Medicare Fraud Detection using Machine Learning Models

Rishabh Sareen

July 16, 2022

Abstract Medicare fraud is more prevalent among medical providers and usually results in higher health care costs, insurance premiums, and taxes for the general population. Medical Providers try to maximize reimbursement received from Medicare which they are not entitled to via illegitimate activities such as submitting false claims. This project will focus on fraud committed by doctors and hospitals. Using supervised machine learning on real-life Medicare claims data, key healthcare fraud indicators and fraudulent provider characteristics are identified which could be used in Medicare fraud investigation.

Introduction

Healthcare is a major industry in the U.S. with both private and government run programs. The costs of healthcare continue to rise, in part due to the increasing population of the elderly. U.S. healthcare spending from 2012 to 2014 has increased by 6.7% to reach $3 trillion and Medicare spending accounts for 20% of all health-care spending in the U.S. at about $600 billion. The impact of healthcare fraud is estimated to be between 3% to 10% of the nation's total healthcare spending and continuing to adversely impact the Medicare program and its beneficiaries.

Healthcare fraud is an organized crime that involves peers of providers, physicians, beneficiaries acting together to make fraud claims. Rigorous analysis of Medicare data has yielded many physicians who indulge in fraud. They adopt ways in which an ambiguous diagnosis code is used to adopt the costliest procedures and drugs. Insurance companies are the most vulnerable institutions impacted due to these bad practices.

Due to this reason, insurance companies increased their insurance premiums and as a result, healthcare is becoming a costly matter day by day. Healthcare fraud and abuse take many forms. Some of the most common types of frauds by providers are billing for services that were not provided, duplicate submission of a claim for the same service, misrepresenting the service provided, charging for a more complex or expensive service than was provided, billing for a covered service when the service provided was not covered.

This project aims to 'predict the potentially fraudulent providers' based on the claims filed by them. Along with this, discovering important features will be helpful in detecting the behavior of potentially fraudulent providers. Thus, from the standpoint of machine learning this is a binary classification problem to determine if the provider has committed some kind of fraud or not, using the claim details submitted by that provider.

Data

The Medicare datasets used in this project are publicly available and sourced from the Centers for Medicare and Medicaid Services (CMS) and FDA/openFDA. It consists of a total of eight CSV files which are further divided into four train data CSV and four test data CSV viz.

1. Beneficiary dataset (train/test)
2. Inpatient dataset (train/test)
3. Outpatient dataset (train/test)
4. label (train/test)

1. Beneficiary data (train/test)
This data contains beneficiary KYC details like health conditions, the region they belong to etc. Some of the categorical features included here are: BeneID(the unique identification number given to a patient), DOD (date of death of the patient),Gender, Race, State, of the patient along with indicators such as RenalDiseaseIndicator, ChronicCondition. It also includes the total

reimbursement amount claim for OPD, OPAnnualReimbursementAmt and the amount paid by the patient annually for OPD, OPAnnualDeductibleAmt.

2. Inpatient data (train/test)
This data provides insights into the claims filed for those patients who are admitted to the hospitals. It also provides additional details like their admission and discharge dates and admits diagnosis codes etc. Some of the variables are ClaimID, InscClaimAmtReimbursed, AttendingPhysician, ClmAdmitDiagnosisCode, Admission date etc.

3. Outpatient data (train/test)
This data provides details about the claims filed for those patients who visit hospitals and are not admitted to them. It is similar to Inpatient data except that it has few more variables like AdmissionDate, DischargeDate and DiagnosisGroupCode.

4. Label data (train/test)
It consists of two columns for train data which are provider id and class labels (whether fraud or not) for each provider respectively. The test dataset has only the column of provider ids.

```
In [10]: # checking train_beneficiary information
         train_beneficiary.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 138556 entries, 0 to 138555
Data columns (total 25 columns):
 #   Column                          Non-Null Count   Dtype
---  ------                          --------------   -----
 0   BeneID                          138556 non-null  object
 1   DOB                             138556 non-null  object
 2   DOD                             1421 non-null    object
 3   Gender                          138556 non-null  int64
 4   Race                            138556 non-null  int64
 5   RenalDiseaseIndicator           138556 non-null  object
 6   State                           138556 non-null  int64
 7   County                          138556 non-null  int64
 8   NoOfMonths_PartACov             138556 non-null  int64
 9   NoOfMonths_PartBCov             138556 non-null  int64
 10  ChronicCond_Alzheimer           138556 non-null  int64
 11  ChronicCond_Heartfailure        138556 non-null  int64
 12  ChronicCond_KidneyDisease       138556 non-null  int64
 13  ChronicCond_Cancer              138556 non-null  int64
 14  ChronicCond_ObstrPulmonary      138556 non-null  int64
 15  ChronicCond_Depression          138556 non-null  int64
 16  ChronicCond_Diabetes            138556 non-null  int64
 17  ChronicCond_IschemicHeart       138556 non-null  int64
 18  ChronicCond_Osteoporasis        138556 non-null  int64
 19  ChronicCond_rheumatoidarthritis 138556 non-null  int64
 20  ChronicCond_stroke              138556 non-null  int64
 21  IPAnnualReimbursementAmt        138556 non-null  int64
 22  IPAnnualDeductibleAmt           138556 non-null  int64
 23  OPAnnualReimbursementAmt        138556 non-null  int64
 24  OPAnnualDeductibleAmt           138556 non-null  int64
dtypes: int64(21), object(4)
```

Methods

In this section, the focus is on cleaning and preprocessing of the data so that machine learning models can be successfully applied later.

1. **Preprocessing of Data**
   As the first step, the variables of all the four datasets are listed to check for any discrepancy and certain changes are made for the easy application of models.

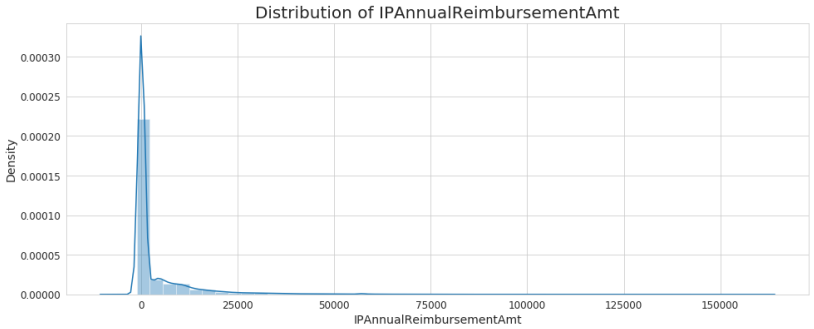
   a. Beneficiary Dataset

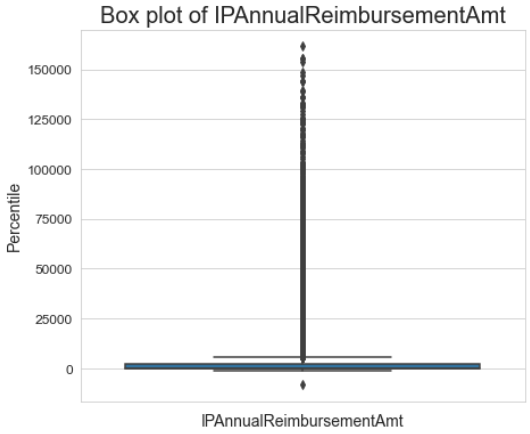
```
In [10]: # checking train_beneficiary information
         train_beneficiary.info()

         <class 'pandas.core.frame.DataFrame'>
         RangeIndex: 138556 entries, 0 to 138555
         Data columns (total 25 columns):
          #   Column                          Non-Null Count   Dtype
         ---  ------                          --------------   -----
          0   BeneID                          138556 non-null  object
          1   DOB                             138556 non-null  object
          2   DOD                             1421 non-null    object
          3   Gender                          138556 non-null  int64
          4   Race                            138556 non-null  int64
          5   RenalDiseaseIndicator           138556 non-null  object
          6   State                           138556 non-null  int64
          7   County                          138556 non-null  int64
          8   NoOfMonths_PartACov             138556 non-null  int64
          9   NoOfMonths_PartBCov             138556 non-null  int64
          10  ChronicCond_Alzheimer           138556 non-null  int64
          11  ChronicCond_Heartfailure        138556 non-null  int64
          12  ChronicCond_KidneyDisease       138556 non-null  int64
          13  ChronicCond_Cancer              138556 non-null  int64
          14  ChronicCond_ObstrPulmonary      138556 non-null  int64
          15  ChronicCond_Depression          138556 non-null  int64
          16  ChronicCond_Diabetes            138556 non-null  int64
          17  ChronicCond_IschemicHeart       138556 non-null  int64
          18  ChronicCond_Osteoporasis        138556 non-null  int64
          19  ChronicCond_rheumatoidarthritis 138556 non-null  int64
          20  ChronicCond_stroke              138556 non-null  int64
          21  IPAnnualReimbursementAmt        138556 non-null  int64
          22  IPAnnualDeductibleAmt           138556 non-null  int64
          23  OPAnnualReimbursementAmt        138556 non-null  int64
          24  OPAnnualDeductibleAmt           138556 non-null  int64
         dtypes: int64(21), object(4)
```

   Observations and Modifications
   - Only data of death column has Nan values
   - Change gender to 1 and 0 from 1 and 2.
   - In RenalDiseaseIndicator replaced Y with 1.
   - If beneficiary doesn't have a chronic condition the datatype is set to be 0.

b. Inpatient Dataset

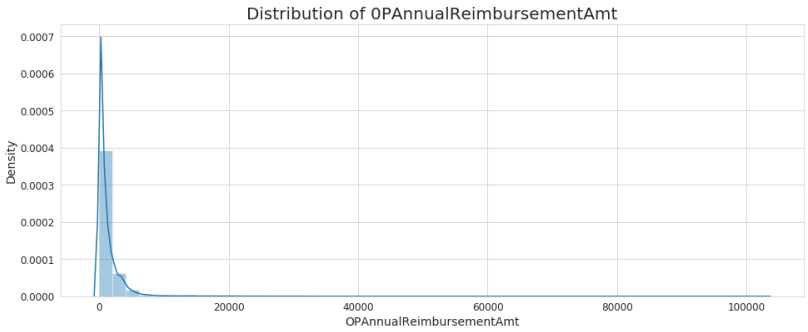


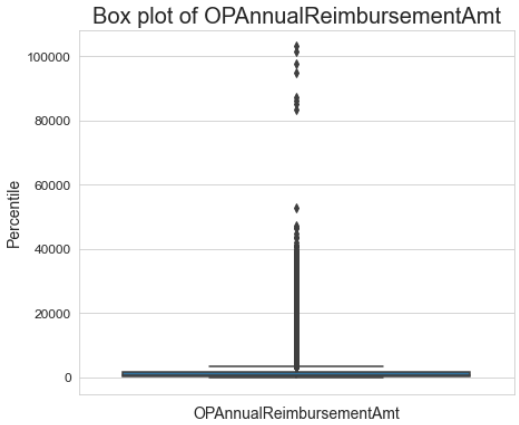


Observations and Modifications

- The range on annual Reimberusemt of inpatient data is between 0 to 150000.
- Most of the patients got zero 'IPAnnualReimbursementAmt' to 25000.
- Very few got reimbursement between 25000 to 1500000.
- The amount of annualReimbersuemt is very high.
- It is following the Gaussian distribution with some skewness.
- The difference between the 99th percentile and 100th percentile is very big.
- Around 80% of beneficiaries got reimbursement less than equal to 5000.

c. Outpatient Dataset





Observations and Modifications

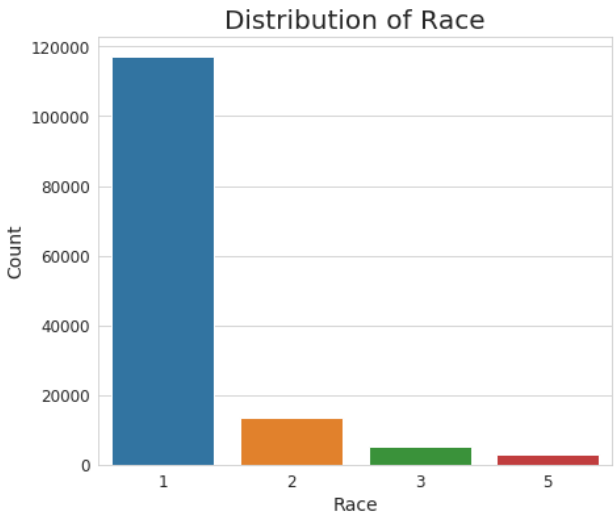
- Maximum amount reimbursement is 100000.
- Most of the beneficiaries got reimbursement less than 10000.
- The max reimbursement amount is 102960
- 75 % of the beneficiary got 1500 or less
- We can see that 100 percentile is very large than 99.9 percentile. It may be an outlier.
- It is following the Gaussian distribution with some skewness.

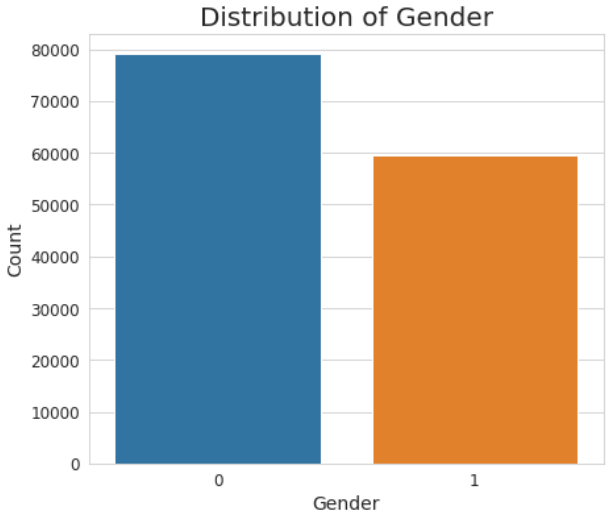
Observations and Modifications

- Class labels in the label data set are highly imbalanced there is 9% of potential fraud and 90.647% is non-fraud labels.
- There are no missing values in the PotentialFraud column (class label).

2. Exploratory Data Analysis:

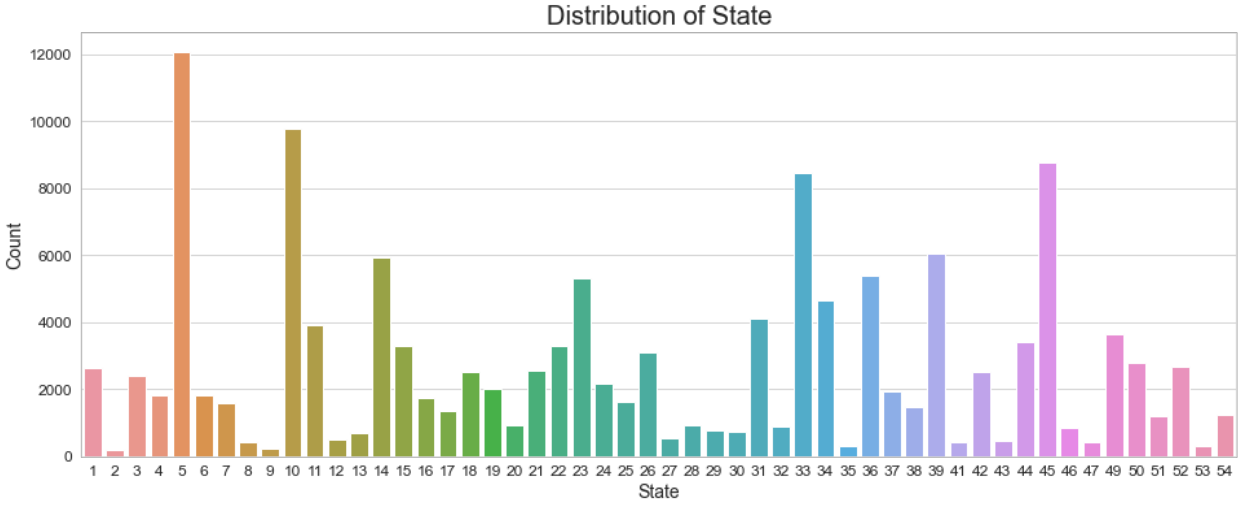
To get a better insight into the data, different graphs were plotted to understand the distribution of patients according to race, gender and state.





Observations

- Maximum of beneficiary belongs to race 1 followed by race 2.
- Beneficiary belongs to race 5 are very less.
- Gender_0 is 57.09% and gender_1 is 42.02%
- The gender is balanced in the beneficiary data set.

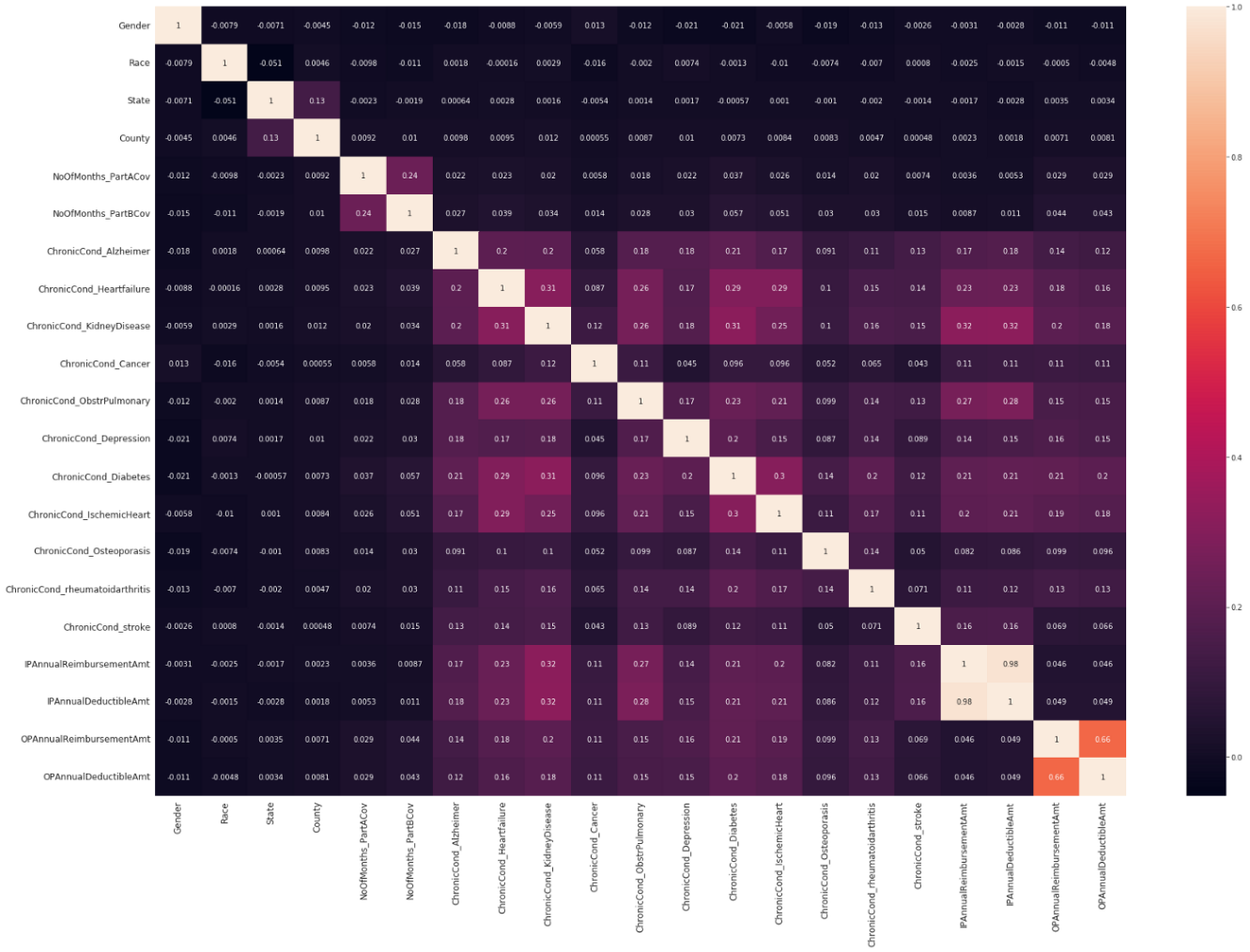


Observations

- Maximum beneficiary comes from state code 5.
- Very few beneficiaries come from state code 2.

3. Feature Selection

In this section, the categorical variables of inpatient data set are plotted against that of the outpatient data set to determine the correlation amongst the variables. This is done to drop the highly correlated variables so that the computational time to run the models is less without affecting the accuracy of the models.



Observations

- IPAnnualDeductibleAmt is collinear with IPAnnualReimbursementAmt with 0.97.
- OPAnnualReimbursementAmt is collinear with OPAnnualDeductibleAmt with 0.66.

Thus, one of the features IPAnnualDeductibleAmt or IPAnnualReimbursementAmt can be dropped.

4. Deploying Machine Learning Models

Now that the datasets are well equipped, the following ML models can be applied to predict the fraudulent Medicare transactions.

1. Logistic Regression
2. Random Forest
3. Support Vector Machine Algorithm

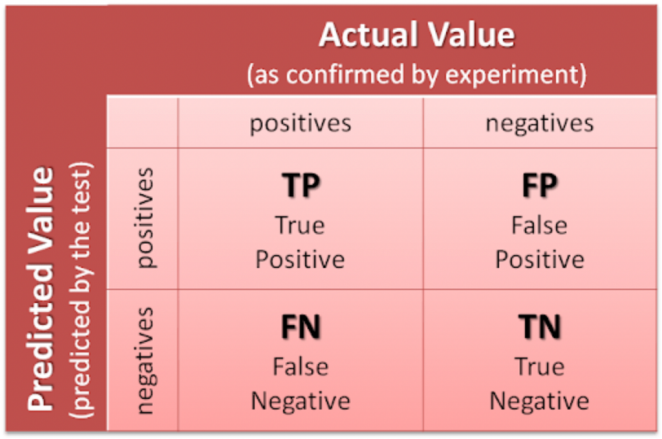
The first step would be to train these classification models using all the feature vectors. Next, these classification models would be retrained using only the important features.

Before moving to these models, the metrics to compute the efficiency of these models must be explained.

Performance Metric

To analyze the efficiency of a model, confusion matrix, F1score and AUC score are used.

Confusion matrix:
It is a table used to investigate the performance of a classification model where the actual test values are known. It has two rows and two columns describing the true positives, false positives, false negatives, and true negatives.



Precision and Recall

Precision measures how accurate is your predictions. i.e. the percentage of your predictions are correct while Recall measures how good you find all the positives. For example, we can find 80% of the possible positive cases in our top K predictions.

F1 score

It is a measure combining both precision and recall. It is generally described as the harmonic mean of the two. Harmonic mean is just another way to calculate an "average" of values, generally described as more suitable for ratios (such as precision and recall) than the traditional arithmetic mean. The formula used for F1-score in this case is:

$$Precision = \frac{TP}{TP + FP}$$

$$Recall = \frac{TP}{TP + FN}$$

$$F1 = 2 \cdot \frac{precision \cdot recall}{precision + recall}$$

TP = True positive

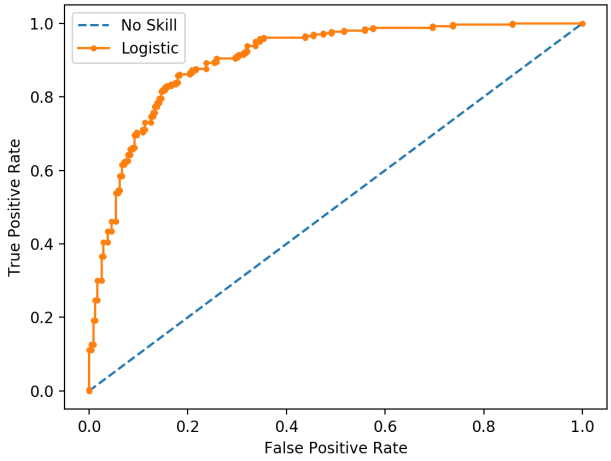
TN = True negative

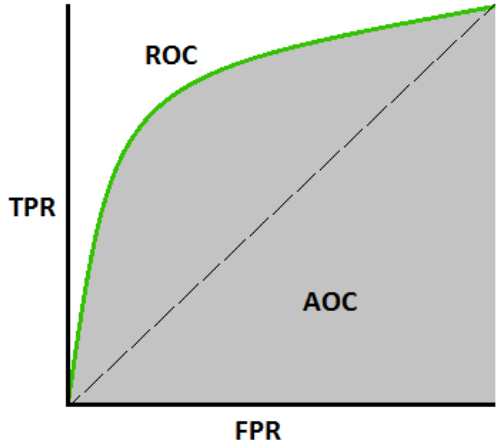
FP = False positive

FN = False negative

AUC score-

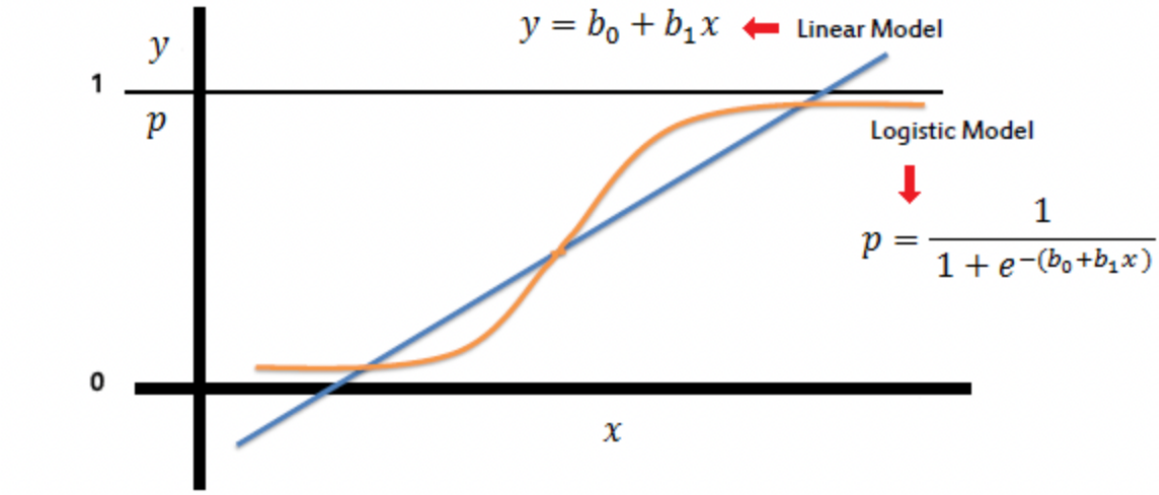
AUC also called an AREA UNDER CURVE. It is used in classification analysis in order to determine which of the used models predicts the classes best. An example of its application is ROC curves. AUC ranges in value from 0 to 1. A model whose predictions are 100% wrong has an AUC of 0 and if the predictions are 100% correct has an AUC of 1.



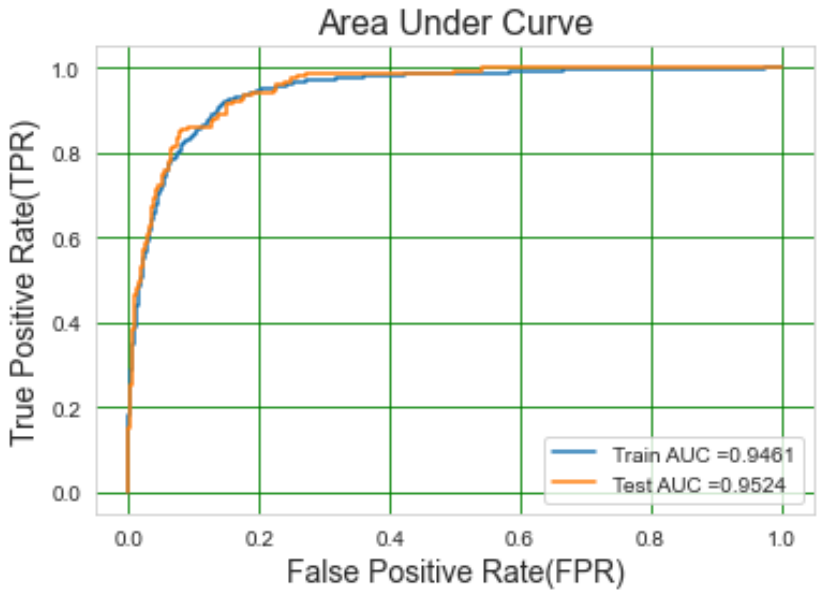


Logistic Regression

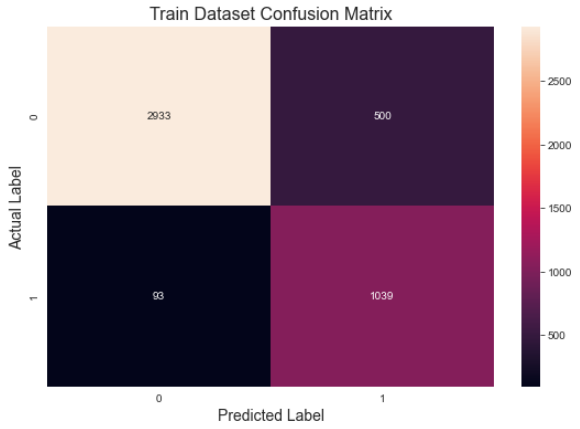
Logistic Regression is a Supervised statistical technique to find the probability of dependent variable. It uses functions called the logit functions, that helps derive a relationship between the dependent variable and independent variables by predicting the probabilities or chances of occurrence. The logistic functions convert the probabilities into binary values which could be further used for predictions.

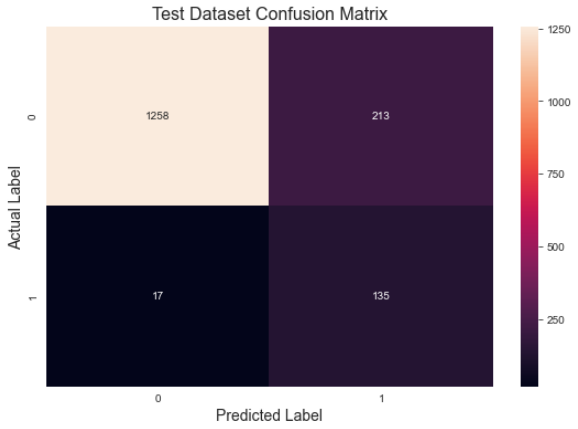


Accuracy of Logistic Regression Model



Train AUC = 0.9461076704074668
Test AUC = 0.9523730723818382

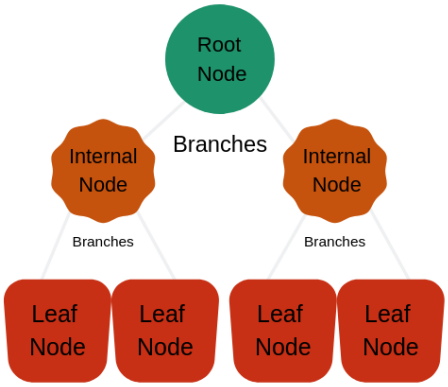


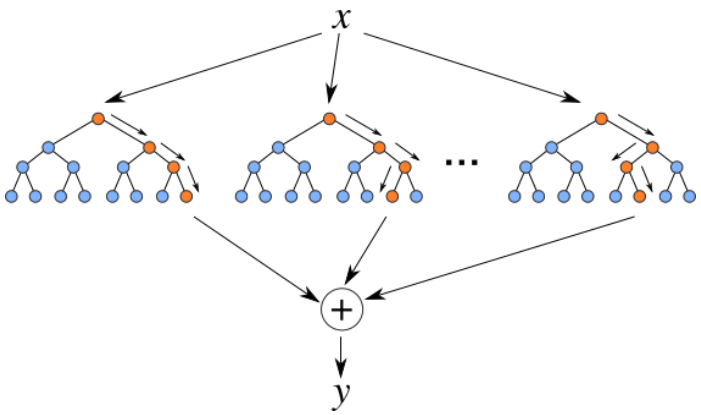


Best Threshold = 0.1441
Model AUC is : 0.9524
Model F1 Score is : 0.5400

Random Forest Algorithm

The random forest algorithm is a supervised learning model; it uses labeled data to "learn" how to classify unlabeled data. It is composed of different decision trees, each with the same nodes, but using different data that leads to different leaves. It merges the decisions of multiple decision trees in order to find an answer, which represents the average of all these decision trees.





When using the Random Forest Algorithm to solve regression problems, the mean squared error (MSE) is used to determine how data branches from each node.

$$MSE = \frac{1}{N} \sum_{i=1}^{N} (fi - yi)^2$$

Where N is the number of data points,
fi is the value returned by the model and
yi is the actual value for data point i.

This formula calculates the distance of each node from the predicted actual value, helping to decide which branch is the better decision for your forest. Here, yi is the value of the data point you are testing at a certain node and fi is the value returned by the decision tree.

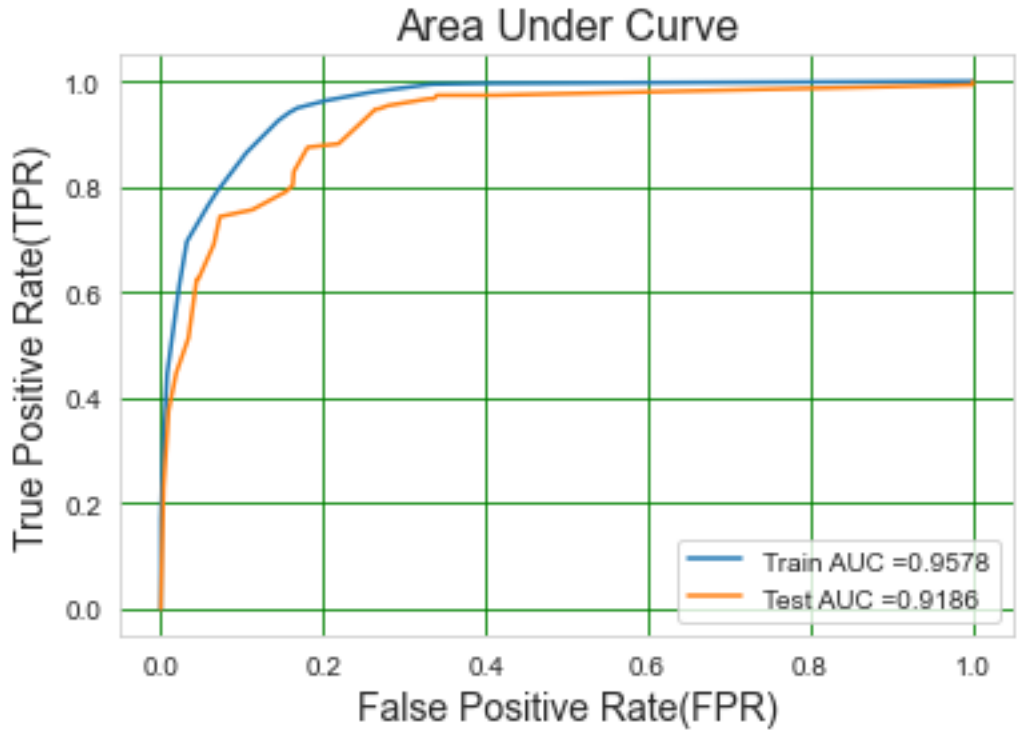
When performing Random Forests based on classification data, as is the case here, the following Gini index is applied to decide how nodes on a decision tree branch.

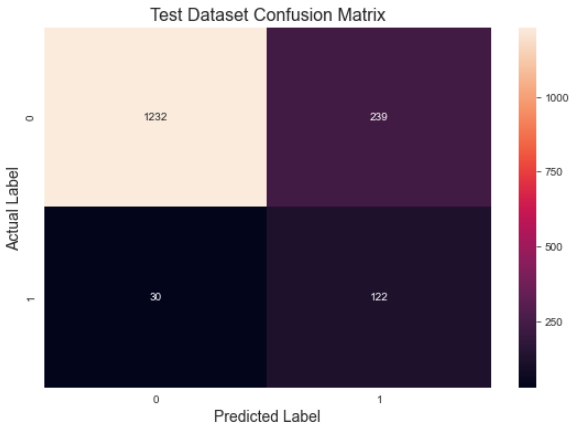
$$Gini = 1 - \sum_{i=1}^{C} (p_i)^2$$

This formula uses the class and probability to determine the Gini of each branch on a node, determining which of the branches is more likely to occur. Here, pi represents the relative frequency of the class observed in the dataset and c represents the number of classes.

Accuracy of Random Forest Model

Train AUC = 0.9577704806497731
Test AUC = 0.9185681956420622





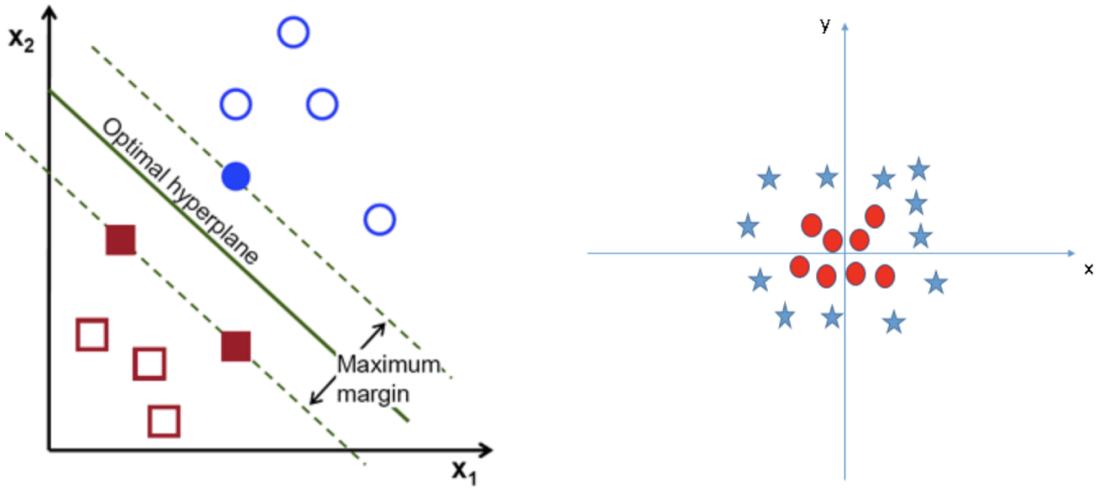
Best Threshold = 0.2727
Model AUC is : 0.9186
Model F1 Score is : 0.475

Support Vector Machine Algorithm

A Support Vector Machine (SVM) is a supervised machine learning algorithm that can be employed for both classification and regression purposes. SVMs are more commonly used in classification problems.

In this algorithm, each data item is marked as a point in n-dimensional space (where n is number of features you have) with the value of each feature being the value of a particular coordinate. Then, classification is performed by finding the optimal hyper-plane that differentiate the two classes very well.

Hyper-plane will be a point in case of 1 dimensional data, line in case of 2 dimensional data, plane in case of 3 dimensional data and so on.



Consider the following case where our goal is to find a hyperplane (line in this case) that will separate the two classes. Here, there does not exist a line that can separate the two classes. The SVM algorithm is implemented in practice using a kernel.

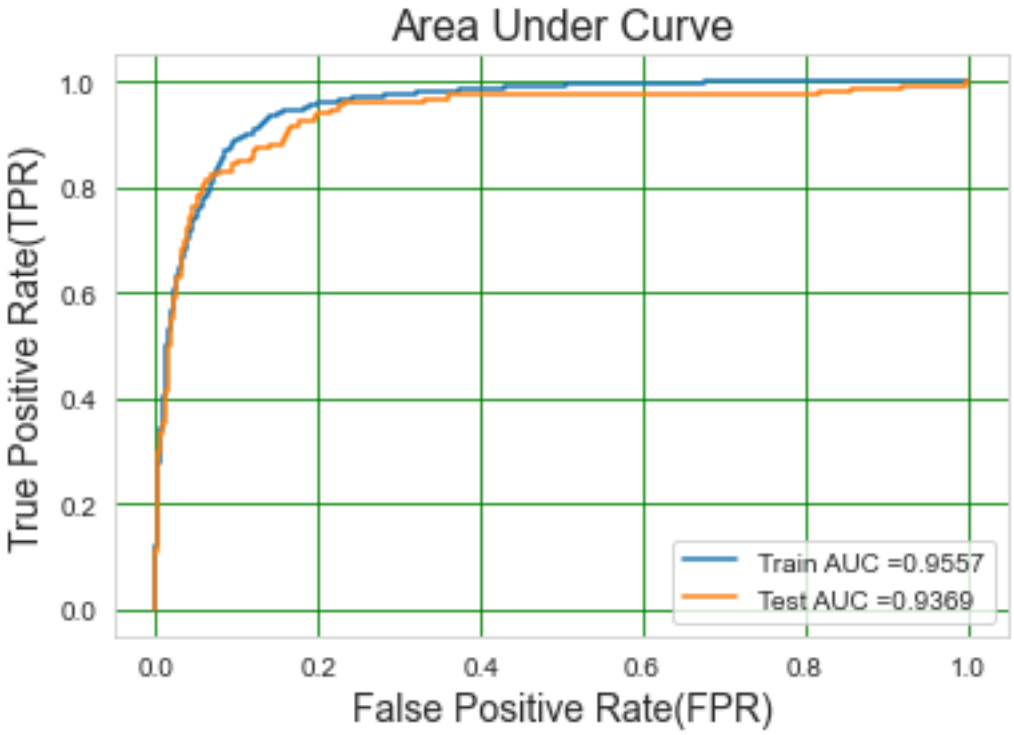
1. Polynomial kernel: It is popular in image processing.
   Equation is:

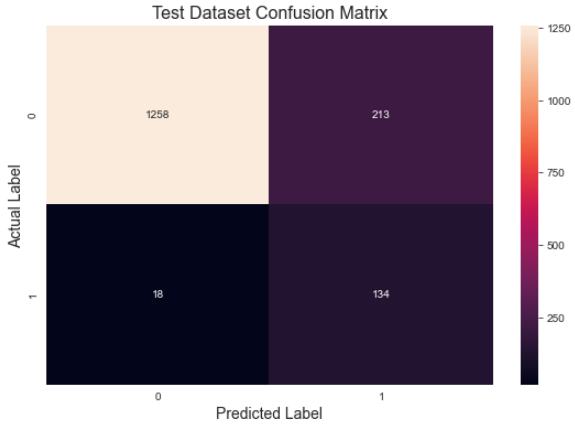
   $$k(\mathbf{x_i}, \mathbf{x_j}) = (\mathbf{x_i} \cdot \mathbf{x_j} + 1)^d$$ where d is the degree of the polynomial.

2. Gaussian kernel
   It is general-purpose kernel; used when there is no prior knowledge about the data.
   Equation is:

   $$k(x, y) = \exp\left(-\frac{\|x - y\|^2}{2\sigma^2}\right)$$

Accuracy of Support Vector Machine Algorithm





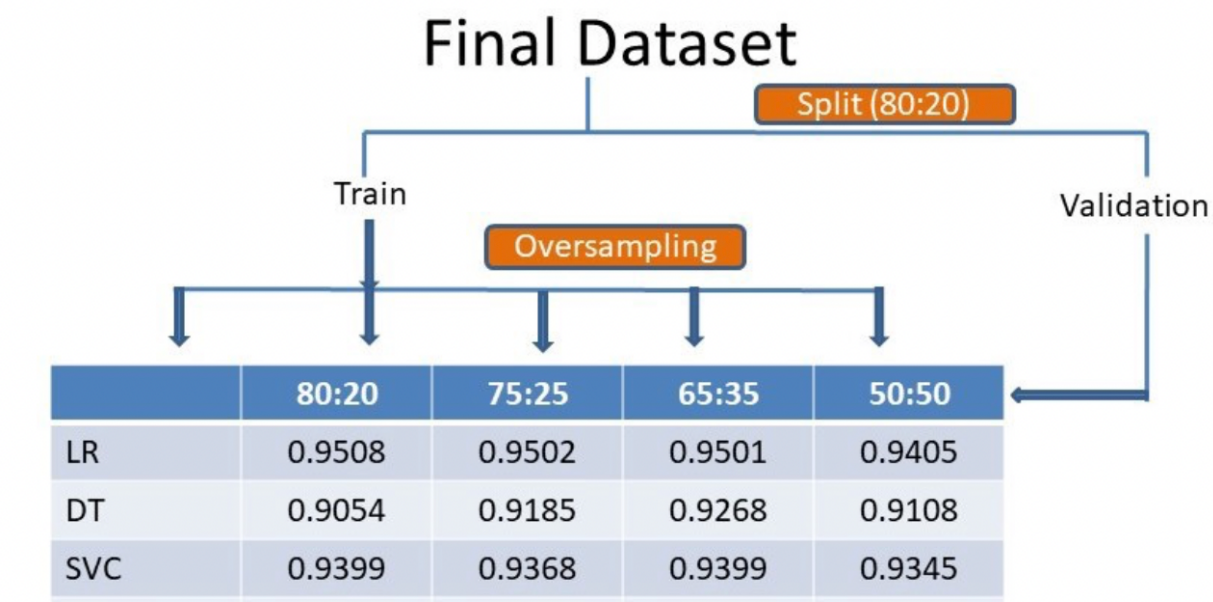
Best Threshold = 0.1561
Model AUC is : 0.9369
Model F1 Score is : 0.5371

Results

To determine the best model we follow the following steps:

Approach 1

1. Split the data into Train and Validation (80:20)
2. Oversample the data (majority: minority) to make 75:25 and 65:35 .
3. Use Logistic Regression, Random Forest(Decision Tree) and Support Vector Classifier, for all these 4 oversampled datasets.
4. Pick the best model based on the performance score.

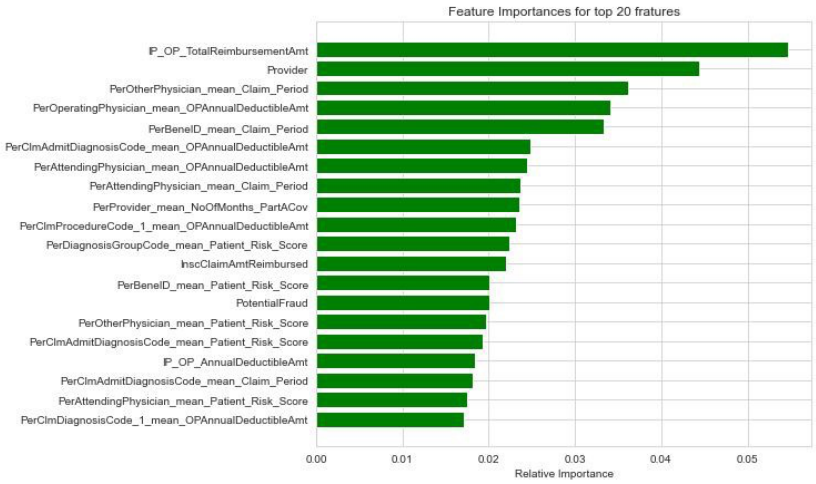


	80:20	75:25	65:35	50:50
LR	0.9508	0.9502	0.9501	0.9405
DT	0.9054	0.9185	0.9268	0.9108
SVC	0.9399	0.9368	0.9399	0.9345

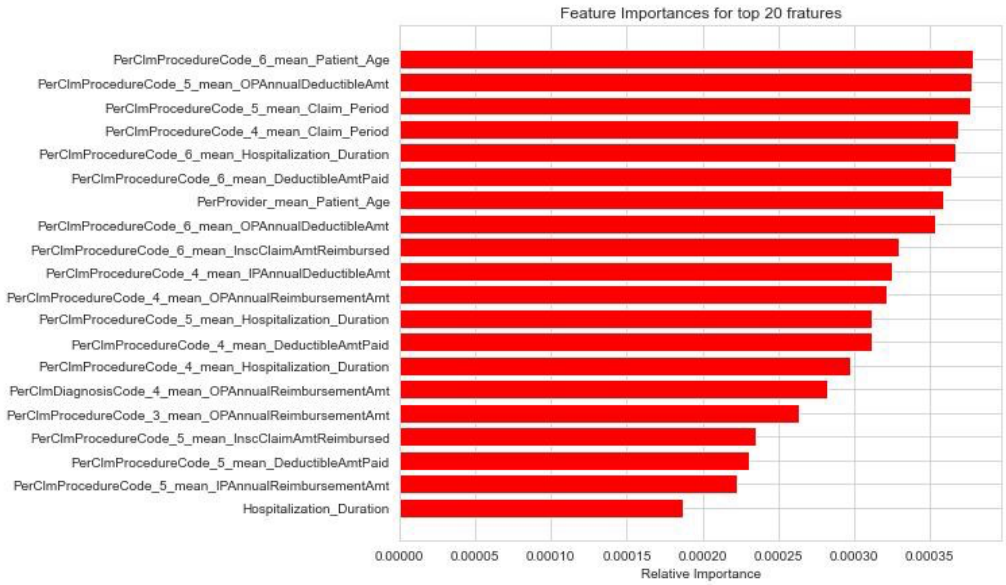
After evaluating the above models across all the oversampled data, it is found that Logistic Regression performed the best with Oversampling Ratio 80:20 with AUC score of 0.9508

Approach 2

In this approach, first feature importance was calculated using the Random Forest model. Based on the decrease of Gini impurity when a variable is chosen to split a node the feature importance is calculated.



20 top important features



20 least important features

When the features with feature importance greater than 0.001, are taken 161 are found and then the ML model using these features is trained.

Following are the observations:

Sampling Ratio	Model	Features	AUC	F1 Score
80:20	Logistic Reg	All Features	0.951	0.551
80:20	Logistic Reg	Important Features	0.942	0.56
80:20	Random Forest	All Features	0.943	0.62
80:20	Random Forest	Important Features	0.943	0.634

Results

1. Logistic Regression(LE) vs Random Forest (RF)t vs Support Vector Algorithm with all the features

Using RF, F1 score increased from LR model with little decrease in AUC. Apparently, it can be said the RF model performing better than LR model. But if I look at the confusion matrix, the False Negative(Predicted Not-Fraud but actually it is Fraud) count is more in RF, which is very dangerous in our case. After looking at all the scores it can be said that LR is performing better than RF.

2. After filtering the important features there is no such improvement in model performance for both LR and RF. F1 score is increased even though False-negative also increased. In our case decreasing False Negative is more important than decreasing False Positive. So, I can say the model is performing better with all features than only using top important features.

3. After considering AUC, F1 Score, FNR it can be said the Logistic Regression model is the best model in healthcare provider fraud detection problem.

Conclusions

Possibly Fraud Providers	**Non-Fraud Providers**
High average claim reimbursement amounts. Some of these providers have the highest reimbursement amounts in the dataset.	Low average claim reimbursement amounts
High average number of patient insurance claims.	Low average number of patient insurance claims.
A narrow range of patient age.	A wider range of patient age.
A narrow range of total patient chronic condition counts.	A wider range of total patient chronic condition counts.
Outpatient – high number of diagnosis codes listed on claims	Outpatient - low number of diagnosis codes listed on claims

- ❑ Beneficiaries having high reimbursement amounts or paying high deductibles also have more chronic conditions and could be more susceptible to fraud.
- ❑ A patient's age being in a certain range, and who their primary doctor is could in certain cases make them more vulnerable to fraud.