library(rtweet)

## Warning: package 'rtweet' was built under R version 3.5.3

library(tidytext)

## Warning: package 'tidytext' was built under R version 3.5.3

library(igraph)

## Warning: package 'igraph' was built under R version 3.5.3

##   
## Attaching package: 'igraph'

## The following objects are masked from 'package:stats':  
##   
## decompose, spectrum

## The following object is masked from 'package:base':  
##   
## union

library(dplyr)

## Warning: package 'dplyr' was built under R version 3.5.3

##   
## Attaching package: 'dplyr'

## The following objects are masked from 'package:igraph':  
##   
## as\_data\_frame, groups, union

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

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

library(ggraph)

## Warning: package 'ggraph' was built under R version 3.5.3

## Loading required package: ggplot2

## Warning: package 'ggplot2' was built under R version 3.5.3

library(sentimentr)

## Warning: package 'sentimentr' was built under R version 3.5.3

library(sqldf)

## Warning: package 'sqldf' was built under R version 3.5.3

## Loading required package: gsubfn

## Loading required package: proto

## Loading required package: RSQLite

library(tidyr)

## Warning: package 'tidyr' was built under R version 3.5.3

##   
## Attaching package: 'tidyr'

## The following object is masked from 'package:igraph':  
##   
## crossing

library(ggplot2)

tweets <- na.omit(read.csv(file="C:/Users/praveen/Documents/tweets.csv",stringsAsFactors = FALSE))  
nrow(tweets)

## [1] 71517

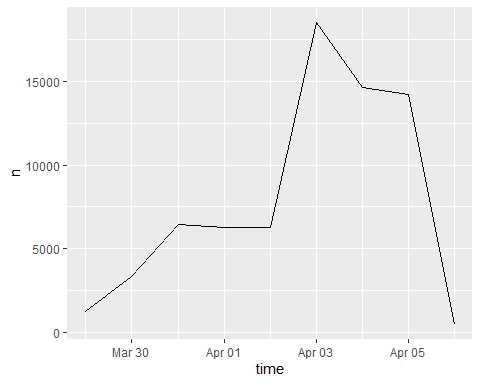
head(tweets)

## created\_at  
## 1 2020-04-05 23:59:26  
## 2 2020-04-05 23:59:24  
## 3 2020-04-05 23:58:51  
## 4 2020-04-05 23:58:05  
## 5 2020-04-04 09:13:02  
## 6 2020-04-05 23:49:58  
## text  
## 1 @MollyJongFast You realize that em the same info he sold his stock based on, was also public way before the stock market started going down, right?  
## 2 @Clayconboy1 I was Shocked when I checked official Government Statistics ( NOT from Liberal Sources, but the actual Stats) that ObamaÃ¢â‚¬â„¢s Economy wasnÃ¢â‚¬â„¢t nearly as bad as Limbaugh and others have said. Look it up. TrumpÃ¢â‚¬â„¢s GDP and Stock Market gains arenÃ¢â‚¬â„¢t better in some cases as ObamaÃ¢â‚¬â„¢s was.  
## 3 This Weeks Stock Market Outlook:\n\nIndexes still Blue White Red Downtrends.\nLackluster FTD Thursday\nNot many promising setups\n\nhttps://t.co/yKmGD3zJze  
## 4 @camjesse88 Show me one other person that predicted a sub 20K crash on the stock market (before it happenedH\n\nIf you do, IÃ¢â‚¬â„¢ll delete my page today. \n\nIÃ¢â‚¬â„¢m waiting.  
## 5 Everyone who doubted my calls has lost badly!\n\nI warned people of doom months ago. \n\nI said the stock market would plummet hard, and same with Bitcoin/Alts! \n\nIÃ¢â‚¬â„¢m so glad I could save people from losing their money, and will continue to do so in the future! https://t.co/UOkijYRuaP  
## 6 The Stock Market will gain new fears later this year when entering Phase 2. \n\nI was the only person who predicted the massive market crash, and many people made money from those calls.\n\nIf you shorted at 29K+ when I said, keep them open for 1+ years! I'll share more soon. https://t.co/tjSylfMwYH

nrow(unique(tweets))

## [1] 70578

ts\_plot(tweets,by="days")



tweets$text <- get\_sentences(tweets$text)

head(tweets)

## created\_at  
## 1 2020-04-05 23:59:26  
## 2 2020-04-05 23:59:24  
## 3 2020-04-05 23:58:51  
## 4 2020-04-05 23:58:05  
## 5 2020-04-04 09:13:02  
## 6 2020-04-05 23:49:58  
## text  
## 1 @MollyJongFast You realize that em the same info he sold his stock based on, was also public way before the stock market started going down, right?  
## 2 @Clayconboy1 I was Shocked when I checked official Government Statistics ( NOT from Liberal Sources, but the actual Stats) that ObamaÃ¢â‚¬â„¢s Economy wasnÃ¢â‚¬â„¢t nearly as bad as Limbaugh and others have said., Look it up., TrumpÃ¢â‚¬â„¢s GDP and Stock Market gains arenÃ¢â‚¬â„¢t better in some cases as ObamaÃ¢â‚¬â„¢s was.  
## 3 This Weeks Stock Market Outlook:\n\nIndexes still Blue White Red Downtrends., Lackluster FTD Thursday\nNot many promising setups\n\nhttps://t.co/yKmGD3zJze  
## 4 @camjesse88 Show me one other person that predicted a sub 20K crash on the stock market (before it happenedH\n\nIf you do, IÃ¢â‚¬â„¢ll delete my page today., IÃ¢â‚¬â„¢m waiting.  
## 5 Everyone who doubted my calls has lost badly!, I warned people of doom months ago., I said the stock market would plummet hard, and same with Bitcoin/Alts!, IÃ¢â‚¬â„¢m so glad I could save people from losing their money, and will continue to do so in the future!, https://t.co/UOkijYRuaP  
## 6 The Stock Market will gain new fears later this year when entering Phase 2., I was the only person who predicted the massive market crash, and many people made money from those calls., If you shorted at 29K+ when I said, keep them open for 1+ years!, I'll share more soon., https://t.co/tjSylfMwYH

sentiment=sentiment\_by(tweets$text)

sentiment

## element\_id word\_count sd ave\_sentiment  
## 1: 1 26 NA 0.156892908  
## 2: 2 53 0.15843836 0.115808624  
## 3: 3 23 0.13631132 0.024217877  
## 4: 4 31 0.10022297 -0.077343767  
## 5: 5 53 0.43707986 -0.258562863  
## ---   
## 71513: 71513 41 0.01778002 -0.013721142  
## 71514: 71514 40 0.01069045 -0.008250002  
## 71515: 71515 40 0.18945428 0.208326613  
## 71516: 71516 27 0.11785113 0.099662836  
## 71517: 71517 22 NA 0.298481003

summary(sentiment$ave\_sentiment)

## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## -2.20663 -0.02774 0.06421 0.06528 0.16599 2.09752

res = data.frame(sentiment$ave\_sentiment,tweets$created\_at)  
colnames(res)=c("sentiment","created\_at")  
head(res)

## sentiment created\_at  
## 1 0.15689291 2020-04-05 23:59:26  
## 2 0.11580862 2020-04-05 23:59:24  
## 3 0.02421788 2020-04-05 23:58:51  
## 4 -0.07734377 2020-04-05 23:58:05  
## 5 -0.25856286 2020-04-04 09:13:02  
## 6 0.10634020 2020-04-05 23:49:58

res = separate(res,"created\_at",c("date","time"),sep=" ")

head(res)

## sentiment date time  
## 1 0.15689291 2020-04-05 23:59:26  
## 2 0.11580862 2020-04-05 23:59:24  
## 3 0.02421788 2020-04-05 23:58:51  
## 4 -0.07734377 2020-04-05 23:58:05  
## 5 -0.25856286 2020-04-04 09:13:02  
## 6 0.10634020 2020-04-05 23:49:58

sec <- sqldf("select sentiment, date, time, count(sentiment) as count from res group by date")  
head(sec)

## sentiment date time count  
## 1 -0.084566720 2020-03-29 20:53:26 1246  
## 2 0.009764924 2020-03-30 12:35:54 3347  
## 3 -0.051447801 2020-03-31 14:09:00 6456  
## 4 0.106904497 2020-04-01 16:41:52 6288  
## 5 0.357221021 2020-04-02 00:45:43 6274  
## 6 0.298481003 2020-04-03 10:40:52 18519

sec1 <- sqldf("select sentiment, date, time, count(sentiment) as count\_neg2 from res where sentiment>=-2.20663 and sentiment<=-0.02774 group by date")  
#sum(sec1$count)  
#sec1$count = sec1$count/sum(sec1$count) \*100  
head(sec1)

## sentiment date time count\_neg2  
## 1 -0.08456672 2020-03-29 20:53:26 265  
## 2 -0.06933752 2020-03-30 16:38:11 720  
## 3 -0.05144780 2020-03-31 14:09:00 1328  
## 4 -0.09943297 2020-04-01 00:21:39 1403  
## 5 -0.03404914 2020-04-02 00:18:32 1522  
## 6 -0.04422719 2020-04-03 10:50:04 4874

sec2 <- sqldf("select sentiment, date, time, count(sentiment) as count\_neg1 from res where sentiment>-0.02774 and sentiment<=0.06421 group by date")  
#sum(sec2$count)  
#sec2$count = sec2$count/sum(sec2$count) \*100  
head(sec2)

## sentiment date time count\_neg1  
## 1 0.006617419 2020-03-29 21:55:05 380  
## 2 0.009764924 2020-03-30 12:35:54 953  
## 3 0.057878776 2020-03-31 00:00:12 1675  
## 4 0.051447801 2020-04-01 00:21:10 1616  
## 5 0.014469694 2020-04-02 00:05:03 1627  
## 6 -0.008250002 2020-04-03 10:41:37 4473

sec3 <- sqldf("select sentiment, date, time, count(sentiment) as count\_pos1 from res where sentiment>0.06421 and sentiment<=0.16599 group by date")  
#sum(sec3$count)  
#sec3$count = sec3$count/sum(sec3$count) \*100  
head(sec3)

## sentiment date time count\_pos1  
## 1 0.15633725 2020-03-29 09:41:34 275  
## 2 0.13258717 2020-03-30 13:56:17 762  
## 3 0.16370590 2020-03-31 00:08:19 2033  
## 4 0.10690450 2020-04-01 16:41:52 1905  
## 5 0.06821079 2020-04-02 00:08:00 1785  
## 6 0.09966284 2020-04-03 10:41:03 4446

sec4 <- sqldf("select sentiment, date, time, count(sentiment) as count\_pos2 from res where sentiment>0.16599 and sentiment<=2.09752 group by date")  
#sum(sec4$count)  
#sec4$count = sec4$count/sum(sec4$count) \*100  
head(sec4)

## sentiment date time count\_pos2  
## 1 0.1796423 2020-03-29 09:59:16 326  
## 2 0.3796346 2020-03-30 00:08:20 912  
## 3 0.2841352 2020-03-31 00:04:13 1420  
## 4 0.3408083 2020-04-01 00:00:05 1364  
## 5 0.3572210 2020-04-02 00:45:43 1340  
## 6 0.2984810 2020-04-03 10:40:52 4726

sec = cbind(sec,sec1$count\_neg2,sec2$count\_neg1,sec3$count\_pos1,sec4$count\_pos2)  
colnames(sec)[5]<-"neg2"  
colnames(sec)[6]<-"neg1"  
colnames(sec)[7]<-"pos1"  
colnames(sec)[8]<-"pos2"  
  
head(sec)

## sentiment date time count neg2 neg1 pos1 pos2  
## 1 -0.084566720 2020-03-29 20:53:26 1246 265 380 275 326  
## 2 0.009764924 2020-03-30 12:35:54 3347 720 953 762 912  
## 3 -0.051447801 2020-03-31 14:09:00 6456 1328 1675 2033 1420  
## 4 0.106904497 2020-04-01 16:41:52 6288 1403 1616 1905 1364  
## 5 0.357221021 2020-04-02 00:45:43 6274 1522 1627 1785 1340  
## 6 0.298481003 2020-04-03 10:40:52 18519 4874 4473 4446 4726

sec$neg2 = sec$neg2/sec$count \* 100  
sec$neg1 = sec$neg1/sec$count \* 100  
sec$pos1 = sec$pos1/sec$count \* 100  
sec$pos2 = sec$pos2/sec$count \* 100  
head(sec)

## sentiment date time count neg2 neg1 pos1  
## 1 -0.084566720 2020-03-29 20:53:26 1246 21.26806 30.49759 22.07063  
## 2 0.009764924 2020-03-30 12:35:54 3347 21.51180 28.47326 22.76666  
## 3 -0.051447801 2020-03-31 14:09:00 6456 20.57001 25.94486 31.49009  
## 4 0.106904497 2020-04-01 16:41:52 6288 22.31234 25.69975 30.29580  
## 5 0.357221021 2020-04-02 00:45:43 6274 24.25885 25.93242 28.45075  
## 6 0.298481003 2020-04-03 10:40:52 18519 26.31892 24.15357 24.00778  
## pos2  
## 1 26.16372  
## 2 27.24828  
## 3 21.99504  
## 4 21.69211  
## 5 21.35799  
## 6 25.51974

sec$date = factor(sec$date)  
head(sec)

## sentiment date time count neg2 neg1 pos1  
## 1 -0.084566720 2020-03-29 20:53:26 1246 21.26806 30.49759 22.07063  
## 2 0.009764924 2020-03-30 12:35:54 3347 21.51180 28.47326 22.76666  
## 3 -0.051447801 2020-03-31 14:09:00 6456 20.57001 25.94486 31.49009  
## 4 0.106904497 2020-04-01 16:41:52 6288 22.31234 25.69975 30.29580  
## 5 0.357221021 2020-04-02 00:45:43 6274 24.25885 25.93242 28.45075  
## 6 0.298481003 2020-04-03 10:40:52 18519 26.31892 24.15357 24.00778  
## pos2  
## 1 26.16372  
## 2 27.24828  
## 3 21.99504  
## 4 21.69211  
## 5 21.35799  
## 6 25.51974

ggplot(data=sec,aes(x=date,y=neg2,color="red",group=1))+geom\_line()+geom\_line(data = sec,aes(x=date,y=neg1,color="violet"))+geom\_line(data = sec,aes(x=date,y=pos1,color="blue"))+geom\_line(data = sec,aes(x=date,y=pos2,color="green"))

