IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Date of publication xxxx 00, 0000, date of current version xxxx 00, 0000.

Digital Object Identifier 10.1109/ACCESS.2022.Doi Number

Machine Learning based Method for Insurance Fraud Detection on Class Imbalance Datasets with Missing Values

AHMED A. KHALIL[1,2], ZAIMING LIU[1], AHMED FATHALLA[3], AHMED ALI[4,5] and, AHMAD SALAH[6,7,8]

[1]School of Mathematics and Statistics, Central South University, Changsha, Hunan, China (e-mail: Ahmedrohim91@aun.edu.eg) (math_lzm@csu.edu.cn)

[2]Faculty of Commerce, Assiut University, Assiut, Egypt.

[3]Department of Mathematics, Faculty of Science, Suez Canal University, Ismailia, Egypt (e-mail: fathalla_sci@science.suez.edu.eg)

[4]Department of Computer Science, College of Computer Engineering and Sciences, Prince Sattam Bin Abdulaziz University, Al-Kharj, Saudi Arabia (e-mail: a.abdalrahman@psau.edu.sa).

[5]Higher Future Institute for Specialized Technological Studies, Cairo 3044, Egypt.

[6]College of Computing and Information Sciences, University of Technology and Applied Sciences, Ibri, Oman (e-mail: ahmad.salah@utas.edu.om).

[7]AI Innovation and Research Center, University of Technology and Applied Sciences, Muscat, Oman.

[8]Department of Computer Science, Faculty of Computers and Informatics, Zagazig University, Zagazig, Sharkeya, Egypt (e-mail: ahmad@zu.edu.eg).

Corresponding author: Ahmed A. Khalil (e-mail: Ahmedrohim91@aun.edu.eg). Zaiming Liu (e-mail: math_lzm@csu.edu.cn).

ABSTRACT Insurance fraud is a prevalent issue that insurance companies must face, particularly in the realm of automobile insurance. This type of fraud has significant cost implications for insurance firms and can have a long-term impact on pricing strategies and insurance rates. As a result, accurately predicting and detecting insurance fraud has become a crucial challenge for insurers. The fraud datasets are usually imbalanced, as the number of fraudulent instances is much less than the ligament instances and contains missing values. Prior research has employed machine learning methods to address this class imbalance dataset problem, but there is limited effort handling the class imbalance dataset present in insurance fraud datasets. Moreover, we could not find an overfitting analysis for the relevant predictive models. This paper addresses these two limitations by employing two car insurance company datasets, namely, an Egyptian real-life dataset and a standard dataset. We proposed addressing the missing data and the class imbalance problems with different methods. Then, the predictive models were trained on processed datasets to predict insurance fraud as a classification problem. The classifiers are evaluated on several evaluation metrics. Moreover, we proposed the first overfitting analysis for insurance fraud classifiers, to our knowledge. The obtained results outline that addressing the class imbalance in the insurance fraud detection dataset has a significant positive effect on the performance of the predictive model, while addressing the problem of missing values has a slight effect. Moreover, the proposed methods outperform all of the existing methods on the accuracy metric.

INDEX TERMS Data imputation, Ensemble learning, Imbalanced data, Insurance fraud, Machine learning, Missing Data handling, Overfitting, Predictive models, Resampling methods.

I. INTRODUCTION

Insurance frauds are getting more complex in recent times, posing a significant difficulty for insurance firms to detect them. The rapid development of contemporary technology and worldwide communication has resulted in a substantial increase in fraudulent activities, leading to the loss of billions of dollars on an international level annually. Insurance firms have been significantly affected by fraudulent operations, which have garnered considerable attention in recent times. Insurance fraud has become an extensive problem, resulting in substantial financial losses for insurance firms and society as a whole [1], [2]. This kind of fraudulent activity constitutes a significant proportion of the expenses incurred by insurance firms, hence impacting their profitability, pricing tactics, and overall socioeconomic advantages in the future.

Currently, insurance companies have inadequate risk control and fraud detection abilities. Insurance fraud results in not only temporary losses but also poses a severe threat to the growth of insurance companies. As fraudulent claims

continue to accumulate, bonuses meant for legal claims, and new business financial support, the solvency of insurance companies declines [3]. Hence, insurance companies must address the problem of accurately identifying risk factors and minimizing losses caused by fraudulent claims.

The identification of fraudulent activities in insurance sectors such as auto insurance or medical insurance has become extremely crucial to lowering the expenses of insurance firms. Most insurance companies rely on the expertise of professionals to detect such fraud. Although knowledge derived from experience is easy to understand and can be reused, its implementation in a straightforward manner often results in some degree of incorrect assessment. Therefore, several approaches for detecting fraud in the different insurance sectors have been suggested. Statistical methods and artificial intelligence (AI) techniques (such as machine learning (ML) and ensemble learning) offer valuable tools for the detection of insurance fraud and have demonstrated successful applications in this domain [4].

Ensemble learning classification techniques are highly regarded as a significant research area in pattern recognition. Due to their impressive performance, these ensemble methods have attracted considerable interest in the fields of pattern recognition [5], [6], [7], [8], [9], [10]. Although several ensemble methods have been employed in detecting fraud in general, there has been comparatively less application of these methods in detecting fraud [11], [12].

Insurance data presents several challenges that can impact data analysis and decision-making processes. In real-world insurance fraud datasets, researchers and data scientists often encounter several challenges that can impact the quality and effectiveness of their models [13], [14]. One of the primary challenges is data quality, as insurance datasets often contain missing or inaccurate information due to manual entry errors, high-dimensional data, outdated records, or inconsistencies across different data sources. Furthermore, imbalanced data is a significant challenge in the insurance industry. Thus, these problems in fraud insurance data have significant impacts on the accuracy and reliability of fraud detection models. Missing values can introduce biases and inaccuracies, hindering the model's ability to learn patterns. Inaccurate assessment of features' importance may result in models giving undue weight to irrelevant variables or overlooking critical fraud indicators. Imbalanced data poses a risk of biased models that excel in predicting the majority class but struggle to identify the minority class, leading to an increased risk of false negatives in fraud detection [12], [13], [15].

Moreover, the availability of datasets for the public to study fraud detection in this field is limited [2]. This research relies on real data provided by insurance companies in the Egyptian auto insurance market. Thus, addressing these challenges through appropriate data preprocessing and modeling techniques is crucial for building robust and effective fraud detection systems in the insurance industry. To our knowledge, efforts to address the class imbalance data and missing values of insurance fraud detection are limited. This shortcoming of the existing methods motivated the current work.

According to all the aforementioned effects, insurance companies have become very interested in the detection of insurance fraud and have been seriously taken into account recently. As a result, there has been a growing interest in insurance fraud data for fraud detection from insurance companies that are looking to apply modern technologies and intelligent systems to cope with these problems rather than depending exclusively on human experts to tackle the difficulties linked to fraudulent operations [16], [17]. Insurers seek to adopt these technologies to bolster their ability to detect fraud, boost precision, and maintain an advantage over ever-changing fraudulent methods. Ultimately, this will lead to more efficient and proactive approaches to combating insurance fraud. Thus, researchers need to carefully consider these factors to build robust fraud detection models in the insurance industry.

The objective of this study is to assess the overall effectiveness, specifically in terms of prediction accuracy, of classic ML and ensemble models. This evaluation will involve implementing strategies for dealing with various challenges commonly encountered in insurance fraud detection, such as missing data, feature importance, and class imbalance datasets. The work aims to address these crucial elements to bridge a gap in the current body of knowledge and provide a valuable contribution to the development of effective and efficient approaches for dealing with fraud, particularly in the field of auto insurance fraud.

Based on surveying literature studies, the main contributions of this paper lie in providing insights and empirical evidence on the efficiency of predictive models when confronted with real-world data challenges in insurance fraud detection. Therefore, the most significant contributions of this study can be concisely outlined as follows:

1. The proposed system explores diverse techniques to tackle the missing data and data imbalance challenges of a real dataset of auto insurance companies in Egypt, as data imbalance is a common problem in most of the

insurance fraud datasets. In addition, the collected dataset is made publicly available.

2. The study tackles the issue of missing data within the dataset, which diminishes the model's capacity to accurately comprehend the underlying patterns by utilizing two different data imputation methods. Then, the two methods are compared to determine the most effective method for tackling the problem.
3. In this study, we proposed using four distinct methodologies to address the problem of class imbalance datasets to improve the insurance fraud detection system and develop predictive models that can achieve higher levels of accuracy in fraud detection. To our knowledge, the first detailed overfitting analysis for detecting fraud transactions in insurance datasets was conducted in this study.
4. To thoroughly evaluate the proposed system, we utilized a public car insurance "Oracle" dataset on Kaggle. The model's performance was evaluated using various metrics, which were then compared to state-of-the-art methods' results from other studies that employed this same dataset. The comparison revealed that the proposed methodology outperformed the other methods of comparison.

The remainder of this work is outlined as follows: A brief introduction about insurance fraud and the prior studies are given in Section 2. Section 3 provides a detailed account of the research material, including data collection, the research design, and the experimental setup. Section 4 describes the methodology and the specific techniques and approaches that are employed to detect insurance fraud in the study and evaluation metrics. The findings are presented and analyzed in Section 5. Finally, we present the conclusion in Section 6.

II. Background and Related works

The International Association of Insurance Supervisors has established the definition of insurance fraud in the literature as "the act or omission of the fraudster or to other parties to obtain the dishonest benefit of that fraudster" (2007, P. 2). This happens if insider dealing occurs or if an asset is misappropriated, intentionally misrepresented, or information is omitted or not revealed, or if the details related to financial decisions or operations and liability abuse, fiduciary relationships or trust relationships have been abused. Moreover, insurance fraud has been split into four categories in the insurance literature, internal fraud, policyholder fraud, broker fraud, and insurer fraud [18]. In this paper, we will focus on policyholder fraud based on the available data.

A. INSURANCE FRAUD

Fraud detection using conventional methods has become very difficult due to the large volume of data. In addition, the development of new technologies is making fraud increase rapidly. Thus, the need for techniques to detect fraud with a high level of accuracy is becoming imperative. AI techniques play a significant role in identifying insurance fraud by analyzing large amounts of data to uncover hidden truths. Data mining involves discovering reliable statistical insights that were previously unknown. Many studies have attempted to detect insurance fraud using AI methods such as ML algorithms, such as Naive Bayesian (NB), random forest (RF), logistic regression (LR), support vector machine (SVM), decision tree (DT), AdaBoost, and neural network (NN) models, which have been used for fraud detection. Some studies show that RF and DT algorithms perform better than other methods for detecting fraud in automobile [4], [15], [19], [20], [21], [22], [23], [24], [25], [26], [27]. Some studies implemented ensemble models to recognize insurance fraud, such as Bagged Ensemble Convolutional Neural Networks in [28], and deep boosting decision trees in [29]. Unsupervised anomaly detection models, such as Spectral Ranking Anomaly (SRA) systems, have also been proposed for detecting fraudulent instances. A hybrid approach using genetic algorithm (GA) based Fuzzy C-Means (FCM) clustering and various supervised classifier models has also been proposed to detect fraud in automobile insurance claims, and the proposed system's efficacy was demonstrated on a real-world automobile insurance dataset [30].

For example, Wongpanti, R & Vittayakorn, S in [31] present a method for detecting auto insurance fraud using a one-dimensional Convolutional Neural Network (1D-CNN). To address data imbalance in fraud detection, the authors use data augmentation techniques like SMOTE and CTGAN and apply Focal Loss to improve classification accuracy for minority classes. This approach enhances the model's ability to detect fraudulent claims, aiming to reduce financial losses in the auto insurance industry. Nordin, et al., in [32] evaluate various models for predicting automobile insurance fraud, comparing classical methods like logistic regression with machine learning models such as neural networks, SVM, decision trees, random forests, and AdaBoost. The tree-augmented naïve Bayes (TAN) model outperforms others, achieving the highest accuracy and sensitivity. The study emphasizes the effectiveness of machine learning in detecting fraudulent claims and suggests improvements in data preparation and model settings for better performance.

Aiemsuwan, P & Srikamdee, S in [33] introduce a hybrid method combining resampling and backward elimination to

improve fraud detection in automobile insurance. By addressing data imbalance, the method enhances the performance of machine learning models like RF and XGBoost. The study demonstrates that this approach significantly boosts accuracy, achieving high F1-scores across multiple datasets, making it effective for detecting fraudulent claims in insurance and other fields. Overall, the results suggest that using sophisticated machine learning techniques can improve the accuracy and speed of detecting insurance fraud.

Statistical approaches have been used to identify instances of insurance fraud. These methods identify possibly abnormal patterns by carefully analyzing the statistical characteristics of insurance customer data. Statistical models are used to detect outlier events by applying specified thresholds or specific criteria. Notable statistical approaches include descriptive statistics, hypothesis testing, and the examination of temporal trends using time series analysis. For example, Badriyah et al., in [34] enhance fraud detection using anomaly detection algorithms, specifically Nearest Neighbor-based methods and Statistical Methods with the interquartile range. The study compares its performance results with those of previous researchers who utilized the same dataset. The experimental findings indicate that the methods employed in this study exhibit superior performance in certain cases, thereby contributing to the advancement of fraud detection capabilities. Lee and Kim in [35] proposed the use of traditional time-series analysis methods, Seasonal Autoregressive Integrated Moving Average (SARIMA) and Seasonal-Trend decomposition using LOESS (STL), for detecting diverse anomalies, including those in noisy and non-periodic data. The combination of SARIMA and STL is shown to achieve high accuracy in anomaly detection. The results highlight the proposed algorithm's effectiveness in various scenarios.

B. Data imputation and resampling methods

On the other side, real-life fraud data always includes common problems such as missing data, and imbalanced data. Therefore, several studies tried to address these problems in various fields, for instance, P. Wang & Chen in [36] introduce a new method to handle missing data by utilizing a three-way ensemble technique. This method entails grouping objects that do not have missing values and filling in the missing values by utilizing the average attributes of each group. The algorithm's performance was proven by experimental findings on datasets from the UCI machine learning repository. Nevertheless, like numerous other methods, the study failed to consider a strategy for handling missing values. Rusdah & Murfi in [37] conducted a study to evaluate the efficacy of the Extreme Gradient Boosting (XGBoost) model in managing missing values for risk prediction in the life insurance industry. Results from the simulations demonstrate that XGBoost models with and without imputation preprocessing achieve similar levels of accuracy. Jadhav et al., in [38] examine seven different imputation techniques using five small numerical datasets. Even though they talk about many missingness patterns, the authors don't say which one they employed in their tests.

The results show that KNN imputation performs best once again. For the imbalance data problem, Sundarkumar et al., in [39] used only one resampling method, namely Random under-sampler, with Probabilistic Neural Network (PNN), DT, SVM, Logistic Regression (LR), and Group Method of Data Handling (GMDH) models. The results of their study showed that the Decision Tree model was the most effective one for fraud detection. Similarly, Hassan & Abraham in [40] used Random under-sampler with DT, NN, and SVM models, and found that the DT model performed the best. Another study by Prasasti et al., in [41] used two resampling methods (i.e., Random under-sampler and Synthetic Minority Over-sampling Technique (SMOTE)) with DT, MLP, and RF models, and found that the RF model was the most effective. Maina et al., in [42] proposed an XGBoost-based model that oversamples class imbalance using SMOTE. The findings show that XGBoost performs well with SMOTE compared to imbalanced training datasets and other techniques.

Based on a comprehensive analysis of available research, it becomes apparent that there is a gap in research addressing multiple challenges, including the treatment of missing data and the imbalanced issue within insurance datasets. Furthermore, the application of these methodologies in resolving fraud issues within the insurance sector. Consequently, it is vital to bridge this research gap by presenting an integrated system that tackles these challenges through diverse scenarios.

III. MATERIAL

A. DATASET OVERVIEW

The main dataset introduced in this study is obtained from an Egyptian car insurance company, wherein the validity of fraud claims has been verified by the competent authorities within the insurance company. The dataset was anonymized by the company by replacing the feature names with code of Xi, where i represents the feature number. The dataset was collected by including all claims reported in the year 202x. The dataset contains a total of 1000 car insurance claims, of which 217 claims have been classified as fraudulent with a percentage of 21.7% and 783 as non-fraudulent with a

percentage of 78.3%, thereby indicating a considerable imbalance in the data. Each claim within the dataset is characterized by 22 distinct features, inclusive of insured personal data (e.g., age and gender), insurance contract particulars (e.g., coverage, and premiums), accident information (e.g., accident severity, crash type, and total claim amount), and a binary "fraud" feature (which serves as the target variable for prediction). The features of the dataset are listed in Table 1.

To evaluate the proposed model's validity, we apply it to the "Oracle" dataset available on Kaggle. We utilize the proposed framework to identify instances of insurance fraud in the auto insurance industry. The specific attribute we focus on is the "fraud report" column. The dataset including information about vehicle insurance claims has a total of 15,420 instances. Out of these instances, 923 cases have been identified as fraudulent, which accounts for around 6% of the total. This indicates a significant imbalance in the distribution of the two classes within the dataset. Each claim in this dataset is characterized by 32 unique attributes, which are listed in Table 2.

TABLE 1. DATASET-1 FEATURE DESCRIPTION

No.	Features	No.	Features
X1	The insured's months	X12	The authorities contacted
X2	The age of the insured	X13	Accident vehicle count
X3	The gender	X14	The property damage
X4	The education level.	X15	Accident victims
X5	The insured's work.	X16	The number of witnesses.
X6	Insured deductible %.	X17	Police report available.
X7	Incident severity.	X18	The total claim amount
X8	Max insurance payout	X19	Injury claim amount
X9	Incident types	X20	Property claim amount
X10	The crash types	X21	Vehicle claim amount
X11	The annual insurance premium	X22	Fraud class (The target feature)

Feature importance analysis is used to find the influence of feature inputs on the outcome of the predictive learning model. It assists in comprehending the features or elements that make the most substantial impact on the classification decisions identified by the proposed model. Furthermore, it can guide data preprocessing and model optimization endeavors. From Fig. 1, we can deduce that accident severity is the most influential feature for predicting insurance fraud, followed by the amount of property claims and the policy premium. Other features such as gender, education level, and accident types also play a role, but to a lesser extent.

TABLE 2. DATASET-2 FEATURE DESCRIPTION

No.	Features	No.	Features
X1	Unique identifier for each insured	X17	The deductible amount of the insured.
X2	The month in which the accident occurred	X18	The driver rating
X3	The week in the month the accident occurred	X19	The days between policy purchase and accident
X4	The days of the week the accident occurred on	X20	The days between policy purchase and claim filed
X5	Vehicle brand	X21	The previous number of claims
X6	The area of the accident occurred	X22	The age of the vehicle at the time of the accident
X7	The day of the week the claim was filled	X23	The intervals of insured age
X8	The month of the year the claim was filled	X24	Police report or not
X9	The week of the month the claim was filled	X25	If there is a witness or not
X10	The insured's gender	X26	The agent who is handling the claim
X11	The insured's marital status	X27	The number of supplements
X12	The age of the insured	X28	insured address change or not
X13	The person responsible for the accident	X29	The number of vehicles involved in the accident
X14	Type of vehicle insurance policy	X30	The year of accident occurred
X15	The categorization of the vehicle	X31	The type of insurance coverage
X16	The price category for vehicles	X32	Fraud class (The target feature)

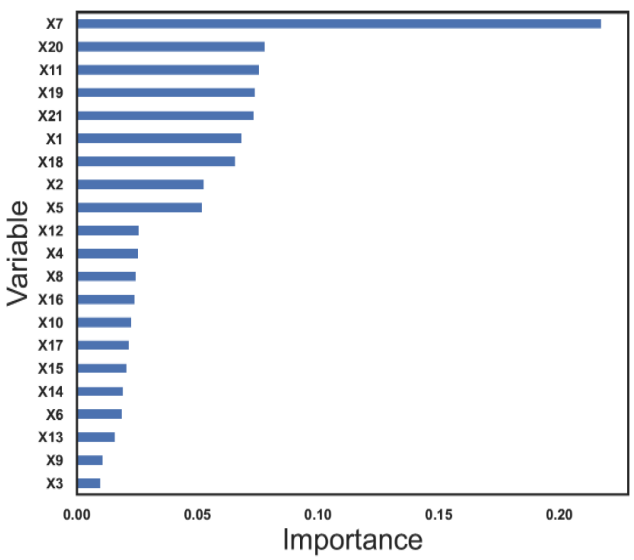

Figure 1. Feature Importance of Dataset-1 Features

To enhance clarity regarding the utilized dataset, the Pearson correlation coefficient is used to assess the pairwise correlation of the independent variable in the correlation study, as shown in Fig. 2. In other words, Fig. 2 illustrates the correlation between several features. This data can be utilized to identify the key characteristics that contribute to the discovery of insurance fraud. Correlation is less when the value is near zero and more when it is much greater. The correlation decreases as the value is near zero and vice versa. In Fig. 2, the target feature represents the outcome variable. With a correlation score of -0.8, the target variable is most linked with feature X7, as shown in Fig. 2. Following that, X21, X18, and X20 are the traits that are second most associated, with correlation values of 0.28, 0.21, and 0.18, respectively. Thus, the correlation analysis revealed a strong negative relationship between the outcome feature on one side and the X7 feature on the other side. Besides, there is a weak positive relationship with X21, X18, and X20 features.

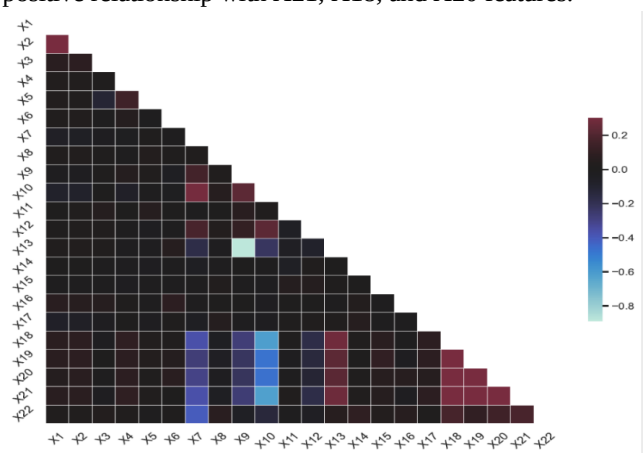

Figure 2. **Correlation Matrix of Dataset-1 Features**

B. THE PROPOSED FRAMEWORK

In this study, the proposed research framework for detecting insurance fraud includes a systematic approach that includes collecting and pre-processing data. This involves carefully cleaning and encoding the dataset, as well as implementing thorough procedures to address missing values. Two separate methodologies for data imputation are utilized. The first approach includes three separate methodologies utilizing artificial intelligence-driven imputation algorithms, whereas the second approach involves eliminating columns that contain missing values. Furthermore, the model confronts the common issue of imbalanced dataset by utilizing four distinct resampling techniques. The next step entails deploying classification models and subsequently conducting a thorough assessment of their performance metrics. Each of these consecutive phases is considered essential to the overall success of the suggested approach. Fig. 3 provides a visual representation of the procedures involved in the systematic identification of insurance fraud.

C. EXPERIMENTAL SETUP

The experiments were conducted on a computer with an Intel(R) Core (TM) i7-9750H CPU @ 2.60GHz and 16−GB for RAM. The operating system used is Windows 10 64-bit. Python is the language of choice for developing the framework's implementation. Moreover, the dataset is loaded using the Pandas [43] data frame. Machine learning models are implemented using Scikit Learn library [44]. In order to ensure the ability to replicate the experiments, parameter settings, and reported findings, we have made the source code, visualizations, and data of the proposed work publicly available through a GitHub link[1].

IV. METHODOLOGY

A. DATA PREPROCESSING

One of the most important steps in the application of ML approaches is data preprocessing, which is also illustrated in the initial phase of Fig. 3. Data must be processed before any future operations because the data may include multiple errors. Thus, this phase involves essential data processing tasks, such as imputing missing values, scoring data, and dividing the data into training and testing datasets.

1) Data cleaning and encoding

Data cleaning involves detecting, correcting, or eliminating inaccurate and corrupted information from datasets or databases, while also identifying missing or incomplete data and removing irrelevant information. This is done to improve the quality and efficiency of the data and to facilitate the identification of important elements during exploration. In addition, data cleaning can improve the results of machine learning models [45], [46]. In the proposed study, the dataset if processed by removing duplicate and incomplete claims to eliminate insignificant data. Another crucial preprocessing task is encoding data, which involves converting the raw data into a numerical format that can be analyzed by algorithms, models, and statistical methods. For example, the proposed system transformed categorical features into a numerical format, such as using "1" for "male" and "0" for "female" for the gender of the insured.

[1] https://github.com/stars-of-orion/Enhancing-Insurance-Fraud-Detection

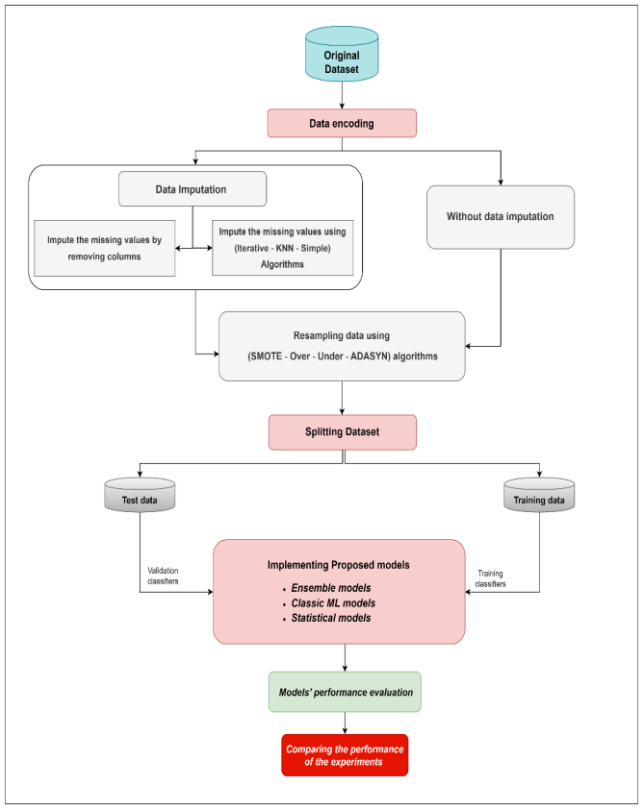

Figure 3. **The framework of Proposed Insurance Fraud Detection System**

2) Data imputation for missing values

Many reasons contribute to the prevalence of missing values in real-world datasets. When training ML algorithms, it's important to avoid datasets with a large number of missing values. Missing data or data entered by humans wrongly causes many instances of erroneous forecasts [47]. Researchers face difficulties, increased computational costs, and skewed results due to missing values in datasets [48].

In the insurance industry, inaccurate risk assessment, policy pricing, and claim processing can result from missing data in insurance records. Furthermore, insurers may be exposed to higher levels of risk if critical data points are missing from fraud detection systems, which could make it harder to identify suspicious patterns and behaviors [49]. For insurance datasets to be used for accurate risk assessment, fair pricing, quick claim processing, and accurate fraud detection, it is essential to manage and minimize the impact of missing values [37]. Typically, missing data rates < 1% are considered insignificant, whereas missing data ratios between 1-5% are considered adaptable. Advanced techniques are utilized to manage rates within the range of 5-15%. Rates over 15% have a significant impact on analysis [50]. For missing value problems, several imputation techniques have been developed. In this study, the missing values in the dataset are imputed by using two approaches, as follows:

- Addressing missing values by implementing ML-driven imputation models to handle the missing data in the study's dataset, namely, univariate imputation algorithm, multivariate imputation algorithm, and K-Nearest Neighbors algorithm [38], [49].
- Addressing missing values by eliminating columns that include missing values from the dataset. This approach is the most common method for dealing with missing values [51]. In the proposed work, we eliminated columns with a high rate of missing values for imputing the missing values.

Table 3 lists the columns/features with missing values in the datasets utilized for the research, as well as the descriptive statistics for that variable.

Table 3. MISSING VALUES PER FEATURE IN DATASETS

Dataset	Feature	Number of observations	Number of missing values	Missing value's rate
Dataset-1	X1	1,000	1	0.001%
	X10		178	17.8 %
	X14		360	36.0 %
	X17		343	34.3 %
Dataset-2	Days Policy Accident	15,420	55	0.35 %
	Age		319	2.06 %
	Past Number of Claims		4,351	28.2 %
	Number of Supplements		7,046	45.7 %

B. MODEL VALIDATION

In this study, two validation strategies were applied to evaluate the performance of the predictive models: Hold-out validation and K-fold cross-validation (CV). The K-fold cross-validation method was primarily used to estimate the generalization performance of the models. In K-fold CV, the dataset is divided into k equal subsets, or folds. The model is trained k times, each time using $k-1$ folds for training and the remaining fold for testing. This process ensures that every fold is used once for testing and helps in mitigating overfitting by providing a more reliable estimate of the model's performance across different subsets of the data.

Alternatively, the hold-out validation method involves splitting the dataset into two distinct subsets: a training set and a test set. In this study, 80% of the data was used for training the model, while the remaining 20% was reserved for testing its predictive accuracy. The model is trained using the training set, and its performance is subsequently evaluated using the test set, which acts as unseen data to assess the model's generalization capabilities. As illustrated in Figure 4, the K-fold cross-validation process is repeated for each fold $i = 1,2,\ldots,k$ where one-fold is used for validation and the remaining $k-1$ folds for training. This iterative process allows for a comprehensive assessment of the model's

performance across multiple splits of the dataset. Table 4 presents a comparative analysis of the prior probabilities among the training, testing, and original datasets, which is shown as follows:

Table 4. COMPARISON OF THE PRIOR TRAINING, TESTING, AND ORIGINAL DATASETS

Dataset	Number of samples	legitimate	fraud
Whole original data	1,000	783	217
Training data	800	632	168
Testing data	200	151	49

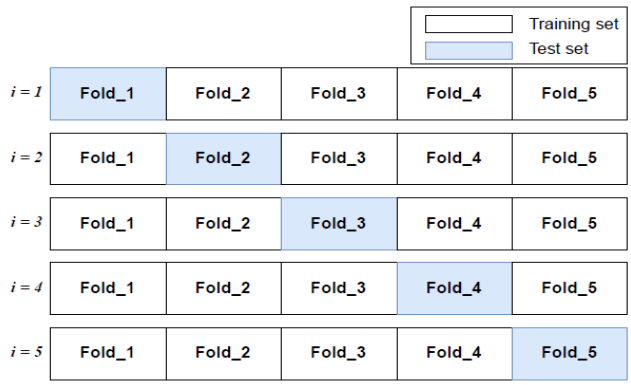

Figure 4. Cross-validation Method

C. RESAMPLING METHODS

The problem of imbalanced data is pervasive in many datasets, leading to biased classifier models that cannot make accurate predictions for minority classes [52]. This issue often arises in datasets where "ligament instances," which refer to the majority class examples that form the backbone of the data distribution, significantly outnumber the minority class. In the context of insurance fraud detection, these ligament instances represent the legitimate claims, which are far more numerous than the fraudulent ones. The databases used in this study, according to an analysis, are quite imbalanced, with 217 fraudulent claims and 783 legitimate claims, respectively, representing the two classes of insurance fraud. Consequently, addressing the issue of imbalanced data is imperative. Various methods have been developed to resolve this issue, with one of the most successful approaches involving the use of sampling-based techniques, such as random over-sampling and random under-sampling [53], [54].

- Random over-sampling method:

The approach under consideration is designed to enhance the weight of the minority class, and it is noteworthy that oversampling techniques are generally preferred over other alternatives. Random oversampling, which is based on bootstrapping and generates synthetic instances from the two groups' estimates of conditional density, is a commonly employed oversampling technique that supports binary classification tasks in the presence of imbalanced classes [55]. It can accommodate both categorical and continuous data, and it entails increasing the size of the dataset by replicating the original samples. A key argument in favor of random oversampling is that it does not generate new samples and preserves the variability of the samples.

- SMOTE method:

By combining two minority samples and one of their K nearest neighbors, SMOTE creates new minority samples. It is a statistical method that aims to increase the number of samples extracted from the minority class in a dataset by creating new instances. This algorithm utilizes the characteristics of the target category and its closest neighbors to generate new samples that combine the characteristics of a specific instance with those of its neighbors. Notably, newly created instances do not duplicate pre-existing minority samples [56].

- Random under-sampling method:

Random under sampling is a straightforward method for addressing imbalanced data that aims to balance the majority and minority classes. This technique involves randomly removing examples from the majority class in the training dataset, a process known as random under-sampling [57].

- Adaptive synthetic (ADASYN) method

The fundamental concept behind the ADASYN method is to employ a density distribution $\hat{r}_i$ as a parameter for determining the appropriate quantity of synthetic samples that should be created for each minority data instance. From a physical standpoint, $\hat{r}_i$ quantifies the weight distribution of minority class examples based on their learning difficulty [58].

D. The utilized classifier models

In the proposed work, the proposed system has integrated three distinct categories of classifier models to comprehensively address the complexity of the task at hand. First, we have incorporated statistical models, with Ridge Regression serving as a prominent representative. These models offer a solid foundation rooted in statistical principles, providing a rigorous framework for analysis and prediction. Moving beyond traditional statistical approaches, we have also leveraged classic ML models, which harness the power of algorithms to learn patterns and make predictions from data. Within this category, we have employed a diverse set of methodologies including Random Forest (RF), Support Vector Machine (SVM), K-Nearest Neighbours (KNN), and Logistic Regression (LR). Each of these models brings its unique strengths and characteristics, allowing for a comprehensive exploration of the data landscape. Additionally, recognizing the potential benefits of ensemble learning techniques in enhancing predictive performance, we

have incorporated ensemble learning models into our framework. Techniques such as Bagging, Boosting, and Stacking have been carefully integrated to harness the collective wisdom of multiple base learners, thereby improving the robustness and generalization capabilities of our model ensemble.

Of note, the efficiency of any ML model is intricately tied to the specific values assigned to its parameters. Thus, thorough parameter tuning, and optimization are essential steps in ensuring the optimal performance of our classifier ensemble. Through this comprehensive approach, we aim to develop a sophisticated and adaptable system capable of effectively addressing the challenges inherent in the domain of interest.

The models utilized in this study were configured with their default hyperparameters, except for the stacking ensemble model, which required to tune its parameters. The stacking model combined three base estimators Random Forest (RF), Logistic Regression (LR), and K-Nearest Neighbors (KNN) with an XGBoost (XGB) classifier serving as the final estimator. To ensure robust performance evaluation, we employed a 10-fold cross-validation technique throughout the analysis. The detailed hyperparameters for all the models used in this study are presented in Table 5, providing a comprehensive overview of the configurations applied to each model.

Table 5. THE DATASETS USED FOR EXPERIMENTATION CONFUSION MATRIX

Model	Parameters
Bagging	estimator= DecisionTreeClassifier, n_estimators=10, max_samples=1.0, max_features=1.0, bootstrap=True
Boosting	n_estimators=50, learning_rate=1.0, algorithm='SAMME.R'
Stacking	Base_estimators: {RandomForestClassifier, LogisticRegression, KNeighborsClassifier}. Meta-learner model: XGBClassifier.
Ridge regression	alpha=1.0
Random Forest (RF)	n_estimators=100, criterion='gini', min< samples_split=2, min_samples_leaf=1, max_features='sqrt', ootstrap=True
Gaussian NB	var_smoothing=1e-09
Logistic Regression (LR)	penalty='l2', tol=0.0001, C=1.0, max_iter=100

Table 5 presents the specific hyperparameters utilized for the various machine learning models in this study. Each model was configured to optimize its performance based on standard parameters or, in some cases, tailored settings. For example, the Bagging model utilized a Decision Tree classifier as its base estimator, with 10 estimators in total, and both the max_samples and max_features parameters were set to 1.0. The Boosting model applied 50 estimators with a learning rate of 1.0 and used the 'SAMME.R' algorithm. The Stacking model combined Random Forest, Logistic Regression, and K-Nearest Neighbors classifiers as base estimators, with the XGBoost classifier functioning as the final estimator. Other models, such as Ridge regression, were configured with an alpha value of 1.0, while the Random Forest model had 100 estimators and applied the 'gini' criterion. Gaussian Naive Bayes and Logistic Regression models were also customized with specific hyperparameters, such as smoothing for Gaussian NB and penalty type for LR. These parameter settings provided a balanced and tailored approach for each model in the experimentation process.

E. Experimental design

In this section, we describe the experimental design used to assess the effectiveness of various data preprocessing strategies for classification tasks using two distinct datasets. The experiments were designed to address common challenges found in real-world data, specifically missing values and class imbalance. Through multiple scenarios across three separate experiments, we systematically examined how different approaches, such as using raw data, applying various imputation techniques, and implementing methods to manage class imbalance, affect the performance of classification models. The aim of these experiments was to identify the most effective preprocessing strategies for enhancing model accuracy and reliability, providing valuable insights into best practices for preparing data for predictive modeling when dealing with incomplete or imbalanced datasets.

The experiments are designed as follows:

Experiment 1: In this experiment, we applied classification techniques to the raw data of datasets, which had missing values and imbalanced classes. We considered two scenarios for this analysis. In the initial case, the unprocessed data is utilized without any prior treatment for missing data or imbalanced distribution concerns. The data is inputted directly into classification models, and subsequently, the performance of the models is assessed. In scenario 2 of experiment 1, the raw data is handled by preprocessing the class imbalance issues without handling the missing values problem.

Experiment 2: In the second experiment, various ML models are used to impute the missing variables. The second experiment also has two more scenarios, the third and the fourth. In the third scenario, the utilized dataset is imputed with three different imputation methods, namely, KNN, Iterative, and Simple imputation. These three methods were applied before training the classification models. The fourth scenario extended this by not only imputing missing data using these methods but also addressing class imbalance in the dataset.

Experiment 3: The third experiment focused on handling missing data by removing columns with a high percentage of missing values (more than 15%). Based on the analysis in Table 3, three columns (X10, X14, and X17) were removed to reduce noise and improve model performance. This experiment included two scenarios: scenario 5 and scenario 6. In scenario 5, no additional steps were taken to address class imbalance; the focus remained solely on removing columns with significant missing data. In scenario 6, both missing data and class imbalance were addressed. Missing data was handled by removing columns with substantial missing values, similar to scenario 5, and class imbalance was managed using four different methods.

F. DISCUSSION ON THE EVALUATION METRICS OF CLASSIFICATION MODELS

Evaluation metrics play a vital role in selecting and comparing the best model as they measure the effectiveness of classifiers. Accuracy is a widely used metric that indicates the proportion of accurate predictions. A higher value of accuracy indicates better overall performance of the classifier. However, accuracy alone may not be reliable for classification problems, especially when dealing with imbalanced data [59], [60]. Car insurance claims are an excellent example of imbalanced data, as the majority of policyholders do not commit fraud. Therefore, relying solely on accuracy may result in bias toward the majority class and prejudice toward the minority class [61]. To address this issue, various measurement methods such as accuracy, specificity, precision, and F1-score are used. The evaluation metrics used in this study to compare various techniques are presented in Table 7. The confusion matrix (CM) is a valuable tool in machine learning and statistical analysis that helps evaluate the performance of a classification model. It provides a comprehensive summary of the model's predictions by comparing them with the actual outcomes. The matrix is organized into four quadrants, representing the four possible outcomes of a binary classification task: true positives (TP), true negatives (TN), false positives (FP), and false negatives (FN) as shown in Table 6. Thus, the confusion matrix (CM) and the receiver operating characteristic curve (ROC) metric are used to visualize the results and compare.

Table 6. THE DATASETS USED FOR EXPERIMENTATION CONFUSION MATRIX

	Actual legitimate (0)	Actual fraud (1)
Predicted legitimate (0)	True negative (TN)	False negative (FN)
Predicted fraud (1)	False positive (FP)	True positive (TP)

Table 7. PERFORMANCE EVALUATION METHOD

Evaluation metrics	Equation
Accuracy =	$\frac{TP+TN}{TP+FP+TN+FN}$
Specificity =	$\frac{TN}{FP+TN}$
Precision =	$\frac{TP}{TP+FP}$
F-measure =	$\frac{2\times TP}{2\times\ TP+FP+FN}$

V. RESULTS AND DISCUSSION

In this study, we conducted a thorough analysis of insurance fraud detection by executing five experiments. We employed several classifiers, imputation techniques for missing data, and resampling algorithms to develop a model that can detect fraud in the insurance industry datasets. This section analyzes the experimental results to determine the efficiency and performance of different predictive models and their interpretation.

For the overfitting analysis, there is no solid accepted value to be considered as a threshold between the training accuracy rate and test accuracy rate. But there is a relation between the difference in the training and test accuracy rates and the dataset size; the more the dataset is, the less the accepted value difference between the training and test accuracy rates. For instance, it is accepted to have a 0.001 difference between training and test accuracy rates for datasets with thousands of records. In this analysis, we utilized two datasets of varying sizes: dataset-1, consisting of 1,000 records, and dataset-2, comprising 15,420 records. To evaluate overfitting, we

examined dataset-1 under two different tolerance levels, defined by the differences between training and test accuracy rates: 0.01 (1%) and 0.05 (5%). Given that dataset-2 is approximately 15 times larger than dataset-1, our analysis for this dataset focused on a 0.01 (1%) difference between training and test accuracy rates. In the subsequent discussion, we refer to these as 1% and 5% overfitting tolerance, respectively. Overfitting tolerance, in this context, refers to the degree to which a model can fit the training data closely without losing its ability to generalize to new, unseen data.

A. DATASET-1

In the following subsections, we will expose the results of dataset-1 through three experiments, where each experiment consists of two different scenarios. In each scenario, a set of predictive models is to be evaluated on different evaluation metrics.

1) Experiment 1

As we mentioned in experimental design, this experiment implemented the classification using the raw data of dataset-1 with two scenarios. We evaluated the performance of various classification models on a dataset with missing values and imbalanced classes. In the first scenario, using the raw, unprocessed data led to significant overfitting, as evidenced by high training accuracies but much lower test accuracies across all models. As shown in Table 7, the ridge regression model achieved a best accuracy score of 78% across experiments where the difference between the training and test accuracy rates was constrained to either 1% or 5%.

In the second scenario, addressing class imbalance through resampling techniques led to significant improvements in model performance. As shown in Table 8, methods such as SMOTE and ADASYN effectively balanced the enhancement of test accuracy while maintaining specificity and precision. Both techniques, when applied with the ridge model, achieved an accuracy rate of 84% when the accepted differences between training and test accuracy rates were limited to 1% or 5%. This represents a 6% improvement in test accuracy compared to the previous scenario. In contrast, undersampling proved less effective, likely due to the reduction in available data.

Table 8. THE RESULTS OF VARIOUS CLASSIFIERS FOR SCENARIO 1 OF EXPERIMENT 1

Evaluation metrics	Classifier							
	Bagging	Boosting	Stacking	Ridge regression	RF	Gaussian NB	LR	XGB
Training Accuracy	0.98	0.84	0.76	**0.79**	1.00	0.72	0.75	1.00
Test Accuracy	0.77	0.77	0.75	**0.78**	0.79	0.73	0.75	0.75
Specificity	0.65	0.66	0.63	0.58	0.66	0.69	0.50	0.63
Precision	0.68	0.68	0.65	0.71	0.71	0.65	0.38	0.65
F1-score	0.66	0.67	0.64	0.58	0.67	0.66	0.43	0.64

Table 9. THE RESULTS OF VARIOUS CLASSIFIERS FOR SCENARIO 2 OF EXPERIMENT 1

Resampling method	Evaluation metrics	Classifier							
		Bagging	Boosting	Stacking	Ridge regression	RF	Gaussian NB	LR	XGB
Oversampling	Train. Accuracy	1.00	0.79	0.99	0.73	1.00	0.63	0.58	1.00
	Test Accuracy	0.88	0.75	0.88	0.72	0.91	0.66	0.61	0.88
	Specificity	0.87	0.75	0.92	0.73	0.90	0.68	0.59	0.90
	Precision	0.88	0.75	0.92	0.73	0.91	0.70	0.59	0.90
	F1-score	0.87	0.74	0.92	0.73	0.90	0.66	0.59	0.90
Undersampling	Train. Accuracy	0.97	0.83	0.74	0.74	1.00	0.64	0.64	1.00
	Test Accuracy	0.71	0.67	0.65	0.70	0.70	0.57	0.55	0.70
	Specificity	0.73	0.72	0.66	0.72	0.76	0.60	0.54	0.70
	Precision	0.73	0.72	0.66	0.72	0.76	0.61	0.54	0.70
	F1-score	0.72	0.72	0.65	0.72	0.76	0.59	0.54	0.70
SMOTE	Train. Accuracy	1.00	0.87	0.99	**0.85**	1.00	0.68	0.62	1.00
	Test Accuracy	0.81	0.80	0.82	**0.84**	0.85	0.69	0.67	0.83
	Specificity	0.80	0.78	0.81	0.84	0.85	0.67	0.62	0.84
	Precision	0.81	0.79	0.81	0.85	0.85	0.71	0.62	0.84
	F1-score	0.80	0.78	0.81	0.84	0.85	0.66	0.61	0.84
ADASYN	Train. Accuracy	0.99	0.85	0.98	0.86	1.00	0.67	0.57	1.00
	Test Accuracy	0.85	0.81	0.84	0.84	0.87	0.66	0.55	0.84
	Specificity	0.84	0.77	0.82	0.83	0.84	0.67	0.65	0.84
	Precision	0.84	0.77	0.82	0.83	0.84	0.70	0.65	0.85
	F1-score	0.84	0.77	0.82	0.83	0.84	0.65	0.65	0.84

Table 10. THE RESULTS OF VARIOUS CLASSIFIERS FOR SCENARIO 3 OF EXPERIMENT 2

Imputation Method	Evaluation metrics	Classifier							
		Bagging	Boosting	Stacking	Ridge regression	RF	Gaussian NB	LR	XGB
Iterative	Train. Accuracy	0.97	0.84	0.78	0.78	1.00	0.73	0.75	1.00
	Test Accuracy	0.78	0.76	**0.77**	0.76	0.77	0.77	0.75	0.76
	Specificity	0.65	0.65	0.65	0.54	0.65	0.70	0.50	0.63
	Precision	0.69	0.67	0.68	0.66	0.68	0.69	0.38	0.66
	F1-score	0.66	0.66	0.66	0.52	0.66	0.69	0.43	0.64
KNN	Train. Accuracy	0.98	0.85	0.79	0.79	1.00	0.73	0.75	1.00
	Test Accuracy	0.74	0.75	**0.77**	**0.77**	0.79	0.76	0.75	0.76
	Specificity	0.61	0.63	0.67	0.57	0.66	0.70	0.50	0.64
	Precision	0.63	0.65	0.68	0.70	0.71	0.68	0.38	0.66
	F1-score	0.61	0.64	0.67	0.56	0.67	0.69	0.43	0.65
Simple	Train. Accuracy	0.98	0.84	0.80	0.78	1.00	0.73	0.75	1.00
	Test Accuracy	0.77	0.76	0.77	**0.77**	0.77	0.77	0.75	0.77
	Specificity	0.64	0.65	0.69	0.55	0.63	0.70	0.50	0.64
	Precision	0.68	0.67	0.69	0.68	0.68	0.69	0.38	0.67
	F1-score	0.65	0.66	0.69	0.55	0.65	0.69	0.43	0.65

Table 11. THE RESULTS OF VARIOUS CLASSIFIERS FOR DIFFERENT CLASS BALANCING METHODS WITH ITERATIVE METHOD IN SCENARIO 4

Resampling method	Evaluation Metrics	Classifier							
		Bagging	Boosting	Stacking	Ridge regression	RF	Gaussian NB	LR	XGB
Over	Train. Accuracy	1.00	0.80	0.98	0.73	1.00	0.63	0.57	1.00
	Test Accuracy	0.92	0.72	0.87	0.70	0.92	0.66	0.59	0.91
	Specificity	0.92	0.74	0.92	0.72	0.92	0.66	0.63	0.92
	Precision	0.92	0.74	0.92	0.72	0.93	0.68	0.63	0.93
	F1-score	0.92	0.74	0.92	0.72	0.92	0.65	0.63	0.92
Under	Train. Accuracy	0.98	0.82	0.85	0.73	1.00	0.65	0.69	1.00
	Test Accuracy	0.73	0.61	0.59	0.68	0.69	0.54	0.59	0.65
	Specificity	0.73	0.69	0.66	0.69	0.72	0.57	0.50	0.72
	Precision	0.73	0.69	0.66	0.69	0.72	0.57	0.50	0.72
	F1-score	0.73	0.69	0.66	0.69	0.72	0.56	0.50	0.72
SMOTE	Train. Accuracy	0.99	0.88	0.99	0.73	1.00	0.63	0.59	1.00
	Test Accuracy	0.84	0.82	0.85	0.75	0.86	0.66	0.61	0.84
	Specificity	0.85	0.85	0.85	0.73	0.87	0.67	0.58	0.85
	Precision	0.86	0.85	0.85	0.73	0.87	0.73	0.58	0.85
	F1-score	0.85	0.85	0.85	0.73	0.87	0.65	0.58	0.85
ADASYN	Train. Accuracy	0.99	**0.87**	1.00	0.74	1.00	0.63	0.68	1.00
	Test Accuracy	0.86	**0.83**	0.85	0.71	0.87	0.63	0.69	0.85
	Specificity	0.85	0.83	0.83	0.72	0.87	0.62	0.65	0.83
	Precision	0.85	0.83	0.84	0.72	0.87	0.65	0.65	0.83
	F1-score	0.85	0.82	0.83	0.72	0.87	0.60	0.65	0.83

2) Experiment 2

Experiment 2 includes two different scenarios as well, i.e., the third and fourth scenarios. In the third scenario, we examined how different imputation methods (KNN, Iterative, and Simple) impacted model performance when applied before training. While these methods generally improved accuracy and stability across most classifiers, indicating effective handling of missing data, the overall impact on predictive accuracy was not positive. As shown in Table 9, all three imputation methods achieved a 77% accuracy rate when the difference between training and test accuracy rates was constrained to either 1% or 2%. However, comparing the results in Tables 7 and 9 reveals that these imputation methods actually had a negative impact on all models, as evidenced by a decrease in accuracy rates. This leads to the conclusion that data imputation alone did not enhance the predictive models' accuracy.

This article has been accepted for publication in IEEE Access. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2024.3468993

Table 12. THE RESULTS OF VARIOUS CLASSIFIERS FOR DIFFERENT CLASS BALANCING METHODS WITH KNN METHOD IN SCENARIO 4

Resampling method	Evaluation Metrics	Classifier							
		Bagging	Boosting	Stacking	Ridge regression	RF	Gaussian NB	LR	XGB
Over	Train. Accuracy	1.00	0.82	0.99	0.74	1.00	0.64	0.58	1.00
	Test Accuracy	0.86	0.75	0.92	0.70	0.91	0.66	0.58	0.89
	Specificity	0.89	0.77	0.90	0.74	0.92	0.68	0.57	0.88
	Precision	0.90	0.77	0.90	0.74	0.92	0.70	0.57	0.89
	F1-score	0.89	0.77	0.90	0.74	0.92	0.67	0.57	0.88
Under	Train. Accuracy	0.98	0.84	0.85	0.76	1.00	0.65	0.65	1.00
	Test Accuracy	0.73	0.70	0.65	0.69	0.69	0.57	0.53	0.72
	Specificity	0.72	0.70	0.64	0.66	0.70	0.55	0.44	0.62
	Precision	0.72	0.70	0.64	0.66	0.70	0.56	0.44	0.62
	F1-score	0.72	0.70	0.64	0.66	0.70	0.53	0.44	0.62
SMOTE	Train. Accuracy	0.99	0.88	0.99	0.74	1.00	0.64	0.57	1.00
	Test Accuracy	0.85	0.83	0.84	0.70	0.86	0.67	0.58	0.85
	Specificity	0.86	0.82	0.87	0.70	0.85	0.66	0.63	0.85
	Precision	0.86	0.83	0.87	0.70	0.85	0.70	0.63	0.85
	F1-score	0.86	0.82	0.87	0.70	0.85	0.64	0.63	0.85
ADASYN	Train. Accuracy	0.99	**0.88**	0.96	0.75	1.00	0.63	0.57	1.00
	Test Accuracy	0.83	**0.86**	0.85	0.71	0.86	0.64	0.55	0.83
	Specificity	0.84	0.86	0.85	0.70	0.87	0.64	0.63	0.85
	Precision	0.84	0.86	0.86	0.70	0.87	0.66	0.63	0.85
	F1-score	0.84	0.86	0.85	0.70	0.87	0.62	0.63	0.85

Table 13. THE RESULTS OF VARIOUS CLASSIFIERS FOR DIFFERENT CLASS BALANCING METHODS WITH SIMPLE METHOD IN SCENARIO 4

Resampling method	Evaluation Metrics	Classifier							
		Bagging	Boosting	Stacking	Ridge regression	RF	Gaussian NB	LR	XGB
Over	Train. Accuracy	1.00	0.81	0.99	0.75	1.00	0.65	0.59	1.00
	Test Accuracy	0.88	0.75	0.91	0.73	0.91	0.65	0.63	0.91
	Specificity	0.92	0.77	0.90	0.72	0.91	0.66	0.58	0.91
	Precision	0.92	0.77	0.90	0.72	0.91	0.67	0.58	0.92
	F1-score	0.92	0.77	0.90	0.72	0.91	0.65	0.58	0.91
Under	Train. Accuracy	0.97	0.85	0.89	0.73	1.00	0.65	0.61	1.00
	Test Accuracy	0.72	0.64	0.63	0.72	0.76	0.59	0.52	0.73
	Specificity	0.73	0.66	0.65	0.68	0.74	0.56	0.49	0.69
	Precision	0.73	0.66	0.65	0.68	0.74	0.55	0.49	0.69
	F1-score	0.73	0.66	0.65	0.68	0.74	0.56	0.49	0.69
SMOTE	Train. Accuracy	0.99	**0.88**	0.99	0.76	1.00	0.64	0.58	1.00
	Test Accuracy	0.83	**0.83**	0.83	0.74	0.86	0.68	0.59	0.84
	Specificity	0.85	0.83	0.85	0.72	0.85	0.67	0.63	0.85
	Precision	0.85	0.83	0.85	0.72	0.85	0.72	0.63	0.85
	F1-score	0.85	0.83	0.85	0.72	0.85	0.64	0.63	0.85
ADASYN	Train. Accuracy	0.99	**0.88**	0.99	0.74	1.00	0.63	0.57	1.00
	Test Accuracy	0.83	**0.83**	0.82	0.71	0.87	0.65	0.56	0.82
	Specificity	0.88	0.83	0.85	0.72	0.86	0.65	0.58	0.86
	Precision	0.88	0.83	0.85	0.72	0.86	0.68	0.58	0.86
	F1-score	0.88	0.82	0.85	0.72	0.86	0.63	0.58	0.86

In the fourth scenario, we combined data imputation with class imbalance handling techniques. The results, presented in Tables 10, 11, and 12, demonstrate that this comprehensive approach led to significant performance improvements, particularly when using ADASYN and SMOTE methods. The highest accuracy rate of 86% was achieved using a combination of the boosting model for regression, ADASYN for class balancing, and the KNN model for data imputation. This result, with only a 2% difference between training and test accuracy, indicates a well-balanced performance in handling classes. In contrast, the undersampling method proved less effective, likely due to the reduced dataset size.

Table 14. THE RESULTS OF VARIOUS CLASSIFIERS WITH REMOVING COLUMNS WITH SIGNIFICANT MISSING VALUES IN SCENARIO 5

Evaluation metrics	Classifier							
	Bagging	Boosting	Stacking	Ridge regression	RF	Gaussian NB	LR	XGB
Training Accuracy	0.98	0.83	0.86	**0.78**	1.00	0.72	0.75	1.00
Test Accuracy	0.77	0.75	0.73	**0.77**	0.77	0.76	0.75	0.78
Specificity	0.65	0.62	0.62	0.57	0.65	0.69	0.50	0.68
Precision	0.68	0.65	0.63	0.67	0.68	0.67	0.38	0.70
F1-score	0.66	0.63	0.63	0.57	0.66	0.68	0.43	0.68

Table 15. THE RESULTS OF VARIOUS CLASSIFIERS WITH REMOVING COLUMNS WITH SIGNIFICANT MESSING VALUES AND ADDRESSING IMBALANCED CLASSES IN SCENARIO 6

Resampling method	Evaluation Metrics	Classifier							
		Bagging	Boosting	Stacking	Ridge regression	RF	Gaussian NB	LR	XGB
Over	Train. Accuracy	0.99	0.81	0.99	0.73	1.00	0.64	0.56	1.00
	Test Accuracy	0.91	0.78	0.89	0.73	0.91	0.66	0.59	0.89
	Specificity	0.91	0.79	0.91	0.72	0.91	0.67	0.71	0.90
	Precision	0.92	0.79	0.91	0.72	0.91	0.69	0.71	0.91
	F1-score	0.91	0.79	0.91	0.72	0.91	0.66	0.71	0.90
Under	Train. Accuracy	0.99	0.83	0.81	0.76	1.00	0.66	0.67	1.00
	Test Accuracy	0.68	0.59	0.64	0.69	0.70	0.62	0.55	0.69
	Specificity	0.75	0.75	0.67	0.75	0.78	0.60	0.52	0.75
	Precision	0.74	0.75	0.67	0.75	0.79	0.61	0.53	0.75
	F1-score	0.76	0.75	0.67	0.75	0.78	0.59	0.52	0.75
SMOTE	Train. Accuracy	1.00	0.84	0.99	0.84	1.00	0.69	0.57	1.00
	Test Accuracy	0.82	0.77	0.80	0.81	0.84	0.68	0.57	0.84
	Specificity	0.82	0.77	0.80	0.83	0.84	0.67	0.72	0.84
	Precision	0.82	0.77	0.81	0.83	0.84	0.69	0.72	0.84
	F1-score	0.82	0.77	0.80	0.83	0.84	0.66	0.72	0.84
ADASYN	Train. Accuracy	0.99	0.84	0.98	**0.85**	1.00	0.68	0.56	1.00
	Test Accuracy	0.82	0.78	0.84	**0.83**	0.85	0.65	0.56	0.83
	Specificity	0.80	0.78	0.82	0.82	0.84	0.65	0.63	0.85
	Precision	0.80	0.78	0.82	0.82	0.84	0.66	0.63	0.86
	F1-score	0.79	0.78	0.82	0.82	0.84	0.64	0.63	0.85

Comparing the results from Table 11 (scenario 4) with those from Table 8 (scenario 2), we observe an improvement in accuracy from 84% to 86%. This suggests that applying data imputation in conjunction with class balancing methods yielded a slight improvement over using class balancing methods alone.

3) Experiment 3

Experiment 3 examined two scenarios to address data quality issues and their impact on model performance. In Scenario 5, columns with a high percentage of missing values were removed to reduce noise and potentially improve model performance. As shown in Table 13, training on raw data yielded 78% accuracy, whereas handling missing values yielded 77% accuracy using the ridge regression model. However, variability in specificity and precision indicated that simply removing columns with missing data did not fully address performance issues caused by class imbalance.

To further investigate this issue, Scenario 6 combined the removal of these columns with class imbalance handling techniques. Table 14 lists the results of this scenario. Comparing Tables 13 and 14 revealed a significant improvement in the accuracy rate when both the class imbalance and missing data issues were addressed simultaneously. The accuracy rate improved from 77% to 83%

Table 16. THE COMPARISON OF THE BEST SCENARIO'S PERFORMANCE FOR DATASET-1 ACCORDING TO OVERFITTING ANALYSIS

Overfitting tolerance level	The best scenario	The best model combination	Accuracy Metrics	
			Training	Test
<= 1%	scenario 2	SMOTE + Ridge	0.84	0.84
<= 5%	scenario 4	KNN + ADASYN + Boosting	0.88	0.86

using a combination of ridge regression, the ADASYN method, and column removal. Notably, the difference between the training and test accuracy rates for this combination was only 2%.

A summary of the best-obtained results at different overfitting tolerance levels is reported in Table 15, providing further insight into the model's performance under various conditions.

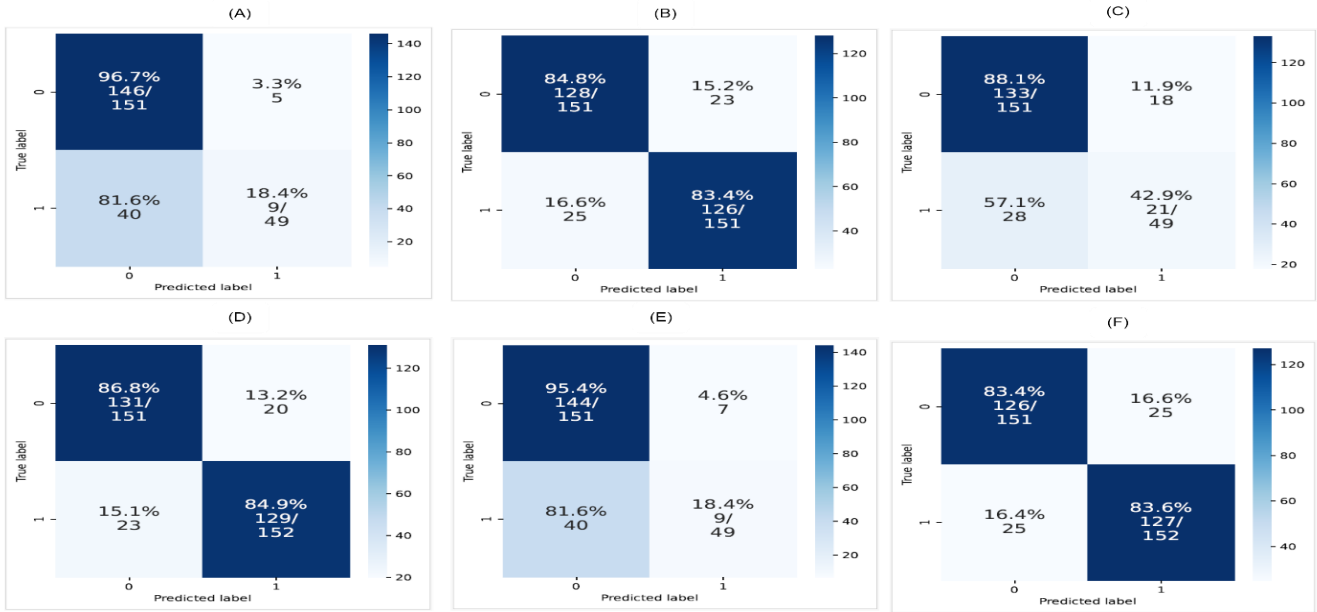

FIGURE 5. **The confusion matrix for the best model in each scenario in the experiments, as follows: (A) scenario 1, (B) scenario 2, (C) scenario 3, (D) scenario 4, (E) scenario 5, and (F) scenario 6.**

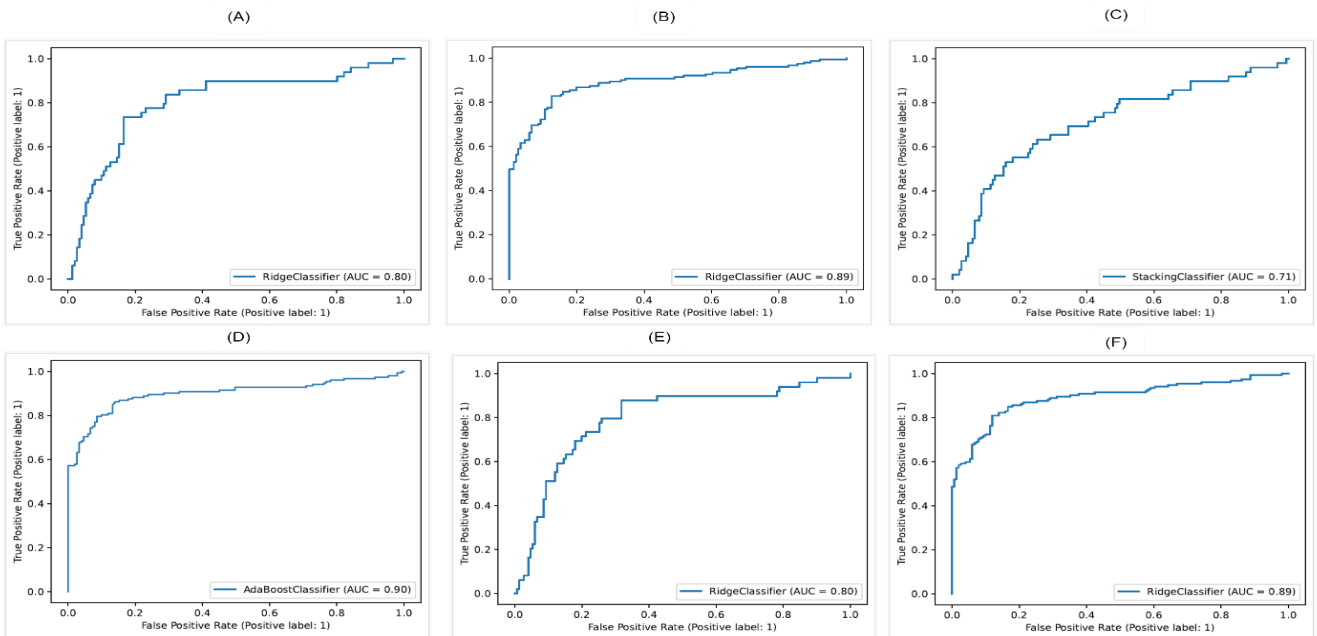

Figure 6. **The ROC plot for the best model in each scenario in the experiments, as follows: (A) scenario 1, (B) scenario 2, (C) scenario 3, (D) scenario 4, (E) scenario 5, and (F) scenario 6.**

These results emphasize that addressing the problem of class imbalance is more impactful than addressing the missing data problem alone. Additionally, the findings suggest that a larger overfitting tolerance should be considered in such scenarios.

To visually compare the different six scenarios, the confusion matrices (CMs) were depicted in Fig. 5 with six subfigures. Figs. 5(b, 5(c, 5(d, and 5(f represent the scenarios (i.e., 2, 3, 4, and 6) that addressed the class imbalance in the dataset show a high accuracy rate. The other scenarios 1 and 5 (i.e., Figs. 5(a, and 5(e) which did not utilize any method to address the class imbalance in the dataset and show a lower accuracy rate

relative to the aforementioned subfigures of Fig. 5. The best-achieved result was reported in Fig. 5(d for scenario 4, which is compatible with the reported results in Table 11. The same performance for the six scenarios is depicted in Fig. 6 on the receiver operating characteristic curve (ROC) metric, where the best achieved performance was depicted in Fig. 6(d. Thus, both the numerical and visual results are consistence.

4) Dataset-2

For the validation of the proposed method, we employed a widely-recognized dataset because the source code for the latest benchmark techniques is not open for public access, preventing us from applying them to the proposed dataset, referred to as dataset-1. Therefore, to facilitate a fair comparison of our model's performance with these leading methods, we made use of an alternative dataset, named dataset-2. In this analysis, the 1% overfitting tolerance will be used only, as the dataset size is about 15 times the dataset-1 size. Of note, the reported results will be of four significant figures, as the training and test accuracy rates for dataset-2 are very close for most of the obtained accuracy rates.

Table 17. THE COMPARISON OF THE BEST SCENARIO'S PERFORMANCE FOR DATASET-2 ACCORDING TO OVERFITTING ANALYSIS OF LESS THAN 1%

Scenario No.	The best model	Accuracy Metric	
		Training	Test
1	Ridge Logistic Regression	0.9402	0.9400
2	Oversampling + RF	1.00	0.9978
3	Iterative + Ridge KNN + Ridge Simple + Ridge	0.9402	0.9400
4	Iterative + Over-sampling +Stacking KNN + Over-sampling +Stacking Simple + Over-sampling +Stacking	1.00	0.9998
5	Column Removing + Ridge	0.9402	0.9400
6	Column Removing + Over-sampling + Stacking	1.00	**1.00**

In Table 17, each scenario is listed with the best combinations of methods (i.e., missing data handling, method, class imbalance handling method, and the predictive model), the training accuracy, and the test accuracy. In Table 17, the models that do not suffer from the overfitting problem are listed. The best achieved results were in scenario 6 where the column removal method was used to handle the missing values issues. We removed two columns, namely, "past number of claims" and "number of supplements," as the percentage of the missing values is more than 15%. The over-sampling method was used to handle the class imbalance issue, and the stacking was used as a predictive model. Scenario 4 achieved very close results to scenario 6 using any of the three methods for data imputation, the over-sampling method for addressing the class imbalance, and the stacking model for the classification task. From Table 17, the conclusions include using the over-sampling method with the stacking model with any of the missing data handling methods (i.e., column removal or data imputation) should achieve the best results.

Comparing the results of the proposed method against the state-of-the-art methods, Table 18 outlines that the proposed methodology outperforms the state-of-the-art methods with two models, i.e., the last two rows. The best performing models were the stacking and RF. The class imbalance issue was addressed by the over-sampling method and the missing values problem was handled using the column removal technique. Moreover, the state-of-the-art methods did not analyze the overfitting issue, which is highly expected for such an imbalanced dataset, i.e., dataset-2. Thus, the obtained results of the state-of-the-art methods need further analysis, which was not possible, as the source code of these methods is not available.

Table 18. THE COMPARISON OF THE SOTA FOR DATASET-2

Study	The Utilized Method(s)	Accuracy
[42]	XGBoost + SMOTE	0.951
[28]	Bagged ensemble learning based CNN	0.980
[62]	GAFCM	0.843
The proposed method	Column Removing + Over-sampling + RF	0.997
The proposed method	Column Removing + Over-sampling + Stacking	1.000

5) Discussion

For dataset-1, the proposed methodology achieved an accuracy improvement of 6% compared to the training on the raw data, i.e., the dataset with the missing value and class imbalance issues with a 1% overfitting tolerance. On the other hand, the accuracy improvement of 8% when a 5% overfitting tolerance was utilized. In the six scenarios, the ridge model was the dominant model, as it was the best model in all scenarios but scenario 4. This can be justified by considering the few numbers of observations and the relatively high number of predictor variables, i.e., 22 features. For the imbalance learning techniques, both SMOTE and ADASYN perform better than the other two methods. For missing data, both the data imputation and column removal technique when used alone, without the data imbalance learning technique, declined the accuracy rates. But, using a method to handle the missing values with a method to handle the class imbalance gave the best results, where handling the missing values slightly contributed to improving the accuracy rates. Thus, it is concluded that addressing the problem of class imbalance is

more rewarding than the missing values problem in the fraud detection task.

For dataset-2, the proposed methodology outperformed the state-of-the-art methods due to the variety of the utilized methods and considering the overfitting issues as well. In Table 16, it is obvious that the stacking and ridge models perform better than the other models. The ridge model performed better when the data imbalance learning technique method is not present and the accuracy rate is limited to 94%, in scenarios 1, 3, and 5. The stacking model performed the best when the class imbalance and missing data issues were addressed. Removing columns proved to be more effective with the stacking model than using data imputation. Meanwhile, employing an over-sampling technique emerged as the most efficient method for addressing class imbalance. All of the data imputation methods achieved the same results. Again, it is concluded that it is more important to balance the classes of the dataset, scenario 2 in Table 17, relative to handling the missing values in the fraud detection task.

VI. CONCLUSION

In the current work, the problem of predicting fraud activities in insurance companies was framed as a classification problem of two outcomes, namely, 1) fraud and 2) legitimate. The fraud insurance dataset usually suffers from two problems, which are class imbalance, as the number of fraud activities is much less, and missing values of the customers. The existing methods did not explore these two problems intensively. Moreover, despite the high possibility of model overfitting due to the class imbalance problem, the current research work did not perform the overfitting analysis. Thus, the proposed work investigated addressing these two problems with different techniques to find out the effect of each of them. The investigations were validated with two datasets, a real dataset from an Egyptian car insurance company and a standard car insurance dataset. The evaluation of the proposed method includes six different scenarios to solve the two different problems of the dataset. The experimental results show that addressing the class imbalance greatly improves the accuracy of the trained model in comparison to training the model with imbalanced classes on one hand. In addition, the study found that addressing the class imbalance problem is more rewarding than addressing the missing values issue. Moreover, the study found that the over-sampling method outperformed the methods when the imbalance ratio is higher while the SMOTE and ADASYN methods performed better when the class imbalance is less served. On the other hand, addressing the missing values problem slightly improves the predictive models' performance. For dataset-1, the best-achieved accuracy without addressing any problem was 78%, while the best-achieved results with handling the missing data only, handling the class imbalance only, and handling both issues were 77%, 84%, and 86%, respectively. For dataset-2, using the column removal and the over-sampling methods with the stacking predictive model outperformed the state-of-the-art methods. The future directions include using generative models for addressing the missing value problem and using deep learning models for prediction purposes.

AUTHORS' CONTRIBUTIONS

A. K. performed conceptualization, data curation, formal analysis, methodology, validation, visualization, and writing, reviewing, & editing of the original draft. Z. L. performed conceptualization, supervision, funding acquisition, project administration, reviewing, and editing of the study. A. F. performed resources, software, validation, visualization, writing, reviewing, & editing of the original draft. A.S. performed formal analysis, investigation, writing, reviewing, & editing. A. A. performed funding acquisition, reviewing, and editing. All the authors have read and agreed to the submitted version of the manuscript.

CONFLICTS OF INTEREST

The authors declare that they have no conflicts of interest.

DATA AVAILABILITY

The data that support the findings of this study are available at this link (https://github.com/stars-of-orion/Enhancing-Insurance-Fraud-Detection).

FUNDING

This study is supported via funding from Prince Sattam bin Abdulaziz University under project number (PSAU/2024/R/1445) with additional support from the National Natural Science Foundation of China (grant no. 12071487). Additionally, this work was partially supported by grant number a research grant from the Omani Ministry of Higher Education, Research, and Innovation under the project number BFP/RGP/ICT/23/382.

REFERENCES

[1] A. A. Khalil, Z. Liu, and A. A. Ali, "Using an adaptive network-based fuzzy inference system model to predict the loss ratio of petroleum insurance in Egypt," *Risk Management and Insurance Review*, vol. 25, no. 1, pp. 5–18, 2022, doi: 10.1111/rmir.12200.

[2] C. Bockel-Rickermann, T. Verdonck, and W. Verbeke, "Fraud analytics: A decade of research: Organizing challenges and solutions in the field,"

Expert Syst Appl, vol. 232, p. 120605, 2023, doi: https://doi.org/10.1016/j.eswa.2023.120605.

[3] Y. Wang and W. Xu, "Leveraging deep learning with LDA-based text analytics to detect automobile insurance fraud," *Decis Support Syst*, vol. 105, pp. 87–95, 2018, doi: https://doi.org/10.1016/j.dss.2017.11.001.

[4] B. Itri, Y. Mohamed, Q. Mohammed, and B. Omar, "Performance comparative study of machine learning algorithms for automobile insurance fraud detection," in *2019 Third International Conference on Intelligent Computing in Data Sciences (ICDS)*, 2019, pp. 1–4. doi: 10.1109/ICDS47004.2019.8942277.

[5] R. P. B. Piovezan, P. P. de Andrade Junior, and S. L. Ávila, "Machine Learning Method for Return Direction Forecast of Exchange Traded Funds (ETFs) Using Classification and Regression Models," *Comput Econ*, 2023, doi: 10.1007/s10614-023-10385-4.

[6] A. A. Khalil, Z. Liu, A. Salah, A. Fathalla, and A. Ali, "Predicting Insolvency of Insurance Companies in Egyptian Market Using Bagging and Boosting Ensemble Techniques," *IEEE Access*, vol. 10, pp. 117304–117314, 2022, doi: doi: 10.1109/ACCESS.2022.3210032.

[7] N. Boodhun and M. Jayabalan, "Risk prediction in life insurance industry using supervised learning algorithms," *Complex & Intelligent Systems*, vol. 4, no. 2, pp. 145–154, 2018, doi: 10.1007/s40747-018-0072-1.

[8] D. Tiwari, B. Nagpal, B. S. Bhati, A. Mishra, and M. Kumar, "A systematic review of social network sentiment analysis with comparative study of ensemble-based techniques," *Artif Intell Rev*, vol. 56, no. 11, pp. 13407–13461, 2023, doi: 10.1007/s10462-023-10472-w.

[9] M. Liao, S. Tian, Y. Zhang, G. Hua, W. Zou, and X. Li, "PDA: Progressive Domain Adaptation for Semantic Segmentation," *Knowl Based Syst*, vol. 284, p. 111179, 2024, doi: https://doi.org/10.1016/j.knosys.2023.111179.

[10] A. Khalil, Z. Liu, and A. Ali, "Precision in Insurance Forecasting: Enhancing Potential with Ensemble and Combination Models based on the Adaptive Neuro-Fuzzy Inference System in the Egyptian Insurance Industry," *Applied Artificial Intelligence*, vol. 38, no. 1, p. 2348413, Dec. 2024, doi: 10.1080/08839514.2024.2348413.

[11] A. K. I. Hassan and A. Abraham, "Modeling insurance fraud detection using ensemble combining classification," *International Journal of Computer Information Systems and Industrial Management Applications*, vol. 8, pp. 257–265, 2016.

[12] V. R. Shetty and R. L. Malghan, "Safeguarding against Cyber Threats: Machine Learning-Based Approaches for Real-Time Fraud Detection and Prevention," *Engineering Proceedings*, vol. 59, no. 1, p. 111, 2023.

[13] A. R. Khalid, N. Owoh, O. Uthmani, M. Ashawa, J. Osamor, and J. Adejoh, "Enhancing Credit Card Fraud Detection: An Ensemble Machine Learning Approach," *Big Data and Cognitive Computing*, vol. 8, no. 1, p. 6, 2024.

[14] A. A. Khalil, Z. Liu, and A. Ali, "Enhancing operational efficiency of insurance companies: a fuzzy time series approach to loss ratio forecasting in the Egyptian market," *Journal of Business Analytics*, pp. 1–19, doi: 10.1080/2573234X.2024.2393609.

[15] M. Hanafy and R. Ming, "Improving imbalanced data classification in auto insurance by the data level approaches," *International Journal of Advanced Computer Science and Applications*, vol. 12, no. 6, 2021.

[16] B. Baesens, S. Höppner, I. Ortner, and T. Verdonck, "robROSE: A robust approach for dealing with imbalanced data in fraud detection," *Stat Methods Appt*, vol. 30, no. 3, pp. 841–861, 2021, doi: 10.1007/s10260-021-00573-7.

[17] S. Subudhi and S. Panigrahi, "Effect of Class Imbalanceness in Detecting Automobile Insurance Fraud," in *2018 2nd International Conference on Data Science and Business Analytics (ICDSBA)*, 2018, pp. 528–531. doi: 10.1109/ICDSBA.2018.00104.

[18] T. Olalekan Yusuf and A. Rasheed Babalola, "Control of insurance fraud in Nigeria: an exploratory study (case study)," *J Financ Crime*, vol. 16, no. 4, pp. 418–435, Jan. 2009, doi: 10.1108/13590790910993744.

[19] R. Bhowmik, "Detecting auto insurance fraud by data mining techniques," *Journal of Emerging Trends in Computing and Information Sciences*, vol. 2, no. 4, pp. 156–162, 2011.

[20] K. Nian, H. Zhang, A. Tayal, T. Coleman, and Y. Li, "Auto insurance fraud detection using unsupervised spectral ranking for anomaly," *The Journal of Finance and Data Science*, vol. 2, no. 1, pp. 58–75, 2016, doi: https://doi.org/10.1016/j.jfds.2016.03.001.

[21] S. S. Waghade and A. M. Karandikar, "A comprehensive study of healthcare fraud detection based on machine learning," *International Journal*

of Applied Engineering Research, vol. 13, no. 6, pp. 4175–4178, 2018.

[22] R. Roy and K. T. George, "Detecting insurance claims fraud using machine learning techniques," in *2017 International Conference on Circuit ,Power and Computing Technologies (ICCPCT)*, 2017, pp. 1–6. doi: 10.1109/ICCPCT.2017.8074258.

[23] G. Kowshalya and M. Nandhini, "Predicting Fraudulent Claims in Automobile Insurance," in *2018 Second International Conference on Inventive Communication and Computational Technologies (ICICCT)*, 2018, pp. 1338–1343. doi: 10.1109/ICICCT.2018.8473034.

[24] L. Goleiji and M. Tarokh, "Identification of influential features and fraud detection in the Insurance Industry using the data mining techniques (Case study: automobile's body insurance)," *Majlesi J Multimed Process*, vol. 4, pp. 1–5, 2015.

[25] S. Goundar, S. Prakash, P. Sadal, and A. Bhardwaj, "Health Insurance Claim Prediction Using Artificial Neural Networks," *International Journal of System Dynamics Applications (IJSDA)*, vol. 9, no. 3, pp. 40–57, 2020.

[26] J. Debener, V. Heinke, and J. Kriebel, "Detecting insurance fraud using supervised and unsupervised machine learning," *Journal of Risk and Insurance*, vol. 90, no. 3, pp. 743–768, Sep. 2023, doi: https://doi.org/10.1111/jori.12427.

[27] A. Urunkar, A. Khot, R. Bhat, and N. Mudegol, "Fraud Detection and Analysis for Insurance Claim using Machine Learning," in *2022 IEEE International Conference on Signal Processing, Informatics, Communication and Energy Systems (SPICES)*, 2022, pp. 406–411. doi: 10.1109/SPICES52834.2022.9774071.

[28] Y. Abakarim, M. Lahby, and A. Attioui, "A Bagged Ensemble Convolutional Neural Networks Approach to Recognize Insurance Claim Frauds," *Applied System Innovation*, vol. 6, no. 1, 2023, doi: 10.3390/asi6010020.

[29] B. Xu, Y. Wang, X. Liao, and K. Wang, "Efficient fraud detection using deep boosting decision trees," *Decis Support Syst*, vol. 175, p. 114037, 2023, doi: https://doi.org/10.1016/j.dss.2023.114037.

[30] S. Subudhi and S. Panigrahi, "Use of optimized Fuzzy C-Means clustering and supervised classifiers for automobile insurance fraud detection," *Journal of King Saud University - Computer and Information Sciences*, vol. 32, no. 5, pp. 568–575, 2020, doi: https://doi.org/10.1016/j.jksuci.2017.09.010.

[31] R. Wongpanti and S. Vittayakorn, "Enhancing Auto Insurance Fraud Detection Using Convolutional Neural Networks," in *2024 21st International Joint Conference on Computer Science and Software Engineering (JCSSE)*, 2024, pp. 294–301. doi: 10.1109/JCSSE61278.2024.10613702.

[32] S.-Z. S. Nordin, Y. B. Wah, N. K. Haur, A. Hashim, N. Rambeli, and N. A. Jalil, "Predicting automobile insurance fraud using classical and machine learning models," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 14, no. 1, pp. 911–921, 2024.

[33] P. Aiemsuwan and S. Srikamdee, "A Novel Hybrid Method for Imbalanced Automobile Insurance Fraud Detection," in *2024 16th International Conference on Knowledge and Smart Technology (KST)*, 2024, pp. 12–17. doi: 10.1109/KST61284.2024.10499643.

[34] T. Badriyah, L. Rahmaniah, and I. Syarif, "Nearest neighbour and statistics method based for detecting fraud in auto insurance," in *2018 International Conference on Applied Engineering (ICAE)*, IEEE, 2018, pp. 1–5.

[35] S. Lee and H. K. Kim, "ADSaS: Comprehensive Real-Time Anomaly Detection System," in *Information Security Applications*, B. B. Kang and J. Jang, Eds., Cham: Springer International Publishing, 2019, pp. 29–41.

[36] P. Wang and X. Chen, "Three-way ensemble clustering for incomplete data," *IEEE Access*, vol. 8, pp. 91855–91864, 2020.

[37] D. A. Rusdah and H. Murfi, "XGBoost in handling missing values for life insurance risk prediction," *SN Appl Sci*, vol. 2, no. 8, p. 1336, 2020, doi: 10.1007/s42452-020-3128-y.

[38] A. Jadhav, D. Pramod, and K. Ramanathan, "Comparison of Performance of Data Imputation Methods for Numeric Dataset," *Applied Artificial Intelligence*, vol. 33, no. 10, pp. 913–933, Aug. 2019, doi: 10.1080/08839514.2019.1637138.

[39] G. G. Sundarkumar, V. Ravi, and V. Siddeshwar, "One-class support vector machine based undersampling: Application to churn prediction and insurance fraud detection," in *2015 IEEE International Conference on Computational Intelligence and Computing Research (ICCIC)*, 2015, pp. 1–7. doi: 10.1109/ICCIC.2015.7435726.

[40] A. K. I. Hassan and A. Abraham, "Modeling Insurance Fraud Detection Using Imbalanced Data Classification," in *Advances in Nature and Biologically Inspired Computing*, N. Pillay, A. P. Engelbrecht, A. Abraham, M. C. du Plessis, V. Snášel, and A. K. Muda, Eds., Cham: Springer International Publishing, 2016, pp. 117–127.

[41] I. M. N. Prasasti, A. Dhini, and E. Laoh, "Automobile Insurance Fraud Detection using Supervised Classifiers," in *2020 International Workshop on Big Data and Information Security (IWBIS)*, 2020, pp. 47–52. doi: 10.1109/IWBIS50925.2020.9255426.

[42] D. G. Maina, J. C. Moso, and P. K. Gikunda, "Detecting Fraud in Motor Insurance Claims Using XGBoost Algorithm with SMOTE," in *2023 International Conference on Information and Communication Technology for Development for Africa (ICT4DA)*, 2023, pp. 61–66. doi: 10.1109/ICT4DA59526.2023.10302229.

[43] W. McKinney, "Data structures for statistical computing in python," in *Proceedings of the 9th Python in Science Conference*, Austin, TX, 2010, pp. 51–56.

[44] F. Pedregosa *et al.*, "Scikit-learn: Machine learning in Python," *the Journal of machine Learning research*, vol. 12, pp. 2825–2830, 2011.

[45] P. Cerda and G. Varoquaux, "Encoding High-Cardinality String Categorical Variables," *IEEE Trans Knowl Data Eng*, vol. 34, no. 3, pp. 1164–1176, 2022, doi: 10.1109/TKDE.2020.2992529.

[46] P. Li, X. Rao, J. Blase, Y. Zhang, X. Chu, and C. Zhang, "CleanML: A Study for Evaluating the Impact of Data Cleaning on ML Classification Tasks," in *2021 IEEE 37th International Conference on Data Engineering (ICDE)*, 2021, pp. 13–24. doi: 10.1109/ICDE51399.2021.00009.

[47] T. Shadbahr *et al.*, "The impact of imputation quality on machine learning classifiers for datasets with missing values," *Communications Medicine*, vol. 3, no. 1, p. 139, 2023, doi: 10.1038/s43856-023-00356-z.

[48] S. I. Khan and A. S. M. L. Hoque, "SICE: an improved missing data imputation technique," *J Big Data*, vol. 7, no. 1, p. 37, 2020, doi: 10.1186/s40537-020-00313-w.

[49] T. Emmanuel, T. Maupong, D. Mpoeleng, T. Semong, B. Mphago, and O. Tabona, "A survey on missing data in machine learning," *J Big Data*, vol. 8, no. 1, p. 140, 2021, doi: 10.1186/s40537-021-00516-9.

[50] Doreswamy, I. Gad, and B. R. Manjunatha, "Performance evaluation of predictive models for missing data imputation in weather data," in *2017 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, 2017, pp. 1327–1334. doi: 10.1109/ICACCI.2017.8126025.

[51] F. Martínez-Plumed, C. Ferri, D. Nieves, and J. Hernández-Orallo, "Fairness and Missing Values," *ArXiv*, vol. abs/1905.12728, 2019, [Online]. Available: https://api.semanticscholar.org/CorpusID:170078966

[52] L. Dongdong, C. Ziqiu, W. Bolu, W. Zhe, Y. Hai, and D. Wenli, "Entropy-based hybrid sampling ensemble learning for imbalanced data," *International Journal of Intelligent Systems*, vol. 36, no. 7, pp. 3039–3067, Jul. 2021, doi: https://doi.org/10.1002/int.22388.

[53] M. S. Basit, A. Khan, O. Farooq, Y. U. Khan, and M. Shameem, "Handling Imbalanced and Overlapped Medical Datasets: A Comparative Study," in *2022 5th International Conference on Multimedia, Signal Processing and Communication Technologies (IMPACT)*, 2022, pp. 1–7. doi: 10.1109/IMPACT55510.2022.10029111.

[54] J. G. Avelino, G. D. C. Cavalcanti, and R. M. O. Cruz, "Resampling strategies for imbalanced regression: a survey and empirical analysis," *Artif Intell Rev*, vol. 57, no. 4, p. 82, 2024, doi: 10.1007/s10462-024-10724-3.

[55] X. Gu, P. P. Angelov, and E. A. Soares, "A self-adaptive synthetic over-sampling technique for imbalanced classification," *International Journal of Intelligent Systems*, vol. 35, no. 6, pp. 923–943, Jun. 2020, doi: https://doi.org/10.1002/int.22230.

[56] A. D. Amirruddin, F. M. Muharam, M. H. Ismail, N. P. Tan, and M. F. Ismail, "Synthetic Minority Over-sampling TEchnique (SMOTE) and Logistic Model Tree (LMT)-Adaptive Boosting algorithms for classifying imbalanced datasets of nutrient and chlorophyll sufficiency levels of oil palm (Elaeis guineensis) using spectroradiometers and unmanned aerial vehicles," *Comput Electron Agric*, vol. 193, p. 106646, 2022, doi: https://doi.org/10.1016/j.compag.2021.106646.

[57] T. Liu, X. Zhu, W. Pedrycz, and Z. Li, "A design of information granule-based under-sampling method in imbalanced data classification," *Soft comput*, vol. 24, no. 22, pp. 17333–17347, 2020, doi: 10.1007/s00500-020-05023-2.

[58] H. Yang, Z. Zhang, L. Xie, and L. Zhang, "Network security situation assessment with network attack behavior classification," *International Journal of Intelligent Systems*, vol. 37, no. 10, pp. 6909–6927, Oct. 2022, doi: https://doi.org/10.1002/int.22867.

[59] M. Liao, S. Tian, Y. Zhang, G. Hua, W. Zou, and X. Li, "Preserving Label-Related Domain-Specific Information for Cross-Domain Semantic Segmentation," *IEEE Transactions on Intelligent*

Transportation Systems, pp. 1–15, 2024, doi: 10.1109/TITS.2024.3386743.

[60] M. Hossin and M. N. Sulaiman, "A review on evaluation metrics for data classification evaluations," *International journal of data mining & knowledge management process*, vol. 5, no. 2, p. 1, 2015.

[61] M. Rashidpoor Toochaei and F. Moeini, "Evaluating the performance of ensemble classifiers in stock returns prediction using effective features," *Expert Syst Appl*, vol. 213, p. 119186, 2023, doi: https://doi.org/10.1016/j.eswa.2022.119186.

[62] S. Subudhi and S. Panigrahi, "Use of optimized Fuzzy C-Means clustering and supervised classifiers for automobile insurance fraud detection," *Journal of King Saud University - Computer and Information Sciences*, vol. 32, no. 5, pp. 568–575, 2020, doi: https://doi.org/10.1016/j.jksuci.2017.09.010.

Ahmed A. Khalil is studying PhD in Statistics and Risk Management at School of Mathematics and Statistics, Central South University, Hunan, China. he received the M.Sc. degree in Insurance from Assiut University, Assiut, Egypt in 2019. He has published many papers in Insurance, Risk management, and Artificial intelligence. His research interests are mainly in insurance, risk management, statistical learning, and machine learning.

ZAIMING LIU is a Professor at School of Mathematics and Statistics, and Doctoral Supervisor of Central South University, Hunan, China. He was former Dean of the School of Mathematics and Statistics, and former Secretary of the Party Committee. His research interests are mainly in queuing theory and queuing networks, insurance risk theory, and stochastic process and its application.

AHMED FATHALLA received the PhD degree in computer science and technology in 2021, Hunan University, Changsha, China. His research interests are mainly in machine learning.

AHMED ALI is working as an assistant professor in the Department of Computer Science, College of Computer Engineering and Sciences, Prince Sattam Bin Abdulaziz University, Alkharj, Saudi Arabia. He has received his Ph.D. degree in 2016 from the College of Information Science and Engineering at Hunan University, China. Earlier, he finished his M.Sc. degree in the same college at Hunan University in 2011. He has graduated in Computer Science in 2005 from the College of Computers and Informatics, Suez Canal University, Ismailia, Egypt. His research interests include machine learning, signal processing, channel estimation and modulation, PAPR, IoT, and VANETs.

AHMAD SALAH received the Ph.D. degree in computer science from Hunan University, China, in 2014. He received his master's degree in CS from Ain-shams University, Cairo, Egypt. He is currently an associate professor of Computer Science and Technology at Zagazig University, Egypt. He has published more than 40 papers in peer-reviewed journals, such as the IEEE Transactions on Parallel and Distributed Systems and IEEE-ACM Transactions on Computational Biology Bioinformatics, and ACM Transactions on Parallel Computing. His current research interests are parallel computing, computational biology, and machine learning.