

Spotify.Description.for.Students

November 4, 2024

1 Project 4: Music Popularity Prediction

This project will take data features collected for songs that have been on the Top 200 Weekly (Global) charts of Spotify in 2020 & 2021. The popularity of the song will be predicted using a tree-based regression model trained on these features.

The goals for the project are:

- Minimize the cross-validated *root mean squared error* (*RMSE*) when predicting the popularity of a new song.
- Determine the importance of the features in driving the regression result. The project will be done using tree-based regression techniques as covered in class. The parameters of the trees should be carefully selected to avoid over-fitting.

There are three main challenges for this project:

1. Determining the outcome (i.e. target). There is a “popularity” column. But other columns may or may not be more appropriate indicators of popularity.
2. Choosing appropriate predictors (i.e. features). When building a machine learning model, we want to make sure that we consider how the model will be ultimately used. For this project, we are predicting the popularity of a new song. Therefore, we should only include the predictors we would have for a new song.
3. Data cleaning and feature engineering. Some creative cleaning and/or feature engineering may be needed to extract useful information for prediction.

Once again, be sure to go through the whole data science process and document as such in your Jupyter notebook.

The data is available AWS at <https://ddc-datascience.s3.amazonaws.com/Projects/Project.4-Spotify/Data/Spotify.csv> .

```
[246]: # !pip install cudf-cu11 --extra-index-url=https://pypi.ngc.nvidia.com
# !apt-get update
# !apt-get install -y nvidia-cuda-toolkit

# import locale
# locale.getpreferredencoding = lambda: "UTF-8"

# %load_ext cudf.pandas
```

2 Imports

```
[247]: import sys
       print(sys.executable)
```

/usr/local/bin/python

```
[248]: import pandas as pd
       import numpy as np
       import matplotlib.pyplot as plt
       import seaborn as sns

       from sklearn.preprocessing import StandardScaler
       from sklearn.preprocessing import MinMaxScaler
       from sklearn.model_selection import train_test_split
       from sklearn.linear_model import LinearRegression
       from sklearn.tree import DecisionTreeRegressor
       from sklearn.ensemble import RandomForestRegressor
       import xgboost as xgb

       from sklearn.metrics import mean_squared_error, root_mean_squared_error, r2_score
```

```
[249]: import pandas as pd
       import numpy as np
       import matplotlib.pyplot as plt
       import seaborn as sns

       from sklearn.preprocessing import StandardScaler
       from sklearn.preprocessing import MinMaxScaler
       from sklearn.model_selection import train_test_split
       from sklearn.linear_model import LinearRegression
       from sklearn.tree import DecisionTreeRegressor
       from sklearn.ensemble import RandomForestRegressor
       import xgboost as xgb

       from sklearn.metrics import mean_squared_error, root_mean_squared_error, r2_score
```

```
[250]: %%capture
       url = "https://ddc-datascience.s3.amazonaws.com/Projects/Project.4-Spotify/Data/
           ↳ Spotify.csv"
       !curl -s -I {url}
```

3 Data Exploration

```
[251]: df_1 = pd.read_csv(url).copy()
```

3.1 Head

```
[252]: df_1.head()
```

```
[252]:
```

	Index	Highest Charting Position	Number of Times Charted	\
0	1	1	8	
1	2	2	3	
2	3	1	11	
3	4	3	5	
4	5	5	1	

	Week of Highest Charting	Song Name	Streams	\
0	2021-07-23--2021-07-30	Beggin'	48,633,449	
1	2021-07-23--2021-07-30	STAY (with Justin Bieber)	47,248,719	
2	2021-06-25--2021-07-02	good 4 u	40,162,559	
3	2021-07-02--2021-07-09	Bad Habits	37,799,456	
4	2021-07-23--2021-07-30	INDUSTRY BABY (feat. Jack Harlow)	33,948,454	

	Artist	Artist Followers	Song ID	\
0	Måneskin	3377762	3Wrjm47oTz2sjIgck11l5e	
1	The Kid LAROI	2230022	5HCyWlXZPP0y6Gqq8TgA20	
2	Olivia Rodrigo	6266514	4ZtFanR9U6ndgddUvNcjcG	
3	Ed Sheeran	83293380	6PQ88X9TkUIAUIZJHW2upE	
4	Lil Nas X	5473565	27NovPIUIRr0ZoCHxABJwK	

	Genre	...	Danceability	Energy	Loudness	\
0	['indie rock italiano', 'italian pop']	...	0.714	0.8	-4.808	
1	['australian hip hop']	...	0.591	0.764	-5.484	
2	['pop']	...	0.563	0.664	-5.044	
3	['pop', 'uk pop']	...	0.808	0.897	-3.712	
4	['lgbtq+ hip hop', 'pop rap']	...	0.736	0.704	-7.409	

	Speechiness	Acousticness	Liveness	Tempo	Duration (ms)	Valence	Chord
0	0.0504	0.127	0.359	134.002	211560	0.589	B
1	0.0483	0.0383	0.103	169.928	141806	0.478	C#/Db
2	0.154	0.335	0.0849	166.928	178147	0.688	A
3	0.0348	0.0469	0.364	126.026	231041	0.591	B
4	0.0615	0.0203	0.0501	149.995	212000	0.894	D#/Eb

[5 rows x 23 columns]

3.2 Tail

```
[253]: df_1.tail()
```

```
[253]:
```

	Index	Highest Charting Position	Number of Times Charted	\
1551	1552	195	1	
1552	1553	196	1	

1553	1554	197	1
1554	1555	198	1
1555	1556	199	1

	Week of Highest Charting	Song Name	Streams \
1551	2019-12-27--2020-01-03	New Rules	4,630,675
1552	2019-12-27--2020-01-03	Cheirosa - Ao Vivo	4,623,030
1553	2019-12-27--2020-01-03	Havana (feat. Young Thug)	4,620,876
1554	2019-12-27--2020-01-03	Surtada - Remix Brega Funk	4,607,385
1555	2019-12-27--2020-01-03	Lover (Remix) [feat. Shawn Mendes]	4,595,450

	Artist	Artist Followers	Song ID \
1551	Dua Lipa	27167675	2ekn2ttSfGqwhhate0LSR0
1552	Jorge & Mateus	15019109	2PWjKmJyTZedpmOUa3a5da
1553	Camila Cabello	22698747	1rfofaqEpACxVEHIZBJe6W
1554	Dadá Boladão, Tati Zaqui, OIK	208630	5F8ffc8KWKNawllr5WsW0r
1555	Taylor Swift	42227614	3i9UVldZOE0aD0JnyfAZZ0

	Genre	... Danceability \
1551	['dance pop', 'pop', 'uk pop']	... 0.762
1552	['sertanejo', 'sertanejo universitario']	... 0.528
1553	['dance pop', 'electropop', 'pop', 'post-teen 0.765
1554	['brega funk', 'funk carioca']	... 0.832
1555	['pop', 'post-teen pop']	... 0.448

	Energy	Loudness	Speechiness	Acousticness	Liveness	Tempo	Duration (ms) \
1551	0.7	-6.021	0.0694	0.00261	0.153	116.073	209320
1552	0.87	-3.123	0.0851	0.24	0.333	152.37	181930
1553	0.523	-4.333	0.03	0.184	0.132	104.988	217307
1554	0.55	-7.026	0.0587	0.249	0.182	154.064	152784
1555	0.603	-7.176	0.064	0.433	0.0862	205.272	221307

	Valence	Chord
1551	0.608	A
1552	0.714	B
1553	0.394	D
1554	0.881	F
1555	0.422	G

[5 rows x 23 columns]

3.3 Shape

```
[254]: df_1.shape
```

```
[254]: (1556, 23)
```

3.4 columns

```
[255]: df_1.columns
```

```
[255]: Index(['Index', 'Highest Charting Position', 'Number of Times Charted',  
        'Week of Highest Charting', 'Song Name', 'Streams', 'Artist',  
        'Artist Followers', 'Song ID', 'Genre', 'Release Date', 'Weeks Charted',  
        'Popularity', 'Danceability', 'Energy', 'Loudness', 'Speechiness',  
        'Acousticness', 'Liveness', 'Tempo', 'Duration (ms)', 'Valence',  
        'Chord'],  
        dtype='object')
```

3.5 Dtypes

```
[256]: df_1.dtypes
```

```
[256]: Index                int64  
Highest Charting Position    int64  
Number of Times Charted      int64  
Week of Highest Charting     object  
Song Name                   object  
Streams                     object  
Artist                     object  
Artist Followers            object  
Song ID                    object  
Genre                      object  
Release Date               object  
Weeks Charted              object  
Popularity                 object  
Danceability               object  
Energy                    object  
Loudness                   object  
Speechiness                object  
Acousticness               object  
Liveness                   object  
Tempo                     object  
Duration (ms)              object  
Valence                    object  
Chord                     object  
dtype: object
```

3.6 Describe

```
[257]: df_1.describe()
```

```
[257]:
```

	Index	Highest Charting Position	Number of Times Charted
count	1556.000000	1556.000000	1556.000000
mean	778.500000	87.744216	10.668380

std	449.322824	58.147225	16.360546
min	1.000000	1.000000	1.000000
25%	389.750000	37.000000	1.000000
50%	778.500000	80.000000	4.000000
75%	1167.250000	137.000000	12.000000
max	1556.000000	200.000000	142.000000

3.7 Isnull Sum

```
[258]: df_1.isnull().sum()
```

```
[258]: Index                                0
Highest Charting Position                 0
Number of Times Charted                   0
Week of Highest Charting                  0
Song Name                                0
Streams                                   0
Artist                                    0
Artist Followers                          0
Song ID                                   0
Genre                                     0
Release Date                             0
Weeks Charted                             0
Popularity                                0
Danceability                              0
Energy                                    0
Loudness                                  0
Speechiness                              0
Acousticness                             0
Liveness                                  0
Tempo                                     0
Duration (ms)                             0
Valence                                   0
Chord                                     0
dtype: int64
```

3.8 unique values

```
[259]: df_1.count('rows').unique().sum()
```

```
[259]: np.int64(1556)
```

```
[260]: df_1.count('columns')
```

```
[260]: 0      23
1      23
2      23
```

```

3      23
4      23
..
1551   23
1552   23
1553   23
1554   23
1555   23
Length: 1556, dtype: int64

```

3.9 Sort_values

```
[261]: df_1.sort_values(by = ['Popularity'], ascending = False).head(10)
```

```
[261]:
```

	Index	Highest Charting Position	Number of Times Charted \
1	2	2	3
2	3	1	11
3	4	3	5
5	6	1	18
4	5	5	1
8	9	3	8
14	15	2	10
7	8	2	10
9	10	8	10
11	12	9	9

	Week of Highest Charting	Song Name	Streams \
1	2021-07-23--2021-07-30	STAY (with Justin Bieber)	47,248,719
2	2021-06-25--2021-07-02	good 4 u	40,162,559
3	2021-07-02--2021-07-09	Bad Habits	37,799,456
5	2021-05-07--2021-05-14	MONTERO (Call Me By Your Name)	30,071,134
4	2021-07-23--2021-07-30	INDUSTRY BABY (feat. Jack Harlow)	33,948,454
8	2021-06-18--2021-06-25	Yonaguni	25,030,128
14	2021-05-21--2021-05-28	Butter	19,985,713
7	2021-06-18--2021-06-25	Todo De Ti	26,951,613
9	2021-07-02--2021-07-09	I WANNA BE YOUR SLAVE	24,551,591
11	2021-07-02--2021-07-09	Qué Más Pues?	22,405,111

	Artist	Artist Followers	Song ID \
1	The Kid LAROI	2230022	5HCyWlXZPP0y6Gqq8TgA20
2	Olivia Rodrigo	6266514	4ZtFanR9U6ndgddUvNcjcG
3	Ed Sheeran	83293380	6PQ88X9TkUIAUIZJHW2upE
5	Lil Nas X	5473565	67BtfxlNbhbMCDR2L2l8qd
4	Lil Nas X	5473565	27NovPIUIRr0Z0CHxABJwK
8	Bad Bunny	36142273	2JPLbjOn0wPCngEot2STUS
14	BTS	37106176	2bgTY4UwhfBYhGT4HUYStN
7	Rauw Alejandro	6080597	4fSIb4hd0Q151TILNsSEaF

9	Måneskin	3377762	4pt5fDVTg5GhEvEtlz9dKk
11	J Balvin, Maria Becerra	29051363	6hf0RpxTb0prT5nnwzkk8e

	Genre	...	Danceability	Energy	\
1	['australian hip hop']	...	0.591	0.764	
2	['pop']	...	0.563	0.664	
3	['pop', 'uk pop']	...	0.808	0.897	
5	['lgbtq+ hip hop', 'pop rap']	...	0.61	0.508	
4	['lgbtq+ hip hop', 'pop rap']	...	0.736	0.704	
8	['latin', 'reggaeton', 'trap latino']	...	0.644	0.648	
14	['k-pop', 'k-pop boy group']	...	0.759	0.459	
7	['puerto rican pop', 'trap latino']	...	0.78	0.718	
9	['indie rock italiano', 'italian pop']	...	0.75	0.608	
11	['latin', 'reggaeton', 'reggaeton colombiano']	...	0.891	0.819	

	Loudness	Speechiness	Acousticness	Liveness	Tempo	Duration (ms)	Valence	\
1	-5.484	0.0483	0.0383	0.103	169.928	141806	0.478	
2	-5.044	0.154	0.335	0.0849	166.928	178147	0.688	
3	-3.712	0.0348	0.0469	0.364	126.026	231041	0.591	
5	-6.682	0.152	0.297	0.384	178.818	137876	0.758	
4	-7.409	0.0615	0.0203	0.0501	149.995	212000	0.894	
8	-4.601	0.118	0.276	0.135	179.951	206710	0.44	
14	-5.187	0.0948	0.00323	0.0906	109.997	164442	0.695	
7	-3.605	0.0506	0.31	0.0932	127.949	199604	0.342	
9	-4.008	0.0387	0.00165	0.178	132.507	173347	0.958	
11	-3.964	0.106	0.0261	0.173	101.968	217773	0.768	

	Chord
1	C#/Db
2	A
3	B
5	G#/Ab
4	D#/Eb
8	C#/Db
14	G#/Ab
7	D#/Eb
9	C#/Db
11	G#/Ab

[10 rows x 23 columns]

4 Data Cleaning and Feature Engineering

4.1 New copy of dataframe

```
[262]: df_cleaning = df_1.copy()
df_cleaning
```

```
[262]:
```

	Index	Highest Charting Position	Number of Times Charted	\
0	1	1	8	
1	2	2	3	
2	3	1	11	
3	4	3	5	
4	5	5	1	
...	
1551	1552	195	1	
1552	1553	196	1	
1553	1554	197	1	
1554	1555	198	1	
1555	1556	199	1	

	Week of Highest Charting	Song Name	Streams	\
0	2021-07-23--2021-07-30	Beggin'	48,633,449	
1	2021-07-23--2021-07-30	STAY (with Justin Bieber)	47,248,719	
2	2021-06-25--2021-07-02	good 4 u	40,162,559	
3	2021-07-02--2021-07-09	Bad Habits	37,799,456	
4	2021-07-23--2021-07-30	INDUSTRY BABY (feat. Jack Harlow)	33,948,454	
...	
1551	2019-12-27--2020-01-03	New Rules	4,630,675	
1552	2019-12-27--2020-01-03	Cheirosa - Ao Vivo	4,623,030	
1553	2019-12-27--2020-01-03	Havana (feat. Young Thug)	4,620,876	
1554	2019-12-27--2020-01-03	Surtada - Remix Brega Funk	4,607,385	
1555	2019-12-27--2020-01-03	Lover (Remix) [feat. Shawn Mendes]	4,595,450	

	Artist	Artist Followers	Song ID	\
0	Måneskin	3377762	3Wrjm47oTz2sjIgck1115e	
1	The Kid LAROI	2230022	5HCyWlXZPP0y6Gqq8TgA20	
2	Olivia Rodrigo	6266514	4ZtFanR9U6ndgddUvNcjcG	
3	Ed Sheeran	83293380	6PQ88X9TkUIAUIZJHW2upE	
4	Lil Nas X	5473565	27NovPIUIRrOZoCHxABJwK	
...	
1551	Dua Lipa	27167675	2ekn2ttSfGqwhhate0LSR0	
1552	Jorge & Mateus	15019109	2PWjKmjyTZedpmOUa3a5da	
1553	Camila Cabello	22698747	1rfofaqEpACxVEHIZBJe6W	
1554	Dadá Boladão, Tati Zaqui, OIK	208630	5F8fffc8KWKNawllr5WsW0r	
1555	Taylor Swift	42227614	3i9UVldZOE0aD0JnyfAZZ0	

	Genre	...	Danceability	\
0	['indie rock italiano', 'italian pop']	...	0.714	

1	['australian hip hop']	...	0.591
2	['pop']	...	0.563
3	['pop', 'uk pop']	...	0.808
4	['lgbtq+ hip hop', 'pop rap']	...	0.736
...
1551	['dance pop', 'pop', 'uk pop']	...	0.762
1552	['sertanejo', 'sertanejo universitario']	...	0.528
1553	['dance pop', 'electropop', 'pop', 'post-teen	0.765
1554	['brega funk', 'funk carioca']	...	0.832
1555	['pop', 'post-teen pop']	...	0.448

	Energy	Loudness	Speechiness	Acousticness	Liveness	Tempo	Duration (ms)	\
0	0.8	-4.808	0.0504	0.127	0.359	134.002	211560	
1	0.764	-5.484	0.0483	0.0383	0.103	169.928	141806	
2	0.664	-5.044	0.154	0.335	0.0849	166.928	178147	
3	0.897	-3.712	0.0348	0.0469	0.364	126.026	231041	
4	0.704	-7.409	0.0615	0.0203	0.0501	149.995	212000	
...	
1551	0.7	-6.021	0.0694	0.00261	0.153	116.073	209320	
1552	0.87	-3.123	0.0851	0.24	0.333	152.37	181930	
1553	0.523	-4.333	0.03	0.184	0.132	104.988	217307	
1554	0.55	-7.026	0.0587	0.249	0.182	154.064	152784	
1555	0.603	-7.176	0.064	0.433	0.0862	205.272	221307	

	Valence	Chord
0	0.589	B
1	0.478	C#/Db
2	0.688	A
3	0.591	B
4	0.894	D#/Eb
...
1551	0.608	A
1552	0.714	B
1553	0.394	D
1554	0.881	F
1555	0.422	G

[1556 rows x 23 columns]

4.2 drop Index

```
[263]: df_cleaning.drop('Index', axis = 1, inplace = True)
```

```
[264]: df_cleaning.transpose()
```

```
[264]:
```

	0	\
Highest Charting Position	1	

Number of Times Charted	8
Week of Highest Charting	2021-07-23--2021-07-30
Song Name	Beggin'
Streams	48,633,449
Artist	Måneskin
Artist Followers	3377762
Song ID	3Wrjm47oTz2sjIgck1115e
Genre	['indie rock italiano', 'italian pop']
Release Date	2017-12-08
Weeks Charted	2021-07-23--2021-07-30\n2021-07-16--2021-07-23...
Popularity	100
Danceability	0.714
Energy	0.8
Loudness	-4.808
Speechiness	0.0504
Acousticness	0.127
Liveness	0.359
Tempo	134.002
Duration (ms)	211560
Valence	0.589
Chord	B

	1	\
Highest Charting Position	2	
Number of Times Charted	3	
Week of Highest Charting	2021-07-23--2021-07-30	
Song Name	STAY (with Justin Bieber)	
Streams	47,248,719	
Artist	The Kid LAROI	
Artist Followers	2230022	
Song ID	5HCyWlXZPP0y6Gqq8TgA20	
Genre	['australian hip hop']	
Release Date	2021-07-09	
Weeks Charted	2021-07-23--2021-07-30\n2021-07-16--2021-07-23...	
Popularity	99	
Danceability	0.591	
Energy	0.764	
Loudness	-5.484	
Speechiness	0.0483	
Acousticness	0.0383	
Liveness	0.103	
Tempo	169.928	
Duration (ms)	141806	
Valence	0.478	
Chord	C#/Db	

2 \

Highest Charting Position	1
Number of Times Charted	11
Week of Highest Charting	2021-06-25--2021-07-02
Song Name	good 4 u
Streams	40,162,559
Artist	Olivia Rodrigo
Artist Followers	6266514
Song ID	4ZtFanR9U6ndgddUvNcjcG
Genre	['pop']
Release Date	2021-05-21
Weeks Charted	2021-07-23--2021-07-30\n2021-07-16--2021-07-23...
Popularity	99
Danceability	0.563
Energy	0.664
Loudness	-5.044
Speechiness	0.154
Acousticness	0.335
Liveness	0.0849
Tempo	166.928
Duration (ms)	178147
Valence	0.688
Chord	A

3 \

Highest Charting Position	3
Number of Times Charted	5
Week of Highest Charting	2021-07-02--2021-07-09
Song Name	Bad Habits
Streams	37,799,456
Artist	Ed Sheeran
Artist Followers	83293380
Song ID	6PQ88X9TkUIAUIZJHW2upE
Genre	['pop', 'uk pop']
Release Date	2021-06-25
Weeks Charted	2021-07-23--2021-07-30\n2021-07-16--2021-07-23...
Popularity	98
Danceability	0.808
Energy	0.897
Loudness	-3.712
Speechiness	0.0348
Acousticness	0.0469
Liveness	0.364
Tempo	126.026
Duration (ms)	231041
Valence	0.591
Chord	B

	4	\
Highest Charting Position	5	
Number of Times Charted	1	
Week of Highest Charting	2021-07-23--2021-07-30	
Song Name	INDUSTRY BABY (feat. Jack Harlow)	
Streams	33,948,454	
Artist	Lil Nas X	
Artist Followers	5473565	
Song ID	27NovPIUIRr0ZoCHxABJwK	
Genre	['lgbtq+ hip hop', 'pop rap']	
Release Date	2021-07-23	
Weeks Charted	2021-07-23--2021-07-30	
Popularity	96	
Danceability	0.736	
Energy	0.704	
Loudness	-7.409	
Speechiness	0.0615	
Acousticness	0.0203	
Liveness	0.0501	
Tempo	149.995	
Duration (ms)	212000	
Valence	0.894	
Chord	D#/Eb	

	5	\
Highest Charting Position	1	
Number of Times Charted	18	
Week of Highest Charting	2021-05-07--2021-05-14	
Song Name	MONTERO (Call Me By Your Name)	
Streams	30,071,134	
Artist	Lil Nas X	
Artist Followers	5473565	
Song ID	67BtfxlNbhBmCDR2L2l8qd	
Genre	['lgbtq+ hip hop', 'pop rap']	
Release Date	2021-03-31	
Weeks Charted	2021-07-23--2021-07-30\n2021-07-16--2021-07-23...	
Popularity	97	
Danceability	0.61	
Energy	0.508	
Loudness	-6.682	
Speechiness	0.152	
Acousticness	0.297	
Liveness	0.384	
Tempo	178.818	
Duration (ms)	137876	
Valence	0.758	
Chord	G#/Ab	

	6	\
Highest Charting Position	3	
Number of Times Charted	16	
Week of Highest Charting	2021-05-14--2021-05-21	
Song Name	Kiss Me More (feat. SZA)	
Streams	29,356,736	
Artist	Doja Cat	
Artist Followers	8640063	
Song ID	748mdHapucXQri7IA08yFK	
Genre	['dance pop', 'pop']	
Release Date	2021-04-09	
Weeks Charted	2021-07-23--2021-07-30\n2021-07-16--2021-07-23...	
Popularity	94	
Danceability	0.762	
Energy	0.701	
Loudness	-3.541	
Speechiness	0.0286	
Acousticness	0.235	
Liveness	0.123	
Tempo	110.968	
Duration (ms)	208867	
Valence	0.742	
Chord	G#/Ab	

	7	\
Highest Charting Position	2	
Number of Times Charted	10	
Week of Highest Charting	2021-06-18--2021-06-25	
Song Name	Todo De Ti	
Streams	26,951,613	
Artist	Rauw Alejandro	
Artist Followers	6080597	
Song ID	4fSIb4hd0Q151TILNsSEaF	
Genre	['puerto rican pop', 'trap latino']	
Release Date	2021-05-20	
Weeks Charted	2021-07-23--2021-07-30\n2021-07-16--2021-07-23...	
Popularity	95	
Danceability	0.78	
Energy	0.718	
Loudness	-3.605	
Speechiness	0.0506	
Acousticness	0.31	
Liveness	0.0932	
Tempo	127.949	
Duration (ms)	199604	
Valence	0.342	

Chord	D#/Eb
	8 \
Highest Charting Position	3
Number of Times Charted	8
Week of Highest Charting	2021-06-18--2021-06-25
Song Name	Yonaguni
Streams	25,030,128
Artist	Bad Bunny
Artist Followers	36142273
Song ID	2JPLbj0n0wPCngEot2STUS
Genre	['latin', 'reggaeton', 'trap latino']
Release Date	2021-06-04
Weeks Charted	2021-07-23--2021-07-30\n2021-07-16--2021-07-23...
Popularity	96
Danceability	0.644
Energy	0.648
Loudness	-4.601
Speechiness	0.118
Acousticness	0.276
Liveness	0.135
Tempo	179.951
Duration (ms)	206710
Valence	0.44
Chord	C#/Db
	9 \
Highest Charting Position	8
Number of Times Charted	10
Week of Highest Charting	2021-07-02--2021-07-09
Song Name	I WANNA BE YOUR SLAVE
Streams	24,551,591
Artist	Måneskin
Artist Followers	3377762
Song ID	4pt5fDVTg5GhEvEtlz9dKk
Genre	['indie rock italiano', 'italian pop']
Release Date	2021-03-19
Weeks Charted	2021-07-23--2021-07-30\n2021-07-16--2021-07-23...
Popularity	95
Danceability	0.75
Energy	0.608
Loudness	-4.008
Speechiness	0.0387
Acousticness	0.00165
Liveness	0.178
Tempo	132.507
Duration (ms)	173347

Valence	0.958
Chord	C#/Db

	...	1546	\
Highest Charting Position	...	143	
Number of Times Charted	...	1	
Week of Highest Charting	...	2019-12-27--2020-01-03	
Song Name	...	JACKBOYS	
Streams	...	5,363,493	
Artist	...	JACKBOYS	
Artist Followers	...	437907	
Song ID	...	62zKJrpbLxz6InR3tGyr7o	
Genre	...	['rap', 'trap']	
Release Date	...	2019-12-27	
Weeks Charted	...	2019-12-27--2020-01-03	
Popularity	...	56	
Danceability	...	0.413	
Energy	...	0.13	
Loudness	...	-25.166	
Speechiness	...	0.0336	
Acousticness	...	0.9	
Liveness	...	0.111	
Tempo	...	123.342	
Duration (ms)	...	46837	
Valence	...	0.0676	
Chord	...	C	

		1547	\
Highest Charting Position		156	
Number of Times Charted		1	
Week of Highest Charting		2019-12-27--2020-01-03	
Song Name		Combachy (feat. MC Rebecca)	
Streams		5,149,797	
Artist		Anitta, Lexa, Luísa Sonza	
Artist Followers		10741972	
Song ID		2bPtwnrpFNEe8N7Q85kLHw	
Genre		['funk carioca', 'funk pop', 'pagode baiano', ...]	
Release Date		2019-11-20	
Weeks Charted		2019-12-27--2020-01-03	
Popularity		64	
Danceability		0.826	
Energy		0.73	
Loudness		-3.032	
Speechiness		0.0809	
Acousticness		0.383	
Liveness		0.0197	
Tempo		150.134	

Duration (ms)	157600
Valence	0.605
Chord	C#/Db

	1548 \
Highest Charting Position	178
Number of Times Charted	1
Week of Highest Charting	2019-12-27--2020-01-03
Song Name	Old Town Road
Streams	4,852,004
Artist	Lil Nas X
Artist Followers	5488666
Song ID	2YpeDb67231RjR0MgVLzsG
Genre	['lgbtq+ hip hop', 'pop rap']
Release Date	2019-06-21
Weeks Charted	2019-12-27--2020-01-03
Popularity	81
Danceability	0.878
Energy	0.619
Loudness	-5.56
Speechiness	0.102
Acousticness	0.0533
Liveness	0.113
Tempo	136.041
Duration (ms)	157067
Valence	0.639
Chord	F#/Gb

	1549 \
Highest Charting Position	187
Number of Times Charted	1
Week of Highest Charting	2019-12-27--2020-01-03
Song Name	Let Me Know (I Wonder Why Freestyle)
Streams	4,701,532
Artist	Juice WRLD
Artist Followers	19102888
Song ID	3wwo0bJvDSorOpNfzEkfXx
Genre	['chicago rap', 'melodic rap']
Release Date	2019-12-07
Weeks Charted	2019-12-27--2020-01-03
Popularity	76
Danceability	0.635
Energy	0.537
Loudness	-7.895
Speechiness	0.0832
Acousticness	0.172
Liveness	0.418

Tempo	125.028
Duration (ms)	215381
Valence	0.383
Chord	G
	1550 \
Highest Charting Position	190
Number of Times Charted	1
Week of Highest Charting	2019-12-27--2020-01-03
Song Name	Ne reviens pas
Streams	4,676,857
Artist	Gradur, Heuss L'enfoiré
Artist Followers	1390813
Song ID	4TnFANpjVwVKWzKxNzIyFH
Genre	['francoton', 'french hip hop', 'pop urbaine', ...]
Release Date	2019-11-29
Weeks Charted	2019-12-27--2020-01-03
Popularity	62
Danceability	0.932
Energy	0.778
Loudness	-3.384
Speechiness	0.0638
Acousticness	0.212
Liveness	0.168
Tempo	124.996
Duration (ms)	188613
Valence	0.933
Chord	A#/Bb

	1551 \
Highest Charting Position	195
Number of Times Charted	1
Week of Highest Charting	2019-12-27--2020-01-03
Song Name	New Rules
Streams	4,630,675
Artist	Dua Lipa
Artist Followers	27167675
Song ID	2ekn2ttSfGqwhhate0LSR0
Genre	['dance pop', 'pop', 'uk pop']
Release Date	2017-06-02
Weeks Charted	2019-12-27--2020-01-03
Popularity	79
Danceability	0.762
Energy	0.7
Loudness	-6.021
Speechiness	0.0694
Acousticness	0.00261

Liveness	0.153
Tempo	116.073
Duration (ms)	209320
Valence	0.608
Chord	A

	1552 \
Highest Charting Position	196
Number of Times Charted	1
Week of Highest Charting	2019-12-27--2020-01-03
Song Name	Cheirosa - Ao Vivo
Streams	4,623,030
Artist	Jorge & Mateus
Artist Followers	15019109
Song ID	2PWjKmJyTZedpmOUa3a5da
Genre	['sertanejo', 'sertanejo universitario']
Release Date	2019-10-11
Weeks Charted	2019-12-27--2020-01-03
Popularity	66
Danceability	0.528
Energy	0.87
Loudness	-3.123
Speechiness	0.0851
Acousticness	0.24
Liveness	0.333
Tempo	152.37
Duration (ms)	181930
Valence	0.714
Chord	B

	1553 \
Highest Charting Position	197
Number of Times Charted	1
Week of Highest Charting	2019-12-27--2020-01-03
Song Name	Havana (feat. Young Thug)
Streams	4,620,876
Artist	Camila Cabello
Artist Followers	22698747
Song ID	1rfofaqEpACxVEHIZBJe6W
Genre	['dance pop', 'electropop', 'pop', 'post-teen ...
Release Date	2018-01-12
Weeks Charted	2019-12-27--2020-01-03
Popularity	81
Danceability	0.765
Energy	0.523
Loudness	-4.333
Speechiness	0.03

Acousticness	0.184
Liveness	0.132
Tempo	104.988
Duration (ms)	217307
Valence	0.394
Chord	D

	1554 \
Highest Charting Position	198
Number of Times Charted	1
Week of Highest Charting	2019-12-27--2020-01-03
Song Name	Surtada - Remix Brega Funk
Streams	4,607,385
Artist	Dadá Boladão, Tati Zaqui, OIK
Artist Followers	208630
Song ID	5F8ffc8KWKNawllr5WsW0r
Genre	['brega funk', 'funk carioca']
Release Date	2019-09-25
Weeks Charted	2019-12-27--2020-01-03
Popularity	60
Danceability	0.832
Energy	0.55
Loudness	-7.026
Speechiness	0.0587
Acousticness	0.249
Liveness	0.182
Tempo	154.064
Duration (ms)	152784
Valence	0.881
Chord	F

	1555
Highest Charting Position	199
Number of Times Charted	1
Week of Highest Charting	2019-12-27--2020-01-03
Song Name	Lover (Remix) [feat. Shawn Mendes]
Streams	4,595,450
Artist	Taylor Swift
Artist Followers	42227614
Song ID	3i9UVldZ0EOaD0JnyfAZZ0
Genre	['pop', 'post-teen pop']
Release Date	2019-11-13
Weeks Charted	2019-12-27--2020-01-03
Popularity	70
Danceability	0.448
Energy	0.603
Loudness	-7.176

Speechiness	0.064
Acousticness	0.433
Liveness	0.0862
Tempo	205.272
Duration (ms)	221307
Valence	0.422
Chord	G

[22 rows x 1556 columns]

4.3 Convert object columns with numbers to float64

```
[265]: # List of columns to convert
columns_to_convert = ['Artist Followers', 'Streams', 'Popularity',
                      'Danceability', 'Energy', 'Loudness',
                      'Speechiness', 'Acousticness', 'Liveness', 'Tempo',
                      'Duration (ms)', 'Valence']

# Convert columns to numeric
for column in columns_to_convert:
    df_1[column] = pd.to_numeric(df_1[column], errors='coerce')
```

```
[266]: df_1.dtypes
```

```
[266]: Index                int64
Highest Charting Position    int64
Number of Times Charted      int64
Week of Highest Charting     object
Song Name                   object
Streams                     float64
Artist                     object
Artist Followers            float64
Song ID                    object
Genre                      object
Release Date               object
Weeks Charted              object
Popularity                 float64
Danceability               float64
Energy                    float64
Loudness                   float64
Speechiness                float64
Acousticness               float64
Liveness                   float64
Tempo                     float64
Duration (ms)              float64
Valence                    float64
Chord                      object
```

dtype: object

5 Data Cleaning Continued: Prepare DataFrame for Modeling and Training

```
[267]: df_1 = df_1.drop("Index", axis = 1)
```

```
[268]: df_1
```

```
[268]:
```

	Highest Charting Position	Number of Times Charted	\
0	1	8	
1	2	3	
2	1	11	
3	3	5	
4	5	1	
...	
1551	195	1	
1552	196	1	
1553	197	1	
1554	198	1	
1555	199	1	

	Week of Highest Charting	Song Name	Streams	\
0	2021-07-23--2021-07-30	Beggin'	NaN	
1	2021-07-23--2021-07-30	STAY (with Justin Bieber)	NaN	
2	2021-06-25--2021-07-02	good 4 u	NaN	
3	2021-07-02--2021-07-09	Bad Habits	NaN	
4	2021-07-23--2021-07-30	INDUSTRY BABY (feat. Jack Harlow)	NaN	
...	
1551	2019-12-27--2020-01-03	New Rules	NaN	
1552	2019-12-27--2020-01-03	Cheirosa - Ao Vivo	NaN	
1553	2019-12-27--2020-01-03	Havana (feat. Young Thug)	NaN	
1554	2019-12-27--2020-01-03	Surtada - Remix Brega Funk	NaN	
1555	2019-12-27--2020-01-03	Lover (Remix) [feat. Shawn Mendes]	NaN	

	Artist	Artist Followers	Song ID	\
0	Måneskin	3377762.0	3Wrjm47oTz2sjIgck1115e	
1	The Kid LAROI	2230022.0	5HCyWlXZPP0y6Gqq8TgA20	
2	Olivia Rodrigo	6266514.0	4ZtFanR9U6ndgddUvNcjcG	
3	Ed Sheeran	83293380.0	6PQ88X9TkUIAUIZJHW2upE	
4	Lil Nas X	5473565.0	27NovPIUIRr0ZoCHxABJwK	
...	
1551	Dua Lipa	27167675.0	2ekn2ttSfGqwhhate0LSR0	
1552	Jorge & Mateus	15019109.0	2PWjKmJyTZEDpmOUa3a5da	
1553	Camila Cabello	22698747.0	1rfofaqEpACxVEHIZBJe6W	
1554	Dadá Boladão, Tati Zaqui, OIK	208630.0	5F8ffc8KWKNAwllr5WsW0r	

1555 Taylor Swift 42227614.0 3i9UVldZ0E0aD0JnyfAZZ0

		Genre	Release Date	...	\
0	['indie rock italiano', 'italian pop']		2017-12-08	...	
1	['australian hip hop']		2021-07-09	...	
2	['pop']		2021-05-21	...	
3	['pop', 'uk pop']		2021-06-25	...	
4	['lgbtq+ hip hop', 'pop rap']		2021-07-23	...	
...		
1551	['dance pop', 'pop', 'uk pop']		2017-06-02	...	
1552	['sertanejo', 'sertanejo universitario']		2019-10-11	...	
1553	['dance pop', 'electropop', 'pop', 'post-teen ...		2018-01-12	...	
1554	['brega funk', 'funk carioca']		2019-09-25	...	
1555	['pop', 'post-teen pop']		2019-11-13	...	

	Danceability	Energy	Loudness	Speechiness	Acousticness	Liveness	\
0	0.714	0.800	-4.808	0.0504	0.12700	0.3590	
1	0.591	0.764	-5.484	0.0483	0.03830	0.1030	
2	0.563	0.664	-5.044	0.1540	0.33500	0.0849	
3	0.808	0.897	-3.712	0.0348	0.04690	0.3640	
4	0.736	0.704	-7.409	0.0615	0.02030	0.0501	
...	
1551	0.762	0.700	-6.021	0.0694	0.00261	0.1530	
1552	0.528	0.870	-3.123	0.0851	0.24000	0.3330	
1553	0.765	0.523	-4.333	0.0300	0.18400	0.1320	
1554	0.832	0.550	-7.026	0.0587	0.24900	0.1820	
1555	0.448	0.603	-7.176	0.0640	0.43300	0.0862	

	Tempo	Duration (ms)	Valence	Chord
0	134.002	211560.0	0.589	B
1	169.928	141806.0	0.478	C#/Db
2	166.928	178147.0	0.688	A
3	126.026	231041.0	0.591	B
4	149.995	212000.0	0.894	D#/Eb
...
1551	116.073	209320.0	0.608	A
1552	152.370	181930.0	0.714	B
1553	104.988	217307.0	0.394	D
1554	154.064	152784.0	0.881	F
1555	205.272	221307.0	0.422	G

[1556 rows x 22 columns]

```
[269]: df_clean_2 = df_1.copy()
```

5.1 Identify Object Columns & Drop them

```
[270]: object_columns = df_clean_2.select_dtypes(include=['object']).columns
df_clean_2 = df_clean_2.drop(columns=object_columns)
```

```
[271]: df_clean_2.isnull().sum()
```

```
[271]: Highest Charting Position      0
Number of Times Charted            0
Streams                          1556
Artist Followers                  11
Popularity                        11
Danceability                      11
Energy                           11
Loudness                          11
Speechiness                       11
Acousticness                      11
Liveness                          11
Tempo                             11
Duration (ms)                     11
Valence                           11
dtype: int64
```

```
[272]: df_clean_2.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1556 entries, 0 to 1555
Data columns (total 14 columns):
#   Column                                Non-Null Count  Dtype
---  -
0   Highest Charting Position             1556 non-null   int64
1   Number of Times Charted               1556 non-null   int64
2   Streams                              0 non-null      float64
3   Artist Followers                     1545 non-null   float64
4   Popularity                           1545 non-null   float64
5   Danceability                         1545 non-null   float64
6   Energy                               1545 non-null   float64
7   Loudness                             1545 non-null   float64
8   Speechiness                          1545 non-null   float64
9   Acousticness                         1545 non-null   float64
10  Liveness                             1545 non-null   float64
11  Tempo                                1545 non-null   float64
12  Duration (ms)                        1545 non-null   float64
13  Valence                              1545 non-null   float64
dtypes: float64(12), int64(2)
memory usage: 170.3 KB
```


5.2 Drop Streams Column (essentially empty)

```
[273]: df_clean_2.drop('Streams', axis = 1, inplace = True)
```

```
[274]: df_clean_2.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1556 entries, 0 to 1555
Data columns (total 13 columns):
#   Column                                Non-Null Count  Dtype
---  -
0   Highest Charting Position             1556 non-null   int64
1   Number of Times Charted               1556 non-null   int64
2   Artist Followers                      1545 non-null   float64
3   Popularity                           1545 non-null   float64
4   Danceability                         1545 non-null   float64
5   Energy                               1545 non-null   float64
6   Loudness                             1545 non-null   float64
7   Speechiness                          1545 non-null   float64
8   Acousticness                         1545 non-null   float64
9   Liveness                             1545 non-null   float64
10  Tempo                                1545 non-null   float64
11  Duration (ms)                        1545 non-null   float64
12  Valence                              1545 non-null   float64
dtypes: float64(11), int64(2)
memory usage: 158.2 KB
```

5.3 Get means and replace null values with mean per column

```
[275]: df_clean_2.isna().sum()
```

```
[275]: Highest Charting Position      0
      Number of Times Charted      0
      Artist Followers             11
      Popularity                   11
      Danceability                  11
      Energy                       11
      Loudness                     11
      Speechiness                   11
      Acousticness                  11
      Liveness                      11
      Tempo                        11
      Duration (ms)                 11
      Valence                       11
      dtype: int64
```

```
[276]: null_columns = df_clean_2.columns[df_clean_2.isnull().any()].tolist()
      print("Columns with null values:")
```

```
null_columns
```

Columns with null values:

```
[276]: ['Artist Followers',  
        'Popularity',  
        'Danceability',  
        'Energy',  
        'Loudness',  
        'Speechiness',  
        'Acousticness',  
        'Liveness',  
        'Tempo',  
        'Duration (ms)',  
        'Valence']
```

```
[277]: for col in null_columns:  
        #Calculate the mean, excluding NaN values  
        mean= df_clean_2[col].mean(skipna=True)  
  
        #replace NaNs with the mean per column  
        df_clean_2[col] = df_clean_2[col].fillna(mean)
```

```
[278]: print("\nNull value count after replacement:")  
        print(df_clean_2.isnull().sum())
```

Null value count after replacement:

Highest Charting Position	0
Number of Times Charted	0
Artist Followers	0
Popularity	0
Danceability	0
Energy	0
Loudness	0
Speechiness	0
Acousticness	0
Liveness	0
Tempo	0
Duration (ms)	0
Valence	0
dtype:	int64

```
[279]: df_clean_2.dtypes
```

```
[279]: Highest Charting Position      int64  
        Number of Times Charted      int64  
        Artist Followers              float64
```

```

Popularity          float64
Danceability        float64
Energy              float64
Loudness            float64
Speechiness         float64
Acousticness        float64
Liveness            float64
Tempo               float64
Duration (ms)       float64
Valence             float64
dtype: object

```

5.4 Drop columns that have no relation to target = “Popularity”

```
[280]: df_clean_2.drop('Highest Charting Position', axis = 1, inplace = True)
```

```
[281]: df_clean_2.drop('Number of Times Charted', axis = 1, inplace = True)
```

```
[282]: df_clean_2.drop('Artist Followers', axis = 1, inplace = True)
```

```
[283]: df_scaling = df_clean_2.copy()
```

6 Data Scaling

6.1 Data Scaling (standard scaler)

```
[284]: df_3_std = df_scaling.copy()
```

```
[285]: df_3_std.head(10)
```

```
[285]:
```

	Popularity	Danceability	Energy	Loudness	Speechiness	Acousticness	\
0	100.0	0.714	0.800	-4.808	0.0504	0.12700	
1	99.0	0.591	0.764	-5.484	0.0483	0.03830	
2	99.0	0.563	0.664	-5.044	0.1540	0.33500	
3	98.0	0.808	0.897	-3.712	0.0348	0.04690	
4	96.0	0.736	0.704	-7.409	0.0615	0.02030	
5	97.0	0.610	0.508	-6.682	0.1520	0.29700	
6	94.0	0.762	0.701	-3.541	0.0286	0.23500	
7	95.0	0.780	0.718	-3.605	0.0506	0.31000	
8	96.0	0.644	0.648	-4.601	0.1180	0.27600	
9	95.0	0.750	0.608	-4.008	0.0387	0.00165	

	Liveness	Tempo	Duration (ms)	Valence
0	0.3590	134.002	211560.0	0.589
1	0.1030	169.928	141806.0	0.478
2	0.0849	166.928	178147.0	0.688
3	0.3640	126.026	231041.0	0.591

4	0.0501	149.995	212000.0	0.894
5	0.3840	178.818	137876.0	0.758
6	0.1230	110.968	208867.0	0.742
7	0.0932	127.949	199604.0	0.342
8	0.1350	179.951	206710.0	0.440
9	0.1780	132.507	173347.0	0.958

```
[286]: x = df_3_std.drop(['Popularity'], axis=1)
y = df_3_std['Popularity']

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

```
[287]: df_4_std = df_3_std.copy()
```

6.2 Scaling training and testing data

```
[288]: scaler = StandardScaler()
X_train_std = scaler.fit_transform(X_train)
X_test_std = scaler.transform(X_test)
```

6.3 Data Scaling Continued (min-max scaler)

```
[289]: df_3_mm = df_scaling.copy()
```

```
[290]: df_3_mm.head(10)
```

```
[290]:
```

	Popularity	Danceability	Energy	Loudness	Speechiness	Acousticness	\
0	100.0	0.714	0.800	-4.808	0.0504	0.12700	
1	99.0	0.591	0.764	-5.484	0.0483	0.03830	
2	99.0	0.563	0.664	-5.044	0.1540	0.33500	
3	98.0	0.808	0.897	-3.712	0.0348	0.04690	
4	96.0	0.736	0.704	-7.409	0.0615	0.02030	
5	97.0	0.610	0.508	-6.682	0.1520	0.29700	
6	94.0	0.762	0.701	-3.541	0.0286	0.23500	
7	95.0	0.780	0.718	-3.605	0.0506	0.31000	
8	96.0	0.644	0.648	-4.601	0.1180	0.27600	
9	95.0	0.750	0.608	-4.008	0.0387	0.00165	

	Liveness	Tempo	Duration (ms)	Valence
0	0.3590	134.002	211560.0	0.589
1	0.1030	169.928	141806.0	0.478
2	0.0849	166.928	178147.0	0.688
3	0.3640	126.026	231041.0	0.591
4	0.0501	149.995	212000.0	0.894
5	0.3840	178.818	137876.0	0.758
6	0.1230	110.968	208867.0	0.742

7	0.0932	127.949	199604.0	0.342
8	0.1350	179.951	206710.0	0.440
9	0.1780	132.507	173347.0	0.958

```
[291]: columns_to_scale = ['Popularity', 'Danceability', 'Energy',
↳ 'Loudness', 'Speechiness', 'Acousticness', 'Liveness', 'Tempo', 'Duration_
↳ (ms)', 'Valence']
```

```
[292]: scaler = MinMaxScaler()
scaler.fit(df_3_mm[columns_to_scale])
```

```
[292]: MinMaxScaler()
```

```
[293]: df_3_mm[columns_to_scale] = scaler.transform(df_3_mm[columns_to_scale])
```

```
[294]: print(df_3_mm.info())
df_3_mm.head()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1556 entries, 0 to 1555
Data columns (total 10 columns):
#   Column                Non-Null Count  Dtype
---  -
0   Popularity            1556 non-null   float64
1   Danceability          1556 non-null   float64
2   Energy                1556 non-null   float64
3   Loudness              1556 non-null   float64
4   Speechiness           1556 non-null   float64
5   Acousticness           1556 non-null   float64
6   Liveness              1556 non-null   float64
7   Tempo                 1556 non-null   float64
8   Duration (ms)         1556 non-null   float64
9   Valence               1556 non-null   float64
dtypes: float64(10)
memory usage: 121.7 KB
None
```

```
[294]:
```

	Popularity	Danceability	Energy	Loudness	Speechiness	Acousticness	\
0	1.00	0.679518	0.814410	0.763187	0.031599	0.127744	
1	0.99	0.531325	0.775109	0.737844	0.029159	0.038507	
2	0.99	0.497590	0.665939	0.754339	0.151952	0.337005	
3	0.98	0.792771	0.920306	0.804274	0.013476	0.047159	
4	0.96	0.706024	0.709607	0.665679	0.044493	0.020397	

	Liveness	Tempo	Duration (ms)	Valence
0	0.360076	0.550500	0.325134	0.588173
1	0.088401	0.777085	0.200129	0.470961
2	0.069192	0.758164	0.265255	0.692714

3	0.365383	0.500196	0.360046	0.590285
4	0.032261	0.651368	0.325923	0.910243

```
[295]: df_4_mm = df_3_mm.copy()
```

7 Model Selection and Training

7.1 Baseline Model: Linear Regression

```
[296]: lr_model = LinearRegression()
lr_model.fit(X_train_std, y_train)
y_pred_lr = lr_model.predict(X_test_std)
print('Linear Regression:')
print(f"RMSE: {np.sqrt(mean_squared_error(y_test, y_pred_lr))}")
print(f"R^2: {r2_score(y_test, y_pred_lr)}")
```

Linear Regression:
 RMSE: 14.364423937752143
 R^2: 0.029078749563385187

7.2 Decision Tree Model

```
[297]: dt_model = DecisionTreeRegressor(random_state=42)
dt_model.fit(X_train_std, y_train)
y_pred_dt = dt_model.predict(X_test_std)

print("\nDecision Tree:")
print(f"RMSE: {np.sqrt(mean_squared_error(y_test, y_pred_dt))}")
print(f"R2 Score: {r2_score(y_test, y_pred_dt)}")
```

Decision Tree:
 RMSE: 24.779928809318072
 R2 Score: -1.8894001251298964

7.3 Random Forest Model

```
[299]: rf_model = RandomForestRegressor(n_estimators=100, random_state=42)
rf_model.fit(X_train_std, y_train)
y_pred_rf = rf_model.predict(X_test_std)

print("\nRandom Forest:")
print(f"RMSE: {np.sqrt(mean_squared_error(y_test, y_pred_rf))}")
print(f"R2 Score: {r2_score(y_test, y_pred_rf)}")
```

Random Forest:
 RMSE: 15.339708471840504
 R2 Score: -0.10724006098445682

7.4 XGBoost Model

```
[304]: xgb_model = xgb.XGBRegressor(n_estimators=100, random_state=42)
xgb_model.fit(X_train_std, y_train)
y_pred_xgb = xgb_model.predict(X_test_std)

print("\nXGBoost:")
print(f"RMSE: {np.sqrt(mean_squared_error(y_test, y_pred_xgb))}")
print(f"R2 Score: {r2_score(y_test, y_pred_xgb)}")
```

XGBoost:

RMSE: 16.524247365963873

R2 Score: -0.28484563649801675