SASTRA

ENGINEERING · MANAGEMENT · LAW · SCIENCES · HUMANITIES · EDUCATION

DEEMED TO BE UNIVERSITY

(U/S 3 of the UGC Act,1956)

THINK MERIT | THINK TRANSPARENCY | THINK SASTRA

HEALTHCARE PROVIDER FRAUD DETECTION ANALYSIS

Submitted by
125018062
Sindhuja S
B.tech-Computer Science and Business systems
Sastra Deemed University
Thanjavur

Submitted to
Swetha Varadarajan

Table of Contents

1. Abstract

Healthcare fraud detection is a crucial part of maintaining the financial health and integrity of healthcare systems. This project utilizes Medicare claims data to classify healthcare providers as fraudulent or non-fraudulent using machine learning algorithms. Three models—Logistic Regression, Random Forest, and MLP Classifier—were implemented to uncover fraud patterns embedded in claims data. Key preprocessing steps involved handling missing data, feature engineering, and target variable creation. The MLP Classifier achieved the highest accuracy of 91.65%, making it the most effective model. This project highlights the importance of handling imbalanced datasets and demonstrates how advanced machine learning techniques can detect fraudulent providers.

2. Introduction

2.1 Significance of the Dataset

The dataset, which includes Medicare claims, covers a wide range of inpatient, outpatient, and beneficiary data. Identifying fraudulent claims within this data is vital for preventing financial losses and maintaining the efficiency of healthcare systems. The dataset contains valuable features like claim amounts, diagnosis codes, and patient demographics, which help detect fraud patterns.

Input: Claims data including diagnosis codes, admission/discharge dates, reimbursement amounts, and patient demographics.

Output: A binary classification label indicating whether the provider is fraudulent

2.2 Objectives

Healthcare fraud detection involves identifying providers submitting fraudulent claims, which cost healthcare systems billions annually. The objective of this project is to create an effective machine learning model to classify providers as either fraudulent or non-fraudulent based on claims data. These models help reduce unnecessary claims and improve healthcare efficiency.

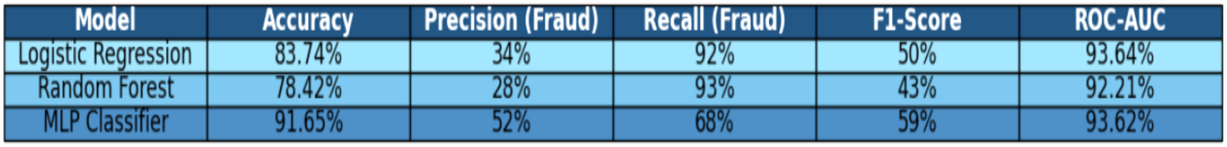
2.3 Approach

The task is framed as a binary classification problem. Three machine learning models—Logistic Regression, Random Forest, and MLP Classifier—are implemented.

The dataset is preprocessed by handling missing values and engineering features such as the number of procedures (NumProc) and claim duration (AdmissionDays). Model performance is evaluated using metrics like accuracy, precision, recall, and ROC-AUC.

2.4 Results

- **MLP Classifier**: Achieved 91.65% accuracy, with a balance of 52% precision and 68% recall for fraud detection.
- **Logistic Regression**: Provided 83.74% accuracy and high recall for fraud (92%) but lower precision (34%).
- **Random Forest**: Showed 93% recall after tuning but still had lower precision (28%).



Model	Accuracy	Precision (Fraud)	Recall (Fraud)	F1-Score	ROC-AUC
Logistic Regression	83.74%	34%	92%	50%	93.64%
Random Forest	78.42%	28%	93%	43%	92.21%
MLP Classifier	91.65%	52%	68%	59%	93.62%

3. Related Work

3.1 Sources of Reference

This project draws inspiration from multiple sources, including datasets and similar studies from Kaggle, academic research papers on fraud detection in healthcare, and consultations with ChatGPT for technical advice and model selection.

3.2 References

1. **Kaggle**:
    - Rohitrox, "Healthcare Provider Fraud Detection Analysis Dataset", Kaggle. Available: https://www.kaggle.com/datasets/rohitrox/healthcare-provider-fraud-detection-analysis.
    - Nissy Joy, "Healthcare Fraud Detection", Kaggle. Available: https://www.kaggle.com/code/nissyjoy/healthcare-fraud-detection.

2. **Research Papers**:
    - Iroju, O., et al. "Healthcare Fraud Detection Using Machine Learning." MDPI Journal of Risks, Vol. 11, Issue 9, 2023. Available: https://www.mdpi.com/2227-9091/11/9/160.
    - IEEE, "Fraud Detection in Healthcare Using ML Models", 2023. Available: https://ieeexplore.ieee.org/document/10696476.

4. Background

4.1 Models Used

Logistic Regression:

Logistic Regression served as the baseline model. It provides a simple yet interpretable approach, utilizing features like reimbursement amounts and chronic conditions to predict fraud.

Random Forest Classifier:

Random Forest, a more complex model, was used to handle non-linear relationships and imbalanced data. Hyperparameter tuning was performed using **GridSearchCV** to optimize parameters such as `n_estimators` and `min_samples_split`.

MLP Classifier:

The MLP (Multilayer Perceptron) Classifier, a neural network-based model, was employed to capture more complex patterns in the data. It used a 2-layer architecture and logistic activation for binary classification.

4.2 Preprocessing Techniques

- **Handling Missing Data**: Missing values in critical columns like `DOD` (date of death) were filled, and features such as `NumProc`, `NumPhysicians`, and `AdmissionDays` were created.
- **Feature Engineering**: Derived features like the number of procedures (`NumProc`), number of unique claims (`NumUniqueClaims`), and patient age were calculated.
- **Target Label Creation**: The target variable, `PotentialFraud`, was mapped to binary labels (1 for fraud, 0 for non-fraud).

5. Methodology

5.1 Experimental Design

The dataset was split into **70% training** and **30% validation**. Each model was trained on the training data and evaluated on the validation set. The following evaluation metrics were considered:

- **Accuracy**: Overall correctness of the model.
- **Precision, Recall, F1-Score**: Important for classifying fraud accurately.
- **ROC-AUC**: Measures the area under the ROC curve, focusing on model discrimination between classes.

5.2 Tools and Environment

- **Pandas and NumPy**: For data manipulation, cleaning, and feature engineering. These libraries allowed for efficient handling of large datasets, transforming raw data into structured formats ready for analysis.
- **Scikit-learn**: For model implementation and evaluation. This library provided essential machine learning algorithms such as Logistic Regression, Random Forest, and MLP Classifier. It also enabled easy use of model evaluation metrics and hyperparameter tuning via GridSearchCV.
- **Matplotlib and Seaborn**: For data visualization, these libraries helped create plots like pair plots, heatmaps, ROC curves, and precision-recall curves. These visualizations were crucial for interpreting the model's performance and understanding the relationships within the dataset.
- **Google Colab**: For code execution.

5.3 Code Locations

The code for preprocessing, model training, and evaluation is available in the Google Colab notebook and GitHub repository.

5.4 Preprocessing Steps

Dataset Size

The size of the combined dataset is substantial, with over `138,556 records` and `25 features.`The dataset includes claims data from multiple sources, including inpatient, outpatient, and beneficiary records. This comprehensive data is used to detect fraud by analyzing various aspects such as diagnosis codes, admission dates, reimbursement amounts, patient demographics, and more.

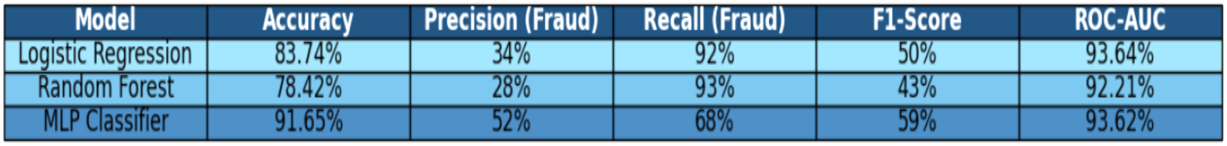
Feature Size

Key features extracted from the dataset include total reimbursement amounts, the number of procedures (e.g., `NumProc`), and patient demographics (e.g., `Gender`, `Race`, `Chronic Conditions`). These features are essential in distinguishing fraudulent providers from non-fraudulent ones. Additional features such as the number of unique claims (`NumUniqueClaims`), admission days, and the number of physicians involved in a claim were also engineered to enrich the dataset.

Outlier Analysis and Feature Reduction

Outliers were identified based on unusually high reimbursement claims, which could indicate fraudulent activities. For instance, claims with excessive `ReimbursementAmounts` or an unusually high number of procedures were flagged for further analysis. Despite the presence of outliers, feature reduction was not applied, as all features were deemed valuable for model performance.

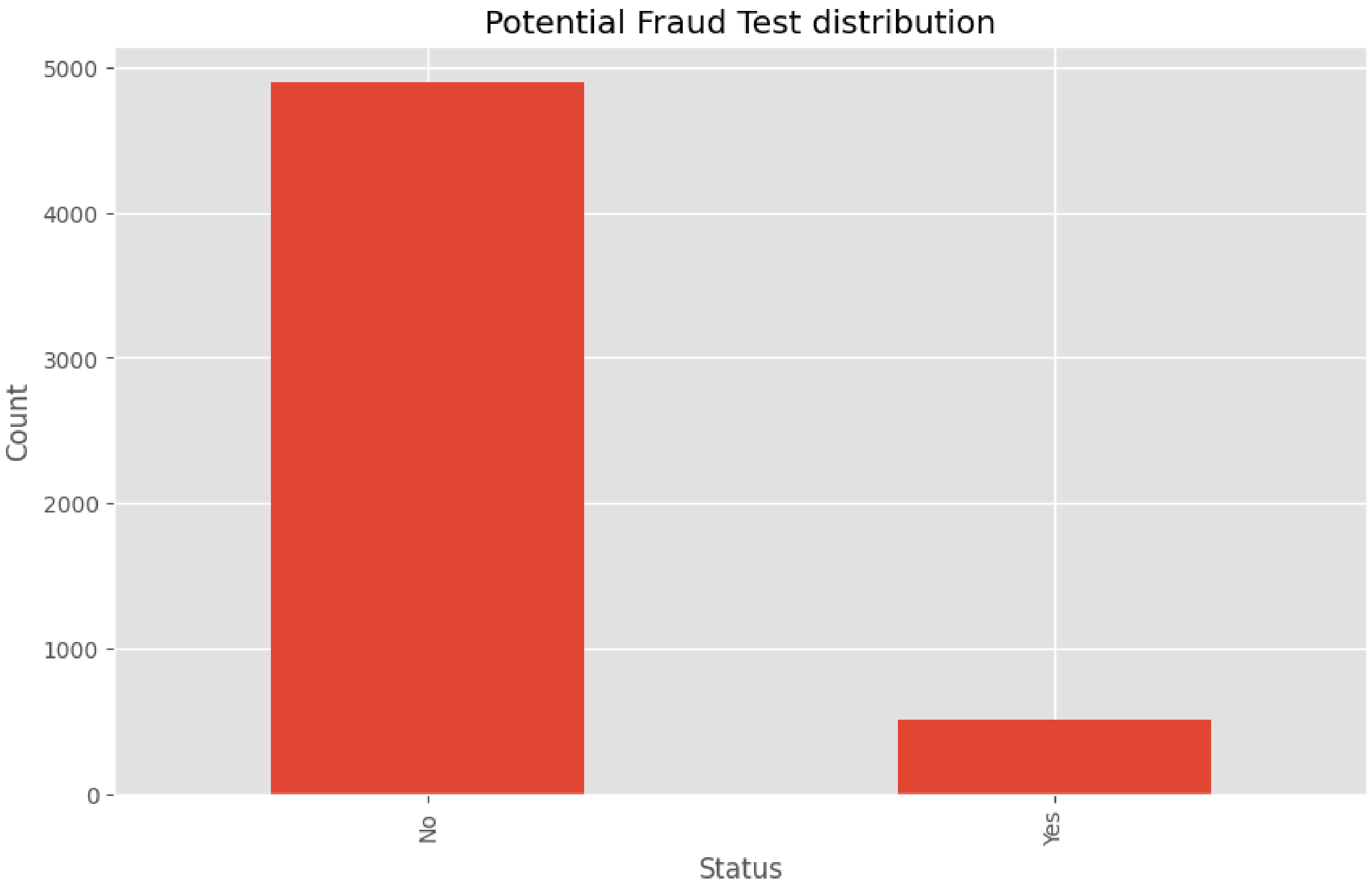
6. Results



Model	Accuracy	Precision (Fraud)	Recall (Fraud)	F1-Score	ROC-AUC
Logistic Regression	83.74%	34%	92%	50%	93.64%
Random Forest	78.42%	28%	93%	43%	92.21%
MLP Classifier	91.65%	52%	68%	59%	93.62%

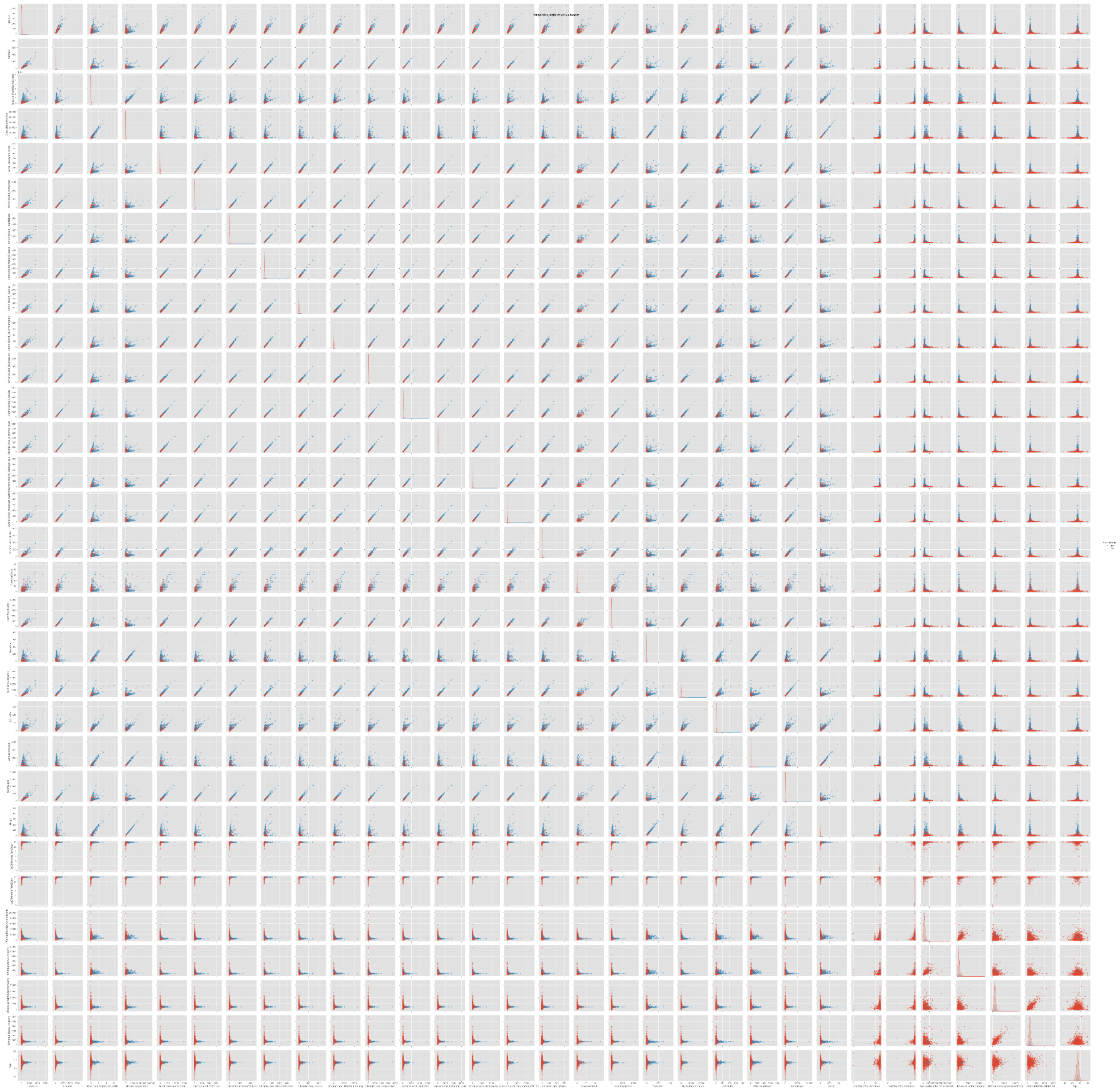
4.1 Fraud Distribution Among Providers

This bar chart shows the distribution of fraudulent vs. non-fraudulent providers. It clearly illustrates the **class imbalance**, where the majority of providers are non-fraudulent. Handling this imbalance was a key challenge in training machine learning models.



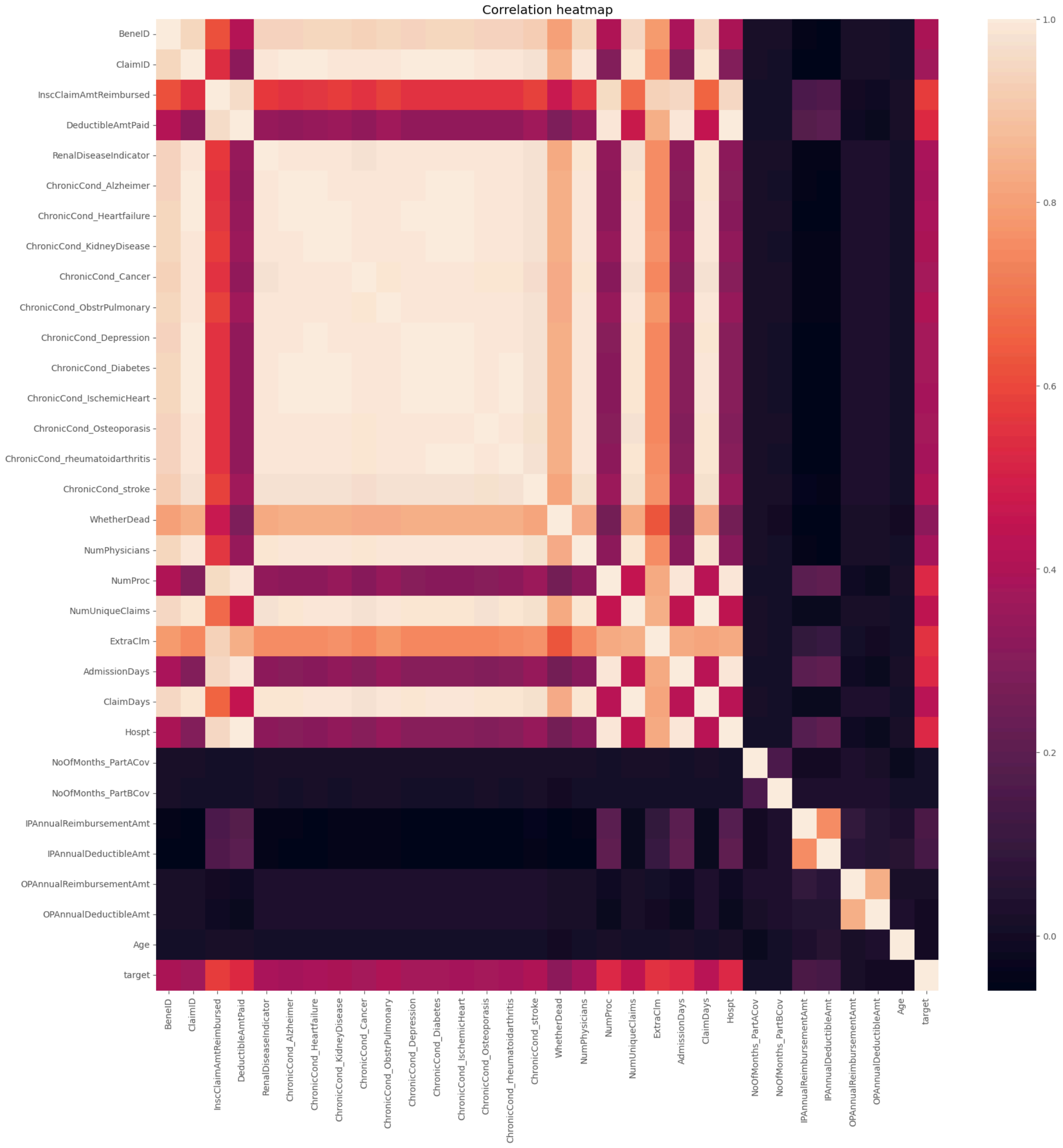
4.2 Pairwise Relationships of Features (Seaborn Pairplot)

We used Seaborn's pairplot to examine the relationships between different features for providers labeled as fraudulent (`PotentialFraud = 1`) and non-fraudulent (`PotentialFraud = 0`). This plot highlights key pairwise interactions between features, such as `TotalCharges`, `NumClaims`, `AdmissionDays`, etc.



4.3 Correlation Heatmap of Features

The correlation heatmap provides insight into how different features are correlated with one another. Features such as `ReimbursementAmounts`, `NumProc`, and `AdmissionDays` show strong correlations, which could be useful in predicting fraudulent claims.

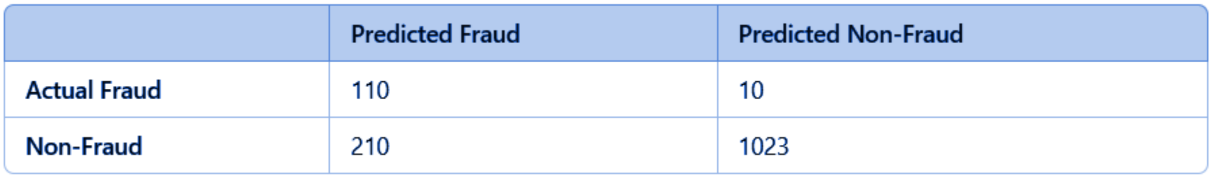


4.4 Logistic Regression Results

- Accuracy: 83.74%
- Precision for fraud: 34%
- Recall for fraud: 92%
- F1-score: 50%

While Logistic Regression achieved a relatively high recall for fraud, it had a poor precision (only 34%). This indicates that while many fraudulent providers were detected, the model also falsely flagged many non-fraudulent providers.

Confusion Matrix:



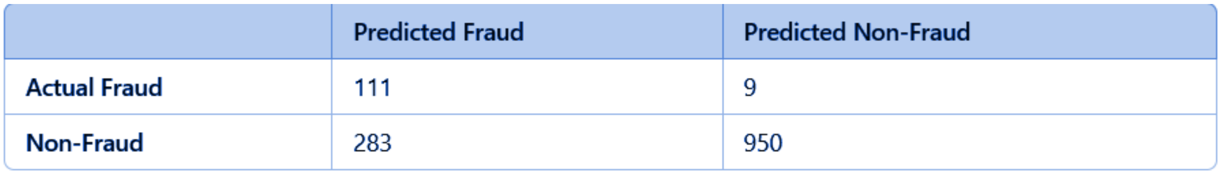
	Predicted Fraud	Predicted Non-Fraud
Actual Fraud	110	10
Non-Fraud	210	1023

4.5 Random Forest Classifier Results

- **Accuracy**: **78.42%**
- **Precision for fraud**: **28%**
- **Recall for fraud**: **93%**
- **F1-score**: **43%**

Random Forest improved recall to 93%, but it still had lower accuracy due to many false positives. After tuning the hyperparameters (`n_estimators = 60` and `min_samples_split = 0.25`), the model performance slightly improved.

Confusion Matrix:



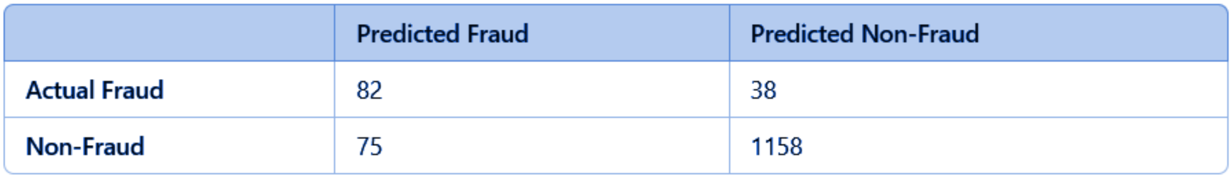
	Predicted Fraud	Predicted Non-Fraud
Actual Fraud	111	9
Non-Fraud	283	950

4.6 MLP Classifier Results

- **Accuracy**: **91.65%**
- **Precision for fraud**: **52%**
- **Recall for fraud**: **68%**
- **F1-score**: **59%**

The MLP Classifier achieved the best performance, balancing both precision and recall. Its higher accuracy, F1-score, and reduced false positives made it the most effective model for this dataset.

Confusion Matrix:



	Predicted Fraud	Predicted Non-Fraud
Actual Fraud	82	38
Non-Fraud	75	1158

ROC-AUC Scores:

- **Logistic Regression**: 93.64%
- **Random Forest**: 92.21%
- **MLP Classifier**: 93.62%

7. Discussion

7.1 Overall Model Performance

The MLP Classifier outperformed other models in terms of accuracy and balance between precision and recall. Logistic Regression had high recall but suffered from low precision, while Random Forest performed better after tuning but still struggled with false positives.

7.2 Overfitting and Underfitting

- **Logistic Regression**: Precision issues were linked to underfitting of the fraudulent class.
- **MLP Classifier**: Regularization techniques were employed to prevent overfitting.

7.3 Hyperparameter Tuning

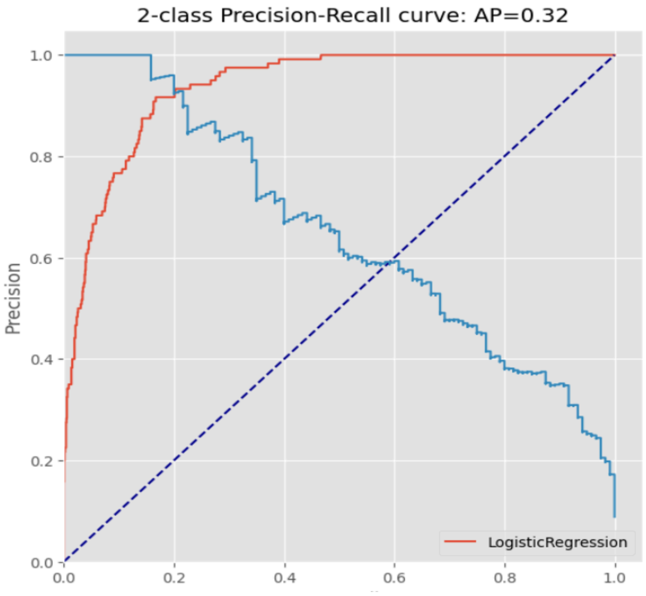
Hyperparameter tuning using **GridSearchCV** was applied to Random Forest to optimize parameters like **n_estimators** and **min_samples_split**, improving recall.

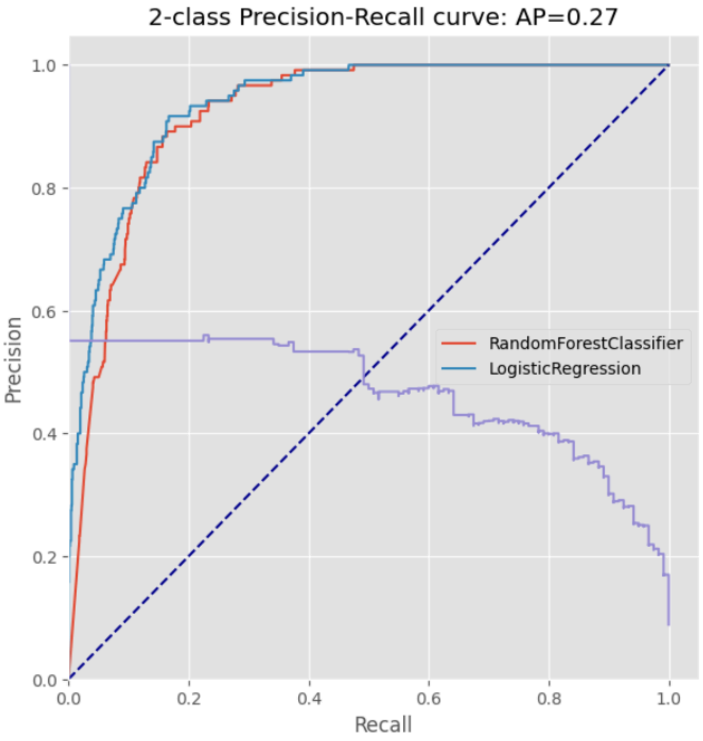
7.4 Model Comparison

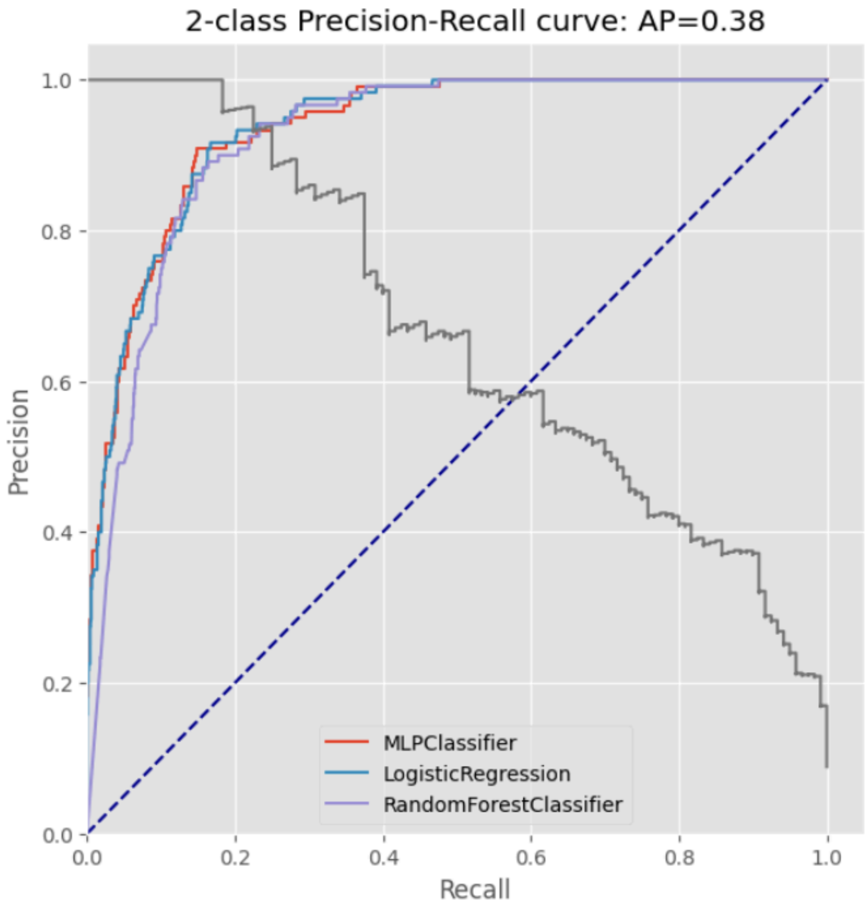
- **Logistic Regression**: Provided high recall for fraud but suffered from poor precision. This made it overly sensitive to fraud, resulting in many false positives.
- **Random Forest**: While better at identifying fraud, it also struggled with precision. However, hyperparameter tuning improved its ability to capture the class imbalance, leading to better recall.

- **MLP Classifier**: Outperformed both Logistic Regression and Random Forest, offering a balance between precision and recall. Its deep learning architecture allowed it to capture more complex patterns in the data.

Figures: ROC Curve Comparison







8. Learning Outcomes

8.1 Google Colab Link

https://colab.research.google.com/drive/1fUMocwpI8G2kdGsfKI_1yVCY82kIQ2uU?usp=sharing

8.2 GitHub Repository Link

https://github.com/Sindhuja016/HEALTHCARE-PROVIDER-FRAUD-DETECTION-ANALYSIS

8.3 Skills Acquired

- **Data Preprocessing and Feature Engineering**:
    - **Handling Missing Data**: Several fields in the dataset, such as `DOD (Date of Death)`, contained missing values. This project improved

skills in handling missing values using techniques like filling, replacing, or dropping missing data, which are critical for preparing clean datasets.

    - **Feature Engineering**: New variables were created from the existing dataset, such as calculating the number of procedures (`NumProc`), hospitalization days (`AdmissionDays`), and the age of patients. These derived features helped enhance the model's performance by making the dataset more informative.
- **Machine Learning Model Implementation**:
    - **Logistic Regression**: Learned to apply logistic regression as a baseline model for binary classification tasks. The simple, interpretable nature of logistic regression made it an excellent starting point for comparison with more advanced models.
    - **Random Forest Classifier**: Gained experience in using Random Forest for handling non-linear patterns and addressing class imbalance. Random Forest's ability to handle high-dimensional data and capture complex interactions between variables was particularly useful.
    - **MLP Classifier**: Learned to implement Multilayer Perceptron (MLP), a type of neural network, which was able to capture more complex patterns and showed the best performance in terms of balancing precision and recall.
- **Model Evaluation and Tuning**:
    - **Performance Metrics**: Understanding and applying metrics such as **accuracy, precision, recall, F1-score**, and **ROC-AUC** to assess model performance. Special emphasis was placed on evaluating models in the context of fraud detection, where precision and recall are more important than accuracy due to the imbalanced nature of the dataset.
    - **Hyperparameter Tuning**: Through the use of **GridSearchCV**, hyperparameters of models (e.g., `n_estimators`, `min_samples_split` for Random Forest) were fine-tuned to improve model performance. This tuning helped optimize models for both precision and recall, especially in handling the minority class (fraudulent providers).
- **Data Visualization**:
    - **Confusion Matrix and Heatmaps:** The ability to visually analyze the relationships between features using correlation heatmaps and to interpret confusion matrices for better understanding of model performance.

8.4 Dataset Used

Medicare claims dataset containing inpatient, outpatient, and beneficiary records.

https://www.kaggle.com/datasets/rohitrox/healthcare-provider-fraud-detection-analysis/data

8.5 Key Learnings

Gained hands-on experience in handling real-world healthcare datasets, implementing machine learning models, and tackling challenges associated with imbalanced data.

- **Binary Classification and Imbalanced Data Handling**: The project helped in mastering how to tackle a binary classification problem with imbalanced datasets. This includes the importance of using appropriate metrics such as **precision, recall, and F1-score**, which provide a more comprehensive picture of model performance when one class is underrepresented.
- **Supervised Learning Algorithms**: In-depth understanding of:
    - **Logistic Regression**: Its simplicity and why it is often used as a baseline model for comparison.
    - **Random Forest**: Its robustness, ability to handle feature importance, and adaptability to non-linear data. This project deepened knowledge of how ensemble learning works, particularly with decision trees.
    - **Neural Networks (MLP Classifier)**: Learned how neural networks like MLPs can outperform traditional models by capturing more complex, non-linear patterns. Additionally, the project demonstrated the trade-offs between computational cost and performance.
- **Fraud Detection Techniques**: The project provided insights into real-world fraud detection scenarios in the healthcare industry. It explored how models can detect fraudulent claims through the analysis of Medicare claims data.
- **Precision-Recall Trade-off**: In fraud detection, a model must balance precision and recall. This project illustrated the difficulty of achieving this balance and the importance of focusing on recall to ensure

9. Conclusion

9.1 Summary of Work

This project successfully implemented machine learning models to identify fraudulent healthcare providers based on Medicare claims data. This project explored different machine learning models to detect fraudulent healthcare providers. The **MLP Classifier** emerged as the best model, achieving an accuracy of **91.65%** and a ROC score of **93.62%**, indicating a strong capability for distinguishing between fraudulent and non-fraudulent cases. The model effectively balanced precision (52%) and recall (68%), although further improvement in precision is necessary for practical applications.

9.2 Project Accomplishment

The project objectives were achieved by accurately identifying fraudulent providers, improving the understanding of fraud detection through machine learning models, and handling imbalanced datasets effectively.

9.3 Advantages and Limitations

- **Advantages**: The models, particularly the MLP Classifier, performed well in detecting fraud.
- **Limitations**: Precision can be further improved to reduce false positives, a crucial requirement for real-world applications.