**Insurance Claim-Fraud Detection**

Submitted By

Shubham Jain

**Table of Contents**

**Introduction**

**Project Flow**

**Conclusion**

**Introduction**

**Problem Statement:**

Business case:

Insurance fraud is a huge problem in the industry. It's difficult to identify fraud claims. Machine Learning is in a unique position to help the Auto Insurance industry with this problem.

In this project, provided with 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.

In this example, you will be working with some auto insurance data to demonstrate how you can create a predictive model that predicts if an insurance claim is fraudulent or not.

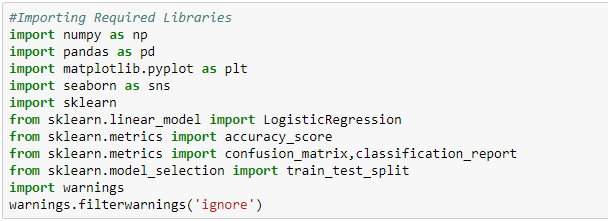
As we can see from the Problem Statement that this is a Classification Problem

Coding Language Used: Python

Environment: Jupyter Notebook

**Project Flow:**

**1. Importing Important Libraries:**



Numpy:

Numpy is core library for Numeric and Scientific Computing

Pandas:

Pandas is core library for Data manipulation and Data Analysis

Matplotlib:

Matplotlib is used for Data Visualisation

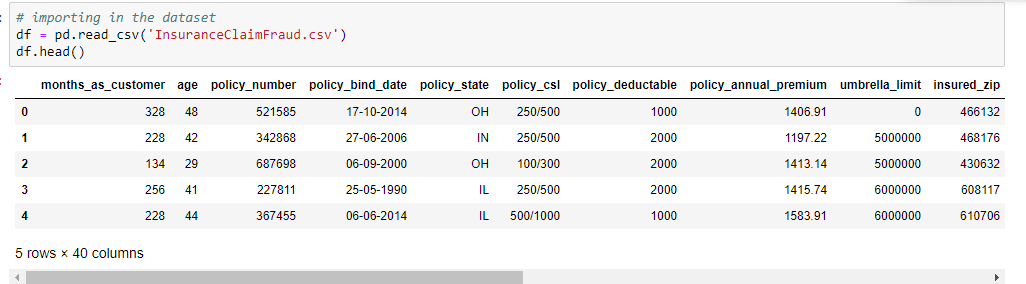
Seaborn:

Seaborn is a Python data visualization library based on matplotlib. It provides a high-level interface for drawing attractive and informative statistical graphics.

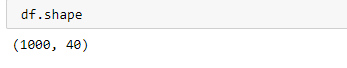
Sklearn:

Scikit-learn (Also known as sklearn) is a [software](https://en.wikipedia.org/wiki/Free_software) [machine learning](https://en.wikipedia.org/wiki/Machine_learning) [library](https://en.wikipedia.org/wiki/Library_(computing)) for the [Python](https://en.wikipedia.org/wiki/Python_(programming_language)) [programming language](https://en.wikipedia.org/wiki/Programming_language). It features various classification, regression and clustering algorithms.

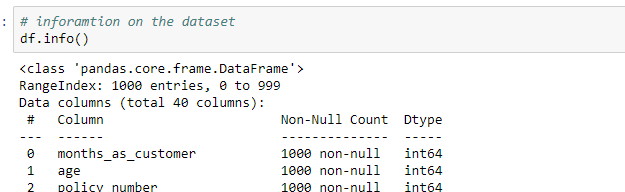
**2. Importing the Dataset** **:**



**3. Performing Basic Operations on data i.e. shape, info**



This gives shape of the dataset



This gives information about the dataset

**4. Checking Correlation**



We see that many of the features have no effect on the target variable so we will remove that.

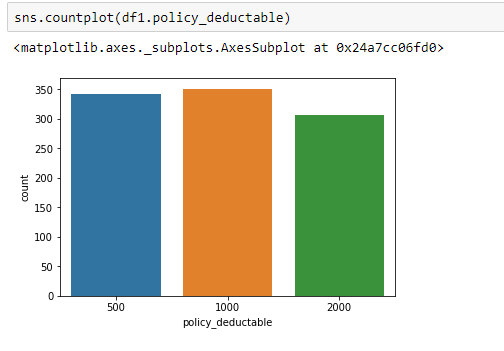
redundant\_cols = ['policy\_number','policy\_csl','policy\_bind\_date','policy\_state','insured\_relationship','insured\_hobbies','incident\_date','authorities\_contacted','incident\_state','incident\_city','incident\_location','incident\_hour\_of\_the\_day','number\_of\_vehicles\_involved','\_c39']



**5. Data Analysis**

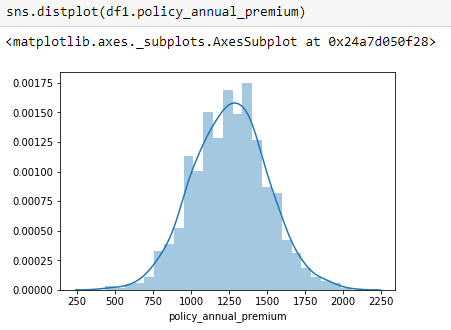
**Univariate Analysis**

**Policy Deductable:**



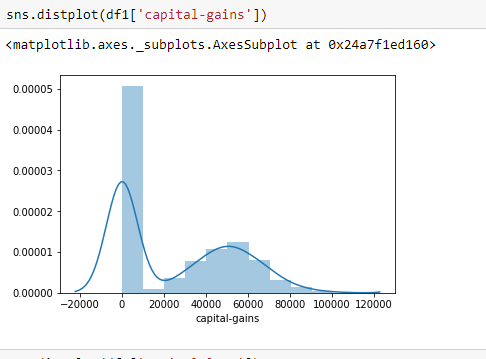
We see that policy deductable is highest for value 1000, then for 500 and then for 2000

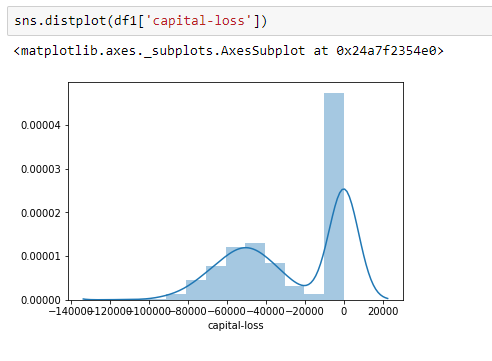
**Annual Policy Premium**



We see that it is highest for Value 1250

**Capital Gains and Capital Loss**

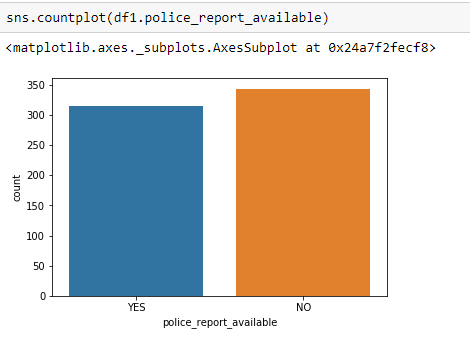




The value of capital gains can be zero as Capital gain is an increase in a capital asset's value. It is considered to be realized when you sell the asset.If you sell the asset at the same price you bought then your capital gains will be zero.

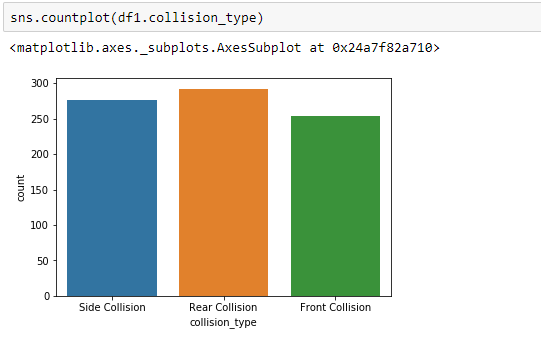
Similar is the capital loss.

**Police Report Available**



We see that Police Report Availability is No for Majority.

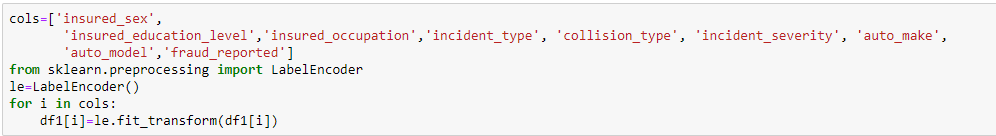
**Collision Type**



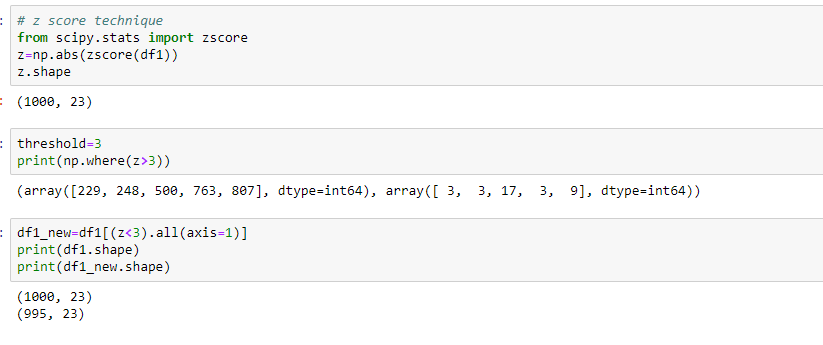
We see that Rear Collision Type is Maximum among Collision Type and then comes Side Collision and then Front Collision

**6. Encoding The Data**

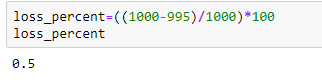
We see that many of the columns are of Object Datatype which needs to be encoded to perform training of the Model



**7. Checking for Outliers**



Used Z Score Technique for Outlier Removal. In this technique z score value greater than 3 will be removed from the dataset but we need to check what is total percentage data loss. If Data Loss is more than 7% , then we will not go for Outlier Removal.



We see that loss of data is only 0.5%. So we will go for Outlier removal using Z score Technique.

**8. Checking Skewness of Data**

We check skewness in data. Skewness limit is +/- 0.65. We see that there is no such skewness present so we will not remove skewness in data



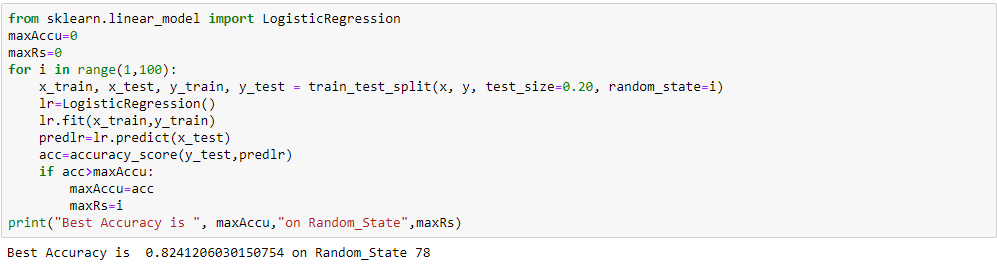
**9. Splitting the Features and Target Variable**



**10. Training the Model**

**Finding Best Random State**

To find best random state, we can use any one algorithm and find the Random\_State for which we get the highest accuracy

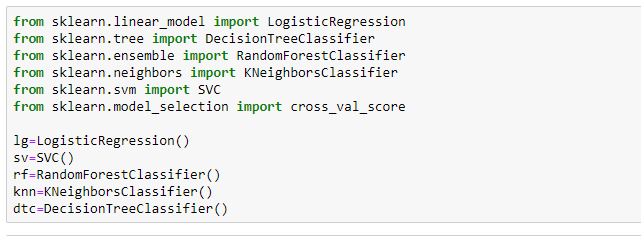


Best Accuracy is Obtained at Random\_State=78, so we will use that random-State to train our model.

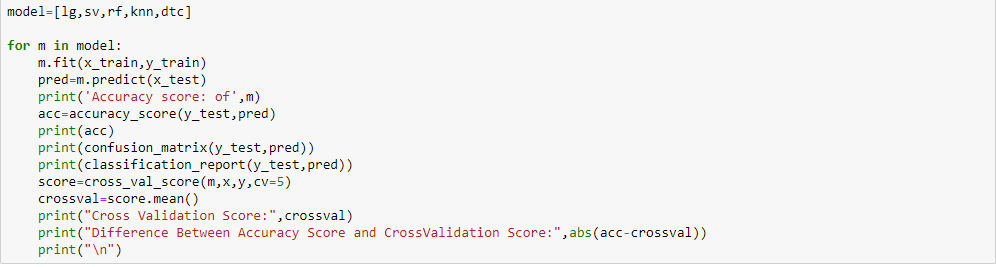
**Train and Test Split**



**Importing all the Models**

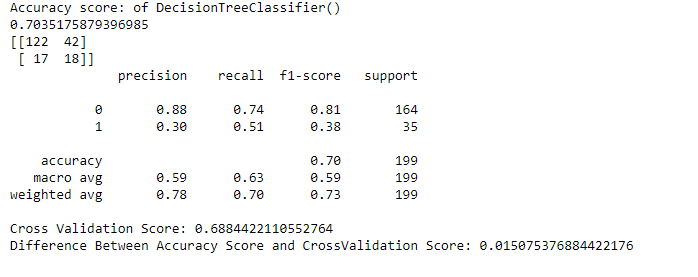


**Training the Model**



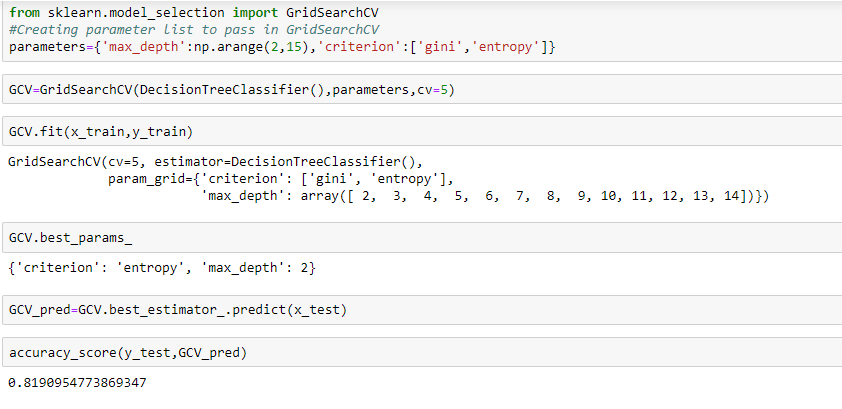
Here we get our Accuracy score, Confusion Matrix, Classification Report and Cross val Score. We will consider the model which is having least difference between Accuracy Score and Cross Validation Score

We see that difference between Accuracy Score and CrossValidation Score is Minimum for Decision Tree Classifier, So DecisionTreeClassifier will be used to train the model.

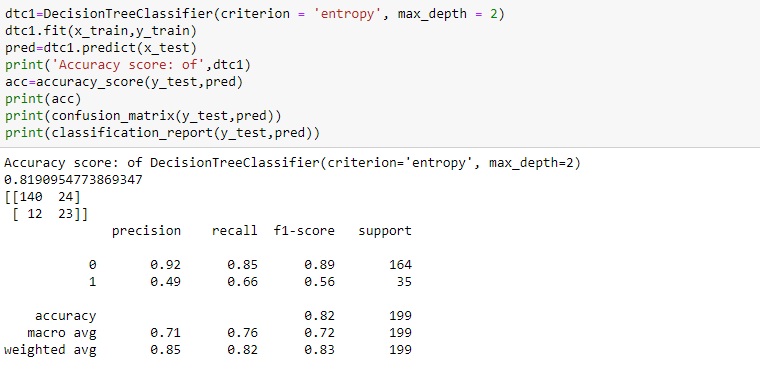


To improve the Accuracy Score of the Model, we have to use HyperParameter Tuning

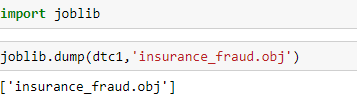
**Hyper Parameter Tuning**



Again training the Model with Best Parameters



**11. Saving The Model**



**Conclusion**

* Of all the models Logistic Regression, SVC, Random Forest Classifier, Knn Classifier and Decision Tree Classifier, the least accuracy was obtained for Decision Tree Classifier, but the minimum difference between Accuracy Score and Cross Validation Score was for Decision Tree Classifier Model.
* So we have used Decision Tree Classifier to Train our Model
* The Accuracy then was improved by using Hyper Parameter Tuning and selecting the best parameters for our model.
* Best Parameters obtained was criterion = 'entropy', max\_depth = 2
* Which when we use to train our model, the accuracy increases of our model to 82%
* Finally, we train our model with the new parameterized Decision Tree Classifier Model and Save the Model.