Swearing and Religion in American Rap in the 21st Century

Henrique Sposito and Livio Silva-Muller

2022-10-14

Data

Loading data

As a case study, let's analyze and compare a few famous American Rappers in the 21st Century.

After loading the packages e datasets, we need to join these together.

If we simply join the datasets though, we will lose the name of the artist. So let's add a name variable to each dataset before joining them.

```
Eminem$rapper <- "Eminem"
Kanye_West$rapper <- "Kanye West"
Jay_Z$rapper <- "Jay Z"
Kendrick_Lamar$rapper <- "Kendrick Lamar"
# # How could you do this in tidy?
# Eminem <- Eminem %>% mutate( rapper= "Eminem")
```

Joining data

Now let's join data!

```
american_rappers <- full_join(Eminem,
                              full_join(Kanye_West,
                                        full_join(Jay_Z, Kendrick_Lamar)))
```

Please notice that we can only join them without specifying the "by =" argument because the datasets have the same variables with the same names.

However, this is not a very elegant solution. A more elegant solution for joining multiple datasets would entail using the ´{purrr}´ package.

```
library(purrr)
american_rappers <- list(Eminem, Kanye_West, Jay_Z, Kendrick_Lamar) %>%  reduce(full_join)
```

Data cleaning and wrangling

Let's clean and wrangle the data now.

First, there are some songs in the dataset that do not come from rappers' albums, but from somewhere else. In the album variables in the dataset, songs that come from an album from one of the rappers start with "album:".

```
american_rappers$album <- ifelse(startsWith(american_rappers$album, "album:"),
                                 gsub("album:", "", american_rappers$album),
                                 NA_character_)
# # The same with tidy:
# american_rappers <- american_rappers %>%
#   mutate(album = ifelse(stringr::str_detect(album, "album:"),
#                         stringr::str_replace(album, "album:", ""),
#                         NA_character_))
```

Second, we can also extract the date from the album varible to create an 4 digit year variable.

```
american_rappers$year <- as.numeric(stringr::str_extract_all(american_rappers$album,
                                                             "[:digit:]{4}"))
```

Third, let's remove songs for which we are missing the album or that come from album collaborations.

```
american_rappers <- na.omit(american_rappers)
```

Fourth, let's clean the text by removing signs, transform to lower case, and more.

```
#removing punctuation
american_rappers$lyrics <- tm::removePunctuation(american_rappers$lyrics)
#making all lowercase
american_rappers$lyrics <- tolower(american_rappers$lyrics)
# remove line markers
american_rappers$lyrics <- gsub("\r|\n", " ", american_rappers$lyrics)
# remove punctuation
american_rappers$title <- tm::removePunctuation(american_rappers$title)
# remove the years from albums
american_rappers$album <- stringr::str_remove_all(american_rappers$album,
                                                  "\\([:digit:]{4}\\)")
```

Analysis

Dictionary

Create a dictionary for religion and swearing: can you help?

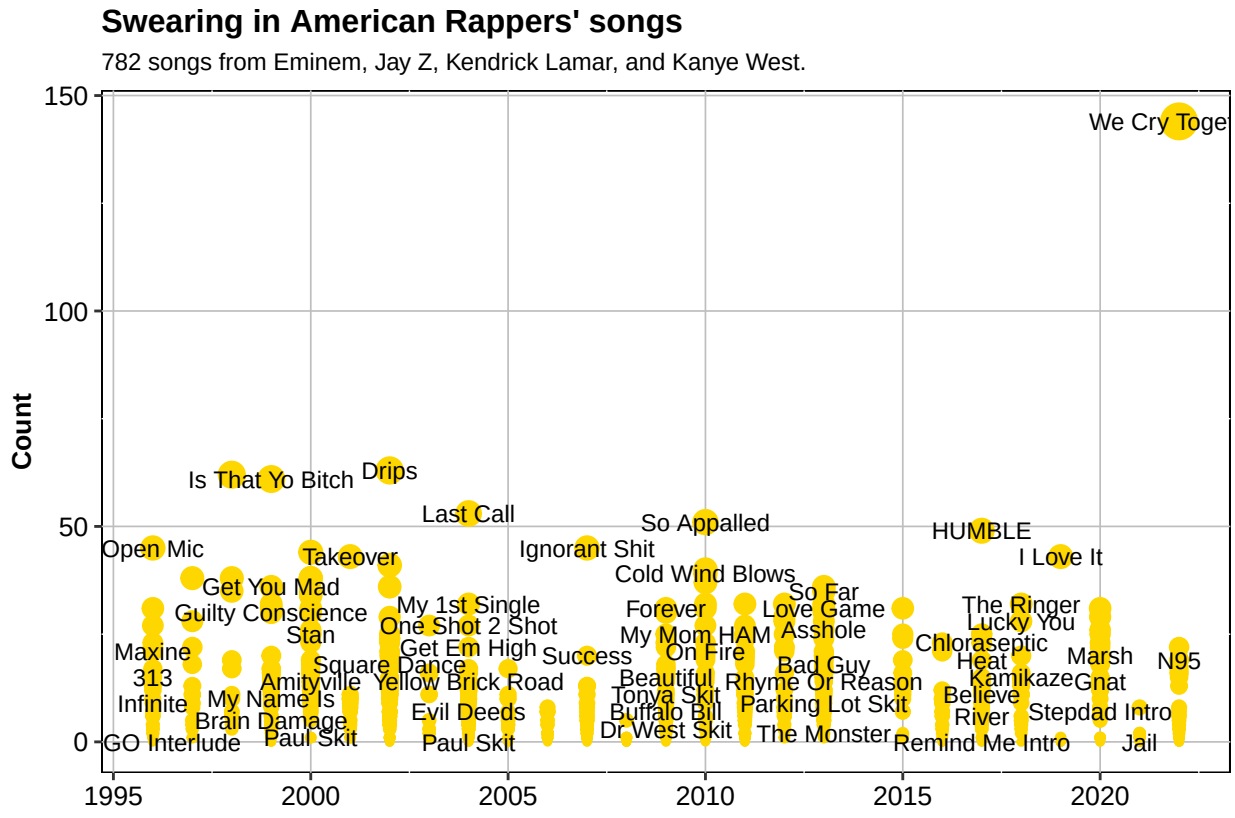
```
swear_words <- "fuck|bitch|pussy|shit|dick|ass|cunt"
religious_words <- "god|bible|jesus|hell|heaven|lord|praise"
```

Count

```
# base for counting swear words
american_rappers$swear_words <- stringr::str_count(american_rappers$lyrics,
                                                   swear_words)
# tidy for religion
american_rappers <- american_rappers %>%
  mutate(religious_words = stringr::str_count(american_rappers$lyrics,
                                              religious_words))
```

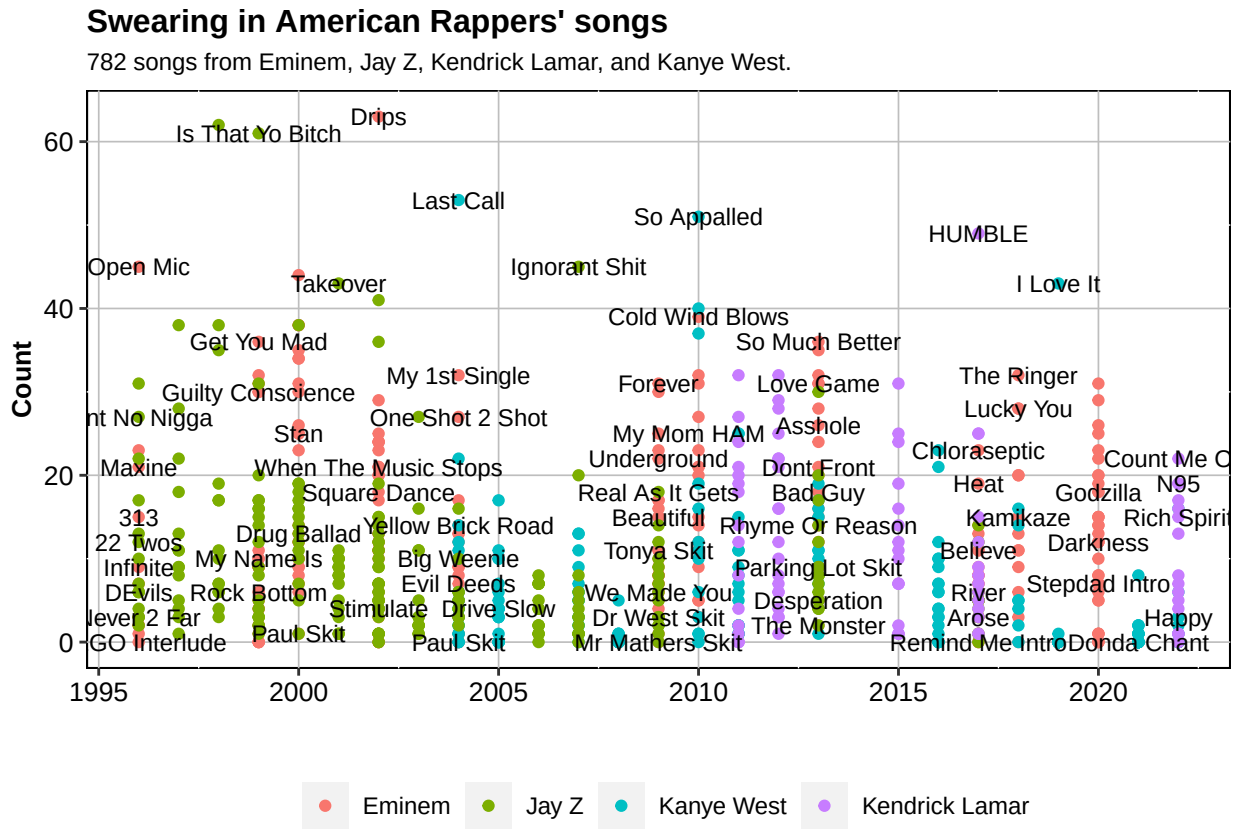
Swearing in American Rappers' songs

```
ggplot(american_rappers, aes(year, swear_words)) +
  geom_point(aes(size = swear_words), color="gold") +
  geom_text(aes(label = title), check_overlap=T, size=3) +
  labs(x = "", y = "Count",
       title = "Swearing in American Rappers' songs",
       subtitle= "782 songs from Eminem, Jay Z, Kendrick Lamar, and Kanye West.")+
  theme(panel.background = element_rect("white", "black", .5, "solid"),
        panel.grid.major = element_line(color = "grey",
                                        size = 0.3,
                                        linetype = "solid"),
        axis.text = element_text(color = "black", size = 10),
        title = element_text(color = "black", size = 10, face = "bold"),
        legend.title = element_blank(),
        plot.subtitle = element_text(color = "black", size = 9, face = "plain"),
        legend.position = "none")
```



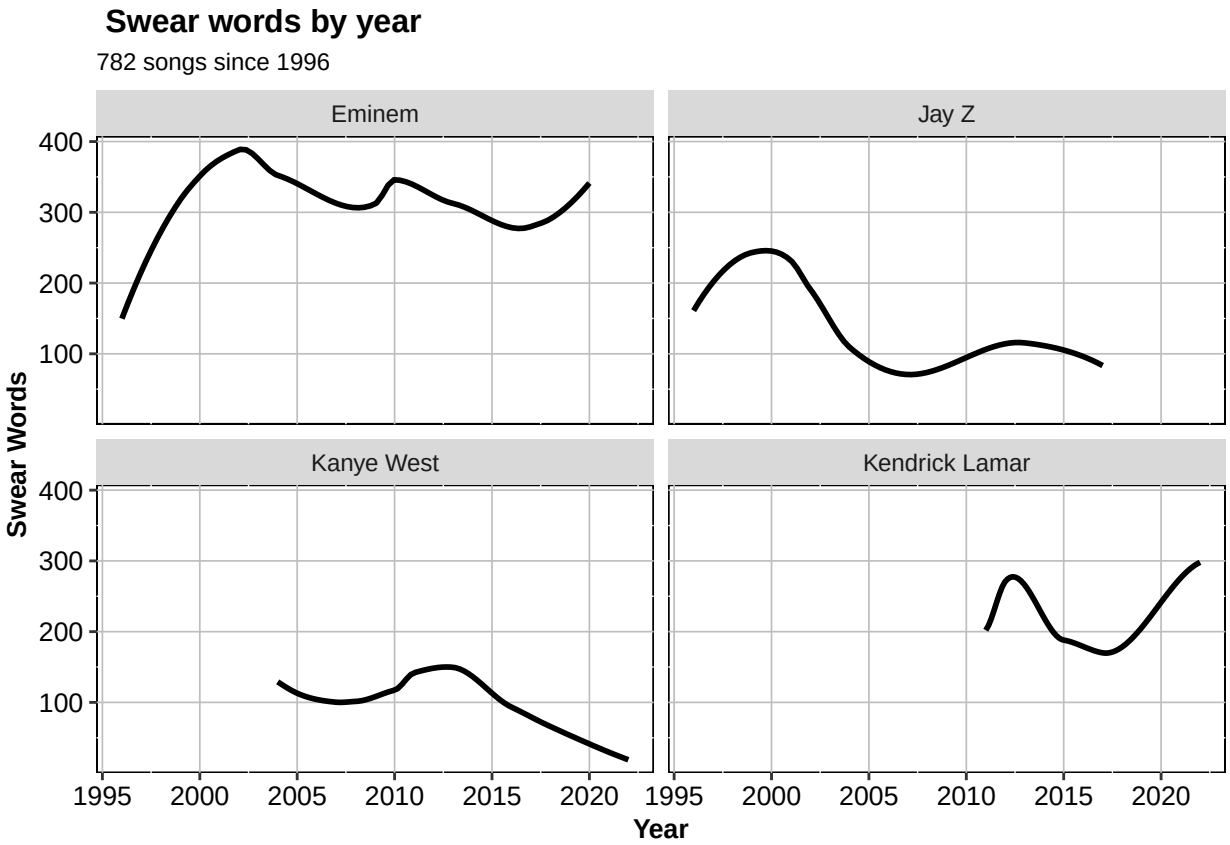
What are some problems with this plot? How can we deal with outliers?

```
american_rappers %>% filter(swear_words < 100) %>%
  ggplot(., aes(year, swear_words)) +
  geom_point(aes(color = rapper)) +
  geom_text(aes(label = title), check_overlap=T, size=3) +
  labs(x = "", y = "Count",
       title = "Swearing in American Rappers' songs",
       subtitle= "782 songs from Eminem, Jay Z, Kendrick Lamar, and Kanye West.")+
  theme(panel.background = element_rect("white", "black", .5, "solid"),
        panel.grid.major = element_line(color = "grey", size = 0.3,
                                        linetype = "solid"),
        axis.text = element_text(color = "black", size = 10),
        title = element_text(color = "black", size = 10, face = "bold"),
        legend.title = element_blank(),
        plot.subtitle = element_text(color = "black", size = 9, face = "plain"),
        legend.position = "bottom")
```



Has swearing increased, or not, in time

```
american_rappers %>% group_by (year, rapper) %>%
  summarise(swear_words = sum(swear_words, na.rm = TRUE))%>%
  ggplot(., aes(year, swear_words)) +
  geom_smooth(se=FALSE, color="black") +
  labs(x = "Year", y = "Swear Words",
       title = " Swear words by year",
       subtitle= "782 songs since 1996")+
  theme(panel.background = element_rect("white", "black", .5, "solid"),
        panel.grid.major = element_line(color = "grey", size = 0.3,
                                        linetype = "solid"),
        axis.text = element_text(color = "black", size = 10),
        title = element_text(color = "black", size = 10, face = "bold"),
        legend.title = element_blank(),
        plot.subtitle = element_text(color = "black", size = 9, face = "plain"),
        legend.position = "none") +
  facet_wrap(~rapper)
```

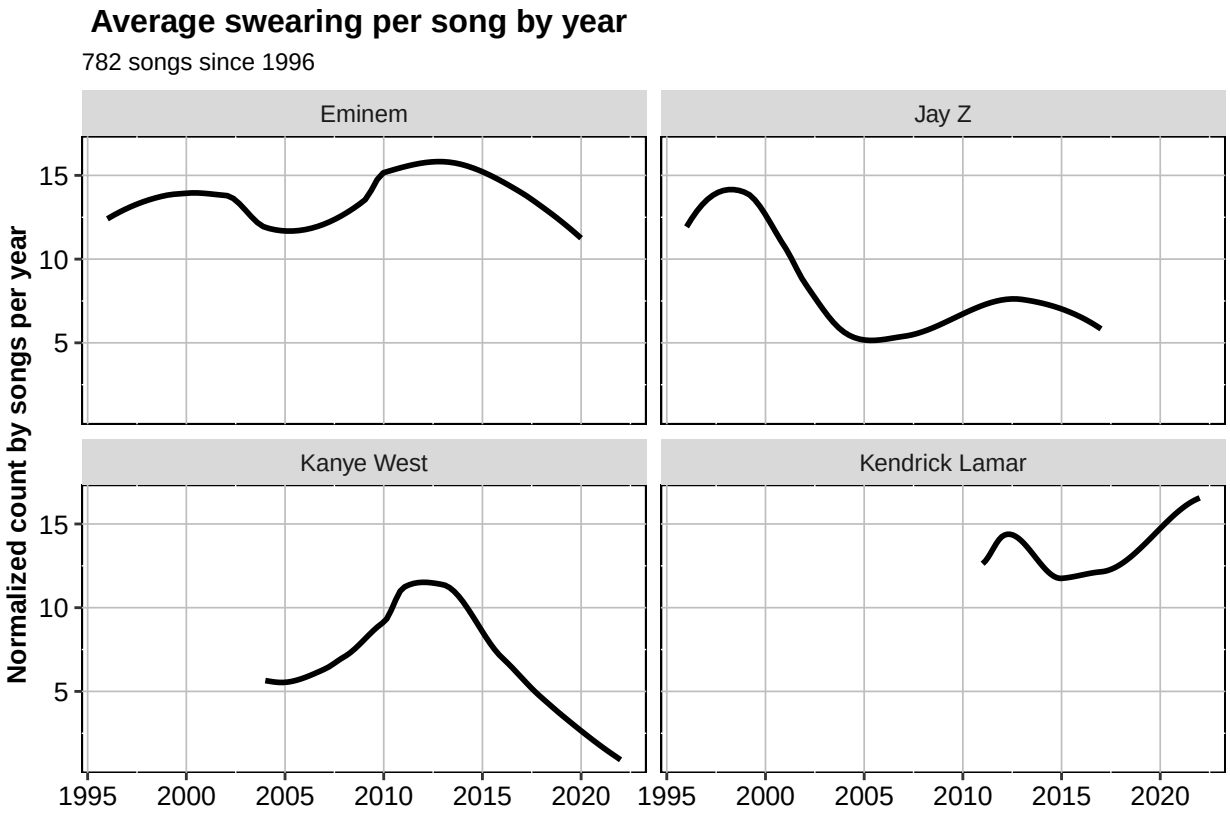


What are some issue is with this analysis?

Yes, we need to normalize scores!

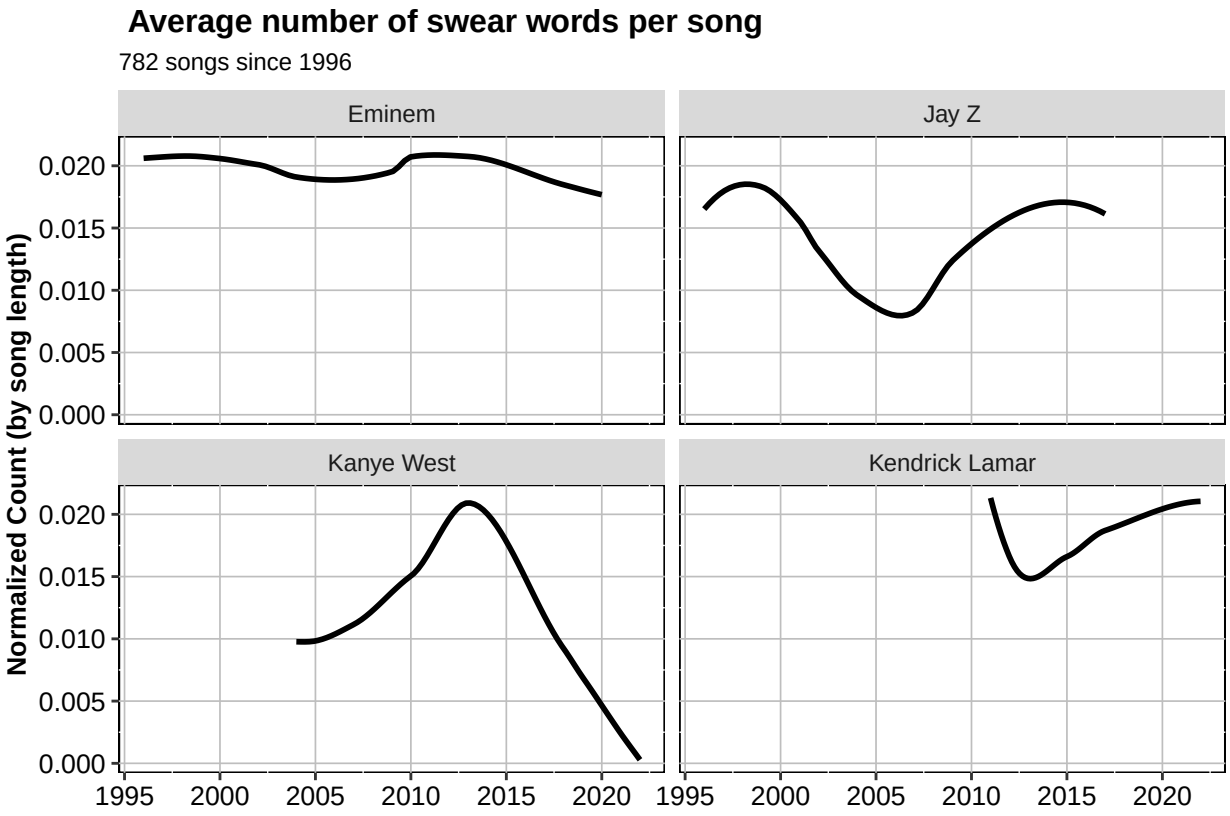
But what would be the best normalization given our data?

```
american_rappers %>%
  group_by(year, rapper) %>%
  mutate(songs_per_year = n()) %>%
  group_by(year,songs_per_year, rapper) %>%
  summarise(swear_words = sum(swear_words, na.rm = TRUE)) %>%
  mutate(normalized_swear_words = swear_words/songs_per_year) %>%
  #all lines until here are normalizing by songs per year
  ggplot(., aes(year, normalized_swear_words)) +
  geom_smooth(se=FALSE, color="black") +
  labs(x = "", y = "Normalized count by songs per year",
       title = " Average swearing per song by year",
       subtitle= "782 songs since 1996")+
  theme(panel.background = element_rect("white", "black", .5, "solid"),
        panel.grid.major = element_line(color = "grey", size = 0.3,
                                        linetype = "solid"),
        axis.text = element_text(color = "black", size = 10),
        title = element_text(color = "black", size = 10, face = "bold"),
        legend.title = element_blank(),
        plot.subtitle = element_text(color = "black", size = 9, face = "plain"),
        legend.position = "none") +
  facet_wrap(~rapper)
```



Let's actually normalize by the number of songs in per album.

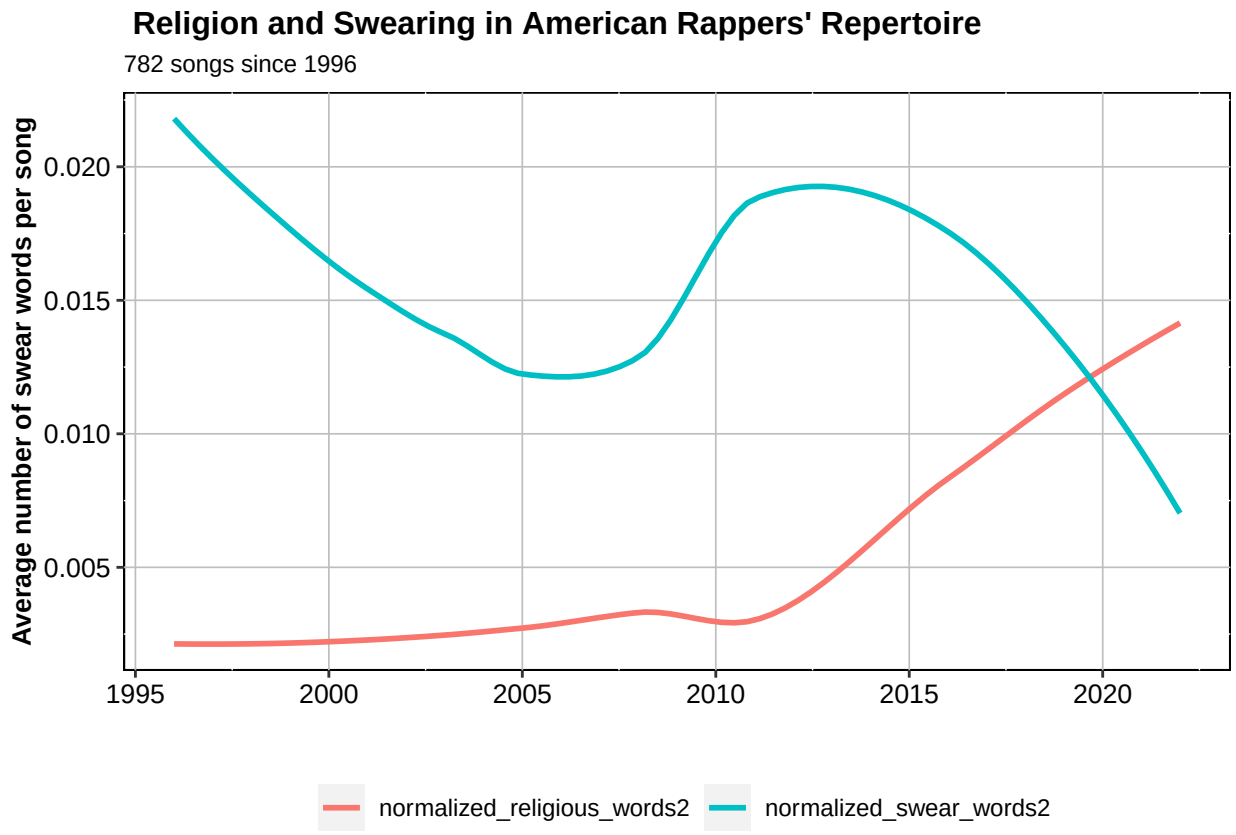
```
american_rappers$word_per_song <-str_count(american_rappers$lyrics, "\\w+")
# counting words in lyrics
american_rappers$normalized_swear_words2 <- american_rappers$swear_words /
  american_rappers$word_per_song # normalize for swera words
american_rappers$normalized_religious_words2 <- american_rappers$religious_words /
  american_rappers$word_per_song # normalize for religious words
# Plot
american_rappers %>%
  ggplot(., aes(year, normalized_swear_words2)) +
  geom_smooth(se=FALSE, color="black") +
  labs(x = "", y = "Normalized Count (by song length)",
       title = " Average number of swear words per song",
       subtitle= "782 songs since 1996") +
  theme(panel.background = element_rect("white", "black", .5, "solid"),
        panel.grid.major = element_line(color = "grey", size = 0.3,
                                        linetype = "solid"),
        axis.text = element_text(color = "black", size = 10),
        title = element_text(color = "black", size = 10, face = "bold"),
        legend.title = element_blank(),
        plot.subtitle = element_text(color = "black", size = 9, face = "plain"),
        legend.position = "none") +
  facet_wrap(~rapper)
```



What happens with the unit of measurement (i.e. average number of swear words per song)?

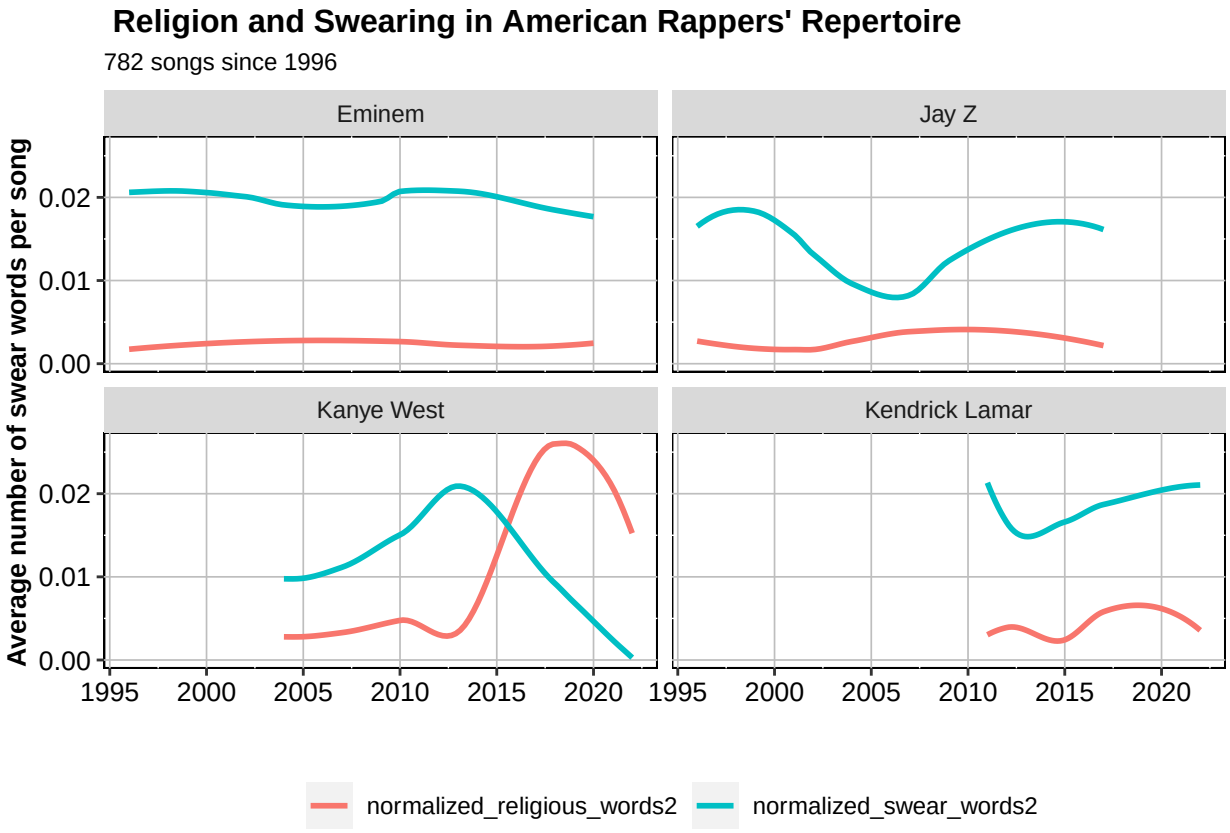
Religious vocabulary in time

```
american_rappers %>%
  gather("topic", "normalized_count2", 9:10) %>%
  ggplot(., aes(x=year, y=normalized_count2, color=topic)) +
  geom_smooth(se=FALSE) +
  labs(x = "", y = "Average number of swear words per song",
       title = " Religion and Swearing in American Rappers' Repertoire",
       subtitle= "782 songs since 1996")+
  theme(panel.background = element_rect("white", "black", .5, "solid"),
        panel.grid.major = element_line(color = "grey", size = 0.3,
                                        linetype = "solid"),
        axis.text = element_text(color = "black", size = 10),
        title = element_text(color = "black", size = 10, face = "bold"),
        legend.title = element_blank(),
        plot.subtitle = element_text(color = "black", size = 9, face = "plain"),
        legend.position = "bottom")
```



Do you think there is a trend a wider trend in American Rappers using more religious words in songs?

```
american_rappers %>%
  gather("topic", "normalized_count2", 9:10) %>%
  ggplot(., aes(x=year, y=normalized_count2, color=topic)) +
  geom_smooth(se=FALSE) +
  labs(x = "", y = "Average number of swear words per song",
       title = " Religion and Swearing in American Rappers' Repertoire",
       subtitle= "782 songs since 1996")+
  theme(panel.background = element_rect("white", "black", .5, "solid"),
        panel.grid.major = element_line(color = "grey", size = 0.3,
                                        linetype = "solid"),
        axis.text = element_text(color = "black", size = 10),
        title = element_text(color = "black", size = 10, face = "bold"),
        legend.title = element_blank(),
        plot.subtitle = element_text(color = "black", size = 9, face = "plain"),
        legend.position = "bottom") +
  facet_wrap(~rapper)
```



Is there really a wider trend or this is all driven by Kayne?

The Kardashian effect

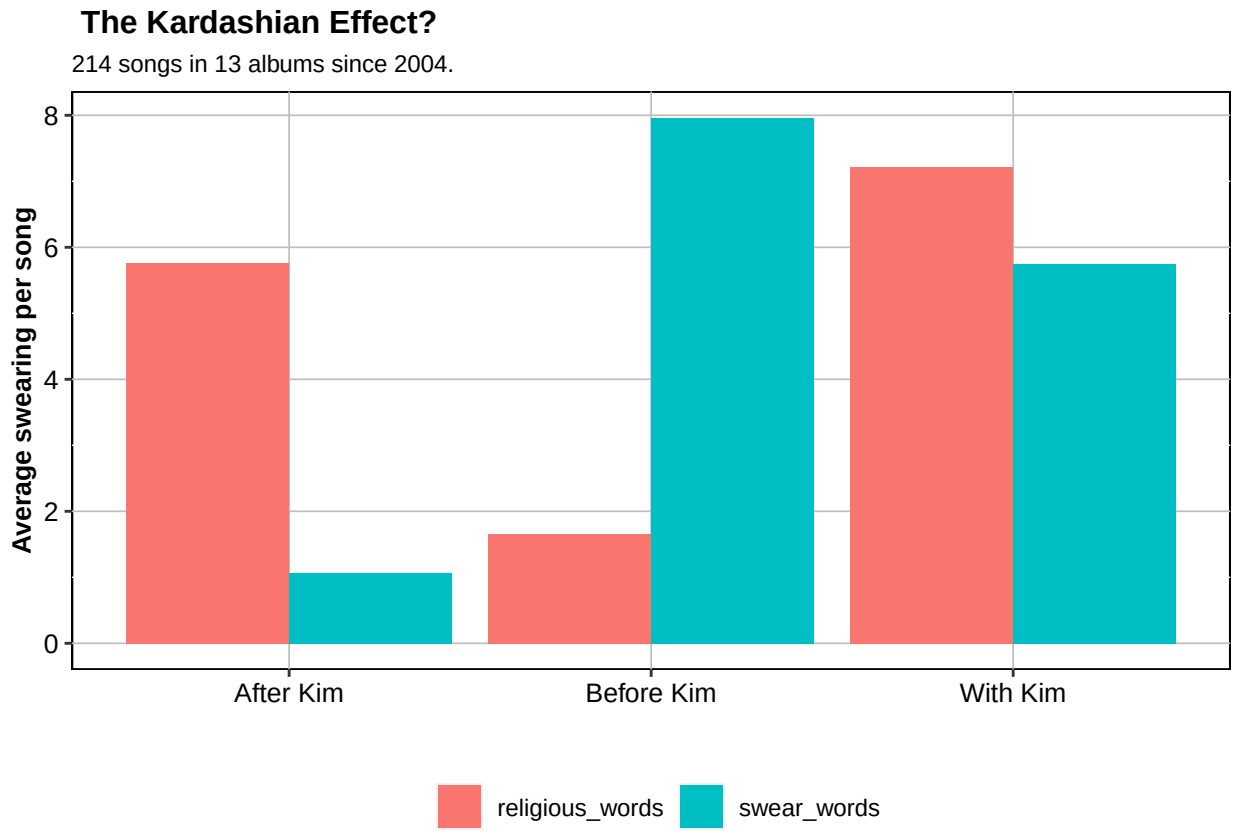
Is there a Kardashian effect in this switch we see for Kayne?

Kanye and Kim were a couple from 2011 to 2020, let's create a Kim variable!

```
Kanye_West <- american_rappers %>%
  filter(rapper== "Kanye West")%>%
  mutate(kim_kardashian= case_when(year > 2010 & year <= 2020  ~ "With Kim",
                                   year < 2011   ~ "Before Kim",
                                   year >2019   ~ "After Kim"))
```

```
Kanye_West %>%
  gather("topic", "word_count", 6:7) %>%
  group_by(year) %>%
  mutate(songs_per_year = n()) %>%
  group_by(songs_per_year, topic, kim_kardashian) %>%
  summarise(word_count = sum(word_count, na.rm = TRUE))%>%
  mutate(normalized_word_count = word_count/songs_per_year)%>%
  ggplot(., aes(x=kim_kardashian, y=normalized_word_count, fill=topic)) +
  geom_bar(stat="identity", position="dodge") +
  labs(x = "", y = "Average swearing per song",
       title = " The Kardashian Effect?",
```

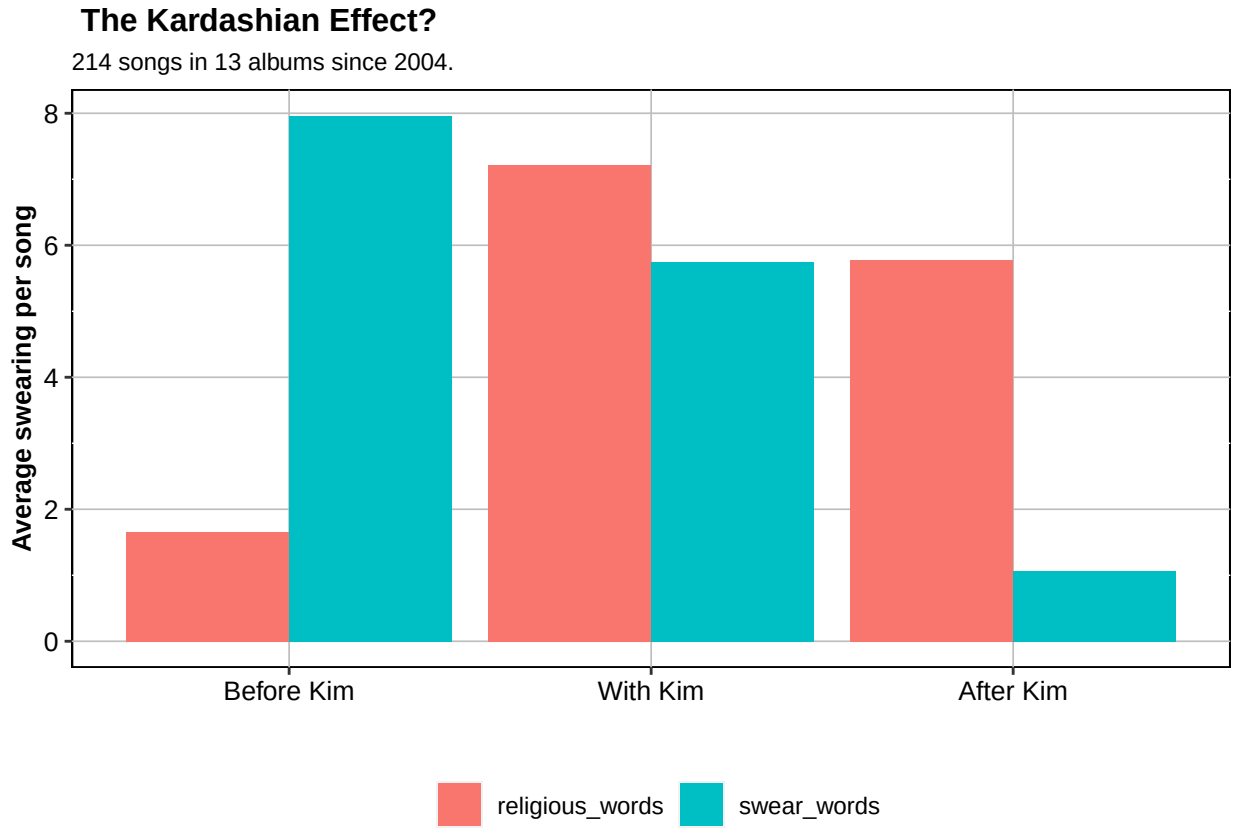
```
      subtitle= "214 songs in 13 albums since 2004.")+
  theme(panel.background = element_rect("white", "black", .5, "solid"),
        panel.grid.major = element_line(color = "grey", size = 0.3,
                                        linetype = "solid"),
        axis.text = element_text(color = "black", size = 10),
        title = element_text(color = "black", size = 10, face = "bold"),
        legend.title = element_blank(),
        plot.subtitle = element_text(color = "black", size = 9, face = "plain"),
        legend.position = "bottom")
```



But this is a weird categorical ordering, no? We can change this!

```
Kanye_West$kim_kardashian <- factor(Kanye_West$kim_kardashian,
                                    levels = c("Before Kim",
                                               "With Kim",
                                               "After Kim"))
Kanye_West %>%
  gather("topic", "word_count", 6:7) %>%
  group_by(year) %>%
  mutate(songs_per_year = n()) %>%
  group_by(songs_per_year, topic, kim_kardashian) %>%
  summarise(word_count = sum(word_count, na.rm = TRUE))%>%
  mutate(normalized_word_count = word_count/songs_per_year)%>%
  ggplot(., aes(x=kim_kardashian, y=normalized_word_count, fill=topic)) +
  geom_bar(stat="identity", position="dodge") +
  labs(x = "", y = "Average swearing per song",
```

```
      title = " The Kardashian Effect?",
      subtitle= "214 songs in 13 albums since 2004.")+
 theme(panel.background = element_rect("white", "black", .5, "solid"),
       panel.grid.major = element_line(color = "grey", size = 0.3,
                                       linetype = "solid"),
       axis.text = element_text(color = "black", size = 10),
       title = element_text(color = "black", size = 10, face = "bold"),
       legend.title = element_blank(),
       plot.subtitle = element_text(color = "black", size = 9, face = "plain"),
       legend.position = "bottom")
```



Lamar's Repertoire

```
ggplot(topWords_t, aes(x=word, y=log(frequency))) +
  geom_bar(stat="identity", fill='gold') +
  coord_flip()+
  geom_text(aes(label=frequency), colour="black",hjust=1.25, size=3.0)+
  labs( x="", y= "", title= "Most frequent words in Lamar's repertoire",
        subtitle="All albums since 2011")+
  theme(panel.background = element_rect ("white", "black", .5, "solid"),
        panel.grid.major = element_line(color="grey", size=0.3, linetype= "solid"),
        axis.text = element_text(color="black", size=10),
        title = element_text(color="black", size=10, face="bold"),
```

```
        axis.text.x=element_blank(),
        plot.subtitle = element_text(color="black", size=9, face= "plain"),
        legend.position = "bottom")
```