Insurance Pricing using Bayesian Tree Models

Yaojun Zhang

School of Mathematics

University of Leeds

Submitted in accordance with the requirements for the degree of

Doctor of Philosophy

31th May 2024

Intellectual Property and Publication Statements
I confirm that the work submitted is my own, except where work which has formed part of jointly authored publications has been included. The contribution of myself and the other authors to this work has been explicitly indicated below. I confirm that appropriate credit has been given within the thesis where reference has been made to the work of others.

- Some materials in Chapter 2 (Sections 2.2 and 2.3), Chapter 3 (Sections 3.1, 3.2, 3.3, and 3.5), and Chapter 6 (Section 6.2) were published in a jointly authored paper: Zhang Y., Ji L., Aivaliotis G. and Taylor C.C., (2024). Bayesian CART models for insurance claims frequency. *Insurance: Mathematics & Economics*, 114, 108-131. In this work, I performed the analysis, calculations and obtained results, all of which are included in the present thesis. I conducted simulated and real data analyses, including the interpretation of the results. I presented this work at the 26th International Congress on Insurance: Mathematics and Economics (Edinburgh, United Kingdom). My co-authors, Dr Ji, Dr Aivaliotis and Professor Taylor designed the study and assisted with some of the steps in the derivation of the different models. Specifically, Dr Ji contributed to the MCMC algorithms for the implementation of Bayesian tree models, and Professor Taylor contributed to providing valuable suggestions for debugging the code. I wrote the first draft of the paper which was refined through iterative feedback and discussions with all of my co-authors. Dr Ji particularly contributed to the final writing of the paper.

This thesis is dedicated to my parents, whose unwavering support and heartfelt encouragement have been my pillars of strength throughout this challenging journey. This work is a testament to the collective effort of those who loved and believed in me. May this dedication reflect a sincere expression of my gratitude to all who have helped me in this academic pursuit.

Acknowledgements

I would like to express my sincere gratitude to all those who have contributed to the completion of this doctoral thesis.

First and foremost, I extend my deepest appreciation to my supervisors, Dr. Lanpeng Ji, Dr. Georgios Aivaliotis, and Professor Charles Taylor. Without their unwavering support, guidance, and invaluable insights, this research would not have been accomplished. Special thanks to my transfer and annual progress reviewer Dr. Jochen Voss, who provided valuable advice about MCMC algorithms. I also extend my gratitude to my thesis reviewers, Dr. Arief Gusnanto and Professor Andrew Parnell, for their invaluable comments.

I am thankful to the anonymous referee for their constructive suggestions for my paper (Zhang Y., Ji L., Aivaliotis G. and Taylor C.C., 2023), in particular for pointing out the necessity to explore the contributions of different tree moves and emphasizing the importance of providing a thorough literature review on the different options for the tree prior.

I am grateful to the University of Leeds for providing a conducive academic environment and resources essential for the successful completion of this thesis. The support from the faculty, staff, and fellow researchers has been truly enriching.

To my family, friends, and colleagues, thank you for your encouragement, understanding, and patience, especially during the challenging Covid lockdown period. Your belief in me has been a constant source of motivation.

Finally, I express my heartfelt thanks to all participants and individuals who, directly or indirectly, contributed to the research process.

Abstract

The accuracy and interpretability of a (non-life) insurance pricing model are essential qualities to ensure fair and transparent premiums for policyholders, that reflect their risk. In recent years, classification and regression trees (CARTs) and their ensembles have gained popularity in the actuarial literature, since they offer good prediction performance and are relatively easy to interpret. In this work, we investigate Bayesian CART models for insurance pricing. In addition to the commonly used Poisson and Negative Binomial (NB) distributions for claims frequency, we combine Bayesian CART models and zero-inflated distributions, namely, zero-inflated Poisson (ZIP) and general zero-inflated Negative Binomial (ZINB) to address the difficulty arising from the imbalanced insurance claims data. In claims severity analysis, we discover that the Weibull distribution has the ability to capture different tail characteristics in tree models. Moreover, we propose and investigate three types of models for aggregate claims modelling. We find that sequential models and joint models, which incorporate dependence between the number of claims and claims severity, are preferable to the standard frequency-severity models. We introduce a general MCMC algorithm using data augmentation methods for posterior tree exploration. We also develop various types of deviance information criterion (DIC) for tree model selection. The proposed models are able to identify trees which can better classify the policyholders into risk groups. The effectiveness of these models' performance are illustrated by several carefully designed simulations and real insurance data.

Notation

$\mathcal{D} = (\boldsymbol{X}, \boldsymbol{y})$	data set
$\boldsymbol{x}_i = (x_{i1}, x_{i2}, \ldots, x_{ip})$	a row vector of explanatory variables
$i = 1, \ldots, n$	the number of observations/policy-holders
$l = 1, \ldots, p$	the number of explanatory variables
$\mathcal{X}$	covariate sample space
$\boldsymbol{y} = (y_1, y_2, \ldots, y_n)$	a row vector of response variable
$\mathcal{Y}$	response variable sample space
$\mathcal{T}$	a binary tree
$\lvert\mathcal{T}\rvert$ or $t = 1, \ldots, b$	the number of terminal nodes in the tree $\mathcal{T}$
$\{\mathcal{A}_1, \ldots, \mathcal{A}_b\}$	a partition of the covariate sample space $\mathcal{X}$
$\boldsymbol{\theta}_t$	the parameter in the t-th terminal node
$j = 1, \ldots, n_t$	the number of observations in the t-th node
$\boldsymbol{v}$	exposure in yearly units
$\boldsymbol{N}$	claims number
λ	claims frequency
$I_{()}$	indicator function
d	depth of the tree
γ	the parameter in the splitting probability
ρ	the parameter in the splitting probability
$\boldsymbol{z} = (z_1, \ldots, z_n)$	a general latent variable
m	the number of test data
$\boldsymbol{\epsilon}$	empirical claims frequency (or severity, cost)
Y_{iN_i}	individual claim amount
$\boldsymbol{S}$	aggregate claim amount
α	the parameter in Gamma/Beta/Weibull distribution
β	the parameter in Gamma/Beta/Weibull distribution
κ	the parameter in NB distribution
μ	the parameter in ZIP/ZICPG/LogNormal distributions

$\boldsymbol{\xi} = (\xi_{t1}, \ldots, \xi_{tn_t})$	a latent variable in NB/ZINB models
$\boldsymbol{\phi} = (\phi_{t1}, \ldots, \phi_{tn_t})$	a latent variable in ZIP/ZINB/ZICPG models
$\boldsymbol{\delta} = (\delta_{t1}, \ldots, \delta_{tn_t})$	a latent variable in ZIP/ZINB/ZICPG models

Abbreviations

CARTs	Classification and Regression Trees
BCARTs	Bayesian Classification and Regression Trees
NB	Negative Binomial
ZIP	Zero-Inflated Poisson
ZINB	Zero-Inflated Negative Binomial
CPG	Compound Poisson Gamma
ZICPG	Zero-Inflated Compound Poisson Gamma
MCMC	Markov Chain Monte Carlo
DIC	Deviance Information Criterion
GLMs	Generalized Linear Models
GAMs	Generalized Additive Models
NN	Neural Network
RF	Random Forest
GB	Gradient Boosting
GBT	Gradient Boosting Trees
DB	Delta Boosting
BART	Bayesian Additive Regression Trees
BMA	Bayesian Model Averaging
MOTR-BART	Model Trees BART
MH	Metropolis–Hastings
RI	Rand Index
ARI	Adjusted Rand Index
MLEs	Maximum Likelihood Estimators
IID	Independent and Identically Distributed
RJMCMC	Reversible Jump Markov Chain Monte Carlo
AIC	Akaike's Information Criterion
WAIC	Watanabe-Akaike Information Criterion
RSS	Residual Sum of Squares
SE	Squared Error

DS	Discrepancy Statistics
NLL	Negative Log-likelihood
MME	Method of Moments Estimation
LPML	Log Pseudo Marginal Likelihood

Contents

List of Figures

List of Tables

Chapter 1

Introduction

This chapter begins with an overview of the insurance background and our objectives, followed by a literature review from two perspectives. One aspect explores various traditional methods for insurance pricing, while the other delves into tree-based models, particularly the Bayesian tree-based models. Subsequently, we outline the main contributions and provide an overview of the thesis structure.

1.1 Background and Objectives

An insurance policy refers to an agreement between an insurance company (the insurer) and a policyholder (the insured), in which the insurer promises to charge the insured a certain fee for some unpredictable losses of the customer within a period of time, usually one year. Non-life insurance includes policies for things like auto, travel, home, and so on. The charged fee is called a *premium* which includes a *pure premium* and other loadings such as operational costs. For each policy, the pure premium is determined by multiple explanatory variables (such as characteristics of the policyholders, the insured objects, the geographical region, etc.), also called *risk factors*; see, e.g., Ohlsson & Johansson (2010). The premium charged reflects the customer's degree of risk; a higher premium suggests a potentially higher risk, and vice versa. If insurance companies were to charge the same premium for everyone, low-risk individuals would likely seek cheaper insurance rates elsewhere, leaving insurers with a larger pool of high-risk individuals, potentially leading to insufficient premium income to cover all losses. This phenomenon, also known as adverse selection, occurs when insurers are more likely to cover bad risks than good risks. Therefore, it is necessary to use risk factors to classify policyholders with similar risk profiles into the same tariff class. The insureds in the same group, all having similar risk characteristics, will pay the same reasonable premium. The

process of constructing these tariff classes is also known as *risk classification*; see, e.g., Denuit *et al.* (2007) and Henckaerts *et al.* (2018). In the basic formula of non-life insurance pricing, the pure premium is obtained by multiplying the expected claims frequency with the conditional expectation of claims severity, assuming independence between claims frequency and severity; see, e.g., Henckaerts *et al.* (2021). In this thesis, we propose efficacious models, namely, Bayesian Classification and Regression Trees (Bayesian CARTs) or BCART models, to analyze insurance claims data. First, we explore imbalanced claims frequency data with BCART models, considering different ways to embed the exposure (typically in yearly units, used to quantify how long the policyholder is exposed to risk) into the models and involving data augmentation techniques. Subsequently, we investigate BCART models to analyze the right-skewed and heavy-tailed claims severity data by using various distributions. Finally, we discuss three types of models that we propose, i.e., frequency-severity models, sequential models and joint models for aggregate claims modelling.

1.2 Literature Review

1.2.1 Insurance Pricing

In order to estimate the relationship between the risk factors and the premium, a statistical model is used. Due to its flexibility in modelling a large number of distributions in the exponential family, generalized linear models (GLMs), developed in Nelder & Wedderburn (1972), have been the industry-standard predictive models for insurance pricing; see, e.g., Denuit *et al.* (2007) and Wuthrich (2022). Explanatory variables enter a GLM through a linear predictor, leading to interpretable effects of the risk factors on the response. Assuming independence between claims frequency and severity, the frequency-severity models treat these two components separately. The frequency element focuses on the occurrence of claims, and the severity element, provided that a claim has occurred, investigates the claim amount. Both elements can use distributions from the exponential family within GLMs; see, e.g., David (2015). Claims frequency is typically modelled using non-negative discrete probability distributions such as Poisson, Negative Binomial (NB), Zero-Inflated Poisson (ZIP) or a general Zero-Inflated Negative Binomial (ZINB), and claims severity is typically modelled using non-negative continuous distributions such as Gamma, LogNormal, Weibull, or a generalized Pareto. As a result, the expected annual claims cost can be calculated as the product of the

expectations of claims frequency and conditional claims severity. Alternatively, the aggregate loss can be directly modelled using the Tweedie Compound Poisson model which considers loss as a Poisson sum of Gamma variables, simplifying the analysis by jointly accommodating discrete and continuous data components; see, e.g., Jørgensen & Paes De Souza (1994). Concurrently, discussions regarding the suitability of GLMs for aggregate claims analysis have revolved around the trade-off between model complexity and predictive performance, emphasizing the benefits and contexts where Tweedie's model excels and where alternative methodologies may offer a better solution; see, e.g., Quijano Xacur & Garrido (2015). Extensions of GLMs to generalized additive models (GAMs) (see Hastie & Tibshirani (1987)) to capture the nonlinear effects of risk factors sometimes offer more flexible models. Certain risk factors exhibit inherent relationships, such as the connection between vehicle value and the driver's salary. The identification of complex interactions among these risk factors poses a crucial challenge for modeling efforts. Both GLMs and GAMs often fail to capture these intricate relationships due to their inherent assumptions and limitations. While GAMs relax the linearity assumption by incorporating smooth non-linear components through the use of splines, they encounter difficulties in capturing interactions characterized by intricate and complex patterns. We refer to Ohlsson & Johansson (2010) for a more comprehensive discussion on this. Another popular classical method which is based on Bayesian statistics, the credibility method, was introduced for balancing policyholder-specific data with industry-wide loss experience, ensuring fair and accurate premium rates. Besides, the efficacy of credibility theory in addressing the challenges posed by multi-level factors and lack of data issues has been discussed; see, e.g., Ohlsson & Johansson (2010) and Bühlmann & Gisler (2005).

Moreover, an increasing body of literature emphasizes the importance of understanding the interrelated nature of claims occurrences and their associated claim amount to improve prediction accuracy, advocating for a relaxation of the independence assumption between claims frequency and severity; see, e.g., Frees *et al.* (2016) and Lee & Shi (2019). To address this concern, conditional GLMs were proposed by Garrido *et al.* (2016), enabling the severity component of the aggregate claims model to depend on the frequency component. This strategy is straightforward to implement and has an easy-to-interpret correction term reflecting the dependence. Additionally, another strategy primarily used to model the dependence structure between random variables, the copula method, has been introduced for capturing the dependence structure of count data in the insurance

industry; see, e.g., Genest & Nešlehová (2007). Copulas are particularly useful for modelling the joint distribution of variables while allowing for flexible marginal distributions and capturing the dependence structure independently. In parallel, the adoption of Bayesian approaches to copula modelling in Smith (2011) contributed to the refinement of copula-based modelling techniques. By augmenting the likelihood with latent variables and employing efficient Markov Chain Monte Carlo (MCMC) sampling schemes, copula models with discrete margins can be estimated using the resulting augmented posterior. This strategy suggests the potential applicability of the proposed method in higher-dimensional settings and underscores the limitations of elliptical copulas in capturing dependence in discrete data; see, e.g., Smith & Khaled (2012). After that, their work is expanded to include situations where some margins are discrete and others are continuous, providing strong theoretical support for the subsequent development of mixed copula models. By incorporating mixed copulas, the dependence between claims frequency and severity can be addressed. However, the challenge persists in selecting the appropriate copula family and parameters; see, e.g., Czado *et al.* (2012), Shi *et al.* (2015) and Lee *et al.* (2019). Because of the limitations of these classical statistical methods and equipped with continually developing technologies, further research has recently turned to machine learning techniques. Several machine learning methods such as neural networks (NN), regression trees, bagging techniques, random forests (RF), and boosting machines have been introduced in the context of insurance by adopting actuarial loss distributions in these models to capture the characteristics of insurance claims. We refer to Blier-Wong *et al.* (2020) for a recent literature review on this topic, and Denuit & Trufin (2019), Wuthrich & Buser (2022) and Wüthrich & Merz (2023) for a more detailed discussion.

1.2.2 Tree-based Models

Insurance pricing models are heavily regulated and they must meet specific requirements before being deployed in practice, which poses some challenges for machine learning methods; see Henckaerts *et al.* (2021). Therein, it is stressed that pricing models must be transparent and easy to communicate to all the stakeholders and that the insurer has the social role of creating solidarity among the policyholders so that the use of machine learning for pricing should in no way lead to an extreme penalization of risk or discrimination. The latter has also been noted recently in, e.g., Denuit *et al.* (2021) and Wüthrich (2020) where it is claimed that prediction

accuracy on an individual level should not be the ultimate goal in insurance pricing; one also needs to ensure the balance property which means it is crucial that the models provide a reasonable premium estimation at the portfolio level. Bearing these points in mind, researchers have concluded that tree-based models are good candidates for insurance pricing due to their capacity to flexibly capture complex, non-linear relationships inherent in diverse data sets. These models provide transparent decision rules, facilitating easy interpretation and trust among stakeholders. Besides, their ability to handle both numerical and categorical variables without extensive pre-processing simplifies the modelling process; see, e.g., Henckaerts *et al.* (2021), Quan (2019), Hu *et al.* (2022), Meng *et al.* (2022) and Lindholm *et al.* (2023). More precisely, the use of CART, first introduced in Breiman *et al.* (1984), partitions a portfolio of policyholders into smaller groups of homogeneous risk profiles based on some risk factors in which a constant prediction is then used for each subgroup.

Acknowledging the challenges posed by handling excessive zeros in imbalanced insurance claims frequency data, a novel decision tree approach designed for zero-inflated count data was introduced, using the likelihood of a ZIP model in the splitting criterion; see, e.g., Lee & Jin (2006). Because of the continuous development of decision trees, their application is expanding in the insurance industry. For example, decision trees have also been employed to predict the likelihood of a claim based on potential risk factors; see, e.g., Frempong *et al.* (2017). Besides, to investigate the diverse characteristics of cyber claims, regression trees were used in Farkas *et al.* (2021). In this contribution, they employed a generalised Pareto likelihood with two parameters in the splitting criterion to model the heavy-tailed cyber claims data. Moreover, the integration of tree-based models into credibility theory was investigated in Diao & Weng (2019), aiming to incorporate covariate information into credibility premium prediction, and the proposed algorithm demonstrated superior prediction accuracy. For a comprehensive understanding of how decision trees can be implemented, we refer to Breiman *et al.* (1984) and Loh (2014) for detailed discussions on the methodology. Additionally, the multivariate regression tree approach has been further explored for its potential application in a wide range of scenarios where the simultaneous co-occurrence of several dependent variables need to be predicted (e.g., the number of claims and claim amount). To obtain theoretical insights into this methodology, we refer to Yu & Lambert (1999), Larsen & Speckman (2004) and Lee (2005).

Although a large number of scholars have carried out empirical and theoretical studies on the effectiveness of CART, limitations of the greedy forward-search

recursive partitioning method used in CART have been identified. In particular, the predictive performance tends to be low, and it is known to be unstable: small variations in the training set can result in greatly different trees and different predictions for the same test examples. Due to these limitations, more complex tree-based models that combine multiple trees in an ensemble have become popular in insurance prediction and pricing to enhance predictive accuracy, such as RF and boosting algorithms; see, e.g., Breiman (2001), Yang *et al.* (2018), Lee (2020, 2021) and Henckaerts *et al.* (2021). In detail, RF refers to generating ensembles of trees with a set of unpruned fully-grown trees, aiming to reduce variability through averaging. During this process, these trees are generated based on a bootstrap sampling of the original data, using a sub-sample of explanatory variables at each splitting step; see, e.g., Breiman (2001). Nevertheless, the trees in RF are generated independently, lacking information sharing among them. Boosting algorithms provide an improvement on this, as they construct trees sequentially. A tree is grown based on residuals from previously grown trees for each new iteration. This procedure creates strong learners by combining weak learners, achieving a good balance between bias and variance through parameter tuning. We refer to Freund & Mason (1999), Freund *et al.* (1996) and Azmi & Baliga (2020) for some discussion on this topic. The term gradient boosting (GB) originated from the use of optimisation based on gradient descent algorithms in the early techniques of gradient boosting trees; see, e.g., Friedman (2002). In contrast to other methods, GB simplifies complex interactions and manages missing values with minimal information loss. Applying the idea of GB to auto insurance loss cost modelling has been demonstrated to obtain superior predictive accuracy compared to GLMs; see, e.g., Guelman (2012). The introduction of a gradient tree-boosting approach to Tweedie Compound Poisson models by Yang *et al.* (2018), is another noteworthy addition. The advent of the R package, `TDboost`, further enhances the accessibility and practicality of this proposed method; more information about this package can be found in Yang & Qian (2022). Besides, the introduction of delta boosting (DB) in Lee & Lin (2018) and Lee (2020), as another novel member of the boosting family, offers promising advancements in boosting algorithms. Unlike GB, DB relies on a new measure called "delta" and demonstrates optimality for various loss functions. Its asymptotic version, customised for certain loss functions, works well and effectively mitigates the biases found in GB. However, while these ensemble methods are able to significantly increase prediction accuracy, they usually introduce additional difficulties in model transparency. From another perspective,

to address the issue of CART producing locally optimal results due to their stepwise search procedure, evolutionary algorithms were introduced (see Grubinger *et al.* (2014)) and implemented in the R package `evtree`, to emphasize the significance of global optimization techniques, enabling a more complete exploration of the parameter space of trees; further details are available in Grubinger & Pfeiffer (2019). However, running evolutionary trees requires more computer memory and processing time. Moreover, the inherent randomness of evolutionary algorithms poses another issue. For larger data sets, different evolutionary trees may produce a similar or even identical evaluation function value. This phenomenon makes their interpretation challenging.

In this thesis, we propose BCART models for insurance pricing. Instead of making an ensemble of trees, we look for one good tree, which can improve the prediction ability by exploring a global covariate space whilst ensuring model transparency, by adopting a Bayesian approach applied to CART.

BCART models were first introduced by Chipman *et al.* (1998) and Denison *et al.* (1998), independently. The method has two basic components, prior specification (for the tree and its terminal node parameters) and a stochastic search. The method is to obtain a posterior distribution given the prior, thus leading the stochastic search towards more promising tree models. Compared with the tree that CART generates by a greedy forward-search recursive partitioning method, the BCART model generates a much better tree by an effective Bayesian-motivated stochastic search algorithm. This has been justified by simulation examples (with Gaussian-distributed data) in the aforementioned papers. Here, we show another simulation example with Poisson-distributed data to illustrate the effectiveness of BCART models. Specifically, we simulate 5,000 Poisson-distributed observations where the Poisson intensity (response, $\boldsymbol{y}$) depends on two explanatory variables (or covariates) x_1 and x_2 as illustrated in Figure 1.1. (See also Subsection 3.5.1 for a slightly more general simulation example.) It is clear from the figure that the optimal partition of the covariate space consists of four regions where the data in each region then follows a homogeneous Poisson distribution. Note that the standard CART will not be able to find the correct partition of the data as the Poisson intensities are almost uniform for both marginal distributions (see Figure 1.1) and no matter how the first split is chosen, it is difficult to distinguish different Poisson intensities on the resulting subspaces. In contrast, the proposed Poisson BCART excels in retrieving the optimal tree structure due to its ability to explore the tree space in a global way (for example, it can modify previously

chosen splits, a capability that significantly contributes to achieving superior results). Moreover, the BCART models demonstrate additional strengths, notably providing a probabilistic framework that reduces the instability of standard CART while maintaining their explainability.

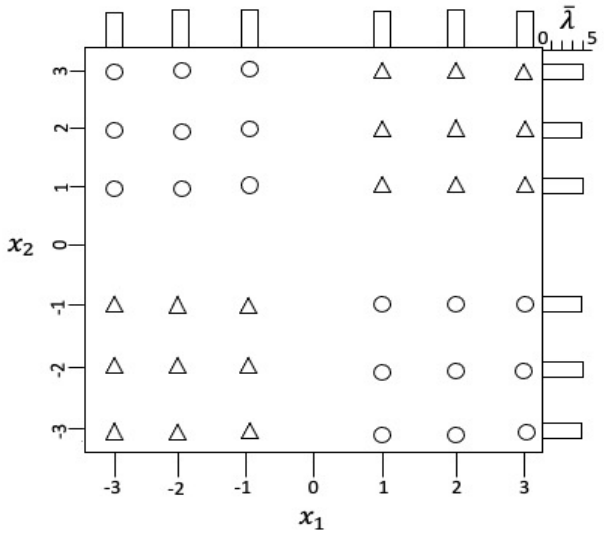

Figure 1.1: Covariate partition for a Poisson-distributed simulation. Two covariates x_1, x_2 follow a discrete Uniform distribution with the support $\{-3, -2, -1, 1, 2, 3\}$. The response variable, which is simulated for the points, has a Poisson intensity equal to 1 (circles) and 7 (triangles). Each bar represents the average value (≈ 4) of Poisson intensity in that row/column of data.

Because of the advantages of BCART models, Wu *et al.* (2007) delved into the prior specification, emphasizing the crucial role of informed assumptions in the Bayesian framework and providing a comprehensive understanding of the interplay between prior distributions and resulting model outcomes. Notably, an explicit specification of both the tree size and the tree shape is made possible by the introduction of the pinball prior. This prior allows the construction of balanced or skewed trees by adjusting a hyper-parameter, providing a formal prior distributional structure for tree production. Furthermore, an extension of BCART models, namely, Bayesian "treed models" was proposed, allowing for a functional relationship between the variable of interest and the predictors to replace a constant prediction used for each subgroup (terminal node) in the tree. This strategy enables local modelling across the predictor space; see, e.g., Chipman *et al.* (2003, 2002).

Since BCART models and their ensemble version – the Bayesian Additive Regression Trees (BART) models – generally outperform other machine learning models because of their ability to quantify uncertainty, they have been extensively studied in the literature; see, e.g., Linero (2017), Chipman *et al.* (2010), Murray (2021), Hill *et al.* (2020) and references therein. In conjunction with the extensive exploration of BART models and their role in machine learning, Kapelner & Bleich (2013) introduced the R package `bartMachine` for implementation; after this, as a supplement, another R package `BART` was provided by Sparapani *et al.* (2021); see Kapelner & Bleich (2023) and McCulloch & Spanbauer (2023). Besides, Bayesian Model Averaging (BMA) approaches were incorporated into BART models. This novel method creates a hybrid algorithm that can handle high-dimensional data by combining elements of both BART and RF; see, e.g., Hernández *et al.* (2018). Another extension of BART models known as Model Trees BART (MOTR-BART) considers piece-wise linear functions at node levels instead of piece-wise constants, enabling more efficient estimation of local linearities compared to the original BART models; see, e.g., Prado *et al.* (2021a). Furthermore, extensions of BART to semi-parametric models were introduced in Prado *et al.* (2021b). By refining the tree-generation moves in BART models, the study addresses bias and non-identifiability concerns between the parametric and non-parametric components, even when they share common covariates. This research serves as a foundational resource in the field of semi-parametric modelling, providing valuable insights for similar research endeavours. In particular, the excellent empirical performance of BART models has also motivated works on their theoretical foundations; see, e.g., Pratola (2016) and Linero & Yang (2018). Recognizing the limitations of classical Metropolis–Hastings (MH) proposals in terms of efficiently exploring the model space, researchers have also introduced some other proposals, such as a novel tree rotation proposal and a rule perturbation proposal, tailored to the topological structure of Bayesian regression trees. These innovative strategies ensure faster convergence and more accurate estimations; see, e.g., Pratola (2016). Besides, Linero & Yang (2018) incorporated smoothness and sparsity within the BART framework. Particularly in the context of high-dimensional data analysis, the model provides more accurate and interpretable results by considering sparsity inducing soft decision trees, where the decisions are treated as probabilistic. Additionally, the study demonstrates the concentration of the posterior distribution at the minimax rate for sparse functions and those with additive structures from a theoretical perspective, which ends by highlighting the fact that only minor adjustments to the existing BART algorithms are sufficient for implementation.

Moreover, to address the issue of posterior concentration in Bayesian regression trees and forests, a spike-and-tree variant of the well-known BCART prior was proposed to establish new theoretical results; see, e.g., Rocková *et al.* (2020).

1.3 Main Contributions

Although numerous studies have been conducted on Bayesian tree-based models, the focus has generally been on Gaussian-distributed data, with some exceptions such as Murray (2021) and Linero *et al.* (2020). Therefore, the existing algorithms do not seem to be directly applicable to insurance data for prediction and pricing. It turns out that a data augmentation approach is needed when dealing with general non-Gaussian data in the insurance industry. To cover this gap, we propose BCART models for insurance pricing that take into account special features of insurance data, such as the large number of zeros, the involvement of exposure for claims frequency, and the right-skewed and heavy-tailed nature of claims severity. The main contributions of this thesis are as follows.

1. We give a general MCMC algorithm for the BCART models applied to data with a general distribution, where a data augmentation may be needed. In doing so, we follow some ideas in Meng & Van Dyk (1999) and Van Dyk & Meng (2001).

2. We introduce a novel model selection method for BCART models based on the deviance information criterion (DIC). Note that DIC was introduced in Spiegelhalter *et al.* (2002) which appeared after the introduction of BCART models (see Chipman *et al.* (1998)). Although the use of DIC is widespread, in recent years, no studies have explored its application in selecting an optimal tree. We propose a three-step approach tailored for this purpose. Additionally, to accommodate various scenarios, such as incorporating the data augmentation technique and treating certain parameters as known, we introduce various types of DIC. The effectiveness of this approach is illustrated by several designed simulation examples and real insurance data.

3. We implement BCART models for various distributions for claims frequency (such as Poisson, NB, ZIP and ZINB), claims severity (such as Gamma and Weibull), and aggregate claims, namely, Compound Poisson Gamma (CPG) and Zero-Inflated Compound Poisson Gamma (ZICPG), which are

not currently available in any existing R packages. In particular, we introduce different ways of incorporating exposure in the NB, ZIP, ZINB and ZICPG models, following Lee (2020, 2021) who focused on claims frequency modelling using delta boosting. The simulation examples and real insurance data analyses show the applicability of these proposed BCART models.

4. We propose three types of models for aggregate claims modelling. First, we introduce two BCART models (two trees) for claims frequency and claims severity independently within the frequency-severity models. Second, to explore the dependence between the number of claims and claims severity, we introduce sequential models. These models treat the number of claims as a covariate in claims severity modelling, exhibiting superior performance compared to frequency-severity models in real insurance data analyses. In doing so, we draw on some ideas from Garrido *et al.* (2016), expanding their application in tree models. Last, as far as we are aware there have been no BCART models discussed for multivariate responses in the current literature. We investigate joint models for a bivariate response (number of claims and aggregate claim amount) to directly model the claims cost using one joint tree. For these joint models, we employ two commonly used distributions, CPG and ZICPG, demonstrating the potential advantages of information sharing (see related discussion in Linero *et al.* (2020)).

5. We introduce some model performance measures specifically designed for testing tree models, including squared error based on a sub-portfolio (terminal node) level and a "lift". We also propose using the adjusted Rand Index (ARI) to assess the similarity between different trees. Although widely used in clusterings, we expand its application to tree models. This index enhances the understanding of the necessity of information sharing, an aspect not covered in the current literature, e.g., Linero *et al.* (2020). Furthermore, we introduce a method to analyse the stability of tree models, aiding our understanding of the superiority of Bayesian tree-based models over standard decision trees.

6. To date, Bayesian tree-based models have not attracted enough attention compared to other machine learning methods in the actuarial community. This first step of applying BCART models for insurance pricing will open the door for more sophisticated tree-based models to meet the needs of the

insurance industry. We illustrate the effectiveness of BCART models using both simulated and real insurance data.

1.4 Structure of the Thesis

In Chapter 2, we review the BCART framework which includes an extension with data augmentation and a model selection method using DIC. Various evaluation metrics are also described. Chapter 3 introduces the notation for insurance claims frequency data and several BCART models. The applicability of the proposed BCART models is discussed using some simulation examples. In Chapter 4, three BCART models (Gamma, LogNormal and Weibull) are introduced for claims severity modelling, followed by a simulation example to assess their ability to handle data with varying tail characteristics. Chapter 5 discusses three types of models for aggregate claims modelling, namely, frequency-severity models, sequential models, and joint models, followed by two simulation examples. Chapter 6 is focused on a real insurance data analysis to illustrate the effectiveness and feasibility of the proposed BCART models. Finally, Chapter 7 concludes the thesis and outlines future directions.

Chapter 2

Bayesian CART Theory

In this chapter, we review the Bayesian CART framework as initially introduced in Chipman *et al.* (1998) and conduct a more comprehensive and in-depth study based on subsequent relevant literature, following an extension with data augmentation and a model selection method using DIC. To begin, we provide a basic introduction to CART and address its limitations, which serve as the motivation for exploring the BCART models. Subsequently, we present a general theory of Bayesian CART, including prior choice, MCMC algorithms and posterior tree selection and prediction. Finally, we introduce various evaluation metrics for validating and comparing different models in the subsequent simulation studies and real data analyses.

2.1 General Theory of CART

We introduce the CART algorithm in this section. Initially, we describe the mathematical expression of CART, providing a comprehensive overview of the general process for generating a tree. Subsequently, we provide an illustrative example to demonstrate how CART works with a specific distribution, namely, the Poisson distribution. Furthermore, we discuss the limitations of CART to underscore the importance of exploring the parameter space of trees globally, leading to the investigation of BCART models in the next section.

Consider a data set $\mathcal{D} = (\boldsymbol{X}, \boldsymbol{y}) = \big((\boldsymbol{x}_1, y_1), (\boldsymbol{x}_2, y_2), \ldots, (\boldsymbol{x}_n, y_n)\big)^\top$ with n observations. For the i-th observation, $\boldsymbol{x}_i = (x_{i1}, x_{i2}, \ldots, x_{ip})$ represents a vector of p explanatory variables (or covariates) sampled from a space $\mathcal{X}$; y_i is a response variable sampled from a space $\mathcal{Y}$. For the severity (or frequency) modelling, $\mathcal{Y}$ is a space of real positive (or integer) values. For the aggregate claims modelling, we

shall discuss models where $\mathcal{Y}$ is a space of 2-dimensional vectors with two components, namely, an integer for number of claims and a real value for aggregate claims amount. Throughout the thesis, observations are assumed to be independent.

Theoretically, decision trees are commonly used to partition data into binary groups, known as binary splits. However, decision trees can also include multi-way splits, which are beneficial when observations need to be allocated to more than two groups; see, e.g., Fulton *et al.* (1995). In addition, when selecting the splitting rule in each split, it is possible to choose either one splitting variable or a combination of multiple variables; we refer to Bennett (1992) for some insights. Nevertheless, the optimal search for both multi-way splits in numerical variables and consideration of combinations of multiple covariates can be significantly demanding. Therefore, the thesis is currently restricted to binary splits, with the choice of a single splitting covariate in each split.

2.1.1 The Structure of a CART Model

The algorithm of CART works by repeatedly partitioning the data into multiple subspaces so that the data in each subspace are as homogeneous as possible. This strategy is technically called *recursive partitioning*. As the name suggests, CART builds a tree that predicts the value of a response variable using a set of predictor variables. The key to CART is how to determine optimal splits using the available data and when to cease splitting to obtain the terminal nodes and their assignments. A major issue is the absence of growing right-sized trees, leading to excessively large trees due to redundant splits and overly optimistic estimates. To address this, the general idea of CART is to continue splitting until all terminal nodes contain minimal data, obtaining a large tree, which is then pruned selectively to yield a decreasing sequence of sub-trees. Finally, the sub-tree with the best estimated goodness of fit, typically determined using a loss function such as log-likelihood, is selected using cross-validation. This subsection will detail the complete process of obtaining the optimal sub-tree, encompassing these two steps, i.e., grow and prune.

To provide a clearer understanding of the tree generation process, the general mathematical structure of a CART model is introduced. A CART has two main components: a binary tree $\mathcal{T}$ with b terminal nodes which induces a partition of the covariate space $\mathcal{X}$, denoted by $\{\mathcal{A}_1, \ldots, \mathcal{A}_b\}$, and a terminal node specific distribution $f\left(y_i \mid \boldsymbol{\theta}_t\right)$ for the response variable y_i if $\boldsymbol{x}_i \in \mathcal{A}_t$, where $\boldsymbol{\theta}_t$ is the parameter value of $f\left(y_i \mid \boldsymbol{\theta}_t\right)$ restricted to terminal node t. We denote $\boldsymbol{\theta} = (\boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \ldots, \boldsymbol{\theta}_b)$

as the terminal node parameter of the tree. Note that here we do not specify the dimension and range of the parameter $\boldsymbol{\theta}_t$ explicitly which should be clear in the considered context below.

By associating observations with all terminal nodes in the tree $\mathcal{T}$, we can represent the data set as

$$(\boldsymbol{X}, \boldsymbol{y}) = \left((\boldsymbol{X}_1, \boldsymbol{y}_1^\top), (\boldsymbol{X}_2, \boldsymbol{y}_2^\top) \ldots, (\boldsymbol{X}_b, \boldsymbol{y}_b^\top)\right)^\top,$$

where, for terminal node t, $\boldsymbol{X_t} = \begin{bmatrix} x_{t11} & x_{t12} & \ldots & x_{t1p} \\ x_{t21} & x_{t22} & \ldots & x_{t2p} \\ \vdots & \vdots & \ddots & \vdots \\ x_{tn_t1} & x_{tn_t2} & \ldots & x_{tn_tp} \end{bmatrix}$ with n_t denoting the number of observations in the t-th terminal node, p representing the number of explanatory variables (or covaraites), and x_{tjl} denoting the value/category of the l-th covariate for the j-th observation; $\boldsymbol{y}_t = (y_{t1}, \ldots y_{tn_t})$ is an analogously defined vector with y_{tj} denoting the j-th observed response variable in the t-th terminal node. We shall make the typical assumption that conditionally on $(\boldsymbol{\theta}, \mathcal{T})$, response variables within a terminal node are independent and identically distributed (IID), and they are also independent across terminal nodes. The CART model likelihood in this case will take the form

$$p(\boldsymbol{y} \mid \boldsymbol{X}, \boldsymbol{\theta}, \mathcal{T}) = \prod_{t=1}^{b} f\left(\boldsymbol{y}_t \mid \boldsymbol{\theta}_t\right) = \prod_{t=1}^{b} \prod_{j=1}^{n_t} f\left(y_{tj} \mid \boldsymbol{\theta}_t\right). \tag{2.1}$$

It is worth noting that instead of the IID assumption within the terminal nodes, more general models can be considered; see, e.g., Chipman *et al.* (2003, 2002) and the references therein.

From (2.1), there are two things that need to be explicitly addressed in order to obtain a fully grown tree: how to choose a good partition $\{\mathcal{A}_t\}_{t=1}^b$ of the feature space $\mathcal{X}$, and how to obtain the estimated parameter $\hat{\boldsymbol{\theta}}_t$ for each terminal node. To delve deeper into the process of tree growth, it is essential to understand the step-by-step progression and the methodology for deriving feature space partitions and corresponding estimated parameters. The following discussion will clarify this aspect.

Let η_0 denote the root node, which is the initial node that can be grown at the first step. The corresponding covariate subspace can be denoted as $\mathcal{B}_{\eta_0}$ (for terminal node this is simply $\mathcal{X}$). The pair (η_1, η_2) represents two child nodes of η_0 with corresponding subspaces $(\mathcal{B}_{\eta_1}, \mathcal{B}_{\eta_2})$, such that $\mathcal{B}_{\eta_1} \cup \mathcal{B}_{\eta_2} = \mathcal{B}_{\eta_0}$ and $\mathcal{B}_{\eta_1} \cap \mathcal{B}_{\eta_2} = \emptyset$. Each subspace $\mathcal{B}_{\eta_1}$ (or $\mathcal{B}_{\eta_2}$) is associated with a parameter θ_{η_1} (or

θ_{η_2}), assuming there is only one parameter within each subspace in this context for simplicity. The objective is to find an optimal solution for the split based on the available data in the current node (which is the whole data for root node). This involves identifying the optimal feature component x_{il} $(l = 1, 2, \ldots, p)$ of $\boldsymbol{x}_i$, and the optimal constant $c \in \mathbb{R}$ (for continuous feature components) or the optimal non-empty category C (for categorical feature components) that minimises an objective loss function $\mathcal{L}$ after the splitting. We consider only continuous features as an example to illustrate the ideas here, for which the minimization problem is given as:

$$\min_{1\leq l\leq p} \min_{c\in \mathrm{CP}_l} \left[\sum_{i:\boldsymbol{x}_i\in\mathcal{B}_{\eta_0}, x_{il}<c} \mathcal{L}\left(\boldsymbol{y}_i, \hat{\theta}_{\eta_1}\right) + \sum_{i:\boldsymbol{x}_i\in\mathcal{B}_{\eta_0}, x_{il}\geq c} \mathcal{L}\left(\boldsymbol{y}_i, \hat{\theta}_{\eta_2}\right) \right], \tag{2.2}$$

where CP_l is a set of available cut points of x_{il}. Given l and c, the values of $\hat{\theta}_{\eta_1}$ and $\hat{\theta}_{\eta_2}$ should be investigated, and the inner optimizations in the brackets can be obtained using the Maximum Likelihood Estimation (MLE), that is, for $h = 1, 2$,

$$\hat{\theta}_{\eta_h} = \underset{\theta_h}{\operatorname{argmin}} \sum_{i:\boldsymbol{x}_i\in\mathcal{B}_{\eta_h}} \mathcal{L}\left(\boldsymbol{y}_i, \theta_h\right), \tag{2.3}$$

where $\mathcal{B}_{\eta_1} = \{\boldsymbol{x}_i \in \mathcal{B}_{\eta_0}, x_{il} < c\}$ and $\mathcal{B}_{\eta_2} = \{\boldsymbol{x}_i \in \mathcal{B}_{\eta_0}, x_{il} \geq c\}$. After this split, η_1 and η_2 become the current terminal nodes that can be grown, and the procedure can be carried out in the same way as before. Following this process, the procedure is repeated until all current terminal nodes meet the specified step-splitting criteria (e.g., minimal data requirements).

Based on the above description, the general procedure for tree generation can be summarized in Algorithm 2.1. The second step, namely, the prune step, begins once the full tree is grown since the full tree may be too large, leading to over-fitting and unnecessary model complexity. Pruning is the process of gradually trimming the terminal nodes of the tree to obtain a sequence of sub-trees. Several related methods have been proposed in the literature to choose the optimal sub-tree during the pruning process; see, e.g., Breiman *et al.* (1984) and Timofeev (2004). The most commonly used approach is to apply the regularized risk estimate to control the complexity and select the optimal sub-tree,

$$R\left(\mathcal{T}, \xi\right) = R\left(\mathcal{T}\right) + \xi \left|\mathcal{T}\right|,$$

where $R\left(\mathcal{T}\right)$ represents the goodness of fit of the tree, typically calculated as the sum of the loss function values across all terminal nodes in the regression setting

or the sum of the impurity function values in the case of classification; $\xi \geq 0$ is a regularization parameter, controlling the trade-off between the number of terminal nodes denoted as $|\mathcal{T}|$ and its goodness of fit to the data. A larger value of ξ results in a greater penalty on the number of terminal nodes, leading to a smaller tree. Setting $\xi = 0$ results in the full unpruned tree. The optimal ξ can be determined by using cross-validation in minimizing the regularized risk $R(\mathcal{T}, \xi)$.

Algorithm 2.1: Tree generating algorithm

Input: Data $(\boldsymbol{X}, \boldsymbol{y})$ and root node η_0 of the tree $\mathcal{T}$.

1: Investigate every splitting rule of the form $x_l < c$ $(l = 1, 2, \ldots, p)$ or $x_l \in C$ according to whether x_l is a continuous or a categorical explanatory variable at the root node η_0 (or any current terminal node η in the following reapplied steps).

2: Select and execute the split that is considered to be the "best" among all allowable splits, as determined by the choice of a goodness-of-split criterion.

3: Obtain two child nodes of the grown node η_0 (or η) after the splitting.

4: Reapply steps 1 to 3 to any current terminal node that does not meet the specified stop-splitting rules (such as minimal data requirements), until all current terminal nodes satisfy the specified stop-splitting rules, stop splitting.

Output: The full tree $\mathcal{T}$ with partitioned data in each terminal node.

Remark 2.1 *(a) The standard loss function for regression problems is the squared error loss, expressed as* $\mathcal{L}\{y, f(\boldsymbol{x})\} \propto \sum_{i=1}^{n} \big(y_i - f(\boldsymbol{x}_i)\big)^2$*, where* y_i *is the observed response and* $f(\boldsymbol{x}_i)$ *is the prediction of the model for the explanatory variables* $\boldsymbol{x}_i$*. Although this loss is commonly used, it might not be a good choice for modelling integer-valued claims frequency data or right-skewed and heavy-tailed claims severity data. In such cases, deviance is usually preferred, defined as a likelihood ratio:* $D\{y, f(\boldsymbol{x})\} = -2\log\big(\mathcal{L}(f(\boldsymbol{x}))/\mathcal{L}(y)\big)$*, where* $\mathcal{L}(f(\boldsymbol{x}))$ *represents the model likelihood and* $\mathcal{L}(y)$ *is the likelihood of the saturated model (i.e., the model in which the number of parameters equals the number of observations); see, e.g., Ohlsson & Johansson (2010).*

(b) There are two approaches to dealing with categorical feature components in the growing process.

- *Find all possible combinations of the categorical feature levels directly and use them when splitting. The benefit of this method is that there is no numerical*

transformation requirement for the categorical feature component, but a notable drawback is its computational complexity, especially when dealing with a categorical feature component that has numerous levels. For example, if x_l has 10 levels, there would be $2^{10-1} - 1 = 511$ possible combinations.

- *Use an ordered version of the categorical feature component to enhance computational efficiency where numerical transformation is needed. For example, in the claims frequency models, each available categorical level, say k, of x_l in that node, can have its empirical frequency calculated by using ratio of sum of claim numbers and sum of exposures and used as a numerical replacement. A subset C_l is then selected based on the* ordered *empirical frequency values. The same treatment can be adopted for claims severity models, where the empirical severity is calculated by using ratio of sum of claim amounts and sum of claim numbers for numerical replacements. This method can be applied to other models as well. However, it should be noted that this approach may encounter challenges when the categorical variables cannot be ordered in a meaningful way. For instance, if a node does not contain a sufficiently large sample size, there is a risk of producing misleading results because certain categorical feature levels are less likely to observe claims sufficiently, and the ranking provided by the empirical frequency and/or severity may not be accurate.*

2.1.2 A Simple Example: Binary Poisson Regression Trees

To gain a deeper understanding of the application of CART in the insurance industry and for its comparison to the subsequently introduced Bayesian CART, this section demonstrates how the Poisson distribution in claims frequency models works in decision trees as a simple example. Based on this illustration, other distributions can also be put into practice.

Consider a claims data set with n policyholders $\mathcal{D} = (\boldsymbol{X}, \boldsymbol{v}, \boldsymbol{N}) = \big((\boldsymbol{x}_1, v_1, N_1), \ldots, (\boldsymbol{x}_n, v_n, N_n)\big)^\top$, where N_i is the number of claims reported, assumed to follow a Poisson distribution for $i = 1, 2, \cdots, n$; $v_i \in (0, 1]$ is the exposure, typically in yearly units, used to quantify how long the policyholder is exposed to risk. For claims frequency analysis, the goal is to explain and predict the claims information N_i based on the rating variables $\boldsymbol{x}_i$ and the exposure v_i for each individual policyholder i, leading to the *claims frequency*, i.e., the number of claims filed per unit year of exposure to risk. Given a regression function $\lambda : \mathcal{X} \to \mathbb{R}^+$, the expected claim counts can be obtained, $\mathbb{E}(N_i) = \lambda(\boldsymbol{x}_i) v_i$. In CART, an algorithm is

created that partitions the feature space $\mathcal{X}$ into disjoint (homogeneous) subspaces $\mathcal{A}_t$. On each subspace $\mathcal{A}_t$, a frequency parameter $\hat{\lambda}_t$ is estimated to describe the expected frequency. Eventually, the (unknown) expected frequency on the total feature space $\mathcal{X}$ can be calculated by

$$\boldsymbol{x} \mapsto \hat{\lambda}(\boldsymbol{x}) = \sum_{t=1}^{b} \hat{\lambda}_t I_{\{\boldsymbol{x} \in \mathcal{A}_t\}}. \tag{2.4}$$

In light of the discussion in the previous subsection, for obtaining a good partition, Poisson deviance needs to be minimized specifically; see, e.g., Wuthrich & Buser (2022). We illustrate the idea for continuous covariates here only, and the same treatment can be used for categorical covariates. In this case, (2.2) becomes

$$\min_{1 \le l \le p} \min_{c \in \mathrm{CP}_l} \left[\sum_{i: \boldsymbol{x}_i \in \mathcal{B}_\eta, x_{il} < c} D\left(N_i, \hat{\lambda}_{\eta_1}\right) + \sum_{i: \boldsymbol{x}_i \in \mathcal{B}_\eta, x_{il} \ge c} D\left(N_i, \hat{\lambda}_{\eta_2}\right) \right],$$

where

$$\begin{aligned} D(N_i, \lambda) &= 2\left(-N_i + N_i \log N_i + \lambda v_i - N_i \log\left(\lambda v_i\right)\right) \\ &= 2N_i \left(\frac{\lambda v_i}{N_i} - 1 - \log\left(\frac{\lambda v_i}{N_i}\right) \right) \ge 0 \end{aligned}$$

is the Poisson deviance loss of $(\boldsymbol{x}_i, v_i, N_i)$ for the expected frequency $\lambda > 0$, and it is set equal to $2\lambda v_i$ for $N_i = 0$ (we interpret $0 \log(0)$ as 0); CP_l has the same meaning as before, i.e., a set of available cut points of x_{il}. By optimizing the above equation, the values of $\hat{\lambda}_{\eta_1}$ and $\hat{\lambda}_{\eta_2}$ can be obtained, for $h = 1, 2$,

$$\hat{\lambda}_{\eta_h} = \underset{\lambda_h > 0}{\operatorname{argmin}} \sum_{i: \boldsymbol{x}_i \in \mathcal{B}_{\eta_h}} D\left(N_i, \lambda_h\right) = \frac{\sum_{i: \boldsymbol{x}_i \in \mathcal{B}_{\eta_h}} N_i}{\sum_{i: \boldsymbol{x}_i \in \mathcal{B}_{\eta_h}} v_i}. \tag{2.5}$$

With the formulas derived above, we can obtain the full Poisson CART, together with Algorithm 2.1 and then use the prune step to obtain an optimal subtree through the cross-validation method (see Subsection 2.1.1). However, even if prune steps can be used to make the tree smaller, CART will still probably overfit (i.e., be too complicated), and the tree structure can become unstable even for a small change in the data set. Adding a prior for the tree is a good way to prevent over-fitting. Additionally, based on (2.5), a common issue that often arises is that in certain terminal nodes t, we may obtain an estimator $\hat{\lambda}_t$ that is equal to zero. This occurs particularly when the expected frequency and the overall volume in the t-th node are small. If the actual value λ_t is greater than zero, but the estimator $\hat{\lambda}_t$ equals zero, we obtain a degenerate Poisson distribution in the t-th node,

which cannot provide useful information from a practical standpoint; see some related discussion in Wuthrich & Buser (2022). Therefore, a Bayesian estimator of λ_t that considers prior information should be used to avoid this problem. Due to these obvious shortcomings of CART, it is necessary to apply Bayesian technology in CART, leading to Bayesian CART models.

2.2 General Theory of Bayesian CART

In this section, we shall first briefly review the BCART framework of the seminal paper, Chipman *et al.* (1998). Afterwards, we introduce an extension with data augmentation and a model selection method using DIC.

Given that $(\boldsymbol{\theta}, \mathcal{T})$ determines a CART model, a Bayesian analysis of the problem is conducted by specifying a prior distribution $p(\boldsymbol{\theta}, \mathcal{T})$, and inference about $\boldsymbol{\theta}$ and $\mathcal{T}$ will be based on the joint posterior $p(\boldsymbol{\theta}, \mathcal{T}|\boldsymbol{y}, \boldsymbol{X})$ using a suitable MCMC algorithm. Since $\boldsymbol{\theta}$ indexes the parametric model whose dimension depends on the number of terminal nodes of the tree, it is usually convenient to apply the relationship

$$p(\boldsymbol{\theta}, \mathcal{T}) = p(\boldsymbol{\theta} \mid \mathcal{T})p(\mathcal{T}) \tag{2.6}$$

and specify the tree prior distribution $p(\mathcal{T})$ and the terminal node parameter prior distribution $p(\boldsymbol{\theta} \mid \mathcal{T})$, respectively. This strategy, introduced by George (1998), offers several advantages for Bayesian model selection as outlined in Chipman *et al.* (1998).

2.2.1 Prior Choice

Specification of Tree Prior $p(\mathcal{T})$

The prior distribution for $\mathcal{T}$ has two components: a tree topology, and a decision rule for each of the internal/branch nodes. We shall adopt the branching process prior for the topology of $\mathcal{T}$ proposed by Chipman *et al.* (1998). Due to its computational effectiveness using Metropolis-Hastings (MH) search algorithms, this prior specification has been the most popular in the literature. Let

$$p(d) = \gamma\,(1+d)^{-\rho} \tag{2.7}$$

be the probability that a node at depth d splits, where $0 < \gamma \leq 1, \rho \geq 0$ are parameters controlling the structure and size of the tree. To draw from this prior, for each node at depth d (with $d = 0$ for the root node), generate two child

nodes with the probability $p(d)$. This process iterates for $d = 0, 1, \ldots,$ until we reach a depth at which all the nodes cease growing. Note that $p(d)$ is not a probability mass function, but instead is the probability of a given node at depth d being converted to a branch node. A sufficient condition for the termination of this branching process is that $\rho > 0$, and the case $\rho = 0$ corresponds to the Galton-Watson process; see, e.g., Athreya & Ney (2004). We refer to Linero (2018) for further theoretical discussion of this prior. Clearly, γ controls the overall rate of branching at a node, and the larger ρ becomes, the less likely that deeper nodes will branch, resulting in relatively smaller trees. In Chipman *et al.* (1998), some simulations about the number of terminal nodes associated with the values of the pair (γ, ρ) are carried out, which have been used as a guidance when choosing these parameters to generate trees with a certain number of terminal nodes. After the tree topology is generated, each internal node is associated with a decision rule of the form $x_l < c_l$ or $x_l \in C_l$, where x_l is selected independently and uniformly among the available explanatory variables for each internal node, and the split value c_l or split category subset C_l are selected uniformly among those available for the selected variable x_l. In practice, we only consider the overall set of possible split values to be finite; if the l-th variable is continuous, the grid for the variable is either uniformly spaced or given by a collection of observed quantiles of $\{x_{il}, i = 1, 2, \ldots, n; l = 1, 2, \ldots, p\}$. If the l-th variable is categorical, the split category subset C_l is usually selected uniformly among all possible subsets. However, this approach may not be efficient in the (Bayesian) tree search, particularly when the number of categorical levels of x_l is large. Instead, we shall adopt the same treatment of categorical variables as in the standard CART greedy search algorithm (see Subsection 2.1.1), and a subset C_l will be selected uniformly based on the numerical transformation ordered values. Additionally, we will update the ordered values for categorical levels in each node after each split.

Certainly, the design of the tree prior can be more intricate than the one proposed in Chipman *et al.* (1998). There have been several alternatives discussed in the literature. In a recent contribution, Rocková *et al.* (2020), the convergence of the posterior distribution with a near-minimax concentration rate is studied, where it is shown that the original proposal given by (2.7) does not decay at a fast enough rate to guarantee the optimal rate of convergence. Instead, a sufficient condition for optimality is induced by the following probability

$$p(d) = \gamma^d, \qquad \text{for some } 0 < \gamma < 1/2.$$

Most recently, it is noted in Saha (2023) that the original proposal (2.7) can still offer better empirical solutions. We believe further theoretical and empirical studies in this direction are still needed. An alternative to the branching process prior is to specify a prior directly on the number of leaves and a conditionally-uniform prior on the space of trees. In Denison *et al.* (1998), a Poisson-distributed prior is used for the number of leaves, and then a uniform prior over valid trees (i.e., trees with no empty bottom leaves) with that number of leaves is imposed. As noticed by Wu *et al.* (2007), the uniform prior over valid trees in Denison *et al.* (1998) tends to produce more unbalanced trees than balanced ones. Instead, they propose a pinball prior which can generate balanced or skewed trees by adjusting a hyper-parameter. Furthermore, instead of uniformly selecting the split value, the Normal distribution is used for the split value in their simulation and real data analyses (see Wu *et al.* (2007)). Recently, some other tree priors have also been introduced for the purpose of variable selection (particularly when $p > n$); see, e.g., Bleich *et al.* (2014), Linero (2018), Rocková *et al.* (2020) and Liu *et al.* (2021). In Linero (2018), the author proposes a sparsity-inducing Dirichlet prior for the splitting proportions of the explanatory variables, resulting in this prior allows the model to perform a fully Bayesian variable selection. Furthermore, in Rocková *et al.* (2020) and Liu *et al.* (2021), a spike-and-tree variant is proposed by injecting one more layer on top of the prior used in Denison *et al.* (1998), that is, a prior over the active set of explanatory variables.

In our current implementation, we adopt the uniform specification for both variable and split value in each of the internal nodes, which is natural and simple. It is also noted in Chipman *et al.* (1998) that it would be beneficial to incorporate expert knowledge on the prior specification (i.e., using a non-uniform prior), however, our simulation studies in Subsection 3.5.1 show that using the uniform prior is able to identify the correct splitting rules even in the presence of noise variables. This seems to be a consequence of the MH random search steps, which tends to not accept noise splitting variables. We refer to Bleich *et al.* (2014) for some relevant discussions with the same conclusion.

Specification of the Terminal Node Parameter Prior $p(\boldsymbol{\theta} \mid \mathcal{T})$

When choosing $p(\boldsymbol{\theta} \mid \mathcal{T})$, it is important to realize that using priors that allow for analytical simplification can greatly reduce the computational burden of posterior calculation and exploration. This is especially true for prior form $p(\boldsymbol{\theta} \mid \mathcal{T})$ for

which it is possible to analytically margin out $\boldsymbol{\theta}$ to obtain the integrated likelihood

$$\begin{aligned} p(\boldsymbol{y} \mid \boldsymbol{X}, \mathcal{T}) &= \int p(\boldsymbol{y} \mid \boldsymbol{X}, \boldsymbol{\theta}, \mathcal{T}) p(\boldsymbol{\theta} \mid \mathcal{T}) d\boldsymbol{\theta} \\ &= \prod_{t=1}^{b} \int f\left(\boldsymbol{y}_t \mid \boldsymbol{\theta}_t\right) p(\boldsymbol{\theta}_t) d\boldsymbol{\theta}_t \\ &= \prod_{t=1}^{b} \int \prod_{j=1}^{n_t} f\left(y_{tj} \mid \boldsymbol{\theta}_t\right) p(\boldsymbol{\theta}_t) d\boldsymbol{\theta}_t, \end{aligned} \tag{2.8}$$

where in the second equality we assume that conditional on the tree $\mathcal{T}$ with b terminal nodes, the parameters $\boldsymbol{\theta}_t, t = 1, 2, \ldots, b$, have IID priors $p(\boldsymbol{\theta}_t)$, which is a common assumption. Examples where this integration has a closed-form expression can be found in, e.g., Chipman *et al.* (1998) and Linero (2017), particularly for Gaussian distributed data $\boldsymbol{y}$. When no such priors can be found, we have to resort to the data augmentation technique (see, e.g., Kindo *et al.* (2016), Linero *et al.* (2020) and Murray (2021)) which will be discussed later.

2.2.2 MCMC

Constructing efficient algorithms for stochastically searching posterior trees and parameters is a significant challenge when implementing BCART models, especially for multidimensional parameters $\boldsymbol{\theta}$. It is rarely possible to derive the posterior distribution of $\boldsymbol{\theta}$ analytically, or even to compute summary statistics of interest such as the mode, mean and variance of the posterior distribution. Therefore, for practical applications, MCMC should be developed to estimate the parameters (posterior distribution or summary statistics). As the name suggests, MCMC has two components, Markov chain and Monte Carlo methods. In a broad sense, Monte Carlo methods generate (simulate) a sequence of IID samples as a tool to investigate the behaviour of statistical models, which crucially requires a large number of samples. When there is difficulty in generating samples from a given distribution, the IID sequence is replaced with a Markov chain that explores the state space of the target distribution (referred to as the posterior distribution in Bayesian statistics), and this approach is known as MCMC. The key idea is to create a memoryless stochastic process where the next state depends only on its current state. By iteratively transitioning between states in a way that maintains the desired distribution, the Markov chain eventually converges to a stationary distribution, which is ideally the target distribution of interest. The samples obtained through this process can be used to estimate various properties and perform

statistical inference for models with intractable likelihoods or high-dimensional parameter spaces. There are many different MCMC algorithms (see, e.g., Roberts & Rosenthal (2004)), and we concentrate on the MH algorithm in the following content.

Combining the integrated likelihood $p(\boldsymbol{y} \mid \boldsymbol{X}, \mathcal{T})$ with tree prior $p(\mathcal{T})$, allows us to calculate the posterior of $\mathcal{T}$

$$p(\mathcal{T} \mid \boldsymbol{X}, \boldsymbol{y}) \propto p(\boldsymbol{y} \mid \boldsymbol{X}, \mathcal{T}) p(\mathcal{T}). \tag{2.9}$$

When using the MH algorithm to conduct Bayesian inference, $\mathcal{T}$ can be updated with the right-hand side of (2.9) used to compute the acceptance ratio. These MH simulations can be used to stochastically search the posterior space over trees to determine the high posterior probability trees from which we can choose a best one. The posterior sequence for $\boldsymbol{\theta}$ is then obtained using an additional Gibbs sampler. Starting from the root node, the MCMC algorithm for simulating a Markov chain sequence of pairs $(\boldsymbol{\theta}^{(1)}, \mathcal{T}^{(1)}), (\boldsymbol{\theta}^{(2)}, \mathcal{T}^{(2)}), \ldots$, using the posterior given in (2.9), is given in Algorithm 2.2.

Algorithm 2.2: One step of the MCMC algorithm for updating the BCART models parameterized by $(\boldsymbol{\theta}, \mathcal{T})$

Input: Data $(\boldsymbol{X}, \boldsymbol{y})$ and current values $(\boldsymbol{\theta}^{(m)}, \mathcal{T}^{(m)})$

1: Generate a candidate value $\mathcal{T}^*$ with probability distribution $q(\mathcal{T}^{(m)}, \mathcal{T}^*)$

2: Set the acceptance ratio $\alpha(\mathcal{T}^{(m)}, \mathcal{T}^*) = \min\left\{ \frac{q(\mathcal{T}^*, \mathcal{T}^{(m)}) p(\boldsymbol{y} \mid \boldsymbol{X}, \mathcal{T}^*) p(\mathcal{T}^*)}{q(\mathcal{T}^{(m)}, \mathcal{T}^*) p(\boldsymbol{y} \mid \boldsymbol{X}, \mathcal{T}^{(m)}) p(\mathcal{T}^{(m)})}, 1 \right\}$

3: Update $\mathcal{T}^{(m+1)} = \mathcal{T}^*$ with probability $\alpha(\mathcal{T}^{(m)}, \mathcal{T}^*)$, otherwise, set $\mathcal{T}^{(m+1)} = \mathcal{T}^{(m)}$

4: Sample $\boldsymbol{\theta}^{(m+1)} \sim p(\boldsymbol{\theta} \mid \boldsymbol{X}, \boldsymbol{y}, \mathcal{T}^{(m+1)})$

Output: New values $(\boldsymbol{\theta}^{(m+1)}, \mathcal{T}^{(m+1)})$

In Algorithm 2.2, commonly used arbitrary but specified proposals (or transitions) for $q(\cdot, \cdot)$ include grow, prune, change and swap (see Chipman *et al.* (1998)), which are usually selected with equal probability (i.e., 1/4 each); see Figure 2.1. Other proposals have been suggested to improve the mixing of simulated trees, but these are often difficult to put into practice; see, e.g., Wu *et al.* (2007) and Pratola (2016). One of the appealing features of these four proposals is that grow and prune steps are reversible counterparts of one another and both change and swap steps are independently reversible. As noticed in Chipman *et al.* (1998),

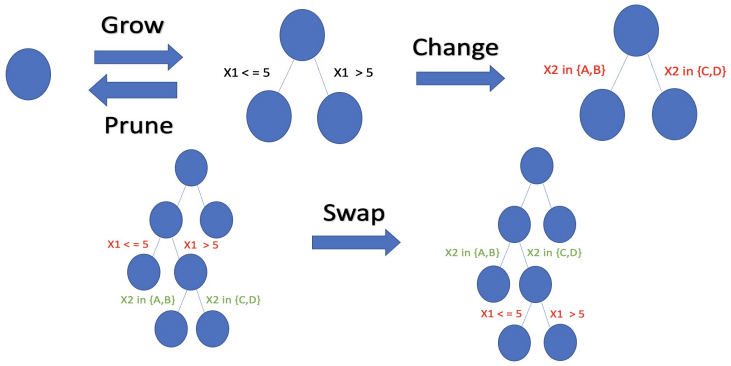

Figure 2.1: Four tree moves.

this is very attractive for the calculation of $\alpha(\mathcal{T}^{(m)}, \mathcal{T}^*)$ in Algorithm 2.2, since there are substantial cancellations in the ratio; for completeness, we include the detailed calculations in Appendix A. In our implementation, we consider these four proposals at each split detailed as follows.

- Grow: Randomly select a terminal node. Split it into two new child nodes and randomly assign it a decision rule according to the prior specified in Subsection 2.2.1 until the resulting two child nodes satisfy a minimum observation requirement. If no such decision rule exists, draw a new terminal node (without replacement) and try again. If no such terminal node exists, stop grow.

- Prune: A terminal node is randomly selected. The chosen node and its sibling node are pruned into the direct parent node which then becomes a new terminal node.

- Change: Apply one of the following two types of change to a randomly selected internal node.

  – Change1: Reassign randomly only the split value/category subset according to the prior specified in Subsection 2.2.1.

  – Change2: Reassign randomly both the splitting variable and the corresponding split value/category subset according to the prior specified in Subsection 2.2.1.

  In each of the above changes, randomly select an internal node with the reassignment selected at random from a set (without replacement) until the updated nodes satisfy the minimum observation requirement. If no such

reassignment exists, draw a new internal node (without replacement) and try again. If no such internal node exists, stop change.

- Swap: Randomly pick a parent-child pair which are both internal nodes and swap their decision rules the updated nodes satisfy the minimum observation requirement. If no such parent-child pair exists, stop swap.

Remark 2.2 *(a) Note that in step 4 of Algorithm 2.2, sampling of* $\boldsymbol{\theta}^{(m+1)}$ *is needed only for those nodes that were involved in the proposed move from* $\mathcal{T}^{(m)}$ *to* $\mathcal{T}^*$ *and only when this move was accepted.*

(b) In comparison to Chipman et al. *(1998), we apply two types of change moves as discussed in Denison* et al. *(1998). The introduction of these two types of change is helpful to improve the mixing of posterior trees, demonstrated by our simulation study in Subsection 3.5.1. Besides, there are two special cases in the swap move.*

- *When two child nodes have the same splitting rule, Chipman* et al. *(1998) proposed to swap the parent's splitting rule with that of both children. We follow this strategy in our implementation.*

- *A swap between a parent-child pair with splits using the same variable is impossible. Considering this in our implementation will improve the computational efficiency.*

The MH algorithm is constructed to sample from a target density with a fixed number of dimensions. Another MCMC algorithm, known as Reversible Jump MCMC (RJMCMC), is constructed for "dimension jumping", allowing movement around the parameter space of a collection of different size models; see, e.g., Green (1995) and Green & Hastie (2009). RJMCMC can be viewed as a generalization of the MH algorithm. This algorithm combines the standard MCMC algorithm for a given model with an additional step that involves moving between different models. Since the state space of the Markov chain changes size when adding or removing nodes in the tree models, that is, when the tree grows or prunes, RJMCMC should be taken into consideration. In BCART models, the difference between MH and RJMCMC lies in the calculation of the acceptance ratio. We have demonstrated that the results of RJMCMC align with the MH algorithm; see detailed calculations in Appendix B. Additionally, it is worth noting that by integrating out $\boldsymbol{\theta}$ in (2.8) we avoid the possible complexities associated with reversible jumps between continuous spaces of varying dimensions; see, e.g., Chipman *et al.* (2010) and Green (1995). Therefore, in our implementation, we do not follow the route of RJMCMC and exclusively focus on the MH algorithm.

2.2.3 MCMC Algorithm with Data Augmentation

In this subsection, we discuss the case where there is no obvious prior distribution $p(\boldsymbol{\theta}_t)$ such that the integration in (2.8) is of closed-form, particularly, for non-Gaussian data $\boldsymbol{y}$. In this case, we shall use a data augmentation method in implementing the MCMC algorithm. Some special cases have been discussed in Chipman *et al.* (2010), Kindo *et al.* (2016), Linero *et al.* (2020) and Murray (2021).

The term data augmentation originated from Tanner and Wong's data augmentation algorithm; see, e.g., Tanner & Wong (1987). It is introduced purely for computational purposes and a latent variable is required so that the original distribution is the marginal distribution of the augmented one. We refer to Van Dyk & Meng (2001) for an overview of data augmentation and relevant theory. For our purpose, we augment the data $\boldsymbol{y}$ by introducing a latent variable $\boldsymbol{z} = (z_1, z_2, \ldots, z_n)$ with n observations, so that the integration in (2.11) below is computable for augmented data $(\boldsymbol{y}, \boldsymbol{z})$. To this end, we shall follow the idea of marginal augmentation introduced in Meng & Van Dyk (1999) (see also Van Dyk & Meng (2001)). In their framework, our parameter $\boldsymbol{\theta}$ can be interpreted as a working parameter, and thus the integrated likelihood is given as

$$p(\boldsymbol{y} \mid \boldsymbol{X}, \mathcal{T}) = \int p(\boldsymbol{y}, \boldsymbol{z} \mid \boldsymbol{X}, \mathcal{T}) d\boldsymbol{z}, \tag{2.10}$$

where

$$\begin{aligned} p(\boldsymbol{y}, \boldsymbol{z} \mid \boldsymbol{X}, \mathcal{T}) &= \int p(\boldsymbol{y}, \boldsymbol{z} \mid \boldsymbol{X}, \boldsymbol{\theta}, \mathcal{T}) p(\boldsymbol{\theta} \mid \mathcal{T}) d\boldsymbol{\theta} \\ &= \prod_{t=1}^{b} \int f\left(\boldsymbol{y}_t, \boldsymbol{z}_t \mid \boldsymbol{\theta}_t\right) p(\boldsymbol{\theta}_t) d\boldsymbol{\theta}_t \\ &= \prod_{t=1}^{b} \int \prod_{j=1}^{n_t} f\left(y_{tj}, z_{tj} \mid \boldsymbol{\theta}_t\right) p(\boldsymbol{\theta}_t) d\boldsymbol{\theta}_t, \end{aligned} \tag{2.11}$$

with $\boldsymbol{z}_t = (z_{t1}, z_{t2}, \ldots, z_{tn_t})$ defined according to the partition of $\mathcal{X}$ and with obvious independence assumed. Following Scheme 3 of Meng & Van Dyk (1999) (see also Section 3 of Van Dyk & Meng (2001)), we propose the following Algorithm 2.3 to simulate a Markov chain sequence of pairs $(\boldsymbol{\theta}^{(1)}, \mathcal{T}^{(1)}), (\boldsymbol{\theta}^{(2)}, \mathcal{T}^{(2)}), \ldots$, starting from the root node.

Note that in some cases introducing one latent variable $\boldsymbol{z}$ is insufficient to obtain a closed-form for the integration in (2.11); more latent variables may be required. In that case, we can easily extend Algorithm 2.3 to include multivariate latent variables and use the Gibbs sampler in step 2. Clearly, the more latent

Algorithm 2.3: One step of the MCMC algorithm for updating the BCART models parameterized by $(\boldsymbol{\theta}, \mathcal{T})$ using data augmentation

Input: Data $(\boldsymbol{X}, \boldsymbol{y})$ and current values $(\boldsymbol{\theta}^{(m)}, \boldsymbol{z}^{(m)}, \mathcal{T}^{(m)})$

1: Generate a candidate value $\mathcal{T}^*$ with probability distribution $q(\mathcal{T}^{(m)}, \mathcal{T}^*)$

2: Propose $\boldsymbol{z}^{(m+1)} \sim p(\boldsymbol{z} \mid \boldsymbol{X}, \boldsymbol{y}, \boldsymbol{\theta}^{(m)}, \mathcal{T}^{(m)})$

3: Set the acceptance ratio

$$\begin{aligned}&\alpha(\mathcal{T}^{(m)}, \mathcal{T}^*)\\ &= \min\left\{\frac{q(\mathcal{T}^*, \mathcal{T}^{(m)})p(\boldsymbol{y}, \boldsymbol{z}^{(m+1)} \mid \boldsymbol{X}, \mathcal{T}^*)p(\mathcal{T}^*)}{q(\mathcal{T}^{(m)}, \mathcal{T}^*)p(\boldsymbol{y}, \boldsymbol{z}^{(m)} \mid \boldsymbol{X}, \mathcal{T}^{(m)})p(\mathcal{T}^{(m)})}, 1\right\}\end{aligned}$$

4: Update $\mathcal{T}^{(m+1)} = \mathcal{T}^*$ with probability $\alpha(\mathcal{T}^{(m)}, \mathcal{T}^*)$, otherwise, set $\mathcal{T}^{(m+1)} = \mathcal{T}^{(m)}$

5: Sample $\boldsymbol{\theta}^{(m+1)} \sim p(\boldsymbol{\theta} \mid \boldsymbol{X}, \boldsymbol{y}, \boldsymbol{z}^{(m+1)}, \mathcal{T}^{(m+1)})$

Output: New values $(\boldsymbol{\theta}^{(m+1)}, \boldsymbol{z}^{(m+1)}, \mathcal{T}^{(m+1)})$

variables used, the slower the convergence of the Markov chain sequence. As discussed in Van Dyk & Meng (2001), it is an "art" to search for efficient data augmentation schemes. We discuss this point in the following chapters for the claims frequency models and aggregate claims models specifically.

Remark 2.3 *Similar to Algorithm 2.2, in step 2 and step 5 of Algorithm 2.3 the sampling is needed only for those nodes that were involved in the proposed move from $\mathcal{T}^{(m)}$ to $\mathcal{T}^*$, and step 5 is needed only when this move was accepted.*

2.2.4 Posterior Tree Selection and Prediction

The MCMC algorithms described in the previous subsection can be used to search for desirable trees. This subsection shall introduce the strategy of selecting an "optimal" tree among all visited trees in MCMC algorithms and obtaining predictions for new data.

As discussed in Chipman *et al.* (1998) and illustrated below in our analysis, the algorithms quickly converge and then move locally in that region for a long time, which occurs because proposals make local moves over a sharply peaked multimodal posterior. Instead of making long runs of search to move from one mode to another better one, we follow the idea of Chipman *et al.* (1998) to repeatedly restart the algorithm. As many trees are visited by each run of the algorithm,

we need a method to identify those trees which are of most interest. Moreover, the structure of trees in the convergence regions is mostly determined by the hyper-parameters γ, ρ (see (2.7)) which also need to be chosen appropriately. In Chipman *et al.* (1998), the integrated likelihood $p(\boldsymbol{y} \mid \boldsymbol{X}, \boldsymbol{T})$ is used as a measure to choose good trees from one run of the algorithm, though other measures, like the residual sum of squares (RSS), could also be introduced. However, there is no discussion on how the tree prior hyper-parameters γ, ρ should be determined optimally. A natural way to deal with this is to use cross-validation which, however, requires repeated model fits and is very computationally expensive. We propose to use DIC for choosing appropriate γ, ρ, and thus introduce a three-step approach for selecting an "optimal" tree among those visited. To this end, we first give a definition of DIC for a Bayesian CART. We refer to Spiegelhalter *et al.* (2002), Celeux *et al.* (2006), Gelman *et al.* (2014) and Spiegelhalter *et al.* (2014) for a more detailed discussion of DIC and its extensions.

Consider the tree $\mathcal{T}$ with b terminal nodes and parameters $\boldsymbol{\theta}_t$ $(t = 1, 2, \ldots, b)$, as previously defined. We first introduce the DIC for each node using the standard definition, the DIC for the tree is then defined as the sum of the DIC of all terminal nodes in the tree due to the independence assumption. For node t, we call

$$D(\boldsymbol{\theta}_t) = -2\log(f(\boldsymbol{y}_t \mid \boldsymbol{\theta}_t)) = -2\sum_{j=1}^{n_t} \log(f(y_{tj} \mid \boldsymbol{\theta}_t)) \tag{2.12}$$

the *deviance*, where the deviance is conditional on the parameter vector $\boldsymbol{\theta}$, which refers to as the "parameters of interest" or "parameters in focus".

Analogously to the Akaike's information criterion (AIC), Spiegelhalter *et al.* (2002) proposed the DIC based on the principle DIC="goodness of fit"+"complexity", which is defined as

$$\mathrm{DIC}_t = D(\bar{\boldsymbol{\theta}}_t) + 2p_{Dt},$$

where $\bar{\boldsymbol{\theta}}_t = \mathbb{E}_{\text{post}}(\boldsymbol{\theta}_t)$ is the posterior mean (with $\mathbb{E}_{\text{post}}$ denoting expectation over the posterior distribution of $\boldsymbol{\theta}$ given data $\boldsymbol{y}$), and p_{Dt} is the *effective number of parameters* given by

$$\begin{aligned} p_{Dt} &= \overline{D(\boldsymbol{\theta}_t)} - D(\bar{\boldsymbol{\theta}}_t) \\ &= -2\mathbb{E}_{\text{post}}\left(\log(f(\boldsymbol{y}_t \mid \boldsymbol{\theta}_t))\right) + 2\log(f(\boldsymbol{y}_t \mid \bar{\boldsymbol{\theta}}_t)) \\ &= 2\sum_{j=1}^{n_t}\Big(\log(f(y_{tj} \mid \bar{\boldsymbol{\theta}}_t)) - \mathbb{E}_{\text{post}}\left(\log(f(y_{tj} \mid \boldsymbol{\theta}_t))\right)\Big). \end{aligned} \tag{2.13}$$

The DIC of the tree $\mathcal{T}$ with b terminal nodes is then defined as

$$\text{DIC} := \sum_{t=1}^{b} \text{DIC}_t = D(\bar{\boldsymbol{\theta}}) + 2p_D, \tag{2.14}$$

where $D(\bar{\boldsymbol{\theta}}) = \sum_{t=1}^{b} D(\bar{\boldsymbol{\theta}}_t)$ and $p_D = \sum_{t=1}^{b} p_{Dt}$ are the deviance and effective number of parameters of the tree.

Next, we introduce DIC for tree models with data augmentation. Depending on whether the latent variable $\boldsymbol{z}$ is treated as a parameter or not, there are three types of likelihoods leading to eight versions of DIC as discussed in Celeux *et al.* (2006). Due to the complexity in implementing any of those eight and motivated by the idea that DIC="goodness of fit"+"complexity", we introduce a new DIC for node t in the tree as follows

$$\text{DIC}_t = D(\bar{\boldsymbol{\theta}}_t) + 2q_{Dt}, \tag{2.15}$$

where $D(\bar{\boldsymbol{\theta}}_t)$ is the deviance defined through the data $\boldsymbol{y}_t$ (as in (2.12)) which represents the goodness of fit, and q_{Dt} is the *effective number of parameters* defined through the augmented data $(\boldsymbol{y}_t, \boldsymbol{z}_t)$ as follows

$$\begin{aligned} q_{Dt} &= -2\mathbb{E}_{\text{post}}\left(\log(f(\boldsymbol{y}_t, \boldsymbol{z}_t \mid \boldsymbol{\theta}_t))\right) + 2\log(f(\boldsymbol{y}_t, \boldsymbol{z}_t \mid \bar{\boldsymbol{\theta}}_t)) \\ &= 2\sum_{j=1}^{n_t}\Big(\log(f(y_{tj}, z_{tj} \mid \bar{\boldsymbol{\theta}}_t)) - \mathbb{E}_{\text{post}}\left(\log(f(y_{tj}, z_{tj} \mid \boldsymbol{\theta}_t))\right)\Big), \end{aligned} \tag{2.16}$$

where $\bar{\boldsymbol{\theta}}_t = \mathbb{E}_{\text{post}}(\boldsymbol{\theta}_t)$, and in this case $\mathbb{E}_{\text{post}}$ denotes expectation over the posterior distribution of $\boldsymbol{\theta}$ given augmented data $(\boldsymbol{y}, \boldsymbol{z})$. By incorporating the augmented data $(\boldsymbol{y}, \boldsymbol{z})$, q_{Dt} can be calculated explicitly and is demonstrated to be effective in our simulation studies. Similarly, the DIC of tree $\mathcal{T}$ with b terminal nodes is thus defined as

$$\text{DIC} = D(\bar{\boldsymbol{\theta}}) + 2q_D, \tag{2.17}$$

where $q_D = \sum_{t=1}^{b} q_{Dt}$.

Remark 2.4 *(a) As we will see in the following chapters, for claims models, the effective number of parameters p_{Dt} or q_{Dt} is approximately the dimension of $\boldsymbol{\theta}_t$ as the sample size n_t in node t tends to infinity.*

(b) $D(\bar{\boldsymbol{\theta}})$ is called the "plug-in" estimate of model deviance. In most cases, the posterior mean of the parameter $\bar{\boldsymbol{\theta}}$ is taken due to the ease of the computation, as in our implementation. However, other "plug-in" estimates of $\boldsymbol{\theta}$, such as the posterior mode or median, can also be used to calculate $D(\bar{\boldsymbol{\theta}})$. Each of them has

unique characteristics, for example, Spiegelhalter et al. (2002) noted that using the posterior mode can be less reliable in flat or multimodal distributions; on the other hand, although the posterior median is robust to outliers and less sensitive to extreme values, making it a good option for skewed distributions, it may not be unique in distributions with flat regions. In conclusion, the choice between them depends on the characteristics of the data and the objectives of the analysis. The mean is often preferred for its statistical properties, while the mode and median are chosen for their robustness in certain situations; see more discussion in Spiegelhalter et al. (2002).

(c) Note that DIC is defined using plug-in prediction densities $f(y_{tj} \mid \bar{\boldsymbol{\theta}}_t)$ *in* (2.13) *(similar to* $f(y_{tj}, z_{tj} \mid \bar{\boldsymbol{\theta}}_t)$ *in* (2.16)*). More recently, a new criterion called Watanabe–Akaike information criterion (WAIC) was introduced by Watanabe & Opper (2010) (see also Gelman et al. (2014) and Spiegelhalter et al. (2014)), where in its definition the plug-in prediction density is replaced by the full prediction density* $\mathbb{E}_{post}(f(y_{tj} \mid \boldsymbol{\theta}_t))$*. When the explicit expression is not available, this posterior expectation is usually computed by a Monte Carlo algorithm as* $S^{-1}\sum_{k=1}^{S} f(y_{tj} \mid \boldsymbol{\theta}^k)$*, where* $\boldsymbol{\theta}^k$ *is simulated from the posterior distribution of* $\boldsymbol{\theta}_t$*. In the following chapters, we will see that this posterior expectation can be obtained explicitly for the Poisson model, but not for other models. It turns out that using WAIC gives the same selected model as DIC in our initial simulation studies. Additionally, as it involves a Monte Carlo algorithm and could be considerably more computationally expensive, we suggest using DIC.*

(d) It is worth noting that if the independence assumption within the terminal nodes is violated (see, e.g., Chipman et al. (2003, 2002)), the DIC may also be used as a tool for model selection but the formulation would not be of the simple summation form as in (2.12)*. We refer to Spiegelhalter et al. (2002) for examples and relevant discussions.*

(e) It is noteworthy that, in addition to the two DICs discussed above, two more specific DICs will be introduced in Sections 3.2 and 4.2 for other models.

Now, we are ready to introduce the three-step approach for selecting an "optimal" tree from the MCMC algorithms. Let $m_s < m_e$ be two user input integers which represent the belief that the optimal number of terminal nodes lies in $[m_s, m_e]$. In practice, these can be estimated first by using some other methods, e.g., a standard CART model. The three-step approach is described in Table 2.1. In what follows, the tree selected by using the three-step approach will be called an "optimal" tree.

Remark 2.5 *(a) The relation between hyper-parameters* (γ_h, ρ_h) *and the distri-*

Table 2.1: Three-step approach for "optimal" tree selection.

Step 1:	Set a sequence of hyper-parameters $(\gamma_h, \rho_h), h = m_s, \cdots, m_e$, such that for (γ_h, ρ_h), the MCMC algorithm converges to a region of trees which have h terminal nodes.
Step 2:	For each h in Step 1, select the tree with maximum likelihood $p(\boldsymbol{y} \mid \boldsymbol{X}, \bar{\boldsymbol{\theta}}, \mathcal{T})$ from the convergence region.
Step 3:	From the trees obtained in Step 2, select the optimal one using DIC.

bution of the number of terminal nodes of the tree has been illustrated in Chipman et al. (1998). It does not seem hard to set values for (γ_h, ρ_h) so that the MCMC algorithms will converge to a region of trees with required h terminal nodes. It is also worth noting that the distribution of the number of terminal nodes is also affected by the data in hand, which can be seen from the calculation of the acceptance ratio in the MCMC algorithms. In our simulations and real data analyses below, we have to select a relatively larger ρ in order to achieve our goals.

(b) In Step 2, the so-called data likelihood $p(\boldsymbol{y} \mid \boldsymbol{X}, \bar{\boldsymbol{\theta}}, \mathcal{T})$, rather than the integrated likelihood $p(\boldsymbol{y} \mid \boldsymbol{X}, \mathcal{T})$, is used, which is driven by our interest in the fit of the parametric model to data. The simulations and real data in the following chapters indicate that these two types of likelihood show a consistency in the ordering of their values, and thus we suspect there is no big difference using either of them.

Suppose $\mathcal{T}$ with b terminal nodes and parameter $\bar{\boldsymbol{\theta}}$ is the optimal tree obtained from the above three-step approach. For a given new $\boldsymbol{x}$ the predicted $\hat{y}$ using this tree model is defined as

$$\hat{y} \mid \boldsymbol{x} = \sum_{t=1}^{b} \mathbb{E}(y \mid \bar{\boldsymbol{\theta}}_t) I_{(\boldsymbol{x} \in \mathcal{A}_t)}, \tag{2.18}$$

where $I_{(\cdot)}$ denotes the indicator function and $\{\mathcal{A}_t\}_{t=1}^{b}$ is the partition of $\mathcal{X}$.

Remark 2.6 *An alternative prediction given $\boldsymbol{x}$ can be defined using the full predictive density as*

$$\hat{y} \mid \boldsymbol{x} = \sum_{t=1}^{b} \mathbb{E}_{post}(\mathbb{E}(y \mid \boldsymbol{\theta}_t)) I_{(\boldsymbol{x} \in \mathcal{A}_t)}. \tag{2.19}$$

However, for claims models, the explicit expression can be found only for the Poisson case, and for other models, the Monte Carlo method is needed to estimate the posterior expectation. Thus, we shall use (2.18) for simplicity.

2.3 Evaluation Metrics

Upon mastering the methodology for constructing and identifying the optimal tree model based on in-sample performance, such as the DIC proposed before, the next step involves the validation and comparison of different models using out-of-sample data. There are several ways to quantify prediction accuracy; however, there is no single ideal metric that applies universally. In this section, we provide an introduction to a few widely used metrics that will be employed for a comparison between different models before proceeding with the analyses of simulation studies and real data in the following chapters. Each of these evaluation metrics is separately described below, along with its interpretation.

Suppose we have obtained a tree with b terminal nodes and the corresponding parameter estimates which we will use to obtain the prediction $\hat{y}_i$ for a test data set with m observations. The number of test data in terminal node t is denoted by m_t, $t = 1, \ldots, b$. The performance measures used are as follows.

2.3.1 Residual Sum of Squares

Residual Sum of Squares (RSS) can be used to identify the level of discrepancy in a data set that cannot be predicted by the model. The smaller the RSS, the closer the predicted values align with the actual values, indicating a more accurate model. This relationship can be observed in a plot; if there is unexplained variability, the line representing predicted values may not pass through all the actual data points. RSS is given by:

$$\mathrm{RSS}(\boldsymbol{y}) = \sum_{i=1}^{m} (y_i - \hat{y}_i)^2.$$

This measure is commonly used for Gaussian-distributed data, but here we also apply it to non-Gaussian data for comparison purposes.

Although RSS is a commonly used metric with its advantages (sensitivity to model fit, simple interpretation, and good mathematical properties), it is important to be aware of its limitations and consider other evaluation metrics, especially when dealing with data sets that contain large outliers. For example, in insurance claims severity data, RSS places considerable emphasis on outliers, i.e., extremely large claim amounts. This can result in an excessively large RSS, which is undesirable.

2.3.2 Squared Error

Given the discussion above, for the tree models, we propose a specifically designed metric called squared error (SE), based on a sub-portfolio (i.e., those instances in the same terminal node) level, which is defined by

$$\mathrm{SE}(\boldsymbol{\epsilon}) = \sum_{t=1}^{b} (\epsilon_t - \hat{\epsilon}_t)^2 ,$$

where $\epsilon_t/\hat{\epsilon}_t$ is the empirical/estimated claims frequency (or claims severity, claims cost) respectively for terminal node t depending on the type of claims model considered. This estimation $\hat{\epsilon}_t$ is obtained using (2.18), assuming unit exposure for both claims frequency and claims cost, and unit claim number for claims severity. This measure is preferred to RSS in tree models as it takes into account accuracy on a (sub-)portfolio level (i.e., balance property) other than an individual level. We refer to Denuit *et al.* (2021) and Wüthrich (2020, 2022) for more details and discussions of the balance property that is required for insurance pricing.

2.3.3 Discrepancy Statistic

Discrepancy statistic (DS) (cf. Naya *et al.* (2008)), is defined as a weighted version of SE, given by

$$\mathrm{DS}(\boldsymbol{\epsilon}) = \sum_{t=1}^{b} \frac{1}{\hat{\sigma}_t^2} (\epsilon_t - \hat{\epsilon}_t)^2 ,$$

where ϵ_t and $\hat{\epsilon}_t$ are the same as in Subsection 2.3.2, and $\hat{\sigma}_t^2$ is the estimated variance of claims frequency (or claims severity, claims cost) for terminal node t. By considering variance, it is better to assess whether different models have the ability to handle over-dispersed data, as seen in the comparison between Poisson and ZIP distributions in claims frequency models.

2.3.4 Negative Log-Likelihood

Negative log-likelihood (NLL) is calculated by using the assumed response distribution in the terminal node with the estimated parameters from the training process. It represents the ex-ante belief of the underlying distribution of the data, and thus a good measure for model comparison; see, e.g., Lee (2021). However, NLL is not measurable in absolute terms and cannot be directly compared between different models or data sets. Therefore, although NLL is a useful tool, it is beneficial to complement its use with other model comparison criteria to obtain a more comprehensive evaluation of model performance.

2.3.5 Lift

In addition to the commonly used statistical indicators described above, we introduce a specific metric that enhances our understanding of the "economic value" of the model, known as lift. Model lift indicates the ability to differentiate between low and high claims frequency (or claims severity, claims cost) policyholders without assuming an underlying distribution. A higher lift illustrates that the model is more capable of separating the extreme values from the average. Since lift focuses on the tails of the distribution, this metric enables actuaries to effectively construct risk mitigation plans using mechanisms beyond pricing, such as underwriting, reinsurance, and enforcement of safety measures. For example, if the model demonstrates an extraordinary ability to identify policyholders with high claims frequency, insurers may take more targeted risk management actions. We refer to Henckaerts *et al.* (2021), Lee (2020, 2021) and references therein for further discussion on lift. We propose a way to calculate lift for the tree model in the following steps.

Step 1: Retrieve the predicted claims frequencies (or claims severities, claim costs) $\hat{\epsilon}_t$, for terminal nodes $t = 1, \ldots, b$, for the optimal tree obtained from the training procedure.

Step 2: Set $\hat{\epsilon}_{\min} = \min_{t=1}^{b} \hat{\epsilon}_t$ and $\hat{\epsilon}_{\max} = \max_{t=1}^{b} \hat{\epsilon}_t$, which identify the least and most risky groups of policyholders, respectively.

Step 3: Use test data in the least and most risky groups/nodes to obtain their total sum of volumes, say $e_{\min}$ and $e_{\max}$. More specifically, volume refers to exposure in both claims frequency and aggregate claims models; volume refers to the number of claims in claims severity models.

Step 4: If $e_{\min} \leq e_{\max}$, then sort the data using volumes in descending order in the most risky group. Calculate the cumulative sums of the sorted volumes until the one equal or greater than $e_{\min}$ is achieved and then calculate the corresponding empirical frequency, i.e., ratio of sum of claim numbers and sum of exposures, (or empirical severity referred to as ratio of sum of claim amounts and sum of claim numbers; empirical cost referred to as ratio of sum of claim amounts and sum of exposures) of these first data involved, say $\lambda^{(em)}_{\max|}$ (or $a^{(em)}_{\max|}$, $\lambda^{(em)}_{\max|} a^{(em)}_{\max|}$); see Chapters 3-5 for more specific mathematical details. The lift is defined as $L = \lambda^{(em)}_{\max|} / \lambda^{(em)}_{\min}$ (for claims frequency models); $L = a^{(em)}_{\max|} / a^{(em)}_{\min}$ (for claims severity models); $L = \lambda^{(em)}_{\max|} a^{(em)}_{\max|} / \lambda^{(em)}_{\min} a^{(em)}_{\min}$ (for

aggregate claims models), where $\lambda_{\text{min}}^{(em)}$ is the empirical frequency of the least risky group; $a_{\text{min}}^{(em)}$ and $\lambda_{\text{min}}^{(em)} a_{\text{min}}^{(em)}$ are defined analogously.

[Similarly, If $e_{\text{min}} > e_{\text{max}}$, then sort the data using volumes in ascending order in the least risky group. Calculate the cumulative sums of the sorted volumes until the one equal or greater than e_{max} is achieved and then calculate the corresponding empirical frequency (or empirical severity, empirical cost) of these first data involved, say $\lambda_{\text{min}|}^{(em)}$ (or $a_{\text{min}|}^{(em)}$, $\lambda_{\text{min}|}^{(em)} a_{\text{min}|}^{(em)}$). The lift is defined as $L = \lambda_{\text{max}}^{(em)} / \lambda_{\text{min}|}^{(em)}$ (for claims frequency models); $L = a_{\text{max}}^{(em)} / a_{\text{min}|}^{(em)}$ (for claims severity models); $L = \lambda_{\text{max}}^{(em)} a_{\text{max}}^{(em)} / \lambda_{\text{min}|}^{(em)} a_{\text{min}|}^{(em)}$ (for aggregate claims models), where $\lambda_{\text{max}}^{(em)}$ is the empirical frequency of the most risky group; $a_{\text{max}}^{(em)}$ and $\lambda_{\text{max}}^{(em)} a_{\text{max}}^{(em)}$ are defined analogously.]

Although lift can help us better understand the economic value of the model, it is a relative measure, similar to NLL. Therefore, it may not provide a clear indication of the actual economic impact or value of a model without additional context. Besides, the effectiveness of lift may vary across different data sets and contexts. Since it may not generalize well to diverse scenarios, it is crucial to consider its application within the specific context of the problem at hand.

Among the five evaluation metrics discussed above, RSS, NLL and lift consistently tend to improve with more splits in tree models, while SE and DS can assist in identifying the optimal model. This aligns with the DIC selection, which will be verified in the simulation examples and real data analyses below. We remark that more performance measures and diagnostic approaches (such as deviance and Gini index) can be introduced following ideas in, e.g., Henckaerts *et al.* (2021), Wuthrich & Buser (2022) and Lee (2020, 2021). However, this is not the main focus of the present thesis, so these are explored elsewhere.

2.4 Summary of Chapter 2

We began this chapter by reviewing the standard decision tree, understanding its basic structure, and providing an example of applying the Poisson distribution in CART. Recognizing the limitations of CART, we delved into Bayesian CART. Building upon the fundamental framework in Chipman *et al.* (1998), we conducted a comprehensive analysis of prior specifications. We proposed a general MCMC algorithm for BCART models applied to data with a general distribution, involving data augmentation techniques. Subsequently, we advocated for using different

DICs to select an optimal tree among all visited trees generated by MCMC algorithms. Following that, a three-step approach was proposed, and its effectiveness will be thoroughly validated in the subsequent simulation examples and real data analyses. Finally, five evaluation metrics were introduced and their advantages and disadvantages were discussed respectively for model comparisons.

Chapter 3

Frequency Modelling with Bayesian CART

In this chapter, we introduce BCART models for insurance claims frequency models by specifying the response distribution in the general framework introduced in Section 2.2. We shall discuss four commonly used distributions in the literature to model the number of claims, namely, Poisson, NB, ZIP and ZINB distributions. Particularly, in the NB, ZIP and ZINB distributions, we not only explore different ways to embed exposure but also utilize the data augmentation technique; see, e.g., Wüthrich & Merz (2023), Murray (2021) and Lee (2020, 2021). Specific formulas for some of the evaluation metrics for each model are provided in their respective sections. Subsequently, three simulation examples are designed to address specific problems, and some conclusions are drawn.

We first review claims frequency data and the purpose of claims frequency analysis. Consider a claims data set with n policyholders $\mathcal{D} = (\boldsymbol{X}, \boldsymbol{v}, \boldsymbol{N}) = \big((\boldsymbol{x}_1, v_1, N_1), \ldots, (\boldsymbol{x}_n, v_n, N_n)\big)^\top$ $(i = 1, 2, \ldots, n)$, where N_i is the number of claims reported, and $v_i \in (0, 1]$ is the exposure. The objective of claims frequency analysis is to explain and predict the claims information N_i based on the explanatory variables $\boldsymbol{x}_i$ and the exposure v_i for each individual policyholder i, leading to the *claims frequency*, i.e., the number of claims filed per unit year of exposure to risk. Subsequently, each section demonstrates how to apply different distributions in BCART models in detail for various purposes.

3.1 Poisson-Bayesian CART

Consider a tree $\mathcal{T}$ with b terminal nodes as discussed in Section 2.2. In a Poisson model, we assume all insurance policyholders $i = 1, 2, \ldots, n$ have independent

claim counts N_i with

$$N_i \mid \boldsymbol{x}, v \;\sim\; \text{Poi}(\lambda(\boldsymbol{x})v)$$

for the i-th observation where $\lambda(\boldsymbol{x}) = \sum_{t=1}^{b} \lambda_t I_{(\boldsymbol{x}\in\mathcal{A}_t)}$ $(\lambda_t > 0)$, and $\{\mathcal{A}_t\}$ is a partition of $\mathcal{X}$. Here we use the standard notation λ_t for claims frequency rather than the generic notation $\boldsymbol{\theta}_t$ for the parameter in terminal node t. The aim is to estimate the regression function $\lambda(\cdot)$, describing the expected claims frequency. Essentially, we have specified the distribution for terminal node t as

$$f_{\text{P}}\left(N_{tj} \mid \lambda_t\,, v_{tj}\right) = \text{P}\left(N_{tj} \mid \lambda_t, v_{tj}\right) = \frac{e^{-\lambda_t v_{tj}}\left(\lambda_t v_{tj}\right)^{N_{tj}}}{N_{tj}!}, \quad N_{tj} = 0, 1, 2, \ldots, \tag{3.1}$$

for the i-th observation such that $\boldsymbol{x}_i \in \mathcal{A}_t$. The mean and variance of N_{tj} are given by

$$\mathbb{E}\left(N_{tj}\right) = \text{Var}\left(N_{tj}\right) = \lambda_t v_{tj}.$$

Note that, for simplicity, here and hereafter, the exposure v_i and explanatory variables $\boldsymbol{x}_i$ will be omitted in some notation. Based on the discussions in Subsection 2.2.1, we choose a common conjugate Gamma prior for λ_t $(t = 1, 2, \ldots, b)$ with hyper-parameters $\alpha, \beta > 0$, that is,

$$p\left(\lambda_t\right) = \frac{\beta^{\alpha}{\lambda_t}^{\alpha-1}e^{-\beta\lambda_t}}{\Gamma(\alpha)}, \tag{3.2}$$

with $\Gamma(\cdot)$ denoting the Gamma function. As in Section 2.2, for terminal node t we denote the associated data as $\left(\boldsymbol{X}_t, \boldsymbol{v}_t, \boldsymbol{N}_t\right) = \left((X_{t1}, v_{t1}, N_{t1}), \ldots, (X_{tn_t}, v_{tn_t}, N_{tn_t})\right)^{\top}$. With the above Gamma prior, the integrated likelihood for terminal node t can be obtained as

$$\begin{aligned}
p_{\text{P}}\left(\boldsymbol{N}_t \mid \boldsymbol{X}_t, \boldsymbol{v}_t\right) &= \int_0^{\infty} f_{\text{P}}\left(\boldsymbol{N}_t \mid \lambda_t\right) p(\lambda_t) d\lambda_t \\
&= \int_0^{\infty} \prod_{j=1}^{n_t} \frac{e^{-\lambda_t v_{tj}}\left(\lambda_t v_{tj}\right)^{N_{tj}}}{N_{tj}!} \frac{\beta^{\alpha}{\lambda_t}^{\alpha-1}e^{-\beta\lambda_t}}{\Gamma(\alpha)} d\lambda_t \\
&= \frac{\beta^{\alpha}\prod_{j=1}^{n_t} v_{tj}^{N_{tj}}}{\Gamma(\alpha)\prod_{j=1}^{n_t} N_{tj}!} \int_0^{\infty} \lambda_t^{\sum_{j=1}^{n_t} N_{tj}+\alpha-1} e^{-(\sum_{j=1}^{n_t} v_{tj}+\beta)\lambda_t} d\lambda_t \\
&= \frac{\beta^{\alpha}\prod_{j=1}^{n_t} v_{tj}^{N_{tj}}}{\Gamma(\alpha)\prod_{j=1}^{n_t} N_{tj}!} \frac{\Gamma(\sum_{j=1}^{n_t} N_{tj}+\alpha)}{(\sum_{j=1}^{n_t} v_{tj}+\beta)^{\sum_{j=1}^{n_t} N_{tj}+\alpha}}.
\end{aligned} \tag{3.3}$$

Clearly, from (3.3), we see that the posterior distribution of λ_t, conditional on $\boldsymbol{N}_t$, is given by

$$\lambda_t \mid \boldsymbol{N}_t \;\sim\; \text{Gamma}\left(\sum_{j=1}^{n_t} N_{tj} + \alpha, \sum_{j=1}^{n_t} v_{tj} + \beta\right). \tag{3.4}$$

The integrated likelihood for the tree $\mathcal{T}$ is thus given by

$$p_{\mathrm{P}}\left(\boldsymbol{N} \mid \boldsymbol{X}, \boldsymbol{v}, \mathcal{T}\right) = \prod_{t=1}^{b} p_{\mathrm{P}}\left(\boldsymbol{N}_t \mid \boldsymbol{X}_t, \boldsymbol{v}_t\right). \tag{3.5}$$

Next, we discuss the DIC for this tree, focusing on the DIC_t for terminal node t. First, we have

$$D\left(\lambda_t\right) = -2\sum_{j=1}^{n_t} \log f_{\mathrm{P}}(N_{tj} \mid \lambda_t) = -2\sum_{j=1}^{n_t}\Big(-\lambda_t v_{tj} + N_{tj}\log\left(\lambda_t v_{tj}\right) - \log\left(N_{tj}!\right)\Big), \tag{3.6}$$

and by (3.4) we get the posterior mean for λ_t as

$$\bar{\lambda}_t = \mathbb{E}_{\mathrm{post}}(\lambda_t) = \frac{\sum_{j=1}^{n_t} N_{tj} + \alpha}{\sum_{i=1}^{n_t} v_{tj} + \beta}. \tag{3.7}$$

Furthermore, we derive that

$$\begin{aligned}
&\overline{D\left(\lambda_t\right)} \\
&= \mathbb{E}_{\mathrm{post}}\left(D\left(\lambda_t\right)\right) \\
&= 2\sum_{j=1}^{n_t}\left(v_{tj}\mathbb{E}_{\mathrm{post}}\left(\lambda_t\right) - N_{tj}\mathbb{E}_{\mathrm{post}}\left(\log\left(\lambda_t\right) + \log\left(v_{tj}\right)\right) + \log\left(N_{tj}!\right)\right) \\
&= -2\left(\psi\left(\sum_{j=1}^{n_t} N_{tj} + \alpha\right) - \log\left(\sum_{j=1}^{n_t} v_{tj} + \beta\right)\right)\sum_{j=1}^{n_t} N_{tj} \\
&\quad +2\left(\frac{\sum_{j=1}^{n_t} N_{tj} + \alpha}{\sum_{j=1}^{n_t} v_{tj} + \beta}\right)\sum_{j=1}^{n_t} v_{tj} - 2\sum_{j=1}^{n_t} N_{tj}\log\left(v_{tj}\right) + 2\sum_{j=1}^{n_t}\log\left(N_{tj}!\right),
\end{aligned} \tag{3.8}$$

where we use the fact that

$$\mathbb{E}_{\mathrm{post}}\left(\log\left(\lambda_t\right)\right) = \psi\left(\sum_{j=1}^{n_t} N_{tj} + \alpha\right) - \log\left(\sum_{j=1}^{n_t} v_{tj} + \beta\right),$$

with $\psi(x) = \Gamma'(x)/\Gamma(x)$ being the digamma function. Using (3.6)–(3.8), we obtain the effective number of parameters for terminal node t as

$$\begin{aligned}
p_{Dt} &= \overline{D(\lambda_t)} - D(\bar{\lambda}_t) \\
&= 2\left(\log\left(\sum_{j=1}^{n_t} N_{tj} + \alpha\right) - \psi\left(\sum_{j=1}^{n_t} N_{tj} + \alpha\right)\right)\sum_{j=1}^{n_t} N_{tj},
\end{aligned}$$

and

$$\begin{aligned}
&\mathrm{DIC}_t \\
&= D\left(\bar{\lambda}_t\right) + 2p_{Dt} \\
&= 2\left(\frac{\sum_{j=1}^{n_t} N_{tj} + \alpha}{\sum_{j=1}^{n_t} v_{tj} + \beta}\right)\sum_{j=1}^{n_t} v_{tj} - 2\sum_{j=1}^{n_t} N_{tj}\left(\log\left(\frac{\sum_{j=1}^{n_t} N_{tj} + \alpha}{\sum_{j=1}^{n_t} v_{tj} + \beta}\right) + \log\left(v_{tj}\right)\right) \\
&\quad +4\left(\log\left(\sum_{j=1}^{n_t} N_{tj} + \alpha\right) - \psi\left(\sum_{j=1}^{n_t} N_{tj} + \alpha\right)\right)\sum_{j=1}^{n_t} N_{tj} + 2\sum_{j=1}^{n_t}\log\left(N_{tj}!\right).
\end{aligned}$$

Then the DIC of the tree $\mathcal{T}$ is obtained by using (2.14).

Remark 3.1 *(a) Since $\psi(x) = \log(x) - \frac{1}{2x}(1 + o(x)), as\ x \to \infty$, we immediately see that $p_{Dt} \to 1$ as $n_t \to \infty$. This explains the name of the effective number of parameters in the Bayesian framework, as 1 is the number of parameters in the terminal node t for the Poisson model if a flat prior is assumed for λ_t. Additionally, in the following models (NB, ZIP, ZINB, Gamma in Sections 4.2 and 5.1, and CPG/ZICPG in Section 5.3), if the Gamma prior is chosen for the parameter, we shall obtain this similar form for the corresponding effective number of parameters, and we can use this property to explicitly obtain the values for effective number of parameters. Therefore, we will not repeat this later.*

With the above (3.4)–(3.5) and DIC obtained, we can use the three-step approach proposed in Subsection 2.2.4 to search for an optimal tree, where (3.4) and (3.5) should be used in step 4 and step 2, respectively, in Algorithm 2.2. Given an optimal tree, the estimated claims frequency $\bar{\lambda}_t$ in terminal node t can be given by the posterior mean in (3.7), using (2.18). It is worth noting that we can obtain the same estimate by using (2.19) instead.

Remark 3.2 *For a Bayesian CART, the frequency parameter λ_t can be estimated in each terminal node by using the posterior distribution of $\lambda_t \mid \boldsymbol{N}_t$ obtained in (3.4). Several different estimators can be used, such as posterior mean, posterior mode, or drawing a random value from the posterior distribution. They all have their own advantages and disadvantages. From a computational viewpoint, it is generally easier to obtain the posterior mean than the posterior mode. This is due to the fact that calculating the posterior mean involves calculating an integral, which can often be done analytically or using numerical methods like MCMC sampling. On the other hand, calculating the posterior mode typically involves optimising a function, which can be more computationally intensive. Consequently,*

Table 3.1: Evaluation metrics for Poisson-BCART. ϵ_t denotes the empirical claims frequency in node t, computed as $\sum_{j=1}^{n_t} N_{tj} / \sum_{j=1}^{n_t} v_{tj}$, and $\bar{\lambda}_t$ denotes the estimated claims frequency for node t in the Bayesian framework, obtained from (3.7).

	Formulas
RSS($\boldsymbol{N}$)	$\sum_{t=1}^{b} \sum_{j=1}^{n_t} (N_{tj} - \bar{\lambda}_t v_{tj})^2$
SE	$\sum_{t=1}^{b} (\epsilon_t - \bar{\lambda}_t)^2$
DS	$\sum_{t=1}^{b} \left((\epsilon_t - \bar{\lambda}_t)^2 / \bar{\lambda}_t \right)$

the posterior mean is more commonly favoured in practical applications. However, some simulation examples studied by Celeux et al. (2006) show that the posterior mean can be a poor estimator within missing data models for various reasons. For example, in the context of non-normal or skewed distributions commonly encountered in missing data scenarios, the posterior mean may not accurately capture the central tendency, making it less robust in comparison to the posterior mode; see more discussion in Celeux et al. (2006). In addition, the random drawing of a value from the posterior distribution is easy, but the randomness of this method is too high, and it is susceptible to extreme values. Further research into the choice of these three kinds of values could therefore be explored. We use posterior mean in our implementation for all BCART models in the thesis because of computational efficiency.

At the end of this section, we provide the specific formulas for some of the evaluation metrics (see Section 2.3) based on Poisson distribution in Table 3.1.

3.2 Negative Binomial-Bayesian CART

The NB distribution, a member of the mixed Poisson family, offers an effective way to handle over-dispersed insurance claims frequency data where excessive zeros are common. Consider a tree $\mathcal{T}$ with b terminal nodes as before. In the NB model, we assume that $N_{tj} \mid X_{tj}, v_{tj}$ follows a NB distribution for all terminal nodes t, $(t = 1, \ldots, b)$. There are different ways to parameterize the NB distribution, particularly with the exposure (one option is to embed the exposure in one parameter, while the other is to embed the exposure in both parameters); see,

e.g., Lee (2020) and Wüthrich & Merz (2023). We shall discuss two models in this section.

3.2.1 Negative Binomial Model 1 (NB1)

We first adopt the most common parameterization of the NB distribution; see, e.g., Murray (2021). That is, for terminal node t,

$$\begin{aligned} &f_{\text{NB1}}(N_{tj} \mid \kappa_t, \lambda_t, v_{tj}) \\ &= \mathrm{P}(N_{tj} \mid \kappa_t, \lambda_t, v_{tj}) \\ &= \frac{\Gamma(N_{tj}+\kappa_t)}{\Gamma(\kappa_t)N_{tj}!}\left(\frac{\kappa_t}{\kappa_t+\lambda_t v_{tj}}\right)^{\kappa_t}\left(\frac{\lambda_t v_{tj}}{\kappa_t+\lambda_t v_{tj}}\right)^{N_{tj}}, \quad N_{tj}=0,1,\ldots, \end{aligned} \tag{3.9}$$

where $\kappa_t, \lambda_t > 0$. It is easy to show that the mean and variance of N_{tj} are given by

$$\mathbb{E}(N_{tj} \mid \kappa_t, \lambda_t) = \lambda_t v_{tj}, \qquad \mathrm{Var}(N_{tj} \mid \kappa_t, \lambda_t) = \lambda_t v_{tj}\left(1+\frac{\lambda_t v_{tj}}{\kappa_t}\right). \tag{3.10}$$

The degree of over-dispersion in relation to the Poisson is controlled by the additional parameter κ_t in the NB model, which converges to the Poisson model as $\kappa_t \to \infty$.

In NB regression, the lack of simple and efficient algorithms for posterior computation has seriously limited routine applications of Bayesian approaches. Recent studies make Bayesian approaches appealing by introducing data augmentation techniques; see, e.g., Zhou *et al.* (2012) and Murray (2021). In order to save on the total computational time of the algorithm and avoid the difficulty of finding an appropriate prior for κ_t with corresponding data augmentation, we shall treat the parameter κ_t as known in the Bayesian framework which can be estimated upfront by using, e.g., the Method of Moments Estimation (MME) method. However, in line with the Poisson model, we shall treat λ_t as unknown and use a conjugate Gamma prior with corresponding data augmentation. Based on the formulas given in (3.10), we can estimate the parameter κ_t using MME; see, e.g., Chapter 2 of Wuthrich (2022) as follows

$$\hat{\kappa}_t = \frac{\hat{\lambda}_t^2}{\hat{V}_t^2 - \hat{\lambda}_t}\frac{1}{n_t-1}\left(\sum_{j=1}^{n_t} v_{tj} - \frac{\sum_{j=1}^{n_t} v_{tj}^2}{\sum_{j=1}^{n_t} v_{tj}}\right), \tag{3.11}$$

where

$$\hat{V}_t^2 = \frac{1}{n_t-1}\sum_{j=1}^{n_t} v_{tj}\left(\frac{N_{tj}}{v_{tj}} - \hat{\lambda}_t\right)^2, \quad \hat{\lambda}_t = \frac{\sum_{j=1}^{n_t} N_{tj}}{\sum_{j=1}^{n_t} v_{tj}}. \tag{3.12}$$

Next, introducing a latent variable $\boldsymbol{\xi}_t = (\xi_{t1}, \xi_{t2}, \ldots, \xi_{tn_t}) \in (0,\infty)^{n_t}$, we can define a data augmented likelihood for the j-th data instance in terminal node t as

$$f_{\text{NB1}}\left(N_{tj}, \xi_{tj} \mid \hat{\kappa}_t, \lambda_t\right) = \frac{\left(\lambda_t v_{tj}\right)^{N_{tj}} e^{-\xi_{tj}(\lambda_t v_{tj}+\hat{\kappa}_t)} \hat{\kappa}_t^{\hat{\kappa}_t} \xi_{tj}^{\hat{\kappa}_t+N_{tj}-1}}{\Gamma(\hat{\kappa}_t) N_{tj}!}. \tag{3.13}$$

It is easily checked that integrating over $\xi_{tj} \in (0,\infty)$ in (3.13) yields the marginal distribution (3.9), since

$$\begin{aligned}
&\int_0^\infty f_{\text{NB1}}\left(N_{tj}, \xi_{tj} \mid \hat{\kappa}_t, \lambda_t\right) d\xi_{tj} \\
&= \frac{\left(\lambda_t v_{tj}\right)^{N_{tj}} \hat{\kappa}_t^{\hat{\kappa}_t}}{\Gamma(\hat{\kappa}_t) N_{tj}!} \int_0^\infty e^{-\xi_{tj}(\lambda_t v_{tj}+\hat{\kappa}_t)} \xi_{tj}^{\hat{\kappa}_t+N_{tj}-1} d\xi_{tj} \\
&= \frac{\left(\lambda_t v_{tj}\right)^{N_{tj}} \hat{\kappa}_t^{\hat{\kappa}_t}}{\Gamma(\hat{\kappa}_t) N_{tj}!} \frac{\Gamma(\hat{\kappa}_t + N_{tj})}{(\lambda_t v_{tj} + \hat{\kappa}_t)^{\hat{\kappa}_t+N_{tj}}}.
\end{aligned}$$

Further, we see that ξ_{tj}, given data N_{tj} and parameters ($\hat{\kappa}_t$ and λ_t), has a Gamma distribution, i.e.,

$$\xi_{tj} \mid N_{tj}, \hat{\kappa}_t, \lambda_t \;\sim\; \text{Gamma}\left(\hat{\kappa}_t + N_{tj}, \hat{\kappa}_t + \lambda_t v_{tj}\right). \tag{3.14}$$

Given the data augmented likelihood in (3.13), the estimated parameter $\hat{\kappa}_t$ using (3.11), and a conjugate Gamma prior for λ_t with hyper-parameters $\alpha, \beta > 0$ (cf. (3.2)), we can derive the integrated augmented likelihood for the terminal node t as follows

$$\begin{aligned}
&p_{\text{NB1}}\left(\boldsymbol{N}_t, \boldsymbol{\xi}_t \mid \boldsymbol{X}_t, \boldsymbol{v}_t, \hat{\kappa}_t\right) \\
&= \int_0^\infty f_{\text{NB1}}\left(\boldsymbol{N}_t, \boldsymbol{\xi}_t \mid \hat{\kappa}_t, \lambda_t\right) p\left(\lambda_t\right) d\lambda_t \\
&= \int_0^\infty \prod_{j=1}^{n_t} \left(\frac{\left(\lambda_t v_{tj}\right)^{N_{tj}} e^{-\xi_{tj}\left(\lambda_t v_{tj}+\hat{\kappa}_t\right)} \hat{\kappa}_t^{\hat{\kappa}_t} \xi_{tj}^{\hat{\kappa}_t+N_{tj}-1}}{\Gamma\left(\hat{\kappa}_t\right) N_{tj}!} \right) \frac{\beta^\alpha \lambda_t^{\alpha-1} e^{-\beta\lambda_t}}{\Gamma(\alpha)} d\lambda_t \\
&= \frac{\beta^\alpha \hat{\kappa}_t^{n_t\hat{\kappa}_t}}{\Gamma(\alpha)\Gamma\left(\hat{\kappa}_t\right)^{n_t}} \prod_{j=1}^{n_t} \left(\frac{v_{tj}^{N_{tj}}}{N_{tj}!} \xi_{tj}^{\hat{\kappa}_t+N_{tj}-1} e^{-\xi_{tj}\hat{\kappa}_t} \right) \int_0^\infty \lambda_t^{\sum_{j=1}^{n_t} N_{tj}+\alpha-1} e^{-\left(\sum_{j=1}^{n_t} \xi_{tj} v_{tj}+\beta\right)\lambda_t} d\lambda_t \\
&= \frac{\beta^\alpha \hat{\kappa}_t^{n_t\hat{\kappa}_t}}{\Gamma(\alpha)\Gamma\left(\hat{\kappa}_t\right)^{n_t}} \prod_{j=1}^{n_t} \left(\frac{v_{tj}^{N_{tj}}}{N_{tj}!} \xi_{tj}^{\hat{\kappa}_t+N_{tj}-1} e^{-\xi_{tj}\hat{\kappa}_t} \right) \frac{\Gamma\left(\sum_{j=1}^{n_t} N_{tj} + \alpha\right)}{\left(\sum_{j=1}^{n_t} \xi_{tj} v_{tj} + \beta\right)^{\sum_{j=1}^{n_t} N_{tj}+\alpha}}.
\end{aligned} \tag{3.15}$$

Moreover, from the above, we see that the posterior distribution of λ_t given the augmented data $(\boldsymbol{N}_t, \boldsymbol{\xi}_t)$, is given by

$$\lambda_t \mid \boldsymbol{N}_t, \boldsymbol{\xi}_t \;\sim\; \text{Gamma}\left(\sum_{j=1}^{n_t} N_{tj} + \alpha, \sum_{j=1}^{n_t} \xi_{tj} v_{tj} + \beta\right).$$

The integrated augmented likelihood for the tree $\mathcal{T}$ is thus given by

$$p_{\mathrm{NB1}}\left(\boldsymbol{N}, \boldsymbol{\xi} \mid \boldsymbol{X}, \boldsymbol{v}, \hat{\boldsymbol{\kappa}}, \mathcal{T}\right) = \prod_{t=1}^{b} p_{\mathrm{NB1}}\left(\boldsymbol{N}_t, \boldsymbol{\xi}_t \mid \boldsymbol{X}_t, \boldsymbol{v}_t, \hat{\kappa}_t\right). \tag{3.16}$$

Now, we discuss the DIC for this tree. Since we only consider uncertainty for $\boldsymbol{\lambda}$ but not for $\boldsymbol{\kappa}$, the DIC defined in (2.17) cannot be adopted directly. Thus, using the idea that DIC="goodness of fit"+"complexity", we can introduce a new DIC_t for terminal node t as follows

$$\mathrm{DIC}_t = D(\bar{\lambda}_t) + 2r_{Dt}.$$

Here, the goodness of fit is given by

$$D(\bar{\lambda}_t) = -2\sum_{j=1}^{n_t} \log f_{\mathrm{NB1}}(N_{tj} \mid \hat{\kappa}_t, \bar{\lambda}_t),$$

and the effective number of parameters r_{Dt} is given by

$$\begin{aligned} r_{Dt} &= \overline{D(\boldsymbol{\theta}_t)} - D(\bar{\boldsymbol{\theta}}_t) \\ &= -2\mathbb{E}_{\mathrm{post}}(\log(f(\boldsymbol{y}_t, \boldsymbol{z}_t \mid \boldsymbol{\theta}_t))) + 2\log(f(\boldsymbol{y}_t, \boldsymbol{z}_t \mid \bar{\boldsymbol{\theta}}_t)) \\ &= 1 + 2\sum_{j=1}^{n_t}\left\{\log\left(f_{\mathrm{NB1}}(N_{tj}, \xi_{tj} \mid \hat{\kappa}_t, \bar{\lambda}_t)\right) - \mathbb{E}_{\mathrm{post}}\left[\log\left(f_{\mathrm{NB1}}(N_{tj}, \xi_{tj} \mid \hat{\kappa}_t, \lambda_t)\right)\right]\right\}, \end{aligned} \tag{3.17}$$

where κ_t is treated as known while still maintaining its status as a model parameter, and we denote its effective number as 1; the second part of the last line is for λ_t,

$$\bar{\lambda}_t = \frac{\sum_{j=1}^{n_t} N_{tj} + \alpha}{\sum_{j=1}^{n_t} \xi_{tj} v_{tj} + \beta}, \tag{3.18}$$

and

$$\begin{aligned} &\mathbb{E}_{\mathrm{post}}\left(\log(f_{\mathrm{NB1}}(N_{tj}, \xi_{tj} \mid \hat{\kappa}_t, \lambda_t))\right) \\ &= 2\sum_{j=1}^{n_t} v_{tj}\mathbb{E}_{\mathrm{post}}\left(\lambda_t\right) - 2\sum_{j=1}^{n_t} N_{tj}\mathbb{E}_{\mathrm{post}}\left(\log\left(\lambda_t\right) + \log\left(v_{tj}\right)\right) + 2\sum_{j=1}^{n_t}\log\left(N_{tj}!\right) \\ &= -2\sum_{j=1}^{n_t} N_{tj}\left(\log(v_{tj}) + \psi\left(\sum_{j=1}^{n_t} N_{tj} + \alpha\right) - \log\left(\sum_{j=1}^{n_t}\xi_{tj}v_{tj} + \beta\right)\right) \\ &\quad + 2\left(\frac{\sum_{j=1}^{n_t} N_{tj} + \alpha}{\sum_{j=1}^{n_t}\xi_{tj}v_{tj} + \beta}\right)\sum_{j=1}^{n_t}\xi_{tj}v_{tj} + 2\left(\log(\Gamma(\hat{\kappa}_t)) - \hat{\kappa}_t\log(\hat{\kappa}_t)\right) \\ &\quad + 2\sum_{j=1}^{n_t}\left(\log\left(N_{tj}!\right) - \left(\hat{\kappa}_t + N_{tj} - 1\right)\log\left(\xi_{tj}\right) + \xi_{tj}\hat{\kappa}_t\right). \end{aligned}$$

Therefore, a direct calculation shows that the effective number of parameters for terminal node t is given by

$$r_{Dt} = 1 + 2\left(\log\left(\sum_{j=1}^{n_t} N_{tj} + \alpha\right) - \psi\left(\sum_{j=1}^{n_t} N_{tj} + \alpha\right)\right)\sum_{j=1}^{n_t} N_{tj}, \tag{3.19}$$

and thus

$$\begin{aligned}
&\mathrm{DIC}_t \\
&= D\left(\bar{\lambda}_t\right) + 2r_{Dt} \\
&= -2\sum_{j=1}^{n_t} N_{tj}\left(\log(v_{tj}) + \log\left(\sum_{j=1}^{n_t} N_{tj} + \alpha\right) - \log\left(\sum_{j=1}^{n_t} \xi_{tj} v_{tj} + \beta\right)\right) \\
&\quad + 2\left(\frac{\sum_{j=1}^{n_t} N_{tj} + \alpha}{\sum_{j=1}^{n_t} \xi_{tj} v_{tj} + \beta}\right)\sum_{j=1}^{n_t} \xi_{tj} v_{tj} + 2\left(\log(\Gamma(\hat{\kappa}_t)) - \hat{\kappa}_t \log(\hat{\kappa}_t)\right) \\
&\quad + 2\sum_{j=1}^{n_t}\left(\log\left(N_{tj}!\right) - \left(\hat{\kappa}_t + N_{tj} - 1\right)\log\left(\xi_{tj}\right) + \xi_{tj}\hat{\kappa}_t\right) \\
&\quad + 2 + 4\left(\log\left(\sum_{j=1}^{n_t} N_{tj} + \alpha\right) - \psi\left(\sum_{j=1}^{n_t} N_{tj} + \alpha\right)\right)\sum_{j=1}^{n_t} N_{tj}.
\end{aligned}$$

3.2.2 Negative Binomial Model 2 (NB2)

We now consider another parameterization of the NB distribution; see, e.g., Lee (2020) and Wüthrich & Merz (2023). Although the parameterization is different, we employ similar techniques to handle the parameters and perform calculations. We shall avoid repeated descriptions and emphasize the differences. For easier reading and completeness, the calculation process and results are still included in the main text. For terminal node t,

$$f_{\mathrm{NB2}}(N_{tj} \mid \kappa_t, \lambda_t, v_{tj}) = \frac{\Gamma(N_{tj} + \kappa_t v_{tj})}{\Gamma(\kappa_t v_{tj}) N_{tj}!}\left(\frac{\kappa_t}{\kappa_t + \lambda_t}\right)^{\kappa_t v_{tj}}\left(\frac{\lambda_t}{\kappa_t + \lambda_t}\right)^{N_{tj}}, \quad N_{tj} = 0, 1, \ldots, \tag{3.20}$$

where $\kappa_t, \lambda_t > 0$. It is easy to show that the mean of N_{tj} is the same as in (3.10), but the variance becomes

$$\mathrm{Var}(N_{tj} \mid \kappa_t, \lambda_t) = \lambda_t v_{tj}\left(1 + \frac{\lambda_t}{\kappa_t}\right). \tag{3.21}$$

This formulation yields a fixed over-dispersion of size λ_t/κ_t which does not depend on the exposure v_{ti}, and thus it is sometimes preferred (see Wüthrich & Merz

(2023)) and has been judged as more effective for real insurance data analyses (see Lee (2020)).

We use the same way to deal with κ_t and λ_t as in the previous subsection. Using the same approach as Chapter 2 of Wuthrich (2022), we can estimate the parameter κ_t as follows

$$\hat{\kappa}_t = \frac{\hat{\lambda}_t^2}{\hat{V}_t^2 - \hat{\lambda}_t}, \tag{3.22}$$

where $\hat{V}_t^2$ and $\hat{\lambda}_t$ are given in (3.12). Note that this parameterization offers a simpler estimation for $\hat{\kappa}_t$, and that $\hat{\lambda}_t$ is a minimal variance estimator; see, e.g., Wuthrich (2022).

As before, by introducing a latent variable $\boldsymbol{\xi}_t = (\xi_{t1}, \xi_{t2}, \ldots, \xi_{tn_t}) \in (0,\infty)^{n_t}$, we can define a data augmented likelihood for the j-th data instance in terminal node t as

$$f_{\text{NB2}}\left(N_{tj}, \xi_{tj} \mid \hat{\kappa}_t, \lambda_t\right) = \frac{\left(\lambda_t v_{tj}\right)^{N_{tj}} e^{-\xi_{tj}(\lambda_t v_{tj} + \hat{\kappa}_t v_{tj})} \left(\hat{\kappa}_t v_{tj}\right)^{\hat{\kappa}_t v_{tj}} \xi_{tj}^{\hat{\kappa}_t v_{tj} + N_{tj} - 1}}{\Gamma(\hat{\kappa}_t v_{tj}) N_{tj}!}. \tag{3.23}$$

Similar to the previous subsection, it can be simply verified that integrating over $\xi_{tj} \in (0,\infty)$ in (3.23) yields the marginal distribution (3.20). Further, we see that ξ_{tj}, given data N_{tj} and parameters ($\hat{\kappa}_t$ and λ_t), has a Gamma distribution, i.e.,

$$\xi_{tj} \mid N_{tj}, \hat{\kappa}_t, \lambda_t \ \sim \ \text{Gamma}\left(\hat{\kappa}_t v_{tj} + N_{tj}, \hat{\kappa}_t v_{tj} + \lambda_t v_{tj}\right).$$

Given the data augmented likelihood in (3.23), the estimated parameter $\hat{\kappa}_t$ using (3.22), and a conjugate Gamma prior for λ_t with hyper-parameters $\alpha, \beta > 0$, we can derive the integrated augmented likelihood for terminal node t as follows

$$\begin{aligned}
&p_{\text{NB2}}\left(\boldsymbol{N}_t, \boldsymbol{\xi}_t \mid \boldsymbol{X}_t, \boldsymbol{v}_t, \hat{\kappa}_t\right) \\
&= \int_0^\infty f_{\text{NB2}}\left(\boldsymbol{N}_t, \boldsymbol{\xi}_t \mid \hat{\kappa}_t, \lambda_t\right) p(\lambda_t) d\lambda_t \\
&= \int_0^\infty \prod_{j=1}^{n_t} \left(\frac{\left(\lambda_t v_{tj}\right)^{N_{tj}} e^{-\xi_{tj}(\lambda_t v_{tj} + \hat{\kappa}_t v_{tj})} \left(\hat{\kappa}_t v_{tj}\right)^{\hat{\kappa}_t v_{tj}} \xi_{tj}^{\hat{\kappa}_t v_{tj} + N_{tj} - 1}}{\Gamma(\hat{\kappa}_t v_{tj}) N_{tj}!} \right) \frac{\beta^\alpha {\lambda_t}^{\alpha-1} e^{-\beta\lambda_t}}{\Gamma(\alpha)} d\lambda_t \\
&= \frac{\beta^\alpha}{\Gamma(\alpha)} \prod_{j=1}^{n_t} \left(\frac{v_{tj}^{N_{tj}} \left(\hat{\kappa}_t v_{tj}\right)^{\hat{\kappa}_t v_{tj}}}{\Gamma(\hat{\kappa}_t v_{tj}) N_{tj}!} \xi_{tj}^{\hat{\kappa}_t v_{tj} + N_{tj} - 1} e^{-\xi_{tj} \hat{\kappa}_t v_{tj}} \right) \\
&\quad \times \int_0^\infty \lambda_t^{\sum_{j=1}^{n_t} N_{tj} + \alpha - 1} e^{-\left(\sum_{j=1}^{n_t} \xi_{tj} v_{tj} + \beta\right)\lambda_t} d\lambda_t \\
&= \frac{\beta^\alpha}{\Gamma(\alpha)} \prod_{j=1}^{n_t} \left(\frac{v_{tj}^{N_{tj}} \left(\hat{\kappa}_t v_{tj}\right)^{\hat{\kappa}_t v_{tj}}}{\Gamma(\hat{\kappa}_t v_{tj}) N_{tj}!} \xi_{tj}^{\hat{\kappa}_t v_{tj} + N_{tj} - 1} e^{-\xi_{tj} \hat{\kappa}_t v_{tj}} \right) \frac{\Gamma(\sum_{j=1}^{n_t} N_{tj} + \alpha)}{\left(\sum_{j=1}^{n_t} \xi_{tj} v_{tj} + \beta\right)^{\sum_{j=1}^{n_t} N_{tj} + \alpha}}.
\end{aligned} \tag{3.24}$$

From the above we see that the posterior distribution of λ_t, given the augmented data $(\boldsymbol{N}_t, \boldsymbol{\xi}_t)$, is given by

$$\lambda_t \mid \boldsymbol{N}_t, \boldsymbol{\xi}_t \ \sim \ \text{Gamma}\left(\sum_{j=1}^{n_t} N_{tj} + \alpha, \sum_{j=1}^{n_t} \xi_{tj} v_{tj} + \beta\right).$$

Although the parameterization way is different, the posterior distribution of λ_t is the same for NB1 and NB2 models. The integrated augmented likelihood for the tree $\mathcal{T}$ is thus given by

$$p_{\text{NB2}}\left(\boldsymbol{N}, \boldsymbol{\xi} \mid \boldsymbol{X}, \boldsymbol{v}, \hat{\boldsymbol{\kappa}}, \mathcal{T}\right) = \prod_{t=1}^{b} p_{\text{NB2}}\left(\boldsymbol{N}_t, \boldsymbol{\xi}_t \mid \boldsymbol{X}_t, \boldsymbol{v}_t, \hat{\kappa}_t\right). \tag{3.25}$$

Now, we discuss the DIC_t for terminal node t of this tree. Similarly, as in the previous subsection, we can derive the same expression for r_{Dt} as in (3.19) and we can easily check that

$$\begin{aligned}
&\text{DIC}_t \\
&= D(\bar{\lambda}_t) + 2r_{Dt} \\
&= -2\sum_{j=1}^{n_t} N_{tj}\left(\log(v_{tj}) + \log\left(\sum_{j=1}^{n_t} N_{tj} + \alpha\right) - \log\left(\sum_{j=1}^{n_t} \xi_{tj} v_{tj} + \beta\right)\right) \\
&\quad + 2\left(\frac{\sum_{j=1}^{n_t} N_{tj} + \alpha}{\sum_{j=1}^{n_t} \xi_{tj} v_{tj} + \beta}\right)\sum_{j=1}^{n_t} \xi_{tj} v_{tj} + 2\sum_{j=1}^{n_t}\Big(\log(\Gamma(\hat{\kappa}_t v_{tj})) + \log\left(N_{tj}!\right)\Big) \\
&\quad + 2\sum_{j=1}^{n_t}\Big(-\hat{\kappa}_t v_{tj}\log(\hat{\kappa}_t v_{tj}) - \left(\hat{\kappa}_t v_{tj} + N_{tj} - 1\right)\log\left(\xi_{tj}\right) + \xi_{tj}\hat{\kappa}_t v_{tj}\Big) \\
&\quad + 2 + 4\left(\log\left(\sum_{j=1}^{n_t} N_{tj} + \alpha\right) - \psi\left(\sum_{j=1}^{n_t} N_{tj} + \alpha\right)\right)\sum_{j=1}^{n_t} N_{tj}.
\end{aligned}$$

For both NB models, the DIC of the tree $\mathcal{T}$ is obtained by using (2.14).

Based on the above discussion, we extend Algorithm 2.3 (see Subsection 2.2.3) to a new Algorithm 3.1 to simulate a Markov chain sequence of pairs $(\boldsymbol{\theta}^{(1)}, \mathcal{T}^{(1)})$, $(\boldsymbol{\theta}^{(2)}, \mathcal{T}^{(2)}), \ldots,$ starting from the root node. For the convenience of reference, we shall describe a general algorithm that is needed for the NB BCART models. Specifically, $\boldsymbol{\theta} = (\boldsymbol{\theta_M}, \boldsymbol{\theta_B})$, where $\boldsymbol{\theta_M}$ is the parameter that is treated as known and is computed using MME (or MLE), and $\boldsymbol{\theta_B}$ is the unknown parameter that needs to be estimated in the Bayesian framework. This newly proposed algorithm is applicable when it involves not only the data augmentation technique but also

includes both known and unknown parameters that need to be estimated using different methods, thereby extending the scope of the application of BCART models. With the above formulas derived in the two subsections for NB models, we can use the three-step approach proposed in Subsection 2.2.4, together with Algorithm 3.1 (treat $\boldsymbol{\theta_M} = \boldsymbol{\kappa}$, $\boldsymbol{\theta_B} = \boldsymbol{\lambda}$, and $\boldsymbol{z} = \boldsymbol{\xi}$), to search for an optimal tree which can then be used to predict new data.

Algorithm 3.1: One step of the MCMC algorithm for the BCART models parameterized by $(\boldsymbol{\theta_M}, \boldsymbol{\theta_B}, \mathcal{T})$ using data augmentation with both known and unknown parameters

Input: Data $(\boldsymbol{X}, \boldsymbol{y})$ and current values $(\hat{\boldsymbol{\theta}}_{\boldsymbol{M}}^{(m)}, \boldsymbol{\theta}_{\boldsymbol{B}}^{(m)}, \boldsymbol{z}^{(m)}, \mathcal{T}^{(m)})$

1: Generate a candidate value $\mathcal{T}^*$ with probability distribution $q(\mathcal{T}^{(m)}, \mathcal{T}^*)$

2: Estimate $\hat{\boldsymbol{\theta}}_{\boldsymbol{M}}^{(m+1)}$, using MME (or MLE)

3: Propose $\boldsymbol{z}^{(m+1)} \sim p(\boldsymbol{z} \mid \boldsymbol{X}, \boldsymbol{y}, \hat{\boldsymbol{\theta}}_{\boldsymbol{M}}^{(m+1)}, \boldsymbol{\theta}_{\boldsymbol{B}}^{(m)}, \mathcal{T}^{(m)})$

4: Set the acceptance ratio

$$\alpha(\mathcal{T}^{(m)}, \mathcal{T}^*) = \min\left\{\frac{q(\mathcal{T}^*, \mathcal{T}^{(m)}) p(\boldsymbol{y}, \boldsymbol{z}^{(m+1)} \mid \boldsymbol{X}, \hat{\boldsymbol{\theta}}_{\boldsymbol{M}}^{(m+1)}, \mathcal{T}^*) p(\mathcal{T}^*)}{q(\mathcal{T}^{(m)}, \mathcal{T}^*) p(\boldsymbol{y}, \boldsymbol{z}^{(m)} \mid \boldsymbol{X}, \hat{\boldsymbol{\theta}}_{\boldsymbol{M}}^{(m)}, \mathcal{T}^{(m)}) p(\mathcal{T}^{(m)})}, 1\right\}$$

5: Update $\mathcal{T}^{(m+1)} = \mathcal{T}^*$ with probability $\alpha(\mathcal{T}^{(m)}, \mathcal{T}^*)$, otherwise, set $\mathcal{T}^{(m+1)} = \mathcal{T}^{(m)}$

6: Sample $\boldsymbol{\theta}_{\boldsymbol{B}}^{(m+1)} \sim p(\boldsymbol{\theta_B} \mid \boldsymbol{X}, \boldsymbol{y}, \hat{\boldsymbol{\theta}}_{\boldsymbol{M}}^{(m+1)}, \boldsymbol{z}^{(m+1)}, \mathcal{T}^{(m+1)})$

Output: New values $(\hat{\boldsymbol{\theta}}_{\boldsymbol{M}}^{(m+1)}, \boldsymbol{\theta}_{\boldsymbol{B}}^{(m+1)}, \boldsymbol{z}^{(m+1)}, \mathcal{T}^{(m+1)})$

Remark 3.3 *(a) Similar to Algorithms 2.2 and 2.3, the sampling steps in Algorithm 3.1 should be done when necessary.*

(b) It is worth noting that our way of dealing with parameter κ is different from that in Murray (2021) where a single κ is sampled from a distribution and used for all terminal nodes. It turns out that that way of dealing with κ cannot give us good estimates in our simulation examples, whereas our way of first estimating κ using MME for each node can give good estimates.

(c) There are other ways to parameterize the NB distribution; see, e.g., Zhou et al. (2012). However, it looks that these ways are normally discussed when there is no exposure involved. Since the involvement of exposure is one of the key features in insurance claims frequency analysis, we will not cover them here.

Table 3.2: Evaluation metrics for NB-BCART. ϵ_t denotes the empirical claims frequency in node t, computed as $\sum_{j=1}^{n_t} N_{tj} / \sum_{j=1}^{n_t} v_{tj}$. $\bar{\lambda}_t$ and $\hat{\kappa}_t$ are parameter estimations that can be obtained from (3.18) and (3.11) (or (3.22)) respectively.

	Formulas
RSS($\boldsymbol{N}$)	$\sum_{t=1}^{b} \sum_{j=1}^{n_t} (N_{tj} - \bar{\lambda}_t v_{tj})^2$
SE	$\sum_{t=1}^{b} (\epsilon_t - \bar{\lambda}_t)^2$
DS	$\sum_{t=1}^{b} \left((\epsilon_t - \bar{\lambda}_t)^2 / (\bar{\lambda}_t (1 + \bar{\lambda}_t / \hat{\kappa}_t)) \right)$

(d) It should be noted that Algorithm 3.1 can be easily extended to accommodate multivariate parameters for both $\boldsymbol{\theta_M}$ and $\boldsymbol{\theta_B}$. This will become clear in the context of ZINB models that follow in this chapter.

As in the previous section, we can obtain the formulas for some of the evaluation metrics based on NB distributions in Table 3.2.

3.3 Zero-Inflated Poisson-Bayesian CART

Insurance claims data normally involves a large volume of zeros. Many policyholders incur no claims, which does not necessarily mean that they were involved in no accidents, but they are probably less risky. Unlike Section 3.2, there is another data augmentation way proposed in Diebolt & Robert (1994) that can be used. Depending on which data augmentation method is used and how the exposure is embedded in the model to better reflect the excessive zeros (see Lee (2021)), we discuss four ZIP models in this section. Although the four models involve similar treatments, to maintain the completeness of the content, calculation procedures and results are still included in the main text, with differences between them emphasized.

3.3.1 Zero-Inflated Poisson Model 1 (ZIP1)

For terminal node t, we use the following ZIP distribution by embedding the exposure into the Poisson part (see Murray (2021)),

$$f_{\text{ZIP1}} \left(N_{tj} \mid \mu_t, \lambda_t, v_{tj} \right)$$

$$= \begin{cases} \frac{1}{1+\mu_t} + \frac{\mu_t}{1+\mu_t} f_{\mathrm{P}}(0 \mid \lambda_t, v_{tj}) & N_{tj} = 0, \\ \frac{\mu_t}{1+\mu_t} f_{\mathrm{P}}(N_{tj} \mid \lambda_t, v_{tj}) & N_{tj} = 1, 2, \ldots, \end{cases}$$
$$= \frac{1}{1+\mu_t} I_{(N_{tj}=0)} + \frac{\mu_t}{1+\mu_t} f_{\mathrm{P}}(N_{tj} \mid \lambda_t, v_{tj}), \quad N_{tj} = 0, 1, 2, \ldots, \tag{3.26}$$

where $f_{\mathrm{P}}(N_{tj} \mid \lambda_t, v_{tj})$ is given as in (3.1), and $\frac{1}{1+\mu_t} \in (0,1)$ is the probability that a zero is due to the point mass component. Note that for computational simplicity we consider a model with two parameters rather than three as in Murray (2021).

Similar to the NB model, a data augmentation scheme is needed for the ZIP model. To this end, we introduce two latent variables $\boldsymbol{\phi}_t = (\phi_{t1}, \phi_{t2}, \ldots, \phi_{tn_t}) \in (0,\infty)^{n_t}$ and $\boldsymbol{\delta}_t = (\delta_{t1}, \delta_{t2}, \ldots, \delta_{tn_t}) \in \{0,1\}^{n_t}$, and define the data augmented likelihood for the j-th data instance in terminal node t as

$$f_{\mathrm{ZIP1}}\left(N_{tj}, \delta_{tj}, \phi_{tj} \mid \mu_t, \lambda_t\right) = e^{-\phi_{tj}(1+\mu_t)} \left(\frac{\mu_t \left(\lambda_t v_{tj}\right)^{N_{tj}}}{N_{tj}!} e^{-\lambda_t v_{tj}} \right)^{\delta_{tj}}, \tag{3.27}$$

where the support of the function f_{ZIP1} is $\big(\{0\} \times \{0,1\} \times (0,\infty)\big) \cup \big(\mathbb{N} \times \{1\} \times (0,\infty)\big)$. This means that we impose $\delta_{tj} = 1$ when $N_{tj} \in \mathbb{N}$ (i.e., $N_{tj} \neq 0$). It can be shown that (3.26) is the marginal distribution of the above augmented distribution. Collecting terms in the augmented variables in (3.27), we have:

- $N_{tj} > 0$, then $\delta_{tj} = 1$.

  In this case, obviously,

  $$f_{\mathrm{ZIP1}}\left(N_{tj}, 0, \phi_{tj} \mid \mu_t, \lambda_t\right) = 0,$$

  and

  $$f_{\mathrm{ZIP1}}\left(N_{tj}, 1, \phi_{tj} \mid \mu_t, \lambda_t\right) = e^{-\phi_{tj}(1+\mu_t)} \frac{\mu_t \left(\lambda_t v_{tj}\right)^{N_{tj}}}{N_{tj}!} e^{-\lambda_t v_{tj}}.$$

  Then we can obtain

  $$\begin{aligned} &\int_0^\infty f_{\mathrm{ZIP1}}\big(N_{tj}, 1, \phi_{tj} \mid \mu_t, \lambda_t\big)\, d\phi_{tj} \\ &= \frac{\mu_t \left(\lambda_t v_{tj}\right)^{N_{tj}}}{N_{tj}!} e^{-\lambda_t v_{tj}} \int_0^\infty e^{-\phi_{tj}(1+\mu_t)} d\phi_{tj} \\ &= \frac{\left(\lambda_t v_{tj}\right)^{N_{tj}}}{N_{tj}!} e^{-\lambda_t v_{tj}} \frac{\mu_t}{1+\mu_t}, \end{aligned}$$

  which is consistent with the definition of ZIP1 model in (3.26) when $N_{tj} > 0$.

- $N_{tj} = 0$ and $\delta_{tj} = 0$.

  In this case, we know

  $$f_{\text{ZIP1}}\left(0, 0, \phi_{tj} \mid \mu_t, \lambda_t\right) = e^{-\phi_{tj}(1+\mu_t)},$$

  and then

  $$\int_0^\infty f_{\text{ZIP1}}\left(0, 0, \phi_{tj} \mid \mu_t, \lambda_t\right) d\phi_{tj} = \int_0^\infty e^{-\phi_{tj}(1+\mu_t)} d\phi_{tj} = \frac{1}{1+\mu_t}.$$

- $N_{tj} = 0$ and $\delta_{tj} = 1$.

  We can easily obtain

  $$f_{\text{ZIP1}}\left(0, 1, \phi_{tj} \mid \mu_t, \lambda_t\right) = e^{-\phi_{tj}(1+\mu_t)} \mu_t e^{-\lambda_t v_{tj}},$$

  and then

  $$\begin{aligned} &\int_0^\infty f_{\text{ZIP1}}\left(0, 1, \phi_{tj} \mid \mu_t, \lambda_t\right) d\phi_{tj} \\ &= \mu_t e^{-\lambda_t v_{tj}} \int_0^\infty e^{-\phi_{tj}(1+\mu_t)} d\phi_{tj} \\ &= e^{-\lambda_t v_{tj}} \frac{\mu_t}{1+\mu_t}. \end{aligned}$$

  Immediately following for $N_{tj} = 0$, sum over δ_{tj},

  $$f_{\text{ZIP1}}\left(0 \mid \mu_t, \lambda_t\right) = \frac{1}{1+\mu_t} + \frac{\mu_t}{1+\mu_t} e^{-\lambda_t v_{tj}},$$

  which aligns with the definition of ZIP1 model in (3.26) when $N_{tj} = 0$.

By conditional arguments, we can also check that δ_{tj}, given data $N_{tj} = 0$ and parameters (μ_t and λ_t), has a Bernoulli distribution, i.e.,

$$\delta_{tj} \mid N_{tj} = 0, \mu_t, \lambda_t \;\sim\; \text{Bern}\left(\frac{\mu_t e^{-\lambda_t v_{tj}}}{1 + \mu_t e^{-\lambda_t v_{tj}}}\right),$$

and $\delta_{tj} = 1$, given $N_{tj} > 0$. Furthermore, ϕ_{tj}, given the parameter μ_t, has an Exponential distribution, i.e.,

$$\phi_{tj} \mid \mu_t \;\sim\; \text{Exp}\left(1 + \mu_t\right).$$

It is noted that the augmented likelihood f_{ZIP1} in (3.27) can actually be factorized as two Gamma-type functions parameterized by μ_t and λ_t respectively. This observation motivates us to assume independent conjugate Gamma priors for μ_t and λ_t with hyper-parameters $\alpha_i, \beta_i > 0$, $i = 1, 2$ (cf. (3.2)). With these Gamma

priors, we can derive the integrated augmented likelihood for terminal node t as follows

$$
\begin{aligned}
& p_{\text{ZIP1}}\left(\boldsymbol{N}_t, \boldsymbol{\delta}_t, \boldsymbol{\phi}_t \mid \boldsymbol{X}_t, \boldsymbol{v}_t\right) \\
&= \int_0^\infty \int_0^\infty f_{\text{ZIP1}}\left(\boldsymbol{N}_t, \boldsymbol{\delta}_t, \boldsymbol{\phi}_t \mid \mu_t, \lambda_t\right) p(\mu_t) p(\lambda_t) d\mu_t d\lambda_t \\
&= \int_0^\infty \int_0^\infty \prod_{j=1}^{n_t} \left(e^{-\phi_{tj}(1+\mu_t)} \left(\frac{\mu_t \left(\lambda_t v_{tj}\right)^{N_{tj}}}{N_{tj}!} e^{-\lambda_t v_{tj}} \right)^{\delta_{tj}} \right) \\
&\quad \times \frac{\beta_1^{\alpha_1} \mu_t^{\alpha_1 - 1} e^{-\beta_1 \mu_t}}{\Gamma(\alpha_1)} \frac{\beta_2^{\alpha_2} {\lambda_t}^{\alpha_2 - 1} e^{-\beta_2 \lambda_t}}{\Gamma(\alpha_2)} d\mu_t d\lambda_t \\
&= \frac{\beta_1^{\alpha_1}}{\Gamma(\alpha_1)} \frac{\beta_2^{\alpha_2}}{\Gamma(\alpha_2)} \prod_{j=1}^{n_t} \left(e^{-\phi_{tj}} v_{tj}^{\delta_{tj} N_{tj}} \left(N_{tj}!\right)^{-\delta_{tj}} \right) \int_0^\infty \mu_t^{\sum_{j=1}^{n_t} \delta_{tj} + \alpha_1 - 1} e^{-\left(\sum_{j=1}^{n_t} \phi_{tj} + \beta_1\right)\mu_t} d\mu_t \\
&\quad \times \int_0^\infty \lambda_t^{\sum_{j=1}^{n_t} \delta_{tj} N_{tj} + \alpha_2 - 1} e^{-\left(\sum_{j=1}^{n_t} \delta_{tj} v_{tj} + \beta_2\right)\lambda_t} d\lambda_t \\
&= \frac{\beta_1^{\alpha_1}}{\Gamma(\alpha_1)} \frac{\beta_2^{\alpha_2}}{\Gamma(\alpha_2)} \prod_{j=1}^{n_t} \left(e^{-\phi_{tj}} v_{tj}^{\delta_{ti} N_{tj}} \left(N_{tj}!\right)^{-\delta_{tj}} \right) \\
&\quad \times \frac{\Gamma\left(\sum_{j=1}^{n_t} \delta_{tj} + \alpha_1\right)}{\left(\sum_{j=1}^{n_t} \phi_{tj} + \beta_1\right)^{\sum_{j=1}^{n_t} \delta_{tj} + \alpha_1}} \frac{\Gamma\left(\sum_{j=1}^{n_t} \delta_{tj} N_{tj} + \alpha_2\right)}{\left(\sum_{j=1}^{n_t} \delta_{tj} v_{tj} + \beta_2\right)^{\sum_{j=1}^{n_t} \delta_{tj} N_{tj} + \alpha_2}}.
\end{aligned} \tag{3.28}
$$

Moreover, from the above, we see that the posterior distributions of μ_t, λ_t given the augmented data $(\boldsymbol{N}_t, \boldsymbol{\delta}_t, \boldsymbol{\phi}_t)$ are given by

$$
\begin{aligned}
\mu_t \mid \boldsymbol{\delta}_t, \boldsymbol{\phi}_t &\sim \text{Gamma}\left(\sum_{j=1}^{n_t} \delta_{tj} + \alpha_1, \sum_{j=1}^{n_t} \phi_{tj} + \beta_1 \right), \\
\lambda_t \mid \boldsymbol{N}_t, \boldsymbol{\delta}_t &\sim \text{Gamma}\left(\sum_{j=1}^{n_t} \delta_{tj} N_{tj} + \alpha_2, \sum_{j=1}^{n_t} \delta_{tj} v_{tj} + \beta_2 \right).
\end{aligned}
$$

The integrated augmented likelihood for the tree $\mathcal{T}$ is thus given by

$$
p_{\text{ZIP1}}\left(\boldsymbol{N}, \boldsymbol{\delta}, \boldsymbol{\phi} \mid \boldsymbol{X}, \boldsymbol{v}, \mathcal{T}\right) = \prod_{t=1}^{b} p_{\text{ZIP1}}\left(\boldsymbol{N}_t, \boldsymbol{\delta}_t, \boldsymbol{\phi}_t \mid \boldsymbol{X}_t, \boldsymbol{v}_t\right). \tag{3.29}
$$

Now, we discuss the DIC for this tree which can be derived as a special case of (2.15) with $\boldsymbol{\theta}_t = (\mu_t, \lambda_t)$. To this end, we first focus on DIC_t of terminal node t. It follows that

$$
D(\bar{\mu}_t, \bar{\lambda}_t) \;=\; -2 \log f_{\text{ZIP1}}(\boldsymbol{N}_t \mid \bar{\mu}_t, \bar{\lambda}_t)
$$

$$= -2\sum_{j=1}^{n_t} \log\left(\frac{1}{1+\bar{\mu}_t} I_{(N_{tj}=0)} + \frac{\bar{\mu}_t}{1+\bar{\mu}_t} \frac{(\bar{\lambda}_t v_{tj})^{N_{tj}}}{N_{tj}!} e^{-\bar{\lambda}_t v_{tj}}\right), \quad (3.30)$$

where

$$\bar{\mu}_t = \frac{\sum_{j=1}^{n_t} \delta_{tj} + \alpha_1}{\sum_{j=1}^{n_t} \phi_{tj} + \beta_1}, \qquad \bar{\lambda}_t = \frac{\sum_{j=1}^{n_t} \delta_{tj} N_{tj} + \alpha_2}{\sum_{j=1}^{n_t} \delta_{tj} v_{tj} + \beta_2}. \quad (3.31)$$

Next, since

$$\begin{aligned}
&\log f_{\text{ZIP1}}\left(\boldsymbol{N}_t, \boldsymbol{\delta}_t, \boldsymbol{\phi}_t \mid \mu_t, \lambda_t\right) \\
&= \sum_{j=1}^{n_t} \Big[-\phi_{tj}(1+\mu_t) + \delta_{tj} \log\left(\mu_t\right) + \delta_{tj} N_{tj} \log\left(\lambda_t v_{tj}\right) - \delta_{tj} \lambda_t v_{tj} - \delta_{tj} \log\left(N_{tj}!\right)\Big],
\end{aligned}$$

we can derive that

$$\begin{aligned}
q_{Dt} &= -2\mathbb{E}_{\text{post}}\left[\log f_{\text{ZIP1}}(\boldsymbol{N}_t, \boldsymbol{\delta}_t, \boldsymbol{\phi}_t \mid \mu_t, \lambda_t)\right] + 2\log f_{\text{ZIP1}}(\boldsymbol{N}_t, \boldsymbol{\delta}_t, \boldsymbol{\phi}_t \mid \bar{\mu}_t, \bar{\lambda}_t) \\
&= 2\left(\log\left(\sum_{j=1}^{n_t} \delta_{tj} + \alpha_1\right) - \psi\left(\sum_{j=1}^{n_t} \delta_{tj} + \alpha_1\right)\right) \sum_{j=1}^{n_t} \delta_{tj} \\
&\quad + 2\left(\log\left(\sum_{j=1}^{n_t} \delta_{tj} N_{tj} + \alpha_2\right) - \psi\left(\sum_{j=1}^{n_t} \delta_{tj} N_{tj} + \alpha_2\right)\right) \sum_{j=1}^{n_t} \delta_{tj} N_{tj}. \quad (3.32)
\end{aligned}$$

Therefore, DIC_t can be obtained from (3.30) and (3.32) as

$$\begin{aligned}
\text{DIC}_t =& D\left(\bar{\mu}_t, \bar{\lambda}_t\right) + 2q_{Dt} \\
=& -2\sum_{j=1}^{n_t} \log\left(\frac{1}{1+\bar{\mu}_t} I_{(N_{tj}=0)} + \frac{\bar{\mu}_t}{1+\bar{\mu}_t} \frac{(\bar{\lambda}_t v_{tj})^{N_{tj}}}{N_{tj}!} e^{-\bar{\lambda}_t v_{tj}}\right) \\
&+ 4\left(\log\left(\sum_{j=1}^{n_t} \delta_{tj} + \alpha_1\right) - \psi\left(\sum_{j=1}^{n_t} \delta_{tj} + \alpha_1\right)\right) \sum_{j=1}^{n_t} \delta_{tj} \\
&+ 4\left(\log\left(\sum_{j=1}^{n_t} \delta_{tj} N_{tj} + \alpha_2\right) - \psi\left(\sum_{j=1}^{n_t} \delta_{tj} N_{tj} + \alpha_2\right)\right) \sum_{j=1}^{n_t} \delta_{tj} N_{tj}.
\end{aligned}$$

Remark 3.4 *ZIP2, ZIP3 and all ZINB models discussed in the following context, also all ZICPG models discussed in Chapter 5 shall use a similar approach to prove that the data augmented likelihood is equal to the data likelihood when the latent variables are integrated out. We provide details only for the ZIP1 model in this subsection, and others are omitted to avoid repetition.*

3.3.2 Zero-Inflated Poisson Model 2 (ZIP2)

For terminal node t, we use the following ZIP distribution by embedding the exposure into the zero mass part (see Lee (2021)),

$$f_{\text{ZIP2}}\left(N_{tj} \mid \mu_t, \lambda_t, v_{tj}\right) = \begin{cases} \frac{1}{1+\mu_t v_{tj}} + \frac{\mu_t v_{tj}}{1+\mu_t v_{tj}} e^{-\lambda_t} & N_{tj} = 0, \\ \frac{\mu_t v_{tj}}{1+\mu_t v_{tj}} \frac{\lambda_t^{N_{tj}}}{N_{tj}!} e^{-\lambda_t} & N_{tj} = 1, 2, \ldots, \end{cases} \tag{3.33}$$

where $\frac{1}{1+\mu_t v_{tj}} \in (0,1)$ is the probability that a zero is due to the point mass component. This formulation stems from an intuitive inverse relationship between the exposure and the probability of zero mass. This way of embedding exposure has been justified to be more effective in Lee (2021).

Similar to before, we introduce the same two latent variables $\boldsymbol{\phi}_t$ and $\boldsymbol{\delta}_t$, and define the data augmented likelihood for the j-th data instance in terminal node t as

$$f_{\text{ZIP2}}\left(N_{tj}, \delta_{tj}, \phi_{tj} \mid \mu_t, \lambda_t\right) = e^{-\phi_{tj}\left(1+\mu_t v_{tj}\right)} \left(\frac{\mu_t v_{tj} \lambda_t^{N_{tj}}}{N_{tj}!} e^{-\lambda_t}\right)^{\delta_{tj}}, \tag{3.34}$$

where the support of the function f_{ZIP2} is the same as in the ZIP1 model. Besides, it can be shown that (3.33) is the marginal distribution of the above augmented distribution.

By conditional arguments, we can also check that δ_{tj}, given data $N_{tj} = 0$ and parameters (μ_t and λ_t), has a Bernoulli distribution, i.e.,

$$\delta_{tj} \mid N_{tj} = 0, \mu_t, \lambda_t \;\sim\; \text{Bern}\left(\frac{\mu_t v_{tj} e^{-\lambda_t}}{1 + \mu_t v_{tj} e^{-\lambda_t}}\right),$$

and $\delta_{tj} = 1$, given $N_{tj} > 0$. Furthermore, ϕ_{tj}, given the parameter μ_t, has an Exponential distribution, i.e.,

$$\phi_{tj} \mid \mu_t \;\sim\; \text{Exp}\left(1 + \mu_t v_{tj}\right).$$

As before, we assume independent conjugate Gamma priors for μ_t and λ_t with hyper-parameters $\alpha_i, \beta_i > 0$, $i = 1, 2$. Given the data augmented likelihood in (3.34) and Gamma priors, we can obtain the integrated augmented likelihood for terminal node t as follows

$$\begin{aligned} &p_{\text{ZIP2}}\left(\boldsymbol{N}_t, \boldsymbol{\delta}_t, \boldsymbol{\phi}_t \mid \boldsymbol{X}_t, \boldsymbol{v}_t\right) \\ &= \int_0^\infty \int_0^\infty f_{\text{ZIP2}}\left(\boldsymbol{N}_t, \boldsymbol{\delta}_t, \boldsymbol{\phi}_t \mid \mu_t, \lambda_t\right) p(\mu_t) p(\lambda_t) d\mu_t d\lambda_t \end{aligned}$$

$$
\begin{aligned}
&= \int_0^\infty \int_0^\infty \prod_{j=1}^{n_t} \left(e^{-\phi_{tj}\left(1+\mu_t v_{tj}\right)} \left(\frac{\mu_t v_{tj} \lambda_t^{N_{tj}}}{N_{tj}!} e^{-\lambda_t} \right)^{\delta_{tj}} \right) \\
&\quad \times \frac{\beta_1^{\alpha_1} {\mu_t}^{\alpha_1 - 1} e^{-\beta_1 \mu_t}}{\Gamma(\alpha_1)} \frac{\beta_2^{\alpha_2} {\lambda_t}^{\alpha_2 - 1} e^{-\beta_2 \lambda_t}}{\Gamma(\alpha_2)} d\mu_t d\lambda_t \\
&= \frac{\beta_1^{\alpha_1}}{\Gamma(\alpha_1)} \frac{\beta_2^{\alpha_2}}{\Gamma(\alpha_2)} \prod_{j=1}^{n_t} \left(e^{-\phi_{tj}} \left(\frac{v_{tj}}{N_{tj}!} \right)^{\delta_{tj}} \right) \int_0^\infty \mu_t^{\sum_{j=1}^{n_t} \delta_{tj} + \alpha_1 - 1} e^{-\left(\sum_{j=1}^{n_t} \phi_{tj} v_{tj} + \beta_1\right)\mu_t} d\mu_t \\
&\quad \times \int_0^\infty \lambda_t^{\sum_{j=1}^{n_t} \delta_{tj} N_{tj} + \alpha_2 - 1} e^{-\left(\sum_{j=1}^{n_t} \delta_{tj} + \beta_2\right)\lambda_t} d\lambda_t \\
&= \frac{\beta_1^{\alpha_1}}{\Gamma(\alpha_1)} \frac{\beta_2^{\alpha_2}}{\Gamma(\alpha_2)} \prod_{j=1}^{n_t} \left(e^{-\phi_{tj}} \left(\frac{v_{tj}}{N_{tj}!} \right)^{\delta_{tj}} \right) \\
&\quad \times \frac{\Gamma\left(\sum_{j=1}^{n_t} \delta_{tj} + \alpha_1\right)}{\left(\sum_{j=1}^{n_t} \phi_{tj} v_{tj} + \beta_1\right)^{\sum_{j=1}^{n_t} \delta_{tj} + \alpha_1}} \frac{\Gamma\left(\sum_{j=1}^{n_t} \delta_{tj} N_{tj} + \alpha_2\right)}{\left(\sum_{j=1}^{n_t} \delta_{tj} + \beta_2\right)^{\sum_{j=1}^{n_t} \delta_{tj} N_{tj} + \alpha_2}}.
\end{aligned} \tag{3.35}
$$

Moreover, from the above, we see that the posterior distributions of μ_t, λ_t given the augmented data $(\boldsymbol{N}_t, \boldsymbol{\delta}_t, \boldsymbol{\phi}_t)$ are given by

$$
\begin{aligned}
\mu_t \mid \boldsymbol{\delta}_t, \boldsymbol{\phi}_t &\sim \text{Gamma}\left(\sum_{j=1}^{n_t} \delta_{tj} + \alpha_1, \sum_{j=1}^{n_t} \phi_{tj} v_{tj} + \beta_1 \right), \\
\lambda_t \mid \boldsymbol{N}_t, \boldsymbol{\delta}_t &\sim \text{Gamma}\left(\sum_{j=1}^{n_t} \delta_{tj} N_{tj} + \alpha_2, \sum_{j=1}^{n_t} \delta_{tj} + \beta_2 \right).
\end{aligned}
$$

The integrated augmented likelihood for the tree $\mathcal{T}$ is thus given by

$$
p_{\text{ZIP2}}\left(\boldsymbol{N}, \boldsymbol{\delta}, \boldsymbol{\phi} \mid \boldsymbol{X}, \boldsymbol{v}, \mathcal{T}\right) = \prod_{t=1}^{b} p_{\text{ZIP2}}\left(\boldsymbol{N}_t, \boldsymbol{\delta}_t, \boldsymbol{\phi}_t \mid \boldsymbol{X}_t, \boldsymbol{v}_t, \mathcal{T}\right). \tag{3.36}
$$

Now, we discuss the DIC_t of terminal node t. It follows that

$$
\begin{aligned}
D(\bar{\mu}_t, \bar{\lambda}_t) &= -2 \log f_{\text{ZIP2}}(\boldsymbol{N}_t \mid \bar{\mu}_t, \bar{\lambda}_t) \\
&= -2 \sum_{j=1}^{n_t} \log \left(\frac{1}{1 + \bar{\mu}_t v_{tj}} I_{(N_{tj}=0)} + \frac{\bar{\mu}_t v_{tj}}{1 + \bar{\mu}_t v_{tj}} \frac{\bar{\lambda}_t^{N_{tj}}}{N_{tj}!} e^{-\bar{\lambda}_t} \right),
\end{aligned} \tag{3.37}
$$

where

$$
\bar{\mu}_t = \frac{\sum_{j=1}^{n_t} \delta_{tj} + \alpha_1}{\sum_{j=1}^{n_t} \phi_{tj} v_{tj} + \beta_1}, \qquad \bar{\lambda}_t = \frac{\sum_{j=1}^{n_t} \delta_{tj} N_{tj} + \alpha_2}{\sum_{j=1}^{n_t} \delta_{tj} + \beta_2}. \tag{3.38}
$$

Next, since

$$
\log f_{\text{ZIP2}}\left(\boldsymbol{N}_t, \boldsymbol{\delta}_t, \boldsymbol{\phi}_t \mid \mu_t, \lambda_t\right)
$$

$$= \sum_{j=1}^{n_t} \Big[-\phi_{tj}(1+\mu_t v_{tj}) + \delta_{tj} \log\left(\mu_t v_{tj}\right) + \delta_{tj} N_{tj} \log\left(\lambda_t\right) - \delta_{tj}\lambda_t - \delta_{tj} \log\left(N_{tj}!\right) \Big],$$

we can derive the same expression for q_{Dt} as in (3.32). Therefore, we can obtain $\text{DIC}_t = D(\bar{\mu}_t, \bar{\lambda}_t) + 2q_{Dt}$ directly from (3.37) and (3.32).

3.3.3 Zero-Inflated Poisson Model 3 (ZIP3)

After considering ZIP1 and ZIP2 models, which embed the exposure into different parts, it is natural to consider placing the exposure into both the Poisson part and the zero mass part. For terminal node t, we use the following ZIP distribution

$$f_{\text{ZIP3}}\left(N_{tj} \mid \mu_t, \lambda_t, v_{tj}\right) = \begin{cases} \frac{1}{1+\mu_t v_{tj}} + \frac{\mu_t v_{tj}}{1+\mu_t v_{tj}} f_{\text{P}}(0 \mid \lambda_t, v_{tj}) & N_{tj} = 0, \\ \frac{\mu_t v_{tj}}{1+\mu_t v_{tj}} f_{\text{P}}(N_{tj} \mid \lambda_t, v_{tj}) & N_{tj} = 1, 2, \ldots, \end{cases} \tag{3.39}$$

where $f_{\text{P}}(N_{tj} \mid \lambda_t, v_{tj})$ is given as in (3.1), and $\frac{1}{1+\mu_t v_{tj}} \in (0,1)$ is the probability that a zero is due to the point mass component.

Similar to before, we introduce the same two latent variables $\boldsymbol{\phi}_t$ and $\boldsymbol{\delta}_t$, and define the data augmented likelihood for the j-th data instance in terminal node t as

$$f_{\text{ZIP3}}\left(N_{tj}, \delta_{tj}, \phi_{tj} \mid \mu_t, \lambda_t\right) = e^{-\phi_{tj}(1+\mu_t v_{tj})} \left(\frac{\mu_t v_{tj} \left(\lambda_t v_{tj}\right)^{N_{tj}}}{N_{tj}!} e^{-\lambda_t v_{tj}} \right)^{\delta_{tj}}, \tag{3.40}$$

where the support of the function f_{ZIP3} is the same as in the ZIP1 model. Besides, it can be shown that (3.39) is the marginal distribution of the above augmented distribution.

By conditional arguments, we can also check that δ_{tj}, given data $N_{tj} = 0$ and parameters (μ_t and λ_t), has a Bernoulli distribution, i.e.,

$$\delta_{tj} \mid N_{tj} = 0, \mu_t, \lambda_t \;\sim\; \text{Bern}\left(\frac{\mu_t v_{tj} e^{-\lambda_t v_{tj}}}{1 + \mu_t v_{tj} e^{-\lambda_t v_{tj}}} \right),$$

and $\delta_{tj} = 1$, given $N_{tj} > 0$. Furthermore, ϕ_{tj}, given the parameter μ_t, has an Exponential distribution, i.e.,

$$\phi_{tj} \mid \mu_t \;\sim\; \text{Exp}\left(1 + \mu_t v_{tj}\right).$$

As before, we assume independent conjugate Gamma priors for μ_t and λ_t with hyper-parameters $\alpha_i, \beta_i > 0$, $i = 1, 2$. Then, we can derive the integrated augmented likelihood for terminal node t as follows

$$p_{\text{ZIP3}}\left(\boldsymbol{N}_t, \boldsymbol{\delta}_t, \boldsymbol{\phi}_t \mid \boldsymbol{X}_t, \boldsymbol{v}_t\right)$$

$$
\begin{aligned}
&= \int_0^\infty \int_0^\infty f_{\text{ZIP3}}\left(\boldsymbol{N}_t, \boldsymbol{\delta}_t, \boldsymbol{\phi}_t \mid \mu_t, \lambda_t\right) p(\mu_t) p(\lambda_t) d\mu_t d\lambda_t \\
&= \int_0^\infty \int_0^\infty \prod_{j=1}^{n_t} \left(e^{-\phi_{tj}(1+\mu_t v_{tj})} \left(\frac{\mu_t v_{tj} \left(\lambda_t v_{tj}\right)^{N_{tj}}}{N_{tj}!} e^{-\lambda_t v_{tj}} \right)^{\delta_{tj}} \right) \\
&\quad \times \frac{\beta_1^{\alpha_1} {\mu_t}^{\alpha_1 - 1} e^{-\beta_1 \mu_t}}{\Gamma(\alpha_1)} \frac{\beta_2^{\alpha_2} {\lambda_t}^{\alpha_2 - 1} e^{-\beta_2 \lambda_t}}{\Gamma(\alpha_2)} d\mu_t d\lambda_t \\
&= \prod_{j=1}^{n_t} \left(e^{-\phi_{tj}} v_{tj}^{\delta_{tj}(1+N_{tj})} \left(N_{tj}!\right)^{-\delta_{tj}} \right) \int_0^\infty \mu_t^{\sum_{j=1}^{n_t} \delta_{tj} + \alpha_1 - 1} e^{-\left(\sum_{j=1}^{n_t} \phi_{tj} v_{tj} + \beta_1\right)\mu_t} d\mu_t \\
&\quad \times \int_0^\infty \lambda_t^{\sum_{j=1}^{n_t} \delta_{tj} N_{tj} + \alpha_2 - 1} e^{-\left(\sum_{j=1}^{n_t} \delta_{tj} v_{tj} + \beta_2\right)\lambda_t} d\lambda_t \times \frac{\beta_1^{\alpha_1}}{\Gamma(\alpha_1)} \frac{\beta_2^{\alpha_2}}{\Gamma(\alpha_2)} \\
&= \frac{\beta_1^{\alpha_1}}{\Gamma(\alpha_1)} \frac{\beta_2^{\alpha_2}}{\Gamma(\alpha_2)} \prod_{j=1}^{n_t} \left(e^{-\phi_{tj}} v_{tj}^{\delta_{tj}(1+N_{tj})} \left(N_{tj}!\right)^{-\delta_{tj}} \right) \\
&\quad \times \frac{\Gamma\left(\sum_{j=1}^{n_t} \delta_{tj} + \alpha_1\right)}{\left(\sum_{j=1}^{n_t} \phi_{tj} v_{tj} + \beta_1\right)^{\sum_{j=1}^{n_t} \delta_{tj} + \alpha_1}} \frac{\Gamma\left(\sum_{j=1}^{n_t} \delta_{tj} N_{tj} + \alpha_2\right)}{\left(\sum_{j=1}^{n_t} \delta_{tj} v_{tj} + \beta_2\right)^{\sum_{j=1}^{n_t} \delta_{tj} N_{tj} + \alpha_2}}.
\end{aligned} \tag{3.41}
$$

Moreover, from the above, we see that the posterior distributions of μ_t, λ_t given the augmented data $(\boldsymbol{N}_t, \boldsymbol{\delta}_t, \boldsymbol{\phi}_t)$ are given by

$$
\begin{aligned}
\mu_t \mid \boldsymbol{\delta}_t, \boldsymbol{\phi}_t &\sim \text{Gamma}\left(\sum_{j=1}^{n_t} \delta_{tj} + \alpha_1, \sum_{j=1}^{n_t} \phi_{tj} v_{tj} + \beta_1 \right), \\
\lambda_t \mid \boldsymbol{N}_t, \boldsymbol{\delta}_t &\sim \text{Gamma}\left(\sum_{j=1}^{n_t} \delta_{tj} N_{tj} + \alpha_2, \sum_{j=1}^{n_t} \delta_{tj} v_{tj} + \beta_2 \right).
\end{aligned}
$$

The integrated augmented likelihood for the tree $\mathcal{T}$ is thus given by

$$
p_{\text{ZIP3}}\left(\boldsymbol{N}, \boldsymbol{\delta}, \boldsymbol{\phi} \mid \boldsymbol{X}, \boldsymbol{v}, \mathcal{T}\right) = \prod_{t=1}^{b} p_{\text{ZIP3}}\left(\boldsymbol{N}_t, \boldsymbol{\delta}_t, \boldsymbol{\phi}_t \mid \boldsymbol{X}_t, \boldsymbol{v}_t\right). \tag{3.42}
$$

Now, we discuss the DIC_t of terminal node t. It follows that

$$
\begin{aligned}
&D(\bar{\mu}_t, \bar{\lambda}_t) \\
&= -2 \log f_{\text{ZIP3}}(\boldsymbol{N}_t \mid \bar{\mu}_t, \bar{\lambda}_t) \\
&= -2 \sum_{j=1}^{n_t} \log \left(\frac{1}{1 + \bar{\mu}_t v_{tj}} I_{(N_{tj}=0)} + \frac{\bar{\mu}_t v_{tj}}{1 + \bar{\mu}_t v_{tj}} \frac{(\bar{\lambda}_t v_{tj})^{N_{tj}}}{N_{tj}!} e^{-\bar{\lambda}_t v_{tj}} \right),
\end{aligned} \tag{3.43}
$$

where

$$
\bar{\mu}_t = \frac{\sum_{j=1}^{n_t} \delta_{tj} + \alpha_1}{\sum_{j=1}^{n_t} \phi_{tj} v_{tj} + \beta_1}, \qquad \bar{\lambda}_t = \frac{\sum_{j=1}^{n_t} \delta_{tj} N_{tj} + \alpha_2}{\sum_{j=1}^{n_t} \delta_{tj} v_{tj} + \beta_2}. \tag{3.44}
$$

Next, since

$$\begin{aligned}&\log f_{\text{ZIP3}}\left(\boldsymbol{N}_t, \boldsymbol{\delta}_t, \boldsymbol{\phi}_t \mid \mu_t, \lambda_t\right)\\&= \sum_{j=1}^{n_t}\Big[-\phi_{tj}(1+\mu_t v_{tj}) + \delta_{tj}\log\left(\mu_t v_{tj}\right) + \delta_{tj}N_{tj}\log\left(\lambda_t v_{tj}\right) - \delta_{tj}\lambda_t v_{tj} - \delta_{tj}\log\left(N_{tj}!\right)\Big],\end{aligned}$$

we still can derive the same expression for q_{Dt} as in (3.32). Therefore, we can obtain $\text{DIC}_t = D(\bar{\mu}_t, \bar{\lambda}_t) + 2q_{Dt}$ directly from (3.43) and (3.32).

3.3.4 Zero-Inflated Poisson Model 4 (ZIP4)

Different from the introduction of two latent variables in the previous subsections, this subsection shall employ another data augmentation method in the ZIP model, which only needs to introduce one latent variable; see, e.g., Tanner & Wong (1987), Rodrigues (2003) and Diebolt & Robert (1994). In the constructions of the models discussed therein with one latent variable, even if the latent variable can be integrated out from the data augmented likelihood, it does not lead to an equation that equals the data likelihood. Meanwhile, using two latent variables proposed in Murray (2021), the data augmented likelihood is equal to the data likelihood if the latent variables are integrated out, which should be more accurate. However, a model using one latent variable should work more efficiently and decrease the randomness compared to a model using two latent variables. Besides, Diebolt & Robert (1994) showed that the two likelihoods (data likelihood and data augmented likelihood) can achieve the same asymptotic convergence even if they are not equal, making this data augmentation method usable as well. We refer to Diebolt & Robert (1994) for a comprehensive analysis and explanation.

For terminal node t, we use the following ZIP distribution by embedding the exposure into the Poisson part,

$$f_{\text{ZIP4}}\left(N_{tj} \mid \omega_t, \lambda_t, v_{tj}\right) = \begin{cases} \omega_t + (1-\omega_t) f_{\text{P}}\left(0 \mid \lambda_t, v_{tj}\right) & N_{tj} = 0 \\ (1-\omega_t) f_{\text{P}}\left(N_{tj} \mid \lambda_t, v_{tj}\right) & N_{tj} = 1, 2, \ldots, \end{cases} \tag{3.45}$$

where $f_{\text{P}}(N_{tj} \mid \lambda_t, v_{tj})$ is given as in (3.1), and ω_t is the probability that a zero is due to the point mass component. Let $M_t = \{N_{tj} : N_{tj} = 0 \ (j = 1, 2, \cdots, n_t)\}$ and $r_t = \#(M_t)$, the likelihood function for terminal node t can be rewritten as:

$$\begin{aligned}&f_{\text{ZIP4}}\left(\boldsymbol{N}_t \mid \omega_t, \lambda_t\right)\\&= \prod_{j:N_{tj}=0} f_{\text{ZIP4}}\left(N_{tj} \mid \omega_t, \lambda_t\right) \prod_{j:N_{tj}>0} f_{\text{ZIP4}}\left(N_{tj} \mid \omega_t, \lambda_t\right)\end{aligned}$$

$$= \prod_{j:N_{tj}=0} \left(\omega_t + (1-\omega_t)e^{-\lambda_t v_{tj}}\right) \prod_{j:N_{tj}>0} (1-\omega_t) \frac{e^{-\lambda_t v_{tj}} \left(\lambda_t v_{tj}\right)^{N_{tj}}}{N_{tj}!}$$

$$= \prod_{j:N_{tj}=0} \left(\omega_t + (1-\omega_t)e^{-\lambda_t v_{tj}}\right) (1-\omega_t)^{n_t - r_t} \prod_{j:N_{tj}>0} \frac{e^{-\lambda_t v_{tj}} \left(\lambda_t v_{tj}\right)^{N_{tj}}}{N_{tj}!}.$$

The elements of the set M_t come from two different groups, either the degenerate distribution at zero or $f_{\mathrm{P}}(N_{tj} = 0)$. In this case, it is natural to define an unobserved latent variable:

$$U_{tj} = \begin{cases} 1, & p(\omega_t, \lambda_t) \\ 0, & 1 - p(\omega_t, \lambda_t) \end{cases}$$

where $j = 1, 2, \cdots, r_t$ and

$$p(\omega_t, \lambda_t) = \frac{\omega_t}{\omega_t + (1-\omega_t)e^{-\lambda_t v_{tj}}}.$$

The latent variable U_{tj} indicates whether the j-th element of set M_t is drawn from the degenerate distribution at zero or not. Therefore, the likelihood function based on the augmented data $(\boldsymbol{N}_t, \boldsymbol{U}_t)$ for terminal node t is:

$$\begin{aligned}
& f_{\mathrm{ZIP4}}\left(\boldsymbol{N}_t, \boldsymbol{U}_t \mid \omega_t, \lambda_t\right) \\
&= f_{\mathrm{ZIP4}}\left(\boldsymbol{N}_t \mid \omega_t, \lambda_t\right) \prod_{j=1}^{r_t} p(\omega_t, \lambda_t)^{U_{tj}} \left(1 - p(\omega_t, \lambda_t)\right)^{1-U_{tj}} \\
&= \prod_{j:N_{tj}=0} \left(\omega_t + (1-\omega_t)e^{-\lambda_t v_{tj}}\right) (1-\omega_t)^{n_t - r_t} \prod_{j:N_{tj}>0} \frac{e^{-\lambda_t v_{tj}} \left(\lambda_t v_{tj}\right)^{N_{tj}}}{N_{tj}!} \\
&\quad \times \prod_{j=1}^{r_t} \left(\frac{\omega_t}{\omega_t + (1-\omega_t)e^{-\lambda_t v_{tj}}}\right)^{U_{tj}} \left(\frac{(1-\omega_t)e^{-\lambda_t v_{tj}}}{\omega_t + (1-\omega_t)e^{-\lambda_t v_{tj}}}\right)^{1-U_{tj}} \\
&= (1-\omega_t)^{n_t - r_t + r_t - \sum_{j=1}^{r_t} U_{tj}} \left(\prod_{j:N_{tj}>0} \frac{e^{-\lambda_t v_{tj}} \left(\lambda_t v_{tj}\right)^{N_{tj}}}{N_{tj}!}\right) \omega_t^{\sum_{j=1}^{r_t} U_{tj}} e^{-\lambda_t \sum_{j=1}^{r_t} v_{tj}(1-U_{tj})} \\
&= (1-\omega_t)^{n_t - \sum_{j=1}^{r_t} U_{tj}} \omega_t^{\sum_{j=1}^{r_t} U_{tj}} e^{-\lambda_t \left(\sum_{j=1}^{n_t} v_{tj} - \sum_{j=1}^{r_t} v_{tj} U_{tj}\right)} \lambda_t^{\sum_{j:N_{tj}>0} N_{tj}} \prod_{j:N_{tj}>0} \frac{v_{tj}^{N_{tj}}}{N_{tj}!}.
\end{aligned} \tag{3.46}$$

By conditional arguments, we can check that U_{tj}, given parameters (ω_t and λ_t), has a Bernoulli distribution, i.e.,

$$U_{tj} \mid \omega_t, \lambda_t \ \sim \ \mathrm{Bern}\left(1, \frac{\omega_t}{\omega_t + (1-\omega_t)e^{-\lambda_t v_{tj}}}\right).$$

This observation of data augmented likelihood motivates us to assume independent conjugate Beta and Gamma priors for ω_t and λ_t respectively with hyperparameters $\alpha_i, \beta_i > 0, i = 1, 2$, that is,

$$p(\omega_t) = \frac{\omega_t^{\alpha_1 - 1}(1-\omega_t)^{\beta_1 - 1}}{\mathrm{B}(\alpha_1, \beta_1)},$$

$$p(\lambda_t) = \frac{\beta_2^{\alpha_2} {\lambda_t}^{\alpha_2 - 1} e^{-\beta_2 \lambda_t}}{\Gamma(\alpha_2)},$$

where $\mathrm{B}(\alpha_1, \beta_1) = \frac{\Gamma(\alpha_1)\Gamma(\beta_1)}{\Gamma(\alpha_1+\beta_1)}$. Then, the integrated likelihood for terminal node t can be obtained as

$$\begin{aligned}
&p_{\text{ZIP4}}\left(\boldsymbol{N}_t, \boldsymbol{U}_t \mid \boldsymbol{X}_t, \boldsymbol{v}_t\right) \\
&= \int_0^\infty \int_0^\infty f_{\text{ZIP4}}\left(\boldsymbol{N}_t, \boldsymbol{U}_t \mid \omega_t, \lambda_t\right) p(\omega_t) p(\lambda_t) d\omega_t d\lambda_t \\
&= \int_0^\infty \int_0^\infty (1-\omega_t)^{n_t - \sum_{j=1}^{r_t} U_{tj}} \omega_t^{\sum_{j=1}^{r_t} U_{tj}} e^{-\lambda_t\left(\sum_{j=1}^{n_t} v_{tj} - \sum_{j=1}^{r_t} v_{tj} U_{tj}\right)} \lambda_t^{\sum_{j:N_{tj}>0} N_{tj}} \\
&\quad \times \frac{\omega_t^{\alpha_1 - 1}(1-\omega_t)^{\beta_1 - 1}}{B(\alpha_1, \beta_1)} \frac{\beta_2^{\alpha_2} {\lambda_t}^{\alpha_2 - 1} e^{-\beta_2 \lambda_t}}{\Gamma(\alpha_2)} d\omega_t d\lambda_t \times \prod_{j:N_{tj}>0} \frac{v_{tj}^{N_{tj}}}{N_{tj}!} \\
&= \prod_{j:N_{tj}>0} \frac{v_{tj}^{N_{tj}}}{N_{tj}!} \frac{\beta_2^{\alpha_2}}{B(\alpha_1, \beta_1)\Gamma(\alpha_2)} \int_0^\infty (1-\omega_t)^{n_t - \sum_{j=1}^{r_t} U_{tj} + \beta_1 - 1} \omega_t^{\sum_{j=1}^{r_t} U_{tj} + \alpha_1 - 1} d\omega_t \\
&\quad \times \int_0^\infty e^{-\lambda_t\left(\sum_{j=1}^{n_t} v_{tj} - \sum_{j=1}^{r_t} v_{tj} U_{tj} + \beta_2\right)} \lambda_t^{\sum_{j:N_{tj}>0} N_{tj} + \alpha_2 - 1} d\lambda_t \\
&= \prod_{j:N_{tj}>0} \frac{v_{tj}^{N_{tj}}}{N_{tj}!} \frac{\beta_2^{\alpha_2}}{B(\alpha_1, \beta_1)\Gamma(\alpha_2)} \times B\left(\sum_{j=1}^{r_t} U_{tj} + \alpha_1, n_t - \sum_{j=1}^{r_t} U_{tj} + \beta_1\right) \\
&\quad \times \frac{\Gamma(\sum_{j:N_{tj}>0} N_{tj} + \alpha_2)}{\left(\sum_{j=1}^{n_t} v_{tj} - \sum_{j=1}^{r_t} v_{tj} U_{tj} + \beta_2\right)^{\Gamma(\sum_{j:N_{tj}>0} N_{tj} + \alpha_2)}}.
\end{aligned} \tag{3.47}$$

Moreover, from the above, we see that the posterior distributions of ω_t and λ_t given the augmented data $(\boldsymbol{N}_t, \boldsymbol{U}_t)$ are given by

$$\begin{aligned}
\omega_t \mid \boldsymbol{U}_t &\sim \text{Beta}\left(\sum_{j=1}^{r_t} U_{tj} + \alpha_1, n_t - \sum_{j=1}^{r_t} U_{tj} + \beta_1\right), \\
\lambda_t \mid \boldsymbol{N}_t, \boldsymbol{U}_t &\sim \text{Gamma}\left(\sum_{j:N_{tj}>0} N_{tj} + \alpha_2, \sum_{j=1}^{n_t} v_{tj} - \sum_{j=1}^{r_t} v_{tj} U_{tj} + \beta_2\right).
\end{aligned}$$

The integrated augmented likelihood for the tree $\mathcal{T}$ is thus given by

$$p_{\text{ZIP4}}\left(\boldsymbol{N}, \boldsymbol{U} \mid \boldsymbol{X}, \boldsymbol{v}, \mathcal{T}\right) = \prod_{t=1}^{b} p_{\text{ZIP4}}\left(\boldsymbol{N}_t, \boldsymbol{U}_t \mid \boldsymbol{X}_t, \boldsymbol{v}_t\right). \tag{3.48}$$

Now, we discuss the DIC for this tree which can be derived as a special case of (2.15) with $\boldsymbol{\theta}_t = (\omega_t, \lambda_t)$. To this end, we first focus on DIC_t of terminal node t. It follows that

$$\begin{aligned} D(\bar{\omega}_t, \bar{\lambda}_t) &= -2 \log f_{\text{ZIP4}}(\boldsymbol{N}_t \mid \bar{\omega}_t, \bar{\lambda}_t) \\ &= -2 \sum_{j=1}^{n_t} \log \left(\bar{\omega}_t I_{(N_{tj}=0)} + (1-\bar{\omega}_t) \frac{(\bar{\lambda}_t v_{tj})^{N_{tj}}}{N_{tj}!} e^{-\bar{\lambda}_t v_{tj}} \right), \end{aligned} \tag{3.49}$$

where

$$\bar{\omega}_t = \frac{\sum_{j=1}^{r_t} U_{tj} + \alpha_1}{\sum_{j=1}^{r_t} U_{tj} + \alpha_1 + n_t - \sum_{j=1}^{r_t} U_{tj} + \beta_1} = \frac{\sum_{j=1}^{r_t} U_{tj} + \alpha_1}{\alpha_1 + n_t + \beta_1},$$

$$\bar{\lambda}_t = \frac{\sum_{j:N_{tj}>0} N_{tj} + \alpha_2}{\sum_{j=1}^{n_t} v_{tj} - \sum_{j=1}^{r_t} v_{tj} U_{tj} + \beta_2}.$$

Next, since

$$\begin{aligned} &\log f_{\text{ZIP4}} \left(\boldsymbol{N}_t, \boldsymbol{U}_t \mid \omega_t, \lambda_t \right) \\ &= \left(n_t - \sum_{j=1}^{r_t} U_{tj} \right) \log(1-\omega_t) + \log(\omega_t) \sum_{j=1}^{r_t} U_{tj} - \lambda_t \left(\sum_{j=1}^{n_t} v_{tj} - \sum_{j=1}^{r_t} v_{tj} U_{tj} \right) \\ &\quad + \left[\log(\lambda_t) + N_{tj} \log(v_{tj}) - \log(N_{tj}!) \right] \sum_{j:N_{tj}>0} N_{tj}, \end{aligned}$$

we can derive that

$$\begin{aligned} q_{Dt} &= -2\mathbb{E}_{\text{post}} \left(\log f_{\text{ZIP4}}(\boldsymbol{N}_t, \boldsymbol{U}_t \mid \omega_t, \lambda_t) \right) + 2 \log f_{\text{ZIP4}}(\boldsymbol{N}_t, \boldsymbol{U}_t \mid \bar{\omega}_t, \bar{\lambda}_t) \\ &= 2 \left(\log \left(\frac{\sum_{j=1}^{r_t} U_{tj} + \alpha_1}{\alpha_1 + n_t + \beta_1} \right) - \psi \left(\sum_{j=1}^{r_t} U_{tj} + \alpha_1 \right) \right) \sum_{j=1}^{r_t} U_{tj} + 2 n_t \psi(\alpha_1 + n_t + \beta_1) \\ &\quad + 2 \left(n_t - \sum_{j=1}^{r_t} U_{tj} \right) \left(\log \left(\frac{n_t - \sum_{j=1}^{r_t} U_{tj} + \beta_1}{\alpha_1 + n_t + \beta_1} \right) - \psi \left(n_t - \sum_{j=1}^{r_t} U_{tj} + \beta_1 \right) \right) \\ &\quad + 2 \left(\log \left(\sum_{j:N_{tj}>0} N_{tj} + \alpha_2 \right) - \psi \left(\sum_{j:N_{tj}>0} N_{tj} + \alpha_2 \right) \right) \sum_{j:N_{tj}>0} N_{tj}, \end{aligned} \tag{3.50}$$

where we use the fact that

$$\mathbb{E}_{\text{post}} \left(\log(\omega_t) \right) = \psi \left(\sum_{j=1}^{r_t} U_{tj} + \alpha_1 \right) - \psi(\alpha_1 + n_t + \beta_1),$$

$$\mathbb{E}_{\text{post}} \left(\log(1-\omega_t) \right) = \psi \left(n_t - \sum_{j=1}^{r_t} U_{tj} + \beta_1 \right) - \psi(\alpha_1 + n_t + \beta_1).$$

Table 3.3: Evaluation metrics for ZIP-BCART (ZIP1-ZIP3). $\hat{N}_{tj} = \bar{\mu}_t\bar{\lambda}_t v_{tj}/(1+\bar{\mu}_t)$ for ZIP1; $\hat{N}_{tj} = \bar{\mu}_t\bar{\lambda}_t v_{tj}/(1+\bar{\mu}_t v_{tj})$ for ZIP2; $\hat{N}_{tj} = \bar{\mu}_t\bar{\lambda}_t v_{tj}^2/(1+\bar{\mu}_t v_{tj})$ for ZIP3. ϵ_t denotes the empirical claims frequency in node t, computed as $\sum_{j=1}^{n_t} N_{tj}/\sum_{j=1}^{n_t} v_{tj}$. $\bar{\mu}_t$ and $\bar{\lambda}_t$ are parameter estimations that can be obtained from (3.31) (or (3.38), (3.44)).

	Formulas
RSS($\boldsymbol{N}$)	$\sum_{t=1}^{b}\sum_{j=1}^{n_t}\left(N_{tj}-\hat{N}_{tj}\right)^2$
SE	$\sum_{t=1}^{b}\left(\epsilon_t-\bar{\mu}_t\bar{\lambda}_t/(1+\bar{\mu}_t)\right)^2$
DS	$\sum_{t=1}^{b}\frac{(1+\bar{\mu}_t)^2}{\bar{\mu}_t\bar{\lambda}_t(1+\bar{\mu}_t+\bar{\lambda}_t)}\left(\epsilon_t-\bar{\mu}_t\bar{\lambda}_t/(1+\bar{\mu}_t)\right)^2$

Therefore, $\text{DIC}_t = D\left(\bar{\omega}_t, \bar{\lambda}_t\right) + 2q_{Dt}$ can be directly obtained from (3.49) and (3.50).

For the above four ZIP models, the DIC of the tree $\mathcal{T}$ is obtained by using (2.14). With the formulas derived in the above four subsections for ZIP models, we can use the three-step approach proposed in Section 2.2.4, together with Algorithm 2.3 (see Subsection 2.2.3), to search for an optimal tree which can then be used to predict new data.

As in the previous section, we can obtain the formulas for some of the evaluation metrics based on ZIP1-ZIP3 models in Table 3.3.

Remark 3.5 *(a) As in the data augmentation method proposed following Murray (2021) there should also be three different ways to embed the exposure (Poisson part, zero mass part, or both Poisson and zero mass parts) by using one latent variable. However, as discussed, theoretically, the accuracy of the ZIP4 model is worse than ZIP1-ZIP3 models because of the different data augmentation methods used. This theoretical statement is also verified by our simulation study (not included in this thesis). Therefore, this technique introduced in this subsection will not be explored further in the thesis.*

(b) The performance of the ZIP3 model does not show a significant improvement compared to the ZIP2 model in some initial simulation studies. Therefore, in the following simulation studies and real data analyses, only ZIP1 and ZIP2 models are considered.

3.3.5 Initial Estimators of ZIP Models

When using Algorithm 2.3 (see Subsection 2.2.3) to search for an optimal tree for ZIP models, the initial estimators of parameters μ_t and λ_t in ZIP1-ZIP3 models (or ω_t and λ_t in ZIP4 model) should be obtained for generating latent variables. After some calculations, we find that the MME is not applicable when exposure is included. However, the involvement of exposure is one of the key features in insurance claims frequency analysis. Thus, in this subsection, we will focus on MLE for estimating parameters; see, e.g., Beckett *et al.* (2014). In the following discussion, we omit the subscript t for ZIP models since the same calculations can be done within any certain node. Besides, when exposure is included, there are different ways to model the response variable. Since it is not the main focus here, only the calculation details of the ZIP1 model are provided below; others can be done similarly.

Recall ZIP1 model has the following probability mass function:

$$f_{\text{ZIP1}}(N_j \mid \mu, \lambda) = \begin{cases} \frac{1}{1+\mu} + \frac{\mu}{1+\mu} e^{-\lambda v_j} & N_j = 0 \\ \frac{\mu}{1+\mu} \frac{e^{-\lambda v_j} (\lambda v_j)^{N_j}}{N_j} & N_j = 1, 2, \ldots . \end{cases}$$

Then, the likelihood function is defined as

$$\begin{aligned} L(\mu, \lambda \mid N) =& \prod_{j=1}^{n} f_{\text{ZIP1}}\left(N_j\right) \\ =& \prod_{j:N_j=0} \left(\frac{1}{1+\mu} + \frac{\mu}{1+\mu} e^{-\lambda v_j} \right) \prod_{j:N_j>0} \left(\frac{\mu}{1+\mu} \frac{e^{-\lambda v_j} (\lambda v_j)^{N_j}}{N_j!} \right), \end{aligned}$$

and the log-likelihood function is given as

$$\begin{aligned} l(\mu, \lambda \mid N) =& \sum_{j=1}^{n} \left[I_{(N_j=0)} \log \left(\frac{1}{1+\mu} + \frac{\mu}{1+\mu} e^{-\lambda v_j} \right) \right. \\ & \left. + I_{(N_j>0)} \left(\log \left(\frac{\mu}{1+\mu} \right) - \lambda v_j + N_j \log \left(\lambda v_j \right) - \log \left(N_j! \right) \right) \right]. \end{aligned}$$

By taking the partial derivatives of $l(\mu, \lambda \mid N)$ with respect to μ and λ respectively, the following equations can be obtained:

$$\begin{aligned} \frac{\partial l}{\partial \mu} &= \sum_{j=1}^{n} \left(I_{(N_j=0)} \frac{e^{-\lambda v_j} - 1}{(1+\mu)(1+\mu e^{-\lambda v_j})} + I_{(N_j>0)} \frac{1}{(1+\mu)\mu} \right), \\ \frac{\partial l}{\partial \lambda} &= \sum_{j=1}^{n} \left[I_{(N_j=0)} \frac{-v_j (1+\mu) e^{-\lambda v_j}}{(1+\mu e^{-\lambda v_j})} + I_{(N_j>0)} \left(\frac{N_j}{\lambda} - v_j \right) \right]. \end{aligned}$$

Obviously, we cannot obtain explicit expressions for $\hat{\mu}_{\text{MLE}}$ and $\hat{\lambda}_{\text{MLE}}$ by setting the above two equations equal to zero manually. However, the optimization problem can be easily solved using software R. In our implementation, the package `optimx` is used; see more details in Nash & Grothendieck (2023).

Remark 3.6 *When applying the package* ***`optimx`****, several starting points are used, yielding consistent results that instill confidence in identifying the global maxima. Consequently, the estimates are deemed sensible as a result of the optimization problem.*

3.4 Zero-Inflated NB-Bayesian CART

ZINB models potentially fit better than ZIP models since they can incorporate additional over-dispersion. In this section, we only consider using the data augmentation way proposed in Murray (2021) due to its accuracy (see the discussion in Subsection 3.3.4). Depending on how the exposure is embedded, we discuss four ZINB models. While the treatments (data augmentation and treating the same parameter κ as known) in all four models are similar, the main text includes the calculation processes and results with an emphasis on their differences for the completeness of the content.

3.4.1 Zero-Inflated NB Model 1 (ZINB1)

For terminal node t, we use the following ZINB distribution by adopting NB1 model (exposure included, see Subsection 3.2.1),

$$\begin{aligned} & f_{\text{ZINB1}}\left(N_{tj} \mid \mu_t, \kappa_t, \lambda_t, v_{tj}\right) \\ =& \mathrm{P}\left(N_{tj} \mid \mu_t, \kappa_t, \lambda_t, v_{tj}\right) \\ =& \begin{cases} \frac{1}{1+\mu_t} + \frac{\mu_t}{1+\mu_t} f_{\text{NB1}}(0 \mid \kappa_t, \lambda_t, v_{tj}) & N_{tj} = 0, \\ \frac{\mu_t}{1+\mu_t} f_{\text{NB1}}(m \mid \kappa_t, \lambda_t, v_{tj}) & N_{tj} = 1, 2, \ldots, \end{cases} \end{aligned} \tag{3.51}$$

where $\kappa_t,\ \lambda_t > 0$; $f_{\text{NB1}}(N_{tj} \mid \kappa_t, \lambda_t, v_{tj})$ is given as in (3.9), and $\frac{1}{1+\mu_t} \in (0,1)$ is the probability that a zero is due to the point mass component. In the same way as Subsection 3.2.1, we shall treat both μ_t and λ_t as unknown in the Bayesian framework and treat the parameter κ_t as known which can be estimated upfront by using MLE.

Similar to NB and ZIP models, a data augmentation scheme is needed for the ZINB model. To this end, we introduce three latent variables $\boldsymbol{\xi}_t = (\xi_{t1}, \xi_{t2}, \ldots, \xi_{tn_t}) \in$

$(0,\infty)^{n_t}$, $\boldsymbol{\phi}_t = (\phi_{t1}, \phi_{t2}, \ldots, \phi_{tn_t}) \in (0,\infty)^{n_t}$ and $\boldsymbol{\delta}_t = (\delta_{t1}, \delta_{t2}, \ldots, \delta_{tn_t}) \in \{0,1\}^{n_t}$, and define the data augmented likelihood for the j-th data instance in terminal node t as

$$\begin{aligned} & f_{\text{ZINB1}}\left(N_{tj}, \delta_{tj}, \xi_{tj}, \phi_{tj} \mid \mu_t, \hat{\kappa}_t, \lambda_t\right) \\ & = e^{-\phi_{tj}(1+\mu_t)} \left(\frac{\mu_t \left(\lambda_t v_{tj}\right)^{N_{tj}} \hat{\kappa}_t^{\hat{\kappa}_t}}{\Gamma(\hat{\kappa}_t) N_{tj}!} e^{-\xi_{tj}(\lambda_t v_{tj} + \hat{\kappa}_t)} \xi_{tj}^{\hat{\kappa}_t + N_{tj} - 1} \right)^{\delta_{tj}}, \end{aligned} \tag{3.52}$$

where the support of the function f_{ZINB1} is $\left(\{0\} \times \{0,1\} \times (0,\infty) \times (0,\infty)\right) \cup \left(\mathbb{N} \times \{1\} \times (0,\infty) \times (0,\infty)\right)$. It can be shown that (3.51) is the marginal distribution of the above augmented distribution.

By conditional arguments, we can also check that δ_{tj}, given data $N_{tj} = 0$ and parameters ($\mu_t, \hat{\kappa}_t$, and λ_t), has a Bernoulli distribution, i.e.,

$$\delta_{tj} \mid N_{tj} = 0, \mu_t, \hat{\kappa}_t, \lambda_t \;\sim\; \text{Bern}\left(\frac{\mu_t \left(\frac{\hat{\kappa}_t}{\hat{\kappa}_t + \lambda_t v_{tj}} \right)^{\hat{\kappa}_t}}{1 + \mu_t \left(\frac{\hat{\kappa}_t}{\hat{\kappa}_t + \lambda_t v_{tj}} \right)^{\hat{\kappa}_t}} \right),$$

and $\delta_{tj} = 1$, given $N_{tj} > 0$. And, ξ_{tj}, given data N_{tj} and parameters ($\hat{\kappa}_t$ and λ_t), has a Gamma distribution, i.e.,

$$\xi_{tj} \mid N_{tj}, \hat{\kappa}_t, \lambda_t \;\sim\; \text{Gamma}\left(\hat{\kappa}_t + N_{tj}, \hat{\kappa}_t + \lambda_t\right).$$

Furthermore, ϕ_{tj}, given the parameter μ_t, has an Exponential distribution, i.e.,

$$\phi_{tj} \mid \mu_t \;\sim\; \text{Exp}\left(1 + \mu_t\right).$$

Given the data augmented likelihood in (3.52), the estimated parameter $\hat{\kappa}_t$, and independent conjugate Gamma priors for μ_t and λ_t with hyper-parameters $\alpha_i, \beta_i > 0$, $i = 1, 2$ (cf. (3.2)), we can derive the integrated augmented likelihood for terminal node t as follows

$$\begin{aligned} & p_{\text{ZINB1}}\left(\boldsymbol{N}_t, \boldsymbol{\delta}_t, \boldsymbol{\xi}_t, \boldsymbol{\phi}_t \mid \boldsymbol{X}_t, \boldsymbol{v}_t, \hat{\kappa}_t\right) \\ & = \int_0^\infty \int_0^\infty f_{\text{ZINB1}}\left(\boldsymbol{N}_t, \boldsymbol{\delta}_t, \boldsymbol{\xi}_t, \boldsymbol{\phi}_t \mid \mu_t, \hat{\kappa}_t, \lambda_t\right) p(\mu_t) p(\lambda_t) d\mu_t d\lambda_t \\ & = \int_0^\infty \int_0^\infty \prod_{j=1}^{n_t} e^{-\phi_{tj}(1+\mu_t)} \left(\frac{\mu_t \left(\lambda_t v_{tj}\right)^{N_{tj}} \hat{\kappa}_t^{\hat{\kappa}_t}}{\Gamma(\hat{\kappa}_t) N_{tj}!} e^{-\xi_{tj}(\lambda_t v_{tj} + \hat{\kappa}_t)} \xi_{tj}^{\hat{\kappa}_t + N_{tj} - 1} \right)^{\delta_{tj}} \\ & \quad \times \frac{\beta_1^{\alpha_1} {\mu_t}^{\alpha_1 - 1} e^{-\beta_1 \mu_t}}{\Gamma(\alpha_1)} \frac{\beta_2^{\alpha_2} {\lambda_t}^{\alpha_2 - 1} e^{-\beta_2 \lambda_t}}{\Gamma(\alpha_2)} d\mu_t d\lambda_t \end{aligned}$$

$$
\begin{aligned}
&= \prod_{j=1}^{n_t} e^{-\phi_{tj}} \left(\frac{v_{tj}^{N_{tj}} \hat{\kappa}_t^{\hat{\kappa}_t}}{\Gamma(\hat{\kappa}_t) N_{tj}!} e^{-\xi_{tj}\hat{\kappa}_t} \xi_{tj}^{\hat{\kappa}_t + N_{tj} - 1} \right)^{\delta_{tj}} \int_0^\infty \mu_t^{\sum_{j=1}^{n_t} \delta_{tj} + \alpha_1 - 1} e^{-\left(\sum_{j=1}^{n_t} \phi_{tj} + \beta_1\right)\mu_t} d\mu_t \\
&\quad \times \int_0^\infty \lambda_t^{\sum_{j=1}^{n_t} \delta_{tj} N_{tj} + \alpha_2 - 1} e^{-\left(\sum_{j=1}^{n_t} \delta_{tj}\xi_{tj} v_{tj} + \beta_2\right)\lambda_t} d\lambda_t \times \frac{\beta_1^{\alpha_1}}{\Gamma(\alpha_1)} \frac{\beta_2^{\alpha_2}}{\Gamma(\alpha_2)} \\
&= \frac{\beta_1^{\alpha_1}}{\Gamma(\alpha_1)} \frac{\beta_2^{\alpha_2}}{\Gamma(\alpha_2)} \prod_{j=1}^{n_t} e^{-\phi_{tj}} \left(\frac{v_{tj}^{N_{tj}} \hat{\kappa}_t^{\hat{\kappa}_t}}{\Gamma(\hat{\kappa}_t) N_{tj}!} e^{-\xi_{tj}\hat{\kappa}_t} \xi_{tj}^{\hat{\kappa}_t + N_{tj} - 1} \right)^{\delta_{tj}} \\
&\quad \times \frac{\Gamma\left(\sum_{j=1}^{n_t} \delta_{tj} + \alpha_1\right)}{\left(\sum_{j=1}^{n_t} \phi_{tj} + \beta_1\right)^{\sum_{j=1}^{n_t} \delta_{tj} + \alpha_1}} \frac{\Gamma\left(\sum_{j=1}^{n_t} \delta_{tj} N_{tj} + \alpha_2\right)}{\left(\sum_{j=1}^{n_t} \delta_{tj}\xi_{tj} v_{tj} + \beta_2\right)^{\sum_{j=1}^{n_t} \delta_{tj} N_{tj} + \alpha_2}}. \qquad (3.53)
\end{aligned}
$$

Moreover, from the above, we see that the posterior distributions of μ_t, λ_t given the augmented data $(\boldsymbol{N}_t, \boldsymbol{\delta}_t, \boldsymbol{\xi}_t, \boldsymbol{\phi}_t)$ are given by

$$
\begin{aligned}
\mu_t \mid \boldsymbol{\delta}_t, \boldsymbol{\phi}_t &\sim \text{Gamma}\left(\sum_{j=1}^{n_t} \delta_{tj} + \alpha_1, \sum_{j=1}^{n_t} \phi_{tj} + \beta_1 \right), \\
\lambda_t \mid \boldsymbol{N}_t, \boldsymbol{\delta}_t, \boldsymbol{\xi}_t &\sim \text{Gamma}\left(\sum_{j=1}^{n_t} \delta_{tj} N_{tj} + \alpha_2, \sum_{j=1}^{n_t} \delta_{tj}\xi_{tj} v_{tj} + \beta_2 \right).
\end{aligned}
$$

The integrated augmented likelihood for the tree $\mathcal{T}$ is thus given by

$$
p_{\text{ZINB1}}\left(\boldsymbol{N}, \boldsymbol{\delta}, \boldsymbol{\xi}, \boldsymbol{\phi} \mid \boldsymbol{X}, \boldsymbol{v}, \hat{\boldsymbol{\kappa}}, \mathcal{T}\right) = \prod_{t=1}^{b} p_{\text{ZINB1}}\left(\boldsymbol{N}_t, \boldsymbol{\delta}_t, \boldsymbol{\xi}_t, \boldsymbol{\phi}_t \mid \boldsymbol{X}_t, \boldsymbol{v}_t, \hat{\kappa}_t\right). \qquad (3.54)
$$

Now, we discuss the DIC for this tree which can be derived as a special case of the new DIC proposed in Subsection 3.2.1 with $\boldsymbol{\theta} = (\boldsymbol{\mu}, \boldsymbol{\kappa}, \boldsymbol{\lambda})$. Particularly, $\boldsymbol{\theta_M} = \boldsymbol{\kappa}$ and $\boldsymbol{\theta_B} = (\boldsymbol{\mu}, \boldsymbol{\lambda})$. To this end, we first focus on DIC_t of terminal node t. It follows that

$$
\begin{aligned}
&D(\bar{\mu}_t, \bar{\lambda}_t) \\
&= -2 \log f_{\text{ZINB1}}(\boldsymbol{N}_t \mid \bar{\mu}_t, \hat{\kappa}_t, \bar{\lambda}_t) \\
&= -2 \sum_{j=1}^{n_t} \log \left(\frac{1}{1+\bar{\mu}_t} I_{(N_{tj}=0)} + \frac{\bar{\mu}_t}{1+\bar{\mu}_t} \frac{\Gamma(N_{tj} + \hat{\kappa}_t)}{\Gamma(\hat{\kappa}_t) N_{tj}!} \left(\frac{\hat{\kappa}_t}{\hat{\kappa}_t + \bar{\lambda}_t v_{tj}} \right)^{\hat{\kappa}_t} \left(\frac{\bar{\lambda}_t v_{tj}}{\hat{\kappa}_t + \bar{\lambda}_t v_{tj}} \right)^{N_{tj}} \right), \qquad (3.55)
\end{aligned}
$$

where

$$
\bar{\mu}_t = \frac{\sum_{j=1}^{n_t} \delta_{tj} + \alpha_1}{\sum_{j=1}^{n_t} \phi_{tj} + \beta_1}, \qquad \bar{\lambda}_t = \frac{\sum_{j=1}^{n_t} \delta_{tj} N_{tj} + \alpha_2}{\sum_{j=1}^{n_t} \delta_{tj}\xi_{tj} v_{tj} + \beta_2}. \qquad (3.56)
$$

Next, since

$$
\log f_{\text{ZINB1}}\left(\boldsymbol{N}_t, \boldsymbol{\delta}_t, \boldsymbol{\xi}_t, \boldsymbol{\phi}_t \mid \mu_t, \hat{\kappa}_t, \lambda_t\right)
$$

$$= \sum_{j=1}^{n_t} \Big[-\phi_{tj}(1+\mu_t) + \delta_{tj} \log(\mu_t) + \delta_{tj} N_{tj} \log\left(\lambda_t v_{tj}\right) - \delta_{tj}\xi_{tj}(\lambda_t v_{tj} + \hat{\kappa}_t)$$
$$-\delta_{tj} \log\left(\Gamma(\hat{\kappa}_t)\right) - \delta_{tj} \log\left(N_{tj}!\right) + \hat{\kappa}_t \log(\hat{\kappa}_t) + (\hat{\kappa}_t + N_{tj} - 1)\log(\xi_{tj}) \Big],$$

we can derive that

$$\begin{aligned} r_{Dt} &= 1 - 2\mathbb{E}_{\text{post}}\left(\log f_{\text{ZINB1}}(\boldsymbol{N}_t, \boldsymbol{\delta}_t, \boldsymbol{\xi}_t, \boldsymbol{\phi}_t \mid \mu_t, \hat{\kappa}_t, \lambda_t)\right) \\ &\quad +2\log f_{\text{ZINB1}}\left(\boldsymbol{N}_t, \boldsymbol{\delta}_t, \boldsymbol{\xi}_t, \boldsymbol{\phi}_t \mid \bar{\mu}_t, \hat{\kappa}_t, \bar{\lambda}_t\right) \\ &= 1 + 2\left(\log\left(\sum_{j=1}^{n_t} \delta_{tj} N_{tj} + \alpha_2\right) - \psi\left(\sum_{j=1}^{n_t} \delta_{tj} N_{tj} + \alpha_2\right)\right) \sum_{j=1}^{n_t} \delta_{tj} N_{tj} \\ &\quad +2\left(\log\left(\sum_{j=1}^{n_t} \delta_{tj} + \alpha_1\right) - \psi\left(\sum_{j=1}^{n_t} \delta_{tj} + \alpha_1\right)\right) \sum_{j=1}^{n_t} \delta_{tj}, \end{aligned} \tag{3.57}$$

Therefore, DIC_t can be obtained from (3.55) and (3.57) as

$$\begin{aligned} &\text{DIC}_t \\ &= D\left(\bar{\mu}_t, \bar{\lambda}_t\right) + 2r_{Dt} \\ &= -2\sum_{j=1}^{n_t} \log\left(\frac{1}{1+\bar{\mu}_t} I_{(N_{tj}=0)} + \frac{\bar{\mu}_t}{1+\bar{\mu}_t} \frac{\Gamma(N_{tj}+\hat{\kappa}_t)}{\Gamma(\hat{\kappa}_t) N_{tj}!} \left(\frac{\hat{\kappa}_t}{\hat{\kappa}_t + \bar{\lambda}_t v_{tj}}\right)^{\hat{\kappa}_t} \left(\frac{\bar{\lambda}_t v_{tj}}{\hat{\kappa}_t + \bar{\lambda}_t v_{tj}}\right)^{N_{tj}}\right) \\ &\quad + 2 + 4\left(\log\left(\sum_{j=1}^{n_t} \delta_{ti} + \alpha_1\right) - \psi\left(\sum_{j=1}^{n_t} \delta_{ti} + \alpha_1\right)\right) \sum_{j=1}^{n_t} \delta_{tj} \\ &\quad + 4\left(\log\left(\sum_{j=1}^{n_t} \delta_{tj} N_{tj} + \alpha_2\right) - \psi\left(\sum_{j=1}^{n_t} \delta_{tj} N_{tj} + \alpha_2\right)\right) \sum_{j=1}^{n_t} \delta_{tj} N_{tj}. \end{aligned}$$

Remark 3.7 *(a) It is worth noting that the way of dealing with the parameter κ is different from that in Section 3.2 where MME is used for each node. It turns out that way of dealing with κ is not applicable in ZINB models, and MLE is used instead.*

(b) ZINB2-ZINB4 models discussed in the subsequent subsections use the same treatments for all parameters and introduce the same latent variables. Besides, it can be shown that all ZINB distributions defined at the beginning of each subsection are the marginal distributions of augmented distributions for each corresponding model. Therefore, we will not repeat these details in the following context.

3.4.2 Zero-Inflated NB Model 2 (ZINB2)

For terminal node t, we use the following ZINB distribution by not only adopting the NB1 model (exposure included, see Subsection 3.2.1) but also embedding the exposure into the zero mass part,

$$f_{\text{ZINB2}}\left(N_{tj} \mid \mu_t, \kappa_t, \lambda_t, v_{tj}\right) = \begin{cases} \frac{1}{1+\mu_t v_{tj}} + \frac{\mu_t v_{tj}}{1+\mu_t v_{tj}} f_{\text{NB1}}(0 \mid \kappa_t, \lambda_t, v_{tj}) & N_{tj} = 0, \\ \frac{\mu_t v_{tj}}{1+\mu_t v_{tj}} f_{\text{NB1}}(N_{tj} \mid \kappa_t, \lambda_t, v_{tj}) & N_{tj} = 1, 2, \ldots, \end{cases} \tag{3.58}$$

where κ_t, $\lambda_t > 0$ and $\frac{1}{1+\mu_t v_{tj}} \in (0, 1)$ is the probability that a zero is due to the point mass component.

Then, the data augmented likelihood for the j-th data instance in terminal node t can be defined as

$$\begin{aligned} &f_{\text{ZINB2}}\left(N_{tj}, \delta_{tj}, \xi_{tj}, \phi_{tj} \mid \mu_t, \hat{\kappa}_t, \lambda_t\right) \\ &= e^{-\phi_{tj}(1+\mu_t v_{tj})} \left(\frac{\mu_t v_{tj} \left(\lambda_t v_{tj}\right)^{N_{tj}} \hat{\kappa}_t^{\hat{\kappa}_t}}{\Gamma(\hat{\kappa}_t) N_{tj}!} e^{-\xi_{tj}(\lambda_t v_{tj} + \hat{\kappa}_t)} \xi_{tj}^{\hat{\kappa}_t + N_{tj} - 1} \right)^{\delta_{tj}}. \end{aligned} \tag{3.59}$$

By conditional arguments, we can also check that δ_{tj}, given data $N_{tj} = 0$ and parameters (μ_t, $\hat{\kappa}_t$, and λ_t), has a Bernoulli distribution, i.e.,

$$\delta_{tj} \mid N_{tj} = 0, \mu_t, \hat{\kappa}_t, \lambda_t \;\sim\; \text{Bern}\left(\frac{\mu_t v_{tj} \left(\frac{\hat{\kappa}_t}{\hat{\kappa}_t + \lambda_t v_{tj}}\right)^{\hat{\kappa}_t}}{1 + \mu_t v_{tj} \left(\frac{\hat{\kappa}_t}{\hat{\kappa}_t + \lambda_t v_{tj}}\right)^{\hat{\kappa}_t}} \right),$$

and $\delta_{tj} = 1$, given $N_{tj} > 0$. And, ξ_{tj}, given data N_{tj} and parameters ($\hat{\kappa}_t$ and λ_t), has a Gamma distribution, i.e.,

$$\xi_{tj} \mid N_{tj}, \hat{\kappa}_t, \lambda_t \;\sim\; \text{Gamma}\left(\hat{\kappa}_t + N_{tj}, \hat{\kappa}_t + \lambda_t\right).$$

Furthermore, ϕ_{tj}, given the parameter μ_t, has an Exponential distribution, i.e.,

$$\phi_{tj} \mid \mu_t \;\sim\; \text{Exp}\left(1 + \mu_t v_{tj}\right).$$

Given independent conjugate Gamma priors for μ_t and λ_t with hyper-parameters $\alpha_i, \beta_i > 0$, $i = 1, 2$, and the estimated parameter $\hat{\kappa}_t$, we can derive the integrated augmented likelihood for terminal node t as follows

$$\begin{aligned} &p_{\text{ZINB2}}\left(\boldsymbol{N}_t, \boldsymbol{\delta}_t, \boldsymbol{\xi}_t, \boldsymbol{\phi}_t \mid \boldsymbol{X}_t, \boldsymbol{v}_t, \hat{\kappa}_t\right) \\ &= \int_0^\infty \int_0^\infty f_{\text{ZINB2}}\left(\boldsymbol{N}_t, \boldsymbol{\delta}_t, \boldsymbol{\xi}_t, \boldsymbol{\phi}_t \mid \mu_t, \hat{\kappa}_t, \lambda_t\right) p(\mu_t) p(\lambda_t) d\mu_t d\lambda_t \end{aligned}$$

$$
\begin{aligned}
&= \int_0^\infty \int_0^\infty \prod_{j=1}^{n_t} e^{-\phi_{tj}(1+\mu_t v_{tj})} \left(\frac{\mu_t v_{tj} \left(\lambda_t v_{tj}\right)^{N_{tj}} \hat{\kappa}_t^{\hat{\kappa}_t}}{\Gamma(\hat{\kappa}_t) N_{tj}!} e^{-\xi_{tj}(\lambda_t v_{tj} + \hat{\kappa}_t)} \xi_{tj}^{\hat{\kappa}_t + N_{tj} - 1} \right)^{\delta_{tj}} \qquad (3.60) \\
&\quad \times \frac{\beta_1^{\alpha_1} {\mu_t}^{\alpha_1 - 1} e^{-\beta_1 \mu_t}}{\Gamma(\alpha_1)} \frac{\beta_2^{\alpha_2} {\lambda_t}^{\alpha_2 - 1} e^{-\beta_2 \lambda_t}}{\Gamma(\alpha_2)} d\mu_t d\lambda_t \\
&= \prod_{j=1}^{n_t} e^{-\phi_{tj}} \left(\frac{v_{tj}^{N_{tj}+1} \hat{\kappa}_t^{\hat{\kappa}_t}}{\Gamma(\hat{\kappa}_t) N_{tj}!} e^{-\xi_{tj}\hat{\kappa}_t} \xi_{tj}^{\hat{\kappa}_t + N_{tj} - 1} \right)^{\delta_{tj}} \int_0^\infty \mu_t^{\sum_{j=1}^{n_t} \delta_{tj} + \alpha_1 - 1} e^{-\left(\sum_{j=1}^{n_t} \phi_{tj} v_{tj} + \beta_1\right)\mu_t} d\mu_t \\
&\quad \times \int_0^\infty \lambda_t^{\sum_{j=1}^{n_t} \delta_{tj} N_{tj} + \alpha_2 - 1} e^{-\left(\sum_{j=1}^{n_t} \delta_{tj} \xi_{tj} v_{tj} + \beta_2\right)\lambda_t} d\lambda_t \times \frac{\beta_1^{\alpha_1}}{\Gamma(\alpha_1)} \frac{\beta_2^{\alpha_2}}{\Gamma(\alpha_2)} \\
&= \frac{\beta_1^{\alpha_1}}{\Gamma(\alpha_1)} \frac{\beta_2^{\alpha_2}}{\Gamma(\alpha_2)} \prod_{j=1}^{n_t} e^{-\phi_{tj}} \left(\frac{v_{tj}^{N_{tj}+1} \hat{\kappa}_t^{\hat{\kappa}_t}}{\Gamma(\hat{\kappa}_t) N_{tj}!} e^{-\xi_{tj}\hat{\kappa}_t} \xi_{tj}^{\hat{\kappa}_t + N_{tj} - 1} \right)^{\delta_{tj}} \\
&\quad \times \frac{\Gamma\left(\sum_{j=1}^{n_t} \delta_{tj} + \alpha_1\right)}{\left(\sum_{j=1}^{n_t} \phi_{tj} v_{tj} + \beta_1\right)^{\sum_{j=1}^{n_t} \delta_{tj} + \alpha_1}} \frac{\Gamma\left(\sum_{j=1}^{n_t} \delta_{tj} N_{tj} + \alpha_2\right)}{\left(\sum_{j=1}^{n_t} \delta_{tj} \xi_{tj} v_{tj} + \beta_2\right)^{\sum_{j=1}^{n_t} \delta_{tj} N_{tj} + \alpha_2}}. \qquad (3.61)
\end{aligned}
$$

Moreover, from the above, we see that the posterior distributions of μ_t, λ_t given the augmented data $(\boldsymbol{N}_t, \boldsymbol{\delta}_t, \boldsymbol{\xi}_t, \boldsymbol{\phi}_t)$ are given by

$$
\begin{aligned}
\mu_t \mid \boldsymbol{\delta}_t, \boldsymbol{\phi}_t &\sim \text{Gamma}\left(\sum_{j=1}^{n_t} \delta_{tj} + \alpha_1, \sum_{j=1}^{n_t} \phi_{tj} v_{tj} + \beta_1 \right), \\
\lambda_t \mid \boldsymbol{N}_t, \boldsymbol{\delta}_t, \boldsymbol{\xi}_t &\sim \text{Gamma}\left(\sum_{j=1}^{n_t} \delta_{tj} N_{tj} + \alpha_2, \sum_{j=1}^{n_t} \delta_{tj} \xi_{tj} v_{tj} + \beta_2 \right).
\end{aligned}
$$

The integrated augmented likelihood for the tree $\mathcal{T}$ is thus given by

$$
p_{\text{ZINB2}}\left(\boldsymbol{N}, \boldsymbol{\delta}, \boldsymbol{\xi}, \boldsymbol{\phi} \mid \boldsymbol{X}, \boldsymbol{v}, \hat{\boldsymbol{\kappa}}, \mathcal{T}\right) = \prod_{t=1}^{b} p_{\text{ZINB2}}\left(\boldsymbol{N}_t, \boldsymbol{\delta}_t, \boldsymbol{\xi}_t, \boldsymbol{\phi}_t \mid \boldsymbol{X}_t, \boldsymbol{v}_t, \hat{\kappa}_t\right). \qquad (3.62)
$$

Now, we discuss the DIC_t of terminal node t of this tree. Similarly, we can derive the same expression for r_{Dt} as in (3.57) and we can easily check that

$$
\begin{aligned}
&\text{DIC}_t \\
&= D\left(\bar{\mu}_t, \bar{\lambda}_t\right) + 2 r_{Dt} \\
&= -2 \sum_{j=1}^{n_t} \log \left(\frac{1}{1 + \bar{\mu}_t v_{tj}} I_{(N_{tj}=0)} + \frac{\bar{\mu}_t v_{tj}}{1 + \bar{\mu}_t v_{tj}} \frac{\Gamma(N_{tj} + \hat{\kappa}_t)}{\Gamma(\hat{\kappa}_t) N_{tj}!} \left(\frac{\hat{\kappa}_t}{\hat{\kappa}_t + \bar{\lambda}_t v_{tj}} \right)^{\hat{\kappa}_t} \left(\frac{\bar{\lambda}_t v_{tj}}{\hat{\kappa}_t + \bar{\lambda}_t v_{tj}} \right)^{N_{tj}} \right) \\
&\quad + 2 + 4 \left(\log\left(\sum_{j=1}^{n_t} \delta_{tj} + \alpha_1 \right) - \psi\left(\sum_{j=1}^{n_t} \delta_{tj} + \alpha_1 \right) \right) \sum_{j=1}^{n_t} \delta_{tj}
\end{aligned}
$$

$$+4\left(\log\left(\sum_{j=1}^{n_t}\delta_{tj}N_{tj}+\alpha_2\right)-\psi\left(\sum_{j=1}^{n_t}\delta_{tj}N_{tj}+\alpha_2\right)\right)\sum_{j=1}^{n_t}\delta_{tj}N_{tj},$$

where

$$\bar{\mu}_t=\frac{\sum_{j=1}^{n_t}\delta_{tj}+\alpha_1}{\sum_{j=1}^{n_t}\phi_{tj}v_{tj}+\beta_1},\qquad \bar{\lambda}_t=\frac{\sum_{j=1}^{n_t}\delta_{tj}N_{tj}+\alpha_2}{\sum_{j=1}^{n_t}\delta_{tj}\xi_{tj}v_{tj}+\beta_2}. \tag{3.63}$$

3.4.3 Zero-Inflated NB Model 3 (ZINB3)

For terminal node t, we use the following ZINB distribution by adopting NB2 model (exposure included, see Subsection 3.2.2),

$$f_{\text{ZINB3}}\left(N_{tj}\mid\mu_t,\kappa_t,\lambda_t,v_{tj}\right)=\begin{cases}\frac{1}{1+\mu_t}+\frac{\mu_t}{1+\mu_t}f_{\text{NB2}}(0\mid\kappa_t,\lambda_t,v_{tj}) & N_{tj}=0,\\ \frac{\mu_t}{1+\mu_t}f_{\text{NB2}}(N_{tj}\mid\kappa_t,\lambda_t,v_{tj}) & N_{tj}=1,2,\ldots,\end{cases} \tag{3.64}$$

where $\kappa_t,\ \lambda_t>0$; $f_{\text{NB2}}(N_{tj}\mid\kappa_t,\lambda_t,v_{tj})$ is given as in (3.20), and $\frac{1}{1+\mu_t}\in(0,1)$ is the probability that a zero is due to the point mass component.

As before, the data augmented likelihood for the j-th data instance in terminal node t can be defined as

$$\begin{aligned}&f_{\text{ZINB3}}\left(N_{tj},\delta_{tj},\xi_{tj},\phi_{tj}\mid\mu_t,\hat{\kappa}_t,\lambda_t\right)\\&=e^{-\phi_{tj}(1+\mu_t)}\left(\frac{\mu_t\left(\lambda_t v_{tj}\right)^{N_{tj}}\left(\hat{\kappa}_t v_{tj}\right)^{\hat{\kappa}_t v_{tj}}}{\Gamma(\hat{\kappa}_t v_{tj})N_{tj}!}e^{-\xi_{tj}v_{tj}(\lambda_t+\hat{\kappa}_t)}\xi_{tj}^{\hat{\kappa}_t v_{tj}+N_{tj}-1}\right)^{\delta_{tj}}.\end{aligned} \tag{3.65}$$

By conditional arguments, we can also check that δ_{tj}, given data $N_{tj}=0$ and parameters ($\mu_t,\hat{\kappa}_t$, and λ_t), has a Bernoulli distribution, i.e.,

$$\delta_{tj}\mid N_{tj}=0,\mu_t,\hat{\kappa}_t,\lambda_t\ \sim\ \text{Bern}\left(\frac{\mu_t\left(\frac{\hat{\kappa}_t}{\hat{\kappa}_t+\lambda_t}\right)^{\hat{\kappa}_t v_{tj}}}{1+\mu_t\left(\frac{\hat{\kappa}_t}{\hat{\kappa}_t+\lambda_t}\right)^{\hat{\kappa}_t v_{tj}}}\right),$$

and $\delta_{tj}=1$, given $N_{tj}>0$. And, ξ_{tj}, given data N_{tj} and parameters ($\hat{\kappa}_t$ and λ_t), has a Gamma distribution, i.e.,

$$\xi_{tj}\mid N_{tj},\hat{\kappa}_t,\lambda_t\ \sim\ \text{Gamma}\left(\hat{\kappa}_t v_{tj}+N_{tj},\hat{\kappa}_t v_{tj}+\lambda_t v_{tj}\right).$$

Furthermore, ϕ_{tj}, given the parameter μ_t, has an Exponential distribution, i.e.,

$$\phi_{tj}\mid\mu_t\ \sim\ \text{Exp}\left(1+\mu_t\right).$$

Given independent conjugate Gamma priors for μ_t and λ_t with hyper-parameters $\alpha_i, \beta_i > 0$, $i = 1, 2$, and the estimated parameter $\hat{\kappa}_t$, we can derive the integrated augmented likelihood for terminal node t as follows

$$
\begin{aligned}
&p_{\text{ZINB3}}\left(\boldsymbol{N}_t, \boldsymbol{\delta}_t, \boldsymbol{\xi}_t, \boldsymbol{\phi}_t \mid \boldsymbol{X}_t, \boldsymbol{v}_t, \hat{\kappa}_t\right) \\
&= \int_0^\infty \int_0^\infty f_{\text{ZINB3}}\left(\boldsymbol{N}_t, \boldsymbol{\delta}_t, \boldsymbol{\xi}_t, \boldsymbol{\phi}_t \mid \mu_t, \hat{\kappa}_t, \lambda_t\right) p(\mu_t) p(\lambda_t) d\mu_t d\lambda_t \\
&= \int_0^\infty \int_0^\infty \prod_{j=1}^{n_t} e^{-\phi_{tj}(1+\mu_t)} \left(\frac{\mu_t \left(\lambda_t v_{tj}\right)^{N_{tj}} \left(\hat{\kappa}_t v_{tj}\right)^{\hat{\kappa}_t v_{tj}}}{\Gamma(\hat{\kappa}_t v_{tj}) N_{tj}!} e^{-\xi_{tj} v_{tj}(\lambda_t + \hat{\kappa}_t)} \xi_{tj}^{\hat{\kappa}_t v_{tj} + N_{tj} - 1} \right)^{\delta_{tj}} \\
&\quad \times \frac{\beta_1^{\alpha_1} \mu_t^{\alpha_1 - 1} e^{-\beta_1 \mu_t}}{\Gamma(\alpha_1)} \frac{\beta_2^{\alpha_2} \lambda_t^{\alpha_2 - 1} e^{-\beta_2 \lambda_t}}{\Gamma(\alpha_2)} d\mu_t d\lambda_t \\
&= \frac{\beta_1^{\alpha_1}}{\Gamma(\alpha_1)} \frac{\beta_2^{\alpha_2}}{\Gamma(\alpha_2)} \prod_{j=1}^{n_t} e^{-\phi_{tj}} \left(\frac{v_{tj}^{N_{tj}} \left(\hat{\kappa}_t v_{tj}\right)^{\hat{\kappa}_t v_{tj}}}{\Gamma(\hat{\kappa}_t v_{tj}) N_{tj}!} e^{-\xi_{tj} v_{tj} \hat{\kappa}_t} \xi_{tj}^{\hat{\kappa}_t v_{tj} + N_{tj} - 1} \right)^{\delta_{tj}} \\
&\quad \times \int_0^\infty \mu_t^{\sum_{j=1}^{n_t} \delta_{tj} + \alpha_1 - 1} e^{-\left(\sum_{j=1}^{n_t} \phi_{tj} + \beta_1\right)\mu_t} d\mu_t \\
&\quad \times \int_0^\infty \lambda_t^{\sum_{j=1}^{n_t} \delta_{tj} N_{tj} + \alpha_2 - 1} e^{-\left(\sum_{j=1}^{n_t} \delta_{tj} \xi_{tj} v_{tj} + \beta_2\right)\lambda_t} d\lambda_t \\
&= \frac{\beta_1^{\alpha_1}}{\Gamma(\alpha_1)} \frac{\beta_2^{\alpha_2}}{\Gamma(\alpha_2)} \prod_{j=1}^{n_t} e^{-\phi_{tj}} \left(\frac{v_{tj}^{N_{tj}} \left(\hat{\kappa}_t v_{tj}\right)^{\hat{\kappa}_t v_{tj}}}{\Gamma(\hat{\kappa}_t v_{tj}) N_{tj}!} e^{-\xi_{tj} v_{tj} \hat{\kappa}_t} \xi_{tj}^{\hat{\kappa}_t v_{tj} + N_{tj} - 1} \right)^{\delta_{tj}} \\
&\quad \times \frac{\Gamma\left(\sum_{j=1}^{n_t} \delta_{tj} + \alpha_1\right)}{\left(\sum_{j=1}^{n_t} \phi_{tj} + \beta_1\right)^{\sum_{j=1}^{n_t} \delta_{tj} + \alpha_1}} \frac{\Gamma\left(\sum_{j=1}^{n_t} \delta_{tj} N_{tj} + \alpha_2\right)}{\left(\sum_{j=1}^{n_t} \delta_{tj} \xi_{tj} v_{tj} + \beta_2\right)^{\sum_{j=1}^{n_t} \delta_{tj} N_{tj} + \alpha_2}}.
\end{aligned} \tag{3.66}
$$

Moreover, from the above, we see that the posterior distributions of μ_t, λ_t given the augmented data $(\boldsymbol{N}_t, \boldsymbol{\delta}_t, \boldsymbol{\xi}_t, \boldsymbol{\phi}_t)$ are given by

$$
\begin{aligned}
\mu_t \mid \boldsymbol{\delta}_t, \boldsymbol{\phi}_t &\sim \text{Gamma}\left(\sum_{j=1}^{n_t} \delta_{tj} + \alpha_1, \sum_{j=1}^{n_t} \phi_{tj} + \beta_1\right), \\
\lambda_t \mid \boldsymbol{N}_t, \boldsymbol{\delta}_t, \boldsymbol{\xi}_t &\sim \text{Gamma}\left(\sum_{j=1}^{n_t} \delta_{tj} N_{tj} + \alpha_2, \sum_{j=1}^{n_t} \delta_{tj} \xi_{tj} v_{tj} + \beta_2\right).
\end{aligned}
$$

The integrated augmented likelihood for the tree $\mathcal{T}$ is thus given by

$$
p_{\text{ZINB3}}\left(\boldsymbol{N}, \boldsymbol{\delta}, \boldsymbol{\xi}, \boldsymbol{\phi} \mid \boldsymbol{X}, \boldsymbol{v}, \hat{\boldsymbol{\kappa}}, \mathcal{T}\right) = \prod_{t=1}^{b} p_{\text{ZINB3}}\left(\boldsymbol{N}_t, \boldsymbol{\delta}_t, \boldsymbol{\xi}_t, \boldsymbol{\phi}_t \mid \boldsymbol{X}_t, \boldsymbol{v}_t, \hat{\kappa}_t\right). \tag{3.67}
$$

Now, we discuss the DIC_t of terminal node t of this tree. Similarly, we can

derive the same expression for r_{Dt} as in (3.57) and we can easily check that

$$
\begin{aligned}
&\mathrm{DIC}_t \\
&= D(\bar{\mu}_t, \bar{\lambda}_t) + 2r_{Dt} \\
&= -2\sum_{j=1}^{n_t} \log\left(\frac{1}{1+\bar{\mu}_t} I_{(N_{tj}=0)} + \frac{\bar{\mu}_t}{1+\bar{\mu}_t} \frac{\Gamma(N_{tj}+\hat{\kappa}_t v_{tj})}{\Gamma(\hat{\kappa}_t v_{tj}) N_{tj}!} \left(\frac{\hat{\kappa}_t}{\hat{\kappa}_t+\bar{\lambda}_t}\right)^{\hat{\kappa}_t v_{tj}} \left(\frac{\bar{\lambda}_t}{\hat{\kappa}_t+\bar{\lambda}_t}\right)^{N_{tj}} \right) \\
&\quad + 2 + 4\left(\log\left(\sum_{j=1}^{n_t} \delta_{tj} + \alpha_1 \right) - \psi\left(\sum_{j=1}^{n_t} \delta_{tj} + \alpha_1 \right) \right) \sum_{j=1}^{n_t} \delta_{tj} \\
&\quad + 4\left(\log\left(\sum_{j=1}^{n_t} \delta_{tj} N_{tj} + \alpha_2 \right) - \psi\left(\sum_{j=1}^{n_t} \delta_{tj} N_{tj} + \alpha_2 \right) \right) \sum_{j=1}^{n_t} \delta_{tj} N_{tj},
\end{aligned}
$$

where $\bar{\mu}_t$ and $\bar{\lambda}_t$ are the same as in (3.56).

3.4.4 Zero-Inflated NB Model 4 (ZINB4)

For terminal node t, we use the following ZINB distribution by not only adopting the NB2 model (exposure included, see Subsection 3.2.2) but also embedding the exposure into the zero mass part,

$$
f_{\mathrm{ZINB4}}\left(N_{tj} \mid \mu_t, \kappa_t, \lambda_t, v_{tj}\right) = \begin{cases} \frac{1}{1+\mu_t v_{tj}} + \frac{\mu_t v_{tj}}{1+\mu_t v_{tj}} f_{\mathrm{NB2}}(0 \mid \kappa_t, \lambda_t, v_{tj}) & N_{tj} = 0, \\ \frac{\mu_t v_{tj}}{1+\mu_t v_{tj}} f_{\mathrm{NB2}}(N_{tj} \mid \kappa_t, \lambda_t, v_{tj}) & N_{tj} = 1, 2, \ldots, \end{cases} \tag{3.68}
$$

where κ_t, $\lambda_t > 0$ and $\frac{1}{1+\mu_t} \in (0,1)$ is the probability that a zero is due to the point mass component.

As before, the data augmented likelihood for the j-th data instance in terminal node t can be defined as

$$
\begin{aligned}
&f_{\mathrm{ZINB4}}\left(N_{tj}, \delta_{tj}, \xi_{tj}, \phi_{tj} \mid \mu_t, \hat{\kappa}_t, \lambda_t\right) \\
&= e^{-\phi_{tj}(1+\mu_t v_{tj})} \left(\frac{\mu_t v_{tj} \left(\lambda_t v_{tj}\right)^{N_{tj}} \left(\hat{\kappa}_t v_{tj}\right)^{\hat{\kappa}_t v_{tj}}}{\Gamma(\hat{\kappa}_t v_{tj}) N_{tj}!} e^{-\xi_{tj} v_{tj}(\lambda_t+\hat{\kappa}_t)} \xi_{tj}^{\hat{\kappa}_t v_{tj}+N_{tj}-1} \right)^{\delta_{tj}}.
\end{aligned} \tag{3.69}
$$

By conditional arguments, we can also check that δ_{tj}, given data $N_{tj} = 0$ and parameters ($\mu_t, \hat{\kappa}_t$, and λ_t), has a Bernoulli distribution, i.e.,

$$
\delta_{tj} \mid N_{tj} = 0, \mu_t, \hat{\kappa}_t, \lambda_t \;\sim\; \mathrm{Bern}\left(\frac{\mu_t v_{tj} \left(\frac{\hat{\kappa}_t}{\hat{\kappa}_t+\lambda_t}\right)^{\hat{\kappa}_t v_{tj}}}{1 + \mu_t v_{tj} \left(\frac{\hat{\kappa}_t}{\hat{\kappa}_t+\lambda_t}\right)^{\hat{\kappa}_t v_{tj}}} \right),
$$

and $\delta_{tj} = 1$, given $N_{tj} > 0$. And, ξ_{tj}, given data N_{tj} and parameters ($\hat{\kappa}_t$ and λ_t), has a Gamma distribution, i.e.,

$$\xi_{tj} \mid N_{tj}, \hat{\kappa}_t, \lambda_t \ \sim \ \text{Gamma}\left(\hat{\kappa}_t v_{tj} + N_{tj}, \hat{\kappa}_t v_{tj} + \lambda_t v_{tj}\right).$$

Furthermore, ϕ_{tj}, given the parameter μ_t, has an Exponential distribution, i.e.,

$$\phi_{tj} \mid \mu_t \ \sim \ \text{Exp}\left(1 + \mu_t v_{tj}\right).$$

Given independent conjugate Gamma priors for μ_t and λ_t with hyper-parameters $\alpha_i, \beta_i > 0$, $i = 1, 2$, and the estimated parameter $\hat{\kappa}_t$, we can derive the integrated augmented likelihood for terminal node t as follows

$$\begin{aligned}
&p_{\text{ZINB4}}\left(\boldsymbol{N}_t, \boldsymbol{\delta}_t, \boldsymbol{\xi}_t, \boldsymbol{\phi}_t \mid \boldsymbol{X}_t, \boldsymbol{v}_t, \hat{\kappa}_t\right) \\
&= \int_0^\infty \int_0^\infty f_{\text{ZINB4}}\left(\boldsymbol{N}_t, \boldsymbol{\delta}_t, \boldsymbol{\xi}_t, \boldsymbol{\phi}_t \mid \mu_t, \hat{\kappa}_t, \lambda_t\right) p(\mu_t) p(\lambda_t) d\mu_t d\lambda_t \\
&= \int_0^\infty \int_0^\infty \prod_{j=1}^{n_t} e^{-\phi_{tj}(1+\mu_t v_{tj})} \left(\frac{\mu_t v_{tj} \left(\lambda_t v_{tj}\right)^{N_{tj}} \left(\hat{\kappa}_t v_{tj}\right)^{\hat{\kappa}_t v_{tj}}}{\Gamma(\hat{\kappa}_t v_{tj}) N_{tj}!} e^{-\xi_{tj} v_{tj}(\lambda_t + \hat{\kappa}_t)} \xi_{tj}^{\hat{\kappa}_t v_{tj} + N_{tj} - 1} \right)^{\delta_{tj}} \\
&\quad \times \frac{\beta_1^{\alpha_1} {\mu_t}^{\alpha_1 - 1} e^{-\beta_1 \mu_t}}{\Gamma(\alpha_1)} \frac{\beta_2^{\alpha_2} {\lambda_t}^{\alpha_2 - 1} e^{-\beta_2 \lambda_t}}{\Gamma(\alpha_2)} d\mu_t d\lambda_t \\
&= \frac{\beta_1^{\alpha_1}}{\Gamma(\alpha_1)} \frac{\beta_2^{\alpha_2}}{\Gamma(\alpha_2)} \prod_{j=1}^{n_t} e^{-\phi_{tj}} \left(\frac{v_{tj}^{N_{tj}+1} (\hat{\kappa}_t v_{tj})^{\hat{\kappa}_t v_{tj}}}{\Gamma(\hat{\kappa}_t v_{tj}) N_{tj}!} e^{-\xi_{tj} v_{tj} \hat{\kappa}_t} \xi_{tj}^{\hat{\kappa}_t v_{tj} + N_{tj} - 1} \right)^{\delta_{tj}} \\
&\quad \times \int_0^\infty \mu_t^{\sum_{j=1}^{n_t} \delta_{tj} + \alpha_1 - 1} e^{-\left(\sum_{j=1}^{n_t} \phi_{tj} v_{tj} + \beta_1\right)\mu_t} d\mu_t \\
&\quad \times \int_0^\infty \lambda_t^{\sum_{j=1}^{n_t} \delta_{tj} N_{tj} + \alpha_2 - 1} e^{-\left(\sum_{j=1}^{n_t} \delta_{tj} \xi_{tj} v_{tj} + \beta_2\right)\lambda_t} d\lambda_t \\
&= \frac{\beta_1^{\alpha_1}}{\Gamma(\alpha_1)} \frac{\beta_2^{\alpha_2}}{\Gamma(\alpha_2)} \prod_{j=1}^{n_t} e^{-\phi_{tj}} \left(\frac{v_{tj}^{N_{tj}+1} (\hat{\kappa}_t v_{tj})^{\hat{\kappa}_t v_{tj}}}{\Gamma(\hat{\kappa}_t v_{tj}) N_{tj}!} e^{-\xi_{tj} v_{tj} \hat{\kappa}_t} \xi_{tj}^{\hat{\kappa}_t v_{tj} + N_{tj} - 1} \right)^{\delta_{tj}} \\
&\quad \times \frac{\Gamma\left(\sum_{j=1}^{n_t} \delta_{tj} + \alpha_1\right)}{\left(\sum_{j=1}^{n_t} \phi_{tj} v_{tj} + \beta_1\right)^{\sum_{j=1}^{n_t} \delta_{tj} + \alpha_1}} \frac{\Gamma\left(\sum_{j=1}^{n_t} \delta_{tj} N_{tj} + \alpha_2\right)}{\left(\sum_{j=1}^{n_t} \delta_{tj} \xi_{tj} v_{tj} + \beta_2\right)^{\sum_{j=1}^{n_t} \delta_{tj} N_{tj} + \alpha_2}}.
\end{aligned} \tag{3.70}$$

Moreover, from the above, we see that the posterior distributions of μ_t, λ_t given the augmented data $(\boldsymbol{N}_t, \boldsymbol{\delta}_t, \boldsymbol{\xi}_t, \boldsymbol{\phi}_t)$ are given by

$$\begin{aligned}
\mu_t \mid \boldsymbol{\delta}_t, \boldsymbol{\phi}_t \ &\sim \ \text{Gamma}\left(\sum_{j=1}^{n_t} \delta_{tj} + \alpha_1, \sum_{j=1}^{n_t} \phi_{tj} v_{tj} + \beta_1\right), \\
\lambda_t \mid \boldsymbol{N}_t, \boldsymbol{\delta}_t, \boldsymbol{\xi}_t \ &\sim \ \text{Gamma}\left(\sum_{j=1}^{n_t} \delta_{tj} N_{tj} + \alpha_2, \sum_{j=1}^{n_t} \delta_{tj} \xi_{tj} v_{tj} + \beta_2\right).
\end{aligned}$$

The integrated augmented likelihood for the tree $\mathcal{T}$ is thus given by

$$p_{\text{ZINB4}}\left(\boldsymbol{N}, \boldsymbol{\delta}, \boldsymbol{\xi}, \boldsymbol{\phi} \mid \boldsymbol{X}, \boldsymbol{v}, \hat{\boldsymbol{\kappa}}, \mathcal{T}\right) = \prod_{t=1}^{b} p_{\text{ZINB4}}\left(\boldsymbol{N}_t, \boldsymbol{\delta}_t, \boldsymbol{\xi}_t, \boldsymbol{\phi}_t \mid \boldsymbol{X}_t, \boldsymbol{v}_t, \hat{\kappa}_t\right). \quad (3.71)$$

Now, we discuss the DIC_t of terminal node t of this tree. Similarly, we can derive the same expression for r_{Dt} as in (3.57) and we can easily check that

$$\begin{aligned}
&\text{DIC}_t \\
&= D(\bar{\mu}_t, \bar{\lambda}_t) + 2r_{Dt} \\
&= -2\sum_{j=1}^{n_t} \log\left(\frac{1}{1+\bar{\mu}_t v_{tj}} I_{(N_{tj}=0)} + \frac{\bar{\mu}_t v_{tj}}{1+\bar{\mu}_t v_{tj}} \frac{\Gamma(N_{tj}+\hat{\kappa}_t v_{tj})}{\Gamma(\hat{\kappa}_t v_{tj}) N_{tj}!} \left(\frac{\hat{\kappa}_t}{\hat{\kappa}_t+\bar{\lambda}_t}\right)^{\hat{\kappa}_t v_{tj}} \left(\frac{\bar{\lambda}_t}{\hat{\kappa}_t+\bar{\lambda}_t}\right)^{N_{tj}}\right) \\
&\quad +2 + 4\left(\log\left(\sum_{j=1}^{n_t}\delta_{tj}+\alpha_1\right) - \psi\left(\sum_{j=1}^{n_t}\delta_{tj}+\alpha_1\right)\right)\sum_{j=1}^{n_t}\delta_{tj} \\
&\quad +4\left(\log\left(\sum_{j=1}^{n_t}\delta_{tj}N_{tj}+\alpha_2\right) - \psi\left(\sum_{j=1}^{n_t}\delta_{tj}N_{tj}+\alpha_2\right)\right)\sum_{j=1}^{n_t}\delta_{tj}N_{tj},
\end{aligned}$$

where $\bar{\mu}_t$ and $\bar{\lambda}_t$ are the same as in (3.63).

For the above four ZINB models, the DIC of the tree $\mathcal{T}$ is obtained by using (2.14). With the above formulas derived in the four subsections for ZINB models, we can use the three-step approach proposed in Section 2.2.4, together with Algorithm 3.1 in Subsection 3.2.2 (treat $\boldsymbol{\theta_M} = \boldsymbol{\kappa}$, $\boldsymbol{\theta_B} = (\boldsymbol{\mu}, \boldsymbol{\lambda})$, and $\boldsymbol{z} = (\boldsymbol{\delta}, \boldsymbol{\xi}, \boldsymbol{\phi})$), to search for an optimal tree which can then be used to predict new data.

As in the previous section, we can obtain the formulas for some of the evaluation metrics based on ZINB distributions in Table 3.4.

Remark 3.8 *In ZINB models, three latent variables need to be employed, which decreases the computational efficiency significantly. Furthermore, when compared to ZIP models, the performance of ZINB models does not improve significantly in some initial simulation studies. Therefore, in the following simulation and real data analyses, we will not include the ZINB models.*

3.5 Simulation Studies

In this section, we illustrate the efficiency of the BCART models for claims frequency introduced in previous sections by using simulated data. In the sequel, we use the abbreviation P-CART to denote CART for the Poisson model, and the

Table 3.4: Evaluation metrics for ZINB-BCART. $\hat{N}_{tj} = \bar{\mu}_t\bar{\lambda}_t v_{tj}/(1+\bar{\mu}_t)$ for ZINB1 and ZINB3; $\hat{N}_{tj} = \bar{\mu}_t\bar{\lambda}_t v_{tj}^2/(1+\bar{\mu}_t v_{tj})$ for ZINB2 and ZINB4. ϵ_t denotes the empirical claims frequency in node t, computed as $\sum_{j=1}^{n_t} N_{tj}/\sum_{j=1}^{n_t} v_{tj}$. $\bar{\mu}_t$ and $\bar{\lambda}_t$ are parameter estimations that can be obtained from (3.56) (or (3.63)); $\hat{\kappa}_t$ is obtained by using MLE.

	Formulas
RSS($\boldsymbol{N}$)	$\sum_{t=1}^{b}\sum_{j=1}^{n_t}\left(N_{tj}-\hat{N}_{tj}\right)^2$
SE	$\sum_{t=1}^{b}\left(\epsilon_t-\bar{\mu}_t\bar{\lambda}_t/(1+\bar{\mu}_t)\right)^2$
DS	$\sum_{t=1}^{b}\frac{(1+\bar{\mu}_t)^2\hat{\kappa}_t}{\bar{\mu}_t\bar{\lambda}_t\left((\hat{\kappa}_t+\bar{\lambda}_t)(1+\bar{\mu}_t)+\bar{\lambda}_t\hat{\kappa}_t\right)}\left(\epsilon_t-\bar{\mu}_t\bar{\lambda}_t/(1+\bar{\mu}_t)\right)^2$

other abbreviations can be similarly understood (e.g., NB1-BCART denotes the BCART for the NB1 model). We will discuss three simulation examples below. Subsection 3.5.1 aims to illustrate that BCART models can do really well for the chessboard data similar to Figure 1.1 for which CART cannot reasonably do anything. In addition, from this simulation study, we also see that BCART models can do well with variable selection. In Subsection 3.5.2, we shall examine how different BCART models can capture the data over-dispersion. In Subsection 3.5.3, we illustrate the performance of ZIP-BCART models for data with exposures.

3.5.1 Poisson Data with Noise Variables

We simulate a data set $\{(\boldsymbol{x}_i, v_i, N_i)\}_{i=1}^n$ with $n = 5,000$ independent observations. Here $v_i \sim \mathrm{U}(0,1)$, $\boldsymbol{x}_i = (x_{i1},\cdots,x_{i8})$, with independent components $x_{i1} \sim \mathrm{U}\{-3,-2,-1,1,2,3\}$, $x_{i2} \sim \mathrm{N}(0,1)$, $x_{ik} \sim \mathrm{U}(-1,1)$ for $k = 3,4$, $x_{ik} \sim \mathrm{N}(0,1)$ for $k = 5,6$, and $x_{ik} \sim \mathrm{U}\{-3,-2,-1,1,2,3\}$ for $k = 7,8$. Moreover, $N_i \sim \mathrm{Poi}(\lambda(x_{i1},x_{i2})\, v_i)$, where

$$\lambda(x_1,x_2) = \begin{cases} 1 & \text{if } x_1x_2 \le 0, \\ 7 & \text{if } x_1x_2 > 0. \end{cases}$$

Obviously, the designed noise variables $x_{ik}, k = 3,\ldots,8$ are all independent of the response $\boldsymbol{N}$. We use P-BCART and P-CART for the above simulated data, where $x_{ik}, k = 1,7,8$ are treated as categorical. We have included both categorical and

continuous variables as noise variables and as significant variables, which is a bit more general than the data shown in Figure 1.1. Note that the same conclusion can be drawn for numeric $x_{ik}, k = 1, 7, 8$, but to better illustrate the effectiveness of the P-BCART we choose to make them as characters (to increase the splitting possibilities of these variables).

We first apply P-CART as implemented in R package `rpart`; see, e.g., Therneau & Atkinson (2023). It is not surprising that P-CART is not able to give us any reasonable tree that can characterize the data, due to its greedy search nature. The smallest tree (except the one with only a root node) that P-CART generated has 25 terminal nodes and the tree found by using cross-validation has 31 terminal nodes. Obviously, both of them are much more complicated than the real model. Furthermore, in these two trees, all the noise variables are used, which indicates that P-CART is sensitive to noise.

Now we discuss the P-BCART applied to the data focusing preliminary on the effect of noise variables to the model. We simply set equal probabilities, i.e., P(Grow)=P(Prune)=P(Change1)=P(Change2)=P(Swap)= 0.2, for the tree proposals. For the Gamma prior of the Poisson intensities λ_t we use $\alpha = 3.2096$ and $\beta = 0.8$ which are selected by keeping the relationship $\alpha/\beta = \sum_{i=1}^{n} N_i / \sum_{i=1}^{n} v_i$. It is worth mentioning that the performance of the algorithm does not change much when choosing different pairs of (α, β) while keeping their ratio. We also observe the same in other simulation examples, so in the following, we will not dwell on their selection.

In Table 3.5 we list the tuned hyper-parameters γ, ρ in the first two columns for which the MCMC algorithms will converge to a region of trees with a certain number of terminal nodes listed (see Step 1 of Table 2.1). For each fixed hyper-parameter γ and ρ, we run 10,000 iterations in the MCMC algorithm and take results after an initial burn-in period of 2,000 iterations, after which the posterior probabilities of the tree structures have been settled for some time. This procedure is done with 3 restarts. The fourth column gives the total number of accepted trees after the burn-in period in the MCMC algorithms. The last columns of Table 3.5 include the total number of times each variable is used in the accepted trees. We see from these columns that all noise variables have a very low selection rate and as expected, the significant variables x_1, x_2 are dominating. Besides, at first glance, it is inferred that the noise variables x_3 and x_4 have a much lower selection rate than the other noise variables which is just because x_3 and x_4 are simulated using a distribution completely different from those of the significant variables. However, when the experiment is run 10 times, we find that the average selection rates of all

Table 3.5: Total count of each variable used amongst all accepted trees from the P-BCART MCMC algorithms (after the burn-in period; equal probabilities for tree moves; one run with 3 restarts).

γ	ρ	# terminal nodes	# accepted trees	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8
0.50	20	2	342	197	156	1	0	2	5	4	4
0.95	17	3	460	408	381	1	4	8	6	8	5
0.99	15	4	800	1261	1239	12	17	30	25	31	20
0.99	12	5	652	1157	1126	9	7	18	15	16	20
0.99	10	6	305	710	680	13	4	18	25	30	12
0.99	6	7	318	825	809	3	8	15	23	14	9
0.99	5	8	210	681	647	2	10	7	13	18	5

Table 3.6: Average frequency of each variable used in all accepted trees from the P-BCART MCMC algorithms (after the burn-in period; equal probabilities for tree moves; ten runs with 3 restarts using the same simulated data).

# terminal nodes	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8
2	183	140	1	1	1	3	2	1
3	422	405	2	3	4	3	3	2
4	1242	1201	11	13	15	13	14	12
5	1207	1162	12	14	12	13	11	14
6	821	828	9	8	10	12	11	8
7	998	976	8	9	10	12	10	8
8	847	795	7	9	9	11	10	8

noise variables are almost the same independent of their distributions (see Table 3.6), which is consistent with the expectation.

In Figure 3.1, we illustrate this procedure for $h = 4$ (the same as that summarized in the third row of Table 3.5), with plots of the number of terminal nodes, the integrated likelihood $p_{\mathrm{P}}(\boldsymbol{N}|\boldsymbol{X}, \boldsymbol{v}, \boldsymbol{\mathcal{T}})$ and the data likelihood $p_{\mathrm{P}}(\boldsymbol{N}|\boldsymbol{X}, \boldsymbol{v}, \bar{\boldsymbol{\lambda}}, \boldsymbol{\mathcal{T}})$ of the accepted trees. The observations are in line with those in Chipman *et al.* (1998); we see from the likelihood plots that the convergence of MCMC can be obtained relatively quickly. Interestingly, the optimal tree is not found in the first round of MCMC which got stuck in a local mode, but the restarts helped where in the second and the third rounds optimal trees can be found. Moreover, we see

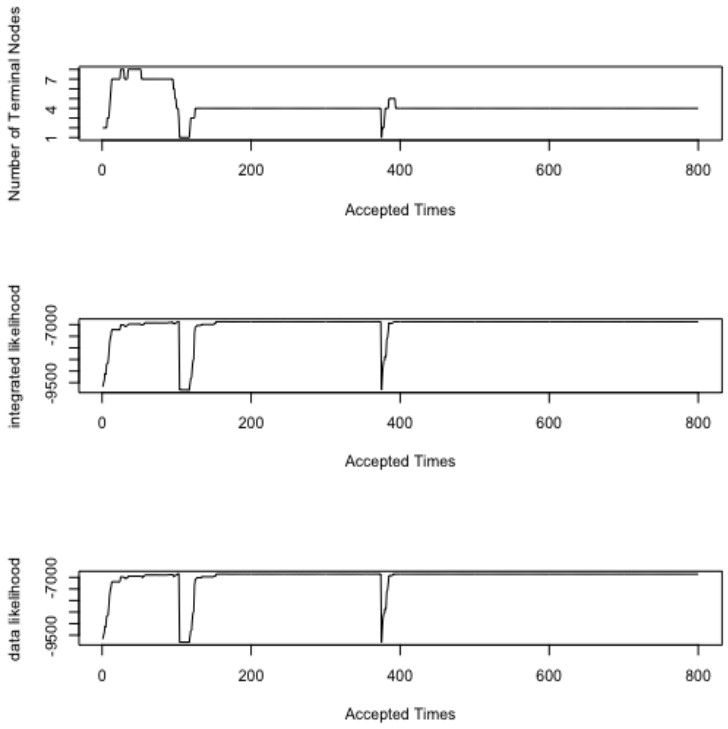

Figure 3.1: Trace plots from MCMC with 3 restarts ($\gamma = 0.99, \rho = 15$).

that there is no big difference shown in the plots of the integrated likelihood and the data likelihood.

Following Step 2 of Table 2.1, for each $h = 2, \ldots, 8$, we select the optimal tree with maximum data likelihood $p_{\mathrm{P}}(\boldsymbol{N}|\boldsymbol{X}, \boldsymbol{v}, \bar{\boldsymbol{\lambda}}, \boldsymbol{\mathcal{T}})$ from the convergence region. The variables used in these optimal trees are listed in Table 3.7, where we can see that none of these trees involves the noise variables. The values for the effective number of parameters p_D reflect the number of parameters in the tree if a flat prior for λ_t is used. Furthermore, we list the DIC for these trees in the last column of Table 3.7. Following Step 3 of Table 2.1 we conclude that the selected optimal tree is the one with 4 terminal nodes which is illustrated in Figure 3.2. We see that this tree is close to a true optimal one with the almost correct topology and accurate parameter estimates.

Using equal probabilities for the proposed tree moves, the above example provides detailed information about how to implement the three-step tree selection procedure in practice and illustrates the effectiveness of the method. Next, we investigate which type of step (particularly, the Change and Swap moves) contributes more to the computational efficiency. To this end, we shall vary the

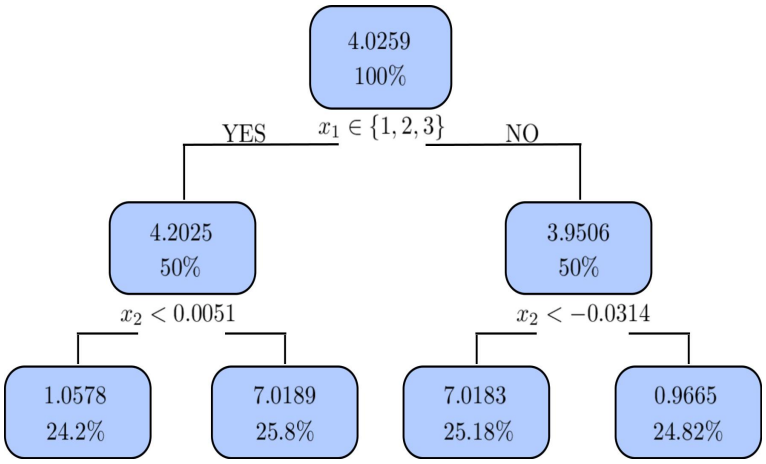

Figure 3.2: Optimal P-BCART. Numbers at each node give the estimated value for the frequency parameter λ_t and the percentage of observations.

Table 3.7: Number of times each variable was used in each chosen optimal tree and the corresponding p_D and DIC (after the burn-in period; equal probabilities for tree moves; one run with 3 restarts). Bold font indicates DIC selected model.

# terminal nodes	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	p_D	DIC
2	1	0	0	0	0	0	0	0	2.00	14221
3	1	1	0	0	0	0	0	0	2.95	14076
4	1	2	0	0	0	0	0	0	3.97	**13526**
5	2	2	0	0	0	0	0	0	4.97	13570
6	3	2	0	0	0	0	0	0	5.93	13629
7	3	3	0	0	0	0	0	0	6.91	13678
8	4	3	0	0	0	0	0	0	7.95	13683

probabilities of the Change and Swap moves, keeping the same probabilities for Grow and Prune moves at 0.2. Different experiments can be designed as in Table 3.8.

We fix $\gamma = 0.99$ and $\rho = 15$, as for Figure 3.1. For each of the experiments E1–E4, we run the P-BCART MCMC algorithms 10 times and for each run, we record the iteration time until an "optimal" tree is found. The average iteration time with the standard deviation (s.d.) of the 10 runs and the average acceptance rates of moves are shown in Table 3.9. The figures in the second row indicate that experiment E4 is faster in finding an "optimal" tree than E1–E3 when at

Table 3.8: Four different experiments (E1–E4) for given probabilities of tree moves. In each case, the probability of Grow and Prune is fixed at 0.2.

	Change1	Change2	Swap
E1	0	0.6	0
E2	0	0.3	0.3
E3	0.3	0	0.3
E4	0.2	0.2	0.2

Table 3.9: Average iteration times to obtain an "optimal" tree (4 terminal nodes) and accepted move rates from the P-BCART MCMC algorithms (after the burn-in period; ten runs with 3 restarts). Experiments E1–E4 are described in Table 3.8.

	E1	E2	E3	E4
Average iteration times (s.d.)	3388 (168)	2710 (187)	2984 (177)	2018 (161)
Acceptance rate of all moves	3.10%	3.23%	3.14%	3.87%
Acceptance rate of Grow	1.50%	1.36%	0.90%	0.65%
Acceptance rate of Prune	1.43%	1.20%	0.54%	0.30%
Acceptance rate of Change1	-	-	6.09%	8.19%
Acceptance rate of Change2	4.17%	4.87%	-	5.66%
Acceptance rate of Swap	-	4.01%	3.49%	4.52%

least one of the Change moves or/and the Swap move is removed. In particular, the comparison between E1 and E2 confirms the essence of the Swap move, as illustrated also in Chipman *et al.* (1998). Moreover, the acceptance rate of all moves is a weighted average of acceptance rates of all individual moves, and we observe that the acceptance rates of the Change and Swap moves (in particular, the Change1 move) are significantly greater than the Grow and Prune moves, which also confirms the significance of the Change and Swap moves (especially, the Change1 move).

We also run several other similar but more complex simulation examples to check the performance of P-BCART, NB-BCART, and ZIP-BCART models. In particular, we tested the case where the values of λ are closer (with 3.5 and 4.5). Our conclusions from these simulations are:

1. BCART models can retrieve the tree structure (including both topology and

parameters) as that used to simulate the data.

2. BCART models are able to avoid choosing noise variables regardless of their distributions.

3. The Change and Swap moves have significant impacts on the BCART models and it is beneficial to include two types of the Change move.

Remark 3.9 *The hyper-parameters α and β for the conjugate Gamma prior in the Poisson model can be estimated from the data, and their values appear not to be crucial as long as the relationship between them is maintained. We will adopt the same strategy used in this simulation example to estimate hyper-parameters in the following models in Chapters 4 and 5.*

3.5.2 ZIP Data with Varying Probability of Zero Mass Component

We simulate a data set $\{(\boldsymbol{x}_i, v_i, N_i)\}_{i=1}^n$ with $n = 5,000$ independent observations. Here $\boldsymbol{x}_i = (x_{i1}, x_{i2})$, with independent components $x_{ik} \sim N(0,1)$ for $k = 1, 2$. We assume exposure $v_i \equiv 1$ for simplicity since it is not a key feature here. Moreover, $N_i \sim \text{ZIP}(p_0, \lambda(x_{i1}, x_{i2}))$, where

$$\lambda(x_1, x_2) = \begin{cases} 7 & \text{if } x_1 x_2 \leq 0, \\ 1 & \text{if } x_1 x_2 > 0, \end{cases}$$

and $p_0 \in (0,1)$ is the probability of a zero due to the point mass component, for which the value is to be specified. The data is split into two subsets: a training set with $n - m = 4,000$ observations and a test set with $m = 1,000$ observations.

In this case, we aim to examine how the P-BCART, NB-BCART, and ZIP-BCART will perform when p_0 is varied. Note that since exposure $v_i \equiv 1$, NB1 and NB2 (ZIP1 and ZIP2) will be essentially the same. Intuition tells us that when p_0 is small NB-BCART should be good enough to capture the over-dispersion introduced by a small proportion of zeros, but when p_0 becomes large ZIP-BCART should perform better for the highly over-dispersed data. This intuition will be confirmed by this study. For simplicity, we shall present two results, one with $p_0 = 0.05$ and the other with $p_0 = 0.95$.

We first discuss the simulation with a small probability of zero mass (i.e., $p_0 = 0.05$). In Table 3.10 we present the hyper-parameters γ, ρ used to obtain MCMC convergence to the region of trees with a certain number of terminal nodes. The last two columns give the effective number of parameters and DIC of

Table 3.10: Hyper-parameters, p_D (or q_D,r_D) and DIC on training data ($p_0 = 0.05$). The number in the bracket after the abbreviation of the model indicates the number of terminal nodes for this tree. Bold font indicates DIC selected model.

Model	γ	ρ	p_D(or q_D, r_D)	DIC
ZIP-BCART (2)	0.50	20	4.00	11451
ZIP-BCART (3)	0.99	20	5.94	11405
ZIP-BCART (4)	0.99	15	7.95	**11322**
ZIP-BCART (5)	0.99	5	9.86	11364
P-BCART (2)	0.50	20	2.00	11369
P-BCART (3)	0.99	20	2.99	11337
P-BCART (4)	0.99	10	3.99	**11262**
P-BCART (5)	0.99	5	4.91	11299
NB-BCART (2)	0.50	30	4.00	11317
NB-BCART (3)	0.99	25	5.99	11273
NB-BCART (4)	0.99	20	7.99	**11192**
NB-BCART (5)	0.99	5	9.90	11237

the optimal trees for each model, respectively. We can conclude from the DIC that by using Step 3 in Table 2.1 we can select the optimal tree with the true 4 terminal nodes for either ZIP-BCART, P-BCART or NB-BCART, and among these, the NB-BCART (with DIC=11192) is the best one. This looks a bit surprising at first glance because our data are simulated from a ZIP model. We suspect that the reason for this may be two-fold: First, the NB is enough to capture the small over-dispersion. Second, we have used data augmentation in the algorithms and thus it is understandable that the NB-BCART with 1 latent variable (see Section 3.2) could achieve better performance than the "real" ZIP-BCART with 2 latent variables (see Section 3.3). Moreover, we see that even the P-BCART performs better than the ZIP-BCART, for similar reasons.

Now, let us look at the performance of these models on test data in Table 3.11. First, we see that for each type of model, ZIP, Poisson, and NB, the optimal tree with 4 terminal nodes achieves the best SE (0.00162, 0.00108, and 0.00070 respectively) and DS (0.000116, 0.000072, and 0.000056 respectively), which is not surprising as these models retrieve the almost true tree structures. Second, we see from the RSS($\boldsymbol{N}$) that for each type of model, the performance becomes better as the number of terminal nodes that we want increases, however, the amount of

Table 3.11: Model performance on test data ($p_0 = 0.05$) with bold entries determined by DIC (see Table 3.10). The number in the bracket after the abbreviation of the model indicates the number of terminal nodes for this tree.

Model	RSS($\boldsymbol{N}$)	SE	DS	NLL	Lift
ZIP-BCART (2)	2013	0.00222	0.000185	1975	1.22
ZIP-BCART (3)	1986	0.00208	0.000169	1953	2.67
ZIP-BCART (4)	1923	**0.00162**	**0.000116**	1890	6.34
ZIP-BCART (5)	1909	0.00182	0.000130	1863	6.56
P-BCART (2)	1758	0.00175	0.000138	1702	1.40
P-BCART (3)	1732	0.00160	0.000123	1673	3.21
P-BCART (4)	1681	**0.00108**	**0.000072**	1612	6.62
P-BCART (5)	1662	0.00126	0.000092	1594	6.75
NB-BCART (2)	1683	0.00145	0.000101	1647	1.58
NB-BCART (3)	1661	0.00131	0.000092	1616	3.53
NB-BCART (4)	1609	**0.00070**	**0.000056**	1536	6.95
NB-BCART (5)	1589	0.00097	0.000074	1502	6.97

improvement becomes smaller after the optimal trees with 4 terminal nodes have been obtained. We observe the same for NLL and lift. It is worth noting that when calculating and comparing lift for different trees, instead of simply following the four steps in Subsection 2.3.5, in Step 4 we first choose the minimum total sum of exposures among the least and most risky groups in all the trees to be compared in Table 3.11, and then calculate other values accordingly using this minimum total sum of exposures as the basis (e_{min} in the first paragraph of Step 4). Third, we see that among these three trees with 4 terminal nodes, the one obtained from NB-BCART gives the best performance on test data based on all these performance measures, which is consistent with the conclusion from training data.

Next, we consider the simulation with a large probability of zero mass (i.e., $p_0 = 0.95$). The results are displayed in Tables 3.12 and 3.13. Similar discussions can be done for this case. In particular, we find that the performance order based on DIC is ZIP-BCART>NB-BCART>P-BCART, which is also consistent with their performance on test data.

We also run several other similar simulation examples to check the performance of P-BCART, NB-BCART, and ZIP-BCART with different values for p_0. Our

Table 3.12: Hyper-parameters, p_D (or q_D,r_D) and DIC on training data ($p_0 = 0.95$). The number in the bracket after the abbreviation of the model indicates the number of terminal nodes for this tree. Bold font indicates DIC selected model.

Model	γ	ρ	p_D(or q_D, r_D)	DIC
ZIP-BCART (2)	0.50	10	3.99	3483
ZIP-BCART (3)	0.99	10	5.99	3452
ZIP-BCART (4)	0.99	8	7.95	**3375**
ZIP-BCART (5)	0.99	3	9.93	3396
P-BCART (2)	0.50	10	1.98	3892
P-BCART (3)	0.99	10	2.96	3863
P-BCART (4)	0.99	5	3.91	**3801**
P-BCART (5)	0.99	2	4.90	3827
NB-BCART (2)	0.50	20	3.99	3726
NB-BCART (3)	0.99	20	5.97	3699
NB-BCART (4)	0.99	10	7.92	**3632**
NB-BCART (5)	0.99	8	9.89	3667

conclusion from these simulations is that when the proportion of zeros in the data is small (reflected by small p_0), the NB-BCART or P-BCART performs better than ZIP-BCART, whereas when the proportion of zeros in the data is large, the ZIP-BCART is preferred to NB-BCART and P-BCART. This finding is consistent with the real insurance data discussed in Chapter 6.

3.5.3 Different Ways to Incorporate Exposure in ZIP Models

The purpose of this case is to compare two different ways of dealing with exposure, namely, ZIP1-BCART and ZIP2-BCART. To this end, we simulate a data set $\{(\boldsymbol{x}_i, v_i, N_i)\}_{i=1}^n$ with $n = 5,000$ independent observations. Here $v_i \sim \mathrm{U}(0,1)$, $\boldsymbol{x}_i = (x_{i1}, x_{i2})$, with independent components $x_{ik} \sim N(0,1)$ for $k = 1, 2$. Moreover, $N_i \sim \mathrm{ZIP}(p_i^{(\tau)}, \lambda(x_{i1}, x_{i2})\, v_i)$, where

$$\lambda(x_1, x_2) = \begin{cases} 7 & \text{if } x_1 x_2 \leq 0, \\ 1 & \text{if } x_1 x_2 > 0, \end{cases}$$

Table 3.13: Model performance on test data ($p_0 = 0.95$) with bold entries determined by DIC (see Table 3.12). The number in the bracket after the abbreviation of the model indicates the number of terminal nodes for this tree.

Model	RSS($\boldsymbol{N}$)	SE	DS	NLL	Lift
ZIP-BCART (2)	721	0.00755	0.00721	699	1.25
ZIP-BCART (3)	715	0.00700	0.00698	690	1.92
ZIP-BCART (4)	682	**0.00571**	**0.00619**	657	2.86
ZIP-BCART (5)	675	0.00613	0.00646	649	3.13
P-BCART (2)	782	0.00967	0.00802	754	1.15
P-BCART (3)	773	0.00891	0.00786	746	1.50
P-BCART (4)	750	**0.00723**	**0.00712**	719	2.40
P-BCART (5)	741	0.00792	0.00739	705	2.72
NB-BCART (2)	775	0.00893	0.00775	740	1.19
NB-BCART (3)	768	0.00810	0.00740	731	1.72
NB-BCART (4)	735	**0.00647**	**0.00667**	701	2.60
NB-BCART (5)	730	0.00703	0.00689	693	2.90

and the probability of zero mass component is given as

$$p_i^{(\tau)} = \frac{\mu(x_{i1}, x_{i2})}{v_i^\tau + \mu(x_{i1}, x_{i2})}, \quad \text{with} \quad \mu(x_{i1}, x_{i2}) \equiv 0.5,$$

and some $\tau \geq 0$ to be specified below. The data is split into two subsets, namely a training set with $n - m = 4,000$ observations and a test set with $m = 1,000$ observations.

In the above simulation setup, we include exposure in both the Poisson component and the zero mass component. In this way, it is not clear which of ZIP1-BCART and ZIP2-BCART will outperform the other. That being said, we could vary the value of τ to control the effect of exposure to the zero mass component. We shall consider two extreme cases, one with a very small τ and the other with a very large τ. More precisely, for a large τ we choose $\tau = 100$. In this case, since many v_i^τ will be small, we have that $p_i^{(\tau)}$ will be close to one, which implies that the Poisson component should play a minor role in exposure modelling and thus we would expect that ZIP2-BCART has a better ability to capture this. On the other hand, for a small value $\tau = 0.0001$, since many v_i^τ will be close to 1 we have that $p_i^{(\tau)}$ will be almost independent of v_i, which implies that zero mass component should play a minor role in exposure modelling. Thus we would expect that

Table 3.14: DIC for ZIP-BCART models with different values of τ on training data. The number in the bracket after the abbreviation of the model indicates the number of terminal nodes for this tree. Bold font indicates DIC selected model.

Model	DIC ($\tau = 100$)	DIC ($\tau = 0.0001$)
ZIP1-BCART (2)	3091	10515
ZIP1-BCART (3)	3055	10437
ZIP1-BCART (4)	**2976**	**10273**
ZIP1-BCART (5)	2997	10330
ZIP2-BCART (2)	2653	10924
ZIP2-BCART (3)	2637	10843
ZIP2-BCART (4)	**2613**	**10685**
ZIP2-BCART (5)	2627	10751

ZIP1-BCART has a better ability to capture this. We report DIC for these two cases in Table 3.14. The model performances on test data are listed in Table 3.15 for $\tau = 100$ and Table 3.16 for $\tau = 0.0001$. From these tables, we can confirm the above intuition that ZIP1-BCART should perform better for small τ and worse for large τ (compared to ZIP2-BCART). We conclude from this simulation study that the ZIP2-BCART works better in capturing the potentially stronger effect of the exposure to the zero mass component, which is also illustrated in the real insurance data discussed in Chapter 6.

3.6 Summary of Chapter 3

In this chapter, we have discussed the application of different distributions (Poisson, NB, ZIP, and ZINB) in BCART models for claims frequency analysis. In particular, to improve model performance, we explored using different ways of handling exposure, as well as employing different data augmentation methods. From our simulation studies, we obtained the following conclusions.

1. BCART models not only can retrieve the tree structure (including both topology and parameters) but also avoid the selection of noise variables. Furthermore, Change and Swap moves, particularly the Change1 move, have a significant impact on computational efficiency in BCART models.

Table 3.15: Model performance on test data ($\tau = 100$) with bold entries determined by DIC (see Table 3.14). The number in the bracket after the abbreviation of the model indicates the number of terminal nodes for this tree.

Model	RSS($\boldsymbol{N}$)	SE (in 10^{-5})	DS	NLL	Lift
ZIP1-BCART (2)	2423	3.06	0.00281	1339	1.01
ZIP1-BCART (3)	2417	2.98	0.00259	1330	1.33
ZIP1-BCART (4)	2376	**2.20**	**0.00209**	1308	1.78
ZIP1-BCART (5)	2333	2.63	0.00219	1302	1.81
ZIP2-BCART (2)	2072	2.76	0.00234	1324	1.06
ZIP2-BCART (3)	2069	2.57	0.00207	1317	1.46
ZIP2-BCART (4)	2056	**2.02**	**0.00179**	1304	1.97
ZIP2-BCART (5)	2049	2.06	0.00189	1295	2.08

Table 3.16: Model performance on test data ($\tau = 0.0001$) with bold entries determined by DIC (see Table 3.14). The number in the bracket after the abbreviation of the model indicates the number of terminal nodes for this tree.

Model	RSS($\boldsymbol{N}$)	SE	DS	NLL	Lift
ZIP2-BCART (2)	6859	0.0093	0.0080	4185	1.02
ZIP2-BCART (3)	6648	0.0080	0.0069	4092	2.10
ZIP2-BCART (4)	6408	**0.0060**	**0.0050**	3913	3.40
ZIP2-BCART (5)	6320	0.0073	0.0062	3853	3.48
ZIP1-BCART (2)	6628	0.0079	0.0072	3827	1.07
ZIP1-BCART (3)	6535	0.0058	0.0055	3763	2.15
ZIP1-BCART (4)	6350	**0.0027**	**0.0024**	3590	3.45
ZIP1-BCART (5)	6282	0.0036	0.0033	3543	3.62

2. When comparing the performance of P-BCART, NB-BCART, and ZIP-BCART, if the proportion of zeros in the data is small, NB-BCART or P-BCART performs better than ZIP-BCART. Conversely, if the proportion of zeros is large especially when it is 10% or higher, ZIP-BCART outperforms NB-BCART and P-BCART. This provides a strategy for selecting models in practical applications by first observing data characteristics. Besides, it raises the idea that in real insurance data with a large number of zeros, ZIP

models should perform the best among these three, which will be confirmed in the real data analyses; see Chapter 6.

3. In the comparison among ZIP models that embed the exposure in different ways, the ZIP2-BCART performs better in capturing the potentially stronger effect of the exposure to the zero mass component, as is also demonstrated in the real insurance data discussed in Chapter 6.

4. It should be noted that a more complex way of embedding exposure does not necessarily lead to better model performance. For example, comparing the ZIP3 model, which embeds exposure into both the Poisson part and zero mass part, with the ZIP2 model, which embeds exposure only in the zero mass part, the former shows no significant improvement. Similarly, it does not imply that a more general model will necessarily yield better performance. For example, ZINB models do not exhibit substantial improvement compared to ZIP models. Additionally, the more complex the model, the lower its computational efficiency. We believe that striking a balance between model performance, efficiency, and complexity is a topic worthy of further exploration.

Chapter 4

Severity Modelling with Bayesian CART

This chapter introduces Bayesian CART models for insurance claims severity. Assume that for each policyholder i ($i = 1, 2, \ldots, n$), the individual claim amounts $Y_{i1}, Y_{i2}, ..., Y_{i_{Ni}}$, given $N_i > 0$, are IID, and *claims severity* refers to the average claim amount per claim

$$\bar{S} = \frac{\sum_{j=1}^{N_i} Y_{ij}}{N_i}.$$

Different models can be used to characterize the behaviour of claim amounts as a function of the explanatory variables. Specifically, there are two ways for claim amount modelling with different response variables; one is to model individual claim amounts $Y_{i1}, Y_{i2}, ..., Y_{i_{Ni}}$, and the other way is to directly model the average claim amount $\bar{S}_i$; see, e.g., Henckaerts *et al.* (2021), Frees *et al.* (2014) and Omari *et al.* (2018). Since the latter is more intuitive and straightforward to obtain claims severity, we shall follow this way within this chapter. Three commonly used distributions in the literature for claims severity modelling, along with their respective properties, are discussed in the first section, namely, Gamma, LogNormal, and Weibull distributions; see, e.g., Wüthrich & Merz (2023). Following that, we demonstrate their applications in BCART models in detail. Specific formulas for some of the evaluation metrics for each model are provided in their respective sections. Subsequently, one simulation example is designed to investigate the performance of the above models, especially their ability to fit the data in each group (terminal node) with varying tail characteristics, and some conclusions are drawn. It is worth noting that when dealing with claims severity data in this chapter, the data with zero claims will be omitted.

4.1 Distributions for Claims Severity

The majority of claims severity data typically display characteristics of positive skewness or heavy tails. Therefore, statistical distributions capable of capturing these features may be suitable for modelling such claims, such as Gamma, Log-Normal, Weibull, Pareto, and more generalized distributions; see, e.g., Wüthrich & Merz (2008). Given that the heavy tail represents significant claim amounts and risks, insurers often pay more attention to the tail of the distribution. This section shall discuss three commonly used distributions, namely, Gamma, LogNormal, and Weibull, and analyze their tail characteristics based on their properties.

The Gamma distribution is a right-skewed, continuous probability distribution with the tail of the distribution considered "light". The probability density function (pdf) is given as:

$$f_{\mathrm{G}}\left(\bar{S} \mid \alpha, \beta\right) = \frac{\beta^{\alpha} \bar{S}^{\alpha-1} e^{-\beta \bar{S}}}{\Gamma(\alpha)}, \tag{4.1}$$

where both the shape parameter α and the rate parameter β are greater than zero. If $\bar{S}_i \sim \text{Gamma}(\alpha, \beta)$, then the mean and variance of $\bar{S}_i$ are given by

$$\mathbb{E}(\bar{S}_i \mid \alpha, \beta) = \frac{\alpha}{\beta}, \qquad \text{Var}(\bar{S}_i \mid \alpha, \beta) = \frac{\alpha}{\beta^2}. \tag{4.2}$$

The LogNormal distribution is a skewed distribution with a low mean value, large variance, and a somewhat heavier tail than the Gamma distribution. It has significantly higher probabilities of large or extreme values and its pdf is given as:

$$f_{\mathrm{LN}}\left(\bar{S} \mid \mu, \sigma\right) = \frac{1}{\bar{S}\sigma\sqrt{2\pi}} \exp\left(-\frac{(\log(\bar{S}) - \mu)^2}{2\sigma^2}\right), \tag{4.3}$$

where $\mu \in \mathbb{R}$ and $\sigma > 0$. If $\bar{S}_i \sim \text{LN}(\mu, \sigma^2)$, then the mean and variance of $\bar{S}_i$ are given by

$$\mathbb{E}(\bar{S}_i \mid \mu, \sigma) = \exp(\mu + \sigma^2/2), \tag{4.4}$$

$$\text{Var}(\bar{S}_i \mid \mu, \sigma) = (\exp(\sigma^2) - 1) \exp\left(2\mu + \sigma^2\right). \tag{4.5}$$

The Weibull distribution is widely used due to its versatility, particularly in modelling data with a high degree of positive skewness. There are different ways to parameterize the Weibull distribution, either with two or three parameters; see, e.g., Rinne (2008). For simplicity, we adopt the common parameterization with two parameters; see, e.g., Fink (1997). The pdf of a Weibull distribution is given as:

$$f_{\mathrm{Weib}}\left(\bar{S} \mid \alpha, \beta\right) = \frac{\alpha}{\beta} \bar{S}^{\alpha-1} \exp(-\bar{S}^{\alpha}/\beta), \tag{4.6}$$

where both shape parameter α and scale parameter β are greater than zero. If $\bar{S}_i \sim \text{Weib}(\alpha, \beta)$, then the mean and variance of $\bar{S}_i$ are given by

$$\mathbb{E}(\bar{S}_i \mid \alpha, \beta) = \beta\Gamma(1 + 1/\alpha), \tag{4.7}$$

$$\text{Var}(\bar{S}_i \mid \alpha, \beta) = \beta^2 \left(\Gamma(1 + 2/\alpha) - \big(\Gamma(1 + 1/\alpha)\big)^2\right). \tag{4.8}$$

The Gamma, LogNormal and Weibull distributions appear to be similar because all of them can accommodate data with positive skewness. Furthermore, the fact that both LogNormal and Weibull distributions can handle data with heavy tails makes them more similar. Selecting among these three models poses a considerable challenge, and scholars have extensively explored this topic; see, e.g., Siswadi & Quesenberry (1982). In claims severity modelling, insurers want to gain more insights into the (right) tail, which describes the behaviour of the distribution at large values. To investigate the tail characteristics, the proper approach is to analyse the distribution function F rather than its density, which is sometimes unknown in many real-world situations. Specifically, since F must asymptotically approach 1 for large arguments, exploring how quickly F approaches that asymptote is necessary. Thus, we need to investigate the behaviour of its survival function $1 - F(x)$ as $x \to \infty$. In particular, distribution F is considered "heavier" than G if and only if F eventually has a higher probability at large values than G, which can be formalized: there must exist a finite number x_0 such that for all $x > x_0$,

$$\text{P}_\text{F}(X > x) = 1 - F(x) > 1 - G(x) = \text{P}_\text{G}(X > x).$$

Based on this discussion, we can directly analyze the survival functions of the Gamma and LogNormal distributions, expanding them around $x \to \infty$ to discover their asymptotic behaviour. The conclusion is that LogNormal distributions have heavier tails than Gamma distributions. On the other hand, both the Gamma and Weibull distributions can be seen as generalisations of the Exponential distribution. By comparing their pdfs (see (4.1) and (4.6)), we can observe the difference in effect. Omitting all the normalising constants, it is evident that the pdf of the Weibull distribution drops off significantly more quickly (for $\alpha > 1$), resulting in light tails or slowly (for $\alpha < 1$), resulting in heavier tails than the Gamma distribution. Both of them reduce to the Exponential distribution when $\alpha = 1$.

In summary, concerning the modelling of insurance losses, the Gamma distribution would be a suitable model for losses that are not catastrophic, such as auto insurance. Additionally, the LogNormal distribution is more suitable for fire insurance, which may exhibit more extreme values than auto insurance. Moreover,

as discussed, the Weibull distribution has the ability to handle different cases by tuning the shape parameter to adapt to different tail characteristics. In each of the following sections, we shall demonstrate how to apply these distributions in BCART models.

4.2 Gamma-Bayesian CART

Consider a tree $\mathcal{T}$ with b terminal nodes as before (see Section 2.2). In a Gamma model, we assume all insurance policyholders $i = 1, 2, \ldots, n$ have independent average claim amounts $\bar{S}_i$ with

$$\bar{S}_i \mid \boldsymbol{x} \ \sim \ \text{Gamma}\left(\alpha(\boldsymbol{x}), \beta(\boldsymbol{x})\right)$$

for the i-th observation, where $\alpha(\boldsymbol{x}) = \sum_{t=1}^{b} \alpha_t I_{(\boldsymbol{x}\in\mathcal{A}_t)}$, $\beta(\boldsymbol{x}) = \sum_{t=1}^{b} \beta_t I_{(\boldsymbol{x}\in\mathcal{A}_t)}$, $\{\mathcal{A}_t\}$ is a partition of $\mathcal{X}$, and t denotes the t-th terminal node. The aim is to estimate the regression functions $\alpha(\cdot)$ and $\beta(\cdot)$, describing the expected claims severity. For terminal node t, we denote the associated data as $(\boldsymbol{X}_t, \bar{\boldsymbol{S}}_{\boldsymbol{t}}) = ((X_{t1}, \overline{S}_{t1}), \ldots, (X_{tn_t}, \overline{S}_{tn_t}))^\top$. To explicitly derive the posterior distribution (see discussions in Subsection 2.2.1), we choose a common Gamma prior for β_t ($t = 1, 2, \ldots, b$) with hyper-parameters $\alpha_\pi, \beta_\pi > 0$ (cf. (3.2)) and use the same way to deal with α_t as in Section 3.2, i.e., treating it as known (using MME to estimate),

$$\hat{\alpha}_t = \frac{(\bar{S})_t^2}{\text{Var}(\bar{S})_t}, \tag{4.9}$$

where $(\bar{S})_t$ and $\text{Var}(\bar{S})_t$ denote the mean and variance of the claims severity in the t-th node respectively. With the above Gamma prior and the estimated parameter $\hat{\alpha}_t$, the integrated likelihood for terminal node t can be obtained as

$$\begin{aligned}
&p_{\text{G}}\left(\bar{\boldsymbol{S}}_t \mid \boldsymbol{X}_t\right) \\
&= \int_0^\infty f_{\text{G}}\left(\bar{\boldsymbol{S}}_t \mid \hat{\alpha}_t, \beta_t\right) p(\beta_t) d\beta_t \\
&= \int_0^\infty \prod_{j=1}^{n_t} \frac{\beta_t^{\hat{\alpha}_t} \overline{S}_{tj}^{\hat{\alpha}_t - 1} e^{-\beta_t \overline{S}_{tj}}}{\Gamma(\hat{\alpha}_t)} \frac{\beta_\pi^{\alpha_\pi} \beta_t^{\alpha_\pi - 1} e^{-\beta_\pi \beta_t}}{\Gamma(\alpha_\pi)} d\beta_t \\
&= \frac{\beta_\pi^{\alpha_\pi} \prod_{j=1}^{n_t} \overline{S}_{tj}^{\hat{\alpha}_t - 1}}{\Gamma(\alpha_\pi)\Gamma(\hat{\alpha}_t)^{n_t}} \int_0^\infty \beta_t^{n_t\hat{\alpha}_t + \alpha_\pi - 1} e^{-(\sum_{j=1}^{n_t} \overline{S}_{tj} + \beta_\pi)\beta_t} d\beta_t \\
&= \frac{\beta_\pi^{\alpha_\pi} \prod_{j=1}^{n_t} \overline{S}_{tj}^{\hat{\alpha}_t - 1}}{\Gamma(\alpha_\pi)\Gamma(\hat{\alpha}_t)^{n_t}} \frac{\Gamma(n_t\hat{\alpha}_t + \alpha_\pi)}{(\sum_{j=1}^{n_t} \overline{S}_{tj} + \beta_\pi)^{n_t\hat{\alpha}_t + \alpha_\pi}}.
\end{aligned} \tag{4.10}$$

Clearly, from (4.10), we see that the posterior distribution of β_t, conditional on $\bar{\boldsymbol{S}}_t$, is given by

$$\beta_t \mid \bar{\boldsymbol{S}}_t \quad \sim \quad \text{Gamma}\left(n_t\hat{\alpha}_t + \alpha_\pi, \sum_{j=1}^{n_t} \overline{S}_{tj} + \beta_\pi\right).$$

The integrated likelihood for the tree $\mathcal{T}$ is thus given by

$$p_{\text{G}}\left(\bar{\boldsymbol{S}} \mid \boldsymbol{X}, \hat{\boldsymbol{\alpha}}, \mathcal{T}\right) = \prod_{t=1}^{b} p_{\text{G}}\left(\bar{\boldsymbol{S}}_t \mid \boldsymbol{X}_t, \hat{\alpha}_t\right). \tag{4.11}$$

Next, we discuss the DIC for this tree. Since we only consider uncertainty for $\boldsymbol{\beta}$ but not for $\boldsymbol{\alpha}$ and there is no data augmentation involved, the three different DICs defined in Subsections 2.2.4 and 3.2.1 cannot be adopted directly. Thus, using again the idea that DIC="goodness of fit"+"complexity", we can introduce a new DIC_t for terminal node t as follows

$$\text{DIC}_t = D(\bar{\beta}_t) + 2s_{Dt}.$$

Here, the goodness of fit is given by

$$\begin{aligned} D(\bar{\beta}_t) &= -2\sum_{j=1}^{n_t} \log f_{\text{G}}(\bar{S}_{tj} \mid \hat{\alpha}_t, \bar{\beta}_t) \\ &= -2\sum_{j=1}^{n_t} \left[\hat{\alpha}_t \log(\bar{\beta}_t) + (\hat{\alpha}_t - 1)\log(\overline{S}_{tj}) - \bar{\beta}_t\overline{S}_{tj} - \log\left(\Gamma(\hat{\alpha}_t)\right)\right], \end{aligned} \tag{4.12}$$

and the effective number of parameters s_{Dt} is given by

$$\begin{aligned} s_{Dt} &= \overline{D(\boldsymbol{\theta}_t)} - D(\bar{\boldsymbol{\theta}}_t) \\ &= -2\mathbb{E}_{\text{post}}\left[\log\left(f(\boldsymbol{y}_t \mid \boldsymbol{\theta}_t)\right)\right] + 2\log\left(f(\boldsymbol{y}_t \mid \bar{\boldsymbol{\theta}}_t)\right) \\ &= 1 + 2\sum_{j=1}^{n_t}\left\{\log(f_{\text{G}}(\bar{S}_{tj} \mid \hat{\alpha}_t, \bar{\beta}_t)) - \mathbb{E}_{\text{post}}\left[\log\left(f_{\text{G}}(\bar{S}_{tj} \mid \hat{\alpha}_t, \beta_t)\right)\right]\right\}, \end{aligned} \tag{4.13}$$

where α_t is treated as known while remaining a model parameter, and we denote its effective number as 1; the second part of the last line is for β_t,

$$\bar{\beta}_t = \frac{n_t\hat{\alpha}_t + \alpha_\pi}{\sum_{j=1}^{n_t}\overline{S}_{tj} + \beta_\pi}. \tag{4.14}$$

Therefore, a direct calculation shows that the effective number of parameters for terminal node t is given by

$$s_{Dt} = 1 + 2\left(\log\left(n_t\hat{\alpha}_t + \alpha_\pi\right) - \psi\left(n_t\hat{\alpha}_t + \alpha_\pi\right)\right)n_t\hat{\alpha}_t, \tag{4.15}$$

and thus

$$
\begin{aligned}
&\mathrm{DIC}_t \\
&= D(\overline{\beta}_t) + 2s_{Dt} \\
&= -2\sum_{j=1}^{n_t}\left(\hat{\alpha}_t \log\left(\frac{n_t\hat{\alpha}_t+\alpha_\pi}{\sum_{j=1}^{n_t}\overline{S}_{tj}+\beta_\pi}\right) + (\hat{\alpha}_t-1)\log(\overline{S}_{tj}) - \frac{n_t\hat{\alpha}_t+\alpha_\pi}{\sum_{j=1}^{n_t}\overline{S}_{tj}+\beta_\pi}\overline{S}_{tj}\right) \\
&\quad + 2\sum_{j=1}^{n_t}(\log(\Gamma(\hat{\alpha}_t))) + 2 + 4\left(\log\left(n_t\hat{\alpha}_t+\alpha_\pi\right) - \psi\left(n_t\hat{\alpha}_t+\alpha_\pi\right)\right) n_t\hat{\alpha}_t.
\end{aligned}
$$

Then the DIC of the tree $\mathcal{T}$ is obtained as

$$
\mathrm{DIC} := \sum_{t=1}^{b} \mathrm{DIC}_t. \tag{4.16}
$$

We extend Algorithm 2.2 (see Subsection 2.2.2) to a new Algorithm 4.1 based on the above discussion, which simulates a Markov chain sequence of pairs $(\boldsymbol{\theta}^{(1)}, \mathcal{T}^{(1)})$, $(\boldsymbol{\theta}^{(2)}, \mathcal{T}^{(2)}), \ldots$, starting from the root node. For the convenience of reference, we shall describe a general algorithm that covers the Gamma BCART models as a special case. More precisely, $\boldsymbol{\theta} = (\boldsymbol{\theta_M}, \boldsymbol{\theta_B})$, where $\boldsymbol{\theta_M}$ is the parameter that is treated as known and is computed using MME (or MLE), and $\boldsymbol{\theta_B}$ is the unknown parameter estimated in the Bayesian framework. This newly proposed algorithm can be used when it is necessary to estimate parameters in different ways without involving data augmentation techniques.

With the above formulas derived for the Gamma case, we can use the three-step approach proposed in Subsection 2.2.4, together with Algorithm 4.1 (treat $\boldsymbol{\theta_M} = \boldsymbol{\alpha}$ and $\boldsymbol{\theta_B} = \boldsymbol{\beta}$), to search for an optimal tree which can then be used to predict new data. Given an optimal tree, the estimated claims severity $\hat{\alpha}_t/\bar{\beta}_t$ in each terminal node t can be determined using (4.9) and (4.14). Some of the evaluation metrics based on Gamma distribution are provided in Table 4.1.

Remark 4.1 *(a) As before, the sampling steps in Algorithm 4.1 should be done when necessary. Besides, Algorithm 4.1 can also be easily extended to accommodate multivariate parameters for both* $\boldsymbol{\theta_M}$ *and* $\boldsymbol{\theta_B}$.

(b) There is another way to model $\bar{S}_i$ *considering* N_i *as model weights, where the individual claim amounts* Y_{ij} *are assumed to follow a common Gamma distribution independently and we can obtain the distribution of* $\bar{S}_i$ *based on the additive property of the Gamma distribution. That is, if* $Y_{ij} \sim Gamma(\alpha, \beta)$, *then* $\sum_{j=1}^{N_i} Y_{ij} \sim Gamma(\alpha, N_i\beta)$; *thus* $\bar{S}_i = \sum_{j=1}^{N_i} Y_{ij}/N_i \sim Gamma(N_i\alpha, N_i\beta)$. *Since*

Algorithm 4.1: One step of the MCMC algorithm for the BCART models parameterized by $(\boldsymbol{\theta_M}, \boldsymbol{\theta_B}, \mathcal{T})$ with both known and unknown parameters

Input: Data $(\boldsymbol{X}, \boldsymbol{y})$ and current values $(\hat{\boldsymbol{\theta}}_{\boldsymbol{M}}^{(m)}, \boldsymbol{\theta}_{\boldsymbol{B}}^{(m)}, \mathcal{T}^{(m)})$

1: Generate a candidate value $\mathcal{T}^*$ with probability distribution $q(\mathcal{T}^{(m)}, \mathcal{T}^*)$

2: Estimate $\hat{\boldsymbol{\theta}}_{\boldsymbol{M}}^{(m+1)}$, using MME (or MLE)

3: Set the acceptance ratio

$$\alpha(\mathcal{T}^{(m)}, \mathcal{T}^*) = \min\left\{\frac{q(\mathcal{T}^*, \mathcal{T}^{(m)})p(\boldsymbol{y} \mid \boldsymbol{X}, \hat{\boldsymbol{\theta}}_{\boldsymbol{M}}^{(m+1)}, \mathcal{T}^*)p(\mathcal{T}^*)}{q(\mathcal{T}^{(m)}, \mathcal{T}^*)p(\boldsymbol{y} \mid \boldsymbol{X}, \hat{\boldsymbol{\theta}}_{\boldsymbol{M}}^{(m)}, \mathcal{T}^{(m)})p(\mathcal{T}^{(m)})}, 1\right\}$$

4: Update $\mathcal{T}^{(m+1)} = \mathcal{T}^*$ with probability $\alpha(\mathcal{T}^{(m)}, \mathcal{T}^*)$, otherwise, set $\mathcal{T}^{(m+1)} = \mathcal{T}^{(m)}$

5: Sample $\boldsymbol{\theta}_{\boldsymbol{B}}^{(m+1)} \sim p(\boldsymbol{\theta_B} \mid \boldsymbol{X}, \boldsymbol{y}, \hat{\boldsymbol{\theta}}_{\boldsymbol{M}}^{(m+1)}, \mathcal{T}^{(m+1)})$

Output: New values $(\hat{\boldsymbol{\theta}}_{\boldsymbol{M}}^{(m+1)}, \boldsymbol{\theta}_{\boldsymbol{B}}^{(m+1)}, \mathcal{T}^{(m+1)})$

Table 4.1: Evaluation metrics for Gamma-BCART. ϵ_t denotes the empirical claims severity in node t, computed as $\sum_{j=1}^{n_t} S_{tj} / \sum_{j=1}^{n_t} N_{tj}$. $\hat{\alpha}_t$ and $\bar{\beta}_t$ are parameter estimations that can be obtained from (4.9) and (4.14) respectively.

	Formulas
RSS($\bar{\boldsymbol{S}}$)	$\sum_{t=1}^{b}\sum_{j=1}^{n_t}(\bar{S}_{tj} - \hat{\alpha}_t/\bar{\beta}_t)^2$
SE	$\sum_{t=1}^{b}\left(\epsilon_t - \hat{\alpha}_t/\bar{\beta}_t\right)^2$
DS	$\sum_{t=1}^{b}(\bar{\beta}_t^2/\hat{\alpha}_t)\left(\epsilon_t - \hat{\alpha}_t/\bar{\beta}_t\right)^2$

the other two distributions discussed in this chapter do not have the additive property, in this chapter, we do not pursue this modelling approach. This approach will be discussed in detail later in Section 5.1.

(c) When dealing with the Gamma distribution in the Bayesian framework, two alternative approaches can be considered.

- *Treat the rate parameter β as known and use a prior for the shape parameter α, i.e., $p(\alpha) \propto a_0^{\alpha-1}\beta^{\alpha c_0}/\Gamma(\alpha)^{b_0}$ where a_0, b_0, c_0 are prior hyper-parameters.*
- *Treat both the shape parameter α and the rate parameter β as unknown and*

use a joint prior for them, i.e., $p(\alpha, \beta) \propto {a_0}^{\alpha-1} e^{-\beta b_0} / \left(\Gamma(\alpha)^{c_0} \beta^{-\alpha d_0}\right)$ *where* a_0, b_0, c_0, d_0 *are prior hyper-parameters; see, e.g., Fink (1997).*

Although the joint prior can obtain estimators for α *and* β *simultaneously in the Bayesian framework, it is not formulated as an exact distribution, leading to less accurate estimators. The first way also has this shortcoming. Therefore, we do not use them in our implementation.*

4.3 LogNormal-Bayesian CART

Consider a tree $\mathcal{T}$ with b terminal nodes as before. In the LogNormal model, we assume that $\bar{S}_i \mid \boldsymbol{x}$ follows a LogNormal distribution. By using (4.3), the data likelihood for terminal node t can be obtained as

$$\begin{aligned} f_{LN}\left(\bar{\boldsymbol{S}}_t \mid \mu_t, \sigma_t\right) &= \frac{1}{\prod_{j=1}^{n_t} \bar{S}_{tj}} \frac{1}{(2\pi)^{n_t/2}} \frac{1}{\sigma_t^{n_t}} \exp\left[-\frac{1}{2\sigma_t^2}\left(n_t r_t^2 + n_t(\bar{w}_t - \mu_t)^2\right)\right] \\ &\propto (1/\sigma_t^2)^{n_t/2} \exp\left(-\frac{n_t}{2\sigma_t^2}(\bar{w}_t - \mu_t)^2\right) \exp\left(-\frac{n_t r_t^2}{2\sigma_t^2}\right) \\ &\propto \exp\left(-\frac{n_t}{2\sigma_t^2}(\bar{w}_t - \mu_t)^2\right), \end{aligned}$$

where $\bar{w}_t = \sum_{j=1}^{n_t} \log(\bar{S}_{tj})/n_t$ denotes the empirical mean and $r_t^2 = \sum_{j=1}^{n_t} \left(\log(\bar{S}_{tj}) - \bar{w}_t\right)^2 / n_t$ represents the empirical variance. Given the specified form of the likelihood, the appropriate choice for the conjugate prior is a Normal distribution

$$p(\mu_t) = \frac{1}{\sigma_\pi \sqrt{2\pi}} \exp\left(-\frac{1}{2}\left(\frac{\mu_t - \mu_\pi}{\sigma_\pi}\right)^2\right) \tag{4.17}$$

with hyper-parameters $\mu_\pi \in \mathbb{R}$ and $\sigma_\pi^2 > 0$. We use the same method to deal with σ_t as in Section 3.2, i.e., treating it as known (using MME to estimate). With the above Normal prior and the estimated parameter $\hat{\sigma}_t$, the integrated likelihood for terminal node t can be obtained as

$$\begin{aligned} & p_{\mathrm{LN}}\left(\bar{\boldsymbol{S}}_t \mid \boldsymbol{X}_t\right) \\ &= \int_{-\infty}^{\infty} f_{\mathrm{LN}}\left(\bar{\boldsymbol{S}}_t \mid \mu_t, \hat{\sigma}_t\right) p(\mu_t) d\mu_t \\ &= \int_{-\infty}^{\infty} \prod_{j=1}^{n_t} \frac{\sqrt{2\pi}}{\bar{S}_{tj}\hat{\sigma}_t} \exp\left(-\frac{(\log(\bar{S}_{tj}) - \mu_t)^2}{2\hat{\sigma}_t^2}\right) \frac{1}{\sigma_\pi\sqrt{2\pi}} \exp\left(-\frac{1}{2}\left(\frac{\mu_t - \mu_\pi}{\sigma_\pi}\right)^2\right) d\mu_t \\ &= \frac{1}{\hat{\sigma}_t^{n_t} \sigma_\pi (2\pi)^{(n_t/2+1)} \prod_{j=1}^{n_t} \bar{S}_{tj}} \end{aligned}$$

$$\times \int_{-\infty}^{\infty} \exp\left[\frac{-1}{2\hat{\sigma}_t^2}\sum_{j=1}^{n_t}\left(\log(\bar{S}_{tj})^2+\mu_t^2-2\log(\bar{S}_{tj})\mu_t\right)+\frac{-1}{2\sigma_\pi^2}\left(\mu_t^2+\mu_\pi^2-2\mu_\pi\mu_t\right)\right]d\mu_t$$

$$= \frac{1}{\hat{\sigma}_t^{n_t}\sigma_\pi\sigma_{*t}(2\pi)^{(n_t/2+2)}\prod_{j=1}^{n_t}\bar{S}_{tj}}\exp\left(-\frac{1}{2}\left(\frac{\mu_t-\mu_{*t}}{\sigma_{*t}}\right)^2\right) \tag{4.18}$$

with

$$\sigma_{*t}^2 = \frac{\hat{\sigma}_t^2\sigma_\pi^2}{n_t\sigma_\pi^2+\hat{\sigma}_t^2},$$

$$\mu_{*t} = \frac{\hat{\sigma}_t^2}{n_t\sigma_\pi^2+\hat{\sigma}_t^2}\mu_\pi + \frac{n_t\sigma_\pi^2}{n_t\sigma_\pi^2+\hat{\sigma}_t^2}\frac{1}{n_t}\sum_{j=1}^{n_t}\log(\bar{S}_{tj}) = \sigma_{*t}^2\left(\frac{\mu_\pi}{\sigma_\pi^2}+\frac{\sum_{j=1}^{n_t}\log(\bar{S}_{tj})}{\hat{\sigma}_t^2}\right).$$

The integrated likelihood for the tree $\mathcal{T}$ is thus given by

$$p_{\text{LN}}\left(\bar{\boldsymbol{S}} \mid \boldsymbol{X}, \hat{\boldsymbol{\sigma}}, \mathcal{T}\right) = \prod_{t=1}^{b} p_{\text{LN}}\left(\bar{\boldsymbol{S}}_t \mid \boldsymbol{X}_t, \hat{\sigma}_t\right). \tag{4.19}$$

Next, we discuss the DIC for this tree. Since we only consider uncertainty for $\boldsymbol{\mu}$ but not for $\boldsymbol{\sigma}$ without data augmentation involved, we can use the DIC proposed in Section 4.2. It follows that

$$\text{DIC}_t = D(\bar{\mu}_t) + 2s_{Dt},$$

where

$$\begin{aligned} D\left(\bar{\mu}_t\right) &= -2\sum_{j=1}^{n_t}\log f_{\text{LN}}(\bar{S}_{tj} \mid \bar{\mu}_t, \hat{\sigma}_t) \\ &= -2\sum_{j=1}^{n_t}\left(-\frac{(\log(\bar{S}_{tj})-\bar{\mu}_t)^2}{2\hat{\sigma}_t^2} - \log(\bar{S}_{tj}\hat{\sigma}_t\sqrt{2\pi})\right), \end{aligned} \tag{4.20}$$

with

$$\bar{\mu}_t = \mu_{*t} = \sigma_{*t}^2\left(\frac{\mu_\pi}{\sigma_\pi^2}+\frac{\sum_{j=1}^{n_t}\log(\bar{S}_{tj})}{\hat{\sigma}_t^2}\right), \tag{4.21}$$

and the effective number of parameters s_{Dt} is given by

$$\begin{aligned} s_{Dt} &= 1 + 2\sum_{j=1}^{n_t}\left\{\log(f_{\text{LN}}(\bar{S}_{tj} \mid \bar{\mu}_t, \hat{\sigma}_t) - \mathbb{E}_{\text{post}}\left[\log(f_{\text{LN}}\left(\bar{S}_{tj} \mid \mu_t, \hat{\sigma}_t\right))\right]\right\} \\ &= 1 + \sum_{j=1}^{n_t}\frac{\sigma_{*t}^2}{\hat{\sigma}_t^2} = 1 + \frac{n_t\sigma_\pi^2}{n_t\sigma_\pi^2+\hat{\sigma}_t^2}, \end{aligned} \tag{4.22}$$

check the calculation here!!! and thus

$$\begin{aligned} \text{DIC}_t &= D\left(\bar{\mu}_t\right) + 2s_{Dt} \\ &= -2\sum_{j=1}^{n_t}\left(-\frac{(\log(\bar{S}_{tj})-\bar{\mu}_t)^2}{2\hat{\sigma}_t^2} - \log(\bar{S}_{tj}\hat{\sigma}_t\sqrt{2\pi})\right) + 2 + \frac{2n_t\sigma_\pi^2}{n_t\sigma_\pi^2+\hat{\sigma}_t^2}. \end{aligned}$$

Table 4.2: Evaluation metrics for LN-BCART. ϵ_t denotes the empirical claims severity in node t, computed as $\sum_{j=1}^{n_t} S_{tj}/\sum_{j=1}^{n_t} N_{tj}$. $\hat{\mu}_t$ is the parameter estimation that is obtained from (4.21) and $\hat{\sigma}_t$ is obtained by using MME (see Remark 4.2 (a)).

	Formulas
RSS($\bar{\boldsymbol{S}}$)	$\sum_{t=1}^{b}\sum_{j=1}^{n_t}\left(\bar{S}_{tj}-\exp(\bar{\mu}_t+\hat{\sigma}_t^2/2)\right)^2$
SE	$\sum_{t=1}^{b}\left(\epsilon_t-\exp(\bar{\mu}_t+\hat{\sigma}_t^2/2)\right)^2$
DS	$\sum_{t=1}^{b}\frac{\left(\epsilon_t-\exp(\bar{\mu}_t+\hat{\sigma}_t^2/2)\right)^2}{(\exp(\hat{\sigma}_t^2)-1)\exp(2\bar{\mu}_t+\hat{\sigma}_t^2)}$

Then the DIC of the tree $\mathcal{T}$ is obtained by using (4.16). With the formulas derived above for the LogNormal case, we can use the three-step approach proposed in Subsection 2.2.4, together with Algorithm 4.1 in Section 4.2 (treat $\boldsymbol{\theta_M}=\boldsymbol{\sigma}$ and $\boldsymbol{\theta_B}=\boldsymbol{\mu}$), to search for an optimal tree which can then be used to predict new data. Similarly, some evaluation metrics based on LogNormal distribution are provided in Table 4.2.

Remark 4.2 *(a) To obtain σ_t upfront, we can solve (4.4) and (4.5). However, there is no explicit solution. One approach is to transform it into an optimization problem by introducing a loss function:*

$$L=\left(\mathbb{E}(\bar{S}_i\mid\mu,\sigma)-\exp(\mu+\sigma^2/2)\right)^2+\Big(Var(\bar{S}_i\mid\mu,\sigma)-(\exp(\sigma^2)-1)\exp\left(2\mu+\sigma^2\right)\Big)^2.$$

Similar to ZIP models (see Subsection 3.3.5), we can use software R to solve this optimization problem by using the package `optimx`.

(b) It is obvious to see that $s_{Dt}\to 2$ as $n_t\to\infty$. This explains the name of the effective number of parameters in the Bayesian framework, as 2 is the number of parameters in the terminal node t for the LogNormal model if a flat prior is assumed for μ_t, and σ_t is assumed known.

(c) There are other ways to deal with the LogNormal distribution in the Bayesian framework by treating different parameters as known and assuming corresponding conjugate priors. For example, a Normal inverse-Gamma joint prior can be used for the parameters μ and σ^2; see, e.g., Fink (1997).

4.4 Weibull-Bayesian CART

Consider a tree $\mathcal{T}$ with b terminal nodes as before. In the Weibull model, we assume that $\bar{S}_i \mid \boldsymbol{x}$ follows a Weibull distribution. As in the previous sections, we choose the inverse Gamma prior for β_t with hyper-parameters $\alpha_\pi, \beta_\pi > 0$ to obtain the posterior distribution in a closed form, that is,

$$p(\beta_t) = \frac{\beta_\pi^{\alpha_\pi}}{\Gamma(\alpha_\pi)} \beta_t^{-\alpha_\pi - 1} \exp(-\beta_\pi/\beta_t), \tag{4.23}$$

and treat α_t as known (using MME to estimate). With the above inverse Gamma prior and the estimated parameter $\hat{\alpha}_t$, the integrated likelihood for terminal node t can be obtained as

$$\begin{aligned}
&p_{\text{Weib}}\left(\bar{\boldsymbol{S}}_t \mid \boldsymbol{X}_t\right) \\
&= \int_0^\infty f_{\text{Weib}}\left(\bar{\boldsymbol{S}}_t \mid \hat{\alpha}_t, \beta_t\right) p(\beta_t) d\beta_t \\
&= \int_0^\infty \prod_{j=1}^{n_t} \frac{\hat{\alpha}_t}{\beta_t} \bar{S}_{tj}^{\hat{\alpha}_t - 1} \exp(-\bar{S}_{tj}^{\hat{\alpha}_t}/\beta_t) \frac{\beta_\pi^{\alpha_\pi}}{\Gamma(\alpha_\pi)} \beta_t^{-\alpha_\pi - 1} \exp(-\beta_\pi/\beta_t) d\beta_t \\
&= \frac{\beta_\pi^{\alpha_\pi} \hat{\alpha}_t^{n_t} \prod_{j=1}^{n_t} \bar{S}_{tj}^{\hat{\alpha}_t - 1}}{\Gamma(\alpha_\pi)} \beta_t^{-n_t - \alpha_\pi - 1} \exp\left(-\frac{1}{\beta_t}\left(\sum_{j=1}^{n_t} \bar{S}_{tj}{}^{\hat{\alpha}_t} + \beta_\pi\right)\right) d\beta_t \\
&= \frac{\beta_\pi^{\alpha_\pi} \hat{\alpha}_t^{n_t} \prod_{j=1}^{n_t} \bar{S}_{tj}^{\hat{\alpha}_t - 1}}{\Gamma(\alpha_\pi)} \frac{\Gamma(n_t + \alpha_\pi)}{(\sum_{j=1}^{n_t} \bar{S}_{tj}^{\hat{\alpha}_t} + \beta_\pi)^{n_t + \alpha_\pi}}.
\end{aligned} \tag{4.24}$$

Clearly, from (4.24), we see that the posterior distribution of β_t, conditional on $\bar{\boldsymbol{S}}_t$, is given by

$$\beta_t \mid \bar{\boldsymbol{S}}_t \quad \sim \quad \text{Inverse Gamma} \left(n_t + \alpha_\pi, \sum_{j=1}^{n_t} \bar{S}_{tj}^{\hat{\alpha}_t} + \beta_\pi\right).$$

The integrated likelihood for the tree $\mathcal{T}$ is thus given by

$$p_{\text{Weib}}\left(\bar{\boldsymbol{S}} \mid \boldsymbol{X}, \hat{\boldsymbol{\alpha}}, \mathcal{T}\right) = \prod_{t=1}^{b} p_{\text{Weib}}\left(\bar{\boldsymbol{S}}_t \mid \boldsymbol{X}_t, \hat{\alpha}_t\right). \tag{4.25}$$

Next, we discuss the DIC for this tree. Since we only consider uncertainty for $\boldsymbol{\beta}$ but not for $\boldsymbol{\alpha}$ without data augmentation involved, we can still use the DIC proposed in Section 4.2. It follows that

$$\text{DIC}_t = D(\bar{\beta}_t) + 2s_{Dt},$$

where

$$\begin{aligned} D(\bar{\beta}_t) &= -2\sum_{j=1}^{n_t} \log f_{\text{Weib}}(\bar{S}_{tj} \mid \hat{\alpha}_t, \bar{\beta}_t) \\ &= -2\sum_{j=1}^{n_t} \Big(\log(\hat{\alpha}_t) - \log(\bar{\beta}_t) + (\hat{\alpha}_t - 1)\log(\bar{S}_{tj}) - \bar{S}_{tj}^{\hat{\alpha}_t}/\bar{\beta}_t\Big), \end{aligned} \tag{4.26}$$

with

$$\bar{\beta}_t = \frac{\sum_{j=1}^{n_t} \bar{S}_{tj}{}^{\hat{\alpha}_t} + \beta_\pi}{n_t + \alpha_\pi - 1}, \tag{4.27}$$

and the effective number of parameters s_{Dt} is given by

$$\begin{aligned} s_{Dt} &= 1 + 2\sum_{j=1}^{n_t} \left\{\log(f_{\text{Weib}}(\bar{S}_{tj} \mid \hat{\alpha}_t, \bar{\beta}_t) - \mathbb{E}_{\text{post}}\left[\log\left(f_{\text{Weib}}(\bar{S}_{tj} \mid \hat{\alpha}_t, \beta_t)\right)\right]\right\} \\ &= 1 + 2\sum_{j=1}^{n_t} \left(\log(n_t + \alpha_\pi - 1) - \psi(n_t + \alpha_\pi) + \frac{\bar{S}_{tj}^{\hat{\alpha}_t}}{\sum_{j=1}^{n_t} \bar{S}_{tj}^{\hat{\alpha}_t} + \beta_\pi}\right), \end{aligned} \tag{4.28}$$

where we use the fact that

$$\mathbb{E}_{\text{post}}\left(\log(\beta_t)\right) = \log\left(\sum_{j=1}^{n_t} \bar{S}_{tj}{}^{\hat{\alpha}_t} + \beta_\pi\right) - \psi(n_t + \alpha_\pi),$$

$$\mathbb{E}_{\text{post}}\left(1/\beta_t\right) = \frac{n_t + \alpha_\pi}{\sum_{j=1}^{n_t} \bar{S}_{tj}{}^{\hat{\alpha}_t} + \beta_\pi}.$$

Thus

$$\begin{aligned} \text{DIC}_t &= D(\bar{\beta}_t) + 2s_{Dt} \\ &= -2\sum_{j=1}^{n_t} \Big(\log(\hat{\alpha}_t) - \log(\bar{\beta}_t) + (\hat{\alpha}_t - 1)\log(\bar{S}_{tj}) - \bar{S}_{tj}^{\hat{\alpha}_t}/\bar{\beta}_t\Big) \\ &\quad + 2 + 4\sum_{j=1}^{n_t} \left(\log(n_t + \alpha_\pi - 1) - \psi(n_t + \alpha_\pi) + \frac{\bar{S}_{tj}^{\hat{\alpha}_t}}{\sum_{j=1}^{n_t} \bar{S}_{tj}^{\hat{\alpha}_t} + \beta_\pi}\right). \end{aligned}$$

Then the DIC of the tree $\mathcal{T}$ is obtained by using (4.16). With the formulas derived above for the Weibull case, we can use the three-step approach proposed in Subsection 2.2.4, together with Algorithm 4.1 in Section 4.2 (treat $\boldsymbol{\theta_M} = \boldsymbol{\alpha}$ and $\boldsymbol{\theta_B} = \boldsymbol{\beta}$), to search for an optimal tree which can then be used to predict new data. Similarly, the formulas for some of the evaluation metrics based on Weibull distribution are provided in Table 4.3.

Remark 4.3 *(a) Similar to Section 4.3, to obtain α_t upfront, we use software R to solve these two equations ((4.7) and (4.8)) by using the package* `optimx`.

(b) Obviously, $s_{Dt} \to 2$ as $n_t \to \infty$, which is exactly the effective number of parameters in the terminal node t for the Weibull model if a flat prior is assumed for β_t, and α_t is assumed known.

Table 4.3: Evaluation metrics for Weib-BCART. ϵ_t denotes the empirical claims severity in node t, computed as $\sum_{j=1}^{n_t} S_{tj} / \sum_{j=1}^{n_t} N_{tj}$. $\hat{\beta}_t$ is the parameter estimation that can be obtained from (4.27); $\hat{\alpha}_t$ can be obtained by using MME (see Remark 4.3 (a)).

	Formulas
RSS($\boldsymbol{S}$)	$\sum_{t=1}^{b} \sum_{j=1}^{n_t} \left(\bar{S}_{tj} - \bar{\beta}_t \Gamma(1+1/\hat{\alpha}_t)\right)^2$
SE	$\sum_{t=1}^{b} \left(\epsilon_t - \bar{\beta}_t \Gamma(1+1/\hat{\alpha}_t)\right)^2$
DS	$\sum_{t=1}^{b} \frac{\left(\epsilon_t - \bar{\beta}_t \Gamma(1+1/\hat{\alpha}_t)\right)^2}{\bar{\beta}_t^2 \left[\Gamma\left(1+2/\hat{\alpha}_t\right) - \left(\Gamma\left(1+1/\hat{\alpha}_t\right)\right)^2\right]}$

There are many other distributions that can also be used to model claims severity, such as Pareto, generalized Gamma, generalized Pareto distributions, and so on. However, they either have too many parameters or are challenging to make explicit calculations in the Bayesian framework. We believe further research into the selection of these generalized distributions could still be explored; see, e.g., Mehmet & Saykan (2005), Shi *et al.* (2015) and Farkas *et al.* (2021).

4.5 A Simulation Example: Weibull Data with Varying Shape Parameters

This section aims to examine how different BCART models for claims severity introduced in previous sections can capture the tail feature of simulated data. To achieve this, we shall use the Weibull distribution to generate data. By tuning the shape parameter of the Weibull distribution, we can control the tail. Simulate a data set $\{(\boldsymbol{x}_i, \bar{S}_i)\}_{i=1}^{n}$ with $n = 5,000$ independent observations. Here $\boldsymbol{x}_i = (x_{i1}, x_{i2})$, with independent components $x_{ik} \sim N(0,1)$ for $k = 1, 2$. Moreover, $\bar{S}_i \sim \text{Weib}(\alpha, \beta(x_{i1}, x_{i2}))$, where

$$\beta(x_1, x_2) = \begin{cases} 50 & \text{if } x_1 x_2 \leq 0, \\ 200 & \text{if } x_1 x_2 > 0, \end{cases}$$

and α is the shape parameter of the Weibull distribution, which is to be varied and specified later. The value for the rate parameter β is chosen to be 50 (or 200)

Table 4.4: Statistics summary for simulated data with different values of the shape parameter α.

	$\alpha = 0.5$	$\alpha = 2$
Mean	255	111
Median	47	74
Max	12539	554
Standard Deviation	657	94
Skewness	7	1
Kurtosis	78	4

to keep the average claim amount $\bar{S}_i$ around 200, which is close to the situation in real data. The data is split into two subsets: a training set with $n - m = 4,000$ observations and a test set with $m = 1,000$ observations. In this case, the goal is to investigate the performance of the Gamma-BCART, LN-BCART, and Weib-BCART when α is varied, leading to different tail characteristics (heavy and light tails). Clearly, Weib-BCART should be the best among these for two reasons: First, the data is generated using the Weibull distribution. Second, in tree models, which involve many groups (terminal nodes), Weib-BCART is flexible to accommodate data in different groups with varying tail features by tuning the shape parameter. Besides, intuition tells us that when $\alpha < 1$, LN-BCART is expected to outperform Gamma-BCART in capturing the heavy tail. Conversely, when $\alpha > 1$, Gamma-BCART is anticipated to be sufficiently effective for the light-tailed data. This study will support this intuition. For simplicity, we shall present two results, one with $\alpha = 0.5$, and the other with $\alpha = 2$. The statistics summary for data simulated using different values of α is provided in Table 4.4. High positive skewness (right-skewed) means that more values are concentrated on the left side (tail) of the distribution, while the right tail of the distribution graph is longer, and high kurtosis indicates that the distribution has more values in the tails, confirming that the smaller the α, the heavier the tail.

We begin our discussion by simulating data with a heavy tail (i.e., $\alpha = 0.5$). We provide the hyper-parameters γ, ρ used to achieve MCMC convergence to the region of trees with a specific number of terminal nodes in Table 4.5. The effective number of parameters and DIC of the optimal trees for each model are shown in the last two columns, respectively. We can conclude from the DIC that by using Step 3 in Table 2.1, we can choose the optimal tree with the true 4 terminal nodes for either Gamma-BCART, LN-BCART, or Weib-BCART, and among these, the

Table 4.5: Hyper-parameters, s_D and DIC on training data (shape parameter $\alpha = 0.5$). The number in the bracket after the abbreviation of the model indicates the number of terminal nodes for this tree. Bold font indicates DIC selected model.

Model	γ	ρ	s_D	DIC
Gamma-BCART (2)	0.50	15	3.97	54882
Gamma-BCART (3)	0.95	15	5.97	54641
Gamma-BCART (4)	0.99	12	7.96	**54303**
Gamma-BCART (5)	0.99	10	9.95	54437
LN-BCART (2)	0.50	10	3.99	54262
LN-BCART (3)	0.95	10	5.98	53960
LN-BCART (4)	0.99	10	7.97	**53501**
LN-BCART (5)	0.99	8	9.96	53708
Weib-BCART (2)	0.50	10	4.00	54189
Weib-BCART (3)	0.90	10	5.99	53843
Weib-BCART (4)	0.95	10	7.99	**53373**
Weib-BCART (5)	0.99	10	9.98	53629

Weib-BCART (with DIC=53373) is the best one. This is consistent with our expectation, and the fact that there is a smaller difference between Weib-BCART and LN-BCART than the difference between LN-BCART and Gamma-BCART shows that LN-BCART has somewhat captured the heavy tail. Now, let us look at how well these models perform on test data in Table 4.6. First, it is not surprising that the best SE (0.365, 0.331, and 0.323 respectively) and DS (5.36×10^{-6}, 5.11×10^{-6}, and 5.02×10^{-6} respectively) are obtained by the optimal tree with 4 terminal nodes for each type of model, i.e., Gamma, LogNormal, and Weibull, which is understandable given that these models retrieve almost true tree structures. Second, RSS($\bar{\boldsymbol{S}}$), NLL, and lift show that, for each type of model, performance improves as the number of terminal nodes rises, however, the amount of improvement becomes smaller after the optimal trees with 4 terminal nodes have been attained (see more discussion in Section 2.3). Third, we observe that the Weib-BCART, among these three trees with 4 terminal nodes, performs the best on test data based on all these evaluation metrics, which is in line with the conclusion from training data.

Next, we consider the simulation with a light tail (i.e., $\alpha = 2$). Tables 4.7 and 4.8 show the results. For this case, similar discussions can be considered. We discover that the performance order based on DIC is Weib-BCART > Gamma-

Table 4.6: Model performance on test data (shape parameter $\alpha = 0.5$) with bold entries determined by DIC (see Table 4.5). The number in the bracket after the abbreviation of the model indicates the number of terminal nodes for this tree.

Model	RSS($\bar{\boldsymbol{S}}$) (in 10^8)	SE	DS (in 10^{-6})	NLL	Lift
Gamma-BCART (2)	1.078	0.395	6.14	13305	1.12
Gamma-BCART (3)	1.043	0.383	5.89	13153	2.34
Gamma-BCART (4)	0.995	**0.365**	**5.36**	12869	3.73
Gamma-BCART (5)	0.994	0.377	5.54	12756	3.80
LN-BCART (2)	1.052	0.385	5.92	13115	1.52
LN-BCART (3)	1.009	0.366	5.68	13005	2.78
LN-BCART (4)	0.989	**0.331**	**5.11**	12699	3.79
LN-BCART (5)	0.984	0.343	5.28	12601	3.85
Weib-BCART (2)	1.044	0.382	5.85	13025	1.60
Weib-BCART (3)	1.000	0.359	5.61	12916	2.93
Weib-BCART (4)	0.986	**0.323**	**5.02**	12605	3.87
Weib-BCART (5)	0.983	0.339	5.14	12528	3.92

BCART > LN-BCART, which is also consistent with their performance on test data. To avoid duplication of content, we omit the detailed analysis here.

Additionally, we run several other simulation examples to compare the performance of Gamma-BCART, LN-BCART, and Weib-BCART with various values of α. We draw the following conclusions from these simulations: LN-BCART outperforms Gamma-BCART when the data has a heavy tail (reflected by a small α), and vice versa; Besides, Weib-BCART is preferred to both Gamma-BCART and LN-BCART because it can flexibly handle data in groups with different tail properties by tuning the shape parameter. This finding is supported by the real insurance data analyses provided in Chapter 6.

4.6 Summary of Chapter 4

The use of several distributions (Gamma, LogNormal, and Weibull) in BCART models for claims severity analysis is covered in this chapter. We found that the Weib-BCART is the best model among these three, capable of handling cases where some groups have lighter tails, and others have heavier tails. Besides, in the comparison between Gamma-BCART and LN-BCART, the former is preferable

Table 4.7: Hyper-parameters, s_D and DIC on training data (shape parameter $\alpha = 2$). The number in the bracket after the abbreviation of the model indicates the number of terminal nodes for this tree. Bold font indicates DIC selected model.

Model	γ	ρ	s_D	DIC
LN-BCART (2)	0.50	12	3.99	43014
LN-BCART (3)	0.95	12	5.97	42891
LN-BCART (4)	0.99	12	7.96	**42688**
LN-BCART (5)	0.99	10	9.95	42742
Gamma-BCART (2)	0.50	10	3.99	42807
Gamma-BCART (3)	0.95	10	5.98	42641
Gamma-BCART (4)	0.99	10	7.98	**42382**
Gamma-BCART (5)	0.99	8	9.96	42459
Weib-BCART (2)	0.50	10	4.00	42729
Weib-BCART (3)	0.90	10	5.99	42520
Weib-BCART (4)	0.95	10	7.99	**42198**
Weib-BCART (5)	0.99	10	9.97	42301

for data with a lighter tail and the latter is more suitable for data with a heavier tail. This finding provides us with a practical strategy for choosing models, which will be further demonstrated in real data analyses in Chapter 6.

Table 4.8: Model performance on test data (shape parameter $\alpha = 2$) with bold entries determined by DIC (see Table 4.7). The number in the bracket after the abbreviation of the model indicates the number of terminal nodes for this tree.

Model	RSS($\bar{\boldsymbol{S}}$) (in 10^6)	SE	DS (in 10^{-8})	NLL	Lift
LN-BCART (2)	2.71	0.0001958	3.619	5513	1.23
LN-BCART (3)	2.59	0.0001934	3.496	5385	2.41
LN-BCART (4)	2.08	**0.0001879**	**3.336**	5143	3.75
LN-BCART (5)	1.94	0.0001890	3.402	5012	3.82
Gamma-BCART (2)	2.32	0.0001897	3.503	5366	1.63
Gamma-BCART (3)	2.14	0.0001880	3.358	5157	2.81
Gamma-BCART (4)	1.69	**0.0001795**	**3.112**	4916	3.81
Gamma-BCART (5)	1.54	0.0001805	3.266	4843	3.87
Weib-BCART (2)	2.23	0.0001889	3.467	5315	1.65
Weib-BCART (3)	1.99	0.0001868	3.302	5120	2.85
Weib-BCART (4)	1.60	**0.0001785**	**3.075**	4888	3.85
Weib-BCART (5)	1.48	0.0001793	3.212	4791	3.90

Chapter 5

Aggregate Claims Modelling with Bayesian CART

Unlike the previous two chapters which model claims frequency and claims severity separately, this chapter introduces probability models to describe the aggregate (total) claim amount $S_i = \sum_{j=1}^{N_i} Y_{ij}$ for each policyholder i ($i = 1, 2, \ldots, n$), where both N_i and $Y_{ij} (j = 1, \ldots, N_i)$ given $N_i > 0$ are random. First, we present two types of models, frequency-severity models (see Omari *et al.* (2018) and Mehmet & Saykan (2005)) and sequential models, both of which use two trees for claims frequency and claims severity respectively. The former considers claims frequency and claims severity independently, so the order in which they are modelled has no influence. The latter, as the name suggests, is affected by the order of the claims frequency and claims severity modelling. A common approach is to first model the claims frequency and then treat the number of claims N_i as a covariate in the claims severity modelling to address the dependence between the number of claims and claims severity. This strategy has gained popularity due to the increased focus on the dependence in aggregate claims modelling. Recent studies have explored it extensively; see, e.g., Garrido *et al.* (2016), Shi *et al.* (2015) and Frees *et al.* (2016). We propose to apply this strategy in BCART models. Following this, we introduce a third model, joint model, which utilize Compound Poisson Gamma (CPG) and Zero-Inflated Compound Poisson Gamma (ZICPG) distributions for bivariate response (number of claims and aggregate claim amount) modelling; see, e.g., Smyth & Jørgensen (2002) and Quijano Xacur & Garrido (2015). Particularly, for ZICPG distributions, we employ the data augmentation technique (see Murray (2021)) and explore different ways to embed the exposure, as discussed in Chapter 3. In contrast to the previous two models, joint models construct one joint tree for (N_i, S_i) to directly model the aggregate claim amount S_i, thereby avoiding

the need to look for two trees. Two simulation examples are provided for specific illustrative purposes. Additionally, to facilitate the comparison between two trees and one joint tree, we will employ the evaluation metrics introduced in Section 2.3. Specific details on their application in the case of two trees are provided.

5.1 Frequency-Severity Models

This section describes a standard model of insurance claims, which independently models claims frequency and claims severity using two trees. The ultimate goal of insurance pricing is to estimate the premium. Under the assumption that claims frequency and claims severity are independent, the pure premium can be calculated as:

$$\text{Pure Premium} = \text{Claims Frequency} \times \text{Claims Severity}.$$

In Chapters 3 and 4, we introduced BCART models for both claims frequency and claims severity separately. These models can be directly used for the premium calculation.

In Chapter 4, we model $\bar{S}_i$ using three different distributions (Gamma, Log-Normal, and Weibull). This section proposes another way that uses the Gamma distribution to model the individual claim amount Y_{ij} instead of modelling $\bar{S}_i$ directly, following Henckaerts *et al.* (2021) and Frees *et al.* (2014) (also see Remark 4.1 (b) in Section 4.2). Assume the insurance policyholder $i = 1, 2, \ldots, n$ have independent claim amounts Y_{ij} $(j = 1, 2, \ldots, N_i)$ given $N_i > 0$. And they follow a common Gamma distribution with parameters $\alpha > 0$ and $\beta > 0$. Based on the additive property of the Gamma distribution, we can obtain the distribution for $\bar{S}_i$, i.e.,

$$\bar{S}_i \sim \text{Gamma}\left(N_i\alpha(\boldsymbol{x_i}), N_i\beta(\boldsymbol{x_i})\right),$$

which incorporates the number of claims N_i into the parameters as model weights for claims severity modelling, in contrast to Section 4.2.

Consider a tree $\mathcal{T}$ with b terminal nodes, following a similar procedure to Section 4.2, for terminal node t, we denote the associated data as $(\boldsymbol{X}_t, \boldsymbol{N}_t, \bar{\boldsymbol{S}}_{\boldsymbol{t}}) = ((X_{t1}, N_{t1}, \bar{S}_{t1}), \ldots, (X_{tn_t}, N_{tn_t}, \bar{S}_{tn_t}))^\top$. We then have

$$f_{\text{G}}\left(\bar{S}_{tj} \mid N_{tj}, \alpha_t, \beta_t\right) = \frac{(N_{tj}\beta_t)^{N_{tj}\alpha_t}\bar{S}_{tj}^{N_{tj}\alpha_t - 1}e^{-N_{tj}\beta_t\bar{S}_{tj}}}{\Gamma(N_{tj}\alpha_t)} \tag{5.1}$$

for the j-th observation such that $\boldsymbol{x}_i \in \mathcal{A}_t$, where N_{tj} can be obtained directly from the data within the node. The mean and variance of $\bar{S}_{tj}$ are given by

$$\mathbb{E}(\bar{S}_{tj} \mid \alpha_t, \beta_t) = \frac{\alpha_t}{\beta_t}, \qquad \text{Var}(\bar{S}_{tj} \mid N_{tj}, \alpha_t, \beta_t) = \frac{\alpha_t}{N_{tj}\beta_t^2}. \tag{5.2}$$

Similar to Section 4.2, we choose the Gamma prior for β_t with hyper-parameters $\alpha_\pi, \beta_\pi > 0$ (cf. (3.2)) and treat α_t as known (using MME to estimate), i.e.,

$$\hat{\alpha}_t = \frac{(\bar{S})_t^2}{\mathrm{Var}(\bar{S})_t \bar{N}_t}. \tag{5.3}$$

where $(\bar{S})_t$ and $\mathrm{Var}(\bar{S})_t$ have the same meaning as in Section 4.2, and $\bar{N}_t$ denotes the average claim number in the t-th node. With the above Gamma prior and the estimated parameter $\hat{\alpha}_t$, the integrated likelihood for terminal node t can be obtained as

$$\begin{aligned}
&p_{\mathrm{G}}(\bar{\boldsymbol{S}}_t \mid \boldsymbol{X}_t, \boldsymbol{N}_t) \\
&= \int_0^\infty f_{\mathrm{G}}(\bar{\boldsymbol{S}}_t \mid \boldsymbol{N}_t, \hat{\alpha}_t, \beta_t) p(\beta_t) d\beta_t \\
&= \int_0^\infty \prod_{j=1}^{n_t} \frac{(N_{tj}\beta_t)^{N_{tj}\hat{\alpha}_t} \bar{S}_{tj}^{N_{tj}\hat{\alpha}_t - 1} e^{-N_{tj}\beta_t \bar{S}_{tj}}}{\Gamma(N_{tj}\hat{\alpha}_t)} \frac{\beta_\pi^{\alpha_\pi} \beta_t^{\alpha_\pi - 1} e^{-\beta_\pi \beta_t}}{\Gamma(\alpha_\pi)} d\beta_t \\
&= \frac{\beta_\pi^{\alpha_\pi} \prod_{j=1}^{n_t} N_{tj}^{N_{tj}\hat{\alpha}_t} \bar{S}_{tj}^{N_{tj}\hat{\alpha}_t - 1}}{\Gamma(\alpha_\pi) \prod_{j=1}^{n_t} \Gamma(N_{tj}\hat{\alpha}_t)} \int_0^\infty \beta_t^{\sum_{j=1}^{n_t} N_{tj}\hat{\alpha}_t + \alpha_\pi - 1} e^{-(\sum_{j=1}^{n_t} N_{tj}\bar{S}_{tj} + \beta_\pi)\beta_t} d\beta_t \\
&= \frac{\beta_\pi^{\alpha_\pi} \prod_{j=1}^{n_t} N_{tj}^{N_{tj}\hat{\alpha}_t} \bar{S}_{tj}^{N_{tj}\hat{\alpha}_t - 1}}{\Gamma(\alpha_\pi) \prod_{j=1}^{n_t} \Gamma(N_{tj}\hat{\alpha}_t)} \frac{\Gamma(\sum_{j=1}^{n_t} N_{tj}\hat{\alpha}_t + \alpha_\pi)}{(\sum_{j=1}^{n_t} N_{tj}\bar{S}_{tj} + \beta_\pi)^{\sum_{j=1}^{n_t} N_{tj}\hat{\alpha}_t + \alpha_\pi}}.
\end{aligned} \tag{5.4}$$

Clearly, from (5.4), we see that the posterior distribution of β_t, conditional on data $(\boldsymbol{N}_t, \bar{\boldsymbol{S}}_t)$, is given by

$$\beta_t \mid \boldsymbol{N}_t, \bar{\boldsymbol{S}}_t \quad \sim \quad \mathrm{Gamma}\left(\sum_{j=1}^{n_t} N_{tj}\hat{\alpha}_t + \alpha_\pi, \sum_{j=1}^{n_t} N_{tj}\bar{S}_{tj} + \beta_\pi \right).$$

The integrated likelihood for the tree $\mathcal{T}$ is thus given by

$$p_{\mathrm{G}}\left(\bar{\boldsymbol{S}} \mid \boldsymbol{X}, \boldsymbol{N}, \hat{\boldsymbol{\alpha}}, \mathcal{T}\right) = \prod_{t=1}^{b} p_{\mathrm{G}}\left(\bar{\boldsymbol{S}}_t \mid \boldsymbol{X}_t, \boldsymbol{N}_t, \hat{\alpha}_t\right). \tag{5.5}$$

Now, we discuss the DIC_t for terminal node t of this tree. Similar to Section 4.2, we can derive

$$D(\bar{\beta}_t) = -2 \sum_{j=1}^{n_t} \Big[N_{tj}\hat{\alpha}_t \log(N_{tj}\bar{\beta}_t) + (N_{tj}\hat{\alpha}_t - 1)\log(\bar{S}_{tj}) - \bar{\beta}_t N_{tj}\bar{S}_{tj} - \log\left(\Gamma(N_{tj}\hat{\alpha}_t)\right) \Big], \tag{5.6}$$

where

$$\bar{\beta}_t = \frac{\sum_{j=1}^{n_t} N_{tj}\hat{\alpha}_t + \alpha_\pi}{\sum_{j=1}^{n_t} N_{tj}\bar{S}_{tj} + \beta_\pi}. \tag{5.7}$$

Therefore, a direct calculation shows that the effective number of parameters for terminal node t is given by

$$s_{Dt} = 1 + 2\left(\log\left(\sum_{j=1}^{n_t} N_{tj}\hat{\alpha}_t + \alpha_\pi\right) - \psi\left(\sum_{j=1}^{n_t} N_{tj}\hat{\alpha}_t + \alpha_\pi\right)\right)\sum_{j=1}^{n_t} N_{tj}\hat{\alpha}_t, \quad (5.8)$$

and thus

$$\begin{aligned}
&\mathrm{DIC}_t \\
&= D(\bar{\beta}_t) + 2s_{Dt} \\
&= -2\sum_{j=1}^{n_t}\left[N_{tj}\hat{\alpha}_t \log(N_{tj}\bar{\beta}_t) + (N_{tj}\hat{\alpha}_t - 1)\log(\bar{S}_{tj}) - \bar{\beta}_t N_{tj}\bar{S}_{tj} - \log\left(\Gamma(N_{tj}\hat{\alpha}_t)\right)\right] \\
&\quad + 2 + 4\sum_{j=1}^{n_t}\left(\log\left(\sum_{j=1}^{n_t} N_{tj}\hat{\alpha}_t + \alpha_\pi\right) - \psi\left(\sum_{j=1}^{n_t} N_{tj}\hat{\alpha}_t + \alpha_\pi\right)\right)\sum_{j=1}^{n_t} N_{tj}\hat{\alpha}_t.
\end{aligned}$$

Then the DIC of the tree $\mathcal{T}$ is obtained as

$$\mathrm{DIC} := \sum_{t=1}^{b} \mathrm{DIC}_t. \quad (5.9)$$

With the above formulas derived, we can use the three-step approach proposed in Subsection 2.2.4 to search for an optimal tree for the claims severity, where Algorithm 4.1 in Section 4.2 (treat $\boldsymbol{\theta_M} = \boldsymbol{\alpha}$ and $\boldsymbol{\theta_B} = \boldsymbol{\beta}$) should be used.

Remark 5.1 *(a) There are many different combinations for the frequency-severity models, i.e., any model that appears in Chapter 3 and any model that appears in either Chapter 4 or the newly introduced one above can be used individually.*

(b) One benefit of modelling claims frequency and claims severity using two trees is that the risks associated with each component can be discovered individually. However, it can be challenging to interpret two trees as a whole, since several policyholders may be in the same group for claims frequency but a different group for claims severity.

5.1.1 Evaluation Metrics for Frequency-Severity Models

Some evaluation metrics introduced in Section 2.3 remain applicable in the case of two trees. However, the process of obtaining the predicted premium for a new observation in the context of two trees in frequency-severity models (and subsequent sequential models in Section 5.2) needs some further discussion. With the

ultimate goal of estimating the premium in mind, RSS($\boldsymbol{S}$) and NLL (see Section 2.3) can be easily employed. Given the independence assumption between claims frequency and severity in the frequency-severity models, RSS($\boldsymbol{S}$) is straightforwardly obtained by multiplying $\hat{N}_i$ and $\hat{\tilde{S}}_i$, i.e., $\hat{S}_i = \hat{N}_i\hat{\tilde{S}}_i$, where $\hat{N}_i$ is obtained from the claims frequency tree and $\hat{\tilde{S}}_i$ is obtained from the claims severity tree. Similarly, NLL can be obtained by summing up the corresponding NLLs from the two trees. Both RSS($\boldsymbol{S}$) and NLL focus on the partitioned data itself without considering the tree structure, while other comparison indexes (SE, DS, and lift) are tree structure dependent. Although it is possible to combine two trees to obtain a joint partition, if both separate trees are already large, the process of deriving their combined partition becomes significantly complex. Therefore, we do not apply them in the subsequent comparisons. From another perspective, two additional indicators, time and memory usage, can be employed to examine the computational efficiency. For the frequency-severity models, time and memory usage are determined by the summation of corresponding values from the two trees.

5.2 Sequential Models

Although the traditional approach of considering claims frequency and claims severity models separately, as discussed in the previous section, can simplify the problem, it is more realistic to consider the dependence between the number of claims and claims severity. There are two widely discussed strategies to address this issue. One is to use a mixed copula to jointly model the discrete variable of claim count and the continuous variable of claim amount (see, e.g., Czado *et al.* (2012), Song *et al.* (2009) and Gao & Li (2023)), which is very flexible since there are many different copulas that can be chosen. Besides, there is a dedicated parameter in the copula that can be used to model the dependence structure specifically. However, the mixed copulas are difficult to apply in the Bayesian framework. Therefore, we do not consider this strategy in this thesis. Another strategy is to include the number of claims N_i as a covariate in the claims severity model to formulate a conditional severity model; see, e.g., Garrido *et al.* (2016). In this section, we shall follow this strategy in the BCART models, where the claim count N_i is treated as a covariate (also treated as model weights in some cases; see Section 5.1) in the claims severity tree, keeping everything else the same as the frequency-severity models in the previous section.

There are usually two ways to consider N_i as a covariate in claims severity modelling: either directly use N_i as a numeric covariate (see Garrido *et al.* (2016)) or treat N_i as a factor with different levels (see Gschlößl & Czado (2007)). Considering that in real life, it is important to allow premium estimation for new customers when there is no observed claim count N_i, we propose a third way, i.e., use the estimation of claim count $\hat{N}_i$ from the frequency model as a numeric covariate to address this issue.

Using the tower property of probability expectation, it is easily seen that the expectation of the aggregate claim amount $S_i = \sum_{j=1}^{N_i} Y_{ij} = N_i \bar{S}_i$ for an individual policyholder i can be given as (omitting the subscript i for simplicity),

$$\mathbb{E}(S) = \mathbb{E}(N\bar{S}) = \mathbb{E}(\mathbb{E}(N\bar{S} \mid N)) = \mathbb{E}(N\mathbb{E}(\bar{S} \mid N)). \tag{5.10}$$

Because of this formulation, we can estimate the expected claims severity $\mathbb{E}(\bar{S} \mid N)$ using N (or $\hat{N}$) as a covariate, as in Garrido *et al.* (2016).

Since the structures of the claims frequency and claims severity trees are essentially the same as in Chapters 3 and 4, except that the claim count N_i (or $\hat{N}_i$) is treated as a predictor variable in the claims severity modelling within the sequential models, we do not repeat the model description here.

Remark 5.2 *(a) If the sequential models do not choose N_i (or $\hat{N}_i$) as a splitting covariate, they would be the same as the frequency-severity models.*

(b) Sequential models also consist of two trees, so the evaluation metrics introduced in Subsection 5.1.1 can be applied directly to these models.

5.2.1 A Simulation Example: Varying Dependencies between the Number of Claims and Claims Severity

Given the similarities and differences between the frequency-severity models and the sequential models, this subsection aims to demonstrate the capability of the sequential models with Bayesian CART to address the dependence between the number of claims and claims severity by using simulated data. The performance of using different forms of N_i (N_i itself and its estimation) within sequential models is also examined. As the current focus is not on comparing different distributions applied for claims frequency and claims severity, which has been extensively discussed in previous Sections 3.5 and 4.5, we consistently employ the Poisson and Gamma distributions for both frequency-severity models and sequential models here for the sake of simplicity. Besides, to simplify the setting and more clearly

reflect the importance of treating N_i (or $\hat{N}_i$) as a covariate in the claims severity modelling, we model $\bar{S}_i$ directly as in Section 4.2, i,e., $\bar{S}_i \sim \text{Gamma}(\alpha, \beta)$, without using N_i as model weights in the Gamma distribution (see Section 5.1). Additionally, since both frequency-severity models and sequential models have the same claims frequency tree, in the following model comparison, we only consider the claims severity tree. The evaluation metrics introduced in Section 2.3 can be directly employed for the comparison between different claims severity trees. In the sequel, we exclusively treat N_i as a numeric variable. In real data, N_i typically encompasses a wide range of distinct values, leading to numerous levels. This abundance of levels makes it unsuitable for a strategy that transforms the numeric covariate into a categorical one.

To this end, we simulate a data set $\left\{(\boldsymbol{x}_i, v_i, N_i, \bar{S}_i)\right\}_{i=1}^{n}$ with $n = 5,000$ independent observations. Here $\boldsymbol{x}_i = (x_{i1}, x_{i2})$, with independent components $x_{ik} \sim N(0,1)$ for $k = 1, 2$. We assume exposure $v_i \equiv 1$ for simplicity, as it is not a key feature in this context. Moreover, $N_i \sim \text{Poi}(\lambda(x_{i1}, x_{i2})v_i)$, where

$$\lambda\left(x_1, x_2\right) = \begin{cases} 1 & \text{if } x_1 x_2 \leq 0, \\ 7 & \text{if } x_1 x_2 > 0. \end{cases}$$

In this simulation setting, $N_i = 0$ for 901 occurrences, leading to the setting of $\bar{S}_i$ to be 0 directly. For the remaining 4099 cases, $\bar{S}_i$ is generated from a Gamma distribution with a pre-specified and varied dependence parameter ζ, i.e.,

$$\bar{S}_i \mid N_i \sim \text{Gamma}\left(\alpha, \beta_\zeta\right),$$

with

$$\beta_\zeta = 0.001 + \zeta N_i,$$

where α is the shape parameter of the Gamma distribution, and for simplicity, it is fixed at 1 since it is also not a key factor here. The basic value for the rate parameter β is set to 0.001 to maintain the average claim amount $\bar{S}_i$ to be around 500, aligning with real-world scenarios. The data is split into two subsets: a training set with $n - m = 4,000$ observations and a test set with $m = 1,000$ observations. In this case, our goal is to examine how the dependence modulated by ζ influences the performance of both frequency-severity models and sequential models, and the performance of incorporating N_i (or $\hat{N}_i$) into the sequential models. Clearly, if the models choose N_i (or $\hat{N}_i$) as a splitting covariate, it would indicate that the claim count plays an important role in claims severity modelling, and thus sequential models should be preferred.

Table 5.1: Statistics summary and conditional correlation between the number of claims and claims severity for simulated data with different values of the dependence parameter ζ.

	$\zeta = 0$	$\zeta = 0.001$
Mean	828	206
Median	494	92
Max	9713	5138
Standard Deviation	983	324
$\text{Corr}(N, \bar{S} \mid N\bar{S} > 0)$	-0.01	-0.41

Table 5.1 presents the statistics summary and conditional correlation coefficients between the number of claims and claims severity for data sets with different values of ζ. It is obvious that by tuning the value of ζ, the conditional correlation between the number of claims and claims severity varies. For simplicity, we shall only focus on the case where $\zeta = 0.001$, indicating a strong dependence between them. Intuition suggests that sequential models are expected to perform better in capturing strong dependence, and the stronger the dependence, the better the performance of sequential models. In contrast, when there is only a weak dependence (e.g., ζ=0.00001) in the data, the claim count N_i (or $\hat{N}_i$) is unlikely to be selected as a splitting covariate in sequential models, resulting in frequency-severity models and sequential models being the same.

First, regarding model selection on training data in Table 5.2, although we do not have knowledge of the true tree structure and the optimal number of terminal nodes, all models consistently choose five terminal nodes based on DIC, indicating the stability of BCART models in some sense. A more in-depth discussion on the stability will be provided in Chapter 6. Notably, the best performing model is Gamma2-BCART (with DIC=2618), validating our proposed approach of treating $\hat{N}_i$ as a covariate. Moreover, the larger difference between Gamma-BCART and Gamma1-BCART compared to the difference between Gamma1-BCART and Gamma2-BCART suggests that Gamma1-BCART has effectively captured the dependence. Subsequently, we compare the performance on test data in Table 5.3. The conclusion is that based on all these evaluation metrics, Gamma2-BCART with 5 terminal nodes performs the best, which confirms the finding from the training data. In addition, when examining the splitting rules used in the optimal tree for each type of model, both Gamma1-BCART and Gamma2-BCART use N_i

Table 5.2: Hyper-parameters, s_D and DIC on training data (dependence parameter $\zeta = 0.001$). The number in the bracket after the abbreviation of the model indicates the number of terminal nodes for this tree. The Gamma1 and Gamma2 models treat the claim count N_i and $\hat{N}_i$ as a covariate respectively, where $\hat{N}_i$ comes from Poisson-BCART. Bold font indicates DIC selected model.

Model	γ	ρ	s_D	DIC
Gamma-BCART (4)	0.95	10	7.95	2769
Gamma-BCART (5)	0.99	10	9.94	**2716**
Gamma-BCART (6)	0.99	7	11.92	2738
Gamma1-BCART (4)	0.95	10	7.97	2698
Gamma1-BCART (5)	0.99	10	9.96	**2644**
Gamma1-BCART (6)	0.99	7	11.94	2663
Gamma2-BCART (4)	0.95	10	7.98	2682
Gamma2-BCART (5)	0.99	10	9.98	**2618**
Gamma2-BCART (6)	0.99	7	11.97	2635

Table 5.3: Model performance on test data (dependence parameter $\zeta = 0.001$) with bold entries determined by DIC (see Table 5.2). The number in the bracket after the abbreviation of the model indicates the number of terminal nodes for this tree. The Gamma1 and Gamma2 models treat the claim count N_i and $\hat{N}_i$ as a covariate respectively, where $\hat{N}_i$ comes from Poisson-BCART.

Model	RSS($\boldsymbol{S}$) (in 10^5)	SE	DS	NLL	Lift
Gamma-BCART (4)	8.34	0.0927	0.0331	412.83	1.42
Gamma-BCART (5)	8.18	**0.0894**	**0.0309**	409.37	1.85
Gamma-BCART (6)	8.11	0.0904	0.0319	407.55	1.92
Gamma1-BCART (4)	8.20	0.0909	0.0321	408.12	1.61
Gamma1-BCART (5)	8.04	**0.0875**	**0.0297**	403.41	2.06
Gamma1-BCART (6)	7.97	0.0886	0.0305	402.78	2.16
Gamma2-BCART (4)	8.09	0.0903	0.0312	404.96	1.65
Gamma2-BCART (5)	7.91	**0.0866**	**0.0292**	401.13	2.09
Gamma2-BCART (6)	7.83	0.0875	0.0300	400.17	2.18

(or $\hat{N}_i$) in the second split step, and they have similar split values by using N_i and

$\hat{N}_i$ respectively.

We also compare these three models with different values of ζ through several additional simulation examples. Based on these simulations, we obtain the following conclusions: 1) when the conditional correlation coefficient between the number of claims and claims severity is close to zero, both frequency-severity models and sequential models exhibit similar performance. In such cases, we recommend the former since they reduce computation time; 2) When the dependence is stronger, sequential models outperform frequency-severity models by considering the number of claims as a covariate when modelling claims severity; 3) Gamma2-BCART outperforms Gamma1-BCART, and we suspect the reason for this may be that when using the original data N_i itself, it can only be integers and contains a large number of zeros. In contrast, the predicted value $\hat{N}_i$ can take non-integers, providing more possibilities for finding better split values, resulting in better splitting rules and data partitions. The real insurance data analyses presented in Chapter 6 support these conclusions.

Remark 5.3 *In the above simulation study, we only present the results for Gamma2-BCART using Poisson-BCART to obtain the prediction of the claim count N_i. We also assess the performance of using different predicted values of N_i (GLMs and Poisson-CART) in the claims severity modelling. The conclusion is that the more accurate the estimated value for claims frequency used, the better the claims severity tree found, which is consistent with the conclusion obtained in real data analyses (see Chapter 6). Motivated by this observation, we can also use other models to obtain the claims frequency estimation, such as random forests (RF), gradient boosting trees (GBT), neural networks (NN) and so on, which remain to be further explored.*

5.3 Joint Models

Different from the previous two types of BCART models where two separate tree models are used for the frequency and severity, in this section we introduce the third type of BCART models, called *joint* BCART models, where we consider (N, S) as a bivariate response; see Jørgensen & Paes De Souza (1994) and Smyth & Jørgensen (2002) for a similar treatment in generalized linear models. We shall discuss two commonly used distributions for aggregate claim amount S, namely, Compound Poisson Gamma distribution (CPG) and Zero-Inflated Compound Poisson Gamma distribution (ZICPG). The presence of a discrete mass at

zero makes them suitable for modelling aggregate claim amount; see, e.g., Quijano Xacur & Garrido (2015), Yang *et al.* (2018) and Denuit *et al.* (2021). Moreover, we employ the data augmentation technique and explore different ways to embed the exposure in ZICPG models, similar to Chapter 3. The advantage of modelling frequency and severity components separately has been recognized in the literature; see, e.g., Quijano Xacur & Garrido (2015) and Frees *et al.* (2016). In particular, this separate treatment can reflect the situation when the covariates that affect the frequency and severity are very different. However, one disadvantage is that it takes more effort to combine the two resulting models, as we have already seen in Subsection 5.1.1. Compared to the use of two separate tree models, the advantage of joint modelling is that the resulting one tree is easier to interpret which simultaneously gives estimates for frequency, pure premium and thus severity. Additionally, for the situation where frequency and severity are linked through shared covariates, using a parsimony one joint tree model might be advantageous; this will be illustrated in our simulation examples in Subsection 5.3.3. The conclusion from this simulation example can be generalized to a wider field, and some relevant discussions are provided in Linero *et al.* (2020). The comparison between sequential models and joint models is discussed in real data analyses; see Chapter 6.

5.3.1 Compound Poisson Gamma-Bayesian CART

A popular method to model the aggregate claim amount directly is using a Tweedie Compound Poisson distribution; see, e.g., Smyth & Jørgensen (2002). The Tweedie distribution is very flexible, encompassing many different distributions, such as Poisson, Gamma, and Compound Poisson Gamma; see, e.g., Ohlsson & Johansson (2010). Below we shall introduce the Compound Poisson Gamma model in its original form.

Define a Compound Poisson Gamma-distributed random variable

$$S = \sum_{j=1}^{N} Y_j,$$

where

- N follows a Poisson distribution with the parameter $\lambda v > 0$, which is denoted by $\text{Poi}(\lambda v)$; see Section 3.1.
- $Y_1, Y_2, ..., Y_N$, given $N > 0$, are independent and follow a common Gamma distribution with a shape parameter $\alpha > 0$ and a rate parameter $\beta > 0$ denoted by $\text{Gamma}(\alpha, \beta)$; see Section 5.1.

- N and $(Y_1, Y_2, ..., Y_N)$ are independent.

Under these assumptions, we shall model S_i by a *Compound Poisson Gamma* distribution denoted by $CPG(\lambda v_i, \alpha, \beta)$, and assume $S_i, i = 1, 2, \ldots, n$, are IID.

Consider a tree $\mathcal{T}$ with b terminal nodes as before (see Section 2.2). The joint distribution of the CPG model can be derived as

$$\begin{aligned} & f_{\text{CPG}}(N_{tj}, S_{tj} \mid \lambda_t, \alpha_t, \beta_t, v_{tj}) \\ & = f_{\text{P}}(N_{tj} \mid \lambda_t, v_{tj}) f_{\text{G}}(S_{tj} \mid N_{tj}, \alpha_t, \beta_t) \\ & = \begin{cases} e^{-\lambda_t v_{tj}} & (N_{tj}, S_{tj}) = (0, 0), \\ \frac{(\lambda_t v_{tj})^{N_{tj}} e^{-\lambda_t v_{tj}}}{N_{tj}!} \frac{\beta_t^{N_{tj}\alpha_t} S_{tj}^{N_{tj}\alpha_t - 1} e^{-\beta_t S_{tj}}}{\Gamma(N_{tj}\alpha_t)} & (N_{tj}, S_{tj}) \in \mathbb{N} \times \mathbb{R}^+, \end{cases} \end{aligned} \tag{5.11}$$

for the j-th observation such that $\boldsymbol{x}_i \in \mathcal{A}_t$. To explicitly obtain the posterior distribution (see discussions in Subsection 2.2.1), we choose independent conjugate Gamma priors for λ_t and β_t with hyper-parameters $(\alpha^{(\lambda)} > 0, \beta^{(\lambda)} > 0)$ and $(\alpha^{(\beta)} > 0, \beta^{(\beta)} > 0)$ respectively, where the subscript (λ) indicates this hyper-parameter is assigned for the parameter λ, and similarly for (β). Besides, α_t can be estimated and updated by using MME in each step before updating β_t using the posterior distribution, i.e.,

$$\hat{\alpha}_t = \frac{(\bar{S})_t^2}{\text{Var}(\bar{S})_t \bar{N}_t}, \tag{5.12}$$

which is the same as in (5.3). For terminal node t we denote the associated data as $(\boldsymbol{X}_t, \boldsymbol{v}_t, \boldsymbol{N}_t, \boldsymbol{S}_t) = \big((X_{t1}, v_{t1}, N_{t1}, S_{t1}), \ldots, (X_{tn_t}, v_{tn_t}, N_{tn_t}, S_{tn_t})\big)^\top$. With the above Gamma priors and the estimated parameter $\hat{\alpha}_t$, the integrated likelihood for terminal node t can be obtained as

$$\begin{aligned} & p_{\text{CPG}}\left(\boldsymbol{N}_t, \boldsymbol{S}_t \mid \boldsymbol{X}_t, \boldsymbol{v}_t, \hat{\alpha}_t\right) \\ & = \int_0^\infty \int_0^\infty f_{\text{CPG}}(\boldsymbol{N}_t, \boldsymbol{S}_t \mid \lambda_t, \hat{\alpha}_t, \beta_t) p(\lambda_t) p(\beta_t) d\lambda_t d\beta_t \\ & = \int_0^\infty \int_0^\infty \prod_{j:N_{tj}=0} e^{-\lambda_t v_{tj}} \left(\prod_{j:N_{tj}>0} \frac{(\lambda_t v_{tj})^{N_{tj}} e^{-\lambda_t v_{tj}}}{N_{tj}!} \frac{S_{tj}^{N_{tj}\hat{\alpha}_t - 1} e^{-\beta_t S_{tj}} \beta_t^{N_{tj}\hat{\alpha}_t}}{\Gamma(N_{tj}\hat{\alpha}_t)} \right) \\ & \quad \times \frac{\beta^{(\lambda)\alpha^{(\lambda)}} \lambda_t^{\alpha^{(\lambda)}-1} e^{-\beta^{(\lambda)}\lambda_t}}{\Gamma(\alpha^{(\lambda)})} \frac{\beta^{(\beta)\alpha^{(\beta)}} \beta_t^{\alpha^{(\beta)}-1} e^{-\beta^{(\beta)}\beta_t}}{\Gamma(\alpha^{(\beta)})} d\lambda_t d\beta_t \\ & = \frac{\beta^{(\lambda)\alpha^{(\lambda)}} \beta^{(\beta)\alpha^{(\beta)}}}{\Gamma(\alpha^{(\lambda)})\Gamma(\alpha^{(\beta)})} \prod_{j:N_{tj}>0} \left(\frac{v_{tj}^{N_{tj}} S_{tj}^{N_{tj}\hat{\alpha}_t - 1}}{N_{tj}! \Gamma(N_{tj}\hat{\alpha}_t)} \right) \int_0^\infty \lambda_t^{\sum_{j:N_{tj}>0} N_{tj} + \alpha^{(\lambda)} - 1} e^{-(\sum_{j=1}^{n_t} v_{tj} + \beta^{(\lambda)})\lambda_t} d\lambda_t \\ & \quad \times \int_0^\infty \beta_t^{\sum_{j:N_{tj}>0} N_{tj}\hat{\alpha}_t + \alpha^{(\beta)} - 1} e^{-(\sum_{j:N_{tj}>0} S_{tj} + \beta^{(\beta)})\beta_t} d\beta_t \end{aligned}$$

$$= \frac{\beta^{(\lambda)\alpha^{(\lambda)}}\beta^{(\beta)\alpha^{(\beta)}}}{\Gamma(\alpha^{(\lambda)})\Gamma(\alpha^{(\beta)})} \prod_{j:N_{tj}>0} \left(\frac{v_{tj}^{N_{tj}} S_{tj}^{N_{tj}\hat{\alpha}_t - 1}}{N_{tj}!\Gamma(N_{tj}\hat{\alpha}_t)} \right) \frac{\Gamma(\sum_{j:N_{tj}>0} N_{tj} + \alpha^{(\lambda)})}{(\sum_{j=1}^{n_t} v_{tj} + \beta^{(\lambda)})^{\sum_{j:N_{tj}>0} N_{tj} + \alpha^{(\lambda)}}}$$
$$\times \frac{\Gamma(\sum_{j:N_{tj}>0} N_{tj}\hat{\alpha}_t + \alpha^{(\beta)})}{(\sum_{j:N_{tj}>0} S_{tj} + \beta^{(\beta)})^{\sum_{j:N_{tj}>0} N_{tj}\hat{\alpha}_t + \alpha^{(\beta)}}}. \tag{5.13}$$

Clearly, from (5.13), we see that the posterior distributions of λ_t and β_t, conditional on data $(\boldsymbol{N}_t, \boldsymbol{S}_t)$ are given by,

$$\lambda_t \mid \boldsymbol{N}_t \sim \text{Gamma}\left(\sum_{j:N_{tj}>0} N_{tj} + \alpha^{(\lambda)}, \sum_{j=1}^{n_t} v_{tj} + \beta^{(\lambda)} \right),$$

$$\beta_t \mid \boldsymbol{N}_t, \boldsymbol{S}_t \sim \text{Gamma}\left(\sum_{j:N_{tj}>0} N_{tj}\hat{\alpha}_t + \alpha^{(\beta)}, \sum_{j:N_{tj}>0} S_{tj} + \beta^{(\beta)} \right).$$

The integrated likelihood for the tree $\mathcal{T}$ is thus given by

$$p_{\text{CPG}}\left(\boldsymbol{N}, \boldsymbol{S} \mid \boldsymbol{X}, \boldsymbol{v}, \hat{\boldsymbol{\alpha}}, \mathcal{T}\right) = \prod_{t=1}^{b} p_{\text{CPG}}\left(\boldsymbol{N}_t, \boldsymbol{S}_t \mid \boldsymbol{X}_t, \boldsymbol{v}_t, \hat{\alpha}_t\right). \tag{5.14}$$

Now, we discuss the DIC for this tree which can be derived as a special case of the new DIC proposed in Section 4.2 with a two-dimensional unknown parameter (λ_t, β_t). To this end, we first focus on DIC_t of terminal node t. It follows that

$$\begin{aligned} & D(\bar{\lambda}_t, \bar{\beta}_t) \\ &= -2 \sum_{j:N_{tj}>0} \left[(N_{tj}\hat{\alpha}_t - 1)\log(S_{tj}) - \bar{\beta}_t S_{tj} + N_{tj}\hat{\alpha}_t \log(\bar{\beta}_t) - \log\left(\Gamma(N_{tj}\hat{\alpha}_t)\right) \right] \\ &\quad - 2 \sum_{j:N_{tj}>0} \left(N_{tj}\log(\bar{\lambda}_t v_{tj}) - \log(N_{tj}!) \right) - 2\sum_{j=1}^{n_t} (-\bar{\lambda}_t v_{tj}), \end{aligned} \tag{5.15}$$

where

$$\bar{\lambda}_t = \frac{\sum_{j:N_{tj}>0} N_{tj} + \alpha^{(\lambda)}}{\sum_{j=1}^{n_t} v_{tj} + \beta^{(\lambda)}}, \tag{5.16}$$

$$\bar{\beta}_t = \frac{\sum_{j:N_{tj}>0} N_{tj}\hat{\alpha}_t + \alpha^{(\beta)}}{\sum_{j:N_{tj}>0} S_{tj} + \beta^{(\beta)}}. \tag{5.17}$$

Therefore, a direct calculation shows that the effective number of parameters for terminal node t is given by

$$s_{Dt}$$

$$= 1 + 2\left(\log\left(\sum_{j:N_{tj}>0} N_{tj}\hat{\alpha}_t + \alpha^{(\beta)}\right) - \psi\left(\sum_{j:N_{tj}>0} N_{tj}\hat{\alpha}_t + \alpha^{(\beta)}\right)\right)\sum_{j:N_{tj}>0} N_{tj}\hat{\alpha}_t$$
$$+2\left(\log\left(\sum_{j:N_{tj}>0} N_{tj} + \alpha^{(\lambda)}\right) - \psi\left(\sum_{j:N_{tj}>0} N_{tj} + \alpha^{(\lambda)}\right)\right)\sum_{j:N_{tj}>0} N_{tj}, \quad (5.18)$$

and thus

$$\begin{aligned}
\mathrm{DIC}_t &= D(\bar{\lambda}_t, \bar{\beta}_t) + 2s_{Dt} \\
&= -2\sum_{j:N_{tj}>0}\left[(N_{tj}\hat{\alpha}_t - 1)\log(S_{tj}) - \bar{\beta}_t S_{tj} + N_{tj}\hat{\alpha}_t\log(\bar{\beta}_t) - \log\left(\Gamma(N_{tj}\hat{\alpha}_t)\right)\right] \\
&\quad -2\sum_{j:N_{tj}>0}\left(N_{tj}\log(\bar{\lambda}_t v_{tj}) - \log(N_{tj}!)\right) - 2\sum_{j=1}^{n_t}(-\bar{\lambda}_t v_{tj}) \\
&\quad +2 + 4\left(\log\left(\sum_{j:N_{tj}>0} N_{tj} + \alpha^{(\lambda)}\right) - \psi\left(\sum_{j:N_{tj}>0} N_{tj} + \alpha^{(\lambda)}\right)\right)\sum_{j:N_{tj}>0} N_{tj} \\
&\quad +4\left(\log\left(\sum_{j:N_{tj}>0} N_{tj}\hat{\alpha}_t + \alpha^{(\beta)}\right) - \psi\left(\sum_{j:N_{tj}>0} N_{tj}\hat{\alpha}_t + \alpha^{(\beta)}\right)\right)\sum_{j:N_{tj}>0} N_{tj}\hat{\alpha}_t.
\end{aligned}$$

Then the DIC of the tree $\mathcal{T}$ is obtained by using (5.9). With the formulas derived above for the CPG case, we can use the three-step approach proposed in Subsection 2.2.4, together with Algorithm 4.1 in Section 4.2 (treat $\boldsymbol{\theta_M} = \boldsymbol{\alpha}$ and $\boldsymbol{\theta_B} = (\boldsymbol{\lambda}, \boldsymbol{\beta})$), to search for an optimal tree which can then be used to predict new data.

Remark 5.4 *The calculation methods in frequency-severity models and joint models are similar, but the former aims to model the claims frequency and claims severity separately, with $\bar{S}_i$ modelled directly; the latter one models (N_i,S_i) jointly instead.*

5.3.2 Zero-Inflated Compound Poisson Gamma-Bayesian CART

It is natural to consider the zero mass part additionally to the CPG distribution, as insurance data involves many zeros; see similar reasons explored in Section 3.3. We shall follow the same data augmentation strategy as in ZIP models to obtain a closed form for the posterior distribution. Similar to ZIP models, three ZICPG models are discussed below based on the way to embed the exposure into the

model. Given that ZICPG and CPG models share identical CPG parts, and the same data augmentation strategy is applied as in ZIP models, we will omit some repetitive details in the following calculations.

Zero-Inflated Compound Poisson Gamma model 1 (ZICPG1)

For terminal node t, we use the same CPG distribution as in Subsection 5.3.1 by embedding the exposure into the Poisson part,

$$\begin{aligned}
&f_{\text{ZICPG1}}\left(N_{tj}, S_{tj} \mid \mu_t, \lambda_t, \alpha_t, \beta_t, v_{tj}\right) \\
&= f_{\text{ZIP1}}(N_{tj} \mid \mu_t, \lambda_t, v_{tj}) f_{\text{G}}(S_{tj} \mid N_{tj}, \alpha_t, \beta_t) \\
&= \begin{cases} \frac{1}{1+\mu_t} + \frac{\mu_t}{1+\mu_t} e^{-\lambda_t v_{tj}} & (N_{tj}, S_{tj}) = (0,0), \\ \frac{\mu_t}{1+\mu_t} \frac{(\lambda_t v_{tj})^{N_{tj}} e^{-\lambda_t v_{tj}}}{N_{tj}!} \frac{\beta_t^{N_{tj}\alpha_t} S_{tj}^{N_{tj}\alpha_t - 1} e^{-\beta_t S_{tj}}}{\Gamma(N_{tj}\alpha_t)} & (N_{tj}, S_{tj}) \in \mathbb{N} \times \mathbb{R}^+, \end{cases}
\end{aligned} \tag{5.19}$$

where $\frac{1}{1+\mu_t} \in (0,1)$ is the probability that a zero is due to the point mass component. For the sake of computational convenience, a data augmentation scheme is needed. To this end, we introduce two latent variables $\boldsymbol{\phi}_t = (\phi_{t1}, \phi_{t2}, \ldots, \phi_{tn_t}) \in (0,\infty)^{n_t}$ and $\boldsymbol{\delta}_t = (\delta_{t1}, \delta_{t2}, \ldots, \delta_{tn_t}) \in \{0,1\}^{n_t}$ (the same as in ZIP1-ZIP3 models and will be omitted in the following models), and define the data augmented likelihood for the i-th data instance in terminal node t as

$$\begin{aligned}
&f_{\text{ZICPG1}}\left(N_{tj}, S_{tj}, \delta_{tj}, \phi_{tj} \mid \mu_t, \lambda_t, \alpha_t, \beta_t\right) \\
=&e^{-\phi_{tj}(1+\mu_t)} \left(\frac{\mu_t \left(\lambda_t v_{tj}\right)^{N_{tj}}}{N_{tj}!} e^{-\lambda_t v_{tj}} \right)^{\delta_{tj}} \frac{\beta_t^{N_{tj}\alpha_t} S_{tj}^{N_{tj}\alpha_t - 1} e^{-\beta_t S_{tj}}}{\Gamma(N_{tj}\alpha_t)},
\end{aligned} \tag{5.20}$$

where the support of the function f_{ZICPG1} is $\left(\{0\} \times \{0\} \times \{0,1\} \times (0,\infty)\right) \cup \left(\mathbb{N} \times \mathbb{R}^+ \times \{1\} \times (0,\infty)\right)$. It can be shown that (5.19) is the marginal distribution of the above augmented distribution.

By conditional arguments, we can also check that δ_{tj}, given data $N_{tj} = S_{tj} = 0$ and parameters (μ_t and λ_t), has a Bernoulli distribution, i.e.,

$$\delta_{tj} \mid N_{tj} = S_{tj} = 0, \mu_t, \lambda_t \ \sim \ \text{Bern}\left(\frac{\mu_t e^{-\lambda_t v_{tj}}}{1 + \mu_t e^{-\lambda_t v_{tj}}} \right),$$

and $\delta_{tj} = 1$, given $N_{tj} > 0$. Furthermore, ϕ_{tj}, given the parameter μ_t, has an Exponential distribution, i.e.,

$$\phi_{tj} \mid \mu_t \ \sim \ \text{Exp}\left(1 + \mu_t\right).$$

Similarly, to explicitly obtain the posterior distribution, we choose independent conjugate Gamma priors for μ_t, λ_t, and β_t with hyper-parameters $(\alpha^{(\mu)} > 0, \beta^{(\mu)} > 0)$, $(\alpha^{(\lambda)} > 0, \beta^{(\lambda)} > 0)$, and $(\alpha^{(\beta)} > 0, \beta^{(\beta)} > 0)$ respectively, where the subscript has the similar meaning as in the CPG model. Besides, α_t can be estimated and updated by using (5.12) in each step before updating other parameters. With these Gamma priors and the estimated parameter $\hat{\alpha}_t$, the integrated augmented likelihood for terminal node t can be obtained as follows

$$
\begin{aligned}
&p_{\text{ZICPG1}}\left(\boldsymbol{N}_t, \boldsymbol{S}_t, \boldsymbol{\delta}_t, \boldsymbol{\phi}_t \mid \boldsymbol{X}_t, \boldsymbol{v}_t, \hat{\alpha}_t\right) \\
&= \int_0^\infty \int_0^\infty \int_0^\infty f_{\text{ZICPG1}}\left(\boldsymbol{N}_t, \boldsymbol{S}_t, \boldsymbol{\delta}_t, \boldsymbol{\phi}_t \mid \mu_t, \lambda_t, \hat{\alpha}_t, \beta_t\right) p(\mu_t)p(\lambda_t)p(\beta_t)d\mu_t d\lambda_t d\beta_t \\
&= \int_0^\infty \int_0^\infty \int_0^\infty \prod_{j=1}^{n_t} \left(e^{-\phi_{tj}(1+\mu_t)} \left(\frac{\mu_t \left(\lambda_t v_{tj}\right)^{N_{tj}}}{N_{tj}!} e^{-\lambda_t v_{tj}} \right)^{\delta_{tj}} \frac{\beta_t^{N_{tj}\hat{\alpha}_t} S_{tj}^{N_{tj}\hat{\alpha}_t - 1} e^{-\beta_t S_{tj}}}{\Gamma(N_{tj}\hat{\alpha}_t)} \right) \\
&\quad \times \frac{\beta^{(\mu)\alpha^{(\mu)}} \mu_t^{\alpha^{(\mu)}-1} e^{-\beta^{(\mu)}\mu_t}}{\Gamma\left(\alpha^{(\mu)}\right)} \frac{\beta^{(\lambda)\alpha^{(\lambda)}} \lambda_t^{\alpha^{(\lambda)}-1} e^{-\beta^{(\lambda)}\lambda_t}}{\Gamma\left(\alpha^{(\lambda)}\right)} \frac{\beta^{(\beta)\alpha^{(\beta)}} \lambda_t^{\alpha^{(\beta)}-1} e^{-\beta^{(\beta)}\lambda_t}}{\Gamma\left(\alpha^{(\beta)}\right)} d\mu_t d\lambda_t d\beta_t \\
&= \frac{\beta^{(\mu)\alpha^{(\mu)}}}{\Gamma\left(\alpha^{(\mu)}\right)} \frac{\beta^{(\lambda)\alpha^{(\lambda)}}}{\Gamma\left(\alpha^{(\lambda)}\right)} \frac{\beta^{(\beta)\alpha^{(\beta)}}}{\Gamma\left(\alpha^{(\beta)}\right)} \prod_{j=1}^{n_t} \left(e^{-\phi_{tj}} v_{tj}^{\delta_{tj}N_{tj}} \left(N_{tj}!\right)^{-\delta_{tj}} \frac{S_{tj}^{N_{tj}\hat{\alpha}_t - 1}}{\Gamma(N_{tj}\hat{\alpha}_t)} \right) \\
&\quad \times \int_0^\infty \mu_t^{\sum_{j=1}^{n_t}\delta_{tj} + \alpha^{(\mu)} - 1} e^{-\left(\sum_{j=1}^{n_t}\phi_{tj} + \beta^{(\mu)}\right)\mu_t} d\mu_t \\
&\quad \times \int_0^\infty \lambda_t^{\sum_{j=1}^{n_t}\delta_{tj}N_{tj} + \alpha^{(\lambda)} - 1} e^{-\left(\sum_{j=1}^{n_t}\delta_{tj}v_{tj} + \beta^{(\lambda)}\right)\lambda_t} d\lambda_t \\
&\quad \times \int_0^\infty \beta_t^{\sum_{j:N_{tj}>0} N_{tj}\hat{\alpha}_t + \alpha^{(\beta)} - 1} e^{-\left(\sum_{j:N_{tj}>0} S_{tj} + \beta^{(\beta)}\right)\beta_t} d\beta_t \\
&= \frac{\beta^{(\mu)\alpha^{(\mu)}}}{\Gamma\left(\alpha^{(\mu)}\right)} \frac{\beta^{(\lambda)\alpha^{(\lambda)}}}{\Gamma\left(\alpha^{(\lambda)}\right)} \frac{\beta^{(\beta)\alpha^{(\beta)}}}{\Gamma\left(\alpha^{(\beta)}\right)} \prod_{j=1}^{n_t} \left(e^{-\phi_{tj}} v_{tj}^{\delta_{tj}N_{tj}} \left(N_{tj}!\right)^{-\delta_{tj}} \frac{S_{tj}^{N_{tj}\hat{\alpha}_t - 1}}{\Gamma(N_{tj}\hat{\alpha}_t)} \right) \\
&\quad \times \frac{\Gamma\left(\sum_{j=1}^{n_t}\delta_{tj} + \alpha^{(\mu)}\right)}{\left(\sum_{j=1}^{n_t}\phi_{tj} + \beta^{(\mu)}\right)^{\sum_{j=1}^{n_t}\delta_{tj} + \alpha^{(\mu)}}} \frac{\Gamma\left(\sum_{j=1}^{n_t}\delta_{tj}N_{tj} + \alpha^{(\lambda)}\right)}{\left(\sum_{j=1}^{n_t}\delta_{tj}v_{tj} + \beta^{(\lambda)}\right)^{\sum_{j=1}^{n_t}\delta_{tj}N_{tj} + \alpha^{(\lambda)}}} \\
&\quad \times \frac{\Gamma(\sum_{j:N_{tj}>0} N_{tj}\hat{\alpha}_t + \alpha^{(\beta)})}{(\sum_{j:N_{tj}>0} S_{tj} + \beta^{(\beta)})^{\sum_{j:N_{tj}>0} N_{tj}\hat{\alpha}_t + \alpha^{(\beta)}}}.
\end{aligned}
\tag{5.21}
$$

Moreover, from the above, we see that the posterior distributions of μ_t, λ_t are the same as in the ZIP1 model; and the posterior distribution of β_t is the same as in the CPG model.

The integrated augmented likelihood for the tree $\mathcal{T}$ is thus given by

$$p_{\text{ZICPG1}}\left(\boldsymbol{N}, \boldsymbol{S}, \boldsymbol{\delta}, \boldsymbol{\phi} \mid \boldsymbol{X}, \boldsymbol{v}, \hat{\boldsymbol{\alpha}}, \mathcal{T}\right) = \prod_{t=1}^{b} p_{\text{ZICPG1}}\left(\boldsymbol{N}_t, \boldsymbol{S}_t, \boldsymbol{\delta}_t, \boldsymbol{\phi}_t \mid \boldsymbol{X}_t, \boldsymbol{v}_t, \hat{\alpha}_t\right). \tag{5.22}$$

Now, we discuss the DIC for this tree which can be derived as a special case of the new DIC proposed in Section 3.2 (for NB models) with a three-dimensional unknown parameter $(\mu_t, \lambda_t, \beta_t)$. To this end, we first focus on DIC_t of terminal node t. It follows that

$$\begin{aligned}
&D(\bar{\mu}_t, \bar{\lambda}_t, \bar{\beta}_t) \\
&= -2 \log f_{\text{ZICPG1}}(\boldsymbol{N}_t, \boldsymbol{S}_t \mid \bar{\mu}_t, \bar{\lambda}_t, \bar{\beta}_t) \\
&= -2 \sum_{j=1}^{n_t} \log \left(\frac{1}{1+\bar{\mu}_t} I_{(N_{tj}=S_{tj}=0)} + \frac{\bar{\mu}_t}{1+\bar{\mu}_t} \frac{(\bar{\lambda}_t v_{tj})^{N_{tj}} e^{-\bar{\lambda}_t v_{tj}}}{N_{tj}!} \frac{\bar{\beta}_t^{N_{tj}\hat{\alpha}_t} S_{tj}^{N_{tj}\hat{\alpha}_t - 1} e^{-\bar{\beta}_t S_{tj}}}{\Gamma(N_{tj}\hat{\alpha}_t)} \right),
\end{aligned} \tag{5.23}$$

where $\bar{\mu}_t$ and $\bar{\lambda}_t$ have the same expressions as in the ZIP1 model (see (3.31)); $\bar{\beta}_t$ has the same expression as in (5.17). Therefore, a direct calculation shows that the effective number of parameters for terminal node t is given by

$$\begin{aligned}
r_{Dt} &= -2\mathbb{E}_{\text{post}}\left[\log f_{\text{ZICPG1}}(\boldsymbol{N}_t, \boldsymbol{S}_t, \boldsymbol{\delta}_t, \boldsymbol{\phi}_t \mid \mu_t, \lambda_t, \beta_t)\right] \\
&\quad + 2 \log f_{\text{ZICPG1}}(\boldsymbol{N}_t, \boldsymbol{S}_t, \boldsymbol{\delta}_t, \boldsymbol{\phi}_t \mid \bar{\mu}_t, \bar{\lambda}_t, \bar{\beta}_t) \\
&= 1 + 2\left(\log\left(\sum_{j=1}^{n_t} \delta_{tj} + \alpha^{(\mu)} \right) - \psi\left(\sum_{j=1}^{n_t} \delta_{tj} + \alpha^{(\mu)} \right) \right) \sum_{j=1}^{n_t} \delta_{tj} \\
&\quad + 2\left(\log\left(\sum_{j=1}^{n_t} \delta_{tj} N_{tj} + \alpha^{(\lambda)} \right) - \psi\left(\sum_{j=1}^{n_t} \delta_{tj} N_{tj} + \alpha^{(\lambda)} \right) \right) \sum_{j=1}^{n_t} \delta_{tj} N_{tj} \\
&\quad + 2\left(\log\left(\sum_{j:N_{tj}>0} N_{tj}\hat{\alpha}_t + \alpha^{(\beta)} \right) - \psi\left(\sum_{j:N_{tj}>0} N_{tj}\hat{\alpha}_t + \alpha^{(\beta)} \right) \right) \sum_{j:N_{tj}>0} N_{tj}\hat{\alpha}_t,
\end{aligned} \tag{5.24}$$

and thus $\text{DIC}_t = D(\bar{\mu}_t, \bar{\lambda}_t, \bar{\beta}_t) + 2r_{Dt}$ can be derived directly from (5.23) and (5.24).

Zero-Inflated Compound Poisson Gamma model 2 (ZICPG2)

For terminal node t, we embed the exposure into the zero mass part, and the CPG part has the distribution $\text{CPG}(\lambda, \alpha, \beta)$ which does not include the exposure and is different from Subsection 5.3.1,

$$
\begin{aligned}
&f_{\text{ZICPG2}}\left(N_{tj}, S_{tj} \mid \mu_t, \lambda_t, \alpha_t, \beta_t, v_{tj}\right) \\
=&f_{\text{ZIP2}}(N_{tj} \mid \mu_t, \lambda_t, v_{tj}) f_{\text{G}}(S_{tj} \mid N_{tj}, \alpha_t, \beta_t) \\
=&\begin{cases} \frac{1}{1+\mu_t v_{tj}} + \frac{\mu_t v_{tj}}{1+\mu_t v_{tj}} e^{-\lambda_t} & (N_{tj}, S_{tj}) = (0,0), \\ \frac{\mu_t v_{tj}}{1+\mu_t v_{tj}} \frac{\lambda_t^{N_{tj}} e^{-\lambda_t}}{N_{tj}!} \frac{\beta_t^{N_{tj}\alpha_t} S_{tj}^{N_{tj}\alpha_t - 1} e^{-\beta_t S_{tj}}}{\Gamma(N_{tj}\alpha_t)} & (N_{tj}, S_{tj}) \in \mathbb{N} \times \mathbb{R}^+, \end{cases}
\end{aligned} \tag{5.25}
$$

where $\frac{1}{1+\mu_t v_{tj}} \in (0,1)$ is the probability that a zero is due to the point mass component. Then, the data augmented likelihood for the j-th data instance in terminal node t can be defined as,

$$
\begin{aligned}
&f_{\text{ZICPG2}}\left(N_{tj}, S_{tj}, \delta_{tj}, \phi_{tj} \mid \mu_t, \lambda_t, \alpha_t, \beta_t\right) \\
=&e^{-\phi_{tj}\left(1+\mu_t v_{tj}\right)} \left(\frac{\mu_t v_{tj} \lambda_t^{N_{tj}}}{N_{tj}!} e^{-\lambda_t} \right)^{\delta_{tj}} \frac{\beta_t^{N_{tj}\alpha_t} S_{tj}^{N_{tj}\alpha_t - 1} e^{-\beta_t S_{tj}}}{\Gamma(N_{tj}\alpha_t)}.
\end{aligned} \tag{5.26}
$$

It is easy to check that (5.25) is the marginal distribution of the above augmented distribution.

By conditional arguments, we can also check that δ_{tj}, given data $N_{tj} = S_{tj} = 0$ and parameters (μ_t and λ_t), has a Bernoulli distribution, i.e.,

$$
\delta_{tj} \mid N_{tj} = 0, \mu_t, \lambda_t \ \sim \ \text{Bern}\left(\frac{\mu_t v_{tj} e^{-\lambda_t}}{1 + \mu_t v_{tj} e^{-\lambda_t}} \right),
$$

and $\delta_{tj} = 1$, given $N_{tj} > 0$. Furthermore, ϕ_{tj}, given the parameter μ_t, has an Exponential distribution, i.e.,

$$
\phi_{tj} \mid \mu_t \ \sim \ \text{Exp}\left(1 + \mu_t v_{tj}\right).
$$

As before, we assume independent conjugate Gamma priors for μ_t, λ_t, and β_t with hyper-parameters $(\alpha^{(\mu)} > 0, \beta^{(\mu)} > 0)$, $(\alpha^{(\lambda)} > 0, \beta^{(\lambda)} > 0)$, and $(\alpha^{(\beta)} > 0, \beta^{(\beta)} > 0)$ respectively. Besides, α_t can be estimated and updated by using (5.12). With these Gamma priors and the estimated parameter $\hat{\alpha}_t$, we can obtain the integrated augmented likelihood for terminal node t as follows

$$
\begin{aligned}
&p_{\text{ZICPG2}}\left(\boldsymbol{N}_t, \boldsymbol{S}_t, \boldsymbol{\delta}_t, \boldsymbol{\phi}_t \mid \boldsymbol{X}_t, \boldsymbol{v}_t, \hat{\alpha}_t\right) \\
=& \int_0^\infty \int_0^\infty \int_0^\infty f_{\text{ZICPG2}}\left(\boldsymbol{N}_t, \boldsymbol{S}_t, \boldsymbol{\delta}_t, \boldsymbol{\phi}_t \mid \mu_t, \lambda_t, \hat{\alpha}_t, \beta_t\right) p(\mu_t) p(\lambda_t) p(\beta_t) d\mu_t d\lambda_t d\beta_t \\
=& \int_0^\infty \int_0^\infty \int_0^\infty \prod_{j=1}^{n_t} \left(e^{-\phi_{tj}\left(1+\mu_t v_{tj}\right)} \left(\frac{\mu_t v_{tj} \lambda_t^{N_{tj}}}{N_{tj}!} e^{-\lambda_t} \right)^{\delta_{tj}} \frac{\beta_t^{N_{tj}\hat{\alpha}_t} S_{tj}^{N_{tj}\hat{\alpha}_t - 1} e^{-\beta_t S_{tj}}}{\Gamma(N_{tj}\hat{\alpha}_t)} \right) \\
&\times \frac{\beta^{(\mu)\alpha^{(\mu)}} {\mu_t}^{\alpha^{(\mu)}-1} e^{-\beta^{(\mu)}\mu_t}}{\Gamma\left(\alpha^{(\mu)}\right)} \frac{\beta^{(\lambda)\alpha^{(\lambda)}} {\lambda_t}^{\alpha^{(\lambda)}-1} e^{-\beta^{(\lambda)}\lambda_t}}{\Gamma\left(\alpha^{(\lambda)}\right)} \frac{\beta^{(\beta)\alpha^{(\beta)}} {\lambda_t}^{\alpha^{(\beta)}-1} e^{-\beta^{(\beta)}\lambda_t}}{\Gamma\left(\alpha^{(\beta)}\right)} d\mu_t d\lambda_t d\beta_t
\end{aligned}
$$

$$
\begin{aligned}
&= \frac{\beta^{(\mu)\alpha^{(\mu)}}}{\Gamma\left(\alpha^{(\mu)}\right)} \frac{\beta^{(\lambda)\alpha^{(\lambda)}}}{\Gamma\left(\alpha^{(\lambda)}\right)} \frac{\beta^{(\beta)\alpha^{(\beta)}}}{\Gamma\left(\alpha^{(\beta)}\right)} \prod_{j=1}^{n_t} \left(e^{-\phi_{tj}} \left(\frac{v_{tj}}{N_{tj}!} \right)^{\delta_{tj}} \frac{S_{tj}^{N_{tj}\hat{\alpha}_t - 1}}{\Gamma(N_{tj}\hat{\alpha}_t)} \right) \\
&\times \frac{\Gamma\left(\sum_{j=1}^{n_t} \delta_{tj} + \alpha^{(\mu)}\right)}{\left(\sum_{j=1}^{n_t} \phi_{tj} v_{tj} + \beta^{(\mu)}\right)^{\sum_{j=1}^{n_t} \delta_{tj} + \alpha^{(\mu)}}} \frac{\Gamma\left(\sum_{j=1}^{n_t} \delta_{tj} N_{tj} + \alpha^{(\lambda)}\right)}{\left(\sum_{j=1}^{n_t} \delta_{tj} + \beta^{(\lambda)}\right)^{\sum_{j=1}^{n_t} \delta_{tj} N_{tj} + \alpha^{(\lambda)}}} \\
&\times \frac{\Gamma(\sum_{j:N_{tj}>0} N_{tj}\hat{\alpha}_t + \alpha^{(\beta)})}{(\sum_{j:N_{tj}>0} S_{tj} + \beta^{(\beta)})^{\sum_{j:N_{tj}>0} N_{tj}\hat{\alpha}_t + \alpha^{(\beta)}}}.
\end{aligned} \tag{5.27}
$$

Moreover, from the above, we see that the posterior distributions of μ_t, λ_t are the same as in the ZIP2 model; and the posterior distribution of β_t is the same as in Subsection 5.3.1.

The integrated augmented likelihood for the tree $\mathcal{T}$ is thus given by

$$
p_{\text{ZICPG2}}\left(\boldsymbol{N}, \boldsymbol{S}, \boldsymbol{\delta}, \boldsymbol{\phi} \mid \boldsymbol{X}, \boldsymbol{v}, \hat{\boldsymbol{\alpha}}, \mathcal{T}\right) = \prod_{t=1}^{b} p_{\text{ZICPG2}}\left(\boldsymbol{N}_t, \boldsymbol{S}_t, \boldsymbol{\delta}_t, \boldsymbol{\phi}_t \mid \boldsymbol{X}_t, \boldsymbol{v}_t, \hat{\alpha}_t\right). \tag{5.28}
$$

Now, we discuss the DIC_t of terminal node t. It follows that

$$
\begin{aligned}
&D(\bar{\mu}_t, \bar{\lambda}_t, \bar{\beta}_t) \\
&= -2 \log f_{\text{ZICPG2}}(\boldsymbol{N}_t, \boldsymbol{S}_t \mid \bar{\mu}_t, \bar{\lambda}_t, \bar{\beta}_t) \\
&= -2 \sum_{j=1}^{n_t} \log \left(\frac{1}{1+\bar{\mu}_t v_{tj}} I_{(N_{tj}=S_{tj}=0)} + \frac{\bar{\mu}_t v_{tj}}{1+\bar{\mu}_t v_{tj}} \frac{\bar{\lambda}_t^{N_{tj}} e^{-\bar{\lambda}_t}}{N_{tj}!} \frac{\bar{\beta}_t^{N_{tj}\hat{\alpha}_t} S_{tj}^{N_{tj}\hat{\alpha}_t - 1} e^{-\bar{\beta}_t S_{tj}}}{\Gamma(N_{tj}\hat{\alpha}_t)} \right),
\end{aligned} \tag{5.29}
$$

where $\bar{\mu}_t$ and $\bar{\lambda}_t$ have the same expressions as in the ZIP2 model (see (3.38)); $\bar{\beta}_t$ has the same expression as in (5.17). Therefore, a direct calculation shows that the effective number of parameters for terminal node t has the same expression as in the ZICPG1 model (see (5.24)). Thus, $\text{DIC}_t = D(\bar{\mu}_t, \bar{\lambda}_t, \bar{\beta}_t) + 2r_{Dt}$ can be derived directly from (5.29) and (5.24).

Zero-Inflated Compound Poisson Gamma model 3 (ZICPG3)

For terminal node t, we embed the exposure into both the Poisson part and the zero mass part. In this case, we use the same CPG distribution $\text{CPG}(\lambda v_{tj}, \alpha, \beta)$ as in Subsection 5.3.1,

$$
\begin{aligned}
&f_{\text{ZICPG3}}\left(N_{tj}, S_{tj} \mid \mu_t, \lambda_t, \alpha_t, \beta_t, v_{tj}\right) \\
=&f_{\text{ZIP3}}(N_{tj} \mid \mu_t, \lambda_t, v_{tj}) f_{\text{G}}(S_{tj} \mid N_{tj}, \alpha_t, \beta_t) \\
=&\begin{cases}
\frac{1}{1+\mu_t v_{tj}} + \frac{\mu_t v_{tj}}{1+\mu_t v_{tj}} e^{-\lambda_t v_{tj}} & (N_{tj}, S_{tj}) = (0,0), \\
\frac{\mu_t v_{tj}}{1+\mu_t v_{tj}} \frac{(\lambda_t v_{tj})^{N_{tj}} e^{-\lambda_t v_{tj}}}{N_{tj}!} \frac{\beta_t^{N_{tj}\alpha_t} S_{tj}^{N_{tj}\alpha_t - 1} e^{-\beta_t S_{tj}}}{\Gamma(N_{tj}\alpha_t)} & (N_{tj}, S_{tj}) \in \mathbb{N} \times \mathbb{R}^+,
\end{cases}
\end{aligned} \tag{5.30}
$$

where $\frac{1}{1+\mu_t v_{tj}} \in (0,1)$ is the probability that a zero is due to the point mass component. Then, the data augmented likelihood for the j-th data instance in terminal node t can be defined as,

$$\begin{aligned} & f_{\text{ZICPG3}}\left(N_{tj}, S_{tj}, \delta_{tj}, \phi_{tj} \mid \mu_t, \lambda_t, \alpha_t, \beta_t\right) \\ &= e^{-\phi_{tj}(1+\mu_t v_{tj})} \left(\frac{\mu_t v_{tj} \left(\lambda_t v_{tj}\right)^{N_{tj}}}{N_{tj}!} e^{-\lambda_t v_{tj}} \right)^{\delta_{tj}} \frac{\beta_t^{N_{tj}\alpha_t} S_{tj}^{N_{tj}\alpha_t - 1} e^{-\beta_t S_{tj}}}{\Gamma(N_{tj}\alpha_t)}. \end{aligned} \tag{5.31}$$

It is easy to check that (5.30) is the marginal distribution of the above augmented distribution.

By conditional arguments, we can also check that δ_{tj}, given data $N_{tj} = S_{tj} = 0$ and parameters (μ_t and λ_t), has a Bernoulli distribution, i.e.,

$$\delta_{tj} \mid N_{tj} = 0, \mu_t, \lambda_t \ \sim \ \text{Bern}\left(\frac{\mu_t v_{tj} e^{-\lambda_t v_{tj}}}{1 + \mu_t v_{tj} e^{-\lambda_t v_{tj}}} \right),$$

and $\delta_{tj} = 1$, given $N_{tj} > 0$. Furthermore, ϕ_{tj}, given the parameter μ_t, has an Exponential distribution, i.e.,

$$\phi_{tj} \mid \mu_t \ \sim \ \text{Exp}\left(1 + \mu_t v_{tj}\right).$$

As before, we assume independent conjugate Gamma priors for μ_t, λ_t, and β_t with hyper-parameters $(\alpha^{(\mu)} > 0, \beta^{(\mu)} > 0)$, $(\alpha^{(\lambda)} > 0, \beta^{(\lambda)} > 0)$, and $(\alpha^{(\beta)} > 0, \beta^{(\beta)} > 0)$ respectively. Besides, α_t can be estimated and updated by using (5.12). With these Gamma priors and the estimated parameter $\hat{\alpha}_t$, we can obtain the integrated augmented likelihood for terminal node t as follows

$$\begin{aligned} & p_{\text{ZICPG3}}\left(\boldsymbol{N}_t, \boldsymbol{S}_t, \boldsymbol{\delta}_t, \boldsymbol{\phi}_t \mid \boldsymbol{X}_t, \boldsymbol{v}_t, \hat{\alpha}_t\right) \\ &= \int_0^\infty \int_0^\infty \int_0^\infty f_{\text{ZICPG3}}\left(\boldsymbol{N}_t, \boldsymbol{S}_t, \boldsymbol{\delta}_t, \boldsymbol{\phi}_t \mid \mu_t, \lambda_t, \hat{\alpha}_t, \beta_t\right) p(\mu_t) p(\lambda_t) p(\beta_t) d\mu_t d\lambda_t d\beta_t \\ &= \int_0^\infty \int_0^\infty \int_0^\infty \prod_{j=1}^{n_t} \left(e^{-\phi_{tj}(1+\mu_t v_{tj})} \left(\frac{\mu_t v_{tj} \left(\lambda_t v_{tj}\right)^{N_{tj}}}{N_{tj}!} e^{-\lambda_t v_{tj}} \right)^{\delta_{tj}} \frac{\beta_t^{N_{tj}\hat{\alpha}_t} S_{tj}^{N_{tj}\hat{\alpha}_t - 1} e^{-\beta_t S_{tj}}}{\Gamma(N_{tj}\hat{\alpha}_t)} \right) \\ &\quad \times \frac{\beta^{(\mu)\alpha^{(\mu)}} {\mu_t}^{\alpha^{(\mu)}-1} e^{-\beta^{(\mu)}\mu_t}}{\Gamma\left(\alpha^{(\mu)}\right)} \frac{\beta^{(\lambda)\alpha^{(\lambda)}} {\lambda_t}^{\alpha^{(\lambda)}-1} e^{-\beta^{(\lambda)}\lambda_t}}{\Gamma\left(\alpha^{(\lambda)}\right)} \frac{\beta^{(\beta)\alpha^{(\beta)}} {\lambda_t}^{\alpha^{(\beta)}-1} e^{-\beta^{(\beta)}\lambda_t}}{\Gamma\left(\alpha^{(\beta)}\right)} d\mu_t d\lambda_t d\beta_t \\ &= \frac{\beta^{(\mu)\alpha^{(\mu)}}}{\Gamma\left(\alpha^{(\mu)}\right)} \frac{\beta^{(\lambda)\alpha^{(\lambda)}}}{\Gamma\left(\alpha^{(\lambda)}\right)} \frac{\beta^{(\beta)\alpha^{(\beta)}}}{\Gamma\left(\alpha^{(\beta)}\right)} \prod_{j=1}^{n_t} \left(e^{-\phi_{tj}} v_{tj}^{\delta_{tj}(1+N_{tj})} \left(N_{tj}!\right)^{-\delta_{tj}} \frac{S_{tj}^{N_{tj}\hat{\alpha}_t - 1}}{\Gamma(N_{tj}\hat{\alpha}_t)} \right) \\ &\quad \times \frac{\Gamma\left(\sum_{j=1}^{n_t} \delta_{tj} + \alpha^{(\mu)}\right)}{\left(\sum_{j=1}^{n_t} \phi_{tj} v_{tj} + \beta^{(\mu)}\right)^{\sum_{j=1}^{n_t} \delta_{tj} + \alpha^{(\mu)}}} \frac{\Gamma\left(\sum_{j=1}^{n_t} \delta_{tj} N_{tj} + \alpha^{(\lambda)}\right)}{\left(\sum_{j=1}^{n_t} \delta_{tj} v_{tj} + \beta^{(\lambda)}\right)^{\sum_{j=1}^{n_t} \delta_{tj} N_{tj} + \alpha^{(\lambda)}}} \end{aligned}$$

$$\times \frac{\Gamma(\sum_{j:N_{tj}>0} N_{tj}\hat{\alpha}_t + \alpha^{(\beta)})}{(\sum_{j:N_{tj}>0} S_{tj} + \beta^{(\beta)})^{\sum_{j:N_{tj}>0} N_{tj}\hat{\alpha}_t + \alpha^{(\beta)}}}. \tag{5.32}$$

Moreover, from the above, we see that the posterior distributions of μ_t, λ_t are the same as in the ZIP3 model; and the posterior distribution of β_t is the same as in the CPG model.

The integrated augmented likelihood for the tree $\mathcal{T}$ is thus given by

$$p_{\text{ZICPG3}}\left(\boldsymbol{N}, \boldsymbol{S}, \boldsymbol{\delta}, \boldsymbol{\phi} \mid \boldsymbol{X}, \boldsymbol{v}, \hat{\boldsymbol{\alpha}}, \mathcal{T}\right) = \prod_{t=1}^{b} p_{\text{ZICPG3}}\left(\boldsymbol{N}_t, \boldsymbol{S}_t, \boldsymbol{\delta}_t, \boldsymbol{\phi}_t \mid \boldsymbol{X}_t, \boldsymbol{v}_t, \hat{\alpha}_t\right). \tag{5.33}$$

Now, we discuss the DIC_t of terminal node t. It follows that

$$\begin{aligned} & D(\bar{\mu}_t, \bar{\lambda}_t, \bar{\beta}_t) \\ &= -2 \log f_{\text{ZICPG3}}(\boldsymbol{N}_t, \boldsymbol{S}_t \mid \bar{\mu}_t, \bar{\lambda}_t, \bar{\beta}_t) \\ &= -2 \sum_{j=1}^{n_t} \log \left(\frac{1}{1+\bar{\mu}_t v_{tj}} I_{(N_{tj}=S_{tj}=0)} + \frac{\bar{\mu}_t v_{tj}}{1+\bar{\mu}_t v_{tj}} \frac{(\bar{\lambda}_t v_{tj})^{N_{tj}} e^{-\bar{\lambda}_t v_{tj}}}{N_{tj}!} \frac{\bar{\beta}_t^{N_{tj}\hat{\alpha}_t} S_{tj}^{N_{tj}\hat{\alpha}_t - 1} e^{-\bar{\beta}_t S_{tj}}}{\Gamma(N_{tj}\hat{\alpha}_t)} \right), \end{aligned} \tag{5.34}$$

where $\bar{\mu}_t$ and $\bar{\lambda}_t$ have the same expressions as in the ZIP3 model (see (3.44)); $\bar{\beta}_t$ has the same expression as in (5.17). Therefore, a direct calculation shows that the effective number of parameters for terminal node t has the same expression as in the ZICPG1 model (see (5.24)), illustrating that the way to embed the exposure does not influence the effective number of parameters. This aligns with the conclusion obtained for NB, ZIP, and ZINB models in Chapter 3. Thus, $\text{DIC}_t = D(\bar{\mu}_t, \bar{\lambda}_t, \bar{\beta}_t) + 2r_{Dt}$ can be derived directly from (5.34) and (5.24).

For the above three ZICPG models, the DIC of the tree $\mathcal{T}$ is obtained by using (5.9). With the formulas derived above for three ZICPG models, we can use the three-step approach proposed in Section 2.2.4, together with Algorithm 3.1 in Section 3.2 (treat $\boldsymbol{\theta_M} = \boldsymbol{\alpha}$, $\boldsymbol{\theta_B} = (\boldsymbol{\mu}, \boldsymbol{\lambda}, \boldsymbol{\beta})$, and $\boldsymbol{z} = (\boldsymbol{\delta}, \boldsymbol{\phi})$), to search for an optimal tree which can then be used to predict new data.

5.3.3 A Simulation Example: Shared Covariates

This section aims to investigate scenarios where identical covariates exhibit similar or distinct impacts on claims frequency and claims severity. The objective is to assess the effectiveness of employing two trees and one joint tree in such cases and obtain a general conclusion. To simplify and better illustrate the necessity of sharing information, we focus on the CPG distribution in joint models in this example.

Besides, since the CPG model involves Poisson and Gamma distributions, we restrict the use of Poisson-BCART and Gamma-BCART in the frequency-severity models to keep consistency for the comparison. Specifically, the CPG distribution models individual claim amount Y_{ij} first and then obtain the distribution for the aggregate claim amount S_i. We shall use Gamma-BCART proposed in Section 5.1 for the frequency-severity models, which also models individual claim amount Y_{ij} first rather than modelling $\bar{S}_i$ directly. In the simulation setting, we would model $\bar{S}_i$ involving N_i as model weights, i,e., $\bar{S}_i \sim \text{Gamma}(N_i\alpha, N_i\beta(\boldsymbol{x}_i))$.

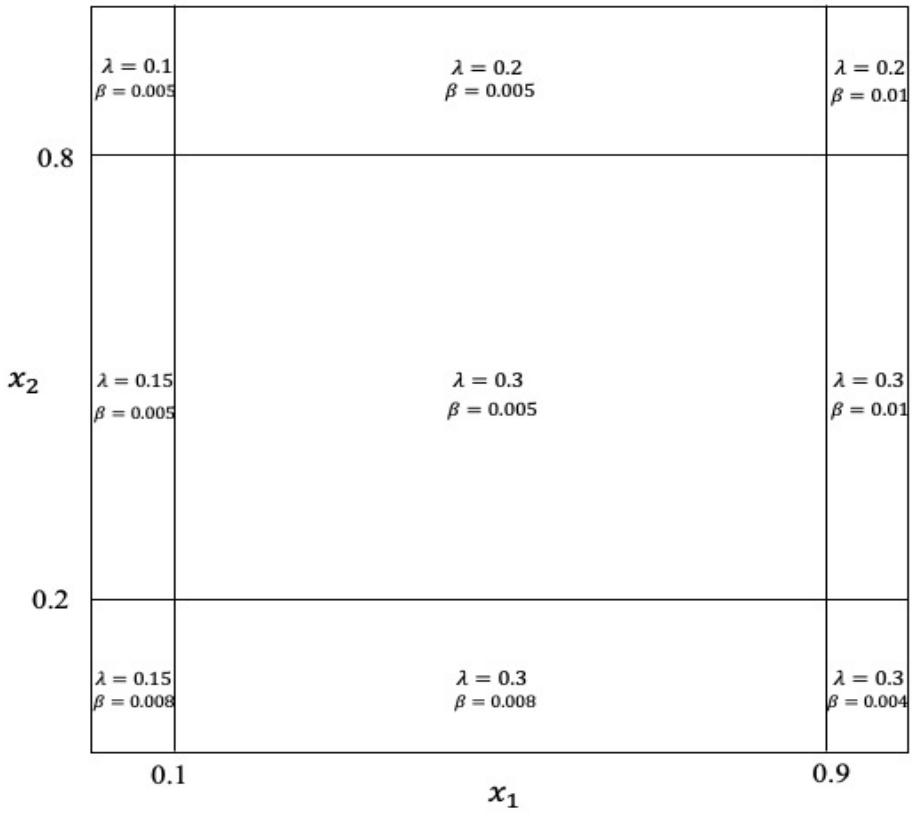

Figure 5.1: Covariate partition for a Compound Poisson Gamma-distributed simulation. Two covariates x_1 and x_2 follow a Normal and Uniform distribution respectively, i.e., $x_1 \sim N(0,1)$, $x_2 \sim U(-1,1)$. The values of parameters λ (in the Poisson model) and β (in the Gamma model) are provided in each region.

We simulate a data set $\left\{(\boldsymbol{x}_i, v_i, N_i, \bar{S}_i)\right\}_{i=1}^{n}$ with $n = 5,000$ independent observations. Here $\boldsymbol{x}_i = (x_{i1}, \ldots, x_{i5})$, with independent components $x_{i1} \sim N(0,1)$, $x_{i2} \sim U(-1,1)$, $x_{i3} \sim U(-5,5)$, $x_{i4} \sim N(0,5)$, $x_{i5} \sim U\{1,2,3,4\}$, and $v_i \sim$

$U(0,1)$. Moreover, $N_i \sim \text{Poi}(\lambda(x_{i1}, x_{i2})v_i)$, where

$$\lambda(x_1, x_2) = \begin{cases} 0.1 & \text{if } x_1 \leq 0.1, x_2 > 0.8, \\ 0.2 & \text{if } x_1 > 0.1, x_2 > 0.8, \\ 0.3 & \text{if } x_1 > 0.1, x_2 \leq 0.8, \\ 0.15 & \text{if } x_1 \leq 0.1, x_2 \leq 0.8, \end{cases}$$

and $N_i = 0$ leads to $\bar{S}_i = 0$ directly. For the remaining non-zero cases, $\bar{S}_i$ is generated from a Gamma distribution, i.e., $\bar{S}_i \sim \text{Gamma}(N_i\alpha, N_i\beta(x_{i1}, x_{i2}))$, where

$$\beta(x_1, x_2) = \begin{cases} 0.005 & \text{if } x_1 \leq 0.9, x_2 > 0.2, \\ 0.01 & \text{if } x_1 > 0.9, x_2 > 0.2, \\ 0.004 & \text{if } x_1 > 0.9, x_2 \leq 0.2, \\ 0.008 & \text{if } x_1 \leq 0.9, x_2 \leq 0.2. \end{cases}$$

As before, α is the shape parameter of the Gamma distribution, which is specified to be fixed at 1 for simplicity since it is not a key feature here. Besides, the design of the value for the rate parameter β is to keep the average claim amount $\bar{S}_i$ around 200, which is close to the situation in real data; see Figure 5.1. Obviously, the designed noise variables $x_{ik}, k = 3, 4, 5$ are all independent of the bivariate response variable $(\boldsymbol{N}, \bar{\boldsymbol{S}})$. The data is split into two subsets: a training set with $n - m = 4,000$ observations and a test set with $m = 1,000$ observations. The special design here is that even though the bivariate response variable $(\boldsymbol{N}, \bar{\boldsymbol{S}})$ is influenced by the same covariates x_1 and x_2, they have different split points. We aim to assess the performance of two trees versus one joint tree in this situation. The intuition is not very clear because, although they share some information (having the same covariates), there are other distinct factors that influence them (different split values), leading to different splitting rules. We shall use the comparison indexes in both training data and test data to clarify it.

We initiate the discussion by assessing the performance on training data in Table 5.4. The DIC indicates that for both Poisson-BCART and Gamma-BCART, the optimal tree with the true 4 terminal nodes can be selected, and for CPG-BCART, it chooses 9 as the optimal number of terminal nodes which is consistent with the simulation setting (see Figure 5.1). Based on this, we can conclude that BCART models are capable of bivariate response modelling using one joint tree. In Table 5.5, we can clearly see that even though joint models (CPG-BCART) use less time and smaller memory usage, they are not as good as the frequency-severity models in the comparison of both RSS($\boldsymbol{S}$) and NLL. Secondly, joint models obtain a tree with 9 terminal nodes to find the optimal solution. Compared with the two optimal trees with 4 terminal nodes found by the frequency-severity models, joint

Table 5.4: Hyper-parameters, p_D (or s_D) and DIC on training data. The number in the bracket after the abbreviation of the model indicates the number of terminal nodes for this tree. Bold font indicates DIC selected model. This table helps to select the optimal tree, and DICs between different models cannot be directly compared.

Model	γ	ρ	p_D (or s_D)	DIC
Poisson-BCART (3)	0.95	15	2.97	3875
Poisson-BCART (4)	0.99	12	3.97	**3669**
Poisson-BCART (5)	0.99	10	4.96	3724
Gamma-BCART (3)	0.95	10	5.97	32156
Gamma-BCART (4)	0.99	10	7.96	**31798**
Gamma-BCART (5)	0.99	8	9.96	31904
CPG-BCART (8)	0.99	5	23.85	36174
CPG-BCART (9)	0.99	3	26.81	**35622**
CPG-BCART (10)	0.99	2	29.79	35781

Table 5.5: Model performance on test data with bold entries determined by DIC (see Table 5.4). $\mathrm{F_PS_G}$ means the frequency-severity models by using Poisson and Gamma distributions separately. The number in the bracket after the abbreviation of the model indicates the number of terminal nodes for this tree. Particularly, two numbers for frequency-severity models indicate the number of terminal nodes for each tree.

Model	RSS($\boldsymbol{S}$) (in 10^8)	NLL	Time (s)	Memory (MB)
$\mathrm{F_PS_G}$-BCART (3/3)	3.21	5330	98	65
$\mathbf{F_PS_G}$-BCART (4/4)	3.04	5028	102	68
$\mathrm{F_PS_G}$-BCART (5/5)	2.95	4934	101	69
CPG-BCART (8)	3.23	5415	63	41
CPG-BCART (9)	3.08	5127	68	43
CPG-BCART (10)	3.01	5015	70	44

models may not be preferred. The models with bold entries (as determined in Table 5.4) do not show the best performance in Table 5.5, since both frequency-severity models and joint models will improve based on RSS($\boldsymbol{S}$) and NLL when

the number of terminal nodes increases (see Section 2.3). Nevertheless, given the demonstrated effectiveness of DIC in all previous simulation examples, we maintain the belief that the models determined by DIC would remain the optimal choice in this case.

Given that both frequency-severity models and joint models identify the optimal trees as expected, and considering the model performance of the two models, our conclusion suggests that there is no need for information sharing. This lack of necessity may arise from significant dissimilarities observed between the two trees. To explore this further, we hypothesize that the greater the similarity between two trees (using the same splitting variables, same or similar split values/categories), the more imperative it is for them to share information in one joint tree to avoid redundant use of the same or similar information. Therefore, we design another case where the values of split points are closer, i.e.,

$$\lambda\left(x_1, x_2\right) = \begin{cases} 0.1 & \text{if } x_1 \leq 0.47, x_2 > 0.52, \\ 0.2 & \text{if } x_1 > 0.47, x_2 > 0.52, \\ 0.3 & \text{if } x_1 > 0.47, x_2 \leq 0.52, \\ 0.15 & \text{if } x_1 \leq 0.47, x_2 \leq 0.52, \end{cases}$$

and for non-zero cases, generate $\bar{S}_i$ by using $\bar{S}_i \sim \text{Gamma}(N_i\alpha, N_i\beta(x_{i1}, x_{i2}))$, where

$$\beta\left(x_1, x_2\right) = \begin{cases} 0.005 & \text{if } x_1 \leq 0.53, x_2 > 0.48, \\ 0.01 & \text{if } x_1 > 0.53, x_2 > 0.48, \\ 0.004 & \text{if } x_1 > 0.53, x_2 \leq 0.48, \\ 0.008 & \text{if } x_1 \leq 0.53, x_2 \leq 0.48, \end{cases}$$

keeping other settings the same as before; see Figure 5.2.

Tables 5.6 and 5.7 show the results. We omit some detailed analysis similar to the previous example to minimise content duplication and highlight different points in this case. For Poisson-BCART and Gamma-BCART, it is easy to check that both of them find the optimal tree with the true 4 terminal nodes, as suggested by DIC on training data in Table 5.6. For CPG-BCART, in the simulation setting, we expect it to still have 9 terminal nodes as in the previous example, but DIC indicates that the tree with 4 terminal nodes is the best one. To explore the reason, we check the tree structure and find the chosen split points for both x_1 and x_2 are close to 0.5, i.e., the mean of two different setting values in $\lambda(x_1, x_2)$ and $\beta(x_1, x_2)$ (0.47 and 0.53 for x_1; 0.52 and 0.48 for x_2). Since the setting split values are very close, it is a good choice to select a splitting value around their mean. In doing so, the tree structure can be simplified significantly. Following that, we evaluate the model performance on test data. From Table 5.7, it is obvious to see that

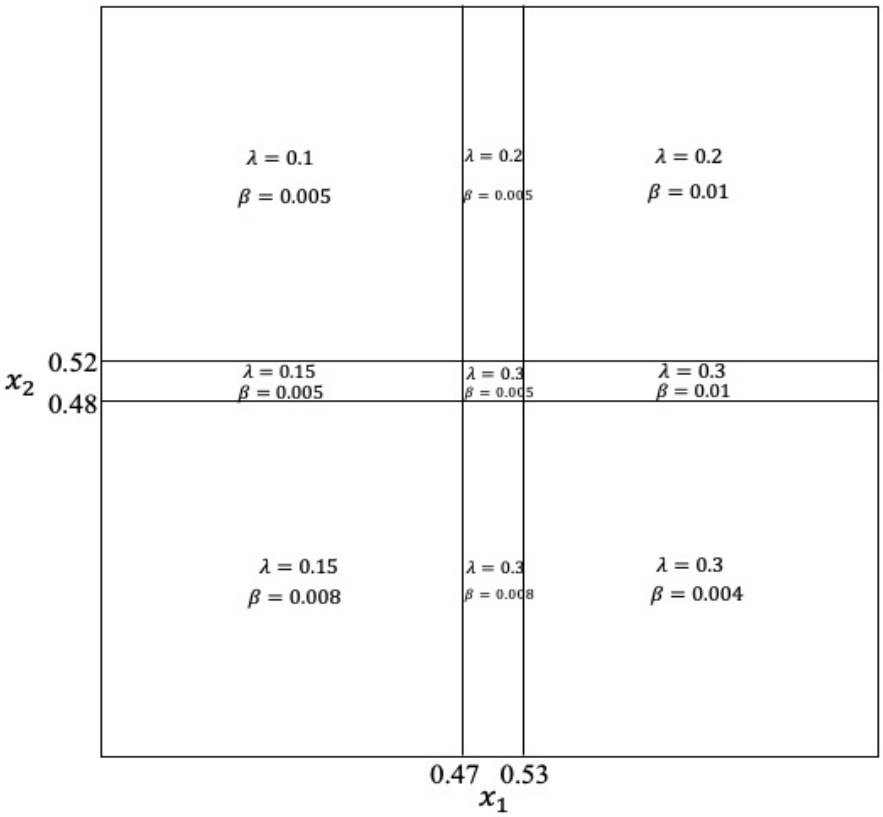

Figure 5.2: Covariate partition for a Compound Poisson Gamma-distributed simulation. Two covariates x_1 and x_2 follow a Normal and Uniform distribution respectively, i.e., $x_1 \sim N(0,1)$, $x_2 \sim U(-1,1)$. The values of parameters λ (in the Poisson model) and β (in the Gamma model) are provided in each region.

joint models (CPG-BCART) perform better among all comparison indexes. In addition, in terms of interpretability, one tree would evidently be better.

Afterwards, we propose to use the adjusted Rand Index (ARI) to examine the similarity between different trees and validate our intuition. The ARI is a widely employed metric for measuring the similarity between different clusterings; see, e.g., Rand (1971), Hubert & Arabie (1985) and Gates & Ahn (2017). The ARI ranges from -1 to 1, where 1 indicates perfect agreement between two clusterings, 0 denotes random agreement, and -1 signifies complete dissimilarity between the two clusterings. Therefore, a higher ARI value indicates greater similarity between the two trees. More details about ARI can be found in Appendix C. We calculate the ARI for these three optimal trees,

$$\text{ARI(Poisson-BCART (4), Gamma-BCART (4))} = 0.8667$$

Table 5.6: Hyper-parameters, p_D (or s_D) and DIC on training data. The number in the bracket after the abbreviation of the model indicates the number of terminal nodes for this tree. Bold font indicates DIC selected model. This table helps to select the optimal tree, and DICs between different models cannot be directly compared.

Model	γ	ρ	p_D (or s_D)	DIC
Poisson-BCART (3)	0.95	15	2.98	3697
Poisson-BCART (4)	0.99	13	3.98	**3572**
Poisson-BCART (5)	0.99	10	4.97	3616
Gamma-BCART (3)	0.95	10	5.97	30586
Gamma-BCART (4)	0.99	10	7.97	**30319**
Gamma-BCART (5)	0.99	8	9.96	30414
CPG-BCART (3)	0.99	5	8.92	34017
CPG-BCART (4)	0.99	4	11.90	**33582**
CPG-BCART (5)	0.99	3	14.89	33711

Table 5.7: Model performance on test data with bold entries determined by DIC (see Table 5.6). $\mathrm{F_PS_G}$ means the frequency-severity models by using Poisson and Gamma distributions separately. The number in the bracket after the abbreviation of the model indicates the number of terminal nodes for this tree. Particularly, two numbers for frequency-severity models indicate the number of terminal nodes for each tree.

Model	RSS($\boldsymbol{S}$) (in 10^8)	NLL	Time (s)	Memory (MB)
$\mathrm{F_PS_G}$-BCART (3/3)	3.03	5276	103	67
$\mathbf{F_PS_G}$-BCART (4/4)	2.89	5024	105	70
$\mathrm{F_PS_G}$-BCART (5/5)	2.81	4936	106	71
CPG-BCART (3)	3.01	5213	72	38
CPG-BCART (4)	2.84	4947	78	40
CPG-BCART (5)	2.78	4859	81	40

$$\text{ARI(Poisson-BCART (4), CPG-BCART (4))} = 0.9404$$

$$\text{ARI(Gamma-BCART (4), CPG-BCART (4))} = 0.9236$$

confirming our speculation, as the ARI value for two trees is close to 1, indicating

high similarity. However, determining a specific ARI threshold that indicates when sharing information becomes more effective is challenging. Further research is needed in this area. In the real data analyses in Chapter 6, we use evaluation metrics to compare different models first. ARI is then employed as an auxiliary tool to explore why two trees or one joint tree may be more effective.

We also run several other simulation examples which are not displayed here. From these analyses, we conclude that: when two trees have more same or similar splitting rules (high ARI), one joint tree is more effective through information sharing. Conversely, if all covariates affecting claims frequency and claims severity are different (ARI is close to -1), two trees outperform one joint tree. This conclusion aligns with our intuition.

Remark 5.5 *There are other ways to calculate the similarity between different trees (see, e.g., Nye et al. (2006)), but they usually require consideration of the tree structure. If the number of terminal nodes or tree structure (balanced/unbalanced) is different, it is hard to calculate the score for similarity. Regarding ARI, it exclusively assesses the partitioned data without considering specific structural differences. This makes it versatile across various clustering methods. Additionally, ARI is easy to implement and calculate in R, using the package* `fossil` *(see Vavrek (2020)).*

5.4 Summary of Chapter 5

In this chapter, we proposed three types of models for the aggregate claim amount. First, we found that the sequential models treating the number of claims as a covariate in the claims severity modelling perform better than the standard frequency-severity models when the dependence between the number of claims and claims severity is strong. Second, we explored the choice between using two trees or one joint tree. In particular, when there are more identical or similar splitting rules between claims frequency and claims severity, it is better to use one joint tree to share information. Third, we provided details on the applications of evaluation metrics in the case of two trees and proposed the use of ARI to quantify the similarity between the two trees, which can assist in explaining the necessity of information sharing. Finally, in the analysis of various joint models, especially the three ZICPG models employing different methods to embed exposure, we address their similarities to the analysis of ZIP and ZINB models discussed in Chapter 3. More detailed discussions about their model performance will be presented in the real insurance data analysis in Chapter 6.

Chapter 6

Insurance Data Analysis

The unique nature of the insurance industry, combined with the complexities of managing extensive data sets, poses significant challenges, such as the necessity of data pre-processing and the optimal selection of distributions that accurately capture the features of the data among massive distributions. Building accurate predictive models that reflect real-world risks is crucial for the insurance industry. Simultaneously, maintaining the interpretability of the model is essential, as customers (the insured) and regulators are not expected to possess sufficient statistical knowledge to understand the model. This chapter uses one real dataset to illustrate the effectiveness and feasibility of our proposed BCART models. Following Chapters 3, 4, and 5, we shall discuss claims frequency modelling, claims severity modelling, and aggregate claims modelling. We will see that the conclusions obtained from real data are consistent with those obtained from the simulation examples we conducted previously in Sections 3.5, 4.5, 5.2.1, 5.3.3. Therefore, we conclude that BCART models have superior performance and can be well applied to the insurance industry.

6.1 Data Description: *dataCar*

The real insurance dataset used to illustrate our methodology is named *dataCar*, available from the library `insuranceData` in R; see Wolny-Dominiak & Trzesiok (2014) for details. This dataset is based on one-year vehicle insurance policies taken out in 2004 or 2005. There are 67,856 policies of which 93.19% made no claims. A summary of the variables used is given in Table 6.1. We split this dataset into training (80%) and test (20%) data sets, in doing so we keep the balance of zero and non-zero claims in both training and test data sets. Tables 6.2 and 6.3 provide some basic information about the number of claims and claims severity

respectively, revealing a substantial mixture of zeros representing no claims and a right-skewed distribution with heavy tails due to large claim amounts, reflected in significant skewness and kurtosis. To better understand the dataset, especially the relationship between covariates and claims frequency and claims severity respectively, scatter plots are provided in the main text; see Figures 6.1-6.12.

When implementing BCART models, it is crucial to transform categorical covariates into numeric ones, particularly when a categorical covariate has numerous levels. In such cases, directly using the combinations of its levels can pose computational challenges. Therefore, we transform the categorical covariates at each node based on their empirical claims frequency (or severity, cost), depending on the specific type of analysis we intend to conduct. More discussions on this topic can be found in Subsections 2.1.1 and 2.2.1. Some basic information about the transformed categorical covariates at the root node is available in Appendix D. After the numerical transformation, more useful information emerges. For example, both empirical claims frequency and severity achieve the smallest and largest values for areas D and F respectively (see Table D.3). We shall explore if our model can efficiently capture this kind of information. Moreover, we refer to Omari *et al.* (2018) and Quijano Xacur *et al.* (2019) for more discussions on the same dataset *dataCar*.

Table 6.1: Description of variables in *dataCar*.

Variable	Description	Type
numclaims	number of claims	numeric
exposure	in yearly units, between 0 and 1	numeric
claimcst0	total claim amount for each policyholder	numeric
veh_value	vehicle value, in $10,000s	numeric
veh_age	vehicle age category, 1 (youngest), 2, 3, 4	numeric
agecat	driver age category, 1 (youngest), 2, 3, 4, 5, 6	numeric
veh_body	vehicle body, includes 13 different types coded as HBACK, UTE, STNWG, HDTOP, PANVN, SEDAN, TRUCK, COUPE, MIBUS, MCARA, BUS, CONVT, RDSTR	character
gender	Female or Male	character
area	coded as A B C D E F	character

Table 6.2: Frequencies of the number of claims in *dataCar*.

Number of Claims	0	1	2	3	4	> 4
Frequencies	63232	4333	271	18	2	0

Table 6.3: Summary statistics of the claims severity in *dataCar*.

	Min	Mean	Max	Standard Deviation	Skewness	Kurtosis
Claims Severity	0	131	55922	1024	18	534

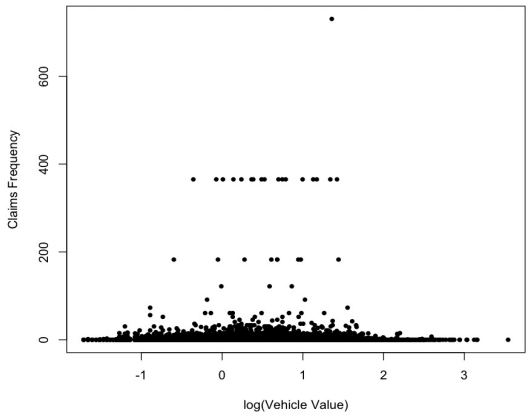

Figure 6.1: Scatter plot between log(vehicle value) and claims frequency in *dataCar*.

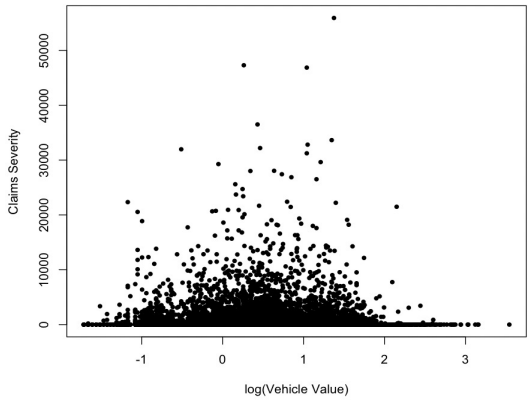

Figure 6.2: Scatter plot between log(vehicle value) and claims severity in *dataCar*.

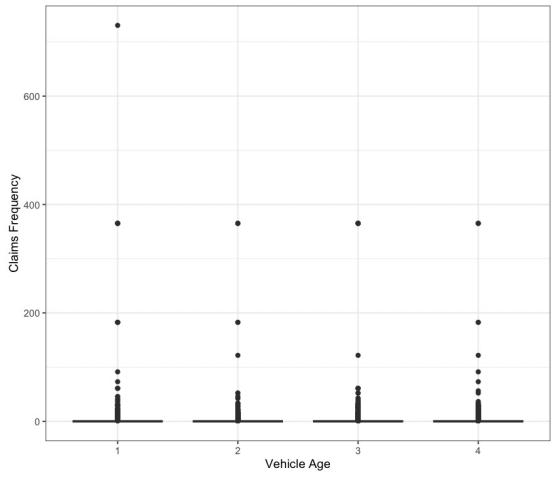

Figure 6.3: Scatter plot between vehicle age and claims frequency in *dataCar*.

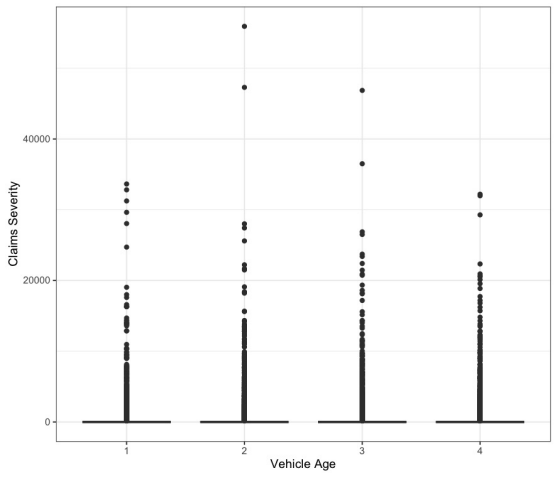

Figure 6.4: Scatter plot between vehicle age and claims severity in *dataCar*.

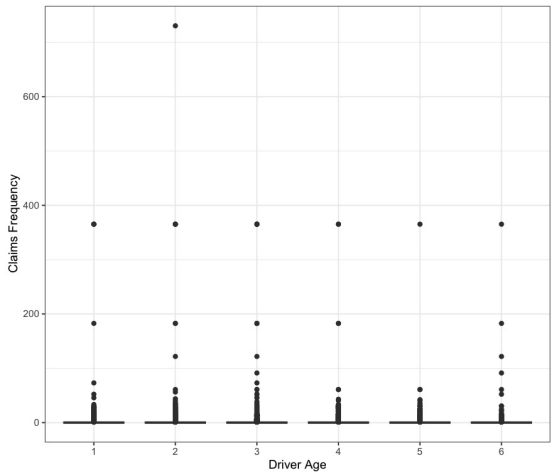

Figure 6.5: Scatter plot between driver age and claims frequency in *dataCar*.

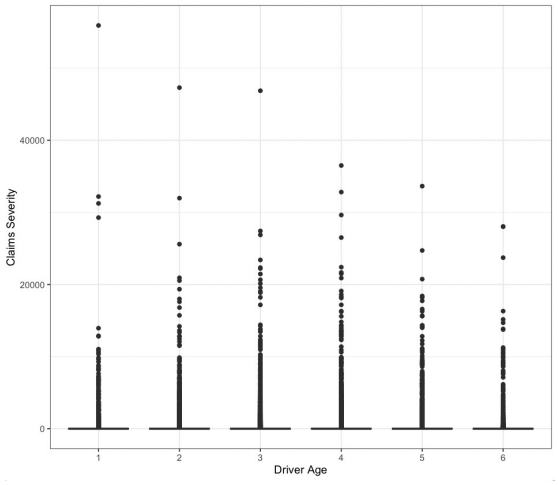

Figure 6.6: Scatter plot between driver age and claims severity in *dataCar*.

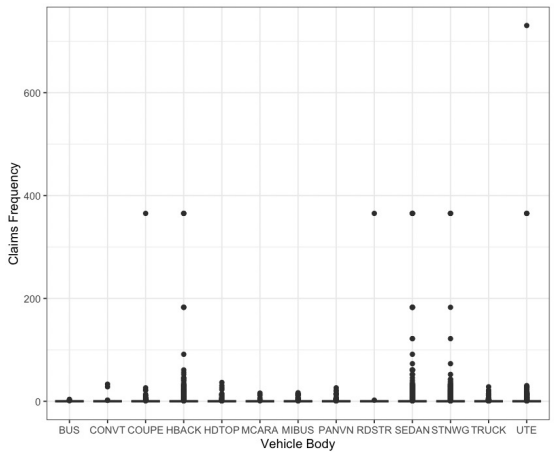

Figure 6.7: Scatter plot between vehicle body and claims frequency in *dataCar*.

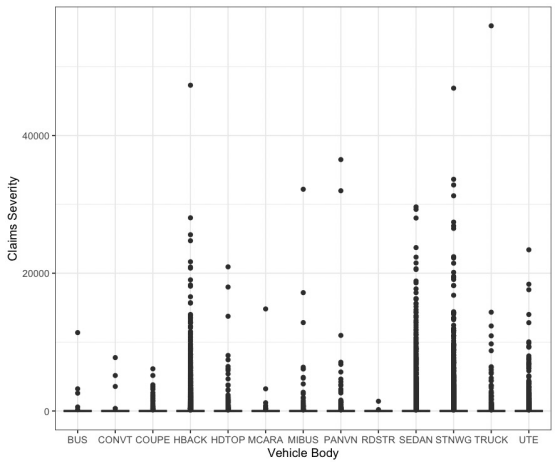

Figure 6.8: Scatter plot between vehicle body and claims severity in *dataCar*.

6.2 Claims Frequency Modelling

We shall apply the BCART models for claims frequency modelling (see Chapter 3) to the training data, where we can use the three-step approach given in Table 2.1 to choose an optimal tree for each model (and also a global optimal one). We also assess the performance of these selected trees on test data.

Running ANOVA-CART on training data, we first use cross-validation to select a base tree size, which has 5 terminal nodes. We also run P-CART in the same way, again resulting in a tree with 5 terminal nodes, and this tree is shown in Figure 6.13. We also apply P-BCART, NB1-BCART, NB2-BCART, ZIP1-BCART, and ZIP2-BCART to the same data. Based on the knowledge learnt from CARTs

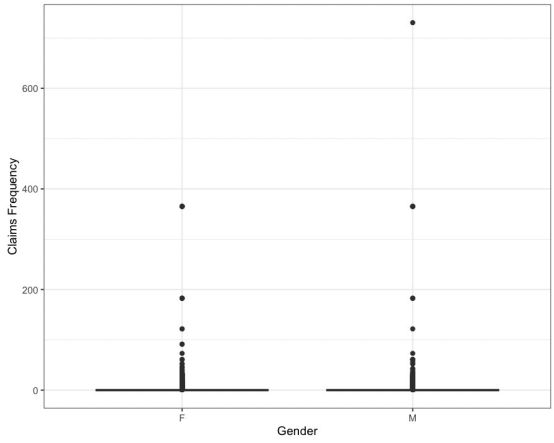

Figure 6.9: Scatter plot between gender and claims frequency in *dataCar*.

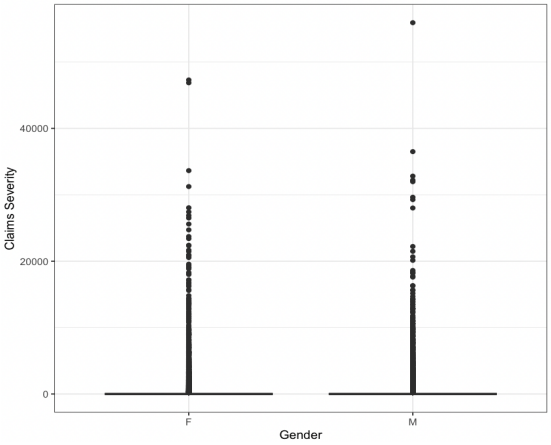

Figure 6.10: Scatter plot between gender and claims severity in *dataCar*.

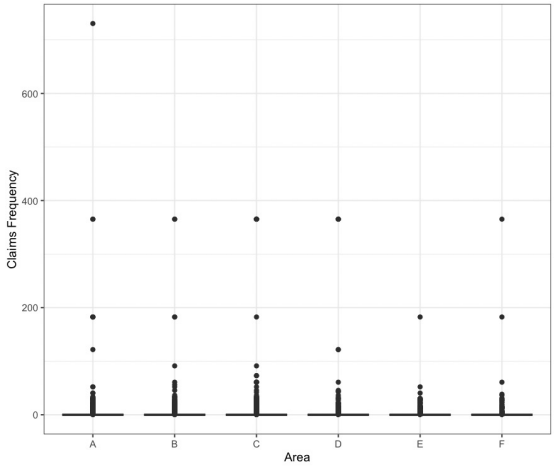

Figure 6.11: Scatter plot between area and claims frequency in *dataCar*.

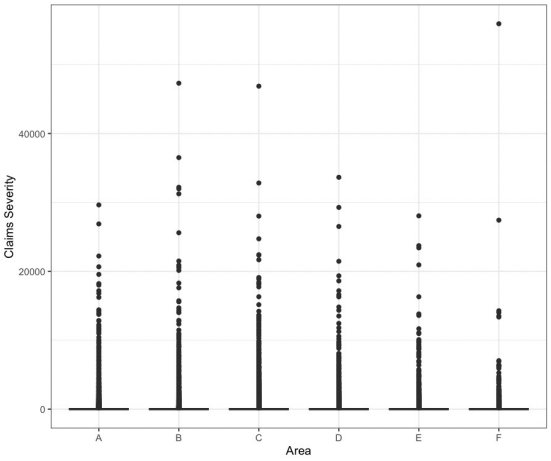

Figure 6.12: Scatter plot between area and claims severity in *dataCar*.

above, we can tune the hyper-parameters γ, ρ (see (2.7)), so that the algorithm will converge to a region of trees with a number of terminal nodes around 5. Some of these, together with the effective number of parameters and DIC, are shown in Table 6.4. We see from this table that all the effective numbers of parameters are reasonable for the model used to fit the data. We conclude from the DIC that all of these BCART models select an optimal tree with 5 terminal nodes using the three-step approach and among these the one from ZIP2-BCART, with the smallest DIC(=25632.5), should be chosen as the global optimal tree to characterize the frequency data.

It is interesting to check if there are the same splitting variables and the same/similar split values/categories used in the trees obtained from different mod-

Table 6.4: Hyper-parameters, p_D (or q_D,r_D) and DIC on training data (*dataCar*) for claims frequency models. The number in the bracket after the abbreviation of the model indicates the number of terminal nodes for this tree. Bold font indicates the DIC selected model.

Model	γ	ρ	p_D(or q_D,r_D)	DIC
P-BCART (4)	0.99	15	4.00	27948.8
P-BCART (5)	0.99	8	5.00	**27943.8**
P-BCART (6)	0.99	6	6.00	27944.4
NB1-BCART (4)	0.99	15	7.98	26002.4
NB1-BCART (5)	0.99	7	9.96	**25892.0**
NB1-BCART (6)	0.99	6	11.96	25945.2
NB2-BCART (4)	0.99	15	7.99	25925.7
NB2-BCART (5)	0.99	6	9.98	**25846.4**
NB2-BCART (6)	0.99	5	11.97	25885.6
ZIP1-BCART (4)	0.99	10	8.05	25688.4
ZIP1-BCART (5)	0.99	5	9.85	**25674.1**
ZIP1-BCART (6)	0.99	3	12.00	25678.3
ZIP2-BCART (4)	0.99	10	7.99	25654.3
ZIP2-BCART (5)	0.99	4	9.91	**25632.5**
ZIP2-BCART (6)	0.99	3	11.93	25641.4

els, including the P-CART, particularly as they all have 5 terminal nodes. For the tree from P-CART illustrated in Figure 6.13, the variable "*agecat*" is first used and then "*veh_value*", followed by "*agecat*" again. The tree from P-BCART (not shown here) also uses "*agecat*" first, but in the following steps, it uses "*veh_value*" and "*veh_body*". The trees from NB1-BCART and NB2-BCART look very similar, and both of them use "*gender*" first and then use "*agecat*", "*veh_value*", and "*veh_body*". Further, the trees from ZIP1-BCART and ZIP2-BCART have the same tree structure and select the same splitting variables as the tree from P-BCART, while the split values/categories are slightly different. The optimal tree from ZIP2-BCART is displayed in Figure 6.14, where the estimated frequency (i.e., the first value in each node) is calculated through (2.18) for the ZIP2 model with unit exposure. Comparing the two trees in Figures 6.13 and 6.14 we see that the ZIP2-BCART model can identify a more risky group (i.e., the one with an estimated frequency equal to 0.2674). Moreover, we also use a Poisson-GLM to fit

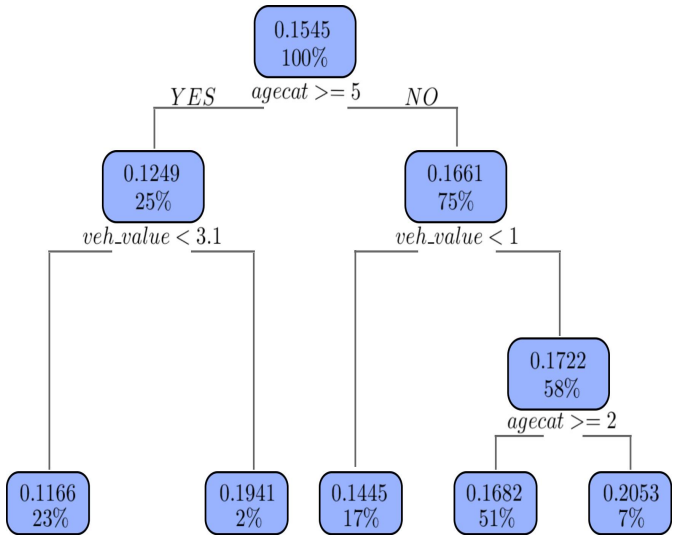

Figure 6.13: Tree from P-CART. Numbers at each node give the estimated frequency and the percentage of observations.

the data. We find that only the variables "*agecat*" and "*veh_body*" are significant, in which we also use the interactions between these two variables. In conclusion, though the variables used in different models can differ slightly, there seems to be a consensus that "*agecat*", "*veh_value*" and "*veh_body*" are relatively significant variables and "*gender*", "*veh_age*" and "*area*" are less significant. This finding is somewhat consistent with our initial analysis of the relationship between covariates and claims frequency, as demonstrated in Figures 6.1 and 6.7, emphasizing the significance of covariates "*veh_value*" and "*veh_body*". Particularly, ZIP2-BCART accurately selects four types of vehicle body (all claims frequency greater than 0.2 after the numerical transformation) for splitting (see Table D.1). This is the reason why ZIP2-BCART can identify a riskier group more effectively. Additionally, BCART models identify another important variable, "*agecat*", which was not initially discovered.

Now, we apply the identified optimal trees to the test data. The performances are given in Table 6.5. We also include the commonly used GLM with Poisson regression models, for which the performance looks not as good as the tree models. From the table, we can conclude that for each of the BCART models, the tree with 5 terminal nodes that is selected by DIC performs better, in terms of SE and DS (see Section 2.3), than the trees with either a smaller or larger number of terminal nodes. This confirms that the proposed three-step approach for the tree model selection in each type of model based on DIC works well in this dataset. Moreover, all the performance measures give the same ranking of models as follows:

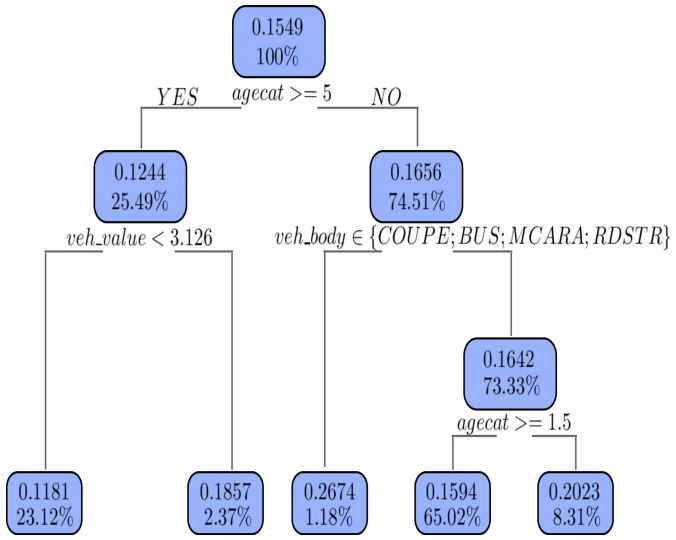

Figure 6.14: Optimal tree from ZIP2-BCART. Numbers at each node give the estimated frequency and the percentage of observations.

ZIP2-BCART, ZIP1-BCART, NB2-BCART, NB1-BCART, P-BCART, P-CART, ANOVA-CART, P-GLM. This ranking is, to some extent, consistent with the conclusions from the simulation examples (see Section 3.5) and as expected. We do not know the exact distribution of real insurance claims frequency data, but we do know that it contains a high proportion of zeros, where the advantage of ZIP comes into play. Further, comparing NB and Poisson distributions, the former is able to handle over-dispersion, so their performance ranking is reasonable. Moreover, the ranking of two ZIP-BCART models and two NB-BCART models are also consistent with the conclusions of Lee (2020, 2021) where it is justified that the non-standard ways of dealing with exposure (i.e., ZIP2-BCART and NB2-BCART) should better fit real insurance data.

In addition to the performance measures, we also record the computation time (in seconds) and memory usage (in megabytes); see the last two columns of Table 6.5. All computations were performed on a laptop with Processor (3.5 GHz Dual-Core Intel Core i7) and Memory (16 GB 2133 MHz LPDDR3). Clearly, BCART models are far inferior to CARTs and GLM in these two respects and as the number of latent variables increases (from P-BCART to NB-CART to ZIP-BCART) these indicators become worse, but we think with such a large training data these are still acceptable and feasible to use in practice. We remark that there have been prior endeavours to address computing issues; see, e.g., Chipman *et al.* (2014), He *et al.* (2019) and Sparapani *et al.* (2021). We believe these two indicators will be

Table 6.5: Model performance on test data (*dataCar*) for claims frequency models with bold entries determined by DIC (see Table 6.4). The number in the bracket after the abbreviation of the model indicates the number of terminal nodes for this tree.

Model	RSS ($\boldsymbol{N}$)	SE	DS	NLL	Lift	Time (s)	Memory (MB)
P-GLM	1057.029	-	-	5532.37	-	1.15	115
ANOVA-CART (5)	1054.061	0.0205	0.0700	5514.06	1.83	2.05	98
P-CART (5)	1042.295	0.0185	0.0681	5476.43	1.97	2.13	98
P-BCART (4)	1042.221	0.0172	0.0680	5473.90	1.74	317.61	364
P-BCART (5)	1042.211	**0.0167**	**0.0602**	5472.86	2.26	291.28	378
P-BCART (6)	1042.205	0.0171	0.0632	5472.27	2.29	325.10	581
NB1-BCART (4)	1041.129	0.0168	0.0445	5470.12	1.80	413.95	628
NB1-BCART (5)	1041.109	**0.0159**	**0.0372**	5469.00	2.46	403.84	569
NB1-BCART (6)	1041.103	0.0162	0.0413	5468.51	2.57	459.70	689
NB2-BCART (4)	1041.127	0.0155	0.0416	5470.01	1.85	431.90	642
NB2-BCART (5)	1041.102	**0.0144**	**0.0352**	5468.68	2.50	441.82	661
NB2-BCART (6)	1041.094	0.0151	0.0390	5468.35	2.58	492.19	721
ZIP1-BCART (4)	1041.102	0.0150	0.0383	5469.07	1.91	548.29	827
ZIP1-BCART (5)	1041.087	**0.0138**	**0.0316**	5468.39	2.56	524.84	792
ZIP1-BCART (6)	1041.075	0.0142	0.0362	5468.02	2.60	569.21	889
ZIP2-BCART (4)	1041.054	0.0145	0.0279	5468.25	2.20	561.98	840
ZIP2-BCART (5)	1041.038	**0.0136**	**0.0241**	5468.01	2.72	570.40	851
ZIP2-BCART (6)	1041.025	0.0141	0.0271	5467.81	2.79	589.24	892

improved after our code is optimized in the future.

We conclude the claims frequency modelling with discussions on the *stability* of the proposed BCART models. Stability is a notion in computational learning theory of how the output of a machine learning algorithm is perturbed by small changes to its inputs. A stable learning algorithm is one for which the prediction does not change much when training data is modified slightly; see, e.g., Arsov *et al.* (2019) and references therein. CART models are known to be unstable. It is thus interesting to examine whether the proposed BCART models can be more stable. To this end, we propose the following approach to assess the stability of the P-CART and ZIP2-BCART (as the best) models.

Step 1: Randomly divide the data into two parts, 80% for training and 20% for

testing.

Step 2: Randomly select 90% of training data for 20 times to construct 20 training subsets, named Data_1, Data_2, ..., Data_{20}.

Step 3: Obtain the optimal tree from P-CART and ZIP2-CART, respectively, for each training subset Data_j, $j = 1, \ldots, 20$.

Step 4: Use the previously obtained trees to get predictions for test data. For each observation in test data, we will have 20 predictions from the 20 P-CART trees for which we calculate the variance, and do the same for the 20 ZIP2-BCART trees to get a variance.

Step 5: Calculate the mean (over the observations on test data) of those variances for P-CART and ZIP2-BCART, respectively.

Since variance can capture the amount of variability, we shall use the above obtained mean of variance to assess the stability (in their prediction ability) of a tree-based model. Namely, the smaller the mean the more stable the model that was used to calculate it. We apply it to the *dataCar* insurance data, the calculated mean for P-CART is 9.3×10^{-5} and for ZIP2-BCART, it is 6.9×10^{-5}. This implies that ZIP2-BCART is more stable than P-CART. Additionally, we also compare the 20 trees from P-CART, where we can observe very different trees in terms of number of terminal nodes (ranging from 3 to 8) and splitting variables selected in the trees. Whereas, the 20 trees from ZIP2-BCART also show some stability regarding the number of terminal nodes (all around 5) and splitting variables selected. The same procedure has also been applied to other BCART models and the conclusions are almost the same. Therefore, we conclude from our studies that the proposed BCART models in this thesis show some stability that the CART models may not possess.

6.3 Claims Severity Modelling

Based on Chapter 4, we aim to directly model the average claim amount $\bar{S}$ for claims severity. The claims severity model considered here only applies to the subset of the data for which the policyholder has at least one claim. This is a traditional and commonly used way to deal with claims severity data; see, e.g., Henckaerts *et al.* (2021) and Frees *et al.* (2014). Among all 67,856 policies, 4,624 policies satisfy this requirement (3,699 in the training data, and 925 in the test

data). Two graphs are used to illustrate which distribution better fits the claims severity data in Figures 6.15 and 6.16. At first glance, distinguishing their performance is challenging in Figure 6.15 since all of them capture the right-skewed feature. However, in Figure 6.16, none of the three fitted distributions correctly describes the right tail of the distribution. The LogNormal distribution might be preferred, as the points from the LogNormal distribution form a line that is closest to the 45-degree reference line. This finding is somewhat surprising and goes against our expectations. We expected that the Weibull distribution would be the best among these three, as discussed in Section 4.5. The reason may be that the plots are drawn by using parameters estimated once with the entire claims severity data, whereas in our proposed BCART models, we update the parameters in the Bayesian framework at every step using subsets of data in each node. The approach we use not only provides more accurate parameter estimates but also allows the model to better capture data characteristics and fit the data more effectively in each group (terminal node).

First, when running ANOVA-CART on training data, the result indicates there is no potential split that would lead to improvement. Apparently, ANOVA-CART is not capable of capturing this right-skewed and heavy-tailed insurance data (see Table 6.3). Another approach involves running ANOVA-CART on $\log(\bar{S})$ and transforming the results back to $\bar{S}$. This method is intuitive and straightforward to implement. However, when examining the results, there is still no split identified, resulting in a root node tree. Consequently, we do not include both of them in the following context. Next, we run Gamma-CART, implemented using the R package `distRforest`. This package extends the applicable distributions for decision trees to include Gamma and LogNormal distributions, allowing the analysis of light/heavy-tailed data. It builds a random forest in the ensemble consisting of individual CARTs based on the package `rpart`; see more details in Henckaerts (2020). Since our focus is on one tree, we restrict the number of trees to one in the random forest, allowing us to obtain Gamma-CART. We use cross-validation to select the tree size, which has 5 terminal nodes. Similarly, we run LN-CART in the same way, again resulting in a tree with 5 terminal nodes. Regarding the Weibull distribution, no package can be used to implement it in decision trees currently, so we do not include it in the following context. Further research may be needed to include more distributions with different characteristics in decision trees. We then apply the proposed Gamma-BCART, LN-BCART, and Weib-BCART to the same data. Similar to the previous claims frequency modelling, we tune the hyper-parameters γ, ρ. The DICs displayed in Table 6.6 indicate that

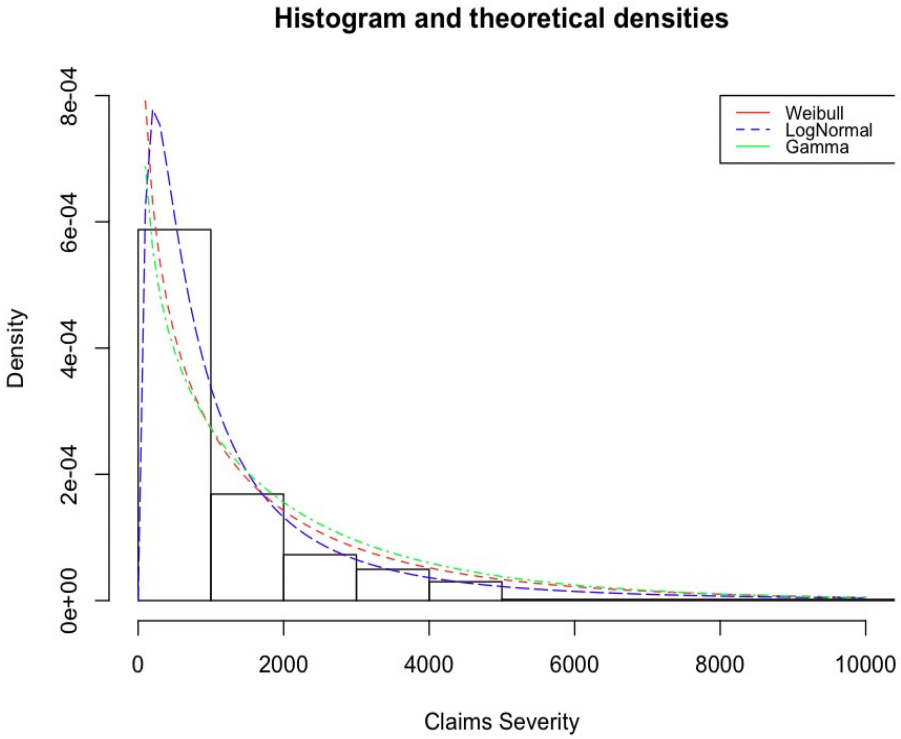

Figure 6.15: Histogram and theoretical densities of Gamma, Log-Normal, and Weibull distributions for claims severity data. Parameters used to generate the plot are estimated by using the *"fitdist"* function in the R package `fitdistrplus` (see more details in Marie-Laure Delignette-Muller & Pouillot (2023)).

all these BCART models choose an optimal tree with 4 terminal nodes, one less than those obtained from CARTs. In Figure 6.17, we present trace plots for trees around $h = 4$ terminal nodes for Gamma-BCART, including plots of the number of terminal nodes, the integrated likelihood $p_{\mathrm{G}}(\bar{\boldsymbol{S}}|\boldsymbol{X}, \boldsymbol{\mathcal{T}})$, and the data likelihood $p_{\mathrm{G}}(\bar{\boldsymbol{S}}|\boldsymbol{X}, \hat{\boldsymbol{\alpha}}, \bar{\boldsymbol{\beta}}, \boldsymbol{\mathcal{T}})$ of the accepted trees. The figure illustrates that although the number of terminal nodes fluctuates between $h = 3$ and $h = 4$, and occasionally jumps to $h = 5$, it predominantly remains at $h = 4$. Besides, the data likelihood and integrated likelihood exhibit minor differences with a similar trend, which is in line with previous conclusions (see more discussions in Subsection 3.5.1). The trace plots of LN-BCART and Weib-BCART have a similar pattern and are thus omitted. Based on Table 6.6, among these BCART models, Weib-BCART, with the smallest DIC(=77646), should be selected as the global optimal tree to characterize the claims severity data.

Similar to claims frequency modelling, we also examine the splitting variables

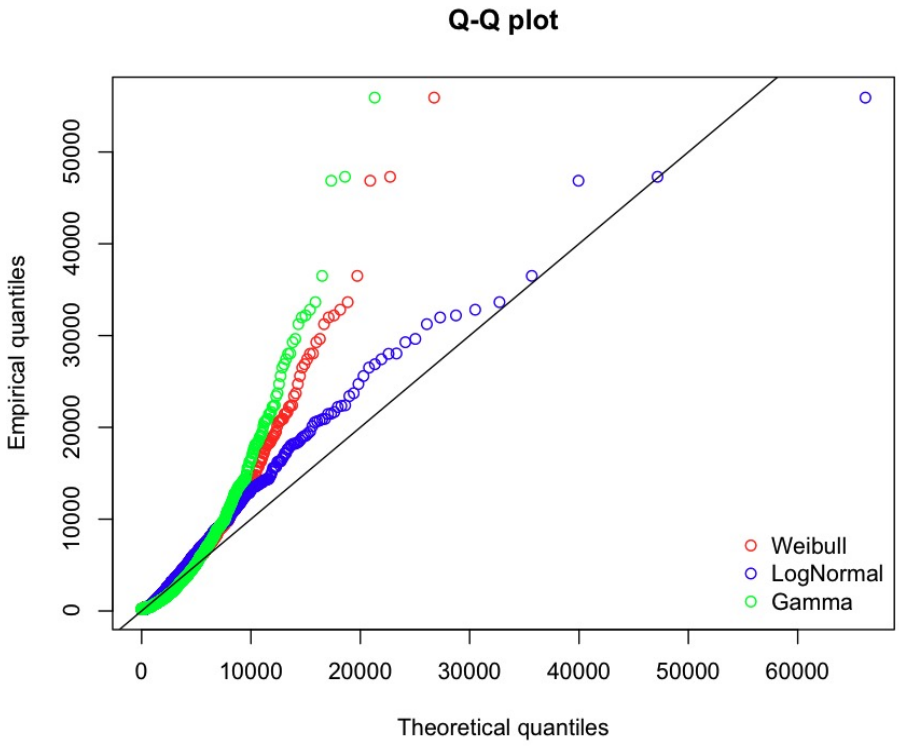

Figure 6.16: Q-Q plot of Gamma, LogNormal, and Weibull distributions for claims severity data. The parameters used to generate the plot are estimated by using the *"fitdist"* function in the R package `fitdistrplus` (see more details in Marie-Laure Delignette-Muller & Pouillot (2023)).

and corresponding split values/categories used in the trees obtained from different models. For Gamma-CART, it uses both "*agecat*" and "*veh_value*" twice, with the first one being "*agecat*". In contrast, LN-CART uses three different variables, "*veh_value*" first, followed by "*veh_body*" and "*area*". All trees from BCART models, i.e., Gamma-BCART, LN-BCART, and Weib-BCART, have the same tree structure and splitting variables ("*agecat*", "*veh_value*", and "*area*"), while the split values/categories are slightly different. Weib-BCART, in particular, identifies a more risky group (i.e., the one with estimated severity equal to 2743.41); see Figure 6.18. This may be because, as discussed in Section 4.5, Weib-BCART can flexibly control the shape parameter to adapt to data with different tail characteristics, allowing it to handle cases where some groups (terminal nodes) have lighter tails, and others have heavier tails. In Figure 6.19, we observe that although all shape parameters are smaller than one, indicating heavy tails for claims severity data within each terminal node, the optimal Weib-BCART tree

Table 6.6: Hyper-parameters, s_D and DIC on training data (*dataCar*) for claims severity models. The number in the bracket after the abbreviation of the model indicates the number of terminal nodes for this tree. Bold font indicates the DIC selected model.

Model	γ	ρ	s_D	DIC
Gamma-BCART (3)	0.99	4	5.97	78061
Gamma-BCART (4)	0.99	3.5	7.97	**77779**
Gamma-BCART (5)	0.99	2	9.95	77982
LN-BCART (3)	0.99	5	5.98	78014
LN-BCART (4)	0.99	4	7.97	**77723**
LN-BCART (5)	0.99	3	9.97	77889
Weib-BCART (3)	0.99	7	5.98	77932
Weib-BCART (4)	0.99	5	7.98	**77646**
Weib-BCART (5)	0.99	4	9.98	77821

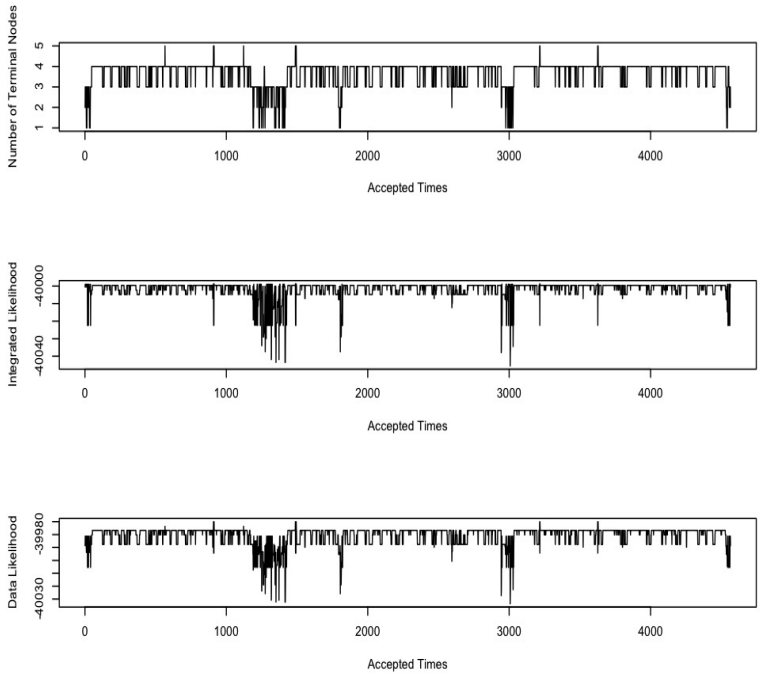

Figure 6.17: Trace plots from MCMC with 3 restarts for Gamma-BCART (γ=0.99, ρ=3.5).

shows improved data fitting compared to Figure 6.16. We also conduct similar analyses for Gamma-BCART and LN-BCART (with their Q-Q plots not shown here). Particularly, for Gamma-BCART, the range of shape parameters is 0.62-

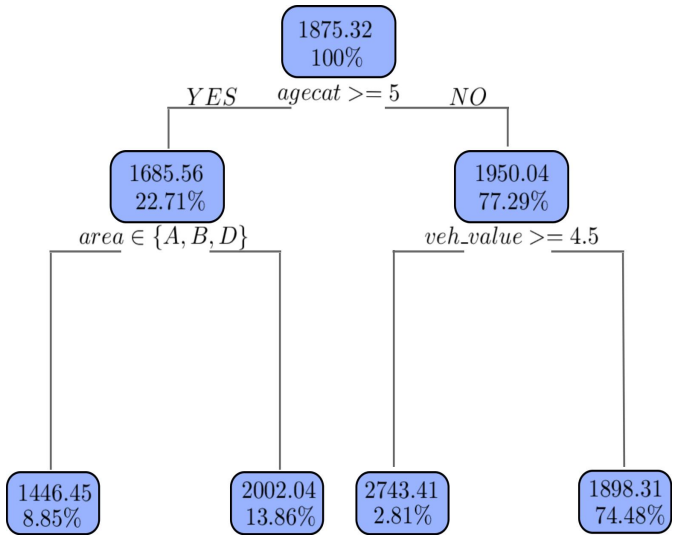

Figure 6.18: Optimal tree from Weib-BCART. Numbers at each node give the estimated severity and the percentage of observations.

0.89, whereas Weib-BCART can distinguish between 0.48-0.96, allowing for better tail control and data fitting. Moreover, we use a Gamma-GLM to fit the data. We find that only the variable "*gender*" is significant, and thus no interaction is considered in the Gamma-GLM. Interestingly, "*gender*" does not appear in all CART and BCART models. In summary, though the variables used for different models may differ, there seems to be a consensus that "*agecat*" is still significantly important for claims severity modelling, as Gamma-CART and all BCART models use it in the first split, and "*veh_value*" is another relatively significant variable. This discovery aligns, to some extent, with our initial analysis of the relationship between covariates and claims severity. Particularly, in comparison to CARTs, BCART models reveal another important variable, "*area*", which was identified after the initial numerical transformation (see Table D.3). Notably, Weib-BCART precisely chooses *areas* A, B, and D (identified as having the three smallest claims severity after the initial numerical transformation) for splitting.

Now, we apply the identified optimal trees to the test data. The performances are given in Table 6.7. It is evident that the Gamma-GLM does not exhibit good performance compared to the tree models. From the table, we also conclude that, for each of the BCART models, the tree with 4 terminal nodes selected by DIC outperforms those with either a smaller or larger number of terminal nodes in terms of SE and DS. This validates the effectiveness of the proposed three-step approach for selecting tree models based on DIC, as observed in claims frequency modelling.

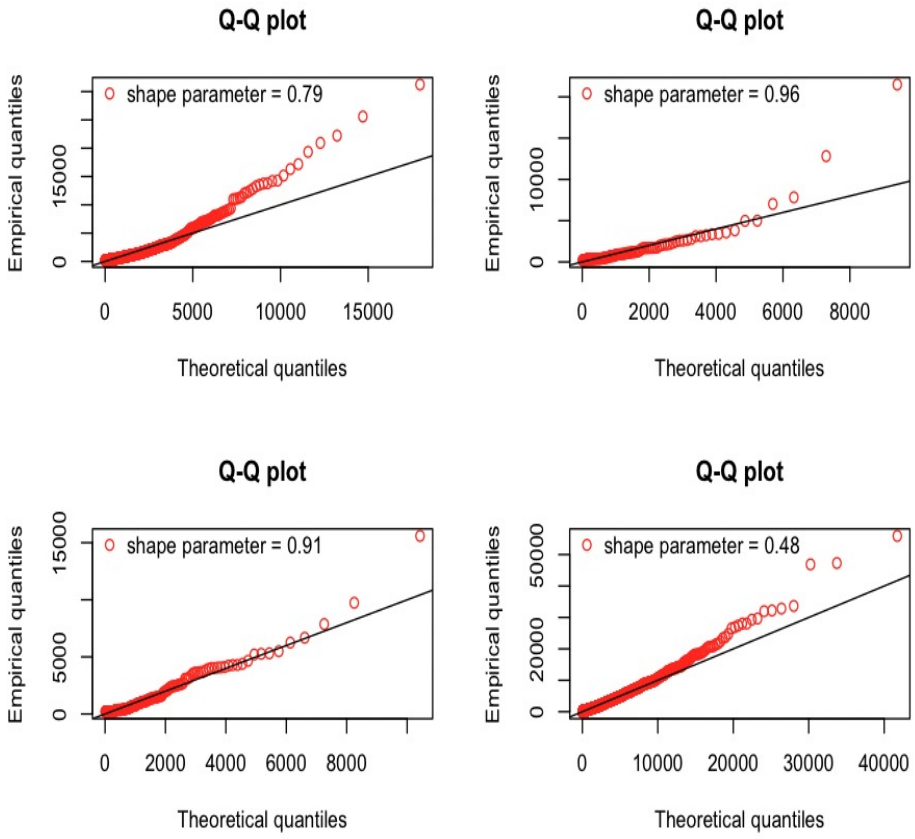

Figure 6.19: Q-Q plots of the Weibull distribution for claims severity data in each terminal node of the optimal Weib-BCART tree. The shape parameter used to generate the plot is estimated by using MLE (see Remark 4.3 (a) in Section 4.4), and the scale parameter is estimated using the posterior distribution (see (4.27)).

For CARTs, having one more terminal node than all optimal trees for each type of BCART model makes an unfair direct comparison using RSS($\boldsymbol{S}$), NLL, and lift, as these measures can usually be improved with more splits (see Section 2.3). However, despite this, the values for these measures are very close between CART and BCART models. In such cases, a more parsimonious model (the tree from BCART models) should be preferred. Regarding SE and DS, CARTs perform worse compared to the optimal tree for each type of BCART model (for example, the comparison between Gamma-CART (5) and Gamma-BCART (4)). Therefore, we obtain a model ranking: Weib-BCART, LN-BCART, Gamma-BCART, LN-CART, Gamma-CART, Gamma-GLM. This ranking is consistent with the conclusions from the simulation example in Section 4.5 and our expectations. Although we lack information about the exact distribution of real insurance claims

Table 6.7: Model performance on test data (*dataCar*) for claims severity models with bold entries determined by DIC (see Table 6.6). The number in the bracket after the abbreviation of the model indicates the number of terminal nodes for this tree.

Model	RSS($\boldsymbol{S}$) (in 10^{10})	SE	DS	NLL	Lift	Time (s)	Memory (MB)
Gamma-GLM	1.4335	-	-	13769	-	0.21	15
Gamma-CART (5)	1.4173	464	0.00171	13693	1.625	1.13	5
LN-CART (5)	1.4168	458	0.00168	13685	1.629	1.42	6
Gamma-BCART (3)	1.4201	486	0.00181	13712	1.567	49.89	94
Gamma-BCART (4)	1.4176	**457**	**0.00154**	13698	1.615	51.53	97
Gamma-BCART (5)	1.4158	472	0.00167	13690	1.643	50.12	96
LN-BCART (3)	1.4189	482	0.00176	13708	1.572	55.68	99
LN-BCART (4)	1.4170	**451**	**0.00145**	13692	1.633	57.11	101
LN-BCART (5)	1.4152	463	0.00159	13683	1.651	57.34	101
Weib-BCART (3)	1.4177	473	0.00164	13700	1.604	58.23	102
Weib-BCART (4)	1.4154	**433**	**0.00131**	13684	1.661	60.01	105
Weib-BCART (5)	1.4136	446	0.00144	13673	1.693	61.87	106

severity data, we are aware of its right-skewed and heavy-tailed nature. Specifically, in tree models, data within groups (terminal nodes) may exhibit different tail characteristics, highlighting the advantage of the Weibull distribution, which can effectively handle varying tail characteristics. With respect to time and memory, similar discussions can be considered as in previous claims frequency modelling.

For stability analysis, the same conclusions can be drawn for claims severity modelling: BCART models are more stable than CART models. To avoid redundancy, we omit the details here.

6.4 Aggregate Claims Modelling

Based on Chapter 5, three types of models can be employed for aggregate claims modelling: frequency-severity models, sequential models, and joint models. In the case of frequency-severity models, numerous combinations arise from claims frequency and claims severity models. In our implementation, there are 5 models for claims frequency (1 Poisson, 2 NB, and 2 ZIP) and 4 models for claims severity (2 Gamma, 1 LogNormal, and 1 Weibull), resulting in 20 frequency-severity

models. Instead of running all these models, we propose using the optimal trees found (ZIP2-BCART and Weib-BCART) in the previous Sections 6.2 and 6.3 as one combination for frequency-severity models. Although ZIP2-BCART and Weib-BCART are identified as the best for claims frequency and claims severity respectively, it is uncertain if they remain optimal when combined. This can be examined in the real insurance dataset. Additionally, given that the proposed joint models employing a CPG distribution, we include the frequency-severity models with Poisson and Gamma distributions in the comparison. Furthermore, three ZICPG distributions in joint models are compared to assess their capability to capture data characteristics with a high proportion of zeros. For sequential models, we ensure consistency of claims frequency modelling as the other models by using Poisson-BCART and ZIP2-BCART. Subsequently, we treat the claim count N_i (or $\hat{N}_i$) as a covariate in the corresponding Gamma-BCART and Weib-BCART for claims severity modelling. The resulting models are called Gamma1-BCART (or Gamma2-BCART) and Weib1-BCART (or Weib2-BCART).

From Table 6.8, it is evident that, for both Gamma and Weibull distributions with N_i (or $\hat{N}_i$) as a covariate, optimal trees are chosen with 4 terminal nodes consistently. Upon inspecting the tree structure, N_i (or $\hat{N}_i$) is indeed used in the first step in all optimal trees for each model (Gamma1-BCART, Gamma2-BCART, Weib1-BCART, and Weib2-BCART). All of them replace the previously used variable "*agecat*", with the only difference being in the split values used. This observation aligns with the results obtained in Subsection 5.2.1. We suspect that the reason for this may be a strong relationship between the covariates N_i and "*agecat*", as verified in the claims frequency analysis (see Section 6.2), where the optimal claims frequency tree selects "*agecat*" in the first split, indicating a strong relationship between N_i and "*agecat*". Consequently, it is reasonable to replace "*agecat*" with N_i (or $\hat{N}_i$) to avoid multicollinearity. Furthermore, by comparing the DIC of all Gamma-BCART and Weib-BCART models in Tables 6.6 and 6.8 (with/without N_i or $\hat{N}_i$ as a covariate), we can conclude that the model performance improves when considering N_i (or $\hat{N}_i$) as a covariate, especially when using $\hat{N}_i$. This approach has a practical advantage as there is no direct information about N_i itself for new customers. For joint models, i.e., CPG-BCART and three ZICPG-BCART models, all of them choose optimal trees with 5 terminal nodes. Among them, ZICPG3-BCART, with the smallest DIC (=102120), is deemed the best. The difference in DIC between the CPG model and ZICPG models is significantly larger than the difference between ZICPG models themselves, illustrating the necessity of considering the zero mass part. Additionally, there is no big

difference between ZICPG2-BCART and ZICPG3-BCART models, aligning with the conclusion obtained in ZIP models (see Remark 3.5 (b) in Subsection 3.3.4). This observation implies that embedding the exposure into both the Poisson part and the zero mass part does not yield substantial improvement; embedding the exposure into the zero mass part is sufficient. However, it cannot be denied that ZICPG3-BCART still exhibits the best performance.

Similar to the previous analysis, we examine the splitting variables and the corresponding split values/categories used in the trees obtained from joint models. All models use the same splitting variables ("*agecat*", "*veh_value*", "*veh_body*", and "*area*"), but the order of use and the tree structures vary. Notably, "*agecat*" remains the first variable used in all models. Among them, ZICPG3-BCART demonstrates the ability to identify a riskier group (i.e., the one with an estimated premium equal to 657.45; see Figure 6.20), possibly for the same reasons discussed in Section 6.2 for the outstanding performance of ZIP2-BCART. Both ZIP2 and ZICPG3 models exhibit the capacity to handle data sets with a substantial number of zeros by incorporating exposure in the zero mass part. Besides, we observe that the tree structure of ZICPG3-BCART is quite similar to ZIP2-BCART. However, ZICPG3-BCART identifies another important variable "*area*", which was recognized as important for claims severity before. Furthermore, we fit a CPG-GLM to the data. We find that only the variable "*agecat*" is significant, aligning with its consistent selection as the first splitting variable in almost all BCART models. Since only one variable is deemed significant in the CPG-GLM, no interactions were included. It is also worth mentioning that CART is not included in this analysis due to the absence of R packages on decision trees that can directly use CPG to process the data.

Now, we apply the identified optimal trees to the test data. The performances are given in Table 6.9. As before, GLM exhibits poorer performance compared to tree-based models, as evidenced by RSS($\boldsymbol{S}$) and NLL. However, for other models, drawing a clear and unified conclusion is challenging, and thus we discuss this from various perspectives.

1. A comparison of the different frequency-severity models reveals that it is indeed preferable to use the combination of two best models for claims frequency and claims severity respectively, i.e., $\mathrm{F}_{\mathrm{ZIP2}}\mathrm{S}_{\mathrm{Weib}}$-BCART > $\mathrm{F}_{\mathrm{P}}\mathrm{S}_{\mathrm{G}}$-BCART. In the sequential models, the same conclusion as in Subsection 5.2.1 is reached: using the prediction of the claim count $\hat{N}_i$ is superior to using N_i itself when treating them as a covariate in the claims severity tree.

Table 6.8: Hyper-parameters, r_D (or s_D) and DIC on training data (*dataCar*) for aggregate claims models. The number in the bracket after the abbreviation of the model indicates the number of terminal nodes for this tree. The Gamma1/Weib1 and Gamma2/Weib2 models treat the claim count N_i and $\hat{N}_i$ as a covariate respectively, where $\hat{N}_i$ comes from Poisson-BCART and ZIP2-BCART respectively. Bold font indicates DIC selected model. This table only helps to select the optimal tree, and DICs between different models cannot be directly compared.

Model	γ	ρ	r_D (or s_D)	DIC
Gamma1-BCART (3)	0.99	4	5.97	78032
Gamma1-BCART (4)	0.99	3.5	7.97	**77750**
Gamma1-BCART (5)	0.99	2	9.96	77854
Gamma2-BCART (3)	0.99	4	5.98	78024
Gamma2-BCART (4)	0.99	3.5	7.97	**77743**
Gamma2-BCART (5)	0.99	2	9.97	77849
Weib1-BCART (3)	0.99	7	5.98	77911
Weib1-BCART (4)	0.99	5	7.98	**77619**
Weib1-BCART (5)	0.99	4	9.98	77804
Weib2-BCART (3)	0.99	7	5.98	77893
Weib2-BCART (4)	0.99	5	7.98	**77608**
Weib2-BCART (5)	0.99	4	9.98	77787
CPG-BCART (4)	0.99	10	11.96	105710
CPG-BCART (5)	0.99	8	14.93	**105626**
CPG-BCART (6)	0.99	7	17.92	105643
ZICPG1-BCART (4)	0.99	11	15.97	102314
ZICPG1-BCART (5)	0.99	10	19.95	**102198**
ZICPG1-BCART (6)	0.99	7.5	23.92	102225
ZICPG2-BCART (4)	0.99	12	15.95	102265
ZICPG2-BCART (5)	0.99	11	19.94	**102134**
ZICPG2-BCART (6)	0.99	8	23.92	102167
ZICPG3-BCART (4)	0.99	14	15.94	102247
ZICPG3-BCART (5)	0.99	12	19.93	**102120**
ZICPG3-BCART (6)	0.99	9	23.90	102158

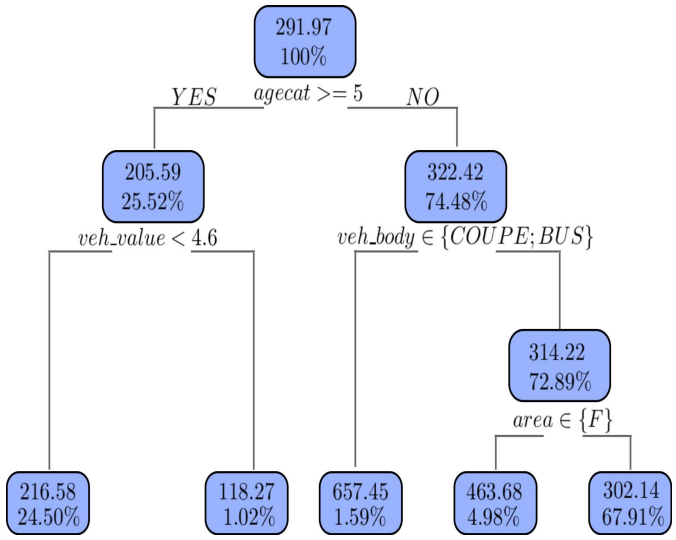

Figure 6.20: Optimal tree from ZICPG3-BCART. Numbers at each node give the estimated premium and the percentage of observations.

Regarding joint models, ZICPG models outperform the CPG model, with ZICPG3-BCART being the best. These conclusions align with those obtained in the training data.

2. When comparing frequency-severity models and sequential models, it is evident that adding N_i (or $\hat{N}_i$) as a covariate improves performance, i.e., $\text{F}_\text{P}\text{S}_\text{G2}$-BCART > $\text{F}_\text{P}\text{S}_\text{G1}$-BCART > $\text{F}_\text{P}\text{S}_\text{G}$-BCART. The same ranking is observed for another combination, i.e., $\text{F}_\text{ZIP2}\text{S}_\text{Weib2}$-BCART > $\text{F}_\text{ZIP2}\text{S}_\text{Weib1}$-BCART > $\text{F}_\text{ZIP2}\text{S}_\text{Weib}$-BCART. This is reasonable, as real data often exhibits a correlation between the number of claims and claims severity (see Garrido *et al.* (2016)), favouring sequential models that consider this correlation over frequency-severity models assuming independence.

3. In comparing frequency-severity models and joint models, $\text{F}_\text{P}\text{S}_\text{G}$-BCART and CPG-BCART (or ZICPG-BCART) are intuitive to examine as they use the same (or similar) distributions. CPG-BCART (or ZICPG-BCART) consistently outperforms $\text{F}_\text{P}\text{S}_\text{G}$-BCART, suggesting that sharing information is necessary for this dataset, and one joint tree exhibits better performance. Further exploration of the reasons is provided below. However, for $\text{F}_\text{ZIP2}\text{S}_\text{Weib}$-BCART and CPG-BCART (or ZICPG-BCART), there is no clear intuition due to the use of different distributions. Performance rankings, based on evaluation metrics, conclude that joint models are superior.

Table 6.9: Model performance on test data (*dataCar*) for aggregate claims models with bold entries determined by DIC (see Table 6.8). The Gamma1/Weib1 and Gamma2/Weib2 models treat the claim count N_i and $\hat{N}_i$ as a covariate respectively, where $\hat{N}_i$ comes from Poisson-BCART and ZIP2-BCART respectively. $\mathrm{F_PS_G}$ represents frequency-severity models using Poisson and Gamma distributions separately, similar to other models. The number in the bracket after the abbreviation of the model indicates the number of terminal nodes for this tree. Particularly, two numbers for frequency-severity models indicate the number of terminal nodes for each tree.

Model	RSS ($\boldsymbol{S}$) (in 10^{10})	NLL	Time (s)	Memory (MB)
CPG-GLM	1.5187	19305.1	2.66	129
$\mathrm{F_PS_G}$-BCART (5/4)	1.4874	19170.9	342.81	475
$\mathrm{F_PS_{G1}}$-BCART (5/4)	1.4813	19144.2	350.18	481
$\mathrm{F_PS_{G2}}$-BCART (5/4)	1.4798	19140.2	349.78	479
$\mathrm{F_{ZIP2}S_{Weib}}$-BCART (5/4)	1.4844	19152.0	630.41	956
$\mathrm{F_{ZIP2}S_{Weib1}}$-BCART (5/4)	1.4790	19138.5	639.55	962
$\mathrm{F_{ZIP2}S_{Weib2}}$-BCART (5/4)	1.4779	19135.2	640.05	963
CPG-BCART (4)	1.4791	19139.0	301.58	401
CPG-BCART (5)	1.4781	19135.5	310.41	406
CPG-BCART (6)	1.4778	19134.9	308.17	404
ZICPG1-BCART (4)	1.4670	19076.3	598.14	968
ZICPG1-BCART (5)	1.4497	19061.2	604.23	991
ZICPG1-BCART (6)	1.4478	19058.1	605.76	990
ZICPG2-BCART (4)	1.4612	19062.9	630.63	966
ZICPG2-BCART (5)	1.4434	19044.3	640.87	1023
ZICPG2-BCART (6)	1.4417	19040.6	638.11	1018
ZICPG3-BCART (4)	1.4598	19055.6	649.56	970
ZICPG3-BCART (5)	1.4415	19040.2	665.22	1012
ZICPG3-BCART (6)	1.4409	19037.4	659.91	1010

4. As for sequential models and joint models, they address dependence in different ways. The former uses two trees, treating the number of claims as a covariate in claims severity modelling to address the dependence issue. In

contrast, the latter uses one joint tree, potentially hiding some dependence in the common variables used to split the nodes and incorporating the number of claims as model weights in the aggregate claim amount distribution. Their performance ranking is obtained through evaluation metrics. Joint models employing ZICPG distributions perform better than all sequential models, possibly due to a small conditional correlation between the number of claims and claims severity (-0.0336) and the dataset involving a high proportion of zeros (93.19%). There is no unified conclusion for joint models employing CPG distributions and sequential models (for example, $F_{ZIP2}S_{Weib2}$ is better than CPG-BCART but $F_{ZIP2}S_{Weib1}$ is worse), indicating the need for further exploration, especially for data with high dependence between the number of claims and claims severity.

To investigate why one joint tree performs better in this dataset, we discuss two aspects. The first perspective focuses on the relationship between covariates and claims frequency (or claims severity). The intuition is that if the majority of covariates exhibit strong relationships with both claims frequency and claims severity, it is beneficial to share this information. In previous Sections 6.2 and 6.3, we discovered that the same two variables are used in optimal claims frequency and claims severity trees, namely, "*agecat*" and "*veh_value*". These two variables account for 2/3 of all splitting variables, indicating a high rate. Moreover, both trees choose "*agecat*" in the first step, illustrating its importance for both claims frequency and claims severity. On the other hand, we can calculate the correlation coefficient numerically. However, as several covariates are categorical, they cannot be used directly in calculations. We use the transformed categorical variables (see Appendix D) to do this; see Table 6.10. For claims frequency, the three variables with the strongest correlation coefficients are used in the optimal tree. For claims severity, "*veh_age*" has the third strongest correlation, but the optimal severity tree does not use it. We explore a strong relationship between covariates "*veh_age*" and "*veh_value*" (see Figure D.1 in Appendix D), so it is reasonable to only use one of them, which has a stronger correlation with the response variable to avoid multicollinearity. Additionally, both claims frequency and claims severity show the strongest correlation with *"agecat"*, illustrating the effectiveness of BCART models for variable selection.

The second perspective involves using ARI (see Subsection 5.3.3 and Appendix C) to assess the similarity between different trees and determine whether information sharing is necessary; see Table 6.11. As discussed in Subsection 5.3.3,

Table 6.10: Correlation coefficients between covariates (numerical ones and transformed categorical ones) and claims frequency (or severity). Bold font indicates the largest correlation coefficient (although they are very small) in each row.

	veh_value	veh_age	agecat	veh_body	gender	area
Claims Frequency	-0.0047	0.0013	**-0.0131**	0.0022	0.0008	-0.0021
Claims Severity	0.0135	-0.0059	**-0.0274**	-0.0035	0.0003	-0.0034

Table 6.11: Values of ARI between different trees. The number in the bracket after the abbreviation of the model indicates the number of terminal nodes for this tree.

	Poisson-BCART (5)	ZIP2-BCART (5)	Gamma-BCART (4)	Weib-BCART (4)	CPG-BCART (5)	ZICPG3-BCART (5)
Poisson-BCART (5)	1	0.7396	0.5398	0.5163	0.8585	0.7823
ZIP2-BCART (5)	-	1	0.5599	0.5179	0.8587	0.8152
Gamma-BCART (4)	-	-	1	0.7430	0.6921	0.6423
Weib-BCART (4)	-	-	-	1	0.6491	0.6022
CPG-BCART (5)	-	-	-	-	1	0.6351
ZICPG3-BCART (5)	-	-	-	-	-	1

although a specific threshold value of ARI for making a direct judgment about the need for information sharing in one joint tree is unknown, it is evident that ARI values between all claims frequency and claims severity trees are greater than 0.5. This suggests significant similarities that cannot be ignored. Therefore, based on these two aspects discussed, we conclude that one joint tree performs better than two trees (in frequency-severity models) in this dataset.

After conducting a thorough analysis of this dataset, we conclude that insurers need to pay more attention to policyholders who are younger and have vehicles with higher values since they are more likely to incur higher risks. In addition, policyholders with specific vehicle body types (such as "*COUPE*" and "*BUS*"), also require additional attention.

Remark 6.1 *We also use other datasets (such as dataOhlsson included in the library* `insuranceData` *in R, freMTPL2freq and freMTPL2sev included in the library* `CASdatasets`*; see more details in Christophe Dutang (2020)), to validate*

our proposed BCART models. Due to the similarity in analysis methods and the consistency of conclusions, we omit the details.

6.5 Summary of Chapter 6

In this chapter, the real dataset *dataCar* was used to validate our proposed BCART models. We found consistent conclusions with previous simulation examples and the conclusions of Omari *et al.* (2018) and Quijano Xacur *et al.* (2019). Given the multitude of available distributions in the insurance industry, it is impractical to run all models each time to determine the best one. Drawing on insights from both simulated and real data, we provide the following recommendations for insurance industry modellers.

1. In claims frequency analysis, a similar set of important variables was obtained by our proposed BCART models and Bayesian GLM models (see Quijano Xacur *et al.* (2019)). Additionally, we extended our consideration to zero-inflated models compared with Omari *et al.* (2018) and observed that the ZIP2 model, which embeds exposure into the zero mass part, outperforms others. Therefore, modellers should prioritize the ZIP2 model when working with data containing a substantial number of zeros.

2. In claims severity analysis, Quijano Xacur *et al.* (2019) employing Bayesian GLM models concludes the same variables chosen by BCART models are important. However, our finding contradicts the conclusion of Omari *et al.* (2018), which suggests that the LogNormal distribution is superior to the Gamma and Weibull distributions. We found that the Weib-BCART performs better in handling data with right-skewed and heavy-tailed characteristics. Actually, the conclusion of Omari *et al.* (2018) aligns with our initial analysis of claims severity data, as shown in Figure 6.16. We attribute the difference in results to two factors: First, they use MLE to obtain estimators once, while we use both MLE and posterior distributions to update parameter estimations at each node. Second, the Weibull distribution in their study has only one shape parameter for the entire dataset, while our approach incorporates different shape parameters in groups (terminal nodes) to adapt to data with different tail features (see Figure 6.19).

3. For aggregate claims data, we found that sequential models and joint models outperform the standard frequency-severity models, especially when incorporating $\hat{N}_i$ as a covariate in claims severity modelling within sequential

models and the ZICPG3 model within joint models. Modellers are advised to assess the conditional correlation between the number of claims and claims severity. If the dependence is strong, it is recommended to prioritize sequential models and ZICPG models (especially for data with many zeros); if the dependence is weak, consider frequency-severity models and joint models. Notably, joint models are preferred when there are many shared important covariates between claims frequency and severity (high ARI between their trees).

Chapter 7

Summary and Discussion

In this thesis, we explored the application of Bayesian CART models in insurance, employing various models to accommodate the unique characteristics of real insurance data. This chapter concludes the thesis by summarizing the key findings of the previous chapters. Additionally, we provide an outline for future research, along with potential industrial and practical applications.

7.1 Concluding Remarks

This work proposes the use of BCART models for insurance pricing. These tree-based models can automatically perform variable selection and detect non-linear effects and possible interactions among explanatory variables. The obtained optimal trees are relatively accurate, stable and straightforward to interpret by a visualization of the tree structure. These are desirable aspects of insurance pricing. We have introduced the framework of the BCART models and presented MCMC algorithms for general non-Gaussian distributed data where data augmentation may be needed in its implementation. We have included BCART models for Poisson, NB, ZIP, and ZINB distributions, which are the commonly used distributions for claims frequency. For the NB, ZIP, and ZINB models, we explored different ways to deal with exposure and concluded from the simulation examples and real data analyses that the non-standard ways of embedding exposure can provide us with better tree models, which is in line with the conclusions of Lee (2020, 2021). Moreover, for claims severity, we incorporated BCART models for Gamma, Log-Normal, and Weibull distributions, which have different abilities to handle data with varying tail characteristics. Concerning aggregate claims modelling, we proposed three types of models. First, the frequency-severity models consider two

trees for claims frequency and claims severity independently. Second, the sequential models treat the number of claims as a covariate in claims severity modelling, aiming to investigate the dependence between the number of claims and claims severity. Moreover, sequential models have demonstrated better performance in real data when compared to the frequency-severity models. Lastly, joint models for the bivariate response modelling (number of claims and aggregate claim amount) employ CPG and ZICPG distributions to directly model the claims cost using one joint tree. Joint models support the conclusion in Linero *et al.* (2020) by illuminating the benefits of information sharing.

Furthermore, we introduced a tree model selection approach based on DIC, which has been seen to be an effective approach using both simulation examples and real insurance data. In particular, we concluded from the real insurance data analyses that the ZIP-BCART with exposure embedded in the zero mass component is the best candidate for claims frequency modelling. It is worth remarking that another ZIP-BCART with exposure embedded in both the zero mass component and Poisson component does not yield significant improvements, but it indeed increases the model complexity. Besides, a general ZINB-BCART can be implemented and may further improve the accuracy, but this requires more latent variables to be introduced and will make the convergence of the MCMC algorithm harder/slower; see Murray (2021) for some insights. For claims severity, we found that the Weib-BCART performs the best among all candidates, as it can deal with cases where some groups have lighter tails, while others have heavier tails. In aggregate claims modelling, we discovered that among all joint models, the ZICPG3-BCART, which embeds exposure in both the zero mass component and Poisson component, delivers the most favourable results in real insurance data. When comparing joint models with frequency-severity models, we proposed using ARI to assess the similarity between two trees. This evaluation helps explain the potential benefits of employing one joint tree for information sharing.

7.2 Future Work

As we conclude our exploration of BCART models, we recognize the progress made in their application in the insurance industry, extending the usage to data with any general distribution. However, the landscape of Bayesian methodologies for tree-based models continues to develop, presenting opportunities for future research and refinement. Below we comment on potential further improvements of the BCART models.

1. In the MCMC algorithms we have only used four common move proposals, namely, Grow, Prune, Change, and Swap, which have made the algorithm quickly converge to a local optimal region. To make it better to explore the tree space, other proposals such as those in Wu *et al.* (2007) and Pratola (2016) can be suggested to improve the mixing of simulated trees. However, we suspect this will significantly increase the computational time, particularly, for high-dimensional large data sets and models requiring data augmentation. To mitigate this effect, we might need to use a non-uniform choice of splitting variables in the tree prior to achieve a better variable selection, e.g., the Dirichlet prior proposed in Linero (2018).

2. The proposed models have imposed several assumptions to simplify calculations. For example, we used conjugate priors for the terminal node distributions, and additional independence assumption as in (2.8) and (3.28). To further improve the analysis, it might be beneficial to incorporate different specifications of the prior for the same distribution scenario without using conjugate priors or independence, while this may require other techniques such as Laplace approximation (see Chipman *et al.* (2003)). We refer also to Chipman & McCulloch (2000) for an interesting incorporation of some hierarchical priors.

3. We have proposed to use a single (optimal) tree induced from the BCART models for insurance pricing. The main reason for this choice is, as we discussed in the introduction, for ease of interpretation. Since stakeholders and regulators may not be statisticians who are able to understand very complex statistical models, a single tree offers intuitive and visual results to them. Although we have proposed an approach to find a single optimal tree, some sub-optimal trees (in the convergence region of the MCMC) which possess similar/different tree structures, may also be as informative as the single optimal tree and should not be simply ignored. Further research can be done in this direction to make better use of the posterior trees by clustering or merging them; see, e.g., Chipman *et al.* (2001) and Banerjee *et al.* (2012).

4. We used sequential models to account for the dependence between the number of claims and claims severity. Alternatively, the copula approach can be adopted, which allows the modelling of the marginals and the dependence structure separately, providing an intuitive way to interpret the dependence. Recent advances in mixed copula models have simplified their application

in insurance; see, e.g., Czado *et al.* (2012) and Zilko & Kurowicka (2016). We propose further exploration into the use of mixed copula models in the Bayesian framework. We refer to Smith & Khaled (2012) and Smith (2011) for some discussions.

5. To further improve the accuracy of these Bayesian tree-based models we could explore BART models for insurance pricing. The BART models are tree ensembles; each tree in BART only accounts for a small part of the overall fit, potentially improving the performance, but model interpretability needs to be explored before it can be used for insurance pricing. To this end, we believe some insights from Henckaerts *et al.* (2021) would be helpful.

6. Because of the remarkable properties of tree-based models, an expanding number of researchers have been attempting to integrate them with other methodologies, as shown in Quan *et al.* (2020) and Diao & Weng (2019). Particularly, evolutionary trees are used to bin the effects of continuous risk factors in insurance claims data, thereby deriving GLMs that approximate the original GAMs in a flexible manner; see, e.g., Henckaerts *et al.* (2018). We expect the potential prospects for integrating Bayesian tree-based models with data-driven strategies for insurance tariff construction, like binning specifically; see, e.g., Lindholm *et al.* (2023) and Henckaerts *et al.* (2018). Binning, involving the discretization of continuous variables into categorical bins, simplifies the modelling process and potentially enhances interpretability. Therefore, this integration may provide benefits for improving interpretability and efficiently managing large data sets.

7. We proposed the use of ARI (see Subsection 5.3.3) to measure the similarity between different trees, aiming to assess the necessity of using one joint tree to share information. However, this method does not accurately provide a threshold for making decisive judgments. This limitation may arise because ARI solely considers the partitioned data itself, disregarding the underlying tree structure. To delve deeper into tree similarity exploration, alternative methods, as suggested in Bakirli & Birant (2017) and Nye *et al.* (2006), can be used. Additionally, there are other evaluation metrics capable of providing insights into the benefits of information sharing (such as the Log Pseudo Marginal Likelihood (LPML)) which are worth exploring; see, e.g., Linero *et al.* (2020).

8. Insurers must collect vast amounts of data to optimize performance and mitigate risks. Data plays a crucial role, and dealing with missing data remains an unavoidable challenge. Consequently, the capability of tree-based models to handle missing data becomes a critical criterion when considering their application in the insurance industry. We refer to Kapelner & Bleich (2015) for a more insightful discussion on this topic.

Appendix A

Metropolis-Hastings for Sampling New Trees

This appendix provides detailed calculations for Algorithm 2.2 described in Subsection 2.2.2. The goal is to explicitly calculate the acceptance ratio $\alpha(\mathcal{T}^{(m)}, \mathcal{T}^*)$ for all possible tree moves. For each tree move, the calculations are organised into three separate ratios: transition ratio $q(\mathcal{T}^*, \mathcal{T}^{(m)})/q(\mathcal{T}^{(m)}, \mathcal{T}^*)$, integrated likelihood ratio $p(\boldsymbol{y} \mid \boldsymbol{X}, \mathcal{T}^*)/p(\boldsymbol{y} \mid \boldsymbol{X}, \mathcal{T}^{(m)})$, and tree prior ratio $p(\mathcal{T}^*)/p(\mathcal{T}^{(m)})$. There are similar calculations discussed in Kapelner & Bleich (2013). We present our calculations first and discuss the similarities and differences in the final remark.

A.1 Grow Move

In grow moves, an arbitrarily terminal node needs to be selected and split into two new child nodes, along with randomly assigned decision rules for the split. Therefore, when a grow move occurs to generate a new tree $\mathcal{T}^*$ from $\mathcal{T}^{(m)}$, we only need to consider the changes in the selected terminal node and the two resulting child nodes, as other nodes remain unchanged.

- Transition ratio.

$$
\begin{aligned}
&q(\mathcal{T}^{(m)}, \mathcal{T}^*) \\
&= \mathbb{P}(\text{GROW})\mathbb{P}(\text{selecting the } t\text{-th terminal node to grow}) \\
&\quad \times \mathbb{P}(\text{selecting one covariate from the available covariates to split on}) \\
&\quad \times \mathbb{P}(\text{selecting one value } c \text{ /category } C \text{ to split on based on the chosen covariate}) \\
&= \mathbb{P}(\text{GROW}) \frac{1}{b} \frac{1}{p_{(m)}(t)} \frac{1}{n_{p(m)}(t)},
\end{aligned}
$$

where b is the number of terminal nodes; $p_{(m)}(t)$ denotes the number of predictors left available to split on, which can be less than p (the total number of explanatory variables) if certain predictors do not have two or more unique values/categories once the data reaches the t-th node in $\mathcal{T}^{(m)}$; $n_{p(m)}(t)$ represents the number of unique values/categories left for the chosen covariate. Furthermore, $\mathbb{P}(\text{GROW})$ is a probability that can be specified in advance. In our implementation, we select equal probability (i.e., 1/4 each) for four tree moves, as suggested in Chipman *et al.* (1998), but it can be varied for different purposes. For example, a higher probability can be specified for grow moves if a larger tree is desired. Additionally, it should be noted that $\mathbb{P}(\text{GROW})$ is set to zero and the step will be automatically rejected if there are no variables with two or more unique values/categories.

Similarly, the transition from the new tree $\mathcal{T}^*$ to the original tree $\mathcal{T}^{(m)}$ involves pruning the grown node:

$$\begin{aligned} q(\mathcal{T}^*, \mathcal{T}^{(m)}) &= \mathbb{P}(\text{PRUNE})\mathbb{P}(\text{selecting the } t^*\text{-th node to prune}) \\ &= \mathbb{P}(\text{PRUNE})\frac{1}{b^*}, \end{aligned}$$

where b^* denotes the number of nodes with two terminal children in the new tree $\mathcal{T}^*$, and $\mathbb{P}(\text{PRUNE})$ is also a known probability as $\mathbb{P}(\text{GROW})$. Thus, the full transition ratio is:

$$\frac{q(\mathcal{T}^*, \mathcal{T}^{(m)})}{q(\mathcal{T}^{(m)}, \mathcal{T}^*)} = \frac{\mathbb{P}(\text{PRUNE}) b p_{(m)}(t) n_{p(m)}(t)}{\mathbb{P}(\text{GROW}) b^*}.$$

- Integrated likelihood ratio. Since the likelihoods are entirely determined by the terminal nodes, the new tree $\mathcal{T}^*$ differs from the original tree $\mathcal{T}^{(m)}$ solely in the t-th terminal node which is chosen to grow and becomes two children nodes after the grow move, denoted by t_L^* and t_R^*. Therefore, the integrated likelihood ratio can be expressed as follows:

$$\frac{p\left(\boldsymbol{y} \mid \boldsymbol{X}, \mathcal{T}^*\right)}{p\left(\boldsymbol{y} \mid \boldsymbol{X}, \mathcal{T}^{(m)}\right)} = \frac{p(\boldsymbol{y}_{t_L^*} \mid \boldsymbol{X}_{t_L^*}) p(\boldsymbol{y}_{t_R^*} \mid \boldsymbol{X}_{t_R^*})}{p\left(\boldsymbol{y}_t \mid \boldsymbol{X}_t\right)}.$$

- Tree prior ratio. For the entire tree,

$$p\left(\mathcal{T}\right) = \prod_{t \in L_{\text{terminals}}} \left(1 - p(d_t)\right) \prod_{t \in L_{\text{internals}}} \left(p(d_t) \frac{1}{p_{(m)}(t)} \frac{1}{n_{p(m)}(t)} \right),$$

where $L_{\text{terminals}}$ denotes the set of terminal nodes and $L_{\text{internals}}$ denotes the set of internal nodes; d_t represents the depth of node t and $p(d_t) =$

$\gamma\,(1+d_t)^{-\rho}$ (see Subsection 2.2.1). Thus, the tree prior ratio can be expressed as:

$$\begin{aligned}\frac{p\left(\mathcal{T}^{*}\right)}{p\left(\mathcal{T}^{(m)}\right)} &= \frac{(1-p(d_{t_L^*}))(1-p(d_{t_R^*}))p(d_{t^*})\frac{1}{p_{(m)}(t^*)}\frac{1}{n_{p(m)}(t^*)}}{\left(1-p(d_t)\right)} \\ &= \frac{\left(1-\frac{\gamma}{(1+d_{t_L^*})^{\rho}}\right)\left(1-\frac{\gamma}{(1+d_{t_R^*})^{\rho}}\right)\frac{\gamma}{(1+d_{t^*})^{\rho}}\frac{1}{p_{(m)}(t^*)}\frac{1}{n_{p(m)}(t^*)}}{1-\frac{\gamma}{(1+d_t)^{\rho}}} \\ &= \frac{\gamma\left(1-\frac{\gamma}{(2+d_t)^{\rho}}\right)^2}{\left((1+d_t)^{\rho}-\gamma\right)p_{(m)}(t^*)n_{p(m)}(t^*)}.\end{aligned}$$

Note $d_{t_L} = d_{t_R} = d_t + 1$ and $d_t = d_{t^*}$.

A.2 Prune Move

In prune moves, a terminal node and its sibling node are randomly selected to be pruned into the direct parent node, which then becomes a new terminal node. In essence, it is the reverse process of grow moves. Consequently, each ratio will be approximately the inverse of the ratios found in grow moves.

- Transition ratio.

$$\begin{aligned}q(\mathcal{T}^{(m)},\mathcal{T}^{*}) =& \mathbb{P}(\text{PRUNE})\mathbb{P}(\text{selecting the } t\text{-th node to prune}) \\ =& \mathbb{P}(\text{PRUNE})\frac{1}{b_2^*},\end{aligned}$$

  where b_2^* denotes the number of nodes with two terminal children in the original tree $\mathcal{T}^{(m)}$, which is different from b^* in the grow move. To transition in the opposite direction, this formula closely resembles the one in the grow move, with the exception that the new tree has one less terminal node due to pruning the original tree, resulting in a $1/(b-1)$ term,

$$q(\mathcal{T}^{*},\mathcal{T}^{(m)}) = \mathbb{P}(\text{GROW})\frac{1}{b-1}\frac{1}{p_{(m)}(t^*)}\frac{1}{n_{p(m)}(t^*)}.$$

  Thus, the full transition ratio is:

$$\frac{q(\mathcal{T}^{*},\mathcal{T}^{(m)})}{q(\mathcal{T}^{(m)},\mathcal{T}^{*})} = \frac{\mathbb{P}(\text{GROW})b_2^*}{\mathbb{P}(\text{PRUNE})(b-1)p_{(m)}(t^*)n_{p(m)}(t^*)}.$$

- Integrated likelihood ratio, which is simply the inverse of the integrated likelihood ratio in the grow move:

$$\frac{p\left(\boldsymbol{y}\mid\boldsymbol{X},\mathcal{T}^{*}\right)}{p\left(\boldsymbol{y}\mid\boldsymbol{X},\mathcal{T}^{(m)}\right)} = \frac{p\left(\boldsymbol{y}_{t^*}\mid\boldsymbol{X}_{t^*}\right)}{p\left(\boldsymbol{y}_{t_L}\mid\boldsymbol{X}_{t_L}\right)p(\boldsymbol{y}_{t_R}\mid\boldsymbol{X}_{t_R})}.$$

- Tree prior ratio, which is also simply the inverse of the tree prior ratio in the grow move:

$$\frac{p\left(\mathcal{T}^*\right)}{p\left(\mathcal{T}^{(m)}\right)} = \frac{\left((1+d_t)^\rho - \gamma\right) p_{(m)}(t^*) n_{p(m)}(t^*)}{\gamma\left(1 - \frac{\gamma}{(2+d_t)^\rho}\right)^2}.$$

A.3 Change Move

The change moves can be implemented in any internal node, as described in Subsection 2.2.2. However, for the sake of simplicity, we only show the computation details of a change move restricted to a single internal node with two terminal children below.

- Transition ratio.

$$\begin{aligned}
&q(\mathcal{T}^{(m)}, \mathcal{T}^*) \\
&= \mathbb{P}(\text{CHANGE})\mathbb{P}(\text{selecting the } t\text{-th node to change}) \\
&\quad \times \mathbb{P}(\text{selecting one (new) covariate from the available covariates to split}) \\
&\quad \times \mathbb{P}(\text{selecting one value } c \text{ /category } C \text{ to split based on the chosen} \\
&\qquad \text{covariate}),
\end{aligned}$$

  where the node t is the parent of two terminal nodes under the given restriction. When calculating the transition ratio, the first two terms in the numerator and denominator are the same, while the last two terms differ due to varying numbers of available covariates in the different chosen nodes and different numbers of unique values/categories available for different splitting covariates. Therefore,

$$\frac{q(\mathcal{T}^*, \mathcal{T}^{(m)})}{q(\mathcal{T}^{(m)}, \mathcal{T}^*)} = \frac{p_{(m)}(t^*) n_{p(m)}(t^*)}{p_{(m)}(t) n_{p(m)}(t)}.$$

  Specifically, for Change1 (see Subsection 2.2.2), the above equation can be simplified to 1 since the splitting covariates are not changed, and the number of unique values/categories available for the same splitting covariate remains the same.

- Integrated likelihood ratio. The new tree differs from the original tree solely in the two child nodes of the selected change node. Thus,

$$\frac{p\left(\boldsymbol{y} \mid \boldsymbol{X}, \mathcal{T}^*\right)}{p\left(\boldsymbol{y} \mid \boldsymbol{X}, \mathcal{T}^{(m)}\right)} = \frac{p(\boldsymbol{y}_{t_L^*} \mid \boldsymbol{X}_{t_L^*}) p(\boldsymbol{y}_{t_R^*} \mid \boldsymbol{X}_{t_R^*})}{p(\boldsymbol{y}_{t_L} \mid \boldsymbol{X}_{t_L}) p(\boldsymbol{y}_{t_R} \mid \boldsymbol{X}_{t_R})}.$$

- Tree prior ratio. Since the new tree essentially has the same structure as the original tree, only the two children of the chosen change node need to be taken into account,

$$\begin{aligned}\frac{p\left(\mathcal{T}^{*}\right)}{p\left(\mathcal{T}^{(m)}\right)} &= \frac{(1-p(d_{t_L^*}))(1-p(d_{t_R^*}))p(d_{t^*})\frac{1}{p_{(m)}(t^*)}\frac{1}{n_{p(m)}(t^*)}}{\left(1-p(d_{t_L})\right)\left(1-p(d_{t_R})\right)p(d_t)\frac{1}{p_{(m)}(t)}\frac{1}{n_{p(m)}(t)}} \\ &= \frac{p_{(m)}(t)n_{p(m)}(t)}{p_{(m)}(t^*)n_{p(m)}(t^*)}.\end{aligned}$$

  Apparently, this is the inverse of the transition ratio.

Therefore, for the change move,

$$\alpha(\mathcal{T}^{(m)},\mathcal{T}^{*}) = \min\left\{\frac{p\left(\boldsymbol{y}\mid\boldsymbol{X},\mathcal{T}^{*}\right)}{p\left(\boldsymbol{y}\mid\boldsymbol{X},\mathcal{T}^{(m)}\right)},1\right\}.$$

A.4 Swap Move

Similar to the change move, swap moves can also be implemented in any internal node, and we only show the computation details of a swap move restricted to a child node within the parent-child swap pair that has two terminal children.

- Transition ratio. Since the splitting rules are swapped rather than changed, we do not need to consider choosing new splitting rules,

$$q(\mathcal{T}^{(m)},\mathcal{T}^{*}) = \mathbb{P}(\text{SWAP})\mathbb{P}(\text{selecting the } t\text{-th node to swap})$$

  where the node t is the child node in the chosen parent-child swap pair under the given restriction. When calculating the transition ratio, the two terms are the same in the numerator and denominator. Therefore,

$$\frac{q(\mathcal{T}^{*},\mathcal{T}^{(m)})}{q(\mathcal{T}^{(m)},\mathcal{T}^{*})} = 1.$$

- Integrated likelihood ratio. Under the given restriction, the new tree differs from the original tree in the sibling of the child node in the chosen parent-child swap pair, denoted as t_1, and the two terminal children of the child node denoted as t_2 and t_3 respectively. Thus,

$$\frac{p\left(\boldsymbol{y}\mid\boldsymbol{X},\mathcal{T}^{*}\right)}{p\left(\boldsymbol{y}\mid\boldsymbol{X},\mathcal{T}^{(m)}\right)} = \frac{p(\boldsymbol{y}_{t_1^*}\mid\boldsymbol{X}_{t_1^*})p(\boldsymbol{y}_{t_2^*}\mid\boldsymbol{X}_{t_2^*})p(\boldsymbol{y}_{t_3^*}\mid\boldsymbol{X}_{t_3^*})}{p(\boldsymbol{y}_{t_1}\mid\boldsymbol{X}_{t_1})p(\boldsymbol{y}_{t_2}\mid\boldsymbol{X}_{t_2})p(\boldsymbol{y}_{t_3}\mid\boldsymbol{X}_{t_3})}.$$

- Tree prior ratio. The new tree has the same structure as the original tree, and the splitting rules remain unchanged. Thus,

$$\frac{p(\mathcal{T}^*)}{p(\mathcal{T}^{(m)})} = 1.$$

Therefore, for the swap move, the same conclusion can be obtained as the change move,

$$\alpha(\mathcal{T}^{(m)}, \mathcal{T}^*) = \min\left\{\frac{p\left(\boldsymbol{y} \mid \boldsymbol{X}, \mathcal{T}^*\right)}{p\left(\boldsymbol{y} \mid \boldsymbol{X}, \mathcal{T}^{(m)}\right)}, 1\right\}.$$

Since only the integrated likelihood ratio needs to be computed for change and swap moves in the acceptance ratio, this is a particularly appealing feature for the implementation.

Remark A.1 *(a) The actual implementation uses the above expressions in logarithmic form for numerical accuracy.*

(b) It should be noted that in the above expressions for a node, as long as there is $*$, it indicates the node belongs to $\mathcal{T}^*$.

(c) Similar detailed calculations can be found in Kapelner & Bleich (2013). However, there are some differences we need to emphasize.

- *We do not restrict the implementation to Gaussian data, allowing the integrated likelihood ratio to be expressed in a broader manner suitable for data with any general distribution.*

- *Unlike Kapelner & Bleich (2013), we consider swap moves in our implementation.*

- *In our approach, change and swap moves are considered for any internal node in the tree, while Kapelner & Bleich (2013) limit their implementation to those with two terminal nodes for change moves. Additionally, we include two types of change moves (see Subsection 2.2.2).*

Appendix B

Reversible Jump MCMC for Sampling New Trees

This appendix illustrates the computation of the acceptance ratio in RJMCMC, demonstrating its equivalence to the MH algorithm in Appendix A. To this end, we provide a basic introduction to RJMCMC first, following the mathematical description of the method as outlined in Voss (2013). Additionally, it is important to note that all letters used in this appendix have specific meanings exclusive to this appendix.

B.1 A Basic Introduction to RJMCMC

More mathematical formalism is needed to state the RJMCMC algorithm than was required for the MH algorithm because of the complex structure of the state space employed in RJMCMC. The general RJMCMC algorithm is provided in this section along with the necessary notation. We begin the exposition by describing the state space mathematically. Let O be a finite or countable set and let $d_h \in \mathbb{N}_0$ for all $h \in O$ be given. Define the state space

$$S = \bigcup_{h \in O} S_h,$$

where $S_h = \{h\} \times \mathbb{R}^{d_h}$ for all $h \in O$. For any element w in the space S, we have the form $w = (h, x)$, where $h \in O$ and $x \in \mathbb{R}^{d_h}$, and the first component, h, indicates which of the spaces $\mathbb{R}^{d_h}$ a point is in while the second component, x, identifies the position in this space. The spaces S_h are disjoint and each $w \in S$ is contained in exactly one of the subspaces S_h because the index h is included as the first component of all elements in S_h.

Next, on the space S, where the Markov chain constructed by the RJMCMC algorithm will move in, the target distribution π is specified. If $W \sim \pi$, then W can be written as $W = (H, X)$, and the joint distributions of $H \in O$ and $X \in \mathbb{R}^{d_H}$ must be specified. A density π that is split between the different subspaces S_h can be used to characterise such a distribution; that is, a function $\pi(\cdot, \cdot)$ such that $\pi(h, \cdot) : \mathbb{R}^{d_h} \to [0, \infty)$ for every $h \in O$ and

$$\sum_{h \in O} \int_{\mathbb{R}^{d_h}} \pi(h, x) dx = 1.$$

Then we have

$$\mathrm{P}(H = h) = \int_{\mathbb{R}^{d_h}} \pi(h, x) dx,$$

and

$$\mathrm{P}(H = h, X \in A) = \int_A \pi(h, x) dx,$$

for all $A \subseteq \mathbb{R}^{d_h}$ and all $h \in O$.

Now, the aim is to construct a Markov chain with stationary distribution π in Bayesian tree models. A pair (H_j, X_j) describes the state at time j for the Markov chain moving in S. For each $(h, x) \in S$, we need to specify the distribution of (H_j, X_j) when $(H_{j-1}, X_{j-1}) = (h, x)$ in order to characterise the transition probabilities of such a Markov chain. The RJMCMC algorithm benefits from first determining the value of H_j before determining the value of X_j from the conditional distribution in a second step that is conditioned on the value of H_j. Thus, the transitions of the Markov chain from $(h, x) \in S$ to $(g, y) \in S$ will be described by probability weights $b(h, x; g)$ with

$$\sum_{g \in O} b(h, x; g) = 1$$

and probability densities $p(h, x; g, \cdot)$ on $\mathbb{R}^{d_g}$ for all $(h, x) \in S$. If $(H_j, X_j)_{j \in \mathbb{N}}$ is described by b and p, then

$$\mathrm{P}(H_j = g, X_j \in A \mid H_{j-1} = h, X_{j-1} = x) = b(h, x; g) \int_A p(h, x; g, y) dy$$

for all $h, g \in O, x \in \mathbb{R}^{d_h}$ and $A \subseteq \mathbb{R}^{d_g}$.

The idea of splitting the transition mechanism into different move types, i.e., tree moves in Bayesian tree models, is the next component incorporated into the RJMCMC algorithm. The set of all potential move types is denoted by M. The

probability of selecting move $m \in M$ for a given state $(h, x) \in S_h$ is represented by $\gamma_m(h, x)$. The probabilities $\gamma_m(h, x)$ satisfy

$$\sum_{m \in M} \gamma_m(h, x) = 1$$

for all $(h, x) \in S$. In the presence of different move types, the transition probabilities given by b and p rely on the move type m, that is rather than b and p, probability weights b_m and probability densities p_m are considered for all $m \in M$.

There are two scenarios to take into account while calculating the corresponding acceptance ratio for the MH algorithm.

1. The proposal (g, y) lies in the same space as the previous state (h, x) does, i.e., $g = h$. This is exactly the case with regard to change and swap moves in tree models. The distribution of the new location X_j, which occurs with probability $b_m(h, x; h)$, is provided by a density $p_m(h, x; \cdot) : \mathbb{R}^{d_h} \to [0, \infty)$. The acceptance ratio for this case can be stated as follows:

$$\alpha_m(h, x; y) = \min\left(\frac{\pi(h, y)\gamma_m(h, y)b_m(h, y; h)p_m(h, y; x)}{\pi(h, x)\gamma_m(h, x)b_m(h, x; h)p_m(h, x; y)}, 1\right)$$

   for all $x, y \in \mathbb{R}^{d_h}$ and all $h \in O$.

2. The proposal (g, y) falls into a space $S_g \neq S_h$. This is exactly the case with regard to grow and prune moves in tree models. In this more complex case, it is necessary to calculate the probability $b_m(h, x; g)$ in the space $S_g = \{g\} \times \mathbb{R}^{d_g}$. When transitioning from S_h to S_g, both $\mathbb{R}^{d_h}$ and $\mathbb{R}^{d_g}$ are temporarily extended to spaces of matching dimension; that is, we consider $\mathbb{R}^{d_h} \times \mathbb{R}^{n_h}$ and $\mathbb{R}^{d_g}$ in place of $\mathbb{R}^{d_h}$ and $\mathbb{R}^{d_g}$, where

$$d_h + n_h = d_g + n_g.$$

   In order to construct the proposal R, we follow these steps: (1) generate an auxiliary random variable $U \in \mathbb{R}^{n_h}$ using a probability density $\psi_m(h, x, \cdot; g) : \mathbb{R}^{n_h} \to [0, \infty)$. (2) Use a map $\varphi_m^{h \to g} : \mathbb{R}^{d_h} \times \mathbb{R}^{n_h} \to \mathbb{R}^{d_g} \times \mathbb{R}^{n_g}$ to obtain $(R, V) = \varphi_m^{h \to g}(x, U)$. The densities ψ_m and the maps $\varphi_m^{h \to g}$ can be chosen as part of designing the algorithm, subject to certain conditions; see more detailed discussion in Voss (2013).

   The acceptance ratio for this case can then be expressed as:

$$\begin{aligned}&\alpha_m(h, x, u; g, y, v)\\ &= \min\left(\frac{\pi(g, y)\gamma_m(g, y)b_m\left(g, y; h\right)\psi_m(g, y, v; h)}{\pi(h, x)\gamma_m(h, x)b_m\left(h, x; g\right)\psi_m(h, x, u; g)} \mid \det D\varphi_m^{h \to g}(x, u) \mid, 1\right),\end{aligned}$$

where $D\varphi_m^{h\to g}(x,u)$ is the Jacobian matrix of $\varphi_m^{h\to g}$ (see Voss (2013)).

After providing a basic introduction to RJMCMC, the objective aligns with that of the MH algorithm (see Appendix A), i.e., to explicitly calculate the acceptance ratio for all possible tree moves. Initially, we specify the notation for tree models: using h to denote the node state of the tree; μ for the splitting variable; ϕ for the split value/category based on the chosen covariate, and θ for the node parameter which can be a vector. For simplicity, in the following calculation, we only consider θ within one dimension as a replacement for multidimensional parameters $\boldsymbol{\theta}$. Considering move types $m \in M$, we have four tree moves, where grow and prune moves are inverses, ensuring identical calculations. The stationary distribution can be expressed as $\pi(h,z)$ with $z = c(\mu,\phi,\theta)$.

B.2 Grow/Prune Move

For grow and prune moves, the dimension has changed, and the acceptance ratio is referred to as the second case,

$$\begin{aligned}&\alpha_m(h,z,u;g,z^*,v)\\ =&\min\left(\frac{\pi(g,z^*)\gamma_m(g,z^*)b_m(g,z^*;h)\psi_m(g,z^*,v;h)}{\pi(h,z)\gamma_m(h,z)b_m(h,z;g)\psi_m(h,z,u;g)}\left|\det D\varphi_m^{h\to g}(z,u)\right|,1\right),\end{aligned}$$

where u and v are random variables. Next, we will calculate each probability shown in the equation.

- The stationary distribution can be expressed as the production of likelihood and each parameter's priors,

$$\begin{aligned}\pi(g,z^*) &= p(g,\mu^*,\phi^*,\theta^* \mid \boldsymbol{y},\boldsymbol{X})\\ &= p(\boldsymbol{y} \mid g,\mu^*,\phi^*,\theta^*,\boldsymbol{X})p(\theta^* \mid g,\mu^*,\phi^*,\boldsymbol{X})p(\phi^* \mid g,\mu^*)p(\mu^* \mid g)p(g).\end{aligned}$$

  A similar equation can be obtained for $\pi(h,z)$. Using the same notation as in the MH algorithm (see Appendix A), we can rewrite the probabilities accordingly and calculate the ratio,

$$\begin{aligned}\frac{\pi(g,z^*)}{\pi(h,z)} &= \frac{p(g,\mu^*,\phi^*,\theta^* \mid \boldsymbol{y},\boldsymbol{X})}{p(h,\mu,\phi,\theta \mid \boldsymbol{y},\boldsymbol{X})}\\ &= \frac{p(\boldsymbol{y}_{t_L^*} \mid \boldsymbol{X}_{t_L^*})p(\boldsymbol{y}_{t_R^*} \mid \boldsymbol{X}_{t_R^*})}{p\left(\boldsymbol{y}_t \mid \boldsymbol{X}_t\right)}\frac{p(\theta_{t_L^*})p(\theta_{t_R^*})}{p(\theta_t)}\frac{1}{n_{p(m)}(t)}\frac{1}{p_{(m)}(t)}\\ &\quad\times\frac{(1-p(d_{t_L^*}))(1-p(d_{t_R^*}))p(d_{t^*})}{\left(1-p(d_t)\right)}.\end{aligned}$$

- γ_m is the probability of choosing tree moves $m \in M$, which is given in advance, so
$$\gamma_m(g, z^*) = \gamma_m(h, z).$$

- $b_m(h, z; g)$ is the probability for move type m to move into space S_g when the current space is S_h,
$$b_m(g, z^*; h) = \mathbb{P}(\text{PRUNE}) \times \frac{1}{b^*},$$
$$b_m(h, z; g) = \mathbb{P}(\text{GROW}) \times \frac{1}{b}.$$

- $\psi_m(g, z^*, v; h)$ is the density of the auxiliary random variable $U \in \mathbb{R}^{n_h}$, when moving from space S_h into space S_g using move type m,
$$\psi_m(h, z, u; g) \overset{g=h+1}{=} \frac{1}{p_{(m)}(t)} \frac{1}{n_{p(m)}(t)} p(\theta_{t_L^*}) p(\theta_{t_R^*}).$$
Given $U = (\mu, \phi, \theta_{t_L^*}, \theta_{t_R^*})$, $\psi_m(g, z^*, u^*; h) = p(\theta_t)$; the map can be obtained,
$$\begin{aligned}(z^*, v) &= (z/\theta_t, \mu, \phi, \theta_{t_L^*}, \theta_{t_R^*}, v) \\ &= \varphi_m^{h \to g}(z, u) = \left(z/\theta_t, \theta_t, u_1, u_2, u_3, u_4\right).\end{aligned}$$
By using the permutation, we can obtain $\left|\det D\varphi_m^{h \to g}(z, u)\right| = 1$.

Finally, by substituting all the probabilities obtained above into the acceptance ratio, we can derive:

$$\alpha_m = \frac{p(\boldsymbol{y}_{t_L^*} \mid \boldsymbol{X}_{t_L^*}) p(\boldsymbol{y}_{t_R^*} \mid \boldsymbol{X}_{t_R^*})}{p\left(\boldsymbol{y}_t \mid \boldsymbol{X}_t\right)} \frac{(1 - p(d_{t_L^*}))(1 - p(d_{t_R^*}))p(d_{t^*})}{\left(1 - p(d_t)\right)} \frac{\mathbb{P}(\text{PRUNE})b}{\mathbb{P}(\text{GROW})b^*},$$

which is the same as the MH algorithm for grow and prune moves in Appendix A.

B.3 Change Move

For change and swap moves, the dimension has not changed, and the acceptance ratio is referred to as the first case,

$$\alpha_m(h, z; z^*) = \min\left(\frac{\pi(h, z^*)\gamma_m(h, z^*)b_m(h, z^*; h)p_m(h, z^*; z)}{\pi(h, z)\gamma_m(h, z)b_m(h, z; h)p_m(h, z; z^*)}, 1\right).$$

Change and swap moves can be implemented in any internal node in the tree. For the sake of simplicity, we only provide the computation details of change and swap moves under the same restriction as in the MH algorithm (see Appendix A).

- For the stationary distribution,

$$\begin{aligned}\frac{\pi(h,z^*)}{\pi(h,z)} &= \frac{p(h,\mu^*,\phi^*,\theta^* \mid \boldsymbol{y},\boldsymbol{X})}{p(h,\mu,\phi,\theta \mid \boldsymbol{y},\boldsymbol{X})} \\ &= \frac{p(\boldsymbol{y}_{t_L^*} \mid \boldsymbol{X}_{t_L^*})p(\boldsymbol{y}_{t_R^*} \mid \boldsymbol{X}_{t_R^*})}{p(\boldsymbol{y}_{t_L} \mid \boldsymbol{X}_{t_L})p(\boldsymbol{y}_{t_R} \mid \boldsymbol{X}_{t_R})} \frac{p(\theta_{t_L^*})p(\theta_{t_R^*})}{p(\theta_{t_L})p(\theta_{t_R})} \frac{n_{p(m)}(t^*)}{n_{p(m)}(t)} \frac{p_{(m)}(t^*)}{p_{(m)}(t)}.\end{aligned}$$

- For γ_m, it is obvious to obtain

$$\gamma_m(h,z^*) = \gamma_m(h,z).$$

- $b_m(h,z;h)$ is the probability for change moves to transition into space S_h when the current space is S_h as well, which is considered alongside p_m below.

- $p_m(h,z;\cdot)$ is the density of the proposal when moving inside space S_h with change moves. Combining probability weights b_m and probability densities p_m gives the transition probabilities,

$$\begin{aligned}b_m(h,z^*;h)p_m(h,z^*;z) &= \mathbb{P}(\text{CHANGE})\mathbb{P}(\text{selecting the } t\text{-th node to change}) \\ &\quad \times \frac{1}{p_{(m)}(t^*)} \frac{1}{n_{p(m)}(t^*)} p(\theta_{t_L})p(\theta_{t_R}),\end{aligned}$$

$$\begin{aligned}b_m(h,z;h)p_m(h,z;z^*) &= \mathbb{P}(\text{CHANGE})\mathbb{P}(\text{selecting the } t\text{-th node to change}) \\ &\quad \times \frac{1}{p_{(m)}(t)} \frac{1}{n_{p(m)}(t)} p(\theta_{t_L^*})p(\theta_{t_R^*}),\end{aligned}$$

  where the node t is the parent of two terminal nodes under the given restriction.

Finally, by substituting all the probabilities obtained above into the acceptance ratio, we can derive:

$$\alpha_m = \frac{p(\boldsymbol{y}_{t_L^*} \mid \boldsymbol{X}_{t_L^*})p(\boldsymbol{y}_{t_R^*} \mid \boldsymbol{X}_{t_R^*})}{p(\boldsymbol{y}_{t_L} \mid \boldsymbol{X}_{t_L})p(\boldsymbol{y}_{t_R} \mid \boldsymbol{X}_{t_R})},$$

which is the integrated likelihood ratio itself, and this result is the same as in the MH algorithm for change moves in Appendix A.

B.4 Swap Move

Recall that under the given restriction as in the MH algorithm (see Appendix A), the new tree differs from the original tree in the sibling of the child node within the parent-child swap pair, denoted as t_1, and the two terminal children of the child node, denoted as t_2 and t_3 respectively.

- For the stationary distribution,

$$\begin{aligned}\frac{\pi(h,z^*)}{\pi(h,z)} &= \frac{p(h,\mu^*,\phi^*,\theta^* \mid \boldsymbol{y},\boldsymbol{X})}{p(h,\mu,\phi,\theta \mid \boldsymbol{y},\boldsymbol{X})} \\ &= \frac{p(\boldsymbol{y}_{t_1^*} \mid \boldsymbol{X}_{t_1^*})p(\boldsymbol{y}_{t_2^*} \mid \boldsymbol{X}_{t_2^*})p(\boldsymbol{y}_{t_3^*} \mid \boldsymbol{X}_{t_3^*})}{p(\boldsymbol{y}_{t_1} \mid \boldsymbol{X}_{t_1})p(\boldsymbol{y}_{t_2} \mid \boldsymbol{X}_{t_2})p(\boldsymbol{y}_{t_3} \mid \boldsymbol{X}_{t_3})} \frac{p(\theta_{t_1^*})p(\theta_{t_2^*})p(\theta_{t_3^*})}{p(\theta_{t_1})p(\theta_{t_2})p(\theta_{t_3})}.\end{aligned}$$

- For γ_m, it is obvious to obtain

$$\gamma_m(h,z^*) = \gamma_m(h,z).$$

- Using the same method to calculate probability weights b_m and probability densities p_m as in change moves, we can obtain,

$$\begin{aligned}b_m(h,z^*;h)p_m(h,z^*;z) = \mathbb{P}(\text{SWAP})\mathbb{P}(&\text{selecting the } t\text{-th node to swap}) \\ &\times p(\theta_{t_1})p(\theta_{t_2})p(\theta_{t_3}),\end{aligned}$$

$$\begin{aligned}b_m(h,z;h)p_m(h,z;z^*) = \mathbb{P}(\text{SWAP})\mathbb{P}(&\text{selecting the } t\text{-th node to swap}) \\ &\times p(\theta_{t_1^*})p(\theta_{t_2^*})p(\theta_{t_3^*}).\end{aligned}$$

Finally, by substituting all the probabilities obtained above into the acceptance ratio, we can derive:

$$\alpha_m = \frac{p(\boldsymbol{y}_{t_1^*} \mid \boldsymbol{X}_{t_1^*})p(\boldsymbol{y}_{t_2^*} \mid \boldsymbol{X}_{t_2^*})p(\boldsymbol{y}_{t_3^*} \mid \boldsymbol{X}_{t_3^*})}{p(\boldsymbol{y}_{t_1} \mid \boldsymbol{X}_{t_1})p(\boldsymbol{y}_{t_2} \mid \boldsymbol{X}_{t_2})p(\boldsymbol{y}_{t_3} \mid \boldsymbol{X}_{t_3})},$$

which is the integrated likelihood ratio itself, and this result is the same as in the MH algorithm for swap moves in Appendix A.

Appendix C

Rand Index and Adjusted Rand Index

In the context of cluster analysis and partition comparison, the Rand Index (RI) and adjusted Rand Index (ARI) serve as essential metrics to evaluate the similarity between different clusterings; see, e.g., Rand (1971), Hubert & Arabie (1985) and Gates & Ahn (2017). This appendix provides a brief introduction to RI and ARI, aiming to help readers better understand their application in tree-based models to assess the similarity between different trees. Besides, it is crucial to remember that each letter used in this appendix holds a unique meaning exclusive to this appendix.

C.1 Rand Index

The RI evaluates the similarity between two clusterings by calculating the percentage of accurately classified pairs of data points, distinguishing between those belonging to the same cluster or distinct clusters. Computed by dividing the total number of pairs of data points by the sum of true positives and true negatives, the value of RI ranges from 0 to 1. A value of 0 indicates no agreement (the two data clusterings disagree on all pairs of points), while a value of 1 signifies perfect agreement (the data clusterings are identical). Specifically, given a set of n elements $O = \{o_1, \ldots, o_n\}$ and two partitions of O, denoted as $X = \{X_1, \ldots, X_r\}$ (a partition of O into r subsets) and $Y = \{Y_1, \ldots, Y_s\}$ (a partition of O into s subsets), we define the following terms to compare these two clusterings.

- a, the number of pairs of elements in O that are in the same subset in both X and Y, representing the number of times that a pair of elements belongs

to the same cluster across two different clustering outcomes.

- b, the number of pairs of elements in O found in different subsets in both X and Y.
- c, the number of pairs of elements in O that are in the same subset in X but in different subsets in Y.
- d, the number of pairs of elements in O that are in different subsets in X but in the same subset in Y.

RI can then be computed as follows:

$$\text{RI} = \frac{a+b}{a+b+c+d} = \frac{a+b}{\binom{n}{2}} = \frac{a+b}{n(n-1)/2}.$$

Intuitively, $a+b$ can be seen as the number of agreements between X and Y, and $c+d$ as the number of disagreements between X and Y. Since the denominator represents the total number of pairs (the number of unordered pairs in a set of n elements), the RI denotes the frequency of agreements over the total pairs or the probability that X and Y would agree on a randomly selected pair. To facilitate a better understanding of the RI, we provide a simple example below.

C.1.1 A Simple Example

Assuming we have a set of six elements: $\{A, B, C, D, E, F\}$. Clustering Method 1 (CM1) forms three clusters: the first two items are in group 1, the third and fourth are in group 2, and the fifth and sixth are in group 3, i.e., $\{1, 1, 2, 2, 3, 3\}$. Clustering Method 2 (CM2) creates two clusters: the first three items belong to group 1, and the last three items are in group 2, i.e., $\{1, 1, 1, 2, 2, 2\}$. To manually calculate the RI, we need to consider each unordered pair to determine a and b. In a set of six elements, there are 15 unordered pairs: $\{A, B\}, \{A, C\}, \{A, D\}, \{A, E\}, \{A, F\}$, $\{B, C\}, \{B, D\}, \{B, E\}, \{B, F\}, \{C, D\}, \{C, E\}, \{C, F\}$,$\{D, E\}, \{D, F\}$, and $\{E, F\}$. Given that a represents the number of times a pair of elements is clustered together by both clustering methods, e.g., A, B and E, F, we find $a = 2$. On the other hand, b represents each time a pair of elements is not clustered together by both clustering methods. Pairs such as $\{A, D\}, \{A, E\}, \{A, F\}, \{B, D\}, \{B, E\}$, $\{B, F\}, \{C, E\}$, and $\{C, F\}$ are not clustered together, resulting in $b = 8$. Consequently, the RI is calculated as $\text{RI} = (2 + 8)/15 = 0.67$.

Table C.1: Contingency table for ARI calculation.

$X\backslash Y$	Y_1	Y_2	$\cdots$	Y_s	sums
X_1	n_{11}	n_{12}	$\cdots$	n_{1s}	a_1
X_2	n_{21}	n_{22}	$\cdots$	n_{2s}	a_2
$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\vdots$
X_r	n_{r1}	n_{r2}	$\cdots$	n_{rs}	a_r
sums	b_1	b_2	$\cdots$	b_s	

C.2 Adjusted Rand Index

Even though the RI provides a useful measure of clustering agreement, it has limitations when dealing with data sets where chance agreement may be substantial. By accounting for the expected similarity between random clusterings, the ARI corrects for chance and offers a more accurate and reliable metric. Additionally, the ARI ranges from -1 to 1, where -1 indicates a completely discordant clustering, 0 suggests random agreement, and 1 implies perfect agreement. Consider two random partitions U and V, each with multiple clusters. The number of elements in both clusters u_i and v_j is denoted by n_{ij}. Besides, let n_i and n_j be the number of elements in clusters u_i and cluster v_i respectively. Given the notations illustrated in Table C.1, the ARI can be expressed as follows:

$$\text{ARI} = \frac{\sum_{ij}\binom{n_{ij}}{2} - \left[\sum_i \binom{a_i}{2}\sum_j \binom{b_j}{2}\right] / \binom{n}{2}}{\frac{1}{2}\left[\sum_i \binom{a_i}{2} + \sum_j \binom{b_j}{2}\right] - \left[\sum_i \binom{a_i}{2}\sum_j \binom{b_j}{2}\right] / \binom{n}{2}}.$$

We still provide a simple example to aid understanding.

C.2.1 A Simple Example

Given the data, $X = (1, 2, 3, 3, 2, 1, 1, 3, 3, 1, 2, 2)$ and $Y = (3, 2, 3, 2, 2, 1, 1, 2, 3, 1, 3, 1)$, we observe three different clusters in each partition. The contingency table can be obtained; see Table C.2. Based on the ARI formula, we can derive ARI = 0.0833.

Table C.2: Contingency table for ARI calculation in a simple example.

$X \backslash Y$	Y_1	Y_2	Y_3	sums
X_1	3	0	1	4
X_2	1	2	1	4
X_3	0	2	2	4
sums	4	4	4	

C.3 Comparison of RI and ARI

In summary, the RI and ARI provide quantitative insights into the similarity of clustering solutions. Researchers and practitioners often rely on these indices to evaluate the efficacy of clustering algorithms and compare different partitionings across various fields. Now, we propose employing these indices in tree-based models due to their simplicity without considering complex tree structures, and easy implementation using the R package `fossil`. However, it is important to note that there are cases where the RI may be high, but the ARI is low, leading to opposite conclusions. This occurs when there are many clusters, increasing the likelihood that a pair of elements in both sets are in different clusters. The RI still counts this as a concordant event, whereas the ARI considers all cluster pairs, providing a more comprehensive assessment. To address this, we opt to use the ARI in this thesis.

Appendix D

Data Explorations for the Dataset *dataCar*

This appendix provides tables and figures, aiming to enhance understanding of the numerical transformation for categorical covariates when implementing BCART models. It also illustrates some relationships within the data, such as those between covariates and claims frequency (or severity), or relationships among different covariates.

Based on the discussions in Subsections 2.1.1 and 2.2.1, when processing categorical variables, we calculate empirical claims frequency (or severity) for each categorical level as a numerical replacement, depending on which type of model under consideration, i.e., claims frequency (or severity) modelling. In detail, at each node, we can gather data for each categorical variable at every level. Subsequently, based on this data, the empirical claims frequency can be calculated as ratio of sum of claim counts and sum of exposure. Similarly, the empirical claims severity can be determined by ratio of sum of claim amounts and sum of claim counts. Tables D.1-D.3 present the empirical claims frequency (or severity) for each transformed categorical variable on training data at the root node, serving as a simple illustration. As discussed in Subsection 2.2.1, in our proposed BCART models, the numerical transformation needs to be done after each split, in each updated node. One thing to note is that in Table D.2, we observe that the empirical claims frequency for females is slightly higher than for males (although the difference is small), which contradicts common sense somewhat. However, the empirical claims severity comparison aligns with common sense. The relationship between covariates "*veh_value*" and "*veh_age*" is also provided.

Table D.1: Empirical claims frequency (or severity) for different vehicle body levels on training data (*dataCar*) at the root node. Bold font indicates the smallest and largest correlation coefficients for each level.

	Frequency	Severity
HBACK	0.151	1947
UTE	0.131	2164
STNWG	0.163	1894
HDTOP	0.174	2168
PANVN	0.166	1958
SEDAN	0.153	1678
TRUCK	0.154	2458
COUPE	0.235	2503
MIBUS	0.142	**2580**
MCARA	0.253	712
BUS	**0.387**	1336
CONVT	**0.092**	2296
RDSTR	0.257	**456**

Table D.2: Empirical claims frequency (or severity) for different genders on training data (*dataCar*) at the root node.

	Frequency	Severity
Female	0.16	1733
Male	0.15	2093

Table D.3: Empirical claims frequency (or severity) for different area levels on training data (*dataCar*) at the root node. Bold font indicates the smallest and largest correlation coefficients for each level.

	Frequency	Severity
A	0.155	1754
B	0.162	1758
C	0.156	1919
D	**0.137**	**1739**
E	0.149	2104
F	**0.176**	**2629**

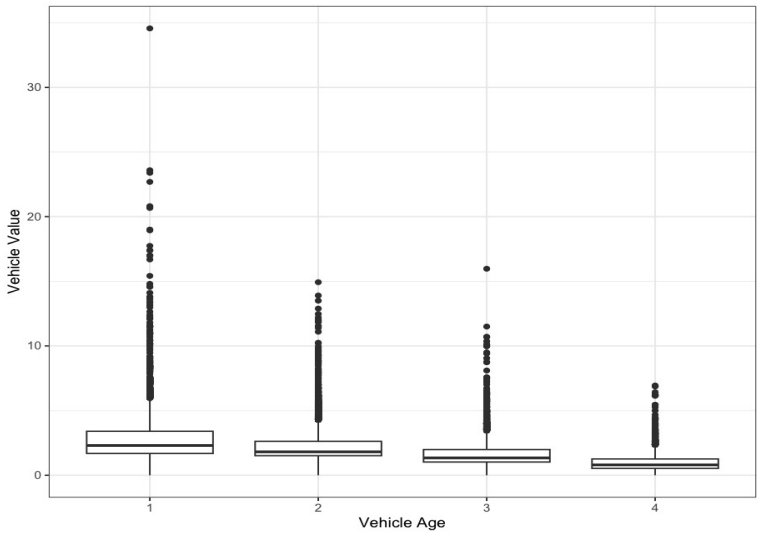

Figure D.1: Scatter plot between vehicle age and vehicle value on training data (*dataCar*).

References

Arsov, N., Pavlovski, M. & Kocarev, L. (2019). Stability of decision trees and logistic regression. *Preprint, https://arxiv.org/pdf/1903.00816.pdf*. 144

Athreya, K.B. & Ney, P.E. (2004). *Branching Processes*. Courier Corporation. 21

Azmi, S.S. & Baliga, S. (2020). An overview of boosting decision tree algorithms utilizing AdaBoost and XGBoost boosting strategies. *Int. Res. J. Eng. Technol*, **7**. 6

Bakirli, G. & Birant, D. (2017). Dtreesim: A new approach to compute decision tree similarity using re-mining. *Turkish Journal of Electrical Engineering and Computer Sciences*, **25**, 108–125. 165

Banerjee, M., Ding, Y. & Noone, A.M. (2012). Identifying representative trees from ensembles. *Statistics in Medicine*, **31**, 1601–1616. 164

Beckett, S., Jee, J., Ncube, T., Pompilus, S., Washington, Q., Singh, A. & Pal, N. (2014). Zero-inflated Poisson (ZIP) distribution: parameter estimation and applications to model data from natural calamities. *Involve, a Journal of Mathematics*, **7**, 751–767. 64

Bennett, K.P. (1992). Decision tree construction via linear programming. Tech. rep., University of Wisconsin-Madison Department of Computer Sciences. 14

Bleich, J., Kapelner, A., George, E.I. & Jensen, S.T. (2014). Variable selection for BART: an application to gene regulation. *The Annals of Applied Statistics*, **8**, 1750–1781. 22

Blier-Wong, C., Cossette, H., Lamontagne, L. & Marceau, E. (2020). Machine learning in P&C insurance: A review for pricing and reserving. *Risks*, **9**, 4. 4

Breiman, L. (2001). Random forests. *Machine Learning*, **45**, 5–32. 6

Breiman, L., Friedman, J., Stone, C.J. & Olshen, R.A. (1984). *Classification and Regression Trees*. CRC press. 5, 16

Bühlmann, H. & Gisler, A. (2005). *A Course in Credibility Theory and its Applications*, vol. 317. Springer. 3

Celeux, G., Forbes, F., Robert, C.P. & Titterington, D.M. (2006). Deviance information criteria for missing data models. *Bayesian Analysis*, **1**, 651–673. 29, 30, 42

Chipman, H. & McCulloch, R.E. (2000). Hierarchical priors for Bayesian CART shrinkage. *Statistics and Computing*, **10**, 17–24. 164

Chipman, H., George, E. & McCulloch, R. (2003). Bayesian treed generalized linear models. *Bayesian Statistics*, **7**, 323–349. 8, 15, 31, 164

Chipman, H., George, E., Hahn, R., McCulloch, R., Pratola, M. & Sparapani, R. (2014). Bayesian Additive Regression Trees, Computational Approaches. *Wiley StatsRef: Statistics Reference Online*, 1–23. 143

Chipman, H.A., George, E.I. & McCulloch, R.E. (1998). Bayesian CART model search. *Journal of the American Statistical Association*, **93**, 935–948. 7, 10, 13, 20, 21, 22, 23, 24, 26, 28, 29, 32, 36, 78, 81, 168

Chipman, H.A., George, E.I. & McCulloch, R.E. (2001). Managing multiple models. In *International Workshop on Artificial Intelligence and Statistics*, 41–48, PMLR. 164

Chipman, H.A., George, E.I. & McCulloch, R.E. (2002). Bayesian treed models. *Machine Learning*, **48**, 299–320. 8, 15, 31

Chipman, H.A., George, E.I., McCulloch, R.E. *et al.* (2010). BART: Bayesian additive regression trees. *The Annals of Applied Statistics*, **4**, 266–298. 9, 26, 27

Christophe Dutang, A.C. (2020). *CASdatasets: Insurance datasets*. R package version 1.0-11. 159

Czado, C., Kastenmeier, R., Brechmann, E.C. & Min, A. (2012). A mixed copula model for insurance claims and claim sizes. *Scandinavian Actuarial Journal*, **2012**, 278–305. 4, 112, 165

David, M. (2015). Auto insurance premium calculation using generalized linear models. *Procedia Economics and Finance*, **20**, 147–156. 2

Denison, D.G., Mallick, B.K. & Smith, A.F. (1998). A Bayesian CART algorithm. *Biometrika*, **85**, 363–377. 7, 22, 26

Denuit, M. & Trufin, J. (2019). *Effective Statistical Learning Methods for Actuaries*. Springer. 4

Denuit, M., Maréchal, X., Pitrebois, S. & Walhin, J.F. (2007). *Actuarial Modelling of Claim Counts: Risk Classification, Credibility and Bonus-malus Systems*. John Wiley & Sons. 2

Denuit, M., Charpentier, A. & Trufin, J. (2021). Autocalibration and Tweedie-dominance for insurance pricing with machine learning. *Insurance: Mathematics and Economics*, **101**, 485–497. 4, 34, 118

Diao, L. & Weng, C. (2019). Regression tree credibility model. *North American Actuarial Journal*, **23**, 169–196. 5, 165

Diebolt, J. & Robert, C.P. (1994). Estimation of finite mixture distributions through Bayesian sampling. *Journal of the Royal Statistical Society: Series B (Methodological)*, **56**, 363–375. 50, 59

Farkas, S., Lopez, O. & Thomas, M. (2021). Cyber claim analysis using Generalized Pareto regression trees with applications to insurance. *Insurance: Mathematics and Economics*, **98**, 92–105. 5, 102

Fink, D. (1997). A compendium of conjugate priors. *See http://www. people. cornell. edu/pages/df36/CONJINTRnew% 20TEX. pdf*, **46**. 91, 97, 99

Frees, E.W., Derrig, R.A. & Meyers, G. (2014). Predictive modeling in actuarial science. *Predictive modeling applications in actuarial science*, **1**. 90, 109, 145

Frees, E.W., Lee, G. & Yang, L. (2016). Multivariate frequency-severity regression models in insurance. *Risks*, **4**, 4. 3, 108, 118

Frempong, N.K., Nicholas, N. & Boateng, M. (2017). Decision tree as a predictive modeling tool for auto insurance claims. *International Journal of Statistics and Applications*, **7**, 117–120. 5

Freund, Y. & Mason, L. (1999). The alternating decision tree learning algorithm. In *ICML*, vol. 99, 124–133. 6

Freund, Y., Schapire, R.E. *et al.* (1996). Experiments with a new boosting algorithm. In *ICML*, vol. 96, 148–156, Citeseer. 6

Friedman, J.H. (2002). Stochastic gradient boosting. *Computational Statistics & Data Analysis*, **38**, 367–378. 6

Fulton, T., Kasif, S. & Salzberg, S. (1995). Efficient algorithms for finding multi-way splits for decision trees. In *Machine Learning Proceedings 1995*, 244–251, Elsevier. 14

Gao, G. & Li, J. (2023). Dependence modeling of frequency-severity of insurance claims using waiting time. *Insurance: Mathematics and Economics*, **109**, 29–51. 112

Garrido, J., Genest, C. & Schulz, J. (2016). Generalized linear models for dependent frequency and severity of insurance claims. *Insurance: Mathematics and Economics*, **70**, 205–215. 3, 11, 108, 112, 113, 156

Gates, A.J. & Ahn, Y.Y. (2017). The impact of random models on clustering similarity. *arXiv preprint arXiv:1701.06508*. 133, 180

Gelman, A., Hwang, J. & Vehtari, A. (2014). Understanding predictive information criteria for Bayesian models. *Statistics and Computing*, **24**, 997–1016. 29, 31

Genest, C. & Nešlehová, J. (2007). A primer on copulas for count data. *ASTIN Bulletin: the Journal of the IAA*, **37**, 475–515. 4

George, E.I. (1998). Bayesian model selection. *Encyclopedia of Statistical Sciences Update*, **3**. 20

Green, P.J. (1995). Reversible jump MCMC computation and Bayesian model determination. *Biometrika*, **82**, 711–732. 26

Green, P.J. & Hastie, D.I. (2009). Reversible jump MCMC. *Genetics*, **155**, 1391–1403. 26

Grubinger, T., Zeileis, A. & Pfeiffer, K.P. (2014). evtree: Evolutionary learning of globally optimal classification and regression trees in R. *Journal of Statistical Software*, **61**, 1–29. 7

Grubinger, Z. & Pfeiffer (2019). *evtree: Evolutionary Learning of Globally Optimal Trees*. R package version 1.0-8. 7

Gschlössl, S. & Czado, C. (2007). Spatial modelling of claim frequency and claim size in non-life insurance. *Scandinavian Actuarial Journal*, **2007**, 202–225. 113

Guelman, L. (2012). Gradient boosting trees for auto insurance loss cost modeling and prediction. *Expert Systems with Applications*, **39**, 3659–3667. 6

Hastie, T. & Tibshirani, R. (1987). Generalized additive models: some applications. *Journal of the American Statistical Association*, **82**, 371–386. 3

He, J., Yalov, S. & Hahn, P.R. (2019). XBART: Accelerated Bayesian additive regression trees. In *the 22nd International Conference on Artificial Intelligence and Statistics*, 1130–1138, PMLR. 143

Henckaerts, R. (2020). *distRforest: Random Forests with Distribution-Based Loss Functions*. 146

Henckaerts, R., Antonio, K., Clijsters, M. & Verbelen, R. (2018). A data driven binning strategy for the construction of insurance tariff classes. *Scandinavian Actuarial Journal*, **2018**, 681–705. 2, 165

Henckaerts, R., Côté, M.P., Antonio, K. & Verbelen, R. (2021). Boosting insights in insurance tariff plans with tree-based machine learning methods. *North American Actuarial Journal*, **25**, 255–285. 2, 4, 5, 6, 35, 36, 90, 109, 145, 165

Hernández, B., Raftery, A.E., Pennington, S.R. & Parnell, A.C. (2018). Bayesian additive regression trees using Bayesian model averaging. *Statistics and computing*, **28**, 869–890. 9

Hill, J., Linero, A. & Murray, J. (2020). Bayesian additive regression trees: A review and look forward. *Annual Review of Statistics and its Application*, **7**, 251–278. 9

Hu, C., Quan, Z. & Chong, W.F. (2022). Imbalanced learning for insurance using modified loss functions in tree-based models. *Insurance: Mathematics and Economics*, **106**, 13–32. 5

Hubert, L. & Arabie, P. (1985). Comparing partitions. *Journal of Classification*, **2**, 193–218. 133, 180

Jørgensen, B. & Paes De Souza, M.C. (1994). Fitting Tweedie's Compound Poisson model to insurance claims data. *Scandinavian Actuarial Journal*, **1994**, 69–93. 3, 117

Kapelner, A. & Bleich, J. (2013). bartMachine: Machine learning with Bayesian additive regression trees. *arXiv preprint arXiv:1312.2171*. 9, 167, 172

Kapelner, A. & Bleich, J. (2015). Prediction with missing data via Bayesian additive regression trees. *Canadian Journal of Statistics*, **43**, 224–239. 166

Kapelner, A. & Bleich, J. (2023). *bartMachine: Bayesian Additive Regression Trees*. R package version 1.3.4.1. 9

Kindo, B.P., Wang, H. & Peña, E.A. (2016). Multinomial probit Bayesian additive regression trees. *Stat*, **5**, 119–131. 23, 27

Larsen, D.R. & Speckman, P.L. (2004). Multivariate regression trees for analysis of abundance data. *Biometrics*, **60**, 543–549. 5

Lee, G.Y. & Shi, P. (2019). A dependent frequency-severity approach to modeling longitudinal insurance claims. *Insurance: Mathematics and Economics*, **87**, 115–129. 3

Lee, S.C. (2020). Delta boosting implementation of Negative Binomial regression in actuarial pricing. *Risks*, **8**, 19. 6, 11, 35, 36, 38, 43, 46, 47, 143, 162

Lee, S.C. (2021). Addressing imbalanced insurance data through zero-inflated Poisson regression with boosting. *ASTIN Bulletin: the Journal of the IAA*, **51**, 27–55. 6, 11, 34, 35, 36, 38, 50, 55, 143, 162

Lee, S.C. & Lin, S. (2018). Delta boosting machine with application to general insurance. *North American Actuarial Journal*, **22**, 405–425. 6

Lee, S.K. (2005). On generalized multivariate decision tree by using GEE. *Computational Statistics & Data Analysis*, **49**, 1105–1119. 5

Lee, S.K. & Jin, S. (2006). Decision tree approaches for zero-inflated count data. *Journal of Applied Statistics*, **33**, 853–865. 5

Lee, W., Park, S.C. & Ahn, J.Y. (2019). Investigating dependence between frequency and severity via simple generalized linear models. *Journal of the Korean Statistical Society*, **48**, 13–28. 4

Lindholm, M., Lindskog, F. & Palmquist, J. (2023). Local bias adjustment, duration-weighted probabilities, and automatic construction of tariff cells. *Scandinavian Actuarial Journal*, 1–28. 5, 165

Linero, A.R. (2017). A review of tree-based Bayesian methods. *Communications for Statistical Applications and Methods*, **24**, 543–559. 9, 23

Linero, A.R. (2018). Bayesian regression trees for high-dimensional prediction and variable selection. *Journal of the American Statistical Association*, **113**, 626–636. 21, 22, 164

Linero, A.R. & Yang, Y. (2018). Bayesian regression tree ensembles that adapt to smoothness and sparsity. *Journal of the Royal Statistical Society. Series B (Statistical Methodology)*, **80**, 1087–1110. 9

Linero, A.R., Sinha, D. & Lipsitz, S.R. (2020). Semiparametric mixed-scale models using shared Bayesian forests. *Biometrics*, **76**, 131–144. 10, 11, 23, 27, 118, 163, 165

Liu, Y., Ročková, V. & Wang, Y. (2021). Variable selection with ABC Bayesian forests. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, **83**, 453–481. 22

Loh, W.Y. (2014). Fifty years of classification and regression trees. *International Statistical Review*, **82**, 329–348. 5

Marie-Laure Delignette-Muller, C.D. & Pouillot, R. (2023). *fitdistrplus: Help to Fit of a Parametric Distribution to Non-Censored or Censored Data*. R package version 1.1-11. xiii, 147, 148

McCulloch, S. & Spanbauer, G. (2023). *BART: Bayesian Additive Regression Trees*. R package version 2.9.4. 9

Mehmet, M. & Saykan, Y. (2005). On a bonus-malus system where the claim frequency distribution is Geometric and the claim severity distribution is Pareto. *Hacettepe Journal of Mathematics and Statistics*, **34**, 75–81. 102, 108

Meng, S., Gao, Y. & Huang, Y. (2022). Actuarial intelligence in auto insurance: Claim frequency modeling with driving behavior features and improved boosted trees. *Insurance: Mathematics and Economics*, **106**, 115–127. 5

Meng, X.L. & Van Dyk, D.A. (1999). Seeking efficient data augmentation schemes via conditional and marginal augmentation. *Biometrika*, **86**, 301–320. 10, 27

Murray, J.S. (2021). Log-linear Bayesian additive regression trees for multinomial Logistic and count regression models. *Journal of the American Statistical Association*, **116**, 756–769. 9, 10, 23, 27, 38, 43, 49, 50, 51, 59, 63, 65, 108, 163

Nash, R.V. & Grothendieck, G. (2023). *optimx: Expanded Replacement and Extension of the "optim" Function*. R package version 2023-10.21. 65

Naya, H., Urioste, J.I., Chang, Y.M., Rodrigues-Motta, M., Kremer, R. & Gianola, D. (2008). A comparison between Poisson and zero-inflated Poisson regression models with an application to number of black spots in Corriedale sheep. *Genetics Selection Evolution*, **40**, 1–16. 34

Nelder, J.A. & Wedderburn, R.W. (1972). Generalized linear models. *Journal of the Royal Statistical Society: Series A (General)*, **135**, 370–384. 2

Nye, T.M., Lio, P. & Gilks, W.R. (2006). A novel algorithm and web-based tool for comparing two alternative phylogenetic trees. *Bioinformatics*, **22**, 117–119. 135, 165

Ohlsson, E. & Johansson, B. (2010). *Non-life Insurance Pricing with Generalized Linear Models*, vol. 174. Springer. 1, 3, 17, 118

Omari, C.O., Nyambura, S.G. & Mwangi, J.M.W. (2018). Modeling the frequency and severity of auto insurance claims using statistical distributions. *Journal of Mathematical Finance*. 90, 108, 137, 160

Prado, E.B., Moral, R.A. & Parnell, A.C. (2021a). Bayesian additive regression trees with model trees. *Statistics and Computing*, **31**, 1–13. 9

Prado, E.B., Parnell, A.C., Murphy, K., McJames, N., O'Shea, A. & Moral, R.A. (2021b). Accounting for shared covariates in semi-parametric Bayesian additive regression trees. *arXiv preprint arXiv:2108.07636*. 9

Pratola, M.T. (2016). Efficient Metropolis–Hastings proposal mechanisms for Bayesian regression tree models. *Bayesian Analysis*, **11**, 885–911. 9, 24, 164

Quan, Z. (2019). *Insurance Analytics with Tree-Based Models*. PhD thesis, University of Connecticut. 5

Quan, Z., Wang, Z., Gan, G. & Valdez, E.A. (2020). Hybrid Tree-based Models for Insurance Claims. *arXiv preprint arXiv:2006.05617*. 165

Quijano Xacur, O.A. & Garrido, J. (2015). Generalised linear models for aggregate claims: to Tweedie or not? *European Actuarial Journal*, **5**, 181–202. 3, 108, 118

Quijano Xacur, O.A. *et al.* (2019). *Computational Bayesian Methods for Insurance Premium Estimation*. Ph.D. thesis, Concordia University. 137, 160

Rand, W.M. (1971). Objective criteria for the evaluation of clustering methods. *Journal of the American Statistical association*, **66**, 846–850. 133, 180

Rinne, H. (2008). *The Weibull distribution: a handbook*. CRC press. 91

Roberts, G.O. & Rosenthal, J.S. (2004). General state space Markov chains and MCMC algorithms. 24

Rocková, V., Van der Pas, S. *et al.* (2020). Posterior concentration for Bayesian regression trees and forests. *Annals of Statistics*, **48**, 2108–2131. 10, 21, 22

Rodrigues, J. (2003). Bayesian analysis of zero-inflated distributions. *Communications in Statistics-Theory and Methods*, **32**, 281–289. 59

Saha, E. (2023). Theory of Posterior Concentration for Generalized Bayesian Additive Regression Trees. *arXiv preprint arXiv:2304.12505*. 22

Shi, P., Feng, X. & Ivantsova, A. (2015). Dependent frequency-severity modeling of insurance claims. *Insurance: Mathematics and Economics*, **64**, 417–428. 4, 102, 108

Siswadi & Quesenberry, C. (1982). Selecting among Weibull, LogNormal and Gamma distributions using complete and censored smaples. *Naval Research Logistics Quarterly*, **29**, 557–569. 92

Smith, M.S. (2011). Bayesian approaches to copula modelling. *arXiv preprint arXiv:1112.4204*. 4, 165

Smith, M.S. & Khaled, M.A. (2012). Estimation of copula models with discrete margins via Bayesian data augmentation. *Journal of the American Statistical Association*, **107**, 290–303. 4, 165

Smyth, G.K. & Jørgensen, B. (2002). Fitting Tweedie's Compound Poisson model to insurance claims data: dispersion modelling. *ASTIN Bulletin: the Journal of the IAA*, **32**, 143–157. 108, 117, 118

Song, P.X.K., Li, M. & Yuan, Y. (2009). Joint regression analysis of correlated data using Gaussian copulas. *Biometrics*, **65**, 60–68. 112

Sparapani, R., Spanbauer, C. & McCulloch, R. (2021). Nonparametric machine learning and efficient computation with Bayesian additive regression trees: the BART R package. *Journal of Statistical Software*, **97**, 1–66. 9, 143

Spiegelhalter, D.J., Best, N.G., Carlin, B.P. & Van Der Linde, A. (2002). Bayesian measures of model complexity and fit. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, **64**, 583–639. 10, 29, 31

Spiegelhalter, D.J., Best, N.G., Carlin, B.P. & Van der Linde, A. (2014). The deviance information criterion: 12 years on. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, **76**, 485–493. 29, 31

Tanner, M.A. & Wong, W.H. (1987). The calculation of posterior distributions by data augmentation. *Journal of the American Statistical Association*, **82**, 528–540. 27, 59

Therneau, T. & Atkinson, B. (2023). *rpart: Recursive Partitioning and Rregression Trees*. R package version 4.1.21. 77

Timofeev, R. (2004). Classification and Regression Trees (CART) theory and applications. *Humboldt University, Berlin*, **54**. 16

Van Dyk, D.A. & Meng, X.L. (2001). The art of data augmentation. *Journal of Computational and Graphical Statistics*, **10**, 1–50. 10, 27, 28

Vavrek, M.J. (2020). *fossil: Palaeoecological and Palaeogeographical Analysis Tools*. R package version 0.4.0. 135

Voss, J. (2013). *An introduction to statistical computing: a simulation-based approach*. John Wiley & Sons. 173, 175, 176

Watanabe, S. & Opper, M. (2010). Asymptotic equivalence of Bayes cross validation and widely applicable information criterion in singular learning theory. *Journal of Machine Learning Research*, **11**. 31

Wolny-Dominiak, A. & Trzesiok, M. (2014). *insuranceData: A Collection of Insurance Datasets Useful in Risk Classification in Non-life insurance*. R package version 1.0. 136

Wu, Y., Tjelmeland, H. & West, M. (2007). Bayesian CART: Prior specification and posterior simulation. *Journal of Computational and Graphical Statistics*, **16**, 44–66. 8, 22, 24, 164

Wüthrich, M.V. (2020). Bias regularization in neural network models for general insurance pricing. *European Actuarial Journal*, **10**, 179–202. 4, 34

Wüthrich, M.V. (2022). The balance property in neural network modelling. *Statistical Theory and Related Fields*, **6**, 1–9. 34

Wuthrich, M.V. (2022). *Non-life Insurance: Mathematics & Statistics*. Available at SSRN 2319328. 2, 43, 47

Wuthrich, M.V. & Buser, C. (2022). *Data Analytics for Non-Life Insurance Pricing (January 9, 2023)*. Available at SSRN:2870308. 4, 19, 20, 36

Wüthrich, M.V. & Merz, M. (2008). *Stochastic claims reserving methods in insurance*. John Wiley & Sons. 91

Wüthrich, M.V. & Merz, M. (2023). *Statistical foundations of actuarial learning and its applications*. Springer Nature. 4, 38, 43, 46, 90

Yang, Y., Qian, W. & Zou, H. (2018). Insurance premium prediction via gradient tree-boosted Tweedie Compound Poisson models. *Journal of Business & Economic Statistics*, **36**, 456–470. 6, 118

Yang, Z. & Qian, G.R. (2022). *TDboost: A Boosted Tweedie Compound Poisson Model*. R package version 1.4. 6

Yu, Y. & Lambert, D. (1999). Fitting trees to functional data, with an application to time-of-day patterns. *Journal of Computational and Graphical Statistics*, **8**, 749–762. 5

Zhou, M., Li, L., Dunson, D. & Carin, L. (2012). LogNormal and Gamma mixed Negative Binomial regression. In *Proceedings of the International Conference on Machine Learning. International Conference on Machine Learning*, vol. 2012, 1343, NIH Public Access. 43, 49

Zilko, A.A. & Kurowicka, D. (2016). Copula in a multivariate mixed discrete-continuous model. *Computational Statistics & Data Analysis*, **103**, 28–55. 165