**Insurance Claims - Fraud Detection Project using Machine Learning Models**

**Introduction:**

Auto Insurance is one of widely used insurance types which provides insurance to customers for vehicle in terms of financials whenever there is an accident or any damage to vehicle or in theft cases.

One of the major challenges faced by Insurance Companies which provides Auto Insurance is dealing with Fraud insurance claims. So, predicting and identifying a fraud claims through machine learning model. Inflating claims, false claims are common insurance fraud occurring in Auto Insurance. Using Machine learning, we can effectively build a model which can predict and identify the fraud claims and help the insurance companies.

In this project, we use 7 US states Auto insurance claims data to build a model that can predict whether the applied claim is a fraudulent claim. Machine Learning is an effective approach model.

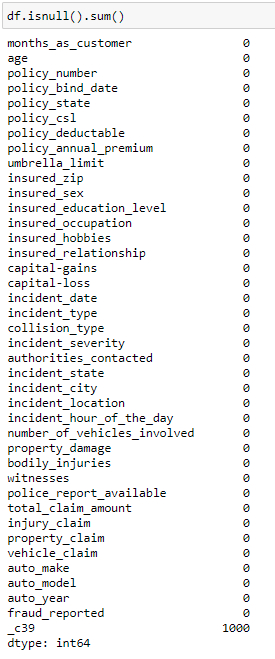
**Data Analysis:**

Dataset used for building the prediction model consists of customer details, insurance policy details, incident type, collision type to describe the claim core reasons. We also have insured personal details like sex, education level, occupation, hobbies and relationship to understand the insured. We have claim amount and injury claim details to provide further more details for predictive.

Dataset consists of 1000 rows with 40 columns. Below is the list of the columns name,

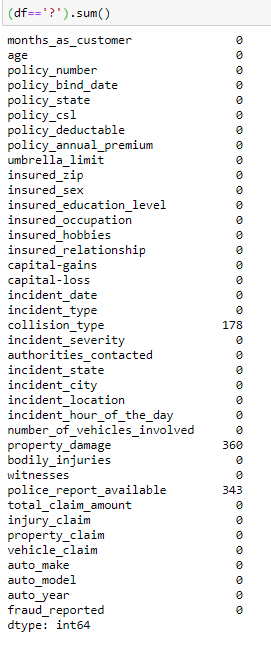
1. months\_as\_customer
2. age
3. policy\_number
4. policy\_bind\_date
5. policy\_state
6. policy\_csl
7. policy\_deductable
8. policy\_annual\_premium
9. umbrella\_limit
10. insured\_zip
11. insured\_sex
12. insured\_education\_level
13. insured\_occupation
14. insured\_hobbies
15. insured\_relationship
16. capital-gains
17. capital-loss
18. incident\_date
19. incident\_type
20. collision\_type
21. incident\_severity
22. authorities\_contacted
23. incident\_state
24. incident\_city
25. incident\_location
26. incident\_hour\_of\_the\_day
27. number\_of\_vehicles\_involved
28. property\_damage
29. bodily\_injuries
30. witnesses
31. police\_report\_available
32. total\_claim\_amount
33. injury\_claim
34. property\_claim
35. vehicle\_claim
36. auto\_make
37. auto\_model
38. auto\_year
39. fraud\_reported
40. \_c39

Below is the snapshot of the same,



Upon observation, found columns like policy\_annual\_premium, policy\_bind\_date, insured\_zip columns have 991, 951 and 995 unique values which is greater than 95% of total data so for a better predictive model, these columns have been dropped for better accuracy

Also, found some invalid values (Null values or missing values) in the dataset. These null values were in the form as “?”. Below is the snapshot of the same,



There are '?' data in 3 columns

1. collision\_type - 178

2. property\_damage - 360

3. police\_report\_available - 343

All these 3 columns are categorical data so we need to fill these '?' data with mode values.

We can also observe that last column ‘\_c39’ has 1000 null values which is 100% unique values so for better efficiency model, dropping this model is an optimal solution.

Most of the Columns Datatype are Object, below are the observations we can come to conclusion from the above snapshots.

**Categorical Data Columns (Object Type):**

Out of 39 Columns, 21 Columns has Categorical Data

1. policy\_bind\_date

2. policy\_state

3. policy\_csl

4. insured\_sex

5. insured\_education\_level

6. insured\_occupation

7. insured\_hobbies

8. insured\_relationship

9. incident\_date

10. incident\_type

11. collision\_type

12. incident\_severity

13. authorities\_contacted

14. incident\_state

15. incident\_city

16. incident\_location

17. property\_damage

18. police\_report\_available

19. auto\_make

20. auto\_model

21. fraud\_reported

**Continuous Data Columns (Int and Float Type):**

Out of 39 Columns, 18 Columns has Continuous Data

1. months\_as\_customer

2. age

3. policy\_number

4. policy\_deductable

5. policy\_annual\_premium

6. umbrella\_limit

7. insured\_zip

8. capital-gains

9. capital-loss

10. incident\_hour\_of\_the\_day

11. number\_of\_vehicles\_involved

12. bodily\_injuries

13. witnesses

14. total\_claim\_amount

15. injury\_claim

16. property\_claim

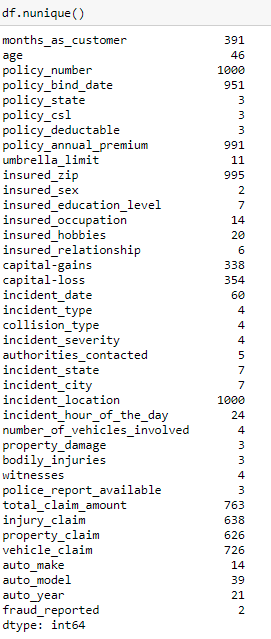
17. vehicle\_claim

18. auto\_year

Target Variable is fraud\_reported

Since the target variable 'fraud\_reported' data type is categorical, so need to approach this data set as Classification Problem

We can conclude the Unique Value Analysis as below,



Below are the observations we can make out from above unique value column data list,

1. policy\_number, incident\_location has 1000 unique values so it would be efficient to drop these columns for better model performance.

2. policy\_state, policy\_csl, policy\_deductable, umbrella\_limit, insured\_sex, insured\_education\_level, insured\_occupation, insured\_hobbies, insured\_relationship, incident\_type, collision\_type, incident\_severity, authorities\_contacted, incident\_state, incident\_city, number\_of\_vehicles\_involved, property\_damage, bodily\_injuries, witnesses, police\_report\_available, auto\_make, and auto\_year has reasonable unique data’s in the entire set; So, need to analyse these columns more with respect to relationship with fraud\_reported.

3. policy\_annual\_premium, policy\_bind\_date, insured\_zip have 991, 951 and 995 unique values which is greater than 95% so we can drop these columns for better accuracy

**Exploratory Data Analysis (EDA):**

**Target Column (Dependent variable) Analysis:**

Target Variable 'fraud\_reported' has 2 unique values

1. N: which is equivalent to No Fraud Reported

2. Y: which is equivalent to Fraud Reported

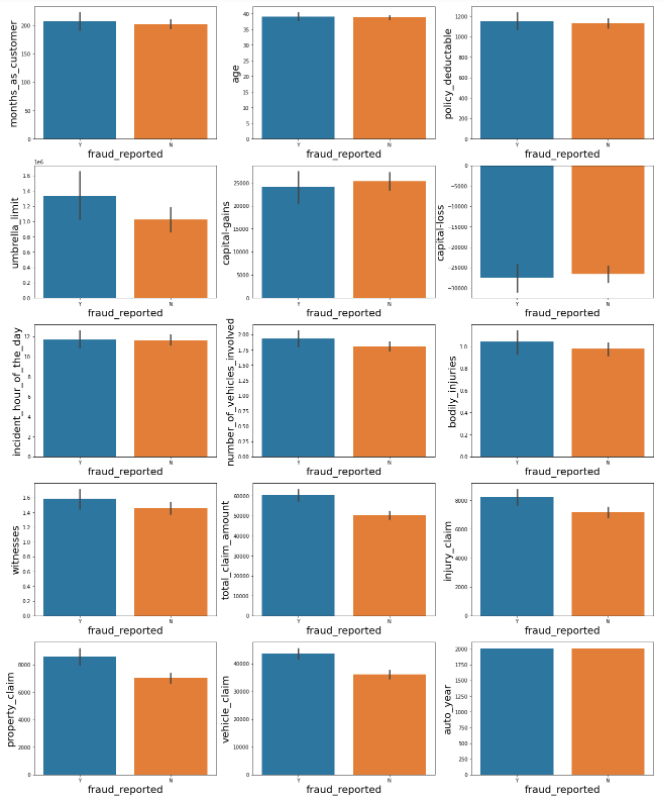
Most of the Insurance claims are not fraud.

We can also observe in below snapshot that there is a class imbalance which needs to be corrected at later stages



**Analysing the relationship between independent variable and dependent Variable:**

**Continuous data columns vs Target Variable:**



Most of the Continuous data columns doesn’t affect the target variable much. Only below listed column have an impact

1. Higher umbrella\_limit has more fraud claims

2. Higher total\_claim\_amount has more fraud claims

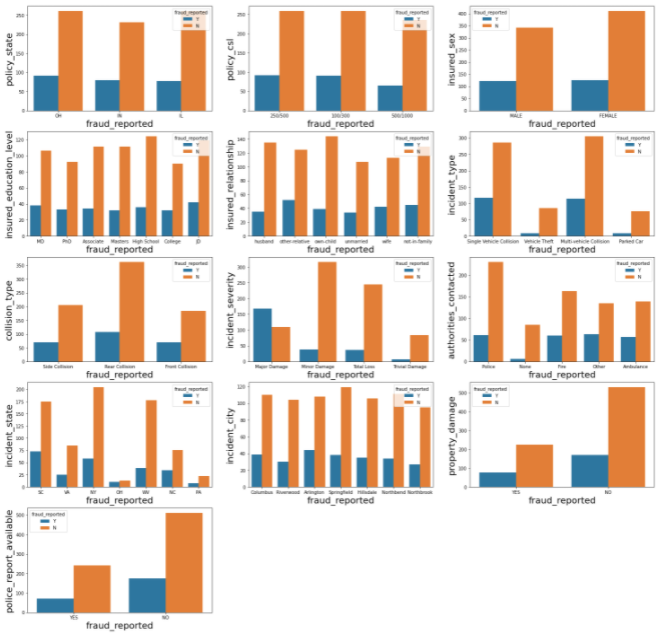
3. Higher injury\_claim amount has more fraud claims

4. Higher property\_claim amount has more fraud claims

5. Higher vehicle\_claim amount has more fraud claims

Rest of the columns relationship with target variable is almost same and doesn’t contribute much in prediction

**Categorical Data columns vs Target Variable:**



From the above graph, we can observe following details,

1. policy\_state - OH and IL state claims has higher count of genuine claims.

Fraud claims range is almost same which means it doesn't contribute much for fraud claims

2. policy\_csl - 250/500 and 100/300 CSL has higher count of genuine claims.

Fraud claims range is almost same which means it doesn't contribute much for fraud claims

3. insured\_sex - Female Insured Claims has higher count of genuine claims.

Fraud claims range is almost same which means it doesn't contribute much for fraud claims

4. insured\_education\_level - claims of Insured with educational level High school has higher count of genuine claims.

Fraud claims range is almost same which means it doesn't contribute much for fraud claims

5. insured\_relationship - Own-child insured relationship has higher count of genuine claims.

other-relative relationship followed by not-in-family has higher count of Fraud claims.

6. incident\_type - Multi-vehicle Collision has higher count of genuine claims followed by Single Vehicle Collision.

Same incident type also has higher count of Fraud claims.

7. collision\_type - Rear Collision has higher count of genuine claims.

Same collision type also has higher count of Fraud claims.

8. incident\_severity - Minor Damage Severity type has higher count of genuine claims followed by Total Loss severity type.

Major Damage severity type has higher count of Fraud claims.

9. authorities\_contacted - Claims where police authority is contacted has higher count of genuine claims.

10. incident\_state - NY state incident claims have higher count of genuine claims followed by SC and WV state incident claims.

SC state incident claims has higher count of Fraud claims.

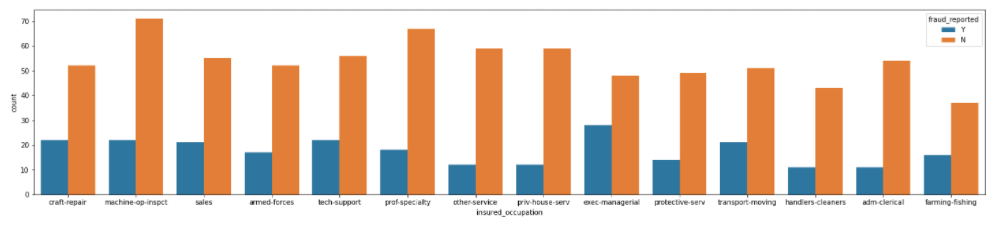
11. incident\_city - Springfield city incident claims have higher count of genuine claims.

Arlington city incident claims has higher count of Fraud claims.

12. property\_damage - No Property damage claims has higher count of genuine claims.

13. police\_report\_available - Most of the columns has No police report available so genuine and fraud claims count is higher in case when there are no police report available.

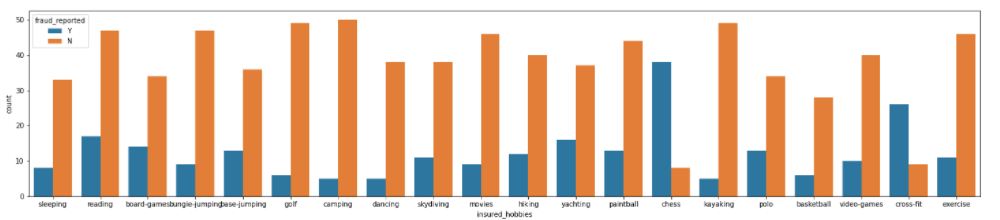
**Insured\_occupation vs fraud\_reported:**



We can observe following things from above comparison graph,

1. Claims with insured occupation as machine-op-inspct has higher genuine claims followed by prof-specialty
2. Claims with insured occupation as exec-managerial has higher fraud claims.

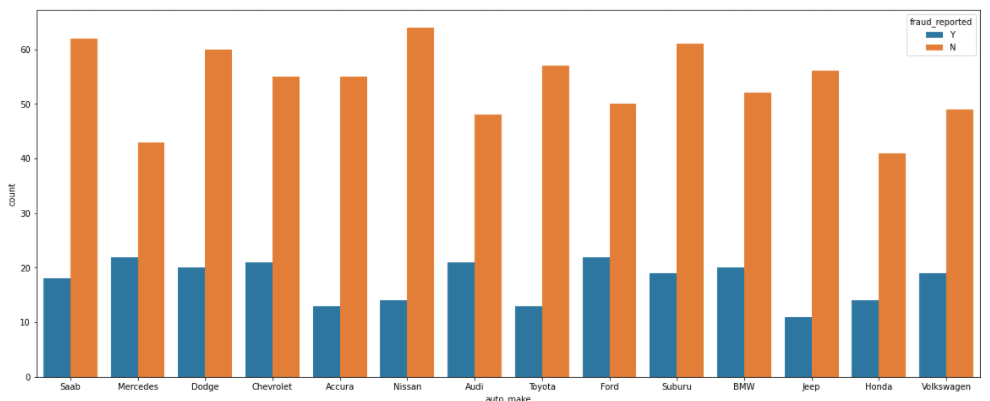
**insured\_hobbies vs fraud\_reported:**



We can observe following things from above comparison graph,

1. Claims with insured hobbies camping has higher genuine claims count followed by kayaking
2. Claims with insured hobbies chess has higher fraud claims count followed by cross-fit

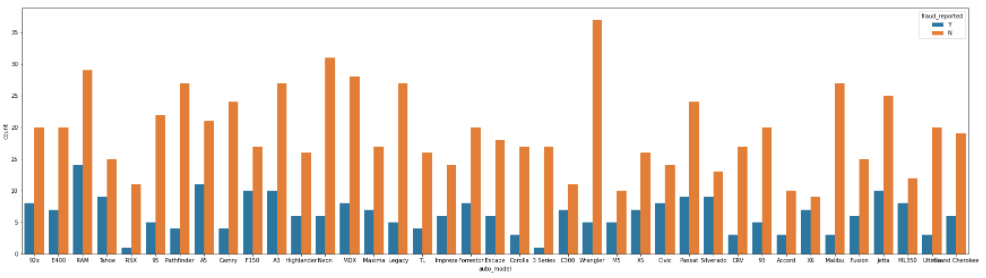
**auto\_make vs fraud\_reported:**



We can observe following things from above comparison graph,

1. Claims with Automobile Make Nissan has higher genuine claims count followed by Saab, Suburu Auto Make.
2. Claims with Automobile Make Ford, Mercedes has higher fraud claims count.

**auto\_model vs fraud\_reported:**



We can observe following things from above comparison graph,

1. Wrangler auto model involved claims has higher genuine claims count followed by Nissan and RAM auto model.
2. RAM auto model involved claims has higher fraud claims count followed A5 auto model

**Data Pre-Processing:**

It is one of the important steps in machine learning model building. Efficiency of the model is dependent on the data pre-processing. Higher quality data will yield better performing better efficiency Model.

This process involves in treating the null values, incomplete data and Noisy data.

Data Cleaning, Data Integration, Data Transformation and Data reduction are the process steps involved under this stage.

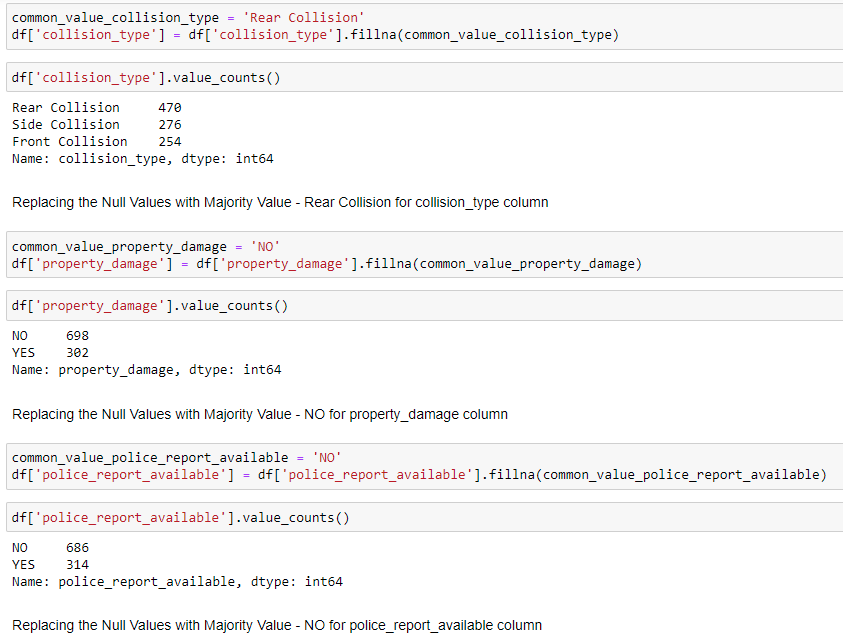
In case of data cleaning, we deal with missing values/ null values, Outliers and skewness treatment.

As the name suggests, in case of data integration, integration of data from multiple stage into single data stage and normalization is involved in data transformation and in data reduction, feature and target selection along with dropping the redundant features is done.

**Invalid ‘?’ data correction (Null/ Missing Values Treatment):**

In earlier observation we came to know that there are ‘?’ data in 3 columns and also 100% null value in \_c39 column. As discussed, \_c39 column is dropped and filling the ‘?’ with mode or common value as discussed.

We shall convert the ‘?’ data to null values and then replace the null values with common value.



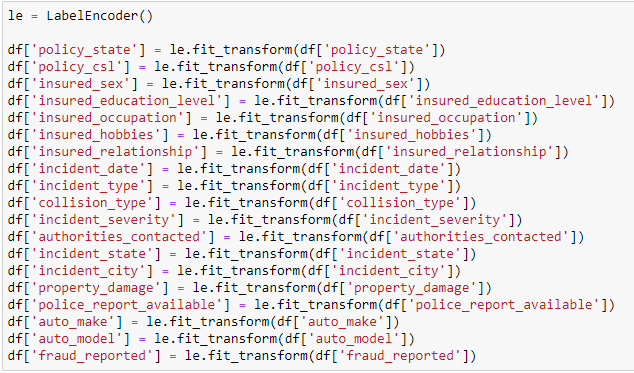
fillna is used as shown in above snapshot

1. Replacing the Null Values with Majority Value - Rear Collision for collision\_type column
2. Replacing the Null Values with Majority Value - NO for property\_damage column
3. Replacing the Null Values with Majority Value - NO for police\_report\_available column

**Encoding of Categorical Column data to Numerical data:**

**Using Label Encoding:**

We have observed that there are categorical data columns in the used dataset. So, we shall use Label Encoding to convert the categorical data to numerical data.

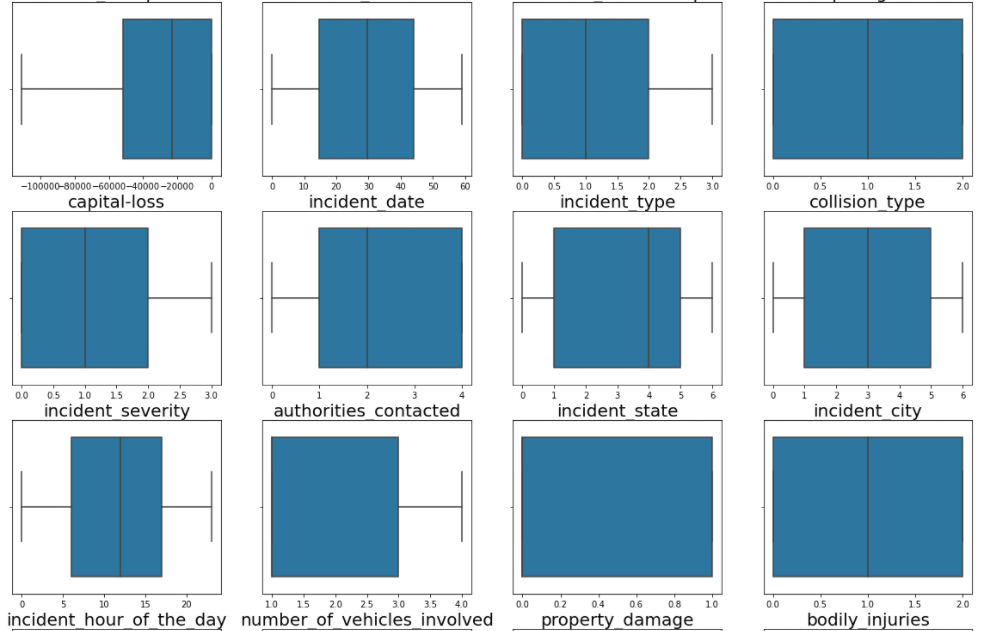
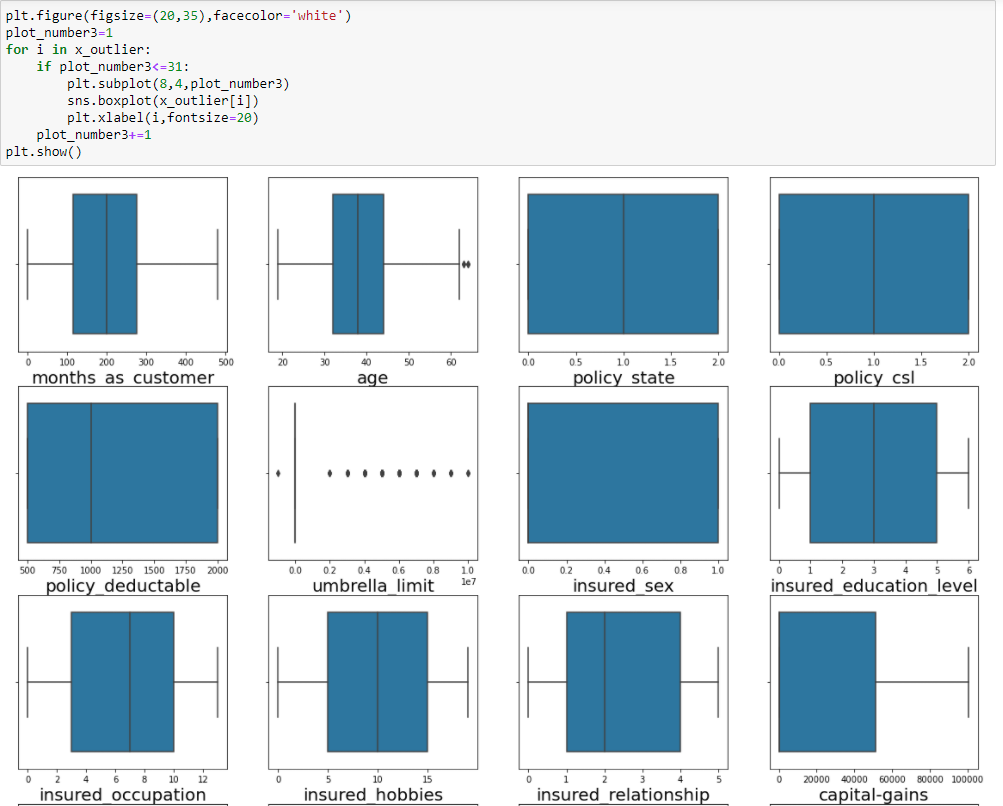
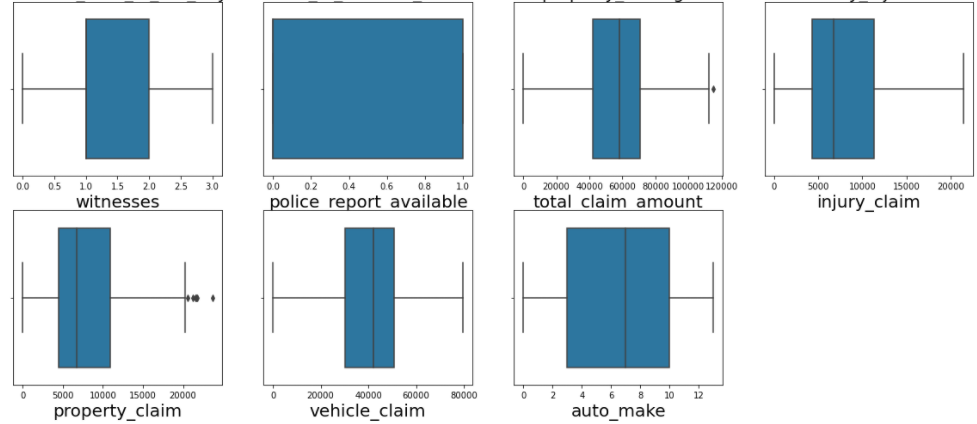


After dropping the unwanted categorical columns, above list of categorical columns are converted to numerical data using Label Encoding.

**Outliers Checking and Correction:**

Outliers are thedata points which are distant from the other similar points. It exists due to variability in the measurement or may indicate experimental errors.

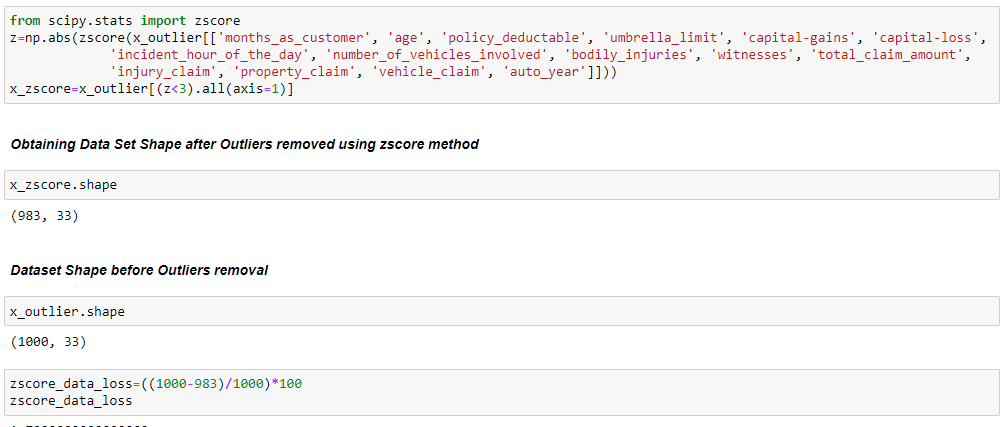
So, it’s important to correct the Outliers. Using Boxplots, outliers presence is checked in columns as shown below,

From above plots, we can confirm that there are outliers in umbrella\_limit.

We can also observe there is a presence of outliers in age, total\_claim\_amount, property\_claim Column Data which are continuous data originally

Using Zscore Technique, removing the Outliers as shown below,



Upon analysing, found 1.70% of Data loss is occurred when ZScore Method is used for Outliers removal for this Data set. which is within acceptable range.

Upon analysing found umbrella\_limit, number\_of\_vehicles\_involved, total\_claim\_amount, vehicle\_claim columns have skewness outside the allowed range of -0.5 to +0.5

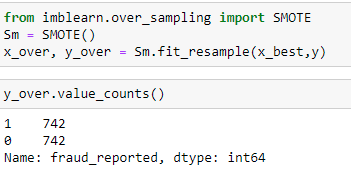
So, skewness is corrected using CBRT Technique to an extent possible.

**Class Imbalance Treatment:**

We have observed in our earlier observations that there is a class imbalance of data under target column. So, we shall use SMOTE (Over sampling method) to balance the class imbalance.

SMOTE (Synthetic Minority Oversampling Technique) works basically through picking a point from the minority class randomly and computing the k-nearest neighbors of this point. Obtained synthetic points are added between the chosen point and its neighbors.

Below is the snapshot of the same,



We can observe that class is balance with equal data count in target variable.

**Model Building:**

Classification and Regression Model building are most common model building types.

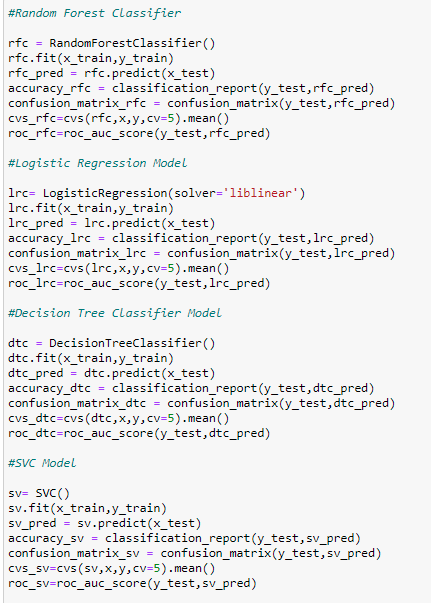
We have observed that this dataset is a classification problem since the objective is to detect whether the claim is fraudulent or not. so, we shall proceed with Classification Model Building

Dependent and Independent Variable are separated and then the data is fitted into the model and then values will be passed to train\_test\_split for random state training and testing.

train\_test\_split method is used tosplit the data arrays into two subsets for data training and data testing. We can either set the random state manually or it will be done automatically in this method. We will be having 4 outputs namely x\_train, x\_test, y\_train and y\_test.

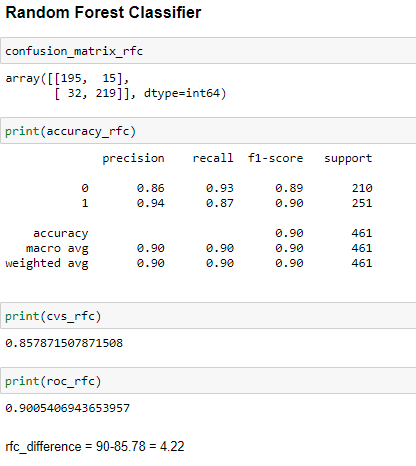
x\_train, x\_test output consists of training and testing predictor variables and y\_train, y\_test consists of training and testing target variables.

Once train\_test\_split method is processed, we need to choose the models to pass the training variables. We shall build 4 models and compare the accuracy score to choose the best model for hyper parameter tuning which is done after the model building.



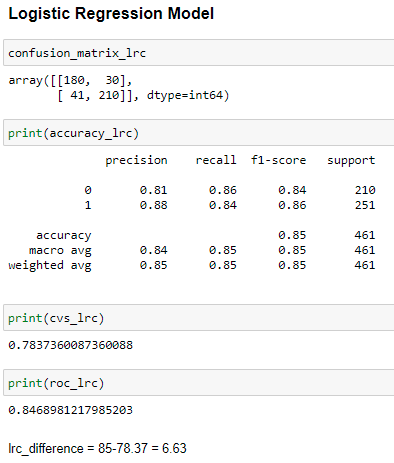
1. **Random Forest Classifier:**

Random forest algorithm creates decision trees on data samples and then gets the prediction from each of them and finally selects the best solution by means of voting. It is an ensemble method which is better than a single decision tree because it reduces the over-fitting by averaging the result.



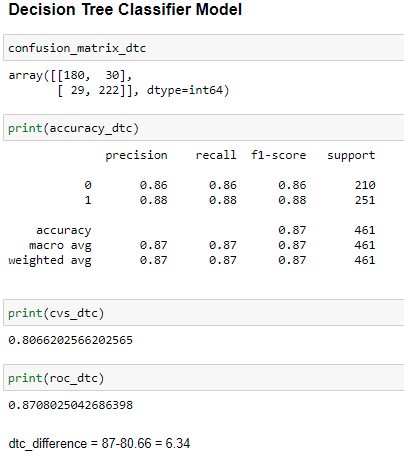
1. **Logistic Regression Model:**

It is a supervised learning classification algorithm used to predict the probability of a target variable. The nature of target or dependent variable is binary, which means there would be only two possible classes 1 (stands for success/yes) or 0 (stands for failure/no). Mathematically, a logistic regression model predicts P(Y=1) as a function of X. It is one of the simplest ML algorithms that can be used for various classification problems such as spam detection, Diabetes prediction, cancer detection etc.



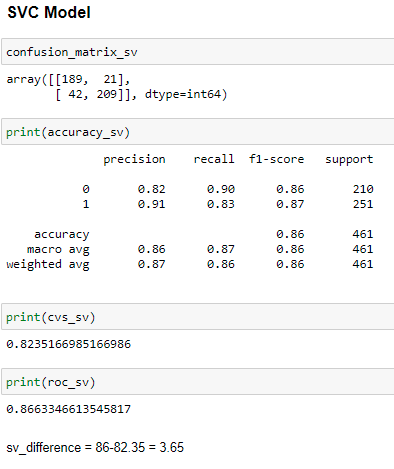
1. **Decision Tree Classifier Model:**

Decision Tree Classifier is a class capable of performing multi-class classification on a dataset. Decision Tree Classifier takes as input two arrays: an array X, sparse or dense, of shape (n\_samples, n\_features) holding the training samples, and an array Y of integer values, shape (n\_samples,), holding the class labels for the training samples. It is capable of both binary (where the labels are [-1, 1]) classification and multi-class (where the labels are [0, …, K-1]) classification.



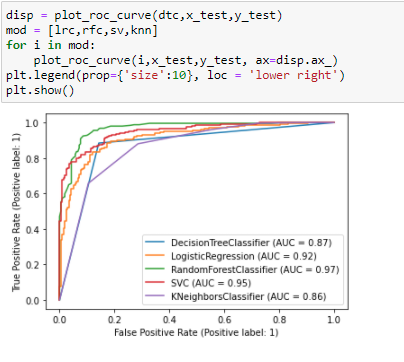
1. **Support Vector Classifier (SVC) Model:**

SVC is a non-parametric clustering algorithm that does not make any assumption on the number or shape of the clusters in the data. In SVC data points are mapped from data space to a high dimensional feature space using a kernel function. In the kernel's feature space, the algorithm searches for the smallest sphere that encloses the image of the data using the Support Vector Domain Description algorithm. This sphere, when mapped back to data space, forms a set of contours which enclose the data points. Those contours are then interpreted as cluster boundaries, and points enclosed by each contour are associated by SVC to the same cluster.



Upon reviewing the difference between Accuracy and cross validation score, KNN Classifier Model ranks with least difference followed by SVC and Random Forest Classifier. However, Random Forest Classifier has best AUC score as shown below compared to SVC and also accuracy is high so choosing Random Forest Classifier as Best Model and proceeding with Hyper Parameter tuning

**AUC - ROC curve:** It is a performance measurement for the classification problems at various threshold settings. ROC is a probability curve and AUC represents the degree or measure of separability. It tells how much the model is capable of distinguishing between classes. Higher the AUC, the better the model is at predicting 0s as 0s and 1s as 1s. The ROC curve is plotted with TPR against the FPR where TPR is on the y-axis and FPR is on the x-axis.

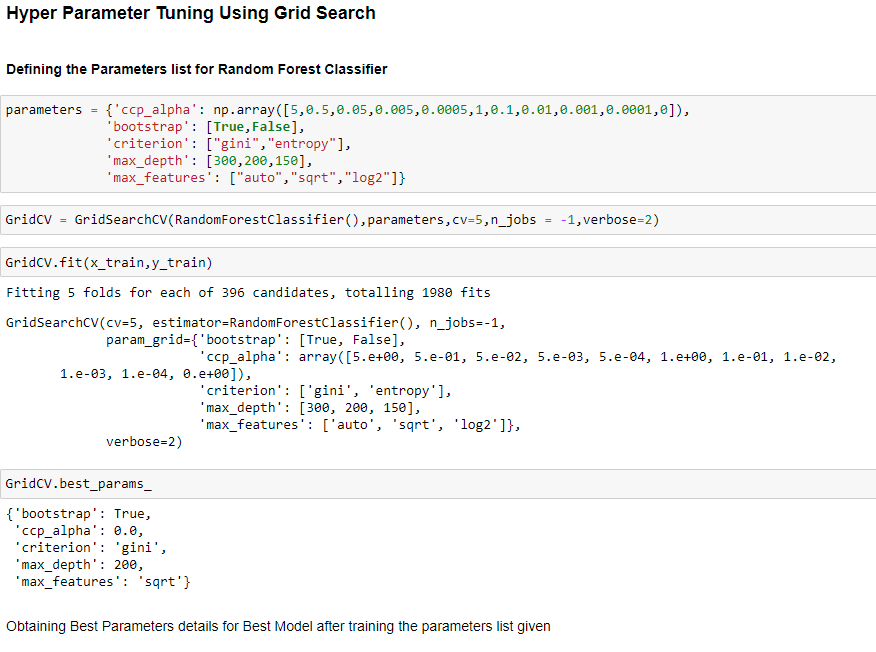


**Hyper Parameter Tuning:**

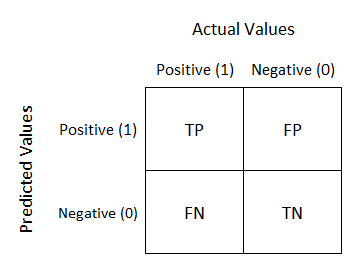
Hyper parameter Tuning is an optimization process involved in finding the hyper parameters of a desired machine learning algorithm measured on a validation set to give best performance.

There are multiple techniques to perform the Hyper Parameter Tuning, we shall proceed with Grid Search CV technique. In this technique, Model is evaluated for the given hyper parameters range.

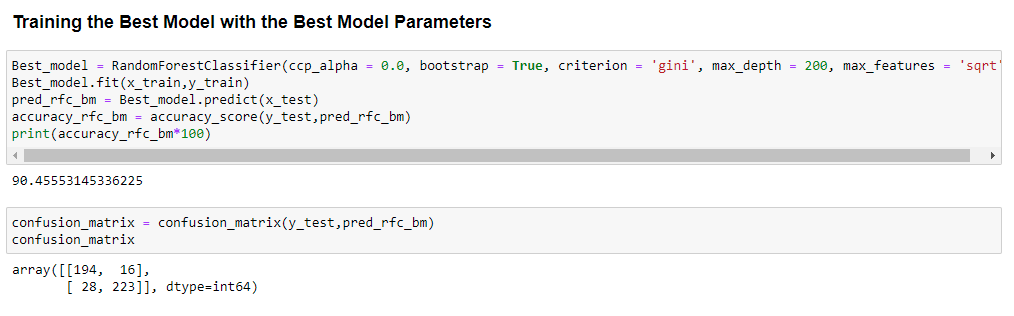
Below is the snapshot the Hyper Parameter Tuning performed on Random Forest Model,



**Confusion matrix:** It is used to describe the performance of a classification model (or ‘classifier’) on a set of test data for which the true values are known.



* **TN/True Negative:** the cases were negative and predicted negative.
* **TP/True Positive:** the cases were positive and predicted positive.
* **FN/False Negative:** the cases were positive but predicted negative.
* **TN/True Negative:** the cases were negative but predicted positive.



We can observe that the Model accuracy after Hyper Parameter tuning is around 90.45 %.

**Conclusion:**

Machine Learning Model (Random Forest Classifier) is built successfully to predict the Auto Insurance fraudulent claims which would assist the insurance companies to perform the business effectively.

***References:***

[***https://scikit-learn.org/***](https://scikit-learn.org/)

[***https://towardsdatascience.com/***](https://towardsdatascience.com/)