Lyrics Analysis: What genre are popular and what emotions they try to express from the songs?

Project Summary: A song can a mean to express emotions and thoughts of our times. Analyzing lyrics may provide insights on what people of those times want to tell. A filtered corpus of 100,000+ song lyrics from MetroLyrics is used for this analysis.

Let's launch all necessary packages

```
library(dplyr); library(ggplot2); library(ggthemes);library(dplyr); library(tidytext)
```

```
##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

Let's load the original dataset "dt_lyrics". You need to change this directory to import from your source.

```
load("/Users/sol/Downloads/processed_lyrics.RData")
```

Let's create a new dataset "lyrics_df"" for analysis while keeping original dataset "dt_lyrics" intact.

```
lyrics_df = dt_lyrics
```

Let's review the structure of "lyrics_df". It appears that there are few songs that have abnormal year or very few count in the year.

```
table(lyrics_df$year)
```

```
##
##   112   702  1968  1970  1971  1972  1973  1974  1975  1976  1977  1978  1979
##     1     1     1   112   125   119   172   120    82    47   205   137   105
##  1980  1981  1982  1983  1984  1985  1986  1987  1988  1989  1990  1991  1992
##   141    99   134    85   129   123   128    71   133   137   809   105   483
##  1993  1994  1995  1996  1997  1998  1999  2000  2001  2002  2003  2004  2005
##   356   374   409   484   424   441   486   672   738   880  1199  1496  2793
##  2006  2007  2008  2009  2010  2011  2012  2013  2014  2015  2016
## 42457 30600  8220  4094  4187  3340  3321  3450  4880  3286  3313
```

Let's remove theses outlier years for more accurate analysis.

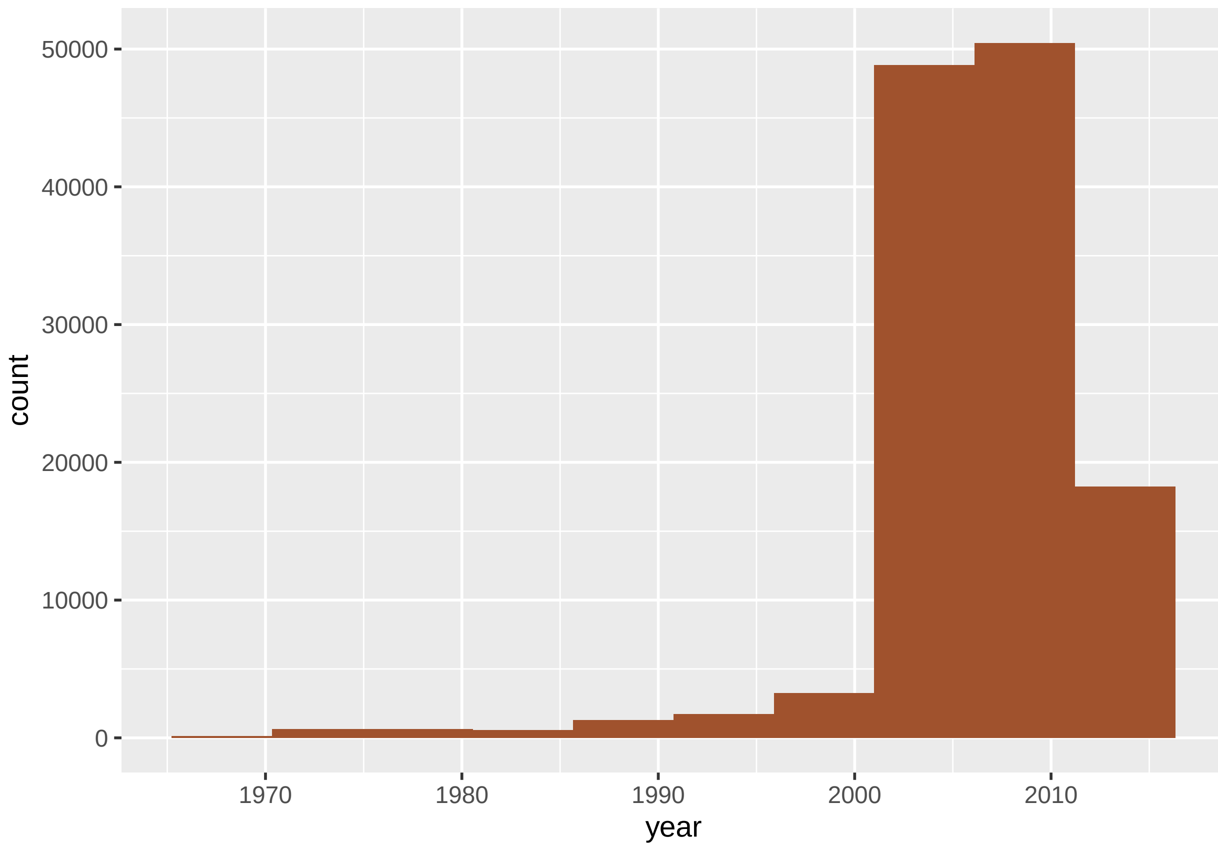
```
lyrics_df = subset(dt_lyrics, year!= 112 & year!= 702 & year!=1968)
table(lyrics_df$year)
```

```
##
##  1970  1971  1972  1973  1974  1975  1976  1977  1978  1979  1980  1981  1982
##   112   125   119   172   120    82    47   205   137   105   141    99   134
##  1983  1984  1985  1986  1987  1988  1989  1990  1991  1992  1993  1994  1995
```

```
##     85    129    123    128     71    133    137    809    105    483    356    374    409
##   1996   1997   1998   1999   2000   2001   2002   2003   2004   2005   2006   2007   2008
##    484    424    441    486    672    738    880   1199   1496   2793  42457  30600   8220
##   2009   2010   2011   2012   2013   2014   2015   2016
##   4094   4187   3340   3321   3450   4880   3286   3313
```

Let's run a histogram of all songs count by year as an exploratory analysis.

```r
ggplot(data=lyrics_df,aes(x=year))+
  geom_histogram(fill='sienna', bins=10)
```

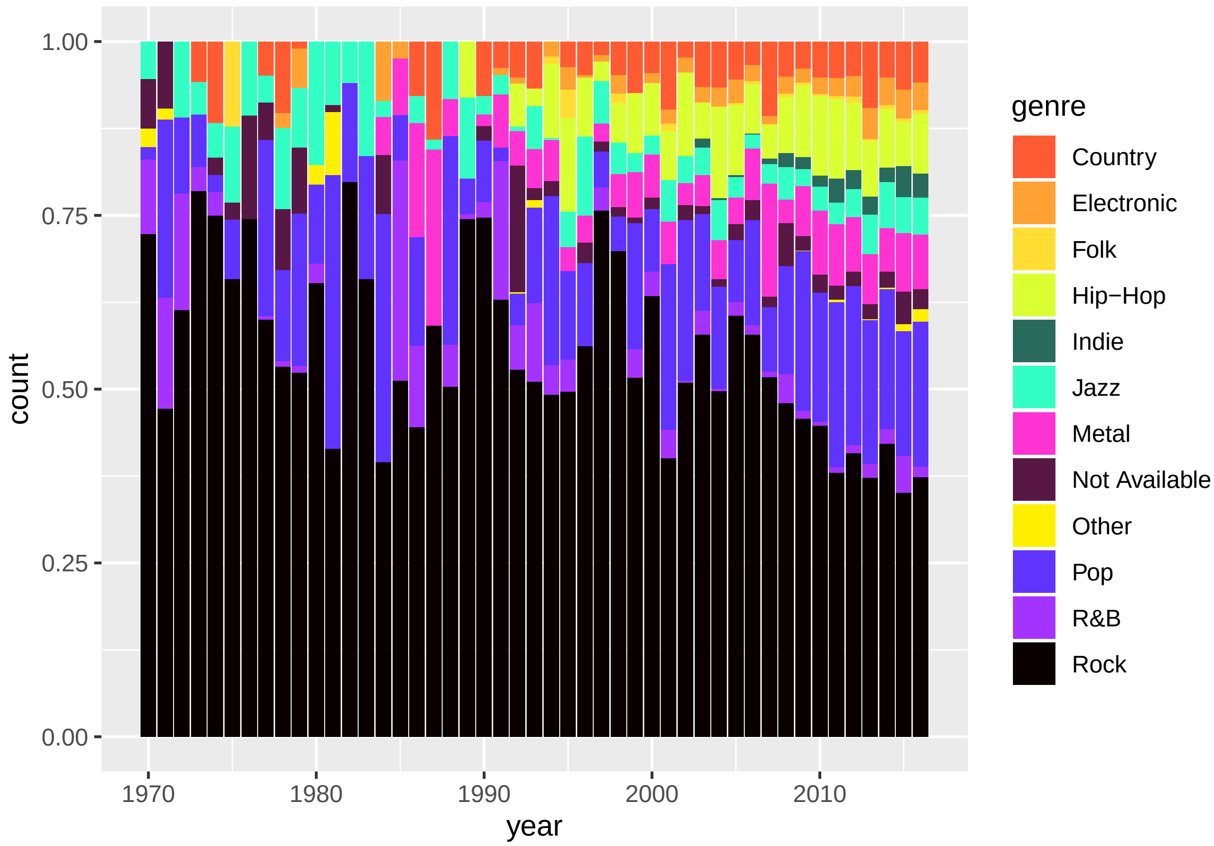


Analyzing how the genres make up the total number of songs in each year, the analysis provides following findings.

1) In average, the three most number of songs across all years in the analysis seem to be Rock, Pop, and R&B. This could imply that Americans' all-time favorite genre are Rock, Pop, and R&B.

2) Hip-Hop started gained ground in 1992 and became a solid favorite genre since then.

```r
p <- ggplot(data = lyrics_df,
            mapping = aes(x = year, fill = genre))
p + geom_bar(position = "fill") + scale_fill_manual(values=c("#FF5B33", "#FFA233", "#FFDD33", "#DAFF33"
```



Let's launch Bing lexicon. It enables sentiment analysis.

```
as.data.frame(get_sentiments('bing'))[1:50,]
```

```
##              word sentiment
## 1         2-faces  negative
## 2        abnormal  negative
## 3         abolish  negative
## 4      abominable  negative
## 5      abominably  negative
## 6       abominate  negative
## 7     abomination  negative
## 8           abort  negative
## 9         aborted  negative
## 10         aborts  negative
## 11         abound  positive
## 12        abounds  positive
## 13         abrade  negative
## 14       abrasive  negative
## 15         abrupt  negative
## 16       abruptly  negative
## 17        abscond  negative
## 18        absence  negative
## 19  absent-minded  negative
## 20       absentee  negative
## 21         absurd  negative
## 22      absurdity  negative
## 23       absurdly  negative
```

```
## 24      absurdness  negative
## 25       abundance  positive
## 26        abundant  positive
## 27           abuse  negative
## 28          abused  negative
## 29          abuses  negative
## 30         abusive  negative
## 31         abysmal  negative
## 32       abysmally  negative
## 33           abyss  negative
## 34      accessable  positive
## 35      accessible  positive
## 36      accidental  negative
## 37         acclaim  positive
## 38       acclaimed  positive
## 39     acclamation  positive
## 40        accolade  positive
## 41       accolades  positive
## 42   accommodative  positive
## 43    accomodative  positive
## 44      accomplish  positive
## 45    accomplished  positive
## 46  accomplishment  positive
## 47 accomplishments  positive
## 48          accost  negative
## 49        accurate  positive
## 50      accurately  positive
```

```r
get_sentiments('bing')%>%
  group_by(sentiment)%>%
  count()
```

```
## # A tibble: 2 x 2
## # Groups:   sentiment [2]
##   sentiment     n
##   <chr>     <int>
## 1 negative   4781
## 2 positive   2005
```

Let's match the words in the Bing dictionary with the ones in the stemmedwords to identify sentiments in each song.

```r
lyrics_df%>%
  group_by(id)%>%
  unnest_tokens(output = word, input = stemmedwords)%>%
  inner_join(get_sentiments('bing'))%>%
  group_by(sentiment)
```

```
## Joining, by = "word"
```

```
## # A tibble: 1,968,110 x 8
## # Groups:   sentiment [2]
##    song        year artist genre  lyrics                          id word  sentiment
##    <chr>      <dbl> <chr>  <chr>  <chr>                        <int> <chr> <chr>
##  1 when-you-~  2009 a      Hip-H~ "I stopped by the house~     1 fast  positive
##  2 when-you-~  2009 a      Hip-H~ "I stopped by the house~     1 love  positive
```

```
##   3 when-you--  2009 a      Hip-H~ "I stopped by the house~     1 cry   negative
##   4 when-you--  2009 a      Hip-H~ "I stopped by the house~     1 wrong negative
##   5 careless--~  2009 a      Hip-H~ "I feel so unsure\nAs I~     2 unsu~ negative
##   6 careless--~  2009 a      Hip-H~ "I feel so unsure\nAs I~     2 lead  positive
##   7 careless--~  2009 a      Hip-H~ "I feel so unsure\nAs I~     2 die   negative
##   8 careless--~  2009 a      Hip-H~ "I feel so unsure\nAs I~     2 sad   negative
##   9 careless--~  2009 a      Hip-H~ "I feel so unsure\nAs I~     2 guil~ negative
##  10 careless--~  2009 a      Hip-H~ "I feel so unsure\nAs I~     2 easy  positive
## # ... with 1,968,100 more rows
```

Analyzing Positive and Negative Words in stemmedwords, it seems there are more negative sentiments than positive sentiments in songs.

```
lyrics_df%>%
  group_by(id)%>%
  unnest_tokens(output = word, input = stemmedwords)%>%
  inner_join(get_sentiments('bing'))%>%
  group_by(sentiment)%>%
  count()
```

```
## Joining, by = "word"
```

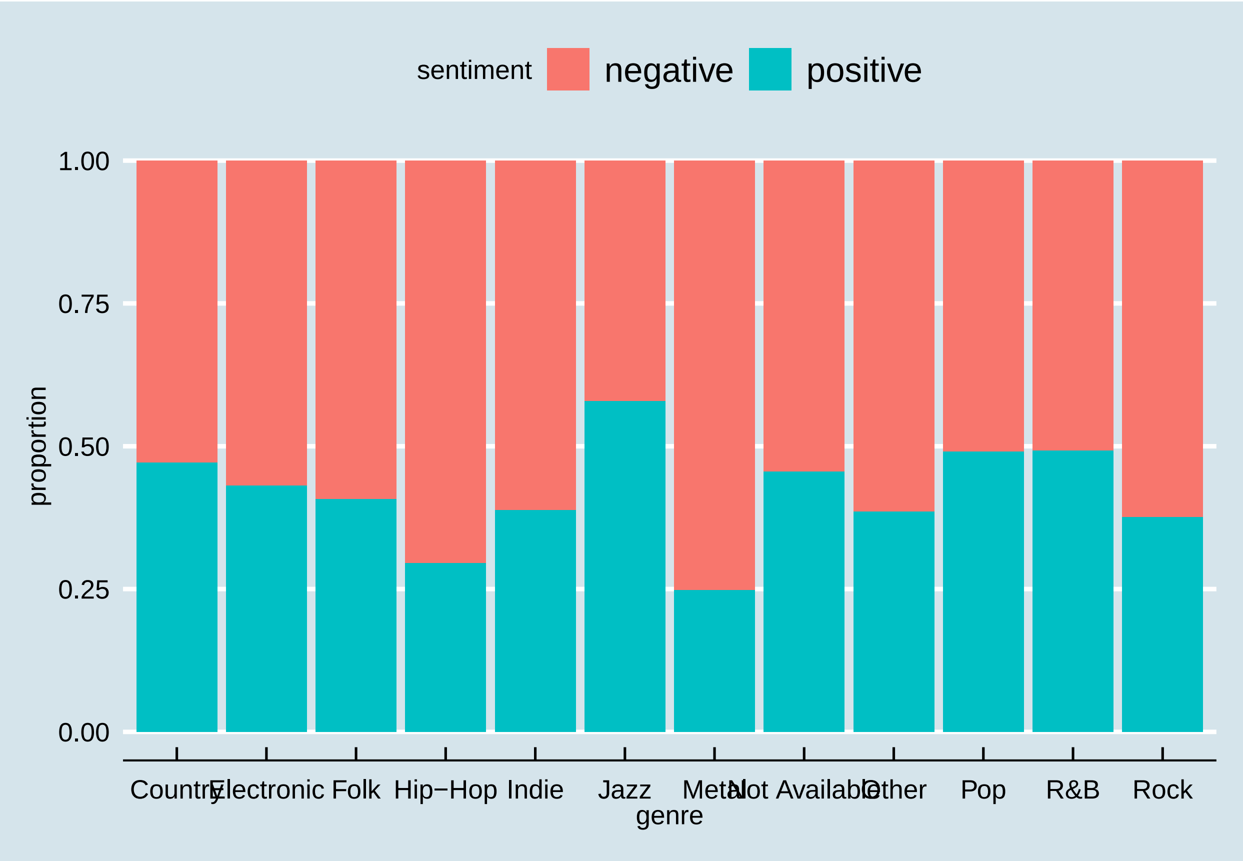
```
## # A tibble: 2 x 2
## # Groups:   sentiment [2]
##   sentiment       n
##   <chr>       <int>
## 1 negative  1215562
## 2 positive   752548
```

Breaking the sentiment analysis by each genre across all years, following findings are gained.

1) Metal and Hip-Hop have higher negative sentiment than positive sentiment compared to other genres
2) Only Jazz has distinguishable positive sentiment than negative sentiment

```
lyrics_df %>%
  select(id,stemmedwords,genre)%>%
  group_by(id)%>%
  unnest_tokens(output=word,input=stemmedwords)%>%
  ungroup()%>%
  inner_join(get_sentiments('bing'))%>%
  group_by(genre,sentiment)%>%
  summarize(n = n())%>%
  mutate(proportion = n/sum(n))%>%
  ggplot(aes(x=genre,y=proportion,fill=sentiment))+geom_col()+theme_economist()
```

```
## Joining, by = "word"
```



To analyze the sentiment deeper, nrc lexicon is applied.

```
library(remotes)
install_github("EmilHvitfeldt/textdata")
```

```
## Skipping install of 'textdata' from a github remote, the SHA1 (2b5e9f7b) has not changed since last i
##   Use `force = TRUE` to force installation
```

```
install_github("juliasilge/tidytext")
```

```
## Skipping install of 'tidytext' from a github remote, the SHA1 (65bc08cd) has not changed since last i
##   Use `force = TRUE` to force installation
```

```
library(tidytext)

get_sentiments('nrc')%>%
  group_by(sentiment)%>%
  count()
```

```
## # A tibble: 10 x 2
## # Groups:   sentiment [10]
##    sentiment        n
##    <chr>        <int>
##  1 anger         1247
##  2 anticipation   839
##  3 disgust       1058
##  4 fear          1476
##  5 joy            689
##  6 negative      3324
```

```
##  7 positive      2312
##  8 sadness       1191
##  9 surprise       534
## 10 trust         1231
```

```
table(get_sentiments('nrc')$sentiment)
```

```
##
##        anger anticipation      disgust         fear          joy     negative
##         1247          839         1058         1476          689         3324
##     positive      sadness     surprise        trust
##         2312         1191          534         1231
```

Let's classify stemmedwords across all songs into emotions provided by NRC.

```
lyrics_df%>%
  group_by(id)%>%
  unnest_tokens(output = word, input = stemmedwords)%>%
  inner_join(get_sentiments('nrc'))%>%
  group_by(sentiment)%>%
  count()
```

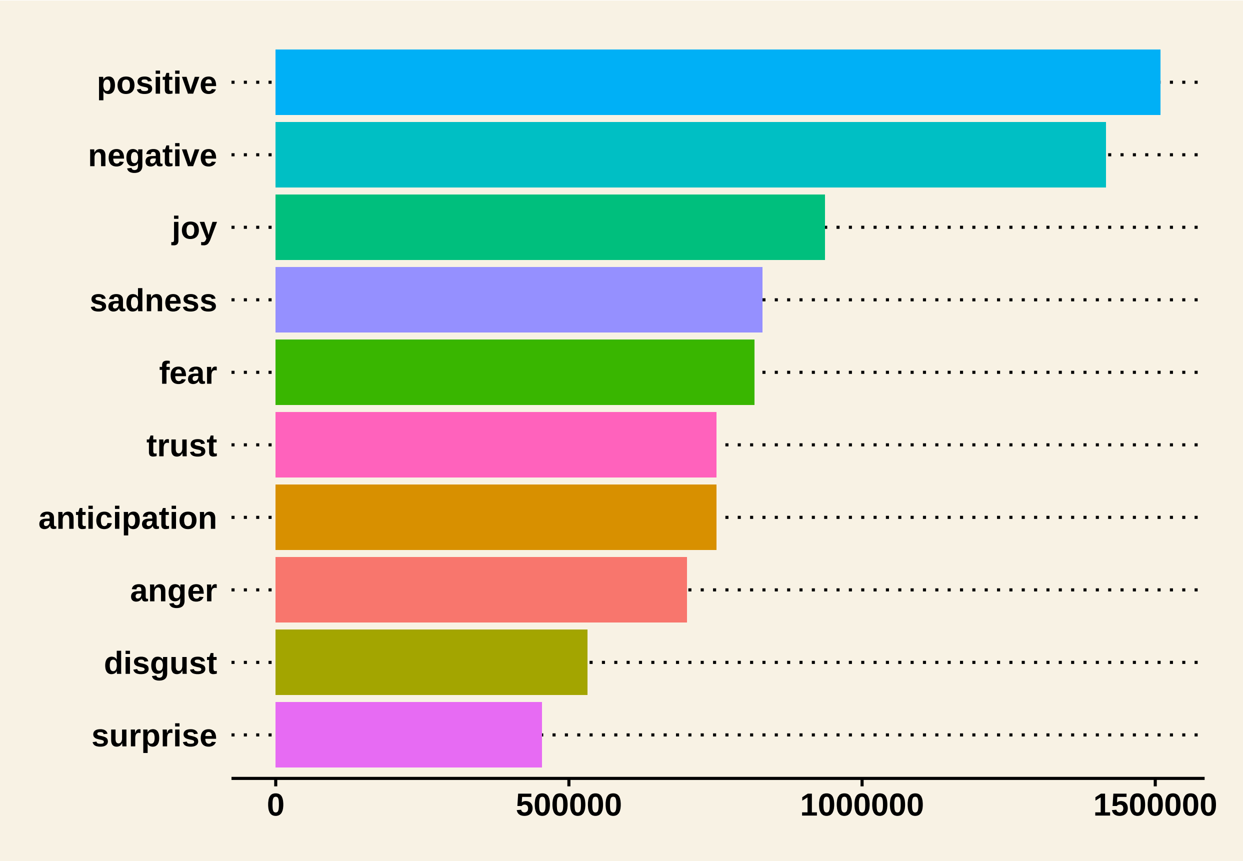
```
## Joining, by = "word"
```

```
## # A tibble: 10 x 2
## # Groups:   sentiment [10]
##    sentiment          n
##    <chr>          <int>
##  1 anger         701479
##  2 anticipation  751685
##  3 disgust       531720
##  4 fear          816644
##  5 joy           936530
##  6 negative     1416050
##  7 positive     1508737
##  8 sadness       830179
##  9 surprise      453953
## 10 trust         751966
```

Visualing the above analysis, joy and sadness make up the two most emotions excluding postive and negative.

```
lyrics_df%>%
  group_by(id)%>%
  unnest_tokens(output = word, input = stemmedwords)%>%
  inner_join(get_sentiments('nrc'))%>%
  group_by(sentiment)%>%
  count()%>%
  ggplot(aes(x=reorder(sentiment,X = n),y=n,fill=sentiment))+geom_col()+guides(fill=F)+coord_flip()+ther
```

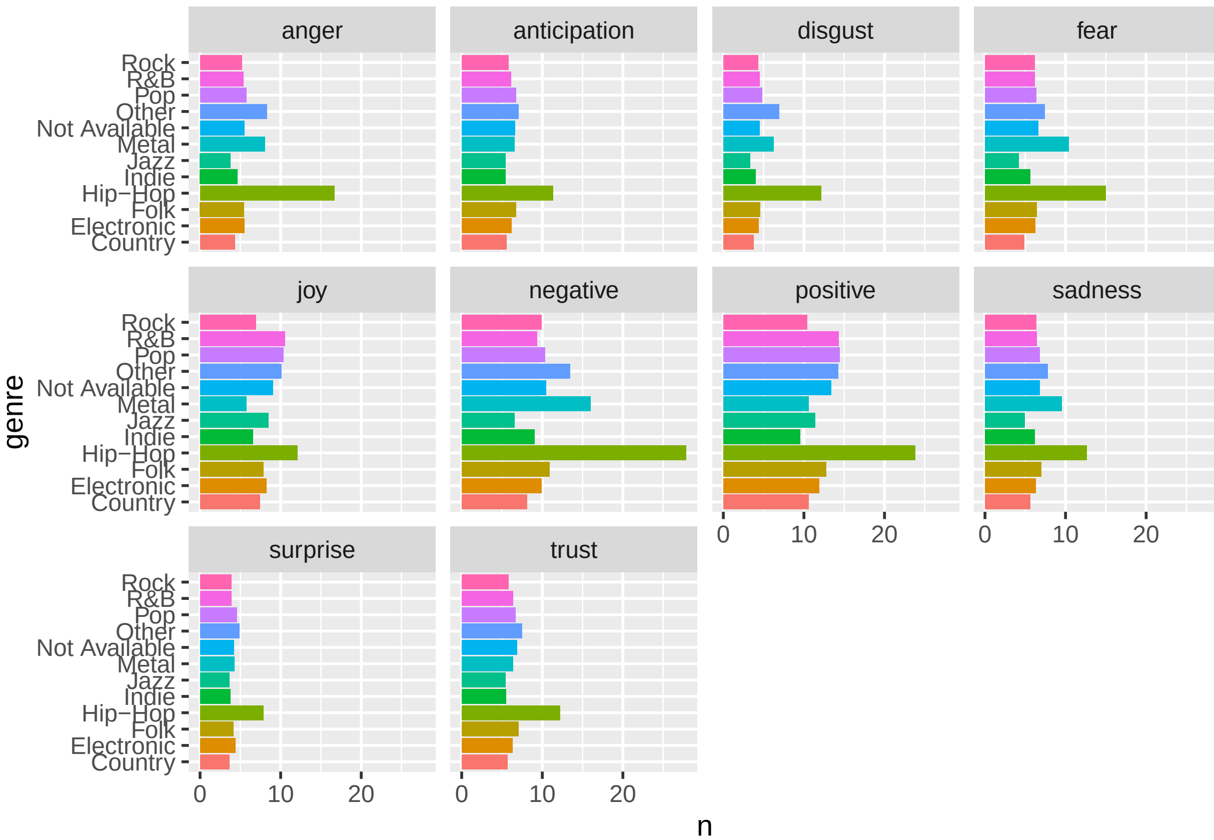
```
## Joining, by = "word"
```



Visualizing which genre stands out in terms of each emotion. It seems that Hip-Hop stands out the most out of all emotions that are negative (anger, disgust, fear, negative, sadness)

```
lyrics_df%>%
  group_by(id)%>%
  unnest_tokens(output = word, input = stemmedwords)%>%
  inner_join(get_sentiments('nrc'))%>%
  group_by(id,sentiment,genre)%>%
  count()%>%
  group_by(sentiment, genre)%>%
  summarize(n = mean(n))%>%
  ungroup()%>%
  ggplot(aes(x=genre,y=n,fill=genre))+
  geom_col()+
  facet_wrap(~sentiment)+
  guides(fill=F)+coord_flip()
```

```
## Joining, by = "word"
```



Visualizing it to analyze emotions by genre, findings are;

1) In general, Hip-Hop songs focus on expressing negative emotions in their lyrics.
2) Next to Hip-Hop, Metal songs also tend to express negative emotions in their lyrics.
3) all other genres show similar patterns of express emotions, with positive sentiment slightly higher than negative sentiment
4) Jazz stood out among all genres by expressing distintive positive sentiment.

```
lyrics_df%>%
  group_by(id)%>%
  unnest_tokens(output = word, input = stemmedwords)%>%
  inner_join(get_sentiments('nrc'))%>%
  group_by(id,sentiment,genre)%>%
  count()%>%
  group_by(sentiment, genre)%>%
  summarize(n = mean(n))%>%
  ungroup()%>%
  ggplot(aes(x=sentiment,y=n,fill=genre))+
  geom_col()+
  facet_wrap(~genre)+
  guides(fill=F)+coord_flip()
```

```
## Joining, by = "word"
```

—- SUMMARY of the project —

Using the dataset retrieved from MetroLyrics, more than 100,000+ songs were analyzed to learn about emotions they aimed to express through the lyrics. While all other genres showed similar patterns of expressing positive emotions slightly higher than negative emotions, Hip-Hop and Metal expressed mostly negative emotions in their lyrics. Considering the fact the Hip-Hop particulary gained its ground starting 1992, we can imply that Hip-Hop became the channel for artists to express their negative emotions about their times. Rock, which is generally the most listened genre across all years, expressed balanced positive and negative emotions. Jazz is the only genre that express more positive emotions than negative emotions distinctively. It implies that people tend to listen to Jazz when they feel happy or want to feel happy.

— Code Citation —

Lecture from Columbia University SPS - APPLIED ANALYTICS FRAMEWORKS & METHDS II - sentimentAnalysis-1.html

Please refer to file "sentimentAnalysis-1.html" in the "lib" directory.