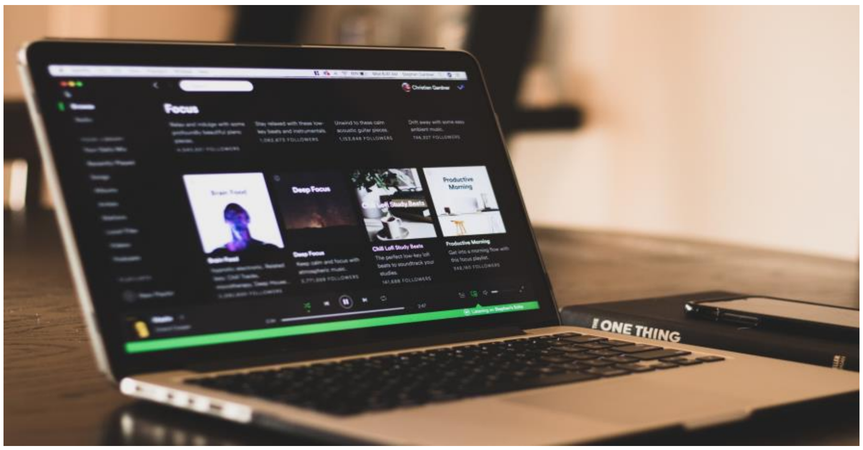
An analytical approach to see if my dad is right about today's music

Michael Hoang



During a car ride with my dad, he happened to catch a listen to my playlist and went on a judgmental rant. Essentially going on to say that (1) there was a missing quality in songs in today's music as compared to his day and that (2) us young people (i.e., 35 and under) have horrible taste in music. While I shrugged this off as another one of his "OK Boomer" moment, I thought I would see if the data backs this up.

Using some data from one of the biggest music streaming platforms, (Spotify), which also happens to have the majority of users being young adults, I will examine trends in audio and track features based on release year. Specifically, comparing some of the popular artists from the past 10 years with the most popular artists in his day (i.e., 1970s and 1980s). Furthermore, I will also look at what are some of the most popular artists in 2020.

The Data

A data set that was scrapped from Spotify Web API was made available on Kaggle that contained information from over 175,000 songs that released between 1921-2020. Aside from who's the credited artists and year of release, this data set also contained scores for various audio features relating to a given track. I've included a list and explanation below:

- Acousticness – to what degree is the sound of the track being produced by non-electric means
- Danceability – how suitable is the track is made for the purpose of dancing based on tempo, rhythm stability, beat per minute and overall regularity
- Duration – length of the track (in milliseconds)
- Energy – the perceptual measure of intensity and activity of the track based on dynamic range, perceived loudness, timbre, onset rate and general entropy
- Instrumentalness – to what degree does the track contain no vocal elements
- Key – what key is the track in according to standard pitch
- Liveness – the degree of presence of a live audience on the track
- Loudness – the overall averaged loudness of the track in decibels (dB)
- Mode – the relative keys in which the track is being played in (i.e., is it in major or minor?)
- Speechiness – the degree to which spoken words are present in the track
- Tempo – the overall speed or pace of the music played (based on beats per minute)
- Valence – the degree to which the sound conveys a sense of positive mood

NOTE: The majority of these features are scored along a scale from 0 to 1 (low-to-high)

```r
spotify = read.csv("spotify_data.csv")
skimr::skim(spotify)
```

```
-- Data Summary ------------------------
                           Values
Name                       spotify
Number of rows             170653
Number of columns          19
_______________________
Column type frequency:
  character                4
  numeric                  15
________________________
Group variables            None

-- Variable type: character ------------------------------------------------------------------------------------------
# A tibble: 4 x 8
  skim_variable n_missing complete_rate   min   max empty n_unique whitespace
* <chr>             <int>         <dbl> <int> <int> <int>    <int>      <int>
1 artists               0             1     5   675     0    34088          0
2 id                    0             1    22    22     0   170653          0
3 name                  0             1     1   203     0   133638          0
4 release_date          0             1     4    10     0    11244          0

-- Variable type: numeric --------------------------------------------------------------------------------------------
# A tibble: 15 x 10
   skim_variable    n_missing complete_rate        mean          sd    p0          p25           p50          p75         p100
 * <chr>                <int>         <dbl>       <dbl>       <dbl> <dbl>        <dbl>         <dbl>        <dbl>        <dbl>
 1 valence                  0             1      0.529       0.263     0       0.317        0.54         0.747          1
 2 year                     0             1   1977.         25.9    1921    1956          1977          1999          2020
 3 acousticness             0             1      0.502       0.376     0       0.102        0.516        0.893          0.996
 4 danceability             0             1      0.537       0.176     0       0.415        0.548        0.668          0.988
 5 duration_ms              0             1 230948.     126118.     5108 169827        207467        262400       5403500
 6 energy                   0             1      0.482       0.268     0       0.255        0.471        0.703          1
 7 explicit                 0             1      0.0846      0.278     0       0            0            0              1
 8 instrumentalness         0             1      0.167       0.313     0       0            0.000216     0.102          1
 9 key                      0             1      5.20        3.52      0       2            5            8             11
10 liveness                 0             1      0.206       0.175     0       0.0988       0.136        0.261          1
11 loudness                 0             1    -11.5         5.70    -60     -14.6        -10.6         -7.18           3.86
12 mode                     0             1      0.707       0.455     0       0            1            1              1
13 popularity               0             1     31.4        21.8       0      11           33           48            100
14 speechiness              0             1      0.0984      0.163     0       0.0349       0.045        0.0756         0.97
15 tempo                    0             1    117.         30.7       0      93.4        115.         136.           244.
```

While it is lucky that there isn't any missing data, there needs to be some processing before the analysis. Specifically, there needs to be:

1) **Separating out the artists individually where the first listed artist is the main artist and everyone else are featured artist.**

```
spotify = spotify %>%
  mutate(
    artist_step_1 = gsub("\\[|\\]", "", artists) # Getting rid of the square
brackets surrounding all of the artists name
  ) %>%
  mutate(
    artist_step_2 = gsub('\\"', "", artist_step_1) # Getting rid of the doubl
e quotes surrounding the artists names
  ) %>%
  mutate(
    artist_step_3 = gsub("\\'", "", artist_step_2) # Getting rid of the singl
e quotes + apostrophes in artists names
  ) %>%
  mutate(
    more_than_one_artist = str_detect(artist_step_3, ", ") # Determine if the
re are more than 1 name associated with a given track
  ) %>%
  separate(
    artist_step_3, into = c("main.artist", "feature.artist.1", "feature.artis
t.2", "feature.artist.3", "feature.artist.4", "feature.artist.5", "feature.ar
tist.6", "feature.artist.7", "feature.artist.8", "feature.artist.9", "feature
.artist.10", "feature.artist.11", "feature.artist.12", "feature.artist.13", "
feature.artist.14", "feature.artist.15", "feature.artist.16", "feature.artist
.17", "feature.artist.18", "feature.artist.19", "feature.artist.20", "feature
.artist.21", "feature.artist.22", "feature.artist.23", "feature.artist.24", "
feature.artist.25", "feature.artist.26", "feature.artist.27", "feature.artist
.28", "feature.artist.29", "feature.artist.30", "feature.artist.31", "feature
.artist.32", "feature.artist.33", "feature.artist.34", "feature.artist.35", "
feature.artist.36", "feature.artist.37", "feature.artist.38", "feature.artist
.39"),
             sep = ","
  )
```

2) **Renaming some variable to something more appropriate**

```
spotify = spotify %>% mutate(mode = as.logical(mode), explicit = as.logical(e
xplicit))
spotify = plyr::rename(spotify, c("mode" = "is_major"))
```

3) **Conversion of duration into a more appropriate unit of time instead of milliseconds**

```
spotify = spotify %>% mutate(duration = duration_ms/1000)
```

4) **Renaming each category of the key variable based on the pitch scale**

```
spotify = spotify %>%
  mutate(
    key = as.factor(key)
  ) %>%
  mutate(
    key = plyr::revalue(key, c('0'='A', '1'='A#/Bb','2'='B','3'='C','4'='C#/D
b','5'='D','6'='D#/Eb','7'='E','8'='F','9'='F#/Gb','10'='G','11'='G#/Ab'))
  )

View(spotify)
```

5) **Removing duplicate tracks based on the criteria that songs that were from the same artist(s), same year of release and same title are excluded from the final data set.**

NOTE: I've kept the featured artist count to 5 since it's really unlikely there will be more than 5 artist on a given track where it will be meaningful.

```
spotify_updated = spotify %>%
  distinct(name, main.artist, year, feature.artist.1, feature.artist.2, featu
re.artist.3, feature.artist.4, feature.artist.5, more_than_one_artist, .keep_
all = T)
```

6) **Removing foreign artists as we're only interested in those that are known in the English-Speaking world**

```
is_foreign_artist_check = function(string) {
  num_of_char = 0
  splitted_string = str_split(string, "")
  splitted_string = splitted_string[[1]]

  for(char in splitted_string) {
    if (str_detect(char, "[^ -~]") == T) {
      num_of_char = num_of_char + 1
    }
  }

  check_status = ifelse(num_of_char >= 5, "yes", 'no')
  return(check_status)
}

list_of_main_artist = spotify_updated$main.artist
placeholder = lapply(list_of_main_artist, is_foreign_artist_check)
is_foreign = unlist(placeholder)
spotify_updated = cbind(spotify_updated, is_foreign)
spotify_updated = spotify_updated %>% filter(is_foreign == "no")
```

7) **Removing one-hit wonders as they really aren't going to be representative of the sound of the decade or music history**

This is essentially accomplished by identifying artists that have at least 3 songs on Spotify and essentially use the list of names as strings as a filter in the data frame. From the output, you'll need to copy paste these list of names about 100 or so at a time until all of these artists have been captured.

```
non_one_hit_wonder_artist = spotify_updated %>%
  group_by(main.artist) %>%
  dplyr::summarise(count = n()) %>%
  arrange(desc(count)) %>%
  filter(count > 2) # 8277 artists with at least 3 songs on Spotify

non_one_hit_wonder_artist = as.character(non_one_hit_wonder_artist$main.artis
t)

paste(non_one_hit_wonder_artist[1:100], collapse = "$|^")

## [1] "Francisco Canaro$|^Wolfgang Amadeus Mozart$|^FrÃ©dÃ©ric Chopin$|^Joha
nn Sebastian Bach$|^Ludwig van Beethoven$|^Frank Sinatra$|^Billie Holiday$|^I
gnacio Corsini$|^Giuseppe Verdi$|^Johnny Cash$|^Elvis Presley$|^Ella Fitzgera
ld$|^Igor Stravinsky$|^Bob Dylan$|^Georgette Heyer$|^Lata Mangeshkar$|^Dean M
artin$|^The Beach Boys$|^Miles Davis$|^The Beatles$|^The Rolling Stones$|^Gia
como Puccini$|^Queen$|^Fleetwood Mac$|^Claude Debussy$|^Lead Belly$|^Johannes
Brahms$|^Richard Wagner$|^Doris Day$|^Duke Ellington$|^Led Zeppelin$|^Shamsha
d Begum$|^Louis Armstrong$|^Umm Kulthum$|^Vicente FernÃ¡ndez$|^Bob Marley & T
he Wailers$|^Nina Simone$|^Oscar Peterson$|^Grateful Dead$|^John Williams$|^T
he Who$|^Orchestra Studio 7$|^H.P. Lovecraft$|^Giorgos Papasideris$|^Marvin G
aye$|^Nat King Cole$|^Elton John$|^Willie Nelson$|^George Strait$|^Hank Willi
ams$|^Pink Floyd$|^Sinclair Lewis$|^Stevie Wonder$|^Thelonious Monk$|^The Kin
ks$|^David Bowie$|^Waylon Jennings$|^Talking Heads$|^Robert Schumann$|^Aretha
Franklin$|^Metallica$|^Unspecified$|^Pyotr Ilyich Tchaikovsky$|^Geeta Dutt$|^
Eminem$|^Sam Cooke$|^Franz Schubert$|^Billy Joel$|^U2$|^Erik Satie$|^Count Ba
sie$|^Franz Joseph Haydn$|^John Coltrane$|^Judy Garland$|^Charles Mingus$|^Ja
vier SolÃs$|^KISS$|^Drake$|^Dolly Parton$|^Genesis$|^AC/DC$|^Mohammed Rafi$|^
Sarah Vaughan$|^Asha Bhosle$|^Neil Young$|^Peggy Lee$|^Michael Jackson$|^Stan
Getz$|^Roza Eskenazi$|^Bruce Springsteen$|^Los Tigres Del Norte$|^Red Hot Chi
li Peppers$|^Taylor Swift$|^George Frideric Handel$|^Bing Crosby$|^2Pac$|^JAY
-Z$|^Richard Strauss$|^Leonard Bernstein$|^Otis Redding"

spotify_updated = spotify_updated %>%
  mutate(
    at_least_3_songs.1 = ifelse(str_detect(main.artist, "^\\$NOT$|^\\$uicideB
oy\\$$|^\\(Hed\\) P.E.$|^\\*NSYNC$|^\\? & The Mysterians$|^\\+44$|^03 Greedo$
|^10$|^10 Years$|^100 gecs$|^101 Strings Orchestra$|^10cc$|^112$|^12 Stones$|
^13th Floor Elevators$|^1422$|^1910 Fruitgum Company$|^1986 Omega Tribe$|^2 C
hainz$|^2 LIVE CREW$|^2 Unlimited$|^20 Fingers$|^2002$|^20th Century Fox Stud
io Orchestra$|^21 Savage$|^24kGoldn$|^2NE1$|^2Pac$|^3 Doors Down$|^311$|^38 S
pecial$|^3LW$|^3OH!3$|^3rd Bass$|^4 Non Blondes$|^45 Grave$|^4Him$|^5 Seconds
```

```
of Summer$|^50 Cent$|^50 Guitars of Tommy Garrett$|^69 Boyz$|^6ix9ine$|^6LACK
$|^702$|^88rising$|^8Ball$|^8Ball & MJG$|^98Âº$|^999$|^a-ha$|^A Boogie Wit da
Hoodie$|^A Day To Remember$|^A Flock Of Seagulls$|^A Great Big World$|^A Perf
ect Circle$|^A R I Z O N A$|^A Rocket To The Moon$|^A Skylit Drive$|^A Taste
Of Honey$|^A Tribe Called Quest$|^A\\$AP Ferg$|^A\\$AP Mob$|^A\\$AP Rocky$|^A
*Teens$|^A. L. Lloyd$|^A. M. Rajah$|^A. P. Komala$|^A. R. Oza$|^A. R. Qureshi
$|^A.B. Quintanilla III$|^A.B. Quintanilla III Y Los Kumbia Kings$|^A.R. Rahm
an$|^Ã“lafur Arnalds$|^Ã“scar ChÃ¡vez$|^Ã“scar Medina$|^Ã‰dith Piaf$|^Ã‰douar
d Lalo$|^Ã\u0081ngeles Del Infierno$|^Aaliyah$|^Aaron Copland$|^Aaron Hall$|^
Aaron Kwok$|^Aaron Lewis$|^Aaron Lohr$|^Aaron Neville$|^Aaron Tippin$|^Aaron
Watson$|^Aaron Y Su Grupo Ilusion$|^Aarti Mukherji$|^Ab-Soul$|^ABBA$|^Abbasud
din Ahmed$|^ABC$|^Abdel Aziz Mahmoud$|^Abdel Halim Hafez$|^Abel Zavala$|^Abha
yapada Chatterjee$|^Abhram Bhagat$|^ABN$|^Above & Beyond$|^Abraham Goldfaden$
|^Abram Chasins$|^AC/DC$|^Academia dos Renascidos$|^Acapulco Tropical$|^Accep
t$|^Ace Frehley$|^Ace Hood$|^Ace of Base$|^Acerina Y Su Danzonera$|^Acid Bath
$|^Acker Bilk$|^Action Bronson$|^Adalberto Santiago$|^Adam Ant$|^Adam Hicks$|
^Adam Lambert$|^Adam Pascal$|^Adam Sandler$|^Adan Chalino Sanchez$|^Adele$|^A
delitas Way$|^Adema$|^Adolescents$|^Adolescents Orquesta$|^Adolfo BerÃ³n$|^Ad
olph Deutsch$|^Adolph Green$|^Adolphe Adam$|^Adolphe BÃ©rard$|^Adriana Caselo
tti$|^Adriano Celentano$|^Adriel Favela$|^Adventure Time$|^Aer$|^Aerosmith$|^
Aesop Rock$|^AFI$|^Afrika Bambaataa$|^Afrojack$|^Afroman$|^After 7$|^After Th
e Burial$|^Against Me!$|^Agathoklis Mouskas$|^Agent Orange$|^Agnostic Front$|
^Agust D$|^AgustÃ-n Barrios MangorÃ©$|^AgustÃ-n Lara$|^Ahmad Jamal$|^Ahmad Ja
mal Quintet$|^Ahmad Jamal Trio$|^Ahmed Dilawar$|^Aida Cuevas$|^Aim$|^Aimee Ma
nn$|^Air$|^Air Supply$|^Airbourne$|^Airplay$|^AJ Mitchell$|^AJ Rafael$|^AJJ$|
^AJR$|^Akira Yamaoka$|^Akon$|^Akwid$|^Al B. Sure!$|^Al Bowlly$|^Al Caiola$|^A
l Di Meola$|^Al Green$|^Al Haig Quartet$|^Al Haig Trio$|^Al Hirt$|^Al Hurrica
ne$|^Al Jarreau$|^Al Jolson$|^Al Kooper$|^Al Martino$|^Al Stewart$|^Alabama$|
^Alabama Shakes$|^Alacranes Musical$|^Alan Hawkshaw$|^Alan Jackson$|^Alan Lom
ax$|^Alan Menken$|^Alan Mills$|^Alan Silvestri$|^Alan Tam$|^Alan Walker$|^Ala
nis Morissette$|^Alannah Myles$|^Alasdair Fraser$|^Alaska Y Dinarama$|^Alban
Berg$|^Albeli$|^Albert Collins$") == T, TRUE, FALSE)
  )

# Repeat like about 43 times
```

As some names may have slipped the cracks, a check needs to be done.

```
spotify_updated %>%
  filter(at_least_3_songs.41 == F) %>%
  group_by(main.artist) %>%
  dplyr::summarise(count = n()) %>%
  arrange(desc(count)) # About 212 artists have slipped through the cracks.

## # A tibble: 12,209 x 2
##    main.artist        count
##    <chr>              <int>
##  1 Javier SolÃs        173
##  2 JAY-Z                152
##  3 James Taylor         133
```

```
##   4 Jackie Gleason       127
##   5 James Brown          103
##   6 Jack Johnson          91
##   7 J. Cole               71
##   8 Jefferson Airplane    69
##   9 Jackson Browne        65
##  10 Jack Teagarden        60
## # ... with 12,199 more rows
```

```
spotify_updated = spotify_updated %>% mutate(at_least_3_songs = at_least_3_so
ngs.43)
```

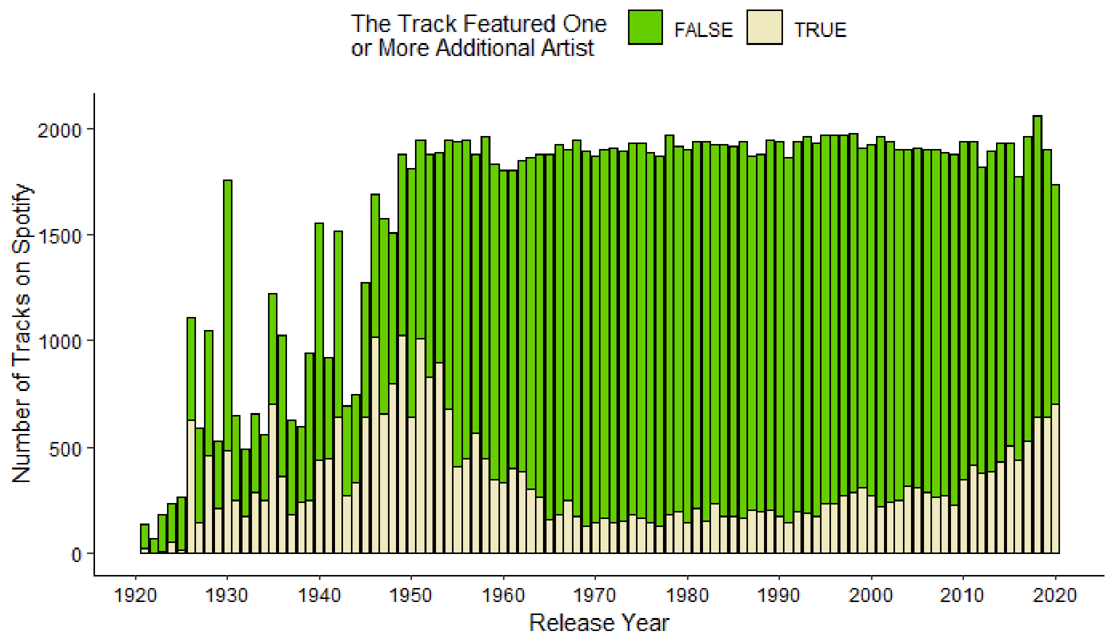
After removing the one-hit wonders, the data frame is re-organized to contain only the necessary final variables for the analysis.

```
spotify_updated = spotify_updated %>% dplyr::select(id, year, name, main.arti
st, more_than_one_artist, feature.artist.1, feature.artist.2, acousticness, a
t_least_3_songs, danceability, duration, energy, explicit, instrumentalness,
is_foreign, is_major, key, liveness, loudness, popularity, speechiness, tempo
, valence)
```

How has songs changed over time?

In order to test my dad's first claim that there is a missing quality in today's music that wasn't in the past, I've investigated to see any changes in the trends of each attribute listed above across year of release. Specifically, I took the mean of most of these features and plotting it across year of release. Below are several plots that to demonstrate these changes.

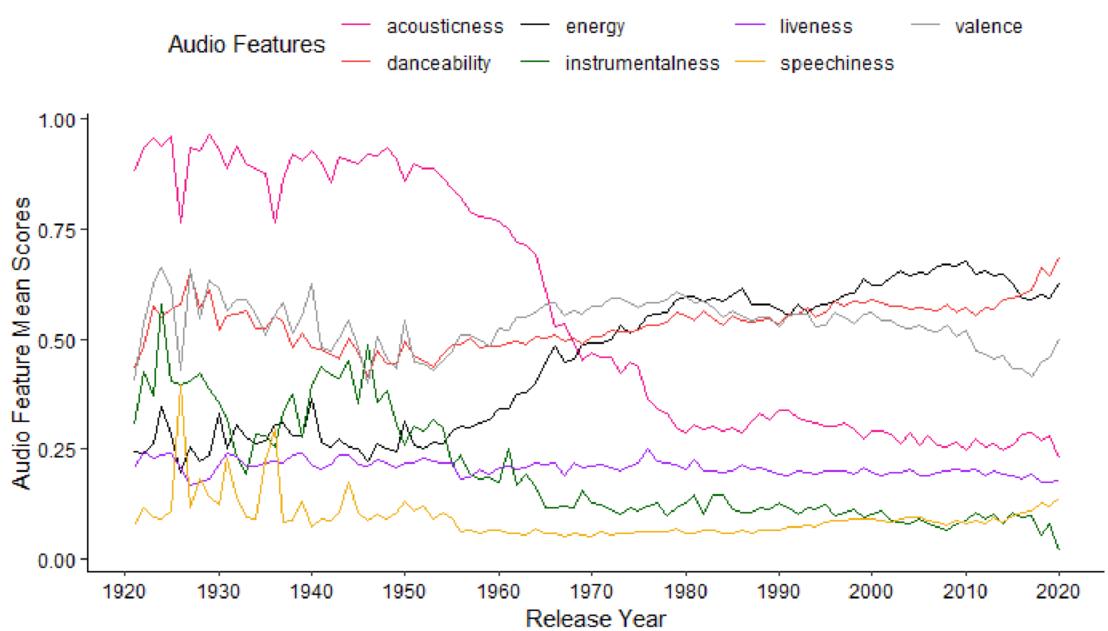
```
spotify_updated %>%
 ggplot(aes(x = year, fill = more_than_one_artist))+
  geom_bar(color = 'black') +
  theme_classic() +
  theme(
    axis.text = element_text(color = 'black'),
    plot.caption = element_text(face = 'bold', hjust = 0),
    legend.position = "top"
  ) +
  labs(x = "Release Year", y = "Number of Tracks on Spotify", caption = "Numb
er of tracks on Spotify distributed across release year by status of a featur
ed artist") +
  scale_fill_manual(name = "The Track Featured One
or More Additional Artist", values = c("chartreuse3","lemonchiffon2")) +
  scale_x_continuous(breaks =  seq(from = 1920, to = 2020, by = 10))
```



Number of tracks on Spotify distributed across release year by status of a featured artist

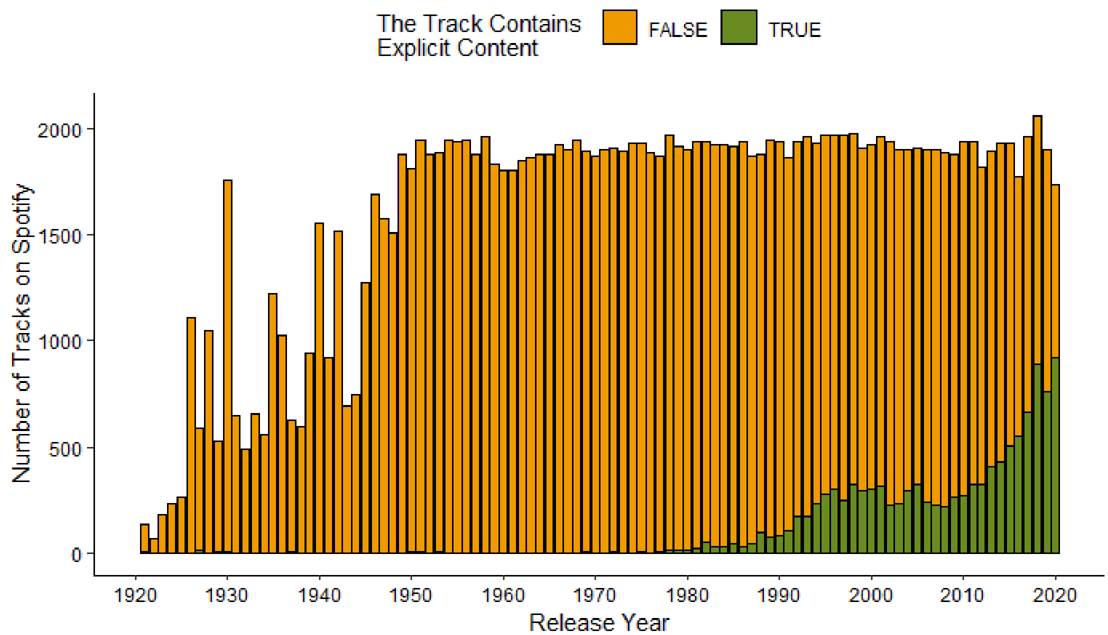
```
spotify_updated %>%
  group_by(year) %>%
  dplyr::summarise(
    acousticness = mean(acousticness),
    danceability = mean(danceability),
    energy = mean(energy),
    instrumentalness = mean(instrumentalness),
    liveness = mean(liveness),
    speechiness = mean(speechiness),
    valence = mean(valence)
  ) %>%
  ggplot(aes(x = year)) +
  geom_line(aes(y = acousticness, color = "acousticness")) +
  geom_line(aes(y = energy, color = "energy")) +
  geom_line(aes(y = danceability, color = "danceability")) +
  geom_line(aes( y = instrumentalness, color = "instrumentalness")) +
  geom_line(aes(y = liveness, color = "liveness")) +
  geom_line(aes(y = speechiness, color = "speechiness")) +
  geom_line(aes(y = valence, color = "valence")) +
  theme_classic() +
  theme(
    axis.text = element_text(color = 'black'),
    plot.caption = element_text(face = 'bold', hjust = 0),
    legend.position = "top"
    ) +
  labs(x = "Release Year",
       y = "Audio Feature Mean Scores",
       caption = "Mean audio feature scores across release year") +
     scale_color_manual(name = "Audio Features",
```

```
                          values = c("acousticness" = "deeppink2",
                                     "danceability" = "brown2",
                                     "energy" = "black",
                                     "instrumentalness" = "darkgreen",
                                     "liveness" = "purple",
                                     "speechiness" = "darkgoldenrod2",
                                     "valence"="grey59"),
  labels = c('acousticness', 'danceability', 'energy', 'instrumentalness', 'liveness', 'speechiness', 'valence')) +
  scale_x_continuous(breaks =  seq(from = 1920, to = 2020, by = 10))
```



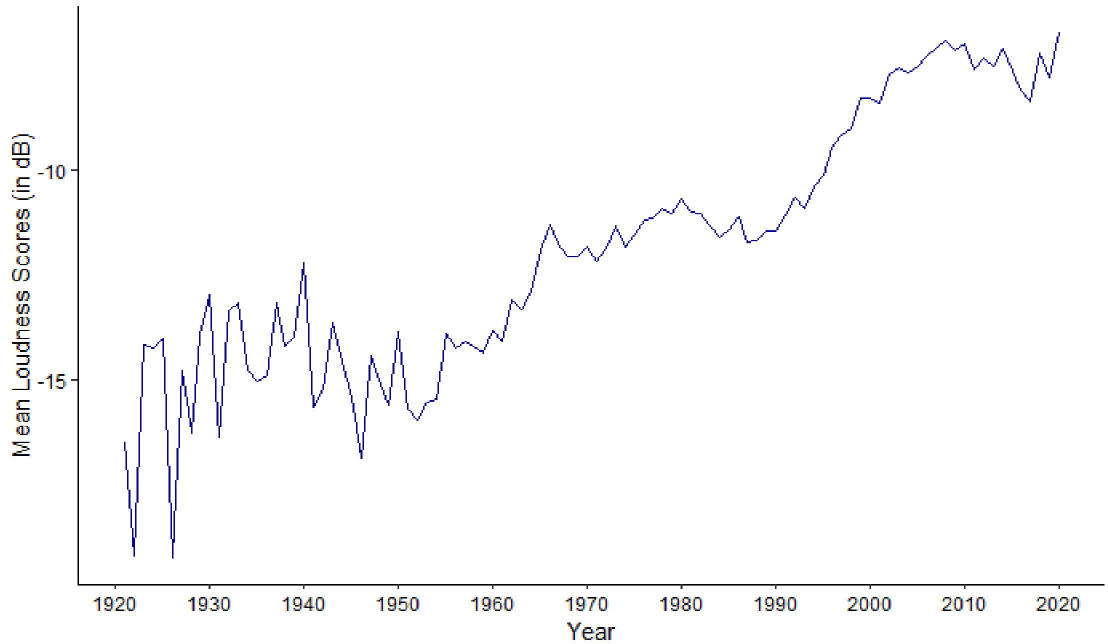
Mean audio feature scores across release year

```
spotify_updated %>%
 ggplot(aes(x = year, fill = explicit))+
  geom_bar(color = 'black') +
  theme_classic() +
  theme(
    axis.text = element_text(color = 'black'),
    plot.caption = element_text(face = 'bold', hjust = 0),
    legend.position = "top"
  ) +
  labs(x = "Release Year", y = "Number of Tracks on Spotify", caption = "Number of tracks on Spotify distributed across release year by explicit content status") +
  scale_fill_manual(name = "The Track Contains
Explicit Content", values = c("orange2","olivedrab4")) +
  scale_x_continuous(breaks = seq(from = 1920, to = 2020, by = 10))
```



Number of tracks on Spotify distributed across release year by explicit content status

```
spotify_updated %>%
  group_by(year) %>%
  dplyr::summarise(avg = mean(loudness)) %>%
  ggplot(aes(x = year, y = avg)) +
  geom_line(color = 'darkblue') +
  theme_classic() +
  theme(axis.text = element_text(color = 'black'),
    plot.caption = element_text(face = 'bold', hjust = 0)
  ) +
  labs(x = "Year", y = "Mean Loudness Scores (in dB)", caption = "Mean loudne
ss of tracks across release year") +
  scale_x_continuous(breaks = seq(from = 1920, to = 2020, by = 10))
```



Mean loudness of tracks across release year

A few observations from these plots showed that:

1) There seems to be a renaissance in the uptick of artist collaboration on a track in recent years as compared to the period from the late 60s to mid 90s.

2) Both acousticness and instrumentalness are at an all-time low as compared to in the past.

3) Conversely, energy and loudness in songs have been on a rise in recent years.

4) There is an all-time high in the number of explicit-content in tracks within recent years as compared to in the past

OK, so how does the greats of the past compare to artists today?

After creating a data frame that filtered out the data to include only the artists listed above and differentiating them as either "classic" or "current", I've compared the differences in each audio and track features.

```
acousticness = cbind(
  "Current
  (n = 17322)" = round(mean(spotify_comparison$acousticness[spotify_compariso
n$decade_group == "current"]), 3),
  "Classic
  (n = 4502)" = round(mean(spotify_comparison$acousticness[spotify_comparison
```

```
$decade_group == "classics"]), 3),
  "Statistic" = round(t.test(spotify_comparison$acousticness ~ spotify_compar
ison$decade_group)$statistic, 3),
  "p-value" = ifelse(t.test(spotify_comparison$acousticness ~ spotify_compari
son$decade_group)$p.value > 0.001, t.test(spotify_comparison$acousticness ~ s
potify_comparison$decade_group)$p.value > 0.001, "< 0.001"))

# repeat for the rest of the audio features
```

Below are the findings presented in a table using the kable package.

```
summarisation = as.table(rbind(acousticness,danceability,duration,energy,at_l
east_one_feature,explicit,instrumentalness,is_major,liveness,loudness,speechi
ness,tempo,valence))

rownames(summarisation) = c("Acousticness", "Danceability", "Duration (in sec
onds)", "Energy", "Contains Featured Artist", "Contains Explicit Content", "I
nstrumentalness", "In Major Key", "Liveness", "Loudness (in dB)", "Speechines
s", "Tempo (in BPM)", "Valence")

summarisation %>%
  kbl(longtable = T,  caption = "Comparison of Audio & Track Features in Song
s between Past and Current Top Artists") %>%
  kable_classic_2(full_width = T, html_font = "Cambria") %>%
  row_spec(0, bold = T, background = "darkblue", color = "white") %>%
  column_spec(1, bold = T) %>%
  footnote(
    number = c("Acousticness, danceability, duration, energy, instrumentalnes
s, liveness, loudness, speechiness, tempo and valence are compared using Stud
ent's T-Test",
               "Presence of a featured artist/explicit content and whether th
e track is in major key are compared using Pearson's Chi-Square Test")
  )
```

Comparison of Audio & Track Features in Songs between Past and Current Top Artists

	Current (n = 17322)	Classic (n = 4502)	Statistic	p-value
Acousticness	0.199	0.31	25.566	< 0.001
Danceability	0.61	0.54	-27.564	< 0.001
Duration (in seconds)	232.452	251.978	15.886	< 0.001
Energy	0.666	0.596	-20.582	< 0.001
Contains Featured Artist	55.87 %	44.13 %	1606.511	< 0.001
Contains Explicit Content	94.64 %	5.36 %	5935.476	< 0.001
Instrumentalness	0.031	0.073	17.013	< 0.001
In Major Key	17.13 %	82.87 %	425.286	< 0.001
Liveness	0.196	0.229	11.546	< 0.001
Loudness (in dB)	-6.415	-10.308	-74.203	< 0.001
Speechiness	0.115	0.054	-33.162	< 0.001
Tempo (in BPM)	121.375	121.456	0.169	1
Valence	0.47	0.58	28.643	< 0.001

[1] Acousticness, danceability, duration, energy, instrumentalness, liveness, loudness, speechiness, tempo and valence are compared using Student's T-Test

[2] Presence of a featured artist/explicit content and whether the track is in major key are compared using Pearson's Chi-Square Test

Overall, it appears that aside from tempo, there were significant differences across the board. Today's top artists appear to really be into the electronic sound with more speech-like lyrical content (thanks Hip Hop + R&B). On top of that, there appears to be a moodier music despite tracks having more energy in them.

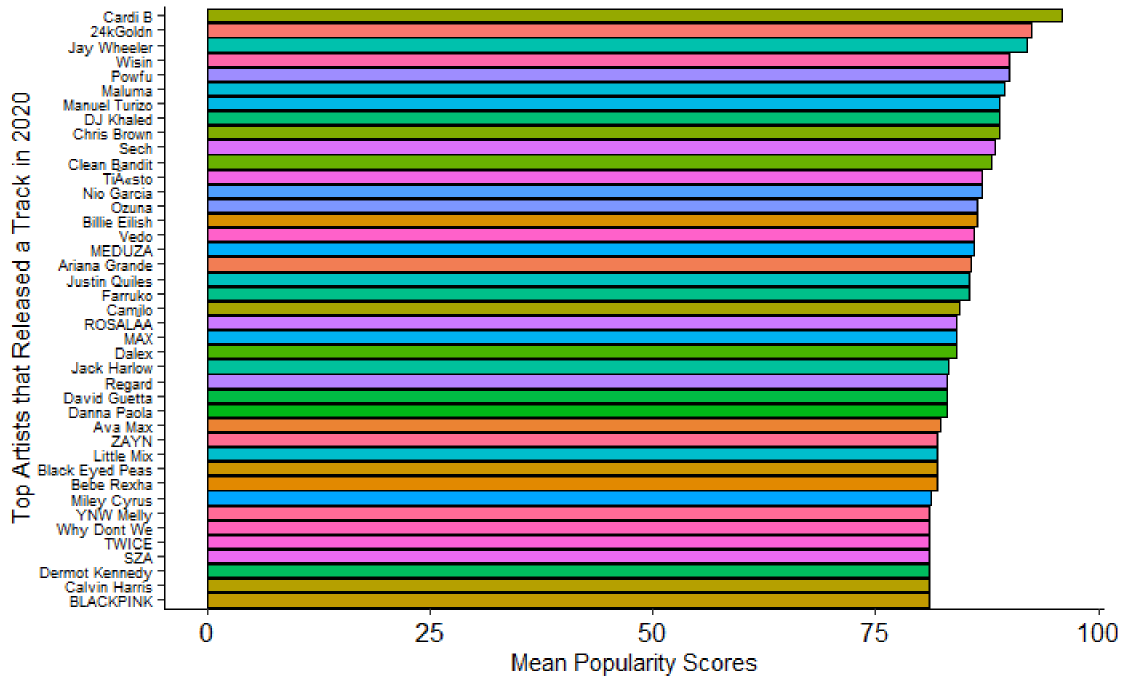
So, who are young people listening to?

Working on the assumption that the vast majority of Spotify users are young people, here's a look at the top 40 artists that had released a track in 2020, according to average popularity score across in that year.

NOTE: This excluded those who did not have at least 3 tracks on Spotify.

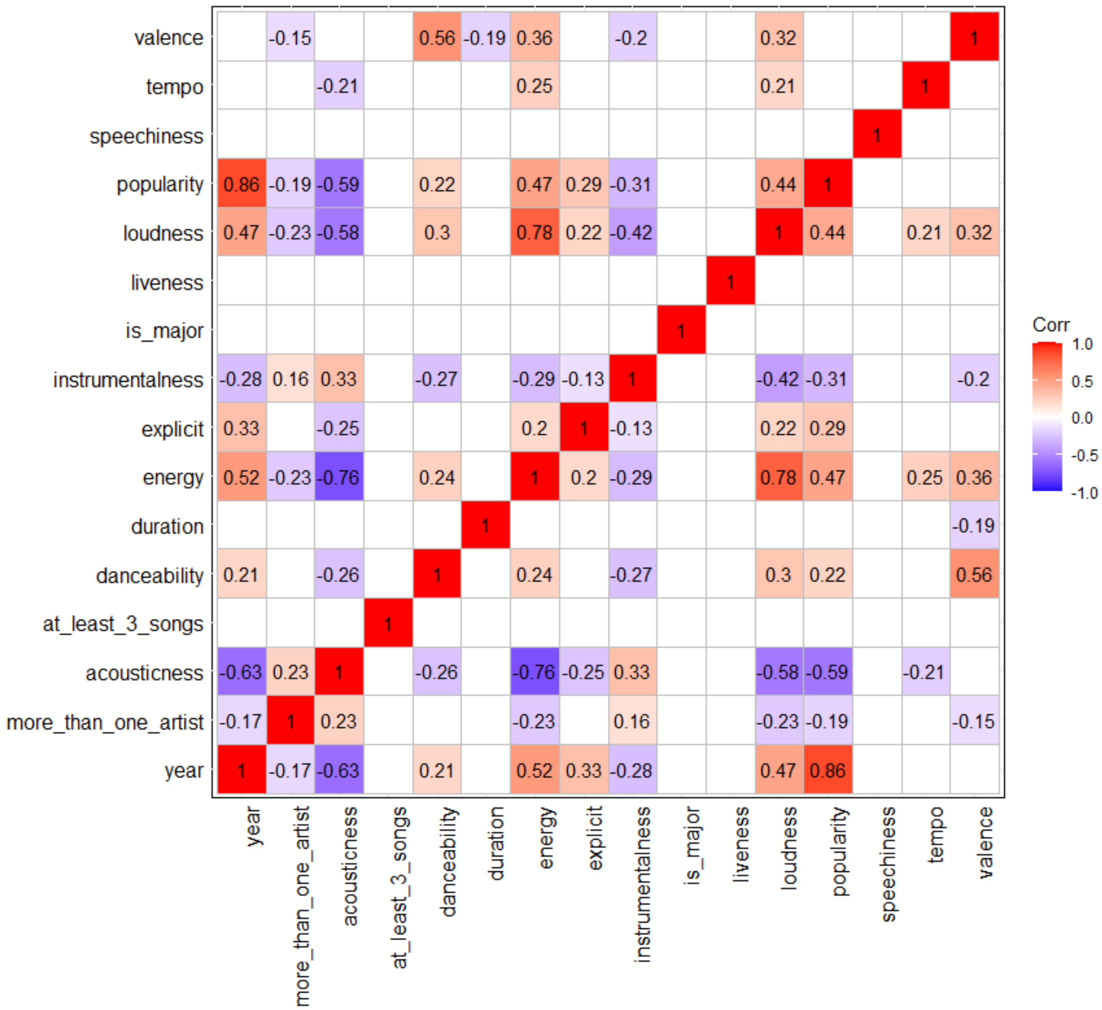
```
spotify_updated_popular = spotify_updated %>%
  mutate(
    main.artist_factor = as.factor(main.artist)
  )

spotify_updated_popular%>%
  filter(is_foreign == "no" & at_least_3_songs == T & year == 2020) %>%
  group_by(main.artist_factor) %>%
  dplyr::summarise(mean_popular= mean(popularity)) %>%
  arrange(desc(mean_popular)) %>%
  top_n(40) %>%
  ggplot(aes(x = reorder(main.artist_factor, mean_popular), y = mean_popular,
fill = main.artist_factor)) +
  geom_bar(stat = 'identity', color = 'black') +
  theme_classic() +
  theme(
```

```
    axis.text = element_text(color = 'black', family = "sans"),
    axis.text.y = element_text(color = 'black', size = 6.5),
    axis.text.x = element_text(size = 12),
    plot.caption = element_text(hjust = 0)
  ) +
  coord_flip() +
  labs(x = "Top Artists that Released a Track in 2020",
       y = "Mean Popularity Scores")+
  guides(fill = F)
```



A quick look here showed a mix of popular Hip Hop and R&B, pop and Latin artists. Interestingly, it appears as though the majority of these artists debuted within the past 5 years or so. Exploring further with a correlation matrix, with the ggcorplot package, it seems that popularity is strongly tied to release year.

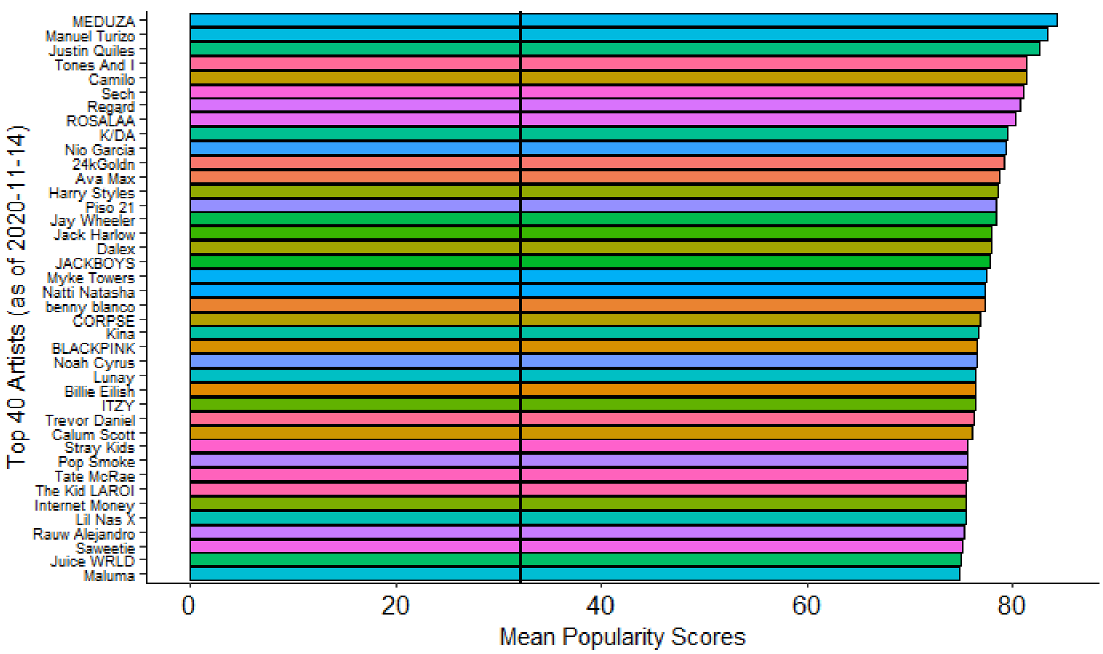
```
spotify_updated_correlation = spotify_updated %>% dplyr::select(-id, -name, -
main.artist, -feature.artist.1, -feature.artist.2, -key, -is_foreign)

corr_spotify_updated <- cor(spotify_updated_correlation)

ggcorrplot(corr_spotify_updated, method = 'square', type = 'full', sig.level
= 0.1, insig = 'blank', p.mat = cor_pmat(corr_spotify_updated), lab = T) +
  theme(
    panel.background = element_rect(fill = 'white'),
    axis.text.x = element_text(angle = 90, hjust = 1),
    axis.text =  element_text(color = 'black')  ,
    plot.caption = element_text(face = "bold", hjust = 0)
  )
```



This point is further emphasized once we compare the mean popularity scores of the most popular artists in 2020 with the all-time top 40 artists. Looking at the comparison, it seems that in spite of the skew towards more contemporary artists, a great deal of listeners is also listening to the greats of the past like Michael Jackson, The Beatles and Prince.

```
spotify_updated_popular%>%
  filter(is_foreign == "no" & at_least_3_songs == T) %>%
  group_by(main.artist_factor) %>%
  dplyr::summarise(mean_popular= mean(popularity)) %>%
  arrange(desc(mean_popular)) %>%
  top_n(40) %>%
  ggplot(aes(x = reorder(main.artist_factor, mean_popular), y = mean_popular,
fill = main.artist_factor)) +
  geom_bar(stat = 'identity', color = 'black') +
  geom_hline(yintercept = mean(spotify_updated$popularity),  color = "black",
size = 1) +
```

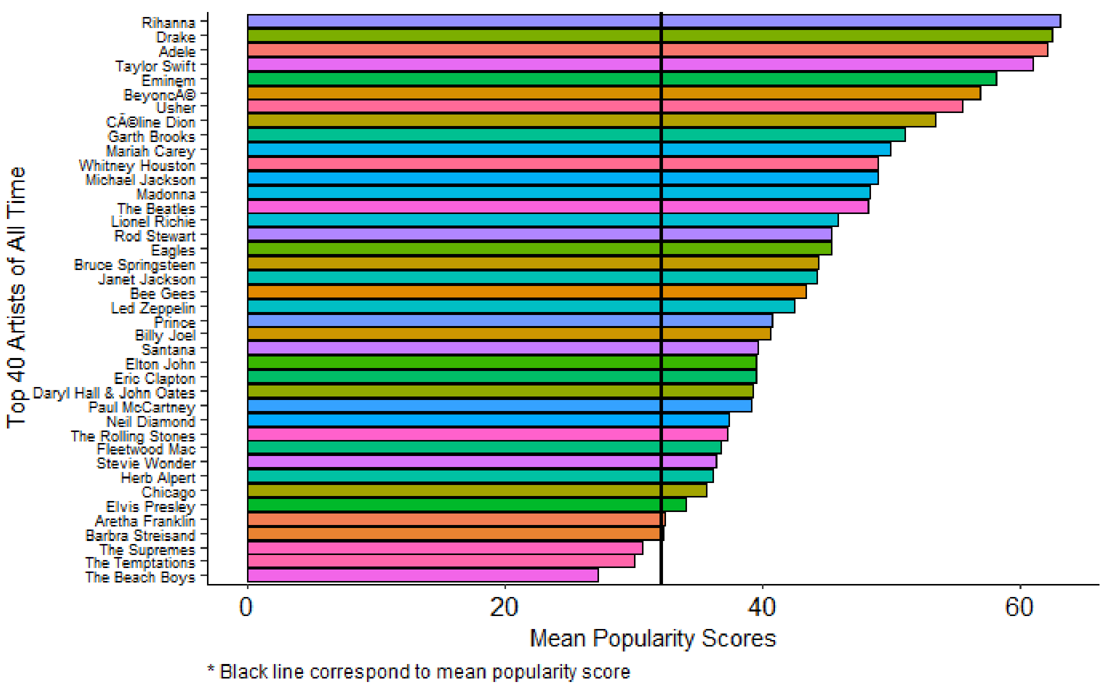
```
  theme_classic() +
  theme(
    axis.text = element_text(color = 'black', family = "sans"),
    axis.text.y = element_text(color = 'black', size = 6.5),
    axis.text.x = element_text(size = 12),
    plot.caption = element_text(hjust = 0)
  ) +
  coord_flip() +
  labs(x = "Top 40 Artists (as of 2020-11-14)",
       y = "Mean Popularity Scores",
       caption = "* Black line correspond to mean popularity score") +
  guides(fill = F)

## `summarise()` ungrouping output (override with `.groups` argument)

## Selecting by mean_popular
```



* Black line correspond to mean popularity score

```
spotify_updated_popular%>%
  filter(
    str_detect(main.artist, "^Led Zeppelin$|^Lionel Richie$|^Eric Clapton$|^B
eyoncÃ©$|^Adele$|^Aretha Franklin$|^Daryl Hall & John Oates$|^The Temptations
$|^CÃ©line Dion$|^Santana$|^Fleetwood Mac$|^The Beach Boys$|^Bee Gees$|^Eagle
s$|^Neil Diamond$|^The Supremes$|^Bruce Springsteen$|^Usher$|^Eminem$|^Garth
Brooks$|^Herb Alpert$|^Billy Joel$|^Rihanna$|^Prince$|^Drake$|^Rod Stewart$|^
Janet Jackson$|^Elvis Presley$|^Paul McCartney$|^Whitney Houston$|^Chicago$|^
Taylor Swift$|^Michael Jackson$|^Madonna$|^Mariah Carey$|^Barbra Streisand$|^
Elton John$|^The Rolling Stones$|^The Beatles$|^Stevie Wonder$") == T) %>%
  group_by(main.artist_factor) %>%
  dplyr::summarise(mean_popular= mean(popularity)) %>%
  arrange(desc(mean_popular)) %>%
```

```
  top_n(40) %>%
  ggplot(aes(x = reorder(main.artist_factor, mean_popular), y = mean_popular,
fill = main.artist_factor)) +
  geom_bar(stat = 'identity', color = 'black') +
  geom_hline(yintercept = mean(spotify_updated$popularity),  color = "black",
size = 1) +
  theme_classic() +
  theme(
    axis.text = element_text(color = 'black', family = "sans"),
    axis.text.y = element_text(color = 'black', size = 6.5),
    axis.text.x = element_text(size = 12),
    plot.caption = element_text(hjust = 0)
  ) +
  coord_flip() +
  labs(x = "Top 40 Artists of All Time",
       y = "Mean Popularity Scores",
       caption = "* Black line correspond to mean popularity score") +
  guides(fill = F)
```



The Verdict

Overall, it seems my old man might be right about the change in sound in today's music. We've appeared to have moved away from playing instruments and that "live" sound to someone making beats with a push of a button and a louder or more energetic sound. However, that's not say we don't have any taste in music as a whole. Sure, the apparent preference to harder content is there that may not be to the taste of the older generation, but many of us still enjoy the hits of the past as well. Hell, there's no way anyone isn't going to enjoy listening to some Earth, Wind and Fire on a sunny day.

If there are any typos or inconsistency, or would like to ask or say something, just drop a comment. For the entire code output and process, along with other projects, check out my GitHub.

Thanks for reading.