Fall 2022

Top Spotify Music

CIS 5250: Visual Analytics – R Project

Team Member
Phue Thant
Ruta Antaliya

Table of Contents

Objective

Listening to music is a daily habit for most people nowadays in this era, whether you go to university on a public transportation or drive your car to your workplace or school or vacation. Spotify is a media and audio streaming service based in Sweden that debuted in October 2008. The platform began as a way to allow listeners to stream their favorite songs while still compensating artists for their work – a major issue caused by file sharing services at the time, like Napster and LimeWire, which severely affected music sales as the services had no legal rights to the music. It is now one of the most popular digital music, podcast, and video streaming services in the world, offering access to millions of songs by artists from all over the world. A journal article published in 2020 (Werner, 2020) mentioned that Spotify had more than 232 million active monthly music listeners back in July 2019, with over 108 million people willing to pay the subscription fees monthly. Like YouTube, another well-known video streaming platform, Spotify allows users to do personal track searches and provides recommending playlists using an algorithm.

The research was conducted in 2020 on finding out the algorithmic effects (Anderson, 2020) of Spotify's programmed playlists on users. They found that Spotify's programmed playlists ranked by relevance to music listeners generated more motivation and streamed more songs. Spotify's success quickly caught the eye of major technology competitors, who have since released their own streaming music platforms such as Apple Music, YouTube Music, and Amazon Music from article (Collins, 2022).

We found these topics interesting because we are a daily user of Spotify, and always listen to different types of tracks and music is a part of communication in our lives as well. This data

analysis project will provide insights into the current Spotify top tracks popularity and give an objective data analysis on tracks by its audio feature, year, and country. It will show how audio features related to a song's popularity and how was it changed over time. This dataset contains information about popular song, artist, year, genre, loudness, and energy of the song. This is primarily beneficial to promote songs to Spotify listeners and enhance their user experience.

In this project, we will examine the popularity of a track using a variety of audio elements from the dataset to see if we can forecast it based on important song-related details. We want to analyze users' listening habits so that Spotify can purchase and recommend related tunes on their platform and enhance user experience. To help engineers and the marketing team better segment users, evaluate trends, and try to enhance revenue and improve user experience, this analysis will help to better understand the various clusters and enable Spotify to produce a better targeted content distribution.

The original dataset is from Kaggle, and the link is stated below.

Dataset URL - https://www.kaggle.com/datasets/leonardopena/top-50-spotify-songs-by-each-country

Dataset Description

• Total Rows = 1000

• Total Columns = 17

• File = Top50List.csv

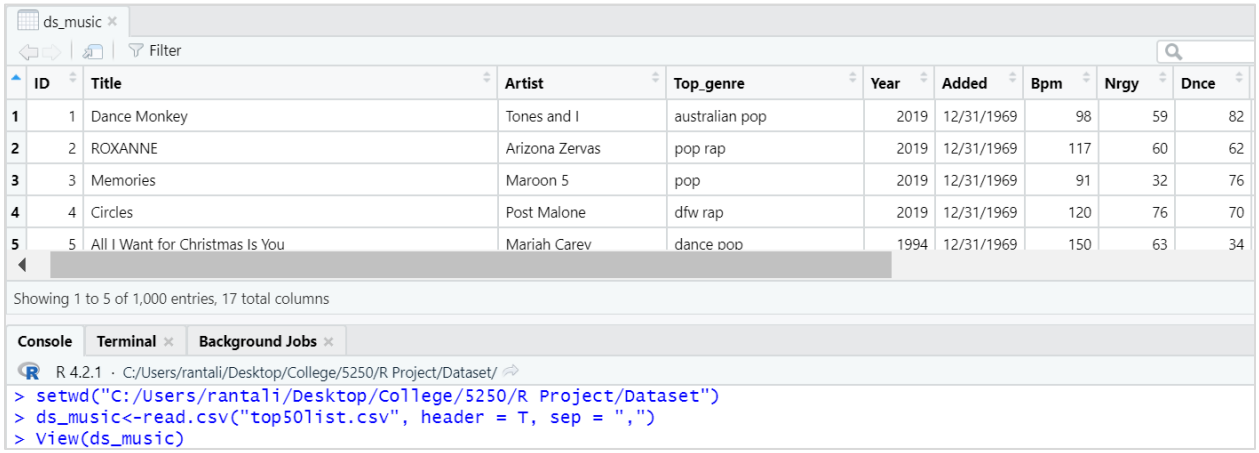
The table below describes the dataset in the details.

No.	Field Name	Data Description
1.	Title	Song's title
2.	Artist	Song's artist
3.	Top_genre	The genre of the track
4.	Year	Song's year
5.	Bpm	Beats.Per. Minute - The tempo of the song
6.	Nrgy	Energy of a song
7.	Dnce	Danceability of the song
8.	DB	Loudness of the song
9.	Live	Liveness of the song
10.	Val	Valance of a song
11.	Dur	Length of a song
12.	Acous	Acoustic the song
13.	Spch	Speechiness – more spoken word of the song
14.	Pop	Popularity of a song
15.	Country	Country where song was famous

Dataset Screenshot

ID	Title	Artist	Top_genre	Year	Added	bpm	nrgy	dnce	dB	live	val	dur	acous	spch	Popularity	Country
1	Dance Mo	Tones and	australian p	2019	12/31/1969	98	59	82	-6	15	51	209	69	9	100	world
2	ROXANNE	Arizona Ze	pop rap	2019	12/31/1969	117	60	62	-6	46	46	164	5	15	99	world
3	Memories	Maroon 5	pop	2019	12/31/1969	91	32	76	-7	8	57	189	84	5	99	world
4	Circles	Post Malo	dfw rap	2019	12/31/1969	120	76	70	-3	9	55	215	19	4	99	world
5	All I Want	Mariah Ca	dance pop	1994	12/31/1969	150	63	34	-7	7	35	241	16	4	98	world
6	everything	Billie Eilish	electropop	2019	12/31/1969	120	23	70	-14	11	24	245	90	10	98	world
7	Falling	Trevor Da	alternative r	2018	12/31/1969	127	43	78	-9	9	24	159	12	4	97	world
8	RITMO (Ba	The Black	dance pop	2019	12/31/1969	105	72	72	-7	24	67	222	3	7	97	world
9	Don't Star	Dua Lipa	dance pop	2019	12/31/1969	124	79	79	-5	10	68	183	1	8	97	world
10	Tusa	KAROL G	latin	2019	12/31/1969	101	72	80	-3	6	57	201	30	30	96	world

Uploading Dataset into R




```
> dim(ds_music)
[1] 1000   17
> head(ds_music)
  ID                              Title         Artist      Top_genre Year      Added Bpm Nrgy Dnce  DB Live Val Dur
1  1                        Dance Monkey    Tones and I australian pop 2019 12/31/1969  98   59   82  -6   15  51 209
2  2                             ROXANNE Arizona Zervas        pop rap 2019 12/31/1969 117   60   62  -6   46  46 164
3  3                            Memories       Maroon 5            pop 2019 12/31/1969  91   32   76  -7    8  57 189
4  4                             Circles    Post Malone        dfw rap 2019 12/31/1969 120   76   70  -3    9  55 215
5  5 All I Want for Christmas Is You    Mariah Carey      dance pop 1994 12/31/1969 150   63   34  -7    7  35 241
6  6             everything i wanted  Billie Eilish     electropop 2019 12/31/1969 120   23   70 -14   11  24 245
  Acous Spch Popularity Country
1    69    9        100   world
2     5   15         99   world
3    84    5         99   world
4    19    4         99   world
5    16    4         98   world
6    90   10         98   world
```

```
> str(ds_music)
'data.frame':   1000 obs. of  17 variables:
 $ ID        : int  1 2 3 4 5 6 7 8 9 10 ...
 $ Title     : chr  "Dance Monkey" "ROXANNE" "Memories" "Circles" ...
 $ Artist    : chr  "Tones and I" "Arizona Zervas" "Maroon 5" "Post Malone" ...
 $ Top_genre : chr  "australian pop" "pop rap" "pop" "dfw rap" ...
 $ Year      : int  2019 2019 2019 2019 1994 2019 2018 2019 2019 2019 ...
 $ Added     : chr  "12/31/1969" "12/31/1969" "12/31/1969" "12/31/1969" ...
 $ Bpm       : int  98 117 91 120 150 120 127 105 124 101 ...
 $ Nrgy      : int  59 60 32 76 63 23 43 72 79 72 ...
 $ Dnce      : int  82 62 76 70 34 70 78 72 79 80 ...
 $ DB        : int  -6 -6 -7 -3 -7 -14 -9 -7 -5 -3 ...
 $ Live      : int  15 46 8 9 7 11 9 24 10 6 ...
 $ Val       : int  51 46 57 55 35 24 24 67 68 57 ...
 $ Dur       : int  209 164 189 215 241 245 159 222 183 201 ...
 $ Acous     : int  69 5 84 19 16 90 12 3 1 30 ...
 $ Spch      : int  9 15 5 4 4 10 4 7 8 30 ...
 $ Popularity: int  100 99 99 99 98 98 97 97 97 96 ...
 $ Country   : chr  "world" "world" "world" "world" ...
```

```
> setwd ("C:/Users/rantali/Desktop/College/5250/R Project/Dataset")

> ds_music<-read.csv ("top50list.csv", header = T, sep = "")

> View(ds_music)

> dim(ds_music)

[1] 1000   17

> head(ds_music)

> str(ds_music)
```

Data Cleaning

1. Delete irrelevant Columns: “ID” & “ADDED”

Removing unnecessary columns from the table. This data cleaning technique demonstrates the technique of removing the specified columns which are not useful and needed for the analysis. We selected the columns “ID” and “Added” from the table and removed them. The reason we removed

these two columns is that R Studio has the ID column itself and the "ID" from the original data might be confusing for the users. For "Added" column, it has the wrong and unusual data as it is only one exact date for all the songs listed.

Original Data

ID	Title	Artist	Top_genre	Year	Added	bpm
1	Dance Monkey	Tones and I	australian pop	2019	12/31/1969	98
2	ROXANNE	Arizona Zervas	pop rap	2019	12/31/1969	117
3	Memories	Maroon 5	pop	2019	12/31/1969	91
4	Circles	Post Malone	dfw rap	2019	12/31/1969	120
5	All I Want for Christmas Is You	Mariah Carey	dance pop	1994	12/31/1969	150

R Code for removing column:

```
> setwd("C:/Users/rantali/Desktop/College/5250/R Project/Dataset")
> ds_music<-read.csv("top50list.csv", header = T, sep = ",")
> View(ds_music)
> data_col_1_6<-ds_music[,-c(1,6)]
> View(data_col_1_6)
```

> setwd("C:/Users/rantali/Desktop/College/5250/R Project/Dataset")

> ds_music<-read.csv ("top50list.csv", header = T, sep = ",")

> View(ds_music)

> data_col_1_6<-ds_music[, -c(1,6)]

> View(data_col_1_6)

After removing unnecessary column below table created.

Modified Data

	Title	Artist	Top_genre	Year	Bpm
1	Dance Monkey	Tones and I	australian pop	2019	98
2	ROXANNE	Arizona Zervas	pop rap	2019	117
3	Memories	Maroon 5	pop	2019	91
4	Circles	Post Malone	dfw rap	2019	120
5	All I Want for Christmas Is You	Mariah Carey	dance pop	1994	150

2. Delete irrelevant rows: Remove NA values in "Top_genre" column

The second data cleaning we have done is checking "NA" or "NULL" values and removing them from the table. Removing null values from the dataset is one of the important steps in data cleaning. These null values adversely affect the performance and accuracy of data analysis. In this project, we have the many "NA" Values in "Top_genre" column and it will have effects the result of the music related to the genre. So, we have removed all the "NA" values in genre column.

Code for check the "NA" values:

```
> is.na(data_col_1_6)
        Title Artist Top_genre  Year   bpm  nrgy  dnce    dB  live   val   dur acous  spch Popularity
   [1,] FALSE  FALSE     FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE      FALSE
   [2,] FALSE  FALSE     FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE      FALSE
   [3,] FALSE  FALSE     FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE      FALSE
   [4,] FALSE  FALSE     FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE      FALSE
   [5,] FALSE  FALSE     FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE      FALSE
   [6,] FALSE  FALSE     FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE      FALSE
   [7,] FALSE  FALSE     FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE      FALSE
   [8,] FALSE  FALSE     FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE      FALSE
   [9,] FALSE  FALSE     FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE      FALSE
  [10,] FALSE  FALSE     FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE      FALSE
  [11,] FALSE  FALSE     FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE      FALSE
  [12,] FALSE  FALSE     FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE      FALSE
  [13,] FALSE  FALSE     FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE      FALSE
  [14,] FALSE  FALSE     FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE      FALSE
  [15,] FALSE  FALSE     FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE      FALSE
  [16,] FALSE  FALSE     FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE      FALSE
  [17,] FALSE  FALSE     FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE      FALSE
  [18,] FALSE  FALSE     FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE      FALSE
  [19,] FALSE  FALSE     FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE      FALSE
  [20,] FALSE  FALSE      TRUE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE      FALSE
  [21,] FALSE  FALSE     FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE      FALSE
  [22,] FALSE  FALSE     FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE      FALSE
  [23,] FALSE  FALSE     FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE      FALSE
  [24,] FALSE  FALSE      TRUE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE      FALSE
  [25,] FALSE  FALSE      TRUE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE      FALSE
```

Original Data

	Title	Artist	Top_genre	Year	Bpm
371	Do They Know It's Christmas?	Band Aid	NA	2004	115
417	Do They Know It's Christmas?	Band Aid	NA	2004	115
434	Wonderful Dream	Melanie Thornton	NA	2001	150
478	Loco	Beele	NA	2019	105
624	Do They Know It's Christmas?	Band Aid	NA	2004	115

R Codes for Removing NA value:

```
> setwd("C:/Users/rantali/Desktop/College/5250/R Project/Dataset")
> View(data_col_1_6)
> delete_genre_raw<-na.omit(data_col_1_6)
> View(delete_genre_raw)
```

> setwd("C:/Users/rantali/Desktop/College/5250/R Project/Dataset")

> View(data_col_1_6)

> delete_genre_raw<-na.omit(data_col_1_6)

> View(delete_genre_raw)

After removing the NA value below table generated.

Modified Data

	Title	Artist	Top_genre	Year	Bpm
372	Blue Christmas	Elvis Presley	adult standards	1957	95
373	Run Rudolph Run	Chuck Berry	blues rock	1986	152
374	Santa Baby	Ariana Grande	dance pop	2013	96
375	Like It's Christmas	Jonas Brothers	boy band	2019	146
376	Step Into Christmas	Elton John	glam rock	1973	140

3. Delete row with zero value: Remove 0 values in "Popularity" column

The third data cleaning we have done is removed popularity rows with zero value. For better visualization purpose we removed zero value.

Original Data

Spch	Popularity	Country
4	0	japan
5	37	africa
11	39	japan
3	43	israel
4	45	australia

R Codes for remove column with zero value:

```
> View(delete_genre_raw)
> dim(delete_genre_raw)
[1] 924  15
> ds_popularity<-delete_genre_raw[!(delete_genre_raw$Popularity=="0"),]
> View(ds_popularity)
> dim(ds_popularity)
[1] 923  15
```

> View(delete_genre_raw)
> dim(delete_genre_raw)
[1] 924 15
> ds_popularity<-delete_genre_raw[!(delete_genre_raw$Popularity=="0"),]
> View(ds_popularity)
> dim(ds_popularity)
[1] 923 15

After removing the data raw with zero value.

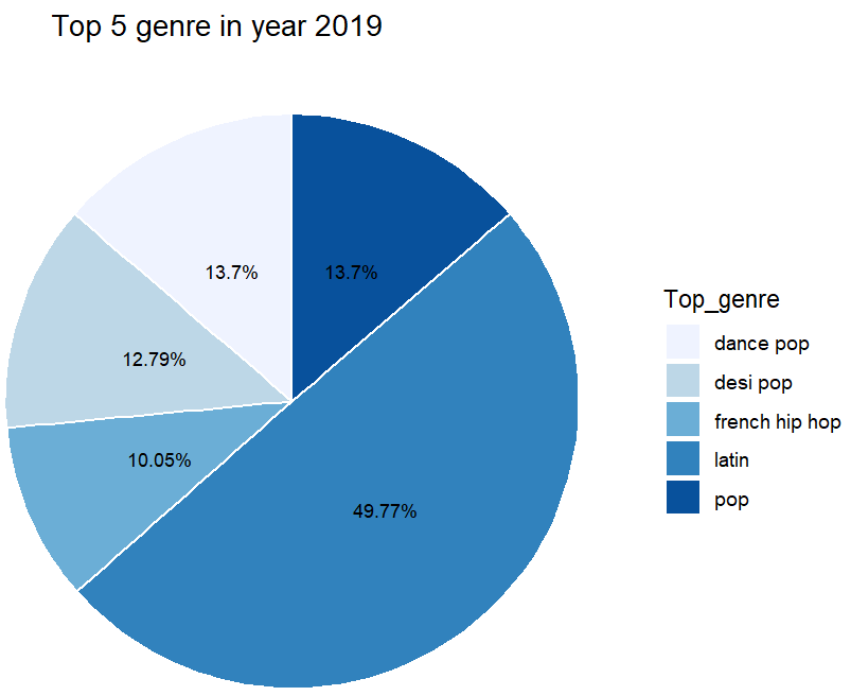
Modified Data

Spch	Popularity	Country
5	37	africa
11	39	japan
3	43	israel
4	45	australia
4	45	colombia

Data Visualization

Analysis Questions

1. **In 2019, what are the top** 5 genres in the top 50 Spotify Tracks??



Code for filter Top 5 genre in year 2019

```
> install.packages("dplyr")
Error in install.packages : Updating loaded packages
> install.packages("dplyr")
WARNING: Rtools is required to build R packages but is not currently installed. Ple
ase download and install the appropriate version of Rtools before proceeding:

https://cran.rstudio.com/bin/windows/Rtools/
Warning in install.packages :
  package 'dplyr' is in use and will not be installed
> library(dplyr)
> Music_Top5_genre<-ds_music%>%
+ filter(Year==2019)%>%
+ count(Top_genre, sort = TRUE)%>%
+ head(5)

> View(Music_Top5_genre)
> Music_Top5_genre<-Music_Top5_genre%>%
+ mutate(Top_genre_per = n / sum(n) * 100)
> View(Music_Top5_genre)
```

> install.packages("dplyr")

> library(dplyr)

> Music_Top5_genre<-ds_music%>%

+ filter(Year==2019)%>%

+ count(Top_genre, sort = TRUE)%>%

+ head(5)

> View(Music_Top5_genre)

Music_Top5_genre<-Music_Top5_genre%>%

+ mutate(Top_genre_per = n / sum(n) * 100)

> View(Music_Top5_genre)

Result of the filter data.

	Top_genre	n	Top_genre_per
1	latin	109	49.77169
2	dance pop	30	13.69863
3	pop	30	13.69863
4	desi pop	28	12.78539
5	french hip hop	22	10.04566

Code For Visualization (Pie Chart)

```
> install.packages("ggplot2")
WARNING: Rtools is required to build R packages but is not currently installed. Please
 download and install the appropriate version of Rtools before proceeding:

https://cran.rstudio.com/bin/windows/Rtools/
Warning in install.packages :
  package 'ggplot2' is in use and will not be installed
> install.packages("Scales")
WARNING: Rtools is required to build R packages but is not currently installed. Please
 download and install the appropriate version of Rtools before proceeding:

https://cran.rstudio.com/bin/windows/Rtools/
Installing package into 'C:/Users/rantali/AppData/Local/R/win-library/4.2'
(as 'lib' is unspecified)
Warning in install.packages :
  package 'Scales' is not available for this version of R

A version of this package for your version of R might be available elsewhere,
see the ideas at
https://cran.r-project.org/doc/manuals/r-patched/R-admin.html#Installing-packages
Warning in install.packages :
  Perhaps you meant 'scales' ?
> install.packages("RColorBrewer")
Error in install.packages : Updating loaded packages
> install.packages("RColorBrewer")
```

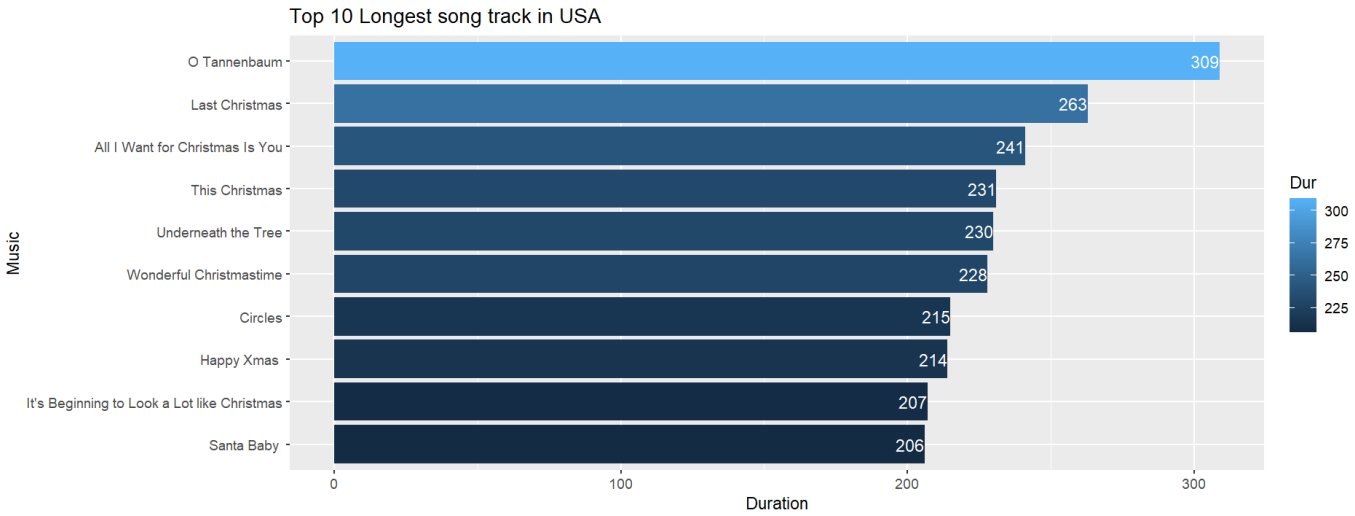
```
> library(ggplot2)
> library(scales)
> library(RColorBrewer)
> ggplot(Music_Top5_genre, aes(x="", y=Top_genre_per, fill=Top_genre)) +
 geom_bar(width = 0.5, stat = "identity", color = "white") + coord_polar
("y", start = 0) + geom_text(aes(label = paste0(round(Top_genre_per,
2),"%")), position = position_stack(vjust = 0.5), size=3) + labs(x = "",
 y = "", title = "Top 5 genre in year 2019") + scale_fill_brewer("Top_gen
re") + theme_void()
```

```
> install.packages("ggplot2")
> install.packages("scales")
> install.packages("RColorBrewer ")
> library(ggplot2)
> library(scales)
> library(RColorBrewer)
> ggplot(Music_Top5_genre, aes(x="", y=Top_genre_per, fill=Top_genre)) +
geom_bar(width = 0.5, stat = "identity", color = "white") + coord_polar("y", start = 0) +
geom_text(aes(label = paste0(round(Top_genre_per,2),"%")), position = position_stack(vjust
= 0.5), size=3) + labs(x = "", y = "", title = "Top 5 genre in year 2019") +
scale_fill_brewer("Top_genre") + theme_void()
```

Description of Pie Chart:

After running the code for TOP 5 genre, we got the results as shown in pie chart above. From it can be seen that “Latin” genre was the most famous of all the genre with 49.77% popularity out of top 50 tracks of Spotify from the year 2019. Other 2 genre have almost the equal popularity i.e. “Pop” and “Dance Pop” with 13.7%. Looking at graph other genre like “Desi Pop” and “French Hip Hop” approximately 10.05% and 12.79% popular in year 2019 among all country.

2. **Top 10 longest song track in the USA.**



Code for filter Top 10 music in USA country

```
> View(ds_music)
> library(dplyr)
> Top_music<-ds_music %>%
+ filter(Country == 'usa')%>%
+ top_n(10,Dur)
> View(Top_music)
```

> View(ds_music)

> library(dplyr)

> Top_music<-ds_music %>%

+ filter(Country == 'usa')%>%

+ top_n(10,Dur)

> View(Top_music)

Code For Vizualization (Column Chart)

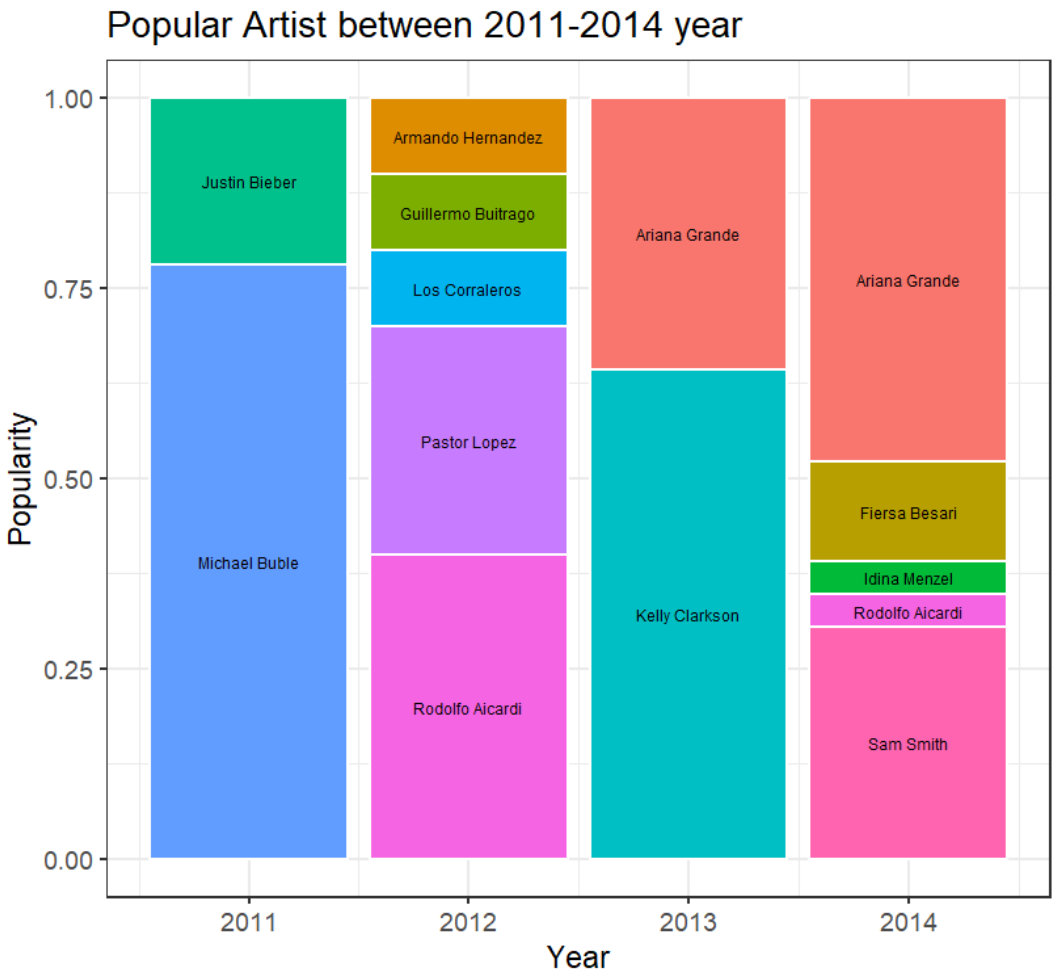
```
> install.packages("forecats")
WARNING: Rtools is required to build R packages but is not currently installed.
 Please download and install the appropriate version of Rtools before proceedin
g:

https://cran.rstudio.com/bin/windows/Rtools/
Installing package into 'C:/Users/rantali/AppData/Local/R/win-library/4.2'
(as 'lib' is unspecified)
Warning in install.packages :
  package 'forecats' is not available for this version of R

A version of this package for your version of R might be available elsewhere,
see the ideas at
https://cran.r-project.org/doc/manuals/r-patched/R-admin.html#Installing-package
s
> library(forcats)
> library(ggplot2)
> ggplot(Top_music, aes(x = forcats::fct_reorder(Title, Dur), y = Dur, fill=Dur)
 ) + geom_col() + coord_flip() + labs(title="Top 10 Longest song track in USA",
 x="Music", y= "Duration") +  geom_text(aes(label = round(Dur, 1)), nudge_y= -5,
 color="white")
```

```
> install.packages("forecats")

> library(forcats)

> library(ggplot2)

> ggplot(Top_music, aes(x = forcats::fct_reorder(Title, Dur), y = Dur, fill=Dur) ) + geom_col()
+ coord_flip() + labs(title="Top 10 Longest song track in USA", x="Music", y= "Duration") +
geom_text(aes(label = round(Dur, 1)), nudge_y= -5, color="white")
```

Description of Column chart:

After running the code for TOP 10 longest durations(second) song for USA, we got the above column chart. “O Tannenbaum” song track has the top duration of 309, then “Last Christmas” with 263 and “All I Want for Christmas Is You” with 241. “Wonderful Christmastime”, “Underneath the Tree” and “This Christmas” has almost same duration of 228, 230 and 231 respectively. Then come the “Happy Xmas” and “Circles” with duration of 214 and 215 respectively. After this we have “It’s Beginning to Look a Lot like Christmas” with 207 and “Santa Baby” having the lowest duration of 206.

3. **Popular artist between year 2011 to 2014.**



Filter data code for top artist

```
> View(ds_music)
> library(dplyr)
> Top_artist<-ds_music %>%
+ group_by(Year) %>%
+ count(Artist) %>%
+ mutate(prop=n/sum(n))%>%
+ filter(Year %in% c(2011,2012,2013,2014))
> View(Top_artist)
```

```
> View(ds_music)
> library(dplyr)
> Top_artist<-ds_music %>%
+ group_by(Year) %>%
+ count(Artist) %>%
+ mutate(prop=n/sum(n))%>%
+ filter(Year %in% c(2011,2012,2013,2014))
> View(Top_artist)
```

The below table shows results of the filter data:

	Year	Artist	n	prop
1	2011	Justin Bieber	9	0.21951220
2	2011	Michael Buble	32	0.78048780
3	2012	Armando Hernandez	1	0.10000000
4	2012	Guillermo Buitrago	1	0.10000000
5	2012	Los Corraleros	1	0.10000000

Code for Visualization (Stacked Chart)

```
> library(ggplot2)
> ggplot(Top_artist, aes(Year, prop, fill = Artist)) + geom_bar(stat='identity',
 color = 'white', show.legend = F) + geom_text(aes(label=paste(Artist)), size=2.
0, color='black', position = position_stack(vjust = .6)) + theme_bw()+ labs(title
='Popular Artist between 2011-2014 year', y='Popularity', x='Year')
```

```
> library(ggplot2)
> ggplot(Top_artist, aes(Year, prop, fill = Artist)) + geom_bar(stat='identity',  color = 'white', show.legend = F) + geom_text(aes(label=paste(Artist)), size=2.0, color='black', position = position_stack(vjust = .6)) + theme_bw() + labs(title='Popular Artist between 2011-2014 year', y='Popularity', x='Year')
```

Description of Stacked bar graph:

The above stacked bar graph shows the artist popularity from year 2011 to 2014. For year 2011, we only have two famous artist, Michael Buble being the most popular artist for the year and then Justin Bieber. For year 2012 we have 5 top artist, Rodolfo Aicardi the most famous of all 5 artist, then come the Pastor Lopez with almost comparable popularity to Rodolfo Aicardi. Then we have the Los Corraleros, Guillermo Buitrago and Armando Hermandez having the similar popularity. In the year of 2013, we have two top artist in which Kelly Clarkson is the most famous and then come the Ariana Grande. For year 2014, Ariana Grande is again among the top artist and the most famous artist of year 2014, then we Sam Smith with popularity score above 0.25. Then we have the Fiersa Besari, Idina Menzel and Rodolfo Aicardi among the top artist of year 2014.

Statistical Summary

Summary of music audio features Energy and Danceability.

```
> View(ds_music)
> summary(ds_music)
    Title              Artist           Top_genre              Year     
 Length:923         Length:923         Length:923         Min.   :1942  
 Class :character   Class :character   Class :character   1st Qu.:2011  
 Mode  :character   Mode  :character   Mode  :character   Median :2019  
                                                          Mean   :2009  
                                                          3rd Qu.:2019  
                                                          Max.   :2019  
      Bpm              Nrgy            Dnce             DB        
 Min.   : 47   Min.   :10.00   Min.   :16.00   Min.   :-23.000  
 1st Qu.: 98   1st Qu.:48.50   1st Qu.:53.00   1st Qu.: -9.000  
 Median :120   Median :65.00   Median :68.00   Median : -6.000  
 Mean   :124   Mean   :61.43   Mean   :64.65   Mean   : -6.888  
 3rd Qu.:147   3rd Qu.:76.00   3rd Qu.:76.00   3rd Qu.: -5.000  
 Max.   :205   Max.   :98.00   Max.   :95.00   Max.   :  0.000  
      Live             Val              Dur             Acous       
 Min.   : 2.00   Min.   : 5.00   Min.   : 85.0   Min.   : 0.00  
 1st Qu.: 9.00   1st Qu.:43.00   1st Qu.:171.0   1st Qu.:10.00  
 Median :13.00   Median :61.00   Median :199.0   Median :29.00  
 Mean   :20.35   Mean   :60.41   Mean   :203.6   Mean   :35.46  
 3rd Qu.:28.00   3rd Qu.:78.00   3rd Qu.:228.0   3rd Qu.:59.50  
 Max.   :97.00   Max.   :98.00   Max.   :464.0   Max.   :99.00  
      Spch          Popularity       Country         
 Min.   : 2.000   Min.   : 37.00   Length:923         
 1st Qu.: 4.000   1st Qu.: 77.00   Class :character   
 Median : 5.000   Median : 85.00   Mode  :character   
 Mean   : 8.815   Mean   : 82.89                      
 3rd Qu.:10.000   3rd Qu.: 91.00                      
 Max.   :56.000   Max.   :100.00                      
```

> View(ds_music)

> summary(ds_music)

To summarize the entirety of the Top50List.csv dataset, we utilized the summary () function in R Studio. It provides the summary values for the columns after summarizing each column. We can see the summary values for min, median, mean and max for all the columns from the top50list.csv dataset.

Summary statistics code for Energy:

```
> summary(ds_music$Nrgy)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  10.00   48.50   65.00   61.43   76.00   98.00
> min(ds_music$Nrgy)
[1] 10
> max(ds_music$Nrgy)
[1] 98
> mean(ds_music$Nrgy)
[1] 61.4312
> median(ds_music$Nrgy)
[1] 65
> sd(ds_music$Nrgy)
[1] 19.33662
> music_energy<-filter(
+ ds_music,
+ Nrgy > 61
+ )
> dim(music_energy)
[1] 532  15
```

> summary(ds_music$Nrgy)

Min. 1st Qu. Median Mean 3rd Qu. Max.

10.00 48.50 65.00 61.43 76.00 98.00

> min(ds_music$Nrgy)

[1] 10

> max(ds_music$Nrgy)

[1] 98

> mean(ds_music$Nrgy)

[1] 61.4312

> median(ds_music$Nrgy)

[1] 65

> sd(ds_music$Nrgy)

[1] 19.33662

```
> music_energy<-filter(
+ ds_music,
+ Nrgy > 61
+ )
> dim(music_energy)
[1] 532  15
```

In this summary analysis, we would like to do the analysis of the energy and the danceability of the song tracks. In the statistical summary of music energy, the median value is 65, maximum values are 98, and the mean is 61.43. The median is the value that divides the dataset in half at the middle. It indicates that the middle number in the column for the percentage of music energy is 65. The mean is the average of value of the dataset. As a result, the average number is 61.43 when we add up all the values in the percent of music energy of the songs column and divide the total value by the sum of all the row counts. We filtered the summary analysis of the music energy which has greater than mean values. At the looking at the result, 532 music tracks have higher energy.

Summary Statistics code for Music danceability:

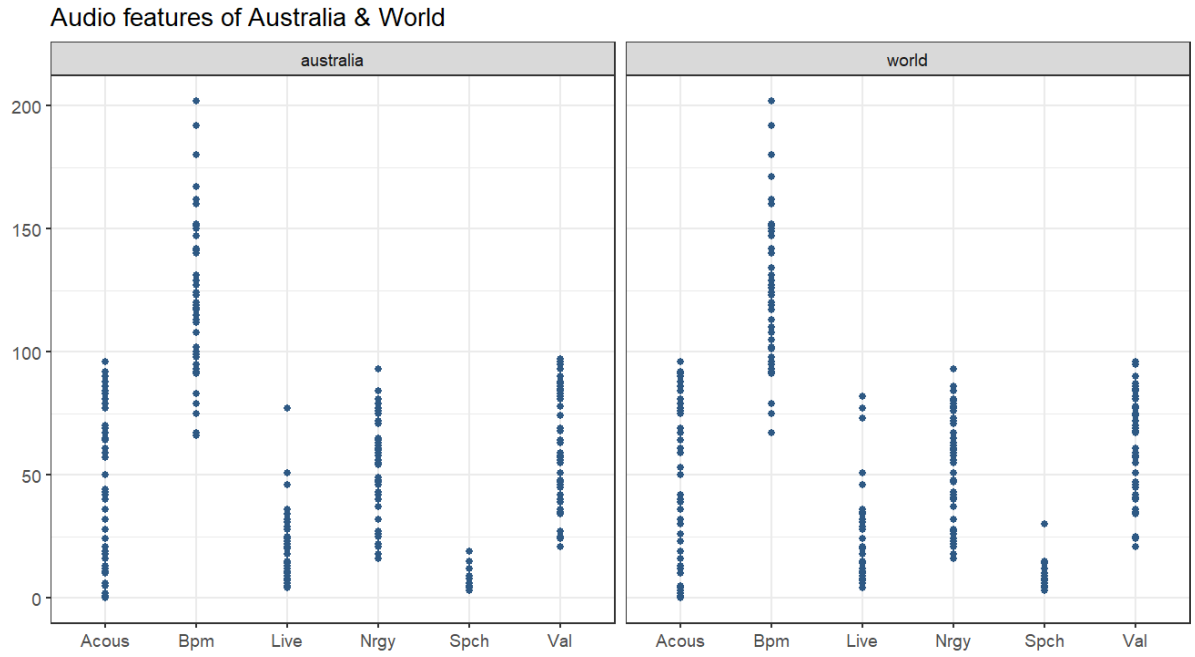
```
> summary(ds_music$Dnce)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  16.00   53.00   68.00   64.65   76.00   95.00
> min(ds_music$Dnce)
[1] 16
> max(ds_music$Dnce)
[1] 95
> mean(ds_music$Dnce)
[1] 64.65005
> median(ds_music$Dnce)
[1] 68
> sd(ds_music$Dnce)
[1] 16.18809
```

```
> summary(ds_music$Dnce)
Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
16.00   53.00   68.00   64.65   76.00   95.00
> min(ds_music$Dnce)
[1] 16
> max(ds_music$Dnce)
[1] 95
> mean(ds_music$Dnce)
[1] 64.65005
> median(ds_music$Dnce)
[1] 68
> sd(ds_music$Dnce)
[1] 16.18809
```

In the statistical summary of audio feature danceability, the median value is 68, maximum values are 95, minimum value are 16, and the mean is 64.65. The median is the value that divides the dataset in half at the middle. Higher the value more suitable the song is for dancing.

User-Defined Function

1. Is there a relationship between individual country's audio features such as Australia and World's audio features?



Code for filter data for Australia and world:

```
> filter_aus_world<-ds_music%>%
+ filter(Country == "australia" | Country == "world")
> View(filter_aus_world)
> dim(filter_aus_world)
[1] 98 15
> MergeAudio_feature <- gather(filter_aus_world, key="Audio_features", value="value",
 c("Bpm", "Live", "Val","Acous","Spch","Nrgy"))
```

> filter_aus_world<-ds_music%>%

+ filter(Country == "australia" | Country == "world")

> View(filter_aus_world)

> dim(filter_aus_world)

> MergeAudio_feature <- gather(filter_aus_world, key="Audio_features", value="value", c("Bpm", "Live", "Val","Acous","Spch","Nrgy"))

Result of the filter data:

Popularity	Country	Audio_features	value
78	australia	Bpm	66
92	world	Bpm	202
92	australia	Bpm	202
80	australia	Bpm	83
87	world	Bpm	131
87	australia	Bpm	131

User defined Function

"Audio_feature" function was used to create the point chart for audio features.

```
library(ggplot2)
library(dplyr)
library(tidyr)
library(scales)

Audio_feature <- function(audiodata, xdata, ydata, datacolor, contrydata, datatitle)
{
    ggplot(audiodata, aes(x=xdata, y=ydata)) +
    geom_point(col=datacolor) + facet_wrap(contrydata) +
    labs(title = datatitle, x="", y="") + theme_bw()
}
```


```
library(ggplot2)
library(dplyr)
library(tidyr)
library(scales)

Audio_feature <- function(audiodata, xdata, ydata, datacolor, contrydata, datatitle)
{
   ggplot(audiodata, aes(x=xdata, y=ydata)) +
   geom_point(col=datacolor) + facet_wrap(contrydata) +
   labs(title = datatitle, x="", y="") + theme_bw()
}
```


Set the working directory to run the function. Function "source" is used to load file.

```
> setwd("C:/Users/rantali/Desktop/College/5250/R Project/Dataset/R Script")
> source("Audio_features.R")
```

“Audio_feature” used to call the function.

```
> Audio_feature(MergeAudio_feature, MergeAudio_feature$Audio_features, Mer
geAudio_feature$value,"#2E5984",~MergeAudio_feature$Country,"Audio feature
s of Australia & World")
```

> setwd("C:/Users/rantali/Desktop/College/5250/R Project/Dataset/R Script")
>
> source("Audio_features.R")
>
> Audio_feature(MergeAudio_feature, MergeAudio_feature$Audio_features, MergeAudio_feature$value,"#2E5984",~MergeAudio_feature$Country,"Audio features of Australia & World")

Audio feature Mean Value for Australia and world

```
> install.packages("plyr")
WARNING: Rtools is required to build R packages but is not currently installed. Please download
 and install the appropriate version of Rtools before proceeding:

https://cran.rstudio.com/bin/windows/Rtools/
Installing package into 'C:/Users/rantali/AppData/Local/R/win-library/4.2'
(as 'lib' is unspecified)
trying URL 'https://cran.rstudio.com/bin/windows/contrib/4.2/plyr_1.8.8.zip'
Content type 'application/zip' length 1155082 bytes (1.1 MB)
downloaded 1.1 MB

package 'plyr' successfully unpacked and MD5 sums checked

The downloaded binary packages are in
        C:\Users\rantali\AppData\Local\Temp\RtmpgB5c1T\downloaded_packages
> library(plyr)
------------------------------------------------------------------------------
You have loaded plyr after dplyr - this is likely to cause problems.
If you need functions from both plyr and dplyr, please load plyr first, then dplyr:
library(plyr); library(dplyr)
------------------------------------------------------------------------------

Attaching package: 'plyr'
```

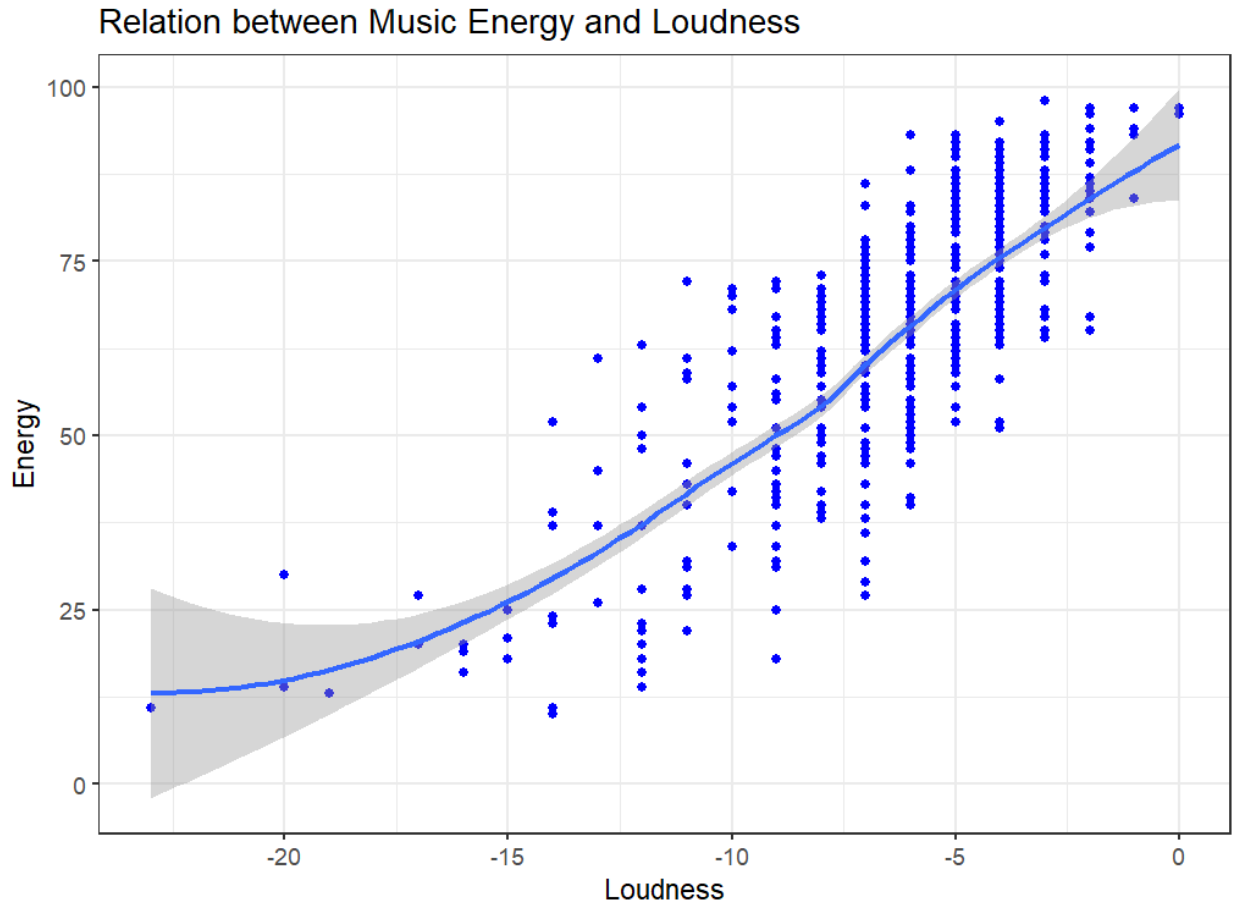
```
The following objects are masked from 'package:dplyr':

    arrange, count, desc, failwith, id, mutate, rename, summarise, summarize

Warning message:
package 'plyr' was built under R version 4.2.2
> group.means<-ddply(MergeAudio_feature,c("Country"),summarise,mean=mean(value))
> group.means
    Country     mean
1 australia 51.61224
2     world 52.71769
```

```
> install.packages("plyr")
> library(plyr)
> group.means<-ddply(MergeAudio_feature,c("Country"),summarise,mean=mean(value))
> group.means
   Country     mean
1 australia 51.61224
2     world 52.71769
```

Description of visualization

The above point chart shows the relation between individual country's audio features such as Australia and World's audio features, we can see that it's almost the same from the graph. Although there is difference between Australia and World Spotify tracks such as Energy, Beats Per Minute and Valance, we can observe from the shape for both point graph it's similar. And, looking at summary we can see mean value of audio feature for Australia and World are approximately similar so, overall difference in the mean values of audio features between is only **1.11**. Therefore, there a correlated relationship between individual country's audio features such as Australia and World's audio features.

2. Relation between energy and danceability



Code for Visualization (Point Chart)

Function Dur_scatterplot used to create chart:

```
library(ggplot2)
library(dplyr)

Dur_scatterplot <- function(data, dataxcoln, dataycoln, fillcolor, datatitle, dataxname, datayname)
{
    ggplot(data, aes(x=dataxcoln, y=dataycoln)) +
    geom_point(color= fillcolor) + geom_smooth(stat = "smooth") +
    labs(title = datatitle, x = dataxname, y = datayname) +
    theme_bw()
}
```

```
library(ggplot2)
library(dplyr)

Dur_scatterplot <- function(data, dataxcoln, dataycoln, fillcolor, datatitle, dataxname,
datayname)
{
   ggplot(data, aes(x=dataxcoln, y=dataycoln)) +
   geom_point(color= fillcolor) + geom_smooth(stat = "smooth") +
   labs(title = datatitle, x = dataxname, y = datayname) +
   theme_bw()
}
```

Called function “Dur_scatterplot”:

```
> Dur_scatterplot(ds_music,ds_music$DB,ds_music$Nrgy,"Blue","Rela
tion between Music Energy and Loudness", "Loudness", "Energy")
```

```
> Dur_scatterplot(ds_music,ds_music$DB,ds_music$Nrgy,"Blue","Relation between Music
Energy and Loudness", "Loudness", "Energy")
```

Description For Chart:

The above graph describes relation between energy and loudness. The loudness is measure in decibel. The graph shows when loudness is increasing the energy of the song is increasing. Which means positive relation between energy and loudness.

References

Werner, A. (2020). Organizing music, organizing gender: Algorithmic culture and Spotify recommendations. Retrieved November 13, 2022, from https://www.tandfonline.com/doi/pdf/10.1080/15405702.2020.1715980

Anderson, A (2020, April 01). Algorithmic effects on the diversity of consumption on Spotify: Proceedings of the web conference 2020. Retrieved November 17, 2022, from https://dl.acm.org/doi/abs/10.1145/3366423.3380281

Collins, B. (2022, November 10). *How Spotify Stayed no. 1 in streaming audio even with Apple, YouTube and Amazon aiming for it*. CNBC. Retrieved December 14, 2022, from https://www.cnbc.com/2022/11/10/how-spotify-stayed-no-1-in-streaming-music-vs-apple-youtube-amazon.html

STDHA. (n.d.). *Ggplot2 Pie chart : Quick Start Guide - R Software and Data Visualization*. STHDA. Retrieved December 14, 2022, from http://www.sthda.com/english/wiki/ggplot2-pie-chart-quick-start-guide-r-software-and-data-visualization