Song Language and Success in the US Music Industry

Madison Martin

2023-06-15

Introduction

This project aims to illustrate the relationship between the language of the lyrics of songs in popular music in the United States and the song's success, using data from Spotify and radio plays. It was inspired by the recent rise in popularity of music sung in non-English languages in the US by artists like BTS, Bad Bunny, Maluma, Måneskin, and Blackpink. *This project is a modified form of an original project completed for credit at the University of Chicago in 2021.*

Research Question: **Does linguistic discrimination (discrimination on the basis of the manner of speaking or language being spoken) occur in the US popular music industry, as evidenced by discrepancies in either audience support or industry support?**

Sourcing the Data

In order to address the research question, I chose to source my songs from the pool of popular songs that have appeared on the Billboard Hot 100 chart, as it has long been an industry standard. A total of 17 consecutive months of chart data were collected (from January 2020 through May 2021), for a total of 7,300 songs. The charts and song data were collected directly from Billboard, and included the song's rank (from 1-100), title, artist, and week dating the chart. In order to measure the success or support of the songs on two levels, I also collected audience support data (Spotify streams) and industry support data (radio plays). The Spotify data was sourced directly from Spotify by week (to match the weekly Hot 100 chart), while the radio play data was sourced from Chartmetric.

Language of Lyrics

In order to assess the impact of lyric language on the support a song gets, I coded each song's lyrics for language on a word-by-word basis, in order to calculate the percentage of each song's lyrics which are sung in a non-English language. I started by identifying which songs were likely to have non-English lyrics using the cld3 package in R. However, the accuracy of language identification tools is poor, and is even less accurate on a singular word basis, so the rest of the coding for word language in each song's lyrics was done by hand. For more information, see the appendix.

Download the data

```
# download the final combined data for 2020 and for partial 2021
# Load the necessary libraries
library(readr)
library(tidyr)
```

```r
library(ggplot2)
library(tidyverse)
library(dplyr)

# read in csv for 2020
total_20_df <- read_csv(file = "final_20.csv")
head(total_20_df)
```

```
## # A tibble: 6 x 16
##    year date        rank song     artist binary language total_words non_e_words
##   <dbl> <date>     <dbl> <chr>    <chr>   <dbl> <chr>          <dbl>       <dbl>
## 1  2020 2020-01-04     1 all i w~ maria~      0 english          356           0
## 2  2020 2020-01-04     2 rockin ~ brend~      0 english          119           0
## 3  2020 2020-01-04     3 jingle ~ bobby~      0 english          185           0
## 4  2020 2020-01-04     4 a holly~ burl ~      0 english          159           0
## 5  2020 2020-01-04     5 circles  post ~      0 english          314           0
## 6  2020 2020-01-04     6 roxanne  arizo~      0 english          355           0
## # i 7 more variables: percent <dbl>, ‘individual songs‘ <dbl>, position <dbl>,
## #   streams <dbl>, ‘spotify date‘ <date>, spins <dbl>, radio_rank <dbl>
```

```r
# read in csv for 2021
total_21_df <- read_csv(file = "final_21.csv")
```

Note: these CSV files already contain the language coding variables along with streaming and radio play data, as all three sources of data were cleaned and combined previous to this project for brevity and clarity. If you are interested in the raw Spotify or Chartmetric data, please contact me.

Precleaning for Merging

```r
# clean up the 2020 data set
# first, remove the column we don't need
total_20_df <- total_20_df %>%
  select(-`spotify date`) %>%
  filter(year == 2020) # only 2020 songs

# remove duplicate songs
library(data.table)
total_2020 <- unique(setDT(total_20_df), by = c("date", "rank", "song"))

# replace NAs in individual songs with 0
total_2020 <- total_2020 %>%
  rename("unique" = "individual songs")

total_2020$unique <- total_2020$unique %>%
  replace_na(0)
```

```r
# clean up the 2021 data set
# first, remove the column we don't need
total_21_df <- total_21_df %>%
  select(-`spotify date`, - `chartmetric date`)
```

```r
# remove duplicate songs
total_2021 <- unique(setDT(total_21_df), by = c("date", "rank", "song"))

# replace NAs in individual songs with 0
total_2021 <- total_2021 %>%
  rename("unique" = "individual songs")

total_2021$unique <- total_2021$unique %>%
  replace_na(0)
```

Merge 2020 and 2021

```r
# merge the final two datasets for analysis
# rename column in 2021 to match
total_2021 <- total_2021 %>%
  rename("position" = "Position",
         "streams" = "Streams")

full_20_21 <- rbind(total_2020, total_2021)
head(full_20_21)
```

```
##    year       date rank                             song          artist binary
## 1: 2020 2020-01-04    1  all i want for christmas is you   mariah carey      0
## 2: 2020 2020-01-04    2 rockin around the christmas tree     brenda lee      0
## 3: 2020 2020-01-04    3                  jingle bell rock    bobby helms      0
## 4: 2020 2020-01-04    4           a holly jolly christmas      burl ives      0
## 5: 2020 2020-01-04    5                           circles    post malone      0
## 6: 2020 2020-01-04    6                           roxanne arizona zervas      0
##    language total_words non_e_words percent unique position streams spins
## 1:  english         356           0       0      1        2 9806687   196
## 2:  english         119           0       0      1        4 8624106   196
## 3:  english         185           0       0      1        5 8361420   196
## 4:  english         159           0       0      1       15 6266059   196
## 5:  english         314           0       0      0        7 7519188  7144
## 6:  english         355           0       0      0        3 9691192  3701
##    radio_rank
## 1:        501
## 2:        501
## 3:        501
## 4:        501
## 5:          2
## 6:         12
```

```r
# export for google sheets
write.csv(full_20_21, file = "full_20_21.csv", row.names = F)
```

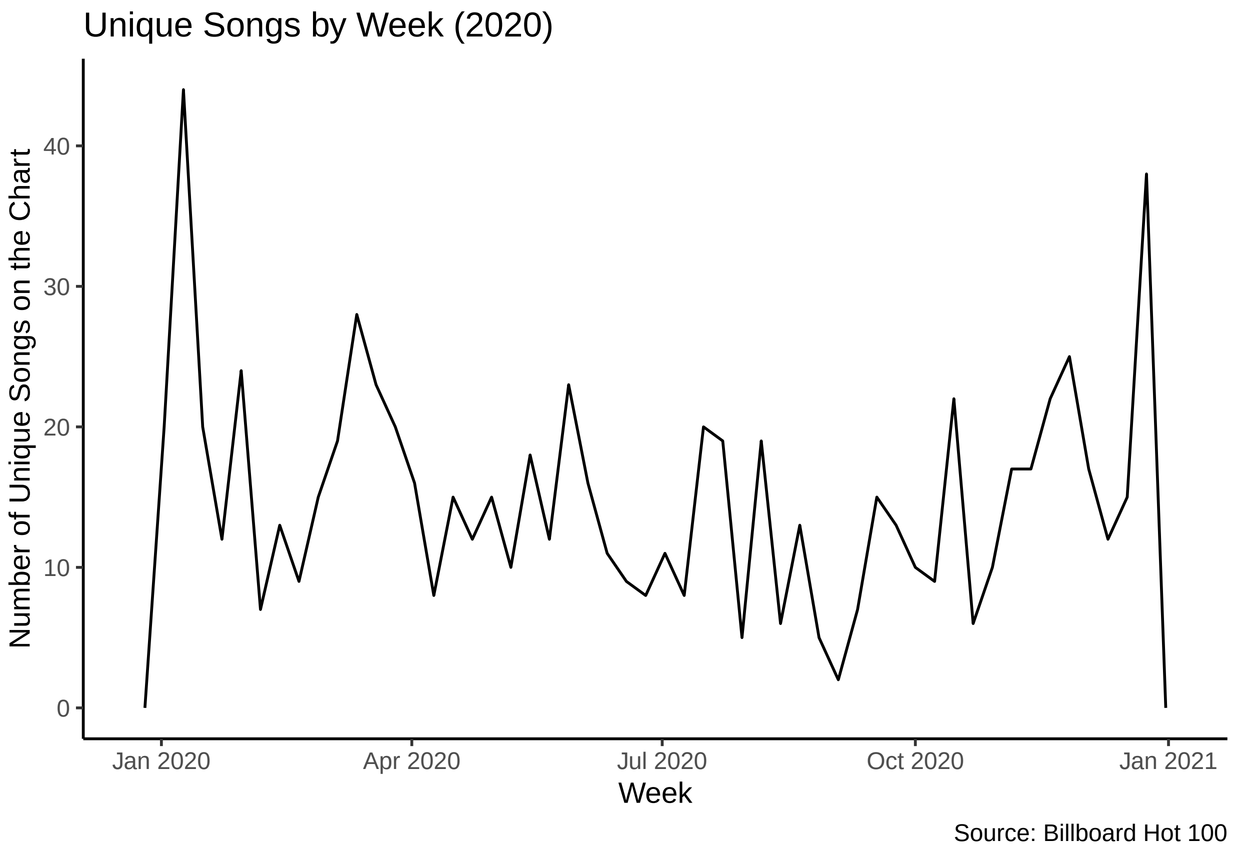
Exploring the Data

Unique Songs, Streams, and Radio Play

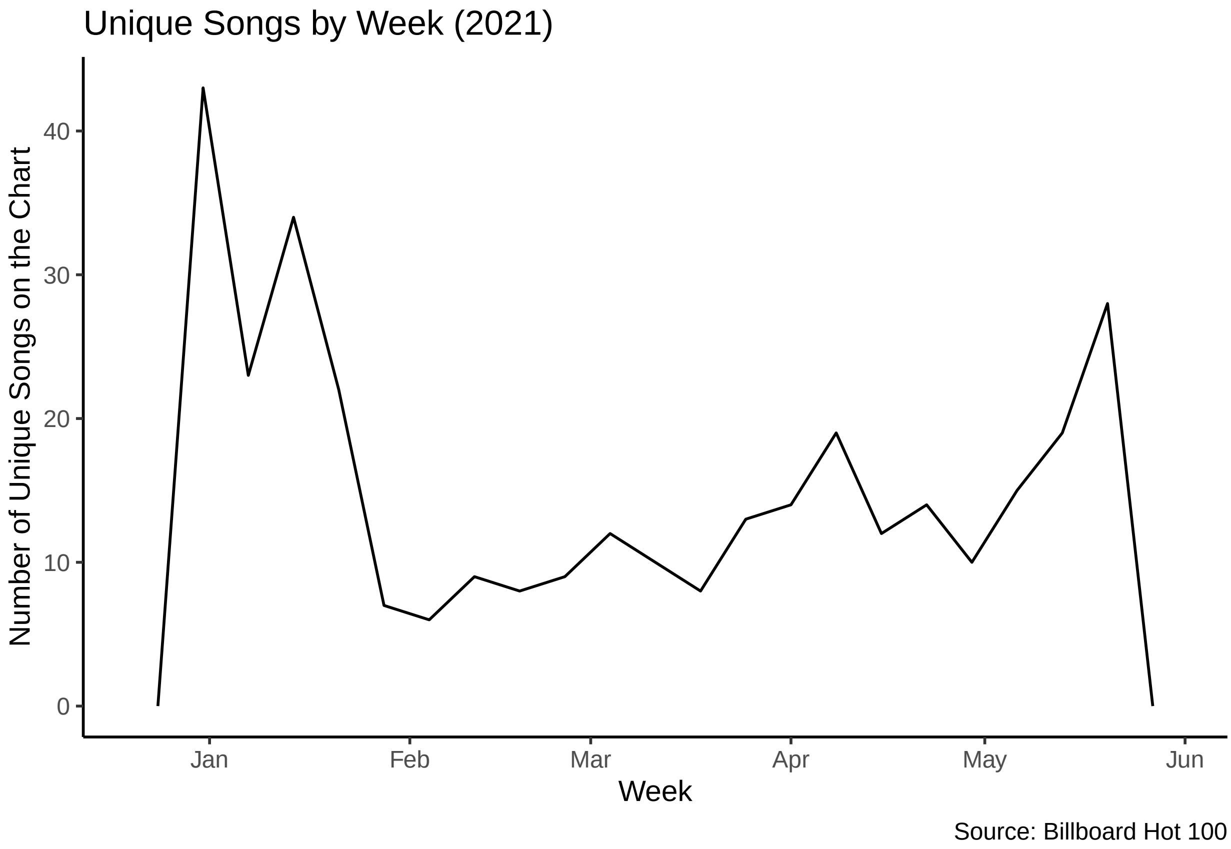
```
# how many unique songs
full_20_21 %>%
  filter(unique == 1) %>%
  count()
```

```
##       n
## 1: 1125
```

```
# unique songs by week for 2020 viz
full_20_21 %>%
  filter(unique == 1,
         year == 2020) %>%
  ggplot(mapping = aes(x = date)) +
  geom_freqpoly(binwidth = 7) +
  theme_classic() +
  labs(
    title = "Unique Songs by Week (2020)",
    x = "Week",
    y = "Number of Unique Songs on the Chart",
    caption = "Source: Billboard Hot 100"
  )
```



```
# same for 2021 viz
full_20_21 %>%
  filter(unique == 1,
         year == 2021) %>%
  ggplot(mapping = aes(x = date)) +
  geom_freqpoly(binwidth = 7) +
  theme_classic() +
  labs(
    title = "Unique Songs by Week (2021)",
    x = "Week",
    y = "Number of Unique Songs on the Chart",
    caption = "Source: Billboard Hot 100"
  )
```

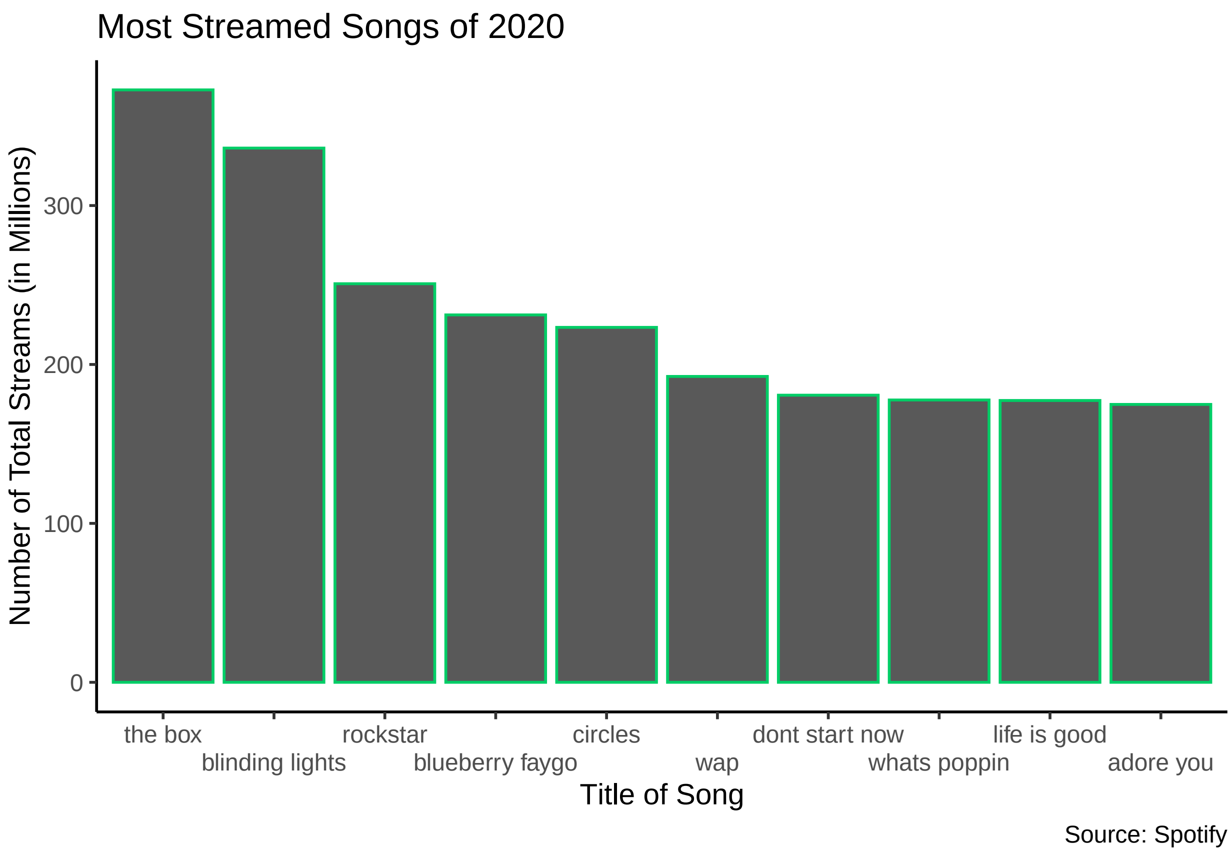


A total of 7,300 songs were analyzed, sourced from two consecutive years (2020-2021) of the Billboard Hot 100 weekly charts. Out of the 17 months analyzed for this draft, there were only 1,125 unique songs (15.41%), due to many songs staying on the Hot 100 charts for multiple weeks, either consecutively or not. Specifically, there were 790 unique songs in 2020 (15.19%), and there were 335 (15.95%) unique songs in the portion of 2021 analyzed.

```
# show most streamed songs of 2020
# add milstream for full df for easier viz
total_df <- full_20_21 %>%
  mutate(mil_stream = streams/1000000)

# add kplay for full df for easier viz
```

```r
total_df <- total_df %>%
  mutate(kspins = spins/1000)

# get mean and sd for streams and plays
mean(total_df$mil_stream)
sd(total_df$mil_stream)
mean(total_df$spins)
sd(total_df$spins)
```

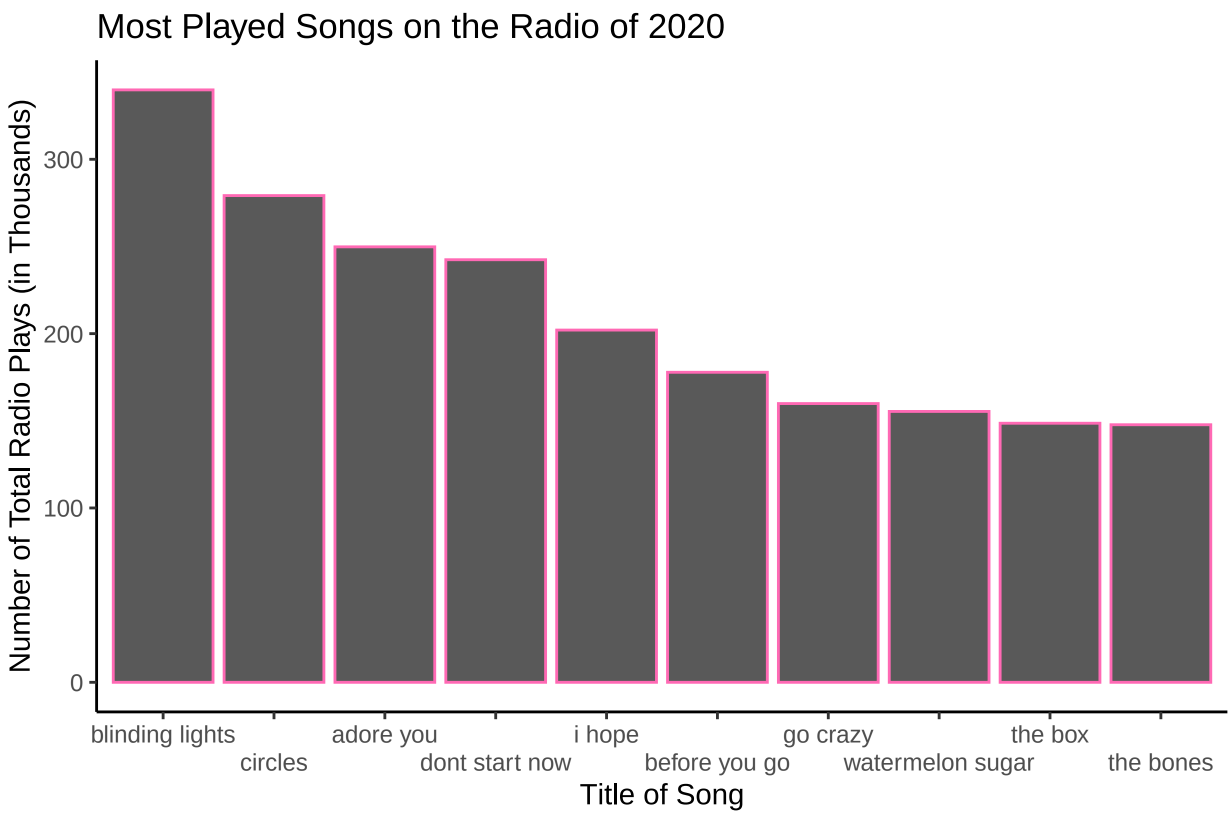
```r
# show most streamed songs of 2020
top_streams <- total_df %>%
  filter(year == 2020) %>%
  group_by(song, artist) %>%
  summarise(total_streams = sum(mil_stream)) %>%
  arrange(desc(total_streams)) %>%
  head(n = 10)

#viz
top_streams %>%
  ggplot(mapping = aes(reorder(song, -total_streams), y = total_streams)) +
  geom_bar(stat = "identity", colour = "springgreen3") + #to match spotify color
  scale_x_discrete(guide = guide_axis(n.dodge = 2)) + #to make it easier to read
  theme_classic() +
  labs(
    title = "Most Streamed Songs of 2020",
    x = "Title of Song",
    y = "Number of Total Streams (in Millions)",
    caption = "Source: Spotify"
  )
```



Most Streamed Songs of 2020

Source: Spotify

```r
# top radio hits
top_spins <- total_df %>%
  filter(year == 2020) %>%
  group_by(song, artist) %>%
  summarise(total_spins = sum(kspins)) %>%
  arrange(desc(total_spins)) %>%
  head(n = 10)

top_spins %>%
  ggplot(mapping = aes(reorder(song, -total_spins), y = total_spins)) +
  geom_bar(stat = "identity", colour = "hotpink") + #chartmetric is also green so chose pink
  scale_x_discrete(guide = guide_axis(n.dodge=2)) +
  theme_classic() +
  labs(
    title = "Most Played Songs on the Radio of 2020",
    x = "Title of Song",
    y = "Number of Total Radio Plays (in Thousands)",
    caption = "Source: Chartmetric"
  )
```



Source: Chartmetric

The average number of Spotify streams per week per song was over 3 million (M = 3118273.47, SD = 2229135.38), while the average number of radio plays per week per song was much less, comparatively (M = 1839.12, SD = 1867.48). The top 10 most streamed songs on Spotify in 2020, as well as the top 10 played songs on the radio in 2020, include songs such as "The Box" by Roddy Rich, "Blinding Lights" by The Weeknd, "Circles" by Post Malone, and "Adore You" by Harry Styles.

Lyric Language

```
# average number of words in the lyrics
# make a df with only the unique songs for easier analysis later
unique_df <- total_df %>%
  filter(unique == 1)

# oops! I have 3 NAs for lyric length, replace with mean
total_df$total_words[is.na(total_df$total_words)] <- mean(total_df$total_words, na.rm = TRUE)
# now calculate mean
mean(total_df$total_words)
```

```
## [1] 432.5719
```

```
# how many songs overall have some non-English lyrics
total_df %>%
  filter(percent > 0) %>%
  count()
```

```
##       n
## 1: 1230
```

```
# how many unique songs have non-E lyrics
unique_df %>%
  filter(percent > 0) %>%
  count()
```

```
##      n
## 1: 225
```

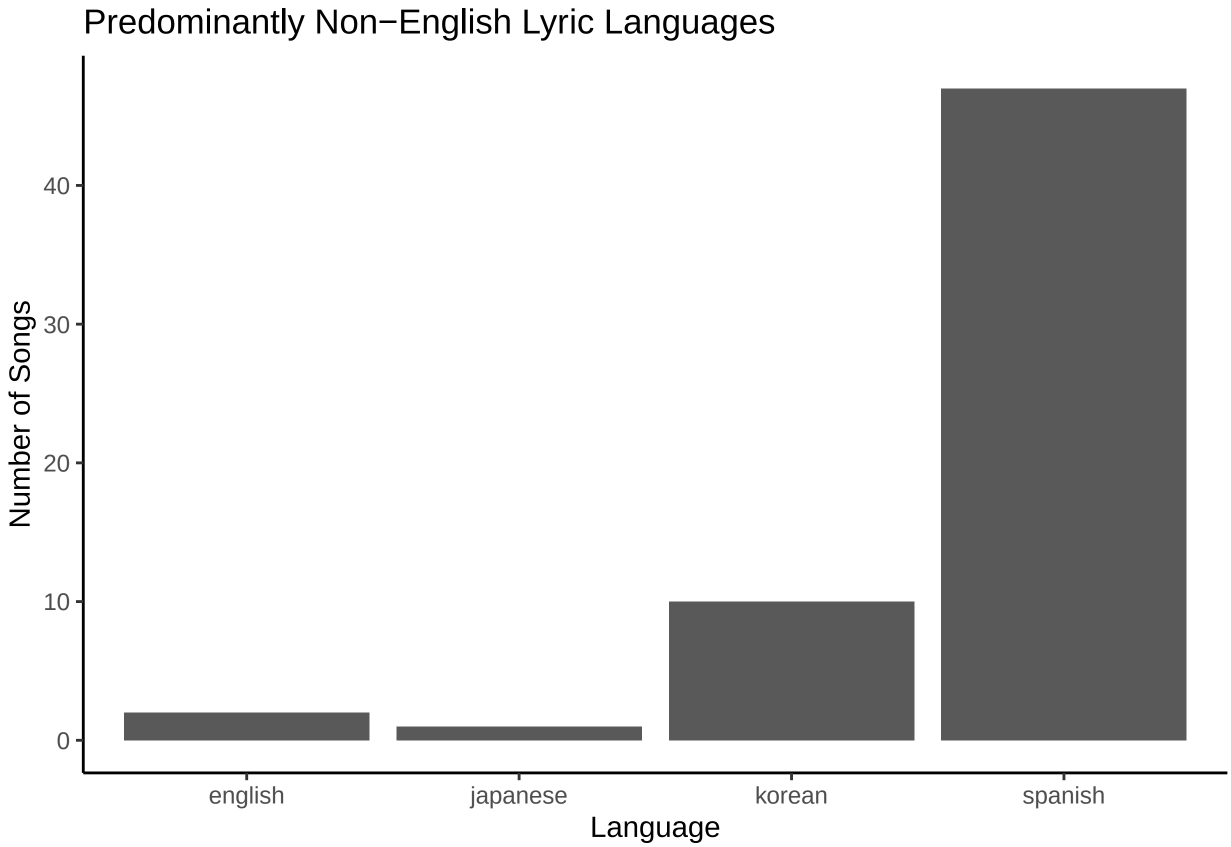
```
# significant portion of non-E lyrics
total_df %>%
  filter(percent > 25) %>%
  count()
```

```
##      n
## 1: 322
```

```
# sig portion of unique songs
unique_df %>%
  filter(percent > 25) %>%
  count()
```

```
##     n
## 1: 69
```

```
#viz for non-E languages
unique_df %>%
  filter(percent > 50) %>%
  ggplot(mapping = aes(x = language)) +
  geom_bar() +
  theme_classic() +
  labs(
    title = "Predominantly Non-English Lyric Languages",
    x = "Language",
    y = "Number of Songs"
  )
```



After coding the lyrical content of the songs for language percentage based on the word count, there were a total of 1,230 songs (16.85%) from 2020-2021 that included some non-English. When considering the unique songs from both years, a total of 225 songs (20%) included some non-English lyrics. Additionally, there were 322 songs (4.41%) which included a significant portion of their lyrics (25% or more) sung in non-English-languages, with that number increasing (6.33 %) when analyzing the unique songs for both years. The average (mean) number of words in the lyrics of all songs analyzed for this project was 433. Languages represented in this data set included Spanish, French, Korean, Hawaiian, Arabic, Hebrew, Japanese, and German. The most common languages represented in the lyrics were Spanish and Korean.

Results

Correlations

```
# create an apa table to be used in paper
library(apaTables)

# prep df for just the table
corr_df <- total_df %>%
  select(rank, binary, percent, position, streams, spins, radio_rank) %>%
  rename("Billboard Rank" = "rank",
         "Language" = "binary",
         "Non-English Lyrics Percentage" = "percent",
         "Spotify Rank" = "position",
```

```r
         "Streams" = "streams",
         "Radio Plays" = "spins",
         "Radio Rank" = "radio_rank")

apa.cor.table(corr_df, filename="Table_APA_paper.doc", show.conf.interval=F)
```

```
## The ability to suppress reporting of reporting confidence intervals has been deprecated in this versi
## The function argument show.conf.interval will be removed in a later version.
```

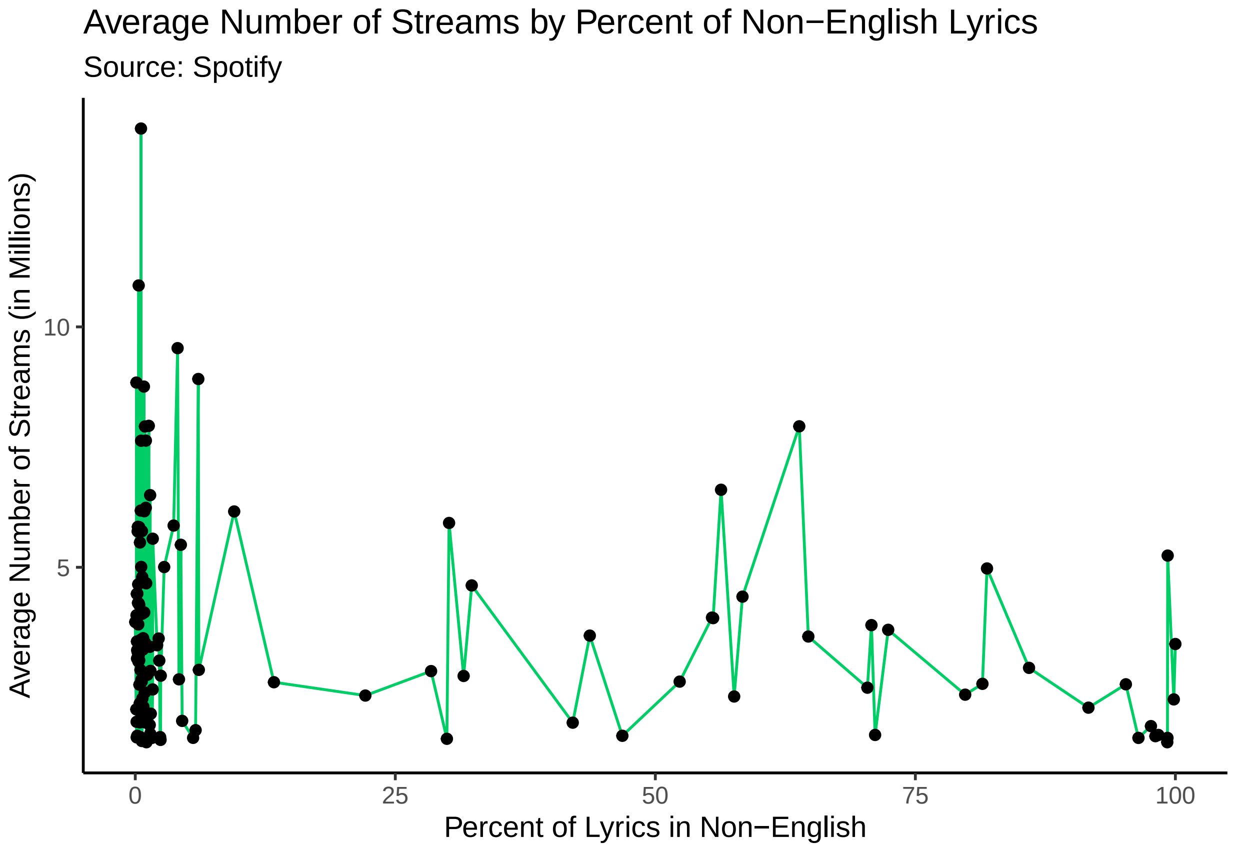
```
## 
## 
## Means, standard deviations, and correlations with confidence intervals
## 
## 
##   Variable                         M          SD         1
##   1. Billboard Rank                50.50      28.87
## 
##   2. Language                      0.03       0.18       .12**
##                                                          [.09, .14]
## 
##   3. Non-English Lyrics Percentage 3.66       17.16      .13**
##                                                          [.10, .15]
## 
##   4. Spotify Rank                  100.72     76.96      .46**
##                                                          [.44, .48]
## 
##   5. Streams                       3118273.47 2229135.38 -.50**
##                                                          [-.51, -.48]
## 
##   6. Radio Plays                   1839.12    1867.48    -.54**
##                                                          [-.55, -.52]
## 
##   7. Radio Rank                    199.76     203.33     .36**
##                                                          [.34, .38]
## 
##   2            3            4            5          6
## 
## 
## 
## 
## 
##   .95**
##   [.94, .95]
## 
##   -.01         -.01
##   [-.04, .01]  [-.03, .01]
## 
##   -.03*        -.03**       -.72**
##   [-.05, -.00] [-.05, -.01] [-.73, -.71]
## 
##   -.14**       -.15**       -.05**       .07**
##   [-.16, -.11] [-.18, -.13] [-.08, -.03] [.04, .09]
## 
```

```
##   .21**        .22**        -.14**       .08**      -.71**
##   [.19, .23]   [.20, .24]   [-.17, -.12] [.05, .10] [-.73, -.70]
##
##
## Note. M and SD are used to represent mean and standard deviation, respectively.
## Values in square brackets indicate the 95% confidence interval.
## The confidence interval is a plausible range of population correlations
## that could have caused the sample correlation (Cumming, 2014).
##  * indicates p < .05. ** indicates p < .01.
##
```

There was a significant correlation between non-English lyric percentage and Billboard rank (r = .13, p < .01) and Spotify streams (r = -.03, p < .01), but not for Spotify Chart rank (1-100) . That is, a higher percentage of non-English lyrics was correlated with a lower Billboard rank (closer to 100 than 1), and less Spotify streams. Additionally, the non-English lyric percentage was significantly correlated with Radio plays (r = -.15, p < .01) and Radio Chart rank (r = .22, p < .01). That is, a higher percentage of non-English lyrics was correlated with a lower rank on the Radio charts (closer to 500 than 1), and less plays on the radio per week.

The correlations were very similar when looking at the language variable, which was the binary variable indicating if the song was predominantly (50% or more) in non-English or not. Specifically, the language variable was significantly correlated with Billboard rank (r = .12, p < .01), Spotify streams (r = -.03, p < .01), Radio plays (r = -.14, p < .01), and Radio chart rank (r = .21, p < .01), but not significant for Spotify chart rank. As for the non-English lyric percent, the binary assignment of the lyrics' predominant language to non-English was correlated with a lower rank on Billboard and Radio play charts, as well as a lower Spotify streams and Radio plays per week.

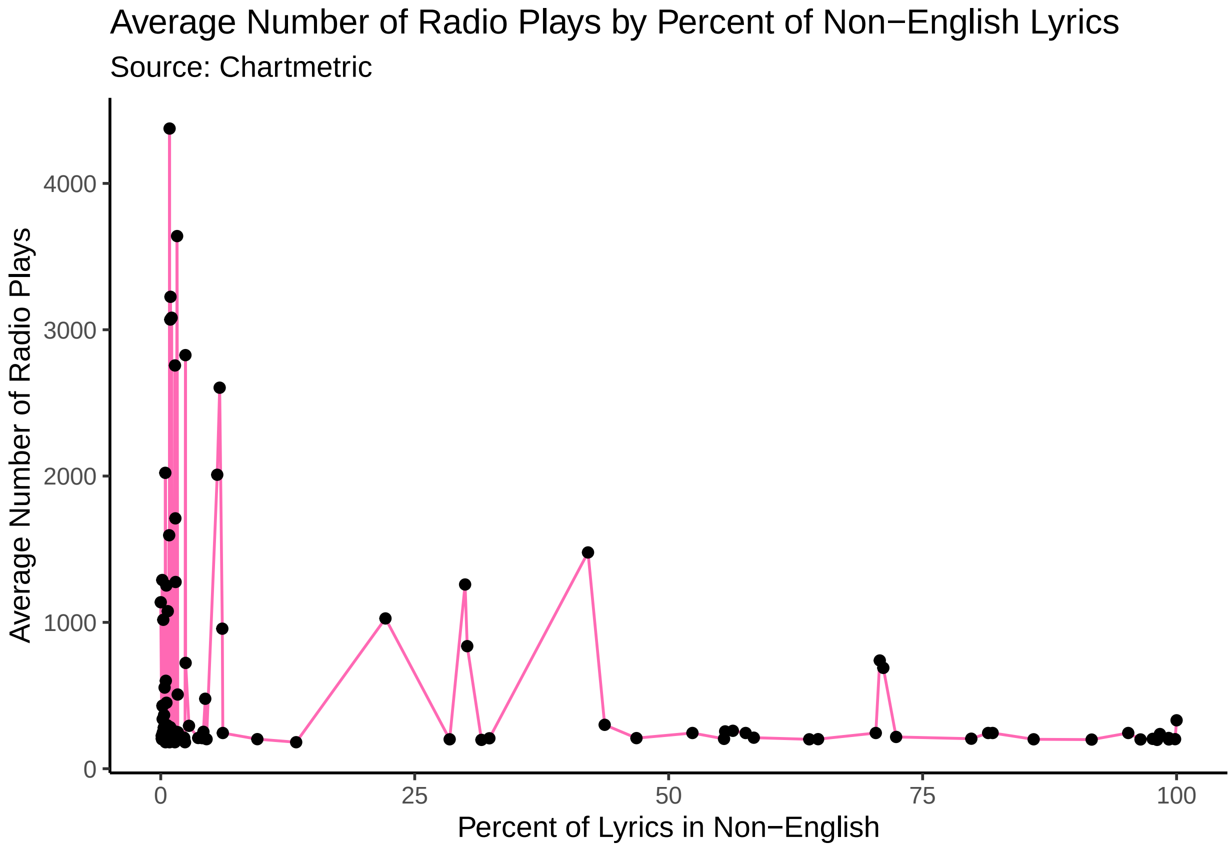
```r
# viz of relationship between average streams and percentage of words in lyrics in non-English
total_df %>%
  filter(unique == 1) %>%
  group_by(percent) %>%
  summarise(mean_streams = mean(mil_stream)) %>%
  ggplot(mapping = aes(x = percent, y = mean_streams)) +
  geom_line(colour = "springgreen3") +
  geom_point() +
  theme_classic() +
  labs(
    title = "Average Number of Streams by Percent of Non-English Lyrics",
    x = "Percent of Lyrics in Non-English",
    y = "Average Number of Streams (in Millions)",
    subtitle = "Source: Spotify"
  )
```



Average Number of Streams by Percent of Non−English Lyrics

Source: Spotify

```
# viz of relationship between average radio plays and percentage of words in lyrics in non-Eng
total_df %>%
  filter(unique == 1) %>%
  group_by(percent) %>%
  summarise(mean_spins = mean(spins)) %>%
  ggplot(mapping = aes(x = percent, y = mean_spins)) +
  geom_line(colour = "hotpink") +
  geom_point() +
  theme_classic() +
  labs(
    title = "Average Number of Radio Plays by Percent of Non-English Lyrics",
    x = "Percent of Lyrics in Non-English",
    y = "Average Number of Radio Plays",
    subtitle = "Source: Chartmetric"
  )
```


Average Number of Radio Plays by Percent of Non−English Lyrics

Source: Chartmetric

As can be seen from these graphs showing the relationship between the percentage of lyrics in non-English and the other measures of song support, while some songs with more non-English lyrics had high streams, the average number of radio plays for songs above 75% non-English were very low compared to the songs with less than 25% non-English lyrics.

Multiple Linear Regression

```
# for the total data set, are streams influenced by language of lyrics when controlling for radio suppo
# here, percent variable is the percentage of the lyric words that are non-English
lm1 <- lm(streams ~ percent + spins, data = total_df)
summary(lm1)
```

```
##
## Call:
## lm(formula = streams ~ percent + spins, data = total_df)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2128070 -1547388  -755770   814541 27414627
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept) 2990488.42   37799.05  79.115  < 2e-16 ***
## percent       -2749.10    1535.74  -1.790   0.0735 .
## spins            74.89      14.11   5.307 1.15e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2224000 on 7292 degrees of freedom
##   (3 observations deleted due to missingness)
## Multiple R-squared:  0.004791,   Adjusted R-squared:  0.004518
## F-statistic: 17.55 on 2 and 7292 DF,  p-value: 2.485e-08
```

```
## language not sig when controlling for radio spins

# for the total data set, are radio spins influenced by langauge of lyrics when controlling for audienc
lm2 <- lm(spins ~ mil_stream + percent, data = total_df)
summary(lm2)
```

```
##
## Call:
## lm(formula = spins ~ mil_stream + percent, data = total_df)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2923.2 -1582.2  -358.9   835.7  9887.8
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 1739.925     37.497  46.402  < 2e-16 ***
## mil_stream    51.367      9.680   5.307 1.15e-07 ***
## percent      -16.490      1.257 -13.114  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1842 on 7292 degrees of freedom
##   (3 observations deleted due to missingness)
## Multiple R-squared:  0.02729,    Adjusted R-squared:  0.02703
## F-statistic: 102.3 on 2 and 7292 DF,  p-value: < 2.2e-16
```

```
## language percent IS sig when controlling for streams!


# final analysis on 2020 only since it's the only full year of data I have
final_2020 <- total_df %>%
  filter(year == 2020)
lm3 <- lm(spins ~ mil_stream + percent, data = final_2020)
summary(lm3)
```

```
##
## Call:
## lm(formula = spins ~ mil_stream + percent, data = final_2020)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2542.1 -1562.6  -346.6   762.8  9909.0
##
```

```
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 1663.797     45.558  36.520  < 2e-16 ***
## mil_stream    68.763     11.397   6.034 1.71e-09 ***
## percent      -18.575      1.526 -12.171  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1868 on 5194 degrees of freedom
##   (1 observation deleted due to missingness)
## Multiple R-squared:  0.03562,    Adjusted R-squared:  0.03525
## F-statistic: 95.93 on 2 and 5194 DF,  p-value: < 2.2e-16
```

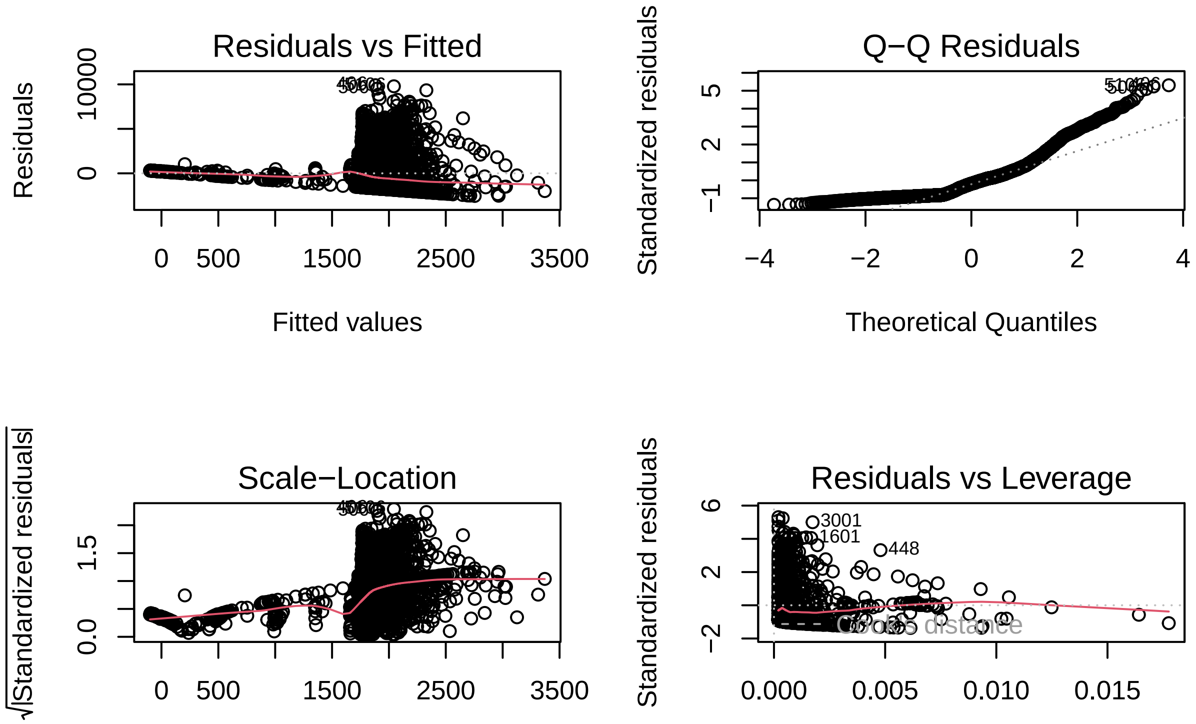
```
## langauge percent has a higher impact here (larger intercept)
```

Results of the multiple linear regression analysis indicated that lyric language (the percentage of words in each song's lyrics that are in a non-English language) was not a significant predictor in audience support (spotify streams), when controlling for radio support. However, when analyzing the data from 2020, there was a collective significant effect of percent of non-English lyrics and Spotify streams on Radio plays (F(2, 5194) = 95.93, p < .001, R2 = .036). The individual predictors were examined further and indicated that percent of non-English lyrics (B = -18.575, p < .001) and Spotify streams (B = 68.763, p < .001) were significant predictors in the model. In other words, the amount of radio plays per song per week could be predicted by the following formula:

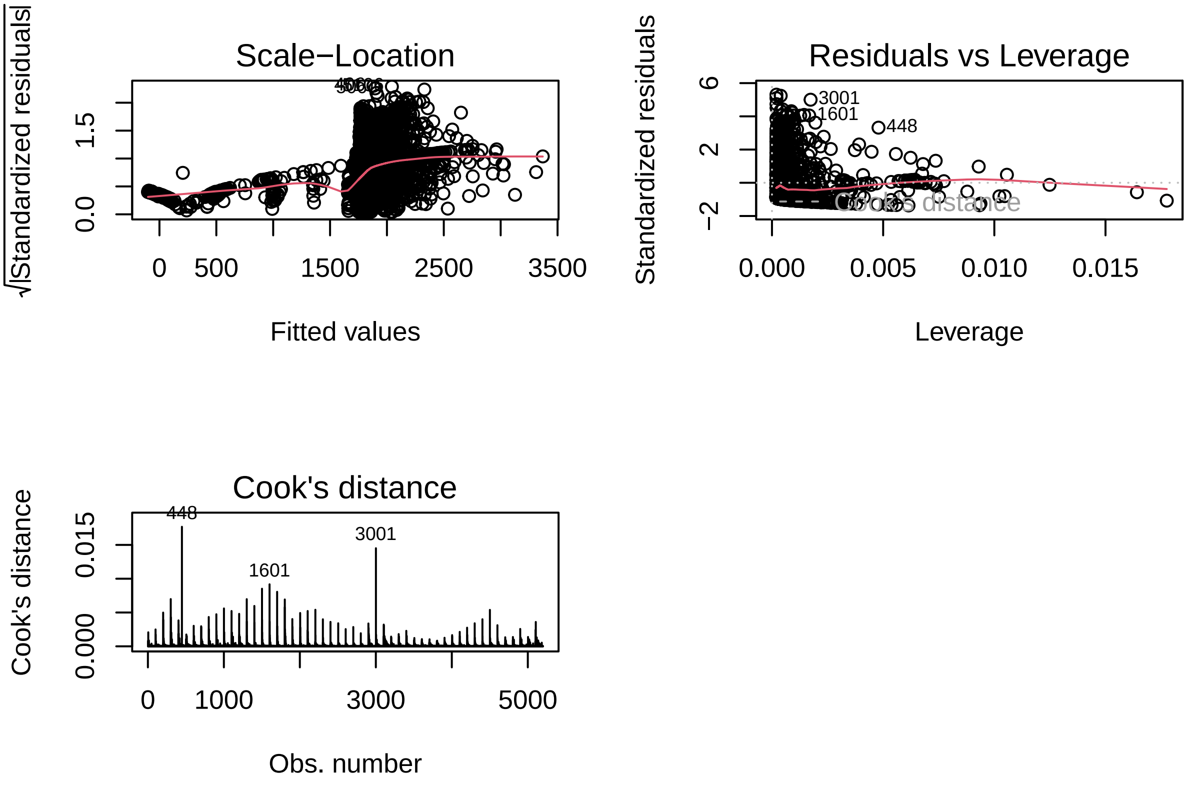
Spins = 1663.8 + 68.76 (streams in millions) - 18.58 (percent of non-English lyrics)

The diagnostics of this multiple linear regression model are presented below.

```
library(broom)
# diagnostics of the model
par(mfrow = c(2, 2))
plot(lm3)
```



```
plot(lm3, 3)

# look for outliers and high leverage points
plot(lm3, 5)
plot(lm3, 4)
```



The amount of explained variance is relatively low in the selected model (R2 = .036), which is to be expected in such a complex issue (how much radio play or audience support a song gets), especially considering this data set did not include any audio-based data which can influence popularity and success like genre, tempo, key, etc. However, let's see if we can increase our explained variance in radio plays with a different model.

```
# going to try a linear regression with all other variables to see if i can increase r2
lm_all <- lm(spins ~ rank + mil_stream + percent + date + total_words, data = final_2020)
summary(lm_all)
```

```
##
## Call:
## lm(formula = spins ~ rank + mil_stream + percent + date + total_words,
##     data = final_2020)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3778.6  -844.1   -75.8   704.9  9175.6
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) 8995.0909  3651.1520   2.464   0.0138 *
## rank         -43.6649     0.8328 -52.430  < 2e-16 ***
## mil_stream  -188.2454    10.5235 -17.888  < 2e-16 ***
## percent       -8.5008     1.2274  -6.926 4.85e-12 ***
## date          -0.2025     0.1975  -1.025   0.3052
## total_words   -1.3729     0.1261 -10.888  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1485 on 5191 degrees of freedom
##   (1 observation deleted due to missingness)
## Multiple R-squared:  0.391,  Adjusted R-squared:  0.3904
## F-statistic: 666.5 on 5 and 5191 DF,  p-value: < 2.2e-16
```

```
## obviously, this has a very high r2
## although I know that there is confounds here with the rank variable
```

When including Billboard Hot 100 rank, charting week date, and the total number of words in the song's lyrics, the collective effect remains significant and the explained variance of the model increases drastically (F(5, 5191) = 666.5, p < .001, R2 = .391). However, the reason this model was not chosen for the final analysis is because by adding the Billboard Hot 100 rank variable, the predictor variables are no longer independent, as Billboard Hot 100 rank is computed utilizing streaming data and radio play data, which are also included here as predictor variables. It is interesting to note, however, that when controlling for all of these other variables, the total number of words in the song lyrics (B = -1.372, p < .001) is a significant predictor in number of radio plays. That is, songs with increased verbosity had less radio streams.

```
# let's try radio rank instead of number of spins, just to see if our finding holds
lm_radiorank <- lm(radio_rank ~ mil_stream + percent + total_words + date, data = final_2020)
summary(lm_radiorank)
```

```
##
## Call:
## lm(formula = radio_rank ~ mil_stream + percent + total_words +
##     date, data = final_2020)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -419.5 -146.4  -87.8  209.1  411.9
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 389.36625  473.03974   0.823    0.410
## mil_stream    7.83108    1.18506   6.608 4.28e-11 ***
## percent       3.15150    0.15749  20.010  < 2e-16 ***
## total_words   0.23873    0.01627  14.677  < 2e-16 ***
## date         -0.01786    0.02560  -0.698    0.486
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 192.7 on 5192 degrees of freedom
##   (1 observation deleted due to missingness)
## Multiple R-squared:  0.1174, Adjusted R-squared:  0.1167
## F-statistic: 172.6 on 4 and 5192 DF,  p-value: < 2.2e-16
```

```
# let's do spotify position instead of actual streaming numbers
lm_position <- lm(spins ~ position + percent + date + total_words, data = final_2020)
summary(lm_position)
```

```
##
```

```
## Call:
## lm(formula = spins ~ position + percent + date + total_words,
##     data = final_2020)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2657.4 -1397.5  -383.7   771.9 10031.8
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) 2148.88325 4503.95648   0.477    0.633
## position      -2.20723    0.34022  -6.488 9.53e-11 ***
## percent      -18.39445    1.50181 -12.248  < 2e-16 ***
## date           0.04779    0.24383   0.196    0.845
## total_words   -2.12352    0.15654 -13.566  < 2e-16 ***
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## Residual standard error: 1839 on 5192 degrees of freedom
##   (1 observation deleted due to missingness)
## Multiple R-squared:  0.0657, Adjusted R-squared:  0.06498
## F-statistic: 91.28 on 4 and 5192 DF,  p-value: < 2.2e-16
```

The results of these additional multiple linear regression models matches our main finding from the lm3 model conducted previously. Specifically, when considering radio rank (1-500) instead of number of radio plays (spins), the percent of non-English lyrics (B = 3.151, p < .001) is still a significant predictor in the model. Here, the radio rank increases (song gets lower on the chart) with an increase in percent of non-English lyrics, predicted by the following formula:

Radio Rank = 389.36 + 7.83 (streams in millions) + 0.24 (total words in lyrics) + 3.15 (percent of non-English lyrics)

Alternatively, when considering position on the Spotify chart (1-100) instead of number of streams as a predictor variable, the percent of non-English lyrics (B = -18.39, p < .001) is still a significant predictor in the model. In other words, the amount of radio plays per song per week can also be predicted by the following formula:

Spins = 2148.88 - 2.21 (Spotify chart position) - 2.12 (total words in lyrics) - 18.39 (percent of non-English lyrics)

Discussion

Language and Audience Support

While there was some evidence for audience preference for fully English songs, as evidenced by the correlation between lyric language percent and Spotify streams, the size of the correlation was relatively small. In other words, there was only evidence of a small amount of potential bias against songs that include non-English lyrics, which might reflect the growing percentage of Americans who speak foreign languages at home or are bilingual (21.9% of residents as of 2018). Additionally, the percentage of non-English lyrics in a song did not significantly impact the rank on the Spotify charts, which further indicates that the audience bias against non-English songs, although present, was not strong enough to significantly impact the song’s success when it comes to support in the form of streaming.

However, it is important to note that given the type and amount of data collected in this study, it is not possible to identify whether the relationship between language and audience support via streaming is evidence of linguistic bias. If songs which include non-English lyrics are less likely to be picked for Spotify's influential playlists (like Today's Top Hits and Viral Hits), then that could explain their less average streams per week. Additionally, the relatively weak audience bias against non-English songs could be explained by a simple difference in preferences, as English still is the most spoken language in the country and has dominated the music industry for decades.

Language and Industry Support

When controlling for audience support or preference, as evidenced by Spotify streams per week, there was still a bias against songs which included non-English language lyrics as seen in the number of radio plays per week. Specifically, the multiple linear regression analysis demonstrates that for this set of data, which spanned from January 2020 to May 2021, songs were assumed to have 16 fewer radio plays per week for every percent increase in non-English language included in the lyrics, with this disparity increases to 19 fewer radio plays per week for every percent non-English lyric increase when analyzing 2020 data alone. This industry bias against radio streams becomes especially meaningful when considering that this data set only included songs which had reached a very high level of success in the industry, as being on the Billboard Hot 100, and that the average radio spins per week to be on the Top 500 Airplay Chartmetric chart was only 1,839.

This disparity between the radio plays per week and audience support by streaming for songs with non-English lyrics clearly demonstrates the presence of an industry bias against non-English songs or artists, as the music industry is heavily involved in choosing what songs are played during radio programs. While radio stations do want to entice and keep listeners by playing music they are interested in, they also have to balance this desire with the drive to promote new music and support industry-backed artists which are being promoted to individual radio stations by music label promotion representatives (Knab, 2010). A string of lawsuits in the early 2000's against these "pay-for-play" practices have not stopped the process of industry bias and corruption in radio, instead driving the industry to invent new and clever ways to influence radio towards any direction they prefer (Leight, 2019). This finding replicates earlier studies which have found industry bias based on performer attributes, such as race and gender, and expands upon the idea of a discriminatory music industry ecosystem by including linguistic discrimination (Lafrance, Worcester & Burns, 2011; Strong & Raine, 2018; Watson, 2019; Lafrance et al., 2017; Laybourn, 2018).

Limitations and Future Directions

There were a few main limitations in this study. Firstly, the sample size of the data was limited by the lack of accurate programming solution to coding modern music lyrics on a singular word-based level, as many lyrics include complicated slang and code-switching which are not reliably marked by traditional spelling-check based methods (Barman et al., 2014). That, in addition to the limited access to high-quality data for the radio play data from Chartmetric, did not allow a large enough sample size to see a broader vision of trends in language composition of popular music. Secondly, the scope of the current study did not allow for inclusion of potential influences on music popularity other than audience support, which is most likely the reason the explained variance for the multiple linear regression model, although still statistically significant, is a small effect size (3.6%). This is understandable, as song characteristics (such as tempo, key, melodic content, and rhythmic patterns), artist characteristics (such as gender, race, age, overall popularity), and song success measures not included in this analysis (sales, YouTube streams, TikTok plays, etc.), along with the other features that have been shown in previous studies to impact song success are most likely producing much of the unexplained variance here (Cosimato et al., 2019; McAuslan & Waung 2018; Napier & Shameer 2018; Nunes & Ordanini, 2014; Ordanini, Nunes & Nanni, 2018).

In order to address these questions and expand upon this study, future research could create a specific language identification function or dictionary that accurately reflects current popular music lyrical contents, and include many more possible sources of variation in song success as evidenced by audience and radio support in order to better assess the presence and strength of linguistic bias in the industry.

Conclusion

In order to assess the role language plays in the American Music Industry, a pool of popular songs from 2020-2021 was collected and analyzed alongside measures of audience support (Spotify Streams) and industry support (Radio plays). By coding the lyrics of the songs in the data set for percentage of non-English words, then performing a multiple linear regression analysis in order to control for audience bias and listening preference, the study was able to demonstrate that there is bias against non-English language songs in the radio promotion of songs, which demonstrates a larger industry bias and potential discrimination.

Appendix

Language Coding

Below is a direct excerpt from the original project paper

The data set of popular songs sourced from the Billboard Hot 100 were then coded for language by analyzing the lyrical compositions of the songs. The lyrics for each song were taken from official sources (like Apple Music or Spotify, which is often powered by Genius) using web scraping methods. If the lyrics for a song were not available there, they were taken from one of two trusted websites (Lyrics.com or AZLyrics.com). First, in order to flag the songs that included multiple languages, a language identification function in R (package cld3) was used to analyze each song's lyrics to see if it included more than one language, or a single language other than English. Then, for each song that was flagged, the total word count of the lyrics was recorded, as well as the total number of non-English words in the lyrics. Finally, the percentage of non-English lyrics (proportion of non-English words to the total words in the lyrics) was calculated. It is worth noting that the accuracy of the language identification function in R was lower than expected, so some songs that were not originally flagged as potential non-English language songs had to be re-coded and analyzed by hand, in addition to the ones that were flagged.

During the lyric analysis, proper nouns (like names and places) were not considered non-English, unless they were the foreign-language equivalent (example: Barcelona was not marked, but Estados Unidos was marked as non-English). Additionally, names of designers and brands were not included, although it is interesting to note that they appeared frequently in the lyric corpus (examples: Gucci, Prada, Louis Vuitton, Lamborghini, etc.). Finally, words or slang that were originally from another language but have been inculcated into the daily vernacular of English speakers in the United States (examples: chardonnay or piñata) were not included in the non-English language count.

Alternatively, words that might be considered well-known in some sectors of the United States, but have direct replacements in the English language were counted as non-English (example: cerveza, which is the Spanish word for beer). In a similar fashion, words that could be considered slang here but were not deemed to be ubiquitous and are sourced from or directly borrowed from a non-English language were counted as non-English (example: fuego, which is the Spanish word for fire, which itself is a current slang term in English). Finally, words that originated in English but were being spoken in the middle of a non-English phrase, possibly with an accent to make them understandable to native speakers, were considered to be non-English.

While Billboard itself does not have an official definition of what songs are considered "predominantly non-English", following the examples of other media giants such as the Grammy's and the Oscar's, the songs were additionally coded in a binary based on whether they were majority (50% or more) non-English (1) or not (0) (Oscars, 2021, p. 19;Grammys, 2020, p. 53). Finally, the songs were coded for the language that made up the majority (50% or more) of their lyrics, like Korean, Spanish, English, etc.