Final Project: Exploratory Analysis of Adele. Can We Teach The Computer To Recognize The Artist?

Team E

2022-05-20

1. Introduction

You are sitting in your car and a new song comes up. You've never heard it before but you immediately recognize it belongs to your favorite artist. How did you recognize it? Was it the voice? The beat? The lyrics? Was it similar to other work by the same artist? in this project we will attempt to build a model that can recognize the artist by past songs.

What do we want to do? The focus of this project is on specific artists, stemming from the belief that a lot of the value of the song is reactionary to the artist and his work. The goal is to see if we can build a ML algorithm that can identify a new song from a given artist based on other works by the artist . Some of the inspiration for this project is from Lerdahl and Jackendoff 's 'Generative Theory' (see appendix).

In order to achieve this feat we will focus on a specific artist, Adele. We will analyze her music in depth and look for interesting patterns. We will then compare the artist to other artists and see if we can "profile" her music and work. Along the way, we hope to learn and see if we come to any interesting conclusions.

We chose to research Adele's music and the content of her songs with the understanding that her works have managed to make a global impact and have greatly influenced the music industry and people from all over the world. We can learn about the connection of her personal life to the way she wrote and the characteristics of the albums she released over the years.

2. Data (see README.md for further refrences)

1. Data from Spotify: In order to use the Spotify API we must first create a developer account with Spotify and get our CLIENT_ID and SECRET, we can then create a token:

```
Sys.setenv(SPOTIFY_CLIENT_ID = 'b9c8fd4ce6074182b41c18e1cb1a6fa5')
Sys.setenv(SPOTIFY_CLIENT_SECRET = '9dcdb05b12e443b99133067a943fd49b')
access_token <- get_spotify_access_token()
```

we can now connect and use the Spotify API

```
adele <- get_artist_audio_features('adele')
save(adele, file = "../data/adele.Rdata") # saved to data folder
```

2. Scraping lyrics off azlyrics.com We scraped the lyrics for adeles first three albums off azlyrics.com. first we created a function that scrapes and cleans (based on the format of the site) the lyrics:

```
lyric_scraper <- function(url) {
  m <- read_lines(url)
  giveaway <- "Sorry about that."
  start <- grep(giveaway, m) + 1
  end <- grep("</div>", m[start:length(m)])[1] + start
  lyrics <- paste(gsub("<br>|</div>", "", m[start:end]), collapse = " ")
  return(lyrics)
}
```

next we will run through all here songs and scrape them of azlyrics:

```
titles <- adele_data["track_title"]
datalist = list()

for (i in 1:35) {
  song <- titles[i,]
  song_clean  <- tolower(gsub("[[:punct:][:blank:]]","",song))
  url <- paste("http://www.azlyrics.com/lyrics/adele/", song_clean, ".html", sep = "")
  dat <- lyric_scraper(url)
  datalist[[i]] <- dat
  Sys.sleep(5)
}
```

We will now save it to the data file.

```
adele_data$lyrics <- datalist
save(adele_data, file = "../data/adele_data.Rdata")
```

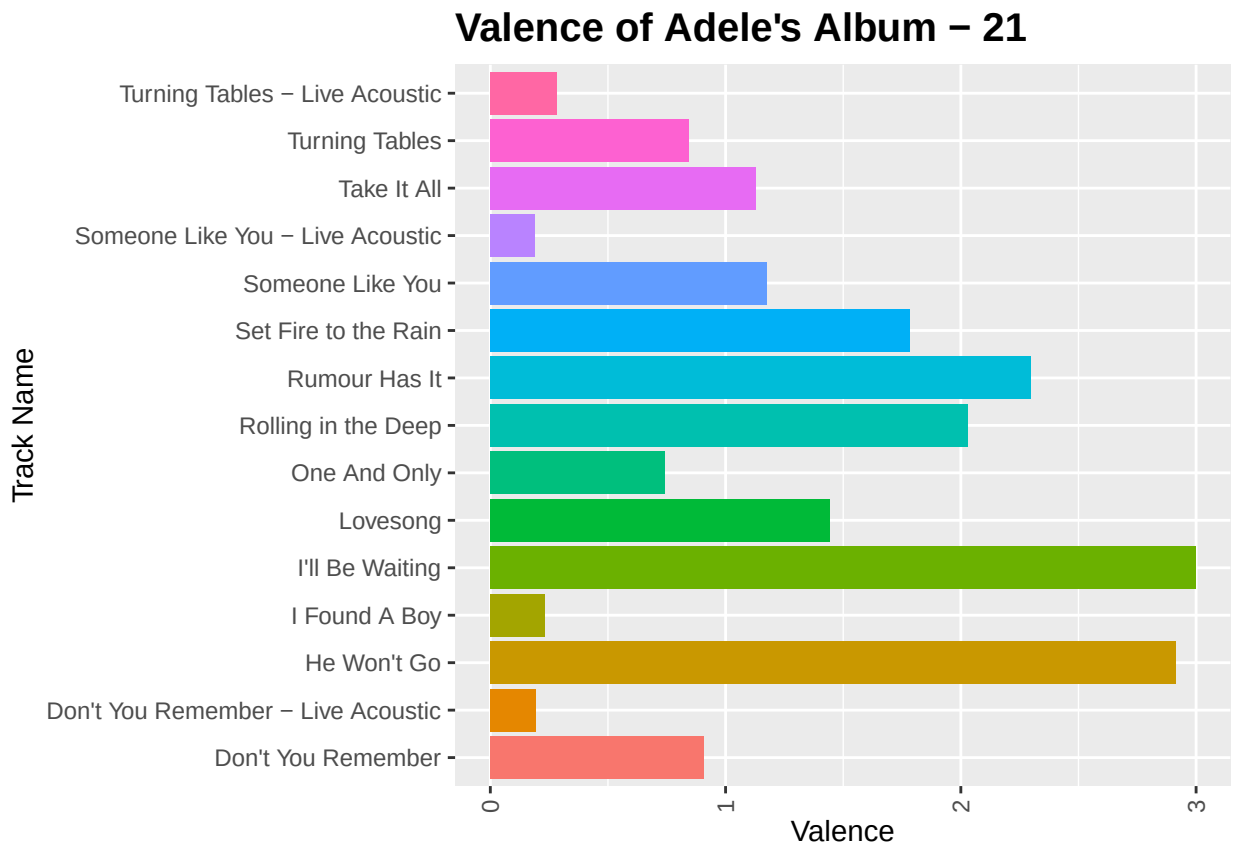
we will use both the metrical data pulled from Spotify and the Lyrical data pulled from azlyrics in our research.

3. Preliminary results

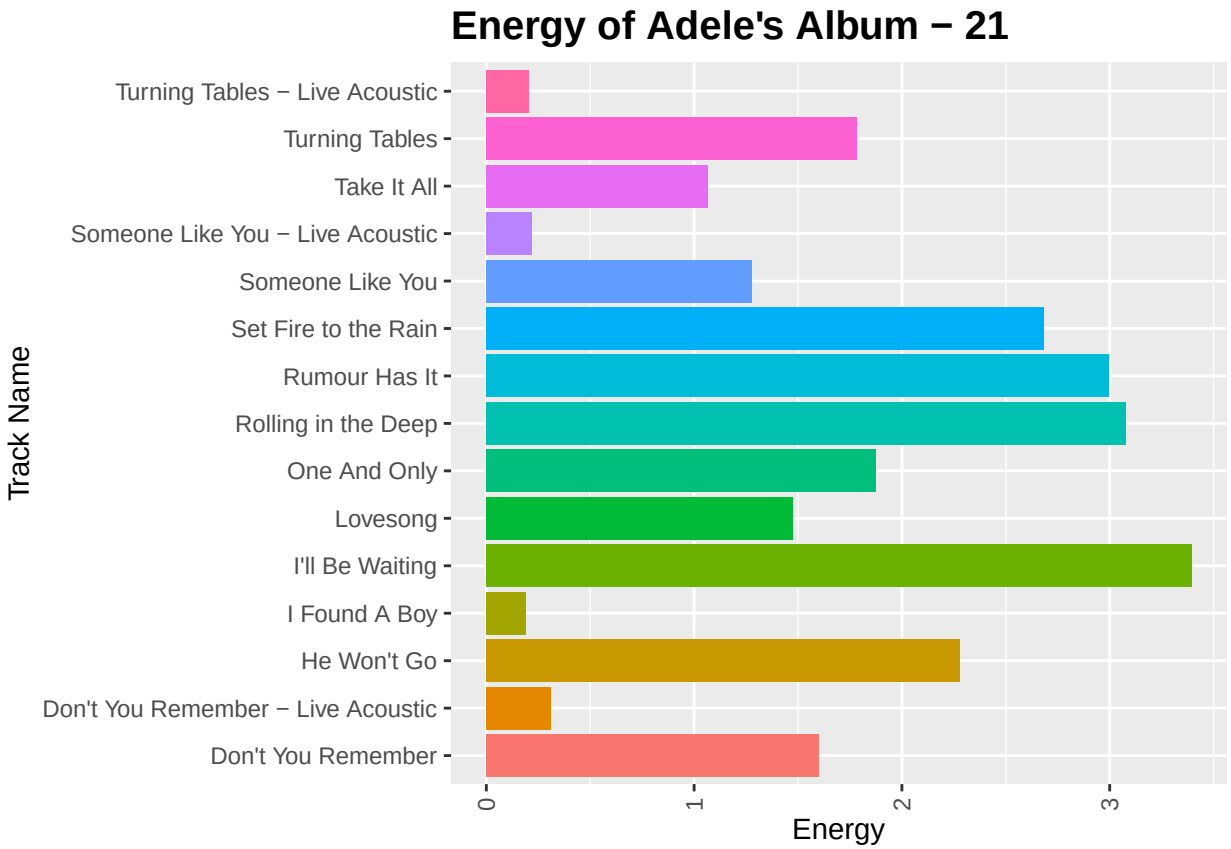
we will start by doing a basic analysis of the 21 album.

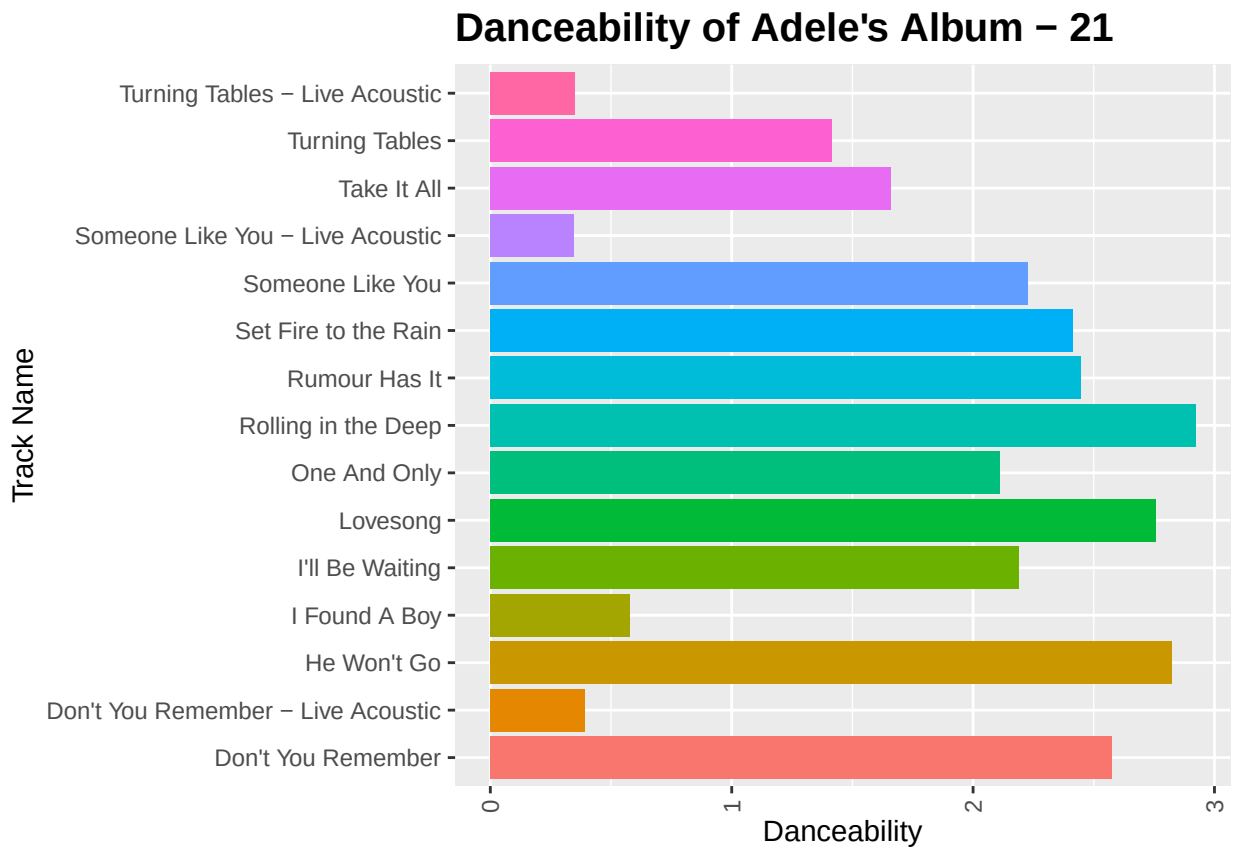
```
## [1] "|track_name                         | danceability| energy| valence|   tempo|"
## [2] "|:-----------------------------------|------------:|------:|-------:|-------:|"
## [3] "|Rolling in the Deep                |        0.730|  0.769|   0.507| 104.948|"
## [4] "|Rumour Has It                      |        0.612|  0.749|   0.574| 120.052|"
## [5] "|Turning Tables                     |        0.353|  0.446|   0.211| 155.476|"
## [6] "|Don't You Remember                 |        0.644|  0.400|   0.227| 115.025|"
```

Now lets make a simple graphs to get a first impression of the data.

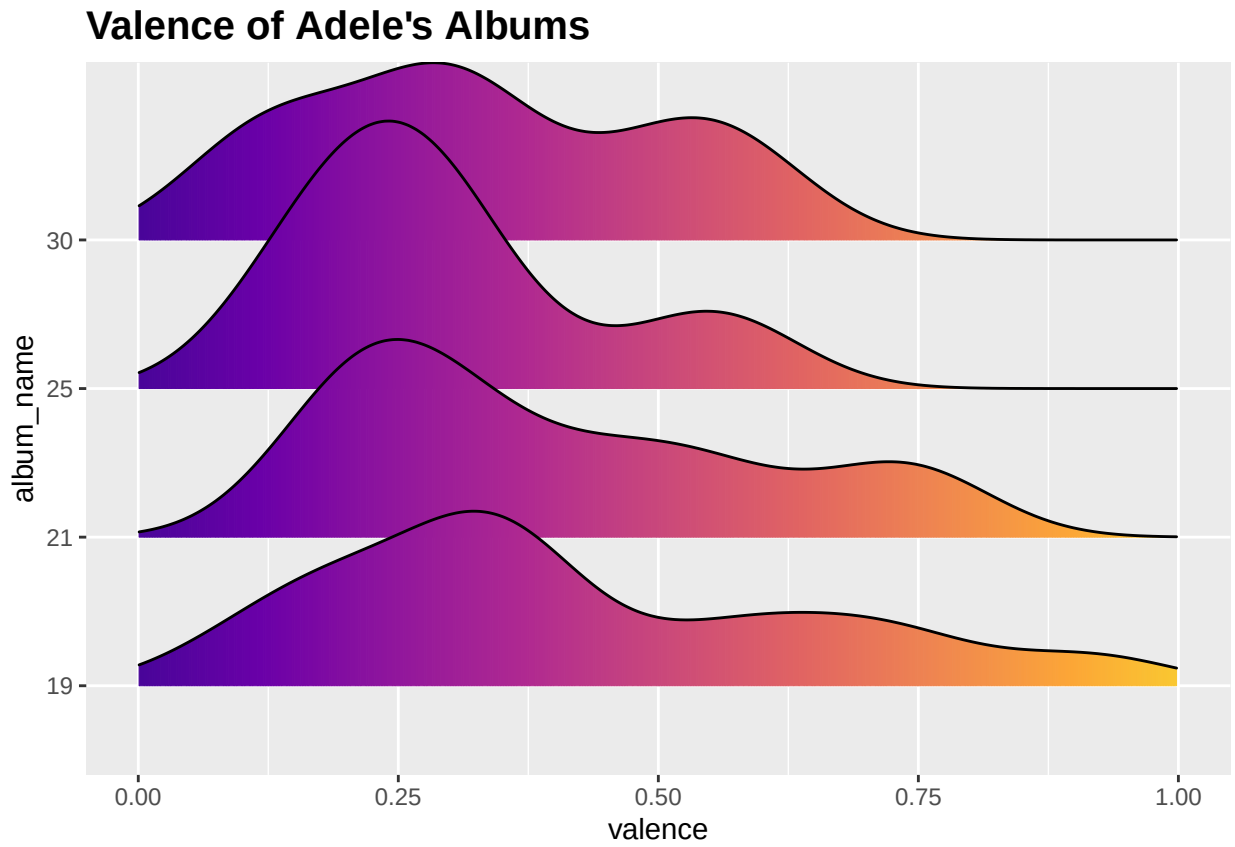


Valence varies across the album, acoustic songs have a lower rating as expected.

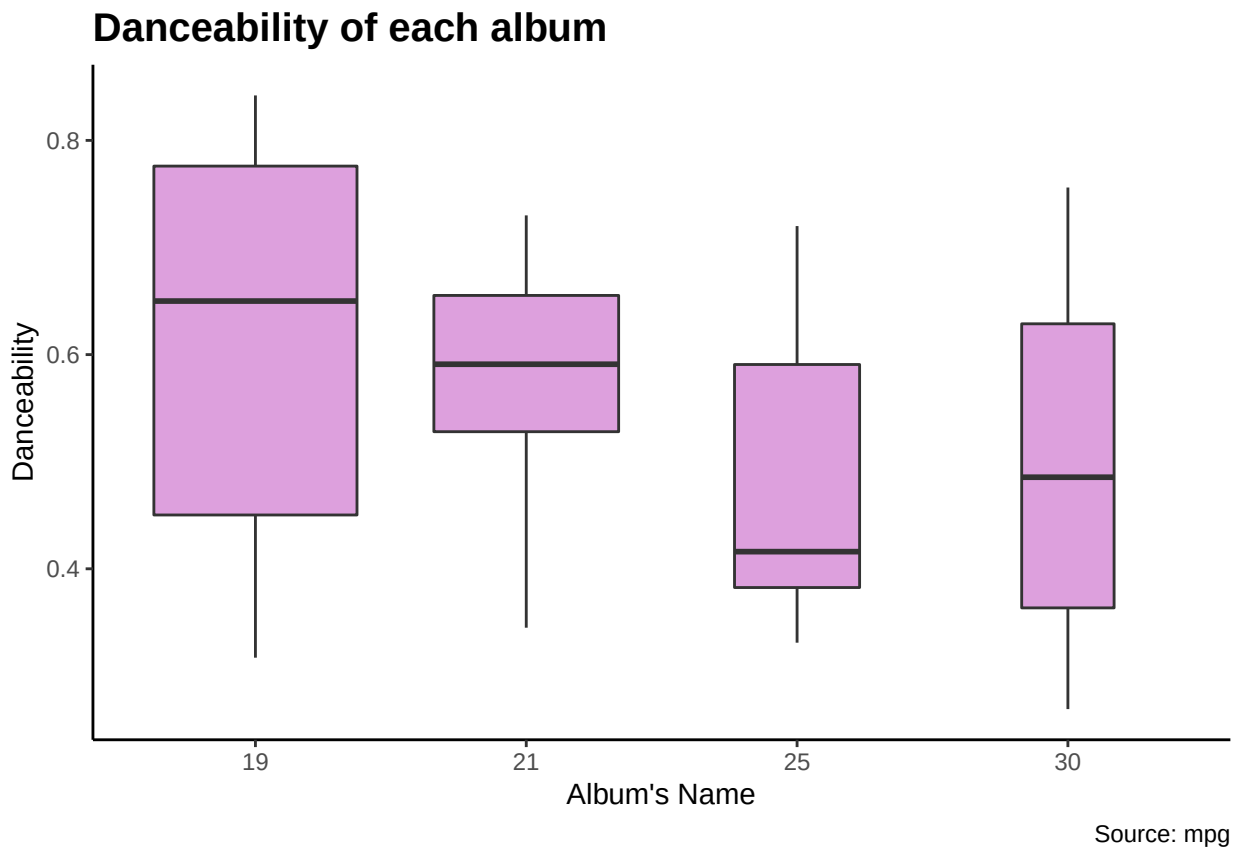

Energy of Adele's Album − 21
Turning Tables − Live Acoustic
Turning Tables
Take It All
Someone Like You − Live Acoustic
Someone Like You
Set Fire to the Rain
Rumour Has It
Rolling in the Deep
One And Only
Lovesong
I'll Be Waiting
I Found A Boy
He Won't Go
Don't You Remember − Live Acoustic
Don't You Remember
Track Name
0
1
2
3
Energy



At first glance we can see there seems to be some correlation between the different metrics for the same song. Non of the metrics are consistent across the album.



The albums seem to Follow a certain pattern. her older albums seem to contain a wider span of emotion and as the albums advance the valence focuses more and more. most of here music seems to be concentrated at about 0.3 valence.

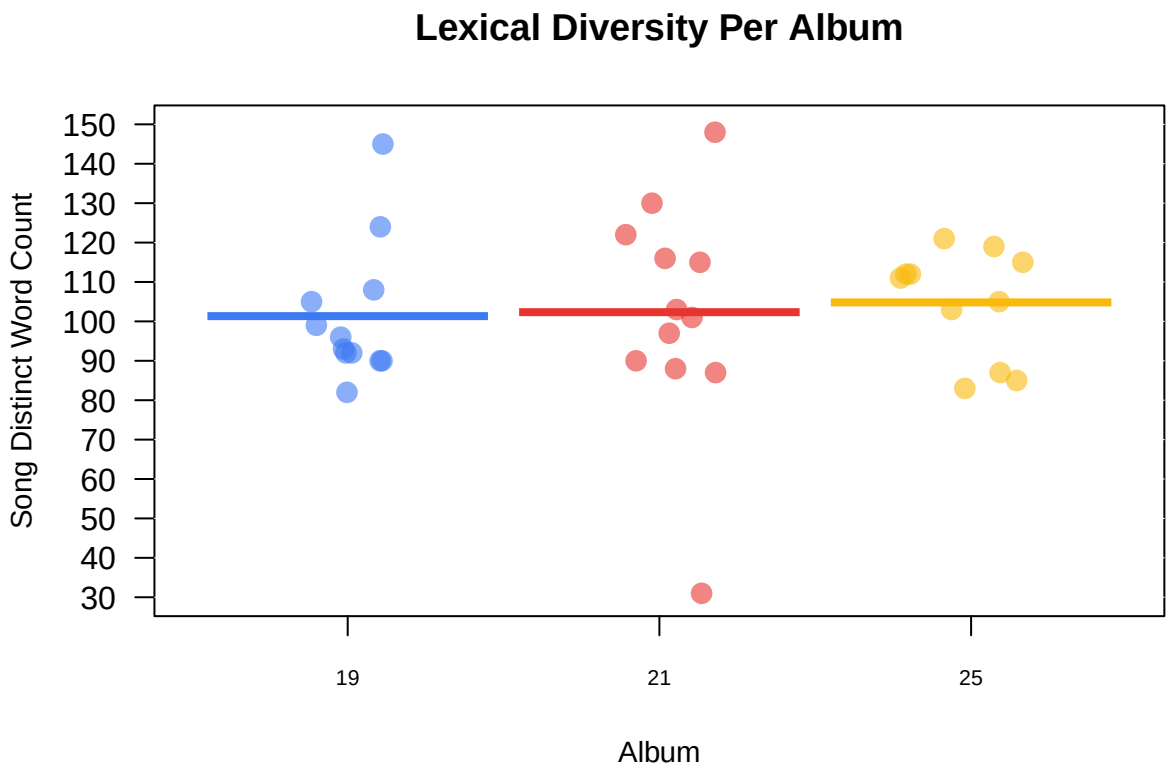


Now that we've played around with the data a bit lets look at the Lyrics. We will start by tidying up the data and creating 1 big data base of all the words.

```
##     track_n       track_title           album              word
##  Min.   : 1.000   Length:2672        Length:2672        Length:2672
##  1st Qu.: 3.000   Class :character   Class :character   Class :character
##  Median : 6.000   Mode  :character   Mode  :character   Mode  :character
##  Mean   : 5.978
##  3rd Qu.: 9.000
##  Max.   :12.000
```

Now lets play with it a bit:

Can we find any interesting patterns? lets compare the lexical diversity between albums:



Preliminary research summary: We were able to collect data relevant to our research on Adele's lyrics and many characteristics of her songs and albums. From an initial analysis of the data we produced models for the analysis of the data in order to understand what we can do with it. We were able to understand that Adele's works have salient features that will help us try to create a machine learning algorithm for identifying a song that belongs to her.

4. Data analysis plan

Goal: Does song Y belong to Adele based on the lyrics, valence, cords and other metrics we will decide are significant during our research.

step 1 - preliminary research on Adele: 1. getting the data (harder than it looks!) 2. Summary of Adele 3. Basic analyzing of each album based on lyrics, valence, energy and a few other metrics 4. comparing the albums

step 2 - Comparison to Similar artists: In order to be able to create a unique identity for Adele we should consider similar artists. to do this we will: 1. map data on similar artists
2. compare metrics with Adele 3. see if we can identify the artists based on data (with out machine learning at this stage)

step 3 - Can we predict if a song belongs to Adele? Regression model: 1. build a model based on our knowledge of Adele and our prior research. 2. decision variables, weights, etc.

step 4 - Can we predict if a song belongs to Adele? machine learning: 1. using machine learning tools, try and build a classification model. 2. train - test using different values.

step 5 - Comparison of our different models 1. Our regression model vs The Machine.

step 6 - Summary 1. What have we learnt? 2. Can we scale the model? 3. What can we use it for?

Teamwork: Nitzan - Data querying from Spotify, Data Visualization and analysis.

Eden - Project design and logic, data analysis, Field expert (big fan of Adele <3)

Yonatan - Data scraping from azlyrics.com, Machine learning modeling.

Appendix

Data README

music and lyrics copyright of Adele. data in Data file.

Source code

```
options(tinytex.verbose = TRUE)
library(webshot)
library(tinytex)
library(knitr) # knit to PDF
library(spotifyr)# for pulling track audio features and info from Spotify's Web API.
library(vembedr)


# graphical and visualization libraries:
library(tidyverse)
library(wordcloud2) # fast visualization tool for creating word cloud
library(ggplot2)# creating graphics
library(ggridges) # for visualizing changes
library(tidytext)
library(dplyr)
library(yarrr) # visualizing features
library(gridExtra)
library(reshape2)
Sys.setenv(SPOTIFY_CLIENT_ID = 'b9c8fd4ce6074182b41c18e1cb1a6fa5')
Sys.setenv(SPOTIFY_CLIENT_SECRET = '9dcdb05b12e443b99133067a943fd49b')
access_token <- get_spotify_access_token()
adele <- get_artist_audio_features('adele')
save(adele, file = "../data/adele.Rdata") # saved to data folder
lyric_scraper <- function(url) {
  m <- read_lines(url)
  giveaway <- "Sorry about that."
  start <- grep(giveaway, m) + 1
  end <- grep("</div>", m[start:length(m)])[1] + start
  lyrics <- paste(gsub("<br>|</div>", "", m[start:end]), collapse = " ")
  return(lyrics)
}
titles <- adele_data["track_title"]
datalist = list()

for (i in 1:35) {
  song <- titles[i,]
  song_clean  <- tolower(gsub("[[:punct:][:blank:]]","",song))
  url <- paste("http://www.azlyrics.com/lyrics/adele/", song_clean, ".html", sep = "")
```

```
  dat <- lyric_scraper(url)
  datalist[[i]] <- dat
  Sys.sleep(5)
}
adele_data$lyrics <- datalist
save(adele_data, file = "../data/adele_data.Rdata")
load(file = "../data/adele_data.Rdata")
load(file = "../data/adele.Rdata")
adele %>%
  filter(album_name %in% "21")-> ad21

a_21 <- ad21 %>%
  select(track_name, danceability, energy, valence, tempo) %>%
  kable()
head(a_21)
ad21 %>%
  arrange(desc(valence))%>%

  ggplot(aes(track_name, valence, fill= track_name)) + geom_col() +
  theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust=1),legend.position = "none") +
  coord_flip()+
  ggtitle("Valence of Adele's Album - 21 ")+theme(plot.title = element_text(size = 15, face = "bold"))
 labs(x = "Track Name", y = "Valence")
ad21%>%
  arrange(desc(energy))%>%

  ggplot(aes(track_name,energy, fill= track_name)) + geom_col() +
  theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust=1),legend.position = "none") +
  coord_flip()+
  ggtitle("Energy of Adele's Album - 21 ") +theme(plot.title = element_text(size = 15, face = "bold"))
 labs(x = "Track Name", y = "Energy")
ad21 %>%
  arrange(desc(danceability))%>%

  ggplot(aes(track_name, danceability, fill= track_name)) + geom_col() +
  theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust=1),legend.position = "none") +
  coord_flip()+
  ggtitle("Danceability of Adele's Album - 21 ") +theme(plot.title = element_text(size = 15, face = "bo
 labs(x = "Track Name", y = "Danceability")

adele %>%
group_by(album_name)  -> Adele_albums

Adele_albums %>%
  ggplot( aes(valence, album_name, fill = ..x..)) +
  geom_density_ridges_gradient() +
  #theme_fivethirtyeight() +
  xlim(0,1) +
  theme(legend.position = "none") +   scale_fill_viridis_c(name = "Temp. [F]", option = "C") +
  ggtitle("Valence of Adele's Albums")+theme(plot.title = element_text(color = "black", size = 15, face
Adele_albums %>%  ggplot(aes( album_name, danceability)) + geom_boxplot(varwidth=TRUE, fill="plum") +
    labs(title="Danceability of each album",
         caption="Source: mpg",
```

```
         x="Album's Name",
         y="Danceability"
        )+ theme(panel.grid.major = element_blank(), panel.grid.minor = element_blank(),
panel.background = element_blank(), axis.line = element_line(colour = "black")) +theme(plot.title = ele
adele_words <- adele_data%>%
  #word is the new column, lyric the column to retrieve the information from
  unnest_tokens(word, lyrics)

adele_tidy <- adele_words %>%
  filter(!nchar(word) < 3) %>%
  anti_join(stop_words)

summary(adele_tidy)

adele_words_counts <- adele_tidy %>%
  count(word, sort = TRUE) %>%
  anti_join(stop_words) %>%
  filter(!word %in% c( "quot", "lea", "yea")) %>%
  wordcloud2( size = 0.7, shape = 'pentagon')
adele_words_counts
word_summary <- adele_words %>%
  group_by(album, track_title) %>%
  mutate(word_count = n_distinct(word)) %>%
  select(track_title, word_count) %>%
  distinct() %>% #To obtain one record per song
  ungroup()

pirateplot(formula = word_count ~ album,
             data = word_summary, #Data frame
   xlab = "Album", ylab = "Song Distinct Word Count", #Axis labels
   main = "Lexical Diversity Per Album", #Plot title
   pal = "google", #Color scheme
   point.o = .6, #Points
   avg.line.o = 1, #Turn on the Average/Mean line
   theme = 0, #Theme
   point.pch = 16, #Point `pch` type
   point.cex = 1.5, #Point size
   jitter.val = .1, #Turn on jitter to see the songs better
   cex.lab = .9, cex.names = .7) #Axis label size
```