Harmonizing Insights: Python-Based Data Analysis of Spotify's Musical Tapestry

Deepesh Trivedi[1], Manas Saxena[1], S. S. P. M. Sharma B[2], and Indrajeet Kumar[3](✉)

[1] School of CSIT, Symbiosis University of Applied Sciences, 453112 Indore, India
[2] School of MT, Symbiosis University of Applied Sciences, Indore, India
[3] School of CSIT, Symbiosis University of Applied Sciences, Indore, India
indrajeet.kumar@suas.ac.in

Abstract. This research paper analysis Spotify data using Python to investigate the characteristics contributing to song popularity. The objectives are to assess the popularity index, identify key attributes of popular songs, and develop a model for predicting song popularity based on current characteristics. The analysis involves data cleaning, exploratory data analysis, and visualization using Python libraries. With over 381 million monthly active users, Spotify provides a rich dataset for understanding music listening habits. Previous studies have explored Spotify's technologies and popularity, enhancing understanding of its protocols and user behavior. This research paper aims to uncover patterns and relationships within the data by applying statistical and machine-learning techniques. The findings will inform actionable recommendations and contribute to a better understanding of music consumption patterns and preferences.

Keywords: Analysis · exploratory data analysis · machine-learning techniques · spotify

1 Introduction

Data analysis has become essential for gaining valuable insights and understanding user behavior in various industries. In the realm of music streaming, Spotify stands out as one of the leading platforms with a vast amount of user data [1]. By leveraging data analysis techniques, we can delve into Spotify's rich dataset to uncover patterns, identify popular song attributes, and gain a deeper understanding of listener preferences [2– 5]. Python, a versatile programming language, provides a powerful ecosystem of libraries and tools for data analysis. With its robust capabilities, Python has become a popular choice for conducting data analysis on platforms like Spotify [6–8]. By utilizing Python's data manipulation, visualization, and machine learning libraries, we can efficiently clean, explore, and analyze Spotify data to extract meaningful insights [9].

The objective of data analysis on Spotify using Python is to explore the relationships between various song characteristics and their popularity, predict song popularity based

P. Pareek et al. (Eds.): IC4S 2023, LNICST 536, pp. 28–44, 2024.
https://doi.org/10.1007/978-3-031-48888-7_3

on specific attributes, and provide actionable recommendations for music professionals and enthusiasts [10–13]. By applying statistical techniques, machine learning algorithms, and data visualization, we can uncover trends, patterns, and correlations that contribute to a better understanding of music consumption habits [14]. Overall, data analysis on Spotify using Python opens up a world of possibilities to uncover hidden trends and patterns within the vast music library. It empowers us to make informed decisions, optimize user experiences, and enhance the overall music streaming ecosystem [15]. Data analysis on Spotify using Python has gained significant attention in the field of music analysis, enabling researchers and analysts to explore the vast amount of data generated by the platform [16].

In recent times, music streaming platforms like Wynk Music, Apple Music, and Spotify have witnessed an overwhelming influx of users and artists. These platforms serve as a hub for artists to upload their audio tracks [17], while also allowing users to discover and listen to their favorite songs [18]. In light of this scenario, the development of a predictive system capable of gauging the popularity of artists becomes crucial [19–22]. The application of Multiple-Input Multiple-Output (MIMO) technology in the context of music streaming services, particularly for Spotify, has been an area of interest in recent research. MIMO, a well-established technique in wireless communications, involves using multiple antennas at both the transmitter and receiver to improve data throughput and link reliability [23–26].

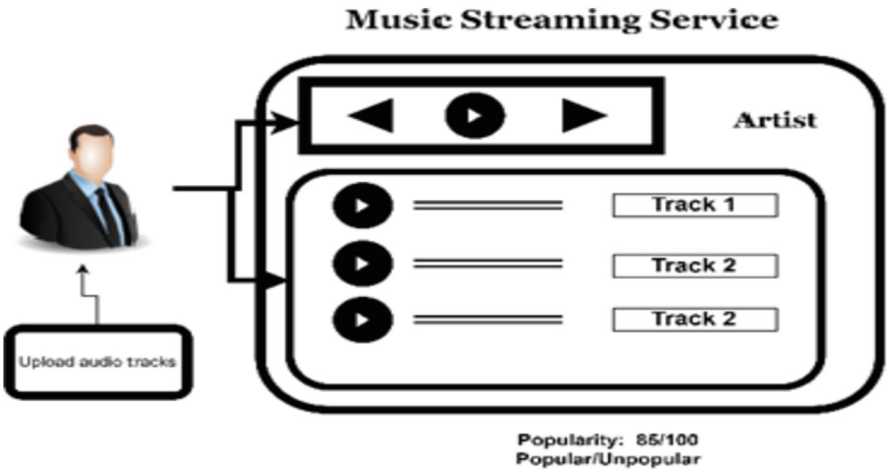

Fig. 1. Illustrating the Efficacy of Popularity Estimation: Demonstrating the Value in Assessing Artist Popularity

In the context of Spotify's music streaming platform, researchers and experts have explored how MIMO can enhance the user experience, optimize network performance, and address the challenges related to audio streaming [27, 28]. One of the primary focuses of MIMO for Spotify is to improve audio quality during music playback. By leveraging multiple antennas, MIMO can mitigate channel fading and reduce the impact of signal degradation, resulting in more stable and consistent audio streaming [29]. This enhancement is crucial for providing listeners with a seamless and immersive music experience without disruptions or audio artifacts [30]. Such a system can greatly benefit artists in planning strategies to enhance their visibility and reach. By leveraging the predictive capabilities of this system (as depicted in Fig. 1), artists can gain insights into the potential impact of uploading their audio tracks or tweaking elements like the content of their biographies [31]. They can simulate the potential popularity achieved through the predicted system, enabling them to make informed decisions regarding their promotional efforts. This growing demand for artist popularity prediction in music streaming services emphasizes the need for effective predictive models to cater to these requirements [32–36].

1.1 Online Music Services

According to Hall (2018), one of the most popular on-demand music services with a large user base is Spotify, which allows listeners to stream full-length content over the Internet without the need for purchasing or downloading. As of July 2017, Spotify had 60 million subscribers, and by January 2018, the number increased to 70 million (Hall 2018) [5]. The extensive repertoire of Spotify includes over 30 million songs, contributing to its widespread adoption and popularity (Hall, 2018). Previous studies conducted by Kreitz (2010), Loiacono (2014), and Verkoelen have examined various aspects of Spotify's technologies and its user base.

In a separate study, researchers focused on investigating the protocols and peer-to-peer architecture of Spotify to gain insights into its functioning and user interactions. They also explored the impact of the peer-to-peer network on user access patterns, properly referencing the specific report or publication where this study is mentioned will ensure accurate attribution.

1.2 Melody in the Machine: Harnessing Machine Learning to Forecast Chart-Topping Hits

The availability of a vast amount of digital music online and advancements in technology have significantly influenced music consumption habits. Kaminskas and Ricci (2012) suggest that users now search for specific music collections and rely on automatic playlist recommendations. In the field of Music Information Retrieval, researchers have been studying these concepts (Kaminskas & Ricci 2012).

This research aims to investigate the potential of utilizing 13 audio factors to predict the success of songs. The study employs four distinct machine learning techniques, namely logistic regression, K-nearest neighbors, Gaussian Naïve Bayes, and Support Vector Machine. The objective is to analyze how these techniques can effectively predict the success or popularity of a given song based on its audio characteristics. The researcher compared the results obtained from these models using the available data (Table 1).

Table 1. Model comparisons result.

S. no	Model	Accuracy
1	K-nearest Neighbours	52.00%
2	Logistic Regression	58.27%
3	Support Vector Machine	51.98%
4	Gaussian Naïve Bayes	60.50%

In this research paper, we aim to collect and clean Spotify data, perform exploratory data analysis, develop models to predict song popularity and create visualizations to effectively communicate our findings. Through this analysis, we can gain valuable insights into listener preferences, identify key factors driving song popularity, and make data-driven decisions for music curation, recommendation systems, and marketing strategies.

2 System Model

To build a system model for data analysis on Spotify using Python, you can follow these general steps:

2.1 Data Collection

The datasets used in this research are obtained from Kaggle. Kaggle is a popular online platform that hosts a wide range of datasets contributed by the data science community. It serves as a repository for diverse datasets across various domains, including music, finance, healthcare, and more. Researchers often rely on Kaggle to access high-quality datasets that are readily available for analysis and experimentation. In this particular study, the researchers downloaded the necessary datasets from Kaggle to conduct their analysis on predicting song success using machine learning techniques.

The Fig. 2 represents a visual depiction of the valuable insights obtained from analyzing the datasets provided by Spotify. These datasets contain a wealth of information related to the music available on the Spotify platform, including details about tracks, artists, genres, popularity, audio features, and more. The figure showcases the process of extracting meaningful insights from the Spotify datasets through data analysis and exploration techniques. It signifies the exploration of patterns, trends, and relationships

within the music data, leading to a deeper understanding of the soundscape offered by Spotify. By studying the Spotify datasets, researchers, analysts, and music enthusiasts can gain valuable insights into various aspects of music consumption, artist popularity, genre preferences, and user behavior. These insights can be used for diverse purposes, such as improving recommendation algorithms, understanding audience preferences, identifying emerging trends, and supporting decision-making in the music industry.

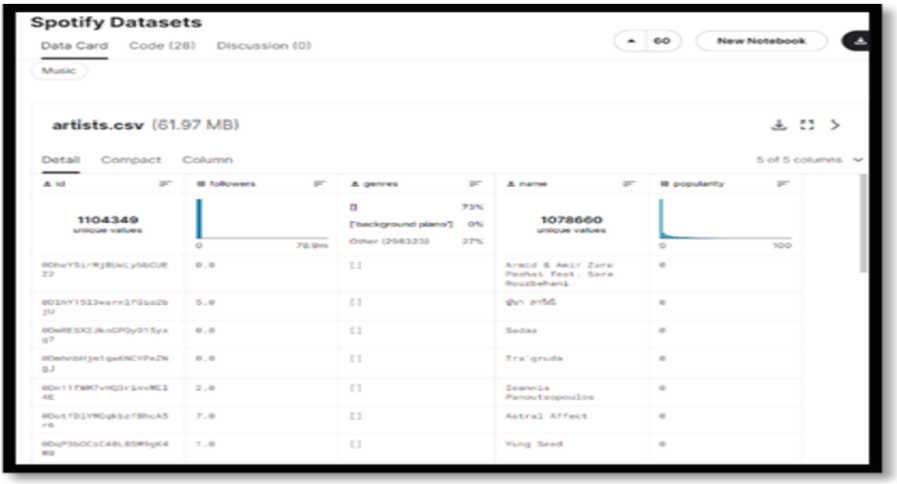

Fig. 2. Sounds of Spotify: Insights from the Spotify Datasets

Fig. 3. Harmonizing the Spotify Soundscape: Unveiling Insights from the Spotify Tracks Database

The Fig. 3 represents a database specifically dedicated to storing and organizing information about tracks from the music streaming platform, Spotify. This database contains a comprehensive collection of data related to various songs available on Spotify, including details such as track titles, artist names, album information, release dates, genres, audio features, and other relevant attributes. The Spotify Tracks DB serves as a valuable resource for researchers, analysts, and music enthusiasts who are interested in exploring and studying the vast musical landscape found on Spotify. It provides a structured and organized repository of track-related information, enabling users to query and analyze the data for various purposes. Researchers can leverage this database to investigate trends, patterns, and relationships within the music catalog, while analysts can derive insights to support decision-making in areas such as playlist curation, artist promotion, and user recommendation systems. Overall, the "Spotify Tracks DB" figure symbolizes the wealth of data available within the database, serving as a foundation for in-depth exploration and analysis of Spotify's vast music collection.

2.2 Data Pre-Processing

(A) To identify null values in the dataset
We have used the 'isnull()' function provided by the Pandas library. This function allows you to check for the existence of missing values within the dataset. In Fig. 4, the data frame is passed as an argument to the 'isnull()' function, which identifies the null values. By using the 'sum ()' function, we can calculate the total number of columns in the dataset that contain null values.

```
id                    0
name                 71
popularity            0
duration_ms           0
explicit              0
artists               0
id_artists            0
release_date          0
danceability          0
energy                0
key                   0
loudness              0
mode                  0
speechiness           0
acousticness          0
instrumentalness      0
liveness              0
valence               0
tempo                 0
time_signature        0
dtype: int64
```

Fig. 4. Null value in data frame

By examining all the columns in the dataset, it was observed that the "song name" column contains a total of 71 null values.

(B) to determine the total number of rows and columns in the dataset, as well as inspect the data types and memory usage, the "Info ()" method can be employed
This method provides a concise summary of the dataset, displaying the column names, number of non-null values, data types, and approximate memory usage. By using the "info()" method, you can obtain this information efficiently (Fig. 5).

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 586672 entries, 0 to 586671
Data columns (total 20 columns):
 #   Column            Non-Null Count   Dtype
---  ------            --------------   -----
 0   id                586672 non-null  object
 1   name              586601 non-null  object
 2   popularity        586672 non-null  int64
 3   duration_ms       586672 non-null  int64
 4   explicit          586672 non-null  int64
 5   artists           586672 non-null  object
 6   id_artists        586672 non-null  object
 7   release_date      586672 non-null  object
 8   danceability      586672 non-null  float64
 9   energy            586672 non-null  float64
 10  key               586672 non-null  int64
 11  loudness          586672 non-null  float64
 12  mode              586672 non-null  int64
 13  speechiness       586672 non-null  float64
 14  acousticness      586672 non-null  float64
 15  instrumentalness  586672 non-null  float64
 16  liveness          586672 non-null  float64
 17  valence           586672 non-null  float64
 18  tempo             586672 non-null  float64
 19  time_signature    586672 non-null  int64
dtypes: float64(9), int64(6), object(5)
memory usage: 89.5+ MB
```

Fig. 5. Total number of rows and columns in dataset, as well as inspect the data types and memory usage.

(C) To retrieve a list of the ten least popular songs from the Spotify dataset

We have employed the "sort_values ()" function to arrange the data in ascending order based on the popularity column. This will allow you to identify the songs with the lowest popularity scores (Fig. 6).

	id	name	popularity	duration_ms	explicit	artists	id_artists	release_date	danceability	energy
546130	181rTRhCogg2PwP2TUcVqm	Newspaper Reports On Abner, 20 February 1935	0	896575	0	['Norris Goff', 'Chester Lauck', 'Carlton Bric...	['3WCwCPDMpGzr0Qz6quumy', '7vk8UqABg0Sga78G13...	1935-02-20	0.595	0.262
546222	0yOCz3V5KMm8i1T8EFc8Oi	恋は水の上で	0	188440	0	['Hibari Misora']	['1m5pMYSbIqJwdxJ7vqQtuN']	1949	0.418	0.388
546221	0y48Hhwe52099UqYjegRCO	私の誕生日	0	173467	0	['Hibari Misora']	['1m5pMYSbIqJwdxJ7vqQtuN']	1949	0.642	0.178
546220	0xCmgzf9ku07hkZg3D6PaV	エル・チョクロ (EL CHOCLO)	0	205280	0	['Hibari Misora']	['1m5pMYSbIqJwdxJ7vqQtuN']	1949	0.695	0.467
546219	0tBXS3VuCPX7KWUFH2nros	恋は不思議なもの	0	185733	0	['Hibari Misora']	['1m5pMYSbIqJwdxJ7vqQtuN']	1949	0.389	0.388
546218	0qrKnQrYDVJhKFAXTHYVS9	ゆうべはどうしたの (WHATSA MALLA U)	0	183427	0	['Hibari Misora']	['1m5pMYSbIqJwdxJ7vqQtuN']	1949	0.631	0.249
546217	0nqsDxOeKSwEzp3AUQAAqS	Screen Director's Playhouse, Music For Million...	0	1767071	0	['Wilms Herbert', 'June Allyson', 'Joseph Kear...	['2rbm8QWvmnVexFo84EVM1h', '4yW5adMgyf8HFzaL9i...	1949-04-10	0.533	0.317
546216	0kGEdsxVLYjCdfxM9tbezd	ブルーマンボ	0	162147	0	['Hibari Misora']	['1m5pMYSbIqJwdxJ7vqQtuN']	1949	0.529	0.546
546215	0bc3PUZurUUX/Y7yqoOxjq	Screen Director's Playhouse, Trade Winds direc...	0	1776652	0	['Wally Maher', 'Tay Garnett', 'Lurene Tuttle'...	['7hkhJTTI3VnUGVWUt8SJXT', '3kYeelpRCgJz4tQYDv...	1949-05-19	0.599	0.321

Fig. 6. Total number of rows and columns in dataset, as well as inspect the data types and memory usage.

2.3 Data Exploration

To obtain descriptive statistics for numerical variables within the dataset, you can use the "describe ()" function. Additionally, applying the "transpose ()" function will provide a more convenient format for the summary statistics. By using these functions, you can gain insights into the central tendency, dispersion, and distribution of the numerical variables in the dataset (Figs. 7 and 8).

	count	mean	std	min	25%	50%	75%	max
popularity	586672.0	27.570053	18.370642	0.0	13.0000	27.000000	41.00000	100.000
duration_ms	586672.0	230051.167286	126526.087418	3344.0	175093.0000	214893.000000	263867.00000	5621218.000
explicit	586672.0	0.044086	0.205286	0.0	0.0000	0.000000	0.00000	1.000
danceability	586672.0	0.563594	0.166103	0.0	0.4530	0.577000	0.68600	0.991
energy	586672.0	0.542036	0.251923	0.0	0.3430	0.549000	0.74800	1.000
key	586672.0	5.221603	3.519423	0.0	2.0000	5.000000	8.00000	11.000
loudness	586672.0	-10.206067	5.089328	-60.0	-12.8910	-9.243000	-6.48200	5.376
mode	586672.0	0.658797	0.474114	0.0	0.0000	1.000000	1.00000	1.000
speechiness	586672.0	0.104864	0.179893	0.0	0.0340	0.044300	0.07630	0.971
acousticness	586672.0	0.449863	0.348837	0.0	0.0969	0.422000	0.78500	0.996
instrumentalness	586672.0	0.113451	0.266868	0.0	0.0000	0.000024	0.00955	1.000
liveness	586672.0	0.213935	0.184326	0.0	0.0983	0.139000	0.27800	1.000
valence	586672.0	0.552292	0.257671	0.0	0.3460	0.564000	0.76900	1.000
tempo	586672.0	118.464857	29.764108	0.0	95.6000	117.384000	136.32100	246.381
time_signature	586672.0	3.873382	0.473162	0.0	4.0000	4.000000	4.00000	5.000

Fig. 7. Descriptive Statistics on the dataset.

(A) To find the top ten popular songs with a popularity score greater than 90.

	id	name	popularity	duration_ms	explicit	artists	id_artists	release_date	danceability
93802	4iJyoBOLtHqaGxP12qzhQI	Peaches (feat. Daniel Caesar & Giveon)	100	198082	1	['Justin Bieber', 'Daniel Caesar', 'Giveon']	['1uNFoZAHBGtllmzznpCI3s', '20wkVLutqVOYrc0kxF...	2021-03-19	0.677
93803	7lPN2DXiMsVn7XUKtOW1CS	drivers license	99	242014	1	['Olivia Rodrigo']	['1McMsnEElThX1knmY4oliG']	2021-01-08	0.585
93804	3Ofmpyhv5UAQ70mENzB277	Astronaut In The Ocean	98	132780	0	['Masked Wolf']	['1uU7g3DNSbsu0QjSEqZtEd']	2021-01-06	0.778
92810	5QO79kh1waicV47BqGRL3g	Save Your Tears	97	215627	1	['The Weeknd']	['1Xyo4u8uXC1ZmMpatF05PJ']	2020-03-20	0.680
92811	6tDDoYIxWvMLTdKpjFkc1B	telepatía	97	160191	0	['Kali Uchis']	['1U1el3k54VvEUzo3ybLPlM']	2020-12-04	0.653
92813	0VjIjW4GlUZAMYd2vXMi3b	Blinding Lights	96	200040	0	['The Weeknd']	['1Xyo4u8uXC1ZmMpatF05PJ']	2020-03-20	0.514
93805	7MAibcTli4IisCtbHKrGMh	Leave The Door Open	96	242096	0	['Bruno Mars', 'Anderson .Paak', 'Silk Sonic']	['0du5cEVh5yTK9QJze8zA0C', '3jK9MiCrA42lLAdMGU...	2021-03-05	0.586
92814	6f3Slt0GbA2bPZlz0aIFXN	The Business	95	164000	0	['Tiësto']	['2o5jDhtHVPhrJdv3cEQ99Z']	2020-09-16	0.798
91866	60ynsPSSKe6O3sfwRnIBRf	Streets	94	226987	1	['Doja Cat']	['5cj0lLjcoR7YOSnhnX0Po5']	2019-11-07	0.749
92816	3FAJ6O0NOHQV8Mc5Ri6ENp	Heartbreak Anniversary	94	198371	0	['Giveon']	['4fxd5Ee7UefO4CUXgwJ7IP']	2020-03-27	0.449
92817	27OeeYzk6klgBh83TSvGMA	WITHOUT YOU	94	161385	1	['The Kid LAROI']	['2tIP7SsRs7vjIcLrU85W8J']	2020-11-06	0.662
92819	1xK1Gg9SxG8fy2Ya373oqb	Bandido	94	232853	0	['Myke Towers', 'Juhn']	['7iK8PXO48WeuP03g8YR51W', '2LmcxBak1alK1bf7d1...	2020-12-10	0.713
93807	4cG7HUWYHBV6R8tHn1gxrl	Friday (feat. Mufasa & Hypeman) - Dopamine Re-...	94	169153	0	['Riton', 'Nightcrawlers', 'Mufasa & Hypeman',...	['7i9j813KFoSBMldGqlh2Z1', '1gALaWbNDnwS2ECV09...	2021-01-15	0.824

Fig. 8. Shows the top ten popular songs with a popularity score greater than 90

(B) To set the release date column as the index column in the dataset.
We have used the "set_index()" function. By applying this function, you can designate the release date column as the new index for the dataset in Fig. 9.

release_date	id	name	popularity	duration_ms	explicit	artists	id_artists	danceability	energy	key
1922-02-22	35iwgR4jXetI318WEWsa1Q	Carve	6	126903	0	['Uli']	['45tIt06XoI0Iio4LBEVpls']	0.645	0.4450	0
1922-06-01	021ht4sdgPcrDgSk7JTbKY	Capítulo 2.16 - Banquero Anarquista	0	98200	0	['Fernando Pessoa']	['14jtPCOoNZwquk5wd9DxrY']	0.695	0.2630	0
1922-03-21	07A5yehtSnoedViJAZkNnc	Vivo para Quererte - Remastrizado	0	181640	0	['Ignacio Corsini']	['5LiOoJbxVSAMkBS2fUm3X2']	0.434	0.1770	1
1922-03-21	08FmqUhxtyLTn6pAh6bk45	El Prisionero - Remastrizado	0	178907	0	['Ignacio Corsini']	['5LiOoJbxVSAMkBS2fUm3X2']	0.321	0.0946	7
1922	08y9GfoqCWfOGsKdwojr5e	Lady of the Evening	0	163080	0	['Dick Haymes']	['3BiJGZsyX9sJchTqcSA7Su']	0.402	0.1580	3

Fig. 9. Shows the release date as the new index.

(C) To obtain the name of the artist present in the 18th row of the dataset.
We can utilize the "iloc[]" method. This method allows you to filter and retrieve specific information from the dataset based on its index location. By specifying the index location as 18, you can extract the artist's name from the corresponding row.

```
Out[9]: artists    ['Victor Boucher']
        Name: 1922-01-01 00:00:00, dtype: object
```

Fig. 10. Shows the Victor Boucher information.

By using the "iloc[]" method and referencing the 18th row in the dataset, we identified that the artist's name associated with that particular row is Victor Boucher shown in Fig. 10.

(D) To convert the duration of songs from milliseconds to seconds.
We can perform the necessary calculation and update the duration column in the dataset. Afterward, you can print the column headers to confirm that the duration has been successfully converted to seconds shown in Fig. 11.

```
Out[11]: release_date
         1922-02-22    127
         1922-01-06     98
         1922-03-21    182
         1922-03-21    177
         1922-01-01    163
         Name: duration, dtype: int64
```

Fig. 11. Shows that songs are present in seconds

(E) Correlation Map.

Let's create a correlation map as our first visualization. To begin, we will remove three unnecessary columns, namely "mode," "explicit," and an unnamed column. We will calculate the Pearson correlation coefficient for the remaining variables. For the correlation map, we will set the figure size to (14,6) and utilize the "heatmap()" function from the seaborn (sns) library. Additionally, we will enable annotations by setting "annotation = True". To format the data values in each cell, we will use "fmt = ".1g"". Lastly, we can choose a color map (cmap) from the seaborn documentation to customize the appearance of the correlation map.

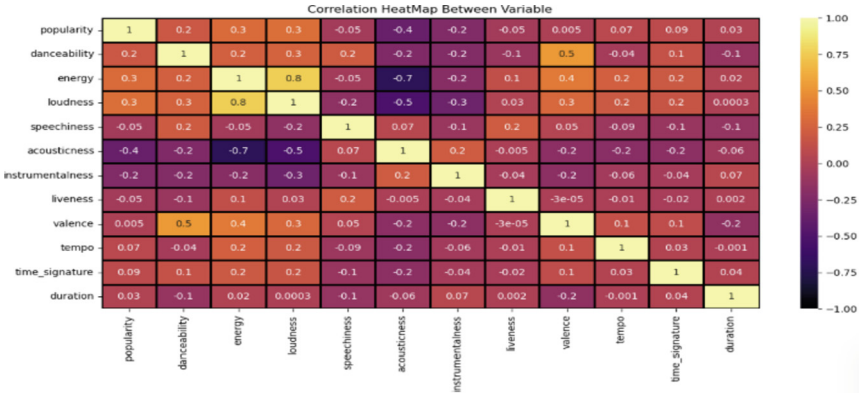

Fig. 12. Shows correlation map.

Upon executing the provided code, the correlation map was generated shown in Fig. 12. The color-coded scale on the right side represents the range from –1 to +1. Negative values near –1 indicate variables with minimal or negative correlation, while positive values greater than 0.0 indicate variables with a positive correlation.

2.4 Model Building

Model building refers to the process of constructing and developing a representation of a system or phenomenon using various techniques and methodologies. It involves creating a simplified version or a conceptual framework that captures the essential characteristics and relationships of the subject under study.

In the context of machine learning and data analysis, model building specifically refers to the construction of mathematical or statistical models that can make predictions, classify data, or uncover patterns and insights from available data. These models are trained on existing data, and their purpose is to generalize and make accurate predictions on new, unseen data.

The process of model building typically involves several steps. First, the problem at hand is defined, and the data relevant to the problem is collected and prepared for analysis. Then, an appropriate modeling technique is selected, considering the nature of the problem and the available data.

Next, the model structure and parameters are defined, and the training data is used to estimate these parameters. This involves using optimization algorithms to find the best fit between the model and the training data, minimizing the error or maximizing the likelihood of the observed data.

Once the model is trained, it is evaluated using validation data to assess its performance and generalization capabilities. This step helps in detecting and addressing potential issues such as overfitting or underfitting, which can occur when the model either memorizes the training data too closely or fails to capture its underlying patterns.

After evaluation, if the model performs well, it can be deployed to make predictions or generate insights on new, unseen data. If the model's performance is not satisfactory, further iterations and refinements may be required, such as adjusting the model's structure, exploring different algorithms, or collecting additional data.

(A) To create a Regression Plot Between Loudness and Energy.
We have utilize the "regplot()" function from the seaborn library. This function enables us to generate a scatter plot with a regression line representing the relationship between the two variables.

The regression plot has been generated shown in Fig. 13, illustrating a significant positive correlation between "Loudness" and "Energy." It is evident from the plot that all the data points or songs are oriented in the same direction. When the energy of a song increases, its loudness also tends to increase. Conversely, if the loudness decreases, the energy of the track also decreases.

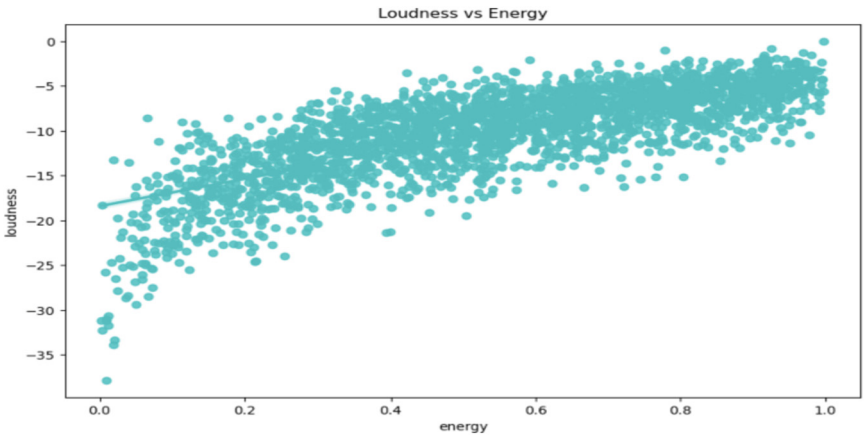

Fig. 13. Shows the regression plot between loudness and energy.

(B) To create the relationship between "Popularity" and "Acousticness".
We can generate a regression plot that displays a regression line. This plot will provide insights into the correlation between the two variables.

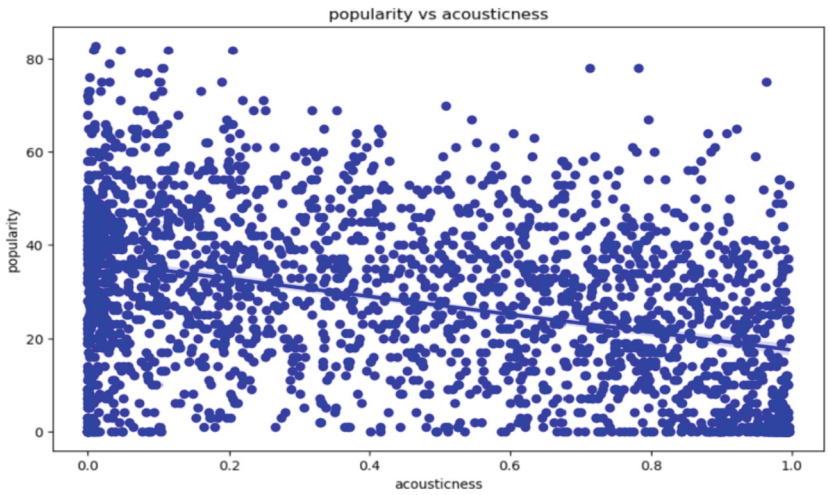

Fig. 14. Shows the relationship between Popularity and Acousticness.

In the regression plot, the downward-sloping blue regression line indicates an inverse relationship between "Acousticness" and "Popularity" shown in Fig. 14. This means that as the acousticness of a song increases, its popularity tends to decrease. Conversely, if the popularity of a song increases, the acousticness tends to decrease.

(C) To visualize the duration of songs for each year.
We can employ the seaborn library and utilize the "lineplot()" function. This line graph will provide a visual representation of how the song durations have varied over different years.

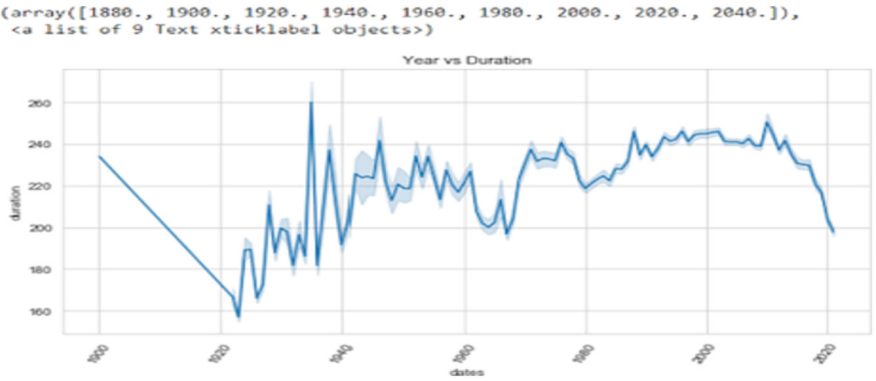

Fig. 15. Shows the duration of songs for each year.

After generating the line plot shown in Fig. 15, we can observe the duration of songs over time. The X-axis represents the years, while the Y-axis represents the duration of songs. Notably, songs from the 1920s to the 1960s were generally shorter in duration. Subsequently, there was a steady increase in song duration until around 2010. However, from 2010 onwards, there was a decline in song duration once again.

3 Simulation Results

These simulations involve creating mathematical or statistical models that capture the key dynamics and variables at play within the Spotify ecosystem. These variables can include user preferences, listening habits, music attributes, social interactions, and more. By incorporating these factors into the model, researchers and data scientists can simulate and study different scenarios to gain insights into how the platform operates and how users interact with it.

(A) To visualize the duration of songs with respect to different genres.
We have utilized the seaborn library and employ the "barplot()" function. This function allows you to create a horizontal bar plot, where each genre is represented on the y-axis, and the duration of the songs is depicted on the x-axis.

The horizontal bar plot displays the genres on the Y-axis and the song durations in milliseconds on the X-axis. Upon analyzing the data shown in Fig. 15, we can observe that the classical and world genres tend to have longer song durations, while children's music exhibits shorter song durations. Find top five genres by Popularity and pot a bar plot for the same.

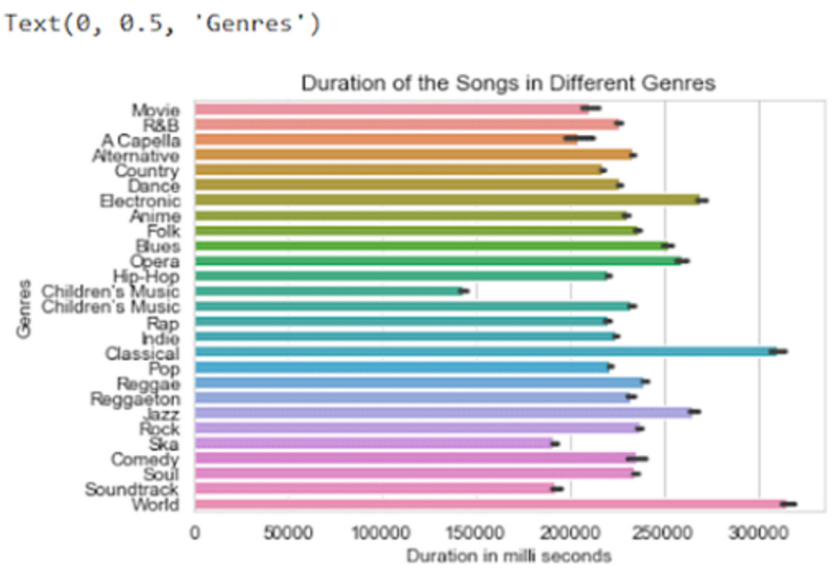

Fig. 16. Shows the duration of the songs in different genres.

(B) Units To find top five genres by Popularity and pot a barplot for the same.

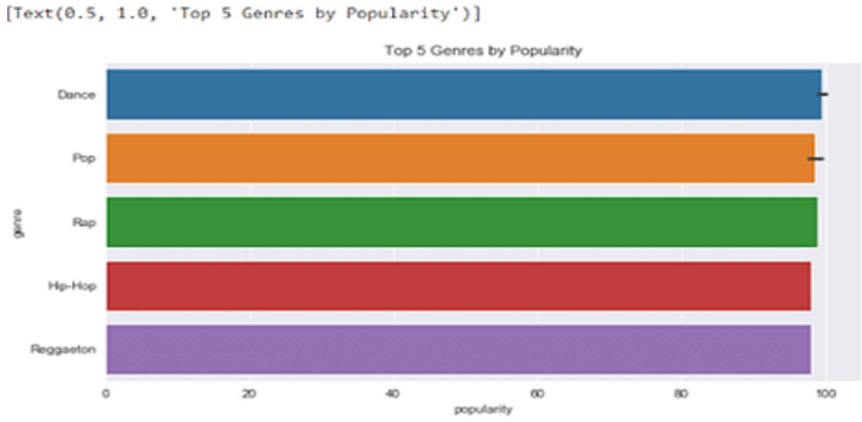

Fig. 17. Shows the top five genres.

According to the level of popularity, we have determined the most popular music genres to be Dance, Pop, Rap, Hip-Hop, and Reggaeton, as depicted in Fig. 16. This indicates that these genres have garnered significant attention and a large fan base among listeners. The term "popularity" here refers to the measure of how widely these genres are appreciated and enjoyed by the audience. It takes into account factors such as radio airplay, streaming numbers, sales, concert attendance, and overall cultural impact. Based on

these indicators, the mentioned genres have emerged as the most prominent and influential in the current music landscape. Figure 17 visually represents this information, possibly through a graph or chart, highlighting the relative positions or proportions of each genre in terms of their popularity. It provides a clear visual representation of the data, allowing viewers to quickly grasp the dominance and significance of these specific genres. By identifying these top genres, it becomes easier for industry professionals, music enthusiasts, and researchers to understand the trends and preferences of music listeners. This information can be valuable for various purposes, such as marketing and promotion strategies, radio programming, playlist curation, and even predicting future music trends.

4 Conclusion

Data analysis on Spotify using Python offers valuable insights into user preferences, music trends, and song popularity. Python's libraries enable effective data collection, pre-processing, analysis, and visualization. Data visualization provides intuitive representations, aiding the communication of insights to stakeholders. Integration of Spotify's API or public datasets ensures access to reliable and diverse data sources. Data analysis on Spotify using Python empowers analysts to understand user preferences, identify popular song attributes, and make informed decisions. It enables personalized recommendations, targeted marketing strategies, and improved user experiences. By leveraging data analysis on Spotify, organizations can gain a competitive edge in the music industry. They can tailor their offerings to match user preferences, optimize marketing campaigns, and enhance the overall user experience.

References

1. Lawrence, D.L.: Addressing the value gap in the age of digital music streaming. In: Vand, J., Transnat'l, L., 52, 511. Clerk Maxwell, J. (eds.) A Treatise on Electricity and Magnetism, 3rd edn., vol. 2, pp. 68–73. Oxford, Clarendon, 1892 (2019)
2. Sciandra, M., Spera, I.C.: A model-based approach to Spotify data analysis: a Beta GLMM. J. Appl. Stat. **49**(1), 214–229 (2022)
3. Pérez-Verdejo, J.M., Piña-García, C.A., Ojeda, M.M., Rivera-Lara, A., Méndez-Morales, L.: The rhythm of Mexico: an exploratory data analysis of Spotify's top 50. J. Comput. Soc. Sci. **4**, 147–161 (2021)
4. Budzinski, O., Gaenssle, S., Lindstädt-Dreusicke, N.: Data (r) evolution: the economics of algorithmic search and recommender services. In: Handbook on Digital Business Ecosystems, pp. 349–366. Edward Elgar Publishing (2022)
5. Skog, D., Wimelius, H., Sandberg, J.: Digital service platform evolution: how Spotify leveraged boundary resources to become a global leader in music streaming (2018)
6. Hujran, O., Alikaj, A., Durrani, U.K., Al-Dmour, N.: Big data and its effect on the music industry. In: Proceedings of the 3rd International Conference on Software Engineering and Information Management, pp. 5–9, January 2020
7. Schulz, W.L., Durant, T.J., Siddon, A.J., Torres, R.: Use of application containers and workflows for genomic data analysis. J. Pathol. Inform. **7**(1), 53 (2016)

8. Salameh, A., Bass, J.: Influential factors of aligning Spotify squads in mission-critical and offshore research papers–a longitudinal embedded case study. In Product-Focused Software Process Improvement: 19th International Conference, PROFES 2018, Wolfsburg, Germany, 28–30 November 2018, Proceedings 19, pp. 199–215. Springer International Publishing (2018)
9. Lin, Y.C., Tsai, H.N., Lee, Y.C.: The effects of product categories, brand alliance fitness and personality traits on customer's brand attitude and purchase intentions: a case of Spotify. J. Stat. Manag. Syst. **23**(3), 677–693 (2020)
10. Mobasher, B., Dettori, L., Raicu, D., Settimi, R., Sonboli, N., Stettler, M.: Data science summer academy for chicago public school students. ACM SIGKDD Explor. Newsl **21**(1), 49–52 (2019)
11. Caveness, E., GC, P.S., Peng, Z., Polyzotis, N., Roy, S., Zinkevich, M.: Tensorflow data validation: data analysis and validation in continuous ml pipelines. In: Proceedings of the 2020 ACM SIGMOD International Conference on Management of Data, pp. 2793–2796, June 2020
12. Chen, C.C., Leon, S., Nakayama, M.: Are you hooked on paid music streaming?: an investigation into the millennial generation. Int. J. E-Business Res. (IJEBR) **14**(1), 1–20 (2018)
13. Fajana, O., Owenson, G., Cocea, M.: Torbot stalker: detecting tor botnets through intelligent circuit data analysis. In: 2018 IEEE 17th International Symposium on Network Computing and Applications (NCA), pp. 1–8. IEEE, November 2018
14. Pedrero-Esteban, L.M., Barrios-Rubio, A., Medina-Ávila, V.: Teenagers, smartphones and digital audio consumption in the age of Spotify. Comunicar. Media Edu. Res. J. **27**(2) (2019)
15. Isson, J.P.: Unstructured Data Analysis: How to Improve Customer Acquisition, Customer Retention, and Fraud Detection and Prevention. John Wiley & Sons (2018)
16. Peters, K., et al.: PhenoMeNal: processing and analysis of metabolomics data in the cloud. Gigascience **8**(2), giy149 (2019)
17. Belcastro, L., Marozzo, F., Talia, D.: Programming models and systems for big data analysis. Int. J. Parallel Emergent Distrib. Syst. **34**(6), 632–652 (2019)
18. Trabucchi, D., Buganza, T., Dell'Era, C., Pellizzoni, E.: Exploring the inbound and outbound strategies enabled by user generated big data: evidence from leading smartphone applications. Creativity Innov. Manag. **27**(1), 42–55 (2018)
19. Poldrack, R.A., Gorgolewski, K.J., Varoquaux, G.: Computational and informatic advances for reproducible data analysis in neuroimaging. Ann. Rev. Biomed. Data Sci. **2**, 119–138 (2019)
20. Ochi, V., Estrada, R., Gaji, T., Gadea, W., Duong, E.: Spotify danceability and popularity analysis using sap. arXiv preprint arXiv:2108.02370 (2021)
21. Smite, D., Moe, N.B., Floryan, M., Levinta, G., Chatzipetrou, P: Spotify guilds. Commun. ACM **63**(3), 56–61 (2020)
22. Ramos, E.F., Blind, K.: Data portability effects on data-driven innovation of online platforms: analyzing Spotify. Telecommun. Policy **44**(9), 102026 (2020)
23. Kumar, I., Mishra, M.K., Mishra, R.K.: Performance analysis of NOMA downlink for next - generation 5G network with statistical channel state information. Ingénierie des Systèmes d'Information, **26**(4), 417–423 (2021). https://doi.org/10.18280/isi.260410
24. Shankar, R., Kumar, I., Mishra, R.K.: Pairwise error probability analysis of dual hop relaying network over time selective Nakagami-m fading channel with imperfect CSI and node mobility. Traitement du Signal **36**(3), 281–295 (2019). . https://doi.org/10.18280/ts.360312
25. Kumar, I., Kumar, A., Kumar Mishra, R.: Performance analysis of cooperative NOMA system for defense application with relay selection in a hostile environment. J. Def. Model. Simul. (2022). https://doi.org/10.1177/15485129221079721

26. Ashish, I.K., Mishra, R.K.: Performance analysis for wireless non-orthogonal multiple access downlink systems. In: 2020 International Conference on Emerging Frontiers in Electrical and Electronic Technologies (ICEFEET), Patna, India, pp. 1–6 (2020). https://doi.org/10.1109/ICEFEET49149.2020.9186987
27. Kumar, I., Mishra, R.K.: An investigation of spectral efficiency in linear MRC and MMSE detectors with perfect and imperfect CSI for massive MIMO systems. Traitement du Signal **38**(2), 495–501 (2021). https://doi.org/10.18280/ts.380229
28. Kumar, I., Mishra, R.K.: An efficient ICI mitigation technique for MIMO-OFDM system in time-varying channels. Math. Model. Eng. Probl. **7**(1), 79–86 (2020). https://doi.org/10.18280/mmep.070110
29. Jacobson, K., Murali, V., Newett, E., Whitman, B., Yon, R.: Music personalization at Spotify. In: Proceedings of the 10th ACM Conference on Recommender Systems, p. 373, September 2016
30. Harris, M., et al.: Analyzing the Spotify Top 200 Through a Point Process Lens. arXiv preprint arXiv:1910.01445 (2019)
31. Duman, D., Neto, P., Mavrolampados, A., Toiviainen, P., Luck, G.: Music we move to: Spotify audio features and reasons for listening. PLoS ONE **17**(9), e0275228 (2022)
32. Gupta, N., Kumar, I., Rathod, I., Sharma B**,** S.S.P.M.: Sustainable production systems with AI and emerging technologies: a moderator-mediation analysis. **12**, Special Issue 8, 2819–2832 (2023). https://doi.org/10.48047/ecb/2023.12.si8.200
33. Lozic, J.: Comparison of business models of the streaming platforms Spotify and Netflix. Economic and Social Development: Book of Proceedings, pp. 110–119 (2020)
34. South, T.: Network analysis of the Spotify artist collaboration graph. Aust. Math. Sci. Inst. 1–12 (2018)
35. Salameh, A., Bass, J.M.: An architecture governance approach for Agile development by tailoring the Spotify model. AI & Soc. **37**(2), 761–780 (2022)
36. Kim, J.: Music popularity prediction through data analysis of music's characteristics. Int. J. Sci. Technol. Soc. **9**(5), 239 (2021)
37. Sharma B, S.S.P.M., Ravishankar Kamath, H., Siva Brahmaiah Rama, V.: Modelling of cloud based online access system for solar charge controller. Int. J. Eng. Technol. **7**(2.21), 58–61 (2018)
38. Shalinee Gupta, Ms., Sharma B, S.S.P.M.: Design and development of an intelligent aqua monitoring system using cloud based online access control systems. Int. J. Rec. Technol. Eng. (IJRTE), **8**(4), November 2019. ISSN: 2277-3878
39. Dr. Ravishankar Kamath, H., Sharma B, S.S.P.M., Siva Brahmaiah Rama, V.: PWM based solar charge controller using IoT. Int. J. Eng. Technol. **7**(2.7), 284–288 (2018)
40. Dr. Ravishankar Kamath, H., Siva Brahmaiah Rama, V., Sharma B, S.S.P.M.: Street light monitoring using IOT. Int. J. Eng. Technol. **7**(2.7), 1008–1012 (2018)
41. Sharma B, S.S.P.M., Kumar, A., Meena, B.K.: An intelligent solar based farm monitoring using cloud based online access control systems. Int. J. Rec. Technol. Eng. (IJRTE), **8**(3), September 2019. ISSN: 2277-3878