Taylor Style

20nguyen.hk

2024-04-06

Question 2

Introduction

The question seeks to investigate the changes of Taylor Swift's musical style throughout her career. Swift's debut album in 2006 marked the beginning of a prolific career, with her ability to consistently produce hit songs keeping her in the spotlight until now. As she has matured both personally and professionally, it's intriguing to explore how, if at all, her music has transformed alongside her and remained successful for almost 20 years. By examining and comparing audio features across her albums, we can identify potential trends and changes in her music. Particularly, we have identified valence, mode, and acousticness to show considerable discrepancies that reveal interesting insights into the change in her music style.

Approach

Our approach includes the use of three types of graph: (1) a relative frequency bar plot overlaid by (2) line chart, (3) a lollipop plot. These plots are suitable for highlighting differences in selected audio feature values across albums. Comparing between songs will be too fine-grained, so we compare across albums by aggregating individual song scores through counts or averages.

The relative frequency bar plot is chosen to compare the distribution of songs in minor versus major modes for each album. This plot type excels at showcasing proportional differences within categories, making it the most suitable choice for visualizing the balance between minor and major modes across albums. The valence, ranged from 0 to 1, is converted to a percentage to align with the relative frequency plot's scale. Line graph is the best choice to show average valence because it is very simple (not overcrowding the relative frequency bar plot), yet effective at depicting trend over time.

For the second plot, a lollipop chart is utilized to compare average acousticness scores across albums. This chart type is particularly suited for highlighting variations in a single metric over different categories—in this case, albums. By plotting average acousticness as 'lollipops,' we can easily spot deviations and trends, and including the average trend line further serves to segment Swift's albums into periods of higher and lower acousticness.

Code

```
library(tidyverse)
```

```
## ── Attaching core tidyverse packages ──────────────────────── tidyverse 2.0.0 ──
## ✔ dplyr     1.1.4     ✔ readr     2.1.5
## ✔ forcats   1.0.0     ✔ stringr   1.5.1
## ✔ ggplot2   3.5.0     ✔ tibble    3.2.1
## ✔ lubridate 1.9.3     ✔ tidyr     1.3.1
## ✔ purrr     1.0.2
## ── Conflicts ────────────────────────────────────────── tidyverse_conflicts() ──
## ✖ dplyr::filter() masks stats::filter()
## ✖ dplyr::lag()    masks stats::lag()
## ℹ Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflic
ts to become errors
```

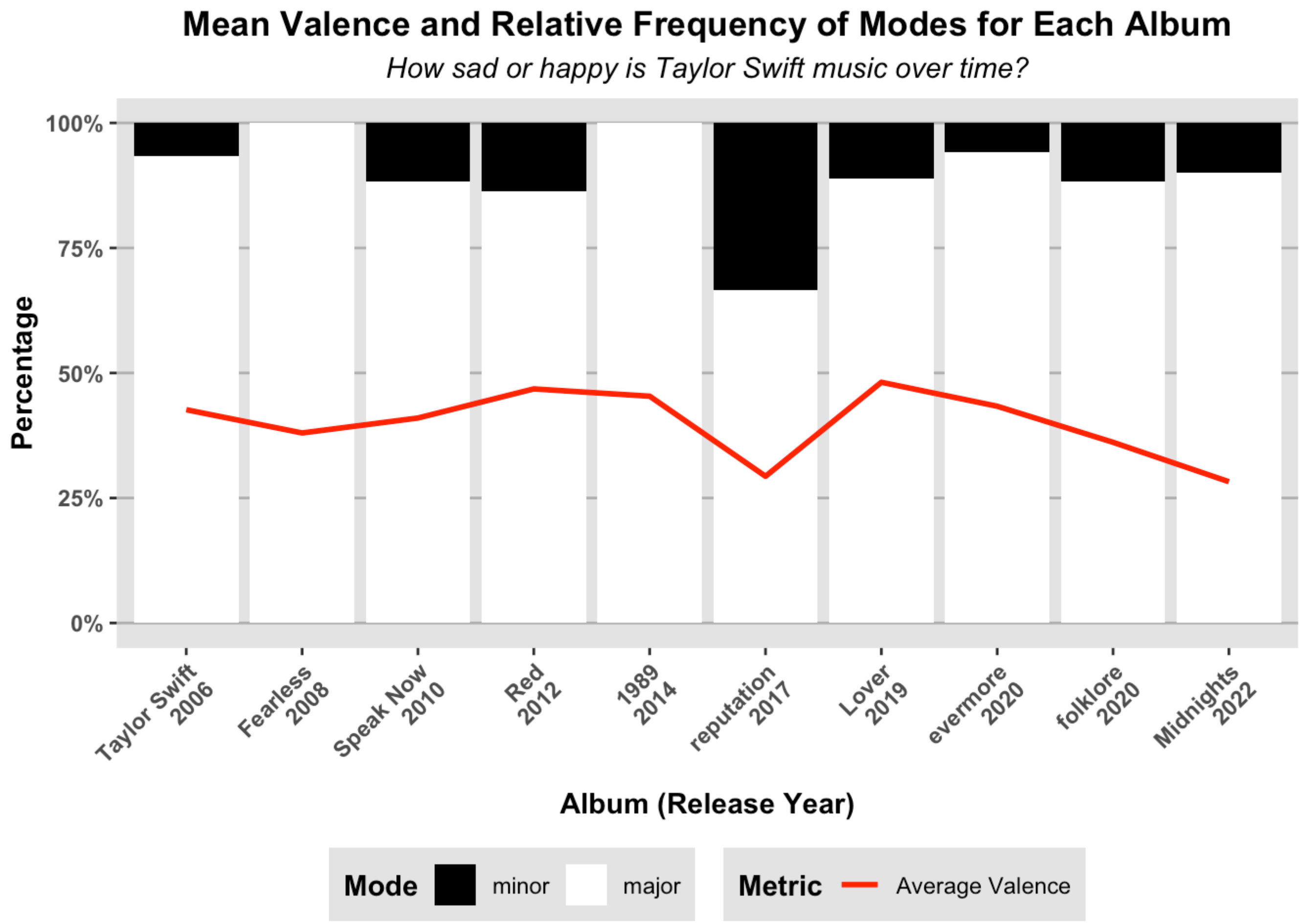
```r
library(dplyr)
library(ggplot2)
all_songs = read.csv("taylor_all_songs.csv")
album_df = read.csv("taylor_albums.csv")
#remove Extended Play (ep) and re-recorded albums (Taylor's Version). We will analyze the remaining 10 main albums.
albums_filtered <- filter(album_df, ep==FALSE & !endsWith(album_name, "(Taylor's Version)"))
#filter songs not in albums_filtered
all_songs_filtered <- all_songs |>
  filter(album_name %in% albums_filtered$album_name)
# Filter out NA entries in mode_name and add a column for the album's release year
all_songs_filtered <- all_songs_filtered |>
  filter(!is.na(mode_name)) |>
  mutate(album_release_year = as.numeric(format(as.Date(album_release), "%Y")))

#theme for 1st chart
theme_piano <- function() {
  theme(
    plot.title = element_text(hjust = 0.5, face = "bold"),
    plot.subtitle = element_text(hjust = 0.5, face = "italic"),
    panel.grid.major.y = element_line(color = "grey70"),
    panel.grid.minor.y = element_blank(),
    panel.grid.major.x = element_blank(),
    panel.grid.minor.x = element_blank(),
    legend.position = "bottom",
    legend.background = element_rect(fill = "grey90"),
    legend.title = element_text(face = "bold"),
    axis.text = element_text(face = "bold"),
    axis.title = element_text(face = "bold"),
    axis.text.x = element_text(angle = 45, hjust = 1),
    panel.background = element_rect(fill="grey90")
  )
}

#theme for 2nd chart
theme_clean <- function() {
  theme(
    plot.title = element_text(hjust = 0.5, face = "bold"),
    panel.grid.major.y = element_line(color = "grey60",linewidth = 0.2, linetype = "dotted"),
    panel.grid.minor.y = element_blank(),
    panel.grid.major.x = element_blank(),
    panel.grid.minor.x = element_blank(),
    legend.position = "bottom",
    axis.text = element_text(face = "bold"),
    axis.title = element_text(face = "bold"),
    axis.text.x = element_text(angle = 45, hjust = 1),
    panel.background = element_rect(fill="grey90")
  )
}
```

```r
# Group by album, calculate total number of songs and count of each mode
album_modes_rf <- all_songs_filtered |>
  # total number of songs in album
  group_by(album_name, album_release) |>
  mutate(total_count = n()) |>
  # mode counts
  group_by(album_name, album_release, mode_name, total_count) |>
  summarise(mode_count = n()) |>
  # relative freq of mode
  mutate(relative_freq = mode_count / total_count) |>
  ungroup()
```

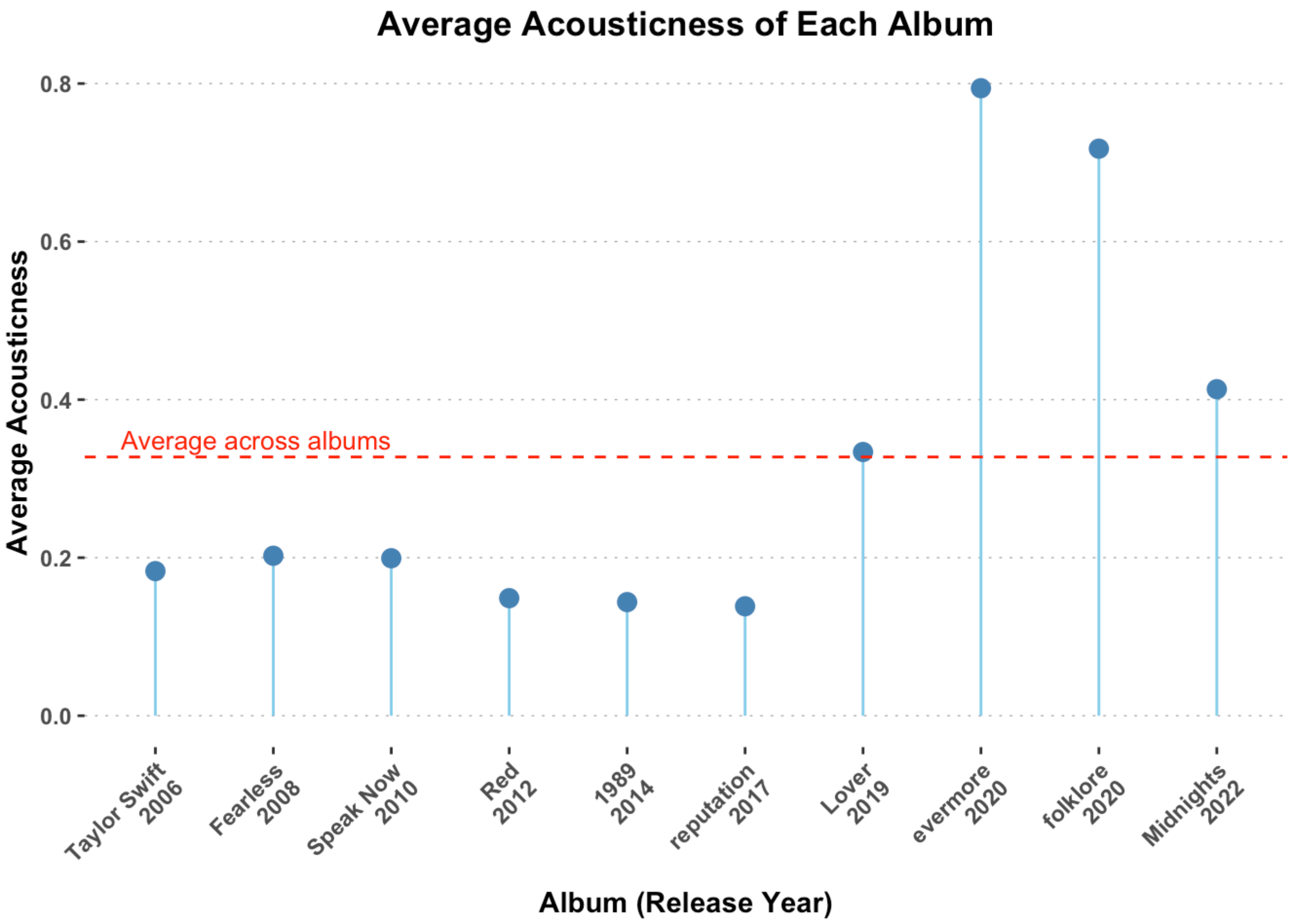
```
## `summarise()` has grouped output by 'album_name', 'album_release', 'mode_name'.
## You can override using the `.groups` argument.
```

```r
#major below, minor above
album_modes_rf$mode_name <- factor(album_modes_rf$mode_name, levels = c("minor", "maj
or"))
# Calculate mean valence for each album
mean_valence_by_album <- all_songs_filtered |>
  group_by(album_name, album_release) |>
  summarise(mean_valence = mean(valence, na.rm = TRUE), .groups = 'drop')

# Join the mean valence data back with the mode frequency data
album_data_combined <- album_modes_rf |>
  left_join(mean_valence_by_album, by = c("album_name", "album_release")) |>
  arrange(album_release) |>
  mutate(album_release_year = as.numeric(format(as.Date(album_release), "%Y")))

# draw
ggplot(album_data_combined, aes(x=reorder(album_name, album_release_year))) +
  geom_bar(aes(fill=mode_name, y=relative_freq), stat="identity", position="fill") +
  geom_line(aes(y=mean_valence, group=1, color="Average Valence"), size=1) +
  # geom_point(aes(y=mean_valence), color="red", size=1.5) +
  scale_y_continuous(labels = scales::percent_format()) +
  theme_piano() +
  scale_fill_manual(values=c("major" = "white", "minor" = "black"), name="Mode") +
  scale_color_manual(values=c("Average Valence" = "red"), name="Metric", labels=c("Av
erage Valence"))+
  labs(x="Album (Release Year)", y="Percentage", fill="Mode",
       title="Mean Valence and Relative Frequency of Modes for Each Album",
       subtitle = "How sad or happy is Taylor Swift music over time?") +

  scale_x_discrete(labels=function(x) paste(x, album_data_combined$album_release_year
[match(x, album_data_combined$album_name)], sep="\n"))
```

```
## Warning: Using `size` aesthetic for lines was deprecated in ggplot2 3.4.0.
## ℹ Please use `linewidth` instead.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.
```



```
# calculate the average acousticness for each album
average_acousticness <- all_songs_filtered |>
  filter(!is.na(acousticness)) |>
  group_by(album_name, album_release_year) |>
  summarise(average_acousticness = mean(acousticness, na.rm = TRUE), .groups = 'dro
p') |>
  arrange(album_release_year, album_name)

overall_average = mean(average_acousticness$average_acousticness, na.rm = TRUE)
# draw
ggplot(average_acousticness, aes(x=reorder(album_name, album_release_year), y=average
_acousticness)) +
  geom_segment(aes(xend=reorder(album_name, album_release_year), yend=0), color="skyb
lue") +
  geom_point(color="steelblue", size=3) + # Draws the dots
  geom_hline(yintercept = overall_average, linetype="dashed", color="red", size=0.5)
+  #trendline
  theme_clean() +
  theme(panel.background = element_rect(fill="white"))+
  annotate("text", x=3, y=overall_average+0.01, label="Average across albums", hjust=
1, vjust=0, color="red", size=3.5) +
  labs(x="Album (Release Year)", y="Average Acousticness",
       title="Average Acousticness of Each Album") +
  scale_x_discrete(labels=function(x) paste(x, average_acousticness$album_release_yea
r[match(x, average_acousticness$album_name)], sep="\n"))
```


Average Acousticness of Each Album
Average Acousticness
0.8
0.6
0.4
0.2
0.0
Average across albums
Taylor Swift 2006
Fearless 2008
Speak Now 2010
Red 2012
1989 2014
reputation 2017
Lover 2019
evermore 2020
folklore 2020
Midnights 2022
Album (Release Year)