Name : Subodh Kumar

Enrollnment No. : 20116095

Branch : ECE, 1st year

Team : 27sub02

Link to jupyter notebook with the code :

https://github.com/subu-2702/maclearn/blob/main/Music%20Genre.ipynb

Link to the github repo having the jupyter notebook with the code :

https://github.com/subu-2702/maclearn

Documentation of the solution to the problem statement of the Beginners_hypothesis_2021

In this project I have designed a machine learning model to predict the genre of music given as the input to the model based on the features associated to it some of which are : energy, loudness, age , instrumentalness , music etc.

It is a supervised classification machine learning problem with 7 different labels possible as the genre of the music taken as input.

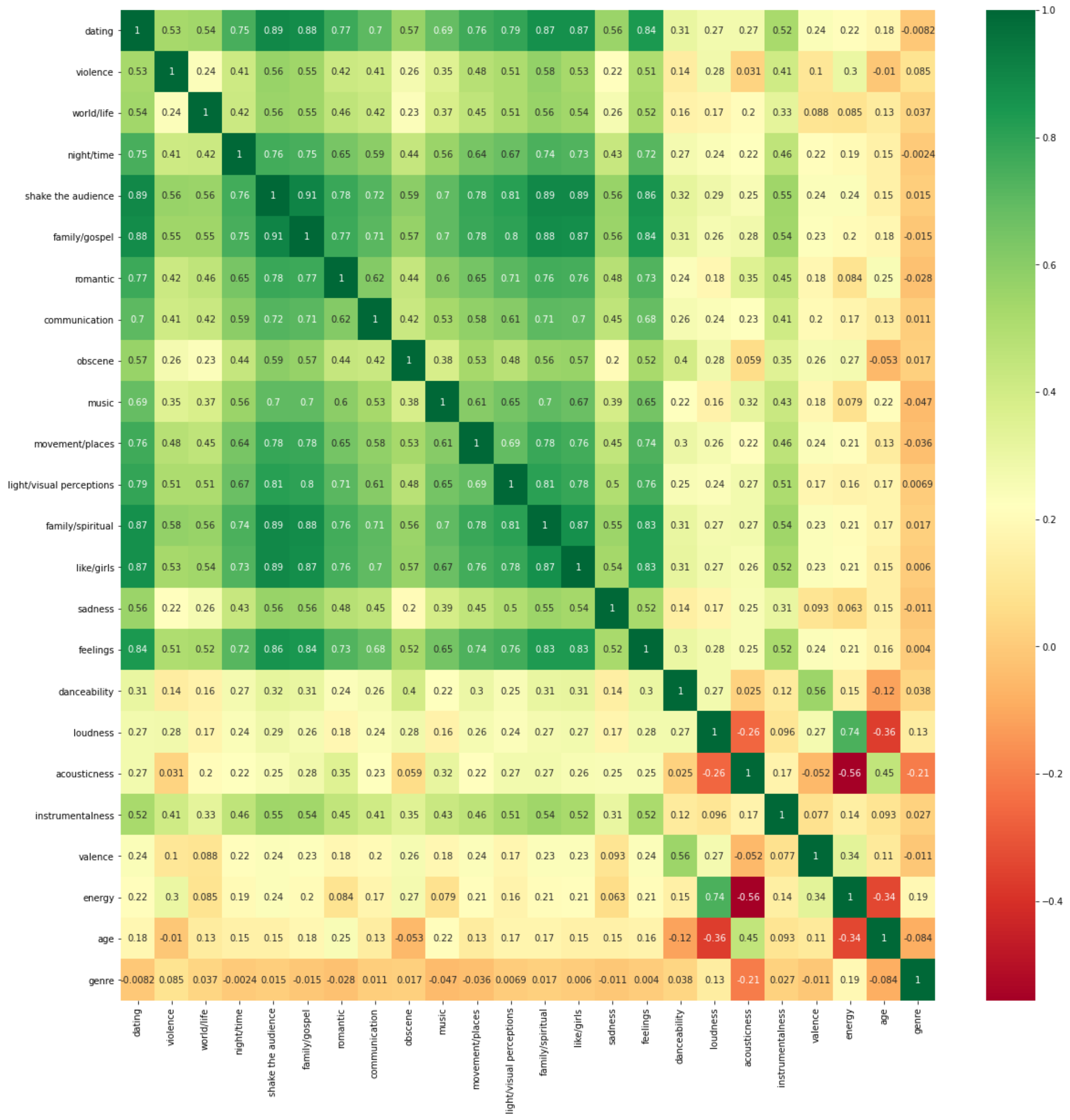
Libraries used : scikit learn, matplotlib, seaborn , pandas and numpy.

First of all I read the 'music_train.csv' file using pd.read_csv('music_train.csv') which converts it to a dataframe.

From this dataframe I dropped the 'id' and the 'release_date' columns because I id was a serial number not a feature of the song .And also release date don't have any impact on the genre of the music too.

Data visualization

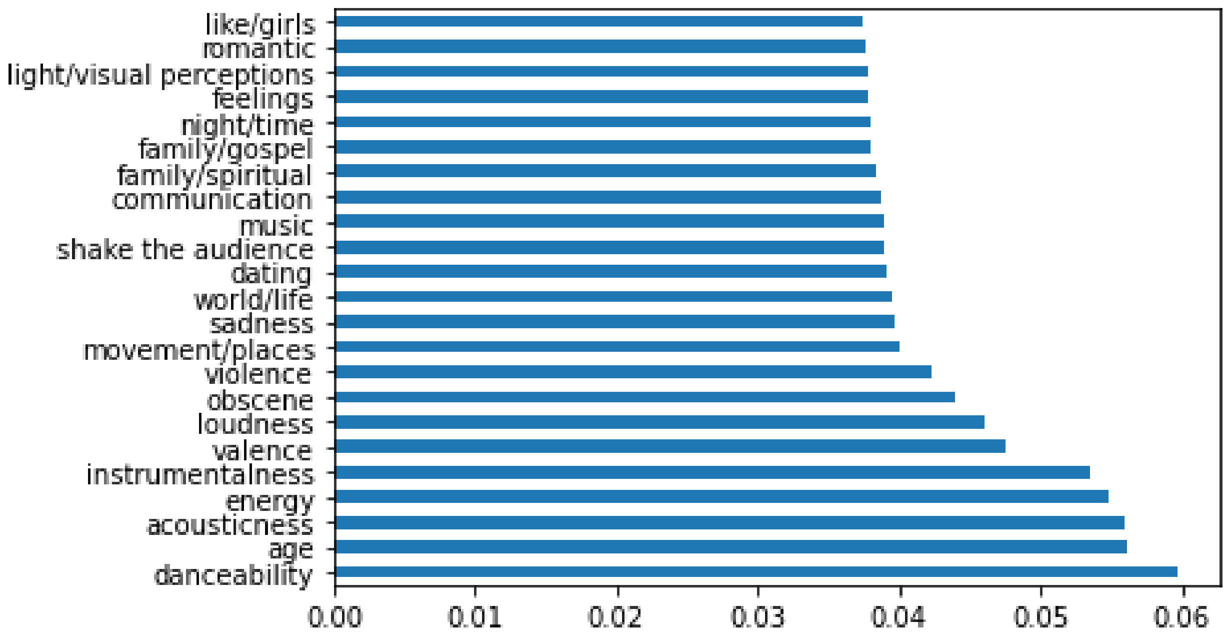
I have used the seaborn heatmap to visualize the correlation of all the features with the target variable “genre”.



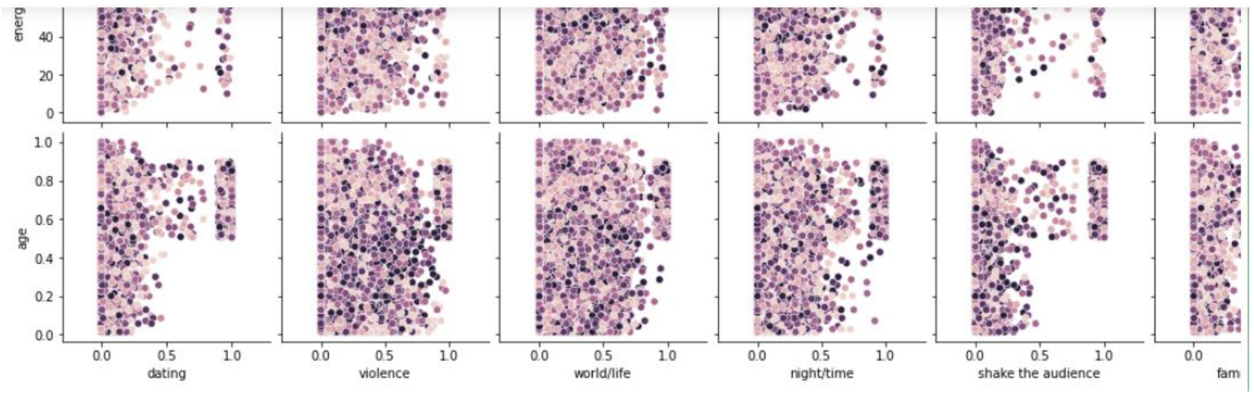
Then I have plotted a bargraph showing the importance of all the attributes

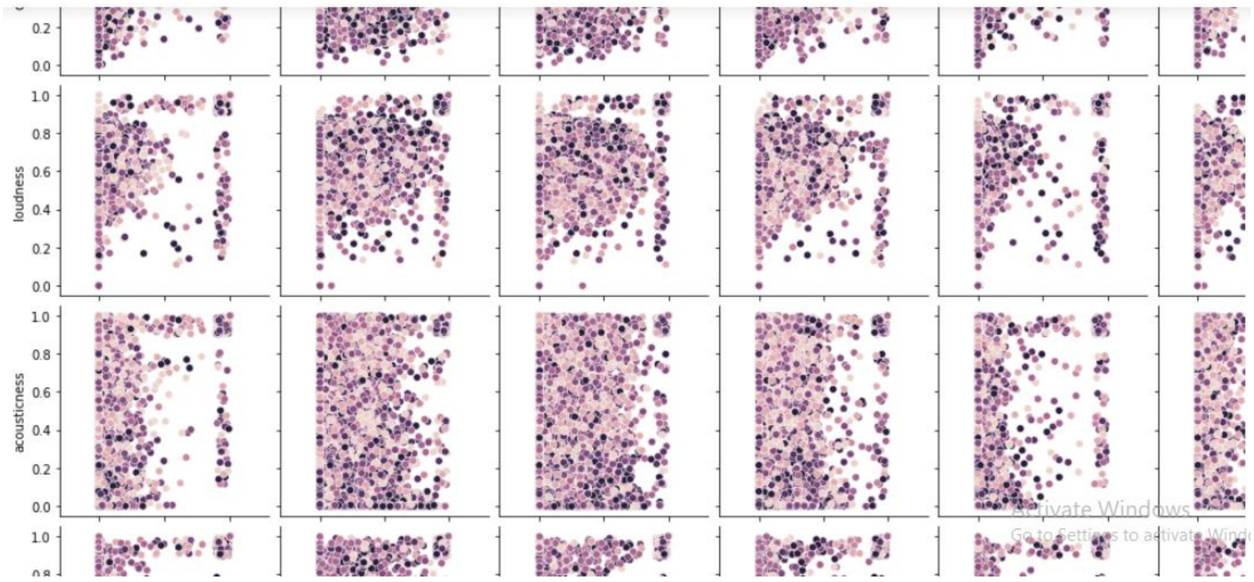
Towards the genre of the music:

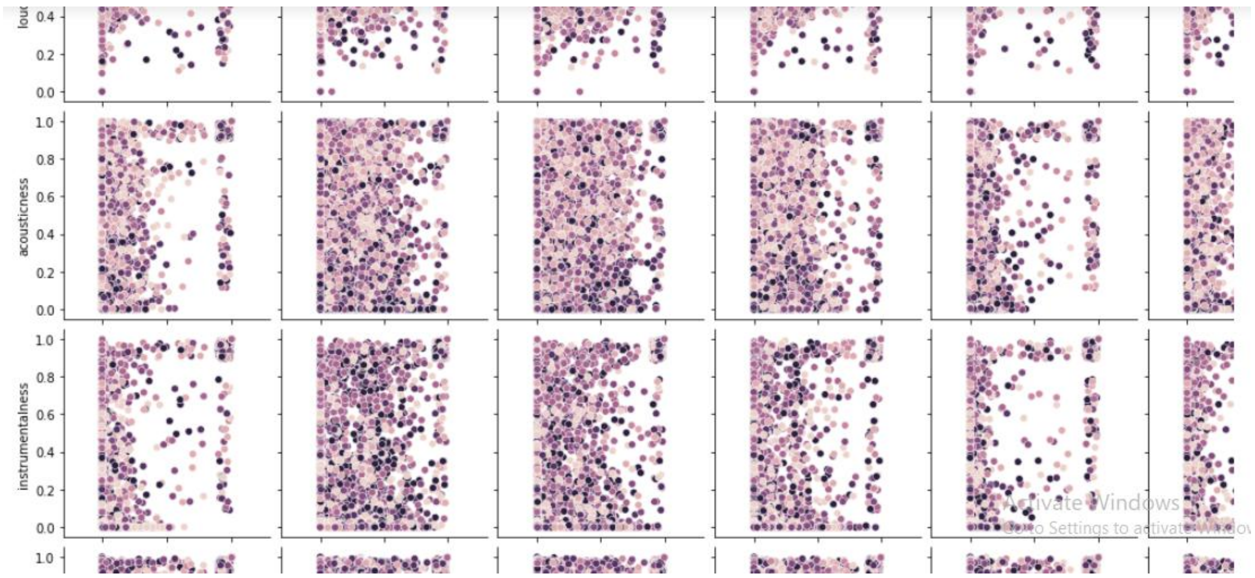
From these bargraph I inferred that all the features have considerable correlation to the genre so I used all the remaining columns as the feature for my model.

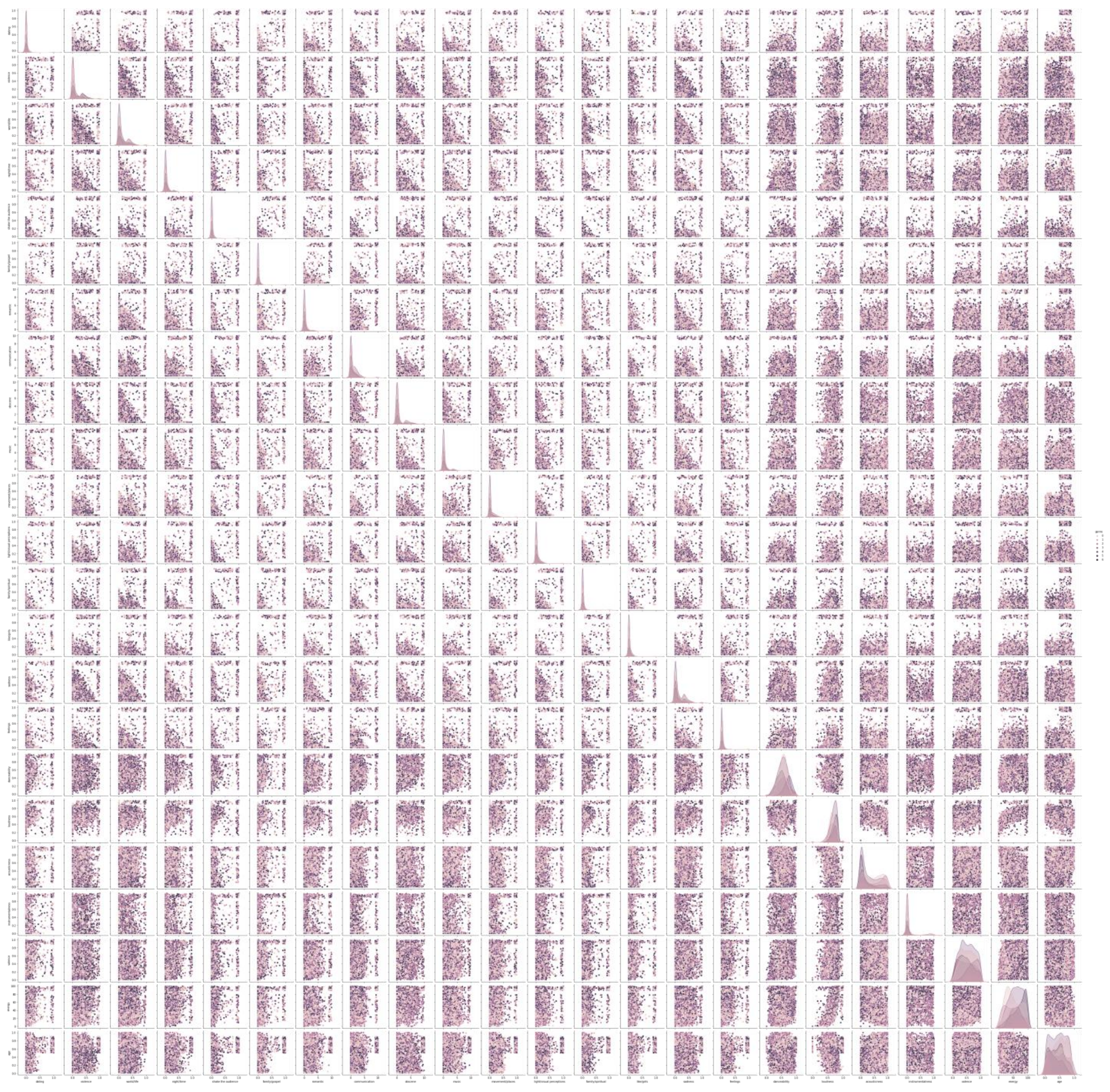


I used the pairplot method of the seaborn library to visualize the pair wise relation of each feature with all the possible 7 labels at the same time.




0.2
0.0
1.0
0.8
0.6
0.4
0.2
0.0
loudness
1.0
0.8
0.6
0.4
0.2
0.0
acousticness
1.0
0.8
Activate Windows
Go to Settings to activate Windo


lou
0.4
0.2
0.0
1.0
0.8
0.6
0.4
0.2
0.0
acousticness
1.0
0.8
0.6
0.4
0.2
0.0
instrumentalness
1.0
Activate Windows
Go to Settings to activate Windo



<u>**Observation from the above pairplots :**</u>

I have noticed that the distribution of the lables with respect to the different features are very randomly distributed as shown in the graph so there is clearly no clear cut boundary between the dispersion of the different labels hence , I inferred that the logistic regression classification algorithm will not work properly in these case.

Also, the concentration of the points of different labels are very very close and very highly concentrated so the KNN classifier classification algorithm should also not work properly because the number of the nearest neighbour points to a input point will be very high and very close in numbers so the predicted label might not be correct.

***Finally I have used the RandomForestClassifier model for these problem as the data was** very much randomized.*

And the RandomForestClassifier have a no. of decision trees within it that trains themselves on the partial rows and partial columns taken randomly and the final prediction is done after taking the majority of outputs given by these different decision trees separately so I thought it will be a better algorithm to train the given data . I tried a few of the n_estimator numbers to check the accuracy of the model I got the highest at n_estimator=100.

Spilliting the given training dataset :

I spllited the given training dataset in 80:20 ratio for train:test of my model.

After spilliting the given music_train dataset to get the training and testing dataset I sliced them to get the features in one dataframe and the label i.e. genre to other dataframe so that it will passed separately to the model for training, testing and making predictions and calculating the accuracy of the model.

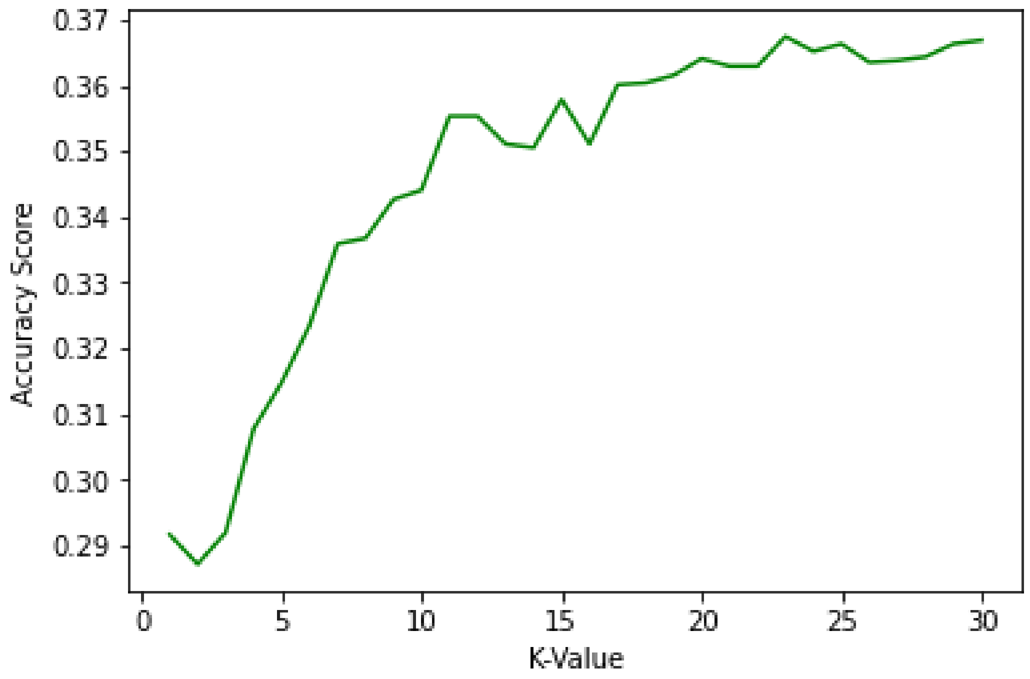
Dealing with missing values of the given dataset

I have used Simpleimputer inorder to replace the missing values of the given data set with the median of the values of all the entries of the respective columns.

Since, almost all the columns had missing values and also the upcoming test dataset for the model can have missing values **I used imputer and also introduced it to my pipeline** so that any data coming either for training or testing will through the pipeline and all its missing values will be replaced with the median of its respective column.

I have also tried KNN classifier model however its accuracy was not at par with the RandomforestClassifier as I have expected and described above in this documentation.

I tried it for n_neighbours = 23 because it was showing in the graph plotted between accuracy and k-value that the accuracy will be highest at this k-value.



<u>Statics obtained for RandomForestClassifier model for my test dataset:</u>

KNN classifier (K=23)

Scores: [2.41197132 2.46696483 2.42884659 2.42710632 2.48675136 2.49693258

 2.40052261 2.41661503 2.50397851 2.45480645]

```
Mean:  2.44944956028758

Standard deviation:  0.03571445593902268


Accuracy : 0.36742744435052127


Classification Report :

           precision    recall  f1-score   support


       0      0.33      0.17      0.22       604

       1      0.34      0.60      0.43       657

       2      0.39      0.35      0.37       120

       3      0.48      0.31      0.37       454

       4      0.36      0.41      0.38       890

       5      0.45      0.39      0.42       327

       6      0.35      0.27      0.31       497


    accuracy                          0.37      3549

   macro avg      0.39      0.36      0.36      3549

weighted avg      0.37      0.37      0.36      3549
```

And these accuracy is less than that the model was giving with the RandomForestClassifier model

Statics obtained for RandomForestClassifier model for my test dataset:

RandomForestClassifier(100)

Scores: [2.26407958 2.37400832 2.38348189 2.39159326 2.38550355 2.36949726

2.34212579 2.36025942 2.38446936 2.34107244]

Mean: 2.35960908646785

Standard deviation: 0.03600298807363526

Accuracy : 0.40180332488024795

Classification Report :

	precision	recall	f1-score	support
0	0.33	0.19	0.24	604
1	0.39	0.57	0.47	657
2	0.48	0.37	0.42	120

```
           3      0.50      0.37      0.42       454

           4      0.38      0.49      0.43       890

           5      0.47      0.41      0.44       327

           6      0.41      0.31      0.35       497


    accuracy                          0.40      3549

   macro avg      0.42      0.39      0.40      3549

weighted avg      0.40      0.40      0.39      3549
```