**Insurance Claims- Fraud Detection**



Fraud occurs when someone knowingly lies to obtain a benefit or advantage to which they are not otherwise entitled or someone knowingly denies a benefit that is due and to which someone is entitled. According to the law, the crime of insurance fraud can be prosecuted when:

* The suspect had the intent to defraud. Insurance fraud is a "specific" intent crime. This means a prosecutor must prove that the person involved knowingly committed an act to defraud.
* An act is completed. Simply making a misrepresentation (written or oral) to an insurer with knowledge that is untrue is sufficient.
* The act and intent must come together. One without the other is not a crime.
* Actual monetary loss is not necessary as long as the suspect has committed an act and had the intent to commit the crime.

## Types of Insurance Fraud:

* Automobile.
* Medical.
* Property.
* Healthcare.
* Workers’ Compensation and others.

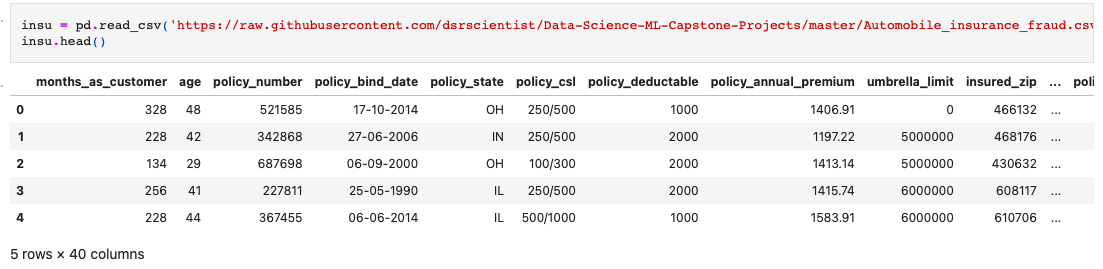
The insurance industry consists of more than 7,000 companies that collect over $1 trillion in premiums each year. The massive size of the industry contributes significantly to the cost of insurance fraud by providing more opportunities and bigger incentives for committing illegal activities.

The total cost of insurance fraud (non-health insurance) is estimated to be more than $40 billion per year. That means Insurance Fraud costs the average U.S. family between $400 and $700 per year in the form of increased premiums.

**Problem Definition:**

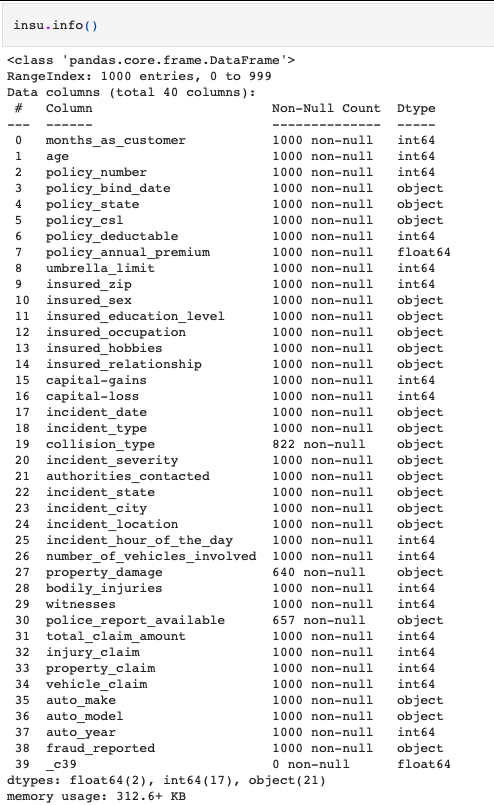
Insurance fraud is a huge problem in the industry. It's difficult to identify fraud claims. Machine Learning is in a unique position to help the Auto Insurance industry with this problem. Fraud in insurance is done by intentional deception or misrepresentation for gaining shabby benefit in the form of showing false expenditures and claim. Data mining tools and techniques can be used to detect fraud in large sets of insurance claim data.

Dataset:



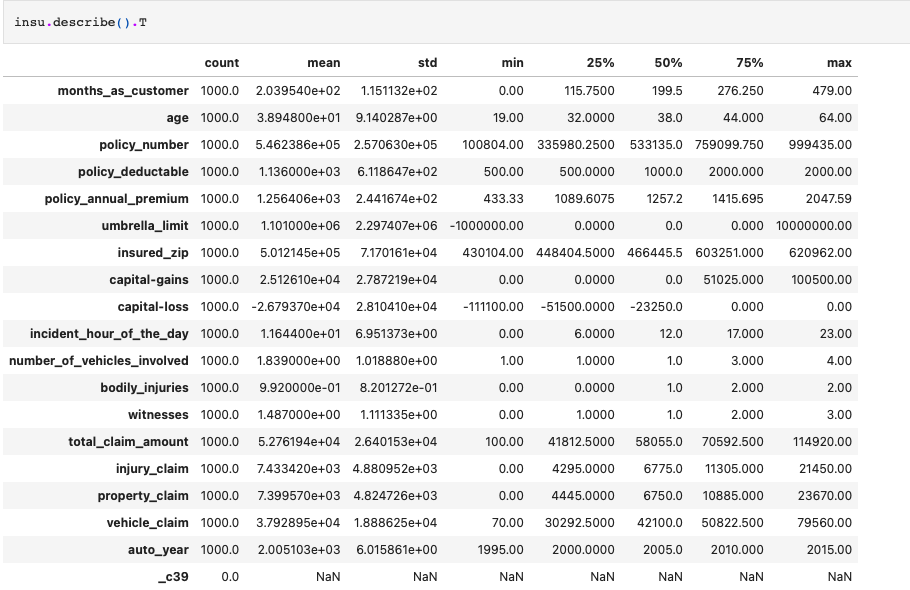
**Data Analysis:**

**Information About the Dataset.**



* RangeIndex: 0 to 999
* Total columns: 40
* dtypes: float64(2), int64(17), object(21)

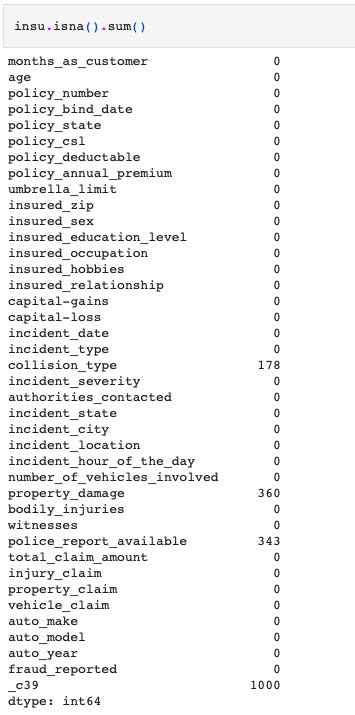
**Description of the Dataset:**



Short description of our dataset

* Counts
* Mean
* Standard deviation
* Minimum
* 1st quantile
* 2nd quantile
* 3rd quantile
* Maximum value of each column.

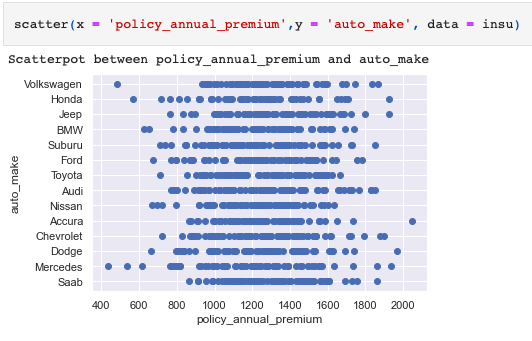
**Null Values:**



There are null values present in our dataset.

**Visualization:**

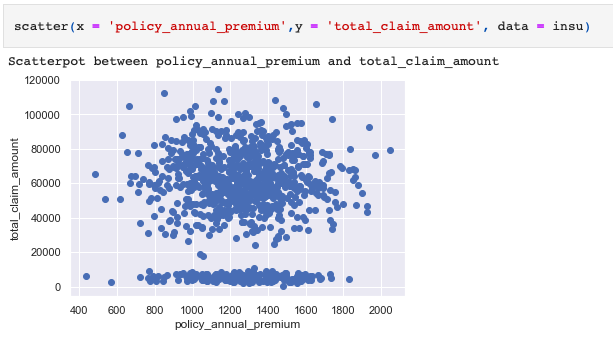
* Scatterplot between policy\_annual\_premium & auto\_make



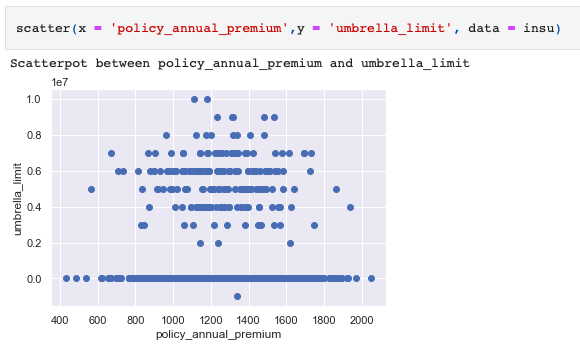
Accura, Dodge, Honda, Jeep, Mercedes is having the highest policy annual premium among all the auto makers.

Nissan, Toyota are the lowest in policy annual premium.

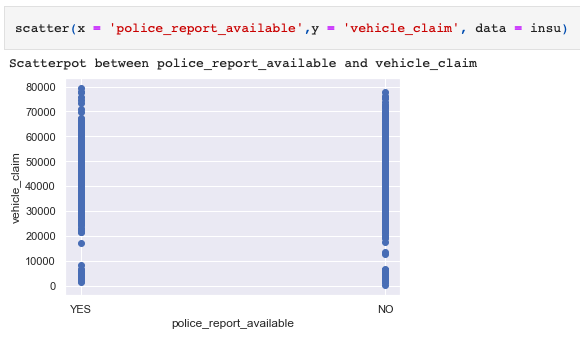
* Scatterplot between policy\_annual\_premium & total\_claim\_amount



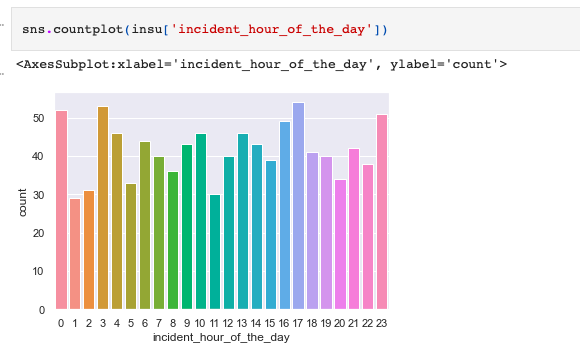
* Scatterplot between policy\_annual\_premium and umbrella\_limit



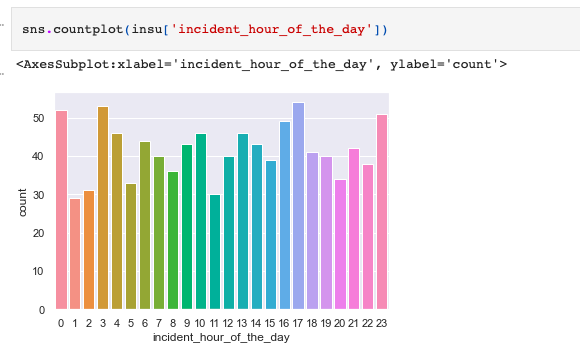
* Scatterpot between police\_report\_available and vehicle\_claim



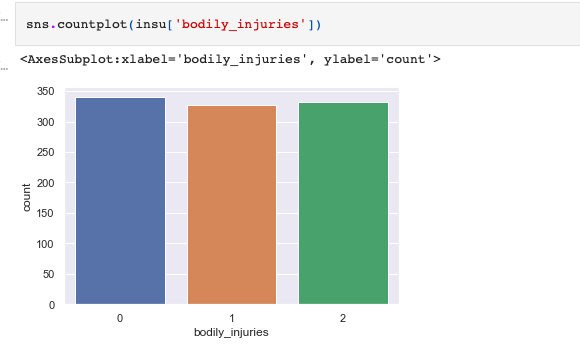
* Countplot of incident\_hour\_of\_the\_day



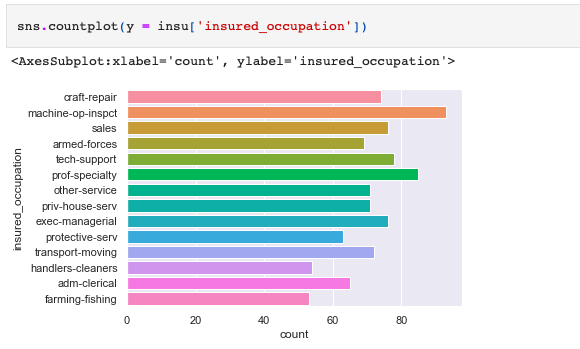
* Countplot of number\_of\_vehicles\_involved



* Countplot of bodily\_injuries



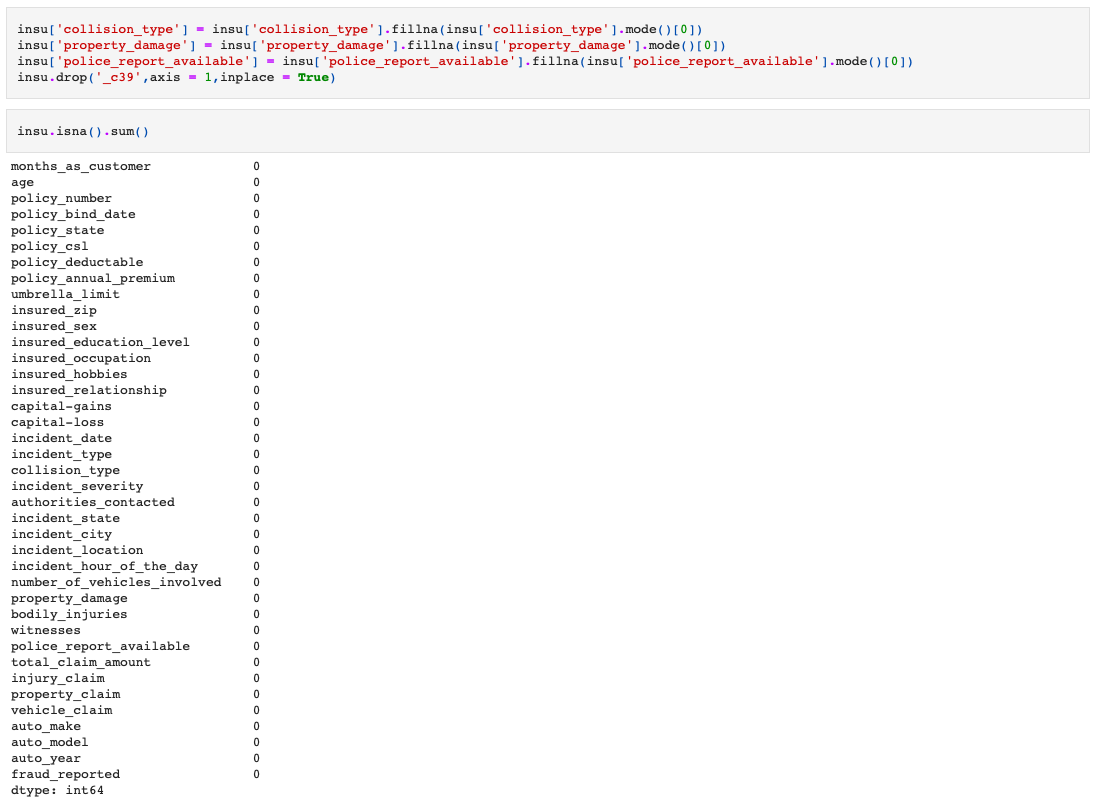
* Countplot of insured\_occupation



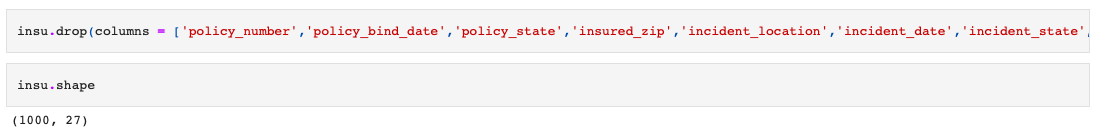
**EDA Concluding Remarks:**

**Data Cleaning**

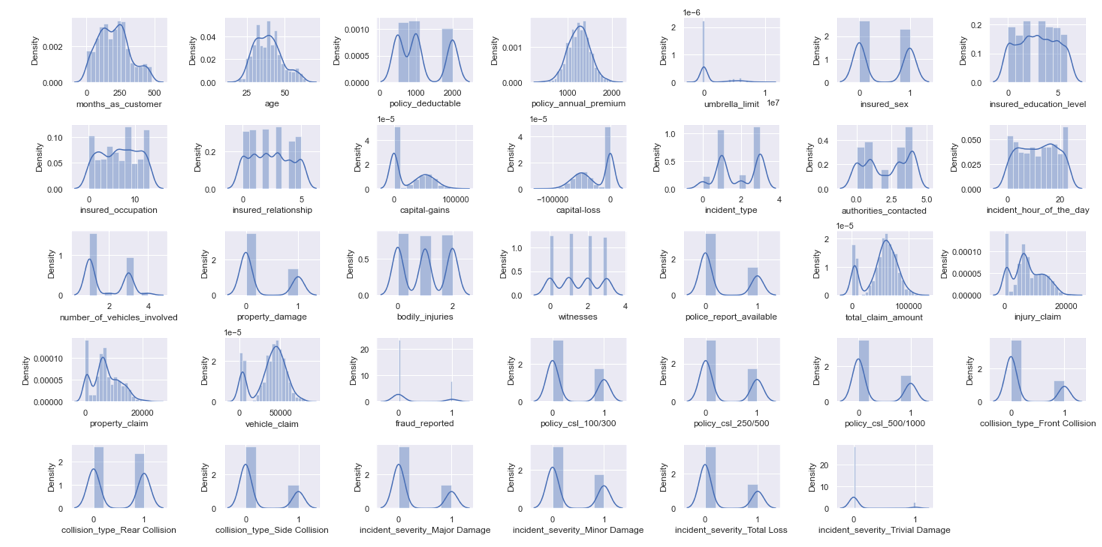
* Filling Null Values & Dropping Column with Complete Null Values.



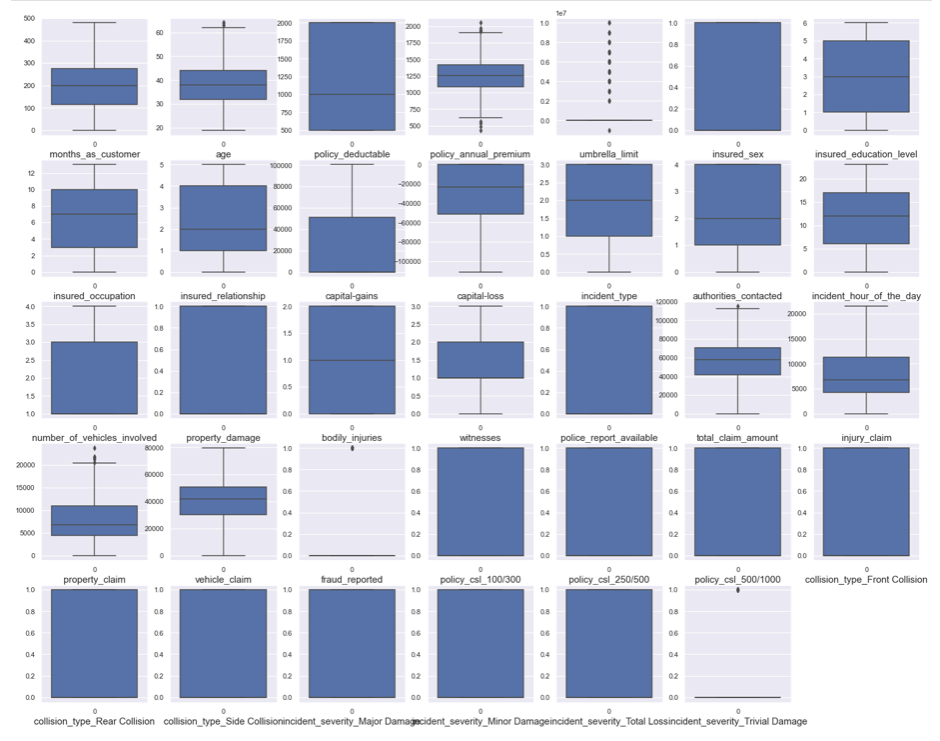
* Dropping Unwanted Columns.



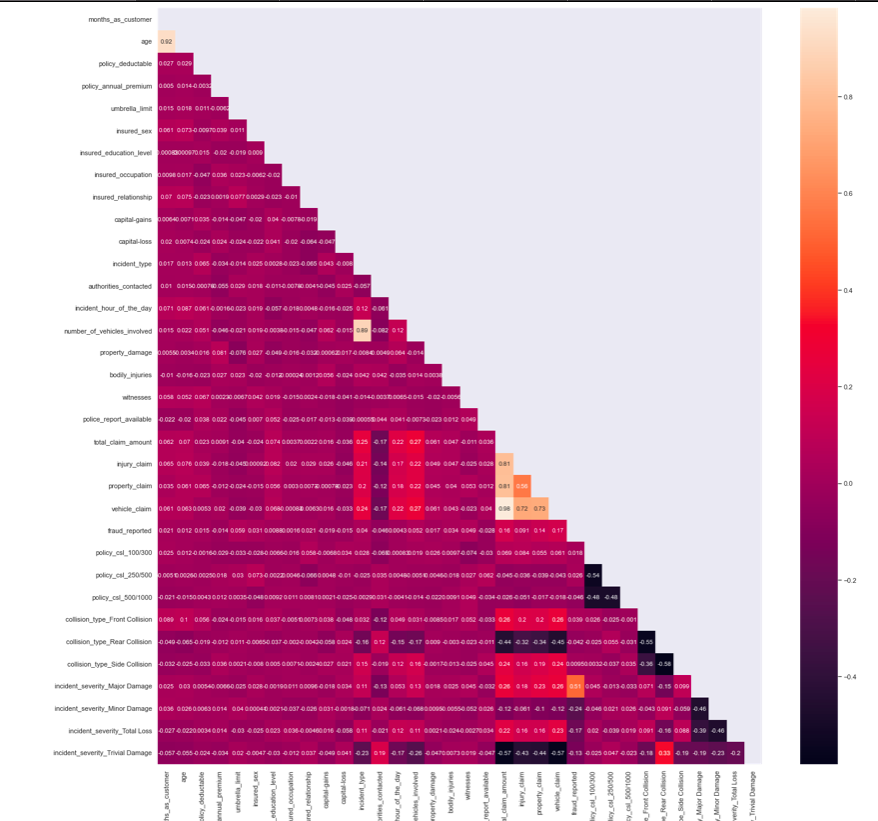
* Normal Distribution



* Dataset is close to Normal Distribution
* Outliers



* Only few outliers are present in our dataset, we can scale them further.
* Correlation



* Found that:

1. age and months\_as\_customer are correlated with each other.
2. total\_claim\_amount and vehicle\_claim are correlated columns.

We are taking the correlation whoes correlation is greater than 90%, So dropping any of the columns which are correlated.

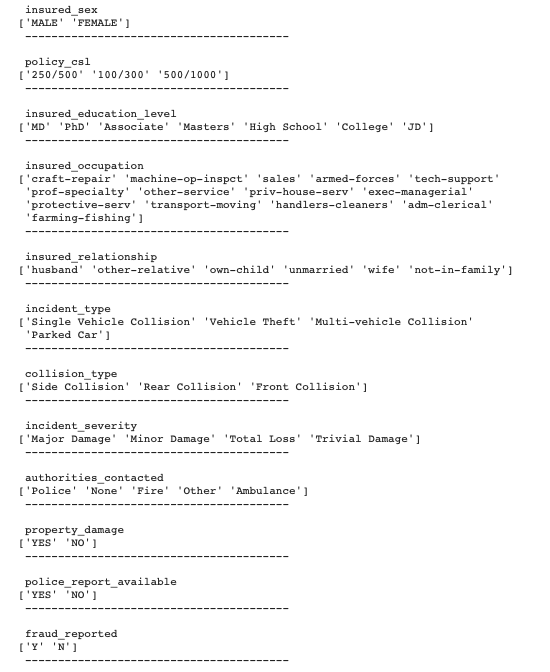
**Pre-processing Pipeline**

**Encoding**

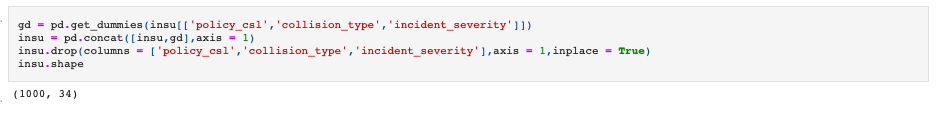
* Object columns



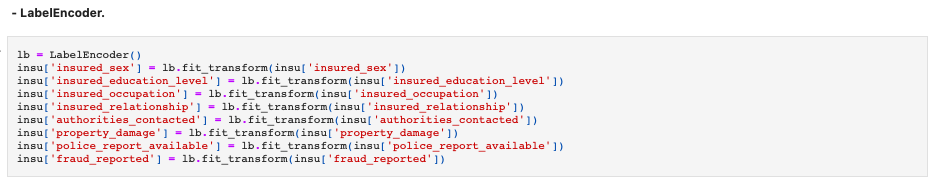
* Unique values of every object column



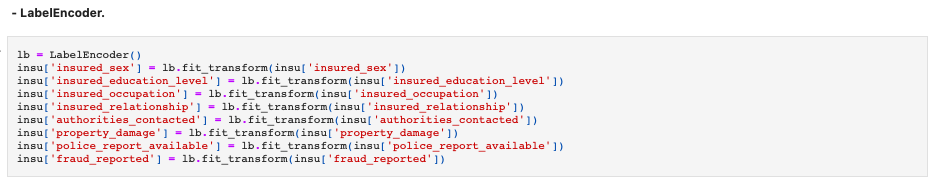
* Applying Dummies to column



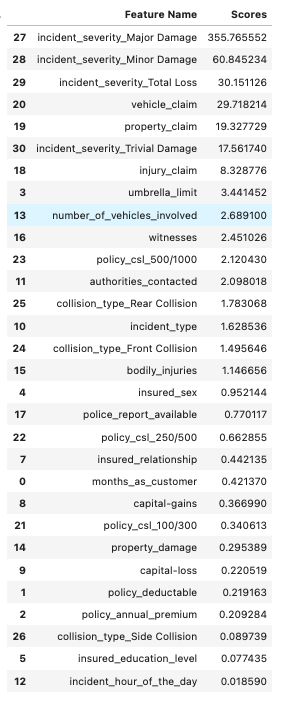
* Applying LabelEncoder



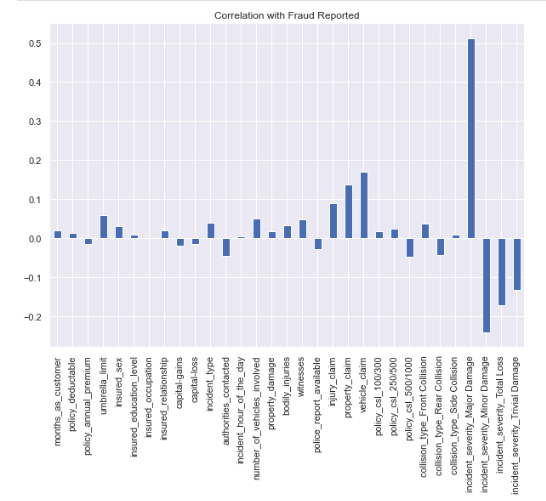
* Replacing Values



* Feature Selection
* Top 30 Columns (Feature Name & Score)

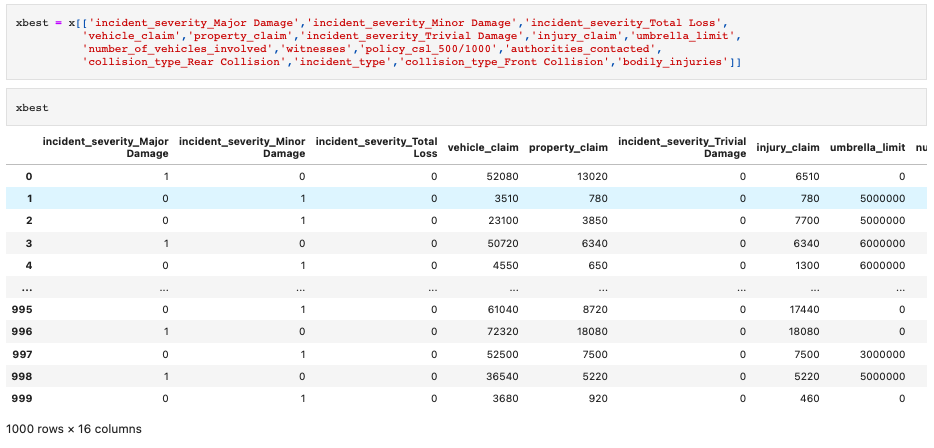


* Graph of correlation with Target Column.

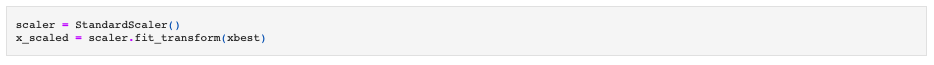


1. Taking top 16 columns according to the score and graph
2. Also taking columns whose score is greater than 1.

* Top 16 Column & Final Dataset



* Scaling the Features (StandardScaler)

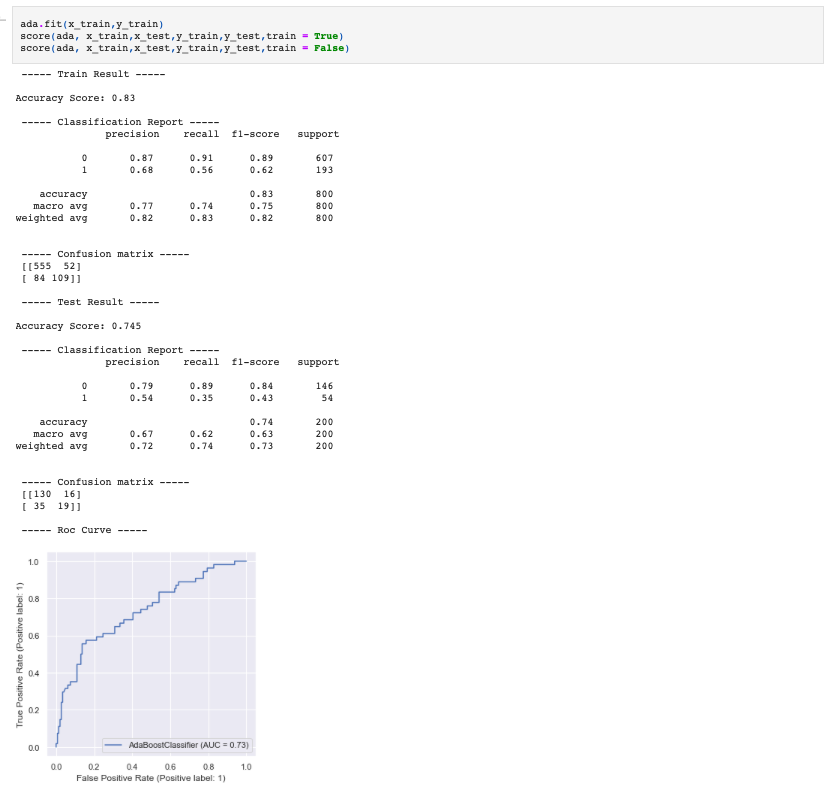


* Train Test Split

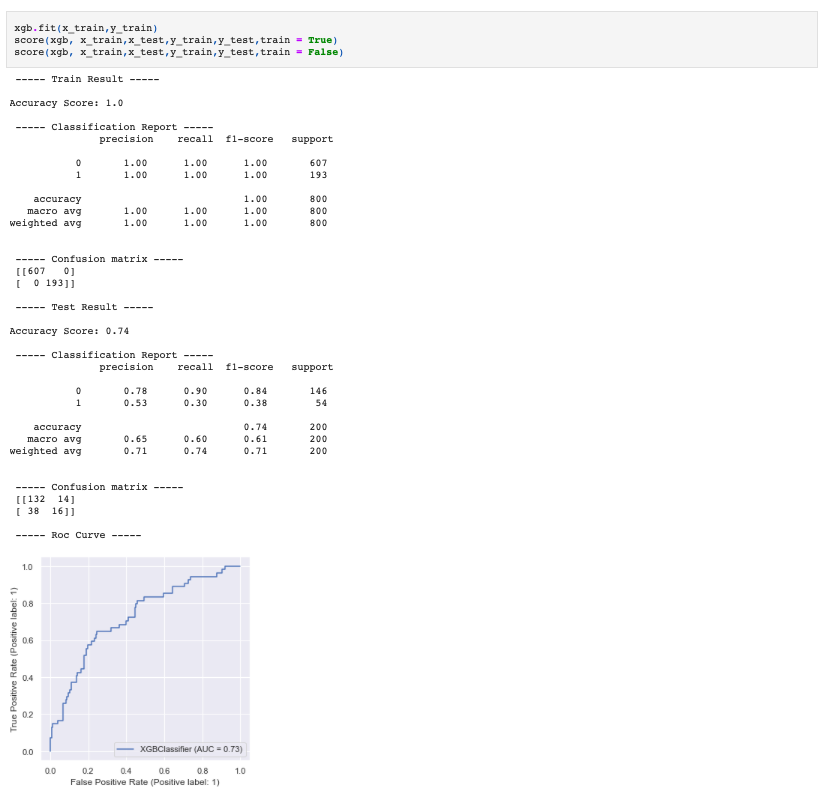


**Building Machine Learning Models**

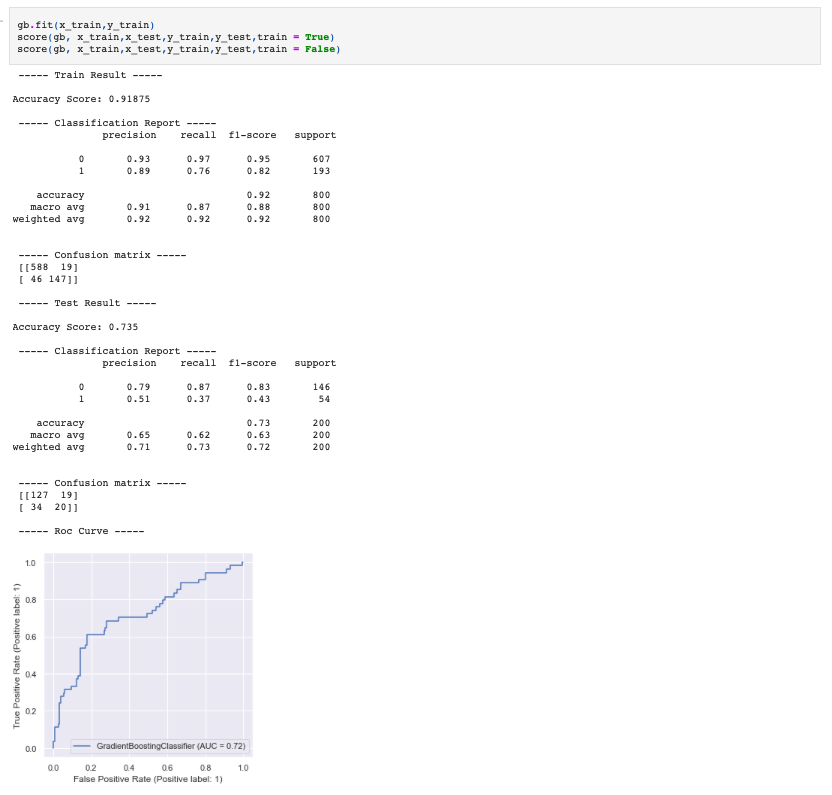
* AdaBoost Classifier



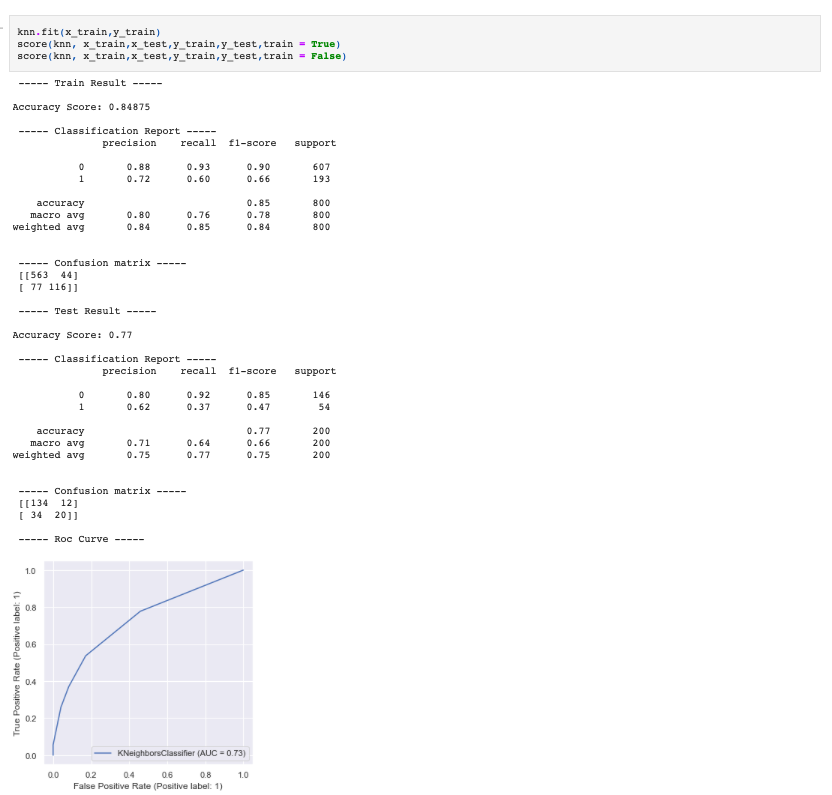
* XGBoost Classifier.



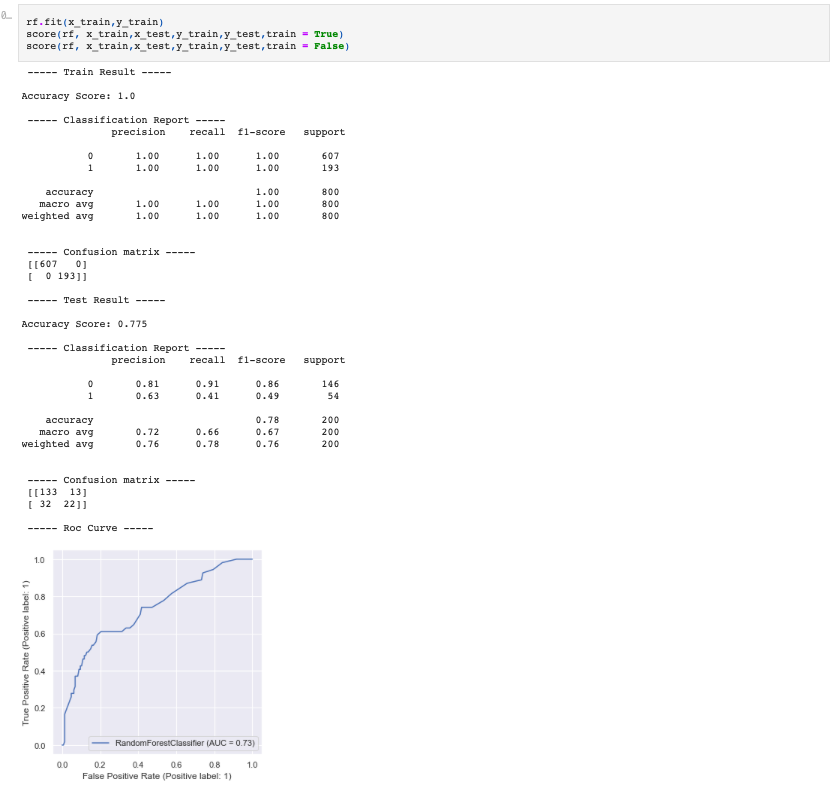
* GradientBoosting Classifier.



* KNeighbors Classifier.



* RandomForest Classifier.

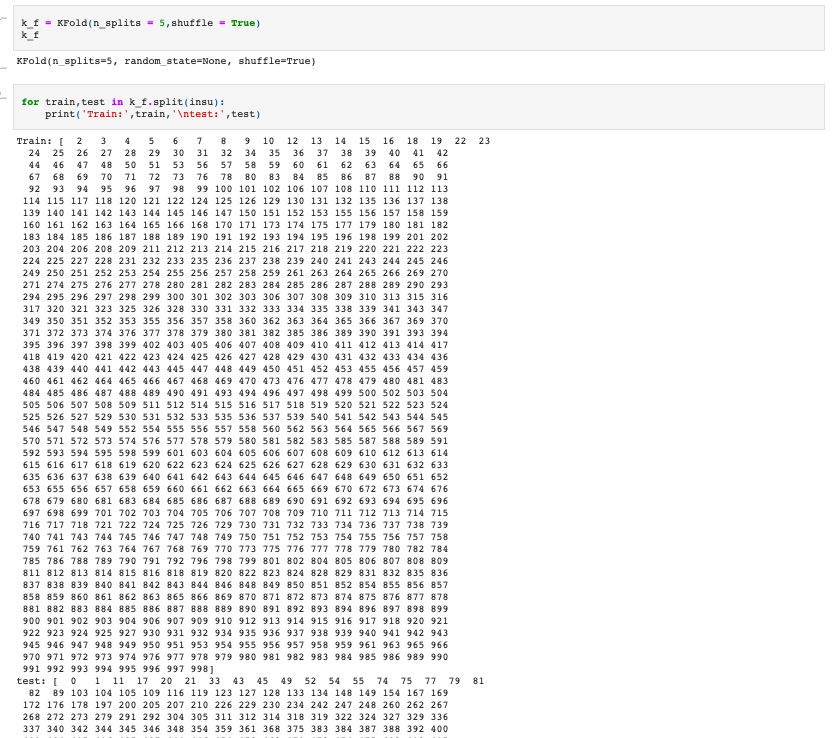


**Concluding Remarks**

* RandomForest Classifier is giving the best score comparatively and all the metrics and curves are in favor of RandomForest Classifier.
* Original vs Predicited



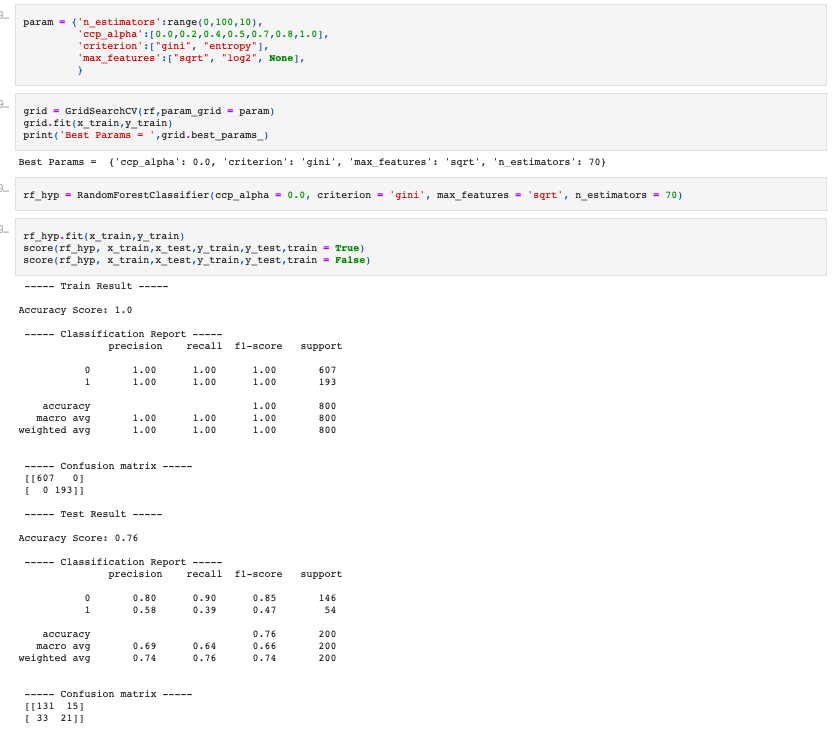
* Cross-Validation



* Cross Validation Score



* cross validation score and model's accuracy score is almost equal, so we can consider that our model isn't overfitting/underfitting.
* Hyperparameter Tuning



* Post tuning result are not better, so taking default parameters only as the best parameters.
* **Library Used:**
* Pandas
* Numpy
* Seaborn
* Matplotlib
* LabelEncoder
* StandardScaler
* SelectKBest
* f\_classif
* Train Test Split
* GridSearchCV
* metrics
* AdaBoostClassifier
* GradientBoostingClassifier
* RandomForestClassifier
* KNeighborsClassifier
* XGBClassifier
* KFold
* cross\_val\_score
* **Project Link:** [**https://github.com/Miteshverma9/Project/blob/main/Projects/Insurance%20Claims-%20Fraud%20Detection.ipynb**](https://github.com/Miteshverma9/Project/blob/main/Projects/Insurance%20Claims-%20Fraud%20Detection.ipynb)