[Insurance Claims- Fraud Detection]



Submitted By:

TRIBENI SUBUDHI

**Introduction:**

Auto Insurance Industry is engaged in providing vehicle insurance services to the customers. Auto insurance or vehicle insurance is a contract between the customers and the insurance company that protects the financial losses in the event of unfortunate conditions like accident or theft. Auto or vehicle insurance provides coverage for the following 3 main categories:

1. **Property:** such as accidental damage or theft of vehicle.
2. **Liability:** legal responsibilities of the customers to others for injury or property damage.
3. **Medical:** this mainly consider the medical cost of injuries, rehabilitation, and sometimes lost wages and funeral expenses.

Insurance fraud is a major problem of the auto industry. It is very difficult for companies to identify the fraud claims. The main objective of this project is to make a machine learning model to identify the fraud claims made by the customers. Machine learning is a very unique method which we can use to help the auto insurance industry to identify the fraud claims. This will help the industry to run the service smoothly and increase their profit.

Insurance fraud covers the range of activities that the customers commit in order to get the favourable outcome from the insurance company. This may include false claims, manipulating the incident to get maximum benefit etc.

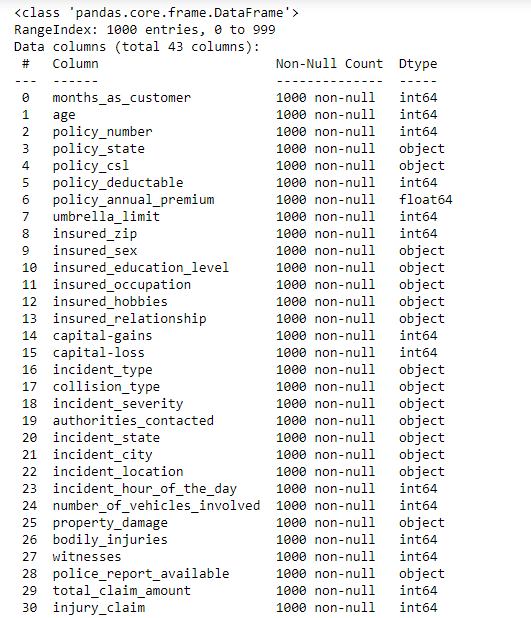
In this project we have been working with some auto insurance data to demonstrate how we can create a predictive model that predicts if an insurance claim is fraudulent or not.

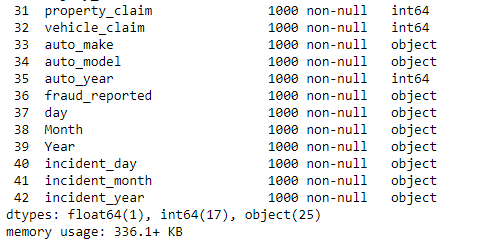
**Data Analysis:**

We have been given with a dataset which contains 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.

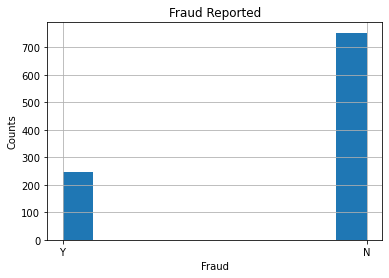
The provided dataset contains 1,000 rows and 40 columns including the target column. Some of the column’ name in the dataset are policy number, age, month as customer, policy bind date etc.

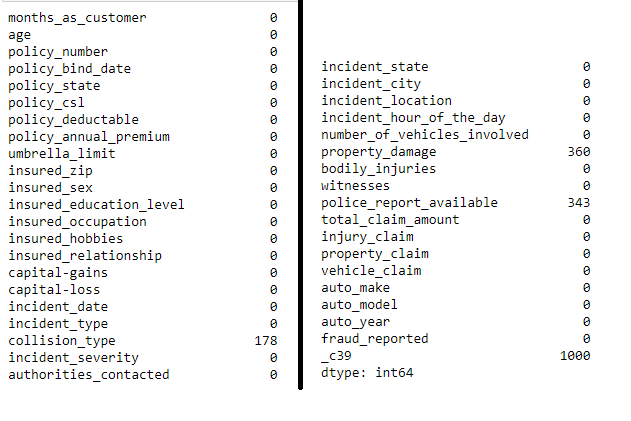
Since the dataset contains date column, we have split the date column into day, month, and year for the better understanding of the prediction model.





As per our observations, we found that some null values or missing values were present in the dataset as values ‘?’. Let’s see the columns along with the number of missing values (null values).





Columns having the null values are:

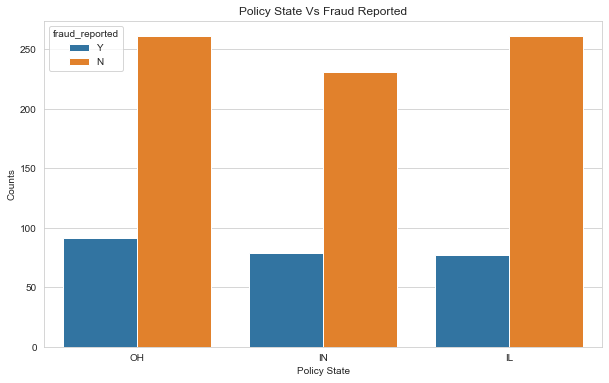
* 1. Collision\_ type - 178 null values
  2. Property\_ damage - 360 null values
  3. Police\_report\_available - 343 null values
  4. \_c39 - 1000 null values

**Exploratory Data Analysis (EDA) :**

* **Target Column (Dependent variable):**

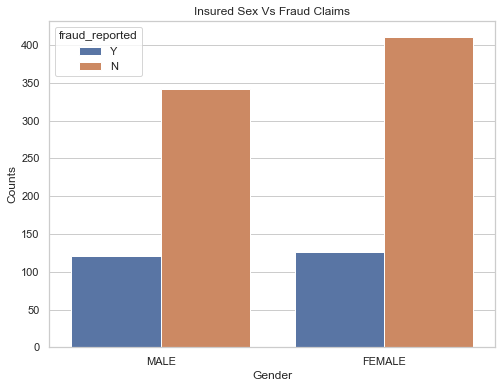
Let’ check if the data is balanced or not of the target column. Fraud\_ reported is the target column. We found that the data is imbalance in the target column. The total number of fraud claims made is 247 which is 24.7% of the total claims made by the customers.

* **State-wise distribution of insurance claims:**



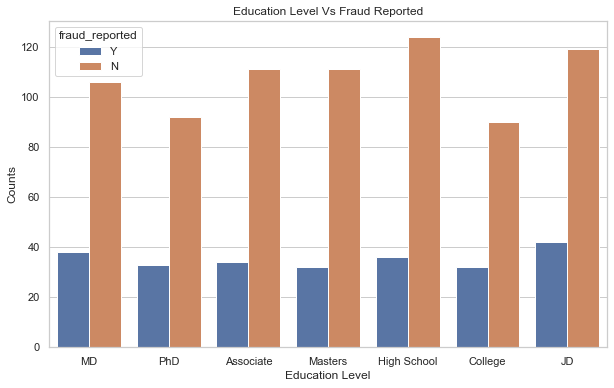
From the graph we can make the observation that the distribution of fraud claims made by the customers are almost equal for all the policy states. It is highest for the state Ohio (OH) state.

* **Distribution of Male & Female:**



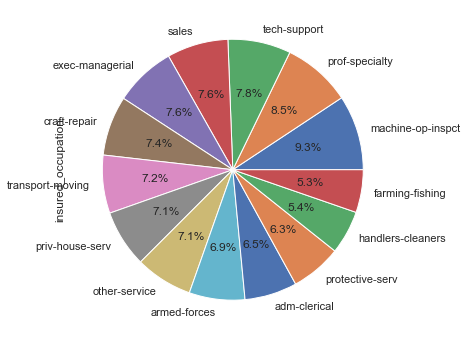
As per the dataset, the number of female customers is more than the males. This implies that the female is more interested in taking the insurance. If we talk about the number of frauds, it is almost equal for both the genders. Also, we can say that the ratio of fraud claims is more for the males as compared with the females.

* **Distribution of Education level:**



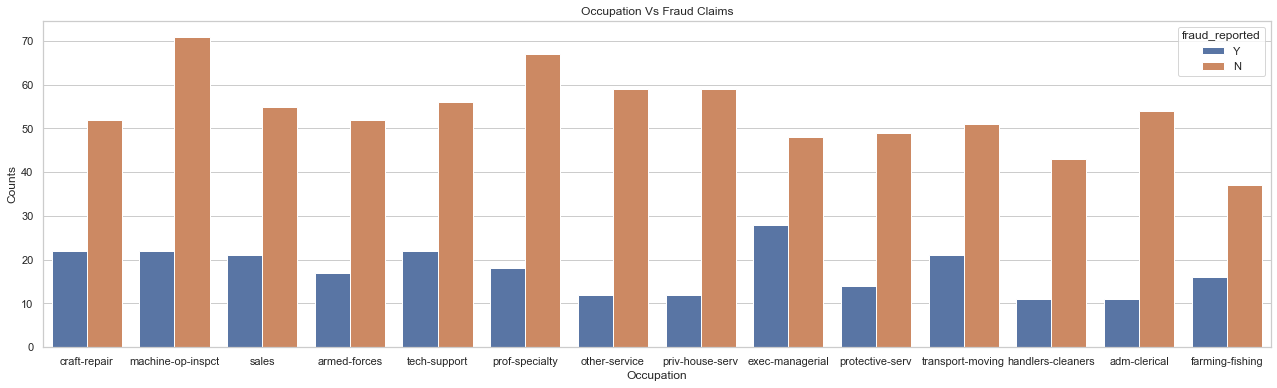
If we look at the above graph, we draw the observation that the highest number of fraud claims are made by the person having the education level of JD.

* **Occupation of the customers:**



We can see that the dataset has almost equal number of customers working in different sectors. Highest number of persons having the occupation machine-op-inspect. Persons involved in farming and fishing having the least number in the dataset, this shows that the persons involved in agriculture sector are least interested to opt for any insurance plan.

Let’s check for the distribution of fraud claims made by the persons from different occupation.



The observations shows that the persons having the occupation of exec-managerial have made highest number of frauds claims among all other.

**Data Pre-processing:**

Data pre-processing is very important step in Machine Learning to get highly accurate and insightful results. The greater quality of data will help us to make highly efficient model. This will lead to the highly reliability of the predicted results.

**Incomplete, Noisy, and Inconsistent data** will lead to the inherent nature of real-world problems. Data pre-processing help us in increasing the quality of data by treating the missing values, removing the noises, and resolving inconsistency.

* **Incomplete Data:** There can be a number of reasons for this. There may be some misunderstanding or technical fault which lead to the incomplete data.
* **Noisy Data:** This can be the incorrect feature values present in the dataset. There can be many reasons for this. The instrument used for the data collection might be faulty. There may be human error at the time of data entry.

**Stages of Data Pre-processing:**

* **Data Cleaning:** In this stage we work on the dataset to remove the unwanted feature columns (columns which are not contributing to the dataset), treating the missing or null values, removing the outliers and skewness if present.
* **Data integration:** integrates data from a multitude if source into 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:** Reduce the data size by dropping out redundant features. Feature selection and feature extraction technique can be used.

**Treating the null values:**

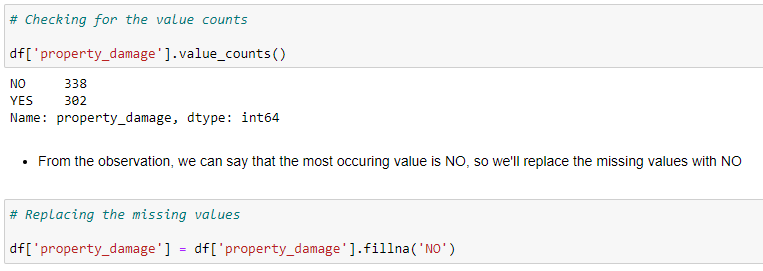
There might be some columns having the missing or unknown values commonly known as null values. We need to remove the missing or null values of the columns to make a good model. Also, sometime we may encounter some columns having no values means the column might be completely empty. Since these columns are not contributing to the dataset so, it is good to remove the unwanted columns from the dataset.

In our dataset the following columns contains the missing values or null values:(As said earlier also)

1. Collision\_type - 178 null values
2. Property\_damage - 360 null values
3. Police\_report\_available - 343 null values
4. \_c39 - 1000 null values

We have dropped the column \_c39 since it was empty (all the rows having null values). Also, we’ve replaced the missing values of the columns collision\_type, property\_damage, and police\_report\_available with the mode values.

There are many ways to remove the missing values but we are using **fillna method** to replace the missing values from our dataset.

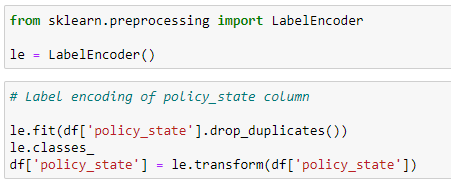


**Label Encoding: (Converting the categorical features to numerical)**

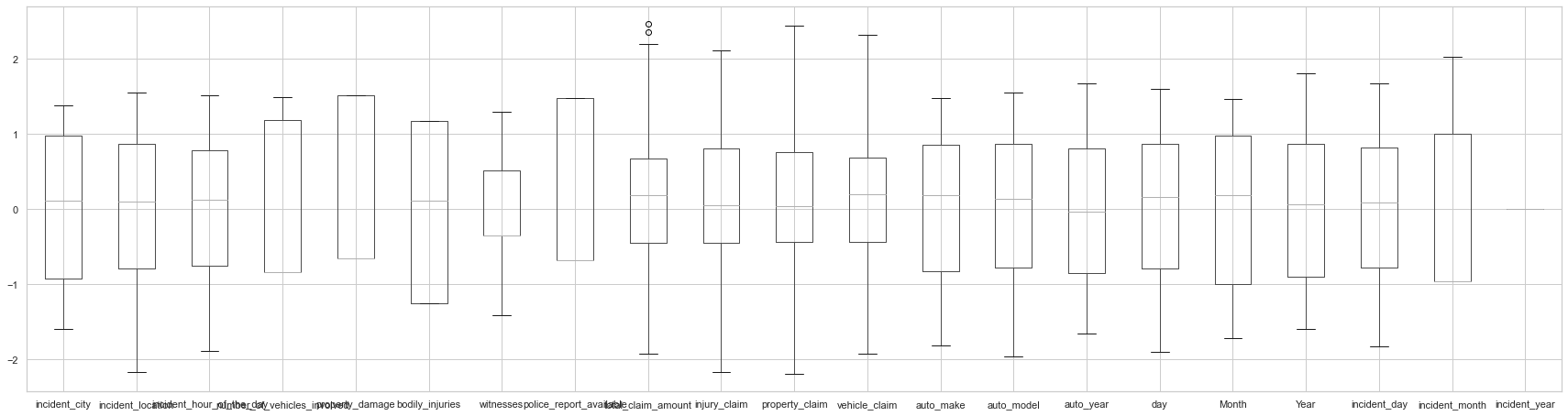
In dataset there can be some columns which will be containing the categorical columns (columns with the datatype of object). For machine learning model we need to convert the categorical data to numerical. We can use label encoding or one hot encoding methods for the conversion.

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. We are treating these columns with the label encoding method.

We can find the Label Encoding in skLearn library, so we can import it. skLearn provides very efficient tool for encoding. Label encoder encode labels with a value between 0 and n\_classes-1. Let’s see an example from our dataset:



**Outliers:** We can define outliers as the data point which is distant from the 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.



We can remove outliers using z-score method. Since there were very few outliers present in our dataset, we gave ignored them.

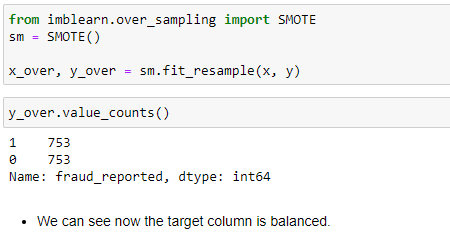
**Balancing the imbalanced dataset:**

We have seen earlier that; our dataset was imbalance. There are different methods available to make the data balanced. We have used SMOTE method.

**SMOTE** (Synthetic Minority Oversampling Technique) works by randomly picking a point from the minority class and computing the k-nearest neighbours of this point. The synthetic points are added between the chosen point and its neighbours.

SMOTE algorithm works in 4 simple steps: -

1. Choose a minority class as input vector.
2. Find its k-nearest neighbours.
3. Choose one of these neighbours and place a synthetic point anywhere on the line joining the point under consideration and its chosen neighbours.
4. Repeat the step until the data is balanced.



Originally our target column fraud\_reported was having 753 with NO value (Labeled as ‘0’) and 247 as YES values (Labeled as ‘1’). The SMOTE algorithm has made it to 753 (the highest value) for both.

**Machine Learning Model Building:**

We can find many types of methods for model building in the skLearn library.

There are two types of models which are present in the skLearn library: 1. Regression and 2. Classification

In the case of our dataset, we have to make machine learning model to predict whether the claim made by the customers are fraud or not. Since we have only two values, so we will use classification models to build the machine learning model.

Before fitting the data to the model, we will need to first separate the dependent variable and independent variables, then we will pass these values to train\_test\_split to generate random training and testing subset of the data.

**About train\_test\_split method:** It is a function, which we can find in skLearn model selection library. We use this function for splitting data arrays into two subsets for training data and testing data. By using this function, we can avoid the manual task to separate the dataset. By default, sklearn train\_test\_split will make random partitions for the two subsets. However, we can also specify a random state for the operation. It gives four output 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.

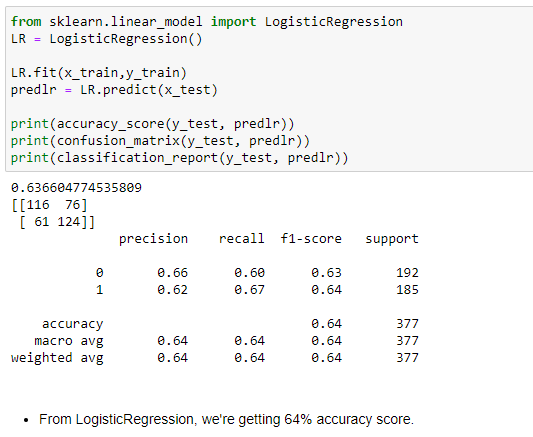
After performing train\_test\_split we have to choose the models to pass the training variable.

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

We have made 4 models to get the closest possible accuracy score to predict the fraud claims.

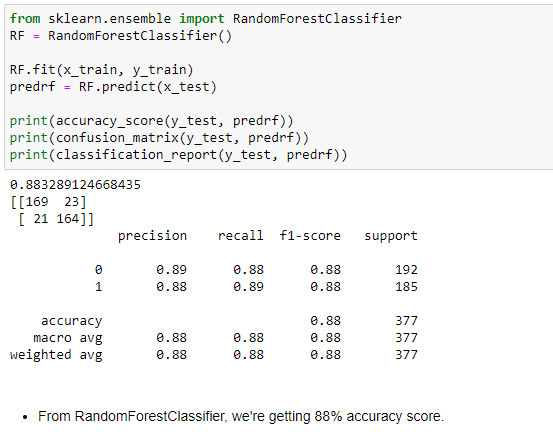
1. **Logistic Regression:**

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.



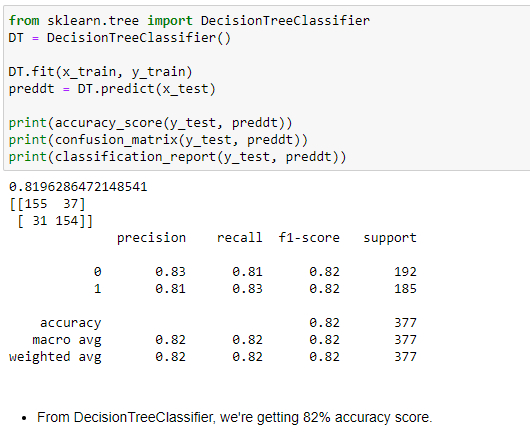
1. **Random Forest Classifier:**

As we know that a forest is made up of trees and more trees means more robust forest. Similarly, 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.



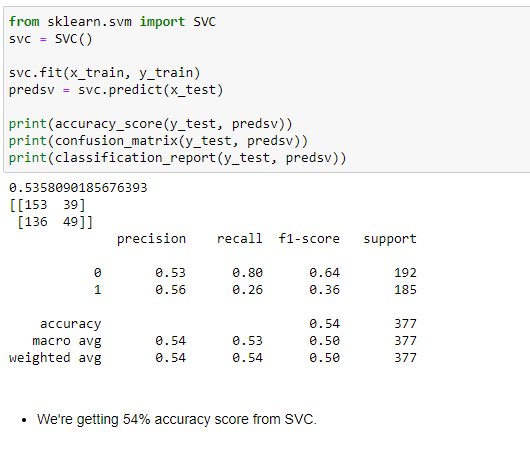
1. **Decision Tree Classifier:**

Decision Tree Classifier is a class capable of performing multi-class classification on a dataset. As with other classifiers, [DecisionTreeClassifier](https://scikit-learn.org/stable/modules/generated/sklearn.tree.DecisionTreeClassifier.html" \l "sklearn.tree.DecisionTreeClassifier" \o "sklearn.tree.DecisionTreeClassifier) takes as input two arrays: an array X, sparse or dense, of shape (n\_samples, n\_features) holding the training samples, and an array Y of integer values, shape (n\_samples,), holding the class labels for the training samples. It is capable of both binary (where the labels are [-1, 1]) classification and multi-class (where the labels are [0, …, K-1]) classification.



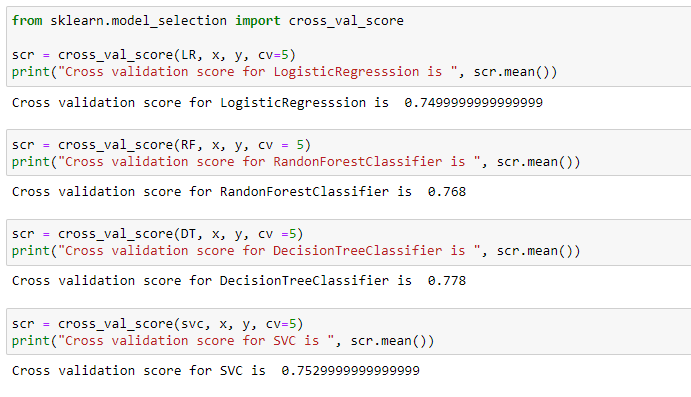
1. **SVC: (Support Vector Classifier)**

SVC is a non-parametric clustering algorithm that does not make any assumption on the number or shape of the clusters in the data. In SVC data points are mapped from data space to a high dimensional feature space using a kernel function. In the kernel's feature space the [algorithm](http://www.scholarpedia.org/article/Algorithm) searches for the smallest sphere that encloses the image of the data using the [Support Vector Domain Description](http://www.scholarpedia.org/w/index.php?title=Support_Vector_Domain_Description&action=edit&redlink=1) algorithm. This sphere, when mapped back to data space, forms a set of contours which enclose the data points. Those contours are then interpreted as cluster boundaries, and points enclosed by each contour are associated by SVC to the same cluster.



If we compare all the models, we found that we are getting maximum accuracy score from Random Forest Classifier. But we cannot decide the best model on the basis of accuracy score only, since this might be possible that our data is over-fitted.

So, to decide the best fit model we will check for the cross-validation score of each model. We can import cross\_val\_score from the skLearn library (sklearn.model\_selection).



We will first calculate the difference between the accuracy score and cross validation to decide the best fit model for our prediction. The model having the lowest difference between the accuracy score and cross validation score will be the best model for our machine learning algorithm.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| S. No. | Model Name | Accuracy Score | Cross Validation Score | Difference |
| 1 | Logistic Regression | 63.66 | 74.99 | -11.33 |
| 2 | Random Forest Classifier | 88.32 | 76.8 | 11.52 |
| 3 | Decision Tree Classifier | 81.96 | 77.8 | 4.16 |
| 4 | SVC | 53.58 | 75.29 | -21.71 |

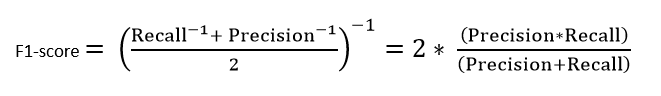
**Conclusion:**

From the above table we can say that the best fit model is Decision Tree Classifier since it is showing minimum difference between the accuracy score and cross validation score.

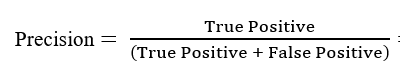
* Our model is showing approx. 82% accuracy score. It has predicted 155 true positive cases out of 192 positive cases and 154 true negative cases out of 185 cases.
* It has predicted 37 false positive cases out of 192 positive cases and 31 false negative cases out of 185 cases.
* It has given the f1 score of approx. 82%.

Let’s understand what does precision recall and f1 score and accuracy means.

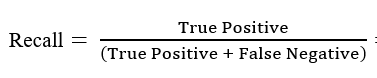
* **F1 score:** It is the harmonic mean of precision and recall and gives a better measure of the incorrectly classified cases than the accuracy matrix.



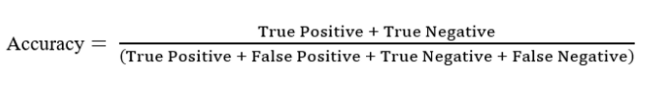
* **Precision:** It is implied as the measure of the correctly identified positive cases from all the predicted positive cases. Thus, it is useful when the costs of False Positives are high.



* **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.

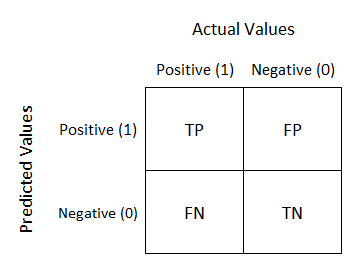


* **Accuracy:** One of the more obvious metrics, it is the measure of all the correctly identified cases. It is most used when all the classes are equally important.



**Let’s understand what confusion matrix is:**

A confusion matrix is 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.



* **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.

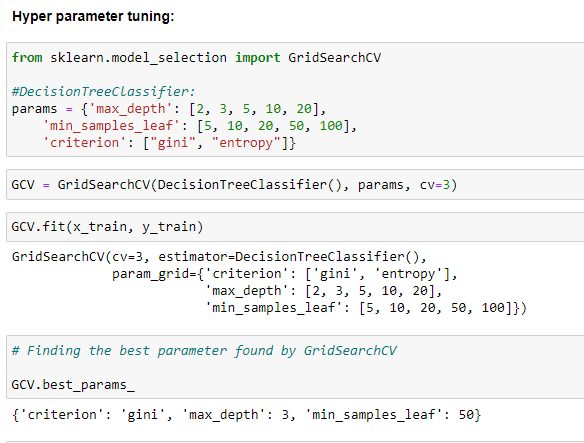
**Hyper parameter tuning:**

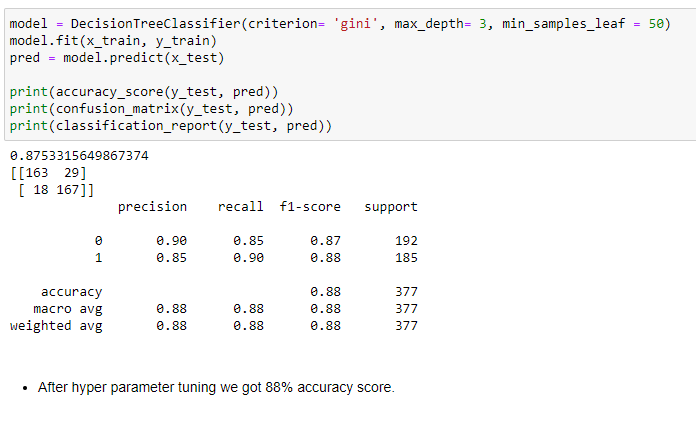
Hyper parameter optimization in machine learning intends to find the hyper parameters of a given machine learning algorithm that deliver the best performance as measured on a validation set. Hyper parameters, in contrast to model parameters, are set by the machine learning engineer before training. The number of trees in a random forest is a hyper parameter while the weights in a neural network are model parameters learned during training. I like to think of hyper parameters as the model settings to be tuned so that the model can optimally solve the machine learning problem.

We will use **GridSearchCV** for the hyper parameter tuning.

**GridSeatchCV:**

In GridSearchCV approach, machine learning model is evaluated for a range of hyper parameter values. This approach is called GridSearchCV, because it searches for best set of hyper parameters from a grid of hyper parameters values.





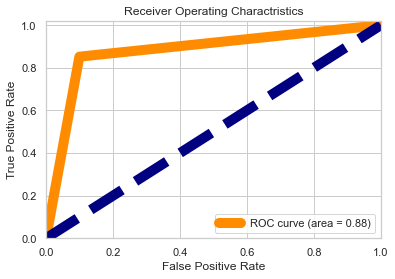
We can see that after hyper parameter tuning our model accuracy score has been increased to approx. 88%.

**AUC ROC Curve**

AUC - ROC curve is 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.

The ROC curve is plotted with TPR against the FPR where TPR is on the y-axis and FPR is on the x-axis.

We are plotting AUC ROC curve for the final model.



**Final Remarks:**

We have successfully built a model to predict the fraud claims made in the auto insurance industry. Our model can help auto insurance companies to increase their profit level by identifying the fraud claims made by the customers. The most challenging job for machine learning model to predict the fraud claims is that the fraud claims are very less common as compared to legit insurance claims.

***References:***

[***https://scikit-learn.org/***](https://scikit-learn.org/)

[***https://towardsdatascience.com/***](https://towardsdatascience.com/)