615 Assignment 4 Task 3

Boyu Chen

12/6/2021

```
 # Get the function from source
source("Book2TN-v6A-1.R")
book<-gutenberg_download(105)
```

```
## Determining mirror for Project Gutenberg from http://www.gutenberg.org/robot/harvest
```

```
## Using mirror http://aleph.gutenberg.org
```

```
# output book into txt file.
# write.table(book,"book.txt")
newbook <- book %>%
  mutate(
    linenumber = row_number(),
    chapter = cumsum(str_detect(text,
                                regex("^chapter [\\divxlc]",
                                      ignore_case = TRUE)))) %>%
  unnest_tokens(word, text)


 # rename the book into tidy_book
tidy_books <- book
 # read txt table
book_fix <- read.table("book.txt", header = T)
 # Upload book to server
# tnBooksFromLines(book_fix$text, "Jane_Austen/persuation")
# query from tnum server
TQ5<- tnum.query('Jane_Austen/persuation/section# has text',max=70000)
```

```
## Returned 1 thru 2877 of 2877 results
```

```
# get the tnum method of words
DF5 <- tnum.objectsToDf(TQ5)
# get the table for book
DF5 %>% dplyr::select(subject:numeric.value)%>% head()
```

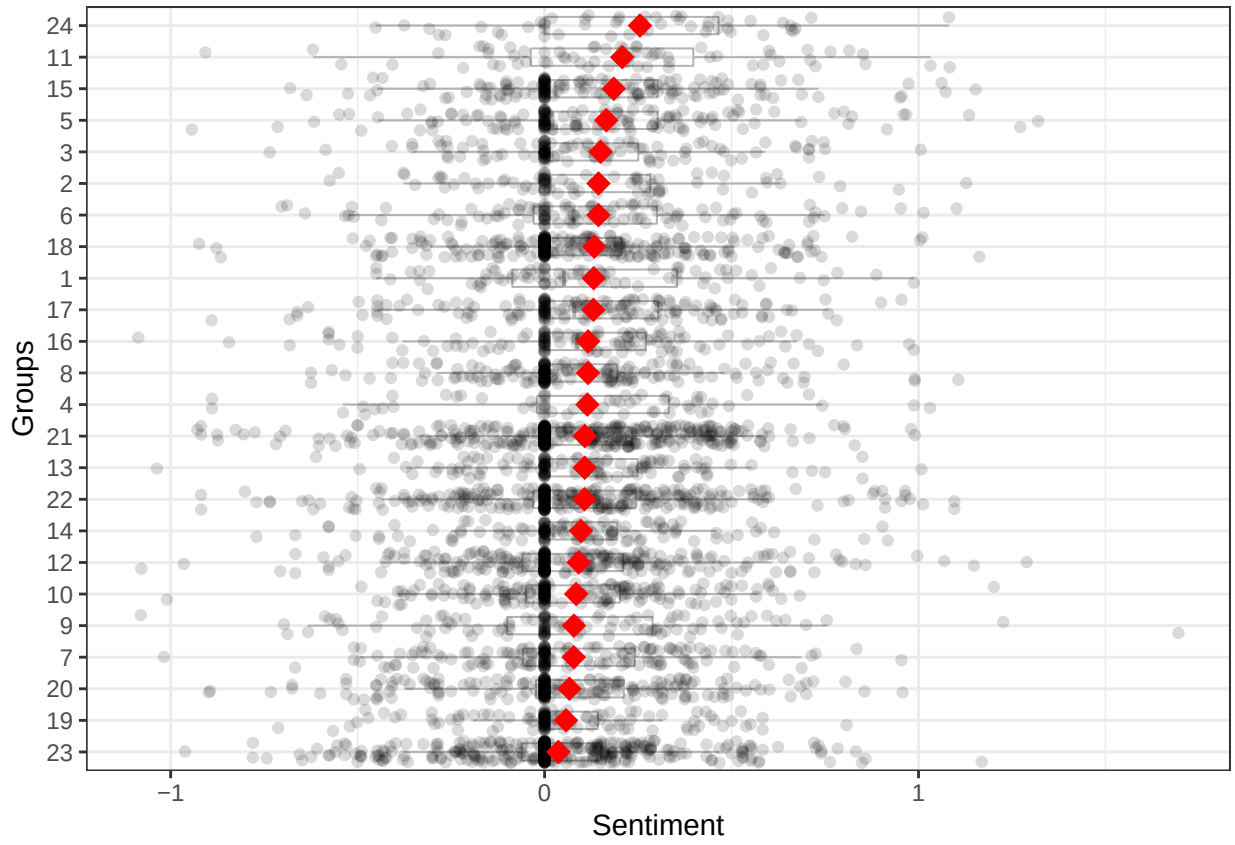
```
##                                                        subject property
## 1 jane_austen/persuation/section:0001/paragraph:0001/sentence:0001     text
## 2 jane_austen/persuation/section:0001/paragraph:0002/sentence:0001     text
## 3 jane_austen/persuation/section:0001/paragraph:0004/sentence:0001     text
```

```
## 4 jane_austen/persuation/section:0001/paragraph:0007/sentence:0001     text
## 5 jane_austen/persuation/section:0001/paragraph:0007/sentence:0002     text
## 6 jane_austen/persuation/section:0001/paragraph:0007/sentence:0003     text
##
## 1
## 2
## 3                                                                 ""Walter Elliot, born March 1, 1760, marri
## 4 "Then followed the history and rise of the ancient and respectable family, in the usual terms; how
## 5
## 6
##   numeric.value
## 1            NA
## 2            NA
## 3            NA
## 4            NA
## 5            NA
## 6            NA
```

```
 # Select sentence from book in subject,section,value method.
book_sentence<-DF5 %>% separate(col=subject,
                  into = c("path1", "path2","section",
                           "paragraph","sentence"),
                  sep = "/",
                  fill = "right") %>%
  dplyr::select(section:string.value)
book_sentence<-book_sentence %>%
  mutate_at(c('section','paragraph','sentence'),~str_extract_all(.,"\\d+") %>%
              unlist() %>% as.numeric())
```

Compare with Task2

```
# I use sentimentr to get sentiment score group by thee
# scores with each section to get the average result
sentence_out <- book_sentence %>%
  dplyr::mutate(sentence_split = get_sentences(string.value)) %$%
    sentiment_by(sentence_split, list(section))
# And plot them
plot(sentence_out)
```

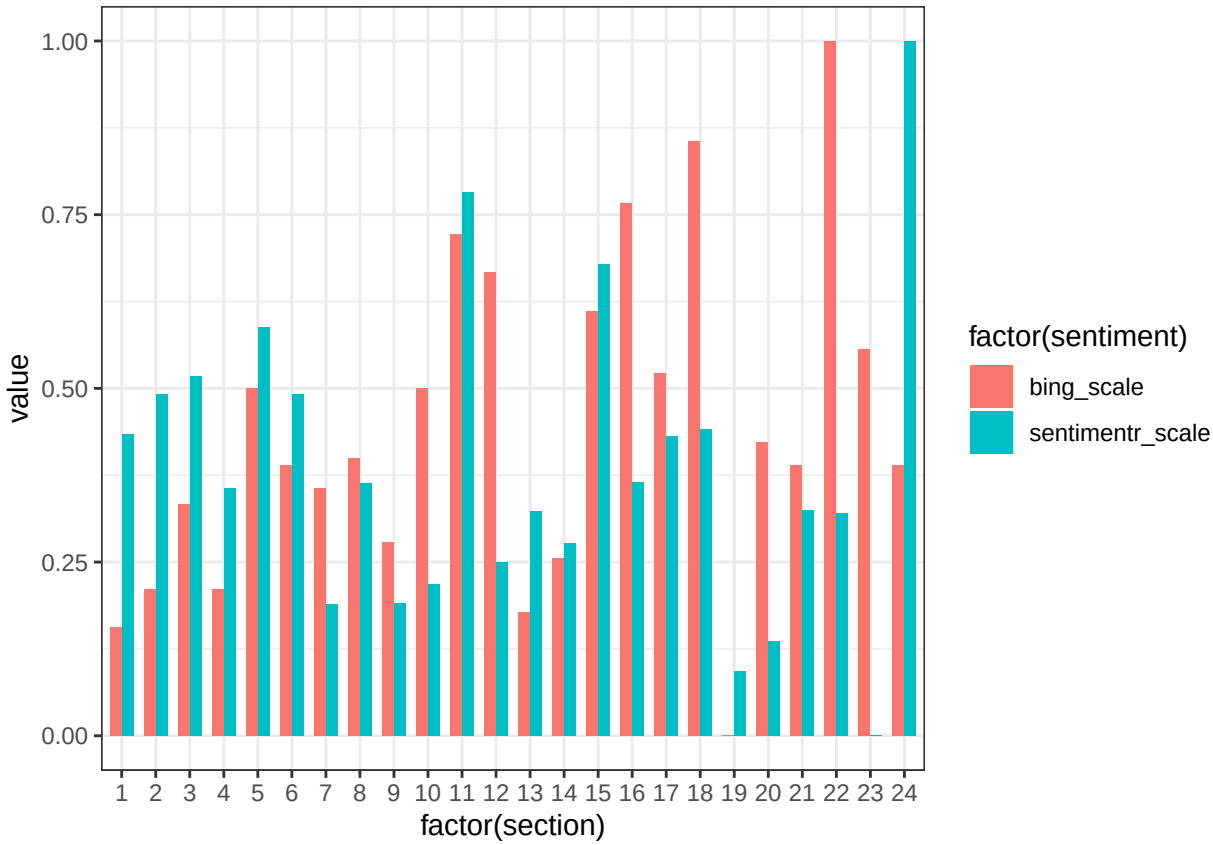



```
# create a new bing with index=chapter
new_bing <- newbook %>%
    inner_join(get_sentiments("bing")) %>%
    mutate(method = "Bing et al.") %>%
    count(method, index = chapter, sentiment) %>%
  pivot_wider(names_from = sentiment,
              values_from = n,
              values_fill = 0) %>%
  mutate(sentiment = positive - negative)
```

```
## Joining, by = "word"
```

```
# Creat zscore function
zscore<-function(input){
  return((input-min(input))/(max(input)-min(input)))

}
# get the zscore for sentence and word in bing dictionary
new_bing2 <- new_bing %>%
  mutate(bing_scale = zscore(sentiment)) %>%
  dplyr::select(method, index, bing_scale)

# Rename colname in order to join by section
colnames(new_bing2)[2]='section'
```

```r
# get the zscore for sentence_out
sentence_out <- sentence_out %>%
  mutate(sentimentr_scale = zscore(ave_sentiment))

# join two data frame together
sentence_out_2method <- left_join(sentence_out,new_bing2,by='section')%>%
  dplyr::select(section,bing_scale,sentimentr_scale)

# use pivot longer for ggplot
sentence_out_2method_plot <- sentence_out_2method %>%
  pivot_longer(cols = c('sentimentr_scale','bing_scale'),
               names_to = 'sentiment')

# create barplot to compare
sentence_out_2method_plot %>%
  ggplot(aes(y = value,x = factor(section))) +
  geom_bar(aes(fill = factor(sentiment)),
           stat = 'identity', position = "dodge",width = 0.7) +
  theme_bw()
```



The graph shows the difference between bing zscore and sentiment zscore. We can find that the trend are similar however, the difference between zscore is much larger.In some of the chapter, some words through bing dictionary are more optimistic than sentimentr. However, in some chapter some words through bing dictionary are pessimistic. I think sentiment is much better

```
# Form book character to find two main role
book_sentence_ch <- book_sentence %>%
  mutate(Anne=str_match(book_sentence$string.value,regex('([Aa]nne)'))[,1],
         Wentworth=str_match(book_sentence$string.value,regex('([Ww]entworth)'))[,1])
```

```
# Use sentiment_by to get the score
sentence_score <- book_sentence_ch %>%
  dplyr::mutate(book_split = get_sentences(string.value)) %$%
  sentiment_by(book_split) %>% `$`(ave_sentiment)
```

```
# Count two characters' number of appearance in each chapter
book_sentence_ch$score <- sentence_score
table1 <- book_sentence_ch %>% group_by(section) %>% summarise(Anne = sum(Anne %>% is.na() %>% `!`()),
                                                               Wentworth = sum(Wentworth%>% is.na() %>% `!`()))
knitr::kable(table1,'simple')
```

section	Anne	Wentworth
1	6	0
2	14	0
3	5	5
4	11	4
5	29	0
6	20	5
7	22	16
8	13	14
9	11	25
10	25	14
11	13	18
12	44	22
13	20	4
14	20	3
15	15	0
16	25	0
17	32	0
18	19	7
19	18	11
20	27	13
21	41	2
22	38	11
23	29	15
24	14	10

This table is the appearnace of two characters appear in the same paragraphs:

```
# use group by to display the result
table2 <- book_sentence_ch %>% group_by(section, paragraph) %>%
  summarise(both = sum(Anne%>% is.na() %>% `!`() & Wentworth%>% is.na() %>% `!`() ))
```

```
## `summarise()` has grouped output by 'section'. You can override using the `.groups` argument.
```

```
knitr::kable(table2 %>% filter(both > 0),'simple')
```

section	paragraph	both
3	30	1
4	1	1
6	33	1
7	22	1
7	24	1
7	32	1
8	1	1
8	28	1
8	29	1
9	4	1
9	8	1
9	9	1
9	16	1
9	21	1
9	23	1
9	25	1
10	1	1
10	20	1
10	22	1
10	33	1
10	34	1
10	36	1
11	7	1
11	15	1
11	16	1
11	21	1
11	23	1
12	5	1
12	11	1
12	19	1
12	39	1
12	43	1
12	44	1
12	54	1
12	68	1
12	73	1
13	20	1
13	32	1
14	25	1
18	29	1
18	51	1

section	paragraph	both
18	55	1
19	1	1
19	4	1
19	10	1
20	35	1
20	41	1
22	34	1
22	50	1
22	56	1
22	67	1
23	37	1
23	55	1
24	1	1
24	3	2
24	5	2
24	10	1
24	12	1

Difference between my code and Jin's code is that due to my book have much more positive words than negative words, I use zscore instead of scale to compare. According to zscore, I can find the similar emotion trend. Without zscore, I can only found the different trend for emotion for each section.

Reference: # 1. https://www.gutenberg.org/ebooks/105

2. Code are reference from Yuli Jin.Github:https://github.com/MA615-Yuli/MA615_assignment4_new

4. Code are reference from https://www.tidytextmining.com/sentiment.html