

Fraud is one of the largest and most well-known problems that insurers face. This article focuses on claim data of a car insurance company. Fraudulent claims can be highly expensive for each insurer. Therefore, it is important to know which claims are correct and which are not. It is not doable for insurance companies to check all claims personally since this will cost simply too much time and money. In this article, we will take advantage of the largest asset which insurers have in the fight against fraud: Data. We employ various attributes about the claims, insured people and other circumstances which are included in the data by the insurer. Separating different groups of claims and the corresponding rates of fraud within those groups provide new insights.

Furthermore, we use machine learning to predict which claims are likely to be fraudulent. This information can narrow down the list of claims that need a further check. It enables an insurer to detect more fraudulent claims.

**Problem Definition:**

The goal of this project is to build a model that can detect auto insurance fraud. The challenge behind fraud detection in machine learning is that frauds are far less common as compared to legit insurance claims.

Insurance fraud detection is a challenging problem, given the variety of fraud patterns and relatively small ratio of known frauds in typical samples. While building detection models, the savings from loss prevention needs to be balanced with the cost of false alerts. Machine learning techniques allow for improving predictive accuracy, enabling loss control units to achieve higher coverage with low false positive rates.

Insurance frauds cover the range of improper activities which an individual may commit in order to achieve a favourable outcome from the insurance company. This could range from staging the incident, misrepresenting the situation including the relevant actors and the cause of incident and finally the extent of damage caused.

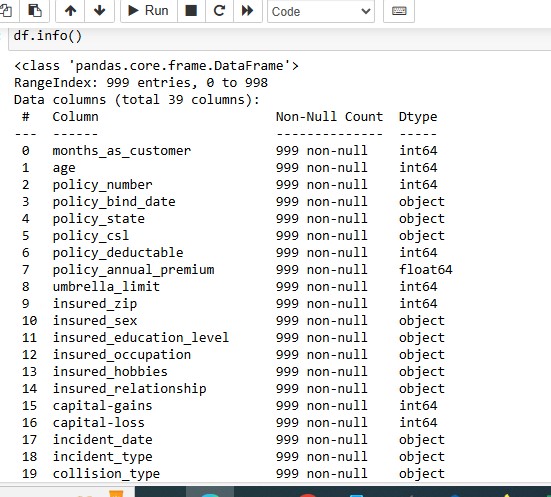
**Independent Variables**

1. Months-as-customer: Number of months of patronage
2. age: the length of time a customer has lived or a thing has existed
3. policy-number: It is a unique id given to the customer, to track the subscription status and other details of customer
4. policy-bind-date : date which document that is given to customer after we accept your proposal for insurance
5. policy-state: This identifies who is the insured, what risks or property are covered, the policy limits, and the policy period
6. policy-cs l: is basically Combined Single Limit
7. policy-deductions: the amount of money that a customer is responsible for paying toward an insured loss
8. policy-annual-premium: This means the amount of Regular Premium payable by the Policyholder in a Policy Year
9. umbrella-limit: This means extra insurance that provides protection beyond existing limits and coverages of other policies
10. insured-zip: It is the zip code where the insurance was made
11. insured-ex: This refers to either of the two main categories (male and female) into which customer are divided on the basis of their reproductive functions
12. insured-education-level: This refers to the Level of education of the customer
13. insured-occupation: This refers Occupation of the customer

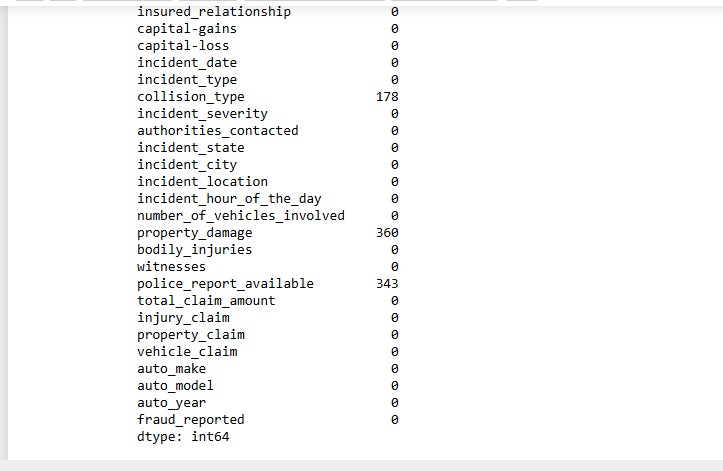
# Data Analysis

In this project, we have a dataset which 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.

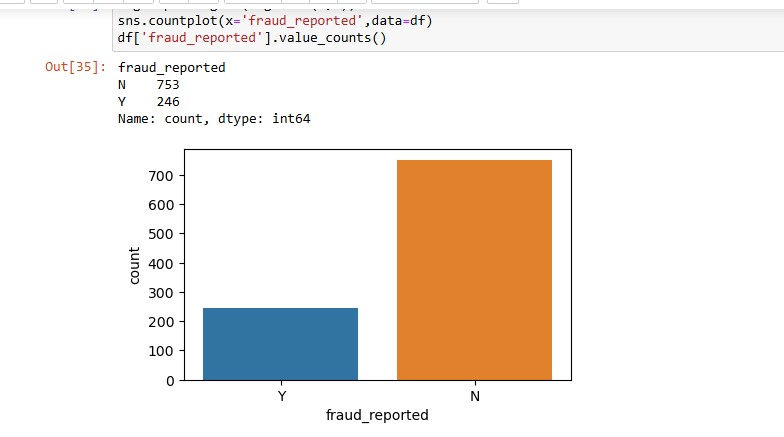
The given dataset contains 999 rows and 39 columns. The column names like policy number, policy bind date, policy annual premium, incident severity, incident location, auto model, etc.



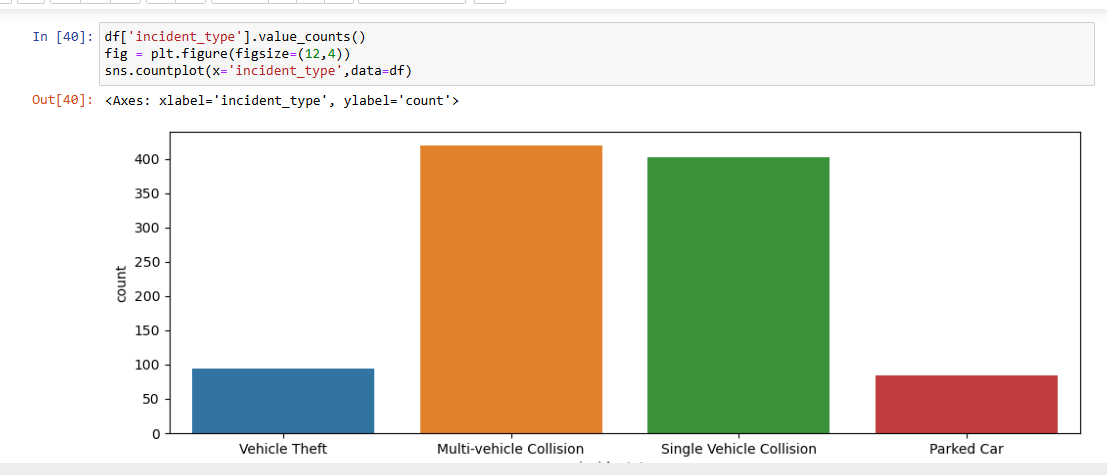
.. There are some variables which contain the null values character ‘?’. The number of null values present is given below.



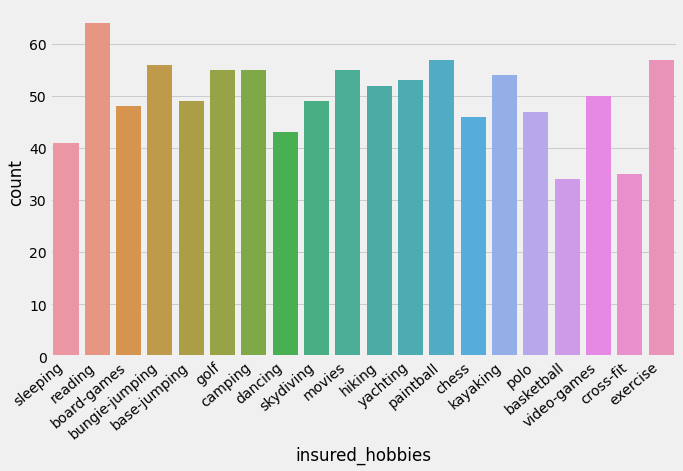
# Exploratory data analysis

* **Dependent variable:** Exploratory data analysis was conducted starting with the dependent variable, Fraud reported. There were 246 frauds and 753 non-frauds. 24.7% of the data were frauds while 75.3% were non-fraudulent claims.
* 

Incident type:

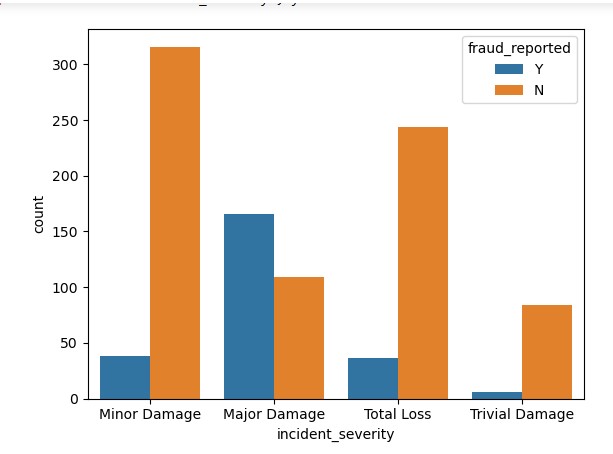


* **Visualizing variables:**The value of fraud reported differs across hobbies of the customer. It seems like chess players and cross fitters have higher tendencies to fraud.

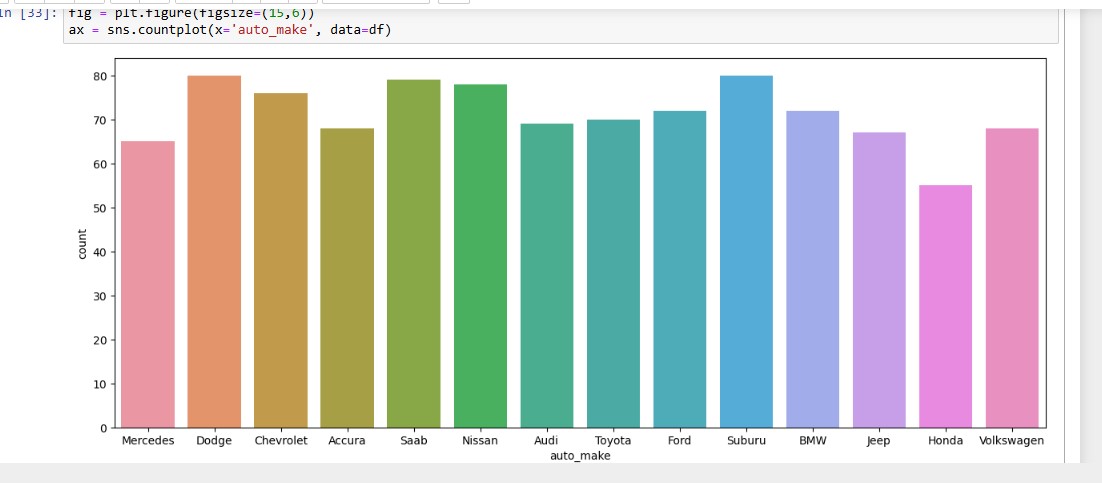


Hobbies of customers with respect to frauds committed

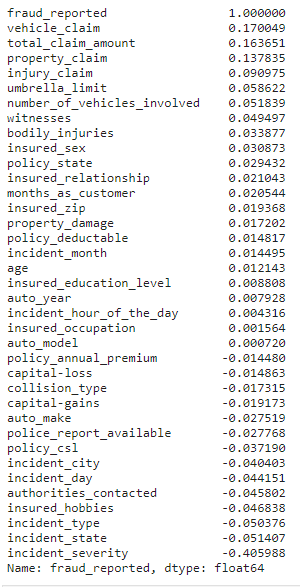
*Major incidents severity seems to have the highest fraud cases that exceeds non fraud cases:*



*The total claim amount is high in Saab and Subaru and Dodge auto make.*



Checking correlation between dependent and independent variables.



## **Pre-processing Pipeline**

Data pre-processing is a predominant step in machine learning to yield highly accurate and insightful results. Greater the quality of data, the greater is the reliability of the produced results. **Incomplete, noisy, and inconsistent data** are the inherent nature of real-world datasets. Data pre-processing helps in increasing the quality of data by filling in missing incomplete data, smoothing noise, and resolving inconsistencies.

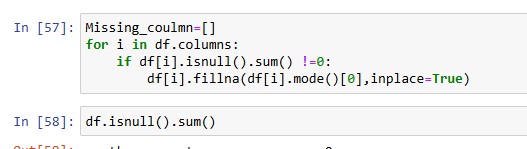
There are many stages involved in data pre -processing.

* **Data cleaning** attempts to impute missing values, removing outliers from the dataset.
* **Data integration**integrates data from a multitude of sources into a single data warehouse.
* **Data transformation**such as normalization, may be applied. For example, normalization may improve the accuracy and efficiency of mining algorithms involving distance measurement.
* **Data reduction**can reduce the data size by dropping out redundant features. Feature selection and feature extraction techniques can be used.

**Treating null values**

Sometimes there are certain columns which contain the null value used to indicate missing or unknown values or maybe the value doesn’t exist. In our dataset the null values are present in columns collision-type, property-damage, police-report-available, and \_c39 with 178, 360, 343 and 1000 number of null values.

There are different ways of replacing null values from the dataset, but we are using fillna to replace the null values from our data.





**Converting labels into numeric**

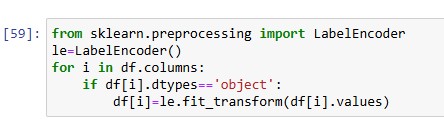
In machine learning, we usually deal with datasets which contain multiple labels in one or more than one column. These labels can be in the form of words or numbers. To make the data understandable or in human readable form, the training data is often labelled in words.

In our data there are columns with categorical values. The columns like incident-severity, incident-state, incident-type, insured-hobbies, authorities-contacted, incident-city, police-report-available, auto-make, collision-type, auto-model, insured-occupation, insured-education-level, property-damage, insured-relationship, policy-state, insured-sex, fraud-reported. These columns have to be treated with one hot encoding or the label encoder. The target variable fraud-reported has to convert by using label encoder only.

**Label Encoder:**

**Label Encoder**refers to converting the labels into numeric form so as to convert it into the Machine-readable form. Machine learning algorithms can then decide in a better way on how those labels must be operated. It is an important Pre-processing step for the structured dataset in supervised learning.

Label encoding in python can be imported from the Sk-learn library. Sk-learn provides a very efficient tool for encoding. Label encoders encode labels with a value between 0 and n\_classes-1.



***Outliers :****are data points that are distant from other similar points. They may be due to variability in the measurement or may indicate experimental errors. If possible, outliers should be excluded from the data set. However, detecting that anomalous instance might be very difficult, and is not always possible.*

**Methods to remove outliers:**

### **Z-score:**

[Z score](https://www.geeksforgeeks.org/z-score-in-statistics/) is also called a standard score. This value/score helps to understand that how far is the data point from the mean. And after setting up a threshold value one can utilize z score values of data points to define the outliers.

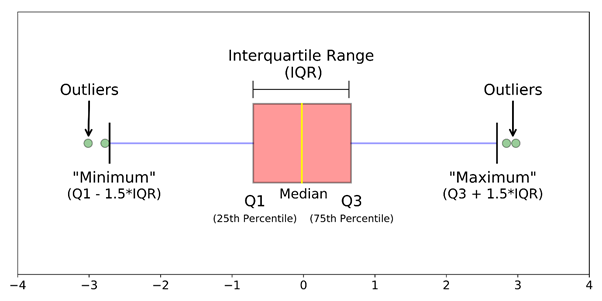
*Z score = (data point -mean) / std. deviation*

### **IQR (Inter Quartile Range):**

### Inter Quartile Range approach to finding the outliers is the most commonly used and most trusted approach used in the research field.

*IQR = Quartile3 – Quartile1*

* *upper = Q3 +1.5\*IQR*
* *lower = Q1 – 1.5\*IQR*



* Calculate the interquartile range for the data by using scipy.stats.iqr module.
* Multiply the interquartile range by 1.5.
* Add 1.5 x interquartile range to the third quartile. Any number greater than this is a suspected outlier.
* Subtract 1.5 x interquartile range from the first quartile. Any number lesser than this is a suspected outlier.

# ****Building machine learning models****

For building machine learning models there are several models present inside the Sklearn module.

Sklearn provides two types of models i.e. regression and classification. Our dataset’s target variable is to predict whether fraud is reported or not. So for this kind of problem we use classification models.

But before fitting our dataset to its model first we have to separate the predictor variable and the target variable, then we pass this variable to the train-test-split method to create a random test and train subset.

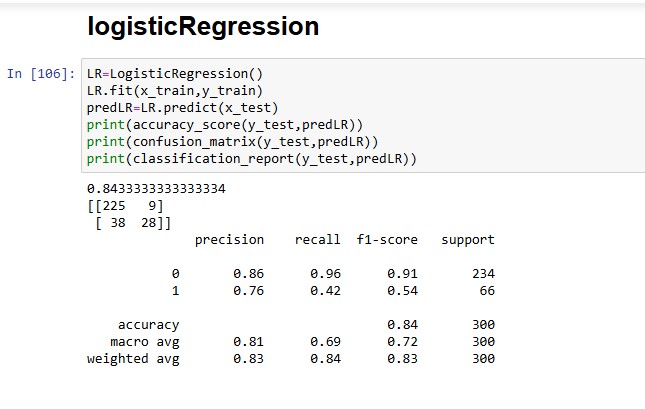
***What is train-test-split****, it**is a function in sklearn model selection for splitting data arrays into two subsets for training data and testing data. With this function, you don’t need to divide the dataset manually. By default, sklearn train-test-split will make random partitions for the two subsets. However, you can also specify a random state for the operation. It gives four outputs x-train, x-test, y-train and y-test. The x-train and x-test contains the training and testing predictor variables while y-train and y-test contains the training and testing target variable.*

After performing train-test- split we have to choose the models to pass the training variable.

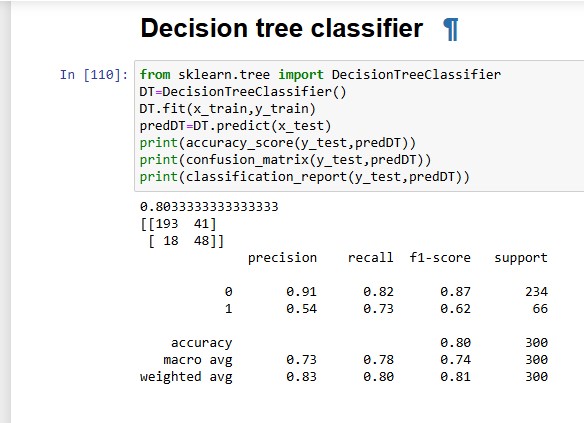
We can build as many models as we want to compare the accuracy given by these models and to select the best model among them.

I have selected 5 models:

**Logistic Regression:** Logistic regression is a supervised learning classification algorithm used to predict the probability of a target variable. The nature of target or dependent variable is binary, which means there would be only two possible classes 1 (stands for success/yes) or 0 (stands for failure/no). Mathematically, a logistic regression model predicts P(Y=1) as a function of X. It is one of the simplest ML algorithms that can be used for various classification problems such as spam detection, Diabetes prediction, cancer detection etc.



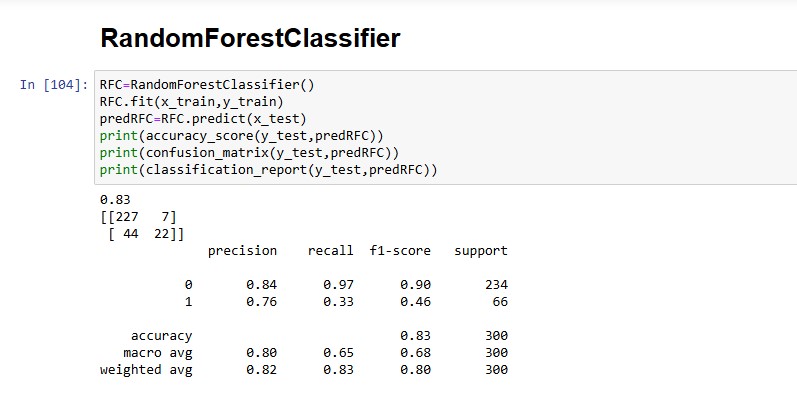
**Decision Tree Classifier:** Decision trees can be constructed by an algorithmic approach that can split the dataset in different ways based on different conditions. The two main entities of a tree are decision nodes, where the data is split and leaves, where we get the outcome.



**Random Forest Classifier:**

The Random Forest classifier creates a set of decision trees from a randomly selected subset of the training set. It is a set of decision trees (DT) from a randomly selected subset of the training set and then It collects the votes from different decision trees to decide the final prediction.

 As we know that a forest is made up of trees and more trees means more robust forest. Similarly, a random forest algorithm creates decision trees on data samples and then gets the prediction from each of them and finally selects the best solution by means of voting. It is an ensemble method which is better than a single decision tree because it reduces the over-fitting by averaging the result.

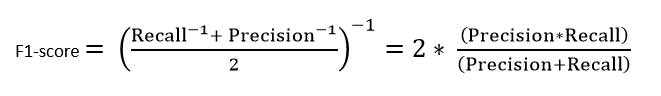


# ****Conclusion from models****

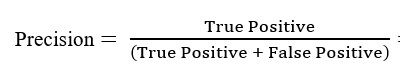
We got our best model i.e. RandomForestClassifier with the accuracy score of 85.39%. Here our model predicts 196 true positive cases out of 218 positive cases and 190 true negative cases out of 234 cases. It predicts 22 false positive cases out of 218 positive cases and 44 false negative cases out of 234 cases. It gives the f1 score of 85.20%.

**Understand what does precision recall and f1 score and accuracy do**

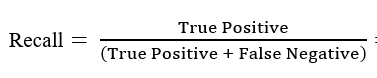
**F1 score**: this is the harmonic mean of precision and recall and gives a better measure of the incorrectly classified cases than the accuracy matrix.



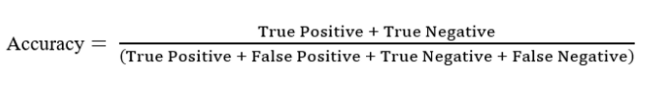
**Precision:**It is implied as the measure of the correctly identified positive cases from all the predicted positive cases. Thus, it is useful when the costs of False Positives are high.



**Recall:**It is the measure of the correctly identified positive cases from all the actual positive cases. It is important when the cost of False Negatives is high.

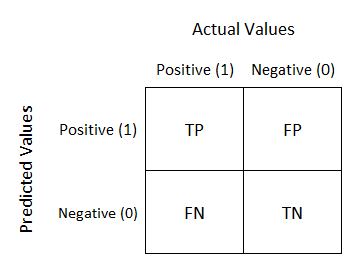


**Accuracy:**One of the more obvious metrics, it is the measure of all the correctly identified cases. It is most used when all the classes are equally important.



**Confusion matrix:**

A table that is often used to describe the performance of a classification model (or ‘classifier’) on a set of test data for which the true values are known.



***NOTE:***

***TN/True Negative:****the cases were negative and predicted negative.*

***TP/True Positive:****the cases were positive and predicted positive.*

***FN/False Negative:****the cases were positive but predicted negative.*

***TN/True Negative:****the cases were negative but predicted positive.*

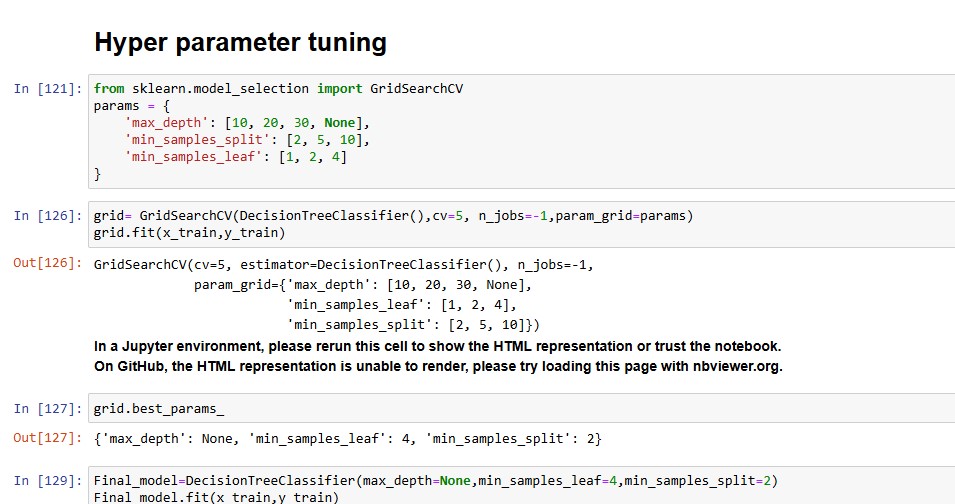
## **Hyperparameter Tuning**

Hyperparameter tuning is the process of selecting the optimal values for Machine learning model’s hyperparameters. Hyperparameters are settings that control the learning process of the model, such as the learning rate, the number of neurons in a neural network, or the kernel size in a support vector machine. The goal of hyperparameter tuning is to find the values that lead to the best performance on a given task.

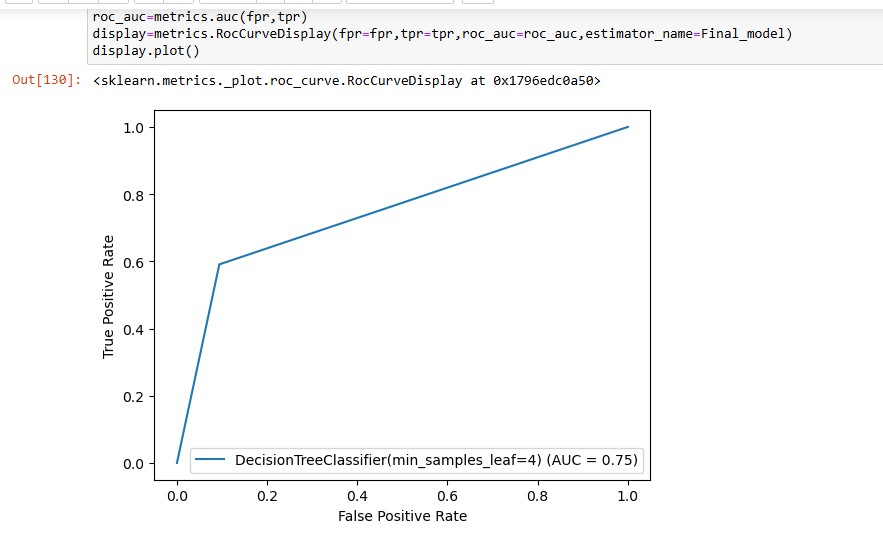
We will use Grid Search CV for the hyper parameter tuning.

**Grid Search CV:**

Grid search CV is the process of performing hyperparameter tuning in order to determine the optimal values for a given model. As mentioned above, the performance of a model significantly depends on the value of hyperparameters.



**ROC curve:**Itis a performance measurement for the classification problems at various threshold settings. ROC is a probability curve and AUC represents the degree or measure of separability. It tells how much the model is capable of distinguishing between classes. Higher the AUC, the better the model is at predicting 0s as 0s and 1s as 1s. By analogy, the Higher the AUC, the better the model is at distinguishing between patients with the disease and no disease.



# ****Remarks****

This project has built a model that can detect auto insurance fraud. In doing so, the model can reduce losses for insurance companies. The challenge behind fraud detection in machine learning is that frauds are far less common as compared to legit insurance claims.

Five different classifiers were used in this project: logistic regression, Bagging classifier, Random Forest, Decision tree, SVM. Four different ways of handling imbalance classes were tested out with these five classifiers: model with class weighting, oversampling with SMOTE, hyper parameter tuning, and plotting roc curve of the models.

The best and final fitted model was a weighted Decision Tree that yelled a F1 score of *0.83* and a ROC AUC of *0.75*. The model performed excellent. The model’s F1 score and ROC AUC scores were the highest amongst the other models. In conclusion, the model was able to correctly distinguish between fraud claims and legit claims with high accuracy.

The study is not without limitations. Firstly, this study is restricted by its small sample size. Statistical models are more stable when data sets are larger. It also generalises better as it takes a bigger proportion of the actual population. Furthermore, the data only capture incident claims of 3 states.