Q3

David

2019-03-06

```
library(knitr)
library(twitteR)
consumer_key = "dOLrbYfXjbDo7f7IZ3MFFDF4E"
consumer_secret = "GKbWkDANMboESv0YwvSSZKWivvbIyEzhzZxroDIODY1thrk0Gz"
access_token = "1050995496381120513-8E1z4xtsUYmNTjTRZCRLN4WYMkJjyZ"
access_secret = "bAGtoWn9wR0EUKPybvr7yhK0FFaFJNRviV2LINV06pgRp"
setup_twitter_oauth(consumer_key, consumer_secret,
access_token, access_secret)
```

```
## [1] "Using direct authentication"
```

a.

```
AStarIsBorn = searchTwitter(searchString =
'starisbornmovie', n = 500, lang = "en")
BlackPanther = searchTwitter(searchString =
'theblackpanther', n = 500, lang = "en")
Vice = searchTwitter(searchString =
'vicemovie', n = 500, lang = "en")
AStarIsBorn = twListToDF(AStarIsBorn)
BlackPanther = twListToDF(BlackPanther)
Vice = twListToDF(Vice)
nrow(AStarIsBorn)
```

```
## [1] 500
```

```
nrow(BlackPanther)
```

```
## [1] 500
```

```
nrow(Vice)
```

```
## [1] 500
```

b.

```
pos = scan("positive-words.txt", what = "character", comment.char = ";")
neg = scan("negative-words.txt", what = "character", comment.char = ";")
#filter out the movie name from negative words
grep("shallow",neg)
```

```
## Warning in grep("shallow", neg): input string 3008 is invalid in this
## locale
```

```
## [1] 3771
```

```
neg=neg[-3771]
grep("vice",neg)
```

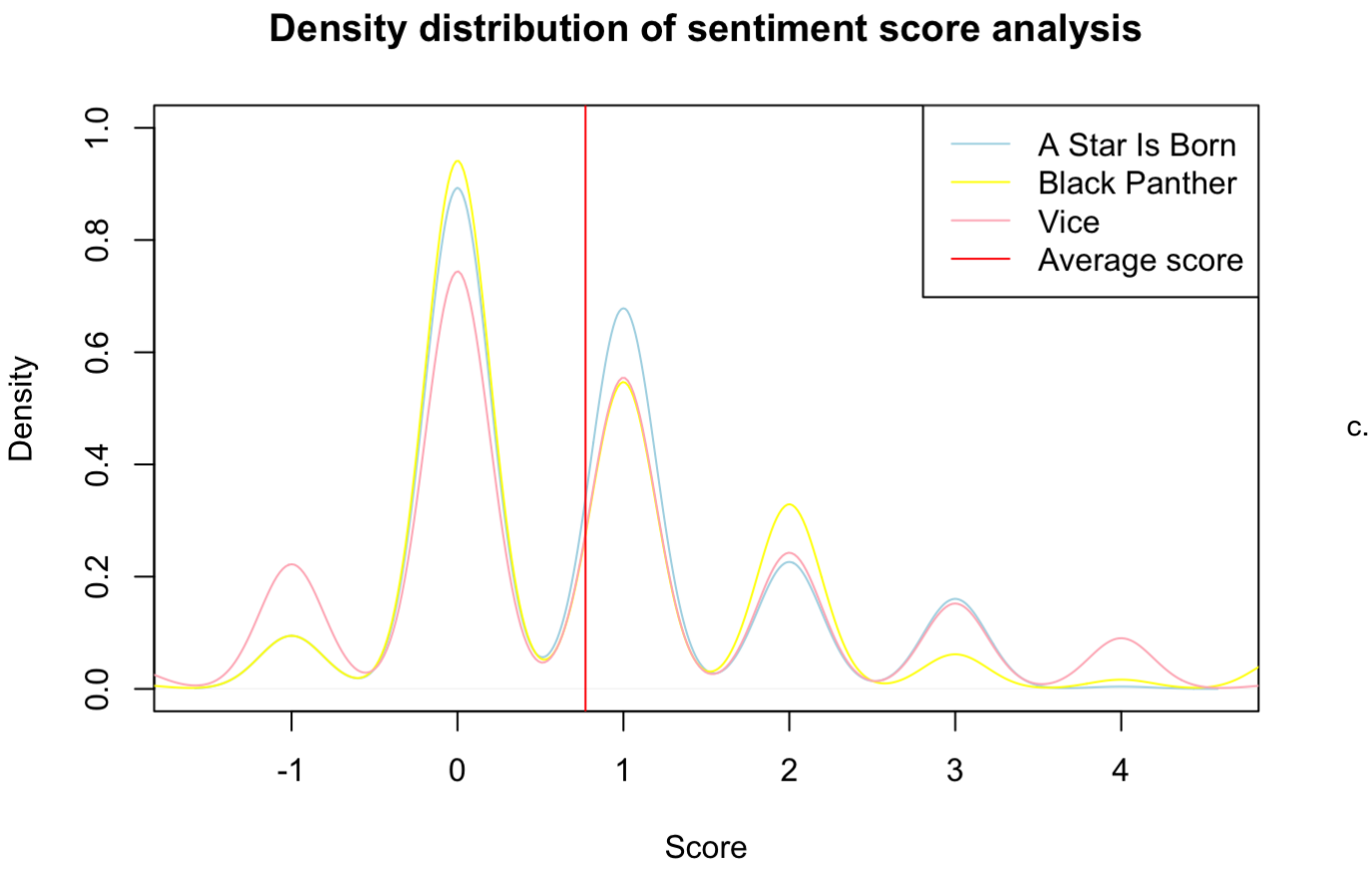
```
## Warning in grep("vice", neg): input string 3008 is invalid in this locale
```

```
## [1] 1291 4630
```

```
neg=neg[-4630]
```

```
getSentimentScore = function(tweet_text, pos, neg) {

tweet_text = tweet_text$text
tweet_text = gsub("(RT|via)(?:\\b\\W*@\\w+)", " ", tweet_text)
tweet_text = gsub("@\\w+", " ", tweet_text)
tweet_text = gsub("&\\w+;", "", tweet_text)
tweet_text = gsub("[[:cntrl:]]", "", tweet_text)
tweet_text = gsub("(n|N)o.", "number", tweet_text)
tweet_text = gsub("[[:digit:]]", "", tweet_text)
tweet_text = gsub("(?!')[[:punct:]]", "", tweet_text, perl = T)
tweet_text = gsub("(\\B'|'\\B)", "", tweet_text)
tweet_text = gsub("(\\B'|'\\B)", "", tweet_text)
tweet_text = gsub("\\w+\\…","", tweet_text)
tweet_text = iconv(tweet_text, "ASCII", "UTF-8", sub = "")
tweet_text = tolower(tweet_text)
tweet_text = gsub("http\\w+", "", tweet_text)
tweet_text = gsub("[ \t]{2,}", " ", tweet_text)
tweet_text = gsub("^\\s+|\\s+$", "", tweet_text)
words=strsplit(tweet_text," ")
score = numeric(length(words))
# loop through each tweet
for (i in 1:length(words)) {
# compare our words to the dictionaries of positive
# & negative terms
pos.matches = match(words[[i]], pos)
neg.matches = match(words[[i]], neg)
# match() returns the position of the matched term
# or NA we just want a TRUE/FALSE:
pos.matches = !is.na(pos.matches)
neg.matches = !is.na(neg.matches)
# and conveniently enough, TRUE/FALSE will be
# treated as 1/0 by sum():
score[i] = sum(pos.matches) - sum(neg.matches)
}
return(score)
}
AStarIsBorn_score=getSentimentScore(AStarIsBorn, pos, neg)
BlackPanther_score=getSentimentScore(BlackPanther, pos, neg)
Vice_score=getSentimentScore(Vice, pos, neg)
plot(density(AStarIsBorn_score),col="lightblue", xlab = "Score", ylim = c(0,1), main =
"Density distribution of sentiment score analysis")
lines(density(BlackPanther_score),col="yellow")
lines(density(Vice_score),col="lightpink")
abline(v=mean(c(AStarIsBorn_score,BlackPanther_score,Vice_score)), col=2)
legend(x="topright", lty = c(1,1,1,1), col=c("lightblue", "yellow", "lightpink", "red"),
 c("A Star Is Born", "Black Panther", "Vice", "Average score"))
```



c.

```
#calculate mean scores and choose the highest mean as most positive
mean(AStarIsBorn_score)
```

```
## [1] 0.746
```

```
mean(BlackPanther_score)
```

```
## [1] 0.794
```

```
mean(Vice_score)
```

```
## [1] 0.774
```

```
sentence = BlackPanther$text
sentence = gsub("(RT|via)(?:\\b\\W*@\\w+)", " ", sentence)
sentence = gsub("@\\w+", " ", sentence)
sentence = gsub("&\\w+;", "", sentence)
sentence = gsub("[[:cntrl:]]", "", sentence)
sentence = gsub("(n|N)o.", "number", sentence)
sentence = gsub("[[:digit:]]", "", sentence)
sentence = gsub("(?!')[[:punct:]]", "", sentence, perl = T)
sentence = gsub("(\\B'|'\\B)", "", sentence)
sentence = gsub("(\\B'|'\\B)", "", sentence)
sentence = gsub("\\w+\\…","", sentence)
sentence = iconv(sentence, "ASCII", "UTF-8", sub = "")
sentence = tolower(sentence)
sentence = gsub("http\\w+", "", sentence)
sentence = gsub("[ \t]{2,}", " ", sentence)
sentence = gsub("^\\s+|\\s+$", "", sentence)
words=unlist(strsplit(sentence," "))
#initialize a vector to store pos_word
pos_word=vector()
# loop through each tweet
for (i in 1:length(words)) {
pos.matches = match(words[i], pos)
pos.matches = !is.na(pos.matches)
if(pos.matches == TRUE){
  pos_word = c(pos_word, words[i])
}
}
```

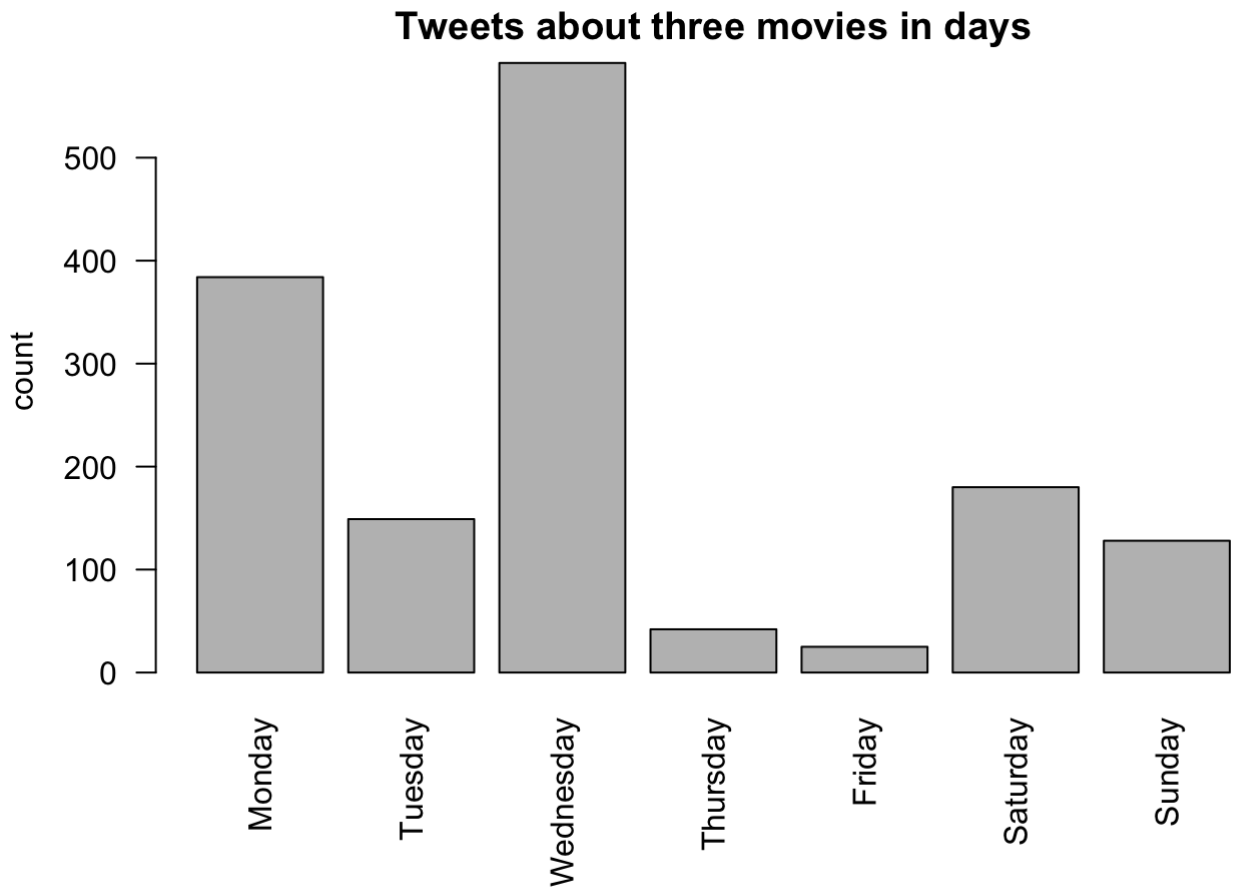
```
library(wordcloud)
```

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

```
freq=sort(table(pos_word),decreasing = TRUE)
layout(matrix(c(1, 2), nrow=2), heights=c(1, 4))
par(mar=rep(0, 4))
plot.new()
text(x=0.5, y=0.25, "Postive words in tweets @theblackpanther")
wordcloud(names(freq), freq, min.freq = 1, rot.per = 0.1,
colors = rainbow(8), random.order = FALSE)
```

d.

```
Combine = rbind(AStarIsBorn, BlackPanther)
Combine = rbind(Combine, Vice)
par(mar=c(7,4,2,2))
barplot(table(weekdays(Combine$created))[c("Monday","Tuesday","Wednesday","Thursday","Fr
iday","Saturday","Sunday")], ylab = "count", las = 2, main = "Tweets about three movies
 in days")
```



Wednesday is the day most tweets talk about the three movies during the week. It's misleading because we don't know which movie is mentioned