STAT240 Lab8

Jiajun Zhang

March 12, 2019

```
library(ROAuth)
library(twitteR)
```

```
setup_twitter_oauth(consumer_key, consumer_secret, access_token, access_secret)
```

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

Question1

a)

```
# TL_tweet=userTimeline("TransLink", n=3200)
# save("TL_tweet", file = "TransLinkTweets.Rdata")
load("TransLinkTweets.Rdata")
TL_df=twListToDF(TL_tweet)
nrow(TL_df)
```

```
## [1] 3200
```

b)

```
c( TL_df[1,"created"], TL_df[3200,"created"] )
```

```
## [1] "2019-03-11 13:01:17 PDT" "2019-02-14 18:07:22 PST"
```

```
#The "created" column is sorted, the most recent time is in the first row, oldest is in the last
c( max(TL_df[,"created"]), min(TL_df[,"created"]) )
```

```
## [1] "2019-03-11 13:01:17 PDT" "2019-02-14 18:07:22 PST"
```

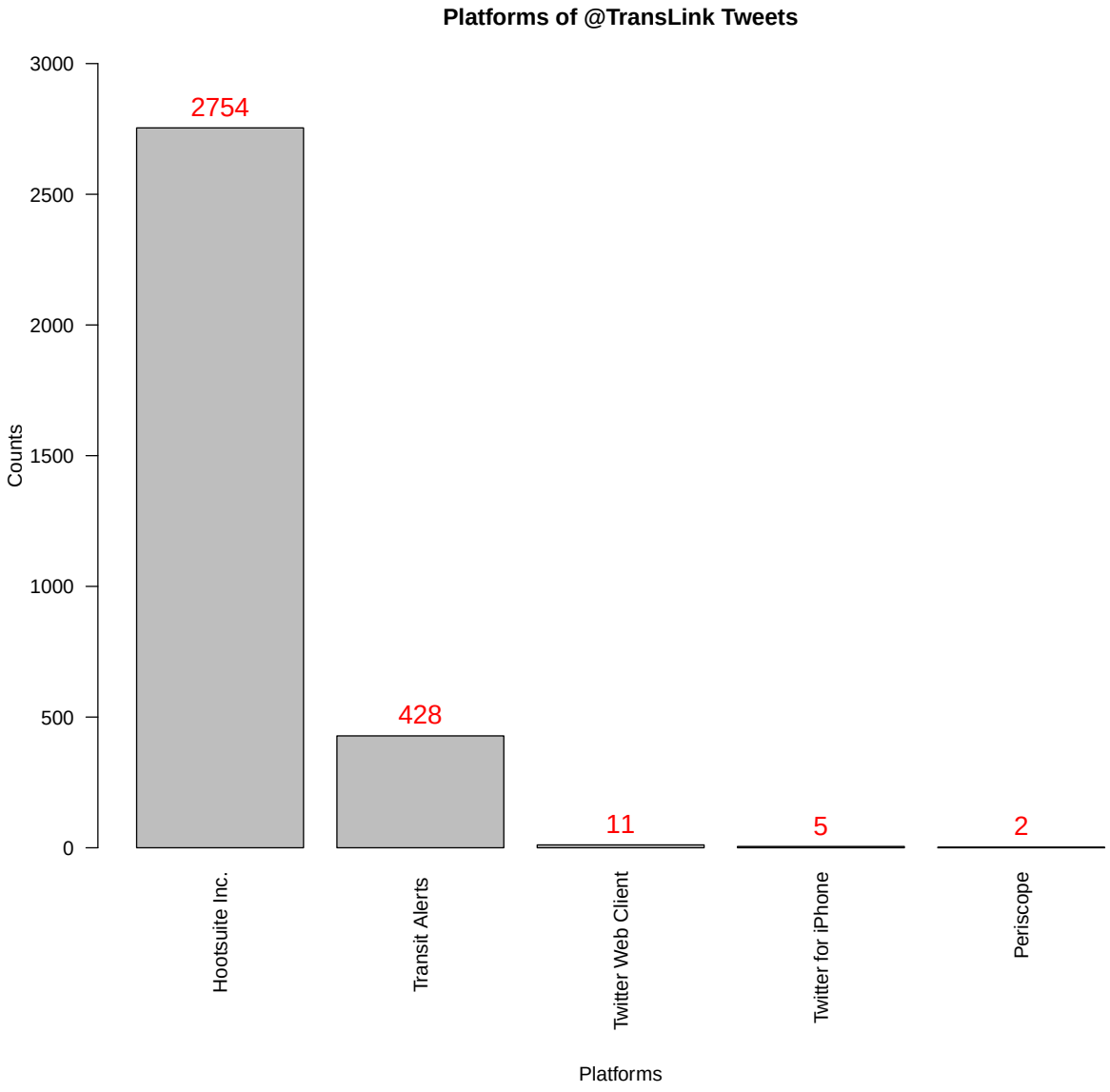
```
difftime(TL_df[1,"created"], TL_df[3200,"created"])
```

```
## Time difference of 24.74578 days
```

c)

```
# head(TL_df$statusSource,25)
TL_source=TL_df$statusSource
remove_http=gsub("^.*?>", "", TL_source)
platform=gsub("</a>", "", remove_http)
# platform
TL_table=as.data.frame(sort(table(platform), decreasing=T))
par(mar=c(13,4,4,2))
bp=barplot(sort(table(platform),decreasing=T), las=2, ylim=c(0, 3000),
```

```
          main="Platforms of @TransLink Tweets", ylab="Counts")
text(x=bp, y=TL_table$Freq, label=TL_table$Freq,
     cex = 1.3, col = "red", pos=3)
title(xlab="Platforms", line=9)
```



Question2

a)

```
library(stringr)
c(str_which(OlsonNames(), "Montreal"), OlsonNames()[str_detect(OlsonNames(), "Montreal")])
```

```
## [1] "166"              "America/Montreal"
```

```
c(str_which(OlsonNames(), "Tokyo"), OlsonNames()[str_detect(OlsonNames(), "Tokyo")])
```

```
## [1] "319"        "Asia/Tokyo"
```

```
c(str_which(OlsonNames(), "Dubai"), OlsonNames()[str_detect(OlsonNames(), "Dubai")])
```

```
## [1] "261"        "Asia/Dubai"
```

b)

```
TL=getUser("TransLink")
TL$location
```

```
## [1] "Metro Vancouver"
```

c)

```
TL_df$created=as.POSIXct(as.integer(TL_df$created),
                         origin = "1970-01-01", tz = "America/Vancouver")
head(TL_df$created,5)
```

```
## [1] "2019-03-11 13:01:17 PDT" "2019-03-11 12:59:20 PDT"
## [3] "2019-03-11 12:58:08 PDT" "2019-03-11 12:54:29 PDT"
## [5] "2019-03-11 12:31:48 PDT"
```

```
tail(TL_df$created,5)
```

```
## [1] "2019-02-14 18:13:43 PST" "2019-02-14 18:12:19 PST"
## [3] "2019-02-14 18:10:52 PST" "2019-02-14 18:07:43 PST"
## [5] "2019-02-14 18:07:22 PST"
```

Question3

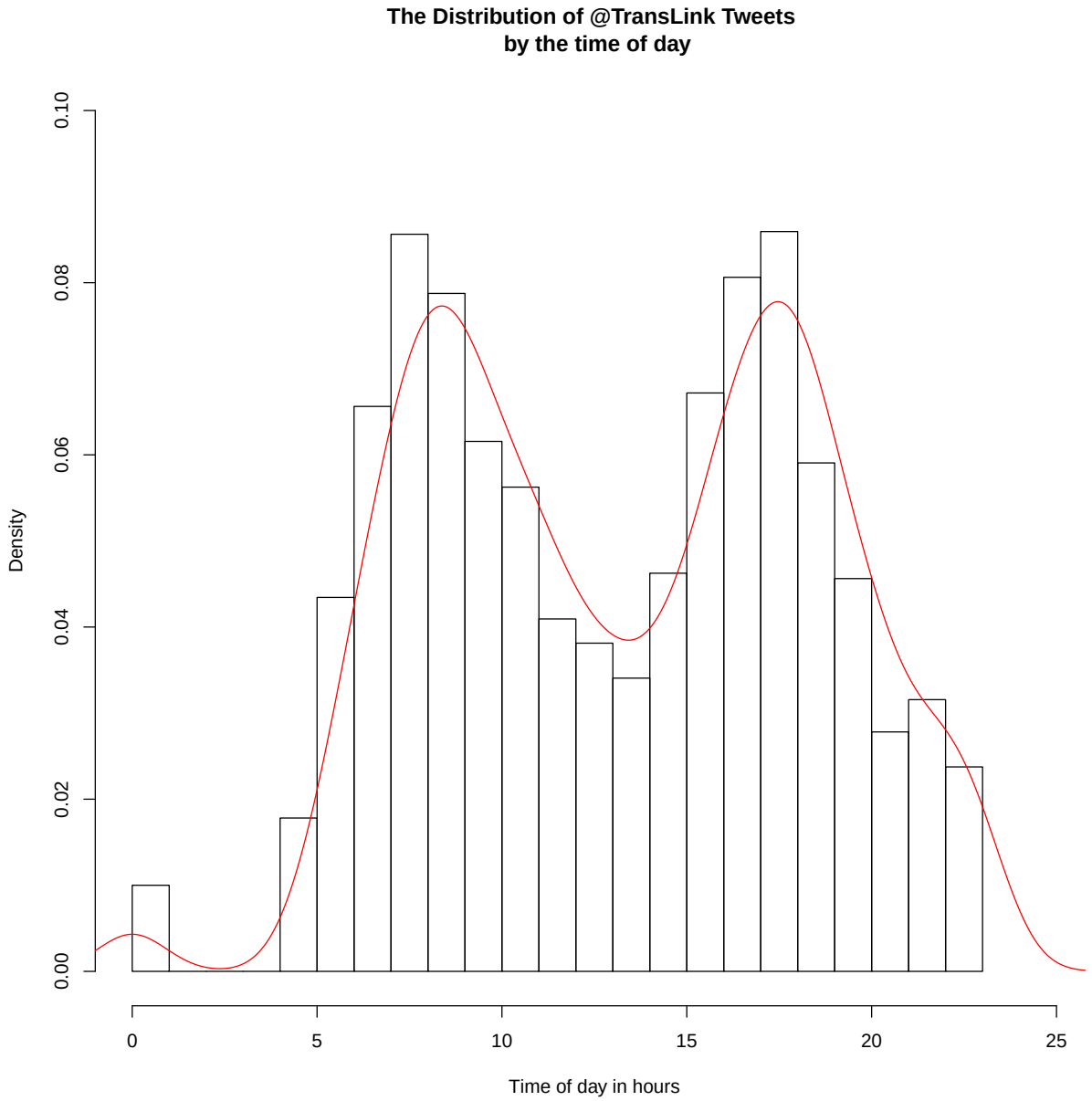
a)

```
TL_hr=trunc(TL_df$created, "hours")
#remove "%yyyy-%mm-%dd"
TL_hr_rm1=gsub("^.*?\\s","", TL_hr)
#get rid of ":00:00"
TL_hr_rm2=gsub("\\:+.+$", "", TL_hr_rm1)
head(TL_hr_rm2, 5)
```

```
## [1] "13" "12" "12" "12" "12"
```

b)

```
hist(as.numeric(TL_hr_rm2), freq=FALSE, xlim=c(0,25), breaks=24,
     ylim=c(0,0.1), xlab="Time of day in hours",
     main="The Distribution of @TransLink Tweets \n by the time of day")
lines(density(as.numeric(TL_hr_rm2)), col="red")
```

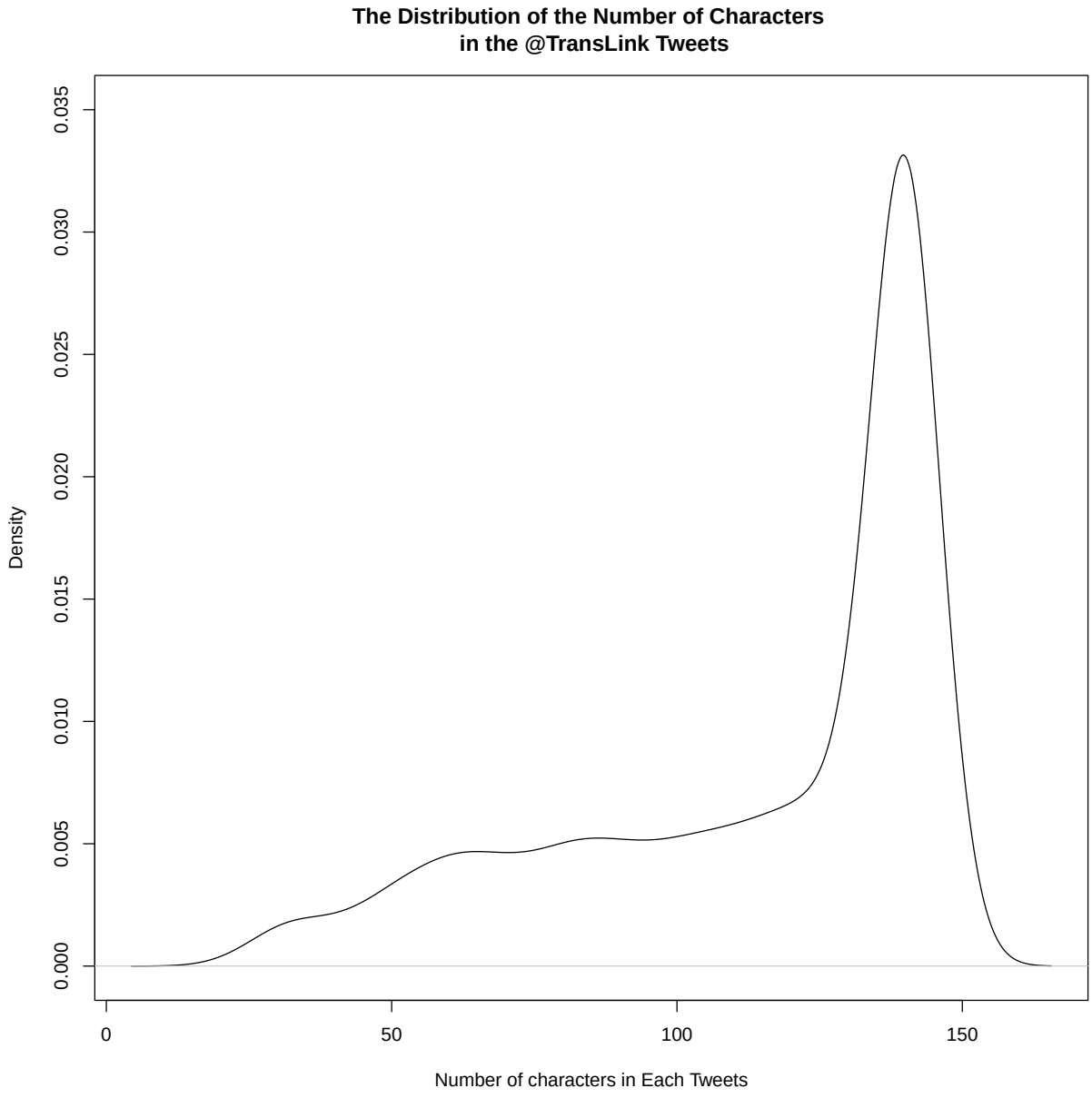


- The user @TransLink most likely sends Tweets in the morning around 8am and in the afternoon around 6pm.

Question4

a)

```
plot(density(nchar(TL_df$text)), ylim=c(0,0.035),
     xlab="Number of characters in Each Tweets",
     main="The Distribution of the Number of Characters \n in the @TransLink Tweets")
```



b)

```
#find the rows where text char is more than 140
TL_chr140=TL_df[nchar(TL_df$text)>140, ]

#convert the dates to "%yyyy/%mm/%dd" form
TL_chr140$created=substr(cut(as.POSIXct(TL_chr140[,"created"]),"days"), start=0, stop=10)
head(TL_chr140$created, 4)
```

```
## [1] "2019-03-11" "2019-03-10" "2019-03-10" "2019-03-10"
```

```
#Check, no dates before "2017-09-26"
table(TL_chr140$created<"2017-09-26")
```

```
##
## FALSE
##   175
```

```
#Assume there is text, then output
TL_chr140$text[TL_chr140$created<"2017-09-26"]
```

```
## character(0)
```

Question5

a)

```
source("getSentimentScore.R")
pos = scan("positive-words.txt", what = "character", comment.char = ";")
neg = scan("negative-words.txt", what = "character", comment.char = ";")

#a)
suppressPackageStartupMessages(library(dplyr))

score=getSentimentScore(TL_df$text, pos, neg)
TL_df_score=cbind(TL_df, score)

Weeks=as.factor(as.Date(cut(TL_df_score$created, "weeks")))
TL_df_score=cbind(TL_df_score, Weeks)

TL_by_weeks=group_by(TL_df_score, Weeks)

avg_df = summarise(TL_by_weeks, Avg.Pos.Words=mean(positive_word_count),
          Avg.Neg.Words=mean(negative_word_count),
          Avg.Sent.Score=mean(sentiment_score))
head(avg_df, 5)
```

```
## # A tibble: 5 x 4
##   Weeks      Avg.Pos.Words Avg.Neg.Words Avg.Sent.Score
##   <fct>              <dbl>         <dbl>          <dbl>
## 1 2019-02-11         0.329         0.447        -0.118
## 2 2019-02-18         0.354         0.523        -0.169
## 3 2019-02-25         0.448         0.519        -0.0708
## 4 2019-03-04         0.403         0.584        -0.181
## 5 2019-03-11         0.417         0.5          -0.0833
```

```
# a=TL_by_weeks[TL_by_weeks[,"Weeks"]=="2019-02-11", ]
# c(mean(a$positive_word_count), mean(a$negative_word_count), mean(a$sentiment_score))
#
# a1=TL_by_weeks[TL_by_weeks[,"Weeks"]=="2019-02-18", ]
# c(mean(a1$positive_word_count), mean(a1$negative_word_count), mean(a1$sentiment_score))
```

b)

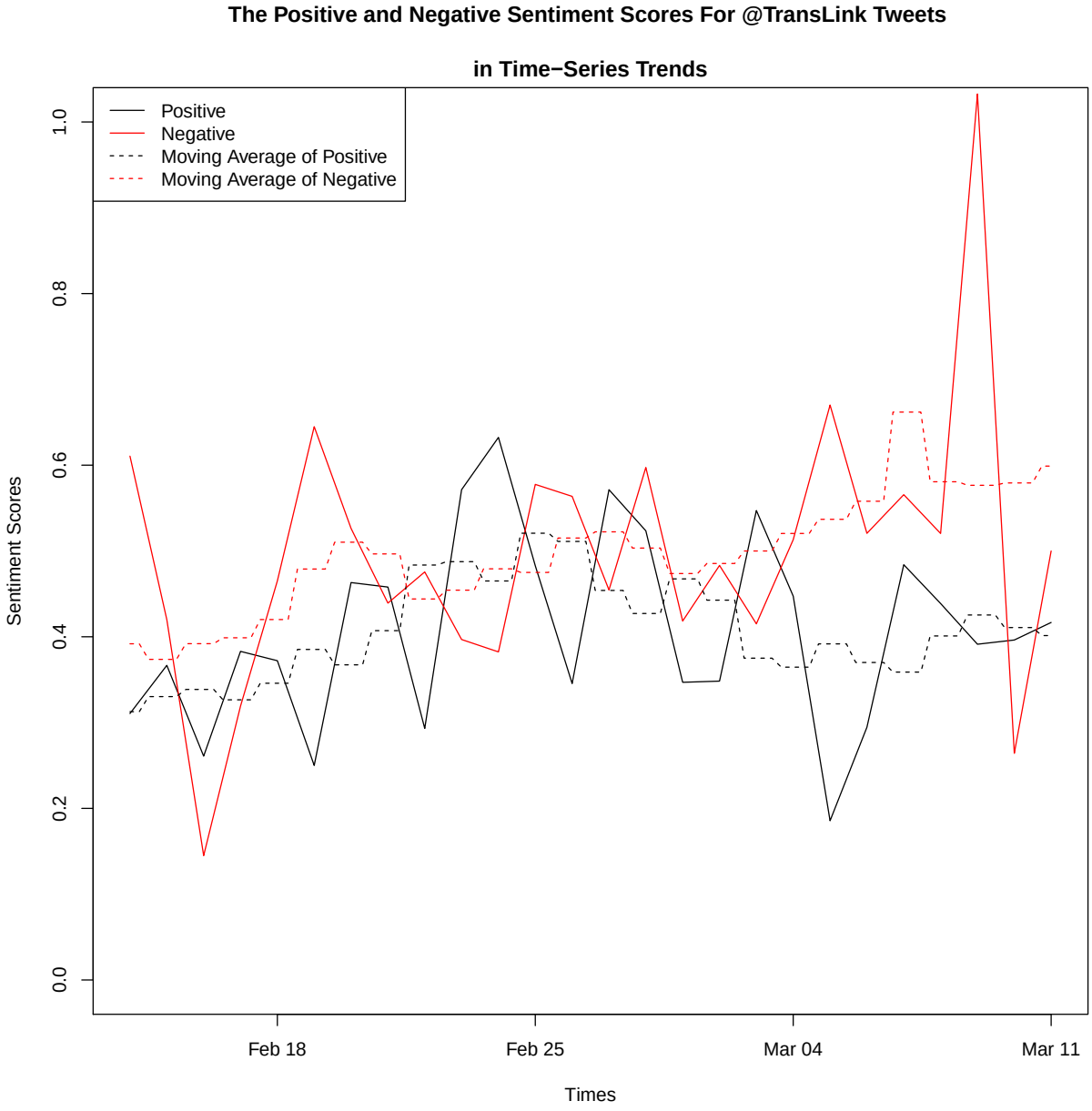
```
# plot(as.numeric(TL_by_weeks$created), TL_by_weeks$positive_word_count, type="l")

# Since cutting dates into "weeks" does not provide a good Time-Series trending;
#here, using dates by "days" instead to make a better plot
Days=as.Date(cut(TL_df_score$created, "days"))
TL_df_score=cbind(TL_df_score, Days)
TL_by_days=group_by(TL_df_score, Days)
avg_days_df=summarise(TL_by_days, Avg.Pos.Words=mean(positive_word_count),
          Avg.Neg.Words=mean(negative_word_count),
          Avg.Sent.Score=mean(sentiment_score))
```

```r
plot(avg_days_df$Days, avg_days_df$Avg.Pos.Words, type="l", xlab="Times",
     ylab="Sentiment Scores", ylim=c(0, 1),
     main="The Positive and Negative Sentiment Scores For @TransLink Tweets \n
in Time-Series Trends")
lines(avg_days_df$Days, avg_days_df$Avg.Neg.Words, type="l", col=2)

lines(ksmooth(avg_days_df$Days, avg_days_df$Avg.Pos.Words, bandwidth=5),
      col=1, type="l", lty=2)
lines(ksmooth(avg_days_df$Days, avg_days_df$Avg.Neg.Words, bandwidth=5),
      col=2, type="l", lty=2)


legend("topleft", c("Positive", "Negative", "Moving Average of Positive",
                    "Moving Average of Negative"),
       col=c(1,2,1,2), lty=c(1,1,2,2))
```


The Positive and Negative Sentiment Scores For @TransLink Tweets
in Time−Series Trends
Positive
Negative
Moving Average of Positive
Moving Average of Negative
Sentiment Scores
0.0
0.2
0.4
0.6
0.8
1.0
Feb 18
Feb 25
Mar 04
Mar 11
Times

Question6

a)

```
TL_text=TL_df$text

#Hashtag
#remove every punctuations but "#" and "@"
TL_text=gsub("(?!(\\#|\\@))[[:punct:]]", "", TL_text, perl=T)

hashtag_words=unlist(str_extract_all(TL_text, "^*\\#+[a-z|A-Z]{1,}[^\\s]"))
table(hashtag_words)
```

```
## hashtag_words
##       #Balanceforbetter                #Compass            #CompassCard
##                       1                       4                       2
##     #DaylightSavingTime              #HandyDART #internationalwomensday
##                       1                       1                       1
##                   #iwd2           #Pinkshirtday             #Rideralert
##                       1                       1                       1
##             #RiderAlert            #RiderAlert3                 #SeaBus
##                     447                       2                       2
##               #SkyTrain              #SkyTrains               #sofancy7
##                      82                       1                       1
##          #StationAccess           #StationAlert           #TransitAlert
##                       2                      65                       8
##                    #WCE                    #YVR
##                       5                       1
```

```
#@Mentions
mention_words=unlist(str_extract_all(TL_text, "^*\\@+[a-z|A-Z]{1,}[^\\s]"))
length(mention_words)
```

```
## [1] 2719
```

```
head(mention_words, 8)
```

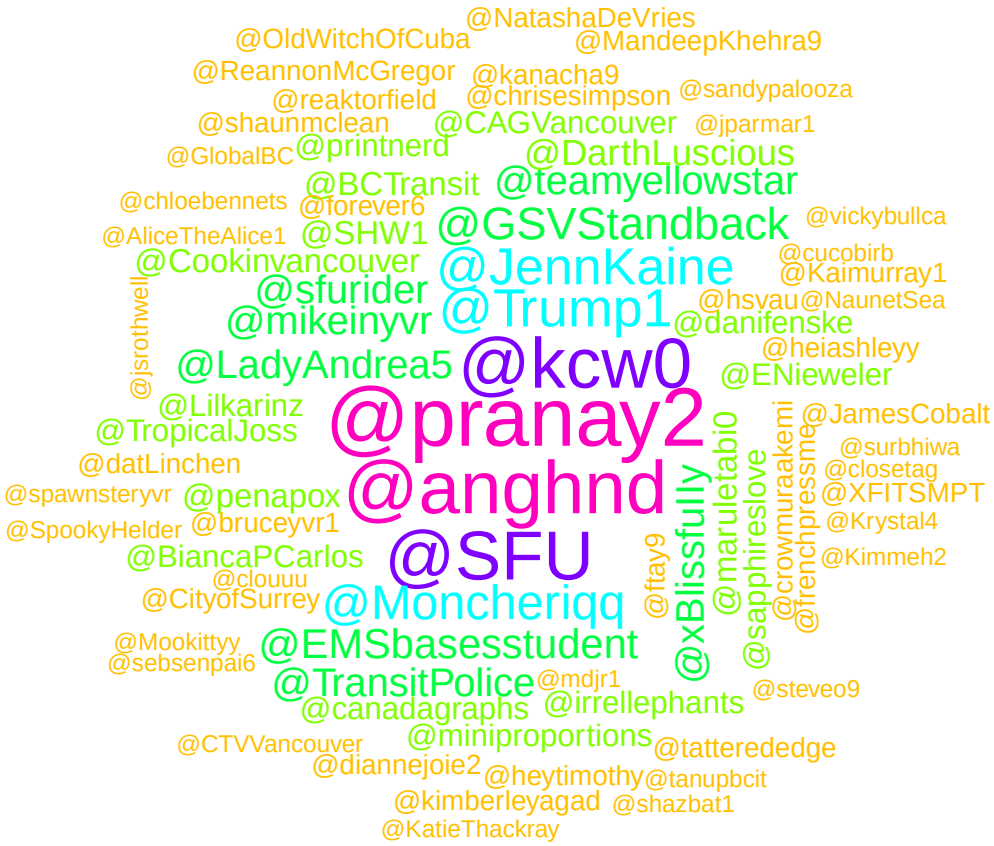
```
## [1] "@TropicalJoss"  "@KevlarGiraffe" "@Earth2"        "@Earth2"
## [5] "@KevlarGiraffe" "@steveo9"       "@TropicalJoss"  "@TropicalJoss"
```

b)

```
library(wordcloud)
```

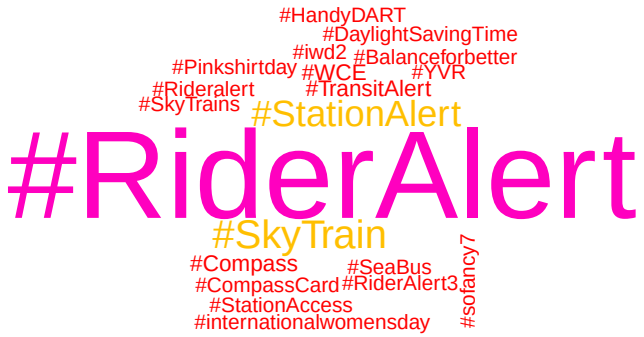
```
## Loading required package: RColorBrewer
```

```
# x11(width=14, height=14)
wordcloud(names(table(mention_words)), table(mention_words), min.freq = 5,
          colors = rainbow(8), random.order = FALSE)
```



c)

```
# x11(width=14, height=14)
wordcloud(names(table(hashtag_words)), table(hashtag_words), min.freq = 1,
          colors = rainbow(8), random.order = FALSE, scale=c(6,1))
```


#HandyDART
#DaylightSavingTime
#iwd2
#Balanceforbetter
#Pinkshirtday
#WCE
#YVR
#Rideralert
#TransitAlert
#SkyTrains
#StationAlert
#RiderAlert
#SkyTrain
#Compass
#SeaBus
#CompassCard
#RiderAlert3
#StationAccess
#internationalwomensday
#sofancy7