Healthcare Fraud Detection and Analysis

Using CMS Inpatient, Outpatient and Beneficiary Data

Sneha Wattamwar [†]
Computer Science
George Mason University
Fairfax VA, USA
swattamw@gmu.edu

Devika Walavalkar
Computer Science
George Mason University
Fairfax VA, USA
dwalaval@gmu.edu

ABSTRACT

Healthcare fraud is a serious problem affecting every patient and consumer. It is estimated that the economic cost of fraud related to healthcare worldwide is 6% of the Global healthcare spending (Estimated $260 billion). Due to this, insurance companies increase their insurance premiums and as a result, healthcare is becoming a costly matter for the common man. Conventional methods of identifying fraudulent providers in health care are time consuming and ineffective. Data mining techniques may be utilized to help third parties such as Medicare organizations to extract useful information from hospital and claims data and identify any anomalies.

In this project we first classified providers in the CMS data as fraudulent or non-fraudulent providers using supervised and unsupervised approaches. Then we evaluated our model using the Kaggle Healthcare Data set. Secondly, we delivered pictorial representation of patterns, detections from the data, comparisons for exploratory data analysis.

INTRODUCTION

Medicare or Health Insurance fraud is a pressing issue and alludes to illegal practices aimed at getting unreasonably high payouts from healthcare programs funded by the government services. This causes substantial increase in the costs of medical insurance programs which affects the common man. The government always receives a large number of claims, and it's a difficult task to review each of the individual claims. We demonstrated the unsupervised anomalous outlier detection technique to detect anomaly on a subset of data from the CMS Medicare information on inpatient and outpatient discharges for beneficiaries' data. Once all data files are merged there is an opportunity to build a model and apply the fraud detection technique to make it more robust. This method then flags providers as fraudulent which can be further investigated by the government. Thus, the project proposal uses machine learning techniques to address the above problem and detect the fraudulent providers from the CMS open datasets.

OBJECTIVE

The objective of this project is to build a model that classifies fraudulent and non-fraudulent providers, thereby identifying the key features contributing to the fraud detection. From this, we then build a comprehensive robust machine learning model that detects fraud pattern based on those features. We evaluated different models by comparing it to Kaggle labelled healthcare fraud data set. By performing exploratory data analysis, we present data visualizations.

PURPOSE

Healthcare fraud is a serious problem affecting every patient and consumer. It is estimated that the economic cost of fraud related to healthcare worldwide is 6% of the Global healthcare. The conclusions from our proposed project can detect fraudulent healthcare providers and help to find patterns between such providers to avoid deceptions in healthcare services.

DATASET

Training data csv files

1. Inpatient Claims Data: The CMS Inpatient Claims contains institutional claims for hospital inpatient services provided to the Medicare beneficiaries in the CMS Beneficiary. Each record in an inpatient file pertains to an inpatient claim. Data size (66774 x 81), File size (16.7 MB)
2. Outpatient Claims Data: The CMS Outpatient Claims contains institutional claims for outpatient services provided to the Medicare beneficiaries in the CMS Beneficiary. Each record in an outpatient file pertains to an outpatient claim. Data size (790791 x 76), File size (161.8 MB)
3. Beneficiary Summary Data: The CMS Beneficiary Summary contains information on demographics, health plan enrollment, chronic conditions, and reimbursement based on a sample of Medicare beneficiaries. Each record represents a Medicare beneficiary. Data size (116353 x 32), File size (14.6 MB)

Testing data csv files

Testing data also has three .csv files that are similar to the training data files.

The details of the data source files are can be found at the bottom of the document.

DATA CLEANING

1. Type conversion of columns: The columns claim start date, claim end date, admission date, discharge date is converted to a date object.
2. Irrelevant columns: The columns HCPCS_CD_1 - HCPCS_CD_45 have codes for common procedure coding system. We removed these columns as they are irrelevant for our predictions.
3. Missing value columns: The columns that have NAN values are filled with 0.
4. Fix column names: All the column names in CMS data are renamed so as to have a readable name format.
5. Convert columns to categorical format: All the chronic condition columns have values 2 or 1. So we replaced value 2 with 0, this indicates the patient does not have the chronic condition. If it is 1 it indicates the patient is suffering from the chronic condition.
6. Create new columns: Column age is calculated using the birth date and the death date. Column whetherDead is created which has the value 1 if the patient is dead, or 0 if patient is alive. Column AdmitForDays is calculated using the discharge date and admission date.
7. Merge columns: All columns from inpatient and outpatient data are merged based on the columns that were common in both. This data is then merged with beneficiary data based on the beneficiary id. This process is done for both training and testing data set.
8. Numerical columns: After the data files are cleaned, we compared the numerical columns of training and testing files. After that we dropped the columns that were not common in both and sent it for dimensionality reduction/ feature selection technique.

FEATURE SELECTION

The dataset used for training consists of CMS data and is formed by merging all the data files (inpatient.csv, outpatient.csv, beneficiary.csv). The first two column that store the beneficiary Id and Claim Id are system generated unique values. Therefore, they were removed as they did not have any significance in any kind of analysis.

We used recursive feature elimination technique to decide the features contributing for the model accuracy. It is a backward selection process of the features. In this technique, it begins by building a model on the entire data set of features and computes an importance score for each feature. The least important features are then removed one-by-one, and the model is re-built, and importance scores are re-computed. The optimal features subset is then utilized to train the final model.

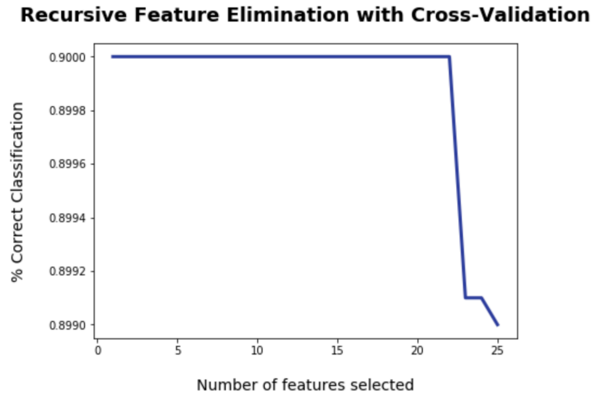
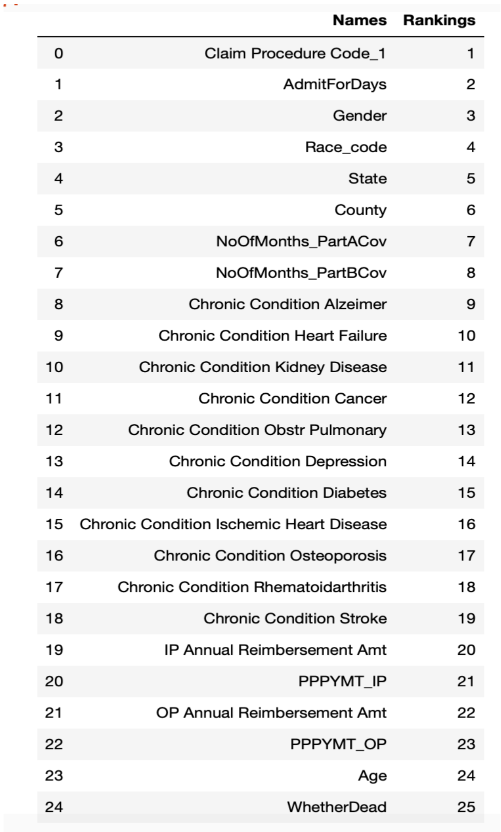


Figure 1: Numbers of features selected VS % Correct Classification

From the above figure we can see that the number of features that contribute toward the model accuracy of 0.90 is from the range 1-22.



	Names	Rankings
0	Claim Procedure Code_1	1
1	AdmitForDays	2
2	Gender	3
3	Race_code	4
4	State	5
5	County	6
6	NoOfMonths_PartACov	7
7	NoOfMonths_PartBCov	8
8	Chronic Condition Alzeimer	9
9	Chronic Condition Heart Failure	10
10	Chronic Condition Kidney Disease	11
11	Chronic Condition Cancer	12
12	Chronic Condition Obstr Pulmonary	13
13	Chronic Condition Depression	14
14	Chronic Condition Diabetes	15
15	Chronic Condition Ischemic Heart Disease	16
16	Chronic Condition Osteoporosis	17
17	Chronic Condition Rhematoidarthritis	18
18	Chronic Condition Stroke	19
19	IP Annual Reimbersement Amt	20
20	PPPYMT_IP	21
21	OP Annual Reimbersement Amt	22
22	PPPYMT_OP	23
23	Age	24
24	WhetherDead	25

Figure 2: Columns and ranking given by feature selection

From the above figure, we can see the list of features and their rankings given by the feature elimination technique.

FEATURE REDUCTION

The features that contribute towards the model accuracy were calculated using the feature elimination technique. In this process, the accuracy remained constant for features from 1-22. So, we tried two feature reduction techniques, PCA and Truncated SVD followed by readings for n_features = 22.

PCA finds a new set of dimensions such that all the dimensions are linearly independent and ranked according to the variance of data along them. It means more important principle axis occurs first as more importance implies more spread of data. Thus, only relevant features that affect the training of the model are taken into consideration.

Truncated SVD performs linear dimensionality reduction by means of truncated singular value decomposition (SVD). Compared to PCA, it first computes singular value decomposition and then computes the center of the data points. This signifies that it can work efficiently with sparse matrices to give better results.

We used both the dimensionality reduction technique on our data. However, we saw that PCA gave better results than truncated SVD.

OVERSAMPLING

In the graph below we can observe that the number of potential fraudulent records is insignificant compared to the non-fraudulent record.

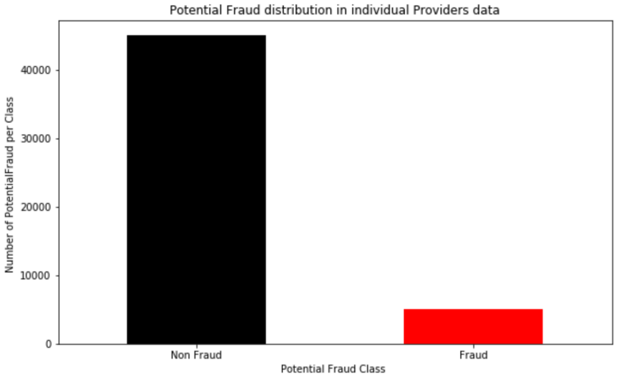
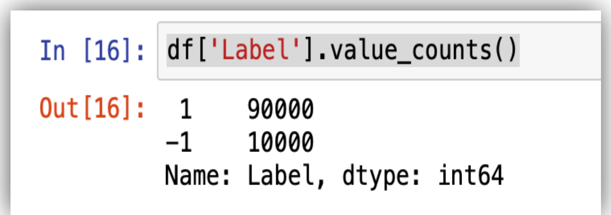


Figure 3: Potential fraud cases VS Number of potential fraud per class

This data is imbalanced as the instances of one class outnumbers the other by a large proportion. So, it is crucial to balance this data as feeding imbalanced data to the classifier can make it biased towards the majority class. The reason is not having enough data to learn from the minority class. We used SMOTE for handling this issue.

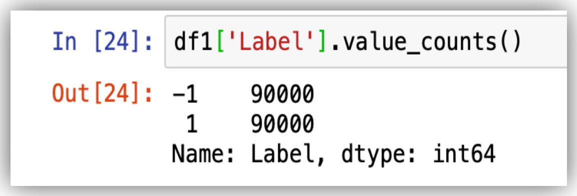
Below screenshot displays the behavior of y_train dataset for 10,000 instances. In the first one it shows that '1' is the majority class and '-1' is the minority class. This is resolved using SMOTE which is an oversampling technique. In the second screenshot it generates synthetic data for the minority class so that the count of both classes is same.



```
In [16]: df['Label'].value_counts()

Out[16]:  1    90000
         -1    10000
         Name: Label, dtype: int64
```

Figure 4: Label data value counts before oversampling



```
In [24]: df1['Label'].value_counts()

Out[24]: -1    90000
          1    90000
         Name: Label, dtype: int64
```

Figure 5: Label data value counts after oversampling

PREDICTION MODEL FOR TRAINING DATA SET USING CLUSTERING

We have used the unsupervised clustering for labelling the training data set. For this, we have used the Local Outlier Factor (LOF) anomaly detection algorithm.

The Local Outlier Factor (LOF) algorithm is an unsupervised anomaly detection method that computes the local density deviation of a given data point with respect to all its neighbors. This calculation compares the neighbors of a given data point to find out the density and compare it to the density of neighboring points. In short, we can say that the density around an outlier object is significantly different from the density around its neighbors.

Using this technique, we were able to find outliers in the dataset. Then, we labelled the outliers as -1 which indicates potential fraudulent records. We labelled the rest of the records as 1 which indicates non-fraudulent records.

PREDICTION MODEL FOR TESTING DATA SET USING CLASSIFICATION

The labelled training data set is then used for training different classifiers. After that we use the Kaggle data set as our testing data for which we predict labels. Then by using the F1-score as over evaluation metric we compare the predicted class with its true class. This true class labels are available for the Kaggle data set.

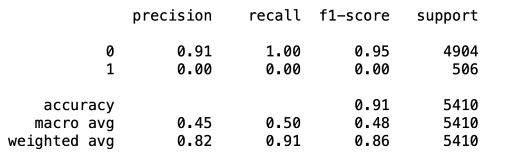
Following is a set of classifiers used to classify the test data records into fraudulent and non-fraudulent classes. We used dimensionality reduction / feature selection techniques such as PCA, Truncated SVD and Standard Scalar. As we can observe that although every classifier performs satisfactorily except for SVC, Logistic CV gives the best results with PCA. The evaluation metric used for this is the F1-score.

Classifiers	PCA	TruncatedSVD	Standard Scalar
Decision Tree	83.66	64.70	85.44
Logistic Regression	86.19	1.5	84.75
KNN	86.19	86.19	83.12
Random Forest	81.01	17.41	83.82
Gradient Boosting	78.43	86.12	86.19
Gausian Naïve Bayes	86.19	86.19	86.17
MLP Classifier	86.19	1.8	86.18
Logistic Regression CV	90.64	1.5	86.19
Linear Regression	85.98	85.98	86.19
Adaboost	84.70	86.18	84.05

Figure 6: Readings of classifiers with different dimensionality reduction techniques

Logistic RegressionCV

A statistical method for analyzing a dataset that contains one or more independent variables which determine the result. The effect is calculated against a dichotomous variable. Using Response as the dependent variable and the factors from stepwise regression as the independent variables, Logistic regression was performed on the given dataset. The model gave an accuracy of 90.64%. The acronym CV stands for cross-validation. We passed the number of folds as an argument cv of the function to perform k-fold cross-validation (StratifiedKFold=10). Following is the classification report of the model.



```
              precision    recall  f1-score   support

           0       0.91      1.00      0.95      4904
           1       0.00      0.00      0.00       506

    accuracy                           0.91      5410
   macro avg       0.45      0.50      0.48      5410
weighted avg       0.82      0.91      0.86      5410
```

Figure 7: Classification report for Logistic Regression CV

We tried to plot ROC curve for the logistic regressionCV model. In that, the method predict_proba predicts probability for each record which we then use to plot the ROC curve. However, our ground truth labels are for each provider institution and not for each record. So, due to this inequality it is not possible to plot the curve.

EXPLORATORY ANALYSIS

We have performed exploratory data analysis on features that are relevant and contribute to the training of the classifier. These features are obtained using the Recursive Feature Elimination technique as mentioned previously.

Frequency of procedures in the potential fraudulent records

The below graph provides a pictorial representation of top 20 most frequent procedure codes that are related to fraudulent cases. This gives us an idea of procedure codes which could contribute towards fraudulent events.
There is a total of 6 procedure code columns in the dataset.

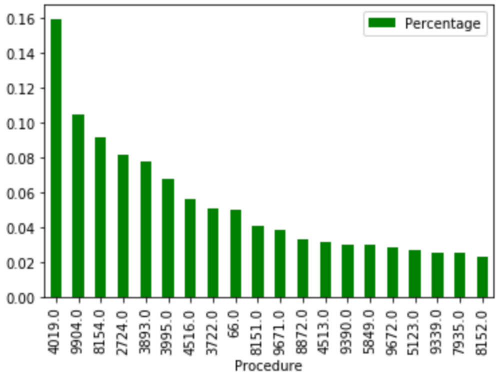


Figure 8: Frequency of procedures in the potential fraudulent records

Frequency of diagnosis group code in the potential fraudulent records:

The below graph provides a pictorial representation of top 20 most frequent diagnosis group codes that are related to fraudulent cases. Similar to Graph 1, it gives us the diagnosis group code where the probability of fraud is greater compared to remaining ones. There is a total of 10 diagnosis group code columns in the dataset.

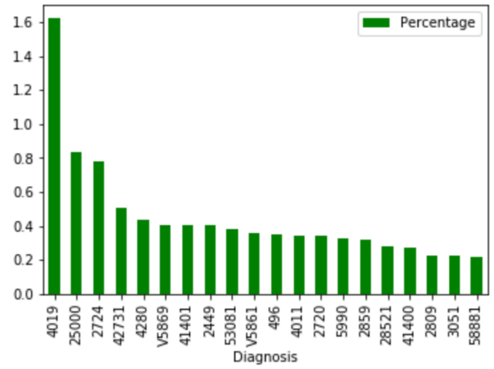
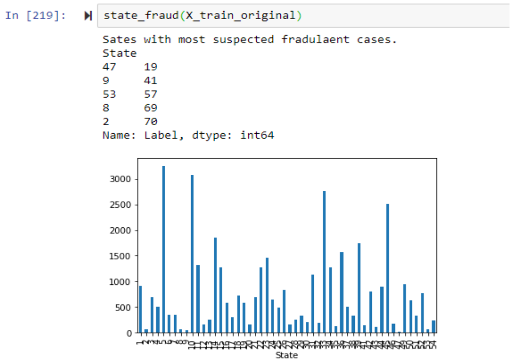


Figure 9: Frequency of diagnosis group code in the potential fraudulent records

States with most suspected fraudulent cases

The following graph plots the number of suspected fraudulent cases in every state. As we can observe, states 47, 9 and 53 have the 3 largest number of such cases. Further, we explore its relation with the diagnosis group code in the next graph.



```
In [219]:    state_fraud(X_train_original)

             Sates with most suspected fradulaent cases.
             State
             47    19
             9     41
             53    57
             8     69
             2     70
             Name: Label, dtype: int64
```

Figure 10: States with most suspected fraudulent cases

Highest frequency of diagnosis code in the state shortlisted above

It is found that the 3 top states have common diagnosis group code. We have listed 3 most frequent diagnosis group code in these states, and we can observe that the code 25000 is common. We can infer that the probability of this diagnosis group code contributing to fraudulent events is increased. Similarly, we can find top 10 or 20 such codes and analyze to find a pattern between the states and the occurring fraud.

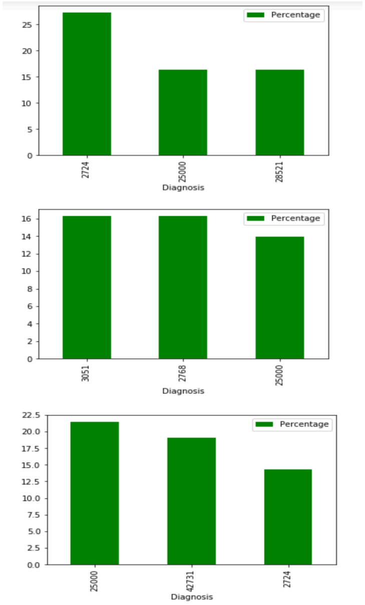


Figure 11: Highest frequency of diagnosis code in the state shortlisted above

Ratio of fraudulent and non-fraudulent records in each state

As observed, the ratio of the percentages of the two classes is high for every state. We can conclude that due to this imbalance, it is difficult to distinguish the two classes for every record. Thus, we require smote to balance the data.

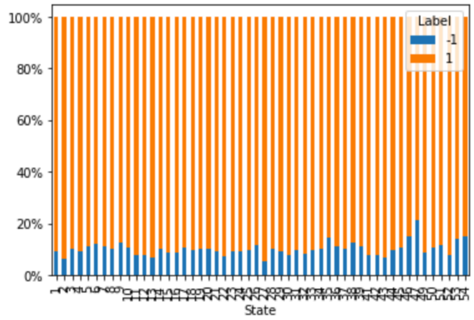


Figure 12: Ratio of fraudulent and non-fraudulent records in each state

Number of unique providers in every state

The following graph gives us the total number of provider institutions in every state. However, an important fact to consider here is that the state with maximum number of unique providers will have a large number of records in the training set mapped to that state. So, a state cannot be declared to be having the most number of potential fraudulent institutions by comparing it with the number of fraudulent/non-fraudulent records of the other states.

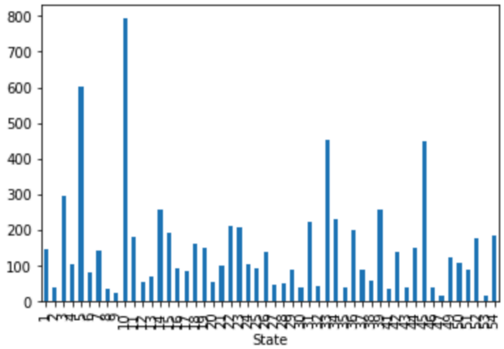


Figure 13: Number of unique providers in each state

Average inpatient claim amount for diagnosis group code

The following graph can be visualized as the total claim payment amount that is registered under each diagnosis group code. Inpatient claim amount is the amount charged to the patient admitted in the institution and provided with the institution services. We calculate the average for normalization purposes by diving the total claim amount by the total number of day s charged for. The 2nd graph shows the diagnosis group codes in the decreasing order of their total average claim payments.

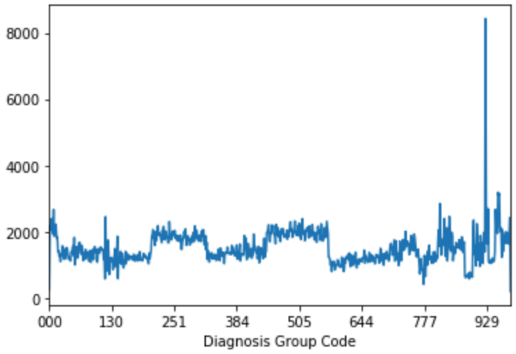


Figure 14: Average inpatient claim amount for diagnosis group code

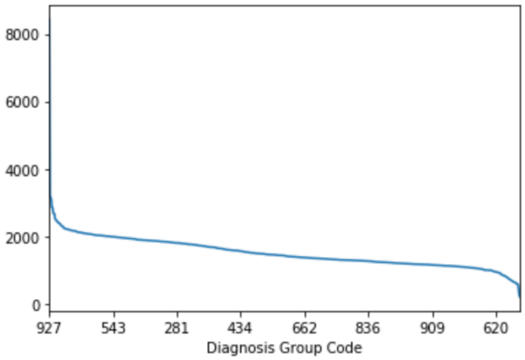


Figure 15: Average inpatient claim amount for diagnosis group code

Annual Reimbursement amount for every Provider institution

Following graph gives and estimate of the total annual reimbursement made to the provider institution for its services. The amount is gathered from the inpatient as well as the outpatient claim records. It can be inferred as the annual income of the provider institution on an abstract level basis. It includes all the procedures and treatments delivered by the institution.

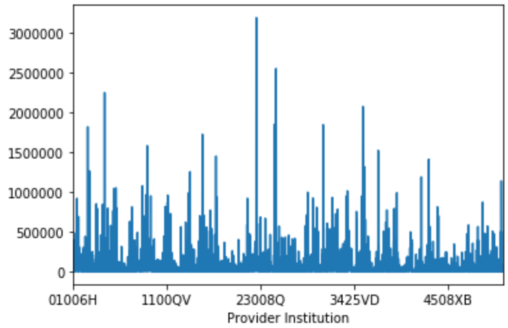
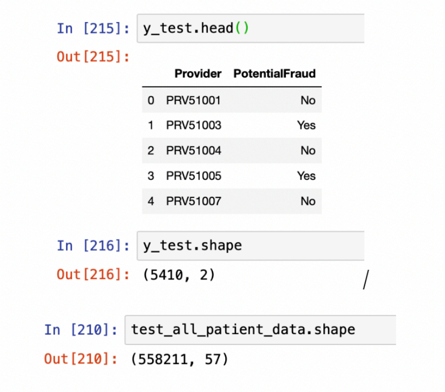


Figure 16: Annual Reimbursement amount for every Provider institution

Number of records vs the number of unique provider institutions

As we can see, the number of unique providers institutions which are to be classified is 5410. However, the test data contains a total of 558211 records i.e. every record has a unique beneficiary id mapped to its provider. Hence, we can infer that the providers are repeated in the test data which is quite obvious as multiple patients can be mapped to a single provider institution. The total number of predictions are 558211. We further map these predictions using Groupby() for every unique provider and label them accordingly as Fraud/Non-fraud.



```
In [215]: y_test.head()
Out[215]:
             Provider   PotentialFraud
          0  PRV51001               No
          1  PRV51003              Yes
          2  PRV51004               No
          3  PRV51005              Yes
          4  PRV51007               No

In [216]: y_test.shape
Out[216]: (5410, 2)

In [210]: test_all_patient_data.shape
Out[210]: (558211, 57)
```

Figure 17: Number of records vs the number of unique provider institutions

CONCLUSION

After performing the classification and analysis in the proposed project, we can conclude that a provider can be labeled as a potential fraudulent one only after considering multiple factors. Every observation needs a valid reasoning and outcome. Numerical values such as claims, duration, medical codes, and categorical values such as state, race, gender, county contribute majorly towards the decision-making process. Patterns is a key outcome of the project which can help in understanding the fraud domain and improving the analysis to concentrate on the potential fraudulent providers and potential regions where such events can occur.
By helping in the detection of healthcare fraud, we can make a better society together.

REFERENCES

[1]Kaggle :
https://www.kaggle.com/rohitrox/healthcare-provider-fraud-detection-analysis
https://www.kaggle.com/speedoheck/inpatient-hospital-charges

[2]CMS.gov (Centers for Medicare and Medicaid Services) :
https://www.cms.gov/Research-Statistics-Data-and-Systems/Downloadable-Public-Use-Files/SynPUFs/DESample01

[3]References :
https://journalofbigdata.springeropen.com/articles/10.1186/s40537-018-0138-3#Sec3
https://blog.dataiku.com/medicare-fraud

INPATIENT DATA

S.No.	Data Field	Data Type	Definition	Example
1	DESYNPUF_ID	Continuous	The beneficiary code	0016F745862898F
2	CLM_ID	Continuous	The claim id	196201177000368
3	SEGMENT	Numerical	The claim line segment	1
4	CLM_FROM_DT	Date	The start date of the claims	2009-04-12
5	CLM_THRU_DT	Date	The end date of the claims	2009-04-18
6	PRVDR_NUM	String	The provider institution number	3900MB
7	CLM_PMT_AMT	Numerical	The payment amount of the claim	26000
8	NCH_PRMRY_PYR_CLM_PD_AMT	Numerical	The primary payer amount paid for the claim	611998537
9	AT_PHYSN_NPI	Numerical	The national provider identifier name of the attending physician	611998537
10	OP_PHYSN_NPI	Numerical	The national provider identifier name of the operating physician	56274
11	OT_PHYSN_NPI	Numerical	The national provider identifier name of the other physician	1119000316
12	CLM_ADMSN_DT	Date	The admission date of the inpatient	2009-09-17
13	ADMTNG_ICD9_DGNS_CD	String	The admitting diagnosis code of the claim	V5789
14	CLM_PASS_THRU_PER_DIEM_AMT	Numerical	The per diem amount of the claim pass	50
15	NCH_BENE_IP_DDCTBL_AMT	Numerical	The beneficiary deductible amount of the inpatient	50
16	NCH_BENE_PTA_COINSRNC_LBLTY_AMT	Numerical	The beneficiary part A which is the coinsurance liability amount	100
17	NCH_BENE_BLOOD_DDCTBL_LBLTY_AMT	Numerical	The blood deductible amount of the beneficiary	200
18	CLM_UTLZTN_DAY_CNT	Numerical	The utilization day count of the claim	10
19	NCH_BENE_DSCHRG_DT	Date	The discharged date of inpatient	2009-09-120
20	CLM_DRG_CD	Numerical	The diagnosis related group code of the claim	45
21-30	ICD9_DGNS_CD_1 - ICD9_DGNS_CD_10	String	The claim diagnosis code from 1-10	V4501
31-36	ICD9_PRCDR_CD_1 - ICD9_PRCDR_CD_6	String	Claim procedure code 1-6	V5866
37-81	HCPCS_CD_1 - HCPCS_CD_45	String	Revenue center HCFA common procedure coding system 1-45	V5789

S.No.	Data Field	Data Type	Definition	Example
1	DESYNPUF_ID	Continuous	The beneficiary code	0016F745862898F
2	BENE_BIRTH_DT	Date	The beneficiary data of birth	196201177000368
3	BENE_DEATH_DT	Date	The beneficiary date of death	1
4	BENE_SEX_IDENT_CD	Category	The beneficiary sex	2009-04-12
5	BENE_RACE_CD	Category	The beneficiary race code	2009-04-18
6	BENE_ESRD_IND	String	The beneficiary end stage renal disease indicator	3900MB
7	SP_STATE_CODE	Numerical	The beneficiary state code	26000
8	BENE_COUNTY_CD	Numerical	The beneficiary county code	611998537
9	BENE_HI_CVRAGE_TOT_MONS	Numerical	The beneficiary total number of months for part A coverage	611998537
10	BENE_SMI_CVRAGE_TOT_MONS	Numerical	The beneficiary total number of months for part B coverage	56274
11	BENE_HMO_CVRAGE_TOT_MONS	Numerical	The beneficiary total number of months for HMO coverage	1119000316
12	PLAN_CVRG_MOS_NUM	Numerical	The beneficiary total number of months for part D coverage	2009-09-17
13	SP_ALZHDMTA	Numerical	The beneficiary chronic condition of alzheimer	1
14	SP_CHF	Numerical	The beneficiary chronic condition of heart failure	2
15	SP_CHRNKIDN	Numerical	The beneficiary chronic condition of kidney disease	1
16	SP_CNCR	Numerical	The beneficiary chronic condition of cancer	2
17	SP_COPD	Numerical	The beneficiary chronic condition of obstructive pulmonary	1
18	SP_DEPRESSN	Numerical	The beneficiary chronic condition of depression	2
19	SP_DIABETES	Numerical	The beneficiary chronic condition of diabetes	2
20	SP_ISCHMCHT	Numerical	The beneficiary chronic condition of ischemic heart disease	1
21	SP_OSTEOPRS	Numerical	The beneficiary chronic condition of osteoporosis	2
22	SP_RA_OA	Numerical	The beneficiary chronic condition of rheumatoid arthritis	1
23	SP_STRKETIA	Numerical	The beneficiary chronic condition of stroke ischemic attack	2
24	MEDREIMB_IP	Numerical	The inpatient annual Medicare reimbursement amount	80
25	BENRES_IP	Numerical	The inpatient annual beneficiary responsibility amount	200
26	PPYMT_IP	Numerical	The inpatient annual primary payer reimbursement amount	5000

OUTPATIENT DATA

S.No.	Data Field	Data Type	Definition	Example
1	DESYNPUF_ID	Continuous	The beneficiary code	0016F745862898F
2	CLM_ID	Continuous	The claim id	196201177000368
3	SEGMENT	Numerical	The claim line segment	1
4	CLM_FROM_DT	Date	The start date of the claims	2009-04-12
5	CLM_THRU_DT	Date	The end date of the claims	2009-04-18
6	PRVDR_NUM	String	The provider institution number	3900MB
7	CLM_PMT_AMT	Numerical	The payment amount of the claim	26000
8	NCH_PRMRY_PYR_CLM_PD_AMT	Numerical	The primary payer amount paid for the claim	611998537
9	AT_PHYSN_NPI	Numerical	The national provider identifier name of the attending physician	611998537
10	OP_PHYSN_NPI	Numerical	The national provider identifier name of the operating physician	56274
11	OT_PHYSN_NPI	Numerical	The national provider identifier name of the other physician	1119000316
12	NCH_BENE_BLOOD_DDCTBL_LBLTY_AMT	Numerical	The blood deductible amount of the beneficiary	8
13-22	ICD9_DGNS_CD_1 - ICD9_DGNS_CD_10	Categorical	The claim diagnosis code from 1-10	Agent
23-28	ICD9_PRCDR_CD_1 - ICD9_PRCDR_CD_6	Numerical	Claim procedure code 1-6	384.8111
29	NCH_BENE_PTB_DDCTBL_AMT	Numerical	NCH Beneficiary Part B Deductible Amount	50
30	NCH_BENE_PTB_COINSRNC_AMT	Numerical	NCH Beneficiary Part B Coinsurance Amount	200
31	ADMTNG_ICD9_DGNS_CD	String	Claim Admitting Diagnosis Code	V5883
32-76	HCPCS_CD_1 - HCPCS_CD_45	String	Revenue center HCFA common procedure coding system 1-45	V5677