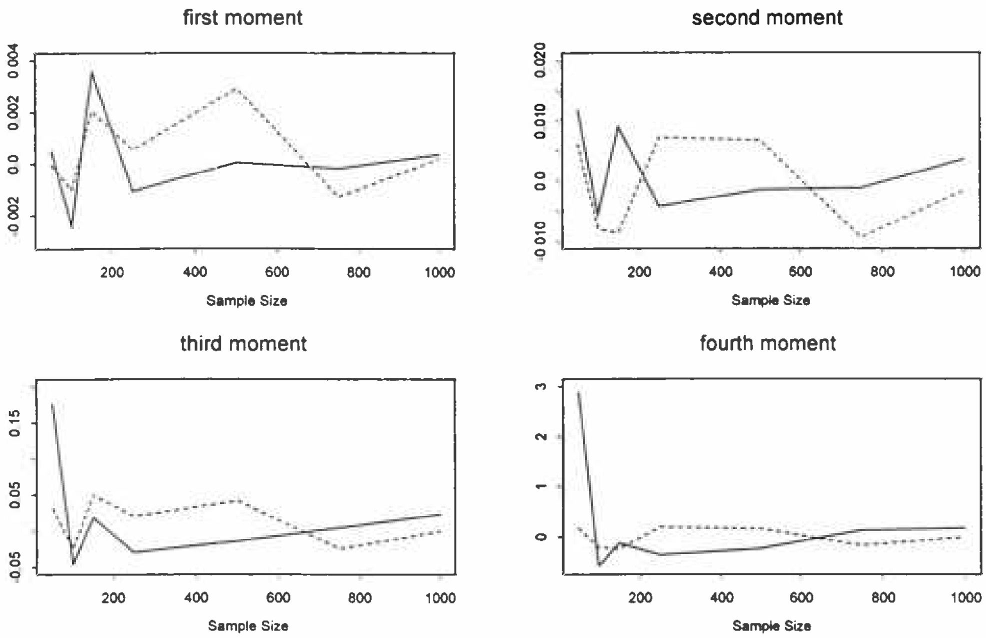
Figure 3.5.1

Plots of $\hat{\mu}_i - \mu_i, i = 1,2,3,4,$ versus Sample Size

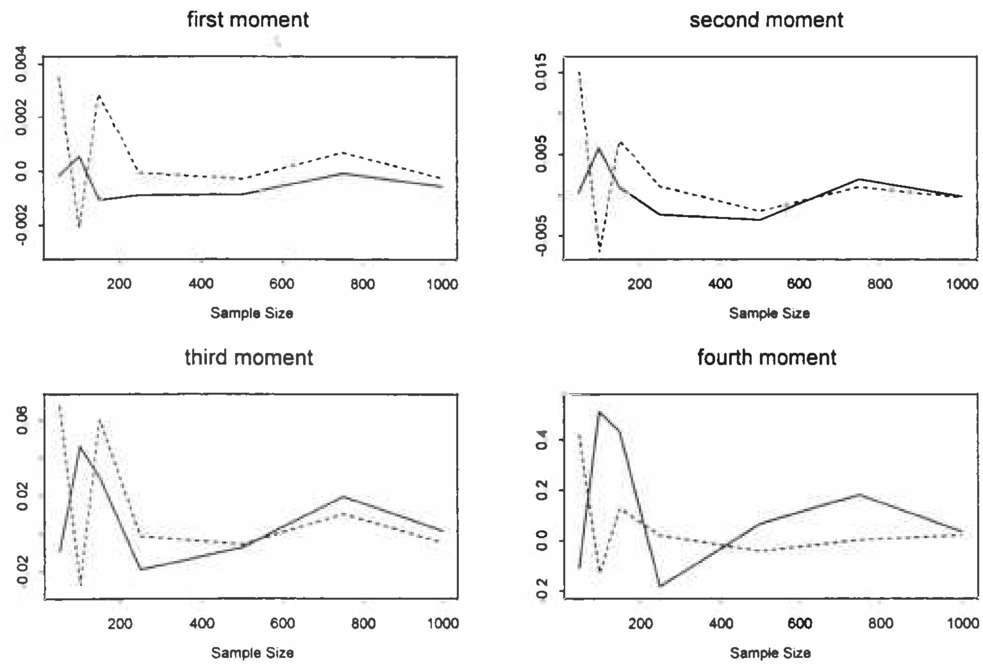
Given set of moments $\mu_1 = 0, \mu \;\; = 1, \mu_3 = 3, \mu_4 = 17$.

Solid line: Pearson's algorithm. Dashed line: Devroye's algorithm

1000 Iterations



5000 iterations



https://doi.org/10.26219/heal.aueb.5743

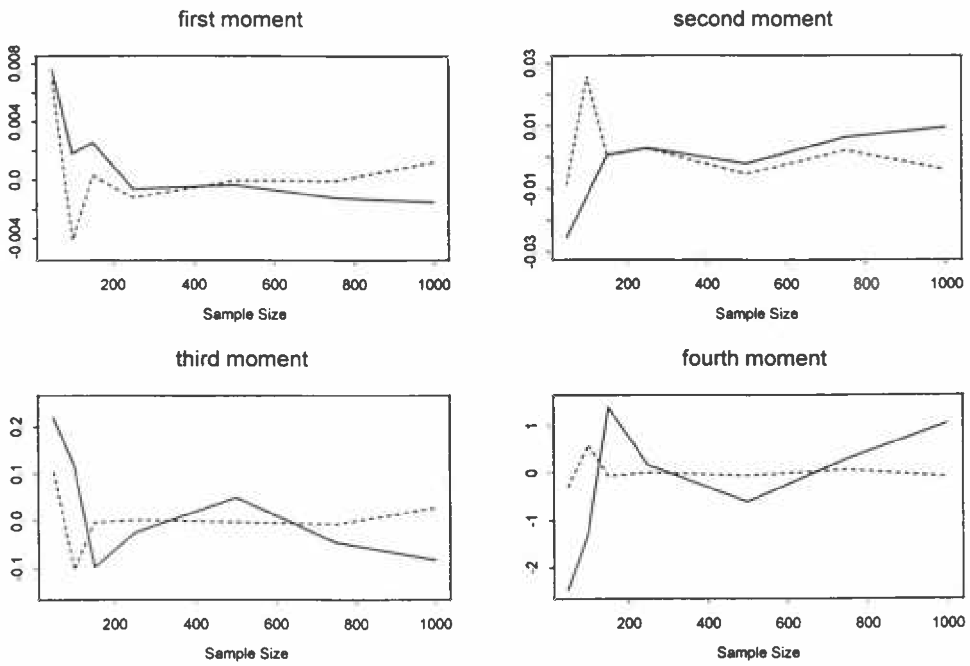
Figure 3.5.2

Plots of $\hat{\mu}_i - \mu_i, i = 1,2,3,4,$ versus Sample Size

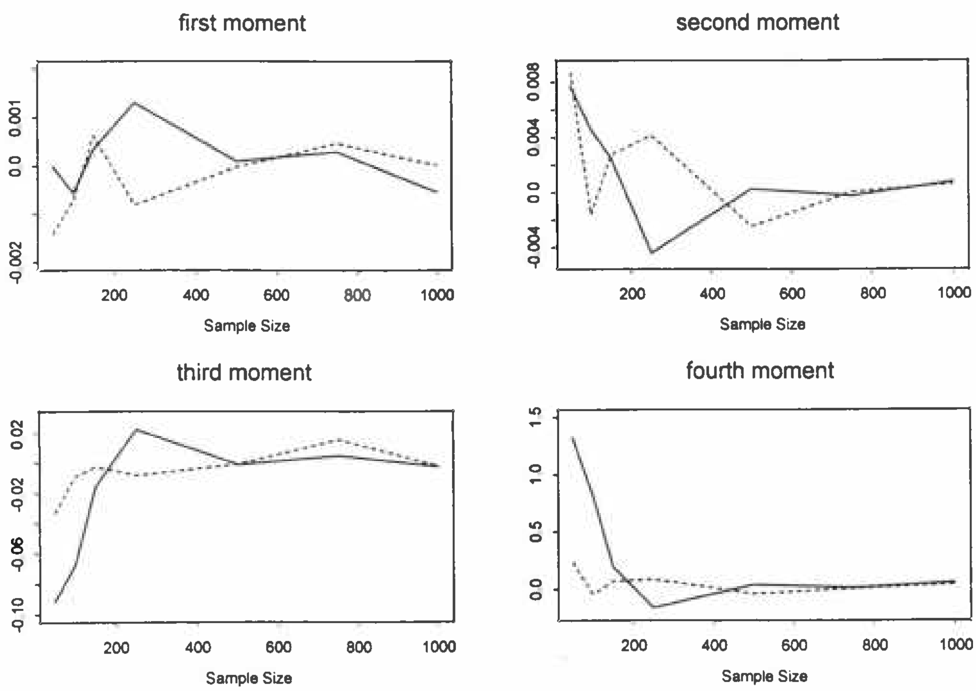
Given set of moments $\mu_1 = 0, \mu_2 = 1, \mu_3 = -3, \mu_4 = 17$.

Solid line: Pearson's algorithm. Dashed line: Devroye's algorithm

1000 Iterations



5000 iterations



https://doi.org/10.26219/heal.aueb.5743

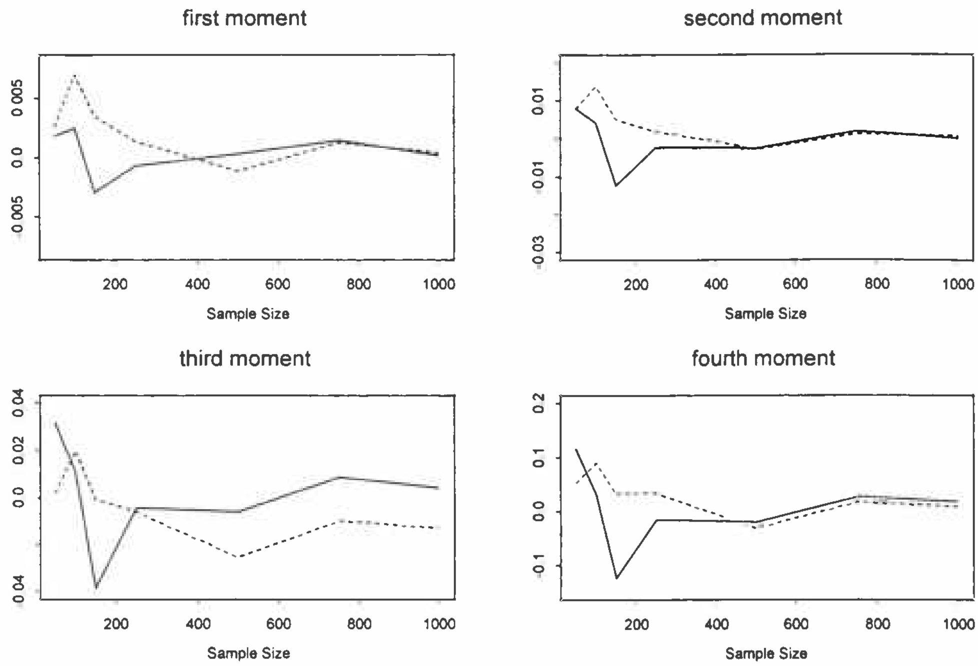
Figure 3.5.3

Plots of $\hat{\mu}_i - \mu_i, i = 1,2,3,4,$ versus Sample Size

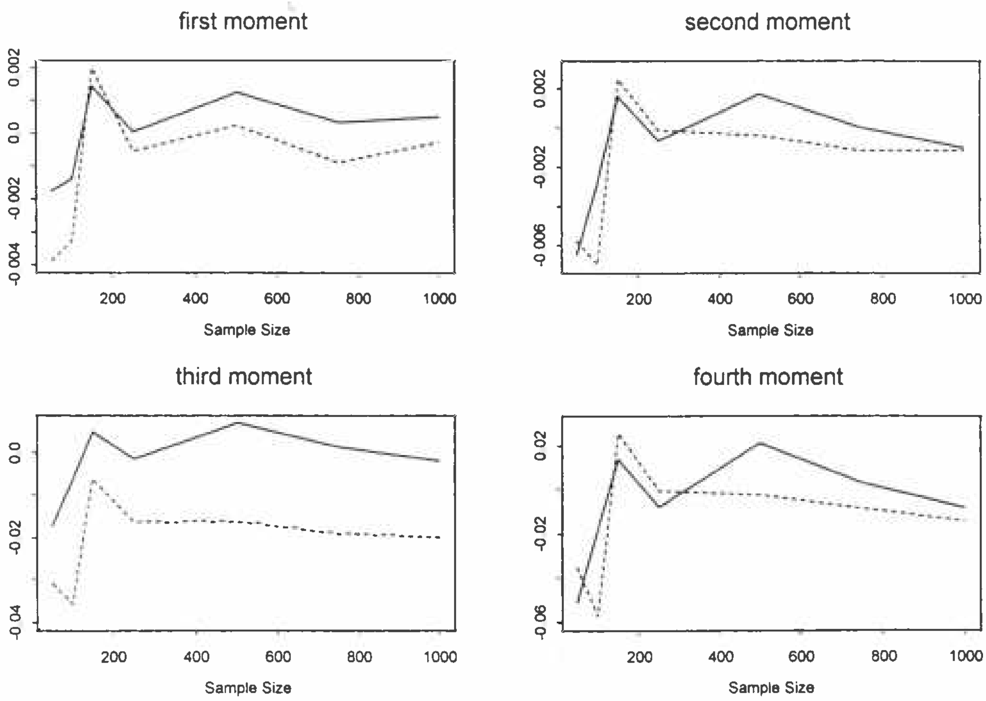
Given set of moments $\mu_1 = 0, \mu_2 = 1, \mu_3 = 2, \mu_4 = 6$.

Solid line: Pearson's algorithm. Dashed line: Devroye's algorithm

1000 Iterations



5000 iterations



https://doi.org/10.26219/heal.aueb.5743

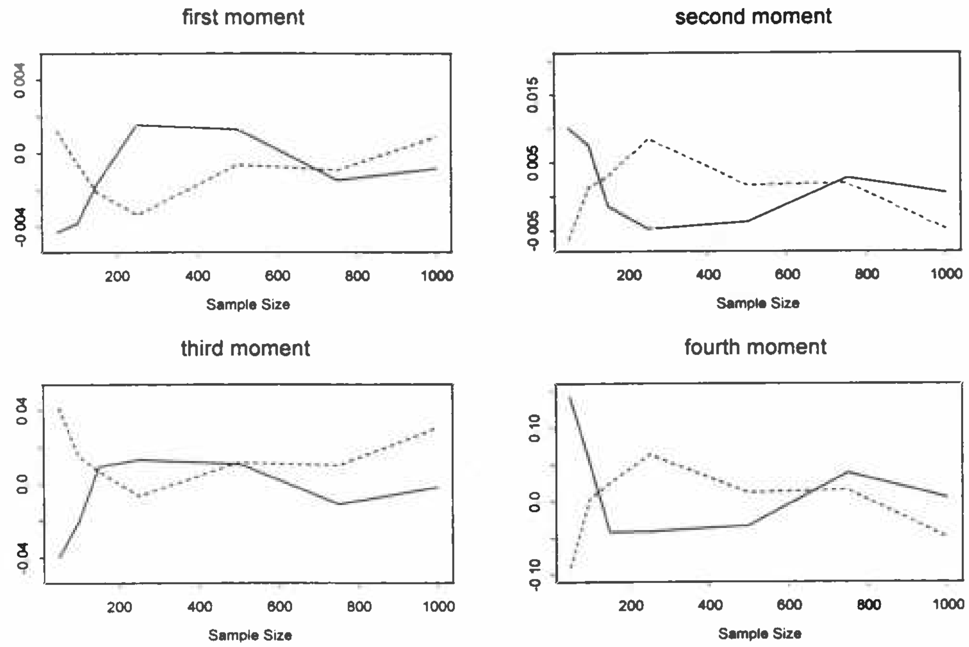
Figure 3.5.4

Plots of $\hat{\mu}_i - \mu_i, i = 1,2,3,4$, versus Sample Size

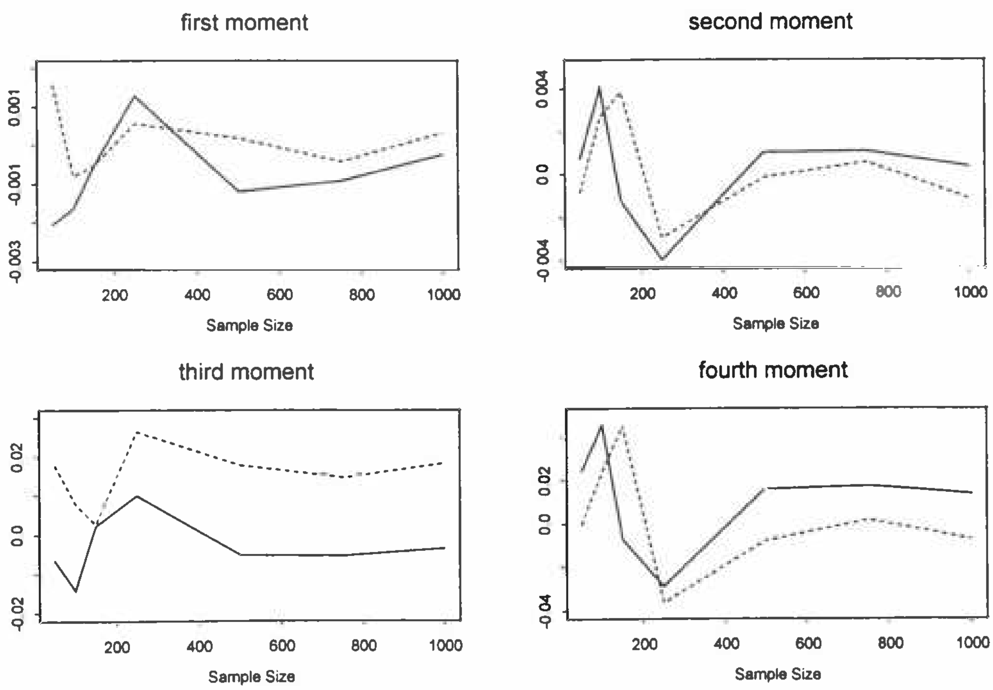
Given set of moments $\mu\ = 0, \mu\ = 1, \mu_3 = -2, \mu_4 = 6$.

Solid line: Pearson's algorithm. Dashed line: Devroye's algorithm

1000 Iterations



5000 iterations



https://doi.org/10.26219/heal.aueb.5743

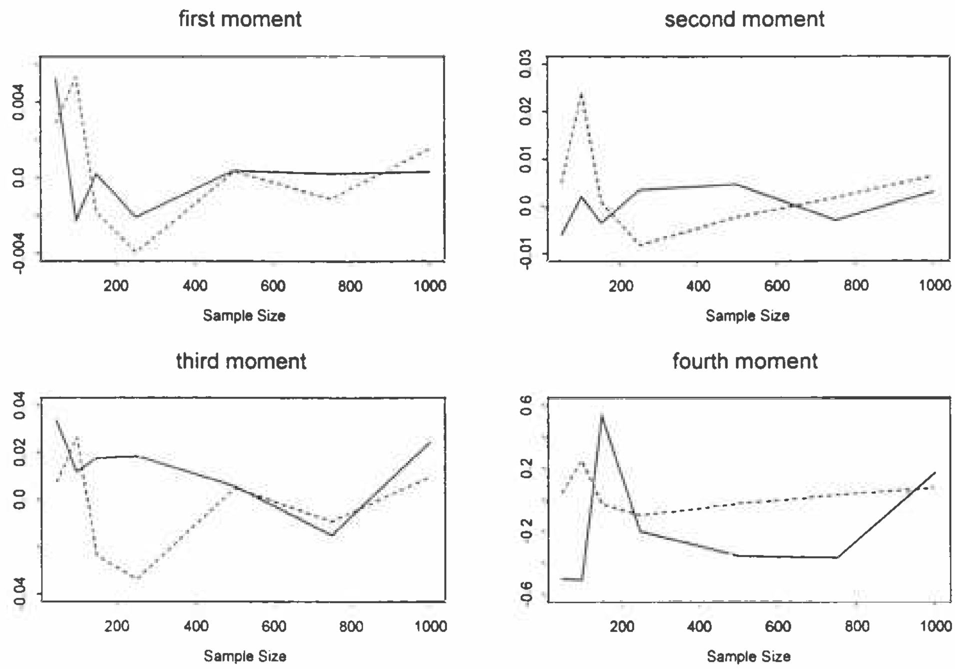
Figure 3.5.5

Plots of $\hat{\mu}_i - \mu_i, i = 1,2,3,4,$ versus Sample Size

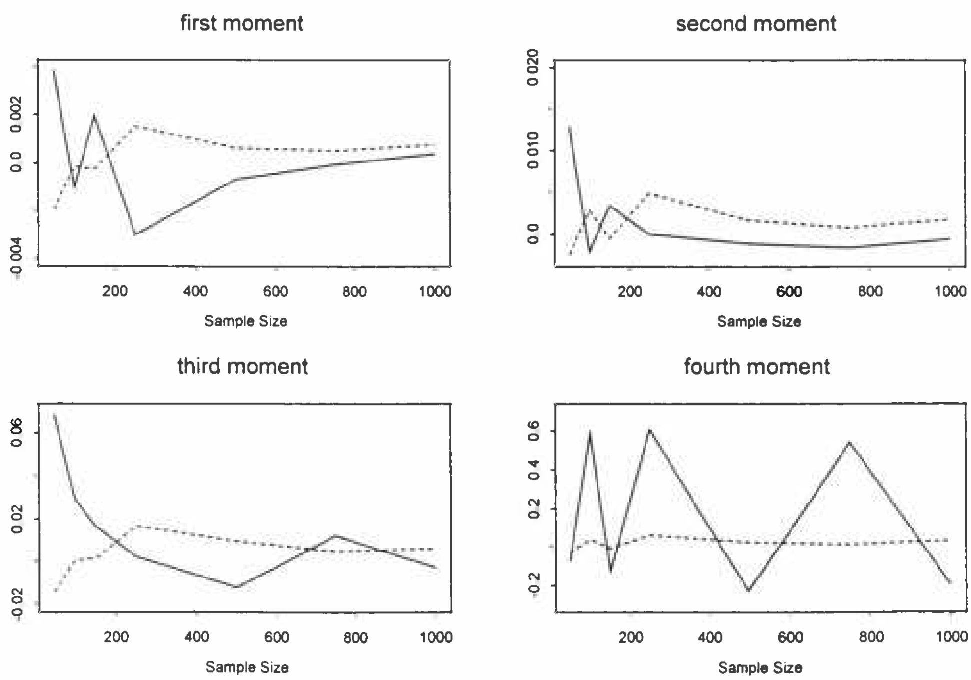
Given set of moments $\mu = 0, \mu = 1, \mu_3 = 1, \mu_4 = 8$.

Solid line: Pearson's algorithm. Dashed line: Devroye's algorithm

1000 iterations



5000 iterations



https://doi.org/10.26219/heal.aueb.5743

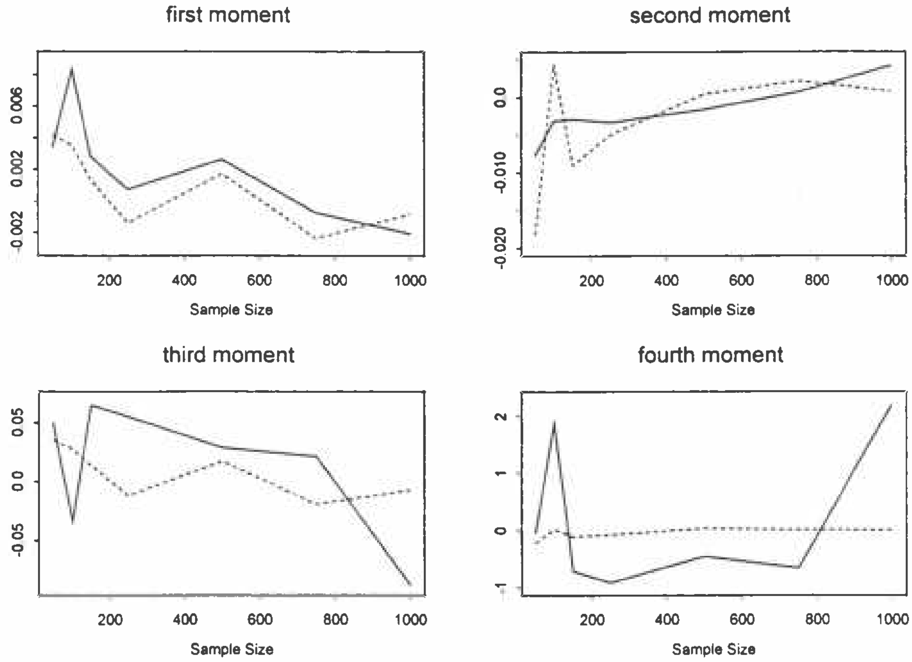
Figure 3.5.6

Plots of $\hat{\mu}_i - \mu_i, i = 1,2,3,4,$ versus Sample Size

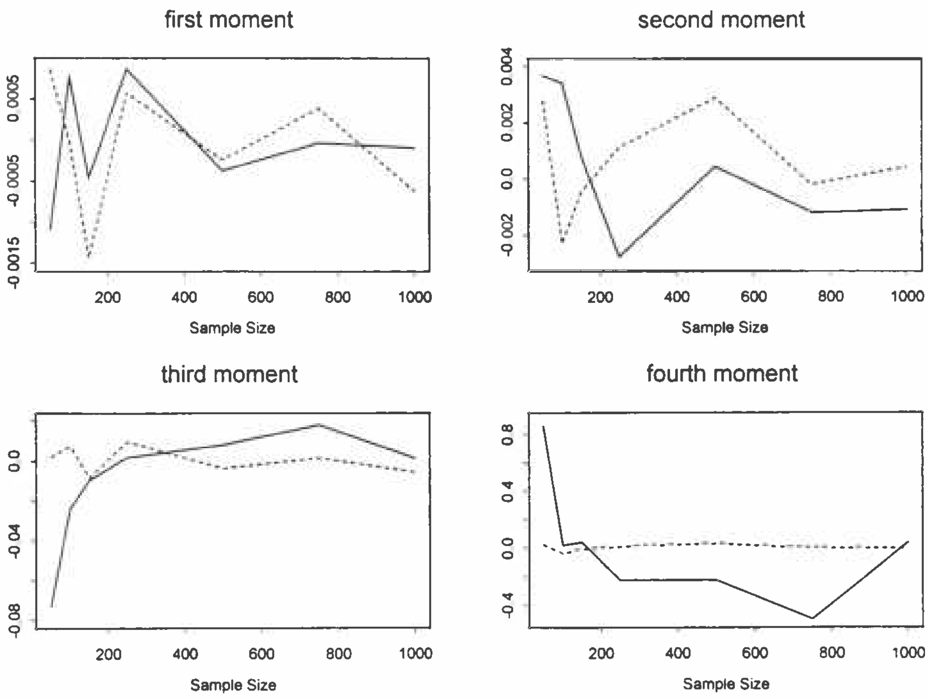
Given set of moments $\mu_1 = 0, \mu_2 = 1, \mu_3 = -1, \mu \; = 8$.

Solid line: Pearson's algorithm. Dashed line: Devroye's algorithm

1000 Iterations



5000 iterations



https://doi.org/10.26219/heal.aueb.5743

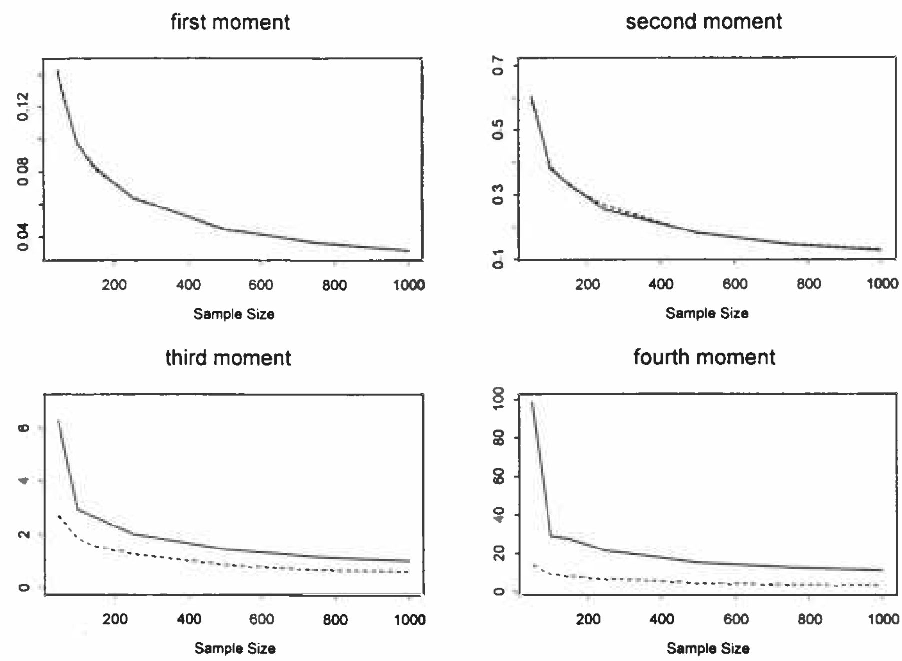
Figure 3.6.1

Plots of moment's Standard Deviation versus Sample Size

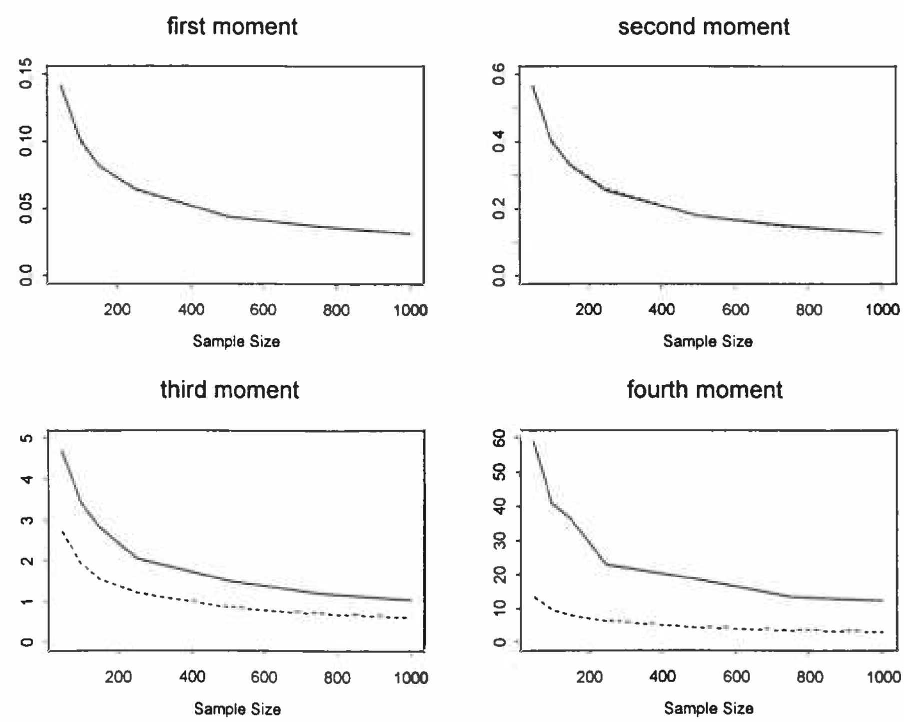
Given set of moments $\mu = 0, \mu_2 = 1, \mu_3 = 3, \mu_4 = 17$.

Solid line: Pearson's algorithm. Dashed line: Devroye's algorithm

1000 Iterations



5000 iterations



https://doi.org/10.26219/heal.aueb.5743

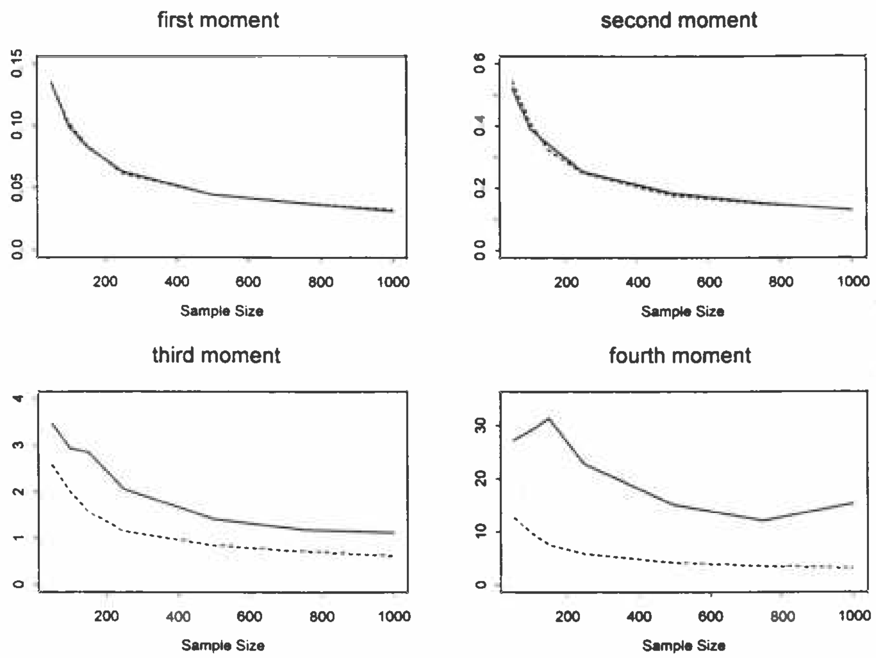
Figure 3.6.2

Plots of moment's Standard Deviation versus Sample Size

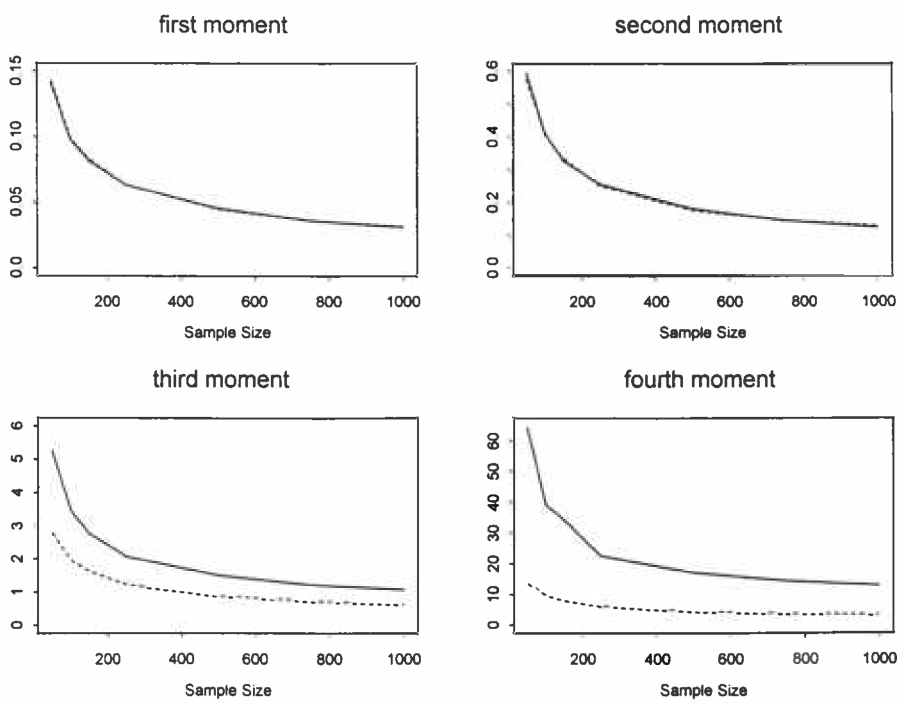
Given set of moments $\mu\ = 0, \mu\ = 1, \mu_3 = -3, \mu_4 = 17$.

Solid line: Pearson's algorithm. Dashed line: Devroye's algorithm

1000 Iterations



5000 iterations



https://doi.org/10.26219/heal.aueb.5743

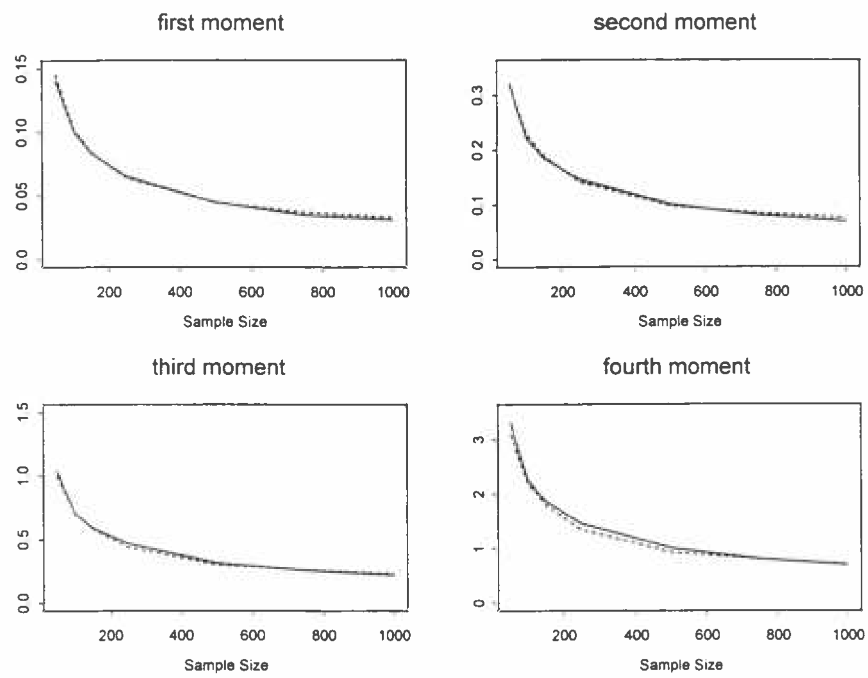
Figure 3.6.3

Plots of moment's Standard Deviation versus Sample Size

Given set of moments $\mu_1 = 0, \mu_2 = 1, \mu_3 = 2, \mu_4 = 6$.

Solid line: Pearson's algorithm. Dashed line: Devroye's algorithm

1000 Iterations



5000 iterations

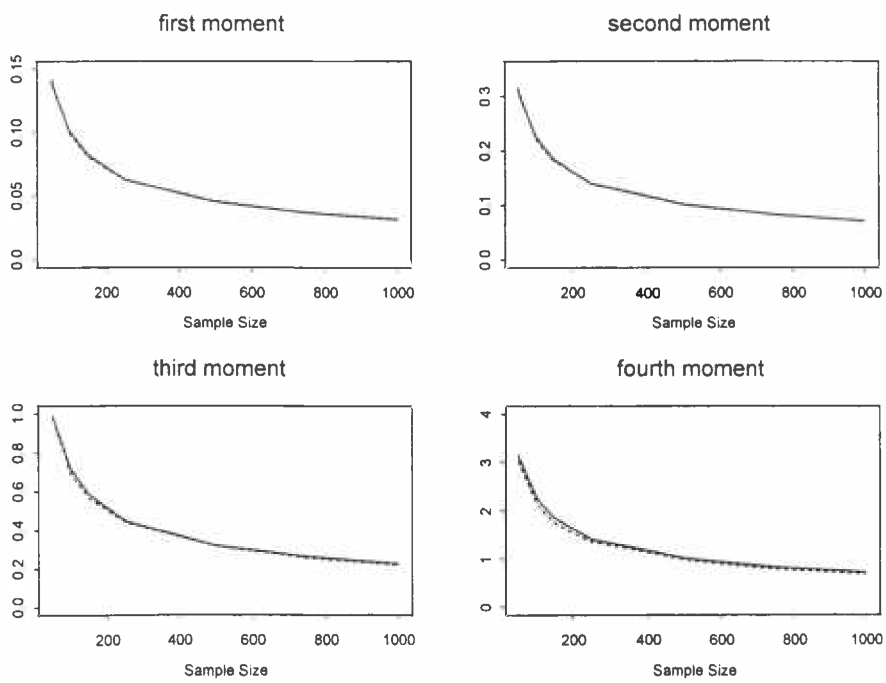


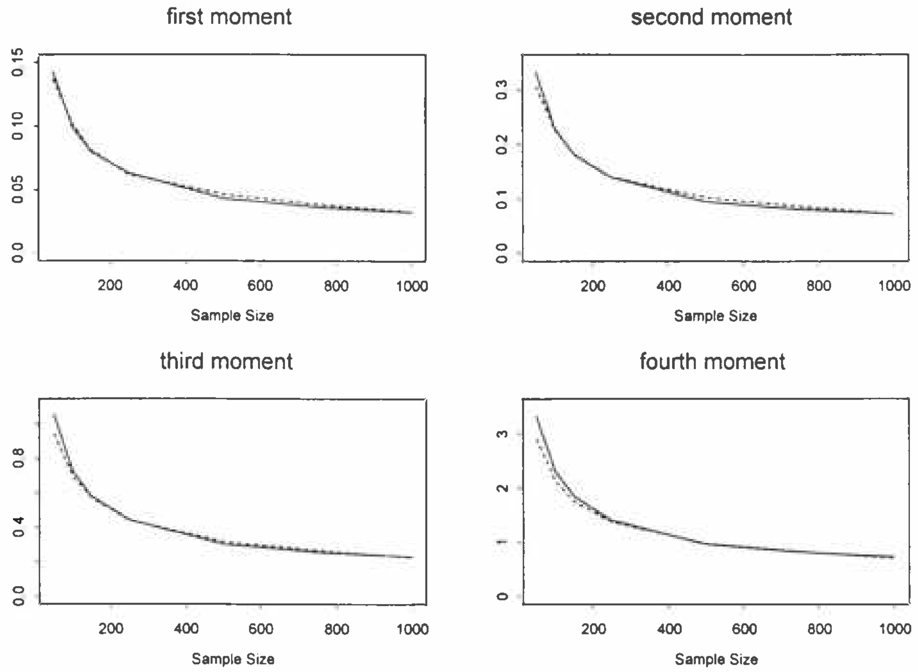
Figure 3.6.4

Plots of moment's Standard Deviation versus Sample Size

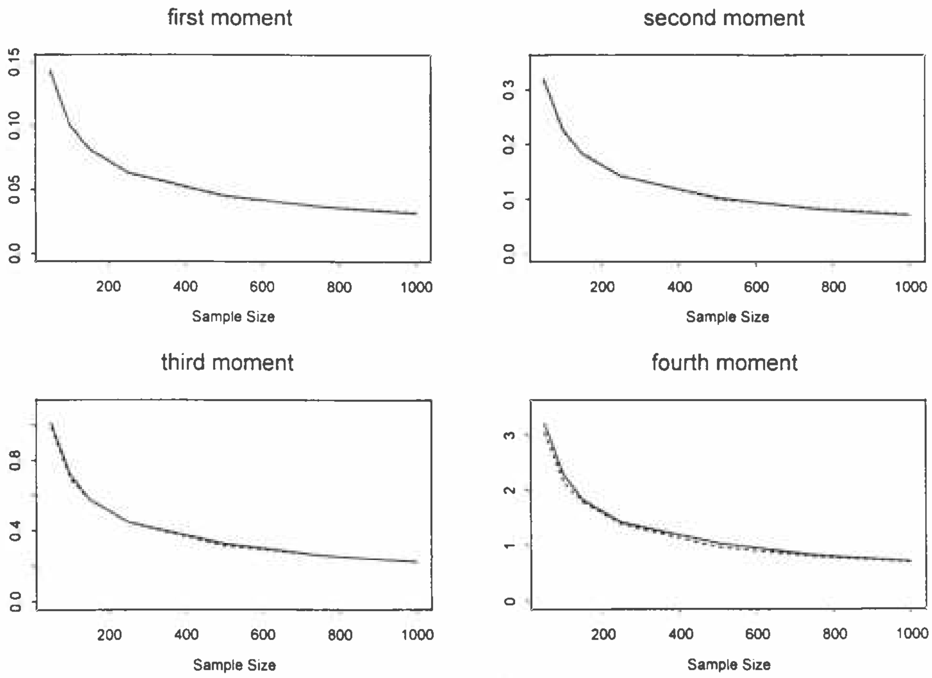
Given set of moments $\mu = 0, \mu = 1, \mu_3 = -2, \mu_4 = 6$.

Solid line: Pearson's algorithm. Dashed line: Devroye's algorithm

1000 Iterations



5000 iterations



https://doi.org/10.26219/heal.aueb.5743

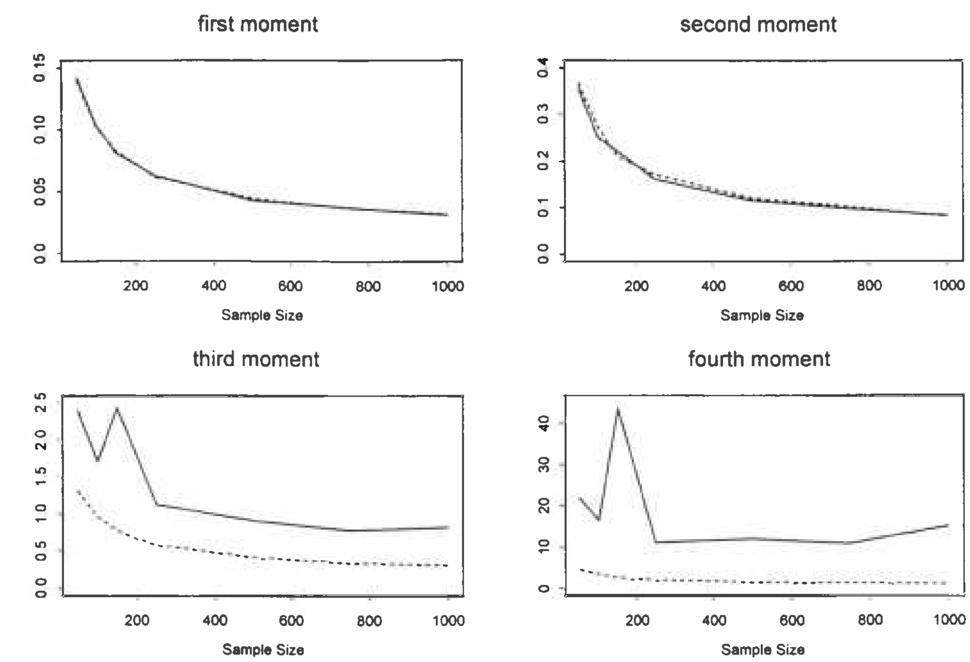
Figure 3.6.5

Plots of moment's Standard Deviation versus Sample Size

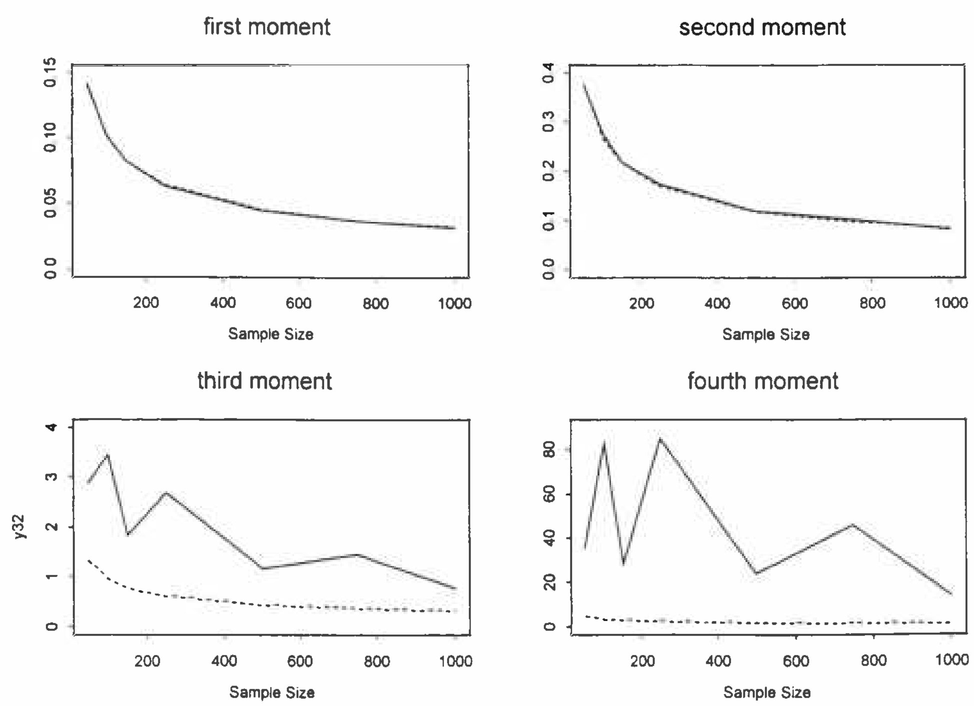
Given set of moments $\mu = 0, \mu = 1, \mu_3 = 1, \mu_4 = 8$.

Solid line: Pearson's algorithm. Dashed line: Devroye's algorithm

1000 Iterations



5000 iterations



https://doi.org/10.26219/heal.aueb.5743

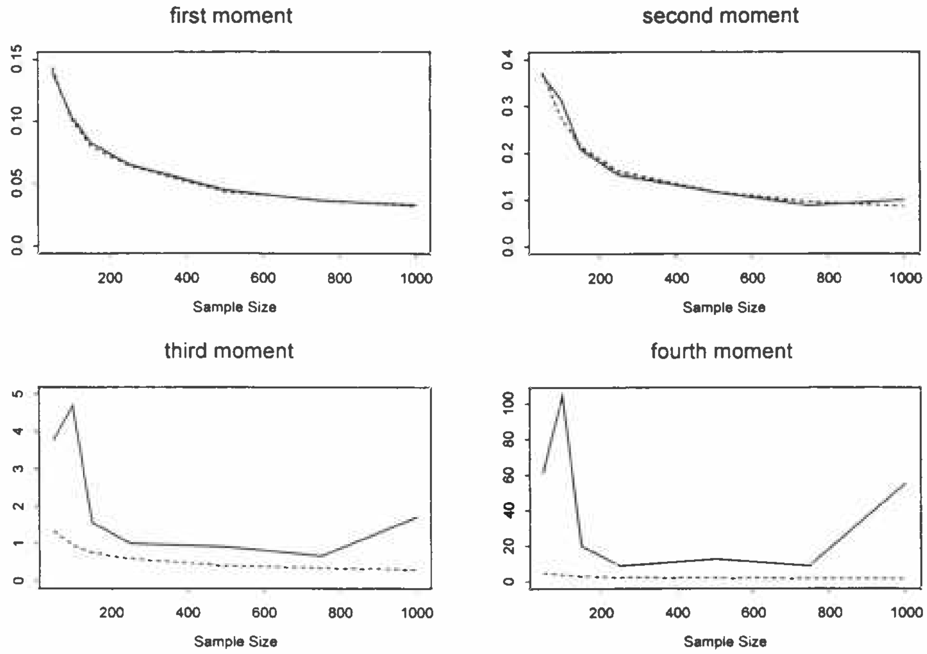
Figure 3.6.6

Plots of moment's Standard Deviation versus Sample Size

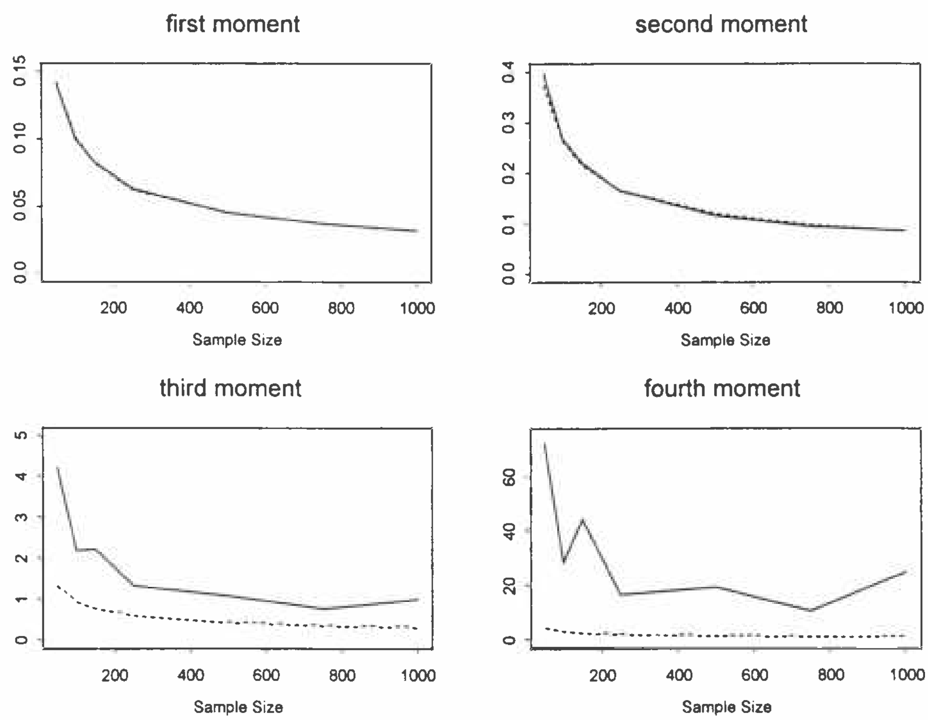
Given set of moments $\mu_1 = 0, \mu_2 = 1, \mu_3 = -1, \mu_4 = 8$.

Solid line: Pearson's algorithm. Dashed line: Devroye's algorithm

1000 Iterations



5000 iterations



https://doi.org/10.26219/heal.aueb.5743

Chapter 4

Generating Random Samples Matching the Moments of the Generalized Lambda Distribution.

The generalized lambda distribution, denoted as GLD (λ_1, λ_2, λ_3, λ_4), is a four parameter distribution that has been used for fitting density curves to real data sets (Ramberg et al., 1979). The GLD (λ_1, λ_2, λ_3, λ_4) was developed by Ramberg and Schmeiser as a generalisation of the Tukey's one-parameter lambda distribution.

The GLD (λ_1, λ_2, λ_3, λ_4) covers a wide range of distribution shapes and for this reason was used for fitting probability density shapes to data sets. Moreover the random variable generation from the GLD (λ_1, λ_2, λ_3, λ_4), as we demonstrate in section 4.2, is trivial through the inversion method. For these reasons, we consider that it is worth trying to construct an algorithm for generating random variables matching the first four moments.

In the first section we give an introduction to the GLD (λ_1, λ_2, λ_3, λ_4) while in section 4.2 we develop an algorithm for generating random variables from the GLD (λ_1, λ_2, λ_3, λ_4) given the first four moments. In the final section, we present the results of a simulation study.

https://doi.org/10.26219/heal.aueb.5743

4.1 Introduction to the GLD ($\lambda_1, \lambda_2, \lambda_3, \lambda_4$)

In 1960, Tukey proposed the one parameter lambda distribution in terms of the following inverse distribution function (Devroye, 1986)

$$F^{-1}(y) = \frac{1}{\lambda}(y^{\lambda} - (1-y)^{\lambda}), \text{ where } 0 \le y \le 1 \text{ and } \lambda \in R.$$

Tukey's one-parameter lambda distribution was later generalized, first to a two parameter distribution and finally to a four parameter distribution denoted as GLD ($\lambda_1, \lambda_2, \lambda_3, \lambda_4$).

The GLD ($\lambda_1, \lambda_2, \lambda_3, \lambda_4$) is most easily described by its inverse distribution function, i.e. quantile function

$$F^{-1}(y) = \lambda_1 + \frac{y^{\lambda_3} - (1-y)^{\lambda_4}}{\lambda_2}, \text{ where } 0 \le y \le 1. \quad (4.1)$$

By computing $\frac{1}{F^{-1}(y)'}$, where $F^{-1}(y)'$ is the first derivative of (4.1), with respect to y, we can compute the density $f(x)$, where y and x are related via the equality $x = F^{-1}(y)$. Hence, the density $f(x)$ corresponding to (4.1) is $f(x) = \frac{\lambda_2}{\lambda_3 x^{\lambda_3 - 1} + \lambda_4 (1-x)^{\lambda_4 - 1}}$.

However, in order for the GLD ($\lambda_1, \lambda_2, \lambda_3, \lambda_4$), specified by (4.1), to be a valid distribution, constraints must be imposed to the values of the parameters $\lambda_i, i = 1 \ldots 4$. Harian, Dudewicz, have shown that the GLD ($\lambda_1, \lambda_2, \lambda_3, \lambda_4$) is a valid distribution if and only if,

$$\frac{\lambda_2}{\lambda_3 y^{\lambda_3 - 1} + \lambda_4 (1-y)^{\lambda_4 - 1}} \ge 0 \quad (4.2)$$

for all y in [0,1], i.e. when $\lambda_2, \lambda_3, \lambda_4$ have the same sign. Alternatively, relation (4.1) describes a valid distribution if and only if,

$\lambda_3 y^{\lambda_3 - 1} + \lambda_4 (1-y)^{\lambda_4 - 1}$ has the same sign, positive or negative, for all $y \in [0,1]$ as long as λ_2 takes the same sign. Using these facts they show that the GLD(λ_1, λ_2, λ_3, λ_4) is valid if $\lambda_3, \lambda_4 > 1, \lambda_3, \lambda_4 < -1, \lambda_3 > 1$ and $\lambda_4 < -1$, $\lambda_3 < -1$and $\lambda \ > 1$.

However, Karian, Dudewicz, McDonald (1996) have expanded the region of the values of λ_3, λ_4 for which the Generalized Lambda is a valid distribution. Particularly, they show that the GLD(λ_1, λ_2, λ_3, λ_4) is not valid if and only if,

$$\frac{(1-\lambda_3)^{1-\lambda_3}}{(\lambda_4 - \lambda_3)^{\lambda_4 - \lambda_3}} (\lambda_4 - 1)^{\lambda_4 - 1} \geq -\frac{\lambda_3}{\lambda_4}. \qquad (4.3)$$

Condition (4.3) relaxes the restrictions that $\lambda_3 > 1$ and $\lambda \ < -1$, $\lambda_3 < -1$ and $\lambda_4 > 1$ and allows λ_4 and λ_3 to take values < -1. Apart from the validity of the GLD (λ_1, λ_2, λ_3, λ_4) a further restriction which guarantees that the first four moments of the GLD (λ_1, λ_2, λ_3, λ_4) exist is $\lambda_3, \lambda_4 \geq -\frac{1}{4}$.

Dudewicz and Karian have constructed tables for estimating the parameters of the GLD (λ_1, λ_2, λ_3, λ_4). Given the sample skewness and kurtosis these tables provide estimates of the parameters λ_1, λ_2, λ_3, λ_4. Ozturk and Dale (1985) have developed a least square algorithm for estimating the parameters of the Generalized Lambda distribution. Our approach here is more simple in the sense that we use an iterative algorithm in order to solve the non-linear system of equations that we get by applying the method of moments for estimating the parameters of the GLD (λ_1, λ_2, λ_3, λ_4).

https://doi.org/10.26219/heal.aueb.5743

4.2 Estimating the Parameters of the GLD(λ_1, λ_2, λ_3, λ_4)

Our objective is to develop an algorithm for random variable generation from the Generalized Lambda distribution matching the first four given moments. Therefore, we have to match the moments of the GLD (λ_1, λ_2, λ_3, λ_4) with the corresponding, given the first four moments. As it is generally well known, according to the method of moments we equate the sample moments with the moments of the GLD (λ_1, λ_2, λ_3, λ_4).

The moments of the Generalized Lambda distribution are given from the following formula (see in Karian et al., 1996):

$$\mu_1 = \lambda_1 + \frac{A}{\lambda_2} \tag{4.4}$$

$$\mu_2 = \frac{(B-A)^2}{\lambda_2^2} \tag{4.5}$$

$$\mu_3 = \frac{(C-3AB+2A^3)}{\lambda_2^3} \tag{4.6}$$

$$\mu_4 = \frac{D-4AC+6A^2B-3A^4}{\lambda_2^4}, \tag{4.7}$$

where the quantities A, B, C and D are defined as

$$A = \frac{1}{1+\lambda_3} - \frac{1}{1+\lambda_4}$$

$$B = \frac{1}{1+2\lambda_3} + \frac{1}{1+2\lambda_4} - 2B(1+\lambda_3, 1+\lambda_4)$$

$$C = \frac{1}{1+3\lambda_3} - \frac{1}{1+3\lambda_4} - 3B(1+2\lambda_3, 1+\lambda_4) + 3B(1+\lambda_3, 1+2\lambda_4)$$

$$D = \frac{1}{1+4\lambda_3} + \frac{1}{1+4\lambda_4} - 4B(1+3\lambda_3, 1+\lambda_4) + 6B(1+2\lambda_3, 1+2\lambda_4) - 4B(1+\lambda_3, 1+3C\lambda_4)$$

https://doi.org/10.26219/heal.aueb.5743

and $B(a,b)$ is the Beta function.

If $\mu_1, \mu_2, \mu_3, \mu_4$ are the first four given moments, then by equating the corresponding moments we can estimate the parameters of the GLD (λ_1, λ_2, λ_3, λ_4). By equating the moments of GLD (λ_1, λ_2, λ_3, λ_4) given in relations (4.4) to (4.7), with the corresponding sample moments we obtain a system of non-linear equations that can not be analytically solved. Therefore we try to solve this system by a numerical method. We must note that generally there are no good methods for solving systems of non-linear equations. The most known algorithm, is the Newton - Raphson Method for non-linear systems of equations which generally gives a very efficient means of converging to a root, assuming you have a sufficiently good initial guess. For details about the Newton - Raphson Method see Press et al. (1992).

Next, we give a short description of this method. By applying the method of moments, we get four non-linear functions $f_i(\lambda_1,...\lambda_4)=0$, $i=1...4$ that need to be solved simultaneously with respect to the parameters $\lambda_j, j=1,...4$. If we denote the vector of values λ_j by λ then each function $f_i(\lambda_1,\lambda_2,\lambda_3,\lambda_4), i=1,2,3,4$ can be expanded in Taylor series as

$$f_i(\lambda+\delta\lambda)= f_i(\lambda)+\sum_{j=1}^{4}\frac{\partial f_i}{\partial \lambda_j}\delta\lambda_j + O(\delta\lambda^2).$$

By neglecting terms of order $\delta\lambda^2$ and higher and by setting $f_i(\lambda+\delta\lambda)=0$ we obtain a system of non - linear equations for the quantities $\delta\lambda_j$ that move each function closer to zero simultaneously, namely

$$\sum_{j=1}^{4} a_{ij}\delta\lambda_j = \beta_i \qquad (4.8)$$

https://doi.org/10.26219/heal.aueb.5743

where $a_{ij} = \frac{\partial f_i}{\partial \lambda_j}$ is a 4x4 matrix whose elements are the partial derivatives of f_i with respect to λ_j,where $i, j = 1,..4$, and $\beta_i = -f_i(\lambda + \delta\lambda)$ is a 4x1 vector.

The matrix equation (4.8) can be solved by LU decomposition(see, for example, Press et al., 1992). At each iteration the quantity $\delta\lambda_j$ is added to the solution vector

$$\lambda_j^{new} = \lambda_j^{old} + \delta\lambda_j, \qquad j=1,\ 2,\ 3,\ 4.$$

The iterative procedure stops if either the sum of the magnitudes of the function f_i is less than a convergence criterion value or the sum of the absolute values of $\delta\lambda_j$ is less than a second convergence criterion value. Note that the Newton - Raphson method does not always converge to a solution.

After experimentation with the algorithm we notice that if a solution exists then it converges relatively fast. How fast the algorithm converges, depends on the initial values of the parameters. The trial of the algorithm with different initial values is a useful tool for checking whether it converges to the solution of the non-linear system or not. Since λ_1 is a location parameter we expect the estimate $\hat{\lambda}_1$ to be close to the mean μ_1. By setting μ_1 as the initial value of $\hat{\lambda}_1$ the algorithm may be slightly faster.

Multiple solution for $\lambda_1, \lambda_2, \lambda_3, \lambda_4$ is another problem that arises when we use this algorithm. Varying the initial values of the parameters and plotting the generated values can help us to distinguish such a case.

Once we have obtained a solution for the system of the non-linear equations, the next step is to simulate random variables from the GLD (λ_1, λ_2, λ_3, λ_4). This can easily be achieved by the inversion method (for details see Morgan (1984), Ripley (1987), Devroye (1986)).

According to the inversion method, if we wish to simulate a continuous random variable, X, with distribution function $F(x) = \Pr(X \leq x)$, where its inverse function $F^{-1}(u)$ is well-defined for $0 \leq u \leq 1$, and U is a Uniform(0,1) random variable then the random variable $X = F^{-1}(U)$ has the required distribution. Hence, a simple algorithm for generating random variables from the GLD (λ_1, λ_2, λ_3, λ_4) is:

Generating random variables from the GLD (λ_1, λ_2, λ_3, λ_4)

- Generate a random variable U from Uniform(0,1)
- Set $X = F^{-1}(U) = \lambda_1 + \dfrac{U^{\lambda_3} - (1-U)^{\lambda_4}}{\lambda_2}$, then $X \sim \mathrm{GLD}(\lambda_1, \lambda_2, \lambda_3, \lambda_4$

https://doi.org/10.26219/heal.aueb.5743

4.3 Simulation Comparison

In this section, we apply the algorithm, described in section 4.2, to given sets of the Generalized lambda distribution moments. Moreover, we apply, to the same moments, the Devroye's and Pearson's algorithm respectively. Table 4.1 gives the parameters of the Generalized Lambda Distribution from which we compute the first four moments. Given the moments of table 4.1 we apply the algorithm of section 4.2. The values of the parameters and the resulting moments were chosen such that the iterative algorithm of section 4.2 converges to a solution and the condition (2.4) is satisfied by the moments respectively. The appropriate member of the Pearson's system, which corresponds to all sets of the given moments in table 4.1, is the Type I distribution.

In tables 4.2.1 to 4.2.6 we report the simulation results of the applied algorithms. We follow the same procedure as in section 3.5. For different sample sizes and for replications of size 1000 and 5000 respectively we generate random variables matching the given moments of GLD (λ_1, λ_2, λ_3, λ_4). First from tables 4.2.1 to 4.2.6, we notice that for all the algorithms the simulated matched moments are close to the given moment sets. Especially, for sample size 1000 the simulated matched moments are very close to the given moments. Second we notice that the estimated standard deviations of the computed, at each replication, moments decreases as the sample size increases. Moreover, the standard deviations of the third and fourth moment for the Generalized Lambda distribution algorithm are a little lower than the corresponding standard deviations of the Pearson's and Devroye's algorithm whereas for the two first moments they are almost the same for both algorithms. Finally we see that for 5000 replications we obtain almost the same results as we do for 1000 replications. Thus, we can say that 1000 replications are adequate for our procedure.

It is worth stating that even if the given moments come from the Generalized Lambda distribution the Pearson's and Devroye's algorithm match the given moments well.

Table 4.1
Parameters and the corresponding moments of the Generalized Lambda distribution

First Set	Parameters	λ_1	λ_2	λ_3	λ_4
		-1.0	0.3	0.04	0. 5
	Moments	μ_1	μ_2	μ_3	μ_4
		-0.017	0.79	0.25	1.43
Second Set	Parameters	λ	λ_2	λ_3	λ_4
		2.0	0.2	0.7	0.025
	Moments	μ_1	μ_2	μ_3	μ_4
		0.04	1.98	-0.6	7.95
Third Set	Parameters	λ	λ_2	λ_3	λ_4
		1.2	0.2	0.3	0.06
	Moments	μ	μ	μ_3	μ_4
		0.33	1.28	-0.7	4.55

https://doi.org/10.26219/heal.aueb.5743

Table 4.2.1

Simulation results with moments $\mu_1=-0.017, \mu_2=0.79, \mu_3=0.25, \mu_4=1.43$.

Standard Deviations are given in brackets

	Replications	n_i	μ_1	μ_2	μ_3	μ_4
Generalized Lambda distribution Algorithm	1000	n_1=100	-0.0157 (0.091)	0.7815 (0.089)	0.2403 (0.107)	1.4044 (0.274)
		n_2=150	-0.0183 (0.0719)	0.7830 (0.0707)	0.2449 (0.0902)	1.4142 (0.2278)
		n_3=250	-0.0181 (0.0552)	0.7868 (0.0573)	0.2481 (0.0689)	1.4285 (0.1838)
		n_4=500	-0.0175 (0.0393)	0.7904 (0.0408)	0.2475 (0.0503)	1.4332 (0.1316)
		n_5=750	-0.0178 (0.0325)	0.7882 (0.0341)	0.2476 (0.0417)	1.4281 (0.1083)
		n_6=1000	-0.0154 (0.0295)	0.7902 (0.0284)	0.2481 (0.0352)	1.4329 (0.0889)
Pearson's Algorithm	1000	n_1=100	-0.0173 (0.0902)	0.7885 (0.0903)	0.24493 (0.1965)	1.42053 (0.3233)
		n_2=150	-0.0176 (0.0721)	0.7881 (0.0712)	0.2457 (0.1530)	1.4226 (0.2529)
		n_3=250	-0.0178 (0.0549)	0.78825 (0.0569)	0.24595 (0.1176)	1.42192 (0.2001)
		n_4=500	-0.0175 (0.0394)	0.78917 (0.0404)	0.24716 (0.0846)	1.42391 (0.1446)
		n_5=750	-0.0175 (0.0318)	0.78906 (0.0337)	0.24805 (0.0695)	1.42494 (0.1209)
		n_6=1000	-0.0164 (0.0284)	0.78892 (0.0289)	0.24955 (0.0620)	1.42510 (0.1032)
Devroye's Algorithm	1000	n_1=100	-0.0144 (0.0907)	0.7939 (0.0904)	0.2533 (0.1924)	1.4372 (0.2951)
		n_2=150	-0.0188 (0.0729)	0.7922 (0.0696)	0.2489 (0.1536)	1.4337 (0.2276)
		n_3=250	-0.0166 (0.0567)	0.7858 (0.0575)	0.2469 (0.1194)	1.4194 (0.1904)
		n_4=500	-0.0148 (0.0397)	0.7897 (0.0397)	0.2552 (0.0819)	1.4309 (0.1308)
		n_5=750	-0.0179 (0.0323)	0.7904 (0.0336)	0.2485 (0.0686)	1.4316 (0.1106)
		n_6=1000	-0.0165 (0.0277)	0.7902 (0.0283)	0.2514 (0.0592)	1.4316 (0.0908)

https://doi.org/10.26219/heal.aueb.5743

Table 4.2.2

Simulation results with moments $\mu_1 = -0.017, \mu_2 = 0.79, \mu_3 = 0.25, \mu_4 = 1.43.$

Standard Deviations are given in brackets

	Replications	n_i	μ_1	μ_2	μ_3	μ_4
Generalized Lambda distribution Algorithm	5000	n_1=100	-0.0178 (0.0883)	0.7828 (0.0895)	0.2428 (0.1111)	1.4158 (0.2849)
		n_2=150	-0.0179 (0.0719)	0.7850 (0.0738)	0.2441 (0.0916)	1.4206 (0.2343)
		n_3=250	-0.0159 (0.0563)	0.7875 (0.0574)	0.2458 (0.0723)	1.4261 (0.1822)
		n_4=500	-0.0170 (0.0396)	0.7897 (0.0397)	0.2482 (0.0493)	1.4323 (0.1265)
		n_5=750	-0.0169 (0.0319)	0.7892 (0.0322)	0.2471 (0.0408)	1.4301 (0.1023)
		n_6=1000	-0.0173 (0.0281)	0.7897 (0.0278)	0.2483 (0.0359)	1.4320 (0.0887)
Pearson's Algorithm	5000	n_1=100	-0.0175 (0.0885)	0.7892 (0.0906)	0.2472 (0.1928)	1.4239 (0.3266)
		n_2=150	-0.0175 (0.0723)	0.7891 (0.0749)	0.2481 (0.1579)	1.4249 (0.2694)
		n_3=250	-0.0161 (0.0556)	0.7896 (0.0574)	0.2504 (0.1217)	1.4272 (0.2060)
		n_4=500	-0.0165 (0.0399)	0.7894 (0.0404)	0.2499 (0.0859)	1.4273 (0.1446)
		n_5=750	-0.0171 (0.0326)	0.7899 (0.0333)	0.2498 (0.0706)	1.4294 (0.1195)
		n_6=1000	-0.0169 (0.0278)	0.7899 (0.0284)	0.2502 (0.0603)	1.4299 (0.1021)
Devroye's Algorithm	5000	n_1=100	-0.0153 (0.0873)	0.7905 (0.0899)	0.2529 (0.1875)	1.4323 (0.2941)
		n_2=150	-0.0173 (0.0733)	0.7893 (0.0740)	0.2481 (0.1554)	1.4281 (0.2429)
		n_3=250	-0.0163 (0.0561)	0.7904 (0.0575)	0.2516 (0.1187)	1.4305 (0.1872)
		n_4=500	-0.0169 (0.0402)	0.7898 (0.0401)	0.2499 (0.0852)	1.4296 (0.1310)
		n_5=750	-0.0166 (0.0328)	0.7903 (0.0327)	0.2513 (0.0691)	1.4317 (0.1068)
		n_6=1000	-0.0173 (0.0279)	0.7905 (0.0288)	0.2500 (0.0592)	1.4312 (0.0945)

https://doi.org/10.26219/heal.aueb.5743

Table 4.2.3

Simulation results with moments $\mu_1 = 0.04, \mu_2 = 1.98, \mu_3 = -0.6, \mu_4 = 7.95.$

Standard Deviations are given in brackets

Algorithm	Replications	n_i	μ_1	μ	μ	μ
Generalized Lambda distribution Algorithm	1000	n_1=100	0.0384 (0.1399)	1.9626 (0.2006)	-0.5759 (0.3996)	7.8889 (1.3029)
		n_2=150	0.0384 (0.1139)	1.9687 (0.1652)	-0.5839 (0.3252)	7.9109 (1.0713)
		n_3=250	0.0415 (0.0892)	1.9721 (0.1284)	-0.5911 (0.2558)	7.9308 (0.8299)
		n_4=500	0.0395 (0.0627)	1.9789 (0.0893)	-0.5927 (0.1795)	7.9582 (0.5865)
		n_5=750	0.0401 (0.0507)	1.9787 (0.0732)	-0.5987 (0.1457)	7.9596 (0.4716)
		n_6=1000	0.0393 (0.0445)	1.9797 (0.0619)	-0.5941 (0.1289)	7.9586 (0.4009)
Pearson's Algorithm	1000	n_1=100	0.0386 (0.1417)	1.9768 (0.2056)	-0.6083 (0.6559)	7.9373 (1.4627)
		n_2=150	0.0394 (0.1147)	1.9778 (0.1655)	-0.6050 (0.5267)	7.9395 (1.1666)
		n_3=250	0.0395 (0.0886)	1.9773 (0.1274)	-0.6043 (0.4093)	7.9349 (0.8981)
		n_4=500	0.0395 (0.0628)	1.9774 (0.0888)	-0.6011 (0.2902)	7.9353 (0.6290)
		n_5=750	0.0397 (0.0506)	1.9789 (0.0734)	-0.6010 (0.2338)	7.9465 (0.5178)
		n_6=1000	0.0398 (0.0439)	1.9787 (0.0635)	-0.6001 (0.2037)	7.9437 (0.4474)
Devroye's Algorithm	1000	n_1=100	0.0388 (0.1371)	1.9853 (0.2001)	-0.6066 (0.6299)	7.9978 (1.3833)
		n_2=150	0.0408 (0.1119)	1.9777 (0.1619)	-0.5958 (0.5138)	7.9338 (1.1142)
		n_3=250	0.0392 (0.0900)	1.9793 (0.1281)	-0.6016 (0.4094)	7.9517 (0.8914)
		n_4=500	0.0399 (0.0624)	1.9797 (0.0893)	-0.5997 (0.2857)	7.9490 (0.6183)
		n_5=750	0.0398 (0.0520)	1.9799 (0.0729)	-0.5987 (0.2371)	7.9528 (0.5087)
		n_6=1000	0.0405 (0.0442)	1.9816 (0.0638)	-0.6013 (0.2002)	7.9612 (0.4433)

Table 4.2.4

Simulation results with moments $\mu_1 = 0.04, \mu_2 = 1.98, \mu_3 = -0.6, \mu_4 = 7.95$.

Standard Deviations are given in brackets

Algorithm	Replications	n_i	μ_1	μ	μ	μ
Generalized Lambda distribution Algorithm	5000	n_1=100	0.0417 (0.1431)	1.9595 (0.2028)	-0.5790 (0.3958)	7.8399 (1.3053)
		n_2=150	0.0384 (0.1149)	1.9640 (0.1628)	-0.5762 (0.3230)	7.8708 (1.0547)
		n_3=250	0.0384 (0.0875)	1.9719 (0.1274)	-0.5861 (0.2427)	7.9379 (0.8254)
		n_4=500	0.0385 (0.0619)	1.9822 (0.0891)	-0.5942 (0.1794)	7.9726 (0.5894)
		n_5=750	0.03881 (0.0518)	1.9761 (0.0761)	-0.5923 (0.1497)	7.9412 (0.4931)
		n_6=1000	0.0422 (0.0471)	1.9785 (0.0659)	-0.5976 (0.1283)	7.9581 (0.4236)
Pearson's Algorithm	5000	n_1=100	0.0388 (0.1404)	1.9820 (0.2020)	-0.6187 (0.6585)	7.9674 (1.4540)
		n_2=150	0.0388 (0.1174)	1.9752 (0.1665)	-0.6001 (0.5427)	7.9152 (1.1858)
		n_3=250	0.0368 (0.0895)	1.9750 (0.1271)	-0.6088 (0.4114)	7.9214 (0.8920)
		n_4=500	0.0386 (0.6115)	1.9768 (0.0899)	-0.6083 (0.2921)	7.9373 (0.6425)
		n_5=750	0.0394 (0.0500)	1.9778 (0.0723)	-0.6050 (0.2352)	7.9395 (0.5123)
		n_6=1000	0.03902 (0.0445)	1.9769 (0.0617)	-0.6075 (0.2084)	7.9341 (0.4416)
Devroye's Algorithm	5000	n_1=100	0.0342 (0.1422)	1.9792 (0.2032)	-0.6349 (0.6515)	7.9517 (1.4133)
		n_2=150	0.0393 (0.1156)	1.9727 (0.1676)	-0.5989 (0.5161)	7.9141 (1.1491)
		n_3=250	0.0428 (0.0833)	1.9702 (0.1328)	-0.5853 (0.3938)	7.8689 (0.9225)
		n_4=500	0.0379 (0.0623)	1.9803 (0.0904)	-0.6037 (0.2899)	7.9497 (0.6249)
		n_5=750	0.0389 (0.0511)	1.9803 (0.0749)	-0.6022 (0.2362)	7.9559 (0.5142)
		n_6=1000	0.0415 (0.0456)	1.9783 (0.0645)	-0.5983 (0.2083)	7.9363 (0.4524)

https://doi.org/10.26219/heal.aueb.5743

Table 4.2.5

Simulation results with moments $\mu_1 = 0.33, \mu_2 = 1.28, \mu_3 = -0.7, \mu_4 = 4.55.$

Standard Deviations are given in brackets

Algorithm	Replications	n_i	μ_1	μ	μ	μ
Generalized Lambda distribution Algorithm	1000	n_1=100	0.3317 (0.1158)	1.2628 (0.1701)	-0.6508 (0.2961)	4.3929 (1.1905)
		n_2=150	0.3289 (0.0932)	1.2684 (0.1394)	-0.6627 (0.2559)	4.4663 (1.0144)
		n_3=250	0.3288 (0.0705)	1.2762 (0.1081)	-0.6772 (0.1881)	4.5444 (0.7766)
		n_4=500	0.3286 (0.0497)	1.2819 (0.0762)	-0.6821 (0.1415)	4.559 (0.5702)
		n_5=750	0.3290 (0.0415)	1.2776 (0.0647)	-0.6834 (0.1174)	4.5476 (0.4736)
		n_6=1000	0.3318 (0.0381)	1.2786 (0.0567)	-0.6836 (0.1006)	4.5473 (0.4108)
Pearson's Algorithm	1000	n_1=100	0.3303 (0.1078)	1.2835 (0.1726)	-0.7005 (0.5727)	4.5928 (1.7641)
		n_2=150	0.3278 (0.0880)	1.2859 (0.1408)	-0.7081 (0.4661)	4.6091 (1.4372)
		n_3=250	0.3293 (0.0683)	1.2841 (0.1116)	-0.7036 (0.3668)	4.5950 (1.1327)
		n_4=500	0.3289 (0.0488)	1.2822 (0.0768)	-0.6985 (0.2561)	4.5769 (0.7793)
		n_5=750	0.3283 (0.0411)	1.2836 (0.0636)	-0.7039 (0.2158)	4.5942 (0.6550)
		n_6=1000	0.3280 (0.0348)	1.2840 (0.0553)	-0.7065 (0.1849)	4.6029 (0.5667)
Devroye's Algorithm	1000	n_1=100	0.3288 (0.1087)	1.2805 (0.1663)	-0.6745 (0.5005)	4.5498 (1.2618)
		n_2=150	0.3313 (0.0875)	1.2838 (0.1349)	-0.6784 (0.4069)	4.5894 (1.0178)
		n_3=250	0.3295 (0.0685)	1.2822 (0.1039)	-0.6922 (0.3178)	4.6032 (0.7939)
		n_4=500	0.3299 (0.0481)	1.2819 (0.0753)	-0.6837 (0.2249)	4.5847 (0.5739)
		n_5=750	0.3311 (0.0385)	1.2786 (0.0605)	-0.6734 (0.1834)	4.5607 (0.4649)
		n_6=1000	0.3298 (0.0342)	1.2791 (0.0536)	-0.6795 (0.1600)	4.5710 (0.4049)

https://doi.org/10.26219/heal.aueb.5743

Table 4.2.6

Simulation results with moments $\mu_1 = 0.33, \mu_2 = 1.28, \mu_3 = -0.7, \mu_4 = 4.55.$

Standard Deviations are given in brackets

Algorithm	Replications	n_i	μ_1	μ	μ	μ
Generalized Lambda distribution Algorithm	5000	n_1=100	0.3287	1.2691	-0.6629	4.4833
			(0.1125)	(0.1698)	(0.3062)	(1.2328)
		n_2=150	0.3286	1.2731	-0.6724	4.5107
			(0.0917)	(0.1405)	(0.2565)	(1.0291)
		n_3=250	0.33312	1.2747	-0.6783	4.5267
			(0.0717)	(0.1088)	(0.1985)	(0.7882)
		n_4=500	0.3296	1.2793	-0.6801	4.5459
			(0.0505)	(0.0768)	(0.1401)	(0.5676)
		n_5=750	0.3299	1.2793	-0.6846	4.5509
			(0.0409)	(0.0622)	(0.1129)	(0.4506)
		n_6=1000	0.3295	1.2798	-0.6829	4.5540
			(0.0357)	(0.0526)	(0.1030)	(0.3878)
Pearson's Algorithm	5000	n_1=100	0.3289	1.2822	-0.6985	4.5769
			(0.1093)	(0.1725)	(0.5733)	(1.7427)
		n_2=150	0.3283	1.2836	-0.7039	4.5942
			(0.0889)	(0.1428)	(0.4738)	(1.4481)
		n_3=250	0.3281	1.2835	-0.7039	4.5937
			(0.0698)	(0.1101)	(0.3702)	(1.1272)
		n_4=500	0.3297	1.2814	-0.6949	4.5679
			(0.0489)	(0.0765)	(0.2573)	(0.7825)
		n_5=750	0.3298	1.2804	-0.6920	4.5565
			(0.0397)	(0.0625)	(0.2085)	(0.6341)
		n_6=1000	0.3301	1.2802	-0.6907	4.5531
			(0.0344)	(0.0539)	(0.1804)	(0.5499)
Devroye's Algorithm	5000	n_1=100	0.3357	1.2735	-0.6582	4.5240
			(0.1059)	(0.1677)	(0.4995)	(1.2746)
		n_2=150	0.3335	1.2749	-0.6624	4.5295
			(0.0877)	(0.1395)	(0.4111)	(1.0436)
		n_3=250	0.3320	1.2734	-0.6629	4.5223
			(0.0668)	(0.1122)	(0.3131)	(0.8137)
		n_4=500	0.3337	1.2769	-0.6674	4.5462
			(0.0483)	(0.0744)	(0.2237)	(0.5592)
		n_5=750	0.3332	1.2769	-0.6695	4.5560
			(0.0408)	(0.0633)	(0.1936)	(0.4922)
		n_6=1000	0.3320	1.2785	-0.6759	4.5668
			(0.0339)	(0.0547)	(0.1604)	(0.4115)

Chapter 5

A Real Data Application

In order to demonstrate the applicability of the algorithms, presented in chapter 3, we apply them to a real data set. In the following sections we give a description of the data and the methodology used for this data set. Finally we present the results of this application.

5.1 The Data

The data were provided from a large Greek insurance company and concern the claim amounts made by car drivers in the company during the year 1996. In particular, they consist of the number of the claims that had been made by a car driver who was exposed to risk during the year 1996, and the total amount of these claims. The data can be thought of as a set of random sums since the number of the car accidents of a driver could be seen as a random quantity and the total claim amount as a sum of the partial claims made by the car drivers.

Let us suppose that the individual claims of the insured drivers are not available and that only the total claims amount and the number of car accidents per driver respectively are known. In such a setting the researchers are interested in the distribution of the partial claims. Our objective is, by computing the sample moments of the total claims and assuming a statistical model for the number of accidents, to estimate the

https://doi.org/10.26219/heal.aueb.5743

moments of the partial claims and then to draw a sample from these moments. The generated sample from the moments of the partial claims could be considered as an approximation of their distribution.

5.2 Methodology

Let us consider the random sums of the form, $R = S_N = X_1 + X_2 + \ldots + X_N$ where $X_1, X_2, \ldots, X_N$ are independent random variables and N is a non-negative integer valued random variable, which is independent of X_i, $i = 1 \ldots N$.

In this application, we assume that N, the number of car accidents per driver, is a Poisson random variable with parameter λ, i.e. N~Poisson(λ). Moreover, we must clarify that we do not make any assumptions about the functional form of the distribution of the random variables X_i (the partial claims made by the car drivers). We only assume that they are identical distributed.

Recall that a generalized distribution is defined as the random sum $S_N = \sum_{i=1}^{N} X_i$, where N is a random variable from a discrete distribution. In our case, since we assume that N is a Poisson random variable, the distribution of the random sums S_N is the Generalized Poisson. Douglas (1980) calls the distribution of S_N a randomly stopped distribution, particularly in our case as Poisson stopped, since the number of terms in the sum, S_N, is determined by a probability distribution.

https://doi.org/10.26219/heal.aueb.5743

The probability generating function, $\phi(t)$, of a continuous random variable X with distribution function $F(x)$ is defined as $\phi(t) = E(t^X) = \int_x t^x dF(x)$ while for a discrete random variable x is defined as $\phi(t) = \sum_x t^x P(X = x)$, where t is such that $\int t^x dF(x) < \infty$ and $\sum_x t^x P(X = x) < \infty$ respectively. Let $h(t)$ be the probability generating function of the components X of the random sum, S_N, and $g(t)$ the probability generating function of the random variable N. Then for the probability generating function of the random sums S_N, denoted as $q(t)$, it holds that:

$$
\begin{aligned}
q(t) = E(t^{S_N}) = \\
E_N\left[E_X(t^{S_N}|N)\right] = \\
E_N\left[E_X(t^{X_1} \cdot t^{X_2} \cdots t^{X_N})|N\right] = \\
E_N\left[E_X(t^{X_1})E_X(t^{X_2}) \cdots E_X(t^{X_N})\right] = \\
E_N\left[(h(t))^N\right] = g(h(t)). \qquad (5.1)
\end{aligned}
$$

Hence, the probability generating function of the random sums S_N is a composite function of the probability generating function of the random variables N and X_i respectively.

Under the assumption that the number of accidents N is a Poisson random variable, the probability generating function of the variable N is $g(t) = \exp(\lambda(t-1))$. Thus, from (5.1) we obtain for the probability generating function of the distribution of the sum S ,

$$q(t) = g(h(t)) = \exp[-\lambda + \lambda h(t)] = \exp[\lambda(h(t) - 1)] \qquad (5.2)$$

where $h(t)$ is the probability generating function of the random variable X_i.

It is known that given the probability generating function, $\phi(t)$, of a random variable X we can compute the $r-th, r=1,2...,$ factorial moment of the random variable X, denoted as $E(X_{(r)})$, by the relation $E(X_{(r)})=\phi^{(r)}(t)|_{t=1}$, where $\phi^{(r)}(t)$ is the $r-th, r=1,2...,$ derivative of $\phi(t)$ with respect to t evaluated at $t=1$ (see in Kati, 1966).

Let us denote the factorial moments of S_N, N and X_i as $E(S_{N(r)}), E(N_{(r)}), E(X_{(r)})$ respectively. Hence, by differentiating (5.2) with respect to t and for $r=1,...4$, we get for the probability generating function of the random sum S_N:

$$q^{(1)}(t)=g^{(1)}(h(t))h^{(1)}(t) \tag{5.3}$$

$$q^{(2)}(t)=g^{(2)}(h(t))(h^{(1)}(t))^2+g^{(1)}(h(t))h^{(2)}(t) \tag{5.4}$$

$$q^{(3)}(t)=g^{(3)}(h(t))(h^{(1)}(t))^3+3g^{(2)}(h(t))h^{(1)}(t)h^{(2)}(t) \tag{5.5}$$

$$\begin{aligned} q^{(4)}(t)=&\,g^{(4)}(h(t))(h^{(1)}(t))^4+6g^{(3)}(h(t))h^{(1)}(t)^2h^{(2)}(t)\\ &+3g^{(2)}(h(t))h^{(2)}(t)^2+4g^{(2)}(h(t))h^{(1)}(t)h^{(3)}(t)\\ &+g^{(1)}(h(t))h^{(4)}(t). \end{aligned} \tag{5.6}$$

If $t=1$ then $h(t)=1$ and from formulae (5.3), (5.4), (5.5), (5.6) we get for the factorial moments of the sum S_N:

$$E(S_{N(1)})=E(N_{(1)})E(X_{(1)}) \tag{5.7}$$

$$E(S_{N(2)})=E(N_{(2)})E(X_{(1)})^2+E(N_{(1)})E(X_{(2)}) \tag{5.8}$$

$$E(S_{N(3)})=E(N_{(3)})E(X_{(1)})^3+3E(N_{(2)})E(X_{(2)})E(X_{(1)})+E(N_{(1)})E(X_{(2)}) \tag{5.9}$$

$$\begin{aligned} E(S_{N(4)})=&\,E(N_{(4)})E(X_{(1)})^4+6E(N_{(3)})E(X_{(1)})^2E(X_{(2)})+\\ &3E(N_{(2)})E(X_{(2)})^2+4E(N_{(2)})E(X_{(1)})E(X_{(3)})+E(N_{(1)})E(X_{(4)}). \end{aligned} \tag{5.10}$$

https://doi.org/10.26219/heal.aueb.5743

Under the assumption that N is a Poisson random variable with parameter λ, the factorial moments of the random variable N are $E(N_{(1)})=\lambda$, $E(N_{(2)})=\lambda^2$, $E(N_{(3)})=\lambda^3$, $E(N_{(4)})=\lambda^4$ from which we compute the simple moments of the Poisson variable N, given by the following relations,

$$\mu_1 = E(N) = \lambda \tag{5.11}$$

$$\mu_2 = E(N^2) = \lambda^2 + \lambda \tag{5.12}$$

$$\mu_3 = E(N^3) = \lambda^3 + 3\lambda^2 + \lambda \tag{5.13}$$

$$\mu_4 = E(N^4) = \lambda^4 + 6\lambda^3 + 7\lambda^2 + \lambda. \tag{5.14}$$

Formulae (5.7), (5.8), (5.9), (5.10) enable us to calculate the factorial moments of the X_i. After a lot of mathematical manipulations for computing the simple from the factorial moments and using the relations (5.11) to (5.14), we find that the first four simple moments of the partial claims are

$$E(X) = \frac{E(S_N)}{\lambda} \tag{5.15}$$

$$E(X^2) = \frac{E(S_N^2) - \lambda^2 E(X)^2}{\lambda} \tag{5.16}$$

$$E(X^3) = \frac{E(S_N^3) - \lambda^3 E(X)^3 - 3\lambda E(X)E(X^2)}{\lambda} \tag{5.17}$$

$$E(X^4) = \frac{E(S_N^4) - \lambda^4 E(X)^4 - 6\lambda^3 E(X^2)E(X)^2 - 4\lambda^2 E(X)E(X^3) - 3\lambda^2 (E(X^2))^2}{\lambda}, \tag{5.18}$$

where $E(S_N^i)$ is the $i-th$, $i=1...4$, simple moment of the total claims and λ the mean of the Poisson distribution. The $i-th$ moment of S_N is a polynomial of degree at most i with respect to the parameter λ. This is a property of the Natural Exponential Families with Quadratic Variance

https://doi.org/10.26219/heal.aueb.5743

Functions, NEF-QVF, (see in Morris, 1982), and enables us to analytically estimate the moments of the mixing distribution, in cases of mixture distributions.

It is clear that from formulae (5.15), (5.16), (5.17) and (5.18) we can compute the moments of the partial claims X_i and then apply the algorithms of chapter 3 for approximating their distribution.

5.3 Results of the Application

From the data given in Table 5.1 we compute the first four sample simple moments of the total claims. The sample moments, in million drachmas, are:

$$\hat{\mu}_1 = 0.0197 \quad \hat{\mu}_2 = 0.0215 \quad \hat{\mu}_3 = 0.1201 \quad \hat{\mu}_4 = 1.9542.$$

The total number of car drivers, who were exposed to car accident risk, during 1996, was 148410. The table 5.1 gives the observed distribution of the number of car accidents.

Table 5.1
Distribution of the Number of Car Accidents

Number of accidents	Frequency
0	132944
1	13054
2	1024
3	98
4	15
5	2

The estimated mean number of accidents is $\overline{N}$=0.1 with sample variance 0.12. Under our assumption for the distribution of the number of car accidents, $\overline{N}$=0.1 is the estimate of the Poisson parameter λ. Since the sample variance is not much larger than the sample mean it is plausible to assume a Poisson distribution for the number of car accidents.

Hence, by substituting the sample estimates of the moments of the total claims and of the parameter λ, to relations (5.15), (5.16), (5.17), (5.18) we can easily estimate the moments of the partial claims. Particularly, the estimated simple moments of the partial claims X_i, denoted as $\overline{\mu}_i$, i =1...4, are:

$$\overline{\mu}_1 = 0.197,\ \overline{\mu}_2=0.211,\ \overline{\mu}_3=1.188,\ \overline{\mu}_4=19.4525.$$

The next step is to generate random samples, given the estimated moments of the partial claims. Therefore, we apply both algorithms of chapter 3 to the estimated moments of the partial claims. The appropriate member of the Pearson's system which corresponds to the given moments is the Type VI distribution.

In table 5.2 we present the matched simple moments of the individual claims after 1000 replications of sample size 10000.

https://doi.org/10.26219/heal.aueb.5743

Table 5.2
Results of the Real Data Application
Standard deviations are given in brackets

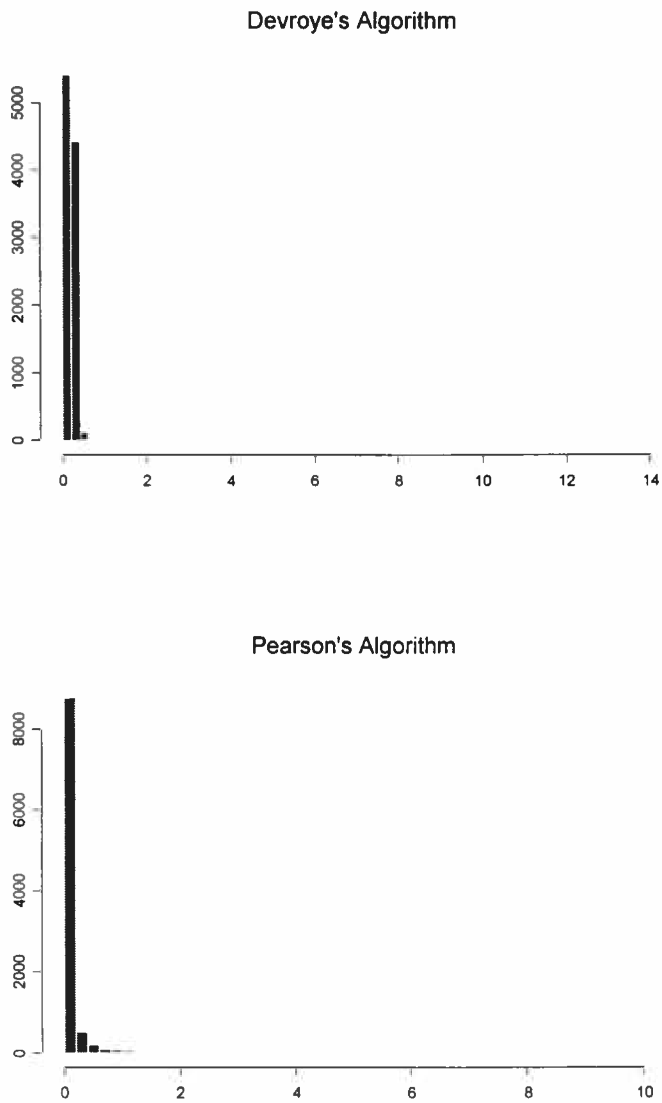
Devroye's algorithm			
$\overline{\mu}$	$\overline{\mu}_2$	$\overline{\mu}_3$	$\overline{\mu}_4$
0.19598	0.21652	1.19171	19.66755
(0.00416)	(0.04315)	(0.51188)	(6.10208)
Pearson's Algorithm			
$\overline{\mu}$	$\overline{\mu}_2$	$\overline{\mu}_3$	$\overline{\mu}_4$
0.19612	0.21652	1.16697	15.15890
(0.00423)	(0.03854)	(1.092019)	(46.74142)

We notice that the Devroye's algorithm matches all the given moments while the algorithm for the Pearson's system of distributions does not match the fourth moment well. The standard deviation of the third and fourth moment is much lower for the Devroye's algorithm than for the Pearson's algorithm. In figure 5.1 we give the histogram of the generated values from both algorithms. Additionally, in figure 5.2, we present the histogram of the actual partial claims made by the drivers during the year 1996. Finally, we note that our algorithm for generating random variables from the Generalized Lambda distribution does not converge to a solution, given the sample moments of the partial claims. Thus, it is not possible to apply the algorithm of section 4.2 to the estimated moments of partial claims.

https://doi.org/10.26219/heal.aueb.5743

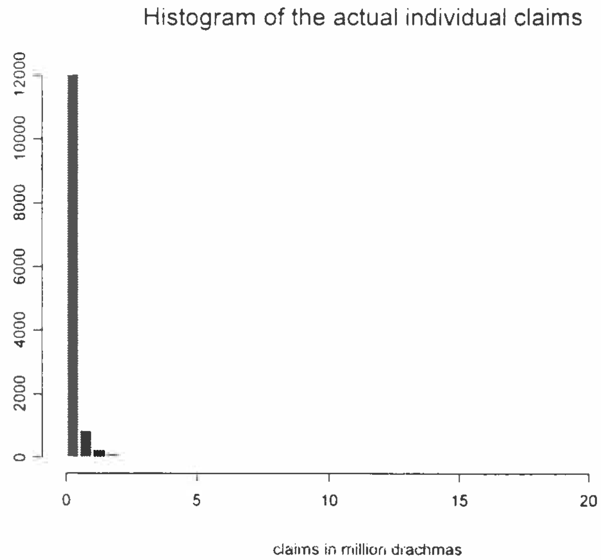
Figure 5.1

Histograms of the generated values given the moments of the individual claims



https://doi.org/10.26219/heal.aueb.5743

Figure 5.2
Histogram of the actual individual claims



Chapter 6

Concluding Remarks

After experimentation with the three algorithms and the simulation results presented in this thesis we give some useful concluding remarks concerning their performance and their use.

The Devroye's algorithm matches the given moments well and we suggest its use for large sample sizes. The main disadvantage of the Devroye's algorithm is that it does not produce smooth samples for small sample sizes.

In addition, it is restrictive because it can not be applied when the condition (2.4) is not satisfied by the given moments. Moreover, it is suitable for producing random samples only on the real line R. Futher research for the modification of the algorithm is needed in order for it to generate random variables over the range $[0,\infty)$ or $(-\infty,0]$.

The Pearson's system of distributions is suitable for moment matching since it does not require more than the first four moments in order for its parameters to be estimated. Moreover, it covers all the possible ranges to which a distribution function can be defined. These two facts urge us to use the Pearson's algorithm for generating samples from a distribution given the first four moments. The Pearson's algorithm produces smooth samples even for small sample sizes. In contrast to the Devroye's algorithm, the Pearson's algorithm is not so restrictive since it can be applied even if the condition (2.4) is violated for the values of the given moments. The disadvantage of the Pearson's algorithm is that it may

https://doi.org/10.26219/heal.aueb.5743

produce samples with high variability for the fourth moment. The Pearson's system of distributions can be generalized for multimodal density functions (see Cobb et all, 1983). Further research for generating random samples matching the moments of multimodal densities is needed. For this purpose, the multimodal generalization of the Pearson's system may possibly be used.

Finally, the algorithm for generating samples from the Generalized Lambda distribution performs well when the given moments come from the same distribution. As we have already stated in chapter 4, our algorithm does not always converge to a solution and as a result we can not estimate the parameters of the distribution. Therefore, we suggest its use only when the moments come from the Generalized Lambda distribution.

https://doi.org/10.26219/heal.aueb.5743

References

Cooper, J.D., Davis, S.A., Dono, N.R. (1965). Pearson Universal Distribution Generator (Purge), Proceedings of the 19th Annual Conference, **American Society for Quality Control**, 402 - 411.

Cobb, L., Koppstein, P., Chen N.H. (1983). Estimation and Moment Recursion Relations for Multimodal Distributions of the Exponential Family, **Journal of the American Statistical Association**, 78, 124-130.

Devroye, L. (1986). Non - Uniform Random Variate Generation, **Springer - Verlag**, New York.

Devroye, L. (1989). On Random Variate Generation When Only Moments or Fourier Coefficients are Known, **Mathematics and Computers in Simulation**, 31, 71-89.

Devroye, L. (1991). Algorithms for Generating Discrete Random Variables With a Given Generating Function or a Given Moment Sequence, **SIAM Journal of Scientific and Statistical Computing**, 12, 107 - 126.

Dharmadhikari, S., Joag-dev, K. (1987). Unimodality, Convexity, and Applications, **Academic Press Inc.**

Douglas, J.B. (1980). Analysis with Standard Contagious Distributions, **Statistical Distributions in Scientific Work Series, Vol 4, ICPH,** Maryland, USA.

Elderton, W.P., Johnson, N.L. (1969). Systems of Frequency Curves, London, **Cambridge University Press**.

Johnson, N.L., Kotz, S. (1970). Continuous Distribution - I, **John Willey,** U.K.

Katti, S.K. (1966). Interrelations Among Generalized Distributions and Their Components**, Biometrics,** 22, 44-52.

Karian, Z.A., Dudewicz, E.J., McDonald, P. (1996). The Extended Generalized Lambda Distribution System for Fitting Distributions to Data: History, Completion of Theory, Tables, Applications, The ' Final Word ' on Moments Fits, **Communications of Statistics - Simulation and Computation,** 25(3), 611-642.

Kendall, M.G. and Stuart, A. (1961). The advanced Theory of Statistics, **Hafner**, New York.

Morgan, B.J.T. (1984). Elements of Simulation, **Ghapman and Hall**, UK.

Morris, C.N. (1982). Natural Exponential Families with Quadratic Variance Functions**, Annals of Statistics,** 10, 65 -80.

Ozturk, A., Dale, R.F. (1985). Least Squares Estimation of the Parameters of the Generalized Lambda Distribution, **Technometrics,** 27, 81-84.

Pearson, E.S., Johnson, N.L., Burr, I.W. (1979). Comparisons of the percentage Points of Distributions with the Same First Four Moments, Chosen From Eight Different Systems of Frequency Curves, **Communications of Statistics - Simulation and Computation,** B8(3), 191-229.

Press, W.H., Teukolsky, S.A., Vetterling, W.T., Flannery, B.P. (1992). Numerical Recipes in Fortran, First Edition, **Cambridge University Press,** U.K.

Ramberg, J.S., Dudewicz, E.J., Tadikamalla, P.R., Mykytka, E.F.(1979). A probability Distribution and its Uses in Fitting Data, **Technometrics,** 21, 201-214.

Ripley, B.D. (1987). Stochastic Simulation, **John Wiley**.

Shohat, J.A., Tamarkin, J.D. (1943). The Problem of Moments, **Waverly Press.**

Springer, M.D. (1979). The Algebra of Random Variables, **John Wiley,** U.S.A.

Smith, A.F.M., Gelfand, A.E. (1992). Bayesian Statistics Without Tears: A Sampling Resampling Perspective, **The American Statistician,** 46, 84-88.

Titterington, D.M., Smith, A.F.M., Makov, U.E. (1985). Statistical Analysis with Mixtures Distributions, **John Wiley,** U.K.

https://doi.org/10.26219/heal.aueb.5743