607_HW10_NCollin

Noah Collin

10/31/2021

Homework 10

The following code was copied from TidyTextMining.com, originally posted here: https://www.tidytextmining.com/sentiment.html

```
#install.packages("tidytext")
#install.packages("textdata")

library(tidytext)
library(stringr)
```

The following sentiments are from here: AFINN : https://www2.imm.dtu.dk/pubdb/pubs/6010-full.html BING: https://www.cs.uic.edu/~liub/FBS/sentiment-analysis.html NRC : https://saifmohammad.com/WebPages/NRC-Emotion-Lexicon.htm

```
afinn <-  (get_sentiments("afinn"))
bing <- get_sentiments("bing")
nrc <- get_sentiments("nrc")
```

Source for the following code: https://www.tidytextmining.com/sentiment.html

```
library(janeaustenr)
library(dplyr)
```

```
##
## Attaching package: 'dplyr'

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

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

```
library(stringr)

tidy_books <- austen_books() %>%
  group_by(book) %>%
```

```
  mutate(
    linenumber = row_number(),
    chapter = cumsum(str_detect(text,
                                regex("^chapter [\\divxlc]",
                                      ignore_case = TRUE)))) %>%
  ungroup() %>%
  unnest_tokens(word, text)
```

```
nrc_joy <- get_sentiments("nrc") %>%
  filter(sentiment == "joy")

tidy_books %>%
  filter(book == "Emma") %>%
  inner_join(nrc_joy) %>%
  count(word, sort = TRUE)
```

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

```
## # A tibble: 301 x 2
##    word          n
##    <chr>     <int>
##  1 good        359
##  2 friend      166
##  3 hope        143
##  4 happy       125
##  5 love        117
##  6 deal         92
##  7 found        92
##  8 present      89
##  9 kind         82
## 10 happiness    76
## # ... with 291 more rows
```

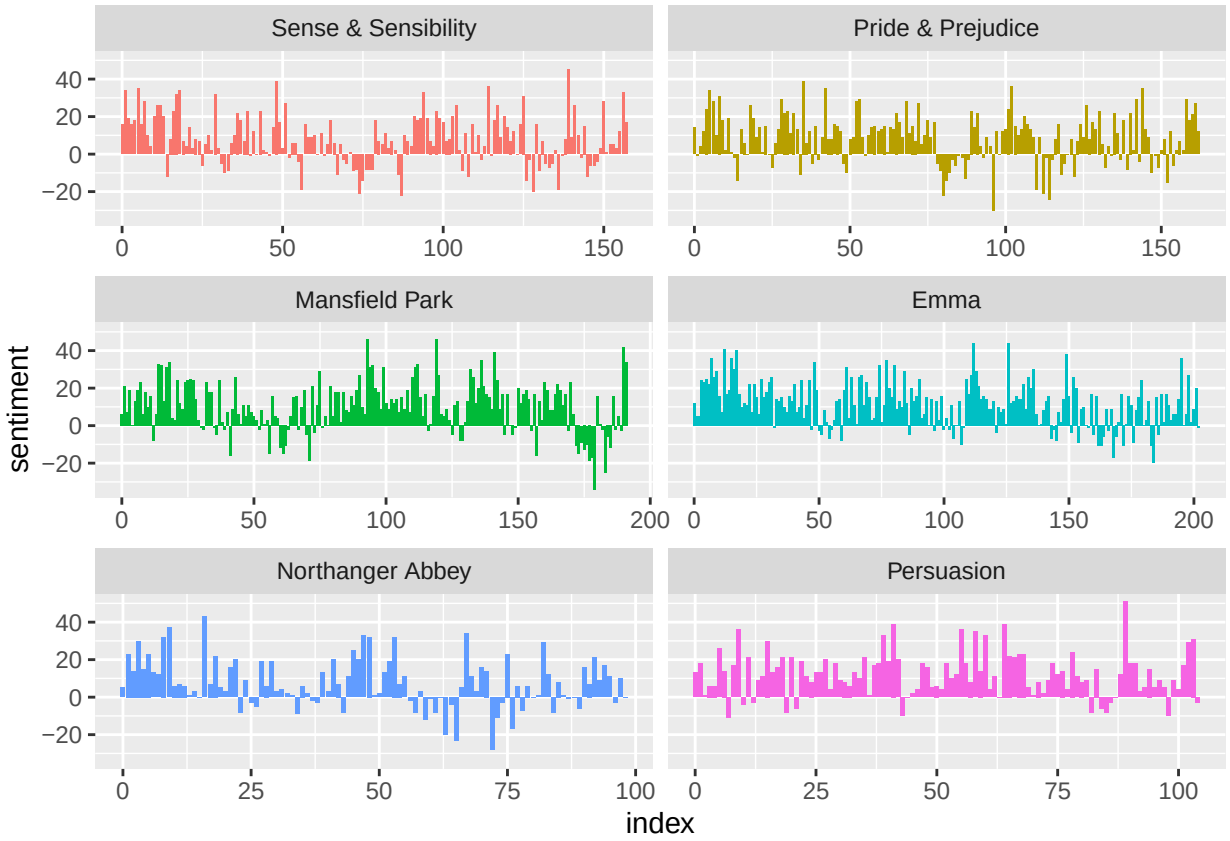
```
library(tidyr)

jane_austen_sentiment <- tidy_books %>%
  inner_join(get_sentiments("bing")) %>%
  count(book, index = linenumber %/% 80, sentiment) %>%
  pivot_wider(names_from = sentiment, values_from = n, values_fill = 0) %>%
  mutate(sentiment = positive - negative)
```

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

```
library(ggplot2)

ggplot(jane_austen_sentiment, aes(index, sentiment, fill = book)) +
  geom_col(show.legend = FALSE) +
  facet_wrap(~book, ncol = 2, scales = "free_x")
```




```
pride_prejudice <- tidy_books %>%
  filter(book == "Pride & Prejudice")
pride_prejudice
```

```
## # A tibble: 122,204 x 4
##    book               linenumber chapter word
##    <fct>                   <int>   <int> <chr>
##  1 Pride & Prejudice           1       0 pride
##  2 Pride & Prejudice           1       0 and
##  3 Pride & Prejudice           1       0 prejudice
##  4 Pride & Prejudice           3       0 by
##  5 Pride & Prejudice           3       0 jane
##  6 Pride & Prejudice           3       0 austen
##  7 Pride & Prejudice           7       1 chapter
##  8 Pride & Prejudice           7       1 1
##  9 Pride & Prejudice          10       1 it
## 10 Pride & Prejudice          10       1 is
## # ... with 122,194 more rows
```

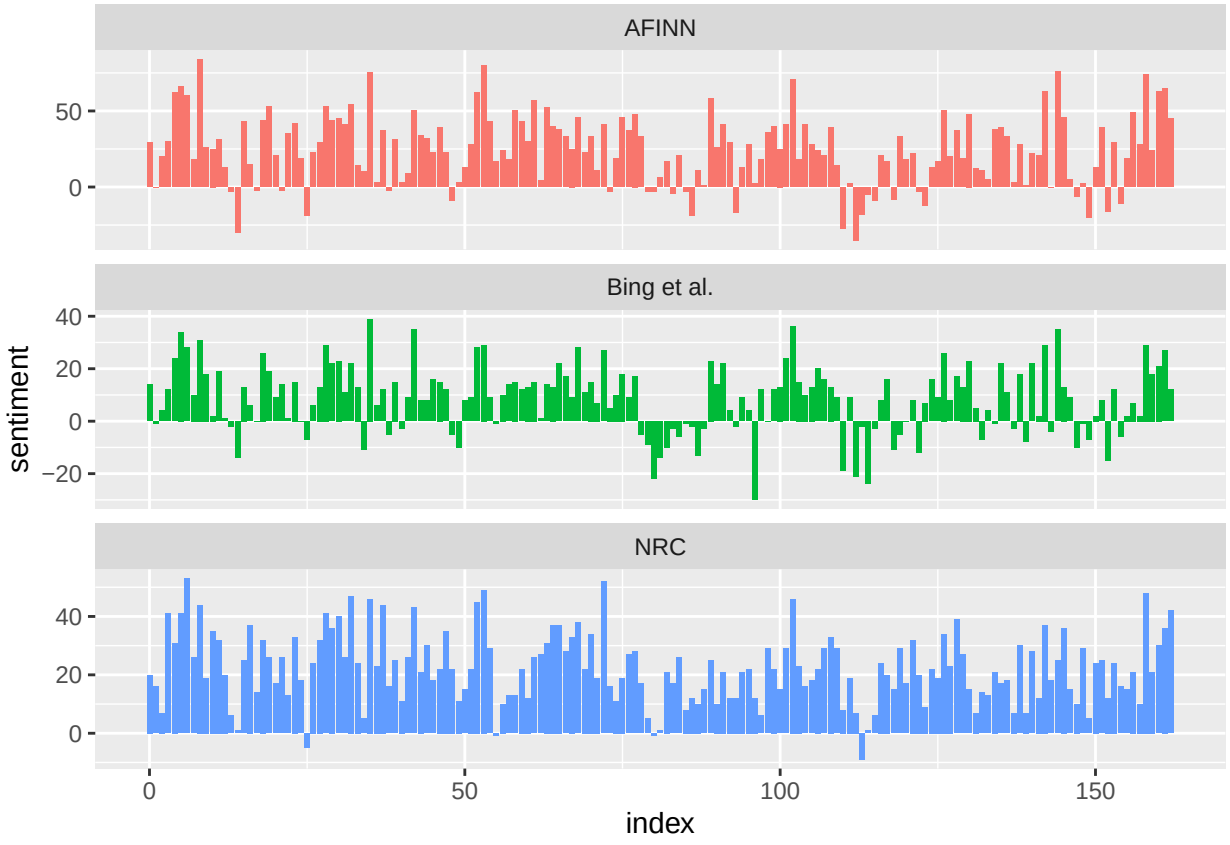
```
afinn <- pride_prejudice %>%
  inner_join(get_sentiments("afinn")) %>%
  group_by(index = linenumber %/% 80) %>%
  summarise(sentiment = sum(value)) %>%
  mutate(method = "AFINN")
```

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

```
bing_and_nrc <- bind_rows(
  pride_prejudice %>%
    inner_join(get_sentiments("bing")) %>%
    mutate(method = "Bing et al."),
  pride_prejudice %>%
    inner_join(get_sentiments("nrc") %>%
                 filter(sentiment %in% c("positive",
                                         "negative"))
    ) %>%
    mutate(method = "NRC")) %>%
  count(method, index = linenumber %/% 80, sentiment) %>%
  pivot_wider(names_from = sentiment,
              values_from = n,
              values_fill = 0) %>%
  mutate(sentiment = positive - negative)
```

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

```
bind_rows(afinn,
          bing_and_nrc) %>%
  ggplot(aes(index, sentiment, fill = method)) +
  geom_col(show.legend = F) +
  facet_wrap(~method, ncol = 1, scales = "free_y")
```



```
get_sentiments("nrc") %>%
  filter(sentiment %in% c("positive", "negative")) %>%
  count(sentiment)
```

```
## # A tibble: 2 x 2
##   sentiment     n
##   <chr>     <int>
## 1 negative   3318
## 2 positive   2308
```

```
get_sentiments("bing") %>%
  count(sentiment)
```

```
## # A tibble: 2 x 2
##   sentiment     n
##   <chr>     <int>
## 1 negative   4781
## 2 positive   2005
```

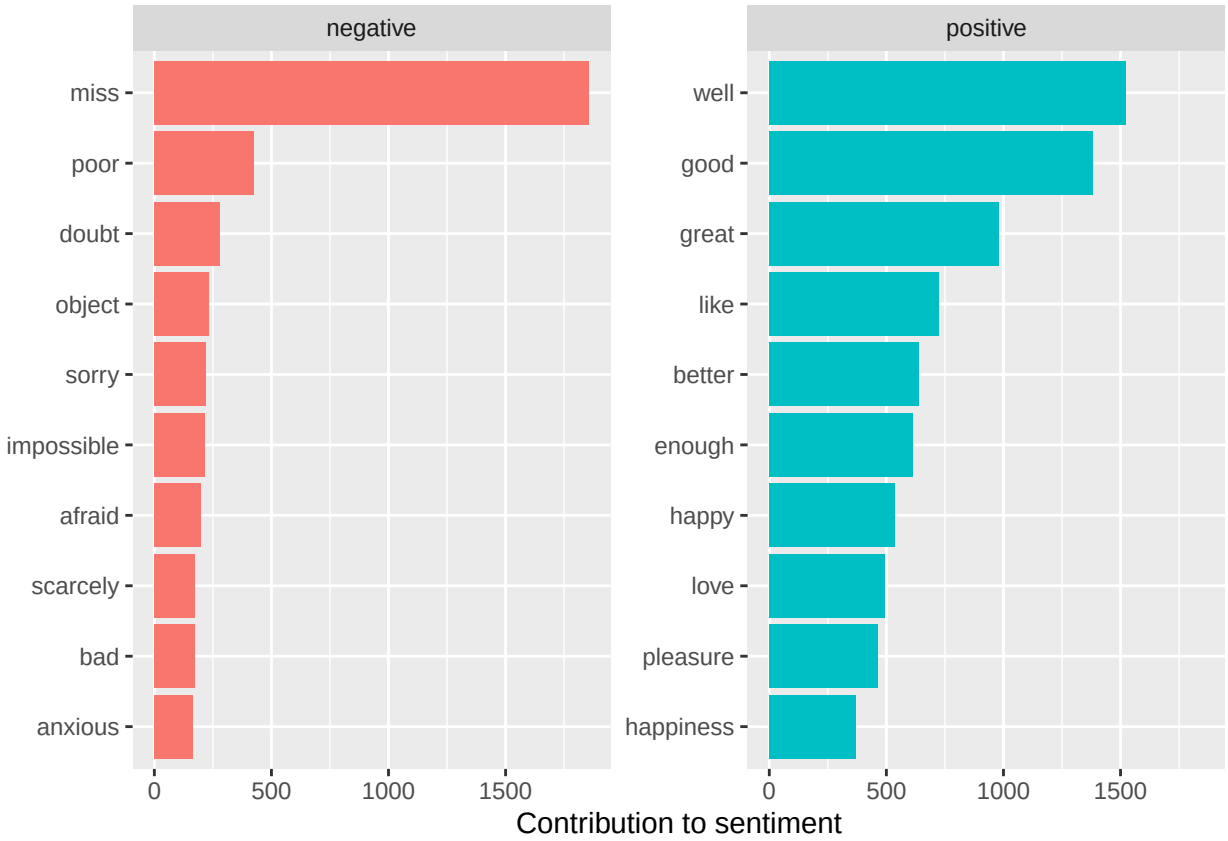
```
bing_word_counts <- tidy_books %>%
  inner_join(get_sentiments("bing")) %>%
  count(word, sentiment, sort = TRUE) %>%
  ungroup()
```

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

```
bing_word_counts
```

```
## # A tibble: 2,585 x 3
##    word      sentiment     n
##    <chr>     <chr>     <int>
##  1 miss      negative   1855
##  2 well      positive   1523
##  3 good      positive   1380
##  4 great     positive    981
##  5 like      positive    725
##  6 better    positive    639
##  7 enough    positive    613
##  8 happy     positive    534
##  9 love      positive    495
## 10 pleasure  positive    462
## # ... with 2,575 more rows
```

```
bing_word_counts %>%
  group_by(sentiment) %>%
  slice_max(n, n = 10) %>%
  ungroup() %>%
  mutate(word = reorder(word, n)) %>%
  ggplot(aes(n, word, fill = sentiment)) +
  geom_col(show.legend = FALSE) +
  facet_wrap(~sentiment, scales = "free_y") +
  labs(x = "Contribution to sentiment",
       y = NULL)
```




```
custom_stop_words <- bind_rows(tibble(word = c("miss"),
                                      lexicon = c("custom")),
                               stop_words)

custom_stop_words
```

```
## # A tibble: 1,150 x 2
##    word        lexicon
##    <chr>       <chr>
##  1 miss        custom
##  2 a           SMART
##  3 a's         SMART
##  4 able        SMART
##  5 about       SMART
##  6 above       SMART
##  7 according   SMART
##  8 accordingly SMART
##  9 across      SMART
## 10 actually    SMART
## # ... with 1,140 more rows
```

```
#install.packages("wordcloud")
library(wordcloud)
```

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

```
tidy_books %>%
  anti_join(stop_words) %>%
  count(word) %>%
  with(wordcloud(word, n, max.words = 100))
```

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

```
## Warning in wordcloud(word, n, max.words = 100): elizabeth could not be fit on
## page. It will not be plotted.
```


```
library(reshape2)
```

```
##
## Attaching package: 'reshape2'
```

```
## The following object is masked from 'package:tidyr':
##
##     smiths
```

```
tidy_books %>%
  inner_join(get_sentiments("bing")) %>%
  count(word, sentiment, sort = TRUE) %>%
  acast(word ~ sentiment, value.var = "n", fill = 0) %>%
  comparison.cloud(colors = c("gray20", "gray80"),
                   max.words = 100)
```

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


```
p_and_p_sentences <- tibble(text = prideprejudice) %>%
  unnest_tokens(sentence, text, token = "sentences")

p_and_p_sentences$sentence[2]
```

```
## [1] "by jane austen"
```

```
austen_chapters <- austen_books() %>%
  group_by(book) %>%
  unnest_tokens(chapter, text, token = "regex",
                pattern = "Chapter|CHAPTER [\\dIVXLC]") %>%
  ungroup()

austen_chapters %>%
  group_by(book) %>%
  summarise(chapters = n())
```

```
## # A tibble: 6 x 2
##   book                chapters
##   <fct>                  <int>
## 1 Sense & Sensibility       51
## 2 Pride & Prejudice         62
```

```
## 3 Mansfield Park            49
## 4 Emma                      56
## 5 Northanger Abbey          32
## 6 Persuasion                25
```

```
bingnegative <- get_sentiments("bing") %>%
  filter(sentiment == "negative")

wordcounts <- tidy_books %>%
  group_by(book, chapter) %>%
  summarize(words = n())
```

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

```
tidy_books %>%
  semi_join(bingnegative) %>%
  group_by(book, chapter) %>%
  summarize(negativewords = n()) %>%
  left_join(wordcounts, by = c("book", "chapter")) %>%
  mutate(ratio = negativewords/words) %>%
  filter(chapter != 0) %>%
  slice_max(ratio, n = 1) %>%
  ungroup()
```

```
## Joining, by = "word"
## `summarise()` has grouped output by 'book'. You can override using the `.groups` argument.
```

```
## # A tibble: 6 x 5
##   book                chapter negativewords words  ratio
##   <fct>                 <int>         <int> <int>  <dbl>
## 1 Sense & Sensibility      43           161  3405 0.0473
## 2 Pride & Prejudice        34           111  2104 0.0528
## 3 Mansfield Park           46           173  3685 0.0469
## 4 Emma                     15           151  3340 0.0452
## 5 Northanger Abbey         21           149  2982 0.0500
## 6 Persuasion                4            62  1807 0.0343
```

From https://www.tidytextmining.com/sentiment.html: "These are the chapters with the most sad words in each book, normalized for number of words in the chapter. What is happening in these chapters? In Chapter 43 of Sense and Sensibility Marianne is seriously ill, near death, and in Chapter 34 of Pride and Prejudice Mr. Darcy proposes for the first time (so badly!). Chapter 46 of Mansfield Park is almost the end, when everyone learns of Henry's scandalous adultery, Chapter 15 of Emma is when horrifying Mr. Elton proposes, and in Chapter 21 of Northanger Abbey Catherine is deep in her Gothic faux fantasy of murder, etc. Chapter 4 of Persuasion is when the reader gets the full flashback of Anne refusing Captain Wentworth and how sad she was and what a terrible mistake she realized it to be."

Assignment

These bodies of text are from the Project Gutenberg. The following texts are downloaded and cited below.

The "physics" assignment line on line number 249 is from https://www.tidytextmining.com/tfidf.html, chapter 3 of the book.

Discourse on Floating Bodies by Galileo Galilei: https://www.gutenberg.org/ebooks/37729 Treatise on Light by Christiaan Huygens: http://www.gutenberg.org/ebooks/14725 Experiments with Alternate Currents of High Potential and High Frequency by Nikola Tesla: http://www.gutenberg.org/ebooks/13476 Relativity: The Special and General Theory by Albert Einstein: http://www.gutenberg.org/ebooks/30155

```
#install.packages("gutenbergr")
library(gutenbergr)
physics <- gutenberg_download(c(37729, 14725, 13476, 30155),
                              meta_fields = "author")
```

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

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

```
#install.packages("syuzhet")
library(syuzhet)
```

```
Tesla <- physics %>% filter(author == "Tesla, Nikola")
syuzhet_Tesla <- get_nrc_sentiment(toString(unlist(Tesla$text)))
(syuzhet_Tesla)
```

```
##   anger anticipation disgust fear joy sadness surprise trust negative positive
## 1    75          120      45  108 103      99       65   189      217      366
```

```
Galileo <-  physics %>% filter(author == "Galilei, Galileo")
Galileo_text <- unlist( Galileo$text)
Galileo_text <- toString(Galileo_text)
Galileo_scores <- get_nrc_sentiment(Galileo_text)

Galileo_scores
```

```
##   anger anticipation disgust fear joy sadness surprise trust negative positive
## 1    63           81      39   77  49      71       32   125      186      223
```