Insurance Risk Premium Development with Model Risk

Minkun Kim

M.Sc in Computing

A dissertation submitted in fulfilment of the requirements for the award of Doctor of Philosophy (PhD)

to the

Dublin City University

School of Computing

Supervisors:

Prof. Martin Crane

Dr. Marija Bezbradica

April, 2025

Declaration

I hereby certify that this material, which I now submit for assessment on the programme of study leading to the award of PhD is entirely my own work, and that I have exercised reasonable care to ensure that the work is original, and does not to the best of my knowledge breach any law of copyright, and has not been taken from the work of others save and to the extent that such work has been cited and acknowledged within the text of my work.

Minkun Kim

ID No.: **18212693**

Date: April 14, 2025

Acknowledgements

I am sincerely grateful to my supervisors, Prof. Martin Crane and Dr. Marija Bezbradica, for their invaluable guidance and support throughout my Ph.D. journey. They not only allow me the freedom to follow my curiosity but also offered their unwavering support—intellectually, financially, and in every way possible. In particular, I learned a great deal from them about analytical thinking and clarity. They taught me how to navigate complex ideas and refine them into more digestible forms. Their questions constantly challenged me to express my thoughts precisely, without losing depth. This training shaped not only my research, but also the way I think and work. In hindsight, I have immense respect for their patience, especially during the many instances when I ignored their advice, stubbornly pursued my own ideas — only to later realize that their guidance had been the most effective path all along.

I would like to express my heartfelt gratitude to my friend and mentor, David Lindberg, in the United States. I've learned so much about Dirichlet processes from him, but more than that, he opened my eyes to a deeper, more intricate world of mathematics that I had never imagined. His insights and encouragement gave me the confidence to embrace new challenges with curiosity and determination.

Last but not least, I am deeply grateful to my family — my mom, Sukja; my wife, Jenny; my father-in-law, Rainer; my mother-in-law, Nicole; and my little brother, Benny. Their belief in me has been a constant source of motivation, and I would not have come this far without them.

Contents

List of Figures

List of Tables

Insurance Risk Premium Development with Model Risk

Minkun Kim

Abstract

Accurate risk premium prediction is critical for competitiveness and growth in general insurance business. Traditional approaches focus on clustering risks into well-defined groups to improve prediction accuracy, but practical challenges such as poorly defined risk classes and unexpected model risks complicate this process.

This thesis tackles diverse model risks in risk premium prediction using a Bayesian framework. Unlike classical actuarial methods that rely solely on data, Bayesian models incorporate parameter knowledge, offering flexibility in handling erroneous data issue. We leverage this advantage to link Bayesian parametric/nonparametric frameworks with state-of-art strategies for managing incomplete data issues, such as Missingness at Random (MAR) and Non-Differential Berkson (NDB) mismeasurement. Additionally, we address other key analytical challenges, including heterogeneity, convolution, and scalability.

The first part of this thesis focuses on Bayesian parametric frameworks, comparing Bayesian partial pooling with traditional error correction method such as Simulation Extrapolation (SIMEX). The second part extends to the Bayesian nonparametric (BNP) framework, investigating the efficiency of Bayesian parameter-free clustering while addressing incomplete data using techniques such as data augmentation and Gustafson correction. We develop a hybrid Dirichlet Process Mixture (DPM) model and compare it with Bayesian hierarchical models and other classical actuarial approaches. The originality of this thesis lies in leveraging existing state-of-the-art approaches and pushing the boundaries of their applicability to a broader analytical framework, encompassing challenges such as heterogeneity, convolution error, scalability, missingness, and mismeasurement.

Based on the combined use of Bayesian parametric and nonparametric models trained on multiple insurance datasets, a critical insight from our study is that correction performance depends on the alignment between two conditional variances in the Gustafson framework—one conditioned on the true covariate and the other on the chosen covariate to approximate the true covariate. We introduce the concept of a *scaling factor* for the first time to measure this alignment, applying it in calibrating the MCMC simulations. Overall, we believe that this thesis enhances the practical application of Bayesian tools for actuaries. Key innovations include:

1. Integrating data augmentation and Gustafson correction with Bayesian predictive modeling frameworks, leveraging unique prior knowledge of variance in the correction process.

2. Introducing log-normal and log-skewnormal convolution techniques for risk premium modeling, enhancing theoretical reliability.

3. Marking the first instance of integrating advanced Bayesian techniques with scalable methodologies tailored for risk premium prediction.

Chapter 1

Introduction

1.1 Motivation

In insurance, pricing non-life products refers to the process of setting the premiums that insurers charge policyholders for protection against the occurrence of insured contingent events. Examples include: property damage, automobile accidents, natural disasters, and more. Just like any other business, pricing in non-life insurance comprises seller's cost and profit components alike; however, unlike other businesses, an insurer's exact costs cannot be predetermined. This is because the insurer's costs are determined based on whether the insured event occurs or not in the future and, if it does, how much the claim amount associated with it turns out to be (Kaas et al. 2008).

In the pricing process, the total insured claim amount is covered by a portion of the total premium, known as the *risk premium* (Werner and Modlin 2010). From the perspective of actuaries, a great deal of importance is given to the risk premium for two reasons: First, unlike other deterministic expenses — administrative expenses, claim adjustment expenses, etc. — that need to be viewed from socioeconomic angles[1], the determination of the total insured claim amounts (covered by the risk premium) is purely rooted in mathematical modeling based on the aggregate pay-

[1]Pricing is asked to consider multiple socioeconomic factors in the dynamics of the supply and demand of insurance policies, and thus the pricing strategy can vary by the type of policies to meet the insurer's particular objective (Parodi 2023).

ment data from the past; Second, as underscored by Phillips 1994, the risk premium serves crucial purposes by offering a major input into many different actuarial tasks. These are not limited to setting premiums (i.e. what premium to charge policyholders), but include reserving (i.e. how much money to be set aside to cover the claim costs), reinsurance arrangements (i.e. what portion of the loss to be transferred to reinsurers), or solvency testing (i.e. how to assess the company's financial position), etc. It goes so far as to say that insurers' entire business strategy can depend on the accuracy of the risk premium estimation (Ohlsson and Johansson 2010). However, in recent years, many actuarial researchers have argued that there is more to risk premium modeling than the prediction accuracy. Over the last few decades, technological advances, changes in customer behaviors and socioeconomic trends, etc. have transformed the insurance product landscape, and actuaries now face a growing array of variations in coverage categories, types of risks, policy structures, and emerging regulatory requirements. These factors compound the complexity of risk scenarios, necessitating a new approach to risk premium modeling (Bhattacharyya 2020).

With regard to the increasing complexity in the risk scenarios, a recent discussion of Parodi 2023 has drawn a few key points to attention for better risk premium modeling. In the view of Parodi 2023, the main attributes required (three overall challenges) by the risk premium modeling framework can be outlined in Figure 1.1 and as follows:

- **1. Coherent Propagation of Uncertainty**

  For complicated risk scenarios, a risk premium modeling framework is required to propagate uncertainty[2] (from parameter estimation to future claim prediction), and allow for quantifying this uncertainty in a more systematic and consistent manner throughout the modeling process. This is because ever-changing risk scenarios coupled with the stochasticity in the insured claim naturally bring various uncertainties to many different levels of the modeling

[2]Coherent propagation of uncertainty helps greatly improve risk assessment accuracy. See Rocquigny and Devictor 2008.

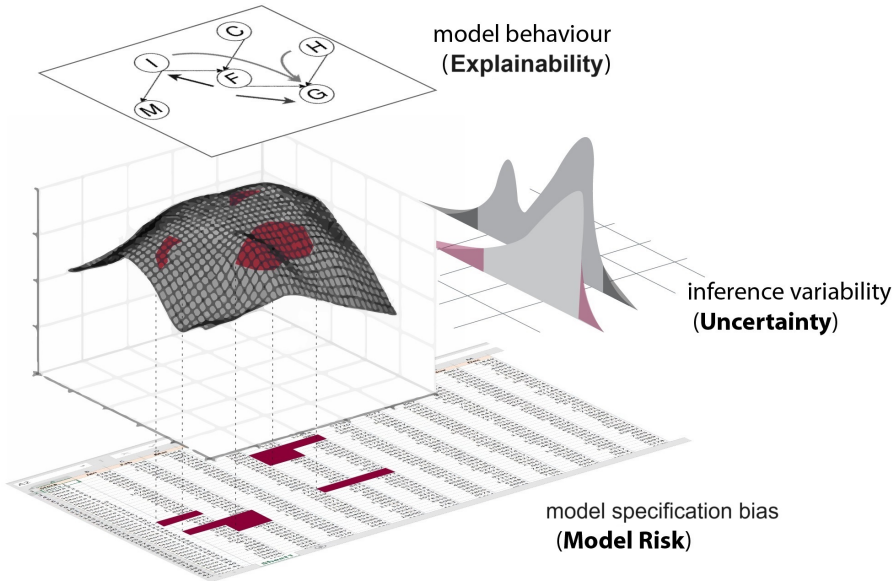

Figure 1.1: Three overall challenges in risk premium modeling framework

pipeline (Jackson and Sharples 2010). To this end, Rocquigny and Devictor 2008 underline the importance of articulating the types of uncertainties lurking in each modeling step. This includes uncertainty in data preprocessing (since the true data may never be known), parameter estimation (the true parameter values may never be known), model comparison (the true models may never be known), and approximation by simulations (where each iteration may produce slightly different results). By properly accounting for the effect of these uncertainties and communicating the effect to stakeholders based on the granular detail of uncertainty exposure, insurers can improve the accuracy of their estimations and make risk-informed decisions on the multifaceted actuarial tasks — pricing, reserving, capital allocation, etc. (Cowling et al. 2011).

- **2. Explainability**

  In order to facilitate communication between the actuaries and stakeholders[3], it is crucial to maintain the explainability of the risk premium modeling framework. Since the landscape of the potential risks is constantly changing,

[3]They are individuals or groups who are affected by the outcomes of the modeling process. This may include policyholders, risk managers, regulators, investors, reinsurers, etc. (Werner and Modlin 2010).

actuaries are required to identify the risk profiles and partition their insurance portfolio into relevant risk classes in a prompt manner (Kuo and Lupton 2023). The key lies in efficient communication between actuaries and stakeholders because, without a clear understanding of the features or context in which different stakeholders operate, swift reaction to potential risks and their dynamics becomes impossible. For example, when the risk premium model leads to a decision denying a claim or hiking a premium, etc., the insurers are obligated to justify these decisions to their stakeholders based on the modeling results. If the model is unexplainable, it would undermine trust in the insurer's decision-making process (Kuo and Lupton 2023). Consequently, informed decisions from the risk premium model should be both easy to understand and aligned with business considerations (Lage et al. 2018).

- **3. Combating Model Risk**

  As the complexity of the risk scenarios grows, an efficient control of *model risk*[4] becomes essential for the proper development of the risk premium. The model risk refers to a risk of adverse consequences from the decisions based on incorrect data or a misused model (Black et al. 2018). It should be noted that an actuarial modeling process is always subject to model risk from a misspecified variable relationship or flawed data input such as truncated values, mismeasured values, etc. (Aggarwal et al. 2016). For example, Rocher and Hendrickx 2019 point out that the problem of flawed or incomplete data often arises from the necessity to integrate various data sources, both internal (customer-related) and external (market-related, etc.). However, due to factors such as the type of policy, domain-specific regulations, or data privacy issues, etc., not all information can be utilized at all times or data may have varying formats, definitions, or units, leading to difficulties in putting them together. Above all, the risk premium model developed on incomplete data would be misspecified, and lead to significant financial losses for the insurers.

[4]Dowd 2003 elaborates all sources of model risk (such as misspecified stochasticity, misinformed risk factors, erroneous relationship, etc.) that need to be considered in finance research.

Actively adopting Parodi 2023's view, the main interest of this thesis is to develop a new risk premium modeling framework. In particular, we prioritize the 'combating of model risk' over other attributes because this risk has the potential to invalidate any attempts to perform a proper analysis.

1.2 General Insurance Terminology

Below, we provide key terminologies to assist readers new to general insurance, ensuring a clearer understanding of the concepts discussed throughout this thesis.

- **General Insurance:** an insurance business domain, also known as *non-life* insurance or *property and casualty* (P&C) insurance. It provides coverage for risks like property damage, liability, accidents. Unlike *life* insurance, it covers specific losses over a fixed short-term period, offering financial protection for tangible assets (Ohlsson and Johansson 2010).

- **Policy:** an agreement between an insurer and a policyholder, where the individual makes regular payments (*premiums*) in return for financial compensation for losses outlined in the policy terms (Werner and Modlin 2010). It is important to differentiate between two common types of non-life insurance policies: *personal* and *group* policies. Personal policies are tailored to suit individual or families' needs (e.g., auto, home, long-term care), while group policies provide coverage for a defined group, such as business operations (Baranoff et al. 2006; Baranoff 2009). In this thesis, we focus exclusively on group policies, where each single policy can cover multiple assets, and thus encompass multiple claim amounts, resulting in an aggregate insured claims for a single policyholder (i.e., the business takes out the policy).

  The following explains the concept of group policy as it relates to the specific data structure consistently considered throughout this thesis.

Insurer's portfolio

Grp.Policy ($h = 1$): $\{N_1, \mathbf{X}_1^F, Y_{1(1)}, \cdots Y_{1(N_1)}, \Sigma_{i=1}^{N_1} Y_{1(i)}, \mathbf{X}_1^S\}$ taken out by oil company.

Grp.Policy ($h = 2$): $\{N_2, \mathbf{X}_2^F, Y_{2(1)}, \cdots Y_{2(N_2)}, \Sigma_{i=1}^{N_2} Y_{1(i)}, \mathbf{X}_2^S\}$ taken out by water company.

$\vdots$

Grp.Policy ($h = H$): $\{N_H, \mathbf{X}_H^F, Y_{H(1)}, \cdots Y_{H(N_H)}, \Sigma_{i=1}^{N_H} Y_{1(i)}, \mathbf{X}_H^S\}$ taken out by car company.

In this formulation:

- N_h: claim count within a policy h.
- $\mathbf{X}_h^F$: a set of covariates (features) associated with N_h within a policy h. Ensure that the $\mathbf{X}_h^F$ is distinct from $\mathbf{X}_h^S$, as the superscript F in $\mathbf{X}_h^F$ represents claim "frequency" and $\mathbf{X}_h^F$ is solely associated with the claim count N_h, while the superscript s in $\mathbf{X}_h^S$ stands for claim "severity" and $\mathbf{X}_h^S$ is solely associated with the claim amounts $Y_{h(1)}, Y_{h(2)}, \cdots, Y_{h(N_h)}$.
- $Y_{h(i)}$: claim amount for the ith asset ($i = 1, \cdots, N_h$) within a policy h. For simplicity, we will denote $Y_{h(i)}$ as Y_{hi} throughout this thesis.
- $\Sigma_{i=1}^{N_h} Y_{h(i)}$: aggregate claim amount across the assets within a policy h. According to Kaas et al. 2008, claim amounts are often assumed to be independent to simplify modeling, as they usually arise from unrelated events, like car accidents involving different policyholders. However, in cases of systemic risks or catastrophes, dependence models such as copulas may be used.
- $\mathbf{X}_h^S$: a set of covariates (features) associated with $Y_{h(1)}, Y_{h(2)}, \cdots, Y_{h(N_h)}$ within a policy h. In other words, each claim amount $Y_{h(1)}, Y_{h(2)}, \cdots, Y_{h(N_h)}$ is influenced by the same set of covariates $\mathbf{X}_h^S$, meaning that claim amounts belonging to the same policy h share the same covariate values.

In group policies, each policyholder is typically an organization or association. Given that each organization can have multiple assets to protect, but typically takes out only one group policy, that policy covers a sum of multiple insured claim amounts per organization.

- **Policyholder:** an individual or organization responsible for paying the premiums, but entitled to the benefits outlined in the policy (Ohlsson and Johansson 2010). This thesis focuses exclusively on an organization as a policyholder.

- **Claim:** a formal request made by a policyholder to an insurer for coverage for a loss as specified in the policy. The claim initiates the insurer's review process (i.e. *claim adjustment process*), during which they assess the details of the incident, verify the validity of the claim, and determine the appropriate amount of compensation (Werner and Modlin 2010).

- **Claim Frequency:** the number of claims (*claim count*) made by policyholders over a specific period. It helps insurers assess the likelihood of claims occurring and is used in premium pricing (Ohlsson and Johansson 2010).

- **Claim Severity:** the insurer's payout (*claim amount*) associated with each claim that is filed by a policyholder. Claim severity is often assessed by examining the average cost of claims (*average claim amount*) in relation to the aggregate claim amounts within a single policy (Ohlsson and Johansson 2010).

- **Aggregate Claim Amount:** the total sum of all individual claim amounts for a single policy within a specific timeframe (Wuthrich 2020). Note that each policy is in the form of a group policy in this thesis.

- **Total Aggregate Claim Amount:** the combined sum of all claims filed across multiple policies in an insurer's portfolio within a specific timeframe (Wuthrich 2020). Note that each policy is a group policy in this thesis.

- **Insurer's portfolio:** the collection of insurance policies held by an insurer. This encompasses all the risks covered by the insurer, reflecting its market strategy, and thus it is essential for determining premium pricing strategies (Wuthrich 2020). This thesis assumes that the insurer's portfolio consists solely of group policies.

1.3 Hypothesis and Research Questions (RQs)

With the aforementioned three attributes (uncertainty propagation, explainability, and model risk) in focus for the new risk premium modeling framework, this thesis hypothesizes that under a *Bayesian* framework, all the three attributes can be efficiently addressed at the same time, which eventually leads to better prediction performance of the risk premium model.

Unlike the Frequentist framework, which relies solely on the quantity and quality of available data, the Bayesian framework incorporates both data and parameter knowledge by developing posterior distributions in the modeling process. This is achieved through a joint density that combines the distributions of parameters and data, updating parameter knowledge (posterior) with the available data (Zyphur and Oswald 2015). Figure 1.2 illustrates the basic idea of the Bayesian framework. The framework is designed to use as much information as possible by blending knowledge of parameters and data, which is favorable when the available data exhibit limitations in reliability. It is true that the Bayesian framework might lead to inaccurate results without proper prior knowledge about parameters. However, as shown in Gupta 2012, once an informative prior is available, the Bayesian method is likely to be far more advantageous than any other inference framework due to the capability of balancing/synthesizing knowledge between parameters and data.

Arguably, such a unified nature of the Bayesian perspective suffices the three requisite attributes of the risk premium modeling framework described in Figure 1.1 to a far greater extent than traditional actuarial modeling approaches. Firstly, the Bayesian framework facilitates the coherent propagation of uncertainty from data to parameters. The prior term captures uncertainty about the parameters, while the likelihood term captures uncertainty about the observed data given the parameters. By constructing a joint density, the Bayesian framework synthesizes uncertainties from both data and parameters, explaining the overall uncertainty about the parameters. This is followed by making probability statements for the likelihood of parameters using the posterior distribution, which provides a full distribution of each

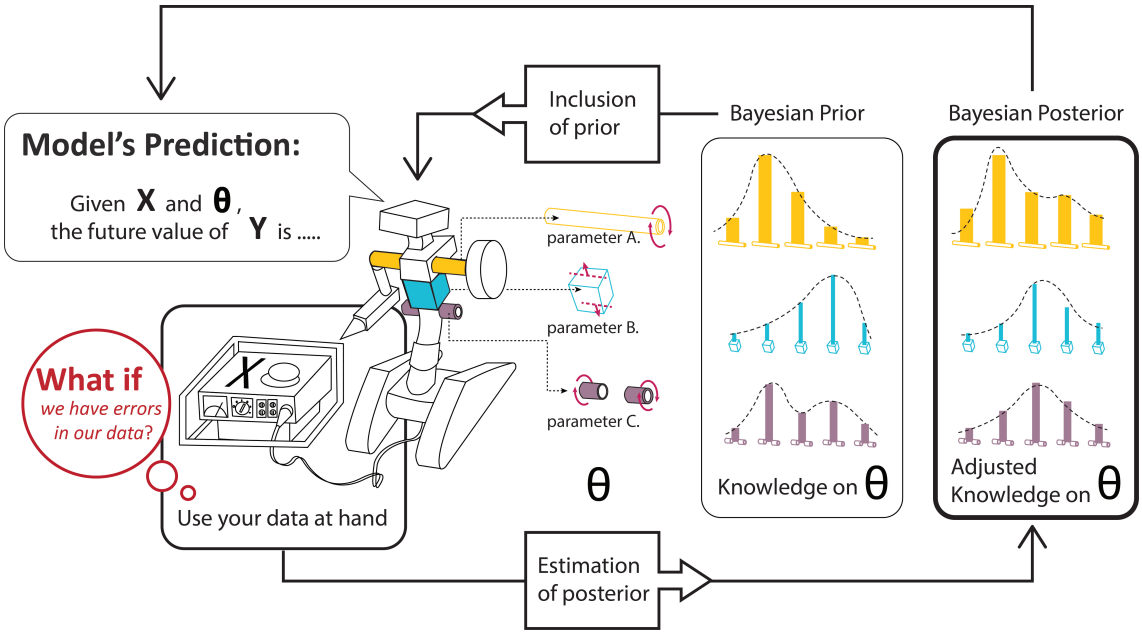

Figure 1.2: Balancing data with parameters via Bayesian thinking: 1) inclusion of prior in the modeling, 2) estimation of posterior based on the data at hand, 3) prediction of outcome based on the data and the estimated posterior.

parameter conditional on the data. This allows different uncertainties of interest to be traced and quantified (Droguett and Mosleh 2008).

Secondly, in the Bayesian framework, the model becomes highly transparent and explainable because all parameters or moments — mean, variance, skewness, etc. — are treated and estimated in a similar fashion (e.g. an analytical inference based on conjugate priors or a simulation inference based on Monte Carlo sampling). Traditional parameter or moment estimation in the non-Bayesian framework, on the other hand, tends to rely on different statistical processes by different situations, influenced by factors like assumptions, sample size, etc. This can needlessly amplify model complexity (Gelman and Carlin 2013). In conjunction with such transparent inference, the Bayesian framework seamlessly integrates the causal-inference paradigm by utilizing a joint density as well. In other words, the Bayesian framework facilitates the exploration of causal relationships between the outcome and other factors, surpassing mere identification of associations. (Gelman and Meng 2004).

Thirdly, the Bayesian framework possesses flexibility for accommodating a wide range of auxiliary techniques such as data augmentation, multiple imputation, etc.,

which is particularly useful for combating model risk issues in a coherent manner (Pollino and Henderson 2010). This is because it allows for specifying customized models and conducting inference within them, using the joint density to potentially support a complex relationship between parameters and data.

Before we transition from formulating our Bayesian hypotheses to articulating a series of research questions, it is important to underscore one aspect that sets the context for our research. This thesis focuses on the scenario where the inclusion of covariates represents a major source of model risk. Covariates provide useful information such as policyholders' characteristics (e.g., age, location, type of coverage), and other attributes of insured events that lead to claims. By default, risk premium modeling involves analyzing various covariates, such as policyholder characteristics and external factors (e.g., economic conditions, weather patterns, etc.), to understand how these variables influence the insured claim amounts. This helps to explain the variations in risk and claim costs across different policies, allowing premium setting to accurately reflect the underlying risk for each policyholder or group of policyholders (Boland 2006). However, model risks such as incorrect model assumption, misspecified variable relationships, or flawed data input in non-life pricing largely arises from the inclusion of covariates in the model (Aggarwal et al. 2016). This point is extensively discussed in detail in Chapter 3.

Building upon our Bayesian hypothesis for the risk premium modeling and the discussion regarding the model risk associated with covariates, we present the summary graphic in Figure 1.3 depicting Bayesian potential and list a series of research questions for investigation in what follows. Throughout this thesis, 'RQ' stands for 'research question'. Given that the property of uncertainty propagation and explainability is already inherent to the Bayesian framework (Gelman and Carlin 2013), all research questions listed here are formulated in relation to the model risk issues with the inclusion of covariates denoted as $\mathbf{X}$. A detailed discussion on coherent uncertainty propagation and explainability issues for the Bayesian risk premium modeling can be found in Appendix B.

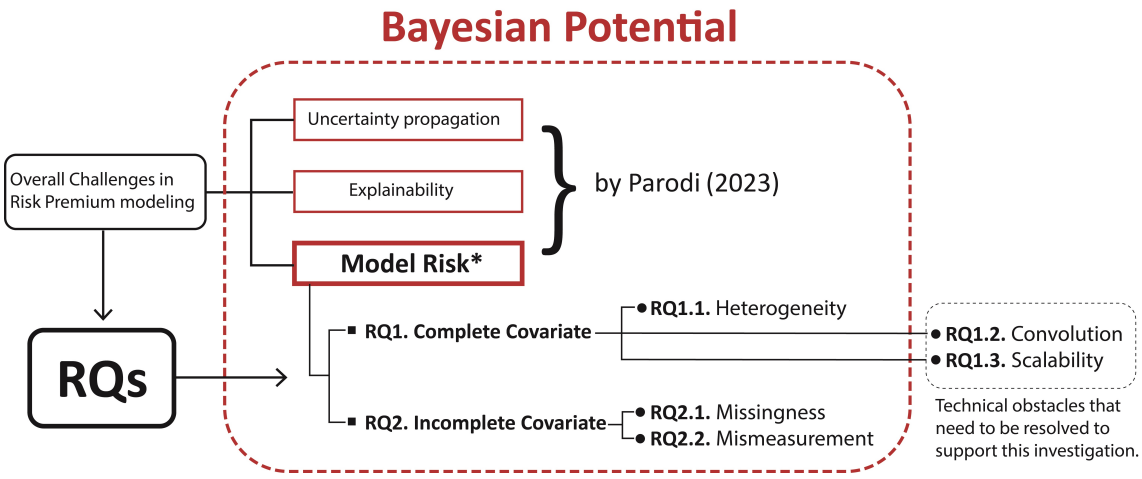

Figure 1.3: Bayesian potential and RQs: three key issues in risk premium modeling — uncertainty propagation, explainability, and model risk (Parodi 2023). This thesis primarily tackles covariate-based model risk using the Bayesian paradigm, while acknowledging that uncertainty propagation and explainability are inherently managed by the Bayesian framework. The emphasis on the covariate-based model risk guides the development of all RQs in this thesis.

For the sake of simplicity of expression, let Y_{hi} and S_h represent the individual claim amount and aggregate claim amount for a policy h respectively.

- **RQ1. Model risks arising from complete covariate: (conventional issues)**

  Even in the absence of concern for the quality of covariate data, the covariate-based model risk can stem from the conventional, inherent mathematical conflicts listed in the RQs that follow. To what extent can a Bayesian framework help mitigate the model risks tied to these conventional issues?

  - **Sub RQ1.1: Within-group heterogeneity and $\mathbf{X}_h$:**

    How can within-group heterogeneity in $S_h|\mathbf{X}_h$ (aggregate claim data for a single policy h conditioned on covariates) as well as $\Sigma S_h|\mathbf{X}_h$ (total aggregate claim data across all policies $h = 1, \cdots, H$ conditioned on covariates) be captured and accounted for to improve the prediction performance of the risk premium model?

  - **Sub RQ1.2: Convolution error and $Y_{hi}|\mathbf{X}_h, S_h|\mathbf{X}_h$:**

    With a lack of closed-form solutions for the log-normal sum and a vi-

olation of the assumptions of the i.i.d[5] summands $Y_{hi}|\mathbf{X}_h$, how can we approximate the aggregate claim $S_h|\mathbf{X}_h$ as well as the total aggregate claim $\Sigma S_h|\mathbf{X}_h$?

- **Sub RQ1.3: Scalability with the growing sample size of $Y_{hi}, S_h, \mathbf{X}_h$:** With scenarios in which claim information is expected to grow over time, how can risk premium modeling exploit the increased sample size to produce more accurate and stable inference results?

- **RQ2. Model risks arising from incomplete covariate: (Missingness/Mismeasurement)**

  With regard to the poor quality of covariate data, the model risk can stem from the model misspecification issues listed below. To what extent can a Bayesian framework help mitigate the model risks tied to the data quality issues in what follows?

  - **Sub RQ2.1: MAR covariate:**

    With the inclusion of Missing at Random (MAR)[6] covariates, how can we mitigate impaired data quality and ensure building a reliable risk premium model?

  - **Sub RQ2.2: NDB covariate:**

    With the inclusion of mismeasured covariates with Non-Differential Berkson error (NDB)[7], how can we mitigate impaired data quality and ensure building a reliable risk premium model?

Figure 1.3 summarises our hypothesis discussed so far and a series of research questions by using the keywords. Again, this thesis aims to examine the Bayesian hypothesis, which asserts the Bayesian paradigm's potential to overcome the current

[5]independent and identically distributed
[6]For the definition, see Section 3.3.1.
[7]For the definition, see Section 3.3.2.

challenges — uncertainty propagation, explainability, model risk — and develop a new approach to risk premium modeling.

As mentioned previously, the focus of the research questions is limited to the cases where the model risk in the risk premium modeling is attributable to the inclusion of covariates. Richardson and Gilks 1993 points out that a Bayesian framework sees the erroneous models as deviations from the true model, and mitigates this discrepancy through calibrating the relevant parameters with external knowledge. The research questions regarding the heterogeneous outcome in **RQ1.1** and missing/mismeasured covariates in **RQ2.1, RQ2.2** are directly linked to erroneous models, as these issues result from poor-quality inputs. This connection makes them suitable focal points for investigation in this thesis within the Bayesian paradigm. However, note that convolution issue in **RQ1.2** and scalability issue in **RQ1.3** may not be resolved by the Bayesian paradigm as they represent inherent mathematical obstacles to the actuarial modeling process itself. Nevertheless, resolving these obstacles (by answering **RQ1.2, RQ1.3**) remains crucial to facilitating the investigation of the Bayesian paradigm's potential on the risk premium modeling.

1.4 Thesis Structure

The structure of this thesis is as follows. Chapter 2 provides a comprehensive literature review, divided into two main sections. The first section explores related work on classical risk premium modeling, focusing on how it addresses the covariate-based model risk in light of the research questions posed. The second section offers a brief overview of popular Bayesian approaches applied to risk premium prediction, again framed by the research questions that deal with the covariate-based model risk. This chapter will also be supplemented by additional material provided in Appendix B.

Chapter 3 outlines the key theories that form the foundation of the series of the methodologies employed to answer the five research questions — RQ1.1, RQ1.2, RQ1.3, RQ2.1, and RQ2.2 — in this thesis. This chapter is structured into five sections. It begins with an overview of the fundamental concepts of risk premium

prediction. The second section discusses our approach to managing model risk arising from the inclusion of complete covariates, while the third addresses model risk associated with incomplete (missing or mismeasured) covariates. The fourth section explores the core theories underlying the Bayesian frameworks adopted in this thesis, emphasizing their key features. Finally, the chapter concludes with a brief discussion on model evaluation methods and the overall modeling process employed in this research.

Chapters 4, 5, and 6 apply the proposed methods to real-world datasets, extending their applicability across diverse analytical contexts shaped by different model risk scenarios. Chapter 4 addresses research questions RQ1.1 and RQ2.2, while Chapter 5 focuses on RQ1.1, RQ1.2, and RQ2.1. Chapter 6 covers RQ1.1, RQ1.2, RQ1.3, and RQ2.2. Both Chapters 4 and 6 emphasize model risks related to non-differential Berkson (NDB) mismeasurement scenario, while Chapter 5 focuses on covariates with missing at random (MAR) scenario. These chapters aim to showcase our novel connections between key Bayesian frameworks and specific measurement error correction or missing data retrieval techniques. Through numerical experiments, we demonstrate how our approaches correct flawed models and restore their utility for accurate risk premium prediction.

Finally, Chapter 7 summarizes the conclusions drawn from the experimental results and theoretical discussions presented throughout this thesis.

Chapter 2

Review of Rival Methods: Classical vs Bayesian

This chapter explores established methodologies, which serve as competing approaches in this thesis, for risk premium development from the perspective of both classical and Bayesian paradigms. Although this thesis mainly focuses on Bayesian methods, both paradigms can have distinct strengths and weaknesses. For instance, Bayesian risk premium models typically require more inputs and involve additional assumptions regarding the parameter distributions (Gelman and Carlin 2013). They rely on prior knowledge about the parameters, which can enhance model robustness in the presence of incomplete data, but this reliance can also introduce potential biases if the choice of prior is not accurate. On the other hand, classical risk premium models, while often simpler and less dependent on parameter knowledge, may struggle with complex data structures and lack effective ways to incorporate uncertainty about parameter estimations (Dudley 2006).

In this regard, we survey different methods from both Classical and Bayesian perspectives, focusing on concerns related to covariate-based model risk — heterogeneity (RQ1.1), missing data (RQ2.1), measurement error (RQ2.2) — as the major part of the risk scenario elaborated previously. Note that other covariate-based model risks arising from convolution error (RQ1.2) and scalability (RQ1.3) are dis-

cussed in Chapter 3 because they represent specific technical problems that hinder the modeling process for the risk premium development, rather than issues related to the modeling paradigm itself.

Section 2.1 describes non-Bayesian, classical regression-based approaches useful for risk premium estimation, and discusses how they deal with the model risk. Section 2.2 views the risk premium estimation and the model risk treatments through the lens of a Bayesian paradigm. It should be noted that the focus of this review is not on providing the details of State-of-the-art risk premium models or a full survey of them, but on delivering a core insight into the comparison of these two modeling paradigms — classical or Bayesian — and examining their limitations in improving the current risk premium modeling framework.

2.1 Classical Risk Premium Modeling

Classical risk premium prediction relies on a regression-based statistical method to describe the stochastic relationships between the insured claim amounts and the covariate information on the policyholders (or insured risk). Its main principle is founded on common, standard distributions derived from Asymptotic Maximum Likelihood theory (Werner and Modlin 2010). The primary concern in classical risk premium modeling is to properly allocate the premiums while maximizing the prediction accuracy (of the insured claim amount) by capturing independent, identical segmentations of the insured claims. This is because the homogeneity in a risk class reflects a policyholder's coherent characteristics, and this helps address the inherent stochasticity observed in the claim amount data (Ohlsson and Johansson 2010). It is also known that capturing the i.i.d. segmentations of the insured claims helps reduce the insurers' risk of adverse selection[1] (Brockman and Wright 1992).

[1] Adverse selection refers to the tendency of high-risk individuals to take out the policy, as they perceive a greater chance of experiencing the insured event (Brockman and Wright 1992).

2.1.1 Model Risks: RQ1.1, RQ2.1, RQ2.2

In this section, we explore rival methods, specifically classical risk premium modeling approaches, to address the covariate-based model risks listed below:

- RQ1.1. Heterogeneity
- RQ2.1. Missingness at Random (MAR)
- RQ2.2. Non-Differential Berkson error (NDB)

(A) **Classical models and RQ1.1**

For modeling risk premium, classical regression-based modeling approaches such as Generalized Linear Models (GLMs) in (**A.1**), Generalized Additive Models (GAMs) in (**A.2**), and Multivariate Adaptive Regression Splines (MARSs) in (**A.3**) are widely used (Francis 2003; Frees 2009; Derrig and Meyers 2014).

(A.1) Generalized Linear Models (GLMs) have established a solid foundation for premium and rating factor analysis in insurance practices due to their interpretability and theoretical soundness. In addition, potential issues with outliers or model fit can be easily identified via a residual analysis (Nelder and Wedderburn 1972). GLMs extend the conventional linear regression by allowing the outcome variable to follow any distribution from the exponential family — Gaussian, Poisson, Gamma, etc. (defined in Equation (B.3) in Appendix B); therefore, it can be used to fit both claim counts and claim amounts for risk premium modeling (Myers and Montgomery 1997). Assuming a continuous distribution (such as Gaussian, Gamma, etc.) for the claim amount outcome, the issue of heteroscedasticity[2] can often arise, and Hooper 1993 suggests to use the Weighted Least Squares (WLS) technique when fitting GLMs. Given a discrete distribution (Poisson, Binomial, etc.) for the claim count outcome, the issue of overdispersion[3] can often become a con-

[2]Heteroscedasticity refers to variability in the claim amount data that changes across different subsets of known/unknown risk factors. This violates the assumption of constant variance of residuals, which leads to data heterogeneity issue. See Šoltés et al. 2019.

[3]Overdispersion refers to high variability in the claim count data, much larger than its mean due to the effect of known/unknown risk classes. This violates the Poisson assumption, which leads to data heterogeneity issue. See Winkelmann 2008

cern. In response, Brockman and Wright 1992 propose the GLM based on Negative Binomial distribution. The most standard approach to addressing heterogeneity in GLMs is the inclusion of an extra class-specific term in its linear predictor (Wu 2009). This allows GLMs to handle variability at different levels or classes of the data hierarchy. Considerable literature on the use of GLMs for the data heterogeneity in risk premium modeling is available; see e.g Lloyd-Smith 2007; Shi and Valdez 2014; Fu 2015; Strežo et al. 2019; Wahl et al. 2022.

A major limitation of GLMs, however, is that it does not fully address unobserved risk factors that can contribute to heterogeneity. This is because the GLM assumes that all heterogeneity originates from the known covariates in the model, which is not the case in practice (Spedicato et al. 2018). Additionally, the GLM structure in the risk premium model is restricted to linear relationships between the transformed claim amounts and rating factors, and thus does not account for the non-linear effects of risk factors, a common occurrence in practical scenarios. (Parodi 2023). For further details on GLMs, refer to Appendix B.

(A.2) Generalized Additive Models (GAMs) extend the capabilities of GLMs, offering a high degree of flexibility as its primary feature. Similar to GLMs, GAMs also cope with the heterogeneity issue by adding an extra class-specific term to its linear predictor (Hastie et al. 2009). However, being equipped with special smoothing terms (i.e. piecewise polynomials denoted as $\Sigma_{m=1}^{M}\beta_m\ h_m(\mathbf{x}_p)$ in Equation (B.4) in Appendix B) that are adapted from the field of numerical analysis, GAMs manage to handle non-linear effects of risk factors (Brockett et al. 2014). The smoothing term helps us move beyond the linearity by using a nonparametric approach, determining its functional form based on the data at hand. They are developed separately for each covariate or combination of covariates (for interaction) and then combined to construct the final linear predictor. Their main drawback is that, unlike GLMs, risk

variations caused by the categorical covariates cannot be modeled (Denuit and Lang 2004). For more details on GAMs, see Appendix B.

(A.3) Multivariate Adaptive Regression Splines (MARSs) are another advanced technique, representing a further development of GLMs. Just like GAMs, MARSs are also renowned for their capability to effectively model non-linear functions. MARSs use the same form of linear predictor as that shown in Equation (B.4) in Appendix B, but the basis function $h_m(\mathbf{x}_p)$ in the smoothing term $\Sigma_{m=1}^{M}\beta_m\ h_m(\mathbf{x}_p)$ is defined differently. Unlike GAMs that fit piecewise curvature lines to data, MARSs break data into multiple intervals, and then fit piecewise linear regressions (straight lines) to these intervals. This makes MARSs computationally faster than GAMs, and particularly suitable in dealing with issues relating to high dimensionality (Francis 2003). For a more in-depth discussion on MARSs, see Appendix B.

In risk premium modeling using GLMs, GAMs, and MARSs, addressing the issue of heterogeneity involves incorporating an additional class-specific effect term into the linear predictor. The idea is that the hidden properties of each risk cluster can be captured through this term, which represents the unique deviation from each cluster mean (Ohlsson and Johansson 2010). However, we argue that this approach places excessive reliance on the researcher's knowledge of the risk classes $j = 1, \cdots J$. The diverse range of risk scenarios may not be easily identifiable — making it unclear how many risk classes should be considered in the first place — since the aggregate claim outcome data often reflects various levels of variation, classes, or unknown structures.

(B) **Classical models and RQ2.1 + RQ2.2**

Depending on the relationship between outcome and covariates, the incompleteness can be classified into different mechanisms. One may refer to Rubin 1976; Fewell 2007; Nab et al. 2021 for a detailed review of these classifica-

tions. In this thesis, we place a particular emphasis on the Missing at Random (MAR) assumption for missingness (Bhaskaran and Smeeth 2014) and the Non-differential Berkson Error (NDB) assumption for mismeasurement (Zhang 2010).

Brief Definitions: Let $\mathbf{x}$, x_i^- and x_i^* represent a covariate vector, missing, and mismeasured data point respectively. MAR is characterized by missingness unrelated to the value of the covariate itself, but largely related to the outcome variable or other covariates: i.e. $f(x_i^-|Y, \mathbf{x}, \mathbf{z}) = f(x_i^-|Y, \mathbf{z})$. NDB is defined in the opposite manner, where the mismeasurement is unrelated to the outcome variable or other covariates, but has a strong correlation with the value of the covariate itself: i.e. $f(x_i^*|Y, \mathbf{x}, \mathbf{z}) = f(x_i^*|\mathbf{x})$. (See Appendix A for the variable definitions. Note that a further examination of NDB error is provided in Section 3.3.2). When it comes to model risk in the case of incomplete covariates, this thesis is mainly developed upon the assumptions of MAR and NDB covariates for the sake of simplicity. It is also known that the assumptions of MAR and NDB covariates provide more attainable conditions that allow the use of well-established data correction techniques in the literature (Graham 2009; Carroll et al. 2006). In what follows, we briefly review the well-known data correction techniques for the classical regression-based modeling framework. This includes the Expectation-Maximization (EM) algorithm in (**B.1**), Regression Calibration (RC) in (**B.2**), and Simulation Extrapolation (SIMEX) in (**B.2**), which are based on the assumptions of MAR and NDB covariates.

(B.1) EM Algorithm with Missing at Random (MAR): The Expectation-Maximization (EM) algorithm is arguably one of the most popular approaches for recovering the missing covariate values under the MAR assumption (Ng and Krishnan 2012). The basic idea is to recover the missing values by exploiting the interdependence between missingness and the model parameters based on the MAR assumption. In other words, it is an iterative process of alternating

between model parameter estimation and the mixing weight approximation in a mixture model for missing data recovery.

To be specific, for observation $i = 1, \cdots n$, let $\mathbf{x} = \{x_{z_1}, x_{z_2}, \cdots, x_{z_n}\}$ be an observed covariate vector that has missing values, and $\mathbf{z} = \{z_1, z_2, \cdots, z_n\}$ be a missingness indicator (membership) variable where '0 : likely to be non-missing' and '1 : likely to be missing' regarding the covariate $\mathbf{x}$. In accordance with the MAR assumption, it is possible that the cluster membership can be determined by another binary covariate correlated to the missingness in $\mathbf{x}$. These two pieces of covariate information can be joined together to define the complete probability distribution (a mixture of clusters) of the covariate $\mathbf{x}$ that has missing values (G. McLachlan 2007).

$$p(x|\theta) = \sum_{z_i} p(z_i, x_{z_i}|\theta_{z_i}) = \sum_{z_i} \boldsymbol{\omega}(z_i|x_{z_i}, \theta_{z_i})\ p(x_{z_i}|\theta_{z_i}) \tag{2.1}$$

where the term $\boldsymbol{\omega}(z_i|x_{z_i}, \theta_{z_i})$ is translated into the mixing weight with the model parameter θ_{z_i} that explains the property (location, scale, etc.) of the cluster $p(x_{z_i}|\theta_{z_i})$; therefore, understanding the model parameter θ_{z_i} helps the computation of the mixing weight that is the probability that x_{z_i} belongs to either cluster '$z_i = 0$' (where $x_{z_i=0}$ does not show a similar feature of missingness) or cluster '$z_i = 1$' (where $x_{z_i=1}$ shows a similar feature of missingness). These cluster-based evaluations are useful in estimating the value of the missing covariate, but they require clarification of the properties of both clusters by estimating the cluster parameters $\theta_{z_i=0}$ and $\theta_{z_i=1}$. However, this cannot be achieved using traditional Maximum Likelihood Estimation (MLE) due to the presence of multiple unknown parameters: θ_{z_i} and z_i.

The EM algorithm offers a solution for this using the 'provisional' parameter values $\theta_{(t)}$, and a detailed description of it is presented in Appendix B. In short, the EM algorithm iteratively takes following two steps until the parameter estimates for θ_{z_i} converge (Ng and Krishnan 2012)

- E-Step. Estimate the probability of each data point that is labeled as 'missing" or 'not-missing" — $x_{z_i=1},\ x_{z_i=0}$ — by approximating the mixing weight $\boldsymbol{\omega}(z_i|x_{z_i}, \theta_{(t)})$ in Equation (B.9) in Appendix B. Impute the missing covariate value, using Equation (2.1).
- M-Step. Subsequently, update (re-estimate) the next provisional parameter value $\theta_{(t+1)}$ through maximizing the objective function in Equation (B.8) in Appendix B.

The EM algorithm is relatively straightforward to implement and the inference is intuitive and transparent. However, this convenience comes at a cost: Since the EM algorithm imputes the 'most likely' values for the missing covariate, there is a clear uncertainty associated with this imputation process. This means that the EM algorithm lacks robustness to address the uncertainty associated with the imputed values (Kofman and Sharpe 2003). Another shortcoming is that it often converges to a 'poor' local optimum when the likelihood function is too complex or multi-modal, etc. (G. McLachlan 2007).

(B.2) RC and SIMEX with Non-Differential Berkson error (NDB):
It is known that Regression Calibration (RC) is a commonly used approach to dealing with the NDB issue in a conventional linear regression analysis. However, Fraser and Stram 2012; Agogo et al. 2014 show that the non-linearity modeling with GLMs does not invalidate the use of RC when unbiased reference measurements (the gold standard) are available. The idea of RC is intuitive: It corrects the errors by utilizing the relationship between the mismeasured covariate and the true covariate; this is followed by regressing on the expected true covariates conditional on the given mismeasured covariates (Freedman et al. 2004). However, modeling the relationship between the mismeasured covariate and the true covariate requires additional gold standard information on the true covariate that can be obtained from extra studies conducted in a smaller sample of the relevant cohort, etc. This process can be expensive and is often impractical in real-life data analysis settings (Carroll

et al. 2006). Skrondal and Kuha 2012 also claim that the parameter estimations with RC can be inconsistent. This inconsistency is directly proportional to the size of the error variances σ_ϵ^2 in Equation (B.10a) in Appendix B. A further description of the RC method is detailed in Appendix B.

As an alternative approach suggested by Cook and Stefanski 1994, the Simulation-Extrapolation (SIMEX) method shares the simplicity of RC, but directly corrects the parameter estimates through simulation instead of predicting the true covariate values. The underlying concept of SIMEX is that simulation can be used to assess the impact of measurement error in the covariate by intentionally introducing artificial noise. By controlling the magnitude and variance of the noise in the simulation, one can observe changes in the model parameter estimates, and SIMEX models these changes as a function of the noise. Consequently, the optimal estimates of the model parameters can be obtained by setting the noise level equal back to zero (Oh et al. 2018). Additional details about the SIMEX method can be found in Appendix B.

Unlike RC, SIMEX does not require external reference data for the error correction, but having sufficiently large data leads to better correction accuracy. SIMEX might require gold standard data as well to calibrate or validate the simulation process (Carroll et al. 2006). The main obstacle for SIMEX, however, lies in the risk of inaccurate extrapolation, stemming from the intricate relationship between parameters and error variance (Oh et al. 2018). While it is common to employ a simple quadratic curve as the extrapolation function for ensuring numerical stability, Carroll et al. 2006 demonstrates that parameter estimation may still be inconsistent until the extrapolation curve thoroughly encapsulates the complex relationship, which remains its historic Achilles' heel.

So far, we have explored a set of rival methods based on classical predictive modeling frameworks in the literature to get the idea of covariate-based model risk in risk premium development. These methods include GLMs, GAMs, MARS, EM

algorithm, RC, and SIMEX. Although the classical framework utilizes standard distributions and straightforward parameter estimation processes, it lacks the flexibility to accommodate unknown risk scenarios due to its reliance on the assumptions such as 'clusters are already discovered', 'all data are valid and properly measured', etc. The EM algorithm, RC, or SIMEX can be worth considering to mitigate the adverse impact of poor data quality. However, their application of a long-term repeated frequency philosophy (based on *Asymptotic Maximum Likelihood theory* (Sundberg 1974)) to individual modeling cases renders the correction accuracy overly dependent on sample size and overly optimistic about the estimation uncertainty being small. This issue is known as the paradox of long-run frequencies and it often leads to certain inconsistencies in the parameter inference (Dudley 2006).

In the following section, we attempt to move beyond the paradox of long-run frequencies. We transition to the rival methods based on a Bayesian predictive modeling framework, and review well-established Bayesian techniques, discussing their advantages and disadvantages in dealing with covariate-based model risk in risk premium development.

2.2 Bayesian Risk Premium Modeling

As outlined in Section 1.2, we hypothesize that Bayesian approaches may offer advantages in terms of the three essential attributes — uncertainty propagation, explainability, combating model risk — of Parodi 2023 within the risk premium modeling framework. Over the last decade, although the Bayesian paradigm has been widely used in many scientific fields, including actuarial science, its application to risk premium modeling remains relatively limited. Makov 2001 highlights that computational challenges associated with Bayesian techniques, often involving high-dimensional mathematical integration, have been significant obstacles. However, the advent of Markov Chain Monte Carlo (MCMC) algorithms has made Bayesian methods more accessible, enabling approximate sampling from complex posterior distributions instead of requiring analytical solutions (Gelman and Carlin 2013). If

the insurer's goal is to find a better pricing approach to account for the complex risk scenarios, there should be no reason, we contend, not to consider the new risk premium modeling approach boosted with the Bayesian paradigm which combats potential model risk in a highly structured manner. In this subsection, we review Bayesian approaches that addresses the flexibility of the risk premium modeling to accommodate a wide range of potential risk scenarios.

2.2.1 Model Risks: RQ1.1, RQ2.1, RQ2.2

In this section, we navigate rival methods, specifically Bayesian risk premium modeling approaches, to address the covariate-based model risks outlined below:

- RQ1.1. Heterogeneity
- RQ2.1. Missingness at Random (MAR)
- RQ2.2. Non-Differential Berkson error (NDB)

(A) **Bayesian models and RQ1.1**

Bayesian ideas and techniques in insurance applications (such as risk premium development) first received attention when Bühlmann 1969 introduced the concept of *empirical Bayesian credibility*. It gained attraction due to the bias-variance tradeoff effect in risk premium estimation achieved by blending external knowledge into the existing local model at hand. Here, we explore popular Bayesian modeling examples that manifest a coherent way of integrating external knowledge with baseline models to address the issue of heterogeneity. This includes the Bayesian Credibility Premium model (BCP) in (**A.1**), Bayesian Generalized Additive Models (BGAMs) in (**A.2**), and Bayesian Variational Autoencoder (VAE) in (**A.3**).

(A.1) Bayesian Credibility Premium model (BCP) tackles the issue of heterogeneity in the aggregate claim data by developing a weighted average of two extremes: the global mean with a homogeneous portfolio assumption (from the external population), and the cluster mean with a heterogeneous portfolio

assumption (from the samples at hand) (Hong and R. Martin 2017). Given that each aggregate claim data belongs to a certain risk cluster $j = 1, \cdots, J$, the future value of the cluster-wise total risk premium $E[S_j]$ for each group policy can be estimated as

$$E[S_j] = E[E[S_j]|S_1, \cdots, S_J] = B_j \bar{S}_j + (1 - B_j)E[E[S_j]] \tag{2.2}$$

where $\bar{S}_j$ is the sample mean for cluster j, $E[E[S_j]]$ is the global mean, and B_j is the credibility factor (mixing weight). The primary goal of the BCP is to ascertain the credibility factor B_j, thereby deriving a fair risk premium value for each risk cluster. Specifically, B_j can be chosen based on the variance and the sample size. For instance, if the sample size of cluster j is large enough, the full credibility $B_j \to 1$ is given because the sample mean $\bar{S}_j$ is considered reliable. Likewise, if the variance of the global mean $V(E[E[S_j]])$ is too high, the full credibility $B_j \to 1$ is again given because the sample mean $\bar{S}_j$ is considered more reliable than the global mean $E[E[S_j]]$. This weighted average technique can be rationalized because it addresses the risks within the cluster j sharing population characteristics with those in other clusters, while keeping distinct cluster-specific properties (Parodi 2023). However, its primary limitation is its exclusive focus on point estimation, overlooking key distributional features — such as shape, skewness, etc. — in the aggregate claim data. A comprehensive understanding of these features is essential for accurately pricing the risk premium (Werner and Modlin 2010).

(A.2) Bayesian Generalized Additive Models (BGAMs) represent a significant advancement on the GAM-based risk premium modeling. As reviewed previously, GAMs are useful in capturing the non-linear effects of risk factors, but incapable of modeling risk variations caused by categorical risk factors. However, boosted by the Bayesian paradigm, BGAMs offer a unified approach by estimating non-linear effects of categorical risk factors and their variations

while simultaneously accounting for cluster-specific heterogeneity (Klein et al. 2014). See Appendix B for further details about BGAMs.

One limitation of BGAMs is their sensitivity to the choice of priors (external knowledge) (Denuit and Lang 2004). Hence, BGAMs often encounter difficulty in specifying appropriate prior and hyperpriors for the smoothing parameters and other components. This is because parameter inference in BGAMs is sometimes overly sensitive to the choice of priors, resulting in entirely different splines of $\Sigma_{m=1}^{M} \beta_m \; h_m(\mathbf{x}_p)$ in Equation (B.4) in Appendix B (Lang and Brezger 2004).

(A.3) Bayesian Variational Autoencoder (VAE) as a Bayesian deep learning technique has gained attraction in various domains — voice recognition, image classification, etc. (An and Cho 2015) —, but its insurance applications are not as widely studied. Jamotton and Hainaut (2023) highlight the potential for VAE applications in risk premium development in relation to addressing heterogeneity in insurance portfolios. This is due to the technology's ability to de-compose and re-compose complex aggregate claim information based on statistical dependencies among the heterogeneous attributes in the data.

The underlying idea of VAE is straightforward. First, the encoder part of VAE constructs the latent variable space, using the given input. In the scheme of the latent vector as a representation of compressed data, the abnormal parts in the data will not fit; therefore, they will not be represented in the latent space. Subsequently, the decoder part of VAE reconstructs the clean input data for a reference purpose. Appendix B offers a brief description regarding the VAE operation.

The ability of VAE to uncover the latent structure within the aggregate claim data presents a clear outlook to streamline the claim prediction process. However, designing an optimal neural network architecture and selecting proper hyperparameters to attain the best results are not trivial tasks (Tomczak and

Welling 2018).

(B) **Bayesian models and RQ2.1 + RQ2.2**

In Section 2.1.1, we have explored the frequentists' various risk premium prediction models for handling covariate-based model risk under the assumption of MAR and NDB. These models' major limitation is the inherent uncertainty of the imputation itself (Ng and Krishnan 2012) and the inconsistency of parameter estimation results, which are particularly sensitive to error variance (Skrondal and Kuha 2012). Now, we shift our focus to the Bayesian counterpart. We delve into the application of another rival method, the Bayesian Multiple Imputation (BMI) technique in the following (**B.1, B.2**) and discuss consistent parameter estimation under the assumption of MAR and NDB, while dealing with the inherent uncertainty arising from the imputation.

(B.1) Multiple Imputation with Missing at Random (MAR): It is argued in Rubin 1976 that the Bayesian Multiple Imputation (BMI) technique is an effective tool to handle MAR covariates. The BMI can have an advantage over EM algorithm due to the minimized imputation variance. As the name suggests, 'multiple imputations' generates multiple versions of a dataset by imputing missing values several times. Therefore, the key to producing reliable imputations lies in formulating an accurate imputation function from which the imputed values are drawn. Using the assumption of MAR, the imputation function is constructed by leveraging the relationship between the missing covariate values and outcome or other covariates to compute a range of probable values for missing data points. In addition, by running the imputation a large number of times, the BMI produces a distribution over the imputed values, which helps address the uncertainty about the missing values. Finally, the model parameters are estimated across many different versions of dataset, and the best parameters values are computed by averaging their estimation results (Parker 2010). See Appendix B for further discussion about the BMI with MAR covariate.

Despite the efficiency of its parameter estimation, it is important to acknowledge that the well-known drawback of the BMI is the lack of standard strategies to deal with data heterogeneity (Graham 2009). Depending on the type of analysis such as clustering, longitudinal data handling, etc., further analytical work is required to explore methods for performing the BMI with particularly complex structures.

(B.2) Multiple Imputation with Non-Differential Berkson error (NDB): Bartlett 2010 points out that the BMI can be useful for correcting mismeasured covariate values. The BMI can approach the NDB issue as a missing data problem whereby the true covariate value x_i is missing for all observations. Similar to the BMI for MAR covariates discussed in (**B.1**), the imputation function for NDB covariate should be developed first, so that multiple datasets can be created by replacing the mismeasured data x_i^* with random draws x_i from the imputation function. However, the assumption of NDB dictates that a baseline model, such as GLMs, that explains the parameters of the imputation function, should be capable of being re-parameterized to relate the gold standard (i.e. samples of true values x_i) to the mismeasured values x_i^* when employing the BMI technique. An example is provided in Equation (B.18, B.19) in Appendix B. If the relationship between the gold standard and mismeasured values is not strong enough, the BMI algorithm reflects this by increasing the imputation variance. Therefore, the BMI corrections can account for uncertainty stemming from both random imputation and measurement errors (Hutcheon et al. 2010).

There are a few limitations to using the BMI for NDB error correction. Firstly, similar to the MAR case, the BMI can be prone to the issue of data heterogeneity without proper strategies (Bartlett 2010). In addition, accurate error correction of the BMI heavily relies on the availability of gold standard data. This dependency exists because its error correction process is contingent upon accurately understanding the relationship between the gold standard and the

mismeasured values (White 2006). However, in practice, the gold standard data is not always available and may sometimes be subject to additional, unknown errors, which can exacerbate the bias of the correction (Cole et al. 2006).

2.3 Summary

Up to this point, we have explored various rival methods rooted in Bayesian frameworks throughout the literature to address covariate-based model risks. These methods include BCP, BGAMs, VAE, and BMI. Unlike non-Bayesian classical approaches, which often lack the flexibility to accommodate uncertainties and impose overly strict assumptions on various aspects of modeling process, the full Bayesian approaches in general tend to be more versatile for tackling a much wider range of modeling challenges in risk premium development (Stamey and Seaman 2021). As we have seen, BCP, BGAMs, and VAE are adept at handling complex hierarchical structures and benefit from the inclusion of prior or hyperprior knowledge to address issues of heterogeneity. MI, through multiple datasets of imputed values, offers a comprehensive picture of the uncertainty in parameter estimates and traces the uncertainty propagation when dealing with incomplete covariates.

While these techniques effectively address specific problems, it is true that they remain isolated, tackling only parts of the broader model risk landscape. However, by leveraging the flexibility of the Bayesian framework, we can develop a more robust and unified approach for predicting risk premiums across a wide range of model risk scenarios. Building on the Bayesian and non-Bayesian techniques discussed so far, the upcoming chapter will explore a series of state-of-the-art techniques corresponding to each of these covariate-based model risks in greater detail. We seek to enhance the model's applicability in the context of various covariate-driven challenges. The systematic investigation of these integrations will be carried out across Chapters 4, 5 and 6.

Chapter 3

Methods: Combating Covariate-based Model Risk

Building on the intuition set down in Chapter 1 and Chapter 2, this chapter now restates the insurance risk premium problems with a bit more precision, along with an overview of the key concepts we propose to address them. At its core, this thesis aims to mitigate the covariate-based model risks across different scenarios, and improve the accuracy of risk premium prediction by using a Bayesian framework. To

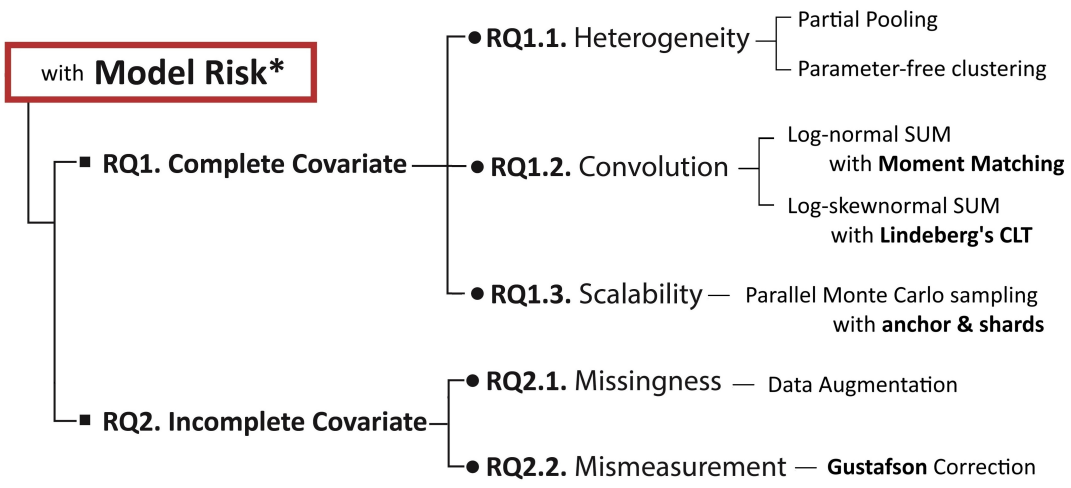

Figure 3.1: Roadmap outlining the five RQs related to the covariate-based model risk and the corresponding theories and established techniques adopted in this thesis to answer each RQ.

assist readers, Figure 3.1 presents a roadmap of the core techniques that form the

basis of this thesis's contributions. To be clear, the primary innovation of this work lies in enhancing the applicability of the Bayesian risk premium model, expanding its scope around these five research questions and integrating state-of-the-art techniques outlined in Figure 3.1. The integrations and applications are presented in Chapter 4, 5, and 6.

3.1 $E[S_h]$, $E[\tilde{S}]$, and Inclusion of Covariates

In order to study the full predictive distribution of the total aggregate claim amount $\tilde{S} = \Sigma_{h=1}^{H} S_h$, we start by scrutinizing each single summand S_h, and identifying any inherent issues that could undermine the reliability of the modeling process. Suppose the individual claim amount $Y_{hi},\ i = 1, 2, \cdots, N_h(t)$ associated with $N_h(t)$ different insured assets within a single group policy h is log-normally distributed due to its right-skewed nature and heavy-tailed behavior (Wuthrich 2020). $N_h(t)$ follows a Poisson process (Kaas et al. 2008) that describes the claim count for a single group policy h during a policy period t. Each policy (as an individual observation) h has a different $N_h(t)$ value because each contract (policy) is associated with a different group of insured assets requiring insurance protection. Hence, for each policy h, the aggregate claim amounts $S_h(t)$ received by an insurer at the end of the policy period t can be defined as a log-normal convolution: $S_h(t) = \Sigma_{i=1}^{N(t)} Y_{hi} = Y_{h1} + Y_{h2} + \ldots + Y_{hN(t)}$, which itself is a random variable and produces a mixture of log-normal distributions.

The presence of the unknown structure that complicates the variations in the data is termed *heterogeneity* (Noroozi 2023), representing one of the primary sources of model risk in this thesis. This underscores a number of problems with the curve development for a single observation $S(t)$. One problem is the inclusion of the time parameter t, which affects the infinitesimal time interval $dt > 0$ before the end of the policy period t. This may introduce an unknown structure into the data space of Y_i, which is illustrated in Figure 3.2. Defined on a probability triple $\Omega, \mathbf{F}, P$[1] in

[1] A probability triple {sample space Ω, event space $\mathbf{F}$, and probability function P} is a mathematical framework that provides a formal definition of randomness and defines probability density functions for the random variables. See Cohn 2013.

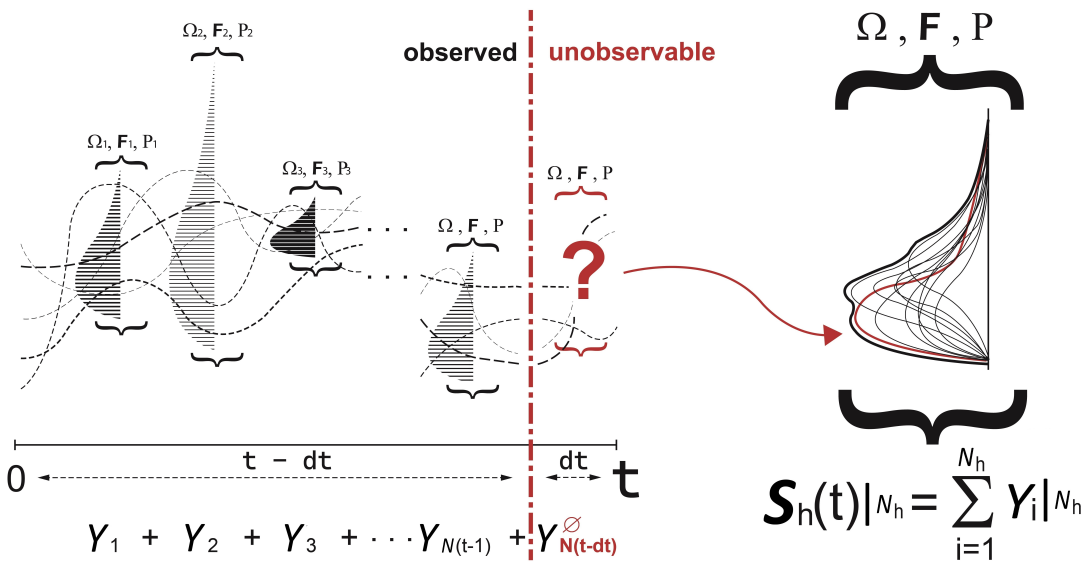

Figure 3.2: Schematics of the distribution of $S_h(t)$ for a policy h that has an unique claim count $N_h(t)$. Due to the hidden or Incurred But Not Reported (IBNR) claims $Y^{\varnothing}_{N(t-dt)}$, etc., it is difficult to obtain an informative curve for $S_h(t)$.

theory, $Y_{i=1}, \cdots Y_{i=N(t)}$ constitute a standalone stochastic process as a collection of independent, identically distributed random variables (Kaas et al. 2008). In reality, however, the collection of these random variables within a policy h is likely to experience additional random changes just before the end of a given time t due to the presence of $dt > 0$. This is because, when considering the time parameter t, the discrete number of insured assets N becomes a continuous $N(t)$, which follows the properties of a Poisson process. This dictates that there can be always at most one additional unknown claim (or *Incurred But Not Reported* (IBNR) claim) $Y^{\varnothing}_{N(t-dt)}$ within a very short time interval dt (Seri and Choirat 2015). The consequence is that the appearance of an unresolved claim $Y^{\varnothing}_{N(t-dt)}$ associated with an unknown asset could enhance heterogeneity, potentially hindering the accurate estimation of the curve of $E[S_h(t)]$.

Another significant source of inherent heterogeneity is the inclusion of covariates $\mathbf{X}$. It is assumed that the policy period t is fixed, and thus can be ignored for simplicity. In order to determine $E[S_h]$, the traditional risk modeling principle (Kaas et al. 2008) proposes the use of two key perspectives: the *frequency-severity* approach and the *compound* approach. If we assume that the summands are mutually i.i.d

to maintain homogeneity, the expected aggregate claim amount $E[S_h] = E[Y_{h1}] + E[Y_{h2}] + \ldots + E[Y_{hN_h}]$ at the policy level can be obtained by

$$\text{Frequency-Severity for a policy } h: \quad E[S_h] = E[N_h] \times E[\bar{Y}_h] \tag{3.1a}$$

$$\text{Compound for a policy } h: \quad E[S_h] = \sum_{i=1}^{N_h} E[Y_{hi}] = E[Y_{h1}] + \ldots + E[Y_{hN_H}] \tag{3.1b}$$

Moving on to a portfolio level (a collection of policies for $\{S_{h=1}, \cdots, S_{h=H}\}$), the expected total aggregate claim amount $E[\tilde{S}] = E[S_1] + E[S_2] + \ldots + E[S_H]$ received by an insurer can be obtained by

$$\text{Frequency-Severity for a portfolio:} \quad E[\tilde{S}] = E[H] \times E[S_h] \tag{3.2a}$$

$$\text{Compound for a portfolio:} \quad E[\tilde{S}] = \sum_{h=1}^{H} E[S_h] = E[S_1] + \ldots + E[S_H] \tag{3.2b}$$

Depending on the analysis goal and characteristics of the data, one can have a choice[2] between the compound approach and the frequency-severity approach.

Now, Let $\mathbf{X} = \{\mathbf{X}^F, \mathbf{X}^S\}$ represent the covariates that are statistically significant to understand $\bar{Y}_h$, N_h, and S_h in Equation (3.1) and (3.2). Before proceeding, we provide brief explanations of the conventions regarding the notation for covariates. They are used throughout the rest of this thesis:

- $\mathbf{X}^S$, by default, represents a matrix of random variables $\{\mathbf{x}^S, \mathbf{z}^S\}$ (associated with severity), while $\mathbf{X}_h^S$ refers to a fixed vector $\{x_h^S, z_h^S\}$ that is specific to a policy h. Consequently, $\mathbf{X}^S$ does not require the subscript h when used in the expression for $E[\ \cdot\ ]$ unless $\mathbf{X}^S$ specifically signifies the emergence of heterogeneity or is involved in a summation (i.e., $\sum_{h=1}^{H} E[S_h|\mathbf{X}_h^S]$).

- When we use $\mathbf{X}$ or $\mathbf{X}^S$ to describe a conditional distribution such as $S_h|\mathbf{X}^S$, we omit the subscript h from $\mathbf{X}_h^S$ because $S_h|\mathbf{X}^S$, as a single distribution, consists

[2]When the claim count and amount are independent, breaking them down to level component (using a frequency-severity principle) helps mitigate targeted risk better as they make it easier to identify statistically significant factors (Shams 2022). On the other hand, for some cases such as highly correlated claim count and amount, involving low-claim count and high-claim amount in particular, using a compound principle can be a more viable choice (Korn 2015).

of $S_h|\mathbf{X}_h^S$ across entire policies $h = 1, \cdots, H$.

- In the context of the compound approach in Equations (3.1b) and (3.2b), when S_h is conditioned on $\mathbf{X} = \{\mathbf{X}^F, \mathbf{X}^S\}$, we omit $\mathbf{X}^F$ from it since N_h is no longer treated as a random variable and no longer influences S_h. In this case, we retain the notation $\mathbf{X}$ without superscript 's' (i.e., $\mathbf{X} = \{\mathbf{x}^S, \mathbf{z}^S\}$, $\mathbf{X}_h = \{x_h^S, z_h^S\}$, $\mathbf{x} = \mathbf{x}^S$, and $\mathbf{z} = \mathbf{z}^S$).

With the inclusion of covariates $\mathbf{X}$, new, unknown structures can be introduced into the data space of $\bar{Y}_h$, N_h, and S_h. This can alter the underlying distributional properties of each individual summand - $\bar{Y}_h|\mathbf{X}_h^S$ and $S_h|\mathbf{X}_h$ - to compute their convolutions - $E[S_h|\mathbf{X}]$ and $E[\tilde{S}|\mathbf{X}]$ - because they can be re-organized into new, unknown hierarchical structures, where observations at one level can be grouped into other levels. At a policy level, based on the assumption of group policy discussed in Section 1.2, one can assume that $Y_{h1}, Y_{h2}, \cdots, Y_{hN_h}$ are influenced by the same covariate vector $\mathbf{X}_h^S = \{x_h^S, z_h^S\}$, and thus the inclusion of $\mathbf{X}$ still leads to $E[S_h|\mathbf{X}] = E[N_h|\mathbf{X}^F] \times E[\bar{Y}_h|\mathbf{X}^S]$ and $E[S_h|\mathbf{X}] = \sum_{i=1}^{N_h} E[Y_{hi}|\mathbf{X}^S]$, as shown in Equation (3.1). At a portfolio level, however, although each summand - $S_1|\mathbf{X}_1, S_2|\mathbf{X}_2, \cdots, S_H|\mathbf{X}_H$ - can still be independent, the assumption of the identically distributed observations (homogeneity) cannot stand anymore because each S_h is influenced by a particular covariates $\mathbf{X}_h$ that vary by each policy h. This heterogeneity introduced from $\mathbf{X}$ results in $E[\tilde{S}|\mathbf{X}] \neq E[H] \times E[S_h|\mathbf{X}_h]$ and $E[\tilde{S}|\mathbf{X}] \neq \sum_{h=1}^{H} E[S_h|\mathbf{X}_h]$ in Equation (3.2); therefore, these convolutions become analytically intractable under the traditional risk modeling principle (Kaas et al. 2008).

In risk premium modeling, the inherent heterogeneity highlighted by the stochasticity discussed in Figure 3.2 is significantly exacerbated by the incorporation of covariates $\mathbf{X}$. The inclusion of covariates $\mathbf{X}$ introduces a new layer of complexity that undermines the conventional risk modeling principle, thereby contributing to the inherent heterogeneity as a primary source of model risk. This covariate-based model risks set the stage for the emergence of new challenges, necessitating a thor-

ough exploration of their implications and the development of advanced modeling techniques.

The solutions proposed in this thesis remain grounded in traditional risk modeling principles described in Equations (3.1) and (3.2). Specifically, the development of the solutions is framed within the context of the frequency-severity principle, outlined in Equation (3.1a), in Chapter 4. Additionally, the compound principle, detailed in Equation (3.1b), is employed in Chapter 5 to address these complexities effectively. Furthermore, this principle, as articulated in both Equations (3.1b) and (3.2d), is again utilized in Chapter 6. However, these solutions implemented in Chapter 4, 5, and 6 go beyond the discourse of the model risk associated with the inherent heterogeneity and traditional risk modeling principles.

To be specific, as briefly mentioned in Chapter 1, the inclusion of covariate $\mathbf{X}$ can introduce further sources of model risks in addition to the inherent heterogeneity. We can classify them into two main categories based on the covariate quality. The first category involves the model risk with complete covariates, where all relevant data points are accurately captured, but challenges stem from mathematical complexities. This includes convolution issues when estimating aggregate effects and scalability concerns as sample sizes grow, requiring more efficient computational methods. The second category focuses on model risk with incomplete covariates, where data quality suffers from missing values or excessive noise. Here, the integrity of the modeling process is at risk, as inaccuracies can lead to biased estimations and flawed conclusions. This situation demands careful data correction techniques to mitigate the impact of these deficiencies on model performance.

The following sections will systematically explore these covariate-based model risks, dividing them into two categories - Complete covariate case / Incomplete covariate case - and examine their implications and corresponding solutions. Our proposed solutions aim to address each type of model risk, emphasizing the theoretical foundations and practical applications. Collectively, these solutions are aligned with the thesis's overarching goal of mitigating the covariate-based model risk within

the Bayesian modeling framework, ensuring a more reliable risk premium prediction of $E[S_h|\mathbf{X}]$ (for a policy level) and $E[\tilde{S}|\mathbf{X}]$ (for a portfolio level).

3.2 RQ1. Complete Covariate Case: (Model Risk arising from conventional issues)

When additional modeling bias emerges from the complete covariate case (where all covariate values are accurately measured), as introduced in Chapter 1, several conventional issues can be considered as the sources of model risks: RQ1.1 heterogeneity, RQ1.2 convolution error, and RQ1.3 scalability issues. In light of these challenges, this section presents the Bayesian approaches adopted in this thesis to mitigate such identified model risks. Furthermore, we propose specific modeling adjustments and variations, such as the log-skewnormal approximation (Li 2008) and parallel MCMC simulations (Ni et al. 2020), etc., designed to streamline the implementation of the Bayesian techniques and enhance the robustness of the risk premium modeling process against the model risks.

3.2.1 Handling Heterogeneity for $E[S_h|\mathbf{X}]$ with RQ1.1

Derrig and Meyers 2014 argue that the key to mitigating the heterogeneity arising from covariates in the model lies in how to re-conceive the significant clusters by capturing unobservable features underlying in the relationship between the outcomes and covariates. With regard to this, this thesis focuses on two Bayesian techniques: 1) *partial pooling* and 2) *parameter-free clustering* to navigate the sharing information across the risk clusters, while taking into account the distinctive information of each cluster at the same time.

(A) **Bayesian Partial Pooling technique**

The idea of the partial pooling technique by Gelman and Carlin 2013 is that, in the presence of the risk clusters, data heterogeneity can be resolved by

compromising between two extremes - global parameter estimation without the clusters (*complete pooling*), and separate estimations of local parameters for each cluster (*no-pooling*). To put it simply, it is about the marriage between the single global model and the multiple local models, and this marriage allows for the sharing of information across different clusters.

Considering the risk clusters for $j = 1, \cdots, J$ identified, the classical regression approach adds indicator variables along with other covariates to explain clusters $j = 1, \cdots, J$, but this puts a limit on the interaction between the clusters and is thus subject to the data heterogeneity issue. Instead of adding the indicator variable, Gelman and Hill 2007 suggests adding a *varying coefficient* term for the partial pooling in the regression to account for the dependency between observations in different clusters. As a random variable or a model in itself, the varying coefficient explains the distinctive feature of individual clusters while modeling the data at the global level.

With the presence of risk clusters $j = 1, \cdots J$, suppose the aggregate claim data S_h follows a log-normal distribution for example, and we fit a simple 'varying coefficient model' on a log scale

$$\begin{aligned} lnS_h &\sim \mathbf{N}\Big(E[lnS_h|\mathbf{X}],\ \sigma^2_{lnS}\Big) \\ E[lnS_h|\mathbf{X}] &= E[\alpha_{[j]}] + \mathbf{X}^T\boldsymbol{\beta},\ \text{for } j = 1, \cdots, J \\ \alpha_{[j]} &\sim \mathbf{N}(\mu_\alpha,\ \sigma^2_\alpha) \end{aligned} \tag{3.3}$$

where $\boldsymbol{\beta}$ is a vector of regression parameters, σ^2_{lnS} denotes the variance of the outcome on a log scale, and the varying intercept $\alpha_{[j]}$ has its own normal density with the parameters - μ_α, σ^2_α -, describing the total cluster mean and variance respectively. Depending on $\alpha_{[j]}$, the model in Equation (3.3) can convey both the no-pooling model and the complete pooling model. In the no-pooling setting, $\alpha_{[j]}$ has J different values, each of which represents a unique feature of each cluster. In the complete pooling setting, $\alpha_{[j]}$ has a single

constant value that represents the overall feature of the data at the population level. Accordingly, the predicted log scale value for a given covariate vector $\mathbf{X}_j$ in cluster j is determined through partial pooling, where $\alpha_{[j]}$ is approximated as a weighted average of the no-pooling intercept estimate, $\overline{lnS_j} - \overline{\mathbf{X}}_j^T \boldsymbol{\beta}_j$, and the complete-pooling intercept, μ_α. Gelman and Hill 2007 provide a useful expression for the partial pooling of $\alpha_{[j]}$

$$E[\alpha_{[j]}] \approx \frac{n_j/\sigma^2_{lnS_j}}{n_j/\sigma^2_{lnS_j} + 1/\sigma^2_\alpha} \cdot (\overline{lnS_j} - \overline{\mathbf{X}}_j^T \boldsymbol{\beta}_j) \; + \; \frac{1/\sigma^2_\alpha}{n_j/\sigma^2_{lnS_j} + 1/\sigma^2_\alpha} \cdot \mu_\alpha \qquad (3.4)$$

in which n_j is the sample size in the cluster j, $\sigma^2_{lnS_j}$ is the variance of the cluster j (within-cluster variance), and σ^2_α is the total cluster variance. As for the interpretation of Equation (3.4), if the cluster j has small samples (i.e. $n_j \to 0$) or the overall data gives small and reliable total cluster variance (i.e. $\sigma^2_\alpha \to 0$), then the weighting pulls $E[\alpha_{[j]}]$ toward the global intercept μ_α and otherwise, closer to the local cluster intercept $\overline{lnS_j} - \overline{\mathbf{X}}_j^T \boldsymbol{\beta}_j$. This partial pooling allows for optimally sharing information between the clusters, thus the prediction can be made with a balance between cluster-level variation and individual-level variation, which greatly mitigates the varying degrees of heterogeneity across the risk clusters.

(B) **Bayesian Parameter-free Clustering technique**

Neal 2000 suggests the parameter-free clustering technique to cope with data heterogeneity by promoting the potential of each data point to form a new, distinct risk cluster with/without the inclusion of any other data points. As in the case of partial pooling, the parameter-free clustering also relies on the multilevel structure. While the multilevel structure accommodates correlations between data points across clusters, the parameter-free clustering algorithm exploits an infinite number of clustering simulations, and reveals the optimal clustering scenario that best addresses the within-cluster correlations. The concept of the infinite clustering simulation is in contrast to the tradi-

tional partial pooling technique, which operates under the assumption that the clusters should be predetermined beforehand. If the aim of partial pooling is to capture the correlation across the given clusters, that of parameter-free clustering is to accommodate the correlation by fabricating new clusters or unknown covariates information to explain the correlation.

To perform parameter-free clustering, a Gibbs sampler[3] is used to ensure the convergence of parameter estimates, considering the choice of the risk cluster membership. The following steps can be considered:

Step a) Consider a set of risk clusters $j = 1, \cdots, J$, in which cluster j has data on outcome and covariates $\{S_j, \mathbf{X}_j\}$ with parameters $\{\boldsymbol{\theta}_j, \boldsymbol{w}_j\}$, but the true J is unobservable. The likelihood components (outcome and covariate model) are defined as $f(S_j|\mathbf{X}_j, \boldsymbol{\theta}_j)$ and $f(\mathbf{X}_j|\boldsymbol{w}_j)$. First, risk cluster membership j is initialized to some reasonable clustering of the data using e.g., hierarchical or k-means (Gershman and Blei 2012).

Step b) Let $\boldsymbol{\theta}_j$ and $\boldsymbol{w}_j$ be the parameters for the outcome model and covariate model respectively tied to the cluster j. Since all observations have been assigned to a particular cluster j in Step a), the parameters of both outcome and covariate model for each cluster $j = 1, \ldots, J$ can be initialized using the posterior form of $\boldsymbol{\theta}_j \sim p(\boldsymbol{\theta}|S_j, \mathbf{X}_j)$ and $\boldsymbol{w}_j \sim p(\boldsymbol{w}|\mathbf{X}_j)$.

Step c) Once the cluster memberships and parameter values have been initialized, we then loop through the Gibbs sampler sufficiently where the cluster memberships assignment and parameter updates are interleaved at each iteration. Each iteration consists of the following tasks - (i),(ii),(iii):

(i) **Re-assigning cluster memberships:** Given that the initial clustering is done, we re-define all clusters for each data point (obser-

[3]Gibbs sampling, as a special case of Metropolis-Hastings (MH) algorithm, is useful when the full conditional distributions are tractable, and the model involves multiple dependent variables, resulting in complex joint distributions (Neal 2000). The parameter-free clustering requires updating both cluster assignments and model parameters in a conditional and cyclic manner, which can be efficiently done by a Gibbs sampler: sequential sampling from the conditional distributions of each variable, given the current values of the other variables.

vation or a policy in this thesis) h using cluster membership index $s_h = 1, \cdots, J$ for observation $h = 1, \cdots, H$. Next, we must see if new clusters need to be added, where this is done by comparing two probabilities:

- Probability of observation h entering into existing discrete cluster (i.e. $s_h = j$) is computed as

$$Pr(s_h = j) = C \cdot \frac{n_j^{-h}}{H - 1 + \alpha} \cdot f(S_h|\mathbf{X}_h, \boldsymbol{\theta}_j) f(\mathbf{X}_h|\boldsymbol{w}_j) \qquad (3.5)$$

  where $f(S_h|\mathbf{X}_h, \boldsymbol{\theta}_j)$ is an outcome model and $f(\mathbf{X}_h|\boldsymbol{w}_j)$ is a covariate model.

- Probability of observation h entering into a new continuous or parameter-free cluster (i.e., $s_h = J+1$: the creation of new cluster)

$$Pr(s_h = J + 1) = C \cdot \frac{\alpha}{H - 1 + \alpha} \cdot f_0(S_h|\mathbf{X}_h) f_0(\mathbf{X}_h) \qquad (3.6)$$

  where $f_0(S_h|\mathbf{X}_h)$ is the parameter-free model for the outcome and $f_0(\mathbf{X}_h)$ represents the covariate model, defined using Lebesgue integrals (Burkill 2004)

$$f_0(S_h|\mathbf{X}_h) = \int f(S_h|\mathbf{X}_h, \boldsymbol{\theta}) dG_0(\boldsymbol{\theta}) \qquad (3.7)$$

$$f_0(\mathbf{X}_h) = \int f(\mathbf{X}_h|\boldsymbol{w}) dG_0(\boldsymbol{w}) \qquad (3.8)$$

  Note that C is a scaling constant to ensure $\sum_{j=1}^{J+1} Pr(s_h = j) = 1$, α is a precision parameter to adjust the degree of clustering (i.e., higher α encourages the creation of more new clusters), and H, n_j, n_j^{-i} are the sample size of the entire data, of cluster j, and of cluster j excluding data point h respectively.

(ii) **Updating cluster parameters:** Once the cluster memberships have been determined, parameters $\boldsymbol{\theta}_j$ and $\boldsymbol{w}_j$ for each cluster $j =$

$1, \ldots, J$ can be updated, using the same posterior form developed in Step b). As for updating the precision parameter α, Escobar and West 1995 suggest its posterior form

$$p(\alpha|J) \propto p_0(\alpha)\alpha^{J-1}(\alpha + H) \cdot \mathbf{Beta}(\alpha + 1,\ H) \tag{3.9}$$

where $p_0(\alpha)$ is the prior distribution for α, J is the number of clusters, H is the sample size, and $\mathbf{Beta}(\cdot, \cdot)$ is the beta function. More details are provided in E.2.1 in Appendix E

(iii) **Monitoring convergence:** The log-likelihood function at each iteration of the Gibbs sampler can be computed to track convergence.

$$\ell(\boldsymbol{\theta}, \boldsymbol{w}|S, \mathbf{X}) = \sum_{h=1}^{H} \log\left[f(S_h|\mathbf{X}_h, \boldsymbol{\theta}_j) f(\mathbf{X}_h|\boldsymbol{w}_j)\right] \tag{3.10}$$

where in this case $\boldsymbol{\theta}_j$ and $\boldsymbol{w}_j$ represent the parameters associated with the cluster that observation h is in. Typically the log-likelihood function will change rapidly at the beginning of the Gibbs sampler as observations move between clusters; however, we expect it to eventually stabilize after a large number of iterations (Gershman and Blei 2012).

3.2.2 Handling Convolutions for $S_h|\mathbf{X}$, $\tilde{S}|\mathbf{X}$ with RQ1.2

With the incorporation of covariates $\mathbf{X}$ in the convolution operation, the shape of each summand as well as its dimension might change, depending on the type and range of the covariates, which increases computational complexity for the curve development on a much greater scale than illustrated in Figure 3.2. In addition to this, there is no explicit algebraic expression or formula (closed-form) that we can use to compute the probability values for the sum of $Y_{hi}|\mathbf{X}_h$ and the sum of $S_h|\mathbf{X}_h$ due to its inherent complexity (Beaulieu and Xie 2003). However, that does not mean

that proper curve development for the convolution is impossible. In this thesis, we suggest to approximate the convolution curves based on two techniques: (A) *Moment Matching* (Li 2008), (B) *Lindeberg's Asymptotic Approximation* (Chatterjee 2006).

(A) $E[S_h|\mathbf{X}]$: **log-normal convolution with Moment Matching**

Let $\mathbf{X} = \{\mathbf{X}^F, \mathbf{X}^S\}$. At a single policy level, we encounter difficulty in computing the probability density curve for the conditioned log-normal convolution: $S_h|\mathbf{X}^S, N_h, \mathbf{X}^F = \left(Y_{h1}|\mathbf{X}_h^S + Y_{h2}|\mathbf{X}_h^S + \cdots + Y_{hN_h}|\mathbf{X}_h^S\right) \mid \mathbf{X}^F$ for a policy h. By the definition of group policy in Section 1.2, we can assume that all $Y_{h1}, Y_{h2}, \cdots, Y_{hN_h}$ are conditioned on the same fixed covariate $\mathbf{X}_h^S = \{x_h^S, z_h^S\}$ (i.e. $\mathbf{X}_h^S$ does not contain random variables) within a policy h. Now we aim to identify a distribution of $S_h|\mathbf{X}^S$ by removing N_h since N_h is a random variable that varies by each policy h (i.e. for each policy h, N_h follows a count distribution with different parameters, which complicates the development of the distribution of S_h). To this end, the following is considered:

$$\begin{aligned} S_h|\mathbf{X}^S, N_h, \mathbf{X}^F &= \left(\Sigma_{i=1}^{N_h} Y_{hi}|\mathbf{X}_h^S\right) \mid \mathbf{X}^F \\ S_h|\mathbf{X}^S &= \sum_{N_h} f(S_h|\mathbf{X}^S, N_h, \mathbf{X}^F)\ f(N_h|\mathbf{X}^F) \end{aligned} \tag{3.11}$$

In this theoretical framework in Equation (3.11), we can determine the expected value of the distribution of $S_h|\mathbf{X}^S$ as below:

$$\begin{aligned} E_{S_h}[S_h|\mathbf{X}^S] &= E_{N_h}\Big[E_{S_h}[S_h|\mathbf{X}^S, N_h, \mathbf{X}^F] \mid \mathbf{X}^F\Big] \\ &= E_{N_h}\Big[E_Y[\Sigma_{i=1}^{N_h} Y_{hi}|\mathbf{X}_h^S] \mid \mathbf{X}^F\Big] \\ &= E_{N_h}\Big[\Sigma_{i=1}^{N_h} E_Y[Y_{hi}|\mathbf{X}_h^S] \mid \mathbf{X}^F\Big] \\ &= E_{N_h}\Big[\Sigma_{i=1}^{N_h} E_Y[Y_h|\mathbf{X}_h^S] \mid \mathbf{X}^F\Big] \text{ since } \mathbf{X}_h^S \text{ is fixed, } E_Y[Y_{hi}|\mathbf{X}_h^S] = E_Y[Y_h|\mathbf{X}_h^S] \\ &= E_{N_h}\Big[N_h \times E[Y_h|\mathbf{X}^S] \mid \mathbf{X}^F\Big] \\ &= E[N_h|\mathbf{X}^F] \times E[Y_h|\mathbf{X}^S] \end{aligned} \tag{3.12}$$

Based on the frequency-severity principle in Equation (3.1a), and the findings presented in Equation (3.12), we conclude that if $S_h = \Sigma_{i=1}^{N_h} Y_{hi}$ adheres to a specific distribution, then the conditional distribution $S_h|\mathbf{X} = \Sigma_{i=1}^{N_h} Y_{hi}|\mathbf{X}$ will also conform to the same distribution, as their moment expressions are equivalent. This indicates that conditioning on $\mathbf{X}$ does not alter the underlying distribution of the aggregate claims S_h despite the stochastic nature of $\mathbf{X} = \{\mathbf{X}^F, \mathbf{X}^S\}$.

Li 2008 demonstrated that the sum of log-normal variables, denoted as S_h in this thesis, can be approximated using a log-skewnormal distribution through the Moment Matching technique[4]. Hence, to compute the conditional log-normal sum curve for the policy h, we also adopt the log-skewnormal approximation method, setting our outcome model as a log-skewnormal density. The bottom line is, how to get the parameter values - location μ, scale σ^2, and shape ξ - that approximate the distribution's shape of $S_h|\mathbf{X}$ while accounting for the conditional characteristics of $\mathbf{X}$ within the log-skewnormal framework?

Here, Li 2008 also provides an idea for the derivation of the log-skewnormal parameters - μ, σ^2, ξ - from the individual log-normal random variables Y_i. For the sake of simplicity, we consider the moments of a skewnormal random variable by taking the log of LS_h, which brings the original LS_h into the plain skewnormal scale. Accordingly, the moment-matching principle dictates

$$E\big[\ ln(LS_h)\ \big] = \mu + \sigma\sqrt{\frac{2}{\pi}\Big(\frac{\xi}{\sqrt{1+\xi}}\Big)} \approx E\big[\ ln(\Sigma_{i=1}^{N_h} Y_{hi})\ \big] \tag{3.13a}$$

$$V\big(\ ln(LS_h)\ \big) = \sigma^2\Big(1 - \frac{2}{\pi}\big(\frac{\xi}{\sqrt{1+\xi}}\big)^2\Big) \approx V\big(\ ln(\Sigma_{i=1}^{N_h} Y_{hi})\ \big) \tag{3.13b}$$

$$SK\big(\ ln(LS_h)\ \big) = \frac{\frac{4-\pi}{2}\frac{2}{\pi}\sqrt{\frac{2}{\pi}}\big(\frac{\xi}{\sqrt{1+\xi}}\big)^3}{\big[1 - \frac{2}{\pi}\big(\frac{\xi}{\sqrt{1+\xi}}\big)^2\big]^{1.5}} \approx SK\big(\ ln(\Sigma_{i=1}^{N_h} Y_{hi})\ \big) \tag{3.13c}$$

where the moment formulas in Equation (3.13) are provided by Azzalini 2013.

[4]This approximation is known for high accuracy in most regions of the density curve along with the high power in the lower region. The high power refers to the ability of a hypothesis test to detect a true effect or difference when it is present (Myors and Murphy 2010).

Once the moments in the right-hand side in Equation (3.13) are properly evaluated, using the individual log-normal random variables Y_{hi}, the major parameters of our interest - μ, σ^2, ξ - can be easily obtained from the system of equations in the left-hand side. The evaluation of the moments can be performed with the classical Monte Carlo integration technique as described below (with the subscript h omitted for simplicity).

(i) **Define the integrals to be computed:**

Suppose $\underline{Y} = \{Y_1, Y_2, \cdots, Y_N\}$, $\boldsymbol{\phi}$ is the parameter vector of the joint likelihood $f(\underline{Y}|\boldsymbol{\phi})$, and $p(\boldsymbol{\phi})$ is its parameter model then the following integrals are defined by the definition of moments (Pearson 1936)

$$
\begin{aligned}
E\big[\, ln(\Sigma_{i=1}^{N} Y_i) \,\big] &= E_{\boldsymbol{\phi}}\big[\, E\big[ln(\Sigma_{i=1}^{N} Y_i)|\boldsymbol{\phi}\big] \,\big] \\
&= \int_{\boldsymbol{\phi}} \int_{Y} ln(\Sigma_{i=1}^{N} Y_i) \cdot f(\underline{Y}|\boldsymbol{\phi}) \cdot p(\boldsymbol{\phi})\, d\underline{Y}\, d\boldsymbol{\phi} \\
V\big(\, ln(\Sigma_{i=1}^{N} Y_i) \,\big) &= E_{\boldsymbol{\phi}}\big[\, V\big(ln(\Sigma_{i=1}^{N} Y_i)|\boldsymbol{\phi}\big) \,\big] \\
&= \int_{\boldsymbol{\phi}} \int_{Y} \{ln(\Sigma_{i=1}^{N} Y_i)\}^2 \cdot f(\underline{Y}|\boldsymbol{\phi}) \cdot p(\boldsymbol{\phi})\, d\underline{Y}\, d\boldsymbol{\phi} - E\big[\, ln(\Sigma_{i=1}^{N} Y_i) \,\big]^2 \\
SK\big(\, ln(\Sigma_{i=1}^{N} Y_i) \,\big) &= E_{\boldsymbol{\phi}}\big[\, SK\big(ln(\Sigma_{i=1}^{N} Y_i)|\boldsymbol{\phi}\big) \,\big] \\
&= \int_{\boldsymbol{\phi}} \int_{Y} \{ln(\Sigma_{i=1}^{N} Y_i)\}^3 \cdot f(\underline{Y}|\boldsymbol{\phi}) \cdot p(\boldsymbol{\phi})\, d\underline{Y}\, d\boldsymbol{\phi} - 3E\big[\, ln(\Sigma_{i=1}^{N} Y_i) \,\big] \\
&\quad \times \int_{\boldsymbol{\phi}} \int_{Y} \{ln(\Sigma_{i=1}^{N} Y_i)\}^2 \cdot f(\underline{Y}|\boldsymbol{\phi}) \cdot p(\boldsymbol{\phi})\, d\underline{Y}\, d\boldsymbol{\phi} + 2E\big[\, ln(\Sigma_{i=1}^{N} Y_i) \,\big]^3
\end{aligned}
\tag{3.14}
$$

(ii) **To compute the integrals, the following is repeated M times:**

- Sample the parameter $\boldsymbol{\phi}$ from the priors $p(\boldsymbol{\phi})$ that are chosen for the series of integrals defined in Equation (3.14).
- Put the parameter sample $\boldsymbol{\phi}$ into the joint $f(\underline{Y}|\boldsymbol{\phi}) \cdot p(\boldsymbol{\phi})$ in Equation (3.14). Compute the output of the functions to be integrated.
- Record each output value for each moment defined in Equation (3.14).

(iii) **With the collection of the output values, divide the sum of all output values by the number of repeats M:**

$$
\begin{aligned}
E\big[\ ln(\Sigma_{i=1}^{N} Y_i)\ \big] &\approx \frac{1}{M}\Sigma_{r=1}^{M}\Big[ln(\Sigma_{i=1}^{N} Y_i)\cdot f(\underline{Y}|\boldsymbol{\phi}_r)\cdot p(\boldsymbol{\phi}_r)\Big] \\
V\big(\ ln(\Sigma_{i=1}^{N} Y_i) &\approx \frac{1}{M}\Sigma_{r=1}^{M}\Big[\{ln(\Sigma_{i=1}^{N} Y_i)\}^2\cdot f(\underline{Y}|\boldsymbol{\phi}_r)\cdot p(\boldsymbol{\phi}_r)\Big] - E\big[\ ln(\Sigma_{i=1}^{N} Y_i)\ \big]^2 \\
SK\big(\ ln(\Sigma_{i=1}^{N} Y_i)\ \big) &\approx \frac{1}{M}\Sigma_{r=1}^{M}\Big[\{ln(\Sigma_{i=1}^{N} Y_i)\}^3\cdot f(\underline{Y}|\boldsymbol{\phi}_r)\cdot p(\boldsymbol{\phi}_r)\Big] \\
&\quad -3E\big[\ ln(\Sigma_{i=1}^{N} Y_i)\ \big]\frac{1}{M}\Big[\{ln(\Sigma_{i=1}^{N} Y_i)\}^2\cdot f(\underline{Y}|\boldsymbol{\phi}_r)\cdot p(\boldsymbol{\phi}_r)\Big] + 2E\big[\ ln(\Sigma_{i=1}^{N} Y_i)\ \big]^3
\end{aligned}
\tag{3.15}
$$

Since all quantities are obtained, one can approximate the log-normal sum curve for the policy h using the system of equations in Equation (3.13). Therefore,

$$
f(S_h|\mathbf{X}) \ \rightsquigarrow\ \mathbf{LogSN}\Big(\mu,\ \sigma^2,\ \xi\Big) \tag{3.16}
$$

and computing the expected log-normal convolution $E[S_h|\mathbf{X}]$ is feasible.

(B) $E[\tilde{S}|\mathbf{X}]$: **log-skewnormal convolution with Lindeberg's condition**

At a portfolio level, given the total aggregate claim amount $\tilde{S}|\mathbf{X} = S_1|\mathbf{X}_1 + S_2|\mathbf{X}_2 + \cdots S_H|\mathbf{X}_H$ across all group policies for $h = 1, \cdots, H$, we consider the variant of the central limit theorem (CLT), called *Lindeberg's CLT* (Chatterjee 2006), to approximate the distribution of the log-skewnormal convolution $\tilde{S}|\mathbf{X}$. In the sequence of independent log-skewnormal random variables $S_1|\mathbf{X}_1, S_2|\mathbf{X}_2, \cdots S_H|\mathbf{X}_H$, as discussed in Section 3.1, the homogeneity of each random variable in the sequence does not hold true due to the inclusion of covariate $\mathbf{X}$. Consequently, the application of the classical CLT: $\tilde{S}|\mathbf{X} \approx H \times E[S_h|\mathbf{X}]$ becomes irrelevant. However, Chatterjee 2006 shows that as long as each random variable in this collection - $S_1|\mathbf{X}_1, \cdots, S_H|\mathbf{X}_H$ - has finite mean and variance, we can replace classical CLT with Lindeberg's CLT

by relaxing the homogeneity assumption, and we will prove that this is the case in this thesis. Consider the pth moment for each log-skewnormal summand denoted as

$$E[S_h^p|\mathbf{X}] = \int_0^\infty S_h^p \cdot \frac{2}{S_h\sigma_j}\phi\Big(\frac{\log S_h - \mathbf{X}^T\boldsymbol{\beta}_j}{\sigma_j}\Big) \cdot \Phi\Big(\xi_j \cdot \frac{\log S_h - \mathbf{X}^T\boldsymbol{\beta}_j}{\sigma_j}\Big) dS_h \tag{3.17}$$

Now we set up the transformation then solve S_h in terms of W_h as

$$W_h = \frac{\log S_h - \mathbf{X}^T\boldsymbol{\beta}_j}{\sigma_j} \in (-\infty, \infty), \quad dW_h = \frac{1}{S_h\sigma_j}dS_h, \quad S_h = \exp(\mathbf{X}^T\boldsymbol{\beta}_j + \sigma_j W_h)$$

Hence, the pth moment for each summand in Equation (3.17) can be re-expressed as

$$\begin{aligned} E[S_h^p|\mathbf{X}] &= e^{\mathbf{X}^T\boldsymbol{\beta}_j \cdot p} \int_{-\infty}^{\infty} e^{W_h \cdot p \cdot \sigma_j} \cdot \underbrace{2\phi\Big(W_h\Big) \cdot \Phi\Big(\xi_j W_h\Big)}_{f_{SN}(W_h,\, \ddot{\mu}=0,\, \ddot{\sigma}=1,\, \ddot{\xi}=\xi_j)} dW_h \\ &= e^{\mathbf{X}_h^T\boldsymbol{\beta}_j \cdot p} \times M_{W_h}(p \cdot \sigma_j) \end{aligned} \tag{3.18}$$

where $M_{W_h}(p \cdot \sigma_j)$ is the moment generating function of the standard skewnormal random variable $W_h \sim SN\big(0,\ 1,\ \xi_j\big)$ when

$$2\phi\Big(W_h\Big) \cdot \Phi\Big(\xi_j W_h\Big) \ = \ \frac{2}{\ddot{\sigma}} \cdot \phi\Big(\frac{W_h - \ddot{\mu}}{\ddot{\sigma}}\Big) \cdot \Phi\Big(\ddot{\xi}_j \cdot \frac{W_h - \ddot{\mu}}{\ddot{\sigma}}\Big) \quad \text{for } \ddot{\mu} = 0,\ \ddot{\sigma} = 1,\ \ddot{\xi} = \xi_j$$

Azzalini 1985 identifies that if W is standard skewnormal distributed, then the moment generating function of W is defined as $M_W(p \cdot \sigma_j) = 2\exp(p^2 \cdot \sigma_j^2/2) \cdot \Phi(p \cdot \delta \cdot \sigma_j)$ where $\delta = \frac{\xi}{\sqrt{1+\xi^2}}$. So, one can say that the pth moment of $S_h|\mathbf{X}_h$ in Equation (3.18) as well as the mean and variance of $S_h|\mathbf{X}_h$ are given

by

$$E[S_h^p|\mathbf{X}] = \ e^{\mathbf{X}^T\boldsymbol{\beta}_j \cdot p} \times 2e^{p^2 \cdot \sigma_j^2/2} \cdot \Phi(p \cdot \delta \cdot \sigma_j) \tag{3.19a}$$

$$E[S_h|\mathbf{X}] = 2e^{\mathbf{X}^T\boldsymbol{\beta}_j + \frac{\sigma_j^2}{2}} \cdot \Phi(\delta \cdot \sigma_j) \tag{3.19b}$$

$$V(S_h|\mathbf{X}) = 2e^{2\mathbf{X}^T\boldsymbol{\beta}_j + 2\sigma_j^2} \cdot \Phi(2\delta \cdot \sigma_j) - \Big[2e^{\mathbf{X}^T\boldsymbol{\beta}_j + \frac{\sigma_j^2}{2}} \cdot \Phi(\delta \cdot \sigma_j)\Big]^2 \tag{3.19c}$$

In Equation (3.19), it is true that both means and variances in this collection - $S_1|\mathbf{X}_1, \cdots, S_H|\mathbf{X}_H$ - are finite, and accordingly, a sum of finite variances $\mathbb{S}_H^2 = \Sigma_{h=1}^H V(S_h|\mathbf{X}_h)$ is obtainable. This suggests that if for every $\epsilon > 0$, Lindeberg's condition

$$\lim_{H\to\infty} \frac{1}{\mathbb{S}_H^2} \sum_{h=1}^{H} \boldsymbol{E}\Big[\big(S_h|\mathbf{X}_h - E[S_h|\mathbf{X}_h]\big)^2 \cdot \mathbb{1}_{\left|S_h|\mathbf{X}_h - E[S_h|\mathbf{X}_h]\right| \,>\, \epsilon \cdot \mathbb{S}_H}\Big] = 0 \tag{3.20}$$

is satisfied, then a sum of the standardized distribution of $S_h|\mathbf{X}_h$ converges to a standard normal random variable, as $H \to \infty$

$$\frac{1}{\mathbb{S}_H} \sum_{h=1}^{H} \Big(S_h|\mathbf{X}_h - E[S_h|\mathbf{X}_h]\Big) \sim \mathbf{N}(0,\ 1) \tag{3.21}$$

Consequently, the distribution of the total aggregate claim amount $\tilde{S}|\mathbf{X}_h$ can be approximated as

$$f(\tilde{S}|\mathbf{X}) \ \rightsquigarrow \ \mathbf{N}\Big(\Sigma_{h=1}^H E[S_h|\mathbf{X}], \quad \mathbb{S}_H^2\Big) \tag{3.22}$$

and computing the expected log-skewnormal convolution $E[\tilde{S}|\mathbf{X}]$ is feasible.

Our proof of the Lindeberg's convergence condition in Equation (3.20) is provided in Appendix A. To our knowledge, this is the first time that Lindeberg's CLT has been studied in the context of the sum of conditional log-skewnormal random variables.

3.2.3 Handling Scalability for $\{S, \mathbf{X}\}$ with RQ1.3

As the number of data points grows, the Bayesian model can have more opportunities to incorporate new information and update parameter knowledge, which renders it a powerful tool for learning with greater precision. The key lies in scaling the Bayesian model to larger datasets while addressing computational limitations through parallel MCMC simulations.

To this end, this thesis adopts the *anchor point* and *shard* technique suggested by Kunkel and Peruggia 2020; Ni et al. 2020. The term 'anchor' refers to a designated subset of observations that serves as a common reference across individual outputs, helping to align the results. A shard, on the other hand, is a disjoint subset of the overall dataset used to run independent MCMC chains, each generating its own outputs. To facilitate parallel Monte Carlo simulations, we divide the large dataset into random shards. However, to ensure consistency across the different shards, we ensure that each retains a set of shared anchor points. These anchor points are then used to synchronize the labeling of clusters when the algorithm combines the MCMC draws from the various shards.

Let $\underline{S_X}$ denote the full dataset: $\{(S_h, \mathbf{X}_h), \cdots, (S_H, \mathbf{X}_H)\}$, $\boldsymbol{\phi}$ denote the parameters, and $\underline{S_X^{(k)}}$ denote the random shard k (disjoint set) for $k = 1, \cdots, K$. If $\underline{S_X}$ is exchangeable, the exact posterior can be expressed as

$$\begin{aligned} p(\boldsymbol{\phi}|\underline{S_X}) &\propto f(\underline{S_X}|\boldsymbol{\phi})p(\boldsymbol{\phi}) \\ &\propto \prod_{k=1}^{K} f(\underline{S_X^{(k)}}|\boldsymbol{\phi})p(\boldsymbol{\phi})^{1/K} \end{aligned} \tag{3.23}$$

where each $f(\underline{S_X^{(k)}}|\boldsymbol{\phi})p(\boldsymbol{\phi})^{1/K}$ is obtained from the parallel Monte Carlo sampling before the shard merging step. The question is how to combine the resulting samples for $k = 1, \cdots, K$ of risk cluster-specific latent structure that features a match between each sample and a certain risk cluster membership. For this, Ni et al. 2020 propose the following tasks:

Task(a) Split the data into $K+1$ disjoint sets, $\underline{(S_1, \mathbf{X}_1)}, \cdots, \underline{(S_K, \mathbf{X}_K)}, \underline{(S_{K+1}, \mathbf{X}_{K+1})}$,

where the set $\underline{(S_k, \mathbf{X}_k)}$ is the collection of observations such that $\underline{(S_k, \mathbf{X}_k)} = \{(S_h, \mathbf{X}_h) : h \in h_k\}$ where $h_k \subset \{1, \cdots, H\} = \cup_{k=1}^{K+1} h_k$ for $k = 1, \cdots, K, K+1$. The sample size of the sets $k = 1, \cdots, K$ are $\frac{H}{K+1}$ and that of the set $k = K+1$ is $H - K\frac{H}{K+1}$.

Task(b) Develop shards $\underline{S_X^{(k)}} = \{\underline{(S_k, \mathbf{X}_k)} \cup \underline{(\ddot{S}_{K+1}, \ddot{\mathbf{X}}_{K+1})}\}$ for $k = 1, \cdots, K$ where $\underline{(\ddot{S}_{K+1}, \ddot{\mathbf{X}}_{K+1})}$ denote the collection of anchor points present in every shard.

Task(c) Run the Monte Carlo simulations on each shard $\underline{S_X^{(k)}}$ for $k = 1, \cdots, K$ in parallel, and compute the cluster parameters as well as class memberships for each data point within each shard. Let $\boldsymbol{\phi}_{kj}$ and $\boldsymbol{F}_{kj}$ be the cluster parameters and the collection of the data point indices respectively in the cluster j within the shard k produced from the simulation.

Task(d) Merge the clusters across the simulation results - $\boldsymbol{\phi}_{kj}$ and $\boldsymbol{F}_{kj}$ - based on the metric to measure the difference between clusters. The measuring process and the metric are described below:

(i) Randomly partition the order of the shards $\underline{S_X^{(k)}}$ for $k = 1, \cdots, K$.

(ii) Determine $\boldsymbol{\varepsilon}$, the threshold (or level of tolerance: minimum proportion of the difference required for a cluster to remain independent).

(iii) Measure the differences between clusters across different shards, using the metric

$$\mathbf{Dist}_{kj,\, k'j'} = \frac{CNT_D(\boldsymbol{F}_{kj}, \boldsymbol{F}_{k'j'})}{CNT_C(\boldsymbol{F}_{kj}, \boldsymbol{F}_{k'j'}) + CNT_D(\boldsymbol{F}_{kj}, \boldsymbol{F}_{k'j'})}$$

where $CNT_D(\boldsymbol{F}_{kj}, \boldsymbol{F}_{k'j'})$ and $CNT_C(\boldsymbol{F}_{kj}, \boldsymbol{F}_{k'j'})$ represent the count of different and duplicate data points respectively between the two selected clusters j, j' across different shards k, k'.

(iv) Compare $\boldsymbol{\varepsilon}$ and $\mathbf{Dist}_{kj,\, k'j'}$, and merge the selected clusters if $\mathbf{Dist}_{kj,\, k'j'} < \boldsymbol{\varepsilon}$. Keep merging the cluster until the completion of comparison with all other clusters from different shards.

In short, the *anchor* $\underline{(\ddot{S}_{K+1}, \ddot{\mathbf{X}}_{K+1})}$ serves as a glue to construct the unified posterior samples without any loss of information. For a comprehensive understanding, refer to Section 6.3.3 and Figure 6.1.

3.3 RQ2. Incomplete Covariate Case: (Model Risk arising from MAR or NDB)

A model in general is only as good as the inputs fed into it. When certain data points are missing or incorrectly measured, the model can be immediately subject to a misspecification error (Swamy et al. 2010). As demonstrated by Ungolo and Kleinow 2020; Fewell 2007, the missing or mismeasured covariates can lead to biased parameter estimations because classical inference algorithms will optimize the wrong joint product, and its uncertainty quantification result is heavily affected by the quality of covariates $\mathbf{X}$. More importantly, the incorporation of incomplete covariates may exacerbate the conventional model risks discussed in Section 3.2. Again, in this case, $S|\mathbf{X}$ and $\tilde{S}|\mathbf{X}$ cannot be computed properly.

A Bayesian approach provides a natural framework to deal with the incomplete data issue due to its ability to incorporate existing parameter knowledge into the modeling process (Ma and Chen 2018). On top of that, recent advances in computing technology and simulation methods have expanded the applicability of the Bayesian approach (Gelman and Meng 2004; Gelman and Carlin 2013; G. M. Martin et al. 2024). Throughout the rest of this section, we expand our model risk study, using two covariates, one continuous and one discrete, denoted by $\mathbf{X} = \{\mathbf{x}, \mathbf{z}\}$, and the following two cases - Missing At Random (MAR) covariate and Non-Differential Berkson (NDB) covariate - from a Bayesian perspective. This consideration can be one of the major motivations behind the procurement of the datasets for the numerical experimentations in Chapter 4, 5, 6

3.3.1 Handling MAR covariates with RQ2.1

In Chapter 2, we have mentioned that Missing at Random (MAR) refers to a situation where the missingness in the covariate is related to the outcome variable or other covariates. This implies that the probability of being missing can vary from one cluster to another (Royston 2004) as each cluster formation is influenced by the unique features of the outcome conditioned on the covariates.

Muthén 2002 compiled the general latent variable approaches to capture the cluster-wise variations of the probability of being missing. The EM algorithm discussed in Section 2.1.1 is a notable example of this approach. As an optimization technique based on the cluster-wise latent structure, the EM algorithm can efficiently handle the MAR problem by alternating between the E-step (Expectation), where the expected values of the latent variables are computed, and the M-step (Maximization), where the model parameters are updated. However, as discussed in Chapter 2, the main issue of the EM algorithm is the lack of robustness to address the uncertainty in the imputation process.

With the limitation of knowledge on the observed data in terms of its size, quality, structure, etc. that might bring additional complexity to the likelihood, Tanner 1987 suggest a Bayesian version of the EM-algorithm, which is what is referred to as *Data Augmentation*. Being employed within a Bayesian framework, the data augmentation technique expands the dataset to maintain the variability in the imputations. Accordingly, the model can be exposed to a broader range of possible imputation results, and the imputation uncertainty can be quantified. Van Dyk 2001 also argues that the data augmentation technique can accelerate the convergence to the target distribution with greater precision. Such a technique replaces the 'E-step' with sampling missing values from the imputation model and the 'M-step' with refining the posterior density based on the incorporation of the prior information. Here, the imputation model refers to the full conditional joint distribution of all data, observed and unobserved conditioned on the parameters for the missing variable. Taking advantage of the manageable joint and Gibbs sampling,

any missing values can be imputed by the draws from the imputation function, thus the observed data becomes augmented.

To be concrete with the data augmentation technique, take our specific scenario as an example where we have one outcome S and two covariates $\mathbf{X} = \{\mathbf{z},\ \mathbf{x}\}$. Given that the binary covariate $\mathbf{z}$ is subject to MAR status, we seek to obtain the collection of the posterior parameters for the outcome and covariate respectively $\{(\boldsymbol{\theta}_1, \cdots, \boldsymbol{\theta}_J), (\boldsymbol{w}_1, \cdots, \boldsymbol{w}_J)\}$ where the subscript $j = 1, \cdots, J$ indicates the posterior samples associated with cluster j. Assuming that the form of the posterior for the covariate $p(\boldsymbol{w}_j|\mathbf{X}) \propto f(\mathbf{X}|\boldsymbol{w}_j) \cdot p_0(\boldsymbol{w}_j)$ is known, the desired posterior for the outcome parameter in the presence of missing data in the covariate $\mathbf{z}$ is

$$p(\boldsymbol{\theta}_j|S, \mathbf{x}, \mathbf{z}, \boldsymbol{w}_j) \propto \prod_{h=1}^{H_j} \underbrace{f(S_h|x_h, z_h, \boldsymbol{\theta}_j)}_{\text{likelihood}} \cdot \underbrace{p_0(\boldsymbol{\theta}_j)^{1/H_j}}_{\text{prior}} \rightarrow \boldsymbol{\theta}_j \text{ is our focus.} \tag{3.24}$$

However, note that the likelihood term $f(S_h|x_h, z_h, \boldsymbol{\theta}_j)$ in Equation (3.24) cannot be evaluated for the missing covariate z_h, and thus can be intractable. As a remedy to this conundrum, one can consider imputing the MAR missing value in $\mathbf{z}$ before computing the posterior parameter $\boldsymbol{\theta}_j$. To this end, an imputation model from which the missing values can be sampled should be developed. This imputation model is derived from the joint distribution, which is the product of the outcome model and the covariate model of missingness.

$$z_h \ \sim\ f(z_h|S_h, x_h, \boldsymbol{\theta}_j, \boldsymbol{w}_j) \propto \underbrace{f(S_h|x_h, z_h, \boldsymbol{\theta}_j)}_{\text{likelihood}} \cdot \underbrace{f(z_h|\boldsymbol{w}_j^z)}_{\text{prior}} \rightarrow z_h \text{ is our focus.} \tag{3.25}$$

The form of the imputation model in Equation (3.25) implies that the outcome term $f(S_h|x_h, z_h, \boldsymbol{\theta}_j)$ provides information on the outcome and other covariate values related to the missing value z_h. The covariate term $f(z_h|\boldsymbol{w}_j^z)$, on the other hand, offers information on the risk clusters in which the missing value z_h belongs to. That is to say, the imputation value z_h emerges from a combination of information on various available factors associated with the missing value of MAR.

In short, with the presence of MAR missing value z_h in the observation h, the data augmentation can be carried out by alternating Step a) and Step b) below, given the initial estimation of the posterior of $p(\boldsymbol{\theta}|S, \mathbf{X}, \boldsymbol{w}_j)$ and $p(\boldsymbol{w}|\mathbf{X})$.

Step a) Cluster-wise imputation:

(i) Sample $\boldsymbol{\theta}_j$ and $\boldsymbol{w}_j$ from the initial estimation of the posterior - $p(\boldsymbol{\theta}_j|S, \mathbf{X}, \boldsymbol{w}_j)$ and $p(\boldsymbol{w}_j|\mathbf{X})$ - to develop the imputation model.

(ii) Sample z_h from the imputation model $f(S_h|x_h, z_h, \boldsymbol{\theta}_j) \cdot f(z_h|\boldsymbol{w}_j^z)$, and plug it into the likelihood term $f(S_h|x_h, z_h, \boldsymbol{\theta}_j)$ to prepare for the posterior inference of $\boldsymbol{\theta}_j$.

Step b) Cluster-wise posterior update:

(i) Once the appropriate likelihood term $f(S_h|x_h, z_h, \boldsymbol{\theta}_j)$ is ready, we marginalize the joint $f(S_h|x_h, z_h, \boldsymbol{\theta}_j) \cdot f(z_h|\boldsymbol{w}_j^z)$ over $\mathbf{z}$ to produce the outcome term $f(S_h|\mathbf{X}_h, \boldsymbol{\theta}_j)$ by reducing the degrees of variance introduced from the imputation. As for the covariate term $f(\mathbf{X}_h|\boldsymbol{w}_j)$, we simply omit the covariate term $f(z_h|\boldsymbol{w}_j^z)$ due to missingness (see Roy et al. 2018 and the references therein).

$$f(S_h|\mathbf{X}_h, \boldsymbol{\theta}_j) = \int f(S_h|x_h, z_h, \boldsymbol{\theta}_j) \cdot f(z_h|\boldsymbol{w}_j^z) d\mathbf{z} = f(S_h|x_h, \boldsymbol{\theta}_j) \quad (3.26a)$$

$$f(\mathbf{X}_h|\boldsymbol{w}_j) = f(x_h|\boldsymbol{w}_j^x) \cdot \cancel{f(z_h|\boldsymbol{w}_j^z)} \quad (3.26b)$$

(ii) With the refined outcome and covariate term in Equation (3.26), the ultimate joint $f(S_h, \mathbf{X}_h|\boldsymbol{\theta}_j, \boldsymbol{w}_j) = f(S_h|x_h, \boldsymbol{\theta}_j) \cdot f(x_h|\boldsymbol{w}_j^x)$ can be formulated to classify the cluster membership j for each observation h. The probabilities that $S_h|\mathbf{X}_h$ belongs to each cluster j are computed based on the joint model, and the cluster membership coming with the highest probability is selected for the observation h. When all observations have been assigned to particular clusters j, the posterior parameters $\boldsymbol{\theta}_j$, $\boldsymbol{w}_j$ are

updated, given the cluster membership j at each iteration in the Gibbs sampling.

3.3.2 Handling NDB covariates with RQ2.2

Consider $\mathbf{X} = \{\mathbf{x},\ \mathbf{z}\}$ where $\mathbf{x}$ is subject to mismeasurement. The measurement error and the error-prone covariate (observed) are denoted by $\boldsymbol{\epsilon}$ and $\mathbf{x}^*$ respectively. Assuming $\boldsymbol{\epsilon} \sim \mathbf{N}(0,\ \sigma_\epsilon^2)$ and $\mathbf{x}^* \sim \mathbf{N}(\mathbf{x},\ \tau^2)$, the approach to the mismeasured covariate problem varies depending on the type of the measurement error $\boldsymbol{\epsilon}$. We first outline the types of measurement error in the following.

- **Additive vs Multiplicative** (Fewell 2007)

  With *additive* error, the mismeasured covariate $\mathbf{x}^*$ takes the form: $\mathbf{x}^* = \mathbf{x} + \boldsymbol{\epsilon}$ while the *multiplicative* error presents: $\mathbf{x}^* = \mathbf{x} \cdot \boldsymbol{\epsilon}$. In practice, although this choice of error type is guided by the nature of the data-generating or transformation process, additive error has a simplicity in terms of model development as well as interpretation.

- **Differential vs Non-Differential** (Romann 2008)

  Depending on the relationship with other variables, the error $\boldsymbol{\epsilon}$ can be categorized as *differential* or *non-differential*. A differential error is better suited when the mismeasured covariate $\mathbf{x}^*$ is correlated with the outcome $\mathbf{x}^* \sim S|\ \mathbf{z}, \mathbf{x}$. In contrast, when the mismeasured covariate $\mathbf{x}^*$ does not entail any further information about the outcome except what is already available in $\mathbf{x}$ (the randomness of the mismeasurement is only related to the values of the covariate $\mathbf{x}$ itself), the error $\boldsymbol{\epsilon}$ is known to be non-differential, thus $\mathbf{x}^* \perp S|\ \mathbf{x}, \mathbf{z}$. This implies that $f(S|\mathbf{x}^*, \mathbf{x}, \mathbf{z}) = f(S|\mathbf{x}, \mathbf{z})$ and $f(\mathbf{x}^*|S, \mathbf{x}, \mathbf{z}) = f(\mathbf{x}^*|\mathbf{x})$.

- **Classical vs Berkson** (Carroll et al. 2006)

  Classical error arises when the error $\boldsymbol{\epsilon}$ is independent of the true covariates $\boldsymbol{\epsilon} \perp S, \mathbf{x}, \mathbf{z}$, thus $E[\boldsymbol{\epsilon}|\mathbf{x}^*, \mathbf{z}] = 0$ and $\tau^2 = V(\mathbf{x}) + V(\boldsymbol{\epsilon}) > V(\mathbf{x})$. On the contrary, if the error $\boldsymbol{\epsilon}$ is independent from the observed covariates, but associated with

other unobserved/latent factors with multiple levels, *Berkson* error should be considered where $\boldsymbol{\epsilon}_j \perp S, \mathbf{x}^*, \mathbf{z}$, thus $E[\boldsymbol{\epsilon}_j|\mathbf{x}^*, \mathbf{z}] = 0$ and $\tau_j^2 < V(\mathbf{x})$ for $j = 1, \cdots, J$. This implies that, featuring heteroscedasticity, the error $\boldsymbol{\epsilon}$ can exhibit different variabilities across different risk clusters brought by the unobserved factors.

This thesis focuses on the case in which the mismeasured covariate $\mathbf{x}^*$ arises from additive measurement error in a manner that is non-differential (blind to the outcome and other covariates) and Berkson (correlated to the latent factors). We refer to it as the 'NDB covariate.' The NDB error is particularly relevant to this study, as it describes the impact of covariate errors in the context of clustered data when predicting aggregated claims (risk premiums). In this setting, measurement errors are not uniform but instead vary across clusters. Moreover, the distinct nature of NDB error necessitates developing tailored modeling approaches to properly account for its influence.

With the assumption of the NDB measurement error, we spotlight the Bayesian framework to account for its cluster-wise heteroscedasticity probabilistically. This is because we are dealing with the error structure τ_j^2 may vary across risk clusters, and the Bayesian framework allows for specifying each structural component to account for these varying variances. In short, from a Bayesian perspective, deviations from true values can be corrected by incorporating prior knowledge that captures the relation between the unobservable true covariate $\mathbf{x}_j$ and the observed NDB covariate $\mathbf{x}_j^*$ for each cluster. In this context, the specification of the model components plays a crucial role in formulating this strategy. Similarly to the case of the data augmentation, this cluster-wise inference can also be achieved by exploiting the manageable joint product and Gibbs sampling. The strength of such a Bayesian strategy to cope with the NDB covariate is well described by Stamey and Seaman 2021; Sinha 2021; Grace et al. 2021.

To provide a more precise description of the Bayesian solution to the NDB covariate problem, we begin by specifying the full joint density of the relevant variables

as

$$f(S, \mathbf{x}^*, \mathbf{x}, \mathbf{z}) = f(S|\mathbf{x}^*, \mathbf{x}, \mathbf{z}) \cdot f(\mathbf{x}^*|\mathbf{x}, \mathbf{z}) \cdot f(\mathbf{x}|\mathbf{z}) \cdot f(\mathbf{z}) \tag{3.27a}$$

$$f(S, \mathbf{x}^*, \mathbf{x}|\mathbf{z}) = f(S|\mathbf{x}^*, \mathbf{x}, \mathbf{z}) \cdot f(\mathbf{x}^*|\mathbf{x}, \mathbf{z}) \cdot f(\mathbf{x}|\mathbf{z}) \tag{3.27b}$$

in which the term for the precisely measured covariate $\mathbf{z}$ is factored out for the sake of simplicity. Owing to the assumption of the non-differential error, as elaborated previously, the conditional joint density in Equation (3.27b) can be further reduced to

$$f(S, \mathbf{x}^*, \mathbf{x}|\mathbf{z}) = f(S|\mathbf{x}^*, \mathbf{x}, \mathbf{z}) \cdot f(\mathbf{x}^*|\mathbf{x}, \mathbf{z}) \cdot f(\mathbf{x}|\mathbf{z}) \tag{3.28}$$

This conditional joint density is referred to as the *complete joint* model (Gustafson 2008). Since the true covariate $\mathbf{x}$ is not observable, one can say that the complete joint model is inaccessible or theoretical. However, throughout the construction of the complete joint model in Equation (3.28), three key components in the relationship between the true covariate $\mathbf{x}$ and the observed covariate $\mathbf{x}^*$ can be derived:

- outcome component $f(S|\mathbf{x}, \mathbf{z})$
- linking (measurement) component $f(\mathbf{x}^*|\mathbf{x})$
- covariate (exposure) component $f(\mathbf{x}|\mathbf{z})$

In particular, the linking component plays a role in incorporating the measurement error mechanism of $\mathbf{x}_j^* \sim \mathbf{N}(\mathbf{x}_j,\ \tau_j^2)$ into the analysis, providing control over the simulation process (Grace et al. 2021). The effect of the error in relation to the unknown risk clusters under the Berkson error assumption can also be investigated with the linking (measurement) component term and τ_j^2.

On the other hand, the conditional joint model available in reality can be called the *incomplete joint* model (Gustafson 2008) that is expressed as

$$f(S, \mathbf{x}^*|\mathbf{z}) = f(S|\mathbf{x}^*, \mathbf{z}) \cdot f(\mathbf{x}^*|\mathbf{z}) \tag{3.29}$$

where the outcome term $f(S|\mathbf{x}^*, \mathbf{z})$ and the exposure term $f(\mathbf{x}^*|\mathbf{z})$ are fully known. Note that the incomplete joint model is the marginal of the complete joint model over the unobservable true covariate $\mathbf{x}$.

Complete Case (unknown)	**Incomplete Case (known)**
$\underbrace{f(S\|\mathbf{x},\mathbf{z})}_{\text{outcome}} \cdot \underbrace{f(\mathbf{x}^*\|\mathbf{x})}_{\text{measurement}} \cdot \underbrace{f(\mathbf{x}\|\mathbf{z})}_{\text{exposure}}$	$\underbrace{f(S\|\mathbf{x}^*,\mathbf{z})}_{\text{outcome}} \cdot \underbrace{f(\mathbf{x}^*\|\mathbf{z})}_{\text{exposure}}$
$= f(S, \mathbf{x}^*, \mathbf{x} \mid \mathbf{z})$	$= f(S, \mathbf{x}^* \mid \mathbf{z})$

If we set these two models above equal to each other by marginalizing the complete case model over the unobservable true covariate $\mathbf{x}$, the following equation emerges, which creates the relationship between the parameters of the complete case model and those of the incomplete case model.

$$\int f(S, \mathbf{x}^*, \mathbf{x} \mid \mathbf{z})\, d\mathbf{x} \;=\; f(S, \mathbf{x}^* \mid \mathbf{z}) \tag{3.30}$$

While solving the integral in Equation (3.30) explicitly may be challenging, comparing the parameterizations on the left-hand and right-hand sides of the equation reveals the relationship between the parameters of the model built on the mismeasured covariate $\mathbf{x}^*$ and those of the true model built on the true covariate $\mathbf{x}$. This relationship appears in the form of a system of equations, eliminating the need to sample the unobservable true covariate $\mathbf{x}$. One of the novel features of this thesis is the first-time derivation of the integral's solution, motivated by Romann 2008 and Grace et al. 2021. Based on this analytically derived system of equations, a unique hybrid Gibbs sampler is also developed here for the first time (this will be further detailed in Chapter 4 and Chapter 6) to correct the bias in the parameter estimates from the misspecified model built on the NDB covariates.

Thus far, we have provided an overview of the key methodologies and tools employed to address the five research questions related to covariate-based model risks. To assist the reader's understanding and improve the flow of the thesis, Table 3.1 has been strategically placed at key points. This table aims to illustrate how

each methodology and tool is interwoven with the broader framework of the thesis, providing clarity on their application in later chapters. As we progress through subsequent chapters, we hope that Table 3.1 ensures a cohesive narrative, guiding the reader through the complexities of the research.

Research Question	Technique	Originality	Extension
RQ1.1 Heterogeneity in AVG claims $\bar{Y}$ for a policy	Partial pooling clustering in CH4 (Bayesian-param)	Established by Gelman and Hill 2007	+ Linked to **RQ2.2 NDB** in CH4
RQ1.1 Heterogeneity in a sum of claims S for a policy	Parameter-free clustering in CH5,6 (Bayesian-nonparam)	Established by Neal 2000, etc.	+ Linked to **RQ2.1 MAR** in CH5 + Linked to **RQ2.2 NDB** in CH6
RQ1.2 Convolution for a sum of claims S	Log-normal SUM in CH5,6	**"Derived by us"** (conditional on X) using Li 2008, etc	+ Linked to **RQ2.1 MAR** in CH5 + Linked to **RQ2.2 NDB** in CH6
RQ1.2 Convolution for a total claim $\tilde{S}$	Log-skewnormal SUM in CH6	**"Derived by us"** (conditional on X) using Chatterjee 2006	+ Linked to **RQ2.2 NDB** in CH6
RQ1.3 Scalability	Parallel Monte Carlo simulation in CH6	Established by Ni et al. 2020, etc.	+ Linked to **RQ2.2 NDB** in CH6
RQ2.1 MAR in a binary covariate **z**	Data augmentation technique in CH5 (Bayesian)	Established by Tanner 2010, etc.	+ Linked to CH5 RQ1.1 Heterogeneity RQ1.2 Convolution
RQ2.2 NDB in a cont. covariate **x**	Gustafson correction equations in CH4,6 (Bayesian)	**"Derived by us"** using Gustafson 2008	+ Linked to CH4,6 RQ1.1 Heterogeneity RQ1.2 Convolution RQ1.3 Scalability

Table 3.1: Overview of research question-specific contributions and their connections in this thesis. Aside from these contributions, the novelty of this thesis lies in enhancing the applicability of state-of-the-art techniques, extending their use to a wider range of analytical contexts shaped by the combination of research questions (as shown in the 'Extension' column above).

3.4 Choice of Bayesian Framework: Parametric or Nonparametric?

This thesis considers both Bayesian parametric and nonparametric settings for risk premium modeling. In the parametric setting, the number of risk clusters is known and fixed, and the model fitting and posterior inference are relatively simple because the risk clustering components (distributions) are chosen by a researcher. In addition, it has been popular because the framework is based on well-established statistical principles (Hogg and Klugman 2009). However, one might potentially have to take the risk of making a 'wrong' choice of the clusters, resulting in misleading predicted values. In truth, we never know beforehand about the properties of the true risk clusters or structures associated with the heterogeneity in the aggregate claim data we have. The parametric approach is known to be effective as long as researchers' selection of the distributions for the risk clustering can largely mirror the true characteristics of the data (G. J. McLachlan et al. 2019).

In the nonparametric setting, the number of risk clusters is not allowed to be fixed, instead, the data determine the properties of the risk clusters (Teh and Jordan 2010). The risk of making a 'wrong' choice of the clustering components can be minimized as the framework formulates a single all-encompassing model that can cover any parametric components. This can be done by treating the clustering components themselves as a bulk of parameters to be estimated without any specified distributional form (Hong and R. Martin 2018). Realistically speaking, however, the risk clusters without distributional forms tend to bring about computation problems. As a result of this, it is a rule of thumb to develop the cluster shapes centered around standard parametric forms at the outset (Shahbaba and Neal 2009), and, eventually, let the data derive the cluster shapes. The parameter-free clustering technique discussed in Section 3.2.1 can be a great example of this. It provides the clustering components based on some distributions, but renders their distributional forms completely data-driven by marginalizing out the parameters (to be parameter-

free), relative to the cluster forms.

In what follows, we further navigate the typical Bayesian parametric model and its important features as well as the Bayesian nonparametric model that is built on top of the parametric model.

3.4.1 Parametric: Bayesian hierarchical model

The Bayesian hierarchical model is a product of the extension of the Bayesian parametric approach. The Bayesian parametric approach is characterized by treating each parameter of a model as a random variable and specifying a well-known probability distribution for each parameter (Fellingham et al. 2015). The Bayesian hierarchical model is a combination of multiple parametric models, each of which explains different model layers such as data, parameter, hyperparameter, etc. Assuming a fixed number of risk clusters (i.e. the clusters should be identified beforehand), the hierarchical layers are jointly performed at both the cluster level and the population level, but each layer communicates with data differently (Gelman and Hwang 2014).

We recall that, in risk premium modeling, covariates (e.g., age, location, driving history) affect claim likelihood, while risk clustering groups policyholders with similar risk profiles for fair premiums. With the presence of the unobservable structure across the risk clusters (i.e. unexplained by the known covariates), the Bayesian hierarchical model exploits the partial pooling technique mentioned in Section 3.2.1 by leveraging information across different levels of the hierarchy. See Figure 3.3 for an illustration of how the partial pooling technique is integrated with the Bayesian hierarchical model. The hierarchical structure brings the extremes of dependency (complete pooling) and independency (no pooling) between the clusters all together: The no-pooling overfits the data within each cluster by modeling each cluster separately; The complete pooling underfits the data by modeling the entire data at once while ignoring variation between clusters. This combination is characterized as a bias-variance tradeoff between these two extremes (Briscoe and Feldman 2011).

In the diagram, each cluster is modeled by a Gaussian regression with a mean

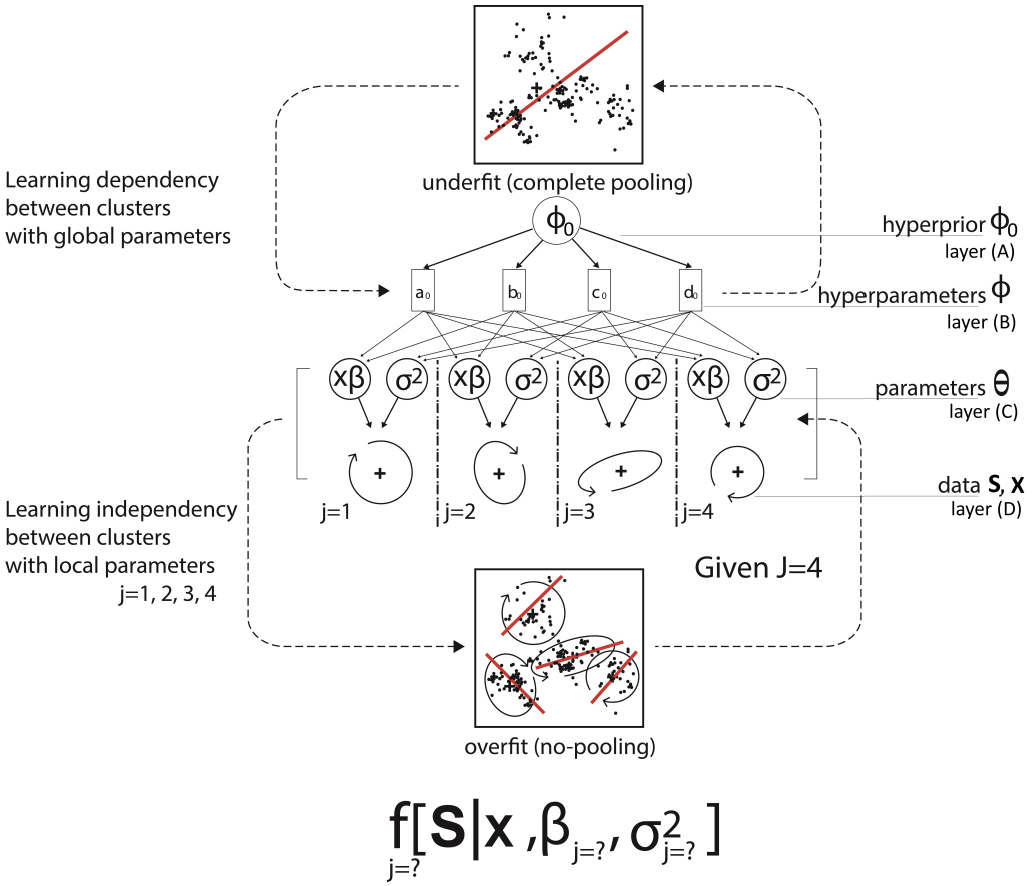

Figure 3.3: As a Bayesian parametric example with $J = 4$ clusters, this diagram depicts a typical Bayesian hierarchical model with the Bias-Variance trade-off through the partial pooling. As for prediction, the class membership j of the data point should be known beforehand.

$\mathbf{X}\boldsymbol{\beta}$ and variance σ^2 for example. In the top layer, the global parameter (hyperparameters) learns dependency between the clusters by using complete pooling. In the bottom layer, the local parameters learn independency between the clusters by ignoring any pooling. Hence, the local structure in the bottom layer updates its parameter cluster-wise, while the update of the global parameter in the top layer is based on the entire data. If the entire data exhibits small variability (i.e., all insured parties are affected by similar risk clusters), the model addresses this by adjusting the local cluster parameters accordingly as explained in Section 3.2.1. When certain risk clusters have too limited data, the model borrows information from other clusters. In this process, the hierarchical model accounts for both individual variation and commonalities within the clusters at the same time and thus resolves RQ1.1

heterogeneity issue (Gelman and Carlin 2013).

To facilitate formal comprehension, consider the following example of ours. Suppose the local cluster j's parameter $\boldsymbol{\theta}_j = \{\boldsymbol{\beta}_j, \sigma_j^2\}$ for $j = 1, \cdots, J$ is an independent sample from a population distribution with a set of global parameters (hyperparameters) $\boldsymbol{\phi} = \{a_0, b_0, c_0, d_0\}$ as described in Figure 3.3. If we assume that the global parameter $\boldsymbol{\phi}$ is unknown, and needs to be estimated, the uncertainty in both $\boldsymbol{\theta}$ and $\boldsymbol{\phi}$ would be assessed, using the joint posterior of $\boldsymbol{\theta}, \boldsymbol{\phi}$ as

$$p(\boldsymbol{\theta}, \boldsymbol{\phi}|S, \mathbf{X}) \propto f(S|\mathbf{X}, \boldsymbol{\theta}, \boldsymbol{\phi}) \cdot p(\boldsymbol{\theta}, \boldsymbol{\phi}) = \prod_{j=1}^{J} \underbrace{f(S_j|\mathbf{X}_j, \boldsymbol{\theta}_j, \boldsymbol{\phi})}_{\text{likelihood}} \cdot \underbrace{p(\boldsymbol{\theta}_j, \boldsymbol{\phi})}_{\text{prior}} \tag{3.31}$$

where the hyperparameters $\boldsymbol{\phi}$ affect the outcome only through the cluster parameter $\boldsymbol{\theta}_j$ as defined in the hierarchy in Figure 3.3. In order to study the joint prior $p(\boldsymbol{\theta}_j, \boldsymbol{\phi})$ in Equation (3.31), we decompose it into the cluster parameter distribution $p(\boldsymbol{\theta}_j|\boldsymbol{\phi})$, and the hyperprior $p(\boldsymbol{\phi})$. The posterior of the cluster parameter distribution and the hyperprior are denoted as $p(\boldsymbol{\theta}_j|\boldsymbol{\phi}, S_j, \mathbf{X}_j)$ and $p(\boldsymbol{\phi}|S, \mathbf{X})$ respectively. Accordingly, one can evaluate them as

$$p(\boldsymbol{\theta}|\boldsymbol{\phi}, S, \mathbf{X}) = \prod_{j=1}^{J} p(\boldsymbol{\theta}_j|\boldsymbol{\phi}, S_j, \mathbf{X}_j) \qquad \rightarrow \text{ layer (C) in Figure 3.3} \tag{3.32}$$

$$p(\boldsymbol{\phi}|S, \mathbf{X}) = \frac{p(\boldsymbol{\theta}, \boldsymbol{\phi}|S, \mathbf{X})}{p(\boldsymbol{\theta}|\boldsymbol{\phi}, S, \mathbf{X})} \qquad \rightarrow \text{ layer (B) in Figure 3.3} \tag{3.33}$$

where the parameter layer for $\boldsymbol{\theta}_j$ is informed by the data points grouped by each cluster j (no-pooling) while the hyperparameter layer for $\boldsymbol{\phi}$ is engaged with the entire population without any clusters (complete pooling). Note that the numerator in Equation (3.33) is just the joint posterior distribution from Equation (3.31), and the denominator in Equation (3.33) is the posterior for $\boldsymbol{\theta}|\boldsymbol{\phi}$ in Equation (3.32). From the above analytic expression, one can see that $\boldsymbol{\theta}_j$ and $\boldsymbol{\phi}$ can be drawn and estimated recursively, which is aligned with Figure 3.3. Finally, the $\boldsymbol{\theta}_j$ and $\boldsymbol{\phi}$ allow for developing the predictive distribution of $S|\mathbf{X}$ as a result of the partial pooling.

3.4.2 Nonparametric: Dirichlet process mixture model

Although the partial pooling with the Bayesian hierarchical model is designed to address RQ1.1 heterogeneity problem by capturing the latent structure across the risk clusters, the incorrect prediction problem stemming from the wrong choice of the clustering components still remains as an inherent risk of the Bayesian parametric approach. The Bayesian nonparametric (BNP) approach reveals superiority by encompassing all possible shapes of the clustering components to resolve this issue (Gershman and Blei 2012). In particular, the Dirichlet process mixture (DPM) model, as a nonparametric extension of the Bayesian hierarchical approach, has been in the research spotlight in insurance applications (see, for example, Hong and R. Martin 2018; Huang and S. Meng 2020; Ungolo and Heuvel 2024, etc.) due to its solid theoretical foundation and computational efficiency. All discussions for the development of the DPM model in this thesis are based on the principles introduced by Ferguson 1973; Antoniak 1974; Sethuraman 1994.

Assuming that there are multiple unknown risk clusters across the individual claim data Y_i, the aggregate claim amount S_h for a policy h would contain many different structures that cannot be explained by fitting a single distribution (Fellingham et al. 2015). In order to approximate the distribution that captures such diverse risk clusters in S_h, the investigation of many different risk clustering scenarios would be desirable. Intuitively speaking, the DP prior caters for such a need in modeling S_h by suggesting many different shapes of the clustering components simultaneously based on simulations of the diverse risk clustering scenarios.

In some scenarios, the DPM model might accrete brand-new clusters and add new unknown risk clusters to our analysis. In others, the DPM model might discard the existing clusters and see if what results better covers the actual settings. It is the parameter-free clustering technique that is at the heart of this investigation (Teh 2010). As discussed in Section 3.2.1, the parameter-free clustering algorithm simulates an infinite number of new clusters and their parameters while examining every corner of the parameter space by capturing the correlation between the clusters and

individual data point (see details presented in Chapter 5). Throughout the simulations, many different clustering scenarios can be developed, shaping a new form of the risk clustering component. As a result, the RQ1.1 Heterogeneity issue can be addressed in a highly efficient way.

To be specific, the simple form of the DPM model is given by

$$\begin{aligned} S_h|\mathbf{X}_j &\sim f(\ \cdot\ |\boldsymbol{\phi}_j) \\ \boldsymbol{\phi}_j :\ \{\boldsymbol{\theta}_j,\ \boldsymbol{w}_j\} &\sim G \\ G := \Sigma_{j=1}^{\infty}\boldsymbol{\omega}_j(\mathbf{X})\delta_{\phi_j}(\cdot) &\sim \mathbf{DP}\big(\alpha,\ G_0\big) \end{aligned} \tag{3.34}$$

where the $\boldsymbol{\phi}_j$ is a pair of the parameter vectors $\{\boldsymbol{\theta}_j, \boldsymbol{w}_j\}$ for the outcome and covariates that define the cluster j together. Given that the joint of the outcome and covariate model composes each cluster j filled with data points $S_h|\mathbf{X}_j \sim f(\ \cdot\ |\boldsymbol{\phi}_j)$, the cluster parameter sample space G is populated by an infinite number of instances of $\boldsymbol{\phi}_j$ because G itself is considered as a continuous distribution with support on the cluster parameter $\boldsymbol{\phi}_j$. It is the *DP prior* that generates G in Equation (3.34).

The DP prior in Equation (3.34) is defined in terms of two parameters, a positive scalar (precision) parameter α, which determines the size of the cluster parameter samples selected from the sample space G (thus larger α allows for the introduction of new clusters more often. See Escobar's posterior development in Equation (3.9) in Section 3.2.1), and a base measure G_0, which supplies a set of new, provisional parameter values $\boldsymbol{\phi}$ to G to perform the parameter-free clustering.

Note that the brand-new clusters (with new $\boldsymbol{\phi}_j$) selected from G are the result of the parameter-free clustering algorithm, hence the choice can vary at each iteration, and so does the makeup of G because G_0 sets up a new parameter space G at each iteration. All the samples selected from G are assigned relevant mixing weights $\boldsymbol{\omega}_j$, and G ensures that the sum of these mixing weights is always equal to 1 in order to define the valid probability model as a mixture of clusters (distributions). We call this a *clustering scenario*. Although the resultant clustering scenario at each iteration is determined by the parameter-free clustering algorithm, it is G_0

that provides all necessary parameters to perform this selection, establishing the countable (discrete) parameter space in G.

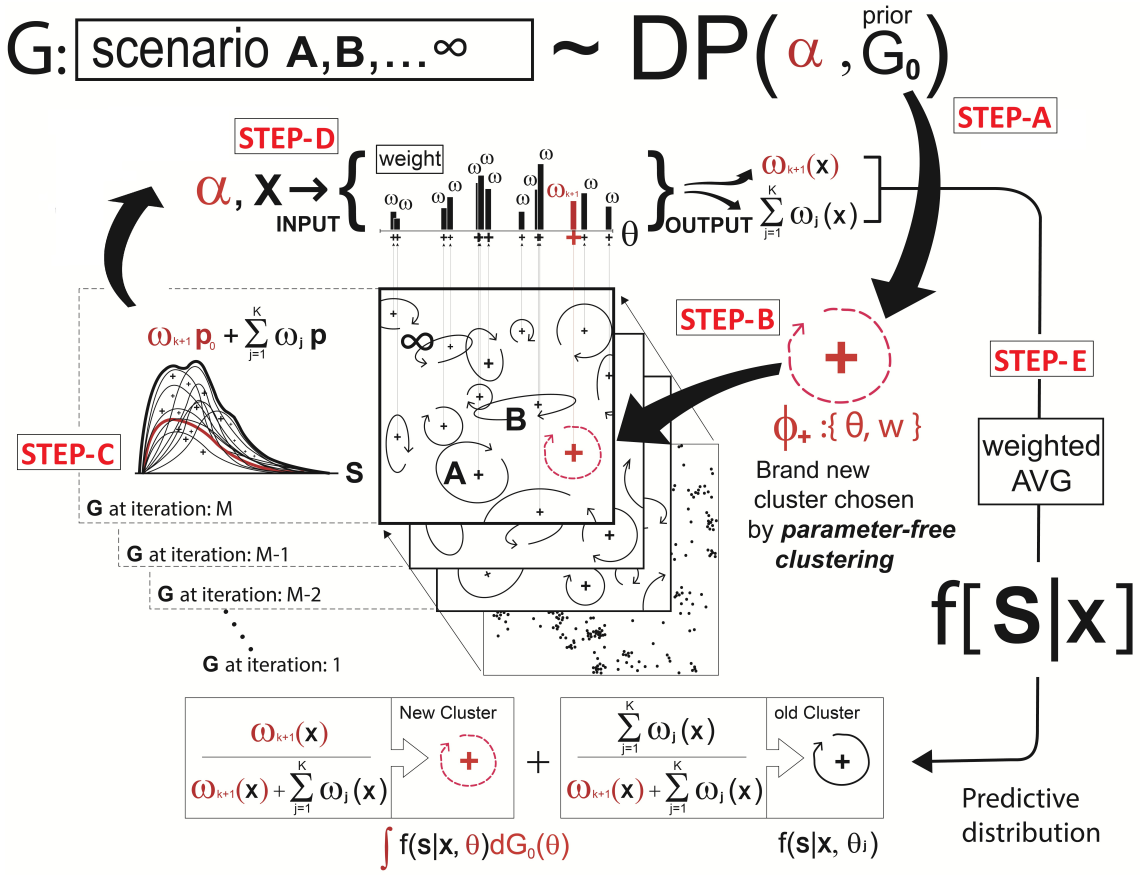

Figure 3.4: As a Bayesian nonparametric example with $J = \infty$, this diagram describes a fluid process of the materialization of the clustering scenarios (as a result of the parameter-free clustering algorithm at each iteration) and the development of a predictive distribution based on the finalized clustering scenarios.

Figure 3.4 briefly displays how the resultant clustering scenarios selected from G multiply throughout the course of the iterations in the parameter-free clustering process, and how the weighted average of these scenarios ultimately delivers the posterior predictive distribution. This scenario generation process described in Figure 3.4 can be broken down as follows:

STEP-A: The process starts with the DP prior that has a base measure G_0 as a master provider of the cluster parameters $\boldsymbol{\phi}_+ : \{\boldsymbol{\theta}, \boldsymbol{w}\}$ (because it is a joint of all parameters used throughout this process, allowing for the sampling of any required parameters) and the precision α as a sampling size controller.

STEP-B: First, G_0 samples an infinite number of the cluster parameter points $\boldsymbol{\phi}_1, \cdots, \boldsymbol{\phi}_\infty$, and these samples constitute G. This is a universe of the cluster parameters.

STEP-C: Once G is ready, the parameter-free clustering algorithm produces M distinct clustering scenarios for iterations $1, \cdots, M$ by selecting brand-new cluster parameters $\boldsymbol{\phi}_+ : \{\boldsymbol{\theta}, \boldsymbol{w}\}$ from the universe of G in each iteration, which is described in Section 3.2.1.

STEP-D: In each clustering scenario, it is essential for the mixture of probability densities to remain valid. Hence, G assigns the mixing weights $\boldsymbol{\omega}_{j=1}(\mathbf{X}), \cdots, \boldsymbol{\omega}_{j=J}(\mathbf{X})$ to the selected cluster parameters $\boldsymbol{\phi}_{j=1} : \{\boldsymbol{\theta}_1, \boldsymbol{w}_1\}, \cdots, \boldsymbol{\phi}_{j=J} : \{\boldsymbol{\theta}_J, \boldsymbol{w}_J\}$ populated in each clustering scenario, with these weights being determined by the covariates associated with those parameter selections.

STEP-E: In the final stage, the predictive distribution $f(S_h|\mathbf{X})$ is developed by averaging out all the different materialized clustering scenarios selected from G.

To explore the process of obtaining the brand-new clusters in detail, Figure 3.5 takes a snapshot of the role of G_0 and the birth of a brand-new cluster at a certain iteration in the parameter-free clustering process. The brand-new cluster construction process described in Figure 3.5 can be broken down in what follows: For clarity, let $\boldsymbol{\phi}$ denote the set of candidate cluster parameters (yet to be selected), and $\boldsymbol{\phi}_+$ represent the cluster parameters chosen by the parameter-free clustering algorithm.

STEP.01: At the outset, given that the birth of a brand-new cluster requires the cluster parameter $\boldsymbol{\phi}_+$ (information on its location, shape, etc.) and the mixing weight $\boldsymbol{\omega}(\mathbf{X})$, a base measure G_0 creates a universe of the cluster parameter points $\boldsymbol{\phi}_1, \cdots, \boldsymbol{\phi}_\infty$ collectively known as G. Their mixing weights $\boldsymbol{\omega}_{j=1}(\mathbf{X}), \cdots, \boldsymbol{\omega}_{j=J}(\mathbf{X})$, $\boldsymbol{\omega}_{j=J+1}(\mathbf{X})$ are ready to be computed, with the precision α being set by the researcher.

STEP.02: Next, the parameter-free clustering algorithm described in Section 3.2.1 selects a set of clustering parameters $\boldsymbol{\phi}_{+j}$ for $j = 1, \cdots J + 1$ registered in G.

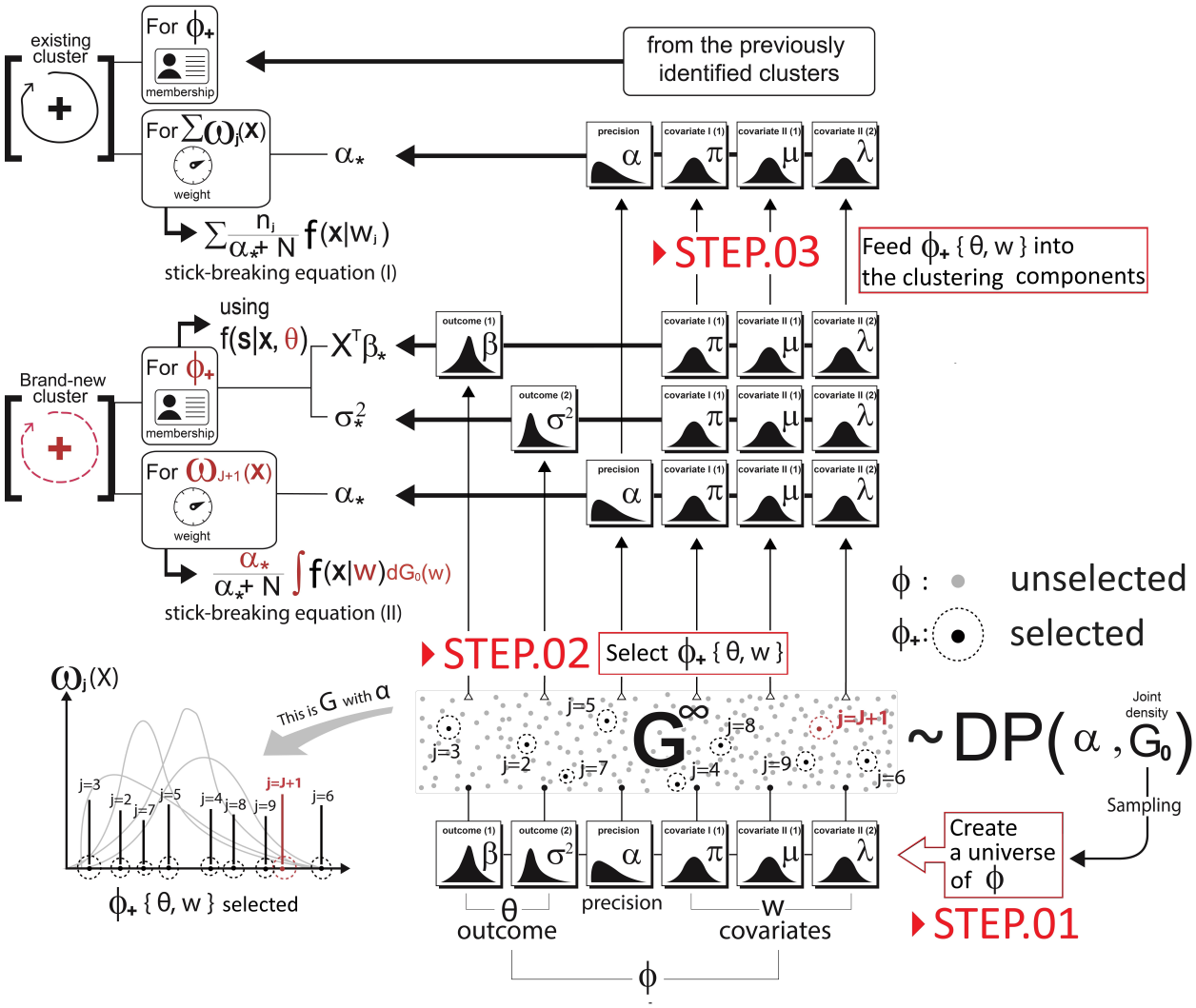

Figure 3.5: This is an anatomy of the birth of brand-new clusters. A joint density G_0 generates all necessary parameters $\boldsymbol{\phi}_+$:$\{\boldsymbol{\beta}, \sigma^2, \pi, \mu, \lambda\}$ to deliver the cluster information - shape, location, etc. - that is required to create brand-new clusters. Only selected parameters $j = 1, \cdots J, J+1$ from G are fed into the clustering components.

Subsequently, their mixing weights $\boldsymbol{\omega}_{j=1}(\mathbf{X}), \cdots, \boldsymbol{\omega}_{j=J}(\mathbf{X})$, $\boldsymbol{\omega}_{j=J+1}(\mathbf{X})$ are immediately computed by the generalized stick-breaking equation (Sethuraman 1994).

STEP.03: Lastly, the selected cluster parameter values are fed into the specified clustering components - the outcome $f(S_h|\mathbf{X}_j, \boldsymbol{\phi}_{+j})$, covariate model $f(\mathbf{X}_j|\boldsymbol{\phi}_{+j})$, and covariate-dependent mixing weight $\boldsymbol{\omega}(\mathbf{X}_j|\boldsymbol{\phi}_{+j})$ -, giving rise to the brand-new clusters along with a new clustering scenario.

Important theory and properties: Regarding G, it is just a provisional parameter space developed by the DP prior. In general, G is known as a continuous probability density with support on a collection of an infinite number of the cluster parameter points $\boldsymbol{\phi}_1, \cdots, \boldsymbol{\phi}_\infty$ that come with a particular mixing weight $\boldsymbol{\omega}_j(\mathbf{X})$ (Gershman and Blei 2012). Therefore, it can also be considered as an infinite mixture density as shown in Equation (3.34). Technically, if $\boldsymbol{\phi}_i = G(A_i)$, then G is a collection of $\{G(A_1), G(A_2) \ldots\}$ sampled from the DP prior, taking independent partitions $A_1, A_2, \ldots$ of the sample space $\bigcup_{i=1}^{\infty} A_i = A$ with support on G_0.

The $\delta_{\phi_j}(\cdot)$ in Equation (3.34) is the *Dirac measure* as an indicator function, aiming to eliminate the unselected parameter points from the emergent clustering scenario of the clusters $j = 1, \cdots, J$ by examining the partition that the parameter point takes on (Escobedo 1986). Each clustering scenario manifestation from G is a fluid process as the parameter-free clustering algorithm continually adapts and reforms the desired partition of the sample space to fit the needs. The parameter space G has an infinite number of possible samples while the total number of clusters J in the materialized scenario is finite, and subject to change at every iteration of the parameter-free clustering process. Therefore, at each iteration, the Dirac measure allows for adjusting all the mixing weight $\boldsymbol{w}_j(\mathbf{X})$ for $j = 1, \cdots, J$ to keep the sum of them equal to 1 by assigning the probability mass 0 to the rest of unselected parameter points $\boldsymbol{\phi}_{i \neq j}$ in G.

Having said that, we cannot, in reality, bring into being the entire parameter points $\boldsymbol{\phi}_i$ for $i = 1, \cdots, \infty$ in G at once. However, if G_0 can be specified, one can evaluate the joint product of n repeated random draws $\boldsymbol{\phi}_{1:n}$ by marginalizing it over G as

$$\begin{aligned} p(\boldsymbol{\phi}_1, \cdots, \boldsymbol{\phi}_n \mid G_0) &= \int \prod_{i=1}^{n} p(\boldsymbol{\phi}_i|G) \cdot p(G|G_0) \cdot p(G_0)\, dG \\ &= p(\boldsymbol{\phi}_1) \cdot p(\boldsymbol{\phi}_2|\boldsymbol{\phi}_1) \cdot p(\boldsymbol{\phi}_3|\boldsymbol{\phi}_{1:2}) \cdots p(\boldsymbol{\phi}_n|\boldsymbol{\phi}_{1:n-1}) \end{aligned} \tag{3.35}$$

which can be computed recursively, using the conditional distribution of each sample

$\boldsymbol{\phi}_n|\boldsymbol{\phi}_{1:n-1}$ specified as (Blackwell and MacQueen 1973)

$$p(\boldsymbol{\phi}_n|\boldsymbol{\phi}_{1:n-1}) = \frac{1}{\alpha+n-1}\Sigma_{i=1}^{n-1}\delta_{\phi_i}(\cdot) + \frac{\alpha}{\alpha+n-1}G_0 \tag{3.36}$$

Using Equation (3.36), one can derive each probability term in Equation (3.35) $p(\boldsymbol{\phi}_1) = G_0$, $p(\boldsymbol{\phi}_2|\boldsymbol{\phi}_1) = \frac{1}{\alpha+1}\delta_{\phi_1}(A_2) + \frac{\alpha}{\alpha+1}G_0$, $p(\boldsymbol{\phi}_3|\boldsymbol{\phi}_{1:2}) = \frac{1}{\alpha+2}\big[\delta_{\phi_1}(A_3) + \delta_{\phi_2}(A_3)\big] + \frac{\alpha}{\alpha+2}G_0, \cdots$, and so on. As mentioned previously, each parameter sample $\boldsymbol{\phi}_i = G(A_i)$ takes on a certain partition of the sample space, exhibiting a unique property, and the Dirac measure $\delta_{\phi_i}(\cdot)$ returns 1 when the sample $\boldsymbol{\phi}_i$ takes on the target partition of the sample space, and 0 otherwise. This delineates that each probability term derived from Equation (3.36) with the Dirac measure deals with the presence of the shared partitions between the different parameter samples by examining every possible combination of them.

From this, the important property of the DP prior comes to light: "Any cluster assignments based on the conditional density described in Equation (3.36) are *exchangeable*." (Ferguson 1973). This is because the joint probability value $p(\boldsymbol{\phi}_1, \cdots, \boldsymbol{\phi}_n \mid G_0)$ in Equation (3.35) does not change regardless of the order of the data points shuffled, if each term $p(\boldsymbol{\phi}_n|\boldsymbol{\phi}_{1:n-1})$ comes from Equation (3.36). In other words, the probability of a particular configuration of clusters does not depend on the order of the data points shuffled. In addition, the different data points can eventually group together according to their parameters that share similar partitions of the sample space. Here we employ the concept of a Chinese Restaurant Process (CRP)[5] in the DPM model because the partition of the sample space is CRP distributed with the precision α under the assumption of exchangeability (for further review, see Chapter 5). For a complete exposition of this theory and mathematical details, see also Teh and Jordan 2010; Blei and Frazier 2011; Gershman and Blei 2012.

[5]This is a probabilistic concept describing how clusters are formed: customers (data points) enter a restaurant with an infinite number of tables (clusters) and either sit at an occupied table with a probability proportional to the number of people already seated there or start a new table with a fixed probability, allowing for a flexible number of clusters (Blei and Frazier 2011).

3.5 Model Evaluation and Modeling Cycle

Throughout Sections 3.1 to 3.4, we have presented a range of background theories and tools, some adopted from literature and others from our own derivation based on literature, to address our research questions - RQ1.1 Heterogeneity, RQ1.2 Convolution error, RQ1.3 Scalability, RQ2.1 MAR covariate, RQ2.2 NDB covariate - regarding different model risk issues.

However, "when it rains, it pours"; The different types of model risk issues in the risk premium modeling often cluster together due to the intricate nature of the claim and covariate data and the diversity of the risk factors (Asmussen and Steffensen 2020). In the upcoming chapters - Chapter 4, Chapter 5, and Chapter 6 -, we focus on integrating these tools presented throughout Sections 3.1 to 3.4 within the Bayesian framework to tackle the different combinations of model risks that commonly arise together in practice. The major novelty of this thesis lies in establishing connections between the Bayesian framework and a series of different model risk correction strategies.

While each strategy presented in Chapter 4, Chapter 5, and Chapter 6 exhibits distinct formulations and approaches, the unified goal is to identify the optimal parameters of the log-skewnormal outcome distribution and obtain the most accurate risk premium predictions. To this end, a range of models, including both Bayesian and non-Bayesian approaches reviewed in Chapter 2, will be employed and compared together to evaluate the model performance.

3.5.1 Model Validation

To validate the performance of the proposed models in the subsequent chapters, the following methods will be considered:

- **LPPD and Log-likelihood :** As for assessing the predictive performance within the Bayesian framework, we can use a metric based on the 'log-likelihood' such as *Log Pointwise Predictive Density* (LPPD) (McElreath 2018). LPPD

integrates predictions across the entire posterior, reducing overfitting while directly quantifying predictive performance. This aligns with Bayesian emphasis on uncertainty propagation (Mara et al. 2016). The computation of LPPD involves the full posterior distribution to capture the model's uncertainty about its parameter estimations. With posterior samples $\boldsymbol{\theta}_1, \cdots \boldsymbol{\theta}_M$, LPPD is calculated by averaging the log-predictive likelihood for each data point over the posterior samples, and summing them up as follows:

$$\mathbf{LPPD}(S_1, \cdots, S_H, \mathbf{X}_1, \cdots, \mathbf{X}_H, \boldsymbol{\theta}) = \sum_{h=1}^{H} \log\Big(\frac{1}{M}\sum_{m=1}^{M} \mathbf{L}(\boldsymbol{\theta}_m; S_h, \mathbf{X}_h)\Big) \tag{3.37}$$

As the likelihood function takes values from 0 to 1 (since it is a probability function), the LPPD takes the values from $-\infty$ to 0. If multiplying the LPPD by -2, the result behaves akin to Mean Squared Error (MSE) such that a perfect fit has a value of zero, and a poor fit has a huge positive value just like MSE. In the context of non-linear modeling, $-2\cdot$LPPD (roughly MSE) values can be interpreted as *Scaled Deviance* (SD) (Cousineau and Allan 2015). If having only one regression model, we compute the two different SDs. One is for comparing the model with its 'Saturated' version[6], and the other is for comparing the model with its 'Null' version[7]. These two SDs are briefly illustrated in Figure 3.6. The model fit is evaluated by comparing these two SDs based on Pearson's χ^2 test, which relies on the assumption that the difference of two SDs follows a χ^2 distribution with P degrees of freedom where P is the number of parameters excluding the intercept (Wood 2002). The disadvantages of the deviance method arise when the outcome distribution is too complex due to multi-modality, extreme skewness, etc., and thus the log-likelihood becomes computationally challenging to access directly (Cousineau and Allan 2015). In

[6]In the Saturated model, each observation has its own parameter, thus H parameters need to be estimated. It implies the extreme case of overfitting (Cousineau and Allan 2015).

[7]In the Null model, only one parameter (intercept) can summarize the behaviour of all of the observations, thus this only needs to be estimated. It implies the extreme case of underfitting (Cousineau and Allan 2015).

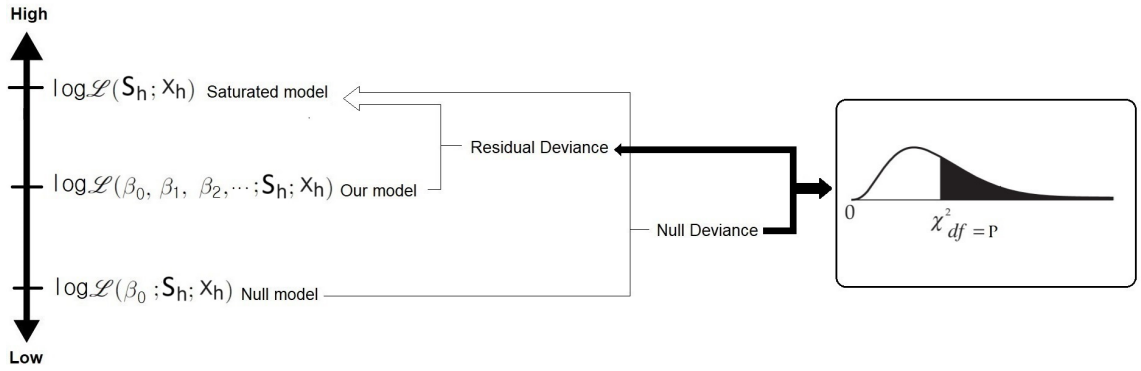

Figure 3.6: A generic diagram to explain two types of scaled deviance for a single non-linear regression. Model fit is measured based on χ^2 Goodness of Fit test.

such cases, approximation methods such as MCMC or variational inference, etc. can be employed to estimate the log-likelihood. Note that while SD focuses on model fit, LPPD emphasizes predictive performance. This is because LPPD not only assesses how well the model fits the training data but also evaluates its generalization to new, unseen data by incorporating parameter uncertainty from the full posterior distribution (McElreath 2018).

- $\boldsymbol{D_{KL}}$ **:** The Kullback-Leibler Divergence D_{KL} can be substituted for the SD method when we need to avoid distributional constraints. The D_{KL} measures how one probability distribution diverges from the other expected probability distribution by comparing their entropies $H[.]$ (Anderson and Burnham 2004). If we have a pair of competing models, then the one that minimizes the D_{KL} is considered a better fit. Suppose our predictive model is $\mathbf{L}(\boldsymbol{\theta}; S, \mathbf{X})$ and the target model is $\mathbf{P}(\boldsymbol{\theta}^{true}; S, \mathbf{X})$. The Kullback-Leibler Divergence $D_{KL}(\mathbf{P}, \mathbf{L}) = H[\mathbf{P}, \mathbf{L}] - H[\mathbf{P}]$ for our model can be computed as:

$$-\sum_{h=1}^{H} \log\Big(\mathbf{L}(\boldsymbol{\theta}; S_h, \mathbf{X}_h)\Big) \cdot \mathbf{P}(\boldsymbol{\theta}^{true}; S_h, \mathbf{X}_h) + \sum_{h=1}^{H} \log\Big(\mathbf{P}(\boldsymbol{\theta}^{true}; S_h, X_h)\Big) \cdot \mathbf{P}(\boldsymbol{\theta}^{true}; S_h, \mathbf{X}_h) \tag{3.38}$$

where $\sum_{h=1}^{H} \log\Big(\mathbf{L}(\boldsymbol{\theta}; S_h, \mathbf{X}_h)\Big)$ is the LPPD discussed previously. While we cannot directly access the true model, or the form of the target model $\mathbf{P}(\boldsymbol{\theta}^{true}; S_h, \mathbf{X}_h)$ is unknown, this does not affect our focus on the 'divergence' between differ-

ent models. This is because the term $\mathbf{P}(\boldsymbol{\theta}^{true}; S_h, \mathbf{X}_h)$ remains constant across comparisons. If our model has a better fit, D_{KL} will shrink, yielding a smaller value. Our interest lies in determining which candidate model results in a greater reduction of D_{KL} and by what margin. This divergence information can be easily obtained by computing the difference between the LPPDs of the competing models.

- **SSPE / SAPE :** The prediction performance can also be measured by quantifying the difference between the predicted values and the given observed values using the Sum of Square Error (SSE) criterion. Specifically, the Sum of Square Prediction Error (SSPE) and the Sum of Absolute Prediction Error (SAPE) metrics can be employed to capture different aspects of prediction accuracy (Parodi 2023). The SSPE focuses on the squared differences between the predicted value $g(\mathbf{X}_h)$ and the actual value S_h while the SAPE is calculated by taking the absolute differences between the predicted value $g(\mathbf{X}_h)$ and actual values S_h across all observations $h = 1, \cdots H$ as follows:

$$\text{SSPE: } \sum_{h=1}^{H} \big(g(\mathbf{X}_h) - S_h\big)^2 \tag{3.39a}$$

$$\text{SAPE: } \sum_{h=1}^{H} |g(\mathbf{X}_h) - S_h| \tag{3.39b}$$

  SSPE and SAPE serve different purposes in assessing prediction performance. While SSPE heavily penalizes large deviations between predicted and actual values, SAPE treats all deviations equally by focusing on absolute differences. Considering our heavily skewed outcome S_h in our study, where outliers on the long tail may occur frequently, SAPE might be preferred over SSPE. This is because each data point carries equal importance in our study, and penalizing larger error values, as done in SSPE, may not be necessary, especially when we are interested in the potential outliers.

- **CTE :** The last focus of this validation process in this thesis is the evaluation of

risk within the predictive distributions of the models, with particular attention to the Conditional Tail Expectation (CTE). CTE is defined as

$$\text{CTE}(q) = E[\ S_h | S_h > Q_q(S_h)\ ], \quad q \in (0,1) \tag{3.40}$$

where $Q_q(S_h)$ is the qth quantile of the predictive distribution. The CTE examines the tail behavior of the predictive distributions to gain insights into the expected aggregate losses (risk premium) under extreme conditions (Brazauskas et al. 2008). It gives an idea of how large losses could be in extreme cases, beyond a certain confidence level; therefore, a smaller CTE value suggests that the model is predicting less severe losses in extreme scenarios.

We have briefly discussed a set of model validation criteria that include: 1) LPPD, 2) D_{KL}, 3) SSPE, 4) SAPE, and 5) CTE. For the series of experiments in the subsequent chapters, the performance and accuracy of our proposed models in predicting the risk premium will be measured based on these criteria.

3.5.2 Modeling Cycle

To enhance the efficiency of the modeling process studied in the forthcoming chapters, Figure 3.7 suggests a specific modeling cycle that includes all the essential phases involved. At the onset of the cycle, we define the goal of our risk premium development and design our experiment. This involves determining the types of outcomes and covariates, as well as establishing the appropriate methodologies to address our target issue. Subsequently, we gather relevant insurance claim data that align with our experimental design and overall objectives. This data collection phase is followed by cleaning and transformation processes to ensure its suitability for our purposes.

Once the data is acquired, we initiate the pivotal model risk assessment phase. Here, we first evaluate the basic features of the data to gauge the scalability of our model against the backdrop of increasing data volumes, which concerns RQ1.3.

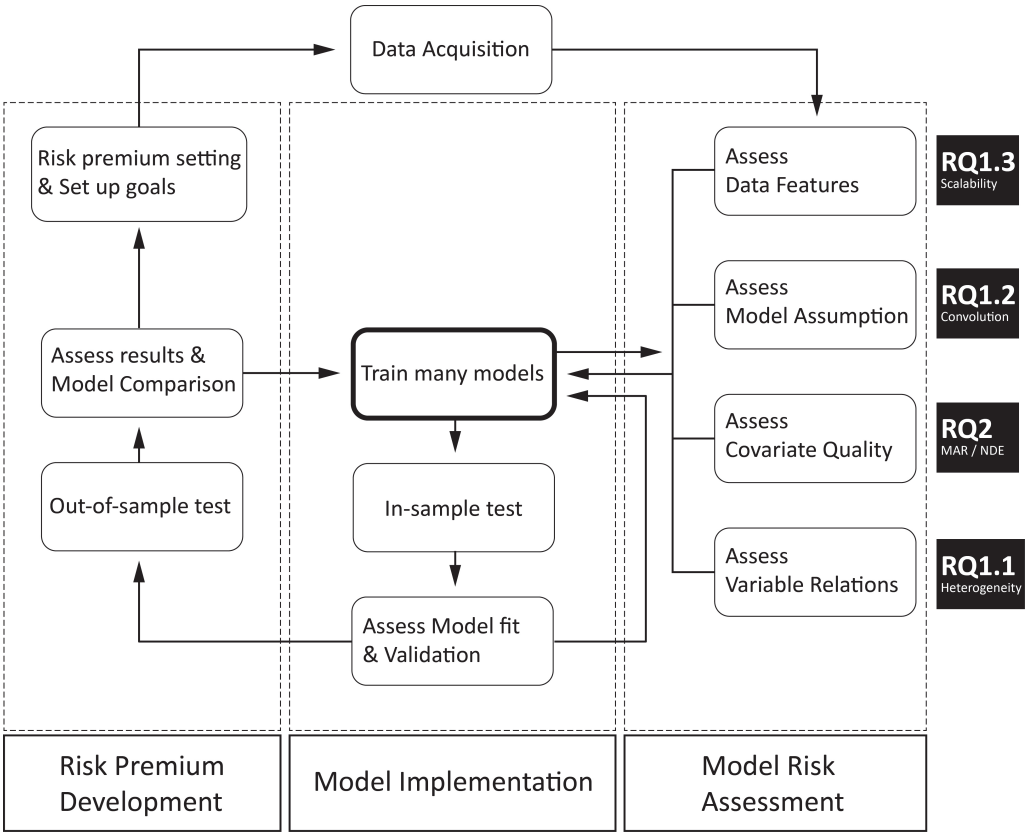

Figure 3.7: Overall risk premium development cycle designed for this thesis.

Additionally, we scrutinize model assumptions to pinpoint relevant distributional components along with their parameters, which concerns RQ1.2. The assessment of covariate quality (incompleteness) is crucial to prepare the error correction techniques for the proper model implementation, which concerns RQ2.1 and RQ2.2. Prior to model implementation, we also conduct an initial examination of variable relationships through exploratory data analysis to glean insights into the impact of risk factors, which concerns RQ1.1. These series of assessments together serve as the bedrock of our subsequent model implementation phase.

Building upon the insights from the model risk assessment phase, in the model implementation phase, we adjust our model training accordingly. The adjusted model training regimen is followed by in-sample testing and model fit assessment procedures to ensure the reliability of our models.

Upon completing the in-sample testing and model fit assessment, we progress to the prediction stage via out-of-sample testing. This marks the final phase of our

risk premium development cycle, wherein we validate the performance of our models against real-world data scenarios. Throughout this phase, we compare multiple models, including our primary model and rival models, across various performance metrics such as prediction accuracy, model risk correction accuracy, and uncertainty measurement. Unsatisfactory results prompt us to revisit and refine our model training methodologies, whereas satisfactory outcomes pave the way for meaningful interpretations and contributions to the new insight of risk premium development.

We expect that this systematic cycle streamlines the modeling process, ensuring the reliability and relevance of our experiments in addressing the model risk challenges for risk premium development.

Chapter 4

Bayesian Parametric: Hierarchical GLM with NDB Covariate

4.1 Introduction: RQ1.1, RQ2.2

In Chapter 3, we have discussed that there are two separate ways of viewing the expected aggregate claim $E[S_h]$ for a policy h.

- The Frequency-Severity approach for a policy h in Equation (3.1a)
- The Compound approach for a policy h in Equation (3.1b)

This chapter considers the 'Frequency-Severity' principle to risk premium modeling, under the assumption of independence between claim counts N_h and amounts $\bar{Y}_h$. Accordingly, the expected aggregate claim amount for a policy h can be defined as: $E[S_h] = E[N_h] \times E[\bar{Y}_h]$, where $\bar{Y}_h = \frac{1}{N_h}\Sigma Y_{hi}$ for $i = 1, \cdots, N_h$. Let $\mathbf{X} = \{\mathbf{X}^F, \mathbf{X}^S\}$ represent a set of covariate matrices used to model both the claim count N_h and the claim amount $\bar{Y}_h$. The covariate matrix for the claim amount $\bar{Y}_h$ is $\mathbf{X}^S = \{\mathbf{x}^S, \mathbf{z}^S\}$, where the continuous covariate vector $\mathbf{x}^S$ includes Non-Differential Berkson mismeasured values (i.e., NDB covariate). With the inclusion of the mismeasured covariate values x_h^S, however, model risks may arise, impeding the development of the predictive curve for S_h by introducing bias in the parameter estimations. This chapter assumes that the bias stems from the presence of heterogene-

ity (RQ1.1) in $S_h|\mathbf{X}$ and the inclusion of the NDB covariate (RQ2.2), resulting in $E[S_h|\mathbf{X}] \neq E[N_h|\mathbf{X}^F] \times E[\bar{Y}_h|\mathbf{X}^S]$. In this regard, an extensive body of literature attempts to tackle the model risks linked to the heterogeneity issue by proposing Generalized Linear Models (GLMs). Additionally, as we have seen, numerous studies have addressed the NDB covariates, employing error correction methods such as Regression Calibration (RC) or Simulation Extrapolation (SIMEX), which are introduced in Chapter 2 of this thesis. However, as per our current understanding, there appears to be no research on risk premium modeling that delves into integrated techniques addressing both heterogeneity (RQ1.1) and NDB error (RQ2.2) problems simultaneously.

4.2 Our Contribution

This chapter is dedicated to the development of a novel strategy for modeling the conditional aggregate claim amount $S_h|\mathbf{X}$ by dealing with the model risks: heterogeneity (RQ1.1) and NDB covariate (RQ2.2). Section 3.4.1 elaborates on how the hierarchical GLM, built upon finite parametric structures, mitigates the heterogeneity issue. As for the NDB covariate, Section 3.3.2 introduces Gustafson correction method to rectify mismeasured covariate values. We concentrate our attention on the hierarchical GLM and Gustafson correction, with the aim of establishing novel connections between them and consolidating them into the Bayesian parametric framework. Given that this chapter is grounded in the frequency-severity principle for risk premium modeling, the convolution issue (RQ1.2) does not fall within the scope of discussions in this chapter, as it pertains to the compound principle. The convolution issue will be addressed in later chapters.

The primary contribution of this chapter is to extend the Bayesian toolbox of the GLMs, which is known to be useful for risk premium modeling, by addressing NDB covariates through the development of the Gustafson correction technique and a novel prior knowledge elicitation for the variance of the NDB covariate conditional on the unknown true covariate, $\mathbf{x}^*|\mathbf{x}$. This is accomplished within the Bayesian

hierarchical GLM framework, which features varying parameters, each governed by its own model. To our knowledge, no prior attempts in risk premium modeling have employed the Gustafson correction within the Bayesian hierarchical GLM framework to mitigate the bias resulting from NDB covariate. This approach involves a complex analytical derivation, contingent upon the selection of distributional components for the prior, and requires evaluating the integral of the marginal densities. This may explain its under-utilization in the literature.

The performance of the Gustafson correction is assessed through the application of several models to an insurance dataset (introduced in Section 4.4.1). Specifically, the effectiveness of our Bayesian hierarchical GLM incorporating the Gustafson correction is compared against classical risk premium models employing the SIMEX approach (introduced in Section 2.1.1). The results demonstrate a significant improvement in risk premium predictions with the Gustafson error correction over SIMEX.

4.3 Modeling Method for $S_h|\mathbf{X}^F, \mathbf{X}^S$

4.3.1 Clustering $S_h|\mathbf{X}^F, \mathbf{X}^S$ with Complete Case Covariate

Given the covariates with correctly measured values, in Sections 2.1 and 3.1, we have underscored the importance of keeping each insured risk cluster as homogeneous as possible to tailor fair premiums. However, model risk - specifically the heterogeneity issue (RQ1.1) - arises with the inclusion of covariates that introduce variability within each risk cluster, making them more heterogeneous. Assuming the risk clusters $j = 1, \cdots, J$ have already been identified, this section introduces our baseline hierarchical GLM with varying coefficients to leverage the partial pooling technique for achieving homogeneous risk clusters. Given that the distribution choices for the outcome and covariates may raise questions, a detailed summary table in the Appendix D is provided for further reference.

Baseline GLM: For each policy $h = 1, \cdots, H$, we propose that the claim count N_h is negative binomial[1] distributed with expectation ξ_h and dispersion parameter ψ. Given the short policy periods typical in non-life insurance, we assume that the mean claim amount $\bar{Y}_h$ for policy h still follows a log-normal distribution. Thus, as is common in actuarial practice, individual mean claim amount on a log scale $\ln \bar{Y}_h$ for a policy h are assumed to be independent and Gaussian distributed with expectation μ_h and variance σ^2. In short, we present these two outcome models as

$$N_h \sim \mathbf{NB}(\xi_h,\ \psi) = \frac{\Gamma(N_h + \psi)}{N_h! \Gamma(\psi)} \left[\frac{\xi_h}{\xi_h + \psi}\right]^{N_h} \left[\frac{\psi}{\xi_h + \psi}\right]^{\psi} \tag{4.1a}$$

$$\bar{Y}_h \sim \mathbf{LogN}(\mu_h,\ \sigma^2) = \frac{1}{\bar{Y}_h \sqrt{2\pi\sigma^2}} \exp(-\frac{1}{2\sigma^2}\Big[\ln \bar{Y}_h - \mu_h\Big]^2) \tag{4.1b}$$

Now, with the inclusion of covariates $\mathbf{X} = \big(\mathbf{X}^F : \{\mathbf{x}^F,\ \mathbf{z}^F\},\ \mathbf{X}^S : \{\mathbf{x}^S,\ \mathbf{z}^S\}\big)$, the covariate effect and the information for risk cluster $j = 1, \cdots, J$ for an insured risk can be integrated with the outcome models through the expectation parameters: ξ_h and μ_h shown in Equation (4.1). To be specific, given that the covariates for claim count (frequency) and amount (severity) are denoted by $\mathbf{X}^F : \{\mathbf{x}^F, \mathbf{z}^F\}$ and $\mathbf{X}^S : \{\mathbf{x}^S, \mathbf{z}^S\}$ respectively, the expectation parameters take the form of GLMs:

$$\xi_h = E[N_h] = E[E[N_h|\mathbf{X}^F\boldsymbol{\beta}^F + \epsilon_h^F]] = E[\exp{(\mathbf{X}^F\boldsymbol{\beta}^F + \epsilon_h^F)}] \approx e^{\mathbf{X}^F\boldsymbol{\beta}^F} \tag{4.2a}$$

$$e^{\mu_h + \frac{1}{2}\sigma^2} = E[\bar{Y}_h] = E[E[\bar{Y}_h|\mathbf{X}^S\boldsymbol{\beta}^S + \epsilon_h^S]] = E[\exp{(\mathbf{X}^S\boldsymbol{\beta}^S + \frac{1}{2}\sigma^2 + \epsilon_h^S)}] \approx e^{\mathbf{X}^S\boldsymbol{\beta}^S + \frac{1}{2}\sigma^2} \tag{4.2b}$$

in which their residuals are normally distributed $\epsilon_h^F \sim \mathbf{N}(0, \sigma^2_{\epsilon^F}),\ \epsilon_h^S \sim \mathbf{N}(0, \sigma^2_{\epsilon^S})$; therefore, the essential property of GLMs listed in Equation (B.1) in Appendix B is ensured in Equation (4.2). In this setting, the conditional expected aggregate claim amount with $\mathbf{X} = \{\mathbf{X}^F,\ \mathbf{X}^S\}$ for a policy h is $E[S_h|\mathbf{X}] = \Sigma_{i=1}^{N_h} E[\bar{Y}_h|\mathbf{X}] = E[N_h|\mathbf{X}^F] \times E[\bar{Y}_h|\mathbf{X}^S],\ i = 1, \cdots, N_h$; therefore, the point estimate of the risk

[1]By convention, the claim count is typically assumed to follow a Poisson distribution. However, due to the heightened level of heterogeneity (e.g., overdispersion, unobserved intra-correlation, etc.), Winkelmann 2008 suggests considering a negative binomial distribution instead.

premium with covariates $\mathbf{X}^F$ and $\mathbf{X}^S$ is given by:

$$E[S_h|\mathbf{X}^F, \mathbf{X}^S] = \exp\left(\mathbf{X}^F\boldsymbol{\beta}^F + \mathbf{X}^S\boldsymbol{\beta}^S + \frac{1}{2}\sigma^2\right) \tag{4.3}$$

Hierarchical GLM with partial pooling: By including covariates - $\mathbf{X}^F$ and $\mathbf{X}^S$ - into the model, however, unobserved risk factors can be introduced, contributing to greater heterogeneity within each risk cluster $j = 1, \cdots, J$. To address this, we present cluster-specific GLM coefficient vectors and dispersion parameters $\boldsymbol{\beta}_j^F, \psi_j, \boldsymbol{\beta}_j^S, \sigma_j^2$ for all policies $h = 1, \cdots, H$. In addition to this, we consider a hierarchical GLM that employs individual models for each parameter to investigate each policy across all risk clusters; hence, the parameters can be either policy-specific or cluster-specific. Accordingly, Equation (4.3) with $j(h) \in \{1, \cdots, J\}$ is now redefined with the following prior selections:

$$E[S_{j(h)}|\mathbf{X}^F, \mathbf{X}^S] = \exp\left(\mathbf{X}^F\boldsymbol{\beta}_j^F + \mathbf{X}^S\boldsymbol{\beta}_j^S + \frac{1}{2}\sigma_j^2\right) \tag{4.4a}$$

$$\text{For } N_h \begin{cases} \boldsymbol{\beta}_j^F \mid \boldsymbol{\beta}_0^F, \Sigma_{\beta_0}^F \sim \mathbf{MVN}(\boldsymbol{\beta}_0^F,\ \Sigma_{\beta_0}^F) \\ \psi_j \mid u_0^F, v_0^F \sim \mathbf{Ga}(\frac{u_0^F}{2}, \frac{v_0^F}{2}) \end{cases} \tag{4.4b}$$

$$\text{For } \bar{Y}_h \begin{cases} \boldsymbol{\beta}_j^S \mid \boldsymbol{\beta}_0^S, \Sigma_{\beta_0}^S \sim \mathbf{MVN}(\boldsymbol{\beta}_0^S,\ \Sigma_{\beta_0}^S) \\ \sigma_j^2 \mid u_0^S, v_0^S \sim \mathbf{InvGa}(\frac{u_0^S}{2}, \frac{v_0^S}{2}) \end{cases} \tag{4.4c}$$

$$\text{For } \mathbf{X}^F \begin{cases} x_{j(h)}^F \sim \mathbf{N}(E[\mathbf{x}_j^F],\ \lambda_j^{2F}) \\ \qquad \lambda_j^{2F} \sim\ \mathbf{InvGa}(\frac{c_0^F}{2}, \frac{d_0^F}{2}) \\ z_{j(h)}^F \sim \mathbf{Bernoulli}(\pi_j^F) \\ \qquad \pi_j^F \sim\ \mathbf{Beta}(g_0^F, h_0^F) \end{cases} \tag{4.4d}$$

$$\text{For } \mathbf{X}^S \begin{cases} x_{j(h)}^S \sim \mathbf{N}(E[\mathbf{x}_j^S],\ \lambda_j^{2S}) \\ \qquad \lambda_j^{2S} \sim\ \mathbf{InvGa}(\frac{c_0^S}{2}, \frac{d_0^S}{2}) \\ z_{j(h)}^S \sim \mathbf{Bernoulli}(\pi_j^S) \\ \qquad \pi_j^S \sim\ \mathbf{Beta}(g_0^S, h_0^S) \end{cases} \tag{4.4e}$$

Equation (4.4) exemplifies a typical hierarchical Bayesian structure, serving as a Bayesian parametric approach. Gelman and Carlin 2013; Winkelmann 2008 suggest a Multivariate Gaussian prior for $\boldsymbol{\beta}$ due to the Normality assumption, a Gamma for ψ, and an Inverse Gamma for σ^2 due to its positive nature and adjustability. The hierarchical structure in Equation (4.4) integrates multiple layers, each of which interacts with the data in a different way to vary the degree of pooling. This partial pooling mechanism is highlighted in Section 3.4.1. In short, the model in Equation (4.4) investigates the unique parameter values for each observed individual (saturated cohort) and each cluster (reduced cohort), pooling them through multiple levels of distributions (risk clusters) rather than simply averaging the available information. This helps the model learn the unobservable structure more effectively because all clusters are simultaneously informed by what the model learns from each individual cluster. By employing this hierarchical structure, we expect that the model risk stemming from the heterogeneity in the conditional aggregate claim amount $S_h|\mathbf{X}$ within each cluster can be mitigated, which can potentially improve the predictive accuracy of the model (Gelman and Carlin 2013).

Assuming that the cluster memberships j are already determined, the hierarchical structure in Equation (4.4) allows the GLM coefficients and dispersion parameters - $\boldsymbol{\beta}_j^F, \psi_j, \boldsymbol{\beta}_j^S, \sigma_j^2$ - to vary by each cluster (i.e. no-pooling). Meanwhile, the corresponding hyperparameters $\boldsymbol{\beta}_0^F, \Sigma_{\beta_0}^F, u_0^F, v_0^F, \boldsymbol{\beta}_0^S, \Sigma_{\beta_0}^S, u_0^S, v_0^S$ are updated by the entire data (i.e. complete pooling) based on the hyperpriors defined as follows:

$$\left.\begin{aligned} \boldsymbol{\beta}_0^F \mid \underline{m}_0, \underline{\delta} &\sim \mathbf{MVN}(\underline{m}_0,\ \tfrac{1}{\underline{\delta}}\Sigma_{\beta_0}^F) \\ \Sigma_{\beta_0}^F \mid \underline{q}_0, \underline{\Lambda} &\sim \mathbf{IW}(\underline{q}_0,\ \underline{\Lambda}) \end{aligned}\right\} \text{for } \boldsymbol{\beta}_j^F \tag{4.5}$$

$$\left.\begin{aligned} u_0^F \mid \underline{\rho}_{u1}, \underline{\rho}_{u2} &\propto \underline{\rho}_{u1}^{\left(\frac{u_0^F}{2}\right)-1} / \ \Gamma\left(\tfrac{u_0^F}{2}\right)^{\underline{\rho}_{u2}} \\ v_0^F \mid \underline{\rho}_{v1}, \underline{\rho}_{v2} &\sim \mathbf{Ga}(\underline{\rho}_{v1},\ \underline{\rho}_{v2}) \end{aligned}\right\} \text{for } \psi_j \tag{4.6}$$

$$\left.\begin{aligned} \boldsymbol{\beta}_0^S \mid m_0, \delta &\sim \mathbf{MVN}(m_0,\ \tfrac{1}{\delta}\Sigma_{\beta_0}^S) \\ \Sigma_{\beta_0}^S \mid q_0, \Lambda &\sim \mathbf{IW}(q_0,\ \Lambda) \end{aligned}\right\} \text{for } \boldsymbol{\beta}_j^S \tag{4.7}$$

$$\left.\begin{aligned} u_0^S \mid \rho_{u1}, \rho_{u2} &\propto \rho_{u1}^{\left(\frac{u_0^S}{2}\right)-1} / \Gamma\left(\tfrac{u_0^S}{2}\right)^{\rho_{u2}} \\ v_0^S \mid \rho_{v1}, \rho_{v2} &\sim \mathbf{Ga}(\rho_{v1},\ \rho_{v2}) \end{aligned}\right\} \text{for } \sigma_j^2 \tag{4.8}$$

where $1/\delta$ is a variance inflation factor[2], m_0 and Σ_{β_0} are the mean vector and variance-covariance matrix of the GLM coefficients, while q_0 and Λ are the degrees of freedom and the scale matrix of an Inverse Wishart hyperprior to sample the variance-covariance matrix, respectively. As numbers of the degrees of freedom q_0 grows, its scale matrix Λ becomes smaller, and thus the variance-covariance matrix Σ_{β_0} becomes more influential (Kennedy and O'Hagan 2001). ρ_{u1} and ρ_{u2} are the shape and rate parameters of a Gamma hyperprior. Such distributional properties discussed here can be considered when we simulate posterior parameter samples with Gibbs sampling.

The choice of hyperpriors in Equations ($4.5 \sim 4.8$) is based on the distributions conjugate to the priors specified in Equation (4.4), which dramatically simplifies the Bayesian updating process. As for the form of the hyperpriors of u_0^F and u_0^S specified in Equations (4.6, 4.8), we adopt the analytically driven kernels by Fink 1997 because they are conjugate with the Gamma and Inverse Gamma distribution of ψ_j and σ_j^2 in Equations (4.4b, 4.4c), explaining their 'shape' parameters. This implies that the distributions of u_0^F and u_0^S will retain their original form of the kernels even after being updated with new values of ψ_j and σ_j^2.

To break down Equations ($4.4 \sim 4.8$), three different layers are involved in the parameter inferences as follows:

- data point level: $N_1, \{Y_{1(1)}, \cdots Y_{1(N_1)}\}, \ldots, N_H, \{Y_{H(1)}, \cdots Y_{H(N_H)}\} \mid \boldsymbol{\theta}_j$
- main parameter level: $\boldsymbol{\theta}_j = \{\boldsymbol{\beta}_j^F, \psi_j, \boldsymbol{\beta}_j^S, \sigma_j^2 \mid \boldsymbol{\phi}\}$
- hyperparameter level: $\boldsymbol{\phi} = \{\boldsymbol{\beta}_0^F, \Sigma_{\beta_0}^F, u_0^F, v_0^F, \boldsymbol{\beta}_0^S, \Sigma_{\beta_0}^S, u_0^S, v_0^S,$
  $\underline{m}_0, \underline{\delta}, \underline{q}_0, \underline{\Lambda},\ \underline{\rho}_{u1}, \underline{\rho}_{u2}, \underline{\rho}_{v1}, \underline{\rho}_{v2}, m_0, \delta, q_0, \Lambda, \rho_{u1}, \rho_{u2}, \rho_{v1}, \rho_{v2}\}$

[2]Variance Inflation Factor refers to the ratio of the virtual sample size to the observation sample size, which represents the impact of the prior. The default choice is $1/\delta = 100$, which means that the prior information is as important as the information brought by one data among 100 observations (Sharple 1990).

Note that the inference for the main parameter level is based on the cluster-specific data, whereas that for the hyperparameters is based on the entire dataset. However, assuming the hyperparameters $\underline{m}_0, \underline{\delta}, \underline{q}_0, \underline{\Lambda}, \ \underline{\rho}_{u1}, \underline{\rho}_{u2}, \underline{\rho}_{v1}, \underline{\rho}_{v2}, m_0, \delta, q_0, \Lambda, \rho_{u1}, \rho_{u2}, \rho_{v1}, \rho_{v2}$ to be fixed, the selection of these values is not trivial. As a golden rule, the flat hyperpriors[3] (assigning equal probability to all possible values of a parameter) should be ensured if proper knowledge about the hyperparameter values is inaccessible. More techniques to guide the choice of the hyperparameters can be found in Fink 1997; Kennedy and O'Hagan 2001; Bousquet 2008 and the references therein.

As a Bayesian parametric model, the hierarchical GLM requires the specification of the likelihood, prior, and hyperprior distributions that constitute each level of the hierarchy before computing posterior estimates for the parameters. This is because computing the marginal posterior means for the parameters of interest relies on the form of the joint distribution across all levels (Gelman and Carlin 2013). The original form of the cluster-specific joint probability (posterior $\propto$ likelihood $\times$ prior) based on Equation (4.4a) is given by:

$$\prod_{h=1}^{H} f(N_{j(h)}|\boldsymbol{\beta}_j^F, \psi_j) \prod_{i=1}^{N_j} f(Y_{j(hi)}|\boldsymbol{\beta}_j^S, \sigma_j^2)\ p(\boldsymbol{\beta}_j^F)\ p(\boldsymbol{\beta}_j^S)\ p(\psi_j)\ p(\sigma_j^2) \quad \text{for cluster } j \tag{4.9}$$

However, given the entire hierarchical setting in Equation (4.4), the baseline joint posterior for the cluster j (to utilize in the Gibbs sampler) takes the form:

$$\begin{aligned}
&\prod_{h=1}^{H} f(N_{j(h)}|\boldsymbol{\beta}_j^F, \psi_j) \prod_{i=1}^{N_j} f(Y_{j(hi)}|\boldsymbol{\beta}_j^S, \sigma_j^2) && \Rightarrow \text{likelihood model} \\
&\quad \times p(\boldsymbol{\beta}_j^F|\boldsymbol{\beta}_0^F, \Sigma_{\beta_0}^F)\ p(\boldsymbol{\beta}_j^S|\boldsymbol{\beta}_0^S, \Sigma_{\beta_0}^S)\ p(\psi_j|u_0^F, v_0^F)\ p(\sigma_j^2|u_0^S, v_0^S) && \Rightarrow \text{prior model} \\
&\quad \times p(\boldsymbol{\beta}_0^F|\underline{m}_0, \underline{\delta})\ p(\Sigma_{\beta_0}^F|\underline{q}_0, \underline{\Lambda})\ p(\boldsymbol{\beta}_0^S|m_0, \delta)\ p(\Sigma_{\beta_0}^S|q_0, \Lambda) && \Rightarrow \text{hyperprior model.I} \\
&\quad \times p(u_0^F|\underline{\rho}_{u1}, \underline{\rho}_{u2})\ p(v_0^F|\underline{\rho}_{v1}, \underline{\rho}_{v2})\ p(u_0^S|\rho_{u1}, \rho_{u2})\ p(v_0^S|\rho_{v1}, \rho_{v2}) && \Rightarrow \text{hyperprior model.II}
\end{aligned} \tag{4.10}$$

in which the hyperparameter layer is unaffected by the cluster membership j. To

[3]The flat prior reflects a state of complete ignorance about the parameter before observing any data, implying no value of the parameter is preferred over any other (Kennedy and O'Hagan 2001).

obtain the expected aggregate claim amount $E[S_{j(h)}|\mathbf{X}]$ for policy h in risk cluster j described in Equation (4.4a), it is now of interest to compute the marginal posterior mean for the main parameters such as $E[\boldsymbol{\beta}_j^F|N_h]$, $E[\psi_j|N_h]$, $E[\boldsymbol{\beta}_j^S|Y_{h1}, Y_{h2}, \cdots Y_{hN_h}]$, and $E[\sigma_j^2|Y_{h1}, Y_{h2}, \cdots Y_{hN_h}]$. This can be followed by constructing credibility intervals with the 5% level (given by the lower 2.5% and upper 2.5% of the posterior distribution for example) for Bayesian inference (to get the idea of the range of the true parameter values) of each varying coefficient and dispersion parameter for the cluster j.

Posterior Computation for Bayesian Inference: The conjugate hyperpriors in Equations (4.5 $\sim$ 4.8) lead to closed-form full conditional hyperpriors, allowing us to draw proper hyperparameter samples for computing the posterior parameter estimates (Gelman and Carlin 2013). This can be achieved using a Gibbs sampler as a hybrid Markov Chain Monte Carlo algorithm employed for both the negative binomial and log-normal GLMs. This random sampling technique relies on the joint conditional posterior distribution of all the parameters (specified in Equation (4.10)), which is derived from the closed-form full conditional hyperpriors (posterior hyperprior $\propto$ prior $\times$ hyperprior) listed viz:

$$\left.\begin{aligned} \boldsymbol{\beta}_0^F \mid \underline{m_0}, \underline{\delta}, N, \mathbf{X}^F,\ \Sigma_{\beta_0}^F \sim\ & \mathbf{MVN}\left(\tfrac{\underline{\delta}}{\underline{\delta}+1}\underline{m_0} + \tfrac{1}{\underline{\delta}+1}\boldsymbol{\beta}^F,\ \tfrac{\Sigma_{\beta_0}^F}{\underline{\delta}+1}\right) \\ \Sigma_{\beta_0}^F \mid \underline{q_0}, \underline{\Lambda}, N, \mathbf{X}^F,\ \boldsymbol{\beta}_0^F \sim\ & \mathbf{IW}\Big(\underline{q_0}+2,\ (\boldsymbol{\beta}_0^F - \boldsymbol{\beta}^F)(\boldsymbol{\beta}_0^F - \boldsymbol{\beta}^F)^T \\ & + \underline{\delta}(\boldsymbol{\beta}_0^F - \underline{m_0})(\boldsymbol{\beta}_0^F - \underline{m_0})^T + \underline{\Lambda}\Big) \end{aligned}\right\} \text{for } \boldsymbol{\beta}^F \quad (4.11)$$

$$\left.\begin{aligned} u_0^F \mid \underline{\rho}_{u1}, \underline{\rho}_{u2}, N, \mathbf{X}^F, v_0^F \propto\ & \left(\psi_j \cdot \tfrac{v_0^F}{2} \cdot \underline{\rho}_{u1}\right)^{\left(u_0^F/2\right)-1} / \ \Gamma\left(u_0^F/2\right)^{\underline{\rho}_{u2}+1} \\ v_0^F \mid \underline{\rho}_{v1}, \underline{\rho}_{v2}, N, \mathbf{X}^F, u_0^F \sim\ & \mathbf{Ga}\left(\underline{\rho}_{v1} + \tfrac{u_{0j}^F}{2},\ \underline{\rho}_{v2} + \psi_j\right) \end{aligned}\right\} \text{for } \psi_j \quad (4.12)$$

$$\left.\begin{aligned} \boldsymbol{\beta}_0^S \mid m_0, \delta, Y, \mathbf{X}^S,\ \Sigma_{\beta_0}^S \sim\ & \mathbf{MVN}\left(\tfrac{\delta}{\delta+1}m_0 + \tfrac{1}{\delta+1}\boldsymbol{\beta}^S,\ \tfrac{\Sigma_{\beta_0}^S}{\delta+1}\right) \\ \Sigma_{\beta_0}^S \mid q_0, \Lambda, Y, \mathbf{X}^S,\ \boldsymbol{\beta}_0^S \sim\ & \mathbf{IW}\Big(q_0+2,\ (\boldsymbol{\beta}_0^S - \boldsymbol{\beta}^S)(\boldsymbol{\beta}_0^S - \boldsymbol{\beta}^S)^T \\ & + \delta(\boldsymbol{\beta}_0^S - m_0)(\boldsymbol{\beta}_0^S - m_0)^T + \Lambda\Big) \end{aligned}\right\} \text{for } \boldsymbol{\beta}^S \quad (4.13)$$

$$\left.\begin{aligned} u_0^S \mid \rho_{u1}, \rho_{u2}, Y, \mathbf{X}^S, v_0^S &\propto \left(\tfrac{1}{\sigma_j^2} \cdot \tfrac{v_0^S}{2} \cdot \rho_{u1}\right)^{\left(u_0^S/2\right)-1} / \Gamma\left(u_0^S/2\right)^{\rho_{u2}+1} \\ v_0^S \mid \rho_{v1}, \rho_{v2}, Y, \mathbf{X}^S, u_0^S &\sim \mathbf{Ga}\left(\rho_{v1} + \tfrac{u_0^S}{2},\ \rho_{v2} + \tfrac{1}{2\sigma_j^2}\right) \end{aligned}\right\} \text{for } \sigma_j^2 \quad (4.14)$$

Equations (4.11 $\sim$ 4.14) address the hyperprior term $p(\boldsymbol{\beta}_0^F|\underline{m}_0, \underline{\delta})\ p(\Sigma_{\beta_0}^F|\underline{q}_0, \underline{\Lambda})\ p(\boldsymbol{\beta}_0^S|m_0, \delta)\times$ $p(\Sigma_{\beta_0}^S|q_0, \Lambda)\ \ p(u_0^F|\underline{\rho}_{u1}, \underline{\rho}_{u2})\ \ p(v_0^F|\underline{\rho}_{v1}, \underline{\rho}_{v2})\ \ p(u_0^S|\rho_{u1}, \rho_{u2})\ \ p(v_0^S|\rho_{v1}, \rho_{v2})$ in Equation (4.10). For the main parameter term $p(\boldsymbol{\beta}_j^F|\boldsymbol{\beta}_0^F, \Sigma_{\beta_0}^F)\ p(\boldsymbol{\beta}_j^S|\boldsymbol{\beta}_0^S, \Sigma_{\beta_0}^S)\ p(\psi_j|u_0^F, v_0^F)\ p(\sigma_j^2|u_0^S, v_0^S)$ in Equation (4.10), we utilize the Metropolis-Hastings (MH) algorithm within the Gibbs sampler because there are no conjugate priors for the main parameters $\boldsymbol{\beta}_j^F, \psi_j, \boldsymbol{\beta}_j^S, \sigma_j^2$ that are compatible with our negative binomial and log-normal outcomes.

Since we have two outcome models - claim counts N_h and claim amounts Y_h - a parallel execution of the two Gibbs samplers can be considered. Applied to our setting, each Gibbs sampler computes the posterior distribution of the varying coefficient and dispersion - $\boldsymbol{\beta}_j^F, \psi_j, \boldsymbol{\beta}_j^S, \sigma_j^2$ to evaluate Equation (4.4a), assuming no errors in the covariate. Figure 4.1 shows the process of re-estimating these model parameters for each cluster j using the two different Gibbs samplers.

Algorithm (D.1) in Appendix C outlines the details of the Gibbs sampler designed specifically for modeling claim amounts Y_h based on the log-normal density for illustrative purposes (the scenario for claim counts N_h based on the negative binomial would be similar to this, and thus repetitive). Before running the Gibbs sampler, it is essential to determine the initial values for the hyperparameters $\boldsymbol{\phi}$ **:** $\{\boldsymbol{\beta}_0^S, \Sigma_{\beta_0}^S, u_0^S, v_0^S, m_0, \delta, q_0, \Lambda, \rho_{u1}, \rho_{u2}, \rho_{v1}, \rho_{v2}, c_0^S, d_0^S, g_0^S, h_0^S\}$, which support the prior choices in Equations (4.4c, 4.4e) for the log-normal GLM. Based on these, the initial parameter values for the outcome $\boldsymbol{\theta}^{(old)}$: $\{\boldsymbol{\beta}_j^{S(old)}, \sigma_j^{2(old)}\}$ and covariate $\boldsymbol{w}$: $\{\pi_j^S, \lambda_j^{2S}\}$ can be obtained. These, in turn, provide the ultimate values for the 'communal hyperparameters' - $\boldsymbol{\beta}_0^{S+}, \Sigma_{\beta_0}^{S+}, u_0^{S+}, v_0^{S+}$ - which underpin both complete pooling and no pooling throughout the entire Gibbs sampling process. The Gibbs sampler for the log-normal model, with its two stages, is performed thus:

[Stage.1] Sampling with Complete Pooling

The Gibbs sampler initially computes the outcome parameters $\boldsymbol{\theta}$ without clustering,

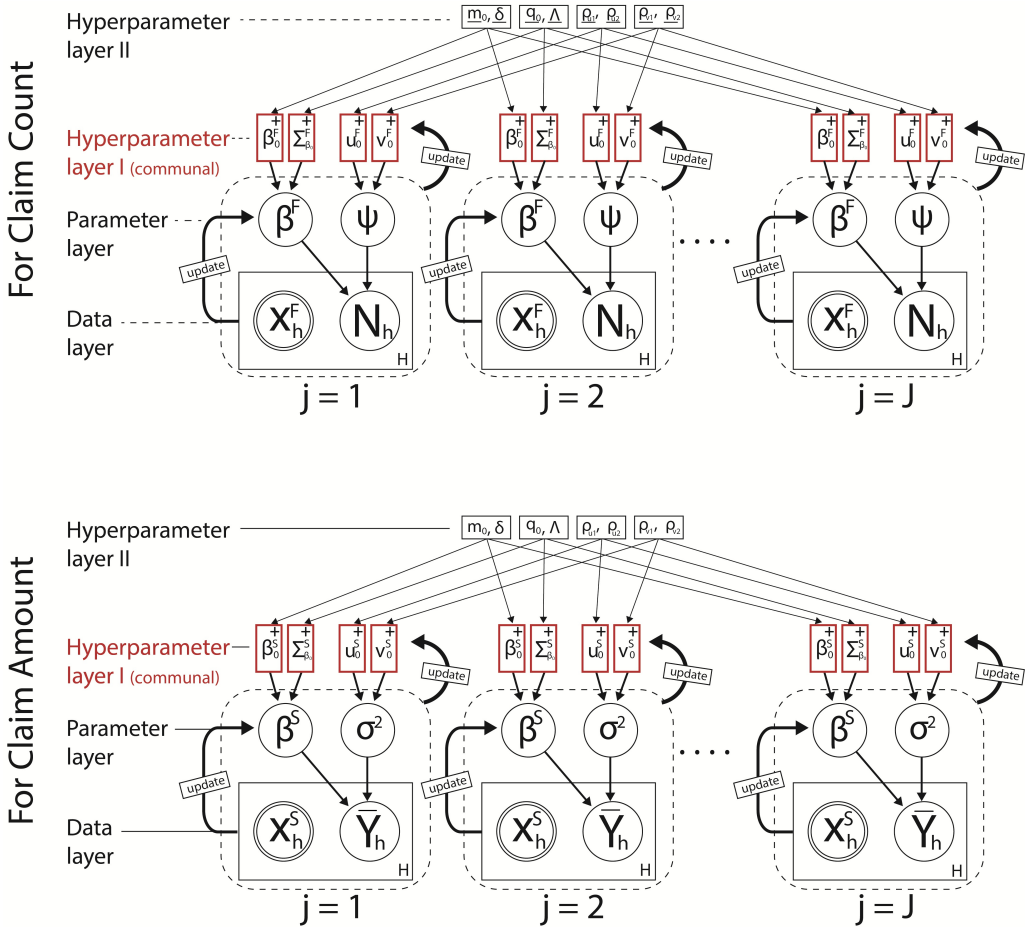

Figure 4.1: The acyclic graphical representation of the flows of the parameter updates in the hierarchical GLM. This is a snapshot for a single iteration (M=1).

a process characterized by complete pooling. The main goal of this stage is to refine appropriate communal hyperparameters $\boldsymbol{\beta}_0^{S+}, \Sigma_{\beta_0}^{S+}, u_0^{S+}, v_0^{S+}$ which are derived from the posterior hyperpriors discussed in Equations (4.13, 4.14). Since the log-normal outcome lacks conjugate priors for $\boldsymbol{\theta}$, the Metropolis-Hastings (MH) algorithm can be employed. Considering the prior as the proposal distribution, the proposal samples $\boldsymbol{\theta}^{(new)}$ are drawn based on the communal hyperparameters $\boldsymbol{\beta}_0^{S+}, \Sigma_{\beta_0}^{S+}, u_0^{S+}, v_0^{S+}$. The resulting samples $\boldsymbol{\theta}^{(*)}$ describe the global mean $E[\bar{Y}_h|\mathbf{X}]$, which again updates the communal hyperparameters above. See Algorithm (D.1) in Appendix C.

[Stage.2] Sampling without Pooling

Assuming that the cluster membership j is already determined, this stage aims to produce accurate parameter estimates $\boldsymbol{\theta}^{(*)}$ for each cluster. These estimates

are informed by the communal hyperparameters $\boldsymbol{\beta}_0^{S+}, \Sigma_{\beta_0}^{S+}, u_0^{S+}, v_0^{S+}$ derived from [**Stage.1**]. By leveraging the communal hyperparameters, the Gibbs sampler ensures that the resulting parameter estimates $\boldsymbol{\theta}_j^{(*)}$ are optimized to enhance the homogeneity within each risk cluster as much as possible, which mitigates within-cluster variability (heterogeneity). The log-likelihood value calculated at the end of each sampling process helps track the convergence of the estimates. The implementation detail is provided in Algorithm (D.1) in Appendix C.

4.3.2 Clustering $S_h|\mathbf{X}^F, \mathbf{X}^S$ with NDB Case Covariate

Section 3.3 has discussed how the model risk associated with a mismeasured covariate increases risk exposure during the risk premium modeling process. In this section, we present our novel approach to addressing the NDB covariate (RQ2.2), developed upon the hierarchical GLM and partial pooling technique for risk premium modeling. In line with the parametric Bayesian principle, we assume that the risk clusters $j = 1, \cdots J$ have been already identified.

Considering that we have two types of covariate matrices denoted by $\mathbf{X}^F$: $\{\mathbf{x}^F, \mathbf{z}^F\}$ for the negative binomial outcome N_h and $\mathbf{X}^S$: $\{\mathbf{z}^S, \mathbf{x}^S\}$ for the log-normal outcome Y_h respectively, suppose the continuous covariate $\mathbf{x}^S$ has mismeasured values with the errors characterized by the Non-Differential Berkson (NDB) type. Assuming that the covariates for the negative binomial outcome are complete, the discussion in this section is mainly centered on the hierarchical model based on the log-normal outcome and its covariates $\mathbf{x}^S$. For the sake of simplicity, in this section, we will omit the superscript S from the continuous covariate $\mathbf{x}^S$, thus we express $\mathbf{x}^S$ as simply $\mathbf{x}$.

In Section 3.3.2, we have described Gustafson's Bayesian error correction framework to mitigate the model risk associated with the NDB covariate. The underlying idea is that the parameter values for the true covariate can be estimated by developing a special joint model that leverages prior knowledge on θ_j, which is extracted from the linking component $f(\mathbf{x}^*|\mathbf{x}, \theta_j)$. This component encapsulates the cluster-

wise relationship between the true covariate $\mathbf{x}$ and the NDB covariate $\mathbf{x}^*$ (Grace et al. 2021). Hence, our primary goal in this chapter is to obtain accurate parameter values θ_j for the special joint model.

To elaborate, the joint model that encompasses the outcome Y, the NDB covariate $\mathbf{x}^*$, the true covariate $\mathbf{x}$, and the additional covariate $\mathbf{z}$ is given by

$$f(Y, \mathbf{x}^*, \mathbf{x}|\mathbf{z}) = \underbrace{f(Y|\mathbf{x}^*, \mathbf{x}, \mathbf{z})}_{\text{outcome}} \times \underbrace{f(\mathbf{x}^*|\mathbf{x}, \mathbf{z})}_{\text{linking component}} \times \underbrace{f(\mathbf{x}|\mathbf{z})}_{\text{covariate}} \tag{4.15}$$

In Gustafson's terminology, Equation (4.15) is referred to as the *complete joint* model since it contains the true covariate $\mathbf{x}$ that is unknown in real life. As discussed in Section 3.3.2, the full joint distribution can be simplified as shown in Equation (4.15) because the NDB covariate $\mathbf{x}^*$ is uncorrelated with any other variables except the true covariate $\mathbf{x}$ itself. To incorporate Gustafson's complete model for NDB covariate in Equation (4.15) into the hierarchical GLM framework, we re-define the form of the risk premium model with the priors, originally specified in Equation (4.4), as follows:

$$E[S_{j(h)}|\mathbf{X}^F, \mathbf{X}^{S*}] = \exp\left(\mathbf{X}^F\boldsymbol{\beta}_j^F + \mathbf{X}^{S*}\boldsymbol{\beta}_j^S + \frac{1}{2}\sigma_j^2\right) \tag{4.16a}$$

$$\text{For } N_h \begin{cases} \boldsymbol{\beta}_j^F \mid \boldsymbol{\beta}_0^F, \Sigma_{\beta_0}^F \sim \mathbf{MVN}(\boldsymbol{\beta}_0^F,\ \Sigma_{\beta_0}^F) \\ \psi_j \mid u_0^F, v_0^F \sim \mathbf{Ga}(\frac{u_0^F}{2}, \frac{v_0^F}{2}) \end{cases} \tag{4.16b}$$

$$\text{For } \bar{Y}_h \begin{cases} \boldsymbol{\beta}_j^S \mid \boldsymbol{\beta}_0^S, \Sigma_{\beta_0}^S \sim \mathbf{MVN}(\boldsymbol{\beta}_0^S,\ \Sigma_{\beta_0}^S) \\ \sigma_j^2 \mid u_0^S, v_0^S \sim \mathbf{InvGa}(\frac{u_0^S}{2}, \frac{v_0^S}{2}) \end{cases} \tag{4.16c}$$

$$\text{For } \mathbf{X}^F \begin{cases} x_{j(h)}^F \sim \mathbf{N}(E[\mathbf{x}_j^F],\ \lambda_j^{2F}) \\ \lambda_j^{2F} \sim\ \mathbf{InvGa}(\frac{c_0^F}{2}, \frac{d_0^F}{2}) \\ z_{j(h)}^F \sim \mathbf{Bernoulli}(\pi_j^F) \\ \pi_j^F \sim\ \mathbf{Beta}(g_0^F, h_0^F) \end{cases} \tag{4.16d}$$

$$
\text{For } \mathbf{X}^{S*} \begin{cases} x^{S*}_{j(h)} | x^{S}_{j(h)} \sim \mathbf{N}(\mathbf{x}^{S}_{j(h)},\ \tau_j^2) \\ \qquad\qquad \tau_j^2 \sim \text{undetermined} \\ x^{S}_{j(h)} | z^{S}_{j(h)} \sim \mathbf{N}(\kappa_{j0} + \kappa_{j1} z^{S}_{j(h)},\ \lambda_j^{2S}) \\ \qquad\qquad \boldsymbol{\kappa}_j \sim \mathbf{MVN}(\tilde{\boldsymbol{\kappa}},\ \lambda_j^{2S}\tilde{\Sigma}_\kappa) \\ \qquad\qquad \lambda_j^{2S} \sim \mathbf{InvGa}(\frac{c_0^S}{2}, \frac{d_0^S}{2}) \\ z^{S}_{j(h)} \sim \mathbf{Bernoulli}(\pi_j^S) \\ \qquad\qquad \pi_j^S \sim \mathbf{Beta}(g_0^S, h_0^S) \end{cases} \tag{4.16e}
$$

Once again, within the broader hierarchical structure outlined in Equation (4.16), in which both models for $N_h|\mathbf{X}^F$ and $\bar{Y}_h|\mathbf{X}^{S*}$ are incorporated to compute $S_h|\mathbf{X}$, the following discussion focuses exclusively on modeling the claim amount $\bar{Y}_h|\mathbf{X}^{S*}$ for demonstration purposes. This is based on the previous assumption that the claim amount model of $\bar{Y}_h|\mathbf{X}^{S*}$ is solely influenced by the NDB covariate $\mathbf{x}^*$. To simplify notation, we drop the superscript S from the covariate matrix $\mathbf{X}^{S*}$ and the continuous covariate vector $\mathbf{x}^{S*}$ referring to them simply as $\mathbf{X}^*$ and $\mathbf{x}^*$ respectively.

To begin with, we attempt to construct the complete joint, described in Equation (4.15), in order to generate the claim amount $\bar{Y}_h|\mathbf{X}^*$. This involves specifying the *linking component* (measurement model in Gustafson's terms (Gustafson 2008)) that relates $\mathbf{x}^*$ to $\mathbf{x}$ through a hierarchical structure. For this chapter, we define the linking component given by (since x^* is still considered a normally distributed random variable)

$$
f_N(x^*|x) = \frac{1}{\sqrt{2\pi\tau_j^2}} \exp\{\frac{-(x^* - x)^2}{2\tau_j^2}\} \tag{4.17}
$$

where $x^*|x = x + \epsilon_j \sim \mathbf{N}(x, \tau_j^2)$, $\quad \tau_j^2 : V(\mathbf{x}^*|\mathbf{x}), \quad \epsilon_j \sim \mathbf{N}(0, \sigma_{j\epsilon}^2)$, and $\tau_j^2 = \sigma_x^2 + \sigma_{j\epsilon}^2$. The implication is that the prior knowledge for the dispersion τ_j^2 in Equation (4.17) accounts for the cluster-specific characteristics of the NDB covariate, basing it on the relationship between the NDB covariate $\mathbf{x}^*$ and the true covariate $\mathbf{x}$. However, estimating the value of τ_j^2 at the outset is not straightforward since the cluster-wise error variance $\sigma_{j\epsilon}^2$ is inaccessible. This concern is indicated in Equation (4.16e).

Additionally, to fully construct the joint model in Equation (4.15), we also define the *outcome* and the *covariate* components (called 'exposure models' in Gustafson's terms (Gustafson 2008)) as detailed below:

$$f_{LN}(\bar{Y}|x,z) = \frac{1}{\bar{Y}\sqrt{2\pi\sigma_j^2}} \exp\{-\frac{1}{2}\Big[\frac{\log\bar{Y} - (\beta_{j0} + \beta_{j1}x + \beta_{j2}z)}{\sigma_j}\Big]^2\} \tag{4.18a}$$

$$f_N(x|z) = \frac{1}{\sqrt{2\pi\lambda_j^2}} \exp\{\frac{-(x - \{\kappa_{j0} + \kappa_{j1}z\})^2}{2\lambda_j^2}\} \tag{4.18b}$$

where $\bar{Y}|x,z \sim \mathbf{LN}(\mathbf{X}\boldsymbol{\beta}_j, \sigma_j^2)$, $\sigma_j^2 : V(\bar{Y}|\mathbf{X})$, $x|z \sim \mathbf{N}(\kappa_{j0} + \kappa_{j1}z,\ \lambda_j^2)$, and $\lambda_j^2 : V(\mathbf{x}|\mathbf{z})$ as shown in Equations (4.16c and 4.16e). Note that Equations (4.17, 4.18) serve as fundamental components, as they must be multiplied together to construct the complete joint presented in Equation (4.15). However, these models in Equation (4.18) are largely hypothetical since the true covariate $\mathbf{x}$ is unknown. Instead, the available models we can utilize for a practical implementation are

$$f_{LN}(\bar{Y}|x^*,z) = \frac{1}{\bar{Y}\sqrt{2\pi\widehat{\sigma}_j^2}} \exp\{-\frac{1}{2}\Big[\frac{\log\bar{Y} - (\widehat{\beta}_{j0} + \widehat{\beta}_{j1}x^* + \widehat{\beta}_{j2}z)}{\widehat{\sigma}_j}\Big]^2\} \tag{4.19a}$$

$$f_N(x^*|z) = \frac{1}{\sqrt{2\pi\widehat{\lambda}_j^2}} \exp\{\frac{-(x^* - \{\widehat{\kappa}_{j0} + \widehat{\kappa}_{j1}z\})^2}{2\widehat{\lambda}_j^2}\} \tag{4.19b}$$

where $\bar{Y}|x^*,z \sim \mathbf{LN}(\mathbf{X}^*\widehat{\boldsymbol{\beta}}_j, \widehat{\sigma}_j^2)$, $\widehat{\sigma}_j^2 : V(\bar{Y}|\mathbf{X}^*)$, $x^*|z \sim \mathbf{N}(\widehat{\kappa}_{j0} + \widehat{\kappa}_{j1}z,\ \widehat{\lambda}_j^2)$, and $\widehat{\lambda}_j^2 : V(\mathbf{x}^*|\mathbf{z})$ (the notation $\widehat{\cdot}$ is used to indicate that these parameters are derived from the covariate with NDB errors, prior to correction). Multiplying these two models - the outcome model and the covariate model - in Equation (4.19), we arrive at the *incomplete joint model* in Equation (4.20), which represents the most feasible solution available in practice.

$$\begin{aligned} f(\bar{Y}, x^* \mid z) = \frac{1}{\bar{Y}(2\pi)\widehat{\sigma}_j\widehat{\lambda}_j} &\times \exp\Big(-\frac{1}{2\widehat{\sigma}_j^2}\Big[(\log\bar{Y} - \widehat{\beta}_{j0} - \widehat{\beta}_{j2}z) - \widehat{\beta}_{j1}x^*\Big]^2\Big) \\ &\times \exp\Big(-\frac{1}{2\widehat{\lambda}_j^2}\Big[x^* - (\widehat{\kappa}_{j0} + \widehat{\kappa}_{j1}z)\Big]^2\Big) \end{aligned} \tag{4.20}$$

It is true that this joint model is characterized by its limitation: the absence of

the true covariate $\mathbf{x}$. Fortunately, bridging the gap between the complete joint model in Equation (4.15) and the incomplete joint model in Equation (4.20) is straightforward. As mentioned in Section 3.3.2, we can marginalize the complete joint model in Equation (4.15) over the true covariate $\mathbf{x}$ by applying the following integral:

$$\begin{aligned}
&\int f(\bar{Y}, x^*, x \mid z)\, dx \\
&\quad = \int_{\mathbf{x}} \frac{1}{\sigma_j \bar{Y}\sqrt{2\pi}} \exp\{-\frac{1}{2}\Big[\frac{\log \bar{Y} - (\beta_{j0} + \beta_{j1}x + \beta_{j2}z)}{\sigma_j}\Big]^2\} \\
&\qquad\qquad \times \frac{1}{\sqrt{2\pi\tau_j^2}} \exp\{\frac{-(x^* - x)^2}{2\tau_j^2}\} \times \frac{1}{\sqrt{2\pi\lambda_j^2}} \exp\{\frac{-(x - \{\kappa_{j0} + \kappa_{j1}z\})^2}{2\lambda_j^2}\}\, dx
\end{aligned} \tag{4.21}$$

The evaluation process for the integral in Equation (4.21) is detailed in D.2 in Appendix C, and the resulting solution is presented below.

$$\begin{aligned}
&\int f(\bar{Y}, x^*, x \mid z)\, dx \\
&= \frac{1}{\bar{Y}(2\pi)\sigma_j\tau_j\lambda_j}\Big(\frac{\sigma_j^2\lambda_j^2\tau_j^2}{\beta_{j1}^2\tau_j^2\lambda_j^2 + \sigma_j^2\lambda_j^2 + \sigma_j^2\tau_j^2}\Big)^{1/2} \\
&\quad \times \exp\Bigg(-\frac{1}{2}\Big(\frac{\lambda_j^2 + \tau_j^2}{\beta_{j1}^2\tau_j^2\lambda_j^2 + \sigma_j^2\lambda_j^2 + \sigma_j^2\tau_j^2}\Big)\Big[(\log \bar{Y} - \beta_{j0} - \beta_{j2}z) - \frac{\beta_{j1}\Big(\frac{x^*}{\tau_j^2} + \frac{\kappa_{j0}+\kappa_{j1}z}{\lambda_j^2}\Big)}{\frac{1}{\tau_j^2} + \frac{1}{\lambda_j^2}}\Big]^2\Bigg) \\
&\quad \times \exp\Bigg(-\frac{1}{2}\Big(\frac{1}{\tau_j^2 + \lambda_j^2}\Big)\Big[\Big(x^* - (\kappa_{j0} + \kappa_{j1}z)\Big)^2\Big]\Bigg) = f(\bar{Y}, x^* \mid z)
\end{aligned} \tag{4.22}$$

from where the unobservable true covariate term $\mathbf{x}$ is completely eliminated. Consequently, the solution of the integral in Equation (4.22), derived from the complete joint model, can be aligned with the incomplete joint model in Equation (4.20). The key insight here is that, although these two solutions originate from different equations, they both describe the same joint model $f(\bar{Y}, x^* \mid z)$ within a practical framework that does not rely on evaluating or sampling of the true covariate $\mathbf{x}$. Therefore we can establish a direct relationship between the parameters of the com-

plete joint model and those of the incomplete joint model by matching the respective parameterizations.

More specifically, this relationship is expressed through a system of equations. Given that the parameters of the incomplete joint model - $\widehat{\beta}_{j0}, \widehat{\beta}_{j1}, \widehat{\beta}_{j2}, \widehat{\sigma}_j^2, \widehat{\lambda}_j^2, \widehat{\kappa}_{j0}, \widehat{\kappa}_{j1}$ - in Equation (4.20) are readily accessible, the parameters of the marginalized version of the complete joint model - $\beta_{j0}, \beta_{j1}, \beta_{j2}, \sigma_j^2, \lambda_j^2, \kappa_{j0}, \kappa_{j1}$ - in Equation (4.22) can be re-expressed in terms of their counterparts in the incomplete joint model, which effectively bridges the gap between the hypothetical and accessible parameters. By solving this system of equations, we might be able to develop a practical framework for correcting parameter estimates in the presence of NDB covariate $\mathbf{x}^*$, thereby mitigating covariate-based model risk (RQ2.2).

All derivations of the system of equations and detailed explanations can be found in D.2 in Appendix C, and the resulting system of equations for the parameters in the complete joint model is presented as follows.

$$\lambda_j^2 = \widehat{\lambda}_j^2 - \tau_j^2 \tag{4.23a}$$

$$\kappa_{j0} = \widehat{\kappa}_{j0} \tag{4.23b}$$

$$\kappa_{j1} = \widehat{\kappa}_{j1} \tag{4.23c}$$

$$\beta_{j1} = \frac{\widehat{\beta}_{j1}\widehat{\lambda}_j^2}{\widehat{\lambda}_j^2 - \tau_j^2} \tag{4.23d}$$

$$\beta_{j0} = \widehat{\beta}_{j0} - \frac{\widehat{\beta}_{j1}\widehat{\kappa}_{j0}\tau_j^2}{\widehat{\lambda}_j^2 - \tau_j^2} \tag{4.23e}$$

$$\beta_{j2} = \widehat{\beta}_{j2} - \frac{\widehat{\beta}_{j1}\widehat{\kappa}_{j1}\tau_j^2}{\widehat{\lambda}_j^2 - \tau_j^2} \tag{4.23f}$$

$$\sigma_j^2 = \widehat{\sigma}_j^2 - \frac{\beta_{j1}^2\tau_j^2(\widehat{\lambda}_j^2 - \tau_j^2)}{\widehat{\lambda}_j^2} \tag{4.23g}$$

The prior for τ_j^2 and scaling factor ζ: As can be seen from Equation (4.23), the accurate estimation of the parameters in the complete joint model largely depends on the value of τ_j^2. However, we have previously mentioned that it is not straightforward to estimate the value of τ_j^2 until the relationship between the NDB

covariate $\mathbf{x}^*$ and the true covariate $\mathbf{x}$ is clarified. Thanks to the system of equations, specifically Equation (4.23a), we can now see that τ_j^2 as $\widehat{\lambda}_j^2 - \lambda_j^2$. In other words, $V(\mathbf{x}^*|\mathbf{x}) = V(\mathbf{x}^*|\mathbf{z}) - V(\mathbf{x}|\mathbf{z})$. From this, several important findings come to light:

(i) $\widehat{\lambda}_j^2 : V(\mathbf{x}^*|\mathbf{z})$ is always greater than $\lambda_j^2 : V(\mathbf{x}|\mathbf{z})$ according to Equation (4.23a)

(ii) Given that $x^*|z \sim \mathbf{N}(\widehat{\kappa}_{j0} + \widehat{\kappa}_{j1}z,\ \widehat{\lambda}_j^2)$ and $x|z \sim \mathbf{N}(\kappa_{j0} + \kappa_{j1}z,\ \lambda_j^2)$, it appears that $\widehat{\kappa}_{j0} = \kappa_{j0}$ and $\widehat{\kappa}_{j1} = \kappa_{j1}$, hence $E[\mathbf{x}^*|\mathbf{z}] = E[\mathbf{x}|\mathbf{z}]$ according to Equation (4.23b and 4.23c).

(iii) Given (i) and (ii), it is safe to say that the variance of the true covariate can be a scalar multiple of the variance of the NDB covariate: $\lambda_j^2 = \zeta \times \widehat{\lambda}_j^2$ where $0 < \zeta < 1$ is a *scaling factor*.

Equation (4.23a), along with the key findings (i), (ii), and (iii), form the foundation of essential prior knowledge regarding the dispersion parameter of the linking component $\tau_j^2 : V(\mathbf{x}^*|\mathbf{x})$. Together, they highlight that

- $\tau_j^2 = \widehat{\lambda}_j^2 - \lambda_j^2$ i.e., $V(\mathbf{x}^*|\mathbf{x}) = V(\mathbf{x}^*|\mathbf{z}) - V(\mathbf{x}|\mathbf{z})$ from Equation (4.23a).
- $\lambda_j^2 = \zeta \times \widehat{\lambda}_j^2$ i.e., $V(\mathbf{x}|\mathbf{z}) = \zeta \times V(\mathbf{x}^*|\mathbf{z})$ from the findings (i),(ii),(iii).

Ultimately, the relationships outlined above can be distilled into the equation:

$$\tau_j^2 = (1-\zeta)\widehat{\lambda}_j^2 \quad \text{or} \quad V(\mathbf{x}^*|\mathbf{x}) = (1-\zeta)\ V(\mathbf{x}^*|\mathbf{z}) \tag{4.24}$$

which illustrates how τ_j^2 can be accounted for by the fraction $(1-\zeta)$ of the variance $\widehat{\lambda}_j^2 : V(\mathbf{x}^*|\mathbf{z})$; therefore, if the estimate of $\widehat{\lambda}_j^2$ is available, the value of τ_j^2 can be determined by the scaling factor $0 < \zeta < 1$. The implication is that, since both $\mathbf{x}^*$ and $\mathbf{z}$ are available, we can leverage $\widehat{\lambda}_j^2 : V(\mathbf{x}^*|\mathbf{z})$ as a proxy for estimating $\tau_j^2 : V(\mathbf{x}^*|\mathbf{x})$. The scaling factor $0 < \zeta < 1$ quantifies the confidence level regarding the adequacy of the known covariate $\mathbf{z}$ as a substitute for the unobservable true covariate $\mathbf{x}$. When the scaling factor ζ is set high, τ^2 incorporates less information from $\mathbf{x}^*|\mathbf{z}$ to capture the knowledge on $\mathbf{x}^*|\mathbf{x}$. Conversely, a lower ζ increases reliance on the observable covariate $\mathbf{x}^*|\mathbf{z}$ to account for $\mathbf{x}^*|\mathbf{x}$.

For this chapter, we suggest using this finding on τ_j^2 as the prior knowledge for the probability distribution of $\mathbf{x}^*|\mathbf{x}$ that explains the cluster-wise relationship between the NDB covariate $\mathbf{x}^*$ and the true covariate $\mathbf{x}$. In addition to this, we incorporate a sensitivity analysis component for ζ, which will allow us to evaluate how variations in ζ affect the estimates of τ_j^2. Consequently, the sensitivity analysis will offer insights into the dependency of the scaling factor ζ on the performance of this error correction technique, and help identify the optimal value of τ_j^2 to improve estimation results affected by the NDB covariate $\mathbf{x}^*$.

Gibbs sampler modification with the Gustafson correction: We propose the following modifications (additional steps) to incorporate the resulting system of equations in Equation (4.23) into the Gibbs sampler for our hierarchical GLM outlined in Algorithm (D.1) in Appendix C:

(a) In line 15 of Algorithm (D.1), assuming the NDB covariate value in $\mathbf{x}$ at observation h within risk cluster j, we introduce an additional step to draw the posterior parameter samples for the linking component in Equation (4.17) and the covariate model in Equation (4.19b). This allows us to perform the parameter adjustment using the system of equations in line 22.

$$\boldsymbol{w}_j : \begin{cases} \pi_j \sim \textbf{Beta}(g_0 + \Sigma z_j,\ h_0 + n_j - \Sigma z_j) \\ \widehat{\boldsymbol{\kappa}}_j \sim \textbf{MVN}\Big(\Big[(\tilde{\Sigma}_k^{-1} + \mathbb{K}_1^T\mathbb{K}_1)^{-1}(\tilde{\Sigma}_k^{-1}\tilde{\boldsymbol{\kappa}} + \mathbb{K}_2)\Big],\ \widehat{\lambda}_j^2\Big[\tilde{\Sigma}_k^{-1} + \mathbb{K}_1^T\mathbb{K}_1\Big]^{-1}\Big) \\ \widehat{\lambda}_j^2 \sim \textbf{InvGa}\big(\frac{c_0+n_j}{2},\ \frac{1}{2}(d_0 + \Sigma(x_j - \widehat{\kappa}_{0j} + \widehat{\kappa}_{1j}z_j)^2)\big) \\ \tau_j^2 = (1-\zeta)\widehat{\lambda}_j^2 \end{cases} \tag{4.25}$$

The derivations of the posterior densities for the covariate model parameters - $\pi_j, \widehat{\boldsymbol{\kappa}}_j, \widehat{\lambda}_j^2$ - are thoroughly detailed in F.2.3 in Appendix F. The value of the scaling factor ζ is predetermined by the researchers based on the sensitivity analysis result, ensuring that the chosen value reflects the expected level of errors in the given NDB covariate under different scenarios. Further discussion

of this experiment is provided in the next section.

(b) In line 22 of Algorithm (D.1), we apply the resulting system of equations presented in Equation (4.23) to adjust the estimated outcome parameter values - $\boldsymbol{\theta}_j^{(*)} : \{\boldsymbol{\beta}_j^{(*)}, \sigma_j^{2(*)}\}$, using the parameter samples of the incomplete joint model - $\widehat{\beta}_{j0}, \widehat{\beta}_{j1}, \widehat{\beta}_{j2}, \widehat{\sigma}_j, \widehat{\lambda}_j, \widehat{\kappa}_{j0}, \widehat{\kappa}_{j1}$ -, and the variance $\tau_j^2 = (1-\zeta)\widehat{\lambda}_j^2$ (where $0 < \zeta < 1$) of the linking component in Equation (4.17). Note that the parameter samples drawn from the incomplete joint model in the Gibbs sampler need to meet certain criteria dictated by Equation (4.23). For example, as specified in Equation (4.23a), $\widehat{\lambda}_j^2$ must always be greater than λ_j^2, and, based on Equation (4.23g), $\widehat{\sigma}_j^2$ must always exceed the value given by $\frac{\beta_{j1}^2 \tau_j^2 \lambda_j^2}{\widehat{\lambda}_j^2}$. These conditions ensure that the sampled parameters maintain valid relationships with the true parameters, and the Gibbs sampler filters out samples that do not meet these criteria.

4.4 Numerical Experiments with NDB Covariate

4.4.1 Data: Local Government Property Insurance Fund

We assess our hierarchical GLM using an insurance dataset drawn from the Wisconsin Local Government Property Insurance Fund (LGPIF). Compiled by the actuarial research team at the University of Wisconsin, this dataset provides information on insurance coverage from $H = 1,679$ policies for various government building units across Wisconsin. Additional insights and comprehensive details about their project are available on the official website[4]. The dataset presents a unique set of challenges, including unobservable heterogeneity (RQ1.1) in the log-normal outcome variable conditioned on the NDB covariate (RQ2.2). Given that this chapter adopts the frequency-severity approach to risk premium modeling, our dataset incorporates four covariates: two for the claim count model - a binary covariate $\mathbf{z}^F$ ($AC15$: 1 or 0)

[4]LGPIF: https://sites.google.com/a/wisc.edu/local-government-property-insurance-fund

and a continuous covariate $\mathbf{x}^F$ (*LnCoverage*) - and two for the claim amount model - also a binary covariate $\mathbf{z}^S$ (*Fire5*: 1 or 0) and a continuous covariate $\mathbf{x}^S$ (*lnDeduct*). The two outcome - claim count and claim amount - variables are denoted as N_h and $\bar{Y}_h$ respectively. Hence, the format of this dataset is given by

$$
\begin{array}{lccc}
 & Year_1, & \cdots, & Year_y \\
\text{Policy } (h=1): & \{(N_1, \mathbf{X}_1^F, Y_{1(1)}, \cdots Y_{1(N_1)}, \mathbf{X}_1^S), & \cdots, & (N_1, \mathbf{X}_1^F, Y_{1(1)}, \cdots Y_{1(N_1)}, \mathbf{X}_1^S)\} \\
\text{Policy } (h=2): & \{(N_2, \mathbf{X}_2^F, Y_{2(1)}, \cdots Y_{2(N_2)}, \mathbf{X}_2^S), & \cdots, & (N_2, \mathbf{X}_2^F, Y_{2(1)}, \cdots Y_{2(N_2)}, \mathbf{X}_2^S)\} \\
 & \vdots & & \\
\text{Policy } (h=H): & \{(N_H, \mathbf{X}_H^F, Y_{H(1)}, \cdots Y_{H(N_H)}, \mathbf{X}_H^S), & \cdots, & (N_H, \mathbf{X}_H^F, Y_{H(1)}, \cdots Y_{H(N_H)}, \mathbf{X}_H^S)\}
\end{array}
$$

The experiment concerns predicting the aggregate claim amount $E[S_h|\mathbf{X}]$, based on the frequency-severity principle, for a given policy in order to develop risk premium. The prediction is stratified by six distinct entity types, which are *city, county, school, town, village*, and *miscellaneous*. These entity types signify the origin of the covered property, essentially grouping the policies into the six fixed risk clusters (i.e. $j = 1, \cdots, 6$). Thus, the different risk characteristics associated with each entity type are taken into account in the prediction. As detailed below, the continuous covariate $\mathbf{x}^S$ for the claim amount model is assumed to be subject to NDB error.

4.4.2 Implementation

Using the simulation data, we compare our hierarchical GLM-based NDB error correction with the SIMEX-based error correction that is discussed in Chapter 2. Both approaches aim to simultaneously address two types of model risk issues - heterogeneity (RQ1.1) and NDB covariates (RQ.2.2) - for risk premium development.

As discussed in Chapter 2, most error correction methods in the literature propose estimating the true covariate using gold standard data such as a subset of the true covariate available (Cook and Stefanski 1994; Freedman et al. 2004; Carroll et al. 2006). Their focus is on the discovery of the relationship between the mismea-

sured covariate and the true covariate by mapping the observed measurements to the true values offered by the gold standard data. In the real-world scenarios, however, the gold standard data is often unavailable. In this context, this thesis attempts to develop methodologies that effectively address measurement errors without relying on gold-standard data, using the prior knowledge we have discovered: $\tau_j^2 = (1-\zeta)\widehat{\lambda}_j^2$ from Section 4.3.2.

We formulate a simulation to study the overall relationship between the scaling factor ζ and the severity of NDB error, represented by the error rate R_{ϵ_x}, in the NDB covariate $\mathbf{x}^*$. This involves two key components: a) datasets considered as gold standards with no errors, and b) datasets with varying error rates under our control in the NDB covariate $\mathbf{x}^*$. By comparing models from error-free data with those derived from datasets containing varying levels of NDB error, we evaluate the effectiveness of Gustafson's correction method, based on a hierarchical GLM, with respect to the error rate R_{ϵ_x} and the scaling factor value $0 < \zeta < 1$. The optimal ζ then determines τ_j^2, enabling accurate adjustment toward the true parameter values using Gustafson's equations in Equation (4.23). The ultimate objective is to develop a guideline for selecting this optimal value of ζ, addressing the unknown variance $\tau_j^2 = (1-\zeta)\widehat{\lambda}_j^2$ for a given error rate R_{ϵ_x}. Once this rule of thumb is established, it can assist us in estimating the true covariate in the absence of gold-standard data.

Design of simulation data: Accordingly, we design simulation datasets with varying error rates R_{ϵ_x} based on real data from the LGPIF dataset while retaining the original LGPIF dataset as a gold standard. These simulation datasets are constructed by artificially introducing controlled NDB errors into the true covariate $\mathbf{x}$ at varying error rates R_{ϵ_x}. Our design of the NDB error generation for the covariate $\mathbf{x}^{S*}$, which creates variations in the error rate R_{ϵ_x}, is briefly illustrated in Figure 4.2. As outlined in Section 3.3.2, the NDB error $\epsilon_j \sim \mathbf{N}(0, \sigma_{j\epsilon}^2)$ is defined by its independence from the outcome and other covariates, while exhibiting correlation with the latent clusters. However, as noted in Equation (4.24), the Gustafson correction method posits that the variance of the linking component $V(\mathbf{x}^*|\mathbf{x})$ can be estimated

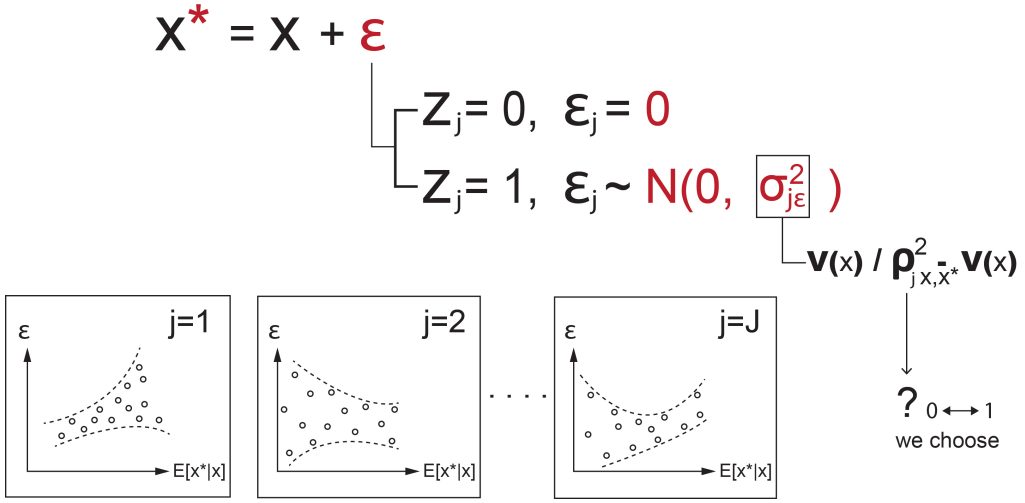

Figure 4.2: Design of Non-Differential Berkson (NDB) Error in $\mathbf{x}^*$ and the induced heteroscedasticity varying by cluster j

based on the variance of the covariate component $V(\mathbf{x}^*|\mathbf{z})$. To replicate this scenario, we relate the NDB covariate $\mathbf{x}^{S*}$ to the binary covariate $\mathbf{z}^S$ as specified in Equation (4.19b), by conditionally introducing NDB errors ϵ_j to the true covariate $\mathbf{x}^S$ only when $\mathbf{z}^S = 1$. In addition, as suggested by Hoffmann et al. 2017, we formulate the cluster-specific error variance $\sigma^2_{j\epsilon}$ for this NDB error generation using the random correlation $-1 < \rho_{j(x,x^*)} < 1$ between the true covariate $\mathbf{x}^S$ and the NDB covariate $\mathbf{x}^{S*}$ to replicate the structure of the NDB error. This approach adopts the NDB noise generation technique developed by Klau et al. 2021, given by

$$\sigma^2_{j\epsilon} = \frac{V(\mathbf{x}^S)}{\rho^2_{j(x^S,x^{S*})}} - V(\mathbf{x}^S) \tag{4.26}$$

Therefore, the selection of cluster-specific random correlations $\rho_{1(x^S,x^{S*})}, \cdots \rho_{J(x^S,x^{S*})}$ results in variations in the different error rates R_{ϵ_x} in the NDB covariate $\mathbf{x}^{S*}$. The different error rates in the NDB covariate $\mathbf{x}^{S*}$ are set at 1%, 10%, and 40% for the respective datasets.

$$R_{\epsilon_x} = \frac{\Sigma_{h=1}^H |x_h^{S*} - x_h^S|}{\Sigma_{h=1}^H x_h^S} : \begin{cases} 0.01\ (1\%) \text{ error rate in } \mathbf{x}^{S*} \text{ for dataset A.} \\ 0.1\ (10\%) \text{ error rate in } \mathbf{x}^{S*} \text{ for dataset B.} \\ 0.4\ (40\%) \text{ error rate in } \mathbf{x}^{S*} \text{ for dataset C.} \end{cases} \tag{4.27}$$

We generate three simulation datasets (based on the LGPIF dataset), each corresponding to one of the scenarios characterized by varying levels of error rates in the NDB covariate $\mathbf{x}^{S*}$.

Candidate models: As mentioned previously, we seek to identify the optimal scaling factor value ζ, which is contingent upon the error rate R_{ϵ_x} - $1\%, 10\%, 40\%$ - in the NDB covariate $\mathbf{x}^{S*}$. The performance of the hierarchical GLM, based on the optimal scaling factor value ζ, is then compared with that of the SIMEX method. Figure 4.3 depicts the development of four risk premium models for a comprehensive comparison of our Gustafson correction method and SIMEX within two analytical frameworks: the Bayesian hierarchical GLM and the conventional GLM. Each model, labeled (A) through (D), is constructed to systematically evaluate the

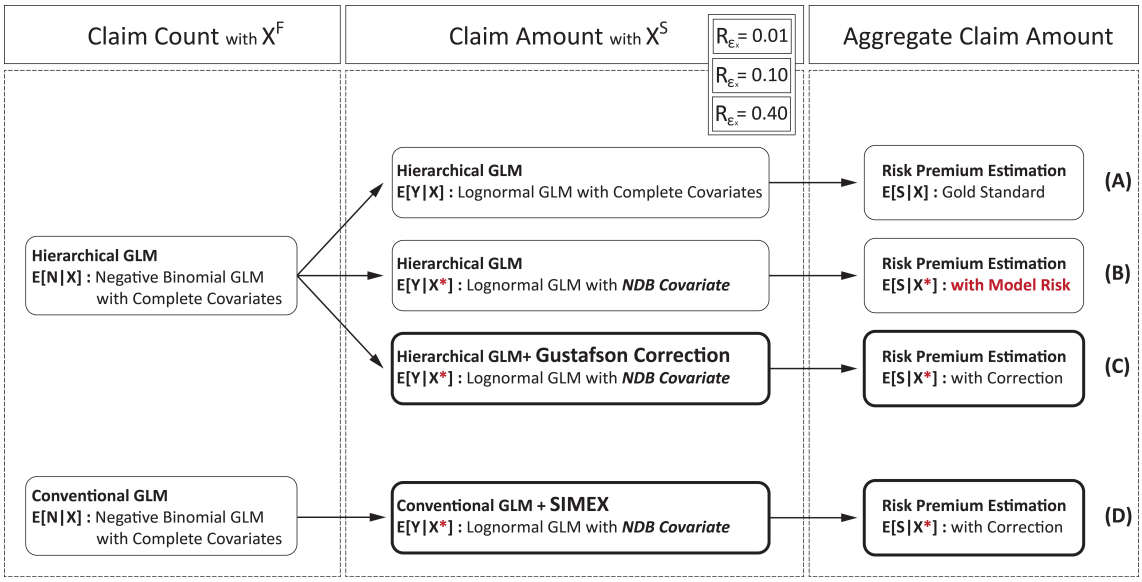

Figure 4.3: Four candidate models - (A) to (D) - for risk premium development. Specifically, Model(B),(C),(D) need to be thoroughly compared across various error rates R_{ϵ_x} - $1\%, 10\%, 40\%$ - in the NDB covariate $\mathbf{x}^*$.

impact of varying error rates in the NDB covariate $\mathbf{x}^{S*}$. Model(A) is built using the true covariates within the Bayesian hierarchical GLM framework and serve as a gold standard to establish a performance benchmark. It provides a reference point for evaluating the effectiveness of our correction methods. Model(B), on the other hand, is built to illustrate the adverse impact (model risk) of the NDB covariate

$\mathbf{x}^{S*}$ within the same Bayesian hierarchical GLM framework. It is important to thoroughly compare Model(B), as it is affected by model risk due to the NDB covariate, with Model(C), which applies Gustafson's correction, and Model(D), which uses the SIMEX correction. The objective of this comparison is to examine the effectiveness of these correction methods in mitigating the adverse impacts of NDB measurement errors in the covariate $\mathbf{x}^{S*}$.

Choice of hyperparameters: To run the Gibbs sampler for the hierarchical models - Model(A),(B),(C) - in Figure 4.3, flat priors have been chosen on the hyperparameters $\boldsymbol{\phi}$ **:** $\{m_0, \delta, q_0, \Lambda, \rho_{u1}, \rho_{u2}, \rho_{v1}, \rho_{v2}, c_0, d_0, g_0, h_0\}$ in Equation (4.11) through (4.14) and Equation (4.16) as

$$\begin{aligned}
\{\underline{m}_0 = \boldsymbol{\beta}^F_{GLM},\ m_0 = \boldsymbol{\beta}^S_{GLM},\ \underline{\delta} = 0.01,\ \delta = 0.01\} \qquad & \text{for } \boldsymbol{\beta}^F_0 \text{ and } \boldsymbol{\beta}^S_0 \\
\{\underline{q}_0 = p+2,\ q_0 = p+2,\ \underline{\Lambda} = \Sigma^F_{GLM},\ \Lambda = \Sigma^S_{GLM}\} \qquad & \text{for } \Sigma^F_{\beta_0} \text{ and } \Sigma^S_{\beta_0} \\
\{\underline{\rho}_{u1} = 0.125,\ \rho_{u1} = 0.125,\ \underline{\rho}_{u2} = 1.5,\ \rho_{u2} = 1.5\} \qquad & \text{for } u^F_0 \text{ and } u^S_0 \\
\{\underline{\rho}_{v1} = 8,\ \rho_{v1} = 8,\ \underline{\rho}_{v2} = 1,\ \rho_{v2} = 1\} \qquad & \text{for } v^F_0 \text{ and } v^S_0 \\
\{c^F_0 = 0.5,\ c^S_0 = 0.5,\ d^F_0 = 0.5,\ d^S_0 = 0.5\} \qquad & \text{for } \lambda^{2F} \text{ and } \lambda^{2S} \\
\{g^F_0 = 0.5,\ g^S_0 = 0.5,\ h^F_0 = 0.5,\ h^S_0 = 0.5\} \qquad & \text{for } \pi^F \text{ and } \pi^S
\end{aligned} \tag{4.28}$$

where $\boldsymbol{\beta}^F_{GLM}, \boldsymbol{\beta}^S_{GLM}$ represent the vectors of the GLM coefficients, which are utilized as starting values for the Gibbs sampler, p denotes the dimension of the covariates in the model, and $\Sigma^F_{GLM}, \Sigma^S_{GLM}$ corresponds to the variance-covariance matrices derived from the GLM results. The specific values in Equation (4.28) have been carefully selected based on preliminary results obtained from pre-running the Gibbs sampler initialized with random values. Although starting the Gibbs sampler with incorrect initial values may not yield efficient posterior samples, it can generate a few posterior samples that provide insights into the true behavior of the posterior distribution. By leveraging the Method of Moments technique[5], we can speculate

[5]The method of moments is a statistical technique that estimates population parameters by equating sample moments (e.g., mean, variance) to theoretical moments derived from a probability

about the posterior samples' characteristics and use this information to inform our choice of starting values, thereby enhancing the efficiency of the Gibbs sampler.

4.4.3 Results with LGPIF ($H = 1,679$)

We construct a training set with 1,276 records for the modeling procedure and a test set with 403 records for the prediction validation. As mentioned in Section 4.4.2 and illustrated in Figure 4.3, Model(A) is a gold standard, Model(B) represents an erroneous model reflecting real-life model risk, Model(C) incorporates the Gustafson correction, and Model(D) uses the SIMEX correction within the traditional GLM framework. We systematically compare the modeling performances - Model(B), (C), and (D) - across three distinct scenarios characterized by different error rates $R_{\epsilon_x} = 0.01, R_{\epsilon_x} = 0.1$, and $R_{\epsilon_x} = 0.4$ in the NDB covariate $\mathbf{x}^{S*}$. To devise a practical guideline for selecting the optimal scaling factor with the Gustafson correction in the absence of gold standard data, we examine the impact of the scaling factor $0 < \zeta < 1$ in Model(C) in each scenario. This effort ultimately aims to establish an aggregate loss $E[S_h|\mathbf{X}^F, \mathbf{X}^{S*}]$ prediction framework in Equation (4.16) for risk premium development, mitigating the model risk outlined in research questions RQ1.1 and RQ2.2.

For each hierarchical GLM model - Model(A), (B), (C) in Figure 4.3 - we ran Markov chains to estimate the parameters, each undergoing $M = 60,000$ iterations of Gibbs sampling based on Algorithm (D.1). Given that Model(A) serves as a gold standard, we focused on Model(B), (C), and (D) to compare their performance under varying error rates. These models were fitted to three datasets with different error rates $R_{\epsilon_x} = 0.01, R_{\epsilon_x} = 0.1$, and $R_{\epsilon_x} = 0.4$ in the NDB covariate $\mathbf{x}^{S*}$. For Model(C), the Gustafson correction was performed by testing a series of the scaling factors: $\zeta = \{0.1, \cdots, 0.9\}$.

Within each Gibbs sampling iteration in Model(A), (B), and (C), a Metropolis-Hastings (MH) technique was embedded to update the outcome parameters - $\boldsymbol{\beta}_j^F, \psi_j, \boldsymbol{\beta}_j^S, \sigma_j^2$

distribution (Pearson 1936).

- due to the lack of conjugate priors in relation to their outcome data models. The initial 10,000 iterations were designated as a burn-in period to allow the chains to reach a stable state and were subsequently discarded. Convergence of the chains was verified using the Brooks-Gelman statistic (Brooks and Gelman 1998), ensuring that the chains had sufficiently mixed. For the conventional rival GLM models - Model(D) in Figure 4.3 -, the traditional Maximum Likelihood Estimation algorithm (MLE) is used for the parameter inference.

Gold standard [Model(A)]: To begin with, the gold standard - Model(A) in Figure 4.3 - developed upon the hierarchical GLM framework is examined and Figure 4.4, 4.5, 4.6, 4.7 summarize the modeling results. As the error-free benchmark model, its predictive distribution $f(S|\mathbf{X}^F, \mathbf{X}^S)$ is obtained using the true covariates. The two fitted models, including the hierarchical negative binomial GLM for $\xi_h = E[N_h|\mathbf{X}^F]$ and the hierarchical log-normal GLM for $\mu_h = \log\left(E[\bar{Y}_h|\mathbf{X}^S]\right) - \frac{1}{2}\sigma_j^2$, together form the basis for predicting the expected total aggregate claim amount $E[S|\mathbf{X}^F, \mathbf{X}^S]$.

Figure 4.4 and 4.5 summarize the modeling results for the claim count component $N_h|\mathbf{X}^F \sim \mathbf{NB}(\xi_h = E[N_h|\mathbf{X}^F],\ \psi_j)$ with fixed clusters $j = 1, \cdots, 6$ (corresponding to six distinct entity types: *city, county, school, town, village*, and *miscellaneous*). Figure 4.4 displays the convergence of the Gibbs sampling - (b) - and the estimated posterior behavior of the dispersion ψ_j based on 60,000 draws from the Gibbs sampler - (a). This indicates that the posterior samples are reliable, and the Gamma distributed ψ_j, representing the variability in claim counts N_h, has mean estimates ranging from 0.23 to 0.26 across the different clusters $j = 1, \cdots, 6$, reflecting a consistent spread of N_h. The cluster-wise predictive distributions for the claim counts $f(N_h|\mathbf{X}^F, \xi_h, \psi)$ obtained from the Gibbs sampling are exhibited in Figure 4.5, which indicate the consistent predictive abilities of the fitted models.

Figure 4.6 and 4.7 provide an overview of the modeling results for the claim amount component $\bar{Y}_h|\mathbf{X}^S \sim \mathbf{LogN}(\mu_h = \log\left(E[\bar{Y}_h|\mathbf{X}^S]\right) - \frac{1}{2}\sigma_j^2,\ \sigma_j^2)$ with $j = 1, \cdots, 6$. Figure 4.6 portrays the convergence of the Gibbs sampling - (b) - and

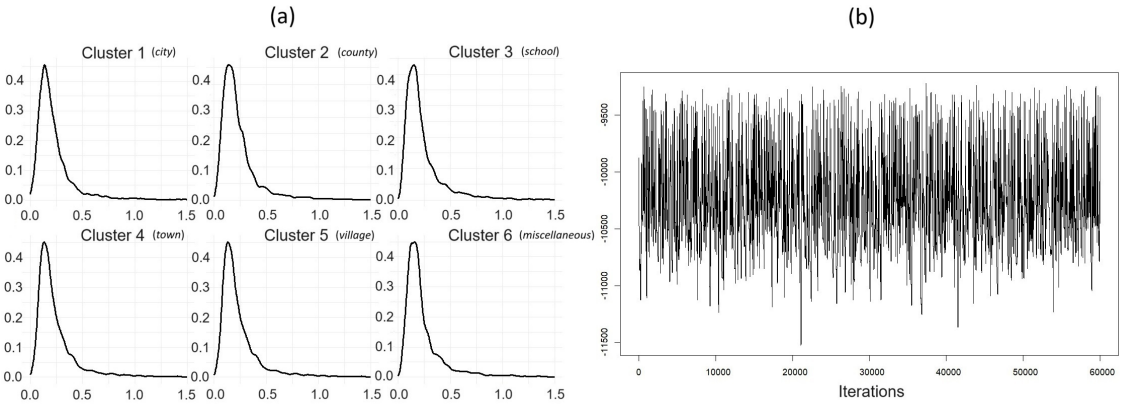

Figure 4.4: Model(A) Result I: Estimated posterior densities of the dispersion parameter ψ_j for $j = 1, \cdots 6$ (a), and MCMC trace plot with 60,000 iterations based on the log-likelihood of the hierarchical negative binomial GLM (b).

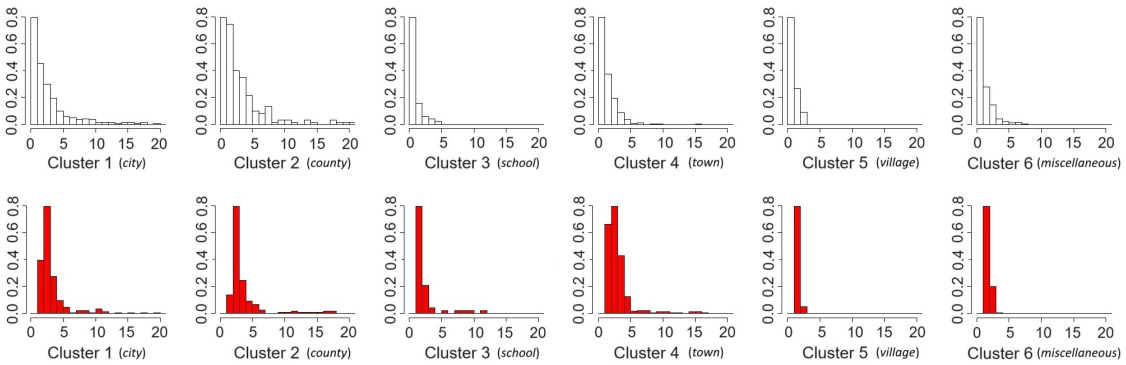

Figure 4.5: Model(A) Result II: The observed distribution of the claim count N_h (white), and the predictive densities for $N_h|\mathbf{X}^F$ across clusters $j = 1, \cdots 6$ (red).

the estimated posterior behavior of the variance of $\ln \bar{Y}_h$ (i.e. σ_j^2) based on 60,000 draws from the Gibbs sampler - (a). This demonstrates the reliability of the posterior samples, with the Inverse Gamma-distributed σ_j^2 accurately capturing the variability in claim amounts $\bar{Y}_h$ on a log scale. The mean estimates of σ_j^2 range from 3.2 to 3.5 across the different clusters $j = 1, \cdots, 6$, presenting a consistent deviation from the mean of $\ln \bar{Y}_h$. The alignment of cluster-wise predictive distributions for the claim amount $f(\ln \bar{Y}_h|\mathbf{X}^S, \mu_h, \sigma^2)$ with the observed data, as depicted in Figure 4.7, reveals that the predictive distributions show tailored fits for each cluster, capturing the distinct characteristics inherent in the respective data segments.

The aggregate claim amount model $f(S_h|\mathbf{X}^F, \mathbf{X}^S, \xi_h, \psi, \mu_h, \sigma^2)$ is derived by integrating these two fitted models: the estimated claim count model $f(N_h|\mathbf{X}^F, \xi_h, \psi)$

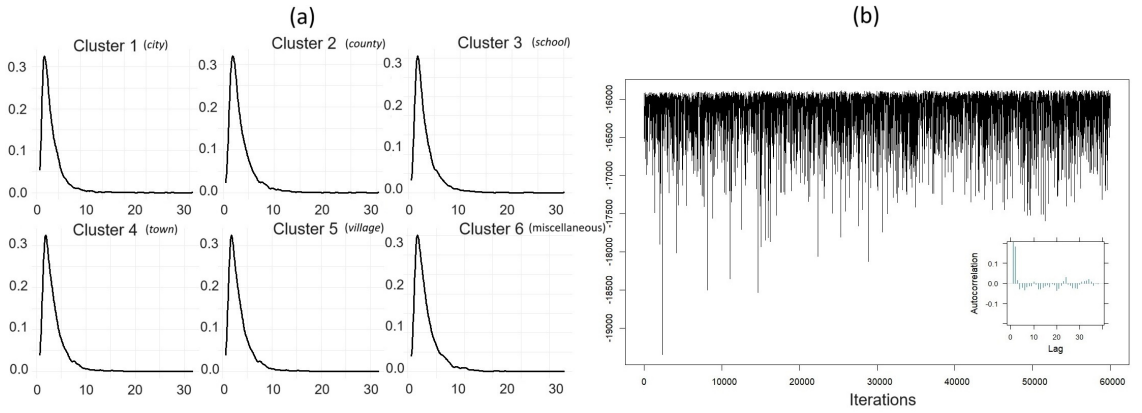

Figure 4.6: Model(A) Result III: Estimated posterior densities of the scale parameter σ_j^2 for $j = 1, \cdots 6$ (a), and MCMC trace plot with 60,000 iterations based on the log-likelihood of the hierarchical log-normal GLM (b).

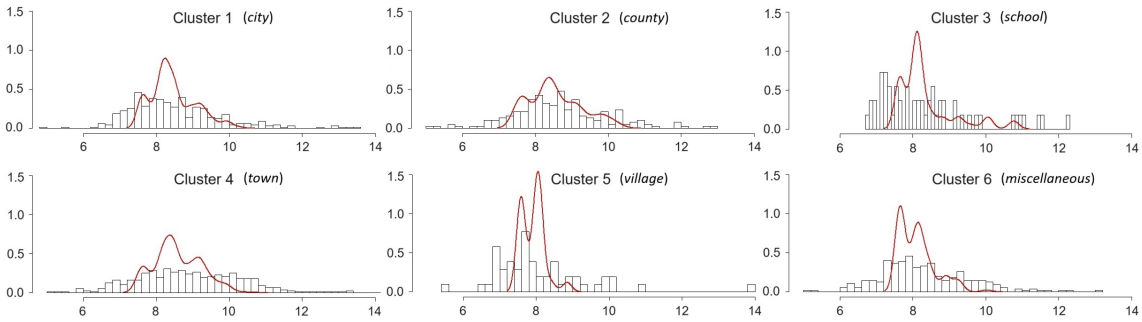

Figure 4.7: Model(A) Result IV: The observed distribution of the claim amount on a log scale $\ln \bar{Y}_h$ (white histogram), and the predictive densities for $\log \bar{Y}_h | \mathbf{X}^S$ across clusters $j = 1, \cdots 6$ (red curve).

and the estimated claim amount model $f(\log \bar{Y}_h | \mathbf{X}^S, \mu_h, \sigma^2)$, using Monte Carlo simulation. First, a claim count sample N_h for a policy h is drawn from $f(N_h | \mathbf{X}^F, \xi_h, \psi)$. Then, N_h claim amount samples are drawn from $f(\log \bar{Y}_h | \mathbf{X}^S, \mu_h, \sigma^2)$. These steps are repeated multiple times, allowing for constructing the distribution of aggregate claims S_h for a policy h (Scollnik 2001). Figure 4.8 presents a comparison of the resulting shapes of the cluster-wise distributions of the aggregate claim amount S_h on a log scale with the overall distribution of S_h, also on a log scale. This comparison aids in assessing the risk profile and contribution of each cluster relative to the overall aggregate claim. Notably, the distribution for Cluster 2 aligns closely with the overall distribution, suggesting that this cluster is likely contributing more significantly to the overall risk premium development.

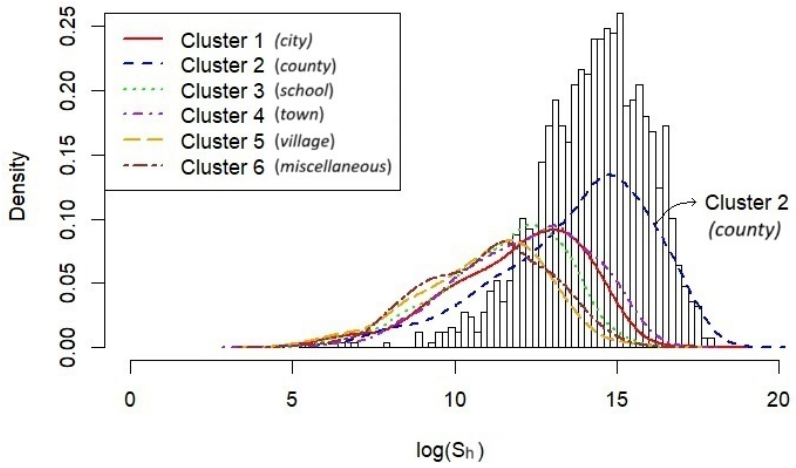

Figure 4.8: Model(A) Result V: A histogram of the overall expected aggregate claim amount on a log scale, overlaid with the individual cluster-wise distributions $logS_h|\mathbf{X}^F, \mathbf{X}^S$.

Gustafson correction [Model(B) vs Model(C) vs Model(D)]: As elaborated in Figure 4.3, our goal is to identify which of the candidate models, Models(C) and (D), produces results that are closest to the established gold standard, Model(A). Specifically, the results of Model(C) based on our Gustafson correction technique are compared with those of Model(D) based on SIMEX correction. Models(A), (B), and (C) are developed within the Bayesian hierarchical GLM framework, while Model(D) is built within the conventional GLM framework, offering a comparison from a frequentist viewpoint.

Given that only the estimation of the claim amount $\bar{Y}_h$ is subject to the model risk stemming from the NDB covariate $\mathbf{x}^{S*}$, this section focuses solely on the results of the claim amount component (hierarchical log-normal GLM). Specifically, we examine the claim amount model represented by $\bar{Y}_h|\mathbf{X}^S \sim \mathbf{LogN}(\mu_h = \log\left(E[\bar{Y}_h|\mathbf{X}^S]\right) - \frac{1}{2}\sigma_j^2,\ \sigma_j^2)$ in Model(B), (C), and (D). Additionally, a sensitivity analysis for the claim amount model has also been conducted. It seeks to identify the optimal scaling factor value ζ by leveraging the prior knowledge established in Section 4.3.2, where $\tau_j^2 = (1-\zeta)\widehat{\lambda}_j^2$, as well as the relationships between the covariates $\mathbf{X}^S : \{\mathbf{x}^S, \mathbf{z}^S\}$ and the claim amounts $\bar{Y}_h$.

Table 4.1 compares the marginal posterior means of the scale parameter σ_j^2 ob-

tained from three different models: the gold standard hierarchical log-normal GLM in Model(A), the hierarchical log-normal GLM incorporating the NDB covariate $\mathbf{x}^{S*}$ in Model(B), and the hierarchical log-normal GLM with the Gustafson correction in Model(C). Each GLM is based on the outcome model: $\bar{Y}_h|\mathbf{X}^S \sim \mathbf{LogN}(\mu_h = ln\big(E[\bar{Y}_h|\mathbf{X}^S]\big) - \frac{1}{2}\sigma_j^2,\ \sigma_j^2)$. Table 4.2 and 4.3 provide additional comparisons between the marginal posterior means of the GLM coefficients $\boldsymbol{\beta}_j$. These comparisons also highlight the sensitivity analysis results to examine the optimal scaling factor ζ.

Model	Scale	Parameter estimates σ_j^2								
Model(A) Gold standard	$\sigma^2_{j=1}$	3.35 with 95% Credible Interval: $\{1.04 \leq \sigma^2_{j=1} \leq 9.85\}$								
	$\sigma^2_{j=2}$	3.44 with 95% Credible Interval: $\{1.05 \leq \sigma^2_{j=2} \leq 10.01\}$								
	$\sigma^2_{j=3}$	3.22 with 95% Credible Interval: $\{1.02 \leq \sigma^2_{j=3} \leq 9.48\}$								
	$\sigma^2_{j=4}$	3.40 with 95% Credible Interval: $\{1.04 \leq \sigma^2_{j=4} \leq 9.78\}$								
	$\sigma^2_{j=5}$	3.40 with 95% Credible Interval: $\{1.23 \leq \sigma^2_{j=5} \leq 10.13\}$								
	$\sigma^2_{j=6}$	3.23 with 95% Credible Interval: $\{1.01 \leq \sigma^2_{j=6} \leq 9.65\}$								
		Error rate R_{ϵ_x} in $\mathbf{x}^*$: 0.01			Error rate R_{ϵ_x} in $\mathbf{x}^*$: 0.10			Error rate R_{ϵ_x} in $\mathbf{x}^*$: 0.40		
Model(B) Before correction	$\sigma^2_{j=1}$	3.36			3.38			3.59		
	$\sigma^2_{j=2}$	3.44			3.51			3.65		
	$\sigma^2_{j=3}$	3.26			3.38			3.46		
	$\sigma^2_{j=4}$	3.43			3.58			3.67		
	$\sigma^2_{j=5}$	3.44			3.57			3.65		
	$\sigma^2_{j=6}$	3.28			3.45			3.51		
		ζ : 0.5	ζ : 0.6	ζ : 0.7	ζ : 0.5	ζ : 0.6	ζ : 0.7	ζ : 0.5	ζ : 0.6	ζ : 0.7
Model(C) After correction	$\sigma^2_{j=1}$	3.28	3.30	3.33	3.22	3.32	3.56	3.36	3.34	3.31
	$\sigma^2_{j=2}$	3.37	3.38	3.41	3.29	3.46	3.48	3.46	3.45	3.33
	$\sigma^2_{j=3}$	3.19	3.22	3.34	3.26	3.26	3.57	3.24	3.23	3.21
	$\sigma^2_{j=4}$	3.27	3.31	3.48	3.31	3.35	3.61	3.41	3.38	3.36
	$\sigma^2_{j=5}$	3.10	3.29	3.49	3.25	3.35	3.57	3.42	3.38	3.35
	$\sigma^2_{j=6}$	3.11	3.29	3.33	3.39	3.32	3.57	3.26	3.25	3.19
LPPD ($\times 10^3$)		-16.79	**-16.77**	-16.78	-16.41	**-16.39**	-16.41	**-16.19**	-16.20	-16.21

Table 4.1: Comparison of the scale parameter σ_j^2 estimates from the hierarchical log-normal GLMs in Model(A),(B), and (C) across risk clusters $j = 1, \cdots, 6$

We begin by examining the erroneous model in Model(B), which is constructed using the NDB covariate $\mathbf{x}^{S*}$ thereby introducing model risk. Our focus is on examining how the parameter estimates, including the scale parameter σ^2 and the GLM coefficients $\boldsymbol{\beta}$, are influenced by varying error rates R_{ϵ_x} (i.e., varying degree of model risk). We seek to determine how these parameter estimates generated

from the erroneous model in Model(B) deviate from the results obtained in the gold standard model in Model(A). In Table 4.1, as well as Tables 4.2 and 4.3, the parameter estimates from Model(B) exhibit a consistent pattern, showing a monotonically increasing trend in response to increasing error rates ($R_{\epsilon_x} : 0.01 \rightarrow R_{\epsilon_x} : 0.40$) in the NDB covariate $\mathbf{x}^{S*}$. To be specific, as the error rate R_{ϵ_x} increases, both the scale parameter σ^2 and the GLM coefficients β_0, β_2, which represent the intercept and the binary covariate $\mathbf{z}^S$ respectively, tend to show an inflationary trend across all six clusters $j = 1, \cdots, 6$. Conversely, the GLM coefficient β_1, which directly corresponds to the NDB covariate $\mathbf{x}^{S*}$, exhibits a deflationary pattern across all six clusters $j = 1, \cdots, 6$.

Model	Intercept	Parameter estimates β_{j0}								
Model(A) Gold standard	$\beta 0_{j=1}$	7.18 with 95% Credible Interval: $\{4.98 \leq \beta 0_{j=1} \leq 9.32\}$								
	$\beta 0_{j=2}$	7.69 with 95% Credible Interval: $\{5.50 \leq \beta 0_{j=2} \leq 9.71\}$								
	$\beta 0_{j=3}$	6.37 with 95% Credible Interval: $\{4.11 \leq \beta 0_{j=3} \leq 8.86\}$								
	$\beta 0_{j=4}$	7.36 with 95% Credible Interval: $\{3.83 \leq \beta 0_{j=4} \leq 10.52\}$								
	$\beta 0_{j=5}$	7.14 with 95% Credible Interval: $\{3.83 \leq \beta 0_{j=5} \leq 10.32\}$								
	$\beta 0_{j=6}$	7.36 with 95% Credible Interval: $\{4.87 \leq \beta 0_{j=6} \leq 10.46\}$								
		Error rate R_{ϵ_x} in $\mathbf{x}^*$: 0.01			Error rate R_{ϵ_x} in $\mathbf{x}^*$: 0.10			Error rate R_{ϵ_x} in $\mathbf{x}^*$: 0.40		
Model(B) Before correction	$\beta 0_{j=1}$	7.28			8.89			9.40		
	$\beta 0_{j=2}$	7.72			9.14			9.51		
	$\beta 0_{j=3}$	6.44			8.82			9.28		
	$\beta 0_{j=4}$	7.33			9.28			9.52		
	$\beta 0_{j=5}$	7.12			9.37			9.57		
	$\beta 0_{j=6}$	7.46			9.11			9.72		
		$\zeta : 0.5$	$\zeta : 0.6$	$\zeta : 0.7$	$\zeta : 0.5$	$\zeta : 0.6$	$\zeta : 0.7$	$\zeta : 0.5$	$\zeta : 0.6$	$\zeta : 0.7$
Model(C) After correction	$\beta 0_{j=1}$	6.08	7.24	7.29	6.86	7.28	8.03	7.21	7.09	6.81
	$\beta 0_{j=2}$	7.35	8.07	8.21	7.37	8.12	8.43	7.56	7.39	7.24
	$\beta 0_{j=3}$	5.20	6.07	6.21	5.93	6.19	7.25	6.35	6.11	5.87
	$\beta 0_{j=4}$	6.40	7.33	7.87	6.91	7.53	8.16	7.39	7.14	6.73
	$\beta 0_{j=5}$	6.52	6.95	7.20	6.23	6.94	7.28	7.18	7.15	7.11
	$\beta 0_{j=6}$	6.20	7.40	7.48	6.87	7.27	7.67	7.39	7.15	6.42
LPPD ($\times 10^3$)		-16.79	**-16.77**	-16.78	-16.41	**-16.39**	-16.41	**-16.19**	-16.20	-16.21

Table 4.2: Comparison of the GLM intercept $\beta 0_j$ estimates from the hierarchical log-normal GLMs (claim amount component) in Model(A),(B), and (C) across risk clusters $j = 1, \cdots, 6$

Model	Slope	Parameter estimates $\beta_{j1}; \beta_{j2}$								
Model(A) Gold standard	$\beta 1_{j=1}$	0.31 with 95% Credible Interval: $\{0.03 \leq \beta 1_{j=1} \leq 0.62\}$								
	$\beta 1_{j=2}$	0.24 with 95% Credible Interval: $\{-0.04 \leq \beta 1_{j=2} \leq 0.59\}$								
	$\beta 1_{j=3}$	0.43 with 95% Credible Interval: $\{0.09 \leq \beta 1_{j=3} \leq 0.67\}$								
	$\beta 1_{j=4}$	0.33 with 95% Credible Interval: $\{-0.10 \leq \beta 1_{j=4} \leq 0.84\}$								
	$\beta 1_{j=5}$	0.34 with 95% Credible Interval: $\{-0.12 \leq \beta 1_{j=5} \leq 0.89\}$								
	$\beta 1_{j=6}$	0.29 with 95% Credible Interval: $\{-0.08 \leq \beta 1_{j=6} \leq 0.67\}$								
		Error rate R_{ϵ_x} in $\mathbf{x}^*$: 0.01			Error rate R_{ϵ_x} in $\mathbf{x}^*$: 0.10			Error rate R_{ϵ_x} in $\mathbf{x}^*$: 0.40		
Model(B) Before correction	$\beta 1_{j=1}$	0.29			0.06			0.01		
	$\beta 1_{j=2}$	0.24			0.04			0.01		
	$\beta 1_{j=3}$	0.42			0.07			0.01		
	$\beta 1_{j=4}$	0.32			0.07			0.01		
	$\beta 1_{j=5}$	0.34			0.06			0.01		
	$\beta 1_{j=6}$	0.28			0.06			0.01		
		$\zeta : 0.5$	$\zeta : 0.6$	$\zeta : 0.7$	$\zeta : 0.5$	$\zeta : 0.6$	$\zeta : 0.7$	$\zeta : 0.5$	$\zeta : 0.6$	$\zeta : 0.7$
Model(C) After correction	$\beta 1_{j=1}$	0.56	0.38	0.21	0.38	0.36	0.27	0.34	0.38	0.42
	$\beta 1_{j=2}$	0.38	0.26	0.11	0.34	0.27	0.20	0.21	0.33	0.38
	$\beta 1_{j=3}$	0.83	0.56	0.32	0.52	0.48	0.39	0.43	0.47	0.51
	$\beta 1_{j=4}$	0.47	0.32	0.28	0.41	0.35	0.26	0.37	0.42	0.48
	$\beta 1_{j=5}$	0.81	0.41	0.29	0.34	0.31	0.28	0.36	0.39	0.45
	$\beta 1_{j=6}$	0.55	0.40	0.31	0.31	0.26	0.19	0.31	0.34	0.38

Model(A) Gold standard	$\beta 2_{j=1}$	0.24 with 95% Credible Interval: $\{-0.55 \leq \beta 2_{j=1} \leq 1.01\}$								
	$\beta 2_{j=2}$	0.21 with 95% Credible Interval: $\{-0.47 \leq \beta 2_{j=2} \leq 0.82\}$								
	$\beta 2_{j=3}$	0.12 with 95% Credible Interval: $\{-0.67 \leq \beta 2_{j=3} \leq 0.94\}$								
	$\beta 2_{j=4}$	0.08 with 95% Credible Interval: $\{-1.06 \leq \beta 2_{j=4} \leq 0.59\}$								
	$\beta 2_{j=5}$	-0.12 with 95% Credible Interval: $\{-1.05 \leq \beta 2_{j=5} \leq 0.60\}$								
	$\beta 2_{j=6}$	0.17 with 95% Credible Interval: $\{-0.63 \leq \beta 2_{j=6} \leq 0.67\}$								
		Error rate R_{ϵ_x} in $\mathbf{x}^*$: 0.01			Error rate R_{ϵ_x} in $\mathbf{x}^*$: 0.10			Error rate R_{ϵ_x} in $\mathbf{x}^*$: 0.40		
Model(B) Before correction	$\beta 2_{j=1}$	0.27			0.31			0.38		
	$\beta 2_{j=2}$	0.24			0.34			0.40		
	$\beta 2_{j=3}$	0.13			0.32			0.40		
	$\beta 2_{j=4}$	0.09			0.17			0.22		
	$\beta 2_{j=5}$	0.10			0.12			0.17		
	$\beta 2_{j=6}$	0.12			0.23			0.36		
		$\zeta : 0.5$	$\zeta : 0.6$	$\zeta : 0.7$	$\zeta : 0.5$	$\zeta : 0.6$	$\zeta : 0.7$	$\zeta : 0.5$	$\zeta : 0.6$	$\zeta : 0.7$
Model(C) After correction	$\beta 2_{j=1}$	0.17	0.26	0.29	0.19	0.28	0.31	0.25	0.23	0.19
	$\beta 2_{j=2}$	0.23	0.27	0.31	0.18	0.23	0.28	0.24	0.21	0.18
	$\beta 2_{j=3}$	0.07	0.11	0.19	0.07	0.12	0.17	0.19	0.17	0.15
	$\beta 2_{j=4}$	0.01	0.03	0.11	0.11	0.13	0.18	0.10	0.07	0.02
	$\beta 2_{j=5}$	-0.11	-0.14	0.16	-0.21	-0.15	-0.07	-0.16	-0.19	-0.21
	$\beta 2_{j=6}$	0.12	0.19	0.21	0.08	0.17	0.21	0.16	0.12	0.09
LPPD ($\times 10^3$)		-16.79	**-16.77**	-16.78	-16.41	**-16.39**	-16.41	**-16.19**	-16.20	-16.21

Table 4.3: Comparison of the GLM slope $\beta 1_j, \beta 2_j$ estimates from the hierarchical log-normal GLMs (claim amount component) in Model(A),(B), and (C) across risk clusters $j = 1, \cdots, 6$.

This behavior is consistent with the notion that as the error rate R_{ϵ_x} grows, the increased noise levels lead to greater variance in the outcomes, and thus the scale parameter σ^2 increases proportionally. The consistent directional change in the coefficient β_1 associated with the NDB covariate $\mathbf{x}^{S*}$ suggests that the noise in $\mathbf{x}^{S*}$ has a systematic pattern of the NDB error.

Shifting our focus to the Gustafson correction model, Model(C), we notice that as the error rate R_{ϵ_x} rises, the estimate results from Model(C) increasingly resemble those of the gold standard, Model(A), which is a promising outcome for our research. As observed in Table 4.1, 4.2 and 4.3, there is a specific range of scaling factors, $0.5 \leq \zeta \leq 0.7$, where Model(C) performs best in this experiment. Within this range, the scale parameter σ^2 and GLM coefficients $\beta_0, \beta_1, \beta_2$ across all six clusters show the strongest alignment with the gold standard model, Model(A), consistently falling within the credible intervals produced by that model. This implies that the connection between the conditional error variance terms - $V(\mathbf{x}^*|\mathbf{x})$ and $V(\mathbf{x}^*|\mathbf{z})$ - is largely captured in this window of ζ, as indicated by the prior knowledge: $\tau_j^2 = (1-\zeta)\widehat{\lambda}_j^2$ in Equation (4.24).

The challenge, however, lies in identifying the optimal range of ζ values for a specific error rate R_{ϵ_x} in the covariate $\mathbf{x}^*$. Our experiments show that parameter estimates corrected by Model(C) often perform worse than the erroneous model, Model(B), when corrections are made with values outside the $0.5 \leq \zeta \leq 0.7$ range. This worsens as the error rate R_{ϵ_x} increases, which is expected, because a higher error rate implies greater bias, making it much harder for the model to correct. Throughout our experiments, however, we found that determining the optimal range of ζ can be effectively searched by evaluating the LPPD for each modeling result, as the LPPD reflects the degree of predictive performance, which is translated into how closely the estimated parameter values align with the gold standard.

Our analysis suggests a rule of thumb: "The optimal ζ maximizes LPPD." Identifying the modeling results with the highest LPPD allows us to systematically determine the optimal ζ, enhancing correction performance without requiring access

to gold-standard data.

Our experimental results on the selection of the optimal scaling factor ζ using the LGPIF dataset are further analyzed in Figure 4.9 and Table 4.4 for the error rate $R_{\epsilon_x} = 0.01$, Figure 4.10 and Table 4.5 for the error rate $R_{\epsilon_x} = 0.10$, and Figure 4.11 and Table 4.6 for the error rate $R_{\epsilon_x} = 0.40$. To compare their estimates, Tables 4.4, 4.5, 4.6 display (i) LPPD, (ii) SSPE, (iii) SAPE, and (iv) Kullback-Leibler Divergence (D_{KL}) for the individual claim amount model $f(\bar{Y}_h|\mathbf{X}^S)$ as well as CTEs for the aggregate claim amount model $f(S_h|\mathbf{X}^F, \mathbf{X}^S)$ within each scenario. For comparison, we also include the results from the SIMEX correction. In Figures 4.9,4.10,and 4.11, the histograms of the testing set and the out-of-sample prediction curves from Model(C) are compared against the gold standard prediction curves from Model(A) and the erroneous prediction curves from Model(B). The prediction curve for Model(D) is omitted due to its subpar performance.

As expected, Tables 4.4, 4.5, 4.6 demonstrate that the gold standard model in Model(A) achieves the greatest LPPD value of $-16,155.90$, while the naïve model with the model risk in Model(B) shows the lowest LPPD values across all datasets with different error rates. Note that the LPPD for the GLM-based SIMEX is not available, as LPPD calculations rely on posterior densities. Applying the Gustafson corrections in Model(C) yields LPPD values of $-16,770.85$ with $\zeta = 0.6$ for the error rate 0.01, $-16,370.44$ with $\zeta = 0.6$ for the error rate 0.10, and -16188.31 with $\zeta = 0.5$ for the error rate 0.40, closely matching the LPPD value observed in the error-free gold standard model in Model(A). This result is consistent with other metrics, such as SSPE and SAPE, across various scenarios with different error rates. The naïve model with model risk in Model(B) consistently shows the highest SSPE and SAPE values, indicating poorer predictive performance due to the model risk. In contrast, the Gustafson correction model in Model(C) yields significantly lower SSPE and SAPE values, closely approximating those of the gold standard model in Model(A). Furthermore, as the error rate increases, the Gustafson correction in Model(C) increasingly outperforms Model(D) (SIMEX), showing its robustness.

$\zeta = 0.6$ $R_{\epsilon_x} = 0.01$	Feature	Model(A): Gold standard	Model(B): with Model Risk	Model(C) Gustafson correction	Model(D): SIMEX correction
$f(ln\bar{Y}_h \mid \mathbf{X}^S)$	LPPD	-16,155.90	-17,437.59	**-16,770.85**	-
	SSPE	784.52	817.16	798.89	816.80
	SAPE	415.21	422.53	419.83	421.60
	D_{KL}	0.00	1.28	0.61	-
$f(lnS_h \mid \mathbf{X}^F, \mathbf{X}^S)$	CTE 10%	48,782.40	42,995.64	49,599.75	45,237.36
	CTE 50%	81,593.58	89,103.11	82,218.10	78,736.26
	CTE 90%	209,761.38	226,599.19	196,273.16	121,342.20
	CTE 95%	274,996.31	260,859.33	269,656.52	170,487.20

Table 4.4: Comparison of predictive performances among three Bayesian hierarchical GLMs—Model (A), (B), and (C)—and the GLM-based SIMEX, based on the LGPIF data with a covariate error rate of $R_{\epsilon_x} = 0.01$ and a scaling factor of $\zeta = 0.6$.

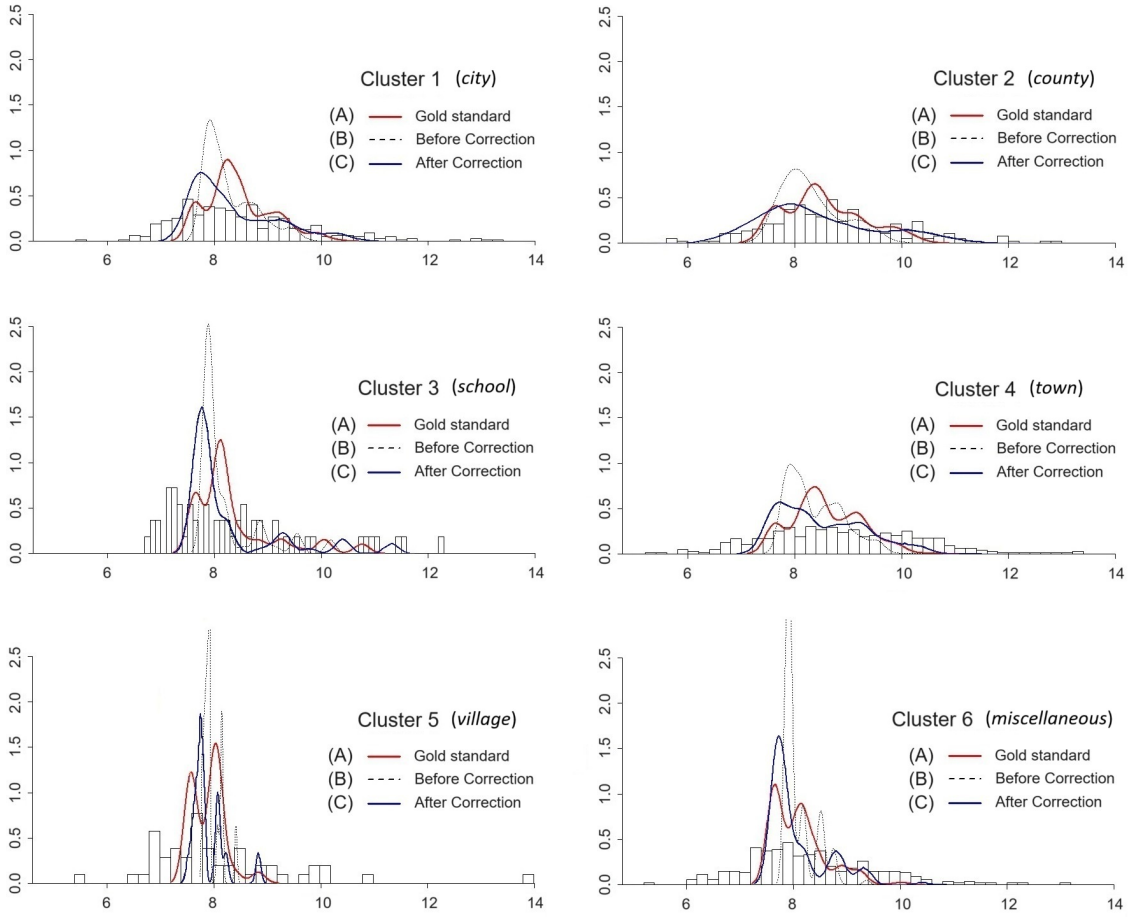

Figure 4.9: Fitted models based on the LGPIF data with the error rate $R_{\epsilon_x} = 0.01$ and the scaling factor $\zeta = 0.6$: Cluster-wise histograms (for $j = 1, \cdots 6$) of the observed claim amount Y_h on a log scale and the out-of-sample predictive densities obtained from Model(A), (B), and (C)

$\zeta = 0.6$ $R_{\epsilon_x} = 0.10$	Feature	Model(A): Gold standard	Model(B): with Model Risk	Model(C) Gustafson correction	Model(D): SIMEX correction
$f(ln\bar{Y}_h \mid \mathbf{X}^S)$	LPPD	-16,155.90	-17,731.03	**-16,390.44**	-
	SSPE	784.52	840.02	795.07	892.72
	SAPE	415.21	430.22	418.68	439.92
	D_{KL}	0.00	1.57	0.41	-
$f(lnS_h \mid \mathbf{X}^F, \mathbf{X}^S)$	CTE 10%	48,782.40	50,222.22	49,764.81	45,846.51
	CTE 50%	81,593.58	84,118.93	83,595.88	73,558.45
	CTE 90%	209,761.38	233,257.58	221,359.16	129,851.81
	CTE 95%	274,996.31	295,325.47	281,371.20	164,324.93

Table 4.5: Comparison of predictive performances among three Bayesian hierarchical GLMs—Model (A), (B), and (C)—and the GLM-based SIMEX, based on the LGPIF data with a covariate error rate of $R_{\epsilon_x} = 0.10$ and a scaling factor of $\zeta = 0.6$.

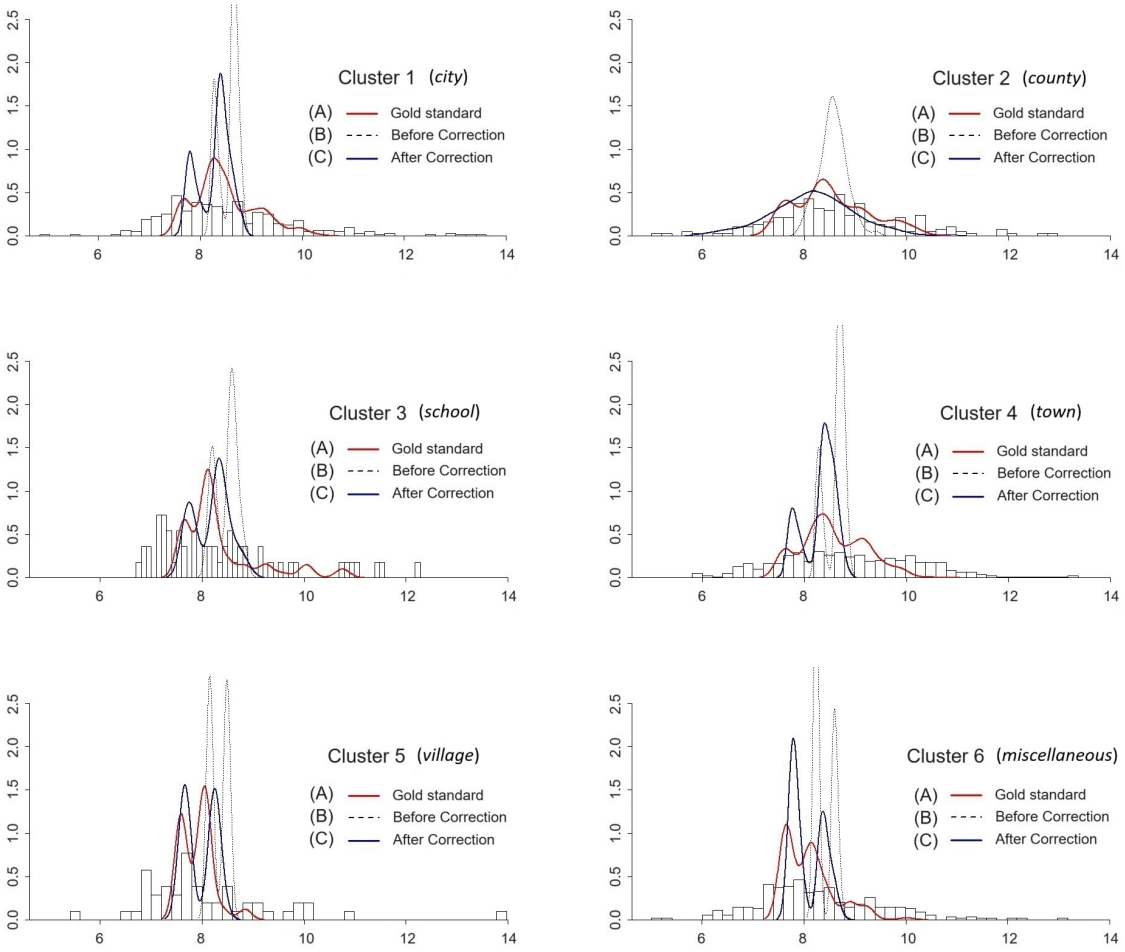

Figure 4.10: Fitted models based on the LGPIF data with the error rate $R_{\epsilon_x} = 0.10$ and the scaling factor $\zeta = 0.6$: Cluster-wise histograms (for $j = 1, \cdots 6$) of the observed claim amount Y_h on a log scale and the out-of-sample predictive densities obtained from Model(A), (B), and (C)

$\zeta = 0.5$ $R_{\epsilon_x} = 0.40$	Feature	Model(A): Gold standard	Model(B): with Model Risk	Model(C) Gustafson correction	Model(D): SIMEX correction
$f(ln\bar{Y}_h \mid \mathbf{X}^S)$	LPPD	-16,155.90	-18,058.43	**-16,188.31**	-
	SSPE	784.52	861.44	793.50	954.04
	SAPE	415.21	437.73	417.36	532.72
	D_{KL}	0.00	1.94	0.33	-
$f(lnS_h \mid \mathbf{X}^F, \mathbf{X}^S)$	CTE 10%	48,782.40	54,671.42	49,155.18	54,716.98
	CTE 50%	81,593.58	89,824.04	84,038.84	75,489.60
	CTE 90%	209,761.38	233,321.29	223,497.02	174,000.54
	CTE 95%	274,996.31	295,939.47	278,098.80	186,720.87

Table 4.6: Comparison of predictive performances among three Bayesian hierarchical GLMs—Model (A), (B), and (C)—and the GLM-based SIMEX, based on the LGPIF data with a covariate error rate of $R_{\epsilon_x} = 0.40$ and a scaling factor of $\zeta = 0.5$.

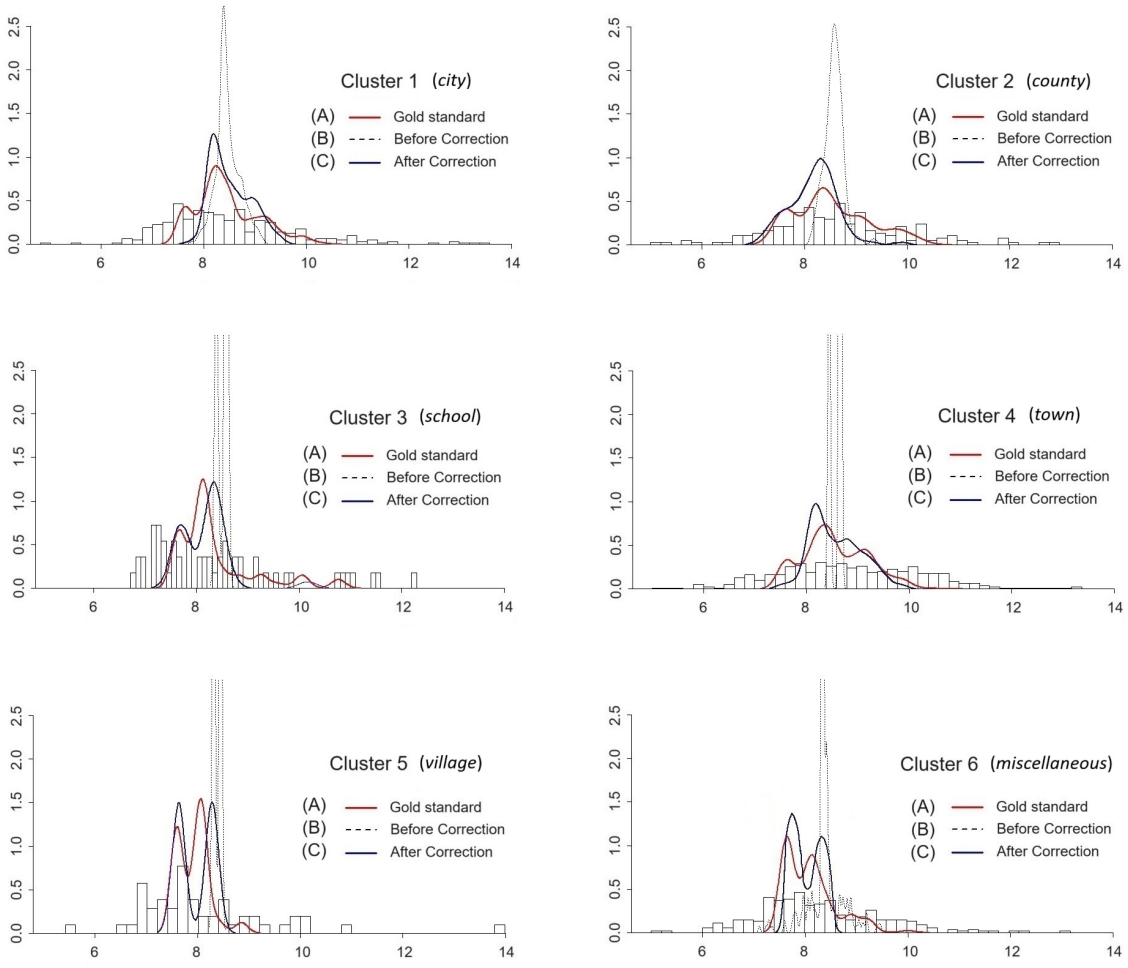

Figure 4.11: Fitted models based on the LGPIF data with the error rate=0.40 and the scaling factor $\zeta = 0.5$: Cluster-wise histograms (for $j = 1, \cdots 6$) of the observed claim amount Y_h on a log scale and the out-of-sample predictive densities obtained from Model(A), (B), and (C)

The effectiveness of the Gustafson correction technique is further substantiated by D_{KL}, which quantifies the distance between the estimated distribution produced by Model(B) and Model(C) and the target (gold standard) distribution, Model(A). A smaller D_{KL} value indicates a more accurate representation of the error correction. Notably, as shown in Tables 4.4, 4.5, and 4.6, within the range of $0.5 \leq \zeta \leq 0.7$, Model(C) exhibits a reduction in D_{KL} from 0.61 to 0.33 as the error rate increases from $R_{\epsilon_x} = 0.01$ to $R_{\epsilon_x} = 0.40$. This suggests that, when $0.5 \leq \zeta \leq 0.7$, the Gustafson correction clearly mitigates the effects of the NDB covariate, and thus enhances the model's fidelity to the true data distribution as the error rates R_{ϵ_x} rise. Note that Model(D) - a GLM with SIMEX - is excluded from this comparison based on the D_{KL} metrics because it is a frequentist model that does not yield LPPD value required for such a divergence analysis.

In the Conditional Tail Expectation (CTE) analysis, the predictive distribution generated by the Gustafson correction in Model(C) exhibits thicker tails compared to Model(D), with higher CTE values of $CTE_{95\%} = 269,656$ at $R_{\epsilon_x} = 0.01$, $CTE_{95\%} = 281,371$ at $R_{\epsilon_x} = 0.10$, and $CTE_{95\%} = 278,099$ at $R_{\epsilon_x} = 0.40$. Indeed, Model(C)'s CTE values lack a consistent trend across error rates. However, its higher CTE values suggest that while the Gustafson correction boosts predictive accuracy, the hierarchical GLM may handle outliers effectively, making it useful for outlier-sensitive applications (Brazauskas et al. 2008).

Upon inspection of Figures 4.9, 4.10, and 4.11, it is clear that the Gustafson correction (Model(C), blue curve) effectively mitigates the model risk from the NDB covariate (Model(B), dotted curve) across clusters $j = 1, \cdots 6$. The hierarchical GLM with Gustafson correction (Model(C), blue curve) aligns closely with the gold standard (Model(A), red curve), particularly in scenarios with higher error rates, such as $R_{\epsilon_x} = 0.40$. However, a slight gap between the two models - Model(A) and (C) - remains, indicating that the correction, while beneficial, is not flawless. In contrast, Model(B) displays significant distortions, including extreme shrinkage in variations and multiple peaks, which worsen as the error rate increases. The

improvement shown in Model(C) is most pronounced at higher error rates such as $R_{\epsilon_x} = 0.40$ and with scaling factors ζ ranging from 0.5 to 0.7. This indicates that the relationship between $V(\mathbf{x}^*|\mathbf{x})$ and $V(\mathbf{x}^*|\mathbf{z})$ in the LGPIF dataset is characterized by this range of ζ. Consequently, it may be valuable to explore other datasets where the relationship between $V(\mathbf{x}^*|\mathbf{x})$ and $V(\mathbf{x}^*|\mathbf{z})$ corresponds to different scaling factor ranges. We will further continue this investigation in Chapter 6.

4.4.4 Discussion

In this chapter, we focused on the development of risk premiums by modeling the aggregate claim amount using the Frequency-Severity principle. Within this framework, we assumed the presence of the NDB covariate in the severity component, which introduces specific covariate-baed model risks. To address these model risks - specifically heterogeneity (RQ1.1) and the NDB covariate (RQ2.2) - we developed a new approach for modeling the conditional claim amount $Y_h|\mathbf{X}^S$.

Concerning RQ.1.1, we first examined the partial pooling effect provided by a Bayesian hierarchical GLM framework to resolve the heterogeneity issue. As for RQ2.2, we proposed extending the Bayesian hierarchical GLM with Gustafson correction to mitigate the covariate-based model risk arising from the NDB covariate. Throughout our experiment, we demonstrated the impact of the model risk when using a naïve model built on the NBD covariate within a Bayesian hierarchical GLM framework. This model misspecification led to significant distortion in the spread and introduced extreme modality in the predictive distribution. In contrast, we have shown the the Gustafson correction's effectiveness in setting of a Bayesian hierarchical GLM, which mitigates the model risk and restores the original properties of the predictive distribution. The effectiveness of mitigating NDB errors was further validated through various performance metrics, e.g. LPPD, SSPE, SAPE, D_{KL}, etc.

A fundamental component of our hybrid modeling framework is the selection of the scaling factor ζ, which our findings indicate is vital for effective error correction. Specifically, ζ regulates the influence of $\widehat{\lambda}^2 : V(\mathbf{x}^*|\mathbf{z})$ to elucidating $\tau^2 : V(\mathbf{x}^*|\mathbf{x})$,

thereby directing the adjustments of erroneous parameters. Our research revealed that both the scaling factor ζ and the error rates R_{ϵ_x} in the NDB covariate are essential, necessitating meticulous calibration for optimal outcomes. To identify the ideal ζ, we established a rule of thumb based on maximizing the LPPD value, offering a practical approach even in the absence of gold-standard data. Our experimental findings suggest that the hierarchical GLM, augmented with the Gustafson correction, demonstrates consistent performance across various error rate scenarios, particularly within the scaling factor range of $0.5 \leq \zeta \leq 0.7$. However, this also implies a strong degree of relation between $\widehat{\lambda}^2 : V(\mathbf{x}^*|\mathbf{z})$ and $\tau^2 : V(\mathbf{x}^*|\mathbf{x})$, as reflected by the prior knowledge: $\tau_j^2 = (1-\zeta)\widehat{\lambda}_j^2$ in Equation (4.24). Moreover, the optimal ζ appeared to be dependent on the dataset, suggesting that unique relationships among covariates may lead to different range of ζ by different datasets. We have yet to investigate additional datasets where this degree of relation might manifest with distinct ranges for ζ. Consequently, it is important to examine how correction performance fluctuates across different error rate scenarios when the optimal range of ζ varies. This underscores the necessity for further research using diverse datasets to comprehensively understand the impact of ζ on the correction performance.

In the upcoming chapter, we will explore a more sophisticated Bayesian approach using the Dirichlet process mixture model to predict risk premiums. While we will continue to focus on covariate-related model risks, our attention will shift from issues related to the NDB covariate to those involving missing covariates under the MAR assumption. However, in Chapter 6, we will return to the challenges associated with the NDB covariate, re-evaluating the effectiveness of the Gustafson correction technique within a more advanced Bayesian framework, on a larger scale, and across multiple datasets. This will enable us to validate our previous findings and further investigate the practical guidelines we have established in this chapter.

Chapter 5

Bayesian Nonparametric I: DPM with MAR Covariate

5.1 Introduction: RQ1.1, RQ1.2, RQ2.1

Chapter 3 has outlined how the expected total claim $E[S_h]$ for a policy h can be developed from two distinct perspectives:

- Frequency-Severity approach for a policy h in Equation (3.1a)
- Compound approach for a policy h in Equation (3.1b)

Following Chapter 4, this chapter examines the 'Compound' principle to the risk premium estimation, under the assumption of a high degree of correlation between claim counts N_h and amounts Y_{hi}. Recall the general definition of the expected aggregate claim for a policy h from Equation (3.1b): $E[S_h] = \sum_{i=1}^{N_h} E[Y_{hi}] = E[Y_{h1}] + E[Y_{h2}] + \ldots + E[Y_{hN_h}]$. Let the covariate $\mathbf{X} = \{\mathbf{z}, \mathbf{x}\}$, where $\mathbf{z}$ is a binary covariate with missing values assumed to be missing at random (MAR). We have discussed that with the inclusion of the covariate that has missing values z_h, the curve development for $E[S_h]$ can encounter the specific types of the model risks due to heterogeneity (RQ1.1), convolution error (RQ1.2), and MAR covariate (RQ2.1), which results in $E[S_h|\mathbf{X}] \neq E[Y_{h1}|\mathbf{X}_{h1}] + E[Y_{h2}|\mathbf{X}_{h2}] + \cdots + E[Y_{hN_h}|\mathbf{X}_{hN_h}]$.

Recent literature on the risk premium modeling - Hong and R. Martin 2018; Huang and S. Meng 2020; Shams 2022; Ungolo and Heuvel 2024 etc. - acknowledge the advantages of using a Dirichlet Process Mixture (DPM) as a Bayesian Nonparametric (BNP) model to mitigate model risk associated with heterogeneity issue (RQ1.1). They illustrate how the DPM can effectively capture complex distributional characteristics of the claim data, such as high skewness, zero inflation[1], hump shape, multi-modality, which arise from heterogeneity issue (Neuhaus and McCulloch 2006).

In Section 3.4.2, we have also elaborated on how the DPM takes into account such sources of heterogeneity via an extensive simulation of S_h and the investigation of multiple mixture scenarios of S_h. By associating each observation with every conceivable risk clustering scenario built upon an infinite dimensional parametric structure, the DPM optimizes prediction values for the future claim amount of S_h. However, with covariates included, less focus has been given to covariate-based model risks like convolution error (RQ1.2) and MAR covariate (RQ2.1), which often cluster with heterogeneity issue (RQ1.1) in the development of $S_h|\mathbf{X}$.

5.2 Our Contribution

In view of the above, this chapter is devoted to resolving these model risks - RQ1.1, RQ1.2, and RQ2.1 - by proposing a Dirichlet process log-skewnormal mixture to model the conditional aggregate claim amount $S_h|\mathbf{X}$. We seek to establish novel links among the covariate-dependent DPM, log-normal convolution, and the MAR covariate imputation, all unified under the BNP framework. For the convolution issue, we adopt the log-skewnormal approximation method studied by Li 2008 to compute the sum of log-normal random variables $\sum_{i=1}^{N_h(t)} E[Y_{hi}|\mathbf{X}]$, which is described in Section 3.2.2. Concerning the problem of the MAR covariate, as outlined in Section 3.3.1, we exploit the generative capability of the data augmentation process

[1]Zero-inflated claim amounts are common in insurance data due to many policyholders not filing claims (Boland 2006).

interlaced with Sethuraman 1994's generalized *stick-breaking*[2] method. This captures the latent structure of data, allowing for a rigorous statistical treatment for the MAR covariate.

To clarify, the contribution of this chapter is twofold. Firstly, by using the log-skewnormal density as the underlying distribution for the outcome S_h in the DPM, we address both the lack of closed-form solutions for log-normal convolution (RQ1.2) and the heterogeneity (RQ1.1) in the log-normal random variable simultaneously. Secondly, we address the adverse effects resulting from the inclusion of covariates in the risk premium modeling framework. This involves addressing the increased heterogeneity (RQ1.1) across Y_{hi} as well as missing information in the MAR covariates (RQ2.2). To our knowledge, this is the first attempt to estimate a log-skewnormal mixture within the BNP framework and apply the DPM to handle MAR covariates in insurance risk premium modeling.

We evaluate the performance of our DPM model by applying it to various insurance datasets with varying sizes and degrees of data missingness (introduced in Section 5.5.1). Empirical results demonstrate the advantages of our DPM model over classical risk premium models, such as the Tweedie-based generalized linear models (GLMs), generalized additive models (GAMs), or multivariate adaptive regression spline (MARS), which are discussed in Chapter 2.

5.3 Modeling Method for $S_h|\mathbf{X}$

5.3.1 Clustering Components

As discussed in Section 3.2.2, modeling log-skewnormal outcomes $S_h|\mathbf{X}$ can address the research question on the convolution error (RQ1.2) at hand. However, given the presence of numerous unidentified risk classes (clusters) across the claim information of $Y_{hi}|\mathbf{X}$ within each policy h, the aggregate claims $S_h|\mathbf{X}$ exhibit diverse characteris-

[2]Stick-breaking method refers to the cluster weight construction technique to generate clustering samples from a Dirichlet process. It involves conceptualizing the breaking a stick of unit length into infinitely many pieces, each of which determines the size of a cluster component (Teh and Jordan 2010).

tics that cannot be adequately captured by fitting a single log-skewnormal distribution (RQ1.1). Therefore, the idea is that, within a DPM framework, we investigate diverse clustering scenarios by leveraging the infinite mixture of log-skewnormal clusters and their complex dependencies as suggested by Hong and R. Martin 2018.

Recall the standard formulation of the DP prior elaborated in Equation (3.34), which generates a distribution over clustering scenarios G:

$$\begin{aligned} \boldsymbol{\phi}_j : \{\boldsymbol{\theta}_j,\ \boldsymbol{w}_j\} &\sim G \\ G &\sim \mathbf{DP}\big(\alpha,\ G_0\big) \end{aligned} \tag{5.1}$$

where the major components (clustering parameters) of G are denoted as

- $\boldsymbol{\theta}_j$: the parameters of the outcome variable defined with cluster j.
- $\boldsymbol{w}_j$: the parameters of the covariates defined with cluster j.

G, as a single realization of the joint cluster probability vector $\{G(A_1), G(A_2)\ldots\}$ sampled from the DP prior, takes independent partitions $A_1, A_2, \ldots$ of the sample space $\bigcup_{k=1}^{\infty} A_k = A$ of the support of G_0. Through ample simulations of G, the DPM explores every conceivable clustering scenario rather than depending on a solitary best estimate (the precision α in the DP prior will be elaborated later).

As for the cluster mixing weight for each $\boldsymbol{\phi}_j$, we note that the original research on the DPM by Hong and R. Martin 2018 focuses on the random mixing weights $\boldsymbol{\omega}_j$, which are independent of the covariates $\mathbf{X} : \{\mathbf{z}, \mathbf{x}\}$. In contrast, we consider integrating covariate effects into the formulation of the mixing weights $\boldsymbol{\omega}_j(\mathbf{X})$. This allows the clustering mixture to maintain homogeneity within each risk cluster while identifying the distinct relationship between the aggregate claim amount and the various risk factors among policyholders of different types. To this end, we adopt a generalized representation of the *stick-breaking process* developed by Sethuraman 1994 and incorporate the covariate effects into the mixing weight.

The DPM with Sethuraman's stick-breaking formulation provides a versatile joint distribution for treating MAR covariates (RQ2.1) as well. Operating as a

generative framework[3], the DPM, in conjunction with the stick-breaking method, jointly models both outcomes S_h and covariates $\mathbf{X}_h$ to determine their latent cluster memberships. This serves as crucial information for identifying the latent structure of the data as well as subsequently recovering the missing information (Shahbaba and Neal 2009). This generative process in the DPM also provides all essential clustering components for constructing the predictive distribution, utilizing infinite clustering scenarios derived from the joint density of observed outcomes S, covariates $\mathbf{X}$, and their latent cluster memberships. To clarify the distribution choices for the outcome and covariates, a summary table in Appendix E provides further insights into the rationale behind these selections.

5.3.2 Discrete and Continuous Clusters

In this chapter, we consider the following baseline model setup: Let the outcome $S = \{S_{h=1}, S_{h=2}, \ldots, S_{h=H}\}$ denote the H different aggregate claims incurred by the H different policies. Regarding the covariates, we assume, as before, that $\mathbf{z}$ is binary, $\mathbf{x}$ is Gaussian, outcome S_h is the zero-inflated log-skewnormal and then our baseline DPM model can be defined as

$$
\begin{aligned}
S_h|z_h, x_h, \boldsymbol{\beta}_j, \sigma_j^2, \xi_j, \tilde{\boldsymbol{\beta}}_j \sim & \\
& \delta(\mathbf{X}_h^T\tilde{\boldsymbol{\beta}}_j)\ \mathbb{1}_{(S_h=0)} + \left[1-\delta(\mathbf{X}_h^T\tilde{\boldsymbol{\beta}}_j)\right]\ \mathbf{LogSN}\big(\mathbf{X}_h^T\boldsymbol{\beta}_j,\ \sigma_j^2,\ \xi_j\big) && \text{(5.2a)}\\
& z_h|\pi_j \ \sim\ \mathbf{Bernoulli}\big(\pi_j\big) && \text{(5.2b)}\\
& x_h|\mu_j, \lambda_j^2 \ \sim\ \mathbf{N}\big(\mu_j,\ \lambda_j^2\big) && \text{(5.2c)}\\
\boldsymbol{\phi}_j : \{\boldsymbol{\theta}_j,\ \boldsymbol{w}_j,\ \boldsymbol{\omega}_j(\mathbf{X}_h)\} \ &\sim\ G && \text{(5.2d)}\\
& G \ \sim\ \mathbf{DP}\big(\alpha,\ G_0\big) && \text{(5.2e)}
\end{aligned}
$$

where: j is the risk cluster index; $\mathbf{X}_h = \{z_h, x_h\}$ for the covariates; $\boldsymbol{\theta}_j = \{\boldsymbol{\beta}_j, \sigma_j^2, \xi_j, \tilde{\boldsymbol{\beta}}_j\}$ for parameters describing the outcome; $\boldsymbol{w}_j = \{\pi_j, \mu_j, \lambda_j^2\}$ for parameters describing

[3]The DPM is a generative model as it aims to learn the joint density of covariates and outcome rather than solely focusing on learning the decision boundary between clusters (Jebara 2012).

the covariates; and $\boldsymbol{\omega}_j(\mathbf{X}_h)$ for the covariate-dependent mixing weight, satisfying $\sum_{j=1}^{\infty} \boldsymbol{\omega}_j(\mathbf{X}_h) = 1$. A combination of a point mass at 0 and positive values distributed with a log-skewnormal density is used in Equation (5.2a) to handle the complications of the zero inflation in the aggregate claim data S_h. In Equation (5.2a), the term $\delta(\mathbf{X}_h^T\tilde{\boldsymbol{\beta}}_j)$ represents a multivariate logistic regression model that predicts the probability of S_h being zero (no claim). Note that this logistic regression as a part of the outcome model uses the parameter vector $\tilde{\boldsymbol{\beta}}_j$ to explain the zero outcomes, while the non-zero outcomes are explained by the log-skewnormal density using the parameter vector $\boldsymbol{\beta}_j$. See Appendix A for a detailed description of variable definitions. Examining a Dirichlet process log-skewnormal mixture to accommodate multiple unknown risk clusters within S_h, it is important to distinguish between the forms of mixture components depending on the types of clusters employed - discrete and continuous. While ensuring that the inference of the cluster parameters remains data-driven, the DPM initially establishes discrete clusters using available claim information and reasonable machine learning algorithms such as *K-means* or *K-prototype* (Ahmad 2014), etc., and then extrapolates new continuous clusters for unknown risk classes by capturing the heterogeneity within each cluster. Throughout this process, the DPM creates various clustering scenarios and evaluates them using specific decision-making algorithms such as Polya Urn scheme (Teh and Jordan 2010), etc., which underpins computationally efficient and asymptotically consistent parameter estimations (Hong and R. Martin 2017).

The discrete clusters in the DPM follow standard distributional forms that are useful for modeling readily observable data such as distinct policy information for S_h, etc. (Diebolt and Robert 1994). Hence, when computing the probabilities associated with discrete clusters, we make the assumption that the non-zero outcomes S_h and covariates $\mathbf{X}_h$ follow the forms given by

$$f_{LSN}\big(S_h|\mathbf{X}_h^T\boldsymbol{\beta}_j,\ \sigma_j^2,\ \xi_j\big) = \frac{2}{S_h\sigma_j}\phi\Big(\frac{\log S_h - \mathbf{X}_h^T\boldsymbol{\beta}_j}{\sigma_j}\Big)\Phi\Big(\xi_j\frac{\log S_h - \mathbf{X}_h^T\boldsymbol{\beta}_j}{\sigma_j}\Big) \tag{5.3a}$$

$$f_{Bern}\big(z_h|\pi_j\big) = \pi_j^{z_h}\left(1-\pi_j\right)^{1-z_h} \tag{5.3b}$$

$$f_N\big(x_h|\mu_j,\lambda_j^2\big)=\frac{1}{\sqrt{2\pi\lambda_j^2}}\exp\left\{-\frac{1}{2\lambda_j^2}\left(x_h-\mu_j\right)^2\right\} \tag{5.3c}$$

where $\phi(\cdot)$ and $\Phi(\cdot)$ are standard normal probability and cumulative density functions for the formulation of a log-skewnormal density. To model the predictive distribution of $S_h|\mathbf{X}$ for the policy h, the general form of the mixture of the discrete log-skewnormal clusters can be expressed as

$$\begin{aligned} &f(S_h|\mathbf{X}_h,\boldsymbol{\theta},\boldsymbol{w}) \\ &\quad=\sum_{j=1}^{J}\boldsymbol{\omega}_j(\mathbf{X}_h|\boldsymbol{w}_j)\Big(\delta(\mathbf{X}_h^T\tilde{\boldsymbol{\beta}}_j)\ \mathbb{1}_{(S_h=0)}+\left[1-\delta(\mathbf{X}_h^T\tilde{\boldsymbol{\beta}}_j)\right]\ f_{LSN}(S_h|\mathbf{X}_h,\boldsymbol{\theta}_j)\Big) \end{aligned} \tag{5.4}$$

where Equation (5.3b) and (5.3c) may constitute the mixing weight term $\boldsymbol{\omega}_j(\mathbf{X}_h|\boldsymbol{w}_j)$ above as well.

Once the finite number of mixture components J, where $J \leq H$, is determined from the available data, however, the base measure G_0 introduces brand-new continuous clusters to tackle the unobservable risk classes. Through the involvement of G_0, the DPM confronts the current clustering result and examines homogeneous clusters more closely, and thus the within-class heterogeneity in $S_h|\mathbf{X}$ can be addressed. Given that the new clusters are considered countably infinite[4], their corresponding forms of the outcome and covariate models to compute the continuous cluster can be defined as

$$f_0(S_h|\mathbf{X}_h)=\int f(S_h|\mathbf{X}_h,\boldsymbol{\theta})\ dG_0(\boldsymbol{\theta}) \tag{5.5a}$$

$$f_0(z_h)=\int f_{Bern}(z_h|\boldsymbol{w})\ dG_0(\boldsymbol{w}) \tag{5.5b}$$

$$f_0(x_h)=\int f_N(x_h|\boldsymbol{w})\ dG_0(\boldsymbol{w}) \tag{5.5c}$$

Equation (5.5a) is referred to as a *parameter-free* outcome model, while Equations (5.5b) and (5.5c) represent *parameter-free* covariate models, respectively. Note

[4]It refers to a type of infinity where the elements can be put into a one-to-one correspondence with the natural numbers (Bouguila and Ziou 2012).

that the term $f(S_h|\mathbf{X}_h, \boldsymbol{\theta})$ in Equation (5.5a) is considered as a mixture of a log-skewnormal given by Equation (5.2a). All of these equations are used to develop the new continuous cluster mixture (Ferguson 1973).

Given a collection of outcome-covariate data pairs $\{S_h, \mathbf{X}_h\}_{h=1}^{H}$, the DPM combines the current discrete clusters with new continuous clusters to update the mixture form in Equation (5.4), employing Monte Carlo simulation (or the parameter-free clustering technique mentioned in Section 3.2.1) to draw posterior samples of the parameters $\boldsymbol{\theta}_j, \boldsymbol{w}_j$ sufficiently. Consequently, the clustering scenario G described in Equation (5.2e) becomes $G = \sum_{j=1}^{\infty} \boldsymbol{\omega}_j(\mathbf{X}_h)\delta_{\phi_j}(\cdot)$, a point mass distribution of $\boldsymbol{\phi}$ sampled from G_0 (further details are discussed in Section 3.4.2). In line with this flexible clustering scenario development, the selected point masses $\phi_1, \cdots \phi_{J+1}$ choose their own mixing weights $\boldsymbol{\omega}_j^{(*)}$ (i.e., finalized version of $\boldsymbol{\omega}_j$), derived from the finite number of discrete and continuous clusters $j = 1, \cdots J+1$, and the predictive distribution of $S_h|\mathbf{X}$ in Equation (5.4) can be re-shaped by the mixing weights $\boldsymbol{\omega}_1^{(*)}(\mathbf{X}_h), \cdots, \boldsymbol{\omega}_{J+1}^{(*)}(\mathbf{X}_h)$ as below.

$$
\begin{aligned}
&f(S_h|\mathbf{X}_h, \boldsymbol{\theta}, \boldsymbol{w}, \alpha) \\
&= \underbrace{\frac{\boldsymbol{\omega}_{J+1}^{(*)}(\mathbf{X}_h)}{\boldsymbol{\omega}_{J+1}^{(*)}(\mathbf{X}_h) + \sum_{j=1}^{J} \boldsymbol{\omega}_j^{(*)}(\mathbf{X}_h)} \cdot f_0(S_h|\mathbf{X}_h)}_{\text{For the new continuous cluster}} + \underbrace{\frac{\sum_{j=1}^{J} \boldsymbol{\omega}_j^{(*)}(\mathbf{X}_h) \cdot f(S_h|\mathbf{X}_h, \boldsymbol{\theta}_j)}{\boldsymbol{\omega}_{J+1}^{(*)}(\mathbf{X}_h) + \sum_{j=1}^{J} \boldsymbol{\omega}_j^{(*)}(\mathbf{X}_h)}}_{\text{For the established discrete clusters}}
\end{aligned}
\tag{5.6}
$$

where the continuous clustering components $f_0(S_h|\mathbf{X}_h)$ above is defined in Equation (5.5a), and the discrete clustering components $f(S_h|\mathbf{X}_h, \boldsymbol{\theta})$ is, as mentioned previously, equal to Equation (5.2a). As for the computation of the mixing weights $\boldsymbol{\omega}_1^{(*)}(\mathbf{X}_h), \cdots, \boldsymbol{\omega}_{J+1}^{(*)}(\mathbf{X}_h)$ in G, Sethuraman 1994's generalized stick-breaking formulations outlined below for continuous and discrete clusters are considered.

$$
\boldsymbol{\omega}_{J+1}^{(*)}(\mathbf{X}_h) = \frac{\alpha}{\alpha + H} \cdot f_0(z_h, x_h) \qquad \text{for continuous clusters} \tag{5.7a}
$$

$$
\boldsymbol{\omega}_{j}^{(*)}(\mathbf{X}_h) = \frac{n_j}{\alpha + H} \cdot f(z_h, x_h|\boldsymbol{w}_j = (\pi_j, \mu_j, \lambda_j^2)) \quad \text{for discrete clusters} \tag{5.7b}
$$

where: α is the precision parameter to control the acceptance rate of the new clusters; n_j is the number of observations in cluster j; $f_0(z_h, x_h)$ is the joint of parameter-free covariate models in Equation (5.5b, 5.5c) to explain the importance of the new continuous clusters; and $f(z_h, x_h|\boldsymbol{w}_j)$ is the joint covariate models in Equation (5.3b, 5.3c) to describe the importance of the current discrete clusters. Therefore, it is evident that these mixing weights $\boldsymbol{\omega}_1^{(*)}(\mathbf{X}_h), \cdots \boldsymbol{\omega}_{J+1}^{(*)}(\mathbf{X}_h)$ mirror the characteristics of the clusters and associated covariates.

Note that the form of the predictive distribution described in Equation (5.6) results from the full joint model of the outcome S_h and covariates $\mathbf{X}_h$ conditioned on the posterior samples. For instance, the major mixture components - $\boldsymbol{\omega}_j^{(*)}(\mathbf{X}_h) \cdot f(S_h|\mathbf{X}_h, \boldsymbol{\theta}_j)$ and $\boldsymbol{\omega}_{J+1}^{(*)}(\mathbf{X}_h) \cdot f_0(S_h|\mathbf{X}_h)$ - in Equation (5.6) comprise the joint distribution of $\{S_h, \mathbf{X}_h\}$ since the mixing weights are proportional to the covariate models. This is based on the rationale of the *Polya Urn* distribution in Equation (3.35) introduced by Blackwell and MacQueen 1973, which is also aligned with the idea of the generalized stick-breaking representation of the DPM in Equation (5.7).

5.3.3 Clustering $S_h|\mathbf{X}$ with Complete Case Covariate

Regarding the research question on the issue of heterogeneity (RQ1.1), Section 3.2.1 briefly introduced parameter-free clustering techniques in connection with a Gibbs sampler. This section now gives a detailed explanation of the DPM Gibbs sampler, specifically focusing on computing the outcome and covariate parameters - $\boldsymbol{\theta}_j, \boldsymbol{w}_j$ - in Equation (5.6) and (5.7), assuming no errors in the covariate.

In short, the DPM Gibbs sampler obtains draws from the analytically intractable posterior, alternating between two stages to ensure convergence - 1) Re-assigning the 'cluster membership' for each observation, and 2) updating the 'cluster parameters' given the cluster partitioning. By iterating through this process numerous times (e.g., M=100,000 iterations), each iteration may yield a slightly varied selection of new clusters according to the Polya Urn scheme (Gershman and Blei 2012).

However, the log-likelihood value computed at the end of each iteration can aid in monitoring the convergence of these selections. We give implementation details of the DPM Gibbs sampler in Algorithm (E.3) in Appendix E. The DPM Gibbs sampler, with its two stages, is further described in what follows.

[**Stage.1**] **Re-assigning cluster memberships with Gibbs sampler:** This stage, with three steps, is briefly illustrated in Figure 5.1.

Step.I As mentioned in the previous section, the initial cluster membership j is determined for the discrete clusters $j = 1, 2, \cdots, J$ using well-known clustering methods such as hierarchical or K-mean clustering, etc. Therefore, the initial number of clusters J built upon the data $\{S_h, \mathbf{X}_h\}$ is presented.

Step.II Let the cluster index for observation h be denoted by s_h. With the candidate parameters $\{\boldsymbol{\theta}_j, \boldsymbol{w}_j\}$ sampled from the DP prior G_0 and the conditional probability term $p(s_h|s_{-h})$ on lines 6 and 9 in Algorithm (E.3) for the observation assignment, the probabilities of the selected observation h belonging to the current discrete clusters and the proposed continuous cluster are computed respectively. The utilization of a nonparametric prior in developing a new continuous cluster enables the shape of the cluster to be influenced by the data. Note that the term $p(s_h|s_{-h})$ is known as *Chinese Restaurant Process* (CRP) probability (see Blei and Frazier 2011 and Section 3.4.2) given by

$$p(s_h|s_{-h}) = \begin{cases} c \cdot \dfrac{n_j^{-h}}{\alpha + H - 1}, & \text{for } h \text{ entering into the existing cluster: } s_h = j. \\ c \cdot \dfrac{\alpha}{\alpha + H - 1}, & \text{for } h \text{ entering into the new cluster: } s_h = J + 1. \end{cases} \tag{5.8}$$

where s_{-h} is the collection of cluster indices $\{s_1, s_2, \cdots, s_{h-1}, s_{h+1}, \cdots, s_H\}$ assigned to every observation without the cluster index s_h of the observation h, and c is a scaling constant to ensure that the probabilities sum to 1. A larger value of precision α gives a higher chance of developing the new continuous cluster and adding to the collection of the existing discrete clusters.

Since the number of clusters is not fixed, and the sequence of clusters (or cluster memberships) to which each observation is assigned cannot be ordered, there may be concerns regarding additional sampling variance or convergence issues in the DPM Gibbs sampler based on Equation (5.8). This may lead to imprecise ad-hoc comparisons of important cluster probabilities depicted in lines 4 to 10 of Algorithm (E.3). However, Neal 2000 proves that, under the CRP described in Equation (5.8), the sequence in which the observation h arrives in the cluster $s_h = j$ is exchangeable, and thus the DPM Gibbs sampler can carry out stable simulations. This exchangeability feature was also discussed in Section 3.4.2.

Step.III Finally, the new cluster membership is determined and re-assigned with the Polya Urn scheme, using a multinomial distribution based on the cluster probabilities comparison by the CRP. Note that the development of the mixing weights $\boldsymbol{\omega}_1^{(*)}(\mathbf{X}_h), \cdots \boldsymbol{\omega}_{J+1}^{(*)}(\mathbf{X}_h)$ discussed previously in Equation (5.7) is made in the cluster membership re-assignment stage as shown in Figure 5.1. Consider a detailed example in the diagram to see how a series of the mixing weights $\boldsymbol{\omega}_1^{(*)}(\mathbf{X}_h), \cdots \boldsymbol{\omega}_{J+1}^{(*)}(\mathbf{X}_h)$ can be computed based on Equation (5.7) and (5.8), using the total sample size H, the number of data points in each cluster (in the diagram, $n_1 = 4, n_2 = 7, n_3 = 1, \cdots, n_{J-1} = 2, n_J = 6$), and the precision α.

[**Stage.2**] **Updating cluster parameters with Gibbs sampler:** Once all observations have been assigned to particular clusters $j = 1, 2, \cdots, J$ at a given iteration in the DPM Gibbs sampling, the parameters of our interest - $\boldsymbol{\theta}_j, \boldsymbol{w}_j$ and α - for each cluster j are updated, given the new cluster membership j. The posterior densities denoted by $p(\boldsymbol{\theta}_j|S_h, \mathbf{X}_h)$, $p(\boldsymbol{w}_j|\mathbf{X}_h)$, and $p(\alpha|J)$ are utilized to simulate the parameter samples $\{\boldsymbol{\theta}_j^{(*)}, \boldsymbol{w}_j^{(*)}, \alpha^{(*)}\}$ based on all observations $S_h, \mathbf{X}_h$ in cluster j. As for the parameterizations of the posterior densities on lines from 17 to 23 in Algorithm (E.3), we detail them in E.2.3 and E.2.4 in Appendix E.

The DPM Gibbs sampler described so far is distinguished by its approach to

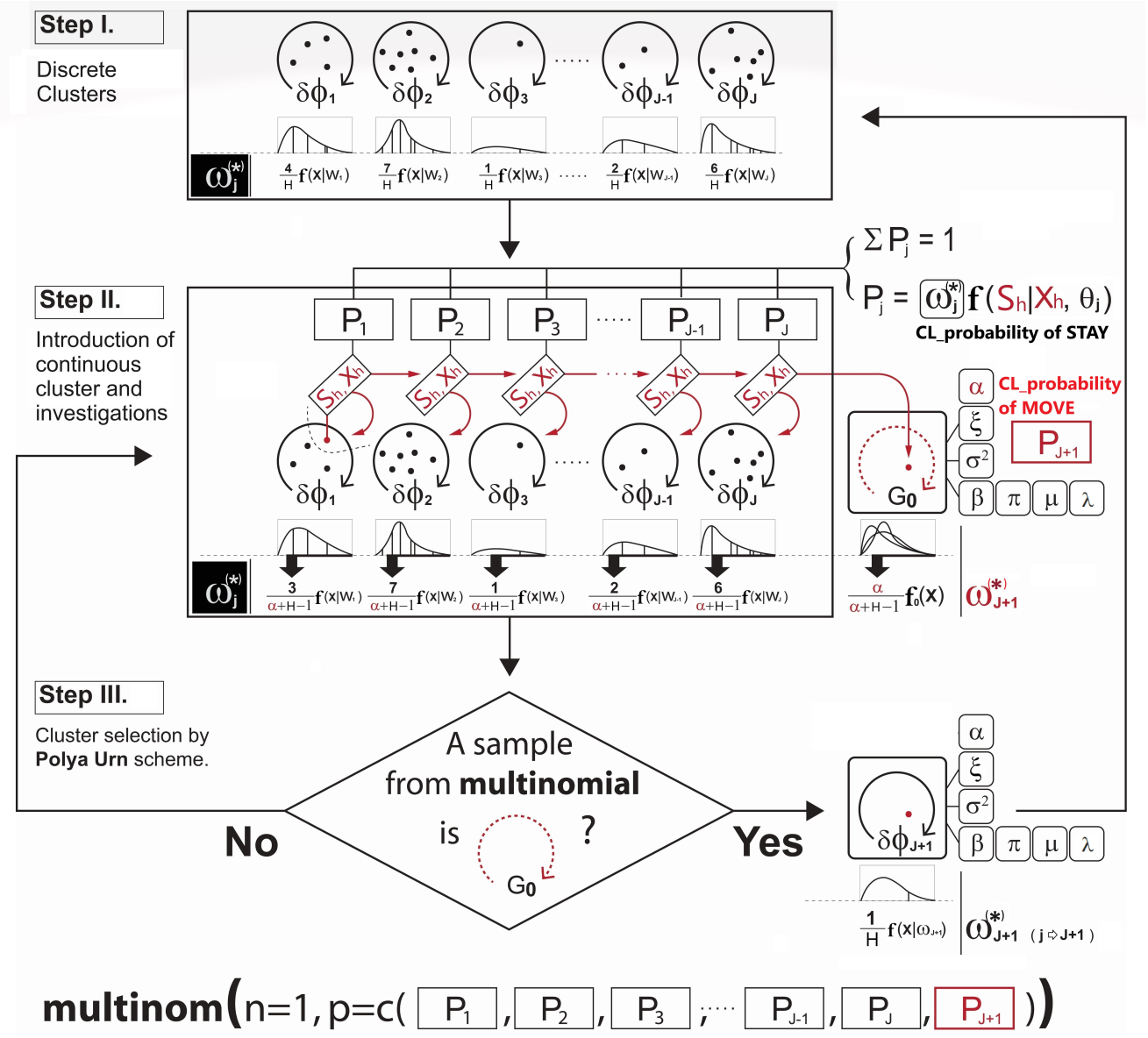

Figure 5.1: A schematic of the 'Re-assigning cluster memberships' in [**Stage.1**], with Step I. Initializing the memberships, Step II. Computing the cluster probability $\mathbf{P}_j$ with the CRP, and Step III. Re-assigning the memberships by the Polya Urn scheme. The cluster membership investigation relies on the computation results of $\boldsymbol{\omega}_j^{(*)}, \boldsymbol{\omega}_{J+1}^{(*)}$.

exploring an infinite number of clustering scenarios using the complete covariates. Although the DPM allows infinite-dimensional clustering, the dimension of the sampling output G is adaptive as it is a mixture with at most finite components determined by data itself (i.e., its dimension cannot exceed the total sample size H). This can be seen from its form of the predictive density in Equation (5.6). In this process of generating clustering scenarios, the risk premium model based on the DPM effectively accommodates all distributional properties of the provided claims, as well as

those of unknown claims, by capturing heterogeneity (RQ1.1) across observations, resulting in improved prediction of the future value of the aggregate claim amount $S_h|\mathbf{X}$ for a policy h.

5.3.4 Clustering $S_h|\mathbf{X}$ with MAR Case Covariate

This section aims to develop a novel DPM Gibbs sampler built upon the MAR covariate (RQ2.1) in which the missing values are explained by the observed data and the cluster membership. The data augmentation technique to resolve the MAR covariate has been elaborated in Section 3.3.1. We will now present how to seamlessly integrate the DPM Gibbs sampler with the data augmentation technique under the assumption of the MAR covariate.

Gibbs sampler modification I for MAR covariate: With the model definition in Equation (5.2), suppose the binary covariate $\mathbf{z}$ has missingness within it. To address this MAR covariate, we propose to incorporate the following modifications (additional steps) into the DPM Gibbs sampler outlined in Algorithm (E.3) in Appendix E:

(a) **Adding an 'Imputation Step' in [Stage.2]:** Assuming we have a missing covariate value in $\mathbf{z}$ at the observation h, the missing covariate z_h directly affects the update of the cluster parameters $\boldsymbol{\theta}_j$ and $\boldsymbol{w}_j$ in **[Stage.2]** '**Updating cluster parameters with Gibbs sampler**' due to the assumption of MAR (i.e., missing values are correlated with the observed data and the cluster membership). For the covariate parameters $\boldsymbol{w}_j = \{\pi_j, \mu_j, \lambda_j^2\}$, only the observations without the missing covariate z_h can be used to update $\boldsymbol{w}_j$ (i.e., we drop the observation h that has the missing value z_h in $\mathbf{z}$). If the cluster j does not have any observations with complete data for that missing covariate z_h, then a random draw from the prior distribution for $\{\pi_j, \mu_j, \lambda_j^2\}$ can be used to update $\boldsymbol{w}_j$. For the outcome parameters $\boldsymbol{\theta}_j = \{\boldsymbol{\beta}_j, \sigma_j^2, \xi_j, \tilde{\boldsymbol{\beta}}_j\}$, however, the imputation for the missing covariates z_h should be made first for all

observations h within the cluster j. This is because the outcome value S_h is conditional on the covariates z_h and x_h.

The imputation can be performed as follows. Given the full joint model $f(S_h|\mathbf{X}_h, \boldsymbol{\theta}_j) \cdot f(\mathbf{X}_h|\boldsymbol{w}_j)$ built on the components in Equation (5.3) and the current cluster parameter values available, we can obtain draws for the MAR covariate z_h from the cluster-wise imputation function based on the joint

$$z_h \sim f_{Bern}(z_h|S_h, x_h, \boldsymbol{\theta}_j, \boldsymbol{w}_j) \propto f(S_h|\mathbf{X}_h, \boldsymbol{\beta}_j, \sigma_j^2, \xi_j, \tilde{\boldsymbol{\beta}}_j) \cdot f_{Bern}(z_h|\pi_j) \quad (5.9)$$

at each iteration in the DPM Gibbs sampling. Each imputation model is proportional to the full joint distribution, which is a product of the outcome model and the covariate model containing missing data. The imputation process is illustrated in depth in Figure 5.2. After imputing all missing covariate values, the parameters of each cluster - $\boldsymbol{\theta}_j$ and $\boldsymbol{w}_j$ - are re-calculated, using the posterior densities. Once this cycle is complete in the DPM Gibbs sampling, the imputed values are discarded and the same imputation steps are repeated every iteration.

(b) **Adding a 'Model Refinement Step' in [Stage.1]:** After updating the cluster parameters, [**Stage.1**] '**Re-assigning cluster memberships with Gibbs sampler**' in Figure 5.1 re-calculates each cluster probability by re-defining its two main components: 1) the covariate model, and 2) the outcome model as depicted in Step III in Figure 5.3. For the covariate model $f(\mathbf{X}_h|\boldsymbol{w}_j)$, we consider only the form of the complete covariates for observation h while discarding the form of the missing covariate (to avoid introducing bias that arises from the missingness). Assuming that $\mathbf{X}_h = \{z_h, x_h\}$, and the covariate $\mathbf{z}$ is missing for observation h, then we drop z_h and only use x_h in the covariate model

$$\text{Refined Covariate: } f(\mathbf{X}_h|\boldsymbol{w}_j) = f_N(x_h|\mu_j, \lambda_j^2) \cdot \cancel{f_{Bern}(z_h|\pi_j)} \quad (5.10)$$

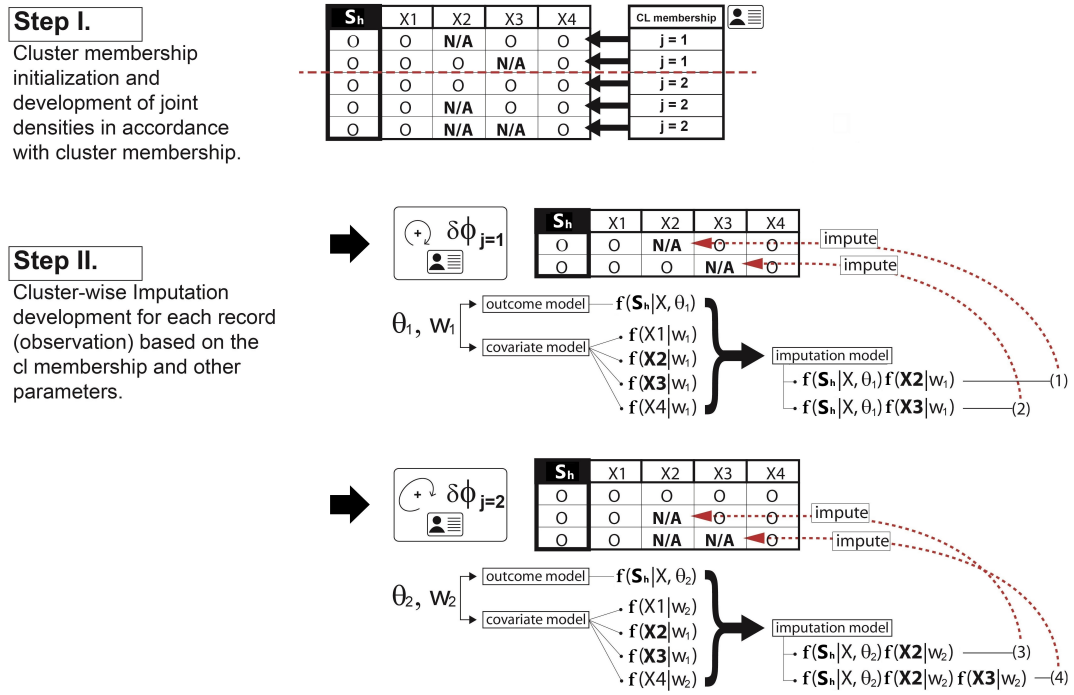

Figure 5.2: An example of the MAR imputation for the [**Stage.2**] in the DPM Gibbs sampler: The imputations are performed cluster membership-wise.

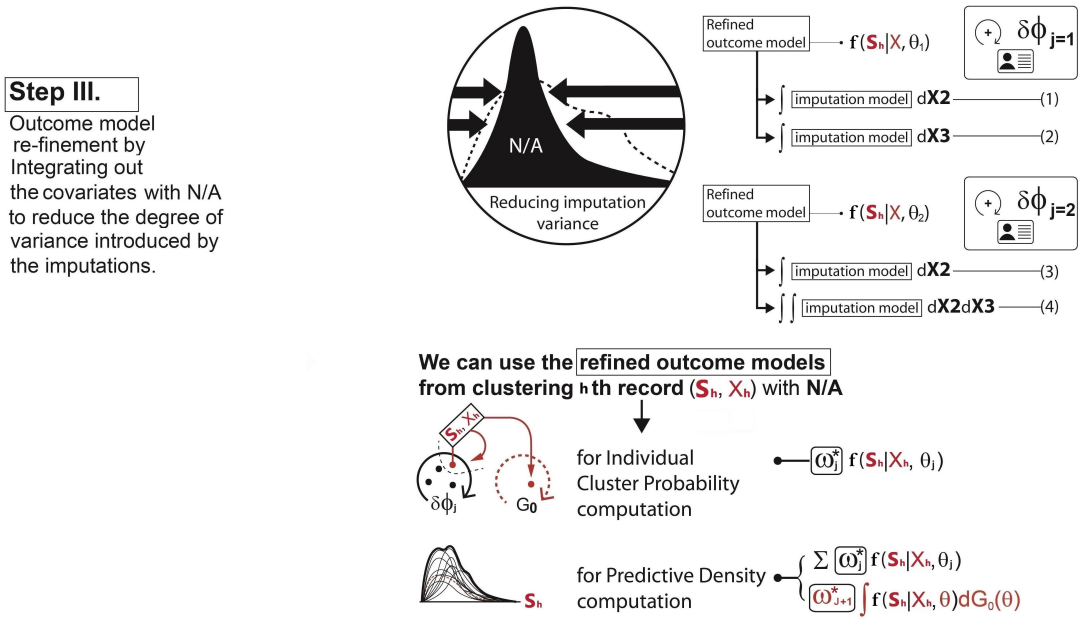

Figure 5.3: An example of the refined outcome model development for [**Stage.1**] in the DPM Gibbs sampler: Step III. Each cluster probability and the predictive density can be calculated based on the model refinement.

by the logic elaborated in Equation (3.25). This is the refined covariate model for the cluster j with the observation h where the data in $\mathbf{z}$ is not available. For the outcome model $f(S_h|\mathbf{X}_h, \boldsymbol{\theta}_j)$, the algorithm simply takes the imputation

function in Equation (5.9) for the observation h and integrates it out the covariate with missingness z_h. As discussed in Section 3.3.1, this reduces the degree of sampling variance introduced by the imputations. The implication is that, since covariate $\mathbf{z}$ is missing for observation h, this missing covariate can be removed from the covariate vector $\mathbf{X}_h$ that is being conditioned on. Therefore, the refined outcome model is given by

$$\text{Refined Outcome: } f(S_h|x_h, \boldsymbol{\theta}_j) \propto \int \underbrace{f(S_h|\mathbf{X}_h, \boldsymbol{\theta}_j) \cdot f_{Bern}(z_h|\pi_j)}_{\text{imputation function}} \, dz_h \quad (5.11)$$

The identical process is conducted for every observation with missing data and every combination of missing covariates. Thus, by utilizing Equations (5.10) and (5.11), the cluster probabilities and the predictive distribution can be derived, ultimately determining a specific clustering scenario.

(c) **Re-updating cluster parameters with Gibbs sampler:** Moving on to [**Stage.2**] '**Updating cluster parameters with Gibbs sampler**' again, the DPM Gibbs sampler re-estimates the cluster parameters - $\boldsymbol{\theta}_j$ and $\boldsymbol{w}_j$ - for each cluster j with the flows of the parameter updates outlined in Figure 5.4. As shown in the diagram, the update for the cluster parameters - $\boldsymbol{\theta}_j$ and $\boldsymbol{w}_j$ - is performed based on the available data, while the precision α is updated according to the results of cluster membership assignments. The update for mixing weights $\boldsymbol{\omega}_j(\mathbf{X}_h)$ is, however, influenced by both the data and the results of cluster membership assignments, which ultimately formulates the clustering scenario development of G as shown in Figure 3.4

Gibbs sampler Modification II with the data augmentation: Now we set out some modification details for the DPM Gibbs sampler implementation described in Algorithm (E.3) in Appendix E to address the MAR covariate $\mathbf{z}$. Below, we elaborate on the modifications integrated into the DPM Gibbs sampler, which mainly involves 're-calculating cluster probabilities' in [**Stage.1**] and 'imputing missing values z_h' to draw outcome parameters $\boldsymbol{\theta}_j^{(*)}$ from the posterior in [**Stage.2**].

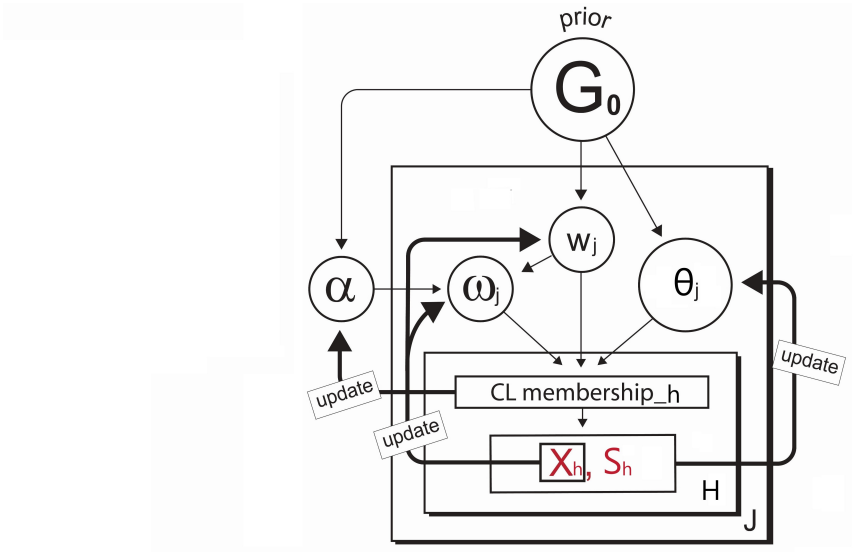

Figure 5.4: The acyclic graphical representation of the flows of the parameter updates in the DPM. This is a snapshot for a single iteration (M=1).

(a) In line 6 in [**Stage.1**] in Algorithm (E.3), with the presence of missing covariate z_h, the modification of the cluster probability computation for the observation $\{S_h, \not{z_h}, x_h\}$ belonging to 'discrete' cluster j can be made, viz:

$$P(s_h = j) = p(s_h|s_{-h}) \cdot f(x_h|\mu_j, \lambda_j^2) \cdot f(S_h|x_h, \boldsymbol{\beta}_j, \sigma_j^2, \xi_j, \tilde{\boldsymbol{\beta}}_j) \tag{5.12}$$

The refined covariate $f(x_h|\mu_j, \lambda_j^2)$ and outcome models $f(S_h|x_h, \boldsymbol{\beta}_j, \sigma_j^2, \xi_j, \tilde{\boldsymbol{\beta}}_j)$ are formally specified in subsequent discussions, as outlined in Equations (5.19) and (5.21), respectively.

(b) In line 9 in [**Stage.1**] in Algorithm (E.3), with the presence of missing covariate z_h, the modification of the cluster probability computation for the observation $\{S_h, \not{z_h}, x_h\}$ belonging to 'continuous' cluster $J+1$ can be made:

$$P(s_h = J+1) = p(s_h|s_{-h}) \cdot f_0(x_h) \cdot f_0(S_h|x_h) \tag{5.13}$$

where the parameter-free covariate model $f_0(x_h)$ is defined from Equation (5.20), and the parameter-free outcome model $f_0(S_h|x_h)$ is determined from Equation (5.22) as will be seen later.

(c) In line 22 in **[Stage.2]** in Algorithm (E.3), with the presence of missing covariate z_h, the imputation should be performed before simulating the outcome parameter $\boldsymbol{\theta}_j^{(*)}$ as follows.

$$\begin{cases} \begin{cases} \text{First, impute } z_h \sim f(S_h|\mathbf{X}_h, \boldsymbol{\beta}_j, \sigma_j^2, \xi_j, \tilde{\boldsymbol{\beta}}_j) \cdot f_{Bern}(z_h|\pi_j) \\ \text{Then sample } \boldsymbol{\theta}_j^{(*)} \text{ from the posterior: } p(\boldsymbol{\theta}|S_h, \mathbf{X}_h) \end{cases} & \text{if } z_h \text{ is missing.} \\ \text{Sample } \boldsymbol{\theta}_j^{(*)} \text{ from the posterior: } p(\boldsymbol{\theta}|S_h, \mathbf{X}_h) & \text{otherwise} \end{cases}$$

The formulation of the imputation function in the above has been discussed in Equation (5.9) and Figure 5.2. Note that the posterior of the outcome parameter model $p(\boldsymbol{\theta}|S_h, \mathbf{X}_h)$ cannot be specified because the conjugate priors for $\boldsymbol{\theta}$ are not available. Instead, we consider the Metropolis-Hastings to obtain the posterior samples of $\boldsymbol{\theta}^{(*)}$.

5.4 Elements of Bayesian Inference

A key aspect of Bayesian inference involves designing a suitable prior and posterior for the parameters of interest to explain the observed data. We refer to the prior and posterior as 'parameter models', and the outcome and covariate terms as 'data models' (Gelman and Carlin 2013). To update the cluster parameters in the Dirichlet process log-skewnormal mixture with the MAR covariate $\mathbf{z}$, we employ Markov Chain Monte Carlo (MCMC) to sample the parameters - $\boldsymbol{\theta} : \{\boldsymbol{\beta}, \sigma^2, \xi, \tilde{\boldsymbol{\beta}}\}$, $\boldsymbol{w} : \{\pi, \mu, \lambda^2\}$, and α - from the joint posterior densities. This requires proper parameterization in both parameter models and data models. However, the presence of the MAR covariates necessitates extra adjustments to the parameterization of the data models, including the outcome and covariate models. This is because the accurate calculation of cluster probabilities relies on the accurate development of data models. In this section, we examine the parameter models and data models in depth.

5.4.1 Parameter Model and Inference

Our Dirichlet process log-skewnormal mixture consists of a three-level hierarchical structure: the first level pertains the data models, such as the log-skewnormal $f(S|\boldsymbol{\theta})$ outcome model and the Bernoulli $f(\mathbf{z}|\boldsymbol{w})$, Gaussian $f(\mathbf{x}|\boldsymbol{w})$ covariate models as defined in Equation (5.3); the second level involves the parameter models such as $p(\boldsymbol{\theta}), p(\boldsymbol{w}), p(\boldsymbol{\theta}|S_h, \mathbf{X}_h)$, and $p(\boldsymbol{w}|\mathbf{X}_h)$, which explain the data with uncertainty; the third level is a group of hyperparameters to set a probabilistic distribution on the parameter vectors $\boldsymbol{\theta} = \{\boldsymbol{\beta}, \sigma^2, \xi, \tilde{\boldsymbol{\beta}}\}$, $\boldsymbol{w} = \{\pi, \mu, \lambda^2\}$, and the precision α. Please refer to Variable Definition in Appendix A for further information on the variables used in this thesis.

The parameter models are specified as follows: Given the model definition in Equation (5.2), we first determine a set of conjugate priors due to its analytical efficiency (Cairns et al. 2011). Considering the types of the data, for the zero-inflated log-skewnormal outcome - $S_h \sim \delta(\mathbf{X}_h^T\tilde{\boldsymbol{\beta}}_j)\ \mathbb{1}(S_h = 0) + \big[1 - \delta(\mathbf{X}_h^T\tilde{\boldsymbol{\beta}}_j)\big]\ \mathbf{LogSN}(\mathbf{X}_h^T\boldsymbol{\beta}_j, \sigma_j^2, \xi_j)$ -, the discrete covariates - $z_h \sim \mathbf{Bernoulli}(\pi_j)$ -, and the continuous covariate - $x_h \sim \mathbf{N}(\mu_j,\ \lambda_j^2)$ -, their prior models come in as

$$\left.\begin{aligned} p_0(\boldsymbol{\beta}_j|\boldsymbol{\beta}_0,\ \Sigma_{\beta_0}): &\quad \mathbf{MVN}(\boldsymbol{\beta}_0,\ \sigma_j^2\Sigma_{\beta_0}) \\ p_0(\sigma_j^2|u_0,\ v_0): &\quad \mathbf{InvGa}(u_0,\ v_0) \\ p_0(\xi_j|\nu_0): &\quad \mathbf{t}(\nu_0) \\ p_0(\tilde{\boldsymbol{\beta}}_j|\tilde{\boldsymbol{\beta}}_0,\ \tilde{\Sigma}_{\beta_0}): &\quad \mathbf{MVN}(\tilde{\boldsymbol{\beta}}_0,\ \tilde{\Sigma}_{\beta_0}) \end{aligned}\right\} \text{for outcome } S_h \tag{5.14}$$

$$\left.\begin{aligned} p_0(\pi_j|g_0,\ h_0): &\quad \mathbf{Beta}(g_0,\ h_0) \\ p_0(\mu_j|\mu_0,\ \lambda_j^2): &\quad \mathbf{N}(\mu_0,\ \lambda_j^2) \\ p_0(\lambda_j^2|c_0,\ d_0): &\quad \mathbf{InvGa}(c_0,\ d_0) \end{aligned}\right\} \text{for covariates } \mathbf{X}_h \tag{5.15}$$

$$\left.\begin{aligned} p_0(\alpha|\gamma_0,\ \psi_0): &\quad \mathbf{Ga}(\gamma_0,\ \psi_0) \end{aligned}\right\} \text{for precision} \tag{5.16}$$

and their corresponding kernels used in this chapter are listed in E.2.2 in Appendix E. Accordingly, the probability measure G_0 of the DP prior can be defined as $G_0 =$

$\mathbf{MVN}(\boldsymbol{\beta}_0,\ \sigma_j^2\Sigma_{\beta_0}) \times \mathbf{InvGa}(u_0, v_0) \times \mathbf{t}(\nu_0) \times \mathbf{MVN}(\tilde{\boldsymbol{\beta}}_0,\ \tilde{\Sigma}_{\beta_0}) \times \mathbf{Beta}(g_0,\ h_0) \times \mathbf{N}(\mu_0,\ \lambda_j^2) \times \mathbf{InvGa}(c_0,\ d_0) \times \mathbf{Ga}(\gamma_0,\ \psi_0)$, representing the product of all models specified in Equations (5.14) to (5.16).

With the observed data inputs $\{S_h, z_h, x_h\}$ for $h = 1, \cdots, H$, the priors for the covariates $\boldsymbol{w}_j = \{\pi_j, \mu_j, \lambda_j^2\}$ and precision α in Equations (5.15) and (5.16) (excluding the prior models for the outcome parameter $\boldsymbol{\theta}_j = \{\boldsymbol{\beta}_j, \sigma_j^2, \xi_j, \tilde{\boldsymbol{\beta}}_j\}$ in Equation (5.14) due to their lack of closed forms), can be transformed into the following posteriors analytically.

$$\left.\begin{aligned} p(\pi_j|g_0,\ h_0,\ \mathbf{z}) &: \mathbf{Beta}(g_{new},\ h_{new}) \\ p(\mu_j|\mu_0,\ \lambda_j^2,\ \mathbf{x}) &: \mathbf{N}(\mu_{0\,new},\ \lambda_{j\,new}^2) \\ p(\lambda_j^2|c_0,\ d_0,\ \mathbf{x}) &: \mathbf{InvGa}(c_{new},\ d_{new}) \end{aligned}\right\} \text{for covariates } \mathbf{X}_h \tag{5.17}$$

$$\left.\begin{aligned} p(\alpha|\gamma_0,\ \psi_0,\ H,\ J,\ \eta,\ \pi_\eta) &: \pi_\eta \mathbf{Ga}(\gamma_0 + J,\ \psi_0 - log(\eta)) \\ &+(1-\pi_\eta)\mathbf{Ga}(\gamma_0 + J - 1,\ \psi_0 - log(\eta)) \end{aligned}\right\} \text{for precision} \tag{5.18}$$

in which their corresponding parameterizations - $g_{new}, h_{new}, \mu_{0\,new}, \lambda_{j\,new}^2, c_{new}, d_{new}$ - are elaborated in E.2 in Appendix E. Note that the value of the precision α depends on the total cluster number J, and does not vary according to the cluster membership j. Therefore, the process of deriving its posterior parameterization is not determined by the principle of Bayesian conjugacy. Hence, we instead adapt the form of the posterior density for the α suggested by Escobar and West 1995, and its derivation is provided in E.2.1 in Appendix E. Similarly, for the outcome parameter $\boldsymbol{\theta}_j = \{\boldsymbol{\beta}_j, \sigma_j^2, \xi_j, \tilde{\boldsymbol{\beta}}_j\}$, there are no available conjugate priors for the log-skewnormal likelihood either. However, their posterior samples can be obtained using the conventional Metropolis-Hastings (MH) algorithm described in Algorithm (E.2.3) in Appendix E. Considering the case of MAR covariate (given that $\mathbf{z}$ contains missing data), the parameterizations of the posterior densities for the covariate parameter model of $\boldsymbol{w}$ in Equation (5.17) and the precision α in Equation (5.18) remain unaffected. However, when computing the posterior for the outcome param-

eters $\boldsymbol{\theta}$ in Algorithm (E.2.3), any observations with the MAR covariate z_h needs to be excluded in the sampling process because n_j and $\mathbf{z}$ are determined by the complete observations in cluster j. Accordingly, the major component $f(S_h|\mathbf{X}, \boldsymbol{\theta}_j)$ of the transition ratio in the MH algorithm in Algorithm (E.2.3) (to produce the posterior samples) is re-defined with the imputed covariate z_h, which is elaborated in E.1.1 in Appendix E. Once the parameters are updated with the imputation, the data models can be constructed according to Equations (5.10) and (5.11).

5.4.2 Data Model and Clustering

Data models - the covariate and outcome models - are the main components for cluster probability computations in [**Stage.1**] '**Re-assigning cluster memberships with Gibbs sampler**' depicted in Figure 5.1. Similar to the construction of parameter models, the covariate model of $\mathbf{X}$ excludes observations with the missing covariate, while the outcome model of S_h requires the completion of the covariates beforehanE. However, the formulation of their densities can be more complex due to the marginalization (integration over the missing variable) process with respect to the missing covariate. Furthermore, as elaborated in Section 5.3.2, the data model development is restrained by the types of clusters such as discrete clusters $f(S_h|\mathbf{X}_h, \boldsymbol{\theta}_j),\ f(\mathbf{X}_h|\boldsymbol{w}_j)$ and continuous clusters $f_0(S_h|\mathbf{X}_h),\ f_0(\mathbf{X}_h)$:

(a) **covariate model for the 'discrete cluster':** $f(\mathbf{X}_h|\boldsymbol{w}_j)$

Focusing on the scenario that $\mathbf{z}$ is binary, $\mathbf{x}$ is Gaussian, and the only covariate with missingness is z_h, we simply drop the covariate z_h to develop the covariate model for the discrete cluster. For instance, when computing the covariate probability term for hth observation in j cluster, the covariate model $f(z_h, x_h|\pi_j, \mu_j, \lambda_j^2)$ simply becomes $f(x_h|\mu_j, \lambda_j^2)$ due to the missingness of z_h. As we have $\mathbf{x}$ that is assumed to be normally distributed as defined in

Equation (5.3), its probability term is

$$f(x_h|\mu_j, \lambda_j^2) = \frac{1}{\sqrt{2\pi\lambda_j^2}} \exp\{\frac{-(x_h - \mu_j)^2}{2\lambda_j^2}\} \tag{5.19}$$

instead of

$$f(z_h, x_h|\pi_j, \mu_j, \lambda_j^2) = \pi_j^{z_h} (1 - \pi_j)^{1-z_h} \cdot \frac{1}{\sqrt{2\pi\lambda_j^2}} \exp\{\frac{-(x_h - \mu_j)^2}{2\lambda_j^2}\}$$

(b) **covariate model for the 'continuous cluster':** $f_0(\mathbf{X}_h)$

If the binary covariate z_h is missing, by the same logic, we drop the covariate z_h for the continuous cluster; however, using Equation (5.5), the covariate model for the continuous cluster integrates out the relevant parameters simulated from the DP prior G_0 as follows:

$$\begin{aligned} f_0(x_h) &= \int f(x_h|\mu, \lambda^2)\, dG_0(\mu, \lambda^2) = \int f(x_h|\mu, \lambda^2) \cdot p(\mu|\lambda^2) \cdot p(\lambda^2)\, d\mu\, d\lambda^2 \\ &= \frac{d_0^{c_0}\Gamma(c_0 + 1/2)}{2\sqrt{\pi}\Gamma(c_0)} \Big(d_0 + \frac{(x_h^2 - \mu_0)^2}{4}\Big)^{-(c_0+1/2)} \end{aligned} \tag{5.20}$$

instead of

$$\begin{aligned} f_0(z_h, x_h) &= \int f(z_h, x_h|\pi, \mu, \lambda^2) \cdot p(\pi) \cdot p(\mu|\lambda^2) \cdot p(\lambda^2)\, d\pi\, d\mu\, d\lambda^2 \\ &= \frac{\boldsymbol{B}(z_h + g_0,\ 1 - z_h + h_0)}{\boldsymbol{B}(g_0, h_0)} \cdot \frac{d_0^{c_0}\Gamma(c_0 + 1/2)}{2\sqrt{\pi}\Gamma(c_0)} \Big(d_0 + \frac{(x_h^2 - \mu_0)^2}{4}\Big)^{-(c_0+1/2)} \end{aligned}$$

The derivation of the distributions above is provided in E.1.2 in Appendix E.

(c) **outcome model for the 'discrete cluster':** $f(S_h|\mathbf{X}_h, \boldsymbol{\theta}_j)$

In developing the outcome model, similar to the parameter model case discussed in Section 5.4.1 and E.1.2 in Appendix E, it should be ensured that the covariate is complete beforehand. With all missing data in z_h imputed, the outcome model for the discrete cluster is obtained by marginalizing the joint

- $f(S_h, z_h|x_h, \boldsymbol{\theta}_j, \pi_j)$ - out the MAR covariate z_h, which is a log skew-normal mixture as follows:

$$
\begin{aligned}
&f(S_h|x_h, \boldsymbol{\beta}_j, \sigma_j^2, \xi_j, \tilde{\boldsymbol{\beta}}_j) = \sum_{z_h=0}^{1} f(S_h|z_h, x_h, \boldsymbol{\beta}_j, \sigma_j^2, \xi_j, \tilde{\boldsymbol{\beta}}_j) \cdot f(z_h|\pi_j) \\
&= f(S_h,\ z_h = 1|x_h, \boldsymbol{\beta}_j, \sigma_j^2, \xi_j, \tilde{\boldsymbol{\beta}}_j, \pi_j) + f(S_h,\ z_h = 0|x_h, \boldsymbol{\beta}_j, \sigma_j^2, \xi_j, \tilde{\boldsymbol{\beta}}_j, \pi_j) \\
&= \delta(\mathbf{X}_h^T\tilde{\boldsymbol{\beta}}_j)\ \mathbb{1}(S_h = 0) + \left[1 - \delta(\mathbf{X}_h^T\tilde{\boldsymbol{\beta}}_j)\right] \cdot \frac{2}{\sigma_j S_h} \\
&\cdot \phi\left(\frac{\log S_h - (\beta_{j0} + \beta_{j1} + \beta_{j2}x_h)}{\sigma_j}\right) \cdot \Phi\left(\xi_j \frac{\log S_h - (\beta_{j0} + \beta_{j1} + \beta_{j2}x_h)}{\sigma_j}\right) \pi_j \\
&+\ \delta(\mathbf{X}_h^T\tilde{\boldsymbol{\beta}}_j)\ \mathbb{1}(S_h = 0) + \left[1 - \delta(\mathbf{X}_h^T\tilde{\boldsymbol{\beta}}_j)\right] \cdot \frac{2}{\sigma_j S_h} \\
&\cdot \phi\left(\frac{\log S_h - (\beta_{j0} + \beta_{j2}x_h)}{\sigma_j}\right) \cdot \Phi\left(\xi_j \frac{\log S_h - (\beta_{j0} + \beta_{j2}x_h)}{\sigma_j}\right) \cdot (1 - \pi_j)
\end{aligned}
\tag{5.21}
$$

instead of

$$
\begin{aligned}
&f(S_h|z_h, x_h, \boldsymbol{\beta}_j, \sigma_j^2, \xi_j, \tilde{\boldsymbol{\beta}}_j) \\
&= \delta(\mathbf{X}_h^T\tilde{\boldsymbol{\beta}}_j)\ \mathbb{1}(S_h = 0) + \left[1 - \delta(\mathbf{X}_h^T\tilde{\boldsymbol{\beta}}_j)\right] \cdot \frac{2}{\sigma_j S_h} \\
&\cdot \phi\left(\frac{\log S_h - (\beta_{j0} + \beta_{j1}z_h + \beta_{j2}x_h)}{\sigma_j}\right) \cdot \Phi\left(\xi_j \frac{\log S_h - (\beta_{j0} + \beta_{j1}z_h + \beta_{j2}x_h)}{\sigma_j}\right)
\end{aligned}
$$

(d) **outcome model for the 'continuous cluster':** $f_0(S_h|\mathbf{X}_h)$

Once the missing covariate $\mathbf{z}$ is fully imputed and the outcome model is marginalized over the MAR covariate z_h, the outcome model $f_0(S_h|x_h)$ for the continuous cluster can also be computed by integrating out the relevant parameters, using Equation (5.5).

$$
f_0(S_h|x_h) = \int f(S_h|x_h, \boldsymbol{\beta}, \sigma^2, \xi, \tilde{\boldsymbol{\beta}}) \cdot p(\boldsymbol{\beta}) \cdot p(\sigma^2) \cdot p(\xi) \cdot p(\tilde{\boldsymbol{\beta}})\ d\boldsymbol{\beta}\ d\sigma^2\ d\xi\ d\tilde{\boldsymbol{\beta}}
\tag{5.22}
$$

However, it can be too complicated to compute its form analytically. Instead, we can integrate the joint model out the parameters, using Monte Carlo integration. For example, we can do the following for each $h = 1, \cdots, H$.

(i) Sample $\boldsymbol{\beta}, \sigma^2, \xi, \tilde{\boldsymbol{\beta}}$ from the DP prior densities G_0 specified previously.

(ii) Input these samples into $f(S_h|x_h, \boldsymbol{\beta}, \sigma^2, \xi, \tilde{\boldsymbol{\beta}}) \cdot p(\boldsymbol{\beta}) \cdot p(\sigma^2) \cdot p(\xi) \cdot p(\tilde{\boldsymbol{\beta}})$.

(iii) Record the output and repeat the above steps many times.

(iv) Divide the sum of all output values by the number of Monte Carlo samples, which will approximate the complex integral.

5.5 Numerical Experiments with MAR Covariate

5.5.1 Data: PnCdemand + LGPIF

We evaluate the performance of our DPM model using two insurance datasets, which present a specific set of challenges such as unobservable heterogeneity (RQ1.1) in the log-normal outcome variable (RQ1.2) and MAR covariates (RQ2.1). For simplicity, each dataset includes only two covariates - binary $\mathbf{z}$ and continuous $\mathbf{x}$ - to elucidate aggregate claim information S_h (outcome variable). Throughout this study, all computations on these two datasets adhere to the same data format given by

$$
\begin{array}{lc}
 & Year_1 \quad Year_2 \quad \cdots, \quad Year_y \\
\text{Policy } (h=a): & \{(S_a,\ \mathbf{X}_a),\ (S_a,\ \mathbf{X}_a),\ \cdots, (S_a,\ \mathbf{X}_a)\} \\
\text{Policy } (h=b): & \{(S_b,\ \mathbf{X}_b),\ (S_b,\ \mathbf{X}_b),\ \cdots, (S_b,\ \mathbf{X}_b)\} \\
 & \vdots \\
\text{Policy } (h=H): & \{(S_H, \mathbf{X}_H),\ (S_H, \mathbf{X}_H),\ \cdots, (S_H, \mathbf{X}_H)\}
\end{array}
$$

The first dataset is **PnCdemand**, which pertains to the International Property and Liability Insurance Demand across 22 countries over a span of 7 years from 1987 to 1993. Additionally, we utilize the dataset sourced from the Wisconsin Local Government Property Insurance Fund (LGPIF) from the previous chapter, containing details regarding insurance coverage for government building units in Wisconsin spanning the years 2006 to 2010.

PnCdemand can be obtained through the publicly available R package **CASdatasets**. It comprises a relatively modest sample size, containing $H = 240$ cases. The outcome variable, denoted as *GenLiab*, represents the individual claim amount Y_h under general insurance policies for each case (a single claim per policy). Regarding covariates, we incorporate one binary variable indicating the statutory law system (*LegalSyst*: 1/0), along with one continuous variable quantifying risk aversion value (*RiskAversion*) for each case. For additional background on this dataset, see Browne et al. 2000.

In the LGPIF dataset, we have insurance coverage data for government properties from $H = 5,660$ policies. The outcome variable represents the aggregate claim amount S_h, denoted as *Total Losses*, for each policy (a sum of multiple claims per policy). Our study focuses on two covariates: *LnCoverage*, and *Fire5*. The latter is a binary covariate that indicates fire-protection levels, while *LnCoverage* is a continuous covariate that denotes a total coverage amount on a logarithmic scale. For further details, see Quan and Valdez 2018. Histograms illustrating the

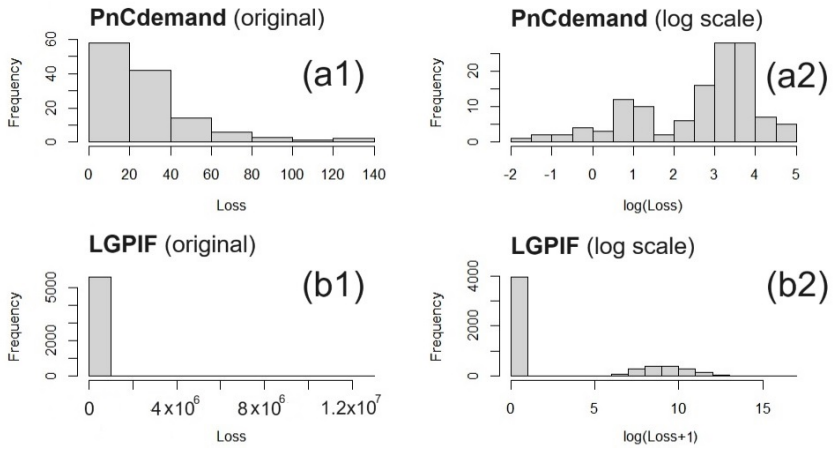

Figure 5.5: Histograms of the original outcomes and log-transformed outcomes for the two datasets: (a) **PnCdemand**, (b) LGPIF.

aggregate claims from both datasets are presented in Figure 5.5. Due to the significant skewness, the outcome values are log-transformed to achieve normality. Each distribution exhibits distinct characteristics regarding skewness, modality, excess of zeros, etc. It is noteworthy that the LGPIF data includes a zero-inflated outcome variable (illustrated by b1 and b2 in Figure 5.5), necessitating a two-part model-

ing approach to separately address the probabilities of the outcome being zero and positive, respectively.

5.5.2 Implementation

We compare our Data Augmentation-based DPM model with other commonly used actuarial models in practice. As benchmarks, we utilize three predictive models - a Generalized Linear Mixture model (GLM), Multivariate Adaptive Regression Spline (MARS), Generalized Additive model (GAM) - and Multiple Imputation (MI), which are elaborated in Chapter 2. In each dataset, we assume different distributions for the outcome variables, leading to the construction of the benchmark models based on distinct outcome data models. For instance, in the **PnCdemand** dataset (depicted as a1, a2 in Figure 5.5), which predominantly consists of small claim amounts with no zero values, we opt for a gamma mixture to model the outcome data. Conversely, for the LGPIF data (illustrated as b1, b2 in Figure 5.5), a Tweedie distribution is considered for the outcome data model to accommodate the zero-inflated aggregate claim amounts. The benchmark models are implemented in R utilizing the **mgcv**, **splines**, and **mice** packages.

All four models are trained, and investigations are conducted regarding their model fit (AIC), prediction accuracy (SSPE, SAPE), and the conditional tail expectation (CTE) of the predictive distribution. It is important to note that the goodness-of-fit value for the DPM is not presented in Table 5.1,5.2. According to Teh 2010, assessing the goodness of fit for the DPM is thought unnecessary. This is because underfitting is mitigated by the unbounded complexity of the DPM, while overfitting is addressed by the approximation of posterior densities over each parameter in the DPM. Since the competing models used in this experiment - GLM, MARS, GAM - are based on the frequentist framework, their performance is evaluated using the Akaike Information Criterion (AIC). AIC is a widely accepted metric for model comparison in the frequentist paradigm, as it balances model fit and complexity by penalizing the number of estimated parameters (Hastie et al. 2009).

5.5.3 Results with PnCdemand ($H = 240$)

For this dataset, we create a training set comprising 160 records of response and covariate pairs $(Y, \mathbf{X})$, along with a test set containing 80 records of response and covariate pairs $(Y', \mathbf{X}')$. We implement the following Dirichlet Process Log-Normal Mixture (DPLNM):

$$\begin{aligned} Y_h|z_h, x_h, \boldsymbol{\beta}_j, \sigma_j^2 &\sim \mathbf{LogN}\big(\mathbf{X}_h^T\boldsymbol{\beta}_j,\ \sigma_j^2\big) \\ z_h|\pi_j &\sim \mathbf{Bernoulli}\big(\pi_j\big) \\ x_h|\mu_j, \lambda_j^2 &\sim \mathbf{N}\big(\mu_j,\ \lambda_j^2\big) \\ \{\boldsymbol{\theta}_j,\ \boldsymbol{w}_j\} &\sim G \\ G &\sim \mathbf{DP}\big(\alpha,\ G_0\big) \end{aligned} \tag{5.23}$$

We select a log-normal likelihood to handle the individual claim amount Y_h:*GenLiab* for a policy h. The covariate $\mathbf{x}$:*RiskAversion* is subject to missingness, which is dependent on Y_h (a MAR case). To address this, we employ an internalized imputation process as portrayed in Figure 5.2. The posterior parameters of $\boldsymbol{\theta}_j, \boldsymbol{w}_j$ are then estimated using our DPM Gibbs sampler outlined in the Algorithm (E.3) in Appendix E and Section 5.3.4. The algorithm iterates sufficient times (e.g., M=100,000 iterations) until its log-likelihood converges.

The resulting clustering mixture scenarios are displayed in Figure 5.6. This plot shows the overlays of predictive densities on the log scale from the last 100 iterations, which are indicative of convergence. Figure 5.7 presents the classical data imputation results utilizing *Multiple Imputation by Chained Equations* (MICE), alongside the predictive densities developed by our competing models - GLM, GAM, MARS built on the imputed dataset. The MICE runs multiple imputation chains and selects the imputation values from the final iteration, producing multiple candidate datasets. The trace plots (a1, a2) monitor the imputation mean and variance for the missing values in the dataset. In the covariate distribution plot (a3), the density of the observed covariate, shown in blue, is compared with the densities of the imputed

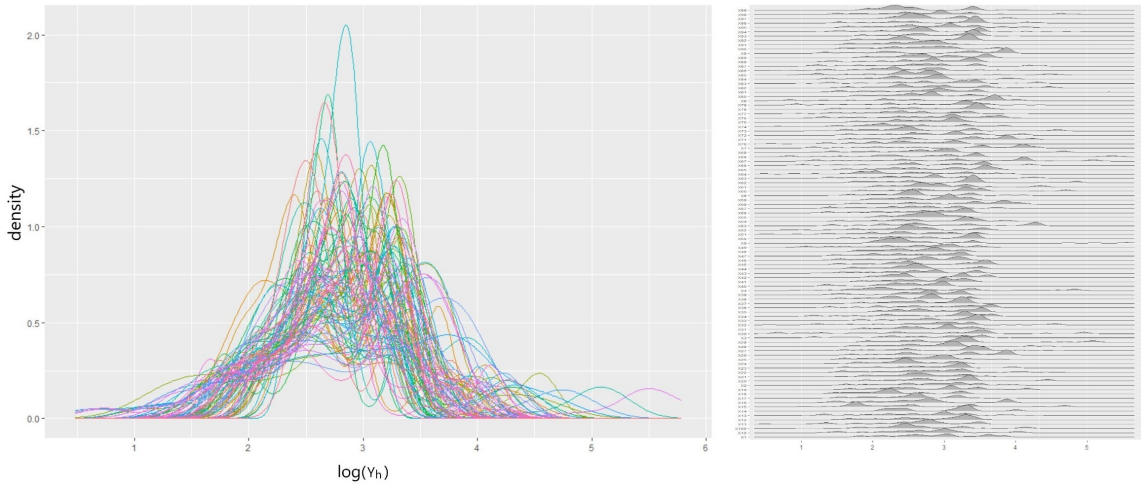

Figure 5.6: Our model: Data Augmentation-based DPLNM with the **PnCdemand** dataset: The last 100 in-sample predictive densities (scenarios) overlaid together.

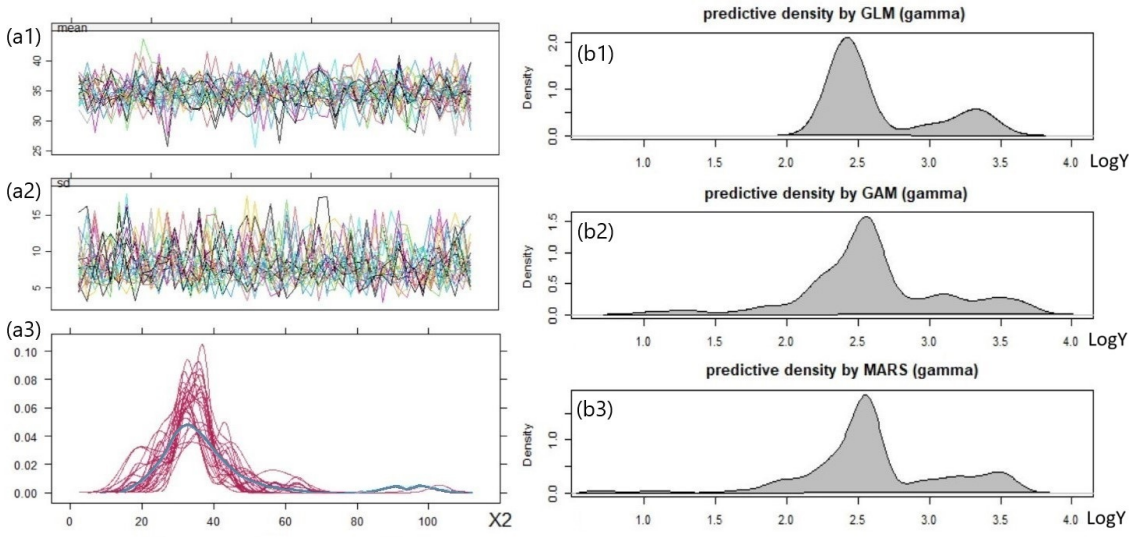

Figure 5.7: Rival models: MICE-based GLM, GAM, MARS with the **PnCdemand** dataset. MICE trace plots (a1,a2), the imputation comparison plot (a3), and in-sample predictive densities (b1,b2,b3) produced from GLM, GAM, MARS.

covariate for each imputed dataset, shown in red. Parameter inferences for the rival models - GLM, GAM, MARS - are then performed based on the imputed datasets that reach convergence (Shah et al. 2014). The gamma distribution is chosen to fit the rival models as the Y_h is continuous and positively skewed with a constant coefficient of variation (Boland 2006). The predictive density plots based on the gamma distribution (b1, b2, b3) estimated with GLM, GAM, and MARS exhibit similar patterns, characterized by noticeable irregularities near the right tail.

In Figure 5.8, a histogram of the outcome data - $\log Y_h$ - in the test set is

presented. Overlaid on the histogram are the posterior mean densities for out-of-sample predictions generated by our DPM, as well as density estimates from the rival models. Upon examination of the plot, it appears that our DPM model provides the most accurate approximation. While the rival models produce smooth, mound-shaped curves for predictions, our DPM captures all discernible peaks and bumps, offering a closer match to the actual data distribution.

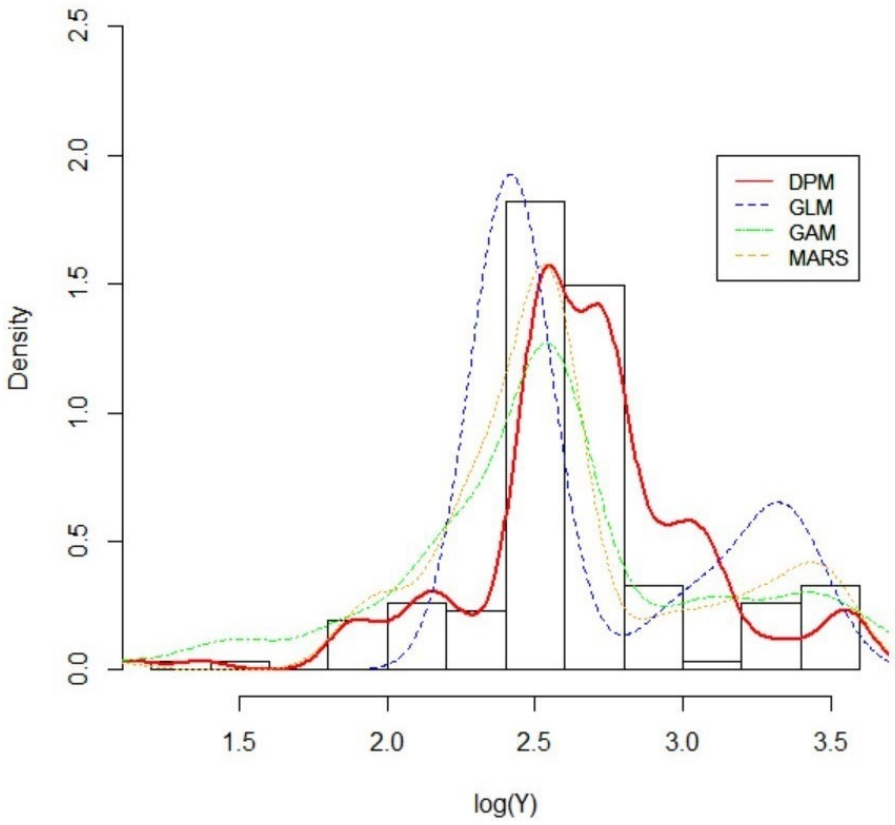

Figure 5.8: All together: a histogram of the observed claim amount Y_h on the log scale and the out-of-sample predictive densities for the typical class of a policy h in the **PnCdemand** dataset.

Moving on to Table 5.1, our DPM exhibits the highest SSPE among the rival models, which might initially appear as a drawback. However, upon closer examination, it becomes evident that the presence of outliers significantly influences its performance. Interestingly, our DPM excels at capturing these outliers, leading to the highest SSPE. This is supported by the lowest SAPE observed in our DPM. In essence, SAPE treats all individual differences equally, suggesting that rival models might overly focus on the most probable data points while missing the majority of

outliers. This could be attributed to insufficient sample size. However, our DPM demonstrates robust performance even with small sample sizes, provided there is ample prior knowledge available. Regarding CTE, Table 5.1 shows that our DPM proposes a heavier tail compared to the rival models, implying that it captures more uncertainties given the limited sample size.

Model	**AIC**	**SSPE**	**SAPE**	**10% CTE**	**50% CTE**	**90% CTE**	**95% CTE**
Ga-GLM	830.6	268.6	139.8	6.5	13.8	54.5	78.0
Ga-MARS	830.6	267.2	138.2	6.1	13.0	57.2	71.1
Ga-GAM	845.9	266.7	136.1	6.2	13.3	58.1	72.2
DPLNM	-	272.0	134.7	6.4	13.8	59.3	79.3

Table 5.1: All together: the comparison of out-of-sample modeling results based on the dataset **PnCdemand**.

5.5.4 Results with LGPIF ($H = 5,660$)

For this dataset, a training set of response and covariates pair $(S, \mathbf{X})$ with 4,529 records, and a test set of response and covariates pair $(S', \mathbf{X}')$ with 1,110 records are constructed. We implement the following Dirichlet Process Log-Skewnormal Mixture (DPLSM):

$$
\begin{aligned}
S_h|z_h,&x_h, \boldsymbol{\beta}_j, \sigma_j^2, \xi_j, \tilde{\boldsymbol{\beta}}_j \\
&\sim \delta(\mathbf{X}_h^T\tilde{\boldsymbol{\beta}}_j)\ \mathbb{1}(S_h = 0) + \left[1 - \delta(\mathbf{X}_h^T\tilde{\boldsymbol{\beta}}_j)\right] \mathbf{LogSN}\big(\mathbf{X}_h^T\boldsymbol{\beta}_j,\ \sigma_j^2,\ \xi_j\big) \\
z_h|\pi_j &\sim \mathbf{Bernoulli}\big(\pi_j\big) \\
x_h|\mu_j, \lambda_j^2 &\sim \mathbf{N}\big(\mu_j,\ \lambda_j^2\big) \\
\{\boldsymbol{\theta}_j,\ \boldsymbol{w}_j\} &\sim G \\
G &\sim \mathbf{DP}\big(\alpha,\ G_0\big)
\end{aligned}
\tag{5.24}
$$

In this dataset, the outcome S_h (*Total Losses*) for a policy h is considered as a sum of log-normal densities. To approximate this convolution, we opt for a log-

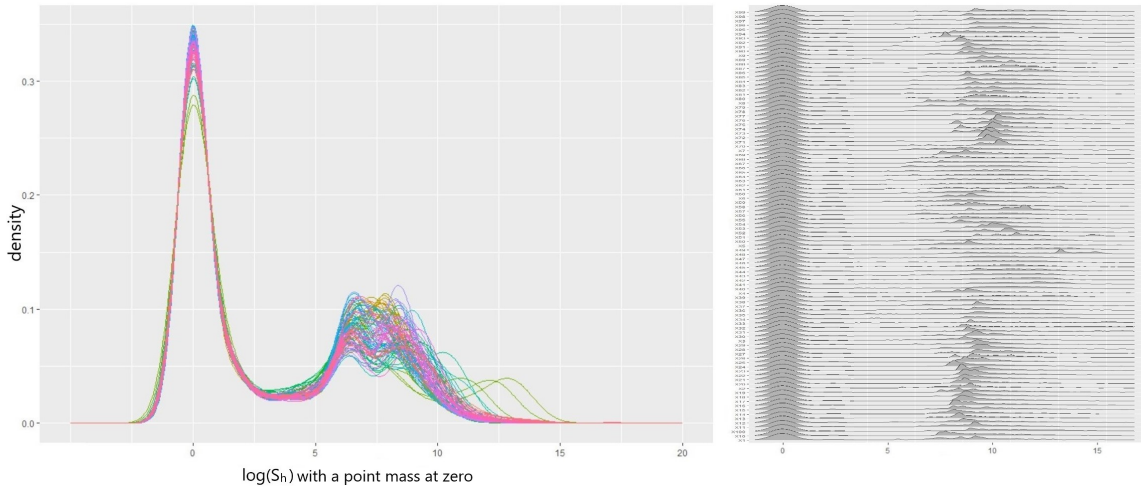

Figure 5.9: Our model: Data Augmentation-based DPLSM with the LGPIF dataset: The last 100 in-sample predictive densities (scenarios) overlaid together.

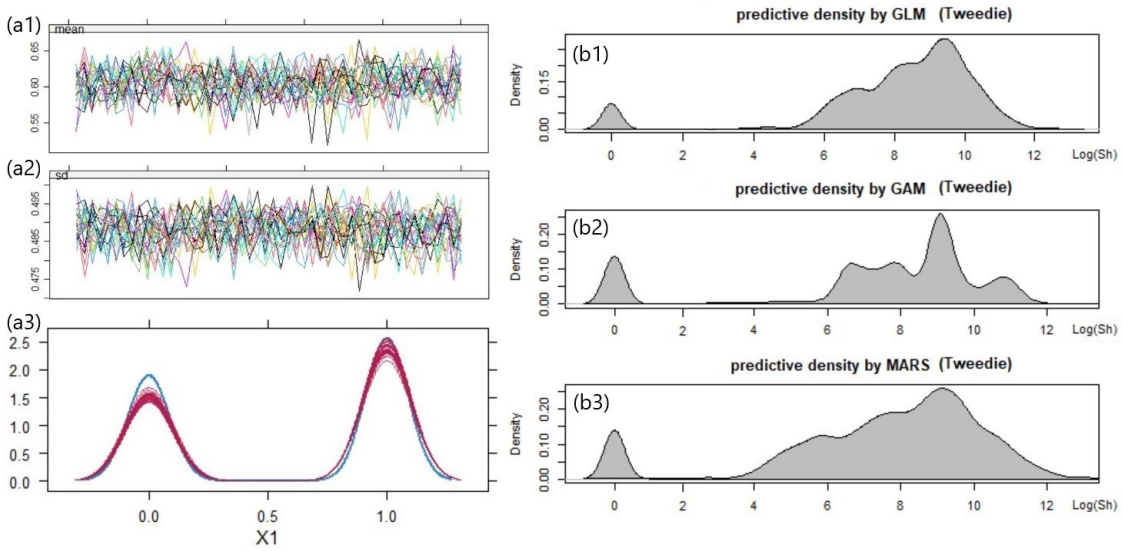

Figure 5.10: Rival models: MICE-based GLM, GAM, MARS with the LGPIF dataset. MICE trace plots (a1,a2), the imputation comparison plot (a3), and in-sample predictive densities (b1,b2,b3) produced from GLM, GAM, MARS.

skewnormal likelihood (Li 2008). The covariate $\mathbf{z}$ (*Fire5*) is subject to missingness under MAR assumption, and we address this issue through the internalized imputation process depicted in Figure 5.2, avoiding the multiplication of imputed datasets. Given the zero inflation observed in the outcome S_h, we adopt a two-part model using a sigmoid and indicator function. Our DPM Gibbs sampler, detailed in Algorithm (E.3) in Appendix E, iterates sufficient times (e.g., M=100,000 iterations) until convergence to derive the posterior parameters of $\boldsymbol{\theta}_j, \boldsymbol{w}_j$. The resulting scenarios of clustering mixture are illustrated in Figure 5.9, showcasing 100 predictive

densities suggested by our DPM, each representing the convergence of the estimation results. Figure 5.10 presents the output of the MICE process and the resulting predictive densities from the rival models. These rival models are constructed based on a Tweedie distribution, chosen for its capability to accommodate a significant number of zero values and its flexibility in capturing claim amount patterns among various classes of policyholders. From the plot, it is evident that all three rival models adequately capture zero inflation. However, the GAM model suggests additional bumps, indicating the need for further evaluation of prediction uncertainty.

The overall out-of-sample prediction comparison is depicted in Figure 5.11, where the histogram of $\log S_h$ is overlaid with predictive density curves generated from our model and three rival models. It is evident from the plot that the posterior predictive density proposed by our DPM provides the most accurate explanation for the new samples, while the rival models continue to produce multiple peaks.

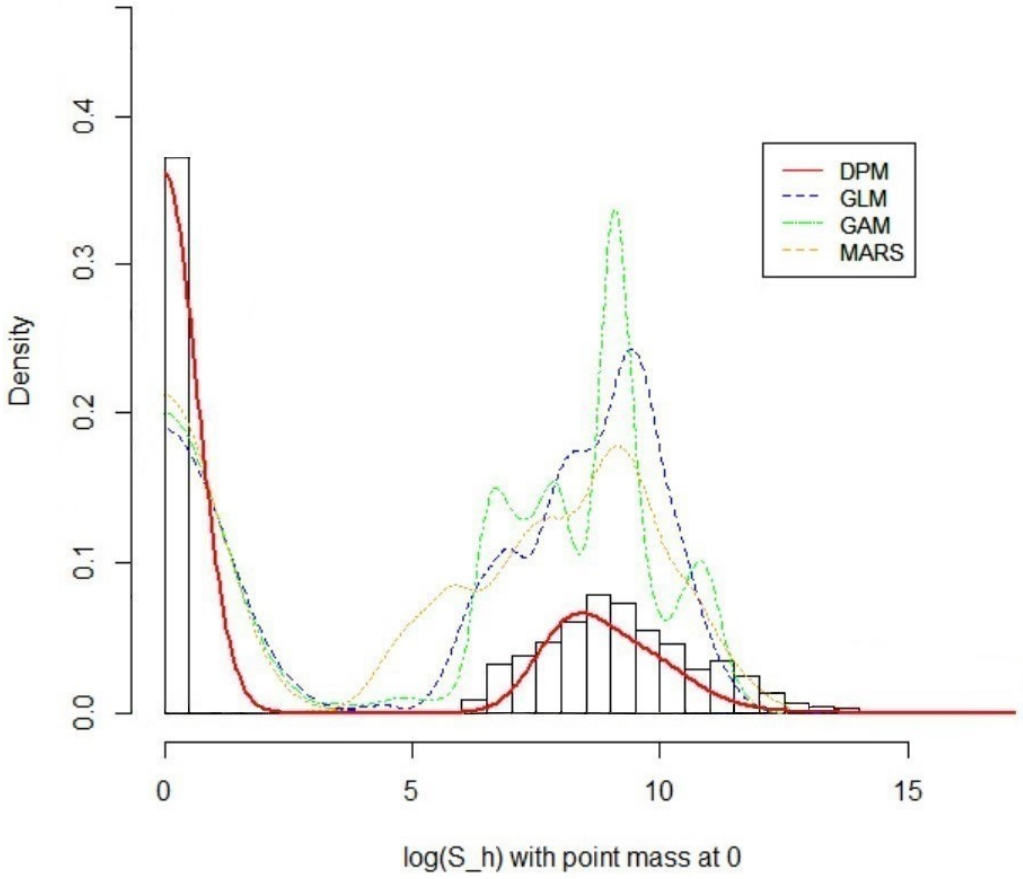

Figure 5.11: All together: a histogram of the observed aggregate claim amount S_h on the log scale and the out-of-sample predictive densities for the typical class of a policy h in the LGPIF dataset.

The improved prediction performance of our DPM is corroborated by its smallest SAPE in Table 5.2. However, in terms of SSPE, our DPM demonstrates the second-highest performance, slightly trailing behind the GAM. This discrepancy can be attributed to our DPM's ability to capture outliers more effectively (which is particularly useful for risk assessment) but is penalized heavily by SSPE due to the squared term. Regarding CTE, all three rival models suggest a similar level of tail behavior, reflecting the knowledge derived solely from the observed data. In contrast, our DPM extends beyond this by proposing a much heavier tail, as it accommodates the presence of outliers and shapes the tail behavior based on a combination of prior parameters and available observations.

Model	**AIC**	**SSPE** $\times 10^{12}$	**SAPE** $\times 10^{6}$	**10% CTE**	**50% CTE**	**90% CTE**	**95% CTE**
Tweedie-GLM	26,270	204	89.3	955.9	12,977	133,374	340,713
Tweedie-MARS	24,721	199	88.5	961.7	10,391	129,409	355,112
Tweedie-GAM	21,948	195	88.2	989.4	13,026	140,199	398,263
DPLSM	-	198	83.8	975.3	13,695	147,486	425,682

Table 5.2: All together: The comparison of out-of-sample modeling results based on the LGPIF dataset.

5.5.5 Discussion

This chapter introduced a novel Dirichlet process log-skewnormal mixture model for risk premium prediction, incorporating MAR covariates. Throughout the numerical experiments, both the log-normal and log-skewnormal DPM models consistently exhibited robust empirical performance in capturing the intricate shapes of claim distributions, achieving accurate out-of-sample predictions, and estimating tails effectively. These results implied that adopting our DPM model could effectively mitigate model risk related to heterogeneity, convolution error, and MAR covariate considerations.

Regarding **RQ1.1** (heterogeneity), this chapter implemented a non-parametric

Bayesian modeling approach, and addressed within-cluster heterogeneity arising from the inclusion of both complete and incomplete covariates. By accommodating an infinite number of clustering scenarios determined by observations and prior knowledge, our DPM model outperformed rival methods in handling cluster membership as a latent variable. The homogeneity of resulting clusters was assessed by fitting cluster-wise GLMs (gamma and Tweedie distributions) and comparing goodness-of-fit, with consistent AICs across all clusters endorsing the advantages of the DPM. While rival methods like GAM or MARS can capture heterogeneity using customized smooth functions across different data subsets, we observed statistically insignificant smooth terms, indicating the presence of heterogeneity in clusters.

For **RQ1.2** (convolution error), we fitted a log-skewnormal density to aggregate claim amount outcomes. To evaluate its performance, various approximation techniques like minimax, least squares, and log-shifted gamma can be considered as competitors. Li 2008 offered a comparative analysis of these techniques using simulated log-normal data with predefined parameters and assumptions. However, real-world scenarios lack such controlled conditions, making the choice of the best approximation technique reliant on dataset-specific characteristics. In our case, the magnitude differences among summands in our dataset rendered the minimax approach unsuitable, while the large volume of data smaller than five in the LGPIF dataset made the log-shifted gamma approach inappropriate. Therefore, we opted for the log-skewnormal density, which offered a relatively simple yet accurate approximation, particularly in the lower region of the distribution.

In addressing **RQ2.1** (MAR covariate), we incorporated a data augmentation process into the parameter and cluster membership update within the DPM Gibbs sampler. This was achieved by leveraging the joint distribution of observed outcomes and missing covariates, ensuring the consistency of imputed values with the observed data and preserving the dataset's correlation structure. To compare our approach with an existing alternative, we also employed a Multiple Imputation (MI) algorithm. Investigation into multiple sets of imputed values from both approaches

revealed that our DPM Gibbs sampler did not significantly outperform the MI algorithm. However, we attribute this result largely to the relatively low dimensionality and simple structure of the datasets used in our study. Real-world datasets with more complex characteristics and dependencies in the data, as well as intricate missing patterns, may yield different results.

In the next chapter, we will continue our examination of the Dirichlet Process Log-Skewnormal Mixture (DPLSM) model for predicting risk premiums. Our focus will shift from addressing model risk related to missing covariates, which are assumed to be Missing At Random (MAR), to managing the model risk posed by mismeasured covariates, particularly those classified under Non-Differential Berkson (NDB) error. This transition will enable us to tackle the complexities associated with poor-quality covariates in a more comprehensive manner and enhance the robustness of our risk premium predictions.

Chapter 6

Bayesian Nonparametric II: DPM with NDB Covariate

6.1 Introduction: RQ1.1, RQ1.2, RQ1.3, RQ2.2

Chapter 3 has outlined how the expected total claim $E[S_h]$ for a policy h can be derived from two different viewpoints:

- Frequency-Severity approach for a policy h in Equation (3.1a)
- Compound approach for a policy h in Equation (3.1b)

Continuing from Chapter 5, the 'Compound' principle to risk premium estimation is further investigated in this chapter, assuming a high correlation between claim counts N_h and amounts Y_{hi}. Especially, this chapter not only considers the expected aggregate claim for a policy h: $E[S_h] = \sum_{i=1}^{N_h} E[Y_{hi}] = E[Y_{h1}] + \ldots + E[Y_{hN_h}]$, but also extends this consideration to determine the expected total aggregate claims for a portfolio (entire policies $h = 1, \cdots H$): $E[\tilde{S}] = \sum_{h=1}^{H} E[S_h] = E[S_1] + \ldots + E[S_H]$. For the covariate $\mathbf{X} = \{\mathbf{x}, \mathbf{z}\}$, similar to Chapter 4, the focus remains on the continuous covariate $\mathbf{x}^*$, which is mismeasured and classified as Non-Differential Berkson (NDB). Previously, we highlighted that incorporating the NDB covariate gives rise to several types of model risk, such as data heterogeneity (RQ1.1) and the NDB covariate itself (RQ2.2), both of which bias the curve estimation for $S_h|\mathbf{X}$ within

a policy h. Moreover, when scaling up to a portfolio level, an additional source of model risk emerges from convolution development (RQ1.2), due to the dependence of the convolution on the covariate, as discussed in Section 3.2.2. Additionally, the increased sample size from this scaling further complicates Bayesian computation, making it more complex and computationally intensive (RQ1.3). All of these issues together lead to $E[S_h|\mathbf{X}] \neq E[Y_{h1}|\mathbf{X}_{h1}] + E[Y_{h2}|\mathbf{X}_{h2}] \ldots + E[Y_{hN_h}|\mathbf{X}_{hN_h}]$ and $E[\tilde{S}|\mathbf{X}] \neq E[S_1|\mathbf{X}_1] + E[S_2|\mathbf{X}_2] + \ldots + E[S_H|\mathbf{X}_H]$.

To accurately construct the aggregate claim prediction curve for a given policy h, as well as the total aggregate predictions across multiple policies $h = 1, \cdots, H$, a hybrid method needs to be considered. In the previous chapters, we explored how the Gustafson correction, integrated with a DPM as a Bayesian Nonparametric method, addresses complex heterogeneity issues and mitigates bias in parameter estimation induced by the NDB covariate. In this chapter, we aim to extend and unify the proposed BNP strategy for managing these multifaceted model risks — RQ1.1 heterogeneity, RQ1.2 convolution error, RQ2.2 NDB covariate — to large-scale implementations (RQ1.3 scalability) in order to develop a scalable solution for both individual policy-level (with $S_h|\mathbf{X}$) and portfolio-level (with $\tilde{S}|\mathbf{X}$) risk assessment.

6.2 Our Contribution

This chapter builds on the Dirichlet Process Log-Skewnormal Mixture (DPLSM) model developed in Chapter 5, proposing a novel BNP strategy for risk premium modeling at both individual policy level and portfolio level. The primary contribution of this chapter is the integration of the BNP model and the Gustafson correction technique, previously studied in Chapter 4, into a comprehensive risk premium modeling framework. This framework is designed to simultaneously tackle multiple model risks posed by RQ1.1 RQ1.2, and RQ2.2, but also enhances the scalability of the models (RQ1.3), allowing them to be applied to larger and more complex datasets, thereby enhancing the robustness of risk premium prediction. To the best of our knowledge, this work represents the first comprehensive application

of the DPLSM model, combined with the Gustafson correction technique, within the domain of insurance risk premium modeling. Accordingly, all the analytical derivations for the key modeling components, alongside the novel design of hybrid Gibbs samplers to ensures efficient posterior sampling, are presented here for the first time.

The performance of the Gustafson correction is evaluated within both hierarchical GLM (Bayesian Parametric) and DPM (Bayesian Nonparametric) frameworks for a comprehensive comparison. Empirical results from applying the Gustafson correction in both frameworks demonstrate significant advantages of the DPM framework over the hierarchical GLM framework. By fitting our DPLSM model to multiple insurance datasets (introduced in Section 6.4.1), spanning from a small sample size of $H \leq 2,000$ to a larger sample size of $H \geq 50,000$, we also demonstrate significant effectiveness of our DPLSM model in mitigating complex model risks associated with the NDB covriate. This underscores its potential for broad applicability in insurance risk assessment, especially in scenarios where practitioners face the issue of covariate mismeasurement due to NDB errors at a large scale.

6.3 Modeling Method for $S_h|\mathbf{X}$ and $\tilde{S}|\mathbf{X}$

6.3.1 Clustering $S_h|\mathbf{X}$ with Complete Case Covariate

Simialar to Section 4.3.1 in Chapter 4, this section begins with the assumption that the covariates are accurately measured, but the model risk associated with the heterogeneity issue (RQ1.1) emerges, disrupting the homogeneity within each insured risk cluster. As discussed in Section 2.1 and 3.1, this disruption poses significant challenges to ensuring fair and accurate risk premium modeling. Chapter 4 employs the partial pooling technique using the hierarchical Bayesian framework to achieve homogeneous risk clusters under that assumption that risk clusters $j = 1, \cdots J$ are already found and fixed. However, aligned with Chapter 5, this chapter advances further by adopting a more practical assumption that the risk clusters $j = 1, \cdots J$

cannot be known or defined prior to the analysis. This makes the partial pooling technique irrelevant, as it relies on predefined clusters. Instead, the focus shifts to parameter-free clustering technique embedded in the DPM framework, which allows for the emergence of brand-new clusters that capture the unexplained variability within the data and form multiple clustering scenarios that best accommodate the underlying variability. In order to enhance understanding of the distribution selections for both the outcome and covariates in this chapter, and to shed light on the underlying rationale, a summary table can be found in the Appendix F.

Baseline DPM with parameter-free clustering: Suppose we have an aggregate claim amount outcome S_h for a policy h that is log-skewnormally distributed, and we have two covariates: $\mathbf{z}$ (binary) and $\mathbf{x}$ (continuous), none of which have mismeasured values. Let the outcome $S = \{S_{h=1}, S_{h=2}, \ldots, S_{h=H}\}$ represent the H different aggregate claims incurred by the H different policies. We consider the following initial model configuration for the DPLSM to employ the parameter-free clustering:

$$S_h|z_h, x_h, \boldsymbol{\beta}_j, \sigma_j^2, \xi_j \sim \mathbf{LogSN}\big(\mathbf{X}_h^T\boldsymbol{\beta}_j,\ \sigma_j^2,\ \xi_j\big) \tag{6.1a}$$

$$z_h|\pi_j \ \sim\ \mathbf{Bernoulli}\big(\pi_j\big) \tag{6.1b}$$

$$x_h|\lambda_j^2 \ \sim\ \mathbf{N}\big(E[\mathbf{x}_j],\ \lambda_j^2\big) \tag{6.1c}$$

$$\boldsymbol{\phi}_j : \{\boldsymbol{\theta}_j,\ \boldsymbol{w}_j,\ \boldsymbol{\omega}_j(\mathbf{X}_h)\} \ \sim\ G \tag{6.1d}$$

$$G \ \sim\ \mathbf{DP}\big(\alpha,\ G_0\big) \tag{6.1e}$$

$$\text{where } \boldsymbol{\omega}_j(\mathbf{X}_h) : \begin{cases} \boldsymbol{\omega}_{J+1}^{(*)}(\mathbf{X}_h) = \dfrac{\alpha}{\alpha + H} \cdot f_0(x_h, z_h) & \text{for continuous clusters} \\ \boldsymbol{\omega}_j^{(*)}(\mathbf{X}_h) = \dfrac{n_j}{\alpha + H} \cdot f(x_h, z_h|\boldsymbol{w}_j) & \text{for discrete clusters} \end{cases}$$

where: j is the risk cluster index; $\mathbf{X}_h = \{z_h, x_h\}$ for the covariates; $\boldsymbol{\theta}_j = \{\boldsymbol{\beta}_j, \sigma_j^2, \xi_j\}$ for parameters describing the outcome; $\boldsymbol{w}_j = \{\pi_j, \lambda_j^2\}$ for parameters describing the covariates; and $\boldsymbol{\omega}_j(\mathbf{X}_h)$ for the covariate-dependent mixing weight, satisfying $\sum_{j=1}^{\infty} \boldsymbol{\omega}_j(\mathbf{X}_h) = 1$. The established (discrete) clusters and brand-new (continuous) clusters, characterized by $\boldsymbol{\phi}_j$, result from a parameter-free clustering algorithm, leading to variability in G (clustering scenario) at each iteration. The

samples (clusters) from G are assigned an array of covariate-based mixing weights $\boldsymbol{\omega}_1^{(*)}(\mathbf{X}_h), \boldsymbol{\omega}_2^{(*)}(\mathbf{X}_h), \cdots, \boldsymbol{\omega}_{J+1}^{(*)}(\mathbf{X}_h)$, developed using a generalized stick-breaking process to enhances homogeneity within risk clusters. Since J cannot be pre-defined, the mixing weight can temporarily take two different states — $\boldsymbol{\omega}_{J+1}^{(*)}(\mathbf{X}_h)$ and $\boldsymbol{\omega}_j^{(*)}(\mathbf{X}_h)$ — thus facilitating parameter-free clustering (Sethuraman 1994).

The distributional forms for the joint covariate models to define the mixing weight $\boldsymbol{\omega}_j(\mathbf{X}_h)$ are given by:

$$
\begin{aligned}
f(x_h, z_h|\boldsymbol{w}_j) &= f(x_h|\lambda_j^2) \cdot f(z_h|\pi_j) && \text{(6.2a)}\\
&= \frac{1}{\sqrt{2\pi\lambda_j^2}} \exp\{\frac{-(x_h - \bar{\mathbf{x}}_j)^2}{2\lambda_j^2}\} \cdot \pi_j^{z_h} \left(1-\pi_j\right)^{1-z_h} \\
f_0(x_h, z_h) &= \int f(x_h, z_h|\pi_j, \lambda_j^2) \cdot p_0(\pi) \cdot p_0(\lambda^2)\, d\pi\, d\lambda^2 && \text{(6.2b)}\\
&= \frac{\boldsymbol{B}(z_h + g_0,\ 1 - z_h + h_0)}{\boldsymbol{B}(g_0, h_0)} \times \frac{1}{\sqrt{\pi}} \cdot \frac{\Gamma(\frac{c_0+1}{2})}{\Gamma(\frac{c_0}{2})} \cdot \frac{d_0^{c_0/2}}{\left[(\mathbf{x} - \bar{\mathbf{x}})^2 + d_0\right]^{\frac{c_0+1}{2}}}
\end{aligned}
$$

See F.1.4 in Appendix F for the detailed derivation of the joint covariate model. Note that the covariate effect is integrated with the outcome model in Equation (6.1a) as well as the mixing weight in Equation (6.1d) to develop the mixture of clusters. By incorporating covariate effects into the mixing weights, more adaptive clustering is facilitated, dynamically capturing the variability in the outcome based on the covariate values, and leading to more interpretable clusters. Accordingly, the predictive value[1] for a policy h based on the DPLSM model emerges from the weighted average of the continuous and discrete clusters described below with the following prior and posterior samples:

$$
\begin{aligned}
&E[S_h|\mathbf{X}, \boldsymbol{\theta}, \boldsymbol{w}, \alpha] \\
&= \underbrace{\frac{\boldsymbol{\omega}_{J+1}^{(*)}(\mathbf{X})}{\boldsymbol{\omega}_{J+1}^{(*)}(\mathbf{X}) + \sum_{j=1}^{J} \boldsymbol{\omega}_j^{(*)}(\mathbf{X})} \cdot E_0[S_h|\mathbf{X}]}_{\text{For the new continuous cluster}} + \underbrace{\frac{\sum_{j=1}^{J} \boldsymbol{\omega}_j^{(*)}(\mathbf{X}) \cdot E[S_h|\mathbf{X}, \boldsymbol{\theta}_j]}{\boldsymbol{\omega}_{J+1}^{(*)}(\mathbf{X}) + \sum_{j=1}^{J} \boldsymbol{\omega}_j^{(*)}(\mathbf{X})}}_{\text{For the established discrete clusters}} \qquad \text{(6.3)}
\end{aligned}
$$

[1]The expression of the predictive value of the log-skewnormal outcome is based on the first moment driven by the moment generating function (Azzalini 2013).

$$\text{where} \begin{cases} E[S_h|\mathbf{X}, \boldsymbol{\theta}_j] = 2\exp\Big(\mathbf{X}^T\boldsymbol{\beta}_j + \frac{1}{2}\sigma_j^2\Big)\Phi\Big(\frac{\sigma_j\xi_j}{\sqrt{1+\xi_j^2}}\Big) \text{ with posterior samples } \boldsymbol{\theta}_j \\ E_0[S_h|\mathbf{X}] = \frac{2}{M}\sum_{m=1}^{M}\exp\Big(\mathbf{X}^T\boldsymbol{\beta}_{(m)} + \frac{1}{2}\sigma_{(m)}^2\Big)\Phi\Big(\frac{\sigma_{(m)}\xi_{(m)}}{\sqrt{1+\xi_{(m)}^2}}\Big) \text{ with prior samples } \boldsymbol{\theta}_{(m)} \end{cases}$$

$$\text{For } S_h \begin{cases} \boldsymbol{\beta}_{(m)} \mid \boldsymbol{\beta}_0, \Sigma_{\beta_0} \sim \mathbf{MVN}(\boldsymbol{\beta}_0,\ \sigma_{(m)}^2\Sigma_{\beta_0}) \\ \sigma_{(m)}^2 \mid u_0, v_0 \sim \mathbf{InvGa}(\frac{u_0}{2}, \frac{v_0}{2}) \\ \xi_{(m)} \mid \nu_0 \sim \mathbf{t}(\nu_0) \end{cases} \tag{6.4a}$$

$$\text{For } \mathbf{X} \begin{cases} z_h \sim \mathbf{Bernoulli}(\pi_{(m)}) \\ \pi_{(m)} \mid g_0, h_0 \sim\ \mathbf{Beta}(g_0, h_0) \\ x_h \sim \mathbf{N}(E[\mathbf{x}_j],\ \lambda_{(m)}^2) \\ \lambda_{(m)}^2 \mid c_0, d_0 \sim\ \mathbf{InvGa}(\frac{c_0}{2}, \frac{d_0}{2}) \end{cases} \tag{6.4b}$$

$$\text{For precision} \qquad \alpha_{(m)} \mid \gamma_0, \psi_0 \sim\ \mathbf{Ga}(\gamma_0, \psi_0) \tag{6.4c}$$

and their corresponding kernels used in this chapter are listed in F.2.1 in Appendix F. Given these prior specifications in Equation (6.4), the probability measure G_0 of the DP prior can be defined as $G_0 = \mathbf{MVN}(\boldsymbol{\beta}_0,\ \sigma_j^2\Sigma_{\beta_0}) \times \mathbf{InvGa}(u_0/2, v_0/2) \times \mathbf{t}(\nu_0) \times \mathbf{Beta}(g_0,\ h_0) \times \mathbf{InvGa}(c_0/2,\ d_0/2) \times \mathbf{Ga}(\gamma_0,\ \psi_0)$, representing the product of all models specified in Equation (6.4). These priors are carefully developed by considering computational expedience to obtain their posteriors and the desired characteristics of the clusters, such as shape, tail thickness, etc. For instance, we assign a multivariate Gaussian prior to the regression coefficients for $\mathbf{X}^T\boldsymbol{\beta}_j$ that describes the mean of the outcome data S_h such that the coefficient vector $\boldsymbol{\beta}_j$ favors $\boldsymbol{\beta}_0$ with a variance matrix Σ_{β_0}. This can be a natural prior choice for the regression coefficients (see Gelman and Hill 2007). The scale parameter σ_j^2 is assumed to follow an inverse gamma density with the choice of the hyperparameters — $u_0/2, v_0/2$ — to give a *unit-information prior*[2], which is due to its simple posterior computation (see Kass and Wasserman 1995). For the shape parameter ξ_j, we consider a Student's t distri-

[2]The unit-information prior is a weakly informative prior designed to minimize the influence of hyperparameter choices, ensuring the posterior is primarily informed by the data without the prior dominating the inference process (Gustafson 2008).

bution to overcome the unbounded likelihood problem pointed out by Eling 2012. As suggested by Bayes and Branco 2007, the centered and fat-tailed shape of the t distribution can promote sparseness, which leads to a good approximation of the Jeffreys prior that is governed by the invariant property under reparameterization.

Given the observed data inputs $\{S_h, z_h, x_h\}$ for $h = 1, \cdots, H$, the priors for the covariates $\boldsymbol{w}_j = \{\pi_j, \lambda_j^2\}$ and precision α in Equations (6.4) can be analytically transformed into the following posteriors:

$$\left.\begin{array}{r} \pi_j|g_0,\ h_0,\ \mathbf{z} \sim \mathbf{Beta}\Big(g_0 + \Sigma_{h=1}^{n_j} z_h, \quad h_0 + n_j - \Sigma_{h=1}^{n_j} z_h\Big) \\ \lambda_j^2|c_0,\ d_0,\ \mathbf{x} \sim\ \mathbf{InvGa}\Big(\frac{n_j+c_0}{2}, \quad \frac{1}{2}\Big[\Sigma_{h=1}^{n_j}(x_h - \bar{\mathbf{x}})^2 + d_0\Big]\Big) \end{array}\right\} \text{for covariates } \mathbf{X}_h \tag{6.5a}$$

$$\left.\begin{array}{r} \alpha|\gamma_0,\ \psi_0,\ H,\ J,\ \eta,\ \pi_\eta \sim \pi_\eta \mathbf{Ga}(\gamma_0 + J,\ \psi_0 - log(\eta)) \\ +(1-\pi_\eta)\mathbf{Ga}(\gamma_0 + J - 1,\ \psi_0 - log(\eta)) \end{array}\right\} \text{for precision} \tag{6.5b}$$

which provides parameter inputs for mixing weight computation in Equation (6.1) and (6.2). Note that, in Equation (6.5a), the choice of the unit-information prior to update λ_j^2 diminishes the impact of the hyperparameters c_0 and d_0 as the information such as n_j or $x_h - \bar{\mathbf{x}}$ obtained from the data becomes dominant in shaping the posterior. The derivations of the posterior parameterization in Equation (6.5) are elaborated in F.2.3 in Appendix F. The posterior samples for the log-skewnormal outcome parameters $\boldsymbol{\theta}_j = \{\boldsymbol{\beta}_j, \sigma_j^2, \xi_j\}$ can be computed through the standard Metropolis-Hastings (MH) algorithm procedure, as outlined in Algorithm (F.2.2) in Appendix F, because there are no available conjugate priors for the log-skewnormal likelihood. Once the parameters are fully updated, they are then substituted into Equation (6.3) to generate the predictive values, the expected aggregate claim amount under the model. This procedure ensures that the predictions, based on the parameter-free clustering, are informed by the most current posterior samples, effectivley accounting for the inherent heterogeneity issue (RQ1.1).

6.3.2 Clustering $\tilde{S}|\mathbf{X}$ with Complete Case Covariate

In this chapter, we broaden our focus from predicting the aggregate claim amount $S_h|\mathbf{X}$ for a policy h to predicting the total aggregate claim amount $\tilde{S}|\mathbf{X}$ across all the policies in a portfolio. This shift allows for a more holistic evaluation of the insurer's overall financial exposure by considering the cumulative effect of all claims within the portfolio. However, this expanded scope introduces a more complex problem: the convolution of conditional log-skewnormal random variables (outlined in RQ1.2). To address this, we here elaborate the process of calculating the convolution for the total aggregate claim amount $\tilde{S}|\mathbf{X}$.

Overall, we hypothesize that $\tilde{S}|\mathbf{X} \approx H \times E[S_h|\mathbf{X}, \boldsymbol{\theta}, \boldsymbol{w}, \alpha]$ for $h = 1, \cdots, H$ with a sufficiently large sample size H, even though S_h is conditioned on $\mathbf{X}$ and log-skewnormally distributed. If this hypothesis holds, approximating the distribution of the total aggregate claim amount $\tilde{S}|\mathbf{X}$ requires only the predicted value of the individual aggregate claim amount $E[S_h|\mathbf{X}, \boldsymbol{\theta}, \boldsymbol{w}, \alpha]$ for a policy h. This belief is grounded in the following reasons:

- To approximate the distribution $f(S_1 + \cdots + S_H|\mathbf{X})$ where $\mathbf{X}$ indicates general covariate inputs, *Lindeberg's CLT* from Section 3.2.2 informs us that, if H is sufficiently large, the distribution converges to $\mathbf{N}\Big(\Sigma_{h=1}^{H} E[S_h|\mathbf{X}], \quad \mathbb{S}_H^2\Big)$, provided that the variance $\mathbb{S}_H^2$ for these summations is finite (i.e. its spread is limited). This convergence condition is proven in Appendix A, and we propose that $E[S_h|\mathbf{X}, \boldsymbol{\theta}, \boldsymbol{w}, \alpha]$ for a policy h can be obtained from our DPM model.

- In the distribution $f(S_1 + \cdots + S_H|\mathbf{X})$, each summand S_h is conditioned on the same covariates $\mathbf{X}$, leading to the expression: $E[S_1 + \cdots + S_H|\mathbf{X}_{(a)}] = E[\tilde{S}|\mathbf{X}_{(a)}] = E[S_1|\mathbf{X}_{(a)}] + \cdots + E[S_H|\mathbf{X}_{(a)}]$. Since each summand $E[S_h|\mathbf{X}_{(a)}]$ is based on the same input $\mathbf{X}_{(a)}$, all summands are identical, and their value can be estimated using our DPM model. As a result, $E[\tilde{S}|\mathbf{X}_{(a)}]$ is equal to $H \times E[S_h|\mathbf{X}_{(a)}, \boldsymbol{\theta}, \boldsymbol{w}, \alpha]$. Now, we have a collection of such samples for $f(S_1 +$

$\cdots + S_H|\mathbf{X}$):

$$\begin{aligned} f(S_1 + \cdots + S_H|\mathbf{X}_{(a)}) &\rightsquigarrow \mathbf{N}\Big(\Sigma_{h=1}^{H} E[S_h|\mathbf{X}_{(a)}, \boldsymbol{\theta}, \boldsymbol{w}, \alpha], \quad \mathbb{S}^2_{H(a)}\Big) \\ f(S_1 + \cdots + S_H|\mathbf{X}_{(b)}) &\rightsquigarrow \mathbf{N}\Big(\Sigma_{h=1}^{H} E[S_h|\mathbf{X}_{(b)}, \boldsymbol{\theta}, \boldsymbol{w}, \alpha], \quad \mathbb{S}^2_{H(b)}\Big) \\ &\vdots \end{aligned} \tag{6.6}$$

Here, the mean of each distribution $\Sigma_{h=1}^{H} E[S_h|\mathbf{X}, \boldsymbol{\theta}, \boldsymbol{w}, \alpha]$ in Equation (6.6) is translated into $E[\tilde{S}|\mathbf{X}] = H \cdot E[S_h|\mathbf{X}, \boldsymbol{\theta}, \boldsymbol{w}, \alpha]$.

Hence, our focus can return to the essential task of obtaining the accurate predictions $E[S_h|\mathbf{X}_{(a)}, \boldsymbol{\theta}, \boldsymbol{w}, \alpha],\ E[S_h|\mathbf{X}_{(b)}, \boldsymbol{\theta}, \boldsymbol{w}, \alpha], \cdots$ at a policy level to develop the predictive curve for the total aggregate claims $\tilde{S}|\mathbf{X}$ at a portfolio level.

6.3.3 Clustering $S_h|\mathbf{X}$ with Parallel Simulations to Scale

In Chapter 4 and 5, our models have been built on the claim data at moderate sample size (such as $H \leq 6,000$). The posterior computations for the Bayesian models at small or moderate sample sizes are relatively stable. Running our DPM model on larger observations ($H \geq 50,000$) is, however, hardly feasible due to limitations of computing power for MCMC simulation (RQ1.3). To scale our DPM model to a bigger dataset, per Section 3.2.3, we adopt the parallel MCMC simulation approach explored by Ni et al. 2020. The core idea behind this approach is to split a large dataset into random subsets — a.k.a *shards* — which are then distributed across multiple machines for independent parallel MCMC sampling. Once each machine has produced its set of posterior samples and corresponding clusters, these clusters are merged to complete the posterior inference.

The problem arises when our Bayesian model utilizes a mixture structure that relies on the resulting clusters to model the irregularly shaped target densities. As different MCMC chains progress across the various shards, they tend to switch between different possible labelings of the clusters, leading to a phenomenon known as *label-switching* (Jasra et al. 2005). While label-invariance does not impede mean-

ingful inference on a single global parameter, it becomes problematic when the focus is on cluster-specific parameters collected from multiple different shards. The arbitrary re-labeling of clusters across shards makes it difficult to consistently distinguish between clustering results and draw meaningful interpretations about their specific characteristics (Celeux et al. 2000). To ensure that the clustering results from the different shards can be seamlessly combined, this thesis employs the *anchor and shard* technique, discussed in Chapter 3. This technique involves introducing a small number of common observations, denoted as anchor points, shared across all shards. With these anchor points, the labels associated with cluster-specific parameters can be aligned across shards, effectively resolving the label-invariance problem when the MCMC samples are aggregated.

Section 3.2.3, Algorithm (F.4) in Appendix F, and Figure 6.1 provide the details of the parallel MCMC simulation. To begin, following determination of the anchor points (denoted, set $\mathbb{A}$), two essential tasks must be undertaken to ensure the proper setup for the parallel MCMC computations:

- To partition the dataset $\{S_h, \mathbf{X}_h\}$ for $h = 1, \cdots, H$ into the specified number $(N.sh)$ of non-overlapping shards $SH_{(1)}, \cdots, SH_{(N.sh)}$, randomly assign a subset membership $(sh = 1, \cdots, N.sh)$ to each observation in the dataset.

- Using the subset memberships, collect the shard-specific indices of both anchor points and observations, and store them in a new variable (e.g. **shard.index.list** in Algorithm (F.4)).

The above two tasks guarantee that each shard contains a distinct set of non-overlapping observations while ensuring that the anchor points, which should appear in multiple shards, are assigned in a balanced manner. The rest of the process can be broken down into the following two stages:

[**Stage.1**] **Parallel MCMC on multiple shards:** Once all indices of observations and anchor points have been assigned to their respective shards, the data for each shard, including the observations $\{S_h, \mathbf{X}_h\}_{sh}$, anchor points $\{\ddot{S}_h, \ddot{\mathbf{X}}_h\}_{sh}$, and their corre-

sponding cluster memberships, are partitioned accordingly. Within each shard, the Gibbs sampler is run independently, using the partitioned data points and anchor points for that shard. As a result, the shard-specific clusters and posterior parameter estimates are computed. For a more comprehensive explanation, refer to Algorithm (F.4).

[**Stage.2**] **Merging clustering results from multiple shards:** To aggregate the clustering results obtained from each shard after running the individual Gibbs samplers, the process involves five key steps:

Step(1): Load MCMC Results:

Load the MCMC output files for cluster memberships and posterior parameters of each shard $SH_{(sh)}$ into a list. Determine the threshold $\boldsymbol{\varepsilon}$ for measuring cluster similarity between shards.

Step(2): Initialize Binary Matrix:

Create a binary matrix from the first shard $SH_{(1)}$, with rows as observations and columns as cluster memberships, using 1 to indicate membership. This serves as the baseline for comparisons with subsequent shards.

Step(3): Sequential Integration of Shards:

Sequentially gather clustering results for $SH_{(2)}, \cdots, SH_{(N.sh)}$ and compute the cluster similarity distance $\mathbf{Dist}_{(1,sh)}$ using anchor points.

Step(4): Merge or Append Clusters:

Compare $\mathbf{Dist}_{(1,sh)}$ with $\boldsymbol{\varepsilon}$:

- If below $\boldsymbol{\varepsilon}$, merge clusters and recompute parameters.
- If above $\boldsymbol{\varepsilon}$, append clusters to $SH_{(1)}$ without merging.

Expand the binary matrix to reflect new clusters.

Step(5): Refine Binary Matrix:

Ensure each observation belongs to a single cluster by removing duplicate 1s in rows of the binary matrix.

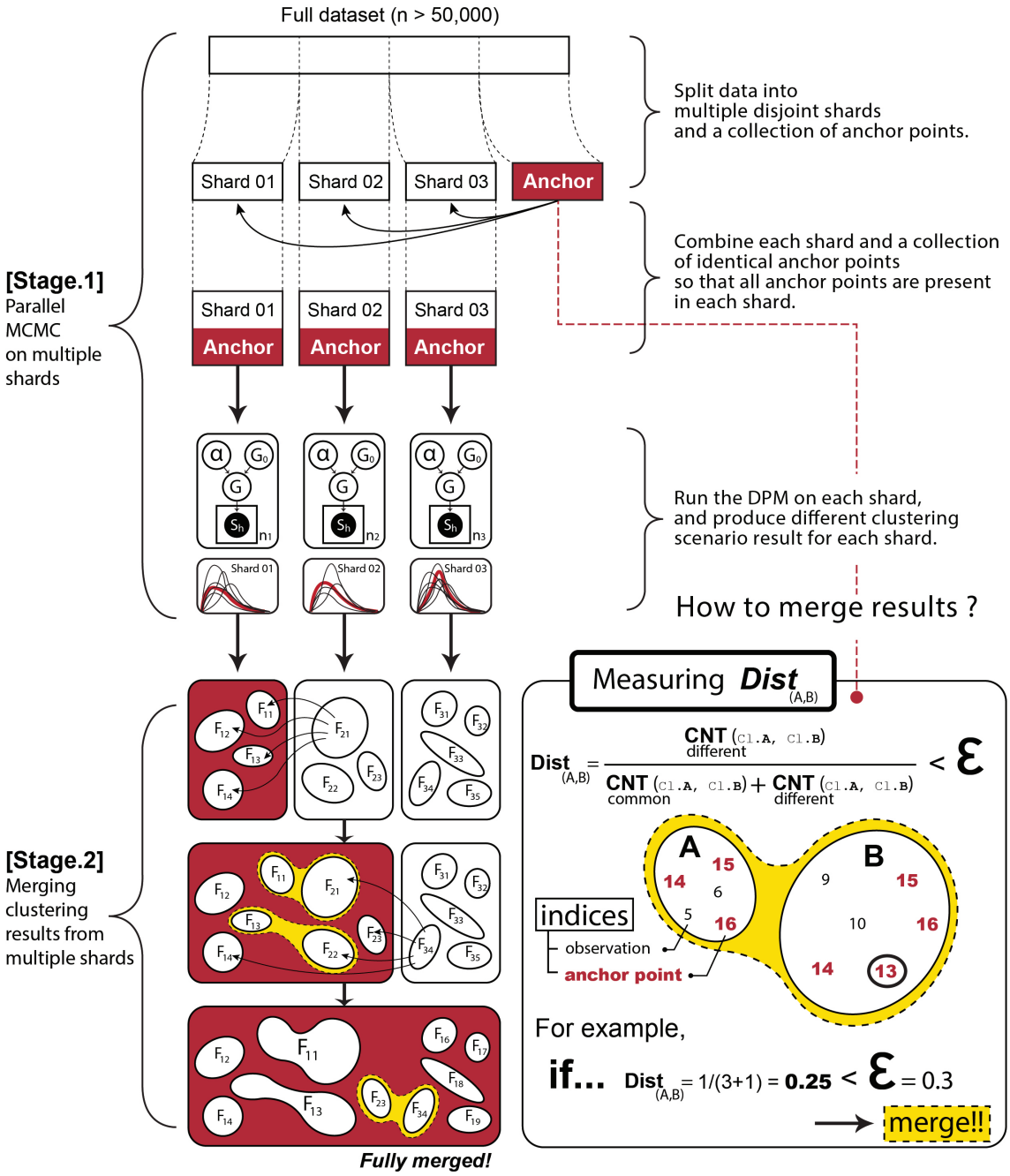

Figure 6.1: Graphical summary of the aggregation process of the clustering results for the large-scale MCMC samples ($n \geq 50,000$) with two stages. The first shard (shard 01 above) continues to grow as the cluster-merging process progresses.

Figure 6.1 provides a concise graphical summary of the two stages discussed earlier. It visually captures the key steps involved in the process. The stages described in Figure 6.1 are also outlined and formalized in Algorithm (F.4) in Appendix F, where each step is translated into actionable procedures.

6.3.4 Clustering $S_h|\mathbf{X}$ with NDB Case Covariate

In this section, we introduce our novel approach to tackling the Non-Differential Berkson (NDB) covariate (outlined in RQ2.2) for risk premium modeling, developed upon the DPLSM with a parameter-free clustering technique. Suppose the continuous covariate $\mathbf{x}^*$ has mismeasured values, with the errors classified as the NDB type. Given the outcome $S = \{S_{h=1}, S_{h=2}, \ldots, S_{h=H}\}$ from H different policies, we consider the following DPLSM formulation:

$$S_h|z_h, x_h^*, \widehat{\boldsymbol{\beta}}_j, \widehat{\sigma}_j^2, \widehat{\xi}_j \sim \mathbf{LogSN}\big(\mathbf{X}_h^T\widehat{\boldsymbol{\beta}}_j,\ \widehat{\sigma}_j^2,\ \widehat{\xi}_j\big) \tag{6.7a}$$

$$z_h|\pi_j \ \sim\ \mathbf{Bernoulli}\big(\pi_j\big) \tag{6.7b}$$

$$x_h^*|x_h \sim \mathbf{N}(x_h,\ \tau_j^2) \tag{6.7c}$$

$$x_h|z_h \sim \mathbf{N}(\kappa_{j0} + \kappa_{j1}z_h,\ \lambda_j^2) \tag{6.7d}$$

$$x_h^*|z_h \sim \mathbf{N}(\widehat{\kappa}_{j0} + \widehat{\kappa}_{j1}z_h,\ \widehat{\lambda}_j^2) \tag{6.7e}$$

$$\boldsymbol{\phi}_j : \{\widehat{\boldsymbol{\theta}}_j,\ \widehat{\boldsymbol{w}}_j,\ \boldsymbol{\omega}_j(\mathbf{X}_h)\} \ \sim\ G \tag{6.7f}$$

$$G \ \sim\ \mathbf{DP}\big(\alpha,\ G_0\big) \tag{6.7g}$$

$$\text{where } \boldsymbol{\omega}_j(\mathbf{X}_h) : \begin{cases} \boldsymbol{\omega}_{J+1}^{(*)}(\mathbf{X}_h) = \dfrac{\alpha}{\alpha + H} \cdot f_0(x_h^*, z_h) & \text{for continuous clusters} \\ \boldsymbol{\omega}_j^{(*)}(\mathbf{X}_h) = \dfrac{n_j}{\alpha + H} \cdot f(x_h^*, z_h|\widehat{\boldsymbol{w}}_j) & \text{for discrete clusters} \end{cases}$$

Note that, due to the absence of the true covariate $\mathbf{x}$, the original covariate model for $x_h|\lambda_j^2$ in Equation (6.1c) becomes hypothetical and is thus replaced by two strategic models, presented in Equations (6.7c) and (6.7d) for $x_h^*|x_h$ and $x_h|z_h$, as suggested by Gustafson 2008. As introduced in Section 4.3.2, the distributions of $x_h^*|x_h$ and $x_h|z_h$ in Equations (6.7c) and (6.7d) serve as the linking component and covariate component respectively, which are integral parts of Gustafson's complete joint model outlined in Equation (3.28).

Accordingly, the joint covariate models for both discrete and continuous clusters are designed to incorporate the NDB covariate effect into the mixing weights $\boldsymbol{\omega}_j(\mathbf{X}_h)$ in Equation (6.7f), which is crucial for performing parameter-free clustering within

the DPM framework. The specifications for the covariate models for both discrete and continuous clusters are given in F.1.3 in Appendix F.

Building upon the DPLSM formulation to accommodate the NDB covariate, we incorporate Gustafson's complete model for NDB covariate in Equation (3.28) into the DPM framework by re-defining the form of the risk premium model with the priors, as originally detailed in Equations (6.3) and (6.4), as follows:

$$
\begin{aligned}
&E[S_h|\mathbf{X},\boldsymbol{\theta},\boldsymbol{w},\alpha] \\
&= \underbrace{\frac{\boldsymbol{\omega}^{(*)}_{J+1}(\mathbf{X})}{\boldsymbol{\omega}^{(*)}_{J+1}(\mathbf{X}) + \sum_{j=1}^{J}\boldsymbol{\omega}^{(*)}_{j}(\mathbf{X})} \cdot E_0[S_h|\mathbf{X}]}_{\text{For the new continuous cluster}} + \underbrace{\frac{\sum_{j=1}^{J}\boldsymbol{\omega}^{(*)}_{j}(\mathbf{X}) \cdot E[S_h|\mathbf{X},\boldsymbol{\theta}_j]}{\boldsymbol{\omega}^{(*)}_{J+1}(\mathbf{X}) + \sum_{j=1}^{J}\boldsymbol{\omega}^{(*)}_{j}(\mathbf{X})}}_{\text{For the established discrete clusters}}
\end{aligned}
\tag{6.8}
$$

$$
\text{where } \begin{cases}
E[S_h|\mathbf{X},\boldsymbol{\theta}_j] = 2\exp\left(\mathbf{X}^T\boldsymbol{\beta}_j + \frac{1}{2}\sigma_j^2\right)\Phi\left(\frac{\sigma_j\xi_j}{\sqrt{1+\xi_j^2}}\right) \text{ with posterior samples } \boldsymbol{\theta}_j \\
E_0[S_h|\mathbf{X}] = \frac{2}{M}\sum_{m=1}^{M}\exp\left(\mathbf{X}^T\boldsymbol{\beta}_{(m)} + \frac{1}{2}\sigma_{(m)}^2\right)\Phi\left(\frac{\sigma_{(m)}\xi_{(m)}}{\sqrt{1+\xi_{(m)}^2}}\right) \text{ with prior samples } \boldsymbol{\theta}_{(m)}
\end{cases}
$$

$$
\text{For } S_h \begin{cases}
\boldsymbol{\beta}_{(m)} \mid \boldsymbol{\beta}_0, \Sigma_{\beta_0} \sim \mathbf{MVN}(\boldsymbol{\beta}_0,\ \sigma^2_{(m)}\Sigma_{\beta_0}) \\
\sigma^2_{(m)} \mid u_0, v_0 \sim \mathbf{InvGa}(\frac{u_0}{2}, \frac{v_0}{2}) \\
\xi_{(m)} \mid \nu_0 \sim \mathbf{t}(\nu_0)
\end{cases} \tag{6.9a}
$$

$$
\text{For } \mathbf{X} \begin{cases}
z_h \sim \mathbf{Bernoulli}(\pi_{(m)}) \\
\pi_{(m)} \mid g_0, h_0 \sim \mathbf{Beta}(g_0, h_0) \\
x^*_h|x_h \sim \mathbf{N}(\mathbf{x}_h,\ \tau^2) \\
\tau^2 \sim \text{undetermined} \\
x_h|z_h \sim \mathbf{N}(\kappa_0 + \kappa_1 z_h,\ \lambda^2_{(m)}) \\
\boldsymbol{\kappa}_{(m)} \mid \tilde{\boldsymbol{\kappa}}, \tilde{\Sigma}_\kappa \sim \mathbf{MVN}(\tilde{\boldsymbol{\kappa}},\ \lambda^2_{(m)}\tilde{\Sigma}_\kappa) \\
\lambda^2_{(m)} \mid c_0, d_0 \sim \mathbf{InvGa}(\frac{c_0}{2}, \frac{d_0}{2})
\end{cases} \tag{6.9b}
$$

$$
\text{For precision} \qquad \alpha_{(m)} \mid \gamma_0, \psi_0 \sim \mathbf{Ga}(\gamma_0, \psi_0) \tag{6.9c}
$$

Note that the final predictive value of the risk premium $E[S_h|\mathbf{X},\boldsymbol{\theta},\boldsymbol{w},\alpha]$ in Equa-

tion (6.8) is computed as a weighted average of two log-skewnormal predictions: $E[S_h|\mathbf{X}, \boldsymbol{\theta}_j]$ based on discrete clusters and posterior samples $\boldsymbol{\theta}_j$, and $E_0[S_h|\mathbf{X}]$ derived from a continuous cluster and prior samples $\boldsymbol{\theta}_{(m)}$. Due to the NDB covariate, the original covariate model of x_h in Equation (6.4b) is replaced by two strategic distributions of $x_h^*|x_h$ and $x_h|z_h$, as shown in Equation (6.9b). Therefore, given the form of the complete joint model: $f(S, \mathbf{x}^*, \mathbf{x}|\mathbf{z}) = f(S|\mathbf{x}, \mathbf{z}) \cdot f(\mathbf{x}^*|\mathbf{x}) \cdot f(\mathbf{x}|\mathbf{z})$, we specify each component as below.

$$f_{LSN}(S|x, z) = \frac{2}{S\sigma_j}\phi\Big(\frac{\log S - (\beta_{j0} + \beta_{j1}x + \beta_{j2}z)}{\sigma_j}\Big)\Phi\Big(\xi_j\frac{\log S - (\beta_{j0} + \beta_{j1}x + \beta_{j2}z)}{\sigma_j}\Big) \tag{6.10a}$$

$$f_N(x^*|x) = \frac{1}{\sqrt{2\pi\tau_j^2}}\exp\{\frac{-(x^* - x)^2}{2\tau_j^2}\} \tag{6.10b}$$

$$f_N(x|z) = \frac{1}{\sqrt{2\pi\lambda_j^2}}\exp\{\frac{-(x - \{\kappa_{j0} + \kappa_{j1}z\})^2}{2\lambda_j^2}\} \tag{6.10c}$$

where $S|x, z \sim \mathbf{LogSN}(\mathbf{X}\boldsymbol{\beta}_j, \sigma_j^2, \xi_j)$, $x^*|x \sim \mathbf{N}(x, \tau_j^2)$, $\tau_j^2 : V(\mathbf{x}^*|\mathbf{x})$, $x|z \sim \mathbf{N}(\kappa_{j0} + \kappa_{j1}z,\ \lambda_j^2)$, and $\lambda_j^2 : V(\mathbf{x}|\mathbf{z})$ as shown in Equation (6.7). However, since the true covariate $\mathbf{x}$ is unknown, we face uncertainty regarding the underlying relationship between $\mathbf{x}^*$ and $\mathbf{x}$, leaving us without a clear specification for τ_j^2. Similar to the approach in Chapter 4, we also substitute hypothetical models in Equations (6.11a) and (6.11c) with the following:

$$f_{LSN}(S|x^*, z) = \frac{2}{S\widehat{\sigma}_j}\phi\Big(\frac{\log S - (\widehat{\beta}_{j0} + \widehat{\beta}_{j1}x + \widehat{\beta}_{j2}z)}{\widehat{\sigma}_j}\Big)\Phi\Big(\widehat{\xi}_j\frac{\log S - (\widehat{\beta}_{j0} + \widehat{\beta}_{j1}x + \widehat{\beta}_{j2}z)}{\widehat{\sigma}_j}\Big) \tag{6.11a}$$

$$f_N(x^*|z) = \frac{1}{\sqrt{2\pi\widehat{\lambda}_j^2}}\exp\{\frac{-(x^* - \{\widehat{\kappa}_{j0} + \widehat{\kappa}_{j1}z\})^2}{2\widehat{\lambda}_j^2}\} \tag{6.11b}$$

where $S|x^*, z \sim \mathbf{LN}(\mathbf{X}^*\widehat{\boldsymbol{\beta}}_j, \widehat{\sigma}_j^2, \widehat{\xi}_j)$, $x^*|z \sim \mathbf{N}(\widehat{\kappa}_{j0} + \widehat{\kappa}_{j1}z,\ \widehat{\lambda}_j^2)$, and $\widehat{\lambda}_j^2 : V(\mathbf{x}^*|\mathbf{z})$

(as defined in Equation (4.19), above). By multiplying the outcome model and the covariate model in Equation (6.11), we derive a more practical incomplete joint model, given by $f(S, \mathbf{x}^* \mid \mathbf{z}) = f_{LSN}(S|\mathbf{x}^*, \mathbf{z}) \times f_N(\mathbf{x}^*|\mathbf{z})$. Specifically,

$$f(S, x^* \mid z) = \frac{2}{S\widehat{\sigma}_j\sqrt{2\pi}} \underbrace{\exp\left(-\frac{1}{2}\Big[\frac{\log S - (\widehat{\beta}_{j0} + \widehat{\beta}_{j1}x^* + \widehat{\beta}_{j2}z)}{\widehat{\sigma}_j}\Big]^2\right)}_{\phi(\cdot)}$$
$$\times \underbrace{\int_{-\infty}^{\widehat{\xi}_j \frac{\log S - (\widehat{\beta}_{j0} + \widehat{\beta}_{j1}x^* + \widehat{\beta}_{j2}z)}{\widehat{\sigma}_j}} \frac{1}{\sqrt{2\pi}} e^{-\frac{u^2}{2}}\, du}_{\Phi(\cdot)} \times \frac{1}{\sqrt{2\pi\widehat{\lambda}_j^2}} \exp\left(\frac{-(x^* - \{\widehat{\kappa}_{j0} + \widehat{\kappa}_{j1}z\})^2}{2\widehat{\lambda}_j^2}\right) \tag{6.12}$$

Indeed, the joint model presented in Equation (6.12) is limited by the absence of the true covariate $\mathbf{x}$. However, we connect the complete joint model to the incomplete joint model by marginalizing the complete joint model over the true covariate $\mathbf{x}$ thus: $\int f(S, \mathbf{x}^*, \mathbf{x}|\mathbf{z})d\mathbf{x} = \int f_{LSN}(S|\mathbf{x}, \mathbf{z}) \times f_N(\mathbf{x}^*|\mathbf{x}) \times f_N(\mathbf{x}|\mathbf{z})d\mathbf{x} = f(S, \mathbf{x}^*|\mathbf{z})$; so,

$$\begin{aligned}
&\int f(S, x^*, x \mid z)\, dx \\
&= \int \Bigg[\frac{2}{S\sigma_j\sqrt{2\pi}} \exp\left(-\frac{1}{2}\Big[\frac{\log S - (\beta_{j0} + \beta_{j1}x + \beta_{j2}z)}{\sigma_j}\Big]^2\right) \\
&\qquad \times \int_{-\infty}^{\xi_j \frac{\log S - (\beta_{j0} + \beta_{j1}x + \beta_{j2}z)}{\sigma_j}} \frac{1}{\sqrt{2\pi}} e^{-\frac{u^2}{2}}\, du \\
&\qquad \times \frac{1}{\sqrt{2\pi\tau_j^2}} \exp\{\frac{-(x^* - x)^2}{2\tau_j^2}\} \times \frac{1}{\sqrt{2\pi\lambda_j^2}} \exp\{\frac{-(x - \{\kappa_{j0} + \kappa_{j1}z\})^2}{2\lambda_j^2}\}\Bigg]\, dx
\end{aligned} \tag{6.13}$$

$$\begin{aligned}
&= \frac{2}{S\sigma_j\tau_j\lambda_j 2\pi}\Big[\frac{\beta_{j1}^2}{\sigma_j^2} + \frac{1}{\tau_j^2} + \frac{1}{\lambda_j^2}\Big]^{-\frac{1}{2}} \times \exp\Bigg(\frac{1}{2}\frac{\frac{\xi_j}{\sigma_j^2}\left(\frac{\beta_{j1}^2}{\sigma_j^2} + \frac{1}{\tau_j^2} + \frac{1}{\lambda_j^2}\right)}{\left[\frac{\beta_{j1}^2}{\sigma_j^2} + \frac{\xi_j^2\beta_{j1}^2}{\sigma_j^2} + \frac{1}{\tau_j^2} + \frac{1}{\lambda_j^2}\right]}\mu_t^2 + \frac{1}{2\left[\frac{\beta_{j1}^2}{\sigma_j^2} + \frac{\xi_j^2\beta_{j1}^2}{\sigma_j^2} + \frac{1}{\tau_j^2} + \frac{1}{\lambda_j^2}\right]} \\
&\Big[\frac{\beta_{j1}}{\sigma_j^2}(\log S - \beta_{j0} - \beta_{j2}z)(1 + \xi_j^2) + \frac{x^*}{\tau_j^2} + \frac{(\kappa_{j0} + \kappa_{j1}z)}{\lambda_j^2}\Big]^2 - \frac{1}{2}\Big[\frac{(1 + \xi_j^2)(\log S - \beta_{j0} - \beta_{j2}z)^2}{\sigma_j^2} + \frac{(x^*)^2}{\tau_j^2} \\
&+ \frac{(\kappa_{j0} + \kappa_{j1}z)^2}{\lambda_j^2}\Big]\Bigg) \times \Phi(f) = f(S_h, x^* \mid z)
\end{aligned} \tag{6.14}$$

where

$$\mu_t = \left[\frac{(\frac{1}{\tau_j^2} + \frac{1}{\lambda_j^2})(\log S - \beta_{j0} - \beta_{j2} z) - \frac{\beta_{j1} x^*}{\tau_j^2} - \frac{\beta_{j1}(\kappa_{j0} + \kappa_{j1} z)}{\lambda_j^2}}{\frac{\beta_{j1}^2}{\sigma_j^2} + \frac{1}{\tau_j^2} + \frac{1}{\lambda_j^2}} \right]$$

and $\Phi(f) = \Phi(\widehat{\xi}_j \frac{\log S - (\widehat{\beta}_{j0} + \widehat{\beta}_{j1} x^* + \widehat{\beta}_{j2} z)}{\widehat{\sigma}_j})$

The detailed steps for deriving this solution in Equation (6.14) are elaborated in F.5 of Appendix F. In the solution presented in Equation (6.14), the unobserved true covariate $\mathbf{x}$ is excluded, resulting in consistency with the incomplete joint model from Equation (6.12). The crucial point is that, despite stemming from distinct equations, both solutions in Equations (6.12) and (6.14) represent the same joint model $f(S_h, x^* \mid z)$ within a practical framework $\mathbf{x}$. This allows for a straightforward correspondence between the parameters of the complete and incomplete joint models by aligning their respective parameterizations. This connection is represented by a system of equations, where the parameters of the complete joint model are expressed in terms of those from the incomplete joint model. All derivations of the system of equations and detailed explanations can be found in F.5 and the resulting system of equations is presented below:

$$\lambda_j^2 = \widehat{\lambda}_j^2 - \tau_j^2 \tag{6.15a}$$

$$\kappa_{j0} = \widehat{\kappa}_{j0} \tag{6.15b}$$

$$\kappa_{j1} = \widehat{\kappa}_{j1} \tag{6.15c}$$

$$\beta_{j1} = \frac{\widehat{\beta}_{j1} \widehat{\lambda}_j^2}{\widehat{\lambda}_j^2 - \tau_j^2} \tag{6.15d}$$

$$\beta_{j0} = \widehat{\beta}_{j0} - \frac{\beta_{j1} \widehat{\kappa}_{j0} \tau_j^2}{\widehat{\lambda}_j^2} \tag{6.15e}$$

$$\beta_{j2} = \widehat{\beta}_{j2} - \frac{\beta_{j1} \widehat{\kappa}_{j1} \tau_j^2}{\widehat{\lambda}_j^2} \tag{6.15f}$$

$$\sigma_j^2 = \widehat{\sigma}_j^2 - \frac{\beta_{j1}^2}{\frac{1}{\tau_j^2} + \frac{1}{\lambda_j^2}} \tag{6.15g}$$

$$\xi_j^2 = \frac{\widehat{\xi}_j^2 \left(\frac{\beta_{j1}^2}{\sigma_j^2} + \frac{1}{\tau_j^2} + \frac{1}{\lambda_j^2} \right)}{\frac{1}{\tau_j^2} + \frac{1}{\lambda_j^2} - \frac{\widehat{\xi}_j^2 \beta_{j1}^2}{\sigma_j^2}} \tag{6.15h}$$

Consequently, biased parameter estimates due to the NDB covariate can be corrected using this system in Equation (6.15), offering insights into how to mitigate the model risk associated with the NDB covariate. Note that, in Equation (6.15), the parameter computation for the complete joint model hinges on τ_j^2, which lacks a clear specification due to the unknown relationship between $\mathbf{x}^*$ and $\mathbf{x}$.

The prior for τ_j^2 and scaling factor ζ: Similarly in Section 4.3.2, we project the connection between the NDB covariate $\mathbf{x}^*$ and the true covariate $\mathbf{x}$, using the known covariate $\mathbf{z}$. To begin, we assume that the variance of the unobservable true covariate $\lambda^2 : V(\mathbf{x}|\mathbf{z})$ is a scaled version of the variance of the observable NDB covariate $\widehat{\lambda}^2 : V(\mathbf{x}^*|\mathbf{z})$, with the scaling factor $0 < \zeta < 1$ (i.e., $\lambda^2 = \zeta \times \widehat{\lambda}^2$). This is because, as shown in Equations (6.15a), (6.15b), and (6.15c), $\widehat{\lambda}^2$ is always larger than λ^2, and both $\mathbf{x}^*|\mathbf{z}$ and $\mathbf{x}|\mathbf{z}$ follow normal distributions with equal means (given the definition in Equation (6.7)). If this holds true, as seen in Equation (6.15a), the variance of the linking component τ^2 can be expressed as $\hat{\lambda}^2 - \lambda^2 = (1-\zeta)\hat{\lambda}^2$, i.e., τ^2 can ultimately be approximated using the observable proxy variance: $\widehat{\lambda}^2 : V(\mathbf{x}^*|\mathbf{z})$.

Building on this, Figure 6.2 summarizes our concept of approximating the linking component $\tau^2 : V(\mathbf{x}^*|\mathbf{x})$ by leveraging $\widehat{\lambda}^2 : V(\mathbf{x}^*|\mathbf{z})$ as a proxy. The implication is that the scaling factor ζ quantifies how reliable the observed variance $\widehat{\lambda}^2 : V(\mathbf{x}^*|\mathbf{z})$ is as a stand-in for the variance of the linking component τ^2. A smaller value of ζ suggests a higher level of alignment, indicating that a significant portion of τ^2 can be effectively captured by $\widehat{\lambda}^2$. Conversely, a larger value of ζ suggests that the known covariate $\mathbf{z}$ is hardly an adequate substitute for the unobserved true covariate $\mathbf{x}$. ζ measures the alignment between τ^2 and $\widehat{\lambda}^2$. Given that the true covariate $\mathbf{x}$ is unobservable, a guideline for determining the optimal ζ should be developed in the absence of gold standard data. Note that the dotted curves in Figure 6.2 simply represent the true distribution of $\mathbf{x}|\mathbf{z}$ in Equation (6.7d), in contrast to the biased distribution of $\mathbf{x}^*|\mathbf{z}$ in Equation (6.7e), highlighting that a greater difference indicates greater usefulness of $\widehat{\lambda}^2 : V(\mathbf{x}^*|\mathbf{z})$. It is as if we are leveraging the idea that the bias present in the unobservable $\mathbf{x}^*|\mathbf{x}$ is transferred to the observable $\mathbf{x}^*|\mathbf{z}$.

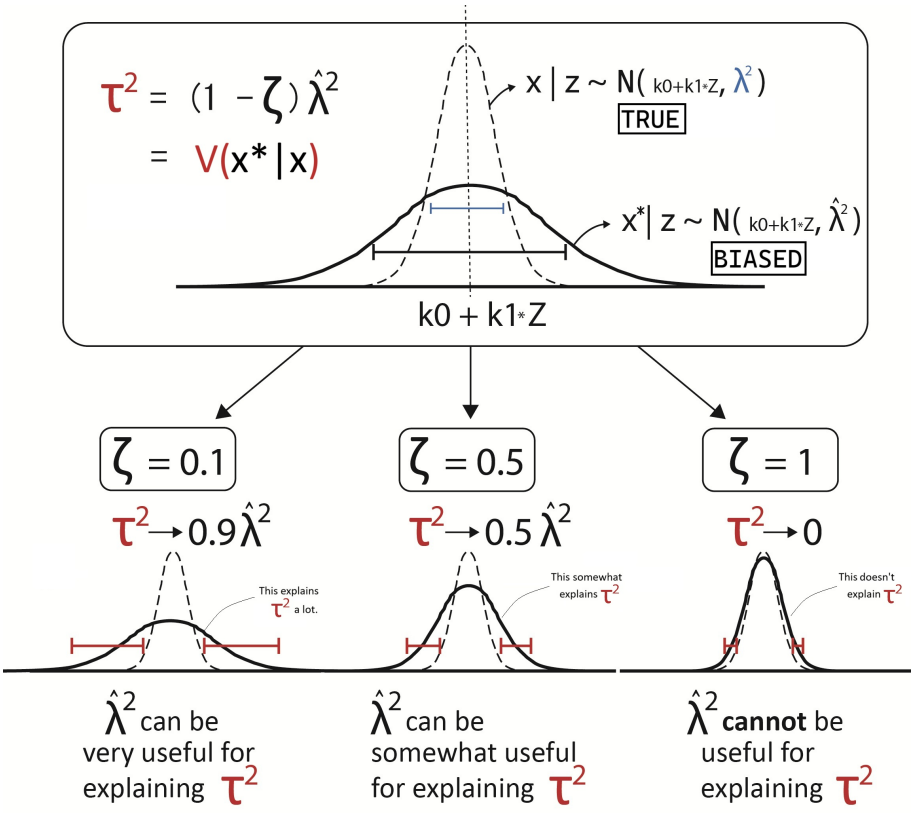

Figure 6.2: A diagram of the development of unknown τ^2 using the scaling factor ζ and $\widehat{\lambda}^2 : V(\mathbf{x}^*|\mathbf{z})$. To what extend the observable $\widehat{\lambda}^2 : V(\mathbf{x}^*|\mathbf{z})$ can be useful for accounting for the unobservable $\tau^2 : V(\mathbf{x}^*|\mathbf{x})$? The optimal ζ answers this question.

Gibbs sampler modification with the Gustafson correction: Drawing from this prior knowledge on τ^2, we now incorporate Gustafson's equations in Equation (6.15) into the DPM Gibbs sampler. We propose the hybrid DPM Gibbs sampler as outlined in Algorithm (F.3) in Appendix F. This involves **[Stage.1] Re-assigning cluster memberships** and **[Stage.2] Updating cluster parameters**.

(a) In line 4 of **[Stage.1]**, we assess whether any clusters are composed of only a single observation, which is an important step in introducing a new continuous cluster. When such a single-observation cluster is identified, it must be re-initialized. This process involves removing the existing cluster membership and its corresponding parameters. The reason for this re-initialization is that the discrete cluster needs to be removed to make way for the formation of a new continuous cluster or absorption into the existing discrete clusters. This

ensures that the clustering can adapt by transitioning from a discrete structure to one that accommodates continuous clusters. Between lines 16 and 25, we calculate the probabilities for clusters, comparing both the existing J discrete clusters and an additional continuous cluster ($j = J+1$). To introduce this new cluster and its parameters into the existing clustering scenario, we employ the Polya Urn scheme, which facilitates the integration of new continuous clusters into the group of discrete clusters. However, the clustering scenario developed at this stage is based on the NDB covariate $\mathbf{x}^*$ and is therefore subject to the covariate-based model risk.

(b) [**Stage.2**] is primarily focused on mitigating covariate-based model risk in the update of the cluster parameter estimates. This stage is designed to reduce the inference bias that arises from the reliance on the NDB covariate $\mathbf{x}^*$. To accomplish this, [**Stage.2**] utilizes the Gustafson correction technique to adjust the cluster parameter estimates. This calibration is based on the relationship between the unobservable complete joint model $f(S, \mathbf{x}^*, \mathbf{x} \mid \mathbf{z})$ and the observable incomplete joint model $f(S, \mathbf{x}^* \mid \mathbf{z})$ discovered from the analytical derivation in F.5 in Appendix F. Between lines 35 and 40, we initialize empty vectors and matrices to store the final outcomes and covariate parameter estimates, denoted as $\boldsymbol{\theta}^{new}$ and $\boldsymbol{w}^{new}$. These storage structures are prepared in advance to ensure that the results of subsequent calculations can be systematically recorded. Before proceeding with the cluster-wise parameter update, the value of the scaling factor ζ is configured, establishing the basis for how the measurement parameter τ^2 will be adjusted. Additionally, a posterior sample of the DPM precision parameter α^{new} is drawn from the distribution derived in F.2.3, which will play a crucial role in limiting the cluster development during the update process.

(c) From line 44, for each cluster j, the posterior parameters τ_j^2 for the linking (measurement) component and $\pi_j, \widehat{\boldsymbol{\kappa}}_j, \widehat{\lambda}_j^2$ for the covariate models are drawn

from below:

$$\boldsymbol{w}_j : \begin{cases} \pi_j \sim \textbf{Beta}(g_0 + \Sigma \mathbf{z}_j,\ h_0 + n_j - \Sigma \mathbf{z}_j) \\ \widehat{\boldsymbol{\kappa}}_j \sim \textbf{MVN}\Big(\Big[(\tilde{\Sigma}_k^{-1} + \mathbb{K}_1^T \mathbb{K}_1)^{-1}(\tilde{\Sigma}_k^{-1}\tilde{\boldsymbol{\kappa}} + \mathbb{K}_2)\Big],\ \widehat{\lambda}_j^2\Big[\tilde{\Sigma}_k^{-1} + \mathbb{K}_1^T \mathbb{K}_1\Big]^{-1}\Big) \\ \widehat{\lambda}_j^2 \sim \textbf{InvGa}\big(\frac{c_0+n_j}{2},\ \frac{1}{2}(d_0 + \Sigma(\mathbf{x}_j - \widehat{\kappa}_{0j} + \widehat{\kappa}_{1j}\mathbf{z}_j)^2)\big) \\ \tau_j^2 = (1-\zeta)\widehat{\lambda}_j^2 \end{cases} \tag{6.16}$$

As mentioned in Chapter 4, the derivation of the posterior parameterization in Equation (6.16) is provided in F.2.3 in Appendix F. Note that we utilize the system of equations derived in Equation (6.15) to refine the estimated outcome parameter values - $\boldsymbol{\theta}_j^{new} : \{\boldsymbol{\beta}_j^{new}, \sigma_j^{2new}, \xi_j^{new}\}$. This adjustment is made by incorporating parameter samples from the incomplete joint model — $\widehat{\beta}_{j0}, \widehat{\beta}_{j1}, \widehat{\beta}_{j2}, \widehat{\sigma}_j, \widehat{\xi}_j, \widehat{\lambda}_j, \widehat{\kappa}_{j0}, \widehat{\kappa}_{j1}$ — along with the variance $\tau_j^2 = (1-\zeta)\widehat{\lambda}_j^2$ (where $0 < \zeta < 1$) of the linking component in Equation (6.10b). It is important to consider that the parameter samples obtained from the incomplete joint model within the Gibbs sampler must satisfy specific conditions as outlined by Equation (6.17). For instance, in line 47, given that Gustafson's equations in Equations (6.15a), (6.15g), and (6.15h) involve squared values, it is crucial to ensure that the following expressions are positive:

$$d1 = \widehat{\lambda}_j^2 - \tau_j^2 > 0; \quad d2 = \widehat{\sigma}_j^2 - \frac{\beta_{j1}^2}{\frac{1}{\tau_j^2} + \frac{1}{\lambda_j^2}} > 0; \quad d3 = \frac{1}{\tau_j^2} + \frac{1}{\lambda_j^2} - \frac{\widehat{\xi}_j^2 \beta_{j1}^2}{\sigma_j^2} > 0 \tag{6.17}$$

These requirements in Equation (6.17) make sure that the drawn parameters preserve their proper associations with the true parameters such as $\beta_{j0}, \beta_{j1}, \beta_{j2}, \sigma_j, \xi_j, \lambda_j, \kappa_{j0}, \kappa_{j1}$, while the Gibbs sampler excludes any deviations from them. This is followed by the posterior parameter updates with Gustafson correction between lines 63 to 66 in Algorithm (F.3).

6.4 Numerical Experiments with NDB Covariate

6.4.1 Data: Swautoins + Brvehins2

We use two distinct insurance datasets to assess the performance of our hybrid DPM approach. These datasets pose the covariate-based model risk challenges that include the presence of unobserved heterogeneity (RQ1.1), convolution of the log-skewnormal outcomes (RQ1.2), growing number of sample size (RQ1.3), and the unknown noise introduced by NDB covariates (RQ2.2). To maintain clarity, each dataset is limited to just two covariates: a binary variable $\mathbf{z}$ and a continuous variable $\mathbf{x}$, both of which are used to explain the aggregate claim data S_h. The continuous covariates $\mathbf{x}$ are assumed to be mismeasured due to NDB errors. Additionally, as in Chapter 5, we consistently adhere to an identical data structure for both datasets, following the specified data format:

$$
\begin{array}{lcccc}
 & Area/Zone_1 & Area/Zone_2 & \cdots & , Area/Zone_? \\
\text{Policy } (h=1): & \{(S_1,\ \mathbf{X}_1), & (S_1,\ \mathbf{X}_1), & \cdots & ,(S_1,\ \mathbf{X}_1)\} \\
\text{Policy } (h=2): & \{(S_2,\ \mathbf{X}_2), & (S_2,\ \mathbf{X}_2), & \cdots & ,(S_2,\ \mathbf{X}_2)\} \\
 & & \vdots & & \\
\text{Policy } (h=H): & \{(S_H, \mathbf{X}_H), & (S_H, \mathbf{X}_H), & \cdots & ,(S_H, \mathbf{X}_H)\}
\end{array}
$$

For each policy, represented as an individual observation h, the outcome S_h and covariates $\mathbf{X}_h : \{z_h, x_h^*\}$ are categorized based on geographic regions, such as specific areas or zones where the claim data are concerned. Ideally, the aggregate claim amounts S_h should be assessed separately for each cluster, indexed by $j = 1, \cdots, J$. However, in contrast to the approach taken in Chapter 4, where clusters were treated as deterministic and fixed (using geographic region, etc.), this chapter, as in Chapter 5, assumes that clusters are stochastic. This means that the determination of clusters is governed by probabilistic mechanisms of the DPM framework rather than fixed assignments (in this chapter, we apply a degree of skepticism to the validity of

fixed clusters). As a result, the risk profile identification associated with each cluster cannot be immediate when modeling the aggregate claim S_h, but this probabilistic nature of clusters in the DPM framework can reflect the diverse risk characteristics in the prediction of S_h.

Our two datasets — **Swautoins** (Swedish motor insurance data), **Brvehins2** (Brazilian motor insurance data) — are used for comparison. The **Swautoins** and **Brvehins2** datasets are sourced from the publicly available R package, **'CAS-datasets'**. They were selected due to their contrasting sample sizes, which serve a crucial purpose in the investigation of Bayesian scalability (RQ1.3) in this chapter. The **Swautoins** datasets contain fewer than 2,000 observations, making it suitable for smaller-scale computations. On the other hand, the **Brvehins2** dataset is significantly larger, comprising over 60,000 observations. This larger dataset provides a rigorous test of our DPM's capacity to maintain performance and reliability when scaling up to more complex, real-world situations where vast amounts of claim data are involved.

Swautoins pertains to motor insurance data collected in 1977 for seven geographical zones in Sweden. This dataset contains a moderate size of policies $H = 1,799$ with non-zero claim amounts. *TotalLoss*, the outcome variable, describes the aggregate claim amount per policy. The binary covariate *Experience* indicates whether the policyholders have had any claims in the past (1 is 'yes', 0 is 'no'). The continuous covariate *Ln-InsuredAVG* represents the log-transformed average value of the assets covered under the policy (to account for skewness in the asset value distribution). For more details on this dataset, refer to Frees 2009.

Brvehins2 contains auto insurance policy information collated from the database of the Brazilian Insurance Oversight Agency (SUSEP) in 2011. As noted earlier, this dataset features a substantial sample size of $H = 62,512$ observations, allowing us to compare our DPM model's performance in moderate-scale studies with its effectiveness in large-scale implementations. The aggregate claim amount (sum of payments) for each policy is described by the outcome variable *AggClaim*, and its

variations across different policies are explained by one binary and continuous covariate. The binary covariate *Affiliation* indicates whether the reported claims are associated with a corporation or an individual policyholder, providing insight into the type of entity involved in the claim (1 is 'individual', 0 is 'corporation'). The continuous covariate *Ln-TotalExp* reflects the level of risk exposure on a log scale to the assets being insured. For more comprehensive details, refer to Charpentier 2014. Figure 6.3 provides a comprehensive view of the relationships between the outcome

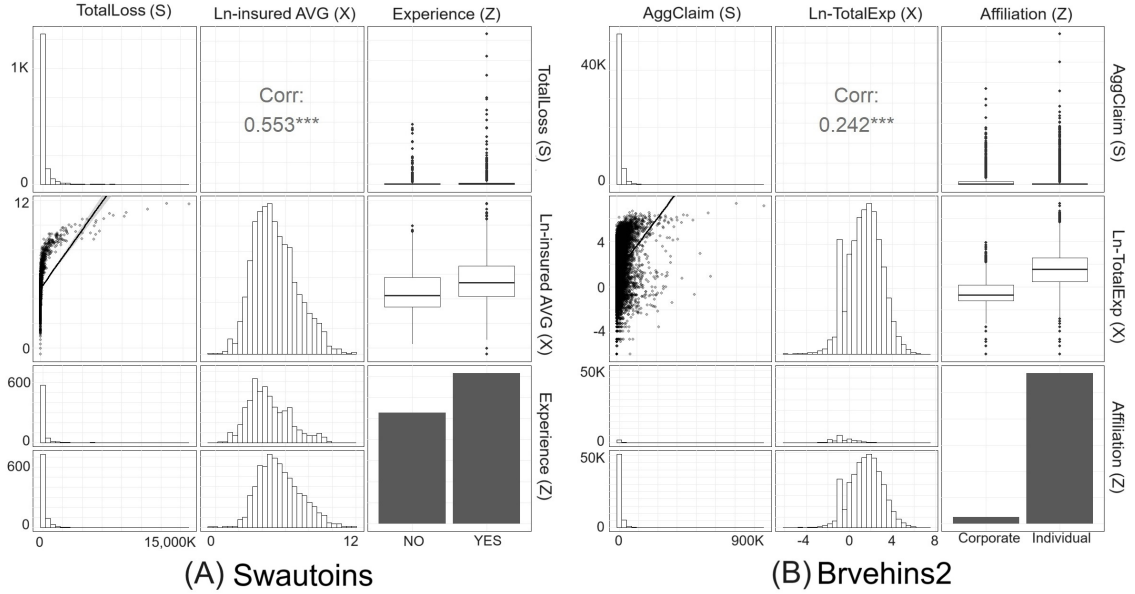

Figure 6.3: The pairwise comparison of variables in the two datasets: **(A) Swautoins** (Swedish Motor Insurance) and **(B) Brvehins2** (Brazilian Motor Insurance).

S and covariates $\mathbf{X} : \{\mathbf{z}, \mathbf{x}\}$ in the two datasets we used. The diagonal displays the shape of the distributions of each variable. Both outcome variables exhibit significant right skewness, while the continuous covariates show Gaussian properties. Note that the binary covariate in the **(A) Swautoins** dataset indicates that groups with prior claims (*Experience*: YES) tend to file more claims. Similarly, in the **(B) Brvehins2** dataset, the majority of claims are made by individual drivers (*Affiliation*: Individual) rather than those driving vehicles registered by corporations. The off-diagonal plots — scatter for cont.+cont. and box for binary+cont. — show that larger binary groups have more outliers and greater variability in both datasets.

6.4.2 Implementation

This experiment's main objective is to predict the aggregate claim amount $E[S_h|\mathbf{X}^*]$ for risk premium development based on the **Swautoins** and **Brvehins2** datasets, while addressing the model risks outlined in research questions: RQ1.1 heterogeneity, RQ1.2 convolution, RQ1.3 scalability, and RQ2.2 NDB covariates. Additionally, continuing from the work presented in Chapter 4, this experiment examines practical guidelines for selecting the optimal scaling factor $0 < \zeta < 1$ when gold standard data is unavailable. We also focus on comparing our DPM-based Gustafson correction with hierarchical GLM-based Gustafson correction in Chapter 4.

Design of simulation data: The experiment hinges on our derived prior knowledge: $\tau_j^2 = (1-\zeta)\widehat{\lambda}_j^2$ (where $0 < \zeta < 1$), with the scaling factor ζ representing the reliability of using $V(\mathbf{x}^*|\mathbf{z})$ as a proxy for $V(\mathbf{x}^*|\mathbf{x})$ to estimate τ_j^2. Hence, one of our main tasks is to develop a rule of thumb to identify the optimal value of ζ across different error rates R_{ϵ_x}, which reflects varying degrees of NDB error.

To this end, simulation data can be generated using predefined error rates $R_{\epsilon_x} = 0.01/0.10/0.25$ based on the **Swautoins** and **Brvehins2** datasets, allowing us to test which value of the scaling factor ζ best improves the accuracy of τ_j^2 estimation across different error rates. As in Chapter 4, we deliberately introduced artificial errors $\boldsymbol{\epsilon}_h$, characterized as NDB errors, into the continuous covariate $\mathbf{x}$ of these three datasets — **Swautoins** and **Brvehins2** datasets. We adopt the noise generation technique developed by Klau et al. 2021 with the process outlined in Figure 4.2 and the formulas provided in Equations (4.26) and (4.27).

Candidate models: Considering the specified error rates $R_{\epsilon_x} = 0.01/0.10/0.25$, we evaluate the effectiveness of the Gustafson correction based on our DPM framework. By comparing DPM model based on gold standard data (without NDB errors) to other DPM models and hierarchical GLM based on the simulation data generated with varying error rates (with NDB errors), we study the effectiveness of these approaches in addressing RQ1.1 on heterogeneity, RQ1.2 on convolution, and RQ2.2 regarding the NDB covariate, with a focus on prediction accuracy.

A comparison plan, detailed in Figure 6.4, outlines the evaluation of the Gustafson correction in tandem with the DPM and hierarchical GLM framework. This plan encompasses four modeling setups, labeled as (A) to (D), designed to assess the impact of the NDB covariate with varying error rates $R_{\epsilon_x} = 0.01/0.10/0.25$ on the development of risk premium. As in Chapter 4, Model(A) serves as the gold standard

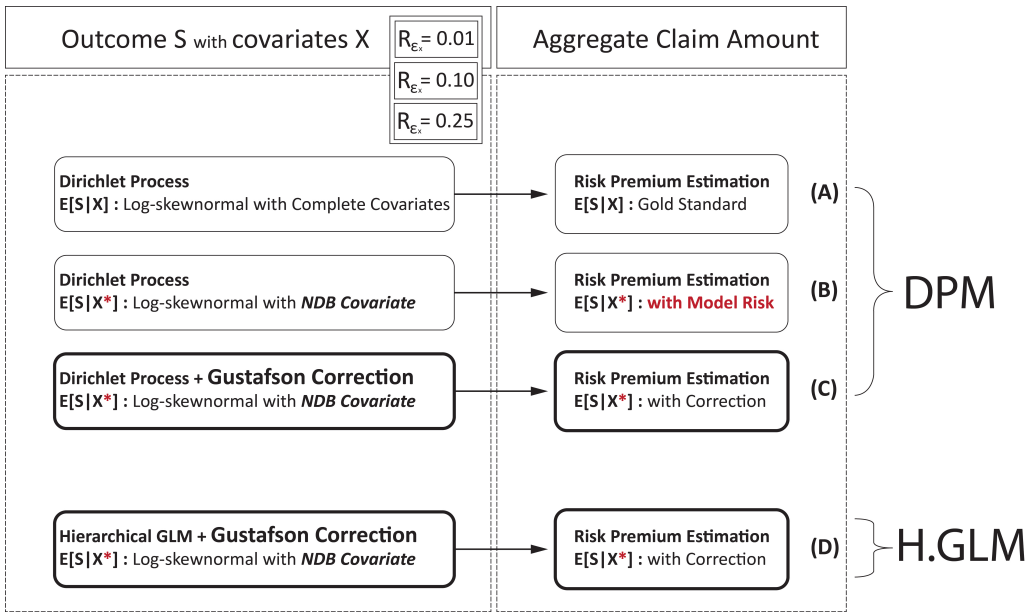

Figure 6.4: Four candidate models — (A) to (D) — for risk premium development. Specifically, Model(B),(C),(D) need to be thoroughly compared across various error rates R_{ϵ_x} — $1\%, 10\%, 25\%$ — in the NDB covariate $\mathbf{x}^*$.

in this chapter, constructed using the true covariates $\mathbf{x}$ within the DPM framework to provide a benchmark for performance comparison. In contrast, Model(B), developed with the NDB covariate $\mathbf{x}^*$ under the same DPM framework, highlights the emergence of model risk due to the NDB error in the covariate. This model clearly illustrates the negative effects that these errors have on risk premium predictions. The analysis is further extended to Model(C), which applies our Gustafson correction within the DPM structure, and Model(D), the main competing model, which integrates the Gustafson correction within the hierarchical GLM framework.

This experiment also addresses the scalability issue outlined in RQ1.3. Specifically, using **Brvehins2** dataset, which contains over 60,000 data points, we implement and evaluate four distinct candidate models — referred to as Model(A)

through (D). Figure 6.5 provides a detailed depiction of how High Performance Computing (HPC) cluster is utilized to handle this large-scale data through parallel simulation techniques, as described earlier in Figure 6.1 in Section 6.3.1. A total of 22 HPC computations, with varying error rates $R_{\epsilon_x} = 0.01/0.10/0.25$ and scaling factors $\zeta = 0.71 \sim 0.99$, were performed. We break down the training set into 24 subsets ($60,063 \approx 2,503 \times 24$), and each HPC cluster computation used 23 shards and 1 anchor block, corresponding to 23 CPUs, to model the risk premium based on the **Brvehins2** dataset. The parameters are estimated for each DPM model —

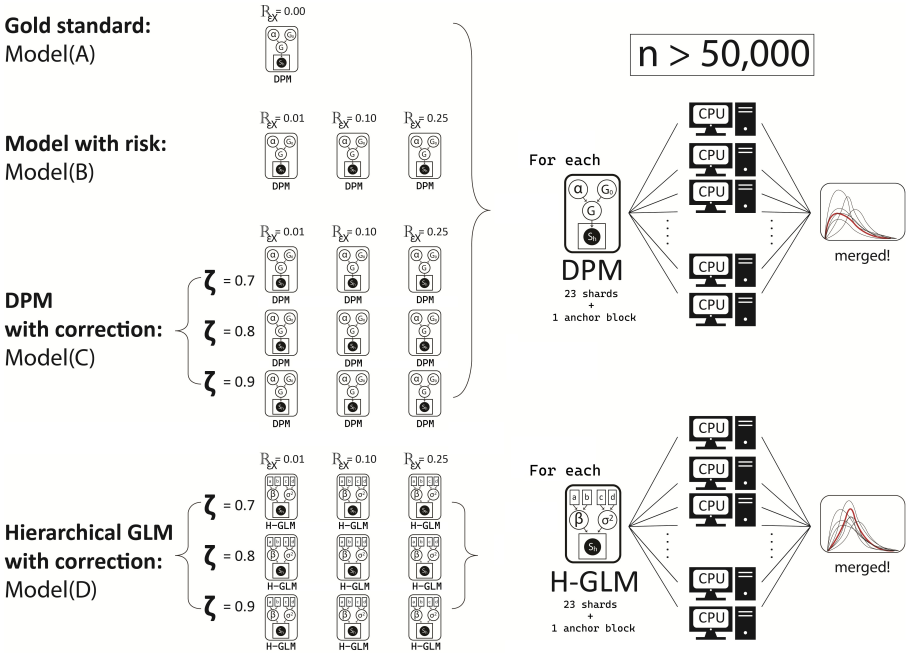

Figure 6.5: Large scale implementation of the four candidate models(A) to (D), using data **Brvehins2** ($n > 50,000$) on HPC. The results were seamlessly integrated, leveraging the parallel simulation techniques introduced in Section 6.3.1. Each parallel computation utilized 23 CPUs.

Model(A), (B), and (C) — as well the Hierarchical GLM model — Model(D) —, using two independent Markov chains, each running $M = 10,000$ Gibbs sampling iterations based on Algorithm (F.3). The first 9,900 iterations were discarded as part of a burn-in phase to ensure the chains reached equilibrium. To verify convergence, the Brooks-Gelman statistic (Brooks and Gelman 1998) was applied, confirming adequate mixing of the chains. Note that, a Metropolis-Hastings (MH) algorithm, outlined in Algorithm (F.2.2), was embedded within each Gibbs sampling iteration

to update the outcome parameters —$\boldsymbol{\beta}_j, \sigma_j^2, \xi_j$ —, since conjugate priors were not available for the log-skenormal outcome data models in Models(A), (B), (C), and (D). Additionally, for technical details on the parallel MCMC simulation technique used for large-scale implementation on the **Brvehins2** dataset, the reader can refer to Algorithm (F.4) in Appendix F.

Choice of hyperparameters: To implement the DPM models – Model(A), (B), (C) – along with the hierarchical GLM — Model(D) — specified in Figure 6.4, the initial parameter sampling for their Gibbs samplers begins with the following hyperparameter values:

$$\text{DPMs}\begin{cases}\text{For } \boldsymbol{\theta}: \begin{cases}\{\boldsymbol{\beta}_0 = \boldsymbol{\beta}_{GLM},\ \Sigma_{\beta_0} = \Sigma_{\beta_{GLM}}\} & \text{to sample } \boldsymbol{\beta}\\ \{u_0 = 1.76,\ v_0 = 7.11\} & \text{to sample } \sigma^2\\ \{\nu_0 = N-1\} & \text{to sample } \xi\end{cases}\\ \text{For } \boldsymbol{w}: \begin{cases}\{g_0 = 0.5,\ h_0 = 0.5\} & \text{to sample } \pi\\ \{\tilde{\boldsymbol{\kappa}} = \boldsymbol{\kappa}_{GLM},\ \tilde{\Sigma}_\kappa = \Sigma_{\kappa_{GLM}}\} & \text{to sample } \boldsymbol{\kappa}\\ \{c_0 = 0.5,\ d_0 = 0.5\} & \text{to sample } \lambda^2\end{cases}\\ \text{For precision}: \{\gamma_0 = 1,\ \psi_0 = 1\} \quad \text{to sample } \alpha\end{cases} \tag{6.18}$$

$$\text{Hierarchical GLM}\begin{cases}\text{For } \boldsymbol{\theta}: \begin{cases}\{m_0 = \boldsymbol{\beta}_{GLM},\ \delta = 0.01\} & \text{to sample } \boldsymbol{\beta}_0 \text{ for } \boldsymbol{\beta}\\ \{q_0 = p+2,\ \Lambda = \Sigma_{\beta_{GLM}}\} & \text{to sample } \Sigma_{\beta_0} \text{ for } \boldsymbol{\beta}\\ \{\rho_{u1} = 0.125,\ \rho_{u2} = 1.5\} & \text{to sample } u_0 \text{ for } \sigma^2\\ \{\rho_{v1} = 8,\ \rho_{v2} = 1\} & \text{to sample } v_0 \text{ for } \sigma^2\\ \{\nu_0 = N-1\} & \text{to sample } \xi\end{cases}\\ \text{For } \boldsymbol{w}: \begin{cases}\{g_0 = 0.5,\ h_0 = 0.5\} & \text{to sample } \pi\\ \{\tilde{\boldsymbol{\kappa}} = \boldsymbol{\kappa}_{GLM},\ \tilde{\Sigma}_\kappa = \Sigma_{\kappa_{GLM}}\} & \text{to sample } \boldsymbol{\kappa}\\ \{c_0 = 0.5,\ d_0 = 0.5\} & \text{to sample } \lambda^2\end{cases}\end{cases} \tag{6.19}$$

Note that hyperparameters in Equation (6.19) for the hierarchical GLM, used as our competing model, introduce an additional layer of parameter estimates that consist of $m_0, \delta, q_0, \Lambda, \rho_{u1}, \rho_{u2}, \rho_{v1}, \rho_{v2}$ for the outcome variable. In contrast, our DPM model

does not include this layer, as its inherent flexibility in capturing the complexity of the data makes it unnecessary (Teh and Jordan 2010). As explained in Chapter 4, a preliminary run of the Gibbs sampler, starting with random values, provided the initial idea of the selection of the specified values in Equations (6.18) and (6.19).

Lastly, this chapter emphasizes several key criteria for model validation, including: 1)LPPD, 2)SSPE, 3)SAPE, 4)D_{KL}, and 5)CTE to compare models' predictive capabilities and reliability. For a more detailed discussion of these criteria, refer to Section 3.5.1 for their full descriptions.

6.4.3 Results with Swautoins ($H = 1,799$)

For this dataset, we construct a training set containing 1,553 instances of aggregate claim outcomes and corresponding covariates $(S, \mathbf{X})$ to develop the model, alongside a test set with 246 records $(S', \mathbf{X}')$ for assessing prediction accuracy. As specified in the previous section and Figure 6.4, Model(A) serves as the gold standard, providing a reference point for comparing the performance of Model(B), Model(C), and Model(D) across different error rate scenarios. These models were fitted to three versions of the **Swautoins** dataset (in a moderate sample size setting) with varying error rates $R_{\epsilon_x} = 0.01, R_{\epsilon_x} = 0.1$, and $R_{\epsilon_x} = 0.25$ in the NDB covariate $\mathbf{x}^*$. While Model(B) demonstrates the impact of model risk due to fitting with the NDB covariates $\mathbf{x}^*$, Models(C) and (D) incorporate the Gustafson correction to evaluate the effectiveness of these methods in mitigating the model risk which is introduced by the NDB covariate across varying error rates. We also assess the reliability of using $\mathbf{z}$ as a substitute for $\mathbf{x}$ by testing a range of scaling factor values $\zeta = \{0.10, \cdots, 0.99\}$, which reflect varying confidence levels in using $\mathbf{x}^*|\mathbf{z}$ as a proxy for unknown $\mathbf{x}^*|\mathbf{x}$.

Gold standard [Model(A)]: The gold standard — Model(A) in Figure 6.4 — developed with the DPM framework is examined. As the error-free benchmark model, its predictive distribution $f(lnS_h|\mathbf{X})$ is obtained using the true covariates: $\mathbf{z}$ and $\mathbf{x}$. In Figure 6.6, we present overlays of the last 100 predictive density scenarios for the aggregate claim amount on a log scale $lnS_h|\mathbf{X} \sim \mathbf{SN}(\mathbf{X}^T\boldsymbol{\beta},\ \sigma^2,\ \xi)$, along

with the MCMC trace plot based on 100 LPPD draws from the 10,000 iterations of the Gibbs sampler. With these scenarios, the ultimate predictive distribution can

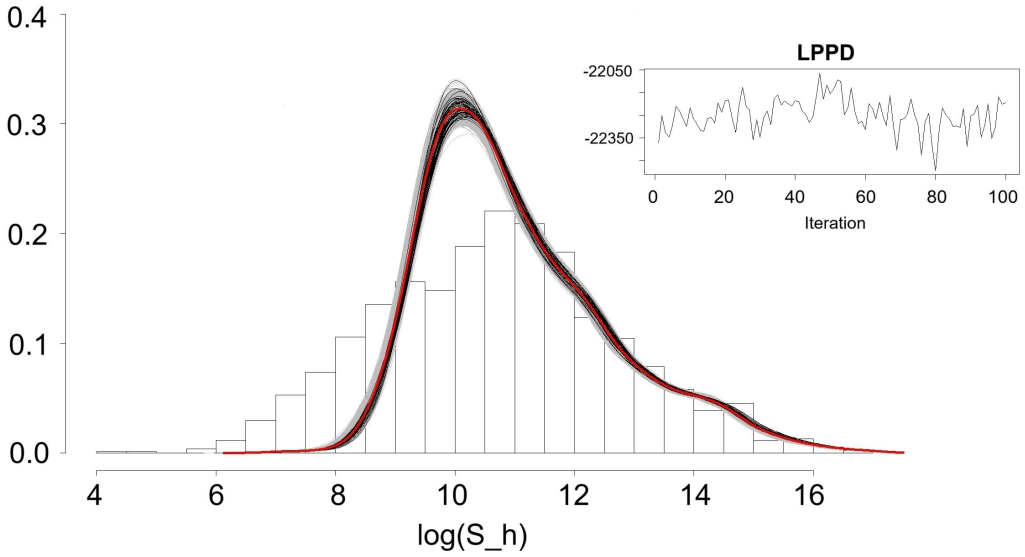

Figure 6.6: Model(A) Results with the **Swautoins** dataset: a histogram of the observed aggregate claim amount on a log scale and the last 100 out-of-sample predictive density scenarios $f(lnS_h|\mathbf{X})$ (black curves) overlaid. The average predictive density, represented by the red curve, results from converged estimates.

be computed using the formula in Equation (6.3). Figure 6.6 shows that the resulting predictive density of aggregate claim amounts, displayed on a log scale, aligns well with the observed values in the upper tail, while it may not fully capture the distortion in the lower range (lower tail). This indicates that the model effectively captures the probabilities associated with more extreme, higher values of lnS_h, but the same precision is not evident in the lower tail. This could suggest that with the inclusion of the covariate $\mathbf{X}$, this predictive modeling decreases the variance of the lnS_h, making the risk premium prediction more conservative by prioritizing the chance of larger losses.

Gustafson correction [Model(B) vs Model(C) vs Model(D)]: As illustrated in Figure 6.4, the goal is now to determine which of the two models — Model(C) and Model(D) — produces results that most closely align with Model(A), the gold standard. In particular, we focus on comparing the performance of Model(C), which incorporates the Gustafson correction within our DPM framework, with that

of Model(D), in which the Gustafson correction is applied within a hierarchical GLM framework.

Table 6.1 presents the results of the sensitivity analysis conducted to explore the optimal scaling factor ζ across three distinct implementations of the DPM models.

Model	$\boldsymbol{\theta}, \tau^2$	Parameter estimates $\boldsymbol{\theta}$: $\{\beta_0, \beta_1, \beta_2, \sigma^2, \xi\}$ and τ^2								
Model(A) Gold standard	β_0	7.08 with 95% Credible Interval: $\{7.02 \leq \beta_0 \leq 7.14\}$								
	β_1	0.84 with 95% Credible Interval: $\{0.82 \leq \beta_1 \leq 0.85\}$								
	β_2	-0.46 with 95% Credible Interval: $\{-0.50 \leq \beta_2 \leq -0.41\}$								
	σ^2	4.05 with 95% Credible Interval: $\{0.89 \leq \sigma^2 \leq 12.34\}$								
	ξ	-0.07 with 95% Credible Interval: $\{-0.69 \leq \xi \leq 0.63\}$								
		Error rate in $\mathbf{x}^*$: 0.01			Error rate in $\mathbf{x}^*$: 0.10			Error rate in $\mathbf{x}^*$: 0.25		
Model(B) Before correction	β_0	7.09			8.07			10.38		
	β_1	0.84			0.73			0.28		
	β_2	-0.50			-0.81			0.16		
	σ^2	4.43			2.99			3.66		
	ξ	-0.06			-0.51			-0.78		
		ζ : 0.9	ζ : 0.95	ζ : 0.99	ζ : 0.85	ζ : 0.9	ζ : 0.95	ζ : 0.85	ζ : 0.9	ζ : 0.95
Model(C) After correction	β_0	7.12	7.04	7.03	7.66	7.77	7.89	9.18	8.85	7.11
	β_1	0.83	0.85	0.86	0.82	0.76	0.74	0.91	0.88	0.81
	β_2	-0.48	-0.51	-0.52	-0.83	-0.78	-0.93	-0.23	-0.34	-0.49
	σ^2	1.65	1.58	1.45	2.03	1.67	1.76	2.11	2.64	2.42
	ξ	-0.32	-0.18	-0.29	-0.03	0.15	0.33	-0.34	-0.29	-0.11
	τ^2	0.21	0.03	0.01	0.42	0.22	0.04	1.26	0.33	0.06
LPPD ($\times 10^3$)		-23.11	**-22.60**	-22.95	**-22.81**	-22.82	-22.83	-22.82	-22.85	**-22.79**

Table 6.1: Comparison of the outcome parameter estimates for the Dirichlet process log-skewnormal mixture (DPLSM) in Models(A),(B), and (C), based on **Swautoins** dataset, across different error rates $R_{\epsilon_x} = 0.01/0.10/0.25$. The objective is to determine the optimal value of ζ.

These include: Model(A), gold standard DPLSM; Model(B), an erroneous DPLSM constructed using the NDB covariate $\mathbf{x}^*$; and Model(C), the DPLSM integrated with the Gustafson correction. The comparison highlights the variations in parameter estimations between the implementations of Model(B) and (C). Specifically, in Table 6.1, the marginal posterior means of the outcome parameters $\boldsymbol{\theta}$: $\{\beta_0, \beta_1, \beta_2, \sigma^2, \xi\}$, along with the estimated scale parameter τ^2, are compared across different scaling factor and varying levels of NDB error $R_{\epsilon_x} = 0.01/0.10/0.25$. Note

that only the estimation results corresponding to a range of values of the scaling factor $\zeta = 0.85 \sim 0.99$, which yield the highest LPPDs, are presented due to page limitations. This suggests that, for the **Swautoins** dataset, $\tau_j^2 : V(\mathbf{x}^*|\mathbf{x})$ might be explained by $0.01 \sim 0.15$ times $\widehat{\lambda}_j^2 : V(\mathbf{x}^*|\mathbf{z})$. In other words, $0.01 \sim 0.15$ of the variance of $\mathbf{x}^*|\mathbf{z}$ might correspond to the unknown variance of $\mathbf{x}^*|\mathbf{x}$, and thus $0.01 \sim 0.15$ times $\widehat{\lambda}_j^2 : V(\mathbf{x}^*|\mathbf{z})$ can serve as an approximate value for $\tau_j^2 : V(\mathbf{x}^*|\mathbf{x})$.

We begin with Model(B), which incorporates model risk from the NDB covariate, to examine how varying error levels influence the inference results of the outcome parameters $\boldsymbol{\theta} : \{\beta_0, \beta_1, \beta_2, \sigma^2, \xi\}$. We observe significant deviations from the gold-standard model. Consistent with Chapter 4, parameter estimations reveal a clear pattern across all error rates $R_{\epsilon_x} = 0.01/0.10/0.25$: minimal distortion at 0.01, with increasingly severe distortions as the error rate exceeds 0.1, complicating meaningful estimations. Specifically, the intercept β_0 tends to inflate with higher error rates while the coefficient β_1 for the NDB covariate $\mathbf{x}^*$ exhibits a downward trend. Notably, the increased NDB errors in the covariate $\mathbf{x}^*$ do not significantly inflate the scale σ^2 of the log-skewnormal outcome. This contrasts with the estimation results in Chapter 4, where increasing error rates gradually inflated the scale σ^2 in the log-normal outcome. Two key factors may explain this difference: the modeling frameworks differ, with Chapter 4 using a hierarchical GLM and this chapter employing a DPM framework; and the outcome distributions differ, as Chapter 4 features a log-normal distribution characterized by a scale σ^2 parameter while this chapter uses a log-skewnormal distribution with both a scale σ^2 parameter and a skewness ξ parameter. These factors contribute to the differences in distributional behavior observed in Table 6.1.

Turning to the inference results of Model(C) — the DPLSM with Gustafson correction — a key observation is that the corrected parameter estimates become significantly more stable and closely align with the gold standard, particularly at specific error rates and scaling factors. For instance, after applying the Gustafson correction, the greatest improvement observed at scaling factors spanning $0.90 <$

$\zeta < 0.95$ when the error rate is $R_{\epsilon_x} = 0.01$. This is supported by the highest LPPD value of -22,604.54, and the estimated parameter values fall within the credible intervals established by the gold standard model.

As we have discussed previously, the correction performance is directly linked to estimating $\tau_j^2 : V(\mathbf{x}^*|\mathbf{x})$. Given Equation (4.24), $\tau_j^2 : V(\mathbf{x}^*|\mathbf{x}) = (1-\zeta)\widehat{\lambda}_j^2 : V(\mathbf{x}^*|\mathbf{z})$, when $\mathbf{x}^*|\mathbf{z}$ is close to $\mathbf{x}^*|\mathbf{x}$, leading to a smaller scaling factor ζ. In such cases, the accurate correction results can be achieved when building the model on the NDB covariates with a higher error rate such as $R_{\epsilon_x} = 0.40$. This has been confirmed with **LGPIF** dataset in Chapter 4. Conversely, in this experiment, $\mathbf{x}^*|\mathbf{z}$ is not closely aligned with $\mathbf{x}^*|\mathbf{x}$, resulting in larger scaling factor ζ (ranging from 0.90 to 0.95), and the correction tends to be only effective when the error rate is very low, such as $R_{\epsilon_x} = 0.01$. These findings emphasize that both the error rate R_{ϵ_x} and scaling factor ζ critically affect the effectiveness of our Gustafson correction.

The further comparisons of Model(A) to Model(C) based on the **Swautoins** dataset are presented in Figure 6.7 and Table 6.2 for an error rate of 0.01, and in Figure 6.8 and Table 6.3 for an error rate of 0.25. These particular results are highlighted as they represent the best-performing scenarios (i.e., with the largest LPPDs and the lowest SSPEs and SAPEs). To enable a thorough comparison, Tables 6.2 and 6.3 provide (i) LPPD, (ii) SSPE, (iii) SAPE, (iv) D_{KL}, and (v) CTE for individual aggregate claims $S_h|\mathbf{X}$ for a given policy h. Results from the hierarchical GLM correction — Model(D) — are also included for comparison as the primary competing method. Figures 6.7 and 6.8 display histograms of the testing set, and compare the out-of-sample prediction curves from Model(C) and Model(D) to the gold standard curves from Model(A) and the erroneous curves from Model(B).

$\zeta = 0.95$ $R_{\epsilon_x} = 0.01$	Feature	Model(A): {DPM} Gold standard	Model(B): {DPM} with Model Risk	Model(C) {DPM} Gustafson correction	Model(D): {H.GLM} Gustafson correction
$f(S_h\|\mathbf{X})$	LPPD	-22,207.38	-22,767.67	**-22,604.54**	-23,107.04
	SSPE	269.13	276.03	274.62	276.01
	SAPE	196.22	199.52	198.86	199.51
	D_{KL}	0.00	0.56	0.40	0.90
	CTE 10%	360,198.30	389,567.01	386,342.10	290,029.10
	CTE 50%	628,291.51	681,351.34	675,617.28	501,164.91
	CTE 90%	2,492,340.03	1,764,142.13	2,679,876.32	1,834,556.02
	CTE 95%	4,037,532.05	2,524,463.30	4,311,994.24	2,881,864.13

Table 6.2: Comparison of predictive performances among three DPMs — Model(A), (B), (C) — and a Hierarchical GLM — Model(D) —, built using the **Swautoins** dataset with a covariate error rate of $R_{\epsilon_x} = 0.01$ and a scaling factor of $\zeta = 0.95$.

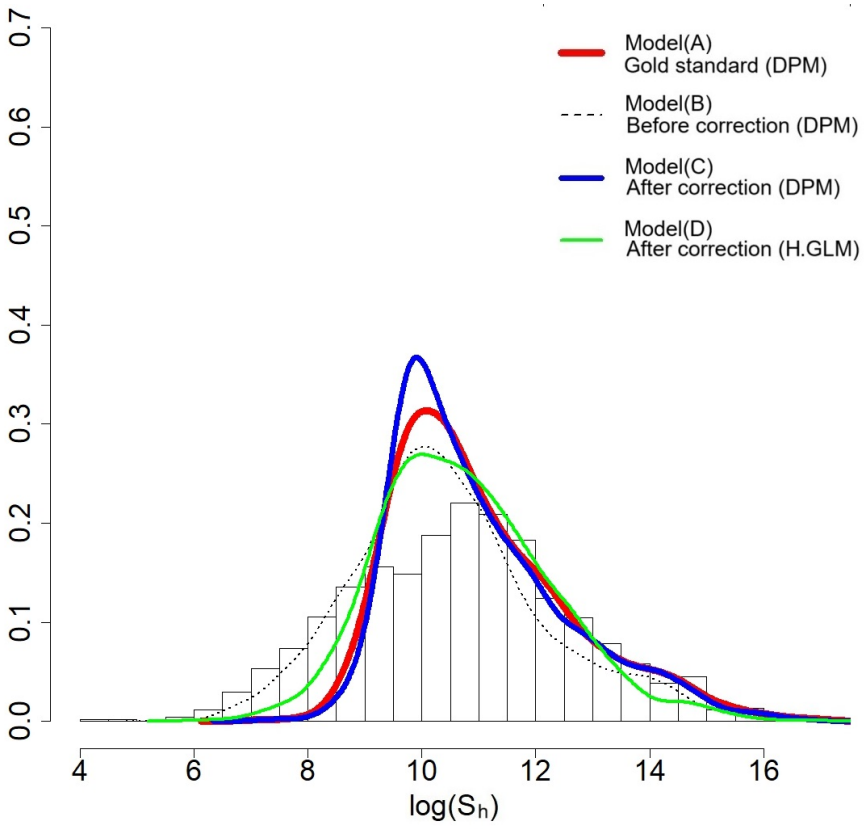

Figure 6.7: Fitted models based on the **Swautoins** dataset with the error rate $R_{\epsilon_x} = 0.01$ and the scaling factor $\zeta = 0.95$: a histogram of the observed aggregate claim amount S_h on a log scale and the out-of-sample predictive densities $f(lnS_h|\mathbf{X})$ obtained from Model(A), (B), (C) and (D).

$\zeta = 0.95$ $R_{\epsilon_x} = 0.25$	Feature	Model(A): {DPM} Gold standard	Model(B): {DPM} with Model Risk	Model(C) {DPM} Gustafson correction	Model(D): {H.GLM} Gustafson correction
$f(S_h\|\mathbf{X})$	LPPD	-22,207.38	-24,028.65	**-22,785.49**	-24,961.46
	SSPE	269.13	690.24	570.91	797.87
	SAPE	196.22	325.60	255.68	401.92
	D_{KL}	0.00	1.82	0.58	2.75
	CTE 10%	360,198.30	131,487.61	294,156.25	135,479.74
	CTE 50%	628,291.51	221,375.93	502,628.80	354,086.27
	CTE 90%	2,492,340.03	980,546.72	2,285,570.11	1,127,867.30
	CTE 95%	4,037,532.05	2,483,786.12	3,875,326.04	1,444,883.74

Table 6.3: Comparison of predictive performances among three DPMs — Model(A), (B), (C) — and a Hierarchical GLM — Model(D) —, built using the **Swautoins** dataset with a covariate error rate of $R_{\epsilon_x} = 0.25$ and a scaling factor of $\zeta = 0.95$.

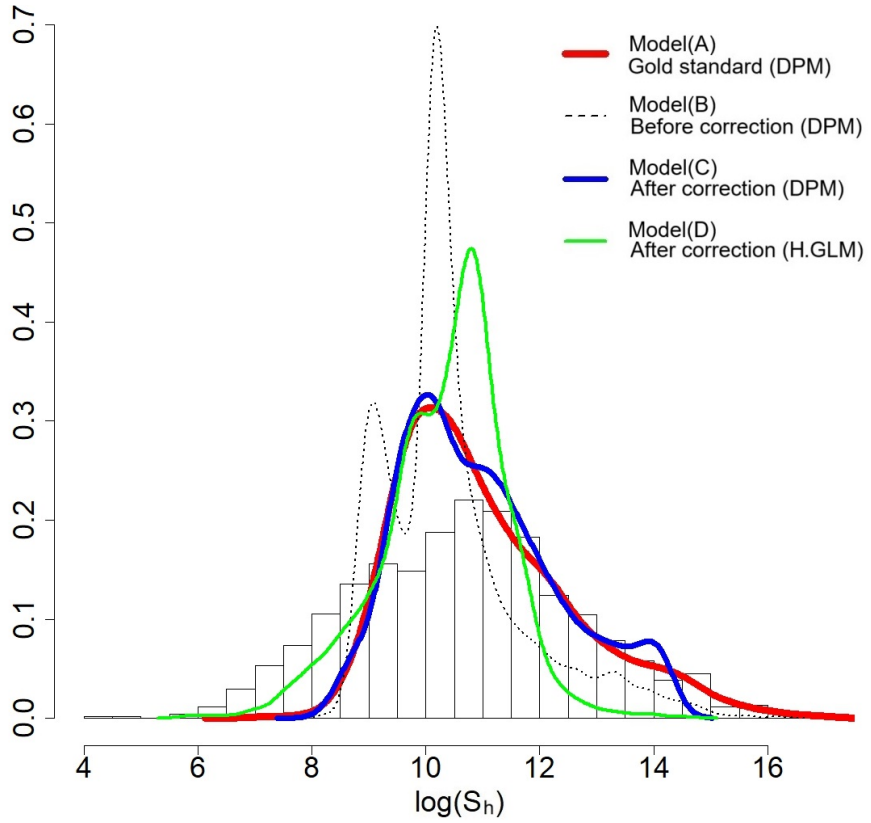

Figure 6.8: Fitted models based on the **Swautoins** dataset with the error rate $R_{\epsilon_x} = 0.25$ and the scaling factor $\zeta = 0.95$: a histogram of the observed aggregate claim amount S_h on a log scale and the out-of-sample predictive densities $f(lnS_h|\mathbf{X})$ obtained from Model(A), (B), (C) and (D).

In Table 6.2, with a covariate error rate of $R_{\epsilon_x} = 0.01$ and a scaling factor of $\zeta = 0.95$, the Gustafson correction based on DPLSM — Model(C) — achieves an

LPPD of -22,604.54, which is the closest value to the gold standard (-22,207.38). However, given the low error rate of $R_{\epsilon_x} = 0.01$, the LPPD values for the other models are quite similar. The differences in prediction errors, as measured by SSPE and SAPE, are minimal, and the same holds true for D_{KL}, as Models(B), (C), and (D) exhibit small divergences, suggesting that with only a 1% error rate, the impact of model risk is not substantial. In terms of CTEs measurement, however, Model(D) consistently shows lower CTE values, across various confidence levels. This suggests that while Model(D) may not be the most accurate in prediction, it could be a suitable option when the insurer's objective is to minimize attention to rare, high-loss events.

In Table 6.3, where the covariate's error rate is increased to $R_{\epsilon_x} = 0.25$, a similar comparison is made. Model(C) again outperforms the others in terms of LPPD, achieving a value of -22,785.49, closely approximating the gold standard. However, Model(D) performs significantly worse with -24,961.46, which is lower than the erroneous DPM, Model (B), which has a LPPD of -24,028.65. This suggests that the DPM model can outperform the hierarchical GLM, even in the presence of increased covariate error. The differences in SSPE and SAPE are also more pronounced, with Model(C) maintaining prediction errors close to Model(A), while Model(D) shows substantially higher values in both measures. A similar pattern is observed in D_{KL}, reinforcing that the DPM framework is generally more robust against higher model risk than the hierarchical GLM framework.

Regarding CTEs, as shown in Tables 6.2 and 6.3, Model(B) and Model(D) exhibit the lowest CTEs across various confidence levels, indicating thinner tail behavior. This suggests that model risk introduced by the covariate's NDB error deflates the upper tail. The Gustafson correction in Model(C) appears to effectively mitigate this, restoring the tail behavior to a more typical level, whereas the correction in Model(D) does not perform as well.

A more intuitive comparison is possible by examining the histograms of observed aggregate claims overlaid with the predictive densities from Models(A), (B), (C),

and (D) in Figures 6.7 and 6.8. The predictive densities are displayed as follows: Model(A) is shown by the red curve, Model(B) by the dotted curve, Model(C) by the blue curve, and Model(D) by the green curve.

With a low error rate of $R_{\epsilon_x} = 0.01$ in Figure 6.7, the curves generally align well, though Model(B) (with the dotted curve), which indicates the DPLSM built on the NDB covariate without any correction, exhibits slightly more spread. This is expected, as the increased NDB error raises variance. However, as the error rate increases ($R_{\epsilon_x} : 0.01 \rightarrow R_{\epsilon_x} : 0.25$) in Figure 6.8, Model(B) (dotted curve) shows a significant reduction in variance, marked by multiple peaks and bumps, indicating the effect of the increased model risk. In contrast, Model(C), which applies the Gustafson correction with the DPLSM, effectively reduces the bias introduced by the NDB covariate, as seen by the closer alignment of the blue curve with the red one. Nonetheless, this correction is not perfect: while Model(C) restores the general shape of Model(A), a gap remains, particularly in the upper tail.

Before proceeding with the next set of experiments, it is important to provide further clarity on the investigation of LPPDs across different combinations of error rates R_{ϵ_x} and scaling factors ζ. Previously, we highlighted the need to develop a practical guideline for identifying the optimal ζ without relying on gold standard data. We have proposed a rule of thumb — the "optimal scaling factor can be captured by the highest LPPD value." For clarity, LPPD serves as a measure of predictive performance, with its maximum value translating to the most optimal alignment between τ^2 and $\widehat{\lambda}^2$. In the experiment with the **Swautoins** dataset, we have demonstrated the validity of our argument through empirical results, confirming the relationship between LPPD values at the optimal ζ and the predictive accuracy. This was explored across combinations of error rates $R_{\epsilon_x} = 0.01/0.10/0.25$ and scaling factors $\zeta = 0.85 \sim 0.99$ as shown in Tables 6.1, 6.2, 6.3, and Figures 6.7, 6.8.

This naturally raises the question: why focus on specific combinations of error rates and scaling factors, such as $R_{\epsilon_x} = 0.01/0.10/0.25$ and $\zeta = 0.85 \sim 0.99$?

The answer lies in the computational burden associated with higher granularity in these evaluations. LPPD heatmaps in Figure 6.9, based on the **Swautoins** dataset, provide a broad overview of the entire investigation, without delving into finer granularities, such as error rates between 0.10, 0.12, 0.18, and so on, or scaling factors between 0.20, 0.25, 0.27, and so on. Due to computational limitations, we focus on identifying the "sweet spot" where LPPD values peak. In Figure 6.9, as indicated by the blue box, for an error rate of 0.01, the optimal scaling factor range is 0.7 to 0.9, yielding maximum LPPD values, while for an error rate of 0.25, the optimal range is 0.8 to 0.9. Based on this observation, we narrow our focus to these windows of interest, with results presented in the main text in Tables 6.1, 6.2, 6.3, and Figures 6.7, 6.8. This approach is consistently applied throughout the remainder experiments in this chapter.

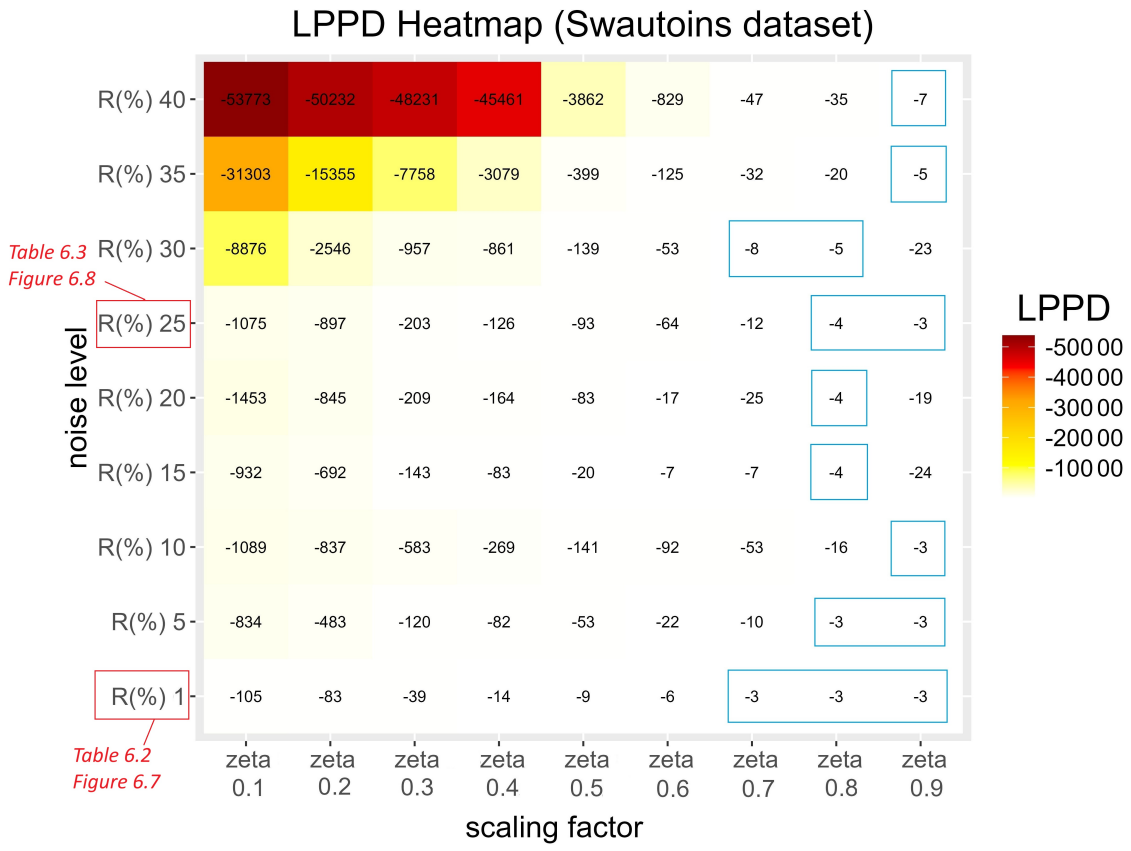

Figure 6.9: A heatmap showing LPPD values across combinations of the scaling factors (from 0.1 to 0.9) and the error rates (from 0.01 to 0.40). The color gradient reveals regions of better (white) or worse (red) predictive performance of our DPLSM — Model(C) — across these settings. The LPPD values are shown in units of 10K.

6.4.4 Results with Brvehins2 ($H = 62,512$)

For this dataset, we construct a training set containing 60,063 instances of aggregate claim outcomes and corresponding covariates $(S, \mathbf{X})$ to develop the model, alongside a test set with 2,449 records $(S', \mathbf{X}')$ for assessing prediction accuracy. Unlike the **Swautoins** dataset, which is relatively small, the **Brvehins2** dataset is quite large, with over 50,000 observations. Due to its scale, the parallel MCMC simulation is employed to efficiently process the data, and the results are combined at the end.

There are two key points to consider in this experiment. First, building on the experiment results in Section 6.4.3, this study aims to develop a practical method for identifying the optimal scaling factor $0 < \zeta < 1$ in the absence of gold standard data. We explore the hypothesis that, without access to gold standard data, the largest LPPD value indicates the optimal scaling factor ζ, reflecting the unknown relationship between $\tau_j^2 : V(\mathbf{x}^*|\mathbf{x})$ and $\widehat{\lambda}_j^2 : V(\mathbf{x}^*|\mathbf{z})$. Second, we previously suggested that both the error rate R_{ϵ_x} and scaling factor ζ influence the effectiveness of the Gustafson correction. Intuitively, error correction should be more straightforward (or easy) when the error rate is low. However, as the error rate increases in the NDB covariate $\mathbf{x}^*$, the correction process becomes less clear. Thus, we pay attention to how identifying the optimal scaling factor ζ at higher error rate affects the overall quality of the Gustafson correction. As outlined earlier, Model(A) serves as the gold standard, built on the DPM framework. Model(B) represents a flawed DPM model, highlighting the impact of NDB model risk. Model(C) incorporates the Gustafson correction within the DPM framework, while Model(D) applies it within a Hierarchical GLM framework. The performance of Models(B) through (D) is evaluated across three error rate scenarios: $R_{\epsilon_x} = 0.01, R_{\epsilon_x} = 0.1$, and $R_{\epsilon_x} = 0.25$. Additionally, the reliability of $\mathbf{z}$ as a substitute for $\mathbf{x}$ is tested using various scaling factors $\zeta = \{0.10, \cdots, 0.99\}$.

Gold standard [Model(A)]: The benchmark model, referred to as Model (A) in Figure 6.4, is analyzed. Serving as the error-free gold standard, its predictive distribution $f(lnS_h|\mathbf{X})$ is obtained using the true covariates: $\mathbf{z}$ and $\mathbf{x}$. In Figure 6.10, we

showcase the last 100 scenarios of predictive density for the aggregate claim amount on a log scale $lnS_h|\mathbf{X} \sim \mathbf{SN}(\mathbf{X}^T\boldsymbol{\beta},\ \sigma^2,\ \xi)$. This is accompanied by an MCMC trace plot, which corresponds to the last 100 scenarios selected from the 10,000 samples (LPPD values) in the Gibbs sampler. With these scenarios, the ultimate predic-

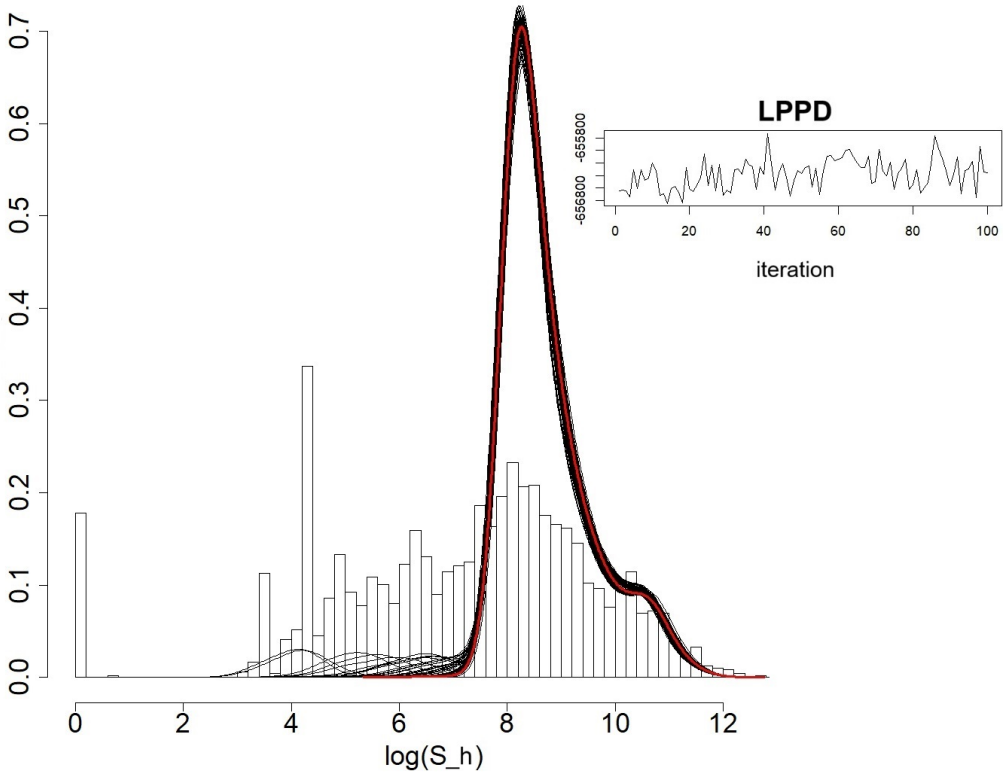

Figure 6.10: Model(A) Result with the **Brvehins2** dataset: a histogram of the observed aggregate claim amount on a log scale and the last 100 out-of-sample predictive density scenarios $f(lnS_h|\mathbf{X})$ (black curves) overlaid. The average predictive density, represented by the red curve, results from converged estimates.

tive distribution can be computed using the formula in Equation (6.3). Figure 6.10 demonstrates that the predictive curves align well with the observed aggregate claim amounts in the upper tail, but tend to overlook the lower tail observations. This is due to the inclusion of covariates, such as $\mathbf{z}$:*Affiliation* and $\mathbf{x}$:*Ln-TotalExp*, which are more closely associated with higher claim values. This discrepancy is particularly clear when comparing the DPM model with covariates to the version without covariates, where the latter successfully captures the full range of the histogram. In the model with covariates, the predictive distribution becomes more concentrated around the mean and the upper tail, focusing on higher aggregate claim amounts.

As a result, the lower tail is overshadowed by the influence of larger values, as the covariates hold more explanatory power in the upper range.

Gustafson correction [Model(B) vs Model(C) vs Model(D)]: Continuing in a similar manner to the previous experiment, we compare the performance of Model(C), which applies the Gustafson correction within the DPM framework, with that of Model(D), where the Gustafson correction is implemented within a hierarchical GLM framework. Table 6.4 summarizes the sensitivity analysis results for the optimal scaling factor ζ across three DPM models: Model(A), the gold standard; Model(B), the erroneous; and Model(C), the Gustafson correction. The comparison reveals variations in parameter estimations across these models under different scaling factors $\zeta = \{0.10, \cdots, 0.99\}$ and error rates $R_{\epsilon_x} = 0.01/0.10/0.25$.

Model	$\boldsymbol{\theta}, \tau^2$	Parameter estimates $\boldsymbol{\theta}$: $\{\beta_0, \beta_1, \beta_2, \sigma^2, \xi\}$ and τ^2								
Model(A) Gold standard	β_0	10.17 with 95% Credible Interval: $\{10.12 \leq \beta_0 \leq 10.18\}$								
	β_1	0.39 with 95% Credible Interval: $\{0.38 \leq \beta_1 \leq 0.40\}$								
	β_2	-1.95 with 95% Credible Interval: $\{-1.98 \leq \beta_2 \leq -1.94\}$								
	σ^2	11.84 with 95% Credible Interval: $\{11.02 \leq \sigma^2 \leq 12.97\}$								
	ξ	-1.11 with 95% Credible Interval: $\{-1.28 \leq \xi \leq -0.97\}$								
		Error rate in $\mathbf{x}^*$: 0.01			Error rate in $\mathbf{x}^*$: 0.10			Error rate in $\mathbf{x}^*$: 0.25		
Model(B) Before correction	β_0	10.14			10.11			10.05		
	β_1	0.40			0.13			0.06		
	β_2	-1.92			-1.37			-1.15		
	σ^2	7.21			7.62			14.28		
	ξ	-0.81			-0.97			-1.56		
		ζ : 0.7	ζ : 0.75	ζ : 0.8	ζ : 0.7	ζ : 0.75	ζ : 0.7	ζ : 0.7	ζ : 0.75	ζ : 0.8
Model(C) After correction	β_0	10.12	10.09	10.15	10.09	10.01	10.02	10.13	10.11	10.10
	β_1	0.38	0.39	0.37	0.19	0.11	0.11	0.40	0.46	0.56
	β_2	-1.95	-1.89	-1.87	-1.52	-1.22	-1.21	-1.97	-1.90	-1.89
	σ^2	10.88	9.15	8.12	9.75	8.67	8.22	10.68	8.42	7.42
	ξ	-0.91	-0.69	-0.44	-0.71	-0.65	-0.61	-0.97	-0.85	-0.74
	τ^2	0.33	0.54	0.09	3.78	2.85	0.46	5.44	8.42	1.62
LPPD ($\times 10^4$)		**-66.85**	-66.89	-66.86	-66.95	-67.03	-67.10	**-65.99**	-66.93	-67.95

Table 6.4: Comparison of the outcome parameter estimates for the Dirichlet process log-skewnormal mixture (DPLSM) in Models(A),(B), and (C), based on **Brvehins2** dataset, across different error rates $R_{\epsilon_x} = 0.01/0.10/0.25$. The objective is to determine the optimal value of ζ.

In Table 6.4, the marginal posterior means of the outcome parameters $\boldsymbol{\theta}$: $\{\beta_0, \beta_1, \beta_2, \sigma^2, \xi\}$ and the estimated scale parameter τ^2 are compared across different scaling factors and NDB error levels $R_{\epsilon_x} = 0.01/0.10/0.25$. Only the results for scaling factors $\zeta = 0.7 \sim 0.8$, which yield the highest LPPDs, are shown due to space constraints. This suggests that, for the **Brvehins2** dataset, $\tau_j^2 : V(\mathbf{x}^*|\mathbf{x})$ could be approximately $0.2 \sim 0.3$ times of $\widehat{\lambda}_j^2 : V(\mathbf{x}^*|\mathbf{z})$.

To examine the impact of bias from NDB errors, Model(B), which incorporates the model risk associated with the NDB covariate, shows increasing distortion in parameter estimates as the error level rises. This trend contrasts with the estimation results from earlier experiments in Section 6.4.3. For instance, both the intercept β_0 and the slope β_1 for NDB covariate $\mathbf{x}^*$ exhibit a clear downward trend with the increasing error rate. This differs from previous results where the intercept β_0 showed an upward trend while β_1 for NDB covariate $\mathbf{x}^*$ still declined with rising error. However, the estimation behavior of the scale σ^2 and skewness ξ parameters remains consistent with observations in the previous section: σ^2 continues to fluctuate erratically, while ξ shows a steady decline as the covariate error increases. Previously, we argue that this was attributed to the distributional properties of the underlying log-skewnormal outcome. As for the corrected estimates produced by Model(C) — the DPLSM with Gustafson correction —, consistent with the earlier experiments in Section 6.4.3, specific scaling factor and error rates tend to yield more accurate corrections, closely aligning with those of the gold standard, Model(A). Notably, the most significant enhancement is seen at a scaling factor of $\zeta = 0.7$ when the error rate is $R_{\epsilon_x} = 0.25$. This is supported by the highest LPPD value of -661,210, with the estimated parameters falling within the credible intervals set by Model(A).

As highlighted earlier, both the error rate R_{ϵ_x} and scaling factor ζ are key determinants of the effectiveness of the Gustafson correction. In the experiment using the **LGPIF** dataset in Chapter 4, $V(\mathbf{x}^*|\mathbf{z})$ closely matches $V(\mathbf{x}^*|\mathbf{x})$, leading to a relatively smaller scaling factor ζ (ranging from 0.5 to 0.7 based on LPPDs). This alignment allows for accurate correction even at higher error levels, such as $R_{\epsilon_x} = 0.40$.

In contrast, the **Swautoins** dataset in the previous section shows a larger scaling factor ζ (ranging from 0.90 to 0.95 based on LPPDs) due to the weaker alignment between the variances, making corrections effective only at low error rates, such as $R_{\epsilon_x} = 0.01$. For the **Brvehins2** dataset in this section, moderate alignment leads to a scaling factor ζ (ranging from 0.7 to 0.8 based on LPPDs), enabling corrections to perform well at moderately high error rates, such as $R_{\epsilon_x} = 0.25$. However, beyond this threshold, when the error rate exceeds $R_{\epsilon_x} > 0.25$, the correction's performance begins to decline.

A detailed comparison of DPM model performance using the **Brvehins2** dataset is illustrated in Figure 6.11 and Table 6.5 for an error rate of 0.01, and in Figure 6.12 and Table 6.6 for an error rate of 0.25. These specific results are emphasized because they correspond to the top-performing cases (i.e., with the largest LPPDs and the lowest SSPEs and SAPEs). Tables 6.5 and 6.6 summarize (i) LPPD, (ii) SSPE, (iii) SAPE, (iv) D_{KL}, and CTEs for individual aggregate claims $S_h|\mathbf{X}$.
Results from the hierarchical GLM adjustment are also included for a baseline comparison. Figures 6.11 and 6.12 show the testing set histograms and juxtapose the out-of-sample prediction curves from Model(C) and Model(D) against the reference curve from Model(A) and the model-risk-induced curve from Model(B).

In Table 6.5, with a covariate error rate set at $R_{\epsilon_x} = 0.01$ and a scaling factor of $\zeta = 0.7$, the Gustafson correction applied in the DPLSM framework (Model(C)) achieves an LPPD of -663,949.46, which is the closest approximation to the benchmark, Model(A), of -656,377.40. At this relatively low error rate, the LPPD values for the other models exhibit minimal variation, accompanied by slight differences in prediction errors — SSPE and SAPE —, and minor divergences in D_{KL}. This reinforces the observations from the previous experiment with the **Swautoins** dataset, indicating that the impact of the model risk arising from the NDB covariate remains relatively minimal at a 1% error rate.

$\zeta = 0.7$ $R_{\epsilon_x} = 0.01$	Feature	Model(A): {DPM} Gold standard	Model(B): {DPM} with Model Risk	Model(C) {DPM} Gustafson correction	Model(D): {H.GLM} Gustafson correction
$f(S_h\|\mathbf{X})$	LPPD	-656,377.40	-668,567.44	**-663,949.46**	-679,145.69
	SSPE	12,420.39	17,173.60	12,872.54	14,426.15
	SAPE	4,285.84	4,757.15	4,294.06	4,704.68
	D_{KL}	0.00	1.21	0.75	2.27
	CTE 10%	2,166.79	1,992.94	1,965.21	3,837.18
	CTE 50%	3,123.79	2,653.75	2,729.23	4,011.48
	CTE 90%	7,546.59	6,670.11	9,683.39	6,702.44
	CTE 95%	10,136.34	9,891.04	12,418.92	8,545.13

Table 6.5: Comparison of predictive performances among three DPLSMs — Model(A), (B), (C) — and a hierarchical GLM — Model(D) —, built using the **Brvehins2** dataset with a covariate error rate of $R_{\epsilon_x} = 0.01$ and a scaling factor of $\zeta = 0.7$.

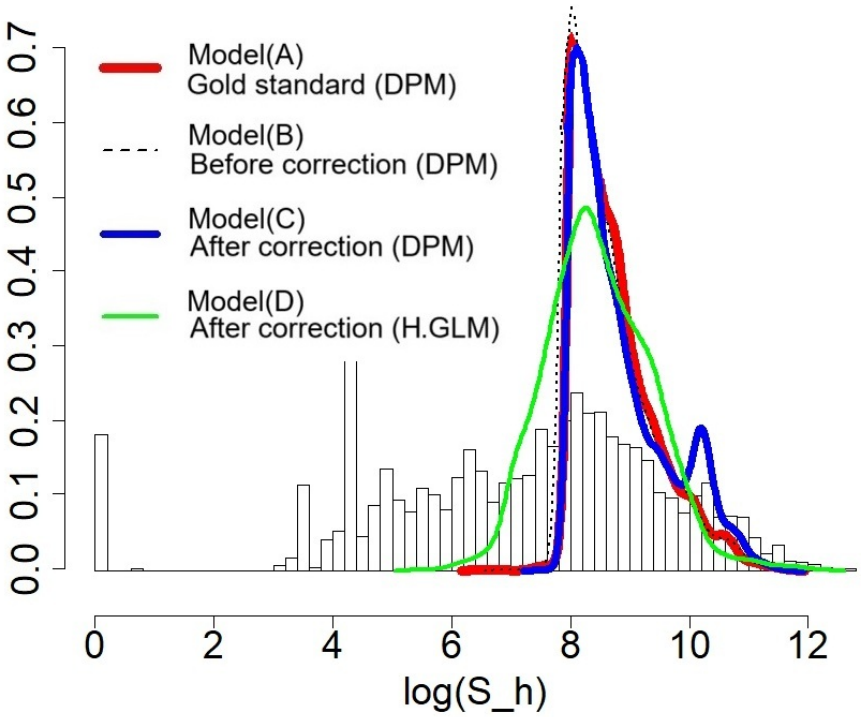

Figure 6.11: Fitted models based on the **Brvehins2** dataset with the error rate $R_{\epsilon_x} = 0.01$ and the scaling factor $\zeta = 0.7$: a histogram of the observed aggregate claim amount S_h on a log scale and the out-of-sample predictive densities $f(lnS_h|\mathbf{X})$ obtained from Model(A), (B), (C) and (D).

$\zeta = 0.7$ $R_{\epsilon_x} = 0.25$	Feature	Model(A): {DPM} Gold standard	Model(B): {DPM} with Model Risk	Model(C) {DPM} Gustafson correction	Model(D): {H.GLM} Gustafson correction
$f(S_h\|\mathbf{X})$	LPPD	-656,377.40	-688,787.98	**-659,956.12**	-698,014.49
	SSPE	12,420.39	18,799.83	14,860.32	16,129.76
	SAPE	4,285.84	4,988.12	4,584.58	5,018.57
	D_{KL}	0.00	3.24	0.35	4.16
	CTE 10%	2,166.79	2,604.73	1,895.45	2,118.19
	CTE 50%	3,123.79	2,394.51	2,989.70	3,891.54
	CTE 90%	7,546.59	6,834.35	8,558.94	5,758.80
	CTE 95%	10,136.34	9,059.72	11,306.08	9,850.79

Table 6.6: Comparison of predictive performances among three DPLSMs — Model(A), (B), (C) — and a hierarchical GLM — Model(D) —, built using the **Brvehins2** dataset with a covariate error rate of $R_{\epsilon_x} = 0.25$ and a scaling factor of $\zeta = 0.7$.

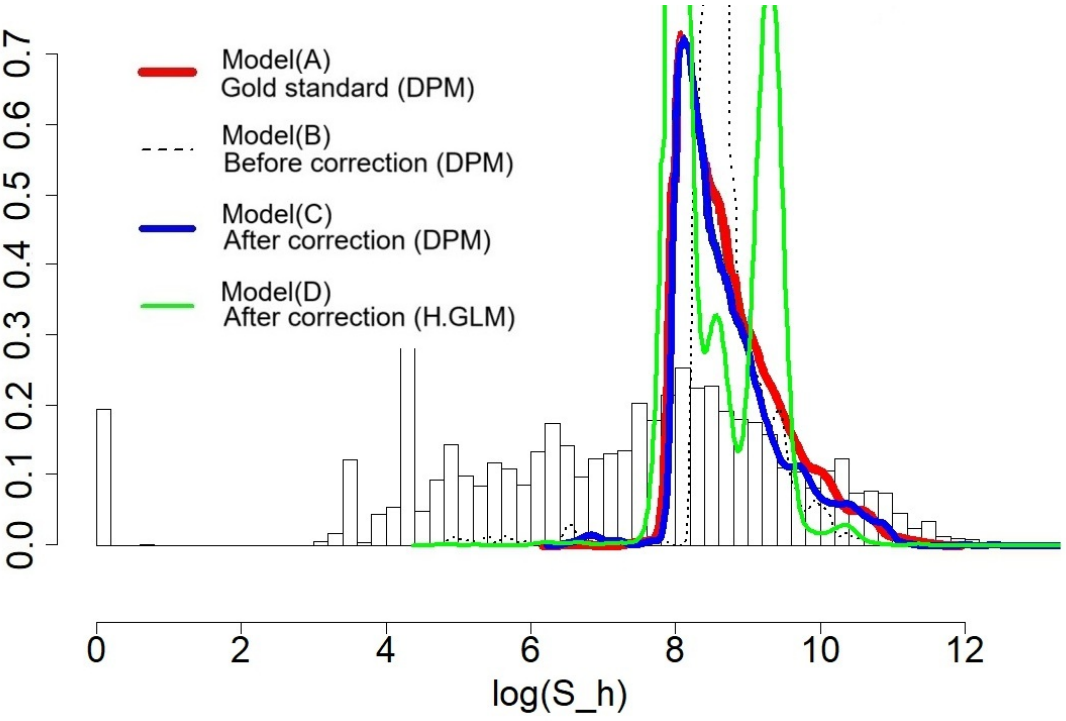

Figure 6.12: Fitted models based on the **Brvehins2** dataset with the error rate $R_{\epsilon_x} = 0.25$ and the scaling factor $\zeta = 0.7$: a histogram of the observed aggregate claim amount S_h on a log scale and the out-of-sample predictive densities $f(lnS_h|\mathbf{X})$ obtained from Model(A), (B), (C) and (D).

However, in contrast to the previous experiment, where Models(B) and (D) consistently showcased the lowest CTE values across various confidence levels, the results in Table 6.5 reveal a slightly different trend. Here, Models(B) and (D) exhibit

higher CTE values at lower confidence levels, but they ultimately yield substantially lower CTEs at higher confidence levels such as CTE 95%, indicating that the NDB covariate weakens the upper tails.

Table 6.6 reveals that when the covariate error rate increases ($R_{\epsilon_x} : 0.01 \rightarrow R_{\epsilon_x} : 0.25$), Model(C) continues to perform strongly, achieving an LPPD of -659,956.12. It is interesting to note that this result aligns much more closely with the gold standard compared to Model(C)'s performance at the lower error rate $R_{\epsilon_x} = 0.01$. In stark contrast, Model(D) in Table 6.6 shows a significant decline, registering an LPPD of -698,014.49, which is inferior to the results from the erroneous DPM, Model(B) with a LPPD of -688,787.98. This indicates that even with heightened covariate error, the DPM framework retains its superiority over the hierarchical GLM framework. A similar trend is evident in SSPE, SAPE, and D_{KL}, and once again we can conclude that the DPM framework is generally more efficient against elevated model risk when compared to the hierarchical GLM approach. In Table 6.6, Models (B) and (D) exhibit higher CTEs at lower confidence levels, closely aligning with the benchmark model's lower tail behavior. However, as confidence levels rise, their CTEs diverge noticeably, leading to a pronounced deflation in the upper tail. Similar to the experiment results with a lower error rate $R_{\epsilon_x} = 0.01$, the Gustafson correction applied within the DPM framework in Model(C) at the higher error rate $R_{\epsilon_x} = 0.25$ effectively resolves this issue, restoring the tail behavior in line with the gold standard model, Model(A).

Figures 6.11 and 6.12 present the histograms of observed aggregate claims, overlaid with the predictive densities from Models(A), (B), (C), and (D), allowing for a more intuitive interpretation of the results based on the curve alignments under varying error rate scenarios: $R_{\epsilon_x} = 0.01$ in Figures 6.11 and $R_{\epsilon_x} = 0.25$ in Figures 6.12. Here, Model(A) is depicted in red, Model(B) in dotted lines, Model(C) in blue, and Model(D) in green. At the error rate of $R_{\epsilon_x} = 0.01$, the predictive densities largely converge; however, Model(D), which employs the hierarchical GLM with the Gustafson correction, exhibits a broader spread due to increased variance,compared

to other DPM based curves. In contrast, Figure 6.12 shows that when the error rate rises to $R_{\epsilon_x} = 0.25$, both Model(D) and Model(B), the erroneous DPM built on the NDB covariate, reveal a significant shrinkage in variance, characterized by multiple peaks that underscore the challenges associated with heightened model risk. Notably, in both error rate scenarios, Model(C) effectively leverages the Gustafson correction to mitigate the bias linked to the NDB covariate, thereby aligning more closely with Model(A).

Across two numerical experiments utilizing the **Swautoins** dataset with a moderate sample size of $h = 1,799$ and the **Brvehins2** dataset with a considerably larger sample size of $h = 62,512$, it becomes apparent that the model risk tied to the NDB covariate primarily results in the deflation of the distribution's upper tail. This deflation leads to a substantial underestimation of rare and extreme events, posing significant risks in practical scenarios. The results demonstrate that the Gustafson correction enhanced by the DPM framework generally outperforms the Gustafson correction based on the hierarchical GLM framework in addressing this tail deflation issue and in prediction accuracy. This underscores the DPM framework's superiority in providing robust modeling methodology for risk management.

6.4.5 Discussion

In this chapter, we integrated the Gustafson correction method into the log-skewnormal DPM framework to enhance risk premium prediction in the presence of the covariate-based model risk. The model risks addressed in this chapter correspond to the following research questions: RQ1.1 heterogeneity, RQ1.2 convolution error, RQ1.3 scalability issue, and RQ2.2 NDB covariate.

For **RQ1.1**, we resolved the heterogeneity issue by leveraging parameter-free clustering within the DPM framework. In relation to **RQ1.2**, we employed a log-skewnormal density for the sum of log-normal outcome to accommodate the aggregate claim amounts. On top of that, the Lindeberg's approximation for the sum

of log-skewnormal outcomes was also examined. Regarding **RQ1.3**, large-scale implementation was achieved using the shard-and-anchor technique combined with parallel MCMC simulation. For **RQ2.2**, we developed a hybrid DPM framework incorporating the Gustafson correction to mitigate the model risk introduced by the NDB covariate.

Our experiments clearly demonstrated the effectiveness of the Gustafson correction within the DPLSM framework in mitigating model risk and restoring the original properties of the predictive distribution. The core of this correction technique is our novel rule of thumb, developed to manage model risk from NDB covariates, especially in the absence of gold standard data. This rule is derived from the foundations of Gustafson's joint modeling theory, where prior knowledge of the variance $\tau^2 : V(\mathbf{x}^*|\mathbf{x})$ plays a pivotal role. We approximate this variance using the currently available $\widehat{\lambda}^2 : V(\mathbf{x}^*|\mathbf{z})$ and the scaling factor ζ, which modulates the strength of the correction. Our experimental results indicate that both the error rate R_{ϵ_x} in the NDB covariate and the choice of scaling factor $0 < \zeta < 1$ significantly influence the effectiveness of the Gustafson correction, suggesting a delicate consideration between these two. As elaborated in Section 4.4.3, we discovered that the optimal value of ζ can be empirically determined by maximizing the LPPD.

A notable trend emerged across our experiments: When the optimal scaling factor ζ appears to be low (i.e., $V(\mathbf{x}^*|\mathbf{z})$ and $V(\mathbf{x}^*|\mathbf{x})$ is highly related), the Gustafson correction become effective across various level of error rate in the NDB covariate, but once the error rate surpasses certain point such as $R_{\epsilon_x} > 0.25$, as observed in the **Swautoins** and **Brvehins2** experiments, the performance of the DPM framework begins to degrade due to excessive cluster generation inherent to its parameter-free clustering mechanism. In such cases, hierarchical GLM could be considered, although its predictive power generally falls short when compared to the DPM. In contrast, when the optimal scaling factor ζ appears to be high (i.e., $V(\mathbf{x}^*|\mathbf{z})$ and $V(\mathbf{x}^*|\mathbf{x})$ is less correlated), the Gustafson correction based on both DPM and hierarchical GLM become only effective at the lower error rate such as $R_{\epsilon_x} \leq 0.10$.

In other words, without the strong degree of relation between $V(\mathbf{x}^*|\mathbf{z})$ and $V(\mathbf{x}^*|\mathbf{x})$, the correction is only effective when the error rate is very low.

For instance, in the **LGPIF** experiment from Chapter 4, which was built on a hierarchical GLM, we observed that $V(\mathbf{x}^*|\mathbf{z})$ and $V(\mathbf{x}^*|\mathbf{x})$ are reasonably-well correlated, and thus relatively small scaling factors $0.5 \leq \zeta \leq 0.7$ produced accurate corrections, even at a high error rate of $R_{\epsilon_x} = 0.40$, which is difficult for the DPM framework due to the over-generation of clusters. On the other hand, the **Swautoins** experiment in this chapter, which employed the DPM framework, revealed that the method proves most successful at lower error rates, such as $R_{\epsilon_x} = 0.01$ with larger scaling factors of $0.90 \leq \zeta \leq 0.95$ due to the weak degree of relation between $V(\mathbf{x}^*|\mathbf{z})$ and $V(\mathbf{x}^*|\mathbf{x})$. Similarly, in the **Brvehins2** experiment in this chapter, we observed moderate alignment between $V(\mathbf{x}^*|\mathbf{z})$ and $V(\mathbf{x}^*|\mathbf{x})$, making the scaling factor $0.70 \leq \zeta \leq 0.80$ optimal. This leads to the correction remaining effective for error rates within $0.01 \leq R_{\epsilon_x} \leq 0.25$. Hence, the empirical results from the three experiments — **LGPIF**, **Swautoins**, and **Brvehins2** — support the idea that the degree of relation (represented by the scaling factor ζ) between $V(\mathbf{x}^*|\mathbf{z})$ and $V(\mathbf{x}^*|\mathbf{x})$ plays a crucial role in determining the range of error rates that can be effectively managed by the Gustafson correction within the Bayesian framework.

Figure 6.13 visually summarizes these insights, highlighting how the interplay between error rate R_{ϵ_x} and scaling factor ζ dictates the overall effectiveness of the correction strategy. Without gold standard data, it can be difficult to determine how well $V(\mathbf{x}^*|\mathbf{z})$ and $V(\mathbf{x}^*|\mathbf{x})$ are correlated, since the true covariate $\mathbf{x}$ remains unknown. However, as the scaling factor ζ reflects the degree of this relation, our experiment reveals that the value of ζ that maximizes the LPPDs can be viewed optimal. This simplifies the process. By plotting the optimal ζ in the quadrant of Figure 6.13, we can visually assess whether our Gustafson correction method can handle higher error rates in the NDB covariate or not. The only information required before modeling is the accurate error rate of the NDB covariate in question — no gold standard data required. In short, selecting the right combination of ζ and the

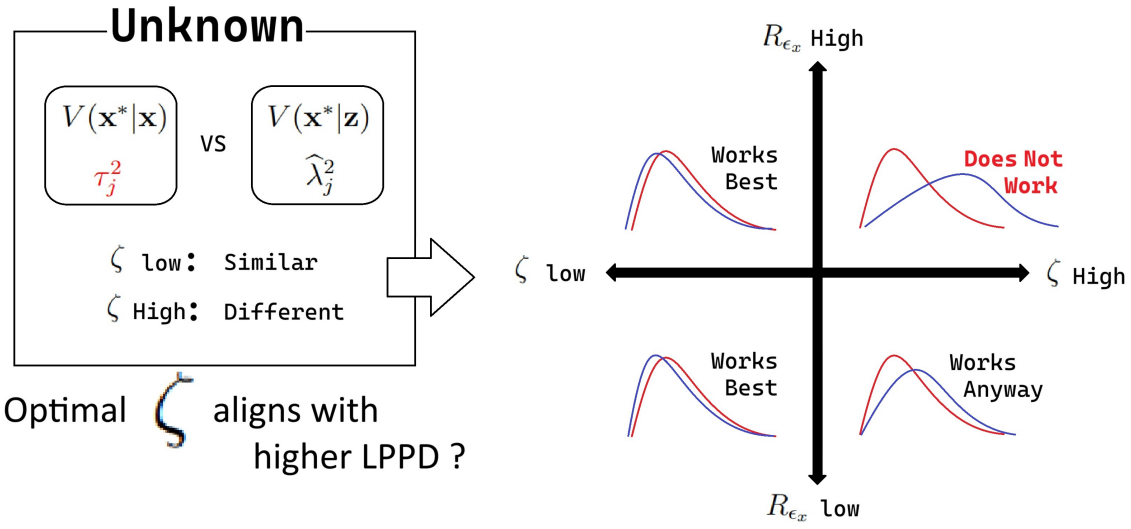

Figure 6.13: Findings regarding the Gustafson correction: The correction performance is determined by the position of the optimal scaling factor ζ within the quadrant defined by R_{ϵ_x} and ζ.

error rate R_{ϵ_x} for our Gustafson correction based on the Bayesian framework can be the key to achieving reliable risk premium predictions in the presence of NDB covariate-related model risk.

Beyond these findings, we present a set of additional assumptions and concerns that emerged from the numerical experiments conducted throughout this thesis (Chapters 4, 5, and 6) in the next chapter, specifically in Section 7.3.

Chapter 7

Discussion and Conclusion

In summary, this thesis proposes a hybrid Bayesian risk premium model designed to effectively mitigate key model risks. Customized approaches are employed to address specific challenges, such as heterogeneity (RQ1.1), convolution error (RQ1.2), scalability (RQ1.3), MAR covariates (RQ2.1), and NDB covariates (RQ2.2). By harnessing the advantages of both parametric and nonparametric Bayesian frameworks, we have demonstrated that integrating these targeted strategies enhances the model's applicability across a wider range of analytical contexts.

Through numerical experiments, we validated the effectiveness of our hybrid Bayesian methods, which consistently outperformed conventional techniques such as SIMEX (for error correction) and Multiple Imputation (for handling missing data) in both accuracy and robustness. Regarding robustness, the conventional procedure occasionally struggled with convergence when handling incomplete data, often triggering warning messages in R. In contrast, our Gibbs sampler consistently converged to the stationary posterior distribution, albeit sometimes requiring a long runtime. The computational performance of our hybrid Gibbs sampler varied notably across different hardware configurations and sample sizes. On a single Intel Core i7-1185G7 CPU @ 3.00GHz, the sampler completed 60,000 iterations in about 4 hours for the hierarchical model experiments in Chapter 4, which involved a small sample size (fewer than 2,000 observations). For the Dirichlet process model experiments in Chapter 5, which had a moderate sample size (around 5,000), the same setup required ap-

proximately 5 hours for 30,000 iterations. However, the large-scale Dirichlet process model experiment in Chapter 6, with a dataset of around 60,000 observations, demanded more substantial computational resources. On a high-performance cluster equipped with a 23 Intel(R) Xeon(R) Gold 6230 CPU @ 2.10GHz, the hybrid MCMC took about 9 hours to complete 30,000 iterations. Notably, this experiment could not be executed on a single CPU due to the memory and processing demands imposed by the larger dataset. The C++ and R code used for these experiments is publicly available on GitHub: https://github.com/mainkoon81?tab=repositories.

Notably, our experiments also demonstrate that the practical guidelines developed - particularly for the Gustafson correction process - allow for effective parameter adjustments without requiring gold standard data, enhancing the applicability of our hybrid Bayesian model such as hierarchical GLM/DPM in real-world contexts. In particular, our finding highlight that the interplay between the error rate R_{ϵ_x} in the NDB covariate and the scaling factor ζ is crucial for the effectiveness of the Gustafson correction within the Bayesian framework. A low ζ, indicating a strong alignment between $V(\mathbf{x}^*|\mathbf{z})$ and $V(\mathbf{x}^*|\mathbf{x})$, allows for appropriate corrections across a range of different error rates. Conversely, a high ζ correlates with weaker alignment between $V(\mathbf{x}^*|\mathbf{z})$ and $V(\mathbf{x}^*|\mathbf{x})$, limiting the correction's effectiveness to low error rates. Overall, this interplay between the scaling factor ζ and the error rate R_{ϵ_x} defines the operational boundaries for applying the Gustafson correction within the Bayesian framework to improve risk premium predictions.

7.1 Our Contributions

It is known that covariate-based model risks fundamentally distort risk assessments in the insurance practice, resulting in biased premium pricing, heightened uncertainty, and potential regulatory compliance challenges (Aggarwal et al. 2016). As a consequence, insurers often feel compelled to inflate premium prices to compensate for these unknown risks, which can diminish the competitiveness of their products.

This thesis makes a notable contribution by establishing a unified Bayesian

framework that integrates a variety of covariate-based model risk mitigation strategies, thereby enhancing the reliability and practicality of risk premium estimation. As discussed in Chapter 2 and Chapter 3, existing literature offers independent solutions to address specific types of model risks, such as heterogeneity, convolution, scalability, missing data, and measurement error. However, bridging these distinct challenges into a cohesive framework remains a far more complex endeavor. This thesis tackles that very challenge, advancing both the theoretical and practical landscape for managing covariate-based model risks in insurance pricing by integrating these disparate issues into a unified approach.

To elaborate, first, it introduces the integration of the data augmentation technique and the Gustafson correction technique within both Bayesian parametric/nonparametric framework. This approach effectively mitigates issues related to heterogeneity and bias arising from a Missing At Random (MAR) or a Non-Differential Berkson (NDB) covariate, particularly marking the first known application of Gustafson's correction in this context. As a result, it allows for improved bias reduction in risk premium modeling without relying on gold-standard data.

Second, based on the assumption that each claim amount $Y_{hi}|\mathbf{X}_h$ is log-normally distributed to accommodate non-negative support, right-skewed curve, and moderately heavy tail, etc., the development of a DPM model with log-skewnormal outcome (to account for aggregate claim amount $S_h|\mathbf{X}_h = \Sigma_{i=1}^{N_h} Y_{hi}|\mathbf{X}_h$) represents a novel mathematical treatment of a log-normal convolution problem. This is particularly noteworthy as it is the first research to explore the log-normal and log-skewnormal convolution within the Bayesian nonparametric framework. By addressing the mathematical development associated with log-normal and log-skewnormal convolutions, this thesis offers a theoretically sound foundation that enhances the reliability of our hybrid Bayesian risk premium modeling.

Third, in addition to the analytical derivation of the Gustafson's system of equations based on log-normal and log-skewnormal distributions, which is attempted for the first time, we significantly enhance the utility of the Gustafson correction

method. This expansion is achieved by introducing our novel prior knowledge that clarifies the relationship between $V(\mathbf{x}^*|\mathbf{z})$ and $V(\mathbf{x}^*|\mathbf{x})$, which has been a central challenge in the traditional Gustafson correction framework. We address this issue by introducing the scaling factor ζ, which quantifies the degree of this relationship, and investigate its performance in relation to varying levels of mismeasurement R_{ϵ_x} in the covariate. This approach is entirely original, adding a valuable dimension to the methodological framework for measurement error correction.

Fourth, this thesis integrates the DPM model with the Gustafson correction technique while enhancing the scalability of Bayesian methods for larger datasets. To our knowledge, this represents the first attempt at such an integration, improving the practicality of the Bayesian tools available to actuaries.

To make this thesis more accessible to actuaries, the proposed hybrid Bayesian risk premium model can be translated into practical, user-friendly tools. First, a dedicated software package in R or Python, accompanied by clear documentation, can be developed to facilitate easy implementation of the model in actuarial practice. Additionally, ongoing support through platforms like GitHub and active community engagement would ensure that actuaries can share feedback and continuously refine the model to meet their evolving needs. Collectively, we believe this thesis can serve as a solid foundation for future advancements in the domain of risk assessments and premium pricing.

7.2 Research Questions Revisited

In exploring the strategies to mitigate the model risks in risk premium modeling, this thesis has addressed the following key research questions, each designed to tackle critical aspects of the model risk arising from the inclusion of complete or incomplete covariates.

Revisiting **RQ1**, we examined model risk stemming from fully observed covariates, focusing on heterogeneity (RQ1.1), convolution error (RQ1.2), and scalability (RQ1.3). To manage heterogeneity (RQ1.1) within risk clusters, two Bayesian

techniques - partial pooling and parameter-free clustering - were explored. Partial pooling blended global and local model estimations, balancing shared and distinct cluster features, while parameter-free clustering dynamically generated brand-new clusters by simulating latent relationships between outcome and covariates.

When addressing convolution issues (RQ1.2) related to the aggregate claim amount $S_h|\mathbf{X}$ and the total aggregate claim amount $\tilde{S}|\mathbf{X}$, closed-form solutions were unfeasible; therefore, two approximation techniques were proposed. First, the Moment Matching method (log-normal convolution) approximated $S_h|\mathbf{X}$ with a log-skewnormal distribution, deriving key parameters by matching log-moment values via Monte Carlo integration. Second, Lindeberg's CLT approach (Chatterji 2007) applies at the portfolio level, approximating $\tilde{S}|\mathbf{X}$ under non-homogeneous conditions by relaxing classical CLT assumptions.

To handle scalability in Bayesian models with large datasets (RQ1.3), this thesis employed the anchor point and shard technique to run parallel MCMC simulations. Anchors - shared data points across shards - ensured consistency when merging shard outputs. Each shard operated as an independent subset, processing MCMC chains simultaneously, while anchors aligned cluster labels across shards. After simulations, a distance-based metric selectively merged similar clusters, preserving the model's latent structure without sacrificing detail. This method facilitated efficient scaling of Bayesian analysis to larger datasets.

Regarding **RQ2**, the incomplete covariate case, specifically Missing at Random (MAR) or Non-Differential Berkson (NDB) mismeasurement, this thesis examined the model risk from uncertainties arising from missing or mismeasured data within both parametric and nonparametric Bayesian frameworks. Handling MAR covariates (RQ2.1) involved data augmentation to address uncertainty in the imputation process. By alternating between missing data imputation and posterior updating steps, this data augmentation technique used a manageable joint and Gibbs sampling to capture cluster-level variations and reduce the variance in missing data imputations. This technique allowed for more accurate inference by continuously refining

the posterior distribution based on both observed data and missingness patterns.

For the NDB covariate (RQ2.2), which arose from non-differential, additive Berkson error, our hybrid Bayesian approach based on the Gustafson correction was also advantageous. By incorporating prior knowledge about the unobservable conditional covariate $\mathbf{x}^*|\mathbf{x}$, our hybrid model adjusted erroneous parameter estimates affected by latent factors and NDB covariates. This approach ultimately mitigates the model risks associated with NDB errors in the modeling process.

These strategies, when combined, support accurate and adaptable risk premium estimates under a wide range of conditions, effectively addressing model risks associated with both fully observed and incomplete covariates.

7.3 Limitations and Future Work

We hope that the contributions of this thesis will aid in addressing various covariate-based model risk challenges in insurance pricing analysis. However, there remain areas and directions that warrant further development:

- **Dimensionality:** To keep our analysis straightforward, we limited our model to two covariates (one binary $\mathbf{z}$ and one continuous $\mathbf{x}$). Expanding the covariate set could improve the precision and sensitivity in defining cluster memberships. However, it may also introduce additional noise, unobserved structures, or an increased computational workload. Therefore, further investigation into handling high-dimensional covariates within the Bayesian framework could provide more meaningful insights.

- **Assumptions on covariates:** This thesis primarily addresses scenarios involving missing data under the assumption of Missing at Random (MAR) for binary covariates $\mathbf{z}$ and Non-Differential Berkson (NDB) errors for continuous covariates $\mathbf{x}$. However, the landscape of assumptions regarding incomplete data is much broader, encompassing various mechanisms beyond MAR, as well as alternative types of measurement errors beyond NDB. This includes

cases such as Missing Not at Random (MNAR), Missing Completely at Random (MCAR), and Differential Classical (DC) measurement errors, among others, which were not considered in this thesis. Furthermore, this thesis has not explored scenarios where the assumptions differ, specifically examining cases where MAR applies to continuous covariates $\mathbf{x}$ and NDB applies to binary covariates $\mathbf{z}$. Addressing these variations would involve additional tweaks and adjustments, but they could enhance the applicability of our modeling approach and findings across diverse contexts.

- **Granularity of error rates:** We examined at what error level the Gustafson correction becomes ineffective. Based on our results, for error rates R_{ϵ_x} below 0.01, the correction is in general unnecessary as the model risk is minimal. Between 0.10 and 0.50, the correction proves useful, though we speculate that for error rates R_{ϵ_x} above 0.50, the correction may face significant limitations (e.g., when total sum of random noise magnitude is half of its total sum of the true values, the noise scale is substantial enough to severely distort the data, making the correction methods less effective). We have not yet explored this scenario primarily due to computational limitations. The complete evaluation of error rate from 0.10 to 0.99 and scaling factor values from 0.1 to 0.9 would require roughly a month per dataset (our Gibbs sampler required approximately 9 hours to complete 30,000 iterations of the hybrid MCMC for each combination of scaling factor and error rate level. We use 23 $\times$ Intel(R) Xeon(R) Gold 6230 CPU @ 2.10GHz). Given our three datasets (**LGPIF**, **Swautoins**, and **Brvehins2**) of varying sizes, the total computation time could extend to nearly half a year.

- **Covariate limitations:** The experiments on the **Brvehins2** dataset revealed a consistent underestimation of small claim values due to the covariates primarily affecting the upper distribution range. Identifying additional covariates that explain lower tail variations of the aggregate claim amount S_h in the **Brvehins2** dataset can be crucial. Moreover, throughout this thesis, we have

used a binary covariate $\mathbf{z}$ for simplicity to create a proxy variance: $V(\mathbf{x}^*|\mathbf{z})$. Given that the true covariate $\mathbf{x}$ is continuous, we may also explore using other continuous covariates to build a proxy variance for a more nuanced view of the variance structure.

- **Choice of threshold ε:** In large-scale implementations using parallel MCMC simulations, after sampling, we merge clustering results from each shard by counting anchor points. This merging depends on a threshold parameter $\boldsymbol{\varepsilon}$, which determines how clusters from different shards are combined. However, there is currently no research on the optimal choice of this threshold. Investigating its impact on clustering accuracy and efficiency could open new research directions.

Bibliography

Pearson, Karl (1936). "Method of moments and method of maximum likelihood". In: *Biometrika* 28.1/2, pp. 34–59.

Bühlmann, Hans (1969). "Experience rating and credibility". In: *ASTIN Bulletin: The Journal of the IAA* 5.2, pp. 157–165.

Nelder, John Ashworth and Robert WM Wedderburn (1972). "Generalized linear models". In: *Journal of the Royal Statistical Society Series A: Statistics in Society* 135.3, pp. 370–384.

Blackwell, David and James B MacQueen (1973). "Ferguson distributions via Pólya urn schemes". In: *The annals of statistics* 1.2, pp. 353–355.

Ferguson, Thomas S (1973). "A Bayesian analysis of some nonparametric problems". In: *The annals of statistics*, pp. 209–230.

Antoniak, Charles E (1974). "Mixtures of Dirichlet processes with applications to Bayesian nonparametric problems". In: *The annals of statistics*, pp. 1152–1174.

Sundberg, Rolf (1974). "Maximum likelihood theory for incomplete data from an exponential family". In: *Scandinavian Journal of Statistics*, pp. 49–58.

Rubin, Donald B (1976). "Inference and missing data". In: *Biometrika* 63.3, pp. 581–592.

Azzalini, Adelchi (1985). "A class of distributions which includes the normal ones". In: *Scandinavian journal of statistics*, pp. 171–178.

Escobedo, Miguel (1986). "On a characterisation of Dirac measures". In: *Proceedings of the Royal Society of Edinburgh Section A: Mathematics* 103.3-4, pp. 253–264.

Tanner, Martin (1987). "The calculation of posterior distributions by data augmentation". In: *Journal of the American statistical Association* 82.398, pp. 528–540.

Sharple, Linda (1990). "Identification and accommodation of outliers in general hierarchical models". In: *Biometrika* 77.3, pp. 445–453.

Brockman, Michael J and TS Wright (1992). "Statistical motor rating: making effective use of your data". In: *Journal of the Institute of Actuaries* 119.3, pp. 457–543.

Hooper, Peter M (1993). "Iterative weighted least squares estimation in heteroscedastic linear models". In: *Journal of the American Statistical Association* 88.421, pp. 179–184.

Richardson, Sylvia and Walter R Gilks (1993). "A Bayesian approach to measurement error problems in epidemiology using conditional independence models". In: *American Journal of Epidemiology* 138.6, pp. 430–442.

Cook, John R and Leonard A Stefanski (1994). "Simulation-extrapolation estimation in parametric measurement error models". In: *Journal of the American Statistical association* 89.428, pp. 1314–1328.

Diebolt, Jean and Christian P Robert (1994). "Estimation of finite mixture distributions through Bayesian sampling". In: *Journal of the Royal Statistical Society: Series B (Methodological)* 56.2, pp. 363–375.

Phillips, Richard D (1994). *Financial pricing of insurance in the multiple-line insurance company*. University of Pennsylvania.

Sethuraman, Jayaram (1994). "A constructive definition of Dirichlet priors". In: *Statistica sinica*, pp. 639–650.

Escobar, Michael and Mike West (1995). "Bayesian density estimation and inference using mixtures". In: *Journal of the american statistical association* 90.430, pp. 577–588.

Kass, Robert E and Larry Wasserman (1995). "A reference Bayesian test for nested hypotheses and its relationship to the Schwarz criterion". In: *Journal of the american statistical association* 90.431, pp. 928–934.

Fink, Daniel (1997). "A compendium of conjugate priors". In: *Environmental Statistics Group* 46.

Myers, Raymond H and Douglas C Montgomery (1997). "A tutorial on generalized linear models". In: *Journal of Quality Technology* 29.3, pp. 274–291.

Schafer, Joseph L (1997). *Analysis of incomplete multivariate data*. CRC press.

Brooks, Stephen P and Andrew Gelman (1998). "General methods for monitoring convergence of iterative simulations". In: *Journal of computational and graphical statistics* 7.4, pp. 434–455.

Hoeting, Jennifer A et al. (1999). "Bayesian model averaging: a tutorial". In: *Statistical science*, pp. 382–401.

Robert, Christian P and George Casella (1999). *Monte Carlo statistical methods*. Vol. 2. Springer.

Browne, Mark J. et al. (2000). "International Property-Liability Insurance Consumption". In: *The Journal of Risk and Insurance* 67.1, pp. 73–90. (Visited on 04/11/2023).

Celeux, Gilles et al. (2000). "Computational and inferential difficulties with mixture posterior distributions". In: *Journal of the American Statistical Association* 95.451, pp. 957–970.

Neal, Radford (2000). "Markov chain sampling methods for Dirichlet process mixture models". In: *Journal of computational and graphical statistics* 9.2, pp. 249–265.

Kennedy, Marc C and Anthony O'Hagan (2001). "Bayesian calibration of computer models". In: *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 63.3, pp. 425–464.

Makov, Udi E (2001). "Principal applications of Bayesian methods in actuarial science: a perspective". In: *North American Actuarial Journal* 5.4, pp. 53–57.

Scollnik, David PM (2001). "Actuarial modeling with MCMC and BUGS". In: *North American Actuarial Journal* 5.2, pp. 96–124.

Van Dyk, David (2001). "The art of data augmentation". In: *Journal of Computational and Graphical Statistics* 10.1, pp. 1–50.

Muthén, Bengt O (2002). "Beyond SEM: General latent variable modeling". In: *Behaviormetrika* 29.1, pp. 81–117.

Wood, GR (2002). "Assessing goodness of fit for Poisson and negative binomial models with low mean". In: *Communications in Statistics: Theory and Methods* 31, pp. 1977–2001.

Beaulieu, Norman C and Qiong Xie (2003). "Minimax approximation to lognormal sum distributions". In: *The 57th IEEE Semiannual Vehicular Technology Conference, 2003. VTC 2003-Spring.* Vol. 2. IEEE, pp. 1061–1065.

Dowd, Kevin (2003). *An introduction to market risk measurement.* John Wiley & Sons.

Francis, Louise (2003). "Martian chronicles: is MARS better than neural networks?" In: *Casualty Actuarial Society Forum*, pp. 75–102.

Jaynes, Edwin T (2003). *Probability theory: The logic of science.* Cambridge University Press.

Kofman, Paul and Ian G Sharpe (2003). "Using multiple imputation in the analysis of incomplete observations in finance". In: *Journal of Financial Econometrics* 1.2, pp. 216–249.

Anderson, D and K Burnham (2004). "Model selection and multi-model inference". In: *Second. NY: Springer-Verlag* 63.2020, p. 10.

Burkill, John Charles (2004). *The lebesgue integral.* 40. Cambridge University Press.

Denuit, Michel and Stefan Lang (2004). "Non-life rate-making with Bayesian GAMs". In: *Insurance: Mathematics and Economics* 35.3, pp. 627–647.

Freedman, Laurence S et al. (2004). "A new method for dealing with measurement error in explanatory variables of regression models". In: *Biometrics* 60.1, pp. 172–181.

Gelman, Andrew and Meng (2004). *Applied Bayesian modeling and causal inference from incomplete-data perspectives.* John Wiley & Sons.

Lang, Stefan and Andreas Brezger (2004). "Bayesian P-splines". In: *Journal of computational and graphical statistics* 13.1, pp. 183–212.

Royston, Patrick (2004). "Multiple imputation of missing values". In: *The Stata Journal* 4.3, pp. 227–241.

Wagenmakers, Eric-Jan and Simon Farrell (2004). "AIC model selection using Akaike weights". In: *Psychonomic bulletin & review* 11, pp. 192–196.

Jasra, A et al. (2005). "Markov Chain Monte Carlo Methods and the Label Switching Problem in Bayesian Mixture Modeling". In: *Statistical Science*, pp. 50–67.

Russell, Louise B (2005). "Comparing model structures in cost-effectiveness analysis". In: *Medical Decision Making* 25.5, pp. 485–486.

Wood, GR (2005). "Confidence and prediction intervals for generalised linear accident models". In: *Accident Analysis & Prevention* 37.2, pp. 267–273.

Anthony, Keith D (2006). "Introduction to causal modeling, bayesian theory and major bayesian modeling tools for the intelligence analyst". In: *USAF National Air and Space Intelligence Center (NASIC)*.

Baranoff, Etti et al. (2006). "Risk Management for Enterprises and Individuals". In: *Journal of organizational Management* 3.2, pp. 23–35.

Boland, Philip J (2006). "Statistical methods in general insurance". In.

Carroll, Raymond J et al. (2006). *Measurement error in nonlinear models: a modern perspective*. Chapman and Hall/CRC.

Chatterjee, Sourav (2006). "A generalization of the Lindeberg principle". In: *Annals of probability: An official journal of the Institute of Mathematical Statistics* 34.6, pp. 2061–2076.

Cole, Stephen R et al. (2006). "Multiple-imputation for measurement-error correction". In: *International journal of epidemiology* 35.4, pp. 1074–1081.

Dudley, Claire (2006). "Bayesian analysis of an aggregate claim model using various loss distributions". In: *Eidenburg: Disertasi Heriot-Watt University*.

Neuhaus, John M and Charles E McCulloch (2006). "Separating between-and within-cluster covariate effects by using conditional and partitioning methods". In:

Journal of the Royal Statistical Society: Series B (Statistical Methodology) 68.5, pp. 859–872.

White, Ian R (2006). "Commentary: Dealing with measurement error: multiple imputation or regression calibration?" In: *International Journal of Epidemiology* 35.4, pp. 1081–1082.

Bayes, Cristian Luis and Márcia D'Elia Branco (2007). "Bayesian inference for the skewness parameter of the scalar skew-normal distribution". In: *Brazilian Journal of Probability and Statistics*, pp. 141–163.

Chatterji, Srishti D (2007). "Lindeberg's central limit theorem à la Hausdorff". In: *Expositiones Mathematicae* 25.3, pp. 215–233.

Fewell, Zoe (2007). "Causal modelling in epidemiology and health services research". PhD thesis. University of Bristol.

Gelman, Andrew and Jennifer Hill (2007). *Data analysis using regression and multilevel/hierarchical models*. Cambridge University Press.

Lloyd-Smith, James O (2007). "Maximum likelihood estimation of the negative binomial dispersion parameter for highly overdispersed data, with applications to infectious diseases". In: *PloS one* 2.2, e180.

McLachlan, Geoffrey (2007). *The EM algorithm and extensions*. John Wiley & Sons.

Bousquet, Nicolas (2008). "Diagnostics of prior-data agreement in applied Bayesian analysis". In: *Journal of Applied Statistics* 35.9, pp. 1011–1029.

Brazauskas, Vytaras et al. (2008). "Estimating conditional tail expectation with actuarial applications in view". In: *Journal of Statistical Planning and Inference* 138.11, pp. 3590–3604.

Droguett, Enrique López and Ali Mosleh (2008). "Bayesian methodology for model uncertainty using model performance data". In: *Risk Analysis: An International Journal* 28.5, pp. 1457–1476.

Gustafson, Paul (2008). *Measurement error and misclassification in statistics and epidemiology: impacts and Bayesian adjustments*. CRC Press.

Kaas, Rob et al. (2008). *Modern actuarial risk theory: using R*. Vol. 128. Springer Science & Business Media.

Li, Xue (2008). "A Novel Accurate Approximation Method of Lognormal Sum Random Variables". PhD thesis. Wright State University.

Rocquigny, Etienne de and Nicolas Devictor (2008). *Uncertainty in industrial practice: a guide to quantitative uncertainty management*. John Wiley & Sons.

Romann, Alexandra (2008). "Evaluating the performance of simulation extrapolation and Bayesian adjustments for measurement error". PhD thesis. University of British Columbia.

Winkelmann, Rainer (2008). *Econometric analysis of count data*. Springer Science & Business Media.

Apanasovich, Tatiyana V et al. (2009). "SIMEX and standard error estimation in semiparametric measurement error models". In: *Electronic journal of statistics* 3, p. 318.

Baranoff, Etti (2009). *Risk management for enterprises and individuals*. URL: `https://saylordotorg.github.io/text_risk-management-for-enterprises-and-individuals/index.html`. (accessed: 08.13.2023).

Frees, Edward (2009). *Regression modeling with actuarial and financial applications*. Cambridge University Press.

Graham, John W (2009). "Missing data analysis: Making it work in the real world". In: *Annual review of psychology* 60, pp. 549–576.

Hastie, Trevor et al. (2009). *The elements of statistical learning: data mining, inference, and prediction*. Vol. 2. Springer.

Hogg, Robert V and Stuart A Klugman (2009). *Loss distributions*. John Wiley & Sons.

Jackson and Thompson (2009). "Accounting for uncertainty in health economic decision models by using model averaging". In: *Journal of the Royal Statistical Society: Series A (Statistics in Society)* 172.2, pp. 383–404.

Shahbaba, Babak and Radford Neal (2009). "Nonlinear models using Dirichlet process mixtures." In: *Journal of Machine Learning Research* 10.8.

Wu (2009). *Mixed effects models for complex data*. CRC press.

Bartlett, Jonathan William (2010). "Correction for Classical Covariate Measurement Error and Extensions Fo Life-course Studies". PhD thesis. London School of Hygiene and Tropical Medicine (University of London).

Hutcheon, Jennifer A et al. (2010). "Random measurement error and regression dilution bias". In: *Bmj* 340.

Jackson and Sharples (2010). "Structural and parameter uncertainty in Bayesian cost-effectiveness models". In: *Journal of the Royal Statistical Society: Series C (Applied Statistics)* 59.2, pp. 233–253.

Myors, Brett and Kevin Murphy (2010). *Statistical power analysis: A simple and general model for traditional and modern hypothesis tests*. Routledge.

Ohlsson, Esbjörn and Björn Johansson (2010). *Non-life insurance pricing with generalized linear models*. Vol. 174. Springer.

Parker, Robin (2010). *Missing data problems in machine learning*. VDM Verlag.

Pollino, CA and C Henderson (2010). "Bayesian networks: A guide for their application in natural resource management and policy". In: *Landscape Logic, Technical Report* 14.

Swamy, PAVB et al. (2010). "Estimation of parameters in the presence of model misspecification and measurement error". In: *Studies in Nonlinear Dynamics & Econometrics* 14.3.

Tanner, Martin (2010). "From EM to data augmentation: the emergence of MCMC Bayesian computation in the 1980s". In: *Statistical science* 25.4, pp. 506–516.

Teh, Yee Whye (2010). "Dirichlet Process." In.

Teh, Yee Whye and Michael I Jordan (2010). "Hierarchical Bayesian nonparametric models with applications". In: *Bayesian nonparametrics* 1, pp. 158–207.

Werner, Geoff and Claudine Modlin (2010). "Basic ratemaking". In: *Casualty Actuarial Society*. Vol. 4, pp. 1–320.

Zhang, Jie (2010). "Statistical technique to address errors in measurement". PhD thesis. The University of Utah.

Bilcke, Joke et al. (2011). "Accounting for methodological, structural, and parameter uncertainty in decision-analytic models: a practical guide". In: *Medical Decision Making* 31.4, pp. 675–692.

Blei, David M and Peter I Frazier (2011). "Distance dependent Chinese restaurant processes." In: *Journal of Machine Learning Research* 12.8.

Briscoe, Erica and Jacob Feldman (2011). "Conceptual complexity and the bias/variance tradeoff". In: *Cognition* 118.1, pp. 2–16.

Cairns, Andrew JG et al. (2011). "Bayesian stochastic mortality modelling for two populations". In: *ASTIN Bulletin: The Journal of the IAA* 41.1, pp. 29–59.

Cowling, CA et al. (2011). "Developing a framework for the use of discount rates in actuarial work A discussion paper". In: *IFoA journal, London* 31.

Bouguila, Nizar and Djemel Ziou (2012). "A countably infinite mixture model for clustering and feature selection". In: *Knowledge and information systems* 33, pp. 351–370.

Eling, Martin (2012). "Fitting insurance claims to skewed distributions: Are the skew-normal and skew-student good models?" In: *Insurance: Mathematics and Economics* 51.2, pp. 239–248.

Fraser, Gary E and Daniel O Stram (2012). "Regression calibration when foods (measured with error) are the variables of interest: markedly non-Gaussian data with many zeroes". In: *American journal of epidemiology* 175.4, pp. 325–331.

Gershman, Samuel J and David M Blei (2012). "A tutorial on Bayesian nonparametric models". In: *Journal of Mathematical Psychology* 56.1, pp. 1–12.

Gopnik, Alison and Henry M Wellman (2012). "Reconstructing constructivism: Causal models, Bayesian learning mechanisms, and the theory theory." In: *Psychological bulletin* 138.6, p. 1085.

Gupta, Sandeep K (2012). "Use of Bayesian statistics in drug development: advantages and challenges". In: *International Journal of Applied and Basic Medical Research* 2.1, p. 3.

Jebara, Tony (2012). *Machine learning: discriminative and generative*. Vol. 755. Springer Science & Business Media.

Ng and Krishnan (2012). "The EM algorithm". In: *Handbook of computational statistics*. Springer, pp. 139–172.

Skrondal, Anders and Jouni Kuha (2012). "Improved regression calibration". In: *Psychometrika* 77.4, pp. 649–669.

Azzalini, Adelchi (2013). *The skew-normal and related families*. Vol. 3. Cambridge University Press.

Cohn, Donald L (2013). *Measure theory*. Vol. 5. Springer.

Gelman, Andrew and John Carlin (2013). *Bayesian data analysis*. CRC press.

Agogo, George O et al. (2014). "Use of two-part regression calibration model to correct for measurement error in episodically consumed foods in a single-replicate study design: EPIC case study". In: *PLoS One* 9.11, e113160.

Ahmad, Izhar (2014). "K-Mean and K-Prototype Algorithms Performance Analysis". In: *International Journal of Computer and Information Technology* 3.04, pp. 823–828.

Bhaskaran, Krishnan and Liam Smeeth (2014). "What is the difference between missing completely at random and missing at random?" In: *International journal of epidemiology* 43.4, pp. 1336–1339.

Brockett, Patrick et al. (2014). "Generalized additive models and nonparametric regression". In: *Predictive modeling applications in actuarial science. Predict Model Tech* 1, p. 367.

Charpentier, Arthur (2014). *Computational actuarial science with R*. CRC press.

Derrig, Richard and Glenn Meyers (2014). *Predictive modeling applications in actuarial science*. Vol. 1. Cambridge University Press.

Gelman, Andrew and Jessica Hwang (2014). "Understanding predictive information criteria for Bayesian models". In: *Statistics and computing* 24.6, pp. 997–1016.

Klein, Nadja et al. (2014). "Nonlife ratemaking and risk management with Bayesian generalized additive models for location, scale, and shape". In: *Insurance: Mathematics and Economics* 55, pp. 225–249.

Shah, Anoop D et al. (2014). "Comparison of random forest and parametric imputation models for imputing missing data using MICE: a CALIBER study". In: *American journal of epidemiology* 179.6, pp. 764–774.

Shi, Peng and Emiliano A Valdez (2014). "Multivariate negative binomial models for insurance claim counts". In: *Insurance: Mathematics and Economics* 55, pp. 18–29.

An, Jinwon and Sungzoon Cho (2015). "Variational autoencoder based anomaly detection using reconstruction probability". In: *Special Lecture on IE* 2.1, pp. 1–18.

Cousineau, Denis and Teresa Allan (2015). "Likelihood and its use in parameter estimation and model comparison". In: *Mesure et évaluation en éducation* 37.3, pp. 63–98.

Fellingham, Gilbert W et al. (2015). "Bayesian nonparametric predictive modeling of group health claims". In: *Insurance: Mathematics and Economics* 60, pp. 1–10.

Fu, Shuai (2015). "A hierarchical Bayesian approach to negative binomial regression". In: *Methods and Applications of Analysis* 22.4, pp. 409–428.

Kaivanipour, Kivan (2015). *Non-life Insurance Pricing using the Generalized Additive Model, Smoothing Splines and L-Curves*.

Korn, Uri (2015). "A Frequency-Severity Stochastic Approach to Loss Development". In: *Casualty Actuarial Society E-Forum, Spring*, pp. 1–28.

Letham, Benjamin et al. (2015). "Interpretable classifiers using rules and bayesian analysis: Building a better stroke prediction model". In: *The Annals of Applied Statistics* 9.3, pp. 1350–1371.

McNeil, Alexander J et al. (2015). *Quantitative risk management: concepts, techniques and tools-revised edition*. Princeton University Press.

Seri, Raffaello and Christine Choirat (2015). "Comparison of approximations for compound Poisson processes". In: *ASTIN Bulletin: The Journal of the IAA* 45.3, pp. 601–637.

Zyphur, Michael J and Frederick L Oswald (2015). "Bayesian estimation and inference: A user's guide". In: *Journal of Management* 41.2, pp. 390–420.

Aggarwal, Ankur et al. (2016). "Model risk–daring to open up the black box". In: *British Actuarial Journal* 21.2, pp. 229–296.

Mara, Thierry A et al. (2016). "A comparison of two Bayesian approaches for uncertainty quantification". In: *Environmental Modelling & Software* 82, pp. 21–30.

Guo and Riebler (2017). "Bayesian bivariate meta-analysis of diagnostic test studies with interpretable priors". In: *Statistics in medicine* 36.19, pp. 3039–3058.

Hastie, Trevor (2017). "Generalized additive models". In: *Statistical models in S*. Routledge, pp. 249–307.

Hoffmann, Sabine et al. (2017). "Accounting for Berkson and classical measurement error in radon exposure using a Bayesian structural approach in the analysis of lung cancer mortality in the French cohort of uranium miners". In: *Radiation Research* 187.2, pp. 196–209.

Hong, Liang and Ryan Martin (2017). "A flexible Bayesian nonparametric model for predicting future insurance claims". In: *North American Actuarial Journal* 21.2, pp. 228–241.

Black, Rob et al. (2018). "Model risk: illuminating the black box". In: *British Actuarial Journal* 23.

Hong, Liang and Ryan Martin (2018). "Dirichlet process mixture models for insurance loss data". In: *Scandinavian Actuarial Journal* 2018.6, pp. 545–554.

Lage, Isaac et al. (2018). "Human-in-the-loop interpretability prior". In: *Advances in neural information processing systems*, pp. 10159–10168.

Ma, Zhihua and Guanghui Chen (2018). "Bayesian methods for dealing with missing data problems". In: *Journal of the Korean Statistical Society* 47, pp. 297–313.

McElreath, Richard (2018). *Statistical rethinking: A Bayesian course with examples in R and Stan*. Chapman and Hall/CRC.

Oh, Eric J et al. (2018). "Considerations for analysis of time-to-event outcomes measured with error: Bias and correction with SIMEX". In: *Statistics in medicine* 37.8, pp. 1276–1289.

Quan, Zhiyu and Emiliano A Valdez (2018). "Predictive analytics of insurance claims using multivariate decision trees". In: *Dependence Modeling* 6.1, pp. 377–407.

Roy, Jason et al. (2018). "Bayesian nonparametric generative models for causal inference with missing at random covariates". In: *Biometrics* 74.4, pp. 1193–1202.

Spedicato, Giorgio Alfredo et al. (2018). "Machine learning methods to perform pricing optimization. A comparison with standard GLMs". In: *Variance* 12.1, pp. 69–89.

Tomczak, Jakub and Max Welling (2018). "VAE with a VampPrior". In: *International conference on artificial intelligence and statistics*. PMLR, pp. 1214–1223.

McLachlan, Geoffrey J et al. (2019). "Finite mixture models". In: *Annual review of statistics and its application* 6, pp. 355–378.

Rocher, Luc and Julien Hendrickx (2019). "Estimating the success of re-identifications in incomplete datasets using generative models". In: *Nature communications* 10.1, pp. 1–9.

Šoltés, Erik et al. (2019). "General linear model: an effective tool for analysis of claim severity in motor third party liability insurance". In: *Statistics* 13.

Strežo, Marek (2019). "Flexible regression modeling of the pure premium using generalized additive model with isotropic smoothing". In: *Ecomomic Review/Ekonomické Rozhl'ady* 48.4.

Strežo, Marek et al. (2019). "Risk premium prediction of motor hull insurance using generalized linear models." In: *Statistika: Statistics & Economy Journal* 99.4.

Wang, Kaiyuan et al. (2019). "A novel moment method using the log skew normal distribution for particle coagulation". In: *Journal of Aerosol Science* 134, pp. 95–108.

Asmussen, Søren and Mogens Steffensen (2020). *Risk and Insurance: A Graduate Text*. Vol. 96. Springer Nature.

Bhattacharyya, Atreyee (2020). "AI and Automation working party - Short term output, January 2020". In: *IFoA Journal, London*, p. 4.

Huang, Yifan and Shengwang Meng (2020). "A Bayesian nonparametric model and its application in insurance loss prediction". In: *Insurance: Mathematics and Economics* 93, pp. 84–94.

Kunkel, Deborah and Mario Peruggia (2020). "Anchored Bayesian Gaussian mixture models". In: *Electronic Journal of Statistics* 14, pp. 3869–3913.

Ni, Yang et al. (2020). "Consensus Monte Carlo for random subsets using shared anchors". In: *Journal of Computational and Graphical Statistics* 29.4, pp. 703–714.

Ungolo, Francesco and Torsten Kleinow (2020). "A hierarchical model for the joint mortality analysis of pension scheme data with missing covariates". In: *Insurance: Mathematics and Economics* 91, pp. 68–84.

Wuthrich, Mario V (2020). "Non-life insurance: mathematics & statistics". In: *Available at SSRN 2319328*.

Grace, Y Yi et al. (2021). *Handbook of measurement error models*. CRC Press.

Klau, Simon et al. (2021). "Examining the robustness of observational associations to model, measurement and sampling uncertainty with the vibration of effects framework". In: *International Journal of Epidemiology* 50.1, pp. 266–278.

Nab, Linda et al. (2021). "Mecor: An R package for measurement error correction in linear regression models with a continuous outcome". In: *Computer methods and programs in biomedicine* 208, p. 106238.

Nuti, Giuseppe et al. (2021). "An explainable Bayesian decision tree algorithm". In: *Frontiers in Applied Mathematics and Statistics* 7, p. 598833.

Sinha, Samiran (2021). "Bayesian approaches for handling covariate measurement error". In: *Handbook of Measurement Error Models*. Chapman and Hall/CRC, pp. 527–554.

Stamey, James D and John W Seaman (2021). "Bayesian adjustment for misclassification". In: *Handbook of Measurement Error Models*. Chapman and Hall/CRC, pp. 507–526.

Clare, Mariana CA et al. (2022). "Explainable artificial intelligence for Bayesian neural networks: Toward trustworthy predictions of ocean dynamics". In: *Journal of Advances in Modeling Earth Systems* 14.11, e2022MS003162.

Shams, Mostafa (2022). "Bayesian Nonparametric Regression Models for Insurance Claims Frequency and Severity". PhD thesis. University of Nevada.

Wahl, Jens Christian et al. (2022). "Spatial modelling of risk premiums for water damage insurance". In: *Scandinavian Actuarial Journal* 2022.3, pp. 216–233.

Jamotton, Charlotte and Donatien Hainaut (2023). *Variational autoencoder for synthetic insurance data*. Tech. rep. Université catholique de Louvain, Institute of Statistics and Biostatistics.

Kuo, Kevin and Daniel Lupton (2023). "Towards Explainability of Machine Learning Models in Insurance Pricing". In: *Variance* 16.1.

Noroozi, Ghazaleh (2023). "Data Heterogeneity and Its Implications for Fairness". PhD thesis. The University of Western Ontario (Canada).

Parodi, Pietro (2023). *Pricing in general insurance*. Chapman and Hall/CRC.

Martin, Gael M et al. (2024). "Approximating Bayes in the 21st century". In: *Statistical Science* 39.1, pp. 20–45.

Ungolo, Francesco and Edwin R van den Heuvel (2024). "A Dirichlet Process Mixture regression model for the analysis of competing risk events". In: *Insurance: Mathematics and Economics*.

Appendix A

A.1 Variable Definition

The following variables and functions are used in this manuscript:

$i = 1, \ldots, N_h$	observation index i in policy h.
$h = 1, \ldots, H$	policy index h with sample (policy) size H.
$j = 1, \ldots, J$	cluster index for J clusters.
s_h	cluster index $j = 1, \ldots, J$ for a policy h.
n_j	number of observations in cluster j, and n_j^{-h} is that in cluster j from where observation h removed.
Y_{hi}, N_h	ith individual loss amount and loss count in a policy h.
$Y_{j(hi)}, N_{j(h)}$	ith individual loss amount and loss count in a policy h in a cluster j.
$\underline{Y}_h : \{Y_{h1}, \cdots, Y_{hN_h}\}$	multiple loss amounts in a policy h.
S_h	outcome variable as $\Sigma_i Y_{hi}$ in a policy h.
$\tilde{S}$	outcome variable as $\Sigma_h S_h$ across entire policies.
$\mathbf{X} : \{\mathbf{X}^F,\ \mathbf{X}^S\}$	list of covariate matrices (including $\mathbf{X}^S, \mathbf{X}^S$) for both frequency and severity.
$\mathbf{X}^F : \{\mathbf{x}^F, \mathbf{z}^F\}$	matrix of covariates (including $\mathbf{x}^F, \mathbf{z}^F$) for claim count outcome (Frequency).
$\mathbf{X}^S : \{\mathbf{x}^S, \mathbf{z}^S\}$	matrix of covariates (including $\mathbf{x}^S, \mathbf{z}^S$) for claim amount outcome (Severity).

$\mathbf{X} : \{\mathbf{x}, \mathbf{z}\}$	matrix of covariates (including $\mathbf{z}^S, \mathbf{x}^S$) for claim amount outcome (Severity). Since CH5 and CH6 focus solely on severity, we omit the superscript "s" for simplicity in these chapters.
$\mathbf{X}_h^F : \{x_h^F, z_h^F\}$	vector of covariates for observation h (Frequency).
$\mathbf{x}^F$	vector of continuous covariate (Frequency).
$\mathbf{z}^F$	vector of binary covariate (Frequency).
x_h^F	individual value of continuous covariate (Frequency).
z_h^F	individual value of binary covariate (Frequency).
$\mathbf{X}_h^S : \{x_h^S, z_h^S\}$	vector of covariates for observation h (Severity).
$\mathbf{x}^S$	vector of continuous covariate, and $\mathbf{x}^*$ indicates the mismeasured (Severity).
$\mathbf{z}^S$	vector of binary covariate (Severity).
x_h^S	individual value of continuous covariate, and x_h^* indicates the mismeasured (Severity).
z_h^S	individual value of binary covariate (Severity).
$\mathbf{X}_h : \{x_h, z_h\}$	vector of covariates for observation h.
$\mathbf{x}$	vector of continuous covariate, and $\mathbf{x}^*$ indicates the mismeasured.
$\mathbf{z}$	vector of binary covariate.
x_h	individual value of continuous covariate, and x_h^* indicates the mismeasured.
z_h	individual value of binary covariate.
$Y_{N(t-dt)}^{\varnothing}$	unreported outcome data that appears before the end of a given policy period t in CH3.
$p_0(\cdot)$	parameter model (for prior).
$p(\cdot)$	parameter model (for posterior).
$f_0(\cdot)$	data model (for continuous cluster).
$f(\cdot)$	data model (for discrete cluster).
$E[\,\cdot\,],\ V[\,\cdot\,]$	Expectation and variance as point estimates.

$\phi(\cdot)$	probability density function of Standard Gaussian density.
$\Phi(\cdot)$	Cumulative density function of Standard Gaussian density.
$\boldsymbol{\theta}_j$	set of parameters - $\boldsymbol{\beta}, \sigma^2, \xi$ - associated with the outcome model $f(\Sigma Y \vert \mathbf{X})$ for j cluster (posterior sample: $\boldsymbol{\theta}_j^{(*)}$).
$\boldsymbol{w}_j$	set of parameters - π, μ, λ^2 - associated with the covariate models $f(\mathbf{X})$ for j cluster (posterior sample: $\boldsymbol{w}_j^{(*)}$).
$\boldsymbol{\omega}_j$	cluster weights (mixing coefficient) for j cluster (finalized sample: $\boldsymbol{\omega}_j^{(*)}$).
$\boldsymbol{\phi} : \{\boldsymbol{\theta}, \mathbf{w}, \boldsymbol{\omega}\}$	a set of parameter vectors of the joint likelihood.
$\boldsymbol{\beta}_j : \{\beta_{j0}, \beta_{j1}, \beta_{j2}\}$	regression coefficient vector for a mean outcome estimation.
$\boldsymbol{\beta}_0, \Sigma_{\beta_0}$	vector of initial regression coefficients and variance-covariance matrix, i.e. $\widehat{\sigma}^2(\mathbf{X}^T\mathbf{X})^{-1} = \mathbf{X}^T\mathbf{X}(\Sigma Y - \Sigma\widehat{Y})^T(\Sigma Y - \Sigma\widehat{Y})/(n-p)$ obtained from the baseline multivariate Gamma regression of $\Sigma\widehat{Y} > 0$.
σ_j^2	cluster-wise variance or scale parameter for the outcome.
ξ_j	skewness parameter for log skew-normal outcome.
π_j	proportion parameter for Bernoulli covariate.
μ_j	location parameter for Gaussian covariate $\mathbf{x}$.
λ_j^2	dispersion parameter for Gaussian covariate $\mathbf{x}$.
$\boldsymbol{\kappa}_j : \{\kappa_{j0}, \kappa_{j1}\}$	regression coefficient vectors to explain the mean of the unobserved Gaussian covariate $\mathbf{x}\vert\mathbf{z}$ in CH4, CH6.
τ_j^2	variance parameter for $\mathbf{x}^*\vert\mathbf{x}$ to indicate the contamination level in the measurement model in CH4, CH6.
α	precision parameter that controls the variance of the clustering simulation in CH5, CH6.
G_0	joint prior for all parameters in the DPM ($\boldsymbol{\beta}, \sigma^2, \xi, \pi, \lambda^2, \boldsymbol{\kappa}$, and α) in CH5, CH6. It allows all continuous, integrable distributions to be supported while retaining theoretical properties such as asymptotic consistency, etc.

m_0, δ	hyperparameters of Multivariate Normal for $\boldsymbol{\beta}_0^S$ in CH4.
q_0, Λ	hyperparameters of Inverse Wishart density for $\Sigma_{\beta_0}^S$ in CH4.
u_0, v_0	hyperparameters of Inverse Gamma density for σ_j^2.
ρ_{u1}, ρ_{u2}	hyperparameters of Fink's function for u_0.
ρ_{v1}, ρ_{v2}	hyperparameters of Gamma density for v_0.
μ_0, λ_j^2	hyperparameters of Gaussian density of μ_j in CH5.
c_0, d_0	hyperparameters of Inverse Gamma density for λ_j^2
ν_0	hyperparameters of Student's t density for ξ_j in CH5, CH6.
g_0, h_0	hyperparameters of Beta density for π_j.
γ_0, ψ_0	hyperparameters of Gamma density for α in CH5, CH6.
η	random probability value ($0 < \eta < 1$) of Gamma mixture density for the posterior on α in CH5, CH6.
π_η	mixing coefficient of Gamma mixture density for the posterior of the precision parameter α in CH5, CH6.
$\mathbb{K}_1 : \begin{pmatrix} 1 & z_1 \\ \vdots & \vdots \\ 1 & z_{n_j} \end{pmatrix}$	$n_j \times 2$ matrix to compute $\Sigma_{h=1}^{n_j}(x_h^* - \widehat{\kappa}_{j0} - \widehat{\kappa}_{j1} z_h)^2$ in CH4, CH6.
$\mathbb{K}_2 : \begin{pmatrix} \sum_{h=1}^{n_j} x_h^* \\ \\ \sum_{h=1}^{n_j} x_h^* z_h \end{pmatrix}$	2×1 matrix to compute $\Sigma_{h=1}^{n_j}(x_h^* - \widehat{\kappa}_{j0} - \widehat{\kappa}_{j1} z_h)^2$ in CH4, CH6.
$\boldsymbol{\theta}_{kj}$	cluster parameter vector in cluster j within shard k.
$\boldsymbol{F}_{kj}$	indices of the observations in cluster j within shard k.
$\boldsymbol{w}_j^z$	set of parameters - π, μ, λ^2 - associated with the covariate models $f(\mathbf{X})$ for j cluster where missing values z_i in covariate $\mathbf{z}$ belong to.
$\boldsymbol{\beta}_j^F : \{\beta_{j0}^F, \beta_{j1}^F, \beta_{j2}^F\}$	regression coefficient vector for a mean claim count (Frequency) estimation in CH4.
$\boldsymbol{\beta}_j^S : \{\beta_{j0}^S, \beta_{j1}^S, \beta_{j2}^S\}$	regression coefficient vector for a mean claim amount (Severity) estimation in CH4.

ξ_h, ψ	parameters - ξ_h (number of failure) and ψ (number of success) - for Negative Binomial in CH4.
$\boldsymbol{\beta}_0^F, \Sigma_{\beta_0}^F$	vector of initial regression coefficients and variance-covariance matrix obtained from the baseline multivariate Poisson regression of $\widehat{N} > 0$ in CH4.
$\boldsymbol{\beta}_0^S, \Sigma_{\beta_0}^S$	vector of initial regression coefficients and variance-covariance matrix obtained from the baseline multivariate Gamma regression of $\widehat{Y} > 0$ in CH4.
u_0^F, v_0^F	hyperparameters of Inverse Gamma density for ψ_j in CH4.
u_0^S, v_0^S	hyperparameters of Inverse Gamma density for σ_j^2 in CH4.
$\underline{m}_0, \underline{\delta}$	hyperparameters of Multivariate Normal for $\boldsymbol{\beta}_0^F$ in CH4.
$\underline{q}_0, \underline{\Lambda}$	hyperparameters of Inverse Wishart density for $\Sigma_{\beta_0}^F$ in CH4.
$\underline{\rho}_{u1}, \underline{\rho}_{u2}$	hyperparameters of Fink's function for u_0^F in CH4.
$\underline{\rho}_{v1}, \underline{\rho}_{v2}$	hyperparameters of Gamma density for v_0^F in CH4.
$\boldsymbol{\beta}_0^{S+}, \Sigma_{\beta_0}^{S+}, u_0^{S+}, v_0^{S+}$	communal hyperparameters for partial pooling in a hierarchical GLM (claim amount) in CH4.
$\boldsymbol{\beta}_0^{F+}, \Sigma_{\beta_0}^{F+}, u_0^{F+}, v_0^{F+}$	communal hyperparameters for partial pooling in a hierarchical GLM (claim count) in CH4.
$\widehat{\boldsymbol{\beta}}_j : \{\widehat{\beta}_{j0}, \widehat{\beta}_{j1}, \widehat{\beta}_{j2}\}$	regression coefficient vector for a mean outcome estimation based on NDB covariate $\mathbf{x}^*$ (before correction) in CH4, CH6.
$\widehat{\sigma}_j^2$	cluster-wise variance or scale parameter for the outcome based on NDB covariate $\mathbf{x}^*$ (before correction) in CH4, CH6.
$\boldsymbol{\theta}^{(old)} : \{\boldsymbol{\beta}_j^{S(old)}, \sigma_j^{2(old)}, \boldsymbol{\beta}_j^{F(old)}, \psi_j^{(old)}\}$	initial value of outcome parameters for the MH algorithm in CH4.
$\boldsymbol{\theta}^{(new)} : \{\boldsymbol{\beta}_j^{S(new)}, \sigma_j^{2(new)}, \boldsymbol{\beta}_j^{F(new)}, \psi_j^{(new)}\}$	candidate value of outcome parameters sampled for the MH algorithm in CH4.
$\boldsymbol{\theta}^{(*)} : \{\boldsymbol{\beta}_j^{S(*)}, \sigma_j^{2(*)}, \boldsymbol{\beta}_j^{F(*)}, \psi_j^{(*)}\}$	finalized value of outcome parameters for the MH algorithm in CH4.

$\widehat{\xi}_j$	skewness parameter for log skew-normal outcome based on NDB covariate $\mathbf{x}^*$ (before correction) in CH4, CH6.
$\widehat{\lambda}_j^2$	dispersion parameter for Gaussian covariate $\mathbf{x}\|\mathbf{z}$ based on NDB covariate $\mathbf{x}^*$ (before correction) in CH4, CH6..
$\widehat{\boldsymbol{\kappa}}_j : \{\widehat{\kappa}_{j0}, \widehat{\kappa}_{j1}\}$	regression coefficient vectors to explain the mean of the unobserved Gaussian covariate $\mathbf{x}\|\mathbf{z}$ based on NDB covariate $\mathbf{x}^*$ (before correction) in CH4, CH6.
$\tilde{\boldsymbol{\kappa}}, \tilde{\Sigma}_\kappa$	hyperparameters of Multivariate Normal for $\boldsymbol{\kappa}$ in CH4, CH6.
R_{ϵ_x}	error rate, representing the proportion of the total noise relative to the total true values within an NDB covariate.
$\sigma_{j\epsilon}^2$	variance of an NDB error.
$\rho_{j(x,x^*)}$	correlation between the true covariate and the NDB covariate in cluster j.
$\boldsymbol{\epsilon}$	NDB measurement error
$\boldsymbol{\varepsilon}$	threshold or level of tolerance as a tuning parameter: $0 < \boldsymbol{\varepsilon} < 1$ used to determine the merging of different clusters in CH3, CH6.
ζ	scaling factor that indicates a proportion of the explained variance reduction in the relationship: $V(\mathbf{x}^*\|\mathbf{x}) = (1-\zeta) \times V(\mathbf{x}^*\|\mathbf{z})$.
$\ddot{S}_h$	outcome value selected as anchor point in CH3, CH6.
$\ddot{\mathbf{X}}_h : \{\ddot{x}_h, \ddot{z}_h\}$	vector of covariate values selected as anchor point in CH3, CH6.

A.2 Proof of Lindeberg's Convergence Condition

Let $\{S_1|\mathbf{X}_1, \cdots, S_H|\mathbf{X}_H\}$ be a sequence of conditional log-skewnormal random variables defined on a probability triple $(\Omega, \boldsymbol{F}, \mathbb{P})$ such that $S_1|\mathbf{X}_1, \cdots S_H|\mathbf{X}_H$ are independent, but not identically distributed due to $\mathbf{X}_h$. Assuming that this sequence has the finite mean $E[S_h|\mathbf{X}_h]$ and variance $V(S_h|\mathbf{X}_h)$, and the sum of the variance is denoted as $\mathbb{S}_H^2 = \Sigma_{h=1}^H V(S_h|\mathbf{X}_h)$, then, for every $\epsilon > 0$, Lindeberg's condition given

by

$$\lim_{H\to\infty}\frac{1}{\mathbb{S}_H^2}\sum_{h=1}^{H}\boldsymbol{E}\Big[\big(S_h|\mathbf{X}_h - E[S_h|\mathbf{X}_h]\big)^2\cdot\mathbb{1}_{\Big|S_h|\mathbf{X}_h - E[S_h|\mathbf{X}_h]\Big| > \epsilon\cdot\mathbb{S}_H}\Big] = 0$$

should be satisfied.

Prior to examining Lindeberg's condition, we derive the expressions of the first two moments of the log-skewnormal distribution based on the form of the moment generating function given by Azzalini 2013

$$E[S^p|\mathbf{X}] = \exp\Big(p\mathbf{X}^T\boldsymbol{\beta} + \frac{1}{2}p^2\sigma^2\Big)\Big[1 + \mathrm{erf}\big(\frac{p\sigma\delta}{\sqrt{2}}\big)\Big], \quad \delta = \frac{\xi}{\sqrt{1+\xi^2}}$$

where 'erf$(\cdot)$' denotes the Gaussian *error function* [1] defined as

$$\mathrm{erf}(r) = \int_0^r \frac{2}{\sqrt{\pi}}e^{-t^2}dt$$

This can be set equal to the standard normal density with $t = \frac{1}{\sqrt{2}}u,\ dt = \frac{1}{\sqrt{2}}du$

$$\begin{aligned}
&\int_0^r \frac{2}{\sqrt{\pi}}e^{-t^2}dt \\
&= \int_0^{\sqrt{2}r}\frac{2}{\sqrt{\pi}}e^{-u^2/2}\frac{1}{\sqrt{2}}du = 2\int_0^{\sqrt{2}r}\frac{1}{\sqrt{2\pi}}e^{-u^2/2}du = 2\left[\int_{-\infty}^{\sqrt{2}r}\frac{1}{\sqrt{2\pi}}e^{-u^2/2}du - \int_{-\infty}^{0}\frac{1}{\sqrt{2\pi}}e^{-u^2/2}du\right] \\
&= 2\Big[\Phi(\sqrt{2}r) - 0.5\Big] = 2\Phi(\sqrt{2}r) - 1
\end{aligned}$$

For the 1st moment ($p = 1$), this gives

$$\begin{aligned}
E[S^{p=1}|\mathbf{X}] = \ &\exp\Big(\mathbf{X}^T\boldsymbol{\beta} + \frac{1}{2}\sigma^2\Big)\Big[1 + \mathrm{erf}\big(\frac{\sigma\delta}{\sqrt{2}}\big)\Big] \\
&= \exp\Big(\mathbf{X}^T\boldsymbol{\beta} + \frac{1}{2}\sigma^2\Big)\Big[1 + 2\Phi\Big(\frac{\sigma\xi}{\sqrt{1+\xi^2}}\Big) - 1\Big] \\
&= 2\exp\Big(\mathbf{X}^T\boldsymbol{\beta} + \frac{1}{2}\sigma^2\Big)\Phi\Big(\frac{\sigma\xi}{\sqrt{1+\xi^2}}\Big) = 2\exp\Big(\mathbf{X}^T\boldsymbol{\beta} + \frac{1}{2}\sigma^2\Big)\Phi\Big(\frac{\sigma\xi}{\sqrt{1+\xi^2}}\Big)
\end{aligned}$$

[1] When a sequence of measurements is normally distributed with mean zero and variance $\sigma^2 = 1/\sqrt{2}$, the probability that the error of each measurement 'erf(r)' lies in a certain range between $-r < \text{error} < r$. See Wang et al. 2019.

For the 2nd moment ($p = 2$), this gives

$$
\begin{aligned}
E[S^{p=2}|\mathbf{X}] &= \exp\Big(2\mathbf{X}^T\boldsymbol{\beta} + 2\sigma^2\Big)\Big[1 + \mathrm{erf}(\frac{2\sigma\delta}{\sqrt{2}})\Big] \\
&= \exp\Big(2\mathbf{X}^T\boldsymbol{\beta} + 2\sigma^2\Big)\Big[\cancel{1} + 2\Phi\Big(\frac{2\sigma\xi}{\sqrt{1+\xi^2}}\Big) - \cancel{1}\Big] = 2\exp\Big(2\mathbf{X}^T\boldsymbol{\beta} + 2\sigma^2\Big)\Phi\Big(\frac{2\sigma\xi}{\sqrt{1+\xi^2}}\Big)
\end{aligned}
$$

Therefore, the variance $V(S_h|\mathbf{X}_h)$ and the sum of the variance $\mathbb{S}_H^2$ can be obtained as below

$$
\begin{aligned}
V(S_h|\mathbf{X}_h) &= E[S_h^2|\mathbf{X}_h] - E^2[S_h|\mathbf{X}_h] \\
&= 2\exp\Big(2\mathbf{X}_h^T\boldsymbol{\beta} + 2\sigma^2\Big)\Phi\Big(\frac{2\sigma\xi}{\sqrt{1+\xi^2}}\Big) - \left[2\exp\Big(\mathbf{X}_h^T\boldsymbol{\beta} + \frac{1}{2}\sigma^2\Big)\Phi\Big(\frac{\sigma\xi}{\sqrt{1+\xi^2}}\Big)\right]^2
\end{aligned}
$$

$\cdots$ to be discussed

Appendix B

For Chapter 2

B.1 Discussion on Explainability and Uncertainty

As discussed in Chapter 1, the ease of interpreting the risk premium model facilitates communication between actuaries and stakeholders, especially when assessing new risk scenarios and addressing problems that require integrating knowledge from multiple domains to make better decisions. Above all, the intuitive form - a linear combination of the rating factors (covariates) with the parameters - of the regression-based risk premium modeling maximizes model explainability the best. This results in its widespread popularity among actuaries (Spedicato et al. 2018). For example, by reading the regression parameters, actuaries can easily quantify the impact of each rating factor on insurance risk, and distribute the risk premium amongst the policyholders, ensuring that policyholders with higher risk profiles pay higher premiums than those with lower risk.

Apart from explainability, the regression-based frameworks also provide various tools for uncertainty assessment. Jackson and Sharples 2010 specify that there are two major sources of uncertainty in the regression-based framework - *model parameters* and *model structure*[1]. For a simple model that describes a single aspect of the unknown of interest, the uncertainty assessment primarily centers around

[1]The uncertainty of model parameters and model structure refers to a chance of misspecification bias regarding parameter choice and model choice respectively (Jackson and Sharples 2010)

the model parameter, which can be quantified through standard errors, confidence intervals, hypothesis tests, etc. For a complex model that describes many different aspects of the unknown of interest, a performance metric such as Akaike Information Criterion (AIC)[2] can be used to assess the uncertainty on the model structure (Bilcke et al. 2011). However, Russell 2005 alleges that it is difficult to take into account both sources of uncertainty harmoniously because constructing the distributions over the selected parameters and structures is not straightforward, and thus tracking and quantifying the propagation of uncertainty through the model can be challenging. This might lead to under-estimating overall uncertainty in the model.

Despite their complex specifications, Bayesian risk premium models can still be as explainable as their regression-based counterparts. This framework is inherently explainable due to its transparent nature. It facilitates interpretable model development through tailored model component specifications (Nuti et al. 2021) and relies on consistent inference methods, such as Monte Carlo sampling. Moreover, its predictions are accompanied by probabilistic reasoning, offering outcomes with associated probabilities (Clare et al. 2022). Nonetheless, challenges persist, particularly when navigating complex models or defending the choice of prior distributions.

This explainable quality is particularly amplified by allowing for the articulation of causality (Jaynes 2003; Letham et al. 2015; Guo and Riebler 2017). As seen from the case of GLMs, the model is deemed explainable by manifesting relationships between variables; however, the relationships learned by classical modeling framework like GLMs are not necessarily indicative of causality (Wood 2005). The Bayesian framework employs conditional probability as a building block to articulate causal relationships between events, and systematically incorporates evidence when new information becomes available (Gopnik and Wellman 2012). This can help clarify the reasoning in making a particular decision based on the modeling results. Take as an example Equation (B.1) which depicts the natural development of causal relationships in the Bayesian framework. Let's say we have two different

[2]AIC refers to an estimator of prediction error and, consequently, the relative quality of statistical models is evaluated (Wagenmakers and Farrell 2004)

covariates - $\mathbf{z}, \mathbf{x}$ - and one outcome variable Y. With an influx of new information - $\theta_z, \theta_x, \theta_Y$ - that explains each variable we have, a series of probabilistic weights - $f(\mathbf{z}|\theta_z), f(\mathbf{x}|\mathbf{z}, \theta_x), f(Y|\mathbf{x}, \theta_Y)$ - are developed and assigned to each component of the model - $p(\theta_z), p(\theta_x), p(\theta_Y)$ - to describe their joint relationships

$$p(\mathbf{z}, \theta_z, \mathbf{x}, \theta_x, Y, \theta_Y) = f(\mathbf{z}|\theta_z)\, p(\theta_z) \cdot f(\mathbf{x}|\mathbf{z}, \theta_x)\, p(\theta_x) \cdot f(Y|\mathbf{x}, \theta_Y)\, p(\theta_Y) \tag{B.1}$$

This multiplication in Equation (B.1) displays a simple causal relationship in which Y results from $\mathbf{x}$, while $\mathbf{x}$ results from $\mathbf{z}$ for example. In this way, the combination of all components elucidates the relationship between variables in detail, rendering the model more transparent (Anthony 2006).

Another essential benefit of the Bayesian framework is the coherent propagation of uncertainty from both parameter and structure (Droguett and Mosleh 2008). Unlike classical methods, which handle parameter uncertainty well but struggle with structural uncertainty, Bayesian methods seamlessly integrate both, offering a more comprehensive assessment of variation (Jackson and Thompson 2009).

To illustrate this, we take an example of Bayesian Model Averaging (BMA) which combines the predictions from multiple models into a single prediction (Hoeting et al. 1999). Consider a vector of outcomes $Y = (Y_1, Y_2, \cdots)$ and a matrix $\mathbf{X} = (\mathbf{x}_1, \mathbf{x}_2, \cdots)$ for different covariate vectors. $\boldsymbol{D}$ denotes a complete dataset, which includes both Y and $\mathbf{X}$. A finite set of specific models $\boldsymbol{M} = (M_1, M_2, \cdots, M_P)$ in which each model M_p explains the relationship between the outcome Y_p and covariate $\mathbf{X}_p$ (i.e. M_p can be "$\mathbf{x}_1 + \mathbf{x}_2$" or "$\mathbf{x}_1^2\mathbf{x}_2 + \mathbf{x}_8$", etc., and its parameters can exhibit dependencies in a given structure), the expression for the predictive model (averaged likelihood) is

$$f(\hat{Y}|\boldsymbol{D}) = \int_{M_p} f(\hat{Y}, M_p|\boldsymbol{D})\, d\boldsymbol{M} = \int_{M_p} f(\hat{Y}|M_p)\, p(M_p|\boldsymbol{D})\, d\boldsymbol{M}, \quad \text{for } p = 1, 2, \cdots P \tag{B.2}$$

where $f(\hat{Y}|M_p)$ is a outcome model and $p(M_p|\boldsymbol{D})$ is the mixing weight (posterior probability) for this outcome model, reflecting the importance or credibility of the model given the observed dataset. These two components are crucial for conducting

uncertainty quantification because, in the complex modeling setting ($p = 1, 2, \cdots P$), they constitute the joint term $f(\hat{Y}, M_p|\boldsymbol{D})$. This allows simultaneous consideration of both model parameters and structure, enabling the handling of complex model uncertainty by averaging over multiple models (Droguett and Mosleh 2008). The implication is that, once we obtain its full distribution $f(\hat{Y}|\boldsymbol{D})$, the uncertainty quantification of both model parameter and structure can be carried out by constructing the credible interval (for the unknown parameter) or prediction interval (for unknown data) for the resulting estimation (Pollino and Henderson 2010).

B.2 GLMs and Risk Premium

We have an insured claim amount Y_j and count N_j given the covariate vectors $\mathbf{X}_j = \{\mathbf{z}_j, \mathbf{x}_j\}$ in risk class j.

Suppose that risk class j follows a certain distribution of the exponential family, then the predictive values $E[Y_j|\mathbf{X}_j]$ for the risk class j can be obtained from a series of GLM equations $g(\ E[Y_j|\mathbf{X}_j]\ )$ as below (Nelder and Wedderburn 1972)

$$
\begin{aligned}
&\text{for } Y_j \sim \text{Gaussian } (\mu_j, \sigma_j^2) : E[Y_j|\mathbf{X}_j] = \mathbf{X}_j^T\boldsymbol{\beta}_j = \mu_j \\
&\text{for } Y_j \sim \text{log-normal } (\mu_j, \sigma_j^2) : E[Y_j|\mathbf{X}_j] = \exp(\mathbf{X}_j^T\boldsymbol{\beta}_j + \frac{1}{2}\sigma_j^2) \\
&\text{for } Y_j \sim \text{Inverse Gaussian } (\mu_j, \lambda_j) : \frac{1}{E[Y_j|\mathbf{X}_j]^2} = \mathbf{X}_j^T\boldsymbol{\beta}_j = \frac{1}{\mu_j^2} \\
&\text{for } Y_j \sim \text{Poisson } (\lambda_j) : E[Y_j|\mathbf{X}_j] = \exp(\mathbf{X}_j^T\boldsymbol{\beta}_j) = \lambda_j \\
&\text{for } Y_j \sim \text{Gamma } (k_j, \lambda_j) : \frac{k_j}{E[Y_j|\mathbf{X}_j]} = \mathbf{X}_j^T\boldsymbol{\beta}_j = \lambda_j \\
&\text{for } Y_j \sim \text{Binomial } (n_j, p_j) : \frac{E[Y_j|\mathbf{X}_j]}{n_j - E[Y_j|\mathbf{X}_j]} = \exp(\mathbf{X}_j^T\boldsymbol{\beta}_j) = \frac{p_j}{1-p_j} \\
&\text{for } Y_j \sim \text{Negative Binomial } (\xi_j, \psi_j) : \frac{\psi_j(\frac{E[Y_j|\mathbf{X}_j]}{E[Y_j|\mathbf{X}_j]+\psi_j})}{\frac{\psi_j}{E[Y_j|\mathbf{X}_j]+\psi_j}} = \exp(\mathbf{X}_j^T\boldsymbol{\beta}_j) = \xi_j
\end{aligned}
\tag{B.3}
$$

With the presence of known/unknown risk classes, however, GLMs are required to cope with correlation or random variation (heterogeneity) among policyholders on the same class. This is crucial in ensuring proper allocation of the premium.

Concerning this, Ohlsson and Johansson 2010 show that GLMs can be extended to Generalized Linear Mixed Models (GLMMs) by adding an extra class-specific term to the linear predictor of the GLM equation to accommodate the variation of cluster effect, and thus deals with the heterogeneity.

B.3 GAMs, MARSs and Risk Premium

Suppose we model the insured claim amounts on the risk class j by adding smoothing terms of covariates, then the final linear predictor (that can replace the linear predictor shown in Equation (B.1)) of GAMs has the form (Hastie et al. 2009)

$$\begin{aligned} g(\ E[Y_j|\mathbf{X}_j]\ ) =& \beta_{j0} + \sum_p f_p(\mathbf{x}_{jp}) + \sum_{p\neq q} f_{p,q}(\mathbf{x}_{jp}\ \mathbf{x}_{jq}) \\ =& \beta_{j0} + f_1(\mathbf{x}_{j1}) + f_3(\mathbf{x}_{j3}) + f_{2,3}(\mathbf{x}_{j2}\ \mathbf{x}_{j3}) + \cdots \end{aligned} \tag{B.4}$$

where each $f_p(\mathbf{x}_p)$ is a smoothing term defined as $\Sigma_{m=1}^{M}\beta_m\ h_m(\mathbf{x}_p)$. Note that $h_m(\cdot)$ denotes a *basis function* that produces m-dimensional piecewise components over m-disjoint regions to capture the non-linear effect of each covariate. By dividing the domain of each covariate into M-multiple intervals, a basis function $h_m(\mathbf{x}_p)$ develops different polynomials or splines for each interval, which has been found to return a remarkable model fit (Brockett et al. 2014).

However, excessive wiggling has been found to cause overfitting as well, and thus it is crucial to control the complexity of the smoothing functions. To this end, Hastie 2017 suggests a regularization method. To be specific, Hastie 2017 defines the objective function (a.k.a the *loss function*) of the Penalized Residual Sum of Squares

(PRSS) with a tuning (smoothing) parameter $\lambda \geq 0$ to penalize its wiggliness

$$\begin{aligned}
&\text{PRSS}(\ \beta_0, f_1, f_2, \cdots f_{1,P}, \cdots, f_{P-1,P}\ ) \\
&\qquad = \sum_{i=1}^{N} \left(Y_{hi} - \beta_0 - \sum_{p=1}^{P} f_p(x_{pi}) - \sum_{p \neq q} f_{p,q}(x_{pi} x_{qi}) \right)^2 \\
&\qquad + \sum_{p=1}^{P} \lambda_p \int f_p''(t_p)^2 \, dt_p + \sum_{p \neq q} \lambda_{p \neq q} \int f_{p,q}''(t_p t_q)^2 \, dt_p t_q
\end{aligned} \tag{B.5}$$

By minimizing the loss function in Equation (B.3), wiggly models are penalized more heavily than smooth models, which helps obtain the optimal regression parameter values β_m. For further details, see Hastie et al. 2009; Kaivanipour 2015; Strežo 2019.

MARSs use the same form of linear predictor as that shown in Equation (B.2), but the basis function $h_m(\mathbf{x}_p)$ in the smoothing term $f_p(\mathbf{x}_p) = \Sigma_{m=1}^{M} \beta_m \ h_m(\mathbf{x}_p)$ is defined differently. In MARSs, each basis function $h_m(\mathbf{x}_p)$ comes in pairs (a.k.a *reflected pair*), taking the form: $h_m(\mathbf{x}_p)_+ = \max(x_{pi} - K_m,\ 0)$ and $h_m(\mathbf{x}_p)_- = \max(K_m - x_{pi},\ 0)$ simultaneously for each interval m on the covariate $\mathbf{x}_p$. This means that each interval m is split again into another two sub-intervals: $x_p > K_m$ and $K_m > x_p$, and two piecewise linear models (two straight lines: $Y_{hi} = \pm\ x_{pi} \mp K_m$) join together, creating a curve (defining an inflection point) in each interval m. The point K_m splitting the interval m into two, and where the curve changes its slope is called a "knot" (Francis 2003).

Using MARSs, a flexible risk premium model with more stable estimates can be developed by strategically placing more knots in the regions where the claim amount changes rapidly and fewer knots in the regions where the claim amount remains relatively stable. As for concerns about overfitting, Francis 2003; Hastie et al. 2009 suggest limiting the number of knots by measuring a goodness-of-fit with a statistical criterion such as Generalized Cross-Validation (GCV) throughout the modeling sequence.

B.4 GLMs with Varying Intercept

Consider the linear predictor that defines the risk cluster j: $g(E[Y_j|\mathbf{X}_j]) = \beta_{j0}\mathbb{1}_j + \mathbf{X}_j^T\boldsymbol{\beta}_j$ where $\mathbf{X}_j^T\boldsymbol{\beta}_j$ displays the cluster mean j, and $\beta_{j0}\mathbb{1}_j$ explains the unique deviation from the cluster mean j. If stacking the linear predictors for all observations $i = 1, \cdots, N$ over all clusters $j = 1, \cdots J$, then we can obtain the general form of the linear predictor with the class-specific effect term

$$g(\ E[Y|\mathbf{X}]\ ) = \boldsymbol{Z}^T\boldsymbol{\alpha}_0 + \mathbf{X}^T\boldsymbol{\beta}, \qquad \boldsymbol{\alpha}_0 \sim \boldsymbol{N}(0,\ \boldsymbol{D}) \tag{B.6}$$

where $\boldsymbol{Z}$ and $\mathbf{X}$ are $J \times N$ and $P \times N$ model matrices, $\boldsymbol{\alpha}_0$ and $\boldsymbol{\beta}$ are $J \times 1$ and $P \times 1$ vectors of parameters, and $\boldsymbol{D}$ is a $J \times J$ covariance matrix for $\boldsymbol{\alpha}_0$ (Hastie et al. 2009). The intuition of the linear predictor in Equation (B.4) is that the stochastic model $\boldsymbol{Z}^T\boldsymbol{\alpha}_0$ can account for the variability that is not explained by $\mathbf{X}^T\boldsymbol{\beta}$, and this resolves the heterogeneity issue. However, we contend that this approach places reliance on knowledge about the risk classes $j = 1, \cdots J$ available at hand, without adequately considering the potential presence of diverse risk class scenarios that are not discovered yet.

B.5 EM Algorithm with MAR assumption

In the EM algorithm in Section 2.1.2, the implication is that the missing covariate values in $\mathbf{x}$ can be recovered from understanding the mixing weight built upon relation between the missingness indicator z_i and the observed data x_{z_i}, which are governed by the cluster parameter θ_{z_i}, and this aligns with the definition of the MAR assumption. In order to find the cluster parameter θ_{z_i}, Ng and Krishnan 2012

suggest to maximize the log-likelihood function given by

$$
\begin{aligned}
ln \prod_{i=1}^{n} p(x_i|\theta) &= \sum_{i=1}^{n} ln \sum_{z_i} \boldsymbol{\omega}(z_i|x_{z_i}, \theta_{z_i})\ p(x_{z_i}|\theta_{z_i}) \\
&= \sum_{i=1}^{n} ln \Big[\boldsymbol{\omega}(z_i = 1|x_{z_i=1}, \theta_{z_i})\ p(x_{z_i=1}|\theta_{z_i}) + \boldsymbol{\omega}(z_i = 0|x_{z_i=0}, \theta_{z_i})\ p(x_{z_i=0}|\theta_{z_i}) \Big]
\end{aligned}
\tag{B.7}
$$

Equation (B.5) exhibits that there are two unknown quantities - $z_i,\ \theta_{z_i}$ - to estimate within the Maximum Likelihood Estimation (MLE) framework. However, as z_i is an unknown, it is impossible to estimate the model parameter θ_{z_i} directly, which is the problem of the MLE approach (Van Dyk 2001).

With this issue, the EM algorithm extends this MLE solution by averaging out (taking the expectation of) the log-likelihood function over the mixing weight term $\boldsymbol{\omega}(z_i|x_{z_i}, \theta)$ with the inclusion of the 'provisional' model parameter $\theta_{(t)}$

$$
Q(\theta, \theta_{(t)}) = E_{\theta_{(t)}} \Big[ln \sum_{z_i} \boldsymbol{\omega}(z_i|x_{z_i}, \theta)\ p(x_{z_i}|\theta) \ \Big|\ x_{z_i}; \theta_{(t)} \Big] \tag{B.8}
$$

which represents an objective function for the algorithm. Accordingly, the mixing weight term $w(z_i|x_{z_i}, \theta_{(t)})$ is re-defined for each data point (Tanner 2010)

$$
E[\boldsymbol{\omega}(z_i = 1|x_{z_i=1}, \theta_{(t)})] = \frac{p(x_{z_i=1}|\theta_{(t)})}{p(x_{z_i=1}|\theta_{(t)}) + p(x_{z_i=0}|\theta_{(t)})} \tag{B.9a}
$$

$$
E[\boldsymbol{\omega}(z_i = 0|x_{z_i=0}, \theta_{(t)})] = \frac{p(x_{z_i=0}|\theta_{(t)})}{p(x_{z_i=1}|\theta_{(t)}) + p(x_{z_i=0}|\theta_{(t)})} \tag{B.9b}
$$

which can be utilized to compute the value of the objective function in Equation (B.6). Given that the starting value of the provisional model parameter $\theta_{(t)}$ can be randomly determined, one can easily attain the expected mixing weight values in Equation (B.7) for each data point x_{z_i}, which is known as the *Expectation Step* in the EM algorithm.

The new and proper model parameter value $\theta_{(t+1)}$ to use next can be computed as $\theta_{(t+1)} = \text{argmax}_\theta\, Q(\theta, \theta_{(t)})$ based on Equation (B.6), which is known as *Maximization*

Step, and the result will be utilized to approximate the distributions (clusters) of the missing values $p(x_{z_i=1})$ and non-missing values $p(x_{z_i=0})$.

B.6 RC, SIMEX with NDB assumption

To break down the RC, let x_i^* represent the mismeasured covariate value at hand, ϵ_i denote an error term, x_i^{obv} signify the covariate value from the extra studies for the gold standard information (substituting in place of the unknown true values x_i), and x_i^{rc} be the final estimated covariate value developed with RC. The framework given by Skrondal and Kuha 2012 to recover the true covariate value is described as follows

$$x_i^* = x_i + \epsilon_i, \quad \epsilon_i \sim N(0,\ \sigma_\epsilon^2) \tag{B.10a}$$

$$x_i^{rc} = E[x_i|x_i^*] = \lambda x_i^* + (1-\lambda)E[x_i^{obv}] \tag{B.10b}$$

$$\text{where } \lambda = \frac{V(\mathbf{x}^{obv})}{V(\mathbf{x}^*)} = \frac{\sigma_x^2}{\sigma_x^2 + \sigma_\epsilon^2} \tag{B.10c}$$

Now we can regress the outcome Y on the corrected covariate $\mathbf{x}^{rc}$, hence $g(\ E[Y|\mathbf{X}]\ ) = \mathbf{X}^T\boldsymbol{\beta}$ as shown in Equation (B.1). The challenging part, however, is that the error variance σ_ϵ^2 in Equation (B.8a) cannot be estimated from the given information at hand, and thus, as mentioned in Section 2.1.2, requires external reference such as validation study, replication study, etc., which is not always available in real life (Carroll et al. 2006).

In SIMEX, the mismeasured covariate is defined as $\mathbf{x}^* = U + \epsilon$ where U is the unobservable true covariate that we seek, and the measurement error follows $\epsilon_i \sim \mathbf{N}(0,\ \sigma_\epsilon^2)$. If the true covariate is considered normally distributed, then the distribution of the mismeasured covariate $\mathbf{x}^*$ can be expressed as (Apanasovich et al. 2009)

$$x_i^* \sim \mathbf{N}\Big(E[U_i], \quad \sigma_U^2 + \sigma_\epsilon^2\Big) \tag{B.11}$$

where $\sigma_U^2 \perp \sigma_\epsilon^2$. Assuming the variance of the measurement error σ_ϵ^2 in Equation (B.8) is known, the artificial error term $\sqrt{\lambda_t}\epsilon_i, \sim \mathbf{N}(0,\ \lambda_t\sigma_\epsilon^2)$ can be formulated where λ_t is a set of values indicating the noise level $0 = \lambda_1 < \lambda_2, \cdots < \lambda_t, \cdots < \lambda_T$. By incorporating this artificial error term into the mismeasured covariate $\mathbf{x}^*$, the surrogate covariate $\mathbf{w}^*(\lambda_t) = U + \sqrt{\lambda_t}\epsilon$ can be developed to simulate the change of the model parameter β^* over the different values of λ_t. To perform this simulation, the distribution of $\mathbf{x}^*$ in Equation (B.9) is re-defined with the surrogate covariate $\mathbf{w}^*(\lambda_t)$

$$w_i^*(\lambda_t) \sim \mathbf{N}\Big(E[U_i], \quad \sigma_U^2 + (1+\lambda_t)\sigma_\epsilon^2\Big) \tag{B.12}$$

Accordingly, the linear predictor of GLMs shown in Equation (B.1) can be re-defined by replacing the mismeasured covariate $\mathbf{x}^*$ with the surrogate covariate $\mathbf{w}_p^*(\lambda_t)$

$$g(\ E[Y|\mathbf{X}]\ ) = \beta_0 + \beta_1\mathbf{x}_1 + \ldots \beta_p^*\mathbf{w}_p^*(\lambda_t) + \ldots \beta_P\mathbf{x}_P \tag{B.13}$$

SIMEX takes two steps to correct the model parameter estimation. The first step involves resampling multiple datasets based on the existing data. Each dataset incorporates additional artificial errors, using the surrogate covariate defined in Equation (B.10). For each dataset, the model parameters shown in Equation (B.11) are estimated, and SIMEX tracks down a trend of estimated parameter values $\hat{\beta}^*$ versus the variance of the artificial errors $(1+\lambda_t)\sigma_\epsilon^2$ sampled for the dataset development. In the second step, once the curve of this trend is properly identified, the true model parameters can be obtained. This is because SIMEX extrapolates this trend back to the point where $\lambda_t = -1$ that makes the variance of the artificial error $(1+\lambda_t)\sigma_\epsilon^2$ become zero (noise-free) (Apanasovich et al. 2009).

B.7 BGAM and VAE

Given the linear predictor of GAMs in Equation (B.2), BGAMs add a categorical covariate $\mathbf{v}_j = \{v_1, \cdots, v_q\}^T$ to the linear predictor

$$g(\ E[Y_j|\mathbf{X}_j]\ ) = f_1(\mathbf{x}_{j1}) + f_3(\mathbf{x}_{j3}) + f_{2,3}(\mathbf{x}_{j2}\ \mathbf{x}_{j3}) + \cdots + \mathbf{v}_j^T\boldsymbol{\gamma}_j \tag{B.14}$$

where $\mathbf{v}_j^T\boldsymbol{\gamma}_j$ is the category-related, strictly linear part of the predictor that accounts for the variations in the risk distribution. As for the smoothing functions $f_p(\cdot) = \Sigma_{m=1}^{M}\beta_m^{(p)}\ h_m(\mathbf{x}_p)$, Lang and Brezger 2004 suggests Bayesian P-splines, approximated polynomial lines of degree l and with equally spaced knots across the domain of each covariate. That is to say, each P-spline (smoothing function) can have a linear combination of the base functions $h_m(\mathbf{x}_p)$ with size M: counts of knots + l. In the regression coefficient vector $\boldsymbol{\beta}^{(p)} = \{\beta_1^{(p)}, \cdots, \beta_M^{(p)}\}^T$, $m = 1, \cdots, M$ of each P-spline $f_p(\cdot)$, the following is considered with Gaussian error ϵ_m

$$\begin{aligned}
\beta_m^{(p)}|\tau_m^2\ &\propto\ \exp\Big(-\frac{1}{2\tau_m^2}\beta_m^{(p)T}K_m\beta_m^{(p)}\Big) \\
\text{1st order random walk:}\ \ \beta_m^{(p)} &= \beta_{m-1}^{(p)} + \epsilon_m \\
\epsilon_m &\sim \boldsymbol{N}(0,\ \tau_m^2) \\
\tau_m^2 &\sim \boldsymbol{InvGa}(a_m, b_m)
\end{aligned} \tag{B.15}$$

where K_m is a penalty matrix based on the prior assumptions about smoothness, and τ_m^2 controls the smoothness (Denuit and Lang 2004). By introducing an additional hyperprior a_m, b_m for the variance parameters τ_m^2, it becomes possible to estimate the level of smoothness concurrently with the regression coefficients $\beta_m^{(p)}$. In a nutshell, instead of relying on a penalty term in the frequentist GAM for regularization, BGAM additionally introduces stochastic random walk terms as smoothness priors, and estimates the regression coefficients by implementing Markov Chain Monte Carlo (MCMC) algorithms.

To understand the VAE, consider the standard form of the predictive model

given by

$$f(Y_{hi}) = \int_z f(Y_{hi}, z)dz = \int_z f(Y_{hi}|z)\ p(z)dz, \quad i \in \{1, ..., n\} \tag{B.16}$$

where Y_{hi} is the given data, z is a latent variable, and $f(Y_{hi})$ is a predictive distribution. The goal is to compare the output distribution of the VAE decoder $f(Y_{hi}|z)$ with the initial data distribution $f_0(Y_{hi})$ to detect heterogeneous features. In order to retrieve $f(Y_{hi}|z)$, the VAE encoder constructs the latent space $p(z|Y_{hi})$ using $p(z)$ which is estimated with the Bayesian *Variational Inference* [3] technique. Then, the resulting distribution of $f(Y_{hi}|z)$ can be compared with the original data distribution $f_0(Y_{hi})$. The essence of this technique is to learn important features of complex data by capturing the latent space. To further illustrate this, we can consider the objective function of the VAE developed with Jansen's lower bound given by

$$\ln f_w(Y_{hi}) \geq E_{q_\phi(z|Y_{hi})}\big[f_w(Y_{hi}|z)\big] - D_{KL}\big(q_\phi(z|Y_{hi})||p(z)\big) \tag{B.17}$$

where $E_{q_\phi(z|Y_i)}\big[f_w(Y_{hi}|z)\big]$ is the expected value of the reconstruction probability as a set of average weights for the resulting data points $Y_{hi}|z$, the *Kullback-Leibler divergence* term $D_{KL}(\cdot)$ is the regularizor, $q_\phi(z|Y_{hi})$ is the latent space term constructed by the encoder and $p(z)$ is simply the reference distribution - Gaussian density - to assess the quality of the latent space construction (An and Cho 2015). From the objective function in Equation (B.15), one can envisage the mixture form of the distribution of interest in which the mixing weight term corresponds to $q_\phi(z|Y_{hi})$, and the observed data density term corresponds to $f_w(Y_{hi}|z)$.

[3] Variational inference as a Bayesian parameter estimation technique, seeks to approximate the true posterior with a variational distribution that we can access more easily. See (An and Cho 2015)

B.8 BMI with MAR & NDB assumption

To further elucidate Bayesian multiple Imputation (BMI), we employ the notation where $\mathbf{x}$ and $\mathbf{x}^-$ represent the observed and missing data vectors, respectively, for a given dataset. Let $f_0(\mathbf{x}^-|\mathbf{x},\ \boldsymbol{\theta})$ and $p_0(\boldsymbol{\theta})$ denote the imputation function and parameter distribution (uninformative prior). In order to sample from the imputation function, new model parameter $\boldsymbol{\theta}^{(m)}$ should be first drawn from the posterior $p(\boldsymbol{\theta}^{(m)}|\mathbf{x},\ p_0(\boldsymbol{\theta}))$ to update the imputation function $f(\mathbf{x}^-|\mathbf{x},\ \boldsymbol{\theta}^{(m)})$. In this process, the uncertainty around $\boldsymbol{\theta}^{(m)}$ can be closely related to the variability observed between imputed datasets (Schafer 1997). With a complex pattern of missingness, Markov Chain Monte Carlo (MCMC) can be used to approximate the posterior $p(\boldsymbol{\theta}^{(m)}|\mathbf{x},\ p_0(\boldsymbol{\theta}))$ because the posterior often fails to have a closed form. (Robert and Casella 1999).

Under the NDB assumption, let $\mathbf{x}_i^* = \{x^*_{k_i=1_i}, \cdots, x^*_{k_i=K_i}\}$ denote a vector of K_i mismeasurements of the covariate value x_i (each covariate value is measured multiple times), and thus $x^*_{k_i} = x_i + \epsilon_{k_i}$ where $\epsilon_{k_i} \sim N(0, \sigma^2_\epsilon)$. The goal here is to construct the imputation function $f(x_i|\mathbf{x}_i^*,\ Y_{hi})$ from which the true covariate value x_i can be drawn. Assuming the true covariate x_i is Gaussian distributed, one can focus on computing the mean and variance of $f(x_i|\mathbf{x}_i^*,\ Y_{hi})$.

Using a random effect term $b_i \sim N(0, \sigma^2_{\mathbf{x}|Y})$ representing the residual from the regression of x_i on Y_{hi}, the mismeasured covariate $x^*_{k_i}$ can be re-expressed as $x^*_{k_i} = \alpha_0 + \alpha_1 Y_{hi} + b_i + \epsilon_{k_i}$. Thus, the mean and variance of the imputation function $f(x_i|\mathbf{x}_i^*,\ Y_{hi})$ can be given by

$$\begin{aligned} E[x_i|\mathbf{x}_i^*,\ Y_{hi}] &= \alpha_0 + \alpha_1 Y_{hi} + E[b_i|\mathbf{x}_i^*, Y_{hi}] \\ V(x_i|\mathbf{x}_i^*,\ Y_{hi}) &= V(b_i|\mathbf{x}_i^*, Y_{hi}) \end{aligned} \tag{B.18}$$

As for the moments of the conditional random effect term in Equation (B.16),

they can be boiled down to as below after some algebra (Bartlett 2010)

$$
\begin{aligned}
E[b_i|\mathbf{x}_i^*,\ Y_{hi}] &= \frac{\sigma^2_{\mathbf{x}|Y}}{\sigma^2_{\mathbf{x}|Y} + \sigma^2_\epsilon/k_i}\ \left[\overline{\mathbf{x}_i^*} - \alpha_0 - \alpha_1 Y_{hi}\right] \\
V(b_i|\mathbf{x}_i^*,\ Y_{hi}) &= \sigma^2_{\mathbf{x}|Y}\left[1 - \frac{\sigma^2_{\mathbf{x}|Y}}{\sigma^2_{\mathbf{x}|Y} + \sigma^2_\epsilon/k_i}\right]
\end{aligned}
\tag{B.19}
$$

where $\overline{\mathbf{x}_i^*}$ denotes the average value of x_i's total K_i measurements. Therefore, the imputation function can be constructed, and BMI imputes all mismeasured covariate values within the Bayesian inference framework as described in the MAR case.

Appendix C

For Chapter 3

C.1 Discussion on Distribution and Risk Measure

C.1.1 Full Distribution for Risk Measure

As mentioned in Chapter 1, the goal of pricing non-life products is to balance the insurer's costs and profits at the margins (to meet the insurer's particular objectives) by setting premiums to cover the costs as well as to generate sufficient profits over time to please other stakeholders. If a premium for an insurance policy is set too high, customers might be driven away to other competing firms offering much cheaper prices. If an insurer sets premiums too low, they may initially attract customers, but eventually risk insolvency as their losses exceed reserves when their premiums fail to cover future claims (Ohlsson and Johansson 2010). Due to its complex dynamics, the premium setting requires collaboration between many professionals[1] to establish proper strategies and meet the insurer's specific objectives (Parodi 2023).

To be specific, the premium in general consists of the expected total aggregate claim amount (an anticipated sum of all claims that the company expects to pay out for all policyholders over a specific period), other running costs, and a profit

[1] Actuaries, underwriters, marketers, distributors, claims adjusters who have different insights on growth strategies, risks of interest, regulations, competitors' actions, etc. work together to find the optimal pricing strategies (Derrig and Meyers 2014)

margin. Let $\tilde{S}$ denotes the total aggregate claim amount, then

$$\text{Premium} = \underbrace{E[\tilde{S}] + \text{LAE} + \text{UWE}}_{\text{Insurer's cost}} + \text{PM} \tag{C.1}$$

where *Loss Adjustment Expense* (LAE) is the costs associated with reviewing and settling claims, *Underwriting Expense* (UWE) is the costs to cover an insurer's business operation activities [2] and *Profit Margin* (PM) is a level of profit determined by an insurer after covering all the costs (Werner and Modlin 2010). In order to achieve competitive pricing in the market, an insurer might seek to keep their premiums as low as possible. With regard to this, Kaas et al. 2008 provide an approximation of the lower limit of an insurer's cost defined in Equation (C.1) in a more practical sense

$$\text{Insurer's cost} \approx \; E[\tilde{S}] \; + \; \frac{1}{2} R \cdot V(\tilde{S}) \; + \; d \cdot u \tag{C.2}$$

where R is a *risk aversion* parameter derived from *ruin theory* (see Kaas et al. 2008 for the terminology), and d and u are the dividend rate and initial capital respectively, which together give an amount of yearly dividend for the shareholders who have supplied the insurer's initial capital u. The bottom line in Equation (C.2) is that setting the insurer's cost largely relies on a thorough understanding of the total aggregate claim $\tilde{S}$. As discussed in Chapter 1, the portion of the premium that covers the total aggregate claim $\tilde{S}$ is called the risk premium, and this is the starting point for this thesis. We aim to develop a full predictive distribution of $\tilde{S}$ for the risk premium modeling, based on accurate predictions of future total aggregate claim amount $E[\tilde{S}]$ and its variance $V(\tilde{S})$, even in the presence of model risk. At this point, we take a step back from a discourse on the insurer's cost and pricing strategy, and attempt to tackle the fundamental question of why we care about a full predictive distribution of the total aggregate claim amount $\tilde{S}$ rather than focus

[2] This includes acquisition of policyholders, commissions, evaluating applications, taxes, etc. (Parodi 2023).

on its best-predicted value (point estimates) such as $E[\tilde{S}]$ and $V(\tilde{S})$.

C.1.2 Why compute a full distribution of $\tilde{S}$?

In a nutshell, having a full predictive distribution allows for measuring the riskiness (volatility) of the insured aggregate claims, which enables actuaries to develop strategies to mitigate the adverse impact on insurers effectively (Asmussen and Steffensen 2020). The risk usually refers to financial losses by multiple random chances. Given a well-developed predictive distribution of the insured aggregate claims, the risk can be calculated by assessing the probability of the insured aggregate claims greater than a certain value $\tilde{s}$ positioned in the distribution (McNeil et al. 2015). This typically involves computing the complementary cumulative probability: $1 - F_{\tilde{S}} = P(\tilde{S} > \tilde{s})$ based on the full predictive distribution of $\tilde{S}$.

The calculated risk can thus serve as a useful tool in financial risk management. With the full predictive distribution, actuaries can assess and quantify the entire spectrum of potential losses to communicate to decision makers or regulators. For example, actuaries can use the risk measures based on the full predictive distribution to determine how to set capital reserves or risk limits by comparing them across portfolios of different segments such as geographic regions, risk profiles, etc. This helps optimize the insurer's capital allocation and improve resilience to adverse events because the risk measures deliver the data-driven evidence for sufficient reserves to cover potential losses within each segment (McNeil et al. 2015).

With this motivation in mind, a primary theme throughout this thesis is the acquisition of a full predictive distribution of $\tilde{S}$ for risk premium modeling. We examine the full predictive distribution of the total aggregate claim $\tilde{S} = \Sigma_{h=1}^{H} S_h$ under the premise that each observation $S_{h=1}, \cdots, S_{h=H}$ represents a sum of individual claim amounts $S_h = \Sigma_{i=1}^{N} Y_i$ produced from a single group policy h. This also means that the insurer's risk portfolio consists of H insurance contracts (policies), each associated with a different organization.

Appendix D

For Chapter 4

D.1 Inference Algorithm for Gustafson H.GLM

Algorithm (D.1) Gustafson Hierarchical LN-GLM Gibbs Sampling ($j = 1, \cdots J$)

Require: initialize

$\boldsymbol{\phi} = \{\boldsymbol{\beta}_0, \Sigma_{\beta_0}, u_0, v_0, m_0, \delta, q_0, \Lambda, \rho_{u1}, \rho_{u2}, \rho_{v1}, \rho_{v2}, c_0, d_0, g_0, h_0\}$

1: $\boldsymbol{\theta}^{(*)} = \boldsymbol{\theta}^{(old)}$: $\begin{cases} \boldsymbol{\beta}^{(old)} \sim \mathbf{MVN}(\boldsymbol{\beta}_0,\ \Sigma_{\beta_0}) & \triangleright \text{ for Complete pooling} \\ \sigma^{2(old)} \sim \mathbf{InvGa}(u_0/2,\ v_0/2) & \triangleright \text{ for Complete pooling} \end{cases}$

2: $(\boldsymbol{\theta}_1^{(old)}, \cdots \boldsymbol{\theta}_J^{(old)})$: $\begin{cases} \boldsymbol{\beta}_j^{(old)} \sim \mathbf{MVN}(\boldsymbol{\beta}_0,\ \Sigma_{\beta_0}) & \triangleright \text{ for No-pooling} \\ \sigma_j^{2(old)} \sim \mathbf{InvGa}(u_0/2,\ v_0/2) & \triangleright \text{ for No-pooling} \end{cases}$

3: **repeat**

4: Sample communal $\Sigma_{\beta_0}^+, \boldsymbol{\beta}_0^+, u_0^+, v_0^+$ from the posterior hyperpriors $\boldsymbol{hq}(\boldsymbol{\theta}^*)$[1]

[Stage.1] Sampling with Complete pooling

5: Sample $\boldsymbol{\theta}^{(new)}$ from the proposal densities $\boldsymbol{q}$: ▷ Choose the priors as $\boldsymbol{q}$.

$\boldsymbol{\beta}^{(new)} \sim \boldsymbol{q}_\beta : \mathbf{MVN}(\boldsymbol{\beta}_0^+,\ \Sigma_{\beta_0}^+), \quad \sigma^{2(new)} \sim \boldsymbol{q}_{\sigma^2} : \mathbf{InvGa}(u_0^+/2,\ v_0^+/2)$

6: **for** $h = 1, \cdots, H$ **do**

7: **for** $\boldsymbol{\theta}^{(new)} = \{\boldsymbol{\beta}^{(new)}, \sigma^{2(new)}\}$ **do**

8: Compute the transition ratio, using the outcome models:

$$Ratio_\theta = \frac{\prod_{h=1}^{H} f(\bar{Y}_h|\mathbf{X},\ \boldsymbol{\theta}^{(new)}) \cdot p_0(\boldsymbol{\theta}^{(new)}) \cdot \boldsymbol{q}_\theta(\boldsymbol{\theta}^{(old)})}{\prod_{h=1}^{H} f(\bar{Y}_h|\mathbf{X},\ \boldsymbol{\theta}^{(old)}) \cdot p_0(\boldsymbol{\theta}^{(old)}) \cdot \boldsymbol{q}_\theta(\boldsymbol{\theta}^{(new)})}$$

Sample $U \sim \boldsymbol{Unif}(0, 1)$

9: **if** $U < Ratio_\theta$ **then** $\boldsymbol{\theta}^{(*)} = \boldsymbol{\theta}^{(new)}$ otherwise $\boldsymbol{\theta}^{(*)} = \boldsymbol{\theta}^{(old)}$

10: **end if**

11: **end for**

12: Record $\boldsymbol{\theta}^{(*)}$

13: **end for**

[1] The posterior hyperpriors $\boldsymbol{hq}(.)$ in Line 4 are given by Equations (4.13, 4.14) in Chapter 4

Algorithm (D.1) *Cont.*

[Stage.2] Sampling with No pooling

14: **for** $j = 1, \cdots, J$ **do**

15: Sample $\boldsymbol{w}_j : \{\pi_j, \hat{\boldsymbol{\kappa}}_j, \hat{\lambda}_j^2, \tau_j^2\}$ from the posteriors (linking, covariates)[2]

16: Sample $\boldsymbol{\theta}_j^{(new)}$ from the proposal densities $\boldsymbol{q}$: ▷ Choose the priors as $\boldsymbol{q}$.

$$\boldsymbol{\beta}_j^{(new)} \sim \boldsymbol{q}_\beta : \mathbf{MVN}(\boldsymbol{\beta}_0^+, \Sigma_{\beta_0}^+), \quad \sigma_j^{2(new)} \sim \boldsymbol{q}_{\sigma^2} : \mathbf{InvGa}(u_0^+/2,\ v_0^+/2)$$

17: **for** $h = 1, \cdots, H$ **do**

18: **for** $\boldsymbol{\theta}_j^{(new)} = \{\boldsymbol{\beta}_j^{(new)}, \sigma_j^{2(new)}\}$ **do**

19: Compute the transition ratio, using the outcome models:

$$Ratio_\theta = \frac{\prod_{h=1}^H f(\bar{Y}_h|\mathbf{X},\ \boldsymbol{\theta}_j^{(new)}) \cdot p_0(\boldsymbol{\theta}_j^{(new)}) \cdot \boldsymbol{q}_\theta(\boldsymbol{\theta}_j^{(old)})}{\prod_{h=1}^H f(\bar{Y}_h|\mathbf{X},\ \boldsymbol{\theta}_j^{(old)}) \cdot p_0(\boldsymbol{\theta}_j^{(old)}) \cdot \boldsymbol{q}_\theta(\boldsymbol{\theta}_j^{(new)})}$$

Sample $U \sim \boldsymbol{Unif}(0, 1)$

20: **if** $U < Ratio_\theta$ **then** $\boldsymbol{\theta}_j^{(*)} = \boldsymbol{\theta}_j^{(new)}$ otherwise $\boldsymbol{\theta}_j^{(*)} = \boldsymbol{\theta}_j^{(old)}$

21: **end if**

22: Sift out and re-calculate the estimates $\boldsymbol{\theta}_j^{(*)}$ with Gustafson's Eq [3]

Compute $\beta_{j0}^{(*)} = \beta_{j0}^{(new)} - \frac{\beta_{j1}^{(new)} \hat{\kappa}_{j0} \tau_j^2}{\hat{\lambda}_j^2 - \hat{\tau}_j^2}$

Compute $\beta_{j1}^{(*)} = \frac{\beta_{j1}^{(new)} \hat{\lambda}_j}{\hat{\lambda}_j^2 - \tau_j^2}$

Compute $\beta_{j2}^{(*)} = \beta_{j2}^{(new)} - \frac{\beta_{j1}^{(new)} \hat{\kappa}_{j1} \tau_j^2}{\hat{\lambda}_j^2 - \hat{\tau}_j^2}$

Compute $\sigma_j^{2(*)} = \sigma_j^{2(new)} - \frac{\beta_{j1}^{2*} \tau_j^2 (\hat{\lambda}_j^2 - \tau_j^2)}{\hat{\lambda}_j^2} > 0$

23: **end for**

24: Record $\boldsymbol{\theta}_j^{(*)}$

25: **end for**

26: **end for**

27: **for** $h = 1, \cdots, H$ **do**

28: Record $LL_h = \ln\left[f(\mathbf{X}_h|\boldsymbol{w}_j) f(\bar{Y}_h|\mathbf{X}_h, \boldsymbol{\theta}_j^{(*)})\right]$ ▷ Monitor convergence

29: **end for**

30: **until** M posterior samples $(\boldsymbol{\theta}_{j=1,\cdots,J}^{(*)})$ obtained. ▷ M is a sufficient sample size

[2] The parameterizations of the posterior densities of $\boldsymbol{w}^{(*)}$ in Line 15 are detailed in Appendix E

[3] The Gustafson's Equation in Line 22 are detailed in D.2.

D.2 Derivation of Gustafson's Equations with Log-normal outcome

With the outcome model definitions:

$$f(\bar{Y}, \mathbf{x}^*, \mathbf{x} \mid \mathbf{z}) = \frac{1}{\bar{Y}\sqrt{2\pi\sigma_j^2}} \exp\{-\frac{1}{2}\Big[\frac{\log \bar{Y} - (\beta_{j0} + \beta_{j1}\mathbf{x} + \beta_{j2}\mathbf{z})}{\sigma_j}\Big]^2\}$$
$$\times \frac{1}{\sqrt{2\pi\tau_j^2}} \exp\{\frac{-(\mathbf{x}^* - \mathbf{x})^2}{2\tau_j^2}\} \times \frac{1}{\sqrt{2\pi\lambda_j^2}} \exp\{\frac{-(\mathbf{x} - \{\kappa_{j0} + \kappa_{j1}\mathbf{z}\})^2}{2\lambda_j^2}\}$$
$$f(\bar{Y}, \mathbf{x}^* \mid \mathbf{z}) = \frac{1}{\bar{Y}\sqrt{2\pi\hat{\sigma}_j^2}} \exp\{-\frac{1}{2}\Big[\frac{\log \bar{Y} - (\hat{\beta}_{j0} + \hat{\beta}_{j1}\mathbf{x}^* + \hat{\beta}_{j2}\mathbf{z})}{\hat{\sigma}_j}\Big]^2\}$$
$$\times \frac{1}{\sqrt{2\pi\hat{\lambda}_j^2}} \exp\{\frac{-(\mathbf{x}^* - \{\hat{\kappa}_{j0} + \hat{\kappa}_{j1}\mathbf{z}\})^2}{2\hat{\lambda}_j^2}\}$$

Now we aim to identify the expressions for $\widehat{\beta}_{j0}, \widehat{\beta}_{j1}, \widehat{\beta}_{j2}, \widehat{\sigma_j}, \widehat{\lambda_j}, \widehat{\kappa}_{j0}, \widehat{\kappa}_{j1}$ above

Unobserved complete case	**Incomplete case**
$\underbrace{f(\bar{Y}\|\mathbf{x}^*, \mathbf{x}, \mathbf{z})}_{\text{outcome}} \cdot \underbrace{f(\mathbf{x}^*\|\mathbf{x})}_{\text{measurement}} \cdot \underbrace{f(\mathbf{x}\|\mathbf{z})}_{\text{exposure}}$	$\underbrace{f(\bar{Y}\|\mathbf{x}^*, \mathbf{z})}_{\text{outcome}} \cdot \underbrace{f(\mathbf{x}^*\|\mathbf{z})}_{\text{exposure}}$
$= f(\bar{Y}, \mathbf{x}^*, \mathbf{x} \mid \mathbf{z})$	$= f(\bar{Y}, \mathbf{x}^* \mid \mathbf{z})$

by matching up the parameterizations from:

$$\int f(\bar{Y}, \mathbf{x}^*, \mathbf{x} \mid \mathbf{z})\, d\mathbf{x} \;=\; f(\bar{Y}, \mathbf{x}^* \mid \mathbf{z})$$

To begin with, we aim to evaluate the following integral:

$$\begin{aligned}
&\int f(\bar{Y}, \mathbf{x}^*, \mathbf{x} \mid \mathbf{z})\, d\mathbf{x} \\
&\quad = \int_{\mathbf{x}} \frac{1}{\sigma_j \bar{Y}\sqrt{2\pi}} \exp\{-\frac{1}{2}\Big[\frac{\log \bar{Y} - (\beta_{j0} + \beta_{j1}\mathbf{x} + \beta_{j2}\mathbf{z})}{\sigma_j}\Big]^2\} \\
&\qquad\qquad \times \frac{1}{\sqrt{2\pi\tau_j^2}} \exp\{\frac{-(\mathbf{x}^* - \mathbf{x})^2}{2\tau_j^2}\} \times \frac{1}{\sqrt{2\pi\lambda_j^2}} \exp\{\frac{-(\mathbf{x} - \{\kappa_{j0} + \kappa_{j1}\mathbf{z}\})^2}{2\lambda_j^2}\}\, d\mathbf{x} \\
&\quad = \int_{\mathbf{x}} \frac{1}{\bar{Y}\sigma_j\tau_j\lambda_j(2\pi)\sqrt{2\pi}} \exp\Bigg(-\frac{1}{2}\Big[\frac{\log \bar{Y} - (\beta_{j0} + \beta_{j1}\mathbf{x} + \beta_{j2}\mathbf{z})}{\sigma_j}\Big]^2 \\
&\qquad\qquad - \frac{1}{2\tau_j^2}(\mathbf{x}^* - \mathbf{x})^2 - \frac{1}{2\lambda_j^2}\big(\mathbf{x} - \{\kappa_{j0} + \kappa_{j1}\mathbf{z}\}\big)^2\Bigg)\, d\mathbf{x}
\end{aligned}$$

In the exponent above, the expressions can be expanded as

$$\begin{aligned}
&-\frac{1}{2}\Big[\frac{\log \bar{Y} - (\beta_{j0} + \beta_{j1}\mathbf{x} + \beta_{j2}\mathbf{z})}{\sigma_j}\Big]^2 - \frac{1}{2\tau_j^2}(\mathbf{x}^* - \mathbf{x})^2 - \frac{1}{2\lambda_j^2}\big(\mathbf{x} - \{\kappa_{j0} + \kappa_{j1}\mathbf{z}\}\big)^2 \\
&\quad = -\frac{1}{2}\left(\Big[\frac{\beta_{j1}^2}{\sigma_j^2} + \frac{1}{\tau_j^2} + \frac{1}{\lambda_j^2}\Big]\mathbf{x}^2 - 2\Big[\frac{\beta_{j1}}{\sigma_j^2}(\log \bar{Y} - \beta_{j0} - \beta_{j2}\mathbf{z}) + \frac{\mathbf{x}^*}{\tau_j^2} + \frac{(\kappa_{j0} + \kappa_{j1}\mathbf{z})}{\lambda_j^2}\Big]\mathbf{x}\right) \\
&\qquad - \frac{1}{2}\left(\frac{(\log \bar{Y} - \beta_{j0} - \beta_{j2}\mathbf{z})^2}{\sigma_j^2} + \frac{(\mathbf{x}^*)^2}{\tau_j^2} + \frac{(\kappa_{j0} + \kappa_{j1}\mathbf{z})^2}{\lambda_j^2}\right)
\end{aligned}$$

To complete the square, we introduce μ_x. This gives

$$\begin{aligned}
&-\frac{1}{2}\Big[\frac{\beta_{j1}^2}{\sigma_j^2} + \frac{1}{\tau_j^2} + \frac{1}{\lambda_j^2}\Big]\left(\mathbf{x}^2 - 2\mu_x\mathbf{x} + \mu_x^2\right) + \frac{1}{2}\Big[\frac{\beta_{j1}^2}{\sigma_j^2} + \frac{1}{\tau_j^2} + \frac{1}{\lambda_j^2}\Big]\mu_x^2 \\
&\qquad - \frac{1}{2}\left(\frac{(\log \bar{Y} - \beta_{j0} - \beta_{j2}\mathbf{z})^2}{\sigma_j^2} + \frac{(\mathbf{x}^*)^2}{\tau_j^2} + \frac{(\kappa_{j0} + \kappa_{j1}\mathbf{z})^2}{\lambda_j^2}\right)
\end{aligned}$$

where

$$\mu_x = \frac{\frac{\beta_{j1}}{\sigma_j^2}(\log \bar{Y} - \beta_{j0} - \beta_{j2}\mathbf{z}) + \frac{\mathbf{x}^*}{\tau_j^2} + \frac{(\kappa_{j0} + \kappa_{j1}\mathbf{z})}{\lambda_j^2}}{\frac{\beta_{j1}^2}{\sigma_j^2} + \frac{1}{\tau_j^2} + \frac{1}{\lambda_j^2}}$$

Therefore, one can say

$$\begin{aligned}
&\int f(\bar{Y}, \mathbf{x}^*, \mathbf{x} \mid \mathbf{z})\, d\mathbf{x} \\
&= \frac{1}{\bar{Y}\sigma_j\tau_j\lambda_j(2\pi)\sqrt{2\pi}} \underbrace{\int_{\mathbf{x}} \exp\left(-\frac{1}{2}\Big[\frac{\beta_{j1}^2}{\sigma_j^2} + \frac{1}{\tau_j^2} + \frac{1}{\lambda_j^2}\Big]\big(\mathbf{x} - \mu_x\big)^2\right) d\mathbf{x}}_{=\sqrt{2\pi\left[\frac{\beta_{j1}^2}{\sigma_j^2} + \frac{1}{\tau_j^2} + \frac{1}{\lambda_j^2}\right]^{-1}}} \\
&\quad \times \exp\left(\frac{1}{2}\left(\Big[\frac{\beta_{j1}^2}{\sigma_j^2} + \frac{1}{\tau_j^2} + \frac{1}{\lambda_j^2}\Big]\mu_x^2 - \frac{(\log \bar{Y} - \beta_{j0} - \beta_{j2}\mathbf{z})^2}{\sigma_j^2} - \frac{(\mathbf{x}^*)^2}{\tau_j^2} - \frac{(\kappa_{j0} + \kappa_{j1}\mathbf{z})^2}{\lambda_j^2}\right)\right)
\end{aligned}$$

In the exponent above, we can simplify as:

$$\frac{1}{2}\left(\Big[\frac{\beta_{j1}^2}{\sigma_j^2}+\frac{1}{\tau_j^2}+\frac{1}{\lambda_j^2}\Big]\mu_x^2-\frac{(\log\bar{Y}-\beta_{j0}-\beta_{j2}\mathbf{z})^2}{\sigma_j^2}-\frac{(\mathbf{x}^*)^2}{\tau_j^2}-\frac{(\kappa_{j0}+\kappa_{j1}\mathbf{z})^2}{\lambda_j^2}\right)$$
$$=\frac{1}{2}\Big(\frac{\beta_{j1}^2}{\sigma_j^2}+\frac{1}{\tau_j^2}+\frac{1}{\lambda_j^2}\Big)^{-1}\Bigg[-\frac{1}{\sigma_j^2}\Big(\frac{1}{\tau_j^2}+\frac{1}{\lambda_j^2}\Big)\Big[(\log\bar{Y}-\beta_{j0}-\beta_{j2}\mathbf{z})-\frac{\beta_{j1}\Big(\frac{\mathbf{x}^*}{\tau_j^2}+\frac{\kappa_{j0}+\kappa_{j1}\mathbf{z}}{\lambda_j^2}\Big)}{\frac{1}{\tau_j^2}+\frac{1}{\lambda_j^2}}\Big]^2$$
$$+\Bigg\{\frac{1}{\sigma_j^2\Big(\frac{1}{\tau_j^2}+\frac{1}{\lambda_j^2}\Big)}\Big[-\Big(\frac{\beta_1^2+\frac{\sigma_j 2}{\tau_j^2}+\frac{\sigma_j^2}{\lambda_j^2}}{\tau_j^2\lambda_j^2}\Big)\Big(\mathbf{x}^*-(\kappa_{j0}+\kappa_{j1}\mathbf{z})\Big)^2\Big]\Bigg\}\Bigg]$$

Accordingly,

$$\int f(\bar{Y},\mathbf{x}^*,\mathbf{x}\mid\mathbf{z})\,d\mathbf{x}$$
$$=\frac{1}{\bar{Y}(2\pi)\sigma_j\tau_j\lambda_j}\Big(\frac{\beta_{j1}^2}{\sigma_j^2}+\frac{1}{\tau_j^2}+\frac{1}{\lambda_j^2}\Big)^{-1/2}$$
$$\times\exp\Bigg(-\frac{1}{2}\Big(\frac{\beta_{j1}^2}{\sigma_j^2}+\frac{1}{\tau_j^2}+\frac{1}{\lambda_j^2}\Big)^{-1}\frac{1}{\sigma_j^2}\Big(\frac{1}{\tau_j^2}+\frac{1}{\lambda_j^2}\Big)\Big[(\log\bar{Y}-\beta_{j0}-\beta_{j2}\mathbf{z})-\frac{\beta_{j1}\Big(\frac{\mathbf{x}^*}{\tau_j^2}+\frac{\kappa_{j0}+\kappa_{j1}\mathbf{z}}{\lambda_j^2}\Big)}{\frac{1}{\tau_j^2}+\frac{1}{\lambda_j^2}}\Big]^2\Bigg)$$
$$\times\exp\Bigg(-\frac{1}{2}\Big(\frac{\beta_{j1}^2}{\sigma_j^2}+\frac{1}{\tau_j^2}+\frac{1}{\lambda_j^2}\Big)^{-1}\frac{1}{\sigma_j^2\Big(\frac{1}{\tau_j^2}+\frac{1}{\lambda_j^2}\Big)}\Big[\Big(\frac{\beta_1^2+\frac{\sigma_j 2}{\tau_j^2}+\frac{\sigma_j^2}{\lambda_j^2}}{\tau_j^2\lambda_j^2}\Big)\Big(\mathbf{x}^*-(\kappa_{j0}+\kappa_{j1}\mathbf{z})\Big)^2\Big]\Bigg)$$
$$=\frac{1}{\bar{Y}(2\pi)\sigma_j\tau_j\lambda_j}\Big(\frac{\sigma_j^2\lambda_j^2\tau_j^2}{\beta_{j1}^2\tau_j^2\lambda_j^2+\sigma_j^2\lambda_j^2+\sigma_j^2\tau_j^2}\Big)^{1/2}$$
$$\times\exp\Bigg(-\frac{1}{2}\Big(\frac{\lambda_j^2+\tau_j^2}{\beta_{j1}^2\tau_j^2\lambda_j^2+\sigma_j^2\lambda_j^2+\sigma_j^2\tau_j^2}\Big)\Big[(\log\bar{Y}-\beta_{j0}-\beta_{j2}\mathbf{z})-\frac{\beta_{j1}\Big(\frac{\mathbf{x}^*}{\tau_j^2}+\frac{\kappa_{j0}+\kappa_{j1}\mathbf{z}}{\lambda_j^2}\Big)}{\frac{1}{\tau_j^2}+\frac{1}{\lambda_j^2}}\Big]^2\Bigg)$$
$$\times\exp\Bigg(-\frac{1}{2}\Big(\frac{1}{\tau_j^2+\lambda_j^2}\Big)\Big[\Big(\mathbf{x}^*-(\kappa_{j0}+\kappa_{j1}\mathbf{z})\Big)^2\Big]\Bigg)\ldots\ldots\ldots\ldots\ldots\ldots\text{ This is Solution I.}$$

and the resulting form above (Solution I.) should square with the incomplete model in Equation (4.20)

$$f(\bar{Y},\mathbf{x}^*\mid\mathbf{z})=\frac{1}{\bar{Y}(2\pi)\widehat{\sigma}_j\widehat{\lambda}_j}\times\exp\Big(-\frac{1}{2\widehat{\sigma}_j^2}\Big[(\log\bar{Y}-\widehat{\beta}_{j0}-\widehat{\beta}_{j2}\mathbf{z})-\widehat{\beta}_{j1}\mathbf{x}^*\Big]^2\Big)$$
$$\times\exp\Big(-\frac{1}{2\widehat{\lambda}_j^2}\Big[\mathbf{x}^*-(\widehat{\kappa}_{j0}+\widehat{\kappa}_{j1}\mathbf{z})\Big]^2\Big)$$

Thus, $\widehat{\lambda}_j^2=\lambda_j^2+\tau_j^2$, $\widehat{\kappa}_{j0}=\kappa_{j0}$, $\widehat{\kappa}_{j1}=\kappa_{j1}$. If τ_j^2 is determined, we can obtain the parameters of the unobservable complete model - $\beta_{j0},\beta_{j1},\beta_{j2},\sigma_j,\lambda_j,\kappa_{j0},\kappa_{j1}$ - based on the estimates of $\widehat{\beta}_{j0},\widehat{\beta}_{j1},\widehat{\beta}_{j2},\widehat{\sigma}_j,\widehat{\lambda}_j,\widehat{\kappa}_{j0},\widehat{\kappa}_{j1}$ in the incomplete model we have.

In summary, the error-free parameters for the complete model can be obtained from Gustafson's system of equations listed as below.

$$\lambda_j^2 = \widehat{\lambda}_j^2 - \tau_j^2$$

$$\kappa_{j0} = \widehat{\kappa}_{j0}$$

$$\kappa_{j1} = \widehat{\kappa}_{j1}$$

$$\beta_{j1} = \frac{\widehat{\beta}_{j1}\widehat{\lambda}_j^2}{\widehat{\lambda}_j^2 - \tau_j^2}$$

$$\beta_{j0} = \widehat{\beta}_{j0} - \frac{\widehat{\beta}_{j1}\widehat{\kappa}_{j0}\tau_j^2}{\widehat{\lambda}_j^2 - \tau_j^2}$$

$$\beta_{j2} = \widehat{\beta}_{j2} - \frac{\widehat{\beta}_{j1}\widehat{\kappa}_{j1}\tau_j^2}{\widehat{\lambda}_j^2 - \tau_j^2}$$

$$\sigma_j^2 = \widehat{\sigma}_j^2 - \frac{\beta_{j1}^2\tau_j^2(\widehat{\lambda}_j^2 - \tau_j^2)}{\widehat{\lambda}_j^2}$$

D.3 Distribution Choices in Chapter 4

Modeling target	**Outcome (count/amount)**	**Covariate.1 (binary)**	**Covariate.2 (cont.)**
Data: $N, \mathbf{z}^F, \mathbf{x}^F$ (alternative)	Negative Binomial: $N \sim \mathbf{NB}(\xi,\ \psi)$ $\Rightarrow N \sim \mathbf{Poi}(\cdot)$?	Bernoulli: $\mathbf{z}^F \sim \mathbf{Bern}(\pi^F)$ $\Rightarrow$ Logistic regression ?	Gaussian: $\mathbf{x}^F \sim \mathbf{N}(E[\mathbf{x}^F], \lambda^{2F})$ $\Rightarrow \mathbf{x}^F \sim \mathbf{t}(\cdot)$?, $\mathbf{x}^F \sim \mathbf{Laplace}(\cdot)$?
Param: ξ, π^F (alternative)	GLM: $\xi = E[N] = \exp(\mathbf{X}^F\boldsymbol{\beta}^F)$ $\Rightarrow$ Poisson/Gamma regression ?	Beta: $\pi^F \sim \mathbf{Beta}(g_0^F, h_0^F)$ $\Rightarrow \pi^F \sim \mathbf{N}(\cdot)$?; $0 \le \pi^F \le 1$	
Scale param: ψ, λ^{2F} (alternative)	Gamma: $\psi \sim \mathbf{Ga}(u_0^F/2, v_0^F/2)$ $\Rightarrow \psi \sim \mathbf{InvGa}(\cdot)$?, $\psi \sim \mathbf{LogN}(\cdot)$?		Inverse Gamma: $\lambda^{2F} \sim \mathbf{InvGa}(c_0^F/2, d_0^F/2)$ $\Rightarrow \lambda^{2F} \sim \mathbf{ScaledInv}\chi^\mathbf{2}(\cdot)$?
Data: $\bar{Y}, \mathbf{z}^S, \mathbf{x}^S$ (alternative)	Lognormal: $\bar{Y} \sim \mathbf{LogN}(\mu,\ \sigma^2)$ $\Rightarrow \bar{Y} \sim \mathbf{Pareto}(\cdot)$?, $\bar{Y} \sim \mathbf{Burr}(\cdot)$?	Bernoulli: $\mathbf{z}^S \sim \mathbf{Bern}(\pi^S)$ $\Rightarrow$ Logistic regression ?	Gaussian: $\mathbf{x}^S \sim \mathbf{N}(E[\mathbf{x}^S], \lambda^{2S})$ $\Rightarrow \mathbf{x}^S \sim \mathbf{t}(\cdot)$?, $\mathbf{x}^S \sim \mathbf{Laplace}(\cdot)$?
Param: μ, π^S (alternative)	GLM: $\mu = \mathbf{X}^S\boldsymbol{\beta}^S = \ln E[\bar{Y}] - 0.5\sigma^2$ $\Rightarrow$ Gaussian/Laplace regression ?	Beta: $\pi^S \sim \mathbf{Beta}(g_0^S, h_0^S)$ $\Rightarrow \pi^S \sim \mathbf{N}(\cdot)$?; $0 \le \pi^S \le 1$	
Scale param: σ^2, λ^{2S} (alternative)	Inverse Gamma: $\sigma^2 \sim \mathbf{InvGa}(u_0^S/2, v_0^S/2)$ $\Rightarrow \sigma^2 \sim \mathbf{ScaledInv}\chi^\mathbf{2}(\cdot)$?		Inverse Gamma: $\lambda^{2S} \sim \mathbf{InvGa}(c_0^S/2, d_0^S/2)$ $\Rightarrow \lambda^{2S} \sim \mathbf{ScaledInv}\chi^\mathbf{2}(\cdot)$?

Table D.1: Distribution choices/alternatives for outcome N, Y and covariates $\mathbf{X}^F, \mathbf{X}^S$ across data, parameter models. The selection of these distributions further informs the specification of hyperparameter models.

Here we explain some conventions about distribution choices. Table D.1 provides a structured comparison of the distributional choices for both outcome variables (e.g., lognormal and Poisson) and covariates (Bernoulli and Gaussian). Additionally, it highlights alternative distributional options, offering insight into the flexibility of the model specifications. A proper guide for choosing

distributions can be found in Gelman and Meng 2004; Gelman and Hill 2007; Gelman and Carlin 2013; Gelman and Hwang 2014 and the references therein.

(i) **Data:** $N, \mathbf{z}^F, \mathbf{x}^F$

- **Outcome:** claim count N

  → our choice: $N \sim \mathbf{NB}(\xi,\ \psi)$; i.e. Negative Binomial

  → possible alternative: $N \sim \mathbf{Poi}(\cdot)$; i.e. Poisson?

  ◦ *mean* $\xi = E[N] = \exp(\mathbf{X}^F\boldsymbol{\beta}^F)$; i.e. Binomial regression

  ◦ *dispersion* $\psi \sim \mathbf{Ga}(\frac{u_0^F}{2}, \frac{v_0^F}{2})$; i.e. Gamma

  : The negative binomial (**NB**) is used for our count data as it accounts for overdispersion, where variance exceeds the mean. While the Poisson (**Poi**) is a simpler alternative, its equal mean-variance assumption is violated in our dataset, making it unsuitable.

  : To model its parameters - mean ξ and dispersion ψ -, we use Binomial regression for the mean and the Gamma (**Ga**) for the dispersion. The Binomial regression captures the relationship between covariates $\mathbf{X}^F$ and the mean count $E[N]$, while the Gamma density is used to capture the positive dispersion values and skewness. Alternatives like Gamma regression for the mean and the Inverse Gamma (**InvGa**) or Log-normal (**LogN**) for the dispersion are also possible, but they are less suitable here due to the complexity they introduce (lack of conjugacy with **NB**, multiplicative nature, and computational challenges, etc).

- **Covariate.1:** binary $\mathbf{z}^F$

  → our choice: $\mathbf{z}^F \sim \mathbf{Bern}(\pi^F)$; i.e. Bernoulli

  → possible alternative: Logistic regression ?

  ◦ *probability* $\pi^F \sim \mathbf{Beta}(g_0^F, h_0^F)$; i.e. Beta

  : The Bernoulli (**Bern**) is used for our binary covariate as it accounts for only two possible values (0/1). While logistic regression could be an alternative, it introduces additional complexity by requiring parameter estimation for the link function.

  : To model its parameters - probability π^F -, we use the beta density (**Beta**) because the range of values is defined between 0 and 1, and the density serves as a conjugate prior for **Bern**. The truncated normal (**N**) or non-informative prior such as uniform or Jeffeys prior, etc. can be alternative

choices, but they come with trade-offs in terms of interpretability, flexibility, and computational complexity.

- **Covariate.2:** continuous $\mathbf{x}^F$

  → our choice: $\mathbf{x}^F \sim \mathbf{N}(E[\mathbf{x}^F], \lambda^{2F})$; i.e. Gaussian

  → possible alternative: $\mathbf{x}^F \sim \mathbf{t}(\cdot)$; i.e. Student's t ?, $\mathbf{x}^F \sim \mathbf{Laplace}(\cdot)$?

  ◦ *location* $= E[\mathbf{x}^F]$

  ◦ *scale* $\lambda^{2F} \sim \mathbf{InvGa}(c_0^F/2, d_0^F/2)$ i.e. Inverse Gamma

  : The Gaussian (**N**) is employed for our continuous covariate, as it effectively models data that tends to cluster around a central mean, while accommodating both positive and negative values. While the Student's t (**t**) or Laplace distributions could serve as alternatives, they are associated with specific characteristics, such as heavier tails or sharper peaks, which may not be suitable for our data.

  : To model its parameters - scale λ^{2F} -, we use the Inverse Gamma density (**InvGa**) because it serves as a conjugate prior for the Gaussian (**N**), while also effectively capturing a wide range of possible values for the scale due to its heavy tail. The scaled inverse Chi-square (**ScaledInvχ^2**) can be an alternative choice, but it may be more suitable when there are prior beliefs about the degrees of freedom.

(ii) **Data:** $\bar{Y}, \mathbf{z}^S, \mathbf{x}^S$

- **Outcome:** AVG claim amount $\bar{Y}$

  → our choice: $\bar{Y} \sim \mathbf{LogN}(\mu,\ \sigma^2)$; i.e. Log-normal

  → possible alternative: $\bar{Y} \sim \mathbf{Pareto}(\cdot)$?, $\bar{Y} \sim \mathbf{Burr}(\cdot)$?

  ◦ *log-space location* $\mu = \mathbf{X}^F\boldsymbol{\beta}^F$; i.e. Log-normal regression

  ◦ *log-space scale* $\sigma^2 \sim \mathbf{InvGa}(u_0^S/2, v_0^S/2)$; i.e. Inverse Gamma

  : The log-normal (**LogN**) is used for our claim amount data as it models strictly positive, right-skewed data with a long tail, capturing the presence of many small claims and few large ones. Alternative distributions include Pareto or Burr, but they are commonly used for heavy-tailed claim data.

  : To model its parameters - log-space location μ and log-space scale σ^2 -, we use Log-normal regression for the log-space location and the Inverse Gamma for the log-space scale. The Log-normal regression captures the relationship between covariates $\mathbf{X}^S$ and the expected AVG amount $E[\bar{Y}]$,

while the Inverse Gamma density is used to serve as a conjugate prior for the Log-normal. Alternatives like Laplace regression for the log-space location and the scaled inverse Chi-square (**ScaledInvχ^2**) for the log-space scale can be considered, but they are less suitable here due to the complexity they introduce (lack of conjugacy with **LogN**, and additional parameters like the degrees of freedom, etc).

- **Covariate.1:** binary $\mathbf{z}^S$

  → our choice: $\mathbf{z}^S \sim \mathbf{Bern}(\pi^S)$; i.e. Bernoulli

  → possible alternative: Logistic regression ?

  - *probability* $\pi^S \sim \mathbf{Beta}(g_0^S, h_0^S)$; i.e. Beta

    : The Bernoulli (**Bern**) is used for our binary covariate as it accounts for only two possible values (0/1). While logistic regression could be an alternative, it introduces additional complexity by requiring parameter estimation for the link function.

    : To model its parameters - probability π^S -, we use the beta density (**Beta**) because the range of values is defined between 0 and 1, and the density serves as a conjugate prior for **Bern**. The truncated normal (**N**) or non-informative prior such as uniform or Jeffeys prior, etc. can be alternative choices, but they come with trade-offs in terms of interpretability, flexibility, and computational complexity.

- **Covariate.2:** continuous $\mathbf{x}^S$

  → our choice: $\mathbf{x}^S \sim \mathbf{N}(E[\mathbf{x}^S], \lambda^{2S})$; i.e. Gaussian

  → possible alternative: $\mathbf{x}^S \sim \mathbf{t}(\cdot)$; i.e. Student's t ?, $\mathbf{x}^S \sim \mathbf{Laplace}(\cdot)$?

  - *location* $= E[\mathbf{x}^S]$
  - *scale* $\lambda^{2S} \sim \mathbf{InvGa}(c_0^S/2, d_0^S/2)$; i.e. Inverse Gamma

    : The Gaussian (**N**) is employed for our continuous covariate, as it effectively models data that tends to cluster around a central mean, while accommodating both positive and negative values. While the Student's t (**t**) or Laplace distributions could serve as alternatives, they are associated with specific characteristics, such as heavier tails or sharper peaks, which may not be suitable for our data.

    : To model its parameters - scale λ^{2S} -, we use the Inverse Gamma density (**InvGa**) because it serves as a conjugate prior for the Gaussian (**N**), while also effectively capturing a wide range of possible values for the scale due

to its heavy tail. The scaled inverse Chi-square (**ScaledInvχ^2**) can be an alternative choice, but it may be more suitable when there are prior beliefs about the degrees of freedom.

As outlined in the main text, we examines two outcome variables - claim count N, and AVG claim amount $\bar{Y}$ - alongside two types of covariates: a binary variable **z** and a continuous variable **x**. This selection was made to streamline the analysis of the Gustafson correction technique, as incorporating additional types of covariates would significantly increase the complexity of its implementation. Given the computational and resource constraints of this PhD program, a more intricate model was not feasible at this stage. However, expanding the model to incorporate a broader range of covariates should be considered in future research.

Appendix E

For Chapter 5

E.1 Data Model Development

E.1.1 Discrete outcome data model with MAR

Prior to the outcome parameter estimation, the missing covariates should be imputed first to obtain the complete covariate model beforehand. In this study, if the binary covariate z_h is the only covariate with missingness, we develop the imputation model to impute the binary covariate z_h, taking the following steps below, then update $\boldsymbol{\beta}, \sigma^2, \xi, \tilde{\boldsymbol{\beta}}$ based on the posterior sampling detailed in Algorithm (E.2.3). The imputation model for z_h is approximated by the joint:

$$f(z_h|S_h, x_h, \boldsymbol{\beta}_j, \sigma_j, \xi_j, \tilde{\boldsymbol{\beta}}_j, \pi_j) \propto f(S_h,\ z_h|x_h, \boldsymbol{\beta}_j, \sigma_j, \xi_j, \tilde{\boldsymbol{\beta}}_j, \pi_j)$$

where

$$\begin{aligned} f(S_h,\ z_h|x_h, \boldsymbol{\beta}_j, \sigma_j^2, \xi_j, \tilde{\boldsymbol{\beta}}_j, \pi_j) &= f(S_h|z_h, x_h, \boldsymbol{\beta}_j, \sigma_j^2, \xi_j, \tilde{\boldsymbol{\beta}}_j) \cdot f_{Bern}(z_h|\pi_j) \\ &= \delta(\mathbf{X}_h^T\tilde{\boldsymbol{\beta}}_j)\ \mathbb{1}_{(S_h=0)} \cdot \pi_j^{z_h}\left(1-\pi_j\right)^{1-z_h} + \left[1-\delta(\mathbf{X}_h^T\tilde{\boldsymbol{\beta}}_j)\right]\frac{2}{\sigma_j S_h} \\ &\cdot \phi\left(\frac{\log S_h - \mathbf{X}^T\boldsymbol{\beta}_j}{\sigma_j}\right) \cdot \Phi\left(\xi_j\frac{\log S_h - \mathbf{X}^T\boldsymbol{\beta}_j}{\sigma_j}\right) \cdot \pi_j^{z_h}\left(1-\pi_j\right)^{1-z_h} \end{aligned}$$

which serves as the joint density that we can use to sample the imputation values. For example,

$$\begin{aligned} f_{Bern}(z_h = 1|S_h, x_h, \boldsymbol{\beta}_j, \sigma_j^2, \xi_j, \tilde{\boldsymbol{\beta}}_j, \pi_j) &\propto f(S_h,\ z_h = 1|x_h, \boldsymbol{\beta}_j, \sigma_j^2, \xi_j, \tilde{\boldsymbol{\beta}}_j, \pi_j) \\ &= \delta(\tilde{\beta}_{j0} + \tilde{\beta}_{j1} + \tilde{\beta}_{j2}x_h)\ \mathbb{1}_{(S_h=0)} \cdot \pi_j + \left[1-\delta(\tilde{\beta}_{j0} + \tilde{\beta}_{j1} + \tilde{\beta}_{j2}x_h)\right]\frac{2}{\sigma_j S_h} \\ &\cdot \phi\left(\frac{\log S_h - (\beta_{j0} + \beta_{j1} + \beta_{j2}x_h)}{\sigma_j}\right) \cdot \Phi\left(\xi_j\frac{\log S_h - (\beta_{j0} + \beta_{j1} + \beta_{j2}x_h)}{\sigma_j}\right)\pi_j \end{aligned}$$

$$
\begin{aligned}
f_{Bern}(z_h = 0|S_h, x_h, \boldsymbol{\beta}_j, \sigma_j^2, \xi_j, \tilde{\boldsymbol{\beta}}_j, \pi_j) &\propto f(S_h,\ z_h = 0|x_h, \boldsymbol{\beta}_j, \sigma_j^2, \xi_j, \tilde{\boldsymbol{\beta}}_j, \pi_j) \\
&= \delta(\tilde{\beta}_{j0} + \tilde{\beta}_{j2}x_h)\ \mathbb{1}_{(S_h=0)} \cdot (1-\pi_j) + \left[1 - \delta(\tilde{\beta}_{j0} + \tilde{\beta}_{j2}x_h)\right] \frac{2}{\sigma_j S_h} \\
&\cdot \phi\left(\frac{\log S_h - (\beta_{j0} + \beta_{j2}x_h)}{\sigma_j}\right) \cdot \Phi\left(\xi_j \frac{\log S_h - (\beta_{j0} + \beta_{j2}x_h)}{\sigma_j}\right) \cdot (1-\pi_j)
\end{aligned}
$$

Then, we can impute z_h with the values sampled from $\textbf{Bernoulli}(\pi_{\mathbf{z}}^*)$ where

$$
\pi_{\mathbf{z}}^* = \frac{f(S_h,\ z_h = 1|x_h, \boldsymbol{\beta}_j, \sigma_j^2, \xi_j, \tilde{\boldsymbol{\beta}}_j, \pi_j)}{f(S_h,\ z_h = 1|x_h, \boldsymbol{\beta}_j, \sigma_j^2, \xi_j, \tilde{\boldsymbol{\beta}}_j, \pi_j) + f(S_h,\ z_h = 0|x_h, \boldsymbol{\beta}_j, \sigma_j^2, \xi_j, \tilde{\boldsymbol{\beta}}_j, \pi_j)}
$$

Note that in R, the computation can be difficult when the numerator is too small. We suggest the following tricks.

$$
\begin{aligned}
p_1 &= f(S_h,\ z_h = 1|x_h, \boldsymbol{\beta}_j, \sigma_j^2, \xi_j, \tilde{\boldsymbol{\beta}}_j, \pi_j) \\
p_0 &= f(S_h,\ z_h = 0|x_h, \boldsymbol{\beta}_j, \sigma_j^2, \xi_j, \tilde{\boldsymbol{\beta}}_j, \pi_j) \\
\pi_{\mathbf{z}}^* &= \frac{e^{log(p_1)}}{e^{log(p_1)} + e^{log(p_0)}} \cdot \frac{e^{-log(p_1)}}{e^{-log(p_1)}} = \frac{1}{1 + e^{log(p_0) - log(p_1)}}
\end{aligned}
$$

Finally, the outcome model that is required to compute the parameter $\boldsymbol{\theta} = \{\boldsymbol{\beta}_j, \sigma_j^2, \xi_j, \tilde{\boldsymbol{\beta}}_j\}$ in Metropolis-Hastings in Algorithm (E.2.3) is obtained by summing the joint of S_h and z_h (marginalize) out the MAR covariate z_h, shown in Equation (5.11), as below.

$$
\begin{aligned}
f(S_h|x_h, \boldsymbol{\beta}_j, \sigma_j^2, \xi_j, \tilde{\boldsymbol{\beta}}_j, \pi_j) &= \sum_{z_h=0}^{1} f(S_h, z_h|x_h, \boldsymbol{\beta}_j, \sigma_j^2, \xi_j, \tilde{\boldsymbol{\beta}}_j, \pi_j) \\
&= f(S_h,\ z_h = 1|x_h, \boldsymbol{\beta}_j, \sigma_j^2, \xi_j, \tilde{\boldsymbol{\beta}}_j, \pi_j) + f(S_h,\ z_h = 0|x_h, \boldsymbol{\beta}_j, \sigma_j^2, \xi_j, \tilde{\boldsymbol{\beta}}_j, \pi_j)
\end{aligned}
$$

E.1.2 Parameter-free covariate data model with MAR

The parameter-free distributions $f_0(Y|\mathbf{X})$ and $f_0(\mathbf{X})$ as data models for continuous clusters are needed to calculate the probabilities of cluster membership and for the post-processing calculations for prediction in the DPM. However, when MAR covariates are present, it gives extra complexity in specifying distribution to integrate out the parameters. Recall the integrals we are attempting to find are the following:

$$
f_0(\mathbf{X}) = \int f(\mathbf{X}|\boldsymbol{w})\ dG_0(\boldsymbol{w}) = \int f(\mathbf{X}|\boldsymbol{w})\ p(\boldsymbol{w})\ d\boldsymbol{w}
$$

If binary covariate $\mathbf{z}$ is missing, then we will need to replace the distribution $f(\mathbf{X}|\boldsymbol{w})$ with the continuous distribution (Gaussian) of $\mathbf{x}$, which is $f(\mathbf{x}|\mu_j, \lambda_j^2)$. The derivation of the parameter-free

distribution $f_0(\mathbf{z})$ and $f_0(\mathbf{x})$ for the continuous cluster is as below.

$$
\begin{aligned}
&f_0(\mathbf{z}) \\
&\quad = \int f(\mathbf{z}|\pi)\, p(\pi)\, d\mu\, d\pi \\
&\quad = \int \pi^{\mathbf{z}} (1-\pi)^{1-\mathbf{z}} \frac{1}{\boldsymbol{Beta}(g_0, h_0)} \pi^{(g_0-1)} (1-\pi)^{(h_0-1)} d\pi \\
&\quad = \frac{1}{\boldsymbol{Beta}(g_0, h_0)} \int \pi^{(\mathbf{z}+g_0-1)} (1-\pi)^{(1-\mathbf{z}+h_0-1)} d\pi \\
&\quad = \frac{\boldsymbol{Beta}(\mathbf{z}+g_0,\ 1-\mathbf{z}+h_0)}{\boldsymbol{Beta}(g_0, h_0)} \cdot \underbrace{\int \frac{\pi^{(\mathbf{z}+g_0-1)} (1-\pi)^{(1-\mathbf{z}+h_0-1)}}{\boldsymbol{Beta}(\mathbf{z}+g_0,\ 1-\mathbf{z}+h_0)}\, d\pi}_{=1,\text{ beta distribution}}
\end{aligned}
$$

$$
\begin{aligned}
&f_0(\mathbf{x}) \\
&\quad = \iint f(\mathbf{x}|\mu, \lambda^2) p(\mu|\lambda^2) p(\lambda^2)\, d\mu\, d\lambda^2 \\
&\quad = \iint \frac{1}{\sqrt{2\pi\lambda^2}} \exp\left\{ -\frac{1}{2\lambda^2} (\mathbf{x}-\mu)^2 \right\} \times \frac{1}{\sqrt{2\pi\lambda^2}} \exp\left\{ -\frac{1}{2\lambda^2} (\mu-\mu_0)^2 \right\} \\
&\qquad\quad \times \frac{d_0^{c_0}}{\Gamma(c_0)} \left(\lambda^2\right)^{-c_0-1} e^{-d_0/\lambda^2}\, d\mu\, d\lambda^2 \\
&\quad = \frac{d_0^{c_0}}{2\pi\Gamma(c_0)} \iint \left(\lambda^2\right)^{-c_0-2} \exp\left\{ -\frac{1}{2\lambda^2} (\mathbf{x}-\mu)^2 - \frac{1}{2\lambda^2} (\mu-\mu_0)^2 - \frac{d_0}{\lambda^2} \right\} d\mu\, d\lambda^2
\end{aligned}
$$

The first step is to integrate with respect to μ. First, we'll simplify the exponent.

$$
\begin{aligned}
&-\frac{1}{2\lambda^2} (\mathbf{x}-\mu)^2 - \frac{1}{2\lambda^2} (\mu-\mu_0)^2 - \frac{d_0}{\lambda^2} \\
&\qquad = -\frac{1}{2\lambda^2} \left[\mathbf{x}^2 - 2\mathbf{x}\mu + \mu^2 + \mu^2 - 2\mu_0\mu + \mu_0^2 \right] - \frac{d_0}{\lambda^2} \\
&\qquad = -\frac{1}{2\lambda^2} \left[2\mu^2 - 2(\mathbf{x}+\mu_0)\mu \right] - \frac{1}{2\lambda^2} \left[\mathbf{x}^2 + \mu_0^2 \right] - \frac{d_0}{\lambda^2} \\
&\qquad = -\frac{2}{2\lambda^2} \left[\mu^2 - (\mathbf{x}+\mu_0)\mu + \frac{(\mathbf{x}+\mu_0)^2}{4} \right] + \frac{1}{\lambda^2} \left(\frac{(\mathbf{x}+\mu_0)^2}{4} \right) \\
&\qquad\qquad - \frac{\mathbf{x}^2 + \mu_0^2}{2\lambda^2} - \frac{d_0}{\lambda^2} \\
&\qquad = -\frac{1}{2(\lambda^2/2)} \left(\mu - \frac{\mathbf{x}+\mu_0}{2} \right)^2 + \frac{(\mathbf{x}+\mu_0)^2}{4\lambda^2} - \frac{\mathbf{x}^2 + \mu_0^2}{2\lambda^2} - \frac{d_0}{\lambda^2}
\end{aligned}
$$

The integrand will have the kernel of a normal distribution for μ with mean $\dfrac{\mathbf{x}+\mu_0}{2}$ and variance $\dfrac{\lambda^2}{2}$.

$$
\begin{aligned}
& f_0(\mathbf{x}) \\
&= \frac{d_0^{c_0}}{2\pi\Gamma(c_0)} \int \underbrace{\sqrt{2\pi(\lambda^2/2)}}_{\text{term from } \mu \text{ integral}} \times \left(\lambda^2\right)^{-c_0-2} \times \exp\left\{ \frac{(\mathbf{x}+\mu_0)^2}{4\lambda^2} - \frac{\mathbf{x}^2+\mu_0^2}{2\lambda^2} - \frac{d_0}{\lambda^2} \right\} \, d\lambda^2 \\
&= \frac{d_0^{c_0}}{2\sqrt{\pi}\Gamma(c_0)} \int \left(\lambda^2\right)^{-c_0-3/2} \exp\left\{ -\frac{1}{\lambda^2} \left(-\frac{\mathbf{x}^2+2\mathbf{x}\mu_0+\mu_0^2}{4} + \frac{\mathbf{x}^2+\mu_0^2}{2} + d_0 \right) \right\} \, d\lambda^2 \\
&= \frac{d_0^{c_0}}{2\sqrt{\pi}\Gamma(c_0)} \int \left(\lambda^2\right)^{-c_0-1/2-1} \exp\left\{ -\frac{1}{\lambda^2} \left(\frac{(\mathbf{x}^2-\mu_0)^2}{4} + d_0 \right) \right\} \, d\lambda^2
\end{aligned}
$$

The integrand is the kernel of an inverse gamma distribution with shape parameter $c_0 + \dfrac{1}{2}$ and scale parameter $\dfrac{(\mathbf{x}^2-\mu_0)^2}{4} + d_0$.

$$
f_0(\mathbf{x}) = \frac{d_0^{c_0}}{2\sqrt{\pi}\Gamma(c_0)} \times \Gamma(c_0+1/2) \left(\frac{(\mathbf{x}^2-\mu_0)^2}{4} + d_0 \right)^{-c_0-1/2}
$$

As shown above, a closed-form expression can be determined, but it is not always the case since it can be extremely complicated. To simplify, we instead might have to consider a Monte Carlo integral.

E.2 Parameter Model Development

E.2.1 Derivation of the posterior: precision α

The parameter model (posterior) of the precision term α is defined as

$$p(\alpha|J) \propto p_0(\alpha) \cdot \alpha^{J-1} \cdot (\alpha + n) \cdot \boldsymbol{Beta}(\alpha + 1, n)$$
$$p(\alpha|J, \eta, \gamma_0, \psi_0) \propto \pi_\eta \boldsymbol{Ga}(\gamma_0 + J,\ \psi_0 - log(\eta)) + (1 - \pi_\eta)\boldsymbol{Ga}(\gamma_0 + J - 1,\ \psi_0 - log(\eta))$$

To derive this, we first consider the distribution of the number of clusters given the precision parameter: $p(J|\alpha)$. Consider a trivial example where we want to determine the number of clusters that $n = 5$ observations fall into. One possible arrangement would be that observations 1, 2, and 5 form new clusters, while observations 3 and 4 join an existing cluster. (note, the order is important):

- observation 1 forms a new cluster, probability $= \dfrac{\alpha}{\alpha}$
- observation 2 forms a new cluster, probability $= \dfrac{\alpha}{\alpha + 1}$
- observation 3 enters into an existing cluster, probability $= \dfrac{2}{\alpha + 2}$
- observation 4 enters into an existing cluster, probability $= \dfrac{3}{\alpha + 3}$
- observation 5 forms a new cluster, probability $= \dfrac{\alpha}{\alpha + 4}$

In this example, we have $J = 3$ clusters. We want to find the probability of this arrangement. The probability is the following:

$$\left(\frac{\alpha}{\alpha}\right)\left(\frac{\alpha}{\alpha+1}\right)\left(\frac{2}{\alpha+2}\right)\left(\frac{3}{\alpha+3}\right)\left(\frac{\alpha}{\alpha+4}\right) \propto \frac{\alpha^3}{\alpha(\alpha+1)(\alpha+2)(\alpha+3)(\alpha+4)}$$
$$= \alpha^3 \frac{\Gamma(\alpha)}{\Gamma(\alpha+5)}$$

Hence the probability of observing J clusters amongst a sample size of n is given by

$$p(J|\alpha) \propto \alpha^J \frac{\Gamma(\alpha)}{\Gamma(\alpha+n)}$$

This is also considered the likelihood function. The posterior on α is proportional to the likelihood times the prior, $p_0(\alpha)$.

$$p(\alpha|J) \propto p(J|\alpha)p_0(\alpha)$$
$$\propto \alpha^J \frac{\Gamma(\alpha)}{\Gamma(\alpha+n)} p_0(\alpha)$$

The beta function $\boldsymbol{Beta}(x, y)$ is defined as the following:

$$\boldsymbol{Beta}(x, y) = \frac{\Gamma(x)\Gamma(y)}{\Gamma(x+y)}$$

We can find the beta function of $\alpha + 1$ and n as follows:

$$\begin{aligned}
\boldsymbol{Beta}(\alpha+1, n) &= \frac{\Gamma(\alpha+1)\Gamma(n)}{\Gamma(\alpha+1+n)} \\
&\propto \frac{\alpha\Gamma(\alpha)}{(\alpha+n)\Gamma(\alpha+n)} \\
\frac{\Gamma(\alpha)}{\Gamma(\alpha+n)} &\propto \boldsymbol{Beta}(\alpha+1, n)\frac{\alpha+n}{\alpha}
\end{aligned}$$

Thus the posterior simplifies to the following:

$$\begin{aligned}
p(\alpha|J) \;&\propto\; \alpha^J \cdot \boldsymbol{Beta}(\alpha+1, n) \cdot \frac{\alpha+n}{\alpha} \cdot p_0(\alpha) \\
&\propto\; p_0(\alpha) \cdot \alpha^{J-1} \cdot (\alpha+n) \cdot \boldsymbol{Beta}(\alpha+1, n)
\end{aligned}$$

Now, under the $\boldsymbol{Ga}(\gamma_0, \psi_0)$ prior for α, substituting $p_0(\alpha)$ with $\boldsymbol{Ga}(\gamma_0, \psi_0)$, then

$$\begin{aligned}
p(\alpha|J, \eta, \gamma_0, \psi_0) \;&\propto \alpha^{\gamma_0+j-2} \cdot (\alpha+n) \cdot e^{-\alpha(\psi_0 - log(\eta))} \\
&\propto \pi_\eta \boldsymbol{Ga}(\gamma_0 + J,\; \psi_0 - log(\eta)) + (1-\pi_\eta)\boldsymbol{Ga}(\gamma_0 + J - 1,\; \psi_0 - log(\eta))
\end{aligned}$$

E.2.2 Prior kernel for outcome, covariates, and precision

$$\begin{aligned}
p_0(\boldsymbol{\beta}_j|\boldsymbol{\beta}_0,\; \Sigma_0) :\; & \boldsymbol{MVN}(\boldsymbol{\beta}_0,\; \sigma_j^2\Sigma_{\beta_0})^* \;\propto\; e^{-\frac{1}{2\sigma_j^2}\{(\boldsymbol{\beta}_j - \boldsymbol{\beta}_0)^T \Sigma_{\beta_0}^{-1}(\boldsymbol{\beta}_j - \boldsymbol{\beta}_0)\}}, \\
p_0(\sigma_j^2|u_0,\; v_0) : & \boldsymbol{InvGa}(u_0,\; v_0) \;\propto\; (\sigma_j^2)^{-(u_0+1)} \cdot e^{-v_0/\sigma_j^2} \\
p_0(\xi_j|\nu_0) :\; & \boldsymbol{T}(\nu_0) \;\propto\; \left(\frac{\xi_j^2}{\nu_0} + 1\right)^{-(\nu_0+1)/2}, \\
p_0(\tilde{\boldsymbol{\beta}}_j|\tilde{\boldsymbol{\beta}}_0,\; \tilde{\Sigma}_0) : & \boldsymbol{MVN}(\tilde{\boldsymbol{\beta}}_0,\; \tilde{\Sigma}_0)^* \;\propto\; e^{\{(\tilde{\boldsymbol{\beta}}_j - \tilde{\boldsymbol{\beta}}_0)^T \tilde{\Sigma}_0^{-1}(\tilde{\boldsymbol{\beta}}_j - \tilde{\boldsymbol{\beta}}_0)\}} \\
p_0(\pi_j|g_0,\; h_0) :\; & \boldsymbol{Beta}(g_0,\; h_0) \;\propto\; \pi_j^{(g_0-1)} \cdot (1-\pi_j)^{(h_0-1)}, \\
p_0(\mu_j|\mu_0,\; \lambda_j^2) : & \boldsymbol{N}(\mu_0,\; \lambda_j^2) \;\propto\; e^{-\frac{1}{2}(\mu_j - \mu_0)^2/\lambda_j^2} \\
p_0(\lambda_j^2|c_0,\; d_0) :\; & \boldsymbol{InvGa}(c_0,\; d_0) \;\propto\; (\lambda_j^2)^{-(c_0+1)} \cdot e^{-d_0/\lambda_j^2}, \\
p_0(\alpha|\gamma_0,\; \psi_0) :\; & \boldsymbol{Ga}(\gamma_0,\; \psi_0) \;\propto\; \alpha^{(\gamma_0-1)} \cdot e^{-\alpha\cdot\psi_0}
\end{aligned}$$

* $\boldsymbol{\beta}_0, \Sigma_0 \sim$ Gamma regression, $\quad \tilde{\boldsymbol{\beta}}_0, \tilde{\Sigma}_0 \sim$ Logistic regression.

E.2.3 Posterior computation for outcome parameters

Algorithm (E.2.3) Posterior samples $\boldsymbol{\theta}_j^{(*)} = \{\boldsymbol{\beta}_j^{(*)}, \sigma_j^{2(*)}, \xi_j^{(*)}, \tilde{\boldsymbol{\beta}}_j^{(*)}\}$ by M.Hastings

Require: initialize $\boldsymbol{\theta}_j^{(old)}$: $\begin{cases} \boldsymbol{\beta}_j \sim \boldsymbol{MVN}(\boldsymbol{\beta}_0,\ \sigma_j^2 \Sigma_{\beta_0}) \\ \sigma_j^2 \sim \boldsymbol{IG}(u_0,\ v_0) \\ \xi_j \sim \boldsymbol{T}(\nu_0) \\ \tilde{\boldsymbol{\beta}}_j \sim \boldsymbol{MVN}(\tilde{\boldsymbol{\beta}}_0,\ \tilde{\Sigma}_{\beta_0}) \end{cases}$

1: **repeat**
2: **for** $j = 1, \cdots, J$ **do** ▷ Assume total J cluster memberships.
3: Sample $\boldsymbol{\theta}^{(new)}$ from the proposal densities $\boldsymbol{q}$: ▷ Choose priors as $\boldsymbol{q}$.
$\boldsymbol{\beta}_j^{(new)} \sim \boldsymbol{q}_\beta, \quad \sigma_j^{2(new)} \sim \boldsymbol{q}_{\sigma^2}, \quad \xi_j^{(new)} \sim \boldsymbol{q}_\xi, \quad \tilde{\boldsymbol{\beta}}_j^{(new)} \sim \boldsymbol{q}_{\tilde{\beta}}$
4: **for** $\boldsymbol{\theta}_j^{(new)} = \{\boldsymbol{\beta}_j^{(new)}, \sigma_j^{2(new)}, \xi_j^{(new)}, \tilde{\boldsymbol{\beta}}_j^{(new)}\}$ **do**
5: Compute the transition ratio, using the outcome models:

$$Ratio_\theta = \frac{\prod_{h=1}^{H} f(S_h|\mathbf{X},\ \boldsymbol{\theta}_j^{(new)})^1 \cdot p_0(\boldsymbol{\theta}_j^{(new)}) \cdot \boldsymbol{q}_\theta(\boldsymbol{\theta}_j^{(old)})}{\prod_{h=1}^{H} f(S_h|\mathbf{X},\ \boldsymbol{\theta}_j^{(old)})^1 \cdot p_0(\boldsymbol{\theta}_j^{(old)}) \cdot \boldsymbol{q}_\theta(\boldsymbol{\theta}_j^{(new)})}$$

Sample $U \sim \boldsymbol{Unif}(0, 1)$
6: **if** $U < Ratio_\theta$ **then** $\boldsymbol{\theta}_j^{(*)} = \boldsymbol{\theta}_j^{(new)}$ otherwise $\boldsymbol{\theta}_j^{(*)} = \boldsymbol{\theta}_j^{(old)}$
7: **end if**
8: **end for**
9: Record $\boldsymbol{\theta}_j^{(*)}$
10: **end for**
11: **until** M posterior samples $(\boldsymbol{\theta}_{j=1,\cdots,J}^{(*)})$ obtained. ▷ M is a sufficient sample size

[1] The outcome density in Line 5 is given by

$$f(S_h|\mathbf{X},\ \boldsymbol{\theta}_j) = \delta(\mathbf{X}_h^T \tilde{\boldsymbol{\beta}}_j)\ \mathbb{1}(S_h = 0) + \left[1 - \delta(\mathbf{X}_h^T \tilde{\boldsymbol{\beta}}_j)\right]\ f_{LSN}(S_h|\mathbf{X}_h, \boldsymbol{\theta}_j)$$

E.2.4 Posterior computation for covariates and precision

$p(\pi_j|g_0, h_0,\ \mathbf{z})$: $\boldsymbol{Beta}(g_{new},\ h_{new})$ $\quad$ $p(\mu_j|\mu_0,\ \lambda_j^2,\ \mathbf{x})$: $\boldsymbol{N}(\mu_{0\ new},\ \lambda_{j\ new}^2)$

$$\begin{cases} g_{new} = g_0 + \sum_{h=1}^{n_j} z_h \\ h_{new} = h_0 + n_j - \sum_{h=1}^{n_j} z_h \end{cases} \qquad \begin{cases} \mu_{0\ new} = (n_j \overline{\mathbf{x}} + \mu_0)/(n_j + 1) \\ \lambda_{j\ new}^2 = \lambda_j^2/(n_j + 1) \end{cases}$$

$p(\lambda_j^2|c_0, d_0,\ \mathbf{x})$: $\boldsymbol{InvGa}(c_{new},\ d_{new})$ $\quad$ $p(\alpha|\gamma_0, \psi_0, H, J, \eta, \pi_\eta)$: $\pi_\eta \boldsymbol{Ga}(\gamma_0 + J, \psi_0 - \ln(\eta))$

$$+ (1 - \pi_\eta)\boldsymbol{Ga}(\gamma_0 + J - 1, \psi_0 - \ln(\eta))$$

$$\begin{cases} c_{new} = c_0 + n_j/2 \\ d_{new} = d_0 + \frac{1}{2}\{\frac{n_j}{n_j+1} \cdot (\overline{\mathbf{x}} - \mu_0)^2 + \sum_{h=1}^{n_j} (x_h - \overline{\mathbf{x}})^2\} \end{cases} \qquad \begin{cases} \eta|\alpha, H \sim \mathbf{Beta}(\alpha + 1,\ H) \\ \pi_\eta = \dfrac{\gamma_0 + J - 1}{\gamma_0 + J - 1 + H \cdot (\psi_0 - \ln(\eta))} \end{cases}$$

E.3 Inference algorithm for DPM

Algorithm (E.3) DPM parameter-free cluster development with Gibbs Sampler

Require: Starting state $(s_1, \cdots, s_H), \alpha, (\boldsymbol{\theta}_1, \cdots, \boldsymbol{\theta}_J), (\boldsymbol{w}_1, \cdots, \boldsymbol{w}_J)$

repeat
  for $h = 1, \cdots, H$ **do**
    [Stage.1] Re-assigning cluster memberships:
      ▷ Take s_h and compute the Cl probabilities using the joint model.
    if $s_h = j$ **then**
      for $j = 1, \cdots, J$ **do**
        $P(s_h = j) = p(s_h|s_{-h}) \cdot f(z_h, x_h|\boldsymbol{w}_j) \cdot f(S_h|z_h, x_h, \boldsymbol{\theta}_j)$
          ▷ for observation h entering into existing discrete clusters.
      end for
    else if $s_h = J + 1$ **then**
      $P(s_h = J + 1) = p(s_h|s_{-h}) \cdot f_0(z_h, x_h) \cdot f_0(S_h|z_h, x_h)$
        ▷ for observation h entering into a new continuous cluster.
    end if
    Draw a Cl index from a multinomial $\{1, 2, \cdots, J + 1\}$
      ▷ with probabilities $\big(P(s_h = 1), P(s_h = 2), \cdots, P(s_h = J + 1)\big)$:Polya Urn.
    if the Cl index $= J + 1$ **then**
      Record $(\boldsymbol{\theta}_1, \cdots, \boldsymbol{\theta}_{J+1}), (\boldsymbol{w}_1, \cdots, \boldsymbol{w}_{J+1})$
    end if

    [Stage.2] Updating cluster parameters:
      ▷ Determine $\{\boldsymbol{\theta}_j, \alpha, \boldsymbol{w}_j\}$ for each cluster based on the posterior densities.
    for $j = 1, \cdots, J + 1$ **do**
      Sample $\boldsymbol{w}_j^{(*)}$ from the posterior: $p(\boldsymbol{w}|\mathbf{X}_h)$.
    end for
    Sample $\alpha^{(*)}$ from the posterior: $p(\alpha|J + 1)$.
    for $j = 1, \cdots, J + 1$ **do**
      Sample $\boldsymbol{\theta}_j^{(*)}$ from the posterior: $p(\boldsymbol{\theta}|S_j, \mathbf{X}_h)$.
    end for
    Record $(\boldsymbol{\theta}_1^{(*)}, \cdots, \boldsymbol{\theta}_{J+1}^{(*)}), (\boldsymbol{w}_1^{(*)}, \cdots, \boldsymbol{w}_{J+1}^{(*)})$
  end for
  Record $\alpha^{(*)}$

  for $h = 1, \cdots, H$ **do**
    (3) Compute the log-likelihood: $\ln\left[f(\mathbf{X}_h|\boldsymbol{w}_j^{(*)}) f(S_h|\mathbf{X}_h, \boldsymbol{\theta}_j^{(*)})\right]$
  end for
until M posterior samples $(\boldsymbol{\theta}_j^{(*)}, \alpha^{(*)}, \boldsymbol{w}_j^{(*)})$ obtained. ▷ M is a sufficient sample size.

E.4 Distribution Choices in Chapter 5

Modeling target	**Outcome (amount)**	**Covariate.1 (binary)**	**Covariate.2 (cont.)**
Data: $S, \mathbf{z}, \mathbf{x}$ (alternative)	Log-Skewnormal: $S \sim \mathbf{LogSN}(\mathbf{X}^T\boldsymbol{\beta}, \sigma^2, \xi)$ $\Rightarrow S \sim \mathbf{LogShiftedGa}(\cdot)$?	Bernoulli: $\mathbf{z} \sim \mathbf{Bern}(\pi)$ $\Rightarrow$ Logistic regression ?	Gaussian: $\mathbf{x} \sim \mathbf{N}(\mu, \lambda^2)$ $\Rightarrow \mathbf{x} \sim \mathbf{t}(\cdot)$?, $\mathbf{x} \sim \mathbf{Laplace}(\cdot)$?
Param: $\boldsymbol{\beta}, \pi, \mu$ (alternative)	Multi-Normal: $\boldsymbol{\beta} \sim \mathbf{MVN}(\boldsymbol{\beta}_0, \sigma^2\Sigma_{\beta_0})$ $\Rightarrow \boldsymbol{\beta} \sim \mathbf{MVt}(\cdot)$?	Beta: $\pi \sim \mathbf{Beta}(g_0, h_0)$ $\Rightarrow \pi \sim \mathbf{N}(\cdot)$?; $0 \leq \pi \leq 1$	Gaussian: $\mu \sim \mathbf{N}(\mu_0, \lambda^2)$ $\Rightarrow \mu \sim \mathbf{t}(\cdot)$?, $\mu \sim \mathbf{Laplace}(\cdot)$?
Scale param: σ^2, λ^2 (alternative)	Inverse Gamma: $\sigma^2 \sim \mathbf{InvGa}(u_0, v_0)$ $\Rightarrow \sigma^2 \sim \mathbf{ScaledInv}\chi^\mathbf{2}(\cdot)$?		Inverse Gamma: $\lambda^2 \sim \mathbf{InvGa}(c_0, d_0)$ $\Rightarrow \lambda^2 \sim \mathbf{ScaledInv}\chi^\mathbf{2}(\cdot)$?
Skewness param: ξ (alternative)	Student's t: $\xi \sim \mathbf{t}(\nu_0)$ $\Rightarrow \xi \sim \mathbf{SN}(\cdot)$?		

Table E.1: Distribution choices/alternatives for outcome S and covariates $\mathbf{X}$ across data, parameter models. The selection of these distributions further informs the specification of hyperparameter models.

Here we explain some conventions about distribution choices. Table E.1 provides a structured comparison of the distributional choices for the log-skewnormal outcome variables and covariates (Bernoulli and Gaussian). Additionally, it highlights alternative distributional options, offering insight into the flexibility of the model specifications. A proper guide for choosing distributions can be found in Gelman and Meng 2004; Gelman and Hill 2007; Gelman and Carlin 2013; Gelman and Hwang 2014 and the references therein.

Data: $S, \mathbf{z}, \mathbf{x}$

- **Outcome:** Aggregate claim amount S

  → our choice: $S \sim \mathbf{LogSN}(\mathbf{X}^T\boldsymbol{\beta}, \sigma^2, \xi)$; i.e. Log-skewnormal

  → possible alternative: $S \sim \mathbf{LogShiftedGa}(\cdot)$; i.e. Log-shifted Gamma ?

  - *log-space location* $\mathbf{X}^T\boldsymbol{\beta}$; i.e. Gaussian regression
  - *log-space scale* $\sigma^2 \sim \mathbf{InvGa}(u_0, v_0)$; i.e. Inverse Gamma
  - *skewness* $\xi \sim \mathbf{t}(\nu_0)$; i.e. Student's t

    : The Log-skewnormal (**LogSN**) distribution is used for our aggregate claim amount data as it offers a simple yet accurate approximation of the sum of Log-normal random variables. Alternative distributions include the Log-shifted Gamma (**LogShiftedGa**), but it can be less stable, especially when the data does not clearly exhibit the characteristics captured by the Gamma's shape parameter. In contrast, the Log-skewnormal tends to provide more robust estimates with fewer convergence issues.

: To model its parameters - log-space location $\mathbf{X}^T\boldsymbol{\beta}$, log-space scale σ^2, skewness ξ -, we use the Multivariate Gaussian for the GLM coefficients $\boldsymbol{\beta}$, the Inverse Gamma for the log-space scale, and the Student's t for the skewness. Since the Log-skewnormal lacks a conjugate relationship, alternatives such as the Multivariate Student's t (**MVt**) for the GLM coefficients, the Scaled Inverse Gamma for the scale, and the Skewnormal (**SN**) for the skewness could be considered. However, these options introduce significant additional complexity.

- **Covariate.1:** binary **z**

  → our choice: $\mathbf{z} \sim \mathbf{Bern}(\pi)$; i.e. Bernoulli

  → possible alternative: Logistic regression ?

  - *probability* $\pi \sim \mathbf{Beta}(g_0, h_0)$; i.e. Beta

    : The Bernoulli (**Bern**) is used for our binary covariate as it accounts for only two possible values (0/1). While logistic regression could be an alternative, it introduces additional complexity by requiring parameter estimation for the link function.

    : To model its parameters - probability π -, we use the beta density (**Beta**) because the range of values is defined between 0 and 1, and the density serves as a conjugate prior for **Bern**. The truncated normal (**N**) or non-informative prior such as uniform or Jeffeys prior, etc. can be alternative choices, but they come with trade-offs in terms of interpretability, flexibility, and computational complexity.

- **Covariate.2:** continuous **x**

  → our choice: $\mathbf{x} \sim \mathbf{N}(\mu, \lambda^2)$; i.e. Gaussian

  → possible alternative: $\mathbf{x} \sim \mathbf{t}(\cdot)$; i.e. Student's t ?, $\mathbf{x} \sim \mathbf{Laplace}(\cdot)$?

  - *location* $\mu \sim \mathbf{N}(\mu_0, \lambda^2)$; i.e. Gaussian
  - *scale* $\lambda^2 \sim \mathbf{InvGa}(c_0, d_0)$; i.e. Inverse Gamma

    : The Gaussian (**N**) is employed for our continuous covariate, as it effectively models data that tends to cluster around a central mean, while accommodating both positive and negative values. While the Student's t (**t**) or Laplace distributions could serve as alternatives, they are associated with specific characteristics, such as heavier tails or sharper peaks, which may not be suitable for our data.

    : To model its parameters - location μ, scale λ^2 -, we use the Gaussian for the location and the Inverse Gamma (**InvGa**) for the scale because they serve

as conjugate priors for the Gaussian (**N**), while the Inverse Gamma effectively capturing a wide range of possible values for the scale due to its heavy tail. The scaled inverse Chi-square (**ScaledInv**$\chi^{\mathbf{2}}$) can be an alternative choice, but it may be more suitable when there are prior beliefs about the degrees of freedom.

Appendix F

For Chapter 6

F.1 Data Model Development

F.1.1 Specification of data models

The outcome and covariate data models for discrete cluster development are given by:

$$
\begin{aligned}
f_{Lsn}(S\,|\mathbf{x}^*,\mathbf{x},\mathbf{z}) &= \frac{2}{\sigma_j S}\cdot\phi\left(\frac{\log S-(\beta_{j0}+\beta_{j1}\mathbf{x}+\beta_{j2}\mathbf{z})}{\sigma_j}\right)\cdot\Phi\left(\xi_j\frac{\log S-(\beta_{j0}+\beta_{j1}\mathbf{x}+\beta_{j2}\mathbf{z})}{\sigma_j}\right)\\
f_{Bern}(\mathbf{z}) &= \pi_j^{\mathbf{z}}\,(1-\pi_j)^{1-\mathbf{z}}\\
f_N(\mathbf{x}) &= \frac{1}{\sqrt{2\pi\lambda_j^2}}\exp\{\frac{-(\mathbf{x}-\bar{\mathbf{x}})^2}{2\lambda_j^2}\}
\end{aligned}
\tag{F.1}
$$

To employ the Gustafson correction technique, we introduce a group of strategic data models:

$$
\begin{aligned}
f_N(\mathbf{x}^*|\mathbf{x}) &= \frac{1}{\sqrt{2\pi\tau_j^2}}\exp\{\frac{-(\mathbf{x}^*-\mathbf{x})^2}{2\tau_j^2}\} && \textbf{: Measurement model}\\
f_N(\mathbf{x}|\mathbf{z}) &= \frac{1}{\sqrt{2\pi\lambda_j^2}}\exp\{\frac{-(\mathbf{x}-\{\kappa_{j0}+\kappa_{j1}\mathbf{z}\})^2}{2\lambda_j^2}\} && \textbf{: True exposure model}\\
f_N(\mathbf{x}^*|\mathbf{z}) &= \frac{1}{\sqrt{2\pi\widehat{\lambda}_j^2}}\exp\{\frac{-(\mathbf{x}^*-\{\widehat{\kappa}_{j0}+\widehat{\kappa}_{j1}\mathbf{z}\})^2}{2\widehat{\lambda}_j^2}\} && \textbf{: Mismeasured exposure model}
\end{aligned}
\tag{F.2}
$$

F.1.2 Parameter-free outcome data model

The parameter-free outcome model for continuous cluster development can be obtained by integrating out the outcome parameters $\widehat{\boldsymbol{\theta}} = \{\widehat{\boldsymbol{\beta}}, \widehat{\sigma}^2, \widehat{\xi}\}$.

$$f_0(S_h|x_h^*, z_h) = \int f(S_h|x_h^*, z_h, \widehat{\boldsymbol{\beta}}_j, \widehat{\sigma}_j^2, \widehat{\xi}_j) \cdot p(\widehat{\boldsymbol{\beta}}_j) \cdot p(\widehat{\sigma}_j^2) \cdot p(\widehat{\xi}_j)\, d\widehat{\boldsymbol{\beta}}\, d\widehat{\sigma}^2\, d\widehat{\xi}$$

However, it can be too complicated to compute its form analytically. Instead, we can integrate the joint model out the parameters, using Monte Carlo integration. For example, we can do the following for each $h = 1, \cdots, H$.

(i) Sample $\widehat{\boldsymbol{\beta}}, \widehat{\sigma}^2, \widehat{\xi}$ from the DP prior G_0.

(ii) Plug in these samples into $f(S_h|x_h^*, z_h, \widehat{\boldsymbol{\beta}}_j, \widehat{\sigma}_j^2, \widehat{\xi}_j) \cdot p(\widehat{\boldsymbol{\beta}}_j) \cdot p(\widehat{\sigma}_j^2) \cdot p(\widehat{\xi}_j)$.

(iii) Repeat the above steps many times, recording each output.

(iv) Divide the sum of all output values by the number of Monte Carlo samples, which will be the approximate integral.

F.1.3 Parameter-free covariate data model.I

Using the **mismeasured exposure model** defined in F.1.1,

$$\begin{cases} f_N(\mathbf{x}^*|\mathbf{z}) = \dfrac{1}{\sqrt{2\pi\widehat{\lambda}_j^2}} \exp\{\dfrac{-(\mathbf{x}^* - \{\widehat{\kappa}_{j0} + \widehat{\kappa}_{j1}\mathbf{z}\})^2}{2\widehat{\lambda}_j^2}\} \text{ where } \widehat{\boldsymbol{\kappa}}_j|\widehat{\lambda}_j^2 \sim \boldsymbol{MVN}(\tilde{\boldsymbol{\kappa}},\ \widehat{\lambda}_j^2\tilde{\Sigma}_\kappa),\ \widehat{\lambda}_j^2 \sim \boldsymbol{InvGa}(\dfrac{c_0}{2},\ \dfrac{d_0}{2}) \\ f_{Bern}(\mathbf{z}) = \pi_j^{\mathbf{z}}\,(1-\pi_j)^{1-\mathbf{z}} \text{ where } \pi_j \sim \boldsymbol{Beta}(g_0,\ h_0) \end{cases}$$

the parameter-free joint covariate model for continuous cluster development is given by:

$$\begin{aligned} f_0(\mathbf{x}^*, \mathbf{z}) &= \int f(\mathbf{x}^*, \mathbf{z} \mid \widehat{\boldsymbol{\kappa}}_j, \widehat{\lambda}_j^2, \pi_j) \cdot p_0(\widehat{\boldsymbol{\kappa}}_j|\widehat{\lambda}_j^2) \cdot p_0(\widehat{\lambda}_j^2) \cdot p_0(\pi_j)\, d\widehat{\lambda}^2\, d\widehat{\boldsymbol{\kappa}}\, d\pi \\ &= \int \frac{1}{\sqrt{2\pi\widehat{\lambda}_j^2}} \exp\{\frac{-(\mathbf{x}^* - \{\widehat{\kappa}_{j0} + \widehat{\kappa}_{j1}\mathbf{z}\})^2}{2\widehat{\lambda}_j^2}\} \times \pi_j^{\mathbf{z}}\,(1-\pi_j)^{1-\mathbf{z}} \\ &\quad \times \frac{1}{(2\pi\widehat{\lambda}^2)^{\frac{2}{2}}|\tilde{\Sigma}_\kappa|^{\frac{1}{2}}} \exp\Big(-\frac{1}{2\widehat{\lambda}^2}(\widehat{\boldsymbol{\kappa}}_j - \tilde{\boldsymbol{\kappa}})^T\tilde{\Sigma}_\kappa^{-1}(\widehat{\boldsymbol{\kappa}}_j - \tilde{\boldsymbol{\kappa}})\Big) \\ &\quad \times \frac{(d_0/2)^{\frac{c_0}{2}}}{\Gamma(c_0/2)}(\widehat{\lambda}_j^2)^{-\frac{c_0}{2}-1}e^{-\frac{d_0}{2\widehat{\lambda}_j^2}} \times \frac{1}{\boldsymbol{B}(g_0,h_0)}\pi_j^{g_0-1}(1-\pi_j)^{h_0-1}\, d\widehat{\lambda}^2\, d\widehat{\boldsymbol{\kappa}}\, d\pi \end{aligned}$$

To solve the integral above, we use:

(a) From $\pi_j \sim \boldsymbol{Beta}\Big(g_0 + z_h,\ h_0 + 1 - z_h\Big)$

$$\int \frac{1}{\boldsymbol{B}(g_0, h_0)} \pi_j^{g_0 + z_h - 1} (1 - \pi_j)^{h_0 + 1 - z_h - 1} d\pi \ = \ \frac{\boldsymbol{B}(g_0 + z_h,\ h_0 + 1 - z_h)}{\boldsymbol{B}(g_0, h_0)}$$

(b) From $\widehat{\boldsymbol{\kappa}}_j | \widehat{\lambda}_j^2 \sim \boldsymbol{MVN}\Big((\mathbb{K}_{1h}\mathbb{K}_{1h}^T + \tilde{\Sigma}_\kappa^{-1})^{-1}(\mathbb{K}_{1h}\mathbf{x}_h^* + \tilde{\Sigma}_\kappa^{-1}\tilde{\boldsymbol{\kappa}}),\quad \widehat{\lambda}_j^2(\mathbb{K}_{1h}\mathbb{K}_{1h}^T + \tilde{\Sigma}_\kappa^{-1})^{-1}\Big)$

$$\int \exp\Big(-\frac{1}{2\widehat{\lambda}_j^2}\Big[(\widehat{\boldsymbol{\kappa}}_j - (\mathbb{K}_{1h}\mathbb{K}_{1h}^T + \tilde{\Sigma}_\kappa^{-1})^{-1}(\mathbb{K}_{1h}\mathbf{x}_h^* + \tilde{\Sigma}_\kappa^{-1}\tilde{\boldsymbol{\kappa}}))^T(\mathbb{K}_{1h}\mathbb{K}_{1h}^T + \tilde{\Sigma}_\kappa^{-1}) \cdot (\widehat{\boldsymbol{\kappa}}_j - (\mathbb{K}_{1h}\mathbb{K}_{1h}^T + \tilde{\Sigma}_\kappa^{-1})^{-1}(\mathbb{K}_{1h}\mathbf{x}_h^* + \tilde{\Sigma}_\kappa^{-1}\tilde{\boldsymbol{\kappa}}))\Big]\Big) d\widehat{\boldsymbol{\kappa}} \ = \ (2\pi\widehat{\lambda}_j^2)^{\frac{2}{2}}\Big|(\mathbb{K}_{1h}\mathbb{K}_{1h}^T + \tilde{\Sigma}_\kappa^{-1})^{-1}\Big|^{\frac{1}{2}}$$

(c) From the Sherman–Morrison formula,

$$(\tilde{\Sigma}_\kappa^{-1} + \mathbb{K}_{1h}\mathbb{K}_{1h}^T)^{-1} \ = \ \tilde{\Sigma}_\kappa - \frac{\tilde{\Sigma}_\kappa \mathbb{K}_{1h}\mathbb{K}_{1h}^T \tilde{\Sigma}_\kappa}{1 + \mathbb{K}_{1h}^T \tilde{\Sigma}_\kappa \mathbb{K}_{1h}}$$

(d) From $\widehat{\lambda}_j^2 \sim \boldsymbol{InvGa}\Big(\frac{c_0 + 1}{2},\ \frac{1}{2}\big(d_0 + \frac{\big[(\mathbf{x}_h^*)^2 - \mathbb{K}_{1h}^T\tilde{\boldsymbol{\kappa}}\big]^2}{1 + \mathbb{K}_{1h}^T\tilde{\Sigma}_\kappa\mathbb{K}_{1h}}\big)\Big)$

$$\int (\widehat{\lambda}_j^2)^{-(\frac{c_0+1}{2})-1} \cdot \exp\Big(-\frac{\frac{1}{2}\big(d_0 + \Big[\frac{\big[(\mathbf{x}_h^*)^2 - \mathbb{K}_{1h}^T\tilde{\boldsymbol{\kappa}}\big]^2}{1 + \mathbb{K}_{1h}^T\tilde{\Sigma}_\kappa\mathbb{K}_{1h}}\Big]\big)}{\widehat{\lambda}_j^2}\Big) d\widehat{\lambda}^2 \ = \ \frac{\Gamma(\frac{c_0+1}{2})}{\Big[\frac{1}{2}\big(d_0 + \frac{\big[(\mathbf{x}_h^*)^2 - \mathbb{K}_{1h}^T\tilde{\boldsymbol{\kappa}}\big]^2}{1 + \mathbb{K}_{1h}^T\tilde{\Sigma}_\kappa\mathbb{K}_{1h}}\big)\Big]^{\frac{c_0+1}{2}}}$$

and they give us:

$$\begin{aligned}
&f_0(x_h^*, z_h) \\
&= \int \frac{1}{\boldsymbol{B}(g_0, h_0)} \pi_j^{g_0 + z_h - 1}(1 - \pi_j)^{h_0 + 1 - z_h - 1} \cdot \frac{1}{\sqrt{2\pi}(2\pi)} \cdot \frac{1}{|\tilde{\Sigma}_\kappa|^{\frac{1}{2}}} \cdot \frac{(d_0/2)^{\frac{c_0}{2}}}{\Gamma(c_0/2)} \\
&\quad \times \exp\Big(-\frac{1}{2\widehat{\lambda}_j^2}\Big[(\mathbf{x}_h^*)^2 - 2\widehat{\boldsymbol{\kappa}}_j^T\mathbb{K}_{1h}\mathbf{x}_h^* + \widehat{\boldsymbol{\kappa}}_j^T\mathbb{K}_{1h}\mathbb{K}_{1h}^T\widehat{\boldsymbol{\kappa}}_j + \widehat{\boldsymbol{\kappa}}_j^T\tilde{\Sigma}_\kappa^{-1}\widehat{\boldsymbol{\kappa}}_j - 2\widehat{\boldsymbol{\kappa}}_j^T\tilde{\Sigma}_\kappa^{-1}\tilde{\boldsymbol{\kappa}} + \tilde{\boldsymbol{\kappa}}^T\tilde{\Sigma}_\kappa^{-1}\tilde{\boldsymbol{\kappa}}\Big]\Big) \\
&\quad \times (\widehat{\lambda}_j^2)^{-\frac{c_0}{2} - \frac{3}{2} - 1} \cdot e^{-\frac{d_0}{2\widehat{\lambda}_j^2}} d\widehat{\boldsymbol{\kappa}}\ d\widehat{\lambda}^2\ d\pi \\
&= \int \frac{1}{\boldsymbol{B}(g_0, h_0)} \pi_j^{g_0 + z_h - 1}(1 - \pi_j)^{h_0 + 1 - z_h - 1} \cdot \frac{1}{\sqrt{2\pi}(2\pi)} \cdot \frac{1}{|\tilde{\Sigma}_\kappa|^{\frac{1}{2}}} \cdot \frac{(d_0/2)^{\frac{c_0}{2}}}{\Gamma(c_0/2)} \\
&\quad \times \exp\Big(-\frac{1}{2\widehat{\lambda}_j^2}\Big[\widehat{\boldsymbol{\kappa}}_j^T(\mathbb{K}_{1h}\mathbb{K}_{1h}^T + \tilde{\Sigma}_\kappa^{-1})\widehat{\boldsymbol{\kappa}}_j - 2\widehat{\boldsymbol{\kappa}}_j^T(\mathbb{K}_{1h}\mathbf{x}_h^* + \tilde{\Sigma}_\kappa^{-1}\tilde{\boldsymbol{\kappa}})\Big] - \frac{1}{2\widehat{\lambda}_j^2}\Big[(\mathbf{x}_h^*)^2 + \tilde{\boldsymbol{\kappa}}^T\tilde{\Sigma}_\kappa^{-1}\tilde{\boldsymbol{\kappa}}\Big]\Big) \\
&\quad \times (\widehat{\lambda}_j^2)^{-\frac{c_0}{2} - \frac{3}{2} - 1} \cdot e^{-\frac{d_0}{2\widehat{\lambda}_j^2}} d\widehat{\boldsymbol{\kappa}}\ d\widehat{\lambda}^2\ d\pi \\
&= \int \underbrace{\frac{1}{\boldsymbol{B}(g_0, h_0)} \pi_j^{g_0 + z_h - 1}(1 - \pi_j)^{h_0 + 1 - z_h - 1}}_{(a)} \cdot \frac{1}{\sqrt{2\pi}(2\pi)} \cdot \frac{1}{|\tilde{\Sigma}_\kappa|^{\frac{1}{2}}} \cdot \frac{(d_0/2)^{\frac{c_0}{2}}}{\Gamma(c_0/2)} \exp\Big(-\frac{1}{2\widehat{\lambda}_j^2} \\
&\quad \Big[\underbrace{(\widehat{\boldsymbol{\kappa}}_j - (\mathbb{K}_{1h}\mathbb{K}_{1h}^T + \tilde{\Sigma}_\kappa^{-1})^{-1}(\mathbb{K}_{1h}\mathbf{x}_h^* + \tilde{\Sigma}_\kappa^{-1}\tilde{\boldsymbol{\kappa}}))^T(\mathbb{K}_{1h}\mathbb{K}_{1h}^T + \tilde{\Sigma}_\kappa^{-1}) \cdot (\widehat{\boldsymbol{\kappa}}_j - (\mathbb{K}_{1h}\mathbb{K}_{1h}^T + \tilde{\Sigma}_\kappa^{-1})^{-1}(\mathbb{K}_{1h}\mathbf{x}_h^* + \tilde{\Sigma}_\kappa^{-1}\tilde{\boldsymbol{\kappa}}))}_{(b)}\Big] \\
&\quad - \frac{1}{2\widehat{\lambda}_j^2} \times -\Big[\underbrace{(\mathbb{K}_{1h}\mathbb{K}_{1h}^T + \tilde{\Sigma}_\kappa^{-1})^{-1}}_{(c)}(\mathbb{K}_{1h}\mathbf{x}_h^* + \tilde{\Sigma}_\kappa^{-1}\tilde{\boldsymbol{\kappa}})\Big]^T \cancel{(\mathbb{K}_{1h}\mathbb{K}_{1h}^T + \tilde{\Sigma}_\kappa^{-1})}\Big[\cancel{(\mathbb{K}_{1h}\mathbb{K}_{1h}^T + \tilde{\Sigma}_\kappa^{-1})^{-1}}(\mathbb{K}_{1h}\mathbf{x}_h^* + \tilde{\Sigma}_\kappa^{-1}\tilde{\boldsymbol{\kappa}})\Big] \\
&\quad - \frac{1}{2\widehat{\lambda}_j^2}\Big[(\mathbf{x}_h^*)^2 + \tilde{\boldsymbol{\kappa}}^T\tilde{\Sigma}_\kappa^{-1}\tilde{\boldsymbol{\kappa}}\Big]\Big) \cdot (\widehat{\lambda}_j^2)^{-\frac{c_0}{2} - \frac{3}{2} - 1} \cdot e^{-\frac{d_0}{2\widehat{\lambda}_j^2}} d\widehat{\boldsymbol{\kappa}}\ d\widehat{\lambda}^2\ d\pi
\end{aligned}$$

Finally,

$$f_0(x_h^*,\ z_h) = \frac{\boldsymbol{B}(g_0+z_h,\ h_0+1-z_h)}{\boldsymbol{B}(g_0,h_0)} \cdot \frac{1}{\sqrt{2\pi}\cancel{(2\pi)}} \cdot \frac{1}{|\tilde{\Sigma}_\kappa|^{\frac{1}{2}}} \cdot \frac{(d_0/2)^{\frac{c_0}{2}}}{\Gamma(c_0/2)} \cdot \cancel{2\pi}\Big|(\mathbb{K}_{1h}\mathbb{K}_{1h}^T + \tilde{\Sigma}_\kappa^{-1})^{-1}\Big|^{\frac{1}{2}}$$
$$\times \underbrace{\int (\widehat{\lambda}_j^2)^{-\frac{c_0}{2}-\frac{1}{2}-1} \cdot e^{-\frac{d_0}{2\widehat{\lambda}_j^2}} \exp\Big(-\frac{1}{2\widehat{\lambda}_j^2}\Big[\frac{\big[(\mathbf{x}_h^*)^2 - \mathbb{K}_{1h}^T\tilde{\boldsymbol{\kappa}}\big]^2}{1+\mathbb{K}_{1h}^T\tilde{\Sigma}_\kappa\mathbb{K}_{1h}}\Big]\Big) d\widehat{\lambda}^2}_{(d)}$$
$$= \frac{\boldsymbol{B}(g_0+z_h,\ h_0+1-z_h)}{\boldsymbol{B}(g_0,h_0)} \cdot \frac{(d_0/2)^{\frac{c_0}{2}}}{\sqrt{2\pi}\ \Gamma(c_0/2)} \cdot \frac{\Big|(\mathbb{K}_{1h}\mathbb{K}_{1h}^T + \tilde{\Sigma}_\kappa^{-1})^{-1}\Big|^{\frac{1}{2}}}{|\tilde{\Sigma}_\kappa|^{\frac{1}{2}}} \cdot \frac{\Gamma(\frac{c_0+1}{2})}{\Big[\frac{1}{2}\big(d_0 + \frac{\big[(\mathbf{x}_h^*)^2 - \mathbb{K}_{1h}^T\tilde{\boldsymbol{\kappa}}\big]^2}{1+\mathbb{K}_{1h}^T\tilde{\Sigma}_\kappa\mathbb{K}_{1h}}\big)\Big]^{\frac{c_0+1}{2}}}$$

F.1.4 Parameter-free covariate data model.II

Using the original complete covariate models defined in F.1,

$$\begin{cases} f_N(\mathbf{x}) = \dfrac{1}{\sqrt{2\pi\lambda_j^2}} \exp\{\dfrac{-(\mathbf{x}-\bar{\mathbf{x}})^2}{2\lambda_j^2}\} \text{ where } \lambda_j^2 \sim \boldsymbol{InvGa}(\dfrac{c_0}{2},\ \dfrac{d_0}{2}) \\ \\ f_{Bern}(\mathbf{z}) = \pi_j^{\mathbf{z}}\,(1-\pi_j)^{1-\mathbf{z}} \text{ where } \pi_j \sim \boldsymbol{Beta}(g_0,\ h_0) \end{cases}$$

the parameter-free joint covariate model for continuous cluster development is given by:

$$f_0(\mathbf{x},\mathbf{z}) = \int f(\mathbf{x},\mathbf{z} \mid \lambda_j^2,\pi_j) \cdot p_0(\lambda_j^2) \cdot p_0(\pi_j)\ d\lambda^2\ d\pi$$
$$= \int \frac{1}{\sqrt{2\pi\lambda_j^2}} \exp\{\frac{-(\mathbf{x}-\bar{\mathbf{x}})^2}{2\lambda_j^2}\} \times \pi_j^{\mathbf{z}}\,(1-\pi_j)^{1-\mathbf{z}}$$
$$\times \frac{(d_0/2)^{\frac{c_0}{2}}}{\Gamma(c_0/2)}(\lambda_j^2)^{-\frac{c_0}{2}-1} e^{-\frac{d_0}{2\lambda_j^2}} \times \frac{1}{\boldsymbol{B}(g_0,h_0)}\pi_j^{g_0-1}(1-\pi_j)^{h_0-1}\ d\lambda^2\ d\pi$$
$$= \frac{\boldsymbol{Beta}(\mathbf{z}+g_0,\ 1-\mathbf{z}+h_0)}{\boldsymbol{Beta}(g_0,h_0)} \times \frac{1}{\sqrt{\pi}} \cdot \frac{\Gamma(\frac{c_0+1}{2})}{\Gamma(\frac{c_0}{2})} \cdot \frac{d_0^{c_0/2}}{\big[(\mathbf{x}-\bar{\mathbf{x}})^2 + d_0\big]^{\frac{c_0+1}{2}}}$$

F.2 Parameter Model Development

F.2.1 Prior kernel for outcome, covariates, and precision

$$
\begin{aligned}
p_0(\boldsymbol{\beta}_j|\boldsymbol{\beta}_0,\ \Sigma_{\beta_0}):\ & \boldsymbol{MVN}(\boldsymbol{\beta}_0,\ \sigma_j^2\Sigma_{\beta_0})^{*} \ \propto\ e^{-\frac{1}{2\sigma_j^2}\{(\boldsymbol{\beta}_j-\boldsymbol{\beta}_0)^T\Sigma_{\beta_0}^{-1}(\boldsymbol{\beta}_j-\boldsymbol{\beta}_0)\}} \\
p_0(\sigma_j^2|u_0,\ v_0):\ & \boldsymbol{InvGa}(\frac{u_0}{2},\ \frac{v_0}{2}) \ \propto\ (\sigma_j^2)^{-(u_0/2+1)}\cdot e^{-v_0/2\sigma_j^2} \\
p_0(\xi_j|\nu_0):\ & \boldsymbol{T}(\nu_0) \ \propto\ \Big(\frac{\xi_j^2}{\nu_0}+1\Big)^{-(\nu_0+1)/2} \\
p_0(\pi_j|g_0,\ h_0):\ & \boldsymbol{Beta}(g_0,\ h_0) \ \propto\ \pi_j^{(g_0-1)}\cdot(1-\pi_j)^{(h_0-1)} \\
p_0(\lambda_j^2|c_0,\ d_0):\ & \boldsymbol{InvGa}(\frac{c_0}{2},\ \frac{d_0}{2}) \ \propto\ (\lambda_j^2)^{-(c_0/2+1)}\cdot e^{-d_0/2/\lambda_j^2} \\
p_0(\widehat{\boldsymbol{\kappa}}_j|\tilde{\kappa},\ \tilde{\Sigma}_\kappa,\ \widehat{\lambda}_j^2):\ & \boldsymbol{MVN}(\tilde{\kappa},\ \widehat{\lambda}_j^2\tilde{\Sigma}_\kappa) \ \propto\ e^{-\frac{1}{2\widehat{\lambda}_j^2}\{(\widehat{\boldsymbol{\kappa}}_j-\tilde{\boldsymbol{\kappa}})^T\tilde{\Sigma}_\kappa^{-1}(\widehat{\boldsymbol{\kappa}}_j-\tilde{\boldsymbol{\kappa}})\}} \\
p_0(\tau_j^2|c_0,\ d_0\ \widehat{\boldsymbol{\kappa}}_j,\zeta):\ & \propto\ (1-\zeta)\cdot(\lambda_j^2)^{-(c_0/2+1)}\cdot e^{-d_0/2/\lambda_j^2} \\
p_0(\alpha|\gamma_0,\ \psi_0):\ & \boldsymbol{Ga}(\gamma_0,\ \psi_0) \ \propto\ \alpha^{(\gamma_0-1)}\cdot e^{-\alpha\cdot\psi_0}
\end{aligned}
$$

* $\boldsymbol{\beta}_0, \Sigma_0 \sim$ Gamma regression

F.2.2 Posterior computation for outcome parameters

Algorithm (F.2.2): Posterior inference $\boldsymbol{\theta}_j^{(*)} = \{\boldsymbol{\beta}_j^{(*)}, \sigma_j^{2(*)}, \xi_j^{(*)}\}$

Require: initialize $\boldsymbol{\theta}_j^{(old)}$: $\begin{cases} \boldsymbol{\beta}_j \sim \boldsymbol{MVN}(\boldsymbol{\beta}_0,\ \sigma_j^2\Sigma_{\beta_0}) \\ \sigma_j^2 \sim \boldsymbol{IG}(u_0,\ v_0) \\ \xi_j \sim \boldsymbol{T}(\nu_0) \end{cases}$

1: **repeat**
2: **for** $j = 1, \cdots, J$ **do** ▷ Assume J cluster memberships.
3: Sample $\boldsymbol{\theta}^{(new)}$ from the proposal densities $\boldsymbol{q}$: ▷ Choose priors as $\boldsymbol{q}$.
$\boldsymbol{\beta}_j^{(new)} \sim \boldsymbol{q}_\beta, \quad \sigma_j^{2(new)} \sim \boldsymbol{q}_{\sigma^2}, \quad \xi_j^{(new)} \sim \boldsymbol{q}_\xi$
4: **for** $\boldsymbol{\theta}_j^{(new)}$=$\{\boldsymbol{\beta}_j^{(new)}, \sigma_j^{2(new)}, \xi_j^{(new)}\}$ **do**
5: Compute the transition ratio, using the outcome models:

$$Ratio_\theta = \frac{\prod_{h=1}^{H} f(S_h|\mathbf{X}_h,\ \boldsymbol{\theta}_j^{(new)})^1 \cdot p_0(\boldsymbol{\theta}_j^{(new)}) \cdot \boldsymbol{q}_\theta(\boldsymbol{\theta}_j^{(old)})}{\prod_{h=1}^{H} f(S_h|\mathbf{X}_h,\ \boldsymbol{\theta}_j^{(old)})^1 \cdot p_0(\boldsymbol{\theta}_j^{(old)}) \cdot \boldsymbol{q}_\theta(\boldsymbol{\theta}_j^{(new)})}$$

Sample $U \sim \boldsymbol{Unif}(0,1)$
6: **if** $U < Ratio_\theta$ **then** $\boldsymbol{\theta}_j^{(*)} = \boldsymbol{\theta}_j^{(new)}$ otherwise $\boldsymbol{\theta}_j^{(*)} = \boldsymbol{\theta}_j^{(old)}$
7: **end if**
8: **end for**
9: Record $\boldsymbol{\theta}_j^{(*)}$
10: **end for**
11: **until** M posterior samples $(\boldsymbol{\theta}_{j=1,\cdots,J}^{(*)})$ obtained. ▷ M is a sufficient sample size

[1] The outcome density in Line 5 is give by $f_{LSN}(S_h|\mathbf{X}_h, \boldsymbol{\theta}_j)$.

F.2.3 Posterior computation for covariates and precision

$$p(\pi_j|\mathbf{z}) \;\propto\; f(\mathbf{z}|\pi_j)\cdot p_0(\pi_j) \qquad \propto \Big[\prod_{h=1}^{n_j}\pi_j^{z_h}\,(1-\pi_j)^{1-z_h}\Big]\cdot \pi_j^{g_0-1}(1-\pi_j)^{h_0-1}$$

$$\propto\; \pi_j^{g_0+\Sigma z_h-1}(1-\pi_j)^{h_0+n_j-\Sigma z_h-1}$$

$$\therefore\; \pi_j \;\big|\; z_1,\ldots,z_{n_j} \sim \boldsymbol{Beta}\Big(g_0+\Sigma_{h=1}^{n_j}z_h,\quad h_0+n_j-\Sigma_{h=1}^{n_j}z_h\Big)$$

$$p(\widehat{\lambda}_j^2|\mathbf{x}^*,\mathbf{z}) \;\propto\; f(\mathbf{x}^*|\widehat{\boldsymbol{\kappa}}_j,\widehat{\lambda}_j^2,\mathbf{z})\cdot p_0(\widehat{\lambda}_j^2)$$

$$\propto \Big[\prod_{h=1}^{n_j}\frac{1}{\sqrt{2\pi\widehat{\lambda}_j^2}}\exp\Big(\frac{-1}{2\widehat{\lambda}_j^2}(x_h^*-\widehat{\kappa}_{j0}-\widehat{\kappa}_{j1}z_h)^2\Big)\Big]\cdot\frac{(d_0/2)^{\frac{c_0}{2}}}{\Gamma(c_0/2)}(\widehat{\lambda}_j^2)^{-\frac{c_0}{2}-1}e^{-\frac{d_0}{2\widehat{\lambda}_j^2}}$$

$$\propto\; (\widehat{\lambda}_j^2)^{-\left(\frac{n_j+c_0}{2}\right)-1}\exp\left(\frac{-1}{\widehat{\lambda}_j^2}\Big[\frac{1}{2}\Big(\Sigma_{h=1}^{n_j}(x_h^*-\widehat{\kappa}_{j0}-\widehat{\kappa}_{j1}z_h)^2+d_0\Big)\Big]\right)$$

$$\therefore\; \widehat{\lambda}_j^2 \;\big|\; x_1^*,\ldots,x_{n_j}^*,z_1,\ldots,z_{n_j},\widehat{\boldsymbol{\kappa}}_j \sim \boldsymbol{InvGa}\Big(\frac{n_j+c_0}{2},\quad \frac{1}{2}\Big(\Sigma_{h=1}^{n_j}(x_h^*-\widehat{\kappa}_{j0}-\widehat{\kappa}_{j1}z_h)^2+d_0\Big)\Big)$$

$$p(\widehat{\boldsymbol{\kappa}}_j|\mathbf{x}^*,\mathbf{z},\widehat{\lambda}_j^2) \;\propto f(\mathbf{x}^*|\widehat{\boldsymbol{\kappa}}_j,\widehat{\lambda}_j^2,\mathbf{z})\cdot p_0(\widehat{\boldsymbol{\kappa}}_j)$$

$$\propto \Big[\prod_{h=1}^{n_j}\frac{1}{\sqrt{2\pi\widehat{\lambda}_j^2}}\exp\Big(\frac{-1}{2\widehat{\lambda}_j^2}(x_h^*-\widehat{\kappa}_{j0}-\widehat{\kappa}_{j1}z_h)^2\Big)\Big]\cdot\frac{1}{|2\pi\tilde{\Sigma}_\kappa|^{\frac{1}{2}}}\exp\Big(\frac{-1}{2\widehat{\lambda}_j^2}(\widehat{\boldsymbol{\kappa}}_j-\tilde{\boldsymbol{\kappa}})^T\tilde{\Sigma}_\kappa^{-1}(\widehat{\boldsymbol{\kappa}}_j-\tilde{\boldsymbol{\kappa}})\Big)$$

$$\propto\; \exp\left(\frac{-1}{2\widehat{\lambda}_j^2}\Big[\Sigma_{h=1}^{n_j}(x_h^*-\widehat{\kappa}_{j0}-\widehat{\kappa}_{j1}z_h)^2\Big]-\frac{1}{2\widehat{\lambda}_j^2}(\widehat{\boldsymbol{\kappa}}_j-\tilde{\boldsymbol{\kappa}})^T\tilde{\Sigma}_\kappa^{-1}(\widehat{\boldsymbol{\kappa}}_j-\tilde{\boldsymbol{\kappa}})\right)$$

$$\propto\; \exp\left(\frac{-1}{2\widehat{\lambda}_j^2}\Big[\Sigma_{h=1}^{n_j}x_h^{*2}-2(\widehat{\kappa}_{j0}+\widehat{\kappa}_{j1}z_h)x_h^*+(\widehat{\kappa}_{j0}+\widehat{\kappa}_{j1}z_h)^2\Big]-\frac{1}{2\widehat{\lambda}_j^2}(\widehat{\boldsymbol{\kappa}}_j-\tilde{\boldsymbol{\kappa}})^T\tilde{\Sigma}_\kappa^{-1}(\widehat{\boldsymbol{\kappa}}_j-\tilde{\boldsymbol{\kappa}})\right)$$

$$\propto\; \exp\left(\frac{-1}{2\widehat{\lambda}_j^2}\Big[\widehat{\boldsymbol{\kappa}}_j^T\mathbb{K}_1^T\mathbb{K}_1\widehat{\boldsymbol{\kappa}}_j-2\widehat{\boldsymbol{\kappa}}_j^T\mathbb{K}_2\Big]-\frac{1}{2\widehat{\lambda}_j^2}(\widehat{\boldsymbol{\kappa}}_j-\tilde{\boldsymbol{\kappa}})^T\tilde{\Sigma}_\kappa^{-1}(\widehat{\boldsymbol{\kappa}}_j-\tilde{\boldsymbol{\kappa}})-\frac{-1}{2\widehat{\lambda}_j^2}\Sigma_{h=1}^{n_j}x_h^{*2}\right)$$

$$\propto\; \exp\left(\frac{-1}{2\widehat{\lambda}_j^2}\Big[\widehat{\boldsymbol{\kappa}}_j^T\tilde{\Sigma}_\kappa^{-1}\widehat{\boldsymbol{\kappa}}_j+\widehat{\boldsymbol{\kappa}}_j^T\mathbb{K}_1^T\mathbb{K}_1\widehat{\boldsymbol{\kappa}}_j-2\widehat{\boldsymbol{\kappa}}_j^T\mathbb{K}_2-2\widehat{\boldsymbol{\kappa}}_j^T\tilde{\Sigma}_\kappa^{-1}\tilde{\boldsymbol{\kappa}}+\tilde{\boldsymbol{\kappa}}^T\tilde{\Sigma}_\kappa^{-1}\tilde{\boldsymbol{\kappa}}\Big]\right)=\exp\Big(-\frac{1}{2}\mathbb{A}^T\mathbb{B}^{-1}\mathbb{A}\Big)$$

$$\text{with } \mathbb{A}=\Big[\widehat{\boldsymbol{\kappa}}_j-(\tilde{\Sigma}_k^{-1}+\mathbb{K}_1^T\mathbb{K}_1)^{-1}(\tilde{\Sigma}_k^{-1}\tilde{\boldsymbol{\kappa}}+\mathbb{K}_2)\Big],\quad \mathbb{B}=\widehat{\lambda}_j^2\Big[\tilde{\Sigma}_k^{-1}+\mathbb{K}_1^T\mathbb{K}_1\Big]^{-1}$$

$$\therefore\; \widehat{\boldsymbol{\kappa}}_j \;\big|\; x_1^*,\ldots,x_{n_j}^*,z_1,\ldots,z_{n_j},\widehat{\lambda}_j^2 \sim \boldsymbol{MVN}\left(\Big[(\tilde{\Sigma}_k^{-1}+\mathbb{K}_1^T\mathbb{K}_1)^{-1}(\tilde{\Sigma}_k^{-1}\tilde{\boldsymbol{\kappa}}+\mathbb{K}_2)\Big],\quad \widehat{\lambda}_j^2\Big[\tilde{\Sigma}_k^{-1}+\mathbb{K}_1^T\mathbb{K}_1\Big]^{-1}\right)$$

$$p(\tau_j^2|\mathbf{x}^*,\,\mathbf{x}) \;\approx\; p(\tau_j^2|\mathbf{x}^*,\,\mathbf{z},\,\widehat{\boldsymbol{\kappa}}_j) \;=\; p(\,(1-\zeta)\widehat{\lambda}_j^2 \mid \mathbf{x}^*,\,\mathbf{z},\,\widehat{\boldsymbol{\kappa}}_j\,)$$

$$\therefore\; \tau_j^2 \;\big|\; x_1^*,\ldots,x_{n_j}^*,z_1,\ldots,z_{n_j},\widehat{\boldsymbol{\kappa}}_j \sim \boldsymbol{InvGa}\left(\frac{n_j+c_0}{2},\quad \frac{(1-\zeta)}{2}\Big[\Sigma_{h=1}^{n_j}(x_h^*-\widehat{\kappa}_{j0}-\widehat{\kappa}_{j1}z_h)^2+d_0\Big]\right)$$

$$
\begin{aligned}
p(\alpha \mid J) \;&\propto\; \alpha^{J} \cdot \frac{\alpha+n}{\alpha} \cdot \boldsymbol{Beta}(\alpha+1, n) \cdot p_0(\alpha) \\
&\propto\; p_0(\alpha) \cdot \alpha^{J-1} \cdot (\alpha+n) \cdot \boldsymbol{Beta}(\alpha+1, n) \\
&\propto \alpha^{\gamma_0+j-2} \cdot (\alpha+n) \cdot e^{-\alpha(\psi_0 - ln(\eta))} \\
\therefore\; \alpha \mid J, \eta, \gamma_0, \psi_0 \;&\sim\; \pi_\eta \boldsymbol{Ga}\Big(\gamma_0 + J,\; \psi_0 - ln(\eta)\Big) + (1-\pi_\eta)\boldsymbol{Ga}\Big(\gamma_0 + J - 1,\; \psi_0 - ln(\eta)\Big)
\end{aligned}
$$

F.3 Inference Algorithm for Gustafson DPM

Algorithm (F.3): Gustafson DPM Gibbs Sampling for new cluster development

Require: Starting state ID:$(s_1, \cdots, s_H)$, α, $\widehat{\boldsymbol{\theta}} : (\widehat{\boldsymbol{\theta}}_1, \cdots, \widehat{\boldsymbol{\theta}}_J)$, $\widehat{\boldsymbol{w}} : (\widehat{\boldsymbol{w}}_1, \cdots, \widehat{\boldsymbol{w}}_J)$

repeat
  for $h = 1, \cdots, H$ **do**
    [Stage.1] Re-assigning cluster memberships (ID):
    if sum(ID == ID[h]) == 1 **then**
      ▷ Check if any cluster contains only a single data point.
      ID[ID > ID[h]] = ID[ID > ID[h]] - 1
      $\widehat{\boldsymbol{w}} = \widehat{\boldsymbol{w}}$[- ID[h]]; $\widehat{\boldsymbol{\theta}} = \widehat{\boldsymbol{\theta}}$[- ID[h]]
      J = J - 1 ▷ Remove the cluster as well as its cluster membership (ID).
    end if

    (A) Initialize the clusters.
    ID[h] = 0
    Create a vector of current cluster size $\{n_1, \cdots, n_J\}$.
    Create an empty vector to perform Polya Urn scheme: $\big[P_1, \cdots, P_J, P_{J+1}\big]$.

    (B) Iterate through each cluster to compute Cl.probabilities.
    if $s_h = j$ **then** ▷ cl.membership (ID) of the observation h is $j = 1 \cdots J$.
      for $j = 1, \cdots, J$ **do**
        $P(s_h = j) = p(s_h|s_{-h}) \cdot f(x_h, z_h|\widehat{\boldsymbol{w}}_j) \cdot f(S_h|x_h, z_h, \widehat{\boldsymbol{\theta}}_j)$
          ▷ STAY: for observation h entering into existing discrete clusters.
      end for
      Record $\big[P(s_h = 1), P(s_h = 2), \cdots, P(s_h = J)\big]$
    else if $s_h = J + 1$ **then** ▷ cl.membership (ID) of the observation h is $J + 1$.
      $P(s_h = J+1) = p(s_h|s_{-h}) \cdot f_0(x_h, z_h) \cdot f_0(S_h|x_h, z_h)$
        ▷ MOVE: for observation h entering into a new continuous cluster.
      Record $\big[P(s_h = 1), P(s_h = 2), \cdots, P(s_h = J), P(s_h = J+1)\big]$
    end if

    (C) Draw a "new Cl.index" from a multinomial $\{1, 2, \cdots, J+1\}$
      ▷ with probabilities $\big[P(s_h = 1), P(s_h = 2), \cdots, P(s_h = J+1)\big]$:Polya Urn.
    if "new Cl.index" $= J + 1$ **then**
      Collect $\widehat{\boldsymbol{\theta}}_{J+1}$ and $\widehat{\boldsymbol{w}}_{J+1}$ from G_0.
      Record $(\widehat{\boldsymbol{\theta}}_1, \cdots, \widehat{\boldsymbol{\theta}}_{J+1}), (\widehat{\boldsymbol{w}}_1, \cdots, \widehat{\boldsymbol{w}}_{J+1})$
      $J = J + 1$ ▷ new J is determined.
    end if
  end for

Algorithm (F.3): *Cont.*

[Stage.2] Updating cluster parameters: ▷ with new J

Create an empty vector of measurement parameter $\boldsymbol{w}^{new}$:$\{\tau_1^2, \cdots, \tau_J^2\}$ for $\mathbf{x}^*|\mathbf{x}$.

Create an empty vector $\boldsymbol{w}^{new}$:$\{\lambda_1^2, \cdots, \lambda_J^2\}$ for the correction of $\mathbf{x}^*|\mathbf{z}$.

Create an empty matrix $\boldsymbol{w}^{new}$:$\{\boldsymbol{\kappa}_1, \cdots, \boldsymbol{\kappa}_J\}$ for the correction of $\mathbf{x}^*|\mathbf{z}$.

Create an empty matrix $\boldsymbol{\theta}^{new}$:$\{\boldsymbol{\beta}_1, \cdots, \widehat{\boldsymbol{\beta}}_J\}$ for the correction of S.

Create an empty vector $\boldsymbol{\theta}^{new}$:$\{\sigma_1^2, \cdots, \sigma_J^2\}$ for the correction of S.

Create an empty vector $\boldsymbol{\theta}^{new}$:$\{\xi_1, \cdots, \xi_J\}$ for the corection of S.

Set the value of ζ to compute τ^2. ▷ $\tau^2 = (1-\zeta) \times \widehat{\lambda}^2$

Sample α^{new} from the posterior: $p(\alpha|J)$ in F.2.3.

for $j = 1, \cdots, J$ **do**

$\boldsymbol{w}_j^{new}$: Sample π_j from the posterior: $p(\pi_j|\mathbf{z}_h)$ in F.2.3.

State the required conditions for Gustafson's equation to be valid in F.4:

d1=$\widehat{\lambda}_j^2 - \tau_j^2 > 0$; d2=$\widehat{\sigma}_j^2 - \frac{\beta_{j1}^2}{\frac{1}{\tau_j^2}+\frac{1}{\lambda_j^2}} > 0$; d3=$\frac{1}{\tau_j^2} + \frac{1}{\lambda_j^2} - \frac{\widehat{\xi}_j^2 \beta_{j1}^2}{\sigma_j^2} > 0$

Sampling under the required conditions:

while d1≤ 0 | d2≤ 0 | d3≤ 0 **do**

Sample $\widehat{\lambda}_j^2$ from the posterior: $p(\widehat{\lambda}_j^2|\mathbf{x}^*)$ in F.2.3.

Sample τ_j^2 from the posterior: $p(\tau_j^2|\mathbf{x}^*)$ in F.2.3.

Sample $\widehat{\boldsymbol{\kappa}}_j$ from the posterior: $p(\widehat{\boldsymbol{\kappa}}_j|\mathbf{x}^*)$ in F.2.3.

Sample $\widehat{\boldsymbol{\beta}}_j$ via MH in F.2.2.

Sample $\widehat{\sigma}_j^2$ via MH in F.2.2.

Sample $\widehat{\xi}_j$ via MH in F.2.2.

Compute $\lambda_j^2 = \widehat{\lambda}_j^2 - \tau_j^2$

Compute $\beta_{j1} = \frac{\widehat{\beta}_{j1}\widehat{\lambda}_j^2}{\widehat{\lambda}_j^2 - \tau_j^2}$

Compute $\sigma_j^2 = \widehat{\sigma}_j^2 - \frac{\beta_{j1}^2}{\frac{1}{\tau_j^2}+\frac{1}{\lambda_j^2}}$

end while ▷ Collect ONLY the samples that meet the conditions in line 47.

Record $(\widehat{\boldsymbol{\theta}}_1, \cdots, \widehat{\boldsymbol{\theta}}_J), (\widehat{\boldsymbol{w}}_1, \cdots, \widehat{\boldsymbol{w}}_J)$

Gustafson correction:

$\boldsymbol{w}_j^{new}$: $\kappa_{j0} = \widehat{\kappa}_{j0}$; $\kappa_{j1} = \widehat{\kappa}_{j1}$; $\lambda_j^2 = \widehat{\lambda}_j^2 - \tau_j^2$

$\boldsymbol{\theta}_j^{new}$: $\beta_{j0} = \widehat{\beta}_{j0} - \frac{\beta_{j1}\widehat{\kappa}_{j0}\tau_j^2}{\widehat{\lambda}_j^2}$; $\beta_{j1} = \frac{\widehat{\beta}_{j1}\widehat{\lambda}_j^2}{\widehat{\lambda}_j^2 - \tau_j^2}$; $\beta_{j2} = \widehat{\beta}_{j2} - \frac{\beta_{j1}\widehat{\kappa}_{j1}\tau_j^2}{\widehat{\lambda}_j^2}$

$\boldsymbol{\theta}_j^{new}$: $\sigma_j^2 = \widehat{\sigma}_j^2 - \frac{\beta_{j1}^2}{\frac{1}{\tau_j^2}+\frac{1}{\lambda_j^2}}$; $\xi_j^2 = \frac{\widehat{\xi}_j^2\left(\frac{\beta_{j1}^2}{\sigma_j^2}+\frac{1}{\tau_j^2}+\frac{1}{\lambda_j^2}\right)}{\frac{1}{\tau_j^2}+\frac{1}{\lambda_j^2}-\frac{\widehat{\xi}_j^2\beta_{j1}^2}{\sigma_j^2}}$

Record $(\boldsymbol{\theta}_1^{new}, \cdots, \boldsymbol{\theta}_J^{new}), (\boldsymbol{w}_1^{new}, \cdots, \boldsymbol{w}_J^{new})$

end for

Record α^{new}

for $h = 1, \cdots, H$ **do**

Compute the log-likelihood (LPPD) using: $\ln[f(\mathbf{X}_h|\boldsymbol{w}_j^{new}) f(S_h|\mathbf{X}_h, \boldsymbol{\theta}_j^{new})]$

end for

until M posterior samples $(\boldsymbol{\theta}_j^{new}, \alpha^{new}, \boldsymbol{w}_j^{new})$ for $j = 1, \cdots, J$ obtained.

▷ M: total iterations

F.4 Shard Computation for Large-scale Inference

Algorithm (F.4): Parallel Simulation Wrapper for Gibbs sampler

Require: Partition the training set $\{S_h, \mathbf{X}_h\}$ for $h = 1, \cdots, H$ into non-overlapping subsets $Set_{(1)}, \cdots, Set_{(N.sh+1)}$ of similar size to create disjoint shards $SH_{(1)}, \cdots, SH_{(N.sh)}$ and one anchor set $\mathbb{A}$. The number of shards $N.sh$ must be pre-determined.

(1) Assign random subset memberships to each observation in the training set.
(2) Collect the indices of anchor points in the subset $\mathbb{A} = Set_{(N.sh+1)}$.
(3) Collect the indices of observations in the subsets $Set_{(1)}, \cdots, Set_{(N.sh)}$
: shard.index.list[[sh]] = a vector of indices where, for $sh = 1, \cdots, N.sh$,
"subset.membership" == sh | "subset.membership" == $N.sh + 1$.
▷ In each shard $SH_{(sh)}$, indices in Set_{sh} and the anchor set $\mathbb{A}$ are merged.

[Stage.1] Parallel MCMC on Multiple Shards:
for $sh = 1, \cdots, N.sh$ **do**
  Collect observations $\{S_h, \mathbf{X}_h\}_{(sh)}$ and $Cl_{(sh)}$ indices for each shard:
    $S_{h_{(sh)}} = S_h$ [shard.index.list[[sh]]] ▷ a vector of outcome in $SH_{(sh)}$.
    $\mathbf{X}_{h_{(sh)}} = \mathbf{X}_h$ [shard.index.list[[sh]]] ▷ a matrix of covariates in $SH_{(sh)}$.
    $Cl_{(sh)} = Cl$ [shard.index.list[[sh]]] ▷ a vector of cl.memberships in $SH_{(sh)}$.
end for

With each machine, ▷ $\Rightarrow N.sh$ CPUs.
for $r = 1, \cdots, M$ **do** ▷ Wrapping a Gibbs sampler with 'M' iterations.
  Run "Gibbs sampler" built on Hierarchical GLM / DPM framework:

$$(\boldsymbol{\theta}_1, \cdots, \boldsymbol{\theta}_J)_{(sh)}, (\boldsymbol{w}_1, \cdots, \boldsymbol{w}_J)_{(sh)} = \mathbf{MCMC}\Big(\{S_h, \mathbf{X}_h\}_{(sh)},\ Cl_{(sh)}\Big)^1$$

end for

Save a file "sh.data(sh).rds": $\begin{cases} (1)\ \text{shard.index.list [[}sh\text{]]} \\ (2)\ \text{anchor.index} \\ (3)\ \underline{Cl}_{(sh)},\ (\boldsymbol{\theta}_1, \cdots, \boldsymbol{\theta}_J)_{(sh)},\ (\boldsymbol{w}_1, \cdots, \boldsymbol{w}_J)_{(sh)} \end{cases}$

[Stage.2] Merging Clustering Results from Multiple Shards:
Step(1): Load the files, store them in a list, and determine the threshold $0 < \epsilon < 1$.
Step(2): Create a binary matrix (row: obs.index; column: cluster membership) to track changes in cluster membership for each observation in the shard:
for $r = 1, \cdots, M$ **do**
  $\underline{Cl}.\text{matrix}\ [[r]] = \mathbf{matrix}\Big(\ 0,\ \text{nrow} = H,\ \text{ncol} = \max(\underline{Cl}_{(sh=1)}[[r]])\ \Big)$
  Marking the anchor points in the first shard:
  for $i = 1, \cdots,$ length(shard.index.list [[$sh = 1$]]) **do**
    $\underline{Cl}.\text{matrix}\ [[r]]$ [shard.index.list [[$sh = 1$]][i], $\underline{Cl}_{(sh)}[[r]][i]$] $= 1$
  end for
end for ▷ The matrix expands with new columns as the merging process continues.

[1] See Algorithm F.2.2, F.3 for further details.

Algorithm (F.4): *Cont.*

for $r = 1, \cdots, M$ **do**

Step(3): Store indices from $\underline{Cl}_{(sh=1)}$ in $SH_{(1)}$ as well as subsequent $\underline{Cl}_{(sh=2,\cdots,N.sh)}$ in $SH_{(2)}, \cdots SH_{(N.sh)}$ and save them in a variable that we can access when computing the cluster distance $\mathbf{Dist}_{(1,sh)}$.

Step(4): Compute the cluster distance $0 \leq \mathbf{Dist}_{(1,sh)} \leq 1$ to measure similarity between clusters, and determine if the clusters should be merged or not:

$$\underline{Cl}_{(sh=1)}[[r]] \quad \text{vs} \quad \underline{Cl}_{(sh=2,\cdots,N.sh)}[[r]]$$

for $sh = 2, \cdots, N.sh$ **do**

for $i = 1, \cdots, \max(\underline{Cl}_{(sh)}[[r]])$ **do**

Collect indices on $\underline{Cl}_{(sh=2,\cdots,N.sh)}$: $\begin{cases} \text{(a) indices of observations in } \underline{Cl}_{(sh)}[[r]][i] \\ \text{(b) indices of anchor points in } \underline{Cl}_{(sh)}[[r]][i]. \end{cases}$

MERGE $= FALSE$

$j = 1$

while $j \leq \max(\underline{Cl}_{(sh=1)}[[r]])$ | MERGE $==$ $FALSE$ **do**

Collect indices on $\underline{Cl}_{(sh=1)}$: $\begin{cases} \text{(c) indices of observations in } \underline{Cl}_{(sh=1)}[[r]][j] \\ \text{(d) indices of anchor points in } \underline{Cl}_{(sh=1)}[[r]][j]. \end{cases}$

Count indices $\begin{cases} \text{same anchor indices: } CNT_c \\ \text{different anchor indices: } CNT_d \end{cases}$

if $CNT_c > 0 \mid CNT_d > 0$ **then**

Compute $\mathbf{Dist}_{(1,sh)} = \frac{CNT_d}{CNT_c + CNT_d}$ and compare with $0 < \epsilon < 1$.

else

Compute $\mathbf{Dist}_{(1,sh)} = 1$ and compare with $0 < \epsilon < 1$.

end if

▷ Anchor points may not be present in some clusters in any shard.

if $\mathbf{Dist}_{(1,sh)} < \epsilon$ **then**

MERGE $= TRUE$

Cl.location $= j$ ▷ jth column needs to be modified $(0 \rightarrow 1)$.

end if

$j = j + 1$

end while

if MERGE $==$ $TRUE$ **then**

Merge clusters:

(a) Set $\underline{Cl.matrix}$ [[r]] [, Cl.location] $= 1$ for all indices in $\underline{Cl}_{(sh)}[[r]][i]$

(b) The resulting parameters for each cluster in the shard $SH_{(1)}$ are re-computed, using a weighted average with cluster, subset sizes.

else

Do not merge clusters:

(a) Append new columns to $\underline{Cl.matrix}$ [[r]] [, $\max(\underline{Cl}_{(sh=1)}[[r]])$]

(b) The resulting parameters for each cluster are simply appended to $\underline{Cl}_{(sh=1)}[[r]]$ for the shard $SH_{(1)}$.

end if

end for

end for

Algorithm (F.4): *Cont.*

52: Step(5): Refine the binary matrix $\underline{Cl.\text{matrix}}$ $[[r]]$ for the shard $SH_{(1)}$ by removing duplicates in the tabulation:

53: **for** index in $\mathbb{A}$ **do**

keep = sample(x=which($\underline{Cl.\text{matrix}}$ $[[r]]$[index,]==1), size = 1)

$\underline{Cl.\text{matrix}}$ $[[r]]$[index, -keep] = 0

54: **end for**

▷ In cases where multiple 1s appear in the same row in the $\underline{Cl.\text{matrix}}$, which are duplicates, we randomly remove all but one.

55: **end for**

F.5 Derivation of Gustafson's Equations with Log-skewnormal outcome

With the outcome model definitions:

$$f(S,\mathbf{x}^*,\mathbf{x}\mid\mathbf{z}) = \frac{2}{\sigma_j S\sqrt{2\pi}}\underbrace{\exp\left(-\frac{1}{2}\left[\frac{\log S-(\beta_{j0}+\beta_{j1}\mathbf{x}+\beta_{j2}\mathbf{z})}{\sigma_j}\right]^2\right)}_{\phi(\cdot)}$$
$$\times\underbrace{\int_{-\infty}^{\xi_j\frac{\log S-(\beta_{j0}+\beta_{j1}\mathbf{x}+\beta_{j2}\mathbf{z})}{\sigma_j}}\frac{1}{\sqrt{2\pi}}e^{-\frac{u^2}{2}}\,du}_{\Phi(\cdot)}\times\frac{1}{\sqrt{2\pi\tau_j^2}}\exp\{\frac{-(\mathbf{x}^*-\mathbf{x})^2}{2\tau_j^2}\}\times\frac{1}{\sqrt{2\pi\lambda_j^2}}\exp\{\frac{-(\mathbf{x}-\{\kappa_{j0}+\kappa_{j1}\mathbf{z}\})^2}{2\lambda_j^2}\}$$

$$f(S,\mathbf{x}^*\mid\mathbf{z}) = \frac{2}{\widehat{\sigma}_j S\sqrt{2\pi}}\underbrace{\exp\left(-\frac{1}{2}\left[\frac{\log S-(\widehat{\beta}_{j0}+\widehat{\beta}_{j1}\mathbf{x}^*+\widehat{\beta}_{j2}\mathbf{z})}{\widehat{\sigma}_j}\right]^2\right)}_{\phi(\cdot)}$$
$$\times\underbrace{\int_{-\infty}^{\widehat{\xi}_j\frac{\log S-(\widehat{\beta}_{j0}+\widehat{\beta}_{j1}\mathbf{x}^*+\widehat{\beta}_{j2}\mathbf{z})}{\widehat{\sigma}_j}}\frac{1}{\sqrt{2\pi}}e^{-\frac{u^2}{2}}\,du}_{\Phi(\cdot)}\times\frac{1}{\sqrt{2\pi\widehat{\lambda}_j^2}}\exp\{\frac{-(\mathbf{x}^*-\{\widehat{\kappa}_{j0}+\widehat{\kappa}_{j1}\mathbf{z}\})^2}{2\widehat{\lambda}_j^2}\}$$

where $\phi(\cdot)$ and $\Phi(\cdot)$ refer to a standard normal probability density function and a cumulative density function respectively. Now we aim to discover the expressions for $\widehat{\beta}_{j0},\widehat{\beta}_{j1},\widehat{\beta}_{j2},\widehat{\sigma}_j,\widehat{\xi}_j,\widehat{\lambda_j},\widehat{\kappa}_{j0},\widehat{\kappa}_{j1}$ above

Unobserved complete case	**Incomplete case**
$\underbrace{f(S\mid\mathbf{x}^*,\mathbf{x},\mathbf{z})}_{\text{outcome}}\cdot\underbrace{f(\mathbf{x}^*\mid\mathbf{x})}_{\text{measurement}}\cdot\underbrace{f(\mathbf{x}\mid\mathbf{z})}_{\text{exposure}}$	$\underbrace{f(S\mid\mathbf{x}^*,\mathbf{z})}_{\text{outcome}}\cdot\underbrace{f(\mathbf{x}^*\mid\mathbf{z})}_{\text{exposure}}$
$=f(S,\mathbf{x}^*,\mathbf{x}\mid\mathbf{z})$	$=f(S,\mathbf{x}^*\mid\mathbf{z})$

by matching up the parameterizations from:

$$\int f(S, \mathbf{x}^*, \mathbf{x} \mid \mathbf{z})\, d\mathbf{x} \;=\; f(S, \mathbf{x}^* \mid \mathbf{z})$$

To begin with, we aim to evaluate the following integral:

$$\begin{aligned}
&\int f(S, \mathbf{x}^*, \mathbf{x} \mid \mathbf{z})\, d\mathbf{x} \\
&\quad= \int \left[\frac{2}{\sigma_j S\sqrt{2\pi}} \exp\left(-\frac{1}{2}\Big[\frac{\log S - (\beta_{j0} + \beta_{j1}\mathbf{x} + \beta_{j2}\mathbf{z})}{\sigma_j} \Big]^2 \right) \times \int_{-\infty}^{\xi_j \frac{\log S - (\beta_{j0} + \beta_{j1}\mathbf{x} + \beta_{j2}\mathbf{z})}{\sigma_j}} \frac{1}{\sqrt{2\pi}} e^{-\frac{u^2}{2}}\, du \right. \\
&\qquad\qquad \left. \times \frac{1}{\sqrt{2\pi\tau_j^2}} \exp\{ \frac{-(\mathbf{x}^* - \mathbf{x})^2}{2\tau_j^2} \} \times \frac{1}{\sqrt{2\pi\lambda_j^2}} \exp\{ \frac{-(\mathbf{x} - \{\kappa_{j0} + \kappa_{j1}\mathbf{z}\})^2}{2\lambda_j^2} \} \right] d\mathbf{x} \\
&= \frac{2}{S\sigma_j\tau_j\lambda_j(2\pi)^2} \int_{\mathbf{x}} \int_{u=-\infty}^{\xi_j \frac{\log S - (\beta_{j0} + \beta_{j1}\mathbf{x} + \beta_{j2}\mathbf{z})}{\sigma_j}} \exp\left(-\frac{1}{2}\Big[\frac{\log S - (\beta_{j0} + \beta_{j1}\mathbf{x} + \beta_{j2}\mathbf{z})}{\sigma_j} \Big]^2 - \frac{u^2}{2} - \frac{1}{2\tau_j^2}(\mathbf{x}^* - \mathbf{x})^2 \right. \\
&\qquad\qquad \left. - \frac{1}{2\lambda_j^2}\big(\mathbf{x} - \{\kappa_{j0} + \kappa_{j1}\mathbf{z}\}\big)^2 \right) du\, d\mathbf{x}
\end{aligned}$$

with the substitution of $u = \frac{\xi_j}{\sigma_j} v$ where $v = \Big[t + \log S - (\beta_{j0} + \beta_{j1}\mathbf{x} + \beta_{j2}\mathbf{z}) \Big]$, $du = \frac{\xi_j}{\sigma_j} dv$,
and $t = v - \log S + (\beta_{j0} + \beta_{j1}\mathbf{x} + \beta_{j2}\mathbf{z})$, $dt = dv$, then

$$\begin{aligned}
&= \frac{2\xi_j}{S\sigma_j^2\tau_j\lambda_j(2\pi)^2} \int_{t=-\infty}^{0} \int_{\mathbf{x}} \exp\left(-\frac{1}{2}\Big[\frac{\log S - (\beta_{j0} + \beta_{j1}\mathbf{x} + \beta_{j2}\mathbf{z})}{\sigma_j} \Big]^2 - \frac{1}{2}\frac{\xi_j^2}{\sigma_j^2}\Big[t + \log S - (\beta_{j0} + \beta_{j1}\mathbf{x} + \beta_{j2}\mathbf{z}) \Big]^2 \right. \\
&\qquad\qquad \left. - \frac{1}{2\tau_j^2}(\mathbf{x}^* - \mathbf{x})^2 - \frac{1}{2\lambda_j^2}\big(\mathbf{x} - \{\kappa_{j0} + \kappa_{j1}\mathbf{z}\}\big)^2 \right) d\mathbf{x}\, dt
\end{aligned}$$

In the exponent above,

$$\begin{aligned}
&-\frac{1}{2}\Big[\frac{\log S - (\beta_{j0} + \beta_{j1}\mathbf{x} + \beta_{j2}\mathbf{z})}{\sigma_j} \Big]^2 - \frac{1}{2}\frac{\xi_j^2}{\sigma_j^2}\Big[t + \log S - (\beta_{j0} + \beta_{j1}\mathbf{x} + \beta_{j2}\mathbf{z}) \Big]^2 - \frac{1}{2\tau_j^2}(\mathbf{x}^* - \mathbf{x})^2 \\
&- \frac{1}{2\lambda_j^2}\big(\mathbf{x} - \{\kappa_{j0} + \kappa_{j1}\mathbf{z}\}\big)^2 = -\frac{1}{2}\left(\Big[\frac{\beta_{j1}^2}{\sigma_j^2} + \frac{\xi_j^2\beta_{j1}^2}{\sigma_j^2} + \frac{1}{\tau_j^2} + \frac{1}{\lambda_j^2} \Big]\mathbf{x}^2 - 2\Big[\frac{\beta_{j1}}{\sigma_j^2}(\log S - \beta_{j0} - \beta_{j2}\mathbf{z}) \right. \\
&\left. + \frac{\xi_j^2\beta_{j1}}{\sigma_j^2}\Big(t + \log S - \beta_{j0} - \beta_{j2}\mathbf{z} \Big) + \frac{\mathbf{x}^*}{\tau_j^2} + \frac{(\kappa_{j0} + \kappa_{j1}\mathbf{z})}{\lambda_j^2} \Big]\mathbf{x} \right) - \frac{1}{2}\left(\frac{(\log S - \beta_{j0} - \beta_{j2}\mathbf{z})^2}{\sigma_j^2} \right. \\
&\left. + \frac{\xi_j^2}{\sigma_j^2}\Big(t + \log S - \beta_{j0} - \beta_{j2}\mathbf{z} \Big)^2 + \frac{(\mathbf{x}^*)^2}{\tau_j^2} + \frac{(\kappa_{j0} + \kappa_{j1}\mathbf{z})^2}{\lambda_j^2} \right)
\end{aligned}$$

to complete the square, we introduce μ_x

$$\begin{aligned}
&-\frac{1}{2}\Big[\frac{\beta_{j1}^2}{\sigma_j^2} + \frac{\xi_j^2\beta_{j1}^2}{\sigma_j^2} + \frac{1}{\tau_j^2} + \frac{1}{\lambda_j^2} \Big]\Big(\mathbf{x}^2 - 2\mu_x\mathbf{x} + \mu_x^2 \Big) + \frac{1}{2}\Big[\frac{\beta_{j1}^2}{\sigma_j^2} + \frac{\xi_j^2\beta_{j1}^2}{\sigma_j^2} + \frac{1}{\tau_j^2} + \frac{1}{\lambda_j^2} \Big]\mu_x^2 \\
&\qquad - \frac{1}{2}\left(\frac{(\log S - \beta_{j0} - \beta_{j2}\mathbf{z})^2}{\sigma_j^2} + \frac{\xi_j^2}{\sigma_j^2}\Big(t + \log S - \beta_{j0} - \beta_{j2}\mathbf{z} \Big)^2 + \frac{(\mathbf{x}^*)^2}{\tau_j^2} + \frac{(\kappa_{j0} + \kappa_{j1}\mathbf{z})^2}{\lambda_j^2} \right)
\end{aligned}$$

where

$$\mu_x = \frac{\frac{\beta_{j1}}{\sigma_j^2}(\log S - \beta_{j0} - \beta_{j2}\mathbf{z}) + \frac{\xi_j^2\beta_{j1}}{\sigma_j^2}\Big(t + \log S - \beta_{j0} - \beta_{j2}\mathbf{z}\Big) + \frac{\mathbf{x}^*}{\tau_j^2} + \frac{(\kappa_{j0}+\kappa_{j1}\mathbf{z})}{\lambda_j^2}}{\Big[\frac{\beta_{j1}^2}{\sigma_j^2} + \frac{\xi_j^2\beta_{j1}^2}{\sigma_j^2} + \frac{1}{\tau_j^2} + \frac{1}{\lambda_j^2}\Big]}$$

therefore,

$$\begin{aligned}
&\int f(S, \mathbf{x}^*, \mathbf{x} \mid \mathbf{z})\, d\mathbf{x} \\
&= \frac{2\xi_j}{S\sigma_j^2\tau_j\lambda_j(2\pi)^2}\int_{t=-\infty}^{0} \underbrace{\int_{\mathbf{x}} \exp\left(-\frac{1}{2}\Big[\frac{\beta_{j1}^2}{\sigma_j^2} + \frac{\xi_j^2\beta_{j1}^2}{\sigma_j^2} + \frac{1}{\tau_j^2} + \frac{1}{\lambda_j^2}\Big]\Big(\mathbf{x} - \mu_x\Big)^2\right) d\mathbf{x}}_{=\sqrt{2\pi\left[\frac{\beta_{j1}^2}{\sigma_j^2} + \frac{\xi_j^2\beta_{j1}^2}{\sigma_j^2} + \frac{1}{\tau_j^2} + \frac{1}{\lambda_j^2}\right]^{-1}}} \\
&\quad \times \exp\left(\frac{1}{2}\left(\Big[\frac{\beta_{j1}^2}{\sigma_j^2} + \frac{\xi_j^2\beta_{j1}^2}{\sigma_j^2} + \frac{1}{\tau_j^2} + \frac{1}{\lambda_j^2}\Big]\mu_x^2 - \frac{(\log S - \beta_{j0} - \beta_{j2}\mathbf{z})^2}{\sigma_j^2} - \frac{\xi_j^2}{\sigma_j^2}\Big(t + \log S - \beta_{j0} - \beta_{j2}\mathbf{z}\Big)^2\right.\right. \\
&\quad \left.\left. - \frac{(\mathbf{x}^*)^2}{\tau_j^2} - \frac{(\kappa_{j0} + \kappa_{j1}\mathbf{z})^2}{\lambda_j^2}\right)\right) dt
\end{aligned}$$

In the exponent above, we can simplify as:

$$\begin{aligned}
&\frac{\left(\Big[\frac{\beta_{j1}^2}{\sigma_j^2} + \frac{\xi_j^2\beta_{j1}^2}{\sigma_j^2} + \frac{1}{\tau_j^2} + \frac{1}{\lambda_j^2}\Big]\mu_x^2 - \frac{(\log S - \beta_{j0} - \beta_{j2}\mathbf{z})^2}{\sigma_j^2} - \frac{\xi_j^2}{\sigma_j^2}\Big(t + \log S - \beta_{j0} - \beta_{j2}\mathbf{z}\Big)^2 - \frac{(\mathbf{x}^*)^2}{\tau_j^2} - \frac{(\kappa_{j0}+\kappa_{j1}\mathbf{z})^2}{\lambda_j^2}\right)}{2} \\
&= -\frac{1}{2}\left(\frac{\xi_j^2}{\sigma_j^2}\left[\frac{\frac{\beta_{j1}^2}{\sigma_j^2} + \frac{1}{\tau_j^2} + \frac{1}{\lambda_j^2}}{\Big[\frac{\beta_{j1}^2}{\sigma_j^2} + \frac{\xi_j^2\beta_{j1}^2}{\sigma_j^2} + \frac{1}{\tau_j^2} + \frac{1}{\lambda_j^2}\Big]}\right]t^2 + 2\frac{\xi_j^2}{\sigma_j^2}\left[\frac{(\frac{1}{\tau_j^2} + \frac{1}{\lambda_j^2})(\log S - \beta_{j0} - \beta_{j2}\mathbf{z}) - \frac{\beta_{j1}\mathbf{x}^*}{\tau_j^2} - \frac{\beta_{j1}(\kappa_{j0}+\kappa_{j1}\mathbf{z})}{\lambda_j^2}}{\Big[\frac{\beta_{j1}^2}{\sigma_j^2} + \frac{\xi_j^2\beta_{j1}^2}{\sigma_j^2} + \frac{1}{\tau_j^2} + \frac{1}{\lambda_j^2}\Big]}\right]t\right) \\
&+ \frac{1}{2\Big[\frac{\beta_{j1}^2}{\sigma_j^2} + \frac{\xi_j^2\beta_{j1}^2}{\sigma_j^2} + \frac{1}{\tau_j^2} + \frac{1}{\lambda_j^2}\Big]}\Big[\frac{\beta_{j1}}{\sigma_j^2}(\log S - \beta_{j0} - \beta_{j2}\mathbf{z})(1 + \xi_j^2) + \frac{\mathbf{x}^*}{\tau_j^2} + \frac{(\kappa_{j0} + \kappa_{j1}\mathbf{z})}{\lambda_j^2}\Big]^2 \\
&- \frac{1}{2}\Big[\frac{(\log S - \beta_{j0} - \beta_{j2}\mathbf{z})^2}{\sigma_j^2} + \frac{(\mathbf{x}^*)^2}{\tau_j^2} + \frac{(\kappa_{j0} + \kappa_{j1}\mathbf{z})^2}{\lambda_j^2} + \frac{\xi_j^2}{\sigma_j^2}(\log S - \beta_{j0} - \beta_{j2}\mathbf{z})^2\Big]
\end{aligned}$$

to complete the square, we introduce μ_t

$$\begin{aligned}
&- \frac{1}{2}\left(\frac{\frac{\xi_j^2}{\sigma_j^2}\Big(\frac{\beta_{j1}^2}{\sigma_j^2} + \frac{1}{\tau_j^2} + \frac{1}{\lambda_j^2}\Big)}{\Big[\frac{\beta_{j1}^2}{\sigma_j^2} + \frac{\xi_j^2\beta_{j1}^2}{\sigma_j^2} + \frac{1}{\tau_j^2} + \frac{1}{\lambda_j^2}\Big]}(t^2 + 2\mu_t t + \mu_t^2) - \frac{\frac{\xi_j^2}{\sigma_j^2}\Big(\frac{\beta_{j1}^2}{\sigma_j^2} + \frac{1}{\tau_j^2} + \frac{1}{\lambda_j^2}\Big)}{\Big[\frac{\beta_{j1}^2}{\sigma_j^2} + \frac{\xi_j^2\beta_{j1}^2}{\sigma_j^2} + \frac{1}{\tau_j^2} + \frac{1}{\lambda_j^2}\Big]}\mu_t^2\right) \\
&+ \frac{1}{2\Big[\frac{\beta_{j1}^2}{\sigma_j^2} + \frac{\xi_j^2\beta_{j1}^2}{\sigma_j^2} + \frac{1}{\tau_j^2} + \frac{1}{\lambda_j^2}\Big]}\Big[\frac{\beta_{j1}}{\sigma_j^2}(\log S - \beta_{j0} - \beta_{j2}\mathbf{z})(1 + \xi_j^2) + \frac{\mathbf{x}^*}{\tau_j^2} + \frac{(\kappa_{j0} + \kappa_{j1}\mathbf{z})}{\lambda_j^2}\Big]^2 \\
&- \frac{1}{2}\Big[\frac{(1 + \xi_j^2)(\log S - \beta_{j0} - \beta_{j2}\mathbf{z})^2}{\sigma_j^2} + \frac{(\mathbf{x}^*)^2}{\tau_j^2} + \frac{(\kappa_{j0} + \kappa_{j1}\mathbf{z})^2}{\lambda_j^2}\Big]
\end{aligned}$$

where

$$\mu_t = \left[\frac{(\frac{1}{\tau_j^2} + \frac{1}{\lambda_j^2})(\log S - \beta_{j0} - \beta_{j2}\mathbf{z}) - \frac{\beta_{j1}\mathbf{x}^*}{\tau_j^2} - \frac{\beta_{j1}(\kappa_{j0}+\kappa_{j1}\mathbf{z})}{\lambda_j^2}}{\frac{\beta_{j1}^2}{\sigma_j^2} + \frac{1}{\tau_j^2} + \frac{1}{\lambda_j^2}}\right]$$

therefore,

$$\int f(S,\mathbf{x}^*,\mathbf{x}\mid\mathbf{z})\,d\mathbf{x}$$

$$= \frac{2\xi_j}{S\sigma_j^2\tau_j\lambda_j(2\pi)^2}\sqrt{2\pi\Big[\frac{\beta_{j1}^2}{\sigma_j^2}+\frac{\xi_j^2\beta_{j1}^2}{\sigma_j^2}+\frac{1}{\tau_j^2}+\frac{1}{\lambda_j^2}\Big]^{-1}} \times \exp\left(\frac{1}{2}\frac{\frac{\xi_j^2}{\sigma_j^2}\left(\frac{\beta_{j1}^2}{\sigma_j^2}+\frac{1}{\tau_j^2}+\frac{1}{\lambda_j^2}\right)}{\Big[\frac{\beta_{j1}^2}{\sigma_j^2}+\frac{\xi_j^2\beta_{j1}^2}{\sigma_j^2}+\frac{1}{\tau_j^2}+\frac{1}{\lambda_j^2}\Big]}\mu_t^2\right.$$

$$+\frac{1}{2\Big[\frac{\beta_{j1}^2}{\sigma_j^2}+\frac{\xi_j^2\beta_{j1}^2}{\sigma_j^2}+\frac{1}{\tau_j^2}+\frac{1}{\lambda_j^2}\Big]}\Big[\frac{\beta_{j1}}{\sigma_j^2}(\log S-\beta_{j0}-\beta_{j2}\mathbf{z})(1+\xi_j^2)+\frac{\mathbf{x}^*}{\tau_j^2}+\frac{(\kappa_{j0}+\kappa_{j1}\mathbf{z})}{\lambda_j^2}\Big]^2$$

$$\left.-\frac{1}{2}\Big[\frac{(1+\xi_j^2)(\log S-\beta_{j0}-\beta_{j2}\mathbf{z})^2}{\sigma_j^2}+\frac{(\mathbf{x}^*)^2}{\tau_j^2}+\frac{(\kappa_{j0}+\kappa_{j1}\mathbf{z})^2}{\lambda_j^2}\Big]\right)$$

$$\times \underbrace{\int_{t=-\infty}^{0}\exp\left(-\frac{1}{2}\Big[\frac{\frac{\xi_j^2}{\sigma_j^2}\left(\frac{\beta_{j1}^2}{\sigma_j^2}+\frac{1}{\tau_j^2}+\frac{1}{\lambda_j^2}\right)}{\Big[\frac{\beta_{j1}^2}{\sigma_j^2}+\frac{\xi_j^2\beta_{j1}^2}{\sigma_j^2}+\frac{1}{\tau_j^2}+\frac{1}{\lambda_j^2}\Big]}\Big](t+\mu_t)^2\right)dt}$$

$$= \int_{w=-\infty}^{f}\exp\{-\frac{1}{2}w^2\}\,dw \;\times\; f^{-1}\mu_t \;=\; \sqrt{2\pi}\times f^{-1}\mu_t\times\Phi(f), \qquad \text{where}$$

$$w \;=\; \frac{\xi_j}{\sigma_j}\left(\frac{\frac{\beta_{j1}^2}{\sigma_j^2}+\frac{1}{\tau_j^2}+\frac{1}{\lambda_j^2}}{\frac{\beta_{j1}^2}{\sigma_j^2}+\frac{\xi_j^2\beta_{j1}^2}{\sigma_j^2}+\frac{1}{\tau_j^2}+\frac{1}{\lambda_j^2}}\right)^{1/2}(t+\mu_t),$$

$$f \;=\; \mu_t\times\frac{\xi_j}{\sigma_j}\left(\frac{\frac{\beta_{j1}^2}{\sigma_j^2}+\frac{1}{\tau_j^2}+\frac{1}{\lambda_j^2}}{\frac{\beta_{j1}^2}{\sigma_j^2}+\frac{\xi_j^2\beta_{j1}^2}{\sigma_j^2}+\frac{1}{\tau_j^2}+\frac{1}{\lambda_j^2}}\right)^{1/2}$$

$$= \frac{2}{S\sigma_j\tau_j\lambda_j 2\pi}\Big[\frac{\beta_{j1}^2}{\sigma_j^2}+\frac{1}{\tau_j^2}+\frac{1}{\lambda_j^2}\Big]^{-\frac{1}{2}}\times\exp\left(\frac{1}{2}\frac{\frac{\xi_j}{\sigma_j^2}\left(\frac{\beta_{j1}^2}{\sigma_j^2}+\frac{1}{\tau_j^2}+\frac{1}{\lambda_j^2}\right)}{\Big[\frac{\beta_{j1}^2}{\sigma_j^2}+\frac{\xi_j^2\beta_{j1}^2}{\sigma_j^2}+\frac{1}{\tau_j^2}+\frac{1}{\lambda_j^2}\Big]}\mu_t^2+\frac{1}{2\Big[\frac{\beta_{j1}^2}{\sigma_j^2}+\frac{\xi_j^2\beta_{j1}^2}{\sigma_j^2}+\frac{1}{\tau_j^2}+\frac{1}{\lambda_j^2}\Big]}\right.$$

$$\Big[\frac{\beta_{j1}}{\sigma_j^2}(\log S-\beta_{j0}-\beta_{j2}\mathbf{z})(1+\xi_j^2)+\frac{\mathbf{x}^*}{\tau_j^2}+\frac{(\kappa_{j0}+\kappa_{j1}\mathbf{z})}{\lambda_j^2}\Big]^2-\frac{1}{2}\Big[\frac{(1+\xi_j^2)(\log S-\beta_{j0}-\beta_{j2}\mathbf{z})^2}{\sigma_j^2}+\frac{(\mathbf{x}^*)^2}{\tau_j^2}$$

$$\left.+\frac{(\kappa_{j0}+\kappa_{j1}\mathbf{z})^2}{\lambda_j^2}\Big]\right)\times\Phi(f) \qquad \text{.. This is Solution I.}$$

and the resulting form above (Solution I.) should square with the incomplete model:

$$f(S,\mathbf{x}^*\mid\mathbf{z}) = \frac{2}{\widehat{\sigma}_j S\sqrt{2\pi}}\underbrace{\exp\left(-\frac{1}{2}\Big[\frac{\log S-(\widehat{\beta}_{j0}+\widehat{\beta}_{j1}\mathbf{x}^*+\widehat{\beta}_{j2}\mathbf{z})}{\widehat{\sigma}_j}\Big]^2\right)}_{\phi(\cdot)}\times\underbrace{\int_{-\infty}^{\widehat{\xi}_j\frac{\log S-(\widehat{\beta}_{j0}+\widehat{\beta}_{j1}\mathbf{x}^*+\widehat{\beta}_{j2}\mathbf{z})}{\widehat{\sigma}_j}}\frac{1}{\sqrt{2\pi}}e^{-\frac{u^2}{2}}\,du}_{\Phi(\cdot)}$$

$$\times\frac{1}{\sqrt{2\pi\widehat{\lambda}_j^2}}\exp\left(\frac{-(\mathbf{x}^*-\{\widehat{\kappa}_{j0}+\widehat{\kappa}_{j1}\mathbf{z}\})^2}{2\widehat{\lambda}_j^2}\right)$$

Since $\Phi(f) = \Phi(\widehat{\xi}_j \frac{\log S - (\widehat{\beta}_{j0} + \widehat{\beta}_{j1}\mathbf{x}^* + \widehat{\beta}_{j2}\mathbf{z})}{\widehat{\sigma}_j})$,

$$
\begin{aligned}
f &= \left[\frac{(\frac{1}{\tau_j^2} + \frac{1}{\lambda_j^2})(\log S - \beta_{j0} - \beta_{j2}\mathbf{z}) - \frac{\beta_{j1}\mathbf{x}^*}{\tau_j^2} - \frac{\beta_{j1}(\kappa_{j0}+\kappa_{j1}\mathbf{z})}{\lambda_j^2}}{\frac{\beta_{j1}^2}{\sigma_j^2} + \frac{1}{\tau_j^2} + \frac{1}{\lambda_j^2}} \right] \times \frac{\xi_j}{\sigma_j} \left(\frac{\frac{\beta_{j1}^2}{\sigma_j^2} + \frac{1}{\tau_j^2} + \frac{1}{\lambda_j^2}}{\frac{\beta_{j1}^2}{\sigma_j^2} + \frac{\xi_j^2\beta_{j1}^2}{\sigma_j^2} + \frac{1}{\tau_j^2} + \frac{1}{\lambda_j^2}} \right)^{1/2} \\
&= \frac{\frac{\xi_j}{\sigma_j}\left(\frac{1}{\tau_j^2} + \frac{1}{\lambda_j^2}\right)\left[(\log S - \beta_{j0} - \beta_{j2}\mathbf{z}) - \frac{\beta_{j1}\mathbf{x}^*}{\tau_j^2}\left(\frac{1}{\tau_j^2} + \frac{1}{\lambda_j^2}\right)^{-1} - \frac{\beta_{j1}(\kappa_{j0}+\kappa_{j1}\mathbf{z})}{\lambda_j^2}\left(\frac{1}{\tau_j^2} + \frac{1}{\lambda_j^2}\right)^{-1}\right]}{\left(\frac{\beta_{j1}^2}{\sigma_j^2} + \frac{1}{\tau_j^2} + \frac{1}{\lambda_j^2}\right)^{1/2}\left(\frac{\beta_{j1}^2}{\sigma_j^2} + \frac{\xi_j^2\beta_{j1}^2}{\sigma_j^2} + \frac{1}{\tau_j^2} + \frac{1}{\lambda_j^2}\right)^{1/2}} \\
&= \widehat{\xi}_j \frac{\log S - (\widehat{\beta}_{j0} + \widehat{\beta}_{j1}\mathbf{x}^* + \widehat{\beta}_{j2}\mathbf{z})}{\widehat{\sigma}_j}
\end{aligned}
$$

thus we can obtain $\widehat{\beta}_{j0}, \widehat{\beta}_{j1}, \widehat{\beta}_{j2}$ as follows:

$$
\begin{aligned}
&\log S - \beta_{j0} - \beta_{j2}\mathbf{z} - \frac{\beta_{j1}\mathbf{x}^*}{\tau_j^2}\left(\frac{1}{\tau_j^2} + \frac{1}{\lambda_j^2}\right)^{-1} - \frac{\beta_{j1}(\kappa_{j0} + \kappa_{j1}\mathbf{z})}{\lambda_j^2}\left(\frac{1}{\tau_j^2} + \frac{1}{\lambda_j^2}\right)^{-1} \\
&\qquad = \log S - \underbrace{\left(\beta_{j0} + \frac{\beta_{j1}\kappa_{j0}\tau_j^2}{\lambda_j^2 + \tau_j^2}\right)}_{=\widehat{\beta}_{j0}} - \underbrace{\frac{\beta_{j1}\lambda_j^2}{\lambda_j^2 + \tau_j^2}}_{=\widehat{\beta}_{j1}}\mathbf{x}^* - \underbrace{\left(\beta_{j2} + \frac{\beta_{j1}\kappa_{j1}\tau_j^2}{\lambda_j^2 + \tau_j^2}\right)}_{=\widehat{\beta}_{j2}}\mathbf{z}
\end{aligned}
$$

Accordingly,

$$
f = \frac{\frac{\xi_j}{\sigma_j}\left(\frac{1}{\tau_j^2} + \frac{1}{\lambda_j^2}\right)\left[\log S - \widehat{\beta}_{j0} - \widehat{\beta}_{j1}\mathbf{x}^* - \widehat{\beta}_{j2}\mathbf{z}\right]}{\left(\frac{\beta_{j1}^2}{\sigma_j^2} + \frac{1}{\tau_j^2} + \frac{1}{\lambda_j^2}\right)^{1/2}\left(\frac{\beta_{j1}^2}{\sigma_j^2} + \frac{\xi_j^2\beta_{j1}^2}{\sigma_j^2} + \frac{1}{\tau_j^2} + \frac{1}{\lambda_j^2}\right)^{1/2}}
$$

then

$$
\frac{\widehat{\xi}_j}{\widehat{\sigma}_j} = \frac{\frac{\xi_j}{\sigma_j}\left(\frac{1}{\tau_j^2} + \frac{1}{\lambda_j^2}\right)}{\left(\frac{\beta_{j1}^2}{\sigma_j^2} + \frac{1}{\tau_j^2} + \frac{1}{\lambda_j^2}\right)^{1/2}\left(\frac{\beta_{j1}^2}{\sigma_j^2} + \frac{\xi_j^2\beta_{j1}^2}{\sigma_j^2} + \frac{1}{\tau_j^2} + \frac{1}{\lambda_j^2}\right)^{1/2}}
$$

For $\widehat{\sigma_j}, \widehat{\xi_j}, \widehat{\lambda_j}, \widehat{\kappa}_{j0}, \widehat{\kappa}_{j1}$, we take following procedure. First, inside of the exponent of Solution I., μ_t is substituted, one can find the common factor "$\Big[\log S - \beta_{j0} - \beta_{j2}\mathbf{z}\Big]$" from below:

$$
\begin{aligned}
&\frac{1}{2}\frac{\frac{\xi_j}{\sigma_j}\Big(\frac{\beta_{j1}^2}{\sigma_j^2}+\frac{1}{\tau_j^2}+\frac{1}{\lambda_j^2}\Big)}{\Big[\frac{\beta_{j1}^2}{\sigma_j^2}+\frac{\xi_j^2\beta_{j1}^2}{\sigma_j^2}+\frac{1}{\tau_j^2}+\frac{1}{\lambda_j^2}\Big]}\mu_t^2+\frac{1}{2\Big[\frac{\beta_{j1}^2}{\sigma_j^2}+\frac{\xi_j^2\beta_{j1}^2}{\sigma_j^2}+\frac{1}{\tau_j^2}+\frac{1}{\lambda_j^2}\Big]}\Big[\frac{\beta_{j1}}{\sigma_j^2}(\log S-\beta_{j0}-\beta_{j2}\mathbf{z})(1+\xi_j^2)\\
&+\frac{\mathbf{x}^*}{\tau_j^2}+\frac{(\kappa_{j0}+\kappa_{j1}\mathbf{z})}{\lambda_j^2}\Big]^2-\frac{1}{2}\Big[\frac{(1+\xi_j^2)(\log S-\beta_{j0}-\beta_{j2}\mathbf{z})^2}{\sigma_j^2}+\frac{(\mathbf{x}^*)^2}{\tau_j^2}+\frac{(\kappa_{j0}+\kappa_{j1}\mathbf{z})^2}{\lambda_j^2}\Big]\\
&=\frac{-1}{2\Big(\frac{\beta_{j1}^2}{\sigma_j^2}+\frac{\xi_j^2\beta_{j1}^2}{\sigma_j^2}+\frac{1}{\tau_j^2}+\frac{1}{\lambda_j^2}\Big)\Big(\frac{\beta_{j1}^2}{\sigma_j^2}+\frac{1}{\tau_j^2}+\frac{1}{\lambda_j^2}\Big)}\times\Bigg[\Big(\frac{\beta_{j1}^4}{\sigma_j^4}+\frac{2\beta_{j1}^2}{\sigma_j^2\tau_j^2}+\frac{2\beta_{1j}^2}{\sigma_j^2\lambda_j^2}+\frac{2}{\tau_j^2\lambda_j^2}+\frac{\xi_j^2\beta_{j1}^4}{\sigma_j^4}+\frac{\xi_j^2\beta_{j1}^2}{\sigma_j^2\tau_j^2}\\
&\quad+\frac{\xi_j^2\beta_{j1}^2}{\sigma_j^2\lambda_j^2}+\frac{1}{\tau_j^4}+\frac{1}{\lambda_j^4}\Big)\Big(\frac{(1+\xi_{j1}^2)(\log S-\beta_{j0}-\beta_{j2}\mathbf{z})^2}{\sigma_j^2}+\frac{(\mathbf{x}^*)^2}{\tau_j^2}+\frac{(\kappa_{j0}+\kappa_{j1}\mathbf{z})^2}{\lambda_j^2}\Big)\\
&\quad-\frac{\xi_j^2}{\sigma_j^2}\Big(\frac{1}{\tau_j^4}+\frac{2}{\tau_j^2\lambda_j^2}+\frac{1}{\lambda_j^4}\Big)(\log S-\beta_{j0}-\beta_{j2}\mathbf{z})^2-\frac{\xi_j^2}{\sigma_j^2}\frac{\beta_{j1}^2(\mathbf{x}^*)^2}{\tau_j^4}-\frac{\xi_j^2}{\sigma_j^2}\frac{\beta_{j1}^2(\kappa_0+\kappa_1\mathbf{z})^2}{\lambda_j^4}\\
&\quad+2\frac{\xi_j^2}{\sigma_j^2}\Big(\frac{1}{\tau_j^2}+\frac{1}{\lambda_j^2}\Big)\Big(\frac{\beta_{j1}\mathbf{x}^*}{\tau_j^2}+\frac{\beta_{j1}(\kappa_0+\kappa_1\mathbf{z})}{\lambda_2}\Big)(\log S-\beta_{j0}-\beta_{j2}\mathbf{z})-\frac{\xi_j^2}{\sigma_j^2}\frac{2\beta_{j1}^2\mathbf{x}^*(\kappa_0+\kappa_1\mathbf{z})}{\tau_j^2\lambda_j^2}\\
&\quad-\Big(\frac{\beta_{j1}^2}{\sigma_j^2}+\frac{1}{\tau_j^2}+\frac{1}{\lambda_j^2}\Big)\Bigg(\frac{\beta_{j1}^2(1+\xi_j^2)^2}{\sigma_j^4}(\log S-\beta_{j0}-\beta_{j2}\mathbf{z})^2\\
&\qquad+\frac{2\beta_{j1}(1+\xi_{j1}^2)}{\sigma_j^2}\Big(\frac{\mathbf{x}^*}{\tau_j^2}+\frac{\kappa_0+\kappa_1\mathbf{z}}{\lambda_j^2}\Big)(\log S-\beta_{j0}-\beta_{j2}\mathbf{z})+\frac{(\mathbf{x}^*)^2}{\tau_j^4}+\frac{(\kappa_0+\kappa_1\mathbf{z})^2}{\lambda_j^4}+\frac{2\mathbf{x}^*(\kappa_0+\kappa_1\mathbf{z})}{\tau_j^2\lambda_j^2}\Bigg)\Bigg]
\end{aligned}
$$

If tidy up this with respect to "$\Big[\log S - \beta_{j0} - \beta_{j2}\mathbf{z}\Big]$",

$$
\begin{aligned}
&=\frac{-1}{2\Big(\frac{\beta_{j1}^2}{\sigma_j^2}+\frac{\xi_j^2\beta_{j1}^2}{\sigma_j^2}+\frac{1}{\tau_j^2}+\frac{1}{\lambda_j^2}\Big)\Big(\frac{\beta_{j1}^2}{\sigma_j^2}+\frac{1}{\tau_j^2}+\frac{1}{\lambda_j^2}\Big)}\times\Bigg[\\
&\quad\frac{1}{\sigma_j^2}\Big(\frac{1}{\tau_j^2}+\frac{1}{\lambda_j^2}\Big)\Big(\frac{\beta_{j1}^2}{\sigma_j^2}+\frac{\xi_j^2\beta_{j1}^2}{\sigma_j^2}+\frac{1}{\tau_j^2}+\frac{1}{\lambda_j^2}\Big)\Big[\log S-\beta_{j0}-\beta_{j2}\mathbf{z}\Big]^2\\
&\quad-2\frac{\beta_{j1}}{\sigma_j^2}\Big(\frac{\mathbf{x}^*}{\tau_j^2}+\frac{\kappa_0+\kappa_1\mathbf{z}}{\lambda_j^2}\Big)\Big(\frac{\beta_{j1}^2}{\sigma_j^2}+\frac{\xi_j^2\beta_{j1}^2}{\sigma_j^2}+\frac{1}{\tau_j^2}+\frac{1}{\lambda_j^2}\Big)\Big[\log S-\beta_{j0}-\beta_{j2}\mathbf{z}\Big]\\
&\quad+\Big(\frac{\beta_{j1}^4}{\sigma_j^4}+\frac{2\beta_{j1}^2}{\sigma_j^2\tau_j^2}+\frac{2\beta_{1j}^2}{\sigma_j^2\lambda_j^2}+\frac{2}{\tau_j^2\lambda_j^2}+\frac{\xi_j^2\beta_{j1}^4}{\sigma_j^4}+\frac{\xi_j^2\beta_{j1}^2}{\sigma_j^2\tau_j^2}+\frac{\xi_j^2\beta_{j1}^2}{\sigma_j^2\lambda_j^2}+\frac{1}{\tau_j^4}+\frac{1}{\lambda_j^4}\Big)\Big(\frac{(\mathbf{x}^*)^2}{\tau_j^2}+\frac{(\kappa_{j0}+\kappa_{j1}\mathbf{z})^2}{\lambda_j^2}\Big)\\
&\quad-\frac{\xi_j^2}{\sigma_j^2}\frac{\beta_{j1}^2(\mathbf{x}^*)^2}{\tau_j^4}-\frac{\xi_j^2}{\sigma_j^2}\frac{\beta_{j1}^2(\kappa_0+\kappa_1\mathbf{z})^2}{\lambda_j^4}-\frac{\xi_j^2}{\sigma_j^2}\frac{2\beta_{j1}^2\mathbf{x}^*(\kappa_0+\kappa_1\mathbf{z})}{\tau_j^2\lambda_j^2}\\
&\quad-\Big(\frac{\beta_{j1}^2}{\sigma_j^2}+\frac{1}{\tau_j^2}+\frac{1}{\lambda_j^2}\Big)\Big(\frac{(\mathbf{x}^*)^2}{\tau_j^4}+\frac{(\kappa_0+\kappa_1\mathbf{z})^2}{\lambda_j^4}+\frac{2\mathbf{x}^*(\kappa_0+\kappa_1\mathbf{z})}{\tau_j^2\lambda_j^2}\Big)\Bigg]
\end{aligned}
$$

To complete the square, we split the expression above into (A) and (B) as follows:

$$
\begin{aligned}
= \; & \frac{-1}{2\Big(\frac{\beta_{j1}^2}{\sigma_j^2}+\frac{\xi_j^2\beta_{j1}^2}{\sigma_j^2}+\frac{1}{\tau_j^2}+\frac{1}{\lambda_j^2}\Big)\Big(\frac{\beta_{j1}^2}{\sigma_j^2}+\frac{1}{\tau_j^2}+\frac{1}{\lambda_j^2}\Big)} \times \Bigg[\\
& \frac{1}{\sigma_j^2}\Big(\frac{1}{\tau_j^2}+\frac{1}{\lambda_j^2}\Big)\Big(\frac{\beta_{j1}^2}{\sigma_j^2}+\frac{\xi_j^2\beta_{j1}^2}{\sigma_j^2}+\frac{1}{\tau_j^2}+\frac{1}{\lambda_j^2}\Big)\Big[\log S-\beta_{j0}-\beta_{j2}\mathbf{z}\Big]^2 \\
& -2\frac{\beta_{j1}}{\sigma_j^2}\Big(\frac{\mathbf{x}^*}{\tau_j^2}+\frac{\kappa_0+\kappa_1\mathbf{z}}{\lambda_j^2}\Big)\Big(\frac{\beta_{j1}^2}{\sigma_j^2}+\frac{\xi_j^2\beta_{j1}^2}{\sigma_j^2}+\frac{1}{\tau_j^2}+\frac{1}{\lambda_j^2}\Big)\Big[\log S-\beta_{j0}-\beta_{j2}\mathbf{z}\Big]\Bigg] \ldots\ldots \text{ (A)} \\
& + \frac{-1}{2\Big(\frac{\beta_{j1}^2}{\sigma_j^2}+\frac{\xi_j^2\beta_{j1}^2}{\sigma_j^2}+\frac{1}{\tau_j^2}+\frac{1}{\lambda_j^2}\Big)\Big(\frac{\beta_{j1}^2}{\sigma_j^2}+\frac{1}{\tau_j^2}+\frac{1}{\lambda_j^2}\Big)} \times \Bigg[\\
& \Big(\frac{\beta_{j1}^4}{\sigma_j^4}+\frac{2\beta_{j1}^2}{\sigma_j^2\tau_j^2}+\frac{2\beta_{1j}^2}{\sigma_j^2\lambda_j^2}+\frac{2}{\tau_j^2\lambda_j^2}+\frac{\xi_j^2\beta_{j1}^4}{\sigma_j^4}+\frac{\xi_j^2\beta_{j1}^2}{\sigma_j^2\tau_j^2}+\frac{\xi_j^2\beta_{j1}^2}{\sigma_j^2\lambda_j^2}+\frac{1}{\tau_j^4}+\frac{1}{\lambda_j^4}\Big)\Big(\frac{(\mathbf{x}^*)^2}{\tau_j^2}+\frac{(\kappa_{j0}+\kappa_{j1}\mathbf{z})^2}{\lambda_j^2}\Big) \\
& -\frac{\xi_j^2}{\sigma_j^2}\frac{\beta_{j1}^2(\mathbf{x}^*)^2}{\tau_j^4}-\frac{\xi_j^2}{\sigma_j^2}\frac{\beta_{j1}^2(\kappa_0+\kappa_1\mathbf{z})^2}{\lambda_j^4}-\frac{\xi_j^2}{\sigma_j^2}\frac{2\beta_{j1}^2\mathbf{x}^*(\kappa_0+\kappa_1\mathbf{z})}{\tau_j^2\lambda_j^2} \\
& -\Big(\frac{\beta_{j1}^2}{\sigma_j^2}+\frac{1}{\tau_j^2}+\frac{1}{\lambda_j^2}\Big)\Big(\frac{(\mathbf{x}^*)^2}{\tau_j^4}+\frac{(\kappa_0+\kappa_1\mathbf{z})^2}{\lambda_j^4}+\frac{2\mathbf{x}^*(\kappa_0+\kappa_1\mathbf{z})}{\tau_j^2\lambda_j^2}\Big)\Bigg] \ldots\ldots\ldots\ldots\ldots\ldots \text{ (B)}
\end{aligned}
$$

If tidy up (A), the square can be made:

$$
\underbrace{\frac{-\frac{1}{\sigma_j^2}\Big(\frac{1}{\tau_j^2}+\frac{1}{\lambda_j^2}\Big)}{2\Big(\frac{\beta_{j1}^2}{\sigma_j^2}+\frac{1}{\tau_j^2}+\frac{1}{\lambda_j^2}\Big)}\Big(\log S-\big(\widehat{\beta}_{j0}+\widehat{\beta}_{j1}\mathbf{x}^*+\widehat{\beta}_{j2}\mathbf{z}\big)\Big)^2}_{\text{from } \phi(\exp\{\cdot\})}+\frac{\frac{1}{\sigma_j^2}\Big(\frac{1}{\tau_j^2}+\frac{1}{\lambda_j^2}\Big)}{2\Big(\frac{\beta_{j1}^2}{\sigma_j^2}+\frac{1}{\tau_j^2}+\frac{1}{\lambda_j^2}\Big)}\Bigg(\frac{\beta_{j1}\Big(\frac{\mathbf{x}^*}{\tau_j^2}+\frac{\kappa_{j0}+\kappa_{j1}\mathbf{z}}{\lambda_j^2}\Big)}{\frac{1}{\tau_j^2}+\frac{1}{\lambda_j^2}}\Bigg)^2
$$

and $\phi(\exp\{\cdot\})$ above describes the exponent term of the Gaussian pdf (component of our Log-skew normal outcome), which is $-\frac{1}{2\widehat{\sigma}_j^2}\Big[\log S-(\widehat{\beta}_{j0}+\widehat{\beta}_{j1}\mathbf{x}^*+\widehat{\beta}_{j2}\mathbf{z})\Big]^2$; therefore,

$$
\widehat{\sigma}_j^2=\Bigg(\frac{\frac{1}{\sigma_j^2}\Big(\frac{1}{\tau_j^2}+\frac{1}{\lambda_j^2}\Big)}{\frac{\beta_{j1}^2}{\sigma_j^2}+\frac{1}{\tau_j^2}+\frac{1}{\lambda_j^2}}\Bigg)^{-1}=\frac{\sigma_j^2\Big(\frac{\beta_{j1}^2}{\sigma_j^2}+\frac{1}{\tau_j^2}+\frac{1}{\lambda_j^2}\Big)}{\frac{1}{\tau_j^2}+\frac{1}{\lambda_j^2}}
$$

Additionally, given that $\dfrac{\widehat{\xi}_j^2}{\widehat{\sigma}_j^2}=\dfrac{\frac{\xi_j^2}{\sigma_j^2}\Big(\frac{1}{\tau_j^2}+\frac{1}{\lambda_j^2}\Big)^2}{\Big(\frac{\beta_{j1}^2}{\sigma_j^2}+\frac{1}{\tau_j^2}+\frac{1}{\lambda_j^2}\Big)\Big(\frac{\beta_{j1}^2}{\sigma_j^2}+\frac{\xi_j^2\beta_{j1}^2}{\sigma_j^2}+\frac{1}{\tau_j^2}+\frac{1}{\lambda_j^2}\Big)}$

$$
\widehat{\xi}_j^2=\frac{\xi_j^2\Big(\frac{1}{\tau_j^2}+\frac{1}{\lambda_j^2}\Big)}{\frac{\beta_{j1}^2}{\sigma_j^2}+\frac{\xi_j^2\beta_{j1}^2}{\sigma_j^2}+\frac{1}{\tau_j^2}+\frac{1}{\lambda_j^2}}
$$

As for the remainder, we have

$$\frac{\frac{1}{\sigma_j^2}\left(\frac{1}{\tau_j^2}+\frac{1}{\lambda_j^2}\right)}{2\left(\frac{\beta_{j1}^2}{\sigma_j^2}+\frac{1}{\tau_j^2}+\frac{1}{\lambda_j^2}\right)}\left(\frac{\beta_{j1}\left(\frac{\mathbf{x}^*}{\tau_j^2}+\frac{\kappa_{j0}+\kappa_{j1}\mathbf{z}}{\lambda_j^2}\right)}{\frac{1}{\tau_j^2}+\frac{1}{\lambda_j^2}}\right)^2+\frac{-1}{2\left(\frac{\beta_{j1}^2}{\sigma_j^2}+\frac{\xi_j^2\beta_{j1}^2}{\sigma_j^2}+\frac{1}{\tau_j^2}+\frac{1}{\lambda_j^2}\right)\left(\frac{\beta_{j1}^2}{\sigma_j^2}+\frac{1}{\tau_j^2}+\frac{1}{\lambda_j^2}\right)}\Bigg[$$
$$\left(\frac{\beta_{j1}^4}{\sigma_j^4}+\frac{2\beta_{j1}^2}{\sigma_j^2\tau_j^2}+\frac{2\beta_{1j}^2}{\sigma_j^2\lambda_j^2}+\frac{2}{\tau_j^2\lambda_j^2}+\frac{\xi_j^2\beta_{j1}^4}{\sigma_j^4}+\frac{\xi_j^2\beta_{j1}^2}{\sigma_j^2\tau_j^2}+\frac{\xi_j^2\beta_{j1}^2}{\sigma_j^2\lambda_j^2}+\frac{1}{\tau_j^4}+\frac{1}{\lambda_j^4}\right)\left(\frac{(\mathbf{x}^*)^2}{\tau_j^2}+\frac{(\kappa_{j0}+\kappa_{j1}\mathbf{z})^2}{\lambda_j^2}\right)$$
$$-\frac{\xi_j^2}{\sigma_j^2}\frac{\beta_{j1}^2(\mathbf{x}^*)^2}{\tau_j^4}-\frac{\xi_j^2}{\sigma_j^2}\frac{\beta_{j1}^2(\kappa_0+\kappa_1\mathbf{z})^2}{\lambda_j^4}-\frac{\xi_j^2}{\sigma_j^2}\frac{2\beta_{j1}^2\mathbf{x}^*(\kappa_0+\kappa_1\mathbf{z})}{\tau_j^2\lambda_j^2}$$
$$-\left(\frac{\beta_{j1}^2}{\sigma_j^2}+\frac{1}{\tau_j^2}+\frac{1}{\lambda_j^2}\right)\left(\frac{(\mathbf{x}^*)^2}{\tau_j^4}+\frac{(\kappa_0+\kappa_1\mathbf{z})^2}{\lambda_j^4}+\frac{2\mathbf{x}^*(\kappa_0+\kappa_1\mathbf{z})}{\tau_j^2\lambda_j^2}\right)\Bigg] \quad \text{..... (B)}$$

If tidy up the above,

$$=\frac{\frac{1}{\sigma_j^2}\left(\frac{1}{\tau_j^2}+\frac{1}{\lambda_j^2}\right)}{2\left(\frac{\beta_{j1}^2}{\sigma_j^2}+\frac{1}{\tau_j^2}+\frac{1}{\lambda_j^2}\right)}\left(\frac{\beta_{j1}\left(\frac{\mathbf{x}^*}{\tau_j^2}+\frac{\kappa_{j0}+\kappa_{j1}\mathbf{z}}{\lambda_j^2}\right)}{\frac{1}{\tau_j^2}+\frac{1}{\lambda_j^2}}\right)^2+\frac{-1}{2\left(\frac{\beta_{j1}^2}{\sigma_j^2}+\frac{\xi_j^2\beta_{j1}^2}{\sigma_j^2}+\frac{1}{\tau_j^2}+\frac{1}{\lambda_j^2}\right)\left(\frac{\beta_{j1}^2}{\sigma_j^2}+\frac{1}{\tau_j^2}+\frac{1}{\lambda_j^2}\right)}\Bigg[$$
$$+\left(\frac{\beta_{j1}^2}{\sigma_j^2}+\frac{\xi_j^2\beta_{j1}^2}{\sigma_j^2}+\frac{1}{\tau_j^2}+\frac{1}{\lambda_j^2}\right)\left(\frac{\beta_{j1}^2}{\sigma_j^2}+\frac{1}{\tau_j^2}+\frac{1}{\lambda_j^2}\right)\left(\frac{(\mathbf{x}^*)^2}{\tau_j^2}+\frac{(\kappa_{j0}+\kappa_{j1}\mathbf{z})^2}{\lambda_j^2}\right)$$
$$-\left(\frac{\beta_{j1}^2}{\sigma_j^2}+\frac{\xi_j^2\beta_{j1}^2}{\sigma_j^2}+\frac{1}{\tau_j^2}+\frac{1}{\lambda_j^2}\right)\left(\frac{\mathbf{x}^*}{\tau_j^2}+\frac{\kappa_{j0}+\kappa_{j1}\mathbf{z}}{\lambda_j^2}\right)^2\Bigg]$$
$$=\frac{\frac{1}{\sigma_j^2}\left(\frac{1}{\tau_j^2}+\frac{1}{\lambda_j^2}\right)}{2\left(\frac{\beta_{j1}^2}{\sigma_j^2}+\frac{1}{\tau_j^2}+\frac{1}{\lambda_j^2}\right)}\left(\frac{\beta_{j1}\left(\frac{\mathbf{x}^*}{\tau_j^2}+\frac{\kappa_{j0}+\kappa_{j1}\mathbf{z}}{\lambda_j^2}\right)}{\frac{1}{\tau_j^2}+\frac{1}{\lambda_j^2}}\right)^2-\frac{1}{2\left(\frac{\beta_{j1}^2}{\sigma_j^2}+\frac{1}{\tau_j^2}+\frac{1}{\lambda_j^2}\right)}\Bigg[$$
$$\left(\frac{\beta_{j1}^2}{\sigma_j^2}+\frac{1}{\tau_j^2}+\frac{1}{\lambda_j^2}\right)\left(\frac{(\mathbf{x}^*)^2}{\tau_j^2}+\frac{(\kappa_{j0}+\kappa_{j1}\mathbf{z})^2}{\lambda_j^2}\right)-\left(\frac{\mathbf{x}^*}{\tau_j^2}+\frac{\kappa_{j0}+\kappa_{j1}\mathbf{z}}{\lambda_j^2}\right)^2\Bigg]$$
$$=\frac{-1}{2\left(\frac{\beta_{j1}^2}{\sigma_j^2}+\frac{1}{\tau_j^2}+\frac{1}{\lambda_j^2}\right)}\left[\left(\frac{\beta_{j1}^2}{\sigma_j^2}+\frac{1}{\tau_j^2}+\frac{1}{\lambda_j^2}\right)\left(\frac{(\mathbf{x}^*)^2}{\tau_j^2}+\frac{(\kappa_{j0}+\kappa_{j1}\mathbf{z})^2}{\lambda_j^2}\right)-\left(\frac{\frac{\beta_{j1}^2}{\sigma_j^2}+\frac{1}{\tau_j^2}+\frac{1}{\lambda_j^2}}{\frac{1}{\tau_j^2}+\frac{1}{\lambda_j^2}}\right)\left(\frac{\mathbf{x}^*}{\tau_j^2}+\frac{\kappa_{j0}+\kappa_{j1}\mathbf{z}}{\lambda_j^2}\right)^2\right]$$
$$=\frac{-1}{2\left(\frac{1}{\tau_j^2}+\frac{1}{\lambda_j^2}\right)(\tau_j^2\lambda_j^2)}\left[(\mathbf{x}^*)^2-2\mathbf{x}^*(\kappa_{j0}+\kappa_{j1}\mathbf{z})+(\kappa_{j0}+\kappa_{j1}\mathbf{z})^2\right]$$

The resulting expression above matches up with the exposure component of the incomplete case:

$$\frac{-1}{2\widehat{\lambda}_j^2}\left[\mathbf{x}^*-(\widehat{\kappa}_{j0}+\widehat{\kappa}_{j1}\mathbf{z})\right]^2=\frac{-1}{2\left(\frac{1}{\tau_j^2}+\frac{1}{\lambda_j^2}\right)(\tau_j^2\lambda_j^2)}\left[(\mathbf{x}^*)^2-2\mathbf{x}^*(\kappa_{j0}+\kappa_{j1}\mathbf{z})+(\kappa_{j0}+\kappa_{j1}\mathbf{z})^2\right]$$

; therefore, $\widehat{\lambda}_j^2=\lambda_j^2+\tau_j^2$, $\widehat{\kappa}_{j0}=\kappa_{j0}$, $\widehat{\kappa}_{j1}=\kappa_{j1}$. If τ_j^2 is known, we can obtain the parameters of the unobservable complete model - $\beta_{j0},\beta_{j1},\beta_{j2},\sigma_j,\xi_j,\lambda_j,\kappa_{j0},\kappa_{j1}$ - based on the estimates of $\widehat{\beta}_{j0},\widehat{\beta}_{j1},\widehat{\beta}_{j2},\widehat{\sigma}_j,\widehat{\xi}_j,\widehat{\lambda_j},\widehat{\kappa}_{j0},\widehat{\kappa}_{j1}$ in the incomplete model we have.

In summary, the error-free parameters for the complete model can be obtained from Gustafson's system of equations listed as below.

$$\lambda_j^2 = \widehat{\lambda}_j^2 - \tau_j^2$$

$$\kappa_{j0} = \widehat{\kappa}_{j0}$$

$$\kappa_{j1} = \widehat{\kappa}_{j1}$$

$$\beta_{j1} = \frac{\widehat{\beta}_{j1}\widehat{\lambda}_j^2}{\widehat{\lambda}_j^2 - \tau_j^2}$$

$$\beta_{j0} = \widehat{\beta}_{j0} - \frac{\beta_{j1}\widehat{\kappa}_{j0}\tau_j^2}{\widehat{\lambda}_j^2}$$

$$\beta_{j2} = \widehat{\beta}_{j2} - \frac{\beta_{j1}\widehat{\kappa}_{j1}\tau_j^2}{\widehat{\lambda}_j^2}$$

$$\sigma_j^2 = \widehat{\sigma}_j^2 - \frac{\beta_{j1}^2}{\frac{1}{\tau_j^2} + \frac{1}{\lambda_j^2}}$$

$$\xi_j^2 = \frac{\widehat{\xi}_j^2\left(\frac{\beta_{j1}^2}{\sigma_j^2} + \frac{1}{\tau_j^2} + \frac{1}{\lambda_j^2}\right)}{\frac{1}{\tau_j^2} + \frac{1}{\lambda_j^2} - \frac{\widehat{\xi}_j^2\beta_{j1}^2}{\sigma_j^2}}$$

F.6 Distribution Choices in Chapter 6

Modeling target	**Outcome (amount)**	**Covariate.1 (binary)**	**Covariate.2 (cont.)**
Data: $S, \mathbf{z}, \mathbf{x}$ (alternative)	Log-Skewnormal: $S \sim \mathbf{LogSN}(\mathbf{X}^T\boldsymbol{\beta}, \sigma^2, \xi)$ $\Rightarrow S \sim \mathbf{LogShiftedGa}(\cdot)$?	Bernoulli: $\mathbf{z} \sim \mathbf{Bern}(\pi)$ $\Rightarrow$ Logistic regression ?	Gaussian: $\mathbf{x} \sim \mathbf{N}(E[\mathbf{x}], \lambda^2)$ $\Rightarrow \mathbf{x} \sim \mathbf{t}(\cdot)$?, $\mathbf{x} \sim \mathbf{Laplace}(\cdot)$?
Param: $\boldsymbol{\beta}, \pi$ (alternative)	Multi-Normal: $\boldsymbol{\beta} \sim \mathbf{MVN}(\boldsymbol{\beta}_0, \sigma^2\Sigma_{\beta_0})$ $\Rightarrow \boldsymbol{\beta} \sim \mathbf{MVt}(\cdot)$?	Beta: $\pi \sim \mathbf{Beta}(g_0, h_0)$ $\Rightarrow \pi \sim \mathbf{N}(\cdot)$?; $0 \leq \pi \leq 1$	
Scale param: σ^2, λ^2 (alternative)	Inverse Gamma: $\sigma^2 \sim \mathbf{InvGa}(u_0/2, v_0/2)$ $\Rightarrow \sigma^2 \sim \mathbf{ScaledInv}\chi^\mathbf{2}(\cdot)$?		Inverse Gamma: $\lambda^2 \sim \mathbf{InvGa}(c_0/2, d_0/2)$ $\Rightarrow \lambda^2 \sim \mathbf{ScaledInv}\chi^\mathbf{2}(\cdot)$?
Skewness param: ξ (alternative)	Student's t: $\xi \sim \mathbf{t}(\nu_0)$ $\Rightarrow \xi \sim \mathbf{SN}(\cdot)$?		

Table F.1: Distribution choices/alternatives for outcome S and covariates $\mathbf{X}$ across data, parameter models. The selection of these distributions further informs the specification of hyperparameter models.

Here we explain some conventions about distribution choices. Table F.1 provides a structured comparison of the distributional choices for the log-skewnormal outcome variables and covariates (Bernoulli and Gaussian). Additionally, it highlights alternative distributional options, offering insight into the flexibility of the model specifications. A proper guide for choosing distributions can be found in Gelman and Meng 2004; Gelman and Hill 2007; Gelman and Carlin 2013; Gelman and Hwang 2014 and the references therein.

Data: $S, \mathbf{z}, \mathbf{x}$

- **Outcome:** Aggregate claim amount S

  → our choice: $S \sim \mathbf{LogSN}(\mathbf{X}^T\boldsymbol{\beta}, \sigma^2, \xi)$; i.e. Log-skewnormal

  → possible alternative: $S \sim \mathbf{LogShiftedGa}(\cdot)$; i.e. Log-shifted Gamma ?

  - *log-space location* $\mathbf{X}^T\boldsymbol{\beta}$; i.e. Gaussian regression
  - *log-space scale* $\sigma^2 \sim \mathbf{InvGa}(u_0/2, v_0/2)$; i.e. Inverse Gamma
  - *skewness* $\xi \sim \mathbf{t}(\nu_0)$; i.e. Student's t

    : The Log-skewnormal (**LogSN**) distribution is used for our aggregate claim amount data as it offers a simple yet accurate approximation of the sum of Log-normal random variables. Alternative distributions include the Log-shifted Gamma (**LogShiftedGa**), but it can be less stable, especially when the data does not clearly exhibit the characteristics captured by the Gamma's shape parameter. In contrast, the Log-skewnormal tends to provide more robust estimates with fewer convergence issues.

    : To model its parameters - log-space location $\mathbf{X}^T\boldsymbol{\beta}$, log-space scale σ^2, skewness ξ -, we use the Multivariate Gaussian for the GLM coefficients $\boldsymbol{\beta}$, the Inverse Gamma for the log-space scale, and the Student's t for the skewness. Since the Log-skewnormal lacks a conjugate relationship, alternatives such as the Multivariate Student's t (**MVt**) for the GLM coefficients, the Scaled Inverse Gamma for the scale, and the Skewnormal (**SN**) for the skewness could be considered. However, these options introduce significant additional complexity.

- **Covariate.1:** binary $\mathbf{z}$

  → our choice: $\mathbf{z} \sim \mathbf{Bern}(\pi)$; i.e. Bernoulli

  → possible alternative: Logistic regression ?

  - *probability* $\pi \sim \mathbf{Beta}(g_0, h_0)$; i.e. Beta

    : The Bernoulli (**Bern**) is used for our binary covariate as it accounts for only two possible values (0/1). While logistic regression could be an alternative, it introduces additional complexity by requiring parameter estimation for the link function.

    : To model its parameters - probability π -, we use the beta density (**Beta**) because the range of values is defined between 0 and 1, and the density serves

as a conjugate prior for **Bern**. The truncated normal (**N**) or non-informative prior such as uniform or Jeffeys prior, etc. can be alternative choices, but they come with trade-offs in terms of interpretability, flexibility, and computational complexity.

- **Covariate.2:** continuous $\mathbf{x}$

  $\rightarrow$ our choice: $\mathbf{x} \sim \mathbf{N}(E[\mathbf{x}], \lambda^2)$; i.e. Gaussian

  $\rightarrow$ possible alternative: $\mathbf{x} \sim \mathbf{t}(\cdot)$; i.e. Student's t ?, $\mathbf{x} \sim \mathbf{Laplace}(\cdot)$?

  - *location* $= E[\mathbf{x}]$
  - *scale* $\lambda^2 \sim \mathbf{InvGa}(c_0/2, d_0/2)$; i.e. Inverse Gamma

    : The Gaussian (**N**) is employed for our continuous covariate, as it effectively models data that tends to cluster around a central mean, while accommodating both positive and negative values. While the Student's t (**t**) or Laplace distributions could serve as alternatives, they are associated with specific characteristics, such as heavier tails or sharper peaks, which may not be suitable for our data.

    : To model its parameters - scale λ^2 -, we use the Inverse Gamma density (**InvGa**) because it serves as a conjugate prior for the Gaussian (**N**), while also effectively capturing a wide range of possible values for the scale due to its heavy tail. The scaled inverse Chi-square (**ScaledInv**$\boldsymbol{\chi^2}$) can be an alternative choice, but it may be more suitable when there are prior beliefs about the degrees of freedom.