Capstone project

Health Insurance Fraud Claim Detection

submitted in partial fulfilment of the requirements for the

Post Graduate Program in Data Science at

Praxis Business School

Submitted by

Team Health Hawks

Harshitha Immaneni	D22019
Laxmi Panchal	D22024
Uma Madduri	D22025
Manoj Balaji Potdar	D22030
Kenwin Dass C	D22051

Under the guidance of

Prof Gourab Nath

Post Graduation Program in Data Science
Praxis Business School
HSR Layout, Bengaluru,
Karnataka 560102
(August 2022 – April 2023)

Table of Contents

Abstract

The health care industry is one of the important service providers that improves people lives. As the cost of the healthcare service increases, health insurance becomes the only way to get quality service in case of an accident or a major illness. As health insurance will reduces the costs and provides financial and economic stability for an individual.

One of the main tasks of healthcare insurance providers is to monitor and manage the data and to provide support to customers. Due to regulations and business secrecy, insurance companies do not share the patient's data but since the data are not integrated and not in sync between insurance providers, there has been an increase in the number of fraud's occurring in healthcare. Often times ambiguous or false information is provided to health insurance companies in order to make them pay for some false claims to the policy holders. The individual policyholder may also claim benefits from multiple insurance providers.

There is a financial loss of billions of dollars each year as estimated by the National Health Care Anti-Fraud Association (NHCAA). In order to prevent health insurance fraud, it is necessary to build a system to securely manage and monitor insurance activities by integrating data from all the insurance companies.

In this research, we propose a model that uses the Boosting algorithm to detect fraudulent health insurance claims. By securely managing and monitoring insurance activities through integrated data from all insurance companies, our model can help prevent healthcare fraud and ensure the provision of quality healthcare services to individuals.

Acknowledgment

We would want to convey our heartfelt gratitude and sincere thanks to **Prof. Gourab Nath**, our mentor and our teacher, for his invaluable advice and assistance in completing our project even outside the working hours. Throughout the project, he was with us every step of the way, offering innovative ideas and motivation, which allowed us to accomplish our task effectively and within the set timeframe. The success of this project was the result of the collective efforts of many people, and we feel fortunate to have received such exceptional guidance and assistance throughout our time at Praxis Business School.

Our mentor's constant encouragement and advice have played an instrumental role in helping us navigate through complex challenges and achieve our goals. His unwavering support and commitment towards our project have been a source of inspiration for us.

Lastly, We would also like to express our heartfelt gratitude to Praxis Business School for providing us with the opportunity to undertake this project. The institution's academic rigor and focus on experiential learning have enabled us to develop our skills and competencies, preparing us for a successful career in the corporate world.

Introduction:

Health insurance is a form of insurance that pays for medical expenses. If you are covered under health insurance, you pay some amount of premium every year to an insurance company and if you have an accident or if you have to undergo an operation or a surgery, the insurance company will pay for the medical expenses. Frauds can be committed by anybody. It can be committed by a policyholder, a health insurance company or even its employees. Frauds committed by a policyholder could consist of members that are not eligible, concealment of age, concealment of pre-existing diseases, failure to report any vital information, providing false information regarding self or any other family member, failure in disclosing previously settled or rejected claims, frauds in physician's prescriptions, false documents, false bills, exaggerated claims.

Types of Fraud related to claims

1. **Not disclosing pre-existing condition:**

   Most individual health policies have a two to three year waiting period for pre-existing conditions/ pre-existing diseases. A policyholder can hide that fact by manipulating the results of the pre-policy health check-up.

2. **False bills:**

   The aim of health insurance is to cover the medical expenses incurred during illness or surgery. It is not purchased with a view to make profits. Hence, when a person submits forged bills, when no expenses have been incurred, or inflated bills, the action qualifies as fraudulent.

3. **Buying multiple policies:**

   If a person has more than one individual health policies or a group policy and an individual policy, he/she must have to disclose it to all the insurers. This is to prevent an individual from making multiple claims and ending up making a profit.

4. **Fake accident:**

   A customer might fake an accident in order to claim compensation for medical bills and hospital bills.

Dishonest and fraudulent claims pose a significant threat to the financial and moral well-being of not just the health insurance sector, but also the entire national economy. The process of verifying such claims involves gathering and analyzing concrete evidence such as documentation, statements from the policyholder's family members, neighbors, and other relevant sources.

Problem statement:

Health insurance fraud is a significant issue, causing financial losses and inappropriate treatments. Fraudulent claims lead to increased premiums, delays in processing legitimate claims, and inconvenience for policyholders. Detecting fraud is challenging, as fraudsters use sophisticated techniques to evade detection. An accurate fraud detection model is crucial to reduce losses and ensure appropriate patient care. Insurance companies need accurate claims data free of fraud to provide quality service. The dataset used for the model consists of 10,48,576 observations and 24 variables. Accurate fraud detection can prevent insurance companies from being taken advantage of by fraudulent claims, improving service quality and reducing financial losses.

Objective:

The objective is to build a model that can accurately identify fraudulent health insurance claims. This will be achieved by utilizing predictive modeling techniques, which involve analyzing patterns and anomalies in the claims data to uncover indicators of fraudulent activity. The goal of this approach is to enable insurance companies to detect fraudulent claims more quickly and efficiently, ultimately reducing the financial impact of these claims on the healthcare system. This type of model will help to prevent losses, identify fraudulent behavior more effectively, and minimize the negative effects of fraud on the insurance industry as a whole.

Impacts of detecting health insurance fraudulent claims:

According to the National Health Care Anti-Fraud Association, health insurance fraud costs the United States around $68 billion yearly. In the US environment, detecting health insurance fraud can have substantial effects and opportunities.

- **Cost savings:** Both insurance companies and the government can save a sizable sum of money by identifying and preventing health insurance fraud. Consumers frequently pay the price of false claims through higher premiums, which can put a strain on the economy. Fraud detection can assist in preventing this expense from being passed forward to customers, which can result in cost savings.

- **Better patient outcomes:** Inappropriate therapies may result from false claims, which can harm patients' outcomes. Healthcare professionals can make sure that patients receive proper care that is covered by their insurance by looking for fraud. Better patient outcomes and a healthier population as a whole may result from this.

- **Increased regulatory compliance:** By spotting health insurance fraud, insurance providers and healthcare professionals can be held more accountable for abiding by rules. As a result, the healthcare sector may be regulated more successfully and fraud may be less likely to occur.

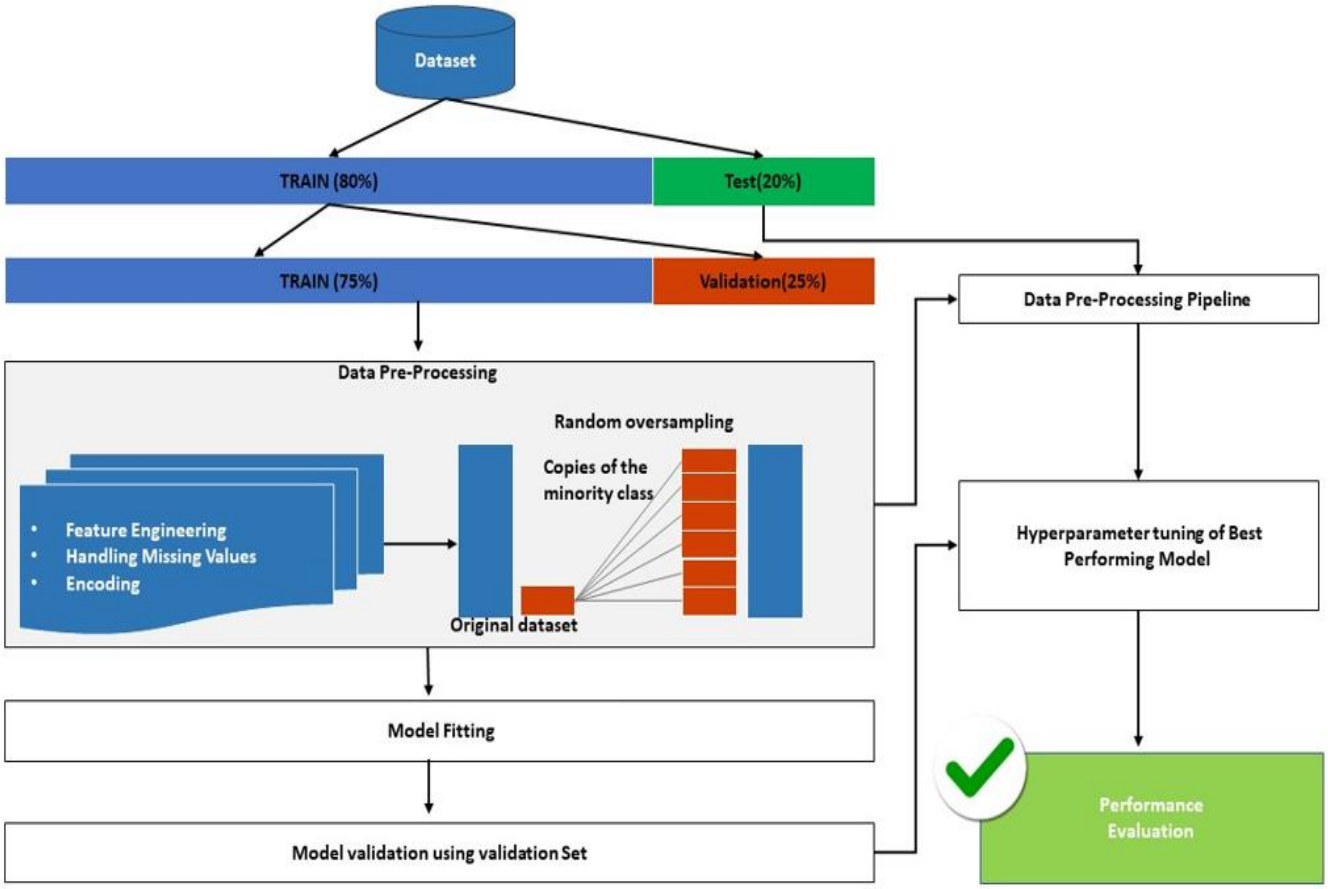
- **Enhanced reputation:** By spotting and preventing fraud, insurance providers and healthcare organisations can build a reputation for being dependable and trustworthy service providers. This may enhance their reputation and draw in more clients, which may result in more sales and more room for expansion.

Designing an analytics solution to predict fraudulent claims can help insurance companies detect and prevent fraud early on. This can minimize financial losses and improve regulatory compliance. By detecting and avoiding fraud, the healthcare sector can operate more effectively and efficiently, improving patient outcomes and benefiting the economy as a whole.

Methodology

Python Code Link: Link

Flow Diagram:



Tools Used:

- Python
- Tableau

Libraries Used:

- **numpy:**
  numpy (imported as np) is a library for numerical computing in Python.
- **Pandas:** pandas (imported as pd) is a library for data manipulation and analysis. It provides data structures for efficiently storing and manipulating large datasets.
- **matplotlib.pyplot:**
  matplotlib.pyplot (imported as plt) is a plotting library for creating visualizations in Python. It provides a range of tools for creating line plots, scatter plots, histograms, and other types of charts.
- **seaborn:**
  seaborn is a library for data visualization based on matplotlib. It provides a high-level interface for creating attractive and informative statistical graphics.
- **Warnings**
  warnings is a built-in Python library for warning control.
- **statsmodels**
  statsmodels is a library for statistical modeling in Python. It provides a range of tools for regression analysis, time series analysis, and other types of statistical modeling.
- **scipy**
  scipy is a library for scientific computing in Python. It provides a range of functions for mathematical optimization, signal processing, and other tasks.
- **sklearn.linear_model**
  sklearn.linear_model contains implementation of linear models in scikit-learn, including LogisticRegression.
- **sklearn.tree**
  sklearn.tree contains implementation of decision trees in scikit-learn, including DecisionTreeClassifier.
- **Lightgbm**
  lightgbm is a gradient boosting framework that uses tree-based learning algorithms.
- **sklearn.ensemble**
  sklearn.ensemble contains implementation of ensemble models in scikit-learn, including RandomForestClassifier and AdaBoostClassifier.

- **sklearn.model_selection**
  sklearn.model_selection contains implementation of model selection in scikit-learn, including GridSearchCV and train_test_split.
- **sklearn.metrics**
  sklearn.metrics contains implementation of evaluation metrics in scikit-learn, including f1_score, classification_report, roc_auc_score, accuracy_score, and recall_score.
- **sklearn.feature_selection**
  sklearn.feature_selection contains implementation of feature selection algorithms in scikit-learn, including SelectFromModel.
- **imblearn.over_sampling**
  imblearn.over_sampling is a library for data resampling to handle class imbalance in datasets, including RandomOverSampler and SMOTE
- **sklearn.ensemble**
  sklearn.ensemble contains implementation of ensemble models in scikit-learn, including GradientBoostingClassifier.
- **sklearn.model_selection**
  sklearn.model_selection contains implementation of cross-validation in scikit-learn, including KFold.
- **xgboost:**
  XGBoost is a popular open-source library used for gradient boosting. It is used for supervised learning problems, especially in classification and regression tasks. The library is optimized for speed and performance and provides a range of hyperparameters that can be tuned to improve the accuracy of the model.
- **AdaBoostClassifier:**
  AdaBoost is another boosting algorithm that is commonly used in machine learning. It is also used for supervised learning problems and can be used for classification or regression tasks.
- **RandomizedSearchCV:**
  RandomizedSearchCV is a method for hyperparameter tuning in machine learning. It works by searching through a range of hyperparameters for a given model and finding the combination that produces the best results.

Data preparation:

Overview of the Dataset: Features, Size, and Target Variable

We are using Insurance Dataset which contains 10,48,575 observations and 24 features.

Target Variable – Result

Predictors –

1. Area Service
2. Hospital County
3. Hospital Id
4. Age
5. Gender
6. Cultural Group
7. ethnicity
8. Days spend hospital
9. Admission type
10. Home/self-care
11. CCS diagnosis code (Clinical Classifications Software diagnosis code)
12. CCS procedure code (Clinical Classifications Software diagnosis code)
13. Apr drg description (All Patients Refined Diagnosis Related Groups)
14. Code illness
15. Mortality risk
16. Surg Description (Surgical Description)
17. Weight of baby
18. Abortion
19. Emergency dept yes/No (Emergency Department)
20. Tot charg (Total Charge)
21. Tot cost (Total Cost)
22. Ratio of total costs to total charges

23. Payment Typology

Description of each predictors:

1. Area Service

In the US health insurance system, area services refer to healthcare services that are available within a specific geographic area. Health insurance plans often have a network of healthcare providers, hospitals, and other facilities that are considered in-network, meaning that they are contracted with the insurance plan to provide services to plan members.

2. Hospital County

In the US health insurance system, hospital county refers to the geographic location of a hospital, typically based on the county in which it is located. The hospital county can have an impact on health insurance coverage, as insurance plans may have different networks of hospitals and providers depending on the geographic location of the plan member.

3. Hospital ID

In the US health insurance system, a hospital ID is a unique identifier assigned to a hospital or healthcare facility by the Centers for Medicare and Medicaid Services (CMS). This identifier is used to track, and process claims for services provided by the hospital or healthcare facility to patients covered by Medicare or Medicaid.

4. Age

Age is an important factor in US health insurance, as it is often used to determine premium rates and eligibility for certain plans. Under the Affordable Care Act (ACA), health insurance companies are prohibited from charging older individuals more than three times the premium charged to younger individuals for the same policy. This is known as the "age rating limit."

5. Gender

Gender is a factor that can impact health insurance coverage and premiums in the US. In the past, health insurance companies often charged higher premiums for women than for men, based on the assumption that women use more healthcare services. However, under the Affordable Care Act (ACA), insurance companies are prohibited from charging different premiums based on gender.

6. Cultural Group

Cultural groups can impact health insurance coverage and access to care in the US. Members of certain cultural groups may face disparities in health outcomes and healthcare access, which can be influenced by a range of factors including social, economic, and environmental factors. Under the Affordable Care Act (ACA),

insurance companies are prohibited from discriminating based on race, color, national origin, sex, age, or disability.

7. Ethnicity

Ethnicity can impact health insurance coverage and access to care in the US. Members of certain ethnic groups may face disparities in health outcomes and healthcare access, which can be influenced by a range of factors including social, economic, and environmental factors.

8. Days Spend Hospital

The number of days spent in the hospital is an important factor in US health insurance coverage and reimbursement. In most cases, insurance companies will cover a certain number of hospital days per year and may have different coverage limits for inpatient and outpatient services.

9. Admission Type

The admission type for a patient is an important factor in US health insurance coverage and reimbursement. There are several types of admission, including emergency, elective, urgent, and observation. In general, insurance companies will cover medically necessary hospital admissions, but coverage and reimbursement may vary depending on the specific plan and provider, as well as the type of admission and length of stay.

10. Home/Self-Care

Home or self-care refers to medical care or support services that are provided outside of a hospital or medical facility, such as in the patient's home or community setting.

Coverage for home or self-care services may be subject to certain eligibility requirements, such as a doctor's order or certification of medical necessity.

11. CCS diagnosis code (Clinical Classifications Software diagnosis code)

CCS (Clinical Classification Software) diagnosis codes are used in US health insurance to classify and group diagnoses and procedures for statistical analysis and reimbursement purposes. The CCS diagnosis codes are assigned based on the primary diagnosis for an encounter, and they are used to classify the diagnosis into one of the 285 clinically meaningful categories. The CCS codes are used by insurance companies to determine the appropriate reimbursement amount for a particular diagnosis or group of diagnoses.

12. CCS procedure code (Clinical Classifications Software diagnosis code)

CCS (Clinical Classification Software) procedure codes are used in US health insurance to classify and group procedures for statistical analysis and

reimbursement purposes. The CCS procedure codes are assigned based on the primary procedure for an encounter, and they are used to classify the procedure into one of the 231 clinically meaningful categories. The CCS codes are used by insurance companies to determine the appropriate reimbursement amount for a particular procedure or group of procedures.

13. Apr drg description (All Patients Refined Diagnosis Related Groups)

All Patients Refined Diagnosis Related Groups (APR-DRGs) is a system used to classify and group patients based on their diagnoses and procedures for reimbursement and analysis purposes. The APR-DRGs system is used by many US states and other countries to pay for hospital inpatient services based on the severity and complexity of the patient's condition.

14. Code illness

In the US health insurance system, illnesses are coded using the International Classification of Diseases (ICD) codes. ICD codes are alphanumeric codes that are used to describe diseases, injuries, and other medical conditions. These codes are used by healthcare providers to document and communicate the patient's diagnosis, and they are also used by insurance companies to determine appropriate reimbursement amounts for healthcare services.

15. Mortality Risk

The mortality risk is a term used in US health insurance to describe the likelihood of a patient dying within a specified period due to a particular disease or condition. The mortality risk is often used to determine appropriate reimbursement rates for healthcare providers who treat patients with serious or life-threatening illnesses. It is often calculated using APR-DRGs.

16. Surg Description (Surgical Description)

Insurance companies use a coding system called Current Procedural Terminology (CPT) to describe medical procedures and services. The CPT codes provide a standardized way to communicate information about the medical services provided and are used to determine reimbursement rates for healthcare providers. For surgical procedures, the CPT code provides a description of the procedure performed, including the body part involved and the specific technique used.

17. Weight of Baby

A baby’s weight at birth may be recorded as part of the medical record and may be used to help assess overall health and development. In some cases, a baby's weight may be an indicator of certain medical conditions or risk factors, such as premature birth or low birth weight. In these cases, medical interventions or follow-up care may be necessary to ensure the baby's health and well-being.

18. Abortion

Abortion is a controversial issue in the US, and access to abortion services and coverage under health insurance plans varies depending on several factors, including the specific state and insurance plan. Under federal law, insurance plans sold on the health insurance marketplace established by the Affordable Care Act (ACA) are required to cover certain preventive health services, including contraception, without cost-sharing for the patient.

19. Emergency dept yes/No (Emergency Department)

Emergency department (ED) visits are an important aspect of US health insurance coverage and reimbursement. Under federal law, insurance plans are required to cover emergency medical services regardless of whether the provider is in-network or out-of-network. This means that if a patient has an emergency medical situation and needs to go to the ED, the insurance plan must cover the costs of that care, even if the hospital or ED is not in the plan's network of providers.

20. Tot charg (Total Charge)

Total charge in US health insurance refers to the total amount billed by healthcare providers for medical services and procedures. This includes charges for hospitalization, physician and specialist services, diagnostic tests, procedures, medications, and other healthcare-related services. The total charge is typically the initial amount billed by healthcare providers, but it does not necessarily reflect the actual amount that the insurance plan will cover.

21. Tot cost (Total Cost)

Total cost in US health insurance refers to the overall amount that is spent on healthcare services for a specific individual or population, including costs paid by the insurance plan and out-of-pocket costs paid by the patient. The total cost of healthcare is influenced by various factors, including the cost of medical services and procedures, the frequency and duration of healthcare utilization, and the health status and needs of the individual or population.

22. Ratio of total costs to total charges

The ratio of the total charge to the total cost in US health insurance can vary widely depending on the specific healthcare services and procedures involved, as well as the type of insurance plan and cost-sharing provisions. In general, the total charge for a medical service or procedure is higher than the actual cost of providing the service or procedure. This is because healthcare providers often charge higher rates to insurance plans as a way to negotiate higher reimbursement rates for their services.

23. Payment Typology

Payment typology in US health insurance refers to the different methods by which healthcare providers are reimbursed for medical services and procedures. There are several types of payment typologies used in US health insurance, including 1. Fee-for-service, 2. Capitation, 3. Bundled payments, 4. Value-based payments, 5. Global payments.

Data pre-processing:

- **Data Cleaning:**
  We have done Removing abnormal values and Mode Imputation of categorical variables to fill missing values

- **Data transformation:**
  We have done One hot encoding of categorical variables to convert categorical variables into a numerical format that machine learning algorithms can understand

Experimentation

Experiment 1: Fitting a baseline model

Aim:

To fit a baseline logistic regression model and evaluate its performance using the F2 score.

Dependencies:

Python libraries such as NumPy, Pandas, Scikit-learn, and Matplotlib are used to preprocess data, fit the model, and evaluate its performance.

Methodology:

The dataset is first preprocessed by handling missing values and categorical variables. Then, the logistic regression model is trained using the preprocessed data. The F2 score is used to evaluate the model's performance.

Observation:

The baseline logistic regression model achieves an F2 score of 0.0015, indicating poor performance in predicting positive cases.

Conclusion:

The baseline logistic regression model demonstrated inadequate performance with an F2 score of 0.0015 due to underfitting and severe data imbalance. Improving the model's performance is possible by optimizing hyperparameters, balancing the data, and exploring other modeling techniques. Addressing these issues can enhance the model's F2 score and predictive performance on the dataset.

Experiment 2: IV Calculation and Model Fitting with Best Features

Aim:

To calculate IV (Information Value) and fit a logistic regression model with the top features based on IV.

Dependencies:

Python libraries such as pandas, numpy, scikit-learn, and scipy.

Methodology:

We first calculated WOE (Weight of Evidence) and regrouped similar WOE values. Then, we calculated the IV score to assess the predictive power of the evaluated variables. we got a low IV score indicates minimal predictive power and a maximum IV of 0.0185 suggests insufficient power for evaluated variables Next, we selected the top 15 IV-based features and fitted a logistic regression model to the data.

Observation:

The logistic regression model's performance yielded a subpar F2 score of 0.015, suggesting ineffective prediction of the dependent variables.

Conclusion:

Alternative feature selection methods are necessary to improve the results, given the model's low F2 score.

Experiment 3: Comparing Balancing Techniques for Model Fitting

Aim:

To compare the performance of different balancing techniques and identify the best one for model fitting.

Dependencies:

Python libraries like imblearn for implementing balancing techniques and scikit-learn for fitting the models.

Methodology:

We applied the SMOTE technique for data balancing and fitted the Decision tree, Random Forest, and Ada Boost models. We then tried Random Oversampling for Data Balancing, and fitted Decision tree, Random Forest, and Ada Boost models.

Observation:

We found that the maximum F2 score achieved with SMOTE was only around 0.10, indicating that it did not significantly improve the model performance. However, we observed a significant improvement in the model's performance with Random Oversampling. The highest attained F2 score was around 0.32.

Conclusion:

Based on our findings, we concluded that Random Oversampling is the best balancing technique for model fitting in this dataset.

Experiment 4: Enhancing F2 score for fraud detection through model selection

Model 1: Decision Tree

Aim:

The aim is to improve the F2 score for fraud detection by exploring different machine learning models. We started with the decision tree model and used random oversampling to address class imbalance.

Dependencies:

Python libraries such as scikit-learn and imbalanced-learn for implementing the Decision Tree model with Random Oversampling.

Methodology:

The initial model used was a Decision Tree model with Random Oversampling. The evaluation metric used was the F2 score, with a focus on recall for the Fraud class.

Observation:

The Decision Tree model with Random Oversampling was used but achieved a low F2 score of 0.16, indicating poor performance in correctly identifying the Fraud class.

Conclusion:

While the Decision Tree model did not perform well in terms of correctly identifying the Fraud class, we can explore other tree-based algorithms such as Random Forest to further improve the F2 score for fraud detection. It is essential to continue fine-tuning the models by optimizing hyperparameters, feature selection, and balancing techniques to achieve the desired performance.

Model 2: Random Forest

Aim:

The aim is to improve the F2 score for fraud detection by exploring different machine learning models.

Dependencies:

Python libraries such as scikit-learn and imbalanced-learn were used to implement the Random Forest model with Random Oversampling.

Methodology:

The Random Forest model with Random Oversampling was implemented, and the evaluation metric used was the F2 score with a focus on recall for the Fraud class.

Observation:

The Random Forest model with Random Oversampling achieved an F2 score of 0.22, which is an improvement compared to the Decision Tree model. However, it is still not satisfactory for the task of fraud detection.

Conclusion:

Further fine-tuning of the Random Forest model may be necessary to achieve the desired performance. This can involve techniques such as hyperparameter optimization, feature selection, and balancing techniques. Additionally, it may be worth exploring various boosting algorithms such as AdaBoost and XGBoost to see if they can further improve the F2 score for fraud detection.

Model 3: Ada Boost

Aim:

The aim is to improve the F2 score for fraud detection by exploring different machine learning models. In this case, we implemented the Ada Boost model with Random Oversampling.

Dependencies:

Python libraries such as scikit-learn and imbalanced-learn were used to implement the Ada Boost model with Random Oversampling.

Methodology:

The Ada Boost model with Random Oversampling was implemented, and the evaluation metric used was the F2 score, with a focus on recall for the Fraud class.

Observation:

The Ada Boost model with Random Oversampling achieved an F2 score of 0.32, which showed a significant improvement compared to the Decision Tree and Random Forest models.

Conclusion:

The results indicate that the Ada Boost model with Random Oversampling is a promising approach for fraud detection. However, further optimization and exploration with other boosting algorithms such as gradient boost and XGBoost, as well as feature engineering techniques, may still be necessary to achieve even better performance.

Model 4: Gradient boosting

Aim:

The aim is to improve the F2 score for fraud detection by exploring different machine learning models. In this case, we implemented the Gradient Boosting model with Random Oversampling.

Dependencies:

Python libraries such as scikit-learn and imbalanced-learn were used to implement the Gradient Boost model.

Methodology:

The Gradient Boost model was implemented, and the evaluation metric used was the F2 score with a focus on recall for the Fraud class.

Observation:

The Gradient Boost model achieved an F2 score of 0.38, which is a notable improvement compared to the Decision Tree, Random Forest, and Ada Boost models. The model's ability to iteratively fit weak learners to residual errors of previous trees results in more accurate predictions.

Conclusion:

The Gradient Boost model has demonstrated promising results for the task of fraud detection, and further fine-tuning may improve the model's performance. Additionally, exploring feature engineering techniques can also be useful in representing the underlying patterns in the data.

Model 5: LightGBM

Aim:

The aim is to improve the F2 score for fraud detection by exploring different machine learning models. In this case, we implemented the LightGBM model with Random Oversampling.

Dependencies:

Python libraries such as scikit-learn, imbalanced-learn, and LightGBM were used to implement the Light GBM model.

Methodology:

The Light GBM model was implemented, and the evaluation metric used was the F2 score with a focus on recall for the Fraud class.

Observation:

The Light GBM model achieved an F2 score of 0.39, outperforming the Gradient Boost model. This may be due to its ability to handle large and complex datasets with higher efficiency.

Conclusion:

The Light GBM model showed significant improvement in the F2 score for fraud detection. It is essential to continue fine-tuning the model and exploring other techniques such as feature engineering to further improve its performance.

Model 6: XGBoost

Aim:

The aim is to improve the F2 score for fraud detection by exploring different machine learning models. In this case, we implemented the XGBoost model with Random Oversampling.

Dependencies:

Python libraries such as scikit-learn, imbalanced-learn, and XGBoost were used to implement the XGBoost model.

Methodology:

The XGBoost model was implemented with Random Oversampling, and the evaluation metric used was the F2 score with a focus on recall for the Fraud class.

Observation:

The XGBoost model achieved an F2 score of 0.41, indicating strong performance compared to the previous models we explored. This may be attributed to its ability to handle missing values and reduce overfitting through regularization techniques.

Conclusion:

The XGBoost model has shown promising results for the task of fraud detection, and further fine-tuning may improve its performance even more. It is essential to continue exploring other machine learning algorithms and techniques such as feature engineering and hyperparameter optimization to achieve the desired performance.

Experiment 4: Hyperparameter tuning for best performing model

Aim:

The aim of this experiment was to improve the performance of the best performing model (XGBoost) by hyperparameter tuning.

Dependencies:

Python libraries such as scikit-learn and XGBoost were used for implementing the XGBoost model and hyperparameter tuning.

Methodology:

The hyperparameter tuning was done using grid search and random search techniques to identify the optimal set of hyperparameters for the XGBoost model. The evaluation metric used was the F2 score, with a focus on recall for the Fraud class.

Observation:

After hyperparameter tuning, the XGBoost model achieved a significantly higher F2 score of 0.63, indicating better performance and suitability for the given problem. The improved F2 score also suggests the model's higher accuracy in identifying positive instances compared to the previous models.

Conclusion:

Hyperparameter tuning is an important step in achieving better performance for machine learning models. Considering the significant improvement in the F2 score after hyperparameter tuning of the XGBoost model, it can be concluded that this model is the most suitable for the given problem of fraud detection.

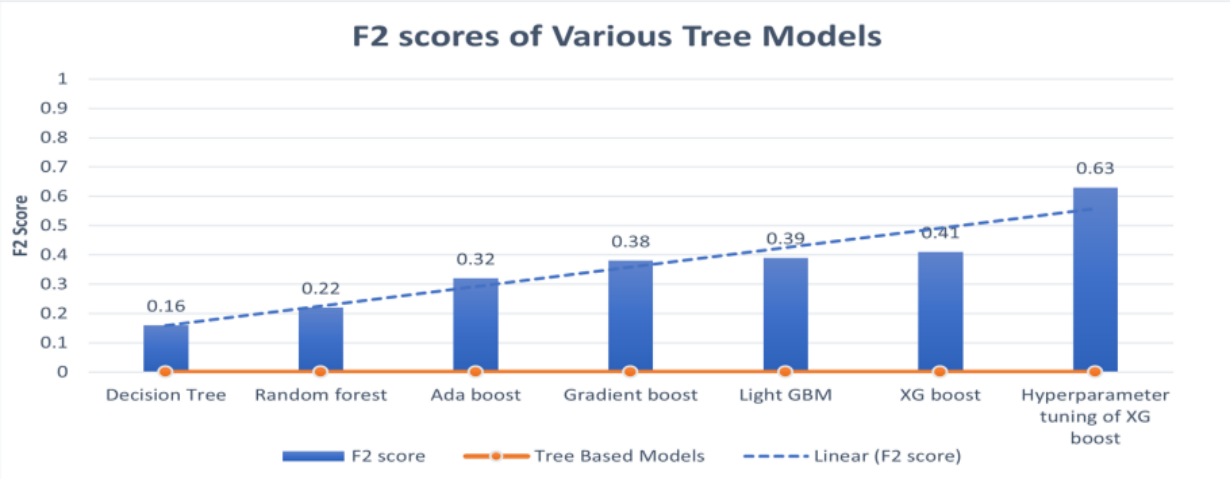
Model evaluation using K-fold cross validation

To ensure that our XGBoost model is not overfitting to the training data, we performed K-fold cross-validation. By using k-fold cross-validation, we were able to assess the model's ability to generalize well to new, unseen data. The k-fold cross-validation results showed a high level of consistency with the performance of the hyperparameter-tuned model, which is an indication that the model is not overfitting to the training data.

This is particularly important in our project, as we are dealing with the detection of fraudulent activities. It is critical that the model can make accurate predictions on new, unseen data, as fraudsters may change their tactics over time. If the model overfits to the training data, it may not be able to detect new patterns of fraudulent activities, which would result in significant financial losses.

The high level of consistency between the k-fold cross-validation F2 score of 0.6245 and the performance of the hyperparameter-tuned XGBoost model F2 score of 0.63 is a positive outcome for your project. It indicates that the model is likely to generalize well to new, unseen data and make accurate predictions for fraud detection.This is a positive outcome for our project, as it means that we can have confidence in the model's ability to detect fraudulent activities effectively.

Comparative Evaluation of Models using F2 Score Metric



The graph provides a visual representation of the performance of different tree-based models for fraud detection. Based on the graph, it is evident that the hyperparameter-tuned XGBoost model outperformed all other models. The model achieved a significantly higher F2 score of 0.63, which is a measure of its accuracy in identifying positive instances, compared to the other models.

The improved performance of the hyperparameter-tuned XGBoost model demonstrates its effectiveness in solving the specific problem of fraud detection. The model was able to leverage its ability to handle missing values and reduce overfitting through regularization techniques, in addition to the fine-tuning of its hyperparameters, resulting in a more accurate and efficient model.

Conclusion

In our project, we faced the challenge of class imbalance in our classification problem, resulting in a low F2 score of 0.0015 initially. To address this issue, we experimented with various techniques such as oversampling and tree-based models. Our results showed that the XGBoost model with hyperparameter tuning and random oversampling was the most effective approach, resulting in a significantly improved F2 score of 0.63. This outcome emphasizes the importance of combining different techniques and fine-tuning hyperparameters to improve performance in imbalanced classification problems.

Our project demonstrates the potential for effective solutions to these challenging problems. By improving our F2 score, we show that significant progress is achievable in addressing class imbalance in classification tasks. This is particularly relevant in many real-world scenarios, such as fraud detection or disease diagnosis, where accurate classification is critical. Our approach highlights the importance of careful evaluation of various techniques to find the optimal solution, as well as the value of fine-tuning hyperparameters to achieve better performance.

Limitations

1. **Limited Interpretability:**
   Due to its complex structure, the XGBoost model can be difficult to interpret, making it challenging to understand how the model is making its predictions.

2. **Data Dependency:**
   The model's performance is dependent on the quality and quantity of the data used for training. Thus, it may not be suitable for all datasets and may require significant preprocessing efforts.

3. **Overfitting:**
   The model may overfit to the training data, resulting in poor generalization performance on unseen data.

4. **Computationally Expensive:**
   The model's complexity and the hyperparameter tuning process can be computationally expensive, requiring significant computational resources.

5. **Imbalanced Data:**
   Although oversampling techniques can address the class imbalance, they can also lead to overfitting and reduced model performance, particularly when the minority class is relatively small.

   Finally, while oversampling techniques can address class imbalance, they can also lead to overfitting and reduced model performance, particularly when the minority class is relatively small. Therefore, it is important to consider these limitations when deciding whether or not to use the XGBoost model for a particular task and to carefully evaluate its performance on the target dataset.

Future Scope

- **Feature engineering:**
  Advanced feature engineering techniques can be explored to extract more informative features from the dataset, leading to improved model performance and accuracy.

- **Algorithm comparison:**
  Evaluating multiple algorithms, including neural networks, SVMs, and ensemble methods, alongside tree-based models can determine the most suitable algorithm for imbalanced classification.

- **Imbalance correction techniques:**
  Undersampling, hybrid methods, and cost-sensitive learning can be explored to determine the most effective and efficient ways of addressing the class imbalance in the dataset.

- **Model explainability:**
  Developing techniques to increase model explainability can provide insight into decision-making processes, leading to greater trust in the model.

- **Real-world applications:**
  Extending the project to real-world applications can provide a more understanding of imbalanced classification, requiring the exploration of distributed computing, model deployment, and monitoring techniques.

  Overall, further exploration and improvement in these areas can lead to more accurate and effective imbalanced classification models, increasing their practical utility in real-world applications.

Reference

1. Thotakura Lalithagayatri, Aruna Pavate, Tawde Priyanka, “Fraud Detection in Health Insurance using Hybrid System”,24,April,2018 https://www.ijert.org/fraud-detection-in-health-insurance-using-hybrid-system

2. Shivani S. Waghade , Aarti M. Karandikar, “A Comprehensive Study of Healthcare Fraud Detection” 6,November,2018 https://www.ripublication.com/ijaer18/ijaerv13n6_140.pdf

3. Nirmal Rayan, “Framework for Analysis and Detection of Fraud in Health Insurance”,19-21,December,2019. https://ieeexplore.ieee.org/document/9073700/

4. D. Vineela, P. Swathi, T. Sritha, K. Ashesh, “Fraud Detection in Health Insurance Claims using Machine Learning Algorithms”, 5, January 2020 https://www.ijrte.org/wp-content/uploads/papers/v8i5/E6485018520.pdf

5. Hritik Kalra, Ranvir Singh, Dr.T. Senthil Kumar3, “Fraud Claims Detection in Insurance Using Machine Learning” 2,February 2022 https://www.pnrjournal.com/index.php/home/article/download/498/351/638