CSCE 5310 - Methods in Empirical Analysis MOST STREAMED SPOTIFY SONG ANALYSIS USING MACHINE LEARNING ALGORITHMS AND JUPYTER NOTEBOOK

Nelapati Manideep - 11697590
Tarun Preetham Chintada - 11690923
Tharun Ramula - 11706360
Siva Kishore Reddy Putluru - 11684843

I. Goals and Objectives

A. Motivation

In this research proposal the problem is related to the project about most streamed songs on Spotify with the usage of machine learning algorithms and jupyter notebooks. The research problem about this topic is detections of songs capacity, detection of music audio data through Spotify and features of acoustics extractions from song databases. This analysis of the problem is important because it will be helping to determine the research topic deeply about the issues faced on Spotify and songs databases. On the other hand, this problem analysis will be helping to predict audio related sets of characteristics for predictions of low and high sounds through machine learning language.

B. *Significance*

This research is focused on analyzing the most streamed Spotify song by using a machine learning algorithm and Jupiter Notebook. In this context, this research has included the first step which is the data collection. In this context, the official API of Spotify is used with the help of identifying the data set followed by collecting reliable information about the data that includes title artist release date streaming count, and many others. The interpretation of the results is also done industries followed by involving the feature importance analysis to understand the audio feature which is most significant in determining the song's popularity. The research has also illustrated data visualization which has been helpful for understanding the data set and it can be helpful to identify the pattern and friends in the Spotify data analysis has also been found to be very helpful for the purpose of understanding the data as well as the trend in the data set. It has also included different data distributions and summary statistics. In this research, the structure of data is shown after the dropping of some objects in the data set. Information and used for the purpose of cleaning the data set followed by mapping the major and minor data However, the analysis also includes the checking of null values in the data set with the help of using the formula is null. However, the sum of the null values is also identified so that the overall quality of the result can be done. However, this research also included the use of a module for the purpose of configuring the model of logistic regression followed by splitting the data into train and test.

C. *Objectives*

• The main aim of the project is to investigate the most streamed songs on the Spotify using proper ML approach.
• To determine the factors that define the characteristics of a most streamed song.
• To implement the audio-based approach for identifying the audio features of the songs on Spotify.
• To identify the acoustic features comparison for determining the hyperparameter optimization as well as song predictions on Spotify

D. *Features*

• Data Collection
• Data preprocessing
• Data Analysis
• Machine learning models (Logistic Regression and Decision Tree)
• Statistical Tests

II. Introduction

Spotify is one of the best music platforms that is used to provide a list of worldwide songs to all users. A list of the songs is displayed to the user by this platform online. This is an online song-providing platform. There are varieties of songs on Spotify which has multiple categories such as English, Hindi, and so on. A variety of songs with multiple categories are provided to the user. Users can choose any option which they like most and play the song through the internet. The listed song assists in understanding the choice of a user. Various user has various choices which highlights the investigation of the streaming of the song. Online streaming on Spotify assists to understand which is the most streamed song in a year. Multiple secondary resources are used to

collect the song data with their streaming value. This assists in investigating the streaming of the song. The data investigation approach is used to investigate the most streamed song from a list of Spotify songs. The dataset contains the details of the song with song name, artist name, streaming value, and many other factors. The main functionality of the project defines the overall investigation of the collected data. This assists in understanding the most popular songs on Spotify. The ML approaches define the use of some model creation process such as "Linear regression" (LR) which is used to investigate the Spotify data.

III. Background

As of now, we've noticed a problem with some online music streaming services: the suggested song selections frequently deviate from the intended tempo continuity, negatively affecting the user experience. Our approach makes use of statistical analysis and predictive modeling grounded in past observations to tackle this problem. With the use of sophisticated statistical models and techniques, we hope to correctly identify trends in song tempos and user preferences. We can improve and optimize song recommendations because of our data-driven approach, which guarantees the accuracy of our analysis. Our objective is to improve the accuracy of tempo-aligned suggestions and give users a seamless and pleasurable listening experience through ongoing iteration and adaptation to user feedback.

IV. Dataset

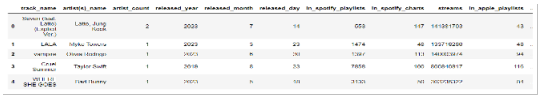

Fig. 1. Dataset

The Kaggle dataset will be utilized to develop a machine learning model for the analysis and prediction of the most popular songs on Spotify. More than 130 songs and 17 features for obtaining metadata are included in this dataset. The Kaggle dataset can be used to discuss the specifics of the data source, such as the need to first select New Notebook and then the option to add data with input and output options. The file from the Kaggle dataset is in the CSV (Comma Separated Value) file format and is roughly 48 kilobytes in size. Yes, pre-processing steps exist; the Python Anaconda terminal is used to run the codes. Thus, the first step how raw data looks like and then it must ensure that it is in usable format. In the second step, data duplicity needs to be checked and can be overcome through unique id features. There are five values in total throughout the entire dataset. There are a number of features in those features that are used to construct a predictive model that finds the inaccurate result. In the dataset, we have column variables like the artist count, in spotify playlists, in spotify charts, in apple playlists, released day, track name, artist name, released year, released month, streams and released day.

V. Analysis and Implementation

The machine learning model for prediction and dataset analysis of the most streamed Spotify songs will be used as the metric in this research proposal for evaluating performances with method feature filter selection and logistics regression (Khan, 2023). With the aid of Python and machine learning algorithms, this research proposal's methodology includes filter selection and logistics regression. Thus, Spotify data can be used to achieve an aim based on findings regarding inferring exploratory analysis. We will talk about the selected machine learning metrics as a model to deliver high popularity-based data with Spotify algorithms and popularity genre. Any quantitative experiments, aside from the evaluation of metrics, rely on music data and its imputed dataset, which includes targeted columns for music popularity. As a result, it can be discovered that 33Popular and unpopular songs on the Spotify music dataset exhibit instability, according to qualitative experiments conducted on the dataset. As a result, the experiment's machine learning algorithms are impacted. The make classification function could be used to collect information about coordinated strategies for averting imbalance. The

```
import warnings
warnings.filterwarnings('ignore')
import numpy as np
import pandas as pd
import matplotlib
import plotly.express as px
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.preprocessing import StandardScaler
```

Fig. 2. Module Import

module importation section is used to import necessary Python modules that assist in creating the necessary environment for the investigation. There are multiple modules are used for the investigation such as 'warnings' which are used to remove the warning message from Python code execution. On the other hand, 'numpy' is used to implement numeric functionality. On the other hand, 'pandas' is used to implement the read functionality of the collected data (Beesa et al., 2023). The visualization method modules of the Python coding are also used in this section which is used in the investigation such as 'matplotlib', 'plotly', and 'seaborn'. The standard calling functionality can be implemented by using the 'sklear' method.

Fig. 3. Read data and create the structure of the data

The reading of the data is the process of reading the collected data with the help of 'pandas'. The method defines the use of the 'pd.read csv' functionality. It defines the process of

execution of the read data functionality. The read functionality is used for each line of the collected dataset. Here the format of the dataset is 'csv'. The encoding process is used to encode the dataset to collect the contents of the dataset. Here the encoding process code is 'cp775'. The 'head' functionality is used to define the structure of the dataset. This functionality assists in constructing the details of the collected data. The information

```
df_sportify.info()
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 953 entries, 0 to 952
Data columns (total 24 columns):
 #   Column                Non-Null Count  Dtype
---  ------                --------------  -----
 0   track_name            953 non-null    object
 1   artist(s)_name        953 non-null    object
 2   artist_count          953 non-null    int64
 3   released_year         953 non-null    int64
 4   released_month        953 non-null    int64
 5   released_day          953 non-null    int64
 6   in_spotify_playlists  953 non-null    int64
 7   in_spotify_charts     953 non-null    int64
 8   streams               953 non-null    int64
 9   in_apple_playlists    953 non-null    int64
 10  in_apple_charts       953 non-null    int64
 11  in_deezer_playlists   953 non-null    object
 12  in_deezer_charts      953 non-null    int64
 13  in_shazam_charts      903 non-null    object
 14  bpm                   953 non-null    int64
 15  key                   858 non-null    object
 16  mode                  953 non-null    object
 17  danceability_%        953 non-null    int64
 18  valence_%             953 non-null    int64
 19  energy_%              953 non-null    int64
 20  acousticness_%        953 non-null    int64
 21  instrumentalness_%    953 non-null    int64
 22  liveness_%            953 non-null    int64
 23  speechiness_%         953 non-null    int64
dtypes: int64(18), object(6)
memory usage: 178.8+ KB
```

Fig. 4. Information of the data

in the data defines the details of the dataset. This functionality demonstrates the details of the dataset which defines column name, null value count, non-null value count, and the exact type of the data. The type of data defines 'int64', and 'object'. The value defines the count of the integer type data which is 18, and the object type data whose count is 6. The functionality of the 'info()' method is demonstrated in this section (Panda et al., 2021). This analysis in the overall investigation and understanding of which types of data can be used for the investigation.

General values are checked by using the formula that has been shown in the above figure followed by providing the information about the sum of null values. It has been found that in shazam charts and key are found to be having some null values in the data the type of all the variables in the data set are found to be in the form of integers.

The total playlist with respect to the key of the songs is shown in the above figure which represents the Spotify playlist with having different keys. In this context almost 70 percent of playlists whereas A hash and B hash have a lower number of songs. However, C hash is also near 90 percent whereas the other keys are also found to be demonstrating the list of songs in Spotify. It has also been identified that the above formula is found to be showing in the above section (Chodos, 2019). The above figure also says the playlist of Spotify and the bar plot related to its data that can be helpful for the purpose of analyzing the total playlist with respect to the key of the songs.

The total playlist with respect to mode is also demonstrated in the figure where the major more consists of 61.4 percent

```
df_sportify.isnull().sum()

track_name              0
artist(s)_name          0
artist_count            0
released_year           0
released_month          0
released_day            0
in_spotify_playlists    0
in_spotify_charts       0
streams                 0
in_apple_playlists      0
in_apple_charts         0
in_deezer_playlists     0
in_deezer_charts        0
in_shazam_charts       50
bpm                     0
key                    95
mode                    0
danceability_%          0
valence_%               0
energy_%                0
acousticness_%          0
instrumentalness_%      0
liveness_%              0
speechiness_%           0
dtype: int64
```

Fig. 5. Checking of null

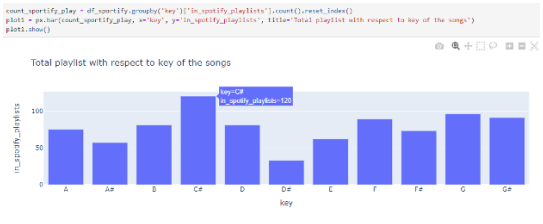

Fig. 6. Total playlist concerning the key of the sings

Fig. 7. Total playlist with respect to mode

of the playlist and the minor more consists of 38.6 percent of the total data set.

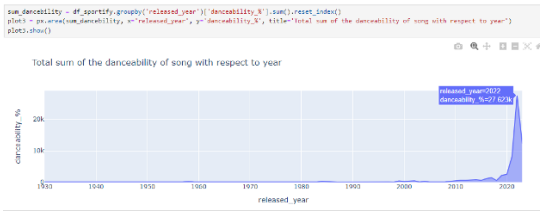

Fig. 8. Total sum of the danceability of the song with respect to the year

The total sum of the danceability of the song with respect to the ear is also demonstrated in the above figure where the danceability percentage and released year are shown in the above figure according to which the danceability percentage is 27.623 cases and in the release year 2022.

Fig. 9. Total sum of the speechiness percentage song with respect to key

The total sum of speechless percentage songs with respect to the key is shown in the above figure pair the speechless percentage is 1448 for key C hash. The other percentage is also shown in the above figure as per the graph illustration.

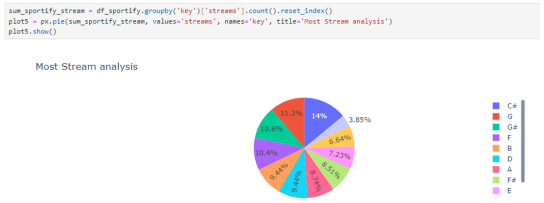

Fig. 10. Most Stream Song Analysis

The most streamed song analysis is shown in the above figure pair the most streamlined song is C which is equivalent to 14 percent and the G hash is equivalent to 10.6 percent. F hash is 8.51 percent and E is equal to 7.23.

Fig. 11. Most Stream Song Analysis with respect to song modes

The most streamed song analysis with respect to the song is also shown in the above figure according to which the major song is equal to 57.7 percent whereas minor songs are equal to 42.3 percent.

```
df_sportify.drop(['artist(s)_name','track_name', 'in_deezer_playlists', 'in_shazam_charts', 'key'], axis=1, inplace=True)
```

Fig. 12. Drop object columns

The drop object columns are shown in the above figure where the different objects in the column are dropped such as the artist's name track name and others. However, the columns are used for dropping the objects for the purpose of enhancing the quality of the overall data after the analysis (Werner, 2020).

```
df_sportify.head()
```

	artist_count	released_year	released_month	released_day	in_spotify_playlists	in_spotify_charts	streams	in_apple_playlists	in_apple_charts	in_deezer_charts	bpm
0	2	2023	7	14	553	147	141381703	43	263	10	125
1	1	2023	3	23	1474	48	133716286	48	126	14	92
2	1	2023	6	30	1397	113	140003974	94	207	14	138
3	1	2019	8	23	7858	100	800840817	116	207	12	170
4	1	2023	5	18	3133	50	303236322	84	133	15	144

Fig. 13. Structure of the data after drop column

The structure of the data after the drop column is shown in a figure that has been found to demonstrate the artist count released year released month and many others. As per the above figure, the structure of data can be visualized after the drop of the column.

```
df_sportify.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 953 entries, 0 to 952
Data columns (total 19 columns):
 #   Column                Non-Null Count  Dtype
---  ------                --------------  -----
 0   artist_count          953 non-null    int64
 1   released_year         953 non-null    int64
 2   released_month        953 non-null    int64
 3   released_day          953 non-null    int64
 4   in_spotify_playlists  953 non-null    int64
 5   in_spotify_charts     953 non-null    int64
 6   streams               953 non-null    int64
 7   in_apple_playlists    953 non-null    int64
 8   in_apple_charts       953 non-null    int64
 9   in_deezer_charts      953 non-null    int64
 10  bpm                   953 non-null    int64
 11  mode                  953 non-null    object
 12  danceability_%        953 non-null    int64
 13  valence_%             953 non-null    int64
 14  energy_%              953 non-null    int64
 15  acousticness_%        953 non-null    int64
 16  instrumentalness_%    953 non-null    int64
 17  liveness_%            953 non-null    int64
 18  speechiness_%         953 non-null    int64
dtypes: int64(18), object(1)
memory usage: 141.6+ KB
```

Fig. 14. Information of cleaned data

The information of clean the data found is demonstrated in the above figure in which the memory usage is shown along with providing the non-null count value for the data type are also shown of all data followed by demonstrating the information of the attributes that are used in the data set (Kalustian, and Ruth, 2021).

The conversion of the object column is also illustrated in the figure which demonstrates the major and minor objects of the data set.

```
df_sportify['mode'] = df_sportify['mode'].map({'Major': 1, 'Minor': 0})
```

Fig. 15. Conversion of an object column

```
df_sportify.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 953 entries, 0 to 952
Data columns (total 19 columns):
 #   Column                Non-Null Count  Dtype
---  ------                --------------  -----
 0   artist_count          953 non-null    int64
 1   released_year         953 non-null    int64
 2   released_month        953 non-null    int64
 3   released_day          953 non-null    int64
 4   in_spotify_playlists  953 non-null    int64
 5   in_spotify_charts     953 non-null    int64
 6   streams               953 non-null    int64
 7   in_apple_playlists    953 non-null    int64
 8   in_apple_charts       953 non-null    int64
 9   in_deezer_charts      953 non-null    int64
 10  bpm                   953 non-null    int64
 11  mode                  953 non-null    int64
 12  danceability_%        953 non-null    int64
 13  valence_%             953 non-null    int64
 14  energy_%              953 non-null    int64
 15  acousticness_%        953 non-null    int64
 16  instrumentalness_%    953 non-null    int64
 17  liveness_%            953 non-null    int64
 18  speechiness_%         953 non-null    int64
dtypes: int64(19)
memory usage: 141.6 KB
```

Fig. 16. Details of the data

The details of the data are shown in the above figure which shows the normal count values along with demonstrating the data type and the memory usage. The attributes and the total entries are also found to be shown in the above figure with the help of that that can be understood in an appropriate manner.

The null values are checked by using the above figure in which the identification of null values is done followed by providing the data type (Al-Beitawi et al., 2020).

The modules are used for the models along with demonstrating the importing of logistic regression and train-test split. The classification report and the accuracy score are also the modules that have been imported for the purpose of getting the appropriate data related to it.

The setting of X and Y data is also in the above figure followed by providing the drop of different values. However, the Spotify data are also used for the purpose of dropping the mode from the axis.

The speed of data is done in the above figure according to which the data are split into train and test taking the test size to be 0.2 and the random state be 42. According to this information, data has been found to be demonstrating accurate value and analysis for the data set.

The logistic regression is implemented in the data set for getting the train and test value in which the logistic regression is used for the purpose of splitting the data set as well as creating the model.

The prediction of implementation is done in the above figure that provides the prediction figure X test data.

```
df_sportify.isnull().sum()
```

```
artist_count          0
released_year         0
released_month        0
released_day          0
in_spotify_playlists  0
in_spotify_charts     0
streams               0
in_apple_playlists    0
in_apple_charts       0
in_deezer_charts      0
bpm                   0
mode                  0
danceability_%        0
valence_%             0
energy_%              0
acousticness_%        0
instrumentalness_%    0
liveness_%            0
speechiness_%         0
dtype: int64
```

Fig. 17. Checking of null

```
from sklearn.linear_model import LogisticRegression
from sklearn.model_selection import train_test_split
from sklearn.metrics import accuracy_score, classification_report
```

Fig. 18. Use modules for model configuration

```
X = df_sportify.drop('mode', axis=1)
y = df_sportify['mode']
```

Fig. 19. : Setting of X and y

```
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
```

Fig. 20. Split of data

```
lr_model = LogisticRegression()
lr_model.fit(X_train, y_train)
```

```
▾ LogisticRegression
LogisticRegression()
```

Fig. 21. Logistic Regression (LR)

```
lr_y_pred = lr_model.predict(X_test)
```

Fig. 22. Prediction implementation

The accuracy of the logistic regression is shown in the above figure which provides the value to be 57 percent.

The classification report is shown in the above figure in which the precision value is seen to be zero for zero and .57 for 1. Macro average value and weighted average value are also shown above. It has also been found that the recall value is also shown. The F1 score values of the accuracy, macro average, and weighted average are also shown in the above figure.

VI. Preliminary Results

The logistic regression is implemented in the data set for getting the train and test value in which the logistic regression is used for the purpose of splitting the data set as well as creating the model. The prediction of implementation is done in the above figure that provides the prediction figure X test data. The accuracy of the logistic regression is shown in the above figure which provides the value to be 57 percent. The classification report is shown in the above figure in which the precision value is seen to be zero for zero and .57 for

```
accuracy_lr = accuracy_score(y_test, lr_y_pred)
print('Accuracy of Logistic Regression is ', round(accuracy_lr,2))
```

```
Accuracy of Logistic Regression is  0.57
```

Fig. 23. LR accuracy

```
lr_report = classification_report(y_test, lr_y_pred)

print('Classification Report:\n', lr_report)
```

```
Classification Report:
              precision    recall  f1-score   support

           0       0.00      0.00      0.00        83
           1       0.57      1.00      0.72       108

    accuracy                           0.57       191
   macro avg       0.28      0.50      0.36       191
weighted avg       0.32      0.57      0.41       191
```

Fig. 24. Classification report


```
lr_model = LogisticRegression()
lr_model.fit(X_train, y_train)
```

```
▾ LogisticRegression
LogisticRegression()
```

```
lr_y_pred = lr_model.predict(X_test)
```


1. Macro average value and weighted average value are also shown above. It has also been found that the recall value is also shown. The F1 score values of the accuracy, macro average, and weighted average are also shown in the above figure.

VII. Project Management

A. *Work Completed*

B. *Description*

We preprocessed and cleaned the data, presented the results using data visualization techniques, and used a variety of models like Logistic Regression to estimate how accurate the outcomes would be. We also performed several statistical tests to look at sample means for the whole population.

C. *Responsibility*

Data preprocessing – Manideep Nelapati
Data cleaning and visualization – Tarun Preetham Chintada
Data modeling – Tharun Ramula
Statistical approach – Siva Kishore Reddy

D. *Contribution*

Nelapati Manideep – 25 percent
Tarun Preetham Chintada – 25 percent
Tharun Ramula – 25 percent
Siva Kishore Reddy Putluru – 25 percent

E. *Work To Be Completed*

F. *Description*

We will use the Decision Tree algorithm to conduct a few more statistical analyses in the final section. It is a supervised learning algorithm that can handle regression and classification tasks. Additionally, it is a tree structure with leaf, internal, and root nodes. Additionally, we'll use a variety of visualization techniques for data visualization.

G. *Responsibility*

Data preprocessing – Manideep Nelapati Data cleaning and visualization – Tarun Preetham Chintada Data modeling – Tharun Ramula Statistical approach – Siva Kishore Reddy

H. Issues

The technical challenges associated with the research topic of Spotify music streaming and its analysis using machine learning algorithms via Jupyter notebooks. Acoustics features comparison, engineering, and hyperparameter optimization, as well as attribute-based Spotify audio features and song prediction from the Spotify music database are examples of technical issues that can be discussed. This research project can employ correlation values and machine learning models to evaluate efficient music search methods. The most searched songs are found using logistic regression and feature filter selection in the context of the most streamed songs on Spotify. The method for improving music evaluation through machine learning and correlation coefficients.

REFERENCES

[1] Al-Beitawi, Z., Salehan, M. and Zhang, S., 2020. What makes a song trend? Cluster analysis of musical attributes for Spotify top trending songs. Journal of Marketing Development and Competitiveness, 14(3), pp.79-91.

[2] Álvarez, P., García de Quirós, J. and Baldassarri, S., 2023. RIADA: A Machine-Learning Based Infrastructure for Recognising the Emotions of Spotify Songs.

[3] Araujo, C.V.S., De Cristo, M.A.P. and Giusti, R., 2019, December. Predicting music popularity using music charts. In 2019 18th IEEE International Conference On Machine Learning And Applications (ICMLA) (pp. 859-864). IEEE.

[4] Beesa, P., Naregavi, V., Imandar, J. and Thatte, S., 2023. Songs Popularity Analysis Using Spotify Data: An exploratory study. Vidhyayana-An International Multidisciplinary Peer-Reviewed E-Journal-ISSN 2454-8596, 8(si7), pp.211-223.

[5] Chodos, A.T., 2019. What does music mean to Spotify? An essay on musical significance in the era of digital curation. INSAM Journal of Contemporary Music, Art and Technology, 1(2), pp.36-64.

[6] Dawson Jr, C.E., Mann, S., Roske, E. and Vasseur, G., 2021. Spotify: You have a Hit!. SMU Data Science Review, 5(3), p.9.

Github Link : https://github.com/ManideepAI-Project/Emprical_Analysis.git