Critical Thinking Assignment Module 6: Option 1

Jessica Ham

MIS581 – Capstone: Business Intelligence and Data Analytics

Colorado State University – Global Campus

Chris den Heijer

June 21, 2020

Table of Contents

I. ABSTRACT

II. INTRODUCTION

III. RESEARCH HYPOTHESIS

IV. OBJECTIVES

V. OVERVIEW OF STUDY

VI. LITERATURE REVIEW

VII. RESEARCH DESIGN

1. Methodology
2. Methods
3. Limitations
4. Ethical considerations

VIII. FINDINGS

IX. CONCLUSION

X. RECOMMENDATIONS

XI. REFERENCES

Critical Thinking Assignment Module 6: Option 1

# ABSTRACT

With the future of live music in question during the midst of a global pandemic, the prediction of which bands are most successful in a live capacity would help companies that book live shows have the best chance of turning the most profit with limited opportunity. With this in mind, I used South by Southwest and Spotify’s top streamed artists to develop a logistic regression model that will classify artists as more popular live or more popular streamed. While there is some obvious overlap in that both sets of music, the resulting model is accurate within a 57-71% confidence interval.

# INTRODUCTION

Since its inception in 1987, South by Southwest (SXSW) has strived to foster the future of entertainment and media endeavors. Inclusiveness and reaching for new things were among the founders core values (Swenson, 2019-2020). The event began as a means to shine a worldwide light on local artists. Through steady growth, Austin, Texas based South by Southwest has become a mecca for all things creative, with festivals and conventions convening each year for musicians, interactive media and film.

For South by Southwest (SXSW) and similar companies whose business is built on large scale public concerts and events, 2020 has brought cancellations and questions about the future of live music. Early estimates have the film and music industries, losing as much as $5 billion in revenue from delayed releases and cancelled performances (Hissong & Millman, 2020). This could prove conservative if you look at the bigger losses for associated industries and area businesses surrounding major events. The 2019 South by Southwest “invited more than 417,000 people to Austin last year… to bring in $356 million to the city” (Hissong & Millman, 2020).

Many experts are indeed forecasting that this is not a market that will even reopen before a vaccine becomes available and effective, an estimated 18-24 months (Sullivan, 2020). Event based companies like SXSW will need to adapt to survive the lull. As I follow the concerns for the future of artists and consider my reaction to this trying time, it raised the question: What makes a live show a live show? Is there something about particular bands or songs that make them pre-disposed to be more successful in radio or streaming play versus played live? How will SXSW decide which bands are best for a small live show and which should be reserved for streaming promotion and play at a lesser profit?

# RESEARCH HYPOTHESIS

With a clear business question based on the needs of the SXSW company, we start to look at how to go about building the model and testing its performance. The hypothesis we are testing for this model is the relationship between our dependent variable, the ‘Live’ determination, with the independent variables, those describing the musical characteristics of each song: Tempo, Energy, Danceability, Loudness, Valence, Acousticness, Speechiness, Key, and Mode (Spotify, 2020). As such, the hypothesis would be phrased as:

H1: There is a relationship between musical characteristics of a song and its popularity as a live performance.

H0: There is no relationship between musical characteristics of a song and its popularity as a live performance.

# OBJECTIVES

The model that will be built out of this hypothesis should determine if this relationship exists and measure it in such a way that we can determine with some confidence whether a song is likely to be a bigger success live than as a streaming or radio hit.

# OVERVIEW OF STUDY

This project uses Python code to parse Tweets about the 2017 SXSW festival into mentions of bands and then generates a Spotify playlist of the most popular songs of each band and a second Spotify playlist of most streamed music from the same time period to create a comparison of popular songs during the era. Attributes of these songs were extracted from Spotify API and analyzed using an R Logistic Regression to create a classification model. The model performance is tested in an ROC curve for accuracy in attempting to determine what characteristics make a song more popular live or streamed.

# LITERATURE REVIEW

Published Journal articles regarding Spotify and using music attributes to classify music or artists are sparse, at best. Similarly papers researching musical festivals and their success are rare. Searching for literature surrounding Music Festivals such as SXSW, there are a few cultural papers on the influence of festivals on society and ideas as well as what aspects add to the success of a festival beyond just music, but nothing readily accessible about the music itself.

There are a few published works on the perception of music and media platforms and their influence on society which specifically reference Spotify as well as some experiments to understand the creation of playlists and music suggestions. In 2015 Hagen wrote in Popular Music & Society about the curation of personal playlists through interviews and diaries of heavy Spotify users (Hagen, 2015) and a couple years later a cultural research paper explored gender bias in music recommendations from Spotify, where it was determined that while male artists are disproportionately represented, there is not a noticeable difference in the recommendations made to male and female listeners (Eriksson & Johansson, 2017).

Looking specifically at published and peer-reviewed literature containing models for classifying music, Herremans et al found a logistic regression outperformed the decision tree model in distinguishing classical composers, correctly identifying between Haydn, Beethoven and Bach 83% of the time based on musical features such as melodic intervals, pitch, and repeated notes (Herremans, et al, 2015). An earlier study attempted to classify accents in pop music to geographic regions with limited success at meeting the performance of a human listener (Mullensiefen, 2009). However, a simple google search for Spotify modeling shows that, in reality there are a multitude of articles similar to one by Kat Wilson that proport to “Learn Data Science with Spotify” (Wilson, 2020). Her classification model is trained to distinguish between the Clash and Jack Johnson- presumably a clear distinction to cut one’s teeth on for classification models. Other training attempts focus on playlist generation using logistic regression, random forest or k-nearest neighbors models to classify music into a genre. McIntire attempted to build his own classification algorithm of his personal taste (McIntire, 2017) and Juan De Dios Santos performed a detailed analysis comparing his taste to a friend who described his music as “boring” (De Dios Santos, 2017).

Each of these are interesting cultural statements on music and society but so far when it comes to musical attributes and predictions, we are not seeing much that is commercially viable. Now we come to the section of research into what makes a song a hit. Two separate “hit” studies, one done by students at Cornell, the other at Stanford, found random forest to be a better predictor of Billboard hits compared to logistic regression, neural network and support vector machine. Also of interest and potentially commercially significant was a study from Menten, et al, from CU Boulder who looked not only at current song popularity, but the span of more than 50 years of popular music, asking questions about both the core characteristics that have withstood the test of time, and more recent or cyclical trends in music preferences. This team found that logistic regression, in general, out performed k-Nearest Neighbors in classifying dancability from other features, and helped identify what specific features most contributed to the determination.

Overall, while music and it’s features have become almost a cornerstone in learning and comparing machine learning models, the use of this method for commercial means is still primarily limited to company proprietary algorithms which suggest music and encourage streaming users to select one model over another, but little is being done for understanding variations in what format a song is most popular and whether certain bands are more suited for a live performance, a festival format or streamed sessions. I would consider that club music would be another layer to consider, again, a different sound than a festival or live show. This paper is an attempt to begin to fill this gap in research.

# RESEARCH DESIGN

## Methodology

This process begins with creating a live band dataset. The available data is a structured scrape of Twitter feeds from 2017 which were tagged with either #SXSW or @SXSW and provided as a public data set by Chandler Nunez via Data World. Using his data, the goal is to build a Python model that can review each tweet looking for tags of band names that were performing at the 2017 event. This is an attempt at understanding the most mentioned bands of that year, under the presumption that, even if we had access to the attendance counts, venue occupancy restrictions would limit a popular show and the counts would be unlikely to capture the SXSW attendees that were wanting to but ultimately unable to attend these shows. Mentions of the bands would show a general enthusiasm for that live artist. The top mentioned performers will make up our ‘Live Band’ list. By using Leonardo Henrique’s Top Spotify Streams by year list which is available from Kaggle to gather the most popular ‘Streamed Bands’ we can start to generate a dataset which represents the distinction we hope to find.

The fastest way to develop this dataset for use in the final model is to use the Python spotipy library and my existing Spotify account (McIntire, 2017). Through a Spotify for Developers account, it is possible to leverage access to existing playlists and retrieve the API data based on the song IDs. The features of each song will guide the model classification attempts.

In order to create a model that can predict where a song falls on a spectrum between best live and best streamed, I plan to use R and logistic regression. When looking at the performance of a logistic regression model, ROC (Receiver Operating Characteristic) curves and lift charts are a means to understand where to set the cut off between the two choices in your model. ROC curves are generally used in instances where getting a false positive but catching all the true positives, may be more important than the cost of testing everyone or the cost of treatments, etc. Conversely a lift chart works best when the question of cut off has little to do with catching every positive response and more to do with where you can invest the least into getting the best gains (Jurczyk, 2019). In our case, there is very little cost associated with testing any and all songs for performance in our model, but a high cost to putting on a live show that does not perform well. In this way, a ROC curve is a better measurement of performance for our model. This test as well as a confusion matrix can give us an idea of how well the model performs.

## Methods

This project uses Python code language run through Py Charm 2018.3 (Edu). Copyright © 2010-2020 JetBrains s.r.o. to generate all iterations of files that culminate in the final csv dataset which was used in the logistic model using R version 3.5.2. Copyright © 2018 R Foundation for Statistical Computing and RStudio version 1.2.1335. Copyright© 2018 RStudio, Inc. Figure outputs in this paper is in R Markdown format.

The code for separating band mentions from tweets searches first for the official band name, as listed on the SXSW line up. Secondly, it accounts for the common lack of grammar and capitalization of today’s social media removing any leading words such as A or The, and searching again for the band without case sensitivity. Adding to that, it searches for what would be a likely twitter handle or hashtag for the band by concatenating the band name into one string. Finally, operating under the assumption that the person publishing the tweet was unlikely to type out a band such as OG Ron C and The Chopstars in its entirety, the code pulls those bands with an ampersand or ‘with’ text apart into separate names, searching tweets in this example first for OG Ron C and then again for Chopstars. The results of each iteration are summed together for a total count of mentions for that band. The resulting list was exported to excel and used to develop a playlist of songs from these bands, which, in turn, was again loaded into Python where spotipy was used to pull the features of each song for analysis. This final data frame was exported from Python into a csv file that was imported into R where the details of the features could be reviewed and a regression model developed.

## Limitations

One challenge in this project came with my intention to use natural language processing (NLP) to review the tweets. This is not a tool with which I have strong familiarity, and while there were a good number of resources that detailed a simple tokenize and count words feature, I was not able to find information on how to instead search for a particular series of words and in several ways. The Avett Brothers were one band who played shows at the 2017 SXSW. There were not any tweets where the band name was fully typed out, as shown above, rather it was just Avett Brothers or their twitter handle which is @avettbrothers. Generating those variations on band names became more realistic to do using lists and iterations versus NLP. Another example that created a challenge however were band names which were common names, such as the band LIFE or the band Her’s (in a year where Women in Tech was the keynote presentation at the tech conference). To resolve these overstatements of some bands required a hand review of the top bands. In cases where I could disqualify enough tweets to put mentions of that band as less than the others on the top 100 list, said band was removed and the others were moved up, until there were 100 bands that seemed to have valid mentions within Twitter. Finally, when using Spotify to check these bands and their most popular song, there were several which I was not able to find on Spotify. The result was a list of 94 top bands, which was matched by 94 bands in the streaming list, using all 50 from 2017 and a random selection of top artists from 2016 and 2018 to fill out the list.

## Ethical Considerations

Fortunately this topic is not rife with ethical considerations. The datasets outlined above are bereft of any real personal data. The SXSW tweet file does contain the handle of the person who posted the tweet, and, depending on that individuals privacy settings, it is possible that those handles could be used to get more information. Arguably however, this is nothing that isn’t already accessible from Twitter directly, as the scrape was performed on public data only. The same goes for the Spotify downloads. Even with a Spotify user name, a developer is only able to access information about the types of music and playlists that the user prefers. I used my personal Spotify account to generate the playlists used. There have been a large number of harassment claims in the Spotify world, but the reality is that they center around someone who already knows something else about the user, such as an ex email stalking the user about their song selection. Most complaints against Spotify are to do with the rampant hacking of accounts. While this sounds quite dramatic, the reality is that it is more of a nuisance than a risk of personal data exposure.

# FINDINGS

The dimensions of the final data set are 188 songs with 16 feature rows. Some of this is metadata, such as song and artist name, but the independent variables can be described as shown in Figure 1 below.

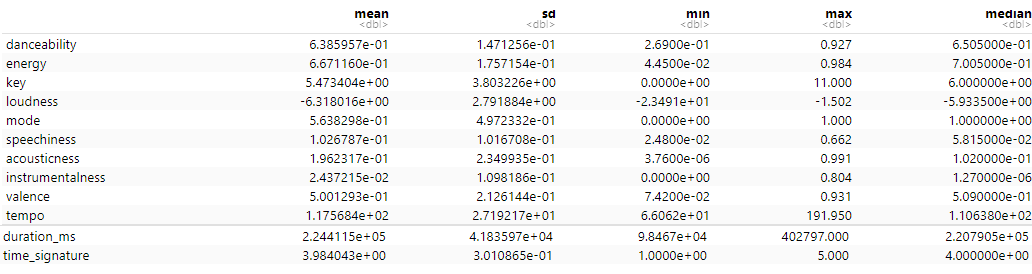


Figure 1 Explore Feature Data

As per the design of our dataset, the songs are equally distributed between live and streaming categories, depending on the source of their popularity in this study (see Figure 2). A version of this distinction becomes our dependent variable.

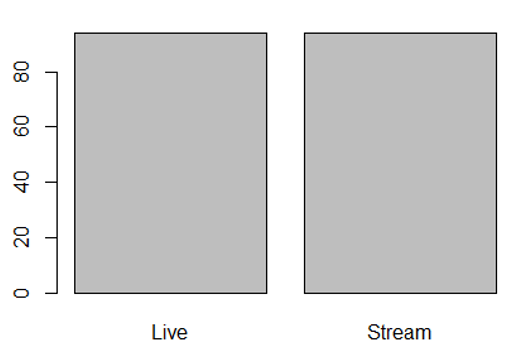


Figure 2 Distribution of Dependent Variable

Using the variables above as the independent features and the Live classification, now converted to a binomial (1=Live, 0=Stream), a logistic regression model is coded into R of which the results are detailed in Figure 3.

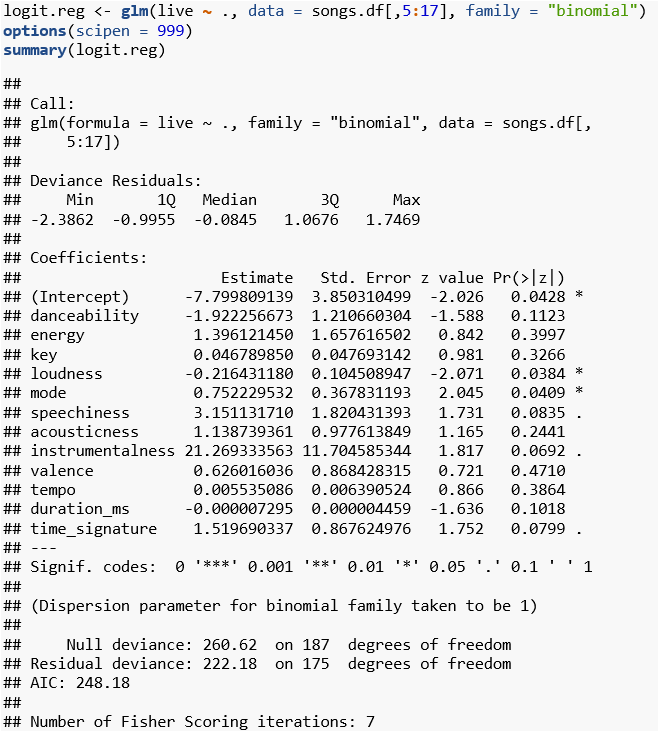


Figure 3 Logistic Regression Summary

# Based on this output the regression equation would be:

# Logit(Live=Yes)= -7.7998 + 21.2693 Instrumentalness + 3.1511 Speechiness - 1.9223 Danceability + 1.5197 Time Signature + 1.3961 Energy + 1.1387 Acousticness + 0.7522 Mode + 0.6260 Valence - 0.2164 Loudness + 0.0468 Key + 0.0055 Tempo

To understand the performance of this model, it can first be coded into a confusion matrix, with, for argument’s sake, a parameter at 0.5, which would classify those above that as performing well live, and below as best streamed. The results are shown in Figure 4.

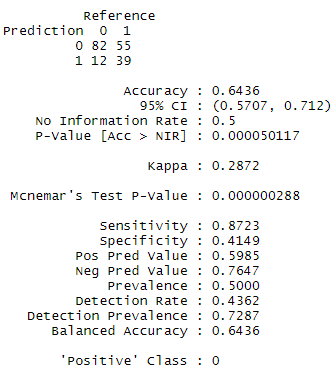


Figure 4 0.5 Confusion Matrix

The first item of note when we run a confusion matrix is to see that both p-Value statistics are less than 0.0001, which means our model is significant and we can reject the null hypothesis. The features of a song do have an effect on its success as a live performance. If we continue to set the decision point at 0.5, the model is accurate about 64% of the time. However, considering the limitations of live shows in this cultural climate, a failed live show (one that is performed to less than a sell-out audience) could, for a company such as SXSW, mean missed profits in a time where opportunity for any profit are slim on the ground. We want to try and minimize the number of times that the model guesses a top live show and is wrong. At that decision point, we have 12 failed shows. To determine at what point we can assure the fewest shows that are unlikely to sell out, we use the code in Figure 5 to generate an ROC Curve (Figure 6).

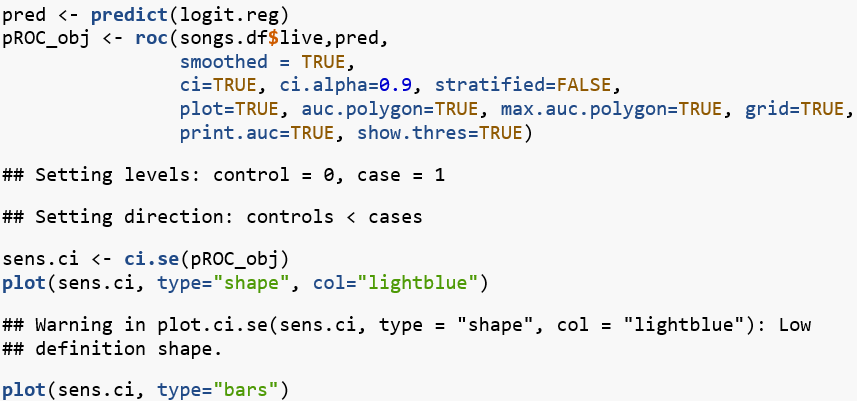


Figure 5 ROC Curve Code Block

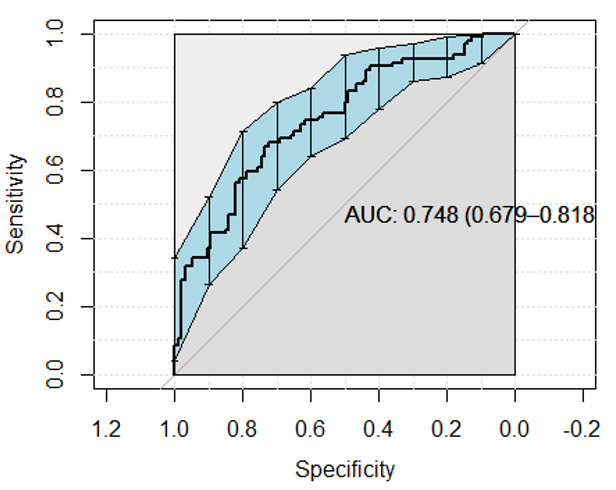


Figure 6 ROC Curve

If the model does not remain fixed at a 0.5 decision point, it has the opportunity to perform with as much as 74.8% accuracy.

# CONCLUSION

To get as few failed shows as possible (improved false accuracy rate) we need a greater specificity, or a higher decision point. While this may hurt the accuracy of the test, it would, at least, ensure the fewest possible failed shows. Figure 7 shows the array of decision points and their accuracy rates.

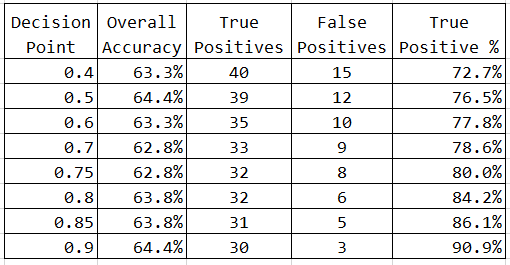


Figure Decision Point Table

# One thing we can see with this view is that, at some point, the accuracy actually comes back up overall, despite the decrease of missed positives, since we are decreasing the false negatives as well.

# RECOMMENDATIONS

It is interesting to me to again note the balance between cost benefit analysis and also with privacy and business usage in data. As mentioned in the conclusions, without changing anything about the model, we have set the business decision makers up to determine how big of a risk a failed show might be. To do that, we should define failure for the company. The reality is that likely a show from a popular radio or streaming song is probably going to be fine as a live performance, it’s just the goal of the model was to identify the best performances- those that got the most buzz and would sell out the available venues. I would expect that in this sense, some false positives in the model would be acceptable, the question is how many.

The other question here would be how many shows you are going to be able to book. SXSW calls in over 2,000 artists in a normal year at present. Since we are already planning to put on fewer shows, it is unlikely that we are missing revenue opportunities by skipping false negative response in the model, or a show that might have been good. If looking at accuracy versus specificity to get the fewest failed shows, I would recommend that 0.9 decision point, still providing a 64% accuracy rate, but only 3 failed shows.

As mentioned, data availability could also have an impact on the model. I would contend that the current model is built entirely on the merits of the song to make a prediction, when reality is there are other factors at play to generate excitement about a performance. Even looking at how I determined the best bands, I would argue that tweets and social media mentions are a great indicator, but attendance figures that are proprietary to SXSW would add to the accuracy of determining best shows. Were I building out a more permanent model for SXSW, it would likely also include information about sales and releases of albums, reviews, mentions in social media around these events and other similar data that likely does factor in to a popular live show. This sort of detailed information would be unlikely to be publicly available and would require more security and permissions to access.

# REFERENCES

De Dios Santos, Juan (2017). Is my Spotify music boring? An analysis involving music, data

and machine learning. Retrieved from: <https://towardsdatascience.com/is-my-spotify-music-boring-an-analysis-involving-music-data-and-machine-learning-47550ae931de>.

Eriksson, Maria and Johansson, Anna (2017). Tracking Gendered Streams. *Culture Unbound:*

*Journal of Current Cultural Research 9:2*, p 163-183. Retrieved from: <http://www.cultureunbound.ep.liu.se/article.asp?DOI=10.3384/cu.2000.1525.1792163>.

Georgieva, Elena, Suta, Marcella and Burton, Nicholas (2018). HitPredict: Predicting Billboard

Hits Using Spotify Data. Stanford University. Retrieved from: <http://cs229.stanford.edu/proj2018/poster/16.pdf>.

Hagen, Anja Nylund (2015). The Playlist Experience: Personal Playlists in Music Streaming

Services. *Popular Music & Society 38:5*, p625-645. Retrieved from: <http://eds.b.ebscohost.com.csuglobal.idm.oclc.org/eds/detail/detail?vid=0&sid=3c8726fe-d46c-4032-bca7-3ceb80ca0e7e%40pdc-v-sessmgr03&bdata=JnNpdGU9ZWRzLWxpdmU%3d#AN=109173610&db=aph>.

Henrique, Leonardo (2019). Top Spotify songs from 2010-2019 – by year. Dataset. Retrieved

from: <https://www.kaggle.com/leonardopena/top-spotify-songs-from-20102019-by-year>.

Herremans, Dorien, Sorenson, Kenneth and Martens, David (2015). Classification and

Generation of Composer-Specific Music Using Global Feature Models and Variable Neighborhood Search. *Computer Music Journal 39:3*, p71-91. Retrieved from: <http://eds.a.ebscohost.com.csuglobal.idm.oclc.org/eds/detail/detail?vid=0&sid=74905d80-a7f0-486e-bca8-a478f362a72f%40sessionmgr4006&bdata=JnNpdGU9ZWRzLWxpdmU%3d#db=cph&AN=109373680>.

Hissong, Samantha & Millman, Ethan (2020). ‘Everything is in Chaos’: The Concert Business

Stands to Lose Billions from Coronavirus. Retrieved from: <https://www.rollingstone.com/music/music-features/live-music-concerts-coronavirus-risks-losses-965482/>.

McIntire, George (2017). A Machine Learning Deep Dive into My Spotify Data. Retrieved

from: <https://opendatascience.com/a-machine-learning-deep-dive-into-my-spotify-data/>.

Menten, Matthew & Ng, Kieren & O'Rourke, Tanner & Holmes, Rourke. (2018). Temporal

Trends in Music Popularity - A Quantitative analysis of Spotify API data. 10.13140/RG.2.2.11551.71843. Retrieved from: <https://www.researchgate.net/publication/334987891_Temporal_Trends_in_Music_Popularity_-_A_Quantitative_analysis_of_Spotify_API_data>.

Middlebrook, Kai and Sheik, Kian (2019). Song Hit Prediction: Predicting Billboard Hits Using

Spotify Data, Cornell University. Retrieved from: <https://arxiv.org/abs/1908.08609>.

Mullensiefen, Daniel, Pfleiderer, Martin and Frieler, Klaus (2009). The Perception of Accents in

Pop Music Melodies. *Journal of New Music Research 38:1*, p19-44. Retrieved from: <http://eds.b.ebscohost.com.csuglobal.idm.oclc.org/eds/detail/detail?vid=0&sid=ed4625b8-5163-401b-b57d-1348539168ed%40pdc-v-sessmgr03&bdata=JnNpdGU9ZWRzLWxpdmU%3d#AN=44651498&db=aph>.

Nunez, Chandler (2017). 2017 SXSW Twitter Traffic. Dataset. Retrieved from:

<https://data.world/chanalytics/2017-sxsw-twitter-traffic>.

Spotify (2020). Spotify for Developers: Audio Features Object. Retrieved from:

[https://developer.spotify.com/documentation/web-api/reference/tracks/get-audio-features/](https://developer.spotify.com/documentation/web-api/reference/tracks/get-audio-).

Sullivan, Jim (2020). What Will the Future of Concerts Look Like after the Pandemic?

Retrieved from: [https://www.wbur.org/artery/2020/05/04/coronavirus-pandemic-music-industry-concerts](https://www.wbur.org/artery/2020/05/04/coronavirus-pandemic-music-).

Swenson, Roland (2019-2020). SXSW History Intro. Retrieved from:

<https://www.sxsw.com/about/history/>.

Wilson, Kat (2020). How to do Logistic Regression in Python and R. Retrieved from:

<https://towardsdatascience.com/how-to-do-logistic-regression-in-python-and-r-b7225ea678e8>.