Songs Popularity Prediction Project

group names and IDs: Basel Husam Mather – 0202247
Adel Kayyali – 0204551
Abd-alrahman Abu Rumman – 0205149

semester: Spring 22

instructor name: Dr. Sherenaz Al-Haj Baddar

Executive Summary

This project puts the prediction of the song's popularity in the spotlight. After we've done our analysis and the EDA process and made it prepared, we found that Random Forest Regressor model was the most suitable algorithm for our data, but there were some issues with the data set we had to deal with.

In the beginning, we got rid of the outliers that would mislead our model to incorrect conclusions. And because the skewnesses for some specific columns led to unfair results for the minority data in our model, fixing the shape of the distribution of the numerical features was a very important step. Finally, we split the data set into training and testing and then fitted it to our model to see our results. Finally, we've looked into the R2 score and the RMSE to evaluate our model.

1: Introduction

If there is an artist who wants to make a hit song, then it must be popular, but how can we know if a song is going to be popular or not? and what are the main properties that make the song a hit and popular song? Our goal is to determine the factors that help and affect the popularity of the song. With the cloud-based platform's resources such as RAM, Hard Disk, and the environment, using machine learning algorithms and techniques, we will make a model that can predict the popularity of a song. *(Kanjilal, 2021)*

2: Methodology

2.1. Model Description

Random Forest is a machine learning algorithm based on ensemble learning, by the decision tree algorithm. It is made for large-scale and complex data analysis *(Qi, 2012)*. we've implied the model with 500 estimators because we've different kinds of scales and a large number of features, so we need a huge number of decisions to make the prediction. after we split the data into features, and target then split them into train and test, we fitted the model with the appropriate train data, then test it by making the predictions with the test data. It was chosen after we've looked into the data visually and statistically and knew that we are dealing with uncorrelated features, because the random forest works only on a subset of features *(Yiu,2019)*, and to be honest, random forest is always a great option for modeling.

2.2. Dataset Description

The song dataset is a free source dataset from Kaggle, the main purpose/target of it is the song's popularity. It has 15 features and 18835 records/instances, each feature represents a specific specialty of a song. A brief description of each feature

- **song popularity:** the target column.
- **song duration ms:** the song duration in milliseconds.
- **acousticness:** confidence measure from 0 to 1.
- **danceability:** describe how suitable a track is for dancing.
- **energy:** movement of energy through a substance in waves.
- **instrumentalness:** represent the amount of vocals in the song.
- **key:** a system of functionally related chords.
- **liveness:** reverberation time.
- **audio mode:** whether the waves are major or minor.
- **speechiness:** detects the presence of spoken words in a track.

The dataset is from the musical industry. We've chosen this dataset to make something unserious and show that even in a big project you can make something unusual, the seriously is important but not always. as a part of the youth community, and because the music is the main part of our daily routines, we found it interesting to make this project.

Firstly, we've cleaned the data by finding outliers and handling them, by making a Box-Plot for each feature, and removing the detected outliers.

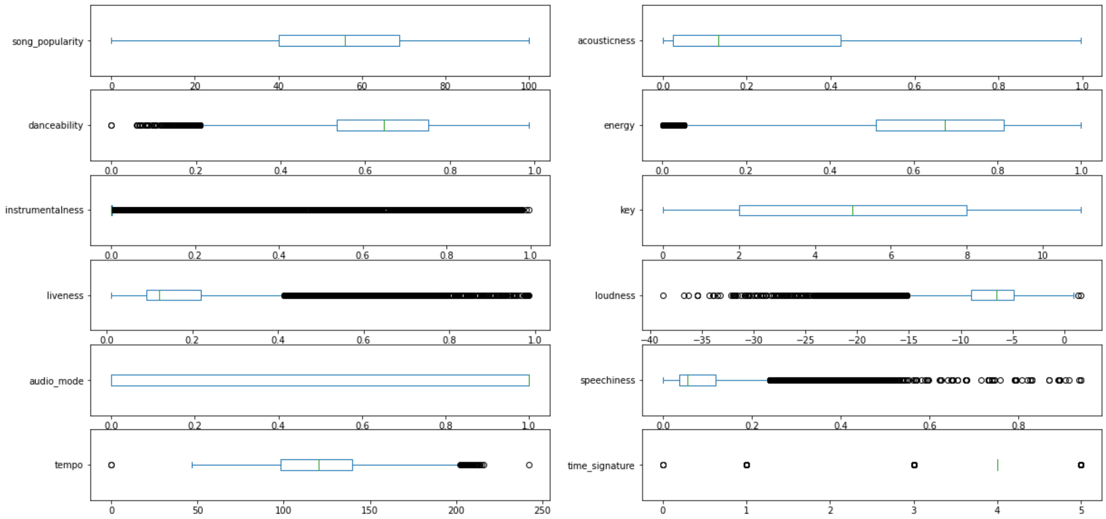

Figure 1 - BoxPlot figure for all columns

Then, we continued our analysis by looking for skewness in columns by making histograms for each column

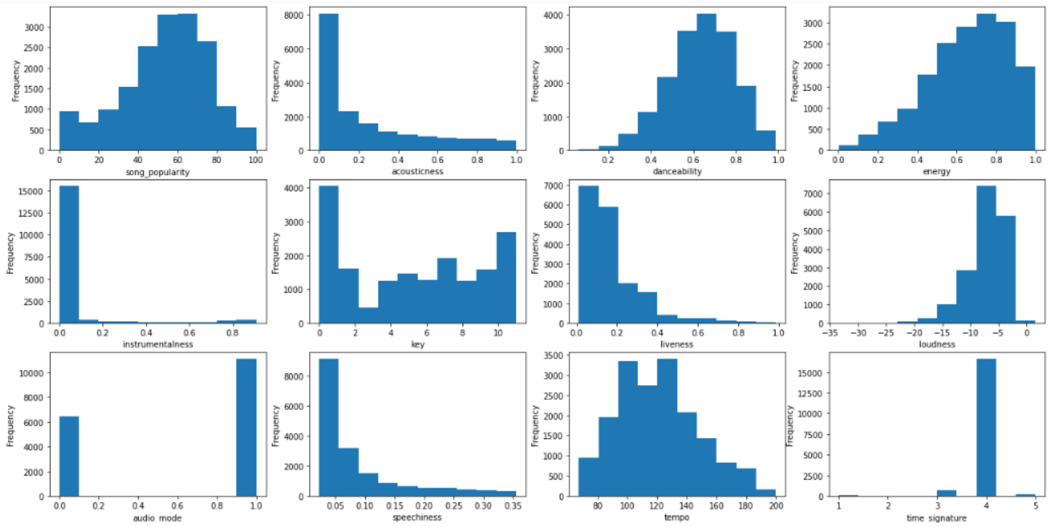

Figure 2 – Histogram figure for all columns

then fixing this skewness by applying a kind of transformation called the square root transformation to reduce the difference between features.

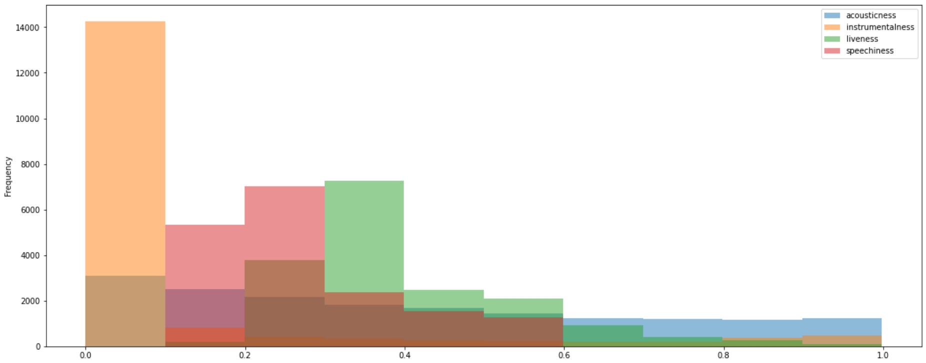

Figure 3 – The distribution after the transformation

After that, we applied the Min-Max Scaler normalization for some features that have a different scale than others. Finally, we looked into the correlation between features by making a heatmap plot using the seaborn library.

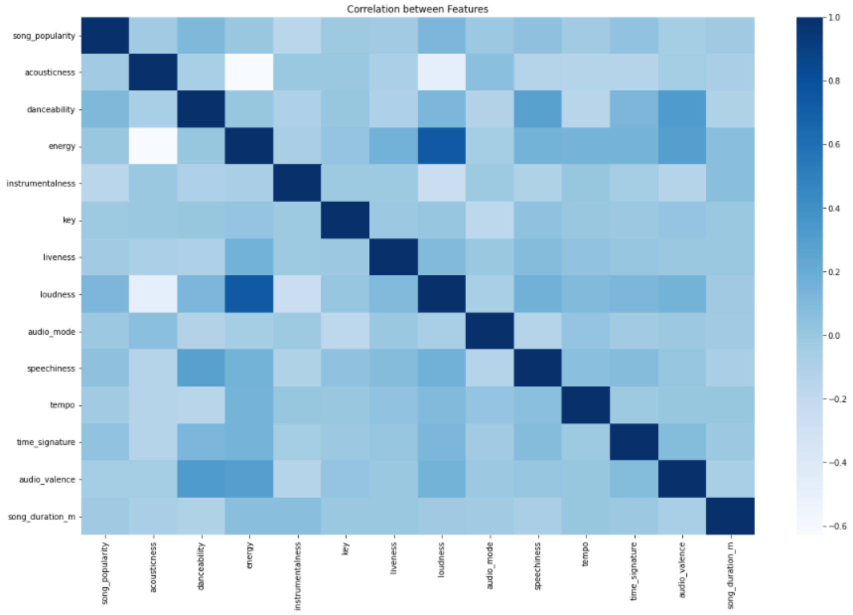

Figure 4 – heatmap figure for correlation between features

3: Results and discussion

For us, as a regression project, the best metrics to evaluate our model are the R2 score, and the RMSE (Root Mean Squared Error). After we've set our hyperparameters for our Random Forest model for the n_estimators as 500, we concluded the results below:

Quality Of The Model		Computational Metrics		
R2	RMSE	RAM (GB)	Time Spent (m)	Hard Disk (GB)
0.4168	16.58	3.38	1.7	38.72

As we can see the R2 score for our model is almost 0.42 which means that we can predict about 42.0% from the data by this model.
The RMSE is almost 16.5 which means that the predictions from the model will be far from the actual value maximum of 16.5, either higher than the actual value or lower.

For Computational Metrics we found that we used from the google colab environment about 3.38 GB of RAM and 38.72 GB of Hard Disk, and it took 1.7 seconds to build and fit our model with the data we have.

4: Conclusions

Here, we have come to the end of the project. The purpose of this research was to identify the popularity of a song. We tried many machine-learning algorithms and found out that the best algorithm and well-suit model for our data is the Random Forest algorithm. We faced many challenges in some of our features like “Instrumentalness” feature where almost all the records were outliers. But on the other hand, we did our best trying to manipulate the data to reach the highest accuracy possible for our model.
As a result, we achieved a stunning Root Mean Square Error (RMSE) which equals 16.58, and here is the features importance percentage in our data:

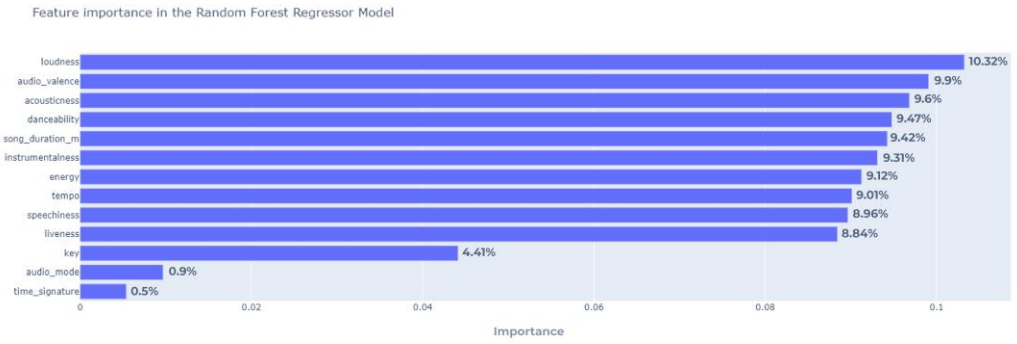

Figure 5 – Feature importance for the model

We found that the “loudness” has the highest importance for our model, so we can conclude that the more the song is loud the more the probability to be a popular song.

The ”key”, “audio_mode” and “time_signature” features have the least importance for the model, maybe that is because the number of unique values in these features is very small.

We do hope that our project will be interesting and maybe even knowledgeable.

References

Kanjilal, J., 2021. Benefits and drawbacks of AI in cloud computing. [online] SearchCloudComputing. Available at: https://www.techtarget.com/searchcloudcomputing/tip/Benefits-and-drawbacks-of-AI-in-cloud-computing [Accessed 31 May 2022].

Qi, Y., 2012. Random forest for bioinformatics. In Ensemble machine learning (pp. 307-323). Springer, Boston, MA.

Yiu, T., 2019. Understanding Random Forest. [online] Medium. Available at: https://towardsdatascience.com/understanding-random-forest-58381e0602d2 [Accessed 31 May 2022].

Date and sign

5 / 31 / 2022 Basel Mather Abd-Alrahman AbuRumman Adel Kayyali

Adel