**INSURANCE CLAIM FRAUD DETECTION PROJECT BLOG -2**

**Author – Vaibhav Tayade**



**Contact:** 8087970061

**Email id:** vaibhav\_t29@rediffmail.com

**PROBLEM DEFINITION & INTRODUCTION:**

Insurance fraud is nothing but any act committed to defraud an insurance process. It occurs when a claimant attempts to get some benefit or advantage they are not entitled to, or when insurer knowingly denies some benefits that is due. Insurance fraud detection is a very challenging problem in today’s world.

Insurance frauds includes the range of improper activities which could range from staging the incident, misrepresenting the situation including the relevant actors and the cause of incident and finally the extent of damage caused. Situations could include 1) Covering-up for a situation that wasn’t covered under insurance (e.g. drunk driving, performing risky acts, illegal activities etc.). 2) Misrepresenting the context of the incident: This could include transferring the blame to incidents where the insured party is to blame, failure to take agreed upon safety measures. As per the FBI (Federal Bureau of Investigation) report, the insurance industry in the USA consists of over 7000 companies that collectively received over $1 trillion annually in premiums. FBI also estimates the total cost of insurance fraud could be more than $40 billion annually.

Hence the insurance industry has an urgent need to develop capability that can help identify potential frauds with a high degree of accuracy, so that other claims can be cleared rapidly while identified cases can be scrutinized in detail.

The traditional way of fraud detection includes fraud indicators, providing cases to investigation team which may work of that particular cases for months or for whole year. This challenges may reply of manual intervention which may lead to may limitations.

Machine learning techniques allow us improving the predictive accuracy, enabling loss control units to achieve higher coverage with low false positive rates. In this blog/article, multiple machine learning techniques for fraud detection are presented and their performance on various data sets examined. The impact of feature engineering, feature selection and parameter tweaking are explored with the objective of achieving superior predictive performance.

In this project of Machine Learning, we’ll be working on Insurance Fraud Detection project, wherein we’ll be understanding the dataset, the relation between various parameters on fraud detection etc.

In this problem, we have been provided with the dataset which includes customer details. It also has details of the accident on the basis of which the claim have been made.

**Content of the dataset:**

The dataset contains following information regarding the policy holder.

**months\_as\_customer** – It gives number of months a customer is holding policy for.

**Age** – It is the age of the policy holder.

**policy\_number** – Policy number of the policy holder.

**policy\_bind\_date** – The date on which policy got activated.

**policy\_state** – The state of Unites States in which policy have been activated.

**policy\_csl** - An insurance class is a type of insurance coverage such as liability, health, legal expenses, or construction risk. An insurance class is a type of insurance coverage such as liability, health, legal expenses, or construction risk. Here it’s given in numeric form as per the insurance industry norms.

**policy\_deductable** – Policy Deductible is the amount that a policy holder has to pay before the insurance company starts paying up.

**policy\_annual\_premium** -- The total amount of premium paid annually is called the annualized premium

**umbrella\_limit** - Umbrella limit is nothing but your net worth. It is the maximum limit you can get an insurance cover.

**insured\_zip** – It is basically an amount which a policy holder can get if any chance of claiming comes in future.

**insured\_sex** – The gender of policy holder.

**insured\_education\_level** – Education level of policy holder.

**insured\_occupation** – Occupation of Policy holder.

**insured\_hobbies** – Hobbies of Policy holder.

**insured\_relationship** – Relationship of insured person with the person paying premium.

**capital-gains** – The profit an insurance company can earn from the policy holder.

**capital-loss** – The loss an insurance company can get from the policy holder.

**incident\_date** – The date on which incident happen for which the claim has been made.

**incident\_type** – The type of the incident.

**collision\_type**- The type of the collision in an incident.

**incident\_severity** – Severity level in the incident.

**authorities\_contacted** – Any government authorities contacted after the incident occurred.

**incident\_state** – The state in which incident occurred.

**incident\_city** – The city in which incident occurred.

**incident\_location** – The Location in the city where incident occurred.

**incident\_hour\_of\_the\_day** – The timing of incident.

**number\_of\_vehicles\_involved** – Number of vehicles involved in the collision.

**property\_damage** – Weather the property has been damaged or not.

**bodily\_injuries** – Weather there’s any physical injury during the collision.

**Witnesses** – Weather there’s any witness during the collision.

**police\_report\_available** – Weather the incident has been reported to police station or not.

**total\_claim\_amount** – The total amount of claim has been made to the company.

**injury\_claim** – The amount claim for the injury.

**property\_claim** – The amount claim for the Property damage.

**vehicle\_claim** – The amount claim for the vehical damage.

**auto\_make** – The name of auto company the vehicle belongs to.

**auto\_model** - The model of the vehicle which was involved in the incident.

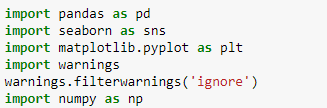
**auto\_year** – The year on which the vehicle was made.

**fraud\_reported** – weather the fraud reported on that particular case or not.

**\_c39** – doesn’t hold any valid information.

**Importing Various Libraries:**

The very first thing which we need to do is importing relevant libraries for data analysis.



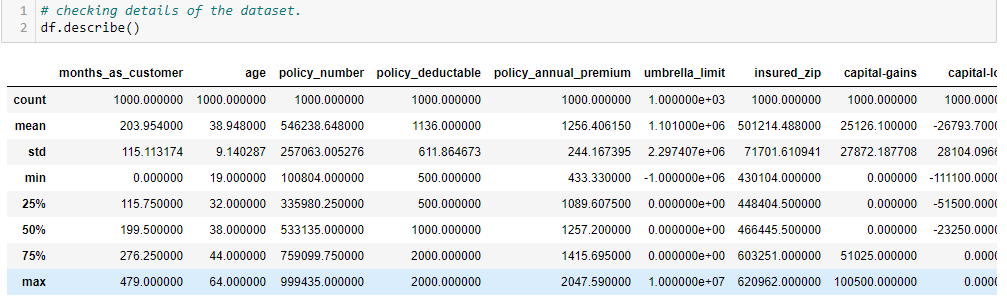
Let’s talk about libraries in brief, pandas is the library used to for data analysis, numpy is another library used for numerical data. seaborn and matplotlib are the library used for data visualization.

**Importing data containing csv file:**



We have imported csv file using pandas and created a data frame named df.

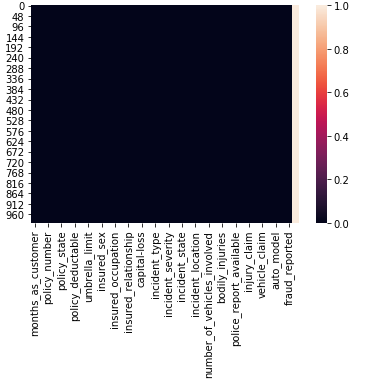
**DATA ANALYSIS:**



**Observation:** We can see that there are few columns like ‘months\_as\_customer’, ‘age’, ‘umbrella\_limit’, ‘insured\_zip’, ‘capital-gains’, ‘total\_claim\_amount’ where we can say that mean Is higher than the median indicating presence of outliers and skewness in the dataset.

**EXPLORATORY DATA ANALSIS:**

**CHECKING FOR NULL VALUES:** We can see that there are no null values in our dataset shown in the below heatmap, except last column i.e. \_c39 which is anyways irrelevant column.



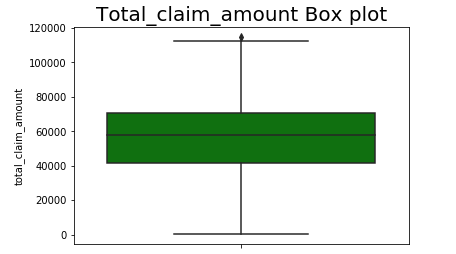
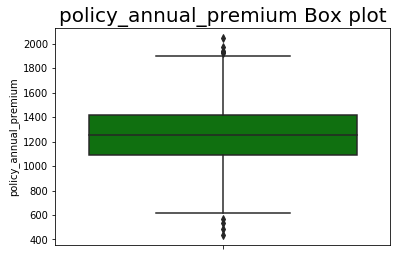
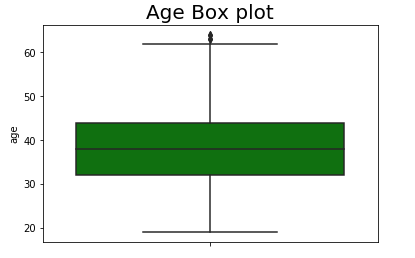
**REMOVING IRRELEVANT DATA:**

We found out that there are columns like ‘months\_as\_customer’, ‘policy\_number’, ‘policy\_bind\_date’, ‘auto\_make’, ‘auto\_model’, ‘incident\_date’, ‘\_c39’, have no direct relationship with the fraudulent cases. We can drop such columns from the datasets.

We also found out that there were few columns like collision type, property damage, police report available which contains values having no significant data i.e. ‘?’, which we need to replace with ‘mode’ value of that particular column. In column ‘property\_damage’, we replaced ‘?’ with ‘NO’, in column ‘collision\_type’ we replaced ‘?’ with ‘Rear Collision’ and in column ‘police\_report\_available’ we replaced ‘?’ with ‘NO’ which were their respective mode values.

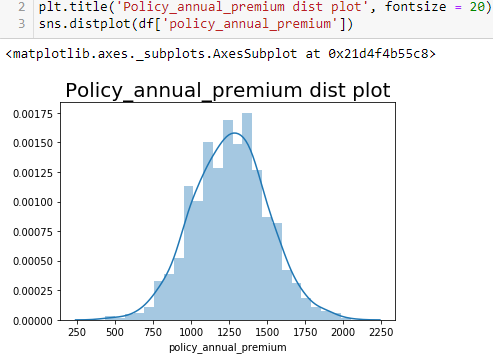
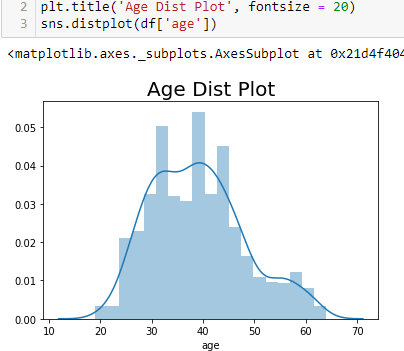
**UNIVARIATE ANALYSIS:** Univariate analysis is nothing but the understanding the dataset column wise. One needs to understand the content of the dataset. There should not be presence of outliers in the dataset, outliers is nothing but the extraordinary values present in the dataset which must be present due to some external factors or wrong mentioning of data while building the dataset or maybe due to some other reasons. In other words they are data records that differ dramatically from all others, they distinguish themselves in one or more characteristics.

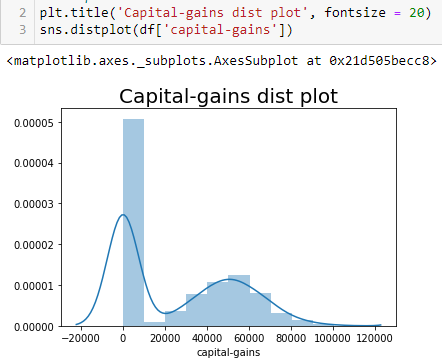
We’ll plot the box plot of all the numerical column in order to identify the outliers in the dataset.

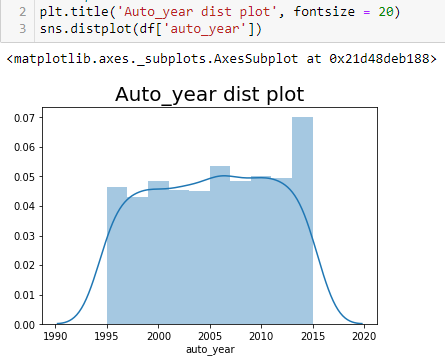
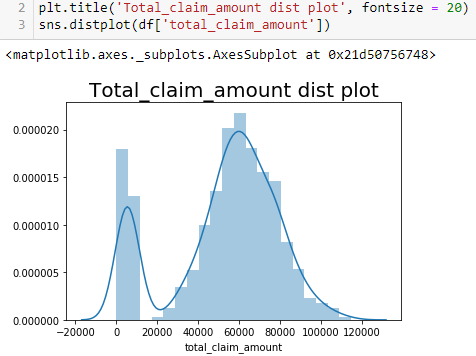
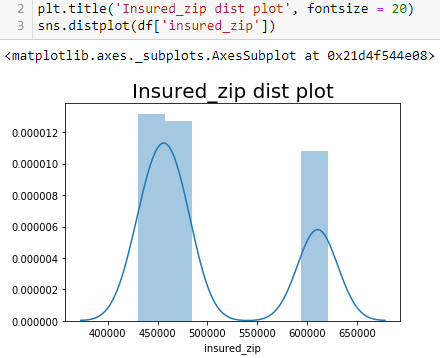
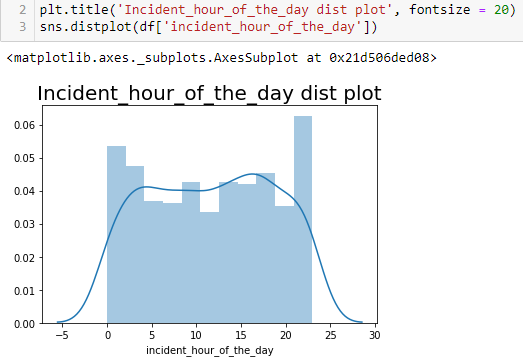
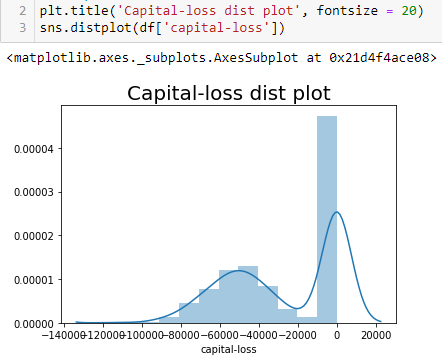


**Observation:** Out of all the columns we can see that this 3 columns i.e. Age, policy\_annual\_premium, Total\_claim\_amount columns contains outliers in them.

The another important process in doing **exploratory data analysis** is checking for **skewness** in the dataset. Skewness is nothing but the measure of symmetry and asymmetry of data distribution. Skewness could be positive (i.e. right sided skewed) or negative (i.e. left sided skewed). Here in our project we tried to check for the skewness in all numeric columns.







**Observation:**

Distplot 1) Age column is somewhat evenly distributed and skewed bit.

Distplot 2) Policy annual premium distplot is evenly distributed and both sidely skewed.

Distplot 3) Capital-gains column distplot is also not evenly distributed and skewed a bit.

Distplot 4) Capital-loss column distplot is also not evenly distributed and skewed a bit.

Distplot 5) Incident\_hour\_of\_the\_day column distplot is somewhat evenly distributed.

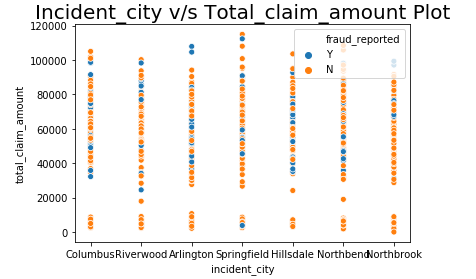
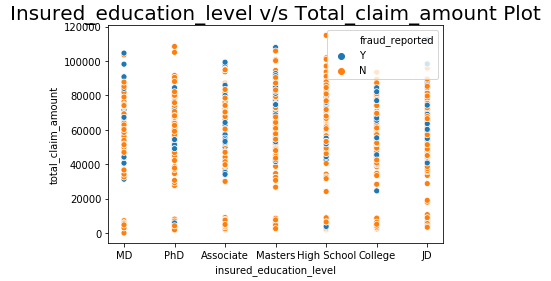
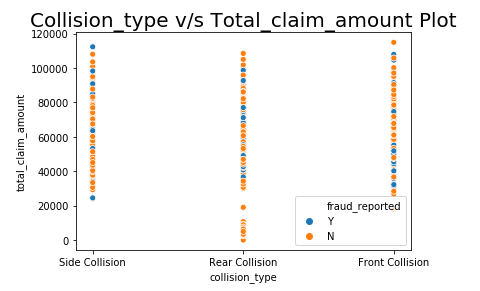
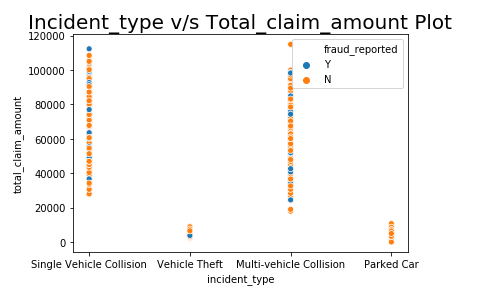
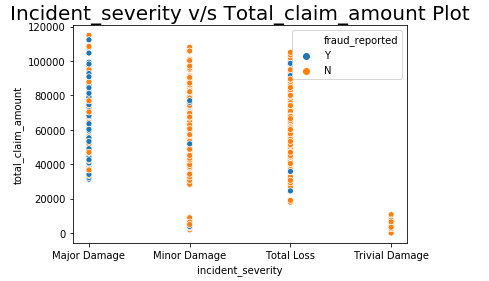
Distplot 6) Insured zip column distplot is not even distributed.

Distplot 7) Total\_claim\_amount column distplot is also not evenly distributed and skewed a bit.

Distplot 8) Auto\_year column distplot is somewhat evenly distributed.

**BIVARIATE ANALYSIS:** Bivariate analysis is nothing but analyzing the dataset using two columns and finding relationship between them.

Here we did bivariate analysis of various columns using scatter plot. Observations are given below.



**Observations:**

**Scatter plot 1)** In this scatter plot, ‘fraudulant cases’ are more in ‘Major Damage’ cases of ‘Incident\_severity’ column and between moderate to high values of ‘total\_claim\_amount’ column.

**Scatter plot 2)** In this scatter plot, No proper relationship between ‘incidentype’ and ‘total claim amount’ column seen.

**Scatter plot 3)** In this scatter plot, ‘side collision’ cases has least cases of ‘fraudulant’, while ‘Rear collision’ and ‘front collision’ has moderate cases of ‘fraudulent’.

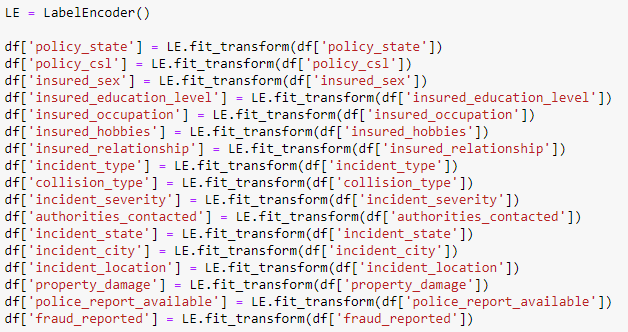
**Scatter plot 4)** In this scatter plot, college and JD education level has moderate cases of fradulant while Masters has minimum cases of fraudulant.

**Scatter plot 5)** In this scatter plot, Hillsdale city has maximum cases of fraudulant while Northbrook has minimum cases of fraudulant and other cities has moderate cases of fraudulant.

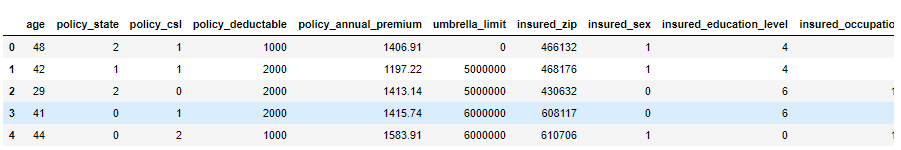
**PRE PROCESSING PIPELINE:**

**LABEL ENCODING THE CATEGORICAL COLUMNS:**

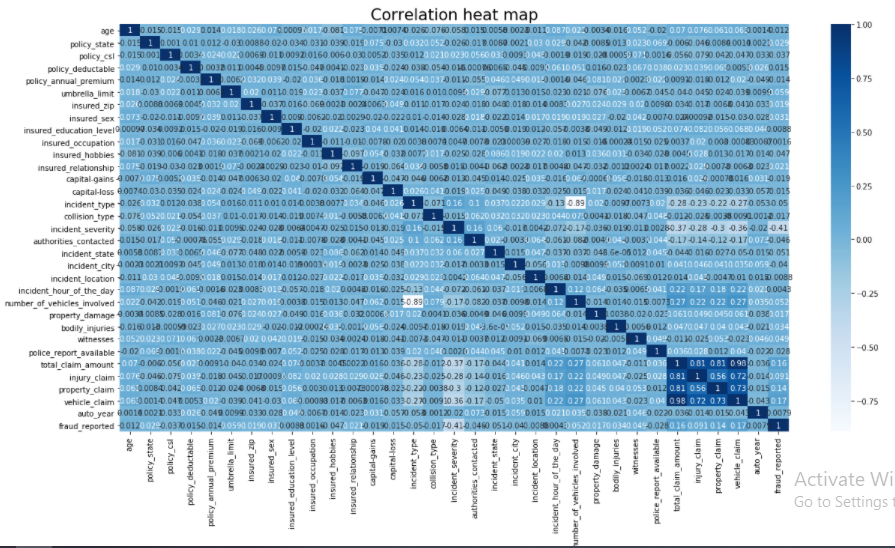
We have observed here is that most of the columns are of object data type in nature, we need to convert them to numeric data type. We can do that using label encoder technique. Label Encoding refers to converting the labels into numeric form so as to convert it into the machine-readable form.



After Label encoder we got all columns converted into numeric data columns.



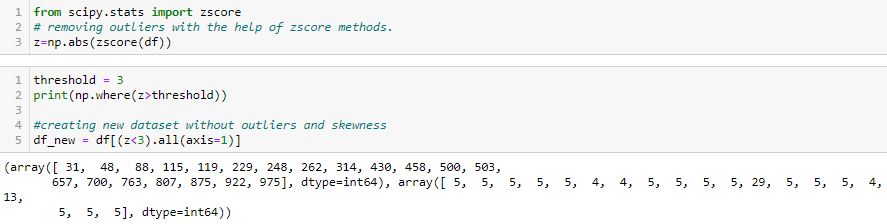
**Correlation Matrix Analysis:** A correlation matrix is a table showing correlation coefficients between variables. Each cell in the table shows the correlation between two variables. A correlation matrix is used to summarize data, as an input into a more advanced analysis, and as a diagnostic for advanced analyses.



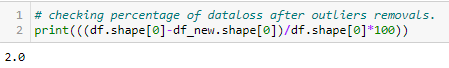
**Observation:** ‘Incident severity’ column is highly correlated with ‘fraud reported’ column, though negatively correlated.

**REMOVING OUTLIERS AND SKEWNESS:** Here in this dataset, we have used z-score method of removing outliers. A **z-score** gives you an idea of how far from the mean a data point is. But more technically it's a measure of how many standard deviations below or above the population mean a raw score is. A z-score can be placed on a normal distribution curve.

Codes are given below:



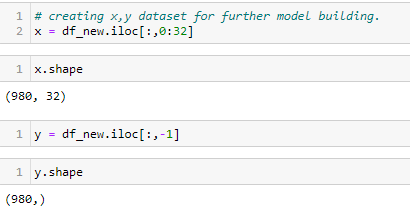
After removing outliers, we need to check for dataloss.



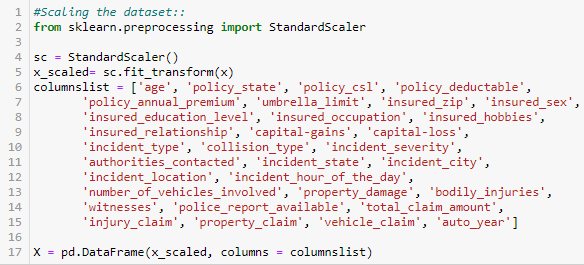
**Observation:** We observed here that there is almost 2% of data loss which is very much fine and within our limits.

**X and Y dataset making:**

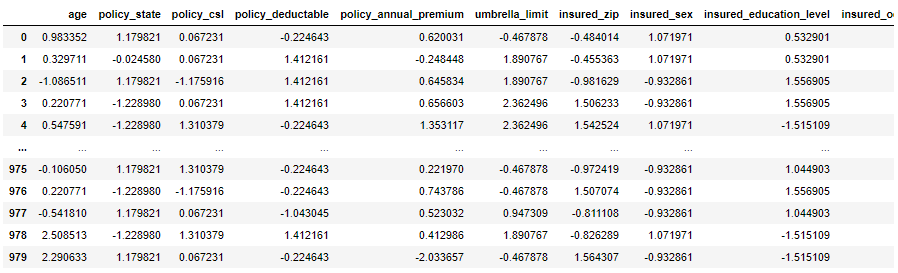
After removing outliers we need to extract ‘x’, ‘y’ data frames from ‘df\_new’ data frame in order to proceed further with the model making. ‘x’ data frame contains all columns except the ‘fraud reported’ column while ‘y’ data frame exclusively contains ‘fraud reported’ column only.



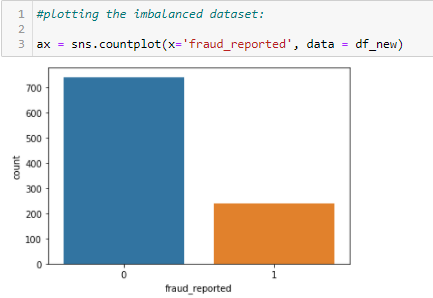
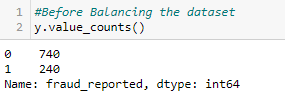
**SCALING THE DATASET:** Scaling is one of the most important step in model making. We will use Standard Scaler library in order to scale all columns.



After scaling the dataset by using Standard Scaler as shown by above method, we got new dataset which is now scaled as shown below.



**CHECKING TARGET COLUMN DATA BALANCING:** Imbalanced data creates biasedness in model making which may hamper accuracy of the model. Data Balancing typically refers to a classification problem where the number of observations per class is not equally distributed; often you'll have a large amount of data/observations for one class (referred to as the majority class), and much fewer observations for one or more other classes (referred to as the minority class), that’s why checking balancing of the target column is must. Here in our problem also we have imbalanced datasets as shown below.

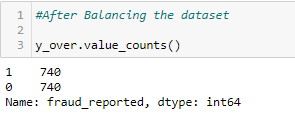


**Observation:** we can see that here also our target column that is ‘fraud reported’, we have 740 cases of 0 i.e. ‘Yes’ (fraud reported) and 240 cases of 1 i.e. ‘No’ (No fraud reported).

In order to balance the dataset we use SMOTE over sampling technique.

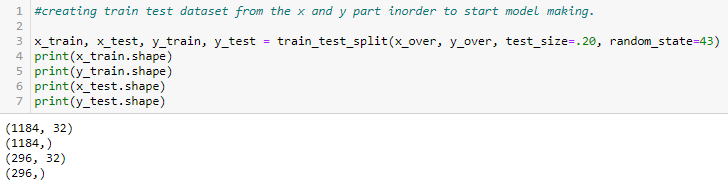


What this technique does is that, it oversamples the minority dataset and tries to match the cases of majority datasets. After balancing the dataset.



Now, we can say that we have equal cases of fraud reported and fraud not reported. So now our model will be unbiased.

**TRAIN-TEST DATASET BUILDING:** The train-test split is a technique for evaluating the performance of a machine learning algorithm. The procedure involves taking a dataset and dividing it into two subsets Here we have kept 20% of dataset for testing the model while 80% of dataset for training the model, and random\_state value of 43 is kept.

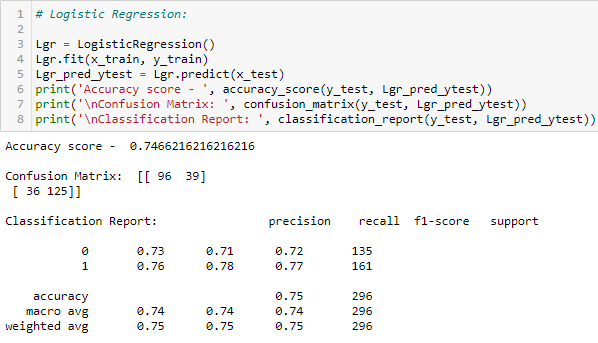


**BUILDING MACHINE LEARNING MODELS:**

**Machine learning model making:** Here in our problem we will try making machine learning model with seven different types of algorithms i.e. Logistic Regression, GaussianNB, Decision Tree Classifier, Random Forest Classifier, Ada Boost Classifier, Support Vector Classifier, K-nearest neighbour classifier methods. We will try to find out each model’s Accuracy score, Confusion matrix, Classification report as well. Accuracy score will give us the percentage accuracy of the model in predicting the test datasets. Confusion matrix is a 26 by 26 matrix with the probability of each reaction to each stimulus. The classification report is used to measure the quality of predictions from a classification algorithm. The report shows the main classification metrics precision, recall and f1-score on a per-class basis. The metrics are calculated by using true and false positives, true and false negatives.

**Logistic Regression Method:** Logistic regression is a statistical model that in its basic form uses a logistic function to model a binary dependent variable, although many more complex extensions exist. Logistic regression is basically a classification type of method, a statistical analysis method used to predict a data value based on prior observations of a data set. Based on historical data about earlier outcomes involving the same input criteria, it then scores new cases on their probability of falling into a particular outcome category.

Logistic regression is used to describe data and to explain the relationship between one dependent binary variable and one or more nominal, ordinal, interval or ratio-level independent variables.



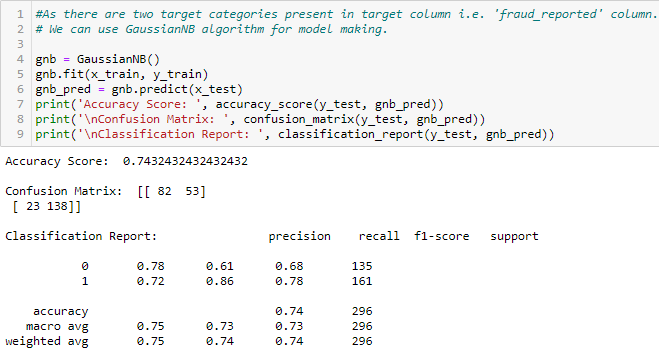
By using logistic regression method, we got accuracy score of **74.66%** which is very good. But this accuracy score may be due to over fitting of the data, we can check for over fitting by checking cross validation score of the model. It is basically a technique for evaluating machine learning models by training several models on subsets of the available input data and evaluating them on the complementary subset of the data.

Overfitting is a modeling error in statistics that occurs when a function is too closely aligned to a limited set of data points. Overfitting the model generally takes the form of making an overly complex model to explain idiosyncrasies in the data under study. Here we’ll check cross validation score on 5 subsets of the datasets.



We can see that the cv score of logistic regression model is **73.51%** which is very much close to Accuracy score of model i.e. **74.66%**. We’ll try to make machine learning model on various other algorithms as well and we’ll check their accuracy score and cross validation score. The model which has least difference in their accuracy and cv score, and also least errors, we’ll select that model as our final model.

**GaussianNB Method:** A Gaussian Naive Bayes algorithm is a special type of NB algorithm. It's specifically used when the features have continuous values. Gaussian Naive Bayes method works on two categorical target datasets only, as our dataset contains two target categories only. We can very well use this method.

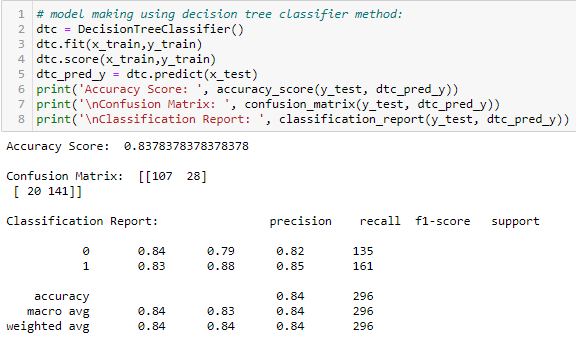


The accuracy score of the GaussianNB method is **74.32%** which is again very good, but we need to check cv score as well.



Cv score of GaussianNB model is **71.48%** which is again not very close to our model’s accuracy score indicating overfitting.

**Decision Tree Classifier Method:** A Decision Tree is an algorithm used for supervised learning problems such as classification or regression. Decision tree methodology is a commonly used data mining method for establishing classification systems based on multiple covariates or for developing prediction algorithms for a target variable. Each leaf of the tree is labeled with a class or a probability distribution over the classes. A tree can be "learned" by splitting the source set into subsets based on an attribute value test.

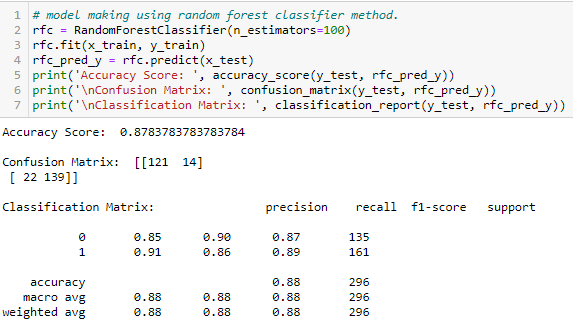


Accuracy of our decision tree model is **83.78%** which is good but again this might be due to overfitting so we need to check cross validation score as well.



Cross validation score of decision tree classifier model is **83.44%** which is very close to decision tree classifier model’s accuracy score indicating less overfitting of the datasets in the model making.

**Random Forest Classifier Method:** Random Forest is a classifier that contains a number of decision trees on various subsets of the given dataset and takes the average to improve the predictive accuracy of that dataset.

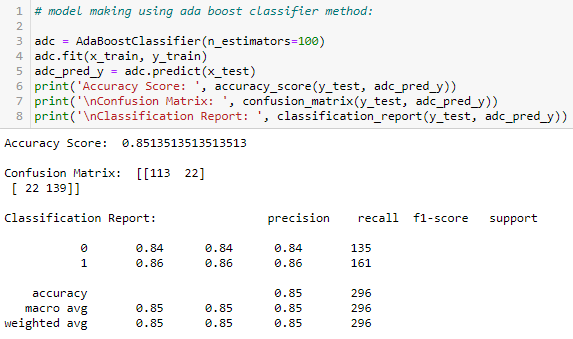


Accuracy of our random forest classifier model is **87.83%** which is amazing but we need to correlate it with cross validation score as well.

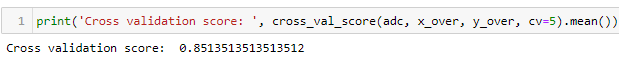


Cross validation score of the random forest classifier model is 87.56% which is very close to the random forest classifier model’s accuracy score indicating less overfitting in the model.

**Ada Boost Classifier Method:** Ada-boost or Adaptive Boosting is one of ensemble boosting classifier. It combines multiple classifiers to increase the accuracy of classifiers. Ada Boost is an iterative ensemble method. Ada boost helps in combining multiple “weak classifiers” into a single “strong classifier”.

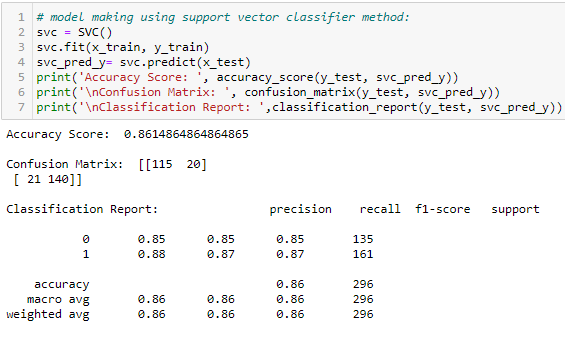


Ada Boost Classifier has an accuracy of **85.13%**, but we need to check it’s cross validation score as well.



Cross validation score of ada boost classifier model is 85.1351% which is very much close to the accuracy score indicating less overfitting in the model.

**Support Vector Classifier Method:** A support vector machine (SVM) is a supervised machine learning model that uses classification algorithms for two-group classification problems. The classifier separates data points using a hyperplane with the largest amount of margin. That's why an SVM classifier is also known as a discriminative classifier. SVM finds an optimal hyperplane which helps in classifying new data points.

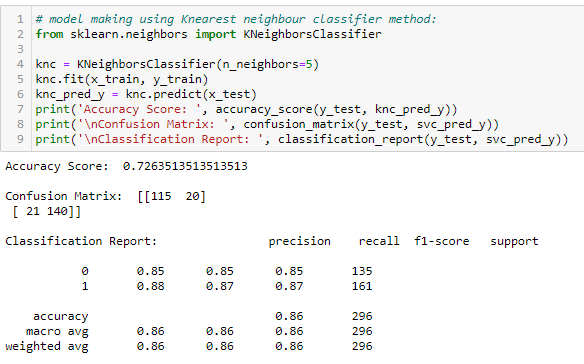


Accuracy score of support vector classifier is **86.14%** which is again good, but we need to check the cross validation score as well.



Cross validation score of the support vector classifier model is **85.40%** which is very much close to the accuracy score of the model indicating less overfitting in the model.

**K-Nearest Neighbour Classifier Method:** The k-nearest neighbors (KNN) algorithm is a simple, supervised machine learning algorithm that can be used to solve both classification and regression problems. In the case of classification and regression, we saw that choosing the right K for our data is done by trying several Ks and picking the one that works best. The nearest neighbour based classifiers use some or all the patterns available in the training set to classify a test pattern. These classifiers essentially involve finding the similarity between the test pattern and every pattern in the training set.



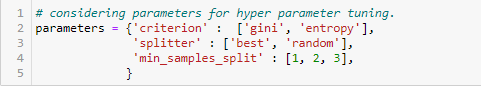
Accuracy score of K-nearest neighbor classifier model is **72.63%** which is very less compared to all other models, but again we need to cross check it with it’s cross validation score as well.



Cross validation score of K-nearest classifier model is **68.17%** which is far below the models accuracy score indicating overfitting in the model.

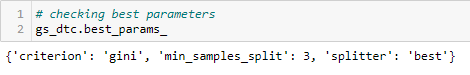
**Observation:** After analyzing each model’s accuracy score and cross validation score, we found out that difference between accuracy score and cross validation score of Decision Tree Classifier model is least also and of good percentage also. we can say that Decision Tree Classifier is our best fit model.

**HYPER PARAMETER TUNING OF MODEL:** Hyperparameter tuning is basically choosing a set of optimal hyperparameters for a learning algorithm which is mostly done with the help of grid search cv method. Grid search cv is basically the most used hyperparameter tuning method. With this technique, we simply build a model for each possible combination of all of the hyperparameter values provided, evaluating each model, and selecting the architecture which produces the best results.



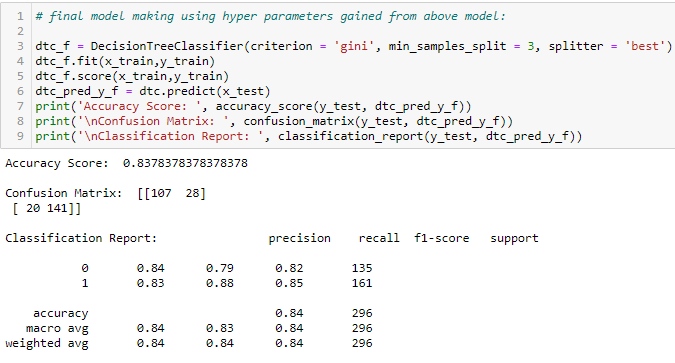
We have used ‘gini’ and ‘entropy’ for ‘criterion’ parameter, ‘best’ and ‘random’ for ‘splitter’ parameter, a range of 1 to 3 values for ‘min\_samples\_split’ parameters.

After passing this range of parameters in grid search cv algorithm we’ll get our best fit parameters for final model making.



We have got ‘gini’ for ‘criterion’, ‘3’ for ‘min\_samples\_split’ and ‘best’ for ‘splitter’ parameters as best fit parameters. We’ll use this parameters in our final model making.

**Final Model Making:**

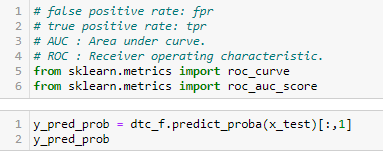


After using the best parameter of grid search cv also we found that we have got accuracy of **83.78%** which is similar to earlier Decision Tree Classifier model, so we can say that’s the maximum accuracy we can get in Decision Tree Classifier model.

We can conclude that now Decision Tree Classifier model is our final model with the accuracy score of **83.78%.**

**AUC- ROC CURVE MAKING:**

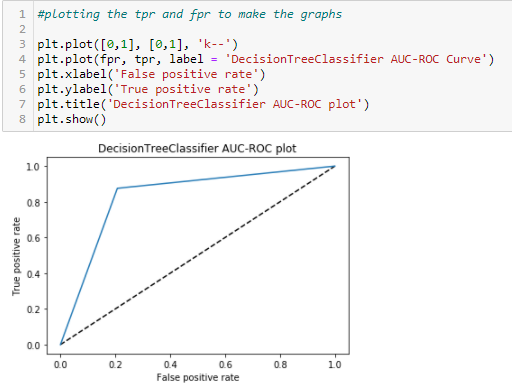
The Area Under the Curve (AUC) is the measure of the ability of a classifier to distinguish between classes and is used as a summary of the ROC curve. The higher the AUC, the better the performance of the model at distinguishing between the positive and negative classes. AUC - ROC curve is a performance measurement for the classification problems at various threshold settings. ROC is a probability curve and AUC represents the degree or measure of separability. We can say that the Higher the AUC, the better the model is at distinguishing between wine with the good quality and with bad quality or not good quality. For auc-roc curve making also we need to import various other libraries as shown below.



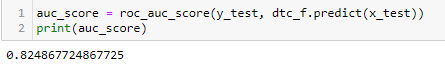
roc\_curve, roc\_auc\_score needs to be imported from the sklearn library.

Fpr – false positive rate

Tpr – true positive rate



**Observation:** From the above Auc-Roc Curve, we can say that the curve auc-roc is having a very sharp curvature indicating very good model building. And the area under curvature is of value **82.48%** which is a very good score.



**SAVING THE MODEL:** We have saved our final model having accuracy of **83.78%** by model name ‘Vaibhav\_Insurance\_Claim\_Fraud\_Detection\_Project\_Model.pkl’.

**CONCLUDING REMARKS:**

**Conclusion:** Our model is very well built and having accuracy of **83.78%.**

**File path:** File is saved on github.https://github.com/vaibhav903174/datasci/blob/bdd997d44f6233c09215ae04b447e22605cfcdcd/Insurance%20claim%20fraud%20detection%20project.ipynb

**Thank You**