

Swearing and Religion in American Rap

Henrique Sposito and Livio Silva-Muller

September 2023

Loading data and packages

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

The datasets above were obtained by running the script "Scrapping_American Rappers.R" for each singer. After loading the packages and 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"
Queen_latifah$rapper <- "Queen Latifah"
Cardi_B$rapper <- "Cardi B"
Nicki_Minaj$rapper <- "Nicki Minaj"

# # How could you do this in tidy?
# Eminem <- Eminem %>% mutate(rapper= "Eminem")
```

Joining data

Now let's join data! One option is to embed a series of full_joins inside full_joins.

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

```
full_join(Nicki_Minaj,
          full_join(Cardi_B, full_join(Queen_latifah,Kendrick_Lamar))))))
```

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. You can learn more about purrr [here](#).

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

There are probably more efficient solutions than this one, but let’s stop here.

Data cleaning and wrangling

Okay, let’s clean and wrangle the data before we can get to the analysis! Open it up and see how messy and dirty it is. What do we need to do?

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$title <- gsub("'", "", american_rappers$title) # getting rid of the titles' quotation marks...

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 variable to create an 4 digit year variable. The package stringr from tidyverse can help us [here](#)! Check it’s nice cheat sheet [here](#).

```
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. The package tm has some nice functions for this.

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

Dictionary and Counting

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

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

# base for counting swear words
american_rappers$swear_words <- stringr::str_count(american_rappers$lyrics,
  swear_words)

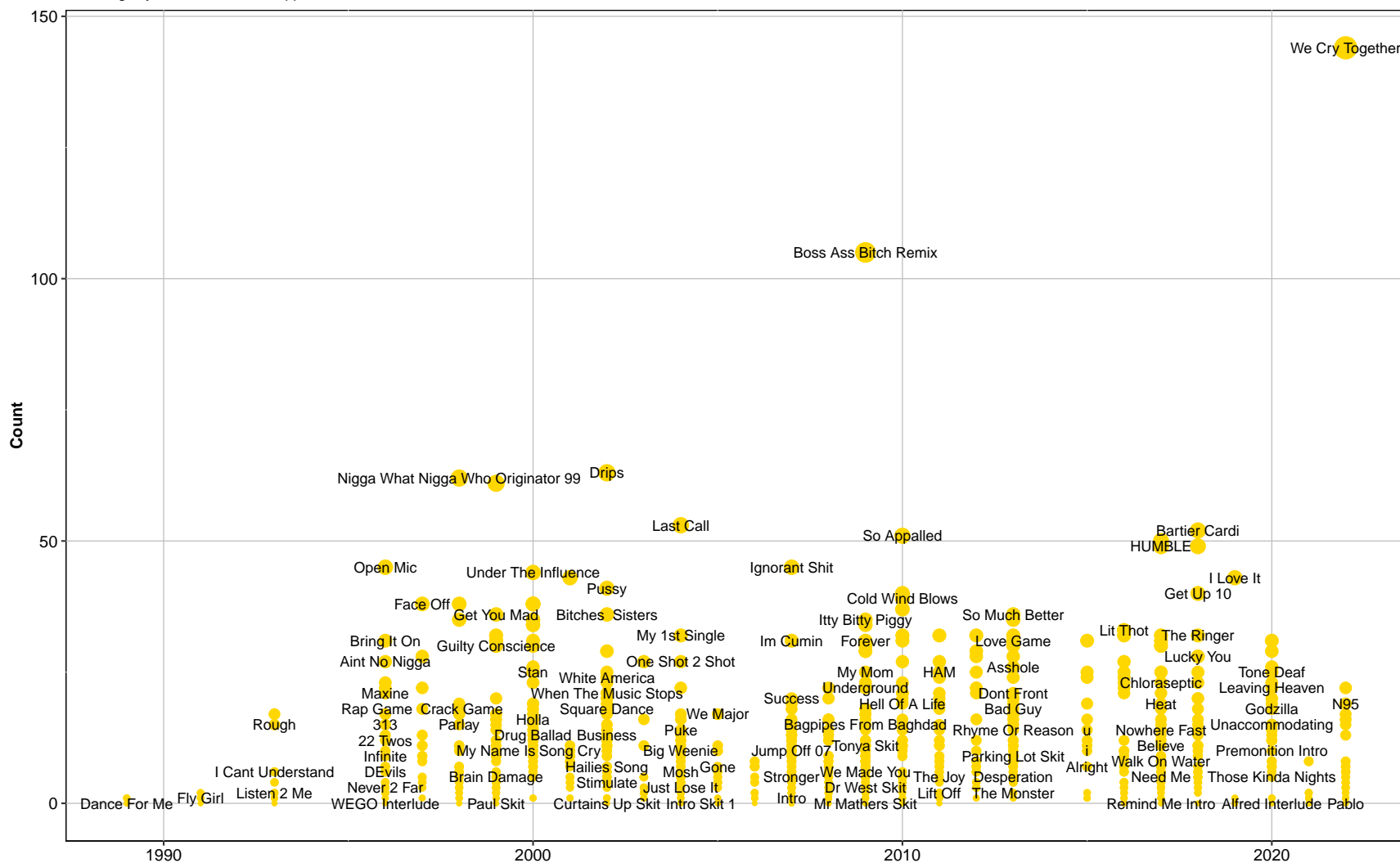
# tidy for religion
american_rappers <- american_rappers %>%
  mutate(religious_words = stringr::str_count(american_rappers$lyrics,
    religious_words))
```

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= "989 songs by seven American Rappers since 1990")+  
  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")
```

Swearing in American Rappers' songs

989 songs by seven American Rappers since 1990



```

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, face="bold") +
  scale_color_brewer(palette="Set2")+
  labs(x = "", y = "Count",
       title = "Swearing in American Rappers' songs",
       subtitle= "989 songs by seven American Rappers since 1990")+
  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")

```

989 songs by seven American Rappers since 1990



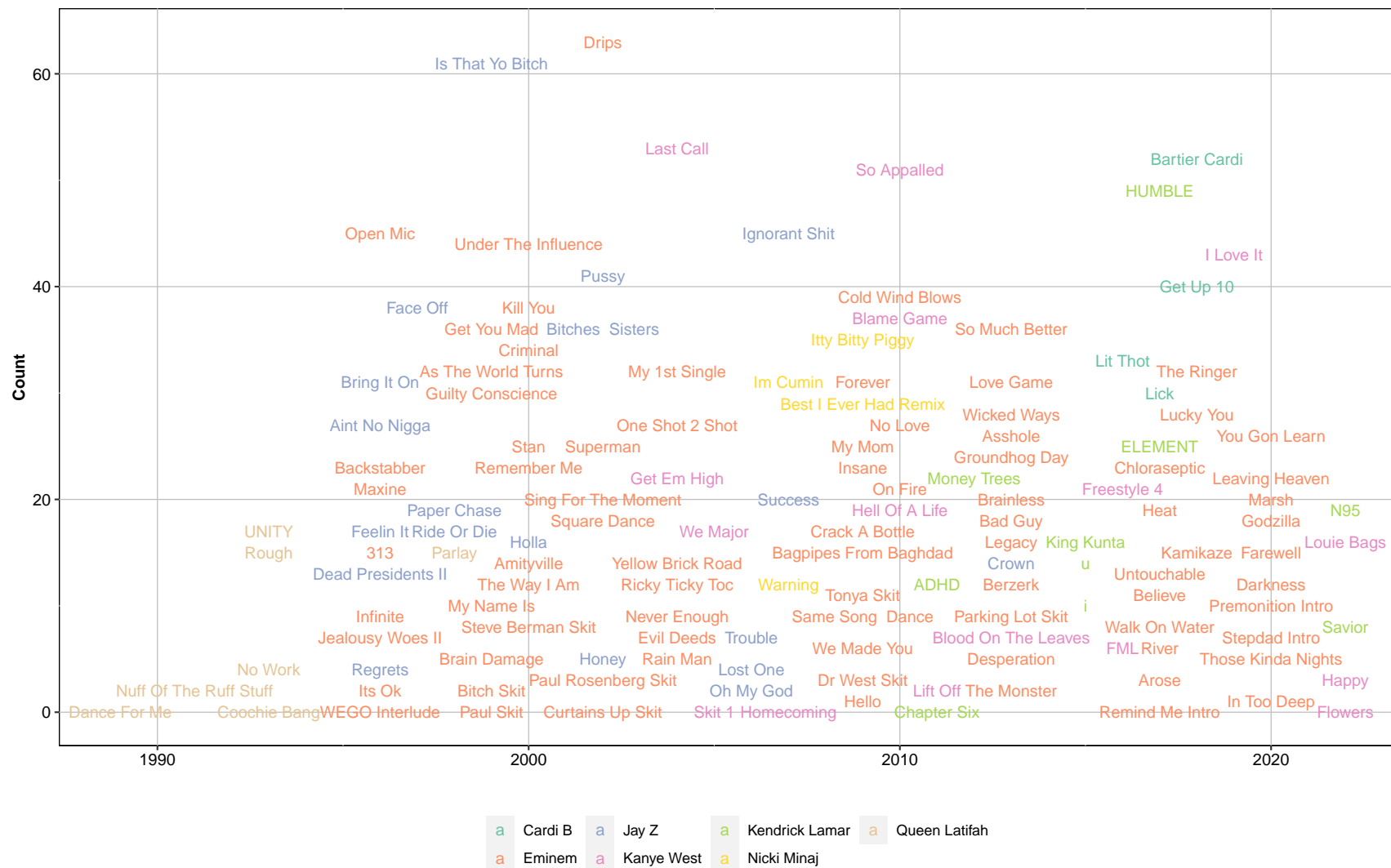
```

american_rappers %>% filter(swear_words < 100) %>%
  ggplot(., aes(year, swear_words)) +
  geom_text(aes(label = title, color=rapper), check_overlap=T, size=3.5, face="bold") +
  scale_color_brewer(palette="Set2")+
  labs(x = "", y = "Count",
       title = "Swearing in American Rappers' songs",
       subtitle= "989 songs by seven American Rappers since 1990")+
  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")

```


Swearing in American Rappers' songs

989 songs by seven American Rappers since 1990



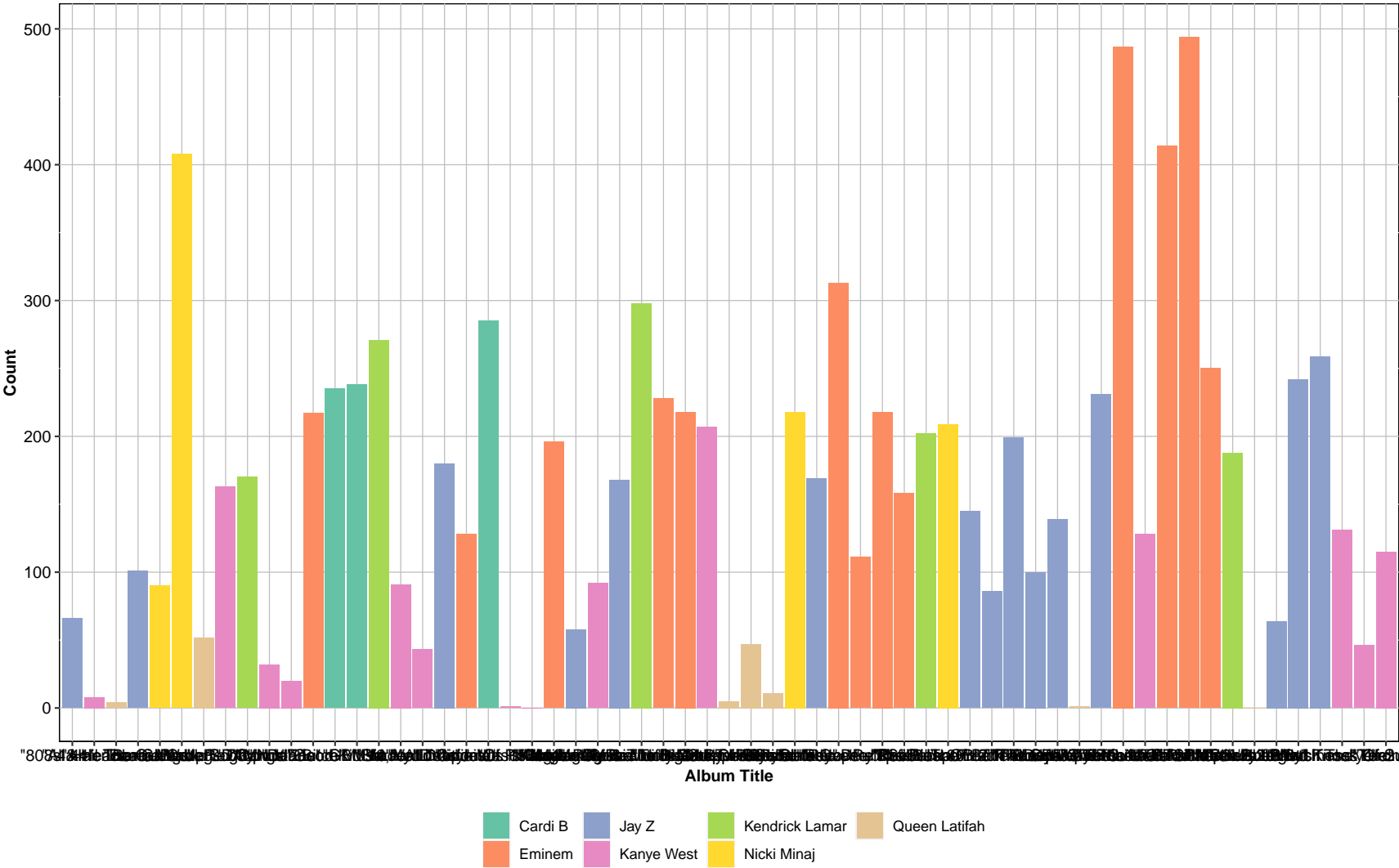
```

# Want to see swearing by albums?
american_rappers %>%
  group_by (album, rapper) %>%
  summarise(swear_words = sum(swear_words, na.rm = TRUE))%>%
  ggplot(., aes(y=swear_words, x=album)) +
  geom_bar(aes(fill=rapper), stat="identity") +
  scale_fill_brewer(palette="Set2")+
  labs(y = "Count", x = "Album Title",
       title = "Swearing in American Rappers' Albums",
       subtitle= "61 Albums 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")

```

Swearing in American Rappers' Albums

61 Albums since 1996



This looks ugly.. How can we improve it a bit?

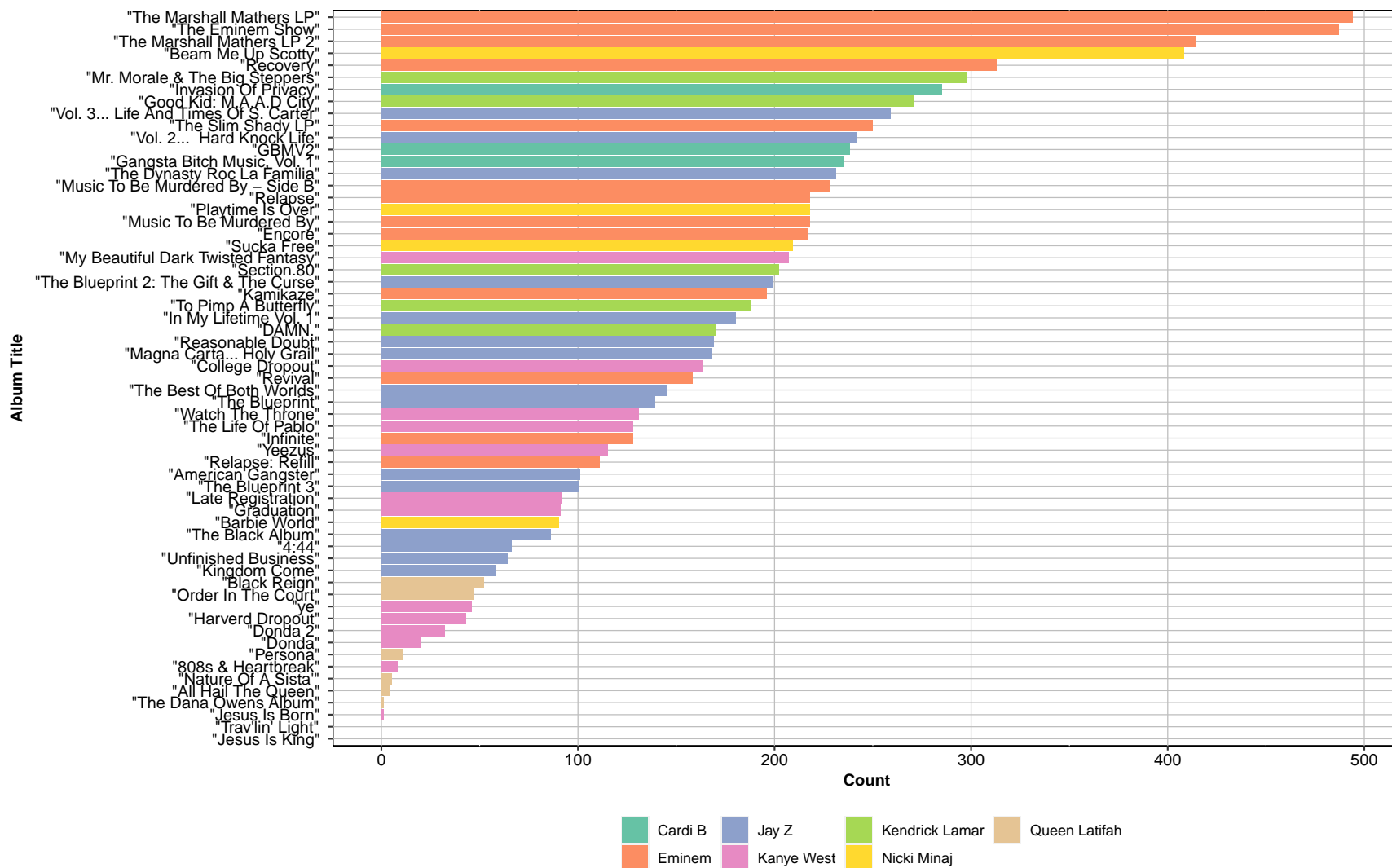
```

# Want to see swearing by albums?
american_rappers %>%
  group_by (album, rapper) %>%
  summarise(swear_words = sum(swear_words, na.rm = TRUE))%>%
  ggplot(., aes(x=swear_words, y=reorder(album, swear_words))) +
  geom_bar(aes(fill=rapper), stat="identity") +
  scale_fill_brewer(palette="Set2")+
  labs(x = "Count", y = "Album Title",
       title = "Swearing in American Rappers' Albums",
       subtitle= "61 Albums 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")

```

Swearing in American Rappers' Albums

61 Albums since 1996

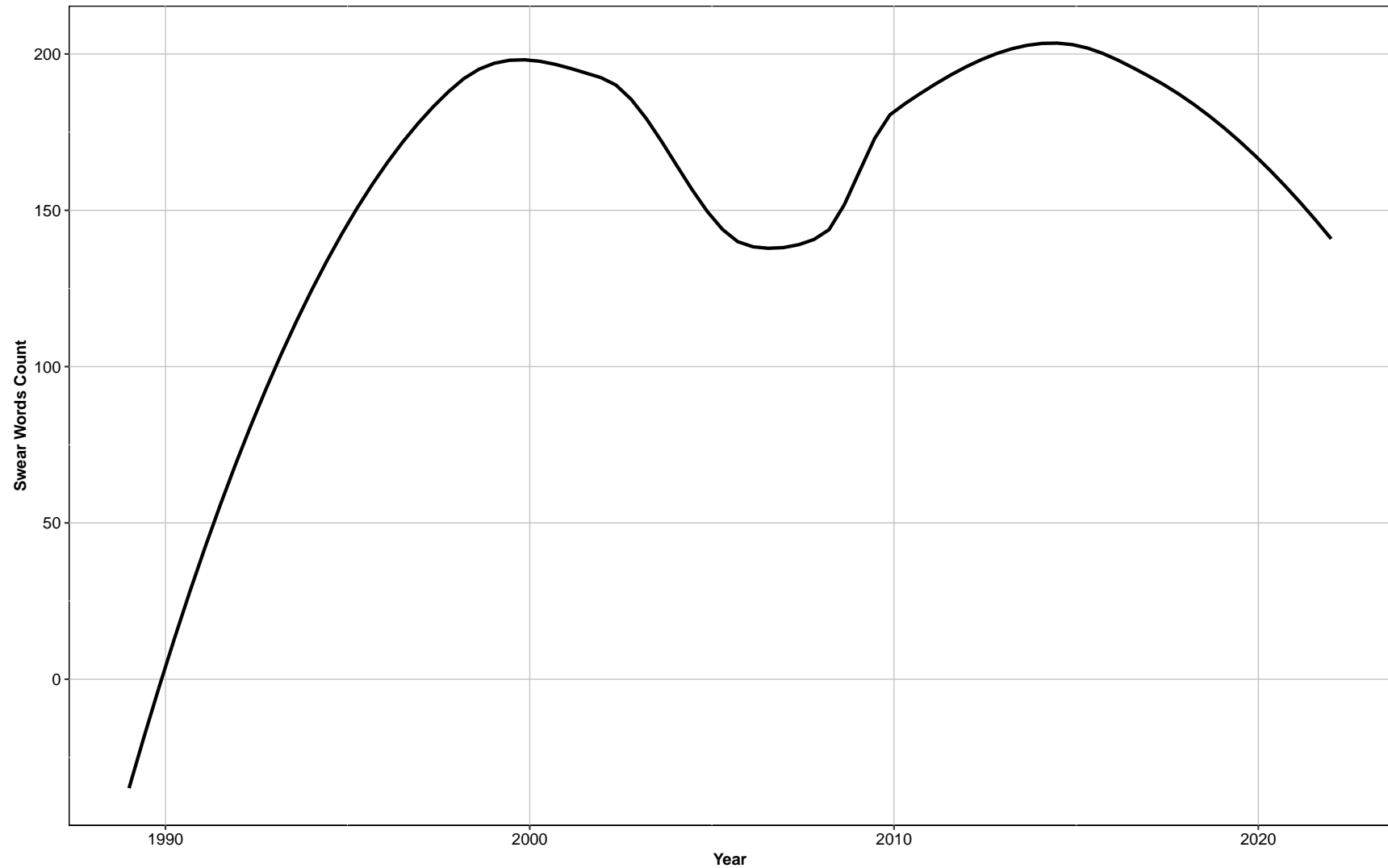


Has swearing increased in time?

```
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 Count",
       title = "Swear words by year",
       subtitle= "989 songs by seven American Rappers since 1990")+
  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")
```

Swear words by year

989 songs by seven American Rappers since 1990



What are the problems with this plot? The dataset is unbalanced, meaning that descriptive trends might reflect different sample compositions along time whether than real patterns in the data:

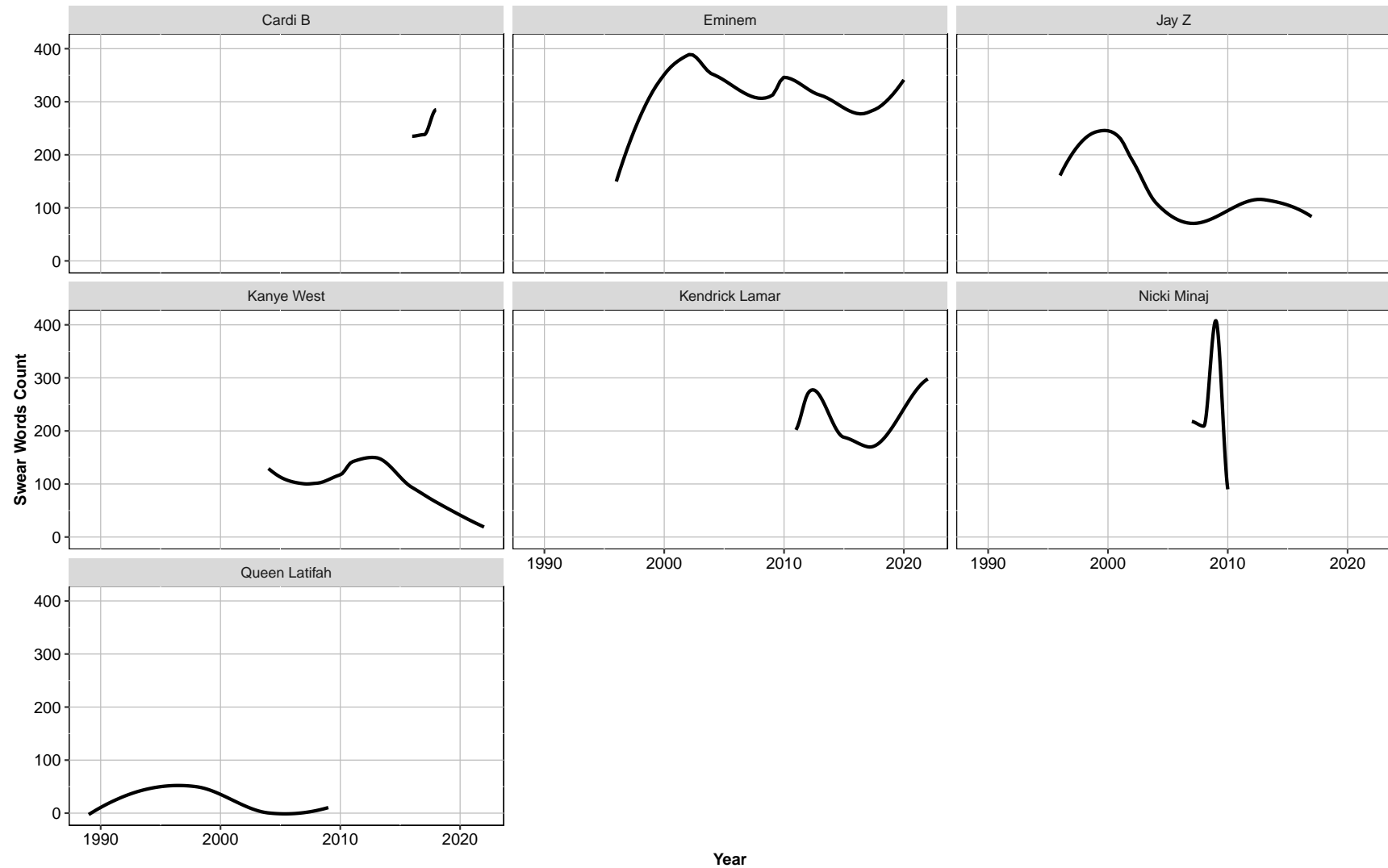
```

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 Count",
       title = "Unbalanced dataset: only three rappers rapping in the 1990s.",
       subtitle= "989 songs by seven American Rappers since 1990")+
  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)

```


Unbalanced dataset: only three rappers rapping in the 1990s.

989 songs by seven American Rappers since 1990



What are some other issue is with this analysis?

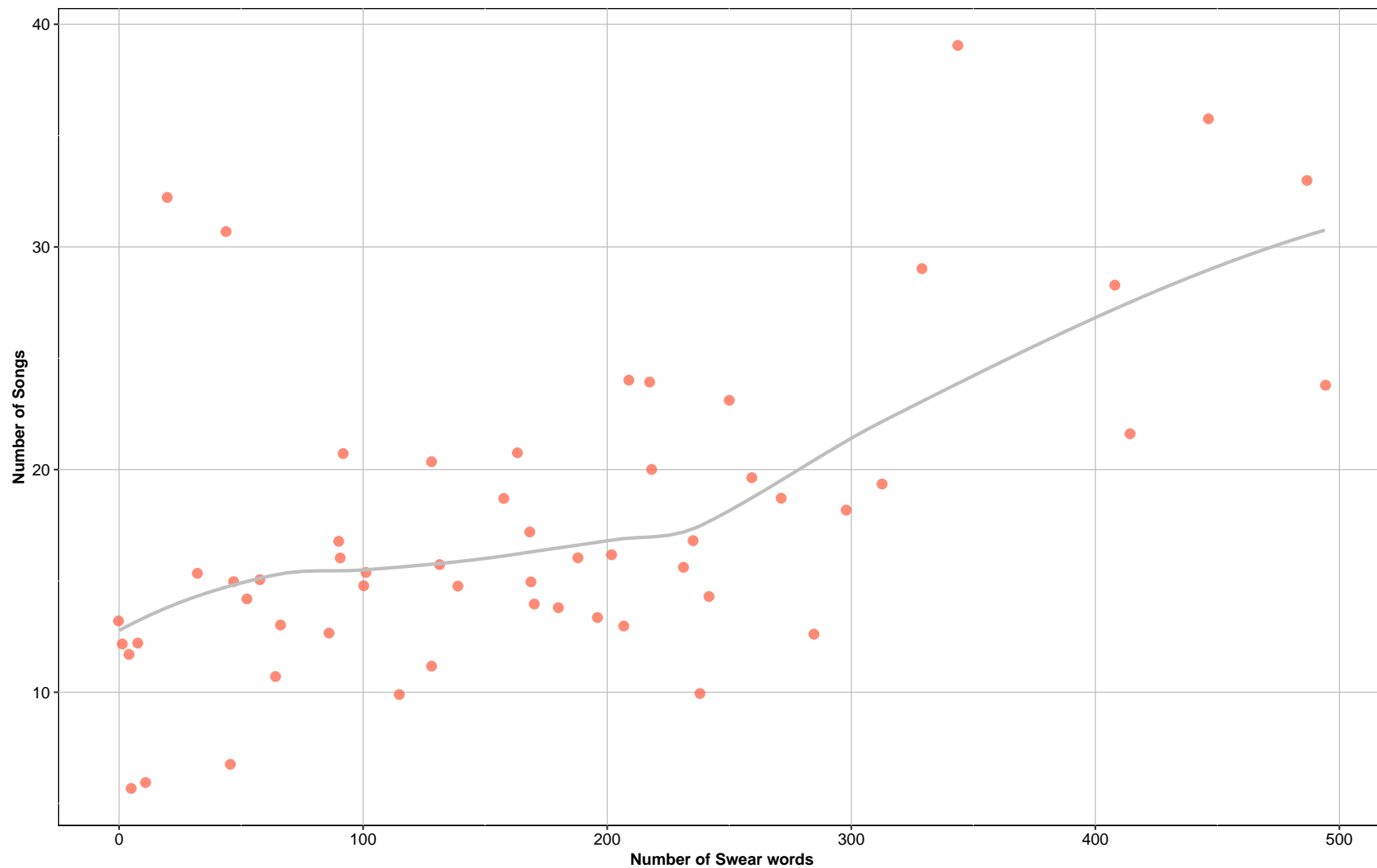
Yes, we need to normalize scores!

But what would be the best normalization given our data?

```
american_rappers %>%
  group_by(year, rapper) %>%
  mutate(songs_per_year = n())%>%
  ungroup()%>%
  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(swear_words, songs_per_year)) +
  geom_jitter(alpha=.75, color="tomato", size=2.5)+
  geom_smooth(se=FALSE, color="grey") +
  labs(x = "Number of Swear words", y = "Number of Songs",
       title = "Yearly relationship between number of songs and swear words",
       subtitle= "989 songs by seven American Rappers since 1990")+
  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")
```

Yearly relationship between number of songs and swear words

989 songs by seven American Rappers since 1990



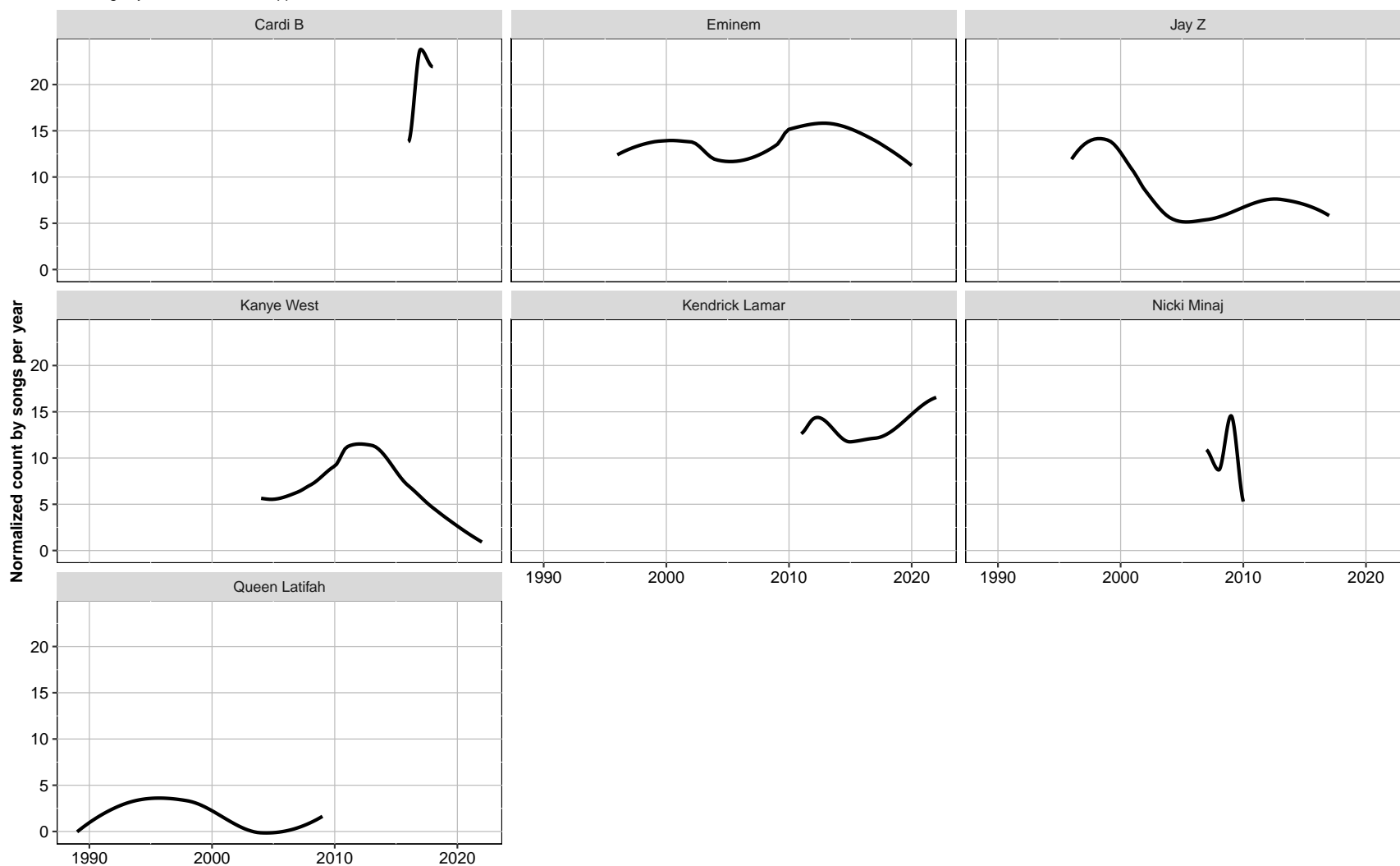
```

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= "989 songs by seven American Rappers since 1990")+
  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)

```

Average swearing per song by year

989 songs by seven American Rappers since 1990



Let's actually normalize by the number of words per song

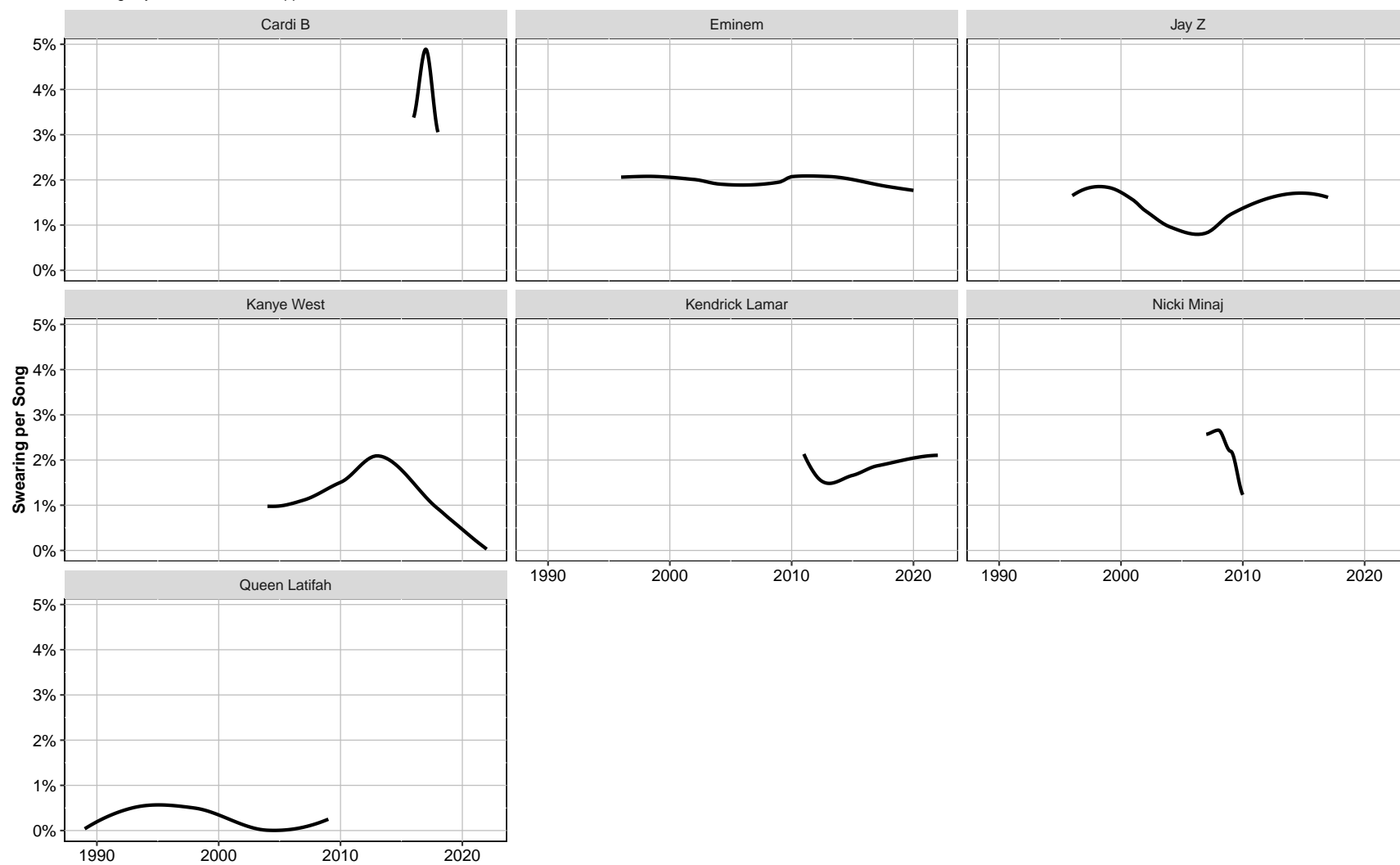
```

american_rappers$word_per_song <-str_count(american_rappers$lyrics, "\\w+")
# counting words in lyrics
american_rappers$Swearing <- american_rappers$swear_words /
  american_rappers$word_per_song # normalize for swear words
american_rappers$Preaching <- american_rappers$religious_words /
  american_rappers$word_per_song # normalize for religious words
# Plot
american_rappers %>%
  ggplot(., aes(year, Swearing)) +
  geom_smooth(se=FALSE, color="black") +
  scale_y_continuous(labels = scales::percent_format())+ # the scales package helps you with making scales more beautiful!
  labs(x = "", y = "Swearing per Song",
        title = "Average number of swear words per song",
        subtitle= "989 songs by seven American Rappers since 1990")+
  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)

```

Average number of swear words per song

989 songs by seven American Rappers since 1990

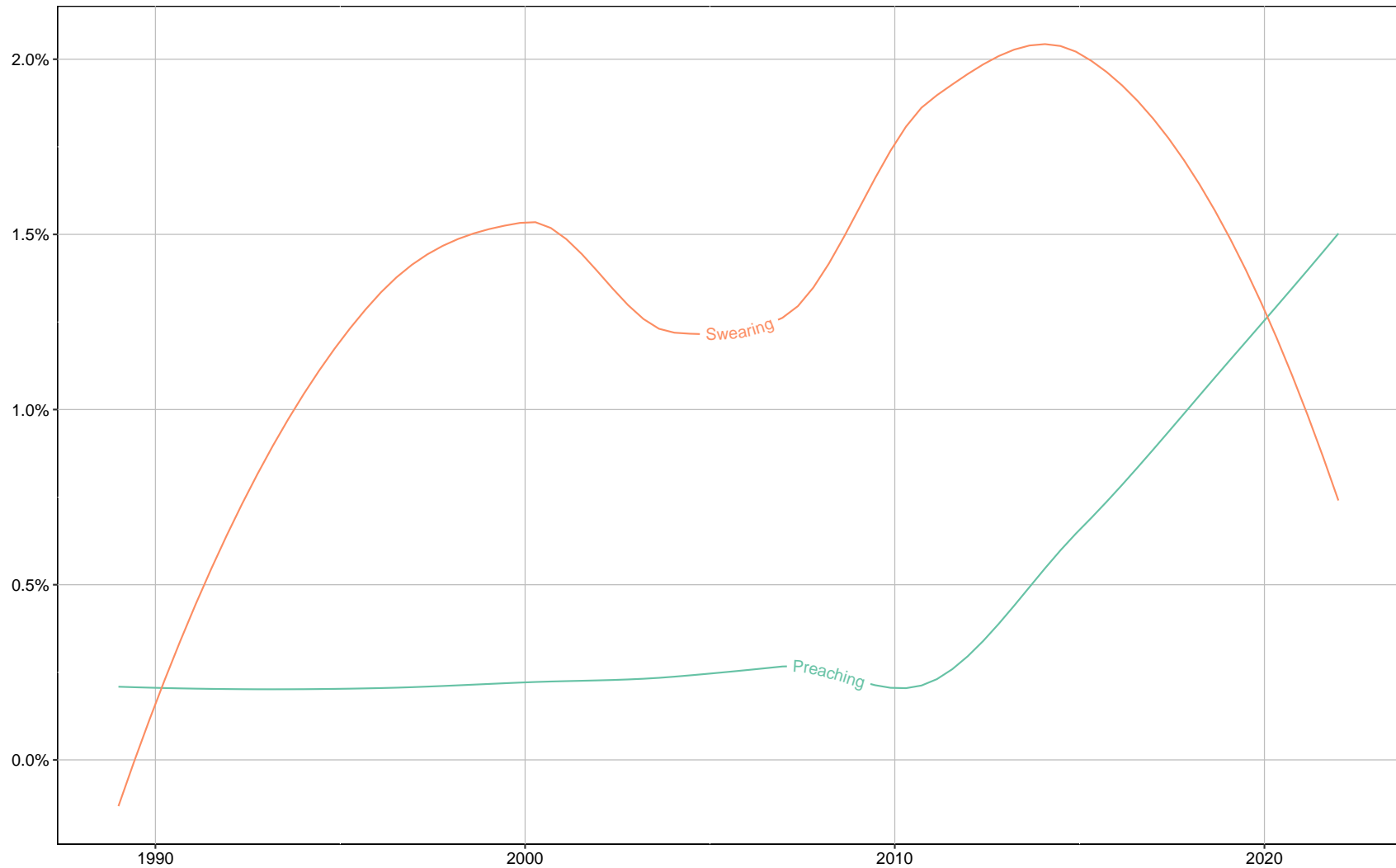


Religious vocabulary in time: smoothing is okay?

```
american_rappers %>%
  group_by(year) %>%
  summarise(Swearing=mean(Swearing),
            Preaching=mean(Preaching)) %>%
  pivot_longer(2:3, names_to="topic", values_to="normalized_count2")%>%
  ggplot(., aes(x=year, y=normalized_count2, color=topic)) +
  geomtextpath::geom_textsmooth(aes(label=topic), #in line legends are nice sometimes, just remember to specify the label argument
    se=FALSE, size=3.5)+
  scale_color_brewer(palette="Set2")+
  scale_y_continuous(labels = scales::percent_format())+
  labs(x = "", y = "",
       title = " Religion and Swearing in American Rappers' Repertoire",
       subtitle= "989 songs by seven American Rappers since 1990")+
  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")
```


Religion and Swearing in American Rappers' Repertoire

989 songs by seven American Rappers since 1990



The plot above feels smooth... Why is it so smooth?

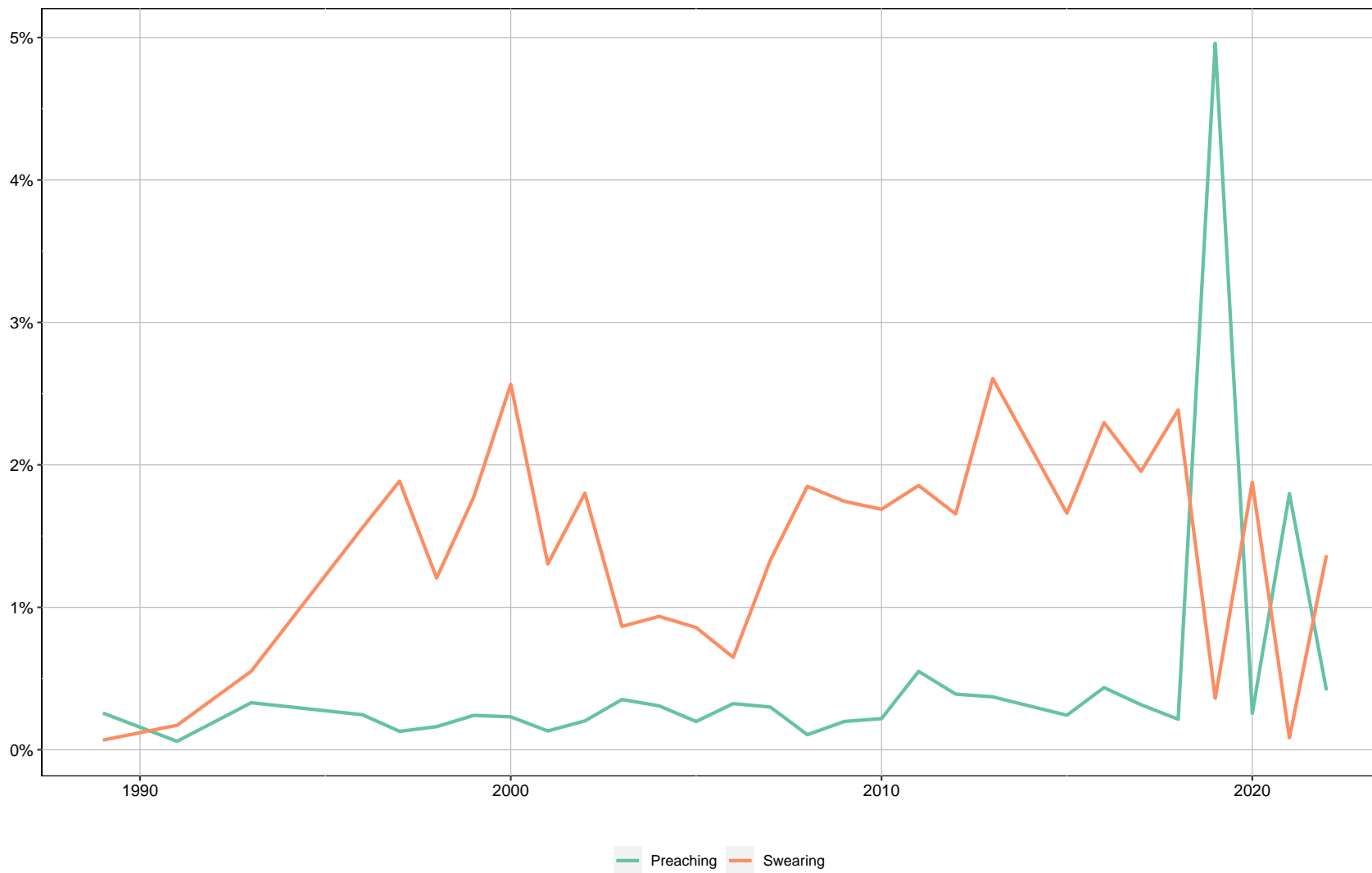
```

american_rappers %>%
  group_by(year) %>%
  summarise(Swearing=mean(Swearing),
            Preaching=mean(Preaching)) %>%
  pivot_longer(2:3, names_to="topic", values_to="normalized_count2") %>%
  ggplot(., aes(x=year, y=normalized_count2, color=topic)) +
  geom_line(size=1)+
  scale_color_brewer(palette="Set2")+
  scale_y_continuous(labels = scales::percent_format())+
  labs(x = "", y = "",
       title = " Religion and Swearing in American Rappers' Repertoire",
       subtitle= "989 songs by seven American Rappers since 1990")+
  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")

```

Religion and Swearing in American Rappers' Repertoire

989 songs by seven American Rappers since 1990



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

```

american_rappers <- american_rappers %>%
  mutate(gender=case_when(rapper == "Cardi B" ~ "Female",
                           rapper == "Nicki Minaj" ~ "Female",
                           rapper == "Queen Latifah" ~ "Female", TRUE ~ "Male"),
         two_thousands=ifelse(year>=2000, "Before 2000s", "After 2000s"),
         swearing = scale(Swearing),
         preaching = scale(Preaching)) #just normalizing the measures to facilitate model interpretation

m1 <- lm(swearing ~ preaching,
         data = american_rappers) #simple linear model with one i.V. and one d.V.

m2 <- lm(swearing ~ preaching + as.factor(gender) +
         as.factor(two_thousands),
         data = american_rappers) # adding a few controls.

m3 <- lm(swearing ~ preaching + as.factor(gender) +
         as.factor(two_thousands)+ as.factor(rapper),
         data = american_rappers)

m4 <- plm(Swearing ~ Preaching + as.factor(gender) + as.factor(two_thousands),
         data = american_rappers, model = "within", index = "rapper") #this one is a fixed effect model indexed by rapper

# a nice way to quickly check your model is tab_model() from the package sjPlot
#tab_model(m4)

```

Some people say tables should always be figures (Kastellec and Leoni 2007)... Let's plot the model:

```

plot_models(m3,m2,m1,
  grid=TRUE,
  ci.lvl = .99,
  m.labels=c("Model 3", "Model 2", "Model 1"),
  axis.labels = c("Rapper(Queen Latifah)", "Rapper(Nicki Minaj)",
                  "Rapper(Kayne West)",
                  "Rapper(Jay Z)", "Rapper(Eminem)",
                  "Period(b4 2000s)", "Gender(male)",
                  "Preaching"),
  show.p = TRUE)+

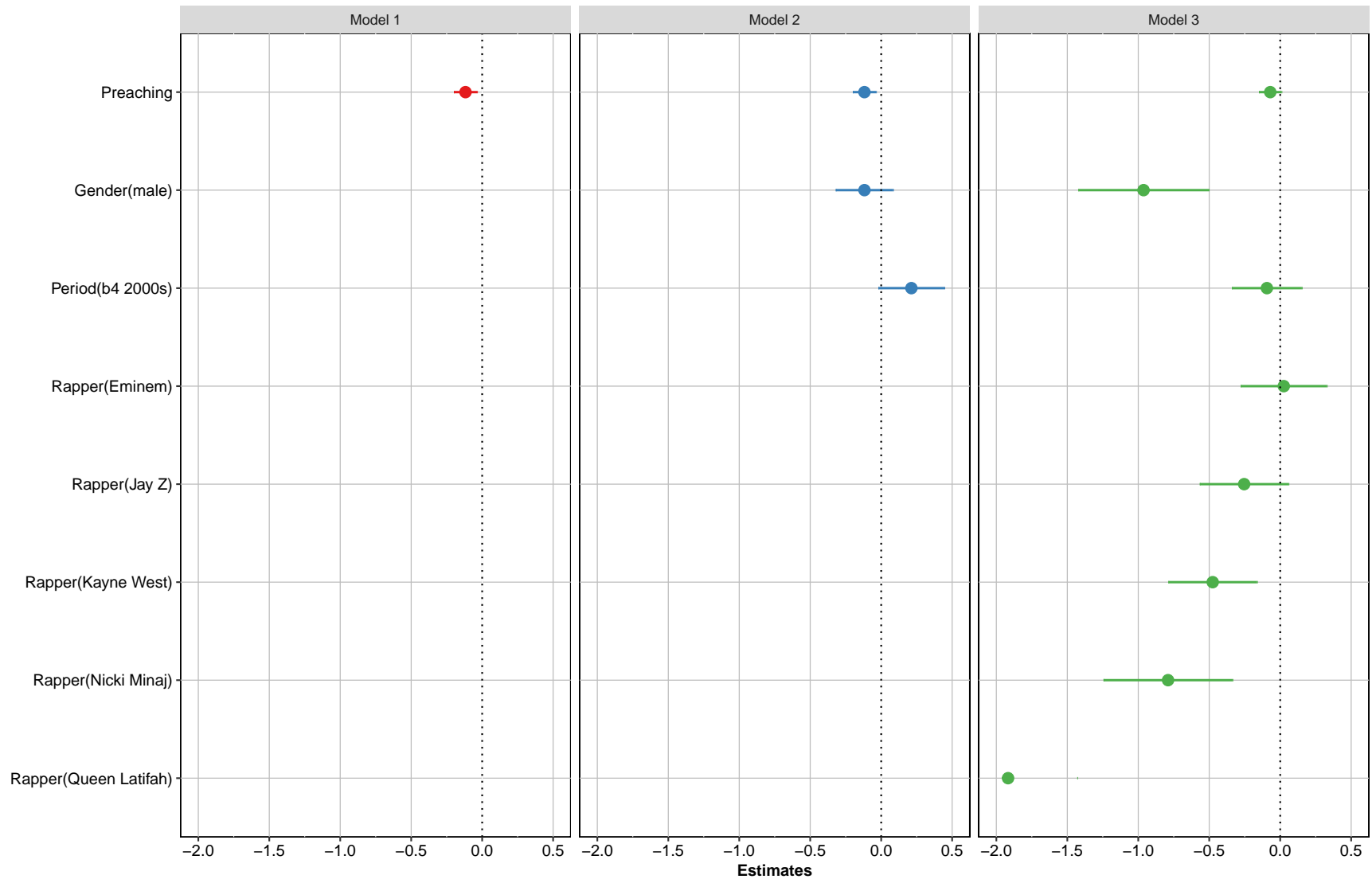
```

```

geom_hline(yintercept = 0, linetype="dotted")+
ylim(-2, 0.5)+
ggtitle("Three simple linear models predicting swearing")+
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")

```

Three simple linear models predicting swearing



Not that good of a model, right?

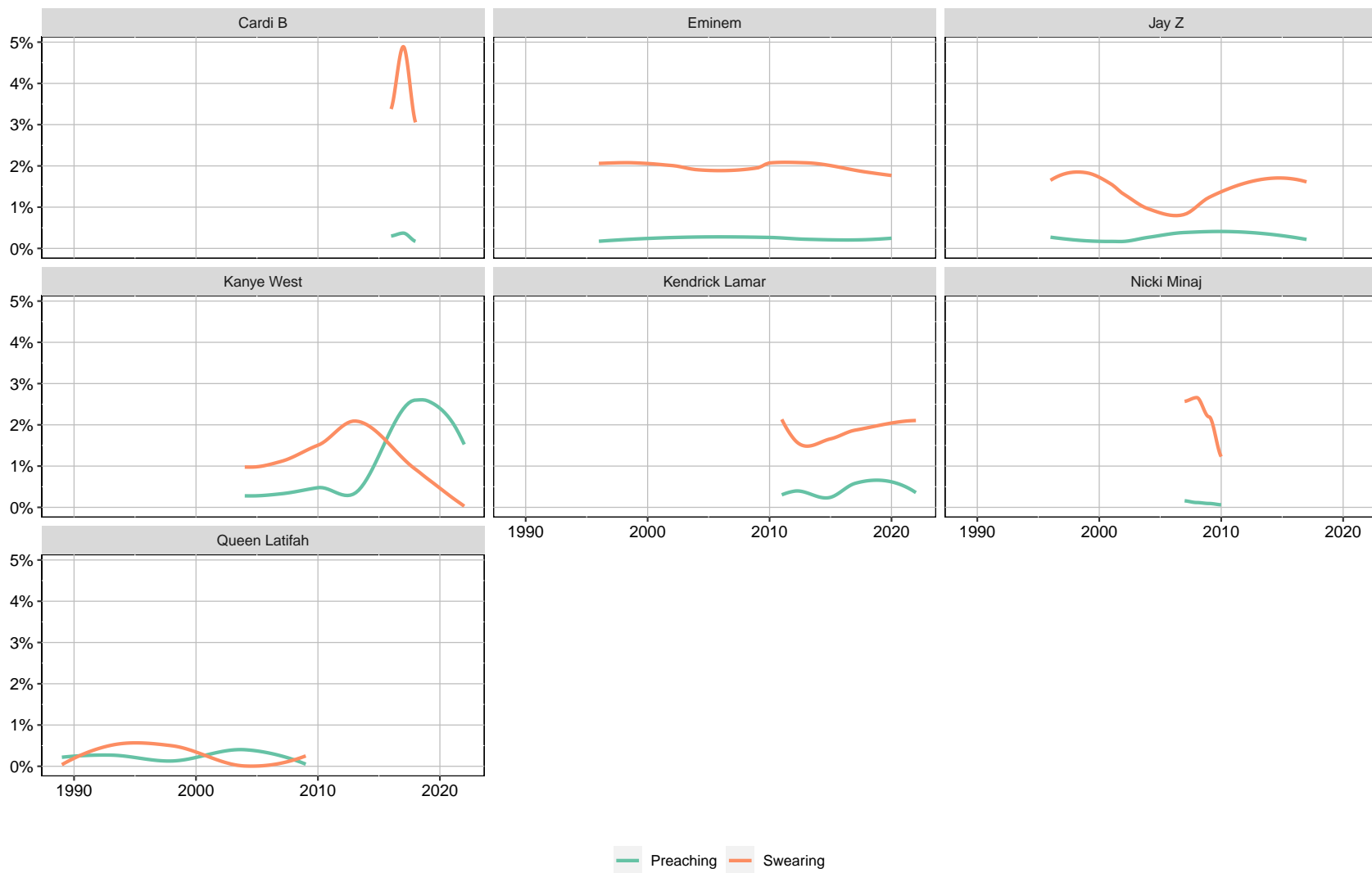
```

american_rappers %>%
  pivot_longer(9:10, names_to= "topic", values_to="normalized_count2") %>%
  ggplot(., aes(x=year, y=normalized_count2, color=topic)) +
  geom_smooth(se=FALSE) +
  scale_color_brewer(palette="Set2")+
  scale_y_continuous(labels = scales::percent_format())+
  labs(x = "", y = "",
       title = " Religion and Swearing in American Rappers' Repertoire",
       subtitle= "989 songs by seven American Rappers since 1990")+
  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)

```

Religion and Swearing in American Rappers' Repertoire

989 songs by seven American Rappers since 1990



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

The Kardashian effect

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

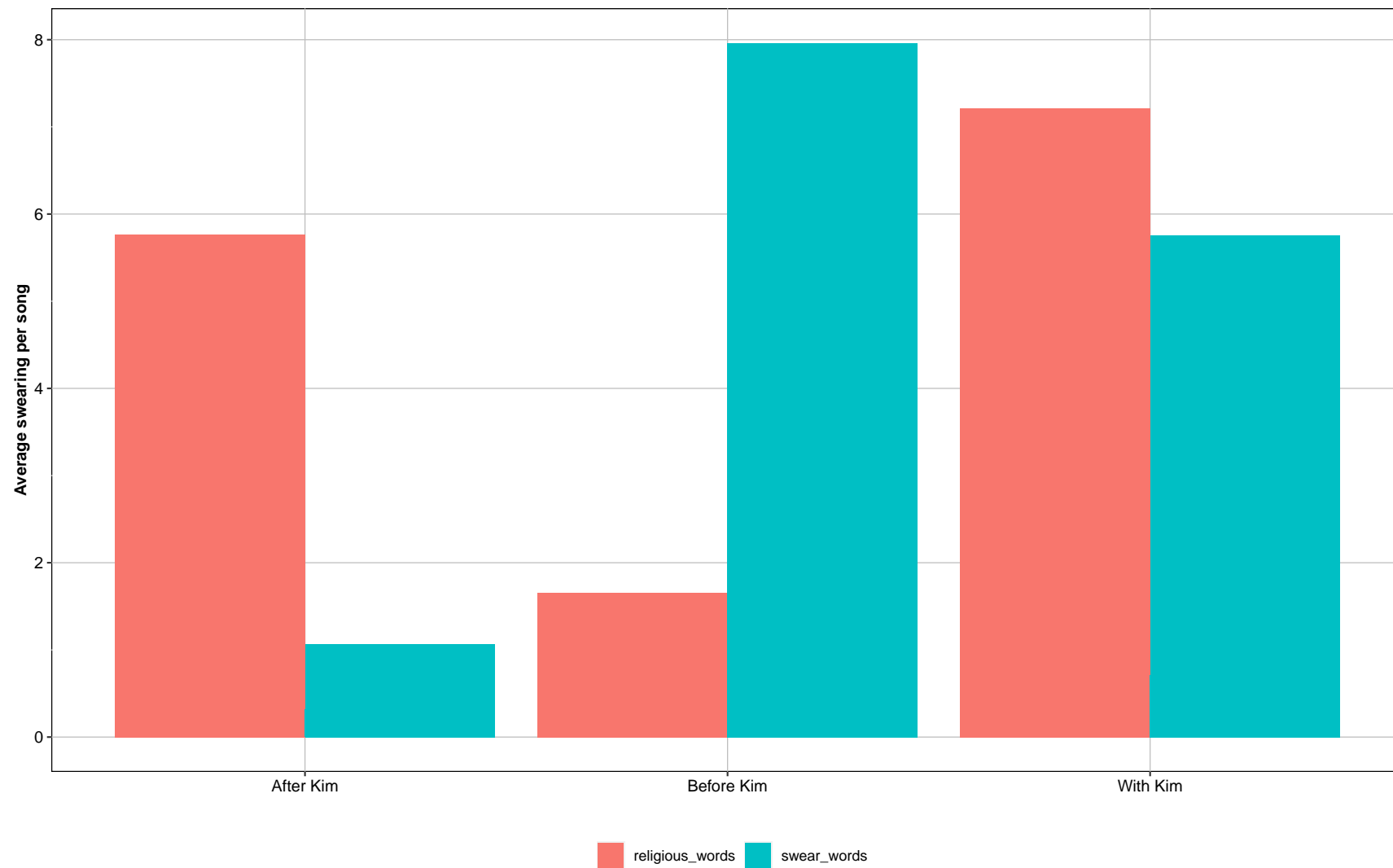
Kanye and Kim were a couple from 2011 to 2020, let's create a Kim variable!

```
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")
```

The Kardashian Effect?

214 songs in 13 albums since 2004.



```

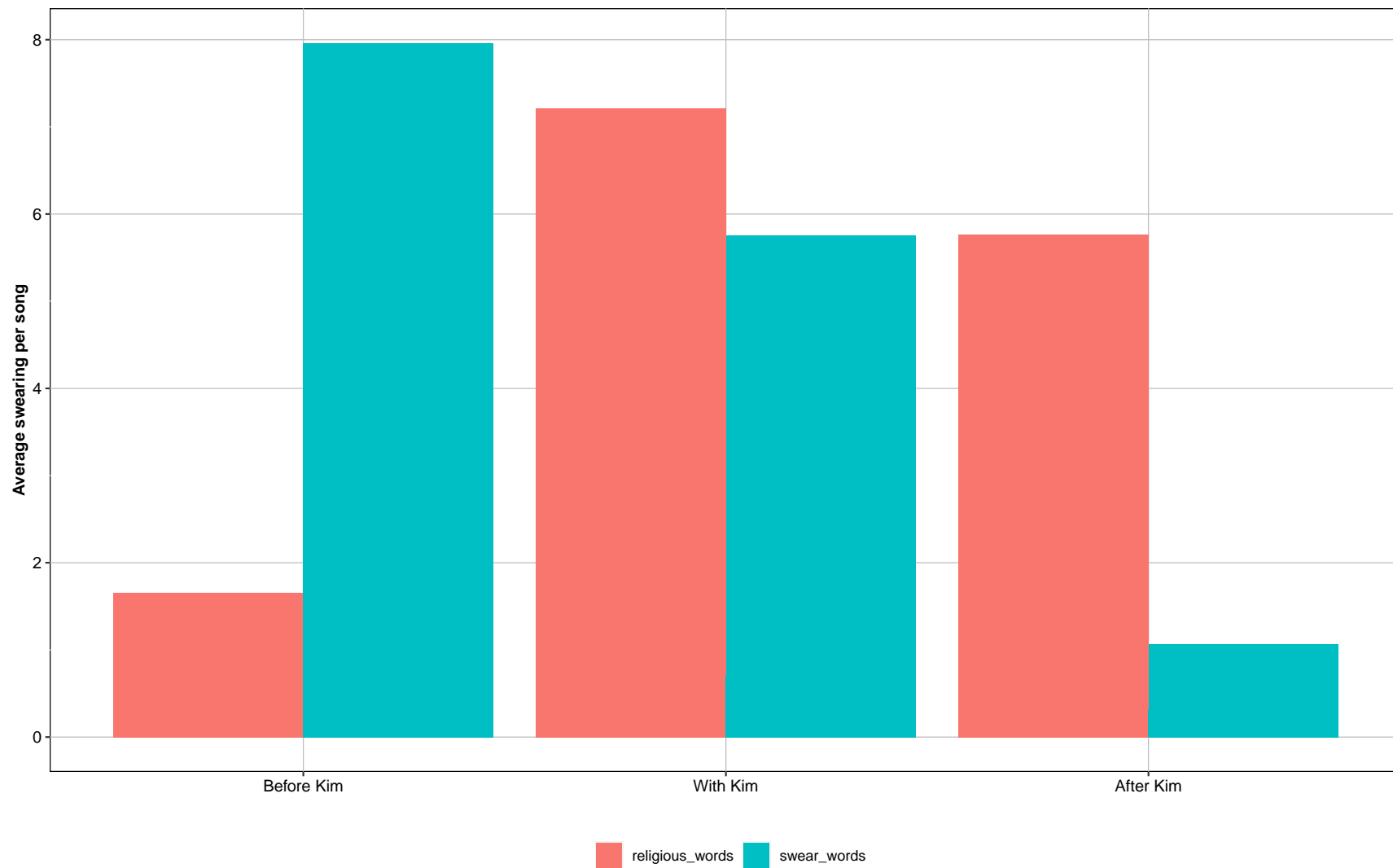
Kanye_West$kim_kardashian <- factor(Kanye_West$kim_kardashian,
                                   levels = c("Before Kim",
                                              "With Kim",
                                              "After Kim")) # reorder and run it again!!

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")

```

The Kardashian Effect?

214 songs in 13 albums since 2004.



Lamar's Repertoire in words

Let's do a basic word frequency analysis.

```
# The function "function()" creates a function;
# in this case, our function cleans the text in multiple ways,
# by applying already existing functions from tm.

cleanCorpus<-function(corpus, customStopwords){
  corpus <- tm_map(corpus, content_transformer(qdapRegex::rm_url))
  corpus <- tm_map(corpus, removeNumbers)
  corpus <- tm_map(corpus, removePunctuation)
  corpus <- tm_map(corpus, stripWhitespace)
  corpus <- tm_map(corpus, removeWords, customStopwords)
  return(corpus)
}

stops <- c(stopwords('SMART'), "dont", "youre")
# stopwords ('SMART') is a vector with common words (a, the, about...) that we want to remove from the songs

Kendrick_Lamar <- american_rappers %>%
  filter(rapper=="Kendrick Lamar") # creating a dataset only for Kendrick

# The next few lines perform a few operations to create a dataset counting appearances of words in all songs.
lamarCorpus_t <- VCorpus(VectorSource(Kendrick_Lamar$lyrics))
lamarCorpus_t <- cleanCorpus(lamarCorpus_t, stops)
LamarTDM_t <- TermDocumentMatrix(lamarCorpus_t)
lamar_dtm_t <- DocumentTermMatrix(lamarCorpus_t)

# create a document term matrix
LamarTDMm_t <- as.matrix(LamarTDM_t)
LamarSums_t <- rowSums(LamarTDMm_t)
LamarFreq_t <- data.frame(word=names(LamarSums_t), frequency=LamarSums_t)
rownames(LamarFreq_t) <- NULL
topWords_t <- subset(LamarFreq_t, LamarFreq_t$frequency >= 50)
#here we are subsetting to those that appear more than 50 times.

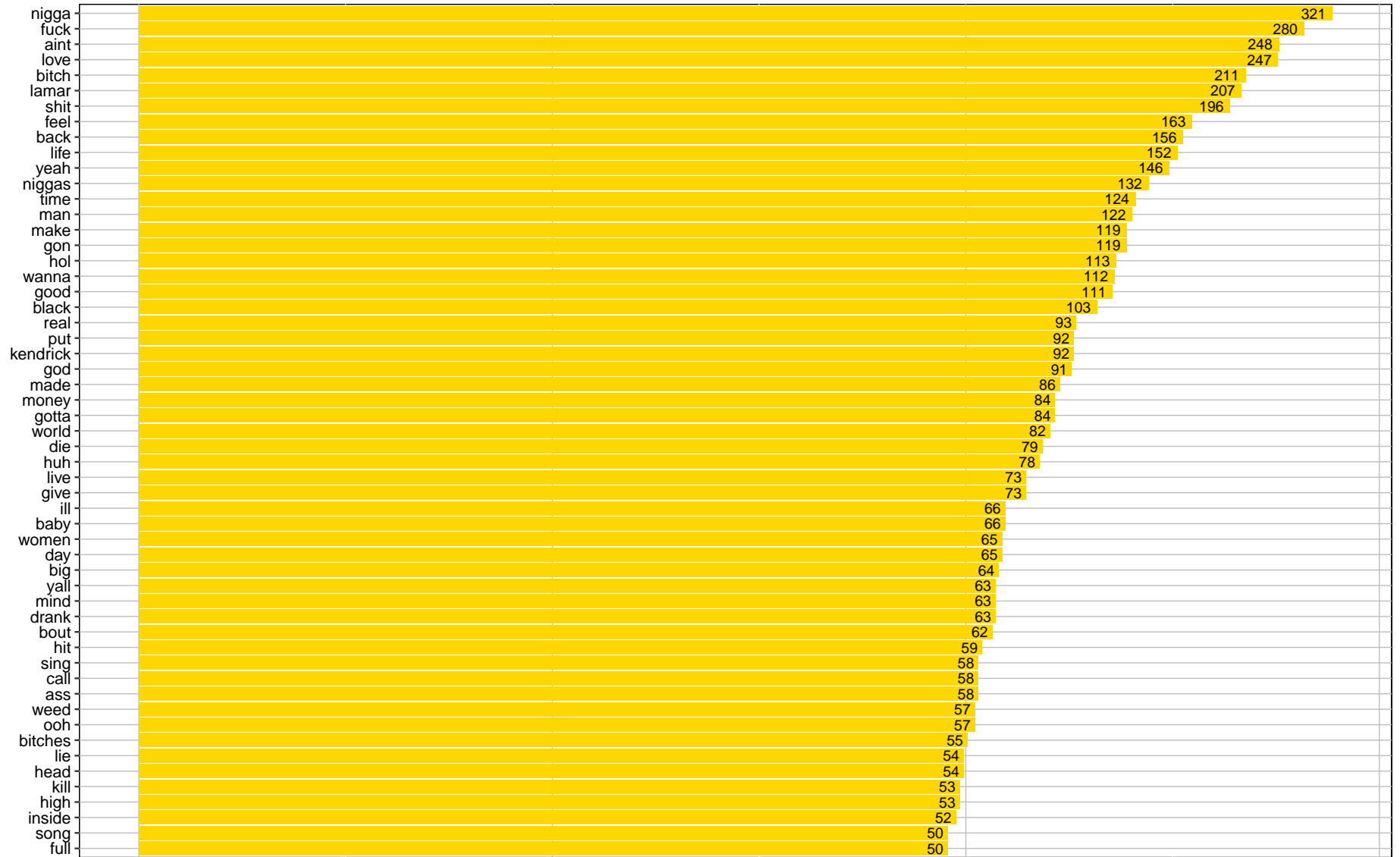
topWords_t <- topWords_t[order(topWords_t$frequency, decreasing=F),]
topWords_t$word <- factor(topWords_t$word, levels=unique(as.character(topWords_t$word)))
```

```
# getting the top words in frequency
```

```
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")
```

Most frequent words in Lamar's repertoire

All albums since 2011



Is our sample of swear/religious word bias?

Do you think this is the same for the other rappers in the sample?

Bibliography

Kastellec, Jonathan P., and Eduardo L. Leoni. 2007. "Using Graphs Instead of Tables in Political Science." *Perspectives on Politics* 5 (04).
<https://doi.org/10.1017/s1537592707072209>.