A Study on Life Insurance Early Claim Detection Modeling by Considering Multiple Features Transformation Strategies for Higher Accuracy

Tham Hiu Huen[1], Lim Tong Ming[2]
Faculty of Computing and Information Technology Tunku Abdul Rahman University of Management and Technology Kuala Lumpur, Malaysia[1]
Centre for Business Incubation and Entrepreneurial Ventures, Tunku Abdul Rahman University of Management and Technology Kuala Lumpur, Malaysia[2]

***Abstract*—Early claims in the life insurance sector can lead to significant financial losses if not properly managed. This paper experiments a number of feature selection such as values regrouping, over or undersampling, and encoding that aim to enhance early claim detection by considering five (5) different machine learning algorithms. Utilizing the built-in feature importance from Random Forest, along with regrouping and correlation techniques, we identify the top seven (7) most significant features from a total 800 feature candidates. Our proposed strategy provides a streamlined and effective way to focus on the most relevant features, thereby improving the accuracy and precision of early claim predictive models for the life insurance domain. The results of this study offer practical insights into reducing fraudulent claims and mitigating financial risk. We used Random Forest besides considering techniques such as LightGBM, XGBoost, Feed Forward Neural Network, and CatBoost to train our model and achieved a maximum accuracy of 0.92 across three samples, indicating that our approach can effectively identify critical features and produce reliable results.**

***Keywords*—Machine learning; feature selection; life insurance; binary classification; Random Forest**

I. Introduction

Early claims in the life insurance industry pose a significant financial loss if the risk of policies sold is high. Life insurance companies, tasked with processing large volumes of claims, are especially vulnerable to early claims that may indicate sophisticated fraud schemes. Without effective detection mechanisms, these organizations may suffer substantial financial loss and reputational harm [1] [2]. Given the complexity and scale of modern insurance operations, efficient early claim detection is more crucial than ever.

One of the critical challenges in early claim detection is dealing with large datasets containing hundreds or even thousands of features. Not all of these features are relevant, and attempting to process all of them can lead to computational inefficiencies and reduced detection accuracy. Therefore, highly reliable feature selection techniques are essential to streamline the process and improve fraud detection outcomes.

In the United States, insurance fraud is thought to cost the country $308.6 billion a year [3]. The average cost of insurance fraud to a customer is estimated to be $900, primarily because the deception raises rates [3]. Health care insurance fraud (including Medicaid and Medicare insurance fraud) is the most expensive category of insurance fraud, costing customers an estimated $105 billion a year. Life insurance fraud comes in second with $74.7 billion, while property and casualty insurance fraud come in third with $45 billion [3].

The impact of insurance fraud activities includes Loss of Personal Income, & Savings, Higher Insurance Premiums, High Personal Costs, Ruined credit, Loss of Jobs, Diverts Government Resources, Loss from Essential Services and Rising cost of Goods & Services [4]. Fraudulent activity in the life insurance industry raises costs and leads to inflated premiums. As a result, having a solid risk management framework is critical for preventing or reducing life insurance fraud [5].

In the following sections, we discuss some past research works and detailed methodology used for data preparation, feature selection and model development. A discussion on the machine learning techniques used in this research and justifications for these techniques will be presented. An experimental setup and analysis and discussion of the results will be presented. We also highlight areas for future research and offer recommendations for implementing our approach in real-world settings.

II. Related Work

In this section, we review past research works on the insurance predictive models building and data preprocessing techniques.

A. Insurance Prediction Model Techniques

The author in study [6] emphasized the significance of the SCOR library for dealing with censored data in the machine learning model family. Various machine learning methods such as XGBoost, CatBoost and LightGBM have been adapted to the specificities of life insurance data, particularly censoring and truncation. On the other hand, [7] introduced data visualization techniques for decision support in the insurance sector. This study proposed that claim analysis can be used to distinguish between fraudulent and genuine claims; it also helped to better understand the customer strata while using the results throughout the underwriting and acceptance/denial stages of insurance enrollment.

The findings in study [8] revealed that ensemble-based approaches (random forest and gradient boosting) and deep neural networks produced the greatest results, outperforming other classifiers, including the widely used logistic regression.

Authors in study [9] aimed to use massive health insurance claims data to predict very high-cost claimants and show that high-performing prediction models may be built using only claims data and publicly available data, even for uncommon high-cost claimants worth more than $250,000. They created a platform with 6,006 variables across all clinical and demographic parameters and built over 100 candidate models. The best model has an area under the receiver operating characteristic curve of 91.2% which indicates that it possesses a high level of accuracy and discriminative power in predicting very high-cost claimants.

On the other hand, research in study [10] constructed and tested an artificial intelligence network-based regression model to forecast health insurance rates. The authors predicted that the health insurance costs experienced by people based on their characteristics and attained an experimental accuracy of 92.72%. In study [11], churn modelling of life insurance policies via statistical and machine learning methods is completed to analyse important features. The authors in the study [12] utilised the Random Forest approach to anticipate policyholders' decisions to lapse life insurance contracts. Even after factoring in feature interactions, the technique beats the logistic model.

The authors in the study [13] examined how car insurance companies employ machine learning into their operations and how ML models might be applied to insurance's large data. They use ML approaches including logistic regression, XGBoost, random forest, decision trees, naïve Bayes, and K-NN to predict claim incidence where the results demonstrated that RF outperformed other approaches. The authors in [14] forecasting motor insurance claims discovered that Random Forest with restricted depth and XGboost, when run on the 15 most relevant variables, outperformed the other models examined.

The study in [15] showed that data imbalance problem contributes significantly to poor model performance in insurance uptake prediction. Learning metrics improved when the data were balanced by either oversampling the minority class (insurance uptake in the instance of the data used) or undersampling the majority class (insurance non-uptake). In [16], the author enhanced the prediction accuracy by adding additional data sets to train and test the model. Features that did not influence the prediction were stripped of their features to examine how different independent factors affected the premium amount. In Table I, a summary of the papers reviewed is tabulated to justify the importance of this research.

The collective insights from the reviewed research paper underscore the significance of advanced machine learning techniques in enhancing insurance claim predictions and identifying fraudulent activities. Our research is distinguished by its comprehensive integration of diverse machine learning algorithms, including Random Forest, CatBoost, LightGBM, XGBoost, and Feedforward Neural Networks, as well as its innovative data preprocessing strategies. Unlike previous studies that focused on individual aspects such as data visualization, dealing with censored data, or specific model comparisons, our research employs a holistic approach. This includes denormalization of complex datasets, handling class imbalance through undersampling, and utilizing placeholder-based imputation for missing values to capture human behavior biases. Additionally, it incorporates quantile-based discretization for simplifying data and iterative feature selection using Random Forest's feature importance, ensuring the most relevant features are retained. The use of a chronological split further validates model performance on future, unseen data, simulating real-world applicability. By combining these methodologies, our research not only builds on the existing body of knowledge but also offers a robust, scalable framework that addresses the nuances of insurance data more effectively, ultimately leading to more accurate and reliable predictions.

TABLE I. SUMMARIZATION OF PASS RESEARCH WORK

Author	Problems	Techniques	Contributions
[6]	Decision support in the insurance sector	Data Visualization	Decision support in the insurance sector
[7]	Applying machine learning to life insurance	Machine Learning methods	Emphasized the significance of appropriate implementations for dealing with censored data in the machine learning model family
[8]	Fraud prediction in property insurance	Machine learning algorithms	Empirical evidence using real-world microdata
[9]	Using massive health insurance claims data to predict very high-cost claimants	Built over 100 candidate models	The best model has an area under the receiver operating characteristic curve of 91.2%.
[10]	Predict health insurance premiums	Regression framework	Attained an experimental accuracy of 92.72%.
[11]	Churn modeling of life insurance	Churn modelling, statistic	Analysis of important features
[12]	A machine learning model for lapse prediction in life insurance contracts	Random Forest	Anticipate policyholders' decisions to lapse life insurance contracts
[13]	Machine learning approaches for auto insurance big data	Logistic regression, XGBoost, random forest, decision trees, naïve Bayes, and K-NN	Predict claim incidence where the results demonstrated that RF outperformed other approaches.
[14]	Predict motor insurance claims occurrence	Random Forest with restricted depth and XGboost	Research on imbalanced machine learning problem

[15]	Predict insurance uptake in Kenya	Oversampling and undersampling	A comparative analysis of machine learning models
[16]	Predict medical insurance cost	Forest regression algorithms	An accurate prediction of medical insurance cost

B. Data Preprocessing Technique

The authors in [17] suggested two strategies to address the issue of numerous majority class examples being disregarded in undersampling. EasyEnsemble selects various subsets from the majority class, trains a learner on each one, and integrates the results. BalanceCascade educates the learners in stages, with the majority of class examples properly identified by the present trained learners being eliminated from consideration at each stage. Experimental data reveal that both approaches have a greater area under the ROC Curve, F-measure, and G-mean values are higher than those of several other class imbalance learning approaches.

Missing data is a systemic issue in real circumstances, resulting in noise and bias when evaluating treatment outcomes. The solution in study [18] is selective imputation, which uses insights from mixed confounded missingness (MCM) to determine which variables should be imputed and which should be excluded. The authors empirically illustrate how selective imputation benefits distinct learners as compared to alternative missing-data methods. In study [19], a Monte Carlo simulation was used to evaluate the influence of the imputation approach on the bias and efficiency of scale-level parameter estimations, such as scale score means, between-scale correlations, and regression coefficients. The empirical data analysis results were consistent with those of the simulation, indicating that researchers should exercise caution when adopting planned missing data designs that require scale-level imputation.

The authors of study [20] presented a quantile-based criterion for the sequential design of trials, similar to the standard anticipated improvement criterion that allows for an elegant treatment of heterogeneous response precision. By analyzing both actual and simulated data, [21] showed that the permutation feature importance metric delivers more precise feature importance rank estimation in the presence of non-additive interactions. The authors of study [22] chose feature selection strategies based on correlation analysis and variance of input characteristics before sending these key features to a classification algorithm. Dimensionality was reduced using correlation and main component analyses.

The study in [23] attempted to identify an ideal strategy to mitigate the negative effects of option overload by assortment classification. This research contends that the number of possibilities under each label is more significant for preventing choice overload than the number of labels. This research discovers that a few labels are useful only when the category ratio falls within the specified ideal range. When categorised with the ideal category ratio, uninformative labels decreased option overload.

The experimental results in study [24] showed that the use of random splits can significantly overestimate predictive performance across all datasets and models. Therefore, the authors suggested that rumour detection models should always be evaluated using chronological splits for minimising topical overlaps. The study in [25] explained the ambiguous terminology, gave explicit principles for distinguishing between measures and metrics for the first time, and presented a new-fully visualised roadmap in a leveled structure for 22 measures and 22 metrics for investigating binary classification performance. In Table II, it tabulates all the papers reviewed in this article and their key contributions.

TABLE II. SUMMARIZATION OF PASS RESEARCH WORK

Author	Problems	Techniques	Contributions
[17]	Exploratory undersampling for class-imbalance learning	EasyEnsemble, BalanceCascade	Address the issue of numerous majority class examples being disregarded in undersampling.
[18]	To impute or not to impute	Mixed confounded missingness	Missing data in treatment effect estimation
[19]	A comparison of item-level and scale-level multiple imputation for questionnaire batteries	Monte Carlo simulation	Evaluate the influence of the imputation approach on the bias and efficiency
[20]	Noisy computer experiments	Quantile-based optimization	Quantile-based optimization of noisy computer experiments with tunable precision
[21]	Genetic association in the presence of non-additive interactions	Random forest models	A comparison of methods for interpreting random forest models of genetic association
[22]	Analysis for accurate breast cancer diagnosis	Correlation analysis, principal component	Feature selection using correlation analysis and principal component
[23]	Search for an optimal solution to reduce choice overload	Category Ratio	Discovers that a few labels are useful only when the category ratio falls within the specified ideal range
[24]	Rethinking evaluation on rumor detection benchmarks using chronological splits	Chronological splits	Evaluation on chronological splits
[25]	Binary classification performance measures/metrics	22 performance metrics	A comprehensive visualized roadmap to gain new insights on performance metrics

The past research paper underscores the multifaceted challenges inherent in data preprocessing and preparation within the realm of machine learning applications, particularly in addressing class imbalance and missing data issues. Our

research endeavors to build upon these insights by implementing a comprehensive approach to data preprocessing and preparation. Inspired by strategies such as EasyEnsemble and BalanceCascade for handling class imbalance, our research employs undersampling techniques while mitigating the potential loss of majority class examples. Additionally, placeholder-based imputation, as suggested in selective imputation, guides the treatment of missing data, ensuring a nuanced approach that minimizes noise and bias in model evaluations. Furthermore, by integrating dimensionality reduction techniques like correlation analysis, our research ensures that only the most relevant and informative features are retained for model training. This approach, validated by empirical and simulated studies, aims to enhance the robustness and efficiency of machine learning models, thereby addressing the challenges highlighted across various studies. Through these meticulously designed data preprocessing and preparation steps, our research aims to contribute to the advancement of predictive modeling in complex domains such as insurance, where data quality and accuracy are paramount.

III. Methodology

In this section, the research flow adopted is explained and justified. A discussion on the resources this research requires and the data to be used in this research is presented.

A. Research Workflow

In Fig. 1, the research flow adopted is presented. Every set of research activities is briefly explained in each stage of the activities.

1) Business understanding: Based on Fig. 1, the business challenge has been identified and the goal of the research is specified. It is important to collaborate extensively with business stakeholders to better understand the challenge and set targets.

2) Study requirement: Next, it is essential to study the information of related areas. This includes conducting an extensive review of the existing literature works related to the research topic. In addition, we identify key theories, concepts, and findings that contribute to the understanding of the research problem.

3) Data acquirement: It is critical to gather a comprehensive dataset that includes historical examples with both input features and corresponding target labels. This dataset will be utilised for training and testing the model. We collaborate with domain experts to ensure the dataset is representative and sufficient for the research objectives.

4) Table denormalization: As illustrated in Fig. 1, the data gathered has been denormalised. To facilitate effective analysis and modeling, we join the relevant tables using the appropriate join keys to denormalize the relational database. This process consolidates the data into a single table, making it easier to manage and analyze.

5) Data cleaning: Data cleaning is a critical step that addresses various issues within the dataset, such as missing values, duplication, and inconsistencies. This process ensures that the data used for modeling is of high quality and reliable. Techniques such as imputation for missing values, statistical methods for outlier detection, and consistency checks are employed to clean the data.

6) Exploratory data analysis (EDA): Exploratory Data Analysis (EDA) involves using data visualization and statistical techniques to understand the dataset's underlying patterns, relationships, and trends. By creating various plots, such as histograms, scatter plots, and correlation matrices, EDA helps in identifying significant variables and detecting anomalies or unusual patterns. Summary statistics provide insights into the distribution and central tendencies of the data, which are crucial for making informed decisions during model development. The findings from EDA guide the feature selection and engineering processes, helping to refine the model and improve its predictive performance.

7) Feature selection: Feature selection is the process of identifying and retaining the most relevant variables that significantly contribute to predicting the target outcome. This step is essential for enhancing the model's performance and reducing its complexity. By analyzing the correlation between features and their importance scores from preliminary models, redundant or highly correlated features are eliminated to avoid issues like multicollinearity. Incorporating domain knowledge helps in understanding the significance of each feature in the business context, ensuring that the selected features are meaningful and valuable for the modelling process.

8) Feature engineering: It is crucial to develop and refine features to more accurately represent the problem. This involves creating new features from existing data and selecting the most relevant ones to enhance model performance. Additionally, employ dimensionality reduction techniques if needed to simplify the model and improve its efficiency.

9) Encoding: Encoding converts categorical variables into a numerical format, which is essential for most machine learning algorithms to process categorical data effectively. For high cardinality features, target encoding or mean encoding may be used to preserve the feature's information without overly increasing the dataset's dimensionality. It's crucial to ensure that the encoding process does not introduce bias or affect the model's interpretability.

10)Train-test split: To evaluate the model's performance, the dataset is split into separate training and test sets. Creating a validation set from the training data is also common to fine-tune model parameters and avoid overfitting. This partitioning allows the model to be trained on one subset of data and tested on another, providing a realistic assessment of how well it generalizes to unseen data.

11)Model training: Then, the dataset is split into training and test sets to evaluate model performance. We choose an appropriate machine learning algorithm to build the predictive model by considering their strengths, weaknesses, and suitability for the business challenge at hand.

12)Model evaluation: The model's performance and its ability to address the business challenge has been thoroughly assessed by accuracy, precision, recall and F1-score. We use a

range of evaluation metrics to gauge the model's effectiveness. Based on the results, refine and optimize the model to ensure it meets the desired accuracy and reliability standards.

13)Deployment: We then implement the final model in a production environment, integrating it seamlessly into existing business processes. Ensure the model operates as intended and delivers the expected results. Ongoing monitoring and maintenance are essential to keep the model effective and responsive to any changes in the business context.

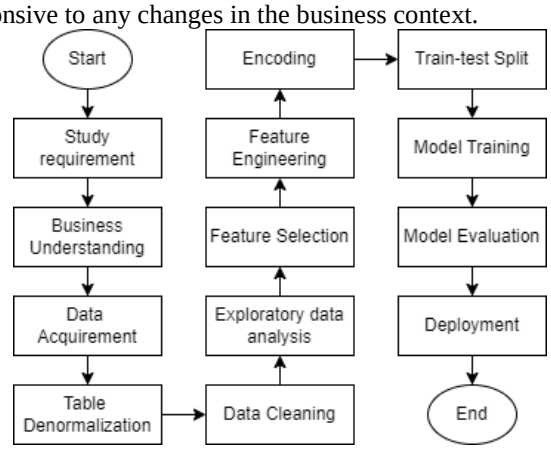

Fig. 1. Research workflow.

B. Resource Used

The hardware resources used in this setup include two servers: a Linux-based workstation and an IBM Power-Server. The Linux workstation offers 144,428.9 MiB (approximately 144.4 GiB) of memory, providing ample capacity for running data-intensive tasks. The IBM Power-Server, with 191,855.6 MiB (approximately 191.9 GiB), is designed for high-performance computing, ensuring that complex computations and large datasets can be processed efficiently. Connectivity between the servers and users is established through a Virtual Private Network (VPN) using FortiClient VPN, which ensures secure and encrypted communication over public networks.

On the software side, the primary integrated development environment (IDE) used is Jupyter Notebook, a versatile platform ideal for data analysis and machine learning tasks. The Python libraries utilised in this environment include PySpark for large-scale data processing, Pandas for data manipulation, Matplotlib for data visualisation, and Scikit-learn (sklearn) for machine learning algorithms. These libraries provide a robust set of tools for analysing, visualising, and modeling data, making the environment suitable for data science and artificial intelligence applications.

C. Data Nature

The data used in this research project is sourced from an insurance information service provider, spanning a comprehensive period of 20 years, from 2003 to 2023. This extensive timespan provides a rich dataset, allowing for in-depth analysis of long-term trends and patterns for this project. The data encompasses a diverse range of information, including policy details, claims history, customer demographics, and financial transactions. Due to data privacy protection regulation, this paper will not reveal other details.

In terms of structure, the data is organised into 12 distinct tables, which collectively contain a total of 7,103,548 rows and 861 columns. This considerable data volume necessitates robust data processing and storage capabilities. The wide variety of columns reflects the intricate nature of insurance-related data, with each table offering specific insights into various aspects of the business. One key aspect of the dataset is the target variable, used for predictive modeling, which has 7,032,993 rows labelled as 0 indicating policyholders have not made any claim whereas 70,555 rows labelled as 1 or 'policyholder have claimed', accounting for approximately 0.01% of the total population of the data. This imbalance data distribution between the two target values suggest that these is a need for advanced techniques to be considered in order to manage the class imbalance prior to the predictive model's development work. In addition, some of the features are found to have a large number of null values and a high number of distinct values. As such, ***strategies*** to manage null values, data binning for features of continuous type and data encoding for categorical features are highly essential for this research. Overall, the data provides a comprehensive foundation for detailed analysis and the development of data-driven strategies within the insurance sector.

IV. Preprocessing and Data Preparation

A. Denormalization

The project used memory-based processing where Python and Spark merge twelve (12) tables into a unified dataset, ensuring scalability and speed. The main method used was the inner join, which retrieves only the rows with matching keys in both datasets. Denormalization reduces field redundancy by joining related tables on common keys. Left joins were used to incorporate information from tables with optional or incomplete relationships, ensuring no key information is lost during the join process. Pivoting was used to transform NBXPROPINS table from long format to wide format (LFKPROPDETINS and LFKPROPDETINS are two columns where they have one one-to-one relationship and if one exists, another will be null), providing a more organised view and simplifying downstream analysis. The process ensured data integrity, handling incomplete data, and transforming complex table structures, providing a robust foundation for further processing and modeling. Fig. 2 depicts the entity relationship of all the tables collected.

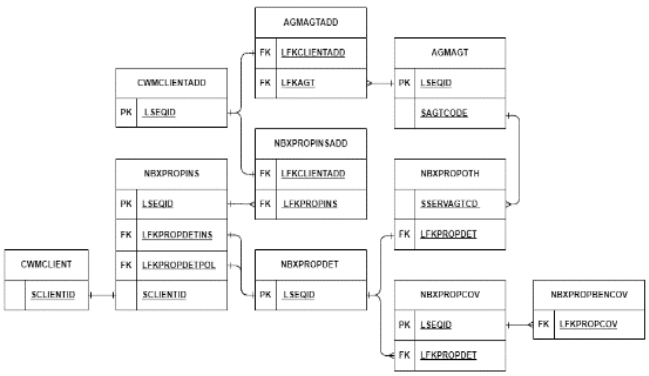

Fig. 2. Entity relationship diagram.

B. Oversampling and Undersampling

Oversampling and undersampling are techniques used to address class imbalance in datasets, which is common in many real-world scenarios [26] [27] [28]. The primary purpose of oversampling is to increase the representation of the minority class by creating synthetic samples or duplicating existing ones, thereby balancing the dataset and improving the model's ability to learn from minority instances [29] [30]. Undersampling, on the other hand, reduces the majority class by removing some of its instances, making the dataset balanced but potentially losing valuable information [31] [32]. Both techniques aim to improve model performance, particularly in classification tasks, by ensuring that the model does not become biased toward the majority class.

In our study, addressing class imbalance was a crucial part of the data preprocessing stage. The target variable exhibited significant skewness, with the majority class value '0' vastly outnumbering the minority class value '1', a common issue in many real-world datasets. To mitigate the imbalance in data distribution, we used an undersampling technique, which involves reducing the number of samples in the class value '0' to create a more balanced dataset. This approach helps to reduce bias and skewness in machine learning model building. While undersampling has the potential drawback of losing information from the majority class values, it effectively combats the tendency of models to overlook the minority class values.

C. Missing Value Handling

In traditional imputation methods, the common approach is to use central tendency statistics like the mean, median, or mode to fill in missing values. [33] Given that human input data can be prone to intentional omission for personal gain, using central tendency-based imputation could lead to misinterpretations and inaccurate predictions. By using placeholder-based imputation, we acknowledge the fact that the data has inherent biases due to human behavior, rather than sensor errors or system malfunctions. This approach can help maintain the context in which the data was originally collected, offering a more accurate representation of missing information [34].

Human input data, unlike automated sensor data, can contain omissions due to personal interests, such as avoiding higher insurance premiums. It's not illegal to leave a field blank, even though it may be against the policy's spirit. For example, someone who smokes might leave the "number of cigarettes per day" field empty to avoid being categorised as a high-risk individual. In such cases, using central tendency for imputation might not truly represent the omitted information, leading to a skewed interpretation of the data. Placeholder-based imputation ensures that the missingness itself is treated as a significant data point, which may suggest a behavioural pattern rather than a random occurrence. This allows the model to better account for intentional data omissions, leading to more robust predictions and insights into potential data-related risks. In this research, we adopt placeholder-based imputation because the context in which the data is collected significantly impacts its interpretation and subsequent analysis. By recognizing that human input data can be intentionally omitted for personal reasons, we address the inherent biases that central tendency-based imputation might overlook.

D. Data Binning using Quantile-based Discretization

Quantile-based discretization is a technique used to transform continuous variables into discrete categories based on their distribution. This approach involves dividing the continuous data into a specified number of intervals, or bins, where each bin contains roughly the same number of data points. This technique is applied to convert all features into categorical data before feature selection.

The advantage of converting continuous variables into categorical is that it simplifies complex data, making it easier for certain machine learning algorithms to process. Additionally, converting continuous variables to categorical can reduce the impact of outliers, which might otherwise skew the analysis. It also enables the use of categorical-specific modeling techniques, such as decision trees, which may perform better with discrete data.

E. Iterative Feature Selection

In this study, we employed an iterative feature selection methodology to identify and select the most appropriate features for our machine learning model. The iterative process allows us to continually refine our feature set until the desired level of model performance is achieved. At this stage, we applied a feature selection technique to identify the most relevant features. We used algorithms with feature importance metrics, such as Random Forests [35][36][37], to rank features based on their contribution to model performance. Features with low importance or high redundancy were removed. After evaluating the model, a decision was made on whether further feature selection was needed. If the results were satisfactory, the process ended. If not, we returned to the feature selection step for further refinement. Fig. 3 illustrates the flow chart of iterative feature selection.

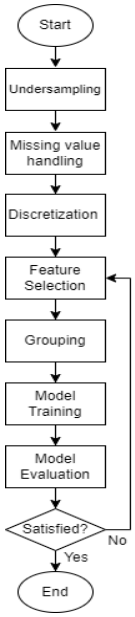

Fig. 3. Flow chart of iterative feature selection.

F. Random Forest's Feature Importance as the Key Selection Indicator

Random forest is adopted as the algorithm produces feature importance scores that are significant as an indicator for the feature selection process in this research. Random forest composed of multiple decision trees, each built from a random subset of features and a random sample of the training data (bootstrapping). As these trees are constructed, each feature is used to split the data, and the quality of these splits is evaluated using metrics such as Gini impurity.

Gini Importance:

$$ni_j = w_j C_j - w_{left(j)} C_{left(j)} - w_{right(j)} C_{right(j)} \quad (1)$$

ni_j = the importance of node j
w_j = weighted number of samples reaching node j
C_j = the impurity value of node j
left(j) = child node from left split on node j
right(j) = child node from right split on node j

G. Correlation Analysis

Correlation analysis is a valuable technique for identifying and dropping features that have high dependency or redundancy. When building a machine learning model, redundant features can lead to overfitting, increased complexity, and reduced interpretability. This research utilises Pearson Correlation Coefficient to measure linear relationships between continuous variables. We select only one feature from each group of highly correlated features. The correlation values range from -1 to 1, where -1 indicates perfect negative correlation, 0 indicates no correlation, and 1 indicates perfect positive correlation.

H. Category Ratio using Target Ratio Grouping

In this research, the challenge of handling categorical features with a high number of unique classes, many of which have limited data and skewed target distributions, is addressed through a method of class grouping based on target ratios. This technique, often referred to as target ratio grouping or data binning, aims to reduce the cardinality of categorical features to improve model robustness and avoid overfitting.

Here, categorical classes are regrouped based on their target ratio, which is calculated as the proportion of one target value within the class. Classes with extreme ratios (such as 1:0 or 0:1) tend to skew the model's performance due to their lack of variability and are prone to overfitting. To mitigate this, classes with similar target ratios are grouped into broader categories according to predefined rules. This approach not only reduces the number of unique classes but also helps ensure the model is not overly sensitive to rarely occurring classes or extreme outliers.

Our grouping rules categorised classes into one of the seven groups based on their target ratio:

- If the target ratio is 0.0 (i.e., the class has no instances of a specific target value), it is assigned to group 1.
- If the target ratio is greater than 0 and less than or equal to 0.2, the class is assigned to group 2.
- A target ratio greater than 0.2 and less than or equal to 0.4 assigns the class to group 3.
- For ratios greater than 0.4 and less than or equal to 0.6, the class falls into group 4.
- Ratios greater than 0.6 and less than or equal to 0.8 are categorised into group 5.
- Ratios greater than 0.8 but less than 1.0 are assigned to group 6.
- Finally, classes with a ratio of 1.0 are grouped into group 7, as they represent a consistent outcome.

By using this method, the cardinality of categorical features is significantly reduced, leading to more manageable datasets and a lower risk of overfitting. This regrouping strategy helps improve model generalisation and efficiency, allowing the model to focus on meaningful patterns without being affected by the noise from rarely occurring or highly skewed classes. This approach has demonstrated benefits in our research, leading to better model performance and reliability.

I. Chronological Split

A chronological split is a method of dividing a dataset into two subsets, typically for training and testing machine learning models. The split is based on a chronological criterion, such as the year or date a policy goes into effect. In this approach, the training set consists of data before year 2020, while the test set includes data from only year 2020. By splitting data in this way, we ensure that the model is trained on earlier information and tested on subsequent, unseen data, reflecting a more realistic scenario. This technique helps evaluate the model's ability to generalize and perform accurately on future data, providing a more robust assessment of its real-world applicability.

Using a time-based split ensures that the machine learning model is evaluated on data from a distinct and future period, which better simulates real-world deployment scenarios. By dividing the dataset according to the year, a policy goes into effect, the model is tested on data that is more representative of future conditions, behaviors, and trends. This approach is particularly useful in time-sensitive domains like insurance, where regulations, customer behavior, and external factors can change over time. It allows the project to assess how well the model can generalize beyond the training data, giving a more realistic indication of its performance in production.

A chronological split provides greater confidence that the model will maintain its accuracy and effectiveness when predicting future data, as it has been validated against a set that follows the temporal sequence of real-world events. This technique also helps identify if a model is overly reliant on historical patterns that may not persist in the future, thereby reducing the risk of overfitting to a particular timeframe or dataset characteristic.

V. Modeling Techniques and Measurement Metrics

In this research, five (5) machine learning algorithms had been evaluated. The following subsections briefly elaborate their theoretical architecture and models.

A. Random Forest

Random Forest is a powerful ensemble learning method that constructs multiple decision trees during training and combines

their predictions through voting or averaging to make final decisions. It excels in handling high-dimensional data and is robust against overfitting, making it well-suited for insurance claim prediction tasks where the dataset may contain numerous input features and a relatively small number of samples. Furthermore, Random Forest provides a measure of feature importance, allowing insurers to identify the most influential variables in predicting claim outcomes. Its ease of implementation and interpretability make it a popular choice for binary classification tasks in the insurance industry, offering a balance between predictive accuracy and model transparency. The architecture of Random Forest is shown in Fig. 4 [38].

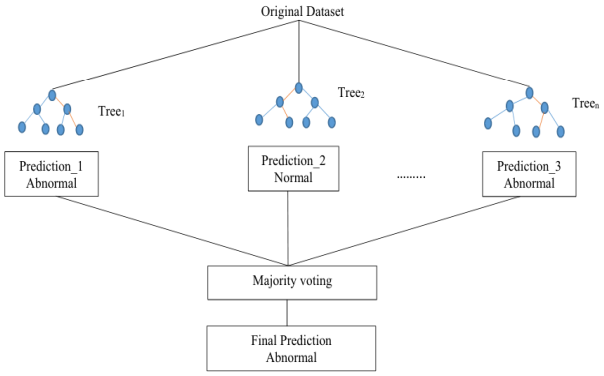

Fig. 4. Architecture of Random Forest.

The diagram in Fig. 4 illustrates a Random Forest classifier, which combines multiple decision trees to improve prediction accuracy. Each tree is built using a different subset of the original dataset and considers a random subset of features for splitting at each node, introducing diversity among the trees. Each tree independently predicts an outcome (e.g., "Abnormal" or "Normal"). The final prediction is determined by majority voting among all the trees' predictions, making the model more robust and reducing the risk of overfitting compared to a single decision tree. In this example, the majority vote results in a final prediction of "Abnormal".

B. CatBoost

CatBoost is a gradient boosting library specifically designed to handle categorical features efficiently, making it an ideal choice for insurance claim prediction tasks where categorical variables play a significant role. It employs gradient boosting techniques to build an ensemble of decision trees, automatically handling categorical features without requiring extensive preprocessing. CatBoost often provides competitive performance out-of-the-box and is less sensitive to hyperparameter tuning compared to other gradient boosting methods. Its ability to handle large datasets and categorical variables effectively makes it a valuable tool for insurers seeking accurate and reliable predictions of claim outcomes while minimizing the need for manual feature engineering. The architecture of Random Forest is shown in Fig. 5 [39].

Fig. 5 illustrates the architecture of CatBoost, a gradient boosting algorithm designed for categorical features. Starting with the UCS (universal concept space) dataset containing *NN* samples and *MM* features, the data is split into bootstrap samples to create multiple training datasets. Each training dataset is used to sequentially build *NN* decision trees (predictors), with each tree improving upon the previous ones. The model incorporates a unique feature called "weight expansion," which adjusts the weights of misclassified samples to emphasize harder-to-classify instances. After training, the individual predictions from each tree are combined through weighted averaging to produce the final prediction, optimizing performance and accuracy by effectively handling categorical variables and reducing overfitting.

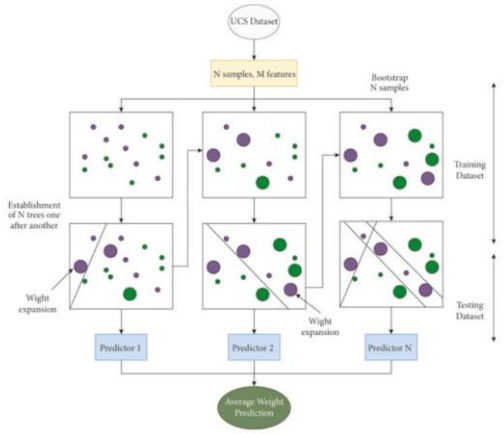

Fig. 5. Architecture of CatBoost.

C. LightGBM

LightGBM is a gradient boosting framework known for its efficiency and speed, making it particularly well-suited for insurance claim prediction tasks involving large volumes of data. It uses a novel tree-based learning algorithm that prioritizes training instances with high gradients, resulting in faster convergence and reduced computational costs. LightGBM is highly scalable and can handle large-scale datasets with millions of samples and features efficiently. Its ability to handle categorical features and missing data effectively further enhances its suitability for insurance claim prediction, where data may be incomplete or heterogeneous. Overall, LightGBM offers a compelling combination of speed, scalability, and predictive accuracy, making it a valuable asset for insurers seeking efficient and reliable models for binary classification tasks. The architecture of Random Forest is shown in Fig. 6 [40].

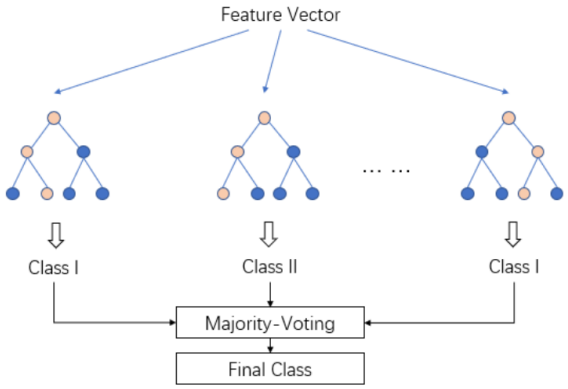

Fig. 6. Architecture of LightGBM.

Fig. 6 illustrates the architecture of a LightGBM (Light Gradient Boosting Machine) ensemble model, which utilizes multiple decision trees to make predictions. Each decision tree receives the same feature vector as input and produces a class prediction. These individual predictions are then aggregated through a majority-voting mechanism to determine the final class. The idea is that by combining the outputs of multiple trees, the model can achieve more accurate and robust predictions, leveraging the collective decision-making of the ensemble rather than relying on a single tree's output. This approach helps reduce overfitting and improves generalization performance.

D. XGBoost

XGBoost, short for eXtreme Gradient Boosting, is a scalable and efficient implementation of gradient boosting. It is widely used in insurance claim prediction tasks due to its exceptional performance and versatility. XGBoost employs a regularized learning objective that combines both gradient descent and second-order gradient descent, allowing it to capture complex patterns in the data while minimizing overfitting. Its ability to handle missing values and categorical features, along with built-in support for parallel computing, makes it well-suited for large-scale datasets common in insurance applications. XGBoost often achieves state-of-the-art results in various machine learning competitions and has become a go-to choice for insurers seeking accurate and robust models for binary classification tasks. Its interpretable nature, feature importance analysis, and ease of use further enhance its appeal, making XGBoost a valuable asset in the insurance industry's quest for reliable predictive models. The architecture of Random Forest is shown in Fig. 7 [41].

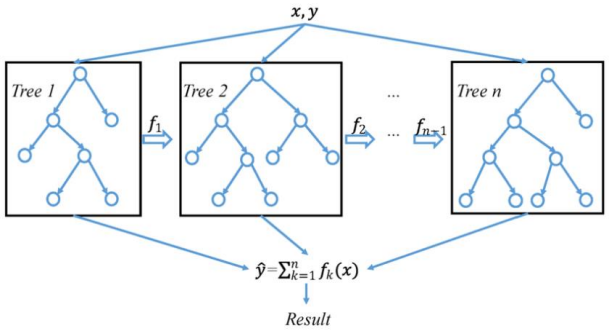

Fig. 7. Architecture of XGBoost.

The diagram in Fig. 7 represents the architecture of the XGBoost (Extreme Gradient Boosting) model, which is an ensemble learning technique that builds multiple decision trees sequentially. Each tree in the sequence is trained to correct the errors made by the previous trees. The input data, consisting of features xx and target yy, is used to train the first tree, $f1f1$. The output from this tree, along with the data, is then used to train the next tree, $f2f2$, and this process continues for all nn trees. The final prediction, $y^y^$, is obtained by summing the outputs of all the trees, expressed as $y^=\sum k=1nfk(x)y^=\sum k=1nfk(x)$. This iterative process allows XGBoost to minimize the overall prediction error, making it a powerful and accurate model for various predictive tasks.

E. Feed Forword Neural Network

A Feed Forward Neural Network implemented using TensorFlow is a deep learning model capable of learning complex patterns in the data through multiple layers of neurons. It offers flexibility in designing and customizing neural network architectures, allowing insurers to adapt the model to the specific characteristics of their data. While neural networks have the potential to outperform traditional machine learning models in certain scenarios, they often require extensive hyperparameter tuning and larger amounts of data to achieve optimal performance. Nevertheless, their ability to learn intricate relationships in the data makes them well-suited for insurance claim prediction tasks where the underlying patterns may be nonlinear or complex. With careful tuning and training, Feed Forward Neural Networks implemented using TensorFlow can offer competitive performance and provide valuable insights into claim outcomes for insurers. The architecture of Random Forest is shown in Fig. 8 [42].

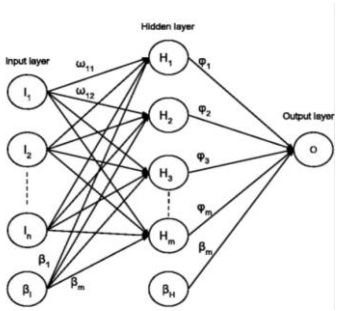

Fig. 8. Architecture of neural network.

The diagram in Fig. 8 illustrates the architecture of a basic feedforward neural network, consisting of three main layers: the input layer, hidden layer, and output layer. The input layer has nodes $I1,2,\ldots,InI1,I2,\ldots,In$, each representing a feature of the input data. These input nodes are connected to nodes in the hidden layer $H1,2,\ldots,HmH1,H2,\ldots,Hm$ through weighted connections $\omega\omega$. Each hidden node applies an activation function $\phi\phi$ to its input, transforming the data in a non-linear manner. The hidden layer nodes are then connected to the output node oo, which combines these inputs to produce the final output. Bias terms $\beta1,\beta H\beta1,\beta H$, etc., are also included in the layers to improve the model's ability to fit the data. This architecture enables the network to learn complex patterns in the data by adjusting the weights and biases through training processes like backpropagation.

F. Performance Metrics

Evaluating the performance of binary classification models is a critical aspect of any machine learning project. This research utilizes accuracy, precision, recall and F1 score as performance metrics.

1) Accuracy: Accuracy measures the proportion of correct predictions (both true positives and true negatives) among the total predictions made by a model. It is calculated as:

$$Accuracy = \frac{True\ Positives + True\ Negatives}{Total\ Predictions} \quad (2)$$

Accuracy is useful when classes are balanced, but it can be misleading when dealing with imbalanced datasets, as it may overstate a model's performance by focusing on the majority class.

2) Precision: Precision measures the proportion of correctly predicted positive outcomes out of all predicted positive outcomes. It is calculated as:

$$Precision = \frac{True\ Positives}{True\ Positives + False\ Positives} \quad (3)$$

Precision is important in scenarios where false positives are costly or undesirable. High precision indicates a low rate of false positives.

3) Recall: Recall, also known as sensitivity or true positive rate, measures the proportion of correctly predicted positive outcomes out of all actual positive outcomes. It is calculated as:

$$Recall = \frac{True\ Positives}{True\ Positives + False\ Negatives} \quad (4)$$

Recall is crucial when false negatives are costly or undesirable. High recall indicates a low rate of false negatives.

4) F1 Score: The F1 score is the harmonic mean of precision and recall, providing a balance between the two metrics. It is useful when you need a single measure to evaluate a model's performance, especially when there's a trade-off between precision and recall. The F1 score is calculated as:

$$F1\ Score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \quad (5)$$

A high F1 score indicates a good balance between precision and recall, making it a robust metric for evaluating models in various scenarios, including imbalanced datasets or cases where both false positives and false negatives have significant consequences.

VI. Experiments and Analysis of Result

We had carried out experimental works using the five (5) machine learning algorithm described in Section V. Out of all the algorithms implemented, Random Forest gives the most consistent and the highest accuracy from model builds. Based on the Random Forest model, the feature that successfully being selected from over 800 features are:

TABLE III. Importance Coefficient in Random Forest

Feature	Sample 1	Sample 2	Sample 3
Benefit Component	0.396595	0.390536	0.390532
Plan Code	0.289045	0.298415	0.325608
Ri Sum At Risk	0.273424	0.270532	0.246030
Installment Premium	0.018618	0.018813	0.016871
Insured Age	0.013365	0.013160	0.013132
Policyholder Occupation	0.004761	0.004536	0.004028
Policyholder Age	0.004191	0.004008	0.003798

Table III shows the feature importance coefficients derived from a Random Forest model across three different samples. These coefficients indicate the relative significance of each feature in making predictions within the model. A higher coefficient suggests that the feature plays a more critical role in determining the outcome. In all three samples, "Benefit Component" emerges as the most important feature, with coefficients consistently near or above 0.39. This implies that variations in this feature are strongly correlated with the model's predictions, suggesting it holds substantial predictive power across multiple datasets.

"Plan Code" and "Ri Sum At Risk" are the next most important features, though their coefficients vary more across the samples. "Plan Code" shows a gradual increase in importance from Sample 1 to Sample 3, indicating its evolving influence in different contexts. "Ri Sum At Risk" has slightly higher importance in the first two samples compared to the third, suggesting its predictive value may vary depending on the data.

The remaining features, such as "Installment Premium," "Insured Age," "Policyholder Occupation," and "Policyholder Age," exhibit significantly lower coefficients, indicating a smaller impact on the model's predictions. Their low importance suggests they may contribute less to the overall predictive accuracy or that their effects are less distinct across the datasets. These insights can guide further feature selection and model optimization by focusing on the most influential features.

Given that the Random Forest model has demonstrated superior performance among all the algorithms, we will focus our discussion on its results in this section. These insights hold particular relevance when derived from a model known for its accuracy and reliability. Our research's primary aim is to pinpoint the most effective machine learning model for the given problem. By showcasing the best model, we directly fulfill this objective, offering findings that are not only pertinent but also actionable. This approach ensures that our analysis is streamlined, emphasizing the significance of the model that has proven to be the most robust and dependable

The performance metrics of each sample is tabulate in Table IV:

TABLE IV. Performance METRIC of Random Forest

Sample	Target	Accuracy	Precision	Recall	F1 Score
1	0	0.92	0.98	0.87	0.92
	1		0.95	0.97	0.91
2	0	0.92	0.98	0.87	0.92
	1		0.96	0.98	0.91
3	0	0.91	0.97	0.87	0.91
	1		0.95	0.96	0.90

The results of the Random Forest model, as presented in the table, illustrate its overall performance across different samples and target classes. The key metrics used to evaluate the model include accuracy, precision, recall, and F1 score, providing a comprehensive view of its effectiveness.

Across all samples, the model demonstrates high accuracy, consistently around 0.91 to 0.92, indicating that a significant proportion of predictions were correct. This high level of accuracy suggests that the model performs well in terms of overall prediction correctness.

When examining precision, which measures the proportion of correct positive predictions out of all predicted positives, the scores range from 0.95 to 0.98 for the majority class (target 0), indicating a very low rate of false positives. Similarly, precision for the minority class (target 1) is also high, with scores between 0.95 and 0.98, demonstrating the model's ability to avoid incorrect positive classifications.

Recall, which represents the proportion of actual positives correctly identified, is slightly lower than precision, particularly for the majority class (target 0), with scores between 0.87 and 0.88. This lower recall indicates that while the model has high precision, it occasionally misses some actual positives. However, the recall for the minority class (target 1) is notably higher, with scores between 0.90 and 0.98, reflecting the model's ability to identify most positive cases in this category.

The F1 score, the harmonic mean of precision and recall, offers a balanced perspective on the model's performance. For the majority class, the F1 score is around 0.91 to 0.92, suggesting a reasonable balance between precision and recall. For the minority class, the F1 score is slightly lower, ranging from 0.90 to 0.91, indicating that while precision is high, recall could be improved for more balanced performance.

Overall, compared to other four (4) algorithms to Random Forest, it was found that the Random Forest model performs more reliably, with high accuracy and precision across all the three (3) samples from the total dataset. The relatively lower recall and F1 scores for some cases point to areas for further tuning and improvement, particularly in identifying a greater proportion of actual positives without compromising precision. This insight can guide future adjustments to the model, focusing on enhancing recall while maintaining high precision.

VII. Discussion

In the context of the insurance industry, each of the features listed can significantly impact the likelihood of an early claim.

The benefit component refers to the type and extent of coverage provided by the insurance policy. Different benefit components have varying risk profiles. Policies offering higher or more comprehensive benefits may attract individuals who anticipate a higher likelihood of claiming soon after policy inception, thus indicating a higher risk of early claims.

Plan Code is an identifier for different insurance plans offered by the company. Certain plans may have been designed for different risk profiles. For example, plans with lower premiums might attract higher-risk customers or those with a higher propensity to claim early. The specific terms and conditions associated with each plan code can also influence early claim likelihood.

Sum Insured is the amount the insurance company would have to pay if a claim is made. Policies with higher sums at risk may be more likely to result in early claims as policyholders might be more motivated to claim early to secure a large payout. Additionally, larger coverage amounts can be indicative of higher risk individuals or those with greater financial needs.

Installment Premium is the periodic payment made by the policyholder to keep the insurance policy active. The premium amount can reflect the risk level assigned to the policyholder. Higher premiums might be associated with higher-risk individuals who are more likely to make early claims. Conversely, lower premiums might attract cost-conscious individuals, potentially leading to different risk profiles.

Insured Age is the age of the person who is covered by the insurance policy. Age is a critical factor in assessing risk. Younger insured individuals might be perceived as lower risk for certain types of policies (e.g., life insurance), but they might claim early for specific reasons like accidents. Conversely, older individuals might be seen as higher risk for health-related claims, including early claims due to pre-existing conditions or health issues.

Policyholder Occupation is the job or profession of the person who holds the insurance policy. Certain occupations are associated with higher risks (e.g., manual labor, construction) and may be more prone to early claims due to accidents or job-related health issues. Occupation can also indicate socioeconomic status, which might correlate with the likelihood of early claims.

Policyholder Age is the age of the person who owns the insurance policy, who may or may not be the same as the insured individual. The policyholder's age can provide insights into their financial planning stage and risk behavior. Younger policyholders might be more cautious and less likely to claim early, whereas older policyholders might have different financial pressures and health concerns that could lead to early claims.

Each feature offers unique information about the risk profile of the policyholder or the insured, helping the model make accurate predictions. These features had reduced variance in the decision trees by creating more homogeneous groups in terms of early claim likelihood. Besides, these features have strong correlations with early claims based on historical data, thus improving the model’s accuracy.

A Random Forest model evaluates the importance of features based on how effectively they split the data to reduce impurity at each node. The features mentioned are likely important because they provide significant information that helps in distinguishing between policyholders who are likely to make early claims and those who are not.

Understanding why these features are important can help insurance companies in risk assessment, pricing strategies, and designing policies that better manage and mitigate risks associated with early claims.

This model's usefulness is confirmed by its effectiveness in the insurance domain, proving its relevance in real-world applications. It has been validated as a beneficial tool to reduce potential financial losses. By mitigating risks, the model offers significant value to stakeholders seeking to safeguard their economic interests.

VIII. Conclusion

The approach outlined in this research offers a practical solution to the challenge of feature selection in the context of early claim detection in the life insurance industry. By focusing on the most relevant features, this approach allows insurance companies to detect and mitigate fraudulent claims more effectively. This, in turn, can lead to significant benefits, including reduced financial risk, enhanced operational performance, and increased customer trust.

Several areas for future research are worth exploring. Given the real-world nature of insurance data, missing data is a common challenge, and advanced imputation techniques could further enhance model performance. Additionally, the significant class imbalance observed in our dataset suggests that advanced methods for handling imbalanced data could improve the robustness and reliability of predictive models. Lastly, as machine learning models become more complex, there's a growing need for approaches that improve model interpretability, allowing insurance professionals to understand and trust the decisions made by these models.

To implement this approach in real-world settings, we recommend beginning with a pilot test to evaluate its impact on existing workflows and claims detection accuracy. Successful integration with existing systems is critical, so it's essential to ensure that the implementation does not disrupt current operations. Additionally, continuous monitoring and adjustment of the model are necessary to maintain optimal performance, as industry trends and regulatory requirements can evolve over time.

In summary, the feature selection approach described in this research has the potential to transform early claim detection in the life insurance industry. By implementing this approach and addressing the challenges outlined, insurance companies can better manage risk, streamline operations, and ultimately deliver a higher level of service to their customers.

Acknowledgment

We would like to convey our heartfelt appreciation to Tunku Abdul Rahman University of Management and Technology for their important assistance and advice during this research effort. We value the insurance partner for their domain knowledge and masked dataset on top of the industry problem statement provided to the research team. We also appreciate the reviewers' critical remarks and the opportunity to present our work. Finally, we are thankful to the individuals and institutions that have helped to promote data science and its uses in the insurance industry.

References

[1] Tennyson, S. (2008). Moral, social, and economic dimensions of insurance claims fraud. *Social Research*, 75(4), 1181–1204.

[2] Picard, P. (2000). Economic analysis of insurance fraud. In G. Dionne (Ed.), *Handbook of Insurance* (Vol. 22). Huebner International Series on Risk, Insurance, and Economic Security. Springer, Dordrecht.

[3] A. Kilroy and K. A. Smith, "Insurance Fraud Statistics 2024," Forbes Media LLC, Mar. 21, 2024.

[4] M. D. J. Chudgar and A. K. Asthana, "Life Insurance Fraud – Risk Management and Fraud Prevention," *International Journal of Marketing, Financial Services & Management Research*, vol. 2, no. 5, pp. 67-78, May 2013.

[5] P. Choudhary, "Life Insurance Frauds in India: Reasons, Impact and Prevention Mechanisms," *The Management Accountant*, vol. 49, no. 6, pp. 78-85, June 2014.

[6] A. Chancel, L. Bradier, A. Ly, R. Ionescu, L. Martin, and M. Sauce, "Applying machine learning to life insurance: some knowledge sharing to master it," *arXiv*, no. 2209.02057, September 2022.

[7] S. Rawat, A. Rawat, D. Kumar, and A. S. Sabitha, "Application of machine learning and data visualization techniques for decision support in the insurance sector," *Journal of Industrial Information Integration*, vol. 25, pp. 100012, 2021.

[8] M. K. Severino and Y. Peng, "Machine learning algorithms for fraud prediction in property insurance: empirical evidence using real-world microdata," *Machine Learning with Applications*, vol. 3, pp. 100074, 2020.

[9] J. M. Maisog, W. Li, Y. Xu, B. Hurley, H. Shah, R. Lemberg, T. Borden, S. Bandeian, M. Schline, R. Cross, A. Spiro, R. Michael, and A. Gutfraind, "Using massive health insurance claims data to predict very high-cost claimants: a machine learning approach," *arXiv*, no. 1912.13032, December 2019.

[10] K. Kaushik, A. Bhardwaj, A. D. Dwivedi, and R. Singh, "Machine learning-based regression framework to predict health insurance premiums," *International Journal of Environmental Research and Public Health*, vol. 19, no. 13, pp. 7898, July 2022.

[11] A. Groll, C. Wasserfuhr, and L. Zeldin, "Churn modeling of life insurance policies via statistical and machine learning methods: analysis of important features," *arXiv*, no. 2202.09182, February 2022.

[12] M. Azzone, E. Barucci, G. G. Moncayo, and D. Marazzina, "A machine learning model for lapse prediction in life insurance contracts," *Expert Systems with Applications*, vol. 188, pp. 116261, March 2022.

[13] H. M. and M. R., "Machine learning approaches for auto insurance big data," *Risks*, vol. 9, no. 2, pp. 42, February 2021.

[14] S. Baran and P. Rola, "Prediction of motor insurance claims occurrence as an imbalanced machine learning problem," *arXiv*, no. 2204.06109, April 2022.

[15] N. K. Yego, J. Kasozi, and J. Nkurunziza, "A comparative analysis of machine learning models for the prediction of insurance uptake in Kenya," *Data*, vol. 6, no. 11, pp. 116, November 2021.

[16] V. Ramachandran, A. R. Kavitha, and R. Pandimeena, "An accurate prediction of medical insurance cost using forest regression algorithms," in *IEEE International Conference on Data Science and Artificial Intelligence*, 2023, pp. 10452541.

[17] X.-Y. Liu, J. Wu, and Z.-H. Zhou, "Exploratory undersampling for class-imbalance learning," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 39, no. 2, pp. 539–550, Dec. 2008.

[18] J. Berrevoets, F. Imrie, T. Kyono, J. Jordon, and M. van der Schaar, "To impute or not to impute? Missing data in treatment effect estimation," in *Proc. 26th Int. Conf. Artif. Intell. Stat.*, 2023, pp. 3568–3590.

[19] A. C. Gottschall, S. G. West, and C. K. Enders, "A comparison of item-level and scale-level multiple imputation for questionnaire batteries," *Multivariate Behavioral Research*, vol. 47, no. 1, pp. 1–25, Feb. 2012

[20] V. Picheny, D. Ginsbourger, Y. Richet, and G. Caplin, "Quantile-based optimization of noisy computer experiments with tunable precision," *Technometrics*, vol. 55, no. 1, pp. 2–13, 2013

[21] A. Orlenko and J. H. Moore, "A comparison of methods for interpreting random forest models of genetic association in the presence of non-additive interactions," *BioData Mining*, vol. 14, no. 9, pp. 1–17, 2021.

[22] S. Ibrahim, S. Nazir, and S. A. Velastin, "Feature selection using correlation analysis and principal component analysis for accurate breast cancer diagnosis," *Journal of Imaging*, vol. 7, no. 11, pp. 225, Oct. 2021.

[23] A. Sharma and S. K. Nair, "Category ratio: A search for an optimal solution to reduce choice overload," *Journal of Consumer Behaviour*, vol. 22, no. 6, pp. 1263–1278, May 2023.

[24] Y. Mu, K. Bontcheva, and N. Aletras, "It's about time: Rethinking evaluation on rumor detection benchmarks using chronological splits," in *Proc. 2023 Conf. Rumor Detection Benchmarks*, 2023.

[25] G. Canbek, S. Sagiroglu, T. Taskaya Temizel, and N. Baykal, "Binary classification performance measures/metrics: A comprehensive visualized roadmap to gain new insights," *IEEE Access*, vol. 5, pp. 3043–3058, Nov. 2017.

[26] A. Y.-C. Liu, "The Effect of Oversampling and Undersampling on Classifying Imbalanced Text Datasets," B.S. thesis, 2004.

[27] H. Shamsudin, U. K. Yusof, A. Jayalakshmi, and M. N. A. Khalid, "Combining oversampling and undersampling techniques for imbalanced classification: A comparative study using credit card fraudulent transaction dataset," in *2020 IEEE 16th International Conference on Control & Automation (ICCA)*, Singapore, 2020, pp. 803-808.

[28] P. S. Singh, V. P. Singh, M. K. Pandey, et al., "Enhanced classification of hyperspectral images using improvised oversampling and undersampling techniques," *Int. J. Inf. Tecnol.*, vol. 14, pp. 389–396, 2022.

[29] Zhuoyuan Zheng, Yunpeng Cai, Ye Li, "Oversampling Method for Imbalanced Classification," *Computing and Informatics*, vol. 34, pp. 1017-1037, 2015.

[30] A. Gosain and S. Sardana, "Handling class imbalance problem using oversampling techniques: A review," in *2017 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, Udupi, India, 2017, pp. 79-85.

[31] X. -Y. Liu, J. Wu, and Z. -H. Zhou, "Exploratory Undersampling for Class-Imbalance Learning," in *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 39, no. 2, pp. 539-550, April 2009.

[32] A. Dal Pozzolo, O. Caelen, and G. Bontempi, "When is Undersampling Effective in Unbalanced Classification Tasks?," in *Machine Learning and Knowledge Discovery in Databases. ECML PKDD 2015*, Lecture Notes in Computer Science, vol. 9284, Springer.

[33] Pavithrakannan, R., Fenn, N. B., Raman, S., & Kalyanara, V. (2021). Imputation Analysis of Central Tendencies for Classification. IEEE, April 2021.

[34] Palanivinayagam, A., & Damaševičius, R. (2023). Effective Handling of Missing Values in Datasets for Classification Using Machine Learning Methods. Information, 14(2), 92.

[35] Menze, B.H., Kelm, B.M., Masuch, R. et al. "A comparison of random forest and its Gini importance with standard chemometric methods for the feature selection and classification of spectral data." *BMC Bioinformatics*, 10, 213 (2009).

[36] Alsagri, H.S., & Ykhlef, M. "Quantifying Feature Importance for Detecting Depression using Random Forest." *International Journal of Advanced Computer Science and Applications*, 11.

[37] S. Gharsalli, B. Emile, H. Laurent, X. Desquesnes and D. Vivet, "Random forest-based feature selection for emotion recognition," in *2015 International Conference on Image Processing Theory, Tools and Applications (IPTA)*, Orleans, France, 2015, pp. 268-272.

[38] A. S. M. Shafi, M. M. I. Molla, J. J. Jui, and M. M. Rahman, "Detection of colon cancer based on microarray dataset using machine learning as a feature selection and classification techniques," SN Applied Sciences, vol. 2, no. 7, July 2020.

[39] N. M. Shahani, M. Kamran, X. Zheng, C. Liu, and X. Guo, "Application of Gradient Boosting Machine Learning Algorithms to Predict Uniaxial Compressive Strength of Soft Sedimentary Rocks at Thar Coalfield," *Advances in Civil Engineering*, November 2021.

[40] Y. Liu, S. Yong, C. He, and X. Wang, "An Earthquake Forecast Model Based on Multi-Station PCA Algorithm," March 2022.

[41] Y. Wang, Z. Pan, J. Zheng, L. Qian, and L. Mingtao, "A hybrid ensemble method for pulsar candidate classification," *Astrophysics and Space Science*, vol. 364, no. 8, August 2019.

[42] A. Adam, M. I. Shapiai, L. C. Chew, and Z. Ibrahim, "A Two-Step Supervised Learning Artificial Neural Network for Imbalanced Dataset Problems," January 2010.