#先讀入 IMDb_Feature Film_2022_review_data 檔案並用 Title 找到想要觀察的電影，我選擇雷神(Thor: Love and Thunder)和德州電鋸殺人狂(Texas Chainsaw Massacre)
#引入 tm 套件
#Corpus()建語料庫
#tm_map()將評論內容轉小寫並移除標點符號
#stopwords()將無關文字情緒的連接詞移除
#將評論轉成 matrix 形式然後篩選出前十個出現最多次的評論字眼
#探討評論：兩者都有 movi、film、charact、good、like，應該是電影情節和角色討論分享相關。其中雷神的 thor、love、marvel 和德州電鋸殺人狂的 chainsaw 則對應到電影本身
#文字雲：利用 wordcloud2()實現（見 Code 中附上的文字雲圖）
#引入 tidytext 套件，用 get_sentimental()獲取文字情緒並用 bing 幫助判斷情緒正負面
#比較不同滿意指標的評論文字情緒：由 table(bing_word_counts$sentiment)可得到兩部電影評論中正、負面情緒字眼出現的次數，雖然 bing 判斷兩者評論的負面情緒都比正面多，但可看出德州電鋸殺人狂負評字眼數量相對正評字眼高出許多，又雷神 Rates(7.1)比德州電鋸殺人狂 Rates(4.8)高，比較這兩部電影可推測 Rates 較低的電影評論裡有較高比例的負面情緒字眼

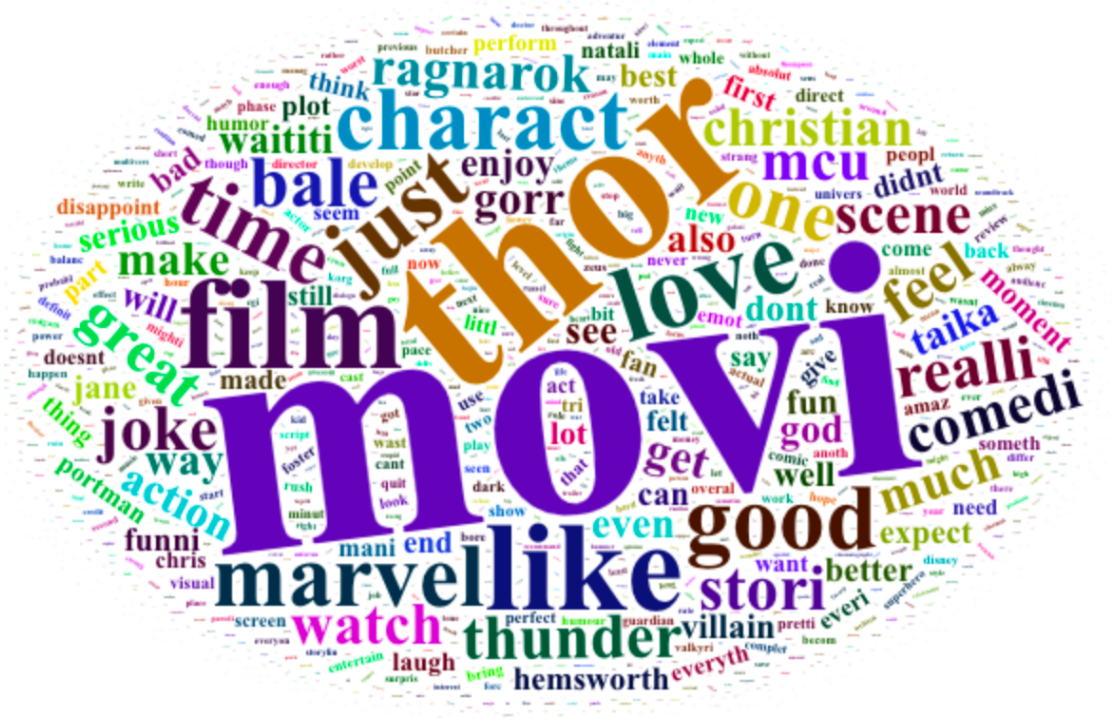
Thor: Love and Thunder(Rates=7.1)
Code:

```
film <- read.csv("IMDb_Feature Film_2022_review_data.csv")
film1=film[which(film$Title=="Thor: Love and Thunder"),]

library(tm)
x1 <- Corpus(VectorSource(film1$Review))#建語料庫
x1 <- tm_map(x1,tolower)
x1 <- tm_map(x1,content_transformer(tolower))
x1 <- tm_map(x1,removePunctuation)
x1StopWords <- c(stopwords(),"the","and","this","that","was","but","for")
x1 <- tm_map(x1,removeWords,x1StopWords)
library(SnowballC)
x1 <- tm_map(x1,stemDocument)

x1tdm <- TermDocumentMatrix(x1)
inspect(x1tdm)
```

```
x1review <- as.matrix(x1tdm)
x1freq <- rowSums(x1review)
x1freq <- sort(x1freq, decreasing=T)
x1freq[1:10]
```

```
> x1freq[1:10]
   movi    thor    film    like    love    just    good charact  marvel    time
   1092     695     422     407     346     301     299     291     290     270
```

```
barplot(x1freq[1:10],las=2,col="blue")
library(wordcloud2)
x1freqframe <- data.frame(word=names(x1freq),num=x1freq)
wordcloud2(x1freqframe,size=1)
```




```
library(tidytext)
library(tidyverse)
bing_word_counts1 <- x1freqframe %>%
   inner_join(get_sentiments("bing")) %>%
   count(word,sentiment,sort=TRUE)
table(bing_word_counts1$sentiment)
```

```
> table(bing_word_counts1$sentiment)

negative positive
     293      178
```

Texas Chainsaw Massacre(Rates=4.8)

Code:

```
film2=film[which(film$Title=="Texas Chainsaw Massacre"),]

library(tm)
x2 <- Corpus(VectorSource(film2$Review))#建語料庫
x2 <- tm_map(x2,tolower)
x2 <- tm_map(x2,content_transformer(tolower))
x2 <- tm_map(x2,removePunctuation)
x2StopWords <- c(stopwords(),"the","and","this","that","was","but","for")
x2 <- tm_map(x2,removeWords,x2StopWords)
library(SnowballC)
x2 <- tm_map(x2,stemDocument)

x2tdm <- TermDocumentMatrix(x2)
inspect(x2tdm)

x2review <- as.matrix(x2tdm)
x2freq <- rowSums(x2review)
x2freq <- sort(x2freq, decreasing=T)
x2freq[1:10]
```

```
> x2freq[1:10]
    movi     film     like  charact     just      one    watch chainsaw
    1458      817      573      568      565      493      490      482
  origin     good
     459      434
```

```
barplot(x2freq[1:10],las=2,col="blue")
library(wordcloud2)
x2freqframe <- data.frame(word=names(x2freq),num=x2freq)
wordcloud2(x2freqframe,size=1)
```


```
library(tidytext)
library(tidyverse)
bing_word_counts2 <- x2freqframe %>%
   inner_join(get_sentiments("bing")) %>%
   count(word,sentiment,sort=TRUE)
table(bing_word_counts2$sentiment)
```

```
> table(bing_word_counts2$sentiment)

negative positive
     395      192
```