proj1-ty2422

Irene

2/5/2020

This is an R Markdown Notebook. When you execute code within the notebook, the results appear beneath the code.

Difference among different genre song lyrics

Step 0 - Install and load libraries

```
library(tm)
```

```
## Loading required package: NLP
```

```
library(wordcloud)
```

```
## Loading required package: RColorBrewer
```

```
library(RColorBrewer)
library(dplyr)
```

```
##
## 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
```

```
library(tidytext)
library(ggplot2)
```

```
##
## Attaching package: 'ggplot2'
```

```
## The following object is masked from 'package:NLP':
##
##     annotate
```

```
library(gridExtra)
```

```
##
## Attaching package: 'gridExtra'
```

```
## The following object is masked from 'package:dplyr':
##
##     combine
```

Step 1 - Read in the song lyrics

```
load("~/Desktop/Spring2020-Project1-Irene98-master/output/processed_lyrics.RData")
```

Step 2 - Text processing

```
myCorpus <- Corpus(VectorSource(dt_lyrics$stemmedwords))
tdm <- TermDocumentMatrix(myCorpus)
tdm.tidy=tidy(tdm)
tail(tdm.tidy,50)
```

```
## # A tibble: 50 x 3
##    term        document count
##    <chr>       <chr>    <dbl>
##  1 built       125704       1
##  2 caught      125704       1
##  3 cave        125704       1
##  4 chain       125704       2
##  5 change      125704       2
##  6 chest       125704       1
##  7 city        125704       1
##  8 clock       125704       2
##  9 consequence 125704       1
## 10 count       125704       2
## # ... with 40 more rows
```

```
tdm.overall=summarise(group_by(tdm.tidy, term), sum = sum(count))
```

Step 3 - Inspect an overall wordcloud

```
set.seed(1234)
wordcloud(words = tdm.overall$term, freq = tdm.overall$sum, min.freq = 1,c(5,0.5),
          max.words=200, random.order=FALSE, rot.per=0.35,
          colors=brewer.pal(8, "Dark2"))
```

Love is the most frequent word in the song lyrcs,and time,youre,baby are also frequently used in song lyrics.

Step 4 - Classification based on genre

```
unique(dt_lyrics$genre)
```

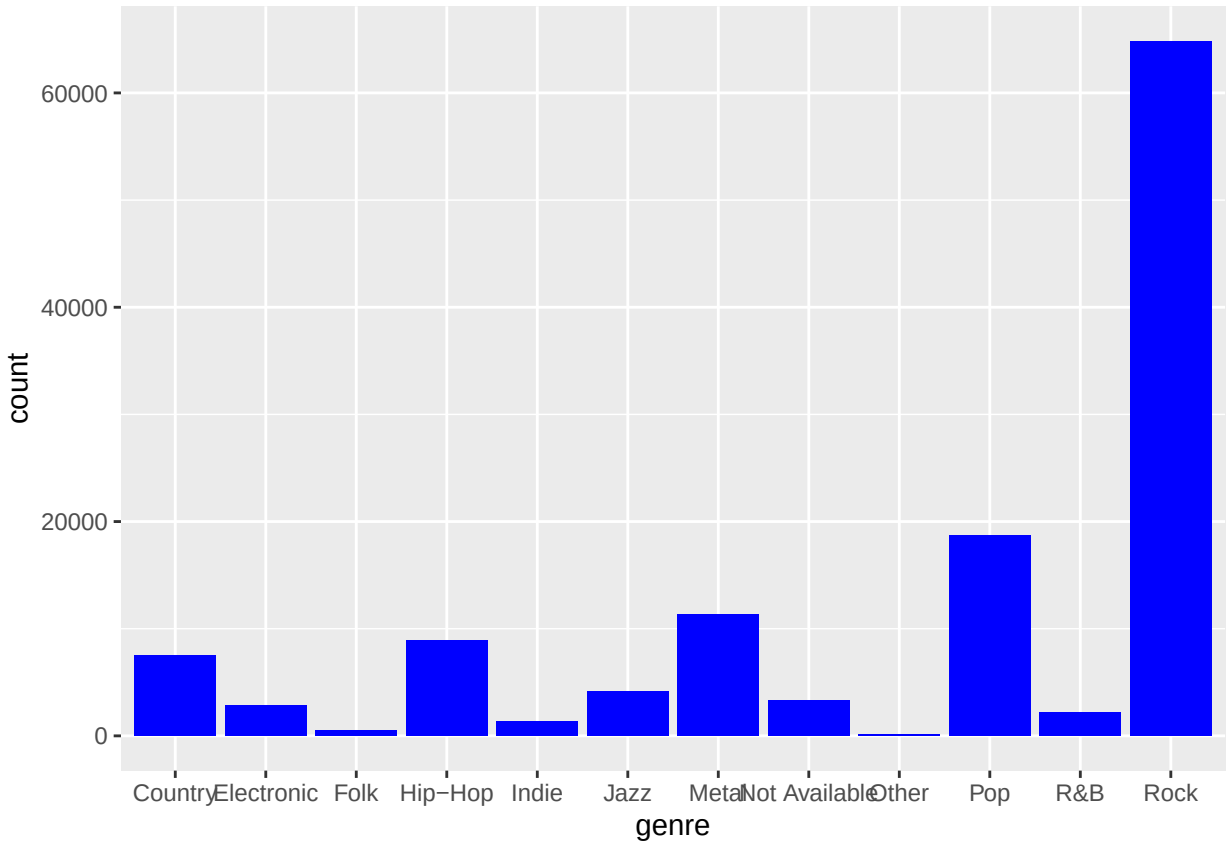
```
##  [1] "Hip-Hop"       "Other"         "Pop"           "Metal"
##  [5] "Rock"          "Country"       "Indie"         "Jazz"
##  [9] "Not Available" "Electronic"    "R&B"           "Folk"
```

```
hip_hop <- dt_lyrics%>%filter(genre == "Hip-Hop")
pop <- dt_lyrics%>%filter(genre == "Pop")
metal <- dt_lyrics%>%filter(genre == "Metal")
rock <- dt_lyrics%>%filter(genre == "Rock")
country <- dt_lyrics%>%filter(genre == "Country")
indie <- dt_lyrics%>%filter(genre == "Indie")
jazz <- dt_lyrics%>%filter(genre == "Jazz")
electronic <- dt_lyrics%>%filter(genre == "Electronic")
rb <- dt_lyrics%>%filter(genre == "R&B")
flok <- dt_lyrics%>%filter(genre == "Folk")
```

Step 5 - Observe difference among songs with different genre

1. Amount

```
plot_amount <- ggplot(dt_lyrics,aes(genre))+geom_bar(fill="blue")
plot_amount
```



We can see that the Rock music has the largest amount nad Flok music is the leatest

2. Top 10 frequent words

```
# define functions
mycorpus <- function(df) {
  Corpus(VectorSource(df$stemmedwords))
}
Tdm.tidy <- function(corpus) {
  tidy(TermDocumentMatrix(corpus))
}
Tdm.overall <- function(tidy) {
  summarise(group_by(tidy, term), sum = sum(count))
}
# Applying the function of all the genre
myCorpus.pop <- mycorpus(pop)
tdm.pop <- Tdm.tidy(myCorpus.pop)
tdm.overallpop <- Tdm.overall(tdm.pop)
myCorpus.rock <- mycorpus(rock)
tdm.rock <- Tdm.tidy(myCorpus.rock)
tdm.overallrock <- Tdm.overall(tdm.rock)
myCorpus.hiphop <- mycorpus(hip_hop)
tdm.hiphop <- Tdm.tidy(myCorpus.hiphop)
tdm.overallhiphop <- Tdm.overall(tdm.hiphop)
myCorpus.metal <- mycorpus(metal)
tdm.metal <- Tdm.tidy(myCorpus.metal)
tdm.overallmetal <- Tdm.overall(tdm.metal)
```

```
myCorpus.country <- mycorpus(country)
tdm.country <- Tdm.tidy(myCorpus.country)
tdm.overallcountry <- Tdm.overall(tdm.country)
myCorpus.indie <- mycorpus(indie)
tdm.indie <- Tdm.tidy(myCorpus.indie)
tdm.overallindie <- Tdm.overall(tdm.indie)
myCorpus.rb <- mycorpus(rb)
tdm.rb <- Tdm.tidy(myCorpus.rb)
tdm.overallrb <- Tdm.overall(tdm.rb)
myCorpus.electronic <- mycorpus(electronic)
tdm.electronic <- Tdm.tidy(myCorpus.electronic)
tdm.overallelectronic <- Tdm.overall(tdm.electronic)
myCorpus.flok <- mycorpus(flok)
tdm.flok <- Tdm.tidy(myCorpus.flok)
tdm.overallflok <- Tdm.overall(tdm.flok)
myCorpus.hiphop <- Corpus(VectorSource(hip_hop$stemmedwords))
tdm.hiphop <- TermDocumentMatrix(myCorpus.hiphop)
tdm.tidyhiphop=tidy(tdm.hiphop)
myCorpus.jazz <- Corpus(VectorSource(jazz$stemmedwords))
tdm.jazz <- TermDocumentMatrix(myCorpus.jazz)
tdm.tidyjazz=tidy(tdm.jazz)
tdm.overalljazz <- summarise(group_by(tdm.tidyjazz, term), sum = sum(count))
myCorpus.hippop <- mycorpus(hip_hop)
tdm.hippop <- Tdm.tidy(myCorpus.hippop)
tdm.overallhippop <- Tdm.overall(tdm.hippop)
myCorpus.flok <- mycorpus(flok)
tdm.flok <- Tdm.tidy(myCorpus.flok)
tdm.overallflok <- Tdm.overall(tdm.flok)
tdm.pop <- Tdm.tidy(myCorpus.pop)
order.hiphop <- tdm.overallhiphop%>%arrange(desc(sum))
order.jazz <- tdm.overalljazz%>%arrange(desc(sum))
order.country <- tdm.overallcountry%>%arrange(desc(sum))
order.rock <- tdm.overallrock%>%arrange(desc(sum))
order.metal <- tdm.overallmetal%>%arrange(desc(sum))
order.indie <- tdm.overallindie%>%arrange(desc(sum))
order.electronic <- tdm.overallelectronic%>%arrange(desc(sum))
order.rb <- tdm.overallrb%>%arrange(desc(sum))
order.flok <- tdm.overallflok%>%arrange(desc(sum))
order.pop <- tdm.overallpop%>%arrange(desc(sum))
top_words <- tibble(metal = head(order.metal$term,10),rock = head(order.rock$term,10),country = head(or
top_words
```

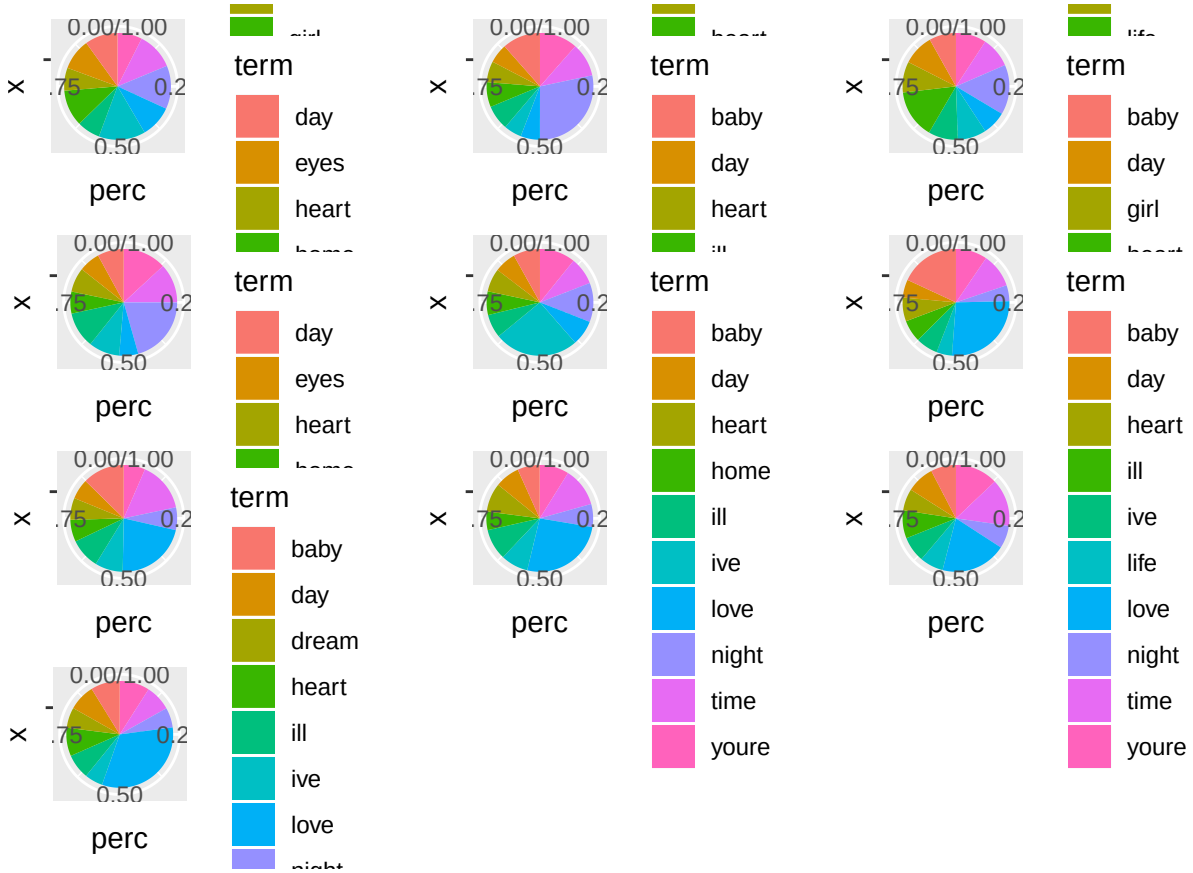
```
## # A tibble: 10 x 10
##    metal rock  country indie jazz  electronic rb    flok  hip_hop pop
##    <chr> <chr> <chr>   <chr> <chr> <chr>      <chr> <chr> <chr>   <chr>
##  1 time  love  love    love  love  love       love  love  love    love
##  2 life  time  time    youre youre time       baby  time  shit    baby
##  3 die   youre ill     time  baby  youre      time  day   time    youre
##  4 eyes  ill   youre   ill   heart world      youre ill   girl    time
##  5 world day   heart   ive   day   baby       girl  ive   baby    heart
##  6 youre ive   ive     day   time  night      ill   night niggas  ill
##  7 live  baby  day     heart ill   ill        heart home  bitch   girl
##  8 love  life  night   home  dream life       day   heart youre   life
##  9 day   night baby    eyes  night heart      night youre ill     day
```

```
## 10 soul  heart home    life  ive   day        life  eyes  chorus  ive
```

Almost all the songs has the most frequent word "love", but metal music has the most frequent word "time". Metal music focus most on time and life, it is different form other type of songs. Hip_hop music is another special one, it has some words that almost never been observed in other songs.

3.Proportion of top 10 words

```
perc_hiphop <- head(order.hiphop,10)%>%mutate(perc = sum/sum(sum))
perc_pop <- head(order.pop,10)%>%mutate(perc = sum/sum(sum))
perc_metal <- head(order.metal,10)%>%mutate(perc = sum/sum(sum))
perc_indie <- head(order.indie,10)%>%mutate(perc = sum/sum(sum))
perc_electronic <- head(order.electronic,10)%>%mutate(perc = sum/sum(sum))
perc_rb <- head(order.rb,10)%>%mutate(perc = sum/sum(sum))
perc_flok <- head(order.flok,10)%>%mutate(perc = sum/sum(sum))
perc_country <- head(order.country,10)%>%mutate(perc = sum/sum(sum))
perc_rock <- head(order.rock,10)%>%mutate(perc = sum/sum(sum))
perc_jazz <- head(order.jazz,10)%>%mutate(perc = sum/sum(sum))
p_hiphop = ggplot(perc_hiphop, aes(x = "", y = perc, fill = term)) +  geom_bar(stat = "identity") +  co
p_pop = ggplot(perc_pop, aes(x = "", y = perc, fill = term)) +  geom_bar(stat = "identity") +  coord_pol
p_metal = ggplot(perc_metal, aes(x = "", y = perc, fill = term)) +  geom_bar(stat = "identity") +  coor
p_indie = ggplot(perc_indie, aes(x = "", y = perc, fill = term)) +  geom_bar(stat = "identity") +  coor

p_electronic = ggplot(perc_electronic, aes(x = "", y = perc, fill = term)) +  geom_bar(stat = "identity

p_rb = ggplot(perc_rb, aes(x = "", y = perc, fill = term)) +  geom_bar(stat = "identity") +  coord_polar

p_flok = ggplot(perc_flok, aes(x = "", y = perc, fill = term)) +  geom_bar(stat = "identity") +  coord_p

p_country = ggplot(perc_country, aes(x = "", y = perc, fill = term)) +  geom_bar(stat = "identity") +

p_rock = ggplot(perc_rock, aes(x = "", y = perc, fill = term)) +  geom_bar(stat = "identity") +  coord_p

p_jazz = ggplot(perc_jazz, aes(x = "", y = perc, fill = term)) +  geom_bar(stat = "identity") +  coord_p

grid.arrange(p_hiphop,p_pop,p_metal,p_indie,p_electronic,p_rb,p_flok,p_country,p_rock,p_jazz)
```



In hip_hop music, the proportion of top 10 word are almost same. In pop music, love has a large proportion. In metal music, life and time account for half of the proportion.And in jazz music, love has the largest propotion than other music.