Maynooth University
National University of Ireland Maynooth

The Nuances of Quantisation

A dissertation submitted for the degree of
Doctor of Philosophy

By:

Sonya Leech

Under the supervision of:

Professor David Malone

Dr Jonathan Dunne

Hamilton Institute
National University of Ireland Maynooth
Ollscoil na hÉireann, Má Nuad

DATE Jan 14th, 2026

Contents

List of Tables

Declaration

I hereby declare that I have produced this manuscript without the prohibited assistance of any third parties and without making use of aids other than those specified.

The thesis work was conducted from [Oct 2020] to [Jan 2026] under the supervision of [Prof David Malone & Dr.Jonathan Dunne] in Hamilton Institute, National University of Ireland Maynooth.

Maynooth, Ireland,

[Jan 14th, 2026]

Acknowledgement

First, I would like to express my deepest gratitude to my professor David Malone and supervisor Dr. Jonathan Dunne, who have both been a tremendous support over the past few years. Throughout this programme they have both been very supportive. Their guidance helped steer my research, and their understanding during demanding times made an enormous difference to me. In the most difficult moments, their compassion, patience, and encouragement gave me the strength to keep going. I also extend my thanks to IBM for their support and contributions toward funding this research. I would also like to extend a further hand of gratitude to Jonathan as he was the one who first encouraged me to undertake this journey. Without his guidance, support, and technical mentorship, I doubt I would ever have begun this program.

My deepest thanks go to my fiancé, Glenn, who has supported me with unwavering patience and love over the past five years. He has remained very understanding and steadfastly supportive throughout this journey. I am very grateful for his strength and for always standing by my side. To my dad, who has always encouraged me and has been very supportive throughout this journey, I will be forever grateful. To my mother, who has never received the full attention she deserved while I have been absorbed in this program, I look forward to spending real, uninterrupted quality time with you. My heartfelt thanks also go to the rest of my family and friends, who have been incredibly supportive throughout this journey. I am truly grateful for each of you.

Abstract

Truncating data to speed up process flows or reduce storage space is quite common in organisations. In many applications, the decimal precision of numerical data is considered of limited importance, particularly when precision extends beyond the commonly used two decimal places. However, this assumption can introduce significant challenges when modelling complex, high-volume production data. This thesis is motivated by practical difficulties encountered in modelling Electronic Data Interchange (EDI) production data within a supply chain organisation.

EDI data is bursty in nature, presenting challenges for modelling, therefore this work develops a structured framework for modelling EDI data. It investigates different approaches to identifying quantisation, ways to overcome quantisation challenges, and what impact, if any, it has on modelling applications. Finally, the research explores techniques for recovering information lost through rounding, addressing the problem from both a data point reconstruction perspective and a distributional recovery perspective.

List of Acronyms

Acronym	**Meaning**
AD	Anderson-Darling
CDF	Cumulative Distribution Function
CvM	Cramér-von Mises
GoF	Goodness of Fit
MLE	Maximum Likelihood Estimation
MME	Method of Moments Estimation
PDF	Probability Density Function
RMSE	Root Mean Square Error
NRMSE	Normalised Root Mean Square Error
EDI	Electronic Data Interchange

CHAPTER 1

Introduction

In this chapter, I discuss the motivations behind the work of the thesis and provide an overview of the material presented in the following chapters.

1.1 Motivation

Queuing systems are common across industries, including manufacturing, healthcare, telecommunications, finance, and cloud computing. Many business operations involve some form of queuing, whether handling customer service requests, managing transaction flows, or processing data pipelines. These systems directly impact throughput, responsiveness, and user experience. Marginal performance gains can lead to significant reductions in both quantitative costs, such as resource utilisation and operational expenditure, while qualitative costs, such as improved user satisfaction and reduced service quality, motivate the study of optimisation research.

Optimisation is a fundamental concept across many business domains, particularly in the context of queuing systems. Underperforming queue applications can lead to issues such as latency, resource starvation, performance inconsistencies, and scalability bottlenecks. Addressing these challenges through optimisation strategies improves overall system efficiency and responsiveness.

The research initially focused on optimisation strategies to improve queue behaviour, particularly around Electronic Data Interchange (EDI) messages.

These electronic transaction messages between customers and trading partners exhibit bursty behaviour, where a single inbound message can trigger a cascade of outbound messages. The cascading messages result in high variability and potential queue congestion, making EDI traffic an excellent case study for queue optimisation strategies. The volatility and heterogeneity of EDI data make it particularly well-suited for this research.

In the early stages of the research, I observed that timestamps in the dataset were quantised to fixed levels of precision. Quantisation masks the continuous nature of the data, introducing discreteness that hinders accurate modelling, distribution fitting and simulation. As a result, parametric models struggle to capture the true characteristics of interarrival and service times, making it difficult to model or simulate system behaviour reliably.

As the research progressed, it became evident that reliable optimisation of queuing systems depends fundamentally on the accuracy of the underlying statistical models. Quantisation introduced distortions that significantly affected distribution fitting, parameter estimation, and goodness-of-fit testing, thereby limiting the reliability of optimisation and simulation efforts. Consequently, this thesis focuses on understanding and mitigating the effects of quantisation as a prerequisite for accurate queue modelling and optimisation.

The research sought to investigate techniques that can reverse this masking effect to recover the true underlying structure of the data. The thesis makes a novel contribution in this area by examining the impact of quantised data on model fitting and proposing strategies to unround the data and improve fitting accuracy in the presence of rounding. Additionally, this research extends previous work by developing a framework for modelling bursty and heterogeneous EDI messaging systems, which are inherently complex and not comprehensively addressed in prior research.

Although this research is motivated by EDI messaging systems, quantisation is pervasive across many modern data-driven applications, including IoT sensor networks, financial transaction systems, healthcare monitoring platforms, and large-scale logging infrastructures. In such systems, timestamps and measurements are frequently rounded or truncated to reduce storage, improve transmission efficiency, or standardise reporting formats. Consequently, the challenges examined in this thesis extend beyond EDI environments and are

relevant to a broader class of statistical modelling and simulation problems involving quantised data.

1.2 Research Questions

This thesis investigates the impact of quantisation on distribution modelling, parameter estimation, and goodness-of-fit (GoF) testing within queuing systems. The research is guided by the following questions:

- How does quantisation affect distribution fitting, parameter estimation, and GoF testing in real-world queuing data?
- To what extent can quantisation-induced fitting errors be mitigated through modelling, transformation, and unrounding techniques?
- Can the underlying continuous distribution of quantised observations be reliably reconstructed?
- Which parameter estimation techniques are most robust when applied to quantised data?

1.3 Contributions of the Thesis

The primary contributions of this thesis are summarised as follows:

- Development of a framework for modelling heterogeneous and bursty EDI message traffic.
- Analysis of the impact of quantisation on GoF testing, particularly the Anderson-Darling test.
- Comparative evaluation of Maximum Likelihood Estimation (MLE) and Method of Moments Estimation (MME) under quantised conditions.
- Investigation of convergence and fitting issues arising from quantised and heavy-tailed data.
- Development and evaluation of unrounding techniques for reconstructing continuous distributions from quantised observations.
- Proposal of an interval-based estimation approach for modelling rounded data directly.

1.4 Overview

The thesis is divided into the following chapters:

Chapter 1: Introduction
The Introduction chapter will contain a broad discussion around the context of the work, which is the motivation behind this research.

Chapter 2: Literature Review
Chapter 2 will cover the literature review, including background information and related research.

Chapter 3: Where We Came From
Chapter 3 investigates EDI message traffic within a large-scale business-to-business (B2B) supply chain network, with a focus on modelling interarrival and service times for queuing applications. By parsing system log files, the chapter reconstructs the traversal paths of EDI messages from input to output, revealing the traversal complexity of message flow. It further investigates how messages propagate through the B2B orchestration layer, highlighting the schematic complexities, temporal characteristics, and operational load of the messages on the network. The chapter examines the challenges posed by bursty EDI message traffic, where a single inbound message can result in numerous downstream outputs. These outputs often conform to multiple, distinct message schemas, such as invoices, purchase orders, and acknowledgements, each defining different structural and semantic attributes. Such heterogeneity complicates the extraction and interpretation of message information, as analytical models must accommodate diverse schema types and varying message behaviours. In addition to message complexity, this chapter addresses quantisation challenges and heavy-tailed behaviour. The analysis applies partitioning strategies to the messages to improve distribution fitting accuracy, thereby improving the accuracy of distribution GoF methods.

Chapter 4: Heterogeneous Message Modelling
Chapter 4 builds on the insights gained from the analysis of EDI messaging complexities, including message independence, bursty traffic behaviour, schematic diversity, and the associated challenges of distribution fitting. Chapter 4 proposes a modelling framework that partitions EDI messages by different schematic attributes and distributional characteristics (e.g. head, tail), thereby

improving fitting accuracy when applying both parametric and non-parametric fitting techniques to the messages. The framework aims to improve the fitting accuracy of distribution modelling of the interarrival and service times. While the framework captures the distributional behaviour of these bursty, heterogeneous message traffic, it also highlights the trade-offs between different modelling approaches when evaluating GoF methods in the presence of quantisation, burstyness and heavy-tailed data characteristics.

Chapter 5: Convergence and GoF Issues in Distribution Modelling
Fitting errors observed in GoF tests discussed in previous chapters, particularly with the Anderson-Darling (AD) GoF test, motivated Chapter 5's investigation into the relationship between data characteristics and the occurrence of AD GoF fitting errors. The chapter explores distribution fitting when applied to non-ideal data conditions relative to the distributions modelled. It presents specific cases in which AD GoF tests fail and investigates the underlying causes of these failures. The analysis will support subsequent chapters that involve fitting data to parametric distribution models.

Chapter 6: Rounding Effects On Parameter Estimation
As quantised data poses fitting challenges in distribution modelling, Chapter 6 investigates the impact of quantisation on parameter estimation techniques, particularly Maximum Likelihood Estimation (MLE) and Method of Moments Estimation (MME) for probability density modelling. Building on prior literature, the chapter aims to identify which parameter estimation methods are less affected by quantisation, yielding parameter estimates that most closely approximate those of the underlying distribution. It also seeks to determine the conditions under which these findings hold. The analysis compares the influence of quantisation on parameter estimates and examines how those estimates impact GoF fitting accuracy. It also explores how variations in parameter values applied to quantised data influence the outcome of MLE and MME. Through these comparisons, between estimation methods and between quantised and original data, the chapter provides a comprehensive understanding of the limitations and sensitivities of parameter estimation techniques when applied to quantised data.

Chapter 7: Challenges Fitting To Rounded Data
Building on the research from previous chapters, this chapter looks at the

types of fitting errors and then investigates approaches to mitigate errors arising from data characteristics that violate the distributional assumptions of the fitted models or the assumptions underlying specific GoF tests. The primary focus is on the log-normal distribution, applying and evaluating a set of mitigation techniques to mitigate convergence and improve GoF fitting on quantised data. While this study considers other distributions, the emphasis remains on the log-normal distribution to demonstrate the effectiveness of the proposed methods.

Chapter 8: UnRounding The Data

Using the experiences gained in applying simple techniques to mitigate the fitting errors both for distributional modelling and GoF testing, this chapter focuses on implementing a range of techniques to restore quantised data to a form that more closely approximates its original, continuous distribution. The primary objective is to unround the data by reconstructing its underlying shape, with an emphasis on point estimation rather than interval estimation. Using synthetic datasets, the empirical shape of the quantised data is compared against known continuous distributions to validate model selection and parameter estimates. Several methods are studied, including various forms of jitter to introduce random noise and various types of rejection sampling to regenerate data points. These techniques offer a practical toolbox for mitigating the effects of quantisation while maintaining statistical accuracy.

Chapter 9: Interval-Based Approach for Estimation

Chapter 9 presents an alternative approach to handling quantised data by incorporating rounding directly into the distribution fitting process rather than attempting to reverse it. By applying a simplified CDF integration technique, this chapter treats observed values as interval-censored rather than exact and estimates the likelihood by integrating over the plausible range of the CDF bounds. It offers an alternative approach to data correction, building on and extending ideas introduced in earlier chapters.

1.5 Publications

Over the lifetime of this research, the following peer-reviewed conference papers, journal and poster sessions have been presented. Papers are grouped by thesis chapter for ease of reference.

Chapter 3: Where We Came From

- Heads or tails: A framework to model supply chain heterogeneous messages (2021 30th Conference of Open Innovations Association FRUCT, 2021)

Chapter 4: Heterogeneous Message Modelling

- A framework to model bursty electronic data interchange messages for queuing systems (Future Internet MDPI, 2022)

Chapter 5: Convergence and GoF Issues in Distribution Modelling

- Lost In Rounding: How Small Data Adjustments Create Statistical Problems For MLE And MME (35th Irish Signals and Systems Conference, 2025)

Chapter 6: Rounding Effects On Parameter Estimation

- Lost In Rounding: How Small Data Adjustments Create Statistical Problems For MLE And MME (35th Irish Signals and Systems Conference, 2025)

Chapter 7: Challenges Fitting To Rounded Data

- Log-normal distribution modelling with quantised data (34th Irish Signals and Systems Conference, 2023)

1.6 Symbols

To assist the reader in understanding the notation used throughout the thesis, the table of symbols is provided in Table 1.1.

Table 1.1: Symbols.

Symbol	**Greek Name**	**Meaning**
e	–	Exponential constant.
erf	–	Error function; used in probability and Normal distributions.
λ	Lambda	Mean event rate (Poisson/exponential); also Weibull scale parameter.
μ	Mu	Mean or expected value of a distribution.
x	–	A value or observation from variable X.
π	Pi	Ratio of the circumference of any circle to the diameter of that circle.
P	–	Probability of an event (between 0 and 1).
N	–	Sample size or number of observations.
σ	Sigma	Standard deviation; measures data spread.
$\surd$	–	Square root.
σ^2	Sigma	Variance; square of the standard deviation.
X	–	Random variable.
∂	–	Partial derivative; change with respect to one variable.
Γ	Gamma	Gamma function; generalisation of the factorial function.
γ	Gamma	Shift parameter in distributions.
γ_1	Gamma	Skewness; measures distribution asymmetry.
γ_2	Gamma	Kurtosis; measures heaviness of tails.
β	Beta	Weibull shape parameter.

CHAPTER 2

Literature Review

In this chapter, a review of the background-related literature is conducted in the field of bursty EDI messages, data quantisation, the effects of rounding on fitting models, parameter estimation methods, and different strategies to unround the quantised data.

2.1 Introduction

The present chapter provides a structured overview of the relevant literature across several domains central to the thesis. These include statistical approaches to modelling queue-based systems, the dynamics of bursty and schema-rich EDI messages, and the challenges posed by quantisation in real-world datasets where continuous-valued timestamps are rounded to discrete values, introducing uncertainty that degrades the accuracy of parametric modelling and simulation. EDI messages are central to this research, as they arrive as single, aggregated payloads that unpack into numerous sub-messages in rapid succession. These messages often contain multiple heterogeneous XML schemas within a single transmission, presenting a mix of structured and unstructured content with low uniformity. The chapter also examines various techniques for reversing data rounding. Methods for distribution fitting under imperfect data conditions will be analysed, including MLE and MME. Hypothesis testing, such as the AD and CvM tests, will also be reviewed. These methods will help quantify and evaluate model assumptions.

The chapter will outline the limitations in the existing literature concerning convergence, parameter sensitivity, and statistical robustness when faced with heavily rounded datasets. Building on this foundation, the thesis contributes novel strategies for unrounding quantised data and stabilising distribution fitting whilst improving fitting accuracy.

Several sections in this thesis are based on the author's previously published work [1]

2.2 Quantisation

Quantisation is defined as *"The division of a quantity into a discrete number of small parts"* [2]. It is typically regarded as a set of values rounded to whole multiples of a fixed quantity. For example, when (3.1,3.2,3.3,3.4) is rounded to integers, it becomes (3,3,3,3); thus, it has been transformed into a discrete set of values. Quantisation can be applied to various data types, including floating-point numbers, integers, and timestamps, and is used in many fields such as image processing [3], colour palettes [4], speech coding [5] and signals and systems [6].

2.2.1 Benefits of Quantisation

Quantisation offers benefits. In Neural Networks, millions of computational operations occur during training, and using smaller data types can lead to notable improvements in performance and efficiency [7]. Converting Neural Network model weights from 32-bit floating-point to 8-bit signed integers can reduce memory requirements [8]. Quantisation can also enhance storage efficiency when binary floating-point numbers are transformed into integers, decreasing the disk space needed for data logging.

2.2.2 Challenges and Errors

Quantisation also has implications. Rounding the timestamp can affect the chronological order of events. Site Reliability Engineers (SREs) may find it challenging to determine the exact sequence of events during an outage [9].

Performance engineers may struggle to provide precise performance readings. A fraction of a second might seem insignificant to one domain, but for another, it could be crucial, especially when managing large volumes of data.

A timestamp resolution of 1 ms might have been sufficient 20 years ago when a logging standard was set, but it creates problems today because jobs run faster and can finish in less than a millisecond. Quantisation can interfere with downstream modelling as Data Scientists may struggle to fit models to quantised data. Quantising values alters distribution shapes, thereby changing their moments [10].

Attempting to perform survival analysis (expected duration of time until an event occurs) [11] might be hampered if typical durations are similar to quantisation error. If a value x is quantised to $\hat{x}$, then the quantisation error is defined as: $\epsilon = x - \hat{x}$. The quantisation error is limited by half the quantisation interval when rounding to the nearest quantised value. For example, if rounding to the nearest integer, the quantisation error falls within the interval $[-0.5, +0.5]$.

Quantisation errors occur when rounded values diverge from intended values and can affect the fit of a model to a specific distribution [12]. For example, one might model high-frequency points of a dataset. If raw values at the start of the data are between zero and one second and are rounded to a whole number, then the resulting values will resemble a Bernoulli distribution.

If the data is within the first ten seconds, rounded values might resemble a Poisson distribution. Neither of these distributions is close to the original [13]. Quantisation errors can be problematic for fitting [14]. When events appear to have occurred at the same time due to quantisation, the distribution may look discrete when, in fact, it is continuous.

2.2.3 Impact on Statistical Modelling

Quantisation may round values to numbers outside the range of a given distribution (e.g., to zero), causing fitting challenges. Values rounded to zero can be problematic because the PDF, for example, of a log-normal distribution, is zero at zero, effectively indicating a zero probability that the sample came from a log-normal distribution, as zero is outside the support of the given distribution. These zero values conflict with the assumption that they were drawn from a log-normal distribution.

If the distribution is not too sensitive to zero values, such as a Normal distribution, then the impact of quantisation might be less problematic. When

zero values cause issues for distribution fitting, like when performing logarithmic transformations on zero, which can lead to fitting errors, different techniques could be used to prevent these errors while having minimal effect on the data's shape. One such method is adding a constant before logarithmic transformations [15, 16, 17].

2.2.4 Techniques for Mitigating Quantisation Effects

Several techniques have been published to address the effects of quantisation, including methods such as adjusting the location parameter or adding jitter to the data [18]. Quantisation error arising from log-normally distributed data was analysed and captured using a statistical model [19]. Sheppard's correction and Quantisation noise models have also been explored, evaluations have shown that Sheppard's correction relies on the assumption that the underlying distribution is smooth, which makes it unsuitable for data with flat segments or sudden jumps, such as piecewise or histogram-like data. The quantisation noise model assumes that the quantisation error is both uniform and independent of the original signal, conditions often violated in real-world scenarios.

These models should only be used when their underlying assumptions are validated against the characteristics of the data [20]. An extension of Sheppard's correction is asymmetric rounding. Rounding is applied to either even or odd numbers [21]. When rounding affects timestamps, resulting in discrete rather than continuous values, models that adjust the hazard function can be effective. The hazard function shows the instant rate at which an event is expected to occur at time t, given that it has not occurred before t. Grimshaw [22] recommends using interval-censored estimation of the Weibull model for discretely measured time-to-event data. Additionally, when the data is truly discrete, a discrete hazard model, such as the logistic model, is more suitable for accurately capturing the behaviour of the data or process.

2.3 Quantisation

Quantisation can be represented mathematically as a mapping from a continuous variable x to a quantised value $Q(x)$:

$$Q(x) = \Delta \cdot \text{round}\left(\frac{x}{\Delta}\right)$$

where Δ represents the quantisation interval and round$(\cdot)$ maps the value to the nearest quantised level. This operation transforms continuous observations into discrete values, thereby altering the empirical structure of the underlying data.

As multiple continuous values may map to the same quantised observation, information about the original distribution may be lost, creating challenges for parameter estimation and statistical inference.

Quantisation is defined as *"The division of a quantity into a discrete number of small parts"* [2]. It is typically regarded as a set of values rounded to whole multiples of a fixed quantity. For example, when (3.1,3.2,3.3,3.4) is rounded to integers, it becomes (3,3,3,3); thus, it has been transformed into a discrete set of values. Quantisation can be applied to various data types, including floating-point numbers, integers, and timestamps, and is used in many fields such as image processing [3], colour palettes [4], speech coding [5] and signals and systems [6].

2.3.1 Benefits of Quantisation

Quantisation offers benefits. In Neural Networks, millions of computational operations occur during training, and using smaller data types can lead to notable improvements in performance and efficiency [7]. Converting Neural Network model weights from 32-bit floating-point to 8-bit signed integers can reduce memory requirements [8]. Quantisation can also enhance storage efficiency when binary floating-point numbers are transformed into integers, decreasing the disk space needed for data logging.

2.3.2 Challenges and Errors

Quantisation also has implications. Rounding the timestamp can affect the chronological order of events. Site Reliability Engineers (SREs) may find it challenging to determine the exact sequence of events during an outage [9].

Performance engineers may struggle to provide precise performance readings. A fraction of a second might seem insignificant to one domain, but for another, it could be crucial, especially when managing large volumes of data.

A timestamp resolution of 1 ms might have been sufficient 20 years ago when a logging standard was set, but it creates problems today because jobs run

faster and can finish in less than a millisecond. Quantisation can interfere with downstream modelling as Data Scientists may struggle to fit models to quantised data. Quantising values alters distribution shapes, thereby changing their moments [10].

Attempting to perform survival analysis (expected duration of time until an event occurs) [11] might be hampered if typical durations are similar to quantisation error. If a value x is quantised to $\hat{x}$, then the quantisation error is defined as: $\epsilon = x - \hat{x}$. The quantisation error is limited by half the quantisation interval when rounding to the nearest quantised value. For example, if rounding to the nearest integer, the quantisation error falls within the interval $[-0.5, +0.5]$.

Quantisation errors occur when rounded values diverge from intended values and can affect the fit of a model to a specific distribution [12]. For example, one might model high-frequency points of a dataset. If raw values at the start of the data are between zero and one second and are rounded to a whole number, then the resulting values will resemble a Bernoulli distribution.

If the data is within the first ten seconds, rounded values might resemble a Poisson distribution. Neither of these distributions is close to the original [13]. Quantisation errors can be problematic for fitting [14]. When events appear to have occurred at the same time due to quantisation, the distribution may look discrete when, in fact, it is continuous.

2.3.3 Impact on Statistical Modelling

Quantisation may round values to numbers outside the range of a given distribution (e.g., to zero), causing fitting challenges. Values rounded to zero can be problematic because the PDF, for example, of a log-normal distribution, is zero at zero, effectively indicating a zero probability that the sample came from a log-normal distribution, as zero is outside the support of the given distribution. These zero values conflict with the assumption that they were drawn from a log-normal distribution.

If the distribution is not too sensitive to zero values, such as a Normal distribution, then the impact of quantisation might be less problematic. When zero values cause issues for distribution fitting, like when performing logarithmic transformations on zero, which can lead to fitting errors, different

techniques could be used to prevent these errors while having minimal effect on the data's shape. One such method is adding a constant before logarithmic transformations [15, 16, 17].

2.3.4 Techniques for Mitigating Quantisation Effects

Several techniques have been published to address the effects of quantisation, including methods such as adjusting the location parameter or adding jitter to the data [18]. Quantisation error arising from log-normally distributed data was analysed and captured using a statistical model [19]. Sheppard's correction and Quantisation noise models have also been explored, evaluations have shown that Sheppard's correction relies on the assumption that the underlying distribution is smooth, which makes it unsuitable for data with flat segments or sudden jumps, such as piecewise or histogram-like data. The quantisation noise model assumes that the quantisation error is both uniform and independent of the original signal, conditions often violated in real-world scenarios.

These models should only be used when their underlying assumptions are validated against the characteristics of the data [20]. An extension of Sheppard's correction is asymmetric rounding. Rounding is applied to either even or odd numbers [21]. When rounding affects timestamps, resulting in discrete rather than continuous values, models that adjust the hazard function can be effective. The hazard function shows the instant rate at which an event is expected to occur at time t, given that it has not occurred before t. Grimshaw [22] recommends using interval-censored estimation of the Weibull model for discretely measured time-to-event data. Additionally, when the data is truly discrete, a discrete hazard model, such as the logistic model, is more suitable for accurately capturing the behaviour of the data or process.

While quantisation offers practical benefits such as reduced storage requirements and improved computational efficiency, it introduces significant statistical challenges. Rounding alters the underlying structure of continuous data, potentially distorting distributional shape, masking variability, and violating assumptions required by parametric modelling and GoF testing. Despite the widespread presence of quantised data in modern systems, limited research has examined how these effects influence distribution fitting, parameter estimation, and statistical inference in high-throughput queuing environments.

2.4 Queuing

Queuing systems are common in many industry sectors, including telephone communications [23], road traffic monitoring [24], hospital waiting lists [25], and banking transactions [26]. They are essential for supply chain operations because they enable reliable, persistent, continuous message processing [27]. These queuing systems assist in managing demand fluctuations by providing mechanisms that support job prioritisation, steady-state operations, and the random arrival of messages [28, 29]. The message order, volume, pace, and dependencies are important for distribution modelling. Understanding these features is important before applying queueing models, such as the GI/GI/1 model, which assumes message independence of the interarrival and service times [1].

However, queues are not immune to performance and reliability issues. Problems include latency, bottlenecks, scalability challenges, noisy neighbour problems, and performance degradation [28]. Resiliency is a key expectation in the Supply Chain domain, and projections of a 25% decline in people's attention span further emphasise the need to minimise wait times [30].

The study was motivated by delays in supply chain networks where EDI messages often throttled or retried, creating bottlenecks. These issues are examined in Chapter 3. In the system analysed, over two million messages are processed on a typical day. Depending on their size, messages may be fragmented into smaller jobs, potentially resulting in over thirty-two billion jobs processed through the enterprise queuing system.

Modelling queues can provide insights into performance constraints, resource allocation, and job starvation. Simulating and analysing these message flows offers a more robust approach to capacity planning and supports the development of innovative solutions to enhance throughput in high-volume environments.

2.4.1 Theory

Agner Erlang first introduced a mathematical framework for queuing theory in 1909 while working for the Copenhagen telephone company [23]. Queuing theory analyses systems in which tasks or jobs arrive, wait in line for service, and are then processed [31, 28]. It enables the prediction and optimisation of

system performance under varying load conditions. Queuing theory is essential for implementing new strategies to optimise queuing systems.

The arrival and service times of jobs can be stochastic (random) or deterministic (fixed). Different strategies are used when jobs are processed through either single-server or multi-server queuing systems. Queuing disciplines such as First-In-First-Out (FIFO), Last-In-First-Out (LIFO), and Shortest Job First (SJF) determine the order in which tasks or jobs are processed [32].

A range of models has been developed to describe queuing systems. Using Kendall's notation, the M/M/1 model assumes exponentially distributed interarrival and service times with a single server. The M/G/1 model extends the M/M/1 model by allowing service times to follow any general distribution rather than strictly exponential [28].

Modelling queuing systems involves analysing arrival rates, service times, and interarrival distributions to evaluate key metrics such as queue length, waiting time, system utilisation, and server idleness [33]. These metrics are essential for understanding performance under different workloads. They serve as a foundation for optimising real-world systems and offer key insights that can improve reliability and scalability in high-throughput environments, such as the supply chain or the Enterprise Messaging System domain.

2.4.2 Applications

With the queuing models understood, different applications exist to support enterprise queuing applications. ActiveMQ [34], Kafka [27], RabbitMQ [35], and IBM Message Queue (IBM MQ) [36] are four typical applications.

ActiveMQ is an open-source Java-based standalone message broker. It supports high availability, load balancing, and asynchronous messaging [37]. RabbitMQ is an open-source distributed message broker. It supports multiple messaging protocols and offers flexible queue routing with various exchange types [38]. Kafka is an open-source distributed event streaming platform [39]. It can store and process streams of data with a guarantee of zero message loss [39]. IBM MQ is part of the WebSphere family of middleware stacks. It supports point-to-point, publish-subscribe and file transfer methods for its messaging and queuing operations, and can transport any type of data as a message [36].

Kafka and RabbitMQ dominate the market share, each serving over thirty thousand companies [40, 41]. In contrast, ActiveMQ has a smaller footprint, with around fourteen thousand customers, roughly half the market share of Kafka and RabbitMQ [40]. High-end social networking companies often use Kafka. Twitter employs Kafka as part of its stream-processing infrastructure [42], and LinkedIn uses it for real-time news feed streaming and offline analytics [43]. In 2021, Netflix utilised it for large-scale data collection [44]. AWS integrates Kafka within its Cloud Service offerings [45].

When considering queuing software, this thesis specifically focuses on IBM MQ due to authorised access to production-level messaging data, where messages are processed through IBM MQ. Studying this system provides a baseline. Production-level metrics that analyse production performance challenges offer a solid foundation for applying and validating novel modelling strategies, thereby enhancing the research's value.

2.4.3 Performance of Message Queuing Systems

Queuing systems rely on performance metrics to determine their suitability for high-throughput and low-latency workloads. Modern Enterprise applications require a scalable messaging infrastructure capable of handling millions of events per second. Several industry platforms, including Kafka and IBM MQ, have been benchmarked under varying workloads. These studies offer insights into the trade-offs between configuration, message characteristics, and infrastructure design.

To meet increased demands for real-time processing, LinkedIn moved from batch-oriented systems to publish–subscribe architectures. The target system was required to handle 10 billion messages per day, with peak loads reaching 172,000 messages per second. ActiveMQ was evaluated, but found to be insufficient for the required throughput. As a result, Kafka was developed internally [46].

Kafka's performance was evaluated by [47], with a focus on tuning system parameters. The evaluation concluded that performance varied significantly across both the underlying infrastructure and message characteristics.

The performance of the IBM MQ JMS Server has also been evaluated, with emphasis on system capacity under varying conditions [48]. The analysis ex-

amined factors such as message size, filter count, and the number of publishers, subscribers, and topics. Message size significantly affected both message and data throughput. The number of topics had minimal impact on overall system capacity; however, server capacity was influenced by the replication grade and the number of filters applied [48].

2.4.4 Practical Enhancements in Modern Queuing Systems

In any queuing system, vendors continually seek opportunities for improvement. IBM, for instance, has introduced a feature called Uniform Clusters, which differs from traditional IBM MQ Clusters. An IBM MQ Cluster provides dynamic message routing and workload balancing, supporting horizontal scaling across matching queues, and automatically establishes connection channels. In contrast, Uniform Clusters build on IBM MQ clustering but are designed for use by a single application or a group of related applications. Each Uniform Cluster supports up to ten queue managers, all of which provide the same messaging services and maintain identical configurations [49]. Despite their advantages, Uniform Clusters present certain limitations. Load balancing issues may occur when the number of queue managers exceeds the number of applications [50]. Removing applications from the cluster may trigger a delayed rebalancing process, resulting in temporary load distribution imbalances.

Another enhancement from IBM is Streaming Queues, which enables the duplication of every message to a secondary queue. Such a feature supports data redundancy for future retrieval and enables SREs or DevOps personnel to analyse messages near real time, with minimal impact on system performance [51].

In Apache Kafka, a broker may remain active yet be unable to establish new connections due to DNS resolution failures, an issue that can be difficult to detect. Version 3.1 of Kafka addresses this problem through KIP-748, which introduces two metrics: “fencedBrokerCount” and “activeBrokerCount”. These metrics allow the controller to monitor and expose the number of active and fenced brokers within the system [52].

Another notable innovation is the concept of “pre-staging” messages at a remote location, as described in a patent that remotely stores a large number of messages in a distributed data storage system configured as a message queue [53].

The remote location may consist of a standalone processor, networked with others. Messages are stored in a memory-based list defined within the remote system, allowing multiple queue managers to access and process them. Many queue managers can then access these messages [53].

2.5 Messaging

Different types of messages are transmitted through queuing systems. In the supply chain field, queues may handle messages such as purchase orders, invoices, payment notifications, inventory updates, and shipping instructions [54]. Outside of the supply chain field, system-level messages like log files, heartbeat signals, and health check metrics can also pass through a queuing system. Supply chain messages typically use structured formats such as XML, EDI, or JSON, whereas non-supply chain messages are often formatted as JSON or plain text.

The production data derived from the supply chain domain mainly consisted of EDI messages in XML format. These messages vary in structure, following common standards such as Accredited Standards Committee X12 (X12), Electronic Data Interchange for Administration, Commerce and Transport (EDIFACT), and Trading Data Communications Standard (TRADACOMS) [55], with most messages based on X12. EDIFACT is an international standard for invoice messages [56]. Many messages include multiple XML files with different schemas and may also contain attachments of various file types and sizes, influencing how messages are processed within the queuing system. These messages are exported to log files for analysis, producing both structured and unstructured text to be parsed.

EDI messages enable the secure electronic exchange of business information, offering features such as message integrity, confidentiality, interoperability and transaction traceability [54]. EDI is widely deployed across industries, as demonstrated in Table 2.1 [57].

Table 2.1: Industries using EDI.

Sector Specific: EDI Standards	**Industry**
ANA	Food, Retail, Distribution
TF2	Health Service
FLEETNET	Fleet Car Industry
EDISHIP	Shipping/Forwarding
PHARMEDI	Pharmaceutical
EDICON	Construction
EDICUG	Components
BEDIS	Book Publishing, Libraries
PIPE	Paper/Printing
EDIA	Banking, Transport
ODETTE	Automotive

EDI, introduced in the UK in 1991, is a vital part of Business-to-Business (B2B) communication [54]. Most of the literature focuses on the advantages of adopting an EDI system, but not on the actual modelling of EDI transactions [58, 59]. Peer influence and industry standards significantly influence organisational decisions to implement EDI [58]. Despite this, operational challenges can hinder wider adoption, for example, within the car industry [60]. Nevertheless, EDI enhances customer service and operational efficiency [59]. In some instances, it can decrease invoice query times by up to fifteen minutes per transaction [61].

In this thesis, the research explores how various EDI message types, such as purchase orders, invoices, and shipment notices, behave under different queuing conditions, and how the distinct characteristics of the messages influence queue throughput performance. The insights from the analysis will support optimisation strategies in Enterprise Messaging Systems. As these EDI messages are written to log files, the timestamp may be rounded to speed up retrieval times and reduce storage size.

Although EDI systems motivate this research, many of the challenges identified extend beyond the supply chain domain. Similar issues arise in IoT telemetry, financial transaction systems, healthcare monitoring, and distributed logging infrastructures, where high-frequency events are commonly quantised for storage and performance reasons. Consequently, the challenges associated with quantisation and distribution fitting represent a broader methodological problem rather than a domain-specific issue.

2.6 Parametric Distributions

When modelling data, distributions can be categorised as continuous or discrete. Continuous distributions are suitable for time-series data taking on any value within a given range. Discrete distributions are suitable for count data, where values are restricted to the integers. Identifying the distribution, under the umbrella of discrete or continuous distributions, that supports the arrival of incoming messages is critical for accurate probabilistic modelling.

Let's now consider some distributions that will be relevant throughout the thesis. A Weibull distribution is a continuous distribution that models the time to failure or the time between events [62]. A continuous exponential distribution models events that occur at random intervals of time. A log-normal continuous distribution models events that occur after a positive period, with multiplicative factors influencing the timing. When the logarithm is applied to the data, the resulting distribution follows a Gaussian distribution.

While many distributions exist, this research focuses on continuous distributions, as message arrival times are naturally modelled as real-valued. Specifically, the Weibull, log-normal, and exponential distributions are emphasised due to their relevance in modelling interarrival and service times.

2.6.1 Weibull

Waloddi Weibull introduced the Weibull distribution in 1951 [62], demonstrating its applicability through five examples, including its use in analysing the strength of Bofors steel, variations in the size of fly ash, and cotton fibre strengths [63]. It is widely employed in various fields, such as reliability analysis [64], psychometric modelling [65], economics [66] and environmental data [67].

As previously mentioned, a two- or three-parameter Weibull distribution models time to failure or time between events [62]. The failure rate can either increase ($\beta > 1$), decrease ($\beta < 1$), or remain constant ($\beta = 1$) over time [64]. The two-parameter distribution includes a shape ($\beta > 0$) and scale ($\lambda > 0$) parameter. [1] Weibull is a generalisation of the exponential distribution when ($\beta = 1$). Its

[1] In some literature, the shape parameter is denoted as β instead of k.

probability PDF is defined as:

Formula: PDF of Weibull Distribution

$$f(x;\beta,\lambda) = \begin{cases} \frac{\beta}{\lambda}\left(\frac{x}{\lambda}\right)^{\beta-1} e^{-(x/\lambda)^{\beta}}, & x > 0, \\ 0, & x \leq 0 \end{cases} \tag{2.1}$$

It also has a cumulative distribution function (CDF), that ranges from 0 to 1. The CDF provides the probability that an event occurs by a specific time t, indicating the likelihood that the observed value is less than or equal to t. For example, "What is the chance that the event has occurred by time t?". The formula for the CDF derived from the PDF of the Weibull distribution is:

Formula: CDF of Weibull Distribution

$$F(x;\lambda,\beta) = 1 - e^{-(x/\lambda)^{\beta}} \tag{2.2}$$

The Weibull distribution's flexibility in modelling different failure rates makes it well-suited for modelling interarrival and service times.

2.6.2 Exponential

The exponential distribution is a continuous probability distribution with a rate parameter $\lambda > 0$ [28]. The distribution developed gradually through the contributions of multiple mathematicians. It is a special case of the Weibull distribution when $\beta = 1$ [64]. A feature of this distribution is that it is memoryless, meaning the time since the last event does not affect the probability of the next event until it occurs [68]. The rate parameter λ is the inverse of the mean: $\lambda = 1/\mu$, and it determines how often events happen. It models the time between events in a Poisson process, where events happen independently and at a steady average rate [28]. It is used in disciplines such as queuing theory [28], reliability engineering [64], and the assessment of cellular manufacturing system reliability [64].

The PDF of the exponential distribution is defined as [28]:

Formula: PDF of Exponential Distribution

$$f(x;\lambda) = \begin{cases} \lambda e^{-\lambda x}, & x \geq 0, \\ 0, & x < 0 \end{cases} \tag{2.3}$$

The formula for the CDF derived from the PDF is [28]:

Formula: CDF of Exponential Distribution

$$F(x;\lambda) = \begin{cases} 1 - e^{-\lambda x}, & x \geq 0, \\ 0, & x < 0 \end{cases} \tag{2.4}$$

The exponential distribution's memoryless property and constant hazard rate make it suitable for modelling the random arrival for probabilistic modelling.

2.6.3 Log-normal

A log-normal distribution is a continuous distribution with a PDF that is right-skewed, starts at zero, and can contain up to four parameters [69]. Galton and McAlister studied it in 1879 [70, 71]. It is widely used as a two-parameter distribution with parameters mean (μ) and standard deviation denoted as (σ). Standard deviation refers to the dispersion of the data. These parameters correspond to the mean and standard deviation of the natural logarithm of the variable, $\ln(X)$, not of the variable X itself. *A log-normal distribution can provide a good representation of a Normal distribution with a small absolute value, less than 0.25 of the coefficient of variation [69].* It is applied in various domains, including modelling network traffic [72], reliability engineering [73] and biology [74].

The PDF of the log-normal distribution is defined as:

Formula: PDF of Log-normal Distribution

$$f(x;\mu,\sigma) = \frac{1}{x\sigma\sqrt{2\pi}} \exp\left(-\frac{(\ln x - \mu)^2}{2\sigma^2}\right), \quad x > 0 \tag{2.5}$$

The formula for the CDF derived from the PDF is:

Formula: CDF of Log-normal Distribution

$$F(x;\mu,\sigma) = \frac{1}{2}\left[1 + \operatorname{erf}\left(\frac{\ln x - \mu}{\sigma\sqrt{2}}\right)\right], \quad x > 0 \tag{2.6}$$

The log-normal distribution is well-suited for applications involving interarrival times, making it relevant for probabilistic modelling addressed in this research.

A summary of the parameters for each distribution, along with its application, is listed in Table 2.2. These distributions are crucial for modelling the interarrival and service times of messages and are vital for the study of this thesis.

Table 2.2: Comparison of Continuous Distributions for Interarrival and Service Time Modelling.

Distribution	**Suitable for interarrival and service times with:**	**Parameters**	**Memoryless**	**Skewness**
Weibull	Variable Rates	λ, β	No (Yes if $\beta = 1$)	Varies with β
Exponential	Constant Rates	λ	Yes	Right-Skewed
Log-normal	Multiplicative Variation	μ, σ	No	Right-Skewed

While these distributions are widely used for modelling interarrival and service times, they generally assume continuous, well-behaved observations. In heavily quantised datasets, these assumptions may be violated, as rounding alters the underlying shape of the data and may introduce discrete artefacts. Consequently, parameter estimation and GoF testing may become unstable or misleading under quantised conditions.

2.7 Distribution Properties and Statistical Moments

Statistical moments are essential tools used across various fields, including reliability engineering [75], parametric modelling [28], and probability theory [76], and are a core element of statistical analysis [77]. They summarise the features of a distribution and aid data-driven process optimisation [78]. Understanding a distribution's moments allows us to visualise changes over

time, helping researchers to better understand a process and its environment, and supporting more informed, data-driven decisions.

Moments of a distribution are calculated differently depending on whether the data is discrete or continuous. The calculation of moments varies depending on data type, with summation for discrete data and integration for continuous data. Since this research concentrates on continuous data, discrete data will not be discussed in the following subsections. Moments of the distributions will be defined using the expected value.

2.7.1 Raw and Central Moments

Statistical moments can be defined either around the origin or the mean of a distribution. Raw moments, also known as non-centered moments, are calculated relative to the origin, whilst central moments are measured with respect to the mean. Central moments are useful because they describe the spread and shape of a distribution in relation to its mean.

- The raw or non-centered moment of order k is: $\mu'_k = \mathbb{E}[X^k]$
- The central moment of order k is: $\mu_k = \mathbb{E}[(X - \mu)^k]$

For example:

- μ_2 is the variance, which measures dispersion.
- μ_3 is related to skewness, indicating asymmetry.
- μ_4 is related to kurtosis, indicating the heaviness of the tails and the sharpness of the peaks.

2.7.2 First Moment

In sequential order, the first moment of a distribution is the expected value [79]. It is defined as:

Formula: Continuous First Moment (Mean)

$$\mathbb{E}[X] = \int_{-\infty}^{\infty} x f(x)\, dx \tag{2.7}$$

2.7.3 Second Moment

The second-moment variance σ^2 is a measure of the spread of the population while s^2 measures the spread of the sample [79]. In some distributions, the standard deviation σ is required, which is the square root of the variance, although it is not a moment in itself. The second moment σ^2 is defined as:

Formula: Continuous Second Moment

$$\mu_2 = \mathbb{E}[(X-\mu)^2] = \int_{-\infty}^{\infty} (x-\mu)^2 f(x)\,dx \tag{2.8}$$

2.7.4 Subsequent Moments

Beyond the first and second moments, subsequent moments describe key characteristics of a distribution's shape. The third and fourth moments will be discussed. The third central moment is defined as:

Formula: Continuous Third Central Moment

$$\mu_3 = \mathbb{E}[(X-\mu)^3] = \int_{-\infty}^{\infty} (x-\mu)^3 f(x)\,dx \tag{2.9}$$

The third-moment, skewness γ_1, measures the asymmetry of a probability distribution around its mean [79]. Asymmetry occurs when the left and right tails of a distribution are not mirror images. Greater skewness signifies a larger deviation of values from the mean, often resulting in longer tails on one side of the distribution. Skewness is the standardised third central moment:

$$\gamma_1 = \frac{\mu_3}{\sigma^3}$$

The fourth central moment kurtosis γ_2 relates to the shape of the peak of a distribution [79]. Kurtosis is defined as:

Formula: Continuous Fourth Moment (Kurtosis)

$$\mu_4 = \mathbb{E}[(X-\mu)^4] = \int_{-\infty}^{\infty} (x-\mu)^4 f(x)\,dx \tag{2.10}$$

Formula: Kurtosis Coefficient

$$\gamma_2 = \frac{\mu_4}{\sigma^4} \tag{2.11}$$

There are three classes of kurtosis: Mesokurtic, Leptokurtic, and Platykurtic.

- Mesokurtic ($\gamma_2 = 3$) similar to a Normal distribution.
- Leptokurtic ($\gamma_2 > 3$) heavy tails, sharp peak.
- Platykurtic ($\gamma_2 < 3$) light tails, flat peak.

In summary, moments describe the central tendency and shape of a distribution. Moments are based on deviations of the random variable from zero. Central moments measure deviations of the random variable from the mean. These concepts are vital to this thesis, as they help guide the development of statistical models and support the fitting of parametric distributions, which are crucial for analysing and interpreting continuous data.

2.8 Non-parametric Distributions

Non-parametric distributions are probability distributions that do not rely on a fixed set of parameters, like the normal, exponential, or Poisson distribution families. These non-parametric distributions are useful when the underlying data structure is unknown or not well-conformed to traditional parametric models. Non-parametric methods make few assumptions about distributional form, offering greater flexibility in modelling complex datasets.

Kernel Density Estimation (KDE), histogram-based estimation, and the empirical cumulative distribution function (ECDF) are techniques used in non-parametric modelling without assuming any parametric form, with KDE being the most common.

2.8.1 Kernel Density Estimation

[2] KDE is a non-parametric (also known as a distribution-free) method for estimating the PDF of a random variable when the sample population does not fit a known probability distribution. It can reveal features of the data, such as multimodality and skewness [80], requiring the selection of both a kernel function and a bandwidth parameter. Placing a kernel function at each data point and summing the overlapping regions produces a smooth

[2] This section is based on the author's previously published work [1].

estimate of the underlying distribution. Kernels assign weights based on the distance between data points, with common types including Epanechnikov [81], Gaussian, Uniform, Box, and Triangle kernels.

The bandwidth parameter determines the spread of each kernel. A large bandwidth may cause oversmoothing and introduce significant bias in the estimated distribution [80]. Several algorithms are available for selecting bandwidth, including Silverman's Rule of Thumb [80], Sheather & Jones [82], and Park & Marron [83].

To assess the accuracy of the KDE models, a GoF measurement is needed to determine whether a non-parametric distribution fits well. Visualisation techniques, such as histograms and the area under the curve (AUC), are used to evaluate the fit. Mean Integrated Squared Error (MISE) is a common metric for quantifying the error between the estimated density and the true underlying distribution [83].

My research concentrates on modelling EDI messages derived from a real-world dataset. Data may be irregular, and traditional parametric assumptions may not hold. Consequently, non-parametric distribution modelling may be necessary to capture the underlying structure of these messages.

Non-parametric GoF tests provide flexibility when the underlying distribution is unknown; however, many of these tests still assume continuous observations and stable empirical distribution behaviour. Under quantised conditions, ties and discretisation effects may distort EDF-CDF comparisons, leading to unstable or misleading test statistics. Existing literature provides limited discussion of these effects in the context of heavily rounded queuing data.

2.9 Goodness of Fit Statistical Tests - Non-Parametric

In this section, the GoF of both rounded and non-rounded data using various non-parametric statistical tests is assessed. These tests are useful when the underlying distribution is unknown or when classical parametric assumptions are not satisfied. The methods discussed include Kolmogorov–Smirnov (KS), Kuiper, CvM, and AD tests. These tests evaluate how well synthetic or

quantised datasets match theoretical continuous distributions using robust GoF methods.

When implementing these GoF tests, critical values vary across distributions and significance levels. These significance levels and critical values determine whether the test statistic rejects or fails to reject the null hypothesis. These critical values depend on whether distribution parameters are known or estimated, categorised into three cases: [84]. In this context, α and β are the parameters of the Weibull distribution where:

1. Case 1: β is known and α is estimated.
2. Case 2: α is known and β is estimated.
3. Case 3: α and β are both unknown and must be estimated.

The critical values and significance levels will be further expanded upon within the GoF tests.

2.9.1 Empirical and Theoretical Distribution Functions

Before jumping straight into statistical tests, it is helpful to understand how different methods represent data distributions, such as histograms, Probability Mass Functions (PMFs), PDFs, and Cumulative Distribution Functions (CDFs). These representations provide foundational insights into how well a dataset can be drawn from a known probability distribution.

Both empirical and theoretical distributions aim to approximate a dataset's CDF, using a PMF for discrete data or a PDF for continuous data. Figure 2.1 shows a visual representation of a dataset's PMF, PDF and CDF. The discrete PMF can be visualised in a histogram with different bin widths and probabilities. The PDF is the density of the sample data. It is the likelihood of a continuous random variable takes on a particular value and is the derivative of the CDF. The CDF provides a probability that a random variable X, takes on a value less than or equal to a specific value x denoted by:

$$F(x) = P(X \leq x)$$

A CDF starts at zero, is right-continuous, non-decreasing and has a range of values from 0 to 1. Each distribution has its own CDF formula. Regardless of the underlying distribution, all CDFs begin at zero for the smallest value of

x and converge to one at the largest value. It is the behaviour between the largest and smallest values that distinguishes the distributions [85].

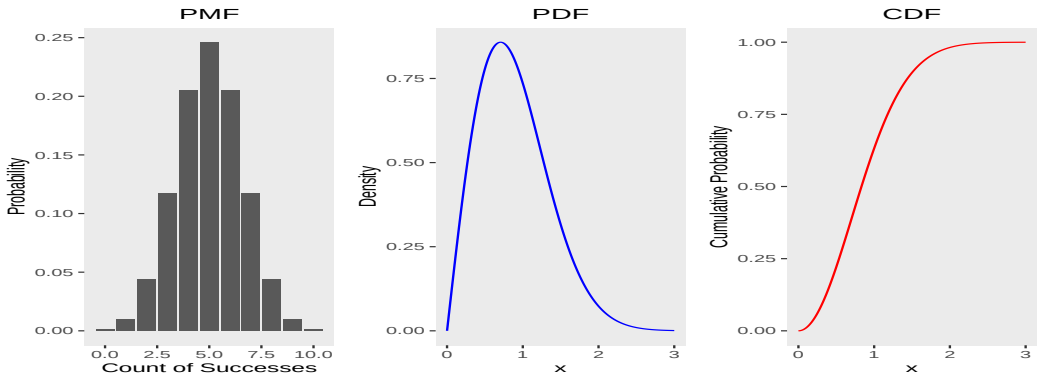


Figure 2.1: (a)PMF, (b)PDF, (c)CDF

2.9.2 Kolmogorov-Smirnov

The Kolmogorov–Smirnov (KS) test, proposed in 1933 is a non-parametric GoF test that measures the maximum absolute difference between an EDF and a CDF [86]. It can also be used to compare the EDFs of two independent samples [87]. For a one-sample KS test, the test statistic D is defined as:

$$\textit{Formula : Kolmogorov–Smirnov} \qquad D = \sup_x |F_n(x) - F(x)| \tag{2.12}$$

Where $F_n(x)$ is the EDF of the sample, $F(x)$ is the CDF of the referenced distribution, and $\sup_x$ is the largest absolute difference between the EDF and the CDF over all values of x [87].

This test is beneficial for assessing whether a sample originates from a specified continuous distribution [85, 88].

2.9.3 Cramer Von Mises

The Cramér–von Mises (CvM) test proposed in 1928 is a non-parametric GoF test that evaluates how well the EDF of a sample dataset could plausibly be drawn from a population with a specified CDF [89]. It determines whether the observed data are likely to have been drawn from a given theoretical distribution.

In the one-sample case, the CvM test evaluates whether a sample comes from a specified theoretical distribution. The equation is defined as:

CvM: One-Sample Equation

$$W^2 = T = n \int_{-\infty}^{\infty} [F_n(x) - F(x)]^2 \, dF(x) \tag{2.13}$$

It is based on the integrated squared difference between the EDF of the sample and the CDF of the theoretical distribution, where n is the sample size, $F_n(x)$ represents the EDF and $F(x)$ denotes the CDF of the theoretical distribution [90]. The term $dF(x)$ signifies the integration performed with respect to the theoretical CDF.

In the two-sample case, CvM assesses whether two samples come from the same (unspecified) continuous distribution [90]. It is defined as:

CvM: Two-Sample Equation

$$Nw^2 = \frac{nm}{n+m} \int_{-\infty}^{\infty} [F_n(x) - G_m(x)]^2 \, dH(x) \tag{2.14}$$

Where Nw^2 calculates the integrated squared difference between the EDF of the first sample $F_n(x)$ and the EDF of the second sample $G_m(x)$. Here, $H(x)$ denotes the pooled empirical distribution function constructed from both samples. Using $dH(x)$, the statistic weights the average of the two EDFs proportionally to the sample sizes using $dH_{n+m}(x)$, where n is the size of the first sample and m is the size of the second sample [90].

Based on the given formulas, CvM applies equal weight to all parts of the distribution and does not favour any specific area, such as the head, the tail or the centre of the distribution. Squaring the differences between the EDF and the CDF quantifies how far the samples deviate from each other in both the center and the tails of the distribution.

For any test to pass a GoF, critical values or cut-off points need to be defined. Table 2.3 shows the significance levels at 90%, 95% and 99% alongside the cut-off points for infinite sample sizes for CvM [90]. If the test statistic W^2 is greater than the critical value, one would reject the null hypothesis: the data

does not come from the sample distribution, or the two samples are not drawn from the same distribution.

Table 2.3: CvM: Critical Values.

Sample Size	Significance Level	CvM: Critical Value
∞	0.10	0.347
∞	0.05	0.461
∞	0.01	0.743

2.9.4 Anderson Darling

The AD GoF test proposed in 1952 is a modification of the KS test. It is a statistical test that places greater emphasis on the tails of the empirical distribution [91]. The hypothesis tests whether a specified PDF represents the distribution of the observed sample data [92]. The formula for the test is outlined below:

Formula : Anderson-Darling

$$A^2 = -n - \frac{1}{n}\sum_{i=1}^{n}\left[(2i-1)\left(\ln F(x_{(i)}) + \ln(1 - F(x_{(n+1-i)}))\right)\right]$$

To understand how AD applies more weight to the tails using the provided formula, $\ln(1 - (F(x_{(n+1-1)})))$ is the tails of the distribution because $1 - F(x)$ is the complementary CDF and is the integral of the PDF from x to ∞. $1 - F(x) = P(X > x)$, which is the area under the curve to the right of any point. The factor $(2i - 1)$ assigns weights that increase the influence based on the data's order. Smaller weights are applied to the head of the data, whilst larger weights are applied to the tail of the data [84].

Table 2.4 shows the critical value cut-off points for the AD test for sample sizes greater than five [84].

Table 2.4: AD: Critical Values.

Sample Size	Significance Level	Critical Values
>5	0.10	1.933
>5	0.05	2.492
>5	0.01	3.857

2.9.5 Kuiper Test

The Kuiper test was introduced by Nicholaas Kuiper in 1960 as an extension of the KS test. It was designed for circular and angular data [93]. The motivation for developing the test was based on being able to determine whether a group of birds had no preference for a flight path direction, where the data represented directions on a circle rather than points on a line [93].

The Kuiper test is suitable for evaluating uniformity in circular distributions, such as time-of-day patterns. It can be applied to test continuous distributions like the uniform, von Mises, or wrapped normal distributions [85]. Unlike linear data, circular data does lacks a fixed starting point, which makes GoF tests less effective. Let $x_1, x_2, \ldots, x_n$ denote a sample of n observed data points, ordered from smallest to largest. For a one-sample test, the Kuiper test statistic V is defined as:

$$\textit{Formula : Kuiper Test} \qquad V = D^+ + D^- \tag{2.15}$$

Where:

$$D^+ = \max_i \left(\frac{i}{n} - F(x_i) \right)$$
$$D^- = \max_i \left(F(x_i) - \frac{i-1}{n} \right)$$

With $F(x_i)$ represents the theoretical CDF, and n denoting the sample size. D^+ and D^- indicate the maximum positive and negative vertical deviations between the EDF and the CDF.

The test selects a random sample of n angular values and evaluates whether the points are randomly dispersed by finding the maximum vertical deviation of the EDF above the CDF (D+) and the maximum vertical deviation of the EDF below the CDF (D-). These two values are then added together to form the GoF test statistic V_n [85]. The KS test only considers the largest single deviation, whereas the Kuiper test combines both extremes.

2.9.6 Summary

As mentioned, which test you choose depends on your hypothesis test. If the hypothesis test focuses more on the tails of a distribution, then AD is a suitable choice. If the shape is more important, then CvM is a better option. Among non-parametric GoF tests for continuous distributions, AD and CvM are the most commonly used. The Kuiper test is used less frequently than the AD test when testing the tails. Although the KS test is widely known, it is generally less powerful than AD.

In some cases, the AD test may return an Inf test statistic in R, depending on different data characteristics. Further examination of this behaviour will be presented in subsequent chapters. Given the focus of this research is on continuous data, both the AD and CvM tests will be used in the subsequent analysis.

Although GoF tests such as AD, CvM, and KS are widely used for validating distributional assumptions, they are primarily designed for continuous observations. Quantisation may introduce ties, discrete artefacts, and support violations that distort EDF-CDF comparisons, potentially leading to unstable test statistics or invalid inference. These limitations are not comprehensively addressed in the existing literature.

2.10 Correlation

There are several types of correlation, including standard correlation, autocorrelation, and cross-correlation. These will be briefly discussed, with more focus placed on statistical correlation tests.

2.10.1 Introduction to Correlation

Correlation can be traced back to Francis Galton in 1888. Correlation is typically measured using correlation coefficients, which quantify the degree and strength of linear association. As Galton described, *two variable organs are said to be co-related when the variation of one is accompanied on the average by more or less variation of the other, and in the same direction. For instance, the length of the arm is said to be correlated with the length of the leg, as individuals with longer arms tend to have longer legs as well* [94].

In empirical analysis, it is useful to assess whether two variables show signs of a correlation, that is, when a change in one variable is associated with changes in another, i.e., increasing or decreasing together. It does not imply causation; it is an association, not a cause-and-effect relationship. In statistical modelling, some models assume independence. When variables are correlated, this assumption is broken, undermining the validity of the models [95].

Autocorrelation, also known as serial correlation, occurs when a variable correlates with a time lag. When modelling time-series data, it is important to assess the presence of autocorrelation since time-series modelling assumes independence. Autocorrelation at lag k measures the linear relationship between values at time t and $t - k$ [95]. Detecting autocorrelation is essential in time series modelling because its presence can invalidate the conclusions of hypothesis tests.

Cross-correlation measures the similarity between two time-series datasets by applying a lag to one of the datasets. It evaluates how one time-series x_t aligns with another y_t across different lag values k [95].

When working with time-series data, selecting the appropriate correlation tests depends on factors like stationarity and linearity. The most frequently used correlation tests are discussed in the following sections.

Modelling EDI transactions over time using queuing theory, which assumes that message arrivals are independent is researached. Testing for a correlation is important to verify this assumption and ensure the validity of the queuing model.

2.10.2 Pearson

Karl Pearson developed the Product-Moment Correlation Coefficient, also known as the Pearson Correlation test, in 1895 [96]. The statistical test measures the strength and direction of a linear relationship between two continuous variables under the assumption of normality. It measures the degree of the correlation, not whether two variables are correlated. The correlation coefficient r ranges from -1 (perfect negative linear correlation) to $+1$ (perfect positive linear correlation), with 0 indicating no linear association [97]. Five key assumptions must be met for Pearson correlation. First, the variables should be continuous. Second, the relationship between them should be linear.

Third, the data should be approximately normally distributed [98]. Fourth, the variance of one variable should be similar across the range of the other (homoscedasticity), and finally here should be no significant outliers. The formula for Pearson is defined as [96]:

Formula:
Product-Moment Correlation Coefficient

$$r = \frac{\sum(x_i - \overline{x})(y_i - \overline{y})}{\sqrt{\sum(x_i - \overline{x})^2 \sum(y_i - \overline{y})^2}} \tag{2.16}$$

Where:

1. x_i is each value of x
2. $\overline{x}$ is the mean of all the x values
3. $x_i - \overline{x}$ is the deviation of each data point from the mean
4. $(x_i - \overline{x})(y_i - \overline{y})$ measure how x and y vary together; this product is used to compute the covariance between the two variables.

The Pearson correlation coefficient is a rescaled version of covariance. Covariance measures the linear association between two variables. Variance is the covariance of a variable with itself [96].

Table 2.5 shows thresholds for the Pearson correlation coefficient r [97]:

Table 2.5: Product-Moment Correlation.

Correlation Coefficient r	**Result**	**Interpretations**
r=0.10	Small	Weak linear relationship between two variables
r=0.30	Medium	Moderate linear relationship
r=0.50	Large	Strong linear relationship

2.10.3 Spearman

In 1904, Charles Spearman introduced the Spearman rank correlation coefficient [99]. It is a non-parametric test assessing the strength and direction of association between two ranked variables. The relationship should be monotonic, meaning that as one variable increases, the other variable either increases, decreases, or remains constant, without changing direction. It proves useful when the data does not follow a Gaussian distribution and linearity is not required. The Spearman rank correlation coefficient ρ is defined as [100]:

Formula : Spearman rank correlation coefficient

$$\rho = 1 - \frac{6 \sum d_i^2}{n(n^3 - 1)} \tag{2.17}$$

where: d_i is the difference between the ranks x_i and y_i and n is the number of observations.

$\rho = 1$ is a perfect positive monotonic relationship (ranks increase together). $\rho = -1$ is a perfect negative monotonic relationship (one rank increases as the other decreases). $\rho = 0$ indicates no monotonic association between the variables.

Table 2.6 shows the typical thresholds for interpreting the strength of Spearman correlation co-efficient at the 0.05 significance level [101].

Table 2.6: Spearman Rank Correlation.

Co-efficient Start Range	**Coefficient End Range**	**Result**
0.05	.29	Weak
0.30	.49	Medium
0.50	1	Strong

2.10.4 Kendall

In 1938, Maurice Kendall developed Kendall's tau correlation coefficient [102]. It is a rank test similar to Spearman's but not identical to Kendall's, and does not measure an association between two random variables. The order of the data determines the rank. If a value is high, it will be assigned a higher rank than a lower value. Smaller values will have lower ranks. Essentially, the data ordering reflects its ranking [103].

The Kendall's tau coefficient is defined as:

Formula: Kendall's tau coefficient

$$\tau = \frac{C - D}{\frac{1}{2}n(n-1)} \tag{2.18}$$

where C is the number of concordant pairs and D is the number of discordant pairs.

2.10.5 Conclusion

Correlation helps analysts understand the relationship between two variables. It can be applied in various domains, including finance, medicine, sales, and marketing. The strength and direction of the association between the two variables guide researchers and analysts in making more informed decisions about the data. The choice of tests to use depends on the data and the characteristics of the tests.

Pearson's correlation is used in regression analysis to measure linear association. For this test, the residuals should follow a Gaussian distribution, and the relationship between the variables should be linear. Spearman's correlation assesses monotonic relationships, making it suitable when data are non-linear or not normally distributed. Kendall's tau is similar to Spearman's rho in that it also measures monotonic association. Still, it is based on the number of concordant and discordant pairs rather than rank differences.

With Pearson correlation using standard deviation and Spearman, using the differences in ranks, problems may arise when a tied ranking occurs, for example, when two athletes tie in the same race. Spearman handles ties by assigning average ranks to tied values. To explain further, if athletes in a race came in at [1, 2, 2, 3, 4] positions. The athletes who came in joint second position have values of two and two and are tied for the second and third place. The average rank is then applied (2+3)/2=2.5; hence, the rank becomes [1, 2.5, 2.5, 4, 5]. Kendall also faced this challenge and overcame it by dividing the variate values of the sum when applying the formula: $\frac{n(n-1)}{2}$ [102].

When comparing Spearman to Kendall, [85] prefers Spearman because it is a simple non-parametric test. Kendall compared his own test to Spearman's and believes they are similar. However, he concurs that Spearman can handle more values than Kendall, giving it a slight advantage [102].

Correlation analysis is particularly important in queuing applications because many classical queueing models assume independence between arrivals and service times. In bursty EDI environments, however, message arrivals may exhibit temporal dependence, cascading behaviour, and clustering effects that violate these assumptions. Quantisation may further obscure or artificially inflate correlation structures by collapsing distinct timestamps into identical

values.

2.11 Parameter Estimation

Modelling a dataset to fit a distribution requires the parameters of that distribution to be defined or estimated if unknown. Distribution fitting makes inferences about sample populations [104]. When estimating the parameters for a population or sample, there is a need to ensure that the parameters support a large proportion of the data while ignoring irrelevant information that may be contained within a small proportion of the population [105]. Using fitted distributions and the laws of probability, one can predict probabilities of specific events. Point and interval estimation are two approaches to estimating parameters. Point estimation provides a single best guess for a parameter, while interval estimation provides a range of values within which the population parameter is likely to fall [68].

Two common methods for point estimation are MME and MLE. These methods support non-linear modelling with data that is not normally distributed and where relationships between variables may not follow a straight line [106]. These will be discussed in the next section.

2.11.1 Point Estimation - MME

MME, introduced by Pearson in 1894, estimates the parameters θ of a probability distribution by comparing and matching the theoretical and empirical moments of a distribution [76, 107]. These moments are defined in section 2.7. To match the empirical to the theoretical moments using, for example, a Normal distribution, the first-order moment, mean, and the second-order moment, variance, are calculated from the observed data. MME then selects the θ values to produce a distribution with those moments. In some situations, the parameters of a distribution are not named the same as the moments; for example, a Weibull distribution has two parameters (shape β and scale λ). Using a Gamma function denoted as $\Gamma()$, one would use the first-order moment μ and the second-order moment σ^2 to estimate the parameters for the Weibull distribution. When using MME, the precision of the second moment can affect the fit to a distribution, as the variance may be closely clustered together.

2.11.2 Point Estimation - MLE

MLE, introduced by Fisher, estimates the parameters of a probability distribution by maximising the likelihood of the observed data [105]. The likelihood function is derived from the PDF of the underlying distribution and measures how well specific parameter values fit the data. MLE finds the parameter values that make the observed data most probable [106]. It does this by evaluating a likelihood function across a family of probability distributions, where each model is characterised by specific parameter values, allowing direct comparison between models based on how effectively they explain the same data [106]. A walk-through of the calculation steps is provided using Weibull as an example.

2.11.3 MLE Calculation Steps

Assuming that the observations $x_1, x_2, \ldots, x_n$ are independent and identically distributed (i.i.d.). The formula for Weibull is defined below:

Formula: Likelihood of Weibull Distribution

$$L(\lambda, \beta) = \prod_{i=1}^{n} f(x_i; \lambda, \beta) = \prod_{i=1}^{n} \frac{\beta}{\lambda} \left(\frac{x_i}{\lambda}\right)^{\beta-1} e^{-(x_i/\lambda)^{\beta}} \tag{2.19}$$

- $L(\lambda, \beta)$ is the likelihood function, expressed as a function of the shape parameter β and the scale parameter λ.
- $f(x_i; \lambda, \beta)$ is the PDF of the Weibull distribution evaluated at each observation x_i.
- The product symbol $\prod$ indicates that the total likelihood is the product of individual PDF values.

With the likelihood function defined for the Weibull distribution, MLE selects the parameter values θ that maximise the likelihood of the observed data.

Maximising Likelihood

Each parameter set θ, defines a specific model within the family of distributions, and MLE identifies the one that best fits the current data while also being able to predict future data [106]. The process effectively chooses the values of

θ that make the observed data most probable [108]. When θ is estimated, the PDF of the entire data are also determined [107].

To simplify the computation of the likelihood, the MLE takes the natural logarithm of the likelihood function, which transforms the product of terms into a sum, making it easier to compute. For the Weibull distribution, the log-likelihood function is given as:

Formula: Log of Likelihood with Respect to Weibull

$$\ell(\lambda, \beta) = \log L(\lambda, \beta) = \sum_{i=1}^{n} \left[\log\left(\frac{\beta}{\lambda}\right) + (\beta - 1) \log\left(\frac{x_i}{\lambda}\right) - \left(\frac{x_i}{\lambda}\right)^{\beta} \right] \tag{2.20}$$

There are multiple methods for finding the optimal $\theta = (\lambda, \beta)$. In calculus, we can take the logarithm of the likelihood or, using differentiation, compute the first and second derivatives of the log-likelihood with respect to each parameter. Using brute force, one can apply a grid search approach. Numerical minimisation techniques for optimisation could also be employed, such as Newton-Raphson (NR), Gradient Descent, Nelder-Mead, EM, and Markov Chain Monte Carlo (MCMC) methods.

Several R packages support distribution fitting, notably MASS and fitdistrplus [109, 110]. Fitdistrplus extends MASS functionality, offering greater flexibility in fitting continuous distributions for MLE and MME. MASS focuses primarily on MLE, with limited ability to model continuous distributions using MME [111]. For this study, the fitdist function from the fitdistrplus package is used due to its support for both MLE and MME estimation methods, enabling direct comparison under identical conditions.

2.11.4 Interval Estimation - Bayesian Inference

Bayesian inference is a statistical technique used to calculate probabilities based on prior evidence, updating those probabilities as new data emerges. The method originated with Thomas Bayes, who developed what is now known as Bayes' Theorem, and was later formalised by Pierre-Simon Laplace in 1812 [112]. Laplace established the basis for modern probability theory and Bayesian inference by viewing probability as a measure of belief or uncertainty [112]. Bayesian inference, widely used in statistics, underpins many computational

techniques, such as Markov Chain Monte Carlo (MCMC) methods, including the Metropolis-Hastings algorithm [113] and Gibbs sampling [114]. Bayesian approaches are essential across a variety of fields, including artificial intelligence [115], finance [116], bioinformatics [117], climate science [118], and natural language processing [119]. For interval estimation, Bayesian estimation is advantageous when prior information is known.

For point estimation, MLE focuses on maximising the likelihood of the observed data and estimates the parameters of a probability distribution. MME is an approximation of the moments that match a distribution, and both MLE and MME are sensitive to outliers [120, 107]. MLE can be computationally expensive for large datasets or complex models with many parameters [106]. Its performance depends on distributions being correctly specified, for example, modelling count data with a Poisson distribution when a Negative-Binomial is more appropriate [121]. Such mis-specifications can lead to poor estimates.

2.12 Zero-Adjusted Models

When modelling service times, especially in supply chain organisations that process EDI messages, it is common to see a high number of zero values. These zero values may arise from the instantaneous processing of events. To address the high volume of zero values, zero-adjusted models can be used to account for the inflation of zeros in the distribution of the data.

With a high concentration of zero-valued service times, this can affect distribution modelling as the majority of service times are concentrated near the head, with relatively few messages in the tail. Models have been developed to support these excess zeros, such as the Hurdle model proposed by Mullahy in 1986 [122], the zero-inflated Poisson (ZIP) regression model proposed by Lambert in 1992 [123], and the zero-inflated negative binomial (ZINB) regression model proposed by Hall in 2000 [124]. Other models do exist, but for this research, the Hurdle and zero-inflated models will be briefly discussed.

Both the Hurdle and zero-inflated models handle excess zeros through a two-part structure; however, they differ in how they treat the zero observations. The Hurdle model is designed to support excess zeros in count data. It consists of two components: a binary model that determines whether a count is zero

or positive, and a truncated count model that estimates the distribution of positive counts.

Only zeros values are generated by the binary model, and the count component models only positive values. The binary component fitted using binomial regression captures the likelihood of crossing the "hurdle" from zero to a positive count. Once that hurdle is crossed, the positive counts are modelled using a zero-truncated Poisson or negative binomial regression models [125]. Zero-inflated models are similar in that they have a binary model and a count component model. However, the count component model is not truncated and includes zeros as a valid outcome [125].

To better model count data with more zeros than expected, zero-inflated models introduce a framework that separates structural zeros from those generated by the count process. The Zip model outperforms a normal Poisson and negative binomial model when excess zeros are due to a mixture of "perfect" and "imperfect" states [123]. A "perfect" state refers to a process that always produces a zero outcome, not by chance but by design. For example, consider incoming messages that are automatically rejected and never processed. These consistently result in a count of zero. In contrast, an "imperfect" state describes a process that can produce either zero or non-zero outcomes, such as one following a Poisson or negative binomial distribution. An example might be messages that are processed but occasionally take negligible time, resulting in a zero-duration count. Although the ZIP model captures excess zeros, additional overdispersion remains, and models such as ZINB offer a better fit [126].

2.13 Complimentary Supporting Tools

Often, data is not smooth or easy to model, especially when working with real-world datasets. Additional support tools may be necessary depending on the results of the data analysis. For example, outliers are a common issue in datasets.

2.13.1 Outlier Detection

Identifying outliers, often caused by rare events, is essential for statistical modelling and general data analysis. Outliers are observations that stray from a dataset's distribution, affecting parameter estimation and skewing

statistical test outcomes. Common statistical methods for detecting outliers are the z-score, the Mahalanobis distance, Cook's distance and the Minimum Covariance Determinant. The Mahalanobis distance applies to multivariate data and measures how far a point is from a distribution, accounting for both variance and correlation in the data [127]. The z-score, used for univariate data, quantifies how many standard deviations a data point is from the mean. It is also called a standard score [128].

Cook's distance combines studentised residuals and leverage to assess the influence of each observation on the fitted model [129]. The Minimum Covariance Determinant (MCD) identifies a subset of observations whose covariance matrix has the smallest determinant, considering it as the most central part of the data, and uses its mean and covariance to compute robust Mahalanobis distances for outlier detection [130]. While Mahalanobis distance is often used for general multivariate outlier detection, Cook's distance is helpful in regression analysis for identifying influential points [131]. Although MCD is computationally intensive, it enhances outlier detection accuracy compared to Mahalanobis distance and reduces bias in statistical estimation [131].

Other metrics could be used to identify extreme values, such as standard deviation, Hills Estimator, and Dixon's Q test. Each has strengths and limitations:

- Standard deviation calculates the deviation from the data's mean. Values exceeding 3 standard deviations are typically classified as extreme values [79]. However, this is primarily used for Gaussian distributions and is unsuitable for non-Gaussian, heavy-tailed datasets.
- Hill's Estimator uses upper-order statistics to analyse data with heavy tails. It estimates the tail index of a distribution. A lower tail index implies a greater likelihood of observing extreme values [132].
- Dixon's Q test can detect extreme values. However, it only supports small datasets [133, 134]. Due to the limitations in the sample size, it is not widely used for outlier detection.
- Skewness, is a statistical measure that describes the asymmetry of the distribution of values around the mean. It can provide insights into

distribution's tails. A skewness value greater than 1 indicates significant right-tailed skewness and the presence of extreme values. However, skewness does not directly tell which data points are outliers and is not a suitable approach [85].

2.14 Industry 4.0

Intelligent advances in information technology have greatly influenced the evolution of supply chain networks. Industry 4.0, often associated with smart factories and advanced manufacturing, refers to the integration of digital technologies into industrial processes [135].

Within the context of B2B transactions, Industry 4.0 supports the automatic exchange of information between business entities. It offers complete traceability and transparency across suppliers, manufacturers, and trading partners. These capabilities support the development of autonomous, streamlined services throughout the supply chain industry.

The supply chain sector plays an important role in ensuring the transportation and traceability of goods. Adopting Industry 4.0 technologies has been linked to improved performance, including a 53% increase in order fulfilment opportunities and a 71% improvement in procurement processes [136].

Despite these advantages, the lack of standardisation in IT security remains a barrier to widespread adoption. Recent research has presented a taxonomy of the advantages and disadvantages of Industry 4.0 in supply chain networks [137], and various frameworks have been proposed to evaluate organisational readiness for adopting such technologies [138].

2.14.1 AmI, IoT and Supply Chains

Ambient Intelligence (AmI) refers to interconnected systems capable of sensing, reasoning, and responding to user and environmental data [139]. AmI systems are commonly deployed in smart spaces, where they apply context-aware reasoning to collected data in order to adapt to users and operational needs [140]. Within supply chain queuing systems, this reasoning capability can support intelligent decision-making, particularly under stress-test scenarios involving EDI messages.

A smart space connects devices in a distributed system, allowing resource sharing and real-time information exchange [141]. Typically deployed in the Internet of Things (IoT) environments, smart spaces rely on sensor data to continuously monitor environmental characteristics. Once sufficient data is gathered, automated actions can be triggered, such as dynamically scaling infrastructure containers based on the over- or under-provisioning of resources [142], enhancing the efficiency and responsiveness of supply chain operations [140].

A range of AmI services has been developed within smart space environments. One example is Smart-M3, an open-source information-sharing platform initiated by Nokia in 2006 and later extended by various research institutions [140]. Smart-M3 offers a shared view of dynamic knowledge across distributed applications using two core components: the Semantic Information Broker (SIB) and the Knowledge Processor (KP) [143]. To meet the latency demands of such environments, fog computing has emerged as a supporting paradigm by integrating resources at both edges and cloud levels. Fog computing enables rapid processing of sensor data and execution of actuated responses using the sense–process–actuate model [144], which is particularly beneficial in B2B contexts, where real-time transaction visibility is critical.

A review of existing research into EDI transaction events indicates that supply chain organisations heavily depend on real-time monitoring and logging systems to trace, manage, and resolve issues in EDI processing. The timing and flow of transactions through queuing and integration platforms offer a meaningful representation of service disruptions and recovery durations. With that, Chapter 3 focuses on modelling the structural and temporal dynamics of the EDI message. Chapter 4 introduces the challenge of heterogeneity across message types while building a foundational framework for modelling these messages. Chapters 5, 6, and 7 then address convergence and GoF issues in distributional modelling due to the implications of time rounding in logged EDI data and the resulting challenges it poses for accurate parameter estimation. Building on this, Chapter 8 explores methods for unrounding or recovering the underlying event times, with Chapter 9 proposing techniques to fit models without requiring explicit correction of rounding errors. Together, these chapters form a coherent progression from foundational modelling to practical solutions for handling imperfect real-world EDI data.

2.15 Research Gap

Existing literature largely assumes continuous and well-behaved observations when applying distribution fitting, parameter estimation, and GoF testing. Limited research has investigated the effects of quantisation on statistical modelling, particularly regarding:

- The impact of rounding on GoF tests.
- Parameter estimation bias under quantised conditions.
- Instability in distribution fitting.
- Reconstruction of continuous distributions from rounded observations.
- The suitability of modelling techniques under heavily quantised data.

This thesis addresses these gaps by investigating how quantisation affects statistical modelling and by proposing methods to mitigate or account for quantisation-induced distortions.

CHAPTER 3

Where We Came From

The electronic exchange of B2B (e.g. purchase orders, inventory data and shipment notices between departments or organisations) information can eliminate the need for human intervention and paper-copy trails. Incorporating EDI standards into an organisation can drastically improve the efficiency of processing times. Modelling the behaviour of EDI messages within a supply chain network's queuing system has many purposes, from understanding the efficiency of queue behaviour to process re-engineering. The present chapter demonstrates that these messages are heterogeneous, correlated, not stationary, challenging to model, and investigates whether a parametric or non-parametric approach is appropriate to model message service and interarrival times. The results show that parametric distribution models are suitable for modelling the distribution's tail, whilst non-parametric KDE models are better suited for modelling the head.

3.1 Introduction

Queuing systems help businesses within the supply chain domain enhance operational efficiency and throughput by supporting high-volume processing of transactional messages [27]. Where demand temporarily exceeds supply, queuing systems help maintain system stability by preventing job loss and enabling message prioritisation. Maintaining a steady-state operation is integral in avoiding supply chain distribution problems [29].

Problems arise within a supply chain network, where message processing times are not clearly understood. Such issues can be detrimental to simulation-based testing strategies, particularly when the modelled messages do not accurately represent the true characteristics of real-world data, or when test scenarios are limited to a narrow subset of message types, failing to capture the full variability of the system. In practice, these heterogeneous EDI messages often suffer from numerous failed message retries and throttling, resulting in bottlenecks within the Enterprise Messaging system.

EDI messages are split into many fragments to support parallel processing and more efficient message throughput through the queuing systems. However, there are disadvantages to message fragmentation when modelling. Fragmentation introduces a correlation between jobs, violating assumptions of independence for queue modelling. Fragmented messages can lead to overdispersion, complicating both parametric and non-parametric modelling techniques. Other reasons for fragmenting messages include file size limitations imposed by transport protocols or by the receiving trading partners' limitations. EDI messages utilise hierarchical structures based on message type, and large batches can be automatically split at various levels to facilitate easier processing. EDI translation tools and middleware systems have configuration options to automatically split messages based on routing and mapping rules.

Here, the chapter investigates the challenges encountered when modelling EDI transactions, including correlation, message bundling, and heavy-tailed data. Various techniques are used to classify, filter, split, and group data to facilitate both parametric and non-parametric modelling. The chapter aims to answer the following questions:

1. Can EDI messages be effectively modelled using parametric or non-parametric techniques?
2. Can service times (ST) and interarrival times (IAT) be accurately modelled using these approaches?
3. To what extent can these messages be classified to reflect the complexity and variability of their underlying structure?

Section 3.2 gives an overview of the data and describes the methods intended to apply to service times, interarrival times, and other aspects of the data. Section 3.4 gives results of the methods with some limited commentary, but the main discussion is saved until Section 3.5. The results indicate that assumptions of i.i.d. behaviour do not hold in raw message data, and that segmentation is essential for uncovering meaningful temporal dynamics in service and interarrival times. The chapter concludes with a summary in Section 3.6.

3.2 Data Overview

The study presented uses an enterprise dataset from a cloud-based supply chain network. Four log file datasets were compiled to form the core source of the research data. The dataset collected for analysis spanned a 13.5-hour window, from midnight to 13:30 on 30th November 2020. While the data was being checked for completeness, messages entered the system every second. A comparison between the expected and actual number of seconds revealed 13,796 gaps, which may indicate either missing data or intervals with no incoming messages. The production dataset reveals that, on an average day, the system processes over two million messages.

For accurate modelling, it is essential to consider both a high-level overview of the system architecture and a finer, trace-level perspective of the messages, which provides a more comprehensive understanding of the message flow, processing logic, and the structural intricacies of the underlying infrastructure.

Figure 3.1 illustrates the flow of a message through the cloud-based supply chain network. The research focuses on the components shaded in grey, which include the Translation Service, a B2B Rules Engine, a Message Dispatcher and a Session Announcement Protocol (SAP) interface server, the primary input source for data originating from the World Wide Web. These components were collectively deployed across seven dedicated servers.

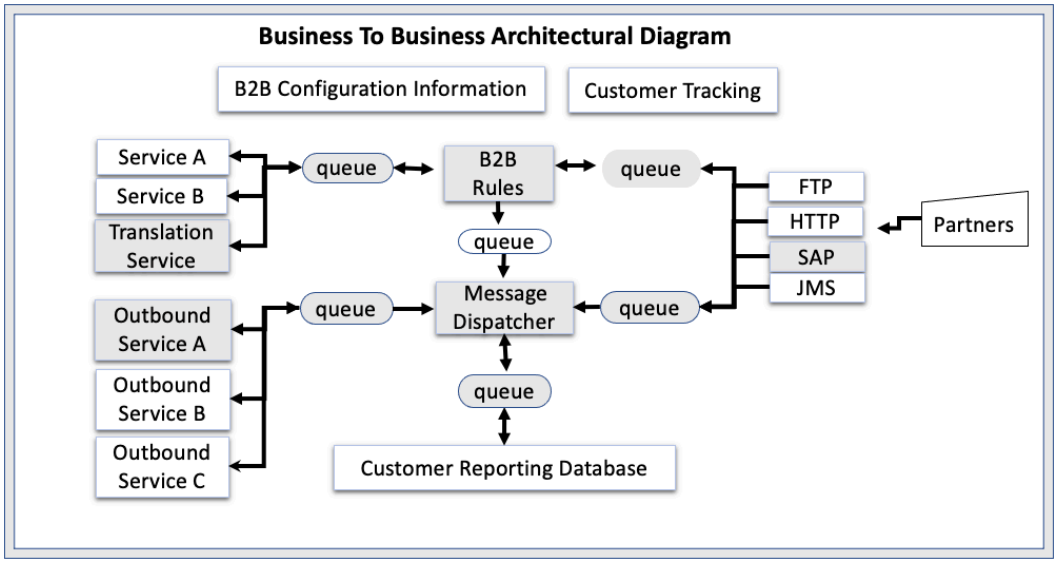


Figure 3.1: Supply Chain Network Architectural Message Flow.

Data from the entities was collected using Graylog, which captured both inbound SAP messages and Outbound Service activity. Logs from the Message Dispatcher, B2B Rules Engine, and Translation Service were aggregated through a coordinated logging process, resulting in a mix of structured, unstructured, and XML-based formats reflecting the system's complexity.

Initial efforts focused on tracing the complete end-to-end flow of messages from their origin to their final destination, specifically, from the SAP interface to the Outbound Service. Figure 3.2 shows an example of the processing complexities of a single message flow end to end.

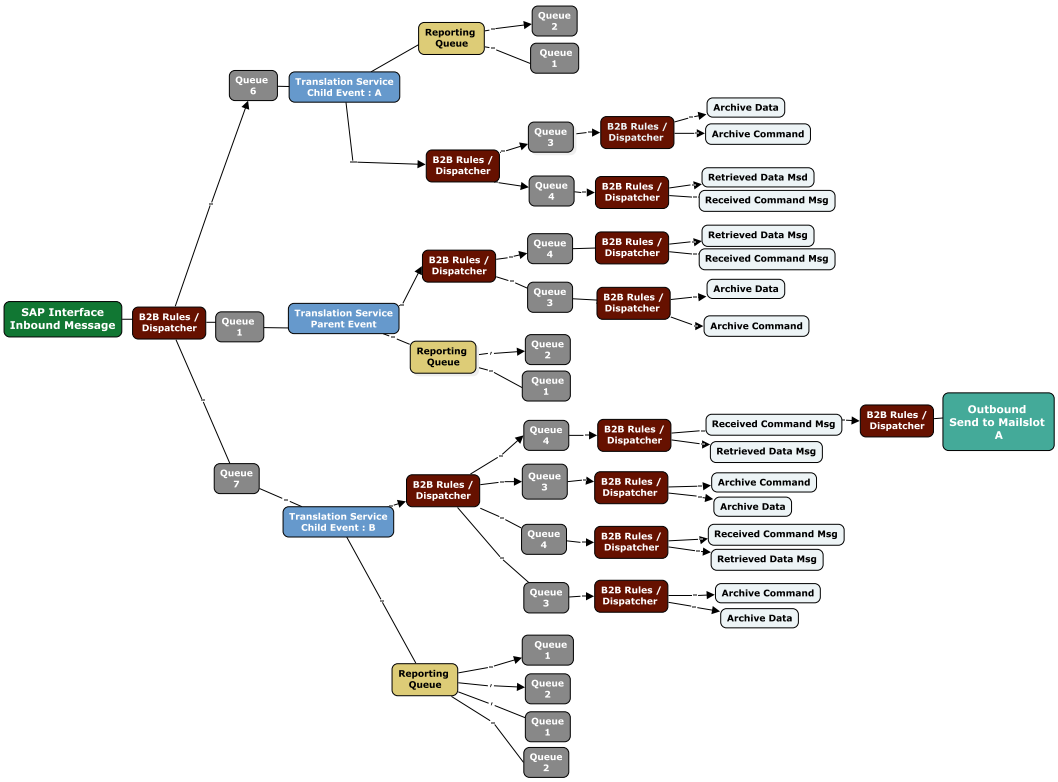


Figure 3.2: B2B Low Level Message Flow.

A traversal trace of the message flow is outlined. The message originates at the SAP source server, depicted on the left-hand side as the "Inbound Message" in green, and subsequently passes through the B2B Rules Engine and the Message Dispatcher, represented by the brown entities. Subsequently, the message is dispatched to three distinct queues noted in gray for downstream processing. The queuing system is IBM MQ. After passing through the first set of IBM MQ queues, the message then gets sent to the Translation Service represented by the blue entities, where the message is split with parent-child relationships. In this case, the message was split into three fragments. The Translation Service processes the message through the Translation queues and then returns it to the B2B Rules Engine / Message Dispatcher, where the Rules Engine initiates *Command Requests* and *Data Requests* shown in the figure as shaded white entities. After the Command Request and Data Request are complete, the Rules Engine and Message Dispatcher send the message to the client's trading partner, which is the last entity to the right of the image in light green. As IBM MQ did not log when a message left the queue, the Rules Engine and

Message Dispatcher *Command Request and Data Request* timestamps were used to determine when the message left the queue.

Figure 3.2 shows that one message entered nineteen queues as part of its lifecycle, which unobfuscates the flow of an EDI message in a way that does not seem to be presented in previous literature.

Based on the available data, only 527 incoming messages could be fully traced as having been initiated from the SAP interface within the log files. Due to the limited usage of this interface at the time of data collection, the analysis shifted to using the Translation Service as the entry point, since approximately 90% of all incoming messages are processed through this component, amounting to roughly two million messages per day.

The Translation Service is responsible for converting messages from one format to another (e.g., X12 to CSV) and for concatenating or extracting documents embedded within a message. Additionally, the Translation Service supports the splitting of a single input file into multiple output files. Such functionality reduces the size of individual messages processed through the queues, thereby decreasing job sizes and enabling more efficient parallel processing.

Figure 3.3 shows the chronological order of steps a message takes within the Translation Service.

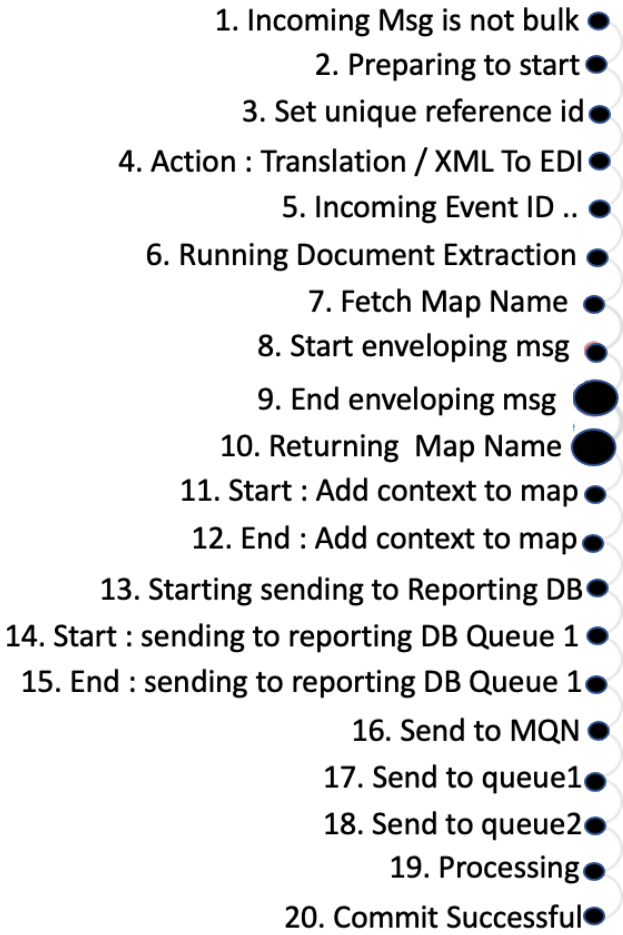


Figure 3.3: Message Translation Steps.

A simple message was chosen to show the processing order, as a more complex message would result in a larger number of steps, rendering the graph across many pages. Step 4 is where the actual Translation Service starts. Document extraction begins in Step 6. In Steps 16 to 18, messages are sent to the Translation queue, as noted in Figure 3.2 with the grey entities with the titles "Queue 3" and "Queue 4". Previous literature does not showcase the order of the steps within the Translation Service. Different message types do not always progress in a strictly chronological order through the numbered translation steps. In some instances, a message may skip a step, while in others, it may loop back and repeat a step.

From the Translation Service, data is observed going into two different queues. The CMD queue is the command queue, where commands are sent. The Data

queue is where the corresponding data is sent. Every message is associated with a CMD and Data Queue entry. If we refer back to Figure 3.2, this single message hits these queues eight times in total, represented by the rounded grey rectangles titled "Queue 3" and "Queue 4". Since this research focuses on modelling service and interarrival times, the role of these queues is crucial, as they represent the primary points of message processing and delays within the system. Table 3.1 shows the volume of messages analysed.

Table 3.1: Translation Service: All Data Message Volume.

Date	CMD Queue	Data Queue
30th Nov	1,036,938	1,036,956

3.2.1 Data Limitations

A number of practical limitations of the dataset were identified. First, it was challenging to trace a message end-to-end. There was no unique identifier between the different log files and Graylog. Whilst every effort was made to identify unique characteristics that trace individual messages through the system, the developed method is somewhat ad-hoc.

Due to the system's logging configuration, only the submission of messages to the queue could be directly observed. The point at which the message exited the queue had to be inferred using log data from downstream applications.

Collecting long-duration data from the system is challenging due to the high message volume processed by the Rules Engine and Message Dispatcher. As a result, log files are retained for only a few hours before being recycled.

3.3 Methods

3.3.1 Message Classification

Partitioning by message type was domain-driven, reflecting operational differences between distinct EDI transaction schemas.

Log-scale plots of service times reveal distinct, overlapping distributions, especially when the messages are classified as Split="1". These may reflect different message classes motivating the research on message classification. Figure 3.4 focuses on service times for messages with Split=1 and zero values

removed. A value of Split = 1 indicates that the message was not divided into multiple submessages. It shows clear groupings between log-values –7 and 0. It appears that the messages from -7 to -4 form one distinct group, suggesting quantisation. The bin widths in the plot provide sufficient resolution without distortion, and identify these structural groupings.

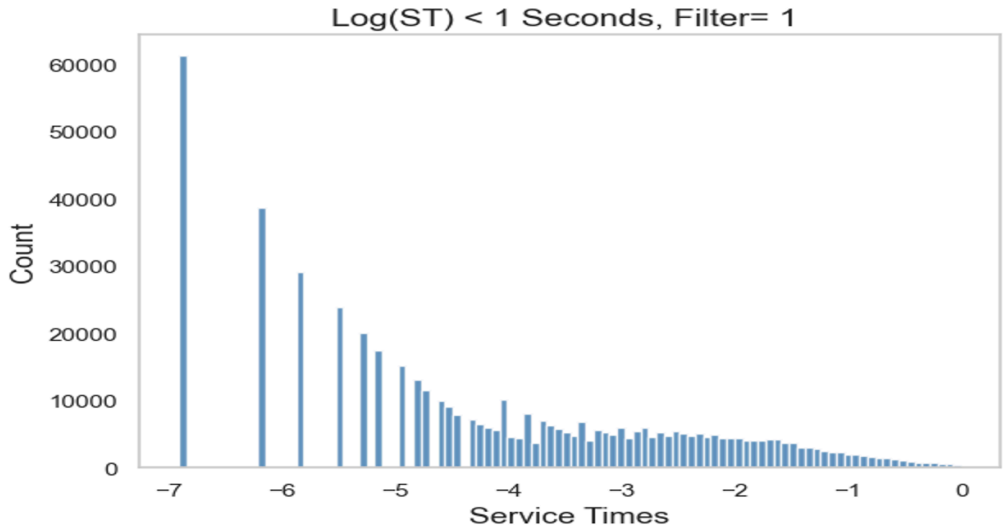


Figure 3.4: Service Times, No Zero's, Log Transform, Split=1, ST< 1 Second.

To further investigate message classification, the range from –4 to –1.5 is examined in greater detail, revealing up to five potentially distinct message types, as illustrated in Figure 3.5.

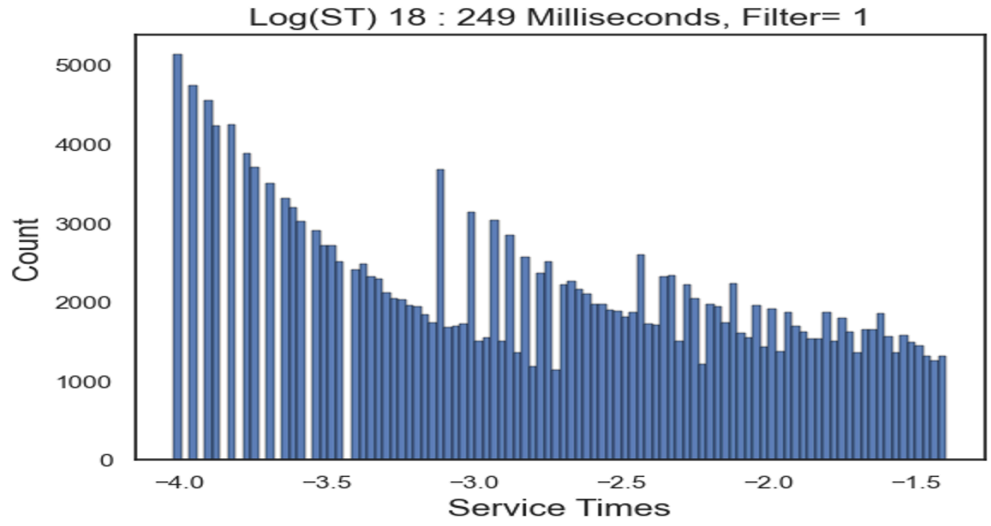


Figure 3.5: Service Times, No Zero's, Log Transform, Split=1.

Together, these visualisations support the case for structural classification of messages. A practical classification scheme might incorporate both service time characteristics and metadata, such as message status or file size. Potential status categories include “Waiting”, “Rejected”, “None”, “Accepted with Errors”, “Accepted”, “Partially Accepted”, and “Received”. Such classification would facilitate subsequent modelling steps by enabling stratification into more homogeneous subgroups, thereby improving both accuracy and interpretability of the results.

3.3.2 EDI Modelling

When fitting data to parametric models, the easiest case is when time-series data shows no dependence on previous values. The independence of arrivals is also a common assumption in queuing models. If a correlation exists, the time-series data may be deemed non-stationary, and different techniques may be used to handle this correlation. The data is checked for a correlation while checking the distribution fit of the data. Referring back to Table 2.6, which shows the Spearman’s rank correlation co-efficient test statistic for non-normal data, this test will be used due to the non-Gaussian nature of the dataset.

In the following subsections, the analytical approach is described.

3.3.2.1 Normal And Busy Periods

Partitioning by operational periods was statistically motivated to reduce the effects of temporal non-stationarity and varying workload intensity across the observation period.

According to the DevOps support team, a specific time range was identified during which the system was believed to have experienced performance issues. When analysing the data, a period was found where the number of messages queued was always bigger than zero and often growing. Accordingly, the data was broken into two periods, busy and normal. Table 3.2 shows the busy period was just under forty minutes, whilst the normal period was just over twelve hours. Such a pattern suggests a similar volume of messages per second is going into each queue (≈ 22 messages per second), suggesting the busy period is caused by a change in STs rather than IATs. Arrivals to the CMD and Data queue are similar in both periods, though there is a slight discrepancy of

eighteen messages between the two queues in the busy period. The difference may be due to some messages containing large attachments. If the message is big, it is paginated, causing more data messages.

Table 3.2: Translation Data: Different Time Periods.

Period	Start Time	End Time	CMD Queue	Data Queue
Normal	12AM	12PM	984,183	984,183
Busy	12:50PM	1:29PM	52,755	52,773

3.3.2.2 Message Split Count

Partitioning by split count was motivated by the hypothesis that fragmented messages exhibit different timing characteristics from single-message transactions.

A significant number of messages were part of *what is called a bundle*, i.e., part of a group with zero seconds between them. Given the bursty nature of EDI message traffic and the diverse characteristics within these messages, splitting messages by their bundle size may help isolate underlying patterns and reduce heteroscedasticity, potentially improving the fit of parametric models. Therefore, messages are classified by the number of messages in each bundle to better capture these dynamics. The messages are categorised into three groups as follows:

First, Split="1" occurs when a single message is sent into the system, and one message gets sent to the queue. The count refers specifically to how many times the message is split and sent to a particular queue, such as the CMD queue. These messages are not part of a bundle and may be small in size. Split="2" occurs when one message enters the system and two messages are sent to the queue. Finally, When Split="Other", a single incoming message results in three or more messages being sent to the queue. The highest number of messages produced from one input, effectively the largest bundle, was 1,617. That means that one message produced 1,617 message splits. Referring back to Section 3.2, thirty-two billion jobs (2 million * 1617) can come from two million messages.

Table 3.3 shows how many messages belong to each split group over the normal period. As seen from the table, most messages are in the Split="1" group.

Table 3.3: Translation Data: Splits Count.

Splits	Normal Period	Percentage
1	558,549	57%
2	233,421	24%
Other	192,213	19%

3.3.3 Correlation

3.3.3.1 Hurdle Modelling

A considerable volume of messages was processed in zero seconds. To improve the distributional fit and account for overdispersion, a Hurdle-type model was applied, with zero values modelled separately. Table 3.4 shows the percentage of messages removed using this approach.

Table 3.4: Normal Period: Hurdle-Type Model.

Splits	Normal Period	Data Removed
1	511,670	9%
2	112,295	48%
Other	119,39	6%

3.3.3.2 Message Bundle

As noted, there are a number of messages that result in a bundle. When calculating the arrival times and service times of these messages, it may make sense to treat them as a single message. In this case, the first message is used in the bundle to calculate the interarrival times and service times, as the other messages appear to have zero duration.

3.3.3.3 Scheduled Versus Un-Scheduled Messages

During the normal period time frame, the data was checked for other signs of burstiness. Figure 3.6 shows the frequency of jobs arriving based on the minute within the hour. Minute 00 shows the highest message frequency, with minutes 30–33 and 43–46 also showing elevated activity. These peaks may correspond to scheduled jobs configured on the servers.

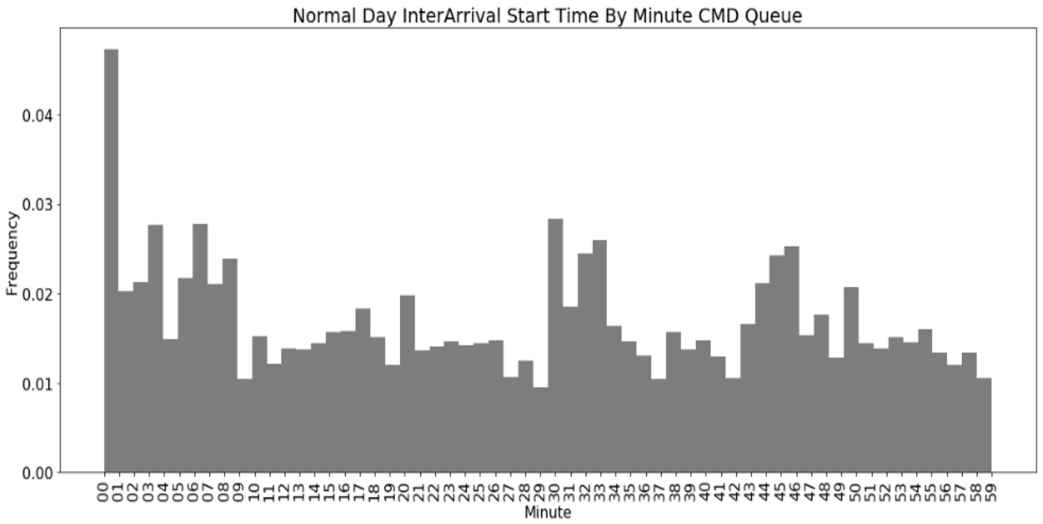


Figure 3.6: Normal Period: Burstiness in Data.

As these scheduled jobs are likely to require separate modelling, the corresponding scheduled times were excluded prior to modelling random arrivals.

3.3.3.4 Map Count

Partitioning by map count was motivated by the hypothesis that additional translation mappings increase processing complexity and influence service-time behaviour.

Referring back to Figure 3.3, in step 7, the Translation Service fetches a map for the message. A map is an XML information document related to the job (e.g., cost, shipment, details, etc.). Some maps may also contain programmatic loops. As different maps may influence message duration, different models are fitted based on the mapping used.

In practice, not every message is associated with a map, and many maps can be associated with one message. Table 3.5 shows the total count of maps for each split. From the analysis, the maximum number of maps for one message was 30.

Table 3.5: Normal Period: Map Count.

Split	Total Maps	Max Maps: Per Message
1	286,622	2
2	135,095	4
Other	14,556	30

3.3.3.5 Messages by Hour

Hourly partitioning was statistically motivated to reduce the impact of temporal non-stationarity and workload fluctuations.

To investigate time-dependent behaviour in EDI messages, the data is segmented by hour. The approach aims to mitigate the impact of non-stationarity where statistical properties such as mean and variance shift over time, by breaking the data into periods of reduced variability, thereby minimising correlation. Modelling by hour provides a clearer understanding of underlying patterns and improves the reliability of subsequent statistical analysis.

3.3.4 Parametric Modelling

Service and interarrival times are modelled using parametric distributions. Service time is defined as the duration required to process a message, from the moment it begins processing until it completes. Interarrival time refers to the time elapsed between the arrivals of successive messages into the queue.

Separating head and tail behaviour was motivated by the strong heavy-tailed characteristics observed in the empirical distributions.

As previously mentioned, a head-and-tail approach is considered for the distributions. For both service and interarrival times, the head of the data is defined as values less than or equal to one second. A range of candidate distributions (see Table 3.6) and data transformations (see Table 3.7) are considered to smooth the data and identify potential models that best fit the dataset. Parameters for each distribution are determined, after which AD GoF tests are applied to evaluate model suitability.

Table 3.6: Parametric Distributions.

Normal	Log	Log-Logistic	Logistic
Cauchy	Gamma	Burr	Inverse Burr
Exponential	Beta	Weibull	Pareto

Table 3.7: Data Transformations.

Log()	Sqrt()	Exp()
Log(log)	Sqrt(exp)	Sqrt(log)

3.3.5 Non-Parametric Modelling

When parametric methods are insufficient, non-parametric techniques like KDE are considered. With KDE, bandwidths are selected using various methods, including Silverman's Rule of Thumb, Sheather & Jones, Biased and Unbiased Cross-Validation, and the Direct Plug-in method.

3.4 Results

The following section presents the results of the analysis.

3.4.1 Message Classification

Referring back to the hypothesis question on message classification, Figure 3.4 shows the head of the data transformed for all messages > 0.000 milliseconds and < 1 second. Table 3.8 presents an attempt to manually classify messages into two groups: Group 1, corresponding to log-transformed values from –7 to –4 (0.001 to 0.018 milliseconds), and Group 2, ranging from –4 to –2.7 (0.019 to 0.036 milliseconds).

Table 3.8: Message Classification: Head Data - Group Comparison.

Group	**Milli-Seconds**	**Doc Type Count**	**Max Map Count**	**File Size Bytes**	**Splits**	**Translation Action**	**EDI Types**
1	0.001: 0.018	2	2	60:50 million	0–1521	Defer, Doc Extract, TX	X12, Edifact, Other, Idoc, Eancom
2	0.019: 0.036	2	2	60:11 million	0–889	Defer, Doc Extract, TX	X12, Edifact, Other, Idoc, Eancom

Comparison of the two groups in Table 3.8 indicates that Group 1 contains messages with larger byte sizes than those in Group 2. The number of message

splits in Group 1 differs from that in Group 2. The more times the message is split, the faster the duration. Thus, it can be concluded that messages in Group 1 are characterised by short processing durations, and this may be due to the number of times the messages are split. Group 2's smaller split count may explain the increased processing time for the messages.

Further analysis is attempted to manually identify distinct message classes based on Figure 3.5. These messages can be approximately grouped into distinct categories, as shown in Table 3.9.

Table 3.9: Message Classification Grouping.

Group	Start	End	Range (ms)	Mean (ms)	Customers
1	-7	-4	0.001 – 18	0.005	651
2	-4	-2.7	18 – 67	0.036	635
3	-3.0	-2.2	49 – 110	0.307	577
4	-2.0	-1.7	137 – 182	0.158	429
5	-1.7	-1.5	182 – 223	0.205	417
6	-1.5	0	223 – 999	0.404	524

These results suggest that EDI messages are not generated from a single homogeneous process, but instead consist of multiple overlapping behavioural classes. The differing timing characteristics indicate substantial heterogeneity within the dataset. This violates the assumption of distributional simplicity commonly assumed in classical queueing and parametric modelling approaches. Consequently, modelling the full dataset using a single parametric distribution is unlikely to adequately capture the underlying structure of the data.

Manual analysis of additional features, including "mapName", "receiverID", "sourceFileSize", "sourceMessageID", "docCount", "sourceFilename", "docExtractMapUsed", "typingMapUsed", "ackStatus", "hostName", "actionCategory", "transactionResponse", and "businessSender" was conducted. However, no clear groupings emerged, highlighting the complexity of distinguishing meaningful structure or uniqueness among the messages.

Table 3.10 presents a summary of how file sizes may relate to message splitting. When large files entered the system and were subsequently split, the resulting fragments were not always uniformly distributed. Inconsistencies were also found in the tagging of file size elements, suggesting unreliable or non-standardised metadata capture. Variability in metadata reinforces the

value of direct behavioural features, such as timing (e.g., service duration), for classification purposes.

Table 3.10: Summary: File Size Analysis.

	File Size Condition				
	Mode = Document	**FileSize = SourceFileSize**	**SourceFileSize is NULL**	**SourceFileSize > FileSize**	**FileSize > SourceFileSize**
Count	860,201	192,734	312,693	226,093	441,454

3.4.2 EDI Modelling

3.4.2.1 Normal and Busy Periods

Table 3.11 shows summary statistics for the service and interarrival times for each queue during both normal and busy periods. The maximum durations of both service and interarrival times are 458.44 and 446.84 seconds, respectively.

Table 3.11: Normal/Busy Period - ST & IAT in Seconds.Milliseconds.

	Normal Period		**Busy Period**	
	CMD Queue	**Data Queue**	**CMD Queue**	**Data Queue**
Service Time (s)				
Min	0.00	0.00	0.00	0.00
Mean	0.04	0.04	0.04	0.04
Max	22.80	22.80	458.44	458.44
Interarrival Time (s)				
Min	0.00	0.00	0.00	0.00
Mean	0.04	0.04	0.04	0.04
Max	22.43	22.43	446.84	446.81

Figures 3.7 and 3.8 show the histograms over the full range contain a single bin where the majority of the data is concentrated. On this scale, adding more bins does not change the histogram's shape, since most messages took under one second to process. These figures do not provide sufficient detail to discern meaningful patterns, aside from indicating a long tail in the data.

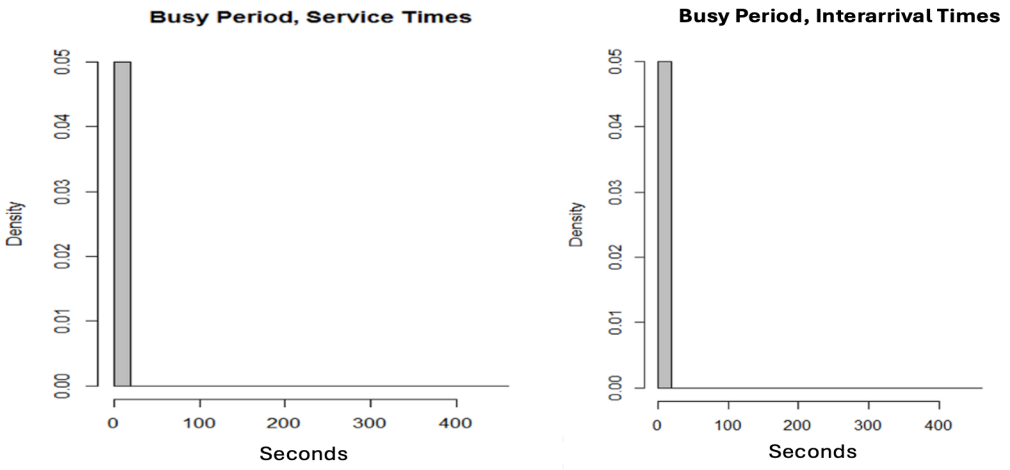


Figure 3.7: Histogram: Busy Periods, IAT and ST.

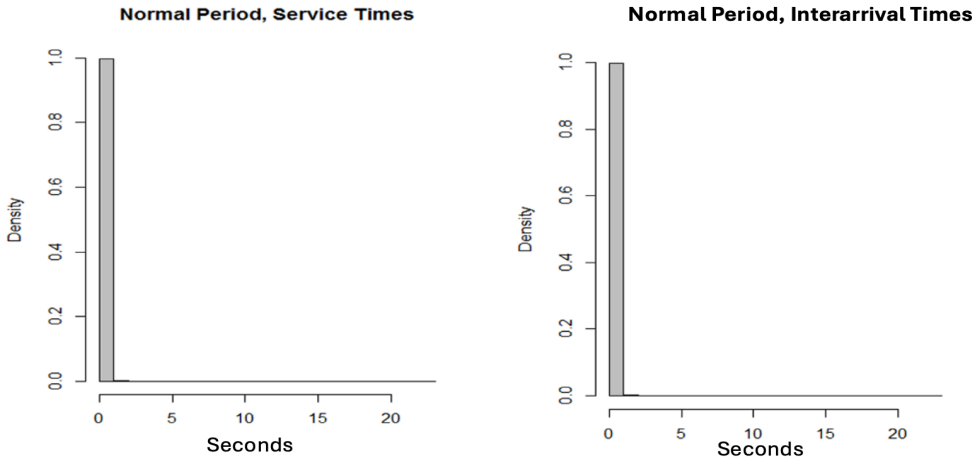


Figure 3.8: Histogram: Normal Periods, IAT and ST.

Segmenting the data using different EDI modelling approaches reveals more discernible and informative patterns as seen in Section 3.4.2.2.

3.4.2.2 Message Split Count

The service and interarrival times are divided into head and tail components. Figure 3.9 shows the splits for both STs and IATs. The bottom row uses coarse binning, providing minute-level granularity that may obscure finer patterns. In contrast, the top row uses finer bins, revealing more structure in short-duration messages.

The improved clarity in the distribution tails results from the combined effect of finer time resolution and segmentation by bundle size. These adjustments highlight patterns that might otherwise be lost, particularly in bursty traffic where short messages are easily smoothed over.

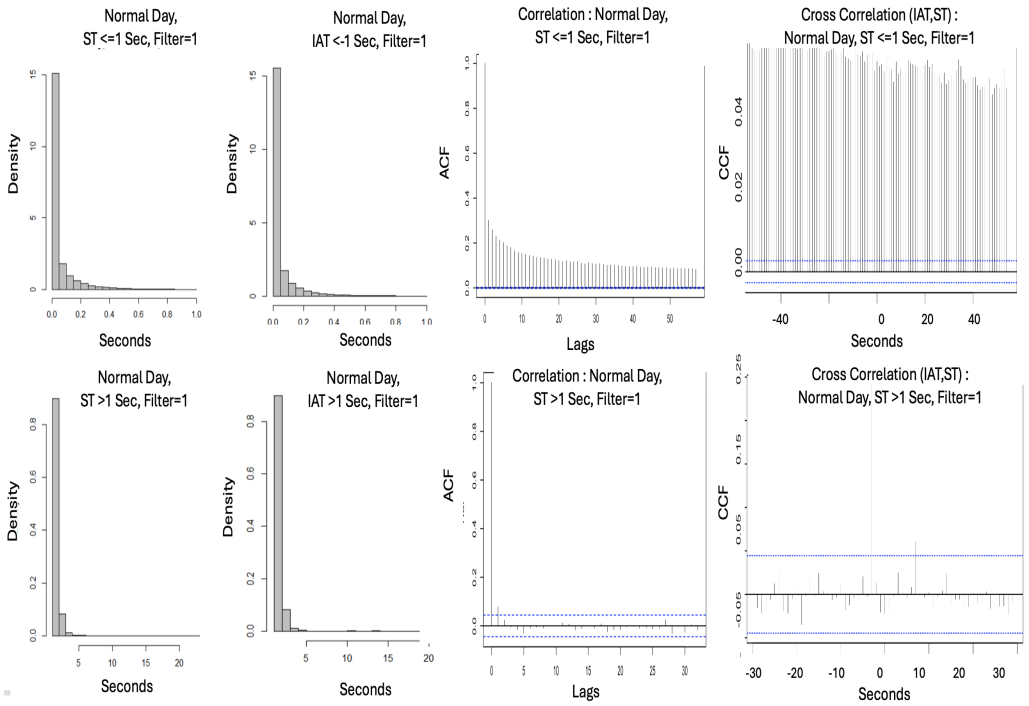


Figure 3.9: Messages - Heads and Tails.

To assess whether messages are independent and identically distributed (i.i.d.), a key assumption for modelling service and interarrival times, autocorrelation (ACF) and cross-correlation (CCF) analyses were performed. Partial autocorrelation (PACF) analysis was not required because the focus was on assessing whether the message times are independent rather than identifying the specific lag structure. The ACF plots indicate persistent temporal dependencies, particularly between service and interarrival times, with significant correlations beyond 60 lags where the service times are < 1 second. When service times exceed > 1 second, a statistically significant autocorrelation is observed only at the second lag, after which the correlations dissipate, indicating a lack of longer-term temporal dependence.

The cross-correlation analysis between service and interarrival times shows that for service times $<= 1$ second, strong and persistent cross-correlations

are observed across multiple lags, indicating a strong relationship between message arrival and process duration. Where service times are > 1 second, cross-correlations are weak. However, a statistically significant spike appears near lag 0, indicating that elevated message traffic has an immediate impact on service time performance. A second correlation at lag 7 may be a delayed feedback effect, where prior service performance influences future message arrivals.

Correlation was further examined by stratifying messages by split count. As shown in Table 3.12, no correlation is evident in the tails, except for interarrival times at split="2", whilst the head displays notable correlation. The region is modelled cautiously unless the dependencies can be removed or adequately explained.

Table 3.12: Correlation Checks by Split.

	Correlation Indicated: ACF		
Test	**Split = 1**	**Split = 2**	**Split = Other**
ST $\leq$ 1 Second	True	True	True
IAT $\leq$ 1 Second	True	True	True
ST $>$ 1 Second	False	True	False
IAT $>$ 1 Second	False	False	False

Note. The 95% confidence cut-off point was 0.00 for $n \leq 1$ second and 0.04 for $n > 1$ second, across all splits.

3.4.3 Correlation

3.4.3.1 Hurdle Modelling

Zero-duration transaction messages were removed from the head of the data to enable hurdle modelling, as standard transformations require either shifting or excluding zeros. Even after transactions of zero duration were removed leaving only service times > 0 and < 1 second, the distribution (as shown in the left histogram of Figure 3.10) does not change substantially. Interestingly, removing these zeros from the data does not remove the correlation, as seen to the right in Figure 3.10.

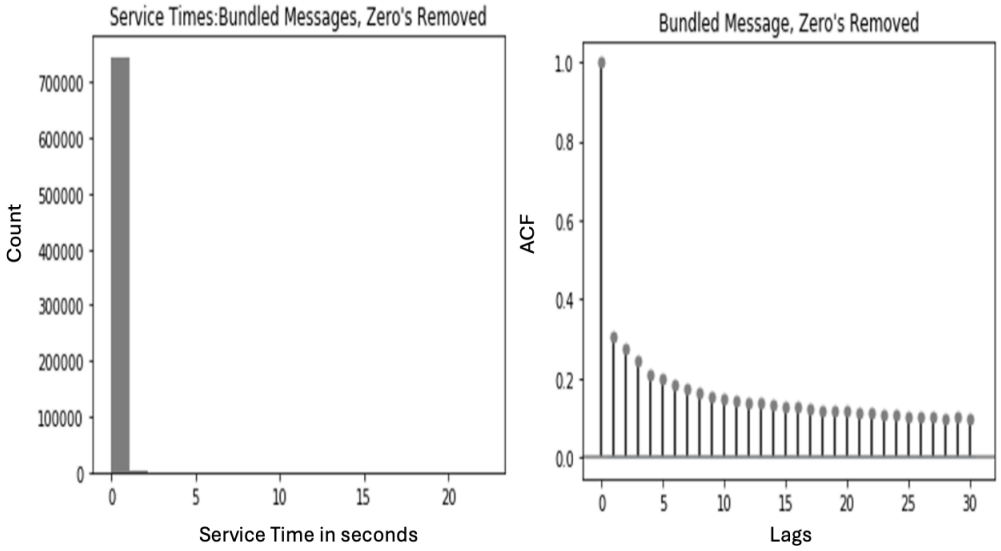


Figure 3.10: Distribution Head : Message Bundle with Hurdle-Type Implementation.

To further investigate the correlation in service times, the data is explored using alternative partitioning methods as discussed in the next set of sub-sections.

3.4.3.2 Message Bundle

When modelling by message bundle, the analysis showed that all messages up to and including the second-last message in each bundle were processed in zero seconds. Only the final part of each bundled message had a duration greater than zero seconds. Modelling was then conducted at the message bundle level to assess whether a correlation persisted. Figure 3.10 (introduced in the previous section) confirms that correlation persists even after modelling at the message bundle level.

Thus, bundles of messages alone are not a full explanation of the correlation in the data.

3.4.3.3 Scheduled Versus Un-Scheduled Messages

Referring back to Figure 3.6, distinct periods of higher than normal message frequency are evident, particularly at minute 00, minutes 30–33, and minutes 43–46. These bursts may contribute to correlation in the data. To assess their impact, scheduled messages were removed, and a correlation assessment was

re-evaluated. However, as shown in Table 3.13, this adjustment did not remove correlation.

Table 3.13: Service Times: Correlation Checks by Schedule.

	Hurdle Implementation	**Removed Minutes 00**	**Removed Minutes 30–33**	**Removed Minutes 43–46**
Correlation (ACF)	True	True	True	True

Different filtering techniques were applied to remove correlation. Table 3.14 is a summary of the different filtering techniques applied and the results of the correlation tests.

Table 3.14: Re-cap of ACF Correlation: Test Results.

Seq	**Correlation Exists?**	**Volume**	**%**	**Hurdle Distribution**	**Split = 1**	**Split = 2**	**Split = Other**	**00 Schedule Removed**	**30:33 Schedule Removed**	**43:46 Schedule Removed**	**<=1 Second**	**> 1 Second**
1	True	558,312	57	False	True	False	False	False	False	False	True	False
2	True	233,373	24	False	False	True	False	False	False	False	True	False
3	False	192,208	20	False	False	False	True	False	False	False	True	False
4	False	511,669	52	True	True	False	False	False	False	False	False	False
5	True	112,294	11	True	False	True	False	False	False	False	False	False
6	True	11,938	1	True	False	False	True	False	False	False	False	False
7	False	487,212	50	True	True	False	False	True	False	False	False	False
8	True	109,971	11	True	False	True	False	True	False	False	False	False
9	True	11,526	1	True	False	False	True	True	False	False	False	False
10	False	440,629	45	True	True	False	False	True	True	False	False	False
11	True	100,447	10	True	False	True	False	True	True	False	False	False
12	True	10,560	1	True	False	False	True	True	True	False	False	False
13	False	394,702	40	True	True	False	False	True	True	True	False	False
14	True	928,75	9	True	False	True	False	True	True	True	False	False
15	True	9,623	1	True	False	False	True	True	True	True	False	False
16	True	558,312	57	False	True	False	False	True	False	False	True	False
17	True	233,373	24	False	False	True	False	True	False	False	True	False
18	True	233,373	24	False	False	False	True	True	False	False	True	False
19	False	235	0	False	True	False	False	False	False	False	False	True
20	False	47	0	False	False	True	False	False	False	False	False	True
21	False	6	0	False	False	True	False	False	False	False	False	True
22	False	216	0	True	True	False	False	True	True	True	False	True
23	False	44	0	True	False	True	False	True	True	True	False	True
24	False	6	0	True	False	False	True	True	True	True	False	True
25	True	394,485	40	True	True	True	False	True	True	True	True	False
26	True	92,831	9	True	False	False	False	True	True	True	True	False
27	True	9,617	1	True	False	False	True	True	True	True	True	False

The organisation has over 600 customers using their system. Splitting the data by each customer was considered; however, that would require considerable effort and is not feasible from a modelling perspective. Ideally, the model should represent the majority of the dataset. Therefore, the data was initially split using a single customer as a guide to assess whether further splitting by additional customers was necessary. It was observed that the shape of the data

remained consistent, indicating that this technique is not feasible to implement from a holistic standpoint.

3.4.3.4 Map Count

To assess the impact of mapping on correlation, the data was partitioned by the number of maps associated with each transaction. Analysis revealed that 40% of transactions had no map name (i.e., no maps applied), while 56% had exactly one map. In total, 96% of transactions had at most one map, with only 3.5% having more than one. As shown in Table 3.15, splitting the data by map count showed evidence of weak correlation, except for messages with more than three maps, a group representing just 0.14% of the dataset. For example, among transactions with exactly one map, the correlation coefficient was $r = 0.07$, supporting the classification as weak.

Table 3.15: Service Time : Correlation Results by Map Count

	Map Count									
Metric	**All**	**1**	**2**	**>2**	**>1**	**<1**	**≤1**	**≤2**	**≤3**	**>3**
Messages	687,467	383,024	22,832	1,311	24,143	280,297	663,321	686,153	686,495	969
Percentage	100.00	55.72	3.32	0.19	3.51	40.77	96.49	99.81	99.86	0.14
Correlation	Weak	Weak	Weak	Weak	Weak	Weak	Weak	Weak	Weak	True

It appears that the number of maps is a contributing factor to the correlation being seen.

3.4.3.5 Message by Hour

The data was aggregated and modelled by hour to evaluate if aggregation could reduce the effect of correlation and to assess whether the data was stationary. However, correlation persists, as shown in Figure 3.11, which displays the ACF of service times starting at midnight. The plot reveals a significant correlation effect at shorter lags, which gradually diminishes but persists beyond lag 100, indicating correlation in the data.

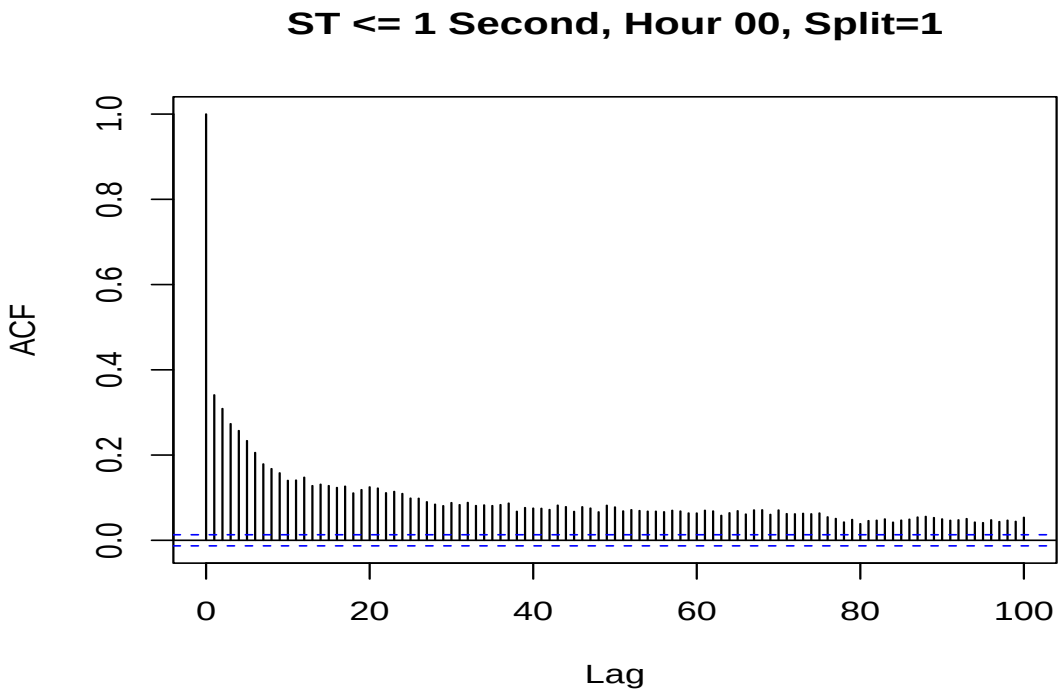


Figure 3.11: Correlation: Hour.

The data also shows evidence of longer-term trends. Table 3.16 indicates unequal mean and variance across time, which is an indication of non-stationarity.

Table 3.16: Service Times ≤1 Second, Split="1": Correlation Summary

Hour Range	Correlation	Mean Per Hour	Variance Per Hour
00–04	True	0.11	0.02
05–08	True	0.07:0.08	0.01:0.02
09–11	True	0.04:0.05	0.00

For instance, mean service times are consistently higher between midnight and 4 a.m. In non-stationary data, a correlation can arise from trends rather than short-term temporal dependencies.

3.4.4 Parametric Modelling

3.4.4.1 Modelling Service and Interarrival Times

Initially, parametric distributions were fitted to the entire dataset without filtering, and GoF was evaluated. However, only specific subsets of the data showed a reasonable fit. For instance, applying a filter (Split="1") to the tail, representing just 0.3% of the data, yielded an improved fit. In this case, a Hurdle-type model was used, and a simple transformation (adding a constant

of 1) was applied to address zero values that interfere with logarithmic GoF calculations.

Service Times

For this 0.3% subset, the CDF plot in Figure 3.12 compares the empirical and fitted Burr distributions. The accompanying P–P plot shows the empirical cumulative probabilities against the theoretical ones. The close alignment with the reference line indicates minimal deviation, suggesting a good fit.

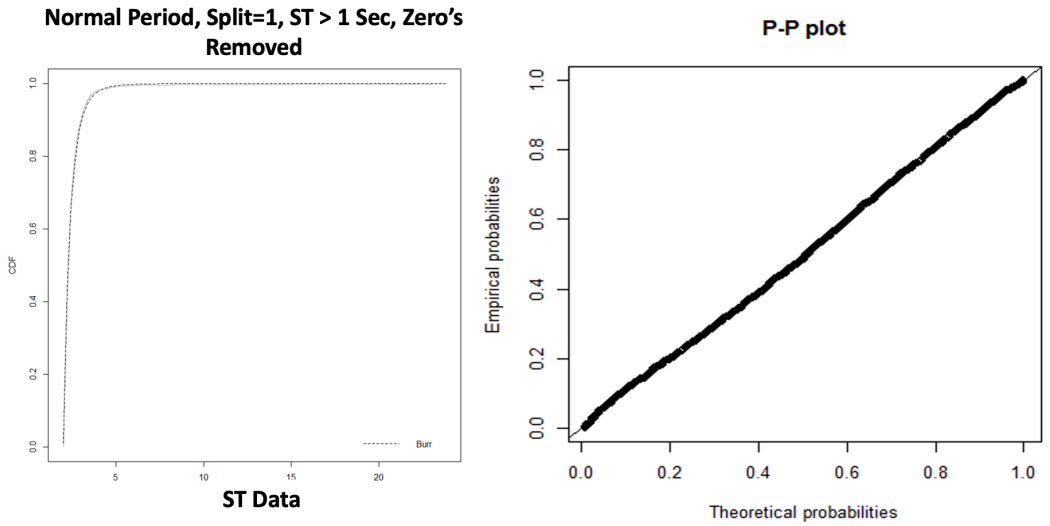


Figure 3.12: Normal Period, Split='1', Burr Fitting, Service Times - > 1 Second.

There is evidence to suggest that the filtered data can be drawn from a Burr distribution. The AD test statistic score of 1.2 in Table 4.13 indicates a reasonable fit.

Table 3.17: AD Test: Normal Period, ST, Tail of Data.

AD Score	P-Value	Test
1.2	0.3	Pass

For the head of the data, defined as service times less than or equal to one second, a parametric fit could not be achieved, regardless of the transformations applied or the use of data-splitting methods. Table 4.14 shows the AD test statistic results. An AD score of 3.5 or lower is considered sufficient to pass the AD GoF test; however, all test results produced significantly high test statistics.

Table 3.18: ST < 1, Filter = 1 : AD Test.

AD Test	**Data**	**Log (data+1)**	**Sqrt (data)**	**Exp (data)**	**Sqrt (log (data+1))**	**Log (log (data+1) +1)**	**Sqrt(exp(data))**
	AD Score						
Log-normal	4146	4279	4146	78917	4279	4427	78917
Log–logistic	1099486	153957	1703852	35541593	1778558	1198064	89237003
Gamma	681391	20247	7959	85258	6757	14646	81636
Weibull	410883	453321	615255	2912272	679932	492797	9870194
Exponential	131869	112365	9986	200079	10607	97906	216494
Cauchy	101117	101117	98590	42829	103782	96204	102433
Logistic	58318	54796	24413	63141	23026	51976	60585
Pareto	4300	4624	9986	200079	10607	4936	216494
Burr	4488	4682	4487	54435	4682	4903	54470
Inv Burr	5210	5651	5216	57655	5690	6089	57675

The majority p-values round to 0.00

Parametric fitting was also attempted on subsets of the data grouped by map count, as prior analysis indicated that splitting service times by map count reduced correlation. Table 3.19 presents the AD GoF results. Despite applying square root and exponential transformations, the resulting AD scores were exceptionally high, and none of the fits satisfied the AD test criteria.

Table 3.19: Map Counts = All - AD Tests.

	Data+1		**Sqrt(data+1)**		**Exp(data+1)**	
Transformation	**AD Test**	**P-Value**	**AD Test**	**P-Value**	**AD Test**	**P-Value**
Log-normal	Inf	0.00	Inf	0.00	Inf	0.00
Log–logistics	53231368	0.00	132692775	0.00	Inf	0.00
Gamma	Inf	0.00	Inf	0.00	Inf	0.00
Weibull	Inf	0.00	40223101	0.00	Inf	0.00
Exponential	273165	0.00	293850	0.00	Inf	0.00
Cauchy	136839	0.00	135069	0.00	140489	0.00
Logistic	Inf	0.00	Inf	0.00	Inf	0.00
Pareto	278707	0.00	293850	0.00	Inf	0.00
Burr	Inf	0.00	Inf	0.00	Inf	0.00
Inverse Burr	inf	0.00	Inf	0.00	Inf	0.00

The table excludes messages with no maps. 40% of messages have no maps.

Interarrvial Times

Similar tests were conducted on the tail of the interarrival times, specifically where the interarrival time exceeds one second. For example, distribution

fitting was attempted on the interarrival times with Split= "1" and without applying a Hurdle-type model. As shown in Table 3.20, the AD test results indicate that the filtered data does not conform to a parametric distribution. However, both the untransformed data and the square root transformations (highlighted in bold) show relatively close alignment with a Burr distribution, though neither meets the AD test criteria.

Table 3.20: IAT > 1, Filter = 1 : AD Test.

AD Test	**Data**	**Log (data+1)**	**Sqrt (data)**	**Exp (data)**	**Sqrt (log (data+1))**	**Log (log (data+1) +1)**	**Sqrt(exp(data))**
	AD Score						
Log-normal	Inf	55	Inf	Inf	55	48	Inf
Log–logistic	16000	45000	59000	Inf	120000	80000	Inf
Gamma	34000	87	Inf	6600	66	61	7100
Weibull	Inf	6600	4800	Inf	24000	18000	.
Expon-ential	480	610	670	Inf	740	680	Inf
Cauchy	110	92	95	170	82	83	140
Logistic	Inf	56	Inf	Inf	46	46	Inf
Pareto	8200	610	670	Inf	740	680	Inf
Burr	**3.9**	6.6	**4**	Inf	6.6	8.5	Inf
Inv Burr	20	15	20	Inf	15	14	Inf

The majority of p-values round to 0.00

Despite applying log transformations and Hurdle-type models, most distributions failed AD GoF tests, particularly for the head of the data. Tests for a uniform distribution on scaled file sizes yielded inconsistent outcomes across statistical methods; for example, they failed the AD GoF tests but passed the chi-square test.

3.4.5 Non-Parametric Modelling

As no suitable parametric model was identified, KDE was applied to the head of the data where Split = "1". Model fit was assessed visually and statistically, following established guidelines for density estimation evaluation [145]. In this context, a "good fit" was defined as one in which the estimated density closely follows the observed data distribution, minimises unexplained white space, and accurately represents both head and tail behaviour.

However, the resulting histograms showed excessive white space, regardless of bandwidth, kernel, bin count, or custom breakpoints, indicating poor fit. The dataset was then reduced to a sample of fifty thousand messages, and k-fold cross-validation was implemented prior to reapplying KDE. As shown in Figure 3.13, the fit improves in the tail, but remains inadequate in the head, particularly within the first bin, where the model appears to under-predict the data.

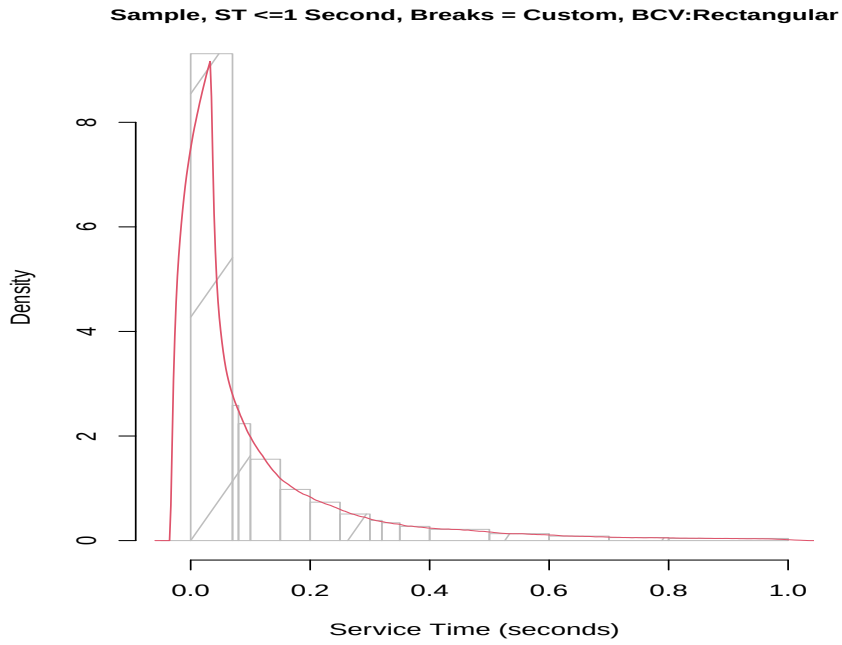


Figure 3.13: KDE: ST<=1 Second.

Encouraged by these results, the data was partitioned by hour. Figure 3.14 presents histograms for each hour across a twelve-hour period, starting at midnight, with the KDE density overlayed on the histogram. For each hourly dataset, multiple bandwidth selectors and kernel functions were evaluated within an iterative loop. The best KDE configuration was determined by selecting the kernel and bandwidth that minimised MISE, providing an objective criterion for model fit. While MISE determined the optimal KDE parameters, custom histogram breakpoints were subsequently chosen to improve the visual alignment between the histogram and the estimated density, particularly in regions of high skewness and sparsity.

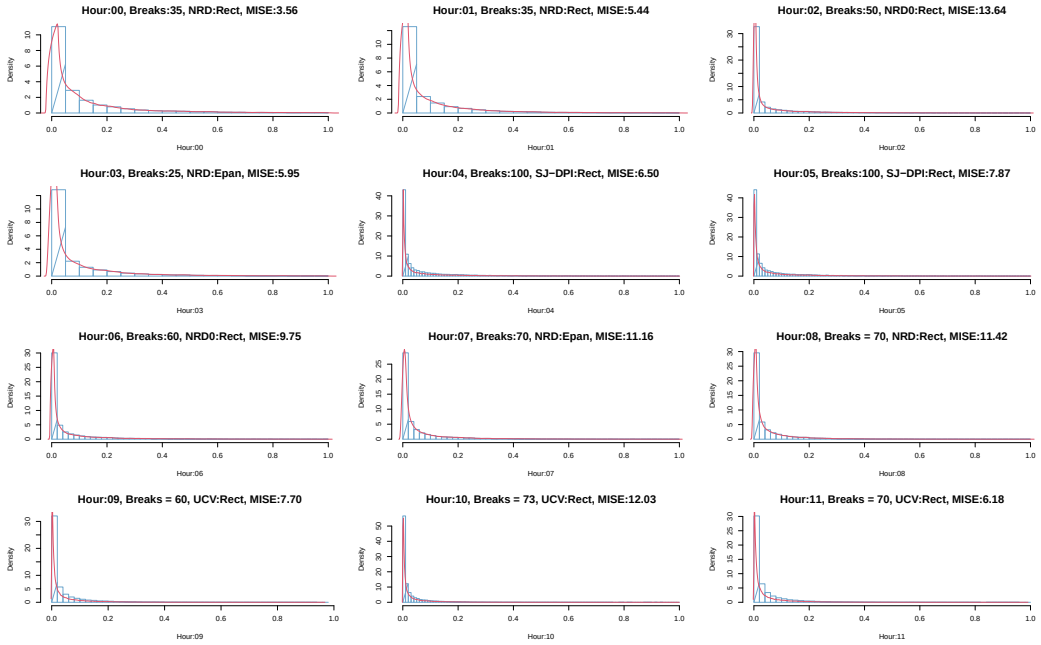


Figure 3.14: KDE: Fitting by Hour.

Table 3.21 summarises the best-performing KDE for each hour, based on the minimum MISE. Lower MISE values correspond to better fits, while the highlighted hours reflect cases where the error was comparatively high. Although KDE demonstrated reasonable performance overall, its application at an hourly granularity is not scalable, as it would require maintaining 24 separate models, one per hour.

Table 3.21: Summary: KDE Fitting by Hour (Best MISE Values).

Hour	**00**	**01**	**02**	**03**	**04**	**05**	**06**	**07**	**08**	**09**	**10**	**11**
Custom Breaks	35	35	50	25	100	100	60	70	70	60	73	70
Selector	NRD : Rect	NRD : Rect	NRD0 : Rect	NRD : Epan	SJ-DPI : Rect	SJ-DPI : Rect	NRD0 : Rect	NRD : Epan	NRD : Rect	UCV : Rect	UCV : Rect	UCV : Rect
MISE	3.56	5.44	13.64	5.95	6.50	7.87	9.75	11.16	11.42	7.70	12.03	6.18

3.5 Discussion

The following analysis revisits the research questions outlined in Section 3.1. Most transactions were processed in under one second and exhibited high discreteness, motivating a division into head and tail components.

3.5.1 Message Classification

The third research question aimed to determine whether EDI messages can be effectively classified. Analyses of the head of the data, up to 0.036 milliseconds, enabled the classification of messages into two distinct groups, with Group 1 exhibiting shorter processing times than Group 2. The distinction between the two groups was the frequency of message splitting and the corresponding byte size of the messages. Further work is required to improve message classification, particularly due to the number of different ways one can classify these messages.

3.5.2 Normal and Busy Periods

The data was segmented into normal and busy periods for modelling, as the latter deviated from standard behaviour. Despite segmentation, neither period supported effective parametric or non-parametric fits without further partitioning. Applying EDI-specific segmentation techniques enabled more accurate fits for subsets of the data. Persistent correlation remains a concern and limits generalisability. Thus, the first research question, whether EDI messages can be modelled parametrically or non-parametrically, has been partially addressed.

3.5.3 Message Split Count

The message split count was used to help split the head of the data. It was further noted that the correlation was not fully explained by splitting on this feature. For the Split="1" group, the tail of the data fitted a parametric distribution, whilst the head of the data required further effort. These findings contribute toward answering the first research question: Can EDI messages be modelled using parametric, or, failing that, non-parametric techniques?

3.5.4 Correlation

3.5.4.1 Hurdle Modelling

A Hurdle-type model was applied to address overdispersion by removing zero-duration messages, focusing the analysis on events of positive duration. While this improved model fit, it was insufficient to pass the AD GoF test and did not account for the observed correlation. Business feedback indicated interest primarily in messages exceeding five minutes, suggesting that modelling should target stakeholder-relevant subsets. These findings contribute toward answering

the first research question by highlighting the importance of aligning modelling efforts with practical use cases.

3.5.4.2 Message Bundle

Some messages arrived in bundles, with only the final message in each bundle having a non-zero duration. Modelling each message individually introduced a correlation and overdispersion due to their time dependence. To address this, bundled messages were grouped and considered as single units prior to modelling. Whilst this reduced the correlation, it did not eliminate it. Nevertheless, bundling offers a partial insight into the first research question by enhancing the understanding of overall service and interarrival times, particularly for heavily split messages.

3.5.4.3 Scheduled Versus Un-Scheduled Messages

Scheduling of messages is important for modelling queue behaviour and burstiness. Understanding the scheduled load versus random load helps support queue capacity planning. Removing the scheduled messages in isolation did not eliminate the correlation, nor did it support parametric modelling. These findings contribute toward answering the first research question: EDI messages can have a scheduled component that may require separate modelling.

3.5.4.4 Map Count

A map is a useful feature of an EDI message. These maps have many different authors and editors. Depending on the skill level of the map editor and the content within these maps, they may take different times to process. Given that map count may serve as a proxy indicator for message complexity, it was hypothesised that it could help explain the presence of a correlation. A combination of data partitioning by map count and the use of message bundling was found to yield uncorrelated data. These findings contribute to answering the first research question.

3.5.4.5 Message by Hour

EDI messages were aggregated by hour to evaluate whether temporal segmentation could improve modelling outcomes and reduce the correlation. Correlation

persisted across all hourly intervals. Additionally, the distribution of service times varied by hour, with elevated mean and variance values observed. The observed variability indicates non-stationarity, in which statistical properties shift over time, complicating the assumptions required for parametric modelling. Overall, this suggests that hourly segmentation alone is not sufficient. More targeted approaches are needed.

The persistent autocorrelation observed across multiple partitioning strategies indicates that the message traffic is not independent and identically distributed (i.i.d.). These dependencies likely arise from bursty traffic behaviour, message fragmentation, and shared processing workflows within the EDI infrastructure. Such behaviour violates a key assumption underlying many classical queueing models and GoF techniques, which assume independent observations. As a result, standard modelling approaches may underestimate variability and produce misleading distribution fits when applied directly to the raw data.

3.5.5 Parametric Modelling

Attempts to model the full range of service times, including the head of the distribution, did not yield a good fit with any known parametric distribution, despite many transformations and EDI specific splitting techniques. In contrast, the tail did fit a Burr distribution. Whilst interarrival times in the tail showed a similar trend, they failed the AD GoF test.

These findings suggest that parametric modelling is only appropriate for a narrow portion of the data. The overall dataset appears to exhibit heavy-tails with discrete characteristics, limiting the effectiveness of standard continuous parametric models. More targeted segmentation or alternative modelling strategies are required.

The inability to obtain acceptable GoF results for the head of the data suggests that the observed service times do not conform to standard parametric assumptions. Quantisation, burstiness, and message heterogeneity collectively distort the empirical distribution, producing behaviour that is difficult to represent using a single continuous distribution. These findings motivate the later chapters of this thesis, which investigate the impact of quantisation on fitting accuracy and explore methods to reconstruct or better model the underlying continuous behaviour.

The second research question has been partially addressed: Can the service times and interarrival times of an EDI message be modelled by a parametric method or a non-parametric method?

3.5.6 Non-Parametric Modelling

Some messages arrived as part of a bundle, where only the final message had a non-zero duration. Modelling each message individually introduced correlation and overdispersion due to temporal dependence and the shared context within message bursts. While independent modelling assumes that observations are i.i.d., real-world message flows often violate this assumption due to system-level interactions and batching effects.

Although overdispersion can be mitigated, doing so depends on accurately identifying its sources. When those factors are complex or unobservable, mitigation becomes more challenging.

To address this, bundled messages were grouped and considered as single units before modelling. Whilst this reduced the correlation, it did not eliminate it. Nevertheless, bundling offers a partial insight into the first research question by enhancing the understanding of overall service and interarrival times, particularly for heavily split messages, and may aid developers in performance analysis.

Table 3.22 summarises the primary data characteristics observed throughout the analysis and the corresponding modelling assumptions they violate.

Table 3.22: Observed Data Characteristics and Violated Modelling Assumptions

Observed Behaviour	**Violated Assumption**
Bursty arrivals	Independence of arrivals
Message fragmentation	Independent observations
Heavy-tailed distributions	Distributional simplicity
Quantised timestamps	Continuity of observations
Changing hourly behaviour	Stationarity
Heterogeneous message classes	Homogeneity of populations

3.5.7 Relative Impact of Data Characteristics

The modelling challenges observed throughout this chapter arise from several interacting data characteristics, including burstiness, message heterogeneity,

heavy-tailed behaviour, and quantisation. While these effects collectively influence distribution fitting and statistical inference, they originate from different sources and affect the modelling process in distinct ways.

Burstiness and message heterogeneity are characteristics of the EDI messaging system. A single inbound transaction may generate multiple downstream messages, producing correlated bursts of activity and non-independent arrival patterns. Additionally, different message schemas exhibit distinct structural and temporal behaviours, resulting in heterogeneous messages that are difficult to model using a single parametric distribution. These characteristics violate assumptions of independence and homogeneity commonly assumed in classical queueing and statistical modelling approaches.

Quantisation introduces additional distortions that are not inherent to the underlying message generation process. Rounded timestamps collapse nearby observations into identical values, masking fine-grained temporal variation and introducing excessive ties into the data. This affects the empirical shape of the distributions and violates continuity assumptions underlying many GoF tests and parameter estimation techniques. The effects are particularly visible in the head of the data, where quantised observations dominate the distributional structure.

The analysis suggests that quantisation is a major contributor to the instability observed in GoF testing and parametric fitting accuracy. However, quantisation alone does not fully explain the modelling difficulties, as burstiness and heterogeneity also contribute to correlation structures, heavy tails, and multimodal behaviour within the dataset.

These findings motivate the subsequent chapters of this thesis, which investigate the effects of quantisation on parameter estimation, GoF testing, and distribution reconstruction in greater detail.

3.6 Conclusion

The study assessed the suitability of parametric and non-parametric models for B2B EDI message processing times. Due to the dataset's complexity, correlation, and discrete features, a hybrid approach was necessary. Parametric models fit the distribution tails, whilst non-parametric techniques like KDE

performed well under segmented (e.g., hourly) conditions. Correlation patterns were driven by message splitting, bundling, scheduling, and map configurations.

Fitting to this data is challenging, even without accounting for correlations; additional challenges will be addressed in later chapters. Although no single model proved sufficient, targeted segmentation emerged as a more effective strategy for EDI analysis. Further work is needed to refine the modelling framework and address the heterogeneity of EDI messages in cloud-based supply chain networks.

CHAPTER 4

Hetrogeneous Message Modelling

The chapter extends the statistical modelling of EDI messages introduced in Chapter 3. It presents a granular framework for feature identification, classification, and modelling, using both parametric and non-parametric techniques to analyse EDI message interarrival and service times. Key performance indicators (KPIs) are defined to assess queue utilisation. The dataset is partitioned along multiple axes, such as message type and temporal distribution, to enhance model efficacy. The research places special emphasis on message interdependence, queue behaviour, and the impact of quantisation noise. The results demonstrate that no single modelling approach is sufficient due to the data's heterogeneous, heavy-tailed nature. Parametric models are most effective for capturing tail behaviour, whilst non-parametric KDE methods are better suited for modelling the head of the distribution. Quantisation noise affects model fidelity. These findings provide practical insights for designing adaptive message ingestion pipelines and performance-aware queuing systems in large-scale B2B supply chain infrastructures. The modelling approach described could improve system reliability and operational intelligence in Enterprise Messaging environments operating under Ambient Intelligence (AmI) principles.

4.1 Introduction

Stress testing of queuing systems is essential for identifying functional and performance-related issues in large-scale infrastructures. Accurate modelling relies on understanding the order, volume, pace, and dependencies of message flows, as these features are critical for fitting appropriate statistical distributions.

Accurate event timing plays a central role in analysing and modelling queue behaviour. Rounding or truncation of timestamps during message logging can obscure the true variability in event timing. Such distortions introduce quantisation noise and complicate the fitting of appropriate statistical distributions [146]. Reliable modelling depends on capturing real-time characteristics, including the temporal ordering of message arrivals, a factor first explored by Lampart [9].

Effective modelling of queue behaviour requires the identifying characteristics that determine whether an EDI message exhibits bursty behaviour. For instance, a single transaction may generate hundreds of EDI messages in quick succession, raising the question of whether the system is processing individual events or bursts. These bursts may not be due to file size but may be due to other characteristics of the message, such as an EDI 810 (invoice), which may be triggered immediately after an EDI 856 (advanced shipment notice), leading to a burst of follow-on messages. Queuing systems must therefore be designed to accommodate both isolated messages and high-volume bursts without compromising performance.

The existing literature addresses EDI messages across a range of domains. However, limited attention has been given to performance testing through simulation, particularly in the context of queuing models and the analysis of interarrival and service times. Bursty behaviour in EDI traffic remains largely unexplored, particularly in identifying such patterns using message attributes. The lack of attention presents a clear gap in current research.

Building on the modelling limitations identified in Chapter 3.1, including burstiness, message heterogeneity, heavy-tailed behaviour, interdependence, and quantisation effects, this chapter proposes a structured framework for modelling heterogeneous EDI message traffic.

To address this gap, a framework has been developed to support the performance testing of EDI messages by understanding message behaviour through interarrival and service time modelling. The framework applies multiple methods to analyse various EDI message attributes, thereby supporting robust stress testing of queuing systems in enterprise environments. Methodologies are used to structure the data appropriately for modelling.

Low-level modelling of EDI messages has not been extensively explored in existing literature, despite its potential to enhance performance evaluation for supply chain systems. Such modelling is also relevant to Industry 4.0 initiatives, which mark a transition toward autonomous, sensor-driven, and self-regulating systems. The initiative supports real-time coordination and the creation of new data-driven business models. Within this context, low-level modelling of EDI messages provides a means to evaluate and enhance supply chain performance. Analysing historical and real-time message patterns can help anticipate workload intensity, detect bursts, and forecast queue build-ups, supporting proactive resource allocation and capacity planning. The G/G/1 queuing model is proposed as a suitable structure for representing heterogeneous EDI message interarrival and service times.

For low-level analysis, subsidiary attributes under the message specification syntax and format categories, such as file size, source file size, byte count, and XML maps, are also examined. Additional features, including message actions and categories, are used as novel indicators for classifying bursty versus non-bursty messages. These attributes are further employed to partition the dataset, enabling more accurate modelling of the underlying queue behaviour.

A framework for simulating queueing system performance by modelling interarrival and service times based on key EDI message attributes is presented in this chapter. The following is investigated:

1. The presence of malformed messages within the queueing system is evaluated.
2. Potential interdependencies between individual messages are examined.
3. Methods for partitioning the dataset, such as segmenting by head and tail regions, are assessed to support accurate distribution modelling.

4. The analysis investigates whether quantisation noise is present in event timestamps.

These steps contribute to a more reliable basis for performance testing in EDI-driven environments.

Building on the limitations identified in Chapter 3.1, including burstiness, message heterogeneity, heavy-tailed behaviour, interdependence, and quantisation effects, this chapter investigates structured modelling strategies for heterogeneous EDI message traffic. The results presented previously demonstrated that fitting single parametric distributions to complete datasets often produced poor GoF behaviour and failed to adequately capture the underlying variability of the message data. This chapter therefore proposes a modelling framework that applies partitioning and transformation strategies to better represent the differing statistical characteristics present within the data.

4.2 Data Overview

The analyses presented in this chapter is based on the dataset described in the Data Overview section of Chapter 3.

4.3 Methods

A detailed understanding of the structural makeup of EDI messages is essential for accurate modelling and system-level analysis. The internal structure of a standard EDIFACT message is examined first. As mentioned in Section 2.5, EDIFACT is an international standard for the electronic exchange of invoice messages. Selected segments are described to illustrate the diversity of fields present and their respective semantic roles in message interpretation. Figure 4.1 presents an example of a typical EDIFACT invoice message.

```
UNB+UNOA:1+01010000253001+O0013000093SCHA-Z59+991006:1902+PAYO0012101221'
UNH+1+INVOIC:D:97A:UN'
BGM+381+1060113800026+9'
DTM+137:199910060000:102'
NAD+BT+XXX MOTORS LTD::91'
RFF+VA:382324067'
NAD+SU+2002993::92'
RFF+VA:123844750'
CUX+2:EUR'
PAT+1'
DTM+140:19991031:102'
LIN+++090346642:IN'
QTY+12:54:PCE'
MOA+203:1960.29'
PRI+AAA:3630.1724::NTP:100:C62'
RFF+SI:165480'
DTM+11:199909280000:102'
RFF+ON:X18V00003'
```

Figure 4.1: EDIFACT Message.

The core components of the EDIFACT message is summarised in Table 4.1. INVOIC is the standard EDIFACT message type for invoices [56].

Table 4.1: EDIFACT INVOIC Segment Descriptions.

Segment	**Description**
UNB	Is the interchange (envelope) header.
UNH	The message header. The `INVOIC` identifier interprets the message as an invoice. The letter "D" signifies draft status, "97" refers to the year of update, and "A" corresponds to the first half of that year.
BGM	Provides the message name, reference number, and transaction type. Code 9 indicates an initial transmission.
DTM	Specifies the date, time, and period. Code 137 denotes the issuance date, while Code 102 adopts the CCYYMMDD date format.
NAD	Encodes the name and address information.
PRI	Indicates pricing details. Code `AAA` identifies the net price inclusive of allowances and charges.

For transactional modelling in queuing systems, it is necessary to trace the lifecycle of a message, including the timing of events across different processing stages, to enable the analysis of interarrival and service times. Chapter 3 presented the complete message pathway, as shown previously in Figure 3.2. Expanding on that, Figure 4.2 illustrates a detailed subset of the traversal

path, highlighting event timestamps across processing stages. Although certain internal processing steps may complete earlier, the image is intended to trace the end-to-end timing of a message, from initial entry at the SAP server at 16:00:36.555 to final dispatch to the outgoing mailbox at 16:00:37.395. The diagram provides millisecond-level temporal resolution, enabling the analysis of timing across queues and intermediate stages.

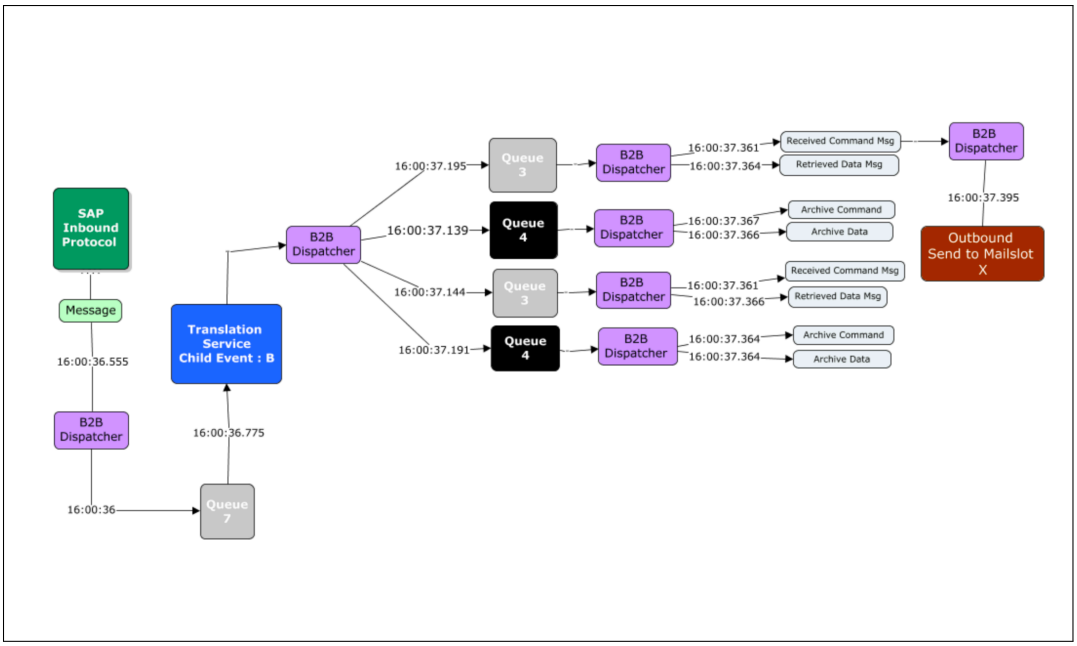


Figure 4.2: Time Stamp: Message Traversal.

Focusing on queue analysis, the system contains multiple queues, such as "Queue 3" and "Queue 4" shown in Figure 4.2. However, an examination of their behaviour revealed that they are similar, therefore, the discussion in the remainder of this chapter focuses on "Queue 3". Timestamp annotations in the diagram mark the transition points at each process node, enabling granular analysis of message latency.

To support accurate transactional modelling based on system logs, relevant timestamp data must be extracted. Section 4.3.2 outlines the framework adapted to perform this extraction.

4.3.1 Proposed Modelling Framework

The proposed framework represents a methodological contribution of this thesis.

Table 4.2: Proposed Heterogeneous Message Modelling Framework

Framework Stage	**Purpose**
Feature Identification	Extract operational, structural, and temporal attributes from EDI logs and XML schemas.
Feature Classification	Organise message attributes into hierarchical modelling categories based on behavioural and schematic properties.
Partitioning	Segment heterogeneous datasets using domain-driven and statistically motivated criteria.
Transformation	Apply transformations and preprocessing techniques to reduce skewness, heavy tails, and quantisation effects.
Parametric Modelling	Fit continuous probability distributions to partitioned message interarrival and service times.
Non-Parametric Modelling	Apply KDE and empirical modelling techniques where parametric fitting is insufficient.
GoF Evaluation	Evaluate fitted models using statistical GoF methods such as AD, CvM, and KS tests.
Quantisation Analysis	Assess the impact of timestamp rounding and discretisation on modelling accuracy and statistical inference.

The partitioning strategies applied throughout this chapter are not arbitrary. Partitioning was guided by both domain-driven and statistical considerations. Domain-driven partitions reflect operational differences between EDI transaction categories, batching behaviour, and message fragmentation characteristics. Statistical partitioning was additionally applied where heavy-tailed behaviour, multimodality, non-stationarity, or quantisation effects reduced the suitability of fitting a single distribution across the complete dataset. The objective of partitioning was therefore to improve modelling representativeness while preserving operational interpretability.

The framework proposed in this chapter does not directly eliminate quantisation effects or reconstruct the original continuous observations. Instead, it provides a structured approach for modelling heterogeneous and quantised data in the presence of rounding, burstiness, and heavy-tailed behaviour. The framework therefore aims to improve modelling representativeness and fitting stability while acknowledging the underlying limitations introduced by quantised observations.

4.3.2 Framework

Modelling EDI messages requires identifying features that best support analytical and system-level requirements. Structured and unstructured log datasets can contain millions of lines of text, with a single message often spanning thousands of entries across multiple log files on different servers.

A framework is proposed to support feature identification and classification, enabling a structured approach to message modelling. The framework serves as a methodological guide to:

1. Identify data relevant to understanding queuing systems. Features under consideration include file sizes, source file sizes, mode, reference IDs, map names, service times, and interarrival times. Categorical data from ticketing systems is also evaluated to identify queuing inefficiencies.
2. Identify attributes suitable for data partitioning, either individually or in combination, regardless of direct impact on system performance.
3. Apply partitioning based on ranges of data values (e.g., splitting the data into a head/tail, where application-specific values might be confined to a particular region).
4. Fit the dataset to a range of parametric and non-parametric distributions using multiple transformation strategies.
5. In cases where model fitting is unsuccessful, use partitioning to explore data characteristics or uncover underlying correlations that complicate parametric modelling.

The proposed framework represents a methodological contribution of this thesis. It provides a structured process for identifying message features, partitioning heterogeneous datasets, and selecting appropriate modelling strategies under challenging conditions such as burstiness, heavy-tailed behaviour, interdependence, and quantisation noise.

The framework accepts raw EDI log data as input, applies feature identification and classification procedures, partitions the data according to operational and statistical characteristics, and produces fitted parametric or non-parametric models suitable for queue performance analysis and simulation.

4.3.2.1 Feature Identification and Selection

Two primary methods are employed to identify candidate features: Extracting key terms from log messages and traversing the structure of XML schemas that define EDI message definitions.

4.3.2.2 Feature Classification

Due to the complexity of the message metadata, a classification model is based on a layered system of attributes arranged in a logical hierarchy, also known as a taxonomy. The taxonomy provides a framework for assessing how attribute layers align with the modelling objectives. The classification structure supports the following tasks:

1. Evaluate the influence of message attributes on service times and inter-arrival times within the queuing system.
2. Assess the role of specific attributes in establishing message dependence or interdependence.
3. Determine whether layering attributes reveals or reduces the correlation.
4. Identify appropriate fitting strategies for heavy-tailed distributions using parametric and non-parametric models.

The classification model also provides guidance for dataset segmentation to expose a correlation or inter-message dependencies. Section 4.3.3 details the modelling techniques used.

4.3.3 Parametric Modelling

Using selected features and the classification model, modelling is performed on service and interarrival times, with particular emphasis on service times. Various combinations from the classification model are evaluated, and the data will be fitted to a range of continuous parametric distributions. Table 4.3 lists the distributions considered for modelling. Previous studies have demonstrated the applicability of these distributions across various domains.

Table 4.3: Parametric Distributions.

Beta	Burr	Cauchy	Expon-ential	Gamma	Inverse Burr	Log	Log–logistic	Logistic	Normal	Pareto	Uniform	Weibull

To improve distributional fit, Tukey's ladder of power transformations will be applied. Table 4.4 lists the transformations considered.

Table 4.4: Data Transformations.

Transformation-Log	Transformation-Sqrt	Transformation-Exp	Transformation-Cube
Log()	Sqrt()	Exp()	Cube()
Log(Log)		Sqrt(Exp)	
Sqrt(Log)			

Box–Cox was not considered because the aim is to compare fixed, non-parametric transformations that are widely used and easily interpretable (e.g., log, square root, cube). In contrast, Box–Cox represents a family of power transformations in which the parameter λ must be estimated from the data [147]. Since the focus of this study is on standard transformations rather than optimisation of transformation parameters, Box–Cox was excluded.

Transforming the data serves to reduce dispersion and skewness, which in turn can improve the fit to parametric distributions. Such adjustments may also lessen heavy-tailed behaviour and reduce excess kurtosis, thereby making the distribution more suitable for modelling. As shown in Table 4.4, multiple combined logarithmic and square root transformations are considered. During log transformation, a constant is added to the data to avoid negative values that may arise from taking the log of small values.

AD tests will be applied to each parametric distribution to assess their suitability, with particular focusing on tail sensitivity. If the data fails to fit any parametric distribution, a non-parametric approach is considered, as described in Section 4.3.4.

While partitioning can substantially improve modelling accuracy and GoF performance, excessive segmentation introduces a risk of overfitting, where models become highly specialised to specific subsets of the observed data and lose generalisability. Increasing partition granularity may improve local distributional fit while simultaneously reducing the simplicity and interpretability of the overall modelling framework. The framework therefore attempts to balance fitting accuracy against model complexity and operational relevance.

These modelling decisions also involve broader methodological trade-offs. Increasing partitioning complexity and reconstruction sophistication may improve

local fitting accuracy and reduce quantisation artefacts, but can also reduce interpretability and generalisability across datasets. Simpler adjustment strategies are easier to implement and explain operationally, whereas more complex reconstruction approaches may better preserve the underlying continuous structure at the cost of increased methodological complexity and reduced transparency.

4.3.4 Non-Parametric Modelling

KDE is employed as a non-parametric modelling technique, particularly suited to datasets with heavy-tailed behaviour. Bandwidth selection is carried out using several methods, including Silverman's rule of thumb, the Sheather–Jones method, biased and unbiased cross-validation, and the direct plug-in approach.

4.3.5 Message Interdependence

Interdependence refers to a mutual relationship between variables, where the state or behaviour of one variable can be influenced by the other. Identifying such relationships using the classification model is essential for accurate queue analysis. From a queuing perspective, understanding the order of message arrivals and determining whether specific messages depend on others could be critical for modelling system behaviour.

In some instances, a parent–child relationship may exist between messages, where the processing of a child message depends on the arrival or completion of its parent. Identifying such dependencies is essential for stress testing, as the timing and sequence of related messages can significantly affect service times. In simulation testing, it is important to define how parent–child relationships are handled and to select an appropriate queuing model, such as the G/G/1 model or a more complicated queuing network. The classification model is used to evaluate the presence of these relationships within the dataset.

Parsed and classified data will be analysed to detect queuing inefficiencies within the application. Section 4.3.6 presents the methodology used to investigate these inefficiencies.

4.3.6 Queuing Problems

Identifying inefficiencies in queuing systems is essential for accurate analysis and optimisation. A production-level support ticketing database will be analysed

to uncover system-level challenges. Many of these tickets have been pre-classified by the DevOps team to help identify recurring failure patterns and prioritise areas for improvement. Message re-processing can degrade queuing system performance. It typically occurs due to invalid content or formatting errors. To detect re-processing, unique message identifiers will be analysed to track repeated occurrences within the system. Identifying instances of EDI malformation and assessing whether specific EDI transformations are more error-prone than others are the aim of this study, which is based on attributes extracted from individual messages.

4.3.7 Quantisation Noise

The log file records show timestamps with millisecond precision (i.e., three decimal places). The potential influence of rounding or truncation on message ordering and data modelling is examined. For instance, if a message is logged as arriving at 08:00:10.435, it is assumed to be the earliest arrival at that point in the system. Any subsequent messages with later timestamps are treated as arriving sequentially. However, if the recorded timestamp is truncated, e.g., the actual arrival time was 08:00:10.435010, such precision loss could affect the true message order. In cases where all messages are contained within a single log file, sequential ordering may still be inferred. Nevertheless, timestamp truncation can alter calculated durations and impact the accuracy of downstream analysis.

Modelling challenges can occur when events appear to occur simultaneously, leading continuous-time distributions to resemble discrete ones. The effect may be attributed to quantisation noise, which arises when a continuous signal is represented using discrete values. To assess the presence of quantisation noise, the dataset is analysed using various modelling techniques derived from the classification model. A later chapter of this thesis returns to quantisation.

4.4 Results

4.4.1 Framework

For EDI message modelling, developing a structured framework is essential for addressing specific business objectives. The initial stage of this framework involves locating relevant features for analysis. The framework is exposed using

UML-type diagrams, so that the flow of the partitioning for the modelling framework can be easily understood.

4.4.1.1 Feature: Identification-Selection

The log files contained a combination of structured and unstructured text, including XML elements embedded within EDI messages. Attribute analysis identified potential features relevant to message modelling, and a hierarchical structure emerged, which was used to develop a taxonomy that categorises attributes by their source within the message. Figure 4.3 presents the identified attributes and their hierarchical organisation based on the log file data.

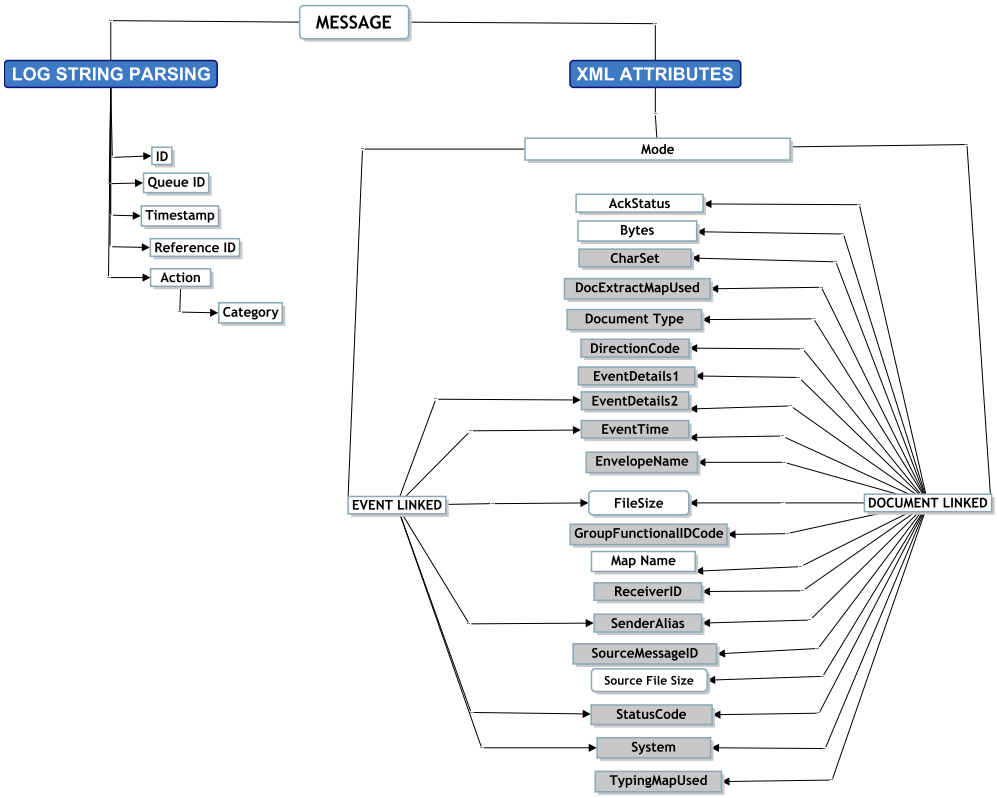


Figure 4.3: Message Attribute: Taxonomy.

Two entities, highlighted in blue, indicate the sources within the logs where attributes were extracted. The "Log String Parsing" entity denotes where keywords were derived from structured text, while the "XML Attributes" entity corresponds to attributes identified within the XML documents.

Within the "Log String Parsing" entity, only the "Category" attribute exhibits a hierarchical dependency on the "Action" attribute. Analysis of the "XML

Attributes" revealed that each message contains a "Mode" attribute, under which "EventLinked" and "DocumentLinked" function as child attributes. Inconsistencies were observed between these two child attributes. As shown in Figure 4.3, for example, the "Ack Status" attribute is associated exclusively with "DocumentLinked" and not "EventLinked".

Based on a detailed review of attribute content, certain attributes were deemed less informative for analysis. These attributes were excluded from further consideration and are shaded in grey. A total of thirteen attributes were retained. Table 4.5 provides a data dictionary describing the information contained in each selected attribute from the taxonomy presented in Figure 4.3. Several attributes were set with a "Status" of "Retained" in the table, due to their relevance in characterising message behaviour. The status corresponds to the shaded and unshaded entities in Figure 4.3.

Table 4.5: Message Attribute: Data Dictionary.

#	Name	Description	Status	Status Reason
1	ID	Combined Company and Message ID. Used as a unique identifier.	Retained	Critical for identifying each message uniquely.
2	Queue ID	Destination queue name.	Retained	Helps trace message routing within the system.
3	Timestamp	Time the message entered the system.	Retained	Necessary for computing interarrival and service times.
4	Reference ID	Identifies the message sender.	Retained	More reliable than "Sender Alias"; frequently populated.
5	Action	Operation performed by the translation service.	Retained	Influences processing logic; relevant to duration.
6	Category	Translation type applied to the message.	Retained	Defines how the message is transformed.
7	Mode	Message type.	Retained	Serves as a root node for XML hierarchy.
8	EventLinked	Short-running, event-based processing type.	Retained	Relevant to performance classification.
9	DocumentLinked	Indicates long-running, document-type messages.	Retained	Impacts processing duration and message flow.
10	Ack Status	Electronic receipt status.	Retained	Indicates message delivery status.
11	Bytes	Message size in bytes. May be zero for "EventLinked".	Retained	Useful for modelling size-related effects.
12	File Size	Post-split size of message file.	Retained	May affect duration and resource usage.
13	Map Name	Transformation map(s) associated with message.	Retained	Impacts processing complexity and time.
–	Sender Alias	Alternate sender identifier.	Excluded	Frequently null; duplicative of "Reference ID".
–	Source File Size	Original file size before splitting.	Excluded	Optional; only captured for select protocols.
–	Direction	Message flow direction (inbound/outbound).	Excluded	No observed effect on performance metrics.
–	EventDetails1 / EventDetails2	Additional message identifiers.	Excluded	Redundant; "ID" attribute provided needed information.
–	System	System name associated with message.	Excluded	Static value across all records; not discriminative.
–	Envelope Name	Envelope wrapper associated with message.	Excluded	One-to-many mapping caused ambiguity.

With the relevant features identified, a classification model is constructed to support and potentially enhance the modelling capabilities outlined in the next section.

4.4.1.2 Feature Classification

Preliminary analysis revealed notable skewness and high kurtosis, indicating heavy-tailed behaviour in the dataset and necessitating the use of multiple techniques to achieve suitable parametric fits. A classification model was built using features derived from exploratory analysis and technical consultation with the DevOps team. Figure 4.4 displays a tree-based representation of the

newly defined classification model.

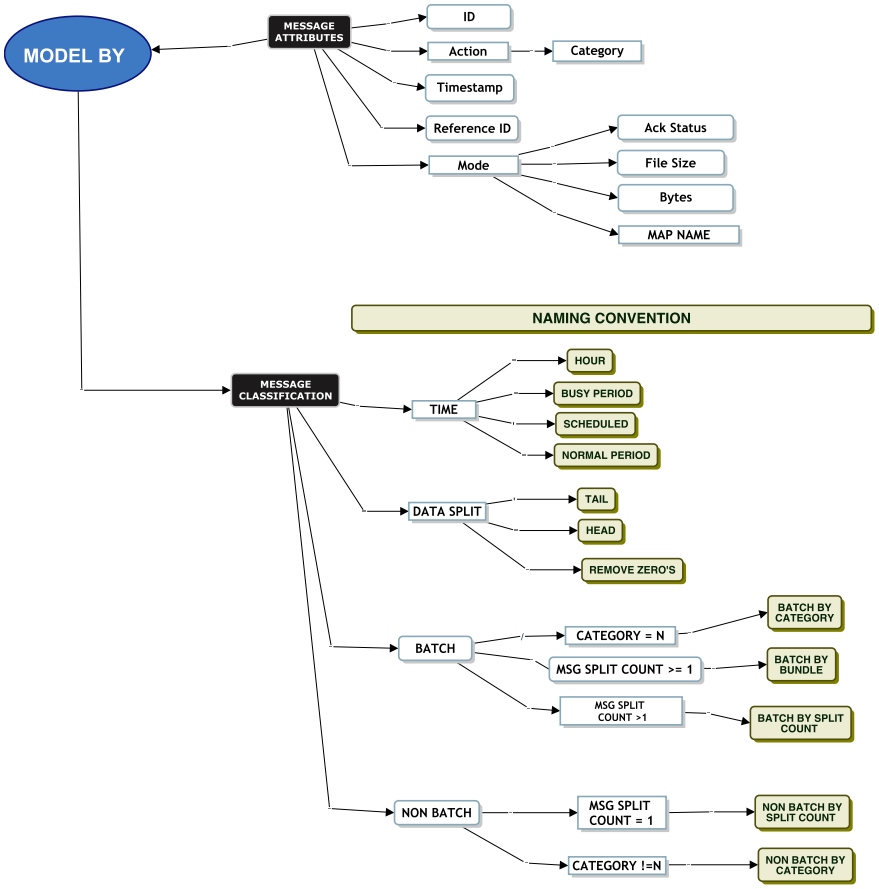


Figure 4.4: Message Classification.

The classification model serves as a guide for distribution fitting, with particular emphasis on meaningful data partitioning strategies. The resulting classifications, combined with Tukey's ladder of powers transformations, support the modelling process. Table 4.6 gives a brief description of each classification, which corresponds to the light yellow coloured entities in Figure 4.4.

Table 4.6: Message Classification Model: Description of Message Filtering and Partitioning Criteria.

Time	
Hour	Model by each hour in the dataset.
Busy Period	The number of messages queued was always greater than zero and growing.
Scheduled	Time ranges with consistently higher message volumes, such as around midnight, or at 15-, 30-, and 45-minute marks.
Normal Period	The opposite of the busy period.
Data Partition	
Tail	Partitioned data retaining only the tail (values > than some threshold).
Head	Partitioned data retaining only the head (values < than some threshold).
Remove Zeros	Remove all zero values from the dataset.
Batch	
Batch By Category	Filters where "Category" includes "Flat Translation" or "Batch XML Translation".
Batch By Bundle	Counts how often a message arrives. If the count is 1, the message is kept. If greater than 1, only the first and last messages are retained, ignoring all others in between.
Batch By Split Count	Counts how often a message arrives and how many XML documents it contains. Kept only if the count of XML documents is greater than 1.
Non-Batch	
Non-Batch By Split Count	Similar to above, but keeps messages with exactly 1 XML document.
Non-Batch By Category	Filters where "Category" excludes "Flat Translation" and "Batch XML Translation".

Parametric modelling is applied to the dataset based on the selected features and the filtering criteria defined by the classification model.

4.4.2 Parametric Modelling

Figure 4.5 presents histograms of service and interarrival times during a normal operational period, illustrating the effect of Tukey's ladder of power transformations on the shape of the data.

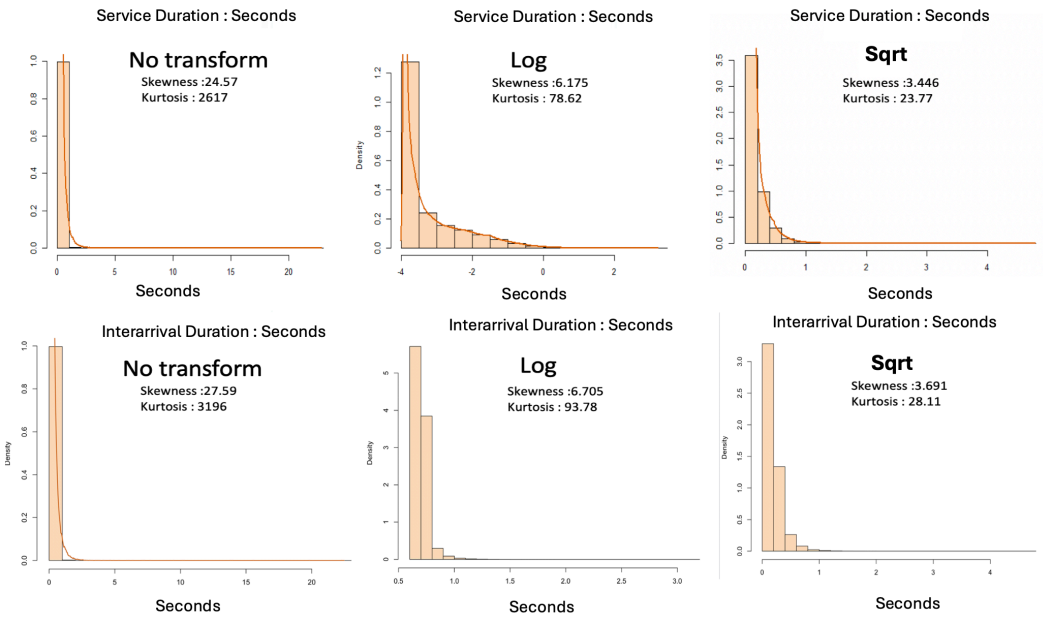


Figure 4.5: Histogram: Normal Period, Service and Interarrival Times; Different transformation strategies.

Two transformation types were applied, logarithmic and square root, with the untransformed data retained as a baseline for comparison. The untransformed data are right-skewed and leptokurtic, with service time skewness and kurtosis measured at 24.57 and 2617, respectively. Similar characteristics are observed in the interarrival time histograms.

The logarithmic transformation substantially reduces skewness and kurtosis (e.g., service time skewness reduces to 6.18); in contrast, the square root transformation yields more modest improvements. Nonetheless, all transformed histograms remain non-symmetric, indicating persistent heavy-tailed behaviour and a high concentration of values near zero.

These data characteristics pose challenges for parametric modelling. The newly developed classification model aims to improve fitting problems using the partitioning from the model and transformations defined earlier in Table 4.4.

4.4.2.1 Model: File Size

Understanding message file size is essential for system testing, as it indicates whether a message requires splitting for queue ingestion and informs estimates of required disk space. File size may also influence service times. Based on feedback from the DevOps team, one inbound connector does not support files larger than 100 MB, and most observed file sizes are approximately 20 kilobytes.

The classification model shown in Figure 4.6 was applied to segment the dataset by file size.

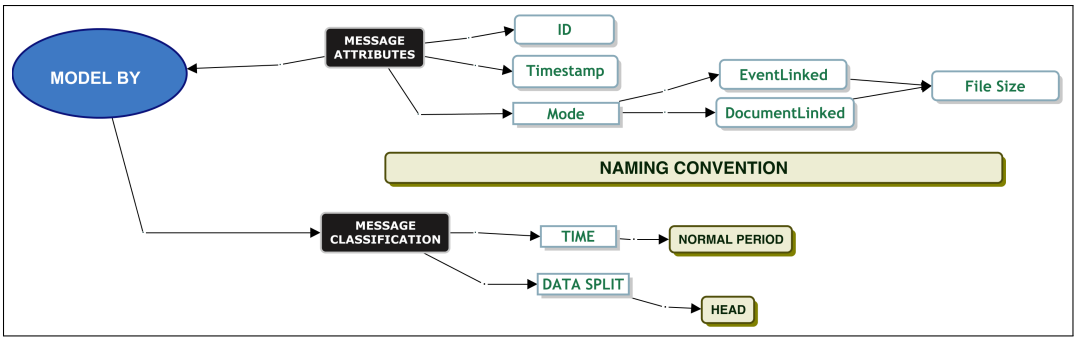


Figure 4.6: Classification Model: File Size.

Data from the "Normal Period" was used, and a partition boundary of $10,000,000$ bytes (≈ 0.1 GiB $\equiv 102$ MiB) was used to distinguish between the head and the tail of the file size. A random sample was then drawn from the dataset due to the volume of messages. Table 4.7 reports the count of observations in each partition prior to sampling.

Table 4.7: File Size: Count.

Status	Count
Head	984,070
Tail	109

Figure 4.7 presents a set of histograms showing the distribution of file sizes before and after partitioning. The top-left chart shows the un-partitioned dataset, whilst the top-right chart shows the result of a random sample from the head of the distribution.

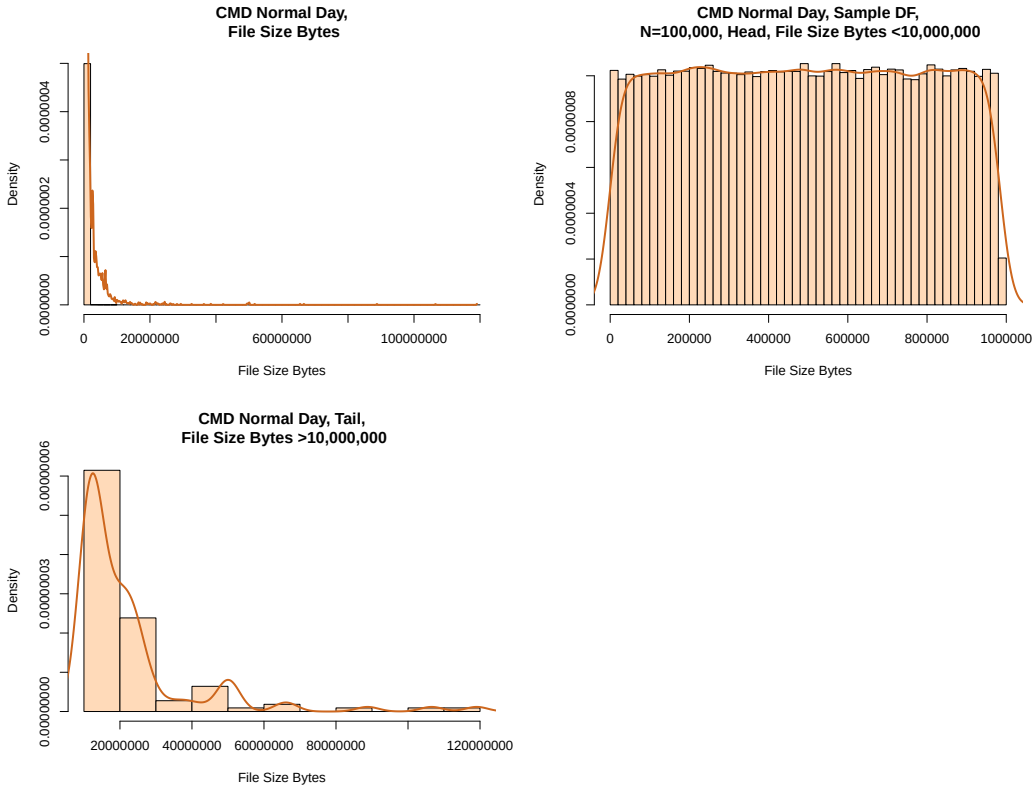


Figure 4.7: Histogram: File Size.

The shape observed in the head suggests alignment with a uniform distribution. The bottom-left chart shows the histogram of the data's tail. A chi-square GoF test was applied to the head with the results shown in Table 4.8. At a significance level of $\alpha < 0.05$, the test results indicate that a uniform distribution provides a reasonable fit for the head of the random sample.

Table 4.8: File Size: Head, Chi-square Uniform Distribution Test.

Test Statistic	**DF**	***p*-value**
18	19	0.05

4.4.2.2 Model: Batch By Category

Partitioning by category was domain-driven, reflecting structural and behavioural differences between EDI transaction classes.

When simulating service and interarrival times, it is essential to determine whether messages are ingested individually or in batches, as batch arrivals

represent a form of interdependence that significantly affects queue modelling. Figure 4.8 shows the classification applied for "Batch by Category".

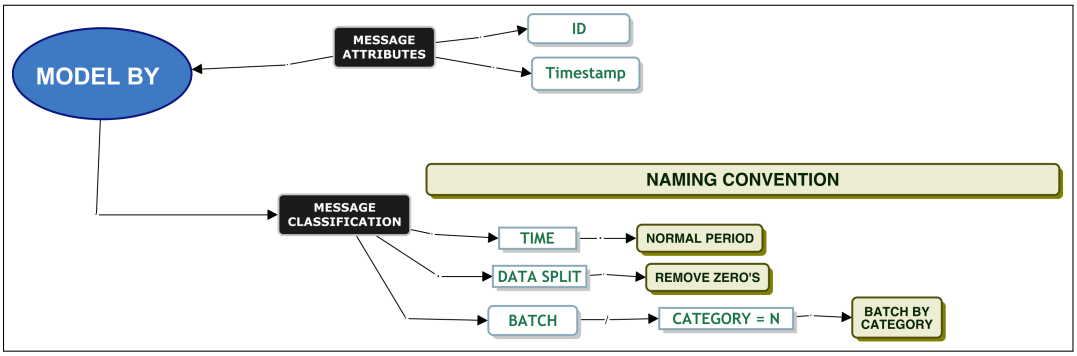


Figure 4.8: Classification Model: Batch by Category.

Prior to distribution fitting, measures of dispersion were assessed with Table 4.9 showing the results.

Table 4.9: Service Times: Batch by Category Statistics.

Service Duration: Second	Total	Min	Mean	Max	95th Percentile	99th Percentile	Var	Skewness	Kurtosis
Batch	126,742	0	0.03	14.04	0.16	0.44	0.01	26.17	2716
Batch Zero Removed	65,878	0.001	0.05	14.04	0.26	0.61	0.01	20.84	1658

Approximately 50% of batch messages were excluded upon removing entries with zero-duration service times. The remaining data exhibited heavy-tailed characteristics, with skewness exceeding 20 and kurtosis exceeding 1600. A 0.10-second difference was observed in the $95th$ percentile between all batch messages and batch messages with zero duration excluded. Messages with non-zero service times generally required less than 0.26 seconds for processing.

According to the DevOps team, messages classified under the "Flat Translation" or "Batch Translation" categories, based on the "Category" attribute, should be treated as batch messages and require file segmentation prior to ingestion. All other messages are classified as non-batch. Inspection of the dataset revealed that messages labelled for "Flat" or "Batch translation" appeared as both single messages and batch-type messages, contrary to expectations. Service times for these messages were subsequently modelled.

The classification model from Figure 4.8 was applied without partitioning,

i.e. no head or tail segments. Table 4.10 presents the two distributions with the lowest AD test statistic results. The closest alignment is to a log-normal distribution; however, the fit remains insufficient, failing to meet the AD test statistic (AD = 2.49) criteria previously noted in Table 2.4.

Table 4.10: Service Times: Batch by Category, AD Results.

Test	Transformation Sqrt	Transformation Cube Root
Burr	362	362
Log-normal	281	281

To further interpret the results, Figure 4.9 shows the results of fitting to a log-normal distribution with a square root transformation. The Q–Q plot indicates that the data deviates from the theoretical quantiles at the lower end, with noticeable divergence in the initial segment of the line. The P–P plot reveals that although the upper tail appears continuous, lower probability regions contain numerous discrete values, suggesting limited continuity in the lower tail of the distribution.

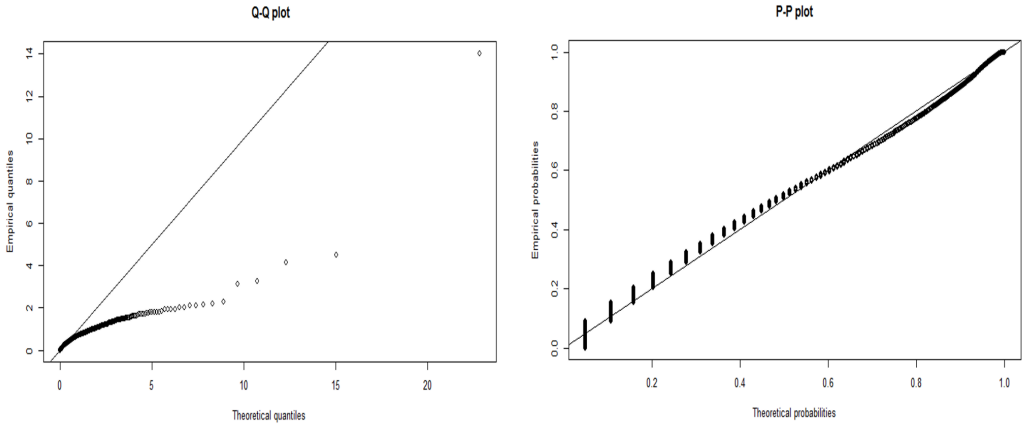


Figure 4.9: Log-normal Service Times: Batch by Category.

The data was further partitioned into heads and tails to separate the discrete values. Figure 4.10 shows the results of the lower tail from the P–P plot, which is the head of the data. From the figure, one can see four discrete Gaussian distributions. These four Gaussian distributions result from KDE applied to the discrete data and are not a feature of the data but a feature of the plotting of the quantised data. The data is expected to be continuous; however, the figure is not a true representation of the original data or the real-world system from which the data was derived.

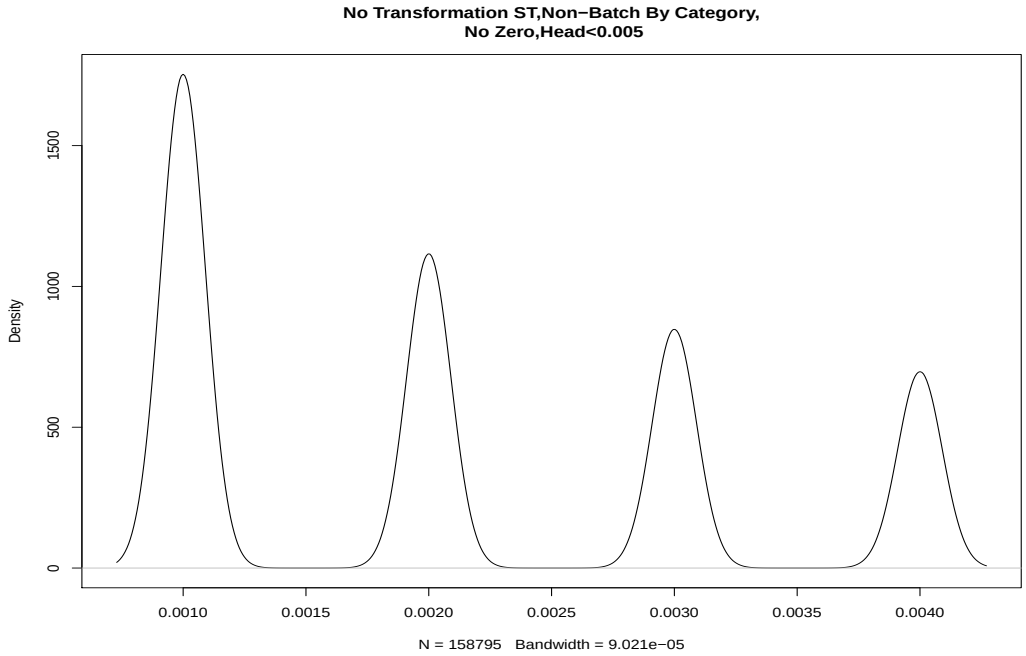


Figure 4.10: No Transformation - Service Times : Batch by Category, Head of Data.

Note. The Gaussian peaks arise from the KDE smoothing process and are not features of the underlying recorded data.

4.4.2.3 Model: Batch By Bundle

Within this section messages associated with a bundle of XML documents, referred to as "Batch By Bundle" are examined. The approach differs from classifications based on the "Type" attribute, which designates messages as batch or non-batch. Figure 4.11 shows the makeup of the classification "Batch by Bundle".

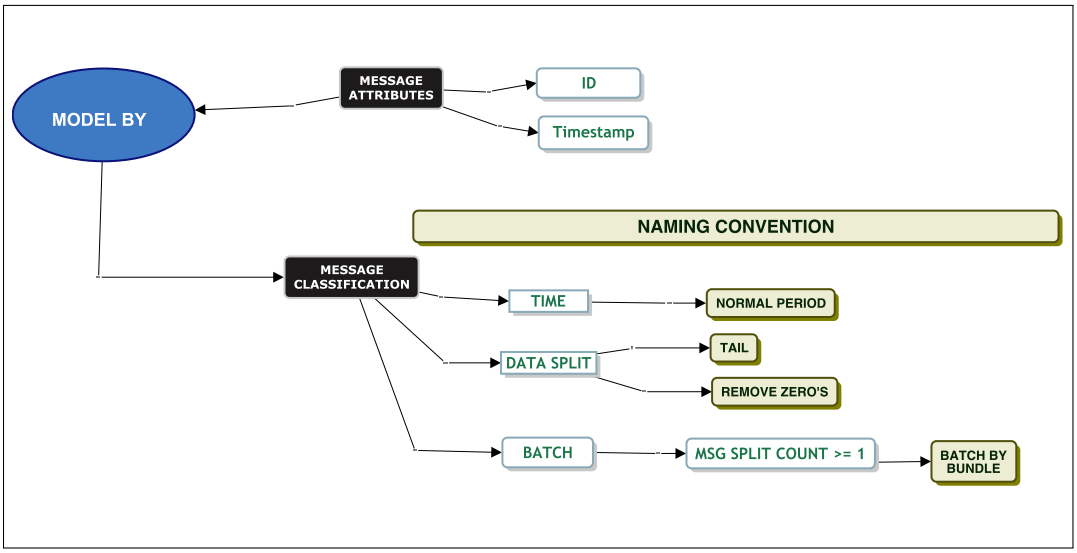


Figure 4.11: Classification Model: Batch by Bundle.

For messages containing a single XML document, corresponding service times are extracted for analysis. For messages with more than one associated XML document, only the first and last message in the bundle are retained. In contrast intermediate messages are excluded due to having zero-duration service times.

Using the classification structure in Figure 4.11, data from the "Normal period", focusing on the tail of the distribution and excluding zero-duration messages, was analysed. The results of the fitting process closely approximated a log-normal distribution when visually inspected; however, it did not satisfy the AD GoF test.

The data was then partitioned into head and tail segments using a 1-second threshold. Results from fitting a log-normal distribution to the tail are illustrated in the left chart of Figure 4.12.

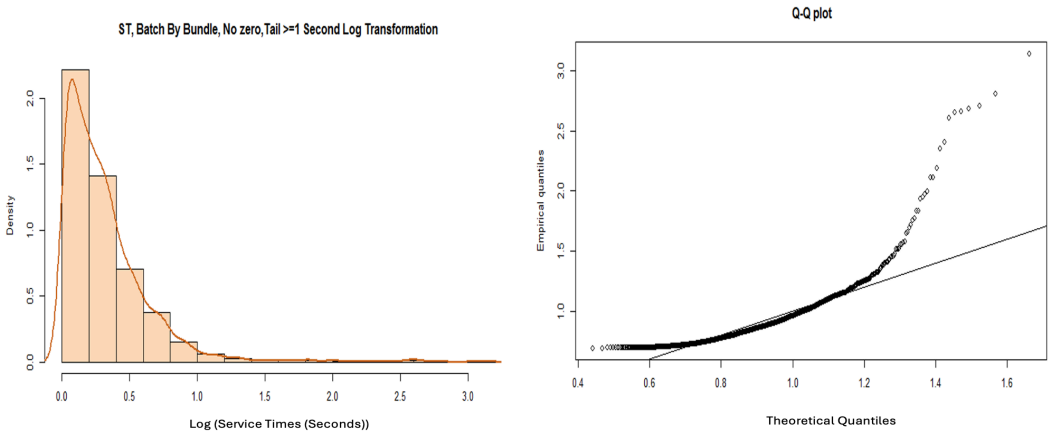


Figure 4.12: Batch by Bundle, Service Times, Tail.

The histogram suggests a shape resembling a log-normal distribution, but the model fails the AD GoF test. The right chart shows the corresponding Q–Q plot, which shows that lower quantiles align more closely with the theoretical line than upper quantiles, although systematic deviations remain evident. These deviations suggest that the data may be multi-modal, and that introducing a split at the first point of departure from the theoretical line, followed by a separate fit to the upper segment, may provide a more accurate representation of the distribution.

Table 4.11 presents the lowest AD GoF test result. A constant offset of 1 was added to the tail of the service time data, followed by a logarithmic transformation to enhance model fit.

Table 4.11: Tail: AD GoF Test-Batch by Bundle.

Tail Test	**AD**	***p*-value**	**Transformation**	**Constant**
Log-normal	61	0.0000003	Log	1

To model the head of the data, Figure 4.13 presents a histogram of service times overlaid with a probability density estimate.

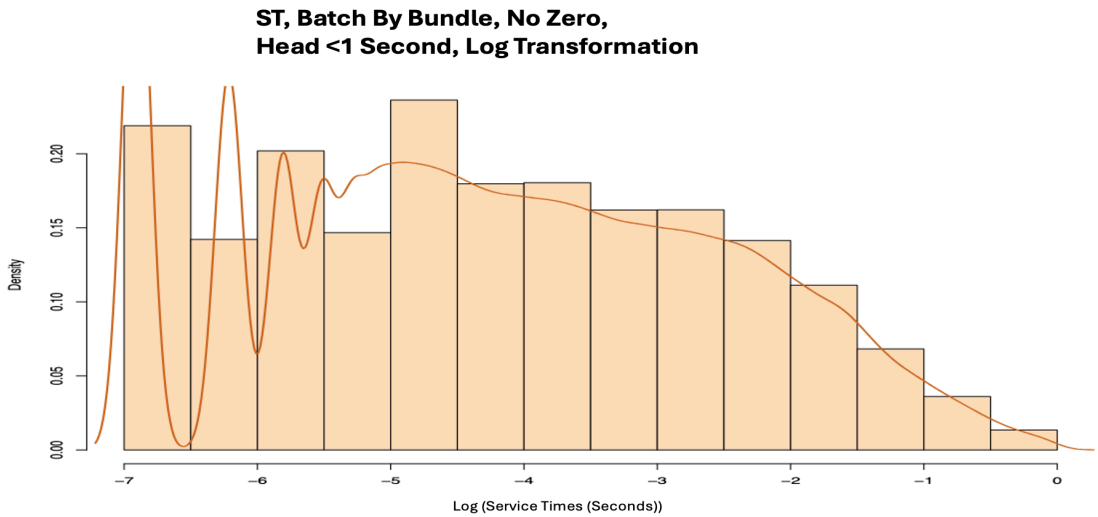


Figure 4.13: Batch by Bundle, Head of Data < 1 Second, log(Service Times).

Several peaks observed at the lower end of the distribution again suggest the presence of quantisation effects. The behaviour indicates that the data is unlikely to conform to a parametric distribution. Further partitioning, application of KDE, or adjustments to account for quantisation may improve modelling outcomes.

4.4.2.4 Model: Batch By Split Count

Partitioning by split count was motivated by the hypothesis that fragmented message bundles exhibit different service-time characteristics from single-message transactions.

Modelling is conducted using the "Batch By Split Count" classification, which captures cases where a single message is split into multiple XML documents before being queued. The classification is defined by an XML document count greater than one (>1) as shown in Figure 4.14.

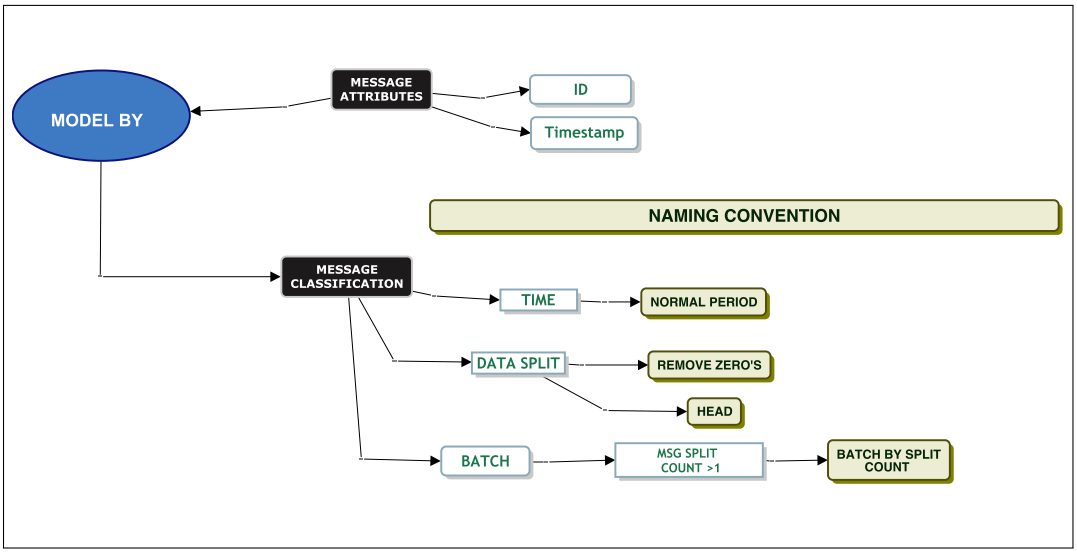


Figure 4.14: Batch: Split Count, Classification Model.

Messages are classified under the "Batch By Split Count" category, where a single message is divided into multiple smaller messages and sent to the queue as a batch. Parametric distributions were fitted to the service times of these batch messages. The analysis applies three partitioning strategies based on XML document count: messages with XML count > 1, XML count $= 2$, and XML count > 2.

No combination of transformation or partitioning techniques produced a satisfactory parametric fit for the service times of batch messages. Table 4.12 presents the two distributions with the lowest AD test statistic scores.

Table 4.12: Batch by Split Count, Service Times ≤ 1 Second, Filter Count > 2, Zero's Removed.

AD Test	No Transformation	Log-(n+1)	Sqrt(n)	Exp(n)	Exp-(log-(n+1))	Sqrt-(log-(n+1))	Log-(log-(n+1)+1)	Sqrt-(exp-(n))
Log-normal	**39**	51	**39**	1500	1300	51	63	1500
Burr	**41**	45	**41**	640	470	45	49	640

In the interests of space, only AD scores are shown. The majority of p-values are rounded to 0.00.

For the head of the service time data, with split count > 2, service time ≤ 1 second, no transformation applied, and zero-duration messages excluded, the log-normal and Burr distributions yielded AD test statistic scores of 39 and 41, respectively, indicating a close but inadequate fit.

Based on the observations in Table 4.12, and looking at the results of the applied model for a log-normal distribution in Figure 4.15, the data does not fit a parametric distribution for the head of the data.

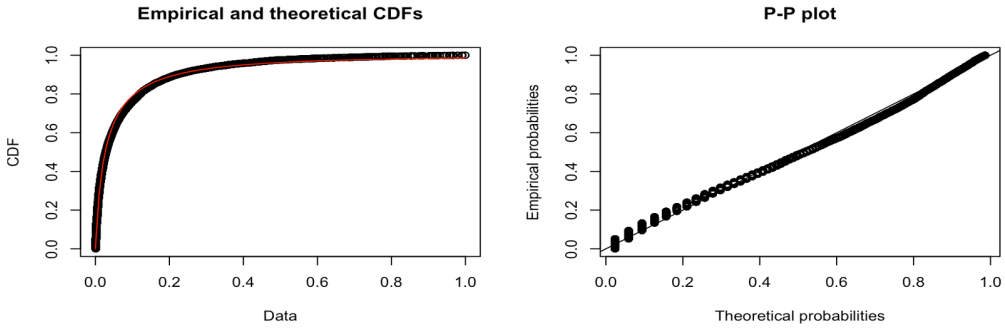


Figure 4.15: Batch by Split Count, log-normal, Fitting Model Results.

It seems that to model this dataset, it might be best to choose a non-parametric technique, such as KDE. The tail of the data also did not conform to any parametric distribution.

Modelling is now focused on the "Non-Batch By Split Count" classification.

4.4.2.5 Model: Non-Batch By Split Count

"Non-Batch By Split Count" refers to the subset of messages where exactly one XML document is associated with each message, complementing the "Batch By Split Count" category. Figure 4.16 shows the classification used.

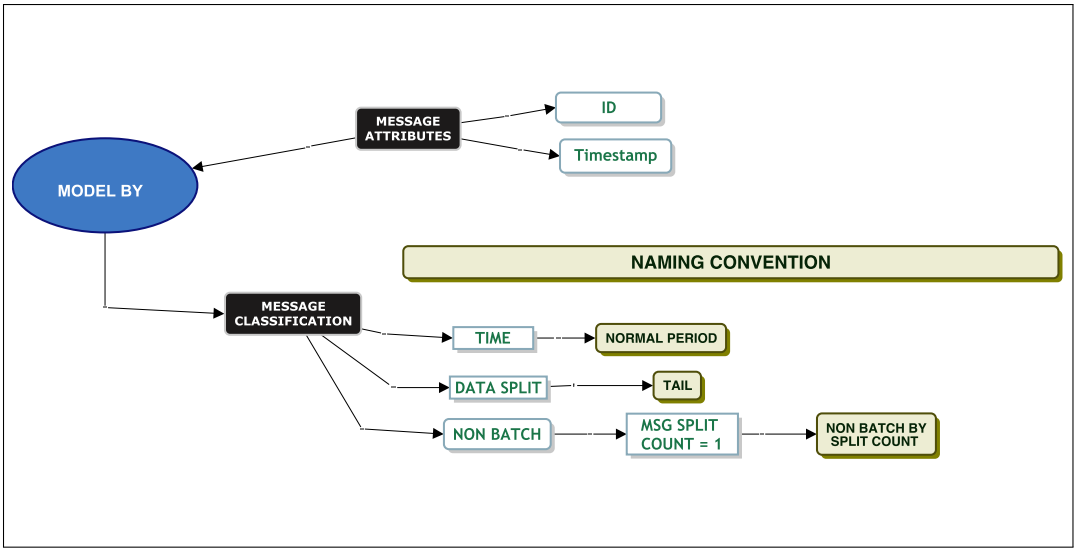


Figure 4.16: Non-Batch: Split Count Classification.

Service and interarrival times were modelled. The analysis identified a Burr distribution as the only suitable parametric fit for the tail of the service time data. The CDF plot in Figure 4.17 illustrates both the empirical distribution and the fitted Burr distribution. The P–P plot shows no significant deviation of observations from the diagonal line, indicating a close alignment between the model and the observed data.

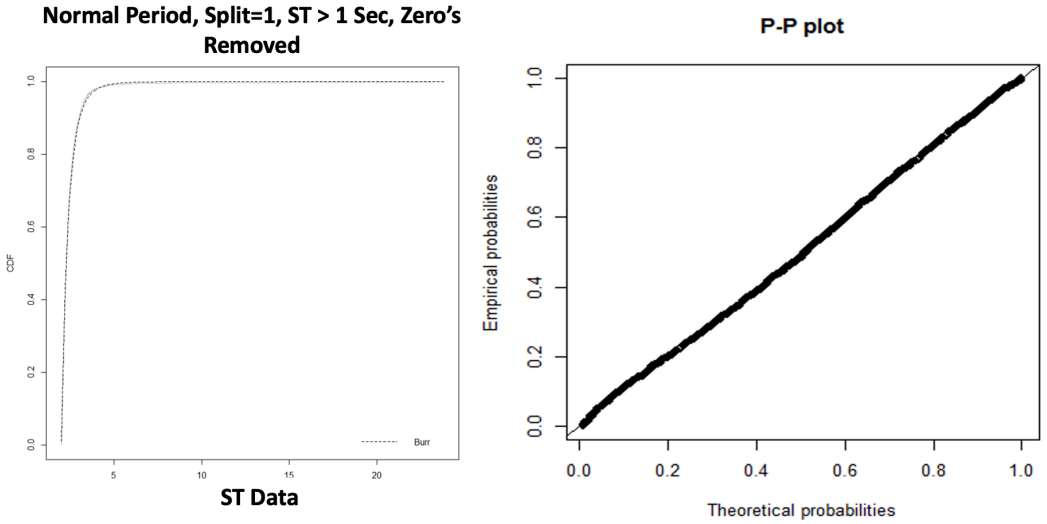


Figure 4.17: Normal Period, Split = "1", Burr Fitting, Service Times - > 1 Second.

The AD test results in Table 4.13 confirm that the tail of the service times fit a Burr distribution.

Table 4.13: AD Test: Normal Period, ST, Tail of Data.

AD Score	***p*-value**	**Test**
1.2	0.3	Pass

A constant of 1 was applied to the dataset before testing in order to avoid convergence errors. Since the Burr distribution is defined only for values greater than zero, the presence of zeros renders the likelihood function undefined and causes MLE to fail to converge.

For the head of the service time data (service times ≤ 1 second), none of the tested parametric distributions achieved a satisfactory fit, regardless of the transformation or grouping techniques applied. As shown in Table 4.14, all AD test scores exceed acceptable thresholds (an AD score ≤ 3.5 is typically required for a valid fit).

Table 4.14: ST < 1, Filter = 1 : AD Test.

AD Test	Untrans-formed Data	Log-(data+1)	Sqrt (data)	Exp(data)	Sqrt(log (data+1))	Log(log-(data+1)-+1)	Sqrt(exp(-data))
	AD Score						
Log-normal	4146	4279	4146	78917	4279	4427	78917
Log–logistic	1099486	153957	1703852	35541593	1778558	1198064	89237003
Gamma	681391	20247	7959	85258	6757	14646	81636
Weibull	410883	453321	615255	2912272	679932	492797	9870194
Exponential	131869	112365	9986	200079	10607	97906	216494
Cauchy	101117	101117	98590	42829	103782	96204	102433
Logistic	58318	54796	24413	63141	23026	51976	60585
Pareto	4300	4624	9986	200079	10607	4936	216494
Burr	4488	4682	4487	54435	4682	4903	54470
Inv Burr	5210	5651	5216	57655	5690	6089	57675

In the interests of space, only AD scores are shown. The majority p-values round to 0.00.

Interarrival times are partitioned into head and tail segments using a 1-second boundary. For the tail segment (greater than 1 second), results from the AD GoF tests in Table 4.15 indicate that the filtered data does not conform to any parametric distribution. Table 4.15 presents the closest fitting model among the tested distributions.

Table 4.15: IAT > 1 Second, Split Count = 1 : AD Test.

AD Test	**Untrans-formed Data**	**Log(-data+1)**	**Sqrt (data)**	**Exp-(data)**	**Sqrt(log-(data+1))**	**Log(log-(data+1) +1)**	**Sqrt(exp-(data))**
			AD Score				
Burr	**3.9**	6.6	**4**	Inf	6.6	8.5	Inf

In the interest of space, only AD scores are shown in the table. The majority of p-values round to 0.00.

The untransformed data and the square root transformation, highlighted in bold, demonstrate relatively close alignment with the Burr distribution, although they do not satisfy the AD test criteria. However, a correlation was identified in the head of the interarrival times.

4.4.2.6 Model: Non-Batch By Category

To emphasise the relevance of modelling using the "Non-Batch By Category" classification, Section 4.4.2.2 provides background context. Figure 4.18 shows

the classification model used for this testing strategy.

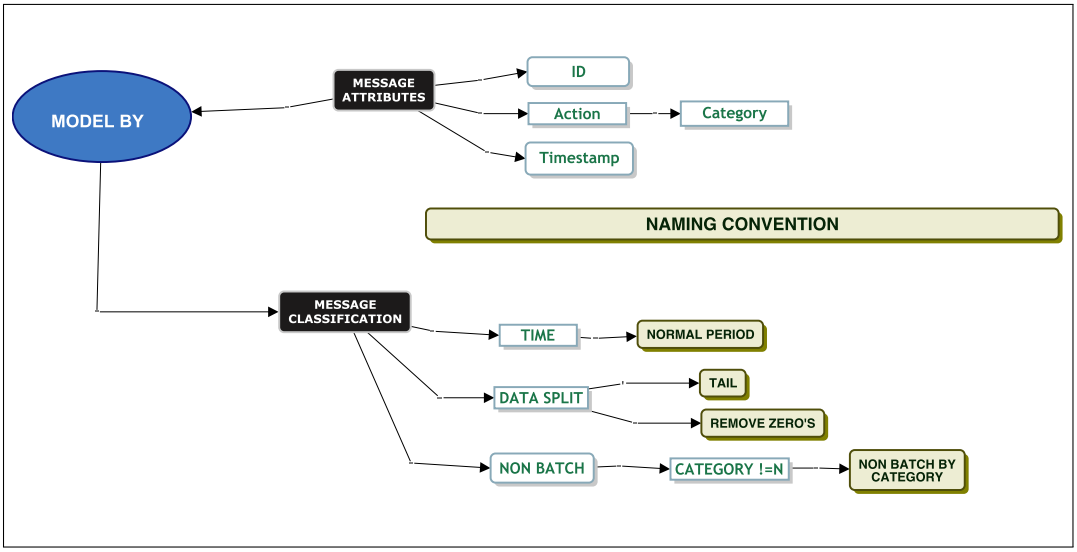


Figure 4.18: Non-Batch by Category: Classification Model.

Before parametric fitting, some prior analysis was conducted. As a first step, dispersion measures were evaluated. Table 4.16 shows the results.

Table 4.16: Non-Batch by Category: Measures of Dispersion.

Service Duration Seconds	Total	Min	Mean	Max	95th Percentile	99th Percentile	Median	Var	Skewness	Kurtosis
Non-Batch	857,437	0	0.04	22.81	0.23	0.60	0.004	0.01	24.24	2550
Non-Batch Zero Removed	570,020	0.001	0.06	22.81	0.31	0.73	0.01	0.02	21.06	1880

Approximately one-third of service time entries are excluded when zero-duration messages are removed. A small difference is observed in the 95th percentile service time between the full dataset and the dataset with zero values excluded. The distribution is highly skewed, with skewness exceeding 20. After removing zero-duration messages, the minimum recorded service time is 0.001 seconds.

For parametric testing, this classification test uses normal period data, focusing on the tails, and excludes messages with zero-second durations. Table 4.17 presents the best performing model based on AD test scores.

Table 4.17: Non-Batch by Category, Tail of ST, AD Results.

Test	No Transform	Log Transform +1	Square Root	Cube Root
log-normal	3942	3966	3942	3942

Despite this, the AD values remain high, indicating that none of the tested distributions adequately fit the data, and a suitable parametric model has not been identified.

When modelling the head of the dataset, evidence of discrete values indicative of quantisation noise was observed. A detailed discussion of the implications of quantisation effects was provided in Section 4.5.6. Using the classification model, the analysis proceeds to a non-parametric modelling approach as an alternative.

4.4.3 Non-Parametric Modelling

When parametric distributions failed to provide a suitable fit, even after applying partitioning via the classification model, KDE was employed. KDE usually provides a good fit to the observed tail data, based on the different algorithms and kernels drawn to support each point's area under the curve. For example, Figure 4.19 illustrates KDE applied to the tail of service time data, partitioned by hour.

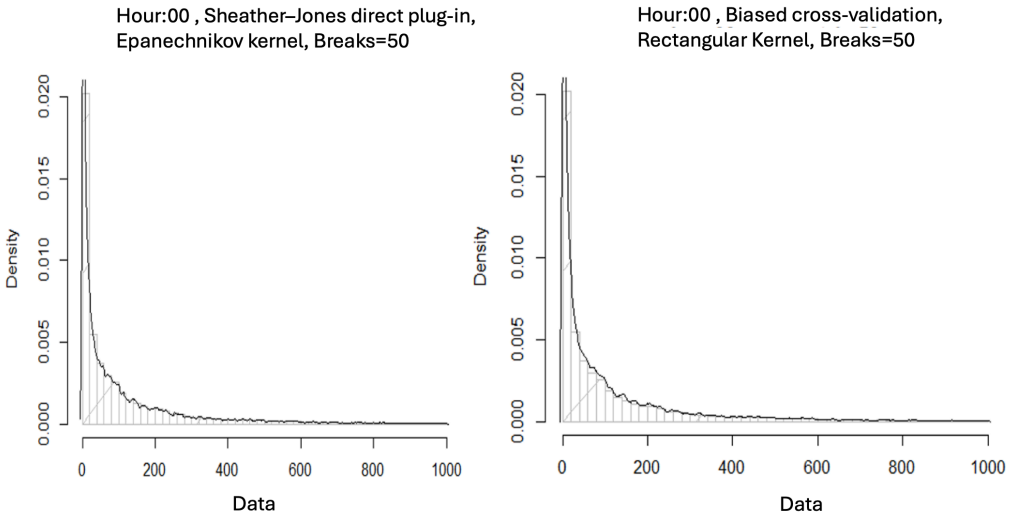


Figure 4.19: Tail of Data, by Hour, ST.

The left-hand histogram uses the Sheather–Jones "plug-in" estimator (dpi)

with an Epanechnikov kernel. In contrast, the right-hand histogram applies an unbiased cross-validation bandwidth selector with a rectangular kernel. However, for the head of the data, KDE did not provide an appropriate fit due to quantisation.

4.4.4 Message Interdependence

Previous analysis of interarrival times, as reported in Chapter 3.1, revealed evidence of a correlation, which suggests that caution is required when applying G/G/1 queuing models, where independence of arrivals is typically assumed. Nevertheless, for the purposes of this research, only single-queue models were considered.

Message interdependence is a critical factor in simulating queuing systems. While modelling the behaviour of individual messages is relatively straightforward, complexity increases when parent–child relationships exist, particularly when these relationships follow a one-to-many structure. In such cases, the service time of a batch message may be influenced by the arrival and processing of all associated child messages. To investigate the presence of interdependent messages, the classification model was applied. Using the structures shown in Figure 4.20, parent–child dependencies were inferred through the Batch By Split Count" and Non-Batch By Split Count" classification models, indicating the presence of message-level dependencies in the dataset.

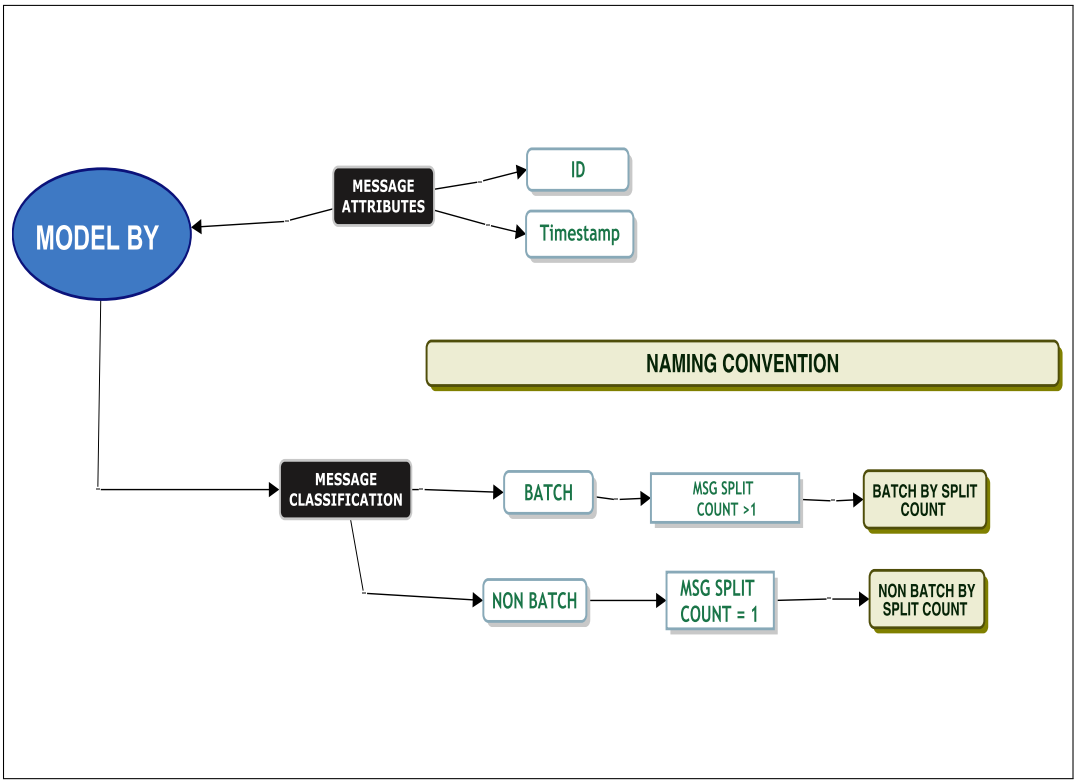


Figure 4.20: Interdependence: By Classification.

Figure 4.21 shows a section of a log file that was examined to provide contextual evidence supporting the identified interdependence relationships. The following section describes what can be deduced from this log message.

```
1
2  [2020-11-30 03:05:22.178] : id1;id2: inbound message start
3  [2020-11-30 03:05:22.178] : id1;id2: setting the state to ..
4  [2020-11-30 03:05:22.186] : id1;id2: REFERENCE_ID is ..
5  [2020-11-30 03:05:22.186] : id1;id2: Action: Translation Category: Flat Translation
6  [2020-11-30 03:05:22.282] : id1;id2: <XML>
7  [2020-11-30 03:05:22.282] : id1;id2: finished processing
8
9  [2020-11-30 03:05:22.178] : id2;id5: inbound message start
10 [2020-11-30 03:05:22.178] : id2;id5: setting the state to ..
11 [2020-11-30 03:05:22.186] : id2;id5: REFERENCE_ID is ..
12 [2020-11-30 03:05:22.186] : id2;id5: Action: Translation Category: Flat Translation
13 [2020-11-30 03:05:22.282] : id2;id5: <XML>
14
15 [2020-11-30 03:05:22.178] : id1;id2: inbound message start
16 [2020-11-30 03:05:22.178] : id1;id2: setting the state to ..
17 [2020-11-30 03:05:22.186] : id1;id2: REFERENCE_ID is ..
18 [2020-11-30 03:05:22.186] : id1;id2: Action: Translation Category: Flat Translation
19 [2020-11-30 03:05:22.282] : id1;id2: <XML>
20 [2020-11-30 03:05:22.282] : id1;id2: finished processing
21
22 [2020-11-30 03:05:22.178] : id4;id3: inbound message start
23 [2020-11-30 03:05:22.178] : id4;id3: setting the state to ..
24 [2020-11-30 03:05:22.186] : id4;id3: REFERENCE_ID is ..
25 [2020-11-30 03:05:22.186] : id4;id3: Action: Translation Category: Flat Translation
26 [2020-11-30 03:05:22.282] : id4;id3: <XML>
27 [2020-11-30 03:05:22.282] : id4;id3: <XML>
```

Figure 4.21: Log Message.

At a high-level, a message first arrives on line 2. The message is set to an associated state defined by a map associated with the message as per line 3, and the customer name is associated in line 4 using a Reference ID. The message is then sent for translation in line 5, and the XML for the message is produced in line 6. The message is then completely processed in line 7. A subset of this message is then processed again, starting at line 15 and finishes at line 20, meaning that this message is split into two messages and is fully complete at line 20.

To investigate inter-message relationships, the message ID was parsed by splitting the identifier into two components: a Company ID (id1) and a Message ID (id2). As illustrated in the hypothetical example in Figure 4.21, the second component (id2) was used to track message instances. For each occurrence of id2, the number of <XML> tags was counted. For example, the message initially enters the system at line 2, with an <XML> tag observed at line 6, representing one XML document. The same message reappears at line 15, followed by another <XML> tag at line 19, yielding a total of two XML documents associated with id2. The process was repeated across all log entries, and the total number of XML documents was aggregated for each message.

It is important to note that if multiple <XML> tags appear consecutively (e.g., on lines 26 and 27), only the first occurrence is counted. Although this behaviour was not observed in the current dataset, it is acknowledged as a possible scenario on other systems and is accounted for accordingly.

Using the identified method, it is possible to determine whether a single message arrival will result in an influx of subsequent messages into the queuing system.

The analysis also investigates whether a dependence exists between the service times of independent and non-independent messages. To examine this, messages were segmented based on whether their service time exceeded 1 or 2 seconds, as shown in Table 4.18. These thresholds were selected to represent the tail of the distribution, allowing for assessment of whether dependent messages are more likely to appear in the tail.

Table 4.18: ST Exceeds 1:2 Seconds.

Classification	ST $\leq$ 1 s	ST $>$ 1 s	ST $\leq$ 2 s	ST $>$ 2 s
Independent	126,585	157	126,731	11
Non-Independent	854,722	2,714	857,160	276

A Fisher's exact test was conducted to assess the independence between message types. As shown in Table 4.19, the test results indicate no statistically significant association ($p > 0.05$) between independent and non-independent messages where service times exceed 1 or 2 seconds.

Table 4.19: Dependence Check: ST Exceeds N Seconds.

Test	Odds Ratio	*p*-value
Exceeds 1 s	2.56	0.0000000
Exceeds 2 s	3.71	0.0000003

4.4.5 Queuing Problems

As an enterprise dataset was used to examine the range of issues documented in the queue ticketing system, Table 4.20 presents the categorisation of problems recorded during a defined operational period.

Table 4.20: Queuing system problems recorded in the ticketing system.

Type	Count	Percent
Unclassified	1900	27
Communication Channels	806	11
Installation	620	9
Security	619	9
Transport Layer Security	566	8
Queue Managers	477	7
Authorised Program Analysis Records (APARs)	458	6
Migration	431	6
Product Documentation	334	5
Replicated Data Queue Manager (RDQM)	256	4
Logging, Recovery	234	3
Connectivity	177	3
Performance	171	2

Notably, 27% of the issues remain "Unclassified", representing the largest category. "Communication Channel" problems, pertaining to connectivity between client and server systems or between servers, account for 11% of the issues. "Performance" related issues constitute the smallest category at

2%. Additionally, 7% of the recorded problems are associated with "Queue Managers".

To assess the extent of message re-processing, two attributes, "Company ID" and "Message ID" were used to identify distinct messages undergoing multiple processing attempts. A message is considered re-processed if its processing time exceeds 20 minutes, triggering a system timeout and resubmission. From a dataset of 1,237,370 messages, only a small proportion exhibited re-processing. As shown in Table 4.21, the majority of re-processed messages occurred only once, although one instance involved 51 re-processing attempts.

Table 4.21: Re-processed Messages.

	Reprocessed Times			
Times	1	2	11	51
Messages	281	6	1	1

As EDI messages traverse various components of a supply chain network, they undergo multiple transformations. One key feature of EDI is the functional acknowledgement (FA), which returns a status message confirming receipt of the original message [148]. The study examined the distribution of status message types to assess the frequency of message malformation. Table 4.22 presents the results.

Table 4.22: Queuing System: Messages by Ack Status.

Metric	**Status Message**						
	Accepted	**Waiting**	**None**	**Accepted With Errors**	**Rejected**	**Received**	**Partially Accepted**
Count	194,255	184,632	97,797	970	563	303	32
Percentage	40.59	38.58	20.44	0.20	0.12	0.06	0.01

The findings indicate that message malformation is relatively rare: only 0.12% of messages were marked as "Rejected", and 0.20% were classified as "Accepted with Errors". A deeper dive is needed to understand the breakdown of the status message and the associated EDI transformation. Table 4.23 shows the results.

Table 4.23: Messages: By Ack Status and Transformation.

Message by Status	Transformation	Count
Accepted	DeEnvelope	194,255
Waiting	Flat Translation	80,819
Waiting	XML Translation	61,013
Waiting	Batch XML Translation	41,154
None	XML Translation	41,198
None	Flat Translation	38,097
None	Batch XML Translation	14,279
None	Send	1,570
None	Extraction Translation	1,534
Accepted With Errors	DeEnvelope	970
Waiting	Flat Potential Translation	856
None	XML Potential Translation	632
Rejected	DeEnvelope	563
Waiting	XML Potential Translation	463
None	Flat Translation	460
Received	DeEnvelope	303
Waiting	Send	269
Waiting	Flat Translation	47
Waiting	Null	11
Partially Accepted	DeEnvelope	32
None	Flat Potential Translation	25
None	Null	2

The table shows status messages alongside their associated EDI transformations, enabling analysis of whether specific transformations are more prone to error. Messages labelled as either "Rejected" or "Accepted with Errors" share the same transformation, namely, the "DeEnvelope" service. The service extracts individual interchanges from a message based on their identified type, separating them into discrete units for business process handling. A comprehensive list of interchange types is available in [149]. Only a subset of messages return an acknowledgement receipt, as confirmed by DevOps, who noted that only messages marked as outbound, according to the "Direction" attribute, generate such status responses.

Most message transformations associated with the DeEnvelope service have a status of "Accepted". No other transformation type returns a status of "Accepted". In contrast, other transformation types display a broader range of status values, the majority being either "Waiting" or "None". Every message is associated with an EDI transformation, which is defined by the associated "Action" and "Category". Table 4.24 shows a data dictionary of some of the Category-type transformations.

Table 4.24: Message Category: Data Dictionary.

Name	Description
DeEnvelope Translation	Identifies the interchange type contained within a message and extracts them to separate messages to be handled by the business process
Flat Translation	One flat file in, many flat files out
XML Translation	Transforms XML to an EDI format
Batch XML Translation	One XML contains many documents which are then split into individual documents
Send	Send a message that has been previously deferred
Extraction Translation	Extraction of a document
Null Translation	Dependent on the "Action"
Flat Potential Translation	Run a preprocessing step on the data to see if it is bad before doing any transformations

The table identifies eight distinct types of message translations. Among these, the most frequently observed are DeEnvelope and Flat Translation. The DeEnvelope translation identifies the envelope type (e.g., an ACH inbound envelope) and invokes the corresponding business process. The process then uses a data extraction tool to parse and output the relevant data. In contrast, Flat Translation reformats flat file inputs into a standardised format, such as X12 or EDIFACT.

The classification of the ticketing system provides context as to where the identified problems lie within the enterprise queuing application. The 27% unclassified and another 25% made up of non-queue issues, such as installation, migration, APARs and documentation, equate to over 50% of the total problem tickets. The bulk of these problems may have easy fixes. Communication channels, security and transport layer make up another 28%. Further investigations may be warranted in the detailed classification of communication channels, security issues and unclassified messages.

4.4.6 Quantisation Noise

Timestamps in the enterprise dataset are recorded to three decimal places, introducing quantisation effects that significantly influence the distribution. For instance, when partitioning by "Batch By Category" and "Non-Batch By Category", the data exhibited a sharp peak near zero and a thin right tail, resulting in most observations concentrated in the first bin of the histogram (Figure 4.22a). To address this, Tukey's ladder of power was applied

to transform the data. The transformed density overlaid on the histogram (Figure 4.22b) shows multimodality suggesting quantisation artefacts. When redrawing the density plot for the head of the service time, four distinct Gaussian-like peaks are observed (Figure 4.23), likely reflecting quantisation noise rather than true underlying distributions.

(a) (b)

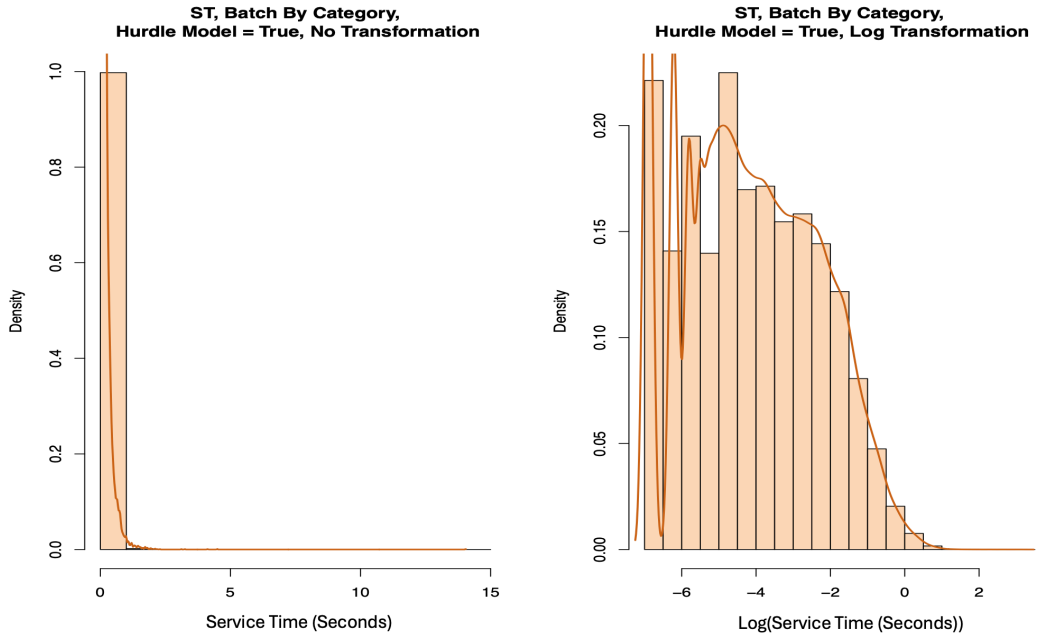


Figure 4.22: Histogram Batch: By Category, Seconds.

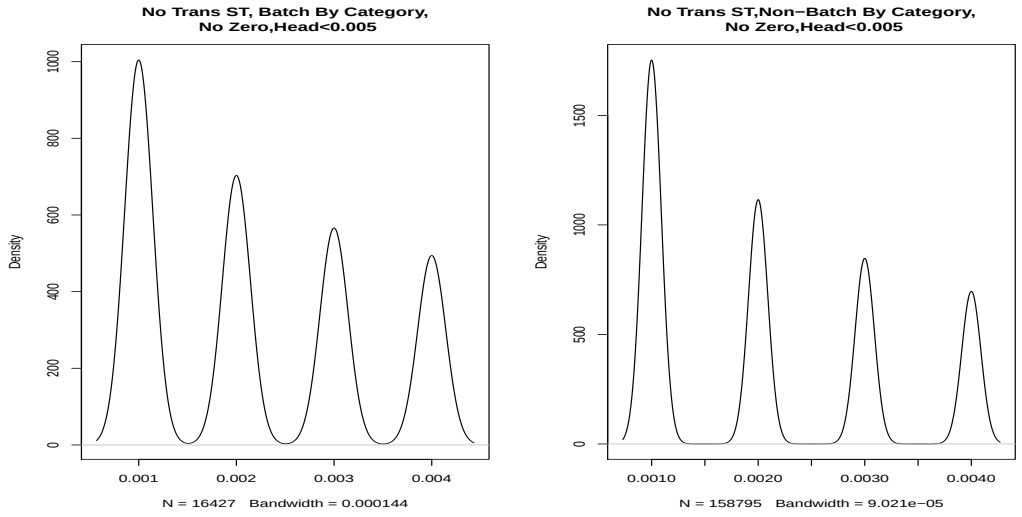


Figure 4.23: Probability Density: Batch and Non-Batch by Category.

Of course, the data contains four distinct values, quantised to 1 ms. However,

the smoothed density plot works well for the larger values, representing them as Gaussian distributions around the discrete values, because a significant amount of the probability is spread between values separated by 1 ms. Quantisation noise is an important factor for the data.

Truncation or rounding of timestamps has several implications. First, even if the underlying service times follow a specific distribution (e.g., Burr), the application of rounding or truncation alters the distribution, making it less representative of the original. Given the high volume of data and the concentration of values at the 1 ms resolution, such distortions are readily detected through statistical tests. Second, truncation may affect the temporal ordering of events, potentially undermining the accuracy of sequence-based analyses.

4.5 Discussion

Building on the limitations identified in Chapter 3.1, this chapter investigated whether structured partitioning strategies could improve the modelling of heterogeneous EDI message data. In Chapter 3.1, attempts to fit single parametric distributions to complete datasets produced poor GoF performance due to burstiness, heterogeneity, correlation, and quantisation effects. The framework presented in this chapter enabled more targeted modelling of specific message partitions, with some subsets exhibiting improved local distributional behaviour compared with the unpartitioned datasets examined previously.

Based on the methods outlined in Section 4.3, the following section presents the corresponding results, as detailed in Section 4.4.

4.5.1 Framework

A structured framework was created to support the modelling of queuing data and to guide the analytical process. The framework established a clear link between the research objective around the analysis of data and guided model selection across parametric and non-parametric approaches, including batch and non-batch type configurations. Such an approach promotes transparency and justifiability of outcomes and may be adapted or refined by other researchers for similar applications. The results of the investigation relating to the framework components are presented in Section 4.5.1.1, focusing on feature identification and selection.

4.5.1.1 Feature: Identification–Selection

During the modelling of EDI messages, different XML schemas were observed to be dependent on the "Mode" attribute. A total of 78 elements were initially identified, of which 23 were selected for further consideration, and this set was subsequently reduced to 8 elements deemed most relevant for analysis.

In addition, six keywords were extracted from the structured text of the log files to support modelling decisions.

By combining information from both the structured text and XML documents, a final selection of 14 elements was made from an initial set of 29. These elements were identified as the most relevant for modelling the service and interarrival times of EDI messages. The selected attributes may also serve as a useful reference for researchers and DevOps teams engaged in similar modelling tasks.

The results corresponding to the next component of the framework, feature classification, are examined in Section 4.5.1.2.

4.5.1.2 Feature Classification

The classification and labelling of combined attributes enabled more granular analysis of message behaviour. These classifications were designed to align with the objectives of queue system modelling; for instance, the classification model allowed for the distinction between single-message and batch-type message processing. Examining how elements of the classification framework affect data partitioning for parametric or non-parametric fitting helps identify interdependencies, enabling the intentional separation of correlated EDI messages so that model assumptions of independence are better maintained.

While the framework's primary purpose is to support feature classification within the modelling process, the understanding it provides of key attribute relationships may also inform practical applications, such as performance evaluation or stress testing. However, these considerations lie beyond the immediate scope of this chapter.

4.5.2 Parametric Modelling

The pronounced skewness and kurtosis observed in both service and interarrival times confirm that the dataset exhibits heavy-tailed characteristics, typical of queuing systems. The log transformation provides the greatest reduction in skewness; however, the distributions remain non-normal even after transformation. These findings suggest that traditional parametric models, which rely on normality assumptions, may be inadequate. Instead, this supports the case for applying partitioning strategies (e.g., head and tail modelling) and non-parametric techniques such as KDE or mixture models. Furthermore, the persistent concentration of values near zero, even post-transformation, indicates a potential influence of quantisation effects in timestamp recording, which is addressed in later sections.

4.5.2.1 Model: File Size

The range of file sizes within the dataset was analysed to assess the potential impact of file size on queue behaviour. The analysis also supports the provisioning of appropriate disk space for the queuing system. The head of the data was found to fit a uniform distribution, aiding in disk space estimation. In contrast, the tail exhibited discrete file sizes, indicating that KDE fitting may be more appropriate. Further work can be done to extend or re-segment the data to include more tail observations and yield a more stable KDE fit, and enhance modelling of large-file queue dynamics.

The hypothesis that batch messages could be identified by file size, under the assumption that large files are split into smaller components, was investigated. However, no consistent relationship between file size and message splitting was observed. Moreover, the "Source File Size" attribute was optional and frequently missing. As a result, alternative fields were used to determine whether messages were split.

In modelling the dataset, an objective was to determine whether messages represented single-type or batch-type behaviour. Various techniques were applied to evaluate whether specific features or parameters could distinguish between the two. The following sections present the results of this analysis, including an assessment of whether the service times for each message type could be effectively modelled using parametric distributions.

4.5.2.2 Model: Batch By Category

Messages were initially modelled using the associated "Category" attribute, per DevOps guidance. However, analysis revealed that messages within the same category could exhibit either single-message or split-message behaviour. Further investigation is needed to identify distinguishing features that reliably support this classification.

Parametric distribution fitting was also investigated for message service times classified under "Batch by Category". The final model approximated a log-normal distribution but failed the AD GoF test. The P–P plot revealed a high concentration of discrete values, indicating that quantisation of the data is likely due to timestamp truncation, which may have affected model fitting. Further partitioning into head and tail segments reinforced this observation, with the head exhibiting quantisation noise. Given these findings, messages classified as "Batch by Category" are more appropriately modelled using KDE techniques.

4.5.2.3 Model: Batch By Bundle

As an alternative approach to identifying batched messages, modelling was conducted on all messages that formed part of a bundle, defined as a sequence of messages where only the final message exhibited a service time greater than zero (Section 4.4.2.3). The method aimed to consolidate message bundles into a single effective arrival. The data was partitioned into head and tail segments. The tail approximated a log-normal distribution, while the head did not exhibit characteristics suitable for parametric modelling.

4.5.2.4 Model: Batch By Split Count

Analysis of messages based on the count of associated XML documents indicated that, from a queuing perspective, these messages exhibit batch-like behaviour, even if not all conform to batch semantics in an EDI context. Attempts to fit a parametric distribution across various partitions were unsuccessful. However, transformations applied to the head of the data showed close alignment with log-normal and Burr distributions. For the tail of the data, KDE provides a more appropriate modelling approach than parametric methods.

4.5.2.5 Model: Non-Batch By Split Count

Messages with an XML document count of one were analysed. The tail of the service times were successfully modelled using a Burr distribution, while the head appeared more suitable for KDE. Interarrival times were also examined; however, prior work in Chapter 3.1 identified the presence of a correlation. Modelling the correlations is left as future work.

4.5.2.6 Model: Non-Batch By Category

The "Non-Batch By Category" classification did not yield a suitable parametric fit; however, it provided clear evidence that quantisation significantly influenced the modelling outcomes.

4.5.3 Non-Parametric Modelling

Although not all aspects of the dataset were suitable for parametric modelling, many data partitions contained sufficient observations to support KDE-based modelling, provided quantisation was not a limiting factor.

4.5.4 Message Interdependence

Aspects of message interdependence were investigated using the classification model. A relationship was identified between message IDs and the associated XML document counts. To assess whether dependent and independent messages were associated with longer service times, a threshold of 1 second was used to define the distribution's tail. Fisher's exact test indicated no statistically significant dependent vs. independent association between message type and the tail of the service time distribution.

4.5.5 Queuing Problems

As discussed in Chapter 1.4, heterogeneous EDI messages may be affected by throttling, leading to retries and potential bottlenecks in the messaging system. To assess whether this was present in the current dataset, the logs were examined for evidence of throttling. Analysis confirmed that throttling was consistently set to zero during the observation period, indicating that no such effects occurred. The frequency of malformed messages requiring reprocessing was below 1. Overall, the enterprise dataset showed minimal issues related to EDI transformations or message formatting errors.

4.5.6 Quantisation Noise

Quantisation noise was not initially anticipated as a prominent characteristic of the dataset. The application of head–tail partitioning techniques was instrumental in localising the impact of this issue. Given that log file timestamps are often rounded to fixed resolutions (e.g., 1 ms), the challenges encountered in fitting parametric distributions to such data are noteworthy. These findings underscore the importance of accounting for truncation or rounding effects in modelling pipelines, as similar complications are likely to arise in other contexts involving discretised observational data.

Analysis of additional application log files revealed that many systems either already support timestamps with six or more decimal places or are transitioning toward higher-resolution timekeeping. It is recommended that DevOps teams consider enabling such high-resolution timestamping and monitor the typical time intervals relevant to their operational context. For instance, while nanosecond-level resolution may currently be sufficient, future reductions in typical job durations may necessitate even finer granularity to preserve the accuracy of performance diagnostics and mitigate quantisation-related artefacts in measurement. The effects of modelling timestamps at the lower precision are discussed in later chapters.

4.6 Conclusions

The study developed a framework to support queue simulation and performance evaluation of systems that handle EDI messages in a supply chain environment. The framework provides a structured approach for modelling queuing systems and identifying message characteristics that influence bursty and non-bursty behaviour. By leveraging EDI attributes, developers and performance engineers can assess their impact on queuing systems and make informed decisions regarding resource provisioning. In the context of smart environments, the framework enables implementers to evaluate system provisioning based on model selection outcomes. Data scientists and researchers can apply the framework to fit both parametric and non-parametric distributions, while organisations may integrate these modelling insights into existing technologies for improved operational awareness.

Notably, the dataset exhibited quantisation noise, likely due to millisecond resolution timestamps, which poses challenges for accurate modelling, particularly for short-lived jobs. The issue is anticipated to be common across similar datasets and highlights the importance of timestamp resolution in log data.

Future research could focus on exploring message-level correlations by incorporating additional EDI attributes. Such analysis may enhance modelling accuracy, especially in the head of distributions. Addressing quantisation artefacts remains an important avenue for improving distributional modelling in time-sensitive queuing systems.

In subsequent work, the effects of quantisation and a correlation observed in some parts of this dataset can be further investigated.

CHAPTER 5

Convergence and Goodness-of-Fit Issues in Distribution Modelling

Within this chapter, the impact of rounded data on distribution modelling, with emphasis on GoF assessment using the AD test is examined. The analysis investigates convergence and fitting challenges for Weibull, exponential, and log-normal distributions, highlighting conditions under which AD statistics return infinite values. Results show that rounding-induced zeros and extreme maxima are the primary drivers of divergence. Comparative analysis demonstrates that AD scores align closely with univariate measures of extremity (z-scores, variance) but not with multivariate distance (Mahalanobis). Convergence errors arise from mis-specified parameters and estimation breakdowns, with MME showing greater robustness than MLE in one parameter distributions.

5.1 Introduction

Distribution fitting involves identifying a probability distribution that best represents the observed data. Accurate fitting is essential for valid statistical inference, prediction, and hypothesis testing. Convergence issues can arise during the fitting process due to factors such as extreme values, rounded data or parameter values approaching boundary conditions (e.g., near zero).

Additional challenges include logarithmic terms, division by zero, roots of negative numbers, boundary constraints, and poor starting parameters [150, 108, 85, 78]. MLE and MME are two standard techniques (Section 2.11). In MLE, convergence failures can arise when the optimiser does not satisfy the stopping condition within the allowed number of iterations [150]. Given the fitting errors observed but not explicitly analysed or understood in previous chapters, it is essential to investigate the causes of distribution fitting failures. The chapter, therefore, aims to provide a comprehensive understanding of these errors, their underlying sources, and their implications for reliable model fitting and statistical inference.

In GoF testing, the AD test is widely applied because of its sensitivity to deviations in the distribution's tails. The AD test can produce extremely large values or return infinity (Inf) when numerical issues occur. Such cases warrant further investigation to identify underlying causes.

Convergence behaviour in Weibull, exponential, and log-normal distributions, and evaluates how MLE and MME perform under different data modifications is explored in this chapter. The conditions under which the AD test returns an infinite statistic are analysed, assessing the influence of parameter values and data characteristics. Finally, the relationship between AD statistics and distance-based metrics is quantified.

This chapter addresses three primary research questions:

1. Which data characteristics (e.g., zero values, near-zero values) contribute to convergence challenges in MLE and MME?
2. Under what conditions does the AD test return an infinite statistic, and are such occurrences associated with specific distributions or data characteristics?
3. In the presence of outliers, what relationships exist between the AD statistic and distance-based metrics, including Mahalanobis distance, z-score, and variance?

5.2 Data Overview

The data from this study will be synthetically generated for three distributions, Weibull, log-normal and the exponential distribution as defined in Table 5.1, to allow full control over distributional parameters, and ensure reproducibility of results. The generated data are then modified to investigate issues that may arise during model fitting, as described in the following sections. The three chosen distributions: Weibull, log-normal, and exponential, are all defined only for non-negative values, and some have zero probability density at $x = 0$. Consequently, exact zero values should not occur in data drawn from these theoretical distributions. Any zeros observed would therefore result from numerical rounding or data preprocessing.

Table 5.1: Parameter Values.

Default Parameters		
Log-normal	**Weibull**	**Exponential**
$\mu = 3, \sigma = 1$	$\beta = 0.5, \lambda = 2$	$\lambda = 1$

Sample sizes used for all distributions		
100	1000	10000

A limitation of this study its reliance on synthetic data, which may not capture the full complexity and variability presented in real-world datasets. While synthetic data allows controlled manipulation of parameters, the results may not generalise to practical real-world applications.

5.3 Methods

5.3.1 Convergence Errors

Within this section, convergence issues when fitting to different distributions are investigated. Using the three distributions and associated parameter values defined in Table 5.1, convergence tests described in Table 5.2 are conducted with a fixed sample size of 100. The five tests target problematic data conditions to assess how MLE and MME handle challenging scenarios:

Table 5.2: MLE/MME Convergence Tests.

Test Number	**Test Type**	**Description**
1	Rounded to Zero	Round the synthetic data.
2	UnRounded Close to Zero	Add zero values to the data
3	Negative Numbers	Add negative values to the data.
4	Distribution Range Exceedance	Add positive outliers to the data.
5	Parameter Values	Change the parameter values, bring them closer to zero.

"Rounded to Zero" rounds synthetic data to zero decimal places, i.e., to whole numbers, introducing exact zeros to evaluate whether rounded values affect distribution fitting.

"Unrounded Close to Zero" uses original synthetic data to six decimal places, testing the impact of values between 0.001 and 0.900. If insufficient values exist, the dataset is regenerated. The test evaluates whether small but positive values near the support boundary affect convergence.

For the "Negative Numbers" test, one negative value is added into the dataset to assess whether the specified distribution can support such data and to identify any fitting failures. Although the log-normal, Weibull, and exponential distributions have support only on the positive real line, a test with one negative value is deliberately included. The purpose is not to evaluate the ability of these distributions to represent such data, but rather to examine how the fitting procedures behave when encountering unsupported observations. Since the probability density function is zero for all $x < 0$, the log-likelihood becomes undefined or tends to $-\infty$ in the presence of negative data. Different estimation algorithms may therefore fail in different ways, for example, by terminating with a numerical error or failing to converge. Including this test helps to characterise such failures explicitly.

The "Distribution Range Exceedance" test adds a single large positive outlier to evaluate the robustness of MLE and MME to extreme values, particularly regarding variance and fit quality.

The "Parameter Values" test reduces parameter values by a factor of ten to examine whether proximity to zero affects fitting accuracy, allowing MLE and MME to re-estimate parameters at each step.

In summary, this section will provide a general classification of convergence and mis-specification errors, grouped by estimation method rather than by distribution. These general error patterns will set the stage for examining how they manifest in specific distributions, which is covered in the next section.

5.3.2 Convergence Behaviour in Different Distributions

Unlike the previous section, which will present convergence errors in a general form, this section analyses their occurrence within the individual distributions.

By applying the same test framework to the Weibull, exponential, and log-normal distributions, the results will highlight distribution-specific patterns of convergence and failure. The parameters for the test are defined in Table 5.1, the sample size is fixed at 100 unless otherwise specified, and the tests follow the test cases described in Table 5.2. These tests will be conducted using MLE and MME parameter estimation methods, which use the "fitdist" function from the "fitdistrplus" package [151].

5.3.3 GoF Test Statistic Returns Inf

As outlined in Section 2.9.4, the AD test statistic can return infinite values when samples contain zero or negative observations, which disrupts the logarithmic transformations used in its calculation. Even small positive values can have a disproportionate effect; for instance, $\log(0.001) = 3$ greatly increases the weighted contribution of the tails.

Unlike convergence failures, which occur during parameter estimation, Inf results arise after estimation, when the GoF statistic itself diverges under certain data conditions. To examine these conditions, synthetic datasets were generated for Weibull and exponential distributions using the parameter values in Table 5.1. Nine scenarios, described in Table 5.3, were tested with a sample size of 100 in most cases, and each experiment was repeated twenty times to account for random variation.

Table 5.3: AD GoF: Inf Scenario Tests.

Test NB	Test	Description
1	Extreme Outliers	Insertion of data points far beyond the typical range to induce large deviations between empirical and theoretical CDFs. Outliers defined as more than three standard deviations from the mean of a Gaussian distribution, where 99.7% of data should lie within this range [79].
2	Zero Variance	Datasets with identical values to test the effect of no spread on the AD statistic and determine if the lack of variability can produce an Inf result.
3	Near Zero Variance	Datasets with minimal spread to assess sensitivity to low variability and whether clustering of points close together inflates the AD statistic towards Inf.
4	Small Sample Size	Datasets with sizes below the recommended thresholds for AD critical values, testing whether insufficient data prevents correct distribution identification and leads to Inf results.
5	Create another distribution like log-normal and AD test on Weibull	Testing a mis-specified distribution (e.g., fitting Weibull data to log-normal) to assess robustness to distributional mismatch, which may return either an Inf value or a large AD statistic.
6	Extremely Large Sample Sizes	Testing computational limits and numerical stability using datasets up to one hundred million observations.
7	Observed Data: Perfectly Matches the Theoretical Distribution	Simulated data drawn exactly from the hypothesised distribution to assess behaviour under ideal conditions and determine if perfect agreement can still yield Inf results.
8	Negative Values	Introducing values outside the theoretical support of the fitted distribution (e.g., negative values for Weibull) to test whether the AD statistic returns Inf due to log-transform errors in the algorithm.
9	Many Zero Values	Inserting varying proportions of zero values (up to 30%) to examine their effect on the AD statistic's logarithmic terms and assess the threshold at which Inf results occur.

The AD test is conducted with the ad.test function from the gofTest R package [152], and parameter estimation for both MLE and MME uses the "fitdist" function from the "fitdistrplus" package [151].

5.3.4 Data Characteristics Underlying Inf AD Results

Having established the conditions under which the AD test statistic returns Inf values, the next step is to investigate the underlying data characteristics that contribute to these outcomes. In this section, the exponential distribution is analysed, with results evaluated at a rounding precision of three decimal places to identify patterns related to kurtosis, sample size, mean value, proportion of excessive zeros, and other distributional features.

5.3.5 Relationship Between AD Scores, Z-score and Mahalanobis Distance

Building on the previous section, this analysis examines how AD scores relate to alternative measures of extremity, with a focus on the effect of extreme

outliers. The objective is to determine the number of standard deviations from the mean at which the AD statistic diverges to infinity, using the Weibull distribution as a case study. Each test uses a sample size of one hundred and is repeated twenty five times to account for randomness. The results are measured at the individual data-point level rather than at the distributional level.

Two complementary distance-based metrics are adopted. The z-score measures univariate deviations from the mean. For example, a z-score of 3 indicates that the observation is three standard deviations above the mean. The Mahalanobis distance uses the covariance structure to capture multivariate deviations. A Weibull distribution is univariate, however, in this analysis, the Mahalanobis distance is calculated in a bivariate framework, where each empirical observation is paired with its corresponding Weibull theoretical quantile. By incorporating the covariance between these two dimensions, the measure captures the joint deviation of observed and expected values. The formula captures how far a point lies from the joint mean of the empirical and theoretical distribution space, while accounting for their covariance structure, making it a more complex measure than the z-score, which just looks at how far a point is from the mean in one dimension.

Using both metrics enables cross-validation of results and strengthens the robustness of the conclusions. Variance is additionally examined to capture changes in dispersion associated with extreme values. By varying the magnitude of the deviation from the mean, the relationship between the AD statistic and the other metrics (z-score, Mahalanobis distance and variance) is quantified, and thresholds at which the statistic diverges are identified.

Using these metrics alongside variance enables cross-validation of results and captures both univariate and joint effects of outliers. By varying the magnitude of deviations, the analysis quantifies the relationship between AD scores and these measures, identifying thresholds at which the AD statistic consistently diverges.

5.4 Results

5.4.1 Convergence Errors

Fitting errors were intentionally induced using the five tests described in Table 5.2. All observed errors are reported, together with their causes and the parameter estimation methods in which they occurred. In this section, the fitting errors are described. Later sections will provide more detail on these fitting errors.

Table 5.4: Convergence and Mis-Specification Issues.

Error	Problem	Issue	Method
Error in manageparam(.. values must be positive to fit an .. distribution')	Mis-specification	Starting values may not be in a structured list, and potential issues computing the starting values.	MLE MME
Error in checkparamlist(..arg, :start should not have NA or NaN values.)	Mis-specification	"checkparamlist" validates the parameters from "manageparam" function are aligned with the distribution's density function.	MLE MME
Error in fitdist(...the function mle failed to estimate the parameters, with the error code 100	Convergence	Optimisation errors when mledist() fails to converge.	MLE
simpleError in optim... function cannot be evaluated at initial parameters	Convergence/ Mis-Specification	Optimisation error, it occurs when the objective function returns NA or Inf values when evaluating the initial parameters	MME
simpleError in if (s.mu * obj..missing value where TRUE/-FALSE needed	Convergence/ Mis-Specification	Optimisation error, it occurs when variables defined in the error are not correctly initialised or contain unexpected or invalid values(e.g., NULL, NA, or Inf)	MME

The "fitdist" function verifies that any defined starting parameter values are either in a structured list or can be converted to one, and that the values are valid (e.g., no negative values when positive values are only supported). A "manageparam" error occurs if either check fails. If the precheck passes the initial assessment, then the "checkparamlist" function ensures that the parameters defined are valid and consistent with the distribution's density function, for example, defining a mean input parameter for a Weibull distribution. It also checks for NA or NaN values. Here, NA indicates missing or undefined input, while NaN reflects an invalid numeric result (for example, the operation $0/0$ produces NaN). Other operations that yield Nan include $\log(-1)$ in the real domain and $\infty - \infty$. Both cases must be ruled out to ensure that parameter estimation proceeds with well-defined inputs.

The "fitdist" error occurs only for MLE. It happens when the MLEDist function

fails to converge and returns error code 100, indicating that optim encountered an internal error [151]. For MME, "simpleError in optim" originates from the optim function, due to invalid starting parameter values or the presence of NA or Inf values in the dataset. The error is classified as both a fitting and convergence error, as it stems from the optimisation routine and may be triggered by either algorithmic calculations or problematic data characteristics. It can also occur if optim cannot handle specific inputs, resulting in NA or Inf values as outputs.

The "simpleError in if(s.mu* ....)" occurs during optimisation when input variables are incorrectly initialised or contain invalid values such as NULL, NA or Inf. It may also occur due to numerical instability in the objective function.

The variable mu determines whether optimisation seeks to maximise or minimise the objective. Variable s.mu, as sign(mu), sets mu's sign and adjusts the function logic accordingly. Depending on mu's value, s.mu adjusts the conditions used to compare current and previous objective values. In such cases, the role of mu and s.mu becomes critical.

5.4.2 Convergence Behaviour in Different Distributions

Cases where distribution fitting fails to converge are studied in this subsection, focusing on each distribution individually. Analysing the distributions separately highlights how specific data characteristics influence the stability of parameter estimation.

5.4.2.1 Weibull

Table 5.5 shows the individual results for the Weibull distribution.

Table 5.5: Weibull: Convergence Results — 100 Sample Size.

Weibull Shape: .5, Scale: 2				
Test	**Range**	**MLE**	**MME**	**Error Type**
Rounded to Zero		Checkparamlist	Checkparamlist	Fitting
UnRounded Close to Zero	0.001:0.900	No Error	No Error	NA
Negative Numbers	-1	Manageparam	Manageparam	Fitting
Distribution Range Exceedance	100k	No Error	No Error	NA
Parameter Values	Shape:[0.001:0.500], Scale:[0.200:4.000]	Fitdist	SimpleError	Convergence
Parameter Values: Sub-Tests				
Shape	**Scale**	**MLE**	**MME**	**Error Type**
≥ 0.040	$= 2$	No Error	No Error	NA
0.009:0.030	$= 2$	Fitdist	No Error	Convergence
≤ 0.040	$= 2$	No Error	SimpleError in optim	Convergence
$= 0.040$	$= 4$	No Error	SimpleError in if (s.mu * obj..)	Convergence
≤ 0.008[1]	$= 2$[1]	Checkparamlist	Checkparamlist	Fitting
$= 0.040$	≤ 0.400	Fitdist	No Error	Convergence

It is important to note that these errors are dependent on the characteristics of the synthetic data. For example, when generating synthetic data using the same parameter values, a "fitdist" error may occur in one instance, whilst a "manageparam" error might occur in another. Even small changes in the data's moments can influence the test results.

The "UnRounded Close to Zero" and the "Distribution Range Exceedance" tests did not suffer from convergence errors. Outliers up to 100,000 were added to the data for the "Distribution Range Exceedance" test. However, when extreme outliers were added to the data, no errors occurred for MLE or MME.

The "Rounded to Zero" and the "Negative Numbers" tests produced fitting errors, typically flagged as "checkparamlist" or "manageparam" errors. These are flagged as fitting errors, not convergence errors, because they indicate computational issues with starting values, often resulting in NaNs, likely due to invalid inputs for logarithmic transformations.

In the "Parameter Values" test, the shape was varied with a fixed scale set to 2 to explore how small values affect model fitting. As the shape decreases, the likelihood surface becomes flatter, making optimisation more difficult. It is observed from the table that convergence has occurred.

Additional tests in the "Parameter Values" case, shown in the lower half of the

[1] See *"Rounded to Zero"* test.

table, were conducted to identify the shape and scale parameter thresholds at which convergence starts and stops.

5.4.2.2 Exponential

Table 5.6 presents the results for each test applied to the exponential distribution.

Table 5.6: Exponential: Convergence Results:100 Sample Size.

Exponential Lambda:1				
Test	**Range**	**MLE**	**MME**	**Result**
Rounded to Zero		No Error	No Error	NA
UnRounded Close to Zero	0.001:0.900	No Error	No Error	NA
Negative Numbers	-1	Manageparam	Manageparam	Fitting
Distribution Range Exceedance	100k	Fitdist	No Error	Convergence
Parameter Values	lambda: 0.0001:100.00000	Fitdist	No Error	Convergence

The results show that MME produced no convergence or fitting errors, except in the "Negative Numbers" test, where a fitting error occurred, consistent with the exponential density being defined only for non-negative values.

For MLE, fitting errors occurred in the "Negative Numbers" test, and convergence errors occurred in the "Distribution Range Exceedance" and "Parameter Values" tests. In the latter, λ values ≤ 0.001 caused "fitdist" convergence errors. These arise because λ must be strictly positive; if $\lambda = 0$, the PDF is identically zero, implying a zero probability of any event, which is incorrect. If $\lambda < 0$, the PDF becomes negative for positive x, violating the requirement that probabilities be non-negative. For large λ, most probability mass is concentrated near zero, which can lead to numerical issues in the optimisation routine during exponential calculations.

5.4.2.3 Log-normal

Table 5.7 shows the individual results for the log-normal tests.

Table 5.7: Log-normal: Convergence Results:100 Sample Size.

Log-normal Mean:3 Standard Deviation:1				
Test	**Range**	**MLE**	**MME**	**Result**
Rounded to Zero		Manageparam	Manageparam	Fitting
UnRounded Close to Zero	Values:0.001:0.900	No Error	No Error	NA
Negative Numbers	-1	Manageparam	Manageparam	Fitting
Distribution Range Exceedance	100k	No Error	No Error	NA
Parameter Values	Standard Deviation: 0.001:10.000	No Error	No Error	NA

Both MLE and MME showed no convergence errors across all tests. Fitting errors arose in the "Rounded to Zero" and "Negative Numbers" tests due to the inability to take logarithms of zero or negative values, as required by the log-normal likelihood.

Extreme outliers did not produce convergence failures, consistent with the log-normal's suitability for heavy-tailed data. Varying σ between 0.0001 and 10 revealed no optimisation instability, while μ was held constant, as it shifts the distribution on the log scale without typically affecting numerical convergence.

5.4.3 GoF Test Statistic Returns Inf

The results of the study that determine the conditions under which the AD test statistic returns Inf for the exponential and Weibull distributions are provided in this section.

5.4.3.1 Weibull

Table 5.8 shows the results of the Weibull tests with cells highlighted in orange where the AD statistic returned an Inf.

Table 5.8: Weibull AD Inf / NULL Test (Repeat Tests=20).

Test NB	Test	Sample Size	Shape	Scale	Avg Var	AD Statistic = Inf	Min AD Statistic	Max AD Statistic
1	Extreme Outliers	100	0.5	2	999,458	True	N/A	N/A
2	Zero Variance	100	2	1	0	False	38.30	344.25
3	Near Zero Variance	100	0.5	0.00	0.00	False	0.56	2.74
4	Small Sample Size	6	1	1	1.11	False	0.39	3.43
5	Create another dist Like log-normal and AD test on Weibull	100	0.5	2	0.98	True	N/A	N/A
6	Extremely Large Sample Sizes	100 million	0.5	2	79.97	False	0.26	1.13
7	Observed Data Perfectly Matches The Theoretical Distribution	100	0.5	5	52.79	False	0.44	2.51
8	Negative Values	100	0.5	2	55.94	True	N/A	N/A
9	Many Zero Values	100	0.5	2	57.88	True	N/A	N/A

In the "Extreme Outlier" test, the mean value of the data was 104.06. Injecting a single value of $10,000$, resulted in the AD test statistic returning an Inf value, which reflects the AD tests' tail sensitivity. The deviation in the upper tail between the empirical and fitted CDF dominates the log-weighted terms.

The "Zero Variance" and "Near Zero Variance" tests do not return Inf values. However, the "Zero Variance" test results in a relatively high AD statistic, whereas the "Near Zero Variance" resulted in a comparatively low AD statistic. The behaviour may be distribution-dependent, and other distributions could exhibit different sensitivity to variance.

In the "Small Sample Size" test, a sample size $n = 6$ matched the smallest sample size for which tabulated critical values are available for the AD test (Case 0) [84]. The AD statistic was finite but large, leading to rejection of the null hypothesis that the sample was drawn from the specified distribution. The high value reflects the test's limited ability to identify the distribution with such a small sample, underscoring the importance of adequate sample size in GoF testing. Increasing the sample size improves the stability of estimated parameters, reduces sampling variability, and enhances the power of the test

to detect genuine departures from the theoretical model.

In the test "Create another distribution like log-normal and test on Weibull", a synthetic log-normal distribution with $\mu = 0$ and $\sigma = 1$, evaluated against a Weibull fit with (shape = 0.5, scale = 2), returned an infinite AD value. Changing the parameters of the log-normal distribution resulted in a high AD test statistic, rather than an infinite value. Distribution parameters affect the AD test's return results and influence the magnitude of the AD test statistic, showcasing the sensitivity of the AD test.

In the "Extremely Large Sample Size" test, a sample of one hundred million produced no infinite AD statistics. Attempts at one billion observations exceeded available memory (32GB). The AD test assumes exact arithmetic, but computers store numbers with finite precision. With extremely large samples, adding millions of terms may accumulate rounding errors, causing the computed statistic and p-values to differ from their true values.

The "Observed Data Perfectly Matches the Theoretical Distribution" was tested on a one hundred sample size, which did not result in an Inf value.

In the "Negative Values" test, the AD statistic was infinite. Distributions such as the normal, Cauchy, and logistic can accommodate negative values and should not produce infinite statistics under the same conditions. However, any further validation of this statement is beyond the scope of this research.

For the "Many Zero Values" test, zero values comprised 30% of the data and were progressively removed until none remained. With a sample size of 100, even a single zero produced an infinite AD statistic. For a Weibull distribution with shape parameter $\beta < 1$, the PDF diverges at $x = 0$, implying an infinite density at the lower bound. Therefore, the presence of zeros in the data introduces a sharp incompatibility between the empirical distribution and the theoretical model, leading the AD statistic to diverge under the parameter setting ($\beta = 0.5$, $\lambda = 2$).

5.4.3.2 Exponential

Table 5.9 shows the results of the exponential test.

Table 5.9: Exponential AD Inf / NULL Test (Repeat Tests=20).

Test NB	Test	Sample Size	Rate	Avg Var	AD Statistic = Inf/Null	Min AD Statistic	Max AD Statistic
1	Extreme Outliers	100	1	99997998.83	True	N/A	N/A
2	Zero Variance	100	1	N/A	N/A	N/A	N/A
3	Near Zero Variance	100	10000	0.00	False	0.30	1.99
4	Small Sample Size	6	1	1.28	False	0.25	2.69
5	Create another dist like log-normal and AD test on exponential	100	1	0.97	True	N/A	N/A
6	Extremely Large Sample Sizes	100 million	1	1.00	False	0.14	4.98
7	Observed Data Perfectly Matches the Theoretical Distribution	100	1	0.98	False	0.25	1.39
8	Negative Values	100	1	1.26	True	N/A	N/A
9	Many Zero Values	100	1	0.95	True	N/A	N/A

For the “Extreme Outliers” test, a single observation of 100,000 was inserted into the data, causing the AD statistic to diverge to infinity, for the same tail-deviation reasons noted in the previous Weibull outlier test.

In the “Zero Variance” test, an exponential distribution cannot have variance equal to zero because $Var(X) = \frac{1}{\lambda^2}$ would require λ to be infinite, implying zero time between events. If the time between events is zero, the data has no range. No range in the data implies no randomness in the events, meaning all events are stacked and not continuous.

For the “Small Sample Size” test, no infinite AD statistics were observed, as a limited sample size does not eliminate data spread or the continuity of random events.

The “Create another dist like log-normal and AD test on exponential” produced an infinite AD statistic. Synthetic data was generated with a log-normal distribution with $\mu = 0$ and standard deviation = 1. A log-normal distribution is right-skewed, with a long tail. The data points can take on large values with a relatively high frequency. The exponential distribution has a shorter tail, meaning large values occur less frequently. When data from a heavy-tailed

distribution like the log-normal is tested against an exponential fit, the AD statistic's tail weighting magnifies the discrepancy in the upper tail, which can cause the statistic to diverge to infinity.

The "Extremely Large Sample Size" test with one hundred million data points, and the "Observed Data Perfectly Matches the Theoretical Distribution" test produced no infinite AD statistic.

The "Negative Values" and the "Many Zero Values" tests produced infinite AD statistics because such values violate the exponential distribution's lower bound of zero.

5.4.4 Data Characteristics Underlying Inf AD Results

Table 5.10 presents descriptive statistics under various rounding precisions and sample sizes of the data characteristics that produced infinite AD statistics, using the exponential distribution as a test case. Cells highlighted in orange indicate the conditions under which the AD test returned an infinite statistic.

Table 5.10: Exponential: AD Inf Value Analysis.

Sample Size	Round Precision	Percent Values <1	Percent Zero Values	Min Data Point Value	Max Data Point Value	Mean	Standard Deviation	Variance (CV)	99th Percentile	Skewness
100	0	32%	32%	0	6	0.97	1.04	108%	4.45	2.12
100	1	66%	6%	0	3.5	0.93	0.82	88%	3.34	1.39
100	2	72%	1%	0	5.49	0.79	0.89	112%	4.85	2.76
100	3	55%	1%	0	4.99	1.15	1.03	89%	4.47	1.42
1000	0	40%	40%	0	6	0.86	0.93	107%	3.59	1.58
1000	1	61%	5.7%	0	7.7	1.02	1.08	106%	5.09	2.17
1000	2	62%	0.2%	0	7.5	1.02	1.02	100%	4.54	1.96
1000	3	62%	0.1%	0	6.83	0.99	0.96	96%	4.35	1.72

Analysis of the data reveals several contributing factors. For both $n = 100$ and $n = 1000$, rounding to zero decimal places produced many zeros (32% and 40%, respectively). Increasing decimal precision sharply reduced values rounded to zero (to below 1% for $n = 100$ and 0.3% for $n = 1000$ at two or more decimals), though extreme maximum values continued to appear, particularly at $n = 100$. Despite these distortions, mean and standard deviation values remained close to the theoretical expectation ($\mu \approx 1$) with coefficients of variation near 100%, indicating that the distribution's mean and variance were preserved. Skewness values (1.39 – 2.76) were consistent with the exponential's right-skewed shape

but increased when large maxima occurred. For the remaining highlighted cells, the divergence appears to stem from the interplay between the "Max Data Point Value" and its corresponding "99th Percentile". When the observed maximum exceeds the theoretical 99th percentile, the distribution's upper tail becomes disproportionately stretched, increasing deviations from the expected exponential form. The tail distortion increases the sensitivity of the AD statistic, often driving it to infinity even when other descriptive statistics (e.g., mean, variance) remain near their theoretical expectations.

Overall, the results reinforce that infinite AD results are associated with both high proportions of zeros and/or extreme values, which intensify deviations in GoF. These findings are consistent with the patterns observed for the Weibull case (Table 5.8), confirming that zero inflation from rounding and extreme values are the primary causes of divergence in AD statistics across both distributions. When rounding to zero decimal places for the exponential distribution, AD Inf values were consistently returned for both MLE and MME, rendering the tests inconclusive.

5.4.5 Relationship Between AD Scores, Z-Score and Mahalanobis Distance

Figure 5.1 presents the results of the tests in four scatter plots. The first three plots compare AD scores with Mahalanobis distance, z-scores, and variance, respectively, while the fourth plot illustrates the relationship between z-scores and Mahalanobis distance.

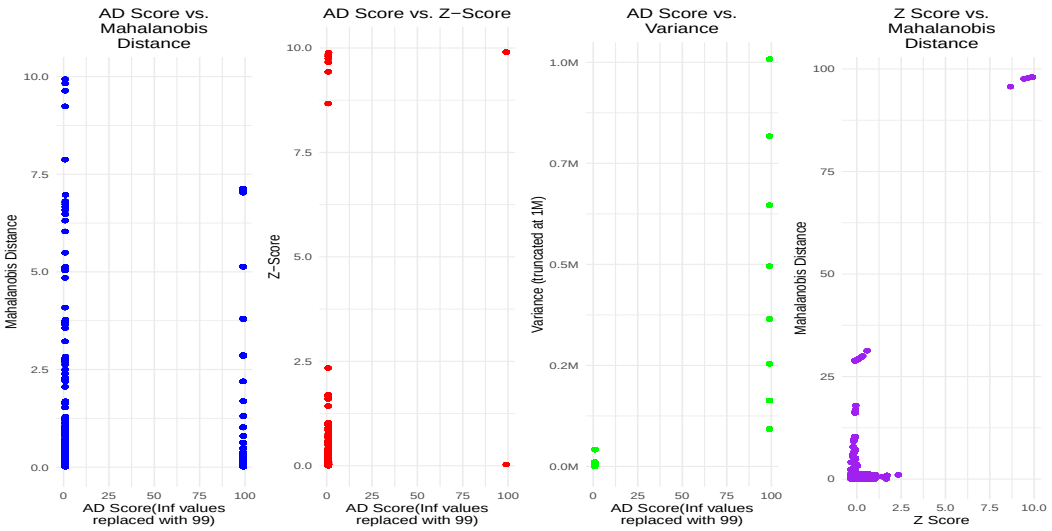


Figure 5.1: Weibull: AD Scores Versus Distance Metrics.

In the first three plots, infinite AD values are replaced with 99 to enable clearer visualisation, and on the third plot most variance values fall below one million, but a few extreme outliers (approximately 10^{20}) distort the scale; thus, the y-axis was truncated at 10^6 to enhance interpretability.

These graphs provide complementary perspectives on deviation from the expected distribution. The Mahalanobis distance, computed by pairing empirical data with Weibull theoretical quantiles, captures joint deviations and accounts for covariance between observed and expected values. In contrast, the z-score reflects univariate distance from the mean in standard deviations.

As discussed in Section 5.3.5, this study looks at the relationship between the AD score and different distance metrics. The left-most plot displays Mahalanobis distance against the corresponding AD score for each data point, illustrating how the two measures relate at the observation level. No clear linear relationship is observed. At a 5% significance level, observations with Mahalanobis distances greater than 2.44 were considered large Mahalanobis distances (observations more than 2.45 units away from the multivariate centroid in standardised Mahalanobis space). A large Mahalanobis distance does not necessarily correspond to extreme tail deviations that drive the AD test to reject the null hypothesis.

The second plot shows the relationship between AD scores and z-scores. The

pattern is non-monotonic. For z-scores near 10, the AD statistic takes on two outcomes, either close to 0 or close to 99 (replacing Inf). AD scores near 100 occur at both small and large z-scores, indicating that the AD statistic does not scale smoothly with deviations from the mean but instead shows threshold-like behaviour.

The third plot shows AD scores against variance, where the y-axis for variance was truncated at one million. 75% of the results have a variance value less than one million. 40% of data points are concentrated at a variance value of (246), even across a wide range of AD scores. For the majority of the data, variance is not strongly related to changes in the AD statistic. A few points appear around or near the upper truncation limit (1M). These are extreme cases where both the AD Score and variance spike together, indicating instability or sensitivity to outliers in the data.

The right-most plot compares z-scores with the Mahalanobis distance. No clear linear relationship is apparent between the two distance metrics, though two clusters of values are observed.

To quantify these relationships, a Spearman rank correlation test was conducted due to non-linearity between AD scores and each distance metric. The results are shown in Table 5.11.

Table 5.11: Spearman Test: AD Score vs Other Metrics.

Comparison	**Spearman ρ**
AD Score vs Mahalanobis Distance	-0.06
AD Score vs Z-score	0.47
AD Score vs Variance	0.91
Z-score vs Mahalanobis Distance	0.09

Note: p-value = 0.00

Using the critical values from Table 2.6, the AD test statistic shows no correlation with Mahalanobis distance. A moderate correlation is observed with the z-score ($\rho = 0.47$), and for variance, the correlation coefficient ($\rho = 0.91$) indicates a strong monotonic relationship. Assessing the relationship between the z-score and Mahalanobis distance shows a weak correlation. These results are not visually apparent in the scatter plots, as the Spearman method measures rank-based association rather than linear association.

5.5 Discussion

5.5.1 Convergence Errors

Convergence test results can be grouped into two categories: mis-specification and convergence errors. Mis-specification errors arise from the "manageparam" and "checkparamlist" functions, due to invalid or poorly structured starting values, incompatibilities between parameters and the chosen distribution, or undefined (NaN) values. Convergence errors include MLE error code 100 and MME "simpleError...", indicating failure to estimate parameters, while other "simpleError..." messages are linked to mis-specification, typically arising from invalid inputs (e.g., NaN values or incompatible parameter choices) or uninitialised variables where required values were not properly defined before estimation.

5.5.2 Convergence Behaviour in Different Distributions

Convergence behaviour varied across the Weibull, log-normal, and exponential distributions. All three produced errors under invalid inputs, such as zero or negative values, reflecting their domain restrictions.

Beyond this common issue, patterns diverged. The Weibull was the most sensitive, with frequent failures when the shape parameter approached zero. The log-normal was stable but still failed on zero and negative values. The exponential, as a single-parameter model, was the most robust, converging where the other two failed, though errors still arose for invalid inputs.

MLE and MME showed broadly similar error types, though MME proved more robust in exponential cases, while both methods struggled with extreme Weibull parameters.

5.5.3 GoF Test Statistic Returns Inf

Across both Weibull and exponential distribution fitting, the AD test returned an Inf value for extreme outliers, model mis-specification with a heavy-tailed distribution, and support violations such as zero or negative values. These tests reflect the test's sensitivity to the tails, and violations of the assumed distribution prevent a valid fit. Large AD statistic results arose in zero-variance and minimal sample size $n = 6$ tests. Near-zero variance, perfectly specified models, and extremely large samples (one hundred million) yielded stable AD

statistics, although finite-precision arithmetic may introduce minor rounding errors.

5.5.4 Data Characteristics Underlying Inf AD Results

The results from the exponential study confirm that infinite AD statistics arise primarily from rounding-induced zeros and extreme maxima above the 99th percentile. Larger samples reduce the proportion of zeros, but extreme values persist across precisions and continue to distort the upper tail. Thus, as with the Weibull case, both zeros and extreme maxima are the key drivers of divergence in the AD test.

5.5.5 Relationship Between AD Scores, Z-Score and Mahalanobis Distance

The results highlight important differences in how distance metrics relate to the AD statistic. Mahalanobis distance showed no meaningful relationship with AD scores ($\rho = -0.06$), consistent with the scattered pattern in the charts, indicating that multivariate distance does not reliably capture deviations relevant to the AD test. In contrast, a moderate positive correlation was observed between AD scores and z-scores ($\rho = 0.47$), indicating that large standardised deviations are linked to divergence in the AD statistic. At first glance, this seems inconsistent with the scatter plots. The difference arises because Spearman correlation measures monotonic rank associations rather than linear trends. While small z-scores consistently map to AD = 0 and larger z-scores often correspond to divergent AD values, the abrupt transitions break the visual continuity in the plots, even though the rank-based relationship still produces a moderate positive correlation. Variance, on the other hand, produced only a threshold-like behaviour, with AD scores capping to either 0 or 100, however, the correlation results indicate a strong monotonic relationship. To clarify this, Figure 5.2 presents a scatter plot of ranked values for AD and variance. Spearman correlation measures how well the ranks of two variables move together, not the raw values, this ranked plot highlights the monotonic relationship between AD and variance ($\rho = 0.91$).

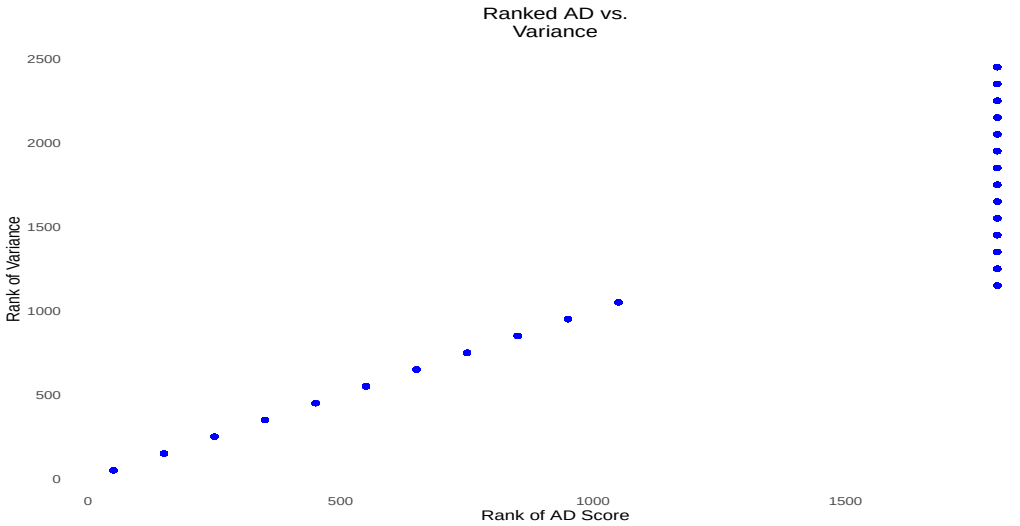


Figure 5.2: Weibull: Rank AD Scores Versus Variance.

These results suggest that variance and z-scores offer complementary support to AD values. At the same time, Mahalanobis distance reflects alternative notions of extremity that do not align well with the AD test.

5.6 Conclusion

Concerning the data characteristics that contribute to convergence challenges in MLE and MME, the results showed that rounding-induced zeros, near-zero values, and support violations (zero or negative data values) are the primary sources of failure.

The analysis also revealed that convergence behaviour differs by distribution. The Weibull distribution is highly sensitive to extreme shape parameters. The log-normal is more robust but still fails under zero and negative values. The exponential displayed the strongest robustness, converging successfully in cases where the others did not.

Regarding the conditions under which the AD test returns an infinite statistic, it was found that extreme maxima, heavy-tailed mis-specification, and zero values consistently produced divergence across both Weibull and exponential distributions.

In examining the relationship between the AD statistic and distance-based metrics, z-scores and variance (after logarithmic transformation) displayed

strong monotonic associations with AD scores, whereas Mahalanobis distance showed no meaningful correlation.

Taken together, these findings show that convergence behaviour and AD divergence are driven primarily by data characteristics and distributional assumptions, and that the AD test responds more strongly to extreme values in single variables (such as large z-scores or variances) than to multivariate distance measures like the Mahalanobis distance.

CHAPTER 6

Rounding Effects on Parameter Estimation

When estimating a probability distribution, the parameter values are important. MLE and MME are two commonly used techniques for estimating parameter values from an observed dataset. The performance of these two techniques may be impacted by data characteristics such as sample size, distributional complexity, and quantisation. This research compares MLE and MME under varying parameter values, sample sizes, and rounding based on the Weibull, exponential, and log-normal distributions. The results show that MLE provides better distributional fit for multi-parameter distributions. The findings also show that the performance of MLE and MME depends on the distribution, data characteristics, and evaluation metrics used, emphasising the need to assess parameter estimation using multiple measures.

6.1 Introduction

When working with a probability density distribution, it is essential to have a solid understanding of statistical methods that can estimate distribution parameters. MLE and MME parameter estimation methods are the focus on this chapter. As previously mentioned in Section 2.11, MLE estimates the parameters of a probability distribution by maximising the likelihood function, whereas MME estimates the parameters of a probability distribution by comparing theoretical and empirical moments. Building on the review

of these two methods, the objective is to determine which method, MLE or MME, provides the closest fit to a distribution and under what conditions. Previous work demonstrated that quantisation significantly affects log-normal parameter recovery and GoF assessment [153], motivating further investigation into how rounding influences parameter estimation performance across different distributions and estimation methods.

The following questions are addressed:

1. How does changing the parameters affect the shape of the distribution, and does rounding impact parameter estimation for both MLE and MME?
2. Which of the two techniques provide the best parameter estimates?
3. From the parameter estimates provided, how do the parameters affect AD and CvM GoF tests?
4. What are the limitations specific to each methodology, and how might these limitations influence the accuracy of parameter estimation and fit to a distribution?

Table 6.1 lists the distributions to be tested, respective parameter values, and sample sizes. Hypothesis testing will be conducted using AD and CvM GoF tests to evaluate the effectiveness of MLE and MME.

Table 6.1: MLE/MME Test Cases.

Test Number	**Test**	**Repeat Tests**	**Description**
1	Shape and scale association with parameter estimates	20	Varying the parameter values to compare the returned parameter estimates.
2	MLE versus MME Performance Comparison	20	Compare MLE and MME estimates
3	GoF: Comparison	20	Comparing the AD and CvM GoF results from estimated parameters.
4	MLE and MME limitations	n/a	Comparing MLE and MME limitations.

Default Parameters		
Log-normal	**Weibull**	**Exponential**
$\mu = 3$, $\sigma = 1$	$\beta = 0.5$, $\lambda = 2$	$\lambda = 1$

Sample sizes for all distributions		
100	1000	10000

The findings in this chapter are expected to generalise to other continuous distributions where estimation depends on precise values or moments.

6.2 Data Overview and Limitations

This study applies conventional statistical theory, which is developed under the assumption of exact (non-rounded) samples, to synthetic datasets that have been rounded. The aim is to identify and quantify the limitations of applying conventional statistical theory to rounded data. These methods aim to reduce the parameter estimation bias and GoF distortions identified in Chapter 6, where quantisation was shown to systematically affect parameter recovery for common probability distributions.

The data from this study will be synthetically generated for three distributions, the Weibull, log-normal and the exponential distributions as defined in Table 6.1, to allow full control over distributional parameters, and ensure reproducibility of results.

For the Weibull distribution β will be assigned to the shape parameter. In the literature often expressed as either β or k, and the two notations are used interchangeably. For this thesis, β is adopted for consistency. A rounding precision ranging from zero to six decimal places will be the basis for these tests. Rounding to 0 decimal places defines the "rounded" data, rounding to 6 defines "unrounded" data. Tests will not be conducted on rounding between one and five decimal places unless problematic issues need to be explored further. Some tests will be repeated to support randomness and variability.

Synthetic datasets were generated from Weibull, log-normal, and exponential distributions using the parameter configurations defined in Table 6.1. For each distribution, datasets were generated at sample sizes of 100, 1,000, and 10,000 observations. Quantisation was applied by rounding the generated data to zero decimal places, while data rounded to six decimal places was treated as effectively unrounded. Each experiment was repeated 20 times to account for stochastic variability in parameter estimation. The selected parameter values were chosen to evaluate the behaviour of MLE and MME across different distributional shapes and scales under rounded and non-rounded conditions.

The limitations of using synthetic data have already been discussed in Chap-

ter 5.2.

6.3 Methods

6.3.1 Choice Of Parameters

The choice of parameters directly influences the behaviour of the distributions. As shown in Figure 6.1, increasing the shape parameter (β) progressively changes the distribution from highly skewed to nearly symmetric, underscoring the importance of parameter selection in distributional modelling.

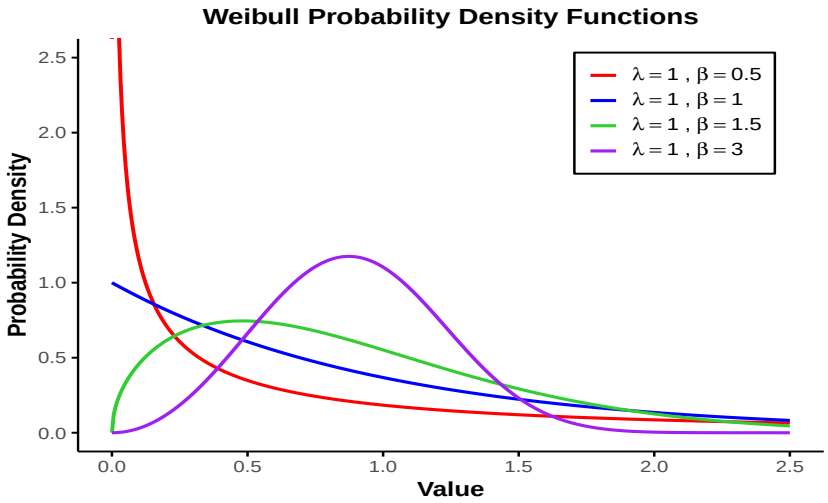


Figure 6.1: Weibull probability density functions for varying shape (β) parameter values with fixed scale $\lambda = 1$ values.

Note: The figure is generated in R and illustrates standard Weibull density behaviour as described in [62].

A Weibull distribution has a failure rate that can either decrease ($\beta < 1$), increase ($\beta > 1$), or remain constant ($\beta = 1$) over time [64]. $\beta < 1$ indicates higher probabilities of early failures. For this research, $\beta = 0.5$ is selected as a baseline value, and variations around this baseline are used to examine the influence of smaller and larger β values on distribution fitting. The scale parameter (λ) affects the shape of the PDF. For this research, a midpoint value of $\lambda = 2$ is used as a baseline, providing a balance between shorter and longer tails in the distribution, which also offers insight into how MLE and MME handle outliers in the data and will be revisited in greater detail in Section 6.4.1.1.

The exponential distribution is a special case of the Weibull distribution with $\beta = 1$ [62]. The rate parameter λ must be positive; therefore, $\lambda = 1$ was chosen [28].

A log-normal distribution is defined by μ and σ, representing the mean and standard deviation of the variable's natural logarithm. Because the logarithm of a log-normal variable follows a Normal distribution, reference values from the Normal distribution can guide parameter selection. To avoid excessive peak clustering while still providing a reasonable starting point, $\mu = 3$ and $\sigma = 1$ were chosen for this study.

6.3.2 A Quick Guide to MLE and MME Techniques

With the tests and parameters defined, this section briefly revisits the calculation steps of MME and MLE, previously introduced in Section 2.11. The discussion begins with MME.

MME selects parameter values such that the theoretical moments of a distribution match the corresponding sample moments. For the log-normal distribution, if a random variable X is log-normally distributed, then the transformed variable $Y = \ln(X)$ follows a normal distribution [154]. In this case, MME relies on the first and second moments of Y, given by:

$$\mu = E[Y], \qquad \sigma^2 = \mathrm{Var}(Y)$$

For the two-parameter Weibull distribution, the shape and scale parameters are obtained by matching the sample moments to the theoretical moments. The shape parameter β is first estimated by solving:

$$\frac{\overline{x^2}}{(\bar{x})^2} = \frac{\Gamma\left(1+\frac{2}{\beta}\right)}{\left[\Gamma\left(1+\frac{1}{\beta}\right)\right]^2},$$

where $\overline{x^2} = \frac{1}{n}\sum_{i=1}^{n} x_i^2$ is the sample second moment and $\bar{x}$ is the sample mean. Once β is obtained, the scale parameter is calculated as:

$$\lambda = \frac{\bar{x}}{\Gamma\left(1+\frac{1}{\beta}\right)}$$

For the exponential distribution, the rate parameter λ represents the mean rate of occurrence of events per unit time, expressed as the reciprocal of the mean. MME estimates λ using the first moment: $\hat{\lambda} = \frac{1}{\bar{X}}$

The key MME relationships for the Weibull, log-normal, and exponential distributions are summarised in Table 6.2.

Table 6.2: Summary of MME Moments: Weibull, Log-normal, Exponential Distributions.

Distributions	**Parameters**	**MME Moments**	**Relationships between sample moments and distribution parameters**
Weibull	Shape(β), Scale(λ)	Mean, Variance	$E(X) = \lambda\Gamma\left(1+\frac{1}{\beta}\right)$, $\text{Var}(X) = \lambda^2\left[\Gamma\left(1+\frac{2}{\beta}\right) - \left(\Gamma\left(1+\frac{1}{\beta}\right)\right)^2\right]$
Log-normal	μ, σ^2 (log-mean, log-variance)	Mean, Variance	$\mu = \ln\left(\frac{\bar{X}^2}{\sqrt{\bar{X}^2+s^2}}\right)$, $\sigma^2 = \ln\left(1+\frac{s^2}{\bar{X}^2}\right)$
Exponential	Rate(λ)	Mean	$\lambda = \frac{1}{\bar{X}}$

With the moments and calculations defined, attention now turns to MLE. MLE as previously discussed in Section 2.11.3, uses the likelihood function which expresses the probability of observing the given data as a function of the distribution's parameters. MLE estimates parameters by choosing the values that maximise the likelihood, thereby identifying the parameter set that makes the observed data most likely.

In practice, estimation begins with an initial guess of the model parameters. These values are refined through iterative numerical optimisation methods, such as NR, until convergence is achieved. The optimisation process is supported by the partial derivatives of the log-likelihood with respect to the parameters, which indicate how the estimates should be updated. For a generic function f, initially set to zero, they find the slope of the likelihood at the point in either the x ($\frac{\partial f}{\partial x}$) or y ($\frac{\partial f}{\partial y}$) direction with the likelihood for Weibull as an example defined in 2.19

The partial derivative with respect to x reflects how the function f changes as x varies, while y remains constant. Similarly, the partial derivative with respect to y indicates how f changes when y varies, holding x constant. Having outlined the general process, the following illustrates its application to the Weibull distribution. In this case, the parameters are β and λ, and the partial derivative of the log-likelihood with respect to λ is:

Formula : Weibull partial derivative of the log-likelihood function with respect to λ

$$\frac{\partial \ln L(\lambda, \beta)}{\partial \lambda} = -\frac{n\beta}{\lambda} + \beta \sum_{i=1}^{n} \frac{x_i^{\beta}}{\lambda^{\beta+1}}$$

With respect to β, as β changes λ remains constant. The partial derivatives for β for Weibull is:

Formula : Weibull partial derivative of the log-likelihood function with respect to β

$$\frac{\partial \ln L(\lambda, \beta)}{\partial \beta} = \frac{n}{\beta} + \sum_{i=1}^{n} \ln(x_i) - n \ln(\lambda) - \sum_{i=1}^{n} \left(\frac{x_i}{\lambda}\right)^{\beta} \ln\left(\frac{x_i}{\lambda}\right)$$

After computing the partial derivatives, a log of the likelihood is calculated, allowing for the summation of all densities, making it summation and maximisation easier. Setting the derivatives to zero and solving for the slope of the likelihood with respect to β and λ helps estimate the parameters by maximising the log-likelihood.

The Weibull distribution has been presented in detail to illustrate the application of MLE. For the exponential distribution, MLE for the rate λ parameter is $\hat{\lambda} = \frac{1}{\bar{X}}$, which is the same as the MME estimator. For the log-normal distribution, the worked solutions follow a similar approach and are well documented in the literature [69]. To avoid redundancy, they are not provided.

It is important to distinguish between the PDF and the likelihood function. The PDF is a function of the data given fixed parameter values, whereas the likelihood function is a function of the parameters given fixed observed data. Although both are expressed using the same mathematical form, their interpretations differ. To illustrate this, see the example in Figure 6.2. The histogram to the left of the image is a PDF function of the data, given a particular set of parameter values, defined by the data on the x-axis and the frequency of the values on the y-axis. In contrast, the image to the right, the likelihood function, is a function of the parameters given a particular set of observed data, defined by the parameters on the x-axis. The figure to the left shows the probability of a particular data value for a fixed parameter, while the right-hand figure shows the likelihood of a specific parameter value for a fixed dataset.

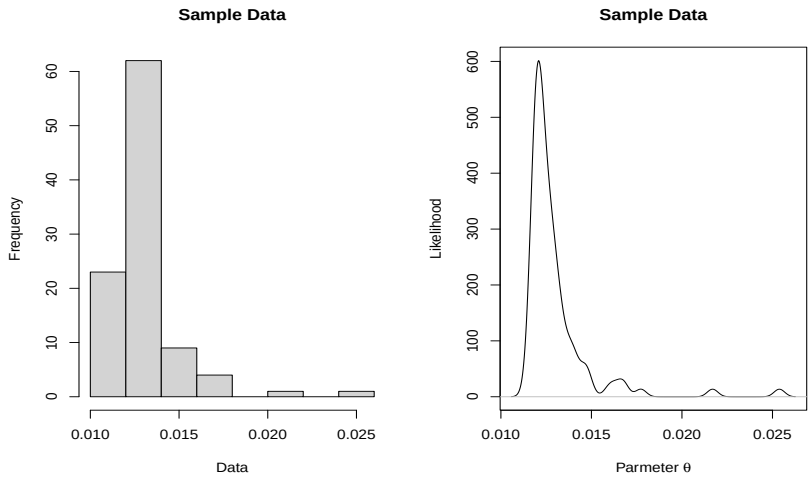


Figure 6.2: (a) Histogram, (b) The likelihood function given the observed data.

6.3.3 Parameter Changes: Effects on Shape and Estimation

With a walk through of the calculation steps complete, MLE and MME will be used to estimate the parameters of a given distribution. Changing parameter values affects both the shape and spread of the distribution, often altering skewness and kurtosis. A high kurtosis (leptokurtic distributions) is characterised by a sharp peak and heavy tails, whereas a lower kurtosis (platykurtic distributions) has a flatter peak and shorter tails.

The performance of MLE and MME is influenced by distributional parameters that govern dispersion, skewness, and kurtosis. Changing these parameters alters the distribution's shape and may hinder the accuracy of the derived return estimates. For instance, heavy tails can introduce outliers that MLE cannot efficiently estimate if the model does not account for this feature, while MME may perform poorly when extreme values distort the moments and lead to unreasonable estimates. Distributions are characterised by location, scale, and shape parameters or by mean and standard deviation (e.g., the normal and log-normal distributions).

Location shifts the distribution along the x-axis, scale stretches or compresses it, controlling the spread, and shape controls higher-order features such as skewness and kurtosis, thereby influencing asymmetry and tail behaviour.

Some distributions use a rate parameter that specifies the frequency of events. Although not all distributions contain all parameter types, together they govern the form and behaviour of the distribution and, in turn, the performance of estimation techniques such as MLE and MME.

The test "Shape and scale association with parameter estimates" from Table 6.1 is examined. It uses the results from the "Parameter Values" test from Table 5.2. The primary objective of this research is to analyse how changes in distribution parameters affect the distribution's shape and the accuracy of MLE and MME parameter estimation techniques.

Before discussing the results, it is important to understand how extreme observations can significantly influence the shape of a distribution and the methods used to estimate its parameters. To better understand extreme values in the data, Figure 6.3 illustrates how such an extreme point can arise within a dataset.

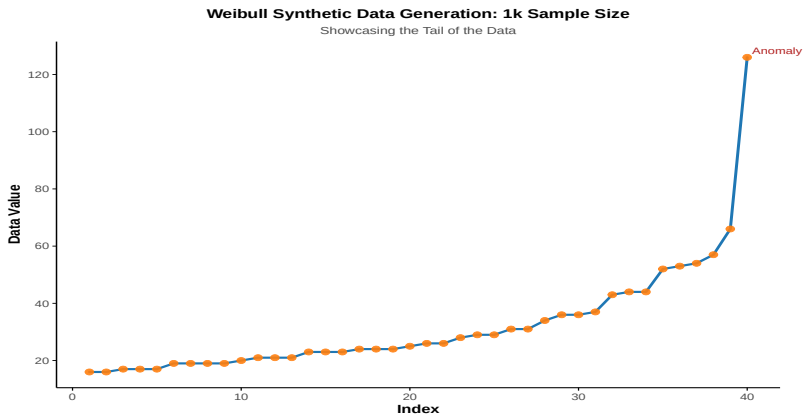


Figure 6.3: Anomalous Point.

The figure was created using synthetic data from a Weibull distribution with a one thousand sample size. The x-axis in this scatter plot represents the index position of the tail of the data, with the first 960 observations omitted to focus on the single outlier, which is the feature of primary interest. The x-axis shows that, on average, the data points are 1.28 units apart. However, the anomalous data point is 59.80 units from the preceding point. The anomaly occurs only at the last data point, showing it is in the data's tail. When analysing the anomalies that occurred when generating the synthetic data, frequent observations showed that anomalous points tend to appear in the tails

rather than near the beginning or middle of the dataset.

Table 6.3 outlines the specific tests planned for this section. No minimum starting values are defined for the log-normal test, as the "Parameter Values" test did not produce any errors when changing σ.

Table 6.3: MLE/MME: Changing Parameter Estimates.

Min Starting Values : Weibull ($\beta >= 0.040$)			
Min Starting Values: Exponential ($\lambda >= 0.001$)			
Sample Sizes :100, 1,000			
Methods: MLE, MME			
New Tests			
Distribution	**Test**	**Shape**	**Scale**
Weibull	Default Parameters	.5	2
Weibull	Decrease Shape	2	2
Weibull	Increase Shape	.8	2
Weibull	Decrease Scale	.5	1
Weibull	Increase Scale	.5	4
		Rate	
Exponential	Default Parameter	1	
Exponential	Decrease Rate Parameter	0.5	
Exponential	Decrease Rate Parameter	0.2	
Exponential	Increase Rate Parameter	3	
Exponential	Increase Rate Parameter	5	
		Mean	**Standard Deviation**
Log-normal	Default Parameter	3	1
Log-normal	Decrease Mean	2	1
Log-normal	Increase Mean	5	1
Log-normal	Decrease Standard Deviation	3	0.5
Log-normal	Increase Standard Deviation	3	1.5

6.3.4 A Comparison of MLE Versus MME Parameter Estimation

From Table 6.1, this section addresses the test "MLE versus MME Performance Comparison". The test will compare MLE and MME performance across the three already defined distributions using the predefined parameters and sample sizes. The data will be generated to six decimal places, rounded to integers, and repeated twenty times to incorporate randomness.

As shown in Table 5.7, fitting errors mainly arose from zero and negative values due to logarithmic transformations. To mitigate this, a constant of 1 will be added to the data where fitting errors may occur, especially in the Weibull and log-normal distributions. Although larger constants could be used, a small

additive value of 1 is preferred to preserve data integrity while ensuring valid logarithmic calculations [17]. Values < 1 do still cause fitting errors.

6.3.5 MLE Versus MME Distribution Fitting GoF Comparison

A comparison of the fit of MLE and MME using AD and CvM GoF tests is examined. The objective is to compare and contrast the GoF test statistic results across different sample sizes and roundings using the parameter estimates from MLE and MME, and not based on the parameters defined during synthetic data creation. The testing strategy for this study is similar to Section 6.3.4, except that these tests look at the GoF results rather than the parameter estimates. The test covers the "GoF: Comparison" from Table 6.1.

6.3.6 MLE and MME limitations

The limitations of MLE and MME in the context of the findings presented in earlier sections are examined. Both methods rely on different principles for parameter estimation, and their performance is shaped by the characteristics of the data, sample size, and the underlying distributional assumptions.

The test "MLE and MME limitations" is covered in this section. The convergence analysis in Chapter 5 demonstrated that rounding causes zero values and support violations (zero or negative inputs), which are the primary drivers of convergence failure, with sensitivity varying across distributions. Weibull estimations were highly unstable when the shape parameter was large ($= 4$), whereas the log-normal failed at zero, and negative values, and the exponential remained comparatively robust.

Building on these findings, the present section considers how these data characteristics and distributional sensitivities constrain the effectiveness of MLE and MME, and how these constraints should be taken into account when evaluating model performance. GoF tests, such as AD and CvM, emphasise different aspects of the empirical distribution, while numerical accuracy measures, such as RMSE, provide direct comparisons between estimated and true parameters. The limitations of MLE and MME must be interpreted not only in terms of their theoretical properties, but also in relation to the evaluation metrics employed and the characteristics of the data under study.

6.4 Results

In the upcoming sections, results will be visualised using ten 1D jittered stripcharts built in R, showing MLE and MME estimates for different parameter settings. Each plot will include lines to indicate true parameter values for visual comparison. These charts display individual data points with slight horizontal jitter to show the distribution of the results while avoiding overlap using stacked points. The y-axis represents the estimated parameters, the bottom x-axis shows the method and parameter, and the top x-axis displays the true distribution parameters. The first and third rows of charts are for non-rounded data. The second and fourth row of charts displays rounded data. The first two rows are for a sample size of one hundred, with the last two rows for sample sizes of one thousand.

6.4.1 Parameter Changes: Effects on Shape and Estimation

6.4.1.1 Weibull

To understand how changes in parameters affect the distribution, Figure 6.4 presents a Weibull distribution for non-rounded data in two different scenarios: one where the shape parameter remains constant while the scale parameter varies and another where the scale parameter remains constant while the shape parameter varies.

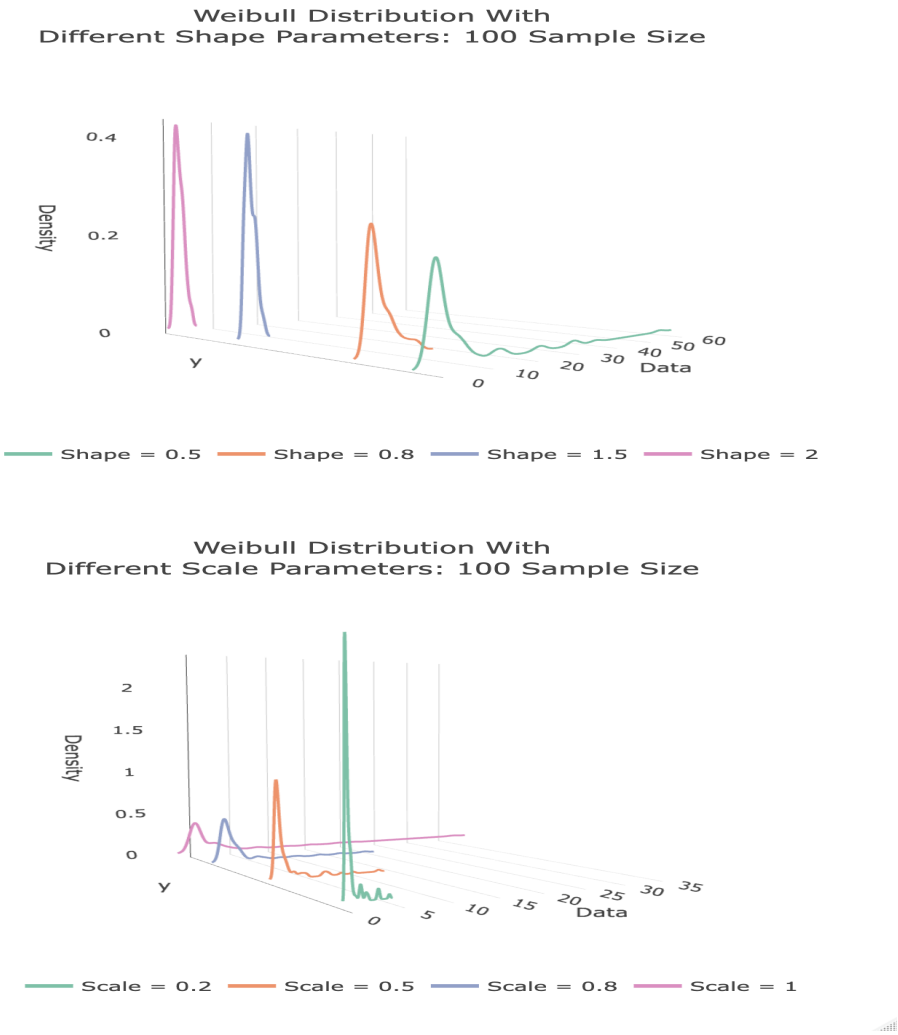


Figure 6.4: Comparison of Weibull: Shape and Scale Parameter Changes - Non-rounded Data.

The first figure shows that as the shape parameter increases, the dispersion decreases, leading to shorter tails and higher peaks in the density, resulting in fewer potential outliers, and less overall dispersion. The second figure shows that as the scale parameter increases, the distribution's tails become more elongated, indicating a greater potential for outliers, while the peak of the density flattens, increasing dispersion.

To explore how variations in the shape and scale parameters influence parameter estimation, strip charts in Figure 6.5 are used. These charts illustrate the outcome of the tests for both rounded and non-rounded data. Threshold

lines for the shape and scale parameters have been added to the charts to support easy visualisation of how close the estimates approximate to the actual parameter values. Each chart includes two lines: one for shape and one for scale. The blue boxes are the shape parameters for MLE, while the brown boxes are for MME. The green boxes correspond to the scale parameter for MLE, while the orange boxes are for MME.

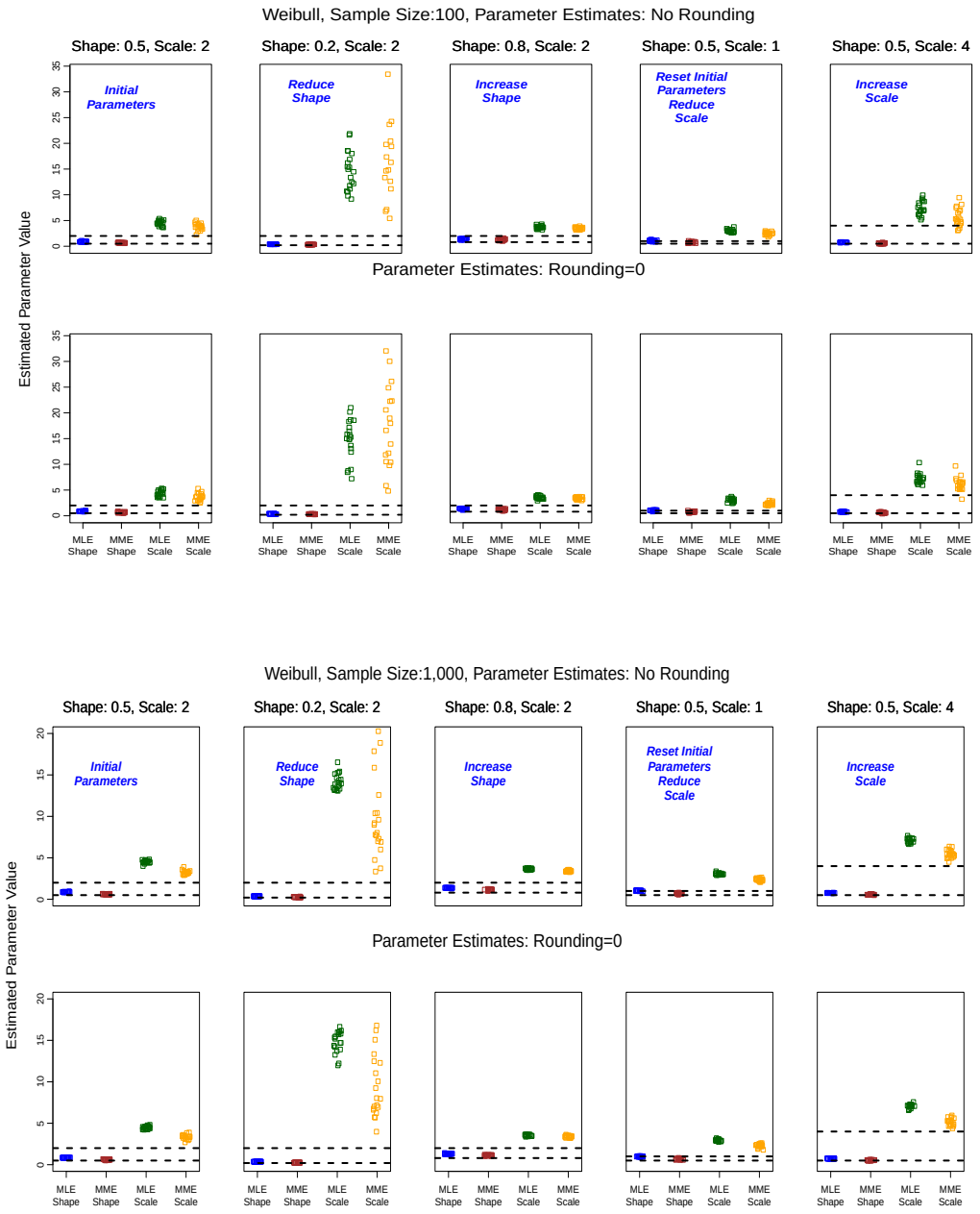


Figure 6.5: Comparison Results: Changing Weibull Shape and Scale Parameter Value Tests.

The results show that across both sample sizes, shape parameter estimates (blue and brown boxes) remain consistent between MLE and MME. MLE estimates cluster around 1, while MME estimates are typically closer to 0.5.

As seen in Figure 6.4, reducing the shape or increasing the scale flattens the peak and increases the tails, leading to more outliers. Reducing the shape parameter to values closer to zero, like 0.2 or increasing the scale parameter to a value of 4 or greater, hinders the performance of both MLE and MME for the scale parameter estimates, which can be seen in the second and fifth columns of the charts. Either more variations occur, or there is a jump in scale parameter estimates for MLE and MME. However, MLE has less variance when estimating the scale parameter than MME. MLE and MME tend to provide unreliable estimates for the scale parameter, regardless of rounding.

Variability is also influenced by sample size. Results from samples of one hundred show greater dispersion (y-axis max 35) compared to samples of one thousand (y-axis max 20). The corresponding skewness values are 5.8 on the one hundred sample size and 23.6 on the one thousand sample size. Although the data becomes more positively skewed with larger samples, the parameter estimates display reduced dispersion, indicated by the y-axis (max 20), improving estimator stability. The contrast suggests that there is no causal relationship between the skewness of the data and the dispersion of the parameter estimators. Instead, the observed variability is more likely linked to numerical instability in the optimisation process, particularly as the estimated parameters approach values near zero, where the likelihood surface becomes flatter, and the optimiser struggles to converge.

6.4.1.2 Exponential

The exponential distribution has one parameter, which simplifies the testing process. As discussed in Chapter 5, results from the "Parameter Values" test indicated that modifying the rate parameter can lead to convergence issues when the rate is small. Rate parameter values should be greater than 0.001 to avoid convergence errors. The test will take these considerations into account. For the exponential distribution specifically, rounding does not materially affect estimation results and is therefore not considered a significant factor. Figure 6.6 shows how changing the rate parameter affects the distribution.

The figure that as the rate increases, the peak of the density increases and the tails of the distribution are reduced, indicating fewer potential outliers. As the rate is reduced, the peak of the density flattens, and the tails of the

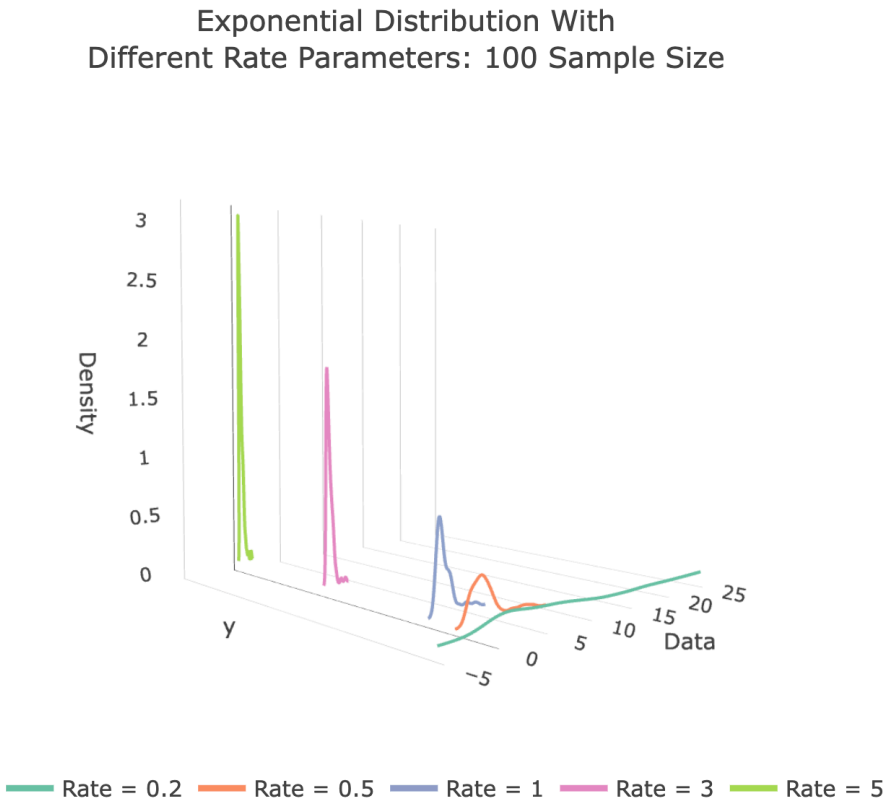


Figure 6.6: Comparison of exponential rate parameter changes - non-rounded data

distribution become more elongated, indicating potential outliers in the data. Figure 6.7 shows the results of the tests.

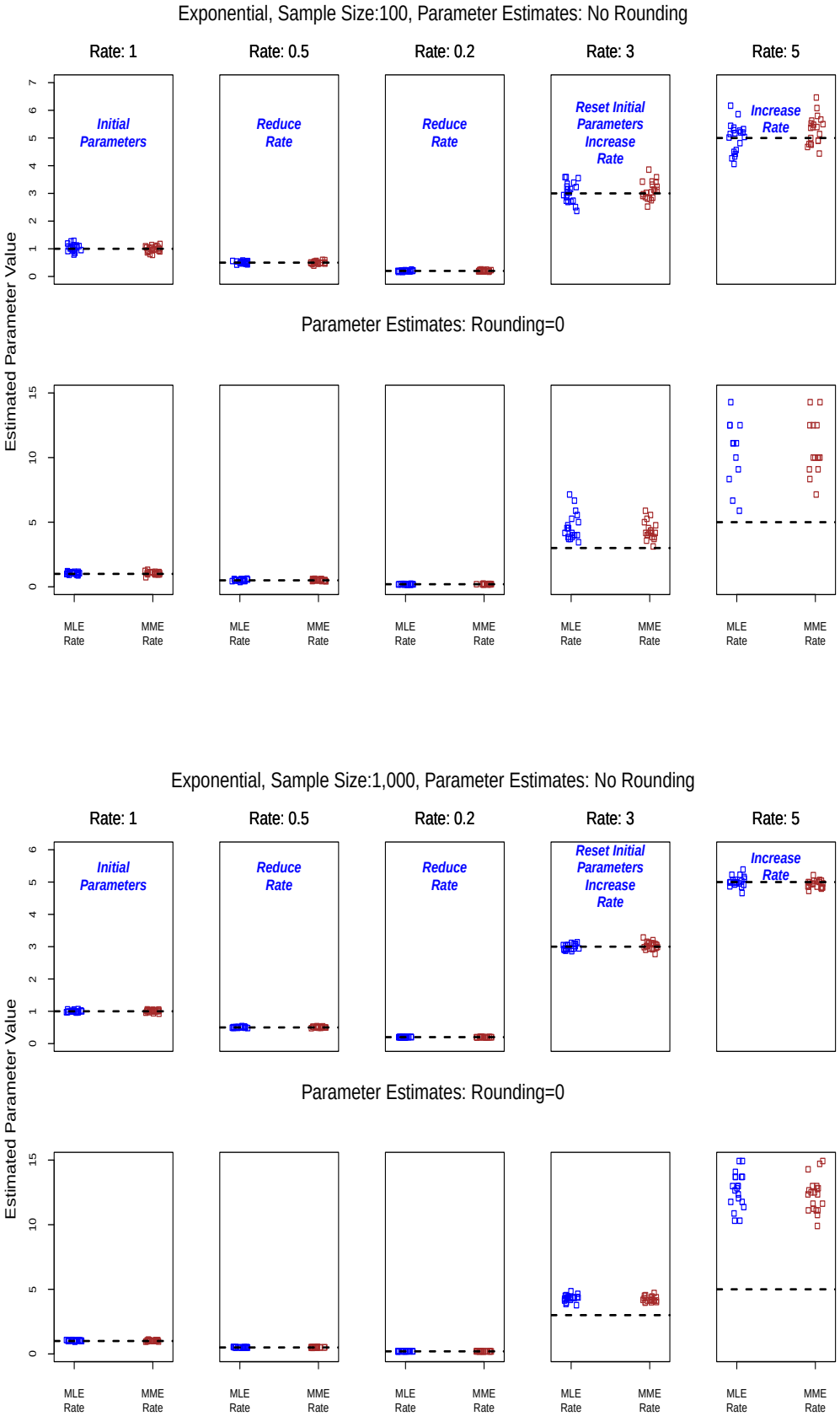


Figure 6.7: Comparison Results: Changing Rate Parameter.

The figure shows that the y-axis limits for rounded and non-rounded data differ. The rounded data has a bigger range of y-axis values compared to the non-rounded data.

Referring back to Figure 6.6, as the rate increases, fewer outliers should be produced, while as the rate is reduced, outliers should be more pronounced in the data. The charts show that as the rate is reduced, there is little variation in the results, and both MLE and MME provide consistent estimates for both rounded and non-rounded data and different sample sizes, which aligns with the analysis from Figure 6.6. MLE and MME are identical for the exponential distribution, and one expects identical behaviour.

As the rate increases, there is more variation in the return results. However, MLE and MME are still similar in their behaviour. At fixed sample size, higher rate values produce greater dispersion in the estimates (as $\text{Var}(\hat{\lambda}) \approx \lambda^2/n$), whereas increasing the sample size reduces this variability. Accordingly, with the one hundred sample size, the spread across rate values is clearly visible, with the one thousand sample size the dispersion contracts. Hence, differences (e.g., between $\lambda = 3$ and $\lambda = 5$) appear less pronounced. Rounding does influence parameter changes.

6.4.1.3 Log-normal

For the log-normal distribution, both the mean and standard deviation parameter values are varied. The mean parameter will not be tested in MLE and MME fitting, as it shifts the central tendency of the distribution without altering its shape. Nevertheless, it will be varied in some cases to illustrate how changes in the mean affect the location of the distribution. Figure 6.8 shows the results of the parameter changes.

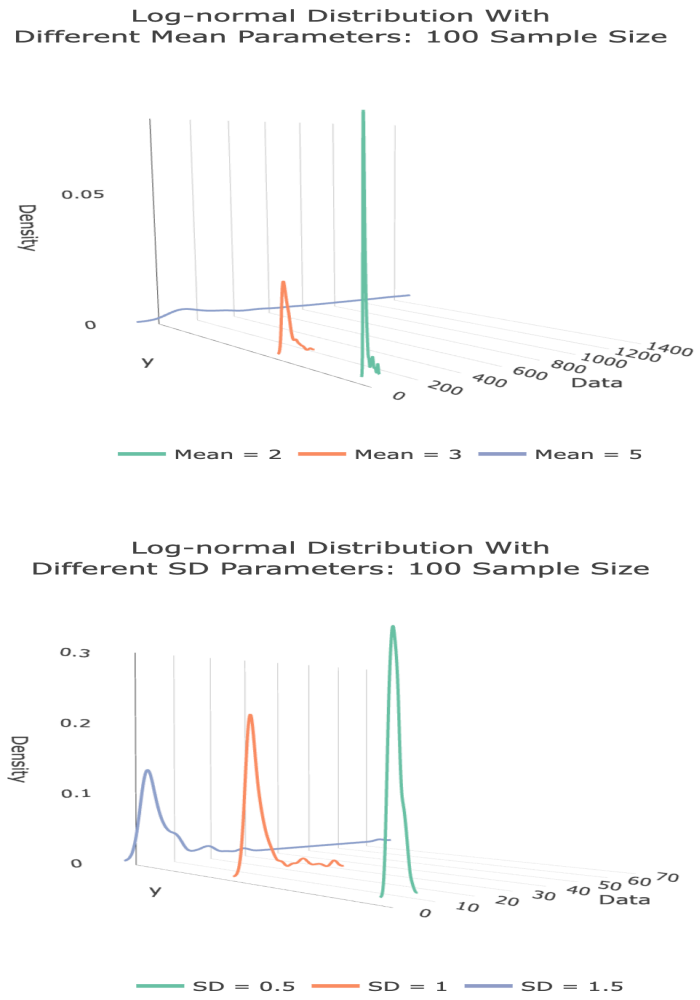


Figure 6.8: Comparison of log-normal Mean and Standard Deviation Parameter Changes - Non-rounded Data.

Higher mean values flatten the peak and extend the tails, while lower mean values sharpen the peak and shorten the tails. Greater standard deviations broaden the distributions with elongated tails, whereas smaller values yield a sharper, more compact form. Having shown how changes in the mean and standard deviation parameters affect the shape of the distribution, Figure 6.9 presents the test results.

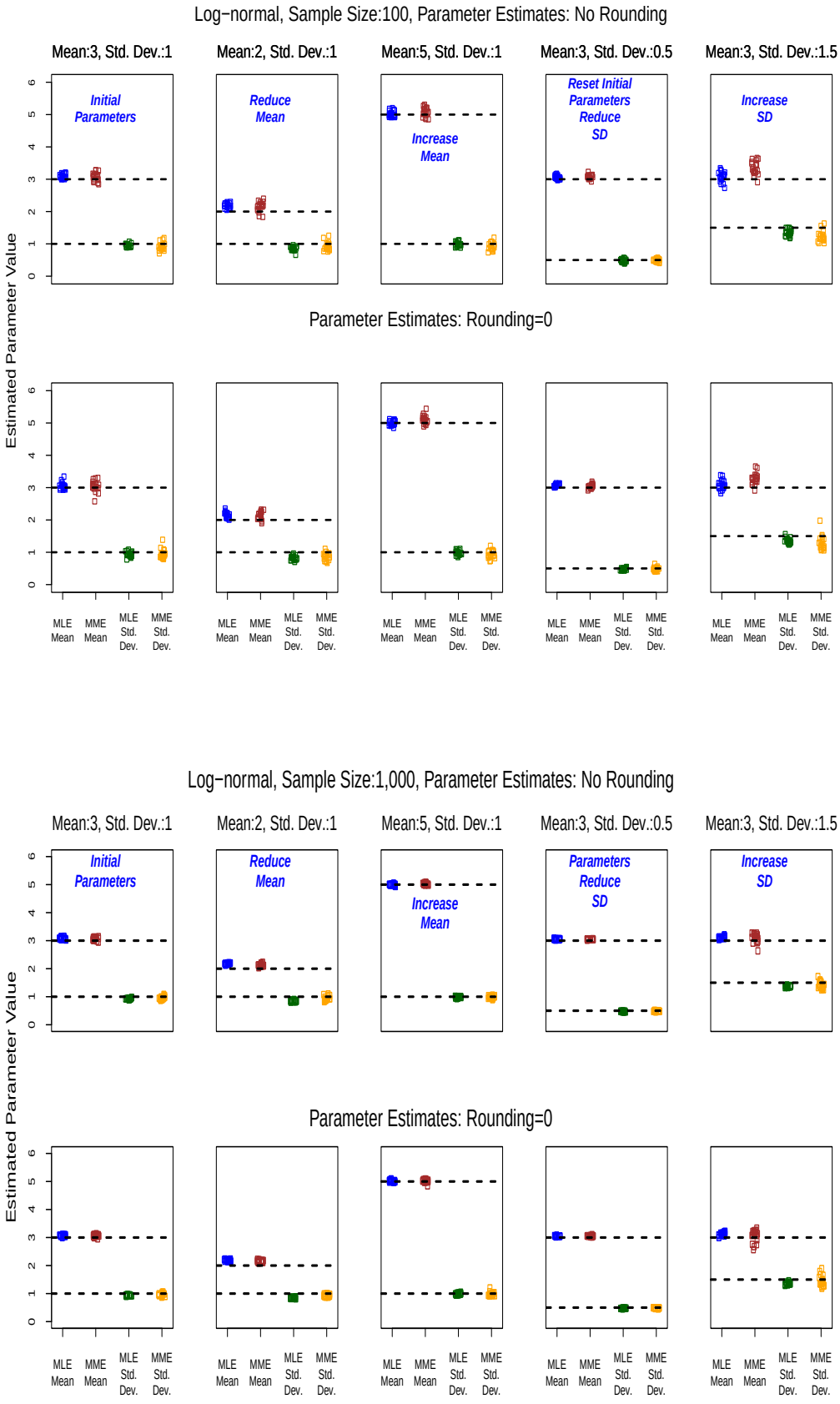


Figure 6.9: Comparison Results: Changing log-normal Mean and Standard Deviation Parameter Value Tests.

First, it is noted that for all charts in the figure, the y-axis limits are set to 6. It is observed that both MLE and MME are consistent in the majority of tests. Increasing or decreasing the mean parameter does not impact the fit to the distribution. However, decreasing the standard deviation parameter decreases the variance for both MLE and MME when compared against the initial parameters from the charts in the first column against that of the charts in the fourth column. Also, increasing the standard deviation parameter increases the variance in the return estimates, more so for MME than for MLE, and more so for the mean parameter at both sample sizes, for both rounded and non-rounded data. It is also observed that rounding the data slightly impacts the one thousand sample size for MME when the standard deviation is increased, causing more variance in the return results and affecting the fit to the distribution.

6.4.2 A Comparison of MLE Versus MME Parameter Estimation

To compare parameter estimates from MLE and MME, the results in the following subsections are presented using the same jittered scatter plots described in earlier tests. Each plot shows MLE and MME estimates on the y-axis, with sample sizes and parameter names on the bottom x-axis, and the estimation method (MLE or MME) on the top x-axis. Threshold lines indicate the true parameter values used to generate the synthetic data, enabling visual comparison.

Estimation accuracy is quantified using RMSE, which provides a single measure of prediction error. RMSE is derived from the mean squared error (MSE), representing the average squared difference between the predicted and true parameter values.

6.4.2.1 Weibull

For the Weibull distribution with $\beta = 0.5$ and $\lambda = 2$, the results of the tests are shown in Figure 6.10.

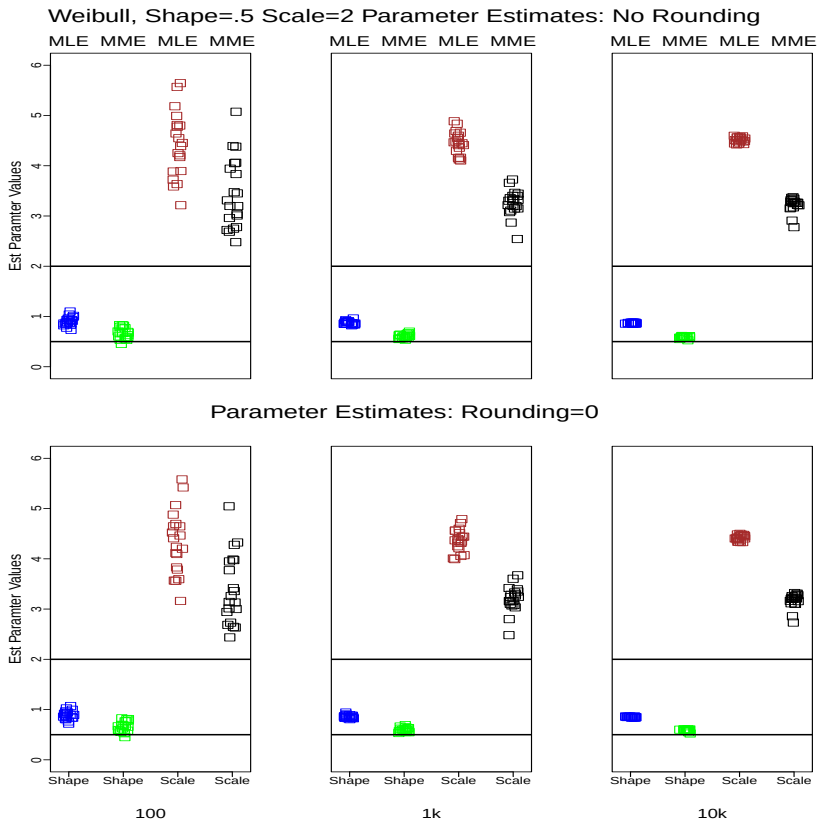


Figure 6.10: Weibull: MLE-MME Parameter Estimation.

The figure illustrates how closely the estimated shape (blue for MLE, green for MME) and scale (brown for MLE, black for MME) parameters align with the true values. For the shape parameter, MME consistently produces lower estimates than MLE, regardless of whether the data is rounded. For the scale parameter, both methods tend to overestimate, with a high degree of variation, though MME estimates are closer to the true value. Increasing the sample size reduces the variance for both methods, particularly for the scale parameter at the ten thousand sample size. Visually, rounding has minimal impact on parameter estimation, but to statistically confirm the visualisation conclusions, Table 6.4 shows the RMSE values for this study.

Table 6.4: MLE/MME Parameter Estimates: Weibull RMSE.

Precision	**Type**	**100**		**1k**		**10k**	
		Shape	**Scale**	**Shape**	**Scale**	**Shape**	**Scale**
6	MLE	0.41	2.50	0.38	2.47	0.34	2.36
6	MME	0.19	0.13	0.11	1.21	0.09	1.23
0	MLE	0.40	2.41	0.37	2.40	0.35	2.42
0	MME	0.19	1.54	0.11	1.23	0.09	1.18

The lowest RMSE values are highlighted in blue. MME consistently produces lower RMSE than MLE, indicating closer alignment with the true parameter values. Rounding has minimal impact overall, though a notable improvement is seen for MME on the scale parameter at the one hundred sample size, where RMSE decreases from 2.50 to 1.54 (shown in orange in the table).

6.4.2.2 Exponential

For the exponential distribution with $\lambda = 1$, Figure 6.11, shows the results of the estimates provided by MLE and MME.

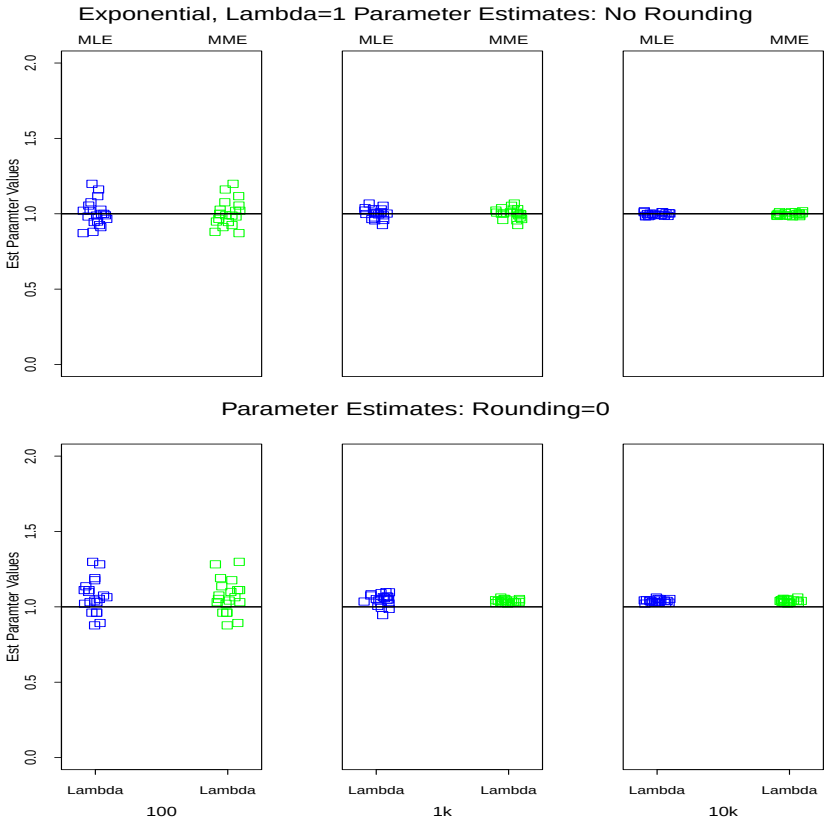


Figure 6.11: Exponential: MLE-MME Parameter Estimation.

Figure 6.11 shows that with a smaller sample size of one hundred, both MLE and MME exhibit greater variance in their estimates. As the sample size increases, the variance decreases, leading to more stable and accurate estimates from both methods irrespective of rounding. Visually, the impact of rounding is small; however, rounded estimates tend to shift above the threshold line, whereas non-rounded estimates are more centered. To support these observations, RMSE results are provided in Table 6.5.

Table 6.5: MLE/MME Parameter Estimates: Exponential RMSE.

Prec	**Type**	**100**	**1k**	**10k**
6	MLE	0.08	0.03	0.01
6	MME	0.08	0.03	0.01
0	MLE	0.13	0.06	0.04
0	MME	0.13	0.06	0.04

Table 6.5 shows that all RMSE values are highlighted in blue, indicating consistency between MLE and MME in parameter estimation. There is no difference in estimation accuracy between the two methods. Rounding the data slightly affects parameter estimation, as seen by the increase in RMSE values for both methods.

6.4.2.3 Log-normal

For the log-normal distribution, parameters were configured to $\mu = 3$ and $\sigma = 1$. Figure 6.12 presents the results of the estimated mean (blue for MLE, green for MME) and standard deviation (brown for MLE, black for MME).

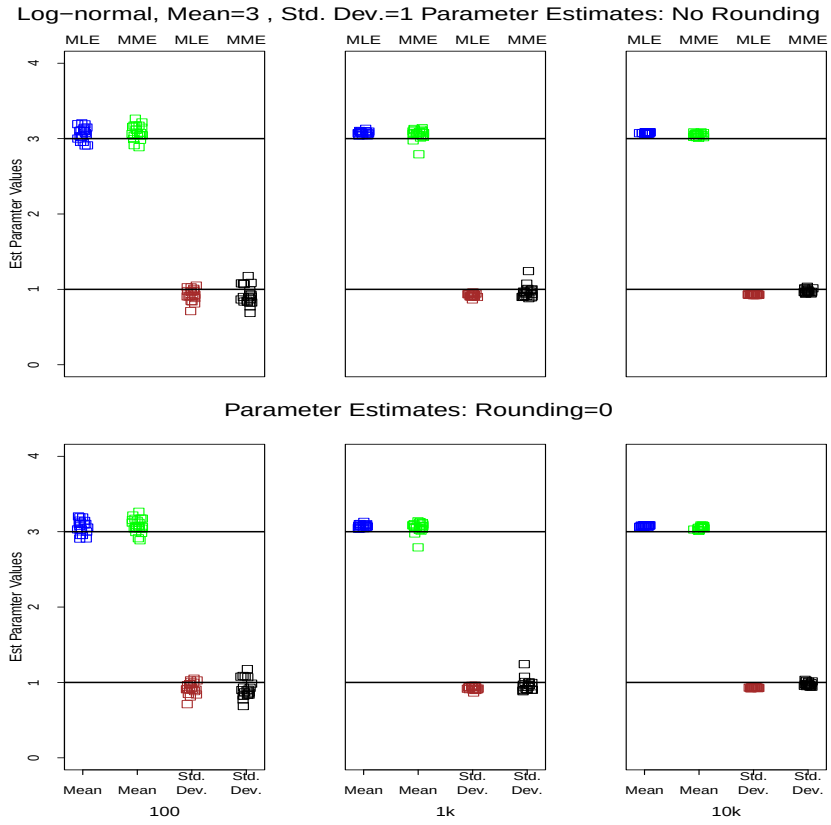


Figure 6.12: Log-normal: MLE-MME Parameter Estimation.

From the charts, it is observed that with $\mu = 3$, both MLE and MME produce estimates centered around true values, indicating strong performance across all sample sizes and rounding conditions. To validate these observations, Table 6.6 presents the corresponding RMSE values.

Table 6.6: MLE/MME Parameter Estimates: Log-normal RMSE.

Prec	**Type**	**100**		**1k**		**10k**	
		Mean	**Standard Deviation**	**Mean**	**Standard Deviation**	**Mean**	**Standard Deviation**
6	MLE	0.11	0.11	0.08	0.08	0.08	0.07
6	MME	0.13	0.14	0.09	0.09	0.05	0.03
0	MLE	0.11	0.11	0.08	0.08	0.08	0.07
0	MME	0.13	0.15	0.09	0.09	0.05	0.03

6.4.3 MLE Versus MME Distribution Fitting GoF Comparison

The results of this study will show multiple jittered scatter plots of the AD and CvM GoF results. Threshold lines incorporated within the charts will indicate AD and CvM cut-off points referencing Tables 2.3 and 2.4.[1]. Additionally, a table will present the average AD and CvM GoF test statistic results.

6.4.3.1 Weibull

The results of the Weibull tests are illustrated in Figures 6.13, 6.14, and Table 6.7. Due to exponential growth in the AD and CvM GoF test statistics, a logarithmic transformation was performed on the GoF test statistics and their threshold values.

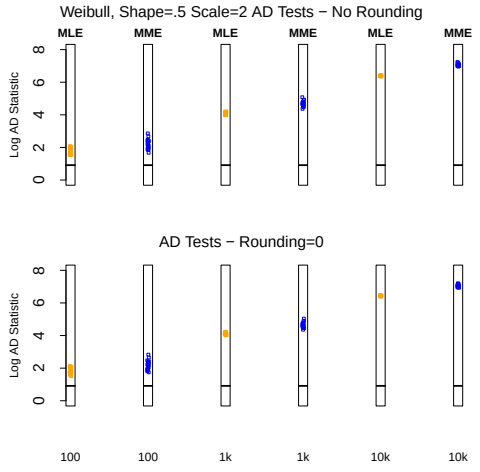


Figure 6.13: 20 Repeat Tests: AD Test Statistics.

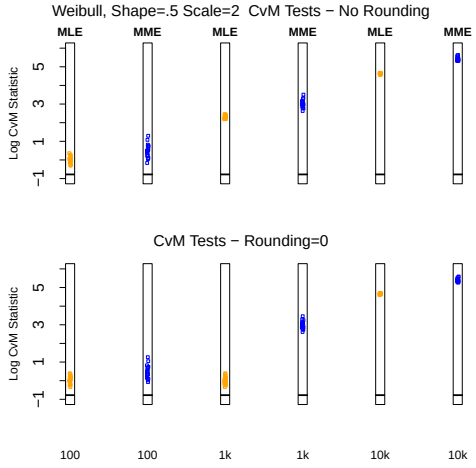


Figure 6.14: 20 Repeat Tests: CvM Test Statistics.

Table 6.7: Weibull GoF Results: Mean Log-Transformed AD and CvM Statistics for MLE and MME.

Weibull: Shape=.5, Scale=2							
	Log(AVG(AD)) Score MLE				**Log(AVG(AD)) Score MME**		
Size	100	1k	10k		100	1k	10k
No Rounding	0.77	1.77	2.77		0.97	2.03	3.07
Rounding	0.79	1.78	2.79		0.96	2.02	3.05

	Log(AVG(CVM)) Score MLE				**Log(AVG(CvM)) Score MME**		
Size	100	1k	10k		100	1k	10k
No Rounding	0.00	1.00	2.01		0.24	1.32	2.36
Rounding	0.01	1.01	2.02		0.23	1.30	2.34

[1] 2.49:AD, 0.46:CvM

It is observed from Figures 6.13 and 6.14, that these tests provide poor parameter estimates for all tests based on all points being above the threshold cut-off point for the GoF test statistics. It is evident that as the sample size increases, the parameter estimates for MLE and MME significantly worsen, as indicated by the iterative shift in the test statistics. There is relatively low variability in MLE's AD and CvM test statistics compared to MME, indicating a more reliable parameter estimate. Referring back to Figure 6.10, both MLE and MME showed significant variance in estimating the scale parameter for all sample sizes, which may have a significant impact on the GoF results. It is also noted from Figures 6.13 and 6.14 that, although the variance was reduced, the minimum estimations shifted upward slightly, which has led to a repeated shift in the GoF test statistic results.

When comparing rounded and non-rounded data, there's no significant difference in results. Table 6.7 confirms that MLE consistently achieves lower average GoF statistics compared to MME.

These poor results may be due to unsuitable starting values for both MLE and MME. When $\beta < 1$, the failure rate extends over a longer time period, creating heavy-tailed data which affects MME, as extreme values distort the mean and variance, making moment-based estimation unreliable. For MLE, as β approaches zero, the PDF becomes sharply peaked, and density values can grow infinitely large at specific points, which can cause numerical instability in the likelihood function, flattening the likelihood and making optimisation difficult. While the optimiser still converges, it often returns poor parameter estimates. As shown in Section 5.4.2.1, β values ≥ 0.040 when $\lambda = 2$ caused no convergence issues, but $\beta < 0.040$ did, which confirms that estimates degrade as parameter values get closer to zero. However, when $\beta > 0.5$, the estimated parameters closely approximate the true values, as demonstrated for $\beta = 0.8$. Higher values were not tested.

In summary, MLE provides better parameter estimates than MME for all sample sizes and tests. Rounding the data does not impact the performance of MLE or MME. MLE provides closer parameter estimates as optimisation happens over the entire likelihood surface, which incorporates the distribution's shape (tails, skewness) because the log-likelihood sums contributions from each

observed data point under the assumed distribution, whereas MME factors the moments.

6.4.3.2 Exponential

When rounding to zero decimal places for the exponential distribution, the AD test consistently returned Inf values for both MLE and MME, rendering the results inconclusive. To understand this issue, refer to Section 5.4.3.2, where it was shown that zero or extreme values in the data lead to divergence in the AD test statistic. Rounding to four decimal places was necessary to avoid these errors; however, tests with zero decimal places were still performed to support CvM analysis.

The results are presented in Figures 6.15 and 6.16, and summarised in Table 6.8.

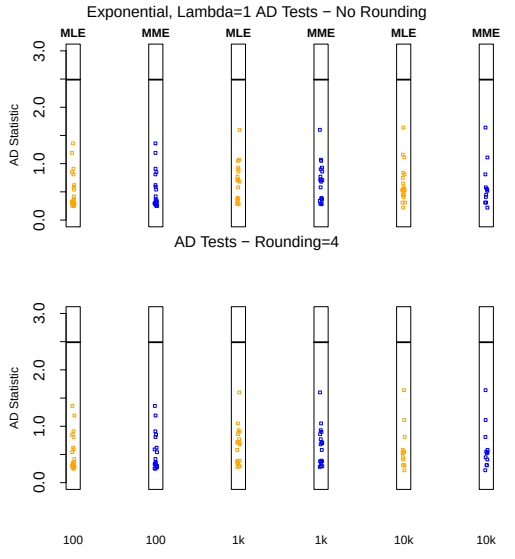


Figure 6.15: Exponential, 20 Repeat Tests: AD Test Statistics.

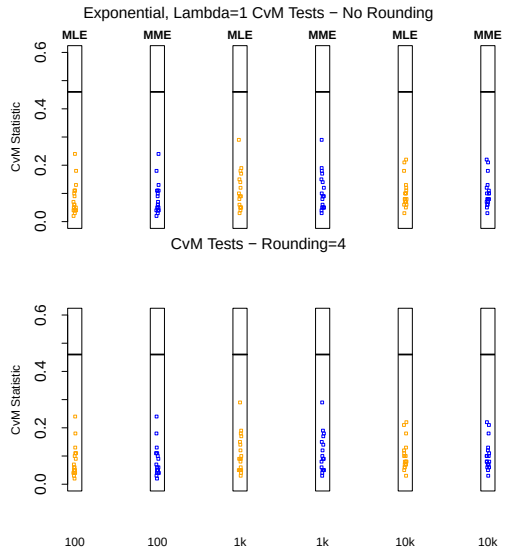


Figure 6.16: Exponential, 20 Repeat Tests: CvM Test Statistics.

Table 6.8: Exponential MLE versus MME GoF Results.

Exponential: Lambda=1							
	AVG MLE AD Score				**AVG MME AD Score**		
Size	100	1k	10k		100	1k	10k
No Rounding	0.52	0.66	0.63		0.52	0.66	0.63
Rounding=0	Inf	Inf	Inf		Inf	Inf	Inf
Rounding=4	0.52	0.64	0.62		0.52	0.64	0.62

	AVG MLE CvM Score				**AVG MME CvM Score**		
Size	100	1k	10k		100	1k	10k
No Rounding	0.08	0.10	0.10		0.08	0.10	0.10
Rounding=0	0.08	0.10	0.10		0.08	0.10	0.10
Rounding=4	0.08	0.10	0.10		0.08	0.10	0.10

For rounding = 0, no constant was added to the data, and the test results returned Inf values. Zero values do not affect the CvM algorithm, therefore, no constant was introduced in this case. The results for rounding = 0 in the CvM test correspond to outcomes without any added constant.

With y-axis limits set to 3 for the AD test and 0.6 for the CvM test, the charts show that MLE and MME perform identically across all GoF tests, with all results meeting the GoF criteria. No method consistently yielded lower test statistics, and rounding to four decimal places has minimal impact across sample sizes. The table confirms these findings through average GoF statistics.

In summary, the comparable performance of MLE and MME in this case is due to the simplicity of the distribution, as it only has a single rate parameter, which reduces sensitivity to outliers. However, rounding introduces constraints by altering the continuity of the data and reducing the precision of the parameter estimates, an effect that was evident in this analysis.

6.4.3.3 Log-normal

For the results of the log-normal distributions for MLE and MME on the AD and CvM GoF tests, Figures 6.17, 6.18, and Table 6.9 show the results.

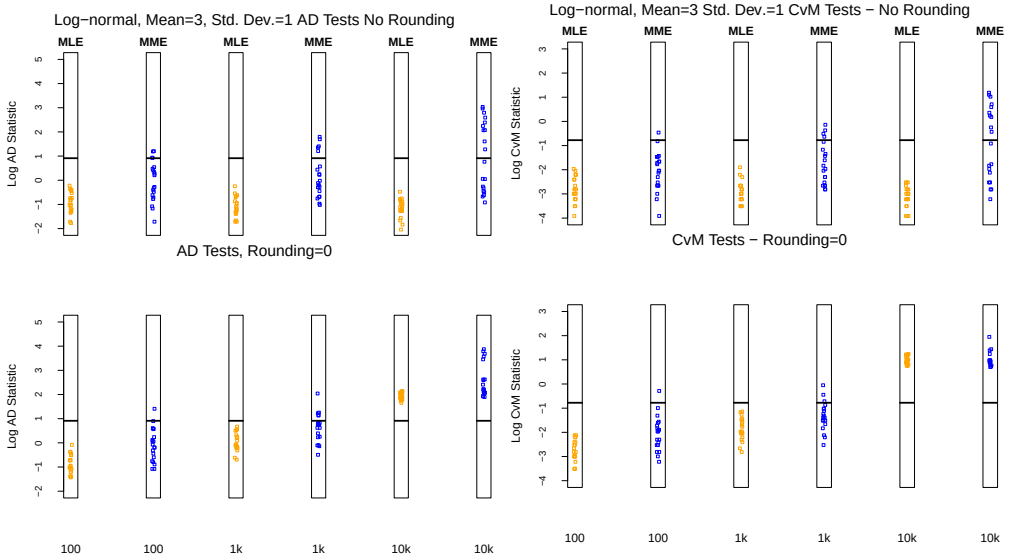


Figure 6.17: 20 Repeat Tests: AD Test Statistics.

Figure 6.18: 20 Repeat Tests: CvM Test Statistics.

Table 6.9: Log-normal AVG MLE versus MME GoF Results.

Log-normal (Mean=3, Standard Deviation=1)							
	AVG MLE AD Score				**AVG MME AD Score**		
Size	100	1k	10k		100	1k	10k
No rounding	0.44	0.36	0.33		1.27	1.78	6.16
Rounding=0	0.45	1.09	6.93		1.14	2.24	16.99

	AVG MLE CvM Score				**AVG MME CvM Score**		
Size	100	1k	10k		100	1k	10k
No rounding	0.06	0.05	0.04		0.16	0.25	0.97
Rounding=0	0.06	0.16	0.98		0.16	0.30	2.07

Figures 6.17 and 6.18 show that MLE has lower variance, with more tightly

clustered points across all GoF tests, indicating greater stability and consistency in parameter estimation. In contrast, MME exhibits greater dispersion and a level shift with increasing sample size, suggesting reduced reliability. Visually, both methods appear to be impacted by rounding. To conclude with the results, Table 6.9 supports these observations and further highlights the effect of rounding. For MLE, rounding has no impact at a sample size of one hundred, but negatively affects both AD and CvM results at larger sample sizes. For MME, rounding slightly improves AD results at the one hundred sample size and has no effect on CvM; however, both tests show a decline in performance as sample size increases.

6.5 Discussion

6.5.1 Parameter Changes: Effects on Shape and Estimation

Regarding the effects on the distribution's shape, variations in distribution parameters modify the shape of the density, influencing kurtosis, the extent of the tails, and dispersion. Increasing the parameters results in a more leptokurtic shape, with sharper peaks and shorter tails, thereby reducing the potential for outliers. Decreasing the parameters, in contrast, leads to more platykurtic shapes, with flatter peaks, longer tails, and increased dispersion. In multi-parameter distributions, changes in one parameter may have effects that are opposite to those of the other parameter, resulting in complementary but distinct influences on the shape.

Varying parameter values influences the accuracy of parameter estimates for MLE and MME in multi-parameter distributions, whereas in single-parameter distributions, there is little impact on estimation accuracy. In more complex two-parameter distributions, altering one parameter can affect both estimators, however, the degree of divergence may differ between parameters and depend on the distribution itself and the parameter being varied, with the effect not being uniform. Some parameters introduce greater instability than others. Visual inspection of the scatter plots suggests that sample size has minimal impact when comparing with parameter variation.

When data are rounded, an effect on parameter estimation is expected because the PDF effectively becomes a PMF, altering the underlying density structure.

The transformation can hinder parameter estimation, regardless of varying parameter values. Furthermore, changing the parameters influences estimation accuracy, as the shape of the density may either concentrate near the distributional bounds or extend beyond regions of acceptable support. Applying rounding under these conditions, while varying parameter values, would therefore be expected to negatively impact estimation performance. However, the results indicate that rounding has little impact on estimation accuracy across the three distributions. There are isolated cases where rounding affects estimation for the exponential distribution when the rate parameter is $\lambda >= 3$. At lower rates, the impact is negligible. Rounding effects are therefore present but limited, primarily depending on the combination of distribution type and parameter values chosen.

6.5.2 A Comparison of MLE Versus MME Parameter Estimation

There is no consistent advantage between MLE and MME across all scenarios. Their effectiveness varies with the distribution, parameter values, and data characteristics. For example, MME is more effective on one parameter distributions as it only has to match the moment. On the other hand, MLE is less effective when parameter values approach zero, where the likelihood surface becomes flatter, and the optimiser struggles to converge precisely. When this occurs, there is a high degree of dispersion in the estimated parameters. The dispersion is not from the skewness of the data, as previously discussed, but rather from numerical instability in the estimation process. On the other hand, MME tends to be less robust in the presence of outliers, as extreme values can distort the sample moments and lead to biased estimates. In such cases, the MLE optimiser is more resilient and produces more reliable parameter estimates than MME.

From a distribution perspective, MME yields more accurate estimates for the Weibull distribution, MLE performs better for the log-normal except at the largest sample size, where MME provides closer parameter estimates to the true parameter values, and both methods show similar accuracy for the exponential distribution. The effects of rounding also depend on the distribution. In some cases, rounding has minimal impact, while in others it can either improve

or degrade estimation accuracy. Such variability highlights that the effect of rounding cannot be generalised across all methods or distributions.

The results indicate that MLE and MME fail in different ways rather than exhibiting equivalent weaknesses. MLE is primarily affected by numerical instability when parameter values approach zero or when the likelihood surface becomes relatively flat, leading to high dispersion in parameter estimates. In contrast, MME is more sensitive to extreme observations because parameter estimation depends directly on sample moments, making it more vulnerable to skewed or outlying data. Across the distributions tested, MLE generally provides more stable estimates for the log-normal distribution, whereas MME performs better for the Weibull distribution. For the exponential distribution, both methods exhibit comparable behaviour. These findings demonstrate that neither method is universally superior, instead, their effectiveness depends on the interaction between the estimation method, distributional characteristics, and the effects of rounding.

In conclusion, the choice between MLE and MME should be carefully considered. Overall, neither method is better; performance is influenced by distribution, sample size, and data characteristics. Furthermore, visual inspections of the charts do not always align with statistical metrics. Therefore, visualisation methods alone are insufficient to determine the most effective estimation method, highlighting the importance of numerical measures, such as RMSE, in assessing the performance of statistical estimation techniques. Table 6.10 has been provided as a summary of the results. The cells in the last two columns for the log-normal distribution are highlighted in blue to emphasise cases where MME outperforms MLE. Without this visual cue, these outcomes could be easily overlooked during quick inspection, as MLE dominates the remaining results.

Table 6.10: RMSE Summary: MLE versus MME (P1 and P2 Estimates).

Sample Size	100		1,000		10,000		100		1,000		10,000	
Parameters	p1	p2	p1	p2	p1	p2	p1	p2	p1	p2	p1	p2
Weibull												
No Rounding	MME	MME	MME	MME	MME	MME	MME	MME	MME	MME	MME	MME
Rounding = 0	MME	MME	MME	MME	MME	MME	MME	MME	MME	MME	MME	MME
Exponential												
No Rounding	Both	Both	Both	Both	Both	Both	Both	Both	Both	Both	Both	Both
Rounding = 4	Both	Both	Both	Both	Both	Both	Both	Both	Both	Both	Both	Both
Log-normal												
No Rounding	MLE	MLE	MLE	MLE	MLE	MLE	MLE	MLE	MLE	MLE	MME	MME
Rounding = 0	MLE	MLE	MLE	MLE	MLE	MLE	MLE	MLE	MLE	MLE	MME	MME

6.5.3 MLE Versus MME Distribution Fitting GoF Comparison

MLE consistently outperforms MME for the log-normal distribution across all GoF tests. The weaker performance of MME can be attributed to its sensitivity to outliers, which distorts the moment-based parameter estimates. Although $\sigma = 1$ should limit heavy tails, being only one standard deviation from the mean, random extreme values may still degrade performance. MLE, by maximising the likelihood function, yields more robust estimates under these conditions. Rounding constrains both methods by reducing variability in the data and converting the continuous distribution into a discrete mass. Since both methods are intended to fit probability density functions, this distortion limits their ability to produce accurate parameter estimates.

While the table provides a summary of average results, the scatter plots offer deeper insight. Averaging the GoF statistics may obscure meaningful deviations, particularly when large variations occur. Greater emphasis is therefore placed on the scatter plots to support the conclusions. Table 6.11 has been provided to offer a summary of the most optimal parameter estimation methods based on the average GoF test statistics.

Table 6.11: Summary: MLE Versus MME AD Performance.

	AD			CvM		
Size	100	1,000	10,000	100	1,000	10,000
Weibull						
No Rounding	MLE	MLE	MLE	MLE	MLE	MLE
Rounding=0	MLE	MLE	MLE	MLE	MLE	MLE
Exponential						
No Rounding	Both	Both	Both	Both	Both	Both
Rounding=4	Both	Both	Both	Both	Both	Both
Log-normal						
No Rounding	MLE	MLE	MLE	MLE	MLE	MLE
Rounding=0	MLE	MLE	MLE	MLE	MLE	MLE

The instability of GoF tests observed in Chapter 5 is partly explained by the parameter estimation errors demonstrated here.

6.5.4 MLE and MME limitations

When assessing the performance of MLE and MME, it is important to consider how to evaluate the results, whether it be based on AD, CvM, or RMSE, as each captures different aspects of model fit and may lead to different conclusions.

Before interpreting the results, it is also important to recall the key properties of a good estimator:

- Consistency: Estimates converge to the true parameter value as sample size increases.
- Efficiency: Has the minimum MSE.
- Bias: The expected value equals the true parameter value.

The extent to which MLE and MME support these properties varies depending on the data characteristics and distributional assumptions.

For simple, single-parameter distributions, both estimators performed well, with no significant limitations observed. However, in multi-parameter distributions, MME often struggled, particularly at small sample sizes, due to its reliance on sample moments, which are sensitive to variance. MLE was more robust, though not always better, as seen in some Weibull cases.

Rounding presented further challenges. Rounded data behaves more like a PMF, yet both methods assume a continuous PDF, such that a mismatch distorts parameter estimates in multi-parameter distributions. However, it had a negligible impact in single-parameter cases.

Both estimators also exhibited degraded performance near boundary parameter values (e.g., near zero), where MLE often faced optimisation issues, and MME returned poor estimates due to inflated or zero variance.

Notably, increasing sample size did not always improve accuracy, contradicting the expectation of estimator consistency. In several cases, larger samples led to more biased estimates.

Changing distribution parameters (e.g., increasing scale or standard deviation) also affected performance. MME proved more sensitive to dispersion and rounding, while MLE sometimes became numerically unstable, providing poor estimates.

Finally, RMSE revealed discrepancies that were less apparent with AD or CvM tests. While MLE often yielded lower test statistics under AD and CvM tests, RMSE occasionally identified MME as more accurate.

6.6 Conclusion

Changing parameter values affects both MLE and MME. The degree of impact varies depending on the distribution and evaluation metric used.

MLE provides a better fit for multi-parameter distributions in the AD and CvM GoF tests. For single-parameter distributions, MLE and MME yield comparable and consistent parameter estimates.

Rounding has a negligible effect on single-parameter distributions but negatively impacts estimation accuracy in multi-parameter distributions.

RMSE directly assesses the accuracy of parameter estimates, providing a complementary measure to GoF metrics that focuses on distributional shape and tail behaviour. RMSE provides more precise insights into parameter estimation accuracy, revealing limitations that are not apparent in AD or CvM tests. Specifically, RMSE shows that MME outperforms MLE for the Weibull distribution, however, using the AD and CvM tests, MLE emerges as

the winner. Also, RMSE implies that MLE provides more optimal parameter estimates, except at the ten thousand sample size, whereas MME is more optimal, highlighting the influence of sample size on performance, especially when using RMSE as an evaluation metric.

This chapter demonstrates that quantisation introduces systematic distortion into parameter estimation for common probability distributions. The results show that both MLE and MME are sensitive to rounding effects. MLE generally provides more stable and accurate parameter estimates across varying sample sizes and distributional settings. Furthermore, the findings highlight that rounding not only affects parameter recovery, but also influences the reliability of AD and CvM GoF statistics.

Table 6.12 shows a summary of the results.

Table 6.12: Summary of MLE/MME: Conclusions.

Varying the Parameter Results			
Test	Peak of density	Tails	Dispersion
Weibull			
As shape increases	Higher	Shorter	Decreases
As shape decreases	Smaller	Longer	Increases
As scale increases	Smaller	Longer	Increases
As Scales Decrease	Higher	Shorter	Decreases
Exponential			
As rate increases	Higher	Shorter	Decreases
As rate decreases	Smaller	Longer	Increases
Log-normal			
As mean increases	Smaller	Longer	Increases
As mean decreases	Higher	Shorter	Decreases
As standard deviation increases	Smaller	Longer	Increases
As standard deviation decreases	Higher	Shorter	Decreases
Results			
Weibull			
Who wins = AD:MLE CvM:MLE RMSE:MME			
MLE and MME are consistent and similar in estimation = No			
Impacted by rounding = No			
Impacted by sample size = No			
Reducing the shape parameter hinders the scale parameter estimates for both methods			
Increasing the scale parameter hinders the scale parameter estimates for both methods			
Other Issues: MLE and MME tend to overestimate the scale parameter estimates, regardless of rounding			
Exponential			
Who wins = AD:Both CvM:Both RMSE:Both			
MLE and MME are consistent and similar in estimation = Yes			
Impacted by rounding = No			
Impacted by sample size = No			
As the rate increases, more variation occurs in the estimates			
Other Issues:N/A			
Log-normal			
Who wins = AD:MLE CvM:MLE RMSE:Generally MLE, but on largest sample size it was MME			
MLE and MME are consistent and similar in estimation = No			
Impacted by rounding = Yes			
Impacted by sample size = Yes			
Increasing the standard deviation parameter does increase the variance in the return estimates			
As the rate increases more variation occurs in the return estimates for both methods			
Other Issues:As the sample size increases the parameter estimates get worse for MME for AD and CvM tests.			

CHAPTER 7

Challenges Fitting To Rounded Data

Quantisation maps a continuous range of values onto discrete intervals. Rounding adjusts a value to the nearest discrete value, whereas truncation removes digits beyond a certain precision without adjustment. When a dataset is modified by rounding or truncation, the data may become altered enough to no longer fit its original distribution due to quantisation. Fitting statistical distributions under these conditions can result in fitting errors, for instance, when values are rounded to zero. The study explores the effects of quantisation on distribution fitting. It introduces a framework to identify the impacts of quantisation, together with techniques to adjust quantised data while mitigating fitting errors and improving GoF statistics. The focus of this study is on the log-normal distribution, where it is demonstrated that adding a relative constant to zero values can mitigate fitting errors and reduce the influence of quantisation.

7.1 Introduction

Measuring random variables over time is useful in data modelling, but timestamps stored with partial precision through rounding or truncation cause quantisation, which alters distributions and their moments [10] as previously seen in Chapter 5. Zero values, resulting from rounding are often discarded, which may be reasonable in domains such as medicine when they fall below

detection limits. However, when zeros contain meaningful information, their removal risks introducing bias.

Rounding values outside a distribution's support (e.g., to zero) can cause convergence problems during distribution fitting. Berry [15] proposed adding a constant before logarithmic transformation, chosen through residual analysis to approximate normality. The influence of adding a constant is scale-dependent. Adding a constant of 1 becomes negligible for observations in the upper range of the distribution, but adding a constant of 1 to timestamps of $0.001s$ might not be so insignificant. Subsequent studies modelled the constant as an additional parameter, systematically testing values between 0.00001 and $20,000$ with 1716 giving the best fit for the observed data [16]. D'Agostino and Stephens [17] instead suggested adding a small arbitrary constant. An alternative approach is the use of hurdle models, which explicitly separate the probability mass at zero from the continuous component of the distribution, thereby addressing excess zeros without distorting the remaining data.

Building on this context, this study develops a framework for identifying quantisation and evaluates techniques designed to reduce its effects. Four key research questions are addressed:

1. How to identify quantisation?
2. How to mitigate fitting errors due to zero values?
3. Which data adjustment techniques most effectively reduce the impact of quantisation?
4. At what sample size and level of decimal precision does quantisation become more pronounced? For example, is there a point at which statistical tests shift from consistently passing to failing as precision decreases?

Several techniques for addressing quantisation are evaluated, including only adding a constant to zero values, adding a constant to all observations, and applying relative constants scaled to the precision of the data. These tests are defined in Table 7.1 and support research questions number 2 and 3. These

approaches are compared to assess their effectiveness in mitigating fitting errors without distorting the underlying distribution.

Table 7.1: Quantisation Adjustment Methods: Mitigate Fitting Errors.

Test Number	Method	Description
1	Apply a fixed constant(1) to zero values	Add a fixed constant to observations equal to zero.
2	Apply a small constant(0.1) to zero values	Add a fixed small constant to observations equal to zero.
3	Apply a relative constant to zero values	Add a constant proportional to decimal precision to only zero values.
4	Apply a relative constant to all values	Add a constant proportional to decimal precision to all observations.

Notes: Test Number:1 is used as a baseline test. Test Numbers 2:4 are repeated five times to support random variation. Sample sizes range from 100 to 1,000,000. Data are rounded from 0 to 7 decimal places.

The analysis focuses on the log-normal distribution, which cannot accommodate zero values under logarithmic transformation, leading to fitting errors. GoF will be assessed with AD and CvM tests, and parameters estimated via MLE. However, more focus will be given to the AD GoF test. Perhaps surprisingly, to the best of current knowledge, no prior work has proposed relative constants tied to data precision. The constant introduced here mitigates fitting issues by slightly shifting data points.

7.1.1 Data Overview and Limitations

Random samples from a log-normal distribution are generated with sizes $N = 10^n$, extending up to one million observations. Default distribution parameters and general sample sizes are the same as Chapter 5 provided in Table 5.1. Data are rounded to m decimal places ($0 \leq m \leq 7$).

The limitations of creating synthetic data are previously discussed in Section 5.2.

7.2 Methods

In this study, the standard published AD and CvM critical values were not used directly[1] instead, custom critical values were derived from the AD [84] and CvM [90] formulas, to better reflect the structure and characteristics of the data under study. The custom thresholds are summarised in Table 7.2 with a cut-off value of 3.01 for AD and 0.76 for CvM. The thresholds determine whether a dataset can be considered as fitting with the assumed distribution.

Table 7.2: AD/CvM Cut Off Points.

AD Score	CvM Score	Colour	Term
<2.52	<0.47	Light Blue	Pass
2.52 : 3.02	0.47 : 0.56	Dark Blue	Borderline Fail
3.03 : 3.54	0.57 : 0.65	Light Grey	Fail
>3.54	>0.65	Dark Grey	Serious Fail

As defined in Table 7.2, results are classified into four categories: "Pass", "Borderline Fail", "Fail", and "Serious Fail". The "Pass" threshold corresponds to the 5% critical value of the AD and CvM statistics, estimated from a reference distribution of ten thousand observations resampled five thousand times. The remaining categories are set at one, two and three standard deviations above the reference value. Table 7.2 also uses colours to illustrate the strength of each test outcome, in the tables later in this chapter. A "Borderline Fail" is defined as a result close to the initial cut-off threshold, with progressively larger deviations classified as more severe failures.

The methods proposed in this chapter are heuristic adjustments designed to mitigate the effects of quantisation. They do not recover the true underlying continuous data

7.2.1 How To identify Quantisation

Quantisation can be identified by analysing whether the data exhibits uniform or non-uniform step patterns. When plotted, uniform quantisation appears as equally spaced steps, while non-uniform quantisation produces irregular

[1] When GoF tests are applied with fitted parameters rather than fixed values (results not shown), they generally indicate a better fit. For the CvM statistic in particular, lower test scores are obtained with fixed parameters, whereas higher scores are observed when parameters are estimated.

spacing. Diagnostic tools, such as histograms, density plots, Q–Q and P–P plots, are commonly used in distribution fitting. Evidence of quantisation in these plots includes vertically stacked points, step-like structures, or isolated points at large intervals.

Kernel smoothing, often applied in density estimation, can obscure these quantisation patterns, where rounded data may appear as multiple similar peaks rather than discrete steps (e.g. see examples in Chapter 5 Figure 4.10). Careful visual inspection of these graphs provides additional evidence when quantisation is suspected, and helps explain unexpected behaviour during distribution fitting. For this reason, inspection of histograms, Q–Q and P–P plots is used to confirm the presence of quantisation.

An easy way to detect quantisation is to examine the spacing between consecutive data values. If the differences form multiples of a common small number, this provides evidence of quantisation. Future work could further develop this method as a complementary tool for detecting quantisation.

7.2.2 How to Address Quantisation

Four methods for mitigating the effects of quantisation on distribution fitting, outlined in Table 7.1, form the basis of the evaluation. For these tests, the data are rounded from 0 to 7 decimal places, and each test is repeated five times to account for random variation. GoF is evaluated using the AD and CvM tests.

7.2.2.1 Zero Value Problem

When fitting data to a log-normal distribution, failures arise if zero values are present, as discussed in Chapter 5. The log-normal distribution is defined only for strictly positive values, and the logarithm of zero values is undefined, leading to fitting errors. To understand the likelihood of such occurrences, the expected number of zero values can be estimated from the probability $P(X \leq 0.5)$ when $m = 0$, since any value less than or equal to 0.5 rounds to zero at zero decimal places. Using the CDF of the log-normal distribution, this probability is approximately 0.0001. For small sample sizes, the chance of observing zero values is negligible. However, for larger sample sizes, zero values are expected to appear.

Two common techniques can be used to address this problem: removing zero values, which can be modelled using a Hurdle distribution, or shifting the data by adding a constant. The latter approach does not directly resolve quantisation but enables a successful log-transformation required for fitting to the theoretical distribution. In this study, a constant value of 1 is applied to the data [15, 17], chosen for simplicity and consistency across experiments, providing a baseline against the effectiveness of alternative approaches while ensuring the feasibility of logarithmic calculations. In addition, when analysing heavy-tailed datasets in the millisecond time range, observations in the first bin may suffer from under- or over-fitting to the distribution, as shown previously in Figure 3.13. The following subsections address this issue while exploring methods to mitigate fitting errors.

7.2.2.2 Apply a Small Constant to Zero Values

In addition to the baseline shift of 1, the effect of applying a smaller constant of 0.1 to only zero values is studied. The motivation is to evaluate whether a reduced shift is sufficient to improve the effects of quantisation, and determine whether using a smaller constant improves GoF as measured by the AD and CvM statistics.

The effect of this adjustment is to shift observations lying on a bin boundary into the next closest bucket. For example, a value of 0.9 is shifted to 1.0, whereas a value of 0.8 remains in the 0 bucket. The procedure is applied only when the small constant is defined relative to the bin width. In such cases, it may slightly smooth the data, improving the distributional fit and partially offsetting the effects of quantisation.

7.2.2.3 Apply a Relative Constant to Zero Values

Continuing on from applying a small constant of 0.1, this section studies applying a relative constant to zero values. Values are chosen based on the precision of the rounding. Where rounding, when $m = 0$, a constant of 0.1 is added to each zero value, and for all other roundings, the formula is $c = 0.1^m$ where ^{m} is the power based on the level of rounding. A rounding to three will have a relative constant of 0.001.

Unlike fixed constants such as 1 or 0.1, which may be disproportionately large

when applied to finely rounded data, this technique preserves the relative scale of the distribution while still enabling logarithmic transformations. It provides a consistent method across different rounding levels, avoiding excessive distortion of the underlying data structure.

7.2.2.4 Apply Relative Constant to all Values

Applying a relative constant to all values adapts the previous section by applying a relative constant to every observation in the dataset, rather than only to values in the zero range. The adjustment introduces a uniform shift across all values, which can reduce the impact of quantisation and improve distributional fit. Although the effect is additive rather than multiplicative, the resulting displacement may smooth the data sufficiently to counteract distortions introduced by rounding.

7.2.2.5 Trade-offs in Data Adjustment

The proposed adjustment methods improve optimisation stability and reduce convergence failures caused by quantisation and zero-valued observations. In particular, adding constants to the data enables likelihood-based fitting procedures to operate when logarithmic transformations or support constraints would otherwise fail. Relative constant methods further improve GoF performance by reducing the concentration of repeated values at zero.

However, these improvements introduce a trade-off between numerical stability and statistical bias. Any modification to the observed data alters the empirical distribution and therefore changes the relationship between the fitted model and the original quantised observations. Adding a fixed constant shifts the entire distribution upward, potentially biasing parameter estimates and affecting tail behaviour. Applying constants only to zero values reduces this distortion but still modifies the lower boundary of the distribution. Similarly, relative constants preserve more of the original structure but continue to alter the empirical density near zero.

These findings demonstrate that adjustment methods should not be interpreted as recovering the true underlying unquantised data. Instead, they provide practical approximations that improve fitting stability and GoF behaviour while introducing controlled levels of distortion. The results therefore high-

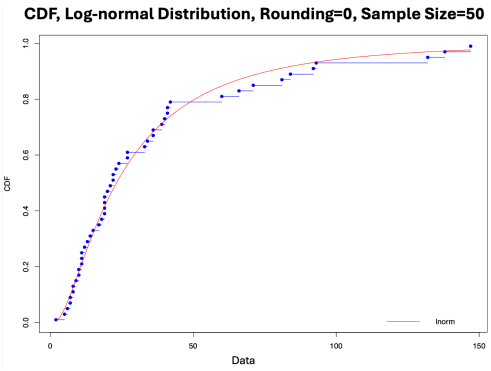
light an inherent trade-off: methods that improve optimisation stability and reduce variance may simultaneously introduce systematic bias into parameter estimation.

The choice of adjustment constant is therefore not arbitrary. Small constants minimise distortion but may fail to resolve convergence instability, whereas larger constants improve numerical robustness at the cost of increased bias. The results suggest that effective adjustment depends on balancing stability against preservation of the original distributional characteristics, with the optimal choice varying across distributions and sample sizes.

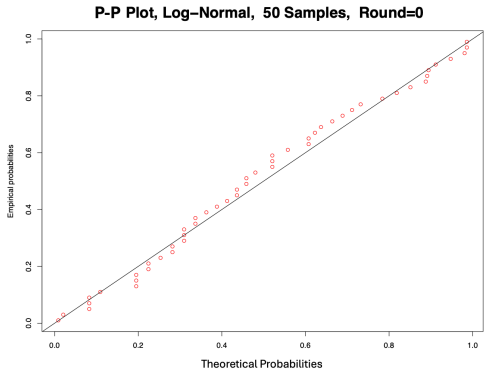
7.3 Results

7.3.1 How to Identify Quantisation

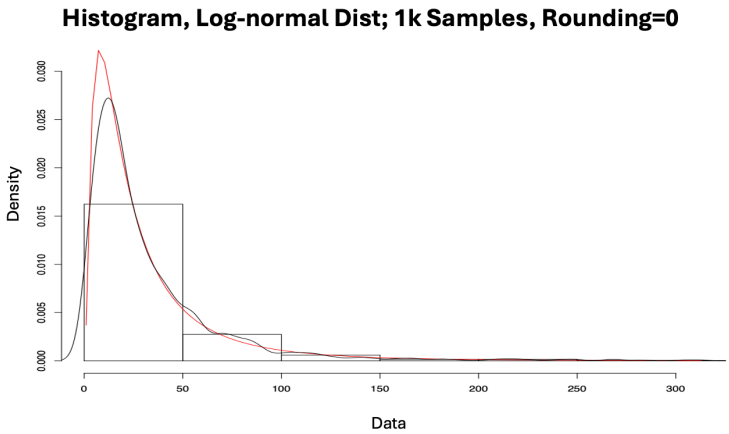
Figure 7.1 presents three diagnostic plots used to identify quantisation in log-normal data rounded to $m = 0$ decimal places.



(a) CDF.



(b) P–P Plot.



(c) Histogram.

Figure 7.1: Log-normal Distribution: Quantisation Detection
(a) CDF, (b) P–P Plot, (c) Histogram.

Panel (a) shows the CDF, where several observations are vertically stacked between 0.6 and 0.8 on the y-axis, indicating quantisation. Widely spaced points above 0.8 reflect the distribution's tail. It should be noted that in log-normal distributions, larger datasets reduce the visibility of quantisation in CDF plots, whereas in other distributions, such as the normal distribution, these effects may be more apparent.

Evidence of quantisation in Q–Q and P–P plots can be identified by stacked points along the distribution line, as rounding forces multiple observations to take identical values. In Q–Q plots, these effects are often difficult to detect because duplicates are spread across the quantile scale, and stacking becomes increasingly obscured as the sample size grows. In contrast, P–P plots amplify the effect by mapping cumulative probabilities, which makes overlapping values

easier to identify. For this reason, Q–Q plots are not presented in this section. Figure 7.1 in panel (b) clearly shows evidence of quantisation. With rounding to zero decimal places, only seventeen unique values remain from fifty observations, producing distinct horizontal bands rather than a continuous diagonal.

Panel (c) shows a histogram overlaid with the empirical density (black) and theoretical density (red). Multiple kernels or peaks in the smoothed density would suggest quantisation, as they reflect repeated kernels at rounded values. However, only a single peak is observed here, and quantisation is therefore not evident. Overall, histograms of log-normal data are less effective for detecting quantisation compared to P–P plots. An example of a histogram with discrete data is in Chapter 3, Figure 3.4, where there is white space between the bars on the histogram.

7.3.2 How To Address Quantisation

To apply the GoF tests, the distribution must first be fitted. Mitigation methods in Section 7.2.2 are employed as an initial step to facilitate fitting.

7.3.2.1 Zero Value Problem

Quantisation can introduce fitting errors as previously discussed in Chapter 5 when zero values are present. To address this, a constant of 1 was first considered and added to the data to enable logarithmic calculations required for parameter estimation and GoF testing. While this adjustment does not reduce quantisation, it provides a practical solution to mitigate fitting failures. Fitting errors were observed when rounding to $m = 0$ decimal places. The approach eliminated fitting errors by shifting all values away from zero and thereby enabling distribution fitting. These results are consistent with the convergence behaviour and error patterns reported in Chapter 5, where zeros, near-zero values, and support violations were identified as primary drivers of fitting failure across Weibull, log-normal, and exponential distributions.

Adding a constant shifts the entire distribution away from zero, reducing the concentration of probability mass at the lower boundary and preventing undefined logarithmic operations during fitting. This improves optimisation stability and enables convergence of the likelihood function. However, the adjustment also introduces systematic upward bias by altering the scale and

relative spacing of the original observations, particularly for heavily quantised datasets where many values are concentrated near zero.

Additional tests with smaller constants $(0.01, 0.001, 0.0001, 0.00001)$ did not produce any fitting problems. However, the minimum constant required to overcome fitting problems for the Weibull distribution was a constant of 1. The Weibull distribution is defined at zero, with its behaviour determined directly by the shape parameter. Shifting the data away from zero is inconsistent with its model assumptions near zero, causing the fitting process fail to converge or yield unreliable results. In one test, the optimiser failed completely when initialised at shape $= 0.1$ and scale $= 0.1$.

The constants were selected to evaluate the sensitivity of the fitting process to progressively smaller perturbations near zero. Larger constants improve optimisation stability by moving observations further from the support boundary, whereas smaller constants preserve more of the original distributional structure. The results indicate that the effectiveness of the adjustment is sensitive to both the magnitude of the constant and the underlying distribution. In particular, the Weibull distribution required a substantially larger adjustment to achieve stable convergence.

As described in Section 7.2.2.1, a constant of 1 was added to the data and used as a baseline for GoF evaluation when zero values were present. After applying the constant, GoF results were assessed using the heatmap in Table 7.3, which covers roundings of up to seven decimal places and all sample sizes. Extending the analysis beyond seven decimal places did not yield any improvement. At larger sample sizes, zero values continued to occur, and the GoF tests failed regardless of the additional precision. Thus, rounding to more than seven decimal places was unnecessary, as sufficient information was already captured at lower levels of rounding.

Table 7.3: Log-normal Distribution: AD GoF Test Results (Mean=3, STD=1, Constant=1.

		Decimal Rounding							
Sample Size	**Repeat**	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**
100	0	2.43	1.52	0.79	1.37	0.69	0.59	0.72	1.89
100	1	2.72	1.08	0.78	0.44	0.21	0.51	0.34	0.34
100	2	0.28	0.32	0.37	0.51	0.29	1.01	3.00	0.48
100	3	1.10	1.83	1.49	1.04	0.58	0.51	0.99	1.04
100	4	1.27	0.92	1.32	0.79	0.73	0.38	1.21	1.41
100	5	0.80	0.88	1.91	3.84	0.53	1.33	1.18	1.24
1000	0	8.00	3.46	1.57	5.81	4.24	2.95	3.71	5.35
1000	1	4.17	9.14	1.82	2.21	1.09	3.53	7.91	3.87
1000	2	3.56	2.47	2.78	3.61	1.73	1.85	2.96	3.19
1000	3	5.89	4.98	6.98	3.30	2.89	2.97	6.30	6.01
1000	4	4.66	5.17	1.69	2.72	2.17	6.99	5.87	2.81
1000	5	6.37	3.48	4.93	2.22	4.83	5.15	9.46	3.71
10000	0	44.26	35.05	37.00	31.20	22.83	19.89	29.94	31.34
10000	1	40.38	33.15	27.64	18.84	24.69	26.87	27.59	31.42
10000	2	46.72	36.36	33.42	28.84	22.86	35.70	41.13	27.29
10000	3	29.47	37.70	36.42	28.69	30.56	36.20	33.12	33.94
10000	4	34.38	26.53	37.16	42.79	33.47	38.72	38.81	47.97
10000	5	33.91	31.19	38.13	35.76	40.34	26.36	21.45	40.01
100000	0	327.42	327.48	324.73	337.34	304.52	324.71	317.99	282.88
100000	1	328.81	314.20	297.41	338.79	312.98	298.57	317.52	300.12
100000	2	333.84	339.86	315.61	348.45	312.05	367.69	343.01	306.80
100000	3	332.90	311.47	311.19	292.81	297.35	346.46	317.44	333.59
100000	4	373.70	339.28	332.48	355.98	315.66	352.59	314.54	313.92
100000	5	373.12	334.04	342.05	353.66	327.72	342.50	349.12	332.12
1000000	0	3425.33	3290.76	3229.87	3203.76	3099.68	3246.40	3235.82	3235.82
1000000	1	3469.17	3238.98	3258.68	3288.20	3353.25	3095.15	3329.97	3177.53
1000000	2	3508.98	3248.81	3215.26	3224.42	3240.57	3204.47	3154.42	3133.09
1000000	3	3501.25	3204.89	3319.26	3342.58	3276.14	3273.30	3254.65	3234.54
1000000	4	3488.21	3228.83	3386.96	3342.49	3196.87	3256.42	3214.04	3329.09
1000000	5	3467.42	3183.61	3228.10	3182.41	3314.93	3260.19	3156.83	3147.72

Note: Legend: Light Blue: Pass, Dark Blue: Borderline Fail, Light Grey: Fail, Dark Grey: Serious Fail.

The repeat tests are shown in this table to showcase the variability of the GoF statistical results across the different rounding tests. Subsequent tables only show the average GoF results because the purpose shifts from examining the variability in the results to comparing overall performance between different rounding scenarios. Showing averaged results avoids unnecessary repetition while still capturing the general behaviour of each method.

Table 7.3 is colour coded based on the strength of the fit, and the colours were previously described in Table 7.2. As a reminder, light blue indicates a satisfactory fit to the distribution, typically observed for smaller sample sizes regardless of quantisation. Dark blue denotes borderline failures, occurring inconsistently at the one thousand sample size. Light and dark grey shades represent progressively weaker fits. At larger sample sizes, all tests resulted in a "Serious Fail", irrespective of quantisation.

For the CvM tests, Table 7.4 shows the results. The results show inconsistent passes up to the one thousand sample size. At ten thousand observations, all tests moderately failed. Collectively, of the 232 repeat tests, only 65 passed the CvM GoF threshold, leading to an overall 28% pass rate.

Table 7.4: Log-normal Distribution: CvM GoF Test Results (Mean=3, STD=1, Constant=1)

		Decimal Rounding							
Sample Size	**Repeat**	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**
100	0	0.46	0.30	0.11	0.21	0.09	0.08	0.11	0.32
100	1	0.50	0.22	0.10	0.08	0.34	0.02	0.07	0.04
100	2	0.03	0.04	0.03	0.07	0.05	0.16	0.49	0.07
100	3	0.14	0.32	0.22	0.18	0.10	0.08	0.17	0.11
100	4	0.18	0.19	0.25	0.09	0.08	0.06	0.24	0.17
100	5	0.10	0.14	0.36	0.75	0.05	0.24	0.19	0.24
1000	0	1.47	0.52	0.15	0.92	0.57	0.30	0.49	0.92
1000	1	0.59	1.45	0.20	0.32	0.11	0.45	1.29	0.59
1000	2	0.53	0.27	0.25	0.48	0.16	0.20	0.36	0.33
1000	3	0.94	0.63	1.19	0.44	0.35	0.32	0.94	1.05
1000	4	0.60	0.82	0.14	0.45	0.23	1.03	0.80	0.35
1000	5	0.86	0.51	0.83	0.23	0.84	0.79	1.49	0.49
10000	0	6.62	5.01	5.38	4.59	2.44	2.10	4.43	4.35
10000	1	5.84	4.58	3.62	2.25	3.09	3.25	3.47	4.11
10000	2	7.12	5.08	4.90	3.35	3.05	4.66	6.39	3.43
10000	3	4.06	5.86	5.47	3.96	4.14	5.32	4.49	4.78
10000	4	5.03	3.04	5.59	6.10	4.40	5.42	5.54	6.82
10000	5	4.26	4.29	5.53	4.98	6.22	3.47	2.36	5.63
100000	0	44.48	45.26	44.26	46.00	41.18	47.37	43.83	37.98
100000	1	45.67	42.38	39.01	46.71	41.94	38.58	44.15	39.81
100000	2	45.97	46.30	43.27	48.20	42.16	50.68	48.86	41.13
100000	3	44.60	42.56	42.39	39.97	38.92	49.05	43.46	45.11
100000	4	53.88	46.99	46.53	50.62	42.92	49.15	42.24	41.85
100000	5	52.33	45.61	46.35	49.21	45.32	47.47	48.77	46.00
1000000	0	470.56	450.23	441.54	452.97	435.28	419.18	441.75	443.19
1000000	1	479.82	440.68	447.54	448.85	461.83	417.31	461.10	427.54
1000000	2	489.52	442.20	435.79	437.82	444.70	433.24	427.29	423.80
1000000	3	485.06	432.93	456.34	461.50	446.14	449.26	440.39	442.26
1000000	4	484.90	447.45	471.44	453.00	433.10	443.35	419.12	458.70
1000000	5	481.07	435.77	441.45	430.57	457.67	444.12	425.96	420.29

Note: Legend: Light Blue: Pass, Dark Blue: Borderline Fail, Light Grey: Fail, Dark Grey: Serious Fail.

The baseline enables comparison of the proposed methods in the following sections.

7.3.2.2 Apply a Small Constant to Zero Values

A constant of 0.1 is added to observations of just the zero values. Previous research indicates that most data are concentrated in the lower range near zero. Table 7.5 presents the results. For the AD GoF test, a significant improvement is observed across all sample sizes except at one hundred. Importantly, the test now passes up to a sample size of one thousand. For the CvM GoF test,

substantial improvements are evident across all sample sizes, with datasets up to one thousand now passing the test. Applying a constant of 0.1 to zero values (i.e., datasets rounded to zero decimal places) substantially improves the distributional fit.

Unlike adding a constant to the full dataset, this approach modifies only observations equal to zero, thereby preserving most of the original distributional structure. The method reduces the distortion introduced by the baseline adjustment while still mitigating convergence failures caused by zero values. However, because the adjustment selectively alters lower-bound observations, it may still bias tail behaviour and affect GoF statistics for large sample sizes.

Table 7.5: Log-normal Comparision: Basline V Apply Small Constant to Zero Values, Rounding=0.

	Average AD Score			**Average CVM Score**	
Size	Baseline	Test		Baseline	Test
100	1.43	1.01		0.24	0.16
1000	5.44	1.11		0.83	0.18
10000	38.19	4.09		5.49	0.56
100000	344.97	31.26		47.82	3.85
1000000	3476.73	304.12		481.82	37.04

7.3.2.3 Apply Relative Constant to Zero Values

Building on the findings presented in Section 7.3.2.2, this analysis applies a relative constant to zero values with the aim of improving upon the results in Table 7.3. For datasets rounded to zero decimal places, no further improvement is anticipated, as this case already incorporates the same constant, as reported in Table 7.5. Table 7.6 shows the results of all decimal precisions and not just at zero decimal places (as seen in Table 7.5).

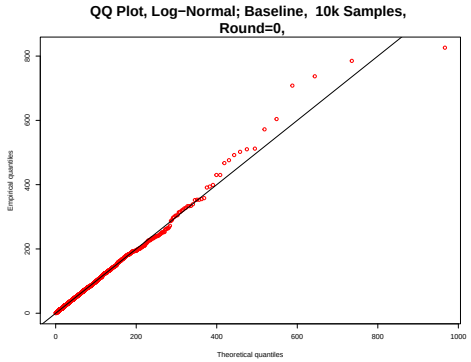
The relative constant approach adapts the adjustment magnitude to the scale of the data, reducing the degree of artificial distortion introduced by fixed constants. This improves the balance between numerical stability and preservation of the original distributional shape. Nevertheless, the method still modifies the empirical distribution near zero and therefore cannot fully recover the underlying unquantised data.

Table 7.6: Log-normal AD Comparison: Baseline v Relative Constant to Zero Values.

	Rounding							
	Baseline: Add Constant of 1 to Zero Values							
Size	0	1	2	3	4	5	6	7
100	1.43	1.09	1.11	1.33	0.82	0.67	1.27	1.07
1000	5.44	4.78	3.30	3.31	2.83	3.91	6.04	4.16
10000	38.19	33.33	34.96	30.94	29.13	30.62	32.01	35.33
100000	344.97	327.72	320.58	337.83	311.71	340.09	326.60	311.57
1000000	3476.73	3241.65	3273.02	3275.30	3264.25	3198.17	3211.02	3209.63
	Add Relative Constant to Zero Values							
100	1.02	0.78	0.66	1.21	0.58	0.75	0.92	0.90
1000	1.29	1.01	0.88	1.47	0.71	0.96	0.79	1.60
10000	4.10	1.02	0.60	1.96	1.08	1.58	1.36	1.52
100000	31.40	1.41	0.94	1.17	1.27	0.82	1.01	1.25
1000000	305.57	4.23	1.03	0.80	1.32	0.73	1.37	1.36

For cases that previously failed at sample sizes of one thousand and above, the majority now pass the GoF test. Although the GoF test statistic at zero decimal places shows a marked improvement compared to the baseline, failures persist from the ten thousand sample size onwards.

Figure 7.2 presents Q–Q plots for $n = 10,000$ $m = 0$ for visual comparative analysis on the results of the fit.



(a) Baseline, Round=0.

(b) Relative Constant on Zero Values, Round=0.

Figure 7.2: Log-normal Comparative analysis: Baseline v Relative Constant to Zero Values.

The left panel shows the baseline, while the right panel displays the results obtained using the relative constant technique. In both cases, points deviate from the upper part of the line, indicating right-tailed skewness. The deviations differ, with points lying above the theoretical line in one case and below it in the other. The left chart suggests a mixture model. The average AD GoF statistic

decreased substantially, from 38.19 to 4.10, indicating a notable improvement in fit. However, deviations from the regression line continue to affect GoF accuracy. A shift in the distributions is also evident. From the overall results, a conclusion can be drawn that adding a relative constant to zero values proves to be an effective technique to mitigate the effects of quantisation.

CvM results (not shown) demonstrate that failures which previously occurred from the ten thousand sample size onwards are now restricted to failing at zero decimal roundings at the one hundred thousand and one million sample sizes.

7.3.2.4 Apply Relative Constant to all Values

In contrast to the previous section, where a relative constant was applied only to zero values, here the constant is applied to all values. The analysis focuses on larger sample sizes where $m < 2$, corresponding to the dark grey cells at the bottom of Table 7.6, where the tests failed. Table 7.7 reports the results.

Table 7.7: Log-normal AVG AD GoF Comparison: Relative Constant to Zero Values v Relative Constant to all Values.

	Before (Relative Constant on Zero Values)	
Sample Size	**Rounding = 0**	**Rounding = 1**
10000	4.09	1.02
100000	31.40	1.41
1000000	305.57	4.23
	After (Relative Constant All Values)	
10000	4.62	1.50
100000	33.51	4.74
1000000	330.82	39.20

Little improvement was observed at zero decimal places for any sample size. At one decimal place, applying the relative constant to all values failed at the one hundred thousand sample size, which has previously passed when applying the relative constant to only zero values.

7.4 Discussion

7.4.1 Zero Value Problem

Fitting errors occur when zero values are present in the data prior to model fitting, as demonstrated in Chapter 5. To mitigate these errors and allow the

prechecks of the manageparam function to pass, a constant of 1 was added as a baseline adjustment. Although smaller constants could be applied, for the Weibull distribution, the minimum required adjustment was 1, and this value was adopted consistently across all distributions. Adding a constant does not alter the distributional shape but instead shifts the data along the continuum. In contrast, changes to the distribution's parameters, such as the shape parameter β in the Weibull case, would directly alter its form, influencing skewness. Adding a constant can be problematic if the distribution is expected to be centered at a particular location (e.g., the log-normal case). To reduce this shift, smaller constants were evaluated, ranging from 0.01 down to 0.00001, all of which permitted successful fitting.

The challenge of zero values is not unique to this study as discussed in Chapter 2.1. It occurs widely in reliability analysis and lifetime modelling, where zeros can cause numerical instability or undefined likelihoods. In such contexts, applying a constant offset serves as a general corrective approach, ensuring numerical stability while preserving the fundamental distributional structure.

7.4.2 How To Identify Quantisation

Quantisation replaces continuous variation with repeated discrete values, introducing gaps between adjacent observations and eliminating continuity in the data. The process causes data points to cluster or stack at identical levels, producing duplicate values. Quantisation can often be detected visually through vertically or horizontally aligned points, or step-like patterns, in diagnostic plots. Several graphical methods were examined, including histograms, CDFs, Q–Q plots, and P–P plots; among these, P–P plots provided the clearest evidence of quantisation effects.

These findings highlight the importance of early diagnostic checks for quantisation effects. Simple frequency tables and visual diagnostics, particularly P–P plots, can detect rounding prior to formal GoF tests. Where quantisation cannot be avoided, modelling with discrete distributions may provide a more appropriate framework.

7.4.3 Apply Small Constant to Zero Values

Coarser rounding increases the likelihood of values being rounded to zero, though in practice, zeros were observed only when $m = 0$. Introducing a small

constant (0.1) shifts values near the rounding thresholds, for example, values of 0.49 are shifted to 0.50 and consequently assigned to the next rounding bin, while values of 0.99 are shifted into the 1.0 bin. As a result, only observations at the boundaries of rounding bins are affected, effectively reallocating edge values to the subsequent bin without altering the full shape of the distribution.

Across all sample sizes, both the AD and CvM scores significantly decreased when compared against the baseline. GoF tests additionally passed at the one thousand sample size, that had failed at the baseline test. The small adjustment to zero values had a significant improvement in overall results. Overall, applying a small constant to a limited proportion of observations substantially improved GoF.

7.4.4 Apply Relative Constant to Zero Values

Applying a relative constant to zero valued observations significantly improves GoF relative to the baseline method of adding a fixed constant (1). Reductions in AD statistics are observed across all rounding levels all rounding levels ($0 \leq m \leq 7$) and sample sizes, showcasing a more appropriate adjustment of zero values.

Where $m = 0$, the AD GoF tests continue to fail from sample sizes of ten thousand onwards despite improvements relative to the baseline. This is likely due to quantisation effects persisting, particularly at large sample sizes where even small deviations from continuity are impacting distributional fit. For $2 \leq m \leq 7$ and $n > 10,000$, all AD tests pass under the relative constant approach, including cases that failed under the baseline, suggesting that the fixed constant shifts the data excessively.

Across all levels of precision, the relative constant yields an average improvement of approximately 67% in AD statistics, demonstrating that smaller adjustments proportional to data precision are sufficient to support consistent alignment with the empirical data and the theoretical model, preserving the underlying distribution. Overall, the relative constant approach provides a more robust method for handling zero values in log-normal modelling, with remaining limitations primarily attributed to coarse rounding and quantisation effects.

7.4.5 Apply Relative Constant to all Values

Applying a relative constant to all values was evaluated to determine whether shifting all values, rather than only zero values, relative to the data's precision provided a better adjustment. The test was primarily focused on the larger sample sizes ($n \geq 10,000$) that were previously failing the AD GoF.

Applying the relative constant to all values results in extensive modification of the data. Over 80% of the data was altered below one million observations, and 99% at one million. No improvement was observed at zero decimal places from ten thousand samples upwards. At the one hundred thousand sample size, there was a negative impact on AD GoF, and the GoF test now fails when a relative constant is applied to all values. At a sample size of one million, the AD statistic deteriorated.

When the relative constant is applied to only zero valued observations, improvements in GoF arise from correcting deviations in the lower tail without perturbing the remainder of the distribution. In contrast, applying the relative constant to all observations produces a full shift in the data, which would manifest in the Q–Q plot (see Figure 7.2) as systematic curvature away from the reference line across the full range of quantiles, particularly in the upper tail. The behaviour is consistent with the increased AD statistics reported in Table 7.7, especially at larger sample sizes. These results confirm that the effectiveness of the relative constant over-corrects the distribution and degrades model fit.

Table 7.8 provides a summary of the results.

Table 7.8: Summary Comparison of Quantisation Adjustment Methods.

Method	**Strength**	**Weakness**	**Best Use Case**
Add Constant to Entire Dataset	Improves optimisation stability and prevents convergence failures caused by zero values.	Introduces systematic upward bias by shifting the full distribution.	Useful when fitting fails completely due to support violations or logarithmic transformations.
Apply Small Constant to Zero Values	Preserves most of the original distribution while reducing instability caused by zeros.	May distort lower-tail behaviour and still introduce bias near zero.	Effective when quantisation is concentrated at zero values.
Relative Constant Adjustment	Provides the best balance between stability and preservation of distributional structure.	Does not recover the true unquantised data and may still alter empirical density near zero.	Most effective for heavily quantised datasets requiring improved GoF performance.

7.4.6 Sensitivity Analysis Summary

The experiments demonstrate that the effectiveness of quantisation adjustment methods is sensitive to distribution type, sample size, rounding precision, and the magnitude of the applied constants. Methods based on relative constants were generally the most robust across varying levels of quantisation, as they improved GoF performance while preserving more of the original distributional structure. Applying small constants only to zero values also produced stable improvements, particularly for moderate sample sizes and datasets where quantisation effects were concentrated near zero.

In contrast, adding a fixed constant to the entire dataset was more fragile. Although this approach consistently improved optimisation stability and prevented convergence failures, it introduced greater distortion into the empirical distribution and increasingly biased the fit as sample size increased. Sensitivity to adjustment magnitude was especially apparent for the Weibull distribution, where small constants were insufficient to stabilise fitting behaviour.

The parameter estimation methods also exhibited different sensitivities. MLE was generally more robust for log-normal distributions but became unstable near flat likelihood surfaces and parameter values close to zero. MME showed

greater robustness for Weibull distributions but remained sensitive to distorted sample moments and outliers introduced through quantisation.

Overall, the results demonstrate that no single adjustment strategy is universally optimal. Robust performance depends on balancing numerical stability against preservation of the original distributional structure, with the most effective method varying according to the characteristics of the data and the severity of quantisation.

7.5 Conclusion

The aim of this study was threefold: to provide practical approaches for identifying quantisation in data, to mitigate convergence errors during fitting, and to reduce the impact of quantisation on GoF outcomes.

Visual diagnostics were shown to detect quantisation, although effectiveness varied by method. In particular, Q–Q plots were less reliable, while P–P plots provided clearer evidence through clustering and step-like structures. To address convergence errors, a constant of 1 was initially applied, followed by several variations of this method. Applying a relative constant to zero values proved most effective, raising the AD GoF pass rate from 13% in the baseline to 85%, and the CvM pass rate from 28% to 83%.

Although the methods in this study were demonstrated on the log-normal distribution, they apply to other continuous distributions, such as the Weibull and Normal distributions. Preliminary analysis in these cases indicates improved GoF when applying a relative constant to the data, compared with adding a fixed small constant. However, a detailed analysis of these distributions lies beyond the scope of the present research.

CHAPTER 8

Unrounding the Data

Rounding and quantisation introduce information loss that can degrade downstream modelling and simulation tasks. An investigation into a set of unrounding techniques aimed at mitigating the effects of rounding and partially recovering the distributional properties of the original unrounded data is presented. The study introduces a range of techniques, including jitter-based, rejection sampling, and binning techniques to unround the data. The jitter-based technique introduces controlled random noise, redistributing observations to plausible adjacent values within the rounding interval. Rejection sampling generates candidate values by accepting samples according to a specified target distribution. Histogram binning techniques based on the Freedman–Diaconis, Sturges, and Scott rules are also considered. All techniques use synthetically generated rounded data, and conclusions are drawn from an evaluation of GoF tests, including AD, CvM and NRMSE. The results show that unrounding the data is driven by alignment with the assumed distribution, rather than by local perturbation or binning methods.

8.1 Introduction

As shown in the preceding chapters, rounding obscures data points, distorting the underlying distributional data characteristics. Chapter 5 introduced practical adjustments to fitting failures and addressed GoF issues caused by zero values, including the addition of a constant to avoid convergence issues. In contrast, Chapter 6 identified that MLE outperformed MME in distribution

fitting. Chapter 7 focused primarily on mitigating fitting failures caused by quantisation and zero values through adjustment-based techniques, such as adding constants to the data. While these methods improved convergence and GoF performance, they did not attempt to reconstruct the latent continuous observations underlying the rounded values.

This chapter extends the analysis by investigating whether plausible continuous reconstructions can be generated to better approximate the underlying distributional structure obscured by quantisation. MLE is used in this chapter to improve fitting stability and to better approximate the latent distributional structure of rounded observations.

The methods developed in this chapter do not recover the true underlying data. Instead, they generate plausible continuous values that are statistically consistent with the quantised observations and the assumed distributional framework. Accordingly, the proposed approach should be interpreted as a reconstruction procedure rather than an exact recovery process.

Based on the preceding chapters, further research is needed to develop techniques that compensate for the effects of rounding in time-series data by identifying plausible adjacent values of the rounded data points, thereby addressing convergence issues and the limitations imposed by distributional assumptions.

8.1.1 Assumptions in Reconstruction

The reconstruction methods considered in this chapter rely on several assumptions regarding the relationship between the rounded observations and their latent continuous values.

First, it is assumed that the true underlying observations lie within the quantisation interval defined by the rounded value and its associated precision. Each rounded observation corresponds to a finite interval of plausible continuous candidates.

Second, unless otherwise specified, values within a quantisation interval are commonly assumed to follow a uniform distribution. This assumption implies that all candidate values inside the interval are equally likely, although alternative distributions may also be considered depending on the application and

the reconstruction method employed.

Third, the observations are generally treated as independent unless an ordering or dependence structure is explicitly modelled. In practical settings, this assumption may not always hold, particularly for time-series, sequential logs, or structured message data where temporal or structural dependencies exist between observations.

These assumptions simplify the reconstruction problem and make the proposed methods computationally tractable. However, they also introduce limitations, since violations of these assumptions may affect the accuracy and interpretability of the reconstructed values.

To provide more guidance on the complexity of unrounding, Table 8.1 illustrates the scale of the unrounding problem with a simple example.

Table 8.1: Unrounding Data: Explosion of Plausible Values.

Original Value	**Precision**	**Rounded Value**	**Range Possible Values (at original precision)**	**Number Plausible Values**
4.9	1	5	4.5, 4.6, . . . , 5.5	10
5.01	2	5	4.50, 4.51, . . . , 5.49	100
5.001	3	5	4.500, 4.501, . . . , 5.499	1000
5.0001	4	5	4.5000, 4.5001, . . . , 5.4999	10000
5.00001	5	5	4.50000, 4.50001, . . . , 5.49999	100000

Several unique values rounding to the same value are observed. Each unique value, recorded at a different level of precision, corresponds to a different number of plausible adjacent values.

A value recorded to 4.9 yields only ten plausible adjacent values, whereas a value of 5.00001 yields one hundred thousand potential candidates. All of these cases collapse to the same rounded value (5), but the amount of hidden information differs enormously. Higher precision, although more precise, can become more problematic when attempting to identify the original unrounded observation. The candidate set grows so large that determining the original value becomes practically infeasible, nevertheless, it remains useful to address this challenge rather than disregard it, since meaningful inference depends on recovering as much information as possible.

In a nuclear reactor, safety settings are triggered at thresholds reported to three decimal places. One such case is the "Chemical and Volume Control System" (CBS), where alarm and control functions are defined at pressures such as 0.012 Megapascals (MPa) and 0.019 MPa [155]. A measurement of 0.0194 MPa, rounded to 0.019 MPa, may appear within tolerance, even though it already exceeds the specified limit. In such settings, even small rounding errors may have serious safety implications, making accurate recovery from quantised data essential.

High levels of precision may still require unrounding. Tiny discrepancies, such as billionths of a second in global atomic timekeeping, can accumulate into errors large enough to cause a practical impact on atomic clocks, making it difficult to define a safe threshold for lost precision.

Reconstructing quantised observations is inherently tricky. For a single rounded value, the original observation is only recoverable within its rounding interval, limiting precision but keeping the uncertainty confined to that single observation. The difficulty grows considerably in the presence of many quantised values. Quantisation can result in multiple observations taking on the same reported value, obscuring their true positions on the underlying continuum.

The challenge intensifies when the aim is not only to restore numerical precision but to infer the form of the underlying distribution, F. Multiple observations quantised to the same value might seem less problematic, since the set of plausible candidates is reduced. The difficulty lies in determining where these observations sit along the continuum, and how they collectively shape the underlying distribution. Some statistical tests, such as probability density estimation and parametric continuous GoF tests like AD, assume continuous data. Rounding can distort their results.

One should also consider the dependence structure associated with the rounded values. If the observations are ordered, then the problem extends beyond recovering unrounded values or the distribution from which they are drawn. It also requires preserving the correct ordering of the original sequence of rounded timestamped messages that can be processed as a batch-type message or a single message, especially in the context of EDI messaging. The added layer of complexity highlights that the unrounding problem extends beyond numerical recovery to include the distributional and structural aspects of the data and

message. In some cases, the order might be recovered from the structure of a log file.

Rounding, therefore, has multiple implications. It is not simply a matter of truncation but also of quantisation, each introducing distortions that are difficult to resolve. Far from being a trivial adjustment to the data, rounding is a significant issue that affects many areas of analysis and should not be taken lightly or considered without regard for the downstream challenges it creates.

Extending the work in Chapter 7, this research evaluates a set of advanced methods for unrounding quantised data. Specifically, it considers a range of jitter-based techniques, binning techniques, and broader rejection sampling strategies.

For this study, the aim is not to create new information, but to approximate the uncertainty introduced by quantisation. The objective of these advanced techniques is to mitigate the distortions introduced by rounding by generating plausible adjacent values near the rounded observations. Although these approaches cannot recover the exact original data or its ordering, they can improve the representation of individual observations and better capture the characteristics of the underlying distribution.

The chapter addresses the following questions:

1. Can different unrounding techniques (e.g., Gaussian noise, jittering and rejection sampling) unround the data, bringing it closer to its original form?

2. Which method yields the most accurate reconstruction of the underlying distributional form, and where possible, provides the closest approximation to the original unrounded values?

The study primarily focuses on the Weibull distribution, with limited tests on log-normal and exponential distributions. GoF is evaluated using NRMSE, AD, and CvM tests, with parameters estimated or fixed via MLE. When fitting, a constant of 1 is added to the data to ensure convergence, and tests are repeated five times to account for random data generation. Synthetic data are used throughout, with distributions, parameters, and sample sizes defined

in Table 5.1, excluding the ten thousand sample size. The set of evaluated methods is summarised in Table 8.2.

Table 8.2: Jittering Tests.

Methods and Configurations						
Test Number	**Test Name**	**GoF Est. Param.**	**GoF Fixed Param.**	**All Data**	**Zero Values Only**	**Description**
1	Gaussian Noise Variants	–	Y	Y	–	Two synthetic Weibull datasets are generated to support rounded and non-rounded values using the same parameters. Rounded values are adjusted with Gaussian noise centered at zero. Variants differ by how the standard deviation is calculated.
2	Interval-Based Uniform Jitter	–	Y	Y	Y	A synthetic Weibull dataset is generated and rounded. A second synthetic Weibull dataset is generated using the estimated parameters from the rounded values. Boundary limits are defined for each rounded value using the minimum and maximum values of its neighbour in the second Weibull dataset. Uniform noise is then drawn within the boundaries of each point.
3	Distance-Based Jitter (Addition)	–	Y	Y	Y	A synthetic Weibull dataset is generated and rounded. A second synthetic Weibull dataset is generated using the estimated parameters from the rounded values. Both datasets are sorted. The distance between each consecutive point on the synthetic dataset is calculated. That difference is then added to the rounded ordered dataset.
4	Pit-Based Even Spaced Jitter	–	Y	Y	–	A synthetic Weibull dataset is generated and rounded. On the rounded dataset, each unique rounded value defines a pit with minimum and maximum bounds. Evenly spaced jitter is applied depending on the length of the pit using a seq() function.
5	Jitter Histogram-Binning	–	Y	Y	–	A synthetic Weibull dataset is generated and rounded. Rounded values are redistributed within histogram bins, where bin widths are set using rules such as Sturges, Scott or FD. Observations within each bin are then evenly spaced using the seq() function, becoming the new values for the rounded dataset.
6	Rejection Sampling	Y	Y	Y	Y	Jittered values are generated by repeatedly sampling from a Weibull distribution. Each candidate is rounded and compared against the rounded target dataset; only those matching are retained, while the rest are rejected.
7	Inverse Method	Y	Y	–	Y	A rounded synthetic Weibull dataset is generated. Upper and lower bounds are identified based on the rounding precision of each value. A random value is then sampled uniformly between the corresponding CDF bounds and transformed back using the inverse CDF to recover a plausible unrounded value.

Note: All tests are conducted using synthetic data. The "Inverse Method" is applied to Weibull, log-normal and exponential distributions.

Some of the methods in Table 8.2 require a second Weibull dataset. The dataset is either generated from parameters estimated from the rounded data or created using the known parameters of the original distribution used to

produce the rounded values. It is helpful to consider two practical reasons for this. In the first case, a genuine unrounded dataset becomes available, for example, system logs that were originally recorded only to millisecond precision may later be reconfigured to record timestamps at the nanosecond level. The high-resolution dataset can then be used to understand the structure of the original data and guide the reconstruction of the earlier, coarsely rounded values. In the second case, no such unrounded dataset exists. Instead, the underlying distribution must be inferred directly from the rounded data, and a synthetic unrounded dataset is generated using the fitted parameters.

The methods proposed in this chapter extend the heuristic adjustment approaches developed in Chapter 7 by explicitly modelling the uncertainty introduced through quantisation. Rather than modifying observations directly through additive corrections, interval-based approaches represent rounded values as ranges of plausible underlying observations, thereby preserving more information about the original continuous distribution.

Table 8.3 provides a summary of the strengths and weaknesses between Chapter 7 and in Chapter 8.

Table 8.3: Comparison of Quantisation Mitigation Approaches Across Chapters.

Approach	**Strength**	**Weakness**	**Description**
Raw Quantised Data	Simple and computationally efficient.	Produces biased parameter estimates and distorted GoF results.	Fits statistical distributions directly to rounded observations without adjustment.
Adjustment Methods (Chapter 7)	Improves fitting stability and reduces convergence failures.	Introduces heuristic bias through artificial modification of observations.	Applies constants or relative adjustments to reduce the effects of quantisation.
Interval-Based Methods (Chapter 8)	Provides a principled representation of quantised observations by modelling plausible underlying intervals.	More computationally complex and dependent on interval assumptions.	Represents rounded observations as intervals rather than fixed point estimates.

Collectively, these tests provide a diverse framework for evaluating whether jittering can effectively restore variability lost through rounding. The techniques developed in this research are intended for real-world applications using real-world datasets.

8.2 Data Overview and Limitations

The data used in this chapter, along with its limitations, have been previously discussed in Chapter 5.2.

Although stochastic jitter methods may generate slightly different reconstructed datasets across repeated runs, the emphasis of this study is on the comparative behaviour of reconstruction methodologies rather than the optimisation of individual random realisations. Consequently, the reported results should be interpreted as representative examples of method behaviour under quantised conditions.

8.3 Methods

Jittering is applied in this study as a means of restoring variability to rounded datasets while preserving the overall distributional structure. The techniques mentioned in these tests adjust discrete values introduced by rounding by adding controlled noise to each data point. The objective is not to jitter the distribution as a whole, but to introduce variability at the level of individual points.

Previous work demonstrated that small additive adjustments can improve fitting stability under quantised conditions, but may also introduce systematic bias into parameter estimation and GoF assessment [156]. This motivates the development of reconstruction-based approaches that aim to preserve more of the original continuous distribution while mitigating the effects of quantisation.

8.3.1 Jitter

In this chapter, the concept of a pit is introduced to distinguish between conventional binning strategies, such as Sturges or FD. A pit is a bin defined by unique rounded values. Unlike conventional histogram bins, pits arise naturally from the rounding process itself. Each distinct rounded value forms its own pit, and all observations rounded to that value fall within that pit. The terminology

is adopted to distinguish bins from other binning strategies discussed later. The pit structure will be used in subsequent sections.

8.3.1.1 Gaussian Noise Variants

To address quantisation, a Gaussian jittering process using bins referred to in this section as pits is studied. Rounded observations are grouped into pits, defined by the rounding precision. The rounding precision d determines the pit width. For example, when rounding to 0 decimal places, the pit width is 1 unit (e.g., all values are ≥ 4.5 and ≤ 5.4 collapse into the integer pit 5). Each pit gives a plausible range of adjacent values.

For this study the aim is to evaluate how applying small Gaussian noise, with variations in the standard deviation introduced through different scalings and variants, influences dispersion and the recovery of smoothness in the empirical distribution. The technique is expected to reduce the distortion in the GoF statistics.

Gaussian jittering was selected because it provides a simple and computationally efficient mechanism for reintroducing variability into quantised observations. The Gaussian distribution allows noise to be concentrated around the rounded value while still permitting controlled dispersion through the standard deviation parameter. This makes it useful for evaluating how different levels of local variability influence both pointwise reconstruction accuracy and distributional smoothness.

Although Gaussian jitter does not represent the same shape as the Weibull distribution. It serves as a generic smoothing mechanism. Consequently, the effectiveness of the reconstruction is sensitive to the choice of variance scaling. Excessive variance can over-smooth the data and inflate the tails, whereas insufficient variance may preserve discreteness and fail to adequately mitigate quantisation effects.

Different variants of the Gaussian noise are presented in Figure 8.1, with workflow illustrations and corresponding mathematical formulas defined for each test.

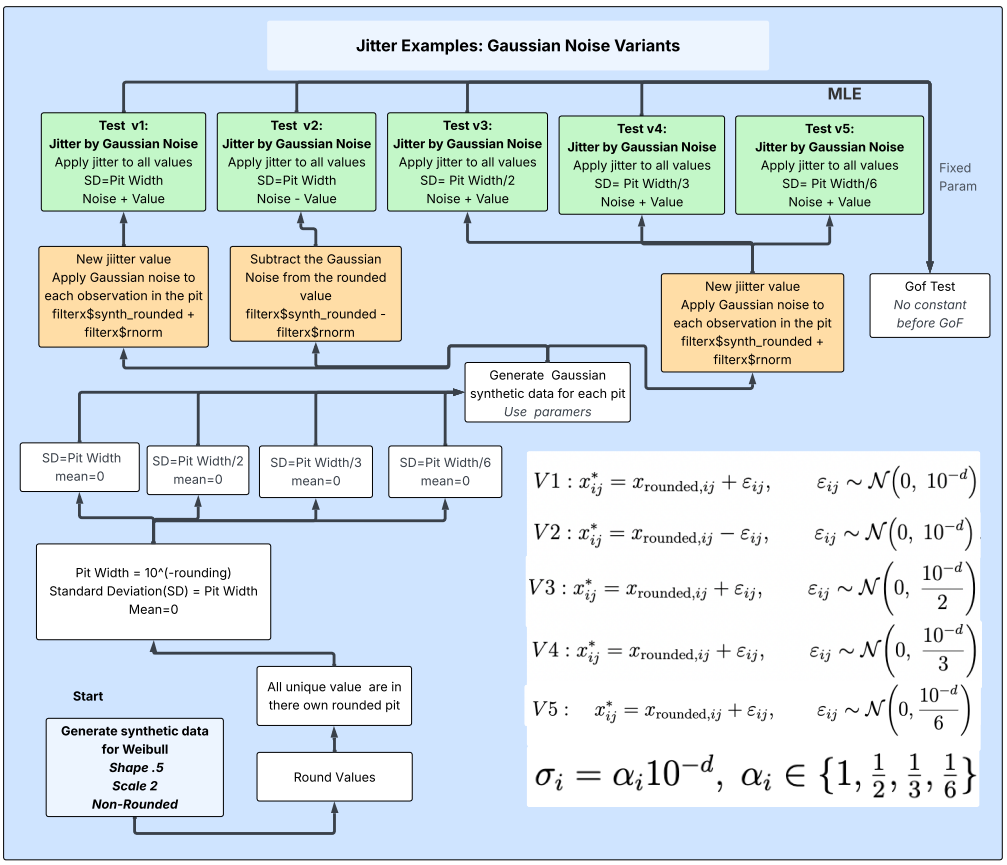


Figure 8.1: Flow Diagram: Gaussian Jitter Variants.

Based on Figure 8.1, five Gaussian-noise jittering tests are evaluated. All tests use Gaussian noise centred at zero ($\mu = 0$), ensuring that jittered values remain close to their rounded observations, while preserving alignment with the original data points, and reintroducing the continuum structure lost through rounding. By centering the noise at zero, the jitter restores local variability relative to each observation without shifting the overall location of the dataset.

Three strategies are used to determine the standard deviation of the Gaussian noise.

For the first strategy, two Weibull datasets are generated with identical parameters, one rounded and one unrounded. For each pit, the standard deviation is estimated from the unrounded values that would fall within that pit (colour coded pink in Figure 8.1). If a pit contains only a single observation, the standard deviation is undefined, and a small constant (0.01) is used instead.

For the second strategy, only the rounded dataset is used. The standard deviation for each pit is set equal to the pit width determined by the rounding precision (colour coded brown in Figure 8.1). When rounding to zero decimal places, the pit width is 1, and therefore the standard deviation also becomes 1. Several scaled versions of the pit-width standard deviations are examined.

The light-green boxes in Figure 8.1 illustrate the Gaussian-noise jittering process when pit-based standard deviations are estimated from unrounded values (pink). The dark-green boxes show the corresponding process when the standard deviation is derived solely from pit width (brown).

A unique noise value is generated for each observation, not per pit. For example, if a pit contains 20 observations with integer value 5, then 20 distinct noise terms are drawn. All jittered values are constrained to remain non-negative (by applying the absolute value), consistent with the Weibull distribution.

8.3.1.2 Interval-Based Uniform Jitter

Applying uniform noise to a dataset is a simple technique for mitigating the effects of quantisation. Noise is drawn from a bounded range and added to the rounded data, thereby restoring a degree of continuity. The jitter bounds are derived from the adjacent spacing in an independently generated Weibull series, with the minimum and maximum values reflecting consecutive order statistics.

Uniform jittering was selected because it reflects the assumption that all values within a quantisation interval are equally plausible in the absence of additional information about the latent continuous observations. This assumption provides a simple and interpretable baseline reconstruction strategy that avoids imposing a strong parametric form within the interval.

However, the reconstruction quality is sensitive to the interval boundaries used to generate the uniform noise. When the interval becomes excessively wide, particularly in heavy-tailed distributions such as the Weibull distribution with shape parameter 0.5, the resulting jitter may generate unrealistic values and artificially smooth the empirical distribution. Consequently, the choice of interval construction substantially influences both pointwise reconstruction accuracy and distributional GoF performance.

In contrast to the Gaussian approach, this method bases its uniform jitter on inter-sample Weibull distances rather than rounding precision. Gaussian noise is applied per-pit using a local Gaussian distribution within each quantisation interval, centered at zero, this per-pit approach preserves the local structure of the data while still mitigating quantisation artefacts. Uniform noise is bounded within fixed limits. Both approaches are simple to implement, but they differ in the magnitude of adjustment used. In practice, it is more effective to apply minimal noise to shift rounded points toward adjacent plausible values without compromising the underlying distribution. The flowchart for the uniform jitter is defined in Figure 8.2.

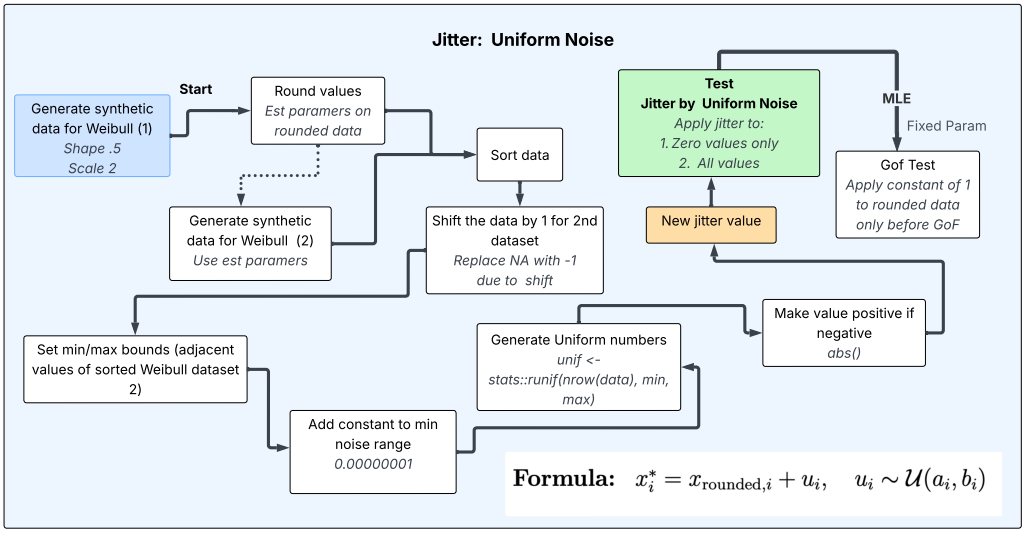


Figure 8.2: Flow Diagram: Uniform Jitter.

To explain the process, two Weibull distributions are generated. The first distribution is rounded to support the tests, while the second distribution is estimated from the rounded data and then used to define upper and lower bounds for each value. For each rounded value, a random draw is taken from a Uniform distribution bounded by the defined intervals for each value, ensuring that the jittered values remain within a plausible range. Any negative jittered values are replaced by their absolute value to ensure that the final jittered values remain non-negative. A small constant is added to the lower bound to prevent zero-width intervals. These can arise in several situations, for example, at the first or last observation, where a neighbour is missing and the bound collapses or when numerical precision causes consecutive values to be treated as

equal. The constant ensures that every observation is assigned a valid non-zero interval for jittering, even in these edge cases.

8.3.1.3 Distance-Based Jitter (Addition)

Distance-based jitter leverages the spacing between ordered values of two distributions, both generated from a Weibull distribution, where one is rounded, and the other is left in its original form. The intuition is that local distances in the unrounded reference distribution can be transferred to the rounded synthetic values. By adding these distances back to the rounded data, continuity is restored and discreteness reduced.

The procedure differs from Gaussian noise and uniform jitter, as previously mentioned. Gaussian noise depends on variance, while Uniform noise is bounded within fixed limits. Neither directly reflects the characteristics of the synthetic distribution. In contrast, distance-based jitter is distribution-aware. The noise is determined from the same Weibull reference distribution that generated the rounded data, ensuring that variability remains consistent with the underlying distribution, rather than imposing artefacts from a different distribution. Figure 8.3 provides pseudocode of the process flow for this technique.

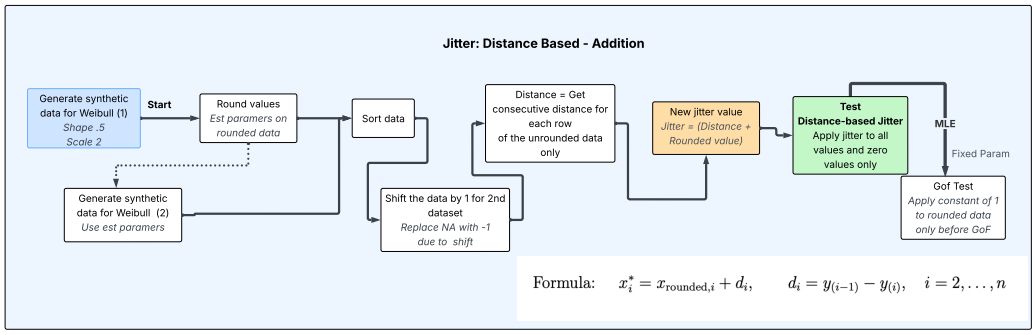


Figure 8.3: Flow Diagram: Jitter by Distance (Addition).

The process begins by ordering both the rounded and unrounded synthetic datasets. Consecutive distances from the ordered unrounded Weibull series are then calculated and added to the corresponding rounded synthetic values, producing new jittered values based on the spacing of the reference distribution.

The technique may exaggerate the distributional shape by increasing tail heaviness, while also slightly reducing the height of the central peak as probability

mass is redistributed away from the rounded center. Nevertheless, it can provide closer alignment with the original values and underlying distribution, particularly in the head of the data where rounding collapses small values into the zero range.

8.3.1.4 Pit-Based Even Spaced Jitter

The "Even Spaced Jitter" technique builds upon previous techniques, where, unlike Gaussian noise, it does not impose assumptions about the noise distribution. Unlike "Distance-Based Jitter", it does not depend on alignment with a secondary distribution. Within each pit, observations are reassigned to evenly spaced positions, with spacing determined by the number of observations in the pit and the width of the bounded interval. The resulting sequence may provide an effective means of de-quantising the data, as it may reduce discreteness on the continuum and offer a closer approximation to the original underlying values. Figure 8.4 illustrates the workflow of this technique.

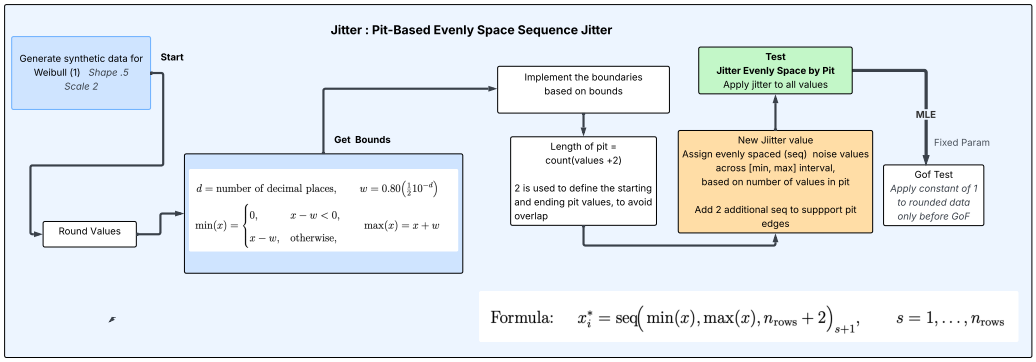


Figure 8.4: Jitter Flow Diagram: Evenly Spaced Sequences Within Rounded Pits.

To provide more detail, the 'Get Bounds' box in Figure 8.4 constructs pit boundaries directly from the rounding precision, which prevents overlapping interval limits for each rounded value. If a value is rounded to d decimal places, the corresponding pit width is: 10^{-d}, and a reduced half-width

$$w = 0.80\left(\tfrac{1}{2}10^{-d}\right)$$

It is used to ensure that neighbouring pits avoid overlap and to avoid endpoints which may be troublesome for fitting. For each rounded observation x, the plausible interval is therefore defined as:

$$[x - w,\ x + w].$$

With the boundaries defined for each unique pit i, a sequence of evenly spaced values is generated across the interval, with length equal to the number of observations in the pit plus two. The addition of two extra sequence points ensures the jittered samples are drawn from the interior of each interval. Together, these steps prevent overlap between pits and stop pit-edge distortion by keeping jittered values away from their boundaries.

Noise values are then assigned by mapping each observation in the pit to the corresponding element of the sequence (excluding the first and last elements). The technique ensures that each rounded observation is linked to a defined range of possible original values. As the Weibull distribution has support on $[0, \infty)$, any lower bound that would fall below zero is truncated to zero, preventing the construction of negative pit limits, which provides a unified rule for defining jitter intervals across all rounded values.

The technique aims to avoid over and under-adjusting the rounded values, supporting a smoother evolution on the continuum.

8.3.1.5 Jitter Histogram-Binning

Histogram binning is a well-known technique for grouping continuous data into intervals of fixed width, referred to as bins. Each observation is assigned to the bin corresponding to its value. By adjusting the bin width, one can control the interval size used to represent the PDF with a histogram. In this study, two variants of histogram-based binning techniques are considered. Version 1 (V1) employs fixed-width bins determined by rounding precision, while Version 2 (V2) defines bin boundaries using standard histogram rules such as Sturges, Scott and FD.

Unlike "Pit-Based Even Spaced Jitter", which defines a separate pit for each unique rounded value, "Histogram-Binning" constructs bins using fixed boundaries that are defined independently of the observed values. As a result, histogram bins are homogeneous in width and remain consistent across the distribution. Such a distinction highlights the difference between "Pit-Based Even Spaced Jitter" and "Histogram-Binning". Figure 8.5 illustrates the process flow for V1 and V2.

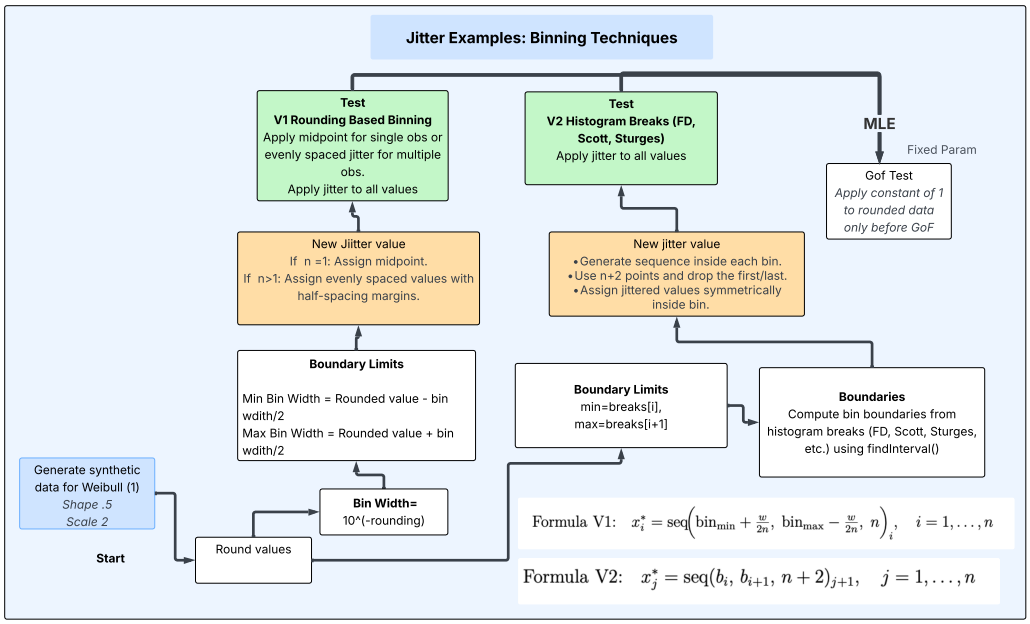


Figure 8.5: Flow Diagram: Jitter Histogram Binning Techniques.

For V1, rounded values are grouped into fixed-width bins determined by the rounding precision. When a bin contains a single observation, the value is shifted to the interval midpoint. When multiple observations fall within the same bin, they are redistributed symmetrically around the midpoint with equal spacing, leaving half-spacing margins at the edges to avoid zero values. For V2, bin boundaries are determined using histogram rules such as Sturges, Scott and FD. After the bin limits are fixed, the redistribution procedure is the same for both variants. The two versions differ only in how the bin boundaries are defined.

Using this technique, one can employ the binning rules to support non-parametric fitting techniques for mitigating the effects of quantisation. By relating "Histogram-Binning" to KDE, one can approximate the underlying distribution more smoothly. Applying different binning rules provides a way to compare how well each rule aligns with the KDE, offering insights into rules for redistributing the quantised data. As discussed in Section 2, rules such as Sturges, Scott and FD represent widely used approaches for data binning techniques. Once values are grouped into the relevant bins, they are reassigned within each bin.

Unlike V1, which applies fixed-width rounding-based bins, V2 employs adaptive

binning rules that mitigate the effects of rounding and provide a more flexible way to redistribute the data.

Tables 8.4 shows a comparison of the different jitter methods.

Table 8.4: Comparison of Jitter Reconstruction Methods.

Comparison of Jitter Reconstruction Methods			
Method	**Strength**	**Weakness**	**Best Use Case**
Gaussian Noise Variants	Reduces repeated rounded values and improves continuity.	May distort tails through artificial random noise.	Useful when smooth stochastic reconstruction is required.
Interval-Based Uniform Jitter	Preserves interval consistency within rounding bounds.	Uniform assumptions may not reflect local density.	Effective when preserving interval structure is prioritised.
Distance-Based Jitter	Uses local spacing to reconstruct observations.	Sensitive to ordering and irregular local distances.	Useful when adjacent observations provide meaningful information about local spacing in the underlying distribution.
Pit-Based Evenly Spaced Jitter	Produces reconstruction within rounded pits.	May suppress natural stochastic variation.	Effective for reducing clustering in repeated values.
Jitter Histogram-Binning	Preserves empirical histogram structure.	Sensitive to bin-width and histogram assumptions.	Useful when global density preservation is important.

These differences demonstrate that no single jitter method is universally optimal. Reconstruction effectiveness depends on the severity of quantisation and the statistical properties that are prioritised during reconstruction.

8.3.2 General Sampling Techniques

Sampling techniques generate random values from a target distribution that represents the underlying continuous data. These sampled values are used to replace or adjust rounded observations. The overall goal is to reconstruct plausible continuous values while maintaining consistency with the assumed distribution.

8.3.2.1 Rejection Sampling

Rejection sampling generates random samples from a specified distribution and can be used to unround quantised data. To explain the process, for each value in the rounded dataset, multiple random samples are drawn from a specified distribution. Each candidate data point is then evaluated to determine how

closely it aligns with the rounded value under consideration. If a candidate, when rounded, matches the corresponding rounded value, it is accepted as the new unrounded value. Accepted values are stored in a list, and when new samples are generated, any values already used are skipped, which ensures that the reconstructed dataset avoids duplication. The procedure is repeated until every point in the rounded dataset has a match. If no match occurs for a given value, the sampling process is repeated until one is obtained. The extent of rejection sampling required depends on the precision of the data points. Lower precision typically results in a smaller number of distinct values, whereas higher precision often requires more iterative looping to obtain a match.

For this chapter, rejection sampling is applied to the synthetic data. For this method, there are no repeat tests. Greater emphasis is placed on the head of the distribution, where quantisation is most pronounced due to the volume of zero values, although the method is applied across the entire dataset for the synthetic dataset.

Given the simplicity of the technique, no process flow diagram has been created.

8.3.2.2 Inverse Method

Rejection sampling is a flexible method for unquantising data using generated random datasets from a specified distribution. While effective, the process can be computationally expensive because many samples may need to be generated and discarded before a sufficient number of matched points are obtained, making it time-consuming for large datasets especially when the precision of the data points grow.

To address this limitation, the inverse method provides a more direct and efficient alternative. Rather than relying on repeated sample generation, the inverse method relies on the principle that a random variable X, with CDF $F(x)$, can be sampled by first drawing a uniform random variable, and then computing the inverse:

$$U \sim \text{Uniform}(0, 1), \quad X = F^{-1}(U)$$

The “Inverse-Method” requires that the CDF of the target distribution can

be inverted either in closed-form or through numerical approximation. For example, the exponential distribution with rate parameter λ is an example of a closed-form. It has a CDF:

$$F(x) = 1 - e^{-\lambda x}$$

And its inverse CDF can be expressed in closed-form as:

$$F^{-1}(u) = -\frac{1}{\lambda}\ln(1-u), \quad u \in (0,1).$$

Not all distributions have a closed-form. For example, a Normal distribution has no closed-form. Instead, it is expressed using the inverse error function:

$$F^{-1}(u) = \sqrt{2}\,\mathrm{erf}^{-1}(2u-1),$$

These non-closed-form distributions can be evaluated numerically using approximation algorithms such as NR. Table 8.5 lists examples of distributions with closed-form inverse CDFs alongside those that require numerical computation.

Table 8.5: Distributions: Inverse CDF.

Inverse CDF Examples		
Distribution	**Inverse CDF**	**Computation**
Exponential (λ)	$F^{-1}(u) = -\frac{1}{\lambda}\ln(1-u)$	Closed-form
Weibull (β, λ)	$F^{-1}(u) = \lambda(-\ln(1-u))^{1/\beta}$	Closed-form
Uniform (a, b)	$F^{-1}(u) = a + (b-a)u$	Closed-form
Normal (μ, σ)	$F^{-1}(u) = \mu + \sigma\sqrt{2}\,\mathrm{erf}^{-1}(2u-1)$	Numerical
Log-normal (μ, σ)	$F^{-1}(u) = \exp\Big(\mu + \sigma\sqrt{2}\,\mathrm{erf}^{-1}(2u-1)\Big)$	Numerical

By eliminating the iterative acceptance rejection step, the "Inverse Method" can produce samples much faster, provided the inverse CDF is available in closed-form or can be computed. Rather than repeatedly drawing samples until one falls in the correct interval, the method simply restricts the uniform variable U to the probability range for that interval and applies the inverse CDF once.

In this study, the "Inverse Method" is adapted to account for rounding in the observed data. Figure 8.6 shows a lower level view of the steps this method takes.

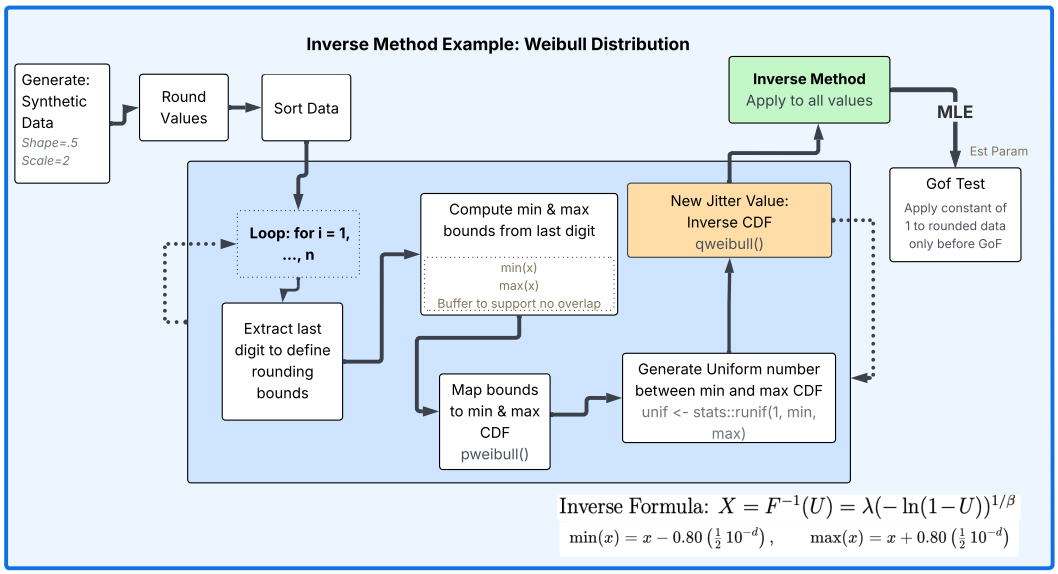


Figure 8.6: Jitter Flow Diagram: Inverse Method.

First, a Weibull synthetic dataset (shape = 0.5, scale = 2) is generated and rounded to the specified decimal precision. The data are then sorted to ensure consistent grouping. For each rounded observation, the rounding precision determines the width of the rounding bin.

Applying the "Inverse-Method" will be a fast version of "Rejection-Sampling". Here, the factor 0.8 is used to define a hybrid sampling technique, but does not imply equivalence with "Rejection-Sampling". Instead, it provides an alternative for comparison, in which the scaling factor can be varied to explore the resulting GoF behaviour.

If the data are rounded to d decimal places, the rounding unit is 10^{-d}. For example, if $x = 0.7$ and $d = 1$ $(10^{-1} = 0.1)$, then

$$w = 0.80\left(\tfrac{1}{2}10^{-d}\right) = 0.04, \quad \min(x) = x - w = 0.66, \quad \max(x) = x + w = 0.74.$$

This gives the interval width $[\min(x),\ \max(x)] = [0.66,\ 0.74]$. The truncated interval retains 80% of the nominal rounding width, preventing overlap between adjacent bins while still restricting jitter values to a plausible neighbourhood of each rounded observation.

These bounds are then mapped to their corresponding Weibull CDF values, and a uniform random probability is sampled within this interval. Applying the inverse Weibull CDF to this probability produces a jittered value consistent

with the underlying distribution. The technique is applied to each rounded value in the synthetic Weibull dataset. GoF tests are then performed on the data, and the entire process is repeated five times to assess variability and robustness.

8.4 Results

To assess how effectively the jittering techniques recover both the underlying distribution and the original point values, AD and CvM tests are applied to evaluate distributional fit, while NRMSE measures point-level recovery by quantifying how closely the adjacent plausible (jittered) values align with the original values. A benchmark comparison is conducted to evaluate and contrast the performance of the different techniques.

8.4.1 Jitter Methods

For each jittering method, results are presented in both tabular and graphical form. Each table summarises the outcomes of the GoF tests, specifically the AD, CvM, and NRMSE statistics under two rounding scenarios: zero decimal places, representing severe loss of precision, and three decimal places, representing a more realistic level of rounding.

For the interpretation of these results, the AD critical value (AD = 2.49, Table 2.4) and the CvM critical value (CvM = 0.46, Table 2.3) are used to assess the effectiveness of jittering relative to the corresponding GoF statistics. To aid the tabular comparison, cells coloured blue represent tests that pass the AD or CvM GoF tests. **Note.** *No colours are presented for three decimal places as the primary focus is on zero decimal places.*

The quantitative results are complemented by P–P plots, based on a sample size of one thousand, which provide a visual illustration of how the application of jitter affects the GoF performance.

RMSE provides a measure of the discrepancy between jittered and original values. While there is no universal threshold that defines an "optimal" RMSE threshold, smaller values generally indicate lower distortion relative to the chosen scale. The interpretation of RMSE strength is scale-dependent and therefore most meaningful when expressed relative to a reference point such as the sample mean. To report error magnitude as a percentage of the mean,

RMSE is normalised (NRMSE) as NRMSE $= \frac{\text{RMSE}}{\bar{y}} \times 100\%$. However, as RMSE squares deviations before averaging, it is highly sensitive to outliers. Caution is required when interpreting results, as a small number of extreme values may disproportionately inflate RMSE and obscure the overall performance of the jittering methods. There is no standardised set of "strong/weak" thresholds for NRMSE, unlike Pearson's r. For this research, the strength of NRMSE will be interpreted using my own classification presented in Table 8.6. Any cells where an NRMSE value is <= 20%, those cells will be coloured blue for acceptance.

Table 8.6: NRMSE Threshold Values.

NRMSE (%)	**Colour Code Cells**	**Interpretation**	**Description**
0%	Blue	Excellent	Minimalistic differences between predicted and observed values
<= 10%	Blue	Strong	Small differences between predicted and observed values.
11 : 20%	Blue	Moderate	Some differences between predicted and observed values but still accepted
> 20%	White	Weak	Large volume of discrepancies between predicted and observed data.

Including an NRMSE strength classifications table provides an additional framework for interpreting the relative accuracy of each jittering method. Analysing NRMSE values within defined performance ranges, becomes easier to compare results objectively and assess the degree to which each method preserves data fidelity.

8.4.1.1 Gaussian Noise Variants

Table 8.7 shows the NRMSE results of the Gaussian jittering techniques.

Table 8.7: Performance Metrics: NRMSE Gaussian Noise.

Test		**Rounding=0**	**Rounding=3**	**Rounding=0**	**Rounding=3**
	Standard Deviation (SD)	**Sample Size = 100**		**Sample Size = 1000**	
Baseline: Rounded vs Original		8.01%	0%	6.76%	0%
Group 1					
V1: All Gaussian Noise	SD	9.98%	0.17%	8.48%	0.23%
V2: Gaussian Noise	SD/2	5.89%	0.22%	6.99%	0.19%
V3: Gaussian Noise	SD/3	5.89%	0.22%	6.99%	0.19%
V4: Gaussian Noise (Subtract Noise)	SD	5.87%	0.22%	6.98%	0.19&
Group 2					
V1a: All Gaussian Noise	Pit Width	6.50%	0.03%	24.55%	0.02%
V2a: Gaussian Noise	Pit Width/2	8.19%	0.01%	13.07%	0.01%
V3a: Gaussian Noise	Pit Width/3	6.21%	0.01%	9.73%	0.01%
V4a: Gaussian Noise (Subtract Noise)	Pit Width	13.56%	0.02%	23.78%	0.02%
V5: Gaussian Noise	Pit Width/6	6.76%	0%	6.43%	0%

Across all sample sizes, Gaussian jittering with reduced variance (e.g., SD/2 or SD/3) consistently yields lower NRMSE than jitter generated using the full standard deviation. Based on the NRMSE classification thresholds in Table 8.6, both these SD-based methods demonstrate strong performance (NRMSE is $<$ 14%).

Comparing the first group of tests to the second group, Group 2, which uses the pit width as the standard deviation, shows greater variability. In particular, V4a (subtract noise) with coarser rounding yields poor pointwise recovery, resulting in a marked degradation (e.g., NRMSE $>$ 13%).

The V5 method performs competitively and remains stable across rounding levels, yielding results consistent with those of Group 1 across sample sizes.

Given that NRMSE values generally remain within the strong performance (<20%) range, a comparable level of fit would be expected in the GoF tests. Subsequent analysis focuses on AD and CvM. Figure 8.7 presents the P–P plots of the reconstituted distributions with respect to the AD test for the five Gaussian noise variants at the one thousand sample size. The sample size was selected for visual display because it reflects typical data volumes encountered in real-world datasets.

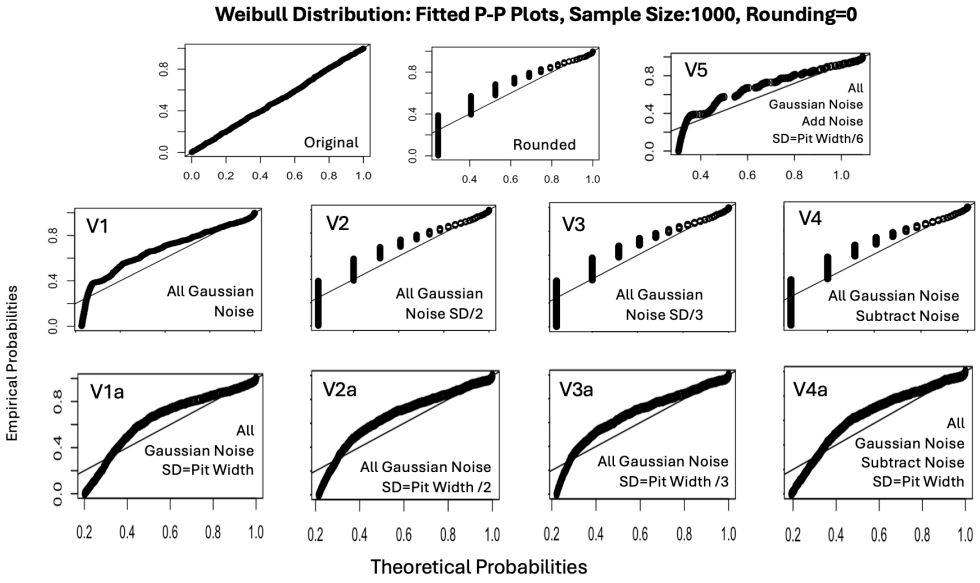


Figure 8.7: AD Distribution Fitting: Jitter Gaussian Noise.

The first row of charts shows the original, rounded and V5 P–P plots. The

second row of charts shows the results of the Group 1 tests, while the third row of charts shows the results of the Group 2 tests.

Interestingly, the results of the plots do not lend towards the same conclusion as NRMSE. The Group 1 charts show the adverse effects of rounding on GoF, with the data points forming discrete steps rather than aligning smoothly with the theoretical line. In contrast, the Group 2 charts show improved smoothness, with points closer to the fitted line, although V5 shows a re-emergence of discreteness toward the lower end of the fitted line. These results are far from acceptable and may not pass GoF for AD; however, V3a and V5 look to be the most optimal.

To support the visual observations, Table 8.8 reports the corresponding average GoF statistics, with greater emphasis placed on these quantitative results than on the visual plots. Any tests at zero decimal places that pass the GoF test are colour coded blue.

Table 8.8: AD/CvM GoF: Gaussian Noise Performance Results.

Tests	**Rounding = 0**	**Rounding = 3**	**Rounding = 0**	**Rounding = 3**
	Sample Size = 100		**Sample Size = 1000**	
	AD Statistics			
Original AD Range	0.64:0.82	0.71:1.37	0.84:0.90	0.62:1.30
Rounded AD Range	Inf	0.65:Inf	Inf	Inf
V1: All Gaussian Noise	Inf	0.75	Inf	1.36
V1a: All Gaussian Noise (pit width)	8.12	0.49	71.17	0.65
V2: Gaussian Noise (SD/2)	Inf	2.67	Inf	Inf
V2a: Gaussian Noise (pit width/2)	3.33	0.61	34.63	0.69
V3: Gaussian Noise (SD/3)	Inf	2.67	Inf	Inf
V3a: Gaussian Noise (pit width/3)	2.16	0.62	20.18	0.67
V4: Gaussian Noise (Subtract Noise)	Inf	2.67	Inf	Inf
V4a: Gaussian Noise (pit width subtract noise)	6.47	0.64	68.78	0.71
V5a: Gaussian Noise (pit width/6)	1.16	0.92	7.23	1.52
	CvM Statistics			
Original CvM Range	0.10:0.14	0.12:0.24	0.11	0.11:0.21
Rounded CvM Range	2.30	0.12:0.24	21.49:21.91	0.11:0.22
V1: All Gaussian Noise	0.20	0.12	0.92	0.21
V1a: All Gaussian Noise (pit width)	1.58	0.12	13.61	0.09
V2: Gaussian Noise (SD/2)	2.30	0.24	21.49	0.11
V2a: Gaussian Noise (pit width/2)	0.55	0.10	5.75	0.08
V3: Gaussian Noise (SD/3)	2.30	0.24	21.49	0.11
V3a: Gaussian Noise (pit width/3)	0.32	0.10	2.93	0.08
V4: Gaussian Noise (subtract noise)	2.30	0.24	21.49	0.11
V4a: Gaussian Noise (pit width subtract noise)	1.22	0.10	13.25	0.08
V5a: Gaussian Noise (pit width/6)	0.17	0.17	0.98	0.26

Note: The original and rounded GoF results will not be colour coded, nor will results at three decimal places.

The "Original AD Range" and "Original CvM Range" are the minimum and maximum range of values observed from the results of the average repeated tests, which were done to reduce the amount of noise in the table.

For AD, interestingly, using the standard deviation of the non-rounded values produces Inf results; however, for the pit width standard deviation, the results are more favourable when applying a third or a sixth of the standard deviation, providing the most optimal results, allowing the AD GoF test to pass its hypothesis test. No tests passed at the larger sample size when rounding to zero decimal places.

It is worth noting that the AD tests under zero-decimal rounding frequently

failed to converge, with four out of five tests failing. For these tests, no constant was added to the data to overcome convergence.

As CvM applies even weight across the distribution, the results may differ from those of AD. Looking at the results, 'V1', 'V3a', and 'V5a' produced the closest fit to the Weibull distribution for the one hundred sample size, with no test passing on the larger sample size.

8.4.1.2 Interval-Based Uniform Jitter

As discussed previously, the tails of the distribution are less affected by rounding due to fewer points in the tail, while the spacing between the points is more significant than seen in the head, as seen by the P–P plot in Figure 8.7. To complement this observation, tests with uniform jitter were conducted both on the head of the distribution (only where zero values occur) and on the full dataset, enabling a direct comparison between results obtained when focusing on the head alone versus the entire dataset.

Table 8.9 shows the NRMSE results.

Table 8.9: Performance Metrics: NRMSE Interval-Based Uniform Jitter.

	Rounding=0	
Metric	**Sample Size = 100**	**Sample Size = 1000**
NRMSE-All Noise	266.76%	286.77%
NRMSE-Zero Values Noise	241.41%	272.56%

An NRMSE above 200% means that, on average, the model's prediction errors are more than twice as large as the mean of the actual values, indicating a complete loss of predictive accuracy. As mentioned previously, NRMSE is strongly influenced by outliers. At this range, jittering introduces greater distortion than correction, and is not suitable for de-quantising or reconstructing the original distribution. Inspection of the pointwise differences shows that, for the one hundred sample case, a jittered value shifted by as much as 34 units (e.g., 2.95 moving to 37.0), which is a substantial displacement along the continuum, placing the reconstructed value far from its original, nearby, and adjacent plausible values. While this displacement is substantial, the evaluation continues with the remaining GoF results to provide a more comprehensive view of how the techniques behave.

As noted in the previous section, greater distortion was observed in the head of the data, suggesting that applying noise to this area rather than to the full

dataset may be more effective. Also, the AD test assigns greater weight to the tails of the distribution, whereas CvM distributes weight more evenly across the full dataset. The distinction between the two GoF tests is expected to influence the outcome of the GoF tests.

Consistent with this reasoning, Table 8.9 shows that the 'NRMSE–Zero Values Noise' variant produced a smaller error magnitude (up to 25%) than when noise was applied to the full dataset.

Figure 8.8 shows the fitted results for the AD GoF test applied to the full dataset.

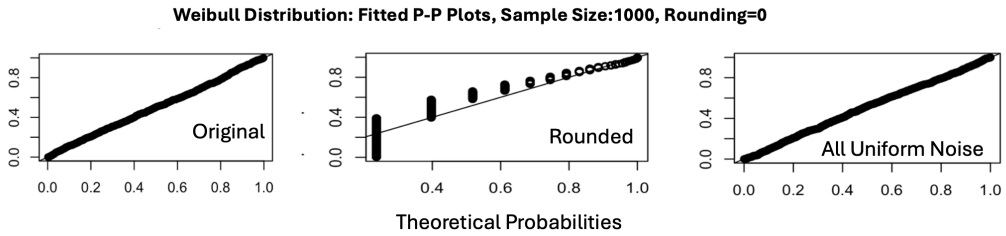


Figure 8.8: AD Distribution Fitting: Interval-Based Uniform Jitter.

Although the P–P plots indicate that the "All Uniform Noise" variant closely reproduces the theoretical distribution, this contradicts the high NRMSE values, which reflect substantial pointwise deviations from the original data. The discrepancy may be due to the fact that NRMSE measures absolute numerical differences, whereas the GoF tests assess distributional similarity. In this case, "Interval-Based Uniform Jitter" smooths the quantised data and restores the overall distributional form, but does not preserve individual data fidelity. To further substantiate these findings, Table 8.10 is analysed to examine the corresponding GoF outcomes.

Table 8.10: AD/CvM GoF: Interval-Based Uniform Jitter Performance Metrics.

	Rounding=0	
Metric	**Sample Size = 100**	**Sample Size = 1000**
Original AD	0.79	0.97
Rounded AD	21.94	222.6
Jitter AD	1.03	0.78
Zero-Value Jitter AD	0.96	2.81
Original CvM	0.14	0.16
Rounded CvM	4.77	48.52
Jitter CvM	0.19	0.11
Zero-Value Jitter CvM	0.18	0.64

Applying "Interval-Based Uniform Jitter" to zero values yields consistently

lower GoF test statistics than applying uniform noise to the full distribution, indicating better mitigation of quantisation effects. In contrast, for the larger sample size, the results reverse. Regardless of this, both methods at the one hundred sample size pass the AD and CvM GoF tests. At the larger sample size, only applying the noise to the full dataset passed the GoF critical values.

8.4.1.3 Distance-Based Jitter (Addition)

The "Distance-Based Jitter" technique builds on the idea of aligning the spacing of rounded observations with those of a reference distribution while introducing controlled noise to remove discreteness. The rounded observations and the fitted Weibull samples are sorted independently, and the ordered series are then compared point-by-point. Successive differences between neighbouring points in the ordered fitted Weibull distribution are calculated. For each rounded value, the successive differences between neighbouring points are added to the rounded data to remove the discreteness introduced by rounding. The distance may be small enough to smooth out the data at the head of the distribution, but it may also be too large when applied to the data's tail. Regardless, the test was applied to both zero values and the full dataset. Table 8.11 shows the NRMSE results.

Table 8.11: Performance Metrics: NRMSE Jitter by Distance.

	Rounding=0	
Metric	**Sample Size = 100**	**Sample Size = 1000**
NRMSE-All Noise	335.90%	314.47%
NRMSE-Zero Values Noise	324.92%	304.62%

The error magnitude is extremely large. It is three times larger than the mean of the actual data. These results are more significant than what was seen in the "Interval-Based Uniform Jitter" and are too large to be considered within an acceptable error range. As previously seen in earlier sections, NRMSE does not always correspond with the outcomes of the AD and CvM tests. It remains necessary to examine the AD and CvM results regardless. Figure 8.9 shows the fitting results when applying AD to the larger sample size.

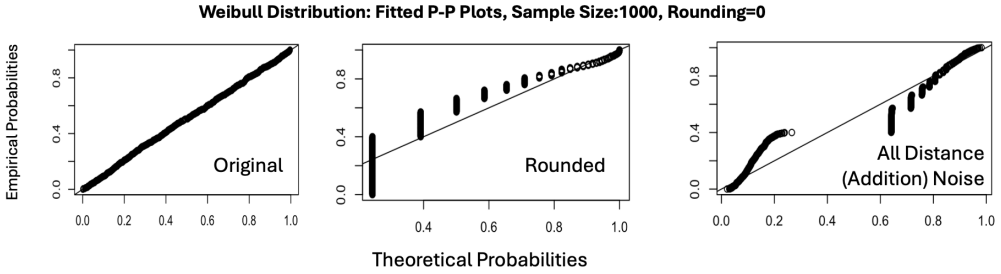


Figure 8.9: AD Distribution Fitting: Jitter by Distance (Addition).

Interestingly, the "Distance-Based Jitter (Addition)" does have a positive impact on the fit to the upper quantile. Still, there is a significant gap around the lower to mid-quantiles, indicating a poor fit. Notably, a technique to resolve the fitting problem could entail using "Distance-Based Jitter (Addition)" to the tail of the data, while applying another technique to the lower and mid-quantiles.

The distortion observed in the "All Distance (Addition) Noise" plot arises because additive distance-based jitter introduces non-uniform addition across the data range. Since the noise magnitude is not scaled relative to the local variance, smaller values are disproportionately inflated, producing a non-linear stretching of the empirical distribution, which results in a deviation from the theoretical Weibull fit in the lower and upper quantiles.

Based on the visual plots, there is no reason to believe this technique will pass the AD GoF test for the larger sample size, but CvM may provide a more favourable outcome. Table 8.12 shows the GoF results.

Table 8.12: AD/CvM GoF: Jitter by Distance Performance Results.

	Rounding=0	
Metric	**Sample Size = 100**	**Sample Size = 1000**
Original AD	0.77	0.93
Rounded AD	22.20	221.56
Jitter AD	11.68	274.46
Zero-Value Jitter AD	11.86	274.56
Original CvM	0.14	0.15
Rounded CvM	4.83	48.28
Jitter CvM	1.26	18.57
Zero-Value Jitter CvM	1.30	18.60

The results indicate that none of the tests for "Distance-Based Jitter" pass the AD or CvM GoF tests. Although the jittered AD values reduced from 22.20 to 11.68, the reduction is not sufficient to pass the AD statistical critical value.

Jittered CvM results show inflated values under integer rounding (1.26 at $n = 100$) and (18.57 at $n = 1000$). Applying this technique only to zero values has no meaningful impact on the overall results, suggesting that separating them is unnecessary. Instead, values nearer the centre of the distribution may be more suitable for further investigation.

8.4.1.4 Pit-Based Even Spaced Jitter

To recap, the "Pit-Based Even Spaced Jitter" technique assigns each observation to a pit and redistributes the points evenly within each pit, with spacing determined by the number of observations and the interval width. A threshold offset based on the rounding precision prevents misallocation to neighbouring pits. The technique helps spread discrete values more uniformly across the continuum, reducing data discreteness. The testing of this technique was applied to the full dataset. Table 8.13 shows the NRMSE results.

Table 8.13: Performance Metrics: Pit-Based Even Spaced Jitter.

	Rounding=0	
Metric	**Sample Size = 100**	**Sample Size = 1000**
NRMSE-All Noise	8.83%	7.26%
NRMSE-Zero Values Noise	NA	NA

It is observed that the results of the NRMSE tests are within the acceptable threshold of 20%. The results indicate that NRMSE is just under 10%, the model's average prediction error, which is approximately one-tenth of the mean of the observed values, reflecting a strong correspondence between predicted and actual values. As seen in previous sub-sections, a strong NRMSE result is not an indication that the new jitter data will conform to the specified fitted distribution model, as has been previously seen. Regardless, GoF tests are conducted on the new jittered dataset. Figure 8.10 shows the results of the AD GoF test on a sample size of one thousand.

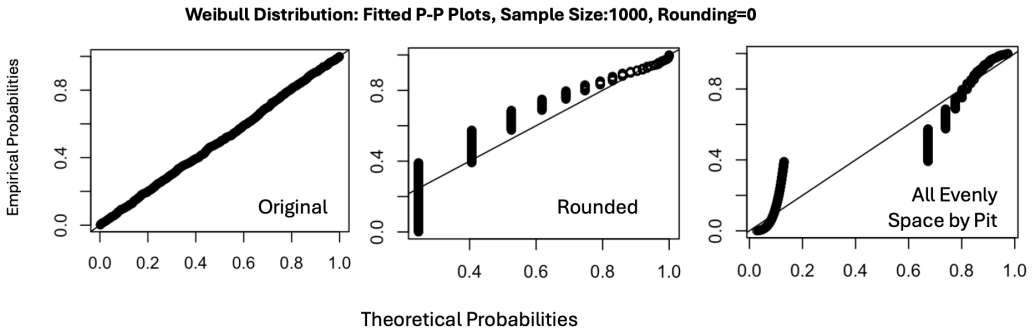


Figure 8.10: AD Distribution Fitting: Pit-Based Even Spaced Jitter.

Figure 8.10 represents a multi-modal type distribution. The curve on the lower quantile is evidence of a mixture model where a single PDF will not be an appropriate fit. It is evident from the P–P plot that there is a clear distinction between the head and tail of the data. Although it is obvious that at a one thousand sample size, this dataset will fail the AD GoF test, it is also observed that the "Pit-Based Even Spaced Jitter" technique does have a positive influence on unrounding the data at the tail. It becomes more obvious when one compares against the "Rounded" P–P plot.

Table 8.14: AD/CvM GoF: Jitter Even Spaced by Pit Performance Results.

	Rounding=0	
Metric	**Sample Size = 100**	**Sample Size = 1000**
Original AD	0.72	0.75
Rounded AD	0.81	Inf
Jitter AD	17.96	185.22
Zero-Value Jitter AD	NA	NA
Original CvM	0.12	0.11
Rounded CvM	1.13	11.13
Jitter CvM	1.05	10.25
Zero-Value Jitter CvM	NA	NA

Turning to the statistical results of the AD and CvM GoF tests for each sample size, Table 8.14 shows that no cells are highlighted in blue, indicating that this technique did not achieve GoF acceptance. Applying this type of jitter, the AD test statistic increases from 0.72 to 17.96 for the sample size of one hundred, and from 0.75 to 185.22 for the one thousand sample size. The CvM statistic rises from 0.12 to 1.05 for the one hundred sample size and from 0.11 to 10.25 for the one thousand sample size.

When comparing the rounded and jittered results, the CvM statistics remain relatively similar, suggesting that this jittering approach has minimal effect on the overall distributional fit. The AD statistic shows substantial diver-

gence between the rounded and jittered data, indicating that the technique substantially alters tail behaviour and leads to poorer alignment with the theoretical distribution. The results are consistent with the P–P plots in Figure 8.10, which show that although the technique partially spreads the jittered points across a continuum, it also introduces deviations in the lower and mid-quantiles.

Despite these distortions, the NRMSE values in Table 8.13 remain relatively low (8.59% for $n = 100$ and 7.2% for $n = 1000$), indicating that, in terms of average predictive error, the jittered datasets still approximate the original values reasonably well. However, the divergence observed in the GoF statistics indicates that small NRMSE values do not imply accurate distributional shape recovery, highlighting the sensitivity of the AD and CvM tests to structural deviations introduced by jittering.

8.4.1.5 Jitter Histogram-Binning

To recap, histogram binning groups continuous data into fixed-width intervals, with each observation assigned to the bin corresponding to its value. Two versions were studied: Version 1 (V1), which uses fixed widths based on rounding precision, and Version 2 (V2), which determines bin boundaries using conventional histogram rules such as Sturges, Scott and FD. Table 8.15 shows the NRMSE results.

Table 8.15: Performance Metrics: Histogram Binning.

	Rounding=0	
Metric	**Sample Size = 100**	**Sample Size = 1000**
V1 Binning NRMSE-All Jitter	46.66%	55.83%
V1 Binning NRMSE-Zero Values Jitter	NA	NA
V2 FD NRMSE-All Jitter	6.37%	15.93%
V2 FD NRMSE-Zero Value Jitter	N/A	N/A
V2 Sturges: NRMSE-All Jitter	46.63%	248.94%
V2 Sturges: NRMSE-Zero Value Jitter	N/A	N/A
V2 Scott NRMSE-All Jitter	48.86%	26.00%
V2 Scott NRMSE-Zero Value Jitter	N/A	N/A

The “V2 FD” binning strategy produced the most consistent performance (averaging approximately just over one-tenth of the mean of the observed values) under zero decimal roundings. By defining bin widths using the FD rule and redistributing values evenly within each bin, the method effectively reduced quantisation distortion.

Binning method exhibited the poorest point-wise recovery, as measured by the highest NRMSE values, on larger sample sizes, with NRMSE exceeding 247%. This indicates severe instability in its bin-width selection under coarse rounding. The remaining binning strategies exhibited similar performance, with both Scott and "V1" binning techniques producing NRMSE values close to half the mean of the observed data on the smaller sample size.

To examine these effects in greater detail, Figure 8.11 presents the binning outcomes for each strategy alongside their corresponding original data points. The "Redistributed" series are the outputs of each rule, not a separate input dataset. As each method did not use the same synthetic dataset, the original values are shown for each test.

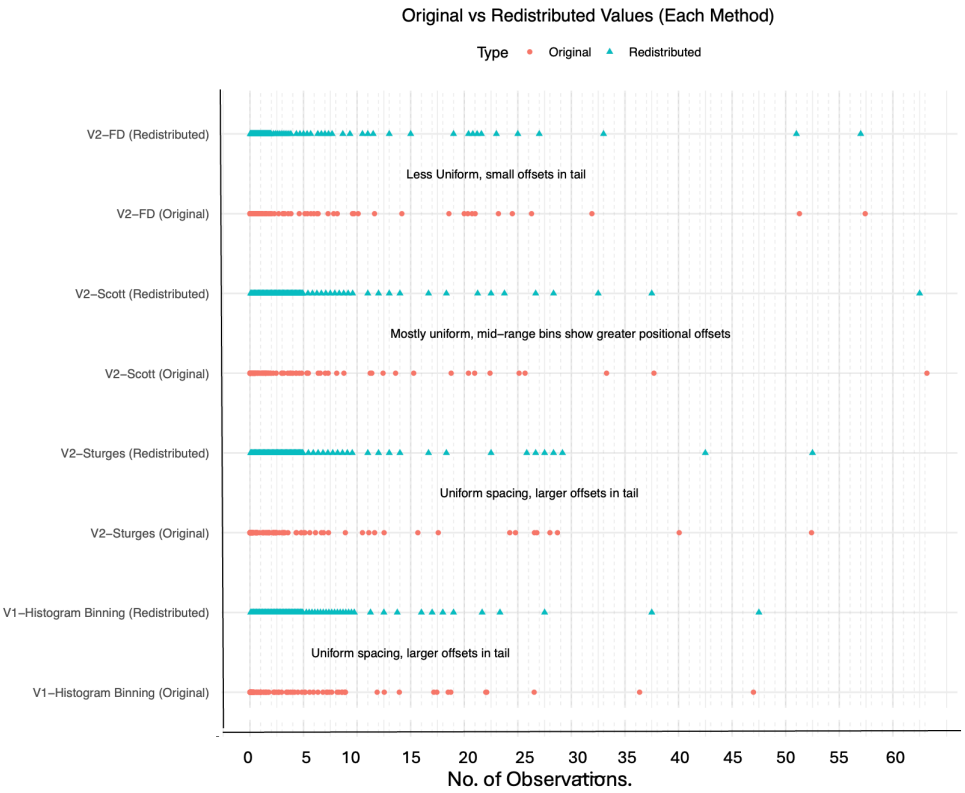


Figure 8.11: Jitter Histogram Binning Strategies (Sample Size:100).

The results of the FD rule ($h = 2, \text{IQR}, n^{-1/3}$) show that the redistributed values are less uniformly spaced than those from the other methods. The spacing varies, reflecting the IQR-based bin widths. Small irregular offsets appear throughout the distribution due to the floating-point bin widths.

The results from Scott ($h = 3.5\,\sigma\,n^{-1/3}$) display mostly uniform alignment between the redistributed values and original values, especially at the head of the dataset. Scott's rule determines bin width based on the overall standard deviation, resulting in wider bins that may not align with the data's rounding precision. As a result, mid-bin positions can shift slightly depending on where the bin edges fall relative to the original rounding intervals, as seen in the chart.

Sturges' rule ($k = \lceil 1 + \log_2 n \rceil$) assumes that the data follows a Normal distribution, and increases the number of bins logarithmically with sample size. For example, if you double your data size, it only adds one additional bin. As Sturges determines only the number of bins rather than bin widths, the bins should be wide and uniform. Uniformity is observed in the redistributed values. However, there are larger offsets in the tail due to the wider bin sizes.

The "V1–Histogram Binning" method uses fixed-width bins determined by the data's rounding precision. As shown in Figure 8.11, this approach produces uniform spacing across most of the data points. Redistributed points at the tail of the data are positioned near the central bin midpoints, reflecting the consistent binning structure used in this method.

Overall, from Figure 8.11, it is hard to identify which method provides optimal uniformity due to the volume of clustered points at the head of the distribution. Table 8.15, indicates that "V1–FD" shows more uniformity with the redistributed points, and the deviations from the original data points are not as large as seen with the other methods in the chart. To statistically confirm this assumption, Table 8.16 shows the statistical results of the binning strategies which are explained below.

Table 8.16: Binning Space Validation

Method	**Discrete Points**	**Bins**	**Mean CV Gap**	**KS Pass Rate**
V1	20	8	0.0000000000	1.00
Sturges	21	8	0.0000000030	1.00
Scott	23	9	0.0000000021	1.00
FD	21	16	0.0000000033	1.00

The rounded data contained up to 23 unique discrete values after quantisation. The number of bins generated by each binning rule ranged from 8 to 16,

depending on the method used. Rounding substantially reduced numerical precision by approximately 92%, leaving only 8 discrete bins under the V1 binning strategy. The "Mean CV Gap" quantifies the average coefficient of variation (CV) of the gaps between redistributed points within each bin; values approaching zero indicate perfect uniform spacing. The "KS Pass Rate" represents the proportion of bins that are statistically consistent with a uniform distribution, with a value of 1 confirming that all bins achieve uniformity. Among the methods tested, "V1 Binning" achieved the lowest bin spacing variability, with a mean CV of 0, which is not consistent with the NRMSE results.

Figure 8.12 shows the fitting results of both the V1 and V2 methods.

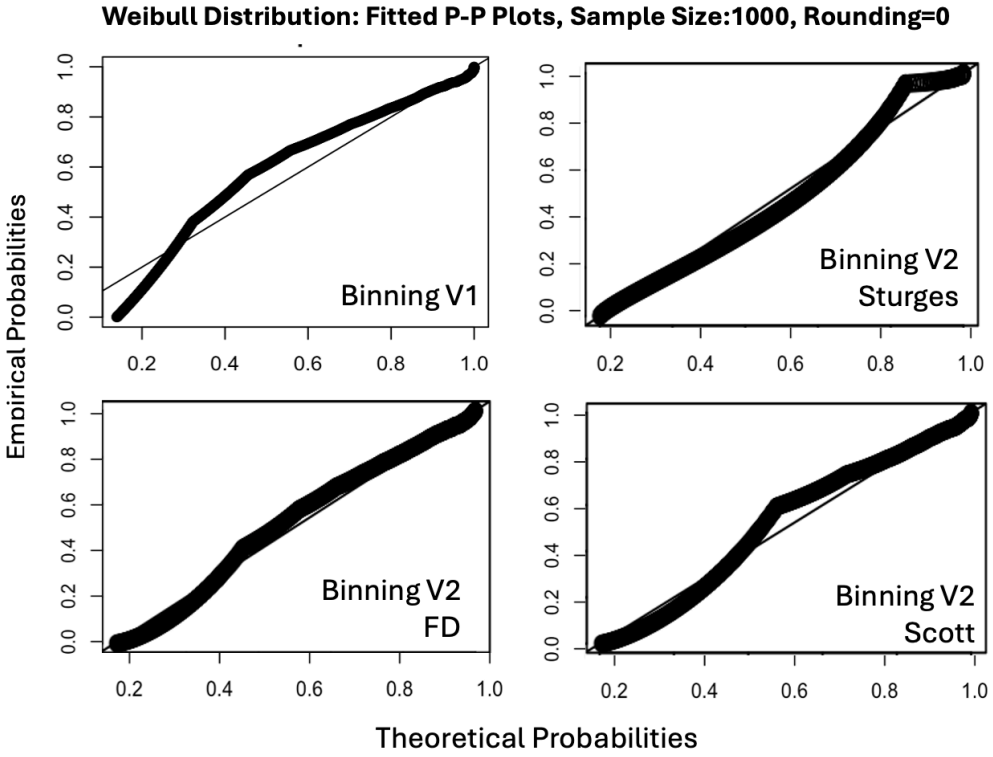


Figure 8.12: AD Distribution Fitting: Histogram-Binning.

Analysing the charts in Figure 8.12, it is evident that the V1 binning method deviates the most from the theoretical Weibull line, particularly in the mid-quantile region, indicating an uneven redistribution of probability mass. Such behaviour reflects the NRMSE (55.83%) results on the larger sample size.

Among the V2 methods, the FD approach exhibits the most uniform P–P

plot alignment, consistent with closer agreement to the AD GoF criterion. In contrast, the Sturges and Scott methods exhibit noticeable curvature in the mid and tail regions, implying under-binning and loss of tail resolution.

Reasoning with the results, these deviations suggest that while FD achieves better overall conformity, the Sturges and Scott rules are limited by their bin-width selection. Scott's rule determines bin width based on the overall standard deviation, resulting in wider bins. As a result, mid-bin positions can shift depending on where the bin edges fall relative to the original rounding intervals. As Sturges determines only the number of bins rather than bin widths, the bins will be wide. Larger offsets in the tail are due to the wider bin sizes.

Table 8.17 shows the statistical results of the AD and CvM GoF tests. To reduce space, only the jittered AD and CvM values are shown.

Table 8.17: AD/CvM GoF:: Histogram-Binning Performance Results.

Rounding = 0		
Method (AD Statistic)	**Sample Size = 100**	**Sample Size = 1000**
V1: Histogram Binning	Inf	Inf
V2: Binning (Sturges)	11.50	74.67
V2: Binning (Scott)	28.85	160.15
V2: Binning (Freedman–Diaconis)	11.04	53.93
Method (CvM Statistic)	**Sample Size = 100**	**Sample Size = 1000**
V1: Histogram Binning	0.30	3.12
V2: Binning (Sturges)	2.33	14.41
V2: Binning (Scott)	6.17	34.10
V2: Binning (Freedman–Diaconis)	2.23	9.94

The results indicate that no method is able to recover the distributional shape of the data using the AD GoF test. However, only the "V1 Binning" method was able to pass the CvM GoF test on the smaller sample size.

8.4.2 General Sampling Methods

8.4.2.1 Rejection Sampling

Rejection sampling was employed to reconstruct unrounded values from quantised datasets by selectively accepting simulated samples that reproduce the original rounded observations. The method iteratively samples candidate points from a specified target distribution and retains only those that, when

rounded, match the observed values, making it computationally intensive. For larger datasets and higher precision, it requires more candidate evaluations per observation, substantially increasing the runtime.

In this study, rejection sampling was applied to the synthetic data.

Table 8.18 shows the NRMSE results for the synthetic dataset where the parameters of the resampled dataset were both fixed and estimated.

Table 8.18: Performance Metrics: NRMSE Rejection Sampling.

	Rounding=0	
Metric	**Sample Size = 100**	**Sample Size = 1000**
NRMSE-Fixed Known All	11.08%	9.28%
NRMSE-Estimated All	7.98%	9.23%

All tests pass the 20% threshold, indicating that the model's average prediction error is approximately one-tenth of the mean of the observed values, reflecting a strong correspondence between predicted and actual values. As shown in previous sections, a low NRMSE does not necessarily imply good distributional fit. We therefore turn to the GoF results.

Figure 8.13 shows the results of the P–P plots obtained from fitting the AD GoF tests at the one thousand sample size.

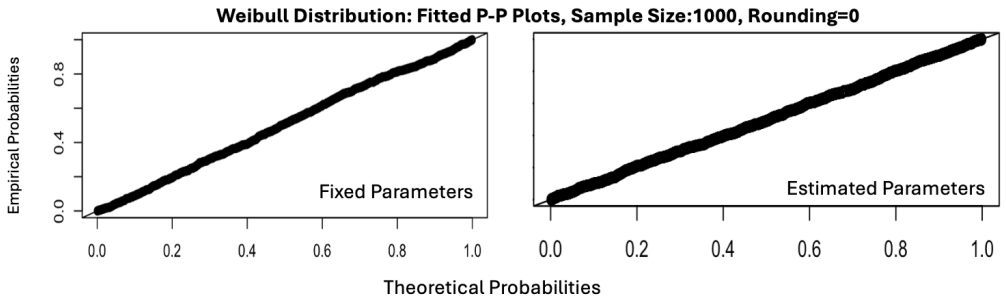


Figure 8.13: AD Distribution Fitting: Weibull Rejection Sampling.

Analysing the plots, there is no deviation from the theoretical distribution, which is an indication that the GoF for the AD test may pass the critical threshold. Motivated by the P–P plot, Table 8.19 shows the statistical results of the tests. These tests are not repeat tests, so in this instance, they are not average values as has been consistently seen throughout other sections in this chapter and other chapters.

Table 8.19: AD/CvM GoF: Rejection Sampling Performance Results.

	Rounding=0	
Metric	**Sample Size = 100**	**Sample Size = 1000**
Original AD	0.98	0.53
Rounded AD	21.88	223.44
Fixed Parameters -Jitter AD	1.24	0.62
Estimated Parameters -Jitter AD	2.49	1.06
Original CvM	0.19	0.08
Rounded CvM	4.74	48.82
Fixed Parameters Jitter CvM	0.24	0.09
Estimated Parameters -Jitter CvM	0.41	0.15

The test results show that all tests pass GoF for both AD and CvM for both sample sizes.

In this instance, a low NRMSE aligns with passing the GoF acceptance criteria. In all other tests conducted in this study, such consistency was not observed.

Overall, all GoF tests meet the acceptance criteria, confirming that "Rejection Sampling" achieves low NRMSE and effectively restores the underlying distributional shape of the rounded data. For the synthetic dataset, both fixed and estimated parameter variants closely matched the theoretical Weibull distribution. These findings demonstrate that the "Rejection Sampling" technique provides a reliable means of reconstructing quantised data, albeit at the cost of considerable computational effort.

8.4.2.2 Inverse Method

Encouraged by the results of the rejection sampling technique and seeking to improve computational efficiency, the "Inverse Method" addresses this limitation by using the CDF and its inverse to map quantised values to their corresponding positions within the target distribution, which eliminates the need for repeated candidate sampling. The introduction of a 0.8 scaling factor although constrained the mapping region, yielding a hybrid approach that allows the bounds to be adjusted for exploratory evaluation against GoF measures.

Log-normal, exponential, and Weibull distributions were used to test this technique as per Table 8.2 and the parameters defined in Table 5.1.

First, the results of this technique are analysed using NRMSE. Table 8.20 shows the results for the three distributions and both sample sizes. For this

test, the method was only applied to the full dataset.

Table 8.20: Performance Metrics: NRMSE Inverse Method.

	Rounding=0	
Distribution	**Sample Size = 100**	**Sample Size = 1000**
Weibull		
NRMSE-All Jitter	5.96%	0.83%
NRMSE-Zero Value Jitter	N/A	N/A
Log-normal		
NRMSE-All Jitter	1.32%	0.88%
NRMSE-Zero Value Jitter	N/A	N/A
Exponential		
NRMSE-All Jitter	28.99%	17.02%
NRMSE-Zero Value Jitter	N/A	N/A

It is observed from the table that NRMSE values in most cases are low ($< 6\%$). As the sample size increases, the magnitude of the error decreases, demonstrating that the inverse method converges rapidly and yields a closer approximation to the true values. However, the exponential distribution does not pass the acceptance threshold at the one hundred sample size and shows a higher-than-average value at the one thousand sample size.

With the noted poor performance of the exponential distribution, further analysis was conducted using a combination of checking distributional skewness and the sensitivity of NRMSE to deviations. In the sample dataset of the one hundred sample size, three observations showed larger than average deviations between the original and recovered values. As an example, original values of 2.42, 3.22, and 3.35 were recovered as 1.72, 2.69, and 2.68, respectively. Although these recovered values remain within plausible adjacent values when rounded to zero decimal places, their deviations are larger than those observed in the data. Across the remaining observations, the average difference between actual and predicted values was 0.0004, whereas these three cases showed an average difference of 0.64, classifying them as outliers. These outliers significantly inflate NRMSE, as discussed previously. Figure 8.14 provides a visual comparison of the actual and predicted points, with notes pinpointing where larger deviations exist.

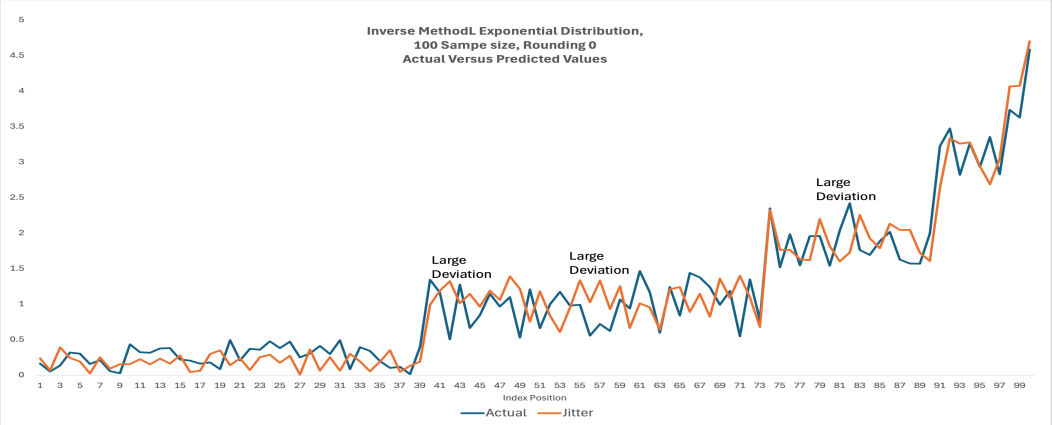


Figure 8.14: Inverse Method: Actual Versus Predicted, 100 Sample Size, Exponential Distribution.

Moving on from NRMSE, Figure 8.15 shows the results of the fitted P–P plots from the AD GoF test of the Weibull, exponential and log-normal distributions.

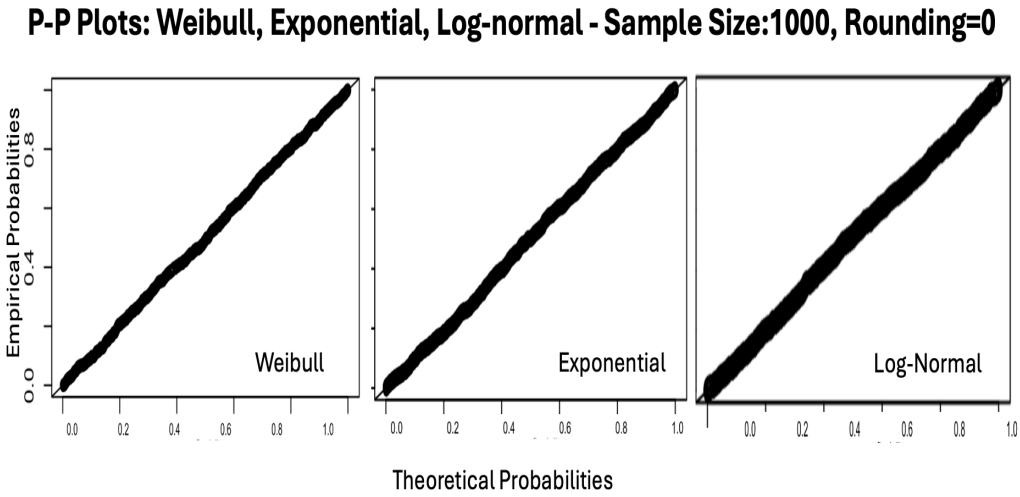


Figure 8.15: Inverse Method: AD Distribution Fitting, Weibull, Exponential, Log-normal.

At a sample size of one thousand for all distributions, the charts show no deviations from the theoretical line, indicating that the tests should pass the AD critical values. Encouraged by these results, Table 8.19 reports the average statistical results from the five repeat tests for AD and CvM

Table 8.21: AD/CvM GoF: Performance Metrics Inverse Method.

	Rounding=0	
Metric	**Sample Size = 100**	**Sample Size = 1000**
Weibull Jitter AD	0.38	0.76
Log-normal Jitter AD	0.46	0.29
Exponential Jitter AD	0.76	0.86
Weibull Jitter CvM	0.06	0.19
Log-normal Jitter CvM	0.07	0.04
Exponential Jitter CvM	0.13	0.11

Note: The test is only applied to the full dataset.

As can be seen by the tests, all distributions pass both AD and CvM GoF tests. The inverse technique successfully reconstructed the underlying data distributions, regardless of the higher NRMSE values in the exponential distribution. All distributions achieved acceptable GoF results, confirming the method's effectiveness in reversing quantisation effects without iterative sampling.

8.5 Discussion

8.5.1 Jitter Methods

8.5.1.1 Gaussian Noise Variants

Gaussian jittering is a straightforward approach to unrounding data. The size of the variance is important for determining at what point the dispersion is too high, effectively increasing the tails of the observations, or too low, reintroducing the discreteness that this method is trying to resolve. Small NRMSE values generally suggest closer agreement to the individual observations. However, they do not imply perfect recovery. As NRMSE is a validation of pointwise recovery, AD and CvM tests are a validation of distributional recovery.

For NRMSE, the majority of tests had an error magnitude $< 10\%$ across all variants of the Group 1 and Group 2 tests, indicating a strong impact on mitigating quantisation effects and recovering the original point values.

For GoF tests, scaling the standard deviation between a 1/3 and a 1/6, thereby concentrating the jitter around the rounded mean, improves the corrective effect for both AD and CvM GoF tests. Any test where the scaling factor of the standard deviation (using non-rounded values) was not $= 1/2$ passed

the CvM GoF. Subtracting rather than adding noise had no impact on AD or CvM results, indicating that the direction of jitter is irrelevant in this context.

As the Gaussian distribution is not aligned with the underlying Weibull distribution, its influence on the GoF tests remains limited.

Overall, the V5 method using a standard deviation of 0.1667 appears to be the most optimal, offering the strongest combination of accurate point recovery and consistent GoF across sample sizes and rounding levels.

8.5.1.2 Interval-Based Uniform Jitter

The "Interval-Based Uniform Jitter" approach constructs its bounds by pairing each rounded value with a neighbouring observation taken from a second synthetic Weibull dataset. Because the Weibull distribution with shape 0.5 has a very heavy right tail, these neighbouring values can become extremely large. As a result, the minimum and maximum interval used to generate uniform noise often becomes excessively wide, leading to jitter values that can be several orders of magnitude larger than the rounded observation (e.g., noise values of 2.95 or 66 where observed for original values of 0.3), producing unrealistic and unstable unrounding behaviour. As both datasets are only similar in distribution parameters, the effects of random generation can significantly impact the results.

NRMSE results show that the prediction errors are more than twice ($> 200\%$) the mean of the actual values, indicating a complete loss of accuracy regardless of the data region it is applied to. This method greatly overestimates the range of admissible continuous values and is therefore unsuitable for unrounding from a pointwise recovery perspective.

From a distributional recovery perspective, applying uniform jitter to the full dataset results in both AD and CvM test statistics falling below their respective critical values. However, when restricted to zero values, GoF fails at larger sample sizes, suggesting that applying the noise to the full dataset is more appropriate.

Adding uniform noise mitigates quantisation and smooths deviations from the theoretical distribution. NRMSE on pointwise recovery is significantly large, which raises a concern. The uniform noise is over-smoothing, which reduces

discreteness and artificially enhances the GoF statistics, without necessarily restoring alignment with the true theoretical distribution.

8.5.1.3 Distance-based Jitter (Addition)

"Distance-Based Jitter (Addition)" adds different levels of noise to the rounded dataset based on a reference distribution.NRMSE is consistently significant ($< 340\%$), indicating a complete loss of pointwise recovery. Both AD and CvM statistics fail the GoF tests across all data regions, indicating an inability to restore distributional properties. The P–P plot showed partial recovery in distributional fit at the upper quantile, but fails to fully de-quantise lower regions of the distribution.

While this additive noise introduces variability, it does not recover the true underlying distributional structure, especially near the lower bounds where quantisation is strongest. Interestingly, while a better fit might be expected near the lower quantiles (due to smaller perturbations) and a degradation toward the upper tail (due to higher variance), the observed results exhibit the opposite. The reversal is likely due to small perturbations failing to separate discrete observations. In contrast, in the upper tail, greater variability allows the added noise to smooth discontinuities more effectively, yielding an improved fit.

8.5.1.4 Pit-Based Even Spaced Jitter

"Pit-Based Even Spaced Jitter" assigns each observation to a pit defined by the rounding precision and redistributes the points evenly within each pit, with spacing determined by the number of observations and the interval width.

While NRMSE values remain low ($< 10\%$), indicating that the jittered data retains numerical proximity to the original values, AD and CvM statistics suggest a substantial loss of shape accuracy in the lower to mid-quantiles as seen in the P–P plots, not preserving distributional fidelity. The step-like effect in the mid quantile increases the AD statistic, confirming that the method is clustering points close to each other, impacting the underlying probabilistic structure.

Overall, the findings demonstrate that while even spacing within pits can

improve the appearance of continuity, it does not restore the statistical characteristics of the original data. The method's low NRMSE but high AD and CvM values illustrate the limitations of relying solely on error-based metrics for assessing distributional recovery. Instead, a comprehensive evaluation using both numerical and distribution-based metrics is required to fully capture the impact of jittering techniques on distributional integrity.

8.5.1.5 Jitter Histogram-Binning

"Jitter Histogram-Binning" groups continuous data into fixed width intervals defined either by rounding precision (V1) or by histogram rules such as Sturges, Scott, and FD.

P–P plots revealed deviations in the upper tail for Sturges, the mid-quantiles for Scott and FD, and a complete divergence for V1 binning. Nevertheless, none of the methods passed the AD GoF tests, with only "V1-Histogram Binning" passing the CvM GoF on the lower sample size, indicating that uniformity does not necessarily correspond to statistical distributional recovery.

For point wise recovery across the four binning strategies, methods such as Sturges and Scott, which depend on sample size and variance, were less robust. However, FD was the only method achieving acceptable pointwise performance (<20%), making it the most reliable method for recovering individual values.

The uniformity diagnostics table showed differences between each approach. V1 binning achieved perfect within-bin spacing (Mean CV Gap = 0), but this uniformity did not recover distributional fit, nor pointwise recovery.

These results highlight a trade-off between pointwise recovery and distributional fit across the binning strategies. The FD rule uses IQR to define the bin widths, capturing only the central 50% of the data. For heavy-tailed distributions, this produces many narrow central bins and larger bin widths at the peripheral edges, leading to irregular jitter spacing and weak distributional recovery. As the sample size increases, Sturges grows the number of bins logarithmically, resulting in large bin widths, causing over-smoothing in the dataset. Scott's rule depends on the sample standard deviation, which is inflated by heavy tails, producing large bins. Together, these behaviours explain the inconsistent NRMSE and GoF performance observed across the three methods.

The poor performance of the V1 binning technique is due to the uniform spacing that is applied to the data. Uniform spacing does not reflect the underlying distribution, leading to a loss of local variability, distortion of the empirical CDF, and poor pointwise recovery as seen with NRMSE (46.66%). The poor performance may also be due to the use of midpoint assignment and half–bin-width spacing, which can displace values within bins away from their plausible adjacent values.

Overall, the findings indicate that using the FD rule is an effective corrective strategy for de-quantising data without reliance on GoF-based validation. However, when GoF assumptions are required, these techniques do not provide distributional recovery.

8.5.2 General Sampling Methods

8.5.2.1 Rejection Sampling

During the development of the "Rejection Sampling" method, multiple iterations were required. Early experiments on the production supply chain dataset revealed that the approach was computationally intensive, often running for several days in R without completion. As rejection sampling relies on iterative looping, repeated sample regeneration, and candidate comparisons, its runtime scales poorly with data size and rounded precision. With more than one million records at a precision of three decimal places, the method proved impractical for large-scale testing on supply chain data. Tests on synthetic data demonstrated that rejection sampling can accurately reconstruct unrounded values, producing near-original data points with minimal deviation. Low NRMSE values and acceptable AD and CvM statistics confirmed that the reconstructed samples retained the statistical properties of the source distribution. However, the approach remains computationally expensive, and its sampling efficiency decreases sharply with increasing data precision.

8.5.2.2 Inverse Method

The "Inverse Method" provides an efficient alternative to "Rejection Sampling" by using the CDF of the underlying distribution. The time to complete was minimal and resolved the long processing times of the "Rejection Sampling" method. While the exponential distribution showed sensitivity to NRMSE due

to the observed outliers in the data, the technique was still able to recover the distributional properties and effectively reverse the effects of quantisation. The log-normal and Weibull distributions performed consistently across all tests, maintaining low NRMSE and acceptable GoF scores.

Introducing the scaling factor moderated the sensitivity of the inverse CDF mapping. Constraining the mapping bounds through scaling allows controlled variability in the effective width of the inverse mapping while preserving distributional consistency.

Overall, the "Inverse Method" achieved comparable accuracy to "Rejection Sampling" while substantially reducing computation time, making it a practical approach for large-scale de-quantisation tasks.

8.5.3 Benchmark

Given the numerous variations applied to unround the data, a benchmark assessment is valuable for evaluating overall performance and trade-offs across synthetic datasets. The assessment applies consistent parameter estimates, sample sizes, rounding levels, and evaluation metrics to enable systematic comparison across methods. The framework employs NRMSE as the primary accuracy metric. NRMSE is computed from the RMSE of the predicted values and normalised by the original data's mean, with an acceptable performance threshold of 20%. Distributional fidelity is assessed using the AD and CvM GoF statistics, with cut-off points (AD = 2.49 and CvM = 0.46). Computational efficiency is considered qualitatively, as execution times are not formally noted. Scalability is assessed with respect to sample size and rounding precision.

Table 8.22 summarises the results of the accuracy and distributional fidelity assessments. In the "Applied" column, "Zero Values" refers to tests that apply the method only to zero values, leaving all other data untouched while still testing on the full dataset. Given NRMSE's sensitivity to outliers, as previously seen with the analysis of the exponential distribution, greater emphasis is placed on GoF results when evaluating performance. Since most real-world datasets contain more than one hundred observations, the interpretation primarily focuses on results from larger sample sizes.

Table 8.22: Benchmark: Unrounding Evaluation Results.

		100			1000		
Test	**Applied**	**NRMSE**	**AD**	**CvM**	**NRMSE**	**AD**	**CvM**
All Gaussian Noise V1 (SD)	All Data	Pass	Fail	Pass	Pass	Fail	Fail
All Gaussian Noise V2 (SD/2)	All Data	Pass	Fail	Fail	Pass	Fail	Fail
All Gaussian Noise V3 (SD/3)	All Data	Pass	Fail	Fail	Pass	Fail	Fail
All Gaussian Noise V4 (SD Subtract Noise)	All Data	Pass	Fail	Fail	Pass	Fail	Fail
All Gaussian Noise V1a (Pit Width)	All Data	Pass	Fail	Fail	Fail	Fail	Pass
All Gaussian Noise V2a (Pit Width/2)	All Data	Pass	Fail	Fail	Pass	Fail	Fail
All Gaussian Noise V3a (Pit Width/3)	All Data	Pass	Pass	Pass	Pass	Fail	Fail
All Gaussian Noise V4a (Pit Width Subtract Noise)	All Data	Pass	Fail	Fail	Fail	Fail	Fail
V5 (Pit Width/6)	All Data	Pass	Pass	Pass	Pass	Fail	Fail
Interval-Based Uniform Jitter	All Data	Fail	Pass	Pass	Fail	Pass	Pass
Interval-Based Uniform Jitter	Zero Values	Fail	Pass	Pass	Fail	Fail	Fail
Distance-Based Jitter	All Data	Fail	Fail	Fail	Fail	Fail	Fail
Distance-Based Jitter	Zero Values	Fail	Fail	Fail	Fail	Fail	Fail
Pit-Based Even Spaced Jitter	All Data	Pass	Fail	Fail	Pass	Fail	Fail
V1 Histogram-Binning Jitter	All Data	Fail	Fail	Pass	Fail	Fail	Fail
V2 FD Histogram-Binning Jitter	All Data	Pass	Fail	Fail	Pass	Fail	Fail
V2 Sturges Histogram-Binning Jitter	All Data	Fail	Fail	Fail	Fail	Fail	Fail
V2 Scott Histogram-Binning Jitter	All Data	Fail	Fail	Fail	Fail	Fail	Fail
Rejection Sampling - Est Par	All Data	Pass	Pass	Pass	Pass	Pass	Pass
Rejection Sampling - Fixed Par	All Data	Pass	Pass	Pass	Pass	Pass	Pass
Inverse Method Weibull	All Data	Pass	Pass	Pass	Pass	Pass	Pass
Inverse Method Log-normal	All Data	Pass	Pass	Pass	Pass	Pass	Pass
Inverse Method Exponential	All Data	Fail	Pass	Pass	Pass	Pass	Pass

The results in Table 8.22 reveal distinct performance patterns across the evaluated unrounding methods. Applying the procedures to either the head or the tail of the data does not indicate that splitting the dataset to capture heavy discreteness improves either pointwise or distributional recovery, suggesting that incorporating all observations yields more stable performance.

In most cases where NRMSE met the acceptable threshold, this occurred consistently across both sample sizes, indicating that the method scales well with increasing data. Conversely, when NRMSE failed to meet the threshold, in most cases the GoF statistics also failed, suggesting that pointwise deviations propagate into broader distortions of the distributional shape. In a small number of cases, the GoF statistics passed while NRMSE failed, highlighting the sensitivity of NRMSE to outliers and its tendency to penalise local deviations.

A closer examination reveals that “Gaussian Noise” variants achieve reasonable pointwise accuracy but fail to preserve distributional fidelity, reflecting their tendency to over-smooth the data. “Histogram-Binning” techniques and

Model-based methods, particularly "Rejection Sampling" and the "Inverse Method" (applied with Weibull, Log-normal, and Exponential distributions), demonstrate the most consistent performance across both accuracy and distributional metrics. The stability arises because these approaches explicitly model the underlying PDF, enabling them to reconstruct the continuous data structure rather than relying on stochastic perturbations alone. By drawing samples directly from the theoretical distribution, these methods preserve the statistical properties of the data even when rounding or quantisation effects are present.

From this study, the insights suggest that similar model-based strategies could be extended to other distributions not covered in this study, providing a general framework for balancing pointwise accuracy with distributional fidelity in data recovery tasks.

8.6 Conclusion

The benchmarking results highlight clear trade-offs between numerical accuracy and distributional fidelity in unrounding synthetic data. While several methods achieve satisfactory NRMSE performance, their inability to reproduce the underlying distributional shape suggests that accurate point estimates alone are insufficient for preserving the distributional shape.

The superior performance of "Rejection Sampling" and the "Inverse Method" across all evaluation criteria underscores the importance of aligning the unrounding process with the assumed distribution, rather than relying solely on jittering or binning approaches. The persistent underperformance of histogram and distance-based methods indicates that they do not generalise well across varying sample sizes.

These results suggest that effective unrounding requires a model-based framework capable of balancing local pointwise accuracy with distributional integrity, particularly when scalability and representativeness are important for downstream analysis.

Future work should consider adaptive, quantile-guided application of the local binning and jittering techniques, allowing for targeted correction where distributional distortion is greatest, rather than relying on fixed partitioning.

A deeper investigation of mixture models with rounded datasets could be investigated. Preliminary results suggest ten clusters, but their structural meaning and contribution to quantisation remain unknown. Additional research will be required to characterise these clusters and assess whether cluster-specific unrounding strategies may further improve distributional recovery.

CHAPTER 9

Interval-Based Approach for Estimation

9.1 Introduction

Chapter 8 introduced reconstruction-based techniques for generating plausible continuous values from rounded observations. These methods focused primarily on approximating individual latent observations and restoring smoothness to the empirical distribution. Although reconstruction improved the representation of the underlying distributional structure, the exact original values remained fundamentally unrecoverable due to the ambiguity introduced by rounding. Building on these findings, Chapter 9 shifts focus from reconstructing individual observations to recovering the parameters of the underlying distribution directly. Rather than attempting to estimate specific latent values, the emphasis is placed on recovering the statistical characteristics of the generating process itself. This distinction is important because, in many practical applications, accurate estimation of distributional parameters is more valuable than recovering individual observations.

In many real-world analytical processes, the precise underlying values may not be important. The goal may be to understand the behaviour of the distribution that generated the data. For example, in reliability analysis, queuing systems

or in large-scale simulation testing, the interest may lie in estimating quantities such as the mean lifetime of a failure-prone event or the parameters associated with its underlying distribution. Estimating the mean lifetime or the underlying parameters is important because these parameters determine the behaviour and performance characteristics of the system being modelled. They allow researchers to assess reliability, predict future behaviour, quantify uncertainty, and make informed decisions. For failure-prone events, for instance, these parameters determine how long events typically persist, how frequently failures occur, and how heavy-tailed the distribution of the message lifetime is. All this information is important for system design, capacity planning and risk assessment. Recovering parameter estimates supports reliable simulation when the individual latent message cannot be observed directly due to rounding or other forms of data distortion.

Attempting to recover individual latent values may be infeasible, unnecessary, or even undesirable, especially when rounding introduces ambiguity. One can never truly recover the unrounded value, as the original value is lost through rounding. Accurate parameter estimation enables one to reconstruct the underlying distributional form, supporting simulation and allowing probabilistic inference about the underlying process.

To recover parameter estimates masked by rounding, this approach is based on a mathematical technique called integration, also known as "quadrature", previously discussed in Chapter 2 [85]. The distributional form of the obscured data can be recovered by evaluating the likelihood using the AUC associated with each rounding interval.

To get the exact AUC $f(x)$ from a to b, one can use an antiderivative $F(x)$:

$$\int_a^b f(x)\,dx = F(b) - F(a).$$

Some functions do not have closed-form antiderivatives. An example is:

$$\sin(x^2)$$

In such cases, the exact value of the definite integral cannot be obtained, as no elementary functions, be it polynomials, exponentials, logarithms, etc., return its exact value. Instead, its value must be approximated numerically,

for example, by partitioning the area into subintervals and summing the terms to estimate the AUC. Figure 9.1 shows the PDF of a Weibull distribution, a histogram of its rounded values, and an illustration of how numerical integration can compute the AUC for each interval of the data.

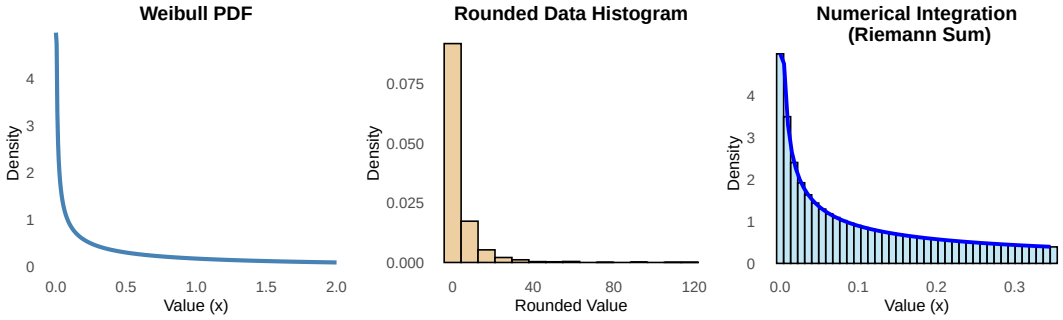


Figure 9.1: Example: Numerical Integration.

From Figure 9.1, the sub-intervals are all equal, although in practice they may differ depending on the behaviour of the function. By summing the narrow rectangles, one can numerically approximate the AUC over each interval, which can then be used to recover distributional parameters obscured by rounding.

In likelihood-based parameter estimation methods such as MLE, observations are typically assumed to be exact values. Under this assumption, the likelihood contribution of an observation is obtained by evaluating the probability density function at that point:

$$L(\theta \mid x_i) = f_\theta(x_i)$$

However, rounded observations do not represent exact measurements, but intervals of possible latent values. If a rounded observation is recorded as x_i, then its true value lies within the interval:

$$X_i \in \left[x_i - \frac{h}{2},\, x_i + \frac{h}{2}\right) \tag{9.1}$$

where h denotes the rounding precision. Treating the rounded value as an exact point therefore produces a misspecified likelihood and may lead to biased parameter estimates.

Instead, the likelihood contribution should be the probability that the latent observation lies within the rounding interval:

$$P(L_i \leq X_i < U_i) = \int_{L_i}^{U_i} f_\theta(x)\, dx$$

Using the cumulative distribution function, this interval probability becomes:

$$P(L_i \leq X_i < U_i) = F_\theta(U_i) - F_\theta(L_i)$$

Thus, the likelihood is constructed from interval probabilities rather than pointwise density evaluations. This allows the estimation process to account explicitly for the rounding mechanism without attempting to reconstruct individual latent observations.

This chapter proposes a principled approach to modelling quantised data by treating observations as interval-censored rather than exact values, and adapting likelihood-based estimation accordingly. The chapter addresses the following questions:

1. Can this "Interval-Based" method recover the true parameters when data are rounded?
2. Does the recovery of true parameter estimates using this "Interval-Based" method hold across different distributions (e.g., Weibull, log-normal and exponential)?

To test this "Interval-Based" method, Weibull, log-normal and exponential distributions will be synthetically created using the parameters and sample sizes defined in Table 5.1. Tests will be repeated ten times to assess the impact at different samplings. Jittered scatter plots will show the return parameter estimates from the ten repeat tests using sample sizes of one hundred and one thousand, rounded to zero decimal places.

9.2 Data Overview and Limitations

The synthetic data used in this chapter, along with its limitations, have been previously discussed in Chapter 5.2.

9.3 Methods

For this study, the methodology is implemented as a set of R functions, and the overall process is summarised in Figure 9.2, which showcases the Weibull distribution as an example.

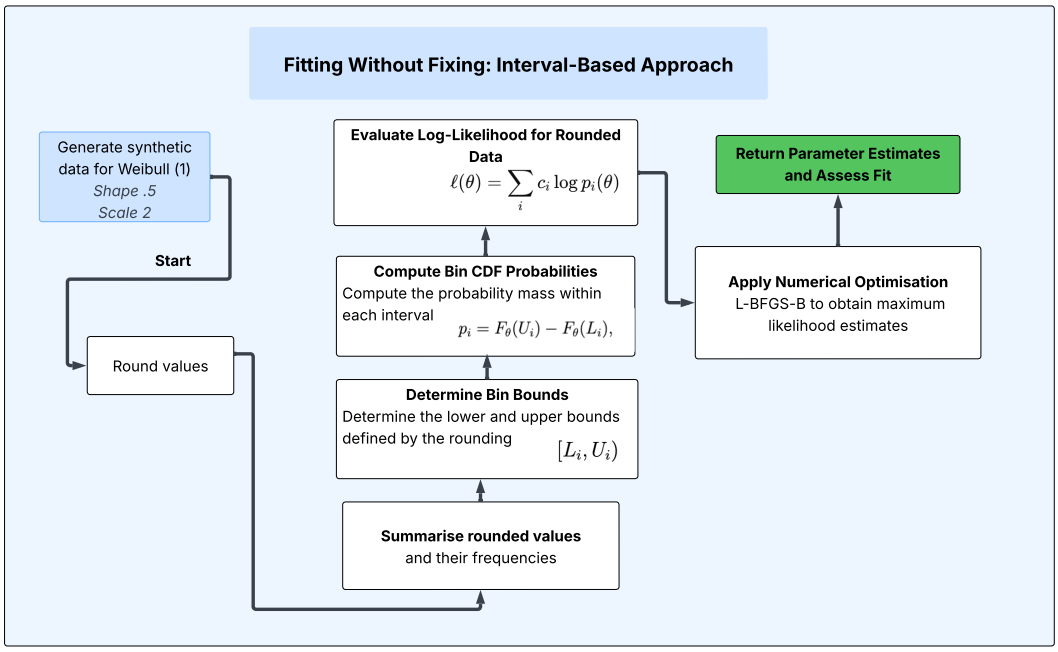


Figure 9.2: Flow Diagram: Interval-Based Approach.

A synthetic dataset of size n is first generated from a specified distribution and rounded according to the chosen precision. Counts of distinct rounded values are recorded. The bin bounds for each rounded observation x_r are determined by the rounding precision as shown in equation 9.1. The probability mass within each interval is computed using the CDF. The log-likelihood is evaluated by weighting these probabilities by the observed frequencies of each rounded value. Numerical optimisation then obtains maximum-likelihood estimates of the distribution's parameters.

The probability mass within each interval is computed using the CDF. The log-likelihood is then evaluated by weighting these probabilities by the observed frequencies of each rounded value. Numerical optimisation is used to obtain maximum-likelihood estimates of the distribution parameters.

For standard MLE, each observation is treated as an exact value. The likelihood contribution for an observation x_i is therefore based on the density evaluated

at that point:

$$f_\theta(x_i)$$

However, when observations are rounded, the recorded value x_i is not exact. Instead, it represents a range of possible latent values. If the rounded value corresponds to the interval $[L_i, U_i)$, then the likelihood contribution should be the probability that the true value lies within that interval:

$$P(L_i \leq X_i < U_i) = \int_{L_i}^{U_i} f_\theta(x)\, dx$$

Using the cumulative distribution function, this interval probability can be written as:

$$P(L_i \leq X_i < U_i) = F_\theta(U_i) - F_\theta(L_i)$$

Thus, the likelihood is constructed from interval probabilities rather than pointwise density values. For repeated rounded observations, each interval probability is weighted by the number of observations falling in that interval.

As multiple values may round to the same observation, to avoid duplicated points, the likelihood uses the CDF differences $F_\theta(U_i) - F_\theta(L_i)$ weighted by the observed frequencies of each rounded value. For example, consider Table 9.1 where observations have been rounded to the same value. Each rounded value corresponds to an interval $[L_i, U_i)$ in the continuous space, and the log-likelihood contribution is weighted by how many times that value appears in the data. The overall log-likelihood becomes the sum of the log-likelihood terms listed in Table 9.1.

Table 9.1: Interval likelihood for repeated rounded value.

Example True Value	**Rounded Value x_r**	**Count**	**Interval $[L, U)$**	**Log-likelihood Term**
1.23	1	3	$[0.5,\ 1.5)$	$3 \log p([0.5, 1.5))$
2.10	2	1	$[1.5,\ 2.5)$	$1 \log p([1.5, 2.5))$
3.42	3	4	$[2.5,\ 3.5)$	$4 \log p([2.5, 3.5))$

Note: Intervals are half-open $[L, U)$, meaning they include the lower boundary but exclude the upper boundary. For example, values such as 1.4 fall inside $[0.5, 1.5)$ and round to 1, whereas a value equal to the upper boundary (e.g. 1.5) would round to 2 and therefore belongs to the next bin, ensuring bins do not overlap.

The likelihood contribution for each point is the probability mass of the model within its rounding interval, thereby avoiding density distortion caused by rounding, especially near boundaries such as zero. The interval likelihood is exact because the CDF provides the integrated probability over each interval.

The interval likelihood is exact because the CDF provides the integrated probability over each interval.

The interval-based estimation procedure used throughout this chapter can be summarised as follows:

1. Identify the quantisation level or rounding precision h.
2. Define interval bounds for each rounded observation:
$$[L_i, U_i) = \left[x_i - \frac{h}{2}, x_i + \frac{h}{2}\right)$$
3. Construct the likelihood using interval probabilities computed from the CDF:
$$p_i = F_\theta(U_i) - F_\theta(L_i)$$
4. Weight the interval probabilities by the observed frequencies of the rounded values.
5. Estimate the distribution parameters by maximising the resulting log-likelihood.

Table 9.2 illustrates an example from a repeat test of the likelihood contributions for the rounded Weibull observations, where sample size n = 1000.

Table 9.2: Likelihood Contributions: Rounded Weibull Observations.

Rounded Value	**Count**	**Lower**	**Upper**	**Prob**	**Log Prob**	**Weighted Log Prob**
0	120	0.0	0.5	0.114	-2.169	-260.329
1	343	0.5	1.5	0.352	-1.043	-357.875
2	269	1.5	2.5	0.274	-1.293	-348.043
3	163	2.5	3.5	0.152	-1.882	-306.904
4	69	3.5	4.5	0.068	-2.680	-184.978
5	26	4.5	5.5	0.026	-3.639	-94.625
6	6	5.5	6.5	0.008	-4.732	-28.395
7	3	6.5	7.5	0.002	-5.943	-17.831
8	1	7.5	8.5	0.000	-7.261	-7.261

For each distinct rounded value, the probability density is integrated over its rounding interval (Upper and lower bounds) to obtain the probability mass contributing to the likelihood. Weighted by the frequency count of each rounded value, these log-likelihood components are summed to estimate the Weibull shape and scale parameters. Distribution parameters are estimated directly from the observed data without alteration or imputation. It is not an approximation; it is exact because the CDF integrates the PDF exactly. The final step of the method is to use numerical optimisation to optimise the log likelihood to estimate the parameters by MLE.

9.3.1 Assumptions of the Interval-Based Approach

The "Interval-Based" method relies on several assumptions regarding the quantised observations and the underlying data-generating process.

First, the method assumes that the quantisation boundaries are known or can be inferred correctly from the rounding precision h. Incorrect specification of the interval widths would distort the probability mass assigned to each rounded observation.

Second, the approach assumes that observations are independent and identically distributed (i.i.d.), consistent with the assumptions underlying standard maximum-likelihood estimation procedures.

Third, the method assumes that the underlying continuous distribution allocates probability mass smoothly across each rounding interval. The approach does not assume a uniform distribution of latent values within each interval; instead, the probability contribution is determined by integrating the fitted PDF across the interval bounds using the corresponding CDF differences.

Finally, the method assumes that the selected parametric distribution is an appropriate representation of the underlying continuous process. Severe model misspecification may still produce inaccurate parameter estimates even when rounding effects are explicitly incorporated.

9.4 Results

Before exploring the results, Figure 9.3 shows the theoretical CDFs and the rounded observed CDFs of the Weibull, log-normal, and exponential distribu-

tions, rounded to zero decimal places, with a sample size of one thousand.

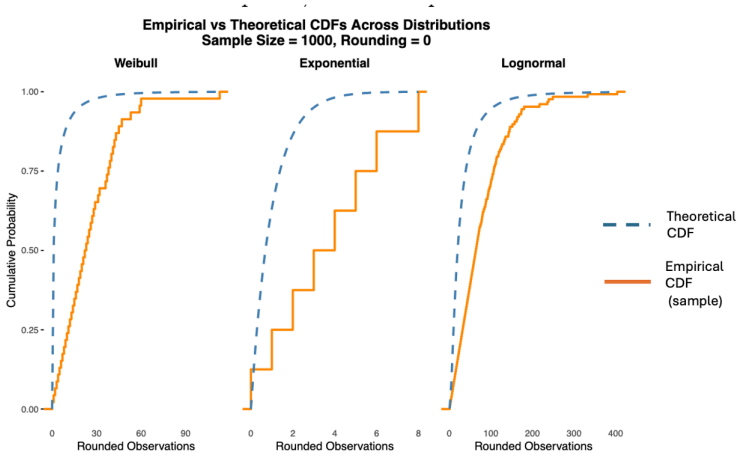


Figure 9.3: Empirical and Theoretical CDFs.

The orange curves in the chart are the discrete cumulative probabilities, while the dashed blue lines denote the theoretical model. The charts show how the ECDF derived from the rounded observations does not align well with the fitted theoretical CDFs for each distribution. Rounding collapses continuous values into discrete bins. The observed ECDF in orange becomes a step function. A closer alignment between the stepwise ECDF and the smooth theoretical CDF is impossible without additional information that has been lost through rounding.

The method suggested does not attempt to reconstruct pointwise latent values; instead, it models the rounding mechanism and computes the likelihood by integrating the density over each rounding interval. Producing exact interval probabilities supports the estimation of distributional parameters without altering or smoothing the data. Figure 9.4 shows the probability mass integrated within the rounded intervals ($p_i = F_\theta(U_i) - F_\theta(L_i)$) with the fitted exponential PDF. To prevent unnecessary repetition, only the exponential distribution is presented.

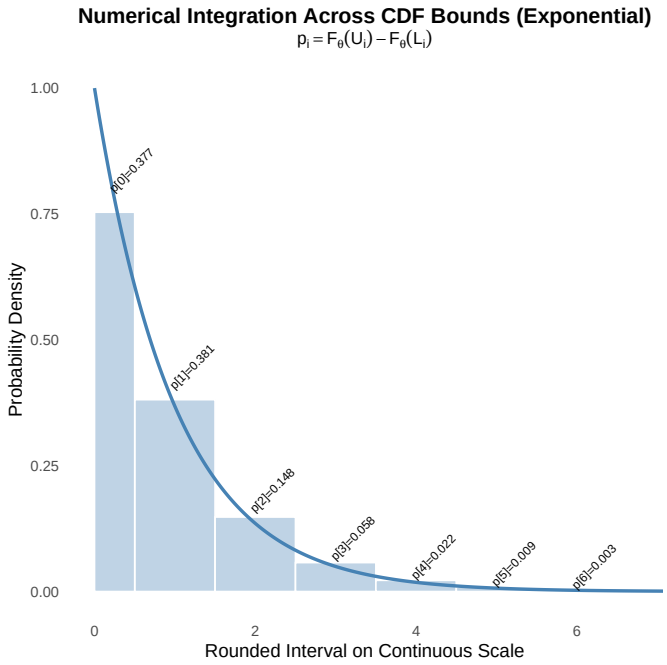
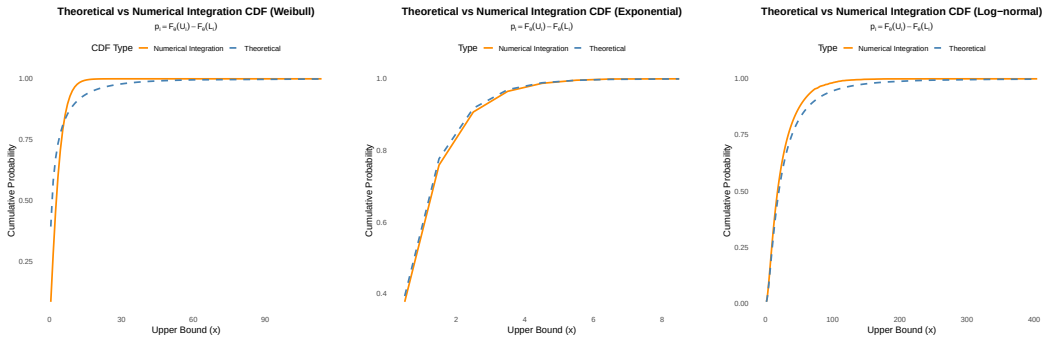


Figure 9.4: Numerical integration across CDF Bounds: Exponential Distribution.

The blue curve represents the continuous density estimated from the rounded data. The bars do not form a histogram. Each bar corresponds to the probability mass that the fitted model assigns to a specific rounding interval. For example, the leftmost bar indicates that 37.7% of the fitted exponential distribution lies within the interval $[0, 1)$. The figure illustrates how rounding converts continuous density into discrete bins, but numerical integration restores the underlying probability structure.

The same procedure applies to the Weibull and log-normal distributions. For each model, the interval probabilities $p_i = F_\theta(U_i) - F_\theta(L_i)$ accumulate to form a CDF, which can then be compared directly with the theoretical CDF implied by the estimated parameters. Of the ten repeat tests, Figure 9.5 presents a comparison of one repeat test across all three distributions.



(a) Weibull (b) Exponential (c) Log-normal

Figure 9.5: Comparison between theoretical and numerical integration CDFs for Weibull, exponential, and log-normal distributions. Each plot shows the alignment between the theoretical cumulative distribution function and the CDF obtained through the application of the "Interval-Based" method over the rounded probability bounds.

The estimated CDF is coloured orange, while the theoretical CDF is a dashed blue line. For the Weibull distribution, the numerically integrated CDF is close to the true CDF; however, there are small discrepancies at the extreme left tail. For the exponential distribution, the numerically integrated CDF and the true CDF are nearly identical and perform consistently well across the whole CDF. The log-normal distribution behaves similarly to the Weibull distribution. The comparison shows that the numerically integrated CDFs are almost identical to the theoretical CDFs across all three distributions, which shows that modelling the rounding behaviour explicitly and using interval probabilities $p_i = F_\theta(U_i) - F_\theta(L_i)$ yields parameter estimates whose implied CDF accurately reproduces the true distributional form despite the loss of pointwise information caused by rounding.

Encouraged by the results from Figure 9.5, Figure 9.6 shows jittered scatter plots comparing estimated and true parameters across the ten repeated tests.

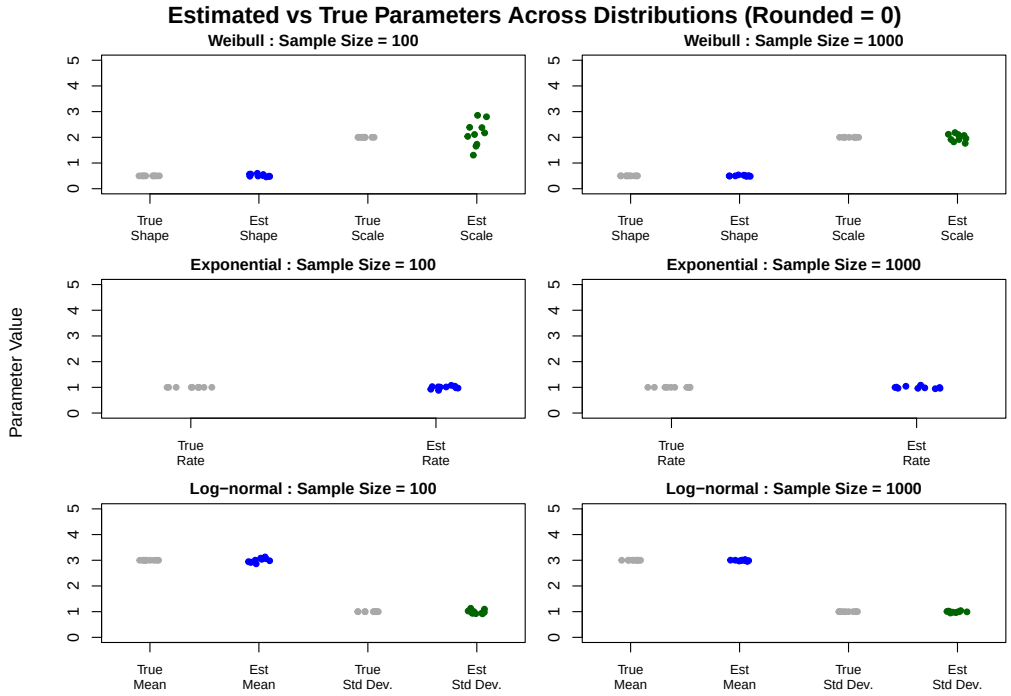


Figure 9.6: Numerical Integration: Estimated Parameter Values.

The scatter plots show that numerical integration reliably recovers the true distributional parameters across all three distributions, even when the observations are rounded. For the Weibull distribution, the shape parameter is recovered with no variability, while the scale parameter exhibits some dispersion at $n = 100$ that reduces at $n = 1000$, reflecting improved stability with increased sample size. For the exponential and log-normal distributions, the estimated parameters cluster tightly around their original values, indicating that the method consistently produces close parameter estimates.

Overall, the results confirm that a simplified numerical integration technique using CDF bounds effectively reverses the information loss due to rounding at the parameter level, enabling accurate recovery of the underlying distribution even when pointwise recovery of latent values is not recoverable.

If we compare the "Interval-Based" method with MLE applied directly to the rounded data (as shown in Figure 9.7, copied from Figure 6.10), a clear difference is seen in the stability and accuracy of the parameter estimates.

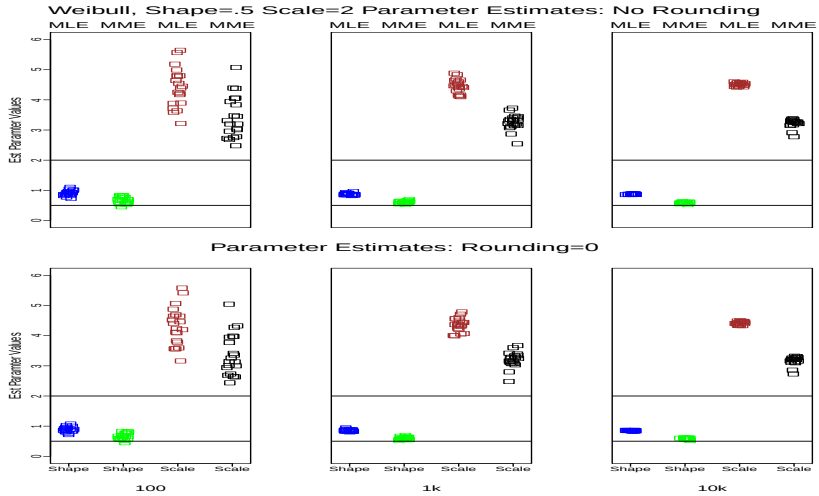


Figure 9.7: Weibull: MLE-MME Parameter Estimation.

In the MLE/MME plots, the estimated Weibull scale parameter shows larger variability, extending up to values near 6 for sample sizes of one hundred and one thousand, whereas for numerical-integration estimates (Figure 9.6), the estimates remain concentrated around the true value, with a narrower vertical spread and an upper range of approximately 5.

The "Interval-Based" method avoids this issue by correctly accounting for the probability mass within each rounding interval, rather than relying on pointwise likelihood contributions. As a result, it retains much more of the original information in the rounded data, leading to parameter estimates that show far less variability and align much more closely with the true underlying values.

9.5 Discussion

This interval-based approach is particularly beneficial when quantisation is significant relative to the variability of the underlying data. In cases where rounding precision is coarse, standard continuous likelihood methods may produce biased parameter estimates or unstable GoF behaviour due to the loss of fine-grained information. By modelling rounded observations as intervals rather than exact values, the method explicitly incorporates quantisation into the estimation process. However, when quantisation effects are minimal relative to the spread of the data, conventional continuous estimation methods may remain sufficient.

Rounding causes a loss of information about the exact location of each observation, replacing a continuous latent value with an interval of possible values. As a result, the empirical CDF becomes a coarse step function that cannot match the smooth theoretical CDF, regardless of sample size. The mismatch reflects information loss and cannot be repaired by smoothing or by increasing precision when the rounded values remain unchanged.

Methods such as "Rejection Sampling" in Chapter 8 are effective for generating plausible latent point values, but they are not appropriate in this study because the goal is no longer pointwise reconstruction. Instead, the task is to recover the underlying distributional parameters themselves. Parameter estimation from rounded data, therefore, requires a likelihood that correctly reflects the interval-censored nature of the observations.

The inflation of the variance in MLE and MME occurs because naïve likelihood estimation treats rounded observations as if they were exact point values. Such behaviour leads to biased and unstable scale estimates, particularly when large portions of the distribution are collapsed into a small number of rounded buckets.

The proposed method achieves this by integrating the model's density over each rounding interval, which produces an exact likelihood contribution for each observation. The resulting integrated CDFs closely match the theoretical CDFs across Weibull, exponential, and log-normal distributions, with only minor tail discrepancies.

The proposed method reliably recovers the true parameters across all distributions, with precision improving at larger sample sizes. In contrast, standard MLE applied directly to rounded values shows inflated variance, particularly for the Weibull scale parameter, because the likelihood is misspecified when rounding is ignored.

The results consistently show that the "Interval-Based" method improves parameter estimation stability and produces fitted CDFs that more closely align with the theoretical distributions than direct fitting methods applied to rounded observations. Compared with the direct MLE/MME estimates shown in Figure 9.7, the interval-based estimates exhibit reduced variability and tighter concentration around the true parameter values. The method also

demonstrates improved GoF behaviour by explicitly modelling the probability mass contained within each rounding interval rather than treating rounded observations as exact point values.

Overall, the evidence shows that although rounding prevents recovery of individual latent values, the underlying distributional parameters remain identifiable, and numerical integration provides a principled and accurate method for recovering them. One can also use the density values based on the distribution parameter, without the need for numerical integration.

The interval-based estimation approach requires interval probability calculations using repeated CDF evaluations rather than direct evaluation of the likelihood at individual points. Although this introduces additional computational steps compared with standard continuous likelihood estimation, the computational overhead remained manageable for the datasets examined in this research. Furthermore, repeated rounded observations can be grouped into shared intervals, reducing redundant likelihood calculations and improving scalability for highly quantised datasets.

9.6 Conclusion

The "Interval-Based" method treats each observed data point as representing an interval rather than an exact value. Instead of assuming that observations are measured with infinite precision, the method integrates the probability density over a small area surrounding each rounded value, thereby capturing all plausible true values that could have produced that observation. The likelihood is not evaluated at a single point, inference remains driven by the underlying PDF. For each observation, the contribution to the likelihood is obtained by integrating the density implied by the candidate parameters over the rounding interval. As a result, parameter estimates are informed by how the density allocates probability mass across these intervals, ensuring that differences in scale, shape, and tail behaviour of the underlying distribution are fully reflected in the inference.

This method performs well because, when an observed value is zero, the model does not infer that the latent value must be exactly zero, it considers the probability mass within the interval around zero by integrating the density across that range.

The study has demonstrated that parameter recovery is robust across distributions, with near-optimal agreement for exponential and log-normal distributions and minimal tail discrepancies for the Weibull distribution. Although rounding removes fine-grained data, it does not prevent accurate inference about the distribution. These insights are essential for any application in which data are limited by precision or measurement constraints, including reliability modelling, queueing systems, simulation-based inference, and modern high-dimensional modelling settings where latent states cannot be recovered directly.

For future work, as the study in this chapter has demonstrated the ability to estimate the parameters without fitting errors (avoiding log(0), adding a constant), this "Interval-Based" method can be applied to estimate the parameters of the distributions for each technique in Chapter 8 and then apply the jitter techniques to unround the data.

CHAPTER 10

Conclusions

In this chapter, the work presented in this manuscript is reviewed and summarised, and suggestions for possible future work are provided.

10.1 Introduction

Preliminary research was conducted to identify ways to optimise queuing systems. The research question addressed was: How can queue behaviour in distributed EDI-driven supply chain environments be modelled and optimised to improve message throughput? Early analysis revealed that the timestamps in the production log data were quantised, causing continuous events to collapse into discrete values. Quantisation prevented accurate continuous distribution modelling and obscured interarrival and service time patterns that drive different queue behaviours. These data issues changed the trajectory of the research, shifting the focus towards understanding the effects of quantisation, developing methods to mitigate its impact on statistical modelling, and creating a systematic approach for modelling and interpreting the behaviour of these high-volume heterogeneous EDI messages.

My research was motivated by persistent performance bottlenecks in queuing applications. Processing these EDI messages, where millions of messages expanded into billions of heterogeneous jobs, caused throttling, message retries, and downstream congestion problems.

Summary of Problems Addressed

A core problem addressed in this research was the lack of a systematic approach in the peer-reviewed literature for modelling high-volume EDI messages, including their associated characteristics, with particular attention to message independence, structural and behavioural characteristics, and the exponential growth patterns exhibited by certain EDI message types.

Another core problem studied in this research was the impact of quantisation on modelling EDI message interarrival and service times. The originally continuous events could no longer be accurately represented, invalidating key assumptions of continuous distribution models and preventing reliable performance testing, simulation, and capacity-planning analyses.

A further challenge studied in this research was the impact of rounding on parameter estimation and GoF methods. The study examined how different parametric methods respond to rounded observational data and the effect of rounding on GoF assumptions.

Methodological Context

The methodological approach in this research combined large-scale enterprise log analysis, distribution modelling, and statistical analysis of message arrival processes within distributed EDI systems. Log data from a production environment was used to reconstruct message flows, identify bottlenecks, and quantify interarrival and service time characteristics. The data was analysed using both parametric and non-parametric techniques, drawing on common continuous distributions in queuing theory, including the log-normal, Weibull, and exponential distributions. GoF assessments, such as AD and CvM tests, were applied to evaluate the suitability of the fitted models against the different continuous distributions.

Due to quantisation in the production logs timestamps, which prevented accurate modelling of the underlying continuous distributions, synthetic data was generated across a range of sample sizes (e.g., 100, 1k, 10k, and 100k observations) and rounded to enable controlled experimentation and comparative analysis under rounded and non-rounded conditions. In our real EDI dataset, the extremely large number of observations made the effects of quantisation particularly pronounced, as repeated rounded values accumulated into

highly visible pits, thereby motivating the need to study quantisation effects systematically across different sample sizes.

10.2 Summary of Key Findings and Contributions

In this section, I review and summarise the work presented in this manuscript.

1. A systematic modelling framework for high-volume EDI message flows (Chapter 3 & Chapter 4).

The manuscript developed the first structured and systematic framework for analysing and modelling EDI message processing behaviour in distributed supply-chain environments. No existing peer-reviewed work provides methods for modelling high-volume EDI message flows, including their burstiness and the associated correlation patterns, making this research novel. The framework integrates several original components like message classification using structural features (including split behaviour, size, and category), temporal segmentation into normal and busy periods, and the treatment of structural drivers, such as split count, bundling, scheduling, and map count, that were identified in the manuscript as previously undocumented sources of correlation and burstiness.

A key finding of this work is that neither parametric nor non-parametric models fully capture the behaviour of the EDI processing times, with quantisation acting as a major contributing factor. However, other factors also include distinct groups of service times identified in the histograms, each with its own characteristics and density patterns. These groups likely correspond to different underlying message types, although their exact origins are not yet fully understood. Parametric models tend to capture the tail behaviour, while non-parametric methods perform better in the head. A new hybrid modelling strategy that combines different segments of the data under distinct categorisations provides a useful framework for understanding message variability, burstiness, non-stationarity, and the structural drivers of queue behaviour in large-scale supply chain messaging systems.

2. Empirical characterisation of data quality constraints affecting queue modelling (Chapter 3 & Chapter 4).

The manuscript demonstrates that the typical assumptions required for queue modelling, especially independence and continuous distributions, are violated in real EDI production logs. While quantisation and rounding are well understood in general statistical literature, their interaction with EDI-specific behaviours, particularly bursty messages logged independently, is not documented in peer-reviewed literature, making this research both novel and necessary to address a clear gap in the existing literature. These types of messages contribute to broken independence and distorted interarrival and service time behaviours. Production logs exhibited millisecond-level quantisation, loss of temporal ordering, and rounding-induced boundary artefacts, including zeros and distributional support violations. These data quality constraints, rather than the choice of statistical models, are often the source of modelling challenges in queue simulation and parameter estimation for EDI workloads.

3. Convergence behaviour and GoF instability (Chapter 5)
Many practitioners come across fitting errors when using MLE or MME with the R fitdistrplus package. A common workaround is to insert arbitrary constants (e.g., replacing zeroes with 1) without understanding the underlying cause of the issue. The contribution of this chapter is to offer a systematic approach as to why these errors occur, rather than simply trying to avoid them.

No peer-reviewed literature was found that tries to provoke and characterise these fitting failures. The manuscript intentionally introduces problematic conditions, such as rounding, zero or negative values, and extreme parameter settings, to explore their impact on convergence and GoF testing. Rather than working around the issues, the analysis deliberately breaks the model to identify the exact scenarios under which fitting errors occur, or GoF tests diverge.

The results show that these failures arise primarily from data characteristics, not from the estimation methods themselves. Convergence behaviour was found to be distribution-specific. These insights are important because they identify the specific data quality constraints that must be addressed before queue modelling, parameter estimation, or GoF assessment is possible in precision-limited data.

4. A comprehensive analysis of MLE and MME parametric fitting methods under quantisation (Chapter 6).
Through extensive experiments, the manuscript revealed how quantisation distorts distributional assumptions and affects MLE and MME under different rounding precisions, parameter values and sample sizes. The research shows how the robustness of the fitting methods differs by distribution and the limitations of these fitting methods.

Although applying MLE and MME via standard statistical packages may appear straightforward, the novelty of this work lies in systematically investigating how and why their behaviour diverges under rounding, something no peer-reviewed study has done in a controlled, side-by-side evaluation of their performance when applied to rounded data.

The chapter fills that gap by demonstrating when and why parameter estimation techniques can differ under different rounding scenarios. The analysis provides practical insights into the conditions under which standard fitting procedures remain reliable or degrade when applied to rounded data.

5. Practical strategies for diagnosing and mitigating quantisation (Chapter 7)

The study evaluated practical methods for working with quantised and zero-inflated data in a log-normal distribution. The chapter addressed the zero-value problem by systematically testing constant-shift strategies. Although the experiments were conducted on a log-normal distribution, the approach generalises to other continuous distributions supported on positive numbers, providing practitioners with a simple, implementable toolkit for stabilising fitting procedures and reducing the impact of quantisation on model assessment. The contribution was necessary to overcome fitting problems during this research. Moreover, this research provides actionable guidance for researchers and practitioners seeking to perform reliable statistical modelling under severe quantisation.

6. Evaluation of unrounding methods and model-based reconstruction under quantisation (Chapter 8)

Different domains use jittering techniques to add random perturbations to datasets for smoothing or for simulation purposes. The novelty in this chapter

lies in treating unrounding as a reconstruction problem for heavily quantised data and systematically evaluating a range of jitter-based approaches for this purpose. No published work has been found during this research that applies jittering techniques to unround the data.

A benchmark for the unrounding techniques is provided. The evaluation is important because the peer-reviewed literature offers no side-by-side comparative analysis on how these techniques behave under quantisation, nor how they impact inference, GoF testing, or distributional reconstruction.

The key insight delivered is a clear understanding of the conditions under which these methods succeed or fail to recover the underlying continuous distribution. Such clarity is essential for practitioners working with quantised data.

7. Interval-Based approach for estimation (Chapter 9).

To recover the distributional shape rather than the latent point values, the manuscript demonstrated that an interval-based numerical approach using the CDF bounds can accurately reconstruct the underlying distribution even when observations are rounded. While MLE and MME treat each observation as an exact point, an assumption that is violated when values are rounded, this interval-based approach produces a correctly specified likelihood for the observed rounded data, overcoming the limitations of MLE and MME under rounding.

Instead of assuming exact values, the method models the probability mass within each rounding interval using the CDF of the underlying distribution. Each bin effectively functions as a category, and the approach estimates parameters by maximising the likelihood. This interval-based CDF approach has not been previously seen in the literature in the context of rounding. The interval-based method is computationally efficient, requiring only basic CDF evaluations rather than complex numerical integration. Importantly, it achieves this without requiring ad hoc perturbations for zero inflation.

10.3 Limitations

The results of this study are subject to a number of limitations that constrain the scope and generalisability of the findings.

First, rounding alters model inference due to collapsing a continuous range of plausible latent values into a single observed outcome. The severity of this collapse does not depend solely on the number of decimal places used in rounding, but also on the shape of the underlying distribution. For distributions with steep density near zero (e.g., Weibull with shape parameter $\beta < 1$), even fine rounding intervals can map large regions of the probability mass to the same recorded value. This means that the set of latent candidates consistent with a rounded observation can be extremely large, making true reconstruction infeasible. At best, one can only approximate the underlying variability.

Second, the production EDI dataset suffered from relatively coarse quantisation. With timestamps recorded at millisecond precision, important variations in interarrival and service times were lost, particularly for short-lived jobs whose true durations often lie well below the rounding threshold. The resulting discreteness introduced artificial clustering, loss of ordering, and violations of independence assumptions, all of which limit the ability of any modelling technique to truly recover the underlying behaviour without improved data precision.

Third, message burstiness created additional structural ambiguity in the logged data. Although downstream messages could be traced back to their parent processes, the discrete timestamps prevented accurate recovery of short-scale temporal relationships, further constraining the analysis.

These limitations highlight that while the methods developed in this manuscript can mitigate some of the effects of rounding, it cannot fully overcome the fundamental loss of information introduced by coarse quantisation.

Message burstiness created structural ambiguities in the logged data. Even though downstream messages could be traced back to their parent, the discrete timestamps prevented accurate reconstruction of causal ordering and temporal dependencies. This limits the extent to which queueing models can be validated against actual system behaviour.

While synthetic data was used to evaluate modelling behaviour under controlled conditions, synthetic datasets approximate only a subset of the complexities encountered in production environments. As such, some conclusions, particularly those concerning the effectiveness of unrounding and jitter methodologies,

should be interpreted in the context of their experimental design.

Finally, the manuscript primarily evaluates statistical modelling techniques rather than queue optimisation algorithms. Although the research question concerns queue modelling for optimisation, the data limitations meant that optimisation strategies could not be evaluated.

10.4 Future Works

Based on the findings in this manuscript, several areas for future research emerge.

Future work should model message splitting, batching, and scheduling using graph-based or hierarchical methods, which capture parent–child relationships more accurately than timestamp-driven methods.

Mixture models may capture the EDI processing behaviour more accurately than the researched single distribution models. KDE, which is like a mixture model, was tried.

The success of the interval-based approach invites further research into embedding this technique directly into large language models, simulation testing and optimisation processes, particularly for systems where rounding or discreteness cannot be avoided.

Further research is needed on data segmentation strategies. Different segmentation techniques (e.g., elbow and knee) can be used to infer message assumptions that are not yet fully understood.

An extension of this work would be to combine the interval-based CDF approach from Chapter 9 with the jittering strategies developed in Chapter 8. As the technique in Chapter 9 provides stable parameter estimates without requiring ad-hoc fixes, jittering could then be used to explore variability and refine model fit around the quantised data.

Given the observed bursty behaviour, quantisation, and the loss of message arrival order, future work could investigate other data elements in the log files that may help recover message ordering while unrounding the data.

10.5 Final Remarks

The manuscript has demonstrated that modelling queue behaviour in distributed EDI-driven supply chain systems requires a careful understanding of EDI message architecture and data quality constraints. By analysing real production data, developing new methods for dealing with quantisation, and evaluating the behaviour of fitting and unrounding techniques across multiple distributions, the research provides a comprehensive examination of the challenges inherent in modelling large-scale message-processing systems under quantisation.

Although the limitations of the data prevented the full evaluation of queue optimisation strategies, the manuscript establishes a foundation for such work and identifies the conditions under which accurate queue modelling becomes possible. The contributions made here, spanning empirical insights, offer valuable guidance for researchers, engineers, and organisations seeking to understand and optimise the performance of distributed supply chain messaging infrastructures.

Taken together, the findings demonstrate that it is possible to recover meaningful distributional structure obfuscated by quantisation, which can help diagnose bottlenecks and formulate reliable modelling strategies.

Bibliography

[1] S. Leech, D. Malone, and J. Dunne. Heads or Tails: A Framework To Model Supply Chain Heterogeneous Messages. In *Proceedings of the 28th Conference of Open Innovations Association (FRUCT)*, 2021. 10, 16, 28

[2] Robert M. Gray and David L. Neuhoff. Quantization. *IEEE Transactions on Information Theory*, 44(6):2325–2383, 1998. 10, 13

[3] Y. Linde, A. Buzo, and R. M. Gray. An Algorithm for Vector Quantizer Design. *IEEE Transactions on Communications*, 28(1):84–95, 1980. 10, 13

[4] Xiaoling Wang, Ting Yao, and Chang-Tsun Li. A Palette-Based Image Steganographic Method Using Colour Quantisation. In *Proceedings of the IEEE International Conference on Image Processing (ICIP)*, volume 2, pages II–1090. IEEE, September 2005. 10, 13

[5] John Makhoul, Stelios Roucos, and Herbert Gish. Vector Quantization in Speech Coding. *Proceedings of the IEEE*, 73(11):1551–1588, 1985. 10, 13

[6] Stuart Lloyd. Least Squares Quantization in PCM. *IEEE Transactions on Information Theory*, 28(2):129–137, 1982. 10, 13

[7] Jilei Hou. Here's why Quantization Matters for AI, March 2019. 10, 13

[8] Anthony Alford. Google Releases Post-Training Integer Quantization for Tensorflow Lite, Jul 2019. 10, 13

[9] Leslie Lamport. Time, Clocks, and the Ordering of Events in a Distributed System. In *Concurrency: The Works of Leslie Lamport*, pages

179–196. Association for Computing Machinery, New York, NY, USA, 2019. 10, 13, 85

[10] A. R. Tricker. Effects of Rounding on the Moments of a Probability Distribution. *Journal of the Royal Statistical Society: Series D (The Statistician)*, 33(4):381–390, 1984. 11, 14, 193

[11] Timothy G. Clark, Michael J. Bradburn, Selina B. Love, and Douglas G. Altman. Survival Analysis Part I: Basic Concepts and First Analyses. *British Journal of Cancer*, 89(2):232–238, 2003. 11, 14

[12] Peter Hall. The Influence of Rounding Errors on Some NonParametric Estimators of a Density and its Derivatives. *SIAM Journal on Applied Mathematics*, 42(2):390–399, 1982. 11, 14

[13] Frank A. Haight. *Handbook of the Poisson Distribution*. John Wiley & Sons, 1967. 11, 14

[14] Christoph W. Ueberhuber. *Numerical Computation 1: Methods, Software, and Analysis*, volume 16. Springer Science & Business Media, 1997. 11, 14

[15] Donald A. Berry. Logarithmic Transformations in ANOVA. *Journal of the American Statistical Association*, 43(2):439–456, 1987. 12, 15, 194, 198

[16] John P. Ekwaru and Paul J. Veugelers. The Overlooked Importance of Constants Added in Log Transformation of Independent Variables With Zero Values: A Proposed Approach for Determining an Optimal Constant. *Statistics in Biopharmaceutical Research*, 10(1):26–29, 2018. 12, 15, 194

[17] R. B. D'Agostino. *Goodness-of-Fit Techniques*. Routledge, 2017. 12, 15, 164, 194, 198

[18] Zoran Pasarić and Kristina Cindrić. Generalised Pareto Distribution: Impact of Rounding on Parameter Estimation. *Theoretical and Applied Climatology*, 136(1):417–427, 2019. 12, 15

[19] Anekal B. Sripad and Donald L. Snyder. Quantization Errors in Floating-Point Arithmetic. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 26:456–463, 1978. 12, 15

[20] Stephen B. Vardeman. Sheppard's Correction for Variances and the Quantization Noise Model. *IEEE Transactions on Instrumentation and Measurement*, 54(5):2117–2119, 2005. 12, 15

[21] Hans Schneeweiß and John Komlos. Probabilistic Rounding and Sheppard's Correction. *Statistical Methodology*, 6(6):577–593, 2009. 12, 15

[22] Scott D Grimshaw, James McDonald, Grant R McQueen, and Steven Thorley. Estimating Hazard Functions for Discrete Lifetimes. *Communications in Statistics–Simulation and Computation*, 34(2):451–463, 2005. 12, 15

[23] C. C. Heyde. Agner Krarup Erlang. In *Statisticians of the Centuries*, pages 328–330. Springer New York, 2001. 16

[24] M. Cremer and J. Ludwig. A Fast Simulation Model for Traffic Flow on the Basis of Boolean Operations. *Mathematics and Computers in Simulation*, 28(4):297–303, 1986. 16

[25] T. Iversen. A Theory of Hospital Waiting Lists. *Journal of Health Economics*, 12(1):55–71, 1993. 16

[26] R. P. S. Hermanto and A. Nugroho. Waiting-time Estimation in Bank Customer Queues Using RPROP Neural Networks. *Procedia Computer Science*, 135:35–42, 2018. 16

[27] S. Ellis, S. Bond, M. Marden, and H. Singh. Driving Strategic Value with IBM Sterling Supply Chain Business Network. https://www.ibm.com/downloads/cas/PZ7LROWL, 2020. IBM White Paper. 16, 17, 49

[28] Sheldon M Ross. *Introduction to Probability Models*. Academic press, 2014. 16, 17, 23, 24, 25, 158

[29] A. Hassellöf. Why Queue Management Systems Are Essential for Modern Businesses. https://ombori.com/blog/modern-queue-management-systems, April 2021. Accessed on 2021-04-13. 16, 49

[30] AACB. Top 10 Future Trends in Supply Chain and Logistics. https://www.aacb.com/trends-in-supply-chain-and-logistics/, July 2021. Accessed on 2021-07-11. 16

[31] A. O. Allen. *Probability, Statistics, and Queueing Theory*. Academic Press, 2014. 16

[32] M. Laguna and J. Marklund. *Business Process Modeling, Simulation and Design*. Chapman and Hall/CRC, 2019. 17

[33] J. F. Shortle, J. M. Thompson, D. Gross, and C. M. Harris. *Fundamentals of Queueing Theory*, volume 399. John Wiley and Sons, 2018. 17

[34] N. Garg. *Apache Kafka*. Packt Publishing Ltd., 2013. 17

[35] M. Toshev. *Learning RabbitMQ*. Packt Publishing Ltd., 2015. 17

[36] IBM. Introduction to IBM MQ. https://www.ibm.com/docs/en/ibm-mq/8.0?topic=overview-introduction-mq, July 2021. Accessed on 2021-07-19. 17

[37] ActiveMQ. Flexible & Powerful Open Source Multi-Protocol Messaging. https://activemq.apache.org/. Accessed on 2021-01-03. 17

[38] RabbitMQ. RabbitMQ Features. https://www.rabbitmq.com/#features. Accessed on 2021-08-19. 17

[39] M. J. Sax. Apache Kafka. In S. Sakr and A. Zomaya, editors, *Encyclopedia of Big Data Technologies*. Springer, Cham, 2018. 17

[40] HG Insights. RabbitMQ. https://discovery.hgdata.com/product/rabbitmq. Accessed on 2021-05-01. 18

[41] HG Insights. Apache Kafka. https://discovery.hgdata.com/product/apache-kafka. Accessed on 2021-01-02. 18

[42] P. Yang. Building a Machine Learning Logging Pipeline with Kafka Streams at Twitter. https://www.confluent.io/blog/how-twitter-built-a-machine-learning-pipeline-with-kafka/, September 2020. Accessed on 2020-09-25. 18

[43] J. Lee. How LinkedIn Customizes Apache Kafka for 7 Trillion Messages per Day. https://engineering.linkedin.com/blog/2019/apache-kafka-trillion-messages, October 2012. Accessed on 2012-10-28. 18

[44] N. Sharma. Featuring Apache Kafka in the Netflix Studio and Finance World. https://www.confluent.io/blog/how-kafka-is-used-by-netflix/, January 2020. Accessed on 2020-01-21. 18

[45] M. Deutscher. AWS Expands Its Serverless Capabilities and Adds a Managed Kafka Service. https://siliconangle.com/2018/11/29/aws-expands-serverless-capabilities-adds-managed-kafka-service, November 2018. Accessed on 2018-11-29. 18

[46] K. Goodhope, J. Koshy, J. Kreps, N. Narkhede, R. Park, J. Rao, and V. Y. Ye. Building LinkedIn's Real-Time Activity Data Pipeline. *IEEE Data Engineering Bulletin*, 35(2):33–45, 2012. 18

[47] H. Wu, Z. Shang, and K. Wolter. Performance Prediction for the Apache Kafka Messaging System. In *2019 IEEE 21st International Conference on High Performance Computing and Communications; IEEE 17th International Conference on Smart City; IEEE 5th International Conference on Data Science and Systems (HPCC/SmartCity/DSS)*, pages 154–161. IEEE, August 2019. 18

[48] R. Henjes, M. Menth, and C. Zepfel. Throughput Performance of Java Messaging Services Using WebSphereMQ. In *26th IEEE International Conference on Distributed Computing Systems Workshops (ICDCSW'06)*, pages 26–26. IEEE, July 2006. 18, 19

[49] IBM. About Uniform Clusters. https://ibm.co/3x5SXwW, 2022. Accessed on 2022-04-06. 19

[50] IBM. Limitations and Considerations for Uniform Clusters. https://ibm.co/3uhsDyh, 2022. Accessed on 2022-04-06. 19

[51] IBM. Streaming Queues. https://ibm.co/3DHMidr, 2022. Accessed on 2022-04-06. 19

[52] R. Dielhenn and C. McCabe. Kip-748: Add Broker Count Metrics. https://bit.ly/3x4mAPi, 2021. Accessed on 2020-04-07. 19

[53] International Business Machines Corporation. Pre-staging Messages at a Remote Location. US Patent US9634962B2, 2015. Accessed on 2022-04-07. 19, 20

[54] K.Y. Jo, J.J. Pottmyer, and E.A. Fetzner. Dod Electronic Commerce/-Electronic Data Interchange (EC/EDI) Systems Modelling and Simulation. In *Proceedings of MILCOM'95*, pages 479–483, San Diego, CA, USA, November 1995. 20, 21

[55] IBM. What is Electronic Data Interchange (EDI)? https://www.ibm.com/topics/edi-electronic-data-interchange. Accessed on 2021-02-05. 20

[56] United Nations Economic Commission for Europe (UNECE). *United Nations Directories for Electronic Data Interchange for Administration, Commerce and Transport*. UNECE, Geneva, 2020. Accessed on 2020-05-19. 20, 88

[57] J. Wells. Making EDI Work in a Multinational Company. In *IEE Colloquium on Standards and Practices in Electronic Data Interchange*, pages 1–6, London, UK, 1991. IET. 20

[58] C. L. Iacovou, I. Benbasat, and A. S. Dexter. Electronic Data Interchange and Small Organizations: Adoption and Impact of Technology. *MIS Quarterly*, 19(4):465–485, 1995. 21

[59] D. Lim and P. C. Palvia. Edi in Strategic Supply Chain: Impact on Customer Service. *International Journal of Information Management*, 21(3):193–211, 2001. 21

[60] D.A. Johnson, B.J. Allen, and M.R. Crum. The State of EDI Usage in the Motor Carrier Industry. *Journal of Business Logistics*, 13:43, 1992. 21

[61] S. Ellis, S. Bond, M. Marden, and H. Singh. Driving Strategic Value with IBM Sterling Supply Chain Business Network. https://ibm.co/39hjJZz, 2022. Accessed on 2022-04-19. 21

[62] Horst Rinne. *The Weibull Distribution: A Handbook*. Chapman and Hall/CRC, 2008. 22, 157, 158

[63] Waloddi Weibull. A Statistical Distribution Function of Wide Applicability. *Journal of applied mechanics*, 1951. 22

[64] K Das. A Comparative Study of Exponential Distribution vs Weibull Distribution in Machine Reliability Analysis in a CMS Design. *Computers & Industrial Engineering*, 54(1):12–33, 2008. 22, 23, 157

[65] Uwe Mortensen. Additive Noise, Weibull Functions and the Approximation of Psychometric Functions. *Vision Research*, 42(20):2371–2393, 2002. 22

[66] Hamza Abubakar and Shamsul Rijal Muhammad Sabri. A Bayesian Approach to Weibull Distribution with Application to Insurance Claims Data. *Journal of Reliability and Statistical Studies*, pages 1–24, 2023. 22

[67] Paulo Alexandre Costa Rocha, Ricardo Coelho de Sousa, Carla Freitas de Andrade, and Maria Eugênia Vieira da Silva. Comparison of Seven Numerical Methods for Determining Weibull Parameters for Wind Energy Generation in the Northeast Region of Brazil. *Applied Energy*, 89(1):395–400, 2012. 22

[68] William A Ericson. Introductory Probability and Statistical Applications, 1966. 23, 40

[69] Norman L Johnson, Samuel Kotz, and Narayanaswamy Balakrishnan. *Continuous Univariate Distributions, Volume 2*, volume 289. John wiley & sons, 1995. 24, 160

[70] Francis Galton. Xii. The Geometric Mean, in Vital and Social Statistics. *Proceedings of the Royal Society of London*, 29(196-199):365–367, 1879. 24

[71] Donald McAlister. Xiii. The law of the Geometric Mean. *Proceedings of the Royal Society of London*, 29(196-199):367–376, 1879. 24

[72] Ioannis Antoniou, Victor V Ivanov, Valery V Ivanov, and PV Zrelov. On the Log-normal Distribution of Network Traffic. *Physica D: Nonlinear Phenomena*, 167(1-2):72–85, 2002. 24

[73] Patrick D. T. O'Connor and Andre V. Kleyner. *Practical Reliability Engineering*. john wiley & sons, 2011. 24

[74] Arthur L Koch. The Logarithm in Biology 1. Mechanisms Generating the Log-normal Distribution Exactly. *Journal of theoretical biology*, 12(2):276–290, 1966. 24

[75] Kailash C Kapur and Michael Pecht. *Reliability Engineering*, volume 86. John Wiley & Sons, 2014. 25

[76] Karl Pearson. Contributions to the Mathematical Theory of Evolution. *Philosophical Transactions of the Royal Society of London*, page 71–110, 1894. 25, 40

[77] Richard J Larsen and Morris L Marx. *An Introduction to Mathematical Statistics*. Prentice Hall Hoboken, NJ, 2005. 25

[78] Jorge Nocedal and Stephen J Wright. *Numerical Optimization*. Springer, 1999. 25, 132

[79] Srinivasan Keshav. *Mathematical Foundations of Computer Networking*. Addison-Wesley, 2012. 26, 27, 45, 136

[80] B. W. Silverman. *Density Estimation for Statistics and Data Analysis*. Routledge, 2018. 28, 29

[81] V. A. Epanechnikov. Non-parametric Estimation of a Multivariate Probability Density. *Theory of Probability & Its Applications*, 14(1):153–158, 1969. 29

[82] S. J. Sheather and M. C. Jones. A Reliable Data-Based Bandwidth Selection Method for Kernel Density Estimation. *Journal of the Royal Statistical Society: Series B (Methodological)*, 53(3):683–690, 1991. 29

[83] B. U. Park and J. S. Marron. Comparison of Data-Driven Bandwidth Selectors. *Journal of the American Statistical Association*, 85(409):66–72, 1990. 29

[84] Michael A Stephens. The Anderson-Darling Statistic. 1979. 30, 33, 143, 196

[85] William H Press. *Numerical Recipes 3rd Edition: The Art of Scientific Computing*. Cambridge university press, 2007. 31, 34, 39, 46, 132, 264

[86] Hubert W. Lilliefors. On the Kolmogorov–Smirnov Test for Normality with Mean and Variance Unknown. *Journal of the American Statistical Association*, 62(318):399–402, 1967. 31

[87] Frank J. Jr. Massey. The Kolmogorov-Smirnov Test for Goodness of Fit. *Journal of the American statistical Association*, 46(253):68–78, 1951. 31

[88] Hubert W Lilliefors. On the Kolmogorov-Smirnov Test for Normality with Mean and Variance Unknown. *Journal of the American statistical Association*, 62(318):399–402, 1967. 31

[89] Nornadiah Mohd Razali, Yap Bee Wah, et al. Power Comparisons of Shapiro-Wilk, Kolmogorov-Smirnov, Lilliefors and Anderson-Darling Tests. *Journal of statistical modeling and analytics*, 2(1):21–33, 2011. 31

[90] Theodore W Anderson. On the Distribution of the Two-Sample Cramer-von Mises Criterion. *The Annals of Mathematical Statistics*, pages 1148–1159, 1962. 32, 196

[91] Theodore W. Anderson and Donald A Darling. A Test of Goodness of Fit. *Journal of the American Statistical Association*, 49:765–769, 1954. 33

[92] Theodore W Anderson and Donald A Darling. A Test of Goodness of Fit. *Journal of the American statistical association*, 49(268):765–769, 1954. 33

[93] Nicolaas H. Kuiper. Tests Concerning Random Points on a Circle. *Proceedings of the Koninklijke Nederlandse Akademie van Wetenschappen. Series A*, 63(1):38–47, 1960. 34

[94] Francis Galton. I. Co-relations and Their Measurement, Chiefly from Anthropometric Data. *Proceedings of the Royal Society of London*, 45(273-279):135–145, 1889. 35

[95] George E. P. Box, Gwilym M. Jenkins, Gregory C. Reinsel, and Greta M. Ljung. *Time Series Analysis: Forecasting and Control*. John Wiley & Sons, Hoboken, NJ, 5th edition, 2015. 36

[96] Joseph Lee Rodgers and W. Alan Nicewander. Thirteen Ways to Look at the Correlation Coefficient. *The American Statistician*, 42(1):59–66, 1988. 36, 37

[97] Dennis E. Hinkle, William Wiersma, and Stephen G. Jurs. *Applied Statistics for the Behavioral Sciences*. Houghton Mifflin, Boston, 2003. 36, 37

[98] Carlos M Jarque and Anil K Bera. Efficient Tests for Normality, Homoscedasticity and Serial Independence of Regression Residuals. *Economics letters*, 6(3):255–259, 1980. 37

[99] Charles Spearman. *The Proof and Measurement of Association Between Two Things*. Appleton-Century-Crofts, New York, 1961. 37

[100] Jerrold H Zar. Significance Testing of the Spearman Rank Correlation Coefficient. *Journal of the American Statistical Association*, 67(339):578–580, 1972. 37

[101] Jacob Cohen. *Statistical Power Analysis for the Behavioral Sciences*. routledge, 2013. 38

[102] Maurice G Kendall. A New Measure of Rank Correlation. *Biometrika*, 30(1-2):81–93, 1938. 38, 39

[103] Juthaphorn Sinsomboonthong. Robust Estimators for the Correlation Measure to Resist Outliers in Data. *J. Math. Fundam. Sci*, 48(3):263–275, 2016. 38

[104] G. Casella and R. L. Berger. *Statistical Inference*. Cengage Learning, 3 edition, 2021. 40

[105] R. A. Fisher. Theory of Statistical Estimation. *In Mathematical proceedings of the Cambridge philosophical society*, 22:700–725, 1925. 40, 41

[106] In Jae Myung. Tutorial on Maximum Likelihood Estimation. *Journal of mathematical Psychology*, 47(1):90–100, 2003. 40, 41, 43

[107] George Casella and Roger Berger. *Statistical Inference*. CRC press, 2024. 40, 42, 43

[108] Stephen J Wright. Numerical Optimization, 2006. 42, 132

[109] Marie Laure Delignette-Muller and Christophe Dutang. *Fitdistrplus: An R Package for Fitting Distributions*, 2023. R package version 1.1-8. 42

[110] Brian Ripley, Bill Venables, Douglas M Bates, Kurt Hornik, Albrecht Gebhardt, David Firth, and Maintainer Brian Ripley. Package 'MASS'. *Cran r*, 538(113-120):822, 2013. 42

[111] W. N. Venables and B. D. Ripley. *Modern Applied Statistics with S. Fourth Edition*. Springer, New York, fourth edition, 2002. ISBN 0-387-95457-0. 42

[112] Pierre Simon Laplace. *Théorie Analytique Des Probabilités*. Courcier, 1820. 42

[113] Ilker Yildirim. Bayesian Inference: Metropolis-Hastings Sampling. *Dept. of Brain and Cognitive Sciences, Univ. of Rochester, Rochester, NY*, 2012. 43

[114] Stuart Geman and Donald Geman. Stochastic Relaxation, Gibbs Distributions, and the Bayesian Restoration of Images. *IEEE Transactions on pattern analysis and machine intelligence*, (6):721–741, 1984. 43

[115] Zoubin Ghahramani. Probabilistic Machine Learning and Artificial Intelligence. *Nature*, 521(7553):452–459, 2015. 43

[116] Eric Jacquier, Nicholas G Polson, and Peter E Rossi. Bayesian Analysis of Stochastic Volatility Models. *Journal of Business & Economic Statistics*, 20(1):69–87, 2002. 43

[117] Matthew Stephens and Peter Donnelly. A Comparison of Bayesian Methods for Haplotype Reconstruction From Population Genotype Data. *The American Journal of Human Genetics*, 73(5):1162–1169, 2003. 43

[118] Claudia Tebaldi and Reto Knutti. The use of the Multi-Model Ensemble in Probabilistic Climate Projections. *Philosophical transactions of the royal society A: mathematical, physical and engineering sciences*, 365(1857):2053–2075, 2007. 43

[119] David M Blei, Andrew Y Ng, and Michael I Jordan. Latent Dirichlet Allocation. *Journal of machine Learning research*, 3(Jan):993–1022, 2003. 43

[120] Brian D. Ripley and W. N. Venables. *MASS: Support Functions and Datasets for Venables and Ripley's MASS*. R Core Team, 2023. R package version 7.3-60. 43

[121] Halbert White. Maximum Likelihood Estimation of MisSpecified Models. *Econometrica: Journal of the econometric society*, pages 1–25, 1982. 43

[122] John Mullahy. Specification and Testing of Some Modified Count Data Models. *Journal of econometrics*, 33(3):341–365, 1986. 43

[123] Diane Lambert. Zero-inflated Poisson Regression, with an Application to Defects in Manufacturing. *Technometrics*, 34(1):1–14, 1992. 43, 44

[124] Daniel B Hall. Zero-inflated Poisson and Binomial Regression with Random Effects: A Case Study. *Biometrics*, 56(4):1030–1039, 2000. 43

[125] A. Zeileis, C. Kleiber, and S. Jackman. Regression Models for Count Data in R. *Journal of Statistical Software*, 27(8):1–25, 2008. 44

[126] Martin Ridout, Clarice G. B. Demétrio, and John Hinde. Models for Count Data with Many Zeros. In *Proceedings of the XIXth International Biometric Conference*, pages 179–192, Cape Town, South Africa, 1998. International Biometric Society. 44

[127] P. C. Mahalanobis. *On the Generalized Distance in Statistics*. Sankhyā: The Indian Journal of Statistics, 1936. 45

[128] Allan Bluman. *Elementary Statistics: A step by step Approach 9e*. McGraw Hill, 2014. 45

[129] R Dennis Cook. Detection of Influential Observation in Linear Regression. *Technometrics*, 19(1):15–18, 1977. 45

[130] Peter J Rousseeuw and Katrien Van Driessen. A Fast Algorithm for the Minimum Covariance Determinant Estimator. *Technometrics*, 41(3):212–223, 1999. 45

[131] Christophe Leys, Olivier Klein, Yves Dominicy, and Christophe Ley. Detecting Multivariate Outliers: Use a Robust Variant of the Mahalanobis Distance. *Journal of experimental social psychology*, 74:150–156, 2018. 45

[132] Bruce M Hill. A Simple General Approach to Inference About the Tail of a Distribution. *The annals of statistics*, pages 1163–1174, 1975. 45

[133] WJ Dixon. Processing Data for Outliers. *Biometrics*, 9(1):74–89, 1953. 45

[134] Wilfred J Dixon. Analysis of Extreme Values. *The Annals of Mathematical Statistics*, 21(4):488–506, 1950. 45

[135] Heiner Lasi, Peter Fettke, Hans-Georg Kemper, Thomas Feld, and Michael Hoffmann. Industry 4.0. *Business & information systems engineering*, 6(4):239–242, 2014. 46

[136] Benny Tjahjono, Carlos Esplugues, Enrique Ares, and Gustavo Pelaez. What does industry 4.0 Mean to Supply Chain? *Procedia manufacturing*, 13:1175–1182, 2017. 46

[137] Rodrigo Goyannes Gusmão Caiado, Luiz Felipe Scavarda, Bruno Duarte Azevedo, Daniel Luiz de Mattos Nascimento, and Osvaldo Luiz Gonçalves Quelhas. Challenges and Benefits of Sustainable Industry 4.0 for Operations and Supply Chain Management. A Framework Headed Toward the 2030 Agenda. *Sustainability*, 14(2):830, 2022. 46

[138] Ethirajan Manavalan and Kandasamy Jayakrishna. A review of Internet of Things (IoT) Embedded Sustainable Supply Chain for Industry 4.0 Requirements. *Computers & industrial engineering*, 127:925–953, 2019. 46

[139] Diane J Cook, Juan C Augusto, and Vikramaditya R Jakkula. Ambient Intelligence: Technologies, Applications, and Opportunities. *Pervasive and mobile computing*, 5(4):277–298, 2009. 46

[140] Dmitry Korzun, Elena Balandina, Alexey Kashevnik, Sergey Balandin, and Francesco Viola, editors. *Ambient Intelligence Services in IoT Environments: Emerging Research and Opportunities*. IGI Global, Hershey, Pennsylvania, USA, 2019. 46, 47

[141] D. G. Korzun, S. I. Balandin, V. Luukkala, P. Liuha, and A. V. Gurtov. Overview of Smart-M3 Principles for Application Development. https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.704.6333&rep=rep1&type=pdf, 2012. Accessed on 2025-07-22. 47

[142] Sergey Balandin and Heikki Waris. Key Properties in the Development of Smart Spaces. In *International Conference on Universal Access in Human-Computer Interaction*, pages 3–12. Springer, 2009. 47

[143] Francesco Morandi, Luca Roffia, Alfredo D'Elia, Fabio Vergari, and T Salmon Cinotti. A Smart-M3 Semantic Information Broker Implementation. In *2012 12th Conference of Open Innovations Association (FRUCT)*, pages 1–13. IEEE, 2012. 47

[144] Dmitry Korzun, Aleksey Varfolomeyev, Anton Shabaev, and Vladimir Kuznetsov. On Dependability of Smart Applications Within Edge-Centric and Fog Computing Paradigms. In *2018 IEEE 9th International Conference on Dependable Systems, Services and Technologies (DESSERT)*, pages 502–507. IEEE, 2018. 47

[145] B.W. Silverman. *Density Estimation for Statistics and Data Analysis*. Chapman and Hall/CRC, 1986. 75

[146] R. M. Gray and D. L. Neuhoff. Quantisation. *IEEE Transactions on Information Theory*, 44(6):2325–2383, 1998. 85

[147] George EP Box and David R Cox. An Analysis of Transformations. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 26(2):211–243, 1964. 93

[148] 1 EDI Source. EDI 997 Functional Acknowledgement Specifications. https://bit.ly/3x0gb7L. Accessed on 2022-04-24. 120

[149] EDI Basics. What are EDI Document Standards? https://bit.ly/3qYlJvy. Accessed on 2022-04-22. 121

[150] Stephen Boyd and Lieven Vandenberghe. *Convex Optimization*. Cambridge university press, 2004. 132

[151] Marie Laure Delignette-Muller and Christophe Dutang. fitdistrplus: An R Package for Fitting Distributions. *Journal of Statistical Software*, 64(4):1–34, 2015. 135, 136, 139

[152] Julian Faraway, George Marsaglia, John Marsaglia, and Adrian Baddeley. *Goftest: Classical Goodness-of-Fit Tests for Univariate Distributions*, 2021. R package version 1.2-3. 136

[153] Sonya Leech, David Malone, and Jonathan Dunne. Log-normal distribution modelling with quantised data. In *2023 34th Irish Signals and Systems Conference (ISSC)*, pages 1–7. IEEE, 2023. 155

[154] National Institute of Standards and Technology. Lognormal Distribution. NIST/SEMATECH Engineering Statistics Handbook. Accessed on 2025-02-03. 158

[155] International Atomic Energy Agency. *Pressurized Water Reactor Simulator*. Number 65 in IAEA Training Course Series. International Atomic Energy Agency, Vienna, 2003. 217

[156] Sonya Leech, David Malone, and Jonathan Dunne. Lost in rounding: How small data adjustments create statistical problems for mle and mme. In *2025 35th Irish Signals and Systems Conference (ISSC)*, pages 1–6. IEEE, 2025. 221