Final Project Report:

An Analysis of Popular Music through Spotify

SI 370

Dr. Chris Teplovs

December 01, 2022

Lauren Fulcher, Ryan Berger,

Abby Williams, & Nikita Gollapudi

I. Purpose

Nature of Project and Why We Chose It

Music is an art enjoyed by billions of people across the world. This is a common interest that we all share and it is interesting how different people can connect through music even if they prefer different genres. From rock and pop to metal and rap, music can be a defining factor in people's lives. But what service do people think of when it comes to actually listening to music? Some might say they use Apple Music, Amazon Music, or maybe even Youtube Music. But these services do not hold a candle to the most dominant music streaming service, Spotify, which holds about a third of the market share in this industry. The data behind Spotify spans a wide range of possibilities for analytics. The top hits and various variables that go into determining what makes a song popular on the service are truly fascinating. In this project, we have decided to take a closer look at what truly determines the popularity of songs on Spotify by looking at some of the highest-charting songs on the platform. Through our analysis, we hope to be able to make inferences about the different variables that go into determining the popularity of a song on Spotify.

Statement of Purpose (Goal Attempting to be Addressed)

Our proposed project is founded on the goal to understand what goes into making a song popular, including its variables and the relationships between them. For example, we want to understand whether or not a song will be popular based on its genre, danceability, tempo, and other quantifiable variables pertaining to the song's features. The ideal outcome behind our analysis would be to understand if there is a correlation between these variables, which variables have the most weight, and what is important to consider when creating a popular song. We also wanted to understand what features are relevant in the classification of a song's genre, with a goal to understand what variables are important in song creation.

II. Analysis

Overview of the Dataset

The data we used is from the 'Spotify Top 200 Charts' dataset on Kaggle. This data includes every song on the Top 200 Charts of Spotify during 2020 and 2021, along with statistics for each song that provide relevant information regarding the various features that the song has and the genres it is considered to be in. The dataset has 23 columns and over 1000 rows of data, so we knew we would have the information necessary to ground our conclusions in enough empirical evidence. We planned on looking at the "Danceability", "Acousticness", "Speechiness", "Liveness", "Valence", and "Energy" of each song - among other variables - and comparing them with popularity to determine if there's a trend for which songs are more popular. The majority of these features are on a scale from 0.0 to 1.0. For instance, if a song's 'Danceability' rating is closer to 1.0, that song is considered to be a great song to dance to. The same logic applies to the other variables on this scale, where 0 is the lowest point, and 1.0 is the highest. One thing that is especially interesting about the data itself is that this dataset is based on the

global top charts on Spotify, as opposed to purely domestic charts. It may be worth noting that the difference between what is popular domestically and internationally may change the highest-ranking songs in the dataset, and we may find songs in the dataset that are unfamiliar to us. Aside from the project itself, after completing initial cleaning and exploratory data analysis, the dataset could also be used for identifying new and unique songs to listen to based on their core features.

Data Cleaning

For data cleaning, we did an initial exploratory deep dive into the data, looking at the dataset's variables by utilizing the 'info()' and 'describe()' functions. From this initial analysis, we noticed that all of the variables, even the numeric variables, were in the form of strings. There existed columns such as 'Streams', 'Artist Followers', and 'Popularity' that needed to be converted to integer values. There also were columns that represented the descriptive features of the songs, like 'Danceability', 'Energy', and 'Acousticness', which needed to be converted to float values because of the existence of a decimal. The 'Genre' column was an especially interesting case. To use it, this column needed to be handled in a way that turned the string representation of a list into an actual list object. In terms of the data as a whole, the data set was clean of any missing values, although a handful of rows contained a space character in place of any of the row's unknown data. These rows were removed from the dataset so that we could work with the most accurate set of information. Before moving forward with any analysis, we ensured that there were no fields with empty values. After all necessary data cleaning was completed, the resulting data frame variable 'spotify' included 1545 rows of individual songs, as well as 23 columns that represent the numeric and categorical information about each song. Because the genres column initially contained a list of genres for each song, given that any song could belong to multiple genres, we needed to "explode" that column and assign this secondary dataset to a new variable. To analyze the data solely based on the genres of each song, we created the data frame variable, 'spot_genres', that exploded the genre column so that each genre for any given song had its own row. Given the two datasets 'spot_genres' and 'spotify' after the initial cleaning, the data was ready to be analyzed.

Analytic Techniques

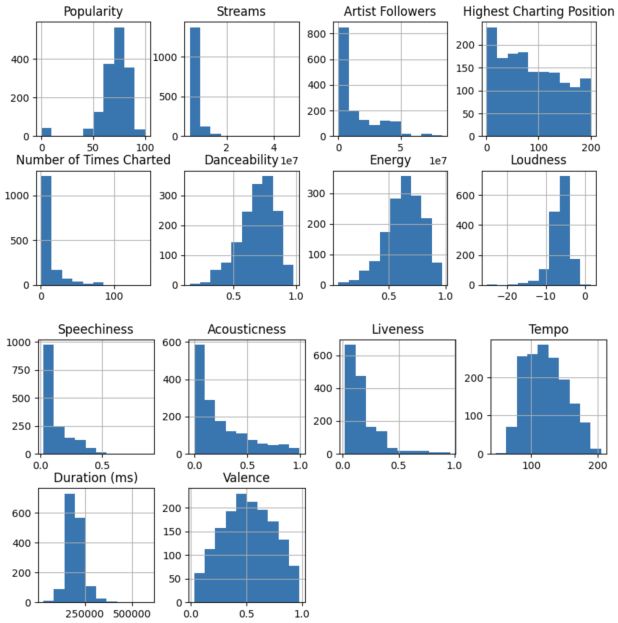
To accomplish the goals established in the statement of purpose, the project planned to cover multiple analytic techniques to reaffirm the relationships between each song's variables. These techniques include exploratory data analysis where we initially created a plot with the distribution of the most significant features in the top 20 songs to visualize which variables we should afford additional attention to for our analysis, running a regression analysis based on which variables strongly affect the popularity of a song, as well as machine learning techniques to emphasize the importance of these relationships in contributing to a song's genre. To incorporate all of these techniques, we began by importing the libraries that gave us the relevant functions to perform our analyses.

The initial major analysis techniques we used after exploratory data analysis were statistical analysis techniques by using reg plots, and regression and ANOVA models to determine the correlation and significance between variables. We visualized these relationships through reg plots to plot the data and the linear regression model fit. Seeing the correlations between 'Popularity' and the rest of the song feature variables was relevant in our understanding of the state of the data. The analysis requires clear insights into what factors make a song popular and high-ranking on the Spotify streaming charts. We used ordinary least squares from the statsmodels library to perform the regression analysis, and the ANOVA.

Another analysis implemented was employing machine learning algorithms to classify a song's genre based on its feature values. We implemented a model that is trained on 80% of the song descriptive variables in the spot_genres dataset and tested on the remaining 20%. The model uses spot_genre's descriptive variables like 'Energy', 'Danceability', and 'Speechiness', among others, to learn how to classify a song as any given genre. These X values were scaled using the sklearn library's 'StandardScaler()' preprocessing function. We then used a Voting classifier to ensemble models together to make predictions, which is further highlighted in our main findings.

Exploratory Data Analysis Visualizations

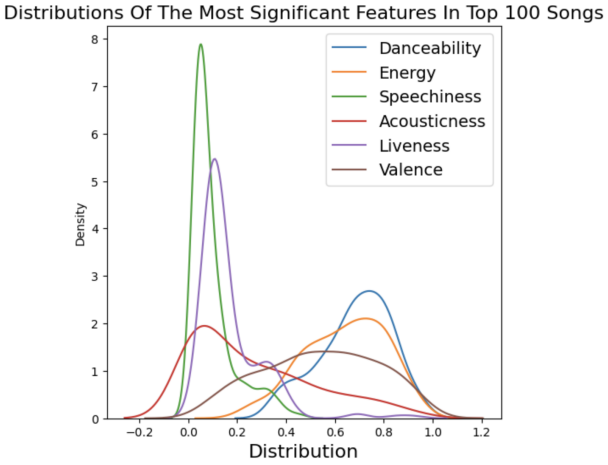
Visualization #1 - Song Feature Histograms



To better understand the variables that make up our dataset, we constructed histograms of each numerical column, shown in visualization #1 (left). Each of these histograms showed the distribution of popular songs with certain characteristics and behaviors associated with them. This was a good way for us to begin to understand what features go into making a song successful, and how these features differ from each other.

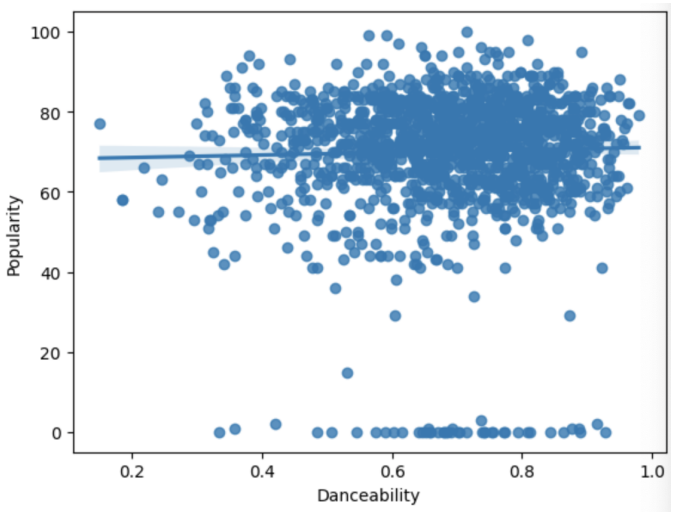
Visualization #2 - Distribution of Features

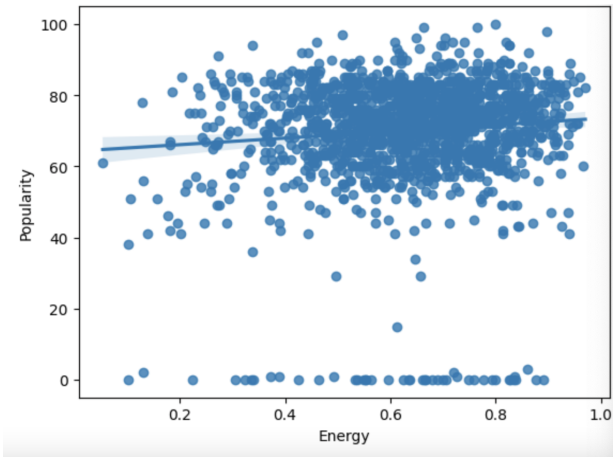
Illustrated by visualization #2 (right) are the distributions of the most significant features in the dataset in order to determine what we should consider looking for. The plot is made up of the top 100 most popular songs in the dataset and the

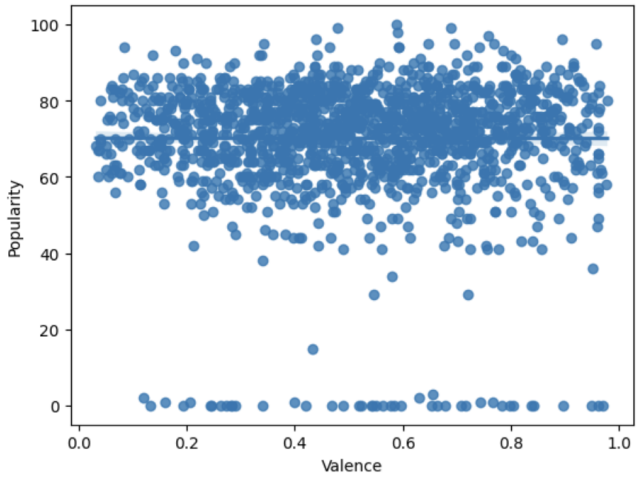


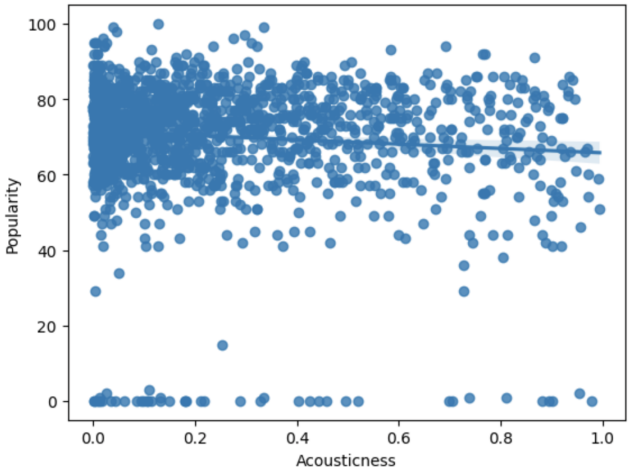
relevant features of these songs. It was interesting to note that a similar distribution can be seen between the ‘Speechiness’ and ‘Liveness’ as well as the ‘Danceability’ and ‘Energy’ of these songs, as these relationships could be investigated further.

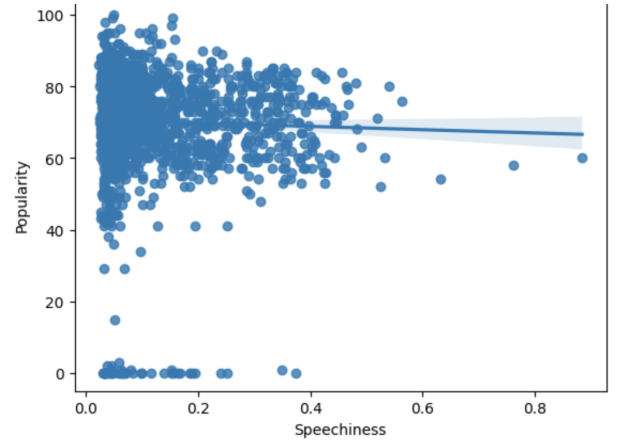
Visualization #3 - Linear Regression Model Fit Plots

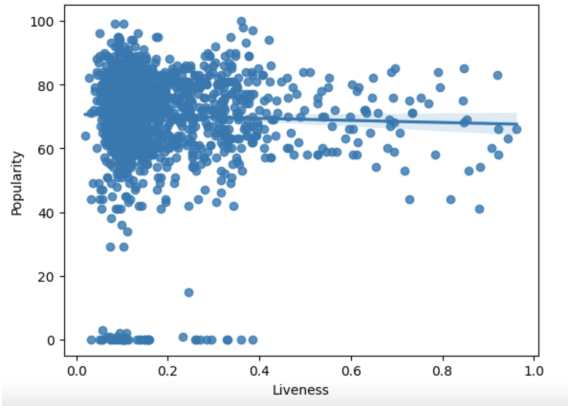






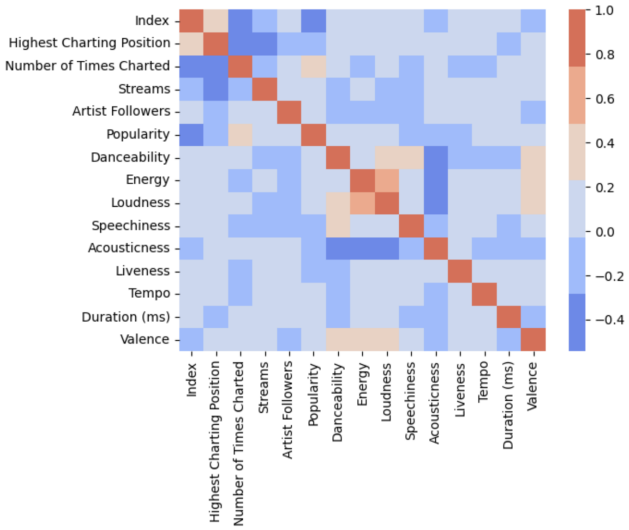






The six plots that form visualization #3 (above) utilize Seaborn's 'regplot' function, which results in the shown scatterplots between a plethora of descriptive features and the popularity of a song. The function plots the fitted "y ~ x" regression model as a line and a 95% confidence interval for that regression on top of the previously mentioned scatterplot. Given this information, we can understand the way that the data points contribute to the regression, and therefore make better sense of the resulting statistics.

Visualization #4 - Feature Relationship Heatmap



To further visualize the correlations between song features in one place, we created visualization #4 (left): a heatmap using Seaborn plots showing the strength of each of these relationships. This heatmap allowed a basic understanding of which variables may be correlated with each other and with popularity or the highest charting position.

III. Main Findings

Interesting Relationships & Insights from Our Analysis

To determine the relationships between the features we focused on with popularity, we decided to conduct a regression analysis on each variable. Based on the regplots above, it seems that there are slight to no correlations between the variables, and we wanted to explore this further. The probability of the F-statistic that is given in each of the ordinary least squares (OLS) regression summaries is an indication of the relationship between variables in the regression. Figures 1 - 6 show the OLS regressions for the six features we decided to focus on. Given these summaries, it was found that 'Energy', 'Valence', and 'Acousticness' were statistically significant, as their P-values were less than 0.05 ('Acousticness', 'Energy', and 'Valence' less than 0.0001, 0.0003, and 0.001 respectively) which typically would determine that these three variables have a relationship with popularity. For the variables of 'Speechiness', 'Liveness', and 'Danceability', the p-values are greater than 0.05, indicating that there would be no statistically significant relationship between those variables and popularity. The one issue with these results is that the r-squared value is close to zero, which signifies that the data is not close to the regression line. Thus we may not be able to confidently state that the features overall have a significant relationship with popularity. The regression model contains low p-values for the three variables

which signifies the existence of relationships, but this low r-squared value could be a cause for concern when it comes to the accuracy of them.

Because of these results, we conducted further analysis to determine if popularity had a relationship with genre, where we found a positive correlation with an r-squared value of 0.277. We also can determine that there is a relationship between a song's genre and its popularity as the p-value is extremely low (shown below), concluding that these results are statistically significant.

```
import statsmodels.api as sm
import statsmodels.formula.api as smf
regression = smf.ols(" Q('Popularity') ~  Q('Genre') ",  data=spot_genres).fit()
regression.summary()
```

Dep. Variable:	Q('Popularity')	R-squared:	0.277
Model:	OLS	Adj. R-squared:	0.220
Method:	Least Squares	F-statistic:	4.887
Date:	Fri, 02 Dec 2022	Prob (F-statistic):	1.67e-135
Time:	02:48:41	Log-Likelihood:	-16551.
No. Observations:	4579	AIC:	3.377e+04
Df Residuals:	4245	BIC:	3.592e+04
Df Model:	333		
Covariance Type:	nonrobust		

OLS Regression Results

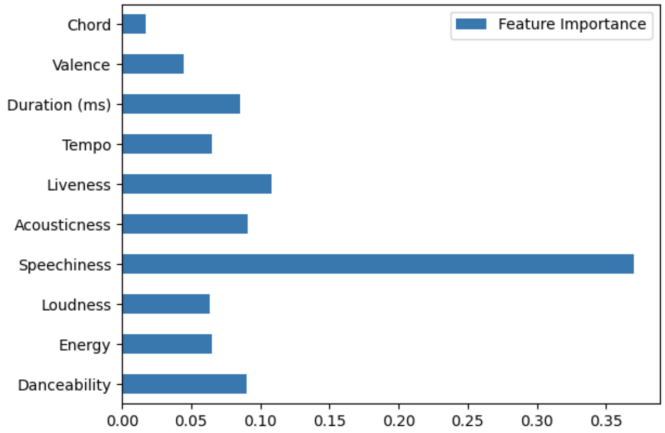
Focusing on 'Popularity' and 'Genre', we also conducted an ANOVA analysis to further back up our regression analysis and determine if the popularity of a song varies significantly between genres. In an ANOVA analysis, the p-value is deemed significant to reject a given null hypothesis if it is less than .05. In this case (shown below) our low p-value tells us that popularity varies significantly between genres. This makes sense, as a genre is not necessarily an indication of a given song's popularity, but rather a separate, independent factor to consider for any given song.

```
from statsmodels.formula.api import ols
genre_anova = ols("Popularity ~ Genre", data=spot_genres).fit()
table = sm.stats.anova_lm(genre_anova, typ=2) # Type 2 ANOVA DataFrame
table
```

	sum_sq float64	df float64	F float64	PR(>F) float64
Genre	141724.88019488	333.0	4.886969096458656	1.6735904021054703e-135
Residual	369691.92696825264	4245.0	nan	nan

We then created a machine learning model with the genres and the corresponding features to determine what features can predict a song's genre. This service is important after determining

through our ANOVA analysis that popularity significantly varies between genres. The most accurate predictions were found to occur when we used a Voting classifier to ensemble multiple models together. The classifiers that were a part of the aforementioned ensemble included K Nearest Neighbors, Decision Tree, Random Forest, MLP, AdaBoost, and Gaussian Naive Bayes. This ensemble model produced a 71% accuracy in its predictions of any song's genre (Figure #7). We also implemented feature selection on the model, which showed that 'Speechiness' is by far the most telling feature of a song's genre, followed by 'Danceability'. Our previous regression analysis supports the claim that an artist should take the genre of their song into consideration if they aim to create more popular music. Through the model's feature importance displayed in the visualization below, insight is given into which features influence the model most. This insight, along with the regression analysis of genre stated earlier, could provide artists with even more detailed information on which components of their music they should put more focus on if they try to fit into a particular genre.



Feature Importance in Song Genre

Our secondary analysis focused on the genre of each song, attempting to predict a song's genre based on its core features. This plot shows which features are more telling of a song's genre. Of all the considered features, 'Speechiness' is by far the most indicating feature of a song's genre, followed by 'Danceability'. This was done to compare variables that are most influential for determining genre in order to predict them with further accuracy.

IV. Limitations and Challenges

What didn't work and why?

Throughout this project, we experienced some limitations and challenges that factored into the span and scope that our analysis could capture. The dataset is only among songs that reached the top 200 ranking on Spotify between these two years. Given that the data is only a representation of the top 200 songs from 2020 - 2021, this is a possible limitation of our analysis, as the popularity of songs varies over years due to factors such as new trends or changes in culture. Additionally, countless songs are outside of this top 200 rank that may have added value to our results, reducing the accuracy of our analysis by some factor. Within the data itself, there was one variable that was not fully explained. Based on the description of variables that accompany the dataset on Kaggle, the 'Popularity' column was described as "the value between 0 and 100, with 100 being the most popular" for each song. Despite being a rather intuitive ranking system, the calculation behind this metric is unknown and not clarified anywhere else on Kaggle. We are

not exactly sure how this ranking was calculated or put together. Given this, the overall accuracy of this ranking system is potentially limited and unknown.

In addition to our limitations, one challenge faced during the project involved our attempt to develop a machine learning model to predict the popularity of songs. Initially, our proposed project goal was to predict the popularity of songs based on the different variables in the dataset, determining features of songs that most accurately go into these predictions. We attempted this based on our classwork but found that the resulting accuracy scores were too low to indicate any meaningful relationships. As we understand it, this inaccuracy was caused by the float type of the 'Popularity' column. This factor made the classification process difficult, as any predictions that were not correct to the actual float decimal number—regardless of proximity—were not counted as correct. We tried to adapt to this challenge by utilizing the 'qcut()' function to create separate bins of popularity values as well as a model that predicts the range of popularity that a song should be depending on the other song features. This resulted in a higher accuracy rating than our initial prediction, but still not high enough to confidently rely on (results shown in Figure #8). After understanding that our challenges with prediction accuracy were not the result of slight differences in prediction values and actual values, we chose to adhere to the possibility that the variables in the dataset may not be the best way to predict popularity. This understanding was what inspired us to change course from popularity prediction to 'Genre' prediction instead, while also including a more in-depth analysis regarding the relationships between the variables.

An additional challenge we faced during the 'Genre' classification analysis was the variety of specific and descriptive genre names that made up the dataset. There were 334 different genres, including everything from 'West Coast Rap' to 'Tennessee Hip Hop'. This made it significantly more difficult to create the model, as there was not a lot of overlap between genres for the model to be trained on. To combat this challenge, we simplified and consolidated the genres. We created a function to remove any descriptive words that accompanied a general genre from that value and applied this function to the 'Genre' column of the data frame. This enabled the model to have enough information about each genre to train and predict more accurately.

V. Recommendations for Future Work

As a continuation of this project in the future, we would recommend a more in depth regression analysis of popularity with the song features, as our results produced very low r-squared values. We would also expand on the machine learning applications in order to increase accuracy. Furthermore, this can be applied to incoming data to predict whether new songs will be popular before the results even come out. The analysis could also benefit from utilizing additional data, such as the top Spotify charts from years prior, to increase the quality of our analysis and make more accurate predictions. Lastly, we could expand our analysis by applying our methods to songs beyond those that are the highest ranking, as the songs on that list are inherently and objectively more popular.

Appendix

Figure #1 - Popularity vs. Danceability Model

```
import statsmodels.api as sm
import statsmodels.formula.api as smf
regression = smf.ols(" Q('Popularity') ~ Q('Danceability')",  data=spotify).fit()
regression.summary()
```

Dep. Variable:	Q('Popularity')	R-squared:	0.001
Model:	OLS	Adj. R-squared:	0.000
Method:	Least Squares	F-statistic:	1.249
Date:	Fri, 02 Dec 2022	Prob (F-statistic):	0.264
Time:	02:26:05	Log-Likelihood:	-6457.7
No. Observations:	1545	AIC:	1.292e+04
Df Residuals:	1543	BIC:	1.293e+04
Df Model:	1		
Covariance Type:	nonrobust		

OLS Regression Results

	coef	std err	t	P>|t|	[0.025	0.975]
Intercept	67.9098	1.992	34.097	0.000	64.003	71.816
Q('Danceability')	3.1588	2.827	1.117	0.264	-2.386	8.704

Omnibus:	836.584	Durbin-Watson:	1.448
Prob(Omnibus):	0.000	Jarque-Bera (JB):	6250.002
Skew:	-2.464	Prob(JB):	0.00
Kurtosis:	11.532	Cond. No.	10.4

Figure #2 - Popularity vs. Acousticness Model

```
import statsmodels.api as sm
import statsmodels.formula.api as smf
regression = smf.ols(" Q('Popularity') ~ Q('Acousticness')",  data=spotify).fit()
regression.summary()
```

Dep. Variable:	Q('Popularity')	R-squared:	0.008
Model:	OLS	Adj. R-squared:	0.008
Method:	Least Squares	F-statistic:	12.95
Date:	Fri, 02 Dec 2022	Prob (F-statistic):	0.000329
Time:	02:21:23	Log-Likelihood:	-6451.9
No. Observations:	1545	AIC:	1.291e+04
Df Residuals:	1543	BIC:	1.292e+04
Df Model:	1		
Covariance Type:	nonrobust		

OLS Regression Results

	coef	std err	t	P>|t|	[0.025	0.975]
Intercept	71.5238	0.565	126.503	0.000	70.415	72.633
Q('Acousticness')	-5.7679	1.603	-3.599	0.000	-8.911	-2.624

Omnibus:	831.126	Durbin-Watson:	1.460
Prob(Omnibus):	0.000	Jarque-Bera (JB):	6201.491
Skew:	-2.444	Prob(JB):	0.00
Kurtosis:	11.511	Cond. No.	4.26

Figure #3 - Popularity Liveness Model

```
import statsmodels.api as sm
import statsmodels.formula.api as smf
regression = smf.ols("Q('Popularity') ~ Q('Liveness')",  data=spotify).fit()
regression.summary()
```

Dep. Variable:	Q('Popularity')	R-squared:	0.001
Model:	OLS	Adj. R-squared:	0.000
Method:	Least Squares	F-statistic:	1.340
Date:	Fri, 02 Dec 2022	Prob (F-statistic):	0.247
Time:	02:21:39	Log-Likelihood:	-6457.7
No. Observations:	1545	AIC:	1.292e+04
Df Residuals:	1543	BIC:	1.293e+04
Df Model:	1		
Covariance Type:	nonrobust		

OLS Regression Results

	coef	std err	t	P>|t|	[0.025	0.975]
Intercept	70.6756	0.647	109.247	0.000	69.407	71.945
Q('Liveness')	-3.2357	2.795	-1.158	0.247	-8.718	2.247

Omnibus:	841.438	Durbin-Watson:	1.448
Prob(Omnibus):	0.000	Jarque-Bera (JB):	6343.323
Skew:	-2.479	Prob(JB):	0.00
Kurtosis:	11.599	Cond. No.	7.18

Figure #4 - Popularity vs. Energy Model

```
import statsmodels.api as sm
import statsmodels.formula.api as smf
regression = smf.ols(" Q('Popularity') ~  Q('Energy') ",  data=spotify).fit()
regression.summary()
```

Dep. Variable:	Q('Popularity')	R-squared:	0.009
Model:	OLS	Adj. R-squared:	0.008
Method:	Least Squares	F-statistic:	13.96
Date:	Fri, 02 Dec 2022	Prob (F-statistic):	0.000194
Time:	02:26:04	Log-Likelihood:	-6451.4
No. Observations:	1545	AIC:	1.291e+04
Df Residuals:	1543	BIC:	1.292e+04
Df Model:	1		
Covariance Type:	nonrobust		

OLS Regression Results

	coef	std err	t	P>|t|	[0.025	0.975]
Intercept	64.2146	1.623	39.574	0.000	61.032	67.397
Q('Energy')	9.2735	2.482	3.736	0.000	4.405	14.142

Omnibus:	836.215	Durbin-Watson:	1.467
Prob(Omnibus):	0.000	Jarque-Bera (JB):	6293.140
Skew:	-2.460	Prob(JB):	0.00
Kurtosis:	11.577	Cond. No.	8.72

Figure #5 - Popularity vs. Speechiness Model

```
import statsmodels.api as sm
import statsmodels.formula.api as smf
regression = smf.ols(" Q('Popularity') ~ Q('Speechiness')",  data=spotify).fit()
regression.summary()
```

Dep. Variable:	Q('Popularity')	R-squared:	0.001
Model:	OLS	Adj. R-squared:	0.000
Method:	Least Squares	F-statistic:	1.591
Date:	Fri, 02 Dec 2022	Prob (F-statistic):	0.207
Time:	02:22:46	Log-Likelihood:	-6457.5
No. Observations:	1545	AIC:	1.292e+04
Df Residuals:	1543	BIC:	1.293e+04
Df Model:	1		
Covariance Type:	nonrobust		

OLS Regression Results

	coef	std err	t	P>|t|	[0.025	0.975]
Intercept	70.6582	0.605	116.882	0.000	69.472	71.844
Q('Speechiness')	-4.6004	3.648	-1.261	0.207	-11.755	2.554

Omnibus:	843.595	Durbin-Watson:	1.445
Prob(Omnibus):	0.000	Jarque-Bera (JB):	6367.334
Skew:	-2.487	Prob(JB):	0.00
Kurtosis:	11.612	Cond. No.	9.20

Figure #6: - Popularity vs. Valence Model

```
import statsmodels.api as sm
import statsmodels.formula.api as smf
regression = smf.ols(" Q('Popularity') ~ Q('Valence') ",  data=spotify).fit()
regression.summary()
```

Dep. Variable:	Q('Popularity')	R-squared:	0.000
Model:	OLS	Adj. R-squared:	-0.001
Method:	Least Squares	F-statistic:	0.001403
Date:	Fri, 02 Dec 2022	Prob (F-statistic):	0.970
Time:	02:19:24	Log-Likelihood:	-6458.3
No. Observations:	1545	AIC:	1.292e+04
Df Residuals:	1543	BIC:	1.293e+04
Df Model:	1		
Covariance Type:	nonrobust		

OLS Regression Results

	coef	std err	t	P>|t|	[0.025	0.975]
Intercept	70.1235	0.997	70.331	0.000	68.168	72.079
Q('Valence')	-0.0664	1.772	-0.037	0.970	-3.542	3.410

Omnibus:	836.551	Durbin-Watson:	1.444
Prob(Omnibus):	0.000	Jarque-Bera (JB):	6232.713
Skew:	-2.466	Prob(JB):	0.00
Kurtosis:	11.515	Cond. No.	5.62

Figure #7: Accuracy of Ensemble

```
Ensemble Accuracy: 70.63%
```

Figure #8: Classification Method Accuracies

```
Nearest Neighbors Accuracy: 18.77%
Linear SVM Accuracy: 24.27%
RBF SVM Accuracy: 9.39%
Decision Tree Accuracy: 29.77%
Random Forest Accuracy: 24.60%
Neural Net Accuracy: 27.83%
AdaBoost Accuracy: 29.13%
Naive Bayes Accuracy: 23.30%
```