* **Problem Statement**:

Insurance companies are those organizations where a person can insure his health or anything that precious to him, In this case we will be dealing with **Auto Insurance industry**. Now by taking yearly or monthly premiums, these insurance companies give that customer the required protection against any kind of accident or any incident that affects the person's life. But as an insurance company deals with premiums (money), there are times that the claims can be fraudulent. Insurance is meant to cover against risks, not a path to enrich the insured. In some cases, a policy holder claims excessive money against his policy, when the severity of the incident is minor .

Some times in Auto mobile industry, policy holders fake their accidents or throwing away his vehicle or sell it out secretly or abundant his vehicle just like that telling that the vehicle has been stolen and claim exaggerate money to buy new vehicle or for any other purpose. But it is a fraudulent activity.

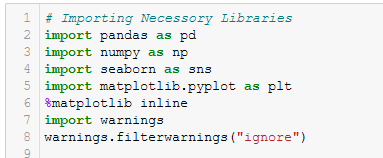
It is very much difficult for any insurance company to identify which claim is fraudulent and which is not as there are lot of **data** involved. There where the power of the machine learning enters. **Machine learning** is known for transforming industries and improving products. If there is a lot of data, it is impossible for human brains to analyse and without analysis it is impossible to predict in a right way, there **ML technology** performs to gain insights, automate the processes and do the predictions.

**To understand which one is a genuine insurance claim and which one is fraudulent**, we need to analyse the various aspects by using machine learning . There are various features that are involved which can be useful to detect the **fraudulent activity**. In this project, the provided dataset has the details of the **insurance policy along with the customer details**. It also has the details of the accident on the basis of which the claims have been made.

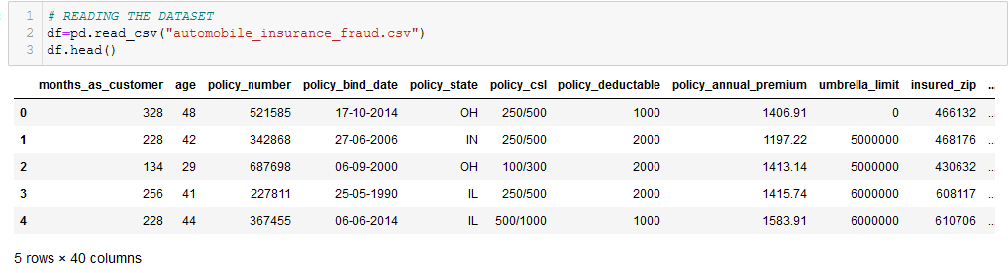
We will be working on this **Auto insurance data** and we will analyse each of the variable and create model that will predict almost perfectly that if an automobile insurance is fraudulent or not.

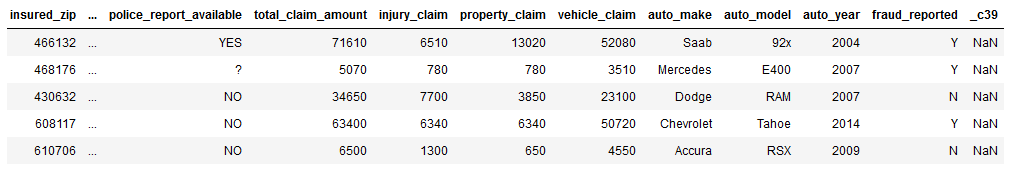
But before starting, we need to import various libraries.

* **Loading and reading the dataset**:



From loading the dataset , reading it to analyse the data, we need those above mentioned libraries. Now let’s start by analysing the data.





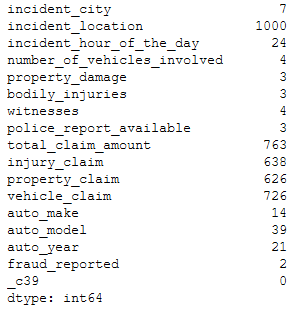
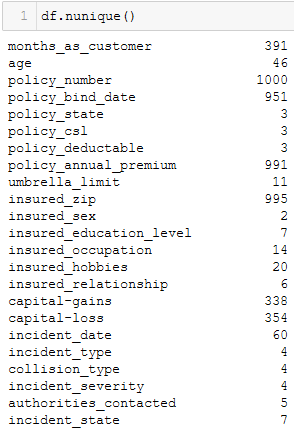
As we can see, this dataset is all about insurance claimers who have claimed insured amounts against their vehicles or other things which has got affected somehow.

There are **40 columns** in this dataset, out of which, depending upon **39 independent variables** we need to build a model that can predict the insurance claim is fake or not (fraud\_reported). As we need to classify between two categories of the target variable fraud\_reported, Hence it is a **binary type of classification**.

The dataset is good blend of numerical, categorical and nominal data. And all the features are in different scales. The 40 columns are as follows,

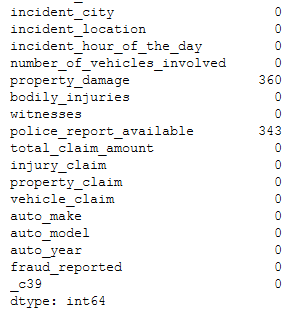
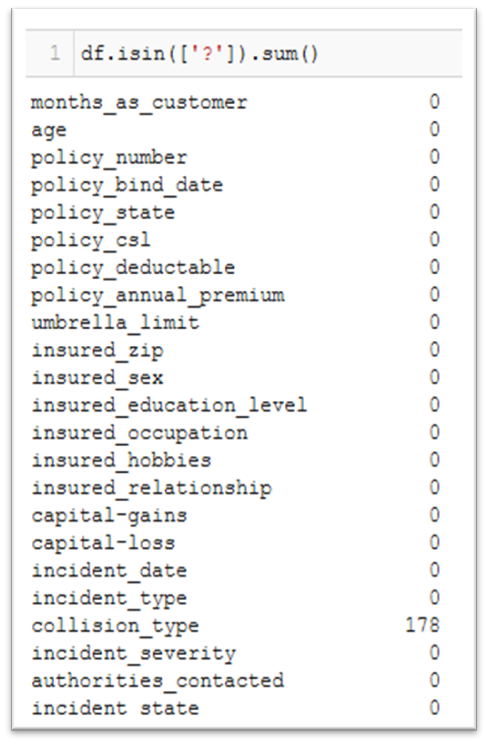
**months\_as\_customer, age, policy\_number , policy\_bind\_date, policy\_state , policy\_csl, policy\_deductable, policy\_annual\_premium, umbrella\_limit, insured\_zip , insured\_sex, insured\_education\_level , insured\_occupation , insured\_hobbies , insured\_relationship, capital-gains, capital-loss ,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**.

These columns are containing Nominal, categorical and continuous data. Each of the column is having different counts of categories . Let’s check that,

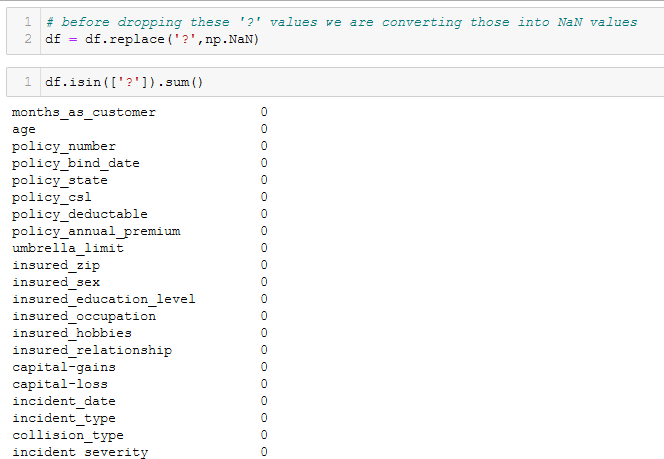
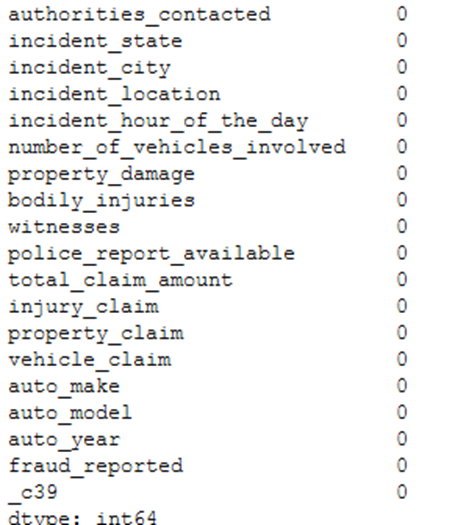


Out of all 40 columns which **fraud\_reported** is the target variable which is containing **2 categorical data**. Different columns are containing different number of categories or numerical values. Column **\_c39** is not containing any data and **policy\_number** column is also containing only unique serial numbers that is not required, Hence we will drop it before model creation.

In this dataset there is no missing value present except from \_c39. But there are values denoted as **‘?’** which are **might be NaN values**. The counts of those columns with ‘?’ categories are shown below.



Columns like **collision\_type**, **property\_damage**, **police\_report**\_available has got **‘?’** type of value. That we need to get rid of. We can do that by converting those values first into NaN values then we can use mode to remove those NaN values. By checking the below figure, we can tell that the **‘?’ values are replaced with NaN values**. But we need to get rid of those null values as well during data preprocessing.

. 

Now after understanding the datatype and presence of missing value we need to check the **statistics** of the data set as well. The statistical part will help us to understand the distribution of the continuous data. Let’s check it out.

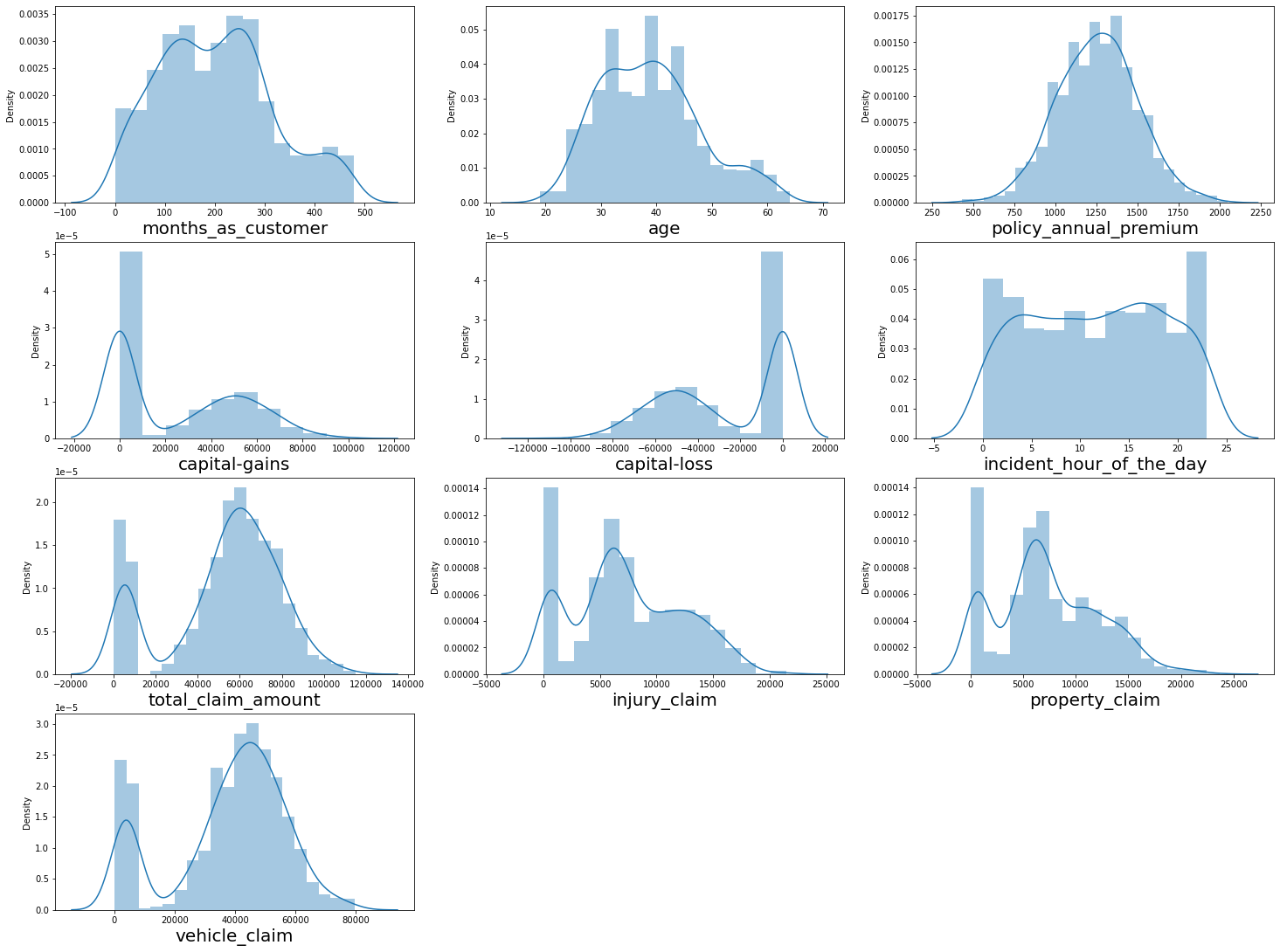


* There is no missing value present in numerical data.
* Minimum age is 19 and maximum age is 64.
* Policy number is the unique id that we will drop as it is not required.
* **policy\_deductable** is a **discrete data**.
* Minimum annual premium is 433.33 and maximum annual premium is 2047.59
* Minimum Umbrella limit is showing negative numbers, we need to analyze that once again.
* **number\_of\_vehicles\_involved**, **bodily\_injuries**, **witnesses** features are containing **discrete data**.
* \_c39 is a column with all missing values that we will get rid of.

**Exploratory Data Analysis:**

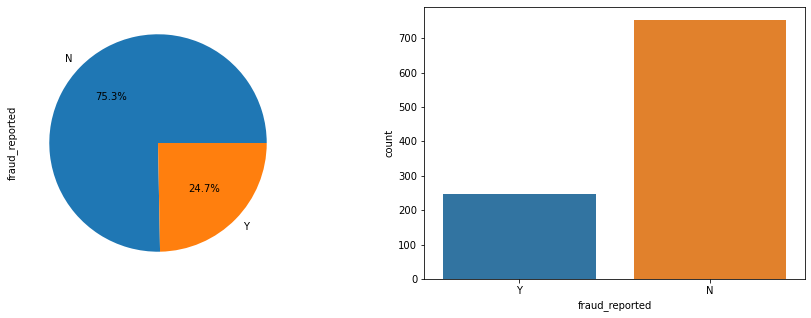
We will do **Univariate analysis**, **bivariate analysis** and check the **multicollinearity** of the features. EDA will help us to understand the data more precisely and by visualization it will be helpful to understand which features are important than others. With those important features, the creation of the model will be more accurate and without bias.

**Univariate Analysis:**

We will do Univariate Analysis to understand the **distribution** of **continuous** features and **count** of the **categorical** features. Let’s visualize, the distribution of the continuous data first.

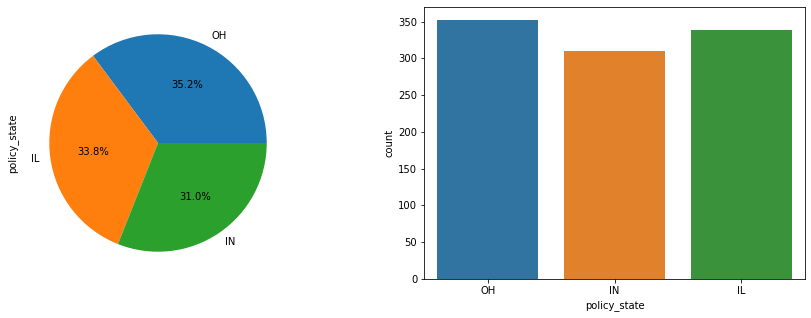
* **Capital gain, Capital loss, total claim amount, injury claim, property claim, vehicle claim** features are having some **skewness**.

Now let's check the **Target variable** also:



* The target (**fraud\_reported**) is a categorical data, having **2 categories**. One is fraud report is **true (Y)** and other is **fraud report is false (N).**
* False fraud report has more count of data 753 (75.3%) and True fraud report has less counts 247 (24.7%).

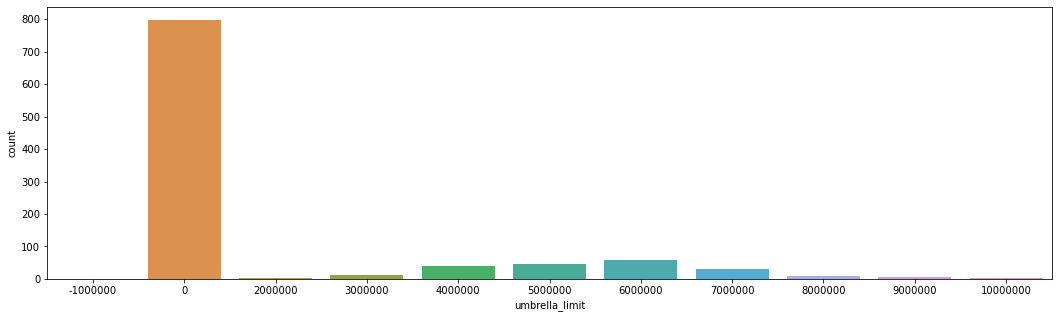
Now let’s analyse the categorical data as well,



* **policy\_state** feature has 3 categories, OH counts highest 352 (35.2%), IL counts 338 (33.8%) and IN counts 310 (31.0%).



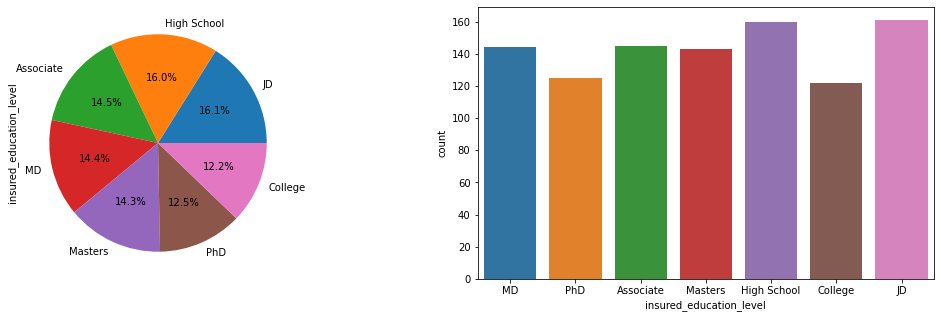
* **Combined single limit (CSL)** is a single number for predetermined coverage of total Bodily Injury,Property Damage per accident. So we can see , there are 3 categories of csl number 100/300,250/500 and 500/1000 out of which 250/500 has the highest count of 351 (35.1%).
* **policy\_deductable**(the amount the customer pay before the insurance company begins paying up) feature has 3 categories. 500,1000 and 2000. Out of which category 1000 has the highest count 351 (35.1%).



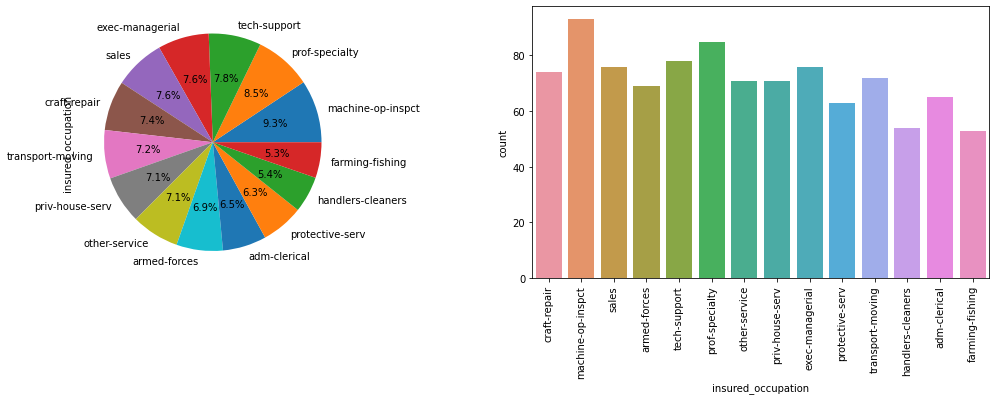
* **Umberella limit** has 11 categories out of which category 0 is most common .



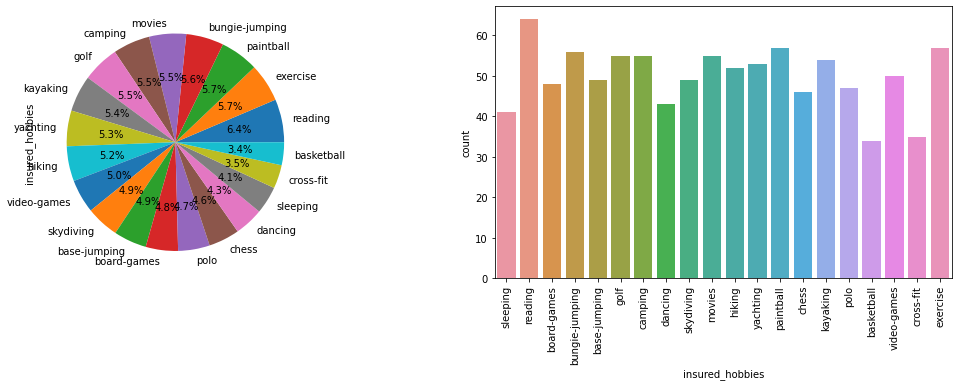
* Count of female individuals is more (537) than male individuals as **insured sex**.



* JD counts highest (161) as **insured education**. And least insured education level is college(122).

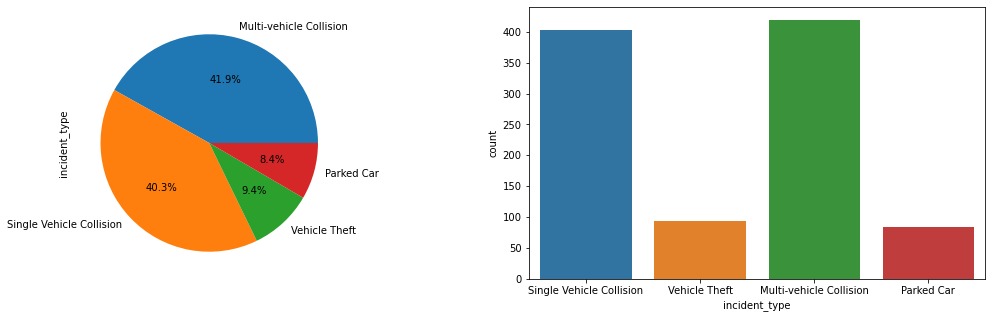
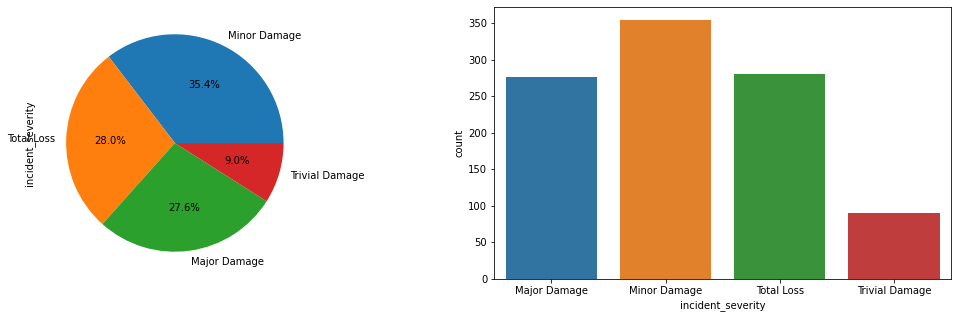


* machine-op-inspct has got the highest count ( 93 ) of insurance regarding **occupation**. Least insured occupation is farming-fishing (53).



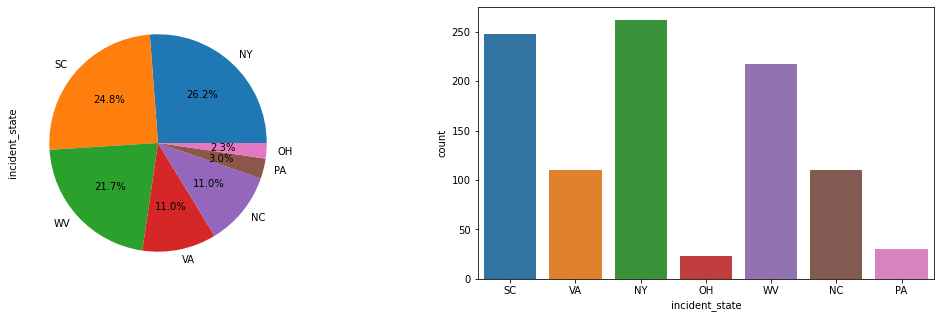
* reading has got the highest count (64) of insurance as a **hobby**. And least insured hobby is basketball (34).



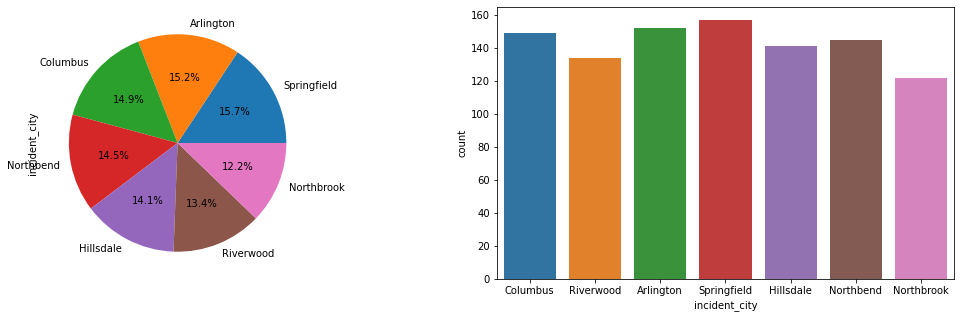
* own-child has got highest counts (183) of insurance as **insured relationship** and least insured relationship is unmarried (141).
*  The most occurred incident is **Multi-vehicle Collision** ( 419 ) and least occurred incident is Parked Car ( 84 ).
*  Minor Damage has got the highest counts ( 354 )as **incident\_severity** and Trivial Damage has got the least counts (90) as incident severity.



* Police has got the highest counts (292) as **Authorities\_contacted** and None has got the least counts (91).



* NY has got the highest counts 262 as **incident state** and OH has got the least counts (23).



* Springfield has got the highest counts (157) as **incident\_city** and Northbrook has got the least counts (122) as incident city.



* category 1 vehicle has got the highest counts (581) as **number of vehicle involved** and category 2 vehicles has got the least count.



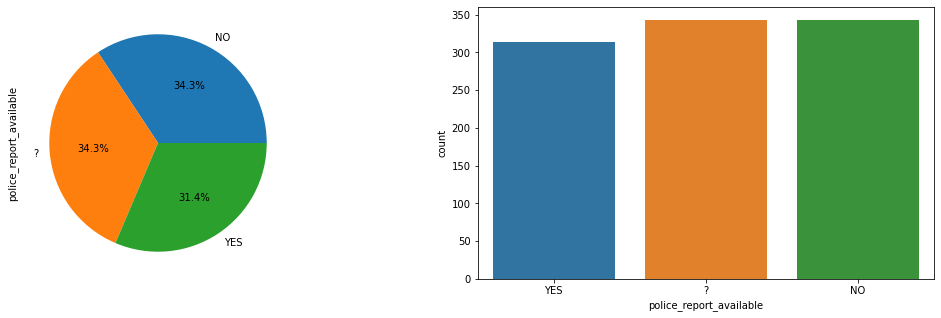
* '?' has got the highest counts (360) as **property damage** but '?' means nan values that we need to deal with. and Category Yes got the least counts (302).



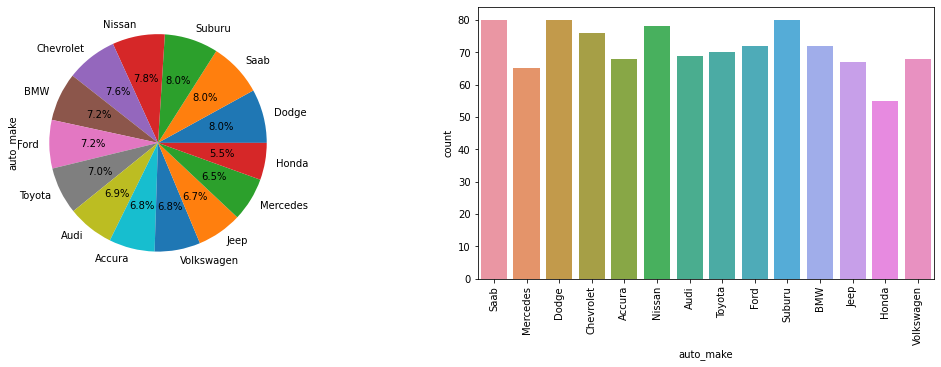
* category 0 has got the highest counts (340) as **bodily injuries** and category 1 has got the least counts (328).



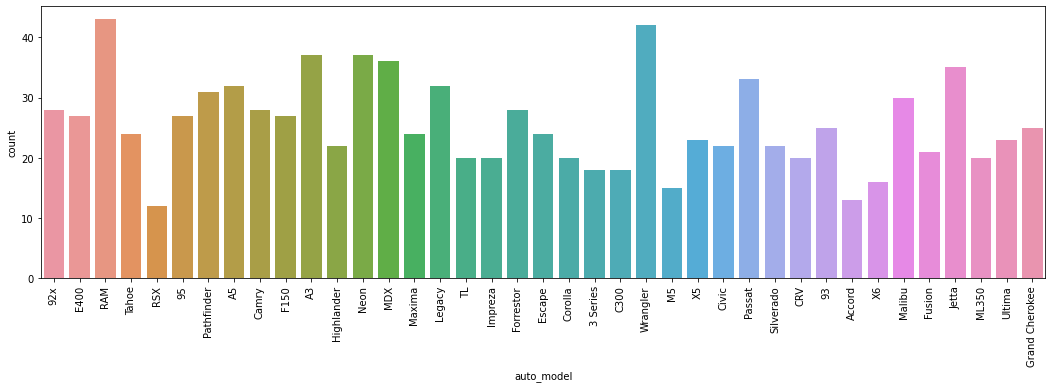
* category 1 has got the highest counts (258) as **witnesses** and category 3 has got the least counts ( 243).



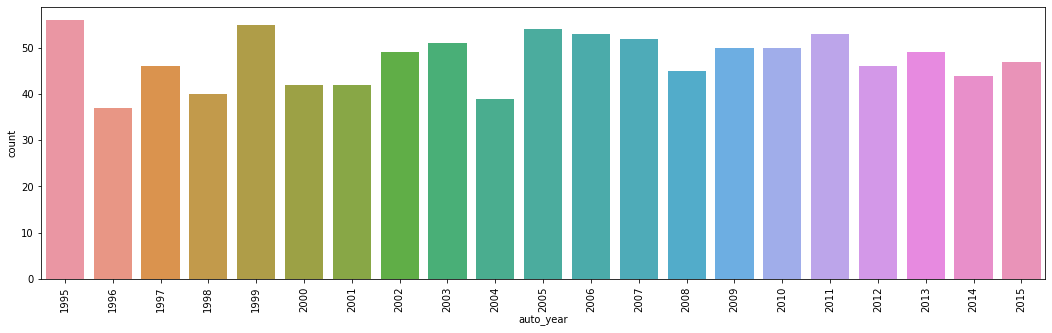
* category No and '?' has got the highest counts (343) respectively as **police\_report\_available** but '?' means nan values that we need to deal with. And Category Yes got the least counts (314).



* Dodge has got the highest counts (80) as **auto\_make** and Honda has got the least counts (55) as auto\_make.

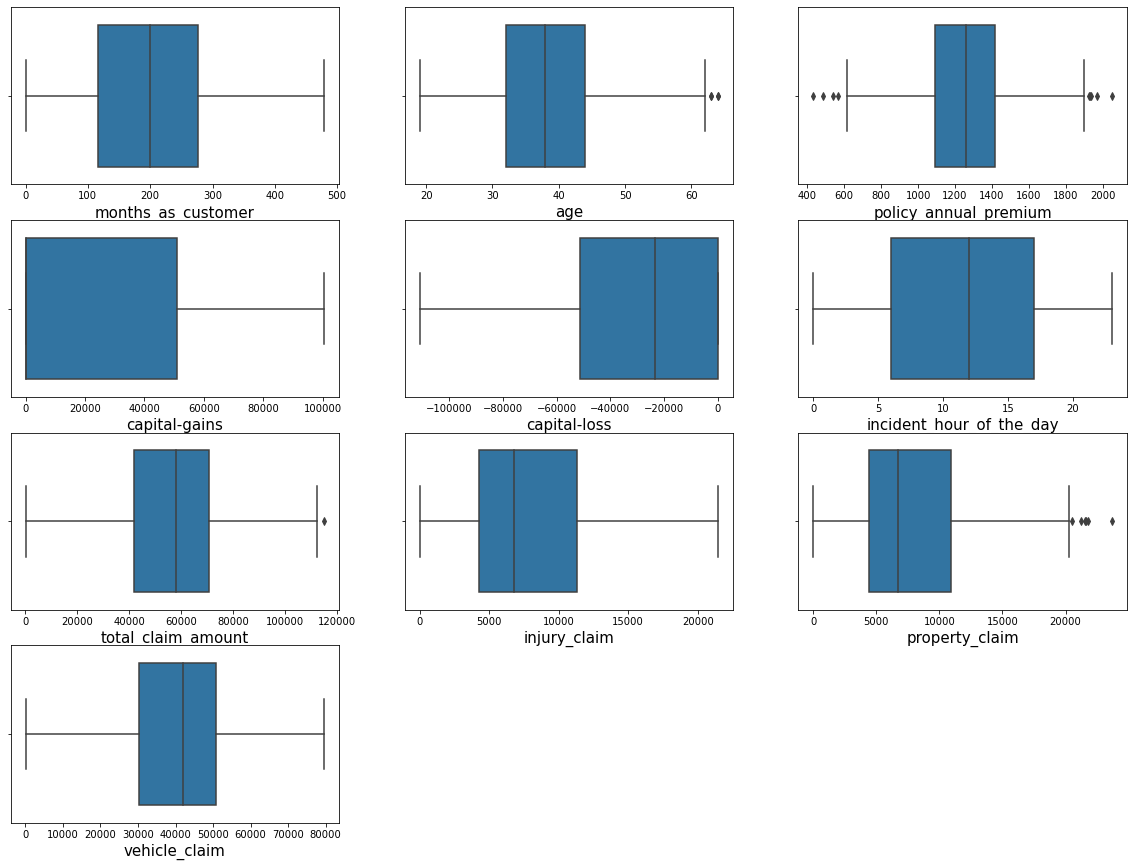


* RAM has got the highest counts (43) as **auto\_model** and RSX has got the least counts (12) as auto\_model.



* year 1995 has got the highest counts (56) as **auto\_year** and 1996 has got the least counts (37) as auto\_year.

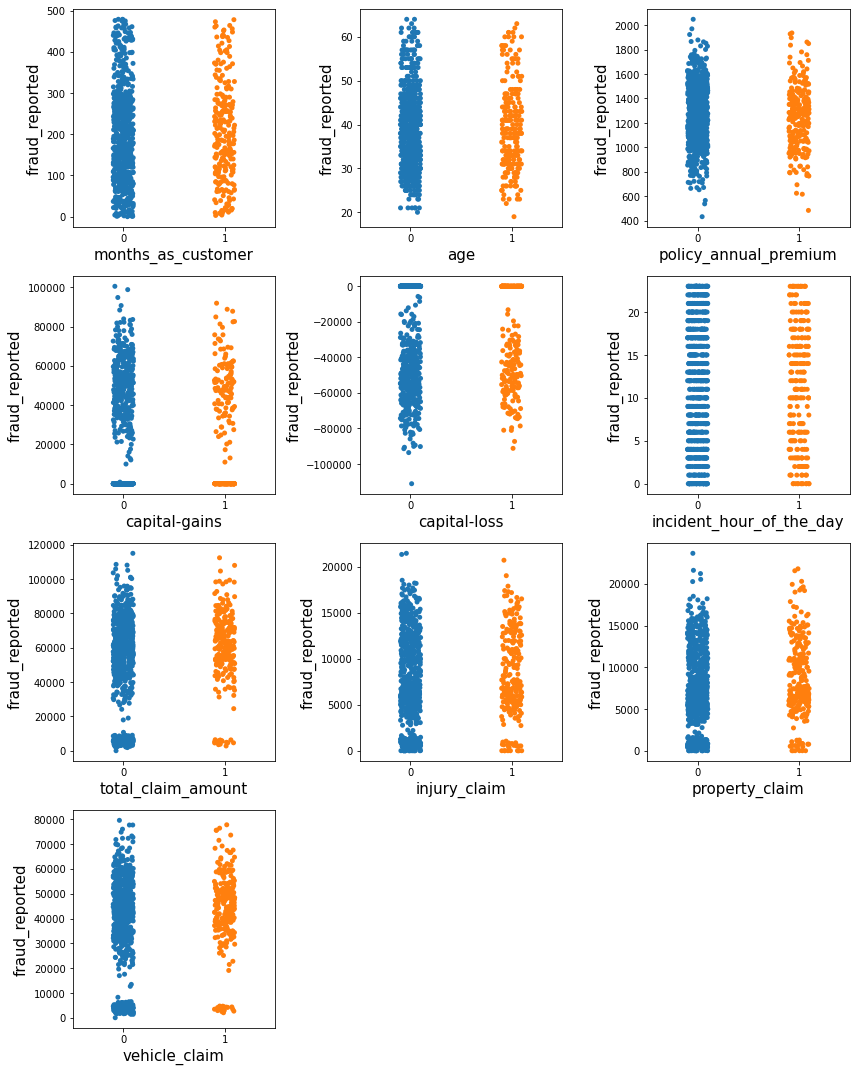
**Outlier Detection**: An outlier is the data that is distinctively different from rest of the data. We need to take care those outliers because we need to get rid of those for creating a good model. Now let's analyse the outliers in continuous data:



* **Age, policy\_annual\_premium, total\_claim\_amount** and **property claim** features have little bit **outliers** that we need to take care.

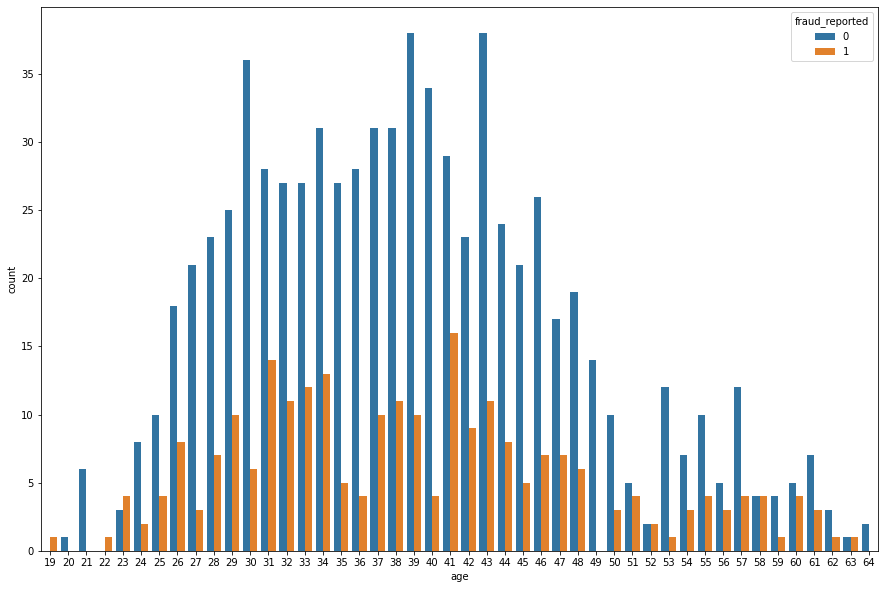
**Bivariate Analysis:**

we will do Bivariate Analysis to understand **relations between** feature vs target and feature vs feature. And for that we will check the relations between continuous features with target variable and after that we will check the relations between the categorical features and the target variable. Let’s do it.



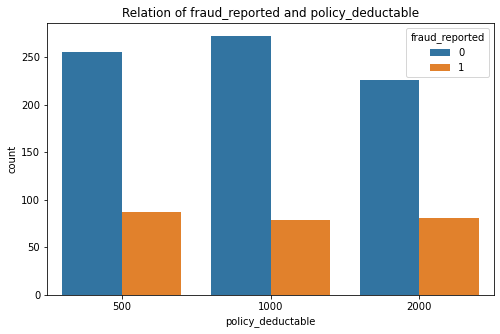
* Up to **300 months** as customer, some people mostly **do not fake** their accidents but some fake their accidents.
* Up to **age 50**, some people **mostly** **do not fake** their accidents but some fake their accidents.
* Customers who **pay** their **premium annually** are **mostly not fraud**.
* Customers who are not fraud, **capital gain** is more for them.
* Customers who are not fraud, **capital loss** is more for them.
* Customers who have faced accidents from **5am to 10am** are mostly not fraud.
* **Customers** who are **not into fraud** **claim property**, their claiming **amounts** are **reasonable and less**. Customers who are not fraud their injury claim is more and property claim is also more.

Now Let’s check the relation between Age and target variable once again.

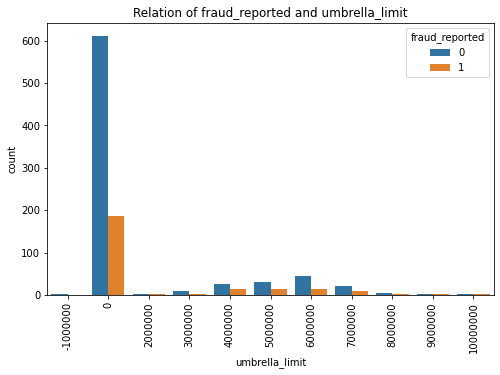


* Customer with **higher age**, the **fraud report** on the **claims reduce**, but from the age of **23 to 48** the number of **fraud report** is **high**.

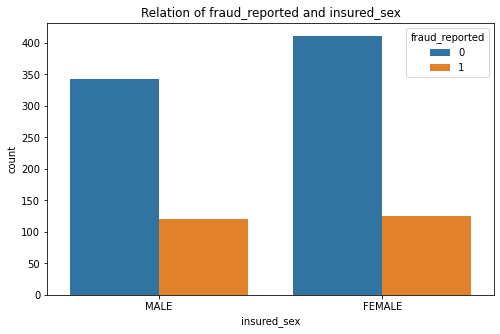
Now let’s check the relation between categorical features and target variable.



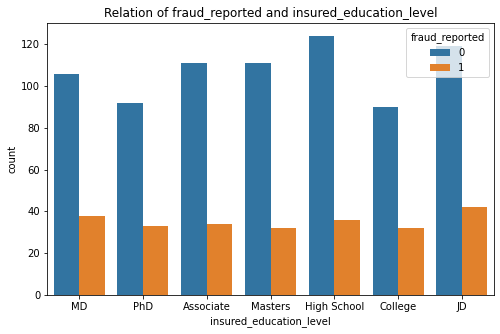
* The fraud report on **policy\_deductable** is less in all **3 types**. But fraud report is present in all the categories.



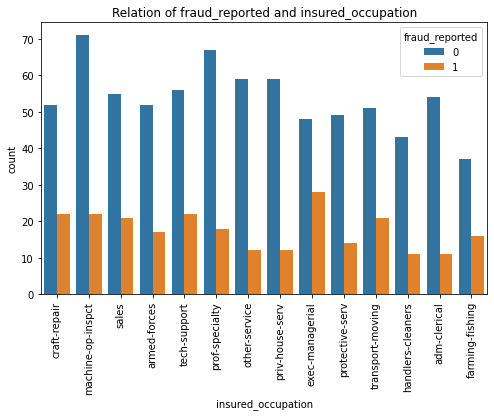
* In the category **0 Umbrella limit** (which is most common ) there is less Fraud report.



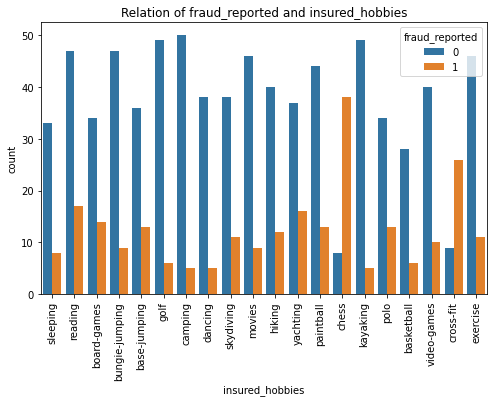
* **Fraud reports** for **both male** and **female** are there.



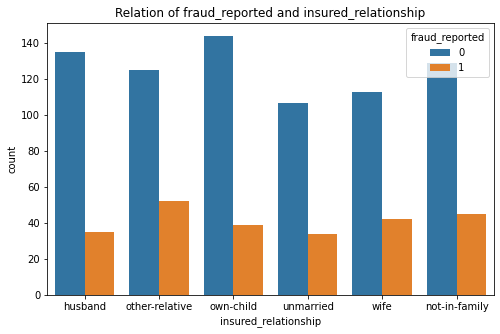
* Fraud report is **almost same** for **all insured education level**. The number is less.



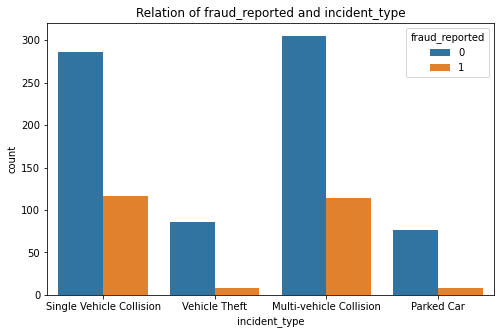
* **Fraud report** is **high** in **exec-managerial** post of all the insured **occupation**.

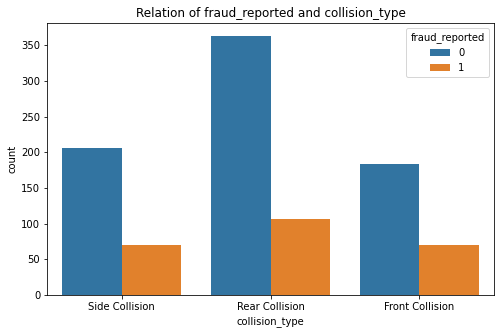


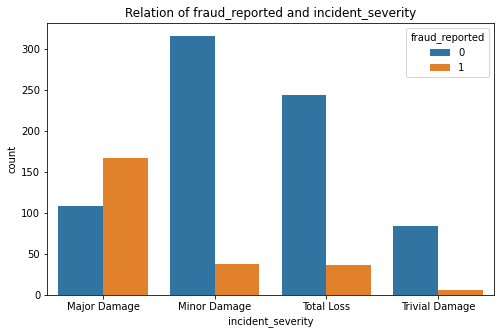
* **Fraud report** is highest in **Chess** of all the **insured hobbies.**



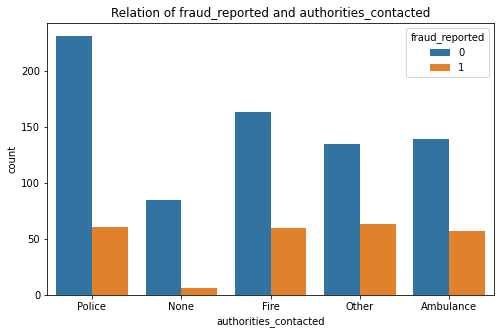
* **Fraud reports** are **more** in counts in case of **other relatives** of insured **relationship**.



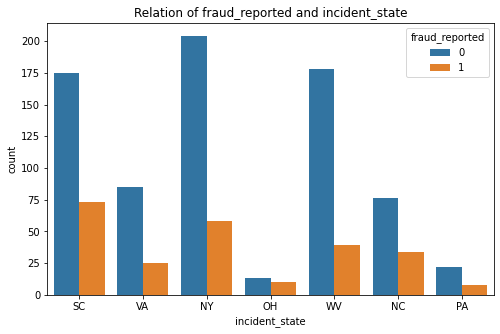
* **Fraud reports** are more in **single vehicle** **collision** and **multi vehicle collision** .
* In all cases of **collision type** fraud report is there. But fraud report is less in count in every category.



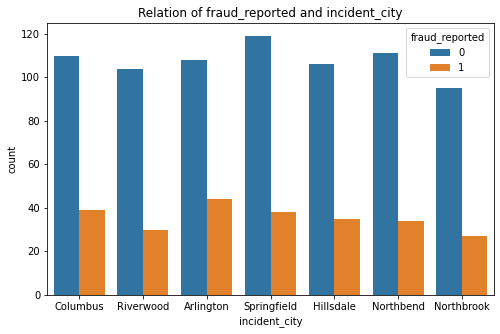
* In case of **major damage** as **incident severity** the **fraud report** is **higher**.



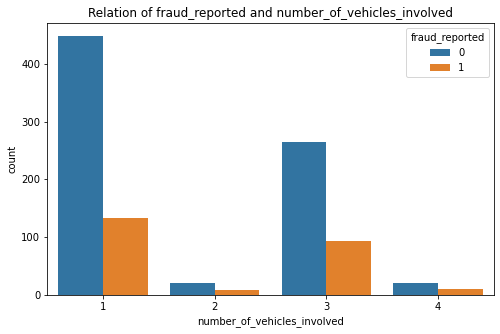
* In all cases of **authorities contacted**, fraud report is there. But fraud report is less in count in every category. And where **police** is involved the case is more truthful.



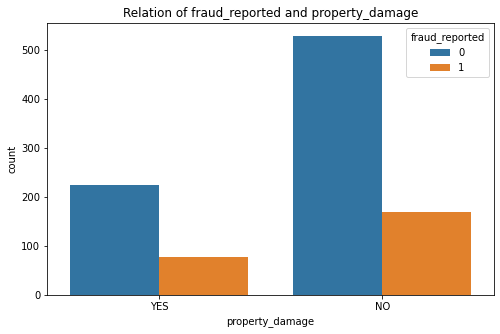
* All the **states** are having more or less fraud reports. But **Non Fraud** reports are most in **NY**.



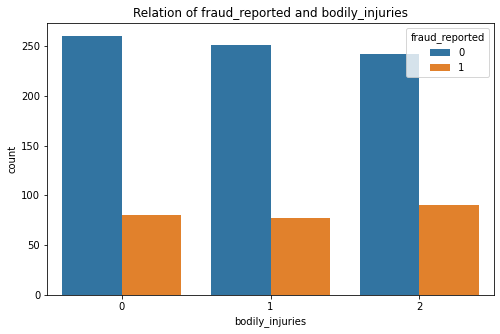
* **Arlington city** has got the **highest fraud reports** as incident city.



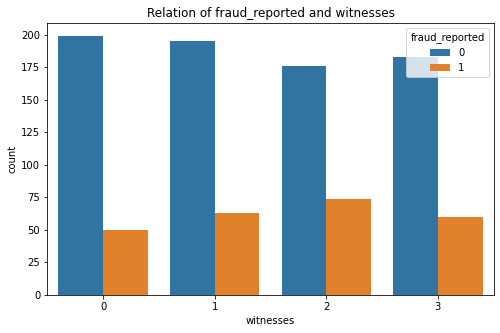
* Mostly **one** **vehicle involved** in accidents and **fraud report** is there in all the categories.



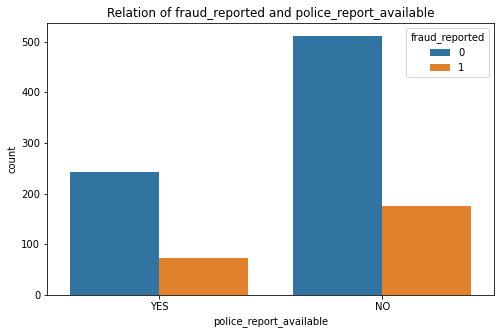
* **Property** is damaged or not fraud report is there.



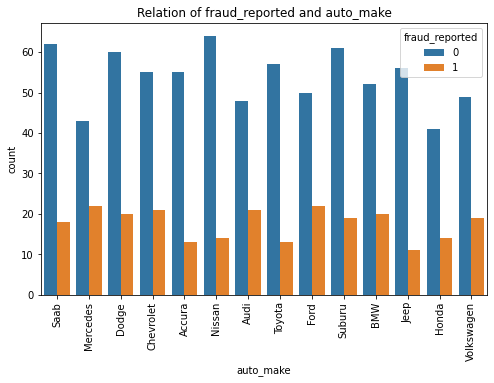
* In case of **bodily injuries** fraud report is there in every case.



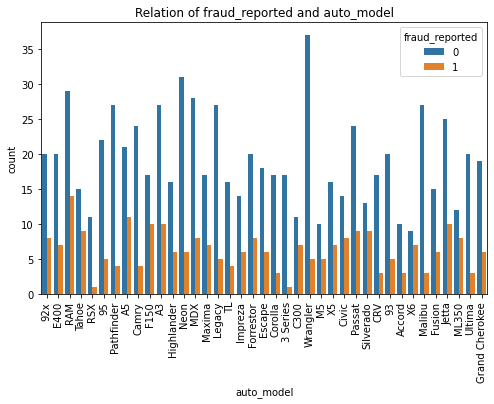
* The **2 witnesses** category having **more** **fraud reports** than other witnesses categories. But fraud report is there in all the categories.



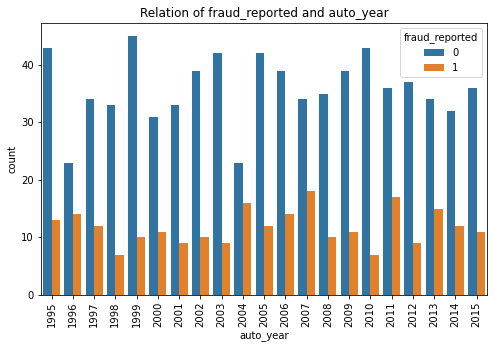
* **Fraud report** is more there where **police report** is **not available**.



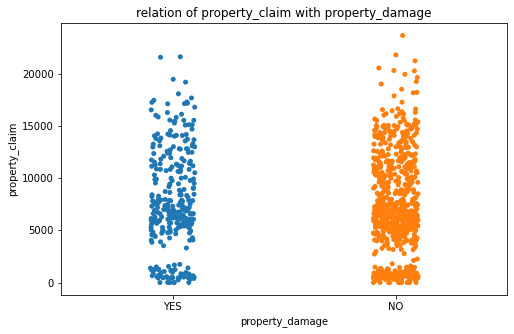
* **Nissan** has got the **highest non-fraud** reports. But in every case including Nissan, fraud report is there.



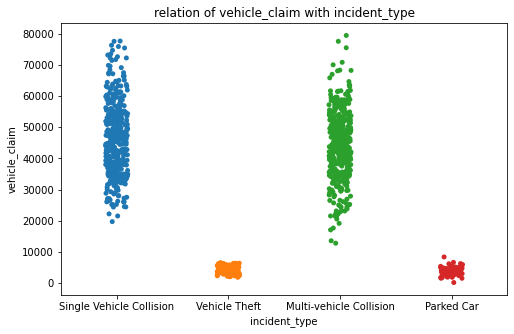
* **Wrangler** has got the **highest non-fraud** reports as **auto model**. But in every case including Wrangler also, fraud report is there. **RAM** has got the **highest fraud reports**.



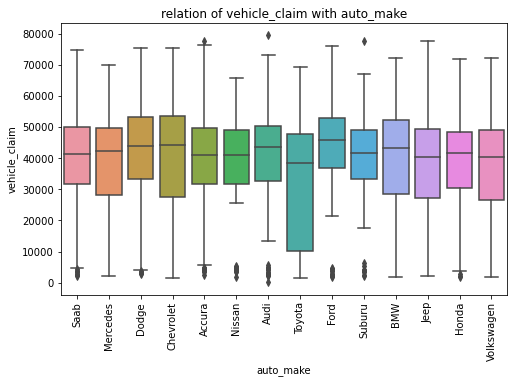
* In all the **years** fraud report is there, But in **2007** it is **highest**.
* Now let’s check the relation between property claim and property damage ( feature Vs feature relation )



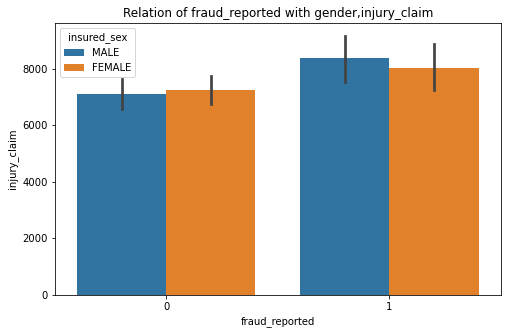
* Customers, **without** **property damage** also **claim money**.



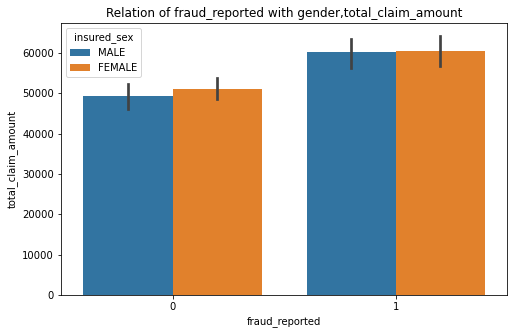
* In case of **Single Vehicle Collision** and **Multi-vehicle Collision** the vehicle claim amount is more.



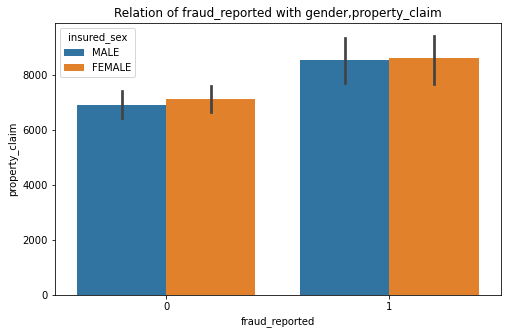
* Customers with **Audi** and **Accura**, do the highest **vehicle claim**.



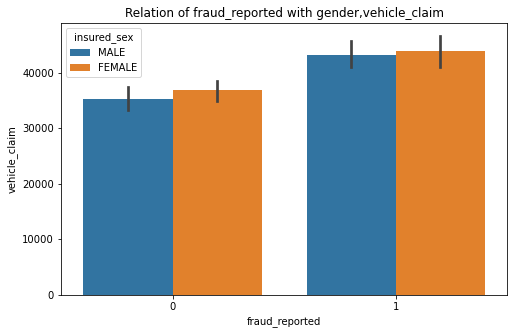
* **Female** individuals has got highest non fraud reports as claiming the injury. And **male** has got **highest** fraud reports as **claiming the injury**.



* **Female individuals** have got more non fraud reports as total claim amount.

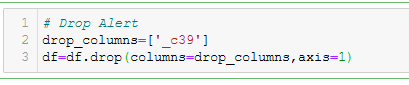


* **Female individuals** do more **property claim** than male individuals. That with fraud reports or without fraud reports.

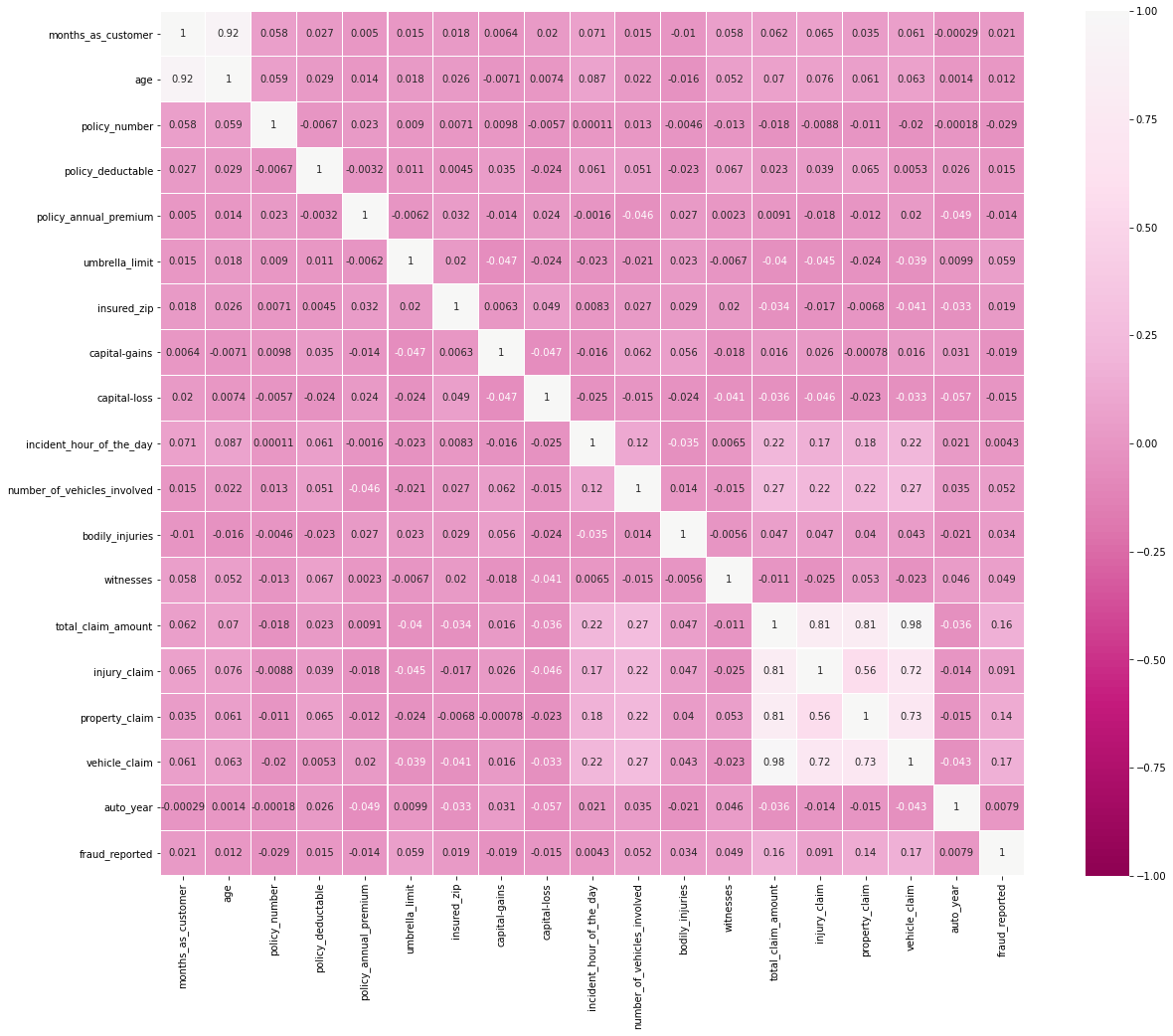


* **Female individuals** do more **vehicle claim** than male individuals. That with fraud reports or without fraud reports.

Now , we are going to drop **'\_c39'** column , as it’s having only missing values.



Now, it’s time to check the heat map to understand the **correlation** and **multicollinearity** in the dataset.



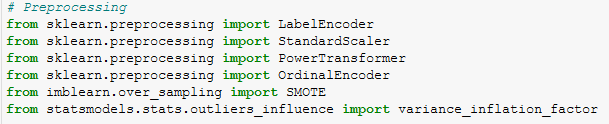
* **Age** and **months as customer** features are having **correlation** (0.92).
* **Vehicle claim**, **property claim**, **injury claim** are having **correlation** with **total claim amount** that we need to take care.
* **Total claim amount** has good **correlation** with **fraud reported**.

**EDA Concluding Remarks**:

1. Most interesting thing to notice that **insurance companies cover hobbies** also and people take **sleeping** as a **hobby**.
2. **Female** individuals do **vehicle claim** more than male individuals. And another thing is they do more fraudulent claims than male individuals.
3. **Male** individuals do more **fraud claims** regarding **claiming injury**.
4. In **1996** and **2004** both the **fraudulent and non fraudulent insurance claims** were **least**.
5. **Springfield** has got **highest** **non fraud reports** where **Arlington city** has got the **highest fraud report**.
6. Where **Police is not involved** and **no report from their sides**, there **more claims** that are **fraud**.
7. **People insured** their **own child** more than **other relatives**. In case of **other relatives** the **fraud claim** is **more**.
8. In the **state OH** **both the claims** are **least**.
9. In most cases **one vehicle** is involved.
10. With **growing age** people claim **less insurance**, may it is fraud report or not.
11. In the **age 21** there is **no fraud claim**.
12. In case of **Toyota** the vehicle claim is ranging from very less to very high. The **variance** is very much there.
13. Injury claiming amount, property claiming , vehicle claiming are already there in total claiming amount. Hence, the **correlation**. After checking VIF we will deal with it.
14. We did not get any idea about the column **\_c39** as it was containing only missing values.
15. All the different columns are having **different unit of values**. Before creating the model, we need to **standerdize** them and making them more standard normal distribution like.

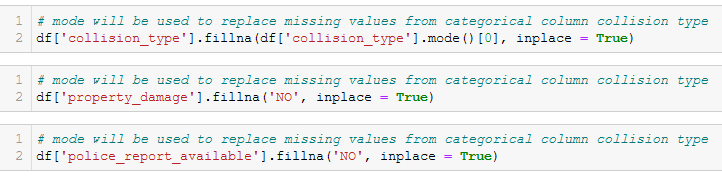
**Data Preprocessing:**

In this dataset we did not have missing values as such but after **replacing** the **‘?’** values that were present as a category in **collision\_type, property\_damage, police\_report\_available** with **NaN values**, now the dataset is having null values that we need to get rid of. Before doing any **preprocessing** we need to **import necessary utility functions** and **transformer classes** from **scikit-learn**.



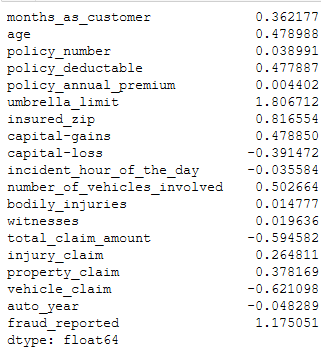
Now, we can perform **preprocessing** of the data.

1. **Filling missing values**: we will use **mode** to replace missing values from categorical column **collision type, property\_damage, police\_report\_available**.

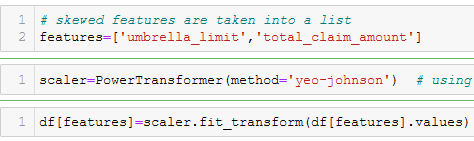


* As for both property\_damage, police\_report\_available features having the highest frequency in ‘No’ category, we are replace the NaN values with that only.

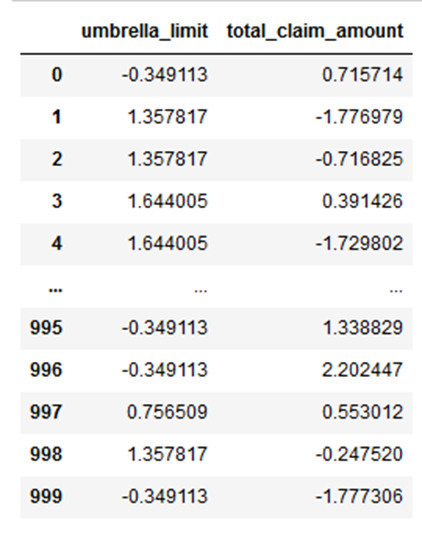
1. **Skewness removal**: There are continuous data which are having **skewness**, for building a better model we need to remove those skewness from the data. The amount of skewness is shown in the picture below.



* There is **skewness** present in **'Umbrella\_limit'** and **'Total claim amount'** features. We will never remove skewness from target variable. To remove skewness we will use **PowerTransformer**.



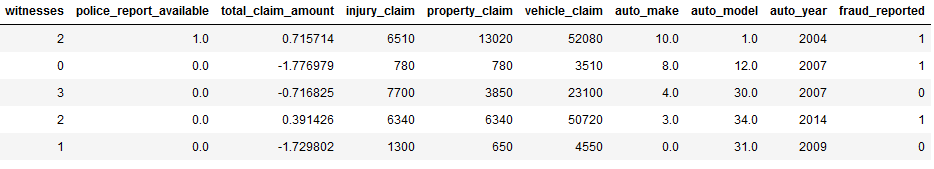
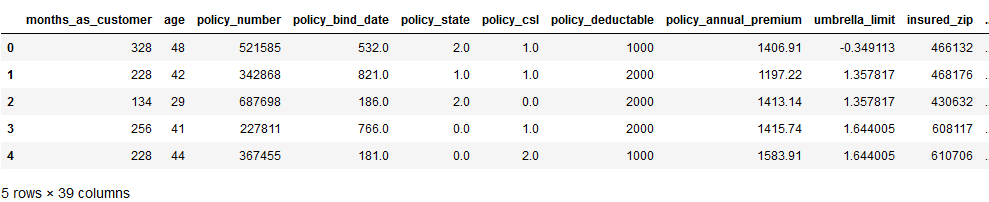
As **box-cox** works with **positive values** ( > 0 ) And **yeo – johnson** works with **both positive and negative values**, we are using **yeo-johnson** method As **total claim amount** is having **negative skewness**.



* Now both the features are looking better than before.

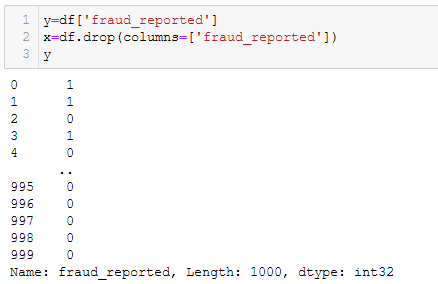
1. **Encoding** : **Machine Learning** models **never understand texts**. **Encoder** helps to **convert** the texts **into numbers** to understand the data properly. As our **target variable** is **categorical data**, we will encode with **LabelEncoder** and for rest of the **Nominal columns** we are using **Ordinal Encoder**.

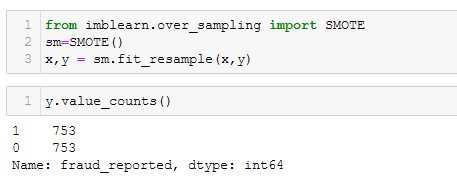
After **Endoding** the target will encoded into **0 (Not fraud )** and **1( fraud )**.



* As we can see our dataset is almost ready for model creation as all the **nominal data** have **encoded**. But before that we will **drop** some **unnecessory** columns like, **policy\_number** , containing serial numbers as **unique id** and **policy\_bind\_date**, as we already have **months as customer** feature we do not need **date column**.

Now after that we will **divide** the dataset into **target variable** and **features** . As target variable needs to be balanced before model creation.



1. **OverSampling** : As we need to increase the **minority class (category fraud or 1)** of the target variable to get a better prediction, we will **over sample** the class by using **SMOTE**. 

* The **class 1** of the target variable had very less data, that is why **there were imbalance** in the variable. Now, it’s **balanced**.

1. **Standardization:** As most of the features are in **different scales** , it can be possible that **variance of a feature** has magnitude that is **larger** than **other features**, and for that our model will get **biased** **towards the feature** and will **perform badly**. For that reason to make all the feature **more normal distribution** like we will **do standardization**.

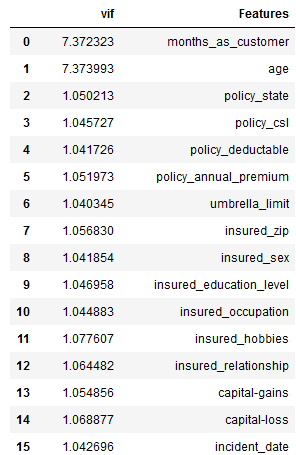
The formula of data scaling is,

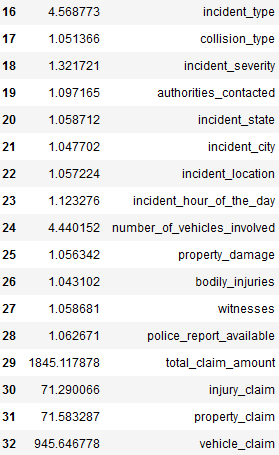
**Z = (x-mean)/ standard deviation**

Where,

* **Z** = The number of standard deviation away from mean,
* **x** = each value of a feature,
* **mean** = mean of that feature ,
* **standard deviation** = variance of the feature



* After standardization, we will check the **multicollinearity** of the features using **VIF(Variance inflation factor)** and all the **VIF values should be less than 5**. If not then we should drop those **correlated features**. 



* There are features like**, months as customer , injury\_claim, property\_claim, vehicle\_claim, incident\_location** which are having **multicollinearity** that we will drop.

After dropping the correlated features now we can create the models. And for that first we will use Hold Out Method to split the data into train and test.

1. **Hold Out Method**:

Model will be built on Hold Out method(**75/25 training and test split**) to gauge how well the classifier is working. First we will train our models with 75% of the data then we will do prediction by using other 25%.

**Model Creation:**

We use **classification models** to deal with **qualitative data** or categorical data. The algorithms used for solving a classification problem **first** **predict the probability** of each of the categories of the qualitative variables, as the basis for making the classification.. In this classification problem we are going to use **six** classifiers. we are using models like,

1. **Logistic Regression**, as the probabilities are continuous numbers, classification using probabilities also behave like regression methods. Logistic regression is one of the such supervised machine learning technique which is used for **binary classification**.
2. **Decision Tree Classifier**, DT is a **rule based** algorithm. Like if the **condition** is true then go this side or go that side. For making **tree** like structure to predict or decide the outcome, the algorithm is called Decision Tree.
3. **Random Forest Classifier**, as RF is a **ensemble** technique that fits several models or estimators in a training dataset . In RF we do **sampling on features**. So, **randomly** features are taken to each **model/estimators**. By doing that all types of features are there in each of the models/estimators . So, if there is **correlation** , Random Forest makes tweak in the working model to **decrease** the **correlation** in the trees.
4. **K-Nearest Neighbors Classifier**, KNN tries to predict the correct class of test data by **calculating the distance between the test data and all the training points**. It then selects the k points which are closest to the test data. Once the points are selected, the algorithm calculate the probability of the test point belonging to the classes of the k training points and the class with highest probability.
5. **Gradient Boosting Classifier,** gradient boosted trees use decision tress as estimators. Evaluate its gradient and approximates it with a simple tree(**stage wisely** , that minimizes the overall error). First it calculates the average of the target, for the first iteration it ( average of actual target) is the predicted target, Then it calculates the **pseudo-residual** by subtracting the first predicted target( average of actual target ) by actual target . It tries to **reduce the error function** as it creates a tree to predict the pseudo-residuals instead of a tree to predict for actual column values. Then When data is not following any pattern gradient boosting trees are very helpful as it tries to optimize the error function.
6. **Support Vector Classifier,** It tries to divide the data using **hyperplanes** and then makes the predictions. SVM is a **non-probabilistic linear classifier**. When other classifiers , when classifying, predict the probability of a data point to belong one group or the other, SVM can predicts to which group the datapoint belongs without using any probability calculation.

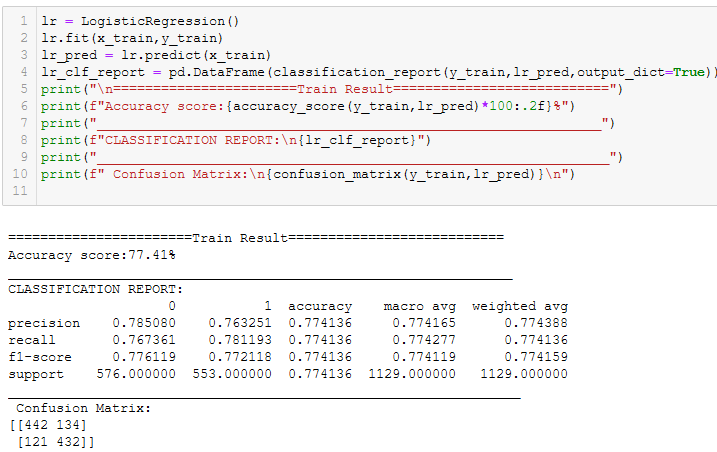
For evaluation we will check the **Recall, Precision , F1 score** and **AUC\_ROC score**.

**Recall**: It is also known as Sensitivity. It is a measurement of total true positive divided by the number of true positive and the number of false negative. **Recall = TP/TP+FN**

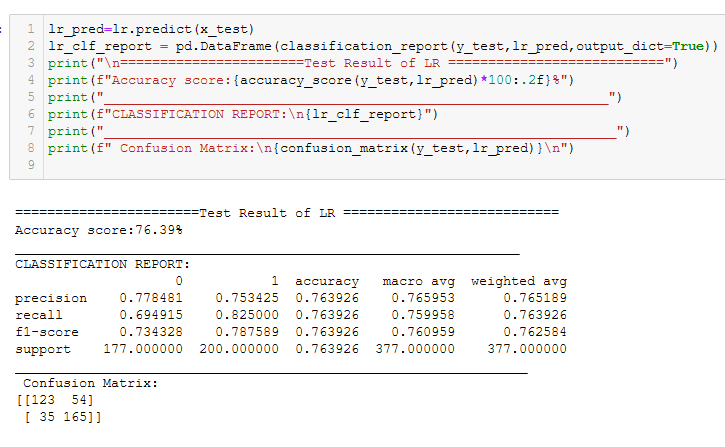
**Precision**: It measures, amongst all the positive results how many are the actually positive. **Precision = TP/TP+FP**

**F1 Score**: It is the combination of precision and recall. **F1 Score=2\*(Precision\*Recall)/(Precision + Recall)**

Now we will start creating the models and eventually do predictions as well.

1. **Logistic Regression:**

We have done training now we are going to predict and evaluate.



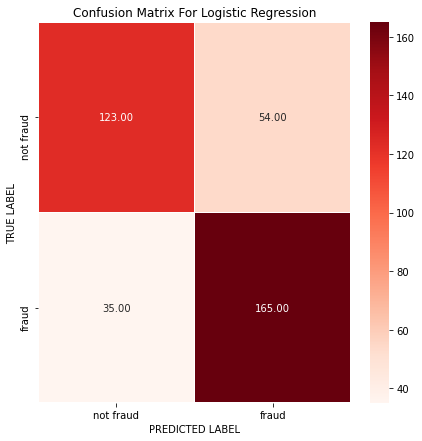
The confusion matrix is showing, the

**True positive**=123, **False positive**= 54,

**False Negative** = 35 , **True Negative** =165

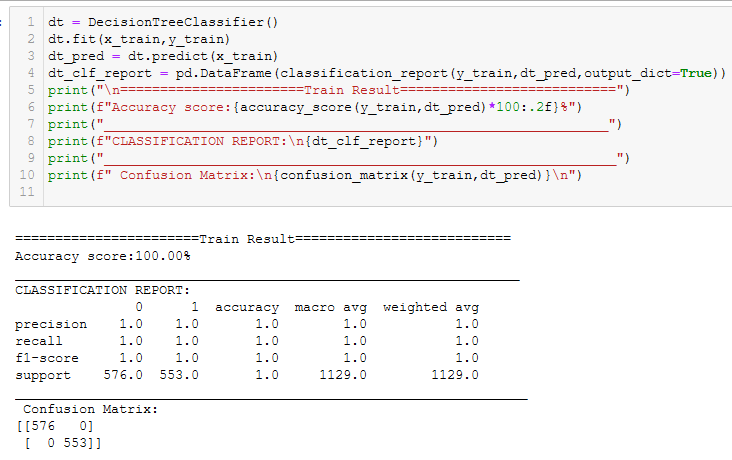
Hence, the **type 1 error** is 54. **Precision** is 0.77 and 0.74, **recall** is 0.69 and 0. 82 And the **f1 Score** is 0.73 and 0.78

**Accuracy** score is 79.39

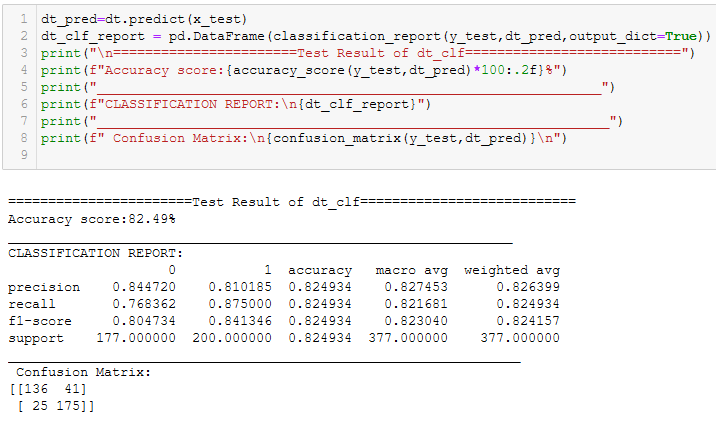


* Now let’s check the **AUC\_ROC score** as **ROC ( Receiver Operator Characteristic)** is a balance between true positive rate and false positive rate for a predictive model using different probability threshold. Hence, the **AUC\_ROC score of Logistic Regression is 0.75**.

1. **Decision Tree Classifier :**



We have trained the model now time for prediction.



The confusion matrix is showing, the

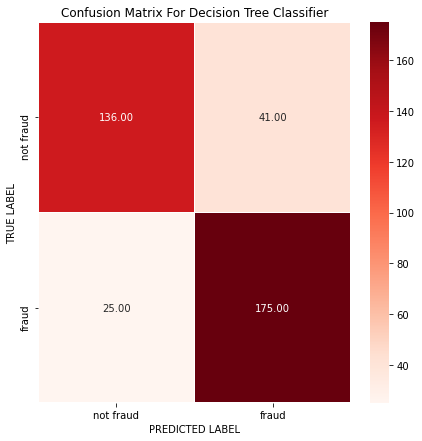
**True positive**=136, **False positive**= 41,

**False Negative** = 25, **True Negative** =175

Hence, the **type 1 error** is 41. **Precision** is 0.84 and 0.81, **recall** is 0.76 and 0. 87 And the **f1 Score** is 0.80 and 0.84

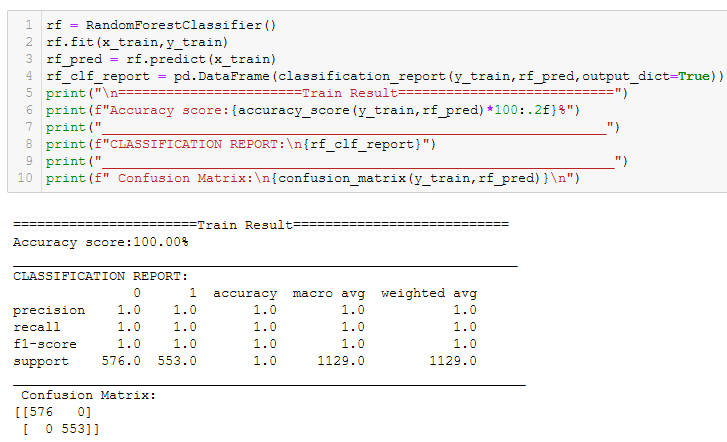
**Accuracy Score** is= 82.49

Now let’s visualize the confusion matrix.

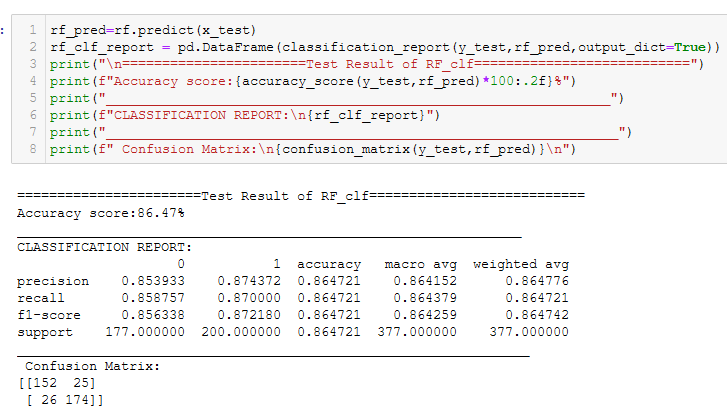


* Now let’s check the **AUC\_ROC score** as **ROC ( Receiver Operator Characteristic)** is a balance between true positive rate and false positive rate for a predictive model using different probability threshold. Hence, the **AUC\_ROC score of Decision Tree Classifier is 0.** **82**.

1. **Random Forest Classifier :**



Let’s do the prediction and the evaluation as well.



The confusion matrix is showing,

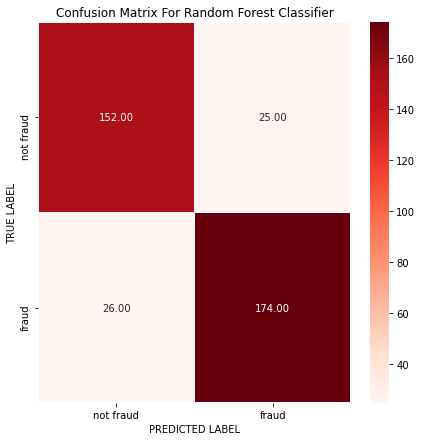
**True positive**=152, **False positive**= 25,

**False Negative** = 26, **True Negat**ive =174

Hence, the **type 1 error** is 25. **Precision** is 0.87 and 0.81**, recall** is 0.85 and 0. 87 And the **f1 Score** is 0.85 and 0.87

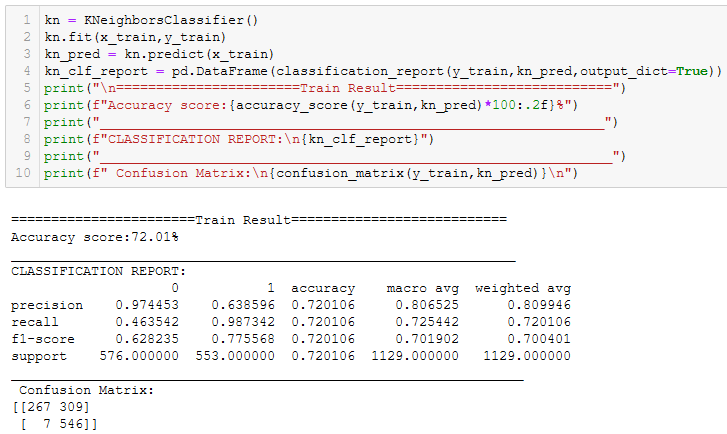
**Accuracy Score** is 86.47

Now let’s visualize the confusion matrix.

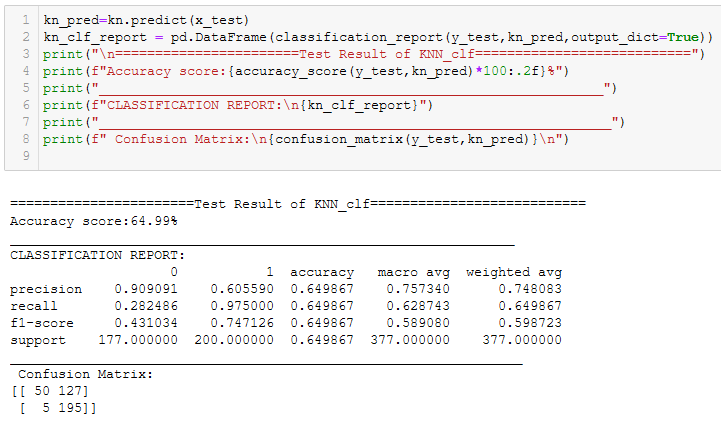


* Now let’s check the **AUC\_ROC score** as **ROC ( Receiver Operator Characteristic)** is a balance between true positive rate and false positive rate for a predictive model using different probability threshold. Hence, the **AUC\_ROC score of Random Forest Classifier is 0.** **86**.

1. **K-Neighbors Classifier :**



Now, let’s check the prediction and the evaluation as well.



The confusion matrix is showing, the

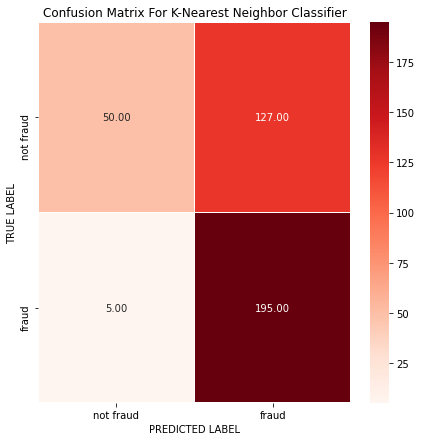
**True positive**=50, **False positive**= 127,

**False Negative** = 5, **True Negative** =195

Hence, the **type 1 error** is 127. **Precision** is 0.90 and 0.60, **recall** is 0.28 and 0. 97 And the **f1 Score** is 0.43 and 0.74

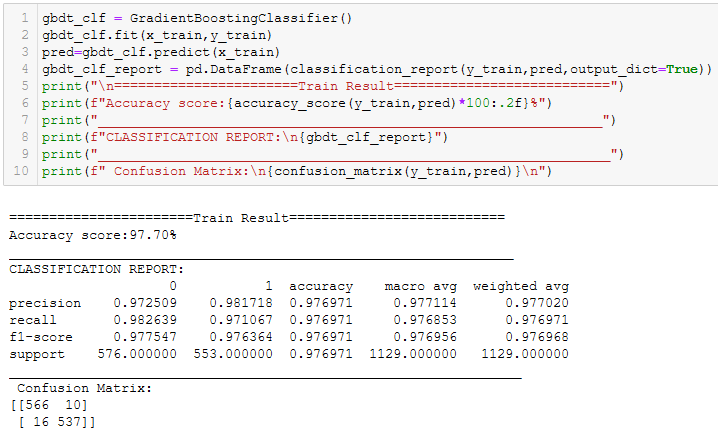
**Accuracy Score** is 64.99

Now let’s visualize the confusion matrix.

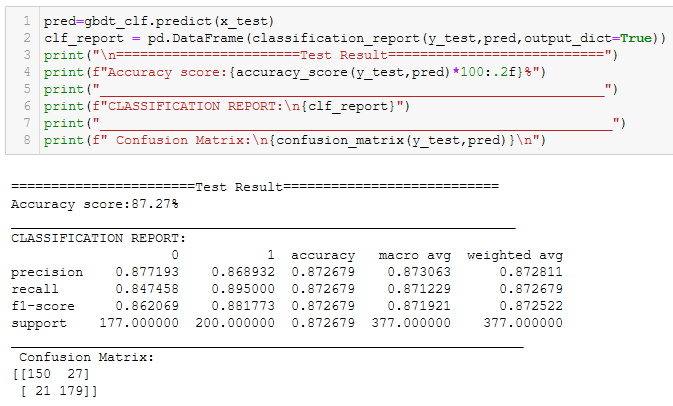


* Now let’s check the **AUC\_ROC score** as **ROC ( Receiver Operator Characteristic)** is a balance between true positive rate and false positive rate for a predictive model using different probability threshold. Hence, the **AUC\_ROC score of K-Neighbors Classifier is 0.** **62**

1. **Gradient Boosting Classifier**:



Now, let’s do prediction and evaluation.



The confusion matrix is showing, the

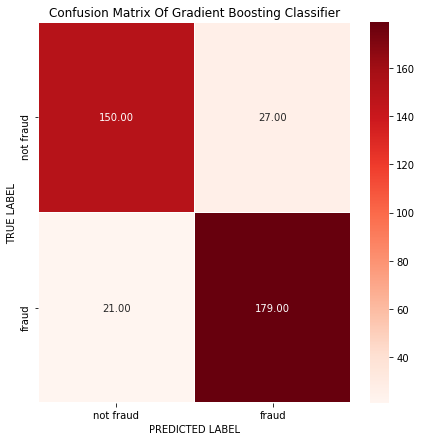
**True positive**=150, **False positive**= 27,

**False Negative** = 21, **True Negative** =179

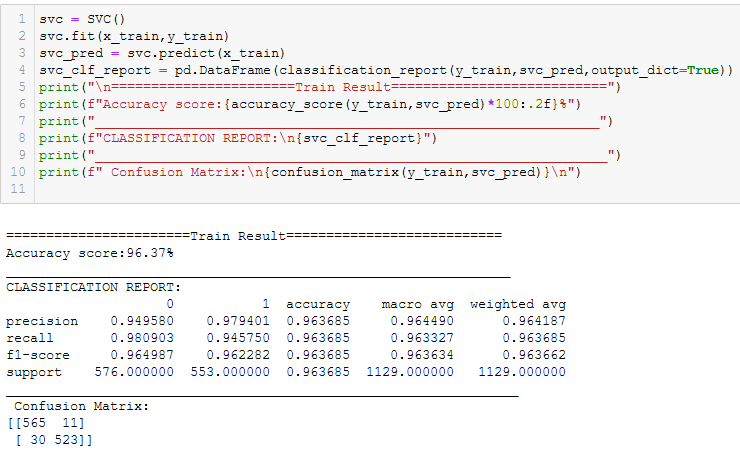
Hence, the **type 1 error** is 27. **Precision** is 0.87 and 0.86, **recall** is 0.84 and 0. 89 And the **f1 Score** is 0.86 and 0.88

**Accuracy Score** is 87.27

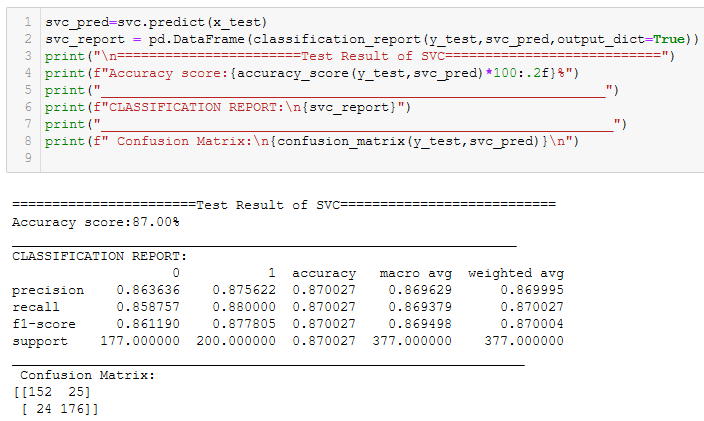
Now let’s visualize the confusion matrix.



* Now let’s check the **AUC\_ROC score** as **ROC ( Receiver Operator Characteristic)** is a balance between true positive rate and false positive rate for a predictive model using different probability threshold. Hence, the **AUC\_ROC score of Gradient Boosting Classifier is 0.** **87**

**6.** **Support Vector Classifier:**

Now let’s check the prediction and evaluation as well.



The confusion matrix is showing, the

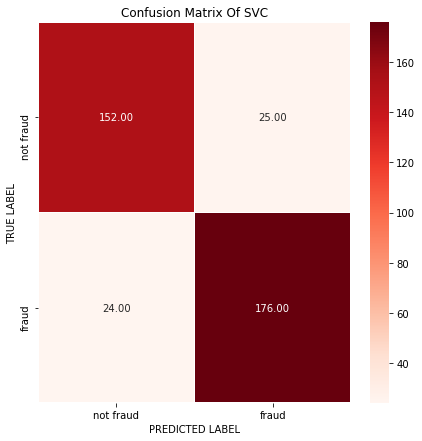
**True positive**=152, **False positive**= 25,

**False Negative** = 24, **True Negative** =176

Hence, the **type 1 error** is 25. **Precision** is 0.86 and 0.87, **recall** is 0.85 and 0. 88 And the **f1 Score** is 0.86 and 0.87

**Accuracy Score** is 87

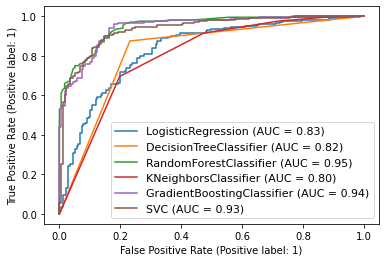
Now let’s visualize the confusion matrix.



* Now let’s check the **AUC\_ROC score** as **ROC ( Receiver Operator Characteristic)** is a balance between true positive rate and false positive rate for a predictive model using different probability threshold. Hence, the **AUC\_ROC score of Support Vector Classifier is 0.** **86**

**Observation:**

We have seen that **Gradient Boosting Classifier** has given the best accuracy of **87.27%** out of 6 different models. Let’s check the **AUC\_ROC Curve** for all the six models.



* **The Area Under Curve (AUC)** is showing **Random Forest Classifier** has most coverage under the curve. Let’s check the **Cross Validation** of all the Models to get the insight which one is the **best model**.

**Cross validation** : **Cross validation** is widely used technique to evaluate the performance of Machine Learning Models. Cross Validation **divides the data into parts**, where one set is used **for traning** and other is used **for testing purpose**. And by applying CV, we can understand if our **model is overfitting on training data or not.** 

Now let's check the **difference between Accuracy Score and Cross Validation Score** to select the best model from the 6 models:

Logistic Regression, 76.39-57.30 = **19.09**

Decision Tree Classifier , 82.49-80.95 = **1.54**

Random Forest Classifier ,86.47-85.60 = **0.87**

K-Neighbors Classifier , 64.99-69.32 = **- 4.33**

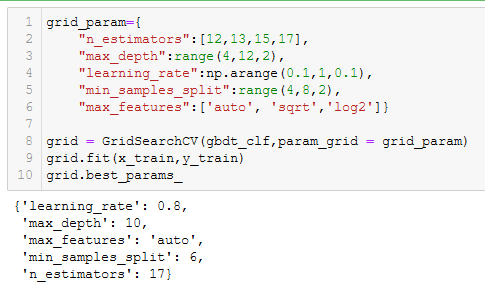
Gradient Boosting Classifier,87.27-86.93 = **0.34**

Support Vector Classifier , 87.00-60.29 = **27.71**,

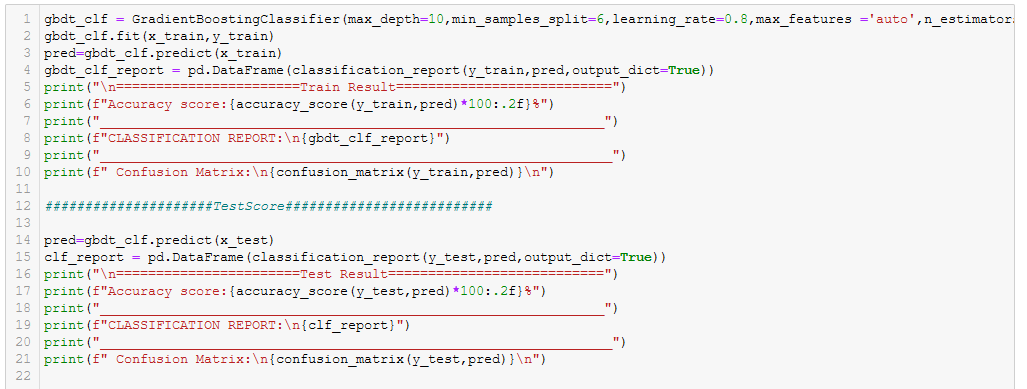
* As the **Accuracy score of Gradient Boosting Classifier is 87%** and as the **difference of Accuracy Score and Cross Validation of** **Gradient Boosting Classifier is least**, it's the **best Model**. Let's do **Hyperparameter Tuning** of **Gradient Boosting Classifier**.

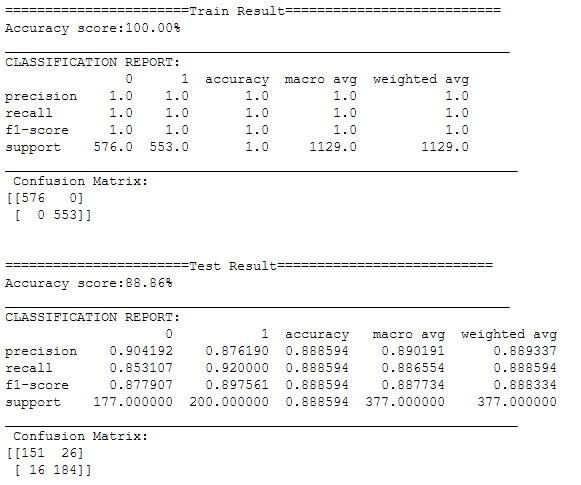
**Hyperparameter Tuning:**

Now we will use **Grid Search Technique** to tune the parameters of Gradient Boosting Classifier model. After getting the best parameters we can put those into the model to obtain best accuracy. We will tuning five hyperparameters right now. We are passing different values for all the parameters.



We have got the best parameters like**, learning rate=0.8, max depth=10, max features= auto, min samples split= 6, n estimators=17 .** now we will tuning the model with it’s best parameters.





**Observation:**

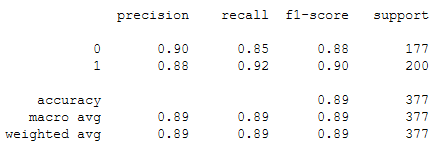
After tuning with best parameters of Gradient Boosting Classifier, we have got best **Accuracy of 88.86%**. The confusion matrix is showing, the

**True positive**=151, **False positive**= 26,

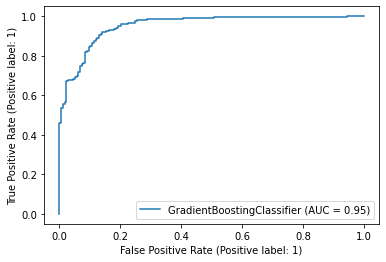
**False Negative** = 16, **True Negative** =184

Hence, the **type 1 error** is 26. **Precision** is 0.90 and 0.88, **recall** is 0.85 and 0. 92 And the **f1 Score** is 0.87 and 0.89

**Classification report** of **Gradient Boosting Classifier after Hyperparameter Tuning**:

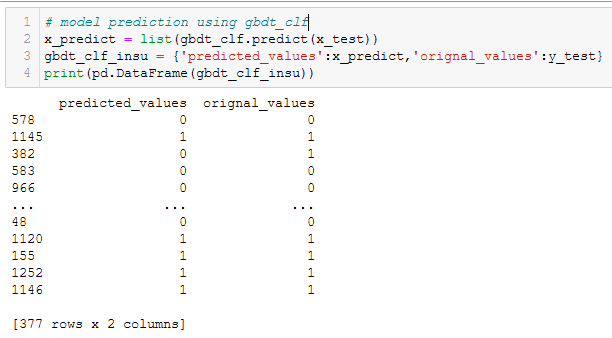


Let’s check the **Area Under Curve Of Gradient Boosting Classifier after hyperparameter tuning.**



* **The AUC curve of Gradient Boosting Classifier** is coving **95%** area under the curve.

So, the model Gradient Boosting Classifier has to be saved for future predictions of the Dataset. But before finally saving the model it had **to be tested** to check whether the **model is predicting correctly or not**.



As the model is predicting correctly. The model now can be saved using joblib library.

**Concluding Remarks:** In Modern life, Insurance is a vital thing for modern people as it has the capability to protect against financial loss. Insurance is a kind of contract that an individual can avail from banks or other insurance companies. A insurance comes with a policy, in which an individual has to pay premiums annually or monthly to protect his property or any important thing like vehicle, gadget and other so many things. But in this sector customers often do fraudulent claim to avail that insured money. But it’s not possible for the insurance company to check each and every claim manually. As there huge data involved.

So, by using this dataset we have analysed the data thoroughly and understand Some of the key points are as follows,

* Customer with **higher age**, the **fraud report** on the **claims reduce**.
* Fraud report is highest in **Chess** of all the insured **hobbies**.
* **Nissan** has got the highest **non-fraud reports**.
* In case of **Single Vehicle Collision** and **Multi-vehicle Collision** the **vehicle claim amount** is **more**.
* **Arlington** city has got the **highest fraud report** as incident city.
* The most surprising analysis is that **each of the category of each data are having more or less fraudulent claims**.

This kind of fraudulent activity can be check by the help of the **police department and technology like machine learning models**. Like in this case , doing **EDA and visualization we have summarized the dataset** and **have got idea about the important features**. And after **creation** **of the model** with **those key data**, our model predicted nicely. As **Gradient Boosting Classifier** has given the **best Accuracy of 88%** .