Medical Insurance Fraud Detection

Aaron Van Der Male   VANDERMALE.A@NORTHEASTERN.EDU
Khoury College of Computer Science
Northeastern University

Editor: Aaron Van Der Male

1. Introduction

Medical insurance fraud involves filing false or inaccurate insurance claims in an attempt to recieve undue insurance payouts. In this project, a synthetic dataset of insurance claims is analyzed to distinguish fraudulent from legitimate insurance claims.

2. Objective

The purpose of this project is to identify and score a machine learning classification model that can accurately classify insurance claims.

3. Dataset

A Kaggle Dataset was analyzed. The csvs are joined in a pandas dataframe before processing:

- Train-1542865627584.csv: Contains provider numbers and value that shows if the provider is potentially fraudulent.

- Test-1542865627584.csv: Contains provider number.

- TrainBeneficiarydata-1542865627584.csv: Beneficiary data such as dob, race, gender, state, chronic condition, annual reimbursement amount, open annual deductible amount.

- TrainInpatientdata-1542865627584.csv

- TrainOutpatientdata-1542865627584.csv

- TestInpatientdata-1542969243754.csv

- TestOutpatientdata-1542969243754.csv

The dataset has 31 features that are to be analyzed. They are introduced in this section. Figure 1 shows the data types of the features.

Field Name	Data Type
inscclaimamtreimbursed	int64
deductibleamtpaid	float64
gender	bool
race	int64
renaldiseaseindicator	int64
state	int64
county	int64
noofmonths partacov	int64
noofmonths partbcov	int64
chroniccond alzheimer	bool
chroniccond heartfailure	bool
chroniccond kidneydisease	bool
chroniccond cancer	bool
chroniccond obstrpulmonary	bool
chroniccond depression	bool
chroniccond diabetes	bool
chroniccond ischemicheart	bool
chroniccond osteoporasis	bool
chroniccond rheumatoidarthritis	bool
chroniccond stroke	bool
ipannualreimbursementamt	int64
ipannualdeductibleamt	int64
opannualreimbursementamt	int64
opannualdeductibleamt	int64
inpt	int64
ageatclaim	float64
duration	int64
los	float64
dead	float64
Att Opr Oth Phy Tot Claims	float64
Prv Tot Att Opr Oth Phys	int64

Figure 1: Feature Data Types

A brief description of each feature is provided in Figure 2

Name	Description
BeneID	The unique id of the beneficiary
Dob	Date of birth of the beneficiary
Dod	Date of death of the beneficiary
Race	race of the beneficiary
Gender	Gender of the beneficiary
RenalDiseaseIndicator	States if patient has kidney disease
ChronicCond	The columns starting with ChronicCond indicates if the patient has that particular chronic disease
IPAnnualReimbursementAmt	Consists of the maximum reimbursement amount for hospitalization annually
IPAnnualDeductibleAmt	Consists of a premium paid by the patient for hospitalization annually
OPAnnualReimbursementAmt	annual maximum reimbursement amount for outpatient visits
OPAnnualDeductibleAmt	Consists of a premium paid by the patient for outpatient visits annually

Figure 2: Feature Names and Descriptions

4. Understanding the Dataset

The frequency of features in the dataset is explored. Figure 3 and 4 depict the amount of data collected along state, gender, and racial divisions. The histograms in figures 5, 6 and 7 characterize the differences between the fraudulent and normal claims.
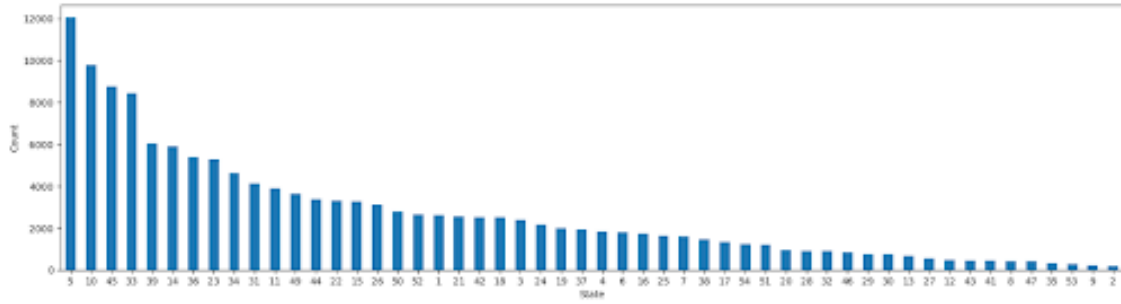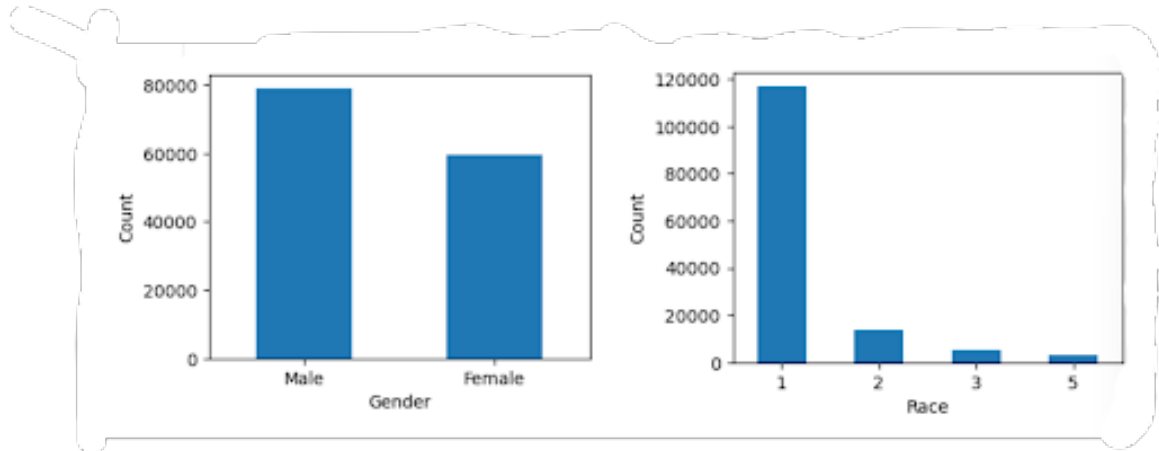

Figure 3: State Distribution
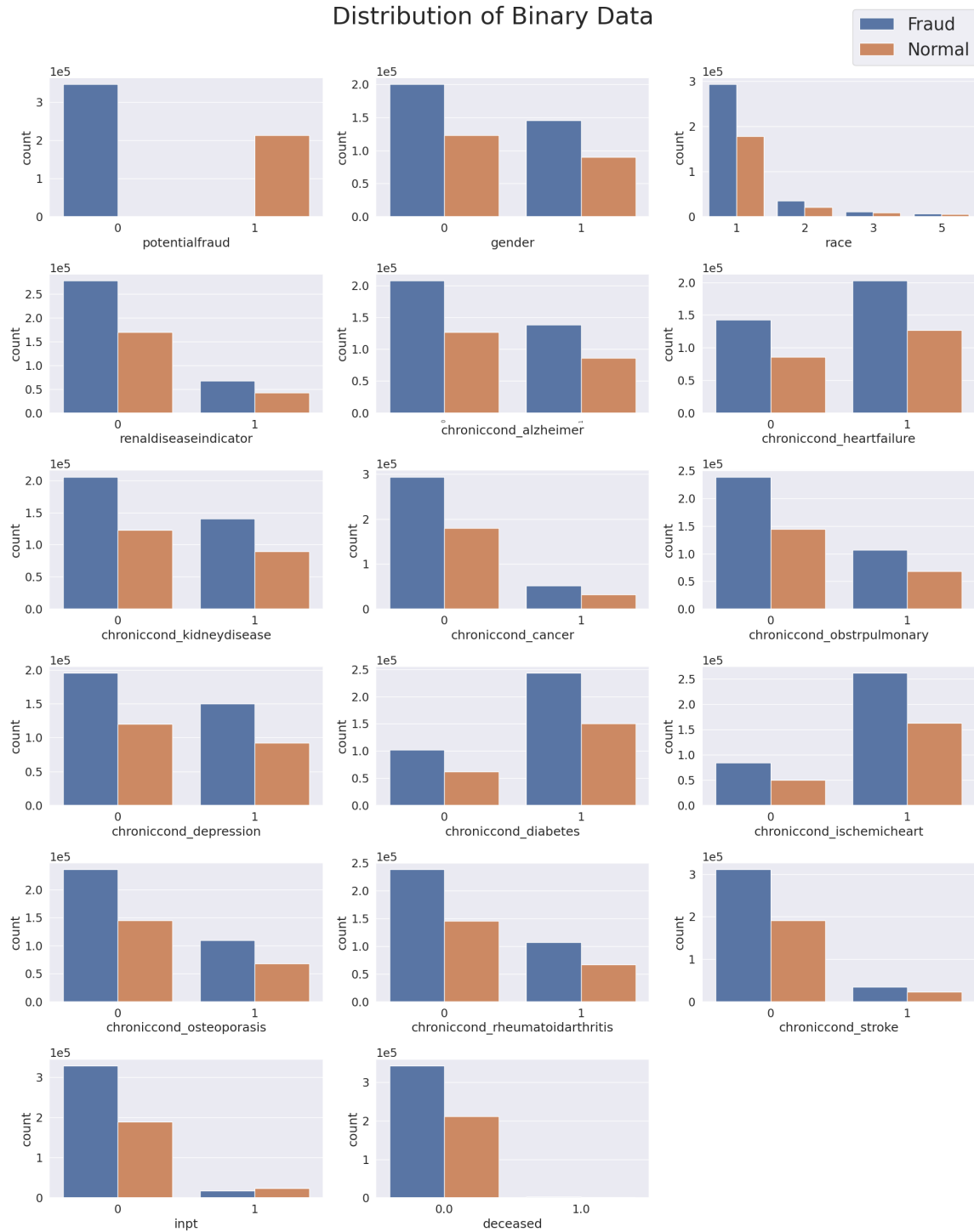

Figure 4: Gender and Race Distribution

Figure 5: Distribution of Binary Data
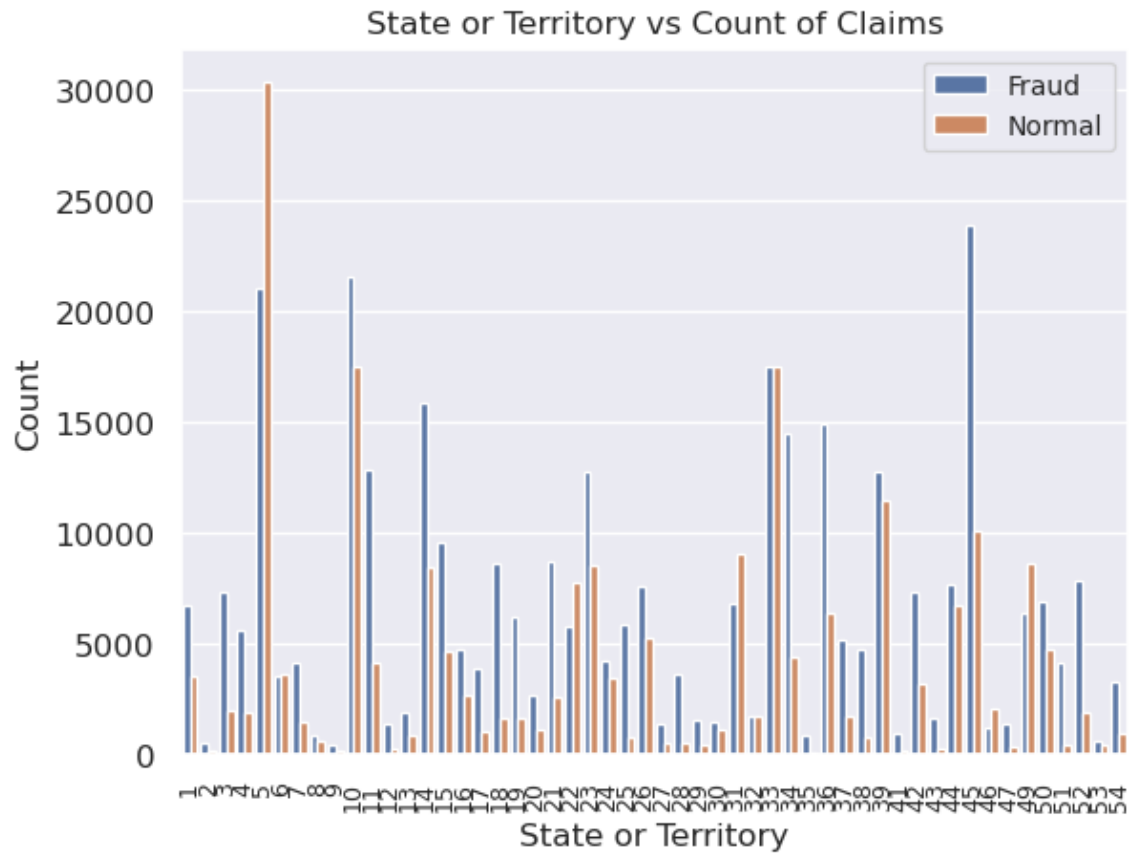
Figure 6: Distribution of State or territory Data. The names of the states are not provided with this dataset, and the data are instead identified with a number.
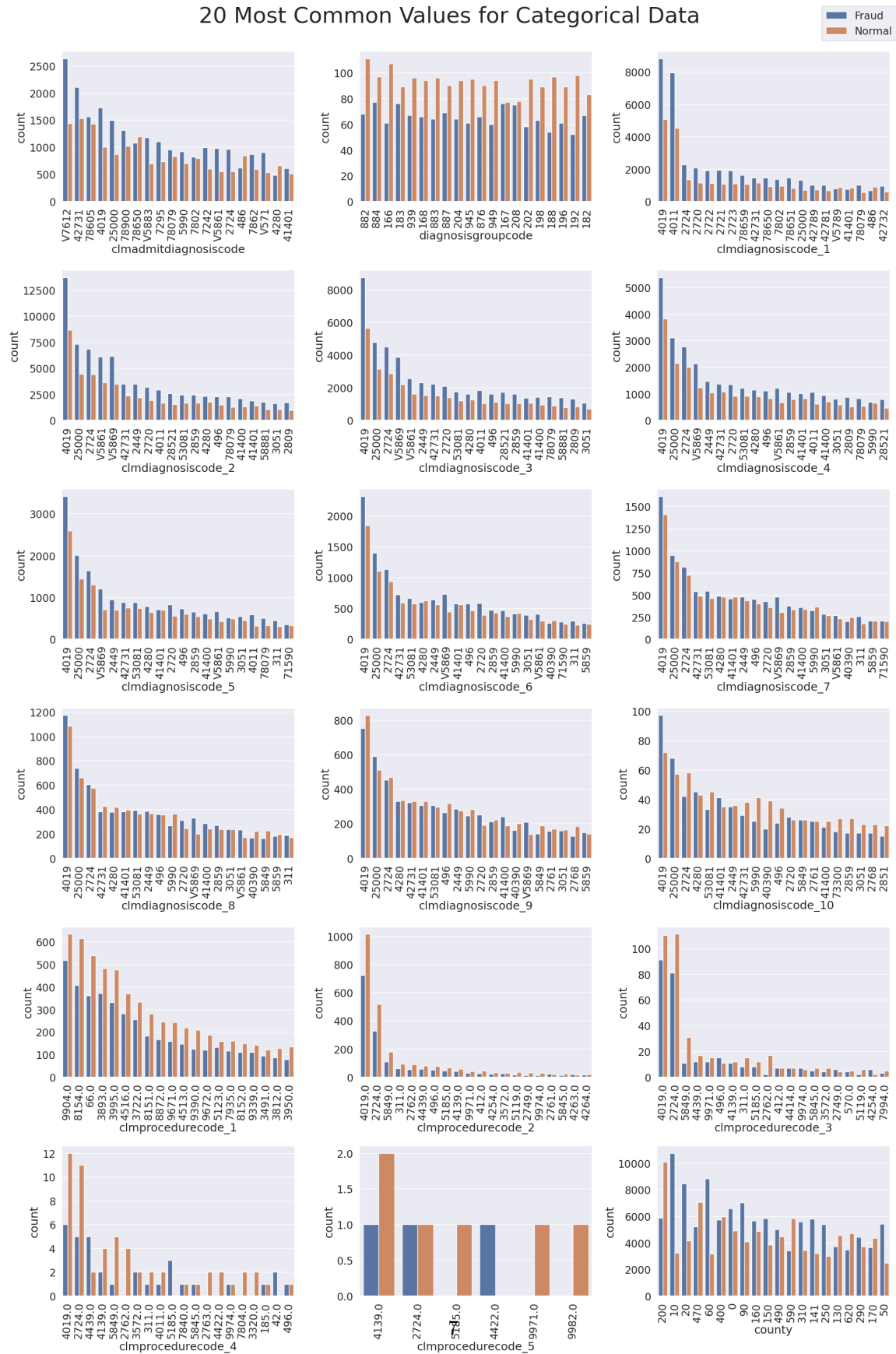
Figure 7: 20 Most Common Values for Categorical Data

To analyze the quantitative data, the boxplots are generated (Figure 7). The values are normalized on a 0-1 scale. The plots show that certain attributes are extremely concentrated and include a number of outliars, such as deductibleamtpaid.
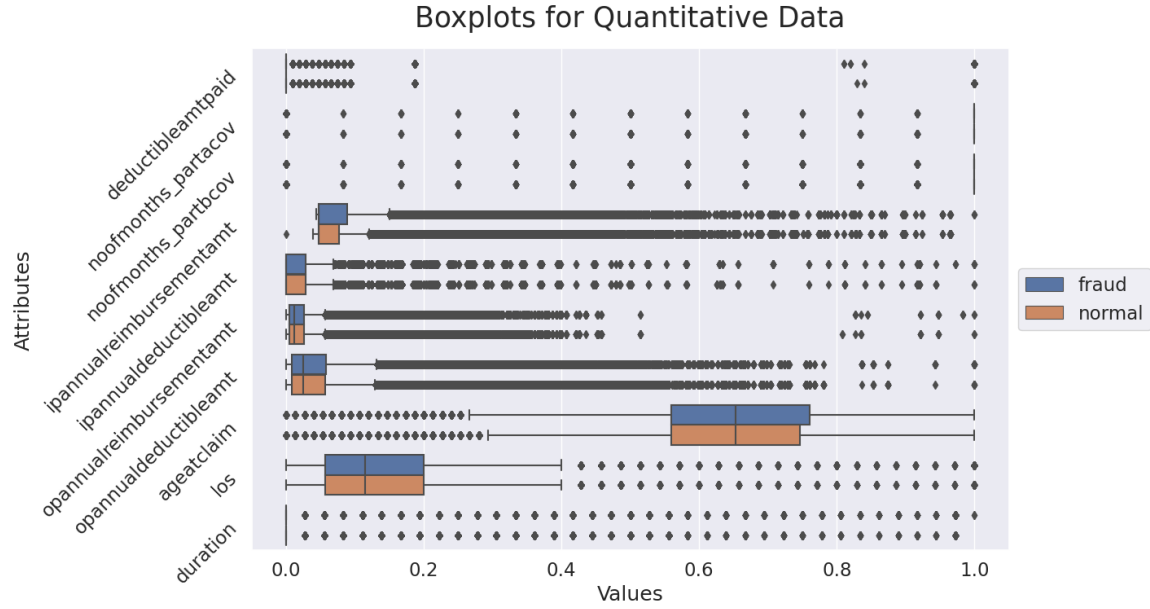

Figure 8: Boxplots for Quantitative Data

PCA is applied in Figure 9, and a subset of the data is plotted in three space.
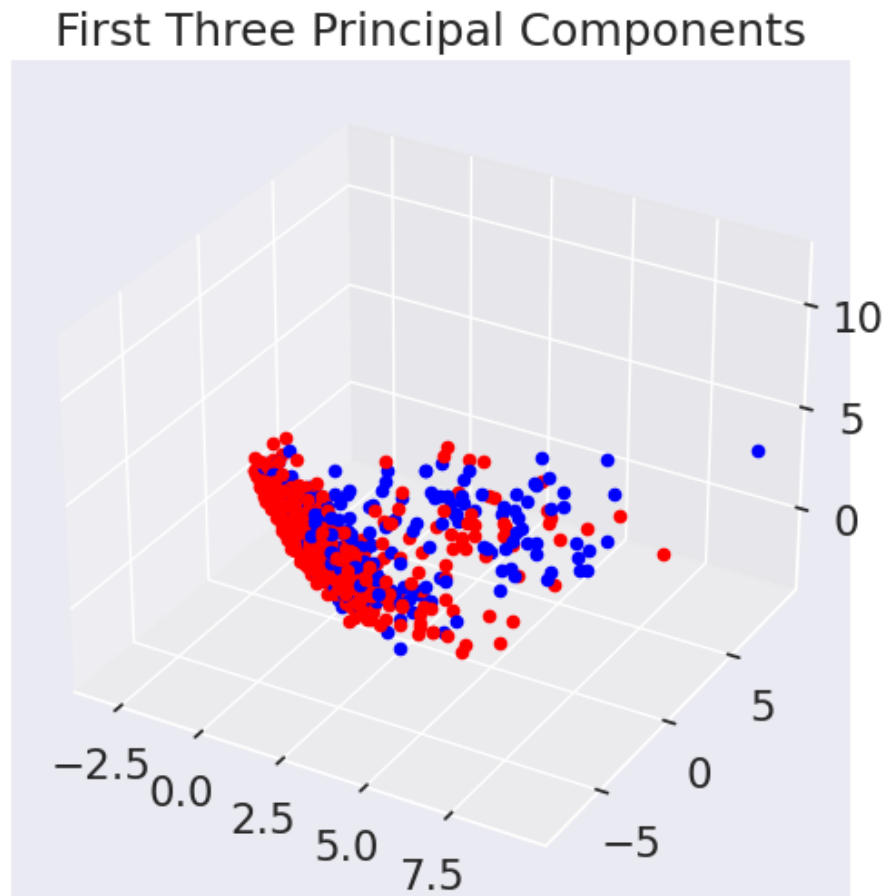
First Three Principal Components

Figure 9: First three principal components of binary and quantitative data. The class of data is separated in blue and red. A degree of clustering is observed, meaning these features are predictive.

A heatmap of the first three principal components is plotted. Lighter colors imply A positive correlation, and dark colors show a negative correlation. Either can demonstrate that the feature is predictive.
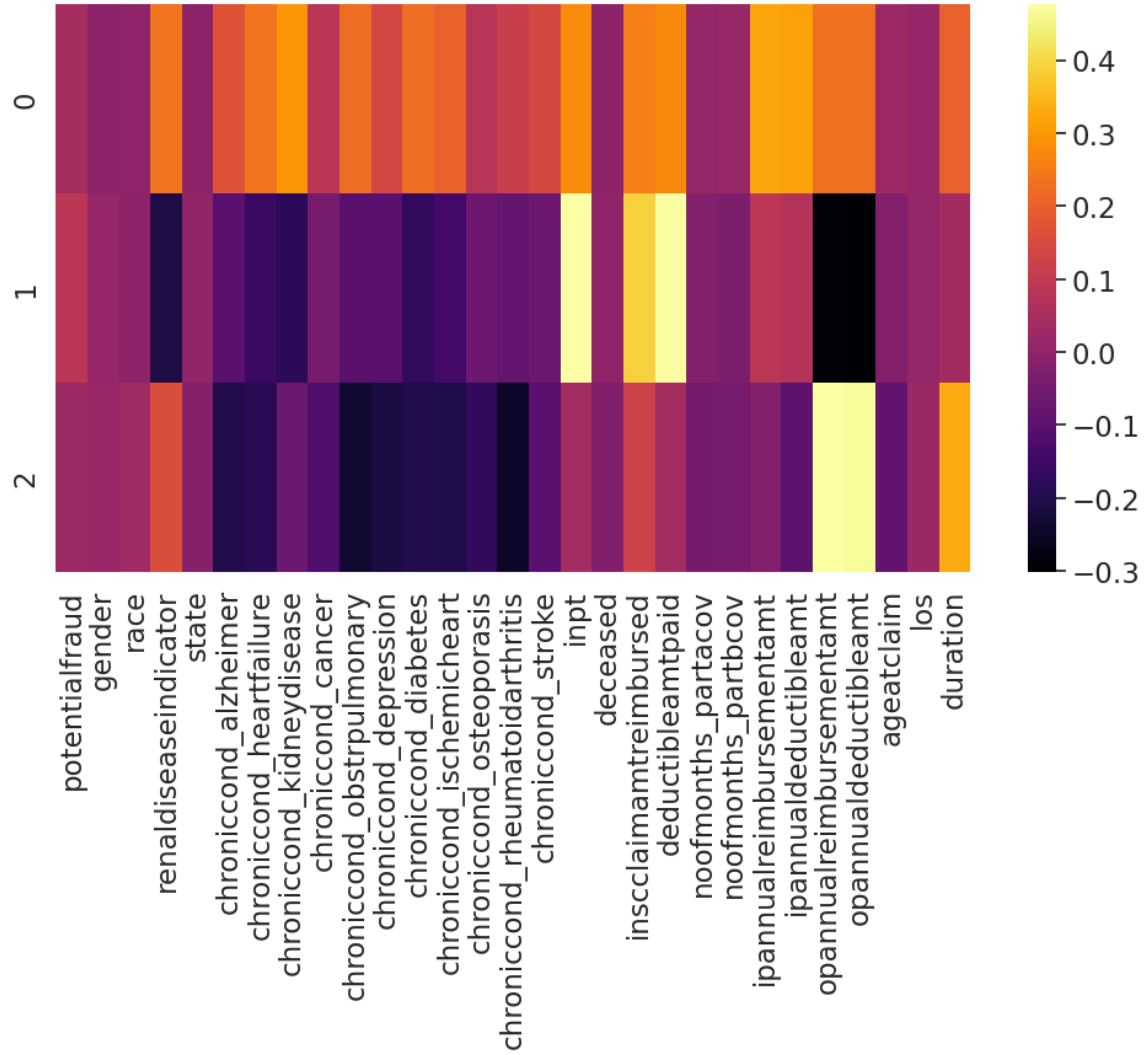

Figure 10: Colorbar of principal components

The correlation between quantitative features is investigated in the following chart. A pair of features that are highly correlated with each other can suggest that only one of the features is needed. In this case, duration and los appear to be redundant. However, because of the PCA results (Figure 10), both features will be kept.
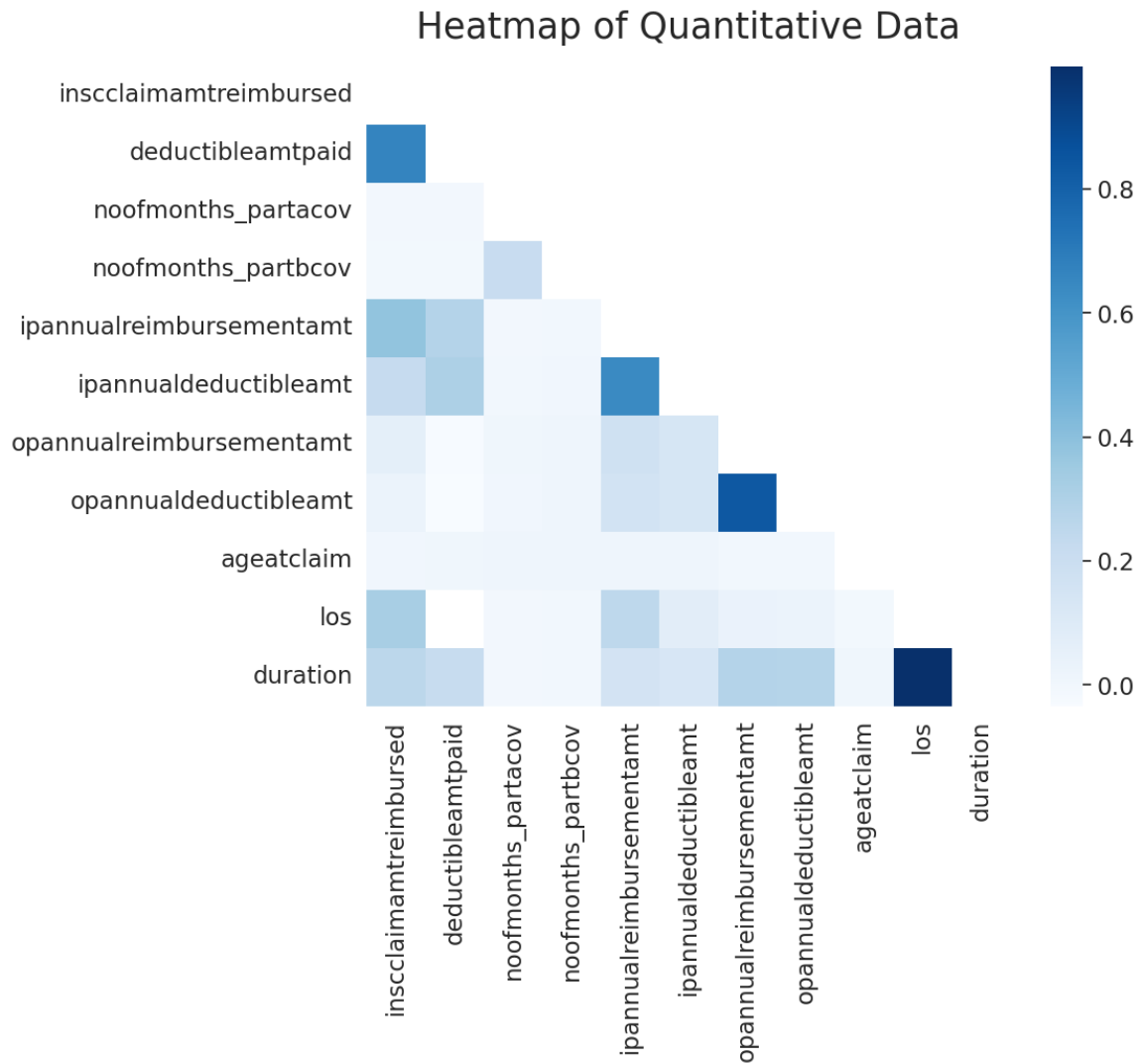

Figure 11: Heatmap for Quantitative Data. Darker colors show the feature pairs are more highly correlated

The following scatter (figure 12) plot shows the relationship between the number of physicians and the percentage of fraudulent claims. The vertical lines on the plot indicate the number of physicians at which the percentage of fraudulent claims reaches 80%, 90%, and 99%.

The plot suggests that a relatively small number of physicians are responsible for a large proportion of fraudulent claims.
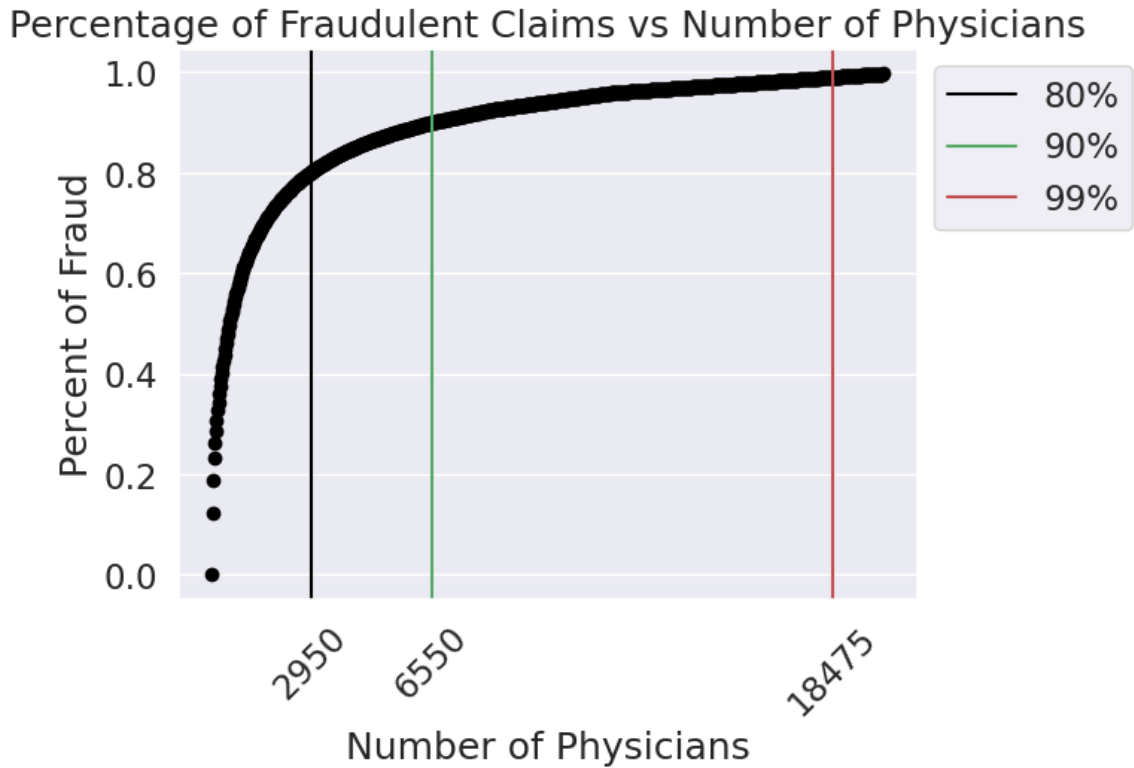

Figure 12: The percentage of fraudulent claims vs number of physicians At the black vertical line, the first 2950 physicians with the most fraudulent claims account for 80% of all fraudulent claims. This represents just 0.53% of all physicians

5. Preprocessing

After observing the dataset, we find only the train dataset has the label (fraudulent or not). Therefore, we choose to use the train dataset and split the dataset into train and validation sets later. Firstly, we import all datasets, we use the function named col_header_clean to switch all characters in columns to lowercase, remove spaces and special chars. Then, to make the training set ML processible, we merge the datasets into one. The inpatient and outpatient sets have similar features, so they can be merged.

An inpatient indicator column will be created to distinguish inpatient and outpatient claims. Then, values of binary categorical variables ('Y'/'N' or '1'/'2') are converted to 1 and 0. Also, values with date and time data will be converted to the correct form. Based on the dates, we also calculate information such as the patient's age at the time of claim, duration of the claim, and length of hospital stay. Finally, we remove features having no records, and add derived features including total count of claims filed by physicians and total number of physicians for each provider.

6. Models and Training

The following classifiers were evaluated. Out of the parameters tested, the ones provided here yielded the best performance.

- Adaboost Classifier
  - n_estimators=100
- Extra Trees Classifier
  - n_estimators=100
- Gradient Boosting Classifier
  - n_estimators=1000
  - maxdepth=8
- K Nearest Neighbors
  - n=3
- Random Forest Classifier
  - n_estimators=1000

The performance of each classifier is summarized in table 1-5.

Table 1: Adaboost Classifier Scores

class	precision	recall	f1-score	support
Normal	0.843	0.976	0.904	69244
Fraudulent	0.946	0.702	0.806	42399

Table 2: Extra Trees Classifier Scores

class	precision	recall	f1-score	support
Normal	0.902	0.972	0.936	69244
Fraudulent	0.947	0.828	0.884	42399

Table 3: Gradient Boosting Classifier Scores

class	precision	recall	f1-score	support
Normal	0.992	0.998	0.995	69244
Fraudulent	0.997	0.987	0.992	42399

Table 4: K Nearest Neighbors Scores (n=3)

class	precision	recall	f1-score	support
Normal	0.865	0.909	0.886	69244
Fraudulent	0.837	0.769	0.802	42399

Table 5: Random Forest Scores

class	precision	recall	f1-score	support
Normal	0.922	0.991	0.955	69244
Fraudulent	0.983	0.863	0.919	42399

The receiver operating characteristic curve is plotted for the algorithms.
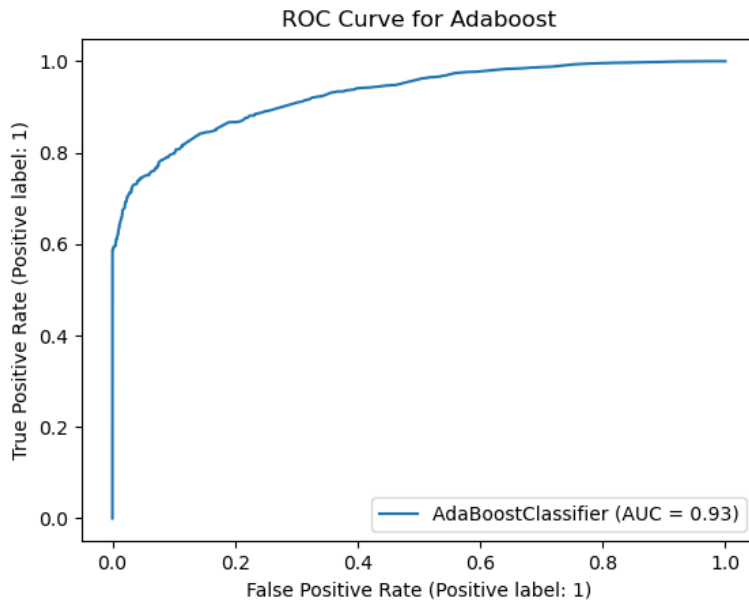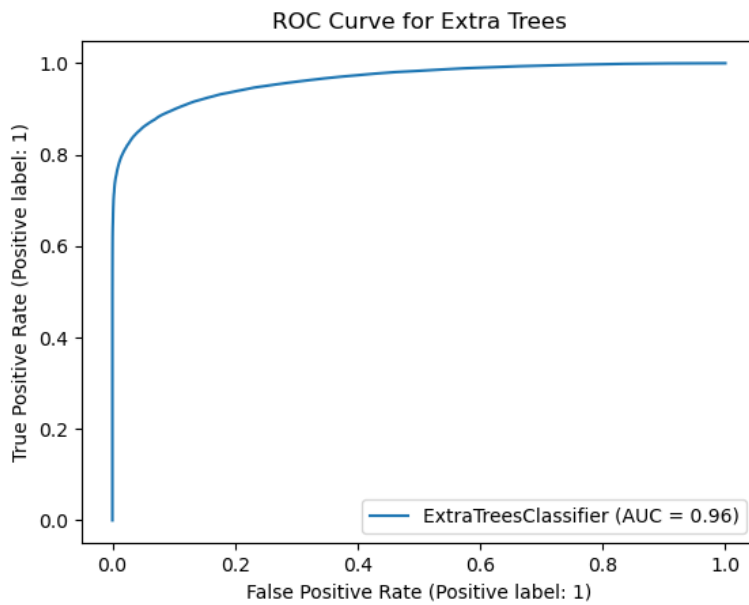

Figure 13: ROC curve for Adaboost Classifier

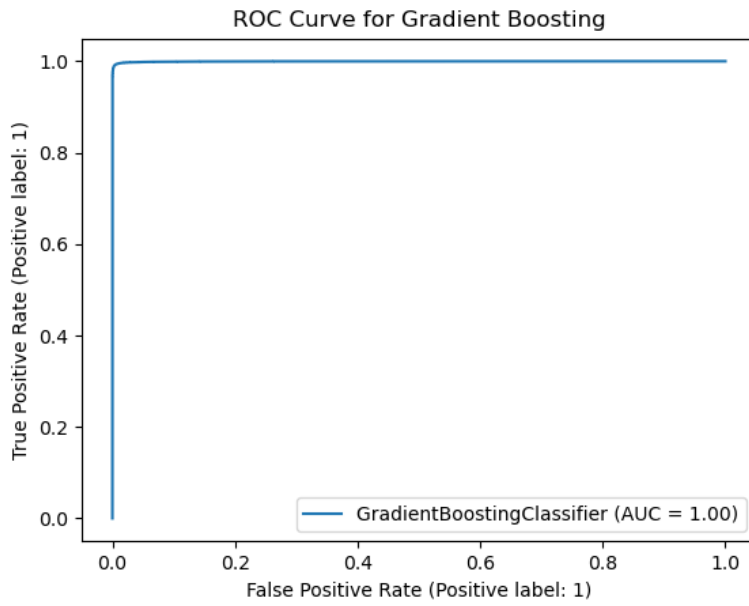Figure 14: ROC curve for extra trees classifier

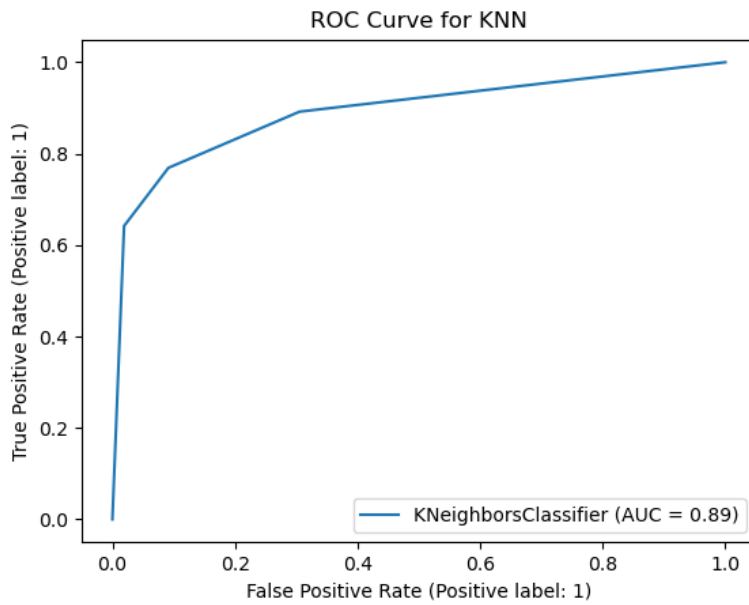Figure 15: ROC curve for gradient boosting classifier
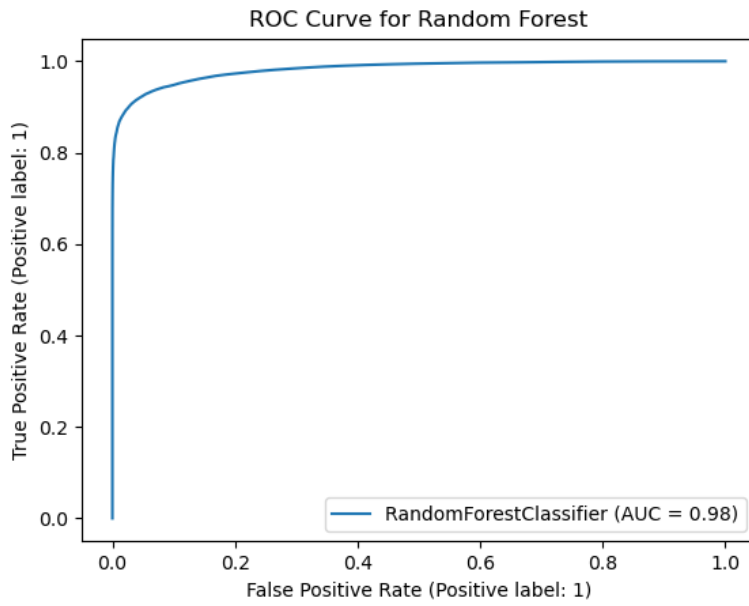

Figure 16: ROC curve for KNN classifier

Figure 17: ROC curve for Random Forest classifier

The gradient boosting classifier outperformed the other algorithms on all metrics, while achieving an area under the curve of nearly 1.00. Because of its excellent ROC Curve, f1-score, precision, and recall, this classifier was selected for use.

A streamlit application was developed to allow users to interact with the selected model. The application is hosted here. Streamlit offers a framework for developing user interfaces for machine learning models as well as a web hosting service. The hosing services requires developers to provide streamlit access to a github repository and a configuration file. Github allows users to upload files of size 50MB for free. This served as an additional constraint when deploying this service. The random forest classifier had a size of 0.5GB, and even after compression, it was far too large. Fortunately, the gradient boosting classifier required 0.12 MB of storage. The storage size of this classifier is sensitive to the maximum depth, and not to the number of classifiers. The user interface is presented in the next section:
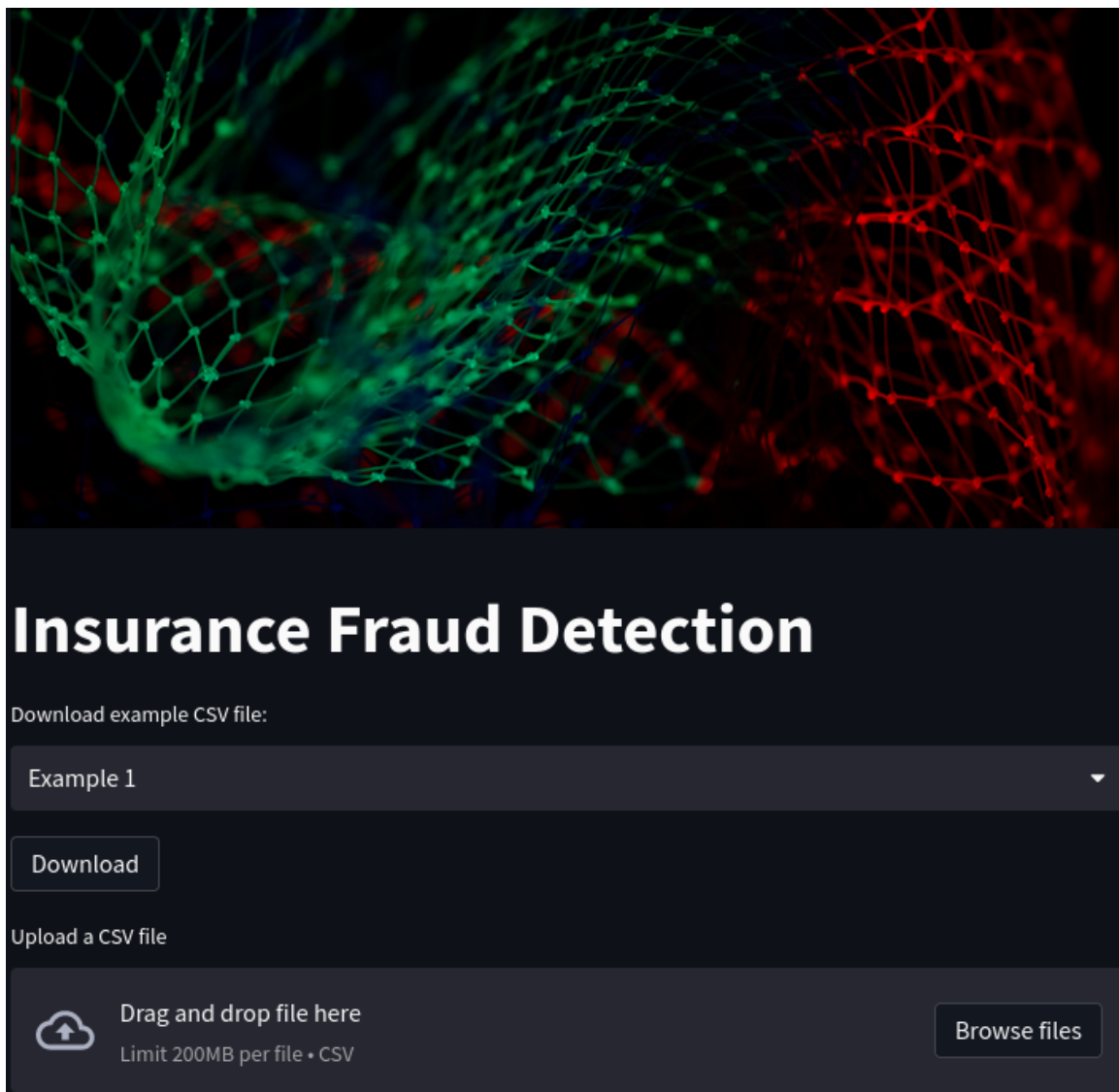

Figure 18: Graphical user interface for Streamlit service

The user can download example data from the application for testing purposes. The CSV can serve as a template for users to fill out with their own data. The upload section allows users to input up to 200MB of data to be classified. As seen in Figure 19, the
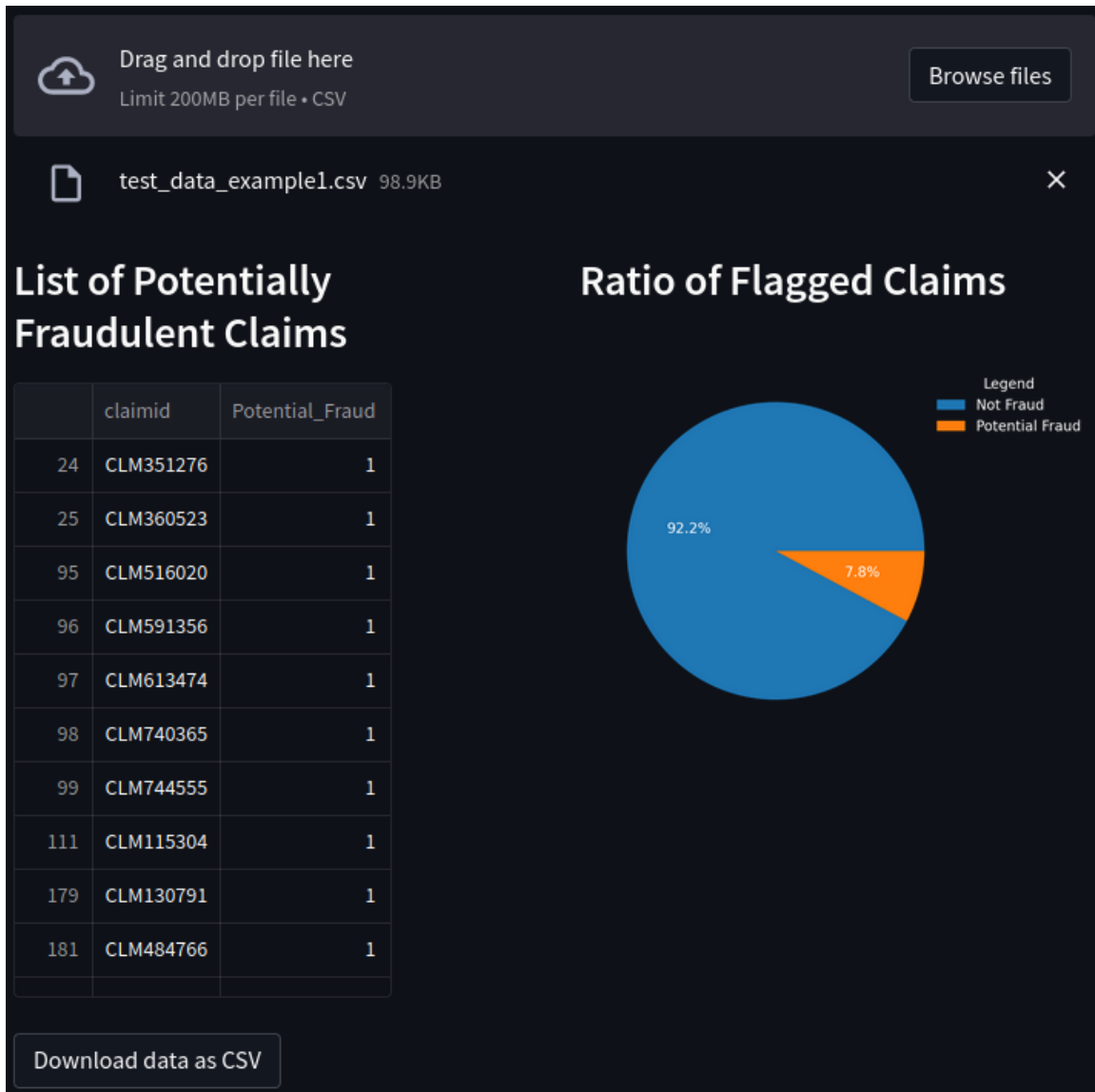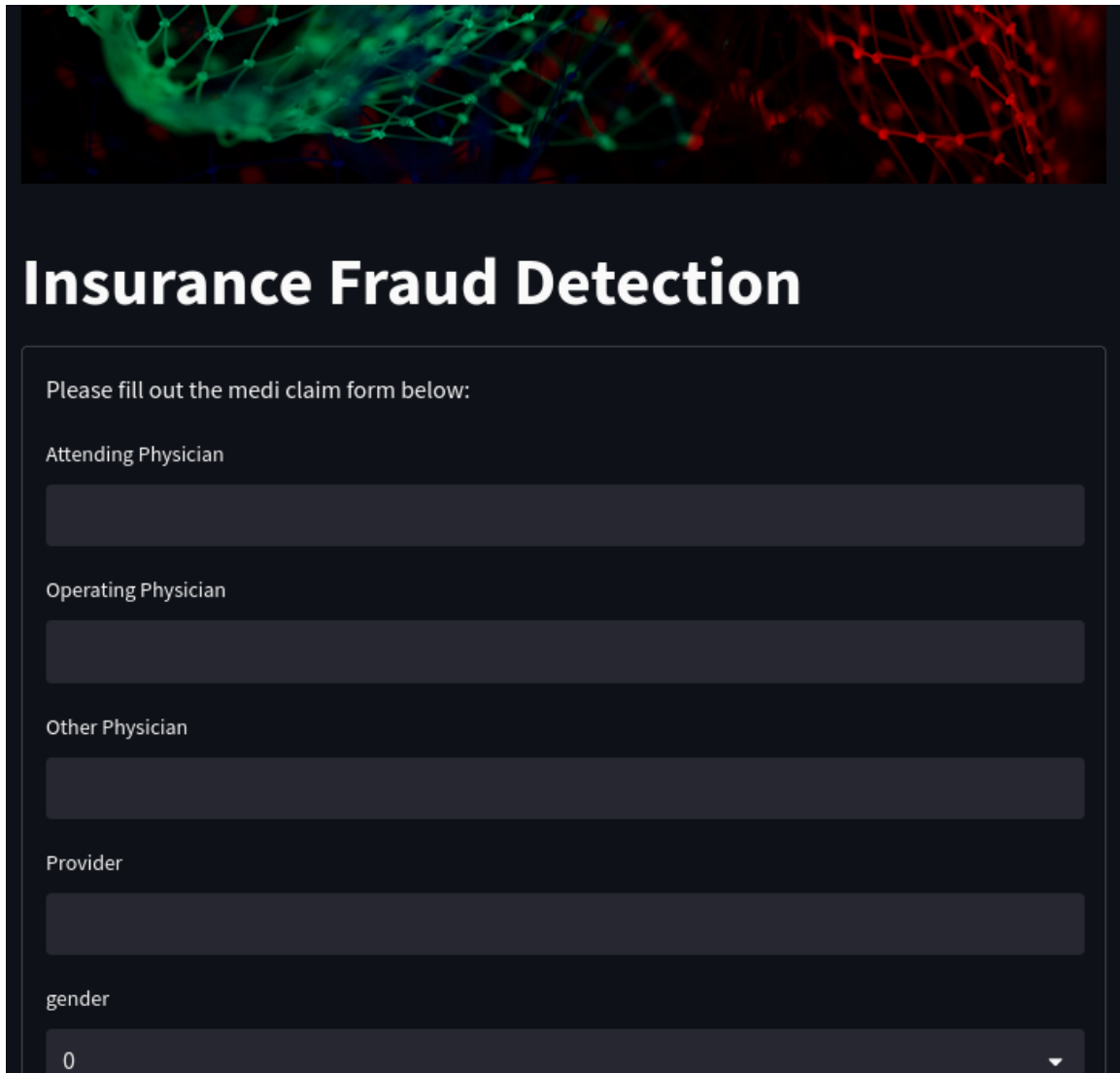

Figure 19: Example output

application will produce a list of claims that are potentially fraudulent. Users can view the table in the application, or download the data and view with any CSV compatible application.

Figure 20: Users can also elect to input a single data point in a fourm

7. Future Enhancements

As previously discussed, the model exhibits exceptional performance. However, there are certainly other models that could outperform the gradient boosted classifier. Additionally, final model was still not overfitted, and given more time, a better performance could likely be achieved with the same classifier.

8. References

1. fbi.gov
2. https://stackoverflow.com/questions/45003577/how-to-output-classification-report-of-sklearn-into-a-csv-file
3. https://seaborn.pydata.org/generated/seaborn.displot.html
4. https://www.kaggle.com/datasets/beenusharma42/fraudlent-claim-in-healthcare
5. https://scikit-learn.org/stable/modules/generated/sklearn.metrics.roc_curve.html
6. https://scikit-learn.org/stable/auto_examples/model_selection/plot_det.html#sphx-glr-auto-examples-model-selection-plot-det-py
7. https://seaborn.pydata.org/generated/seaborn.heatmap.html
8. https://seaborn.pydata.org/examples/horizontal_boxplot.html
9. https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestClassifier.html
10. https://scikit-learn.org/stable/modules/generated/sklearn.metrics.classification_report.html
11. https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.ExtraTreesClassifier.html
12. https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.AdaBoostClassifier.html
13. https://scikit-learn.org/stable/modules/generated/sklearn.neighbors.KNeighborsClassifier.html
14. "Introduction to Machine Learning with Python: A Guide for Data Scientists by Andreas Müller (Author), Sarah Guido (Author) "