Gregorio Del Rio

ECON 494-02R-FA20

Professor Steven Levkoff

October 25, 2020

Spotify Data

Source

The dataset used for this project is "Spotify Dataset 1921-2020, 160k+ Tracks." This dataset was downloaded from Kaggle and uploaded by Yamaç Eren Ay. Since I downloaded this dataset on Oct. 3, 2020, Yamaç Eren Ay has updated it to have more numerical, categorical, and dummy variables. As a result, I am using less variables than would currently be available and observed.

Executive Summary

Over 160,000 songs were collected from Spotify Web API. Each observation has 14 variables connected to it. There are 4 categorical variables and 10 quantitative variables. Of these fourteen variables, I will not use 3 categorical variables; artist_name, track_name, and track_id. I will also not use 1 quantitative variable; liveness. Removing these 4 variables leaves me with 10 variables to analyze this dataset.

Categorical variables

Genre: Twenty-six genres ranging from A Capella to World.

Quantitative Variables[i]

Popularity: Measures ranging from 0 – 100. Lower values indicate a less played song and higher values indicating the most played songs.

Acousticness: A confidence measure from 0.0 to 1.0 of whether the track is acoustic. 1.0 represents high confidence the track is acoustic.

Danceability: Describes how suitable a track is for dancing based on a combination of musical elements including tempo, rhythm stability, beat strength, and overall regularity. A value of 0.0 is least danceable and 1.0 is most danceable.

Duration-min: The duration of the track in minutes. Converted from milliseconds.

Energy: A measure from 0.0 to 1.0 and represents a perceptual measure of intensity and activity. Typically, energetic tracks feel fast, loud, and noisy. For example, death metal has high energy, while a Bach prelude scores low on the scale.

Instrumentalness: Predicts whether a track contains no vocals. "Ooh" and "aah" sounds are treated as instrumental in this context. Rap or spoken word tracks are clearly "vocal". The closer the instrumentalness value is to 1.0, the greater likelihood the track contains no vocal content. A value of 0.0 is least instrumental and 1.0 is most instrumental.

Loudness: The overall loudness of a track in decibels (dB). Loudness values are averaged across the entire track and are useful for comparing relative loudness of tracks. Values typical range between -60 and 0 db.

Speechiness: Detects the presence of spoken words in a track. The more exclusively speech-like the recording (e.g. talk show, audio book, poetry), the closer to 1.0 the attribute value. Values above 0.66 describe tracks that are probably made entirely of spoken words. Values between 0.33 and 0.66 describe tracks that may contain both music and speech, either in sections or layered, including such cases as rap music. Values below 0.33 most likely represent music and other non-speech-like tracks.

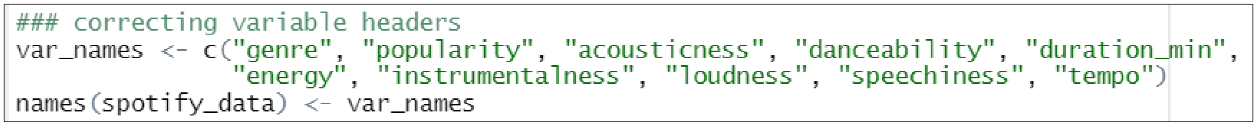
Tempo: The overall estimated tempo of a track in beats per minute (BPM). In musical terminology, tempo is the speed or pace of a given piece and derives directly from the average beat duration.

Goals

1. I would like to see the makeup of this data, so first I will see how many songs are in each genre and what percentage each genre makes of the dataset.
2. To see the popularity distribution of each genre, I'll create a histogram.
3. Build a correlation between all quantitative variables to gauge positive and negative relationships.
4. To see which genres are more popular than others. To do this I will need to take the average popularity of each genre and order them in a bar chart.
5. Boxplot genre by quantitative variables to see how genres compare to each other by each variable.
6. Run a linear regression with the variables who have the strongest relationship to popularity. See if there is a relationship which could explain what leads to songs having a higher popularity.

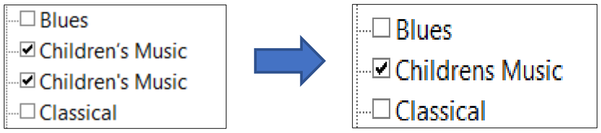
Cleaning Data for Analysis

1. After downloading this dataset, I opened the csv.file in excel and remove the variables; artist_name, track_name, track_id, and liveness.
2. When first importing this data into R Studio, the genre header was being written as "Ï…genre." To ensure all the variable headers were correct, I created a vector with the appropriate text/label for each variable. Using the names() function, I corrected all the variable headers.



```
### correcting variable headers
var_names <- c("genre", "popularity", "acousticness", "danceability", "duration_min",
               "energy", "instrumentalness", "loudness", "speechiness", "tempo")
names(spotify_data) <- var_names
```

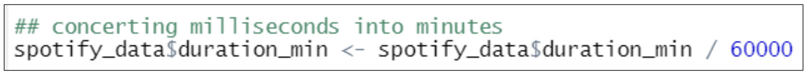
3. Within the genre variable, Childrens Music had two spelling; Children's Music and Children's Music. I relabeled both to be Childrens Music. I performed this relabeling within excel.



□ Blues
☑ Children's Music
☑ Children's Music
□ Classical

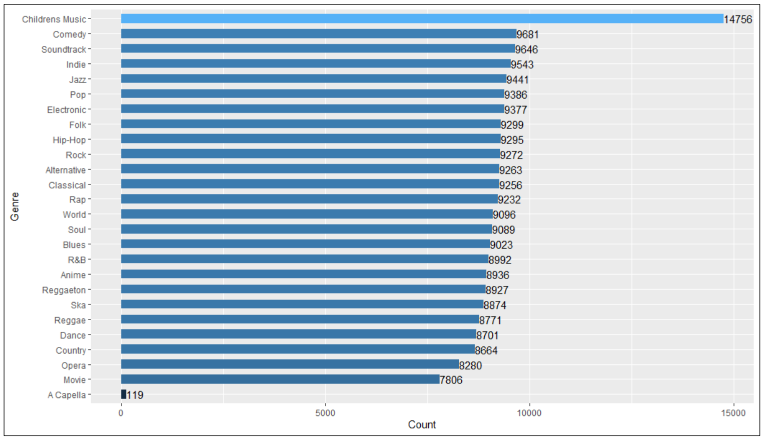
□ Blues
☑ Childrens Music
□ Classical

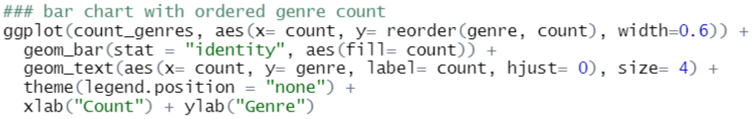
4. Duration_min was originally in milliseconds (1000 milliseconds = 1 second). I divided each observation's duration_ms by 60,000 to convert milliseconds into minutes. I then relabeled duration_ms to duration_min.



```
## concerting milliseconds into minutes
spotify_data$duration_min <- spotify_data$duration_min / 60000
```

1. Amount of Songs in Each Genre and Percentage of Dataset

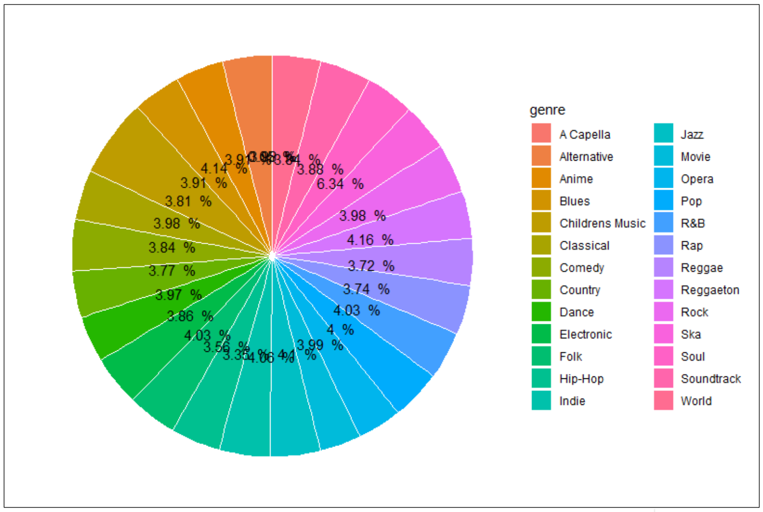


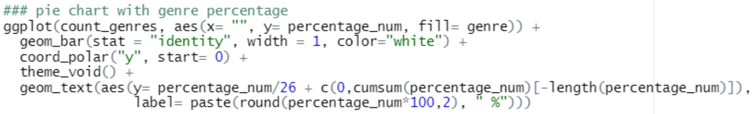


```
### bar chart with ordered genre count
ggplot(count_genres, aes(x= count, y= reorder(genre, count), width=0.6)) +
  geom_bar(stat = "identity", aes(fill= count)) +
  geom_text(aes(x= count, y= genre, label= count, hjust= 0), size= 4) +
  theme(legend.position = "none") +
  xlab("Count") + ylab("Genre")
```

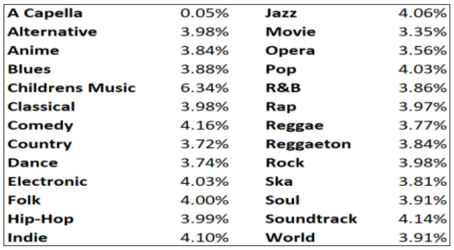
This bar chart shows us how many songs are in each genre. We can see that besides Childrens Music and A Capella, the other 24 genres are close to evenly distributed. With the amount in each genre ranging from 7,806 to 9,681. The total amount of songs (observations) can be summed up to 232,725.

- Close to uniform distribution
- 232,725 total observations





```
### pie chart with genre percentage
ggplot(count_genres, aes(x= "", y= percentage_num, fill= genre)) +
  geom_bar(stat = "identity", width = 1, color="white") +
  coord_polar("y", start= 0) +
  theme_void() +
  geom_text(aes(y= percentage_num/26 + c(0,cumsum(percentage_num)[-length(percentage_num)]),
                label= paste(round(percentage_num*100,2), " %")))
```

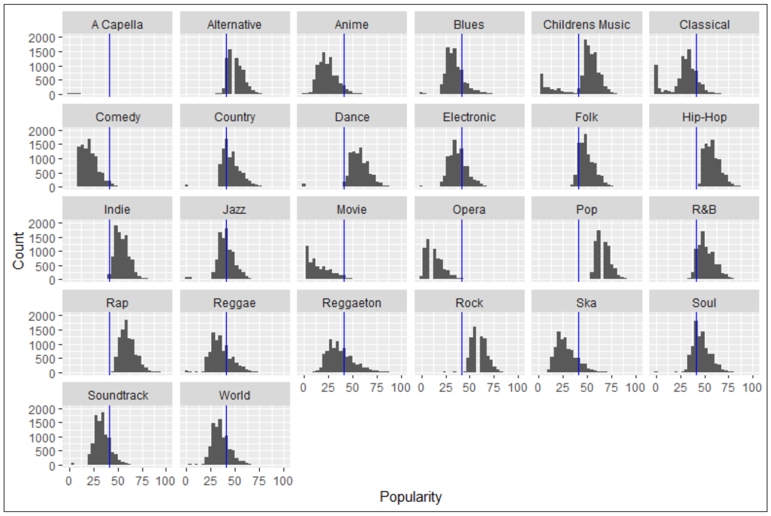


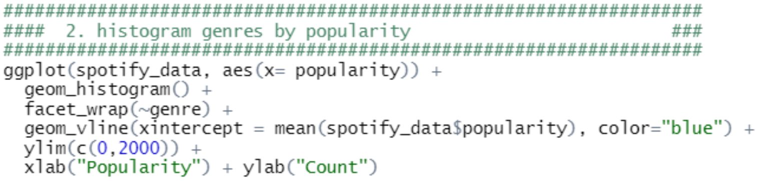
A Capella	0.05%	**Jazz**	4.06%
Alternative	3.98%	**Movie**	3.35%
Anime	3.84%	**Opera**	3.56%
Blues	3.88%	**Pop**	4.03%
Childrens Music	6.34%	**R&B**	3.86%
Classical	3.98%	**Rap**	3.97%
Comedy	4.16%	**Reggae**	3.77%
Country	3.72%	**Reggaeton**	3.84%
Dance	3.74%	**Rock**	3.98%
Electronic	4.03%	**Ska**	3.81%
Folk	4.00%	**Soul**	3.91%
Hip-Hop	3.99%	**Soundtrack**	4.14%
Indie	4.10%	**World**	3.91%

Since we can see from the bar char that the count of genres is close to uniformly distributed, it's expected to see the percentage of each genre to the whole data is close to uniformly distributed as well.

- Genre Average %: 3.85%
- Median: 3.94%
- Low: 0.05% (A Capella)
- High: 6.34% (Childrens Music)

2. Histogram of Each Genre by its Popularity





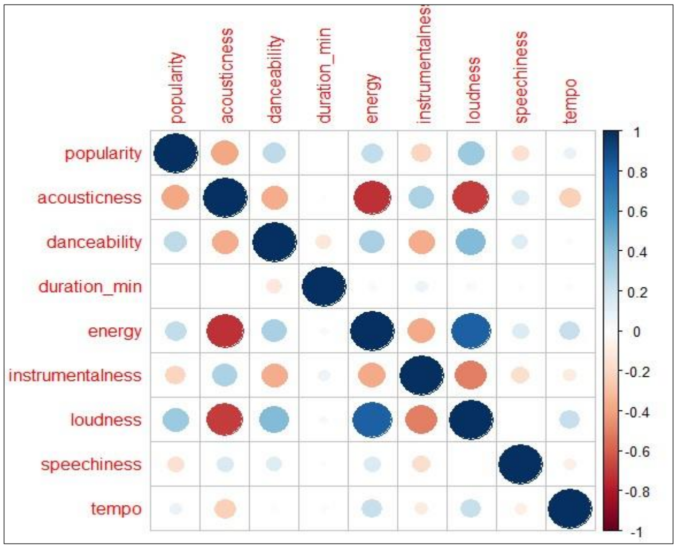
```
##########################################################################
####  2. histogram genres by popularity                               ###
##########################################################################
ggplot(spotify_data, aes(x= popularity)) +
  geom_histogram() +
  facet_wrap(~genre) +
  geom_vline(xintercept = mean(spotify_data$popularity), color="blue") +
  ylim(c(0,2000)) +
  xlab("Popularity") + ylab("Count")
```

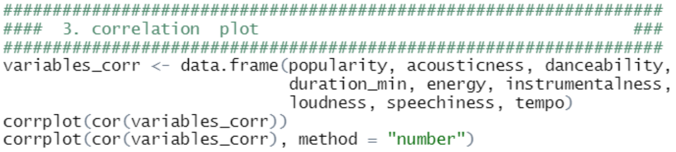
In order to see the distribution of each genre, I facet_wrapped each genre with the x-axis being popularity. The results are histograms showing the spread of popularity rating within each genre.

These histograms show most genres do not have a normal distribution. Additionally, based on each genre mean, most genres have significant skewness.

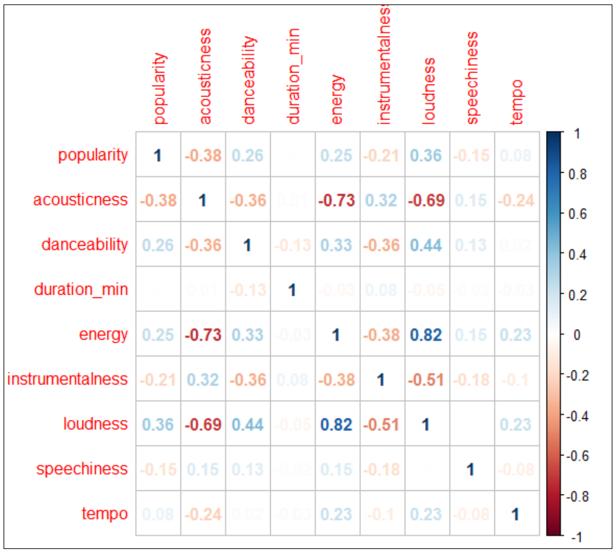
The genres which have the closest representation of a normal distribution are Electronic, Folk, Indie, Jazz, R&B, and Soul.

3. Column Variables Correlation





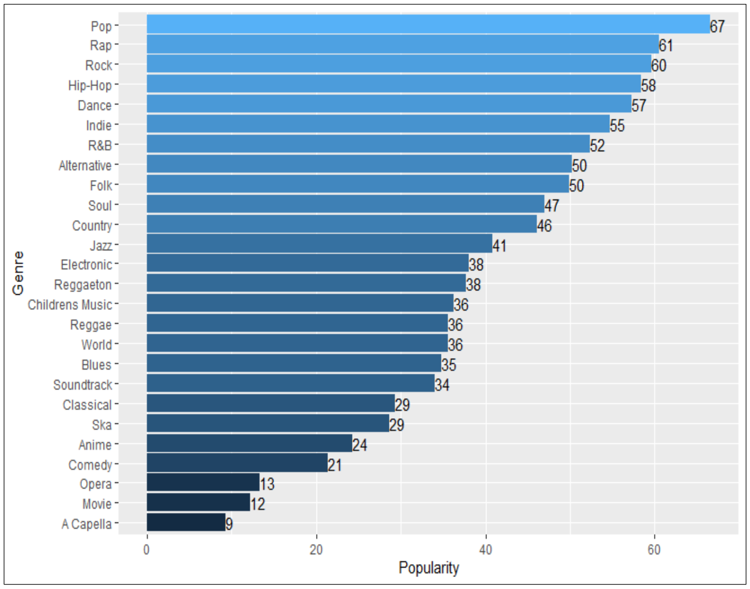
```
##########################################################################
####  3. correlation plot                                             ###
##########################################################################
variables_corr <- data.frame(popularity, acousticness, danceability,
                             duration_min, energy, instrumentalness,
                             loudness, speechiness, tempo)
corrplot(cor(variables_corr))
corrplot(cor(variables_corr), method = "number")
```

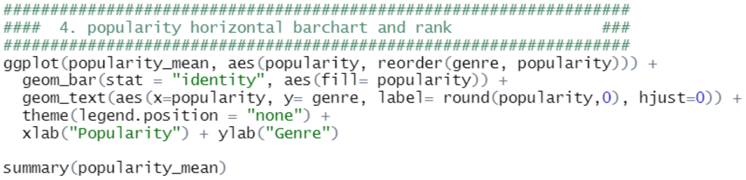


	popularity	acousticness	danceability	duration_min	energy	instrumentalnes	loudness	speechiness	tempo
popularity	1	-0.38	0.26	0	0.25	-0.21	0.36	-0.15	0.08
acousticness	-0.38	1	-0.36	0.01	-0.73	0.32	-0.69	0.15	-0.24
danceability	0.26	-0.36	1	-0.13	0.33	-0.36	0.44	0.13	0.02
duration_min	0	0.01	-0.13	1	-0.03	0.08	-0.05	-0.02	-0.03
energy	0.25	-0.73	0.33	-0.03	1	-0.38	0.82	0.15	0.23
instrumentalness	-0.21	0.32	-0.36	0.08	-0.38	1	-0.51	-0.18	-0.1
loudness	0.36	-0.69	0.44	-0.05	0.82	-0.51	1	0	0.23
speechiness	-0.15	0.15	0.13	-0.02	0.15	-0.18	0	1	-0.08
tempo	0.08	-0.24	0.02	-0.03	0.23	-0.1	0.23	-0.08	1

Looking at popularity in this correlation plot, we can see it doesn't have a strong relationship with any of the quantitative variables. The strongest appears to be acousticness with a negative correlation of -0.38.

4. Popularity Horizontal Bar-chart and Rank



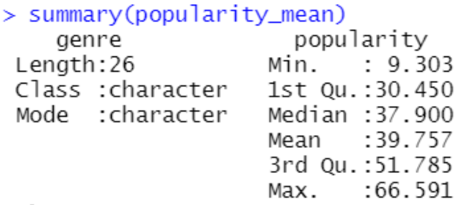


```
########################################################################
####  4. popularity horizontal barchart and rank                    ###
########################################################################
ggplot(popularity_mean, aes(popularity, reorder(genre, popularity))) +
  geom_bar(stat = "identity", aes(fill= popularity)) +
  geom_text(aes(x=popularity, y= genre, label= round(popularity,0), hjust=0)) +
  theme(legend.position = "none") +
  xlab("Popularity") + ylab("Genre")

summary(popularity_mean)
```

In order to see which genres had the highest average popularity, I created a popularity_mean variable. With it, I can then build a bar chart which reorders the genres by average popularity.

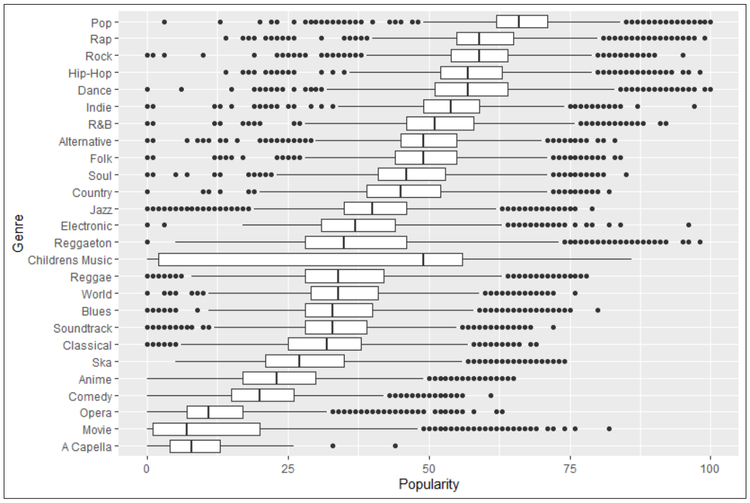
We see only Pop, Rap, and Rock are the only variable with average popularity of 60 or higher.

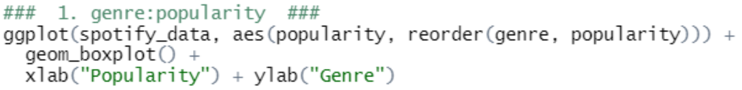


```
> summary(popularity_mean)
    genre             popularity
 Length:26          Min.   : 9.303
 Class :character   1st Qu.:30.450
 Mode  :character   Median :37.900
                    Mean   :39.757
                    3rd Qu.:51.785
                    Max.   :66.591
```

5. Boxplot Genre to Other Column Variables

Genre by Popularity





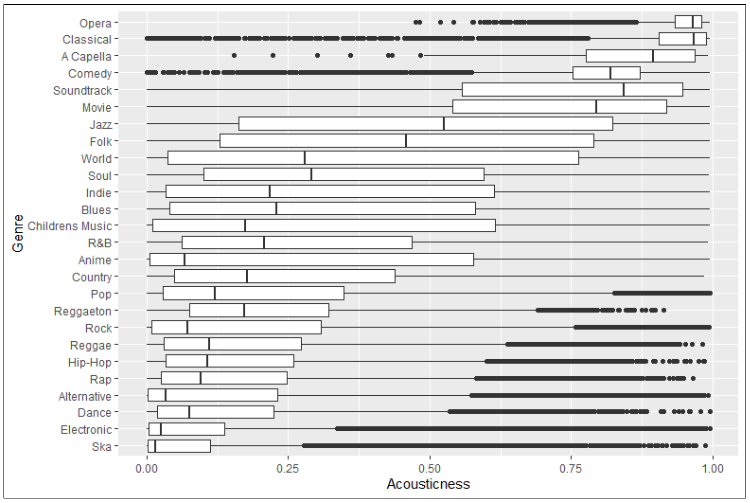
```
###  1. genre:popularity  ###
ggplot(spotify_data, aes(popularity, reorder(genre, popularity))) +
  geom_boxplot() +
  xlab("Popularity") + ylab("Genre")
```

This boxplot mirrors the above bar chart of genre by popularity. We can see the order of genre with the highest average popularity to lowest is the same as the bar chart above.

This boxplot visually shows the spread of popularity rating within each genre.

We can see genres such as Pop and Rap have a smaller spread than Childrens Music and Reggaeton.

Genre by Acousticness

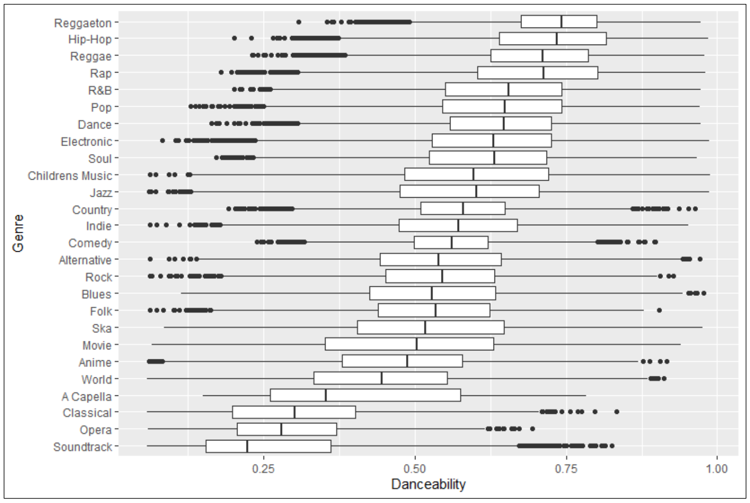


```
### 2. genre:acousticness ###
ggplot(spotify_data, aes(x=acousticness, y= reorder(genre, acousticness))) +
  geom_boxplot() +
  xlab("Acousticness") + ylab("Genre")
```

Opera, Classical, and A Capella are the highest rated by acousticness, which makes sense since these types of music don't tend to be amplified by electronics. Where genres which are heavy amplified by electronics such as Electronic and Dance have the lowest acousticness ratings.

```
> summary(acousticness_mean)
    genre              acousticness
 Length:26          Min.   :0.09973
 Class :character   1st Qu.:0.18846
 Mode  :character   Median :0.30417
                    Mean   :0.38839
                    3rd Qu.:0.49051
                    Max.   :0.94520
```

Genre by Danceability

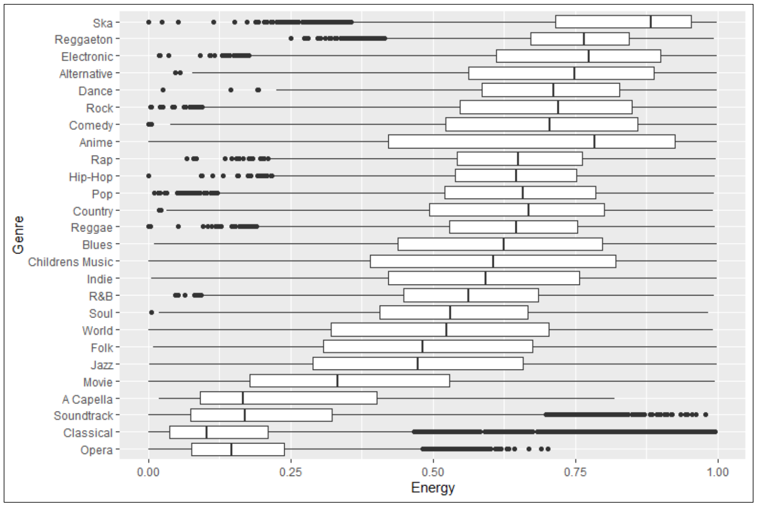


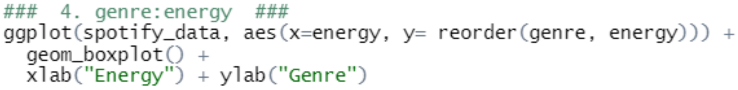
```
### 3. genre:danceability ###
ggplot(spotify_data, aes(x=danceability, y= reorder(genre, danceability))) +
  geom_boxplot() +
  xlab("Danceability") + ylab("Genre")
```

We can see the order of genres based on Danceability is almost the reverse of Acousticness. This may indicate genres which are amplified by electronics tend to be more danceable.

```
> summary(danceability_mean)
    genre              danceability
 Length:26          Min.   :0.2656
 Class :character   1st Qu.:0.5001
 Mode  :character   Median :0.5629
                    Mean   :0.5475
                    3rd Qu.:0.6335
                    Max.   :0.7313
```

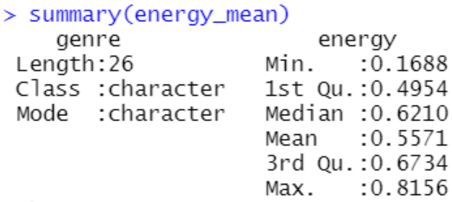
Genre by Energy





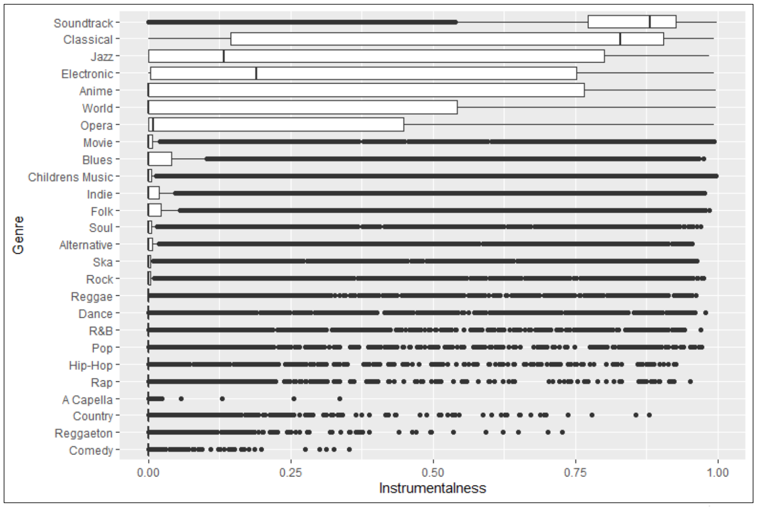
```
###  4. genre:energy  ###
ggplot(spotify_data, aes(x=energy, y= reorder(genre, energy))) +
  geom_boxplot() +
  xlab("Energy") + ylab("Genre")
```

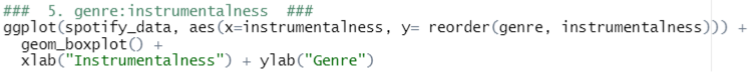
Without surprise, the top 10 of Energy and Danceability share the genres of Reggaeton, Hip-Hop, Reggae, Dance, and Electronic. Since Energy is a measurement of fast, loud, and noisy, songs which are danceable will tend to be highly rated in energy. Unlike Opera, Classical, Soundtrack, and A Capella.



```
> summary(energy_mean)
    genre                energy      
 Length:26          Min.   :0.1688  
 Class :character   1st Qu.:0.4954  
 Mode  :character   Median :0.6210  
                    Mean   :0.5571  
                    3rd Qu.:0.6734  
                    Max.   :0.8156  
```

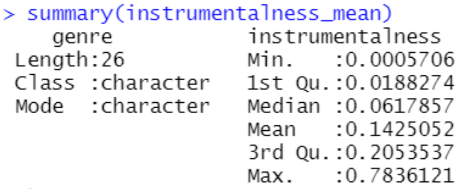
Genre by Instrumentalness





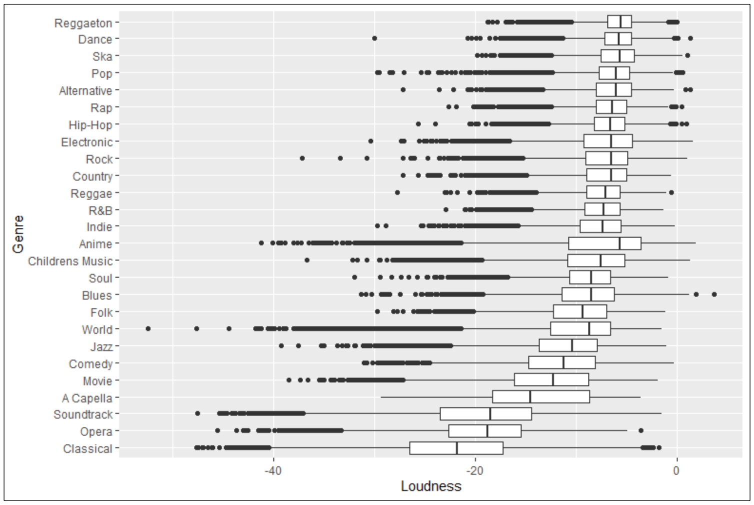
```
###  5. genre:instrumentalness  ###
ggplot(spotify_data, aes(x=instrumentalness, y= reorder(genre, instrumentalness))) +
  geom_boxplot() +
  xlab("Instrumentalness") + ylab("Genre")
```

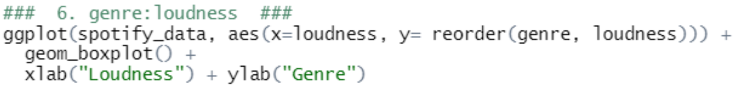
We can see the genres which tend to have little to no vocals are the highest rated in instrumentalness. It's also clear genres which are mostly vocals have extremely low rating in instrumentalness.



```
> summary(instrumentalness_mean)
    genre           instrumentalness   
 Length:26          Min.   :0.0005706  
 Class :character   1st Qu.:0.0188274  
 Mode  :character   Median :0.0617857  
                    Mean   :0.1425052  
                    3rd Qu.:0.2053537  
                    Max.   :0.7836121  
```

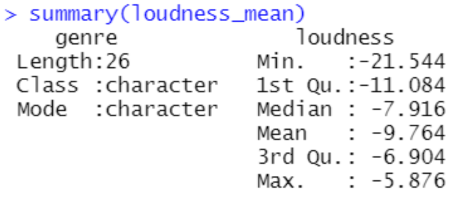
Genre by Loudness





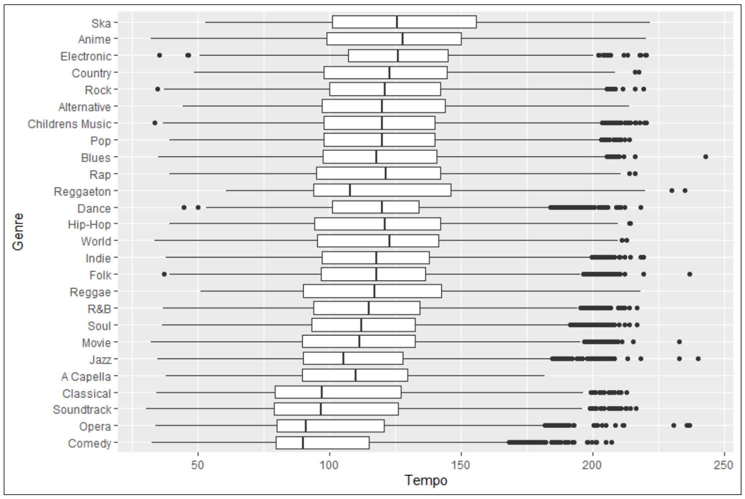
```
###  6. genre:loudness  ###
ggplot(spotify_data, aes(x=loudness, y= reorder(genre, loudness))) +
  geom_boxplot() +
  xlab("Loudness") + ylab("Genre")
```

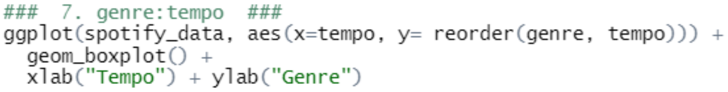
It seems the top 10 genres are closely packed by Loudness. The bottom 4 genres appear to be the only ones which are significantly louder than the rest. Additionally, the bottom 4 genres also are rated high instrumentalness.



```
> summary(loudness_mean)
    genre               loudness
 Length:26          Min.   :-21.544
 Class :character   1st Qu.:-11.084
 Mode  :character   Median : -7.916
                    Mean   : -9.764
                    3rd Qu.: -6.904
                    Max.   : -5.876
```

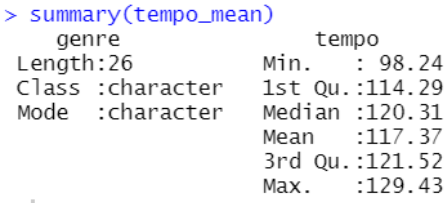
Genre by Tempo





```
###  7. genre:tempo  ###
ggplot(spotify_data, aes(x=tempo, y= reorder(genre, tempo))) +
  geom_boxplot() +
  xlab("Tempo") + ylab("Genre")
```

Interestingly, the range between the Min and Max of average Tempo is less than I think most would expect. With a Max of 123.43, Min of 98.24, and a Mean of 117.37, the average Tempo is closely packed.

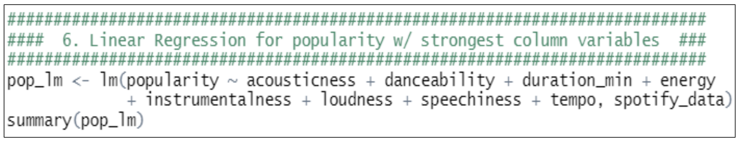


```
> summary(tempo_mean)
    genre               tempo
 Length:26          Min.   : 98.24
 Class :character   1st Qu.:114.29
 Mode  :character   Median :120.31
                    Mean   :117.37
                    3rd Qu.:121.52
                    Max.   :129.43
```

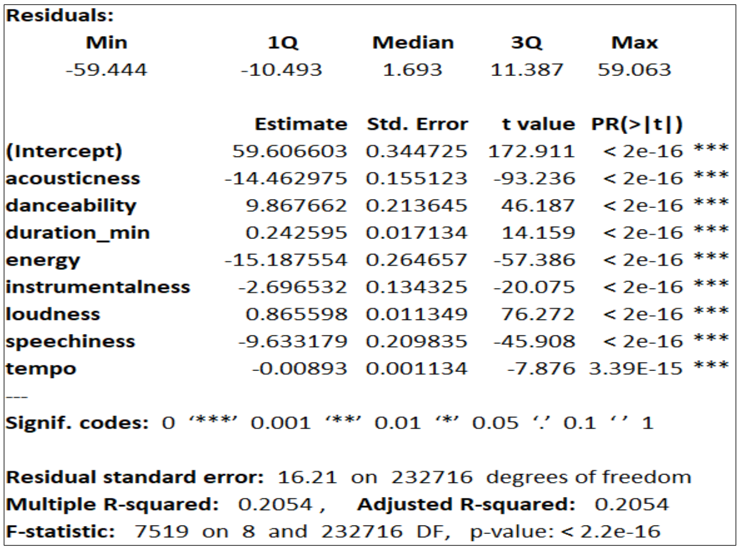
6. Linear Regression Analysis for Popularity

Spotify pays its content creators based on how many listens their content receives. The Popularity variable indicates which songs are listened to more compared to other songs. Knowing which quantitative variables leads to higher popularity rankings would assist content creators to make more popular songs, increasing their expected income.

In order to see the relationships between quantitative variables to popularity, I'll run a linear regression of acousticness, danceability, duration_min, energy, instrumentalness, loudness, speechiness, and tempo's effect on popularity.



```
##############################################################################
####  6. Linear Regression for popularity w/ strongest column variables  ###
##############################################################################
pop_lm <- lm(popularity ~ acousticness + danceability + duration_min + energy
             + instrumentalness + loudness + speechiness + tempo, spotify_data)
summary(pop_lm)
```



```
Residuals:
      Min            1Q       Median       3Q       Max
   -59.444       -10.493       1.693    11.387    59.063

                      Estimate   Std. Error   t value   PR(>|t|)
(Intercept)          59.606603   0.344725    172.911   < 2e-16 ***
acousticness        -14.462975   0.155123    -93.236   < 2e-16 ***
danceability          9.867662   0.213645     46.187   < 2e-16 ***
duration_min          0.242595   0.017134     14.159   < 2e-16 ***
energy              -15.187554   0.264657    -57.386   < 2e-16 ***
instrumentalness     -2.696532   0.134325    -20.075   < 2e-16 ***
loudness              0.865598   0.011349     76.272   < 2e-16 ***
speechiness          -9.633179   0.209835    -45.908   < 2e-16 ***
tempo                -0.00893    0.001134     -7.876  3.39E-15 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error:  16.21  on  232716  degrees of freedom
Multiple R-squared:  0.2054 ,    Adjusted R-squared:  0.2054
F-statistic:  7519  on  8  and  232716  DF,   p-value: < 2.2e-16
```

Regression Equation

$Y = 59.60 - 14.46_{X1} + 9.87_{X2} + 0.24_{X3} - 15.19_{X4} - 2.70_{X5} + 0.87_{X6} - 9.63_{X7} - 0.01_{X8}$

Variables with weakest relationship:

- Duration_min
- Loudness
- Tempo

Variables with strongest relationship:

- Acousticness
- Energy
- Speechiness

P-Value: The P-Value of all our variables are less than .05, meaning they're all significant, so we reject the null hypothesis.

R-Square: Our R-Square is 20.54%, meaning our 8 variables can explain 20.54% of the Popularity rating. Although the 8 variables only account for 20.54% of Popularity, each variable is within the .05 significance range.

Conclusion: With our 8 variables being significant but only making up 20.54% of the Popularity rating, we can conclude that in order to improve the goodness of fit in predicting the Popularity rating, we need additional variability measures.

Sources:

[i] Get Audio Features for a Track: https://developer.spotify.com/documentation/web-api/reference/tracks/get-audio-features/
Reference/Guide: https://rpubs.com/KeyaSatpathy/604834