**Article for**

**Insurance Claims Fraud Detection Project**

Submitted by – Pooja Parakh

Introduction :-

Insurance fraud is a major problem in the United States at the beginning of 21st century. Insurance fraud occurs when an insurance company, agent, adjuster or consumer commits a deliberate deception in order to obtain an illegitimate gain. So Insurance fraud has many categories among them Automobile insurance fraud is the major fraud type.

But now the question is how to predict wheather the insurance climbed is fraudulent or not. For that I have built a Machine learning model which can predict the claim is fraudulent or not. Using various features like **Insured information, insured persons personal details and the incident information totally we have 40 features in the dataset**. So using all these previously known information and analysing the data I have achieved a good model that has **good accuracy**.

Now let’s get into the problem and build a best possible model to predict insurance claim is fraudulent or not.

**Problem Definition:**

Insurance fraud is a very huge problem in the industry and it is very difficult to identify fraud claims or cases. It is an incorrect or misrepresented or false claim by an insured person for financial gain. Insurance fraud can be committed at different levels.

According to a study, estimates approximately of $80 billion in fraudulent claims are made annually in the United States. This includes all lines of insurance. Healthcare fraud alone is estimated to cost Americans $54 billion a year.

The insurance fraud that occurs more frequently, which are staged to claim the insurance money are:

* Auto/motor accidents
* Health insurance
* Theft or burglary
* Motor or car thefts
* Staged home fires

To reduce these fraud claims we need to find whether the insurance claim made is a genuine or a fraudulent one. Machine learning plays a major role in doing so.

This article is basically on ‘Insurance claim-Fraud detection’ that takes you to a step-by-step process to understand the whole Machine learning building process.

**Problem Statement:**

**Business case:**  
Insurance fraud is a huge problem in the industry. It's difficult to identify fraud claims. Machine Learning is in a unique position to help the Auto Insurance industry with this problem.

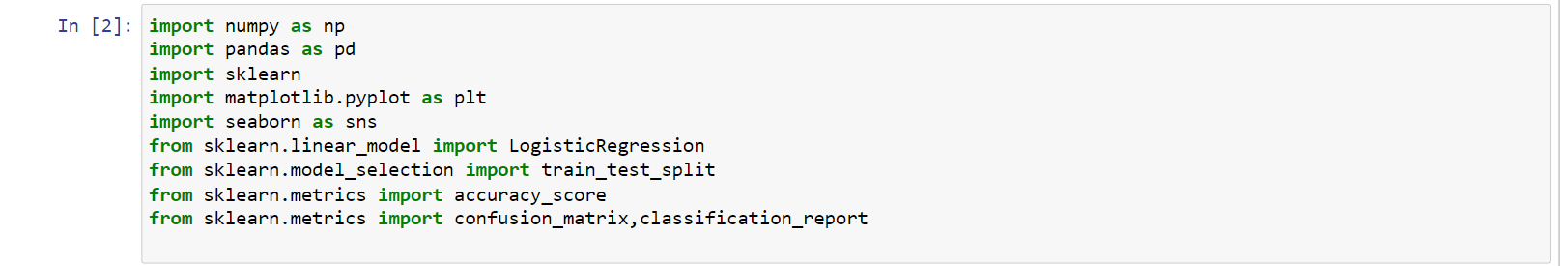
In this project, you are provided a dataset that has the details of the insurance policy along with the customer details. It also has the details of the accident based on which the claims have been made.

In this example, you will be working with some auto insurance data to demonstrate how you can create a predictive model that predicts if an insurance claim is fraudulent or not.

We are going to pick up a data set of auto insurance and perform analysis and predict if an insurance claim is fraudulent or not using Machine learning.

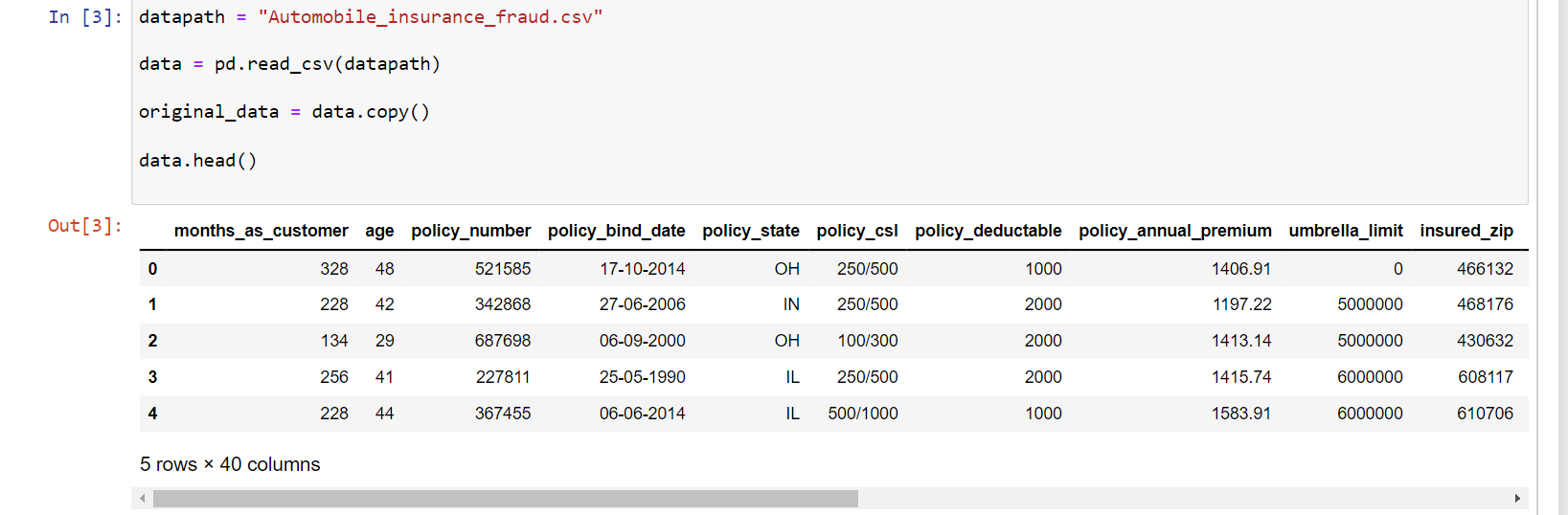
**Importing the data:**

We need to import all the relevant libraries:



We have now imported all the important libraries that we will be needing during analysis.

We need to import the .csv file into the Jupyter notebook.



This data set contains of Independent and Dependent (target) variables.

Independent variable: They are also known as Input variables. These are the input for a process that is being analyzed.

Dependent variable: They are also known as Output or Target variables. They are dependent on Independent variables for their outcome.

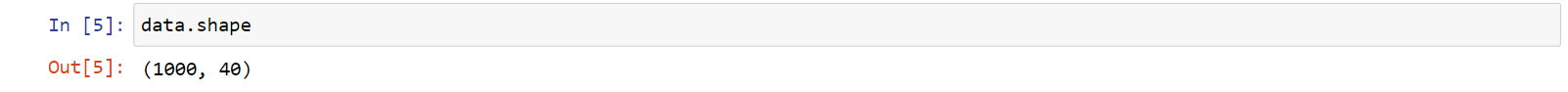
After importing the dataset, display a sample of data. The variables in the dataset are as follows:

* months\_as\_customer,age
* policy\_number
* policy\_bind\_date
* policy\_state
* policy\_csl,
* policy\_deductable
* policy\_annual\_premium
* umbrella\_limit
* insured\_zip
* insured\_sex
* insured\_education\_level
* insured\_occupation
* insured\_hobbies
* insured\_relationship
* capital-gains
* capita\_loss
* incident\_date
* incident\_type
* collision\_type
* incident\_severity
* authorities\_contacted
* incident\_state
* incident\_city
* incident\_location
* incident\_hour\_of\_the\_day
* number\_of\_vehicles\_involved
* property\_damage
* bodily\_injuries
* witnesses
* police\_report\_available
* total\_claim\_amount
* injury\_claim
* property\_claim
* vehicle\_claim
* auto\_make
* auto\_model
* auto\_year
* fraud\_reported
* \_c39

**Data Analysis (EDA)**

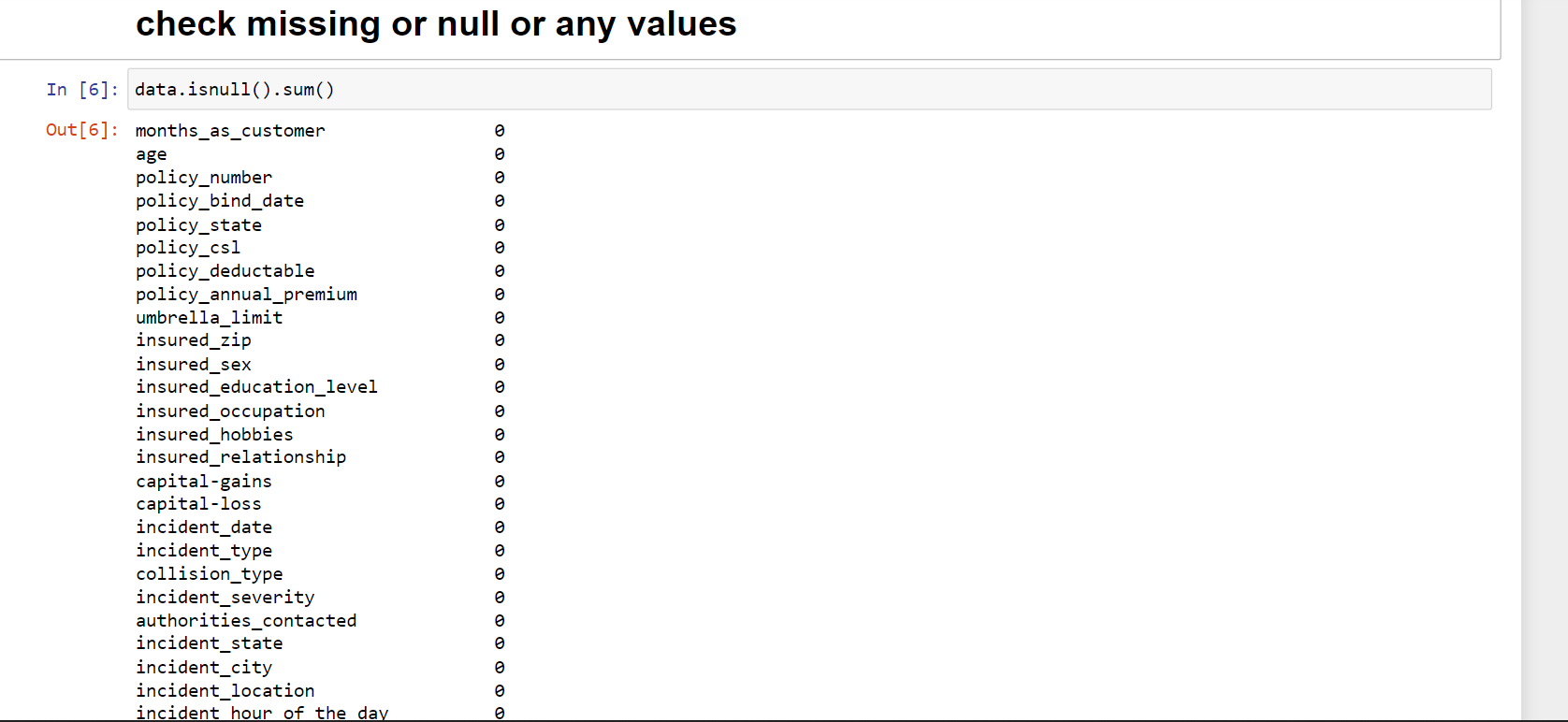
**Now we need to understand the dataset by performing Exploratory Data Analysis.**

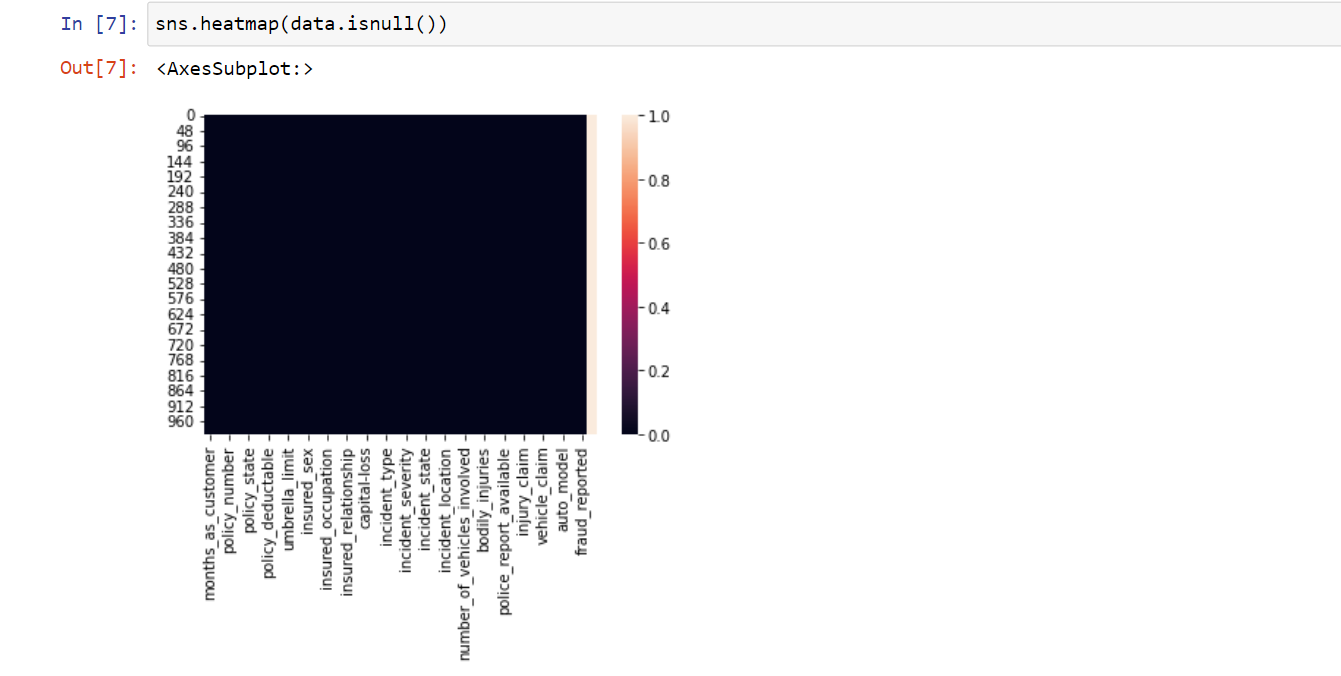
Let’s check the shape of the data set:



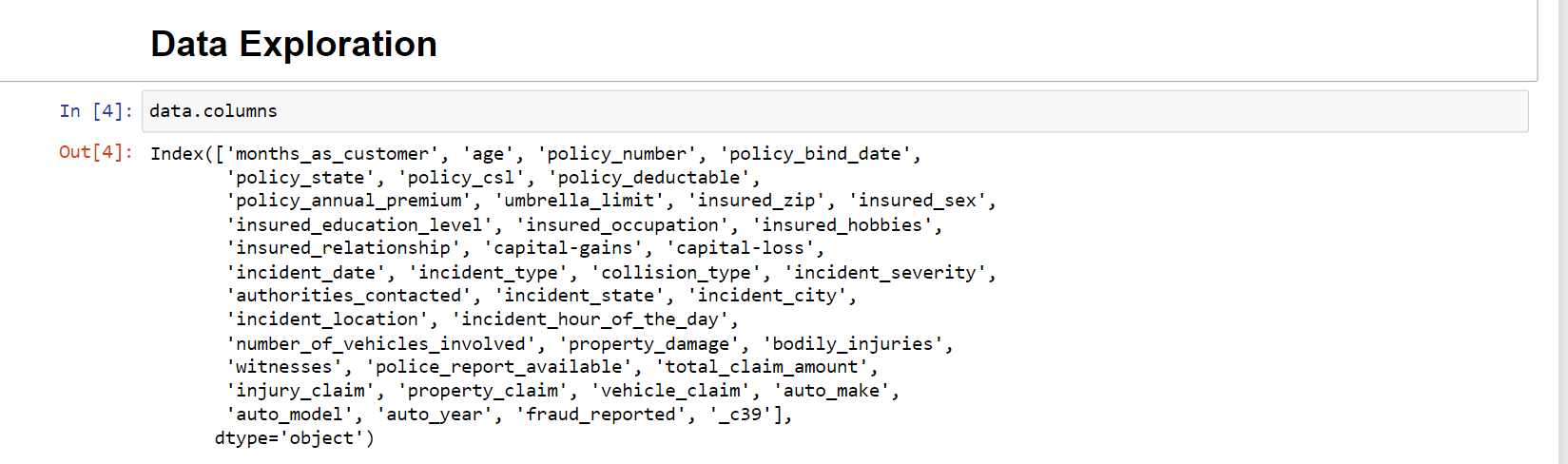
We can see that there are 1000 rows and 40 columns in the dataset.

We cannot have null values in the data as this will affect the data and eventually the predicted result. Therefore we must check for any null values in the dataset.





we can see that there are 1000 null values in ‘data’ column and total no of columns is 1000.

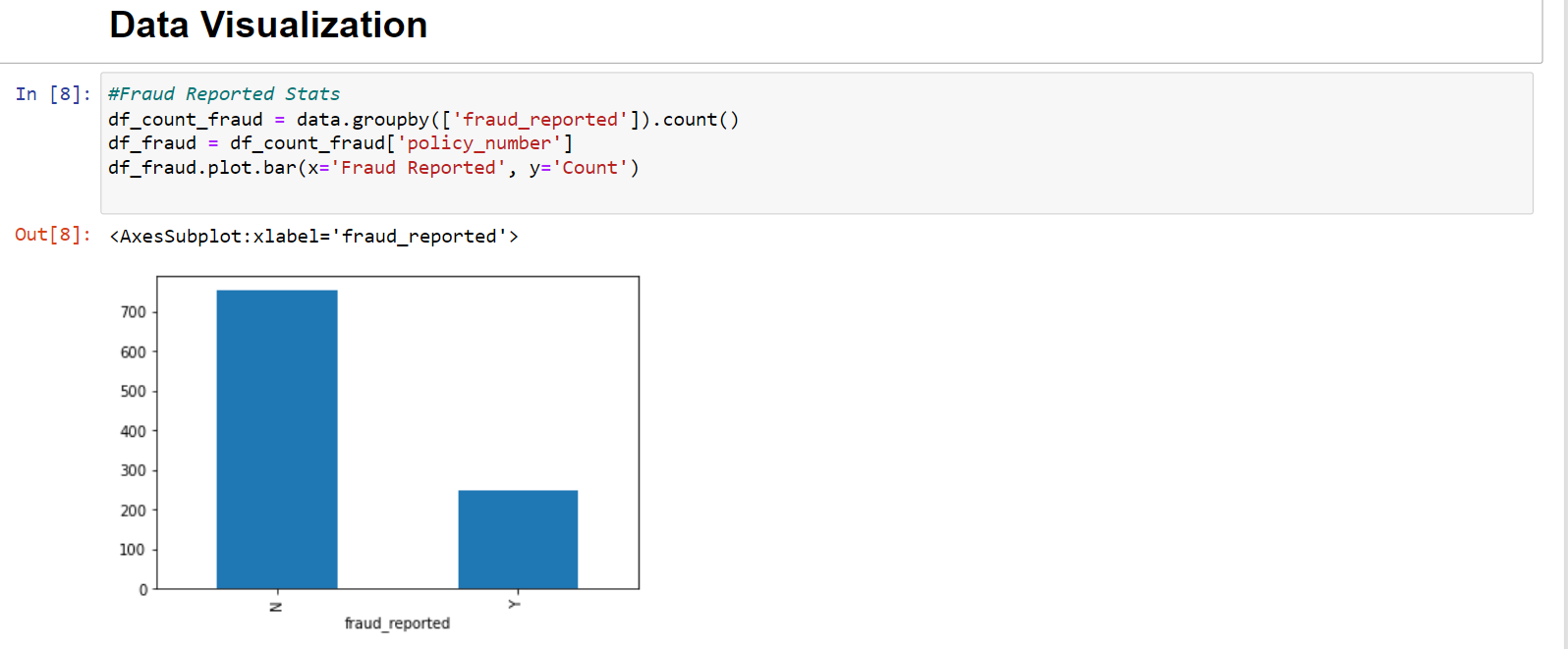


.

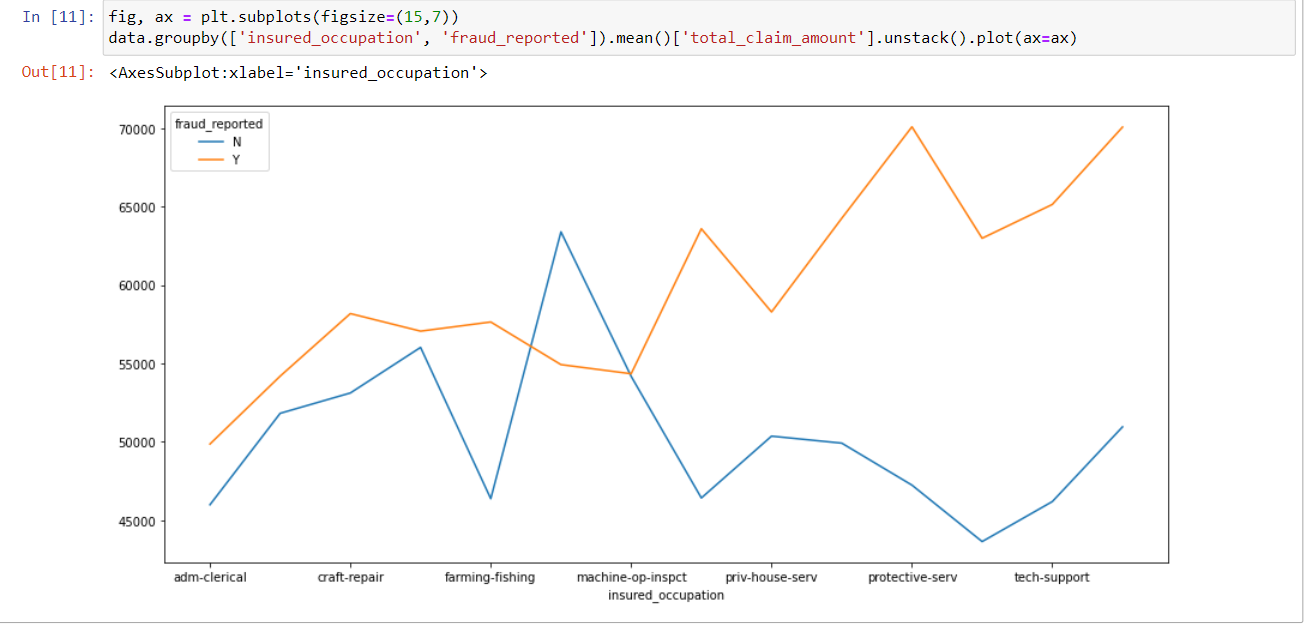
**Data Visualization and EDA Concluding Remarks:**

In the given data, ‘fraud\_reported’ feature is the Target feature or variable. The unique values of this feature are only 2 i.e Y and N (Yes and No), which means it has only two classes. So, as there are only two unique values this is a ‘Classification Problem.’

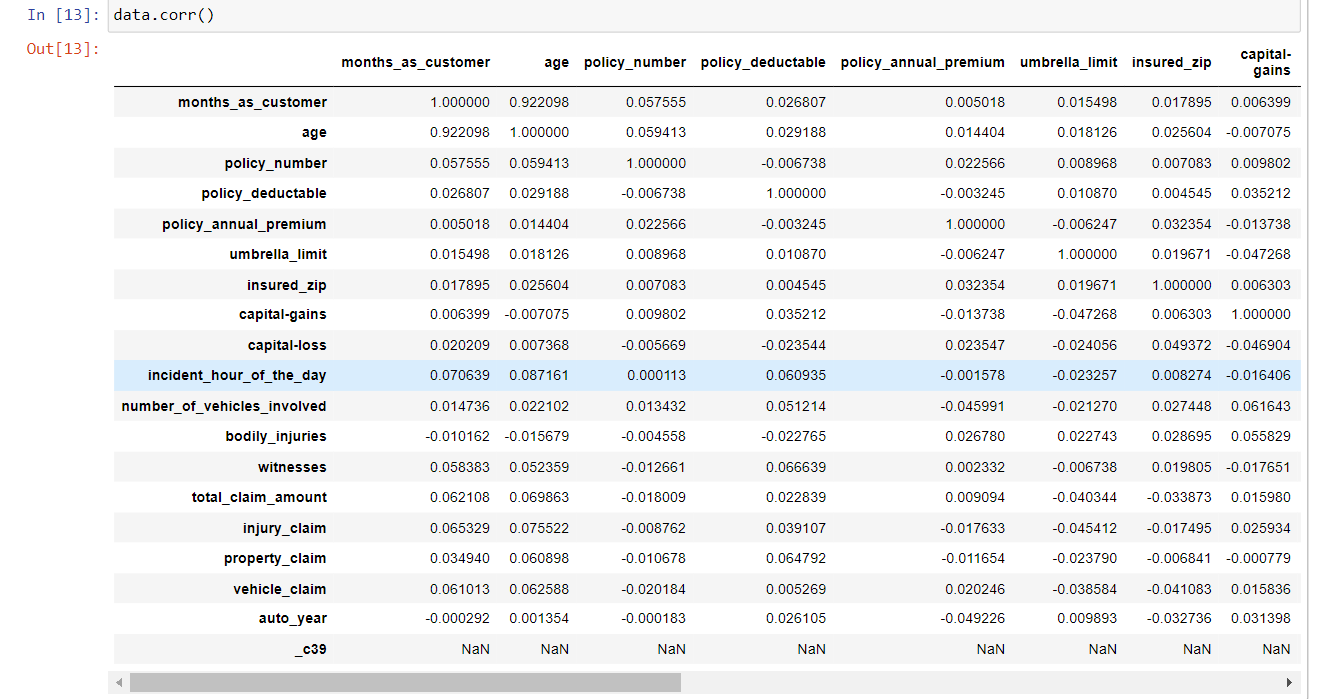
The dependent or target variable has 753 nonfraudulent cases and 247 fraudulent cases which can be seen below in the form of value counts and bar chart.







**Checking the Data Correlation**

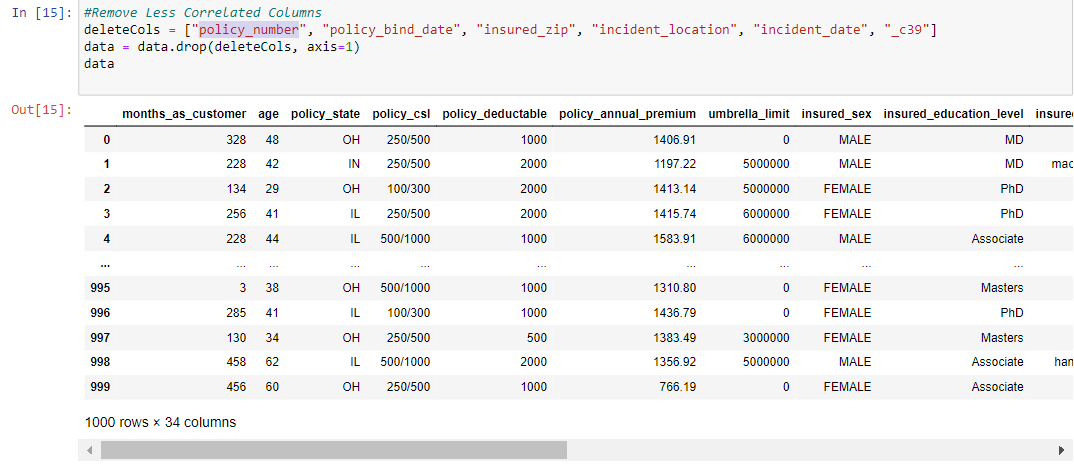


**Data Correlation Visualization**



We can see from the above correlation heat map that correlation is high between month\_as\_customer and age as they both represent no. of months. We can also see there is a high correlation for total\_claim\_amount, injury\_claim, property\_claim, and vehicle\_claim as total\_claim is the sum of injury\_claim, property\_claim and vehicle\_claim. Therefore dropping them will not affect the dataset

**Removing the less correlated Data**



**Pre-processing Pipeline:**

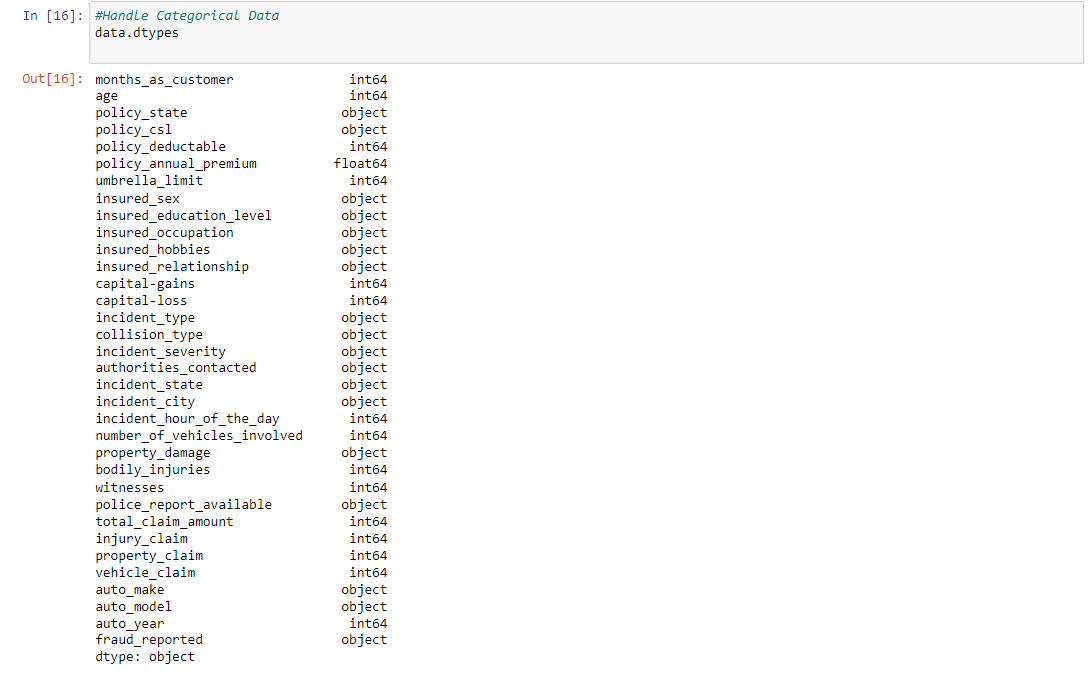
The data set has variables in both object type and numerical type (int and float)

Therefore we have to pre-process the data to move forward.

All the float type or int type variables should be converted into the same scale since the range of values of raw data varies widely, in some machine learning algorithms, objective functions do not work correctly without normalization.

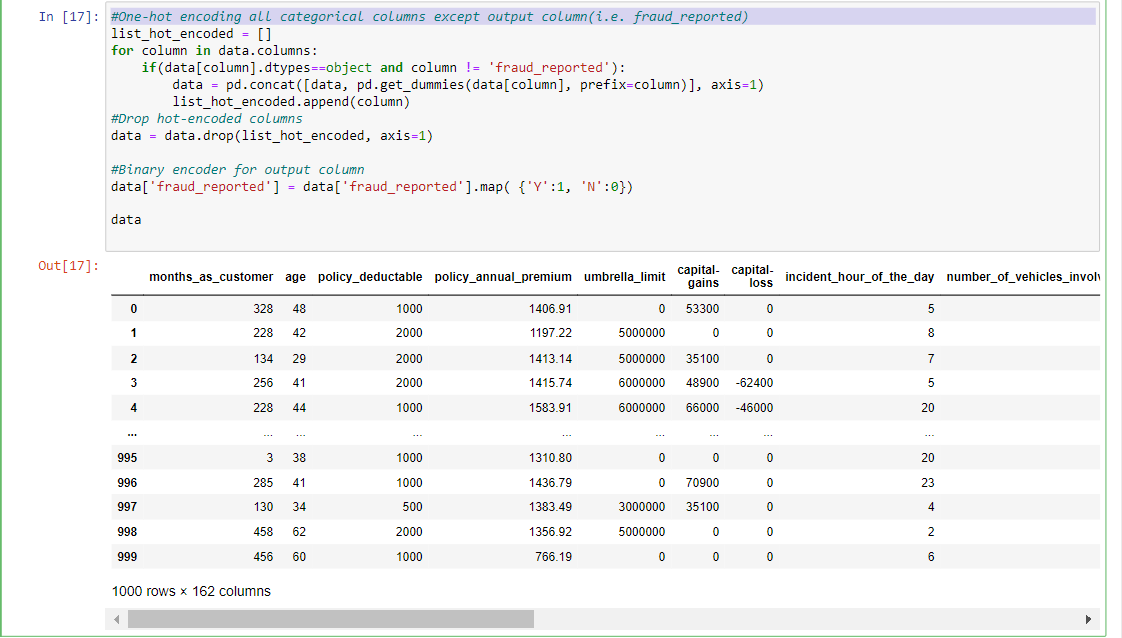
Therefore, normalization is to be performed only on the numerical type (int and float type) variables

**Handling The Categorical Data**



**Encoding the Data**

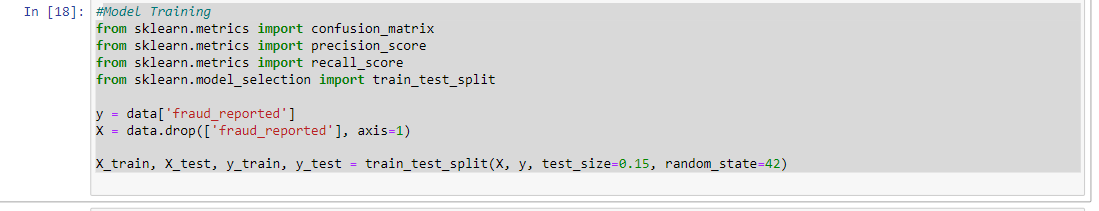
**#One-hot encoding all categorical columns except output column(i.e. fraud\_reported)**



Now the preprocessing is completed. We now have to move to data modeling and prediction.

**Building Machine Learning Models:**

We have to now split the data into independent and target variables.



Here the target variable is fraud\_reported and the rest of them are independent variables.

We have to now split the independent and target variables into training and testing datasets as shown below.

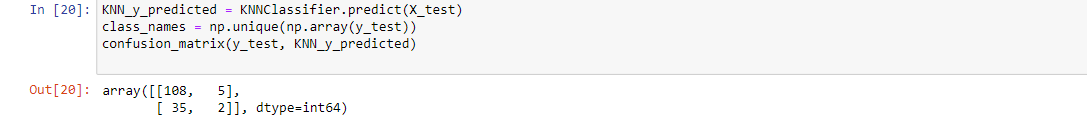
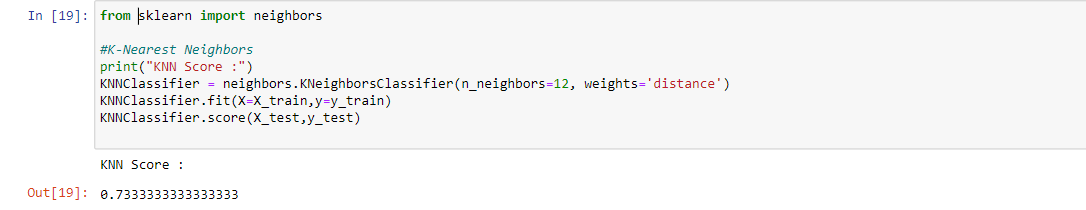
We will use a machine-learning algorithm to learn from the training set and use the model to predict the testing set and compare it with the predicted data with the target testing set to know how close the values. If the error between the predicted and target testing data is less that means the accuracy of the model is high and we can use this model to predict the result of similar datasets.

In this, we have used 6 Machine learning Algorithms

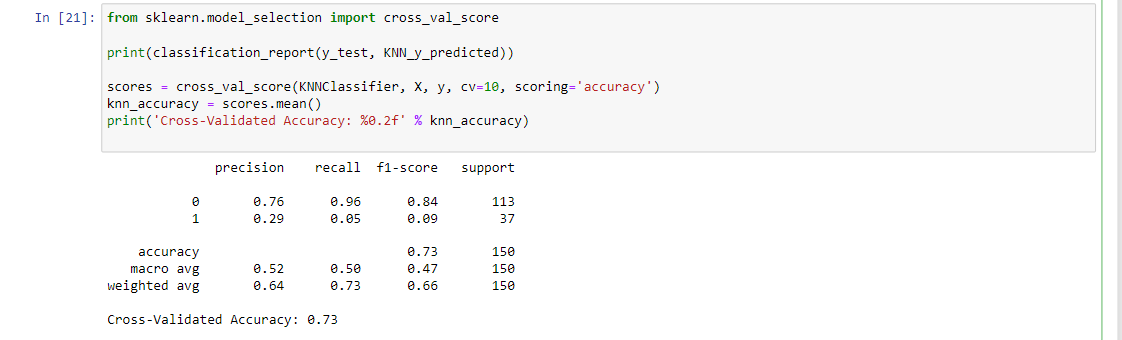
* LogisticRegression
* LinearDiscriminantAnalysis
* GaussianNB
* Supportvectorclassifier
* RandomForestClassifier
* KNeighborsClassifier
* DecisionTreeClassifier

We can train and predict the data using the above 6 ML algorithms and save the model which has the highest frequency.

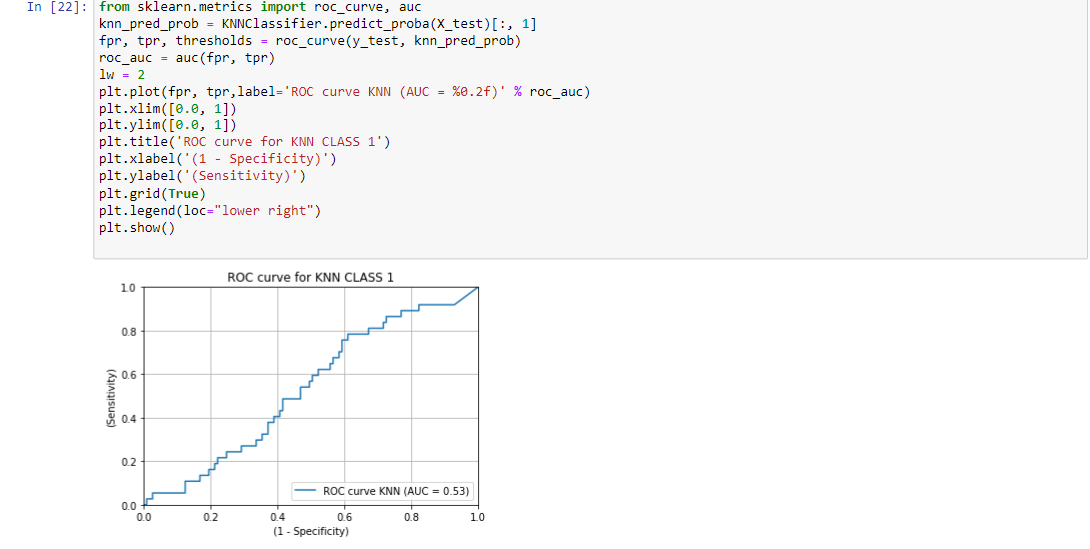
**KNeighborsClassifier**



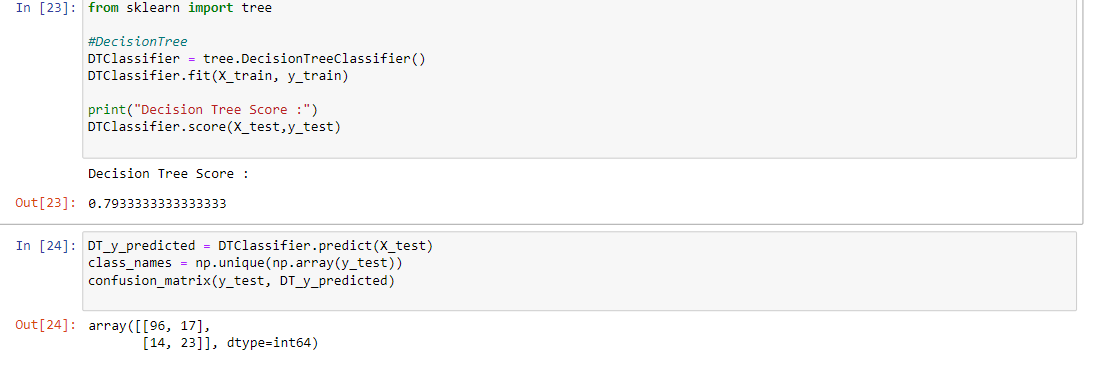
**Cross Validation of KNeighborsClassifier**



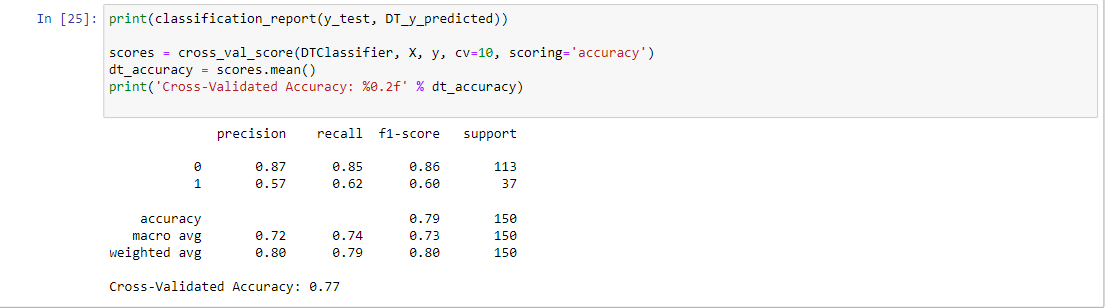
# AUC ROC CURVE **KNeighborsClassifier**:



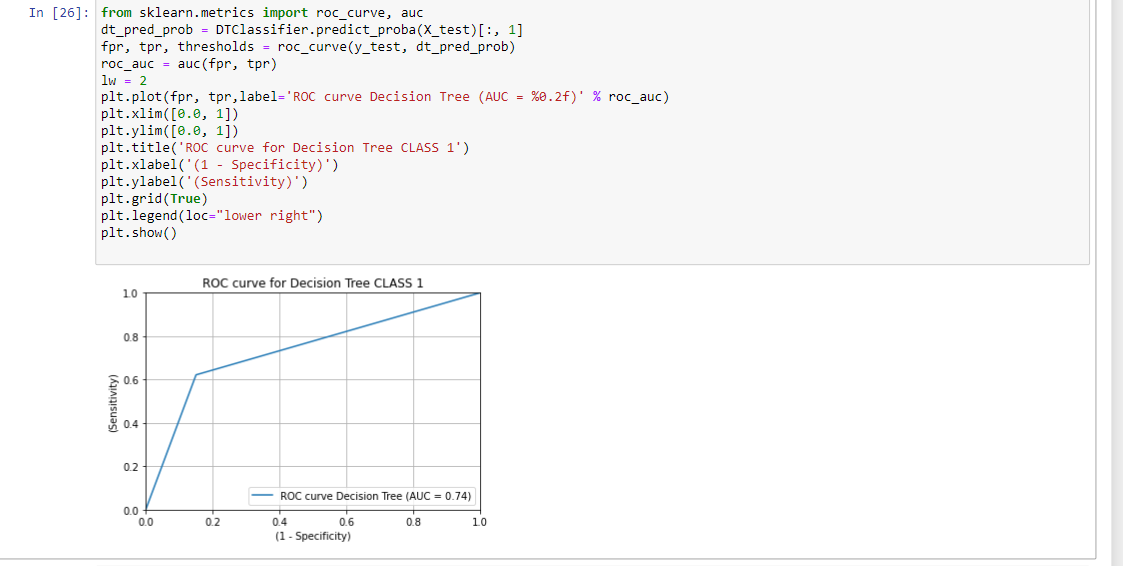
**DecisionTreeClassifier**



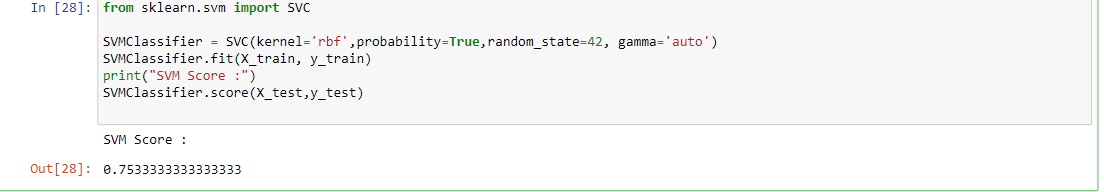
**Cross Validation of DecisionTreeClassifier**

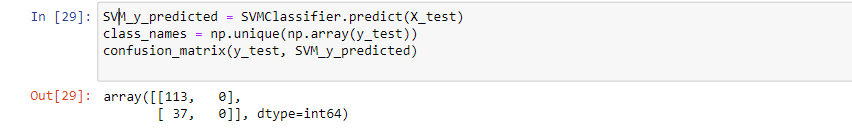


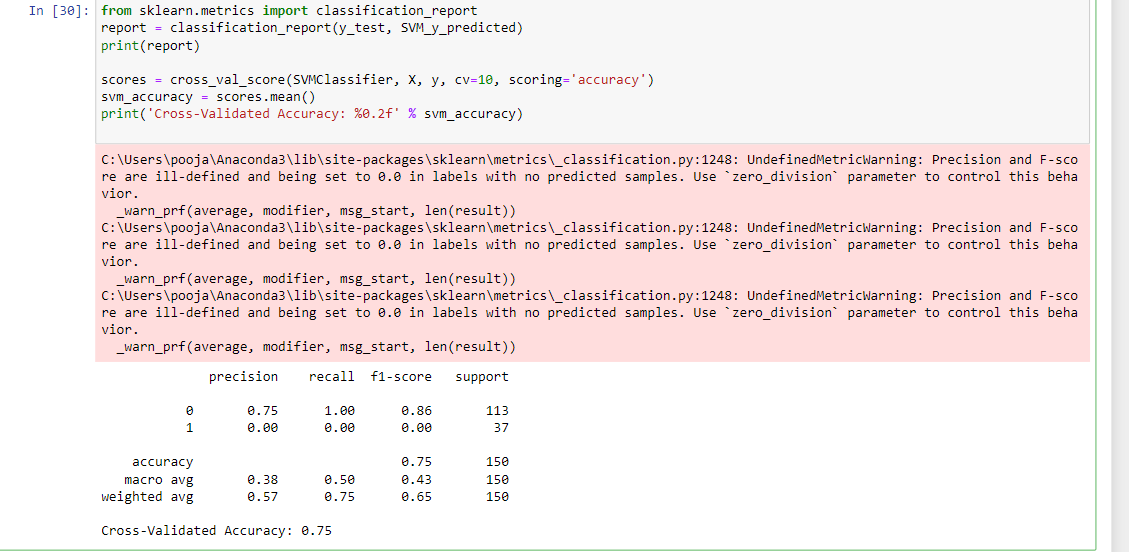
# AUC ROC CURVE **DecisionTreeClassifier**:



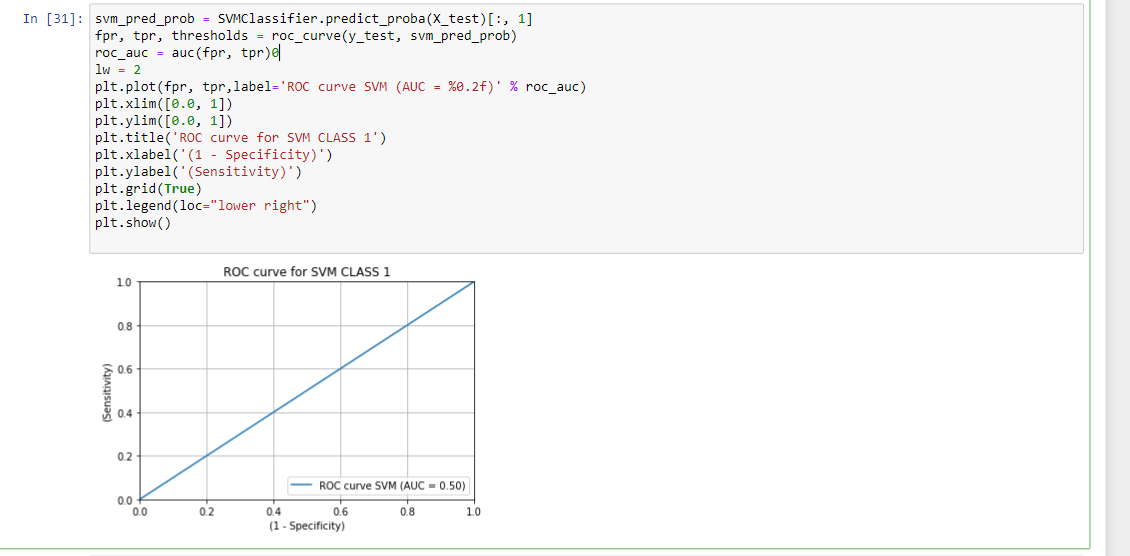
**Supportvectorclassifier**



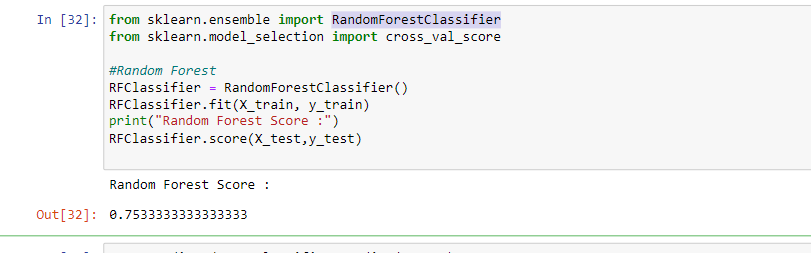
**Cross Validation of Supportvectorclassifier**

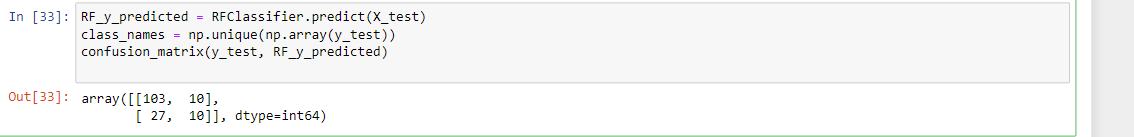


# AUC ROC CURVE **Supportvectorclassifier**:

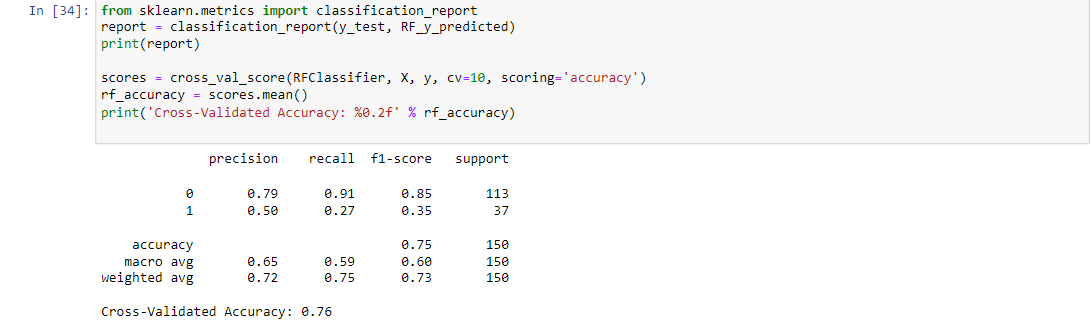


**RandomForestClassifier**





**Cross Validation of RandomForestClassifier**



# AUC ROC CURVE RandomForestClassifier:

# 

# **LinearDiscriminantAnalysis**

# 

# **Cross Validation of LinearDiscriminantAnalysis**

# 

# AUC ROC CURVE **LinearDiscriminantAnalysis**:

# 

# **Naive Bayes Classifier**

# **Cross Validation of Naive Bayes Classifier**

# 

# AUC ROC CURVE ML algorithms

# 

# 

# Accuracy score and Cross Validation score are given in the below table:

|  |  |  |
| --- | --- | --- |
| **Machine Learning Model** | **Accuracy** | **Cross Validation score** |
| LogisticRegression | 0.756 | 0.771 |
| GaussianNB | 0.8 | 0.82 |
| Supportvectorclassifier | 0.75 | 0.753 |
| RandomForestClassifier | 0.75 | 0.76 |
| LinearDiscriminantAnalysis | 0.826 | 0.84 |
| KNeighborsClassifier | 0.73 | 0.73 |
| DecisionTreeClassifier | 0.79 | 0.77 |

According to Cross val score and accuracy we can see that the LinearDiscriminantAnalysis has the least difference between Accuracy and Cross val score, therefore we select LinearDiscriminantAnalysis model.

**Concluding Remarks:**

From the above results of the data modelling and prediction we can see that the LinearDiscriminantAnalysis is performing well as the accuracy score, cross val score and Roc score are good also the maximum of the area under the curve fall under true positive rate. Therefore we can save the model as .obj file so that it can be used to predict the result of the different data sets.

In this kind of problems Pre-processing and data-cleaning is the most important thing. We need to handle both the categorical and numerical data properly and also need to check by building different ML model on the same dataset. We need to check accuracy and cross val score of each model and chose the one which has the best of the same.