

ABSTRACT

Insurance fraud, particularly in automobiles, presents substantial challenges for insurance companies worldwide. Fraudulent activities by policyholders, such as document falsification and evidence fabrication, aim to deceive insurers and unlawfully obtain funds, resulting in significant financial losses. However, many insurers rely solely on traditional financial assessment methods, which may be insufficient in detecting fraud, especially in cases involving sophisticated schemes or high claim volumes. This study seeks to develop an effective fraud detection model tailored to the characteristics of real-world data using machine learning algorithms. The dataset comprises 1000 insurance claims related to car collisions across seven US states in 2015. Results indicate that employing an Early Fusion deep learning model, integrating Decision Tree and Random Forest, mitigates the limitations of traditional models. The findings demonstrate that the proposed model outperforms previous traditional models and enhances fraud detection capabilities. Building on these findings, the objective is to improve fraud detection within the Vietnamese insurance industry, thereby reducing instances of contract fraud. Our framework supports sustainable development by promoting innovation, efficiency, and technological advancement for SDG 9. It also combats fraud, strengthens institutions, and fosters transparency and accountability for SDG 16 in the insurance sector. Future research will focus on leveraging prominent deep learning models globally and utilizing diverse datasets to refine and develop the most robust feasible model.

KEYWORDS. Insurance fraud, imbalanced dataset, detection, neural network, fraud prevention efforts, sustainable, SDGs.

TABLE OF CONTENTS

[**1. Introduction 8**](#_heading=h.2et92p0)

[**1.1. Problem define 8**](#_heading=h.4d34og8)

[**1.2. Problem statement 8**](#_heading=h.26in1rg)

[**1.3. Related works 11**](#_heading=h.lnxbz9)

[**1.4. Thesis Approach and Contribution 12**](#_heading=h.35nkun2)

[**1.5. Thesis outline 13**](#_heading=h.1ksv4uv)

[**2. Previous analysis 13**](#_heading=h.44sinio)

[**2.1. Evaluation Metrics 13**](#_heading=h.2jxsxqh)

[**2.2. Traditional Methods 14**](#_heading=h.z337ya)

[**2.2.1. Overview 14**](#_heading=h.1y810tw)

[**2.2.2. Traditional model analysis 15**](#_heading=h.4i7ojhp)

[**2.3. Deep learning model 18**](#_heading=h.2xcytpi)

[**2.3.1. Formula of reLu, Batch Normalization and Drop out 18**](#_heading=h.1ci93xb)

[**2.3.2 Stacking learning model 20**](#_heading=h.2bn6wsx)

[**2.4. Strategies to improve performance of model 21**](#_heading=h.3as4poj)

[**2.4.1. Recursive Feature Elimination 21**](#_heading=h.1pxezwc)

[**2.4.2. class\_dict\_weight and SMOTE 22**](#_heading=h.49x2ik5)

[**2.4.3. Hyperparameter tuning with RandomSearch 23**](#_heading=h.2p2csry)

[**3. Materials and Methods 24**](#_heading=h.147n2zr)

[**3.1. AutoMobile Claims Dataset and Challenges 24**](#_heading=h.3o7alnk)

[**3.2. Our proposed method 30**](#_heading=h.23ckvvd)

[**3.3. Experiments setup 32**](#_heading=h.ihv636)

[**4. Results 34**](#_heading=h.32hioqz)

[**4.1. Experiments on dataset 34**](#_heading=h.1hmsyys)

[**4.2. Experiments on Traditional Model 35**](#_heading=h.41mghml)

[**4.3. Experiments on Deep Learning Model 37**](#_heading=h.2grqrue)

[**4.4. Experiment on Stacking model 38**](#_heading=h.vx1227)

[**5. Discussion 39**](#_heading=h.3fwokq0)

[**6. Conclusion & Future Works 40**](#_heading=h.1v1yuxt)

[**7. Reference 41**](#_heading=h.4f1mdlm)

LIST OF FIGURES

[Figure 1: Illustration of the measurement of process times and quantities along the core stages of claims management. Source: (Mahlow & Wagner, 2016)](#bookmark=id.ptgylhgsnmwm)….…………….…. 8

[Figure 2: Illustration of how the ML-driven approach uses AI & replaces rule-based methods that require human interventions to detect fraud……..……………](#bookmark=id.apyaz6mrvf9m)…………………………………………….…………………. 9

[Figure 3. Comparison of deep learning and traditional machine learning methods. Source: (Chen et al., 2019)](#bookmark=id.cnkgkhnm45r5) ……………………………………………………………10

[Figure 4: The Support Vector Paradigm](#bookmark=id.xcabxpynnf8t)……………………………...………………. [1](#bookmark=id.xcabxpynnf8t)4

[Figure 5: A graphical representation of Decision Trees](#bookmark=id.ihy6u1fk0hze)………………………..……. 16

[Figure 6: The depiction of a 600-tree Random Forest. The average of the 600 predictions is the final prediction of the Random Forest…………………………………………](#bookmark=id.6yo0xv1337tu). 17

[Figure 7. Batch Normalization formula………..…………………….………..........… 1](#bookmark=id.ck2v3l41m6sv)8

[Figure 8: Stacking model process………………………………………………...…... 1](#bookmark=id.37cn70pml279)9

[Figure 9: Workflow Diagram of Recursive Feature Elimination (RFE)](#bookmark=id.pjjj1o7ujiet)........................ 21

[Figure 10: Synthetic Minority Over-sampling Technique……………………………. 2](#bookmark=id.z0p0jns7513)2

[Figure 11: RandomSearch layout………………………………………………….…. 2](#bookmark=id.vp7gfa2vthau)2

[Figure 12: The distribution of data in the Automobile Insurance Claims Dataset…………………………………………………………………………….…. 25](#bookmark=id.lnxbz9)

[Figure 13: The correlation among variables in the Automobile Insurance Claims Dataset……………………………………………………………………..………… 25](#bookmark=id.35nkun2)

[Figure 14: Segment the total amount of regular requests ……...……………...….…. 26](#bookmark=id.1ksv4uv)

[Figure 15: Features for fraud detection………………………………...………...….... 27](#bookmark=id.5qxptpf7b9qc)

[Figure 16: Information transformation……………………...………………….……. 28](#bookmark=id.8ugq15s8mgo1)

[Figure 17: Stacking model process…………………………………………………… 30](#bookmark=id.83eqigvz1d9a)

[Figure 18: Process of experiment setup……………………………………...………. 31](#bookmark=id.apfppjfp5b38)

[Figure 19: Accuracy and F1- score of traditional models………………………….…. 34](#bookmark=id.lbrgohomsr8d)

[Figure 20: The ROC curve of the traditional model………………………………...... 35](#bookmark=id.jvdza64tkcv)

[Figure 21: Confusion Matrix for Random Forest and Decision Tree………………... 36](#bookmark=id.4i7ojhp)

[Figure 22: Confusion Matrix for Stacking model.…………….…………](#bookmark=id.2xcytpi)…...……… 37

LIST OF TABLES

[Table 1: Descriptive statistics for categorical variables in the dataset………………... 24](#bookmark=id.rrfap3tywtsn)

[Table 2: The previous proposed models in previous analysis in this topic…..........…. 2](#bookmark=id.2jxsxqh)9

[Table 3: A result for imbalanced methods in dataset…………….……………….….. 33](#bookmark=id.1y810tw)

[Table 4: A result for experiment models in dataset………………………………..…. 3](#bookmark=id.45nzxlwxh7ry)8

LIST OF ABBREVIATIONS

| Ordinal number | Symbol  abbreviation | Explain |
| --- | --- | --- |
| 1 | AI | Artificial intelligence |
| 2 | AUC | Area under the ROC Curve |
| 3 | BG | Gradient Boosting model |
| 4 | CV | Cross-Validation |
| 5 | DTC | Decision Tree Classifier |
| 6 | EDA | Exploratory Data Analysis |
| 7 | KNC | K Nearest Neighbour Classifier model |
| 8 | LDA | Means Linear Discriminant Analysis |
| 9 | LSTM | Long short-term memory |
| 10 | MCC | Matthews Correlation Coefficient |
| 11 | ML | Machine learning |
| 12 | NB | Naive Bayes model |
| 13 | NN | Neural Network |
| 14 | PCA | Principal Component Analysis |
| 15 | ReLU | Rectified linear unit |
| 16 | RF | Random Forest |
| 17 | RFE | Recursive Feature Elimination |
| 18 | RNN | Recurrent neural network |
| 19 | ROC | Receiver operating characteristic |
| 20 | SMOTE | Synthetic Minority Over-sampling Technique |
| 21 | SRA | Unsupervised Spectral Ranking for Anomaly |
| 22 | SVM | Support Vector Machine |

# Introduction

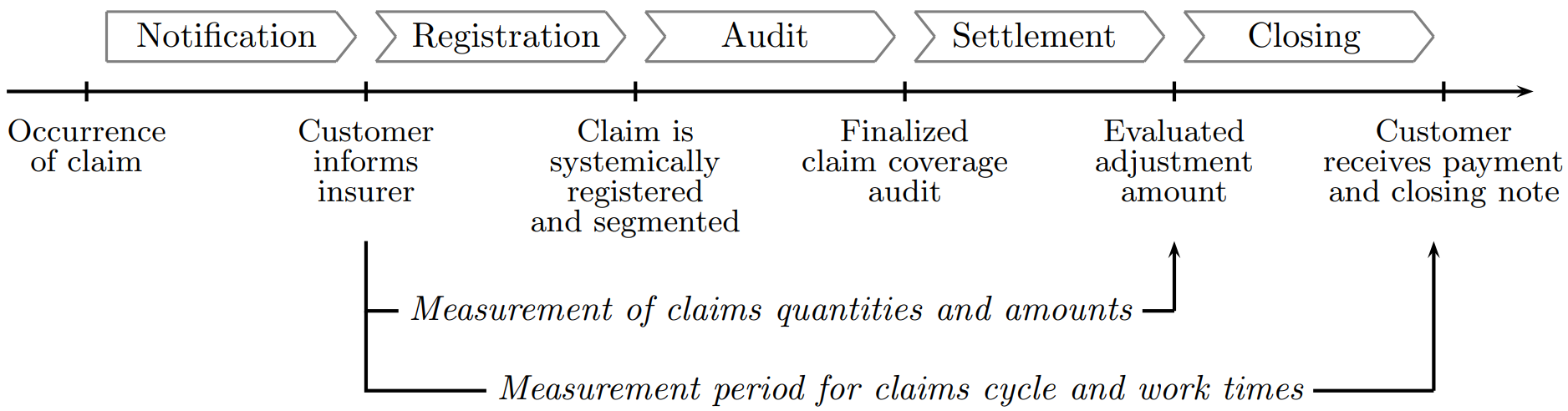
## 1.1. Problem define

Automobile insurance compensates the insured for losses from vehicle incidents like accidents, theft, and damage. It provides coverage for liability, vehicle damage, and medical expenses, with details varying by policy and location. As the number of vehicles on the road increases, the need for automobile insurance grows to protect against risks associated with driving. However, as the automobile insurance sector grows, it has also become a more frequent target for fraudulent activity. This includes false or exaggerated claims for financial gain, affecting insurers and honest policyholders alike ([Kowshalya & Nandhini, 2018](#bookmark=id.46r0co2)), [(Derrig, 2002)](#bookmark=id.2grqrue). Fraudsters utilize various tactics, such as inflating service costs, staging accidents, or filing claims for incidents that never occurred, challenging the industry to develop robust fraud detection and prevention measures.

Fraud detection is the process of identifying and preventing fraudulent activities. It involves using data mining, machine learning, and deep learning algorithms to automate the detection process and improve accuracy. By analyzing historical data and detecting patterns, these algorithms can recognize fraudulent behavior in real-time data. In the context of automobile fraud detection, various methodologies have been used such as machine learning algorithms and LSTM RNN networks (Kabir, 2022). Furthermore, there are advanced statistical and machine learning models such as the "FraudBuster" framework, specifically created to detect potential fraud during the underwriting process by analyzing patterns in policy characteristics and loss ratios ([Nagrecha et al., 2018](#bookmark=id.nmf14n)). These endeavors highlight the significance of advanced analytics in combating insurance fraud and bolstering the integrity of the insurance sector. Additionally, data technologies have been adopted to identify instances of insurance fraud, while machine learning and AI are utilized in risk management and fraud detection ([Bart, 2017b](#bookmark=id.3o7alnk)), [(Aziz & Dowling, 2018)](#bookmark=id.2p2csry).

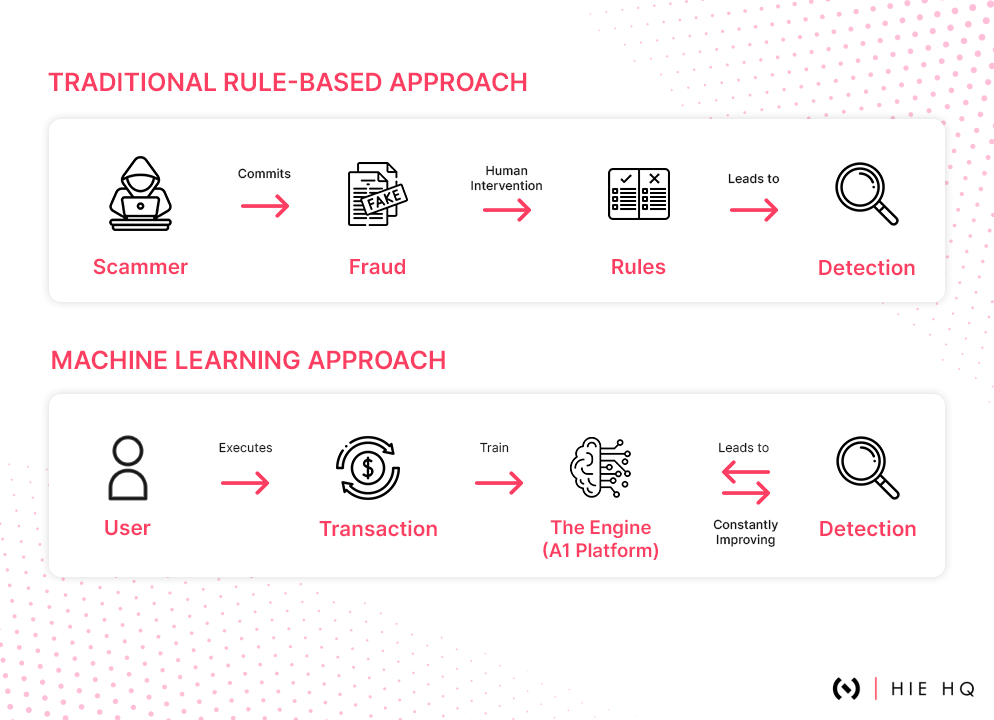
## 1.2. Problem statement

Insurance fraud involving fabricating evidence or making invalid claims can occur in various ways. One method is by fabricating insurance subject matter, where a non-existent insurance subject matter is created to deceive the insurer [(Zhen, Jing., 2022)](#bookmark=id.206ipza). Another way is by providing untruthful or incomplete information in insurance applications or claims, such as submitting a claim based on misleading circumstances or exaggerating a genuine claim ([Belias et al., 2019](#bookmark=id.32hioqz)). These fraudulent practices not only burden insurance companies financially but also result in higher premiums for honest policyholders . To combat insurance fraud, deterrence and detection methods are employed, including specialized investigation units, statistical analysis of claims information, and fraud detection techniques [(Feinman, 2016)](#bookmark=id.3fwokq0). [Figure 1](#bookmark=id.ptgylhgsnmwm) below shows an overview of the process in common claims management for insurance companies



*Figure 1:* *Illustration of the measurement of process times and quantities along the core stages of claims management. Source: (*[*Mahlow & Wagner, 2016*](#bookmark=id.o07kok8w62ce)*)*

In automotive insurance, several techniques exist to identify fraudulent activities, which can occur either in a skilled or unskilled manner ([Ajay et al., 2022](#bookmark=id.147n2zr)). The goal is to detect multiple frauds involving behavior changes, which can be challenging for traditional machine learning approaches ([Caruana & Grech, 2021](#bookmark=id.41mghml)). The current traditional methods used in insurance fraud detection are usually financial evaluation methods. These methods suffer from performance degradation caused by imbalanced data and fail to consider the low rankness prior in the intrinsic samples when outliers emerge ([Ali et al., 2015](#bookmark=id.1pxezwc)). Traditional models often struggle when handling imbalanced datasets, particularly in machine learning applications such as fraud detection. These models tend to produce suboptimal results for the minority class, which is vastly outnumbered by the majority class. Various techniques have been explored to address this issue, including resampling methods. Oversampling techniques have been found to be particularly effective in improving model performance on the minority class, resulting in higher accuracy and lower false negatives ([De Zarzà et al., 2023](#bookmark=id.vx1227)).



*Figure 2:* *Illustration of how the ML-driven approach uses AI & replaces rule-based methods that require human interventions to detect fraud. Source: (*[*Sinha, 2023*](#bookmark=id.jw52kwxd2tnf)*)*

Additionally, traditional statistics-based methods are found to be more cost-effective than machine-learning based methods in detecting automobile insurance fraud ([Benedek et al., 2023](#bookmark=id.23ckvvd)). AI-based fraud detection methods tested on a real database are also found to be less cost-effective than traditional statistical-econometric methods ([Benedek et al., 2023](#bookmark=id.ihv636)). Therefore, the disadvantages of traditional methods in insurance fraud detection include performance degradation due to imbalanced data, failure to consider low rankness prior, and lower cost-effectiveness compared to traditional statistics-based methods and AI-based methods ([Tongesai et al., 2022](#bookmark=id.2zbgiuw)) .

Deep learning models are generally faster than traditional model-based methods for data processing. Traditional model-based methods rely on mathematical formulations and can be sensitive to inaccuracies, leading to slower performance in complex or dynamic systems ([Shen et al., 2022](#bookmark=id.206ipza)). On the other hand, deep learning models learn their mapping from data and can operate in complex environments, resulting in faster processing. However, deep learning models typically require large training sets and computational resources, which can limit their applicability in certain scenarios ([Kabir, Nagrecha., 2021](#bookmark=id.2lwamvv)). Therefore, while deep learning models are faster, they may not always be the most efficient choice for all data processing tasks.

Hence, we have put forth a strategic plan to address the limitations of conventional models and integrate them with deep learning models in the subsequent manner. Firstly, we will select the conventional models that yield the most optimal outcomes. In our analysis of research articles, it was observed that Random Forest and Support Vector Machine, two traditional models, exhibit superior performance. Hence, secondly, our focus will be on enhancing the challenges faced by these two models. Our direction is to integrate contemporary deep learning models with the Stacking technique to enhance the accuracy of result prediction.

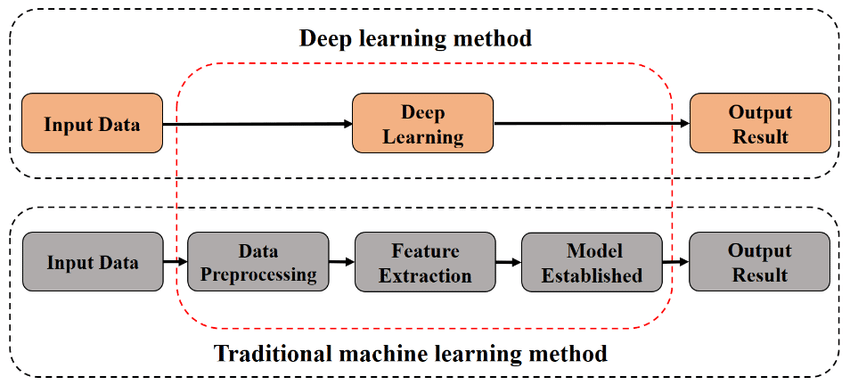


Figure 3. *Comparison of deep learning and traditional machine learning methods. Source:* [*(Chen et al., 2019)*](#bookmark=id.9fvirpaagd67)

## 1.3. Related works

In prior scholarly investigations, it has been ascertained that two distinct methodologies have been employed to identify instances of insurance fraud. The initial approach encompasses traditional models such as Logistic Regression, Support Vector Machine, Naive Bayes, and Random Forest. These traditional models have been extensively utilized in the field to uncover instances of fraudulent activities within the insurance industry. In this paper, we aim to examine the Support Vector Machine (SVM) model by employing the kernel trick and radial basis function techniques ([Rustam et al., 2018](#bookmark=id.111kx3o)). Additionally, our research also encompasses an investigation into the Random Forest model, which will be analyzed based on the K-Nearest Neighbor classification mechanism [(Li et al., 2018](#bookmark=id.19c6y18)). Secondly, in order to embrace more contemporary approaches, we delve into the realm of deep learning models, employing methodologies like reLu, Batch Normalization, and Drop out. Finally, we lay down the theoretical foundation of data processing techniques or feature selection to optimize the problem-solving approach. Through this exploration, we aim to overcome the constraints imposed by conventional models and discover novel solutions.

## 1.4. Thesis Approach and Contribution

The insurance industry faces significant challenges, particularly in dealing with fraudulent claims, which result in substantial financial losses. Currently, conventional financial evaluation methods are predominantly used to assess fraud cases. Recognizing the urgency and importance of addressing this issue, we have chosen this topic as an academic research theme to serve the property insurance sector and the insurance domain in general. Our research study aims to enhance fraud detection performance by constructing a Stacking model. We developed and assessed conventional models and a self-built deep learning model using the Keras network. We evaluated our methodology using a dataset obtained from Kaggle, which included one thousand insurance claim complaints regarding car collisions in seven states of the United States in 2015. The dataset consists of 40 variables and 1000 observational samples. Among these complaints, a subset of fraudulent claims was identified. Our main objective with this dataset was to classify legitimate insurance contracts from deceptive ones. To evaluate the models, we first identified two baseline models, Random Forest and Decision Tree, that exhibited superior performance. Subsequently, we constructed a Stacking model using the meta-learning approach, referring to the entire process as the Keras deep learning model. Insurance companies have the potential to reduce losses caused by fraudulent activities by leveraging essential data and implementing the insights derived from this research study to forecast compensation claims. This methodology surpasses the constraints of conventional techniques, enhancing both reliability and effectiveness. Subsequent investigations will concentrate on refining and enhancing the performance of the model.

The utilization of our findings derived from the application of the novel framework for detecting fraud in automobile insurance can serve as a means to tackle the challenges of sustainable development. By improving the accuracy and efficiency of fraud detection, insurance companies can potentially reduce the financial losses associated with fraud. This can be achieved by fostering responsible and sustainable practices within the insurance industry, promoting economic stability, and ensuring the long-term viability of insurance services for both individuals and communities. The implementation of our innovative model for detecting fraud in automobile insurance can contribute to the achievement of SDG9 by stimulating industry innovation, improving efficiency, and fostering technological advancement. Furthermore, it aligns with SDG16 by preventing fraudulent activities, strengthening institutions, and promoting peace and justice through transparent and accountable practices within the insurance sector.

## 1.5. Thesis outline

The subsequent sections of the research are structured in the following manner. Part two commences by introducing and analyzing both traditional and deep learning models used in previous research. Subsequently, a strategy for developing research directions will be formulated based on the synthesis of these models. Part three provides a comprehensive description of the dataset, highlights any limitations in the data, and outlines the testing and implementation methods employed in our model. This section will delve into the details of our methods and experiments. Part four is dedicated to presenting the results obtained from our model. Part five engages in a discussion of the findings and emphasizes the significance of the research. Finally, the concluding section summarizes the study and proposes avenues for future research.

# Previous analysis

In this study, we focus on defining metrics and analyzing both traditional and deep learning models. Firstly, we examine the metrics used, including accuracy, recall, F1 score, and precision, and explain the rationale behind our selection of each metric. Next, we define and analyze the strengths and weaknesses of the two selected traditional models. Subsequently, we delve into the formulas and methods of deep learning models, including ReLU, Batch Normalization, and Dropout, along with an analysis of the limitations and proposed methods in the study. We also define and introduce in detail the techniques we employ in the study. This will provide an overview of how the various components of this research are interconnected to address the issue of detecting fraud in automobile insurance claims.

## 

## 2.1. Evaluation Metrics

The metric commonly used for evaluating models is accuracy because it reflects the proportion of correct predictions compared to the actual outcomes. However, in some cases, it is necessary to consider other metrics to provide a more objective and accurate assessment. Our study used Precision, Recall, Accuracy, F1 Score, for classification evaluation. These metrics are calculated as follows:

This is particularly relevant in the project we are undertaking, as we have identified an issue during experimentation. Despite achieving an accuracy of over 73%, a relatively high performance on experimental results, some models exhibit a bias towards the majority class in their predictions. This results in many observations being misclassified, with an inability to detect fraud, indicating that accuracy is not the sole metric for evaluating the problem.

Therefore, the commonly used metric - accuracy is no longer the sole metric used in the context of imbalanced data, and AUC is a popular performance evaluation metric in imbalanced classification [(Seng et al., 2021)](#bookmark=id.3l18frh). Evaluating model performance using accuracy alone has been criticized when the original dataset is imbalanced [(Malhotra & Lata, 2021)](#bookmark=id.3tbugp1). Thus, the evaluation of models by the team will also consider metrics such as recall, precision, F1-score, MCC, or AUC-ROC, rather than solely relying on accuracy.

Finally, to visualize the research results, confusion matrices are used to clearly depict the actual performance that the model provides. Based on this foundation, we have a more comprehensive understanding of how experimental models perform to achieve appropriate adjustments. This is an important tool for evaluating the practical performance that models deliver, especially for binary classification problems like insurance fraud detection, which is the subject of our research.

## 2.2. Traditional Methods

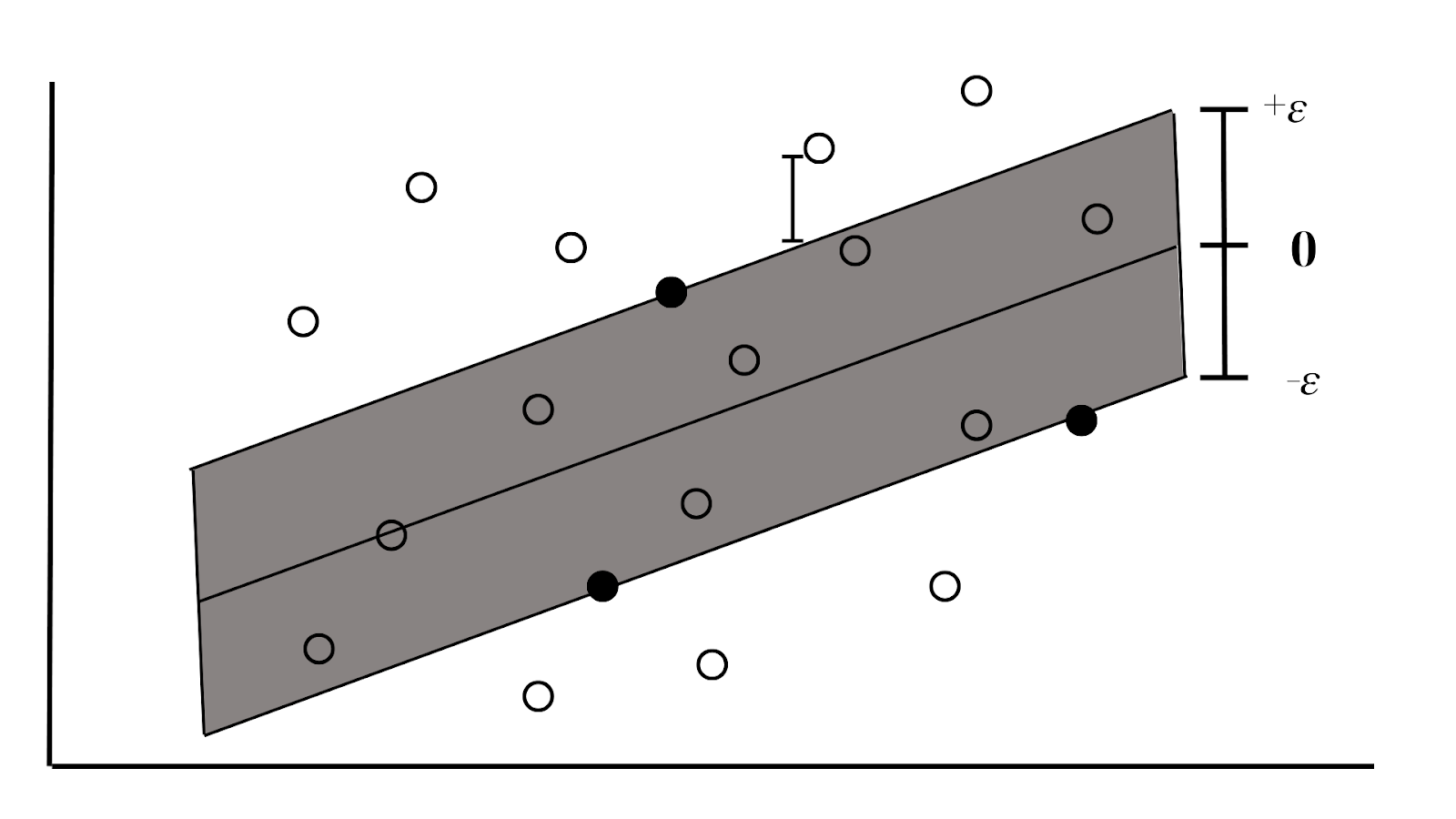
### 2.2.1. Overview

Significant progress has been made in the field of insurance fraud detection, as researchers continuously seek new approaches to improve fraud prevention systems. A recent research study aims to contextualize its findings by comparing its methods and results with previous literature on the topic. The study includes a historical view of fraud detection, acknowledging the significance of conventional approaches and referencing prior research that has employed machine learning and data mining methods ([Gomes et al., 2021](#bookmark=id.1v1yuxt)). The new model uses Logistic Regression, Support Vector Machine, and Naive Bayes, along with the Boruta algorithm for feature selection. Comparative evaluation shows that SVM achieves high accuracy, while Logistic Regression exhibits exceptional recall and sensitivity ([Kumar & Gopal, 2010](#bookmark=id.2u6wntf)). These findings align with Kumar et al.'s research but highlight the ongoing debate about the optimal fraud detection model, suggesting a requirement for further investigation.

Gomes et al.'s research is limited due to its narrow concentration on auto insurance data from the U.S. in 2020, leading to questions about how applicable the findings are to present-day market circumstances. Additionally, there is limited consideration of prediction models, prompting a need for exploration of other models that may perform better in fraud detection. Furthermore, the study's perspective predominantly from an auto insurance company viewpoint indicates potential for future research to incorporate the viewpoints of policyholders and auto repair workshops to broaden perspectives. While Gomes et al.'s work offers unique methodologies, its limitations emphasize the ongoing need for diverse perspectives and further exploration of alternative models to advance the field of insurance fraud detection.

### 2.2.2. Traditional model analysis

SVM is a machine learning algorithm for classification and regression tasks. In classic regression, the goal is to minimize the least squared errors by finding the regression coefficients 𝑤. The basic concept of SVC involves creating a tolerance band around the regression line within which all data points are expected to lie. The objective of SVC is to minimize the coefficients through l2-norm while handling errors using constraints with a maximum error ε. To allow larger than ε errors, we add a slack parameter 𝜁𝑖 in our model. This allows points to lie outside the band but we want them to remain as close to it as possible. [Figure 4](#bookmark=id.xcabxpynnf8t) illustrates the SVC paradigm with slack variables.

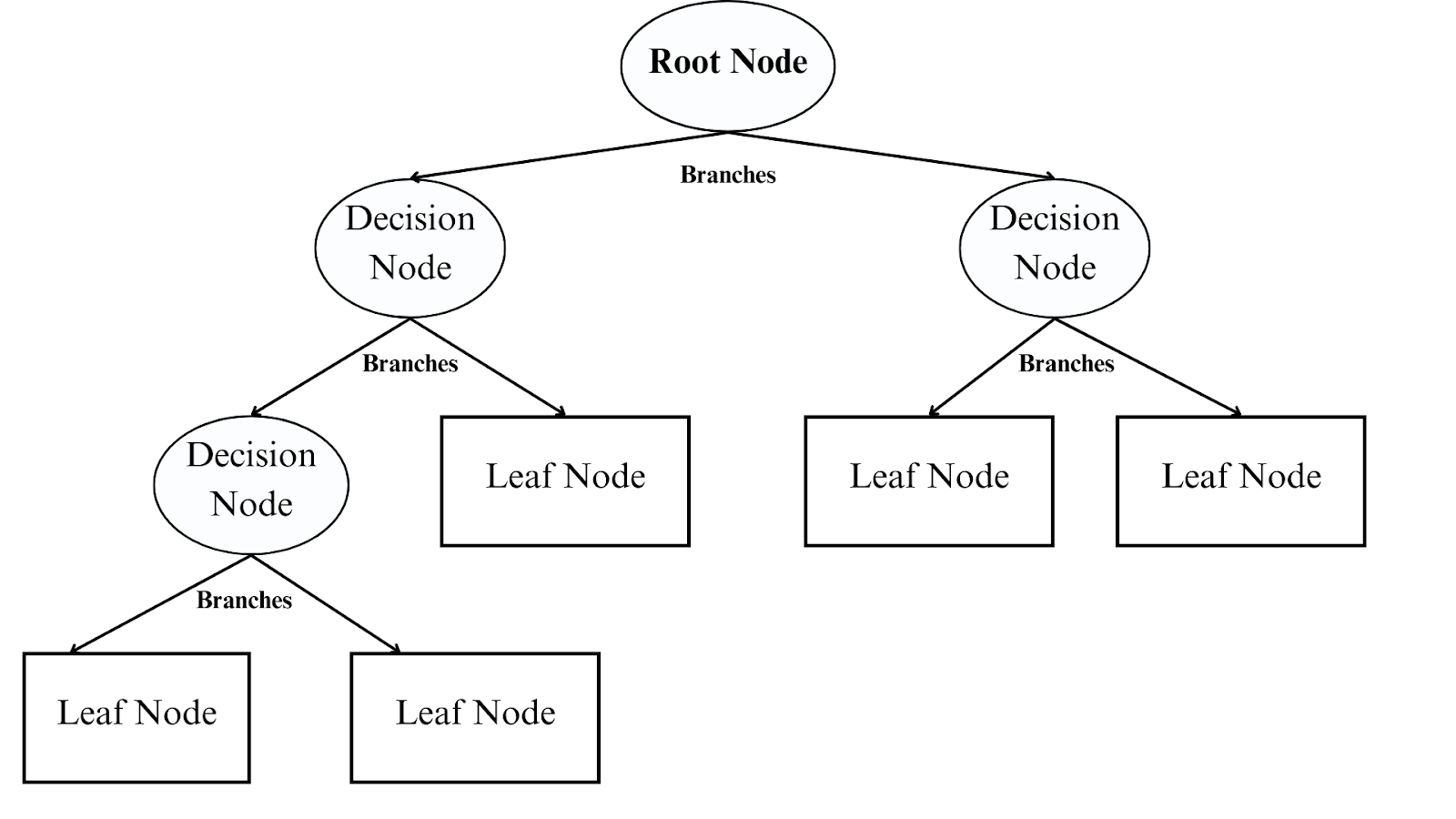


*Figure 4: The Support Vector Paradigm. Source: (*[*Poufinas et al., 2023c*](#bookmark=id.1mrcu09)*)*

The ε tolerance band around the regression line is represented by the gray area. Data points within this zone have no impact on the objective function, while those outside it contribute 𝜁𝑖 to the minimization objective function. The support vectors are identified as marginal black points located within this gray zone and play a key role in defining the position of the regression line.

When a problem cannot be handled with a linear classifier, the kernel trick is used to transform the dimensionality of the optimization space during training. Data points from an n-dimensional variable space (in our example, two dimensions) are projected into a higher dimensional feature space where the regression hyperplane allows for acceptable error. If the kernel function is non-linear, then the resulting SVC model is also non-linear. In the study of auto insurance fraud detection, [Nian et al.](#bookmark=id.37m2jsg) introduced a novel method utilizing advanced unsupervised techniques, termed “Unsupervised Spectral Ranking for Anomaly” (SRA), to construct a fraud diagnosis model. This approach combines SVM with spectral optimization techniques and classification strength measurement computation, without requiring labels in the data. Instead, it relies on the Laplacian matrix to identify anomalous cases in auto insurance data. In this manner, SVM is employed within SRA to optimize the process of identifying anomalies, while the choice of ranking reference aids in evaluating the model's performance. The results of this study indicate that the SRA model outperforms existing outlier-based fraud detection methods, offering significant promise for practical application in diagnosing fraud within the auto insurance industry. ([Nian et al .,2016](#bookmark=id.37m2jsg))

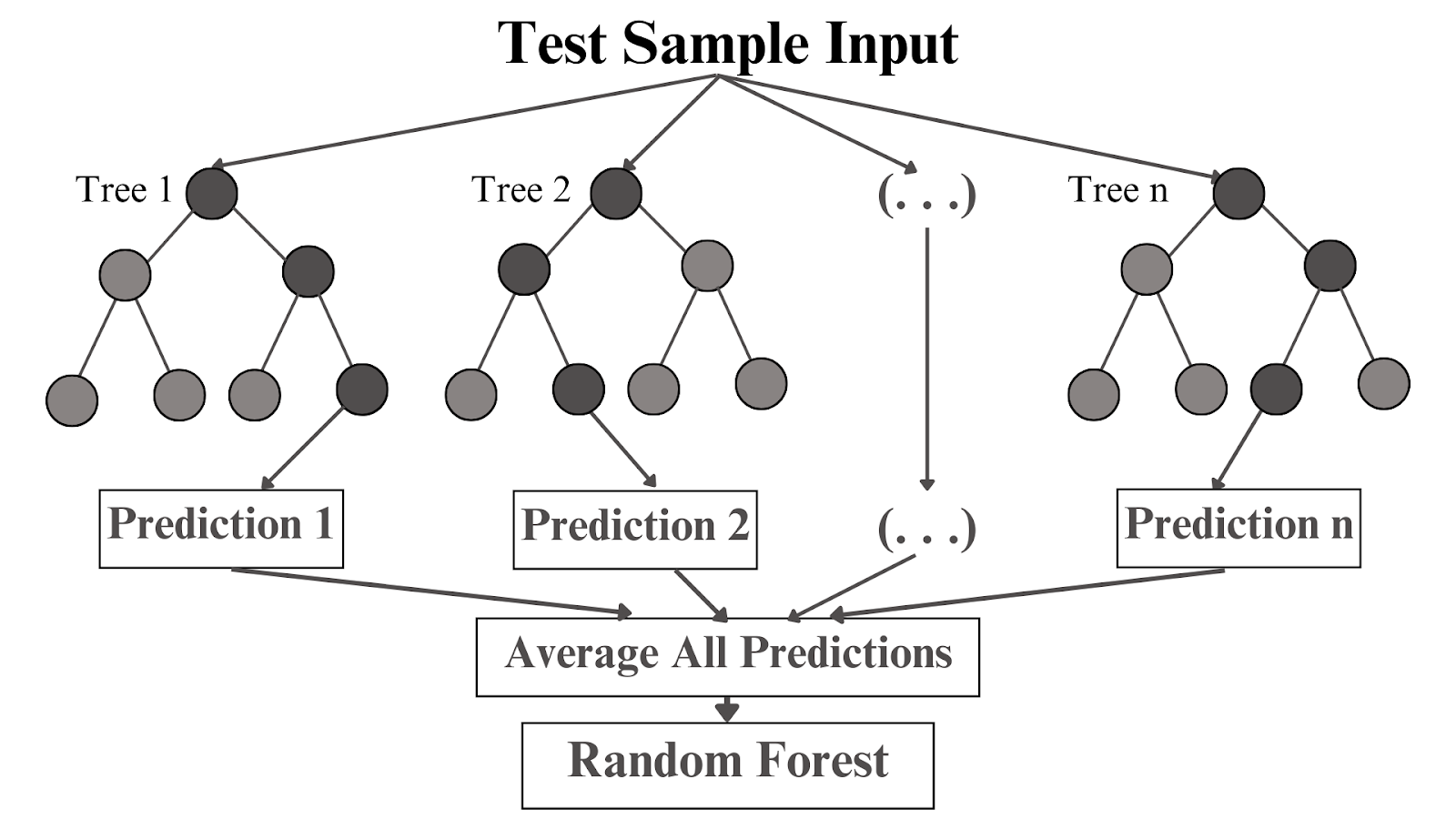
Decision Trees are a supervised machine learning algorithm that is used for both classification and regression tasks. It works by recursively partitioning the data based on the most informative features. They are flowchart-like top-down structures of nodes and branches; see [Figure 5](#bookmark=id.ihy6u1fk0hze). In regression tasks, the decision tree predicts the value of the target variable by averaging the values of the training data points that fall into the same leaf node.



*Figure 5: A graphical representation of Decision Trees. Source: (*[*De Sá et al., 2016*](#bookmark=id.t8au92g02dam)*)*

Every decision node is an if statement regarding one of the variables on the set. Every leaf node corresponds to a final value for the regression. The top node is the root node representing the complete dataset. The decision nodes are if statements on one variable of the dataset. The two branches below each decision node split the dataset into two subsets: a subset where the decision node is true and a subset where the decision node is false. The nodes that do not split any further are called leaf (or terminal) nodes and depict the final outcomes of the decision-making process (in our case, a value for the regression procedure). There is research exploring the use of Naïve Bayesian classification network and Decision Tree-Based algorithms to classify auto fraud claims as fraudulent or honest ([Bhowmik, 2011](#bookmark=id.1hmsyys)). They assessed the model performance using performance parameters and utilized rule-based classification for visualization [(Bhowmik, R., 2011)](#bookmark=id.1hmsyys).

The Random Forests model employs a bootstrapping-aggregating algorithm known as bagging, as illustrated in [Figure 6](#bookmark=id.6yo0xv1337tu). This approach combines numerous decision trees. In each tree, a different randomly selected replacement subsample, equivalent in size to the initial dataset, is used for training. Additionally, a randomly selected subset of the initial independent variables is utilized. The model's ability to generalize to unknown data is estimated using observations not included in the training process, commonly referred to as out-of-bag data in machine learning terminology. Consequently, the Random Forests algorithm incorporates a stochastic process for selecting both the training observations (rows of the data matrix) and the independent variables (columns of the data matrix). For regression tasks, predictions are made by calculating the average value of the target variable from the data points within each leaf node. The Random Forest model is enhanced by combining the following methods: considering the classification mechanism from the perspective of K-Nearest Neighbor, replacing the majority voting mechanism to avoid information loss from out-of-bag (OOB) samples, employing Principal Component Analysis transformation to convert the data into PCA space for building an efficient insurance fraud prediction model ([Li et al., 2018](#bookmark=id.19c6y18)).



*Figure 6: The depiction of a 600-tree Random Forest. The average of the 600 predictions is the final prediction of the Random Forest. Source: (*[*Team, 2023*](#bookmark=id.wqc9xekmbh4l)*)*

In this study, we decided to use two traditional models, Random Forest and Support Vector Machine (SVM), to build our model. This is because they often perform well when working with small datasets. Although they may face challenges when dealing with large datasets, Random Forest and SVM are still capable of effectively handling large amounts of data. However, these traditional models also have some drawbacks. They require the selection of important features to achieve good performance, which can be time-consuming. Additionally, Random Forest and SVM may perform poorly when dealing with datasets with too many observations, potentially leading to overfitting or underfitting phenomena.

## 2.3. Deep learning model

### 2.3.1. Formula of reLu, Batch Normalization and Drop out

In order to enhance the deep learning model's accuracy in predicting outcomes, techniques such as ReLU (Rectified Linear Unit), Batch Normalization, and Drop-out are employed in machine learning models to improve performance and stabilize the training process.

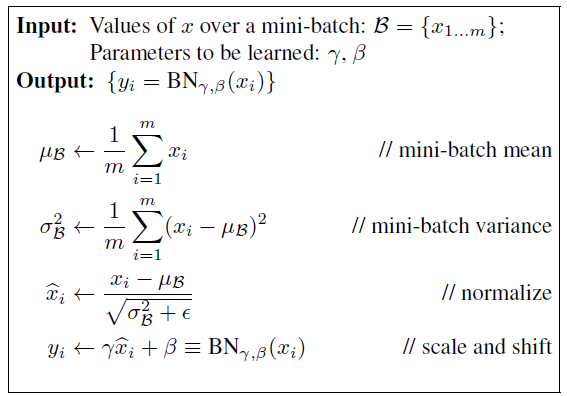
Definition and formula of ReLU

In this article, we will focus solely on the ReLU activation function as it remains the default and most popular choice for use in the majority of deep learning tasks [(Krishnamurthy, 2022)](#bookmark=id.dop8t0sfwy6).ReLU is a popular activation function used in machine learning models and neural networks. The ReLU function helps address the issue of vanishing gradients and is commonly preferred in the hidden layers of neural networks. The ReLU function is represented by the formula:

f (x) = max (0,x)

Where x is the input to the function. max is the function that takes the maximum value between 0 and x. ReLU returns the value of x if it is positive and returns 0 if x is negative. This function preserves the values of positive numbers and eliminates negative values, stimulating neuron activation and enhancing the non-linearity of the model. ReLU involves simple calculations, addresses the issue of vanishing gradients, and has the ability to improve the performance of neural networks in various scenarios. However, it also has the drawback of potential “dead neurons” when the input value is negative, impacting the learning capability of the model in certain cases.

Definition and formula of Batch Normalization  
Batch Normalization is a technique that normalizes the output values of hidden layers in a neural network before they are fed into the activation function. This helps alleviate the issue of distributional changes in the outputs between layers during the training process. Batch Normalization is typically applied after the linear layer and before the activation function in the architecture of a neural network.



*Figure 7: Batch Normalization formula. Source: (*[*Kratzert, 2016*](#bookmark=id.heqgk3ge8aes)*)*

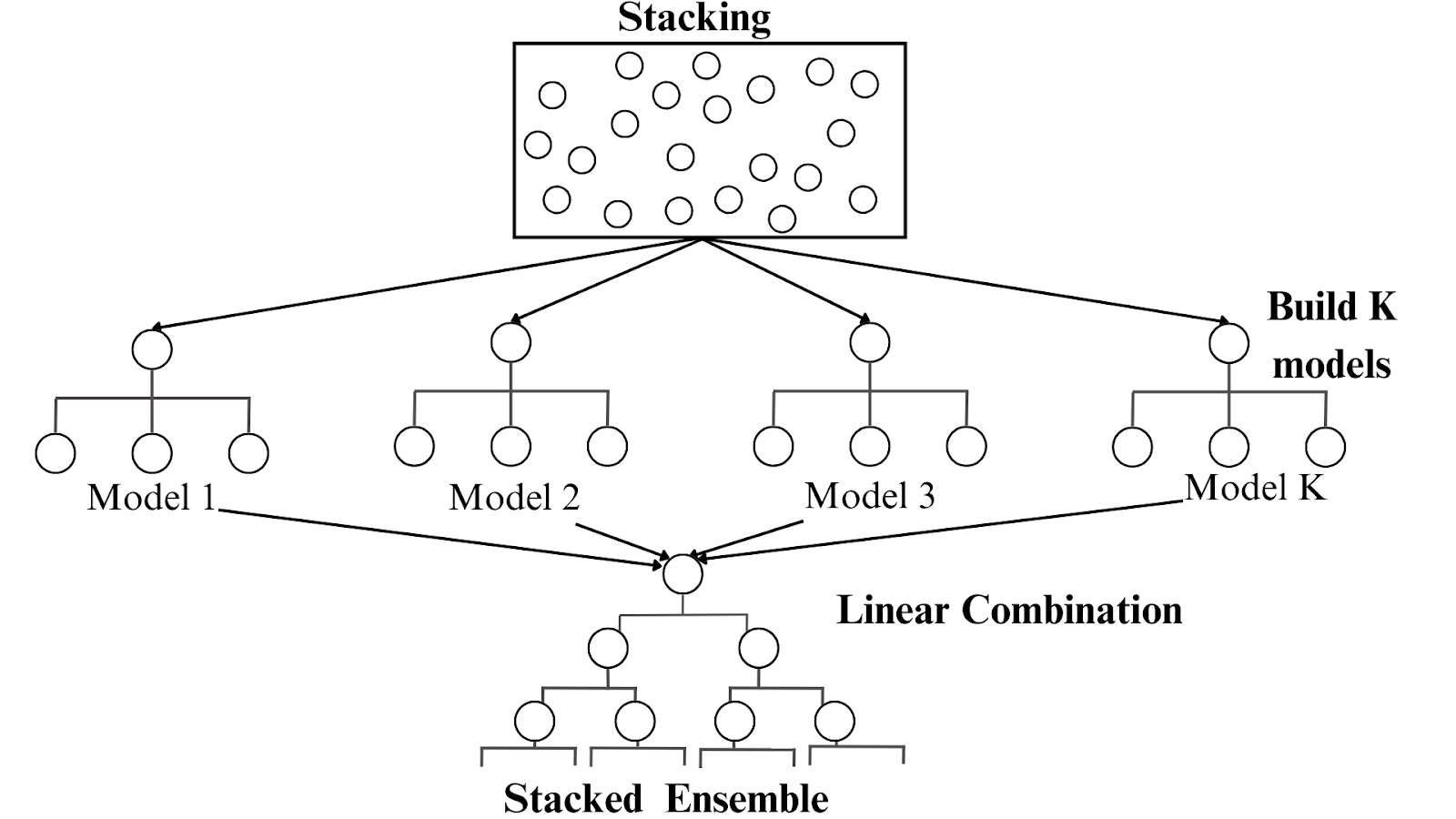
Batch Normalization helps improve the convergence of the model, mitigates the issue of distributional changes between mini-batches, reduces the problem of vanishing gradients, and may even allow the use of larger learning rates during training. This makes the training process of the model more stable and contributes to achieving better performance.

Definition and formula of Drop-out

When applying Drop-out in a layer, each neuron has a probability of being “dropped out” in each backward pass. This probability is usually pre-set as a hyperparameter of the model. During the forward pass, each neuron is retained with a probability of 1−p and is multiplied by a scaling factor of 1/(1−p). This helps maintain the expected value of the neuron.

When the model has been trained and is used for prediction, no neurons are dropped out, and the value of each neuron is multiplied by 1−p to maintain the output ratio. This acts as randomly “dropping out” several neurons in each layer during training, preventing the model from overly relying on specific relationships in the training data. This helps prevent overfitting and makes the model generalize better to new data. Dropout is commonly applied after fully connected and convolutional layers in neural networks, enhancing generalization and reducing the risk of overfitting.

### 2.3.2 Stacking learning model



*Figure 8: Stacking model process. Source: (Vũ,H.,2021)*

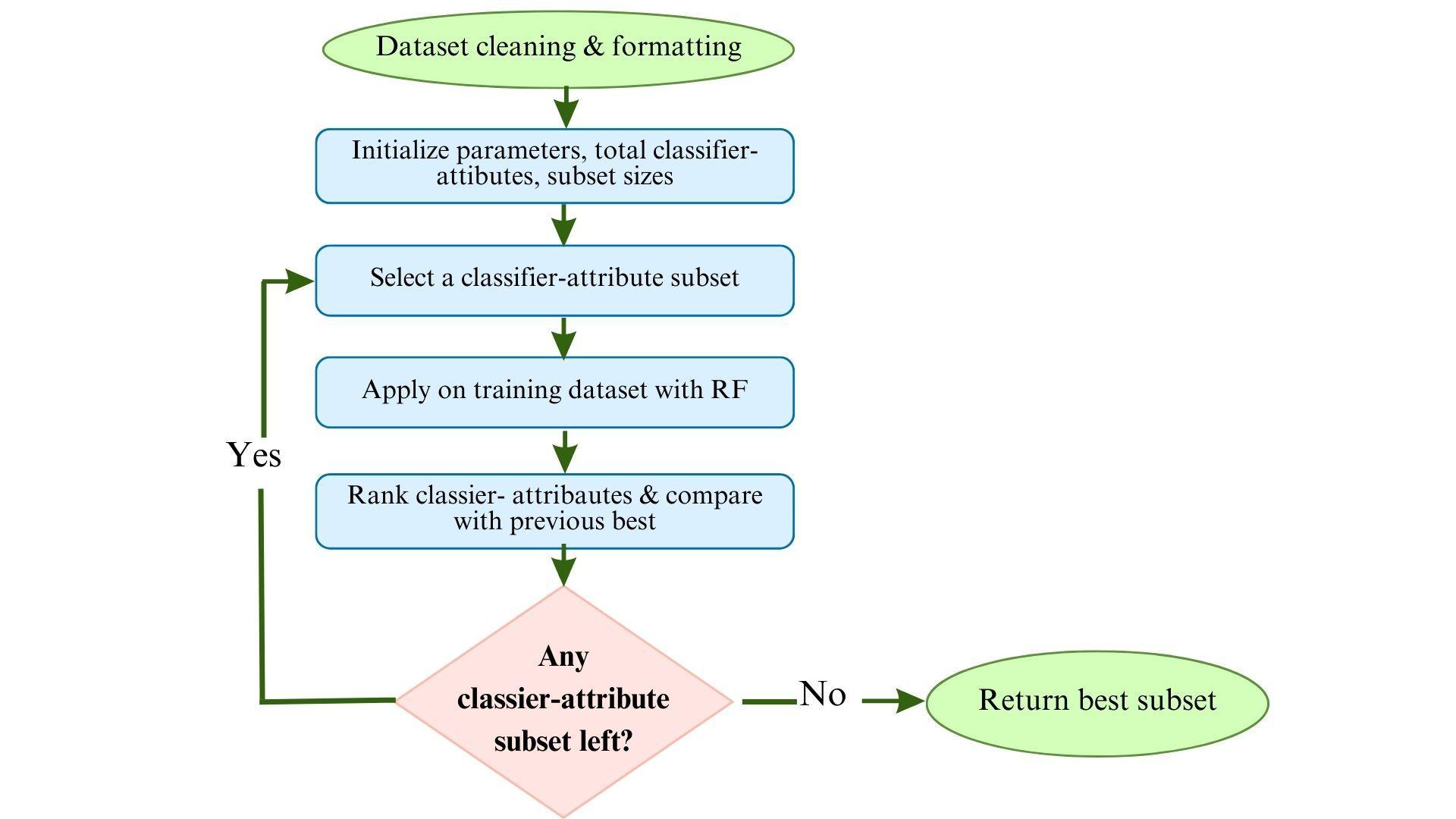
Similar to traditional and deep learning models, evaluation metrics such as accuracy, precision, recall, and F1-score can be used to assess the performance of the stacking model. Therefore, we chose the stacking model. This is a common method in machine learning, not only as a tool to combine multiple different models but also as a way to enhance the system's ability to make accurate predictions. The process of this model begins by splitting the data into normal training and testing sets. Next, a series of base models are selected. Each base model is trained independently on the training data. Then, they make predictions on the test data. These predictions from each base model are then used as input features for a meta-model. The meta-model is then trained on the predictions of the base models on the test data. During this process, the meta-model learns to combine the predictions from the base models to produce a final prediction with higher predictive capability.

One strength of the stacking model is its ability to leverage the best models from a diverse set. Instead of relying on a single model, the stacking model can select the optimal models from a range of models to provide the best possible predictive performance. Particularly, the stacking model can help address issues such as overfitting and underfitting by leveraging the strengths of multiple models. This helps balance the exploration of complex data patterns and maintaining the generalizability of the model.

With the desire to improve prediction performance and flexibility in selecting the best models, the stacking model is an important tool in machine learning, especially in complex problems such as detecting fraud in auto insurance.

## 2.4. Strategies to improve performance of model

### 2.4.1. Recursive Feature Elimination



*Figure 9: Workflow Diagram of Recursive Feature Elimination (RFE).*

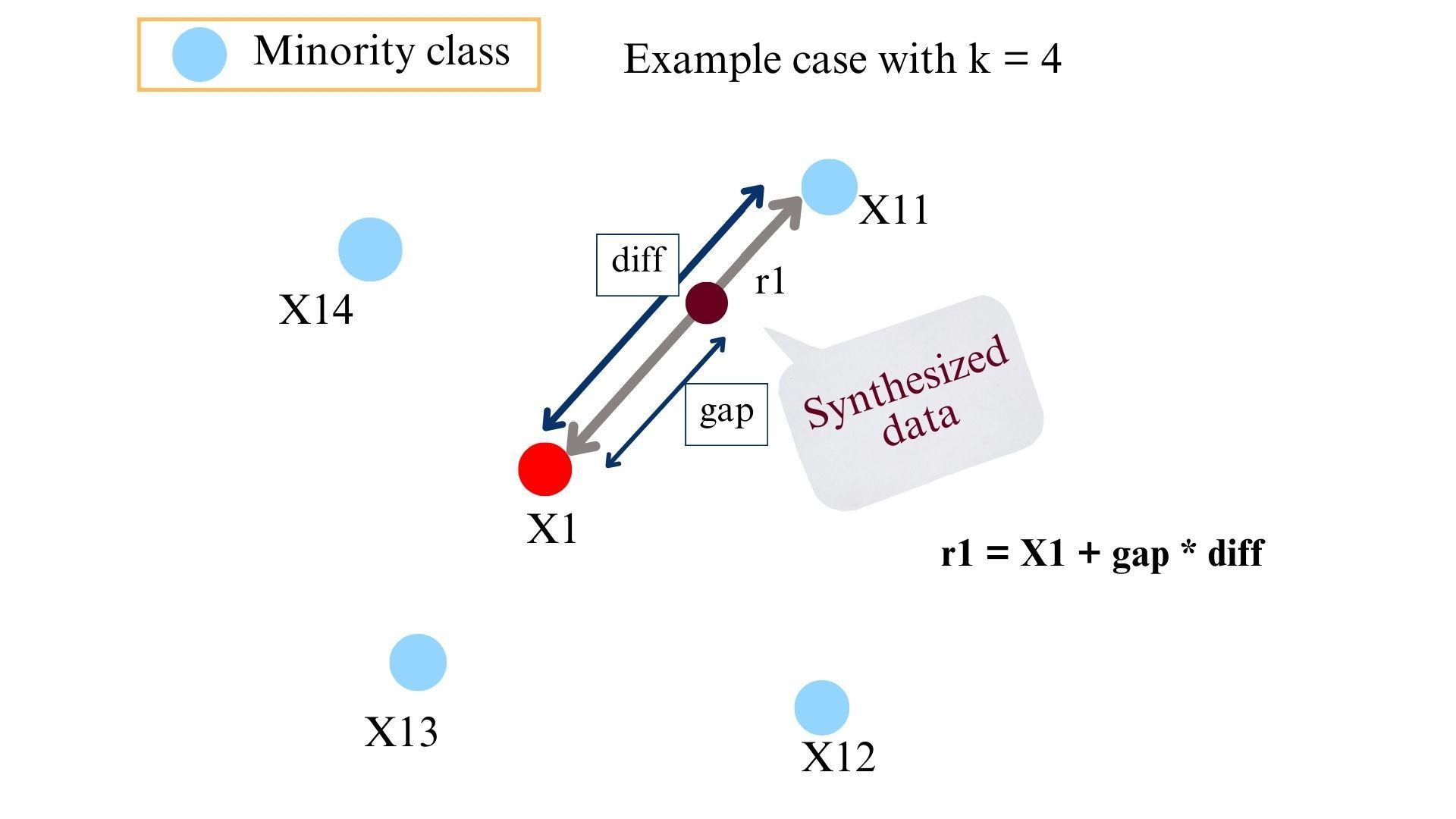
*Source: (*[*Faysal et al., 2022*](#bookmark=id.ccwjjnsxqrv7)*).*

Recursive Feature Elimination (RFE) is a dimensionality reduction method commonly used in machine learning model construction. It aims to select the most important features from a large feature set by sequentially removing the least important features from the dataset until the number of remaining features reaches a predefined threshold. RFE is often employed with traditional models such as Linear Regression, Support Vector Machines (SVM), or tree-based models like Decision Trees.

In traditional RFE, the method iteratively eliminates features that contribute the least to the model”s performance, typically measured by a drop in classification accuracy after building a classification model. RFE is a powerful technique for feature selection, as it helps improve model performance by reducing dimensionality and focusing on the most relevant features.

### 2.4.2. class\_dict\_weight and SMOTE

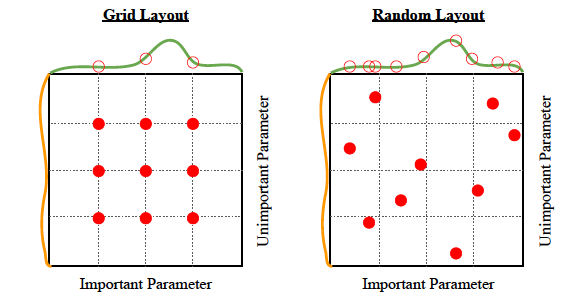
Class\_dict\_weight and SMOTE are two popular techniques used in handling imbalanced data in machine learning, especially in classification problems where the number of samples belonging to different classes is uneven. class\_dict\_weight is a method based on adjusting the weights of different classes in the machine learning model. It enhances the influence of samples from the minority class by increasing their weights during training. However, class\_dict\_weight does not generate new data samples but only adjusts weights to make the model focus more on correctly classifying samples from the minority class. On the other hand, SMOTE is a technique that creates synthetic data samples from existing samples in the minority class.



*Figure 10: Synthetic Minority Over-sampling Technique. Source: GitHub*

SMOTE is an algorithm used to address the issue of imbalanced data in machine learning. This algorithm is utilized to generate synthetic data samples by combining features of the nearest neighbors within the same class. It helps balance the ratio between classes in the data without directly modifying the structure of the model, thus improving the model's generalization ability during training. SMOTE is commonly employed in complex machine learning models that require flexibility, such as neural networks.

### 2.4.3. Hyperparameter tuning with RandomSearch



*Figure 11: RandomSearch layout. Source: (*[*Grid Search Vs Random Search, n.d*](#bookmark=id.vkt5hdxwx6pe)*.)*

RandomSearch is a method for hyperparameter tuning in machine learning models. In RandomSearch, values for hyperparameters are randomly selected from a predefined distribution instead of exhaustively searching through all possible values of the parameters as in the GridSearch method. This approach helps reduce computational time compared to GridSearch while still being able to find optimal values for the model's hyperparameters.

Additionally, RandomSearch explores the parameter space in a random and comprehensive manner, allowing for the discovery of optimal values for model parameters in a generalized and efficient way, thereby improving the performance of the proposed model.

# Materials and Methods

## 3.1. AutoMobile Claims Dataset and Challenges

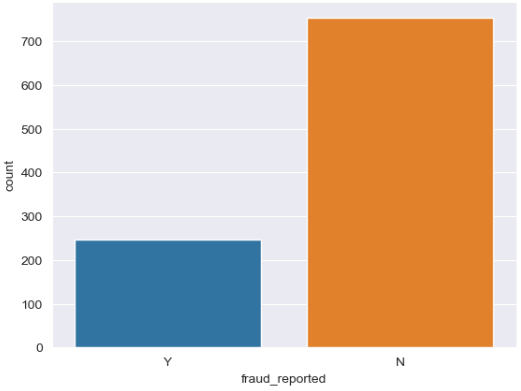
In this study, we evaluated our approach on the dataset of automobile insurance claims sourced from Kaggle. This dataset comprises 40 variables and 1000 observational samples. This dataset focuses on classifying valid insurance contracts and fraudulent insurance contracts. You can find the dataset at the following link: [Automobile Insurance Claims Dataset](#bookmark=id.28h4qwu).

[Table 1](#bookmark=id.26in1rg) will illustrate the variables of the dataset, providing us with an overview of important information regarding the characteristics of automobile insurance claims and the frequency of different categories within each variable. By analyzing this data, we gain deeper insights into trends related to various aspects of insurance claims, such as the types of claims submitted, individual demographics of policyholders, and geographical distribution of claims. This information is crucial for insurance companies to make decisions regarding risk assessment, fraud detection, and policy pricing strategies.

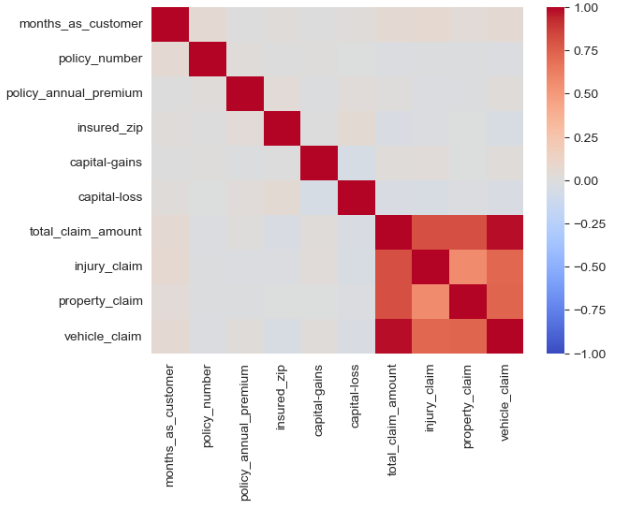
*Table 1: Descriptive statistics for categorical variables in the dataset.*

|  | Fraud | | | No Fraud | | |
| --- | --- | --- | --- | --- | --- | --- |
|  | Obs | Mean | Std. Dev. | Obs | Mean | Std. Dev. |
| Months\_as\_customer | 247 | 208.08 | 119.82 | 753 | 202.60 | 113.57 |
| Age | 247 | 39.14 | 9.65 | 753 | 38.88 | 8.97 |
| Policy\_number | 247 | 533030.21 | 256334.03 | 753 | 550571.30 | 257323.90 |
| Policy\_deductable | 247 | 1151.82 | 628.12 | 753 | 1130.81. | 606.77 |
| Policy\_annual\_premium | 247 | 1250.24 | 253.26 | 753 | 1258.43 | 241.25 |
| Umbrella\_limit | 247 | 1336032.39 | 2494798.91 | 753 | 1023904.38 | 2225209.09 |
| Insured\_zip | 247 | 503637.96 | 70487.50 | 753 | 500419.54 | 72123.98 |
| Capital\_gains | 247 | 24193.52 | 27766.25 | 753 | 25432.01 | 27918.46 |
| Capital\_loss | 247 | -27522.67 | 27603.23 | 753 | -26554.58 | 28280.49 |
| Incident\_hour\_of\_the day | 247 | 11.70 | 6.89 | 753 | 11.63 | 6.98 |
| Number\_of\_vehicles\_involve | 247 | 1.93 | 1.05 | 753 | 1.81 | 1.01 |
| Bodily\_injuries | 247 | 1.04 | 0.83 | 753 | 0.98 | 0.82 |
| Witnesses | 247 | 1.58 | 1.07 | 753 | 1.46 | 1.12 |
| Total\_claim\_amount | 247 | 60302.11 | 20746.28 | 753 | 50288.61 | 27575.19 |
| Injury\_claim | 247 | 8208.34 | 4550.31 | 753 | 7179.23 | 4961.20 |
| Property\_claim | 247 | 8560.12 | 4631.74 | 753 | 7018.88 | 4828.92 |
| Vehicle\_claim | 247 | 43533.64 | 14849.39 | 753 | 36090.49 | 19698.05 |
| Auto\_year | 247 | 2005.19 | 6.07 | 753 | 2005.08 | 6.00 |
| c39\_ | 247 | N/A | N/A | 753 | N/A | N/A |

The distribution of data between the two labels in our study reveals an imbalance in the dataset. The valid insurance contracts (753 instances) outnumber the fraudulent insurance contracts (247 instances), approximately three times as much. This imbalance in fraudulent insurance contracts poses challenges in detecting fraudulent contracts based on limited data.

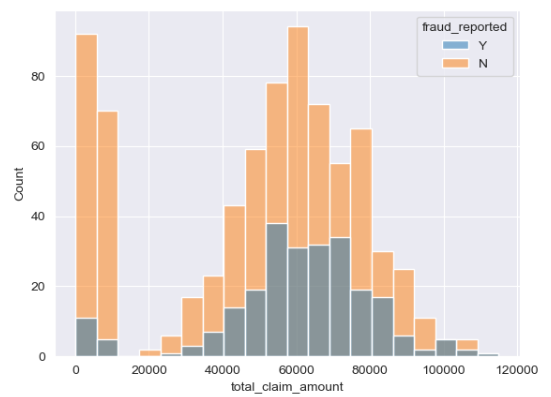


*Figure 12: The distribution of data in the Automobile Insurance Claims Dataset. Source: Made by the author's group.*



*Figure 13: The correlation among variables in the Automobile Insurance Claims Dataset. Source: Made by the author's group.*

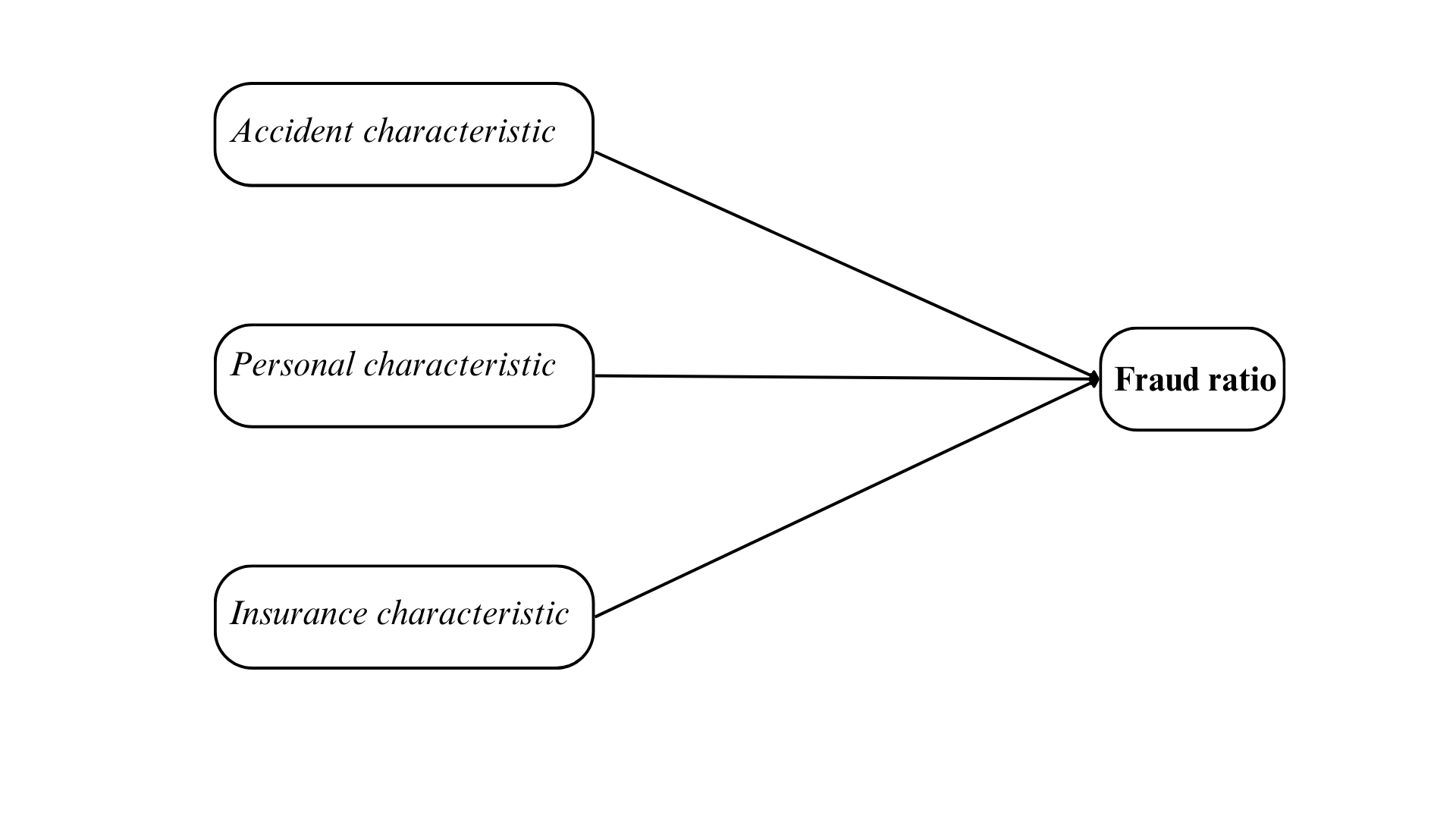
According to [Figure 14](#bookmark=id.1ksv4uv), only the variables “total\_claim\_amount”, “vehicle\_claim” and “property\_claim” exhibit a strong positive correlation, where an increase in one variable tends to correspond with an increase in the others. However, many variables show no significant correlation with each other, making it challenging to identify underlying relationships. Moreover, unrelated variables may introduce noise that hinders the accurate prediction of the model.



*Figure 14: Segment the total amount of regular requests. Source: Made by the author's group.*

The majority of insurance compensation claims tend to concentrate in the range from 40,000 to 80,000 USD. This may reflect the common nature of compensation requests in the automotive insurance sector. However, there are still some exceptional cases with total compensation amounts below 10,000 USD. This variability may introduce fluctuations and requires special attention to examine these situations. Based on the histogram, it is possible to identify “hotspots” in the data, where there is a sudden increase or decrease in the total compensation amounts. This indicates the presence of specific patterns or influencing factors. This observation provides an overall insight into the distribution of insurance compensation amounts and sets the groundwork for questioning how to develop more detailed analyses in the data preprocessing phase.

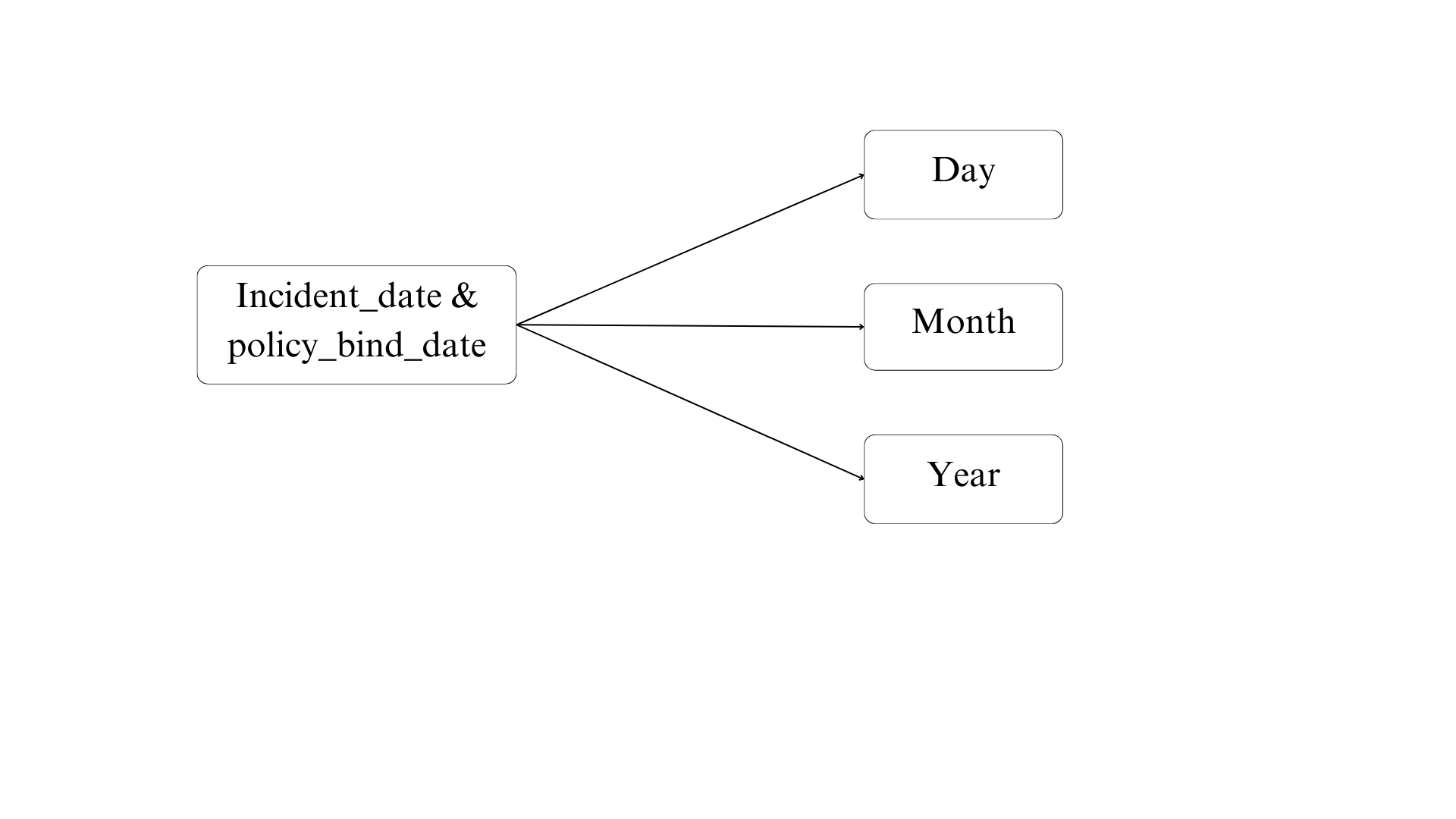
Based on the discoveries above, we have categorized the dataset into three feature groups: Accident Characteristics, Personal Characteristics, and Insurance Characteristics. This segmentation aims to analyze whether there is fraud in insurance contracts based on these distinct sets of features.



*Figure 15: Features for fraud detection. Source: Made by the author's group.*

In our research on the dataset of Automobile Insurance Claims, we have identified several limitations. Firstly, we attempted to update and expand the dataset to better reflect the current market reality. However, there are still discrepancies in the data compared to the actual market, particularly as the dataset was obtained from the year 2015 in 7 states of the United States. This limitation may impact the comprehensiveness and overall representation of the auto insurance industry. Secondly, the lack of correlation between variables can affect the predictive capability of the model. We addressed this issue through data exploration, analysis, and employing various data processing strategies.Thirdly, the dataset's small size and the limited compatibility and correlation among variables may pose challenges. This impacts data quality and could lead to overfitting if not managed carefully. Fourthly, there is an imbalance in the data between predicting fraud and non-fraud cases. This imbalance reduces the predictive performance of the model. These limitations should be carefully considered when evaluating and applying the model results. Exploratory Data Analysis (EDA) has been conducted to explore and gain a better understanding of the distribution of total claim amounts in the automobile insurance dataset.

The data preprocessing process has undergone several crucial steps to ensure the data is of sufficient quality and suitable for machine learning models. Firstly, we removed the “c\_39” column which contained missing values, not providing useful information. Additionally, we checked and handled missing values to ensure data integrity. Unnecessary columns such as customer IDs were eliminated.



*Figure 16: Information transformation. Source: Made by the author's group.*

Furthermore, we extracted date information (day, month, year) from the “incident\_date” and “policy\_bind\_date” columns and removed the original columns to optimize the data. For discrete variables like “age” and “number of witnesses”,We decided to categorize them into groups to enhance machine learning capabilities. Outliers were addressed using the IQR method, and subsequently, the data was normalized depending on the chosen strategy.

Analysis of correlation coefficients helped we gain a better understanding of the relationships between variables, enabling appropriate data transformations to optimize prediction models. Finally, an imbalance in the data was observed, with fraudulent claims accounting for less than 29% among the 1000 claims for compensation and complaints. This highlighted the need for a more accurate approach to addressing data imbalance.

In this study, we focus on referencing previous research on the Automobile Insurance Claims dataset from various sources. We emphasize the importance of this in enhancing our understanding of the automobile insurance domain and identifying future directions. By analyzing the methods, techniques, and results applied in previous studies, our aim is to develop new plans and strategies for our research.

| Title | Author | Method |
| --- | --- | --- |
| Prediction of Insurance Fraud Detection using Machine Learning Algorithms | Laiqa Rukhsar, Waqas Haider Bangyal, Kashif Nisar, Sana Nisar | Decision Tree |
| Machine Learning in Forecasting Motor Insurance Claims | Thomas Poufinas, Periklis Gogas, Theophilos Papadimitriou, Emmanouil Zaganidis | Random Forest |
| Use of Data Mining Techniques for Data Balancing and Fraud Detection in Automobile Insurance Claims | Slokashree Padhi, Suvasini Panigrahi | SVM-Balanced-  Ensemble (SBE) |

*Table 2: The previous proposed models in previous analysis in this topic.*

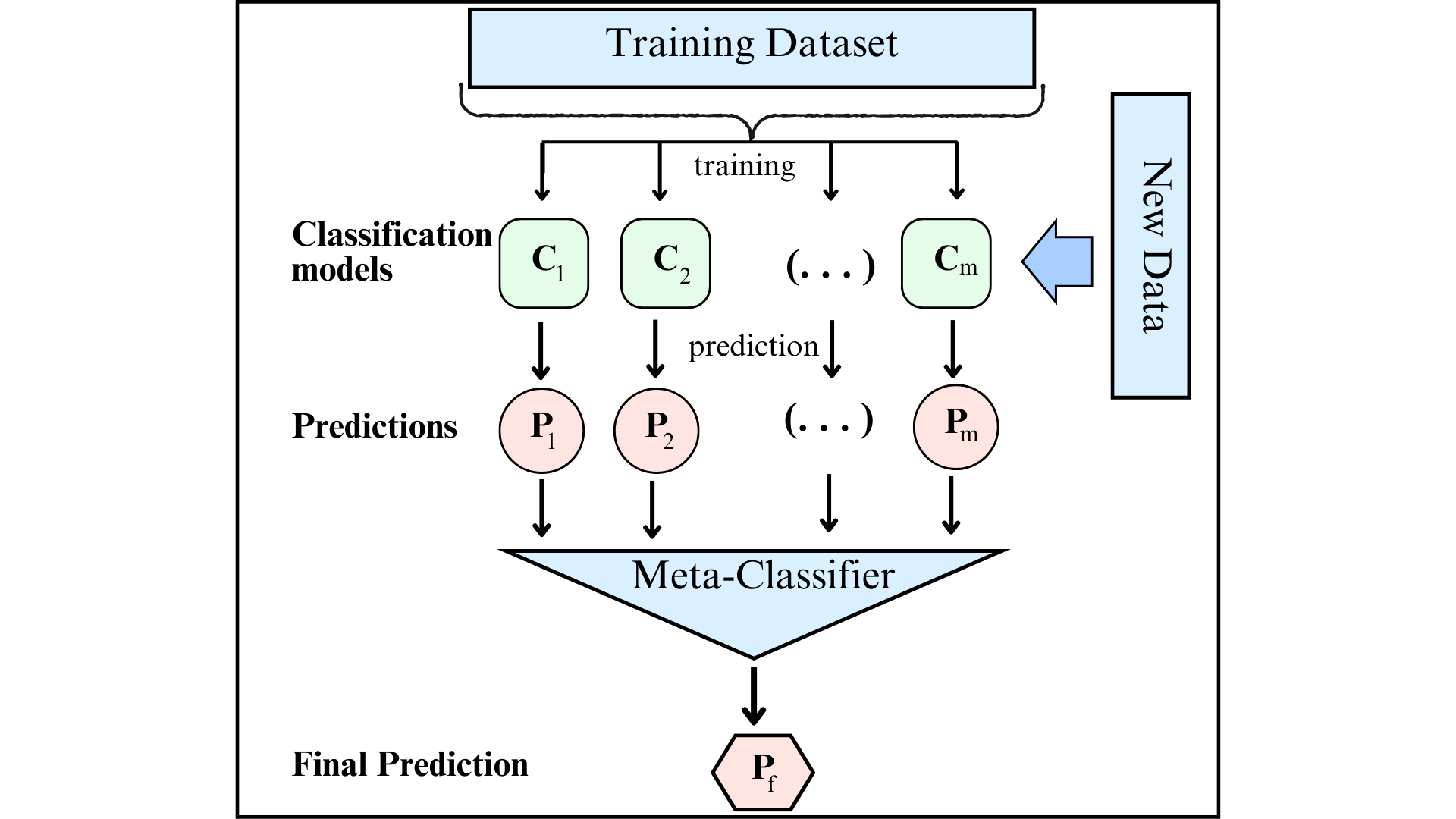
Based on previous research, we can build our model on the foundation of established knowledge and continue to develop new research ideas and methods. Understanding the challenges described in the dataset, the objective of this study is to propose an effective deep learning model for classifying valid and fraudulent insurance contracts. It is anticipated that this will be a valuable classification application to assist insurance companies in preventing instances where fraudsters exploit the system for personal financial gains. After reviewing the research directions, we have decided to build a deep learning model combined with the strategies mentioned above to improve the accuracy of predicting fraudulent claims in insurance compensation requests.

In this study, we conducted an analysis and evaluation of the dataset regarding automobile insurance compensation claims. This dataset comprises a variety of variables and observational samples, focusing on classifying valid and fraudulent insurance contracts. From the data analysis, we identified several challenges such as imbalance between the data labels, lack of correlation among variables, and variability in the distribution of compensation claims. Additionally, we found that dividing the dataset into feature groups could be beneficial for classifying insurance contracts. Based on these findings, we recognize the importance of developing effective models for classifying insurance contracts to assist insurance companies in risk assessment and fraud detection. We expect that utilizing deep learning models along with appropriate data processing strategies will enhance prediction capabilities and mitigate risks associated with fraud in automobile insurance.

## 3.2. Our proposed method

Traditional models, despite always achieving high performance with small datasets, encounter numerous issues with the constant need to select important features and perform poorly on datasets with too many observations. Therefore, a significant challenge faced by models such as Random Forest ,SVM or Decision Tree is the requirement to undergo a complex data processing step, slowing down progress and desired performance when dealing with datasets of varying sizes. Conversely, for deep learning neural network models, processing large-sized data poses no difficulty and does not require complex data transformation or feature selection, but they are prone to instability when working with small datasets.

Therefore, the model we propose uses the Stacking ensemble technique based on early-fusion technique, which can be referred to as the Stacking model. The Stacking model consists of two important parts: the ensemble of base models and the main model that undertakes meta-learning. With the aim of creating a model that can perform well without undergoing any complex data transformation processes, we decide to select two traditional models as the base models and construct a simple deep learning model evaluation as the meta-learning model to leverage the optimization that deep learning models operate on complex datasets. Thanks to this combination, it limits the underperformance when operating on datasets with few observations of this neural network model.



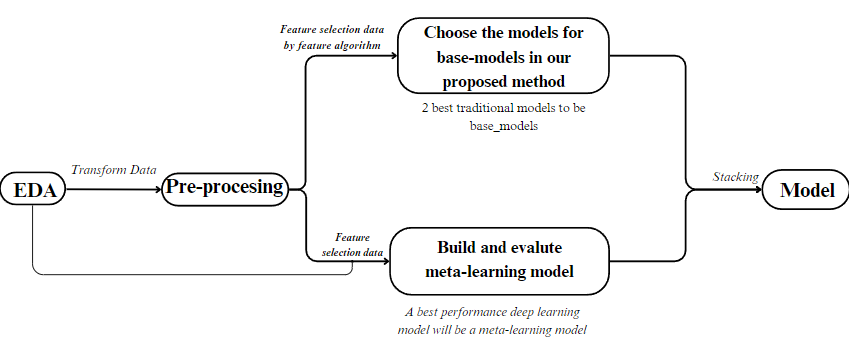
*Figure 17: Stacking model process. Source: (Raschka, S. 2014)*

Based on the findings after the data exploration process and to ensure the model we propose achieves the best performance, experimental processes will be conducted exactly as the [Figure 18](#bookmark=id.apfppjfp5b38). Coming to the data preprocessing phase, we will identify data normalization methods for both continuous and categorical variables, and strategies for handling imbalanced data, which are the imbalanced weight method and the SMOTE method. These strategies are applied to the dataset to help optimize the model right from the first steps.

When building the base models, for traditional models such as Random Forest, Support Vector Machine, Decision Tree, etc., feature selection must be conducted before the training process. Some techniques like RFE, Select from model, etc., will be applied to find the most suitable method. To ensure the model performs at its best, we use the Random Search algorithm for hyperparameter tuning, where parameters will be randomly tuned within a predefined range to find the optimal parameter set for the models. Finally, we will conduct experiments to select the two best models to serve as the base models.

When developing the deep learning model to serve as the meta-learner, the team will also apply the aforementioned data processing techniques to clearly record the model's performance on this dataset. We will also apply parameter tuning with the Random Search CV function (operating with the same mechanism as the parameter tuning technique applied to traditional models but specifically designed for deep learning models), aiming to evaluate and create the highest-performing model.

In the end, we will proceed to build the Stacking method and conduct a trial run on the evaluated dataset to clearly see the performance of the problem.



*Figure 18: Process of experiment setup. Source: Made by the author's group*

## 3.3. Experiments setup

Training Process: We separated the dataset into two subsets comprising testing data and training data with the ratio value of 66,7/33,3. For supervised learning models, we incorporate K-fold cross-validation to divide the dataset into k Training/Validation sets for training and evaluating the model. In this case, we divide 5 folds. The training data is augmented using SMOTE from the imbalanced-learn library to address class 2 imbalance issues. It is then normalized using the StandardScaler method and encoded using LabelEncoder. Machine learning models are evaluated based on scores for Accuracy, Precision, Recall, and F1 across the folds.

First, we selected and evaluated all models. Subsequently, we narrowed down our choice to the two best-performing traditional models, namely Random Forest and Decision Tree. We trained these models using Adam optimization with a learning rate of 0.0001 while we implemented the Early Stopping algorithm and a Learning Rate Schedule to enhance the efficiency and performance of the models. Following that, we constructed a deep learning model using the Keras library, designing the model architecture with layers, “relu” activation functions, Adam optimizer, and a learning rate of 0.0001. Finally, we utilized the base models such as Random Forest and Support Vector Machines. The Meta-Model is a deep learning model implemented using the Keras library to learn how to combine the outputs of the base models to generate the final predictions. In summary, the stacking model is a combination of base models and a meta-learner model to leverage the strengths of both types of models, enhancing the synthesis of information and predictions.

Our project consists of three main stages: evaluating and selecting traditional models as base models, constructing and optimizing the performance of the deep learning model using the Keras library, and finally building a Stacking model to serve the project. Regarding the group of traditional models, after constructing individual models and parameter tuning using the randomSearch function, the group proceeded to feature selection to achieve the best performance for the project. The group chose Adam as the optimizer, and 'binary-crossentropy' as the loss function. Finally, after specific evaluation steps, the group identified two models that met the requirements: Decision Tree and Random Forest.

Moving on to the next stage, the team proceeded to build a simple deep learning model using the Keras library. Initially, the results were unfavorable as the model only provided biased predictions toward class 0. However, after implementing callbacks such as early stopping to prevent overfitting and learning schedules to adjust the learning rate for optimal model performance, significant improvement was observed as the issues were addressed. To achieve the best results for the project, we decided to fine-tune the number of kernels in each layer of the model using RandomSearch, similar to the traditional models, to obtain a complete our proposed method after 600 epochs with a learning rate of 0.0001.

Using the Stacking method with a meta-learning model, the team proceeded to build the main model for their project. The base models were Random Forest and Decision Tree, while the primary meta-model was a simple deep learning model constructed through the aforementioned steps. Firstly, the team conducted training to obtain the output results of the base models. Subsequently, they concatenated the two model outputs into a 4-dimensional array for input into the team's Stacking model. Finally, employing the previously mentioned Callbacks techniques and training the Stacking model, the team achieved results after 120 epochs with a learning rate of 0.0001.

# 

# Results

## 4.1. Experiments on dataset

As mentioned in the Proposed Method section, two techniques were introduced to address the imbalance of the dataset, namely the class weight balancing method and the SMOTE technique. Through experimental results as shown in Table 3, by quickly running on the test set, we conclude that SMOTE is the more suitable method to address this condition.

|  | None | Class weight | SMOTE |
| --- | --- | --- | --- |
| Decision Tree | 71.5% | 76.07% | 82% |
| Random Forest | 76.47% | 75.57% | 80.8% |
| Linear Discriminant Analysis | 73.3% | 75.57% | 81% |
| SVM | 70.25% | 72.27% | 75.07% |
| Simple deep learning model | 73.3%% | 73.3%% | 66.67% |

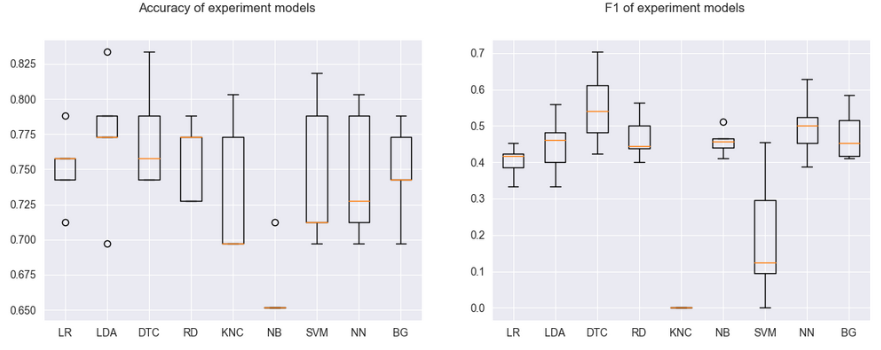
*Table 3: A result for imbalanced methods in dataset.*

It can be seen that the SMOTE technique is more suitable based on the results of the aforementioned models. Notably, in the simple deep learning model we are currently developing, where the majority class, class 0, constitutes up to 73.3%, our deep learning model is unable to distinguish, as it predicts 100% of the output as the majority class, resulting in such high results. Only after data augmentation was this condition addressed, although the performance remains very low.

## 4.2. Experiments on Traditional Model

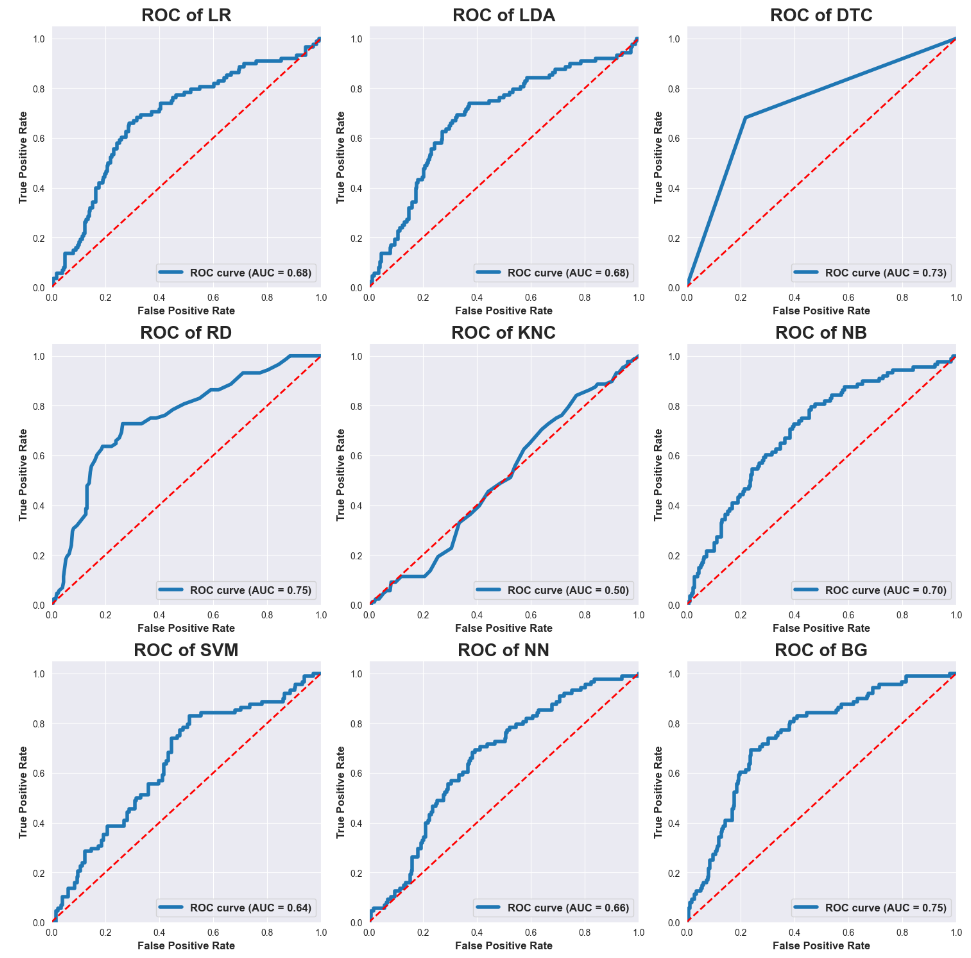
We conducted cross-validation using K-fold and then proceeded to analyze the model performance through the mentioned metrics: accuracy, F1 score, and the AUC curve.

The results on accuracy and f1 score in testing are visualized using Boxplot charts, which help us have a visual understanding of the metrics as well as the variability of each model. The baseline models were selected based on performance criteria, with a high degree of stability. Consequently, models such as Linear Discriminant Analysis (LDA), Decision Tree (DTC), and Random Forest (RF) are appropriate choices for the group of baseline models.

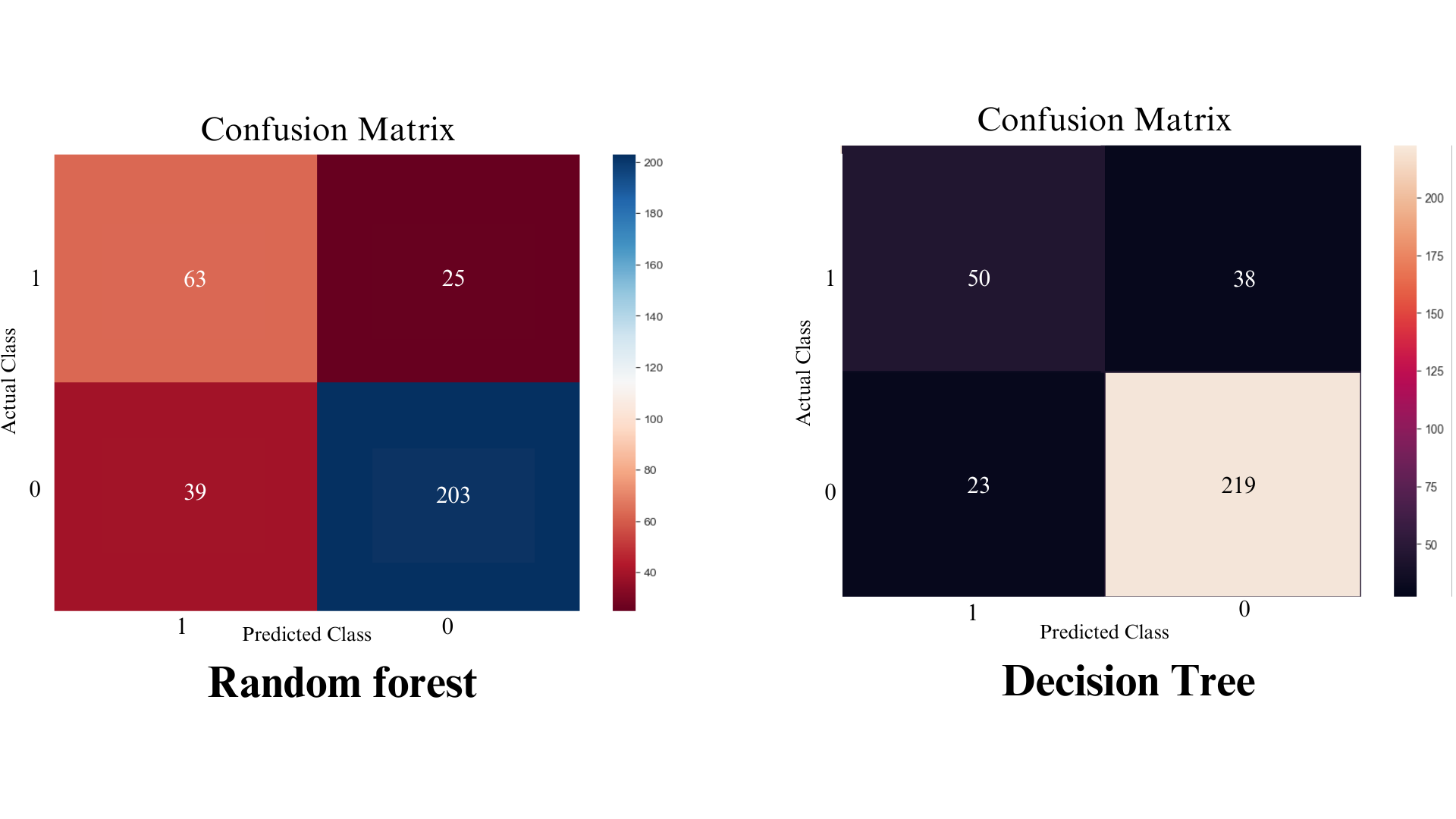


*Figure 19: Accuracy and F1- score of traditional models. Source: Made by the author's group.*

When it comes to analyzing ROC curves to select the top two models for fraud detection, we realize that Decision Tree and Random Forest are the most suitable models for the task. Therefore, the base model ensemble will be built from these two models.

*Figure 20: The AUC curve of the traditional model. Source: Made by the author's group*

After employing various strategies to select suitable traditional models for the project, the results obtained were the Decision Tree model and the Random Forest model. These models not only achieved a high accuracy level but also attained reasonable recall and precision values to ensure the performance in detecting fraudulent contracts. Following the evaluation and parameter tuning techniques, the Decision Tree model achieved an accuracy of 81.27%, with a fraud detection rate of approximately 70.3% of the actual cases. The Random Forest model reached an accuracy level of 80.8%, with a fraud detection rate of 75.5%.



*Figure 21: Confusion Matrix for Random Forest and Decision Tree. Source: Made by the author's group.*

To explain why these experimental results indicate that these are the two optimal models, Decision Tree and Random Forest are both optimal models for datasets with numerous features and weak correlations among attributes. Decision Tree excels in adapting to overfitting in high-dimensional feature spaces, while Random Forest can rank the importance of features and reduce overfitting. Additionally, due to the dataset's characteristic of being skewed towards the majority class, the models were evaluated not only based on accuracy but also on recall, precision, and F1 score. Some other models like SVM and LDA achieved good accuracy performance, but their other metrics were not highly evaluated, highlighting why Decision Tree and Random Forest are suitable models.

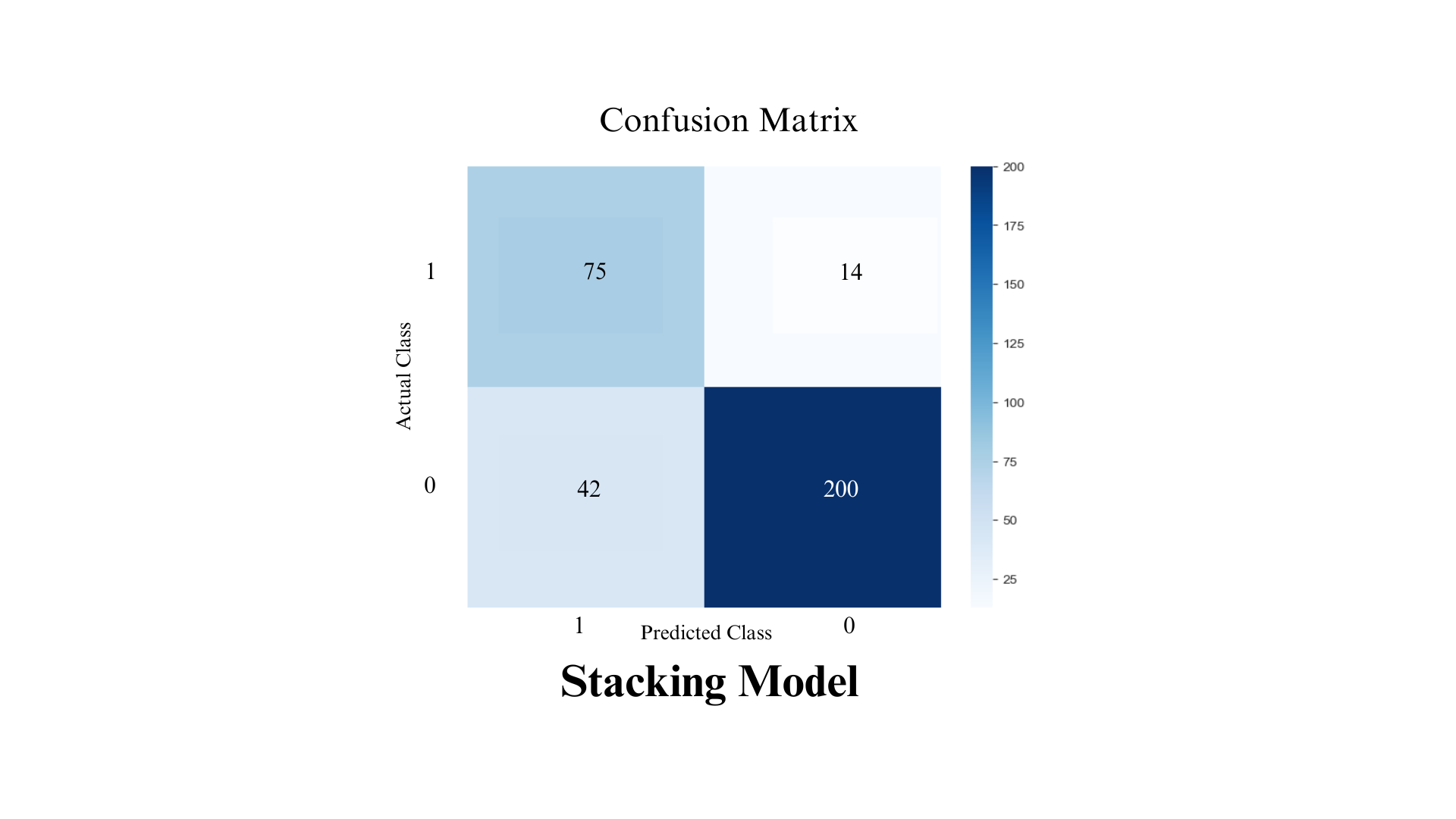
## 4.3. Experiments on Deep Learning Model

Initially, the simple deep learning model developed by the team exhibited a skewed result towards a value of 0 due to the imbalanced nature of the dataset, as mentioned earlier. By applying weight balancing strategies, Synthetic Minority Over-sampling Technique (SMOTE) for minority class augmentation, and utilizing relevant callbacks, this model improved to 66.67%.

The explanation for this improvement lies in the team's efforts not only to address the issue through appropriate data strategies and techniques but also by increasing the number of training epochs and implementing Early Stopping to prevent overfitting. Through the analysis of the dataset's specific characteristics and observing patterns during training, parameter tuning was performed to achieve the highest performance for our model we experimented with.

## 4.4. Experiment on Stacking model

The team decided to enhance the performance of the model in the previous part to its maximum by applying ensemble techniques derived from meta-learning, combining the outputs of the Random Forest and Decision Tree models, with the neural network model we built at 3.2 as the main learning model. The resulting ensemble model achieved a performance of 83.3%, with the ability to accurately predict 85.27% of the total fraudulent contracts in real-world scenarios. This result is on par with traditional models for a small and complex dataset.



*Figure 22: Confusion Matrix for Stacking model. Source: Made by the author's group.*

During the implementation, besides using it as a baseline model for the topic, we initially experimented with training traditional models as reference outcomes with the proposed methods in the article. These models include SVM, Decision Tree, Random Forest and LDA. According to the results recorded in Table 1, it shows that although deep learning models are not the optimal choice for the dataset of the topic, the results achieved by the Stacking model we proposed are significant, with the detection rate of fraudulent contracts in reality reaching up to 85%. This indicates that our method has provided quite good performance.

|  | Accuracy | Recall | F1 score |  |
| --- | --- | --- | --- | --- |
| SVM | 76.47% | 75.07% | 80% |  |
| Random Forest | 80.8% | 75.5% | 81% |  |
| Linear Discriminant Analysis | 74.4% | 69.54% | 70.1% |  |
| Decision Tree | 81.27% | 70.3% | 74% |  |
| Simple neural network model | 66.67% | 60.25% | 62% |  |
| Stacking model (base model: SVM,RF + meta learner: Simple neural network model) | 83.3% | 85.23% | 84% |  |

*Table 4: A result for experiment models in dataset.*

# Discussion

The deep learning model may not be suitable for small datasets, prone to volatility, and often lacks high stability. In the research paper “Survey on deep learning with class imbalance” (Johnson & Khoshgoftaar, 2019), the authors addressed the shortcomings in the field of deep learning concerning imbalanced data. Despite the demonstrated efficacy of deep learning models across various domains, it is essential to acknowledge their limitations when dealing with small and volatile datasets. The non-linearity and complexity of the model pose new challenges, raising concerns about its stability and generalizability. To address these challenges, we have mitigated the difficulties by integrating deep learning models with advanced methods. This includes employing balanced data sampling techniques, data augmentation, and applying regularization strategies to control overfitting. The amalgamation of deep learning models with traditional ones can result in a robust system, harnessing the strengths of deep learning while minimizing its drawbacks, particularly in scenarios involving imbalanced data. In summary, despite the drawbacks of deep learning models when confronted with small and volatile datasets, a flexible and innovative combination of methods allows us to leverage their advantages in numerous real-world applications. A profound understanding of the data and model characteristics is a crucial key to optimizing performance and ensuring stability throughout the training process.

The traditional model and traditional methods may be the optimal choice for such a dataset.The paper presents a novel strategy to enhance the effectiveness of SVM models in handling imbalanced datasets. This involves maintaining a good margin between the trained decision boundary of SVM and each class, aiming to improve classification performance on imbalanced data. Experimental results on five well-known imbalanced datasets demonstrate that this strategy not only enhances classification performance on the minority class but also exhibits lower sensitivity to parameter selection compared to other strategies such as SMOTE. In summary, the paper introduces a traditional SVM model as an excellent choice to address the imbalanced data issue, especially when applying the proposed strategy outlined in their research. Therefore, there is a basis to assert that traditional models such as SVM, Decision Tree are considered an excellent choice in our dataset.

The deep learning model is typically suitable for large datasets, high complexity, and often does not require feature selection.The stability and generalization ability of a deep learning model are often enhanced when trained on large datasets. This allows the model to comprehend the significant variations in the data and reduces the risk of overfitting. In the research paper titled “Automatic analysis of insurance reports through deep neural networks to identify severe claims” (Sabban et al., 2021), a novel method is proposed for automatically classifying insurance claims based on the predicted severity level.

The outstanding advantage of this model compared to previous models lies in its. For the simple neural network model we built , it is not only suitable for small datasets but also optimizes performance when applied to large and complex datasets. Regarding the advantages of deep learning models over traditional ones, deep learning often excels in handling large and complex data, eliminating the need for explicit feature selection as in traditional models. The ability to autonomously learn features from data helps minimize effort and time required for feature processing, reduces preprocessing steps, and enhances automation in the data analysis process. This makes deep learning models an efficient choice for tasks demanding flexibility and the ability to process large datasets.

# Conclusion & Future Works

In this study, we aim to propose a breakthrough model compared to traditional models in predicting insurance fraud conditions. To better understand the challenges in assessing the fraudulent status of companies, the team selected a dataset from a Kaggle competition, which closely resembles real-world information. However, through the exploration process, we also identified deep-seated issues within the dataset, such as imbalances, variables lacking correlation, and an excessive number of attributes significantly influencing prediction outcomes. Following data exploration and processing, the team's project comprises three main stages: Selection and evaluation of traditional models, Building and enhancing the Keras deep learning model, and constructing a Stacking model to accomplish the project. Utilizing a highly complex dataset, we observed that the project performed well at various outstanding points, achieving a performance of 83.34%.

Insurance companies, as the primary target audience of this study, can minimize losses due to insurance fraud by utilizing necessary information such as accident details, customer information, and insurance registration contract details. In the future, they can apply the results of this study to predict a large volume of compensation claims, optimizing performance and minimizing fraudulent occurrences, overcoming the limitations often encountered by traditional assessment methods.

This outcome not only addresses the instability issues commonly encountered by deep learning models but also improves performance, allowing the model to operate effectively with complex and extremely large datasets. It proves to be more efficient and optimized compared to traditional financial learning methods.

Future research directions will focus on optimizing the model and improving the problem's performance. Additionally, a more complex Keras network model will be proposed to adapt well to different datasets influencing various aspects of the fraud condition. We will also conduct experiments to further study famous deep learning models for the method proposed by the topic. In a broader sense, our objective is to explore the use of emerging technologies like smart contracts to enhance transparency and reduce fraud. This aligns with SDG 16 by strengthening the resilience of financial institutions against corruption. Advancements in AI can also improve fraud detection accuracy through machine learning and deep learning, supporting SDG 9 by fostering innovation and resilient infrastructure. Additionally, studying the environmental impact of fraud and how AI-powered detection can contribute to sustainability links to SDG 13. Developing a framework for evaluating fraud detection technologies can optimize their implementation and contribute to the global partnership for sustainable development outlined in SDG 17. Overall, insurance fraud detection technology has the potential to create a sustainable and equitable future in line with the United Nations' SDGs.

# Reference

Ali, A., Shamsuddin, S. M., & Ralescu, A. (2015). Classification with class imbalance problem: A review. Int. J. Advance Soft Compu. Appl, 7(3), 176–204.

Aslam, F., Hunjra, A. I., Ftiti, Z., Louhichi, W., & Shams, T. (2022). Insurance fraud detection: Evidence from artificial intelligence and machine learning. Research in International Business and Finance, 62, 101744.

Aziz, S., & Dowling, M. (2018). AI and Machine Learning for Risk Management. Social Science Research Network.

Automobile Insurance Fraud Detection: An Overview. (2022, April 27). IEEE Conference Publication | IEEE Xplore.

Bart, V. L. (2017). Machine learning: A revolution in risk management and compliance?

Benedek, B., Ciumaș, C., & Nagy, B. Z. (2023). On the Cost-efficiency of Automobile Insurance Fraud Detection Methods: A Meta-analysis. Global Business Review.

Benedek, B., & Nagy, B. Z. (2023). Traditional versus AI-Based Fraud Detection: Cost Efficiency in the Field of Automobile Insurance. Financial and Economic Review, 22(2), 77–98.

Belias, D., Velissariou, E., Chondrogiannis, M., Kyriakou, D., Varsanis, K., Vasiliadis, L., & Κουστέλιος, Α. (2018). Exploring insurance fraud and tourists’ misbehaviour. In Tourism, hospitality & event management (pp. 147–156).

Bhowmik, R. (2011). Detecting Auto Insurance Fraud by Data Mining Techniques. Journal of Emerging Trends in Computing and Information Sciences, 2(4), April.

Brownlee, J. (2020, August 20). A gentle introduction to the rectified Linear Unit (RELU). MachineLearningMastery.com.

Caruana, M. A., & Grech, L. (2021). Automobile insurance fraud detection. Communications in Statistics: Case Studies, Data Analysis and Applications, 7(4), 520–535.

Chen, Q., Xie, Q., Yuan, Q., Huang, H., & Li, Y. (2019). Research on a Real-Time Monitoring method for the wear state of a tool based on a convolutional bidirectional LSTM model. *Symmetry*, *11*(10), 1233.

Derrig, R. A. (2002b). Insurance fraud. *Journal of Risk and Insurance*, *69*(3), 271–287.

De Sá, J. a. S., Da Costa Almeida, A., Da Rocha, B. R. P., Da Mota, M. a. S., De Souza, J. R. S., & Dentel, L. (2016). Lightning Forecast Using Data Mining Techniques On Hourly Evolution Of The Convective Available Potential Energy. Brazilian Congress on Computational Intelligence, Fortaleza, November, 8-11.

De Zarzà, I., De Curtò, J., & Calafate, C. T. (2023). Optimizing neural networks for imbalanced data. Electronics, 12(12), 2674.

Faysal, J. A., Mostafa, S. T., Tamanna, J. S., Mumenin, K. M., Arifin, M. M., Awal, M. A., Shome, A., & Mostafa, S. S. (2022). XGB-RF: a hybrid Machine learning approach for IoT intrusion Detection. Telecom, 3(1), 52–69.

Feinman, J. M. (n.d.). The Regulation of Insurance Claim Practices. UCI Law Scholarly Commons.

Gomes, C., Jin, Z., & Yang, H. (2021). Insurance fraud detection with unsupervised deep learning. Journal of Risk and Insurance, 88(3), 591–624.

[Grid Search vs Random Search. (n.d.)](https://datarian.io/blog/grid-search-random-search).

Jing, Z. (2022). Fabricating insurance subject matter and defrauding insurance money: a civil wrong or a criminal offence? Asia Pacific Law Review, 30(1), 145–166.

Johnson, J., & Khoshgoftaar, T. M. (2019). Survey on deep learning with class imbalance. Journal of Big Data, 6(1).

Kratzert, F. (2016, February 12). Understanding the backward pass through Batch Normalization Layer.

Krishnamurthy, B. (2022, October 28). *An introduction to the RELU activation function*. Built In.

Kumar, M., & Gopal, M. (2010). A hybrid SVM based decision tree. Pattern Recognition, 43(12), 3977–3987.

Li, Y., Chun, Y., Liu, W., & Li, M. (2018). A principle component analysis-based random forest with the potential nearest neighbor method for automobile insurance fraud identification. Applied Soft Computing, 70, 1000–1009.

Malhotra, R., & Lata, K. (2021). Handling class imbalance problem in software maintainability prediction: an empirical investigation. Frontiers of Computer Science, 16(4).

Mahlow, N., & Wagner, J. (2016). Process landscape and efficiency in non-life insurance claims management. *The Journal of Risk Finance*, *17*(2), 218–244.

Mitchellodili. (2022, January 21). Automobile insurance claims. Kaggle. <https://www.kaggle.com/code/mitchellodili/automobile-insurance-claims/input>

Nagrecha, S., Johnson, R. A., & Chawla, N. V. (2018). FraudBuster: Reducing Fraud in an Auto Insurance Market. Big Data, 6(1), 3–12.

Nian, K., Zhang, H., Tayal, A., Coleman, T., & Li, Y. (2016). Auto insurance fraud detection using unsupervised spectral ranking for anomaly. The Journal of Finance and Data Science, 2(1), 58-75.

Poufinas, T., Gogas, P., Papadimitriou, T., & Zaganidis, E. (2023). Machine learning in forecasting motor insurance claims. Risks, 11(9), 164.

Padhi, S. N., & Panigrahi, S. (2020). Use of data mining techniques for data balancing and fraud detection in automobile insurance claims. In Advances in intelligent systems and computing (pp. 221–230).

Predicting fraudulent claims in automobile insurance. (2018b, April 1). IEEE Conference Publication | IEEE Xplore.

Kabir, M. a. M. R. a. A. (2022). Fraud Detection In Automotive Insurance Domain Using Machine Learning.

Rustam, Z., Yaurita, F., & Segovia-Vergas, M. (2018). Application of support vector machines in evaluating the internationalization success of companies. Journal of Physics: Conference Series, 1108, 012038.

Sabban, I. C., Lopez, O., & Mercuzot, Y. (2021). Automatic analysis of insurance reports through deep neural networks to identify severe claims. Annals of Actuarial Science, 16(1), 42–67.

Salmi, M., & Atif, D. A (2022). A Data Mining Approach for Imbalanced Automobile Insurance Fraud Data with Evaluation of Two Sampling Techniques and Two Filters. Journal of Information Assurance and Security, 17 (2022), 122-135.

Seng, Z., Kareem, S. A., & Varathan, K. D. (2021). A Neighborhood Undersampling Stacked Ensemble (NUS-SE) in imbalanced classification. Expert Systems With Applications, 168, 114246.

Shen, H., Li, Y., Tian, X., Xiao-Fan, C., Li, C., Bian, Q., Wang, Z., & Wang, W. (2022). Mass data processing and multidimensional database management based on deep learning. Open Computer Science, 12(1), 300–313.

Shlezinger, N., Whang, J., Eldar, Y. C., & Dimakis, A. G. (2023). Model-Based Deep learning. Proceedings of the IEEE, 111(5), 465–499.

Sinha, S. (2023, January 20). *How fintech firms are leveraging ML in financial fraud detection*. Blog - Product Insights by HQ.

Team, C. (2023, November 22). Random Forest. Corporate Finance Institute.

Tongesai, M., Godfrey, M., & Zvarevashe, K. (2022). Insurance Fraud Detection using Machine Learning. 2022 1st Zimbabwe Conference of Information and Communication Technologies (ZCICT).