Introduction:

1. Decision Tree

Decision tree is the most powerful and popular tool for classification and prediction. A Decision tree is a flowchart like tree structure, where each internal node denotes a test on an attribute, each branch represents an outcome of the test, and each leaf node (terminal node) holds a class label.

Strengths and Weakness of Decision Tree approach

The strengths of decision tree

* + Decision trees are able to generate understandable rules.
  + Decision trees perform classification without requiring much computation.
  + Decision trees are able to handle both continuous and categorical variables.
  + Decision trees provide a clear indication of which fields are most important for prediction or classification.

The weaknesses of decision tree

* + Decision trees are less appropriate for estimation tasks where the goal is to predict the value of a continuous attribute.
  + Decision trees are prone to errors in classification problems with many class and relatively small number of training examples.
  + Decision tree can be computationally expensive to train. The process of growing a decision tree is computationally expensive. At each node, each candidate splitting field must be sorted before its best split can be found. In some algorithms, combinations of fields are used and a search must be made for optimal combining weights. Pruning algorithms can also be expensive since many candidate sub-trees must be formed and compared.

1. Random Forest

Random forests or random decision forests are an ensemble learning method for classification, regression and other tasks that operates by constructing a multitude of decision trees at training time and outputting the class that is the mode of the classes (classification) or mean prediction (regression) of the individual trees.

The advantages of random forest are:

|  |  |
| --- | --- |
|  | It is one of the most accurate learning algorithms available. For many data sets, it produces a highly accurate classifier. |
|  | It runs efficiently on large databases. |
|  | It can handle thousands of input variables without variable deletion. |
|  | It gives estimates of what variables are important in the classification. |
|  | It generates an internal unbiased estimate of the generalization error as the forest building progresses. |
|  | It has an effective method for estimating missing data and maintains accuracy when a large proportion of the data are missing. |

The disadvantages of random forest are:

 Random forests have been observed to overfit for some datasets with noisy classification/regression tasks.

 For data including categorical variables with different number of levels, random forests are biased in favor of those attributes with more levels. Therefore, the variable importance scores from random forest are not reliable for this type of data.

3. Multiple Linear Regression

Multiple linear regression (MLR), also known simply as multiple regression, is a statistical technique that uses several explanatory variables to predict the outcome of a response variable. The goal of multiple linear regressions (MLR) is to model the linear relationship between the explanatory (independent) variables and response (dependent) variable.

Data Set:

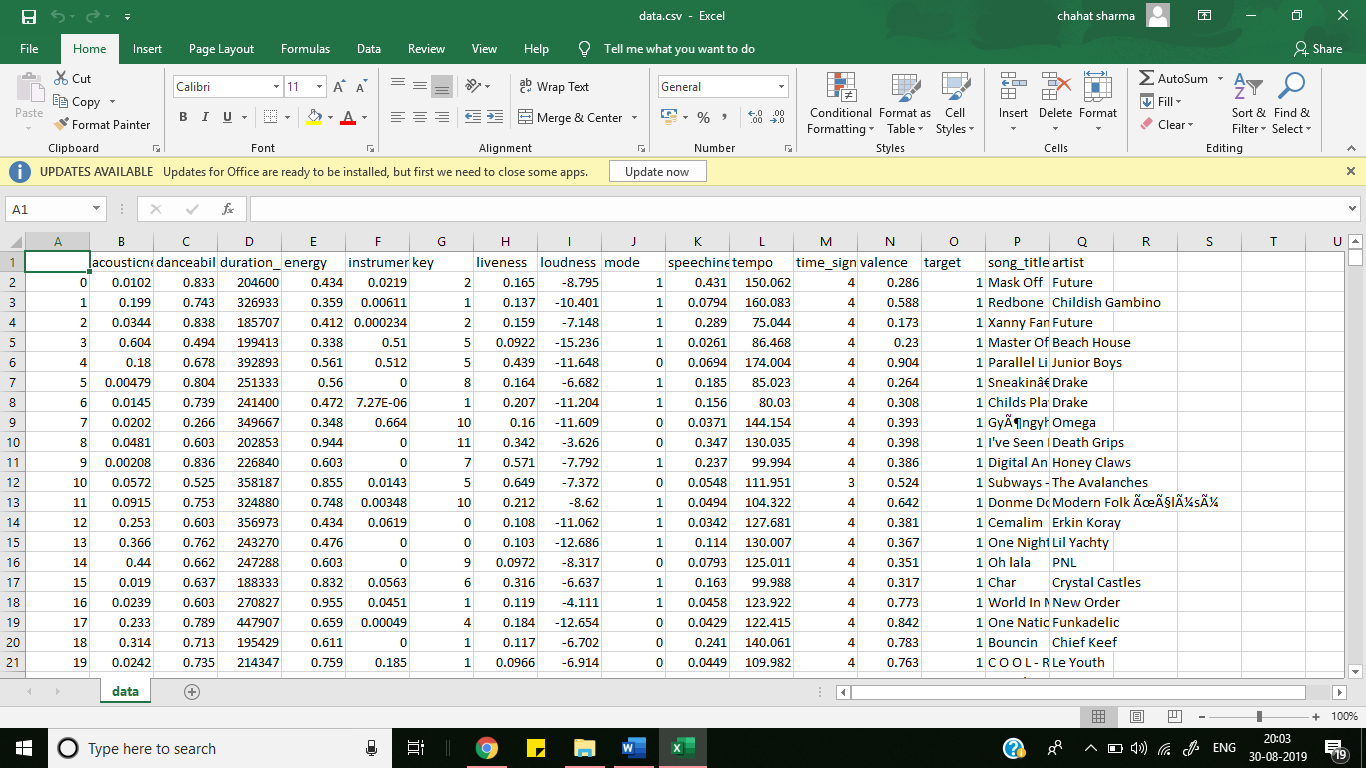
https://www.kaggle.com/aeryan/spotify-music-analysis/data

Data Set description:

The dataset consist of following columns

1. Acousticness
2. Danceability
3. Energy
4. Instumentalness
5. Key loudness
6. Mode
7. Speechiness
8. Valence
9. Artist

Screenshot of Data Set



Explanation

1. First of all we have to import some python libraries like numpy (for handling numeric

data), Pandas(for reading file from the system), matplotlib(for plotting graphs) and

sklearn (for using the already made training algorithms).

2. Read the dataset using pandas library.

3. Define the features and output from the imported dataset.

4. Now split the dataset into two parts (here we took 80:20) using test\_train\_split.

5. Now import the required Algorithm(Random Forest ,Multiple Regressor , Decision

Tree)

6. Train the model using the selected algorithm.

7. Now predict the output using x\_test dataset and store it in y\_pred.

8. Compare the y\_pred output with the y\_test and get the accuracy of the model by using

inbuilt accuracy function in sklearn.

9. Print the score.

Conclusion

For this project, we have collected synthetic Dataset from kaggle.com.

By Data Visualization, the irrelevant data has been removed and ranking of the data is done as per their influence in predicting the output. Furthermore, we took three different training algorithms namely-Decision Tree, Random Forest, Multiple Linear Regression for training the dataset and out of them Random Forest predicted with maximum accuracy. Out of the whole dataset about 80% of data were used for the training purpose and the remaining 20% for the testing purpose. In this project we have done analysis on music. “1” is used to denote that the audience will like the song and “0” is used to denote that they won’t like that song. Finally, from this project we will be able to predict that if any new song is come into play then whether audience like or dislike it.