text mining

Christina

11/29/2021

Download and Tidy

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

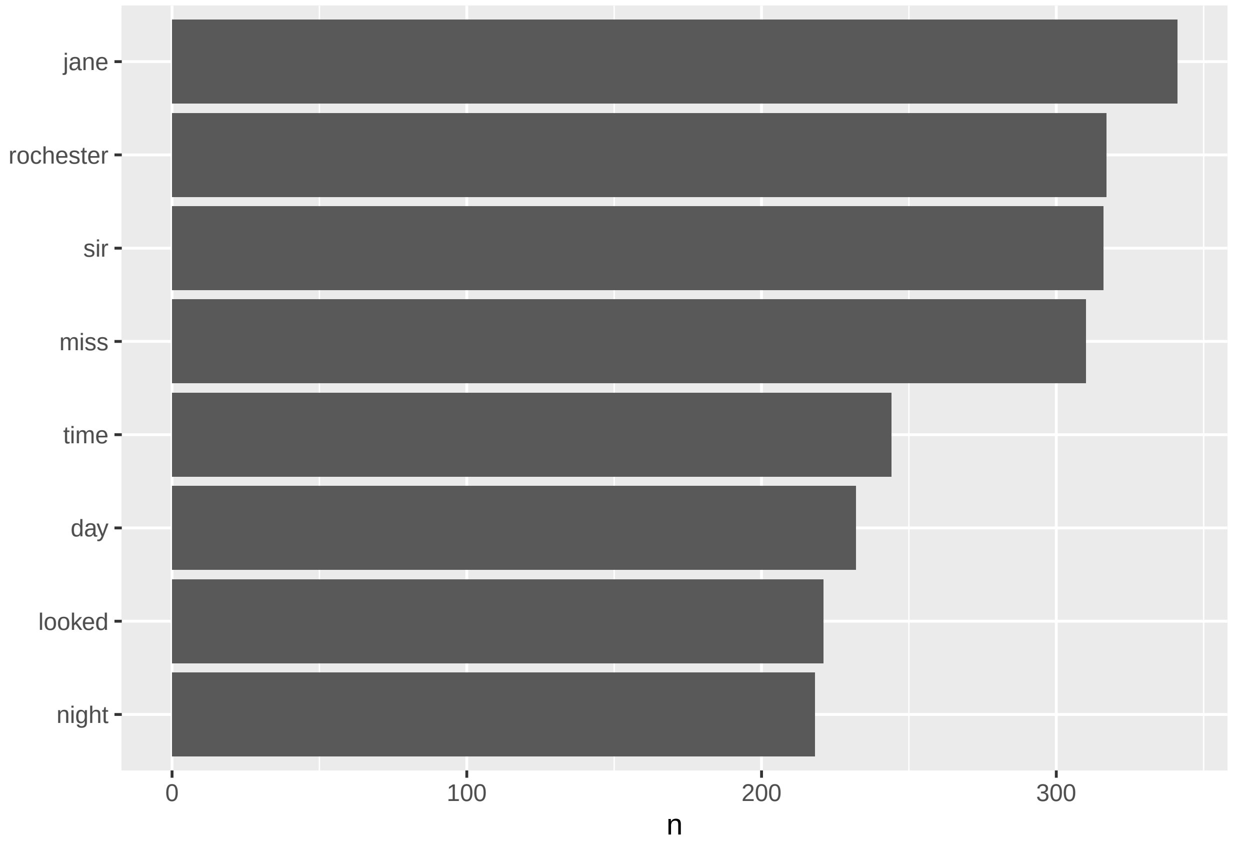
```
## Using mirror http://aleph.gutenberg.org
```

```
## # A tibble: 188,399 x 4
##    gutenberg_id linenumber chapter word
##           <int>      <int>   <int> <chr>
##  1         1260          1       0 jane
##  2         1260          1       0 eyre
##  3         1260          2       0 an
##  4         1260          2       0 autobiography
##  5         1260          4       0 by
##  6         1260          4       0 charlotte
##  7         1260          4       0 brontë
##  8         1260          6       0 _illustrated
##  9         1260          6       0 by
## 10         1260          6       0 f
## # ... with 188,389 more rows
```

```
## [1] 39
```

```
## # A tibble: 39 x 2
##    chapter     n
##      <int> <int>
##  1       0   959
##  2       1  1954
##  3       2  2771
##  4       3  3230
##  5       4  5866
##  6       5  5046
##  7       6  2923
##  8       7  3604
##  9       8  3015
## 10       9  3276
## # ... with 29 more rows
```

```
## Joining, by = "word"
## # A tibble: 12,309 x 2
##    word          n
##    <chr>     <int>
##  1 jane        341
##  2 rochester   317
##  3 sir         316
##  4 miss        310
##  5 time        244
##  6 day         232
##  7 looked      221
##  8 night       218
##  9 eyes        187
## 10 john        184
## # ... with 12,299 more rows
```



Sentiment analysis with tidy data

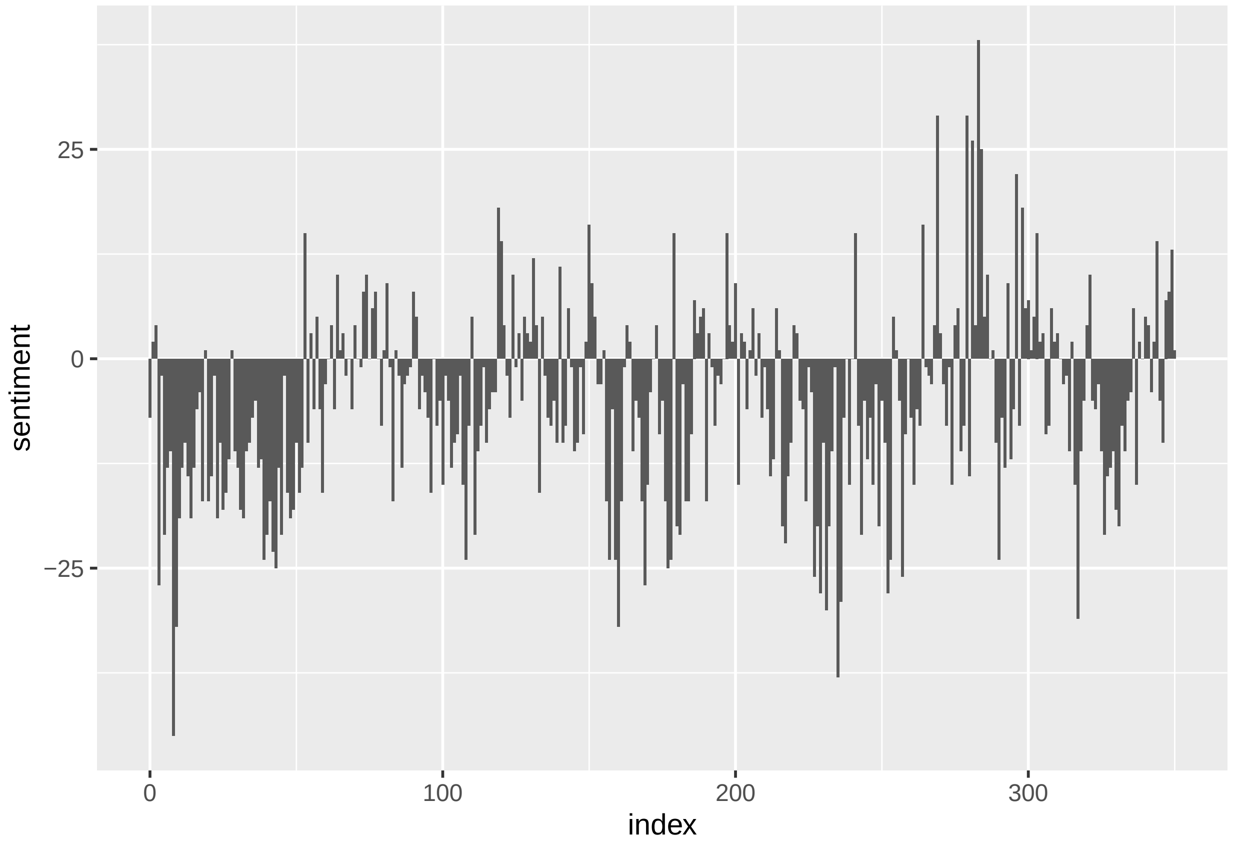
In the previous part, I explored what changed the original book into the tidy text format and showed how this format can be used to approach questions about word frequency. Next I would like to address the topic of sentiment analysis. When human readers approach a text, we use our understanding of the emotional intent of words to infer whether a section of text is positive or negative. So here I use the tools of text mining to appraoch the emotional content of text.

```
## Joining, by = "word"
## # A tibble: 404 x 2
##    word        n
##    <chr>   <int>
##  1 love      151
##  2 found     126
##  3 god        96
##  4 child      88
##  5 hope       75
##  6 feeling    67
##  7 happy      54
##  8 marry      49
##  9 smile      47
## 10 mother     45
## # ... with 394 more rows
```

I found that mostly positive, happy words about love, hope and happy here.

In addition, I want to examine how sentiment changes throughout each novel. First, I find a sentiment score for each word using the Bing lexicon and 'inner_join()'. Next I count up how many positive and negative words there are in defined sections of the book and define an "index" here to keep track of where we are in the narrative; this index (using integer division) counts up sections of 60 lines of text. In addition, I use 'pivot_wider()' so that we can have negative and positive sentiment in separate columns and finally calculate a net sentiment (positive - negative).

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

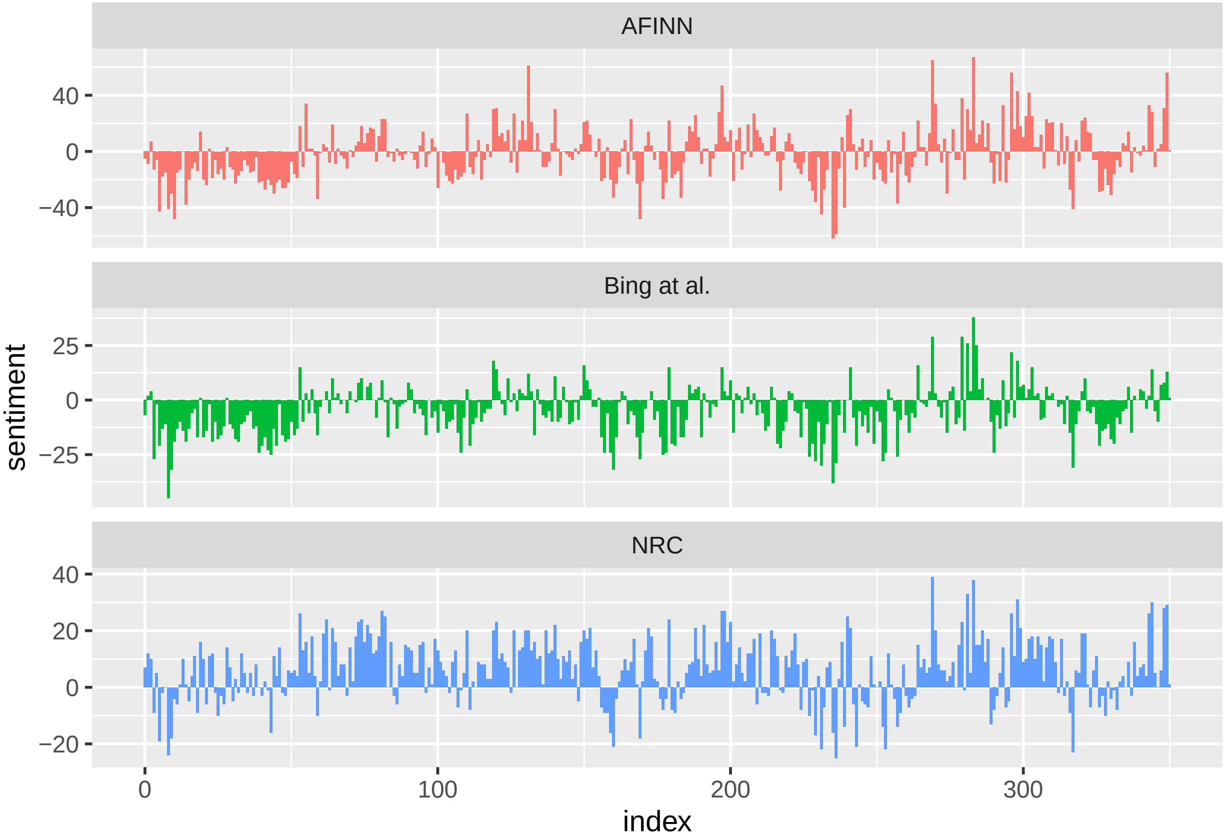


We can see how the plot of each novel changes toward more positive or negative sentiment over the trajectory of the story. From the plot above, I find that there are more negative sentiments during the early index and more positive sentiments during the late index. I think this fits the story. During Eyre's childhood, she was abuted by her aunt,Sarah Reed, she did not have a happy childhood in her early years. Therefore the sentiment is negative. But when she met Rochester as written in the late plots, she was married with him and had a sweet life. So there are more positive ones here.

Compare the three sentiment dictionaries

- AFINN: the AFINN lexicon measures sentiment with a numeric score between -5 and 5;
- Bing: the Bing lexicon categorizes words in a binary fashion into positive and negative categories;
- nrc: the nrc lexicon categorizes words in a binary fashion ("yes"/"no") into categories of positive, negative, anger, anticipation, disgust, fear, joy, sadness, surprise and trust.

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



The three different lexicons for calculating sentiment give results that are different in an absolute sense but have similar relative trajectories through the novel. We can see similar dips and peaks in sentiment at about the same places in the novel, but the absolute values are significantly different. The AFINN lexicon gives the largest absolute values, with high positive values. The Bing lexicon has lower absolute values. The NRC lexicon are shifted higher realtive to the other two, labeling the text more positively.

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

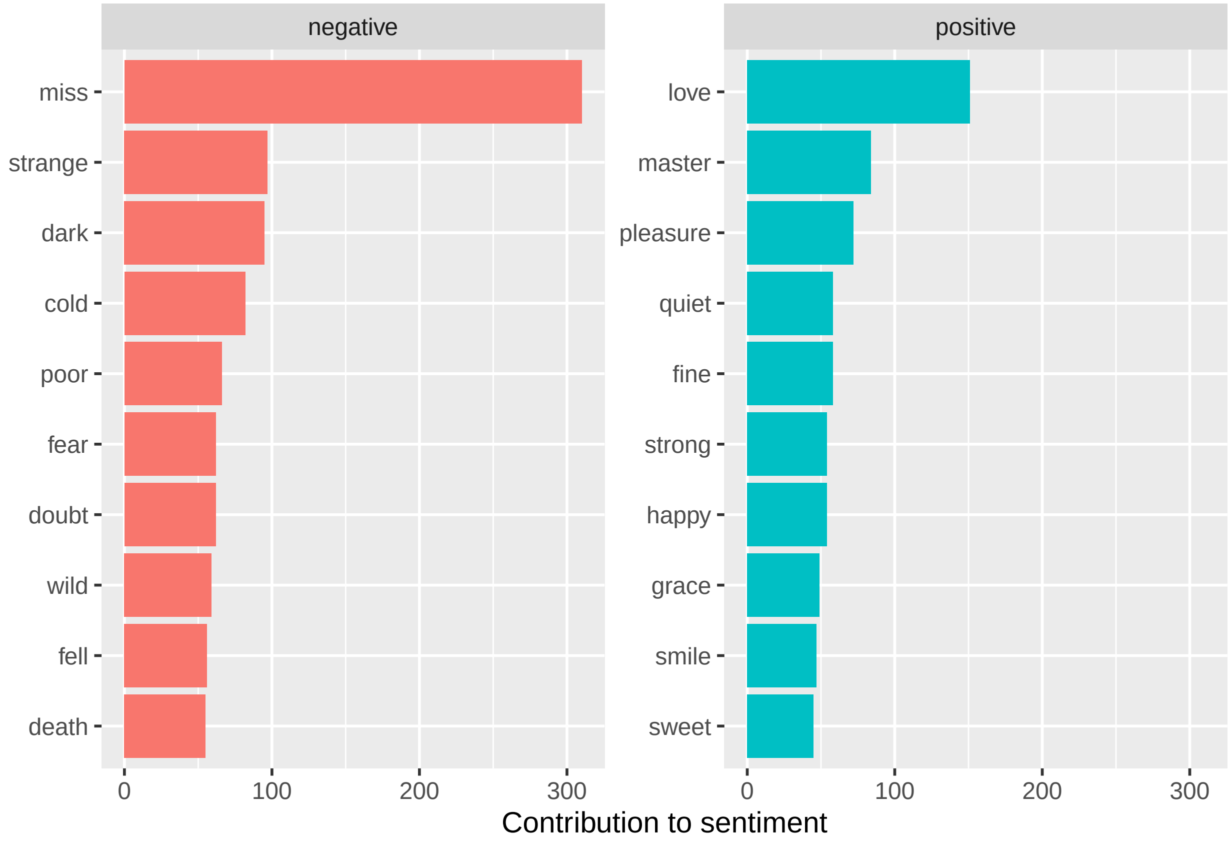
```
## # A tibble: 2 x 2
##   sentiment     n
##   <chr>     <int>
## 1 negative   4781
## 2 positive   2005
```

Both lexicons have more negative than positive words, but the ratio of negative to positive words is higher in the Bing lexicon than the NRC lexicon. This will contribute to the effect we see in the plot above, as will any systematic difference in word matches, e.g. if the negative words in the NRC lexicon do not match the words that Jane Austen uses very well. Whatever the source of these differences, we see similar relative trajectories across the narrative arc, with similar changes in slope, but marked differences in absolute sentiment from lexicon to lexicon. This is all important context to keep in mind when choosing a sentiment lexicon for analysis.

Most common positive and negative words – One advantage of having the data frame with both sentiment and word is that we can analyze word counts that contribute to each sentiment

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

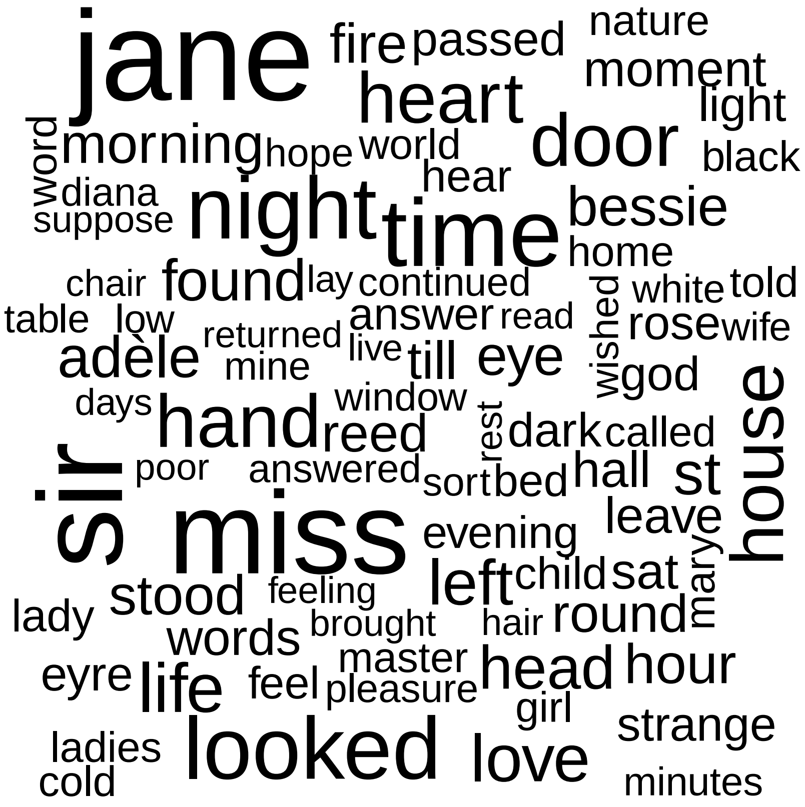
```
## # A tibble: 2,326 x 3
##    word     sentiment     n
##    <chr>    <chr>     <int>
##  1 miss     negative    310
##  2 love     positive    151
##  3 strange  negative     97
##  4 dark     negative     95
##  5 master   positive     84
##  6 cold     negative     82
##  7 pleasure positive     72
##  8 poor     negative     66
##  9 doubt    negative     62
## 10 fear     negative     62
## # ... with 2,316 more rows
```



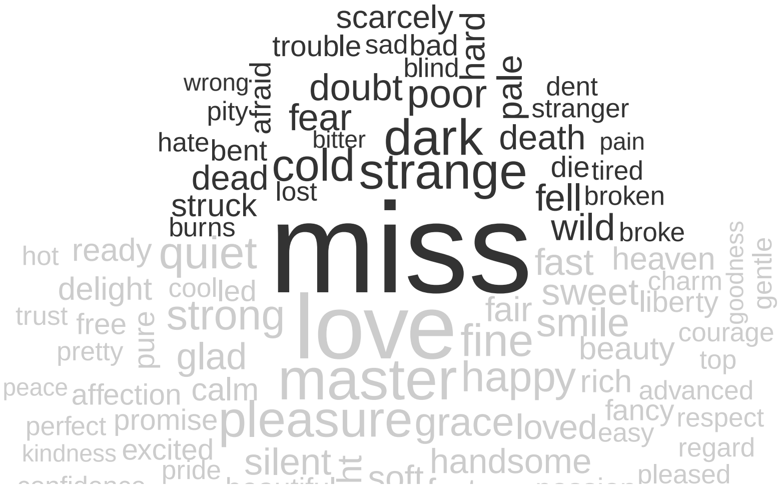
The plot above shows words that contribute to positive and negative sentiment in Eyre.

Wordclouds

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

## Warning in strwidth(words[i], cex = size[i], ...): conversion failure on 'don't'
## in 'mbcsToSbcs': dot substituted for <e2>

## Warning in strwidth(words[i], cex = size[i], ...): conversion failure on 'don't'
## in 'mbcsToSbcs': dot substituted for <80>

## Warning in strwidth(words[i], cex = size[i], ...): conversion failure on 'don't'
## in 'mbcsToSbcs': dot substituted for <99>

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

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



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



The size of a word's text is in proportion to its frequency within its sentiment. We can use this visualization to see the most important positive and negative words. Here the most important positive word is 'positive' and the most important negative word is 'miss'.

Task three

Truenumbers (TN) is a system for building data resources that are hosted on a server and accessed through clients that interact with with server through the Truenumbers API. The system uses natural language descriptions of data items and includes a tagging function for augmenting data items, defining subsets, and tracking process metadata. Following I would like to use TN to do analysis on Eyre book.

In this part, I used the tnum package to explore and tag text from the book Jane Eyre. I explored the frequency of words and characters throughout each section of the book, and created visualizations to show these frequencies. The process is described in more detail below. ## Download packages and the book

Assign the return from 'tnum.query' function to a variable so that we can examine the list items in the environment. And then convert the TN list to a data frame (use the 'tnum.objectstoDF()' function) so that each TN is a row in the data frame.

```
## Returned 1 thru 500 of 1637 results
```

```
## Returned 1 thru 10 of 1637 results
```

```
## Returned 1 thru 6 of 6 results
##                                               subject   property string.value
## 1 jane_eyre/section:0001/paragraph:0005/sentence:0001 count:word         <NA>
## 2 jane_eyre/section:0001/paragraph:0005/sentence:0002 count:word         <NA>
## 3 jane_eyre/section:0001/paragraph:0005/sentence:0003 count:word         <NA>
##   numeric.value error unit tags       date                                 guid
## 1             6    NA   NA      2021-12-03 87a187ac-d130-4bb1-9799-d8107c27b51f
## 2            10    NA   NA      2021-12-03 9dcffed8-d075-40d6-bf79-8f5a9ef76fb7
## 3            27    NA   NA      2021-12-03 ea301cab-7e8c-474b-8052-f3e5f6f700b3
```

```
## Returned 1 thru 2 of 2 results
```

```
##                                               subject property string.value
## 1 jane_eyre/section:0001/paragraph:0005/sentence:0002     text  "TO W.. M."
##   numeric.value error unit tags       date                                 guid
## 1            NA    NA   NA      2021-12-03 0c259a19-3ca8-4a38-9a1a-14e31322136c
```

```
## Returned 1 thru 6 of 6 results
```

Use tnum in text analysis

In this part, I want to use TNs for text analysis. As I did before, loading the libraries I need, authorizing the server if needed and setting the number space to “test2”.

```
## Returned 1 thru 1637 of 1637 results
```

```
## Returned 4 thru 21 of 1637 results
```

```
## Returned 1 thru 24 of 24 results
## Returned 1 thru 24 of 24 results
```

```
## Joining, by = "subject"
```

```
##
## Attaching package: 'magrittr'
```

```
## The following object is masked from 'package:purrr':
##
##     set_names
```

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

```
## Returned 1 thru 10 of 74 results
```

```
## Returned 1 thru 3 of 4887 results
```

```
## Returned 1 thru 1 of 1 results
## Returned 1 thru 1 of 1 results
## Returned 1 thru 3 of 3 results
```

```
## [1] "\""Jane, you are under a mistake: what is the matter with you?\""
## [2] "\"Why do you tremble so violently?\""
## [3] "\"Why do you tremble so violently?? Would you like to drink some water?\""
```

```
## Returned 1 thru 9 of 9 results
```

```
## [1] NA
## [2] NA
## [3] NA
## [4] NA
## [5] NA
## [6] NA
## [7] "\""Jane, you are under a mistake: what is the matter with you?\""
## [8] "\"Why do you tremble so violently?\""
## [9] "\"Why do you tremble so violently?? Would you like to drink some water?\""
```

```
## [1] 9
```

```
## Returned 1 thru 9 of 9 results
```

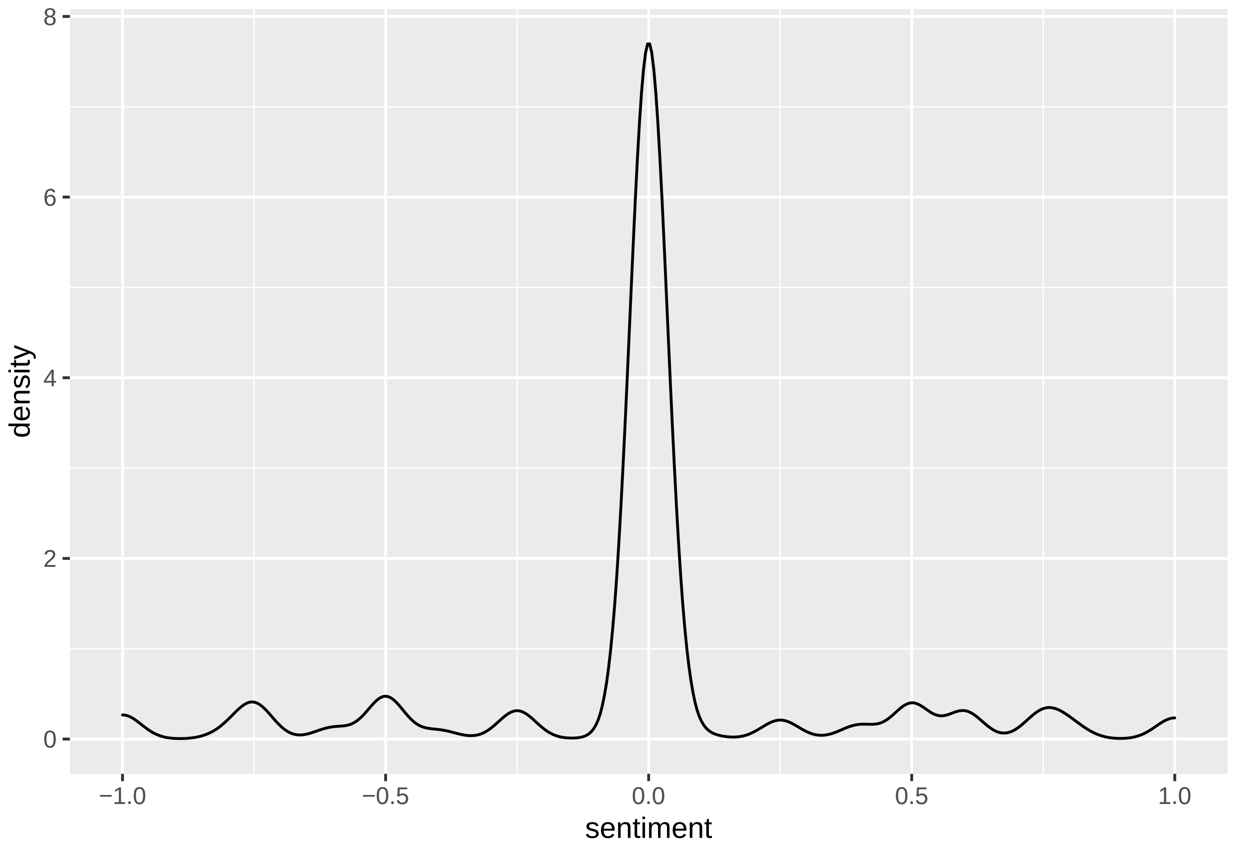
Sentimentr

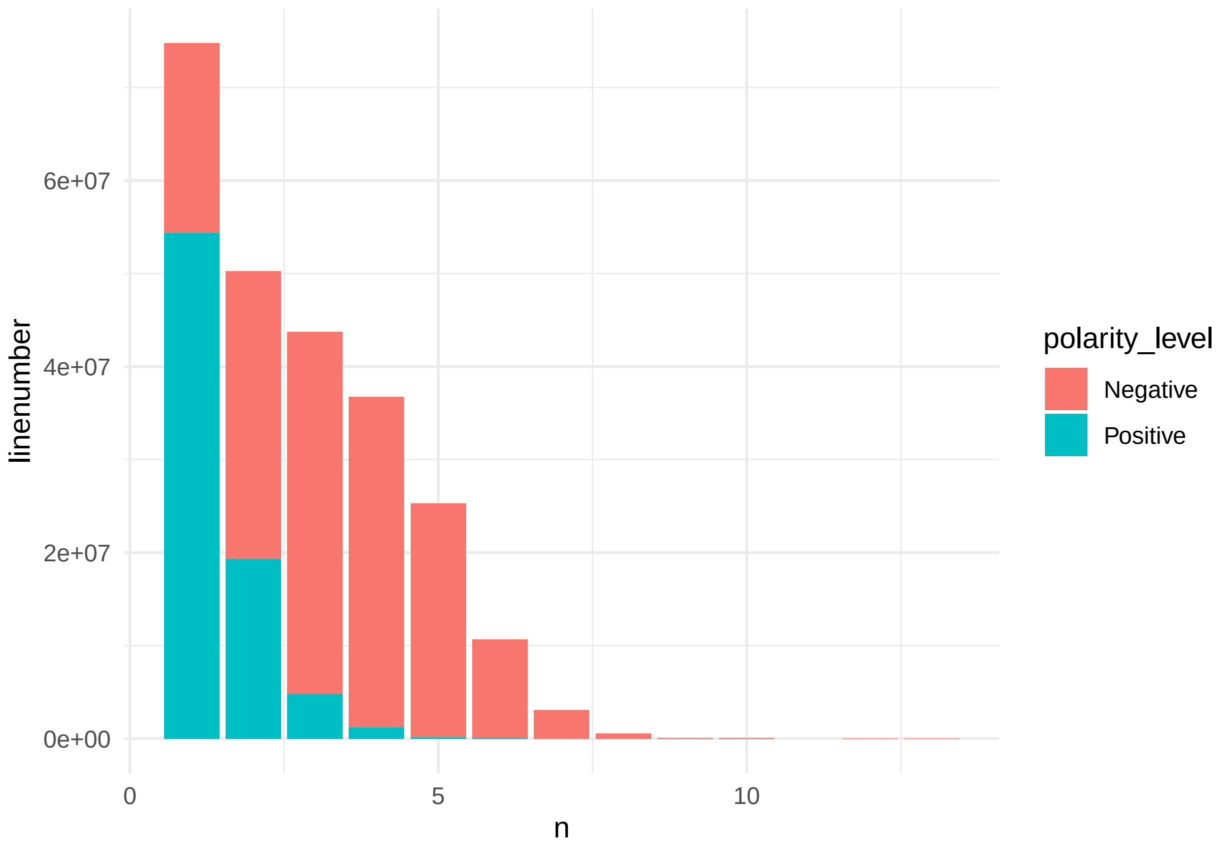
```
##    element_id sentence_id word_count sentiment
## 1:          1           1          2         0
## 2:          1           2          2         0
## 3:          1           3          1         0
## 4:          1           4          2         0
## 5:          1           5          1         0
## 6:          1           6          2         0
```

```
##    element_id word_count sd ave_sentiment
## 1:          1         10  0             0
```

```
## Aggregate function missing, defaulting to 'length'
```

```
##     element_id sentence_id word_count         emotion_type emotion_count
##  1:          1           1          2                anger             0
##  2:          1           1          2        anger_negated             0
##  3:          1           1          2         anticipation             0
##  4:          1           1          2 anticipation_negated             0
##  5:          1           1          2              disgust             0
##  6:          1           1          2      disgust_negated             0
##  7:          1           1          2                 fear             0
##  8:          1           1          2         fear_negated             0
##  9:          1           1          2                  joy             0
## 10:          1           1          2          joy_negated             0
## 11:          1           1          2              sadness             0
## 12:          1           1          2      sadness_negated             0
## 13:          1           1          2             surprise             0
## 14:          1           1          2     surprise_negated             0
## 15:          1           1          2                trust             0
## 16:          1           1          2        trust_negated             0
## 17:          1           2          2                anger             0
## 18:          1           2          2        anger_negated             0
## 19:          1           2          2         anticipation             0
## 20:          1           2          2 anticipation_negated             0
## 21:          1           2          2              disgust             0
## 22:          1           2          2      disgust_negated             0
## 23:          1           2          2                 fear             0
## 24:          1           2          2         fear_negated             0
## 25:          1           2          2                  joy             0
## 26:          1           2          2          joy_negated             0
## 27:          1           2          2              sadness             0
## 28:          1           2          2      sadness_negated             0
## 29:          1           2          2             surprise             0
## 30:          1           2          2     surprise_negated             0
## 31:          1           2          2                trust             0
## 32:          1           2          2        trust_negated             0
## 33:          1           3          1                anger             0
## 34:          1           3          1        anger_negated             0
## 35:          1           3          1         anticipation             0
## 36:          1           3          1 anticipation_negated             0
## 37:          1           3          1              disgust             0
## 38:          1           3          1      disgust_negated             0
## 39:          1           3          1                 fear             0
## 40:          1           3          1         fear_negated             0
## 41:          1           3          1                  joy             0
## 42:          1           3          1          joy_negated             0
## 43:          1           3          1              sadness             0
## 44:          1           3          1      sadness_negated             0
## 45:          1           3          1             surprise             0
## 46:          1           3          1     surprise_negated             0
## 47:          1           3          1                trust             0
## 48:          1           3          1        trust_negated             0
## 49:          1           4          2                anger             0
## 50:          1           4          2        anger_negated             0
## 51:          1           4          2         anticipation             0
## 52:          1           4          2 anticipation_negated             0
## 53:          1           4          2              disgust             0
## 54:          1           4          2      disgust_negated             0
## 55:          1           4          2                 fear             0
## 56:          1           4          2         fear_negated             0
## 57:          1           4          2                  joy             0
## 58:          1           4          2          joy_negated             0
## 59:          1           4          2              sadness             0
## 60:          1           4          2      sadness_negated             0
## 61:          1           4          2             surprise             0
## 62:          1           4          2     surprise_negated             0
## 63:          1           4          2                trust             0
## 64:          1           4          2        trust_negated             0
## 65:          1           5          1                anger             0
## 66:          1           5          1        anger_negated             0
## 67:          1           5          1         anticipation             0
## 68:          1           5          1 anticipation_negated             0
## 69:          1           5          1              disgust             0
## 70:          1           5          1      disgust_negated             0
## 71:          1           5          1                 fear             0
## 72:          1           5          1         fear_negated             0
## 73:          1           5          1                  joy             0
## 74:          1           5          1          joy_negated             0
## 75:          1           5          1              sadness             0
## 76:          1           5          1      sadness_negated             0
## 77:          1           5          1             surprise             0
## 78:          1           5          1     surprise_negated             0
## 79:          1           5          1                trust             0
## 80:          1           5          1        trust_negated             0
## 81:          1           6          2                anger             0
## 82:          1           6          2        anger_negated             0
## 83:          1           6          2         anticipation             0
## 84:          1           6          2 anticipation_negated             0
## 85:          1           6          2              disgust             0
## 86:          1           6          2      disgust_negated             0
## 87:          1           6          2                 fear             0
## 88:          1           6          2         fear_negated             0
## 89:          1           6          2                  joy             0
## 90:          1           6          2          joy_negated             0
## 91:          1           6          2              sadness             0
## 92:          1           6          2      sadness_negated             0
## 93:          1           6          2             surprise             0
## 94:          1           6          2     surprise_negated             0
## 95:          1           6          2                trust             0
## 96:          1           6          2        trust_negated             0
##     element_id sentence_id word_count         emotion_type emotion_count
##     emotion
##  1:       0
##  2:       0
##  3:       0
##  4:       0
##  5:       0
##  6:       0
##  7:       0
##  8:       0
##  9:       0
## 10:       0
## 11:       0
## 12:       0
## 13:       0
## 14:       0
## 15:       0
## 16:       0
## 17:       0
## 18:       0
## 19:       0
## 20:       0
## 21:       0
## 22:       0
## 23:       0
## 24:         0
## 25:         0
## 26:         0
## 27:         0
## 28:         0
## 29:         0
## 30:         0
## 31:         0
## 32:         0
## 33:         0
## 34:         0
## 35:         0
## 36:         0
## 37:         0
## 38:         0
## 39:         0
## 40:         0
## 41:         0
## 42:         0
## 43:         0
## 44:         0
## 45:         0
## 46:         0
## 47:         0
## 48:         0
## 49:         0
## 50:         0
## 51:         0
## 52:         0
## 53:         0
## 54:         0
## 55:         0
## 56:         0
## 57:         0
## 58:         0
## 59:         0
## 60:         0
## 61:         0
## 62:         0
## 63:         0
## 64:         0
## 65:         0
## 66:         0
## 67:         0
## 68:         0
## 69:         0
## 70:         0
## 71:         0
## 72:         0
## 73:         0
## 74:         0
## 75:         0
## 76:         0
## 77:         0
## 78:       0
## 79:       0
## 80:       0
## 81:       0
## 82:       0
## 83:       0
## 84:       0
## 85:       0
## 86:       0
## 87:       0
## 88:       0
## 89:       0
## 90:       0
## 91:       0
## 92:       0
## 93:       0
## 94:       0
## 95:       0
## 96:       0
##     emotion
```





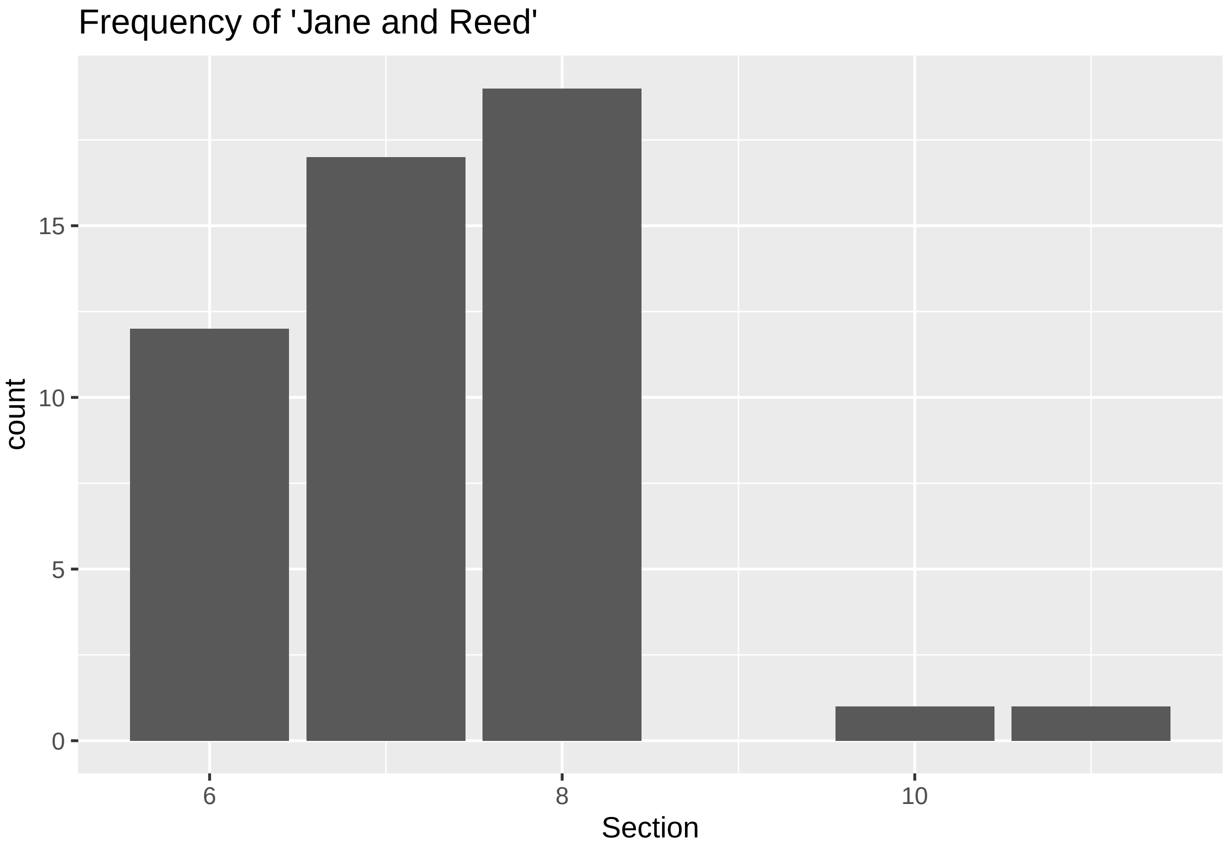
From the density plot, we can see most of the sentiments are zero. For the plot with 'polarity_level', each linenumber has different number of positive and negative sentiments.

Extra attempts: tag

From the content, I would like to analysis the relationship between Jane and her uncle, Reed. Firstly I create a tag to 'Jane|Reed' in order to find where they comes out together in this book.

```
## Returned 1 thru 50 of 141 results
```

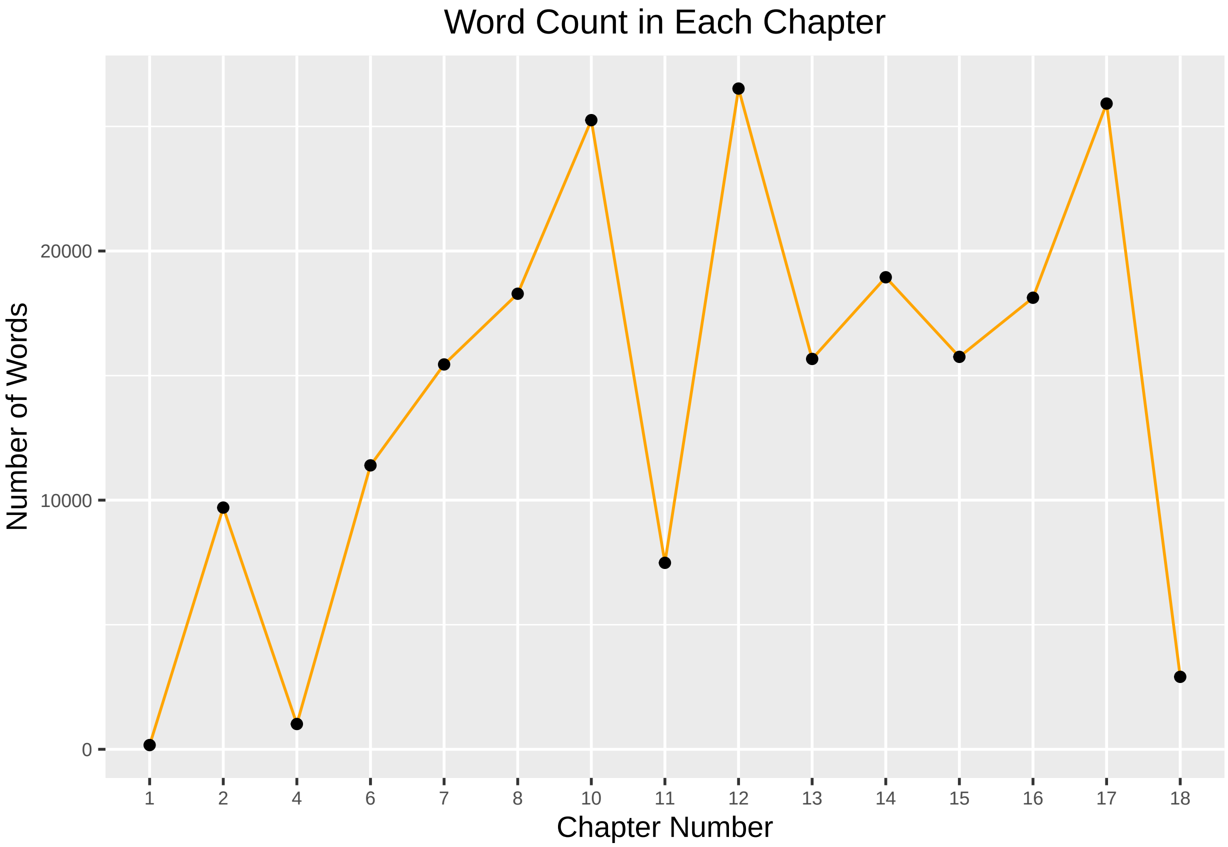
```
## list(modifiedCount = 141, tagged = 141, removed = 0)
```

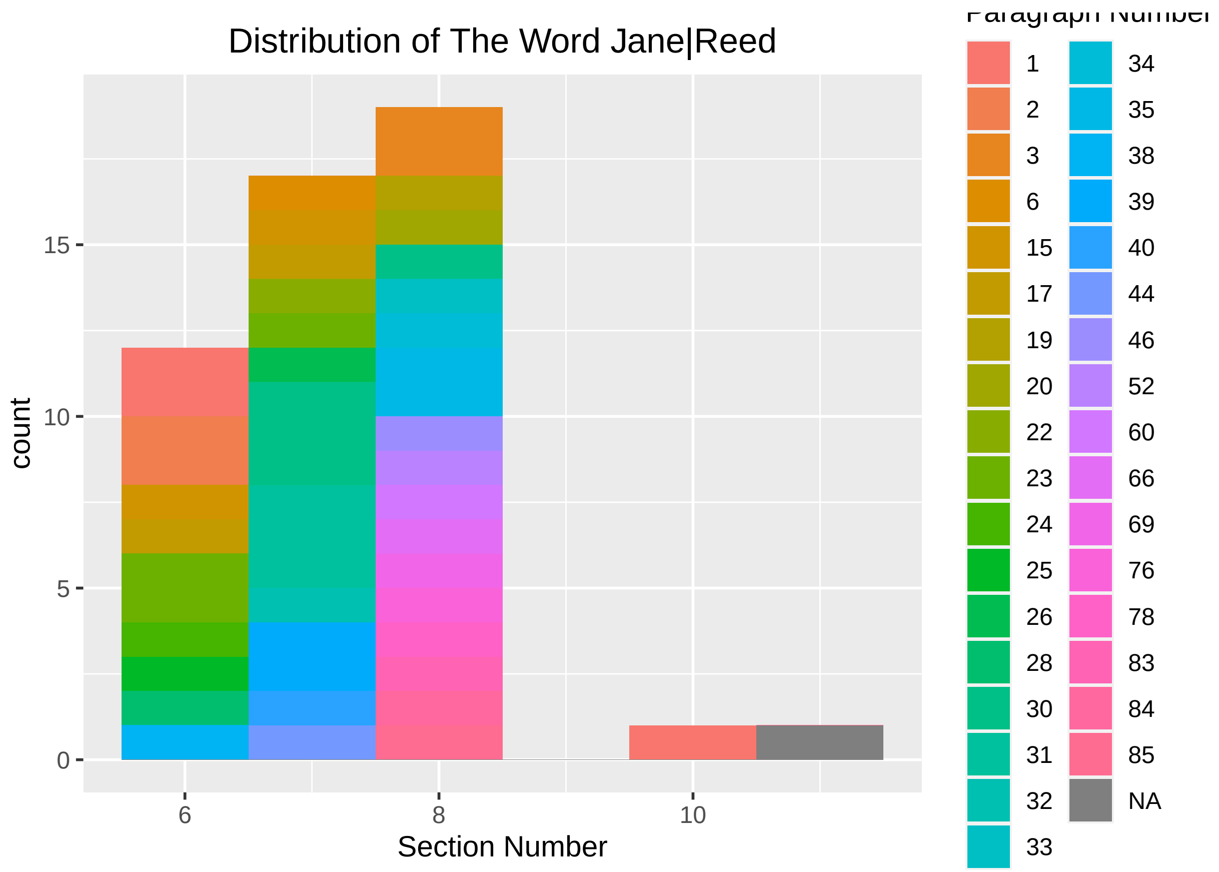


From the plot above, it shows that ‘Jane and Reed’ comes out together in several sections. The most important part is in the earlier part of the book, which makes sense. This fits the story that the book tells – her uncle, Reed, raises Jane when she was a child.

Show virsually

```
## Returned 1 thru 1500 of 1613 results
```


Word Count in Each Chapter
Number of Words
20000
10000
0
1
2
4
6
7
8
10
11
12
13
14
15
16
17
18
Chapter Number



```
## PhantomJS not found. You can install it with webshot::install_phantomjs(). If it is installed, please
```

This part I would like to show the relationship between Jane and Reed, but I cannot use the function such as ‘tnum.plotGraph()’ in the file you showed us in class. There is an error saying these functions do not exist. What I can do is drawing a plot showing the distribution of the Word ‘Jane|Reed’ occurances. I will work on the part that does not go well in the future.