ADVANCES IN THE APPLICATIONS OF DISTRIBUTION THEORY:
IMPROVEMENTS ON RANK-SIZE DISTRIBUTIONS AND IN SIGNAL PROCESSING

A thesis submitted to the University of Manchester
for the degree of Doctor of Philosophy
in the Faculty of Science and Engineering

2019

Martin W. Wiegand
Department of Mathematics

Contents

Word count 27980

List of Tables

List of Figures

The University of Manchester

Martin W. Wiegand
Doctor of Philosophy
Advances in the Applications of Distribution Theory:
Improvements on Rank-Size distributions and in Signal Processing
October 21, 2019

This thesis is a collection of projects focused on the advancement of the applications of distribution theory undertaken during the past years. It contains 11 chapters, with chapter 1 serving as an introduction to provide an overview of the topics addressed in the thesis and other projects that have been completed, but have not been included in this dissertation. Furthermore, a summary of other academic activities such as participation in international conferences will also be given. The last chapter will conclude the thesis with a summary of the results that have been obtained in the individual chapters, and will focus on future extensions of the work presented.

The topics included in this thesis fall into two sections. The first part addresses distribution theory applications in finance, linguistics and natural hazards, with a focus on rank-size distribution functions. We investigate the adequacy of various distributions on these relations and develop new multi-sectioned models. We close the section by introducing a software package that offers an accessible implementation of the multi-sectioned distribution model.

The second part is focused around advances in signal processing through aspects of distribution theory. We derive novel more efficient and robust approximation methods for common statistical functions, such as the multivariate Rayleigh distribution and the lognormal characteristic function. We furthermore derive a formulation of the general moments of the round-off errors of random variables of arbitrary distribution. Lastly, a software package which contains implementations for multivariate empirical densities is introduced and their performance is compared to previous algorithms.

Declaration

No portion of the work referred to in the thesis has been submitted in support of an application for another degree or qualification of this or any other university or other institute of learning.

Copyright Statement

Acknowledgements

Throughout the past three years I have received a great deal of assistance and guidance that aided me in the completion of this dissertation. First an foremost my thanks go to my supervisor and mentor Saralees Nadarajah. Not only did he advise me on how to be an independent researcher and statistician, but his aid in overcoming the many unforeseen obstacles along the way was invaluable. Through his caring nature I have learned what it means to be generous to both students and colleagues of all countries and backgrounds.

I am also grateful for the countless discussions with all the colleagues and staff of the School of Mathematics, which offered perspectives I could not always see myself. I would like to thank Philipp Jakob, Chico Rocha, Stefan Stein, Julia Tietjens and Mantė Žemaitytė for their companionship over the last years and beyond.

My deepest gratitude goes to my parents, who always encouraged and supported our endeavours, no matter what we did or where we chose to go in life. Thank you to my brother, who offered me advice and direction, as well as the occasional squabble from one ocean away.

My special thanks goes to Chen Qu, who has been an anchor point in my life and managed to keep calm in times I could not.

Chapter 1

Introduction

1.1 Motivation

This thesis is to be understood as a collection of all the results of the various research projects from the previous three years have provided. Each chapter represents the summary of one project, with the final outcome either having been published in a suitable, refereed academic journal or is still under review by an editorial board.

The aim of this thesis was largely motivated by the numerous application areas of distribution and estimation theory. By its very nature this field is heavily tied to the practical utilisation of theoretical methods. We have therefore tackled topics that originated from diverse fields such as engineering, signal processing, finance, economics, linguistics, medical sciences and cosmology. The appropriate construction of a distribution function is essential in encapsulating the behaviour of observed data into a numerical function. Quantitative measures based on these distributions, such as the statistical moments, or characteristic functions provide mathematical representations of data features.

However, while specialists of these application areas come across problems that heavily rely on an accurate model to describe data at hand, or to more realistically estimate a parameter, the manner in which these issue are approached is not always optimal. Many times simplistic models or historic methods are common practice, despite the fact that they violate framework conditions, are ill-fitted or rest on unrealistic assumptions. Our main goal was to rectify some of these misconceptions and provide novel, more accurate and justifiable methods.

Distribution theory itself can be divided into two main focus areas. Firstly, the study and analysis of real-life uni- or multivariate data sets. This includes the modelling of said data or the relationships between different aspects of the observations

through parametric distributions. Adequate distributions have to be chosen with respect to properties of the data at hand, such as skewness, asymptotic behaviour etc. and may require the development of new, or the modification of existing distributions to better suit the investigated data sets. Recommendations on the selection of distributions are based on the performance of candidate functions in relation to each other, assessed through various error measures and goodness of fit tests.

The second branch of distribution theory deals with the development of efficient approximation methods to compute distributions which lack a closed form, or properties of such distributions. Many distributions and their properties are necessary for application areas such as engineering, finance or physics among others. Yet many of the approximations that are in frequent use are restricted to certain parameter ranges, dimensions or evaluation points. Hence the development of more accurate, and less restrictive approximations opens up new opportunities in the application fields.

This thesis addresses both main fields of distributions theory, and discusses common pitfalls and problems using examples from various applications, and how to address these issues with novel or improved statistical tools.

The decision to compose this thesis in the alternative or journal format style has followed naturally as an immediate consequence of the highly diverse application fields we address. The traditional thesis format would not have offered the flexibility to adequately engage readers from different subject areas that may benefit from the findings we present. Due to their vast range of focus areas and scopes regarding the presentation of work, extracting relevant information from a less modular thesis would've been significantly complicated.

1.2 Contents and Structure

Chapters 2-5 build the first thematic block, which revolves around inverse ranking functions for the statistical analysis of population groups in various application areas, and the development of multi-sectioned distribution models.

In Chapter 2 we focus on a financial data set, which is provided annually by Forbes magazine. Titled the 'Forbes G2000' it contains 4 metrics of financial quantities of the 2000 largest corporations in the world. Motivated by previous literature, we introduced a new model to describe the relationship between the rank of a company on the list and the metric values. We propose an extended version of multi-sectioned distributions and discuss the choice of partial distributions and performance with regards

to the increased degrees of freedom.

The expression of thoughts and ideas between human beings relies primarily on the efficient use of language. Zipf's law is an established model to link frequency of word usage to their rank among other words of a language, reflecting expressiveness and efficiency of particular words and languages as a whole. In chapter 3 we propose new distribution functions of the Pareto-type family under consideration of tail behaviour of large data sets and show their favourable comparison to traditional models. As Zipf's law claims accuracy regardless of language, genre of literature and authorship (multi- or single-author literature) we carry out the analysis for various works of literature in more than 100 translations and in auto-generated texts to verify their general applicability.

To make the proposed multi-section models more accessible, we have created an R package that is freely available on the CRAN library, and offers an implementation of the standard statistical functions. The package we introduce in Chapter 4 is a modification of the code used to obtain the results in Chapter 2. Furthermore, the model has been extended to an arbitrary number of sections and features different modifications of the original multi-section distribution.

The second thematic block stretches across chapters 5 to 9 and focuses on the applications of distribution and estimation theory in engineering, with a particular focus on signal processing. We provide new approximation methods for distributions and distribution properties that surpass previous approaches' efficiency or limitations.

Round-off errors of random variables arise in various applications in signal processing, such as in signal filtering processes. Determining statistical properties of the round-off error of random variables of various distributions greatly enhances the understanding of the effect of filtering on a signal. Most prominently the measures of the mean value and the variance of said error are of importance. However, often other more general measures of higher order are of interest as well, such as skewness or kurtosis. In chapter 5 we generalise the moment computation to arbitrary degrees and give examples for commonly used distributions in the application field. We provide these measures for underlying random variables of both bounded and unbounded support. Numerical simulations verify the theoretical moment formulas.

The log-normal distribution is prominently featured in application fields that stretch from financial modelling to medical sciences and engineering or communications. The characteristic function entirely encapsulates the properties of a given distribution, and is a powerful statistical tool. As there is no closed-form currently known, numerical

approximation methods are necessary in the explicit evaluation of the characteristic function. In chapter 6 we offer novel methods for efficient evaluation, which in contrast to previously proposed methods are not subject to certain parameter ranges to guarantee robustness. The proposed methods are based on series expansions and quadrature methods and perform favourably to a number of previously introduced approximations.

Rayleigh distributions are one of the most utilised distributions in the analysis and modelling of wireless communication systems including multiple channels. However, closed-forms of the multivariate Rayleigh distribution are not yet known, necessitating efficient numerical approximations. While various approximation methods have been proposed, all of these approaches are subject to dimensional restrictions or limited to covariance matrices of specific structure. In chapter 7 we introduce a generalisation of a previous approach that releases dimensional constraints, and in chapter 8 we further develop this approach to allow covariance matrices of arbitrary make-up, therefore providing a truly general multivariate formulation.

For the numerical computation of the multivariate Rayleigh distribution necessary for the validation of our proposed approximations, a multivariate empirical approach was necessary to derive a density from the sampled random values. To this end, we have written a software package that offers a number of standard, and newly implemented approaches, that have not been available in previous packages. The methods contained in this package and comparisons to other algorithms are contained in chapter 8.

Chapter 10 closes this thesis with a brief summary of the results we have retrieved and outline further research topics we intend to tackle, for which the basis has been established in this thesis.

1.3 Publications

The chapters of this thesis have been either submitted to refereed journals or have already been accepted for publication as noted by the list below:

- **Chapter 2** has been published as 'New composite distributions for modeling industrial income and wealth per employee' in Physica A: Statistical Mechanics and its Applications (Vol. 492, p. 1901-1908; 2018)
- **Chapter 3** has been published as 'Word frequencies: A comparison of Pareto type distributions' in Physics Letters A (Vol 382, Issue 9, p. 621-632; 2018)

- **Chapter 4** is under review as 'Compdist: Multisection Composite Distributions' in the R Journal.
- **Chapter 5** has been published as 'General moments of roundoff error' in Communications in Statistics - Simulation and Computation (2019).
- **Chapter 6** has been published as 'Approximation methods for Lognormal Characteristic Functions' in the Journal of Statistical Computation and Simulation (Vol. 88, Issue 18, p. 3650-3663;- 2018).
- **Chapter 7** has been published as 'A Series Representation for Multidimensional Rayleigh Distributions' in International Journal of Communications Systems (31:e3510; 2018).
- **Chapter 8** has been accepted for publication as 'Series approximations for Rayleigh distributions of arbitrary dimensions and covariance matrices' in Signal Processing (2019).
- **Chapter 9** has been published as 'MEPDF: Multivariate empirical density functions' in Communications in Statistics - Simulation and Computation (2019).

In addition to the chapters of the thesis we have worked on several projects not included in this thesis, due to thematic differences and limitations in scope. In collaboration with Nicolas Tessore and Sarah Bridle of the astronomical group of the University of Manchester's physics department we have developed statistical estimators to the spatial properties of galaxies.

Based on an original physical problem Tessore and Bridle had managed to reduce the problem of computing the galactic ellipticity (also commonly known as 'flattening') to a purely statistical task, based on the observed measurements of high-powered terrestrial telescopes. Due to physical effects such as the gravitational influence of high mass stellar bodies or weak lensing on the light travelling vast distances across space, random noise is introduced into the measurements recorded by the telescopes. This necessitates accurate estimators that make use of all the information that is available. Finding such estimators proved to be a difficult task, as only a single, or in rare cases few measurements are available, rendering standard asymptotic estimation approaches inadequate.

The project resulted in the publication:

New Estimators for Galactic Ellipticity
M. Wiegand, S. Nadarajah
Monthly Notices of the Royal Astronomical Society
Vol, 484, Issue 3, p. 3984-4007; 2019

Further collaborations included a one-year project with the Zurich University of Applied Sciences (ZHAW) in Winterthur, Switzerland, lasting from mid-2017 to mid-2018. The goal of the project was to develop numerical methods to approximate the price of options on stocks, with a natural focus on American and exotic style options for which no closed pricing functions based on factors such as interest rates, volatility and maturity dates, exists.

Monte Carlo sampling methods can be utilised to determine close approximations to a price for a given set of variable values but has to be repeated for each new set of values. The numerical expensiveness of this approach makes a multivariate interpolation function of these fixed value prices desirable to minimise the numerical effort otherwise needed to compute arbitrary values.

The final results have been compiled under the title 'A comparative Study for Nested Monte Carlo Regression Techniques for Option Pricing'. The project was partially funded by industrial partners in combination with the Swiss national science foundation (SNF) and has not been cleared for general publication due to the private investment in the project. A copy of the results can be provided to the examiners upon request.

During the duration of the project I have also been co-supervising the master's dissertation of Mr. Andreas Gabler of the ZHAW, on the topic of 'Pricing, Loss and Sensitivity Analysis of Barrier Options via Regression'. Due to the thematic overlap of Andreas Gabler's research and my own work for the department, a close collaboration on the development and implementation of regression methods took place. The results of this project can be found on the open-access platform Social Science Research Network (SSRN).

Another project rooted in distribution theory has been carried out with other researchers of the School of Mathematics at the University of Manchester. With Filippo Pagani several multivariate density extensions of the common Rosenbrock function have been developed. These probability functions have various properties that make them ideal to test the robustness and performance of new and existing MCMC sampling algorithms. The results of this project have been submitted to and are under consideration as

An n-dimensional Rosenbrock Distribution for MCMC Testing
F. Pagani, M. Wiegand, S. Nadarajah
The American Statistician.

The most recent collaborative project began in early 2019 with the medical faculty

of the University of Manchester. The research group of the cough clinic under the leadership of Jaclyn Smith specialises in respiratory diseases which cause patients to suffer from frequent coughing fits.

The current tasks are the development of statistical models that encapsulate the distribution of interarrival times between coughs in a patient. This is done with special attention paid to the structure of the data, which includes both the interarrival times within a heavy bout of coughs in a patient, as well as the interarrival times between these bouts ('rest' periods). We have found very potent candidates based on the research included in this thesis to more accurately describe the behaviour of the observed coughs statistically.

Next steps include the development of a statistical diagnostic tool, which is able to categorise patients to disease groups, based solely on the observed cough data. We have seen good results so far employing non-parametric similarity measures and machine learning techniques, with results expected to be submitted for publication later this year.

1.4 Contributions

In this section I would like to detail the projects in which other people besides myself have contributed, and to which degree they were involved. I will break down the various collaborators and their individual input to the chapters.

- **Chapter 2** *New Composite Distribution for Modeling Industrial Income and Wealth per Employee*

  All computations and writing have been carried out by myself. The project was supervised and the final submission edited by Saralees Nadarajah.

- **Chapter 3** *Word Frequencies: A Comparison of Pareto Type Distributions*

  All computations and writing have been carried out by myself. The project was supervised and the final submission edited by Saralees Nadarajah.
  Yuancheng Si of the Distribution theory group has aided in the collection of the data that was analysed.

- **Chapter 4** *CompDist: Multisection Composite Distributions*

The software package has been written entirely by myself, as well as the accompanying documentation in the chapter and paper. The project was supervised and the final submission edited by Saralees Nadarajah.

- **Chapter 5** *General Moments of Roundoff Erorr*

  All computations and writing have been carried out by myself. The project was supervised and the final submission edited by Saralees Nadarajah.

- **Chapter 6** *Approximation Methods for Lognormal Characteristic Functions*

  All computations and writing have been carried out by myself. The project was supervised and the final submission edited by Saralees Nadarajah.

- **Chapter 7** *A Series Representation for Multidimensional Rayleigh Distributions*

  All computations and writing have been carried out by myself. The project was supervised and the final submission edited by Saralees Nadarajah.

- **Chapter 8** *Series Approximations for Rayleigh Distributions of Arbitrary Dimensions and Covariance Matrices*

  All computations and writing have been carried out by myself. The project was supervised and the final submission edited by Saralees Nadarajah.

- **Chapter 9** *MEPDF: Multivariate Empirical Density Functions*

  The software package has been written entirely by myself, as well as the accompanying documentation in the chapter and paper. The project was supervised and the final submission edited by Saralees Nadarajah.

As several minor errors and typing mistakes have slipped the editors and journal referees, as well as us during revision processes, some alterations were made to the original paper texts. All alterations have been marked in footnotes with [A] for alterations, [C] for comments and [O] for omission of content, with the specific changes from the published version listed and explained.

1.5 Conference Talks

During the duration of my PhD program, I was able to represent the School of Mathematics at national and international conferences on statistical advances. On these

occasions I have been able to present the results of my work to the scientific community, as well as learn about developments in statistical modelling in engineering and other application areas.

- **2017 - The second uncertainty quantification and management study group with industry**
  University of Liverpool, UK.

- **2017 - COST 2 Conference: Artificial Intelligence in Industry and Finance**
  ZHAW Winterthur, Switzerland.

  Organisation and attendance as part of ZHAW staff. Hosted by the group of finance and risk modeling at the ZHAW in Winterthur.

- **2018 - PCM 125: International Conference in Probability and Statistics**
  "Composite distributions to model income and wealth per capita"
  Hosted by the Indian Statistical Institute at the University of Kolkata, India.

- **2018 - International Multiconference of Engineers and Computer Scientists**
  "A new series representation of multivariate Rayleigh distributions"
  Hosted by the IAENG in Hong Kong.

  Talk presented on the topic of newly discovered Rayleigh distribution approximations with relaxed covariance matrix and dimensional limitations. The presentation has been awarded the 'Best student paper award' by the IAENG. Funding has been obtained through the Data Science Institute's travel award.

- **2018 - MRSC Mathematics Research Students' Conference**
  "Word frequencies: A comparison of Pareto type distributions"
  Hosted by the postgraduate student representatives at the University of Manchester.

  The presentation was recognised with the IBM best talk award by a jury of academics and industrial partners.

- **2019 (planned) - XV Latin American Congress of Probability and Mathematical Statistics**
  "New Estimators for galactic Ellipticity"
  Hosted by the Sociedad Latino Americana de Probabilidad y Estadística

Matemática (SLAPEM) in Merida-Yucatán, México.

The participation in the conference has been funded by the IMS through the Hannan travel award.

1.5.1 Other Activities

I am honoured to have been part of the 'Educate Africa' initiative, brought into being by Saralees Nadarajah, which focuses on making higher education more accessible to students of any African nation. Through online courses, students are provided with multiple lectures every week, where staff shortages would otherwise prevent courses to take place or severely limit academic opportunities. The project pursues a sustainable approach to realise the full potential of budding researchers in Africa, that aim to improve the lives of their fellow people by tackling issues in health, finance or engineering through statistical methods.

The project has been recognised with several awards, such as the Better World Showcase award for 'Outstanding contribution to equality, diversity and inclusion', the People's vote and a commendation by the Making a Difference awards.

The initiative is ongoing with the intention to expand in both size and scope, with more volunteers offering more courses, and remote supervision of Master's student. Additionally we are engaged in the investigation of publication bias against and perception of African researchers on a global level.

1.5.2 Common Notation

In Table 1.1 we introduce common notation and abbreviations we have used in all chapters, and briefly explain the differences between the notations. Definitions can be found in the respective chapters, and can differ based on the context.

	Abbr.	Description
Probability density function	PDF	Standard abbreviation, common in statistics.
Cumulative distribution function	CDF	
Penalised Error Measures		
Akaike-Information-Criterion	AIC	
Modified Akaike-Information-Criterion	AICc	Model is fitted by maximum likelihood method, except for Chapter **??** where a modification is introduced
Bayesian-Information-Criterion	BIC	
Consistent Akaike-Information-Criterion	CAIC	
Hannan-Quinn-Criterion	HQC	
Other Error Measures		
Average Absolute Error	AAE	
Average Squared Error	ASE	For goodness of fit assessments involving data, the evaluation points are chosen where data is available. For approximation methods in the second part of the thesis, equidistant evaluation points are chosen if not stated otherwise. The AGE uses midpoints of the evaluation grid, and is explained explicitly.
Maximum Absolute Error	MAE	
Kolmogorov-Smirnov Statistic	KS	
Anderson-Darling Statistic	AD	
Average Relative Absolute Error	ARAE	
Aggregated Absolute Error	AGAE	
Aggregated Squared Error	AGSE	
Average Grid Error	AGE	

Table 1.1: Common abbreviations used in the following chapters.

Chapter 2

New Composite Distributions for Modeling Industrial Income and Wealth per Employee

Chapter Abstract

Forbes Magazine offers an annual list of the 2000 largest publicly traded companies, shedding light on four different measurements: Sales, profits, market value and assets held. Soriano-Hernández et al. [Soriano-Hernández et al. (2016)] modeled these wealth metrics using composite distributions made up of two parts. In this chapter, we introduce different composite distributions to more accurately describe the spread of these wealth metrics.

2.1 Introduction

In this chapter, we investigate the Global 2000 data set, compiled by Forbes Magazine. It features the 2000 largest companies, with their most important financial indicators, namely annual profits, sales, market value and assets along with employee count and a ranking system based on a combined value.

An analysis was published by Soriano-Hernández et al. [Soriano-Hernández et al. (2016)], where two part models were introduced along with a number of different distribution combinations to predict the percentage of companies below a certain wealth threshold. The tail distribution remained a Pareto distribution of type 1 for all estimations, combined with either a log-normal, gamma or exponential distribution modelling the body [1] part of the sample. These distributions were chosen on the grounds of previous successful modelling in finance [Soriano-Hernández et al. (2017)], [Gibrat (1931)], [Gini (1921)] or modelling of gas propagation in physics [Chakrabarti et al. (2013)].

The basic principle was to divide the data into two parts, and introduce a partial distribution for each part. Both distributions would then be connected by a hard cut-off, leading to a non-continuous, abrupt transition. Formally, the probability density function (PDF) and the cumulative distribution function (CDF) of the composite model can be specified by

$$f(x) = \begin{cases} f_1(x), & \text{if } x \leq \theta, \\ [1 - F_1(\theta)]\, f_2(x), & \text{if } \theta < x, \end{cases}$$

and

$$F(x) = \begin{cases} F_1(x), & \text{if } x \leq \theta, \\ F_2(x) + F_1(\theta) - F_1(\theta)F_2(x), & \text{if } \theta < x, \end{cases}$$

respectively. Here f_1, f_2 denote two probability densities and F_1, F_2 their respective CDFs. Here $F_{2,X}, f_{2,X}$ denote the CDF and density of the Pareto distribution, specifically $X \sim Pareto(\theta)$.

[1] [C] We refer to the different section of the data in the order of the respective company's rank from lowest to highest as follows: Body (ranks $\sim 2000 - 1500$), mid (ranks $\sim 1500 - 500$) and tail section (ranks $\sim 1 - 500$). The rank ranges are not fixed and are as parameters subject to the optimisation process.

Rather than modeling the values provided directly, it was decided that a quotient of a metric and the employee count would be more expressive, giving insight into how much returns the employees generate. This led to the assumption that all observed businesses can be divided into two categories, depending on the number workforce employed. The first category would be companies in retail like Wal-Mart, which have to rely on large numbers of employees for their services. On the other hand we have companies like Apple, which due to the nature of their products and production processes can perform with comparatively low employee number in relation to their revenue.

The focus on the application of this model is less in fitting a PDF or CDF to the data, but in a decreasing percentage function $100\left[1-\widehat{F}(x)\right]$ to describe the amount of companies below a certain wealth metric. In this chapter, we introduce a new approach which suggests a third subpopulation in Section 2.2. We then show that it provides a considerable improvement in fits over the old approach. We verify our results with a number of error measures reflecting the goodness of fit and comparative plots in Section 2.3, visualising the areas of greatest improvement. In Section 2.4 we summarise our findings and discuss further work. [2]

2.2 Composite Models

Contrary to the previously proposed model, we argue that a distinct third population group is present, leading us to the construction of a three part composite model. Upon visual inspection, the best fitting two part models in Figure 2.1, while offering a reasonable model for sections of the data, do not appear to entirely capture the proper curvature of the percentage function. The chosen distributions are the lognormal and gamma distribution for the body section, and Pareto type I distribution for the tail section. [3].

Due to the log-log scale, the inaccuracies for the Sales and Profits metrics seem more evident in the range of top $15-20\%$ of companies. While the parts closer to the catenation point still appear to be captured adequately by the previous model, higher ranked company values stray further away. In the tail section we hypothesise that a third section at around 0.1 billion and 0.05 billion, respectively, could diminish this

[2] [A] Section labels added.

[3] [A] The published paper does not explicitly state that this comment refers to visual inspection only, and that the deviations we improve upon also stem from the mid and body sections (ranks ca. 2000-500), which can be seen much more clearly in Figure 2.3

deviation. [4]

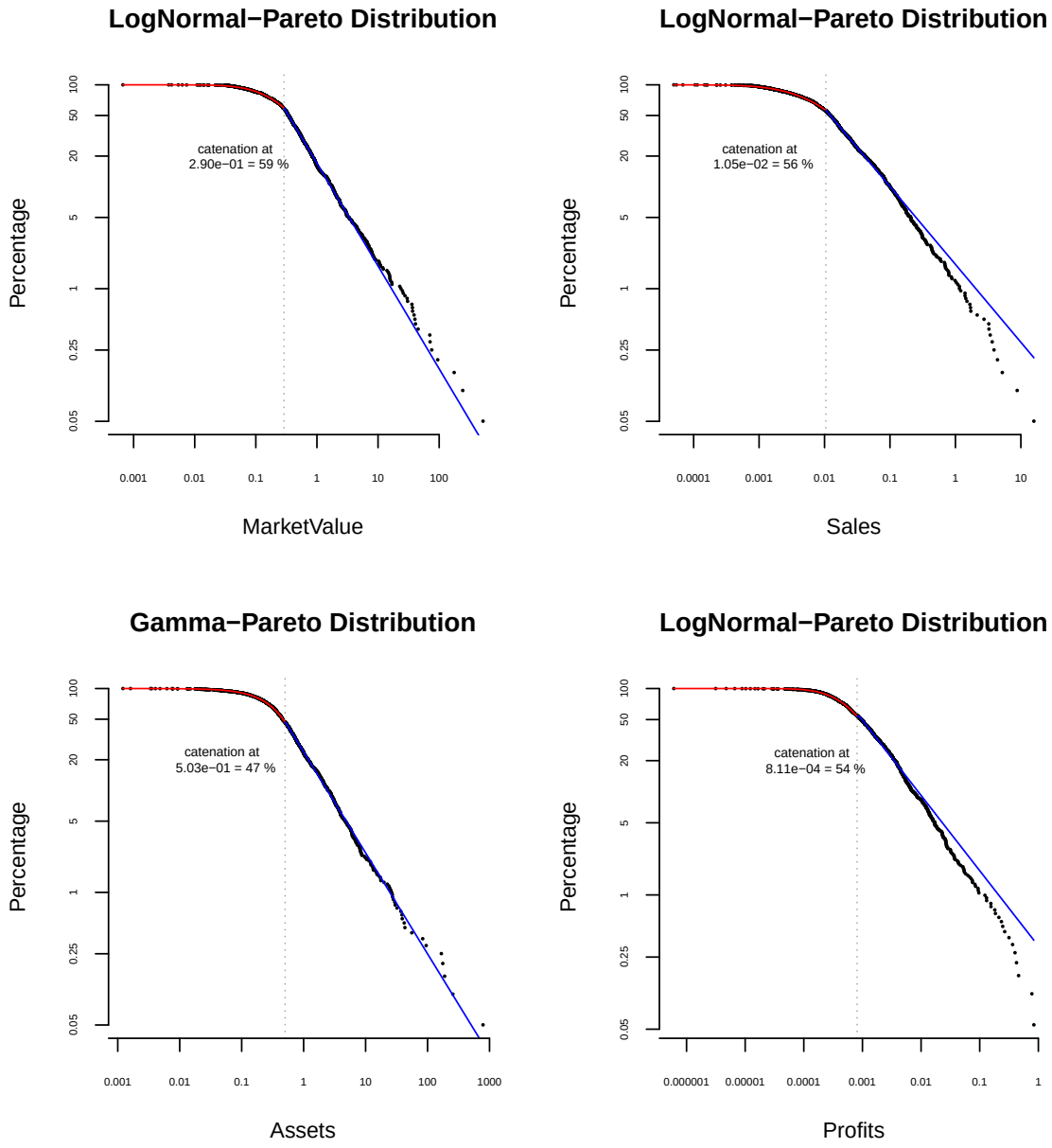


Figure 2.1: The four metrics vs. the rank percentage $100(1-\hat{F}(x))$ with the respective best fit two part models.

Furthermore, we like to introduce a smoother variant of the composite model,

[4][A] It is necessary to stress that the plots of Figure 2.1 put emphasis on the highest ranks, and the fit in the body to mid-section is somewhat obscured. Following Tables and Figures better illustrate this.

which in its two part form has been introduced by Bakar and Nadarajah [Bakar, Nadarajah (2013)]. We can choose the parameter value of Φ to guarantee that $\lim_{x\uparrow\theta} f(x) = \lim_{x\downarrow\theta} f(x)$ holds for the composite density f, providing a continuous transition between sections. The PDF and CDF are provided below for arbitrary distributions with PDFs f_1, f_2 , where f_1 is the density of the first component ($x \leq \theta$) and f_2 a density for the second component ($\theta < x$) and CDFs F_1, F_2 merged at point $\theta \in \mathbb{R}$ with weight $\Phi \in \mathbb{R}^+$:

$$f(x) = \begin{cases} \frac{1}{1+\Phi}\frac{f_1(x)}{F_1(\theta)}, & \text{if } x \leq \theta, \\ \frac{\Phi}{1+\Phi}\frac{f_2(x)}{1-F_2(\theta)}, & \text{if } \theta < x, \end{cases}$$

and

$$F(x) = \begin{cases} \frac{1}{1+\Phi}\frac{F_1(x)}{F_1(\theta)}, & \text{if } x \leq \theta, \\ \frac{1}{1+\Phi}\left[1 + \Phi\frac{F_2(x)-F_2(\theta)}{1-F_2(\theta)}\right], & \text{if } \theta < x. \end{cases}$$

We now expand this model by a third partial distribution with PDF f_3 and CDF F_3. Additionally we now mark $\theta_1 < \theta_2$ as the catenation points between the composites and $\Phi, \Psi \in \mathbb{R}^+$ as weights. This yields

$$f(x) = \zeta \begin{cases} \frac{f_1(x)}{F_2(\theta_1)}, & \text{if } x \leq \theta_1, \\ \Phi\frac{f_2(x)}{F_2(\theta_2)-F_2(\theta_1)}, & \text{if } \theta_1 < x \leq \theta_2, \\ \Psi\frac{f_3(x)}{1-F_3(\theta_2)}, & \text{if } \theta_2 < x, \end{cases} \tag{2.1}$$

and

$$F(x) = \zeta \begin{cases} \frac{F_1(x)}{F_1(\theta_1)}, & \text{if } x \leq \theta_1, \\ 1 + \Phi\frac{F_2(x)-F_2(\theta_1)}{F_2(\theta_2)-F_2(\theta_1)}, & \text{if } \theta_1 < x \leq \theta_2, \\ 1 + \Phi + \Psi\frac{F_3(x)-F_3(\theta_2)}{1-F_3(\theta_2)}, & \text{if } \theta_2 < x. \end{cases}$$

For convenience, we introduce $\zeta = \frac{1}{1+\Phi+\Psi} \in \mathbb{R}^+$ as scaling parameter. Soriano-Hernández et al. [Soriano-Hernández et al. (2016)] have proposed most prominently the gamma and log-normal distributions for the body (f_1, F_1) and a Pareto type 1 distribution for the tail (f_2, F_2). We mostly agree with the choice of distributions being used for the body component, but we propose the beta Weibull (beta Wb) distribution, which we will employ for the body as well as the mid proportion of the data. We have tried several heavy tailed distributions due to the heavily slanted nature of the relation between rank and metrics. [5] The beta Wb distribution due to Lee et al.

[5][A] Some of the worse performing distribution were: Mills distribution, Pareto type I-IV, Beta

[Lee et al. (2007)] has the PDF and CDF specified by:

$$f_{\lambda,k,\alpha,\beta}(x) = k\lambda^k x^{k-1} \exp\left[-\beta(\lambda x)^k\right] \left\{1 - \exp\left[-\beta(\lambda x)^k\right]\right\}^{\alpha-1}$$

and

$$F_{\lambda,k,\alpha,\beta}(x) = I_{1-\exp\left[-\beta(\lambda x)^k\right]}(\alpha, \beta),$$

respectively, for $x > 0$, $\lambda > 0$, $k > 0$, $\alpha > 0$ and $\beta > 0$, where $B(a, b)$ and $I_x(a, b)$ are the beta function and incomplete beta function ratio defined by

$$B(a, b) = \int_0^1 t^{a-1}(1-t)^{b-1}dt$$

and

$$I_x(a, b) = \frac{1}{B(a, b)} \int_0^x t^{a-1}(1-t)^{b-1}dt,$$

respectively, for $0 \leq x < 1$ and $\min\{\Re(a), \Re(b)\} > 0$, also respectively.

While the Pareto Type-I distribution has been commonly used to describe the distribution of wealth among the inhabitants of a country or larger region, especially the more prosperous ones, we wish to transfer this approach to the wealth spread among international companies. However, we believe that the general shape of the type I distribution cannot accurately capture the most extreme points of the tail. Asymptotic arguments such as the Pickands-Balkema-de Haan theorem mandate the shift to the generalised Pareto distribution of higher order, offering a better fit through its more flexible shape. This argument has already found its way into application (see Degen and Embrechts [Degen, Embrechts (2008)] or Kleiber and Kotz [Kleiber, Kotz (2003)]) and it seems sensible to us to update the preexisting approach.

2.3 Model Comparison

We are now left with three body, two midsection and one tail distribution of interest to us. The next step is to fit all six different section combinations onto the four wealth

prime, Beta Pareto, Beta Lognormal, Beta Weibull, Gamma Pareto, Weibull Pareto, Weibull pareto and the modfied Champernowne distribution. They have been tested for their fit in the respective sections and as part of a composite function, but performed worse or not well enough to justify the parameter count.

metric data sets and compare the resulting error measures for previous and current approaches. In addition to the squared and absolute aggregate errors, we investigate a number of variations on the Akaike information criterion (AIC) due to Akaike [Apostol (1974)], the Kolmogorov-Smirnov (KS) test statistic (here the maximum deviation of the percentage function) and the Anderson-Darling (AD) statistic. Specifically, we use the Bayesian information criterion (BIC) due to Schwarz [Schwarz (1978)], consistent Akaike information criterion (CAIC) due to Bozdogan [Bozdogan (1987)], corrected Akaike information criterion (AICc) due to Hurvich and Tsai [**?**] and Hannan-Quinn criterion (HQC) due to Hannan and Quinn [Hanna, Quinn (1979)]. The results can be found in Table 2.1. [6]

[6][A] The table has been altered, and now only includes the AIC, as the modifications showed almost identical results. The full table can be found in the Appendices A.1.

Model	Metric	Partial Distribution			Optimized via Percentage Function							
		Body	Mid	Tail	Abs.	Sq.	Rel. Abs.	Rel. Sq.	KS	AD	AIC	Function
Two part hard cut-off	Market Value	LogNorm	-	Pareto1	673.20	347.18	0.338	0.174	1.27	1999	7416	optim
		Gamma	-	Pareto1	825.37	690.55	0.414	0.346	2.43	1943	7853	optim
		Expon	-	Pareto1	1593.89	2825.0	0.800	1.417	4.16	1861	10287	optim
	Sales	LogNorm	-	Pareto1	861.61	530.10	0.276	0.266	1.37	1934	4062	optim
		Gamma	-	Pareto1	966.82	1015.43	0.485	0.509	2.86	1875	-2067	optim
		Expon		Pareto1	1148.76	1058.17	0.576	0.531	2.46	1891	7524	optim
	Assets	LogNorm	-	Pareto1	1189.89	1095.36	0.597	0.550	1.86	2089	9909	optim
		Gamma	-	Pareto1	630.88	309.96	0.317	0.156	0.99	2015	6682	optim
		Expon	-	Pareto1	1889.23	2436.54	0.948	1.223	2.92	1871	11410	optim
	Profits	LogNorm	-	Pareto1	1041.46	801.72	0.574	0.442	1.43	1834	-4150	optim
		Gamma	-	Pareto1	1345.97	1396.64	0.742	0.769	2.29	1741	501	optim
		Expon	-	Pareto1	2375.88	5842.95	1.309	3.219	4.69	1637	2953	optim
Three part composite model	Market Value	LogNorm	Pareto4	Pareto4	595.08	259.95	0.299	0.130	1.05	2006	2456	optim
		Gamma	Pareto4	Pareto4	550.04	252.90	0.276	0.127	1.33	1983	2455	optim
		BetaWb	Pareto4	Pareto4	459.11	173.07	0.230	0.087	0.95	2162	2446	optim
		LogNorm	BetaWb	Pareto4	517.76	224.03	0.260	0.112	1.17	1998	2461	optim
		Gamma	BetaWb	Pareto4	497.78	202.54	0.250	0.102	1.01	1999	2467	optim
		BetaWb	BetaWb	Pareto4	593.58	231.47	0.298	0.116	0.92	2011	2505	optim
	Sales	LogNorm	Pareto4	Pareto4	493.65	190.27	0.248	0.095	0.84	2019	-9878	optim
		Gamma	Pareto4	Pareto4	514.17	195.64	0.258	0.098	0.96	2008	-10232	optim
		BetaWb	Pareto4	Pareto4	482.84	186.34	0.242	0.093	0.90	2101	-9740	optim
		LogNorm	BetaWb	Pareto4	515.18	223.08	0.258	0.112	1.17	1986	-9891	optim+nlm
		Gamma	BetaWb	Pareto4	426.80	134.63	0.214	0.068	0.72	2022	-9914	optim+nlm
		BetaWb	BetaWb	Pareto4	502.86	198.86	0.252	0.100	1.03	1982	-9922	optim+nlm
	Assets	LogNorm	Pareto4	Pareto4	340.82	93.27	0.171	0.047	0.65	2016	3813	optim
		Gamma	Pareto4	Pareto4	349.24	93.86	0.175	0.047	0.58	2017	3784	optim
		BetaWb	Pareto4	Pareto4	353.57	98.92	0.177	0.050	0.63	2002	3824	optim
		LogNorm	BetaWb	Pareto4	348.30	90.91	0.175	0.046	0.57	2013	3811	optim
		Gamma	BetaWb	Pareto4	337.12	88.60	0.169	0.044	0.59	2008	3796	optim
		BetaWb	BetaWb	Pareto4	337.00	89.79	0.169	0.045	0.67	2006	3802	optim
	Profits	LogNorm	Pareto4	Pareto4	547.78	248.90	0.302	0.137	1.03	1926	-18081	optim
		Gamma	Pareto4	Pareto4	578.11	326.53	0.319	0.180	1.22	1802	-18107	optim
		BetaWb	Pareto4	Pareto4	602.18	290.41	0.332	0.160	1.07	1857	-18032	optim
		LogNorm	BetaWb	Pareto4	474.65	193.04	0.262	0.106	0.85	1910	-17998	optim
		Gamma	BetaWb	Pareto4	432.23	161.55	0.238	0.089	0.85	1903	-17881	optim
		BetaWb	BetaWb	Pareto4	596.69	286.05	0.329	0.158	1.07	1862	-17831	optim

Table 2.1: Error measure compilation for composite models, loop tolerance=10^{-8} for percentage function fitting.

To ensure the optimal fit for the model and to cut down on computation time, we first computed maximum likelihood estimates (MLEs) for each of the partial distributions and used the resulting parameter estimates as initial values for the MLEs of the composite model. The MLEs obtained in this way were used to compute on the one hand the AIC, BIC, etc measures, and secondly serve as initial values for the minimization of the deviation between the modeled percentage function and empirical percentages (for example, proportion of the data sample below a given value). As the main attention analogously to the original work in [Soriano-Hernández et al. (2016)] lies on the fit of the distribution model to the inverse CDF (scaled up by 100 to resemble percentages), we chose to optimise parameters accordingly through minimisation of the deviation from the inverse CDF, rather than MLE for the parameters. All computations were performed using the R software [R Development Team (2017)]. The optimization function in the R software used to find the MLEs is shown in the last column of Table 2.1.

If we look into the percentage based measures (for example, squared and absolute aggregate errors), it becomes evident that the tri-composite model provides a better fit, with up to almost 60 percent reduction in squared error and up to 50 percent reduction in absolute error. For most instances the choice of body distribution only plays a minor part, as long as it is sufficiently heavy-tailed (gamma, log-normal, beta Wb distributions tend to perform mostly similarly). We omitted the exponentially distributed body-variant, since it consistently provided the highest error margins in our fitting process. With the introduction of a mid-section and the inherent improved flexibility of the composite distribution, we anticipated lower error measures. Nonetheless by the amount the measures have changed, we see our hypothesis of a third distinct subgroup within the company samples verified.

The density based measures draw a similar picture. What strikes us, is that not only do the composite models possess a lower AIC, BIC etc. value than the hard cut-off model, but the values are within a very tight grouping. We attribute this to the fact that $f(x)$ as described in (2.1) consists of disjunct partitions, which do not need to be necessarily continuous at the catenation points. Furthermore, the tail distribution remains identical and the mid distributions only differ slightly from one another. Since we developed the model in order to more accurately determine the percentage function as characterized in [Soriano-Hernández et al. (2016)], we give direct error measures of the percentage deviation more weight.

Next we turn our attention to the comparative plots in Figure 2.2:

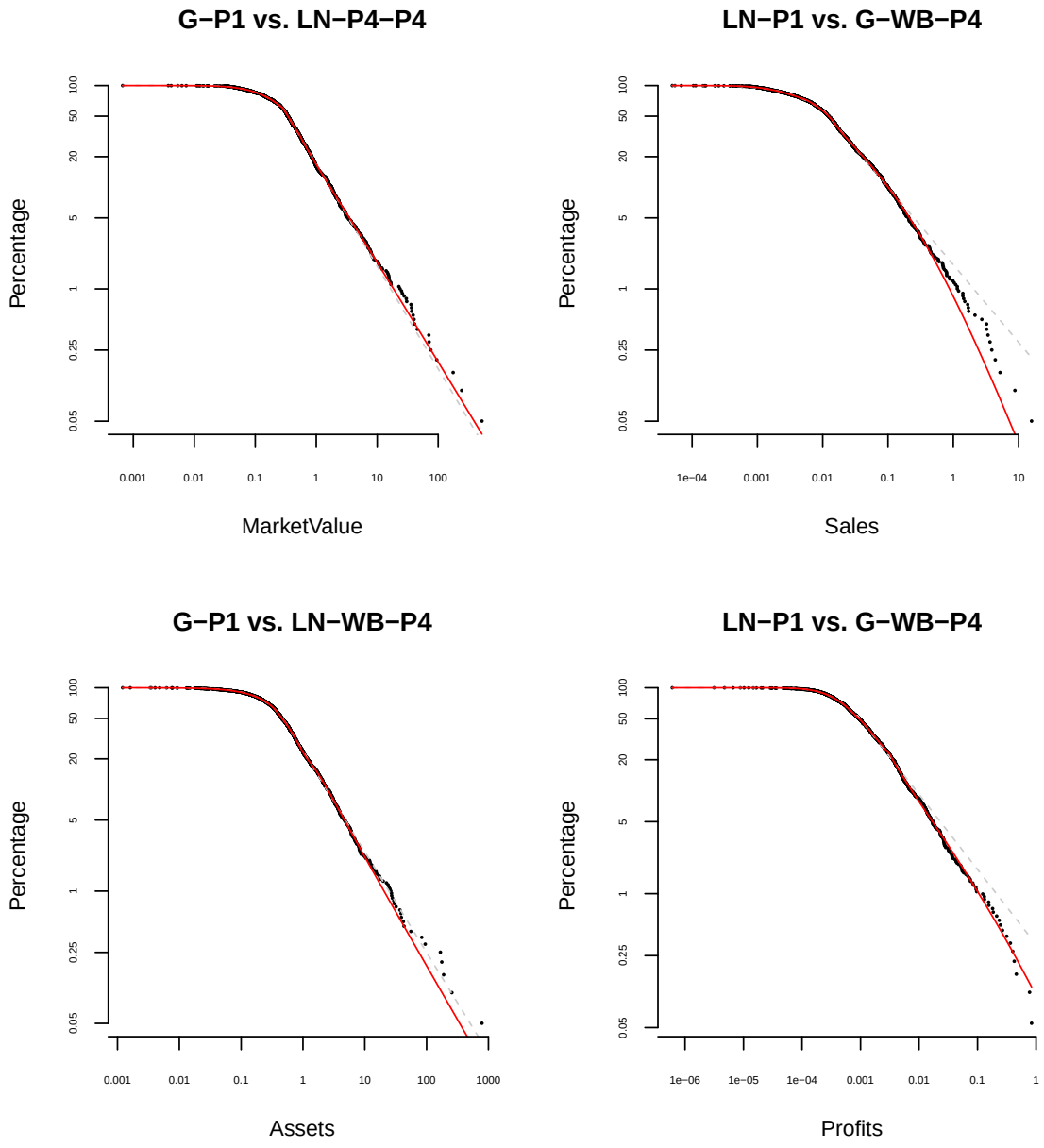


Figure 2.2: A comparative log-scale plot between the best fit variants of old (grey) and new (red) models for metric x vs rank percentage $100(1 - \hat{F}(x))$.

All variants of the newer model indeed capture the distribution of the wealth metrics more closely, with the exception of the Assets metric, where the Pareto type I distribution visually appears to be a more viable option. [7]

[7][A] This counterpoint has been glossed over in the published version. However, the plots serve only as a visual aid, and error measures featured in the tables provide a more reliable assessment criterion

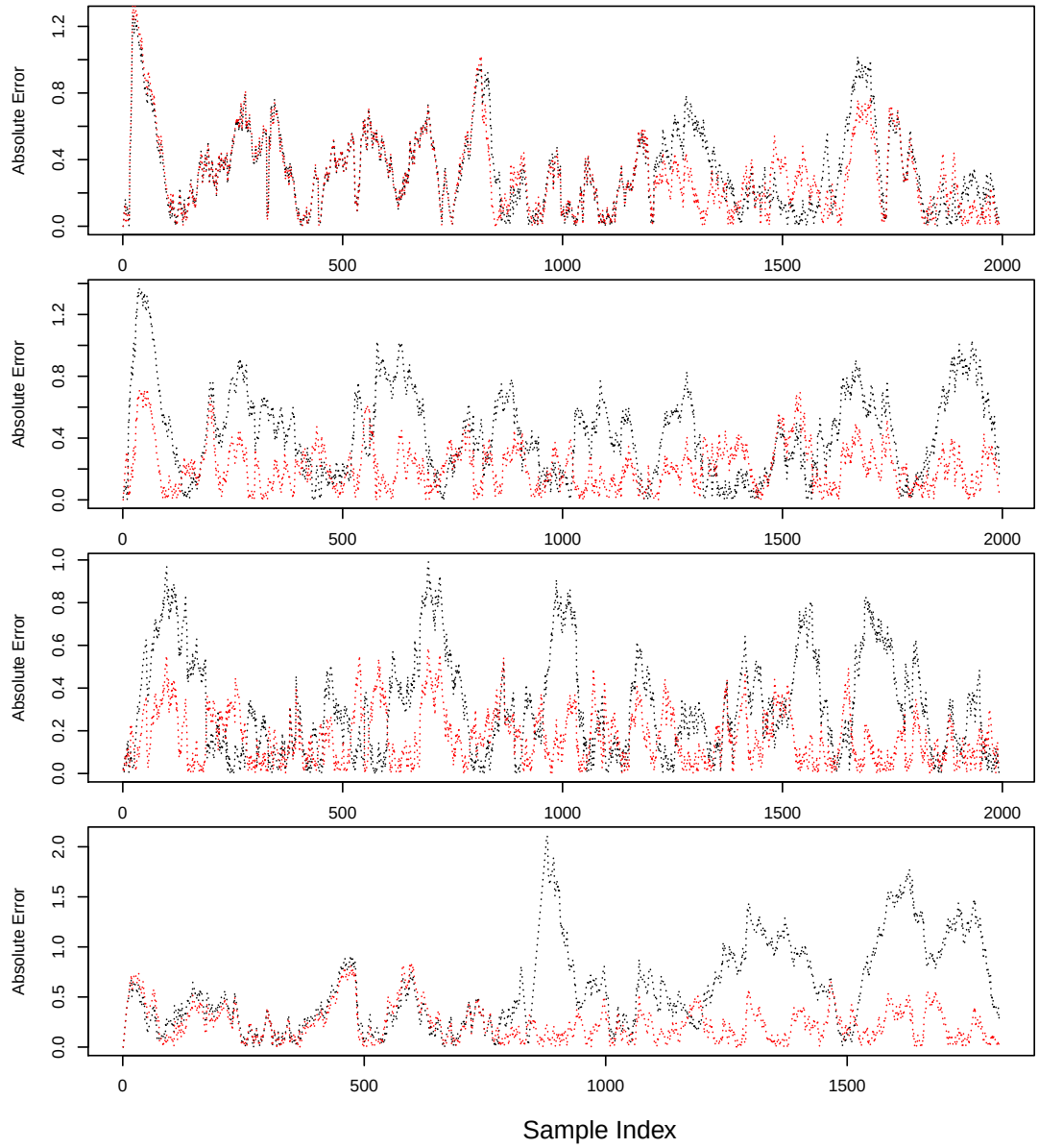


Figure 2.3: From top to bottom: market value, sales, assets and profits with the absolute error of the two part (black) and three part (red) models.

While the logarithmic nature of the plots overemphasizes the tail deviations, where especially sales and profits data can seen with increased accuracy, a large part of the decrease in deviation has been achieved in the body and mid-sections of the data (for companies ranked $\sim 2000 - 500$) [8]. We therefore have provided a plot of the absolute error propagation throughout the data sample, with the previous model in gray and our approach in red once more (see 2.3) [9].

[8] [A] Figure 2.4 fails to visualise this, Figure 2.3 more clearly shows deviations by company rank
[9] [A] False Figure label has been amended

As we can see for the sales and assets metric, the sharp error peaks through the entire sample have been trimmed considerably (Figure 2.3). It seems that the improvement of the 3-part model streches through all sections for those two particular attributes. The samples of the market values and especially profits draw a somewhat different picture. Most of the improvement of the new model over the previous two part approach is accumulated within the last section, or more accurately the third tercile.

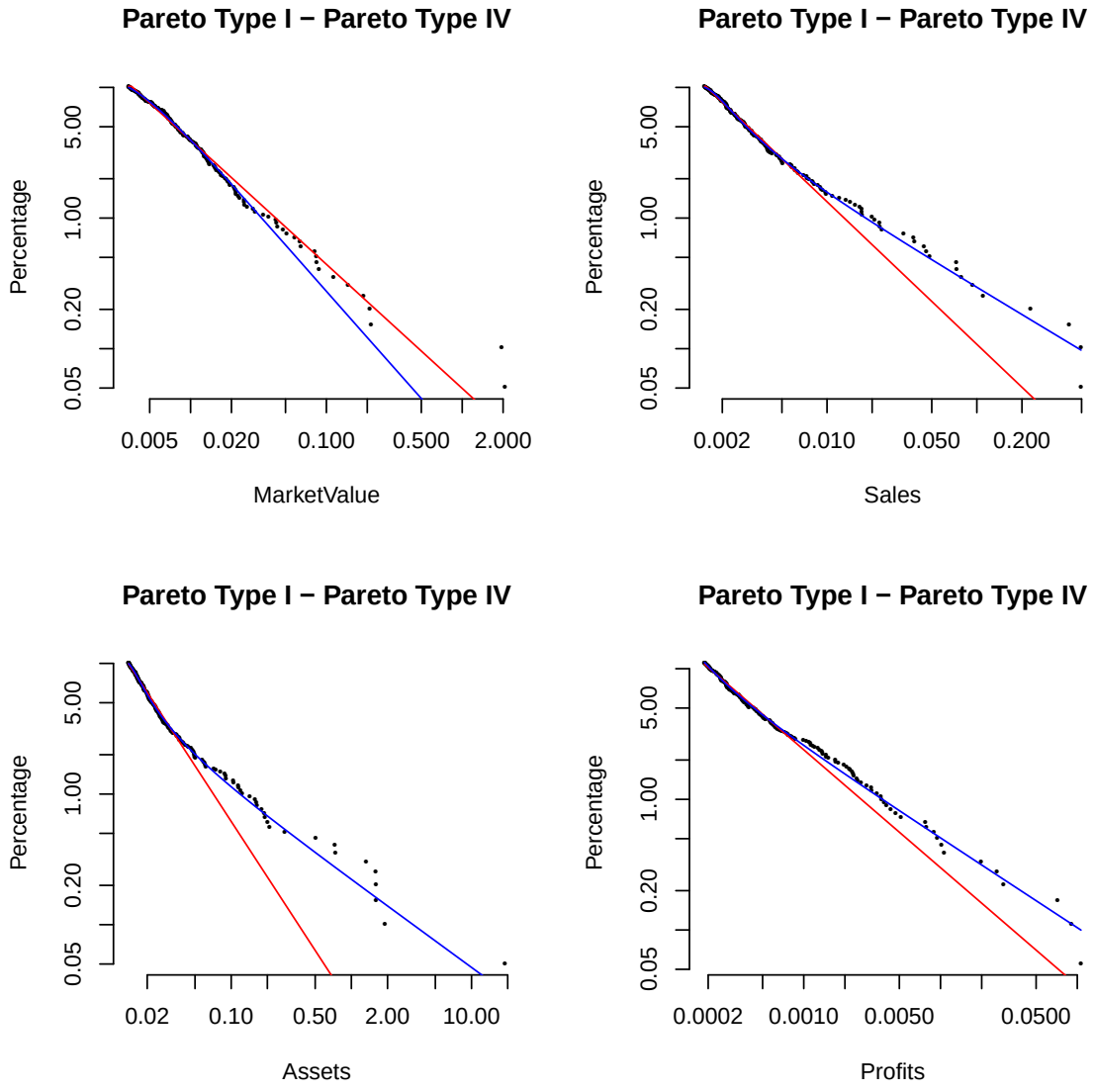


Figure 2.4: A log-scale plot of the top 10% of the respective samples, with only Pareto-I and Pareto-IV distributions fitted. Metric value x vs rank percentage value $100(1 - \hat{F}(x))$

The improvement in the lower two parts are due to higher flexibility of the composite distribution as a whole and a more accurate choice of partial distributions, based on empirical findings with a large number of parametric functions. Contrarily the reason the tail distribution achieves a superior fit lies within the expected asymptotic convergence of the population tail towards a generalized pareto distribution [Degen, Embrechts (2008)]. While the historically applied Pareto type-I distribution is an appropriate starting point, the shape remains too restrictive to provide an accurate fit.

It is expected as stated earlier that the sample tail converges against a Pareto distribution of high order, as the tail threshold is moved higher towards the last sample point. [10] The introduction of the additional section thus concentrates this partial distribution better on the respective data, e.g. a higher quantile of samples, to capture this effect.

In Figure 2.3 we have singled out this fact. We can clearly see how the standard Pareto distribution fails to accurately describe the behaviour of the very last data points. Especially in the latter three metrics this circumstance presents itself quite dramatically. Due to the double logarithmic scale which is used, the differences in the higher quantiles are somewhat exaggerated. [11]

	Absolute Error		Squared Error		Kolmogorov - Smirnov	
Pareto Dist.	Type I	Type IV	Type I	Type IV	Type I	Type IV
Assets	14.2098	1.650316	42.37595	15.14181	0.690889	0.241456
M. V.	9.353292	2.781869	34.75241	18.93321	0.530084	0.329805
Profits	15.17699	4.587656	47.47998	23.92047	0.542113	0.343262
Sales	6.682243	1.578278	30.39588	14.12695	0.450729	0.213245

Table 2.2: Deviations of the Pareto Type distributions

We therefore provide a table of the deviation measures to compare both tail modeling approaches numerically (see Table 2.2). We can see that the improvement of

[10] [C] This follows from the Pickands-Balkema-de Haan theorem, which states the shape of $G(y)$ defined as $F_u(y) = \mathbb{P}(X - u \leq y | X > u) \to G(y)$ as $u \to \infty$ to be best described by the generalised Pareto distribution. This implies increasingly good fit of Pareto-type distributions or power-laws higher for higher quantiles of data sets, and their frequent use for tail-data modelling.

[11] [C] For the market value metric in 2.4 the type I Pareto distribution appears to cover the last 20 values better than the type IV distribution. In a higher resolution it becomes evident that for virtually all values below this extreme tail the higher order is the superior model. Table 2.2 makes this clear.

the type IV distribution over the standard Pareto distribution is considerable. As anticipated we can see how the more general distribution is better suited to capture the extreme tails of all four metrics.

2.4 Conclusions

As our initial observations in the original model suggested, a third distinct subgroup in the sample seems likely, and its addition into the composite model greatly enhanced the distribution fit, more precisely the proposed percentage function $100\left[1-\widehat{F}(x)\right]$. Not only did the errors in the newly introduced section drop, but it also allowed a more precise fit for the body and tail sections, which now stretch over a significantly smaller sample.

The new proposed model requires up to 16 parameters, which leads to a significant increase in computation time, the smooth tri-composite distribution does give a more accurate impression of the wealth distribution between major international companies as all tested error measures verify. [12] In our opinion this handicap is negligible, since the data set is released annually, hence the actual fitting procedure only has to be done sporadically. Furthermore, the multi step optimization approach we used (parameter estimates of section distributions used as initial values) narrows down the search process significantly, such that a practical use of the composite model seems likely to us.

Since we noticed that the actual choice of the body distributions does in most cases not affect the accuracy of the model too greatly, one might go as far as suggesting that for most practical purposes a single model (for example, the gamma-beta Wb-Pareto type 4 model, which has consistently performed well) can be utilized to replace all incarnations of the previous two part model.

[12] [A] Phrasing altered.

Chapter 3

Word Frequencies: A Comparison of Pareto Type Distributions

Chapter Abstract

Mehri and Jamaati [Mehri, Jamaati (2017)] used Zipf's law to model word frequencies in Holy Bible translations for one hundred live languages. We compare the fit of Zipf's law to a number of Pareto type distributions. The latter distributions are shown to provide the best fit, as judged by a number of comparative plots and error measures. The fit of Zipf's law appears generally poor.

3.1 Introduction

The primary means of communication among humans relies on the use of language to express ideas and emotions to one another. Depending on the language spoken, there seems to be a seemingly limitless amount of words. Strikingly, certain words or word groups appear more often than others. This observation was first described by Zipf [Zipf (1949)], who popularised an explanation based on the assumption that humans would use the most efficient way to describe a given concept.

Thus one would use specific, concise phrasing rather than a long-winded explanation with the same amount of informational value conveyed. Similar explanations had been mentioned earlier by Auerbach [Auerbach (1913)] and Estoup as claimed by Manning and Schutze [Manning, Schutze (1999)].

To quantify this assumption Zipf provided a power law relationship between frequency and ranked usage. This relation can be applied to a number of naturally occurring sequence frequencies, such as medical or financial data [Csordas et al. (2003)], [Baayen (2002)]. Mehri and Jamaati [Mehri, Jamaati (2017)] applied Zipf's law to the word distribution of different languages based on one hundred translations of the Bible [University Edinburgh, 2017].

Zipf's Law and Pareto distributions are ubiquitous in language. They even exist when language is treated as networks: structural properties of weighted networks [Masucci, Rodgers (2009)]; modeling in random texts [Cancho, Elvevag (2010)]; structure-semantics interplay in complex networks [Amancio et al. (2012)]; statistical properties of unknown texts in the Voynich manuscript [Amancio et al. (2013)]; authorship recognition via fluctuation analysis of network topology [Amancio (2015)].

We believe that the established method of a power law does not provide an appropriate fit and that the related Pareto type distributions could offer superior alternatives. After introducing the original formulation of Zipf's power law and applying it to the bi-dimensional data, we do the same for the generalized Pareto, as well as Pareto types I-III distributions. We apply the Kolmogorov-Smirnov (KS) test statistic along with a R squared measure and a squared error. The R squared measure has several definitions which are not always equivalent, and as we are strictly speaking not performing regression, we define the version used in this chapter in Equation 3.1 to avoid confusion. For an arbitrary word $i = 1, \ldots, n$ let the respective frequency be f_i and the computed rank ρ_i, then follows the following error measure:

$$R^2 = 1 - \frac{\sum_{i=1}^{n} \left(\rho - \hat{\rho}(f_i)\right)^2}{\sum_{i=1}^{n} \left(\rho - \bar{\rho}\right)^2}. \tag{3.1}$$

Here $\hat{\rho}(f)$ denotes the approximation for the rank based on the frequency f and $\bar{\rho} = \frac{1}{N}\sum_{i=1}^{N} \rho_i$ the average rank, with N being the number of distinct words observed. [1]

Our approach is essentially considering the (relative) frequency f_i of a word i as a random variable with a cumulative distribution function $F(f)$. We are therefore interested in the optimal fit of a parametric CDF to the empirical CDF $\hat{F}$ given through the recorded frequencies. For a proposed CDF $\tilde{F}(f, \theta)$ we are interested in the optimal parameter and the respective error as follows:

$$\hat{\theta} = \arg\min_{\theta} \sum_{i=1}^{n} \left(\tilde{F}(f_{i,r}, \theta) - \hat{F}(f_{i,r})\right)^2. \tag{3.2}$$

The different fits will be visualised by a number of comparative Log-Log plots for selected languages. Additionally we ran the fitting process on all languages stated in [Mehri, Jamaati (2017)], and have plotted the error measures accordingly, to visualise the effectiveness of both approaches. To verify the outcome for single author literary works we have added results on a number of different texts, as well as for randomly generated texts of different lengths (see [Cancho, Elvevag (2010)]) in Section 3.2. We will conclude this chapter with a summary on different models and their suitability for further applications in Section 3.3. [2]

We would like to mention that there is a large body of work committed to understanding Zipf's law, more appropriate representations for rank frequency distributions, and why/when Zipf's law is broken. See [Mandelbrot (1955)], [Clauset et al. (2009)], [Cancho, Sole (2001)], [Gerlach, Altman (2013)] and [Williams et al. (2015)].

3.2 Pareto Type Distributions and Bible Translations

From each translation of an identical Bible version a word frequency analysis is fashioned and words are ranked by use. Let N_v denote vocabulary size (number of used words) and N_t the total number of words used in the investigated text. These are

[1] [A] The formulation of the altered R squared measure has been missing from the published version.

[2] [A] Section labels added.

easily determined by tools such as [Word Count App], [WriteWords]; let ρ and f denote rank and frequency, respectively. The relative parameters are thus $\rho_r = \rho/N_v$ and $f_r = f/N_t$. At this point we like to note, that the original paper [Mehri, Jamaati (2017)] was ranking the frequencies successively, meaning each rank was only given once and no two words could share the same value. This of course leads to a large amount of low-frequency words which occur only once or twice, covering a large range of ranks. This leads to the development of rank bands (Figure 3.1). This is the result of successive ranking, for same frequencies. For example, if 5 words have the same frequency, assigning the ranks $i+1, \ldots, i+5$ would create such as vertical line.

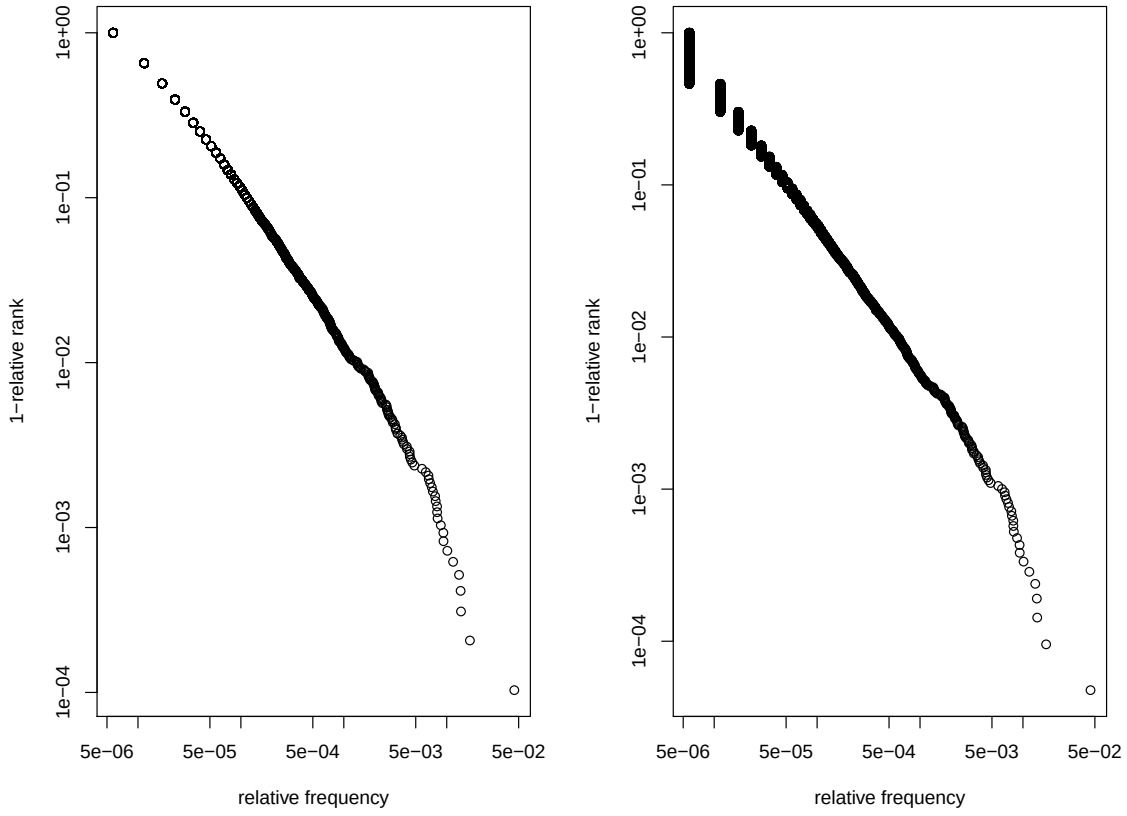


Figure 3.1: A comparison for the achuar language of the multiple ranks (left) and the single rank approach (right).

We believe this successive ranking to be misleading, since given a large enough data set, words with the same frequency will display a difference in ranking in the hundreds or thousands, and words with the same number of occurences should have the same rank. It is easy to see, how these bands will cause the deviation error to have a certain static base error, since no distribution will capture the entire band. This obscures the results, since the differences in goodness of fit would be miniscule. We have therefore chosen to allow multiple equal rank for equal frequencies, see the right hand side plot

in 3.1. [3]

We seek to establish a relation similar to Zipf's relation, that has ostensibly been introduced in [Mehri, Jamaati (2017)]:[4]

$$1 - \rho_r = \exp\left[a \log\left(f_r\right) + b\right],$$

where $a \in \mathbb{R}^-$ and $b \in \mathbb{R}$. Along with this formulation we provide the performance results of the original relation provided by the Zipf law and the Zipf-Mandelbrot version given below.

$$freq_{Zipf}(\rho; s, N) = \frac{\frac{1}{\rho^s}}{\sum_{i=1}^{N} \frac{1}{i^s}} \qquad \text{and} \qquad freq_{Zipf-M}(\rho; s, q, N) = \frac{\frac{1}{(\rho+q)^s}}{\sum_{i=1}^{N} \frac{1}{(i+q)^s}},$$

for

where $\rho \in \mathbb{N}$ is the absolute rank and s, q the respective distribution parameters. [5]

As we will see in later plots, this relationship manifests itself as a straight line in a Log-Log plot. Especially in both lower and upper tails the distribution does not accurately capture the expected ranking. Down below we have listed the cumulative distribution functions (CDFs) of the tested Pareto-type distributions and the tested relationships between relative rank and frequency:

$$F_{P-I}(x) = 1 - \left[\frac{x}{\sigma}\right]^{-\alpha}, \qquad \rho_r = \left[\frac{f_r}{\sigma}\right]^{-\alpha},$$

$$F_{P-II}(x) = 1 - \left[1 + \frac{x-\mu}{\sigma}\right]^{-\alpha}, \qquad \rho_r = \left[1 + \frac{f_r-\mu}{\sigma}\right]^{-\alpha},$$

$$F_{P-III}(x) = 1 - \left[1 + \left(\frac{x-\mu}{\sigma}\right)^{1/\gamma}\right]^{-1}, \qquad \rho_r = \left[1 + \left(\frac{f_r-\mu}{\sigma}\right)^{1/\gamma}\right]^{-1},$$

and

$$F_{PGPD}(x) = 1 - \left[1 + \xi\left(\frac{x-\mu}{\sigma}\right)\right]^{-1/\xi}, \qquad \rho_r = \left[1 + \xi\left(\frac{f_r-\mu}{\sigma}\right)\right]^{-1/\xi}.$$

[3] [A] Caption has been amended to clearly describe which plot describes which ranking approach.

[4] The relation was not properly defined in [Mehri, Jamaati (2017)], but appeared to be a linear relation in a log-plot. We hence used this modified version in addition to Zipf's original and the Zipf-Mandelbrot law, to cover all possible variations.

[5] [A] The parameters are not named, nor do they immediately refer to properties of the distribution.

The parameters $\alpha > 0$, $\gamma > 0$ and $-\infty < \xi < \infty$ control the shape of these distributions. The parameter $\sigma > 0$ controls the scale of these distributions. The parameter $-\infty < \mu < \infty$ controls the location of these distributions. Smaller values of α correspond to heavier tails of the Pareto type I-III distributions. Larger values of γ correspond to heavier tails of the Pareto type III distribution. The generalized Pareto distribution has a finite tail if $\xi < 0$. It has an infinite tail if $\xi \geq 0$. The exponential distribution is the limiting case of the generalized Pareto distribution for $\xi \to 0$. The Pareto type II distribution is the particular case of the Pareto type III distribution for $\gamma = 1$. The Pareto type II distribution is a location scale variant of the Pareto type I distribution.

The four Pareto-type distributions and Zipf's law were fitted to the data by the minimization of the square deviation $\frac{1}{N}\sum_{i=1}^{N}\left(\rho_{i,r} - \hat{\rho}(f_{i,r})\right)^2$ of the predicted relative rank through a function of the relative frequency to the observed relative rank. Since the relation which was observed in [Mehri, Jamaati (2017)] was based on the CDFs of distributions, this was a more direct approach than density based approaches, say the MLE for example. We therefore provide the aggregated squared error, the Kolmogorov-Smirnov statistic as well as the R^2 measure on untransformed data, to diversify our measures. The squared error was minimized using the routine optim in the R software [R Development Team (2017)]. The routine uses a quasi Newton algorithm. The log-transformations are not considered at this point, that is during optimization. The logarithmic data is used solely to provide a better display of performances.

The fit of the distributions for eight exemplary translations are shown in Figure 3.2 (Inverse CDF in Appendix B.1). [6] The picture is similar for all other languages. Zipf's method (red) does not adequately cover the curvature of the word distribution, whereas all other Pareto distributions offer a better fit. The Pareto type III distribution repeatedly exhibits a squared error less than 0.5 thus providing the best overall performance, see Tables 3.1, 3.2, and Tables B.1,B.2,B.3,B.4,B.5 and B.6 for further texts and Tables 3.4, 3.5 and 3.6 for results by language family. One of the three distributions also provides the best overall performance with respect to the KS statistic. We have added the measure multiplied by the number of their parameters, to penalize for their complexity, yet the results remain the same, since the improvements of fit are influenced by the different number of parameters.

The KS statistic, squared error value and R^2 value are shown in Tables 3.1 and

[6] [A] Several figures have been moved to Appendix B, if they were not making a new point, but rather supplementary information.

3.2 for the eight selected languages. The KS statistic for the Pareto type III distribution stays below the critical value for all instances. The standard Pareto distribution shows the largest deviation amongst the Pareto type distributions, yet still outperforms Zipf's law.

Figure 3.2 (and B.1) provide further evidence for what we had already assumed. In these figures, the Pareto type II and generalized Pareto distributions overlap, showing identical results in the comparative tables. The Pareto type II distribution can be emulated by the generalized Pareto distribution and vice versa by setting $\alpha = 1/\xi$ and $\xi = 1$. In Figure 3.4 the Pareto type III distribution provides a tight grouping around 0 and 1 for the KS test statistic and the R squared measure, respectively. The Pareto type III distribution outperforms Zipf's law almost without fail.

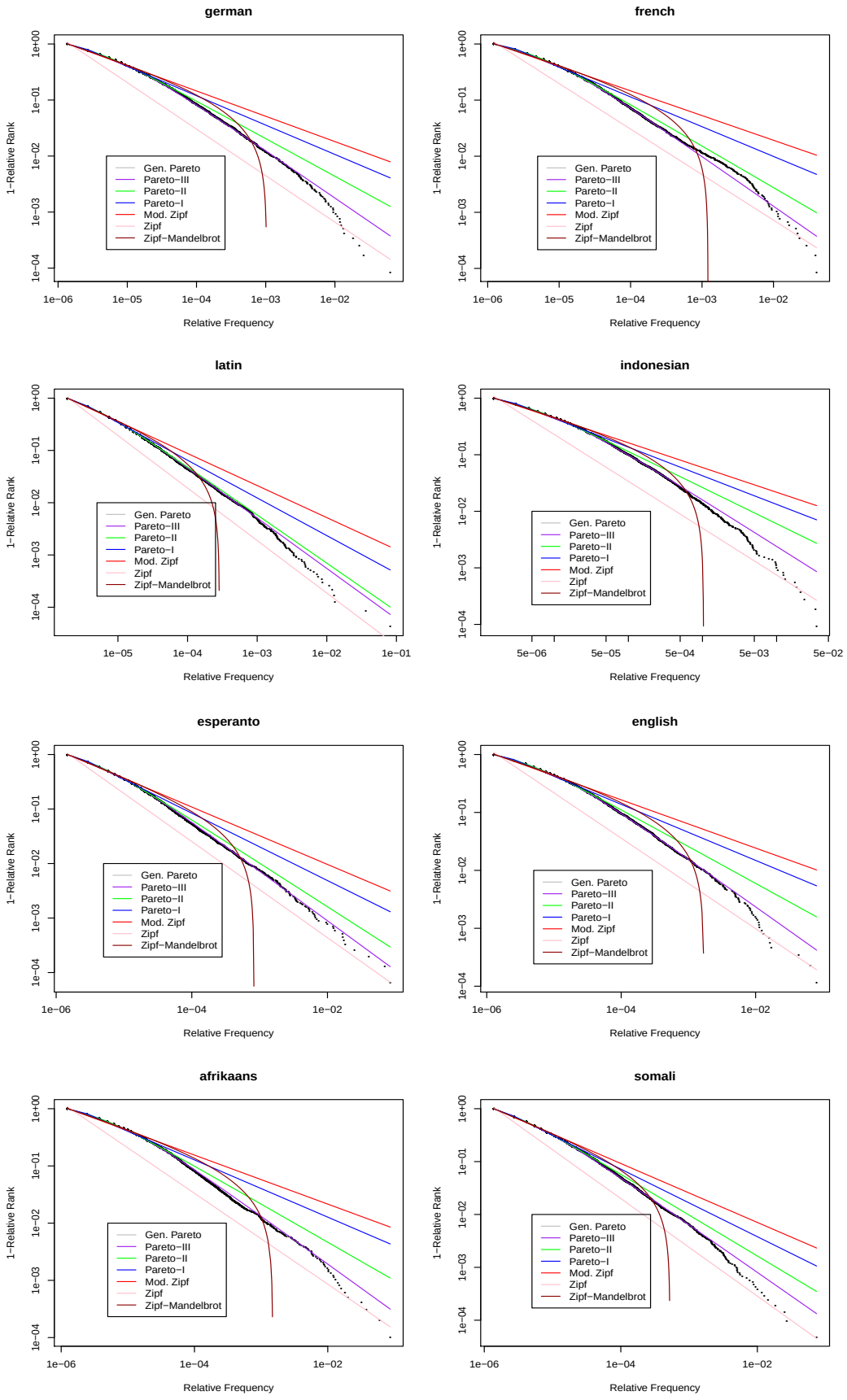


Figure 3.2: Log-Log inverse CDF plot of word relative frequency versus relative rank.

	Distribution	KS statistic	Squared Error	R squared	Sq. Error x DoF
German	Gen. Pareto Dist.	0.012941	0.4847	0.9997585	1.4541
	Pareto Dist. Type III	0.007695	0.2879	0.9998566	0.8636
	Pareto Dist. Type II	0.012939	0.4847	0.9997585	1.4541
	Pareto Dist. Type I	0.036879	4.4943	0.9977609	8.9888
	Zipf-Mandelbrot Law	0.066079	7.8892	0.9960695	15.7785
	Zipf's Power Law (mod.)	0.062875	16.0795	0.9919891	32.1590
	Zipf's Power Law (orig.)	0.232340	415.5079	0.7929907	415.5079
French	Gen. Pareto Dist.	0.011481	0.2173	0.9998815	0.6517
	Pareto Dist. Type III	0.008465	0.2183	0.9998809	0.6518
	Pareto Dist. Type II	0.011485	0.2173	0.9998815	0.6517
	Pareto Dist. Type I	0.042907	5.8038	0.9968332	11.6075
	Zipf-Mandelbrot Law	0.058310	15.6081	0.9914835	31.21628
	Zipf's Power Law (mod.)	0.076270	25.2547	0.9862199	50.5094
	Zipf's Power Law (orig.)	0.236336	434.2414	0.7630579	434.2414
Esperanto	Gen. Pareto Dist.	0.010794	0.2820	0.9999024	0.8460
	Pareto Dist. Type III	0.005557	0.0264	0.9999909	0.0791
	Pareto Dist. Type II	0.010795	0.2820	0.9999024	0.8460
	Pareto Dist. Type I	0.032084	3.7774	0.9986926	7.5548
	Zipf-Mandelbrot Law	0.0457038	11.0022	0.9961921	22.0044
	Zipf's Power Law (mod.)	0.060283	18.0983	0.9937361	36.1966
	Zipf's Power Law (orig.)	0.195324	415.5079	0.8602944	403.6524
English	Gen. Pareto Dist.	0.015998	0.5933	0.9995604	1.7799
	Pareto Dist. Type III	0.008585	0.1792	0.9998672	0.5377
	Pareto Dist. Type II	0.015996	0.5933	0.9995604	1.7799
	Pareto Dist. Type I	0.043663	4.7441	0.9964850	9.4882
	Zipf-Mandelbrot Law	0.062942	7.9992	0.9940733	15.9984
	Zipf's Power Law (mod.)	0.070629	14.8486	0.9889985	29.6971
	Zipf's Power Law (orig.)	0.241482	326.2466	0.7582786	326.2466

Table 3.1: Kolmogorov-Smirnov statistic, squared error value and the R squared value for four selected languages (Part I).

	Distribution	KS statistic	Squared Error	R squared	Sq. Error x DoF
Latin	Gen. Pareto Dist.	0.063925	0.1266	0.9999746	0.3797
	Pareto Dist. Type III	0.065341	0.0375	0.9999925	0.1125
	Pareto Dist. Type II	0.063925	0.1266	0.9999746	0.3797
	Pareto Dist. Type I	0.023692	3.1248	0.9993729	6.2496
	Zipf-Mandelbrot Law	0.067431	9.5414	0.9980852	19.0827
	Zipf's Power Law (mod.)	0.052449	20.6222	0.9958614	41.2445
	Zipf's Law (orig.)	0.187220	571.0397	0.8853997	571.0397
Indonesian	Gen. Pareto Dist.	0.016838	0.5856	0.9996690	1.7569
	Pareto Dist. Type III	0.010506	0.4492	0.9997461	1.3476
	Pareto Dist. Type II	0.016836	0.5856	0.9996690	1.7569
	Pareto Dist. Type I	0.039726	3.9837	0.9977487	7.9674
	Zipf-Mandelbrt Law	0.074594	6.0653	0.9965722	12.1306
	Zipf's Power Law (mod.)	0.065197	14.8299	0.9916190	29.6599
	Zipf's Power Law (orig.)	0.235931	412.3725	0.7669506	412.3725
Afrikaans	Gen. Pareto Dist.	0.019155	0.7884	0.9994930	2.3651
	Pareto Dist. Type III	0.009418	0.2494	0.9998396	0.7481
	Pareto Dist. Type II	0.019036	0.7882	0.9994931	2.3651
	Pareto Dist. Type I	0.047467	5.7356	0.9963114	11.4711
	Zipf-Mandelbrot Law	0.061464	9.8136	0.9936887	19.6273
	Zipf's Power Law (mod.)	0.074162	17.4008	0.9888094	34.8016
	Zipf's Law (orig.)	0.238492	356.1892	0.7709310	356.1892
Somali	Gen. Pareto Dist.	0.007878	0.1362	0.9999664	0.4086
	Pareto Dist. Type III	0.008510	0.4814	0.9998814	1.9255
	Pareto Dist. Type II	0.007877	0.1362	0.9999664	0.4086
	Pareto Dist. Type I	0.022617	2.2622	0.9994425	4.5244
	Zipf-Mandelbrot Law	0.047973	7.3160	0.9981971	14.6320
	Zipf's Power Law (mod.)	0.045634	14.6678	0.9963854	29.3356
	Zipf's Power Law (orig.)	0.186582	503.0458	0.8760347	503.0458

Table 3.2: Kolmogorov-Smirnov statistic, squared error value and the R squared value for four selected languages (Part II).

We investigate the fit of the proposed distributions on randomly generated texts. These types of texts are commonly used in graphics design and other text processing or editorial tasks, to serve as a place holder for the later inserted content. These texts are not meant to follow grammar rules or be intelligable, but it is often preferred that they do follow sentence structures to closer emulate real texts, thus allowing for better typesetting. The choice of words normally is not derived from word frequencies, but rather based on sentence structure and the co-occurence of certain word-classes (most simple case, noun-verb-object). The results can be found in Figure 3.3 and Table 3.3. We can see an overall decline in the performance of all distributions, since the generated texts display distinct plateaus in their frequency distributions, which cannot be accurately covered by any distribution we have investigated. However, we can confirm the findings of [Cancho, Elvevag (2010)], that random texts are not adequately described by Zipf's law, similar to our findings with real-life literary texts. Especially the longer randomly generated texts show how the Pareto-type distributions outperform the standard and modified Zipf laws. We deduced a modified Zipf law from the plots in [Mehri, Jamaati (2017)] as the relation given in Equation 3.3. [7]

$$f_r = \exp\left(a + b\log(\rho_r)\right) \text{ with } a \in \mathbb{R}^+, b \in \mathbb{R} \tag{3.3}$$

To verify our findings, and prove the general validity of the conclusions we reach, we have repeated the procedure for a representative selection of different languages with additional single author texts, available in a number of translations (see [Collodi (1883)],[Saomt-Exupery (1943)] and [Marx (1867)]).

The results are summarised in Tables B.1-B.6,listing the goodness of fit of every distribution investigated to each one of the literary works in question. We have tried to incorporate different subject matters and lengths of texts, to further diversify the type of text. Figures B.2 to B.7 show similar comparative plots as for the Bible translations. For all three texts, we can see the same picture as for both the Bible translations and random texts. [8]

[7] [A] The explicit formulation of the modified Zipf-power law has been missing from both the published version and the original paper. [Mehri, Jamaati (2017)]

[8] [C] The plots and tables are more or less the same as we created for the Bible translations, hence we have transferred these to Appendix B. However, the similar performance reaffirms our claim that the Pareto distributions are preferable to the Zipf law regardless of literature.

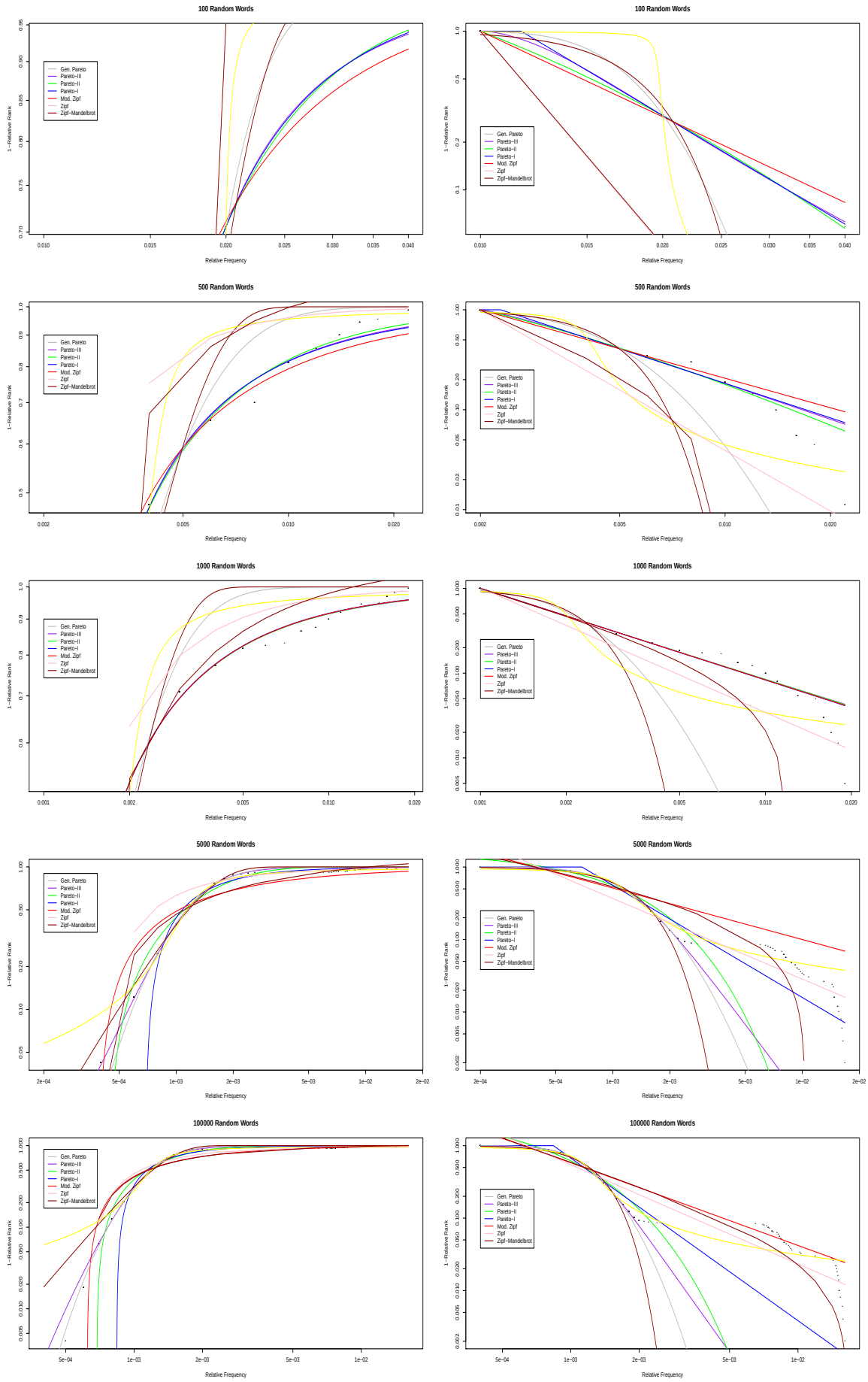


Figure 3.3: CDF and Inverse CDF for randomly generated texts of differents lengths.

	Pareto III		**Pareto II**		**Pareto I**	
	Sq. Error	KS	Sq. Error	KS	Sq. Error	KS
100	0.000022	0.004100	0.000004	0.000592	0.000005	0.002043
500	0.043866	0.060563	0.034992	0.050022	0.047384	0.063213
1000	0.026140	0.037953	0.026867	0.037523	0.026919	0.036400
5000	0.106204	0.006427	1.598728	0.282884	1.241508	0.121458
10000	0.119651	0.008050	2.920860	0.549974	1.295437	0.127016
Average	0.059177	0.023419	0.916290	0.184199	0.522251	0.070026

	Zipf (mod.)		**Log-Normal**		**Burr**	
	Sq. Error	KS	Sq. Error	KS	Sq. Error	KS
100	0.002396	0.024533	0.040302	0.003188	0.205219	0.038105
500	0.081811	0.084213	1.112249	0.079027	2.130549	0.107465
1000	0.026939	0.036709	1.931369	0.053344	3.600039	0.076878
5000	8.156660	0.605744	0.137575	0.021195	0.285452	0.020260
10000	7.139636	0.653773	0.137968	0.007791	0.453746	0.028206
Average	3.081488	0.280994	0.671893	0.032909	1.335001	0.054183

	Log-Cauchy		**Zipf (org.)**		**Zipf-Mandelbrot**	
	Sq. Error	KS	Sq. Error	KS	Sq. Error	KS
100	0.039486	0.000127	0.790015	0.249888	0.779849	0.248997
500	2.028688	0.017787	4.663579	0.274449	3.215130	0.248954
1000	2.616131	0.019662	1.927380	0.110341	0.231910	0.098123
5000	0.251352	0.040519	20.697590	1.203869	6.503947	0.480730
10000	0.028206	0.301588	0.030985	8.588758	0.913065	6.884853
Average	1.047449	0.021816	7.333464	0.550322	3.523138	0.336576

Table 3.3: Error Measures for randomly generated texts of different lengths.

	Pareto III		**Pareto II**		**Pareto I**	
Family	Sq. Error	KS	Sq. Error	KS	Sq. Error	KS
tai-kadai	0.065302	0.003905	0.016737	0.010281	0.781058	0.006824
sino-tibetan	5.068242	0.024913	3.927167	0.029489	4.548020	0.047344
japonic	0.018946	0.003123	0.001569	0.015856	0.066971	0.003066
basque	0.156630	0.008545	0.136694	0.009928	0.604888	0.023432
afro-asiatic	0.283393	0.009507	0.206428	0.010692	0.758778	0.015251
arawakan	0.009370	0.002996	0.016206	0.002483	0.046477	0.008969
iroquoian	0.405554	0.013269	0.402185	0.009065	0.414602	0.010478
dravidian	1.422745	0.334967	0.148587	0.017108	0.774644	0.352427
quechuan	0.025806	0.004399	0.039814	0.006743	0.565217	0.019794
uralic	1.342475	0.003346	0.136192	0.005012	1.739880	0.018118
algic	0.025976	0.003481	0.019568	0.003645	0.010523	0.003014
jivaroan	0.032366	0.005167	0.035617	0.005568	0.246817	0.012621
niger-congo	0.130750	0.003792	0.091129	0.008542	0.399769	0.016327
indo-european	1.101680	0.014092	0.984024	0.018624	3.322701	0.033752
constructed	0.026372	0.004063	0.282008	0.010371	3.777402	0.031718
equatorial	0.111452	0.005852	0.048002	0.006193	0.457512	0.016752
uto-aztecan	0.053420	0.003911	0.019017	0.003392	0.069841	0.007244
tucanoan	0.944945	0.000193	0.043409	0.003561	0.173757	0.011925
austronesian	1.852265	0.013659	1.257256	0.035925	2.228201	0.028665
altaic	0.739430	0.011218	49.646454	0.054695	3.324282	0.060736
austro-asiatic	4.561460	0.045128	11.855410	0.076074	16.119360	0.089358
oto-manguean	0.184014	0.013816	0.069150	0.014102	0.361999	0.024272
carib	0.217801	0.001357	0.190786	0.017052	0.843702	0.033886
west-papuan	0.096526	0.000007	0.017813	0.002130	0.142358	0.015625
nilo-saharan	0.303121	0.002407	0.223856	0.016738	1.497231	0.034246
chibchan	0.184356	0.032563	0.003739	0.004996	0.004843	0.005607
mayan	0.806186	0.043184	1.440334	0.043389	1.997668	0.055113
creole	0.290654	0.014703	0.261785	0.031554	0.663347	0.044242
DoF	3	3	3	3	2	2
Weighted Avrg.	**0.9471**	**0.02323**	**1.8662**	**0.0191**	**2.1658**	**0.0390**
AVG*DoF	2.8412	0.0697	5.5985	0.0573	4.3316	0.0779

Table 3.4: Distribution performances grouped by language family (Pareto I-III).

	Log-Normal		**Burr**		**Log-Cauchy**	
Family	Sq. Error	KS	Sq. Error	KS	Sq. Error	KS
tai-kadai	11.48706	0.001752	543.9214	0.437138	11.6002	0.000007
sino-tibetan	187.4464	0.059947	724.5601	0.113205	333.8738	0.066933
japonic	0.165725	0.000009	73.8866	0.721251	0.165653	0.000005
basque	68.0338	0.075123	125.8344	0.107579	139.9554	0.098658
afro-asiatic	108.5567	0.086064	218.1322	0.121146	234.9150	0.098667
arawakan	78.7972	0.072113	187.2809	0.129435	172.7570	0.012430
iroquoian	107.7848	0.095982	229.7598	0.142486	214.3262	0.081926
dravidian	151.8516	0.388227	337.4577	0.407341	353.4760	0.412338
quechuan	62.6687	0.076391	116.3066	0.092716	127.3862	0.101962
uralic	197.3718	0.083800	395.1414	0.102263	426.3963	0.115003
algic	86.0274	0.087877	259.4433	0.178275	157.2805	0.009448
jivaroan	89.0697	0.085011	168.8642	0.116562	202.3414	0.092674
niger-congo	113.4743	0.083661	245.5685	0.113876	252.0589	0.105271
indo-european	117.5473	0.075931	214.2465	0.100760	243.6846	0.102248
constructed	100.9535	0.064536	195.8095	0.093275	215.2511	0.096269
equatorial	66.9886	0.075920	120.2282	0.108957	132.3156	0.095931
uto-aztecan	31.9293	0.083293	55.5848	0.101172	67.0228	0.121929
tucanoan	48.5471	0.072079	88.8384	0.104550	95.2285	0.094956
austronesian	76.5910	0.074512	137.7862	0.100597	155.1827	0.101208
altaic	92.7943	0.069873	194.0101	0.099586	237.8325	0.108679
austro-asiatic	26.9776	0.091200	47.9036	0.082134	55.2344	0.085804
oto-manguean	37.9419	0.080740	68.6705	0.102466	77.5479	0.103263
carib	24.4751	0.067208	45.5266	0.097894	47.9234	0.093342
west-papuan	15.5638	0.078274	40.9342	0.072608	29.9923	0.170894
nilo-saharan	31.0193	0.064340	59.2493	0.089631	62.0644	0.090875
chibchan	2.285363	0.084615	5.547749	0.125775	3.0411	0.098307
mayan	50.25648	0.083073	88.9309	0.103374	96.4888	0.106407
creole	8.8864	0.067522	16.6969	0.092612	15.2480	0.079270
DoF	3	3	2	2	2	2
Weighted Avrg.	**98.1492**	**0.0859**	**207.7725**	**0.12568**	**205.3839**	**0.1058**
AVG*DoF	294.4476	0.2580	415.5450	0.251369	410.7678	0.2115

Table 3.5: Distribution performances grouped by language family (Log-normal, Burr and Log-Cauchy).

	Zipf-Mandelbrot		**Zipf (mod.)**		**Zipf (orig.)**	
Family	Sq. Error	KS	Sq. Error	KS	Sq. Error	KS
tai-kadai	151.3369	0.285007	0.781058	0.006826	151.3423	0.285010
sino-tibetan	178.1839	0.173669	15.873243	0.070492	643.6726	0.305140
japonic	3.535142	0.168982	0.066971	0.003064	3.536550	0.169014
basque	1.553431	0.096101	4.865370	0.044303	195.6224	0.185665
afro-asiatic	4.606928	0.097665	5.459691	0.033900	233.2255	0.171364
arawakan	1.141258	0.071912	1.449499	0.027142	105.0363	0.143394
iroquoian	0.989415	0.048020	0.455707	0.013352	89.3474	0.139589
dravidian	6.015302	0.078935	4.297268	0.361656	371.7396	0.197567
quechuan	1.477274	0.092096	4.439572	0.041852	174.3442	0.182790
uralic	5.421048	0.049767	11.47364	0.039835	394.4552	0.166132
algic	5.708840	0.103938	0.090348	0.007685	82.2895	0.149091
jivaroan	1.376517	0.058414	2.987229	0.030980	148.0259	0.144993
niger-congo	4.916187	0.072571	2.798109	0.032948	207.4293	0.174900
indo-european	6.861703	0.068882	11.80852	0.055665	330.2769	0.191888
constructed	11.00218	0.045704	18.09828	0.058735	403.6524	0.195324
equatorial	1.603569	0.029523	3.055777	0.032649	133.6798	0.159985
uto-aztecan	1.079570	0.025091	1.255562	0.025568	18.43536	0.086134
tucanoan	2.170866	0.031344	2.959732	0.034879	86.08112	0.140711
austronesian	6.634514	0.054628	10.170533	0.052553	238.8151	0.189547
altaic	13.41288	0.186525	27.993004	0.093638	578.7161	0.322971
austro-asiatic	7.814408	0.139316	16.386250	0.093848	321.0372	0.396108
oto-manguean	0.670217	0.041358	1.876963	0.038708	89.61647	0.188926
carib	1.173454	0.055765	2.917771	0.054351	99.66124	0.196119
west-papuan	1.347248	0.044067	1.823594	0.051011	32.73058	0.144357
nilo-saharan	2.286997	0.049582	5.219225	0.053364	174.1898	0.223889
chibchan	0.061381	0.016976	0.085436	0.018528	2.194719	0.090855
mayan	2.560421	0.074116	4.518950	0.072316	156.1429	0.195163
creole	0.235162	0.051215	1.310209	0.054700	61.63103	0.232228
DoF	2	2	2	2	1	1
Weighted Avrg.	**12.1220**	**0.0776**	**8.2466**	**0.0583**	**262.6190**	**0.1914**
AVG*DoF	24.2441	0.1552	16.4933	0.1166	262.6190	0.1914

Table 3.6: Distribution performances grouped by language family (Zipf-Mandelbrot, modified Zipf and original Zipf).

In Tables 3.4-3.6 we have provided a data table of the results, grouped by language family along with a listing of the respective groupings (Table 3.7). The overall results mirror our previous findings, but we can find aberrations for specific groups of languages. An example are the altaic languages, where the Pareto II distribution performs severly worse than the Zipf-Mandelbrot law.

Language Families

afro-asiatic	**chibchan**	danish	lukpa
hebrew	cabecar	spanish	
amharic		italian	**nilo-saharan**
coptic	**constructed**	french	dinka
tachelhit	esperanto	nepali	zarma
syriac		afrikaans	
wolaytta	**creole**	german	**oto-manguean**
tuareg	aukan	portuguese	amuzgo
somali	creole	swedish	chinantec
kabyle		norwegian	
arabic	**dravidian**	manx	**quechuan**
	telugu	english	quichua
algic	malayalam	gujarati	
potawatomi	kannada	romani	**sino-tibetan**
ojibwa			chinese
	equatorial	**iroquoian**	myanmar
altaic	camsa	cherokee	paite
turkish			
korean	**indo-european**	**japonic**	**tai-kadai**
	farsi	japanese	thai
arawakan	latvian		
campa	ukranian	**jivaroan**	**tucanoan**
	armenian	shuar	barasana
austro-asiatic	lithuanian	aguaruna	
vietnamese	croatian	achuar	**uralic**
	latin		hungarian
austronesian	hindi	**mayan**	finnish
chamorro	albanian	kiche	estonian
indonesian	polish	qeqchi	
uma	czech	uspanteco	**uto-aztecan**
malagasy	slovene	cakchiquel	nahuatl
cebuano	russian	jakalteko	
tagalog	bulgarian	mam	**west-papuan**
maori	greek		galela
	slovak	**niger-congo**	
basque	romanian	zulu	
basque	icelandic	xhosa	
	serbian	ewe	
carib	marathi	swahili	
akawaio	gaelic	wolof	

Table 3.7: Grouping of all analysed languages into their respective families

For for the japonic languages (japanese) and chibchuan we see a strong performance of the logarithmic distributions and the original Zipf law, which stand out distinctly from the remaining findings. These remain isolated cases though, and the Pareto III distribution specifically never completely fails to capture the behaviour of any given language translation.

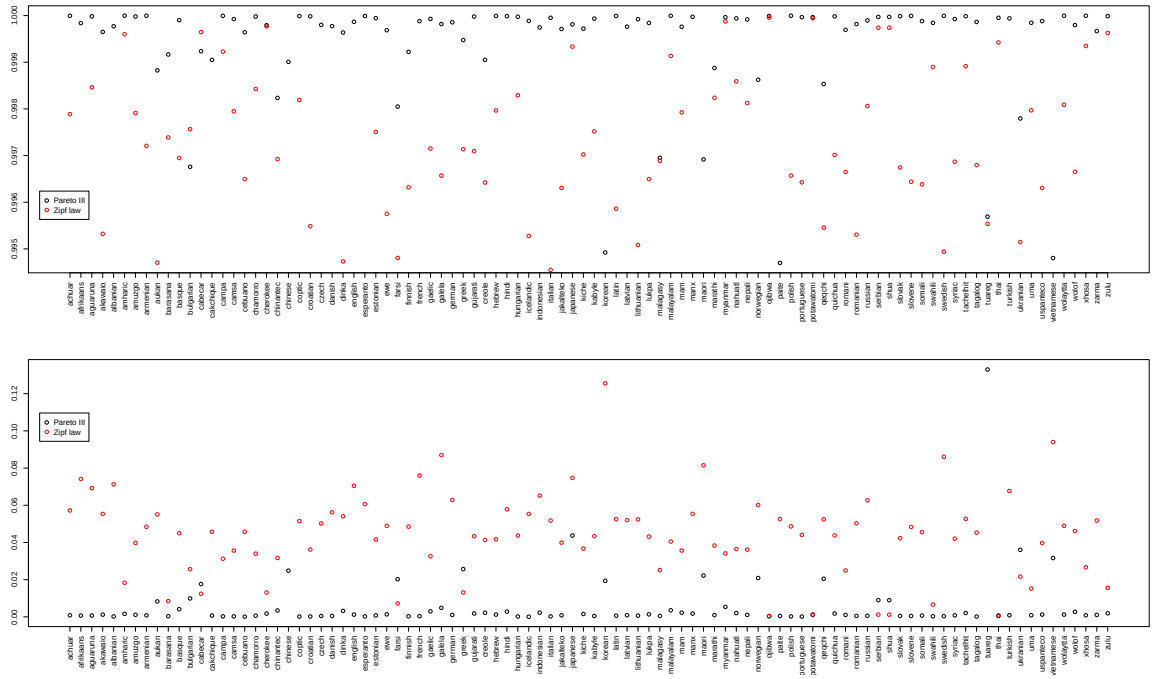


Figure 3.4: The R squared measure (top) and K-S statistic (bottom) for all languages compared.

3.3 Conclusions

We have tested a number of distributions to accurately capture word frequencies. In comparison to the more established Zipf's law the Pareto type distributions have performed far better, supported by the Log-Log plots as well as the number of error measures we have calculated.

The greater flexibility of the newly introdced distribution expectedly resulted in a better fit. Even though the Pareto distributions possess more parameters, the improvement of the fit is disproportionate to the increased complexity. The various incarnations of the Zipf law are in essence a rigid line on a Log-Log plot, which works somewhat for the body of the samples, but the Pareto-type distributions greatly improve the tail fit on boths ends of the frequencies.

It is evident to us that the models in this chapter are superior to Zipf's law and could greatly improve future modelling approaches. [9]

[9][C] A common question has been why the sectioned models of Chapter 2 hasn't been tested. Previous models such as the Zipf law are fairly simple in their construction, and we were aiming to deliver a comparatively simple approach. However, a multi-sectioned distribution is definitely considerable, as the different properties of different word classes may be much better captured by sectioning the data. Due to the amount of literary works required to challenge the Zipf law's claim of universality, we have delegated this approach to a future work.

Chapter 4

CompDist: Multisection Composite Distributions

Chapter Abstract

We introduce a new R package and its functions written by the authors of this chapter. The package contains multi-sectioned composite distributions, frequently used for financial data among a multitude of applications. It offers a wide range of standard distributions for the individual sections.

4.1 Introduction

Composite Distributions are a popular modelling approach for data where distinct ranges of the data exhibit significantly different behaviour from each other. We will give an overview of the applications and origin of one such model in Section 4.1. Section 4.2 introduces the novel R package and the standard functions associated with a distribution, such as PDF, CDF, quantile function or random data generation. We close by summarising the results and functions presented in this chapter in Section 4.3. [1]

Examples of financial applications are included in [Campiràn-Chaàve (2017)] with a focus on wealth distribution for publicly traded companies. Another set of examples can be found in Degen and Embrechts [Degen, Embrechts (2008)], mostly with regards to insurance models.

A two-part example for such a distribution model was given by Bakar and Nadarajah in [Bakar, Nadarajah (2013)]:

$$f(x) = \begin{cases} \frac{1}{1+\Phi}\frac{f_1(x)}{F_1(\theta)} & \text{if } x \leq \theta, \\ \frac{\Phi}{1+\Phi}\frac{f_2(x)}{1-F_2(\theta)} & \text{if } \theta < x, \end{cases}$$

$$F(x) = \begin{cases} \frac{1}{1+\Phi}\frac{F_1(x)}{F_1(\theta)} & \text{if } x \leq \theta, \\ \frac{1}{1+\Phi}\left(1 + \Phi\frac{F_2(x)-F_2(\theta)}{1-F_2(\theta)}\right) & \text{if } \theta < x. \end{cases}$$

The notation and the parameter constraints $\Phi \in \mathbb{R}^+$ and $\phi \in \mathbb{R}$ are the same as introduced by the original authors and Chapter 2 We generalise this distribution to an arbitrary number of sections. The PDF and CDF then take the respective forms depicted below:

$$f(x) = \begin{cases} 0 & \text{if } x < \theta_1, \\ \zeta\phi_i\frac{f_i(x)}{F_i(\theta_{i+1})-F_i(\theta_i)} & \text{if } \theta_i < x \leq \theta_{i+1} \text{ for } i = 1, \ldots, n_s, , \\ 0 & \text{if } \theta_{n+1} < x \end{cases}$$

$$F(x) = \begin{cases} 0 & \text{if } x < \theta_1, \\ \zeta\left(\sum_{j=1}^{i-1}\phi_j + \phi_i\frac{F_i(x)-F_i(\theta_i)}{F_i(\theta_{i+1})-F_i(\theta_i)}\right) & \text{if } \theta_i < x \leq \theta_{i+1} \text{ for } i = 1, \ldots, n_s, \\ 1 & \text{if } \theta_{n+1} < x. \end{cases}$$

[1] [A] Labels added.

Here θ_i marks the sections and ϕ_i the weights for each respective partial distribution, for $i = 1, \ldots, n_s + 1$, where n_s denotes the number of sections. We note that $\phi_1 = 1$ for the first section, therefore $\zeta = 1/(1 + \sum_{i=1}^{n_s-1} \phi_i)$ is the scaling factor. The partial F_i, f_i can be chosen from a list of distributions provided in the package, and cover the most commonly used heavy-tailed functions. The composite distributions, as well as the R package allow for the continuity and differentiability assumptions $\lim_{x\uparrow\theta_i} f(x) = \lim_{x\downarrow\theta_i} f(x)$ and $\lim_{x\uparrow\theta_i} \frac{\partial}{\partial x} f(x) = \lim_{x\downarrow\theta_i} \frac{\partial}{\partial x} f(x)$ for all $i = 1, \ldots, n$ to be fulfilled. This essentially equates to the composite density function f and its derivative having a continuous transition between sections i and $i + 1$ at the catenation point θ_i. For the assumptions to hold, the weights $\phi = \{\phi_1, \ldots, \phi_n\}$ and a distribution parameter of choice (or thresholds $\theta = \{\theta_1, \ldots, \theta_n\}$) can be adjusted accordingly (function included in the package). As we have up to two assumptions per section, two parameters have to be chosen in accordance with these requirements. Let $f_{\phi,\theta}$ be the composite density function, with the vector of weights ϕ and the thresholds θ. This means for $f_{l,\theta,\phi}(x)$ and $f_{r,\theta,\phi}(x)$ being the composite density (and $\frac{\partial}{\partial x} f_{\theta,\phi}(x) = f'$ their respective derivatives) on the left-hand and right-hand side of a threshold θ, respectively, we are left with the root finding problems 4.2:

$$\hat{\phi} = (\phi | f_{l,\theta,\Phi}(\theta) - f_{r,\theta,\phi}(\theta) = 0) \tag{4.1}$$

$$\hat{\theta} = \left(\theta | f'_{l,\theta,\Phi}(\theta) - f'_{r,\theta,\phi}(\theta) = 0\right) \tag{4.2}$$

The package handles this exactly in the same way, by minimising the difference between left hand side and right hand side of the density or its derivative at each catenation point, with respect to the weights, thresholds or even parameters of the section densities if possible. This is specified via the `borders` argument in each function, which defines the parameter ranges of the variables that are to be optimised to satisfy the continuity and differentiability arguments. If the argument is not specified, no such optimisation is carried out. The `par.pos` argument in turn defines which variable is minimised to fulfill the continuity condition, via its position in the parameter vector. Defaults are 1 for the weights, 2 for the thresholds, and successive numbers would then refer to the density parameters (for `dnorm` 3 would be the mean value or 4 the standard deviation). For density parameters we are always optimising the left-hand side density to fulfill the conditions.

Distribution Name	Selection Text	Number of Parameters
LogNormal Distribution	lnorm	2
Chi-Squared Distribution	chisq	1
F-Distribution	f	2
Generalized Extreme Value Distribution	gev	3
Burr Distribution	burr	2
Gompertz Distribtion	gompertz	2
Levy Distibution	levy	2
Pareto Distribution	pareto	2
Generalized Pareto Distribution	gpareto	2
LoMax Distribution	lomax	2
Rayleigh Distribution	rayleigh	1
Log- LaPlace Distribution	loglap	3
Gamma Distribution	gamma	2
Beta Distribution	beta	2
Exponential Distribution	exp	2
Weibull Distribution	weibull	2
Pearson Distribution	pearson	4

Table 4.1: Table of supported partial distributions

4.2 Function Usage and Examples

4.2.1 Density Function

We begin by generating a parameter list, for which we first choose the desired number and type of section distributions. In our example, we have used a LogNormal/ Gamma distribution. It is sufficient to type the handle names from the list below in Table 4.1 into a vector and pass it to the generating function `dcomp`.

This returns the distribution density. We now generate a parameter list, where the first argument is reserved for the θ values, e.g. the section intervals. The second argument is always used for the respective section weights ϕ, where the first entry is automatically set as 1, hence one less value is needed. The remaining list slots are taken up by the section distribution parameters.

Another important feature is the option of continuous and differentiable catenation points. This can be chosen by providing a list of parameter bounds, and a selection of the same which have to be handed down to the function.

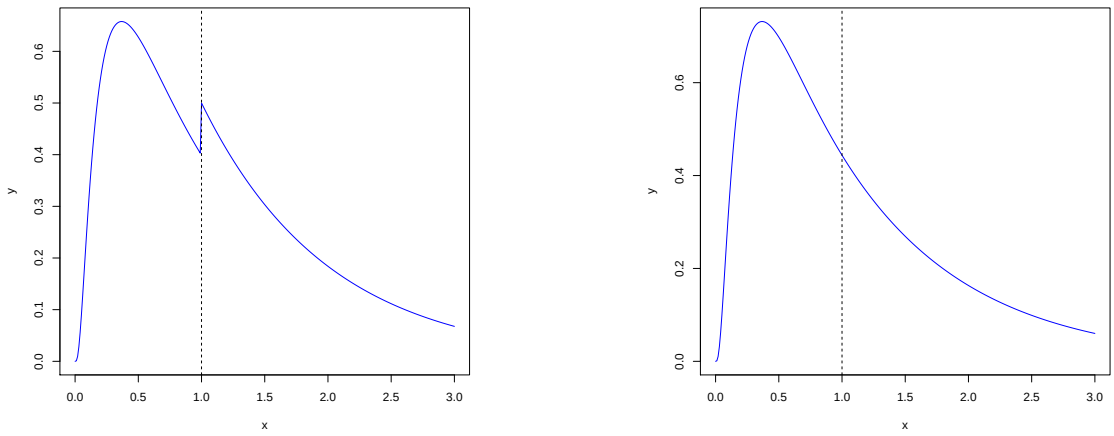
```
> par<-list()
```

```
> distvec<-c("lnorm","gamma")
> par[[1]]<-c(0,1,Inf)
> par[[2]]<-c(1)
> par[[3]]<-c(0,1)
> par[[4]]<-c(1,1)

# non-continuous case
> dnormgamma<-dcomp(distvec,par)

# or continuous case
> dnormgamma<-dcomp(distvec,par,borders=list(c(0.00001,10)))
> x<-seq(0,3,0.01)

> y<-dnormgamma(x)
> plot(x,y,type="l")
> abline(v=1)
```



(a) Density plot, non- cont.

(b) Density plot, continuous

Figure 4.1: Density plots

In Figure 4.1 one we can see the plot of the composite density. The horizontal line marks the catenation point between LogNormal and Gamma distribution.

4.2.2 Cumulative Distribution Function

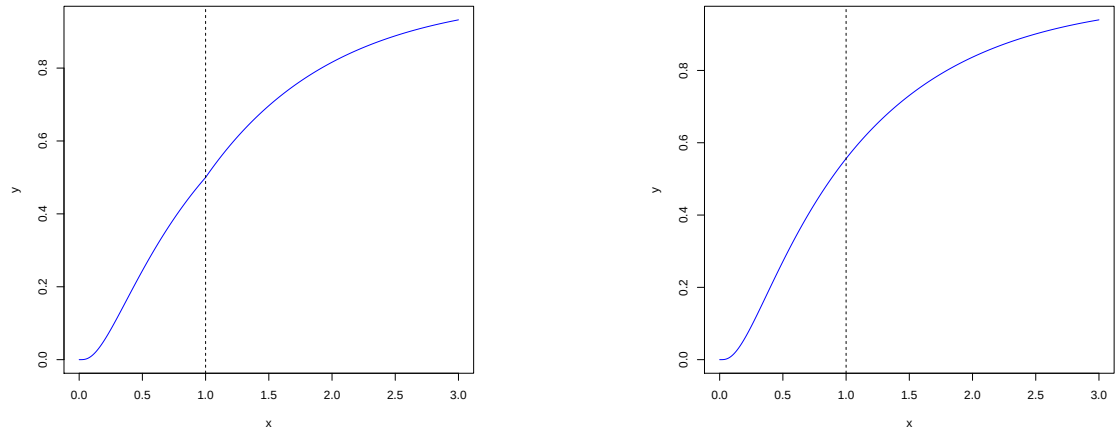
We proceed with a CDF for the same exemplary composite function. The parameter list has already been generated, hence we only need to load the CDF by `pcomp` with the distribution selection `distvec`. We use the resulting CDF similarly as the density before.

```
# non-continuous case
> pnormgamma<-pcomp(distvec,par)

# or continuous case
> pnormgamma<-pcomp(distvec,par,borders=list(c(0.00001,10)))
> x<-seq(0,3,0.01)
> y<-pnormgamma(x)

> plot(x,y,type="l",col="blue")
> abline(v=1,lty=2)
```

In Figure 4.2 we have plotted the output of the CDF in the previous paragraph. Again we have visualised the interval limit with a horizontal line.



(a) Cumulative distribution plot, non-cont.

(b) Cumulative distribution plot, continuous

Figure 4.2: CDF plots

4.2.3 Quantile Function

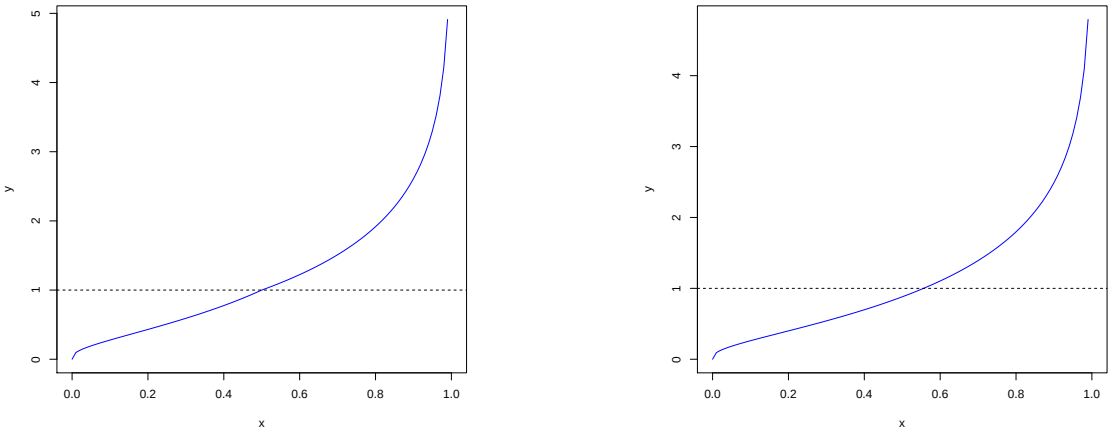
The quantile function works much in the same way. First, we initialise the function with the desired combination of listed distributions. We then can reuse the previously fashioned parameter list and plot the quantile function.

```
# non-continuous case
> qnormgamma<-qcomp(distvec,par)
# or continuous case
> qnormgamma<-qcomp(distvec,par,borders=list(c(0.00001,10)))
> x<-seq(0,1,0.01)
```

```
> y<-qnormgamma(x,par)

> plot(x,y,type="l",col="blue")
> abline(h=1,lty=2)
```

Again we notice the now horizontal line, which denotes the catenation point of the two partial distributions in Figure 4.3 below.



(a) Quantile function plot, non-cont. (b) Quantile function plot, continuous

Figure 4.3: Quantile function plots

We now move on to random sample generation.

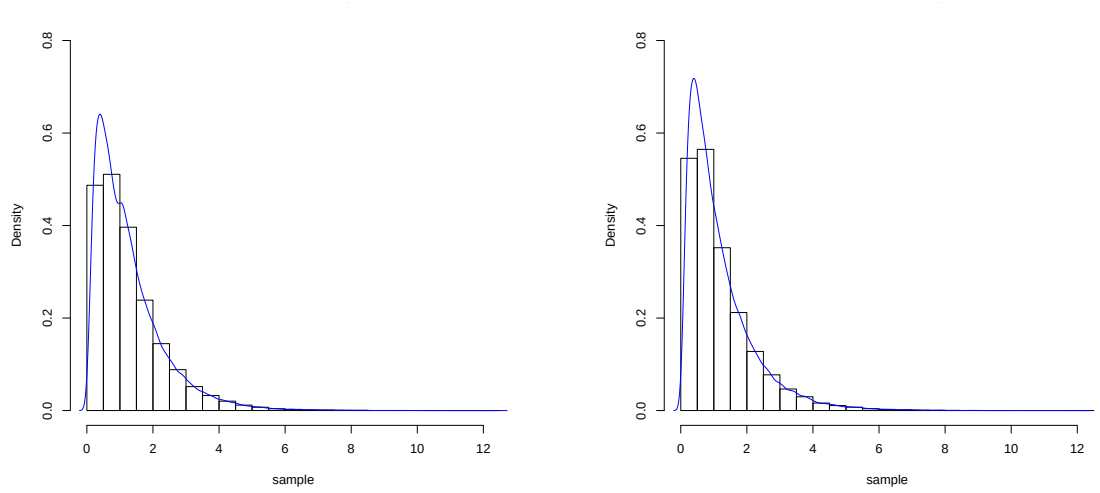
4.2.4 Random Sample Generation

The package also provides a function to generate random data according to the user specified function. As before we load the function `rcomp` with the desired distributions, and can generate a number of random samples with parameters `par`.

```
# non-continuous case
> rnormgamma<-rcomp(distvec,par)
# or continuous case
> rnormgamma<-rcomp(distvec,par,borders=list(c(0.00001,10)))

> sample<-rnormgamma(1000)
> hist(sample)\\
```

In Figure 4.4 we display a histogram of the random sample we have generated with `rcomp`. The first two bars mark the LogNormal part, whereas the rest of the data follows a Gamma distribution.



(a) Histogram vs density plot, non-cont. (b) Histogram vs density plot, continuous

Figure 4.4: Random Sample Plots

In the last subsection we conclude with the parameter fitting function `par.fit`.

4.2.5 Data Fitting

In this example, we generate LogNormal random data, and then fit the previously used LogNormal-Gamma composite distribution to it. This is done via `par.fit` with the function strings '`lnorm`' and '`gamma`' as before. Then we run the actual fitting process with a starting list of parameters, and the number of iterations we want the process to complete. The return value is a list of not only the parameter values but also the error values of the fitted distribution in regards to the data provided.

```
> sample<-rlnorm(10000)

# non-continuous case
> pfit<-par.fit(distvec,par)

# or continuous case
> pfit<-par.fit(distvec,par,borders=list(c(0.00001,10)))
> estimate<-pfit(sample,optit=2)
```

```
# non-continuous case
> dc<-dcomp(distvec,estimate$Parameter)

# or continuous case
> dc<-dcomp(distvec,estimate$Parameter,
  borders=list(c(0.00001,10)))

> x<-seq(0,30,0.1)
> y<-dc(x)

> hist(sample,probability = T,breaks=40)
> lines(x,y,col="red")

> estimate$Parameter
$Parameter[[1]]
[1] 0.000000 1.008731      Inf

$Parameter[[2]]
[1] 1.03252

$Parameter[[3]]
[1] -0.8185483  0.6273804

$Parameter[[4]]
[1] 2.592689 0.967091

$LogLikelihood
[1] 1548.73

$AIC
[1] -3091.46

$BIC
[1] -3076.736

$AICc
[1] -3091.436

$CAIC
[1] -3073.736
```

```
$HQC
[1] -3085.864
```

Also featured in Figure 4.4 is a histogram of the generated data and the fitted curve in red. Furthermore, we have provided the output of our example above, which quantifies the quality of fit.

4.3 Conclusions

We have developed a new R package which provides an arbitrary number of sections for composite models, for a wide array of commonly used distributions. With intuitive to use functions and options such as the continuity feature, we believe the provided software to be a powerful tool for multisection data sample analysis. The package could be easily modified to add more distributions as needed.

Chapter 5

General Moments for Roundoff Error

Chapter Abstract

Li and Nadarajah [Li, Nadarajah (2016)] derived expressions for mean and variance of roundoff error for any continuous random variable. Here, we derive expressions for general moments of the roundoff error, allowing one to study other aspects of roundoff error than just mean and variance. The expressions are specialized for 10 commonly used distributions in signal processing. Numerical studies checking the correctness of the derived expressions are given.

5.1 Introduction

Round off errors arise in many areas of signal processing: wave digital-filters, fast state-space decimator and polynomial FIR predictors and predictive FIR differentiators; to mention a few. See Li and Nadarajah [Li, Nadarajah (2016)] for other areas and references.

Two measures of roundoff errors are their mean and variance. Gadzhiev [Gadzhiev (2015)] derived expressions for these when X is a centered uniform or a centered Gaussian random variable. Li and Nadarajah [Li, Nadarajah (2016)] extended Gadzhiev [Gadzhiev (2015)] for any continuous random variable X defined on either the real line or a finite interval. But often higher order moments are of interest, not just the mean and variance. Examples include skewness and kurtosis. Skewness can be used to know if roundoff errors are more likely to be positive than negative. [1] Kurtosis can be used to know how concentrated roundoff errors are around zero.

The aim of this chapter is to derive general moments of roundoff errors for any continuous random variable X. Throughout, $\lfloor x \rfloor$ (respectively, $\lceil x \rceil$) denotes rounding to the largest (respectively, smallest) integer less (respectively, greater) than or equal to x. $\lfloor x + 1/2 \rfloor$ means rounding to the nearest integer. The derived expressions for moments of roundoff errors are simple. Simple computer programs have been written by the authors that implement the derived expressions for any continuous random variable X. The programs can be obtained from the corresponding author.

Various distributions have been used to model roundoff errors in the signal processing area: uniform distribution in Press [Press (1969)], Barnes et al. [Barnes et al. (1985)], Wong [Wong (1990)], Vladimirov and Diamond [Vladimirov, Diamond (2002)], Csordas et al. [Csordas et al. (2003)]; normal distribution in Ardalan and Alexander [Ardalan, Alexander (1987)], Yu and Lim [Yu, Lim (2006)]; trapezoidal distribution in Kawarai and Murakami [Kawarai, Murakami (1989)]; triangular distribution in Csordas et al. [Csordas et al. (2003)]; the sinusoidal distribution, the convolution of triangular and uniform distributions and the convolution of triangular and triangular distributions in Widrow and Kollar [Widrow, Kollar (2008)]. The derived formulas for the general moment can be used to provide basic measures of roundoff error for each of these distributions.

The contents of this chapter are organized as follows: four theorems deriving expressions for general moments of $X - \lfloor X \rfloor$ and $X - \left\lfloor X + \frac{1}{2} \right\rfloor$ are given in Section 5.2;

[1] [O] Embedded sentence omitted.

specific forms of the general moments for some special distributions are derived in Section 5.3; a numerical study showing the use of the theorems and checking correctness of their derivations is given in Section 5.4.

5.2 Theoretical Considerations

Theorems 5.2.2 and 5.2.3 derive the general moments of $X - \lfloor X \rfloor$ and $X - \lfloor X + \frac{1}{2} \rfloor$ when X is a random variable on the real line. Theorems 5.2.4 and 5.2.5 derive the general moments of $X - \lfloor X \rfloor$ and $X - \lfloor X + \frac{1}{2} \rfloor$ when X is a bounded random variable. The proofs of the theorems are given in Appendix C.

We adopt the notation below: for a random variable X with probability density function f (PDF) and cumulative distribution function F (CDF),

$$M_t(x) = \int_{-\infty}^{x} z^t f(z) dz = x^t F(x) - t \int_{-\infty}^{x} z^{t-1} F(z) dz.$$

For CDFs F with infinite non-zero domains, the following statement

$$z^t F(z) \xrightarrow{z \to -\infty} 0,$$

needs to hold. This applies specifically to the functions described in Theorems 5.2.2 and 5.2.3, and needs to be verified in these cases. From this we gather two more properties for $M_t(x)$ in the following proposition:

Proposition 5.2.1. *For the CDF F with the bounded support $[a, b]$, $z^t F(z) \xrightarrow{z \to -\infty} 0$. If $x \leq a$ then*

$$M_t(x) = 0.$$

Likewise for $b \leq x$,

$$M_t(x) = M_t(b).$$

This lets us derive the general moments for the round off error in Theorems 5.2.2 to 5.2.5.

Theorem 5.2.2. *Let X be a continuous random variable on the domain $(-\infty, \infty)$ with PDF f and CDF F. Then for $k \in \mathbb{N}$,*

$$\mathbb{E}\left[(X - \lfloor X \rfloor)^k\right] = \sum_{i=0}^{k} (-1)^i \binom{k}{i} \sum_{j=-\infty}^{\infty} j^i \left[M_{k-i}(j+1) - M_{k-i}(j)\right].$$

Theorem 5.2.3. *Let X be a continuous random variable on the domain $(-\infty, \infty)$ with PDF f and CDF F. Then for $k \in \mathbb{N}$,*

$$\mathbb{E}\left[\left(X - \left\lfloor X + \frac{1}{2}\right\rfloor\right)^k\right] = \sum_{i=0}^{k}(-1)^i\binom{k}{i}\sum_{j=-\infty}^{\infty} j^i\left[M_{k-i}\left(j+\frac{1}{2}\right) - M_{k-i}\left(j-\frac{1}{2}\right)\right].$$

Theorem 5.2.4. *Let X be a continuous random variable on the domain (a, b) with $-\infty < a < b < \infty$ with PDF f and CDF F that satisfy the conditions of Proposition 5.2.1. Then for $k \in \mathbb{N}$,*

$$\mathbb{E}\left[(X - \lfloor X\rfloor)^k\right] = \sum_{i=0}^{k}(-1)^i\binom{k}{i}\left[\sum_{j=p}^{q-1} j^i\left(M_{k-i}(j+1) - M_{k-i}(j)\right)\right],$$

with $p = \lfloor a\rfloor$ and $q = \lceil b\rceil$.

Theorem 5.2.5. *Let X be a continuous random variable on the domain (a, b) with $-\infty < a < b < \infty$ with PDF f and CDF F that satisfy the conditions of Proposition 5.2.1. Then for $k \in \mathbb{N}$,*

$$\mathbb{E}\left[\left(X - \left\lfloor X + \frac{1}{2}\right\rfloor\right)^k\right] = \sum_{i=0}^{k}(-1)^i\binom{k}{i}\sum_{j=p}^{q-1} j^i\left[M_{k-i}\left(j+\frac{1}{2}\right) - M_{k-i}\left(j-\frac{1}{2}\right)\right],$$

with $p = \left\lfloor a + \frac{1}{2}\right\rfloor$ and $q = \left\lceil b + \frac{1}{2}\right\rceil$.

5.3 Commonly Used Distributions in Practice

We address the explicit forms of the previously derived general moment formulas for common distributions. [2] We use the notation $k_1 = k - i + 1$, $k_2 = k - i + 2$ and $c(k, j) = (j+1)^k - j^k$ throughout the following paragraphs.

These newly derived general formulations now rest solely on the computation of the functional $M_t(x)$, which in turn largely depend on the underlying distribution function. We seek to simplify this further with the following remark.

[2] [A] The examples of uniform, triangular and normal distributions are provided in this section, for which we have computed error tables and provided numerical comparison plots. Further commonly used distributions can be found in Appendix C.

Remark 5.3.1. *With the notation being the same as in Theorems 5.2.2 through 5.2.5 and $x < \infty$ we can express the functional $M_t(x)$ under the assumption $x^{k-l}F^{(-l)}(x) \to 0$, for $x \to -\infty$ for $l = 0, \ldots, n$ as following:*

$$M_t(x) = \sum_{k=0}^{t} (-1)^k x^{t-k} \frac{t!}{(t-k)!} F^{(-k)}(x),$$

with $F^{(-k)}$ marking the k-th anti-derivative.

Proof.

$$\begin{aligned} M_t(x) &= \int_{-\infty}^{x} z^t f(z) dz \\ &\overset{\text{def.}}{=} x^t F(x) - t \left[x^{t-1} F^{(-1)}(x) - (t-1) \int_{-\infty}^{x} z^{t-2} F^{(-1)}(z) dz \right] \\ &\overset{\text{i. b. p.}}{=} x^t F(x) - t x^{t-1} F^{(-1)}(x) + t(t-1) \left[x^{t-2} F^{(-2)}(x) - \int_{-\infty}^{x} z^{t-3} F^{(-2)}(z) dz \right] \\ &\vdots \\ &= \sum_{k=0}^{t} (-1)^k x^{t-k} \frac{t!}{(t-k)!} F^{(-k)}(x). \end{aligned}$$

Thus performing integration by parts successively for k times, we are left we the formulation of remark 5.3.1. [3] □

Here, the term k-th anti-derivative emulates the nomenclature of Liu et al. [Liu et al. (2008)], and refers to the function $F^{(-k)}(x)$ such that $\frac{\partial F^{(-k)}}{\partial x^k} = F(x)$

With the previous remark we have reduced the computation of the general moments, to computing the k-th anti-derivative of the underlying random variables cumulative distribution function. In the next section we give a list of the necessary functional $M_t(x)$ for an array of commonly used distributions.

5.3.1 Uniform Distribution (Widrow and Kollar [Widrow, Kollar (2008)], I.7, page 679)

For a uniform random variable X with PDF and CDF specified by:

[3] [C] The notation of $M(x) = \int_{-\infty}^{x} z f(z) dz$ has been introduced in [Li, Nadarajah (2016)], but the n-dimensional extension necessitates an efficient expansion.

$$f_X(x) = \frac{1}{b-a}\mathbb{I}_{\{x\in[a,b]\}}, \qquad F_X(x) = \begin{cases} 0 & \text{if } x < a, \\ \frac{x-a}{b-a} & \text{if } x \in [a,b], \\ 1 & \text{if } x > b. \end{cases}$$

With $-\infty < a < b < \infty$ it follows that:

$$M_k(x) = F^{(-k)}(x) = \begin{cases} 0 & \text{if } x < a, \\ \frac{1}{b-a}\left(\frac{x^{k+1}}{(k+1)!} - \frac{a^{k+1}}{(k+1)!}\right) & \text{if } x \in [a,b], \\ \frac{1}{b-a}\left(\frac{b^{k+1}}{(k+1)!} - \frac{a^{k+1}}{(k+1)!}\right) & \text{if } x > b. \end{cases}$$

5.3.2 Triangular Distribution (Widrow and Kollar [Widrow, Kollar (2008)], I.8, page 680)

For a triangular (Widrow and Kollar [Widrow, Kollar (2008)], I.8, page 680) random variable X with PDF and CDF specified by:

$$f_X(x) = \begin{cases} 0 & \text{if } x \le a, \\ \frac{2(x-a)}{(b-a)(c-a)} & \text{if } a < x \le c, \\ \frac{2(b-x)}{(b-a)(b-c)} & \text{if } c < x \le b, \\ 0 & \text{if } x > b \end{cases} \qquad F_X(x) = \begin{cases} 0 & \text{if } x < a, \\ \frac{(x-a)^2}{(b-a)(c-a)} & \text{if } x \in [a,b], \\ 1 - \frac{(b-x)^2}{(b-a)(b-c)} & \text{if } x > b, \\ 1 & \text{if } x \ge b, \end{cases}$$

for $-\infty < a \le c \le b < \infty$. For $k \in \mathbb{N}$ the functional results as follows:

$$M_k(x) = F^{(-k)}(x) = \begin{cases} 0 & \text{if } x < a, \\ \frac{1}{(b-a)(c-a)}\left(x^{k+1}\left(\frac{x}{(k+2)!} - \frac{a}{(k+1)!}\right)\right. & \\ \left. -a^{k+2}\left(\frac{1}{(k+2)!} - \frac{1}{(k+1)!}\right)\right) & \text{if } a \le x < c, \\ \frac{1}{(b-a)(c-a)}\left(c^{k+1}\left(\frac{c}{(k+2)!} - \frac{a}{(k+1)!}\right)\right. & \\ \left. -a^{k+2}\left(\frac{1}{(k+2)!} - \frac{1}{(k+1)!}\right)\right) & \\ +\frac{1}{(b-a)(b-c)}\left(x^{k+1}\left(\frac{b}{(k+1)!} - \frac{x}{(k+2)!}\right)\right. & \text{if } c \le x < b, \\ \left. -c^{k+2}\left(\frac{1}{(k+1)!} - \frac{1}{(k+2)!}\right)\right) & \\ \frac{1}{(b-a)(c-a)}\left(c^{k+1}\left(\frac{c}{(k+2)!} - \frac{a}{(k+1)!}\right)\right. & \\ \left. -a^{k+2}\left(\frac{1}{(k+2)!} - \frac{1}{(k+1)!}\right)\right) & \\ +\frac{1}{(b-a)(b-c)}\left(b^{k+2}\left(\frac{1}{(k+1)!} - \frac{1}{(k+2)!}\right)\right. & \text{if } x \ge b. \\ \left. -c^{k+2}\left(\frac{1}{(k+1)!} - \frac{1}{(k+2)!}\right)\right) & \end{cases}$$

5.3.3 Normal Distribution ([Widrow, Kollar (2008)], I.1, page 633)

For a centralised ($\mathbb{E}(X) = 0$)) normal random variable X with PDF and CDF specified by

$$f_X(x) = \frac{1}{\sqrt{2\pi}a} \exp\left(-\frac{x^2}{2a^2}\right), \qquad F_X(x) = \Phi\left(\frac{x}{a}\right),$$

for $-\infty < x < \infty$ and $a > 0$, $\Phi(x)$ denotes the Laplace function.

The normal distribution does not possess a closed- form antiderivative. Yet numerical integration can determine $M_t(x)$ for arbitrary values for both Theorems 5.2.2 and 5.2.3.

All the results for the example distributions can be finally derived by inserting the respective functions into the Theorems 5.2.2 - 5.2.5. Through the formulation of $M_t(x)$ as a generalised antiderivative, the computation of moments has been reduced to integral evaluation. Here the cases with finite domain are treated by Theorem 5.2.4 for the roundoff function, and by Theorem 5.2.5 for the rounding function to the nearest integer. Similarly the distributions with infinite domain in form of the normal

distribution utilise formulation 5.2.2 and 5.2.3, respectively. The evaluations of the different integral functions then lead to the formulations as stated in Sections 5.3.1 - 5.3.3.

Whenever a closed-form for the antiderivatives does not exist, numerical measures can be considered. Depending on the power t in the functional, the incline of the integrand may be too steep for standard integration methods such as trapezoid methods to work, and quadrature methods (Gaussian) or Simpson's rule may be preferable. [4] Since the underlying density functions are usally 'well behaved', in the sense that they do not exhibit oscillatory behaviour or troublesome singularities, which can often cause problems when evaluating integrals numerically . We have touched upon this with the normal and sinusoidal distribution.

5.4 Numerical Results

5.4.1 Scatterplots

In this section, we present the results of our numerical computations. The plots show the first four moments, computed by the previously derived formulas and via the numerical moments, given by the Monte Carlo estimator $\sum_{i=1}^{n}(X_i - \lfloor X_i \rfloor)^k$, $k = 1, \ldots, 4$ and $\sum_{i=1}^{n}(X_i - \lfloor X_i + \frac{1}{2} \rfloor)^k$ for a randomly generated sample of size $n = 10000$. [5] See Figures 5.1 - C.4. [6]

For the sample distributions we have decided on centred versions of the uniform, triangular and normal distributions. Therefore, we have X a uniform random variable with $b = -a$, or X a triangular random variable with $b = -a$, $c = 0$ or X a normal random variable with zero mean and standard deviation a. We let the parameter a increase by increments of 0.1 from 1 to 10 ($a = 0.1, 0.2, \ldots, 10$), leaving us with 100 plotted points for each moment. The tight grouping and overall shape of the plots gives us reason to believe in the correctness of our previous results.

The experiment was repeated for the shifted error moments of $X - \left\lfloor X + \frac{1}{2} \right\rfloor$, with the same distributions and sample size. For all experiments we see similar structures in the figures. The first moment experiment exhibits a horizontal line, as the theoretical

[4] [A] Specific recommendations were added after questions regarding this were raised during annual review.

[5] [A] The estimator had not been explicitly named as Monte Carlo Estimator, which has been amended for clarity.

[6] [A] Figures for both errors of the normal distribution are depicted below, similar FIgures for uniform and triangular example distributions can be found in the appendix C.

values are always either 0 or $\frac{1}{2}$ depending on the variable, but remains the same for symmetric, centred distributions regardless of variance. Higher moments change according to the altered parameters, but as the moment stagnates for sufficiently high variance, the randomness of the samples draw a horizontal line once again. [7]

Plots for the Moments of $X - \lfloor X \rfloor$

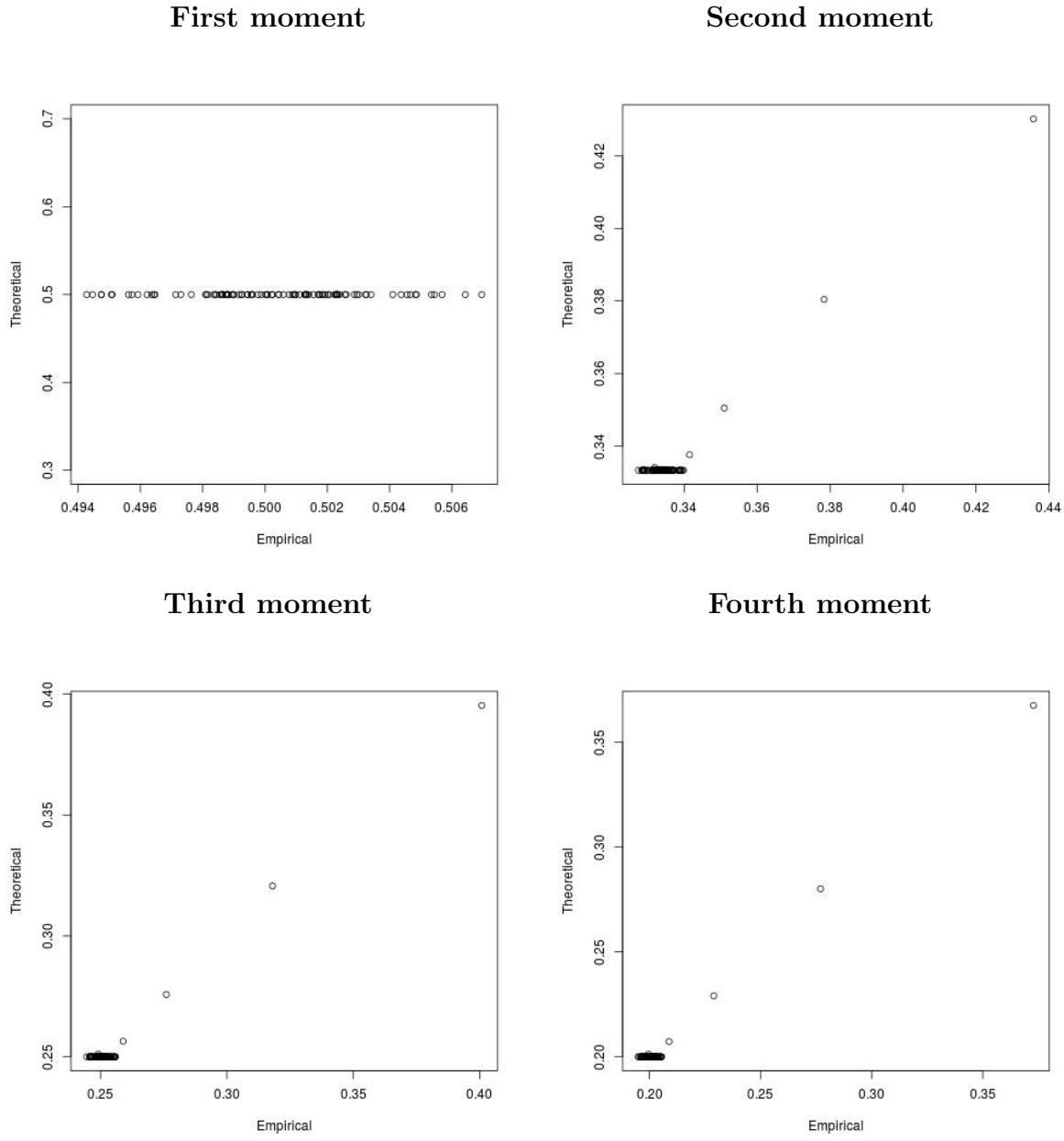


Figure 5.1: First four moments of $X - \lfloor X \rfloor$ for the normal distribution, with parameters $\mu = 0$ and standard deviation $a = 0.1, 0.2, \ldots, 10$.

[7] [A] The last paragraph has been expanded after questions were brought up regarding the structure of the point cloud.

Plots for the Moments of $X - \left\lfloor X + \frac{1}{2} \right\rfloor$

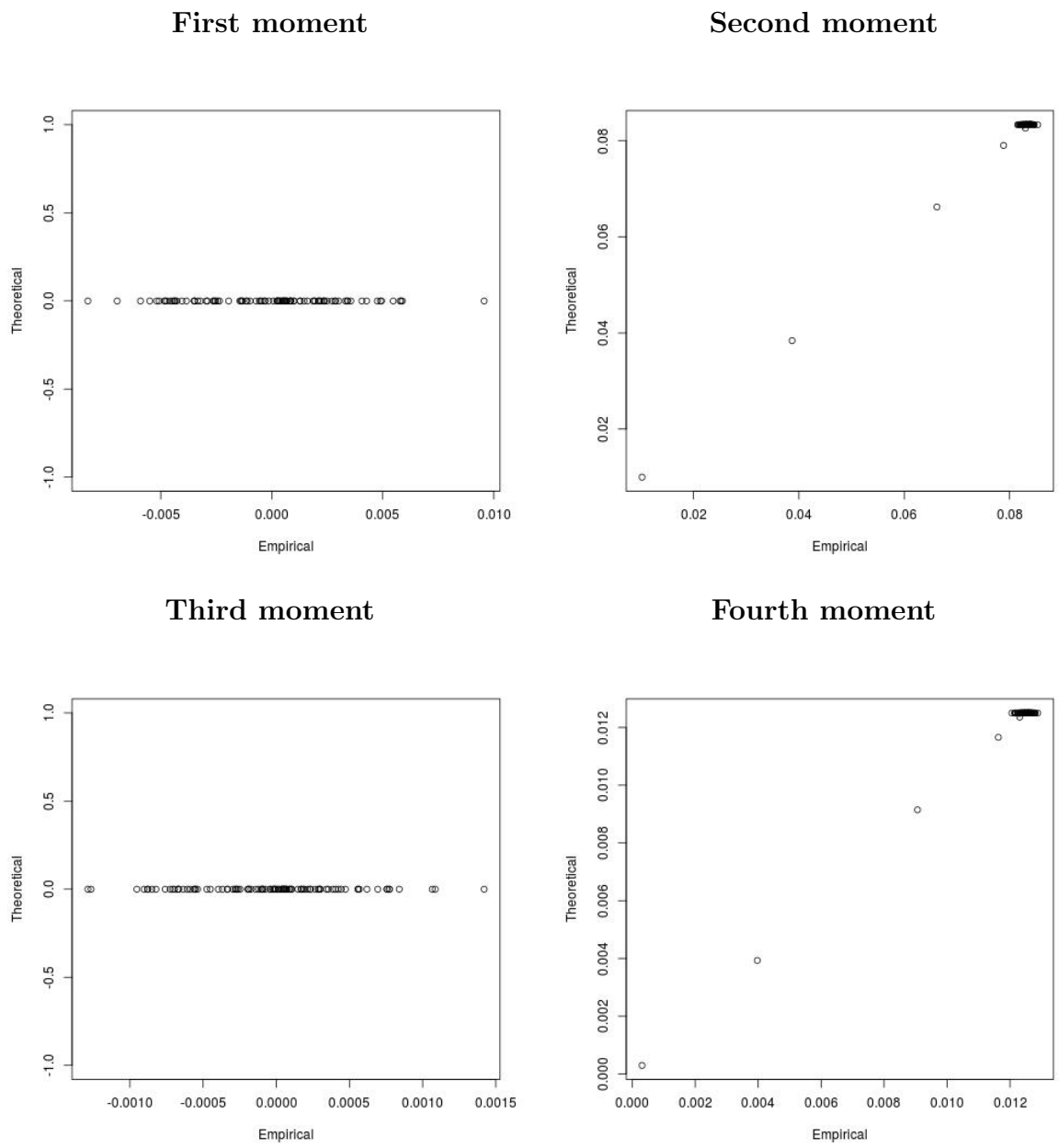


Figure 5.2: First four moments of $X - \left\lfloor X + \frac{1}{2} \right\rfloor$ for the normal distribution, with parameters $\mu = 0$ and standard deviation $a = 0.1, 0.2, \ldots, 10$.

5.4.2 Error Tables

In addition to the visualised results of Section 5.4.1, we provide tables of the average squared error between numerical and theoretical values of the k-th moment for $k = 1, 2, 3, 4$, See Tables 5.1 to C.4. [8] The distributions remain the same while the deviation has been computed for selected parameter values of a.

Average Squared Errors for the Moments of $X - \lfloor X \rfloor$

a	k	1	2	3	4
0.1		2.364196e-05	3.090310e-05	2.984747e-05	2.722596e-05
1		9.033631e-06	1.318285e-06	5.234990e-08	1.142558e-06
2		1.884854e-06	7.064957e-07	2.127054e-07	6.517809e-08
3		8.582170e-07	9.349346e-08	8.328996e-08	4.806649e-07
4		2.728113e-06	4.661411e-06	5.100822e-06	4.918637e-06
5		1.683766e-06	9.190482e-07	6.206288e-07	5.312952e-07
6		3.359889e-06	1.850607e-06	4.870644e-07	2.729104e-08
7		1.595674e-06	8.675618e-07	4.078100e-07	1.846421e-07
8		1.468702e-06	2.132272e-06	2.906684e-06	3.418431e-06

Table 5.1: ASE of the empirical and theoretical values for the normal distribution.

Average Squared Errors for the Moments of $X - \left\lfloor X + \frac{1}{2} \right\rfloor$

a	k	1	2	3	4
0.1		1.897385e-07	5.128911e-08	1.438813e-09	1.504597e-10
1		4.848491e-05	3.929998e-07	5.765655e-07	4.875014e-09
2		6.846606e-06	3.016127e-10	1.352258e-07	1.778393e-09
3		9.816704e-07	2.411716e-08	2.606065e-10	3.236768e-09
4		3.340776e-05	6.215697e-08	1.633690e-07	2.822030e-10
5		2.103549e-05	8.081596e-07	6.759402e-07	2.861358e-08
6		2.237473e-05	4.477350e-09	9.073246e-07	4.855412e-12
7		7.358063e-06	2.029926e-07	8.169570e-08	7.196841e-09
8		2.259615e-05	1.060460e-07	3.841394e-07	1.398618e-09

Table 5.2: ASE of the empirical and theoretical values for the normal distribution.

[8] [A] Tables for the normal distribution are depicted below, similar tables for uniform and triangular example distributions can be found in Appendix C.

Overall the difference between numerical and theoretical approaches is fairly close to zero, for both regular rounded error and shifted error moments. This puts forth further evidence for the feasibility of the theoretical results in Sections 5.2 and 5.3.

5.5 Conclusions

We have derived explicit expressions for arbitrary order moments of regular rounded errors and shifted rounded errors. The general expressions have been specialised to ten distributions arising in the signal processing area (references to applications of these distributions in signal processing are given): uniform distribution, triangular distribution, trapezoidal distribution, house distribution, curved trapezoidal distribution, hexagonal distribution, sinusoidal distribution, convolutions of uniform and triangular distributions, convolutions of two triangular distributions and the normal distribution. The specialised expressions are all simple except for the normal and sinusoidal distribution. [9]

We have checked the correctness of the derivations through two numerical studies: i) by plotting values of the derived expressions versus those obtained by simulation; ii) by tabulating values of the derived expressions versus those obtained by simulation. Both studies show that the derived expressions are accurate.

A future work is to extend this work to other forms of rounding and also to consider rounding of more than one variable of interest.

[9] [A] Both normal and sinusoidal distribution have non-closed form anti-derivatives.

Chapter 6

Approximation Methods for Lognormal Characteristic Functions

Chapter Abstract

The characteristic function of the lognormal distribution is of interest in a number of scientific fields yet an analytic solution remains elusive, making reliable and efficient approximations necessary. In this article, we build on the results of N. C. Beaulieu and A. Saberali in "New approximations to the lognormal characteristic function' ', by introducing a Taylor- and Bessel function-based partial expansion of the integrand and a Chebyshev quadrature approach. Through computer simulations we show that the Taylor expansion remains accurate and efficient for all commonly computed values, and specify the range of values for which the other two approaches show a significantly stronger performance.

6.1 Introduction

The lognormal distribution features in a number of application fields in science, such as biology, medicine, communications or finance [Limpert et al. (2001)]. For example the distribution among other heavy-tailed distributions is used to model the severity of impact of a loss incident in operational risk management, or the income distribution for populations [Clementi, Gallegati (2005)].

The characteristic function can be used for a multitude of statistic properties of a given distribution. Moments, if finite, may be directly computed via:

$$\mathbb{E}[X^n] = i^{-n} \left[\frac{\partial^n}{\partial t^n} \varphi_X(t) \right]_{t=0}, \tag{6.1}$$

where $n \in \mathbb{N}$ and φ is the respective characteristic function of the RV X. Other direct applications of the CF include the computation of probabilities. Often this can be achieved by applying the inverse Fourier transform on the characteristic function of the distribution, which reduces the problem to properties of the normal distribution, which are well understood.

A more general application lies in the fact that the logarithm of a product is the sum of the logarithms of the factors, as the sum of i.i.d. random variables tends towards a normal distribution the product of i.i.d. random variables will converge to a lognormal distribution as a direct result of the Central Limit Theorem (see also [Asmussen et al. (2016)]).

Let Z be a random variable of lognormal distribution, with $z > 0, -\infty < \mu < \infty$ and $\sigma > 0$. Then the probability density function is defined in 6.2:

$$f_Z(z) = \frac{1}{\sqrt{2\pi}\sigma z} \exp\left[-\frac{(\log(z) - \mu)^2}{2\sigma^2} \right]. \tag{6.2}$$

We write $Z \sim LN(\mu, \sigma^2)$, with μ and σ^2 being the mean and variance, respectively.

The importance of the characteristic function (CF) is founded in its determination of behavior of the probability function of the underlying random variable. The CF of Z is commonly stated as

$$\Phi_Z(\omega) = \int_0^\infty \frac{1}{\sqrt{2\pi}\sigma z} \exp\left[-\frac{(\log(z) - \mu)^2}{2\sigma^2} \right] e^{i\omega z} dz, \tag{6.3}$$

where $i = \sqrt{-1}$. Currently, no closed-form of this integral is known, creating the necessity of accurate and fast approximation. Depending on the variance the integrand can be highly oscillatory with a heavy tail. This may make the computation by standard numerical approaches time consuming and unstable.

A number of series representations have been put forth over time, often originating out of the field of communications. Especially interesting for wireless communication systems, the distribution has been employed to model large-scale signal fading, the interference of mobile and other wireless networks [Tellambura, Senaratne (2010)] or shadowing effects in radio transmissions [Beaulieu (2010)]. Earlier approximations and their applications include Bakarat's approach which aimed to model optical propagation through the atmosphere [Bakarat (1976)]. Depending on their intended applications, these approximations focus on a special range of parameter values. In communications the ratio of power is expressed in decibels (dB), the range of interest is therefore $\sigma_y = 10/\log 10\sigma$ [Heliot et al. (2009)] from 10^{-2}dB to 13dB ($0.0023 < \sigma < 3$ (as seen in [Yeh, Schwartz (1984)], [Beaulieu (2012)]) in which the methods are required to operate reliably.

This chapter is structured into four parts. In Section 6.2 we begin by introducing a partial Taylor expansion into the integrand and determine its analytic solution for finite intervals of appropriate length. Secondly, we propose another approach based on the series expansion of the exponential function utilising modified Bessel functions. We follow this up with a simulation study of different techniques, based on both the central limit theorem and series expansions. We will compare computational efficiency as well as absolute and relative error measures.

In Section 6.3 we cover quadrature methods, such as the Chebyshev type quadrature, which we introduce to a integral formulation of the CF, as proposed by Gubner [Gubner (2006)]. Once again we run simulations to compare efficiency and stability of the different approaches against one another. We close with a terse statement about the performances and suitable parameter regions of all investigated methods 6.4.

6.2 Expansion Approaches for the Characteristic Function

6.2.1 Partial Taylor Expansions on Finite Intervals

We reformulate by splitting up the CF into real and imaginary parts for convenient handling. We will only examine the computation of the real part for now, as the

imaginary part can be evaluated analogously, by substituting z by $x = \omega z$, so that the second factor function is not dependent on any other variables. Also as previously stated throughout literature [Beaulieu, Saberali (2012)], we can assume $\mu = 0$ without loss of generality.

Note that the following holds: [1]

$$\begin{aligned}
\Phi_Z(\omega) &= \int_0^\infty \frac{1}{\sqrt{2\pi}\sigma z} \exp\left[-\frac{\log(z)^2}{2\sigma^2}\right] [\cos(\omega z) + i\sin(\omega z)]\, dz \\
&= \frac{1}{\sqrt{2\pi}\sigma} \int_0^\infty \frac{\omega}{x} \exp\left[-\frac{(\log(x) - \log(w))^2}{2\sigma^2}\right] \cos(x)dx + iIm\left(\Phi_Z(\omega)\right) \\
&= \frac{1}{\sqrt{2\pi}\sigma} \int_0^\infty \frac{\omega}{x} \exp\left[-\frac{\log(x)^2 - 2\log(w)\log(x) + \log(w)^2}{2\sigma^2}\right] \cos(x)dx \\
&\quad + iIm\left(\Phi_Z(\omega)\right) \\
&= \frac{\exp\left[-\frac{\log(\omega)^2}{2\sigma^2}\right]}{\sqrt{2\pi}\sigma} \int_0^\infty \frac{\omega}{x} \exp\left[-\frac{\log(x)^2}{2\sigma^2}\right] \exp\left[\frac{1}{\sigma^2}\log(\omega)\log(x)\right] \cos(x)dx \\
&\quad + iIm\left(\Phi_Z(\omega)\right) \\
&= \underbrace{\frac{\omega \exp\left[-\frac{\log(\omega)^2}{2\sigma^2}\right]}{\sqrt{2\pi}\sigma}}_{c_{(\omega,\sigma)}} \int_0^\infty \exp\left[-\frac{\log(x)^2}{2\sigma^2}\right] x^{\frac{1}{\sigma^2}\log(\omega)-1} \cos(x)dx + iIm\left(\Phi_Z(\omega)\right).
\end{aligned} \tag{6.4}$$

The integral above still has no analytic solution, which means that from this point on we will work on approximate representations of the CF, rather than exact formulations. We split the domain of integration into an infinite number of sections $[a_k, a_{k+1}]$ with $k = 0, \ldots, \infty$ and $a_k = kt$ where t is the interval width:

$$\Phi_Z(\omega) = c_{(\omega,\sigma)} \sum_{k=0}^{\infty} \int_{a_k}^{a_{k+1}} \exp\left[-\frac{\log(x)^2}{2\sigma^2}\right] x^{\frac{1}{\sigma^2}\log(\omega)-1} \cos(x)dx.$$

In each interval, we now introduce a straightforward Taylor series representation of the cosine function (note that the j-th derivative of the cosine function is $\cos(x)^{(j)} = \cos(x + j\frac{\pi}{2})$). We do this in order to gain sum terms for which the integral has a closed-form solution. Given the shape of the integrand, which is oscillating around zero with a dampening amplitude as x grows larger, the interval size of each individual integral section needs to be chosen appropriately with respect to the stability of the section's integration (the larger the intervals, the harder it is to find an

[1] [A] There were some mistakes in writing in the published version, such as the power of 2 in line three extended to the entire numerator, and $\frac{1}{z}$ not being substituted until the last line.

accurate, stable approximation).

There has to be a balance between interval size and the order of the Taylor expansion. The series expansion is centred around the midpoint of the corresponding interval $\overline{a}_k = \frac{a_{k+1}-a_k}{2}$. Hence, we obtain the following series representation:

$$\Phi_Z(\omega) = c_{(\omega,\sigma)} \sum_{k=0}^{\infty} \int_{a_k}^{a_{k+1}} \exp\left[-\frac{\log(x)^2}{2\sigma^2}\right] x^{\frac{1}{\sigma^2}\log(\omega)-1} \sum_{j=0}^{\infty} \frac{\cos\left(\overline{a}_k + j\frac{\pi}{2}\right)}{j!} (x-\overline{a}_k)^j \, dx.$$

Since the exponential term will eventually outweigh the polynomials or the cosine function and cause the integrand to be zero, we can take a sufficiently large finite number of intervals m as an approximation. We may also interrupt the Taylor series after n steps, leaving us with a finite double sum. We then rearrange the terms in the following:

$$\begin{aligned}
\Phi_Z(\omega) &\approx c_{(\omega,\sigma)} \sum_{k=0}^{m} \int_{a_k}^{a_{k+1}} \exp\left[-\frac{\log(x)^2}{2\sigma^2}\right] x^{\frac{1}{\sigma^2}\log(\omega)-1} \sum_{j=0}^{n} \frac{\cos\left(\overline{a}_k + j\frac{\pi}{2}\right)}{j!} (x-\overline{a}_k)^j \, dx \\
&= c_{(\omega,\sigma)} \sum_{k=0}^{m} \sum_{j=0}^{n} \frac{\cos\left(\overline{a}_k + j\frac{\pi}{2}\right)}{j!} \int_{a_k}^{a_{k+1}} \exp\left[-\frac{\log(x)^2}{2\sigma^2}\right] x^{\frac{1}{\sigma^2}\log(\omega)-1} (x-\overline{a}_k)^j \, dx \\
&= c_{(\omega,\sigma)} \sum_{k=0}^{m} \sum_{j=0}^{n} \frac{\cos\left(\overline{a}_k + j\frac{\pi}{2}\right)}{j!} \\
&\quad \times \int_{a_k}^{a_{k+1}} \exp\left[-\frac{\log(x)^2}{2\sigma^2}\right] x^{\frac{1}{\sigma^2}\log(\omega)-1} \sum_{h=0}^{j} \binom{j}{h} x^h \left(-\overline{a}_k\right)^{j-h} dx \\
&= c_{(\omega,\sigma)} \sum_{k=0}^{m} \sum_{j=0}^{n} \sum_{h=0}^{j} \frac{\cos\left(\overline{a}_k + j\frac{\pi}{2}\right)}{j!} \binom{j}{h} \left(-\overline{a}_k\right)^{j-h} \\
&\quad \times \int_{a_k}^{a_{k+1}} \exp\left[-\frac{\log(x)^2}{2\sigma^2}\right] x^{\frac{1}{\sigma^2}\log(\omega)-1+h} dx. \qquad (6.5)
\end{aligned}$$

The transformations in 6.5 are essentially a rearrangement of sum terms. We expand the monomials of the Taylor expansion by means of the binomial theorem. We are left with a simplified integrand, for which we can find a closed-form solution. The remaining composite function of exponential and polynomial factors can hence be expressed as follows:

$$\int \exp\left[-\beta \log(x)^2\right] x^{\alpha} dx = -\frac{\sqrt{\pi}}{2\sqrt{\beta}} e^{\frac{(\alpha+1)^2}{4\beta}} \mathrm{Erf}\left[\frac{\alpha - 2\beta\log(x) + 1}{2\sqrt{\beta}}\right],$$

where $\alpha = \frac{\log(\omega)}{\sigma^2} - 1 + h$, $\beta = \frac{1}{2\sigma^2}$ and $\mathrm{Erf}(\cdot)$ denotes the error function. We therefore conclude:

$$
\begin{aligned}
I(\omega,\sigma,h,k) &:= \int_{a_k}^{a_{k+1}} \exp\left[-\frac{\log(x)^2}{2\sigma^2}\right] x^{\frac{1}{\sigma^2}\log(\omega)-1+h} dx \\
&= \exp\left[\frac{h^2\sigma^2+\log(\omega)^2}{2\sigma^2}\right]\sqrt{\frac{\pi}{2}}\sigma w^h \\
&\quad\times\left[\mathrm{Erf}\left(\frac{h\sigma^2-\log a_k+\log\omega}{\sqrt{2}\sigma}\right)-\mathrm{Erf}\left(\frac{h\sigma^2-\log a_{k+1}+\log\omega}{\sqrt{2}\sigma}\right)\right].
\end{aligned}
\tag{6.6}
$$

Expression 6.6 can then be efficiently evaluated using existing numerical methods in Mathematica or similar software. [2]

Hence, we have an approximation dependent on the three parameters n, m and t, being the number of intervals, order of Taylor expansion and interval width, respectively: [3]

$$
\Phi_Z(\omega) \approx \widehat{\Phi}_Z(\omega) = c_{(\omega,\sigma)} \sum_{k=0}^{m}\sum_{j=0}^{n}\sum_{h=0}^{j} \frac{\cos\left(\overline{a}_k + j\frac{\pi}{2}\right)}{j!}\binom{j}{h}(-\overline{a}_k)^{j-h} I(\omega,\sigma,h,k). \tag{6.7}
$$

To compute the imaginary part of the CF the cosine function in 6.7 may simply be replaced by a sine function.

6.2.2 Bessel Function Series Expansion

We begin with the expression in Equation 6.2, and introduce a series expansion for the exponential function in 6.7 (9.6.37) [Abramowitz, Stegun (1964)]:

$$
e^z = I_0(z) + 2\sum_{k=1}^{\infty} I_k(z), \tag{6.8}
$$

where $I_k(z)$ denotes the modified I-Bessel function of first kind of order k.

The series representation can easily be altered to replace the trigonometric functions as well, the result of which can be seen in 6.9 (see 9.1.47 [Abramowitz, Stegun

[2] [C] The error functions are common throughout statistics, and numerous efficient evaluation methods have been proposed.

[3] [C] The approximation works best if intervals are chosen under consideration of the variance. Generally speaking interval width or sum terms and interpolation orders can to some degree be balanced out against each other, e.g. smaller intervals allow for lower order interpolation and vice versa. Our experience was that efficient values of t ranged between $0.5 - 2.5$ and the inner term number $m > 30$ should be significantly higher than $n \geq 4$.

(1964)]):

$$\cos(z) = J_0(z) + 2\sum_{k=1}^{\infty}(-1)^k J_{2k}(z). \tag{6.9}$$

where $J_{2k}(z)$ denotes the J-Bessel function of first kind of order $2k$. We substitute 6.9 into 6.4 and obtain a new series representation in 6.10: [4]

$$\begin{aligned}
\Phi_Z^{re}(\omega) &= c_{(\omega,\sigma)} \int_0^{\infty} \exp\left[-\frac{\log(x)^2}{2\sigma^2}\right] x^{\frac{1}{\sigma^2}\log(\omega)-1} \left[J_0(x) + 2\sum_{k=1}^{\infty}(-1)^k J_{2k}(x)\right] dx \\
&= c_{(\omega,\sigma)} \Bigg\{ \int_0^{\infty} \exp\left[-\frac{\log(x)^2}{2\sigma^2}\right] x^{\frac{1}{\sigma^2}\log(\omega)-1} J_0(x) dx \\
&\qquad + 2\int_0^{\infty} \exp\left[-\frac{\log(x)^2}{2\sigma^2}\right] x^{\frac{1}{\sigma^2}\log(\omega)-1} \sum_{k=1}^{\infty}(-1)^k J_{2k}(x) dx \Bigg\} \\
&= c_{(\omega,\sigma)} \Bigg\{ \int_0^{\infty} \exp\left[-\frac{\log(x)^2}{2\sigma^2}\right] x^{\frac{1}{\sigma^2}\log(\omega)-1} \sum_{m=0}^{\infty} \frac{(-1)^m}{(m!)^2} \left(\frac{x}{2}\right)^{2m} dx \\
&\qquad + 2\int_0^{\infty} \exp\left[-\frac{\log(x)^2}{2\sigma^2}\right] x^{\frac{1}{\sigma^2}\log(\omega)-1} \\
&\qquad \times \sum_{k=1}^{\infty}(-1)^k \sum_{m=0}^{\infty} \frac{(-1)^m}{m!(m+2k)!} \left(\frac{x}{2}\right)^{2m+2k} dx \Bigg\} \\
&\approx c_{(\omega,\sigma)} \Bigg\{ \sum_{m=0}^{r} \frac{(-1)^m}{(m!)^2} \left(\frac{1}{2}\right)^{2m} \int_0^{c} \exp\left[-\frac{\log(x)^2}{2\sigma^2}\right] x^{\frac{1}{\sigma^2}\log(\omega)-1+2m} dx \\
&\qquad + 2\sum_{k=1}^{b}\sum_{m=0}^{r} \frac{(-1)^{(m+k)}}{m!(m+2k)!} \left(\frac{1}{2}\right)^{2m+2k} \\
&\qquad \times \int_0^{c} \exp\left[-\frac{\log(x)^2}{2\sigma^2}\right] x^{\frac{1}{\sigma^2}\log(\omega)-1+2(m+k)} dx \Bigg\}.
\end{aligned} \tag{6.10}$$

By discontinuing the series representation after a finite amount of sum terms we are once again evaluating an approximation instead of an exact expression. The cosine expansion as well as the Bessel function expansion are convergent, therefore the number of sum terms can be chosen to balance out accuracy and computation speed, in accordance with the required approximation properties. The parameters b, r, c denote the number of cosine series terms, exponential expansion terms (containing the J-Bessel function) and the upper integration limit, respectively. We have empirically

[4] [A] The equations in 6.10 erroneously used z instead of x in parts in the published version. We have replaced the variable notation where necessary.

found the appropriate combination of parameters to be between 15 and 30 sum terms per interval.

The Bessel expansion has proven to be stable enough for the integration interval not to be segmented. However, to ensure the desired upper integration limit c can be reached, a sufficiently large amount of series terms are evaluated. If c is chosen too large (or b, r too small) we may be integrating a strongly deviating approximate integrand.

The integral part which requires numerical evaluation is expressed in 6.11:

$$\mathrm{Int}(h,\sigma,c) := \int_0^c e^{\frac{-\log(z)^2}{2\sigma^2}} x^h dz = \exp\left[(h+1)^2\frac{\sigma^2}{2}\right]\sigma\sqrt{\frac{\pi}{2}}\mathrm{Erfc}\left(\frac{(1+h)\sigma^2-\log(c)}{\sqrt{2}\sigma}\right), \tag{6.11}$$

where $\mathrm{Erfc}(\cdot)$ denotes the complementary error function. Hence, the new approximation follows in 6.12:

$$\begin{aligned}\widehat{\Phi}_Z^{re}(\omega) =& c_{(\omega,\sigma)}\left[\sum_{m=0}^{r}\frac{(-1)^m}{(m!)^2}\left(\frac{1}{2}\right)^{2m}\mathrm{Int}\left(\frac{1}{\sigma^2}\log(\omega)-1+2m,\sigma,c\right)\right.\\ &+2\sum_{k=1}^{b}\sum_{m=0}^{r}\frac{(-1)^{(m+k)}}{m!(m+2k)!}\left(\frac{1}{2}\right)^{2m+2k}\\ &\left.\times\,\mathrm{Int}\left(\frac{1}{\sigma^2}\log(\omega)-1+2(m+k),\sigma,c\right)\right].\end{aligned} \tag{6.12}$$

This approximation relies on a double instead of a triple sum, thus requiring significantly fewer evaluations of the computationally expensive error function.

6.2.3 Numerical Results for Expansion Approaches

We may numerically generate accurate results for arbitrary values of ω to assess the quality of our approximations by using a Monte Carlo estimator with the sample variable $Z \sim LN(0,\sigma^2)$ as in Equation 6.13: [5]

$$\hat{\Phi}_{Z,n}(\omega) = \frac{1}{n}\sum_{k=1}^{n} e^{i\omega Z_k}. \tag{6.13}$$

[5] [A] Estimator has not been explicitly referred to as Monte Carlo estimator in the publication.

We have determined that a sample size of $n = 10^7$ is sufficient for our needs, as larger sample sizes have not influenced the outcome of the simulation. The σ values were selected according to [Beaulieu, Saberali (2012)] and [Gubner (2006)], where a wide range of commonly used parameter values where investigated.

			Real part		**Imaginary part**	
		CPU time	Absolute error	Relative error	Absolute error	Relative error
$p = 30$	ABT	0.312002	nc	nc	nc	nc
	AHG	0.218401	0.404562	6.970940	0.317532	3.325102
	ATE	0.202801	0.098071	1.533669	0.094710	0.906951
	ABE	1.248008	0.043085	0.907838	0.081884	0.743698
$p = 20$	ABT	-	3.557634	71.452960	3.989855	39.566053
	AHG	-	0.315587	4.104002	0.164405	1.339807
	ATE	-	0.081599	1.021646	0.066494	0.548009
	ABE	-	0.026484	0.355239	0.057447	0.426060
$p = 10$	ABT	-	0.524513	3.560700	0.586144	3.484317
	AHG	-	0.152516	1.217180	0.059637	0.320366
	ATE	-	0.040261	0.279177	0.024702	0.150795
	ABE	-	0.011364	0.084232	0.030353	0.171276

Table 6.1: Standard deviation $\sigma = 2.30$, error in comparison to simulation of size $n = 10^7$. ALS left out.

			Real part		**Imaginary part**	
		CPU time	Absolute error	Relative error	Absolute error	Relative error
p = 30	ALS	∼0	nc	nc	nc	nc
	ABT	0.280802	nc	nc	nc	nc
	AHG	0.405603	0.148581	9.666702	0.121496	2.135366
	ATE	0.296402	0.078974	4.154304	0.046139	1.839942
	ABE	1.450809	0.057933	3.247697	0.062595	2.732703
p = 20	ALS	-	2.654283	208.844190	2.540205	37.404072
	ABT	-	0.859905	40.190289	2.624051	97.059373
	AHG	-	0.101641	6.990070	0.114132	1.531057
	ATE	-	0.051824	2.590322	0.030506	0.655504
	ABE	-	0.032847	1.807182	0.036726	0.814317
p = 10	ALS	-	1.744072	174.713509	1.779992	14.595444
	ABT	-	0.080129	6.798724	0.233602	1.666544
	AHG	-	0.051519	5.017843	0.072447	0.544649
	ATE	-	0.016758	1.216712	0.009415	0.076183
	ABE	-	0.011703	0.977689	0.011583	0.102156

Table 6.2: Standard deviation $\sigma = 1.38$, error in comparison to simulation of size $n = 10^7$.

We compare different methods to one another under the criteria of accuracy, computational speed and stability. To be more precise, we compare the absolute aggregate and relative error for $\omega = 1, \ldots, p$ of both the imaginary and real parts, as well as the CPU time using the `Timing` function in Mathematica. We have omitted the run times for $p = 10$ and $p = 20$, since the computational effort for more evaluation points adds up linearly. Thus the run times for $p = 10$ for example are one third of the overall run times. Tables 6.1 through 6.5 list the outcomes of simulations for different standard deviations.

			Real part		**Imaginary part**	
		CPU	Absolute	Relative	Absolute	Relative
		time	error	error	error	error
p = 30	ALS	∼0	2.403216	856.444300	2.333043	11796.501600
	ABT	0.249602	nc	nc	nc	nc
	AHG	0.390002	0.105072	15.896518	0.096489	98.336868
	ATE	0.327602	0.030766	15.225331	0.025837	45.289454
	ABE	0.826805	0.088290	49.010400	0.071125	1013.500
p = 20	ALS	-	1.715903	396.304685	1.713352	161.115716
	ABT	-	0.675569	212.587615	0.434617	129.338026
	AHG	-	0.091947	8.280796	0.089558	4.681927
	ATE	-	0.021644	8.450659	0.020610	2.180470
	ABE	-	0.039157	15.317600	0.024437	4.085990
p = 10	ALS	-	0.827391	22.685714	0.802635	28.381456
	ABT	-	0.031429	0.944421	0.033314	1.143417
	AHG	-	0.071534	1.204072	0.059236	1.868045
	ATE	-	0.002985	0.064876	0.007444	0.089640
	ABE	-	0.005781	0.191632	0.003703	0.124309

Table 6.3: Standard deviation $\sigma = 0.70$, error in comparison to simulation of size $n = 10^7$.

The abbreviations read as follows: ATE and ABE are the partial Taylor and Bessel expansions, respectively. ALS denotes the localised expansion formula and AHG stands for the hypergeometric function approximation, both put forth by Beaulieu and Saberali [Beaulieu, Saberali (2012)]. ABT stands for the truncated lognormal function as used in Beaulieu [Beaulieu (2010)].

For $\sigma = 2.3$ we have omitted the ALS method, since for standard deviation values of that size no sensible output was produced, as predicted by the original investigator [Beaulieu, Saberali (2012)]. The failure to converge or produce output has been noted by *nc* for 'no computation'. Similarly, we note that the truncated lognormal method becomes instable for either large σ or ω values. Even though this method has shown good performance for small parameter values and small computational effort, the instability severely limits its operating range.

The hypergeometric function formula was chosen as our benchmark method as it performs well for any given parameter combination with comparatively small computational effort. The number of sum terms has been chosen in accordance with the specifications made by Beaulieu for the respective σ values [Beaulieu, Saberali (2012)].

			Real part		**Imaginary part**	
		CPU time	Absolute error	Relative error	Absolute error	Relative error
p = 30	ALS	~ 0	0.215030	466.752643	0.218099	1209.040113
	ABT	0.202801	nc	nc	nc	nc
	AHG	0.390002	0.119693	80.59962	0.118690	72.49969
	ATE	0.374402	0.050140	20.30160	0.058299	16.72387
	ABE	0.436803	0.124511	1785.250	0.072307	2572.940
p = 20	ALS	-	0.182132	47.57666	0.175392	32.706530
	ABT	-	0.041104	28.65944	0.029539	25.00917
	AHG	-	0.114796	19.25712	0.113521	16.52755
	ATE	-	0.048757	2.92060	0.057162	2.341563
	ABE	-	0.018411	11.41160	0.006797	4.51306
p = 10	ALS	-	0.091365	2.103879	0.091022	1.652426
	ABT	-	0.002104	0.046956	0.001840	0.049079
	AHG	-	0.084618	2.253233	0.076022	2.313334
	ATE	-	0.047165	2.114443	0.054698	1.029962
	ABE	-	0.002095	0.044906	0.001523	0.041799

Table 6.4: Standard deviation $\sigma = 0.3$, error in comparison to simulation of size $n = 10^7$.

In contrast we can see the Bessel expansion approach being rather accurate for small values of ω, while becoming more inaccurate for large values (especially for small standard deviation). Nevertheless we did manage to obtain good accuracy near $\omega = 0$ or for large variances with this approach.

The vulnerability of the ABE approach is most likely due to the Bessel function expansion diverging the further we stray from the origin. The Taylor approach does not have this issue, since the expansion only needs to be locally stable in each individual interval $[a_k, a_{k+1}]$.

With adaptive interval and polynomial order settings (larger intervals for higher order polynomials, high order polynomials for high integrand oscillation) we managed to outperform the AHG benchmark method for the entirety of the tested values with slightly smaller CPU strain.As the number of intervals m and interval width t directly define the range of the approximation as $t \times m$. this is chosen so that the integrand $f \sim 0$. Depending on the volatility of the integrand, and the chosen number of interval width, an adequate order n of the Taylor approximation has to be determined. For small values of ω (ca. ¡15) step sizes of $t \sim 1.5$ are necessary, where larger values allow for values around $t \sim 2.5$. This in turn requires $m \sim 30$ and $m \sim 50$ intervals, respectively, for most σ values (as little as ~ 20 for the highest variance values). The

order $n \sim 5$ has proven sufficient for most computations.

			Real part		**Imaginary part**	
		CPU time	Absolute error	Relative error	Absolute error	Relative error
p = 30	ALS	0.015600	0.007522	0.074963	0.006494	0.156856
	ABT	0.187201	0.003040	0.053925	0.006083	0.182432
	AHG	0.343202	0.099302	0.489928	0.101704	0.390938
	ATE	0.312002	0.013115	0.094753	0.011093	0.359235
	ABE	0.904806	0.307162	4.126880	0.232750	1.087630
p = 20	ALS	-	0.004698	0.053300	0.003521	0.008572
	ABT	-	0.002020	0.046583	0.001494	0.004253
	AHG	-	0.021894	0.123830	0.020197	0.037871
	ATE	-	0.005960	0.055378	0.005177	0.012603
	ABE	-	0.001967	0.046330	0.001465	0.004213
p = 10	ALS	-	0.000738	0.001572	0.000725	0.001679
	ABT	-	0.000187	0.000708	0.000242	0.000875
	AHG	-	0.001397	0.003048	0.000953	0.002405
	ATE	-	0.001030	0.002032	0.000812	0.001916
	ABE	-	0.000187	0.000708	0.000242	0.000875

Table 6.5: Standard deviation $\sigma = 0.05$, error in comparison to simulation of size $n = 10^7$.

For an especially small standard deviation (e.g. $\sigma = 0.05$) the ALS shows great accuracy, with almost no computational effort (note that Mathematica sets 0.0156 per default as smallest time increment, hence we get values between 0 and 0.0156). As mentioned in [Beaulieu, Saberali (2012)] the closed-form is only applicable for σ converging to zero. The partial Taylor expansion of the ALS method is developed around its evaluation point $\sigma = 0$, therefore the expansion is increasingly unreliable for larger σ values. Beaulieu and Saberali have narrowed down the range where the closed-form remains accurate to $\sigma < 0.3$. Therefore the only two stable approximations, regardless of the range of their input parameter ω and σ, are the AHG and ATE approaches.

6.3 Integral Transformation and Quadrature Methods

6.3.1 Chebyshev-Type Quadrature

Gubner [Gubner (2006)] proposed a quadrature approach for the CF integration, which focused around the transformation of the integrand to a function which can be easily evaluated by numerical methods. The choice of a suitable integration method depends on the image of the transformation. In Table 6.6 we find a selection of commonly used quadrature formulas, as taken from Abramowitz and Stegun [Abramowitz, Stegun (1964)].

Name	Interval	Orthogonal polynomials	Weights
Gauss-Legendre quadrature	[-1,1]	Legendre polynomials	1
Gauss-Hermite quadrature	$(-\infty, \infty)$	Hermite polynomials	e^{-x^2}
Chebyshev-Gauss quadrature	(-1,1)	Chebyshev polynomials (first kind)	$\frac{1}{\sqrt{1-x^2}}$
Chebyshev-Gauss quadrature	[-1,1]	Chebyshev polynomials (second kind)	$\sqrt{1-x^2}$

Table 6.6: Common quadrature methods.

Gubner considers the CF as a contour integral, along the path

$$C = \{z(t) = t + i\pi/2 : -\infty < t < \infty\}$$

. From this derives 6.14: [6]

$$\begin{aligned}\Phi_Z(\omega) &= \int_C \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{z^2}{2\sigma^2}} e^{i\omega e^z} dz \\ &= \frac{\exp\left[(\pi/(2\sigma))^2/2\right]}{\sqrt{2\pi}\sigma} \int_{\mathbb{R}} e^{-\omega e^t} e^{-i\pi t/(2\sigma^2)} e^{-(t/\sigma)^2/2} dt\end{aligned}$$

[6][C] Gubner uses $\mathbb{E}[e^{i\omega e^Z}]$ with Z being normally distributed.

$$\overset{(*)}{=} \frac{\exp\left[(\pi/(2\sigma))^2/2\right]}{\sqrt{\pi}} \underbrace{\int_{\mathbb{R}} e^{-\omega e^{\sqrt{2}\sigma x}} e^{-i\pi x/(\sqrt{2}\sigma)} e^{-x^2} dx}_{\psi(\omega)}. \tag{6.14}$$

With $(*)x = t/(\sqrt{2}\sigma)$. [7] This formulation lends itself well to the Hermite-Gauss quadrature, through which we generate an approximation for $\psi(\omega)$. The weights w_k for the approximation order n can be iteratively generated by introducing the k-th root of the n-th order Hermite polynomial $x_k^{(n)}$ into the $(n-1)$-th order polynomial:

$$\psi(\omega) \approx \sum_{k=1}^{n} w_k e^{-\omega e^{\sqrt{2}\sigma x_k^{(n)}}} e^{-i\omega x_k^{(n)}/(\sqrt{2}\sigma)}. \tag{6.15}$$

The original paper [Gubner (2006)] suggested that the maximum accuracy is reached around an order of the Hermite polynomial of $n = 45$. The results of the quadrature become stationary beyond that point, which we were able to confirm in our replications.

Although the quadrature method showed great stability and performance speed we will try improving upon this approach by introducing a slightly different transformation and consequently quadrature method. We start out with expression 6.14 and introduce the remapping $x = L\cot(\theta)$, where L is an approximation parameter [Boyd (1987)]:

$$\begin{aligned}
\psi(\omega) &= \int_{\mathbb{R}} e^{-\omega e^{\sqrt{2}\sigma x}} e^{-i\omega x/(\sqrt{2}\sigma)} e^{-x^2} dx \\
&= L \int_0^{\pi} \frac{1}{\sin^2(\theta)} \underbrace{e^{-\omega e^{\sqrt{2}\sigma(L\cot\theta)}} e^{-i\omega L\cot(\theta)/(2\sigma)1/2} e^{-(L\cot(\theta))^2}}_{=g(L\cot\theta)} d\theta \\
&\approx \frac{L\pi}{N} \sum_{k=1}^{N-1} \frac{1}{\sin^2(k\pi/N)} g\left(L\cot\left(\frac{k\pi}{N}\right)\right).
\end{aligned} \tag{6.16}$$

This approximation is essentially a specialisation of the Chebyshev-Gauss quadrature, which utilises the periodicity of the transformed integrand. The numerical integration may reach up to an exponential rate of convergence. The choice of L can heavily influence the outcome of the approximation and the necessary number of terms. Through experimentation we have found $L \sim N$ or lower to be optimal for our purposes, achieving convergence in as little as 15 terms.

[7] [A] Some of Gubner's formula had been copied with a number of typing errors.

6.3.2 Numerical Results for Quadrature Methods

We generate control results as before in Section 6.2 and investigate the performance of the quadrature and numerical integration approaches (Tables 6.7 - 6.11).

The method denoted by ACI was proposed by Beaulieu in 'A simple integral form of the lognormal characteristic functions convenient numerical computation' [Beaulieu (2006)]. The proposed reformulation of the CF was specifically constructed in [Beaulieu (2006)] for easier numerical evaluation by the trapezoidal method. By ARB we refer to Bakarat's approach utilising Taylor and Hermite polynomials [Bakarat (1976)].

			Real part		**Imaginary part**	
		CPU time	Absolute error	Relative error	Absolute error	Relative error
p = 30	ACI	1.201208	0.312258	6.725943	0.242923	2.506923
	ARB	0.046800	260.1168	673.9727	399.4355	1605.971
	AHW	0.109201	0.005609	0.096914	0.005893	0.061754
	ACC	0.046800	0.005374	0.094003	0.005726	0.059398
p = 20	ACI	-	0.168468	1.949448	0.135603	1.093077
	ARB	-	259.8095	663.9727	398.6571	1595.971
	AHW	-	0.004061	0.046484	0.002747	0.021391
	ACC	-	0.003969	0.048314	0.002704	0.020812
p = 10	ACI	-	0.102310	0.774836	0.050557	0.296870
	ARB	-	259.2383	653.9727	397.5900	1585.971
	AHW	-	0.002445	0.017995	0.001162	0.006519
	ACC	-	0.002332	0.017802	0.001264	0.007244

Table 6.7: Standard deviation $\sigma = 2.30$, error in comparison to simulation of size $n = 10^7$.

			Real part		**Imaginary part**	
		CPU time	Absolute error	Relative error	Absolute error	Relative error
p = 30	ACI	1.263608	0.030297	1.539971	0.056413	2.675018
	ARB	0.078001	74.46770	612.7762	54.14810	199.7956
	AHW	0.046800	0.005884	0.280866	0.005856	0.209535
	ACC	∼0	0.005736	0.292645	0.005849	0.207896
p = 20	ACI	-	0.017747	0.803512	0.029743	0.630684
	ARB	-	74.29133	602.7767	54.00300	189.7956
	AHW	-	0.004543	0.200430	0.003795	0.072062
	ACC	-	0.004549	0.220980	0.003820	0.072902
p = 10	ACI	-	0.007221	0.381207	0.005600	0.044814
	ARB	-	74.03240	592.7762	53.5993	179.7956
	AHW	-	0.002234	0.112679	0.001735	0.013930
	ACC	-	0.002148	0.129722	0.001736	0.013943

Table 6.8: Standard deviation $\sigma = 1.38$, error in comparison to simulation of size $n = 10^7$.

The AHW and ACC are both quadrature methods, one using Hermite-Gauss quadrature as stated by Gubner [Gubner (2006)], whereas we devised the later approach based on a Chebyshev quadrature in 6.16 (denoted as ACC).

			Real part		**Imaginary part**	
		CPU time	Absolute error	Relative error	Absolute error	Relative error
p = 30	ACI	1.560010	0.006288	2.588322	0.004892	47.773835
	ARB	0.046800	1.568322	33.5406	3.459413	108.6734
	AHW	0.046800	0.005940	2.470303	0.004889	53.0809
	ACC	0.015600	0.005788	2.395781	0.004978	51.7819
p = 20	ACI	-	0.004467	1.322967	0.002347	0.233787
	ARB	-	1.551014	23.5406	3.453067	98.6734
	AHW	-	0.004156	1.218673	0.002345	0.234062
	ACC	-	0.004009	1.153123	0.002383	0.241749
p = 10	ACI	-	0.002176	0.039155	0.001227	0.041273
	ARB	-	1.518430	13.5387	3.346234	88.6735
	AHW	-	0.002178	0.039288	0.001234	0.041471
	ACC	-	0.002121	0.038010	0.001249	0.043617

Table 6.9: Standard deviation $\sigma = 0.70$, error in comparison to simulation of size $n = 10^7$.

			Real part		**Imaginary part**	
		CPU time	Absolute error	Relative error	Absolute error	Relative error
p = 30	ACI	1.201208	0.004813	18.372205	0.005173	16.355073
	ARB	0.046800	0.066545	18.875980	0.069413	17.206620
	AHW	0.046800	0.004813	18.371627	0.005172	16.344597
	ACC	0.015600	0.004845	17.998674	0.005131	16.337516
p = 20	ACI	-	0.003436	0.862905	0.004030	1.359070
	ARB	-	0.064301	8.115747	0.066770	7.030268
	AHW	-	0.003436	0.862865	0.004030	1.359085
	ACC	-	0.003477	0.866225	0.003988	1.342209
p = 10	ACI	-	0.002080	0.044183	0.001519	0.041719
	ARB	-	0.036754	1.434342	0.041446	1.011095
	AHW	-	0.002080	0.044184	0.001519	0.041719
	ACC	-	0.002122	0.044253	0.001548	0.045194

Table 6.10: Standard deviation $\sigma = 0.30$, error in comparison to simulation of size $n = 10^7$.

As we can see in the performance tables of higher valued standard deviation, the Bakarat approach does not remain stable for arbitrary σ values, despite producing good results for values of roughly $\sigma < 0.5$ with little computational effort.

The AHW approach serves as a benchmark quadrature method, and is able to produce accurate results in a runtime of less than 0.11 seconds for the 30 ω values, but failed to converge for σ close to zero. Similarly, the ACC method struggles in the same area of values, as the approximation works much in the same way as the Hermite quadrature does.

			Real part		Imaginary part	
		CPU time	Absolute error	Relative error	Absolute error	Relative error
p = 30	ACI	0.561604	0.003458	0.065723	0.002894	0.170994
	ARB	0.046800	0.003458	0.065723	0.002894	0.170952
	AHW	-	nc	nc	nc	nc
	ACC	-	nc	nc	nc	nc
p = 20	ACI	-	0.002020	0.046583	0.001494	0.004253
	ARB	-	0.002020	0.046583	0.001494	0.004254
	AHW	-	nc	nc	nc	nc
	ACC	-	nc	nc	nc	nc
p = 10	ACI	-	0.000187	0.000708	0.000241	0.000875
	ARB	-	0.000187	0.000708	0.000242	0.000875
	AHW	-	nc	nc	nc	nc
	ACC	-	nc	nc	nc	nc

Table 6.11: Standard deviation $\sigma = 0.05$, error in comparison to simulation of size $n = 10^7$.

However, with the ACC approach we are able to achieve an accuracy comparable to the AHW method (in σ regions where both methods converge), in only a quarter of AHW's computation time. We attribute this to the correct choice of the parameter L and the fact that no wights or nodes have to be pre-determined for evaluation.

6.4 Conclusions

In Sections 6.2 and 6.3, we have introduced a new approximation approaches and compared them to existing methods. While the Bessel function based approximation exhibited the greatest accuracy for specific parameter values, both the ATE and ACC approaches are more widely applicable due to their stability. With the exception of σ values very close to zero (< 0.25), the Chebyshev quadrature has shown the best overall performance. We believe that pairing the Chebyshev quadrature for larger variances with the ALS approach for small variances may provide a strong technique for the accurate computation of the lognormal CF for arbitrary parameter values. Given that we may not know the parameters for which the chosen approximation must perform accurately, the Taylor expansion approach provides a viable alternative, taking into account its computational effort, accuracy and stability across the entirety of tested parameter ranges. While the combination method of quadrature and closed-form approaches might offer a quick and accurate evaluation, the vulnerability of the local

approximations to certain ranges of standard deviations may not offer enough stability for some applications. [8]

We did consider Filon-type integration for the highly oscillatory integrands with $\sigma < 0.25$, as for example suggested by Iserles and Norsett [Iserles, Norsett (2000)], but the accuracy and simplicity of computation inherent to the ALS method in this range makes it the preferred bridge for the gap the quadrature methods leaves open. The only approaches which are truly applicable to arbitrary domains of values are the partial Taylor expansion, the hypergeometric function approach and Beaulieu's convenient integral formulation, amongst which we have determined the Taylor expansion to be the most efficient way of computation.

[8] [C] It is noteworthy that none of the investigated methods perform consistently best, or even without any drawbacks in all combinations of parameter regions. The partial Taylor expansion method did not appear to be divergent or completely inadequate for any parameter combination, and could be seem as a 'middle ground' solution that provides reasonable results without having to pay attention to parameter ranges which may cause instability. Furthermore, the expansions could be extended to include dynamic term numbers, and intervals based on ω, σ as well as the placement of the integration limits (a_k, a_{k+1}) with respect to x.

Chapter 7

A Series Representation for Multidimensional Rayleigh Distribution

Chapter Abstract

The Rayleigh distribution is of paramount importance in signal processing and many other areas, yet an expression for the probability density function of arbitrary dimensions has remained elusive. In this chapter, we generalise the results of Beard and Tekinay [Beard, Tekinay (2017)] for quadrivariate random variables to cases of unconstrained order and provide a simple algorithm for evaluation. The assumptions of cross-correlation between in-phase and quadrature, as well as non-singularity of the covariance matrix, are retained throughout our computations.

7.1 Introduction

Correlated Rayleigh random variables arise in signal processing and many other areas: correlated Rayleigh fading channels, correlated Rayleigh scattering spectroscopy, correlated Rayleigh envelopes, correlated Rayleigh co-channel interferers, correlated Rayleigh clusters and correlated Rayleigh targets; to mention just a few.

For correlated Rayleigh random variables Rice [Rice (1944)] and Miller [Miller (1969)] obtained probability distribution representations for the bivariate and trivariate cases. Their method of expressing the distribution via an underlying Gaussian distribution has still been utilised in recent publications and will be essential to our approach as well.

On the basis of the previous progressions, Beard and Tekinay [Beard, Tekinay (2017)] have derived a series representation for a quadrivariate Rayleigh distribution around a Bessel function expansion. We believe that the dimensional restriction can be relaxed, while the original assumption of non-singularity of the covariance matrix is maintained. In Section 7.2, we circumnavigate the denomination problem of increasingly many Bessel function sum terms, as noted by Beard and Tekinay [Beard, Tekinay (2017)], with a Cauchy sum. The resulting integrals can then be more easily evaluated via the complex exponential notation of the cosine function. We also provide a pseudocode for the algorithm. Some practical applications of the pseudocode including runtimes are given in Section 7.3. Some simulation results are given in Section 7.4. Some final remarks on what has been done are given in Section 7.5. Throughout, only the formulas cited in the text are numbered.

7.2 Multivariate Rayleigh Distribution

We begin by introducing the $2n$ dimensional random variable

$$Z = \left(z_{I_1}, z_{Q_1}, \ldots, z_{I_n}, z_{Q_n}\right),$$

where I_i represents the in-phase and Q_i the quadrature part of a signal. The joint distribution is assumed to be a $2n$-variate Gaussian ($\mu = 0$, $\sigma^2 = \zeta$), with the distribution function given below (as in Beard and Tekinay [Beard, Tekinay (2017)] or Rice [Rice (1944)]):

$$f\left(z_{I_1}, z_{Q_1}, \ldots, z_{I_n}, z_{Q_n}\right) = \frac{1}{(2\pi)^n |K|^{1/2}} \exp\left(-\frac{Z^T K^{-1} Z}{2}\right), \tag{7.1}$$

where

$$K = \zeta \begin{bmatrix} 1 & 0 & \rho_1 & \cdots & & \cdots & \rho_{n-1} & 0 \\ 0 & 1 & 0 & \rho_1 & \cdots & & \cdots & \rho_{n-1} \\ \rho_1 & 0 & 1 & & & & & \vdots \\ \vdots & \rho_1 & & \ddots & & & & \\ & \vdots & & & & & & \vdots \\ & & & & & \ddots & & \rho_1 \\ & & & & & & 1 & 0 \\ & & & & \cdots & \rho_1 & 0 & 1 \end{bmatrix} \in \mathbb{R}^{2n} \tag{7.2}$$

denotes the covariance matrix for $2n$ dimensions with values $-1 \leq \rho_i \leq 1$. The in-phase and quadrature parts are presumed to be uncorrelated, for example, $\mathbb{E}\left[z_{I_i} z_{Q_i}\right] = 0$ without loss of generality. A more general result for correlated random variables can be obtained analogously when computing $f_{R,\theta}$. Note that $Cor(Z_{I_i}, Z_{I_j}) = Cor(Z_{Q_i}, Z_{Q_j}) = \rho_{|i-j|}$ for $i \neq j$ denotes the correlation.

We transform the cartesian coordinates of the input vector $z \in \mathbb{R}^{2n}$ into polar coordinates:

$$z = \begin{bmatrix} r_1 \cos(\theta_1) \\ r_1 \sin(\theta_1) \\ \vdots \\ \\ \\ \vdots \\ r_n \cos(\theta_n) \\ r_n \sin(\theta_n) \end{bmatrix}.$$

The determinant of the Jacobian for the transformation $|J| = r_1 \cdots r_n$ can hence be written as a factor outside the exponential function.

To further expand the matrix vector product, we determine a general expression for the inverse matrix. We employ Cramer's rule utilising the cofactor matrix C such that $K^{-1} = \frac{1}{|K|} C^T$ holds:

$$C = \begin{bmatrix} c_0 & 0 & c_1 & \cdots & & \cdots & c_{n-1} & 0 \\ 0 & c_0 & 0 & c_1 & \cdots & & \cdots & c_{n-1} \\ c_1 & 0 & c_0 & & & & & \vdots \\ \vdots & c_1 & & \ddots & & & & \\ & \vdots & & & & & & \vdots \\ & & & & & \ddots & & c_1 \\ & & & & & & c_0 & 0 \\ & & & & \cdots & c_1 & 0 & c_0 \end{bmatrix}. \tag{7.3}$$

We note that the cofactor matrix has to retain the original shape of K, which aids in the evaluation of the exponent.

We introduce (7.2) and (7.3) into the $2n$-variate Gaussian distribution in (7.1), and apply common trigonometric identities [Abramowitz, Stegun (1964)] onto the resulting sine and cosine product terms. This yields the following result:

$$\begin{aligned} f\left(r_1, \theta_1, \ldots, r_n, \theta_n\right) &= \frac{|J|}{(2\pi)^n |K|^{1/2}} \\ &\quad \times \exp\left(-\frac{1}{2|K|} \left(\sum_{i=1}^{n} r_i^2 c_0 + 2 \sum_{(i,k,l \in P^n)} c_i r_k r_l \cos\left(\theta_k - \theta_l\right) \right) \right) \\ &= \underbrace{\frac{|J|}{(2\pi)^n |K|^{1/2}} \exp\left(-\frac{1}{2|K|} \sum_{i=1}^{n} r_i^2 c_0 \right)}_{=\gamma_{n,K}} \\ &\quad \times \prod_{(i,k,l \in P^n)} \exp\left(-\frac{1}{|K|} c_i r_k r_l \cos\left(\theta_k - \theta_l\right) \right). \end{aligned}$$

Here, $P^n = \left\{ (i,k,l) \in \mathbb{N}^3 |\ 1 \leq l < k \leq n,\ i = 1, \ldots, n-1,\ |k-l| = i \right\}$ is the set of all feasible coefficient combinations of c_i and r_i as they arise from the vector matrix product. We relabel the variables according to the counting scheme of Algorithm 7.1, and dispose of the set P^n in favour of a summation for $t = 1, \ldots, n(n-1)/2 = p = |P^n|$.

First we substitute the theta variables by $x_{l-1} - x_{k-1} = \theta_k - \theta_l$ [Beard, Tekinay (2017)], where $x_0 = 0$ needs to be introduced for consistency.

Algorithm 7.1: Renaming algorithm.

Result: Retain new variables $\overline{x}_t$ and coefficients a_t

```
Renaming t=1
for l = 1, ..., n do
    for k = l + 1, ..., n do
        x̄_t = x_{l-1} − x_{k-1};
        a_t = −(1/|K|) c_{|k−l|} r_k r_l;
        t++;
    end
end
```

This naturally leaves us with a more compact version of the density as we can see below: [1]

$$f\left(r_1, \theta_1, \ldots, r_n, \theta_n\right) = \gamma_{n,K} \prod_{t=1}^{p} \exp\left(a_t \cos\left(\overline{x}_t\right)\right). \tag{7.4}$$

Next we consider an n-variate Rayleigh distribution by generating the marginal probability distribution. Integration over the angle component of the polar coordinates reduces the function in (7.4) to an n-dimensional one, solely dependent on the radius component:

$$f\left(r_1, \ldots, r_n\right) = \gamma_{n,K} \int_0^{2\pi} \cdots \int_0^{2\pi} \prod_{t=1}^{p} \exp\left(a_t \cos\left(\overline{x}_t\right)\right) dx_1 \cdots dx_n.$$

The composition of exponential and cosine functions proves to be difficult to integrate analytically, we hence use the following Bessel function expansions [Beard, Tekinay (2017)] [Abramowitz, Stegun (1964)] to reduce the problem to a trigonometric integral:

$$\exp\left[a\cos(x)\right] = I_0(a) + 2\sum_{j=1}^{\infty} I_j(a)\cos(jx),$$

$$\exp\left[-a\cos(x)\right] = I_0(a) + 2\sum_{j=1}^{\infty} (-1)^j I_j(a)\cos(jx).$$

[1] [A] Grammar corrected.

Additionally, we rename the appearing coefficients, to combine both cases, therefore making the series representation more lucid to us and to reduce the exceptions:

$$b_{t,j_t} = \begin{cases} I_0\left(|a_t|\right), & \text{if } j_t = 0, \\ 2(-1)^{j_t \mathbb{I}_{(a_t<0)}} I_{j_t}\left(|a_t|\right), & \text{if } j_t > 0. \end{cases} \tag{7.5}$$

With the series expansion and revised notation in place, we can move on to the arithmetic:

$$\begin{aligned}
& f\left(r_1, \ldots, r_n\right) \\
=& \gamma_{n,K} \int_0^{2\pi} \cdots \int_0^{2\pi} \prod_{t=1}^{p} \left[I_0\left(a_t\right) + \sum_{j_t=1}^{\infty} I_{j_t}\left(a_t\right) \cos\left(j_t \overline{x}_t\right) \right] dx_1 \cdots dx_n \\
\overset{(7.5)}{=}& \gamma_{n,K} \int_0^{2\pi} \cdots \int_0^{2\pi} \prod_{t=1}^{p} \sum_{j_t=0}^{\infty} b_{t,j_t} \cos\left(j_t \overline{x}_t\right) dx_1 \cdots dx_n \\
=& \gamma_{n,K} \int_0^{2\pi} \cdots \int_0^{2\pi} \sum_{j_1=0}^{\infty} \sum_{j_2=0}^{j_1} \cdots \sum_{j_p=0}^{j_{p-1}} b_{1,j_p} \cos\left(j_p \overline{x}_1\right) b_{2,j_{p-1}-j_p} \cdots \\
& \cdots b_{p,j_1-j_2} \cos\left(\left(j_1 - j_2\right) \overline{x}_p\right) dx_1 \cdots dx_n \\
\overset{j_{p+1}=0}{=}& \gamma_{n,K} \int_0^{2\pi} \cdots \int_0^{2\pi} \sum_{j_1=0}^{\infty} \sum_{j_2=0}^{j_1} \cdots \sum_{j_p=0}^{j_{p-1}} \prod_{t=1}^{p} b_{t,j_{p-t+1}-j_{p-t+2}} \cos\left(\left(j_{p-t+1} - j_{p-t+2}\right) \overline{x}_t\right) dx_1 \cdots dx_n \\
=& \gamma_{n,K} \sum_{j_1=0}^{\infty} \sum_{j_2=0}^{j_1} \cdots \sum_{j_p=0}^{j_{p-1}} \prod_{t=1}^{p} b_{t,j_{p-t+1}-j_{p-t+2}} \qquad (7.7) \\
& \times \int_0^{2\pi} \cdots \int_0^{2\pi} \prod_{t=1}^{p} \cos\left(\left(j_{p-t+1} - j_{p-t+2}\right) \overline{x}_t\right) dx_1 \cdots dx_n. \qquad (7.8)
\end{aligned}$$

Expression 7.8 was derived by introducing a Cauchy product into Equation [Apostol (1974)], enabling us to exchange product and sum of the series expansion, which in turn allows us then to further simplify the integral (convergence holds as a result of the expansion).

We then insert Euler representation of the cosine $\cos(z) = \frac{1}{2}\left[\exp(iz) + \exp(-iz)\right]$ into (7.8) to alleviate the integral evaluation. This yields the following result with the now modified index $j_t^* = j_{p-t+1} - j_{p-t+2}$ and we can determine the complex integral:

$$f\left(r_1, \ldots, r_n\right) = \gamma_{n,K} \sum_{j_1=0}^{\infty} \sum_{j_2=0}^{j_1} \cdots \sum_{j_p=0}^{j_{p-1}} \prod_{t=1}^{p} b_{t,j_t^*} \int_0^{2\pi} \cdots \int_0^{2\pi} \prod_{t=1}^{p} \cos\left(j_t^* \overline{x}_t\right) dx_1 \cdots dx_n$$

$$
\begin{aligned}
=&\gamma_{n,K}\sum_{j_1=0}^{\infty}\sum_{j_2=0}^{j_1}\cdots\sum_{j_p=0}^{j_{p-1}}\prod_{t=1}^{p}b_{t,j_t^*}\\
&\times\int_0^{2\pi}\cdots\int_0^{2\pi}\prod_{t=1}^{p}\frac{\left(\exp\left(ij_t^*\overline{x}_t\right)+\exp\left(-ij_t^*\overline{x}_t\right)\right)}{2}dx_1\cdots dx_n\\
=&\frac{\gamma_{n,K}}{2^p}\sum_{j_1=0}^{\infty}\sum_{j_2=0}^{j_1}\cdots\sum_{j_p=0}^{j_{p-1}}\prod_{t=1}^{p}b_{t,j_t^*}\\
&\times\int_0^{2\pi}\cdots\int_0^{2\pi}\sum_{\boldsymbol{\rho}\in\{-1,1\}^p}\prod_{t=1}^{p}\exp\left(ij_t^*\rho_t\overline{x}_t\right)dx_1\cdots dx_n\\
=&\frac{\gamma_{n,K}}{2^p}\sum_{j_1=0}^{\infty}\sum_{j_2=0}^{j_1}\cdots\sum_{j_p=0}^{j_{p-1}}\prod_{t=1}^{p}b_{t,j_t^*}\sum_{\boldsymbol{\rho}\in\{-1,1\}^p}\\
&\times\int_0^{2\pi}\cdots\int_0^{2\pi}\exp\left(i\sum_{t=1}^{p}j_t^*\rho_t\overline{x}_t\right)dx_1\cdots dx_n,
\end{aligned}
\tag{7.9}
$$

where $\boldsymbol{\rho}=(\rho_1,\ldots,\rho_p)$.

As all the factors $j_t^*\rho_t\in\mathbb{Z}$, it is clear that the integral in 7.9 is exactly non-zero ($(2\pi)^n$ in fact), when the exponent itself is zero [Gradshteyn, Ryzhik (2007)]. This is precisely the case when the sum over the coefficients corresponding to their x_i are zero, for all x_i. To further clarify this, we denote $\alpha_t=j_t^*\rho_t$ and design two upper triangular matrices with $\overline{x}_t$, $t=1,\ldots,p$ and their corresponding coefficients as their respective elements:

$$
X=\begin{bmatrix}
x_1 & x_1-x_2 & x_2-x_3 & \cdots & \cdots & x_{n-3}-x_{n-2} & x_{n-2}-x_{n-1}\\
x_2 & x_1-x_3 & x_2-x_4 & & & x_{n-3}-x_{n-1} & 0\\
x_3 & x_1-x_4 & \vdots & & & 0 & \vdots\\
x_4 & \vdots & & & & \vdots & \\
\vdots & & \vdots & & & & \\
 & \vdots & x_2-x_{n-1} & & & & \\
\vdots & x_1-x_{n-1} & 0 & & & & \vdots\\
x_{n-1} & 0 & & & & \cdots & 0
\end{bmatrix},
$$

$$A = \begin{bmatrix} \alpha_1 & \alpha_n & \cdots & \cdots & \alpha_{p-2} & \alpha_p \\ \alpha_2 & \alpha_{n+1} & & & \alpha_{p-1} & 0 \\ \alpha_3 & \vdots & & & 0 & \vdots \\ \vdots & & & & \vdots & \\ & & & & & \\ & \vdots & & & & \\ \vdots & \alpha_{2n-2} & 0 & & & \vdots \\ \alpha_{n-1} & 0 & & & \cdots & 0 \end{bmatrix}.$$

The matrix entries for A have been generated as $A_{o,p} = \alpha_{\sum_{i=0}^{p-1}(n-i-1)} + o$ (filling the columns with progressively fewer elements). We can obtain the sum over all coefficients for one x_i with their respective signs by adding the n-th column to the n-th entry in the first column, and then subtracting the minor counter diagonal's negative entries. We denote the resulting i-th sum as follows:

$$\Sigma_{x_i} = A_{i,1} + \sum_{w=1}^{n-i} A_{w,i} - \sum_{w=1}^{i-1} A_{i-w,1+w}.$$

We can now very simply write the integral as a product of Kronecker Deltas, dependent on the previously defined coefficient sums. [2] With this last step we are thus finished:

$$f\left(r_1, \ldots, r_n\right) = (2\pi)^n / 2^p \gamma_{n,K} \sum_{j_1=0}^{\infty} \sum_{j_2=0}^{j_1} \cdots \sum_{j_p=0}^{j_{p-1}} \prod_{t=1}^{p} b_{t,j_t^*} \sum_{\boldsymbol{\rho} \in \{-1,1\}^p} \prod_{w=1}^{n} \delta_{\{\Sigma_{x_w}=0\}}. \tag{7.10}$$

Equation 7.10 enables us to formulate a simple algorithm to evaluate the PDF of the Rayleigh distribution for a given vector $r_1, \ldots, r_n$ in arbitrary dimensions. [3] We provide a pseudocode below to illustrate the basic principles on how the series expansion can be evaluated. [4]

[2] [A] Spelling.

[3] [A] The published version erroneously states π^p. As the implementation of both Rayleigh distribution chapters use the coefficient $\frac{|J|}{2^p |K|^{1/2}} \exp(\ldots)$, e.g. where both $(2\pi)^n$ terms have canceled out, this has no consequence.

[4] [A] First for loop has been amended, as it depicted infinitely many terms. This has been changed to N.

Algorithm 7.2: Evaluation algorithm with notation as before and a_t as introduced in Algorithm 7.1.

Result: Approximation to the PDF of an n-variate Rayleigh distribution for $r_1, \ldots, r_n$

temp=0;
for $j_1 = 0, \ldots, N$ **do**
  $\ldots$
  for $j_p = 0, \ldots, j_{p-1}$ **do**
    $c = \prod_{t=1}^{p} b_{t,j_{p-t+1}-j_{p-t+2}}$;
    for $\boldsymbol{\rho} \in \{-1, 1\}^p$ **do**
      /* Set up the coefficient matrix */
      for $o = 1, \ldots, n-1$ **do**
        for $p = 1, \ldots, n-o$ **do**
          $t = \sum_{i=0}^{p-1}(n-i-1) + o$;
          $A_{o,p} = (j_{p-t+1} - j_{p-t+2})\,\rho_t$;
        end
      end
      /* Construct coefficient sums */
      for $i = 1, \ldots, n-1$ **do**
        $S_i = A_{i,1} + \sum_{w=1}^{n-i} A_{w,i} - \sum_{w=1}^{i-1} A_{i-w,1+w}$
      end
      /* Determine integral and add non-zero terms to the return value */
      if $S_i == 0$ *for all* i **then**
        temp = temp + c;
      end
    end
  end
end
return $(2\pi)^n / 2^p \gamma_{n,K}$ temp

7.3 Applications

7.3.1 Outage Probabilites

Chen and Tellambura [Chen, Tellambura (2005)] discuss the application of multivariate Rayleigh distributions to determine the outage probability of three and four branch selection combining in correlated Rayleigh fading. The outage probability is defined below according to [Chen, Tellambura (2005)], where γ is an output threshold and

$\gamma_1, \ldots, \gamma_n$ are the respective average outputs:

$$\mathbb{P}_{out} = F_R\left(\sqrt{\frac{\gamma\rho_1}{\gamma_1}}, \ldots, \sqrt{\frac{\gamma\rho_n}{\gamma_n}}\right). \tag{7.11}$$

Where F_R is multivariate CDF of the Rayleigh distribution, with a covariance matrix K filled with the exemplary random values $\rho_0 \sim 0.5, \rho_1 \sim -0.12, \rho_2 \sim -0.09$ as covariances (values which we will recycle for the simulation testing in Section 7.4 later on). Consider $\left(\sqrt{\gamma\rho_1/\gamma_1}, \ldots, \sqrt{\gamma\rho_n/\gamma_n}\right) = (1, 1, 1)$, such that the outage probability is visualised by the area underneath our density plot, see Figure 7.1. The following plot therefore represents the probability in $\mathbb{P}[\mathbf{x} = \bar{\gamma}]$. To visualise the three-dimensional density, we have decided to use results along different three-dimensional vectors $(1, 1, 1), (2/3, 2/3, 1/3)$ and $(2/3, 1/3, 1/3)$. The plot depicts the approximation values of the density for γ being set to the scaled vectors $x * (1, 1, 1)$, $x * (2/3, 2/3, 1/3)$ and $x * (2/3, 1/3, 1/3)$ on the y-axis in relation to x depicted on the x-axis.

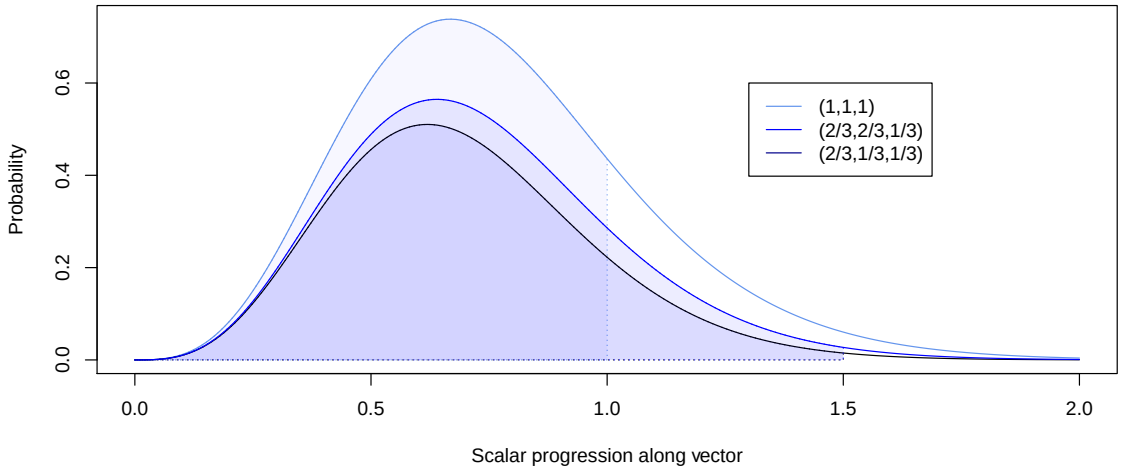


Figure 7.1: Three dim. Rayleigh density approximation, with the area underneath representing the outage probability.

Alternatively we can write the input as $\sqrt{\gamma}\bar{\gamma}$ where $\bar{\gamma} = \left(\sqrt{\rho_1/\gamma_1}, \ldots, \sqrt{\rho_n/\gamma_n}\right)$. Note that the direction vectors are scaled; hence the cutoff points for the value $\sqrt{\gamma}$ are $(1, 1, 1), (1, \frac{1}{2}, \frac{1}{2})$ and $(1.\frac{1}{2}, \frac{1}{2})$ for the different directions, visualised by the dotted line at the scalar value 1.

A multivariate Integral over the Rayleigh Distribution then reveals the cumulative distribution function, for example, the outage probability as described in (7.11). We have formulated the probabilities as such below:

$$\mathbb{P}_{out}(\gamma) = \int_0^{\sqrt{\frac{\gamma\rho_1}{\gamma_n}}} \cdots \int_0^{\sqrt{\frac{\gamma\rho_1}{\gamma_1}}} f_R(t_1, \ldots, t_n) dt_1 \ldots dt_n. \tag{7.12}$$

The evaluation of the integral in 7.12 is carried out numerically. While this could be improved on for practical purposes in future work, it is sufficient for the sole purpose of providing application examples. We visualise these results for different configurations of ρ_i and γ_i in Figure 7.2. The x axis holds the given threshold γ, more specifically $\sqrt{\gamma}$ which we treat as a scalar factor to the different vectors of $\bar{\gamma}$ in $\mathbb{P}[\mathbf{x} = \bar{\gamma}]$.

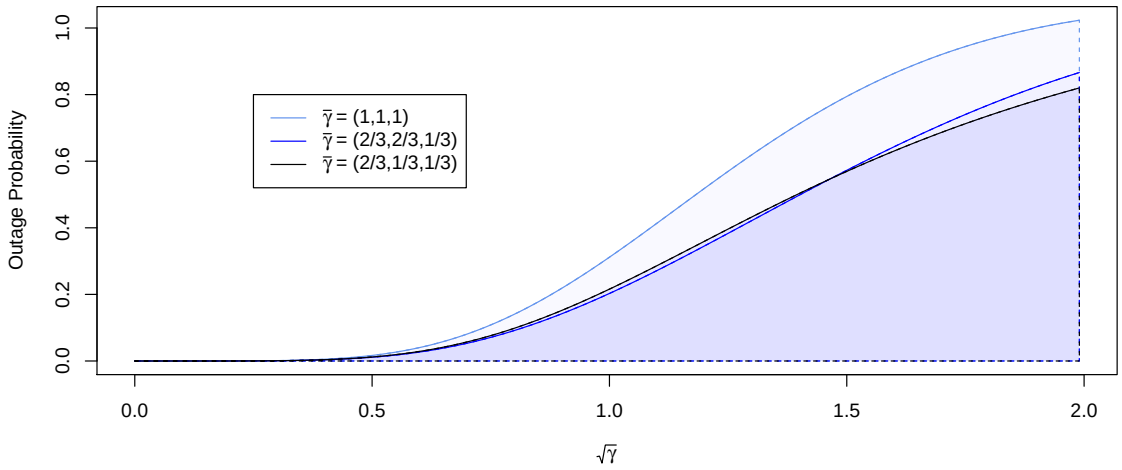


Figure 7.2: Outage probability for a three dimensional model, computed through the approximate CDF of the the respective Rayleigh distribution.

As the dimension of 3 was chosen arbitrarily, and can be done for higher orders simultaneously, we have therefore found a mode of computation of outage probabilites for arbitrary branch selection combining correlated Rayleigh fading.

7.3.2 AMC Level Change Probabilites

The second application is based on a stochastic channel model expanded upon by Beard and Tekinay [Beard, Tekinay (2017)]. Here the authors intend to compute the probability for a channel to change adaptive modulation and coding levels (AMC) from one level to another. The introduction of an additional correlation variable (time - frequency) mandated a quadrivariate Rayleigh distribution representation.

Without the dimensional restriction, we can now evaluate arbitrary Rayleigh distributions, necessitated by models which use more than four correlated input variables. In Figure 7.3 we show an example of a six dimensional Rayleigh distribution, which

can be used for stochastic channel modelling with two or more additional correlated model variables (for example material constant or transmission type properties). Such applications have been touched upon in literature before (see [Beard, Tekinay] for the specific details), and rely on the modeling of Rayleigh distributed random variables. The availability of higher dimension Rayleigh density approximations opens the up opportunity of more complex models, that allow for correlated channels.

The correlation coefficients are $\rho_0 = 2.5, \rho_1 = 0.3, \rho_2 = -0.1, \rho_3 = 0.1, \rho_4, = -0.15 \rho_5 = 0.2$. In this example, as mentioned before, two variables can be chosen to model Ω_t and A_t. We believe that these examples have emphazised the possible potential uses opened up by the newly gained representation.

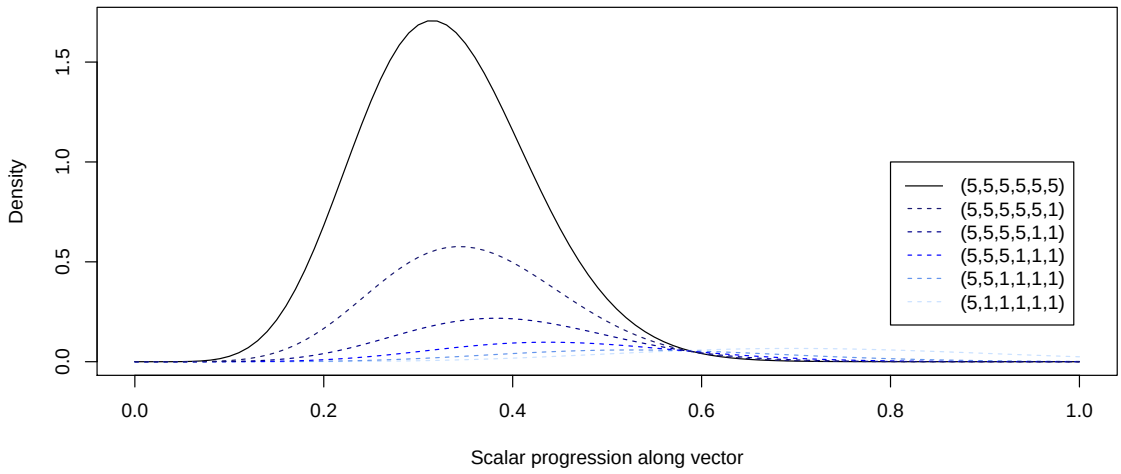


Figure 7.3: 6 dim. Rayleigh Density along 6 different directional vectors.

7.3.3 Complexity

We construct an upper limit for the evaluation effort of Algorithm 7.2, by setting the matrix A and evaluating the coefficients in $(n-1)(\frac{n}{2}-1)$ and $(n-1)n$ operations, respectively. The inner loop over the coefficients ρ runs 2^p iterations, while the outer loop has an upper bound of p^N iterations.

	Runtime		Contribution		
Highest order	Average	Total	Min	Average	Max
0	0.048	0.48	-	-	-
1	0.316	3.16	0	0	0
2	1.258	12.58	0	0	0
3	3.874	38.74	1.21E-15	1.28E-08	3.48E-08
4	10.007	100.07	0	9.64E-11	3.32E-10
5	22.865	228.65	0	2.22E-13	9.74E-13
6	47.234	472.34	0	5.55E-17	2.78E-16
7	95.969	959.69	0	2.78E-18	2.78E-17
8	171.666	1716.66	0	0	0
9	297.695	2976.95	0	0	0

Table 7.1: Runtimes and summand contribution for $n = 10$ evaluation points.

This gives a total upper limit of $\mathcal{O}(p^N 2^p n^2)$ computational operations. This provides in turn a criterion for the choice of N, as a stoppage can be implemented for sufficient N so that the additional terms drop below a user-defined threshold. For four dimensions this gives us a maximum computation time increase by factor eight for increasing the maximum term count N by one, which Table 7.1 shows we do not exceed. [5] However, this does lessen the practical application of the series expansion, as the additional contribution of the Bessel function addends are negligible. Virtually without loss of accuracy we can fix the maximum term order to zero, eliminating the need for a coefficient matrix (the approximation consists only of the $b_{t,0} = I_0(a_t)$ factors now, along with the lead coefficient $\pi^p \gamma_{n,K}$). This equates to a maximum computational effort of $\Theta(p) = \Theta(n^2)$. We have therefore used zero-order approximations exclusively in the following section.

[6]

Unfortunately we could not realise the pseudocode given by Tekinay and Beard [Beard, Tekinay (2017)] as a functioning program. The Bessel expansion used does implicitly assume only positive values, since no distinction is made with respect to the argument sign. This means that Bessel functions of negative arguments are undefined in [Beard, Tekinay (2017)]. Furthermore, we believe that some of the terms looped over in the method Beard and Tekinay put forth are not correctly specified. For example

[5] [A] Phrasing altered.

[6] [C] It may be noteworthy that in contrast to the pseudo code provided in this chapter we have vectorised our operations, as they have been carried out using *R*, which does not perform well using loops. The Bessel functions have been evaluated with the `besselI` function, included in the base package. The evaluation algorithm itself is not immediately accessible, but goes back to original algorithms formulated by David J. Sookne (1973).

in algorithm 1 line 4 introduces a triple sum, with addends not containing the sum variables. This makes the loop over these terms somewhat redundant. [7]

7.4 Simulation

It is known that Rayleigh distributed random variables can be simulated through Gaussian distributed random variables. We are therefore able to simulate a Rayleigh distributed random sample (similarly to the ansatz of our approximation) with $X \sim \mathcal{N}^n(0, S)$ and $Y \sim \mathcal{N}^n(0, S)$ where $S \in \mathbb{R}^{n \times n}$. The covariance matrix is equivalent to the Matrix K defined in Section 7.2, with the zero value diagonals removed.

We generate a random sample of normal RVs and convert them to Rayleigh random variables. The resulting samples can then be used to form empirical density functions via an n-dimensional grid of cubes, containing different amounts of the sample. We have assumed the three dimensional case to lower the computational effort (which grows to the power of the dimension) from the channel modelling example in Section 7.3 We generated 10^9 samples for each vector for optimal results, e.g. 3 Billion three dimensional samples altogether, with an adaptive grid mesh ranging between 0.01 and 0.001, depending on the local sample count. Finally, we use the resulting empirical density function for the Rayleigh density to verify the series expansion that has been derived.

[7] [A] Phrasing has been altered.

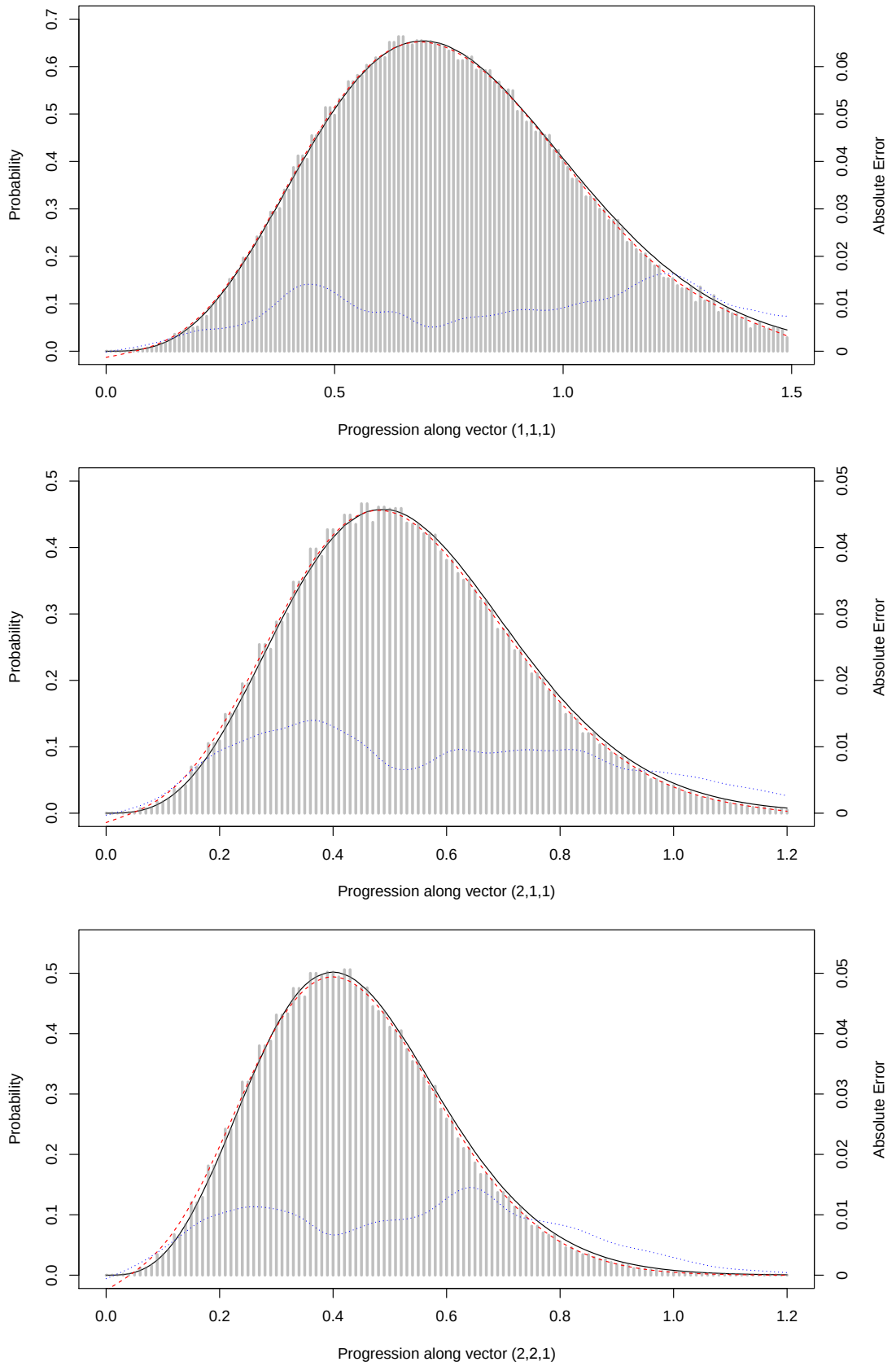


Figure 7.4: Histogram of simulated data with kernel estimator and approximation of order zero.

Additionally we applied a kernel density estimator to the epdf to compensate for random peaks within the sample (dotted red). The plots in Figure 7.4 show the results along different vector directions, giving the approximation (solid black) along with both density estimation and epdf cell values (grey bars), as well as the interpolated absolute deviation between epdf and Expression (7.10) (dotted blue).

We can clearly see that the shape of the approximation holds up, and the probabilities of the empirical, kernel density estimation and series approximation show only marginal deviations from one another. This gives strong indication for the correctness of the approximation.

7.5 Conclusions

We have considered the computation of the pdf of a Rayleigh distribution with arbitrary dimensions. By revising the algorithm to determine the non-zero series contributions of the complex integral addends, we were able to compute the series' terms more efficiently. With expression 7.10 we have therefore devised a new, more general series representation for now arbitrary dimensions.

Besides the covariance matrix assumptions about uncorrelated in-phase and quadrature parts [Beard, Tekinay (2017)] no further restrictions have to be considered. The provided pseudocode gives a blueprint on how to implement computations for practical applications.

Chapter 8

Series Approximations for Rayleigh Distributions of Arbitrary Dimensions and Covariance Matrices

Chapter Abstract

The multivariate Rayleigh distribution is of crucial importance to many applied problems of engineering, such as in the analysis of multi-antenna wireless systems. Due to the lack of a generalised closed-form of the distribution, the dependence on effective approximation methods for evaluation has created numerous numerical approaches with considerable restrictions in both dimensionality, as well as the structure of covariance matrices. In this chapter we extend a previously introduced method [Wiegand, Nadarajah (2018)] without either of these limitations. We then compare the performance of the new algorithms to recent integration methods of fixed dimension, presented by Beaulieu and Zhang [Beaulieu, Zhang (2017)] and highlight the advantages of the new method.

8.1 Introduction

The Rayleigh distribution is essential to various applications in the field of signal processing, being the most fundamental model to describe signal fading in wireless systems. Common applications include the computation of outage probabilities, as evidenced in much of the literature dedicated to the subject [Beaulieu, Zhang (2017)] [Beard, Tekinay (2017)] [Chen, Tellambura (2005)], or the interference of elementary waves [Beckmann (1964)]. One of the most frequent utilisation of multivariate Rayleigh distributions (along with Rician or Nagakami distributions) lies in the modelling and analysis of multi-antenna wireless communications systems, where the dimension of the distribution corresponds to the number of channels or antennae in the system [Beaulieu, Hemachandra (2010)]. A Rayleigh approximation without dimensionality constraints does therefore allow for systems of arbitrary size.

So far no closed-form density function for Rayleigh distributed random variables has been put forth, apart from the univariate case. Therefore fast and reliable numerical approximations are necessary to evaluate various Rayleigh-based models. Over the years many such approximations have been proposed, utilising series representations, integration methods or a combination of both. Current numerical evaluations are defined for a fixed number of dimensions, requiring new formulations for every dimension, while others may simply not be extended to more general cases. These approaches have been used with varying success and occasionally poor computational performance. Some of the most recent advances in approximation methods have been addressed or developed by Le [Le (2015)] [Le (2018-1)] [Le (2018-2)] [Le (2016)], or N. Beaulieu and K. Hemachandra [Beaulieu, Hemachandra (2010)] among others. However, while some of these have proposed approximations that generalise the number of channels in the observed systems, correlation structures remain restrictive by postulating equal correlations [Le (2015)] [Le (2018-1)] [Le (2018-2)] or other specific structures between channels [Beaulieu, Hemachandra (2010)]. In an earlier work we introduced a generalised series representation, which can be used for distributions of arbitrary dimensions [Wiegand, Nadarajah (2018)]. However, based on the original method, we introduced a similar restriction on the structure of the covariance matrix, requiring identical values on the minor diagonals. We denote the k-th minor diagonal as $a_{j,i}$ with $|j - i| = k$. In this chapter we extend [Wiegand, Nadarajah (2018)] by introducing arbitrary covariance matrices to the base approach.

This chapter is structured as follows: After the introduction where we reviewed previous methods of approximation and their applications, we move on to Section 8.2 discussing the generalisation of [Wiegand, Nadarajah (2018)] to arbitrary covariance matrices. In Section 8.3, we investigate the performance of our newly proposed series

approximation and compare the results to one of the most recently proposed integration approximations [Beaulieu, Zhang (2017)]. To validate our claims we test all methods for a constructed and an arbitrary covariance matrix of both low and high correlations, all in a three and a four dimensional space. We investigate the convergence speed of the series expansion against the accompanying increase in computational effort. Section 8.4 offers insight into possible applications of the series representation by computing outage probabilities of multi-channel systems. Lastly, we conclude this chapter by summarising our new findings and propose further research projects in Section 8.5.

8.2 Approximation Method

Virtually every approximation method exploits the Rayleigh distribution's relation to the length (L^2 norm) of a normally distributed, multivariate random vector. This approach has been used for decades to derive fixed dimension densities [Rice (1944)] [Miller (1969)], by setting up an initial expression from which the Rayleigh distribution may be derived as a marginal. We construct an n-dimensional representation of the Rayleigh distribution, by introducing $2n$ zero-mean Gaussian random variables $X = X_1, \ldots, X_n$ and $Y = Y_1, \ldots, Y_n$ with their respective variances $\sigma_1^2, \ldots, \sigma_n^2$. The variables X_i and X_j as well as Y_i and Y_j are correlated by $\rho_{i,j}$ for $i < j$, whereas the vectors X and Y remain uncorrelated to each other. In this setup, the random variable X denotes the in-phase and Y the quadrature part of a given signal. We hence start out with the joint Gaussian distribution in 8.1 for a given sample $X = (x_1, \ldots, x_n)$ and $Y = (y_1, \ldots, y_n)$:

$$f\left(x_1, \ldots, x_n, y_1, \ldots, y_n\right) = \frac{1}{(2\pi)^n |\Sigma|^{1/2}} \exp\left[-\frac{(x,y)^T \Sigma^{-1} (x,y)}{2}\right]. \tag{8.1}$$

We denote the combined covariance matrix for (X, Y) by Σ in Equation 8.2. [1]

This stands in contrast to the previous approaches [Wiegand, Nadarajah (2018)] and [Beard, Tekinay (2017)], where the covariance matrix was limited to fixed values on the main and minor diagonals, such that $\rho_{i,j} = \rho_{i+k,j+k}$ remains the same for all $k \leq n - \max\{i, j\}$ (see [Beard, Tekinay (2017)] eq. 2).

[1] [A] Added label due to necessary formatting changes.

$$\Sigma = \begin{pmatrix} \sigma_1^2 & 0 & \sigma_1\sigma_2\rho_{1,2} & \cdots & \cdots & 0 & \sigma_1\sigma_n\rho_{1,n} & 0 \\ 0 & \sigma_1^2 & 0 & \sigma_1\sigma_2\rho_{1,2} & & & 0 & \sigma_1\sigma_n\rho_{1,n} \\ \sigma_1\sigma_2\rho_{1,2} & 0 & \sigma_2^2 & \ddots & & & & 0 \\ \vdots & \sigma_1\sigma_2\rho_{1,2} & \ddots & \ddots & & & & \vdots \\ & & & & & & & \\ \vdots & & & & & & & \\ 0 & & & & & \ddots & \ddots & \vdots \\ \sigma_1\sigma_n\rho_{1,n} & 0 & & & & \ddots & \sigma_n & 0 \\ 0 & \sigma_1\sigma_n\rho_{1,n} & 0 & \cdots & & \cdots & 0 & \sigma_n \end{pmatrix}. \quad (8.2)$$

Analogously to the approach of [Wiegand, Nadarajah (2018)] we convert the cartesian coordinates into polar coordinates as $X_i = r_i \cos\theta_i$ and $Y_i = r_i \sin\theta_i$. The determinant of the Jacobian matrix of the transformation, $|J| = r_1 \cdots r_n$, appears as a factor in the density, and will be incorporated into the scalar factor later on. More importantly we use the identity $K^{-1} = 1/|\Sigma|C^T$, where $C \in \mathbb{R}^{2n}$ is the cofactor matrix corresponding to Σ. The cofactor matrix retains the same sparse shape of the original matrix, as we can see in 8.3:

$$C = \begin{pmatrix} c_{1,1} & 0 & c_{1,2} & 0 & \cdots & & \cdots & c_n & 0 \\ 0 & c_{1,1} & 0 & & & & & & c_n \\ c_{1,2} & 0 & c_{2,2} & \ddots & & & & & \vdots \\ 0 & & \ddots & \ddots & & & & & \\ \vdots & & & & & & & & \\ & & & & & & & & \\ \vdots & & & & & & \ddots & \ddots & \vdots \\ c_{1,n} & & & & & & \ddots & c_{n,n} & 0 \\ 0 & c_{1,n} & \cdots & & & & \cdots & 0 & c_{n,n} \end{pmatrix}. \quad (8.3)$$

This stands in contrast to the previous restrictions of [Wiegand, Nadarajah (2018)]

and [Beard, Tekinay (2017)] we discussed earlier, which demanded $c_{i,j} = c_{i+k,j+k}$ for all $k \leq n - \max\{i, j\}$ and which may now be ignored. The familiar formulation of 8.5 is obtained by performing the matrix-vector multiplication and the trigonometric identities $\sin(x)^2 + \cos(x)^2 = 1$ and $\cos(a - b) = \cos(a)\cos(b) + \sin(a)\sin(b)$. In order to further clarify and streamline the notation, we denote the exponential coefficient in the product as $a_{i+j-2} = r_i r_j c_{i,j}$ with the index $t = 1, \ldots, n(n-1)/2 = p$. Accordingly, we relabel the angle components $\overline{x}_{i+j-2} = \theta_j - \theta_i$ in the same way:

$$f\left(r_1, \ldots, r_n, \theta_1, \ldots, \theta_n\right) = \underbrace{\frac{\prod_{i=1}^{n} r_i}{(2\pi)^n |K|^{1/2}} \exp\left(-\frac{1}{2|K|} \sum_{i=1}^{n} r_i^2 c_{i,i}\right)}_{=\gamma(r,c)}$$

$$\times \prod_{i<j}^{n} \exp\left[-\frac{1}{|K|} c_{i,j} r_i r_j \cos\left(\theta_j - \theta_i\right)\right] \tag{8.4}$$

$$\overset{(*)}{=} \gamma(r,c) \prod_{t=1}^{n(n-1)/2} \exp\left[-\frac{1}{|K|} a_t \cos\left(\overline{x}_t\right)\right]. \tag{8.5}$$

This constitutes in essence a conflation of multiple indices of different factor combinations, into a single variable with a single progressive index. In $(*)$ we substitute $x_{l-1} - x_{k-1} = \theta_k - \theta_l$, with $x_0 = 0$, as has been done in both [Wiegand, Nadarajah (2018)] and [Beard, Tekinay (2017)].

We can then proceed with 8.5 to derive the final density function by computing the marginal density by means of integration over the domain $[0, 2\pi]$ in Equation 8.6 to derive the final result in 8.8 after some arithmetic and rearranging (for details see [Wiegand, Nadarajah (2018)]).

$$f(r_1, \ldots, r_n) = \int_0^{2\pi} \cdots \int_0^{2\pi} f(r_1, \ldots, r_n, \theta_1, \ldots, \theta_n) d\theta_1 \ldots d_n \tag{8.6}$$

$$= \gamma(r,c) \int_0^{2\pi} \cdots \int_0^{2\pi} \prod_{t=1}^{n(n-1)/2} \exp\left[-\frac{1}{|K|} a_t \cos\left(\overline{x}_t\right)\right] d\theta_1 \ldots d_n \tag{8.7}$$

$$= (2\pi)^n / 2^p \gamma(r,c) \sum_{j_1=0}^{\infty} \cdots \sum_{j_p=0}^{j_{p-1}} \prod_{t=1}^{n(n-1)/2} b_{t,j_t^*} \sum_{\rho \in \{-1,1\}^p} \prod_{w}^{n} \delta_{\{\Sigma_{x_w}=0\}}. \tag{8.8}$$

The final formula of 8.8 makes use of the previous variables defined in the equations of 8.9 below:

$$\Sigma_{x_w} = A_{w,1} + \sum_{l=1}^{n-w} A_{l,w} + \sum_{l=1}^{w-1} A_{w-l,1+l},$$

$$b_{t,j_t^*} = \begin{cases} I_0(|a_t|) & \text{if } j_t^* = 0, \\ 2(-1)^{j_t^* \mathbb{I}_{(a_t<0)}} I_{j_t^*}(|a_t|) & \text{if } j_t^* > 0, \end{cases}$$

$$j_t^* = j_{p-t+1} - j_{p-t+2},$$

$$\alpha_t = j_t^* \rho_t,$$

$$A = \begin{bmatrix} \alpha_1 & \alpha_n & \dots & \dots & \alpha_{n(n-1)/2-2} & \alpha_{n(n-1)/2} \\ \alpha_2 & \alpha_{n+1} & & & \alpha_{n(n-1)/2-1} & 0 \\ \alpha_3 & \vdots & & & 0 & \vdots \\ \vdots & & & & \vdots & \\ & & & & & \\ & \vdots & & & & \\ \vdots & \alpha_{2n-2} & & & & \vdots \\ \alpha_{n-1} & 0 & & & \dots & 0 \end{bmatrix}. \tag{8.9}$$

The relaxation of the restrictions on the covariance matrix gives us a much larger scope in potential application areas, and the basis for a truly universal formulation of the Rayleigh distribution, without dimensional limitations. [2]

We have now assumed arbitrary cofactor matrix values $c_{i,j}$, and therefore arbitrary covariance matrix values $\Sigma_{i,ij}$. Therefore we may directly compare the newly derived series approximation to recent fixed-dimension approaches without restrictiions on correlation or covariance. The next section will assess the performance of both the series expansion and the integration-based method of Beaulieu and Zhang [Beaulieu, Zhang (2017)].

8.3 Comparison

The method developed by Beaulieu and Zhang [Beaulieu, Zhang (2017)] relies on the computation of the multivariate Rayleigh distribution by means of integrals necessary in the transition from joint Gaussian to marginal distributions ([Wiegand, Nadarajah (2018)], [Beard, Tekinay (2017)]).

[2] [C] It is reasonable to note that the computation or evaluation of the Rayleigh distribution does essentially stay the same.

	Name	Type	Function	R package	Note
Univariate	Kronrod	Quadrature	`integral`	`pracma`	
	Simpson	Quadrature	`integral`	`pracma`	
	TOMS614	Quadrature/ Adaptive	`int`	`rmutil`	Taken from the ACM algorithm collection
	Romberg	Extrapolation	`int`	`rmutil`	
	Mixed	Quadrature	`integrate`	`stats`	R interface for C++ code
Bivariate	Romberg	Extrapolation	`int2`	`rmutil`	
	Quadrature	Cubature	`quad2d`	`pracma`	R wrapper for MATLAB code
	Cubature	Cubature	`pcubature`	`cubature`	R wrapper for C code

Table 8.1: An overview of the different numerical integration methods and their implementations.

The principal idea in this proposition is to determine two integrals analytically, leaving a formulation which consists of $n-2$ integrals, where n is the dimension of the Rayleigh distribution. The remaining integrals are then to be evaluated by numerical means. The functions introduced in the table have been used with the default settings implemented by the original authors of the respective packages. As with most numerical approximation implementations, there is an option to specify the target accuracy or other input parameters, such as maximum function evaluations or quadrature points (which may in turn be controlled via the accuracy settings). While higher precision may be possible, this would naturally incur higher computational effort. We will therefore provide both evaluation time, as well as error measures, to allow for comparison and balancing of both factors. Beaulieu and Zhang [Beaulieu, Zhang (2017)] provide explicit formulations for the three and four-dimensional Rayleigh densities. We have given the functions in 8.10 below:

$$f(r_1, r_2, r_3) = \prod_{i=1}^{3} \frac{r_i}{\sigma_i^2} \exp\left\{ -\frac{1}{2|\Omega|} \left[\frac{r_i^2 \left(1 - \rho_{jk}^2\right)}{\sigma_i^2} \right] \right\} \times \int_0^{\pi} \frac{\exp L_2 \cos t_2}{\pi |\Omega|} I_0 \left(\sqrt{L_1^2 + L_3^2 + 2 L_1 L_3 \cos t_2} \right) dt_2, \tag{8.10}$$

where $i \neq j \neq k$, σ^2 denotes the variance of each Gaussian random variable, while $\rho_{l,m}$ denotes their respective correlation. The coefficients L_1, L_2 and L_3 are defined in Beaulieu and Zhang [Beaulieu, Zhang (2017)] as follows:

$$L_1 = \frac{r_2 r_3 \left(\rho_{2,3}\right) - \rho_{12}\rho_{13}}{\sigma_2\sigma_3|\Omega|}, \quad L_2 = \frac{r_1 r_3 \left(\rho_{1,3}\right) - \rho_{12}\rho_{23}}{\sigma_1\sigma_3|\Omega|}, \quad L_3 = \frac{r_1 r_2 \left(\rho_{1,2}\right) - \rho_{13}\rho_{23}}{\sigma_1\sigma_2|\Omega|}. \tag{8.11}$$

To evaluate this formulation in practice, we will have to rely on numerical integration methods which determine the value of the remaining integrals. We select methods in Table 8.1 to provide a sufficiently varied picture of performances.

Since there is no closed-form of the multivariate Rayleigh distribution available to compare the approximations against, we need to select a boundary case for exact results. In order to do so, we chose the covariance matrix to simplify the integrand in 8.6 adequately to derive a closed-form. In particular we need all entries of the cofactor matrix other than the main and the first minor diagonals to disappear. We may write this condition as $c_{i,j} = 0$ for all $|i-j| > 1$ (referred to as (**)). The series expansion then collapses to the simpler case denoted in 8.12:

$$\begin{aligned} f\left(r_1,\ldots,r_n\right) &= \gamma_{n,\Sigma,r} \int\limits_{[0,2\pi]^n} \prod_{t=1}^{p} \exp\left[a_t \cos\left(\overline{x_t}\right)\right] dx_1 \cdots dx_n \\ &\overset{(**)}{=} \gamma_{n,\Sigma,r} \int\limits_{[0,2\pi]^n} \exp\Big[a_1 \cos\left(x_1 - x_2\right) + a_n \cos\left(x_2 - x_3\right) \\ &\quad + a_{2n-3} \cos\left(x_3 - x_4\right) + \cdots + a_p \cos\left(x_{n-1} - x_n\right)\Big] dx_1 \cdots dx_n \\ &= \gamma_{n,\Sigma,r} (2\pi)^n I_0\left(a_1\right) I_0\left(a_n\right) \left(a_{2n-3}\right) \cdots I_0\left(a_p\right) \\ &= \frac{\prod_{i=1}^{n} r_i}{|\Sigma|^{1/2}} \exp\left(-\frac{1}{2|\Sigma|} \sum_{i=1}^{n} r_i^2 c_{i,i}\right) \prod_{l=1}^{n-1} I_0\left(-\frac{1}{|\Sigma|} r_l r_{l+1} c_{l,l+1}\right). \end{aligned} \tag{8.12}$$

We note that the result in 8.12 constitutes a boundary case where approximation and true solution coincide, due to the zero-value sum terms of the approximation that dissappear. This of course is because the random variable pairs $\{R_i, R_j\}$ for $k \leq |i-j|$, whose missing cofactors simplify the distribution significantly. The coefficient γ is dependent on solely the covariance matrix and radius parameters and has been defined as follows:

$$\gamma_{n,\Sigma,r} = \frac{\prod_{i=1}^{n} r_i}{(2\pi)^n |\Sigma|^{1/2}} \exp\left(-\frac{1}{2|\Sigma|} \sum_{i=1}^{n} r_i^2 c_{i,i}\right).$$

We have therefore produced a closed-form solution for specific correlation and variance values. The formulation retrieved in 8.12 will be used as the true value, to which we will compare the approximations of the integral and series representation approaches. In order to quantify the accuracy of the proposed methods we will settle on a number of familiar error measures. Across an equidistant multidimensional grid,

we will evaluate the true solution, the integration-based method for multiple numerical integration approaches, as well as the series expansion with different numbers of sum terms. We have computed the average absolute error (AAE), average relative absolute error (ARAE - in percentages), aggregated absolute error (AGAE) and the maximum absolute error (MAE):

$$
\begin{aligned}
AAE &= \frac{1}{N}\sum_{i}^{N}\left|\widehat{f}\left(x_i\right) - f\left(x_i\right)\right|, \\
ARAE &= \frac{100}{N}\sum_{i}^{N}\left|\frac{\widehat{f}\left(x_i\right) - f\left(x_i\right)}{f\left(x_i\right)}\right|, \\
AGAE &= \sum_{i}^{N}\left|\widehat{f}\left(x_i\right) - f\left(x_i\right)\right|, \\
MAE &= \max_{i}\left|\widehat{f}\left(x_i\right) - f\left(x_i\right)\right|,
\end{aligned}
$$

where $\hat{f}$ denotes the approximation for the density, and f the true solution (boundary cases) or the maximum order series expansion. The evaluation points are equidistantly distributed across an area $[0, b]^n$. The upper bound b is chosen such that the hypercube encompasses the area of interest (until $f(b) \approx 0$). In addition to the accuracy we are concerned with the evaluation and setup speed of both procedures. With setup speed we refer to the computation time needed for the algorithm to determine the non-zero sum terms of the index value combinations $j_1, \ldots, j_n$. Whether or not an index value combination leads to a non-zero sum term is independent of the covariances, and can therefore be computed once and stored for other computations of the same dimension. To make the algorithm more transparent, we have decided to include the one-time computational effort for the algorithm setup regardless. We have hence recorded both times for all approaches, to give us an impression of the computational effort involved.

8.3.1 Three-Dimensional Case

In a first step we will investigate a constructed three-dimensional example. We have chosen the variances as $\sigma_1^2 = 1.5$, $\sigma_2^2 = 2$, $\sigma_3^2 = 1.5$ and the correlations as $\rho_{1,2} = -1/\sqrt{3}$, $\rho_{2,3} = -1/\sqrt{3}$, $\rho_{1,3} = 1/3$. The corresponding cofactor matrix is depicted in

8.13:

$$C = \begin{pmatrix} 4 & 0 & 2 & 0 & 0 & 0 \\ 0 & 4 & 0 & 2 & 0 & 0 \\ 2 & 0 & 4 & 0 & 2 & 0 \\ 0 & 2 & 0 & 4 & 0 & 2 \\ 0 & 0 & 2 & 0 & 4 & 0 \\ 0 & 0 & 0 & 2 & 0 & 4 \end{pmatrix}, \tag{8.13}$$

providing us the cofactor matrix structure necessary to gain a closed-form boundary case of the Rayleigh density. We proceed to analyse how both approximations fared in the simulation. Table 8.2 lists the previously discussed metrics which characterise the performances of both methods. [3] The upper part of the table is focused on the integration-based approaches performed by different numerical methods. The second half of the table depicts the error measures for the different numbers of sum terms of the series expansion. The grid has been created to span the cube $[0, 3]^3$ and was evaluated at one million nodes ($f = 100$ nodes in each dimension).

All computations have been carried out on a personal computer, equipped with 4 Intel(R) Core(TM) i7-6700 CPU @ 3.40GHz cores, which are divided into 8 Logical Processors and with 16GB available physical memory. While both setup and evaluation of our algorithm can be easily parallelised, we ran the computations on only one processor for better comparability.

[3] [A] To improve readability of the Tables 8.2-refR2tbl7 we omitted non-significant digits.

$f =$ 100	TYPE	AAE	ARAE	AGAE	MAE	EVAL (min)	SETUP (s)
Integral	Kronrod	2.2182E-11	7.6139E-08	2.2182E-05	1.2746E-10	13.730	0
	TOMS614	2.4085E-08	7.7379E-05	2.4085E-02	1.6797E-07	20.121	0
	Simpson	2.2197E-11	7.6201E-08	2.2197E-05	1.2747E-10	49.837	0
	Mixed	2.2182E-11	7.6139E-08	2.2182E-05	1.2747E-10	2.471	0
	Romberg	2.2673E-11	7.7920E-08	2.2673E-05	1.2754E-10	24.426	0
Series	Closed	4.6112E-18	1.6406E-14	4.6112E-12	1.1102E-16	0.245	0
	Series 3	3.8398E-18	1.3969E-14	3.8398E-12	9.7145E-17	2.387	0.243
	Series 6	3.8398E-18	1.3969E-14	3.8398E-12	9.7145E-17	3.295	0.975
	Series 9	3.8398E-18	1.3969E-14	3.8398E-12	9.7145E-17	4.287	2.904
	Series 12	3.8398E-18	1.3969E-14	3.8398E-12	9.7145E-17	5.185	5.874
	Series 15	3.8398E-18	1.3969E-14	3.8398E-12	9.7145E-17	6.158	10.663
	Series 18	3.8398E-18	1.3969E-14	3.8398E-12	9.7145E-17	7.160	17.338
	Series 21	3.8398E-18	1.3969E-14	3.8398E-12	9.7145E-17	8.287	26.205

Table 8.2: Performance table for three-dimensional approximations.

We can see that most of the integration algorithms perform similarly, with a rather low average error. All four error measures depict almost the same values within a very small margin for the integration approach. Only the TOMS614 algorithm seems to perform significantly worse, all error measures being a multiple of the values we see in the other four integration methods, by as much as a factor of 1000. Compared to the series expansion however, the integration method performs worse by a wide margin, regardless of error measure or integration method ($\sim 2.2 \times 10^{-11}$ vs $3.8 \sim 10^{-18}$). With only a single sum term used in the series expansion, which we have denoted as *closed* in the table, the expansion method already underbids the computed error measures by factors of 10^7, 10^6, 10^7 and 10^6 for the AAE, ARAE, AGAE and MAE, respectively. The subsequent series approaches of higher order show only little further improvement before stagnating at around $\sim 3.810^{-18}$. The differences in performance are at this point in the order of magnitude of the machine precision, and have therefore little to say about convergence or performance of the methods. In terms of the computation time, we suspected the series representation to be faster as the evaluation algorithm itself may appear more complex yet relies on basic operations, many of which were carried out before the series evaluation, regardless of parameter values. The highest order of series expansion completes the computation in 8.2 min, whereas all the integration

methods besides the mixed C method (which completes in ~ 2.5 min) are more time consuming. Our observations are largely matching our expectations, albeit with the exception of the "mixed" algorithm of the `stats` package. However, as we noted in Table 8.1 this numerical integration approach is merely an R frontend for C code, thus has to be somewhat disqualified for the sake of consistency, as the two programming languages have very different inherent evaluation speeds and computational efficiency.

We constructed 8.13 merely to verify the validity of our approximation in a testable boundary case. However, the example we created through the specific choice of covariance matrix may not be representative for the problem as a whole, as the series expansion collapses to the true solution in this particular case. For arbitrary covariance matrices, no closed-form does exist, which is why we are looking for an accurate approximation in the first place. In [Wiegand, Nadarajah (2018)], we have derived the series representation as an exact expansion, which can be evaluated to arbitrary precision. To repeat our experiment in a more general setting, we have therefore chosen a series expansion of $n = 30$ terms as a stand-in for the true solution, as the additional contributions for higher orders dropped below the machine precision (in the order of magnitude $\sim E - 20$). We therefore define a second test example with parameters as follows: $\sigma_1^2 = 0.25$, $\sigma_2^2 = 0.1$, $\sigma^2 = 0.5$ and $\rho_{1,2} = 0.3$, $\rho_{1,3} = -0.4$, $\rho_{2,3} = 0.1$. This selection of variances and correlations does not pose a special case cofactor matrix, and can therefore be used to derive more meaningful results.

The various integration methods appear to perform roughly the same as in the previous test problem in terms of accuracy, with only minor differences, see Table 8.3. The ARAE appears a little more volatile between methods than before, and the maximum absolute error across the domain has roughly doubled its value. The overall error values however remain largely the same, with the TOMS614 method once again performing noticeably weaker than the other integration processes, with each error measure about 1000 times higher compared to the other integration approaches. The different combination of variances and correlations in this setup does not immediately favour the series approach, and initially we get less accurate estimates for comparatively few sum terms, starting out at an average error of as high as 1.18×10^{-3}. At 12 sum terms we manage to match most integration methods in almost all performance categories (except for ARAE and MAE), and with 15 terms we have surpassed the benchmark method in every respect of accuracy.

$f =$ 100	TYPE	AAE	ARAE	AGAE	MAE	EVAL (min)	SETUP (s)
Integral	Kronrod	2.01005E-11	4.6518E-08	2.0010E-05	2.4080E-10	10.035	0
	TOMS614	3.1056E-08	6.9400E-05	3.1056E-02	5.9412E-07	15.212	0
	Simpson	2.0034E-11	4.6806E-08	2.0034E-05	2.4080E-10	46.190	0
	Mixed	2.0099E-11	4.6518E-08	2.0001E-05	2.4080E-10	1.865	0
	Romberg	3.5527E-11	6.3926E-07	3.5527E-05	2.7449E-10	19.971	0
Series	Series 1	1.1895E-03	1.2463E01	1.1895E03	1.0393E-02	0.183	0
	Series 3	1.6491E-05	5.4567E-01	1.6491E01	2.0979E-04	1.783	0.187
	Series 6	1.8220E-07	1.4814E-02	1.8220E-01	4.2033E-06	2.473	0.797
	Series 9	1.5974E-09	2.6247E-04	1.5974E-03	6.5957E-08	3.166	2.039
	Series 12	1.1127E-11	3.1788E-06	1.1127E-05	8.1953E-10	3.870	4.539
	Series 15	6.1772E-14	2.7316E-08	6.1772E-08	7.8258E-12	4.628	7.817
	Series 18	2.7447E-16	1.7170E-10	2.7447E-10	5.7405E-14	5.372	13.531
	Series 21	9.8076E-19	8.0997E-13	9.8076E-13	3.3297E-16	6.019	19.721

Table 8.3: Performance table for three-dimensional approximations, for a random covariance matrix.

The computation times for the integrals range from 10 to 45 minutes for one million evaluations. We believe that this many nodes do not only offer a greater accuracy in assessing the error measures, but also a more precise picture for the computational effort. Even at sum term numbers that enable the series expansion to perform more accurately than the integration approach, we have to invest less than half the evaluation time (6 min maximum for 21 sum terms). This is of course once again with the exception of the mixed C-code variant, which admittedly may have a practical use, but cannot be used to assess the computational complexity of the approximation method itself, as a series representation in C-code would undeniably be much faster than the R counterpart (which will be part of our further project development). The highest tested order of the series expansion takes little over 6 minutes to evaluate, with an accuracy that approaches machine precision at $\sim 9.810^{-19}$. Essentially we have shown that we can surpass the computational accuracy of the integration approach, while still preserving more efficient evaluation.

The last thing we need to address is the setup time. While the integration method can be evaluated at a given point directly, we need to compute a number of things

before we carry on to determine the density at a given evaluation point. The setup time expectedly grows with increased order of the sum terms, but stays still within the bounds of less than 1 minute for the examples we have investigated. Also the growth in setup time appears to be diminishing as the number of sum terms increases, potentially suggesting a slower than linear growth relationship. Compared to the integration approach this may seem like a disadvantage for the computation of small sample sizes, as it increases the total computation time. However, the coefficients that are computed for each order of expansion are independent from the covariance matrix and any other parameters. Therefore we would have to do this computation for any given order of the series expansion only one time, regardless of covariance matrix. As a consequence we omit the setup time as an argument against the series expansion.

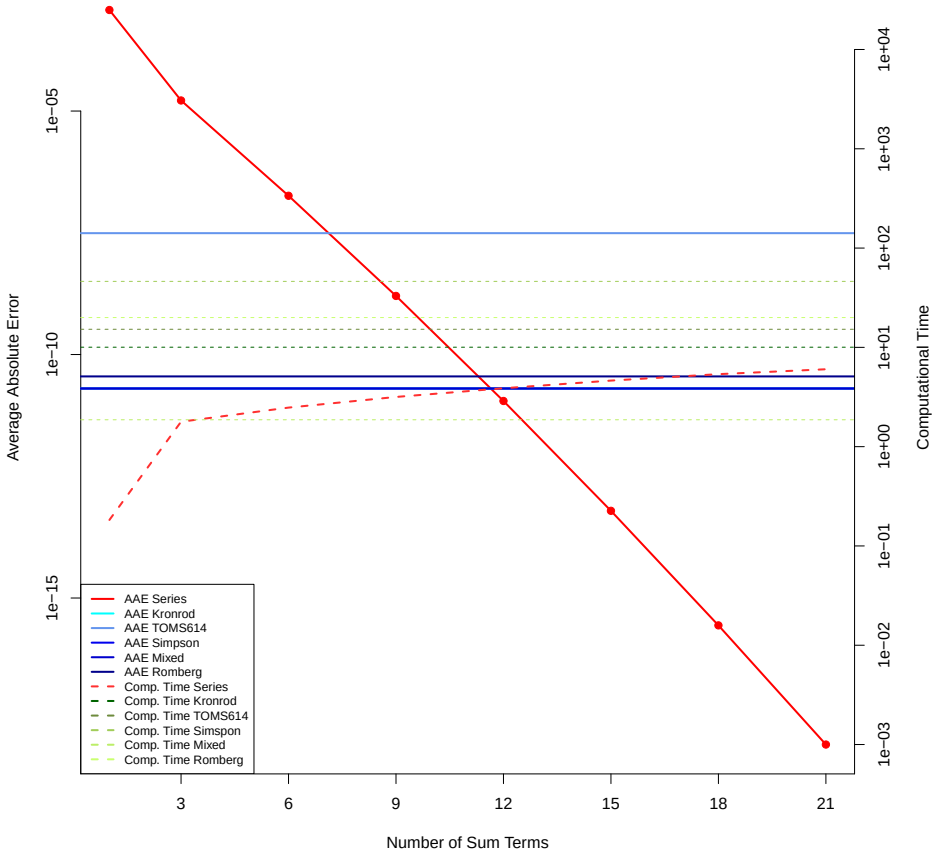


Figure 8.1: A graphical comparison between accuracy and runtimes of all tested three-dimensional approximation methods.

In Figure 8.1 we have visualised this performance disparity. The average absolute error decreases linearly in a logarithmic plot, suggesting an exponential relationship with the sum term order with a base of roughly 10. This shows the tremendous convergence speed of the series expansion, while the computational efficiency is only surpassed by the C-code implementation. However, as the main novelty of this series expansion is the lack of dimensional and covariance matrix restraints, we are eager to

know how these behaviours react in higher dimensions, and if the disparity between the methods grows with a larger random vector.

Lastly, we investigate one more set of correlations, to shed light on the impact of covariance matrices on the accuracy of the approximations. To this end we have chosen the variances as before $\sigma_1 = 0.5, \sigma_2 = 1$ and $\sigma_3 = 1.5$, the correlations in this experiment are set to $\rho_1 = 0.9$, $\rho_2 = 0.8$ and $\rho_3 = 0.7$, which covers the higher end of correlations. The results are depicted in Table 8.4 below.

$f =$ 100	TYPE	AAE	ARAE	AGAE	MAE	EVAL (min)	SETUP (s)
Integral	Kronrod	-	-	-	-	-	0
	TOMS614	1.3228E-05	9.6296E-02	1.3228E01	6.0913E-04	19.343	0
	Simpson	-	-	-	-	-	0
	Mixed	5.8994E-12	1.8596E-08	5.8994E-06	3.1360E-10	2.358	0
	Romberg	4.4575E-09	3.3304E-05	4.4575E-03	1.0839E-07	18.779	0
Series	Series 1	2.7053E-02	1.1643E02	2.7054E04	5.7033E-01	0.220	0
	Series 3	5.4629E-03	2.8822E01	5.4629E03	2.2689E-01	1.898	0.192
	Series 6	8.3076E-04	5.0763E00	8.3076E02	5.9214E-02	2.612	0.807
	Series 9	9.8804E-05	6.8282E-01	9.8804E01	1.1478E-02	3.382	2.058
	Series 12	9.4472E-06	7.3384E-02	9.4472E00	1.7043E-03	4.068	4.471
	Series 15	7.4163E-07	6.4933E-03	7.4163E-01	1.9573E-04	4.874	7.718
	Series 18	4.8602E-08	4.8316E-04	4.8602E-02	1.7883E-05	5.533	12.737
	Series 21	2.6955E-09	3.0712E-05	2.6955E-03	1.3327E-06	6.481	18.412

Table 8.4: Performance table for three-dimensional approximations, for a high correlation covariance matrix.

During the review of the results of the simulation, we quickly noticed that the Kronrod and Simpson's integration methods did not show any error measures. [4] This is due to the fact that after 24h of runtime, the methods did not manage to procure the evaluations of the 1 million observed points. We therefore omit these methods from the analysis as impractical, or unsuited for high covariance ranges. The mixed

[4][A] Published version erroneously names Romberg and Simpson, when it should be Kronrod and Simpson's integration methods

method implemented in underlying C code is more flexible, and does not change much in terms of accuracy, and only marginally increased in computation time from 1.86 min to 2.35 min.

For higher covariances the only directly comparable integration method is the Romberg method, which presents an accuracy of $\sim 4.45 \times 10^{-9}$ average absolute error and $\sim 3.3 \times 10^{-5}$ average relative error. With a computation time of ~ 19 min the evaluation did not take significantly longer, but the accuracy has visibly decreased. The TOMS614 algorithm on the other hand can only reach an average absolute error as low as 1.32×10^{-5}, with an evaluation time of 19.34 min. Similarly the series representation converges more slowly, and the highest displayed order of 21 sum terms reaches an AAE of 2.69×10^{-9}, which matches the Romberg methods accuracy. However, the evaluation time is around ~ 6.5 min, around 3 times lower than the Romberg integration methods needs for computation.

8.3.2 Four-Dimensional Case

Beaulieu and Zhang [Beaulieu, Zhang (2017)] offered a four-dimensional version of the previously stated approximation as well. We cite the four-dimensional extension directly from the source in 8.14:

$$\begin{aligned} f\left(r_1, r_2, r_3, r_4\right) =& \frac{1}{|\Omega|} \prod_{i=1}^{4} \frac{r_i}{\sigma_i^2} \exp\left(-\frac{1}{2|\Omega|} \sum_{i=1}^{4} \frac{r_i^2 \left|\Phi_i\right|}{\sigma_i^2}\right) \\ & \times \int_0^{2\pi} \int_0^{2\pi} \exp\left[L_{13}\cos\left(t_1\right) + L_{14}\cos\left(t_2\right) + L_{34}\cos\left(t_2 - t_1\right)\right] \\ & \times I_0\Bigg(\left(L_{12}^2 + L_{23}^2 + L_{24}^2 + 2L_{12}L_{23}\cos\left(t_1\right) + 2L_{12}L_{24}\cos\left(t_2\right)\right. \\ & \left. + 2L_{23}L_{24}\cos\left(t_1 - t_2\right)\right)^{1/2}\Bigg) dt_1 dt_2. \end{aligned} \tag{8.14}$$

The submatrices involved are denoted as follows:

$$\Phi_i = \begin{bmatrix} 1 & \rho_{jk} & \rho_{jl} \\ \rho_{jk} & 1 & \rho_{kl} \\ \rho_{jl} & \rho_{kl} & 1 \end{bmatrix}, \quad L_{ij} = \frac{r_i r_j}{\sigma_i \sigma_j} \begin{bmatrix} \rho_{ij} & \rho_{jk} & \rho_{jl} \\ \rho_{ik} & 1 & \rho_{kl} \\ \rho_{jl} & \rho_{kl} & 1 \end{bmatrix}, \quad \Omega = \begin{bmatrix} 1 & \rho_{12} & \rho_{13} & \rho_{14} \\ \rho_{12} & 1 & \rho_{23} & \rho_{24} \\ \rho_{13} & \rho_{23} & 1 & \rho_{34} \\ \rho_{14} & \rho_{24} & \rho_{34} & 1 \end{bmatrix}. \tag{8.15}$$

The initial four-dimensional integral can once more be evaluated for the first two variables, after which numerical methods are necessary. We are interested in the impact

of dimensionality on the accuracy of the proposed approximations and their computational effort. In this section, we will therefore assess the performance of both the integration and series expansion approaches for four-dimensional benchmark problems.

$$C = \begin{pmatrix} 0.08 & 0 & 0.04 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0.08 & 0 & 0.04 & 0 & 0 & 0 & 0 \\ 0.04 & 0 & 0.08 & 0 & 0.04 & 0 & 0 & 0 \\ 0 & 0.04 & 0 & 0.08 & 0 & 0.04 & 0 & 0 \\ 0 & 0 & 0.04 & 0 & 0.08 & 0 & 0.04 & 0 \\ 0 & 0 & 0 & 0.04 & 0 & 0.08 & 0 & 0.04 \\ 0 & 0 & 0 & 0 & 0.04 & 0 & 0.08 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0.04 & 0 & 0.08 \end{pmatrix}. \tag{8.16}$$

We begin with a test problem equivalent to the first example of the three-dimensional test section. By conveniently choosing the right parameters, we once more obtain a cofactor matrix for which a closed-form of the density function does in fact exist. We have chosen $\sigma_i^2 = 0.8$ for $i = 1, \ldots, 4$ and the covariances $\rho_{1,2} = -0.6123724$, $\rho_{1,3} = 0.4082483$, $\rho_{1,4} = -0.2500000$, $\rho_{2,3} = -0.6666667$, $\rho_{2,4} = 0.4082483$, $\rho_{3,4} = -0.6123724$. This results in the desired cofactor matrix in 8.16:

The outcome of this experiment listed in Table 8.5 is highly reminiscent of the three-dimensional case. Once more the series expansion collapses to the true closed-form, with the absolute errors somewhere in the order of magnitude of the machine precision. As we were expecting an outcome of the accuracy like this, we are using this predominantly as a benchmark to assess the computational effort of the two-dimensional integration techniques.

	$f = 15$ TYPE	AAE	ARAE	AGAE	MAE	EVAL (s)	SETUP (s)
Integral	Romberg	6.4950E-10	2.2030E-05	6.4950E-06	3.7201E-08	8401.98	0
	Quadrature	3.3087E-10	4.4827E-06	3.3087E-06	7.4404E-09	19.26	0
	Cubature	3.8458E-10	1.1552E-05	3.8458E-06	1.9018E-08	176.16	0
Series	Closed	7.4633E-19	9.1178E-15	7.4633E-15	5.5511E-17	0.19	0
	Series 3	0	0	0	0	5.84	0.78
	Series 6	0	0	0	0	22.73	7.97
	Series 9	0	0	0	0	60.40	49.82
	Series 12	0	0	0	0	126.35	3.33
	Series 15	0	0	0	0	227.83	9.79
	Series 18	0	0	0	0	377.30	25.15
	Series 21	0	0	0	0	583.28	55.37

Table 8.5: Performance table for four-dimensional approximations.

As a second example we have chosen random covariance values, to allow for a more general case. The variances were chosen as $\sigma_1^2 = 0.5$, $\sigma_2^2 = 1$, $\sigma_3^2 = 1.5$, $\sigma_4^2 = 1$, and the correlations as $\rho_{1,2} = 0.1$, $\rho_{1,3} = 0.05$, $\rho_{1,4} = -0.1$, $\rho_{2,3} = 0.025$, $\rho_{2,4} = 0.2$, $\rho_{3,4} = -0.01$. Furthermore, we have reduced the grid size to $f = 15$ in each dimension, leading to the evaluation of $15^4 = 50625$ nodes. As the additional dimension severely increased evaluation times of individual nodes, practical considerations and the fact that additional nodes did not change the error measures to any noteworthy degree led us to this decision.

The results in Table 8.6 are more revealing about the approximations' behaviours in higher dimensions. While the integration approach maintains roughly the same accuracy as in the constructed experiment, the series approach matches and surpasses the results with only 6 sum terms. This marks a considerable further intensification of the disparity of both methods performance we had observed in the three-dimensional case. Due to the numerical integration methods being increasingly computationally expensive in higher dimensions, the rift between the performances of both methods grows even further.

$f = 15$	TYPE	AAE	ARAE	AGAE	MAE	EVAL (s)	SETUP (s)
Integral	Romberg	2.5719E-10	4.2484E-06	2.5719E-06	5.2298E-09	8260.27	0
	Quadrature	2.4813E-10	3.2429E-06	2.4813E-06	5.2151E-09	17.80	0
	Cubature	3.3382E-10	9.2748E-06	3.3382E-06	1.3071E-07	108.51	0
Series	Series 1	4.8770E-05	4.7235E00	4.8770E-01	8.9364E-04	0.23	0
	Series 3	1.3468E-07	8.8945E-02	1.3468E-03	2.9143E-06	5.75	0.76
	Series 6	3.1143E-10	8.6364E-04	3.1143E-06	9.6460E-09	22.35	7.85
	Series 9	5.7774E-13	4.6229E-06	5.7774E-09	3.2583E-11	59.16	48.90
	Series 12	8.4061E-16	1.4715E-08	8.4061E-12	9.0553E-14	124.85	196.24
	Series 15	9.4898E-19	2.9655E-11	9.4898E-15	1.6458E-16	224.14	582.90
	Series 18	8.4795E-22	4.0288E-14	8.4795E-18	2.3293E-19	370.08	1469.78
	Series 21	4.4006E-25	3.5156E-17	4.4006E-21	1.6941E-21	570.20	3267.89

Table 8.6: Performance table for four-dimensional approximations.

The recorded computation times appear to confirm these observations. The evaluation of the series expansion takes only a fraction of the evaluation time the numerical integration requires. The setup time, while having been recorded for consistency, may once more be neglected. As we earlier elaborated, the necessary coefficients in the setup process have to be computed only once, regardless of covariance matrix. In the following discussion we refer to the Romberg method only, as the implementations of the cubature methods are once more merely wrappers for other, faster coding languages and can therefore not be directly compared to a pure R implementation. However, we do acknowledge a potential practical use, while keeping in mind a C or MATLAB implementation of the series representation would likely surpass the integration approach in terms of accuracy and efficiency.

At 9 sum terms we have surpassed the only intergration method implemented in R, with an average error of 5.77×10^{-13} versus 2.57×10^{-10}. Remarkably, the computation time stands at $59.15s$ against the integration methods' $8260.27s$, reducing the computational time by 140 times. We observe that at 6 sum terms and an average error of $\sim 3.1 \times 10^{-10}$ we are in the same accuracy range of the integration approaches at $\sim 2.5 - 3.3 \times 10^{-10}$, while approaching the evaluation time of the C code based quadrature evaluation at $\sim 18s$ with $\sim 22.3s$ for the series expansion.

As for the 3-dimensional examples, we test another example covariance matrix with

high correlation values in four dimensions, to assess the influence on the accuracy of the approximations. The variances are chosen as $\sigma_1 = 0.5$,$\sigma_2 = 1$, $\sigma_3 = 1.5$ and $\sigma_4 = 1$, and the correlations are $\rho_1 = 0.7$,$\rho_2 = 0.8$,$\rho_3 = 0.9$,$\rho_4 = 0.7$, $\rho_5 = 0.8$ and $\rho_6 = 0.7$.

When reviewing the results in Table 8.7, we have once more omitted the Romberg integration method due to slow convergence/ numerical instability (method failed to compute the grid points within 24h). The comparison here is therefore somewhat problematic, as the quadrature and cubature methods rely on MATLAB and C code subroutines as stated earlier. We therefore lack a direct comparison between integration and series expansion methods within the same implementation environment.

$f = 15$	TYPE	AAE	ARAE	AGAE	MAE	EVAL (s)	SETUP (s)
Integral	Romberg	-	-	-	-	NA	NA
	Quadrature	1.1718E-09	1.6790E-05	5.9325E-05	3.1585E-08	32.56	0
	Cubature	3.9786E-09	1.0148E-04	2.0143E-04	7.6008E-05	1144.98	0
Series	Series 1	3.8739E-03	5.3439E01	1.9611E02	1.3373E-01	0.73	0
	Series 3	9.6289E-04	1.6784E01	4.8747E01	3.6315E-02	58.20	1.71
	Series 6	1.6824E-04	3.7997E00	8.5173E00	7.9270E-03	184.08	22.89
	Series 9	2.3977E-05	6.9750E-01	1.2138E00	1.5658E-03	481.63	109.69
	Series 12	2.9253E-06	1.0906E-01	1.4810E-01	2.8439E-04	1026.02	472.93
	Series 15	3.1189E-07	1.4869E-02	1.5789E-02	4.7914E-05	1793.02	1425.01
	Series 18	2.9382E-08	1.7899E-03	1.4874E-03	6.7969E-06	2527.43	3748.00
	Series 21	2.4632E-09	1.9153E-04	1.2470E-04	9.2180E-07	2958.49	5274.39

Table 8.7: Performance table for four-dimensional approximations with high correlation covariance matrix.

We do notice however that the higher correlation values negatively influence the performance of all methods, both integration and series expansion based. Both quadrature and cubature integration methods lose some of their previous accuracy, increasing the AAE from $\sim 3 \times 10^{-10}$ to $1.17 \times 10^{-9}/4 \times 10^{-9}$. Meanwhile the computation time has increased for the quadrature method by 100%, whereas the cubature approach has increased tenfold.

The series approach converges significantly slower as well, as the errors seemingly decreased by $\sim 10^{-3}$ for an additional 3 sum terms. This has now been decreased to

roughly $\sim 10^{-1}$ per 3 sum terms. We note that the computation time is still below the Romberg method's computation time for the lower correlation covariance matrix example. For the hig correlation the Romberg method was not able to complete the computation within what we felt was a reasonable amount of time (< 24h). We note once more that the qudrature and cubature methods are implemented in C and MATLAB code, with only an R front end. This means the algorithms efficiency cannot be directly compared. Moving forward we will therefore implement the fairly straightforward algorithm of the series approximation in C code or a comparable language, as R itself is not suitable for practical purposes.

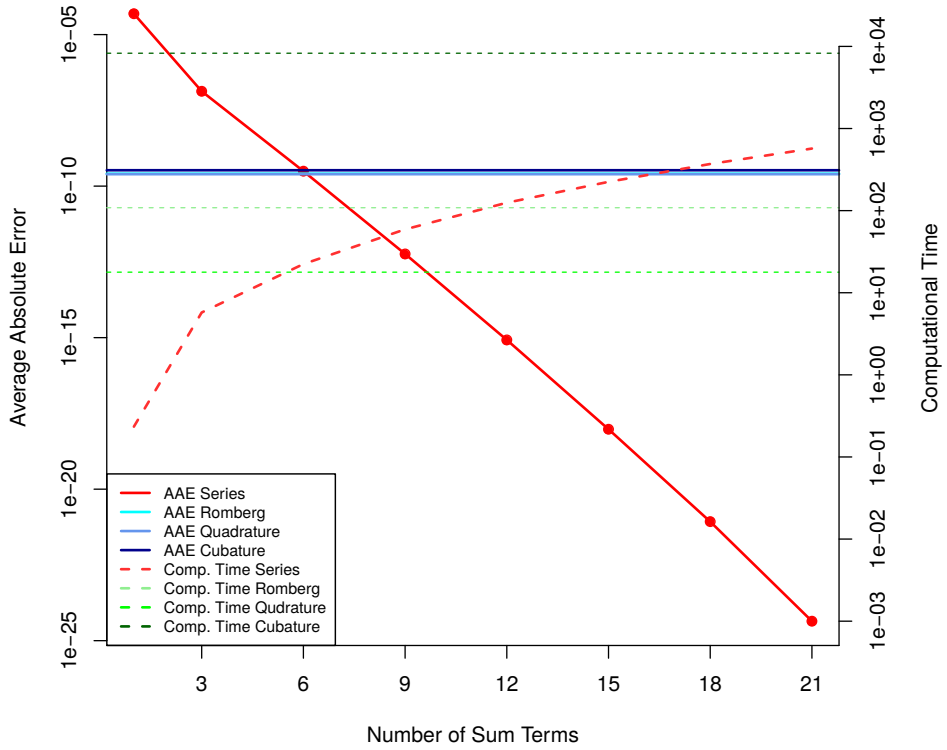


Figure 8.2: A graphical comparison between accuracy and runtimes of all tested four-dimensional approximation methods.

The performance behaviour and influence of sum terms become evident in Figure 8.2, where we see the accuracy of the series representation quickly surpassing the integration approaches, conserving a convergence in the order of $\sim 10^{-n}$, while preserving computational effort.

Paired with the more general applicability of the series presentation, its availability for arbitrary dimensions and with no restrictions on the composition of the covariance matrix (except for invertibility) the performance for three and four-dimensional cases heavily favoured the series representation. While these examples showcased the performance of the series expansion, we are predicting that this difference in performance

may become more pronounced as more channels are added. However, due to the lack of comparable algorithms in higher dimensions, we are currently unable to investigate this suspicion further. We therefore conclude that the approach introduced in this chapter is superior, and has the potential to enable new and yet untapped fields of application and research.

8.4 Applications

One of the most common applications for the Rayleigh distribution is the computation of outage probabilities. Generally this is done via the formula given in 8.17, which is the outage probability for a three-dimensional setup. We are utilising the definition noted by Chen and Tellambura [Chen, Tellambura (2005)]:

$$\begin{aligned} P_{\text{out}}(\gamma_{th}) &= \int_0^{\sqrt{\frac{\gamma_{th}\Sigma_{1,1}}{\overline{\gamma}_1}}} \int_0^{\sqrt{\frac{\gamma_{th}\Sigma_{2,2}}{\overline{\gamma}_2}}} \int_0^{\sqrt{\frac{\gamma_{th}\Sigma_{3,3}}{\overline{\gamma}_3}}} f_R(r_1, r_2, r_3)\, dr_1 dr_2 dr_3 \\ &= F_R(\gamma_{th}, \gamma_{th}, \gamma_{th}), \end{aligned} \tag{8.17}$$

where γ_{th} denotes the threshold signal to noise ratio (SNR). We demonstrate how the series representations can be used to compute the probabilities in relation to the inverse average SNR ratio $\overline{\gamma}_i$ of a single channel.

In Figures 8.3 and 8.4 we see the results for each approximation approach. The structure and setup of the outage probability graphs have been taken from Simon and Alouini [Simon, Alouini (1999)], and depict the SNR and the corresponding outage probability. The outage probability is in turn superimposed with the absolute error distribution across the SNR values, not only highlighting the overall quality of the approximation, but also the performance in relation to the input values.

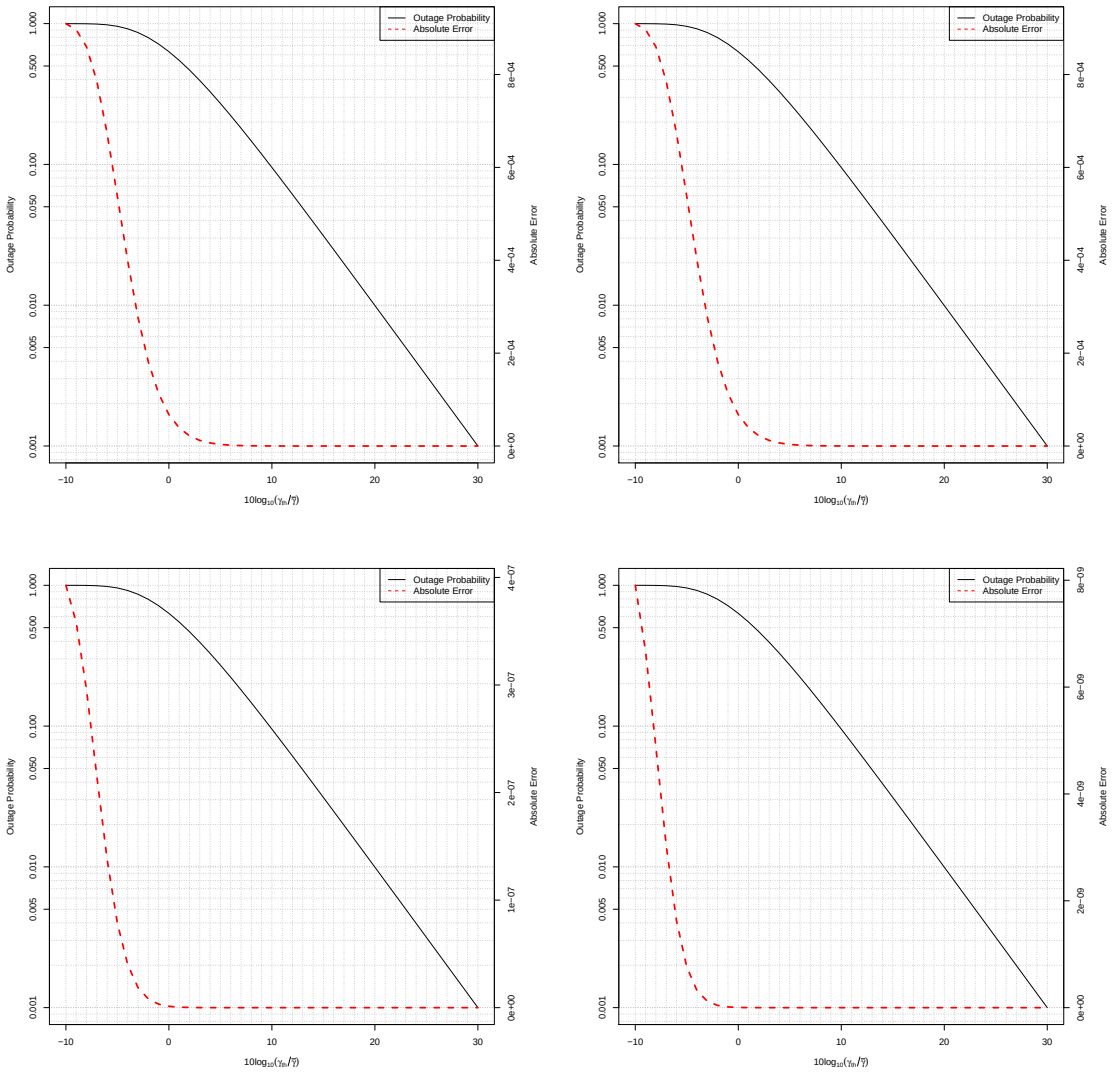


Figure 8.3: Outage probabilities versus average SNR with error distribution for all three-dimensional approximations: series expansion 3 terms (first row, left); series expansion 6 terms (first row, right); series expansion 9 terms (second row, left); series expansion 12 terms (second right, right)).

The overall shape of the error function remains similar for each approach, but we do notice that the series expansion of higher order performs significantly better than the integration approach. The slope of the error distribution changes drastically, and for $p = 15$ sum terms, the absolute error approaches zero almost instantly. Unsurprisingly the performance results are largely in agreement with the observations made in the previous section, as only a layer of numerical integration has been added in this application.

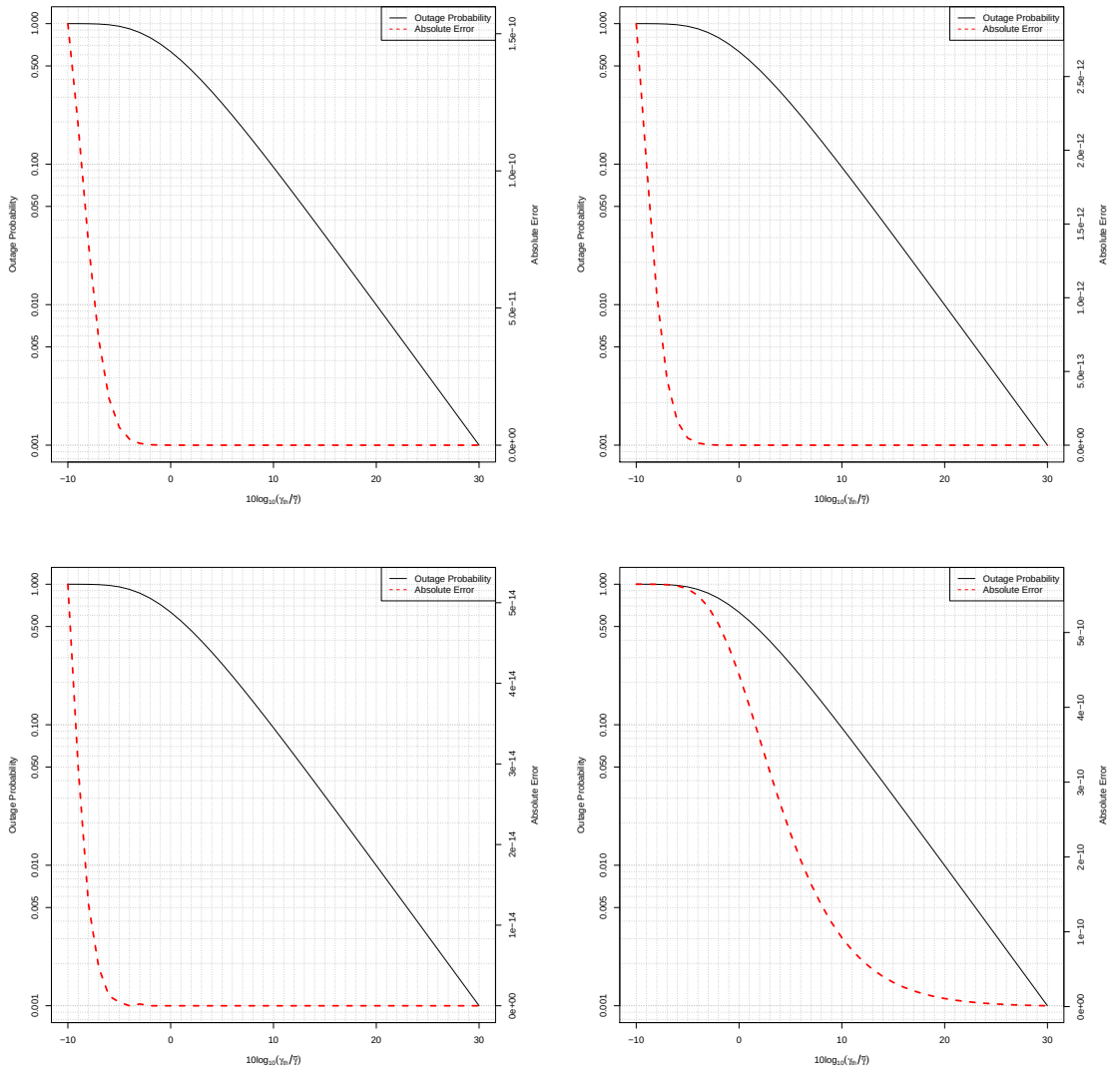


Figure 8.4: Outage probabilities versus average SNR with error distribution for all three-dimensional approximations: series expansion 15 terms (third row, left); series expansion 18 terms (third row, right); series expansion 21 terms (fourth row, left); integral based approximation (fourth row, right).

8.5 Conclusions

In this chapter, we have generalised the results of [Wiegand, Nadarajah (2018)] to retrieve a series expansion for the Rayleigh distribution for arbitrary dimension, with no restrictions on the covariance matrix values. Additionally we have investigated the approximation's performance in simulation studies of three and four dimensions with a recent integration based method. The series expansion compared favourably in both accuracy and computational effort to the proposed benchmark approach. In higher dimensions the difference in performance became more pronounced as the numerical

integration methods required added computational effort. The implementation of the series expansions and the integration methods are available in the R package `DRAYL` on CRAN [Wiegand (2018)].

In the previous section, we gave examples for applications in signal processing, by computing the outage probability of a three branch system through both [Beaulieu, Zhang (2017)]'s method and the series expansion, which matched our observations made during the performance analysis. This showed that the series approach remains stable and accurate for practical applications as well. In a next step we will extend the series expansion to include the cumulative distribution function (CDF) of Rayleigh distributions as well, eliminating the need for numerical integration. Given the frequent use of the CDF to compute outage probabilities, it would be desirable to have a more direct method of computation, instead of the numerical integration of density approximations. Additionally, a simpler version of the presented approximation may be available, as uncountable series expansions exist which may also be investigated, and provide a more accesible formula. This will also be part of upcoming projects, along with the development of a more practically suitable implementation of the existing algorithms, as our simple prototype in R leaves a lot of room for improvement and other languages and implementation techniques will be considered.

Chapter 9

MEPDF: Multivariate Empirical Density Functions

Chapter Abstract

We introduce a new R package and its functions written by the authors. This package computes an empirical density function for arbitrary dimensions with adjustable grid sizes.

9.1 Introduction

The empirical probability density function (EPDF) or more commonly referred to as histogram, is one of the simplest tools available to estimate the density of any given data set, yet remains one of the most reliable instruments for statisticians. In virtually any field of statistics the EPDF is used to verify a new representation of a density via random sampling, or to get a first attempt at the true distribution of the data at hand.

A number of different approximations for the empirical density function have existed for years [Bentley (1980)], along with efficient implementations. For example, [Duong (2017)] provided a kernel density estimator for higher dimensional data with the ks package.

Despite the widespread use of approximations, implementations were available for limited dimensions only ([Warnes et al. (2016)],[Eklund (2017)],[Sievert (2018)]). However, many problems are not limited in dimensionality, and require a flexible approach which can be used regardless of the number of variables. Kernel density estimators are often a popular choice, but due to their construction which necessitates the evaluation of a distribution for every sample point, this approach can be computationally expensive, if not infeasible for large data sets and higher dimensions. Therefore, we have developed an R package [R Development Team (2017)] which offers a density function for data sets of arbitrary dimension. The package is named MEPDF ([Wiegand, Nadarajah (2017)]).

In Section 9.2, we elaborate on the algorithm for computing the EPDF. Thereafter, we provide an overview of the functionality of the package and provide a number of examples for different possible configurations in Section 9.3. In Section 9.4, we compare our method with previously existing ones, in terms of runtimes and accuracy, and discuss benefits and disadvantages of the methods. Section 9.5 discusses three dimensional examples. We close this chapter with conclusions on what we have accomplished 9.6. [1]

9.2 Method

Let us assume we have a data set of dimension n and sample size N. The first step is to determine what the domain of the density function will look like. This means we assume an n dimensional hyperrectangle defined by a minimum and a maximum

[1] [A] Missing section labels added.

corner point:

$$R = \left\{ \mathbf{x} \in \mathcal{R} \left| x_i^{min} \leq x_i \leq x_i^{max}, \forall i = 1, \ldots, n \right. \right\}.$$

This domain is then subdivided into cells of dimensions $\mathbf{g} = (g_1, \ldots, g_n) \in \mathcal{R}^+$, so that we have a number of $\left(x_i^{max} - x_i^{min}\right)/g_i$ cells making up side i of the domain. Each of these cells can be once again defined by an upper and lower end point, p_j^{min} and p_j^{max} for all $j = 1, \ldots, \left(x_i^{max} - x_i^{min}\right)/g_i$. For any $\mathbf{x} \in \mathcal{R}^n$, we define $p^{min}(\mathbf{x})$ and $p^{max}(\mathbf{x})$ to be the corners of the the cell containing $\mathbf{x}$. If overlapping grids of different sizes are used, the cell with the smallest dimensions is given preference. With sample vectors $x_i = (x_{j1}, \ldots, x_{jn})$ for $i = j, \ldots, N$ and $\mathbf{x} = \{\mathbf{x}_1, \ldots, \mathbf{x}_N\}$ we can then define the cell counts as follows:

$$c(\mathbf{x}) = \#\left\{ \mathbf{x} | p_i^{min}(\mathbf{x}) \leq \mathbf{x}_i \leq p_i^{max}(\mathbf{x}), \forall i = 1, \ldots, N \right\}.$$

With this number we can scale for the sample size and cell dimension, and are left with the density estimator:

$$\widehat{f}(\mathbf{x}) = \frac{c(\mathbf{x})}{N \left\{ \prod_{i=1}^{n} \left[p_i^{max}(\mathbf{x}) - p_i^{min}(\mathbf{x}) \right] \right\}}. \tag{9.1}$$

Since the method is rather simplistic (essentially the multivariate generalisation of the well-known histogram), and heavily dependent on the distribution of the sample at hand, there may be grid entries for which there is no value observed (for example, no sample happens to fall within the respective grid), thus returning an estimate of $\hat{f}(x) = 0$. To offer a smoothing option to the users of this package, we introduce a technique borrowing heavily from kernel density estimators, yet modify the principle to suit our estimation approach. Essentially, we pick any given non-zero grid value, and distribute the value across adjacent cells. If $n \in \mathbb{N}$ is the dimension of the data set, p the initial estimate and $\#r \in \mathbb{N}$ the number of 'spheres' around the central grid cell, then we can express the neighbouring values as follows:

$$p_r = \frac{\frac{p}{\#r}}{(2r+1)^n - (2r-1)^n}.$$

This means, if we denote the cell value as computed in 9.1 as p, and spread this value across all cells adjacent to the initial grid cell, we derive a new value for all grid cells that provides a much 'smoother' variant with fewer zero-value cells. We may

expand this approach from direct neighbours, to neighbouring cells r cells away from the central grid cell. On a technical level, we divide the initial value p evenly across each of the r layers ($r+1$ including the central cell), and then divide $p/(r+1)$ across all cells that have the same distance to the central grid cell, thus decreasing the added 'mass' for each cell, the more rings or layers we provide. This is then repeated for all non-zero grid cells, the new results added up , and the results are saved in a new grid of equal size. This approach is often referred to as average shifted histogram, see for example [Bourel, Fraiman and Ghattas (2014)]. The function can be called via `pseudokernel`, which has the same input parameters as the standard grid method, with the optional parameter `rings`, indicating the amount of layers around each cell. This approach is akin to the averaged shifted histograms (ASH) introduced by Scott [Scott (1992)] of which implementations are available for up to two dimensions [Scott (2015)]. Once again we offer a multivariate version of a similar approach, to make the method more accessible to problems of arbitrary dimensions.

9.3 Implementation

As with the univariate case, the multivariate EPDF rests on the organization of the data domain into sections and counting the contained data sample fraction. In higher dimensions these sections become cells of the respective dimension as described in Section 9.2. Each function therefore begins with setting up a grid, based on user specifications.

```
R> data <- mvrnorm(100000, mu = c(0, 0), Sigma = diag(2))
R> density <- epdf(data = data,
+         min.corner = c(-4, -4),
+         max.corner = c(4, 4),
+         main.gridsize = c(0.05, 0.05))
R> data2 <- exp(data)
R> density2 <- epdf(data = data2,
+         min.corner = c(0, 0),
+         max.corner = c(0.25, 0.25),
+         main.gridsize = c(0.025, 0.025))
```

In the code above, we describe the general usage of a single grid EPDF. The examples given use bivariate, normally distributed and log-normally distributed data.

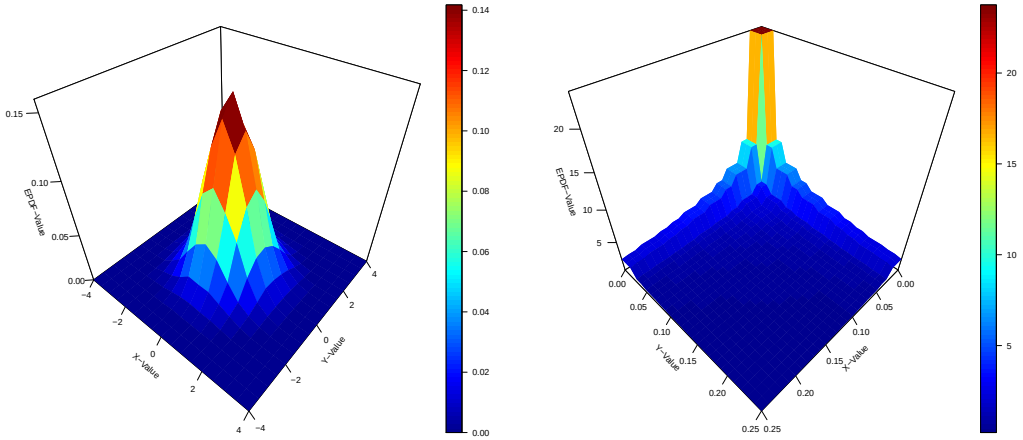


Figure 9.1: Single grid EPDFs: normal distribution (left) and log-normal distribution (right).

The code produces the EPDF density with a cropped domain between the points $(-4, -4)$ and $(4, 4)$ as well as $(0, 0)$ and $(0.25, 0.25)$. The cells are defined via the argument main.gridsize, giving the cell dimension, which are set to be squares with side lengths 0.05 and 0.025. The visualisation of the output in a three dimensional surface is shown in Figure 9.1.

For arbitrary data samples, not all regions of the plot require the same level of cell resolution. We have therefore added an option to superimpose regions of greater or lesser accuracy on top of one another. The EPDF function can therefore be called with an additional argument rescubes. This is a list of lower and upper corners of the additional grid as well as the respective grid sizes. Even though we have used cells of same side length in this example, rectangular cells are possible by specifying different side lengths.

In the code below, we describe how to call a grid with multiple resolutions on top of one another. The data set is once again a multivariate normally distributed one.

```
R> a <- list(mn = c(-1, -1),
+          mx = c(1, 1),
+          grid.size = c(0.05, 0.05))
R> b <- list(mn = c(-2, -2),
+          mx = c(2, 2),
+          grid.size = c(0.1, 0.1))
R> cubes <- list(a, b)
R>  pdf <- epdf(data = data,
+  max.corner = c(4, 4),
```

```
+   min.corner = c(-4, -4),
+   main.gridsize = c(0.2, 0.2),
+   rescubes = cubes)
```

While the main grid has the coarse resolution of 1×1, we add two more grids on top of the existing one, see Figure 9.2. The first grid stretches from $(-2, -2)$ to $(2, 2)$ with resolution 0.1×0.1 and the second from $(-1, -1)$ to $(1, 1)$ with resolution 0.05×0.05.

Notice that hyper rectangles, if added later to the argument, stand higher in the evaluation hierarchy. Thus if two additional grids overlap, the one further down in the rescubes argument will be evaluated. [2]

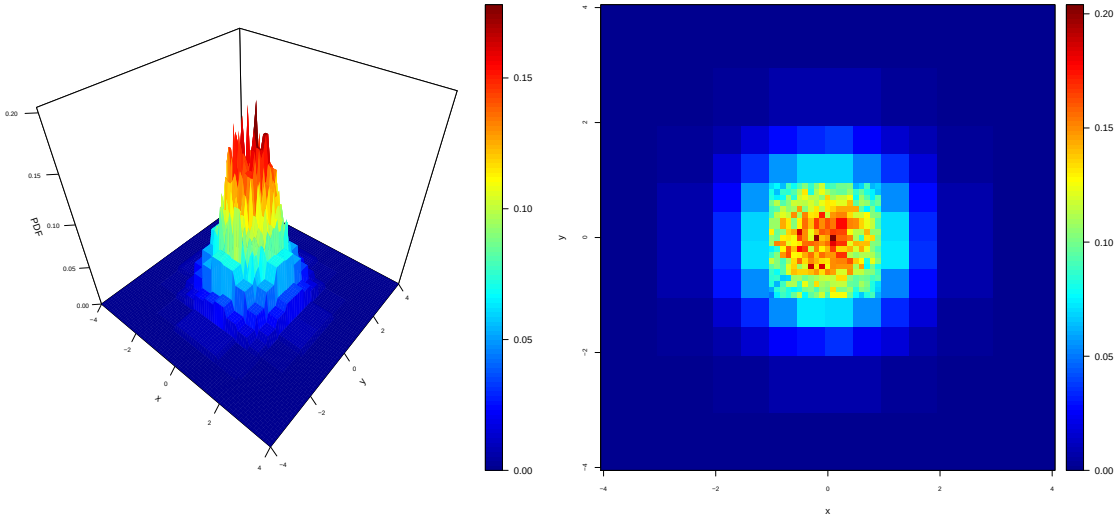


Figure 9.2: Normal distribution with two additional grids.

9.4 Comparison to other Approximations

Lee and Joe [Lee, Joe (2017)] introduced efficient sorting algorithms for bivariate and trivariate data samples, to compute the cumulative distribution function (CDF) at each sample point. We implemented the bivariate version of the modified sorting algorithm, and have tested the runtime and accuracy. We adapted the grid approach in Section 9.2 to compute the CDF, to retrieve comparable results ([van der Vaart (1998)];[Coles (2001)]; [Madsen et al. (2006)]). This was done by simply adding all

[2] [O] Typographical error removed.

cell values which are defined by coordinates lower than the cell of interest. By this successive adding of grid values, we create what is essentially a multivariate empirical CDF.

We generated random samples of a bivariate standard normal distribution for a number of different grid sizes (with 20 repetitions for each sample size, and grid size). To compare the results we have computed the approximated CDF at the sample point, both via the quicksort algorithm and the grid method. The quicksort algorithm normally refers to a sorting algorithm for data samples. The sample is successively sectioned in parts of the sample above and below the median (or mean, several other options exist as well) of subsets of the data, which ultimately results in an ordered data set. From this an empirical CDF can be easily derived. For higher dimensional data new algorithms have been developed [Lee, Joe (2017)], as well as more efficient modification of the original approach. [3] The deviation is computed via the average absolute error, as well as the error average error measured at every grid node (AGE): [4]

$$\text{AAE} = \frac{1}{n_x n_y} \sum_{i=1}^{n_x} \sum_{j=1}^{n_y} \left| f\left(x_{(i,j)}\right) - \hat{f}\left(x_{(i,j)}\right) \right|,$$

$$\text{AGE} = \frac{1}{N} \sum_{i=1}^{N} \left| f\left(x_i\right) - \hat{f}\left(x_i\right) \right|.$$

The points $\mathbf{x_i}$ for $i + 1, \ldots, N$ correspond to the sample points, whereas $\mathbf{x_{(i,j)}}$ denote the grid points between $[-2.5, -2.5]$ and $[2.5, 2.5]$ equidistantly distributed for $i = 1, \ldots, n_x$ and $j = 1, \ldots, n_y$. Due to the square area of interest and the grid shape in this case $n_x = n_y$ holds. Table 9.1 provides a collection of the outcomes. While we did anticipate the grid algorithm to be inherently more dependent on the grid size, and potentially slower than the more refined and optimized quicksort algorithm, we call to mind one of the main benefits of the grid. The set up of the grid and computation of each value takes up more time than just sorting the data frame, but the evaluation for a single point is almost instant once the matrix of values has been set up. To make the package more user-friendly we have added Scott's optimal bandwidth approximation [Scott (1992)] as the default grid size rule:

$$h_k^* = 2 \cdot 3^{1/\left(2+n\pi^{n/(4+2n)}\widehat{\sigma}_k N^{-1/(2+n)}\right)}. \tag{9.2}$$

[3] [A] A more detailed explanation of what the quicksort algorithm is has been added, which hadn't been included in the published version.

[4] [A] The abbreviations of the errors have been changed to 'AAE' and 'AGE' to be consistent with the other chapters.

For data of dimension n and sample size N as well as the empirical variance of the kth variable $\widehat{\sigma_k}^2$, the optimal grid size along the kth dimension can be described as in (9.2). Both Scott and Silverman's rule of thumb bandwidth matrices can be utilised by specifying the argument `rule = 'silverman'` or `'scott'` in the function `ekde`.

Contrary to our anticipation, the grid method we have proposed performs mostly on par with, or better than the quicksort estimation. This is heavily dependent on the chosen gridsize, as large cells might squander the amount of information provided by the sample, yet small cells might distort the true function we are estimating. Thus the results for the grid method are better than the quicksort algorithms performance, yet only when choosing the adequate cell size for the sample size at hand.

	Sample size	**Grid size**	**Setup time**	**Ev. time**	**AAE**	**AGE**
Grid method	100	0.2	0.0000	0.0000	0.05494155	0.01363199
		0.1	0.0000	0.0000	0.02559191	0.01062833
		0.05	0.0000	0.0000	0.01630517	0.00814666
		0.01	0.0700	0.0100	0.03091841	0.01841441
	1000	0.2	0.0400	0.0300	0.02400647	0.00923922
		0.1	0.0300	0.0300	0.02103825	0.00587147
		0.05	0.0400	0.0300	0.01104718	0.00489833
		0.01			0.01021250	0.00250230
	10000	0.2	0.3300	0.3000	0.02837931	0.01012616
		0.1	0.4120	0.3460	0.01358187	0.00587147
		0.05	0.3710	0.3060	0.00677197	0.00312645
		0.01	0.5340	0.6080	0.00245682	0.00119964
	100000	0.2	3.4210	2.9190	0.02872715	0.00974854
		0.1	3.4110	3.1930	0.01381017	0.00511228
		0.05	3.5360	3.1770	0.00868906	0.00312839
		0.01	4.5940	3.6480	0.00146289	0.00093547
			Computation time			
QS method	100	-	0.0197		0.04008379	-
	1000	-	0.0686		0.01158748	-
	10000	-	1.1554		0.00606940	-
	100000	-	43.3694		0.00488396	-

Table 9.1: Runtime and error measure comparison for different methods and sample sizes.

The difference in computation time between the quicksort algorithm and the grid method seems to be marginal for most of the smaller sample sizes, as both computational efforts seem to increase linearly. However, for the highest sample size of $N = 100000$ the grid method performs very clearly faster, leading us to believe that the computational effort for the quicksort method exceeds linearity.

The choice of cell size, or grid granularity is essential to the performance of the estimation approach we present. Naturally, a finer grid, with small cells provides a more accurate result for the immediate data points around it, yet may lead to more zero-value grids that do not contain any samples. Hence the grid sizes has to be balanced with the sample size available. In Table 9.1 we can see for the sample size $N = 100$ what happens if the grid is chosen too fine, for too few samples. While the AGE decreases for smaller grid sizes, it ultimately increases again for the smallest cell length of 0.01, as there aren't sufficiently many samples available.

Another observation is that the average grid error is considerably lower than the average error at the sample points. We attribute this to the fact that the normal distribution does not change much outside the 3σ area around the mean value. Therefore, the CDF values will not change much from 0 or 1, depending on which corner of the observed area we are on. More importantly, the grid gives opportunity to evaluate the CDF for arbitrary values, whereas sorting algorithms by nature can only make a statement on values at the individual sample points. The grid can therefore to some degree be used to interpolate between values, or to extrapolate beyond the samples, depending on the choice of grid sizes.

As the grid has to be set up only once, and the evaluation at arbitrary points is sufficiently fast, we believe this algorithm still has its place as benchmark or initial estimator when analyzing the distribution of data samples.

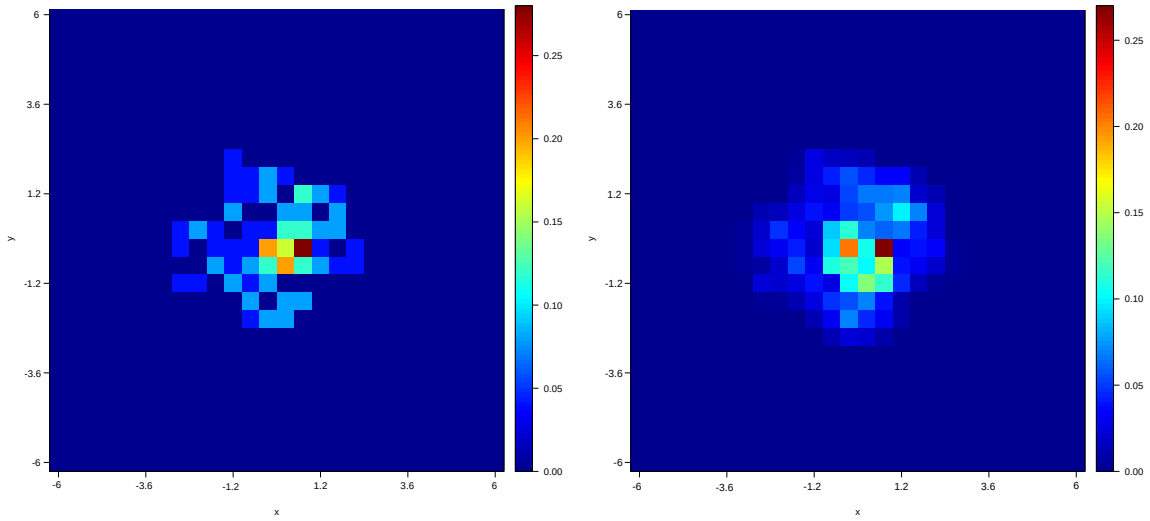


Figure 9.3: A two dimensional comparison between the standard, and pseudo-kernel options.

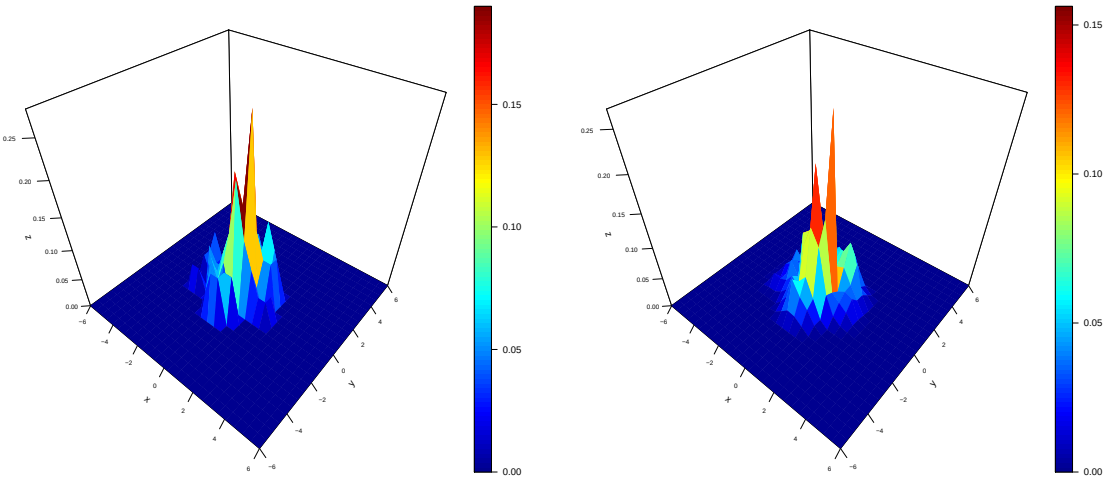


Figure 9.4: A three dimensional comparison between the standard, and pseudo-kernel options.

Lastly, we highlight the differences between the standard grid method, and the pseudo-kernel estimator we provide in this package. Due to the smoothing properties of this approach, the function can be particularly helpful for small sample sizes, which would otherwise give an incomplete picture of the underlying distribution. To visualise the differences between the grid method with and without the additional distribution of the cell values, we have provided a standard example with the two dimensional normal distribution of the comparatively low sample size $N = 100$. In Figure 9.3, we see a two dimensional colour plot of the results for both estimators side-by-side, as

well as a three dimensional version of the same results in Figure 9.4. The functions are a simple result of calling the `pseudokernel` package:

```
R> data <- mvrnorm(n = 100, mean = rep(0,2), sigma = diag(2))
R> est <- pseudokernel(data = data, mn = c(-6,-6),mx = c(6,6),
+                      grid.sizes = c(0.5,0.5), rings = 1)
R> image2D(est$grid1)
R> image2D(est$grid2)
```

As we can clearly see, the addition of layers around the non-zero cells provides a closer approximation to the true distribution. Especially around the outer edges, we get a smooth gradient instead of the previous abrupt decline to zero values in the standard grid. Furthermore, we can detect zero-value grid points in the two dimensional plots, where the sample at hand did not happen to have any observations. This quite obviously stands in contrast to the real distribution, thus mandating a corrective measure. With a single additional distribution layer, these gaps can be corrected as we see in the second picture of Figure 9.3.

We close this section by comparing our newly proposed approach to the kernel density estimator (KDE). The evaluation of n kernels every time the estimator is called was what initially made us consider a simpler, more robust approach for vastly higher sample sizes.

In Table 9.2 we can see the performance of both the KDE and the proposed histogram approach on a simple standard normal distribution in two dimensions, from which we sample progressively larger amounts of data. The KDE is much more accurate for low sample sizes, and increases only slightly for higher sample sizes, whereas the evaluation time increases with an accelerating speed (see Figure 9.5).

	KDE		MEPDF			
Sampe Size	**Runtime (s)**	**AAE**	**Grid**	**Setup (s)**	**Eval (s)**	**AAE**
100	0.215	0.01541	0.2	0.002	0.001	0.24825
			0.1	0.005	0.001	0.97098
			0.05	0.019	0.001	4.01098
			0.01	0.664	0.001	99.851
500	4.833	0.01366	0.2	0.004	0.005	0.07236
			0.1	0.007	0.005	0.20301
			0.05	0.018	0.005	0.77506
			0.01	0.532	0.005	20.0807
1000	19.062	0.01315	0.2	0.009	0.009	0.05824
			0.1	0.008	0.01	0.12632
			0.05	0.018	0.01	0.42071
			0.01	0.701	0.008	10.0631
2000	76.391	0.01284	0.2	0.014	0.018	0.03535
			0.1	0.016	0.019	0.07478
			0.05	0.079	0.017	0.20913
			0.01	0.599	0.018	4.95486
50000	-	-	0.2	0.341	0.431	0.01193
			0.1	0.357	0.438	0.01417
			0.05	0.36	0.505	0.02685
			0.01	1.281	0.431	0.2112

Table 9.2: A performance table between the kernel density estimator and proposed approach.

While the proposed approach is initially significantly less accurate (depending on grid cell sizes), the setup and evaluation times are almost negligible compared to the KDE evaluation times. For the highest sample size of $n = 50000$ we had to interrupt the process. Figure 9.5 explains why, as the computational effort increases exponentially with the number of samples. Hence it becomes clear why we created the efficient implementation of the multivariate histogram, in a simulation problem of higher dimensions, where we can sample arbitrary amounts of data points the KDE is

simply to expensive to evaluate.

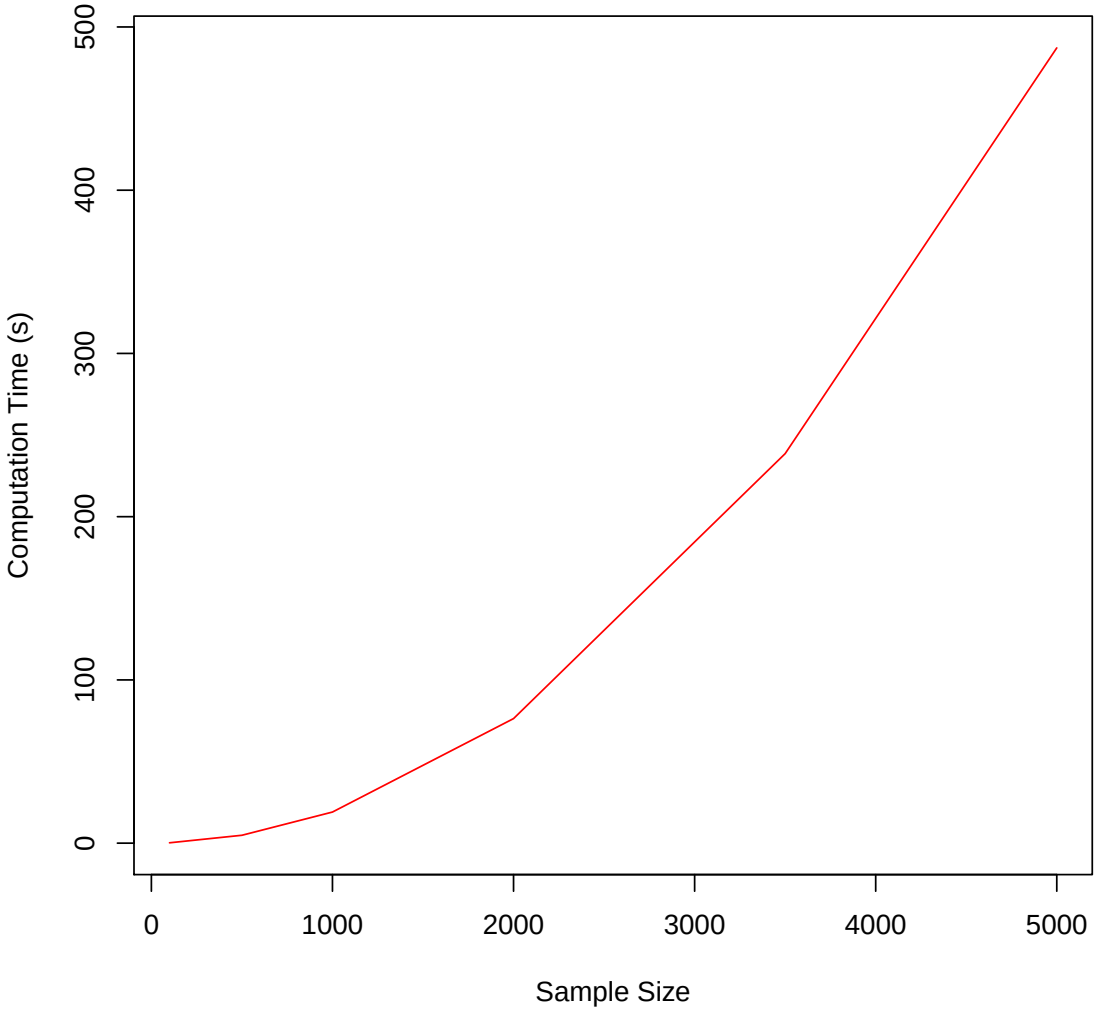


Figure 9.5: Computational effort for increased sample sizes for the kernel density estimator.

9.5 Three Dimensional Examples

To highlight the versatility of the proposed estimation, we introduce some example applications of three dimensions in this section. In Figure 9.6, we can see the empirical density for a three dimensional normal distribution on the left hand side, with a cube size of 0.2. Additionally, we like to emphasize the practical applications of the estimation technique.

Therefore we have tested the empirical density on a real life data set. In the `MASS` library, we find the `gilgais` data set, listing parameters of soil properties in New South Wales, Australia.

In Figure 9.6 on the right hand side, we have chosen the pH level of the soil in depths of 0-10cm, 30-40cm and 80-90cm. The data provides three dimensions for the empirical density example.

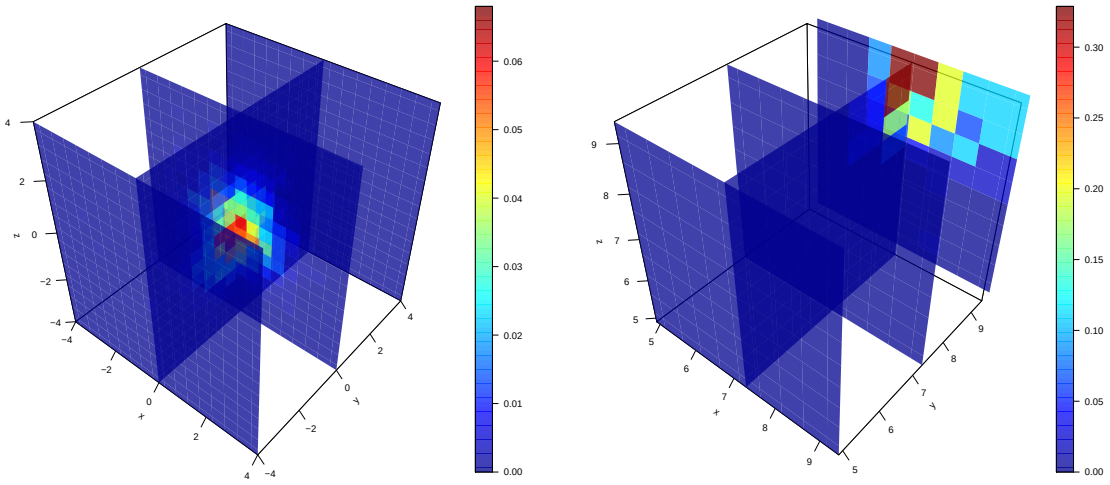


Figure 9.6: Crosssection of the three dimensional empirical PDF for a sampled three dimensional normal sample and data from the `gilgais` stock data set.

Lastly, for the three dimensional examples, we wish to give the reader an impression of the computational effort and performance behaviour of the grid method in higher dimensions. We have therefore repeated the experiment for both CDF and PDF of the three dimensional multivariate distribution in Table 9.3 [5], and have recorded the average absolute error of the estimation, along with the time required for the setup of the grid, and the evaluation for all the sample values. We see much of the same behaviour in terms of grid size and sample size, where we see the performance changing across grid sizes, if the mesh is chosen too fine. However, we can clearly see that the computation time remains not impacted by the grid size, as well as linear in relation to the input sample size.

[5] [A] Labels that times are in seconds added.

Sample size	**Grid size**	**Setup time (s)**	**Ev. time (s)**	**AAE (PDF)**	**AAE (CDF)**
1000	1	0.017	0.010	0.006853	0.142104
	0.5	0.019	0.010	0.011401	0.059951
	0.2	0.021	0.011	0.034676	0.032020
	0.1	0.020	0.009	0.038892	0.008663
10000	1	0.222	0.101	0.006039	0.134859
	0.5	0.185	0.095	0.004254	0.057359
	0.2	0.175	0.106	0.012245	0.020445
	0.1	0.182	0.089	0.032079	0.008012
100000	1	1.969	1.104	0.006011	0.130826
	0.5	1.718	0.955	0.003181	0.060196
	0.2	2.301	0.926	0.004022	0.021299
	0.1	1.901	0.932	0.010923	0.009881
1000000	1	19.640	10.801	0.005981	0.131438
	0.5	20.532	10.252	0.003036	0.059838
	0.2	19.604	10.469	0.001708	0.022123
	0.1	19.111	10.487	0.003593	0.010352

Table 9.3: Error values and computation times for the grid method in three dimensions.

For even higher dimensions, such as $n = 10$, the same approach does still work in an analogous way. However, the setup time will increase according to the dimensions. Especially with regards to the average shifted histograms, e.g. the additional spheres, or 'rings' around grid cells will be come more expensive to evaluate. The number of adjacent grid cells will increase exponentially in higher dimensions, which will delay the initial computation of the grid. However, we believe this to be acceptable, as the setup has to occur only once, and the evaluation effort will remain virtually unaffected, as it is enough to call the value stored in the correct multivariate cell.

9.6 Conclusions

With the MEPDF [Wiegand, Nadarajah (2017)] package, which can be found on the CRAN repository, we have provided an implementation for empirical density functions of arbitrary dimension. This gives a standardized tool with all necessary functions to work with higher dimensional data sets.

In Section 9.4, we have compared the grid algorithm to recent optimized sorting algorithms in terms of runtimes and accuracy. The results and wide range of applications for the grid algorithm have led us to believe that this very simple algorithm remains relevant, providing a fast and simple estimate with the right implementation.

While the quicksort algorithm serves as an example of a more sophisticated algorithm, it is by no means as versatile. Due to the nature of the quicksort algorithm, it can only give evaluations of the empirical CDF at the sample sizes, not allowing for results between the samples. Additionally, the algorithm needs readjusting, or even a completely new conceptualisation for higher dimensions. The grid approach on the other hand can easily be transferred to arbitrary dimensions, or be modified to compute CDFs instead of probabilities.

The grid method can be very easily parallelised as every cell is independent (in fact we will add the option to the package as a built-in feature), whereas the quick sort algorithm is intrinsically sequential (every new sorting decision is dependent on the previous one) and cannot be parallelised. This offers a decisive advantage to the grid method on a technical level.

Chapter 10

Conclusions

As the previous chapters have contained individual conclusions on the results obtained, this summary is focused on the implications our work has for future research.

The first thematic block was focused on the adequate choice of distribution functions for inverse CDFs and ranking functions in application areas such as linguistics, finance and natural disaster analysis. The development of a multisectioned distribution function has enabled us to more accurately capture real life data. The added complexity of the distribution models has been justified by the significant reduction in deviation from the observed data.

Inverse distribution functions and ranking functions, in particular, are of interest in population studies, which may either refer to the population numbers of cities and nations, or the study of characteristics of certain subgroups of the population or businesses among other applications. The different properties in sections of the observed data lead to a frequent classification of these sections into body, mid section and tail [Liu (2015)] [Lin et al. (2014)] [Laherrere, Sornett (1998)]. Dependent on the data and the application area, more sections may be sensible or necessary to consider. To this end, we have developed a general case of the proposed distribution family and provided an R package, which makes the models easily accessible to researchers of different academic backgrounds. With a free choice of distributions to be used for each section, both the package and the model itself can be used in a wide range of applications.

A common issue addressed in rank-size distributions is the so-called 'king-problem', e.g. the first ranks and their respective frequency within the observed data noticeably deviating from standard models, such as Zipf's law or other power laws [Jayadev (2008)]. Since this has been frequently observed in population distribution analysis, and the causes of this effect have been reasonably well explained, we believe future

modelling approaches should cover the tail behaviour more accurately. The introduction of a separate tail section, which clearly incorporates the results of the Pickands-Balkema-deHaan Lemma [Pickands (1975), Balkema, De Haan (1974)] would more fully include an understanding of tail-behaviour.

As touched upon in the introduction, we are currently working on the analysis of cough inter arrival-times in patients who suffer from respiratory diseases. Since the events (recorded coughs) are heavily clustered, the data itself breaks down into two sections, the inter-arrival times within a coughing bout and the times between bouts. This is another problem that suits the sectioned distribution rather well, as the behaviour of inter-arrival times differs drastically between inside the clusters and outside. With an adequate distribution model we are then able to model the severity of a patient's disease and can map the efficiency of novel treatments.

The second block of this manuscript encapsulates chapters 5- 9, and focuses on the advancement of distribution theory in engineering applications, particularly signal processing. Here we have further generalised statistical models extensively used in this field beyond previous limitations. While previous approximations existed to compute specific values of the log-normal characteristic functions, many of these methods focused on a specific region of importance and did not provide general robustness. We succeeded in developing several series expansions and approximation techniques which offer more robust, efficient evaluation techniques regardless of parameter ranges.

Similarly we have extended the moment formulas of the round-off errors to formulations for the n-th moments. While the previous approach already offered definitions for the first and second moments, along with the resulting variance, the derivation of a general formulation makes statistical measures of higher order such as kurtosis or skewness immediately accessible.

The results of chapters 8 and 9 are much more fundamental in their nature. Previous literature repeatedly offered approximation methods of the Rayleigh distribution with either dimensional constraints or premises on the structure of the covariance matrix. A truly general formulation has been thought infeasible or that 'a revolutionary work has to be done for higher # of RVs' [Beard, Tekinay (2017)].

We began by extending the dimensional aspect of the approximation in chapter 8, and continued in chapter 9 to release the constrictions imposed upon the construction of the covariance matrices. This has resulted in the desired non restrictive general formulation we derived. However, our work procured the density function but many

other functions are of interest in models frequently used in signal processing to describe the behaviour of multi-channel wireless systems.

The most immediate next step to extend the impact of the multivariate Rayleigh distribution approximation will be to compute the series expansion necessary to obtain the cumulative distribution function. The CDF is vital to compute the probability of outages as touched upon in chapters 8 and 9. While the computation in the previous chapters has been carried out numerically, we intend to transform the necessary integration into a series expansion as well, thus not only enabling a less computationally expensive evaluation, but also a dimensional extension beyond what is currently possible. We have tested several expansion techniques, that work well for the CDF, and make an evaluation in higher dimensions possible.

Other properties of Rayleigh-fading in multi-channel systems will be more taxing, and do require more in depth knowledge of the application area. As the specific application in multiple input-multiple output systems (MIMO) [Kumar, Singh (2013)] [Gonzalez-Aurioles (2014)] frequently does not include the correlation between channels, multivariate models that allow for arbitrary correlations could open up entirely new modelling approaches.

Other commonly used distributions such as the Rice distribution [Jayaweera, Poor (2005)] or the Nakagami distribution [Karagiannidis (2003)] have similar issues. Both distributions do not possess closed-form multivariate expansions, and the proposed approximations of the specialised literature do include the same limitations of the covariance matrices (such as constant or exponential correlations). Finding similar appropriate dimensional extensions to these distributions would have a comparable impact for other signal modelling approaches, which require a different distribution. We intend to further pursue these research topics, which represent the logical and natural continuation of the work we have presented in this thesis.

Bibliography

[Abramowitz, Stegun (1964)] M. Abramowitz, I. Stegun;
Handbook of Mathematical Functions with Formulas, Graphs, and Mathematical Tables - Chapter 25.4, Integration.
United States Department of Commerce, National Bureau of Standards (NBS) (1964).

[Akaike (1974)] H. Akaike;
A new look at the statistical model identification.
IEEE Transactions on Automatic Control. p. 716-723 (1974).

[Amancio (2015)] D. R. Amancio;
Authorship recognition via fluctuation analysis of network topology and word intermittency.
Journal of Statistical Mechanics, doi:10.1088/1742-5468/2015/03/P03005 (2015).

[Amancio et al. (2013)] D. R. Amancio, E. G. Altmann, D. Rybski, O. N. Oliveira Jr, L. F. Costa;
Probing the statistical properties of unknown texts: Application to the Voynich manuscript.
PLoS ONE, 8, e67310, doi: 10.1371/journal.pone.0067310 (2013).

[Amancio et al. (2012)] D. R. Amancio, O. N. Oliveira Jr, L. F. Costa;
Structure-semantics interplay in complex networks and its effects on the predictability of similarity in texts.
Physica A, 391, 4406-4419 (2012).

[Apostol (1974)] T. Apostol;
Mathematical analysis (2nd edition).
Addison Wesley, p. 204 (1974).

[Ardalan, Alexander (1987)] S. Ardalan, S. T. Alexander;
Fixed-point roundoff error analysis of the exponentially windowed RLS algorithm for time-varying systems.
IEEE Transactions on Acoustics, Speech and Signal Processing Vol. 35, pp. 770-783 (1987).

[Asmussen et al. (2016)] S. Asmussen, J. L. Jensen, L. Rojas-Nandayapa;
On the Laplace transform of the lognormal distribution.
Methodology and Computing in Applied Probability, vol. 18, pp. 441-458, (2016).

[Auerbach (1913)] F. Auerbach;
Das gesetz der bevolkerungskonzentration.
Petermanns Geographische Mitteilungen, 59, 74-76 (1913).

[Baayen (2002)] R. H. Baayen;
Word Frequency Distributions.
Springer Science and Business Media (2002).

[Bakar, Nadarajah (2013)] S. A. A. Bakar, S. Nadarajah;
CompLognormal: An R Package for Composite Lognormal Distributions.
The R Journal, Vol. 5/2, December (2013).

[Bakarat (1976)] R. Barakat;
Sums of independent lognormally distributed random variables.
Journal of the Optical Society of America, vol. 66, pp. 211-216 (1976).

[Balkema, De Haan (1974)] A. Balkema, L. De Haan; *Residual life time at great age.*
Annals of Probability, 2, 792-804 (1974).

[Barnes et al. (1985)] C. W. Barnes, B. Tran, S. Leung;
On the statistics of fixed-point roundoff error.
IEEE Transactions on Acoustics, Speech and Signal Processing Vol. 33, pp. 595-606 (1985).

[Beard, Tekinay (2017)] C. Beard, M. Tekinay;
A method to construct infinite series representation of quadrivariate Rayleigh distribution.
IEEE Wireless Communications Letters (2017).

[Beard, Tekinay] C. Beard, M. Tekinay;
"Stochastic analysis of temporal channel behavior using a multivariate correlated SNR distribution for reduced complexity and increased efficiency in LTE scheduling.
Under review.

[Beaulieu (2006)] N. C. Beaulieu;
A simple integral form of lognormal characteristic functions convenient for numerical computation.
Proceedings of the IEEE GLOBECOM '06, San Francisco, CA, pp. 1-3 (2006).

[Beaulieu (2008)] N. C. Beaulieu;
Fast convenient numerical computation of lognormal characteristic functions.
IEEE Transactions on Communications, vol. 56, pp. 331-333 (2008).

[Beaulieu (2010)] N. C. Beaulieu;
A power series expansion for the truncated lognormal characteristic function.
25th Biennial Symposium on Communications (2010).

[Beaulieu (2012)] N. C. Beaulieu;
An extended limit theorem for correlated lognormal sums.
IEEE Transactions on Communications, vol. 60, pp. 23-26 (2012).

[Beaulieu, Hemachandra (2010)] C. Beaulieu, K. Hemachandra;
Novel simple forms for multivariate Rayleigh and Rician distributions with generalised correlation.
IEEE Globecom (2010).

[Beaulieu, Saberali (2012)] N. C. Beaulieu, S. A. Saberali;
New approximations to the lognormal characteristic function.
Globecom 2012 - Communication Theory Symposium, pp. 2168-2172 (2012).

[Beaulieu, Zhang (2017)] N. C. Beaulieu, Y. Zhang;
New simplest exact forms for the 3-D and 4-D multivariate Rayleigh PDFs with applications to antenna array geometries.
IEEE Transactions on Communications, Vol. 65, No. 9 (2017).

[Beckmann (1964)] P. Beckmann;
Rayleigh distributions and its generalisations.
Radio Science Journal of Research, Vol. 68 D, No. 9 (1964).

[Bentley (1980)] J. Bentley;
Multidimensional divide and conquer.
Communications of the ACM 23: p. 214-229 (1980).

[Bourel, Fraiman and Ghattas (2014)] M. Bourel, R. Fraiman, B. Ghattas;
Random average shifted histograms.
Computational Statistics & Data Analysis, Vol. 79, p. 149-164. (2014)

[Boyd (1987)] J. P. Boyd;
Exponentially convergent Fourier–Chebshev quadrature schemes on bounded and

infinite intervals.
Journal of Scientific Computing, vol. 2, pp. 99-109 (1987).

[Bozdogan (1987)] H. Bozdogan;
Model selection and Akaike's Information Criterion (AIC): The general theory and its analytical extensions.
Psychometrika, 52, 345-370 (1987).

[Campiràn-Chaàve (2017)] I. Campiràn-Chaàvez, M. del Castillo-Mussot,J.A. Montemazor-Aldrete,P.Soriano-Hernàndez;
Wealth of the world's richest publicly traded companies per industry and per employee: Gamma, Log-normal and Pareto power-law as universal distributions?
Physica A (2017).

[Cancho, Elvevag (2010)] R. F. Cancho, B. Elvevag;
Random texts do not exhibit the real Zipf's law-like rank distribution.
PLoS ONE, 5, e9411, doi: 10.1371/journal.pone.0009411 (2010).

[Cancho, Sole (2001)] R. F. Cancho, R. V. Sole;
Two regimes in the frequency of words and the origins of complex lexicons: Zipf's law revisited.
Journal of Quantitative Linguistics, 8, 165 (2001).

[Cancho, Sole (2003)] R. F. Cancho, R. V. Sole;
Least effort and the origins of scaling in human language.
Proceedings of the National Academy of Sciences of the United States of America, 100, 788-791 (2003).

[Chakrabarti et al. (2013)] B.K. Chakrabarti, A. Chakrabarti, S.R. Chakravarty, A. Chatterjee;
Econophysics of Income and Wealth Distributions.
Cambridge University Press (2013).

[Chen, Tellambura (2005)] Y. Chen, C. Tellambura;
Infinite series representations of the trivariate and quadrivariate Rayleigh distribution and their applications.
IEEE Transactions on Communications, Vol. 53 (2005).

[Clauset et al. (2009)] A. Clauset, C. R. Shalizi, M. E. J. Newman;
Power-law distributions in empirical data.
SIAM Review, 51, 661 (2009).

[Clementi, Gallegati (2005)] F. Clementi, M. Gallegati;
Pareto's law of income distribution: Evidence for Germany, the United Kingdom,

and the United States.
EconWPA (2005).

[Coles (2001)] S. Coles;
An introduction to statistical modeling of extreme values.
London: Springer Verlag (2001).

[Collodi (1883)] C. Collodi;
Pinocchio.
Giornale per I bambini, first edition (1883).

[Csordas et al. (2003)] P. Csordas, A. Mersich, I. Kollar;
Digital dither: Decreasing round-off errors in digital signal processing.
in: Proceedings of the IEEE International Symposium on Intelligent Signal Processing, pp. 9-14 (2003).

[Degen, Embrechts (2008)] M. Degen, P. Embrechts;
EVT-Based Estimation of Risk Capital and Convergence of High Quantiles.
Advances in Applied Probability, Vol. 40, No. 3, p. 696-715 (2008).

[Duong (2017)] T. Duong;
ks: Kernel smoothing.
URL https://cran.r-project.org/web/packages/ks/index.html, R package version 1.10.7 (2017).

[Eklund (2017)] A. C. Eklund;
squash: Color-based plots for multivariate visualization.
R package version 1.0.8 (2017).

[Gadzhiev (2015)] Y. A. Gadzhiev;
On the variance of a centered random value roundoff error.
Signal Processing Vol. 106, pp. 30-40 (2015).

[Gerlach, Altman (2013)] M. Gerlach, E. G. Altmann;
Stochastic model for the vocabulary growth in natural languages.
Physical Review X, 3, 021006 (2013).

[Gibrat (1931)] R. Gibrat;
Les égalités économiques.
Paris, (1931).

[Gini (1921)] C. Gini;
Measurement of inequality of incomes.
Econometric Journal, 31, 124-126 (1921).

[Gonzalez-Aurioles (2014)] S. Gonzalez-Aurioles, J. L. Padilla, P. Padilla, J. F. Valenzuela-Valdes, J. C. Gonzalez-Macias;
On the MIMO Capacity for Distributed System under Composite Rayleigh/Rician Fading and Shadowing.
International Journal of Antennas and Propagation, Volume 2015, Article ID 105017 (2014).

[Gradshteyn, Ryzhik (2007)] I.S. Gradshteyn, I.M. Ryzhik;
Table of integrals, series and products (7th edition).
Elsevier Inc. (2007).

[Gubner (2006)] J. A. Gubner;
A new formula for lognormal characteristic functions.
IEEE Transactions on Vehicular Technology, vol. 55, pp. 1668-1671 (2006).

[Hanna, Quinn (1979)] E.J. Hannan, B.G. Quinn;
The determination of the order of an autoregression.
Journal of the Royal Statistical Society, B, 41, 190-19 (1979).

[Heliot et al. (2009)] F. Heliot, X. Chu, R. Hoshyar, R. Tafazolli;
A tight closed-form approximation of the log-normal fading channel capacity.
IEEE Transaction on Wireless Communications, vol. 8, pp. 2842-2847 (2009).

[Hou et al. (2009)] J. Hou, J. H. Ge and J. Li;
Trapezoidal companding scheme for peak-to-average power ratio reduction of OFDM signals.
State Key Lab of Integrated Services Networks, Xidian University (2009).

[Iserles, Norsett (2000)] A. Iserles, S. P. Norsett;
On quadrature methods for highly oscillatory functions and their implementation.
BIT, vol. 40, pp. 1-4 (2000).

[Jayadev (2008)] S. K. Jayaweera, H. V. Poor;
A power law tail in India's wealth distribution: Evidence from survey data.
Physica A: Statistical Mechanics and its Applications, Vol. 387, Iss. 1 (2008).

[Jayaweera, Poor (2005)] A. Jayadev;
On the Capacity of Multiple Antenna Systems in Rician Fading.
arXiv:cs/0501051v1, (2005).

[Karagiannidis (2003)] G. K. Karagiannidis, D. A. Zogas, S. A. Kotsopoulos;
An Efficient Approach to Multivariate Nakagami-m Distribution Using Greens Matrix Approximation.
IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS, VOL. 2, NO. 5 (2003).

[Kawarai, Murakami (1989)] S. Kawarai, T. Murakami;
An optimization procedure to minimize the roundoff noise in cascade floating-point digital filters.
Proceedings of the International Conference on Acoustics, Speech, and Signal Processing, pp. 884-887 (1998).

[Kleiber, Kotz (2003)] C. Kleiber, S. Kotz;
Statistical Size Distribution in Economics and Actuarial Sciences.
Wiley (2003).

[Kumar, Singh (2013)] M. Kumar, A. Singh;
Channel Capacity Comparison of MIMO Systems with Rician Distributions, Rayleigh Distributions and Nakagami-m.
International Journal of Engineering Research & Technology, Vol. 2 Issue 6, June (2013).

[Laherrere, Sornett (1998)] J. Laherrrere and D. Sornett;
Stretched exponential distributions in nature and economy: "fat tails" with characteristic scales.
The European Physical Journal B - Condensed Matter and Complex Systems, Vol. 2, Iss. 4 (1998).

[Lee, Joe (2017)] D. Lee, H. Joe;
Efficient computation of multivariate empirical distribution functions at the observed values.
Computational Statistics, doi: 10.1007/s00180-017-0771-x (2017).

[Lee et al. (2007)] C. Lee, F. Famoye, O. Olumolade;
Beta-Weibull distribution: Some properties and applications to censored data.
Journal of Modern Applied Statistical Methods, 6, Article 17 (2007).

[Li, Nadarajah (2016)] R. Li, S. Nadarajah;
Mean and Variance of round off error.
Signal Processing Vol. 127, pp. 185-190 (2016).

[Le (2015)] K. N. Le;
Comments on 'Distribution functions of selection combiner output in equally correlated Rayleigh, Rician, and Nakagami-m fading channels'.
IEEE Transactions on Communications, Vol. 63, No. 12, pp. 5283-5287 (2015).

[Le (2018-1)] K. N. Le;
Distributions of multiuser-MIMO under correlated generalised-Rayleigh fading.'
Signal Processing, Vol. 150, No. 9, pp. 228-232 (2018).

[Le (2018-2)] K. N. Le;
Selection combiner output distributions of multivariate equally-correlated generalised-Rician fading for any degrees of freedom.
IEEE Transactions on Vehicular Technology, Vol. 67, No. 3, pp. 2761-2765 (2018).

[Le (2016)] K. N. Le;
On selection combiner output quadrivariate cumulative distribution functions in correlated Nakagami-m and Rayleigh fading.
IEEE Communications Letters, Vol. 20, No. 9, pp. 1717-1720 (2016).

[Limpert et al. (2001)] E. Limpert, W. A. Stahel, M. Abbt;
Lognormal distributions across the sciences: Keys and clues.
BioScience, vol. 51, pp. 341-352 (2001).

[Lin et al. (2014)] R. Lin, Q. Ma, C. Bian;
Scaling laws in human speech, decreasing emergence of new words and a generalized model.
arXiv:1412.4846 (2014).

[Liu (2015)] L. Liu;
The Small Head, the Medium Body, and the Long Tail .. so, where's Microsoft?
TechNet Archive, Lawrence Liu's report from the inside (2015).

[Liu et al. (2008)] X. Liu, S. Nevo, X. Pang;
On the k-th derivative of meromorphic functions with zeros of multiplicity at least $k+1$.
Journal of Mathematical Analysis and Applications Vol. 348, pp. 516–529 (2008).

[Madsen et al. (2006)] H. Madsen, S. Krenk, S. Lind;
Methods of structural safety
New York: Dover Publications (2006).

[Mandelbrot (1955)] B. B. Mandelbrot;
An informational theory of the statistical structure of languages.
In: Communication Theory, pp. 486-502 (1955).

[Manning, Schutze (1999)] C. D. Manning, H. Schutze;
Foundations of Statistical Language Processing.
MIT Press (1999).

[Marx (1867)] K. Marx;
Das Kapital. Kritik der politischen Ökonomie
Otto Meisner, Volume I (1867).

[Masucci, Rodgers (2009)] A. P. Masucci, G. J. Rodgers;
Differences between normal and shuffled texts: Structural properties of weighted networks.
Advances in Complex Systems, 12, doi: 10.1142/S0219525909002039 (2009).

[Mehri, Jamaati (2017)] A. Mehri, M. Jamaati;
Variation of Zipf's exponent in one hundred live languages: A study of the Holy Bible translations.
Physics Letters A, 381, 2470-2477 (2017).

[Miller (1969)] K.S. Miller;
Complex Gaussian processes.
SIAM Review, Vol. 11, pp. 544-567 (1969).

[Papoulis, Pillai (2002)] A. Papoulis, S. U. Pillai;
Probability, Random Variables and Stochastic Processes.
New York: McGraw-Hill (2002).

[Pickands (1975)] J. Pickands; *Statistical inference using extreme order statistics.* Annals of Statistics, 3, 119-131 (1975).

[R Development Team (2017)] R Development Core Team;
R: A language and environment for statistical computing.
Vienna, Austria: R Foundation for Statistical Computing (2017).

[Rappaport (1996)] T. S. Rappaport;
Wireless Communications - Principles and Practice.
New Jersey: Prentice Hall (1996).

[Reig (2009)] J. Reig;
Multivariate Nakagami- distribution with constant correlation model.
AEU - International Journal of Electronics and Communications, Volume 63, Issue 1, Pages 46-51 (2009).

[Rice (1944)] S.O. Rice;
Mathematical analysis of random noise.
The Bell System Technical Journal, Vol. 23, pp. 282-332 (1944).

[Saomt-Exupery (1943)] A. de Saomt-Exupery;
Le Petit Prince.
Editions Gallimard (1947).

[Schwarz (1978)] G.E. Schwarz;
Estimating the dimension of a model.
Annals of Statistics, 6, 461-464 (1978).

[Scott (1992)] D. W. Scott;
Multivariate density estimation: Theory, practice and visualisation.
New York: John Wiley and Sons (1992).

[Scott (2015)] D. W. Scott;
ash: David Scott's ASH routines.
https://cran.r-project.org/web/packages/ash/index.html, R package version 1.0-15 (2015).

[Sievert (2018)] C. Sievert;
plotly for R.
R package version 4.8.0 (2018).

[Simon, Alouini (1999)] M. K. Simon, M. -S. Alouini;
A unified performance analysis of digital communication with dual selective combining diversity over correlated Rayleigh and Nakagami-m fading channels.
IEEE Transactions on Communications, Vol. 47, No. 1 (1999).

[Soriano-Hernández et al. (2016)] P. Soriano-Hernández, M. del Castillo-Mussot, I. Campirán-Chávez, J.A. Montemayor-Aldrete;
Wealth of the world's richest publicly traded companies per industry and per employee: Gamma, log-normal and Pareto power-law as universal distributions?
Physica A, 471, 733-749 (2016).

[Soriano-Hernández et al. (2017)] P. Soriano-Hernández, M. del Castillo-Mussot, O. Córdoba-Rodrígez, R.M. Mansilla-Corona;
Non-stationary individual and household income of poor, rich and middle classes in Mexico
Physica A, 465, 403-413 (2017).

[Tellambura, Senaratne (2010)] C. Tellambura, D. Senaratne;
Accurate computation of the MGF of the lognormal distribution and its application to sum of lognormals.
IEEE Transactions on Communications, vol. 58, pp. 1568-1577 (2010).

[Papoulis, Pillai (2001)] A. Papoulis, S. Pillai;
Probability,Random Variables and Stochastic Processes.
Tata McGraw-Hill (2002).

[Press (1969)] M. Press;
Round-off error of floating-point digital filters.
Papers on Digital Signal Processing, Vol. 1, pp. 94-102 (1969).

[University Edinburgh, 2017] [1] *The School of Informatics, University of Edinburgh.*
http://homepages.inf.ed.ac.uk/s0787820/bible (2017).

[van der Vaart (1998)] A. van der Vaart;
Asymptotic statistics..
Cambridge: Cambridge University Press (1998).

[Vladimirov, Diamond (2002)] I. G. Vladimirov, P. Diamond;
A uniform white-noise model for fixed-point roundoff errors in digital systems.
Automation and Remote Control Vol. 63, pp. 753-765 (2002).

[Warnes et al. (2016)] G. R. Warnes, B. Bolker, L. Bonebakker, R. Gentleman, W. H. A. Liaw, T. Lumley, M. Maechler, A. Magnusson, S. Moeller, M. Schwartz and B. Venables.
gplots: Various R programming tools for plotting data.
R package version 3.0.1 (2016).

[Widrow, Kollar (2008)] B. Widrow, I. Kollar;
Quantization Noise: Round-off Error in Digital Computation, Signal Processing, Control and Communications.
Cambridge University Press, New York (2008).

[Wiegand (2018)] M. Wiegand;
DRAYL: Approximations for multivariate Rayleigh density functions.
Comprehensive R Archive Network (CRAN, 2018).

[Wiegand, Nadarajah (2017)] M. Wiegand, S. Nadarajah;
MEPDF: Multivariate empirical density function.
URL https://cran.r-project.org/web/packages/MEPDF/index.html, R package version 3.0. (2017).

[1][A] The original citation only provided the webpage link. The data set itself has been provided for public access by the Computer Science/ Informatics department of the University of Edinburgh/ shared with us by the original author.

[Wiegand, Nadarajah (2018)] M. Wiegand, S. Nadarajah;
A series representation for multidimensional Rayleigh distributions.
International Journal of Communications Systems, Vol. 31, Issue 6 (2018).

[Williams et al. (2015)] J. R. Williams, J. P. Bagrow, C. M. Danforth, P. S. Dodds;
Text mixing shapes the anatomy of rank-frequency distributions.
Physical Review E, 91, 052811 (2015).

[Wong (1990)] P. W. Wong;
Quantization noise, fixed-point multiplicative roundoff noise and dithering.
IEEE Transactions on Acoustics, Speech and Signal Processing Vol. 38, pp. 286-300 (1990).

[Word Count App] [2] *Simple word count app offered on browserling.com*
https://www.browserling.com/tools/word-frequency (2017).

[WriteWords] [3] *Word Frequency Tool*
Supplied by the WriteWords Authors' community.
http://www.writewords.org.uk/wordcount.asp/ (2017).

[Yeh, Schwartz (1984)] Y. Yeh, S. C. Schwartz;
Outage probability in mobile telephony due to multiple lognormal interferers.
IEEE Transactions on Communications, vol. 32, pp. 380-388 (1984).

[Yu, Lim (2006)] Y. J. Yu, Y. C. Lim;
Roundoff noise analysis of signals represented using signed power-of-two terms.
Proceedings of the 14th European Signal Processing Conference, pp. 1-4 (2006).

[Zipf (1949)] G. K. Zipf;
Human Behavior and the Principle of Least Effort.
Addison-Wesley, Massachusetts (1949).

[2] [A] This app was used to retrieve the total word count of each translation.
[3] [A] WriteWords is an online platform for authors, which provided an online tool to list words and their occurrence.

Appendix A

Composite Model Supplementary Material

The data used in Chapter 2 has 5 columns. For each listed company we are provided with the annual profits, sales, asset value, market value and the employee count. The data used in the development of the distribution models were the 4 financial properties divided by employee count.

Model	Metric	Partial Distribution			Penalised Measures				
		Body	**Mid**	**Tail**	**AIC**	**BIC**	**AICc**	**HQC**	**CAIC**
Two part hard cut-off	Market Value	LogNorm	-	Pareto1	7416	7438	7416	7424	7442
		Gamma	-	Pareto1	7853	7876	7853	7572	7880
		Expon	-	Pareto1	10287	10303	10287	10293	10306
	Sales	LogNorm	-	Pareto1	4062	4085	4062	4070	4089
		Gamma	-	Pareto1	-2067	-2045	-2067	-2059	-2041
		Expon		Pareto1	7524	7540	7524	7530	7543
	Assets	LogNorm	-	Pareto1	9909	9932	9909	9917	9936
		Gamma	-	Pareto1	6682	6705	6682	6690	6709
		Expon	-	Pareto1	11410	11426	11410	11416	11429
	Profits	LogNorm	-	Pareto1	-4150	-4128	-4150	-4142	-4124
		Gamma	-	Pareto1	501	523	501	509	527
		Expon	-	Pareto1	2953	2970	2953	2959	2973
Three part composite model	Market Value	LogNorm	Pareto4	Pareto4	2456	2534	2456	2485	2548
		Gamma	Pareto4	Pareto4	2455	2534	2455	2484	2548
		BetaWb	Pareto4	Pareto4	2446	2536	2446	2479	2552
		LogNorm	BetaWb	Pareto4	2461	2539	2461	2490	2553
		Gamma	BetaWb	Pareto4	2467	2546	2468	2496	2560
		BetaWb	BetaWb	Pareto4	2505	2595	2506	2538	2611
	Sales	LogNorm	Pareto4	Pareto4	-9878	-9800	-9878	-9849	-9786
		Gamma	Pareto4	Pareto4	-10232	-10153	-10231	-10203	-10139
		BetaWb	Pareto4	Pareto4	-9740	-9650	-9739	-9707	-9634
		LogNorm	BetaWb	Pareto4	-9891	-9813	-9891	-9862	-9799
		Gamma	BetaWb	Pareto4	-9914	-9836	-9914	-9885	-9822
		BetaWb	BetaWb	Pareto4	-9922	-9833	-9922	-9889	-9817
	Assets	LogNorm	Pareto4	Pareto4	3813	3891	3813	3841	3905
		Gamma	Pareto4	Pareto4	3784	3862	3784	3813	3876
		BetaWb	Pareto4	Pareto4	3824	3914	3824	3857	3930
		LogNorm	BetaWb	Pareto4	3811	3890	3812	3840	3904
		Gamma	BetaWb	Pareto4	3796	3875	3797	3825	3889
		BetaWb	BetaWb	Pareto4	3802	3891	3802	3834	3907
	Profits	LogNorm	Pareto4	Pareto4	-18081	-18004	-18080	-18052	-17990
		Gamma	Pareto4	Pareto4	-18107	-18030	-18107	-18079	-18016
		BetaWb	Pareto4	Pareto4	-18032	-17944	-18032	-18000	-17928
		LogNorm	BetaWb	Pareto4	-17998	-17920	-17997	-17969	-17906
		Gamma	BetaWb	Pareto4	-17881	-17803	-17880	-17847	-17789
		BetaWb	BetaWb	Pareto4	-17831	-17743	-17831	-17799	-17727

Table A.1: Error measure compilation for composite models, loop tolerance=10^{-8} for percentage function fitting.

Appendix B

Word Frequencies Supplementary Material

The word frequency data has three columns for each language and piece of literature. It features the word, its respective number of uses in the literary work and the corresponding rank. Data has been obtained for 100 bible translations, as well as for 8 translations each for 'Le Petit Prince', 'Pinocchio' and 'Das Kapital'. Furthermore, random texts of various length have been generated by the PHP Lorem Ipsum algorithm sampler created by Mathew Tinsley (2009). This algorithm requires a training text from which structures are emulated and random texts of various length can be generated. Besides the study of word frequencies and language behaviour, algorithms such as this are also frequently used to provide filler texts for typesetting and webdesign.

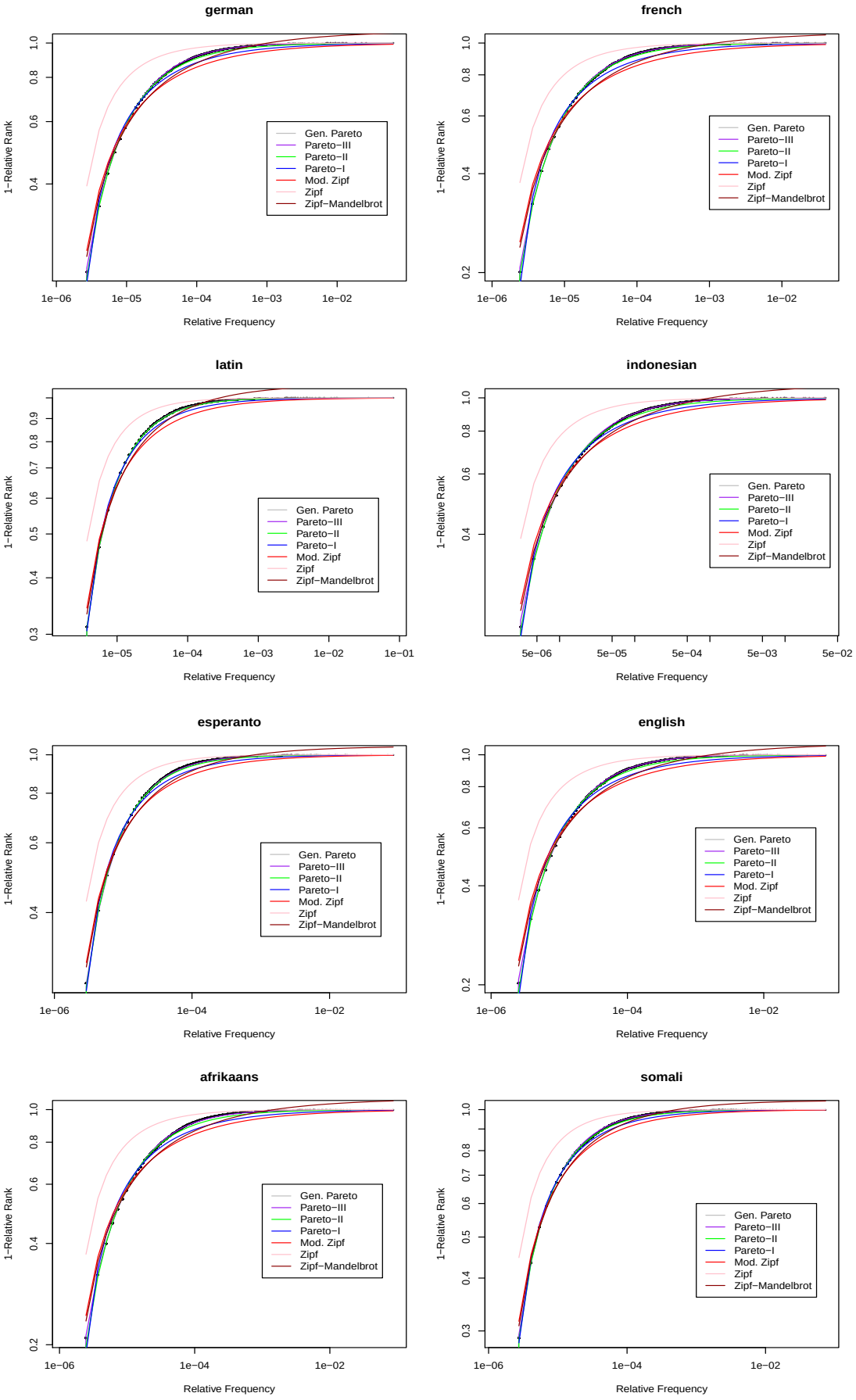


Figure B.1: Log-Log CDF plot of word relative frequency versus relative rank.

	Distribution Characteristics		**Das Kapital**		
Lang.	Distribution	DoF	KS Statistic	Squared Error	ASE times DoF
GERMAN	Pareto Dist. Type III	4	0.000989	0.101761	0.305
	Gen. Pareto Dist.	3	0.003131	0.044452	0.133
	Log Normal	2	0.125374	115.441400	230.883
	Pareto Dist. Type I	2	0.017812	0.691607	1.383
	Zipf-Mandelbrot Law	2	0.068001	2.018978	4.038
	Zipf's Power Law (Mod.)	2	0.040478	5.950488	11.901
	Burr	2	0.176045	211.143100	487.643
	Log Cauchy	2	0.135197	243.821600	274.126
	Zipf's Law (classic)	1	0.180526	274.125700	422.286
FRENCH	Pareto Dist. Type III	4	0.009758	0.338596	1.016
	Gen. Pareto Dist.	3	0.006765	0.083589	0.251
	Log Normal	2	0.112751	93.134930	186.270
	Pareto Dist. Type I	2	0.021963	1.098625	2.197
	Zipf-Mandelbrot Law	2	0.055796	4.431023	8.862
	Zipf's Power Law (Mod.)	2	0.047227	9.309323	18.619
	Burr	2	0.166082	173.945500	392.236
	Log Cauchy	2	0.111906	196.117800	299.774
	Zipf's Law (classic)	1	0.187909	299.773500	347.891
TURKISH	Pareto Dist. Type III	4	0.055628	0.003987	0.012
	Gen. Pareto Dist.	3	0.011121	0.068492	0.205
	Log Normal	2	0.127100	63.798500	127.597
	Pareto Dist. Type I	2	0.020189	0.164572	0.329
	Zipf-Mandelbrot Law	2	0.123286	1.140907	2.282
	Zipf's Power Law (Mod.)	2	0.034198	0.454144	0.908
	Burr	2	0.180358	89.186420	204.098
	Log Cauchy	2	0.129173	102.048800	126.688
	Zipf's Law (classic)	1	0.189347	126.687500	178.373
CHINESE	Pareto Dist. Type III	4	0.029207	0.092366	0.277
	Gen. Pareto Dist.	3	0.042020	0.260400	0.781
	Log Normal	2	0.070829	2.903704	5.807
	Pareto Dist. Type I	2	0.080017	1.466973	2.934
	Zipf-Mandelbrot Law	2	0.214256	0.524249	1.048
	Zipf's Power Law (Mod.)	2	0.108540	3.713127	7.426
	Burr	2	0.103798	6.982924	15.283
	Log Cauchy	2	0.120486	7.641737	67.054
	Zipf's Law (classic)	1	0.300499	67.053630	13.966

Table B.1: Error measures for 'Das Kapital' by Karl Marx (German, French, Turkish and Mandarin Chinese).

	Distribution Characteristics		**Das Kapital**		
Lang.	Distribution	DoF	KS Statistic	Squared Error	ASE times DoF
INDONESIAN	Pareto Dist. Type III	4	0.010335	0.026238	0.079
	Gen. Pareto Dist.	3	0.011806	0.025009	0.075
	Log Normal	2	0.124870	21.219420	42.439
	Pareto Dist. Type I	2	0.020668	0.117719	0.235
	Zipf-Mandelbrot Law	2	0.071161	0.246596	0.493
	Zipf's Power Law (Mod.)	2	0.035496	0.711013	1.422
	Burr	2	0.179344	37.597010	79.493
	Log Cauchy	2	0.313728	39.746680	45.496
	Zipf's Law (classic)	1	0.169070	45.495900	75.194
ENGLISH	Pareto Dist. Type III	4	0.008435	0.215744	0.647
	Gen. Pareto Dist.	3	0.010435	0.132124	0.396
	Log Normal	2	0.097387	63.291270	126.583
	Pareto Dist. Type I	2	0.028562	1.752500	3.505
	Zipf-Mandelbrot Law	2	0.066723	3.752940	7.506
	Zipf's Power Law (Mod.)	2	0.052403	8.702154	17.404
	Burr	2	0.150615	119.992500	262.432
	Log Cauchy	2	0.470600	131.215800	259.606
	Zipf's Law (classic)	1	0.204248	259.605600	239.985
RUSSIAN	Pareto Dist. Type III	4	0.003992	0.042233	0.127
	Gen. Pareto Dist.	3	0.005131	0.028837	0.087
	Log Normal	2	0.131363	213.313700	426.627
	Pareto Dist. Type I	2	0.012634	0.568300	1.137
	Zipf-Mandelbrot Law	2	0.072847	3.317209	6.634
	Zipf's Power Law (Mod.)	2	0.032475	8.037282	16.075
	Burr	2	0.179039	394.726000	925.695
	Log Cauchy	2	0.296138	462.847700	398.102
	Zipf's Law (classic)	1	0.162784	398.102200	789.452
SPANISH	Pareto Dist. Type III	4	0.003907	0.011462	0.034
	Gen. Pareto Dist.	3	0.009693	0.204077	0.612
	Log Normal	2	0.107839	96.216990	192.434
	Pareto Dist. Type I	2	0.022948	1.539986	3.080
	Zipf-Mandelbrot Law	2	0.041964	3.555577	7.111
	Zipf's Power Law (Mod.)	2	0.042811	7.157101	14.314
	Burr	2	0.159146	175.628500	390.770
	Log Cauchy	2	0.122077	195.384900	226.193
	Zipf's Law (classic)	1	0.175983	226.193100	351.257

Table B.2: Error measures for 'Das Kapital' by Karl Marx (Indonesian, English, Russian and Spanish).

	Distribution Characteristics		**The little Prince**		
Lang.	Distribution	DoF	KS Statistic	Squared Error	ASE times DoF
GERMAN	Pareto Dist. Type III	4	0.010283	0.014934	0.045
	Gen. Pareto Dist.	3	0.091233	0.023103	0.092
	Log Normal	2	0.125873	12.742430	50.970
	Pareto Dist. Type I	2	0.023751	0.094740	0.379
	Zipf-Mandelbrot Law	2	0.122776	0.802791	3.211
	Zipf's Power Law (Mod.)	2	0.042614	0.434373	1.737
	Burr	2	0.177032	23.013410	92.054
	Log Cauchy	2	0.281583	27.173910	108.696
	Zipf's Law (classic)	1	0.190018	28.772930	115.092
FRENCH	Pareto Dist. Type III	4	0.010826	0.024786	0.074
	Gen. Pareto Dist.	3	0.041232	0.028924	0.116
	Log Normal	2	0.138717	14.398440	57.594
	Pareto Dist. Type I	2	0.086805	0.060529	0.242
	Zipf-Mandelbrot Law	2	0.108780	0.762251	3.049
	Zipf's Power Law (Mod.)	2	0.035853	0.216583	0.866
	Burr	2	0.184818	25.873200	103.493
	Log Cauchy	2	0.296210	31.752670	127.011
	Zipf's Law (classic)	1	0.173797	23.529340	94.117
TURKISH	Pareto Dist. Type III	4	0.015492	0.055673	0.167
	Gen. Pareto Dist.	3	0.014462	0.047144	0.189
	Log Normal	2	0.158114	13.498670	53.995
	Pareto Dist. Type I	2	0.012170	0.017706	0.071
	Zipf-Mandelbrot Law	2	0.109375	0.737615	2.950
	Zipf's Power Law (Mod.)	2	0.029099	0.355008	1.420
	Burr	2	0.247408	24.305830	97.223
	Log Cauchy	2	0.169249	27.254480	109.018
	Zipf's Law (classic)	1	0.166607	20.661150	82.645
CHINESE	Pareto Dist. Type III	4	0.015158	0.030115	0.090
	Gen. Pareto Dist.	3	0.013652	0.028329	0.113
	Log Normal	2	0.163180	3.319680	13.279
	Pareto Dist. Type I	2	0.048063	0.637872	2.551
	Zipf-Mandelbrot Law	2	0.129672	1.047841	4.191
	Zipf's Power Law (Mod.)	2	0.086325	3.017451	12.070
	Burr	2	0.157066	7.734464	30.938
	Log Cauchy	2	0.170562	8.228197	32.913
	Zipf's Law (classic)	1	0.252295	44.228530	176.914

Table B.3: Error measures for 'The little prince' by Antoine de Saint-Exupery (German, French, Turkish and Mandarin Chinese).

	Distribution Characteristics		**The little Prince**		
Lang.	Distribution	DoF	KS Statistic	Squared Error	ASE times DoF
INDONESIAN	Pareto Dist. Type III	4	0.016569	0.023964	0.072
	Gen. Pareto Dist.	3	0.016388	0.027125	0.108
	Log Normal	2	0.310109	4.714009	18.856
	Pareto Dist. Type I	2	0.023566	0.060322	0.241
	Zipf-Mandelbrot Law	2	0.097035	0.265873	1.063
	Zipf's Power Law (Mod.)	2	0.037071	0.207894	0.832
	Burr	2	0.371639	9.852654	39.411
	Log Cauchy	2	0.421351	12.694100	50.776
	Zipf's Law (classic)	1	0.176325	7.731691	30.927
ENGLISH	Pareto Dist. Type III	4	0.073006	0.082995	0.249
	Gen. Pareto Dist.	3	0.010435	0.132124	0.396
	Log Normal	2	0.124206	9.123872	36.495
	Pareto Dist. Type I	2	0.014737	0.014279	0.057
	Zipf-Mandelbrot Law	2	0.085270	0.466470	1.866
	Zipf's Power Law (Mod.)	2	0.043061	0.878903	3.516
	Burr	2	0.181343	17.184570	68.738
	Log Cauchy	2	0.137855	18.049550	72.198
	Zipf's Law (classic)	1	0.170356	20.530430	82.122
RUSSIAN	Pareto Dist. Type III	4	0.079520	0.414370	1.243
	Gen. Pareto Dist.	3	0.083436	0.352813	1.411
	Log Normal	2	0.211210	6.684591	26.738
	Pareto Dist. Type I	2	0.065747	0.920781	3.683
	Zipf-Mandelbrot Law	2	0.324089	9.219785	36.879
	Zipf's Power Law (Mod.)	2	0.079103	2.263712	9.055
	Burr	2	0.150758	13.293000	53.172
	Log Cauchy	2	0.264715	20.097300	80.389
	Zipf's Law (classic)	1	0.337714	46.661530	186.646
SPANISH	Pareto Dist. Type III	4	0.009998	0.010424	0.031
	Gen. Pareto Dist.	3	0.007514	0.011410	0.046
	Log Normal	2	0.142627	13.028270	52.113
	Pareto Dist. Type I	2	0.014903	0.035024	0.140
	Zipf-Mandelbrot Law	2	0.087250	0.405491	1.622
	Zipf's Power Law (Mod.)	2	0.029674	0.207669	0.831
	Burr	2	0.197122	23.277600	93.110
	Log Cauchy	2	0.292608	27.608610	110.434
	Zipf's Law (classic)	1	0.162215	20.093140	80.373

Table B.4: Error measures for 'The little prince' by Antoine de Saint-Exupery (Indonesian, English, Russian and Spanish).

	Distribution Characteristics		Pinocchio		
Lang.	Distribution	DoF	KS Statistic	Squared Error	ASE times DoF
GERMAN	Pareto Dist. Type III	4	0.010920	0.054067	0.162
	Gen. Pareto Dist.	3	0.008274	0.013924	0.056
	Log Normal	2	0.125667	27.007040	108.028
	Pareto Dist. Type I	2	0.018940	0.104781	0.419
	Zipf-Mandelbrot Law	2	0.107907	0.898324	3.593
	Zipf's Power Law (Mod.)	2	0.035474	1.459127	5.837
	Burr	2	0.181333	49.486870	197.947
	Log Cauchy	2	0.123230	56.322290	225.289
	Zipf's Law (classic)	1	0.190238	73.065740	292.263
FRENCH	Pareto Dist. Type III	4	0.007784	0.026488	0.079
	Gen. Pareto Dist.	3	0.007600	0.027607	0.110
	Log Normal	2	0.105425	21.197170	84.789
	Pareto Dist. Type I	2	0.024887	0.299564	1.198
	Zipf-Mandelbrot Law	2	0.076005	1.017654	4.071
	Zipf's Power Law (Mod.)	2	0.053776	2.163238	8.653
	Burr	2	0.159304	40.450520	161.802
	Log Cauchy	2	0.269505	47.360050	189.440
	Zipf's Law (classic)	1	0.179127	55.309500	221.238
TURKISH	Pareto Dist. Type III	4	0.042636	0.026488	0.079
	Gen. Pareto Dist.	3	0.006587	0.018381	0.074
	Log Normal	2	0.107828	17.839130	71.357
	Pareto Dist. Type I	2	0.020472	0.130040	0.520
	Zipf-Mandelbrot Law	2	0.123267	0.976386	3.906
	Zipf's Power Law (Mod.)	2	0.049950	1.876398	7.506
	Burr	2	0.159509	34.397370	137.589
	Log Cauchy	2	0.110917	7.832272	31.329
	Zipf's Law (classic)	1	0.196562	61.574900	246.300
CHINESE	Pareto Dist. Type III	4	0.007770	0.018062	0.054
	Gen. Pareto Dist.	3	0.109392	0.059852	0.239
	Log Normal	2	0.265203	2.756889	11.028
	Pareto Dist. Type I	2	0.058623	1.648950	6.596
	Zipf-Mandelbrot Law	2	0.160274	1.782611	7.130
	Zipf's Power Law (Mod.)	2	0.095002	5.119466	20.478
	Burr	2	0.291168	7.256345	29.025
	Log Cauchy	2	0.291153	19.339500	77.358
	Zipf's Law (classic)	1	0.279496	69.237590	276.950

Table B.5: Error measures for 'Pinocchio' by Carlo Collodi (German, French, Turkish and Mandarin Chinese).

	Distribution Characteristics		**Pinocchio**		
Lang.	Distribution	DoF	KS Statistic	Squared Error	ASE times DoF
INDONESIAN	Pareto Dist. Type III	4	0.014447	0.023032	0.069
	Gen. Pareto Dist.	3	0.045285	0.017133	0.069
	Log Normal	2	0.105582	9.066435	36.266
	Pareto Dist. Type I	2	0.030346	0.166342	0.665
	Zipf-Mandelbrot Law	2	0.190379	1.030238	4.121
	Zipf's Power Law (Mod.)	2	0.060260	1.236936	4.948
	Burr	2	0.163023	17.619480	70.478
	Log Cauchy	2	0.538656	25.695250	102.781
	Zipf's Law (classic)	1	0.201514	34.707650	138.831
ENGLISH	Pareto Dist. Type III	4	0.009223	0.031210	0.094
	Gen. Pareto Dist.	3	0.008701	0.020349	0.081
	Log Normal	2	0.095324	11.657100	46.628
	Pareto Dist. Type I	2	0.031876	0.383924	1.536
	Zipf-Mandelbrot Law	2	0.124107	1.149552	4.598
	Zipf's Power Law (Mod.)	2	0.067266	3.063546	12.254
	Burr	2	0.151321	23.615840	94.463
	Log Cauchy	2	0.222663	79.252970	317.012
	Zipf's Law (classic)	1	0.216428	63.470480	253.882
RUSSIAN	Pareto Dist. Type III	4	0.044922	0.463791	1.391
	Gen. Pareto Dist.	3	0.051328	0.570696	2.283
	Log Normal	2	0.237524	39.735600	158.942
	Pareto Dist. Type I	2	0.057004	0.650418	2.602
	Zipf-Mandelbrot Law	2	0.140141	6.074273	24.297
	Zipf's Power Law (Mod.)	2	0.057002	0.650418	2.602
	Burr	2	0.315230	62.200950	248.804
	Log Cauchy	2	0.224976	79.252970	317.012
	Zipf's Law (classic)	1	0.215321	49.786120	199.144
SPANISH	Pareto Dist. Type III	4	0.095358	0.058764	0.176
	Gen. Pareto Dist.	3	0.093716	0.088388	0.354
	Log Normal	2	0.115551	25.078960	100.316
	Pareto Dist. Type I	2	0.023150	0.313913	1.256
	Zipf-Mandelbrot Law	2	0.066820	0.423785	1.695
	Zipf's Power Law (Mod.)	2	0.039450	1.223027	4.892
	Burr	2	0.175001	45.366850	181.467
	Log Cauchy	2	0.262715	51.498490	205.994
	Zipf's Law (classic)	1	0.174348	50.285390	201.142

Table B.6: Error measures for 'Pinocchio' by Carlo Collodi (Indonesian, English, Russian and Spanish).

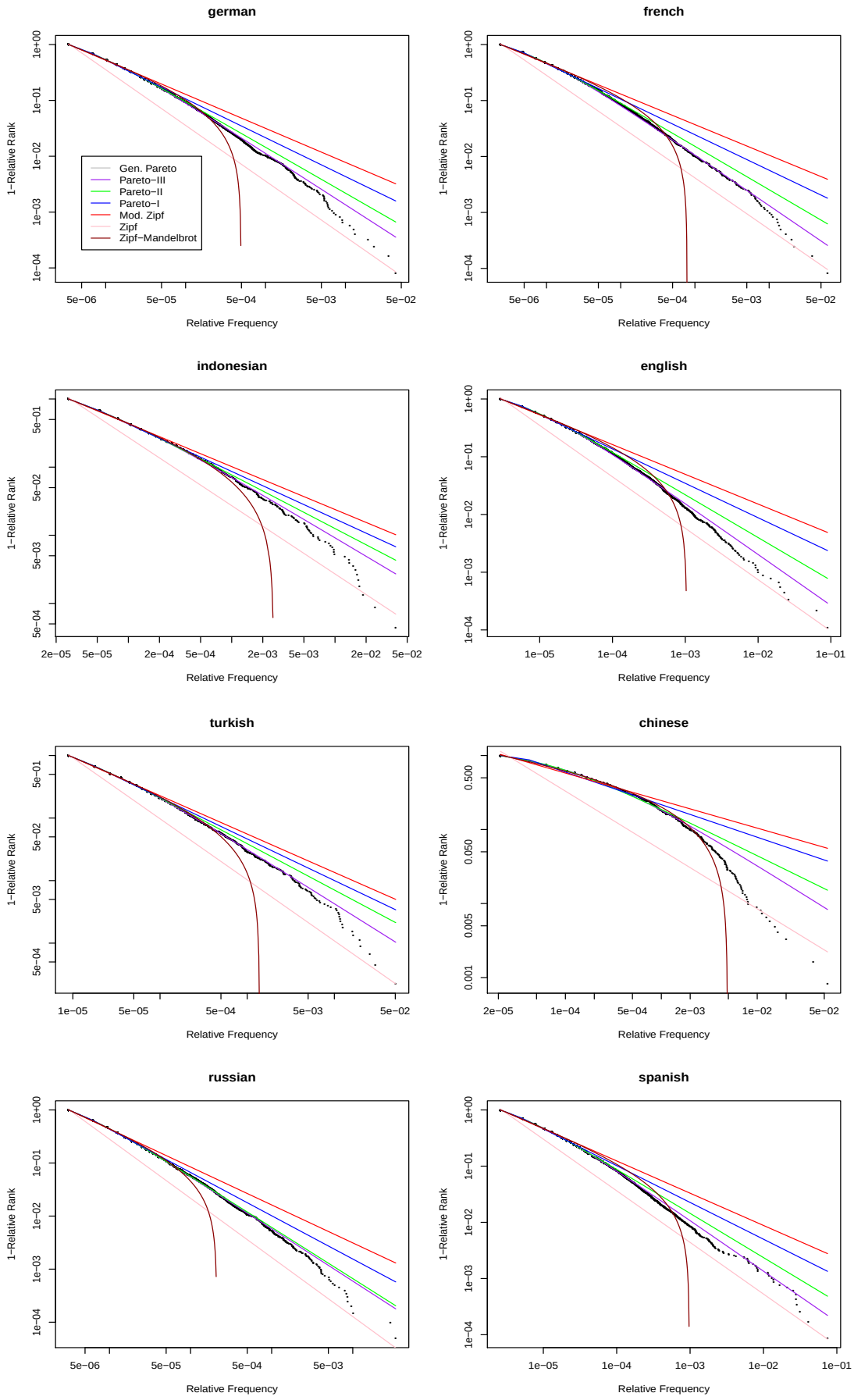


Figure B.2: Log-Log CDF plot of word relative frequency versus relative rank for 'Das Kapital'.

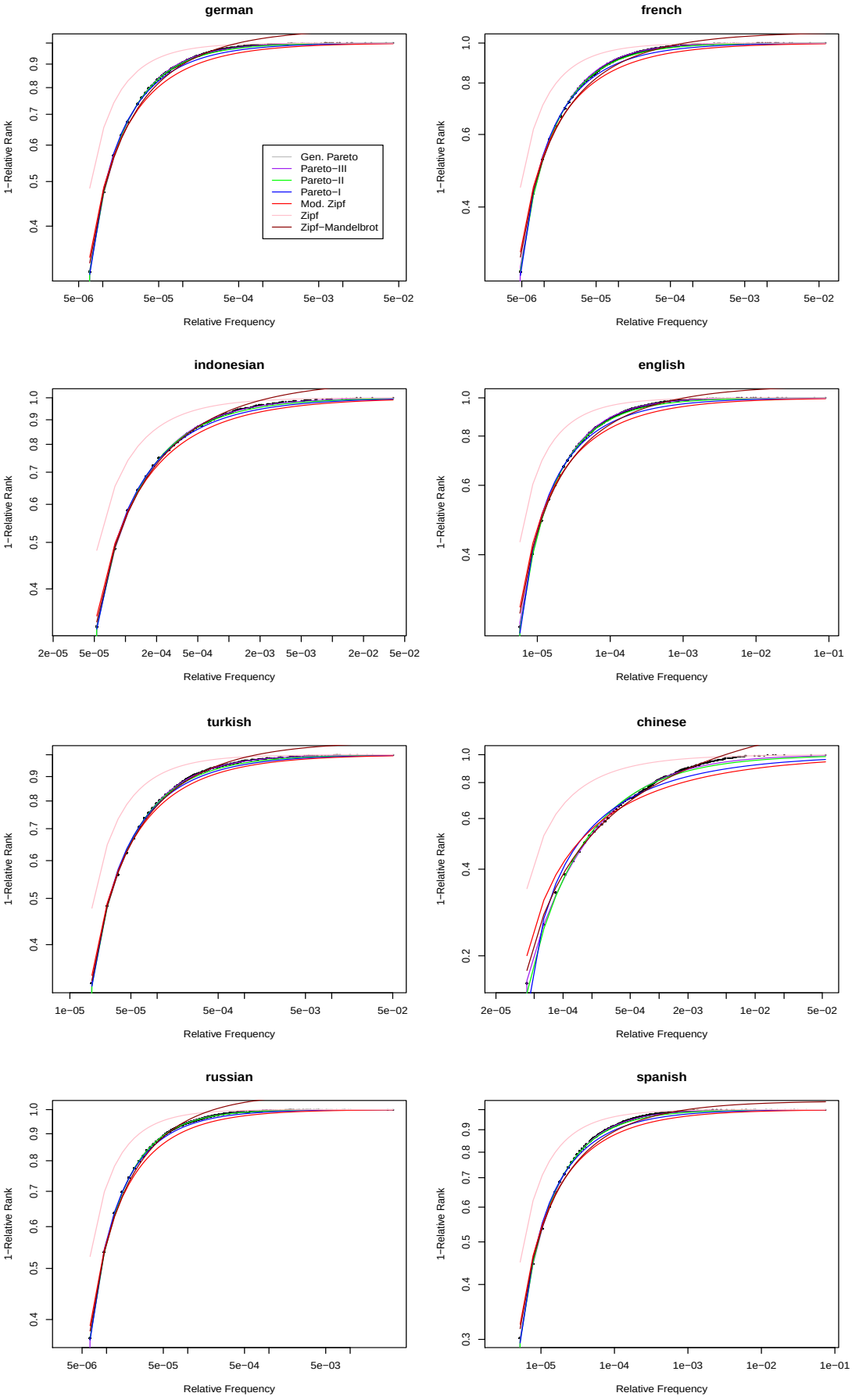


Figure B.3: Log-Log CDF plot of word relative frequency versus inverse relative rank for 'Das Kapital'.

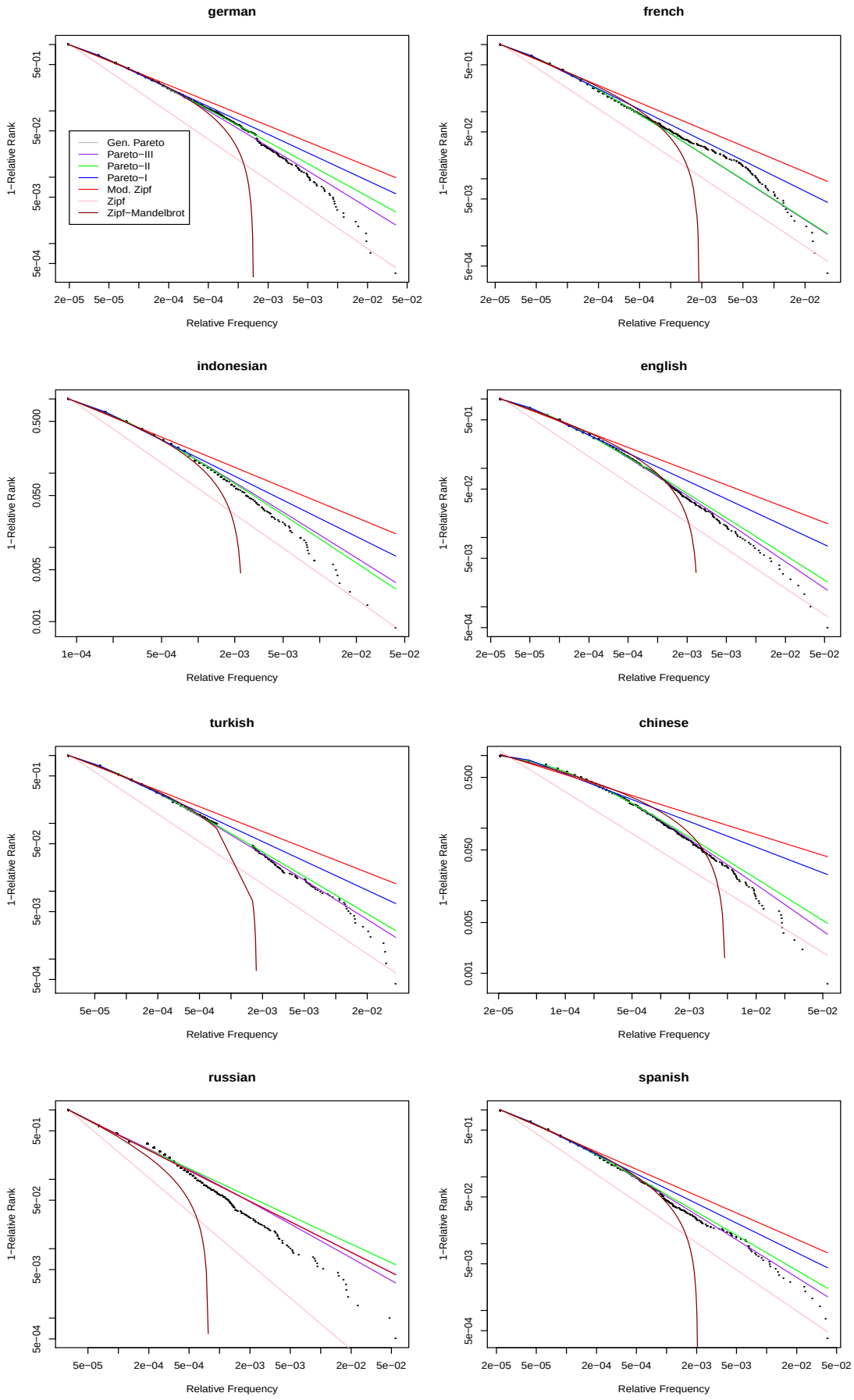


Figure B.4: Log-Log CDF plot of word relative frequency versus relative rank for 'Pinocchio'.

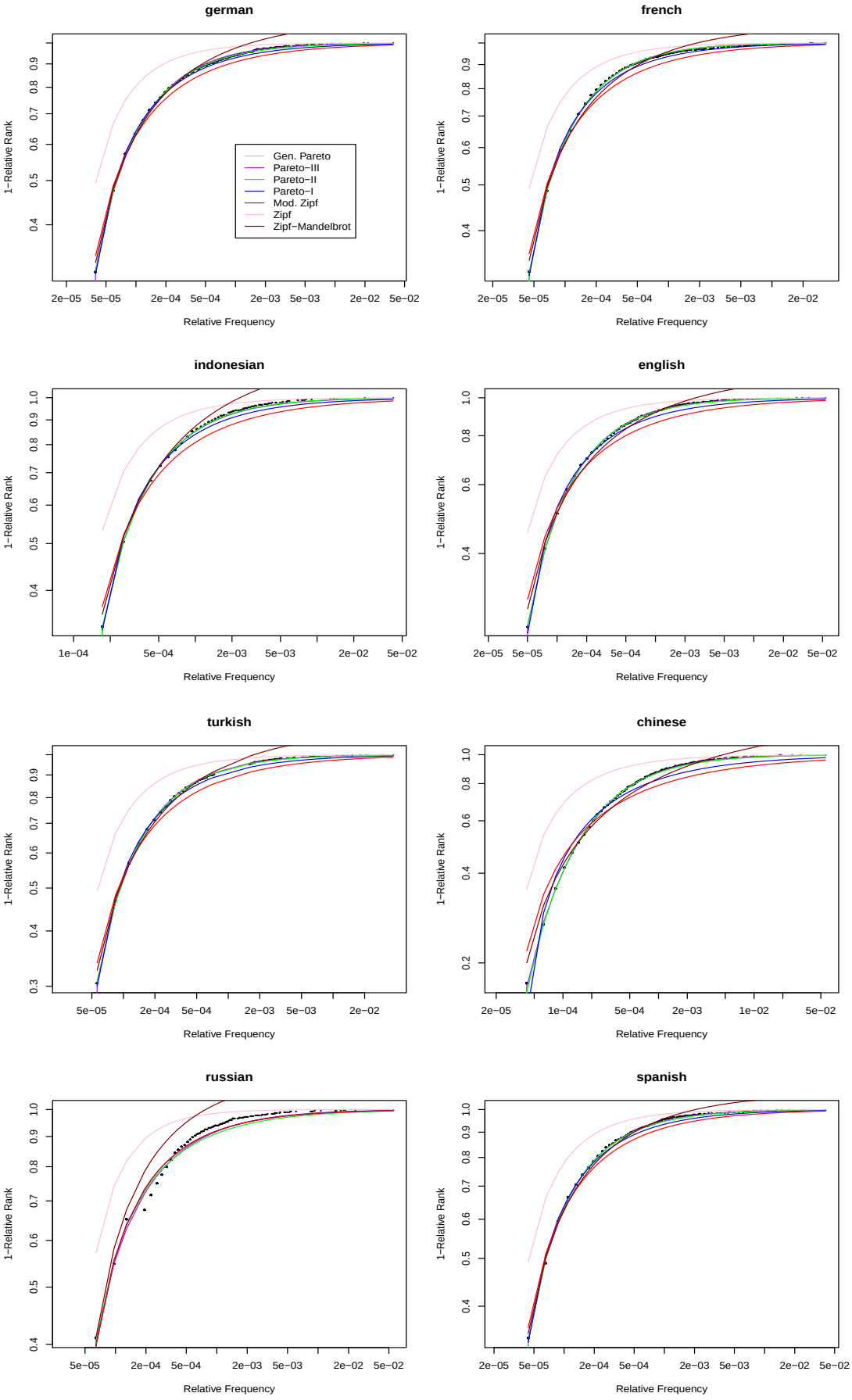


Figure B.5: Log-Log CDF plot of word relative frequency versus inverse relative rank for 'Pinocchio'.

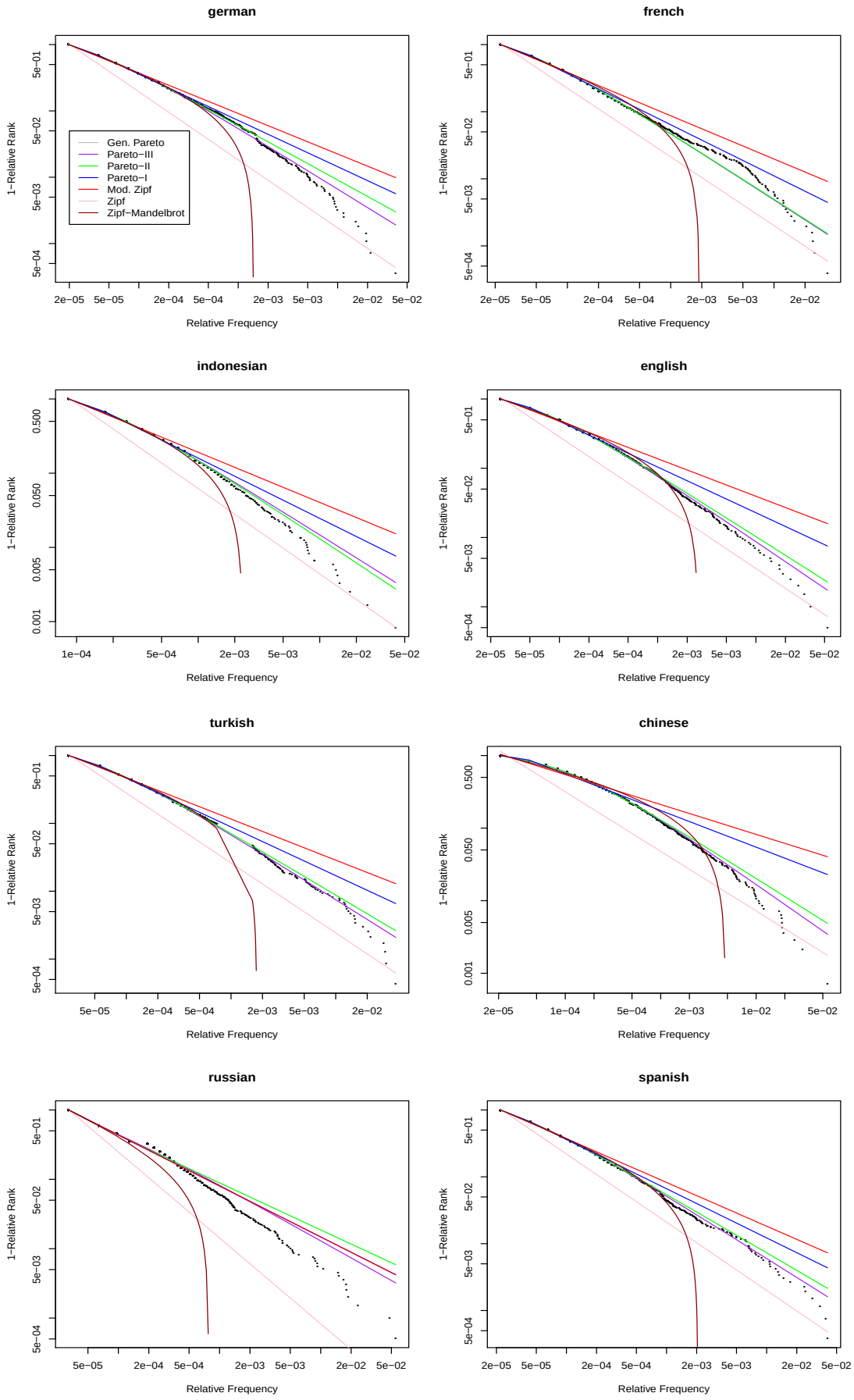


Figure B.6: Log-Log CDF plot of word relative frequency versus relative rank for 'The little Prince'.

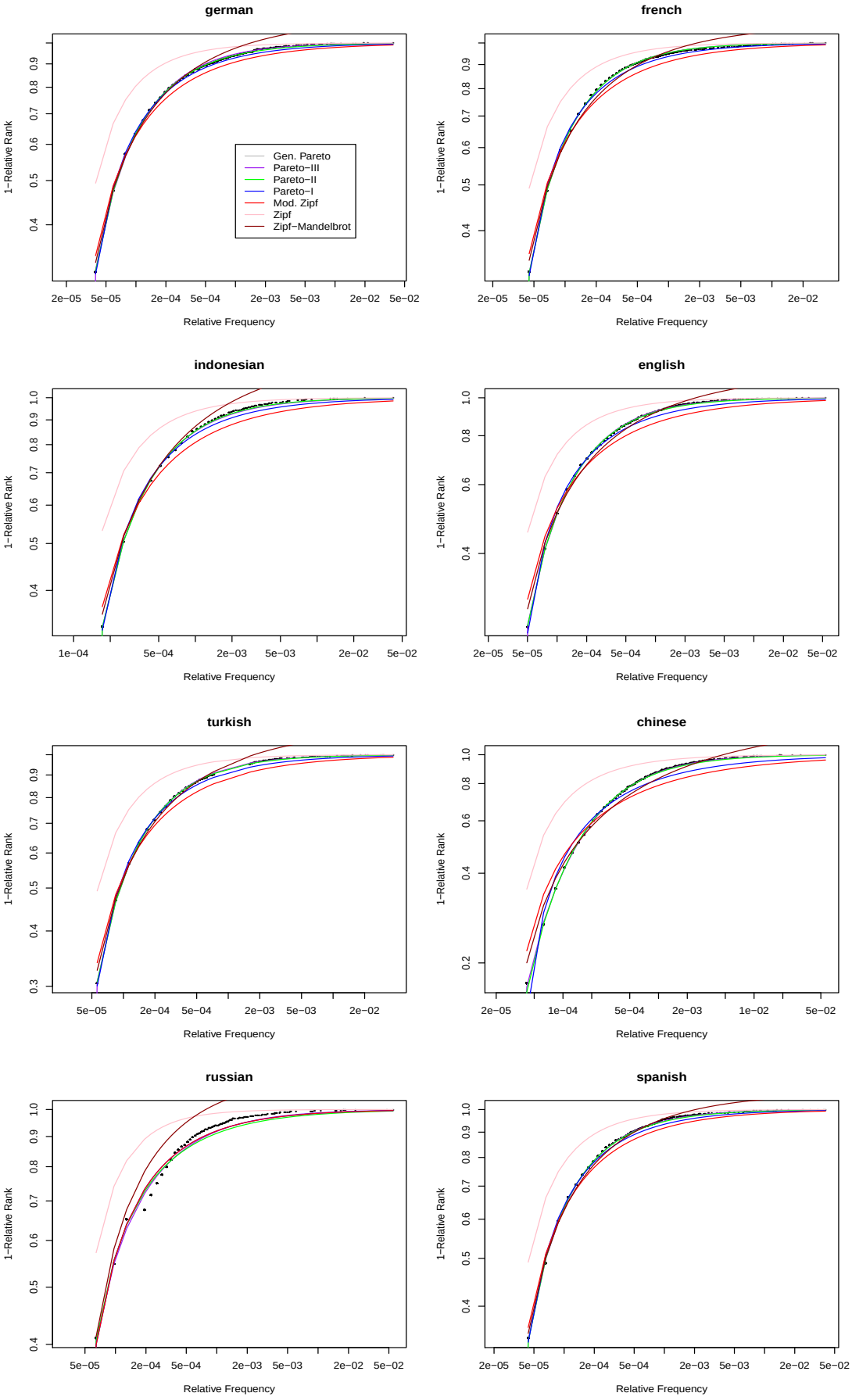


Figure B.7: Log-Log CDF plot of word relative frequency versus inverse relative rank for 'The little Prince'.

Appendix C

General Moments of Round off Errors Proofs and Supplementary Material

C.1 Proofs of Theorems 5.2.2 to 5.2.5

Proof of Theorem 5.2.2

Note that

$$\mathbb{E}\left[(X - \lfloor X \rfloor)^k\right] = \sum_{i=0}^{k} (-1)^i \binom{k}{i} \mathbb{E}\left[X^{k-i} \lfloor X \rfloor^i\right].$$

We investigate the expected values, and follow the basic principle of the proof for Theorems 1 and 3 in Li and Nadarajah [Li, Nadarajah (2016)]:

$$\begin{aligned}
\mathbb{E}\left[X^{k-i} \lfloor X \rfloor^i\right] &= \int_{-\infty}^{\infty} x^{k-i} \lfloor x \rfloor^i f(x) dx \\
&= \sum_{j=-\infty}^{\infty} j^i \int_{j}^{j+1} x^{k-i} f(x) dx \\
&= \sum_{j=-\infty}^{\infty} j^i \left[\int_{-\infty}^{j+1} x^{k-i} f(x) dx - \int_{-\infty}^{j} x^{k-i} f(x) dx\right] \\
&= \sum_{j=-\infty}^{\infty} j^i \left[M_{k-i}(j+1) - M_{k-i}(j)\right].
\end{aligned}$$

By inserting the result above into the first equation, we have completed the proof.

Proof of Theorem 5.2.3

Once more, by the binomial theorem:

$$\mathbb{E}\left[\left(X - \left\lfloor X + \frac{1}{2}\right\rfloor\right)^k\right] = \sum_{i=0}^{k}(-1)^i\binom{k}{i}\mathbb{E}\left[X^{k-i}\left\lfloor X + \frac{1}{2}\right\rfloor^i\right].$$

Set $Y = X + \frac{1}{2}$. Then

$$\begin{aligned}\int_{-\infty}^{\infty} x^{k-i}\left\lfloor x + \frac{1}{2}\right\rfloor f(x)dx &= \int_{-\infty}^{\infty}\left(y - \frac{1}{2}\right)^{k-i}\lfloor y\rfloor\, f\left(y - \frac{1}{2}\right)dy \\ &= \sum_{j=-\infty}^{\infty} j^i \int_j^{j+1}\left(y - \frac{1}{2}\right)^{k-i} f\left(y - \frac{1}{2}\right)dy \\ &= \sum_{j=-\infty}^{\infty} j^i\left[\int_{-\infty}^{j+\frac{1}{2}}(y)^{k-i} f(y)\,dy - \int_{-\infty}^{j-\frac{1}{2}}(y)^{k-i} f(y)\,dy\right] \\ &= \sum_{j=-\infty}^{\infty} j^i\left[M_{k-i}\left(j + \frac{1}{2}\right) - M_{k-i}\left(j - \frac{1}{2}\right)\right].\end{aligned}$$

Proof of Theorem 5.2.4

We know that for a CDF F on the domain $\mathbb{R}$,

$$\mathbb{E}\left[(X - \lfloor X\rfloor)^k\right] = \sum_{i=0}^{k}(-1)^i\binom{k}{i}\sum_{j=-\infty}^{\infty} j^i\left[M_{k-i}(j+1) - M_{k-i}(j)\right].$$

With the premises of Theorem 5.2.4, $f(x) = 0$ for $x < a$ or $b < x$. This also satisfies the condition for Proposition 2.1, such that $M_t(x) = 0$ for $x < a$ and $M_t(x) = b$. This in turn leads to $M_{k-i}(j+1) - M_{k-i}(j) = 0 - 0 = 0$ for $j + 1 < a$. On the upper bound a similar result holds true: $M_{k-i}(j+1) - M_{k-i}(j+1) = M_{k-i}(b) - M_{k-i}(b) = 0$ for all $j > b$. We focus on the inner sum over the index j, as the rest remains the same. We note that all terms $j^i\left[M_{k-i}(j+1) - M_{k-i}(j)\right]$ are vanishing for $j > q$ or $j < p$ with $p = \lfloor a\rfloor$ and $q = \lceil b\rceil$. The remaining non-zero sum terms give us the final result of the proof:

$$\mathbb{E}\left[(X - \lfloor X\rfloor)^k\right] = \sum_{i=0}^{k}(-1)^i\binom{k}{i}\left[\sum_{j=p}^{q-1} j^i\left(M_{k-i}(j+1) - M_{k-i}(j)\right)\right].$$

Proof of Theorem 5.2.5 Since, obviously, $j + \frac{1}{2} \leq a \leftrightarrow j \leq a - \frac{1}{2} \leftarrow j \leq \left\lfloor a - \frac{1}{2}\right\rfloor \leq a - \frac{1}{2}$ as well as $j - \frac{1}{2} \geq b \leftrightarrow j \geq b + \frac{1}{2} \leftarrow j \geq \left\lceil b + \frac{1}{2}\right\rceil \geq b + \frac{1}{2}$,

we have from the statement of Theorem 5.2.3

$$\mathbb{E}\left[\left(X - \left\lfloor X + \frac{1}{2}\right\rfloor\right)^k\right] = \sum_{i=0}^{k}(-1)^i\binom{k}{i}\sum_{j=-\infty}^{\infty} j^i\left[M_{k-i}\left(j + \frac{1}{2}\right) - M_{k-i}\left(j - \frac{1}{2}\right)\right].$$

Successively,

$$\begin{aligned}
&\sum_{j=-\infty}^{\infty} j^i \left[M_{k-i}\left(j+\frac{1}{2}\right) - M_{k-i}\left(j-\frac{1}{2}\right)\right] \\
&= \sum_{j=-\infty}^{\lfloor a-\frac{1}{2}\rfloor} j^i \left[M_{k-i}\left(j+\frac{1}{2}\right) - M_{k-i}\left(j-\frac{1}{2}\right)\right] \\
&\qquad + \sum_{j=\lfloor a-\frac{1}{2}\rfloor+1}^{\lceil b+\frac{1}{2}\rceil-1} j^i \left[M_{k-i}\left(j+\frac{1}{2}\right) - M_{k-i}\left(j-\frac{1}{2}\right)\right] \\
&\qquad + \sum_{j=\lceil b+\frac{1}{2}\rceil}^{\infty} j^i \left[M_{k-i}\left(j+\frac{1}{2}\right) - M_{k-i}\left(j-\frac{1}{2}\right)\right] \\
&= \sum_{j=\lfloor a+\frac{1}{2}\rfloor}^{\lceil b+\frac{1}{2}\rceil-1} j^i \left[M_{k-i}\left(j+\frac{1}{2}\right) - M_{k-i}\left(j-\frac{1}{2}\right)\right].
\end{aligned}$$

The proof is complete.

C.1.1 Further Commonly Used Distributions

Trapezoidal Distribution (Hou et al. [Hou et al. (2009)])

For a trapezoidal random variable X with PDF and CDF specified by:

$$f_X(x) = \frac{2}{d+b-a-c}\begin{cases} 0 & \text{if } x < a \\ \frac{x-a}{c-a} & \text{if } a \le x < c \\ 1 & c \le x < d \\ \frac{b-x}{b-d} & \text{if } d \le x < b \\ 0 & \text{if } x \ge a \end{cases},$$

$$F_X(x) = \frac{2}{d+b-a-c}\begin{cases} 0 & \text{if } x < a \\ \frac{x\left(\frac{x}{2}-a\right)+\frac{a}{2}}{c-a} & \text{if } a \le x < c \\ (x-c)+\frac{c-a}{2} & \text{if } c \le x < d \\ \frac{x\left(b-\frac{x}{2}\right)-d\left(b-\frac{d}{2}\right)}{b-d}+(d-c)+\frac{c-a}{2} & \text{if } d \le x < b \\ 1 & \text{if } x \ge b \end{cases}$$

$$M_k(x) = F^{(-k)}(x)$$

$$= \frac{2}{d+b-a-c} \begin{cases} 0 & \text{if } x < a \\ \frac{1}{(c-a)}\left(x^{k+1}\left(\frac{x}{(k+2)!} - \frac{a}{(k+1)!}\right) - a^{k+2}\left(\frac{1}{(k+2)!} - \frac{1}{(k+1)!}\right)\right) & \text{if } a \le x < c \\ \frac{1}{(c-a)}\left(c^{k+1}\left(\frac{c}{(k+2)!} - \frac{a}{(k+1)!}\right) - a^{k+2}\left(\frac{1}{(k+2)!} - \frac{1}{(k+1)!}\right)\right) + \left(\frac{x^{k+2}}{(k+2)!} - \frac{c^{k+2}}{(k+2)!}\right) & \text{if } c \le x < d \\ \frac{1}{(c-a)}\left(c^{k+1}\left(\frac{c}{(k+2)!} - \frac{a}{(k+1)!}\right) - a^{k+2}\left(\frac{1}{(k+2)!} - \frac{1}{(k+1)!}\right)\right) + \left(\frac{d^{k+2}}{(k+2)!} - \frac{c^{k+2}}{(k+2)!}\right) + \frac{1}{(b-d)}\left(x^{k+1}\left(\frac{b}{(k+1)!} - \frac{x}{(k+2)!}\right) - d^{k+1}\left(\frac{b}{(k+1)!} - \frac{d}{(k+2)!}\right)\right) & \text{if } d \le x < b \\ \frac{1}{(c-a)}\left(c^{k+1}\left(\frac{c}{(k+2)!} - \frac{a}{(k+1)!}\right) - a^{k+2}\left(\frac{1}{(k+2)!} - \frac{1}{(k+1)!}\right)\right) + \left(\frac{d^{k+2}}{(k+2)!} - \frac{c^{k+2}}{(k+2)!}\right) + \frac{1}{(b-d)}\left(b^{k+2}\left(\frac{1}{(k+1)!} - \frac{1}{(k+2)!}\right) - d^{k+1}\left(\frac{b}{(k+1)!} - \frac{d}{(k+2)!}\right)\right) & \text{if } b \le x \end{cases}$$

C.1.2 House Distribution ([Widrow, Kollar (2008)], 3.9, page 55)

For a house random variable X with PDF and CDF specified by

$$f_X(x) = \begin{cases} 0 & \text{if } x < -A \\ (1+\alpha)B + \frac{\alpha B x}{A} & \text{if } -A \le x \le 0 \\ (1+\alpha)B - \frac{\alpha B x}{A} & \text{if } 0 < x \le A \\ 0 & \text{for } x > A \end{cases}$$

$$F_X(x) = \begin{cases} 0 & x < -A \\ (1+\alpha)B(A+x) + \frac{\alpha B\left(x^2 - A^2\right)}{2A} & -A \le x \le 0 \\ (1+\alpha)B(x) - \frac{\alpha B x^2}{2A} + \frac{1}{2} & 0 < x \le A \\ 1 & x > A \end{cases}$$

for $A > 0$ and $\alpha > 0$, where $B = \frac{1}{2A + A\alpha}$, we have:

$$M_k(x) = F^{(-k)}(x)$$

$$= \frac{2}{d+b-a-c} \begin{cases} 0 & \text{if } x < -A \\ (1+\alpha)\dfrac{Bx^{k+1}}{(k+1)!} + \dfrac{\alpha B}{A}\dfrac{x^{k+2}}{(k+2)!} & \\ -\left((1+\alpha)\dfrac{B(-A)^{k+1}}{(k+1)!} + \dfrac{\alpha B}{A}\dfrac{(-A)^{k+2}}{(k+2)!}\right) & \text{if } -A \le x < 0 \\ \left((1+\alpha)\dfrac{B(-A)^{k+1}}{(k+1)!} + \dfrac{\alpha B}{A}\dfrac{(-A)^{k+2}}{(k+2)!}\right) & \\ +\left((1+\alpha)B\dfrac{x^{k+1}}{(k+1)!} - \dfrac{\alpha B}{A}\dfrac{x^{k+2}}{(k+2)!}\right) & \text{if } 0 \le x < A \\ \left((1+\alpha)\dfrac{B(-A)^{k+1}}{(k+1)!} + \dfrac{\alpha B}{A}\dfrac{(-A)^{k+2}}{(k+2)!}\right) & \\ +\left((1+\alpha)B\dfrac{A^{k+1}}{(k+1)!} - \dfrac{\alpha B}{A}\dfrac{A^{k+2}}{(k+2)!}\right) & \text{if } 0 \le x < A \end{cases}$$

Curved Trapezoidal Distribution (Widrow and Kollar [Widrow, Kollar (2008)], 7.11, page 168; Kawarai and Murakami [Kawarai, Murakami (1989)], page 884)

For a curved trapezoidal random variable X with PDF and CDF specified by

$$f_X(x) = \begin{cases} a\left(-\frac{\rho}{x} - 1\right) & \text{if } -\rho \le x < -\frac{\rho}{2} \\ a & \text{if } -\frac{\rho}{2} \le x < \frac{\rho}{2} \\ a\left(\frac{\rho}{x} - 1\right) & \text{if } \frac{\rho}{2} \le x \le \rho \\ 0 & \text{else} \end{cases}$$

$$F_X(x) = \begin{cases} -a\left[x + \rho + \rho\log\left(\frac{-x}{\rho}\right)\right] & \text{if } -\rho \le x < -\frac{\rho}{2} \\ ax + a\rho\log(2) & \text{if } -\frac{\rho}{2} \le x < \frac{\rho}{2} \\ a\left[(\rho - x) + \log\left(\frac{x}{\rho}\right) + \rho\log(4)\right] & \text{if } \frac{\rho}{2} \le x \le \rho \\ 1 & x > \rho \end{cases}$$

for $\rho > 0$ and $a = \frac{1}{\rho \log(4)}$, we have:

$$
\begin{aligned}
M_k(x) &= F^{(-k)}(x) \\
&= a \begin{cases}
0 & \text{if } x < -\rho \\
-\frac{x^k}{k!}\left(log(x) - H(k)\right) - \frac{x^{k+1}}{(k+1)!} & \\
+\frac{(-\rho)^k}{k!}\left(log(-\rho) - H(k)\right) + \frac{(-\rho)^{k+1}}{(k+1)!} & \text{if } -\rho \le x < -\frac{\rho}{2} \\
-\frac{(\frac{-\rho}{2})^k}{k!}\left(log\left(\frac{-\rho}{2}\right) - H(k)\right) - \frac{\frac{-\rho}{2}^{k+1}}{(k+1)!} & \\
+\frac{(-\rho)^k}{k!}\left(log(-\rho) - H(k)\right) + \frac{(-\rho)^{k+1}}{(k+1)!} & \text{if } -\frac{\rho}{2} \le x < \frac{\rho}{2} \\
+\frac{x^{k+1}}{(k+1)!} - \frac{(\frac{-\rho}{2})^{k+1}}{(k+1)!} & \\
-\frac{(\frac{-\rho}{2})^k}{k!}\left(log\left(\frac{-\rho}{2}\right) - H(k)\right) & \\
-\frac{\frac{-\rho}{2}^{k+1}}{(k+1)!} + \frac{(-\rho)^k}{k!}\left(log(-\rho) + H(k)\right) & \\
+\frac{(-\rho)^{k+1}}{(k+1)!} + \frac{(\frac{\rho}{2})^{k+1}}{(k+1)!} & \\
-\frac{(\frac{-\rho}{2})^{k+1}}{(k+1)!} + \frac{x^k}{k!}\left(log\left(x\right) - H(k)\right) & \text{if } \frac{\rho}{2} \le x < \rho \\
-\frac{x^{k+1}}{(k+1)!} + \frac{(-\rho)^k}{k!}\left(H(k) - log\left(\frac{\rho}{2}\right)\right) & \\
+\frac{\left(\frac{\rho}{2}\right)^{k+1}}{(k+1)!} & \\
-\frac{(\frac{-\rho}{2})^k}{k!}\left(log\left(\frac{-\rho}{2}\right) - H(k)\right) - \frac{\frac{-\rho}{2}^{k+1}}{(k+1)!} & \\
+\frac{(-\rho)^k}{k!}\left(log(-\rho) + H(k)\right) + \frac{(-\rho)^{k+1}}{(k+1)!} & \\
+\frac{(\frac{\rho}{2})^{k+1}}{(k+1)!} - \frac{(\frac{-\rho}{2})^{k+1}}{(k+1)!} & \text{if } \rho \le x \\
+\frac{\rho^k}{k!}\left(log\left(\rho\right) - H(k)\right) - \frac{\rho^{k+1}}{(k+1)!} & \\
+\frac{\left(\frac{\rho}{2}\right)^k}{k!}\left(H(k) - log\left(\frac{\rho}{2}\right)\right) + \frac{\left(\frac{\rho}{2}\right)^{k+1}}{(k+1)!} &
\end{cases}
\end{aligned}
$$

with $H(n)$ being the n-th harmonic number.

Hexagonal Distribution (Widrow and Kollar [Widrow, Kollar (2008)], E3.13.1, page 56)

For a curved trapezoidal random variable X with PDF and CDF specified by:

$$f_X(x) = \begin{cases} 0 & \text{if } x < -3a \\ \frac{x+3a}{6a^2} & \text{if } -3a \le x < -2a \\ \frac{1}{6a} & \text{if } -2a \le x < -a \\ \frac{1}{6a} + \frac{x+a}{6a^2} & \text{if } -a \le x < 0 \\ \frac{1}{6a} + \frac{-x+a}{6a^2} & \text{if } 0 \le x < a \\ \frac{1}{6a} & \text{if } a \le x < 2a \\ \frac{-x+3a}{6a^2} & \text{if } 2a \le x \le 3a \\ 0 & \text{if } x > 3a \end{cases} \quad F_X(x) = \begin{cases} 0 & \text{if } x < -3a \\ \frac{x^2}{12a^2} + \frac{x}{2a} + \frac{3}{4} & \text{if } -3a \le x < -2a \\ \frac{x}{6a} + \frac{5}{12} & \text{if } -2a \le x < -a \\ \frac{x^2}{12a^2} + \frac{x}{3a} + \frac{1}{2} & \text{if } -a \le x < 0 \\ \frac{x}{3a} - \frac{x^2}{12a^2} + \frac{1}{2} & \text{if } 0 \le x < a \\ \frac{x}{6a} + \frac{7}{12} & \text{if } a \le x < 2a \\ \frac{-x^2}{12a^2} + \frac{x}{2a} + \frac{1}{4} & \text{if } 2a \le x \le 3a \\ 1 & \text{if } x > 3a \end{cases}$$

for $a > 0$ and $a = \frac{1}{\rho \log(4)}$.

$$M_k(x) = F^{(-k)}(x) = A + B$$

With the sub-terms A and B designed as follows:

$$A = \begin{cases} 0 & \text{else} \\ \left[\frac{1}{6a^2}\left(\frac{x^{k+2}}{(k+2)!} + 3a(\frac{x^{k+1}}{(k+1)!} - \frac{(-3a)^{k+2}}{(k+2)!} - 3a(\frac{(-3a)^{k+1}}{(k+1)!}\right)\right] & \text{if } -3a \le x < -2a \\ \left[\frac{1}{6a^2}\left(\frac{(-2a)^{k+2}}{(k+2)!} + 3a(\frac{(-2a)^{k+1}}{(k+1)!} - \frac{(-3a)^{k+2}}{(k+2)!} - 3a(\frac{(-3a)^{k+1}}{(k+1)!}\right) + \frac{1}{6a}\left(\frac{x^{k+1}}{(k+1)!} - \frac{(-2a)^{k+1}}{(k+1)!}\right)\right] & \text{if } -2a \le x < -a \\ \left[\frac{1}{6a^2}\left(\frac{(-2a)^{k+2}}{(k+2)!} + 3a(\frac{(-2a)^{k+1}}{(k+1)!} - \frac{(-3a)^{k+2}}{(k+2)!} - 3a(\frac{(-3a)^{k+1}}{(k+1)!}\right) + \frac{1}{6a}\left(\frac{(-a)^{k+1}}{(k+1)!} - \frac{(-2a)^{k+1}}{(k+1)!}\right) + \frac{1}{6a}\left(\frac{2x^{k+1}}{(k+1)!} + \frac{x^{k+2}}{a(k+2)!} - \frac{2(-a)^{k+1}}{(k+1)!} - \frac{(-a)^{k+2}}{a(k+2)!}\right)\right] & \text{if } -a \le x < 0 \\ \left[\frac{1}{6a^2}\left(\frac{(-2a)^{k+2}}{(k+2)!} + 3a(\frac{(-2a)^{k+1}}{(k+1)!} - \frac{(-3a)^{k+2}}{(k+2)!} - 3a(\frac{(-3a)^{k+1}}{(k+1)!}\right) + \frac{1}{6a}\left(\frac{(-a)^{k+1}}{(k+1)!} - \frac{(-2a)^{k+1}}{(k+1)!}\right) - \frac{1}{6a}\left(\frac{2(-a)^{k+1}}{(k+1)!} + \frac{(-a)^{k+2}}{a(k+2)!}\right) + \frac{1}{6a}\left(\frac{2x^{k+1}}{(k+1)!} - \frac{x^{k+2}}{a(k+2)!}\right)\right] & \text{if } 0 \le x < a \end{cases}$$

$$
B = \begin{cases}
0 & \text{else} \\
\left[\frac{1}{6a^2}\left(\frac{(-2a)^{k+2}}{(k+2)!} + 3a(\frac{(-2a)^{k+1}}{(k+1)!} - \frac{(-3a)^{k+2}}{(k+2)!} - 3a(\frac{(-3a)^{k+1}}{(k+1)!}\right)\right. & \\
\qquad + \frac{1}{6a}\left(\frac{(-a)^{k+1}}{(k+1)!} - \frac{(-2a)^{k+1}}{(k+1)!}\right) & \\
\qquad - \frac{1}{6a}\left(\frac{2(-a)^{k+1}}{(k+1)!} + \frac{(-a)^{k+2}}{a(k+2)!}\right) & \text{if } a \le x < 2a \\
\qquad + \frac{1}{6a}\left(\frac{2a^{k+1}}{(k+1)!} - \frac{a^{k+2}}{a(k+2)!}\right) & \\
\qquad \left. + \frac{1}{6a}\left(\frac{x^{k+1}}{(k+1)!} - \frac{a^{k+1}}{(k+1)!}\right)\right] & \\
\left[\frac{1}{6a^2}\left(\frac{(-2a)^{k+2}}{(k+2)!} + 3a(\frac{(-2a)^{k+1}}{(k+1)!} - \frac{(-3a)^{k+2}}{(k+2)!} - 3a(\frac{(-3a)^{k+1}}{(k+1)!}\right)\right. & \\
\qquad + \frac{1}{6a}\left(\frac{(-a)^{k+1}}{(k+1)!} - \frac{(-2a)^{k+1}}{(k+1)!}\right) & \\
\qquad - \frac{1}{6a}\left(\frac{2(-a)^{k+1}}{(k+1)!} + \frac{(-a)^{k+2}}{a(k+2)!}\right) & \text{if } 2a \le x < 3a \\
\qquad + \frac{1}{6a}\left(\frac{2a^{k+1}}{(k+1)!} - \frac{a^{k+2}}{a(k+2)!}\right) & \\
\qquad + \frac{1}{6a}\left(\frac{(2a)^{k+1}}{(k+1)!} - \frac{a^{k+1}}{(k+1)!}\right) & \\
\qquad \left. + \frac{1}{6a^2}\left(\frac{-x^{k+2}}{(k+2)!} + \frac{3ax^{k+1}}{(k+1)!} + \frac{(2a)^{k+2}}{(k+2)!} - \frac{3a(2a)^{k+1}}{(k+1)!}\right)\right] & \\
\left[\frac{1}{6a^2}\left(\frac{(-2a)^{k+2}}{(k+2)!} + 3a(\frac{(-2a)^{k+1}}{(k+1)!} - \frac{(-3a)^{k+2}}{(k+2)!} - 3a(\frac{(-3a)^{k+1}}{(k+1)!}\right)\right. & \\
\qquad + \frac{1}{6a}\left(\frac{(-a)^{k+1}}{(k+1)!} - \frac{(-2a)^{k+1}}{(k+1)!}\right) & \\
\qquad - \frac{1}{6a}\left(\frac{2(-a)^{k+1}}{(k+1)!} + \frac{(-a)^{k+2}}{a(k+2)!}\right) & \text{if } 3a \le x \\
\qquad + \frac{1}{6a}\left(\frac{2a^{k+1}}{(k+1)!} - \frac{a^{k+2}}{a(k+2)!}\right) & \\
\qquad + \frac{1}{6a}\left(\frac{(2a)^{k+1}}{(k+1)!} - \frac{a^{k+1}}{(k+1)!}\right) & \\
\qquad \left. + \frac{1}{6a^2}\left(\frac{-(3a)^{k+2}}{(k+2)!} + \frac{3a(3a)^{k+1}}{(k+1)!} + \frac{(2a)^{k+2}}{(k+2)!} - \frac{3a(2a)^{k+1}}{(k+1)!}\right)\right] &
\end{cases}
$$

Sinusoidal Distribution (Widrow and Kollar [Widrow, Kollar (2008)], I.6, page 677)

For a sinusoidal random variable X with PDF and CDF specified by:

$$
f_X(x) = \begin{cases} \frac{1}{\pi\sqrt{A^2-x^2}} & \text{if } -a < x < a \\ 0 & \text{else} \end{cases}, \qquad F_X(x) = \begin{cases} 0 & \text{if } x \le -a \\ \frac{1}{2} + \frac{\arctan\left(\frac{x}{\sqrt{a^2-x^2}}\right)}{\pi} & \text{if } -a < x < a \\ 1 & \text{if } x \ge b \end{cases}
$$

with $a > 0$. For the sinusoidal distribution no simple closed-form of the k-th antiderivative can be given. However, most software with analytical capabilities can determine the solution for a given k value. Both Theorem 5.2.4 and 5.2.5 can be used as usual.

Convolutions of Uniform and Triangular Distributions (Widrow and Kollar [Widrow, Kollar (2008)], 19.6.3, page 501)

For X a random variable obtained by convoluting uniform and triangular random variables with PDF specified by

$$f_X(x) = \frac{1}{(b-a)(e-d)} \begin{cases} b-c & e > b-c+d \wedge b-e < x < c-d \\ (e-d)\frac{-2b+d+e+2x}{(b-c)} & \begin{cases} (e = b-c+d \wedge x = b-e) \\ \vee (d < e < b-c+d \wedge c-d < x \leq b-e) \end{cases} \\ \frac{(c+e+x-2b)(c-e-x)}{(b-c)} & \begin{cases} (e = b-c+d \wedge c-e < x < b-e) \\ \vee (e > b-c+d \wedge c-e < x \leq b-e) \\ \vee (d < e < b-c+d \wedge c-e < x \leq c-d) \end{cases} \\ \frac{(d+x-b)^2}{(b-c)} & \begin{cases} (e = b-c+d \wedge b-e < x < b-d) \\ \vee (e > b-c+d \wedge x = c-d) \vee \\ (e > b-c+d \wedge c-d < x < b-d) \vee \\ (d < e < b-c+d \wedge b-e < x < b-d) \end{cases} \\ 0 & \text{else} \end{cases}$$

$$+ \frac{1}{(b-a)(e-d)} \begin{cases} c-a & a > 0 \wedge e > -a+c+d \wedge c-e < x < a-d \\ (e-d)\frac{d+e+2x-2a}{(a-c)} & \begin{cases} (a > 0 \wedge e = -a+c+d \wedge x = c-e) \\ \vee (a > 0 \wedge d < e < -a+c+d \wedge a-d < x \leq c-e) \end{cases} \\ \frac{-(-a+e+x)^2}{(a-c)} & \begin{cases} (a > 0 \wedge e = -a+c+d \wedge a-e < x < c-e) \\ \vee (a > 0 \wedge e > -a+c+d \wedge a-e < x \leq c-e) \\ \vee (a > 0 \wedge d < e < -a+c+d \wedge a-e < x \leq a-d) \end{cases} \\ \frac{(2a-c-d-x)(c-d-x)}{(a-c)} & \begin{cases} (a > 0 \wedge e = -a+c+d \wedge c-e < x < c-d) \\ \vee (a > 0 \wedge e > -a+c+d \wedge a-d \leq x < c-d) \\ \vee (a > 0 \wedge d < e < -a+c+d \wedge c-e < x < c-d) \end{cases} \\ 0 & \text{else} \end{cases}$$

where $a < b$ are the uniform boundaries and $c \leq e \leq d$ the triangular parameters, respectively.

$$M_t(x) = F^{(-t)}(x)$$

$$= \frac{1}{(b-a)(e-d)} \begin{cases} b-c & e > b-c+d \wedge b-e < x < c-d \\ (e-d)\frac{-2b+d+e+\frac{2}{(k+1)!}x^{k+1}}{(b-c)} & \begin{cases} (e = b-c+d \wedge x = b-e) \vee \\ (d < e < b-c+d \wedge c-d < x \leq b-e) \end{cases} \\ \frac{1}{(b-c)} \times \left(\frac{(c^2 - ec + ec^2 - e^2 + 2be)}{(k+1)!} x^{k+1} + \frac{(2b-2e)}{(k+2)!} x^{k+2} + \frac{2}{(k+3)!} x^{k+3} \right) & \begin{cases} (e = b-c+d \wedge c-e < x < b-e) \vee \\ (e > b-c+d \wedge c-e < x \leq b-e) \vee \\ (d < e < b-c+d \wedge c-e < x \leq c-d) \end{cases} \\ \frac{\frac{2(d+x-b)^{k+3}}{(k+3)!}}{(b-c)} & \begin{cases} (e = b-c+d \wedge b-e < x < b-d) \vee \\ (e > b-c+d \wedge x = c-d) \vee \\ (e > b-c+d \wedge c-d < x < b-d) \vee \\ (d < e < b-c+d \wedge b-e < x < b-d) \end{cases} \\ 0 & \text{else} \end{cases}.$$

$$+\frac{1}{(b-a)(e-d)}\begin{cases} c-a & a>0\wedge e>-a+c+d\wedge c-e<x<a-d \\ (e-d)\frac{(d+e+2x-2a)^{k+2}}{(k+2)!(a-c)} & \begin{cases}(a>0\wedge e=-a+c+d\wedge x=c-e)\vee\\ (a>0\wedge d<e<-a+c+d\wedge a-d<x\leq c-e)\end{cases} \\ \frac{-2(-a+e+x)^{(}k+3)}{(k+3)!(a-c)} & \begin{cases}(a>0\wedge e=-a+c+d\wedge a-e<x<c-e)\vee\\ (a>0\wedge e>-a+c+d\wedge a-e<x\leq c-e)\vee\\ (a>0\wedge d<e<-a+c+d\wedge a-e<x\leq a-d)\end{cases} \\ \frac{1}{(a-c)}\times\Big(\frac{1}{(k+1)!}x^{k+1} & \begin{cases}(a>0\wedge e=-a+c+d\wedge c-e<x<c-d)\vee\\ (a>0\wedge e>-a+c+d\wedge a-d\leq x<c-d)\vee\\ (a>0\wedge d<e<-a+c+d\wedge c-e<x<c-d)\end{cases} \\ +(2d-2a)\frac{1}{(k+2)!}x^{k+2} & \\ +\frac{2}{(k+3)!}x^{k+3}\Big) & \\ 0 & \text{else}\end{cases}.$$

Convolutions of two Triangular Distributions (Widrow and Kollar [Widrow, Kollar (2008)], 19.6.4, page 502)

For X a random variable obtained by convoluting two triangular random variables with PDF specified by

$$f_X(x)=\frac{2}{3(b-a)(e-d)}\begin{cases} -\frac{(e-f)(-3a+e+2f+3x)}{(a-c)} & \begin{cases}(a>0\wedge f=a-c+e\wedge x=a-f)\vee\\ (a>0\wedge a-c+e<f<e\wedge a-f\leq x\leq c-e)\end{cases} \\ \frac{(a-c)(a+2c-3(e+x))}{(e-f)} & \begin{cases}(a>0\wedge f<a-c+e\wedge x=a-f)\vee\\ (a>0\wedge f<a-c+e\wedge c-e<x<a-f)\end{cases} \\ \frac{(a-e-x)^3}{(a-c)(e-f)} & \begin{cases}(a>0\wedge f=a-c+e\wedge a-e<x<a-f)\vee\\ (a>0\wedge a-c+e<f<e\wedge a-e<x<a-f)\vee\\ (c>a\wedge f<a-c+e\wedge a-e<x\leq c-e)\end{cases} \\ -\frac{(c-f-x)}{(a-b)(a-c)(d-e)(e-f)}\times & \begin{cases}(a>0\wedge f=a-c+e\wedge a-f<x<c-f)\vee\\ (a>0\wedge a-c+e<f<e\wedge c-e<x<c-f)\vee\\ (a>0\wedge f<a-c+e\wedge a-f<x<c-f)\end{cases} \\ \big(-2c^2+(3e-2f+x)c & \\ +3a(c-2e+f-x) & \\ +(3e-2f+x)(f+x)\big) & \\ 0 & \text{else}\end{cases}$$

$$+\frac{2}{3(b-a)(e-d)}\begin{cases} \frac{(a-c)(a+2c-3(d+x))}{(d-f)} & a>0\wedge f>-a+c+d\wedge c-f<x<a-d \\ \frac{(d-f)(-3a+d+2f+3x)}{(a-c)} & \begin{cases}(a>0\wedge f=-a+c+d\wedge x=c-f)\vee\\ (a>0\wedge d<f<-a+c+d\wedge a-d<x\leq c-f)\end{cases} \\ \frac{(a-3d+2f-x)(-a+f+x)^2}{(a-c)(d-f)} & \begin{cases}(a>0\wedge f=-a+c+d\wedge a-f<x<c-f)\vee\\ (a>0\wedge f>-a+c+d\wedge a-f<x\leq c-f)\vee\\ (a>0\wedge d<f<-a+c+d\wedge a-f<x\leq a-d)\end{cases} \\ -\frac{(3a-2c-d-x)(-c+d+x)^2}{(a-c)(d-f)} & \begin{cases}(a>0\wedge f=-a+c+d\wedge c-f<x<c-d)\vee\\ (a>0\wedge f>-a+c+d\wedge a-d\leq x<c-d)\vee\\ (a>0\wedge d<f<-a+c+d\wedge c-f<x<c-d)\end{cases} \\ 0 & \text{else}\end{cases}$$

$$
+\frac{2}{3(b-a)(e-d)}\begin{cases}
-\frac{(b-c)(b+2c-3(d+x))}{(d-f)} & f>b-c+d\wedge b-f<x<c-d\\
\frac{(d-f)(-3b+d+2f+3x)}{(b-c)} & \begin{cases}(f=b-c+d\wedge x=b-f)\vee\\(d<f<b-c+d\wedge c-d<x\le b-f)\end{cases}\\
\frac{c-f-x}{(b-c)(d-f)}\big(-2c^2+(3d-2f+x)c & \begin{cases}(f=b-c+d\wedge c-f<x<b-f)\vee\\(f>b-c+d\wedge c-f<x\le b-f)\vee\\(d<f<b-c+d\wedge c-f<x\le c-d)\end{cases}\\
\quad+3b(c-2d+f-x) & \\
\quad+(3d-2f+x)(f+x)\big) & \\
-\frac{(b-d-x)^3}{(b-c)(d-f)} & \begin{cases}(f=b-c+d\wedge b-f<x<b-d)\vee\\(f>b-c+d\wedge x=c-d)\vee\\(f>b-c+d\wedge c-d<x<b-d)\vee\\(d<f<b-c+d\wedge b-f<x<b-d)\end{cases}\\
0 & \text{else}
\end{cases}
$$

$$
+\frac{2}{3(b-a)(e-d)}\begin{cases}
-\frac{(b-c)(b+2c-3(e+x))}{(e-f)} & f<-b+c+e\wedge b-e<x\le c-f\\
-\frac{2(e-f)(-3b+e+2f+3x)}{(b-c)} & \begin{cases}(f=-b+c+e\wedge x=c-f)\vee\\(-b+c+e<f<e\wedge c-f\le x\le b-e)\end{cases}\\
\frac{(3b-2c-e-x)(-c+e+x)^2}{(b-c)(e-f)} & \begin{cases}(f=-b+c+e\wedge c-e<x<c-f)\vee\\(-b+c+e<f<e\wedge c-e<x<c-f)\vee\\(f<-b+c+e\wedge c-e<x\le b-e)\end{cases}\\
-\frac{2(b-3e+2f-x)(-b+f+x)^2}{3(a-b)(b-c)(d-e)(e-f)} & \begin{cases}(f=-b+c+e\wedge c-f<x<b-f)\vee\\(-b+c+e<f<e\wedge b-e<x<b-f)\vee\\(f<-b+c+e\wedge c-f<x<b-f)\end{cases}\\
0 & \text{else}
\end{cases}
$$

for $-\infty < d \le f \le e < \infty$ and $-\infty < a \le c \le b < \infty$, the moments are the same as those in Section 8 of [Csordas et al. (2003)].

$$M_k(x) = F^{(-k)}(x) = A + B$$

$A =$

$$
\frac{2}{3(b-a)(e-d)}
\begin{cases}
-\frac{(e-f)(-3a+e+2f+3x)^{(k+2)}}{(k+2)!(a-c)} & \begin{cases}(a>0 \wedge f=a-c+e \wedge x=a-f)\vee\\ (a>0 \wedge a-c+e<f<e \wedge a-f\le x\le c-e)\end{cases}\\
\frac{(a-c)(a+2c-3(e+x))^{(k+2)}}{(k+2)!(e-f)} & \begin{cases}(a>0 \wedge f<a-c+e \wedge x=a-f)\vee\\ (a>0 \wedge f<a-c+e \wedge c-e<x<a-f)\end{cases}\\
\frac{6(a-e-x)^{(k+4)}}{(k+4)!(a-c)(e-f)} & \begin{cases}(a>0 \wedge f=a-c+e \wedge a-e<x<a-f)\vee\\ (a>0 \wedge a-c+e<f<e \wedge a-e<x<a-f)\vee\\ (c>a \wedge f<a-c+e \wedge a-e<x\le c-e)\end{cases}\\
\left(-\frac{1}{(a-b)(a-c)(d-e)(e-f)}\right)\times & \\
\Big((3ac^2-6eac-3af^2+6eaf & \\
-2c^3+3ec^2+2f^3-3ef^2) & \begin{cases}(a>0 \wedge f=a-c+e \wedge a-f<x<c-f)\vee\\ (a>0 \wedge a-c+e<f<e \wedge c-e<x<c-f)\vee\\ (a>0 \wedge f<a-c+e \wedge a-f<x<c-f)\end{cases}\\
\times\frac{x^{(k+1)}}{(k+1)!}(6ef+3f^2+3c^2 & \\
+6ea-6ac)\frac{x^{(k+2)}}{(k+2)!} & \\
+2(3a-3e)\frac{x^{(k+3)}}{(k+3)!}-6\frac{x^{(k+4)}}{(k+4)!}\Big) & \\
0 & \text{else}
\end{cases}
$$

$$
+\frac{2}{3(b-a)(e-d)}
\begin{cases}
\frac{(a-c)(a+2c-3(d+x))^{(k+2)}}{(k+2)!(d-f)} & a>0 \wedge f>-a+c+d \wedge c-f<x<a-d\\
\frac{(d-f)(-3a+d+2f+3x)^{(k+2)}}{(k+2)!(a-c)} & \begin{cases}(a>0 \wedge f=-a+c+d \wedge x=c-f)\vee\\ (a>0 \wedge d<f<-a+c+d \wedge a-d<x\le c-f)\end{cases}\\
\frac{1}{(a-c)(d-f)} & \\
\times\Big((a^3-3a^2d+6adf & \\
-3af^2-3df^2+2f^3) & \\
\times\frac{x^{(k+1)}}{(k+1)!} & \begin{cases}(a>0 \wedge f=-a+c+d \wedge a-f<x<c-f)\vee\\ (a>0 \wedge f>-a+c+d \wedge a-f<x\le c-f)\vee\\ (a>0 \wedge d<f<-a+c+d \wedge a-f<x\le a-d)\end{cases}\\
+(6ad-3a^2-6df)\frac{x^{(k+2)}}{(k+2)!} & \\
+2(3a-3d)\frac{x^{(k+3)}}{(k+3)!} & \\
-\frac{6x^{(k+4)}}{(k+4)!}\Big) & \\
-\frac{1}{(a-c)(d-f)} & \\
\times\Big((3ac^2-6acd+3ad^2 & \\
-2c^3+3c^2d-d^3) & \\
\times\frac{x^{(k+1)}}{(k+1)!} & \begin{cases}(a>0 \wedge f=-a+c+d \wedge c-f<x<c-d)\vee\\ (a>0 \wedge f>-a+c+d \wedge a-d\le x<c-d)\vee\\ (a>0 \wedge d<f<-a+c+d \wedge c-f<x<c-d)\end{cases}\\
+(-6ac+6ad+3c^2-3d^2)\frac{x^{(k+2)}}{(k+2)!} & \\
+2(3a-3d)\frac{x^{(k+3)}}{(k+3)!} & \\
-\frac{6x^{(k+4)}}{(k+4)!}\Big) & \\
0 & \text{else}
\end{cases}
$$

$$
\begin{aligned}
B = \\
&+\frac{2}{3(b-a)(e-d)}
\begin{cases}
-\frac{(b-c)(b+2c-3(d+x))^{(k+2)}}{(k+2)!(d-f)} & f > b-c+d \wedge b-f < x < c-d \\
\frac{(d-f)(-3b+d+2f+3x)^{(k+2)}}{(k+2)!(b-c)} & \begin{cases}(f = b-c+d \wedge x = b-f) \vee \\ (d < f < b-c+d \wedge c-d < x \leq b-f)\end{cases} \\
\frac{1}{(b-c)(d-f)} & \\
\times \Big((3bc^2 - 6bcd + 6bdf - 3bf^2 & \\
-2c^3 + 3c^2 d - 3df^2 + 2f^3) & \\
\times \frac{x^{(k+1)}}{(k+1)!} & \begin{cases}(f = b-c+d \wedge c-f < x < b-f) \vee \\ (f > b-c+d \wedge c-f < x \leq b-f) \vee \\ (d < f < b-c+d \wedge c-f < x \leq c-d)\end{cases} \\
+(6bd - 6bc + 3c^2 - 6df + 3f^2)\frac{x^{(k+2)}}{(k+2)!} & \\
+2(3b-3d)\frac{x^{(k+3)}}{(k+3)!} & \\
-\frac{6x^{(k+4)}}{(k+4)!}\Big) & \\
-\frac{6(b-d-x)^{(k+4)}}{(k+4)!(b-c)(d-f)} & \begin{cases}(f = b-c+d \wedge b-f < x < b-d) \vee \\ (f > b-c+d \wedge x = c-d) \vee \\ (f > b-c+d \wedge c-d < x < b-d) \vee \\ (d < f < b-c+d \wedge b-f < x < b-d)\end{cases} \\
0 & \text{else}
\end{cases} \\
&+\frac{2}{3(b-a)(e-d)}
\begin{cases}
-\frac{(b-c)(b+2c-3(e+x))^{(k+2)}}{(k+2)!(e-f)} & f < -b+c+e \wedge b-e < x \leq c-f \\
-\frac{2(e-f)(-3b+e+2f+3x)^{(k+2)}}{(k+2)!(b-c)} & \begin{cases}(f = -b+c+e \wedge x = c-f) \vee \\ (-b+c+e < f < e \wedge c-f \leq x \leq b-e)\end{cases} \\
\frac{1}{(b-c)(e-f)} & \\
\times \Big((3bc^2 - 6ebc + 3e^2 b - 2c^3 + 3ec^2 - e^3) & \\
\times \frac{x^{(k+1)}}{(k+1)!} & \begin{cases}(f = -b+c+e \wedge c-e < x < c-f) \vee \\ (-b+c+e < f < e \wedge c-e < x < c-f) \vee \\ (f < -b+c+e \wedge c-e < x \leq b-e)\end{cases} \\
+(6eb - 6bc + 3c^2 - 3e^2)\frac{x^{(k+2)}}{(k+2)!} & \\
+(3b-3e)\frac{x^{(k+3)}}{(k+3)!} & \\
-6\frac{x^{(k+4)}}{(k+4)!}\Big) & \\
-\frac{1}{3(a-b)(b-c)(d-e)(e-f)} & \\
\times \Big((2b^3 - 6eb^2 - 6bf^2 + 12ebf + 4f^3 - 6ef^2) & \\
\times \frac{x^{(k+1)}}{(k+1)!} & \begin{cases}(f = -b+c+e \wedge c-f < x < b-f) \vee \\ (-b+c+e < f < e \wedge b-e < x < b-f) \vee \\ (f < -b+c+e \wedge c-f < x < b-f)\end{cases} \\
+(12eb - 6b^2 + 12eb + 6f^2 - 12ef)\frac{x^{(k+2)}}{(k+2)!} & \\
+(6b-6e)\frac{x^{(k+3)}}{(k+3)!} & \\
+12\frac{x^{(k+4)}}{(k+4)!}\Big) & \\
0 & \text{else}
\end{cases}
\end{aligned}
$$

C.1.3 Supplementary Plots for $X - \lfloor X \rfloor$

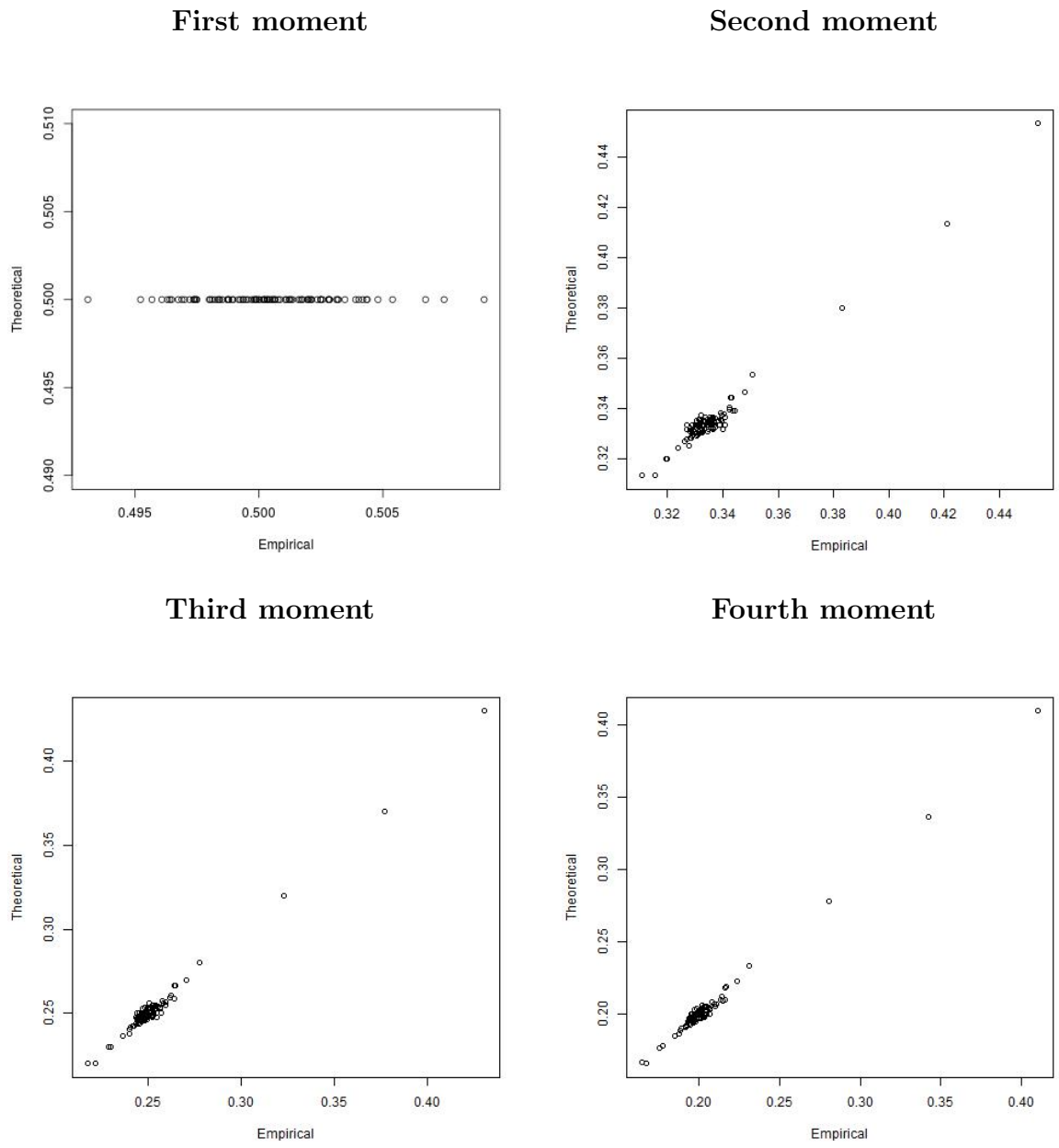


Figure C.1: First four moments of $X - \lfloor X \rfloor$ of the uniform distribution, with parameters $-a = b$ for $a = 0.1, 0.2, \ldots, 10$.

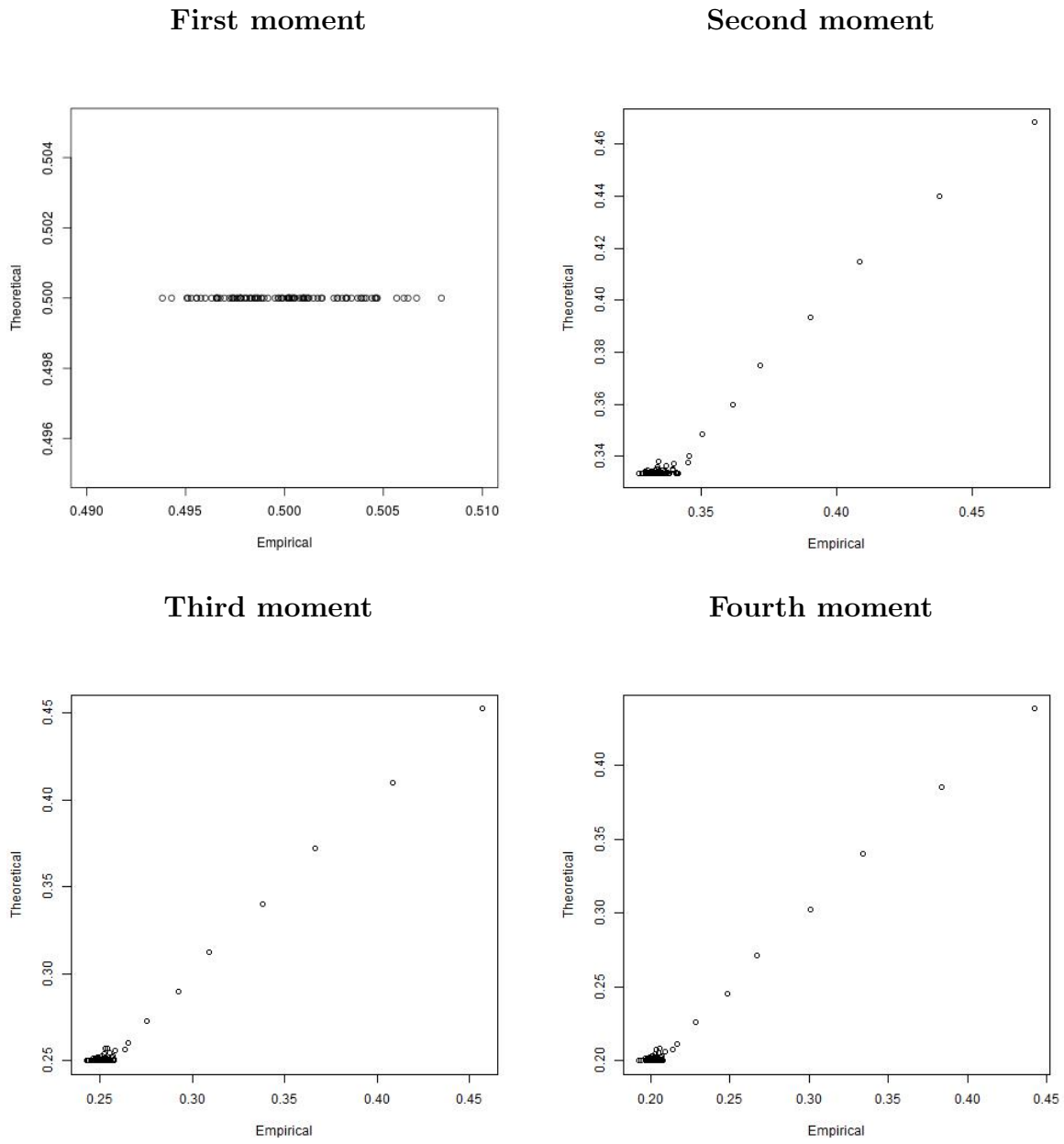


Figure C.2: First four moments of $X - \lfloor X \rfloor$ of the triangular distribution, with parameters $c = 0, -a = b$ for $a = 0.1, 0.2, \ldots, 10$.

C.1.4 Supplementary Plots for $X - \lfloor X + \frac{1}{2} \rfloor$

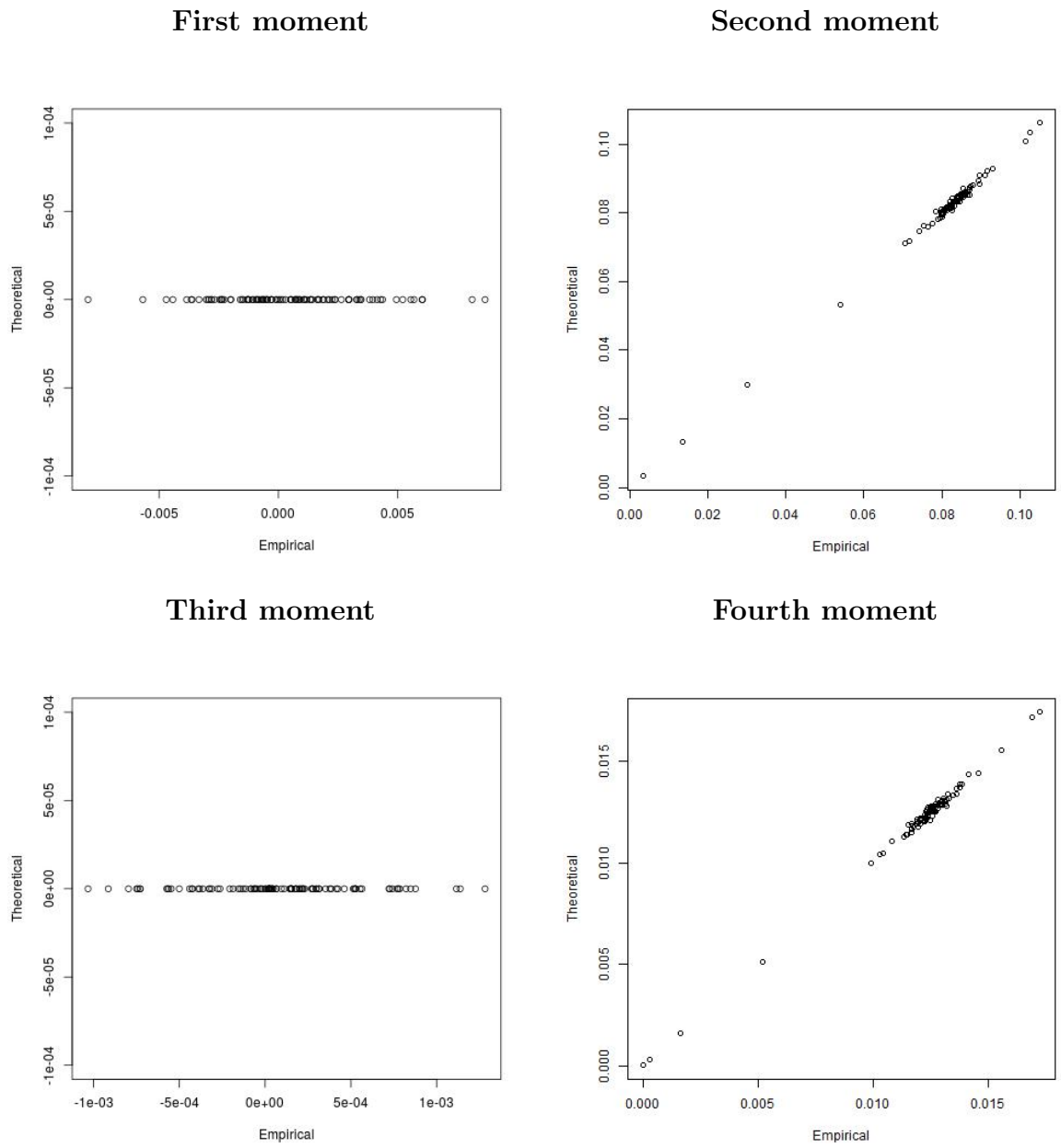


Figure C.3: First four moments of $X - \lfloor X + \frac{1}{2} \rfloor$ of the uniform distribution, with parameters $-a = b$ for $a = 0.1, 0.2, \ldots, 10$.

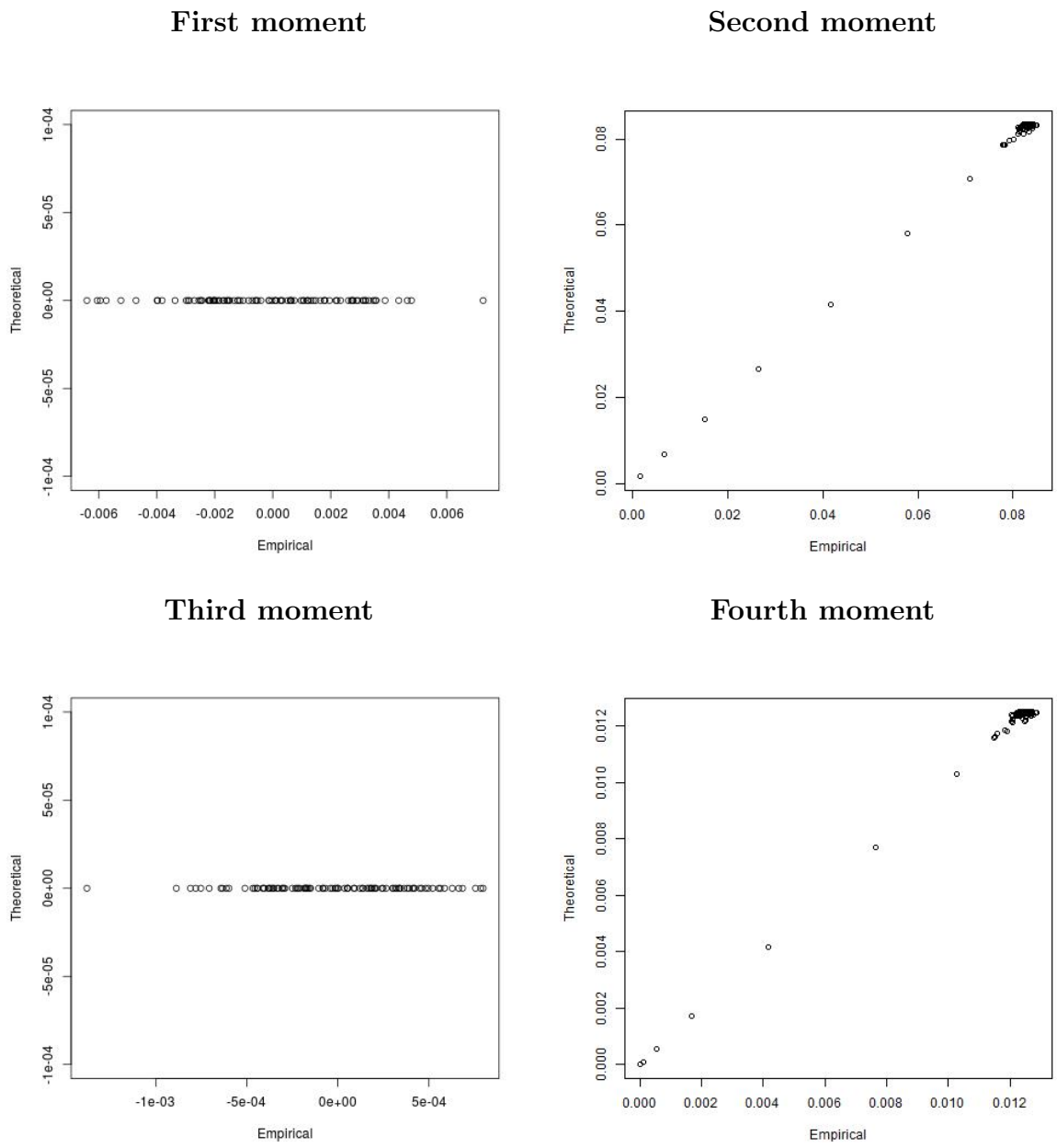


Figure C.4: First four moments of $X - \left\lfloor X + \frac{1}{2} \right\rfloor$ of the triangular distribution, with parameters $c = 0, -a = b$ for $a = 0.1, 0.2, \ldots, 10$.

C.1.5 Supplementary Tables for $X - \lfloor X \rfloor$

a	k	1	2	3	4
0.1		5.900041e-08	2.340302e-07	3.294760e-07	4.138349e-07
1		8.992801e-06	1.018269e-05	9.367884e-06	8.361929e-06
2		2.610486e-06	2.310805e-06	1.476468e-06	8.137844e-07
3		1.240942e-05	7.234127e-06	4.356404e-06	3.142111e-06
4		1.130977e-07	4.766217e-06	7.697295e-06	7.981190e-06
5		1.500870e-06	8.881806e-07	4.367888e-07	2.420640e-07
6		4.756723e-05	4.185227e-05	3.364696e-05	2.674131e-05
7		4.012009e-06	3.343534e-06	3.480836e-06	3.739969e-06
8		3.143884e-06	3.913011e-06	2.297650e-06	1.124660e-06

Table C.1: ASE of the empirical and theoretical values for the uniform distribution.

a	k	1	2	3	4
0.1		2.175290e-05	2.027581e-05	1.796827e-05	1.586509e-05
1		1.188664e-05	9.232077e-06	6.423183e-06	4.522853e-06
2		2.291893e-06	1.470480e-06	1.168993e-06	1.098933e-06
3		2.512014e-07	8.698671e-09	1.147041e-08	5.433561e-08
4		9.296816e-07	1.047689e-06	9.158490e-07	6.850873e-07
5		1.058201e-05	6.332940e-06	3.538913e-06	2.173855e-06
6		1.148803e-05	1.189744e-05	1.022145e-05	8.669491e-06
7		1.789967e-05	4.095275e-05	5.251031e-05	5.476888e-05
8		7.053805e-06	9.674796e-06	9.798778e-06	8.776999e-06

Table C.2: ASE of the empirical and theoretical values for the triangular distribution.

C.1.6 Supplementary Tables for $X - \lfloor X + \frac{1}{2} \rfloor$

a	k	1	2	3	4
0.1		2.592975e-07	2.208311e-09	1.219367e-11	1.742729e-13
1		6.867934e-07	1.472720e-08	3.942234e-08	3.159564e-09
2		3.571091e-06	3.178441e-09	1.243143e-07	5.069440e-11
3		4.005817e-07	3.658153e-07	2.387804e-08	1.567003e-08
4		8.359841e-08	2.490602e-07	2.355163e-09	4.251432e-08
5		1.188630e-06	7.675209e-08	1.423636e-07	1.764381e-08
6		5.980640e-07	5.767026e-08	1.649205e-09	3.222833e-09
7		1.957862e-05	2.150672e-09	1.065707e-06	1.170508e-08
8		7.498646e-05	1.437785e-08	1.293426e-06	8.886361e-10

Table C.3: ASE of the empirical and theoretical values for the uniform distribution.

a	k	1	2	3	4
0.1		1.565134e-08	4.913611e-10	3.399670e-13	3.169243e-14
1		2.221216e-05	5.532335e-07	7.866869e-07	1.350012e-08
2		3.734955e-07	7.588791e-07	6.265540e-08	2.387643e-08
3		4.633213e-06	2.142906e-08	3.093012e-08	2.244004e-10
4		2.193109e-06	9.394391e-07	4.695382e-08	4.935507e-08
5		2.607279e-07	1.200193e-08	3.504874e-08	3.221698e-09
6		1.133684e-05	1.881535e-09	2.081667e-07	2.630884e-10
7		1.451247e-06	3.338605e-07	2.714815e-07	4.224998e-13
8		3.292962e-05	1.449909e-07	5.660277e-07	1.530374e-09

Table C.4: ASE of the empirical and theoretical values for the triangular distribution.