Module 4 Assignment - Sentiment Analysis

# Sentiment Analysis of Prince Lyrics

## Load the required libraries

## Read in the data

read-in the following data set in R: prince\_text.csv (the data set is available on Canvas).The Data is the result of scraping Billboard Chart information and Prince lyrics from various sites.

## Modify the dataset

## Tokenize and preprocess text

## 1. Calculate the Match Ratio between the tidy text data frame and the three lexicons bing, NRC, and Afinn(10 points)

bing<-get\_sentiments("bing")  
nrc<-get\_sentiments("nrc")  
afinn<-get\_sentiments("afinn")  
#convert the values in the afinn lexicon to positive and negative sentiments  
afinn\_neg\_pos <- afinn %>%  
 mutate( sentiment = ifelse( value >= 0, "positive",  
 ifelse( value < 0,  
 "negative", value)))  
afinn\_neg\_pos <-afinn\_neg\_pos %>%  
 select(word, sentiment)  
#Combine the three lexicons  
sentiments <-bind\_rows(list(bing=bing,nrc=nrc,afinn=afinn\_neg\_pos),.id = "lexicon")  
new\_sentiments <- sentiments %>%  
 group\_by(lexicon) %>%  
 mutate(words\_in\_lexicon = n\_distinct(word)) %>%  
 ungroup()  
  
tidy\_prince %>%  
 mutate(words\_in\_reviews = n\_distinct(word)) %>%  
 inner\_join(new\_sentiments) %>%  
 group\_by(lexicon,words\_in\_reviews, words\_in\_lexicon) %>%  
 summarise(lex\_match\_words = n\_distinct(word)) %>%  
 ungroup() %>%  
 mutate(total\_match\_words = sum(lex\_match\_words), #Not used but good to have  
 match\_ratio = lex\_match\_words / words\_in\_reviews) %>%  
 select(lexicon, lex\_match\_words, words\_in\_reviews, match\_ratio)

## Joining, by = "word"

## `summarise()` has grouped output by 'lexicon', 'words\_in\_reviews'. You can override using the `.groups` argument.

## # A tibble: 3 x 4  
## lexicon lex\_match\_words words\_in\_reviews match\_ratio  
## <chr> <int> <int> <dbl>  
## 1 afinn 772 7879 0.0980  
## 2 bing 1186 7879 0.151   
## 3 nrc 1679 7879 0.213

### “afinn” has a match ratio of 0.098

### “bing” has a match ratio of 0.151

### “nrc” has a match ratio of 0.213

## 2. Sentiment analysis (10 points)

Implement sentiment analysis using the inner join function and the “nrc” lexicon by performing an inner\_join() on the get\_sentiments() function.

prince\_nrc <- tidy\_prince %>%  
 inner\_join(get\_sentiments("nrc"))

## Joining, by = "word"

## 3. Which words contribute to the sentiment scores? (10 points)

It’s important to understand which words specifically are driving sentiment scores, and since we are using tidy data principles, it’s not too difficult to check.

Count by word and sentiment to find which words are contributing most overall to the sentiment scores. Group by sentiment. Take the top 10 words for each sentiment using top\_n(). Set up the plot using aes(), with the words on the x-axis, the number of uses n on the y-axis, and fill corresponding to sentiment. Explain the results.

### For positive sentiment, nothing beats “love”. It shows up almost 2000 times. The next highest is “cool” with just less than 250. The usages trail down to “gold”.

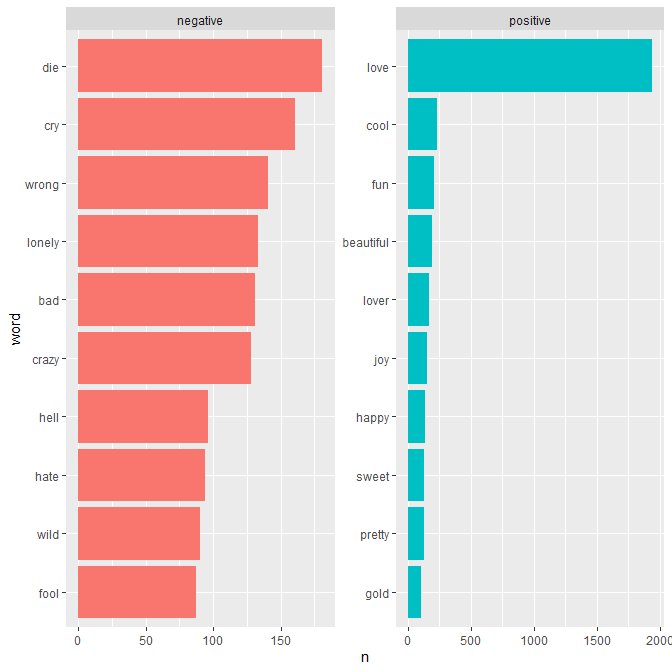
### For negative sentiment, the results are more grouped because of the lack of

### the strong sentiment like “Love”. “die” shows up more than 175 times (although the song “I Would Die 4 You” is, I think, a positive sentiment). Other top words like “crazy”, “cry”, lonely“, and”bad" usually have a negative connotation.

prince\_nrc %>%  
 # Count by word and sentiment  
 inner\_join(get\_sentiments("bing")) %>%  
 count(word,sentiment,sort = TRUE) %>%   
 # Group by sentiment  
 group\_by(sentiment)%>%  
 # Take the top 10 words for each sentimentiment)%>%  
 top\_n(10)%>%  
 ungroup() %>%  
 mutate(word = reorder(word, n)) %>%  
 # Set up the plot with aes() using the ggplot() and geom\_col(). set the graph aes so x is word and y is n and the columns are filled with sentiment.   
 ggplot(aes(word,n, fill=sentiment)) +  
 geom\_col(show.legend = FALSE) +  
 facet\_wrap(~ sentiment, ncol =5, scales = "free")+  
 coord\_flip()

## Joining, by = c("word", "sentiment")

## Selecting by n



## 4. Which song uses the most positive words? (15 points)

Make a new column called song\_total in the dataframe that tallies the total number of words from each song; the mutate() verb will make a new column and the function n() counts the number of observations in the current group:mutate(song\_total=n()).

Define a new column percent using mutate() that is n divided by song\_total, the proportion of words that belong to that sentiment. Filter only for the positive sentiment rows. Arrange by percent so you can see the results sorted by proportion of positive words.Explain the results.

### There are some very positive songs! “Jam of the Year” has a 59% positive rate with 33/56 words being positive, followed closely by “rock hard in a funky place” with 58.3%

### At first that doesn’t seem like a lot of words, but looking through the data they are not outliers. When you think about popular songs, they don’t have a lot of words and many of them are repeated in the chorus.

### After looking at the top 10 songs another filter we could run on this is to only include songs that have more than X words. Doing sentiment analysis when a song has only two words does not seem to have a lot of value.

prince\_sentiment\_song <- prince\_nrc%>%  
 # Group by song  
 group\_by(song)%>%  
 # Define a new column song\_total  
 mutate(song\_total=n()) %>%  
 # Ungroup  
 ungroup()  
  
prince\_sentiment\_song %>%  
 count(song, sentiment, song\_total) %>%  
 # Define a new column percent that is n divided by song\_total  
 mutate(percent=n/song\_total) %>%   
 # Filter only for positive words  
 filter(sentiment=="positive") %>%  
 # Arrange by percent  
 arrange(desc(percent)) %>%  
 top\_n(10)

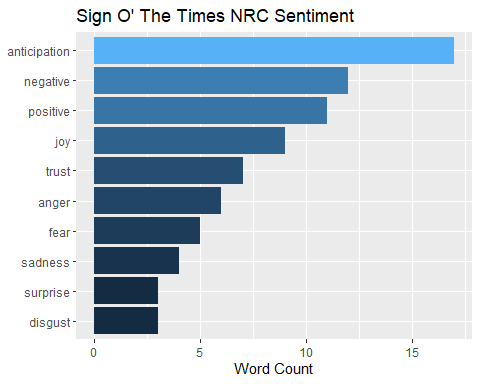
## Selecting by percent

## # A tibble: 11 x 5  
## song sentiment song\_total n percent  
## <chr> <chr> <int> <int> <dbl>  
## 1 jam of the year positive 56 33 0.589  
## 2 rock hard in a funky place positive 96 56 0.583  
## 3 the glamorous life positive 51 29 0.569  
## 4 rockhard in a funky place positive 88 47 0.534  
## 5 walk dont walk positive 29 15 0.517  
## 6 young and beautiful positive 31 16 0.516  
## 7 flutestramental positive 2 1 0.5   
## 8 in love positive 36 18 0.5   
## 9 orgasm positive 2 1 0.5   
## 10 segue vii positive 2 1 0.5   
## 11 segue x positive 2 1 0.5

## 5.Sign O’ the Times (15 points)

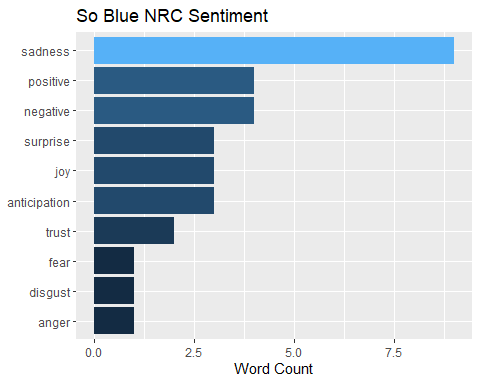
In 1987, Prince wrote a song called “Sign O’ the Times”. What is the mood of this song using the “nrc” lexicon? Visualize and explain the results. Try this for the song “so blue”. what is the mood of that song?

prince\_nrc %>%  
# Filter songs in "sign o the times"  
filter(song %in% "sign o the times") %>%  
# Group by sentiment  
 group\_by(sentiment)%>%  
 summarise(word\_count = n()) %>%  
 ungroup() %>%  
# Define a new column sentiment and reorder the sentiment based on the word\_count   
 mutate(sentiment = reorder(sentiment, word\_count)) %>%  
# Visualize the results using ggplot() and geom\_col(). set the graph aes so x is sentiment and y is word\_count and the columns are filled with word\_count  
 ggplot(aes(sentiment,word\_count, fill=word\_count)) +  
 geom\_col(show.legend = FALSE) +   
 guides(fill = FALSE) +  
 labs(x = NULL, y = "Word Count") +  
 ggtitle("Sign O' The Times NRC Sentiment") +  
 coord\_flip()



### “Sign o the times” looks like a pretty dark song. Anticipation is a sentiment that shows up the most, and other positives in the top 10 are “positive”, “joy”, “trust”, and “surprise”. Negative sentiments that show up are “negative”, “anger”, “fear”, “sadness”, and “disgust”. Not a love song!

prince\_nrc %>%  
# Filter songs in "sign o the times"  
filter(song %in% "so blue") %>%  
# Group by sentiment  
 group\_by(sentiment)%>%  
 summarise(word\_count = n()) %>%  
 ungroup() %>%  
# Define a new column sentiment and reorder the sentiment based on the word\_count   
 mutate(sentiment = reorder(sentiment, word\_count)) %>%  
# Visualize the results using ggplot() and geom\_col(). set the graph aes so x is sentiment and y is word\_count and the columns are filled with word\_count  
 ggplot(aes(sentiment,word\_count, fill=word\_count)) +  
 geom\_col(show.legend = FALSE) +   
 guides(fill = FALSE) +  
 labs(x = NULL, y = "Word Count") +  
 ggtitle("So Blue NRC Sentiment") +  
 coord\_flip()



### With a song like “So Blue”, it’s not a surprise that the top sentiment is “sadness” with 8 word counts. Sentiments “positive” and “negative” and tied with 4 word counts. Most of the other words look pretty negative.

## 6. Polarity by chart level (15 points)

Break down your analysis to the chart level using the Bing lexicon. Create a graph of the polar sentiment per chart level. Use spread() to separate the sentiments into columns and mutate() to create a polarity (positive - negative) field and a percent\_positive field (positive/totalsentiment∗100), for a different perspective.

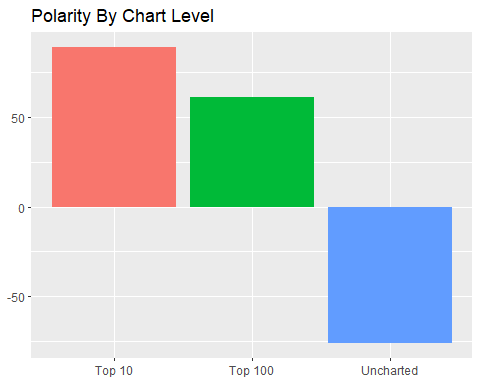
#Implement sentiment analysis using the inner join function and the "bing" lexicon  
prince\_bing <- tidy\_prince %>%  
 inner\_join(get\_sentiments("bing"))

## Joining, by = "word"

prince\_polarity\_chart <- prince\_bing %>%  
#Count sentiment by chart\_level  
 count(chart\_level, sentiment) %>%  
  
#Use spread() to separate the sentiments into columns  
 spread(sentiment, n, fill = 0) %>%  
   
#Use mutate() to create a polarity (positive - negative) field and a percent\_positive field (positive / (positive + negative) \* 100)  
 mutate(polarity = positive - negative) %>%  
 mutate(percent\_positive = positive / (positive + negative) \* 100)

Visualize the results.

prince\_polarity\_chart %>%  
 ggplot(aes(chart\_level, polarity, fill=chart\_level)) +  
 geom\_col(show.legend = FALSE) +   
 xlab(NULL) +   
 ylab(NULL) +  
 ggtitle("Polarity By Chart Level")



Does this say that charted songs are typically more positive than negative? If so, what does this tell you about what society wants to hear? Can you even make these assumptions? Note that the Bing lexicon itself has more negative than positive words.

### From the data we’ve seen, it looks like Prince’s charted songs are more positive than the uncharted songs. From an artist’s perspective, I think positive songs are more fun to write and perform. People don’t want to party to negative songs. Perhaps the uncharted songs are more likely to be “filler songs” on an album? Positive songs are the money makers, so that’s what the artist is going to spend time on.

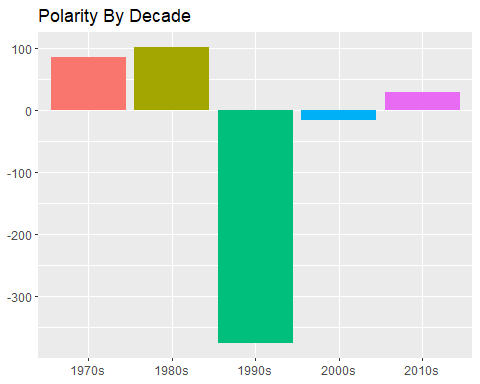
## 7. Polarity by decades (15 points)

Break down your analysis to the decades using the Bing lexicon. Create a graph of the polar sentiment per decade. Use spread() to separate the sentiments into columns and mutate() to create a polarity (positive - negative) field and a percent\_positive field (positive/totalsentiment∗100), for a different perspective.

prince\_polarity\_decade <- prince\_bing %>%  
 #Count sentiment by decade  
 count(decade, sentiment) %>%  
 filter(!decade == "NA" )%>%  
 spread(sentiment, n, fill = 0) %>%  
 mutate(polarity = positive - negative) %>%  
 mutate(percent\_positive = positive / (positive + negative) \* 100)  
   
#Use spread() to separate the sentiments into columns  
#Use mutate() to create a polarity (positive - negative) field and a percent\_positive field (positive/totalsentiment∗100)

Visualize and explain the results.

prince\_polarity\_decade %>%  
# Visualize the results using ggplot() and geom\_col(). set the graph aes so x is decade and y is polarity and the columns are filled with decade  
 ggplot(aes(decade, polarity, fill=decade)) +  
 geom\_col(show.legend = FALSE) +   
 xlab(NULL) + ylab(NULL) +  
 ggtitle("Polarity By Decade")



### It looks like Prince was very positive in the 1970’s and 1980’s and then hit a very dark time in the 1990’s. In the 2000’s and 2010’s he was positive but not as much as he was early in his career.

A list of Prince’s life events is attached, collected from popular sources such as Rolling Stone Magazine, Biography.com, etc. Compare Princ’s life events with the sentiment.

### I can remember when his songs came out in the early 80’s. The song “Little Red Corvette” was the first song I heard on a Walkman. Even though I wasn’t a fan of his music, it sounded great. He was so popular and living the dream. Like most artists with a meteoric rise, it couldn’t last forever.

### When he started doing stuff like changing his name to a symbol you could tell something was up. From this life events list, it looks like the 90s were not good for him. I wish the events would include when his Top 10 hits charted.

## 8. Reflect on this assignment (10 points)

### a. What have you learned from this assignment?

### R has three great ways to evaluate sentiment, and they can all be used in a variety of ways. It’s certainly not definitive, but it is another tool in our analytical tool belts.

### b. What else do you want to know about the data?

### I think there’s too much emphasis being put on uncharted songs. How many of these uncharted songs were even on an album? 374 of them were not (sum(is.na(prince$album)) . I think if there weren’t important enough to put on an album then we shouln’t be analyzing them. They were just an artist’s work that wasn’t good enough to publish. We’ve all done things that we weren’t please with an just put aside. I would love to see these analyzed separately.