DATA IMPUTATION FOR LOSS RESERVING

DATA IMPUTATION FOR LOSS RESERVING

By YILONG ZHAI, BMath

A Thesis Submitted to the School of Graduate Studies in Partial

Fulfillment of the Requirements for

the Degree Master of Science

McMaster University

MASTER OF SCIENCE (2024)

Hamilton, Ontario, Canada (Mathematics and Statistics)

TITLE:	Data Imputation For Loss Reserving
AUTHOR:	Yilong Zhai BMath (Statistics), University of Waterloo, Waterloo,Canada
SUPERVISOR:	Dr. Anas Abdallah Dr. Mathieu Pigeon
NUMBER OF PAGES:	ix, 104

Abstract

This master thesis delves into machine learning predictive modelling to predict missing values in loss reserving, focusing on predicting missing values for individual features (age, accident year, etc) and annual insurance payments. Leveraging machine learning techniques such as random forest and decision trees, we explore their performance for missing value prediction compared to traditional regression models. Moreover, the study transforms individual payments into run-off triangle versions. It uses the imputed dataset and complete dataset to compare the performance of different data imputation models by the loss reserves estimation from the Mack and GLM reserves model. By evaluating the performance of these diverse techniques, this research aims to contribute valuable insights to the evolving landscape of predictive analytics in insurance, guiding industry practices toward more accurate and efficient modelling approaches.

Acknowledgements

I really appreciate the support from my family, my parents always stand by my side for all my decisions, I have not come to home for four years during the pandemic, but I am pretty sure that the love they give me remains the same. I also want to thank my girlfriend Wenjun here, she gave me really a lot of help and love when I was struggling with my study and my life, I will always remember the time she spent with me when I was confused about the future. I could not be there without the support from Dr. Abdallah and Dr. Pigeon, they led me into the field of research, they gave me great patience and respect, and always care about my research and my life. Thank you Dr. Jeganathan and Dr. Forman, it is my pleasure to invite you as the committee members. I am grateful to every friend I met on this journey, and always look forward to the next journey.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

The insurance industry, a pillar of modern economic systems, relies heavily on robust financial strategies to ensure stability and continuity. At the core of these strategies lies the practice of loss reserving, which involves estimating future claim payments based on historical data [Pigeon et al., 2013]. Accurate loss reserving is essential for insurers to allocate resources appropriately, meet financial obligations, and comply with regulatory requirements [Grace and Leverty, 2010]. These reserves represent the estimated amount of money set aside by insurers to cover anticipated claim payments and associated expenses, thereby ensuring financial solvency and stability [Berquist and Sherman, 1977]. By accurately estimating loss reserves, insurers can effectively manage their liabilities, assess their financial health, and fulfill regulatory requirements. Loss reserves encompass various categories of claims, including those arising from property damage, bodily injury, liability lawsuits, and other insured events. Determining adequate reserves demands a sophisticated understanding of actuarial principles, statistical modeling techniques, and industry-specific factors, as insurers strive to balance prudence and profitability. However, missing data within historical claim records poses a significant challenge to generating precise and reliable loss

reserve estimates.

Missing data can arise for various reasons, such as reporting errors, inconsistencies, or incomplete documentation. Traditional methods developed by [Efron, 1994] of addressing this issue, such as mean imputation or omitting incomplete records, can introduce bias and inaccuracies into loss reserve estimates. As the insurance landscape grows in complexity and scale, the demand for more sophisticated techniques to handle missing data has become increasingly apparent [Zhang, 2016].

In response to these challenges, machine learning (ML) has emerged as a powerful tool for enhancing data imputation in loss reserving processes [Jordan and Mitchell, 2015]. By leveraging the capabilities of decision trees and random forests, among other ML techniques, insurers can transform incomplete datasets into more comprehensive and accurate resources for loss reserving analysis (see [Breiman, 1984] and [Breiman, 2001]). These methods offer the potential to discern intricate patterns within the data and generate intelligent imputations that align with the underlying distribution of the information.

This thesis aims to explore and leverage the capabilities of machine learning methodologies, specifically Random Forest and Decision Trees, to address data imputation challenges within the domain of loss reserving. In addition to these modern machine learning techniques, we employ conventional statistical methods, such as regression models, to impute missing values in insured features [Draper and Smith, 1998]. Subsequently, we continue to utilize these models to make predictions regarding payment indicators and annual payments for the following five consecutive development years. We intend to apply a comprehensive evaluation framework. For categorical missing variables, we employ the F1 score, a standard metric for assessing the quality of classification models. We utilize the mean squared error, a prevalent measure for quantifying predictive accuracy, to gauge the performance of

machine learning techniques and traditional statistical methods in the context of missing value imputation and individual payment prediction. Furthermore, we aggregate individual payments into a summary format, generating a run-off triangle [Kaplan and Ruback, 1995]. Each cell in these tables encapsulate the cumulative payment amount for a specific accident year and all previous development years. This aggregation process facilitate a broader analysis and understanding of the overall payment patterns over time. In the final phase of our investigation, we leverage the aggregate tables to make predictions for loss reserves spanning five consecutive years. This is achieved using the established Mack and GLM reserve models, two well-recognized actuarial methodologies for estimating outstanding claims and loss reserves in insurance (see [Mack, 1994] and [Björkwall et al., 2011]).

In Chapter 2, we introduce the statistical tools used to do the data imputation task and the evaluation matrices used to measure the performance of different data imputation methods. In Chapter 3, the motivation of loss reserves, deterministic and stochastic models of loss reserves estimation are presented; meanwhile, Section 3.5 shows how the models in Chapter 2 and Chapter 3 are used. In Chapter 4, the results are interpreted for both the data imputation and loss reserves estimation tasks.

This research seeks to contribute to actuarial and data science by exploring innovative approaches to address data imputation challenges in the context of loss reserving. It particularly emphasizes the applicability and performance of machine learning techniques in this specialized domain.

Chapter 2

Statistical Tools

In this chapter, we explore data imputation techniques comprehensively, aiming to fill the void in missing values for individual features and annual insurance payments. In Section 2.1, we delve into traditional regression models. Section 2.3 introduces the power of machine learning techniques, specifically random forest and decision trees, elevating the precision and efficiency of our predictive models. Furthermore, in Section 2.4, we meticulously examine and compare the performance of these techniques through the lens of evaluation metrics. The F1 score and Mean Squared Error (MSE) take center stage as our guiding measures, providing a nuanced understanding of the efficiency and accuracy achieved by both traditional and machine learning approaches in the complicated environment of data imputation for insurance analytics.

2.1 The Bootstrap Method

The bootstrap, a statistical tool first introduced by [Efron, 1979], plays an essential role in the statistical inference. Rooted in resampling with replacement, bootstrap is a powerful

tool for estimating the variability and uncertainty associated with statistical estimators. Bootstrap is also introduced to the data imputation area by [Efron, 1994].

The fundamental premise of the bootstrap method involves drawing random samples, with replacement, from the observed data to create multiple surrogate datasets. Through these resampled datasets, an array of new estimates is derived, enabling the construction of empirical distributions for the statistic of interest; this process mimics the inherent randomness in the data generation process, facilitating a comprehensive understanding of the sampling variability of the estimators.

The advantage of the bootstrap method lies in its broad applicability across diverse statistical problems, providing a versatile and data-driven approach to uncertainty quantification. Through its detailed resampling protocol, bootstrap has become an indispensable tool in modern statistical practice, which could be used in various domains, including insurance [Ostaszewski and Rempala, 2000], econometrics, and machine learning.

The bootstrap involves drawing samples with replacements from the original dataset. To resample the data, we present the non-parametric bootstrap introduced by [Efron, 1982]. Let $Z = \{Z_1, Z_2, \ldots, Z_n\}$ be the original random sample with size n and independent and identically distributed observed values $\{z_1, z_2, \ldots, z_n\}$. A bootstrap sample Z_b^* is drawn by randomly selecting n observations with replacement from Z, where $b = 1, 2, \ldots B$:

$$Z_b^* = (Z_1^*, Z_2^*, \ldots, Z_n^*).$$

The bootstrap estimator of a statistic Θ is computed based on the resampled data:

$$\hat{\theta}_b^* = f(z_b^*),$$

where $f(\cdot)$ is a function that computes the estimator.

Then, we obtained a set a B bootstrap replicates $\{\hat{\theta}_1^*, \hat{\theta}_2^*, \ldots, \hat{\theta}_B^*\}$.

A bootstrap confidence interval (CI) [DiCiccio and Efron, 1996] for the statistic Θ is constructed by finding the $\alpha/2$ and $1-\alpha/2$ percentiles of the bootstrap pseudo response:

$$\text{CI}_\alpha = \left(\hat{\theta}^*_{(\alpha/2)}, \hat{\theta}^*_{(1-\alpha/2)}\right).$$

The bootstrap bias of an estimator is calculated as:

$$\text{Bias}(\hat{\theta}_b^*) = \mathbb{E}(\hat{\theta}_b^*) - \theta,$$

where θ is the true value of the statistic.

2.2 Classical Techniques

Regression is one of the most efficient and commonly used classical methods for missing data imputation developed by [Scheffer, 2002]. They have been successfully applied in numerous scenarios and are particularly effective when dealing with simple cases of missing data. The reliability of regression-based imputation methods largely depends on the quality and relevance of the features and the underlying assumptions of the regression model offered by [Musil et al., 2002]. Regression models can provide accurate imputations in relatively straightforward cases with well-understood relationships between the variables. They work well when there is a linear relationship between the variables, and the data is missing at random or completely at random (MCAR), presented by [Heitjan and Basu, 1996]. MCAR is a strong assumption that the missingness is not systematically related to any variable, observed or unobserved. Missing at random is a more relaxed assumption

compared to MCAR. It allows the probability of missingness to depend on observed variables in the dataset but not on the unobserved values. In a scenario where missing data is entirely random, the probability of missingness for a variable is unrelated to the observed or unobserved values in the dataset. In these situations, regression imputation is a robust approach. However, the reliability of regression-based imputation methods can be challenged in more complex cases, such as high-dimensionality. High-dimensional datasets, characterized by abundant predictors relative to observations, pose substantial challenges for regression methods. Issues encompass overfitting, multicollinearity, increased computational complexity, and difficulties in variable selection [Osman et al., 2018].

The missing values can belong to three types of variables: discrete, categorical, and continuous. For categorical variables, we employ a logistic regression model [Wright, 1995] or multinomial logistic regression if the variables have more than two categories [Kwak and Clayton-Matthews, 2002]. This method allows us to estimate the probabilities of each category relative to a reference category and handle multi-class classification. For discrete and continuous variables, we employ generalized linear models [Nelder and Wedderburn, 1972].

2.2.1 Generalized Linear Model

The generalized linear model (GLM) is a cornerstone of statistical modeling for exponential family distributions. GLM emerges as a robust and adaptable solution, amplifying our capacity to address missing values effectively. In this context, we indicate the computational efficiency and versatility that a generalized linear model brings to the forefront, highlighting its pivotal role in our pursuit of precision in predictive modeling.

Let $\mathbf{Y}$ be a random vector with dimension of $n \times 1$, containing the response variable and $\mathbf{X}$

be a design matrix with dimension of $n \times m$ containing covariates (also known as features or variables). n is the sample size and m is the number of features. Each row of $\mathbf{X}$ represents a data point, and each column represents a different covariate. Moreover, a column of 1 is inserted as the first column to model the intercept. The main formula is

$$g(E(\mathbf{Y})) = \mathbf{X}\boldsymbol{\beta}, \tag{2.2.1}$$

where $\boldsymbol{\beta}$ is a vector with dimension of $m + 1 \times 1$ of unknown parameters and $g(\cdot)$ is the link function, describes the relationship between the expected value (mean) of the response variable $\mathbf{Y}$ and the predictors $\mathbf{X}$.

The choice of the link function depends on the type of response variable (e.g., Gaussian, Poisson, Binomial) and the specific GLM being used. $\mathbf{Y_i}$ typically represents the response variable for the i-th observation in the dataset. Each $\mathbf{Y_i}$ is assumed to be independently and identically distributed according to a specific probability distribution from the GLM family. The exponential family is a class of probability distributions that plays a fundamental role in statistics and mathematical modeling developed by [Brown, 1986]. The exponential family encompasses a wide range of probability distributions commonly encountered in statistical analyses. The family is characterized by a specific mathematical form that facilitates elegant statistical inference and estimation procedures. From the exponential family, the probability density function (pdf) or probability mass function (pmf) is often written in the canonical form:

$$f(x|\theta) = h(x) \exp\left(\frac{T(x) \cdot \theta - A(\theta)}{C(\phi)}\right)$$

where x is the observed data, θ is the parameter of interest, $T(x)$ come up the sufficient

statistic of the random variable, $h(x)$ come up the base measure, $A(\theta)$ come up the cumulant function, and $C(\phi)$ come up the dispersion parameter (if applicable).

The Gaussian family is commonly used for non-negative and continuous response variables [Goodman, 1963]. In this family, the response assumes a normal distribution. The identity link function establishes a direct relationship between the linear predictor η_i and the mean μ_i:

$$Y_i \sim \mathcal{N}(\mu_i, \sigma^2) \quad \text{with} \quad \eta_i = \mu_i,$$

For binary or count response variables, the Binomial family is often employed [Hilbe, 2011]. The response follows a binomial distribution. The logit link function connects the linear predictor η_i to the probability of success p_i:

$$Y_i \sim \text{Binomial}(n_i, p_i) \quad \text{with} \quad \eta_i = \text{logit}(p_i),$$

The Poisson family is suitable for count data, assuming a Poisson distribution for the random component [Aryal and Yousof, 2017]. The log link function establishes a connection between the linear predictor η_i and the rate parameter λ_i:

$$Y_i \sim \text{Poisson}(\lambda_i) \quad \text{with} \quad \eta_i = \log(\lambda_i).$$

Here is a summarized table for the link functions.

Table 2.1: Link Functions for Different Exponential Family Distributions

Distribution	**Link Function**
Binomial	Logit: $g(\mu) = \log\left(\frac{\mu}{1-\mu}\right)$
	Probit: $g(\mu) = \Phi^{-1}(\mu)$
	Complementary Log-Log: $g(\mu) = \log(-\log(1-\mu))$
Poisson	Log: $g(\mu) = \log(\mu)$
Gamma	Inverse: $g(\mu) = -\frac{1}{\mu}$
Inverse Gaussian	Inverse Squared: $g(\mu) = -\frac{1}{\mu^2}$
Normal	Identity: $g(\mu) = \mu$
Exponential	Log: $g(\mu) = \log(\mu)$

These link functions play a crucial role in connecting the linear predictor to the parameters of the probability distribution, allowing for flexibility in modeling various types of data in the GLM framework.

Generalized Linear Models (GLMs) involve estimating unknown parameters to model the relationship between predictor variables and a response variable. The estimation procedure typically relies on Maximum Likelihood Estimation (MLE) present by [Myung, 2003]. We explore the estimation process for different GLM families. MLE is a technique for estimating statistical model parameters by maximizing the likelihood function, which denotes the probability of observing the given data within the assumed model. The process involves defining the likelihood function based on the model and data, optionally taking the logarithm for computational convenience, calculating the score function (a derivative of the log-likelihood), setting the score function to zero, and solving for the parameters.

Depending on the model's complexity, analytical solutions or iterative methods, such as numerical optimization algorithms, may be employed to determine parameter estimates. The likelihood function is given by [Myung, 2003].

For a response variable follows exponential family distribution, the likelihood function is given by:

$$L(\theta|x) = \prod_{i=1}^{n} f(x_i|\theta).$$

The MLE estimates of $\hat{\boldsymbol{\theta}}$ are obtained by maximizing the log-likelihood function, which involves solving nonlinear equations.

Now, take the logarithm to obtain the log-likelihood:

$$\ell(\theta|x) = \log L(\theta|x),$$

$$\ell(\theta|x) = \sum_{i=1}^{n} \left(\frac{T(x_i) \cdot \theta - A(\theta)}{C(\phi)} + \log h(x_i) \right).$$

To find the MLE estimates of $\hat{\boldsymbol{\theta}}$, we differentiate the log-likelihood with respect to $\boldsymbol{\theta}$ and set the derivatives equal to zero. The score function, denoted as $S(\theta)$ or $\frac{\partial \ell}{\partial \theta}$, plays a crucial role in finding the parameter values that maximize the likelihood function [Murphy and Van der Vaart, 2000]. Mathematically, the score function is defined as:

$$S(\theta) = \frac{\partial \ell}{\partial \theta},$$

where ℓ is the log-likelihood function. The score function provides information about the direction and magnitude of change in the log-likelihood function as the parameter values

vary. Specifically, it indicates how the log-likelihood function changes with respect to each parameter. In MLE, the goal is to find the parameter values that maximize the log-likelihood function, or equivalently, set the score function equal to zero:

$$S(\hat{\theta}) = 0,$$

where $\hat{\theta}$ represents the maximum likelihood estimate of the parameter vector. The score function is essential for iterative optimization algorithms, such as the Newton-Raphson method or the Fisher scoring algorithm, which iteratively update the parameter estimates based on the score function until convergence is achieved. By computing and analyzing the score function, one can identify the optimal parameter values that best fit the observed data and maximize the likelihood of observing the data given the model.

For a Poisson-distributed response variable, commonly used for count data, the likelihood function is given by:

$$\mathcal{L}(\boldsymbol{\beta}) = \prod_{i=1}^{n} \frac{e^{-\lambda_i} \lambda_i^{y_i}}{y_i!},$$

where λ_i is the rate parameter. The MLE estimates $\hat{\boldsymbol{\beta}}$ are obtained by maximizing the log-likelihood function.

Now, take the logarithm to obtain the log-likelihood:

$$\ell(\boldsymbol{\beta}) = \sum_{i=1}^{n} \left(-\lambda_i + y_i \log(\lambda_i) - \log(y_i!)\right),$$

To find the MLE estimates $\hat{\boldsymbol{\beta}}$, we differentiate the log-likelihood with respect to $\boldsymbol{\beta}$ and set the derivatives equal to zero.

The predicted mean (rate) for a new observation is given by:

$$\hat{\lambda}_{\text{new}} = e^{\eta_{\text{new}}},$$

where $\eta_{\text{new}} = \boldsymbol{x}_{\text{new}}^T \hat{\boldsymbol{\beta}}$, and $\hat{\boldsymbol{\beta}}$ represents the estimated coefficients.

2.2.2 Logistic Regression

In the earlier Section 2.2.1, we introduced GLMs. Consequently, in this section, the emphasis shifts to logistic regression models, a specific type of GLM, as a more proficient solution for classification challenges is presented by [Pearce and Ferrier, 2000]. Logistic regression is relatively simple and interpretable compared to more complex models. It is easy to implement and understand, making it a good choice for scenarios where interpretability is preferred. Logistic regression provides probabilities rather than discrete predictions. This is beneficial in scenarios where understanding the confidence or uncertainty associated with forecasts is essential. This modification underscores the suitability of logistic regression in the context of categorical missing data imputation.

Recall that $\mathbf{Y}$ is a random vector containing the response variable and $\mathbf{X}$ is a design matrix. Now, we assume that the response variable can take only two values: 0 and 1

$$P(Y = 1|\mathbf{X}) = \frac{1}{1 + e^{-(\beta_0 + \beta_1 x_1 + \beta_2 x_2 + ... + \beta_p x_p)}},$$

$$P(Y = 0|\mathbf{X}) = \frac{e^{-(\beta_0 + \beta_1 x_1 + \beta_2 x_2 + ... + \beta_p x_p)}}{1 + e^{-(\beta_0 + \beta_1 x_1 + \beta_2 x_2 + ... + \beta_p x_p)}},$$

and 0 elsewhere. where $P(Y = 1|\mathbf{X})$ represents the probability of the binary outcome (class 1) given the predictor variables $\mathbf{X}$. $\beta_0, ..., \beta_p$ are the coefficients (parameters) of the logistic regression model. The coefficients for each predictor variable represent how

much the log-odds of the probability of the corresponding category being the positive class changes with a one-unit increase in that predictor variable, assuming all other predictor variables remain fixed.

The formula of multinomial logistic regression model is:

For $k = 1, 2 \ldots, K$, where K is the number of classes of the multinomial logistic regression model,

$$P(Y = k|\mathbf{X}) = \frac{\exp(\beta_{k0} + \beta_{k1}x_1 + \beta_{k2}x_2 + \ldots + \beta_{kp}x_p)}{\sum_{j=1}^{K} \exp(\beta_{j0} + \beta_{j1}x_1 + \beta_{j2}x_2 + \ldots + \beta_{jp}x_p)}, \tag{2.2.2}$$

and 0 elsewhere. where $P(Y = k|\mathbf{X})$ is the probability of the data point belonging to class k given the random sample $\mathbf{X}$. $\beta_{k0}, ..., \beta_{kp}$ are the coefficients (parameters) for class k. These represent the relationship between the predictor variables and the log-odds of the data point being in class k.

For multinomial logistic regression we use a 'one vs. others' approach to address the multiple categories. In the "one vs. others" approach, the classification task is decomposed into multiple binary classification sub-problems. Specifically, for each class, a separate binary classifier is trained to discriminate instances of that class from the rest of the classes collectively. Consequently, the overall problem is transformed into a series of binary classification tasks, each addressing the identification of instances belonging to a specific class against all other classes.

$$\begin{aligned}
\text{logit}(P(Y_i = 1)|X) &= \beta_{01} + \beta_{11}X_{i1} + \beta_{21}X_{i2} + \cdots + \beta_{M1}X_{iM} \\
\text{logit}(P(Y_i = 2)|X) &= \beta_{02} + \beta_{12}X_{i1} + \beta_{22}X_{i2} + \cdots + \beta_{M2}X_{iM} \\
&\cdots \\
\text{logit}(P(Y_i = K)|X) &= \beta_{0K} + \beta_{1K}X_{i1} + \beta_{2K}X_{i2} + \cdots + \beta_{MK}X_{iM}
\end{aligned} \tag{2.2.3}$$

where $P(Y_i = k|X)$ is the probability of instance i belonging to category k, logit $(\cdot)$ is the log-odds function, $\beta_{0k}, \beta_{1k}, \ldots, \beta_{Mk}$ are the coefficients for category k, $X_{i1}, X_{i2}, \ldots, X_{iM}$ are the predictor variables for instance i.

2.3 Machine Learning Techniques

The conventional techniques are generally introduced by [Donders et al., 2006], which may include mean or median imputation and linear regression, which can be time-consuming and less efficient, especially when dealing with large datasets. Moreover, these methods may need to be revised to accurately capture the underlying patterns and complexities in the data, leading to suboptimal imputation results. In response to these challenges, [Lakshminarayan et al., 1996] first propose the adoption of advanced machine learning techniques, specifically random forest and decision tree algorithms, for data imputation tasks. By leveraging the power of these algorithms, we aim to enhance both the efficiency and accuracy of the imputation process.

Machine Learning is a transformative paradigm in data science. Machine learning algorithms autonomously learn patterns and make informed predictions, allowing for a dynamic and adaptive modeling process. In this section, we utilize the random forests and decision trees, two machine learning techniques, to address missing values in our insurance data. The essence of machine learning lies in its ability to uncover the complicated relationships within the data, enabling more accurate predictions and ushering in a new era of predictive modeling.

2.3.1 Decision Tree

Decision tree modeling introduced by [Breiman, 1984] is a fundamental and versatile non-parametric predictive modeling technique that is widely employed in various academic disciplines and practical domains. Rooted in the fields of machine learning and statistics, a decision tree is a hierarchical structure that systematically partitions input of features into subsets based on the values of outcome features, ultimately leading to the prediction or classification of an outcome variable [Song and Ying, 2015]. This approach embodies a top-down recursive methodology, where each level of the tree corresponds to a decision or splitting point based on specific feature conditions, resulting in a tree-like structure developed by [Zimmerman et al., 1971] that embodies a sequence of logical choices.

Decision trees are valued for their interpretability, adaptability to different data types, and capacity to handle both classification and regression tasks [Pathak et al., 2018]. The structure of a decision tree mirrors a flowchart-like representation of decision-making processes, enabling researchers and analysts to comprehend the sequential steps leading to an outcome. This interpretability is pivotal in domains where understanding the reasoning behind predictions is essential, such as medicine, finance, and ecology.

One of the notable advantages of decision trees is their innate ability to handle non-linear relationships and interactions among features without requiring explicit feature engineering. Furthermore, decision trees can be ensembled to form more sophisticated models, such as random forests and gradient boosting, which enhance predictive accuracy and robustness by aggregating the predictions of multiple decision trees.

Decision trees have broad applications across various domains, including medicine and ecology. In the field of actuarial science, decision trees play a crucial role in tasks such as

developing pricing models and detecting fraudulent activities [Sahin et al., 2013]. Researchers often study decision tree algorithms to understand their properties, including bias-variance trade-offs, scalability, and interpretability. Moreover, decision trees provide a foundation for advanced topics like ensemble methods, tree pruning techniques, and handling imbalanced datasets.

Consider a dataset $\mathcal{D}$ with features $\mathbf{X}$ and corresponding labels $\mathbf{Y}$. The decision tree algorithm recursively splits the dataset into subsets $\mathcal{D}_j$ based on feature X_i and a threshold t_i, which represents the value of a feature at which a split occurs. When constructing a decision tree, the algorithm evaluates different values of features and thresholds to determine the best way to split the data into subsets. The threshold is the point at which the decision is made regarding which branch to follow in the tree. For a classification task, the decision is based on majority voting, and for a regression task, it is based on the mean of the target values. The decision tree algorithm involves recursively binary partitioning the dataset based on the values of features to create a tree that maximizes information gain (for classification) or variance reduction (for regression) in the outcome.

In classification problem, [Su and Zhang, 2006] found the probability of different classes by conditional probability and law of total probabilities.

$$\Pr(c_i|x_p, x) = \frac{\Pr(c_i|x_p)\Pr(x|x_p, c_i)}{\Pr(x|x_p)} = \frac{\Pr(c_i|x_p)\Pr(x|x_p, c_i)}{\sum_{i=1}^{|C|}\Pr(c_i|x_p)\Pr(x|x_p, c_i)}, \tag{2.3.1}$$

where x_p is the value of the set of attributes along the path from the current node to the root, c_i is the class i, x is the value of attributes.

Suppose that each candidate attribute is independent of the path attribute assignment x_p, which means $\Pr(x|x_p, c) = \Pr(x|c)$, then we have

$$\Pr(c_i|x_p, x) = \frac{\Pr(c_i|x_p)P(x|c_i)}{\sum_{i=1}^{|C|} \Pr(c_i|x_p) \Pr(x|c_i)}.$$

We also build a regression decision tree $tree_j$ using the sampled data:

$$\hat{y}_n(x, \Theta_j, \mathcal{D}_n) = \sum_{i \in \mathcal{D}_n(\Theta_j)} \frac{\mathbf{1}_{\mathbf{x_i} \in \mathbf{C_n}(\mathbf{x}, \mathbf{\Theta_j}, \mathcal{D}_\mathbf{n})} Y_i}{P_n(x, \Theta_j, \mathcal{D}_n)}, \tag{2.3.2}$$

where $\mathcal{D}_n$ is the training sample $\{(X_1, Y_1), (X_2, Y_2), \ldots, (X_n, Y_n)\}$,

$\mathcal{D}_n(\Theta_j)$ is the random selected data points before the tree construction,

$C_n(x, \Theta_j, \mathcal{D}_n)$ is the partition containing $\mathbf{x}$,

$P_n(x, \Theta_j, \mathcal{D}_n)$ is the number of points that fall into $C_n(x, \Theta_j, \mathcal{D}_n)$.

Information gain is a metric used to measure the effectiveness of a particular attribute in classifying the data. Information Gain helps decide the order in which attributes are chosen to split the data at each node of the tree. The decision tree algorithm selects the attribute that maximizes information gain at each step. Variance reduction, in the context of decision trees and regression tasks, refers to a measure used to assess the quality of a split in the dataset during the construction of the decision tree. The goal is to find splits that lead to a reduction in the variance of the target variable within each subset. This reduction in variance helps to create homogeneous subsets, making predictions more accurate.

In the context of decision trees for classification problems, the calculation of entropy is a fundamental concept used to quantify the impurity or disorder within a dataset. Entropy serves as a criterion for assessing the homogeneity of class labels in a partition of the data resulting from a potential split. The lower the entropy values indicate higher homogeneity, making them desirable for constructing effective decision trees.

The entropy of a dataset $\mathcal{D}$ is calculated using the formula:

$$H(\mathcal{D}) = -\sum_{c=1}^{C} p_c \log_2(p_c),$$

where C is the number of distinct class labels in the dataset, p_c is the proportion of instances in $\mathcal{D}$ belonging to class c.

$$IG(X_i, t_i) = H(\mathcal{D}) - \sum_{j=1}^{k} \frac{|\mathcal{D}_j|}{|\mathcal{D}|} H(\mathcal{D}_j),$$

where $H(\mathcal{D})$ is the entropy of the dataset, and k is the number of subsets after the split, information gain IG is used to measure the performance of the split way.

For regression, the variance reduction VR is used:

$$VR(X_i, t_i) = \text{Var}(\mathcal{D}) - \sum_{j=1}^{k} \frac{|\mathcal{D}_j|}{|\mathcal{D}|} \text{Var}(\mathcal{D}_j).$$

Cross validation is a crucial technique in machine learning and statistics to deal with model performance assessment, mitigating overfitting, and hyperparameter tuning presented by [Kohavi et al., 1995]. K-fold cross-validation method introduced by [Rushing et al., 2015] is employed in this study to assess the model performance.

Assume there is a dataset $\mathcal{D}$ of size n and we want to perform k-fold cross-validation for a decision tree. The Algorithm 1 introduces how to do a K-fold cross-validation for decision

tree. Algorithm 1 shows the steps of finding the accuracy of the performance.

Algorithm 1: K-Fold Cross Validation

Result: Mean Accuracy $= \frac{1}{k}\sum_{i=1}^{k} \text{Acc}_i$

Split the Data into k Folds:

$$\mathcal{D} = \mathcal{D}_1 \cup \mathcal{D}_2 \cup \ldots \cup \mathcal{D}_k$$

while $i = 1, 2, \ldots, k$ **do**

Training Set: $\mathcal{D}_{\text{train}}^{(i)} = \mathcal{D} \setminus \mathcal{D}_i$, the dataset excluding the i-th fold.

Testing Set: $\mathcal{D}_{\text{test}}^{(i)} = \mathcal{D}_i$, the i-th fold.

Train the Decision Tree: Train a decision tree model on $\mathcal{D}_{\text{train}}^{(i)}$.

Evaluate the Model: Evaluate the decision tree on $\mathcal{D}_{\text{test}}^{(i)}$ to obtain performance metrics.

$$\text{Acc}_i = \frac{\text{CP}_i}{\text{NP}_i} \times 100,$$

where:

Acc_i is the accuracy of $\mathcal{D}_{\text{test}}^{(i)}$,

CP_i is the number of correct predictions of $\mathcal{D}_{\text{test}}^{(i)}$,

NP_i is the total number of predictions of $\mathcal{D}_{\text{test}}^{(i)}$.

end

Despite its utility, decision trees are susceptible to overfitting, wherein the model may excessively adapt to the training data, resulting in diminished generalization performance on unseen instances [Kotsiantis, 2013]. To address this limitation, the random forest algorithm emerges as a potent ensemble learning technique that leverages the strength of multiple decision trees. By constructing a multitude of trees and aggregating their predictions through

a voting mechanism, random forests mitigate overfitting tendencies and enhance predictive accuracy. This ensemble approach introduces an element of robustness, as the diversity among individual trees helps capture complex relationships within the data [Ali et al., 2012]. Consequently, the transition from decision trees to random forests becomes imperative in scenarios where improved generalization, resilience against noise, and heightened predictive performance are desired, substantiating the rationale for incorporating random forests despite the presence of decision trees.

2.3.2 Random Forest

Random forest, proposed by [Breiman, 2001], is a powerful ensemble learning method that has gained widespread popularity in machine learning for its robustness and ability to handle complex tasks. Random forest is an extension of decision tree algorithms. The key innovation lies in its ensemble nature, combining the predictions of multiple decision trees to enhance overall predictive accuracy and generalization.

A random forest consists of a collection of decision trees, each trained on a bootstrap sample of the training data and incorporating randomness in feature selection and data sampling. The law of large numbers and the decorrelation of trees in the ensemble mitigates the risk of overfitting, a common challenge in individual decision trees. During training, each tree is constructed by considering a random subset of features at each split point, promoting diversity among the trees.

The strength of random forest lies in its ability to aggregate the predictions of diverse trees through a process known as bagging. The bootstrap sampling method discussed in Section 2.1 is an essential component of the bagging ensemble technique. Bagging is also known

as bootstrap aggregating, introduced by [Breiman, 1996]. Bagging is an ensemble learning method that leverages the bootstrap technique. In machine learning, bagging involves creating multiple subsets of the original dataset by applying bootstrap resampling. Then, a base model is trained on each subset independently. The final prediction is made by aggregating the base models' predictions. This results in a robust model that excels in handling noise and capturing intricate relationships within the data. Furthermore, the random forest provides a natural measure of feature importance, aiding in identifying key variables contributing to predictive performance. Its versatility spans various domains, from classification and regression tasks to handling missing data. As a widely employed machine learning algorithm, random forest continues to demonstrate its effectiveness in addressing real-world challenges and has become a staple tool for practitioners and researchers alike.

Our research systematically compare the performance of regression imputation (logistic regression and generalized linear model) against machine learning imputation. We evaluate the imputation accuracy and efficiency in each scenario, considering the specific nature of the data and the distribution of missing values.

By conducting this comparative analysis, we aim to provide insights into the strengths and weaknesses of each imputation method in handling missing values for different types of variables. This research contributes to the literature on imputation techniques, offering valuable guidance to researchers and practitioners in choosing the most suitable approach for their specific missing data challenges.

Building decision trees using sample data, as outlined in equations 2.3.1 or 2.3.2, represents the initial phase in constructing a random forest model. Random forests offer versatility, robustness to outliers, and scalability, making them a valuable choice in scenarios where traditional models may fall short.

For classification problem, the mode (most frequent class) of the predictions from individual decision trees is used as the final output for the ensemble. This mode aggregation is based on the principle of majority voting developed by [Paul et al., 2018]:

$$\hat{y}_{T,n} = \text{mode}(\hat{y}_n(x, \Theta_1, \mathcal{D}_b), \hat{y}_n(x, \Theta_2, \mathcal{D}_n), \ldots, \hat{y}_n(x, \Theta_T, \mathcal{D}_n)). \tag{2.3.3}$$

To elucidate the functioning of the CART-split criterion presented by [Lewis, 2000], we initially consider the construction of a decision tree without subsampling, utilizing the entire and original dataset $\mathcal{D}_n$. Let A represent a generic partition, $p_{0,n}(A)$ and $p_{1,n}(A)$ be the empirical probability of a data point labeled 0 and 1, respectively, in cell A. Within this context, a cut in A is defined as a pair (v, z), where v is a value (dimension) selected from the set $\{1, \ldots, p\}$, and z signifies the position of the cut along the v-th coordinate, constrained within the limits of A. We denote the set of all such possible cuts in A as C_A, the classification CART-split criterion is:

$$\begin{aligned} L_{class,n}(v, z) = p_{0,n}(A)p_{1,n}(A) - \frac{N_n(A_L)}{N_n(A)} p_{0,n}(A_L)p_{1,n}(A_L) \\ - \frac{N_n(A_R)}{N_n(A)} p_{0,n}(A_R)p_{1,n}(A_R), \end{aligned}$$

where $AL = \{x \in A : x^{(v)} < z\}$, $AR = \{x \in A : x^{(v)} \geq z\}$. For each cell A, the best cut $(v_{\text{opt},n}, z_{\text{opt},n})$ is selected by maximizing $L_{class,n}(v, z)$ over C_A.

For regression problem, the mean of the predictions from individual decision trees is used as the final output for the ensemble. The averaging approach of prediction result could reduce variance and improve generalization proposed by [Breiman, 2001]:

$$\hat{y}_{T,n} = \frac{1}{T}\sum_{j=1}^{T} \hat{y}_n(x, \Theta_j, \mathcal{D}_n), \tag{2.3.4}$$

where $\hat{y}_{T,n}$ is the final prediction result of random forest with T trees, and $\hat{y}_n(x, \Theta_j, \mathcal{D}_n)$ is the prediction result of Tree j. K folder cross validation algorithm 1 could be used to validate the performance of random forest.

Let A represent a generic cell, and $N_n(A)$ denote the number of data points falling within A. $X_i = (X_{1i}, \ldots, X_{pi})$, for any $(v, z) \in C_A$, the regression CART-split criterion takes the form:

$$\begin{aligned} L_{reg,n}(v, z) = &\frac{1}{N_n(A)}\sum_{i=1}^{n}(Y_i - \bar{Y}_A)\mathbf{1}_{X_i \in A} \\ &- \frac{1}{N_n(A)}\sum_{i=1}^{n}\left(Y_i - \bar{Y}_{AL}\mathbf{1}_{X_i^{(v)} < z} - \bar{Y}_{AR}\mathbf{1}_{X_i^{(v)} \geq z}\right)^2 \mathbf{1}_{X_i \in A}, \end{aligned}$$

where $AL = \{x \in A : x^{(v)} < z\}$, $AR = \{x \in A : x^{(v)} \geq z\}$. $\bar{Y}_A$, $\bar{Y}_{AL}$, and $\bar{Y}_{AR}$) is the average of the Y_i belonging to A, AL, and AR,respectively, with the convention that the average is equal to 0 when no point X_i belongs to A, AL, and AR, respectively. For each cell A, the best cut $(j_{\text{opt},n}, z_{\text{opt},n})$ is selected by maximizing $L_{reg,n}(j, z)$ over C_A.

2.4 Evaluation metrics

The assessment of predictive model performance is a critical facet of our analytical framework. To gauge the efficacy of imputation methods, we employ the F1 score as a metric for models producing categorical outcomes, while Mean Squared Error (MSE) for numerical results. Models with larger F1 scores or lower MSE values are better models.

2.4.1 F1 Score

In a classification problem, accuracy serves as a pivotal metric to gauge the effectiveness of a machine learning model by quantifying the proportion of correctly predicted instances relative to the total number of cases in the dataset [Diebold and Mariano, 2002]. A higher accuracy percentage indicates that the model has successfully made accurate predictions across all classes, while a lower accuracy suggests less reliable compared to other models. Despite its widespread use, accuracy should be interpreted cautiously, as it does not consider class imbalances and may not adequately reflect the model's performance in scenarios with asymmetric misclassification costs. Therefore, while accuracy provides a valuable measure of overall predictive performance, it is often complemented with additional metrics such as precision, recall, and F1-score to evaluate the model's capabilities more comprehensively.

The F1 score [Davis and Goadrich, 2006] is a widely used metric in classification tasks that provides a balanced evaluation of a model's precision and recall. It is particularly valuable when the dataset is imbalanced, meaning one class dominates the other(s). The F1 score is the harmonic mean of precision and recall, ensuring that both false positives and false negatives are considered, making it suitable for scenarios where the cost of these errors varies.

Precision measures the proportion of true positive predictions (correctly predicted positive instances) out of all instances predicted as positive. Recall, on the other hand, calculates the proportion of true positive predictions out of all actual positive instances. The F1 score combines these two metrics, emphasizing the balance between precision and recall. It is especially useful when striving to achieve high accuracy for both positive and negative classes.

$$\text{F1} = 2 * \frac{\text{precision} \times \text{recall}}{\text{precision} + \text{recall}},$$

where precision $= \frac{\text{TP}}{\text{TP+FP}}$, recall $= \frac{\text{TP}}{\text{TP+FN}}$, TP is True Positive, examples correctly labeled as positive, FP is False Positive, negative examples incorrectly labeled as positive, and FN is False Negative, positive examples incorrectly labeled as negative.

The $F1$ score ranges between 0 and 1, with 1 representing perfect precision and recall. A higher F1 score implies a better balance between precision and recall, suggesting that the model is effective in both correctly identifying positive instances and minimizing false positives. It indicates a more robust performance in situations where achieving a balance between precision and recall is crucial, such as in scenarios with imbalanced class distributions or when false positives and false negatives have significant consequences.

2.4.2 Mean Squared Error

Mean Squared Error (MSE) developed by [Casella and Berger, 1990] is a commonly used metric in statistics and machine learning to measure the average squared difference between predicted and actual values. It serves as an objective measure of how well a predictive models outputs match the true outcomes or observations. The lower the MSE value, the better the model's predictions align with the actual data.

MSE is particularly useful in regression tasks, where the goal is to predict a continuous numerical value. It is also a fundamental component in many optimization algorithms, including those used for training machine learning models. Here is a closer look at MSE and its significance:

Mathematically, the MSE is calculated by summing the squared differences between predicted and actual values and then dividing by the number of data points. For a dataset with

n data points, the MSE can be expressed as:

$$\text{MSE} = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2,$$

where n is the number of data points,

y_i is the actual value of the target variable for the i^{th} data point,

$\hat{y}_i$ is the predicted value of the target variable for the i^{th} data point.

In this experiment, y_i is the actual values of the missing variables and annual claims. $\hat{y}_i$ is the predicted values of the missing variables and annual claims predicted by classical techniques and machine learning techniques.

Chapter 3

Loss Reserves

In Chapter 2, a comprehensive presentation was made, clarifying a selection of statistical tools designated for utilization in the ensuing experiments. Building upon this foundation, Chapter 3 is dedicated to explaining the methodological application of these statistical tools within the experimental framework. The primary focus of this chapter is how these tools are strategically employed in the critical task of computing loss reserves. Section 3.1 introduces the business constraints of loss reserving in Canada and the claim development of insurance. Section 3.2 introduces the run-off triangle, which is applied to the reserves calculation. In Section 3.4, two loss reserves prediction models are presented. In Section 3.5, the methods of how to deal with missing values in this thesis are proposed.

3.1 Loss Reserving

3.1.1 Motivations

Reserves in the business and legal context often refer to the practice of setting aside funds or resources for a specific purpose, for example, addressing potential losses. In the financial sector, particularly in insurance and banking, the concept of loss reserves is crucial for managing risk and ensuring financial stability. Loss reserves represent the estimated amount of money that an organization sets aside to cover anticipated future losses, liabilities, or claims presented by [Grace and Leverty, 2010].

The establishment of loss reserves is subject to various business and legal constraints to ensure transparency, accountability, and compliance with regulatory standards [Berquist and Sherman, 1977]. In Canada, robust regulatory oversight is provided by entities such as the Office of the Superintendent of Financial Institutions (OSFI), which plays a pivotal role in supervising federal financial institutions. Complementing this federal oversight, the Canadian Council of Insurance Regulators (CCIR) and the Canadian Insurance Services Regulatory Organizations (CISRO) collaborate to establish harmonized regulations for the insurance sector. These regulatory bodies mandate specific requirements for the calculation and reporting of loss reserves. These stringent constraints are rooted in objectives such as ensuring the financial stability of insurance companies and financial institutions, protecting the interests of policyholders, inspiring confidence among investors, and promoting effective risk management practices. By stipulating that companies maintain adequate reserves, the regulatory framework seeks to create a resilient and responsible financial environment, thereby upholding the integrity of the Canadian financial system. Additionally, accounting standards and principles play a significant role in governing the practices related to

loss reserves. Companies are often required to adhere to generally accepted accounting principles (GAAP) [Epstein et al., 2009] or International Financial Reporting Standards (IFRS) [Brown, 2013], which provide guidelines on the proper accounting treatment of reserves. Compliance with these standards is essential for accurate financial reporting and maintaining the trust of shareholders and the broader financial market. These constraints are designed to promote financial stability, protect stakeholders, and ensure that organizations adequately prepare for potential future losses and liabilities. Both overestimating and underestimating loss reserves present significant risks for insurance companies and other entities. Overestimating reserves can tie up capital, reduce competitiveness, and distort financial reporting, while underestimating reserves can lead to financial instability, regulatory non-compliance, and reputational damage. Striking the right balance and accurately estimating reserves is essential for maintaining financial stability, meeting regulatory requirements, remaining competitive in the marketplace, and preserving stakeholder trust.

3.1.2 Claim Development

The individual development of an insurance claim involves a series of sequential stages, each characterized by specific timelines and processes introduced by [Pigeon et al., 2013]. Payment timelines are influenced by factors such as policy terms, legal obligations, and negotiation dynamics between the insured and the insurer. Figure 3.1 is an example of calim development timeline.

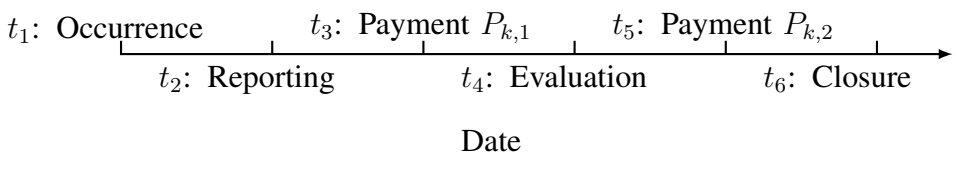


Figure 3.1: Timeline of Claim Development.

The first phase of Figure 3.1 is the occurrence of the event triggering the claim, such as an accident or loss. At the onset, the policyholder files the First Notice of Loss (FNOL) with the insurance company, serving as the initial communication of the incident. This initiates the intricate series of steps that collectively make up the claim development process. Following the occurrence at time t_1, the insured party is obligated to report the incident to the insurance company within a stipulated timeframe, known as the reporting period, which is from t_1 to t_2. This period is a critical aspect of claims management, as timely reporting allows insurers to assess the situation promptly and begin the claims processing.

Once the claim is reported at time t_2 in Figure 3.1, the insurance company initiates an investigation to evaluate the validity and extent of the claim. Upon receipt of the FNOL, the insurance company assigns a claims adjuster or examiner to investigate the claim. This skilled professional plays a pivotal role in gathering evidence, assessing damages, and determining the validity of the claim. The documentation and investigation phase involves a thorough examination of relevant information, such as police reports, medical records, and other pertinent documents. This meticulous review ensures that the insurance company has a comprehensive understanding of the circumstances surrounding the claim.

This investigation period varies depending on the complexity of the claim, the nature of the incident, and regulatory requirements. Following the investigation, the insurer determines the amount to be paid and proceeds with the payment process.

The payment random variable for the individual claim M_k is defined as $M_k = P_{k,1} + P_{k,2}$, where $P_{k,1}$ and $P_{k,2}$ are payments for the claim, made at time t_3 and t_5, in Figure 3.1 the total reported but not settled (RBNS) reserves amount is predicted as

$$\hat{R}^{RBNS} = \sum_{k=1}^{K} (\hat{M}_k - m_k),$$

where m_k is the observed total paid amount at the evaluation date for this individual claim k, $\hat{M}_k$ is the estimated total payment for individual claim k, and K is total number of reported but not settled claims.

The valuation date typically coincides with the financial reporting period of the insurance company. Constructing reserves involves a meticulous analysis of each outstanding claim's development, considering factors like the likelihood of settlement, potential legal costs, and adjustments for inflation. Actuaries play a central role in reserve construction, utilizing statistical models, historical data, and industry benchmarks to estimate future claim payments accurately. The reserve is established as the projected amount needed to cover the ultimate cost of settling all outstanding claims. Periodic adjustments are made to the reserve as new information emerges, claims develop, or external factors impact the estimates. The valuation date is a crucial reference point for constructing loss reserves, which are meticulously calculated to ensure insurers are adequately prepared to meet their financial obligations and maintain solvency. This intricate interplay of events and financial assessments underscores the complexity of the insurance claims and reserve management process.

Settlement, the final stage in the claims process at time t_6 in Figure 3.1, involves reaching an agreement between the parties involved. This can involve negotiation, mediation, or even legal proceedings. The duration of settlement negotiations is contingent on the complexity of the claim and the willingness of both parties to reach a mutually agreeable resolution. Delays in insurance claim development can result from legal proceedings, where the resolution involves court actions and legal disputes, often extending the timeline. Additionally, disagreements over liability and disputes regarding the appropriate compensation amount can further contribute to prolonged claim settlement processes. These complexities

highlight the multifaceted nature of insurance claim resolution, which may involve legal intricacies and negotiations to reach a satisfactory outcome. In the context of loss reserves, a valuation date is a pivotal point in time used to assess the financial impact of outstanding claims and estimate the required reserves presented by [Pigeon et al., 2014].

In conventional insurance practices, data is typically organized based on accident years and development years. The term "accident year," denoted as i, where $i = 1, \ldots, I$, refers to the set of claims that transpired in the i-th year subsequent to τ, a common arbitrary starting point applied uniformly to all claims. For a specific claim k, any payment made during the development year j, where $j = 1, \ldots, I$, signifies a payment executed in the j-th year following the initial occurrence at t_0. This payment is denoted as $P_{i,m}$, and it satisfies the condition $\{j - 1 < t_m - t_1 < j\}$. This formulation applies to development years $j = 1, \ldots, I$.

$$P_i^j = \sum_{m \in S_i^j} P_{i,m},$$

where $S_i^j = \{m : j - 1 < t_m - t_1 < j\}$. For a single accident year, the claim payments for every claim is calculated as

$$P_{ij} = \sum_{i \in K_i} P_i^j,$$

where K_i is the set of all claims in accident year i and $i, j = 1, 2, \ldots I$, the prediction of total reserves is obtained by

$$\hat{R}^{RBNS+IBNR} = \sum_{i=2}^{I} \sum_{j=I+2-i}^{I} \hat{P}_{ij}.$$

Technology is increasingly significant in streamlining and enhancing the claim development process. Advanced analytics and artificial intelligence are employed to expedite

claims handling, improve accuracy in valuation, and detect patterns that contribute to more efficient decision-making. Moreover, ethical considerations and regulatory compliance remain paramount throughout the process, ensuring that the insurance industry maintains the trust and confidence of policyholders and stakeholders. In essence, the claim development process underscores insurance companies' commitment to deliver fair, efficient, and transparent resolutions in the aftermath of covered incidents.

3.2 Run-Off Triangle

The run-off triangle introduced by [De Jong, 2006], a fundamental concept in actuarial science, serves as a graphical representation of historical insurance losses over successive periods. The loss triangle provides a structured framework for analyzing and projecting the development of insurance claims. The triangle's horizontal axis typically represents accident or policy years, while the vertical axis denotes cumulative losses. As claims mature over time, the diagonal cells of the triangle reflect the development of losses from reported incidents to ultimate payouts. Actuaries employ the loss triangle to assess the pattern and emergence of incurred losses, aiding in estimating future liabilities and reserves [De Felice and Moriconi, 2003]. The loss triangle is typically created using data accumulated over different evaluation dates, introduced in Section 3.1.2, rather than being explicitly designed on a single evaluation date. The triangle represents the historical development of insurance claims over multiple periods and accident years.

The triangular format encapsulates the temporal evolution of insurance claims, enabling practitioners to derive valuable insights into the progression and ultimate cost of risk events.

Loss triangles are a cornerstone in actuarial reserving and risk management practices, embodying a powerful tool for informed decision-making within the insurance industry. [Abdallah et al., 2015].

$$C_{i,j} = \sum_{k=1}^{j} X_{i,k} \tag{3.2.1}$$

where $C_{i,j}$ represents the cumulative losses at the intersection of the i-th accident or policy year and the j-th development period in the loss triangle. $X_{i,j}$ denotes the incremental losses reported during the i-th accident or policy year and the k-th development period. Table 3.1 is a loss triangle, where DY_j refer to development year j, AY_i refer to accident year i, $C_{i,j}$ refer to the cumulative cash flow for AY_i and DY_j. This example is supposed to contain data for three years.

Table 3.1: Loss Triangle.

	DY_1	DY_2	DY_3
AY_1	$C_{1,1}$	$C_{1,2}$	$C_{1,3}$
AY_2	$C_{2,1}$	$C_{2,2}$	
AY_3	$C_{3,1}$		

Here we propose a toy example of loss triangle with three years' data.

Table 3.2: Loss Triangle Example.

	DY_1	DY_2	DY_3
AY_1	100	150	200
AY_2	130	170	
AY_3	120		

In the provided illustration, the numerical values denote cumulative cash flows. For instance, the notation $C_{21} = 130$ signifies that the cumulative cash flow for claims during the second accident year amounts to 130. Similarly, $C_{22} = 170$ indicates that the cumulative cash flow encompasses 170 for the second accident year. This, in turn, implies that the cash flows attributed to claims during the second accident year are distributed as 130 for the first development year and 40 for the second development year ($170 - 130 = 40$). The cumulative nature of these values reflects the aggregation of cash flows over the specified accident and development periods.

3.3 Deterministic Models

3.3.1 Chain-Ladder Model

Chain ladder models are actuarial methods widely employed in insurance and risk management to estimate future claims reserves. These models are rooted in statistical techniques and are particularly valuable in situations where historical data is the primary basis for predicting future claims development. Introduced initially by [Stanard, 1985], chain ladder models have since evolved into a cornerstone of actuarial practice due to their simplicity and effectiveness. At its core, the chain ladder method utilizes historical claims data to forecast future claim payments. The approach derives its name from the sequential nature of its calculations, where estimates for each future period are derived from the known data of prior periods, forming a ”chain” of calculations.

The basic premise of the chain ladder method involves estimating development factors, which represent the ratio of claims paid in one period to those paid in a subsequent period. These development factors are applied to known historical claims data to project future

payments. The choice of development factors and the method used to estimate them can vary depending on the specific characteristics of the insurance portfolio and the underlying claims experience. Chain ladder models can be further refined through various techniques, including age-to-age factors, the Bornhuetter-Ferguson method, refer to equation 3.3.1, and other credibility adjustments to account for specific features of the data or to enhance predictive accuracy.

The basic chain ladder method estimates the development factors f_i could be calculated as equation 3.4.1.

3.3.2 Bornhutter-Fergusson

Bornhuetter-Ferguson method is a pioneering actuarial technique utilized for estimating future claims reserves and evaluating loss ratios in insurance portfolios. Introduced independently by [Bornhuetter and Ferguson, 1972], this method revolutionized actuarial practice by incorporating both historical experience and expected future development into reserve calculations.

The B-F method combines elements of the chain ladder technique and a priori estimates of ultimate loss ratios to produce more robust reserve estimates. It assumes that ultimate loss ratios are a linear combination of observed historical loss ratios and a credibility factor. By integrating this credibility weighting mechanism.

The B-F method estimates the ultimate loss ratio (R) using the formula:

$$R = (1 - w) \cdot \hat{R} + w \cdot \frac{\sum_i w_i \cdot R_i}{\sum_i w_i}, \tag{3.3.1}$$

Where R is the ultimate loss ratio, $\hat{R}$ is the a priori estimate of R, w is the credibility factor, R_i is the observed loss ratio for development year i, w_i is the credibility weight assigned to

each observed loss ratio R_i.

3.4 Stochastic Models

In the ensure section, the focus shifts to using stochastic models to estimate loss reserves. Commencing with the aggregation of annual payments into a structured loss triangle, the subsequent step involves the application of the Mack model introduced by [Mack, 1993]. This predictive modeling approach is systematically deployed to estimate reserves for both machine learning imputation datasets and regression imputation datasets. By adopting a unified approach in employing the Mack model across varied imputation methodologies, this analytical framework aims to derive robust and comparable loss reserves.

3.4.1 Mack Model

The Mack model represents a pivotal advancement in actuarial science and insurance reserving methodologies. This model addresses a fundamental challenge actuaries face: predicting future claims reserves, refining the conventional Chain-Ladder method, and offering a more sophisticated and statistically rigorous framework. One of the distinctive features of the Mack model is its integration of a stochastic element into the traditional deterministic Chain-Ladder approach. The Mack model assumes that the mean of the development factors remains constant across accident years. This implies that, on average, claims development follows a predictable pattern over time, allowing for estimating future loss reserves based on historical data. The model incorporates a distribution-free methodology that captures the observed development factors and quantifies their inherent variability developed by [Mack, 1994]. This Bayesian framework aligns with the principles of credibility theory,

thereby providing a more dynamic and insightful approach to loss reserving proposed by [Taylor, 2015]. This adaptability enhances its applicability in real-world insurance scenarios where the conventional methods might fall short. Moreover, the Mack model has influenced the evolution of actuarial practices by introducing a distribution-free standard error calculation. This provides a more accurate assessment of the uncertainty associated with reserve estimates and facilitates a more comprehensive understanding of the range of possible outcomes.

In order to calculate the loss reserves, we need to estimate the development factor for developmental year k, $\hat{f}_k$, first.

$$\hat{f}_k = \frac{\sum_{i=1}^{d+1-k} C_{i,k}}{\sum_{i=1}^{d+1-k} C_{i,k-1}}, \tag{3.4.1}$$

where $C_{i,k}$ denotes the incurred losses reported during the i-th accident or policy year and the k-th development period, which is referred to Table 3.1. d is the total number of development years. $\hat{f}_k$ is the unbiased estimator of f_k. With the example in Table 3.2, we get the development factor of development year 2, $\hat{f}_2$, is calculate as $\hat{f}_2 = \frac{\sum_{i=1}^{3+1-2} C_{i,2}}{\sum_{i=1}^{3+1-2} C_{i,2-1}} = \frac{\sum_{i=1}^{2} C_{i,2}}{\sum_{i=1}^{2} C_{i,1}} = \frac{150+170}{100+130} = 1.39$

The projected loss $\hat{C}_{i,k}$ for the development year k in the accident year i could be calculated as:

$$\hat{C}_{i,k} = \hat{f}_k * C_{i,k-1}, \tag{3.4.2}$$

In Table 3.2, the estimation of $C_{3,2}$ is $\hat{C}_{3,2} = \hat{f}_2 * C_{3,1} = 1.39 * 120 = 167$.

The estimate of standard error $\hat{\sigma}_k$ of the development factor $\hat{f}_k$ is:

$$\hat{\sigma}_k = \frac{1}{d-k} \sum_{i=1}^{d+1-k} C_{i,k-1}(\frac{C_{i,k}}{C_{i,k-1}} - \hat{f}_k)^2,$$

$$Var(\hat{f}_k) = \frac{\hat{\sigma_k}}{\sum_{i=1}^{d+1-k} \hat{C}_{i,k-1}},$$

where $\hat{\sigma}_k$ is the unbiased estimator of σ_k

The formula of standard error of total reserves is:

$$Var(\hat{C}_{k+1}) = h_k \hat{\sigma}^2_{k+1} + h_k^2 Var(\hat{f}_k)^2 + \hat{f}^2_{k+1} Var(\hat{C}_k),$$

where $h_k = \sum_{i=d-k+1}^{d} \hat{C}_{i,k}$.

The evaluation of various data imputation and claim prediction techniques is discerned through the prediction of loss reserves across distinct datasets. These datasets encompass those imputed by regression models, machine learning methodologies, and the original dataset. This comprehensive assessment brings the performance disparities among diverse techniques to light. This analytical framework elucidates the nuanced impact of different data imputation strategies and predictive models on the accuracy of loss reserve estimations, thereby providing valuable insights into these techniques' overall performance and suitability within the context of insurance risk management.

3.4.2 GLM Reserve Model

In the previous Section 3.4.1, the Mack model has been introduced to predict the loss reserves for loss triangles. In this section, we introduce GLM based reserves model proposed by [Björkwall et al., 2011].

In the context of loss reserves, GLMs typically incorporate key covariates such as policy attributes and other relevant variables that influence the frequency and severity of insurance claims. By accounting for these factors, insurers can build models that better reflect the

underlying risk and tailor their reserve estimates accordingly. GLMs also provide a mechanism for assessing the uncertainty associated with reserve estimates, aiding in risk management and decision-making processes. Moreover, GLMs offer interpretability, allowing actuaries and insurance professionals to understand the impact of different variables on the response variables [England and Verrall, 2002].

Here we introduce how GLM reserves model predict the reserves. For a random variable Y_{ij}, which is incremental aggregate payment for accident year i and development year j, the probability density function (pdf) or probability mass function (pmf) is often written in the form of exponential family in Section 2.2.1

$g(\mathbf{E}(Y_{ij})) = \beta_0 + \alpha_i + \beta_j$, where $g()$ is the link function that has been presented in Section 2.2.1, Table 2.1 showed the link functions for exponential families, β_0 is the intercept, α_i is the accident year effect, and β_j is the development year effect. Y_{ij} is predicted as

$$\hat{Y}_{ij} = g^{-1}(\beta_0 + \alpha_i + \beta_j), \tag{3.4.3}$$

The prediction of Reserves amount is calculated as

$$\hat{R} = \sum_{i=2}^{I} \sum_{J=I+2-i}^{I} \hat{Y}_{ij}, \tag{3.4.4}$$

where $i, j = 1, 2, \ldots, I$.

Recall example Table 3.2, a incremental loss triangle Yable 3.3 with all values are notated as Y_{ij} with $I = 3$, The estimated Y_{23}, Y_{32}, and Y_{33} could be calculated by equation 3.4.3, and we find the total reserves at time 3 by equation 3.4.4.

Table 3.3: Incremental Loss Triangle Example.

	DY_1	DY_2	DY_3
AY_1	100	50	50
AY_2	130	40	
AY_3	120		

The estimator of ϕ is

$$\hat{\phi} = \frac{1}{n-p} \sum_{i+j \geq t} \omega_{ij} \frac{(y_{ij} - \hat{\mu}_{ij})^2}{Var(\hat{\mu}_{ij})},$$

where $n - p$ is the degree of freedom of model, $\hat{\mu}_{ij}$ is the unbiased estimator of Y_{ij}, t is the number of years in the balance sheet.The estimates of β_0, α_i, and β_j could be found by maximum likelihood, which has been introduced in Section 2.2.1.

In this study, we intend to employ two distinct methodologies, namely the Mack model and Generalized Linear Models (GLMs), to forecast loss reserves associated with insurance claims. Subsequently, we aim to employ bootstrap resampling techniques to generate empirical distributions for the loss reserve estimates derived from both methods following the steps in Section 2.1. The difference between Mack model and GLM reserve model is that for Mack model, the input is C_{ij} in Table 3.1 for accident year i and development year j; however, for GLM reserve model, the input is $Y_{ij} = C_{ij} - C_{i(j-1)}$.

3.4.3 Risk Measures

Risk measures are quantitative metrics used to evaluate the level of risk associated with uncertain outcomes in various domains, including finance, insurance, and engineering. These measures provide insights into the potential losses or adverse consequences that may occur,

allowing decision-makers to assess and manage risk effectively. Two essential properties of risk measures are coherence and sub-additivity. A risk measure is considered coherent if it satisfies desirable properties reflecting intuitive risk management principles. One fundamental property of coherence is the law of large numbers, which states that the risk measure of a portfolio should converge to the risk measure of its individual components as the portfolio size increases. Coherent risk measures must also be robust, meaning they should not be overly sensitive to extreme outcomes or minor changes in the probability distribution. Moreover, coherent risk measures should be monotonic, meaning they should increase as the level of risk in the underlying distribution increases.

A risk measure ρ is considered coherent if it satisfies the following properties:

1. Translation Invariance: For any random variable X and constant c, $\rho(X + c) = \rho(X) + c$.

2. Monotonicity: If $X \leq Y$ almost surely, then $\rho(X) \leq \rho(Y)$.

3. Positive Homogeneity: For any random variable X and positive constant a, $\rho(aX) = a \cdot \rho(X)$.

4. Sub-additivity: For any two random variables X and Y, $\rho(X + Y) \leq \rho(X) + \rho(Y)$.

Sub-additivity is a specific property of risk measures that reflects the idea that the risk of a combined portfolio should not exceed the sum of the risks of its individual components. Tail Value at Risk (TVaR) is an example of such a measure. TVaR extends the Value at Risk (VaR) concept by considering the probability of extreme losses and the expected magnitude of losses beyond the VaR threshold. TVaR satisfies the sub-additivity property, unlike VaR, which makes it a coherent risk measure. The coherence of TVaR arises from

its ability to capture the impact of extreme losses on a portfolio's overall risk in a consistent and mathematically rigorous manner. By considering the conditional expectation of losses exceeding the VaR threshold, TVaR provides a more comprehensive and robust measure of downside risk, aligning with the principles of coherence in risk management. Therefore, while VaR may need more coherence due to its failure to satisfy sub-additivity, TVaR stands out as a coherent risk measure that offers valuable insights into the tail risk of a portfolio.

We utilize both Value at Risk (VaR) and Tail Value at Risk (TVaR) to assess the effectiveness of various loss reserving estimation methods introduced in Section 3.4.1 and 3.4.2. To find the risk measures of loss reserves, we firstly use the bootstrapping method to find the distribution of estimated loss reserves, which we have introduced in Section 2.1; it involves repeatedly sampling from observed historical data with replacement to generate multiple simulated datasets, from which reserve estimates are computed. This approach allows actuaries to obtain a distribution of possible reserve values.

Value at Risk (VaR) is a widely used risk measure in finance, particularly in risk management and investment analysis [Duffie and Pan, 1997]. It quantitatively assesses the potential loss a portfolio or investment may incur over a given time horizon within a specified level of confidence [Basak and Shapiro, 2001]. The concept of VaR is rooted in the fundamental principle of risk management, aiming to quantify the downside risk associated with an investment or portfolio. Moreover, VaR represents the maximum loss expected from an investment with a certain probability over a predefined time horizon. For instance, a 5% VaR over a one-day horizon at the 95% confidence level indicates a 5% likelihood that the portfolio incur losses exceeding the VaR within the next day with a confidence level of 95%. There are various methods for calculating VaR, the most common being the percentile method. This method uses historical data or statistical models to estimate the

probability distribution of portfolio returns.

$$\text{VaR}_\alpha(X) = \inf\{x \in \mathbb{R} : P(X \geq x) \geq 1 - \alpha\}$$

where X represents the random variable of interest, and α represents the confidence level. Tail Value at Risk (TVaR), also known as Conditional Value at Risk (CVaR) or Expected Shortfall, is a risk measure widely employed in finance to complement Value at Risk (VaR) by providing additional insights into the tail behaviour of the distribution of potential losses [Rockafellar et al., 2000]. TVaR extends beyond VaR's focus on the magnitude of potential losses and considers the expected value of losses exceeding the VaR threshold [Sarykalin et al., 2008]. As such, it offers a more comprehensive assessment of downside risk, particularly in scenarios with extreme market conditions or tail events. The concept of TVaR builds upon the foundation of VaR, which quantifies the maximum potential loss at a specific confidence level over a given time horizon. However, while VaR provides a single-point risk estimate, TVaR goes further by estimating the average magnitude of losses beyond the VaR threshold. Unlike VaR, which focuses solely on the probability distribution of portfolio returns up to the VaR threshold, TVaR considers the entire tail of the distribution.

$$\text{TVaR}_\alpha(X) = \frac{1}{1-\alpha} \int_{\text{VaR}_\alpha(X)}^{\infty} x f(x)\, dx$$

where $f(x)$ is the probability density function of X.

3.5 Dealing with Missing Data in Loss Reserving

Missing data in insurance databases can arise due to a multitude of factors, including incomplete policy information provided by policyholders, data entry errors during manual

input processes, data quality issues such as duplicate records or inconsistent formatting, policy changes and updates that may not be accurately reflected in the database, privacy and compliance concerns leading to redaction of sensitive information, and challenges during data migration or integration efforts. The frequency of missing data can vary widely depending on the size and complexity of the dataset, as well as the effectiveness of data management practices employed by insurance companies. Despite efforts to mitigate missing data through validation checks, error detection algorithms, and data imputation methods, it remains a common challenge in insurance data management, highlighting the importance of robust data governance and quality assurance measures.

In the ensuing exposition, we explain the methodological application of tools previously introduced in Chapter 2 to address the dual objectives of imputing missing data in loss reserving. In the scope of this research endeavour, our approach entails supervised learning methodologies developed by [Caruana and Niculescu-Mizil, 2006]. Consequently, it is imperative to acknowledge that the occurrence of missing variables is inherently constrained; that is to say, a scenario where all variable values are absent is not feasible within the confines of our supervised learning framework. The requisite presence of particular variables is crucial for effectively applying supervised learning algorithms, ensuring the availability of information necessary for model training and prediction tasks. This distinction elucidates the practical constraints the supervised learning paradigm imposes on the extent of missing variables within the dataset.

The initial phase of our research project involves the critical process of data imputation, addressing missing variables across three distinct types: continuous, discrete, and categorical. Each category demands a tailored approach for effective imputation. Specifically, for categorical missing values, predictive modeling techniques are enlisted, including the

application of logistic regression models (refer to Section 2.2.2), decision tree classifiers (see Section 2.3.1), and random forest classifiers (see Section 2.3.2) [Kushwah et al., 2022] [Segal, 2004]. These models are deployed to predict and impute missing values within the categorical variable domain. Conversely, a different suite of predictive models is employed in the case of continuous and discrete missing variables. Generalized linear models (refer to Section 2.2.1), decision tree regressors, and random forest regressors are leveraged for predicting and filling in the missing values. Significantly, all available predictors, excluding the variable with the missing values, are utilized as input features for these models. The evaluation metrics F1 score and MSE introduced in Section 2.4 are used to measure the performance of these models; the models with higher F1 scores or lower MSE values are considered outperformed.

3.5.1 Categorical Missing Value

The initial phase of our research project involves the critical process of employing two distinct yet powerful machine learning classification methodologies: a decision tree classifier and a random forest classifier, each thoughtfully configured to address the complexity of our dataset and research objectives. For the decision tree classifier, we have utilized information gain as the criterion for splitting nodes; all other hyperparameters can be found in Appendix A. The probability of different classes could be seen by equation 2.3.1, where c_i means different classes of the categorical missing values and x refers to the value of all variables used to predict the missing values. This strategic configuration aims to balance model complexity and predictive accuracy, enabling us to construct an interpretable yet practical decision tree capable of capturing intricate patterns within the data. In parallel,

we harness the ensemble learning prowess of a random forest classifier, consisting of an ensemble of 100 decision trees; the random forest is constructed using equation 2.3.3, where the training sample $\mathcal{D}_n$ refers to all predictors as $X_1, X_2, \ldots, X_n$ and the known values of categorical response variable as $Y_1, Y_2, \ldots, Y_n$. Each tree within the ensemble is designed with a maximum depth of eight levels, and we maintain a minimum number of samples per leaf node as one. This ensemble approach offers enhanced predictive robustness by aggregating the insights from numerous decision trees, each contributing its unique perspective to the classification task. These meticulously chosen configurations reflect best practices in machine learning and align with the complexity and diversity of the data at hand. By parallelizing the results of the decision tree classifier and the random forest classifier, we aim to provide a holistic understanding of the dataset's underlying patterns and complexities, furthering our research objectives and enriching the scientific discourse in our field.

We employ a multinomial logistic regression framework in the context of regression imputation. Given four distinct categories within the categorical variable, we construct four separate and independent logistic regression models, denoted as 'Model1,' 'Model2,' 'Model3,' and 'Model4.' Each model adopt a 'one vs. all' strategy developed by [Wu et al., 2006], where one specific category is treated as the positive outcome. In contrast, the remaining three categories are grouped as the composite negative outcome. Refer to equation 2.2.3, $Y_i = k$ refers to the categorical missing value belonging to the class k, $X_{i1}, X_{i2}, \ldots, X_{iM}$ refers to the corresponding predictors of missing value Y_i. For instance, within 'Model1,' we have two possible outcomes: 1 representing the positive category and 2, 3, 4 encompassing the aggregated negative categories. The coefficients derived from these logistic regression models encapsulate the effect of a one-unit change in a predictor variable on the

log odds of the positive outcome occurring, i.e., category 1. Consequently, the exponentiated coefficient, denoted as e^{β_i}, signifies the odds ratio associated with a one-unit increase in the predictor variable. This odds ratio quantifies the marginal change in the odds of the event, 1, transpiring. In practical terms, the impact of this marginal increase in odds depends on the specific context of our analysis. We assess the statistical significance of these coefficients and evaluate their practical implications, considering the nuanced relationships between predictor variables and the categories. By establishing these four distinct logistic regression models, we aim to comprehensively understand and quantify the influences of predictor variables on each category.

We employ the F1 score as our primary evaluation metric to evaluate their effectiveness, which we have introduced in Section 2.4.1. By comparing the F1 scores obtained from the Machine Learning Imputation and Regression Imputation, we gain valuable insights into the accuracy of each method; this comparison informs us about the robustness and reliability of the imputed values [Davis and Goadrich, 2006]. Ultimately, the findings aid practitioners in selecting the most suitable imputation approach when working with missing data in categorical features, ensuring more robust and reliable analyses in various domains. Moreover, memory usage is another performance criterion; memory usage refers to the amount of computer memory (RAM - Random Access Memory) that a program or script consumes during its execution [Wen et al., 2020]. Understanding and managing memory usage is essential for writing efficient and optimized code, especially for large-scale applications or when dealing with substantial datasets.

3.5.2 Discrete And Continuous Missing Value

We employ two distinct and robust machine learning classification methodologies: the decision tree regressor and the random forest regressor. For the decision tree regressor, we opt for variance reduction as the criterion for node splitting, where the training sample $\mathcal{D}_n$ in equation 2.3.2 refers to all predictors as $X_1, X_2, \ldots, X_n$ and the known values of discrete or continuous response variable as $Y_1, Y_2, \ldots, Y_n$. . Simultaneously, we harness the ensemble learning capabilities of a random forest regressor, comprising 100 decision trees, which is constructed by equation 2.3.4. Each tree in the ensemble is configured with a maximum depth of six levels for discrete variables and seven levels for continuous variables, maintaining a minimum number of samples per leaf node as one. This ensemble approach enhances predictive robustness by aggregating insights from multiple decision trees, each contributing a unique perspective to the classification task. Through parallelizing the results of the decision tree regressor and the random forest regressor, our objective is to offer a comprehensive understanding of the underlying patterns and complexities within the dataset.

In the realm of regression imputation, our chosen methodology involves the implementation of a Generalized Linear Model (GLM) with a Poisson family, where $\mathbf{Y}$ in equation 2.2.1 refer to the random vector of the missing values and $\mathbf{X}$ refers to the design matrix of all predictors. By employing the Poisson family within the GLM framework, we aim to capture and model the inherent relationships in the data, ensuring a robust and statistically sound regression imputation process. This strategic selection aligns with the nature of the data distribution and underscores our commitment to employing suitable models that address the intricacies of the regression imputation task at hand.

To gauge the efficacy of these imputation techniques, we have elected to employ a robust

and widely accepted evaluation metric—the Mean Squared Error (MSE). By meticulously comparing the MSE values derived from both the Machine Learning Imputation and Regression Imputation methodologies, we aim to derive profound insights into the accuracy, reliability, and overall performance of each method. Ultimately, it positions us to make informed decisions regarding the most suitable imputation strategy in scenarios characterized by missing data, further advancing our understanding of predictive modeling in real-world applications.

3.5.3 Loss Reserving Prediction

After finishing the missing value imputation, we aggregate the 5-year annual claims which is predicted by GLM, where $\mathbf{Y}$ in equation 2.2.1 refer to the random vector of the annual claim and $\mathbf{X}$ refers to the design matrix of all predictors and the imputed missing value. The individual annual claims are aggregated into the run off triangle developed in Section 3.2, which contains data for 5 accident years and 5 development years, where $L_{i,k}$ in equation 3.2.1 refers to the total annual claims for accident year i and development year k.

The Mack model introduced in Section 3.4.1 and GLM reserve model introduced in Section 3.4.2 are used to estimate the loss reserves using the loss triangle. The runoff triangle example 3.2 could be used here to introduce how does the reserves been estimated. For Mack model, the development factor and projected loss can be calculated as equation 3.4.1 and 3.4.2, respectively. The total loss reserves is the sum of all projected loss $\hat{C}_{i,k}$. For GLM reserve model, the individual loss $\hat{Y}_{ij}$ for accident year i and development year j can be calculated in equation 3.4.3 and the total loss reserves can be calculated in equation 3.4.4.

Chapter 4

Numerical Results

Chapter 2 provides a comprehensive exposition of various statistical tools. Chapter 3 describes the methodology for computing loss reserves. This chapter presents the empirical findings derived from an experiment using simulated data as illustrated in Section 4.1. In Section 4.2, we introduce how to deal with different kinds of missing values using methods from Chapter 2. In Section 4.3, we introduce how to predict individual annual payments and their corresponding payment indicators. Then, Section 4.3.3 presents how to use a dataset with missing values to predict individual annual payments. Section 4.4 considers two estimation methods to estimate loss reserves.

4.1 Simulated Data

In risk assessment and insurance modeling, accurately predicting individual claim history is pivotal in pricing and underwriting decisions. The paper from [Gabrielli and V. Wüthrich, 2018] presents a novel and innovative approach to address this challenge. The authors introduce a simulation machine, a sophisticated computational tool designed to model and

simulate individual claim history (or trajectory). The paper's innovative approach provides a unique opportunity to access rich and realistic data that closely mirrors real-world insurance scenarios.

Our research uses simulated datasets to train and rigorously test our models. By using this dataset, we aim to develop and fine-tune our predictive model, enhancing its capacity to make accurate predictions related to insurance claims. The advantages of employing this simulated data are manifold. Firstly, it enables us to assess our model's performance under controlled and diverse conditions, encompassing various insurance contexts and scenarios. Secondly, it provides a benchmark against which we can compare the model's predictive capabilities, ensuring that it aligns with the complexities of the insurance landscape. Furthermore, utilizing this dataset allows us to contribute to the broader research community by evaluating the model's performance on a dataset that closely mimics the challenges faced by insurance practitioners. We explore the model's ability to accurately predict individual claim histories, assess risk, and aid in pricing and underwriting decisions.

Table 4.1 presents some features of a simulated dataset.

Table 4.1: Variables of simulated data.

Variable	Data type	Scale
Line of business	categorical	$\{1, 2, 3, 4\}$
Claim code	categorical	$\{1, 2, 3, 4, 5\}$ (with possible gap)
Accident year	discrete	$\{1994, 1995, \ldots, 2005\}$
Accident quarter	categorical	$\{1, 2, 3, 4\}$
Age	continuous	$[15, 70]$
Injury part	categorical	$\{1, 2, \ldots, 99\}$ (with possible gap)
Reporting delay	discrete	$\{0, 1, \ldots, 11\}$
Annual claim amount	continuous	$[0, \infty)$
Annual payment indicator	categorical	$\{0, 1\}$

This essay addresses missing values within three distinct categories: accident quarter, accident year, and age. Each of these categories represents a specific type of missing value, with accident quarter as an example of a categorical missing value, accident year as an example of a discrete missing value, and age as an example of a continuous missing value. Furthermore, the accident quarter provides insights into temporal patterns and seasonality, helping with resource allocation and strategic planning. Analyzing accidents over multiple years through the accident year variable enables insurers to discern long-term trends, influencing policy adjustments and pricing strategies. Age, a critical factor in driver risk profiling, helps insurers set premiums and implement targeted safety interventions. These variables contribute to a comprehensive understanding of risk factors, allowing insurers to make informed decisions and optimize their insurance portfolios for profitability and risk management.

Moreover, our analysis encompasses claim amounts for each development year from year 1 to year 5. These claim amounts serve as individual annual payments, forming the foundation for constructing the aggregate data table. It plays a pivotal role in our subsequent analysis, enabling us to gain insights into cumulative payment trends over time. Our dataset has $64{,}556$ rows: we select the first $53{,}799$ rows as the training set and the rest of the data as the testing set.

4.2 Missing Value Prediction

This numerical example uses a multifaceted approach to address missing value imputation. Our methodological framework encompasses two kinds of imputation methods, classical (regression model) and machine learning approaches (decision tree and random forest), to seamlessly handle three distinct types of missing values: categorical, discrete, and continuous. As part of our study, we add the traditional "mean imputation" approach [Donders et al., 2006]. This inclusion enables us to evaluate and contrast these different techniques' performance rigorously. This comprehensive experimental design positions us strategically to assess the effectiveness and efficiency of each method in the nuanced context of missing values imputation across diverse data types. Such a comprehensive analysis aligns with best practices in the field, as it allows us to explore the spectrum of imputation strategies and their implications for predictive modeling in the domain of our research [Donders et al., 2006].

4.2.1 Categorical Variable

In the Experiment 1 of our study, we choose the "accident quarter" variable and aim to predict this missing variable. We compare all 4 models using the accuracy measure and the F1 score defined in Chapter 2. For machine learning techniques, we find optimal values for hyperparameters over a grid search, the parameters could be found in Appendix Listing A.1. Based on the simulated dataset, Table 4.2 presents results on the testing set.

Table 4.2: Data imputation result for Experiment 1.

Model	Accuracy	F1 Score	Memory Usage
Regression Model	0.30	0.27	5.9 MB
Decision Tree	0.30	0.30	3.4 MB
Random Forest	0.31	0.32	4.8 MB
Mean Imputation	0.25	0.24	0.6 MB

Based on Table 4.2, it becomes evident that the regression model exhibits the highest memory utilization, which is 1.7 times the memory usage of the decision tree and 1.2 times the memory usage of the random forest. Moreover, the regression model registers the lowest F1 score and accuracy among the models under consideration, which only achieved 87% F1 score for the decision tree and random forest. When assessing the performance of "decision tree" and "random forest" models, a notable pattern emerges: both models demonstrate comparable levels of accuracy. However, it is essential to delve deeper into their performance metrics to gain a more comprehensive understanding. While accuracy provides an overarching measure of correct predictions, the F1 score offers a more nuanced perspective, especially in scenarios where dataset balance is critical. In this context, the "random

forest" model distinctly outshines the "decision tree" model, exhibiting a higher F1 score of 0.02. Although the F1 scores for random forest and decision tree are similar, upon closer examination, it is evident that the F1 score distribution across the four categories is more balanced for random forest since the F1 score for the third category is only 0.16 for decision tree. Still, the F1 scores for all categories are all greater than 0.25. This balance across categories indicates that random forest performs consistently across all classes, whereas decision trees may excel in some categories while underperforming in others. Therefore, despite the similarity in overall F1 scores, the more balanced performance across categories suggests that random forest is more suitable for our application. The F1 score illuminates the "random forest" model's prowess in delivering more robust and reliable predictions in such challenging contexts. Furthermore, the regression and machine learning models demonstrated superior performance compared to mean imputation. These models effectively predict outcomes, showcasing their efficacy in handling the data and making more accurate predictions than the simple mean imputation method.

4.2.2 Discrete Variable

In the second experiment of our study, we consider all 4 models to predict a missing numerical discrete variable, namely, the "accident year." To gauge the impact of these imputation techniques, we consider a widely accepted evaluation metric: the Mean Squared Error (MSE), as defined in Chapter 2. Again, we find optimal values for hyperparameters over a grid search, the parameters could be found in Appendix Listing A.2. Based on our dataset, Table 4.3 presents results of the testing set.

Table 4.3: Data imputation result for Experiment 2.

Model	MSE	Memory Usage
Regression Model	10.50	19.8 MB
Decision Tree	10.45	1.9 MB
Random Forest	10.40	3.1 MB
Mean Imputation	11.00	0.5 MB

Drawing upon the results from Table 4.3, a parallel conclusion emerges, echoing the observations made in Experiment 1. Specifically, the performance characteristics of the regression model persist in this context, notably its pronounced memory consumption, which is 10 times the memory usage of the decision tree and 6 times the memory usage of the random forest, concurrent with the highest Mean Squared Error (MSE) value among the models considered, which is 0.5% higher than the MSE value for decision tree and random forest. Although the MSE values are similar for these three models, the regression model spends too much memory to achieve a similar MSE value. Conversely, the "Random Forest" model exhibits a distinct profile: a lower MSE value than the "Decision Tree" model. However, this achievement in predictive precision is offset by relatively higher memory utilization, as observed in our results. This consistent trend underscores a pivotal trade-off in our analysis. While random forest demonstrates enhanced predictive accuracy, it concurrently necessitates a moderate increase in memory resources, distinguishing it from the economic memory demands of the decision tree model. These empirical observations offer valuable insights into the interplay between computational efficiency and predictive prowess, thereby contributing to a comprehensive understanding of the model selection and resource allocation in real-world predictive modeling scenarios.

4.2.3 Continuous Variable

In the third experiment of our study, we consider all 4 models to predict a missing continuous variable, namely, the "age." Again, we use the Mean Squared Error (MSE) as our main criterion, and optimal values for hyperparameters are shown in Appendix Listing A.3. Table 4.4 presents results of the testing set.

Table 4.4: Data imputation result for Experiment 3.

Model	MSE	Memory Usage
Regression Model	170.3	27.5 MB
Decision Tree	166.7	1.9 MB
Random Forest	163.8	3.1 MB
Mean Imputation	171.0	0.59 MB

Based on the compelling findings emerging from Experiment 3, our discerning analysis reveals a nuanced portrait when addressing missing continuous values. In this context, both the "Random Forest" and "Decision Tree" models exhibit comparable performance, notably evidenced by slight differences between their Mean Squared Error (MSE) values. However, it is imperative to discern a notable divergence in resource utilization. While the "Decision Tree" model demonstrates commendable predictive accuracy, it has done so while maintaining a relatively lower memory footprint. This characteristic is pivotal, particularly in resource-constrained environments, and underscores the efficiency of the "Decision Tree" approach. Conversely, the regression model, a constant presence in our

experiments, continues to exhibit a voracious appetite for memory resources. Unfortunately, this propensity for high memory consumption is compared with the model's consistently highest MSE value, reaffirming its challenges in achieving predictive accuracy in this context.

These empirical observations further enrich our understanding of the intricate interplay between model performance, memory utilization, and predictive accuracy, shedding valuable light on the pragmatic considerations that govern the selection of imputation techniques in the realm of missing data. Based on the experimental results, it is evident that the random forest performed much better than regression models for categorical variables and continuous variables based on the F1 score and MSE, respectively. Meanwhile, random forest saved much more memory than regression models for all experiments.

4.3 Response Variable Prediction

In this section, we forecast five successive claim payments based on covariates. To initiate this prediction process, we employ payment indicators, which serve as binary flags to denote the occurrence of a claim payment within a specific year. These payment indicators are essential for our analytical framework, enabling us to model and project the timing and magnitude of claim payments over a five-year horizon. By leveraging these indicators, we aim to gain insights into the prospective cash flows associated with each claim, which are essential for financial planning and risk management.

4.3.1 Payment Indicator

In this phase of our analysis, we consider all available covariates to predict the annual payment indicator, and we use three approaches: logistic regression, random forest, and decision tree:

$$I_j = \begin{cases} 1, & \text{if there is an annual payment,} \\ 0, & \text{if there is no annual payment,} \end{cases}$$

for development period j.

We assume independence between years and obtain optimal values for hyperparameters by searching over a grid of values. Those values could be found in Appendix Listing A.4. We compare results using the accuracy measure and the F1 Score, as in the previous section. We aim to identify the most effective model for forecasting payment indicators. This process contributes to obtaining valuable insights into the dynamics of claim disbursements and enhances our capacity to make informed decisions in the realm of insurance and risk management. Based on our dataset, Table 4.5 presents results.

Table 4.5: Payment Indicators Prediction.

Model	Accuracy	F1 Score	Memory Usage
First Period			
GLM	0.79	0.76	14.0 MB
Decision Tree	0.85	0.84	4.5 MB
Random Forest	0.86	0.84	6.2 MB
Second Period			
GLM	0.69	0.66	13.3 MB
Decision Tree	0.77	0.77	5.1 MB
Random Forest	0.78	0.79	5.9 MB
Third Period			
GLM	0.96	0.94	16.6 MB
Decision Tree	0.96	0.94	4.7 MB
Random Forest	0.96	0.95	6.4 MB
Fourth Period			
GLM	0.98	0.98	18.3 MB
Decision Tree	0.98	0.98	4.9 MB
Random Forest	0.98	0.98	6.6 MB
Fifth Period			
GLM	0.99	0.99	19.9 MB
Decision Tree	0.99	0.99	5.1 MB
Random Forest	0.99	0.99	6.8 MB
Overall			
GLM	0.88	0.86	82.1 MB
Decision Tree	0.91	0.90	24.3 MB
Random Forest	0.92	0.91	31.9 MB

In evaluating the overall performance, it is discernible from Table 4.5 that the random forest algorithm outperforms the decision tree method, surpassing the logistic regression model. This observation is made concerning both accuracy metrics and memory utilization considerations. Based on the dataset, most payments are made in the first three periods. From the first three sub-tables of Table 4.5, we could see that the F1 score of random forest is higher than that of regression model for 11%, 20%, and 1%, respectively, while the memory usage of random forest is lower than that of regression model from 50% to 65%. Although the memory usage of random forest is higher than that of decision tree from 1 MB to 2 MB, random forest also has a higher accuracy or F1 score for 1 to 2 points. Moreover, the for the last three indicators, the F1 score for values equal to 1 is 0.23 for random forest, this value is 0.09 for decision tree and 0.00 for regression model, respectively. This finding indicates that regression model only predict 0, which means no payment, for the last three indicators; however, random forest and decision tree could correct predict some payments, random forest outperformed decision tree in predicting payments for the last three payment years, which had fewer payment occurrences. The enhanced performance of random forest in this context suggests its capability to handle data with limited instances more effectively, potentially due to its ability to reduce overfitting and capture more complex patterns in the data.

The findings suggest that, within the scope of this analysis, the random forest algorithm emerges as the superior choice. It demonstrates enhanced efficacy in predictive accuracy while concurrently exhibiting more efficient memory usage compared to both the decision tree and the logistic regression model.

4.3.2 Payment Prediction

In this section, we predict the severity of each payment, i.e., when $I_j = 1$, using all available covariates and the historical record of previous payments. To achieve this predictive task, we use three modeling techniques: Random Forest, Decision Tree, and Generalized Linear Model (GLM) with possion family. By incorporating covariates and historical payment information into our predictive models; we aim to unravel the complex dynamics governing claim payments. Our choice of Random Forest, Decision Tree, and Generalized Linear Model reflects a commitment to rigorously evaluating multiple modeling approaches, enabling us to identify the most effective method for predicting payments. We assume independence between years and obtain optimal values for hyperparameters by searching over a grid of values. Those values could be found in Appendix Listing A.5. Based on our dataset, Table 4.6 presents results.

Table 4.6: Payment Prediction.

Model	MSE	Memory Usage
	First Period	
GLM	2.40×10^7	303.5 MB
Decision Tree	1.82×10^7	3.4 MB
Random Forest	1.60×10^7	4.8 MB
	Second Period	
GLM	2.30×10^7	351.2 MB
Decision Tree	1.8×10^7	3.7 MB
Random Forest	1.18×10^7	5.1 MB
	Third Period	
GLM	7.4×10^6	330.0 MB
Decision Tree	6.3×10^6	3.8 MB
Random Forest	5.0×10^6	5.2 MB
	Fourth Period	
GLM	7.4×10^5	506.8 MB
Decision Tree	2.49×10^6	4.1 MB
Random Forest	4.4×10^5	5.4 MB
	Fifth Period	
GLM	2.75×10^5	506.8 MB
Decision Tree	4.21×10^5	4.1 MB
Random Forest	1.77×10^5	5.4 MB
	Overall	
GLM	1.1×10^7	1998.3 MB
Decision Tree	9×10^6	19.1 MB
Random Forest	6.7×10^6	25.9 MB

Upon scrutinizing the comprehensive performance evaluation presented in the results table, it becomes evident that the random forest algorithm exhibits superior performance when compared to the decision tree method since the MSE value of the random forest is lower than that of the decision tree from 12% to 80% for the 5 years. Notably, the decision tree method, in turn, surpasses the Generalized Linear Model for the MSE value of the first three years, which are the periods when most payments are made, from 10% to 30%. This discernment is drawn by considering both Mean Squared Error (MSE) and memory utilization metrics, where the memory usage of the decision tree and the random forest is no larger than 2% of a regression model. The implications of these results suggest that, within the confines of this analysis, the random forest algorithm stands out as the optimal choice. It showcases heightened efficacy in predictive accuracy while demonstrating more efficient memory usage compared to the decision tree and GLM methodologies.

4.3.3 Two-Step Modeling Approach: Imputation and Response Prediction

This section focuses on harnessing the predictive power of imputed missing values. Specifically, we employ these imputed values to forecast the payment indicators and the sequence of five yearly payments. To achieve this, we consider two modeling techniques: Random Forest and Regression Models. Our overarching goal is to conduct a comprehensive performance evaluation by comparing the predictions derived from actual values, those imputed using random forest and those imputed using regression models. This multifaceted assessment allows us to gauge our imputation strategies' effectiveness and the selected models' predictive accuracy. By examining the performance across these different scenarios, we gain valuable insights into the robustness and reliability of our predictive framework.

We use "accident quarter" as the missing value in our example. Our simulated dataset has no missing values: a notably pristine dataset compared to those we may have in practice. Consequently, we randomly "forget" 20% of the values for the "accident quarter" covariate. In this section, our primary endeavor is twofold: first, we consider the random forest algorithm to predict the payment indicators using all covariates (as in the previous subsection). It establishes a baseline for comparison under ideal conditions, i.e., where there are no missing values. Then, we use the random forest imputation technique to fill in the missing values. Subsequently, with this augmented dataset, we predict the payment indicators once more using random forest. This dual-phase analysis not only affords us insights into the model's inherent predictive capabilities under optimal conditions but also demonstrates its resilience in handling missing data through imputation strategies. Table 4.7 shows results. They can significantly impact the insurance and risk management domain, as they underscore the model's adaptability and reliability characterized by both complete and partially imputed data.

Table 4.7: F1 Score of Payment Indicators for Random Forest.

Payment Indicator	I_1	I_2	I_3	I_4	I_5
Actual values	0.79	0.84	0.95	0.98	0.99
Random Forest Imputation	0.76	0.80	0.94	0.97	0.98

In Table 4.8, our objective is to replicate the analytical steps previously undertaken in Table 4.7. However, we introduce a distinct approach for addressing missing data. Specifically, we employ Logistic Regression to impute the missing value within the "accident quarter" covariate. Subsequently, we employ Logistic Regression to predict the payment indicators. By following this methodology, we aim to evaluate the effectiveness of Logistic

Regression in handling missing data and compare it with the performance of random forest employed in Table 4.7.

Table 4.8: F1 Score of Payment Indicators for Regression.

Payment Indicator	I_1	I_2	I_3	I_4	I_5
Actual values	0.66	0.76	0.94	0.98	0.99
Regression Imputation	0.60	0.73	0.94	0.97	0.98

In Table 4.9, our analysis takes us through a series of comprehensive steps with the primary goal of evaluating and comparing predictive models' performance:

1. Evaluating MSE with Actual Values: we use a random forest model with no missing values and known payment indicators to predict the severity of each payment. This step establishes a baseline for assessing model performance when data is complete, and payment indicators are already known $\rightarrow$ "Actual values" in Table 4.9.

2. Imputation with Random Forest: we use the random forest algorithm for imputation to predict all missing "accident quarters."

3. Predicting Payment Indicators: with the missing values filled in, we predict the payment indicators individually using random forest.

4. Payment Prediction with Predicted Indicators: building on the predicted payment indicators, we employ random forest once again, this time incorporating the covariates, filled-in missing values, and the predicted payment indicators $\rightarrow$ "Random Forest Prediction" in Table 4.9.

By systematically following these steps, we gain valuable insights into the predictive capabilities of random forest models in scenarios where data completeness varies and payment indicators may be partially imputed and predicted. This comprehensive analysis informs us about the model's robustness and adaptability to different data conditions, with practical implications for risk assessment and forecasting in insurance contexts.

Table 4.9: MSE for Payment using Random Forest.

Payment	P_1	P_2	P_3	P_4	P_5
Actual values	1.60×10^7	1.18×10^7	5.0×10^6	4.4×10^5	1.77×10^5
Random Forest Pred.	2.71×10^7	3.30×10^7	1.34×10^7	2.62×10^6	8.3×10^5

In Table 4.10, we closely mirror the methodology employed in Table 4.9, but with a notable distinction: we adopt different statistical techniques for imputation and prediction. Here is a breakdown of the key steps and methodologies:

1. Evaluating MSE with Actual Values: we use a generalized linear model with no missing values and known payment indicators to predict the severity of each payment. This step establishes a baseline for assessing model performance when data is complete, and payment indicators are already known $\rightarrow$ "Actual values" in Table 4.10.

2. Imputation with Random Forest: we use the random forest algorithm for imputation to predict all missing "accident quarters."

3. Predicting Payment Indicators: with the missing values filled in, we predict the payment indicators individually using Logistic Regression.

4. Payment Prediction with Predicted Indicators: building on the predicted payment indicators, we employ a Generalized Linear Model (GLM) with Poisson family incorporating the covariates, filled-in missing values, and the predicted payment indicators → "Regression Prediction" in Table 4.10.

Through this comprehensive analysis, we aim to compare the performance of regression models and random forests concerning imputation and prediction tasks. This comparative assessment provides valuable insights into the strengths and weaknesses of different statistical techniques when dealing with missing data and payment forecasting in insurance-related scenarios.

Table 4.10: MSE for Payment using Regression.

Payment	P_1	P_2	P_3	P_4	P_5
Actual values	2.40×10^7	2.3×10^7	7.4×10^6	7.4×10^5	2.75×10^5
Regression Pred.	2.75×10^7	3.33×10^7	1.35×10^7	3.21×10^6	8.76×10^5

4.4 Loss Reserves Estimation

In the property and casualty insurance domain, the accurate prediction of loss reserves is paramount to upholding insurance companies' financial viability and profitability. The process of loss reserving, which involves estimating the monetary provisions required to meet impending claims and obligations, serves as the linchpin of prudent financial planning and risk management within the insurance sector. Within the scope of our research project, our principal objective is to predict loss reserves. We consider the time-tested Mack model (see Section 3.4.1) and some generalized linear models for loss reserving (see Section 3.4.2).

These methods, well-known in the actuarial industry, are employed to project forthcoming claim payments.

As a starting point, we present in Table 4.11 the run-off triangle constructed using our simulated dataset and the end of Dev Year 5 is used as the valuation date, where Dev Year means the development year. Using the original dataset, we calculate the actual outstanding payment, i.e., the ”true reserve.” Then, we estimate the outstanding reserving using Mack model.

Table 4.11: Run-off Triangle of Simulated Dataset.

	Dev Year 1	Dev Year 2	Dev Year 3	Dev Year 4	Dev Year 5
Accident Year 1	3 307	5 028	5 340	5 461	5 556
Accident Year 2	3 518	5 333	5 704	5 896	
Accident Year 3	3 155	4 786	5 096		
Accident Year 4	3 723	5 736			
Accident Year 5	3 417				

In the second step, we construct two new run-off triangles using annual payments predicted by random forest and annual payments predicted by a generalized linear model. Those two triangles are presented in Table 4.13 and Table 4.12, respectively.

Table 4.12: Run-off Triangle of Dataset with Regression Imputation.

	Dev Year 1	Dev Year 2	Dev Year 3	Dev Year 4	Dev Year 5
Accident Year 1	3 350	4 999	5 258	5 383	5 479
Accident Year 2	3 354	5 081	5 448	5 631	
Accident Year 3	3 074	4 689	4 960		
Accident Year 4	3 620	5 420			
Accident Year 5	3 319				

Table 4.13: Run-off Triangle of Dataset with Random Forest Imputation.

	Dev Year 1	Dev Year 2	Dev Year 3	Dev Year 4	Dev Year 5
Accident Year 1	3 345	4 843	5 166	5 315	5 416
Accident Year 2	3 501	5 029	5 414	5 602	
Accident Year 3	3 291	4 810	5 147		
Accident Year 4	3 645	5 459			
Accident Year 5	3 476				

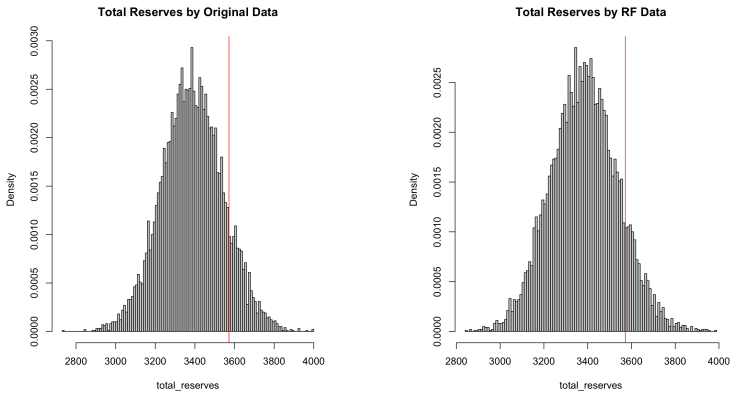
We then use Mack model to calculate loss reserves. Although this situation is not realistic, it allows us to assess better the impact of the imputation method on the reserve amount. A future analysis using much more sophisticated models, i.e., granular models could be made from these results [Zhu et al., 2018]. Table 4.14 presents results (expected values).

Table 4.14: Reserves using Mack Model.

	True Reserves	GLM Imputation	RF Imputation	Original
1	0	0	0	0
2	124	100	106	103
3	217	234	265	236
4	1 043	600	689	659
5	2 245	2 236	2 245	2 389
Total	3 629	3 170	3 305	3 387

In Table 4.14, row 1 to row 5 contain annual mean values for the loss reserves, and row 6 contains the total reserves amount for the first 5 years. Based on those results, it is evident that the values of RF reserves exhibit a closer proximity to the Original dataset than those generated by the GLM. The RF's ability to capture the underlying patterns and complexities in the data may contribute to its closer alignment with the Original-Mack, offering more reliable estimates. Conversely, the GLM's performance may be hindered by its reliance on simplifying assumptions or limitations in capturing nonlinear relationships within the data. In practice, insurers are mainly interested in the predictive distribution of the reserve (for example, to make capital allocation) and, in particular, in the high quantiles of this distribution. To study those distributions, we use bootstrap resampling techniques. The primary objective is to characterize the uncertainty in the reserve estimates and explore how well the generated data aligns with the actual reserves. Subsequently, we visualize these distributions through plots, incorporating a red vertical line to represent the true reserves. In Figure 4.15, the observation of the red vertical line, representing the true reserve, being closer to the mode of the distribution of reserves for both the original data and the data

generated from random forest imputation. This proximity suggests that the data obtained through random forest imputation more closely resembles the actual data compared to that obtained through regression imputation. The closeness of the true reserve to the distribution mode implies a higher accuracy and fidelity in capturing the underlying patterns and characteristics of the original data. Random forest imputation likely preserved the key features and variations in the true data, resulting in a distribution of reserves that closely aligns with the actual distribution. Conversely, the distance between the true reserve and the mode of the distribution for the data from regression imputation indicates that this method may have introduced more variability or distortion in the dataset. This discrepancy suggests that regression imputation may not have adequately captured the nuances and complexities in the original data, leading to a less accurate representation of the true reserves. The observation underscores the importance of employing robust imputation techniques, such as random forest, to enhance the accuracy and reliability of reserve estimations in insurance risk management.



(a) Bootstrap for Original Data

(b) Bootstrap for Random Forest

(c) Bootstrap for Generalized Linear Model

Figure 4.1: Bootstrap Resampling For all Simulation Data from Mack Model

In financial risk management, calculating Value-at-Risk (VaR) and Tail-Value-at-Risk (TVaR) for loss reserves is a critical endeavor aimed at quantifying the potential downside risk associated with these reserves. Table 4.15 presents level 90% and 95% VaR and TVaR for our three approaches.

Table 4.15: Risk Measures for Mack Reserves.

Level	GLM Imputation	RF Imputation	Original
90% VaR	3 431	3 515	3 580
90% TVaR	3 537	3 597	3 658
95% VaR	3 513	3 580	3 639
95% TVaR	3 605	3 648	3 708

In our quest for a detailed risk assessment, we have delved into the 95% VaR and TVaR metrics for individual lines of business. A key observation from Table 4.16 reveals an interesting dynamic: the sum of 95% VaR values for individual business lines is less than the 95% VaR of total reserves. This insight suggests a nuanced risk landscape. While total reserves pose a certain level of risk at the 95% confidence level, the distribution of risk across individual business lines seems to offer a form of risk diversification. The cumulative 95% VaR for individual lines of business implies a potential risk-mitigating effect, indicating that the combined risk exposure is less pronounced than total reserves alone. Examining individual business lines allows for identifying concentration or diversification areas, enabling tailored risk mitigation strategies. This approach accurately portrays the organization's overall risk profile, offering actionable insights to enhance risk management effectiveness.

The results presented in Table 4.15 highlight a consistent trend wherein the random forest

model consistently outperforms other models in terms of both the 90% and 95% risk capital measures. The comparison of the 95% confidence intervals for Original, RF Imputation, and GLM Imputation provides valuable insights into the accuracy and reliability of reserve estimations derived from different methodologies. For the original dataset, the 95% confidence interval, which is the range from 2.5th quantile to the 97.5th quantile, $[3\,095; 3\,693]$ includes the true reserve of $3\,629$. It indicates a high probability (95%) that the true reserve falls within this interval, demonstrating the effectiveness of the original dataset in providing accurate reserve estimates. Similarly, for the RF Imputation, the 95% confidence interval $[3\,002; 3\,631]$ also includes the true reserve. It suggests that the RF Imputation produces reserve estimates consistent with the true value within the specified confidence level, further validating the reliability of this approach. However, for the GLM Imputation, the 95% confidence interval $[2\,789, 3\,583]$ does not include the true reserve. It indicates a discrepancy between the estimated reserves and the true value, potentially raising concerns about the accuracy of the GLM Imputation in this particular scenario. These findings suggest that the random forest model excels in providing estimations of risk measures, and these estimates closely align with the actual reserves. The robust performance of random forest in this context underscores its efficacy as a predictive modeling tool, especially when quantifying risk exposure and potential financial losses. It implies that the model consistently captures and predicts higher risk scenarios, making it a valuable asset in risk management and financial decision-making.

Table 4.16: Risk measures for different lines of business using Mack model

Line of Business	0	1	2	3	Sum	Total Res.	Gain
Risk Measure for GLM Imputation Reserves							
95% VaR	481	369	1 240	1 996	4 086	3 513	14%
95% TVaR	508	415	1 372	2 053	4 348	3 605	17%
Risk Measure for RF Imputation Reserves							
95% VaR	632	272	976	2 217	4 097	3 580	13%
95% TVaR	670	300	1 083	2 272	4 325	3 648	16%
Risk Measure for Original Reserves							
95% VaR	525	383	1 251	2 052	4 211	3 639	14%
95% TVaR	556	422	1 370	2 103	4 451	3 708	17%

Furthermore, an insightful examination of Table 4.16 reveals noteworthy findings regarding the aggregation of risk across different lines of business. The first four columns shows the values of 95% VaR and TVaR for different lines of business, the fifth column shows the value of sum of lines of business. Specifically, the summation of 95% VaR values pertaining to distinct lines of business exceeds the 95% VaR of total reserves. This observation substantiates the premise that the combination of diverse lines of business yields a mitigating effect on overall risk exposure and potential losses. Total reserves gained more than 13% for 95% VaR and 16% for 95% TVaR compare to the sum of different line of business. The gain for 95% TVaR is larger than 95% VaR shows that TVaR is generally more sensitive to extreme or tail events compared to VaR and we should place emphasis on capturing and managing extreme tail risks. It suggests that there are diversification benefits

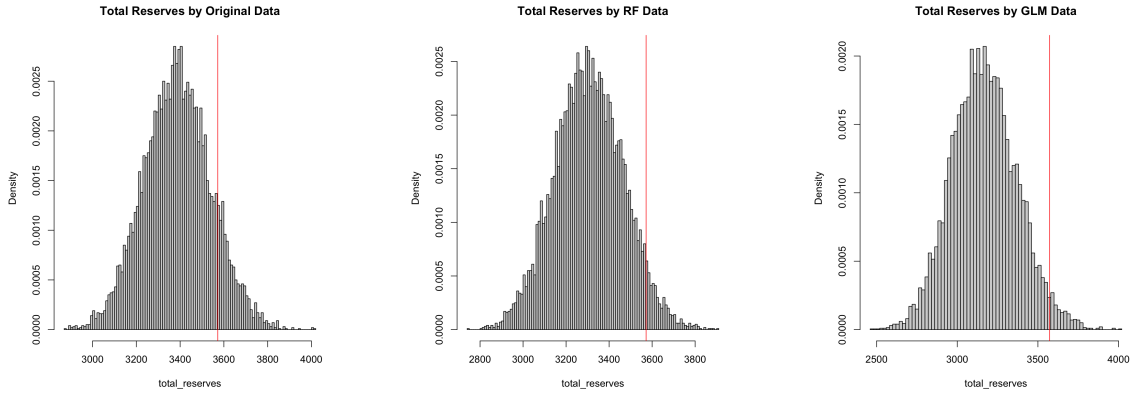
present in the portfolio. Diversification benefits occur when the risk of the portfolio as a whole is lower than the sum of the risks of its individual components. In other words, by combining different lines of business, the overall risk of the portfolio is reduced due to the imperfect correlation or negative correlation between the risks of different lines of business. This analysis underscores the importance of portfolio diversification as a risk management strategy.

In this section, we also consider a Generalized Linear Model for loss reserving. Consequently, we anticipate comparable outcomes to those yielded by the Mack model.

Table 4.17: Reserves using GLM.

	True Reserves	GLM Imputation	RF Imputation	Original
1	0	0	0	0
2	124	100	106	103
3	217	234	265	235
4	1 043	599	690	659
5	2 245	2 236	2 246	2 388
Total	3 629	3 169	3 307	3 385

Based on Table 4.17, results are similar to those in Table 4.14.



(a) Bootstrap for Original Data

(b) Bootstrap for Random Forest

(c) Bootstrap for Generalized Linear Model

Figure 4.2: Bootstrap Resampling For all Simulation Data from GLM Reserves

Based on Figure 4.2, we could see the bootstrap result of GLM reserves model is also similar to Figure 4.1

Table 4.18: Risk measures using GLM for loss reserving.

Level	GLM Imputation	RF Imputation	Original
90% VaR	3 431	3 517	3 578
90% TVaR	3 540	3 596	3 655
95% VaR	3 518	3 578	3 635
95% TVaR	3 607	3 648	3 708

The values of risk measurements 90% and 95% of VaR and TVaR for GLM reserves model have been shown in Table 4.18, which is similar results in Table 4.15: they indicate that we could obtain the same results from GLM reserves and Mack model.

Table 4.19: Risk measures for different lines of business using GLM Reserves

Line of Business	0	1	2	3	Sum	Total Res.	Gain
Risk Measure for GLM Imputation Reserves							
95% VaR	483	367	1 230	1 993	4 073	3 518	14%
95% TVaR	510	417	1 366	2 053	4 346	3 607	17%
Risk Measure for RF Imputation Reserves							
95% VaR	632	271	963	2 213	4 079	3 578	13%
95% TVaR	671	302	1 076	2 270	4 319	3 648	16%
Risk Measure for Original Reserves							
95% VaR	524	381	1 253	2 054	4 212	3 635	14%
95% TVaR	555	421	1 371	2 103	4 450	3 708	17%

In Table 4.19, the 95% VaR and TVaR for different line of business with different datasets (original dataset, GLM predicted dataset, and random forest predicted dataset) showed the similar results compared to Table 4.16.

Drawing upon the outcomes derived from both the Mack model and the GLM reserves model, it can be inferred that, given their common foundation within the GLM framework, these methodologies are likely to yield similar results in terms of reserves estimation. This similarity extends across aspects such as reserves distribution and risk measurement, thereby substantiating a consistent outcome between the two approaches. However, the variance of the total reserves of these two methods is different. The variance of the reserves with original dataset for the Mack model is 23,892, while that for the GLM Reserves is 24,580, the variance of the reserves with random forest imputation dataset for the Mack

model is $25{,}422$, while that for the GLM Reserves is $26{,}103$, the variance of the reserves with GLM imputation dataset for the Mack model is $39{,}490$, while that for the GLM Reserves is $41{,}162$. The variance of Mack model is smaller than that of GLM Reserves, it suggests that the predictions or outcomes produced by Mack model are more consistent or less variable than those of GLM Reserves. Moreover, the variance for reserves of dataset with GLM imputation is much larger that that for two other datasets, which indicate that random forest imputation dataset is closer to the original dataset, and it is outperformed than the GLMimputation.

The comprehensive analysis conducted on scenarios involving missing values in both continuous and discrete variables, with detailed results provided in the Appendix B, serves as a testament to the stability and robustness of the findings presented in Section 4.4. The inclusion of such extensive analyses not only validates the results obtained but also enhances the credibility and reliability of the conclusions drawn. By examining various scenarios and providing evidence of consistent outcomes across different data settings, the study demonstrates the resilience of the methodologies employed in estimating reserves.

Chapter 5

Conclusion

The present study investigates the efficacy of employing random forest for data imputation within the domain of loss reserving. Results indicate that random forest outperforms traditional regression techniques regarding data imputation accuracy, as evidenced by achieving higher F1 scores for categorical missing data or lower Mean Squared Error (MSE) for discrete and continuous missing data while concurrently exhibiting markedly reduced memory usage. Even though the random forest algorithm handles missing values in the input, it has less information and is less efficiency than the imputed dataset when we predict annual claims and loss reserves. The random forest also gains more information from the imputed data.

Following the completion of the data imputation phase, the study utilized both random forest and regression models to forecast annual payment indicators and payments over five consecutive accident years. Random forest exhibited superior performance in predictive accuracy, achieving higher F1 scores for payment indicators prediction and lower Mean Squared Error (MSE) for payment prediction compared to the regression model. This suggests that random forest is adept at imputing missing data and also excels in predicting

payment indicators and annual individual payments, thereby underscoring its utility and efficacy within the domain of loss reserving.

After the payment prediction, we aggregate the individual payments into the run-off triangles. The original dataset and datasets subjected to regression imputation and random forest imputation were utilized to estimate loss reserves using the Mack and Generalized Linear Model (GLM) reserves models. Notably, the dataset employing random forest imputation closely approximated the performance of the complete simulated dataset in contrast to the dataset employing regression imputation. Moreover, employing a bootstrap methodology to estimate the distribution of loss reserves across different datasets revealed that the true reserves lie within the 95% confidence interval of the distribution of loss reserves derived from both the original dataset and the dataset employing random forest imputation. Meanwhile, the 95% VaR and TVaR of the dataset employing random forest imputation is much closer to that of the complete simulated dataset than the regression imputation dataset. Moreover, the results of risk measures on the entire portfolio and different lines of businesses suggest that we could receive risk diversification benefits from aggregating individual lines of businesses into the portfolio, which is the property of sub-additivity holds.

This observation underscores the superiority of random forest as a data imputation method compared to traditional regression techniques in the context of loss reserves, illuminating its robustness in handling complex datasets characterized by missing values. Meanwhile, regression models are struggling to capture the complex relationship between variables. Using a simulated dataset in our study demands an acknowledgment of the potential disparities it may harbour compared to real-world datasets. Recognizing the divergence between simulated and actual data is essential. While the insights obtained from our analysis

offer valuable perspectives, they may not seamlessly align with what is inherent in real-world scenarios. For example, some non-randomness missing values exist in real-world data, which we do not mention in the simulated dataset. Central to our investigation were micro-level reserving prediction methodologies, examined to clarify their efficacy within the context of the simulated dataset. However, it is essential to acknowledge the limitations imposed by the simulated environment and the confined scope of our findings. Subsequent phases of our inquiry are earmarked for a transition towards authentic datasets. This strategic pivot serves a dual purpose: firstly, to transcend the constraints of simulation and access the richness of real-world data, and secondly, to enable the exploration and application of macro-level reserving techniques for predictive analytics. By delving into authentic datasets, we aim to enrich our understanding, refine our methodologies, and, ultimately, foster a more robust framework for predictive analysis in insurance and related domains.

Appendix A

Coding

	LoB	cc	AY	AQ	age	inj_part	RepDel	Pay00	Pay01	Pay02	Pay03	Pay04
7	3	3	1994	3	30	51	0	0	3133	0	0	0
27	1	3	1994	3	28	36	0	106	0	0	0	0
43	3	5	1994	2	45	60	0	2463	0	0	0	0
45	4	1	1994	3	36	36	0	593	0	0	0	0
54	3	1	1994	4	33	23	0	0	0	0	0	0
...	...	...	...	...	...	...	...	...	...	...	...	...
2001631	3	5	2005	1	90	36	0	1454	0	0	0	0
2001705	4	4	2005	4	34	44	1	0	256	0	0	0
2001759	4	4	2005	3	47	63	0	713	0	0	0	0
2001823	4	3	2005	1	32	30	0	798	0	0	0	0
2001852	2	5	2005	4	42	15	0	0	0	0	0	0

Figure A.1: Data Example.

Listing A.1: Categorical Missing Value

```
1 # Create a Decision Tree Classifier
2 tree = DecisionTreeClassifier(max_depth=15,min_samples_leaf=100)
3
```

```
# Create a Random Forest model
ram = RandomForestClassifier(n_estimators=10, random_state=1000,
    max_depth=100,min_samples_leaf=1000)


# Create and train the linear regression model
res = np.insert(xinput1, 0, 1, axis=1)
model = sm.MNLogit(yinput1, res)
result=model.fit()
```

Listing A.2: Discrete Missing Value

```
# Create a Decision Tree Regressor
tree = DecisionTreeRegressor(max_depth=1,min_samples_split=100,
    min_samples_leaf=1)

# Create a Random Forest model
ram = RandomForestRegressor(n_estimators=100,max_depth=1,
    min_samples_leaf=1)

res = np.insert(xinput1, 0, 1, axis=1)

model = sm.GLM(yinput1, res, family=sm.families.Poisson())
models=model.fit()
```

Listing A.3: Continuous Missing Value

```

```

```
2  # Create a Decision Tree Regressor
3  tree = DecisionTreeRegressor(max_depth=3,min_samples_split=10,
       min_samples_leaf=1)
4
5  # Create a Random Forest model
6  ram = RandomForestRegressor(n_estimators=100,random_state=5,
       max_depth=5,min_samples_leaf=1)
7
8  res = np.insert(xinput1, 0, 1, axis=1)
9
10 model = sm.GLM(yinput1, res, family=sm.families.Poisson())
11 models=model.fit()
```

Listing A.4: Payment Indicator

```
1
2  # Payment indicator of year 1
3  # Create a Decision Tree Classifier
4  tree1 = DecisionTreeClassifier(max_depth=5,min_samples_leaf=100)
5  # Create a Decision Tree Classifier
6  ram1 = RandomForestClassifier(n_estimators=50,max_depth=15,
       min_samples_leaf=1)
7
8  # Payment indicator of year 2
9  # Create a Decision Tree Classifier
10 tree1 = DecisionTreeClassifier(max_depth=10,min_samples_leaf=10)
11 # Create a Decision Tree Classifier
```

```
ram1 = RandomForestClassifier(n_estimators=50,max_depth=15,
   min_samples_leaf=10)

# Payment indicator of year 3
# Create a Decision Tree Classifier
tree1 = DecisionTreeClassifier(max_depth=5,min_samples_leaf=10)
# Create a Decision Tree Classifier
ram1 = RandomForestClassifier(n_estimators=50,max_depth=13,
   min_samples_leaf=10)

# Payment indicator of year 4
# Create a Decision Tree Classifier
tree1 = DecisionTreeClassifier(max_depth=12,min_samples_leaf=10)
# Create a Decision Tree Classifier
ram1 = RandomForestClassifier(n_estimators=50,max_depth=15,
   min_samples_leaf=5)

# Payment indicator of year 5
# Create a Decision Tree Classifier
tree1 = DecisionTreeClassifier(max_depth=15,min_samples_leaf=100)
# Create a Decision Tree Classifier
ram1 = RandomForestClassifier(n_estimators=50,max_depth=15,
   min_samples_leaf=10)
```

Listing A.5: Individual Annual Payments

```
# Create a Decision Tree Regressor
```

```
2  tree01 = DecisionTreeRegressor(max_depth=5,min_samples_leaf=20)
3
4  # Fit the model to the data
5  tree01.fit(xinput2, yinput2)
6
7  # Make predictions on the test data
8  y_tree01= tree01.predict(xoutput2)
9
10 # Create a Random Forest model
11 ram01 = RandomForestRegressor(n_estimators=100,random_state=7,
       max_depth=15,min_samples_leaf=40)
12
13 # Train the model
14 ram01.fit(xinput2, yinput2)
15
16 # Predict using the test data
17 y_ram01 = ram01.predict(xoutput2)
18
19
20 # Create and train the GLM model
21
22 res = np.insert(xinput2, 0, 1, axis=1)
23
24 model3 = sm.GLM(yinput2, res, family=sm.families.Poisson())
25 result01=model3.fit()
26
```

```
res1 = np.insert(xoutput2, 0, 1, axis=1)
y_pred01 = result01.predict(res1)
```

Listing A.6: Mack Model Loss Reserves

```
result3<- round((1/1000)*ddply(data2, .(AY), summarise, Pay00 =
   sum(Pay00),Pay01 = sum(Pay01),Pay02 = sum(Pay02),
                                Pay03 = sum(Pay03),Pay04 = sum(
                                   Pay04))[,2:6])[1:5,]

for (j in 2:5){result3[,j] <- result3[,j-1] + result3[,j]}


tri_dat3 <- array(NA, dim(result3))
reserves3 <- data.frame("true Res." = numeric(), "CL Res." =
   numeric(), "MSEP^(1/2)" = numeric())

reserves3 <- setNames(reserves3, c("true Res.","CL Res.","MSEP
   ^(1/2)"))
for (i in 0:4){
  for (j in 0:(4-i)){tri_dat3[i+1,j+1] <- result3[i+1,j+1]}
  reserves3[i+1,1] <- result3[i+1,5]-result3[i+1,5-i]
}

reserves3[6,1] <- sum(reserves3[1:5,1])
tri_dat3<- as.triangle(as.matrix(tri_dat3))

```

```
19 dimnames(tri_dat3)=list(origin=1:5, dev=1:5)
20
21 Mack1 <- MackChainLadder(tri_dat3,est.sigma="Mack")
22
23 for (i in 0:4){reserves3[i+1,2] <- round(Mack1$FullTriangle[i
      +1,5]-Mack1$FullTriangle[i+1,5-i])}
24 reserves3[6,2] <- sum(reserves3[1:5,2])
25 reserves3[1,3] <-0
26 reserves3[1:5,3] <- round(Mack1$Mack.S.E[,5])
27 reserves3[6,3] <- round(Mack1$Total.Mack.S.E)
```

Appendix B

Reserves Estimation

B.1 Discrete Data

Reserves of Mack model.

Table B.1: Reserves using Mack Model.

	True Reserves	GLM Imputation	RF Imputation	Original
1	0	0	0	0
2	139	101	108	103
3	178	241	255	236
4	1 061	611	664	659
5	2 194	2 212	2 301	2 389
Total	3 572	3 165	3 328	3 387

Risk measurement of Mack model.

Table B.2: Risk Measures for Mack Reserves.

Level	GLM Imputation	RF Imputation	Original
90% VaR	3 429	3 530	3 580
90% TVaR	3 535	3 604	3 658
95% VaR	3 510	3 591	3 639
95% TVaR	3 600	3 669	3 708

Reserves of GLM reserves model.

Table B.3: Reserves For GLM Reserves Model.

	True Reserves	GLM Imputation	RF Imputation	Original
1	0	0	0	0
2	139	101	108	103
3	178	236	265	235
4	1 061	600	685	659
5	2 194	2 234	2 250	2 388
Total	3 572	3 171	3 308	3 385

Risk measurement of GLM reserve model.

Table B.4: Risk Measures for GLM Reserves.

Level	GLM Imputation	RF Imputation	Original
90% VaR	3 430	3 519	3 578
90% TVaR	3 539	3 599	3 655
95% VaR	3 520	3 581	3 635
95% TVaR	3 610	3 650	3 708

B.2 Continuous Data

Reserves of Mack model.

Table B.5: Reserves using Mack Model.

	True Reserves	GLM Imputation	RF Imputation	Original
1	0	0	0	0
2	139	100	105	103
3	178	235	258	236
4	1 061	612	666	659
5	2 194	2 221	2 298	2 389
Total	3 572	3 169	3 327	3 387

Risk measurement of Mack model.

Table B.6: Risk Measures for Mack Reserves.

Level	GLM Imputation	RF Imputation	Original
90% VaR	3 430	3 518	3 580
90% TVaR	3 537	3 601	3 658
95% VaR	3 509	3 588	3 639
95% TVaR	3 605	3 658	3 708

Table B.7: Risk Measures for Mack Reserves.

Reserves of GLM reserves model.

Table B.8: Reserves For GLM Reserves Model.

	True Reserves	GLM Imputation	RF Imputation	Original
1	0	0	0	0
2	139	99	106	103
3	178	237	264	235
4	1 061	598	678	659
5	2 194	2 229	2 260	2 388
Total	3 572	3 163	3 308	3 385

Risk measurement of GLM reserve model.

Table B.9: Risk Measures for GLM Reserves.

Level	GLM Imputation	RF Imputation	Original
90% VaR	3 431	3 522	3 578
90% TVaR	3 541	3 601	3 655
95% VaR	3 522	3 587	3 635
95% TVaR	3 611	3 654	3 708

References

Anas Abdallah, Jean-Philippe Boucher, and Hélène Cossette. Modeling dependence between loss triangles with hierarchical archimedean copulas. *ASTIN Bulletin: The Journal of the IAA*, 45(3):577–599, 2015.

Jehad Ali, Rehanullah Khan, Nasir Ahmad, and Imran Maqsood. Random forests and decision trees. *International Journal of Computer Science Issues (IJCSI)*, 9(5):272, 2012.

Gokarna R Aryal and Haitham M Yousof. The exponentiated generalized-g poisson family of distributions. *Stochastics and Quality Control*, 32(1):7–23, 2017.

Suleyman Basak and Alexander Shapiro. Value-at-risk-based risk management: optimal policies and asset prices. *The review of financial studies*, 14(2):371–405, 2001.

James R Berquist and Richard E Sherman. Loss reserve adequacy testing: A comprehensive, systematic approach. In *Proceedings of the Casualty Actuarial Society*, volume 64, pages 123–184, 1977.

Susanna Björkwall, Ola Hössjer, Esbjörn Ohlsson, and Richard Verrall. A generalized linear model with smoothing effects for claims reserving. *Insurance: Mathematics and Economics*, 49(1):27–37, 2011.

Ronald L Bornhuetter and Ronald E Ferguson. The actuary and ibnr. In *Proceedings of the casualty actuarial society*, volume 59, pages 181–195, 1972.

Leo Breiman. *Classification and regression trees*. Routledge, 1984.

Leo Breiman. Bagging predictors. *Machine learning*, 24:123–140, 1996.

Leo Breiman. Random forests. *Machine learning*, 45:5–32, 2001.

Lawrence D Brown. Fundamentals of statistical exponential families: with applications in statistical decision theory. Ims, 1986.

Philip Brown. International financial reporting standards: what are the benefits? In *Financial Accounting and Equity Markets*, pages 297–313. Routledge, 2013.

Rich Caruana and Alexandru Niculescu-Mizil. An empirical comparison of supervised learning algorithms. In *Proceedings of the 23rd international conference on Machine learning*, pages 161–168, 2006.

George Casella and Roger L Berger. *Statistical inference*. Cengage Learning, 1990.

Jesse Davis and Mark Goadrich. The relationship between precision-recall and roc curves. In *Proceedings of the 23rd international conference on Machine learning*, pages 233–240, 2006.

M De Felice and F Moriconi. Risk based capital in p&c loss reserving or stressing the triangle. *Research Group on Insurance Companies and Pension Funds*, 2003.

Piet De Jong. Forecasting runoff triangles. *North American Actuarial Journal*, 10(2): 28–38, 2006.

Thomas J DiCiccio and Bradley Efron. Bootstrap confidence intervals. *Statistical science*, 11(3):189–228, 1996.

Francis X Diebold and Robert S Mariano. Comparing predictive accuracy. *Journal of Business & economic statistics*, 20(1):134–144, 2002.

A Rogier T Donders, Geert JMG Van Der Heijden, Theo Stijnen, and Karel GM Moons. A gentle introduction to imputation of missing values. *Journal of clinical epidemiology*, 59(10):1087–1091, 2006.

Norman R Draper and Harry Smith. *Applied regression analysis*, volume 326. John Wiley & Sons, 1998.

Darrell Duffie and Jun Pan. An overview of value at risk. *Journal of derivatives*, 4(3):7–49, 1997.

Bradley Efron. Bootstrap methods: another look at the jackknife. *The Annals of Statistics*, 7(1):1–26, 1979.

Bradley Efron. Thejackknife, the bootstrap and other resampling plans. *Philadelphia: Society for Industrial and Applied Mathematics*, 38:92, 1982.

Bradley Efron. Missing data, imputation, and the bootstrap. *Journal of the American Statistical Association*, 89(426):463–475, 1994.

Peter D England and Richard J Verrall. Stochastic claims reserving in general insurance. *British Actuarial Journal*, 8(3):443–518, 2002.

Barry J Epstein, Ralph Nach, and Steven M Bragg. *Wiley GAAP 2010: Interpretation and application of generally accepted accounting principles*. John Wiley & Sons, 2009.

Andrea Gabrielli and Mario V. Wüthrich. An individual claims history simulation machine. *Risks*, 6(2):29, 2018.

Nathaniel R Goodman. Statistical analysis based on a certain multivariate complex gaussian distribution (an introduction). *The Annals of mathematical statistics*, 34(1):152–177, 1963.

Martin F Grace and J Tyler Leverty. Political cost incentives for managing the property-liability insurer loss reserve. *Journal of Accounting Research*, 48(1):21–49, 2010.

Daniel F Heitjan and Srabashi Basu. Distinguishing "missing at random" and "missing completely at random". *The American Statistician*, 50(3):207–213, 1996.

Joseph M Hilbe. *Negative binomial regression*. Cambridge University Press, 2011.

Michael I Jordan and Tom M Mitchell. Machine learning: Trends, perspectives, and prospects. *Science*, 349(6245):255–260, 2015.

Steven N Kaplan and Richard S Ruback. The valuation of cash flow forecasts: An empirical analysis. *The journal of Finance*, 50(4):1059–1093, 1995.

Ron Kohavi et al. A study of cross-validation and bootstrap for accuracy estimation and model selection. In *Ijcai*, volume 14, pages 1137–1145. Montreal, Canada, 1995.

Sotiris B Kotsiantis. Decision trees: a recent overview. *Artificial Intelligence Review*, 39: 261–283, 2013.

Jitendra Singh Kushwah, Atul Kumar, Subhash Patel, Rishi Soni, Amol Gawande, and Shyam Gupta. Comparative study of regressor and classifier with decision tree using modern tools. *Materials Today: Proceedings*, 56:3571–3576, 2022.

Chanyeong Kwak and Alan Clayton-Matthews. Multinomial logistic regression. *Nursing research*, 51(6):404–410, 2002.

Kamakshi Lakshminarayan, Steven A Harp, Robert P Goldman, Tariq Samad, et al. Imputation of missing data using machine learning techniques. In *KDD*, volume 96, 1996.

Roger J Lewis. An introduction to classification and regression tree (cart) analysis. In *Annual meeting of the society for academic emergency medicine in San Francisco, California*, volume 14. Department of Emergency Medicine Harbor-UCLA Medical Center Torrance San . . . , 2000.

Thomas Mack. Distribution-free calculation of the standard error of chain ladder reserve estimates. *ASTIN Bulletin: The Journal of the IAA*, 23(2):213–225, 1993.

Thomas Mack. Which stochastic model is underlying the chain ladder method? *Insurance: mathematics and economics*, 15(2-3):133–138, 1994.

Susan A Murphy and Aad W Van der Vaart. On profile likelihood. *Journal of the American Statistical Association*, 95(450):449–465, 2000.

Carol M Musil, Camille B Warner, Piyanee Klainin Yobas, and Susan L Jones. A comparison of imputation techniques for handling missing data. *Western journal of nursing research*, 24(7):815–829, 2002.

In Jae Myung. Tutorial on maximum likelihood estimation. *Journal of mathematical Psychology*, 47(1):90–100, 2003.

John Ashworth Nelder and Robert WM Wedderburn. Generalized linear models. *Journal of the Royal Statistical Society Series A: Statistics in Society*, 135(3):370–384, 1972.

Muhammad S Osman, Adnan M Abu-Mahfouz, and Philip R Page. A survey on data imputation techniques: Water distribution system as a use case. *IEEE Access*, 6:63279–63291, 2018.

Krzysztof Ostaszewski and Grzegorz A Rempala. Parametric and nonparametric bootstrap in actuarial practice. *University of Louisville*, 2000.

Soham Pathak, Indivar Mishra, and Aleena Swetapadma. An assessment of decision tree based classification and regression algorithms. In *2018 3rd International Conference on Inventive Computation Technologies (ICICT)*, pages 92–95. IEEE, 2018.

Angshuman Paul, Dipti Prasad Mukherjee, Prasun Das, Abhinandan Gangopadhyay, Appa Rao Chintha, and Saurabh Kundu. Improved random forest for classification. *IEEE Transactions on Image Processing*, 27(8):4012–4024, 2018.

Jennie Pearce and Simon Ferrier. An evaluation of alternative algorithms for fitting species distribution models using logistic regression. *Ecological modelling*, 128(2-3):127–147, 2000.

Mathieu Pigeon, Katrien Antonio, and Michel Denuit. Individual loss reserving with the multivariate skew normal framework. *ASTIN Bulletin: The Journal of the IAA*, 43(3): 399–428, 2013.

Mathieu Pigeon, Katrien Antonio, and Michel Denuit. Individual loss reserving using paid–incurred data. *Insurance: Mathematics and Economics*, 58:121–131, 2014.

R Tyrrell Rockafellar, Stanislav Uryasev, et al. Optimization of conditional value-at-risk. *Journal of risk*, 2:21–42, 2000.

Christel Rushing, Anuradha Bulusu, Herbert I Hurwitz, Andrew B Nixon, and Herbert Pang. A leave-one-out cross-validation sas macro for the identification of markers associated with survival. *Computers in biology and medicine*, 57:123–129, 2015.

Yusuf Sahin, Serol Bulkan, and Ekrem Duman. A cost-sensitive decision tree approach for fraud detection. *Expert Systems with Applications*, 40(15):5916–5923, 2013.

Sergey Sarykalin, Gaia Serraino, and Stan Uryasev. Value-at-risk vs. conditional value-at-risk in risk management and optimization. In *State-of-the-art decision-making tools in the information-intensive age*, pages 270–294. Informs, 2008.

Judi Scheffer. Dealing with missing data. 2002.

Mark R Segal. Machine learning benchmarks and random forest regression. 2004.

Yan-Yan Song and LU Ying. Decision tree methods: applications for classification and prediction. *Shanghai archives of psychiatry*, 27(2):130, 2015.

James N Stanard. *A simulation test of prediction errors of loss reserve estimation techniques*. New York University, Graduate School of Business Administration, 1985.

Jiang Su and Harry Zhang. A fast decision tree learning algorithm. In *Aaai*, volume 6, pages 500–505, 2006.

Greg Taylor. Bayesian chain ladder models. *ASTIN Bulletin: The Journal of the IAA*, 45 (1):75–99, 2015.

Cheng Wen, Haijun Wang, Yuekang Li, Shengchao Qin, Yang Liu, Zhiwu Xu, Hongxu Chen, Xiaofei Xie, Geguang Pu, and Ting Liu. Memlock: Memory usage guided fuzzing.

In *Proceedings of the ACM/IEEE 42nd International Conference on Software Engineering*, pages 765–777, 2020.

Raymond E Wright. Logistic regression. 1995.

Wei Wu, Xiaorong Gao, and Shangkai Gao. One-versus-the-rest (ovr) algorithm: An extension of common spatial patterns (csp) algorithm to multi-class case. In *2005 IEEE Engineering in Medicine and Biology 27th Annual Conference*, pages 2387–2390. IEEE, 2006.

Zhongheng Zhang. Missing data imputation: focusing on single imputation. *Annals of translational medicine*, 4(1), 2016.

Xiubin Zhu, Witold Pedrycz, and Zhiwu Li. Granular models and granular outliers. *IEEE Transactions on Fuzzy Systems*, 26(6):3835–3846, 2018.

Martin Huldrych Zimmerman, Claud L Brown, et al. *Trees: structure and function.* New York, USA, Springer-Verlag., 1971.