**Insurance Claims- Fraud Detection**

**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:**

The purpose of an Insurance is to provide protection against the risk of any financial loss.

Insurance is a form of risk management in which an insurer agrees to take the risk of the insured entity against future events, uncertain loss due to Tsunami, earthquake or damage against the vehicle or personal property.

The main aim of the project is to predict the Automobile insurance claim is fraudulent or not.

We will be working with some auto insurance data to demonstrate how we can create a predictive model that predicts if an insurance claim is fraudulent or not.

**Data Analysis (EDA)**

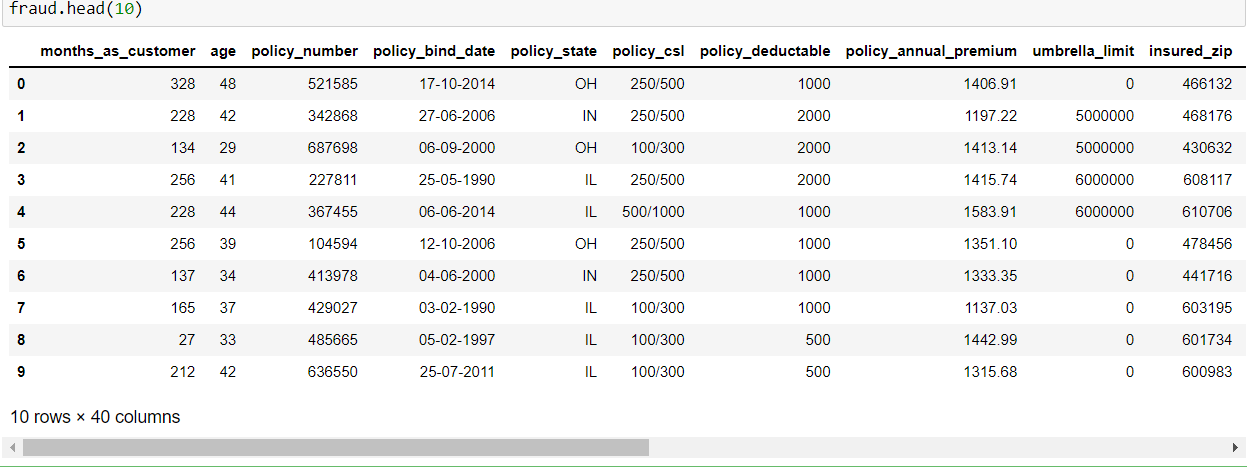
We need to import all the relevant libraries:

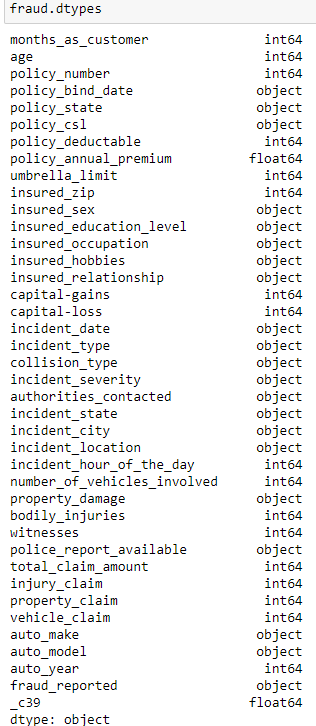
We need to import the .csv file into the Jupyter notebook.

We will follow standard procedure for Data cleaning

* Read & preview the dataset.
* Variable identification # looking the input data # what will be the output variable
* Univariate analysis - tacking small variable and plot bar chart and finding histogram
* Bivariate analysis - tacking two column or two variable and looking the relationship between them, and also find correlation and covariance within two variable.
* Handling and removing null values- missing data.
* Handling Categorical Variable.
* Finding and removing outliers.

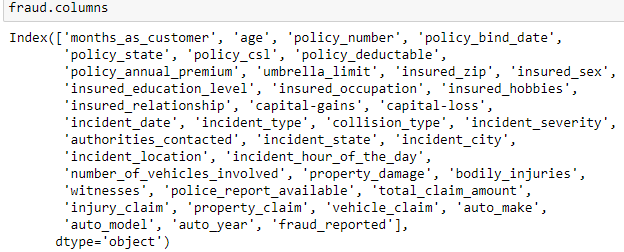
Preview the dataset and check the datatypes of all feature which present in dataset





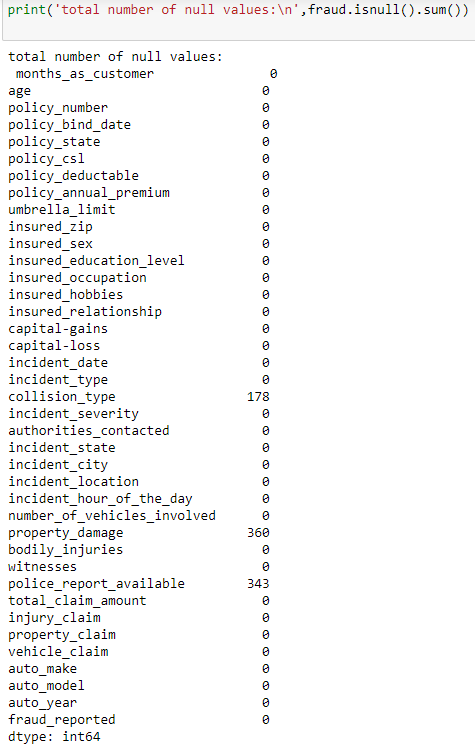
We have several string (categorical) columns in our dataset, along with some integers and floats.

We need to drop unnecessary column and then check for all the columns and shape of dataset.

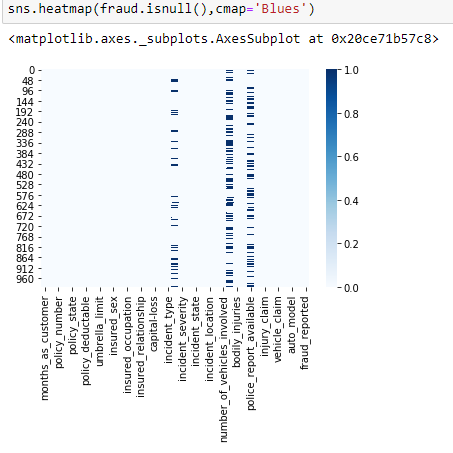




We will check for any null and nan values whether it is available in the dataset or not.



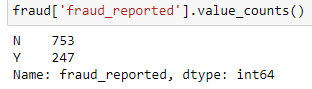
We plot heatmap to visualize null values



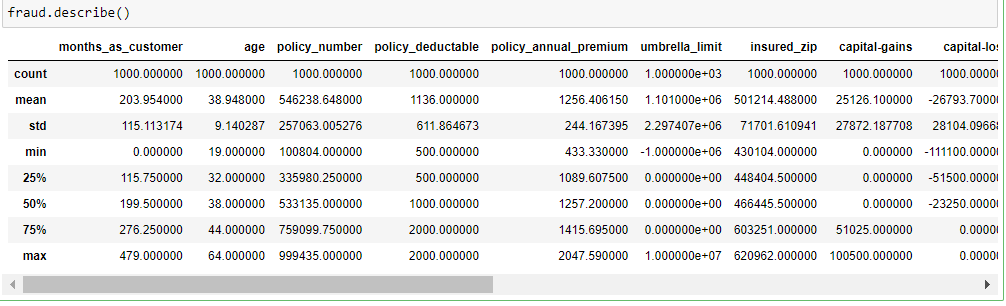
We will check how many unique values presents in each column.



We Check the value count of class are available in to fraud \_reports



We will use describe function, it provides statistics summary of continuous variable.



**Key Observation: -**

* The automobile insurance claim Fraud dataset has 1000 rows with 39 features.
* we checked above in the customer feature has total 9134 unique id of customers that means no customer repeat so it will not help in claim amount prediction.
* We observed that some categorical columns have 900+ unique values. We will remove these columns from our dataset because these columns not help in model accuracy. below present those columns.

1. policy number (1000)
2. policy bind date (951)
3. insured zip (995)
4. insured location (1000)
5. incident date (60)

* the heatmap and data information confirms, there is missing values presence in collision\_type, property\_damage and police\_report\_available columns in the datasets.
* in which "fraud\_reported" is our Target variable.

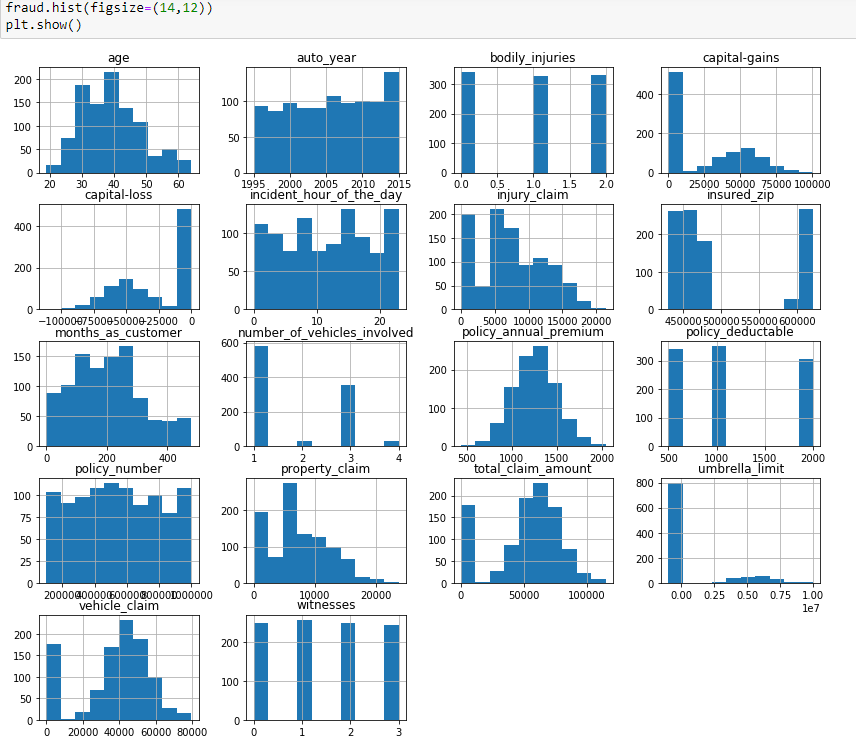
**Statistic summary observations: -**

* the mean is more than median (50th percentile) in all columns except age and capital loss columns.
* there is a large difference in 75th -- - percentile and max in the Total Claim Amount, property\_claim, capital-gain, policy\_number and months\_as\_customer columns in the dataset.
* the 1 and 2 observations suggest that there is outlier present in these five columns.

Now let’s do some plotting to know how the data columns are distributed in the dataset

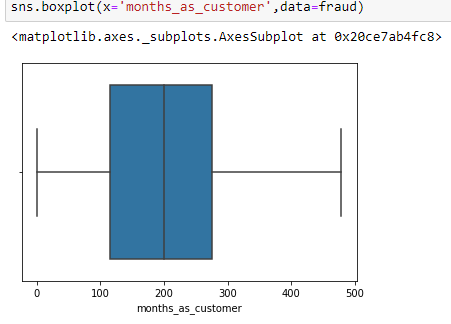
This will be univariate data analysis by graphical representation.

Let's plot histogram for numerical variables to check all feature in the dataset whether normal distribution or not.

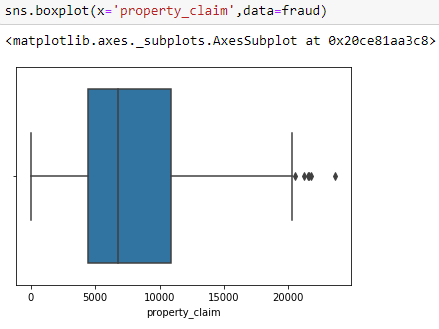


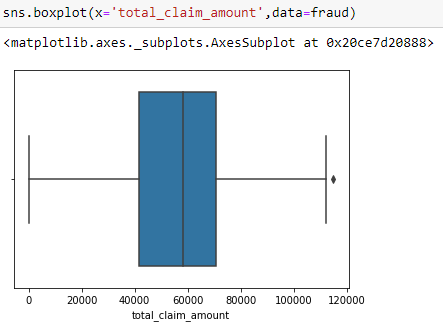
The histogram show all columns in form of normal distribution except umbrella limit in the dataset.

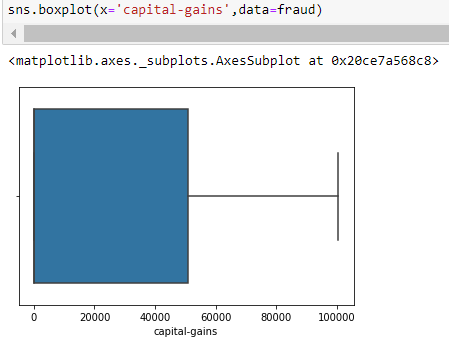
Now let’s plot Box Plot for understanding the distribution and observe the presence of outliers.



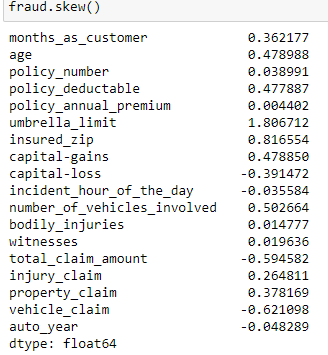
no outliers presence in this column



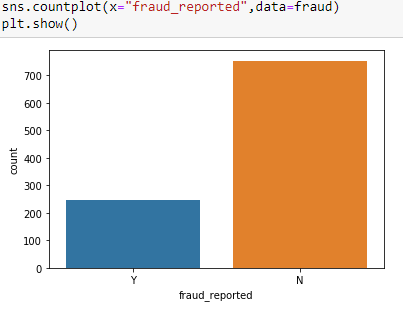




Now let’s check the skewness of the dataset using skew function



Now we will plot the countplot of 'fraud reported’ feature to understand distribution of the data.

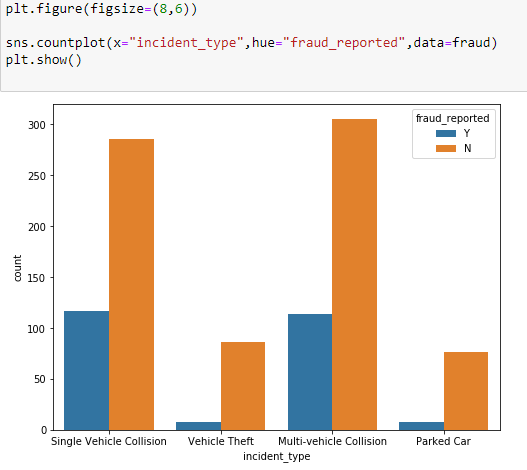


In the given data, ‘fraud\_reported’ feature is the Target feature or variable. The unique values of this feature are only 2 i.e Y and N (Yes and No), which means it has only two classes. So, as there are only two unique values this is a ‘Classification Problem.’

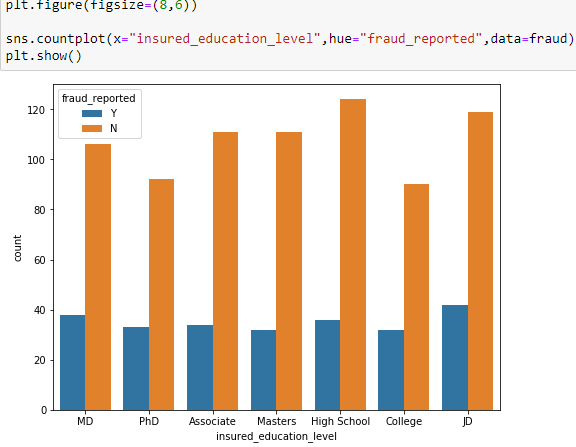
The dependent or target variable has 753 nonfraudulent cases and 247 fraudulent cases which can be seen below in the form of value counts and bar chart.

From the graph we can say that it is imbalanced dataset because out of 1000 auto insurance claim only 250 insurance claims are fraud.

Now let’s check the count of "Fraud\_reported" considering with "incident\_type"

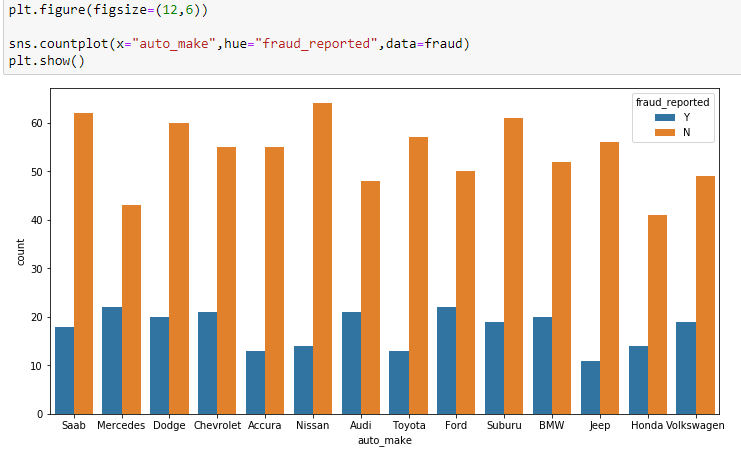


Now let’s check the count of "Fraud\_reported" considering with "insured\_education\_level"

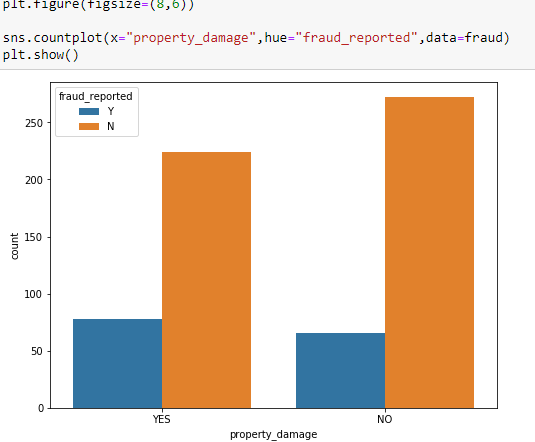


According to the graph we can say that auto claim fraud reported high where insured education is "Jd"

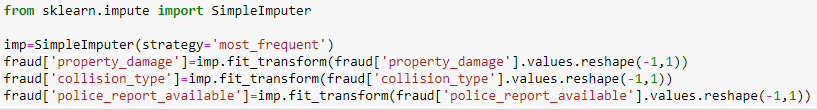
Now we will check the count of "Fraud\_reported" considering with "auto\_make"



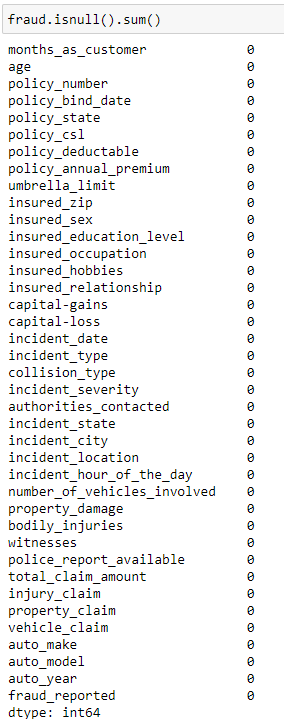
Now let’s check the count of "Fraud\_reported" considering with "property\_damage"



Now let's fill missing values first using mean, median and mode method. We fill missing values with most frequent values in Property \_damage, collision\_type and police\_report\_available feature using SimpleImputer method.



To check this we will check null values



No null values present in the dataset.

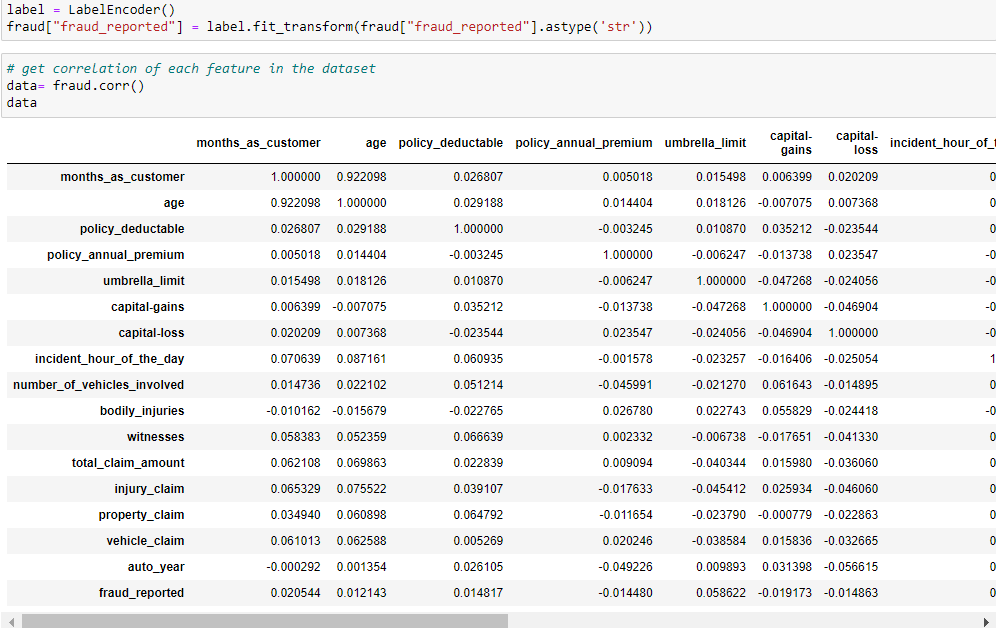
Let’s remove columns having many unique values



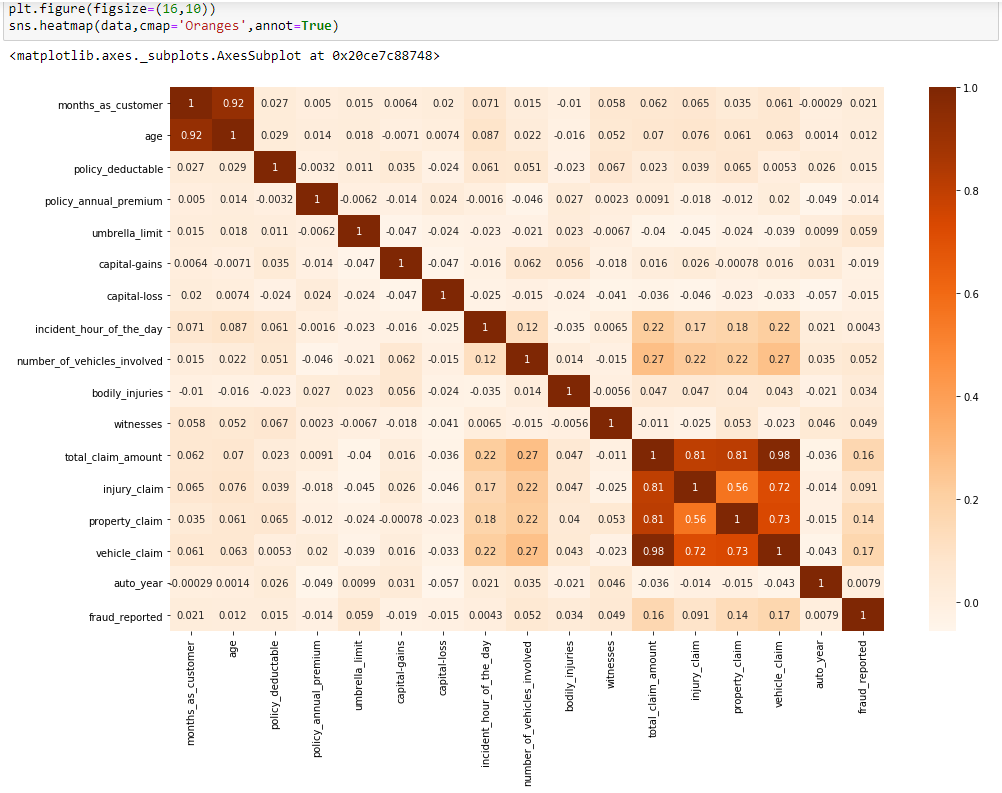
Now let’s check shape of dataset



To check relation of variable with numerical column let’s convert “fraud\_reported”, also check correlation



Getting correlation of each feature in the dataset that is visualize correlation matrix using heatmap.



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

# Observation: -

* dark shades are highly correlated with each other.
* the total\_claim\_amount, injury\_claim, property\_claim and vehicle\_claim columns is highly positive correlated with each other and also positively correlated with fraud\_reported feature.
* month\_as\_customer and age feature highly correlated with each other.

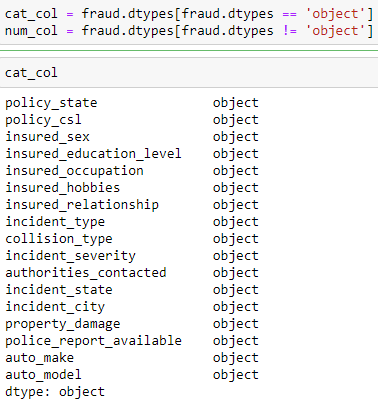
**Pre-processing Pipeline:**

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.

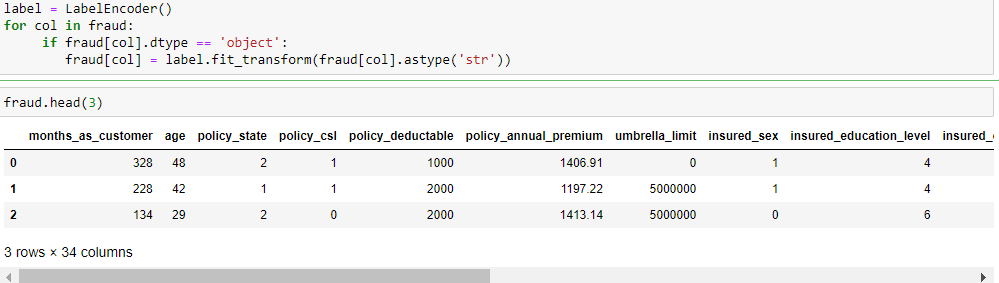
Further we will Separate categorical and numerical columns



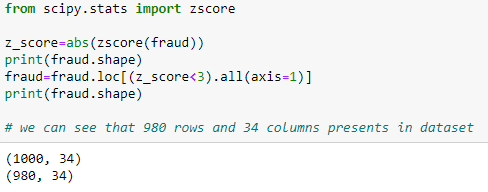
We can see that there are object-type variables too. These variables contain string data that cannot be passed into the machine learning model as it won’t be able to recognize string data type. It only recognizes numerical data.

Therefore, we need to convert the string data into numerical data. This can be done by manually encoding or by using an encoder such Label Encoder, one-hot encoder etc. For example: the target variable fraud\_reported consists of only two unique values, Y & N. after encoding this will get converted to 0 and 1. Similarly, if there are three unique values then it will be converted to 0,1, and 2.

Now we will convert all our categorical variables into numeric by encoding the categories, we will be using LabelEncoder for that.



Now let's check and remove outliers in the dataset using z\_score method



**Building Machine Learning Models:**

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

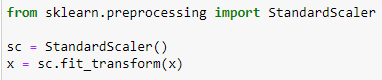
Let’s separate the dataset as input and output variable.



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

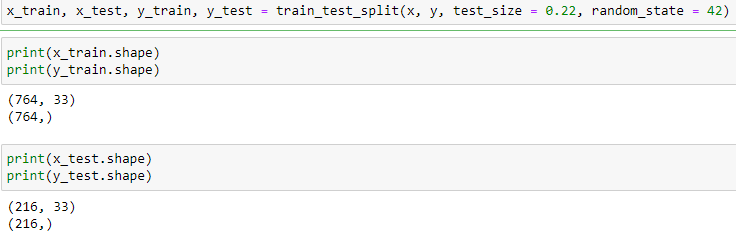
We have to now split the independent and target variables into training and testing datasets as shown below.

Applying Standard scaling to get optimized result



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.

Now splitting data for Train and Test



# Our training and testing data is ready now to perform machine learning algorithm¶

## The Auto insurance claim fraud prediction is a classification problem, so we can use Multiple classification algorithm with hyperparameter tune.

* First, we use Logistic regression model because the target variable hold binary classification (0 and 1) to check accuracy score level.
* we also used different classification model to check and compare whether we get high accuracy score or not, this exercise help us to select best model.
* We will use the following algorithms
* Logistic Regression
* DecisionTrees
* Random Forests
* SVM
* Naviebyes

**Concluding Remarks:**

# I used multiple algorithms to get highest accuracy score corresponding to random state hyperameter tune and gridsearchcv tune.

After we have seen till now SVC is best model as compared to other model which are apply in this because we got max accuracy score is 83% at random state 59 and also get high cross val accuracy score 77%

So we save model SVC for production.

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.