Wine Quality prediction using Machine Learning



Wine quality, as Maynard Amerine once said, It is easier to detect than define. This is partially due to quality being primarily subjective and strongly influenced by extrinsic factors. Correspondingly, defining wine quality in terms of its chemistry will never be more than partially successful.

According to experts, the wine quality is differentiated according to smell, Flavor, and Color but still, it is very hard to define whether a Wine is of Good Quality or Not and we are also not an expert to say the wine is good or not

**Problem description**: -

Whether a wine is good or bad. Deciding that is one heck of a job and for that, we need experts who can taste the wine tell us whether it is good or not and that makes it very expensive. But we are not experts so we are going to use machine learnings to decide whether a wine is good or not depending on the below mention features and as we know the quality above 7 is considered as a good quality wine and anything below 7 is Bad quality of Wine. This is a classification problem and this project aims to determine which features are the best quality red wine indicators and generate insights into each of these factors to our model’s red wine quality.

**Overview**:

* Basic understanding of Wine.
* Data description
* Importing modules
* Study dataset
* Visualization
* Handle null values
* Split dataset
* Normalization
* Applying model
* Endnote

**Let’s start with Our Machine Learning projects. So if you download the dataset you have come through the below mention features**

* volatile acidity: Volatile acidity *is the* gaseous acids present in wine.
* fixed acidity: Primarily fixed acids found in wine are tartaric, succinic, citric, and malic
* residual sugar: Amount of sugar left after fermentation.
* citric acid: It is a weak organic acid, found in citrus fruits naturally.
* chlorides: Amount of salt present in wine.
* free sulfur dioxide: So2 is used for the prevention of wine by oxidation and microbial spoilage.
* total sulfur dioxide
* pH: In wine, pH is used for checking acidity
* density
* sulfates: Added sulfites preserve freshness and protect wine from oxidation, and bacteria.
* alcohol: Percent of alcohol present in wine.
* quality (score between 0 and 10)

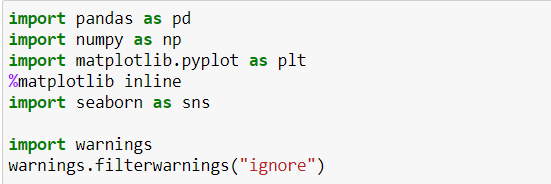
**Data Analysis**: -

In this dataset, there are two categories: A score of 7 and beyond is considered as a good quality wine and below that is considered bad quality.

It is very hard to define whether a Wine is of Good Quality or Not and we are also not an expert to say the wine is good or not. What do we do then? Here we use machine learning to decide whether a wine is good or not

**Let’s start with our Machine Learnings project**

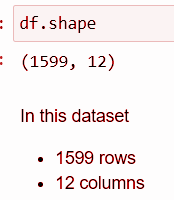
First of all, we need to import some basic libraries:



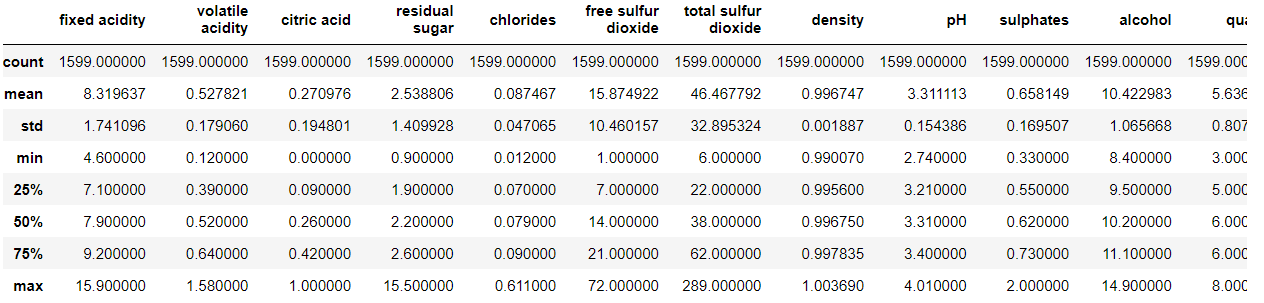
You must be familiar with Pandas, NumPy, seaborn, and matplotlib. Here we use seaborn and matplotlib for data visualization.

Now we import data from a repository by providing the link in ‘**pd.read\_csv()’**  Now we need to analyze the data for a deep understanding of the features which are important and responsible for the good taste of wine

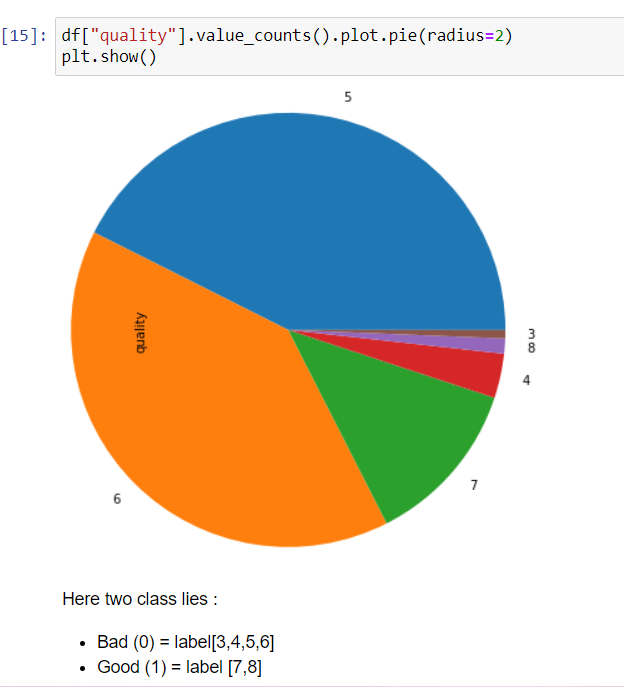
We are going to check the shape of the dataset, Null values, statistics, and basic info about the dataset to get to know about mean and deviation from the mean to get to know about the skewness in the dataset.



**Checking for basic statistics in Tabular Form:**



From the above basic statistics, we can say that there is right skewness in the data because if we look at mean and standard deviation few features are having very little difference between standard deviation and mean therefore, we can see that there is skewness in the data.

* In Our dataset, we have several scores in Quality columns and we have labeled Quality scores below 7 as Bad and score 7 and above as good

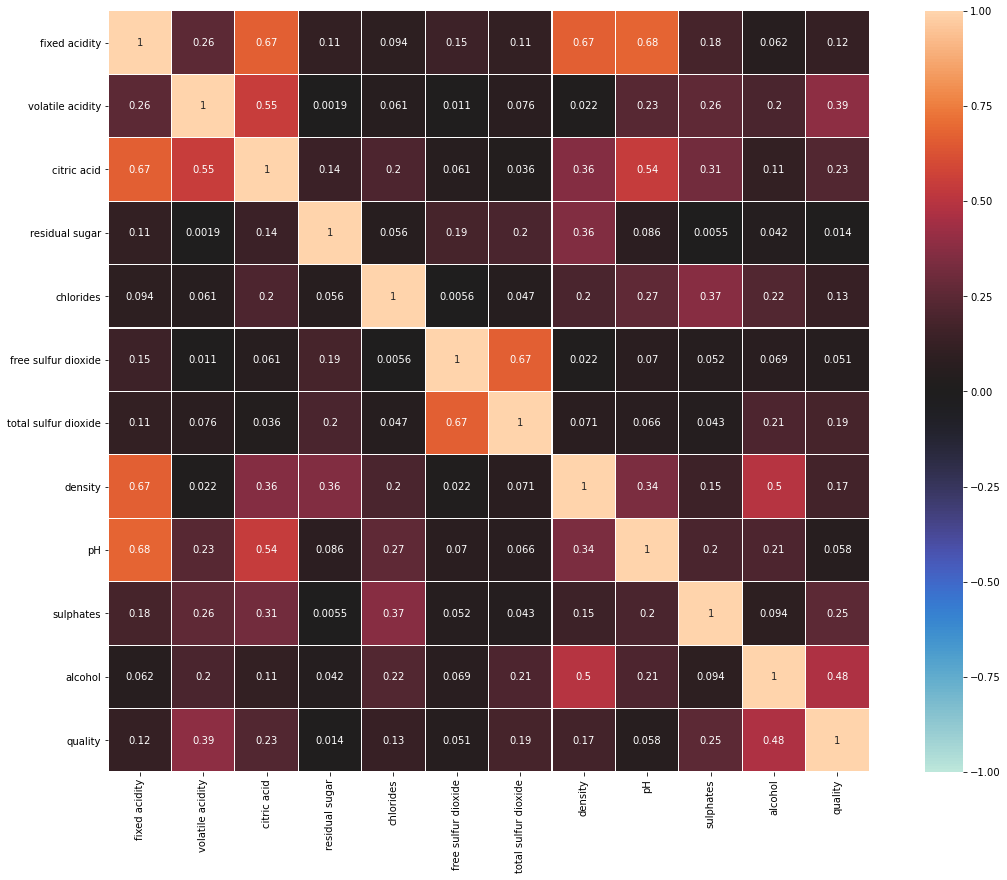
So, we are going to convert this problem into a classification problem by replacing 3,4,5,6 with bad and 7,8 with Good quality.

**EDA :-**

There is some kind of skewness in the data in columns: fixed acidity, volatile acidity, citric acidity, residual sugar, chlorides, free sulfur dioxide, total\_sulphur\_dioxide, density, and chloride had the maximum skewness. And we can see

But before plotting the Heatmap we need to encode all the data so we know the machine did not understand the string. So I am using Label Encoder to Encode the categorical data

To see which features are correlated to the target Variable I plot a Heatmap to show the correlation of each variable with the Target Variable.



In order of highest correlation, these variables are:

1. Alcohol: the amount of alcohol in wine should be balanced because people prefer Wine with balanced alcohol content.

2. Volatile acidity: are high acetic acid in wine which leads to an unpleasant vinegar taste

3. Sulphates: a wine additive that contributes to SO2 levels and acts as an antimicrobial and antioxidant

4. Citric Acid: acts as a preservative to increase acidity (small quantities add freshness and flavor to wines)

5. Total Sulfur Dioxide: is the amount of free + bound forms of SO2

6. Density: sweeter wines have a higher density

7. Chlorides: the amount of salt in the wine

8. Fixed acidity: These are non-volatile acids that do not evaporate readily

9. pH: the level of acidity

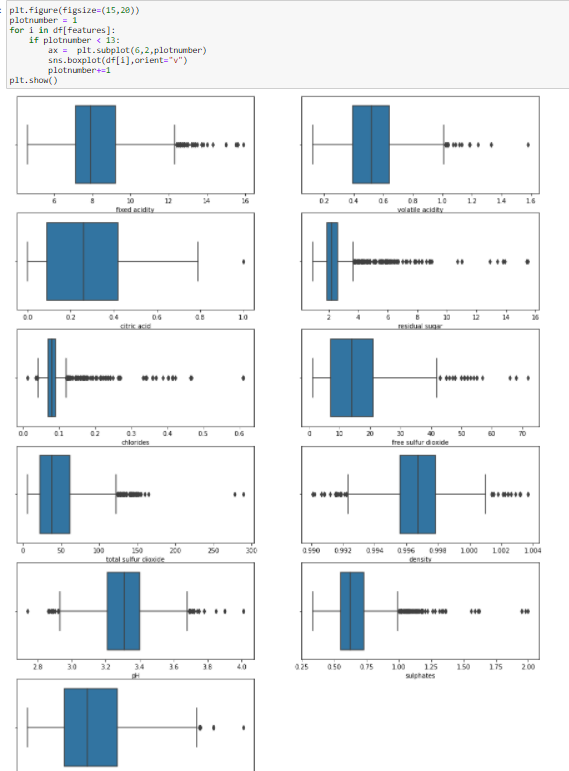
10. Free Sulfur Dioxide: it prevents microbial growth and the oxidation of wine

11. Residual sugar: is the amount of sugar remaining after fermentation stops. The key is to have a perfect balance between — sweetness and sourness (wines > 45g/ltrs are sweet)

**EDA Conclusion**: - All the features are somehow related to the Quality of wine so taking any particular features to decide whether the Wine is good or not is not good for our model.

**Data Cleanings**:

As we know that there is skewness in the data may be because it might be containing some Outliers in the data. So let’s check the Outliers in the dataset

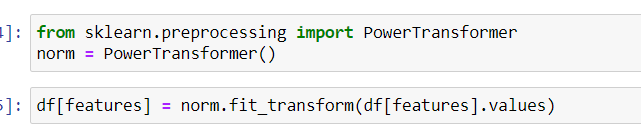


There are Outliers in each Independent variable and we need to take care of that to remove the skewness from the dataset. There are so many methods available to take care of the Outliers such as Zscore, Interquartile and quantile method

A Z-score is a numerical measurement that describes a value's relationship to the mean of a group of values. Z-score is measured in terms of standard deviations from the mean and applying that method here to remove Outliers won’t affect much because the Outliers also lies between z score range. If we adjust the threshold value of the z score the data loss will be great

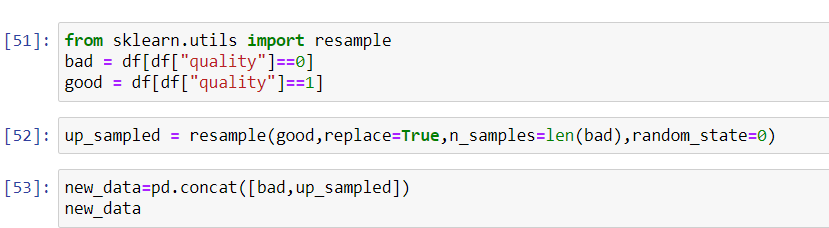
Removing Outliers using interquartile and quantile method tend to lose More data than expected So keeping Outliers in the dataset is a good option here because sometimes we need to think rationally about that. Is it really essential to remove Outliers? Removing skewness is a completely different story because having skewness in the dataset is bad for some models. So we need to take care of that using any Data Transformation method i.e. log transformation, Square root, Cube Root, and Power Transformation. Completely depend upon the dataset. Log transformation is considered to be the primary Transformation technique but we are having zero so applying log transformation is not a good option.

Here we are using the Power Transformation technique to remove the skewness in our data and passing only that feature which is having skewness beyond +5 and -5.



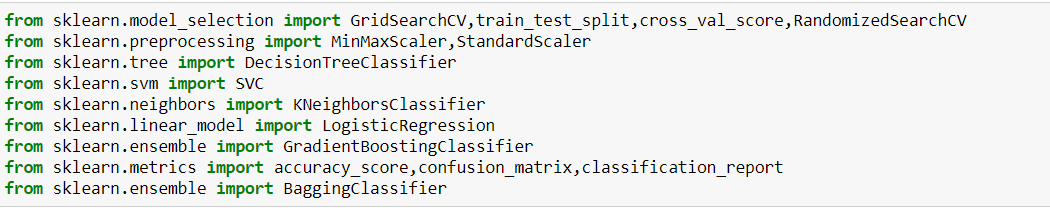
**Balancing the dataset**:

As we know there is a class imbalance in the Target variable. And we need to balance it because getting only a few numbers samples during training and testing our machine won’t get a good idea about that category and it is also possible that if the dataset is too small that all of the categories goes to the training there is nothing left for testing so we are going to use Oversampling/ Upsampling of the dataset to make balanced of both the category in the dataset.

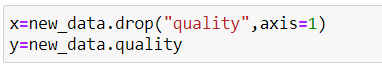


**Modeling**:-

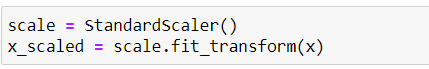
Importing some important scikit libraries for modeling, splitting, and standardization of the dataset :



Before modeling, we need to separate dependent and Independent variable



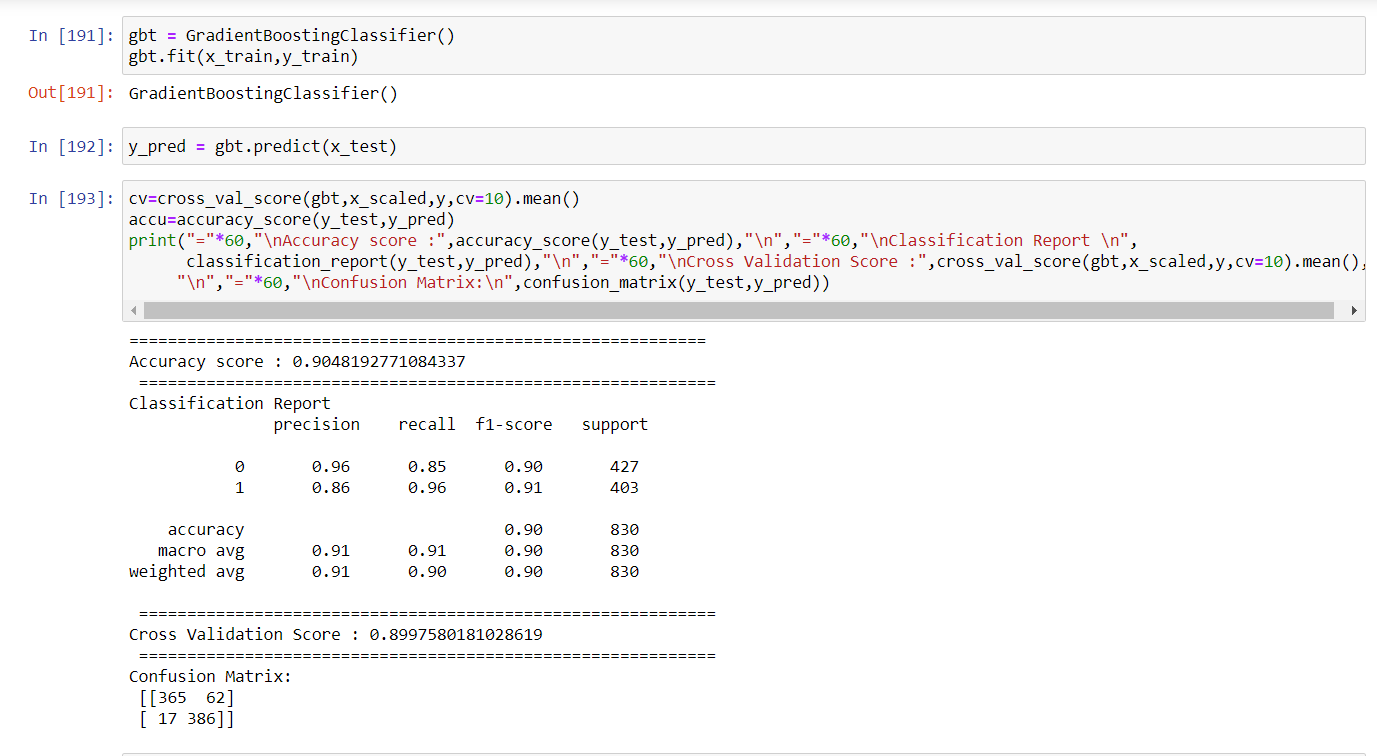
1. Standardization using StandardScaler:- StandardScaler removes the mean and scales each feature/variable to unit variance and for more information click [here](https://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.StandardScaler.html?highlight=standardscaler#sklearn.preprocessing.StandardScaler)



1. Train test split:- It is a method to split the data into four-part i.e (x\_train,x\_test,y\_train,y\_test )2 for training and 2 for testing



1. I have built several models i.e Logistic Regression, GradientBoosting, KNeighbors, Bagging, and lastly DecisionTree.

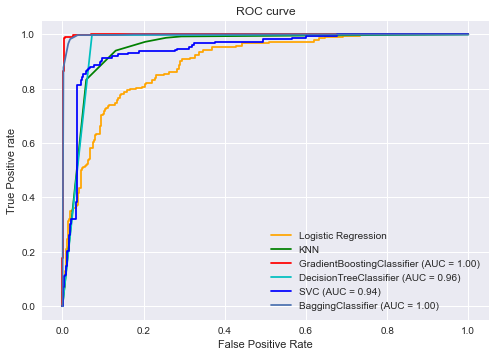


First I instantiate the model in an object then passing the training data to the model using the .fit method.

When the model learns and Understands the pattern of the data it is ready to predicting so I have passed the testing data and stored that data in y\_pred and compare that data with the actual testing data for that I use accuracy score and cross-validation to check the model accuracy.

But we can’t depend on one model only so we make multiple models and check their accuracy and cross-validation score and their difference to find the best model. The best model has the least difference between the Accuracy and Cross-validation-score. Here as you can see in my case it is

**GradientBoosting is the Best Model** And we trying to increase the Accuracy of that model using Hyperparameter Tunning and then we are going to plot the Auc\_roc\_curve to check how much accuracy and region it covers



**Final Remarks: -**

In general, GradientBoostingClassifier as our best model for prediction, I determined four of the features as the **most influential: volatile acidity, citric acid, sulphates, and alcohol**. To be more specific, high-quality wines seem to have lower volatile acidity, higher alcohol, and medium-high sulphate values. Meanwhile, lower-quality wines tend to have low values for citric acid.

This model accuracy be further increased if we include more relevant data features, like the year of harvest, brew time, location, or wine type.

**Flight ticket Price prediction Using Machine Learnings :-**



**Problem Definition**: -

The objective of this article is to predict flight prices given the various parameters. Flight ticket prices can be something hard to guess, today we might see a price, check out the price of the same flight tomorrow, it will be a different story. We might have often heard travelers saying that flight ticket prices are so unpredictable. As data scientists, we are gone prove that given the right data anything can be predicted. Here you will be provided with prices of flight tickets for various airlines between the months of March and June of 2019 and between various cities. This will be a regression problem since the target or dependent variable is the price

**FEATURES**:

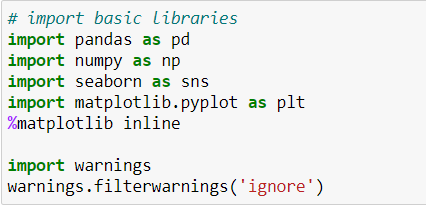
1. Airline: The name of the airline.
2. Date\_of\_Journey: The date of the journey
3. Source: The source from which the service begins.
4. Destination: The destination where the service ends.
5. Route: The route was taken by the flight to reach the destination.
6. Dep\_Time: The time when the journey starts from the source.
7. Arrival\_Time: Time of arrival at the destination.
8. Duration: Total duration of the flight.
9. Total\_Stops: Total stops between the source and destination.
10. Additional\_Info: Additional information about the flight
11. Price: The price of the ticket

**Introduction**

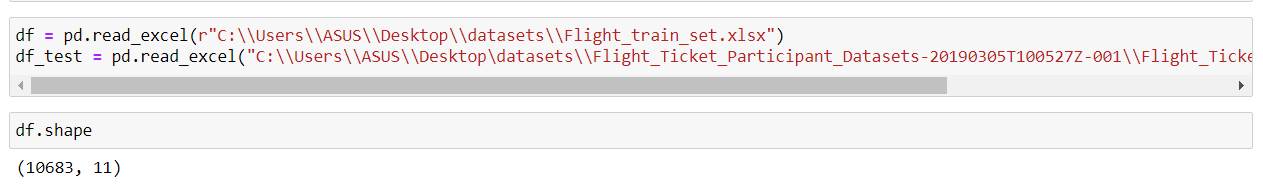
Airline companies use complex algorithms to calculate flight prices depend on various factors like market, fuel prices, Social factors. In previous years the people taking flights significantly and the prices of the flight ticket changes dynamically

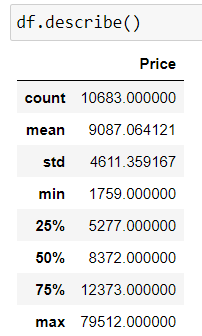
**Data Analysis: -**

Starting extracting information from the raw data. First, we need to import basic library

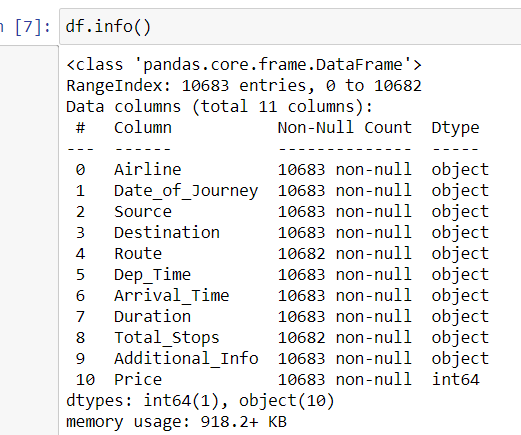


Here we have two datasets one for training and one for testing so we import both the dataset using “**pd.read\_csv()**”

In this dataset, we have only one continuous column i.e price.



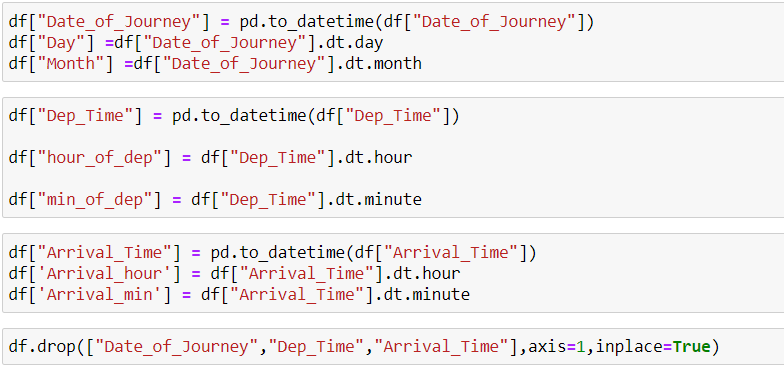
Checking the basic info about the dataset: -

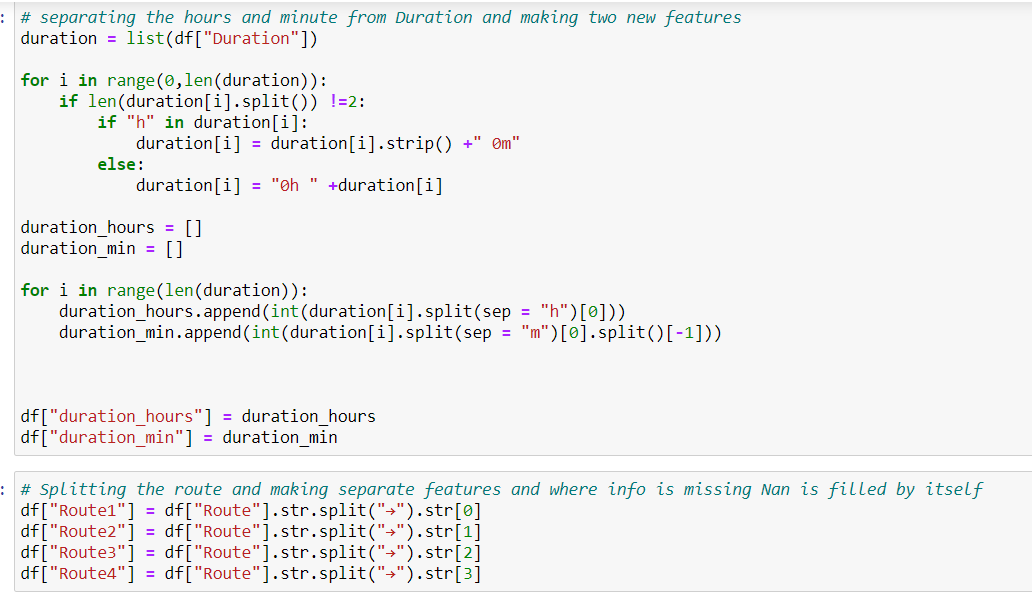


In this dataset, there are some missing values in Route and Total\_stops and dropping them is the best option we had.

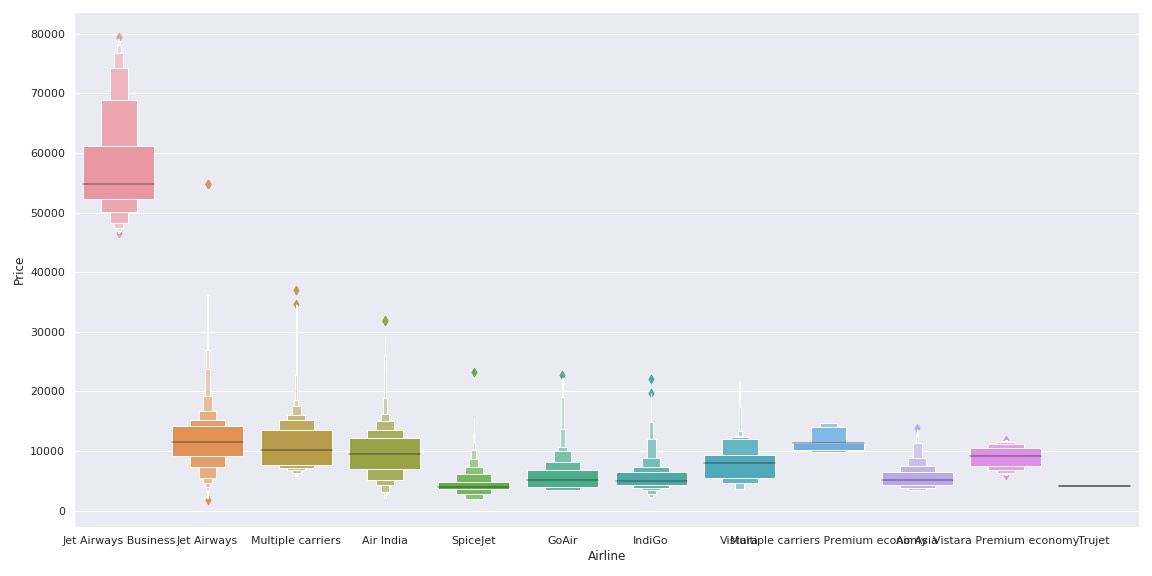
**Data cleanings: -**

First of all, we need to take care of the date column because the system does not Understand data and time in DateTime format. So need to use feature engineering to create new features from date of journey, Dep\_time, Route, Arrival\_time and Duration





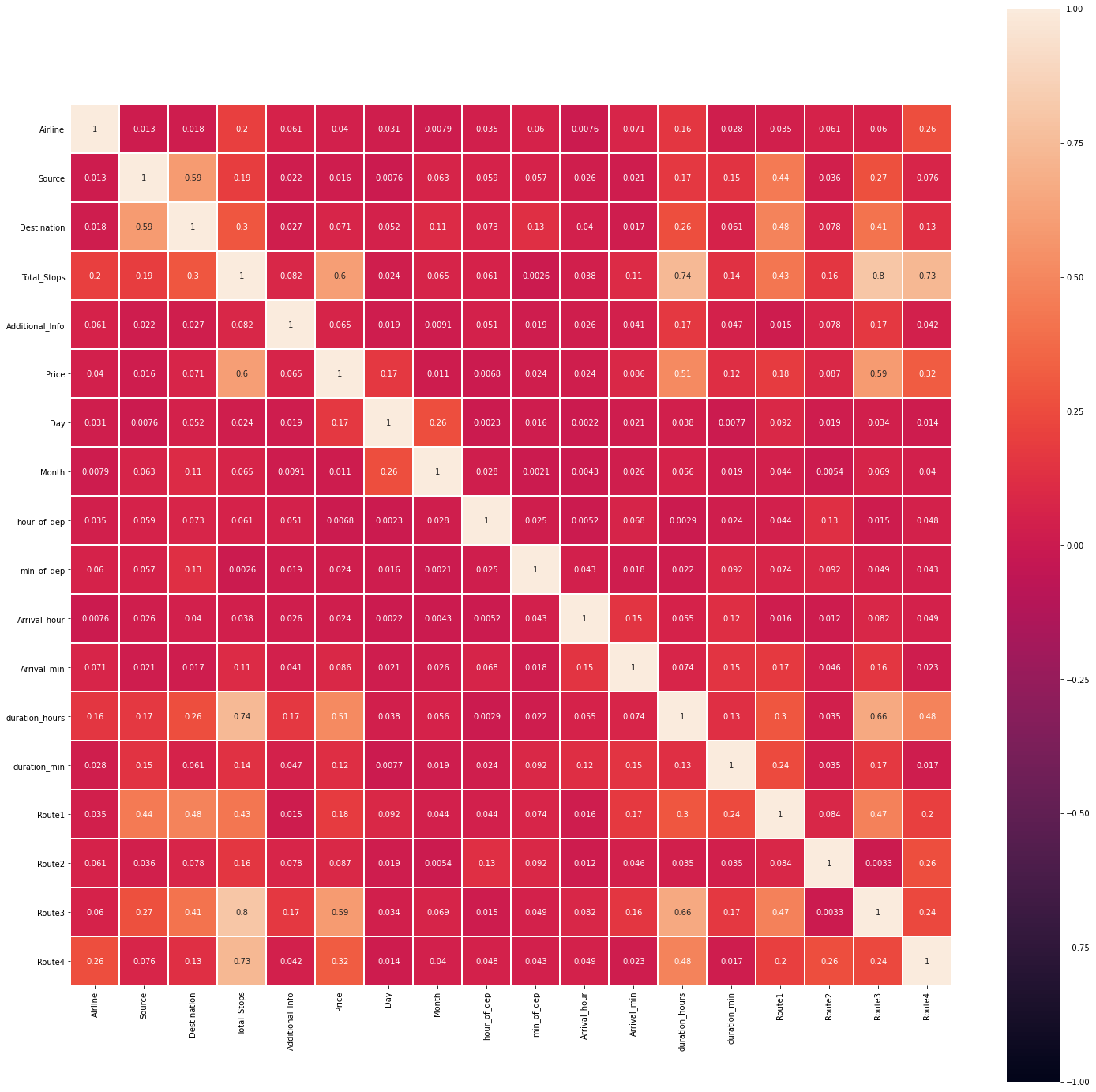
Now checking the fare of each airline’s companies



As we can see the name of the airline matters. ‘Jet Airways Business’ has the highest price range. Other airlines price also varies.

Dropping all the columns from where we have already extracted the required information using Feature Engineering and then Encode the remaining categorical columns using Ordinal Encoder

**Heatmap:-** Plotting heatmap to see the correlation of each feature with the target variable by plotting heatmap



It seems that the features have multicollinearity issue need to take if that using feature selection technique.

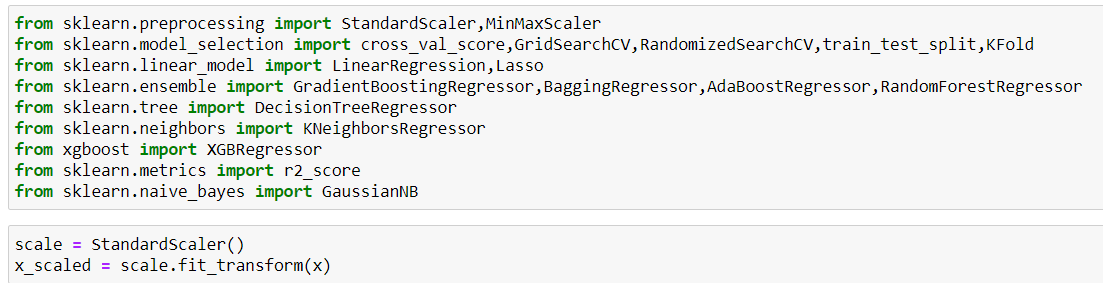
Checking and removing Outliers is not necessary for this dataset as the features are either categorical or Target variables and we do not remove outliers and skewness from categorical and Target variables.

**Feature Selection using Select Percentile:**

It is one of the feature selection techniques used to select the important features for predicting our target variable and before applying that technique we need to split the data into dependent and independent variable

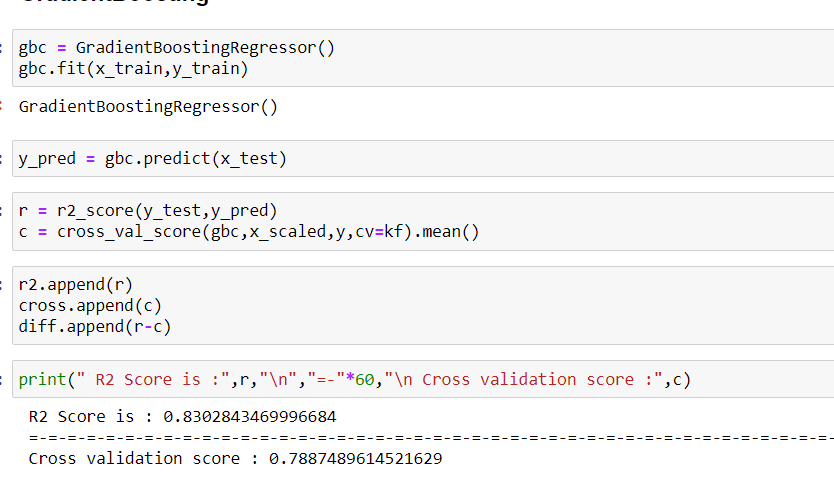
**Modeling: -**

Before modeling, we need to pass the selected features to x and then scaled them using StandardScaler.



**Steps to Build a Model:-**

1. First, instantiate model in an object
2. Then pass the training data into the model using the .fit method
3. Then use that model to predict testing data and check the Accuracy using the respective metrics. If it is regression then r2\_score. MAE, MSE, RMSE, and cross-validation score we can use.



But we can’t depend on one model so build multiple models and check the accuracy and decide which model is best for the given dataset and after that use the Hyperparameter Tunning to increase the accuracy score.

**Final Remark:**

As GradientBoosting is our Best Model. But I notice that Date of Journey, Dep\_time, Duration, and Total\_stop is the most influential features because the price of the flight ticket increases on the weekends and holidays, and total\_stop and duration it takes to reach the destination. Maybe fuel consumption is more as the duration and total\_stop increases and on holiday and festival the demand of the ticket is increases.