library(rvest)

library(XML)

library(magrittr)

**#################### IMDB Reviews #############################**

aurl <- "https://www.imdb.com/title/tt4154796/reviews?ref\_=tt\_urv"

IMDB\_reviews <- NULL

for (i in 1:10){

murl <- read\_html(as.character(paste(aurl,i,sep="=")))

rev <- murl %>%

html\_nodes(".show-more\_\_control") %>%

html\_text()

IMDB\_reviews <- c(IMDB\_reviews,rev)

}

length(IMDB\_reviews)

#430

write.table(IMDB\_reviews,"endgame.txt",row.names = F)

getwd()

str(IMDB\_reviews)

#chr [1:430]

#"There is no way that I could describe my emotions for this movie.

#I'm totally speechless. I haven't laughed (ev"| \_\_truncated\_\_ ...

fin\_txt <- IMDB\_reviews

fin\_txt

library(tm)

fin\_corpus<- Corpus(VectorSource(fin\_txt))

inspect(fin\_corpus[100])

fin\_clean<-tm\_map(fin\_corpus, removePunctuation)

fin\_clean<-tm\_map(fin\_clean, content\_transformer(tolower))

fin\_clean<-tm\_map(fin\_clean, removeWords, stopwords("english"))

fin\_clean<-tm\_map(fin\_clean,removeNumbers)

fin\_clean<-tm\_map(fin\_clean, stripWhitespace)

fin\_clean<-tm\_map(fin\_clean, removeWords, c("gameofthrones")) ## clean some words

fin\_clean

#<<SimpleCorpus>>

# Metadata: corpus specific: 1, document level (indexed): 0

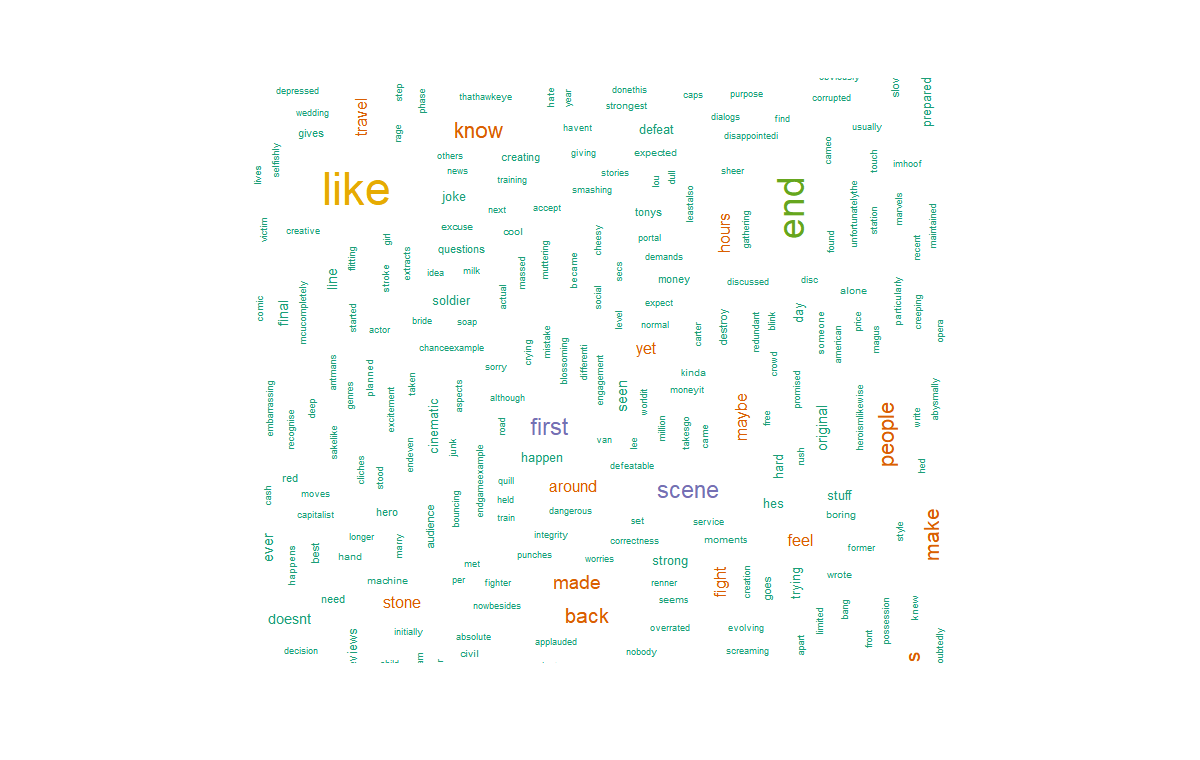
#Content: documents: 430

library(wordcloud)

wordcloud(fin\_clean, random.order = F, max.words = 5,colors=rainbow(50))



wordcloud(fin\_clean, rot.per=0.5, random.order=TRUE,colors=brewer.pal(8, "Dark2"))



**# Term document matrix**

**# converting unstructured data to structured format using TDM**

tdm <- TermDocumentMatrix(fin\_clean)

dtm <- t(tdm)

tdm <- as.matrix(tdm)

# Bar plot

w <- rowSums(tdm)

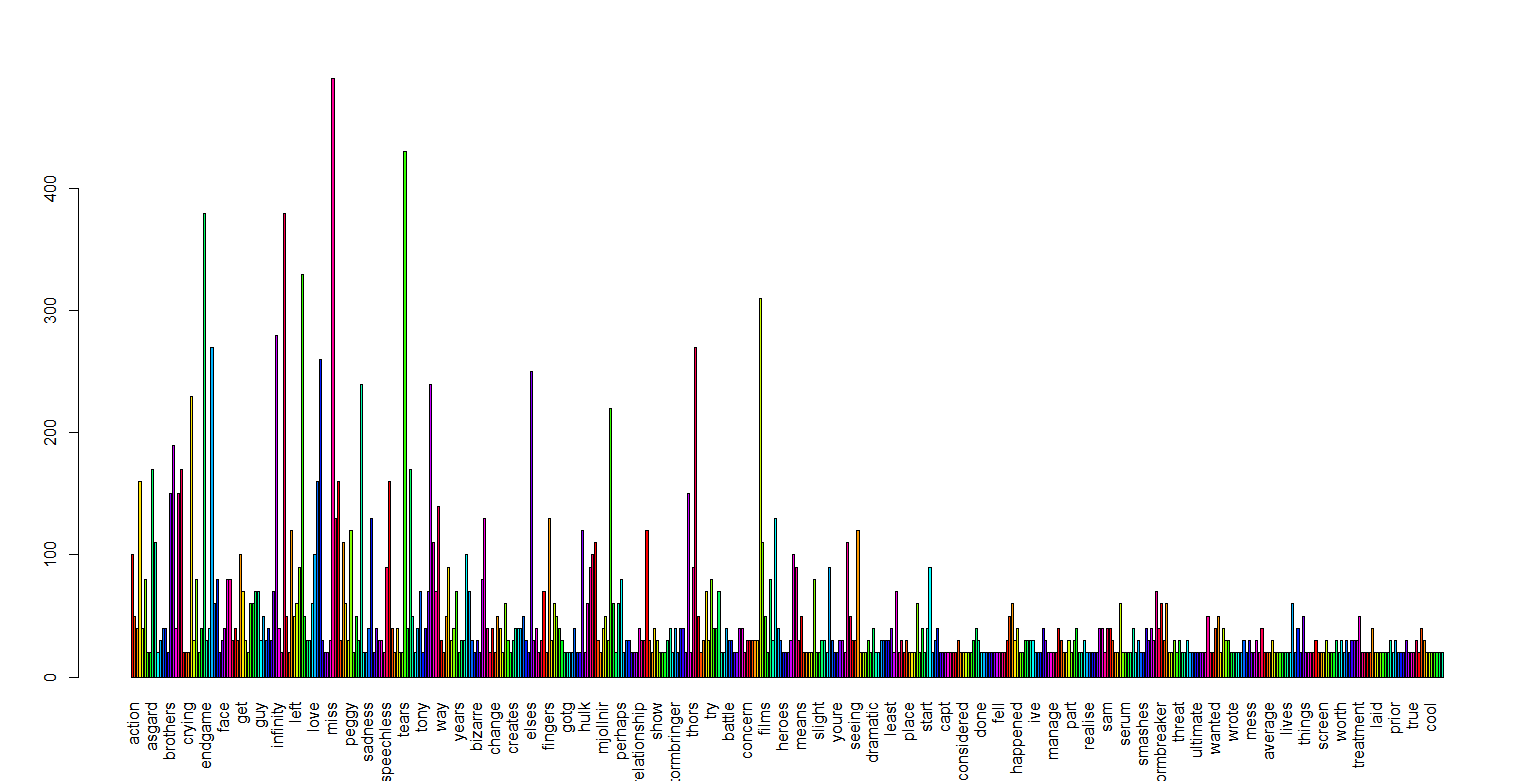
w

w\_sub <- subset(w, w >= 20)

w\_sub

windows()

barplot(w\_sub, las=3, col = rainbow(20))



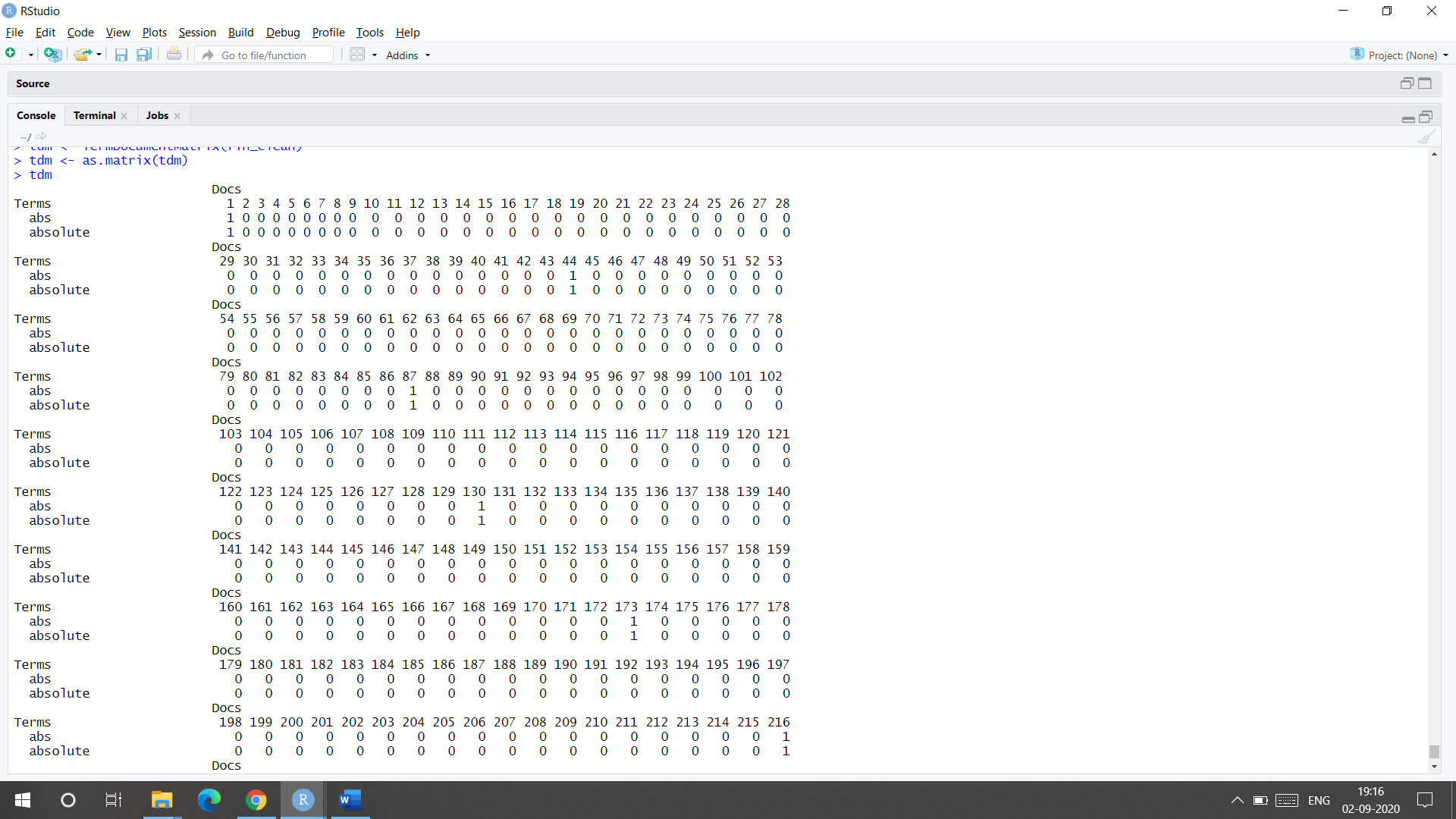
fin\_clean <- tm\_map(fin\_clean, removeWords, c('apple','air','can','mcds','macbook','product','windows'))

fin\_clean <- tm\_map(fin\_clean, stripWhitespace)

tdm <- TermDocumentMatrix(fin\_clean)

tdm <- as.matrix(tdm)

tdm



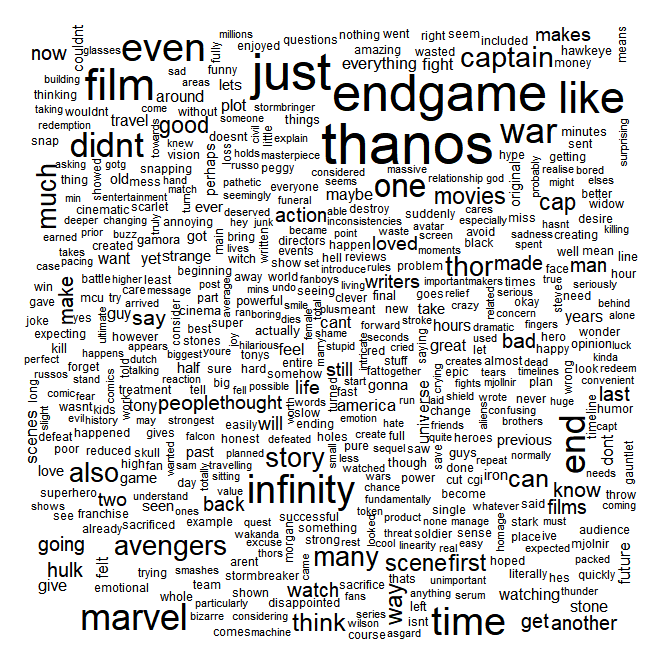
**# Word cloud**

#install.packages("wordcloud")

#library(wordcloud)

windows()

wordcloud(words = names(w\_sub), freq = w\_sub) # wordcloud with only subset of words

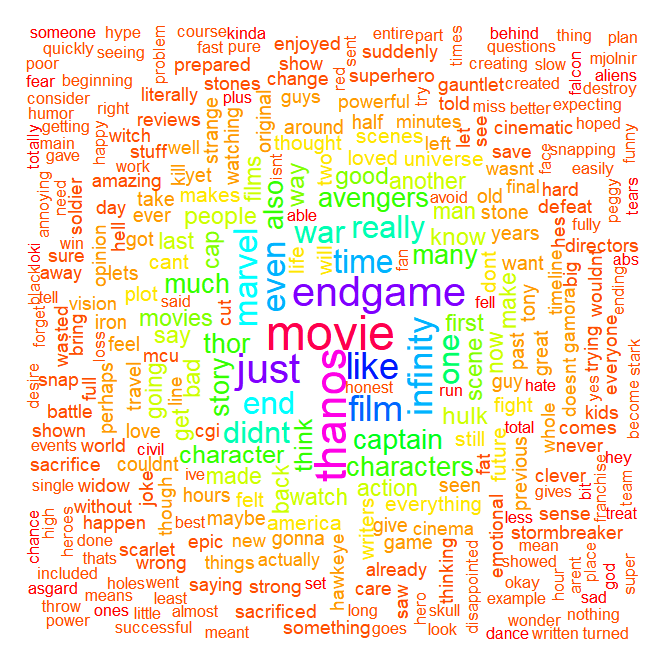


w\_sub1 <- sort(rowSums(tdm), decreasing = TRUE)

wordcloud(words = names(w\_sub1), freq = w\_sub1) # all words are considered

windows()

wordcloud(words = names(w\_sub1), freq = w\_sub1, random.order = F, colors = rainbow(20), scale=c(3,1), rot.per = 0.3)



**# lOADING +VE AND -VE dictonaries**

pos.words = scan(file.choose(), what="character", comment.char=";") # read-in positive-words.txt

#Read 2006 items

neg.words = scan(file.choose(), what="character", comment.char=";") # read-in negative-words.txt

#Read 4783 items

pos.words = c(pos.words,"wow", "kudos", "hurray") # including our own positive words to the existing list

# Positive wordcloud

pos.matches = match(names(w\_sub1), c(pos.words))

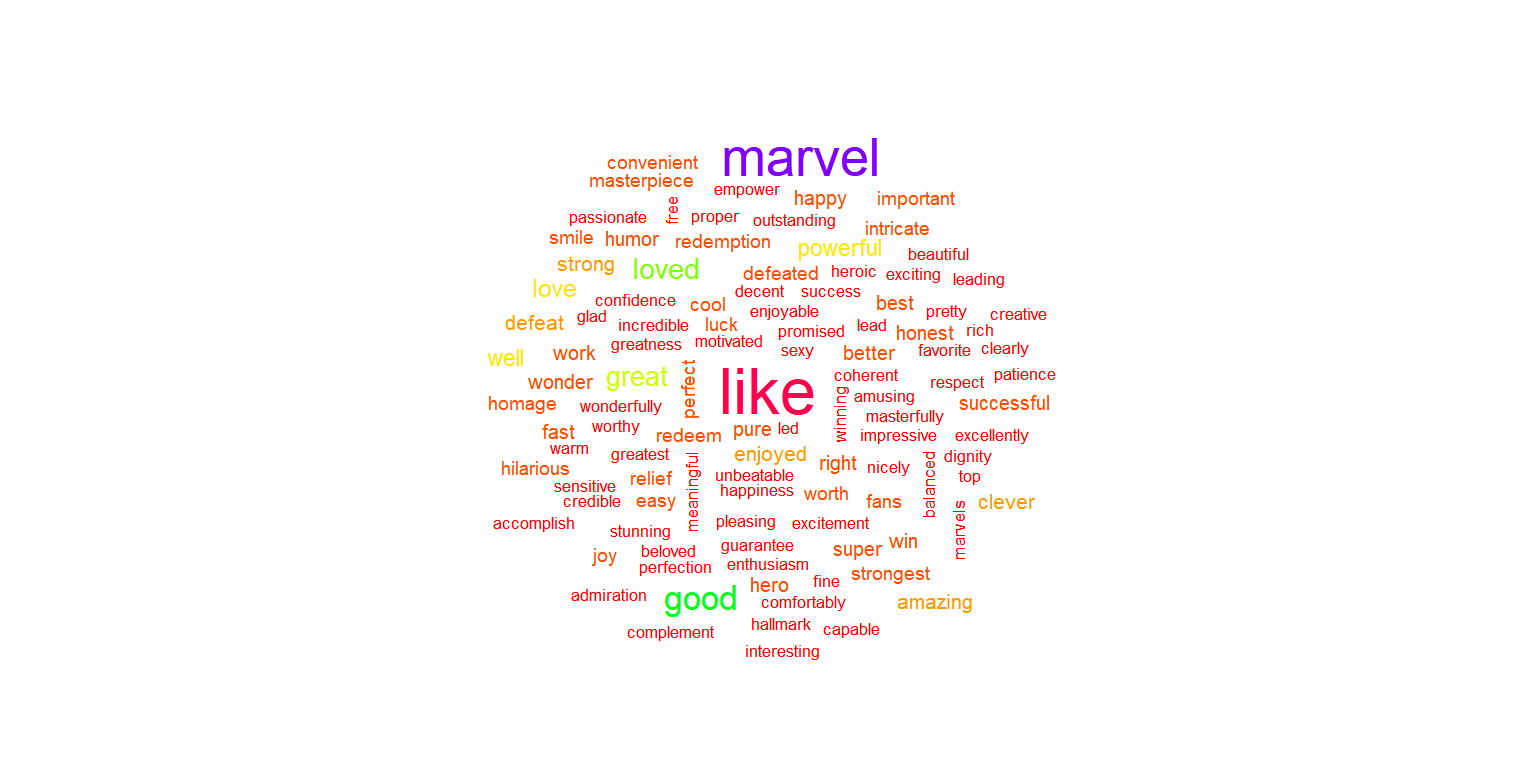
pos.matches = !is.na(pos.matches)

freq\_pos <- w\_sub1[pos.matches]

p\_names <- names(freq\_pos)

windows()

wordcloud(p\_names,freq\_pos,scale=c(4,1),colors = rainbow(20))



# Negative wordcloud

neg.matches = match(names(w\_sub1), c(neg.words))

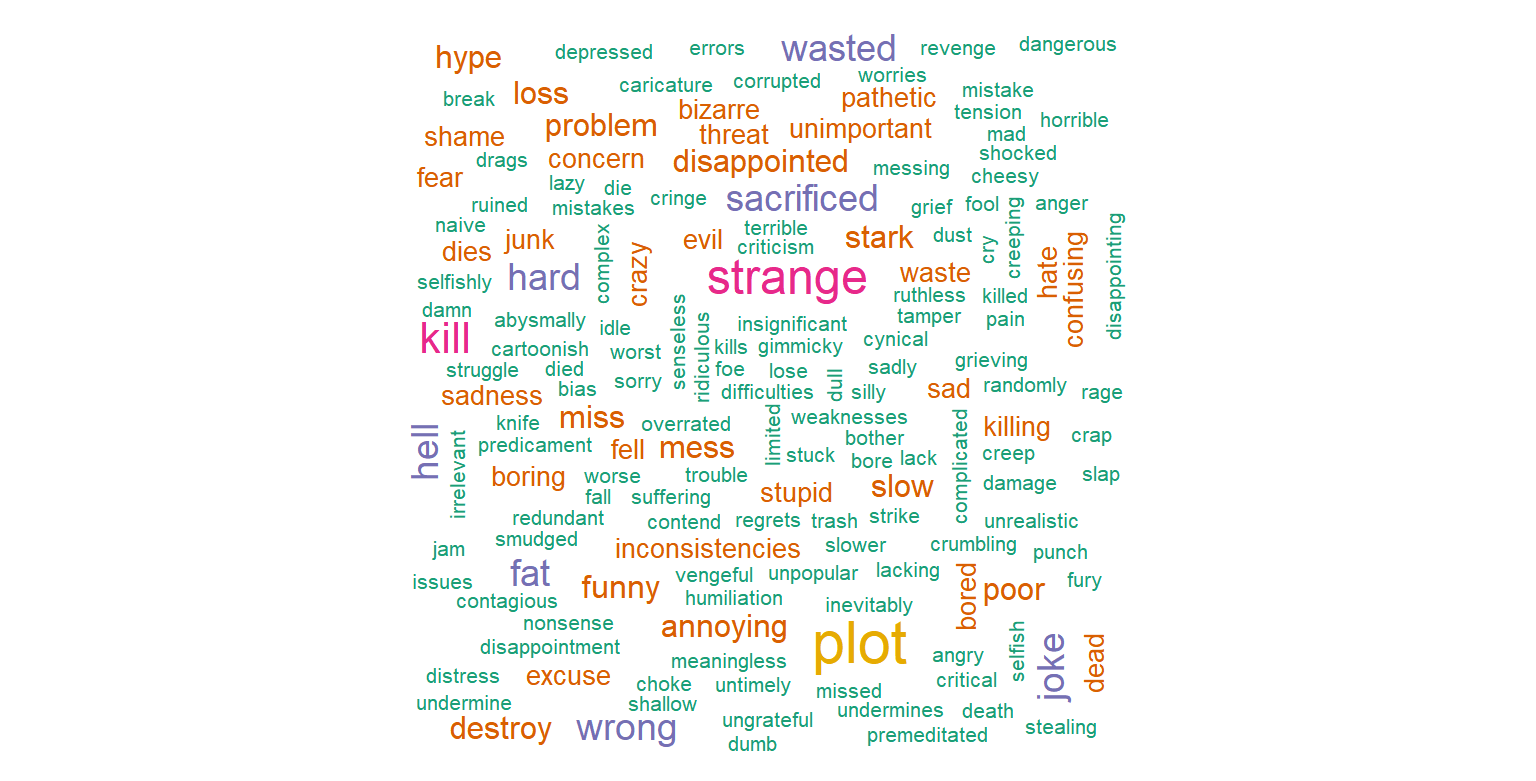
neg.matches = !is.na(neg.matches)

freq\_neg <- w\_sub1[neg.matches]

n\_names <- names(freq\_neg)

windows()

wordcloud(n\_names,freq\_neg,scale=c(5,1),colors = brewer.pal(8,"Dark2"))



**#### Emotion mining**

install.packages("syuzhet")

library("syuzhet")

library(lubridate,ggplot2)

library(ggplot2)

library(scales)

library(dplyr)

library(reshape2)

x <- get\_nrc\_sentiment(fin\_txt)

head(x,n=5)

#anger anticipation disgust fear joy sadness surprise trust negative positive

#1 10 16 8 12 20 11 10 16 17 32

#2 6 10 4 9 7 8 7 11 18 18

#3 4 10 4 7 6 8 5 11 11 20

#4 0 0 0 0 0 0 0 0 0 0

#5 0 0 0 0 0 0 0 0 0 0

fin\_txt[4]

get\_nrc\_sentiment('happy')

#anger anticipation disgust fear joy sadness surprise trust negative positive

#1 0 1 0 0 1 0 0 1 0 1

get\_nrc\_sentiment('boring')

#anger anticipation disgust fear joy sadness surprise trust negative positive

#1 0 0 0 0 0 0 0 0 1 0

get\_sentiment('boring',method="afinn")

#[1] -3

get\_sentiment('happy',method="afinn")

#[1] 3

#each sentences

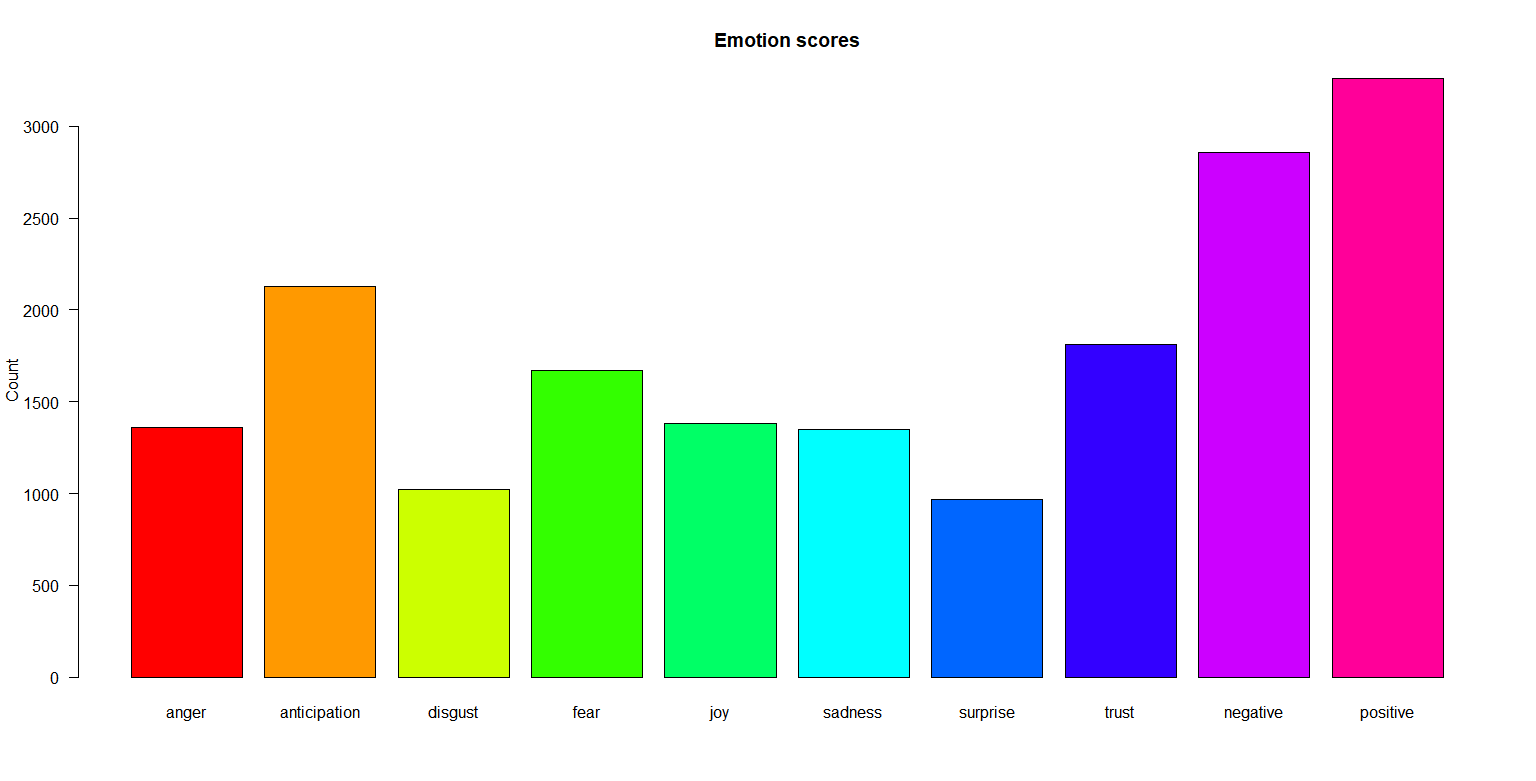
example<-get\_sentences(fin\_txt)

nrc\_data<-get\_nrc\_sentiment(example)

# Bar plot for emotion mining

windows()

barplot(colSums(nrc\_data), las = 1, col = rainbow(10), ylab = 'Count', main = 'Emotion scores')



sentiment\_vector<-get\_sentiment(example,method="bing")

sentiment\_afinn<-get\_sentiment(example,method="afinn")

sentiment\_nrc<-get\_sentiment(example,method="nrc")

sum(sentiment\_afinn) #[1] 1130

mean(sentiment\_afinn) #[1] 0.2783251

summary(sentiment\_afinn)

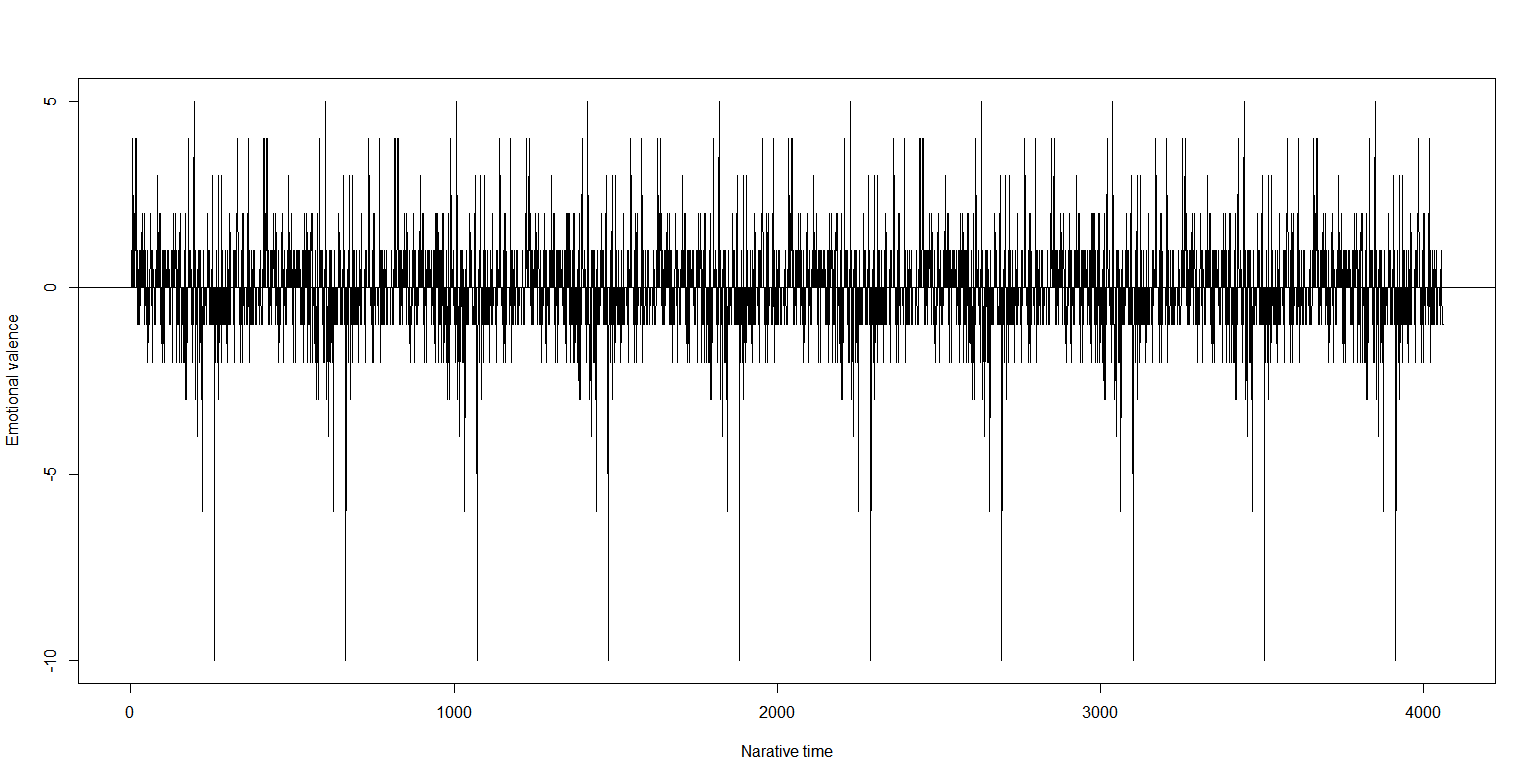
#Min. 1st Qu. Median Mean 3rd Qu. Max.

#-21.0000 -1.0000 0.0000 0.2783 1.0000 15.0000

windows()

plot(sentiment\_vector,type='l',maim='Plot trajectory',xlab='Narative time',ylab='Emotional valence')

abline(h=0,color='red')



plot(

sentiment\_vector,

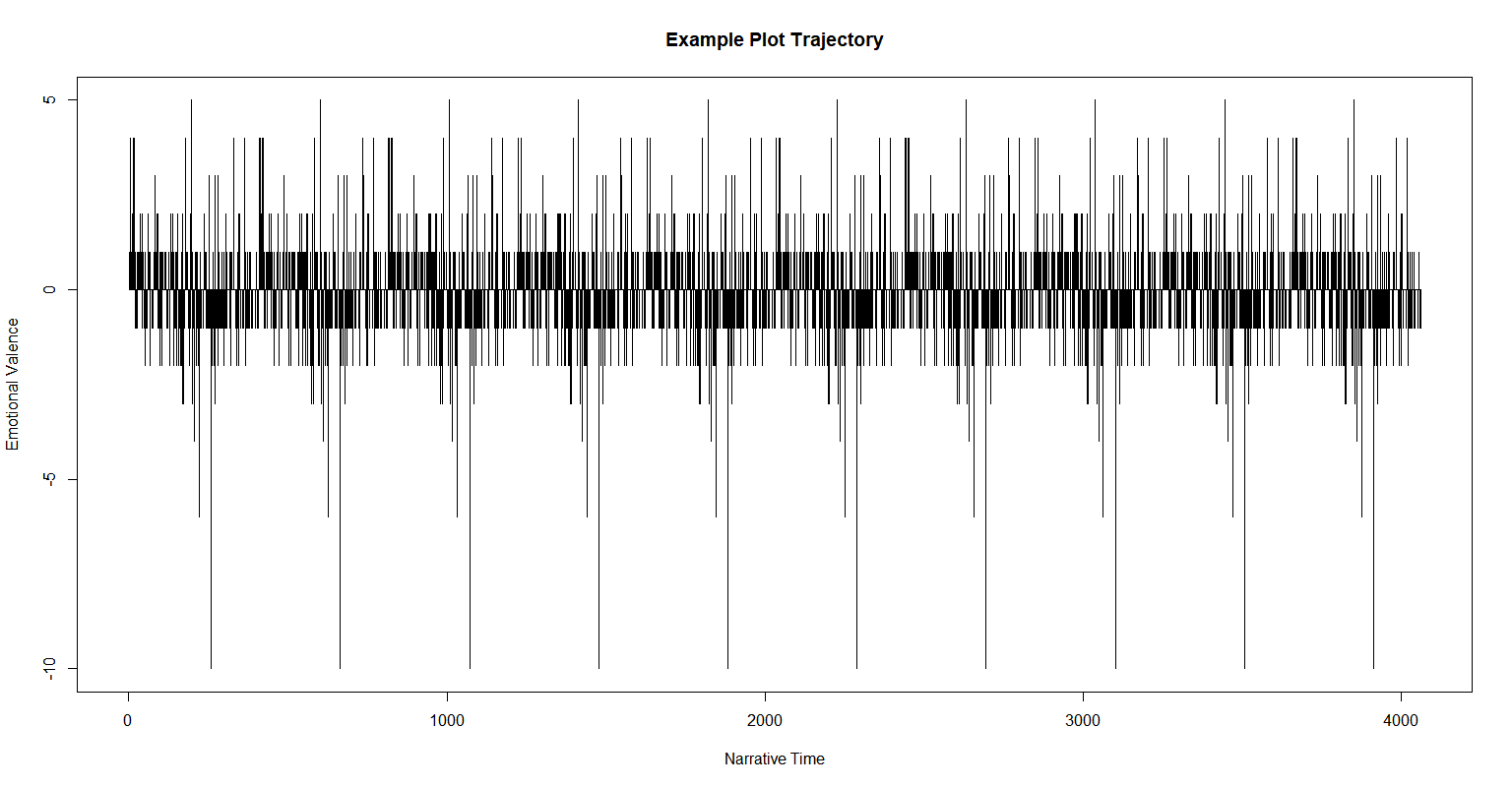
type="h",

main="Example Plot Trajectory",

xlab = "Narrative Time",

ylab= "Emotional Valence"

)



**##Shape smoothing and normalization using a Fourier based transformation and**

**##low pass filtering is achieved using the get\_transformed\_values function as shown below.**

ft\_values <- get\_transformed\_values(

sentiment\_vector,

low\_pass\_size = 3,

x\_reverse\_len = 100,

padding\_factor = 2,

scale\_vals = TRUE,

scale\_range = FALSE

)

plot(

ft\_values,

type ="l",

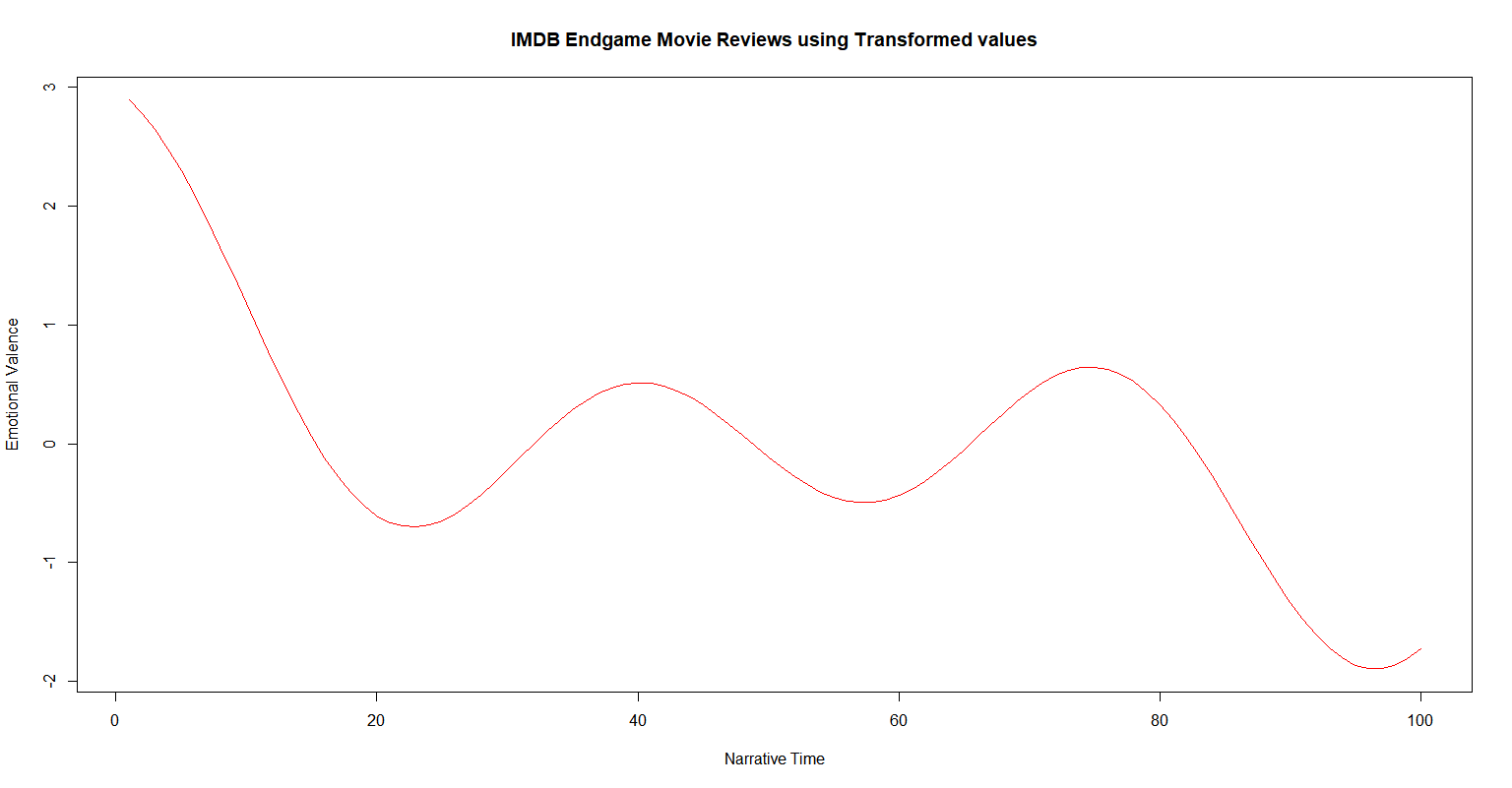
main ="IMDB Endgame Movie Reviews using Transformed values",

xlab = "Narrative Time",

ylab = "Emotional Valence",

col = "red"

)



**#Most Negative and Positive reviews**

negative<-example[which.min(sentiment\_vector)]

negative

#1] "Seriously how did this junk made so much money is beyond me are people really crazy they #don't know what they have done,this film is pure nonsense,and i hate to say this Thanos was a total #looser here also what happened to hulk its a shame people will realise this in future that what a #load of crap nonsense this was a total waste of time and money,it should have not even made 50 #bucks at box office that's how bad this is,Scarlet Johannson was sacrificed in this mess just so #people can shed

#a tear and throw money at the screen ,this makes me wonder the makers and people really #enjoyed killing her black widow and iron man for cash ,both of them are dead now.,besides that #this film is contagious cesspool of bad Cgi ,horrible cheesy Dialogs and Cartoonish action,and they #say this can match avatar 2009 avatar was a masterpiece holds great repeat value then this any #day,endgame holds no value as a film at all forget about repeat value,how is this a film it does not #even qualifies for direct to Disc DVD Or hallmark movie,i feel like this is a bad joke its so bad that #you cannot even tell it to yourself alone my rating is 1/10."

positive<-example[which.max(sentiment\_vector)]

print(positive)

#[1] "To be honest, the 'rules of engagement' had been set so well in IW that when in Endgame Thanos massed his ranks against the Avengers and then ALL of the good guys showed up, I was almost expecting the camera to pull away and not show the fight given that it was so obviously going to be an easy win for the good guys."