Introduction to Big Data and Analytics

Group 4 - Project 3

Introduction

In this project we have to perform text analysis on a book - APrincessOfMars.txt which has different chapters in it. We will do our analysis specifically on first 11 chapters of the book. To begin with we will set the working directory for our project which has the R file and the txt file -

```
>
> # set the directory
> setwd("C:/Masters/gwu/big data/data")
> getwd()
[1] "C:/Masters/gwu/big data/data"
>
```

We install the required package -

```
> install.packages("tm")
Installing package into 'C:/Users/prani/AppData/Local/R/win-library/4.2'
(as 'lib' is unspecified)
trying URL 'https://cran.rstudio.com/bin/windows/contrib/4.2/tm_0.7-11.zip'
Content type 'application/zip' length 989664 bytes (966 KB)
downloaded 966 KB
```

```
> (library(tm))
Loading required package: NLP
[1] "tm"        "NLP"       "stats"     "graphics"  "grDevices" "utils"     "datasets"  "methods"   "base"
Warning message:
package 'tm' was built under R version 4.2.3
>
```

The Text File

This particular text file includes 11 chapters, to extract these chapters we are using the **which** function. This code finds the indices of particular texts ("CHAPTER I", "CHAPTER II", etc.) in the character vector my_text.

The indices produced are saved in variables index_ch1, index_ch2, and so forth till index_ch12.

```
> my_text <- readLines("APrincessOfMars.txt")
> # logic to extract chapters 1 - 11
> index_ch1 <- which(my_text == "CHAPTER I", arr.ind = TRUE)
> index_ch2 <- which(my_text == "CHAPTER II", arr.ind = TRUE)
> index_ch3 <- which(my_text == "CHAPTER III", arr.ind = TRUE)
> index_ch4 <- which(my_text == "CHAPTER IV", arr.ind = TRUE)
> index_ch5 <- which(my_text == "CHAPTER V", arr.ind = TRUE)
> index_ch6 <- which(my_text == "CHAPTER VI", arr.ind = TRUE)
> index_ch7 <- which(my_text == "CHAPTER VII", arr.ind = TRUE)
> index_ch8 <- which(my_text == "CHAPTER VIII", arr.ind = TRUE)
> index_ch9 <- which(my_text == "CHAPTER IX", arr.ind = TRUE)
> index_ch10 <- which(my_text == "CHAPTER X", arr.ind = TRUE)
> index_ch11 <- which(my_text == "CHAPTER XI", arr.ind = TRUE)
> index_ch12 <- which(my_text == "CHAPTER XII", arr.ind = TRUE)
```

This code below extracts the text for each chapter from the original text vector my_text using the indices of the chapter headings (discovered in the previous code block).
Each line of code generates a new variable (book_ch1, book_ch2, etc.) containing the text for each chapter, as determined by the beginning and ending points of the respective chapter headings.
For example, book_ch1 contains the text between the first appearance of "CHAPTER I" and the second appearance of "CHAPTER II", book_ch2 contains the material between the second appearance of "CHAPTER II" and the third appearance of "CHAPTER III", and so on.

```
> book_ch1 <- my_text[(index_ch1+1):(index_ch2-1)]
> book_ch2 <- my_text[(index_ch2+1):(index_ch3-1)]
> book_ch3 <- my_text[(index_ch3+1):(index_ch4-1)]
> book_ch4 <- my_text[(index_ch4+1):(index_ch5-1)]
> book_ch5 <- my_text[(index_ch5+1):(index_ch6-1)]
> book_ch6 <- my_text[(index_ch6+1):(index_ch7-1)]
> book_ch7 <- my_text[(index_ch7+1):(index_ch8-1)]
> book_ch8 <- my_text[(index_ch8+1):(index_ch9-1)]
> book_ch9 <- my_text[(index_ch9+1):(index_ch10-1)]
> book_ch10 <- my_text[(index_ch10+1):(index_ch11-1)]
> book_ch11 <- my_text[(index_ch11+1):(index_ch12-1)]
```

Then we created a new directory called "chapters" using the dir.create() function. It then uses the write.table() function to write the text for each chapter (as stored in the variables book_ch1 through book_ch11) to separate text files. Each text file is saved in the "chapters" directory with the chapter number as the filename.

The write.table() sep, row.names, col.names, and quote arguments are used to indicate the separator between columns (tab), whether to include row and column names (FALSE), and whether to enclose fields in quotes (FALSE). This code block creates distinct text files in a new directory called "chapters" for each chapter of the book:

```
> dir.create("chapters")
> write.table(book_ch1, file = "chapters/book_ch1.txt", sep = "\t", row.names = FALSE, col.names = FALSE, quote = FALSE)
> write.table(book_ch2, file = "chapters/book_ch2.txt", sep = "\t", row.names = FALSE, col.names = FALSE, quote = FALSE)
> write.table(book_ch3, file = "chapters/book_ch3.txt", sep = "\t", row.names = FALSE, col.names = FALSE, quote = FALSE)
> write.table(book_ch4, file = "chapters/book_ch4.txt", sep = "\t", row.names = FALSE, col.names = FALSE, quote = FALSE)
> write.table(book_ch5, file = "chapters/book_ch5.txt", sep = "\t", row.names = FALSE, col.names = FALSE, quote = FALSE)
> write.table(book_ch6, file = "chapters/book_ch6.txt", sep = "\t", row.names = FALSE, col.names = FALSE, quote = FALSE)
> write.table(book_ch7, file = "chapters/book_ch7.txt", sep = "\t", row.names = FALSE, col.names = FALSE, quote = FALSE)
> write.table(book_ch8, file = "chapters/book_ch8.txt", sep = "\t", row.names = FALSE, col.names = FALSE, quote = FALSE)
> write.table(book_ch9, file = "chapters/book_ch9.txt", sep = "\t", row.names = FALSE, col.names = FALSE, quote = FALSE)
> write.table(book_ch10, file = "chapters/book_ch10.txt", sep = "\t", row.names = FALSE, col.names = FALSE, quote = FALSE)
> write.table(book_ch11, file = "chapters/book_ch11.txt", sep = "\t", row.names = FALSE, col.names = FALSE, quote = FALSE)
```

VCorpus Object

The str() function displays the book_corpus object's structure. Running str(book_corpus) will provide basic information about the corpus, such as its class, number of documents, and source.

The output indicates that the VCorpus object includes 11 documents, each of which is represented as a list with two elements: "content" and "meta". The "content" element holds the document's text, and the "meta" element contains document metadata such as the author, date, description, heading, ID, language, and origin. All of the documents currently have no metadata. Each document belongs to the "PlainTextDocument" and "TextDocument" classes.

```
> book_corpus <- VCorpus(DirSource("chapters", mode = "text"))
> str(book_corpus)
Classes 'VCorpus', 'Corpus'  hidden list of 3
 $ content:List of 11
  ..$ :List of 2
  .. ..$ content: chr [1:267] "" "ON THE ARIZONA HILLS" "" "" ...
  .. ..$ meta   :List of 7
  .. .. ..$ author       : chr(0)
  .. .. ..$ datetimestamp: POSIXlt[1:1], format: "2023-05-06 18:40:27"
  .. .. ..$ description  : chr(0)
  .. .. ..$ heading      : chr(0)
  .. .. ..$ id           : chr "book_ch1.txt"
  .. .. ..$ language     : chr "en"
  .. .. ..$ origin       : chr(0)
  .. .. ..- attr(*, "class")= chr "TextDocumentMeta"
  .. ..- attr(*, "class")= chr [1:2] "PlainTextDocument" "TextDocument"
  ..$ :List of 2
  .. ..$ content: chr [1:358] "" "CHAMPION AND CHIEF" "" "" ...
  .. ..$ meta   :List of 7
  .. .. ..$ author       : chr(0)
  .. .. ..$ datetimestamp: POSIXlt[1:1], format: "2023-05-06 18:40:27"
  .. .. ..$ description  : chr(0)
```

```
> inspect(book_corpus)
<<VCorpus>>
Metadata:  corpus specific
Content:  documents: 11

[[1]]
<<PlainTextDocument>>
Metadata:  7
Content:  chars: 13962

[[2]]
<<PlainTextDocument>>
Metadata:  7
Content:  chars: 19113

[[3]]
<<PlainTextDocument>>
Metadata:  7
Content:  chars: 13151

[[4]]
<<PlainTextDocument>>
Metadata:  7
Content:  chars: 9000

[[5]]
<<PlainTextDocument>>
Metadata:  7
Content:  chars: 14190
```

10 Longest Words and Sentences

tidy(book_corpus): This line uses the tidy() function from the tidytext package to convert the book_corpus object (which we believe includes a corpus of texts) into a tidy format. This function divides the corpus into individual words, with each word associated with the document id from which it came.

unnest_tokens(text, word): This line employs the tidytext package's unnest_tokens() function to "unnest" the words column formed by tidy(), dividing each document into individual words. The first argument, word, provides the name of the new column in which the words will be stored. The second input, text, indicates the name of the existing column containing the unnested text.

select(id, word): This line employs the dplyr package's select() method to choose only the id and word columns from the generated data frame. This

removes any additional columns that may have been created by the preceding stages.
mutate(word_length = nchar(word)): This line calls the dplyr package's mutate() method to create a new column named word_length, which holds the number of characters in each word. To get the length of each word, use the nchar() function.
arrange(desc(word_length)): This line uses the dplyr package's arrange() function to sort the resulting data frame in descending order of word_length, with the longest words at the top. To sort in descending order, use the desc() function.

Packages for finding longest words and sentences -

```
56
57 install.packages("dplyr")
58 install.packages("tidytext")
59 library(dplyr)
60 library(tidytext)
61
```

Longest words

```
> # 10 longest words
> book_words <- tidy(book_corpus) %>%
+   unnest_tokens(word, text) %>%
+   select(id, word) %>%
+   mutate(word_length = nchar(word)) %>%
+   arrange(desc(word_length))
Warning message:
Outer names are only allowed for unnamed scalar atomic inputs
```

Using the filter() function and the data frame's id column, this code below filter the book_words data frame by each chapter of the book. Each line filters the data frame for a single chapter by matching the id column with the file name for that chapter (book_ch1.txt, book_ch2.txt, and so on). This displays the top ten longest words in each chapter of the book.

```
> book_words %>% filter(id=="book_ch1.txt")
# A tibble: 2,616 × 3
   id            word           word_length
   <chr>         <chr>                <int>
 1 book_ch1.txt subconsciously          14
 2 book_ch1.txt characteristic          14
 3 book_ch1.txt understanding           13
 4 book_ch1.txt comparatively           13
 5 book_ch1.txt sensitiveness           13
 6 book_ch1.txt consternation           13
 7 book_ch1.txt perpendicular           13
 8 book_ch1.txt resuscitation           13
 9 book_ch1.txt comparatively           13
10 book_ch1.txt resurrection            12
# ℹ 2,606 more rows
# ℹ Use `print(n = ...)` to see more rows
> book_words %>% filter(id=="book_ch2.txt")
# A tibble: 1,683 × 3
   id            word           word_length
   <chr>         <chr>                <int>
 1 book_ch2.txt contemplation           13
 2 book_ch2.txt metamorphosis           13
 3 book_ch2.txt particularly            12
 4 book_ch2.txt predicaments            12
 5 book_ch2.txt interruption            12
 6 book_ch2.txt overstrained            12
 7 book_ch2.txt bewilderment            12
 8 book_ch2.txt surroundings            12
 9 book_ch2.txt unfathomable            12
10 book_ch2.txt crystallized            12
# ℹ 1,673 more rows
# ℹ Use `print(n = ...)` to see more rows
```

```
> book_words %>% filter(id=="book_ch3.txt")
# A tibble: 2,609 × 3
   id            word           word_length
   <chr>         <chr>                <int>
 1 book_ch3.txt characteristics         15
 2 book_ch3.txt irregularities          14
 3 book_ch3.txt characteristic          14
 4 book_ch3.txt consciousness           13
 5 book_ch3.txt independently           13
 6 book_ch3.txt accouterments           13
 7 book_ch3.txt noiselessness           13
 8 book_ch3.txt gesticulating           13
 9 book_ch3.txt comparatively           13
10 book_ch3.txt interminable            12
# ℹ 2,599 more rows
# ℹ Use `print(n = ...)` to see more rows
> book_words %>% filter(id=="book_ch4.txt")
# A tibble: 2,112 × 3
   id            word           word_length
   <chr>         <chr>                <int>
 1 book_ch4.txt circumstances           13
 2 book_ch4.txt consideration           13
 3 book_ch4.txt manifestation           13
 4 book_ch4.txt consideration           13
 5 book_ch4.txt therapeutics            12
 6 book_ch4.txt scintillated            12
 7 book_ch4.txt introduction            12
 8 book_ch4.txt instructions            12
 9 book_ch4.txt observation             11
10 book_ch4.txt immediately             11
# ℹ 2,102 more rows
# ℹ Use `print(n = ...)` to see more rows
```

```
> book_words %>% filter(id=="book_ch10.txt")
# A tibble: 3,522 × 3
   id            word           word_length
   <chr>         <chr>                <int>
 1 book_ch10.txt responsibilities       16
 2 book_ch10.txt characteristics        15
 3 book_ch10.txt companionship          13
 4 book_ch10.txt manifestation          13
 5 book_ch10.txt ludicrousness          13
 6 book_ch10.txt precipitately          13
 7 book_ch10.txt authoritative          13
 8 book_ch10.txt understanding          13
 9 book_ch10.txt theoretically          13
10 book_ch10.txt companionship          13
# ℹ 3,512 more rows
# ℹ Use `print(n = ...)` to see more rows
> book_words %>% filter(id=="book_ch11.txt")
# A tibble: 2,394 × 3
   id            word           word_length
   <chr>         <chr>                <int>
 1 book_ch11.txt accouterments          13
 2 book_ch11.txt circumstances          13
 3 book_ch11.txt questioningly          13
 4 book_ch11.txt eavesdropping          13
 5 book_ch11.txt comparatively          13
 6 book_ch11.txt intermarrying          13
 7 book_ch11.txt irretrievably          13
 8 book_ch11.txt architecture           12
 9 book_ch11.txt compositions           12
10 book_ch11.txt conversation           12
# ℹ 2,384 more rows
# ℹ Use `print(n = ...)` to see more rows
```

Longest Sentences

Extract the 10 longest sentences from the book_corpus object. Here's a breakdown of what's happening:

tidy(book_corpus) converts the book_corpus object into a tidy format.

unnest_tokens(sentence, text, token = "regex", pattern = "(?<!\\b\\p{L}r)\\.") separates the text into sentences by splitting on periods. The regular expression "(?<!\\b\\p{L}r)\\." matches periods that are not preceded by a word boundary followed by the letter "r".

select(id, sentence) selects the columns "id" (which indicates which book the sentence comes from) and "sentence".

mutate(sentence_length = nchar(sentence)) adds a new column "sentence_length" that contains the number of characters in each sentence.

arrange(desc(sentence_length)) sorts the sentences in descending order of length, so the longest sentences appear first.

```
> # TODO: 10 longest sentences
> book_sentences <- tidy(book_corpus) %>%
+   unnest_tokens(sentence, text, token = "regex", pattern = "(?<!\\b\\p{L}r)\\.") %>%
+   select(id, sentence) %>%
+   mutate(sentence_length = nchar(sentence)) %>%
+   arrange(desc(sentence_length))
Warning message:
Outer names are only allowed for unnamed scalar atomic inputs
```

```
> book_sentences %>% filter(id == "book_ch1.txt")
# A tibble: 79 × 3
   id           sentence                                                                                  sentence_length
   <chr>        <chr>                                                                                               <int>
 1 book_ch1.txt "  however, i\nam not prone to sensitiveness, and the following of a sense of duty,\nwherever it may le…           377
 2 book_ch1.txt "  the fact that it\nis difficult to aim anything but imprecations accurately by moonlight,\nthat they …           372
 3 book_ch1.txt "\n\nsince we had entered the territory we had not seen a hostile indian,\nand we had, therefore, becom…           365
 4 book_ch1.txt "\n\nin this instance i was, of course, positive that powell was the center\nof attraction, but whether…           347
 5 book_ch1.txt "\n\nthe morning of powell's departure was, like nearly all arizona\nmornings, clear and beautiful; i c…           326
 6 book_ch1.txt "\n\nmy horse was traveling practically unguided as i knew that i had\nprobably less knowledge of the e…           310
 7 book_ch1.txt "\n\ni do not believe that i am made of the stuff which constitutes heroes,\nbecause, in all of the hun…           289
 8 book_ch1.txt "\n\ni was positive now that the trailers were apaches and that they wished\nto capture powell alive fo…           279
 9 book_ch1.txt "\n\ni soon became so drowsy that i could scarcely resist the strong desire\nto throw myself on the flo…           273
10 book_ch1.txt "  i know that the average\nhuman mind will not believe what it cannot grasp, and so i do not\npurpose …           269
# ℹ 69 more rows
# ℹ Use `print(n = ...)` to see more rows
> book_sentences %>% filter(id == "book_ch2.txt")
# A tibble: 52 × 3
   id           sentence                                                                                  sentence_length
   <chr>        <chr>                                                                                               <int>
 1 book_ch2.txt "\n\nfew western wonders are more inspiring than the beauties of an arizona\nmoonlit landscape; the sil…           456
 2 book_ch2.txt "  i reasoned with\nmyself that i had lain helpless for many hours within the cave, yet\nnothing had mo…           394
 3 book_ch2.txt "\n\nto be held paralyzed, with one's back toward some horrible and unknown\ndanger from the very sound…           373
 4 book_ch2.txt "  fear is a relative term and so\ni can only measure my feelings at that time by what i had experience…           359
 5 book_ch2.txt "\n\nlate in the afternoon my horse, which had been standing with dragging\nrein before the cave, start…           352
 6 book_ch2.txt "  my\nfirst thought was, is this then death!  have i indeed passed over\nforever into that other life!…           275
 7 book_ch2.txt "\n\nfrom then until possibly midnight all was silence, the silence of the\ndead; then, suddenly, the a…           264
 8 book_ch2.txt "  my only alternative seemed to lie\nin flight and my decision was crystallized by a recurrence of the…           244
 9 book_ch2.txt "  there also\ncame to my nostrils a faintly pungent odor, and i could only assume\nthat i had been ove…           217
10 book_ch2.txt "\n\ni had not long to wait before a stealthy sound apprised me of their\nnearness, and then a war-bonn…           213
# ℹ 42 more rows
# ℹ Use `print(n = ...)` to see more rows
> book_sentences %>% filter(id == "book_ch3.txt")
# A tibble: 86 × 3
   id           sentence                                                                                  sentence_length
   <chr>        <chr>                                                                                               <int>
 1 book_ch3.txt "  the throwing down of his weapons and the\nwithdrawing of his troop before his advance toward me woul…           408
 2 book_ch3.txt "  he sat his mount as we\nsit a horse, grasping the animal's barrel with his lower limbs, while\nthe h…           336
 3 book_ch3.txt "  their eyes were set at the extreme sides of their heads a\ntrifle above the center and protruded in …           335
 4 book_ch3.txt "\n\nand his mount!  how can earthly words describe it!  it towered ten feet\nat the shoulder; had four…           305
 5 book_ch3.txt "\n\nthe respite my unexpected agility had given me permitted me to\nformulate plans for the immediate …           288
 6 book_ch3.txt "\nthe result is that they are infinitely less agile and less powerful, in\nproportion to their weight,…           283
 7 book_ch3.txt "\n\ncoming, as they did, over the soft and soundless moss, which covers\npractically the entire surfac…           280
 8 book_ch3.txt "  but the little sound caused me to\nturn, and there upon me, not ten feet from my breast, was the poi…           263
 9 book_ch3.txt "\n\ninstead of progressing in a sane and dignified manner, my attempts to\nwalk resulted in a variety …           254
10 book_ch3.txt "\n\nbehind this first charging demon trailed nineteen others, similar in\nall respects, but, as i lear…           251
# ℹ 76 more rows
# ℹ Use `print(n = ...)` to see more rows

> book_sentences %>% filter(id == "book_ch4.txt")
# A tibble: 69 × 3
   id           sentence                                                                                  sentence_length
   <chr>        <chr>                                                                                               <int>
 1 book_ch4.txt "\n\ni saw no signs of extreme age among them, nor is there any appreciable\ndifference in their appear…           449
 2 book_ch4.txt "  they first repeated\nthe word \"sak\" a number of times, and then tars tarkas made several\njumps, r…           411
 3 book_ch4.txt "\n\nwhat struck me as most remarkable about this assemblage and the hall in\nwhich they were congregat…           410
 4 book_ch4.txt "\n\nmy exhibition had been witnessed by several hundred lesser martians,\nand they immediately broke i…           371
 5 book_ch4.txt "\n\nthe room was well lighted by a number of large windows and was\nbeautifully decorated with mural p…           357
 6 book_ch4.txt "  owing to\nthe waning resources of the planet it evidently became necessary to\ncounteract the increa…           353
 7 book_ch4.txt "\nevidently, then, there were other denizens on mars than the wild and\ngrotesque creatures into whose…           306
 8 book_ch4.txt "\n\nas he banged me down upon my feet his face was bent close to mine and i\ndid the only thing a gent…           285
 9 book_ch4.txt "  toward the center\nof the city was a large plaza, and upon this and in the buildings\nimmediately su…           276
10 book_ch4.txt "  had the men been strangers, and therefore\nunable to exchange names, they would have silently exchan…           271
# ℹ 59 more rows
# ℹ Use `print(n = ...)` to see more rows
> book_sentences %>% filter(id == "book_ch5.txt")
# A tibble: 46 × 3
   id           sentence                                                                                  sentence_length
   <chr>        <chr>                                                                                               <int>
 1 book_ch5.txt "  the nights are either brilliantly illumined or\nvery dark, for if neither of the two moons of mars h…           392
 2 book_ch5.txt "\n\nthis last device produces an intensely brilliant far-reaching white\nlight, but as the natural oil…           367
 3 book_ch5.txt "  i could not but\nwonder what this ferocious-looking monstrosity might do when left alone\nin such cl…           348
 4 book_ch5.txt "  and it is well that\nnature has so graciously and abundantly lighted the martian night, for\nthe gre…           346
 5 book_ch5.txt "  it came, as i later discovered, not from\nan animal, as there is only one mammal on mars and that on…           312
 6 book_ch5.txt "  the work had evidently been wrought by a master hand,\nso subtle the atmosphere, so perfect the tech…           280
 7 book_ch5.txt "\n\nboth of mars' moons are vastly nearer her than is our moon to earth;\nthe nearer moon being but ab…           280
 8 book_ch5.txt "\nthe nearer moon of mars makes a complete revolution around the planet\nin a little over seven and on…           275
 9 book_ch5.txt "  across the threshold lay stretched the\nsleepless guardian brute, just as i had last seen him on the…           268
10 book_ch5.txt "  this\ngirl alone, among all the green martians with whom i came in contact,\ndisclosed characteristi…           267
# ℹ 36 more rows
# ℹ Use `print(n = ...)` to see more rows
> book_sentences %>% filter(id == "book_ch6.txt")
# A tibble: 50 × 3
   id           sentence                                                                                  sentence_length
   <chr>        <chr>                                                                                               <int>
 1 book_ch6.txt "  my beast had an advantage\nin his first hold, having sunk his mighty fangs far into the breast of\nh…           431
 2 book_ch6.txt "\n\ni am ever willing to stand and fight when the odds are not too\noverwhelmingly against me, but in …           378
 3 book_ch6.txt "\n\ni had at least two friends on mars; a young woman who watched over me\nwith motherly solicitude, a…           343
 4 book_ch6.txt "\n\nsuddenly i came to myself and, with that strange instinct which seems\never to prompt me to my dut…           338
 5 book_ch6.txt "\n\nit is true i held the cudgel, but what could i do with it against his\nfour great arms?  even shou…           297
 6 book_ch6.txt "  evidently\ndevoid of all the finer sentiments of friendship, love, or affection,\nthese people fairl…           291
 7 book_ch6.txt "\n\ni was standing near the window and i knew that once in the street i\nmight gain the plaza and safe…           265
 8 book_ch6.txt "  with a shriek of fear the ape\nwhich held me leaped through the open window, but its mate closed in …           256
 9 book_ch6.txt "  i glimpsed him just before he reached the doorway and the\nsight of him, now roaring as he perceived…           253
10 book_ch6.txt "  they seemed to be deep in argument, and\nfinally one of them addressed me, but remembering my ignora…           248
# ℹ 40 more rows
# ℹ Use `print(n = ...)` to see more rows
```

```
> book_sentences %>% filter(id == "book_ch7.txt")
# A tibble: 62 × 3
   id           sentence                                                                                  sentence_length
   <chr>        <chr>                                                                                               <int>
 1 book_ch7.txt "  between these walls the\nlittle martians scampered, wild as deer; being permitted to run the\nfull l…          432
 2 book_ch7.txt "\nchild-raising on mars\n\n\nafter a breakfast, which was an exact replica of the meal of the\nprecedi…          372
 3 book_ch7.txt "\n\ni do not mean that the adult martians are unnecessarily or\nintentionally cruel to the young, but …          332
 4 book_ch7.txt "  entirely\nunknown to their mothers, who, in turn, would have difficulty in\npointing out the fathers…          277
 5 book_ch7.txt "  they were\nnot wanted, as their offspring might inherit and transmit the tendency\nto prolonged incu…          274
 6 book_ch7.txt "\n\nevery one but myself--men, women, and children--were heavily armed, and\nat the tail of each chari…          271
 7 book_ch7.txt "\"  i saw that he wanted me to repeat my performance of\nyesterday for the edification of lorquas ptom…          270
 8 book_ch7.txt "  it is the universal\nlanguage of mars, through the medium of which the higher and lower\nanimals of …          268
 9 book_ch7.txt "  as i later learned, they had been to the subterranean\nvaults in which the eggs were kept and had tr…          264
10 book_ch7.txt "\n\nsola's duties were now doubled, as she was compelled to care for the\nyoung martian as well as for…          262
# ℹ 52 more rows
# ℹ Use `print(n = ...)` to see more rows
> book_sentences %>% filter(id == "book_ch8.txt")
# A tibble: 59 × 3
   id           sentence                                                                                  sentence_length
   <chr>        <chr>                                                                                               <int>
 1 book_ch8.txt "  for example,\na proportion of them, always the best marksmen, direct their fire\nentirely upon the w…          454
 2 book_ch8.txt "\n\ninstantly the scene changed as by magic; the foremost vessel swung\nbroadside toward us, and bring…          450
 3 book_ch8.txt "\n\nas sola and i entered the plaza a sight met my eyes which filled my\nwhole being with a great surg…          409
 4 book_ch8.txt "\n\nsola and i had entered a building upon the front of the city, in fact,\nthe same one in which i ha…          346
 5 book_ch8.txt "  i could not fathom the seeming hallucination, nor could i\nfree myself from it; but somewhere in the…          344
 6 book_ch8.txt "  whether they had\ndiscovered us or simply were looking at the deserted city i could not\nsay, but in…          343
 7 book_ch8.txt "  this operation required several\nhours, during which time a number of the chariots were requisitione…          336
 8 book_ch8.txt "  the sight was\nawe-inspiring in the extreme as one contemplated this mighty floating\nfuneral pyre, …          331
 9 book_ch8.txt "\n\nas the craft neared the building, and just before she struck, the\nmartian warriors swarmed upon h…          305
10 book_ch8.txt "  it had never\nbeen given me to see such deadly accuracy of aim, and it seemed as\nthough a little fi…          295
# ℹ 49 more rows
# ℹ Use `print(n = ...)` to see more rows
> book_sentences %>% filter(id == "book_ch9.txt")
# A tibble: 46 × 3
   id           sentence                                                                                  sentence_length
   <chr>        <chr>                                                                                               <int>
 1 book_ch9.txt "  oh, it is one continual, awful period of bloodshed from\nthe time we break the shell until we gladly…          315
 2 book_ch9.txt "  with this added incentive i nearly drove sola distracted by\nmy importunities to hasten on my educat…          285
 3 book_ch9.txt "\n\n\"when,\" asked one of the women, \"will we enjoy the death throes of the\nred one? or does lorqua…          281
 4 book_ch9.txt "  customs have\nbeen handed down by ages of repetition, but the punishment for ignoring\na custom is a…          280
 5 book_ch9.txt "\n\nthe training of myself and the young martians was conducted solely by\nthe women, who not only att…          274
 6 book_ch9.txt "  i could not but note the\nunnecessary harshness and brutality with which her guards treated her;\nso…          274
 7 book_ch9.txt "  after they had retired for\nthe night it was customary for the adults to carry on a desultory\nconve…          274
 8 book_ch9.txt "\ni knew that she was fond of me, and now that i had discovered that she\nhated cruelty and barbarity …          261
 9 book_ch9.txt "  they live at peace with all their\nfellows, except when duty calls upon them to make war, while we a…          260
10 book_ch9.txt "\n\ni did not see the prisoner again for several days subsequent to our\nfirst encounter, and then onl…          245
# ℹ 36 more rows
# ℹ Use `print(n = ...)` to see more rows
```

```
> book_sentences %>% filter(id == "book_ch10.txt")
# A tibble: 108 × 3
   id           sentence                                                                                  sentence_length
   <chr>        <chr>                                                                                               <int>
 1 book_ch10.txt "\n\nwhat words of moment were to have fallen from his lips were never\nspoken, as just then a young w…          431
 2 book_ch10.txt "  numerous brilliantly\ncolored and strangely formed wild flowers dotted the ravines and from\nthe su…          416
 3 book_ch10.txt "  i saw\nthat the body of my dead antagonist had been stripped, and i read in\nthe menacing yet respe…          416
 4 book_ch10.txt "  what strange manner\nof man are you, that you consort with the green men, though your form\nis that…          405
 5 book_ch10.txt "\"\n\n\"then you too are a prisoner?  but why, then, those arms and the\nregalia of a tharkian chieft…          393
 6 book_ch10.txt "  realizing that i was a somewhat favored character, and also\nconvinced that the warriors did not kn…          387
 7 book_ch10.txt "  i\nwas soon successful as her injuries amounted to little more than an\nordinary nosebleed, and whe…          353
 8 book_ch10.txt "\n\nthe reason for the whole attitude displayed toward me was now apparent;\ni had won my spurs, so t…          345
 9 book_ch10.txt "  i could not resist the\nludicrousness of the spectacle, and holding my sides i rocked back and\nfor…          336
10 book_ch10.txt "\"\n\nordinarily i am not given to long speeches, nor ever before had i\ndescended to bombast, but i …          334
# ℹ 98 more rows
# ℹ Use `print(n = ...)` to see more rows
> book_sentences %>% filter(id == "book_ch11.txt")
# A tibble: 77 × 3
   id           sentence                                                                                  sentence_length
   <chr>        <chr>                                                                                               <int>
 1 book_ch11.txt "\nduring the ages of hardships and incessant warring between their own\nvarious races, as well as wit…          521
 2 book_ch11.txt "\n\n\"because, john carter,\" she replied, \"nearly every planet and star\nhaving atmospheric conditi…          492
 3 book_ch11.txt "  do not tell me that you have thus returned!  they would\nkill you horribly anywhere upon the surfac…          346
 4 book_ch11.txt "\n\na similar wave of feeling seemed to stir her; she drew away from me\nwith a sigh, and with her ea…          341
 5 book_ch11.txt "  i can readily perceive that you are not of\nthe barsoom of today; you are like us, yet different--b…          336
 6 book_ch11.txt "\n\nthe shores of the ancient seas were dotted with just such cities, and\nlesser ones, in diminishin…          328
 7 book_ch11.txt "\n\n\"and whereto, then, would your prisoner escape should you leave her,\nunless it was to follow yo…          308
 8 book_ch11.txt "\n\nthese three great divisions of the higher martians had been forced into\na mighty alliance as the…          308
 9 book_ch11.txt "\n\nthese ancient martians had been a highly cultivated and literary race,\nbut during the vicissitud…          293
10 book_ch11.txt "\nonly in the valley dor, where the river iss empties into the lost sea\nof korus, is there supposed …          287
# ℹ 67 more rows
# ℹ Use `print(n = ...)` to see more rows
```

To access the text content of a document in the quanteda package, we can use the $content operator. In the code you provided, btext is a single document from the book_corpus object, and $content retrieves the text content of that document. Therefore, btext[1]$content would give you the text content of the first document in the book_corpus object.

```
> btext <- book_corpus[[1]]
> btext
<<PlainTextDocument>>
Metadata:  7
Content:  chars: 13962
> btext[1]
$content
  [1] ""
  [2] "ON THE ARIZONA HILLS"
  [3] ""
  [4] ""
  [5] "I am a very old man; how old I do not know.  Possibly I am a hundred,"
  [6] "possibly more; but I cannot tell because I have never aged as other"
  [7] "men, nor do I remember any childhood.  So far as I can recollect I have"
  [8] "always been a man, a man of about thirty.  I appear today as I did"
  [9] "forty years and more ago, and yet I feel that I cannot go on living"
 [10] "forever; that some day I shall die the real death from which there is"
 [11] "no resurrection.  I do not know why I should fear death, I who have"
 [12] "died twice and am still alive; but yet I have the same horror of it as"
 [13] "you who have never died, and it is because of this terror of death, I"
 [14] "believe, that I am so convinced of my mortality."
 [15] ""
 [16] "And because of this conviction I have determined to write down the"
 [17] "story of the interesting periods of my life and of my death.  I cannot"
```

DocumentTermMatrix is a R function that generates a matrix containing the number of occurrences of each term (word) in each document in a specified corpus. Book_corpus is a corpus of 11 documents (probably the 11 chapters of a book), and bookDTM is a DocumentTermMatrix object containing 5003 unique terms (words) across all 11 documents. The matrix contains 9204 non-zero entries and has an 83% sparsity, indicating that many of the phrases are not included in all of the texts. The term frequency (tf) weighting is applied, which counts the number of times each term appears in each document.

```
> bookDTM <- DocumentTermMatrix(book_corpus)
> bookDTM
<<DocumentTermMatrix (documents: 11, terms: 5003)>>
Non-/sparse entries: 9204/45829
Sparsity           : 83%
```

The inspect() function is used to print the DocumentTermMatrix object's contents in a readable format. It displays a table with the number of occurrences of each term in each document in this scenario. The rows represent documents, while the columns represent terms. The numbers in the table show how many times each term appears in each document.

```
> inspect(bookDTM)
<<DocumentTermMatrix (documents: 11, terms: 5003)>>
Non-/sparse entries: 9204/45829
Sparsity           : 83%
Maximal term length: 19
Weighting          : term frequency (tf)
Sample             :
             Terms
Docs          and but for had that the upon was which with
  book_ch1.txt  90  13  20  25   50 190   12  36    20   25
  book_ch10.txt 113 18  34  41   50 193   18  54    15   26
  book_ch11.txt 82  14   9  29   37 119   13  18    13   23
  book_ch2.txt  58  13  17  20   19 137   13  31    12   10
  book_ch3.txt  92  15  18  19   26 166   20  36    34   20
  book_ch4.txt  68  21  10  22   19 148   10  26    16   20
  book_ch5.txt  56  14  13  14   11  96    8  22     9    8
  book_ch6.txt  53  10  14  28   11 117   13  17    11   24
  book_ch7.txt  64   9  18  16   12 166    7  17    22   12
  book_ch8.txt  81   7  15  24   11 184   22  32    17   11
```

The str() function is used to output the DocumentTermMatrix object's structure. It demonstrates that the object is a list composed of six elements: i, j, v, nrow, ncol, and dimnames. The i and j elements are integer vectors that describe the row and column indices of the non-zero entries in the document-term matrix's sparse matrix representation. The v element is a numeric vector containing the non-zero elements' values. The elements nrow and ncol specify the number of rows and columns in the matrix. The dimnames element is a list that contains the row and column names. This is a DocumentTermMatrix object that is also a simple_triplet_matrix, as indicated by the class attribute.

```
> str(bookDTM)
List of 6
 $ i       : int [1:9204] 1 1 1 1 1 1 1 1 1 1 ...
 $ j       : int [1:9204] 36 37 38 41 42 60 68 74 78 81 ...
 $ v       : num [1:9204] 1 1 1 1 7 1 1 4 1 1 ...
 $ nrow    : int 11
 $ ncol    : int 5003
 $ dimnames:List of 2
  ..$ Docs : chr [1:11] "book_ch1.txt" "book_ch10.txt" "book_ch11.txt" "book_ch2.txt" ...
  ..$ Terms: chr [1:5003] "'gentleman'" "\"and" "\"as" "\"because," ...
 - attr(*, "class")= chr [1:2] "DocumentTermMatrix" "simple_triplet_matrix"
 - attr(*, "weighting")= chr [1:2] "term frequency" "tf"
```

We have created a data frame with a single column "content" which contains the text of the first document in your book_corpus.

```
> book_df <- data.frame(btext[1])
> book_df[1]
                                                                    content
1
2                                                          ON THE ARIZONA HILLS
3
4
5      I am a very old man; how old I do not know.  Possibly I am a hundred,
6        possibly more; but I cannot tell because I have never aged as other
7    men, nor do I remember any childhood.  So far as I can recollect I have
8         always been a man, a man of about thirty.  I appear today as I did
9        forty years and more ago, and yet I feel that I cannot go on living
10     forever; that some day I shall die the real death from which there is
11       no resurrection.  I do not know why I should fear death, I who have
12    died twice and am still alive; but yet I have the same horror of it as
13     you who have never died, and it is because of this terror of death, I
14                          believe, that I am so convinced of my mortality.
15
16        And because of this conviction I have determined to write down the
17    story of the interesting periods of my life and of my death.  I cannot
18        explain the phenomena; I can only set down here in the words of an
19        ordinary soldier of fortune a chronicle of the strange events that
20   befell me during the ten years that my dead body lay undiscovered in an
21                                                                Arizona cave.
22
23    I have never told this story, nor shall mortal man see this manuscript
24     until after I have passed over for eternity.  I know that the average
25         human mind will not believe what it cannot grasp, and so I do not
26     purpose being pilloried by the public, the pulpit, and the press, and
27        held up as a colossal liar when I am but telling the simple truths
28       which some day science will substantiate.  Possibly the suggestions
```

The removeNumPunct function is a user-defined function that takes a string x as input and removes any characters that are not letters or spaces from the string using regular expressions. It specifically use the gsub function to substitute an empty string for each character that fits the regular pattern [[:alpha:][:space:]]*. [[:alpha:][:space:]]* matches any sequence of characters that are not letters or spaces (i.e., punctuation and numbers) and replaces it with nothing.

Corpus Cleaning - Data Wrangling

Removing numbers and punctuation

```
> # remove punctuation and numbers
> removeNumPunct <- function(x) gsub("[^[:alpha:][:space:]]*", "", x)
> removeNumPunct
function(x) gsub("[^[:alpha:][:space:]]*", "", x)
```

The following code generates a new variable called bookcl by transforming book_corpus with the tm_map() function from the tm package. It converts the corpus to plain text using the content_transformer() function and then removes all non-alphabetic and non-space characters from the text using the removeNumPunct() function. The resulting corpus bookcl has the same 11 documents as book_corpus, but without the punctuation and numbers.

```
> bookcl <- tm::tm_map(book_corpus, content_transformer(removeNumPunct))
> bookcl
<<VCorpus>>
Metadata:  corpus specific: 0, document level (indexed): 0
Content:  documents: 11
> str(bookcl)
Classes 'VCorpus', 'Corpus'  hidden list of 3
 $ content:List of 11
  ..$ :List of 2
  .. ..$ content: chr [1:267] "" "ON THE ARIZONA HILLS" "" "" ...
  .. ..$ meta   :List of 7
  .. .. ..$ author       : chr(0)
  .. .. ..$ datetimestamp: POSIXlt[1:1], format: "2023-05-06 18:40:27"
  .. .. ..$ description  : chr(0)
  .. .. ..$ heading      : chr(0)
  .. .. ..$ id           : chr "book_ch1.txt"
  .. .. ..$ language     : chr "en"
  .. .. ..$ origin       : chr(0)
```

```
> inspect(bookcl)
<<VCorpus>>
Metadata:  corpus specific: 0, document level (indexed): 0
Content:  documents: 11

[[1]]
<<PlainTextDocument>>
Metadata:  7
Content:  chars: 13711

[[2]]
<<PlainTextDocument>>
Metadata:  7
Content:  chars: 18669

[[3]]
<<PlainTextDocument>>
Metadata:  7
Content:  chars: 12812

[[4]]
<<PlainTextDocument>>
Metadata:  7
Content:  chars: 8844

[[5]]
<<PlainTextDocument>>
Metadata:  7
Content:  chars: 13921

[[6]]
<<PlainTextDocument>>
Metadata:  7
Content:  chars: 11591
```

- To lower case

The code below applies the tolower() function to all documents in the bookcl corpus using tm_map() from the tm package, which converts all characters to lower case. The resulting corpus is stored in the variable booklow.

```
> # with lower case
> booklow <- tm::tm_map(bookcl, content_transformer(tolower))
> booklow
<<VCorpus>>
Metadata:  corpus specific: 0, document level (indexed): 0
Content:  documents: 11
> str(booklow)
Classes 'VCorpus', 'Corpus'  hidden list of 3
 $ content:List of 11
  ..$ :List of 2
  .. ..$ content: chr [1:267] "" "on the arizona hills" "" "" ...
  .. ..$ meta   :List of 7
  .. .. ..$ author       : chr(0)
  .. .. ..$ datetimestamp: POSIXlt[1:1], format: "2023-05-06 18:40:27"
  .. .. ..$ description  : chr(0)
  .. .. ..$ heading      : chr(0)
  .. .. ..$ id           : chr "book_ch1.txt"
  .. .. ..$ language     : chr "en"
  .. .. ..$ origin       : chr(0)
  .. .. ..- attr(*, "class")= chr "TextDocumentMeta"
  .. ..- attr(*, "class")= chr [1:2] "PlainTextDocument" "TextDocument"
  ..$ :List of 2
  .. ..$ content: chr [1:358] "" "champion and chief" "" "" ...
  .. ..$ meta   :List of 7
  .. .. ..$ author       : chr(0)
  .. .. ..$ datetimestamp: POSIXlt[1:1], format: "2023-05-06 18:40:27"
  .. .. ..$ description  : chr(0)
  .. .. ..$ heading      : chr(0)
  .. .. ..$ id           : chr "book_ch10.txt"
  .. .. ..$ language     : chr "en"
  .. .. ..$ origin       : chr(0)
  .. .. ..- attr(*, "class")= chr "TextDocumentMeta"
  .. ..- attr(*, "class")= chr [1:2] "PlainTextDocument" "TextDocument"
  ..$ :List of 2
```

Computing DTM

From the preprocessed corpus booklow, the procedure below generates a Document-Term Matrix (DTM). The DTM contains the frequency of occurrence of each term (word) in each corpus document.
According to the output, the DTM contains 11 rows and 3866 columns, which match to the number of documents and unique phrases in the corpus. The DTM's i, j, and v properties are three integer vectors that represent the row indices, column indices, and values of non-zero entries, respectively. The DTM's dimnames attribute is a 2-length list holding the names of the documents and words. The DTM's weighting attribute reveals the weighting system used to construct the matrix's elements, which is term frequency (tf) in this case.
Finally, the inspect function is used to display a sample of the DTM that shows the frequency of occurrence of the top ten terms in each document. The output shows that the DTM is very sparse, with a sparsity of 81%, implying that the majority of the matrix elements are zero.

```
> # computing DTM
> bookDTM <- DocumentTermMatrix(booklow)
> bookDTM
<<DocumentTermMatrix (documents: 11, terms: 3866)>>
Non-/sparse entries: 8205/34321
Sparsity           : 81%
Maximal term length: 17
Weighting          : term frequency (tf)
> str(bookDTM)
List of 6
 $ i       : int [1:8205] 1 1 1 1 1 1 1 1 1 1 ...
 $ j       : int [1:8205] 2 3 18 25 30 32 35 57 59 66 ...
 $ v       : num [1:8205] 1 7 1 1 4 1 1 1 1 3 ...
 $ nrow    : int 11
 $ ncol    : int 3866
 $ dimnames:List of 2
  ..$ Docs : chr [1:11] "book_ch1.txt" "book_ch10.txt" "book_ch11.txt" "book_ch2.txt" ...
  ..$ Terms: chr [1:3866] "ability" "able" "about" "above" ...
 - attr(*, "class")= chr [1:2] "DocumentTermMatrix" "simple_triplet_matrix"
 - attr(*, "weighting")= chr [1:2] "term frequency" "tf"
> inspect(bookDTM)
<<DocumentTermMatrix (documents: 11, terms: 3866)>>
Non-/sparse entries: 8205/34321
Sparsity           : 81%
Maximal term length: 17
Weighting          : term frequency (tf)
Sample             :
              Terms
Docs           and but for had that the upon was which with
  book_ch1.txt  93  13  20  27   50 190   12  38    20   25
  book_ch10.txt 117 20  34  41   50 193   19  54    16   26
  book_ch11.txt  89  14  10  29   38 122   13  19    13   23
  book_ch2.txt   59  13  17  20   20 137   13  33    13   10
  book_ch3.txt   94  17  18  19   26 166   20  37    34   20
  book_ch4.txt   70  21  10  22   19 148   11  26    17   20
  book_ch5.txt   56  14  13  14   11  96    8  22     9    8
  book_ch6.txt   57  10  15  29   11 117   13  18    12   24
  book_ch7.txt   67   9  18  16   12 166    7  18    24   12
  book_ch8.txt   84   7  15  24   11 184   22  32    17   11
```

Removing the stopwords

The myStopwords vector provides a list of frequent English stop words, and we removed these words from the booklow corpus using the tm_map function from the tm package. Then we used inspect(bookStop[1]) to inspect the first document in the bookStop corpus, which revealed that the stop words had been removed from the text.

```
> # stopwords
> myStopwords <- c(tm::stopwords("en"))
> myStopwords
  [1] "i"          "me"         "my"         "myself"     "we"         "our"        "ours"       "ourselves"  "you"        "your"
 [11] "yours"      "yourself"   "yourselves" "he"         "him"        "his"        "himself"    "she"        "her"        "hers"
 [21] "herself"    "it"         "its"        "itself"     "they"       "them"       "their"      "theirs"     "themselves" "what"
 [31] "which"      "who"        "whom"       "this"       "that"       "these"      "those"      "am"         "is"         "are"
 [41] "was"        "were"       "be"         "been"       "being"      "have"       "has"        "had"        "having"     "do"
 [51] "does"       "did"        "doing"      "would"      "should"     "could"      "ought"      "i'm"        "you're"     "he's"
 [61] "she's"      "it's"       "we're"      "they're"    "i've"       "you've"     "we've"      "they've"    "i'd"        "you'd"
 [71] "he'd"       "she'd"      "we'd"       "they'd"     "i'll"       "you'll"     "he'll"      "she'll"     "we'll"      "they'll"
 [81] "isn't"      "aren't"     "wasn't"     "weren't"    "hasn't"     "haven't"    "hadn't"     "doesn't"    "don't"      "didn't"
 [91] "won't"      "wouldn't"   "shan't"     "shouldn't"  "can't"      "cannot"     "couldn't"   "mustn't"    "let's"      "that's"
[101] "who's"      "what's"     "here's"     "there's"    "when's"     "where's"    "why's"      "how's"      "a"          "an"
[111] "the"        "and"        "but"        "if"         "or"         "because"    "as"         "until"      "while"      "of"
[121] "at"         "by"         "for"        "with"       "about"      "against"    "between"    "into"       "through"    "during"
[131] "before"     "after"      "above"      "below"      "to"         "from"       "up"         "down"       "in"         "out"
[141] "on"         "off"        "over"       "under"      "again"      "further"    "then"       "once"       "here"       "there"
[151] "when"       "where"      "why"        "how"        "all"        "any"        "both"       "each"       "few"        "more"
[161] "most"       "other"      "some"       "such"       "no"         "nor"        "not"        "only"       "own"        "same"
[171] "so"         "than"       "too"        "very"
> bookStop <- tm::tm_map(booklow, tm::removeWords, myStopwords)
> inspect(bookStop[1])
<<VCorpus>>
Metadata:  corpus specific: 0, document level (indexed): 0
Content:  documents: 1

[[1]]
<<PlainTextDocument>>
Metadata:  7
Content:  chars: 9812
```

Calculating the TDM

The bookStopTDM TermDocumentMatrix has 3768 distinct terms and 11 documents. It has 7405 non-zero entries out of a possible 34043 entries, indicating that the matrix is sparse (82% sparsity). The maximum length of a word is 17 characters. The matrix's weighting is based on term frequency (tf), therefore each row indicates the number of times a term appears in a document.

```
> # TDM again
> bookStopTDM <- tm::TermDocumentMatrix(bookStop)
> bookStopTDM
<<TermDocumentMatrix (terms: 3768, documents: 11)>>
Non-/sparse entries: 7405/34043
Sparsity           : 82%
Maximal term length: 17
Weighting          : term frequency (tf)
```

Frequent terms

The tm package's findFreqTerms() method is used to find the most common terms in a term-document matrix (TDM). It takes a TDM object as input and a threshold lowfreq parameter that specifies the least frequency required for a phrase to be regarded "frequent."

bookStopTDM is a TDM object derived from a text corpus that has had stop words deleted and the lowfreq parameter set to 5. As a result, freqTerms contains a list of all terms in the corpus that appear at least 5 times. These are the most likely and relevant terms in the corpus, and they may be valuable for additional analysis and interpretation.

```
> # frequent terms
> freqTerms <- tm::findFreqTerms(bookStopTDM, lowfreq = 5)
> freqTerms
  [1] "ability"      "across"       "act"          "advanced"     "affection"    "afterward"    "age"          "ages"
  [9] "aid"          "air"          "almost"       "alone"        "also"         "always"       "among"        "ancient"
 [17] "animal"       "animals"      "another"      "answered"     "answering"    "ape"          "appearance"   "appeared"
 [25] "approach"     "approached"   "approaching"  "arizona"      "arm"          "arms"         "around"       "asked"
 [33] "attempt"      "attention"    "attitude"     "audience"     "away"         "back"         "barsoom"      "battle"
 [41] "bearing"      "beast"        "beautiful"    "became"       "become"       "behind"       "believe"      "beside"
 [49] "better"       "beyond"       "blow"         "bodies"       "body"         "bottom"       "breast"       "bring"
 [57] "broke"        "brought"      "brute"        "building"     "buildings"    "call"         "came"         "can"
 [65] "captive"      "carried"      "carry"        "carter"       "catch"        "caught"       "cause"        "caused"
 [73] "cautiously"   "cavalcade"    "cave"         "ceased"       "center"       "chamber"      "chariots"     "chieftain"
 [81] "chieftains"   "children"     "city"         "clear"        "cliff"        "close"        "cold"         "color"
 [89] "come"         "common"       "community"    "conditions"   "considerable" "continued"    "conversation" "convinced"
 [97] "council"      "course"       "craft"        "creature"     "creatures"    "creeping"     "crude"        "cruel"
[105] "cudgel"       "customs"      "dark"         "darkness"     "day"          "daylight"     "days"         "dead"
[113] "deadly"       "death"        "decided"      "decks"        "dejah"        "deserted"     "desire"       "determined"
[121] "direction"    "discovered"   "distance"     "doubt"        "drew"         "duty"         "early"        "ears"
[129] "earth"        "earthly"      "easily"       "education"    "effort"       "eggs"         "either"       "enclosure"
[137] "encounter"    "end"          "enormous"     "enough"       "entered"      "entire"       "entirely"     "entrance"
[145] "escape"       "even"         "ever"         "every"        "everything"   "evidently"    "except"       "exchanged"
[153] "explained"    "expression"   "extreme"      "extremely"    "eyes"         "face"         "fact"         "faint"
[161] "fair"         "fairly"       "fallen"       "far"          "fear"         "feeling"      "feet"         "fell"
[169] "fellow"       "fellows"      "felt"         "female"       "females"      "ferocious"    "ferocity"     "fifty"
[177] "fight"        "figure"       "filled"       "finally"      "fire"         "first"        "five"         "flight"
[185] "floor"        "follow"       "followed"     "following"    "food"         "force"        "forced"       "form"
[193] "former"       "forth"        "forty"        "found"        "four"         "friend"       "full"         "fully"
[201] "furs"         "gained"       "gave"         "gazed"        "general"      "gentleman"    "girl"         "given"
[209] "good"         "grasping"     "great"        "green"        "ground"       "guardian"     "guards"       "hair"
[217] "hand"         "hands"        "hatched"      "head"         "heard"        "heart"        "heavens"      "height"
[225] "held"         "hideous"      "high"         "higher"       "highly"       "hills"        "home"         "hope"
[233] "horrible"     "horse"        "hours"        "however"      "huge"         "human"        "hundred"      "immediate"
[241] "immediately"  "incubator"    "indeed"       "individual"   "instant"      "iss"          "jed"          "john"
[249] "just"         "kill"         "kind"         "kindliness"   "knew"         "know"         "land"         "landscape"
[257] "language"     "large"        "last"         "later"        "laugh"        "laughter"     "lay"          "learn"
[265] "learned"      "least"        "leave"        "led"          "ledge"        "left"         "legs"         "length"
[273] "less"         "lesser"       "level"        "life"         "light"        "like"         "line"         "lips"
[281] "little"       "live"         "located"      "long"         "longer"       "look"         "looked"       "looking"
[289] "lorquas"      "lost"         "love"         "low"          "lower"        "made"         "make"         "making"
[297] "male"         "man"          "manner"       "many"         "mars"         "martian"      "martians"     "matter"
[305] "may"          "means"        "meet"         "men"          "merely"       "met"          "metal"        "might"
```

After stopword removal, the output below shows the term frequency of each word in the first document in the bookStop corpus. The tm::The termFreq function computes the frequency with which each term appears in a document.

For example, the term "able" appears just once in the document, while "account" and "accurately" appear twice. The term "arizona" appears five times in the paper, while the words "cartridge", "cartridges", "company", "captain", "captains", and "cave" appear only once each.

```
> booktf <- tm::termFreq(bookStop[[1]])
> booktf
```

able	account	accurately	across	acted	acts	advent	adventures	afternoon
1	1	1	4	1	1	1	1	1
aged	ago	agreed	ahead	aid	aim	alive	almost	alone
1	1	1	2	1	1	2	1	1
already	alternative	always	among	animals	another	antelope	anything	apaches
1	1	3	1	2	1	1	1	3
apartments	appear	apprehension	arizona	arm	armed	arming	army	arose
1	1	1	5	1	1	1	3	1
around	arrows	assure	attacked	attempt	attention	attraction	attributed	average
3	3	1	1	1	1	1	1	1
await	back	backward	balls	bathed	beast	beautiful	became	become
1	3	1	1	1	1	1	1	2
befell	believe	belt	belts	best	bestowed	better	bidding	body
2	3	1	1	1	1	3	1	5
borne	bottom	bows	braves	brief	bright	brisk	bristling	broad
1	1	1	1	1	1	1	1	1
broke	brought	burros	came	camp	can	canteen	canter	captain
1	1	1	3	3	3	1	1	2
captains	capture	carbine	careless	carry	carter	cartridge	cartridges	casually
1	1	1	1	1	2	1	1	1
catch	catching	cavalry	cave	center	certain	chamber	chances	characteristic
3	1	1	8	2	1	1	1	1
charging	chase	childhood	chronicle	civil	civilization	claim	clear	cliff
1	1	1	2	1	1	2	2	2
close	clustered	clutches	colossal	colt	come	commenced	commission	company
3	1	1	1	1	1	1	1	1
comparatively	confederate	conjured	consternation	constituted	constitutes	continue	continued	continuing
2	2	1	1	1	1	1	2	1
continuously	conviction	convince	convinced	country	course	cowardice	creeping	crept
1	1	1	5	1	1	1	1	1
crude	cunning	customary	cut	dangerous	dangers	dark	darkness	dash
2	1	1	1	1	1	1	1	1
dawn	day	daylight	dead	deadly	death	debouched	decided	decorations
1	3	1	4	1	7	1	1	1
deepest	defend	defile	dense	departure	deserted	desire	determined	diameter
1	1	1	1	1	1	1	3	1
die	died	difficult	difficulty	direction	disappearing	disclosed	discover	discovered
1	2	2	1	3	1	1	1	2
dismay	dismounting	distance	distinguish	dollars	dots	dreams	drew	drop
1	1	1	2	2	2	1	2	1
drowsiness	drowsy	drunkenly	dusk	duty	earlier	easily	edge	education
1	1	1	1	2	1	1	1	1
effort	emperor	endeavors	ended	endure	enemy	engineer	entered	entering
1	1	1	1	1	1	1	3	1

inspect(bookStopTDM) returns a Term Document Matrix (TDM) with 3768 terms and 11 documents. The non-sparse entries suggest a total of 7405 word occurrences, however the sparsity of 82% indicates that the majority of the entries in the matrix are zero.

The maximum term length is 17, implying that at least one term in the matrix has 17 characters. The term frequency (tf) weighting approach is utilized, which distributes weights based on the frequency of occurrence of each term in each document.

The sample depicts the frequency of occurrence of ten terms in each of the eleven papers. For example, the word "feet" appears 5 times in book_ch1.txt, 5 times in book_ch10.txt, 0 times in book_ch11.txt, and 1

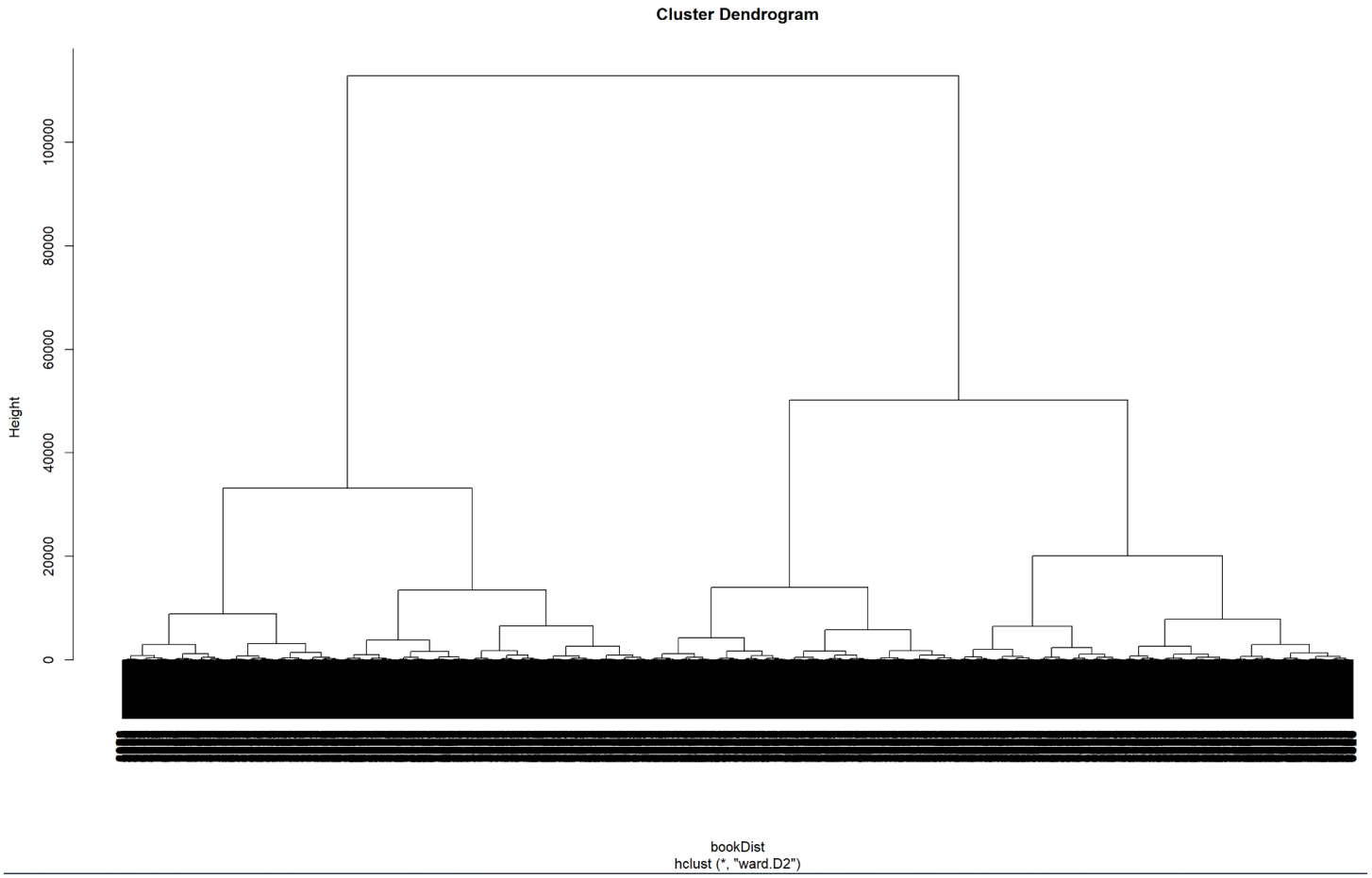
time in book_ch2.txt. Similarly, the term "upon" appears 12 times in book_ch8.txt, 19 times in book_ch9.txt, and 13 times in book_ch10.txt.

```
> inspect(bookStopTDM)
<<TermDocumentMatrix (terms: 3768, documents: 11)>>
Non-/sparse entries: 7405/34043
Sparsity           : 82%
Maximal term length: 17
Weighting          : term frequency (tf)
Sample             :
         Docs
Terms     book_ch1.txt book_ch10.txt book_ch11.txt book_ch2.txt book_ch3.txt book_ch4.txt book_ch5.txt book_ch6.txt book_ch7.txt
  feet               5             5             0            1           15           9           2           4           2
  first              3             6             3            5            5           6           1           3           3
  little             4             4             4            2           11           3           2           0          10
  mars               1             4             2            1           11           6           9           2           4
  martian            0            13             3            0            6           9           5           2          10
  martians           0             1             4            0            5           8           1           7          11
  one                3            19             1            2            7           5           6           5           9
  sola               0             6            11            0            0           2           5           4           7
  toward             3             9             1            3           10           6           3           3           2
  upon              12            19            13           13           20          11           8          13           7
         Docs
Terms     book_ch8.txt
  feet               1
  first              5
  little             4
  mars               1
  martian            7
  martians           3
  one                8
  sola               3
  toward             5
  upon              22
```

Dendrogram

The dendrogram output indicates that the hierarchical clustering was done using the "ward.D2" approach, which minimizes the variance between groups at each level. The dendrogram contains information on the clusters' merging order, the heights of the nodes in the tree, the order of the documents in the final cluster, the clustering method, the call used to create the dendrogram, and the distance metric used (in this case, the euclidean distance). Because the labels were not supplied in the code, they are not displayed.

Plot:



Word Cloud

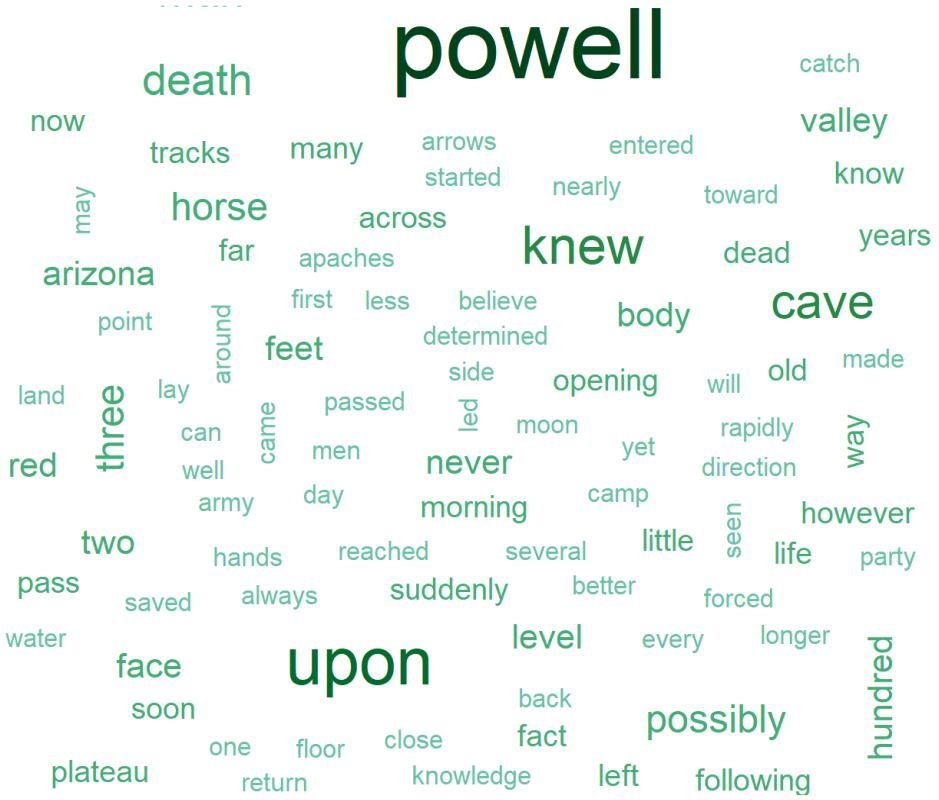
The code below generates a wordcloud based on the word frequency in the book chapters. The wordcloud package is loaded, and the names function is used to extract the words from the booktf object (which was previously constructed as a term frequency matrix).
Resulting words vector comprises all of the book's unique terms along with their frequency. This vector is then fed into the wordcloud function, which generates a graphical representation of the most frequently occurring terms in the text. Each word's size in the cloud is proportional to its frequency in the text.

```
> # word cloud
> install.packages("wordcloud")
Installing package into 'C:/Users/The Marry/AppData/Local/R/win-library/4.2'
(as 'lib' is unspecified)
trying URL 'https://cran.rstudio.com/bin/windows/contrib/4.2/wordcloud_2.6.zip'
Content type 'application/zip' length 436970 bytes (426 KB)
downloaded 426 KB

package 'wordcloud' successfully unpacked and MD5 sums checked

The downloaded binary packages are in
        C:\Users\The Marry\AppData\Local\Temp\RtmpGKvz3H\downloaded_packages
> library(wordcloud)
Loading required package: RColorBrewer
Warning message:
package 'wordcloud' was built under R version 4.2.3
> words <- names(booktf)
> words
  [1] "able"           "account"        "accurately"     "across"         "acted"          "acts"           "advent"
  [8] "adventures"     "afternoon"      "aged"           "ago"            "agreed"         "ahead"          "aid"
 [15] "aim"            "alive"          "almost"         "alone"          "already"        "alternative"    "always"
 [22] "among"          "animals"        "another"        "antelope"       "anything"       "apaches"        "apartments"
 [29] "appear"         "apprehension"   "arizona"        "arm"            "armed"          "arming"         "army"
 [36] "arose"          "around"         "arrows"         "assure"         "attacked"       "attempt"        "attention"
 [43] "attraction"     "attributed"     "average"        "await"          "back"           "backward"       "balls"
 [50] "bathed"         "beast"          "beautiful"      "became"         "become"         "befell"         "believe"
 [57] "belt"           "belts"          "best"           "bestowed"       "better"         "bidding"        "body"
 [64] "borne"          "bottom"         "bows"           "braves"         "brief"          "bright"         "brisk"
 [71] "bristling"      "broad"          "broke"          "brought"        "burros"         "came"           "camp"
 [78] "can"            "canteen"        "canter"         "captain"        "captains"       "capture"        "carbine"
 [85] "careless"       "carry"          "carter"         "cartridge"      "cartridges"     "casually"       "catch"
 [92] "catching"       "cavalry"        "cave"           "center"         "certain"        "chamber"        "chances"
 [99] "characteristic" "charging"       "chase"          "childhood"      "chronicle"      "civil"          "civilization"
[106] "claim"          "clear"          "cliff"          "close"          "clustered"      "clutches"       "colossal"
[113] "colt"           "come"           "commenced"      "commission"     "company"        "comparatively"  "confederate"
[120] "conjured"       "consternation"  "constituted"    "constitutes"    "continue"       "continued"      "continuing"
[127] "continuously"   "conviction"     "convince"       "convinced"      "country"        "course"         "cowardice"
[134] "creeping"       "crept"          "crude"          "cunning"        "customary"      "cut"            "dangerous"
[141] "dangers"        "dark"           "darkness"       "dash"           "dawn"           "day"            "daylight"
[148] "dead"           "deadly"         "death"          "debouched"      "decided"        "decorations"    "deepest"
```

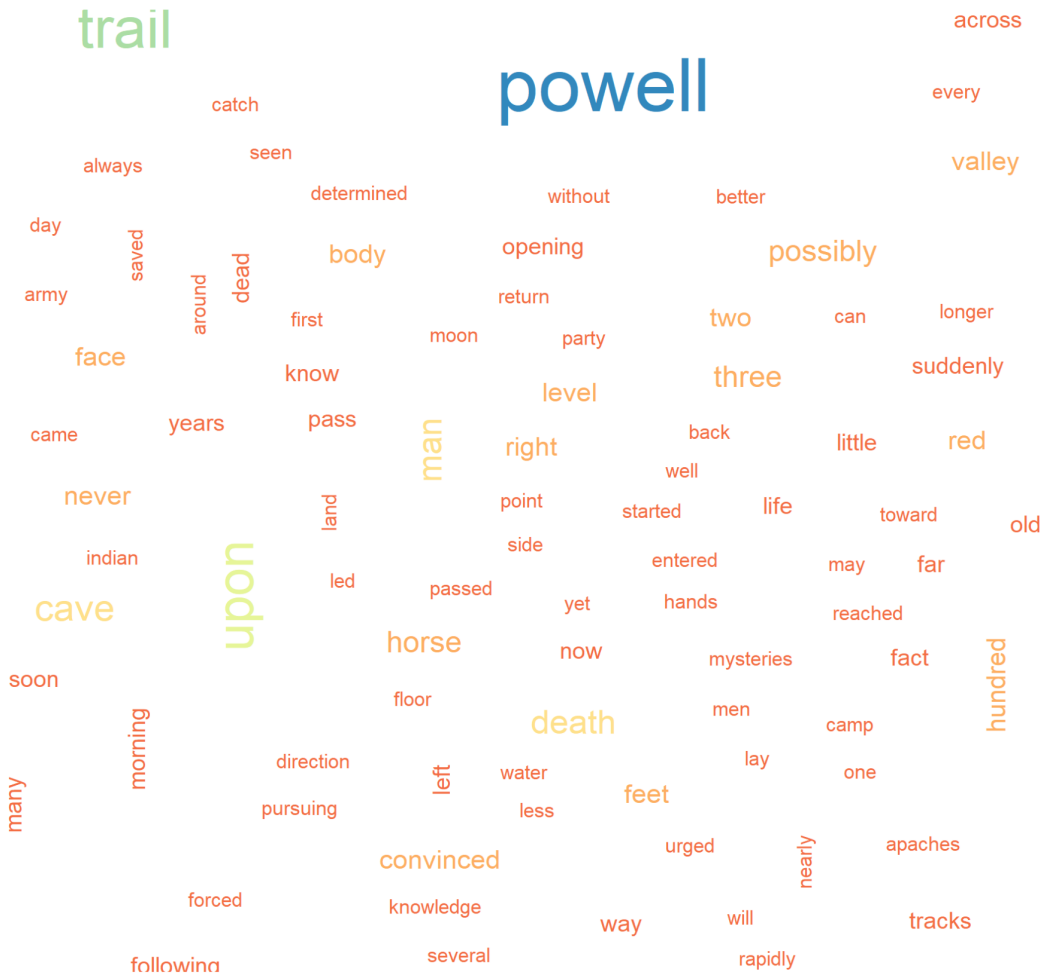
Using the brewer, the code below generates a color palette friend with 9 colors.The RColorBrewer package's friend() method. The colors range from mild to dark green and blue.

Then, using their frequencies (booktf) as input data, the wordcloud() function from the wordcloud package is used to generate a word cloud of the words vector. The colors argument is set to pal[-(1:4)], which signifies that the palette's first four colors are ignored.

The resulting word cloud is saved in the booksWc object, which is of the NULL type, suggesting that the function was used for its side effect of making a plot rather than its return value. The plot displays the words from the words vector with font size proportional to frequency and colors from the pal palette.



The "BuGn" argument defines the color scheme to utilize, while the "9" argument indicates the number of colors to include in the palette. The str function is then used to print information about the pal object, revealing that it is a character vector with nine hex color codes.



Quanteda

is a R program for quantitative textual data analysis. It includes text preparation, exploratory analysis, and statistical modeling tools. Quanteda is intended to be fast, efficient, and scalable when dealing with huge text datasets. Corpus management, tokenization, stemming, stopword removal, ngrams, collocations, sentiment analysis, topic modeling, and text classification are among its features. Quanteda also allows for parallel computing and integration with other R programs like tidyverse, ggplot2, and tm. Quanteda is frequently utilized in the social sciences, computational linguistics, digital humanities, and industry.

We have loaded the quanteda package and assigned the first book in your corpus (bookcl) to the variable bookText. Then printed out the first 10 lines of the content of the book using the bookText$content command.

```
> library(quanteda)
Package version: 3.3.0
Unicode version: 13.0
ICU version: 69.1
Parallel computing: 8 of 8 threads used.
See https://quanteda.io for tutorials and examples.

Attaching package: 'quanteda'

The following object is masked from 'package:tm':

    stopwords

The following objects are masked from 'package:NLP':

    meta, meta<-

Warning message:
package 'quanteda' was built under R version 4.2.3
> bookText <- bookcl[[1]]
> bookText$content[1:10]
 [1] ""
 [2] "ON THE ARIZONA HILLS"
 [3] ""
 [4] ""
 [5] "I am a very old man how old I do not know  Possibly I am a hundred"
 [6] "possibly more but I cannot tell because I have never aged as other"
 [7] "men nor do I remember any childhood  So far as I can recollect I have"
 [8] "always been a man a man of about thirty  I appear today as I did"
 [9] "forty years and more ago and yet I feel that I cannot go on living"
[10] "forever that some day I shall die the real death from which there is"
```

bookTokens is a list of tokens produced by applying the quanteda::tokens() function to the first ten documents in the bookText object. The tokens() function tokenizes text by breaking it down into smaller pieces, such as words or n-grams, and provides a common framework for encoding text data that can then be processed and analyzed.

The result of str(bookTokens) indicates that bookTokens is a list of ten elements, each of which corresponds to a document in the bookText object. Each element is a vector of tokens that represent the relevant document's individual words and punctuation marks. The attr(*, "types") attribute of the tokens object shows the unique tokens in the text, and the attr(*, "docvars") attribute stores the metadata associated with each document, such as its name and ID.

```
> bookTokens <- quanteda::tokens(bookText$content[1:10])
> str(bookTokens)
List of 10
 $ text1 : chr(0)
 $ text2 : chr [1:4] "ON" "THE" "ARIZONA" "HILLS"
 $ text3 : chr(0)
 $ text4 : chr(0)
 $ text5 : chr [1:17] "I" "am" "a" "very" ...
 $ text6 : chr [1:13] "possibly" "more" "but" "I" ...
 $ text7 : chr [1:15] "men" "nor" "do" "I" ...
 $ text8 : chr [1:15] "always" "been" "a" "man" ...
 $ text9 : chr [1:15] "forty" "years" "and" "more" ...
 $ text10: chr [1:14] "forever" "that" "some" "day" ...
 - attr(*, "types")= chr [1:66] "ON" "THE" "ARIZONA" "HILLS" ...
 - attr(*, "padding")= logi FALSE
 - attr(*, "class")= chr "tokens"
 - attr(*, "docvars")='data.frame':     10 obs. of  3 variables:
  ..$ docname_: chr [1:10] "text1" "text2" "text3" "text4" ...
  ..$ docid_  : Factor w/ 10 levels "text1","text2",..: 1 2 3 4 5 6 7 8 9 10
  ..$ segid_  : int [1:10] 1 1 1 1 1 1 1 1 1 1
 - attr(*, "meta")=List of 3
  ..$ system:List of 5
  .. ..$ package-version:Classes 'package_version', 'numeric_version'  hidden list of 1
  .. .. ..$ : int [1:3] 3 3 0
  .. ..$ r-version      :Classes 'R_system_version', 'package_version', 'numeric_version'  hidden list of 1
  .. .. ..$ : int [1:3] 4 2 2
  .. ..$ system         : Named chr [1:3] "Windows" "x86-64" "The Marry"
  .. .. ..- attr(*, "names")= chr [1:3] "sysname" "machine" "user"
  .. ..$ directory      : chr "C:/Users/The Marry/Documents/GWU/Spring 2023/Big Data/Project 3"
  .. ..$ created        : Date[1:1], format: "2023-05-06"
  ..$ object:List of 6
  .. ..$ unit        : chr "documents"
  .. ..$ what        : chr "word"
  .. ..$ ngram       : int 1
  .. ..$ skip        : int 0
  .. ..$ concatenator: chr "_"
  .. ..$ summary     :List of 2
  .. .. ..$ hash: chr(0)
  .. .. ..$ data: NULL
  ..$ user  : list()
```

Using the quanteda package in R, the code below generates a document-feature matrix (dfm). The bookTokens object most likely contains a corpus of text data that has been tokenized.

The dfm() function transforms tokenized data into a matrix with rows representing documents and columns representing characteristics (for example, words or n-grams). The matrix values represent the frequency of each feature in each document.

The resulting dfm object contains numerous slots containing information about the matrix, such as document names, feature names, and matrix values. The str() function is used to display the dfm object's structure and slots.

```
> booksDFM <- quanteda::dfm(bookTokens)
> str(booksDFM)
Formal class 'dfm' [package "quanteda"] with 8 slots
  ..@ docvars :'data.frame':    10 obs. of  3 variables:
  .. ..$ docname_: chr [1:10] "text1" "text2" "text3" "text4" ...
  .. ..$ docid_  : Factor w/ 10 levels "text1","text2",..: 1 2 3 4 5 6 7 8 9 10
  .. ..$ segid_  : int [1:10] 1 1 1 1 1 1 1 1 1 1
  ..@ meta    :List of 3
  .. ..$ system:List of 5
  .. .. ..$ package-version:Classes 'package_version', 'numeric_version'  hidden list of 1
  .. .. .. ..$ : int [1:3] 3 3 0
  .. .. ..$ r-version      :Classes 'R_system_version', 'package_version', 'numeric_version'  hidden list of 1
  .. .. .. ..$ : int [1:3] 4 2 2
  .. .. ..$ system         : Named chr [1:3] "Windows" "x86-64" "The Marry"
  .. .. .. ..- attr(*, "names")= chr [1:3] "sysname" "machine" "user"
  .. .. ..$ directory      : chr "C:/Users/The Marry/Documents/GWU/Spring 2023/Big Data/Project 3"
  .. .. ..$ created        : Date[1:1], format: "2023-05-06"
  .. ..$ object:List of 9
  .. .. ..$ unit        : chr "documents"
  .. .. ..$ what        : chr "word"
  .. .. ..$ ngram       : int 1
  .. .. ..$ skip        : int 0
  .. .. ..$ concatenator: chr "_"
  .. .. ..$ weight_tf   :List of 3
  .. .. .. ..$ scheme: chr "count"
  .. .. .. ..$ base  : NULL
  .. .. .. ..$ k     : NULL
  .. .. ..$ weight_df   :List of 5
  .. .. .. ..$ scheme   : chr "unary"
  .. .. .. ..$ base     : NULL
  .. .. .. ..$ c        : NULL
  .. .. .. ..$ smoothing: NULL
  .. .. .. ..$ threshold: NULL
  .. .. ..$ smooth      : num 0
  .. .. ..$ summary     :List of 2
  .. .. .. ..$ hash: chr(0)
  .. .. .. ..$ data: NULL
  .. ..$ user  : list()
  ..@ i       : int [1:80] 1 8 1 9 1 1 4 5 6 7 ...
  ..@ p       : int [1:64] 0 2 4 5 6 12 13 15 16 17 ...
  ..@ Dim     : int [1:2] 10 63
  ..@ Dimnames:List of 2
  .. ..$ docs    : chr [1:10] "text1" "text2" "text3" "text4" ...
  .. ..$ features: chr [1:63] "on" "the" "arizona" "hills" ...
  ..@ x       : num [1:80] 1 1 1 1 1 1 3 2 3 2 ...
  ..@ factors : list()
```

The document frequency of each unique term in the booksDFM object, which is a document-feature matrix produced in R using the quanteda package, is stored in the bookDocFreq variable. The number of documents in which a term appears at least once is known as document frequency.

str(bookDocFreq) returns that bookDocFreq is a named integer vector of length 63, with each element corresponding to a distinct term in the booksDFM object. The names of the elements, which are the terms themselves, are contained in the attr(*, "names") attribute.

The second section of the output displays the real bookDocFreq values, where each value reflects the document frequency of the relevant phrase. For example, the term "on" appears in two texts, as does the term "the" in two documents and so on.

```
> bookDocFreq <- quanteda::docfreq(booksDFM)
> str(bookDocFreq)
 Named int [1:63] 2 2 1 1 6 1 2 1 1 2 ...
 - attr(*, "names")= chr [1:63] "on" "the" "arizona" "hills" ...
> bookDocFreq
       on       the   arizona     hills         i        am         a      very       old       man       how        do       not      know
        2         2         1         1         6         1         2         1         1         2         1         2         1         1
 possibly   hundred      more       but    cannot      tell   because      have     never      aged        as     other       men       nor
        2         1         2         1         2         1         1         2         1         1         3         1         1         1
 remember       any childhood        so       far       can recollect    always      been        of     about    thirty    appear     today
        1         1         1         1         1         1         1         1         1         1         1         1         1         1
      did     forty     years       and       ago       yet      feel      that        go    living   forever      some       day     shall
        1         1         1         1         1         1         1         2         1         1         1         1         1         1
      die      real     death      from     which     there        is
        1         1         1         1         1         1         1
```

The output displays the document-feature matrix bookWeights after applying a weighting scheme to booksDFM with the quanteda package's dfm_weight() function.

The bookWeights matrix comprises ten rows for the ten documents and 63 columns for the 63 features (words) in the documents. The 63 features are listed alphabetically in the features column.

After applying the weighting technique, the values in the matrix represent the frequency of the related feature in each document. Because no weighting method is supplied in this scenario, the default count scheme is applied, therefore the values in the matrix are the raw frequency counts of the features in each document.

```
> bookWeights <- quanteda::dfm_weight(booksDFM)
> str(bookWeights)
Formal class 'dfm' [package "quanteda"] with 8 slots
  ..@ docvars :'data.frame':    10 obs. of  3 variables:
  .. ..$ docname_: chr [1:10] "text1" "text2" "text3" "text4" ...
  .. ..$ docid_  : Factor w/ 10 levels "text1","text2",..: 1 2 3 4 5 6 7 8 9 10
  .. ..$ segid_  : int [1:10] 1 1 1 1 1 1 1 1 1 1
  ..@ meta    :List of 3
  .. ..$ system:List of 5
  .. .. ..$ package-version:Classes 'package_version', 'numeric_version'  hidden list of 1
  .. .. .. ..$ : int [1:3] 3 3 0
  .. .. ..$ r-version      :Classes 'R_system_version', 'package_version', 'numeric_version'  hidden l
  .. .. .. ..$ : int [1:3] 4 2 2
  .. .. ..$ system         : Named chr [1:3] "Windows" "x86-64" "The Marry"
  .. .. .. ..- attr(*, "names")= chr [1:3] "sysname" "machine" "user"
  .. .. ..$ directory      : chr "C:/Users/The Marry/Documents/GWU/Spring 2023/Big Data/Project 3"
  .. .. ..$ created        : Date[1:1], format: "2023-05-06"
  .. ..$ object:List of 9
  .. .. ..$ unit        : chr "documents"
  .. .. ..$ what        : chr "word"
  .. .. ..$ ngram       : int 1
  .. .. ..$ skip        : int 0
  .. .. ..$ concatenator: chr "_"
  .. .. ..$ weight_tf   :List of 3
  .. .. .. ..$ scheme: chr "count"
  .. .. .. ..$ base  : NULL
  .. .. .. ..$ k     : NULL
  .. .. ..$ weight_df   :List of 5
  .. .. .. ..$ scheme   : chr "unary"
  .. .. .. ..$ base     : NULL
  .. .. .. ..$ c        : NULL
  .. .. .. ..$ smoothing: NULL
  .. .. .. ..$ threshold: NULL
  .. .. ..$ smooth      : num 0
  .. .. ..$ summary     :List of 2
  .. .. .. ..$ hash: chr(0)
  .. .. .. ..$ data: NULL
  .. ..$ user  : list()
  ..@ i       : int [1:80] 1 8 1 9 1 1 4 5 6 7 ...
  ..@ p       : int [1:64] 0 2 4 5 6 12 13 15 16 17 ...
  ..@ Dim     : int [1:2] 10 63
  ..@ Dimnames:List of 2
  .. ..$ docs    : chr [1:10] "text1" "text2" "text3" "text4" ...
```

```
> bookWeights
Document-feature matrix of: 10 documents, 63 features (87.30% sparse) and 0 docvars.
       features
docs    on the arizona hills i am a very old man
  text1  0   0       0     0 0  0 0    0   0   0
  text2  1   1       1     1 0  0 0    0   0   0
  text3  0   0       0     0 0  0 0    0   0   0
  text4  0   0       0     0 0  0 0    0   0   0
  text5  0   0       0     0 3  2 2    1   2   1
  text6  0   0       0     0 2  0 0    0   0   0
[ reached max_ndoc ... 4 more documents, reached max_nfeat ... 53 more features ]
```

The dfm object has 8 slots:

docvars: a data frame with 3 variables: docname_ containing the document names, docid_ containing a factor with the document IDs, and segid_ containing an integer with the segment IDs.

meta: a list with metadata about the dfm object. It contains information about the system, the object, and the user.

i: an integer vector indicating the non-zero elements in the dfm object.

p: an integer vector indicating the start of each column in the i and x vectors.

Dim: an integer vector indicating the dimensions of the dfm object.

Dimnames: a list with two elements: docs containing the document names and features containing the feature (word) names.

x: a named numeric vector containing the values of the non-zero elements in the dfm object.

factors: an empty list.

The bookingTFIDF object is a tf-idf weighted dfm object created from the booksDFM object using the quanteda::dfm_tfidf() function. The scheme_tf argument is set to "count" to count the number of times a

term appears in a document, and the scheme_df argument is set to "inverse" to apply inverse document frequency weighting. The resulting tf-idf weighted dfm object contains 10 documents and 63 features.

```
> bookingTFIDF <- quanteda::dfm_tfidf(booksDFM, scheme_tf = "count", scheme_df = "inverse")
> str(bookingTFIDF)
Formal class 'dfm' [package "quanteda"] with 8 slots
  ..@ docvars :'data.frame':    10 obs. of  3 variables:
  .. ..$ docname_: chr [1:10] "text1" "text2" "text3" "text4" ...
  .. ..$ docid_  : Factor w/ 10 levels "text1","text2",..: 1 2 3 4 5 6 7 8 9 10
  .. ..$ segid_  : int [1:10] 1 1 1 1 1 1 1 1 1 1
  ..@ meta    :List of 3
  .. ..$ system:List of 5
  .. .. ..$ package-version:Classes 'package_version', 'numeric_version'  hidden list of 1
  .. .. .. ..$ : int [1:3] 3 3 0
  .. .. ..$ r-version      :Classes 'R_system_version', 'package_version', 'numeric_version'  hidden list of 1
  .. .. .. ..$ : int [1:3] 4 2 2
  .. .. ..$ system         : Named chr [1:3] "Windows" "x86-64" "The Marry"
  .. .. .. ..- attr(*, "names")= chr [1:3] "sysname" "machine" "user"
  .. .. ..$ directory      : chr "C:/Users/The Marry/Documents/GWU/Spring 2023/Big Data/Project 3"
  .. .. ..$ created        : Date[1:1], format: "2023-05-06"
  .. ..$ object:List of 9
  .. .. ..$ unit        : chr "documents"
  .. .. ..$ what        : chr "word"
  .. .. ..$ ngram       : int 1
  .. .. ..$ skip        : int 0
  .. .. ..$ concatenator: chr "_"
  .. .. ..$ weight_tf   :List of 3
  .. .. .. ..$ scheme: chr "count"
  .. .. .. ..$ base  : NULL
  .. .. .. ..$ k     : NULL
  .. .. ..$ weight_df   :List of 2
  .. .. .. ..$ scheme: chr "inverse"
  .. .. .. ..$ base  : num 10
  .. .. ..$ smooth      : num 0
  .. .. ..$ summary     :List of 2
  .. .. .. ..$ hash: chr(0)
```

syuzhet package

contains methods for extracting sentiment and emotional information from text. It calculates the emotional "valence" or polarity of individual words in a given text using a collection of pre-defined sentiment dictionaries, and then aggregates these values to provide an overall score for the entire text. The software can be used to assess the emotional content of vast amounts of text, such as books, articles, or social media messages.

The get_text_as_string function is not part of the syuzhet package, but is likely a custom function defined by the user to read in the text of a book from a file and convert it to a character string. This string can then be passed to the get_sentiment function from the syuzhet package to analyze its emotional content.

```
> install.packages("syuzhet")
Installing package into 'C:/Users/The Marry/AppData/Local/R/win-library/4.2'
(as 'lib' is unspecified)
trying URL 'https://cran.rstudio.com/bin/windows/contrib/4.2/syuzhet_1.0.6.zip'
Content type 'application/zip' length 3107383 bytes (3.0 MB)
downloaded 3.0 MB

package 'syuzhet' successfully unpacked and MD5 sums checked

The downloaded binary packages are in
        C:\Users\The Marry\AppData\Local\Temp\RtmpGKvz3H\downloaded_packages
> library(syuzhet)
Warning message:
package 'syuzhet' was built under R version 4.2.3
> bookTextDf <- as.data.frame(bookText$content)
> bookTextDf
                                                                bookText$content
1
2                                                                ON THE ARIZONA HILLS
3
4
5          I am a very old man how old I do not know  Possibly I am a hundred
6          possibly more but I cannot tell because I have never aged as other
7       men nor do I remember any childhood  So far as I can recollect I have
8            always been a man a man of about thirty  I appear today as I did
9          forty years and more ago and yet I feel that I cannot go on living
10      forever that some day I shall die the real death from which there is
11         no resurrection  I do not know why I should fear death I who have
12      died twice and am still alive but yet I have the same horror of it as
13        you who have never died and it is because of this terror of death I
14                              believe that I am so convinced of my mortality
15
16         And because of this conviction I have determined to write down the
17      story of the interesting periods of my life and of my death  I cannot
18          explain the phenomena I can only set down here in the words of an
```

```
> bookAsStings <- get_text_as_string("APrincessOfMars.txt")
> bookAsStings
 CHAPTER I  ON THE ARIZONA HILLS   I am a very old man; how old I do not know.  Possibly I am a hundred, possibly more; but I cannot tell because
I have never aged as other men, nor do I remember any childhood.  So far as I can recollect I have always been a man, a man of about thirty.  I a
ppear today as I did forty years and more ago, and yet I feel that I cannot go on living forever; that some day I shall die the real death from w
hich there is no resurrection.  I do not know why I should fear death, I who have died twice and am still alive; but yet I have the same horror o
f it as you who have never died, and it is because of this terror of death, I believe, that I am so convinced of my mortality.  And because of th
is conviction I have determined to write down the story of the interesting periods of my life and of my death.  I cannot explain the phenomena; I
can only set down here in the words of an ordinary soldier of fortune a chronicle of the strange events that befell me during the ten years that
my dead body lay undiscovered in an Arizona cave.  I have never told this story, nor shall mortal man see this manuscript until after I have pass
ed over for eternity.  I know that the average human mind will not believe what it cannot grasp, and so I do not purpose being pilloried by the p
ublic, the pulpit, and the press, and held up as a colossal liar when I am but telling the simple truths which some day science will substantiat
e.  Possibly the suggestions which I gained upon Mars, and the knowledge which I can set down in this chronicle, will aid in an earlier understan
ding of the mysteries of our sister planet; mysteries to you, but no longer mysteries to me.  My name is John Carter; I am better known as Captai
n Jack Carter of Virginia.  At the close of the Civil War I found myself possessed of several hundred thousand dollars (Confederate) and a captai
n's commission in the cavalry arm of an army which no longer existed; the servant of a state which had vanished with the hopes of the South. Mast
erless, penniless, and with my only means of livelihood, fighting, gone, I determined to work my way to the southwest and attempt to retrieve my
fallen fortunes in a search for gold.  I spent nearly a year prospecting in company with another Confederate officer, Captain James K. Powell of
Richmond.  We were extremely fortunate, for late in the winter of 1865, after many hardships and privations, we located the most remarkable gold-
bearing quartz vein that our wildest dreams had ever pictured.  Powell, who was a mining engineer by education, stated that we had uncovered over
a million dollars worth of ore in a trifle over three months.  As our equipment was crude in the extreme we decided that one of us must return to
civilization, purchase the necessary machinery and return with a sufficient force of men properly to work the mine.  As Powell was familiar with
the country, as well as with the mechanical requirements of mining we determined that it would be best for him to make the trip.  It was agreed t
hat I was to hold down our claim against the remote possibility of its being jumped by some wandering prospector.  On March 3, 1866, Powell and I
packed his provisions on two of our burros, and bidding me good-bye he mounted his horse, and started down the mountainside toward the valley, ac
ross which led the first stage of his journey.  The morning of Powell's departure was, like nearly all Arizona mornings, clear and beautiful; I c
ould see him and his little pack animals picking their way down the mountainside toward the valley, and all during the morning I would catch occa
sional glimpses of them as they topped a hog back or came out upon a level plateau.  My last sight of Powell was about three in the afternoon as
he entered the shadows of the range on the opposite side of the valley.  Some half hour later I happened to glance casually across the valley and
was much surprised to note three little dots in about the same place I had last seen my friend and his two pack animals.  I am not given to needl
ess worrying, but the more I tried to convince myself that all was well with Powell, and that the dots I had seen on his trail were antelope or w
ild horses, the less I was able to assure myself.  Since we had entered the territory we had not seen a hostile Indian, and we had, therefore, be
come careless in the extreme, and were wont to ridicule the stories we had heard of the great numbers of these vicious marauders that were suppos
```

The syuzhet package's get_sentences() function takes a text document as input and extracts all of the sentences contained within it. The function detects the conclusion of a phrase using natural language processing techniques based on the presence of punctuation symbols such as periods, question marks, and exclamation marks.

The **get_sentences()** function is applied to the bookAsStings object, which includes the text of a book in string format, in this example. The function's result is assigned to the bookSentences object, which is a vector of character strings, each member of which represents a sentence from the book.

```
 [981] "I pondered over this report for some time, finally asking, \"What might a sorak be, Sola?\""
 [982] "\"A little animal about as big as my hand, which the red Martian women keep to play with,\" explained Sola."
 [983] "Not fit to polish the teeth of her grandmother's cat!"
 [984] "I must rank pretty low in the consideration of Dejah Thoris, I thought; but I could not help laughing at the strange figure of speech,
homely and in this respect so earthly."
 [985] "It made me homesick, for it sounded very much like \"not fit to polish her shoes.\""
 [986] "And then commenced a train of thought quite new to me."
 [987] "I began to wonder what my people at home were doing."
 [988] "I had not seen them for years."
 [989] "There was a family of Carters in Virginia who claimed close relationship with me; I was supposed to be a great uncle, or something of t
kind equally foolish."
 [990] "I could pass anywhere for twenty-five to thirty years of age, and to be a great uncle always seemed the height of incongruity, for my t
ughts and feelings were those of a boy."
 [991] "There were two little kiddies in the Carter family whom I had loved and who had thought there was no one on Earth like Uncle Jack; I co
d see them just as plainly, as I stood there under the moonlit skies of Barsoom, and I longed for them as I had never longed for any mortals be
re."
 [992] "By nature a wanderer, I had never known the true meaning of the word home, but the great hall of the Carters had always stood for all t
t the word did mean to me, and now my heart turned toward it from the cold and unfriendly peoples I had been thrown amongst."
 [993] "For did not even Dejah Thoris despise me!"
 [994] "I was a low creature, so low in fact that I was not even fit to polish the teeth of her grandmother's cat; and then my saving sense of
mor came to my rescue, and laughing I turned into my silks and furs and slept upon the moon-haunted ground the sleep of a tired and healthy fig
ing man."
 [995] "We broke camp the next day at an early hour and marched with only a single halt until just before dark."
 [996] "Two incidents broke the tediousness of the march."
 [997] "About noon we espied far to our right what was evidently an incubator, and Lorquas Ptomel directed Tars Tarkas to investigate it."
 [998] "The latter took a dozen warriors, including myself, and we raced across the velvety carpeting of moss to the little enclosure."
 [999] "It was indeed an incubator, but the eggs were very small in comparison with those I had seen hatching in ours at the time of my arrival
n Mars."
[1000] "Tars Tarkas dismounted and examined the enclosure minutely, finally announcing that it belonged to the green men of Warhoon and that th
cement was scarcely dry where it had been walled up."
 [ reached getOption("max.print") -- omitted 1321 entries ]
> str(bookSentences)
 chr [1:2321] "CHAPTER I  ON THE ARIZONA HILLS   I am a very old man; how old I do not know." ...
```

The syuzhet package's get_sentiment() function is used to compute the sentiment of each sentence in a text. It is being applied to the bookSentences object in this case, which includes a vector of sentences extracted from the book "A Princess of Mars" that was earlier saved in bookAsStings. The function accepts two arguments: the vector of sentences to be analyzed and the sentiment analysis technique to be applied. The default syuzhet algorithm is used here.

get_sentiment() returns a numeric vector containing the sentiment values for each sentence in bookSentences. These ratings range from -1 (most negative) to 1, with 0 representing neutral emotion.

```
> bookSentiment <- get_sentiment(bookSentences, "syuzhet")
> bookSentiment
   [1]  0.000000e+00  6.000000e-01  0.000000e+00 -1.000000e+00 -3.400000e+00  0.000000e+00 -9.000000e-01 -2.500000e-01 -3.000000e-01
  [10]  3.200000e+00  8.000000e-01 -1.000000e-01 -1.050000e+00 -6.500000e-01  9.500000e-01  6.500000e-01  2.000000e+00  2.600000e+00
  [19]  5.000000e-01  8.000000e-01  1.400000e+00  0.000000e+00  8.000000e-01  5.000000e-01 -4.950000e+00  1.650000e+00  6.500000e-01
  [28]  1.000000e+00  0.000000e+00 -7.500000e-01 -2.500000e-01  2.850000e+00  1.500000e+00 -8.500000e-01  7.000000e-01 -1.500000e+00
  [37] -5.000000e-01 -7.500000e-01  8.000000e-01 -1.500000e+00  1.050000e+00  1.200000e+00  1.800000e+00  5.000000e-02 -1.000000e+00
  [46] -1.250000e+00  2.500000e-01  7.000000e-01 -2.000000e+00  2.500000e-01 -1.350000e+00 -6.500000e-01 -1.250000e+00 -1.250000e+00
  [55] -2.500000e-01  1.300000e+00  1.000000e+00 -3.500000e-01  2.000000e-01 -6.000000e-01  3.000000e-01  1.000000e+00  1.200000e+00
  [64]  8.000000e-01  8.000000e-01  0.000000e+00 -1.000000e+00 -4.000000e-01 -4.500000e-01  3.700000e+00 -6.000000e-01  1.200000e+00
  [73] -7.500000e-01 -9.500000e-01  1.250000e+00  5.000000e-02 -8.500000e-01  6.000000e-01 -1.500000e+00 -5.000000e-01  0.000000e+00
  [82] -4.000000e-01 -9.000000e-01  4.000000e-01 -2.000000e-01  1.550000e+00 -1.150000e+00 -1.500000e-01  0.000000e+00 -5.000000e-01
  [91]  6.500000e-01  0.000000e+00 -2.050000e+00 -1.600000e+00 -1.250000e+00 -5.000000e-01 -2.750000e+00 -4.700000e+00 -1.050000e+00
 [100] -5.000000e-01 -1.050000e+00 -4.900000e+00 -2.750000e+00  9.500000e-01 -1.250000e+00 -1.550000e+00 -6.500000e-01 -1.250000e+00
 [109] -7.500000e-01  0.000000e+00  4.000000e-01 -2.500000e-01 -1.000000e+00 -2.000000e-01 -1.500000e-01  2.500000e-01 -2.600000e+00
 [118] -1.500000e+00  2.150000e+00 -1.250000e+00  3.850000e+00  2.250000e+00  3.250000e+00 -1.250000e+00  1.700000e+00  1.100000e+00
 [127] -1.350000e+00 -4.500000e-01 -7.500000e-01 -1.250000e+00  5.000000e-02  0.000000e+00  4.000000e-01  1.000000e-01 -5.000000e-01
 [136]  1.450000e+00  0.000000e+00  8.500000e-01  1.800000e+00  5.000000e-02 -7.500000e-01  2.000000e+00  1.850000e+00 -1.600000e+00
 [145]  2.500000e-01  1.300000e+00  2.500000e-01  1.150000e+00  0.000000e+00 -1.750000e+00  0.000000e+00  1.300000e+00  4.000000e-01
 [154]  2.500000e-01  0.000000e+00  6.000000e-01  4.000000e-01 -6.000000e-01  4.000000e-01 -2.050000e+00  4.000000e-01 -4.500000e-01
 [163]  1.750000e+00 -1.500000e+00  4.500000e-01 -7.000000e-01  7.500000e-01 -1.200000e+00 -1.400000e+00 -1.550000e+00  4.000000e-01
 [172]  2.200000e+00  0.000000e+00  0.000000e+00  1.000000e-01  4.000000e-01  9.000000e-01  0.000000e+00 -6.000000e-01  8.500000e-01
 [181] -6.500000e-01 -4.000000e-01  1.000000e+00  1.850000e+00 -4.000000e-01  1.900000e+00  0.000000e+00  1.000000e+00  6.000000e-01
 [190]  1.950000e+00  2.850000e+00  1.250000e+00  2.000000e-01  4.500000e-01 -1.000000e-01  1.500000e+00 -1.500000e+00  1.450000e+00
 [199] -6.000000e-01 -6.000000e-01 -6.000000e-01  7.000000e-01 -1.400000e+00 -1.000000e-01 -4.000000e-01 -2.500000e-01  1.550000e+00
 [208]  1.750000e+00  5.000000e-01  1.550000e+00  7.500000e-01  8.000000e-01 -2.500000e-01 -5.000000e-01  0.000000e+00  9.000000e-01
 [217]  0.000000e+00 -7.500000e-01  5.500000e-01  0.000000e+00 -7.500000e-01 -1.100000e+00 -3.500000e-01  0.000000e+00  0.000000e+00
 [226] -1.100000e+00  0.000000e+00  1.200000e+00  6.000000e-01 -2.550000e+00 -2.750000e+00 -2.500000e-01  1.950000e+00  2.500000e-01
 [235] -6.000000e-01  0.000000e+00  1.000000e-01  1.900000e+00  1.000000e+00  1.100000e+00  2.500000e-01  1.300000e+00  1.750000e+00
 [244]  2.500000e-01 -2.000000e+00  3.500000e-01  0.000000e+00  2.500000e-01  2.500000e-01  0.000000e+00  8.500000e-01 -2.500000e-01
```

The sentiment analysis of the book using the Bing lexicon reveals that the majority of the sentences have a neutral or slightly positive sentiment, with only a few negative sentences. There are, however, some really unfavorable statements with ratings as low as -8.
It should be noted that sentiment analysis can be subjective and is dependent on the terminology employed. Furthermore, context is important in detecting a sentence's attitude, and the algorithm may not always capture the intricacies of language and tone. As a result, it is always prudent to interpret the results cautiously and in conjunction with a human comprehension of the text.

```
> bookSBing <- get_sentiment(bookSentences, "bing")
> bookSBing
   [1]  0  0  0 -2 -4  0 -2  0 -1  1  1  0  1  0  2  1  2  2  0  2  3  0  0 -1 -8  1  0  0  0  0 -1  2  3 -1  0 -2 -1  0 -1 -2 -1  0  1 -1 -1
  [46] -1  2  1 -2  1 -2  0 -3  0 -3  0  1 -2  0 -1  1  1  1  0  1  0 -2 -1  0  2 -2  0 -3 -1  0 -2 -1  1 -2 -1 -1  0 -3  0 -1  2 -1 -3  0 -1
  [91]  0  0 -4 -1 -1  1 -3 -8 -2 -1 -6 -5 -5  1 -2 -2 -3 -3 -1  0  1 -1 -3  0 -1  0 -5 -2  2 -2  3  2  3  0  2  0  2 -2 -2 -2 -2  0  0 -1 -1
 [136] -2  0  0  0  0 -1  0  4  0 -1  2  1  2  0 -1  0 -1  0  0  0 -1  0 -1  0 -1  0  1  1 -3 -1  0  0 -1  0 -2  0  3  0  0 -1  2  1  0  1 -1
 [181] -1 -1  1  0 -1  2  0 -1  1  1  3  1  0  0 -2  0 -2  2  1  0  0  0  0  0  0 -1  1  1  1  1  1  0 -1  0  0  0  0 -1 -1  1 -1 -1  0  0  0
 [226] -1  0  2  0 -2 -3  0  0 -1 -1  0  0  2  1  1  0  3  1  0 -4  0  0  0  1  0  0  0  0  0  1  2  1  2 -5  0  0 -1 -1 -3 -3 -6  1 -2  0  0
 [271]  0  2  1 -1  0  0  0  2  1  0  1  2 -2  1  1  0 -1  0  0  0  2  1 -1  0  1  0  1  0  0 -2 -1  0 -1  1 -1 -2  0  1  1 -1 -1  0 -1 -1  2
 [316]  0  4 -2  5 -1 -1 -1  0  0  1  1 -1  1 -1  0  0  1  1  1  2  0  0 -3  1 -1 -3  2 -2  1 -2 -3 -3 -2 -4 -6  0  1 -3  0  2  1 -1  1 -3  0
 [361] -2 -1 -1 -1 -1 -2  1  2  7 -1 -4  1 -1 -1 -1 -1  1 -1 -1  0 -3  2  0  2  0  1  1  0 -1  1  1  0 -2  1  0  0  1  0  0  0  0  0  0  2 -1
 [406] -1  0  1  1 -2  0  0 -2 -1 -3 -4 -1  0  1  1  0  1  0  0  0 -1  0  0  1  0  0 -1 -2  0  1  0  2  3  0  0  0  2 -1  1 -1 -1 -1 -2  1  1
 [451]  4 -2  1 -4  0  2  1  0  0 -1  0  0 -3 -1  0  0  0  0  0  0  0 -2 -2 -1 -1  1  0 -1  0 -2  0 -1 -2 -2 -2 -2 -2 -1 -2 -1  0  0 -1  3  1
 [496]  1  2 -1 -1 -1 -1  2 -1  0 -1  1  0  2 -1  0  0  0 -1  1 -1  2 -1 -1 -1  2  1  0 -1 -1 -5  0  0  0  0  0  0  1 -2  0 -1  0 -1 -2  0  2
 [541] -7  0 -1 -5  2 -1  0  0  4  1  3  1 -3  0 -1  0  1  1  2 -1  0  1  0  3  0  4 -1 -5  0  1  1  0  2  0  1  0 -1  0 -3 -1  3  0 -1  2 -1
 [586]  0 -2 -4  0 -7 -3 -3  1  0 -1  0  0  0  1  0  5 -3  1 -1  0  0 -1  1  0 -1  1  0  0  3  0  0 -3 -3  0 -3  0 -2 -1 -4  0 -2  0  0 -1  1
 [631]  1 -1 -2  0 -2  0 -3  1 -1  0  0  0 -1  1 -6 -1  2 -1 -1  0  2  1 -3 -2 -1  0  0  3  1  2  0  0  0  1 -3  0 -1  1  1  0  1  0  0 -1 -4
 [676] -3 -1  0  0  2  1  1 -1  0  0  0 -1 -1  0  0  2  4  0  1  2  0  0  0  0  1  0  0  1  0  0 -1  0 -1 -3 -4  0 -2  0  0  0  2  2  0  2  1
 [721]  1 -1  2 -1  0 -1  1 -1 -1  1  1  0 -3  1  1  1  0  0 -3 -2 -2 -1 -2  0  0  0  0  0  2  2 -2 -1  2  0  2  0  0  0  0  0  1  0 -5 -1  0
 [766]  0  1 -2  0  2  0  0  2  0  0 -1  0  1  2  0 -1 -4 -4 -2  1 -2  0  0  0  0  1  0  1  0 -1  2  2 -2  1  1 -1  1  0  0  0  8 -4 -1  0  0
 [811]  0 -2  0  2  0 -1  0  0  0  1 -3  2  1  0  0  1  0 -1  0 -1  0 -2 -1  1  1  1  4 -3  1  1  0  2  1  1  4  3  3  1  0 -1  1 -1  1  0 -1
 [856]  0  2  0  4  1  6  0  0  0  0  0 -1  0 -3  1  0  0  0  0 -3 -3 -3  1  0  0 -1 -5 -1  0  0  2 -1  1  0 -2  0 -1  3 -2  1  0  0 -1  0  1
 [901] -1  0  2 -1  1  0 -1  1 -1 -2 -1 -2  1  0  0  1  2  0  0  1  0  0  1  1  1  0 -1 -3  1 -1  0 -1  1  5  1  0  2  0 -2  1  0  0  0  0  0
 [946]  0  0  0 -3  0  0  2  0 -1 -1  0  0  0 -1  1  1  1  1 -2 -1 -1 -1  0 -1  0  0 -1  0  1  0  0  0  0 -3  0  0  0  0  1  1  0  1  0  0  1
 [991]  2 -1 -1  1 -2 -1  1  0  0 -1
 [ reached getOption("max.print") -- omitted 1321 entries ]
```

BSDictionary is a variable that stores a sentiment dictionary. A sentiment dictionary is a collection of words or phrases with associated sentiment scores that indicate the emotional polarity or valence of the word or phrase.

```
> BSDictionary <- get_sentiment_dictionary()
> BSDictionary
                word value
1           abandon -0.75
2         abandoned -0.50
3         abandoner -0.25
4       abandonment -0.25
5          abandons -1.00
6          abducted -1.00
7         abduction -0.50
8        abductions -1.00
9          aberrant -0.60
10       aberration -0.80
11            abhor -0.50
12         abhorred -1.00
13        abhorrent -0.50
14           abhors -1.00
15        abilities  0.60
16          ability  0.50
17           abject -1.00
18           ablaze -0.25
19         abnormal -0.50
20           aboard  0.25
21          abolish -0.50
22       abominable -0.50
23       abominably -1.00
24        abominate -1.00
25      abomination -0.50
26            abort -0.50
27          aborted -0.80
28         abortion -0.80
29         abortive -1.00
```

get_sentiment_dictionary("bing") function retrieves the Bing Liu lexicon, which is a sentiment lexicon containing 6780 English words and phrases classified as either positive or negative. Each word or phrase is assigned a sentiment score of either +1 (positive) or -1 (negative):

```
> BSDictionaryBing <- get_sentiment_dictionary("bing")
> BSDictionaryBing
                    word value
1                     a+     1
2                 abound     1
3                abounds     1
4              abundance     1
5               abundant     1
6             accessable     1
7             accessible     1
8                acclaim     1
9              acclaimed     1
10           acclamation     1
11              accolade     1
12             accolades     1
13         accommodative     1
14          accomodative     1
15            accomplish     1
16          accomplished     1
17        accomplishment     1
18       accomplishments     1
19               accurate     1
20             accurately     1
21             achievable     1
22            achievement     1
23           achievements     1
24             achievible     1
25                 acumen     1
```

bookSentiment is a numeric vector containing sentiment ratings for various parts of a book; the total sentiment score of the book is derived by adding all of the sentiment scores in this vector.

According to the output of BSSum, the total sentiment score of the book is 63.3. However, it is impossible to interpret this figure without knowing the scale of the emotion scores (e.g., are they on a scale of 0 to 1 or -1 to 1?).

```
> BSSum <- sum(bookSentiment)
> BSSum
[1] 63.3
>
```

The sum of the sentiment scores using the Bing lexicon is -393, which indicates a negative overall sentiment for the book.

```
> BSBingSum <- sum(bookSBing)
> BSBingSum
[1] -393
```

The sentiment analysis performed on the book yielded an average sentiment score of 0.027 for BSD and -0.169 for Bing. This suggests that the words in the book have a slightly good emotion for BSD and a negative feeling for Bing on average. It should be noted, however, that sentiment analysis is not always reliable, and it is likely that some terms or circumstances were not adequately caught by the sentiment dictionaries employed. Furthermore, sentiment analysis can only provide a basic approximation of a text's overall sentiment and should be used with caution.

```
> BSMean <- mean(bookSentiment)
> BSMean
[1] 0.02727273
> BSBingMean <- mean(bookSBing)
> BSBingMean
[1] -0.1693236
```

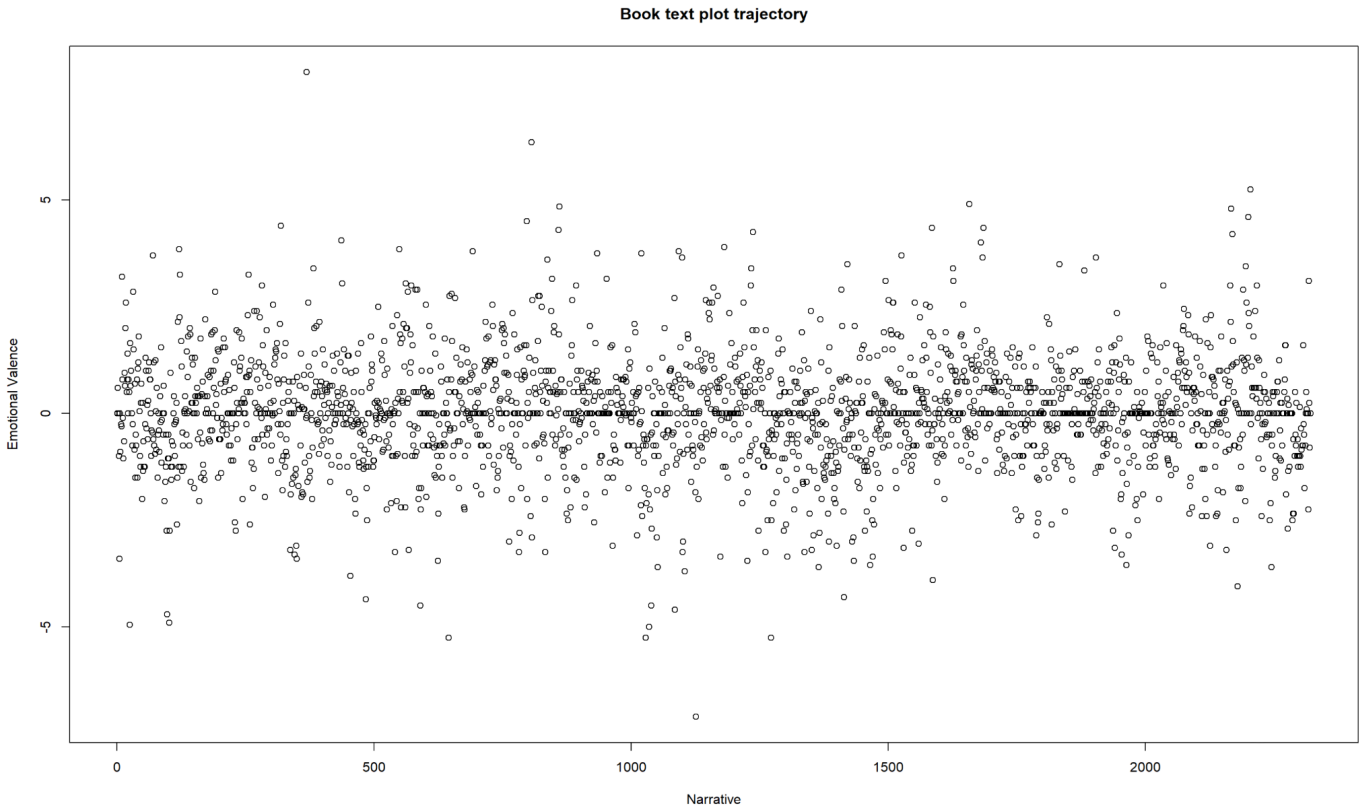
A statistical summary of a vector is provided by the summary function, which includes the minimum, first quartile, median, mean, third quartile, and maximum values.

```
> summary(BSBingMean)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
-0.1693 -0.1693 -0.1693 -0.1693 -0.1693 -0.1693
```

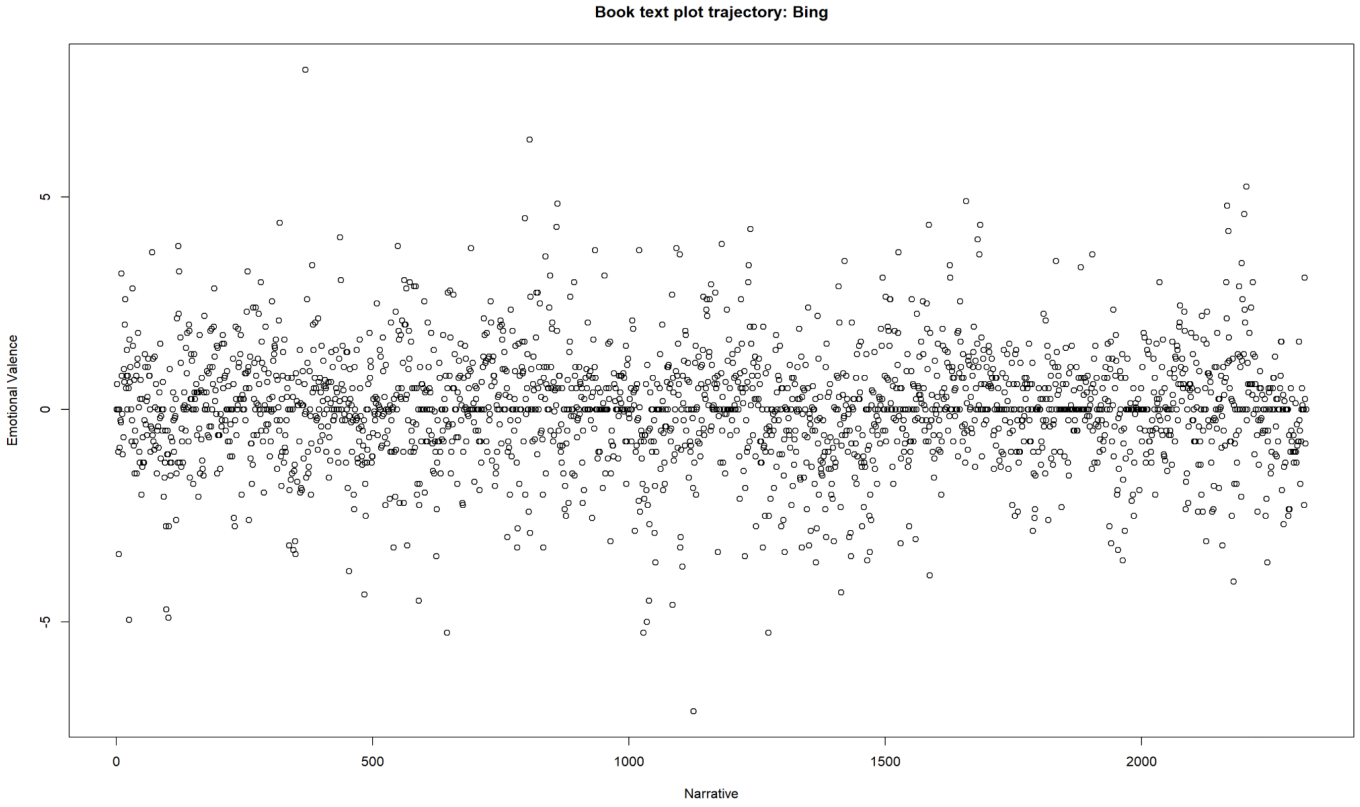
Plot:

The bookSentiment variable contains a vector of sentiment scores for each sentence in the book, and these scores are plotted on the y-axis. The x-axis represents the narrative, which could be the order of sentences in the book or some other measure of narrative progression. The title of the

plot is "Book text plot trajectory", and the x-axis and y-axis labels are "Narrative" and "Emotional Valence", respectively:



The code plot(bookSentiment, main = "Book text plot trajectory: Bing", xlab = "Narrative", ylab = "Emotional Valence") is used to create a plot of the emotional valence of the text of a book, as determined by the Bing sentiment dictionary. The plot shows the trajectory of emotional valence throughout the narrative of the book.




```
> BSSentimentPctValue <- get_percentage_values(bookSentiment, bins = 10)
> structure(BSSentimentPctValue)
          1           2           3           4           5           6           7           8           9          10
0.006437768 0.156681034 -0.006250000 0.240732759 -0.113362069 -0.179310345 -0.082758621 0.197844828 0.043750000 0.009051724
```

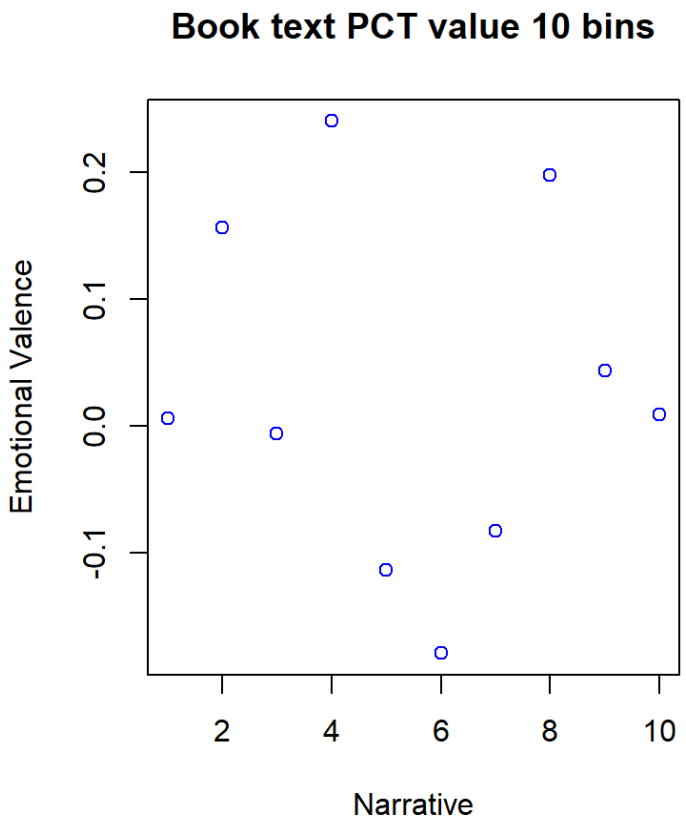
The output shows a numerical vector of length 10, with each element representing the percentage of positive or negative sentiment in each of 10 segments of the book. The values range from -0.179 to 0.241, indicating the varying degrees of positive or negative sentiment in different parts of the book.

BSSentimentPctValue is a numeric vector containing the percentage of positive or negative sentiment in each of the 10 equal-sized bins of the book text.

The plot() function is called with the BSSentimentPctValue vector as the first argument, and the main title, x-axis label, and y-axis label are specified using the main, xlab, and ylab arguments, respectively. The col

argument is used to specify the color of the points in the plot, in this case, blue.

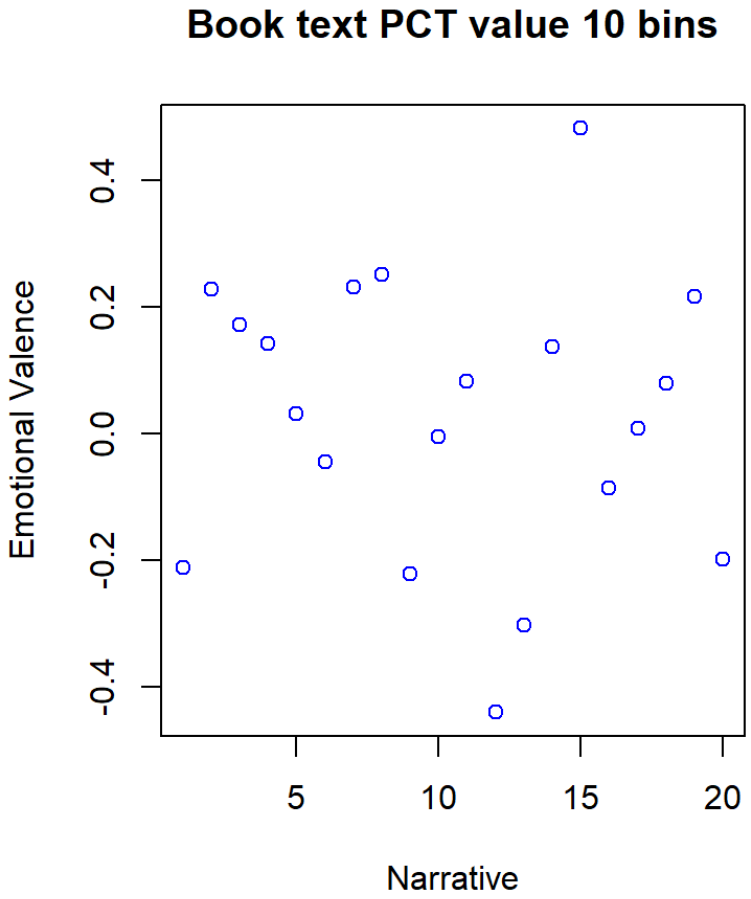
The resulting plot shows the percentage of positive or negative sentiment in each of the 10 bins, allowing the reader to see the general trajectory of emotional valence throughout the book:



The code BSSentimentPctValue - get_percentage_values(bookSentiment, bins = 20) divides the book text into 20 bins to calculate the percentage values of emotional valence. BSSentimentPctValue is a numerical vector containing the average percentage value of emotional valence for each bin. The structure() function is used to display the object's contents.

```
> structure(BSSentimentPctValue)
           1           2           3           4           5           6           7           8           9          10
 0.006437768  0.156681034 -0.006250000  0.240732759 -0.113362069 -0.179310345 -0.082758621  0.197844828  0.043750000  0.009051724
> plot(BSSentimentPctValue, main = "Book text PCT value 10 bins", xlab = "Narrative", ylab = "Emotional Valence", col = "blue")
> BSSentimentPctValue <- get_percentage_values(bookSentiment, bins = 20)
> structure(BSSentimentPctValue)
           1           2           3           4           5           6           7           8           9          10
-0.212393162  0.227155172  0.171551724  0.141810345  0.031896552 -0.044396552  0.231465517  0.250000000 -0.221551724 -0.005172414
          11          12          13          14          15          16          17          18          19          20
 0.081465517 -0.440086207 -0.302586207  0.137068966  0.481465517 -0.085775862  0.008620690  0.078879310  0.216810345 -0.198706897
> plot(BSSentimentPctValue, main = "Book text PCT value 10 bins", xlab = "Narrative", ylab = "Emotional Valence", col = "blue")
```

The code plot(BSSentimentPctValue, main = "Book text PCT value 10 bins", xlab = "Narrative", ylab = "Emotional Valence", col = "blue") creates a plot of the percentage values of the book's emotional sentiment scores across 10 bins. The x-axis represents the narrative of the book and the y-axis represents the emotional valence. The col = "blue" argument sets the color of the plot points to blue. This plot can help visualize how the emotional sentiment of the book changes over the course of the narrative.



Chapter-wise sentiment analysis

The for loop iterates over each book in the book_corpus and performs sentiment analysis on each chapter of the book.

The doc variable extracts the content of each book.

The get_sentences function from the syuzhet package is used to split the text into individual sentences.

The get_sentiment function from the same package is then used to compute the sentiment score for each sentence.

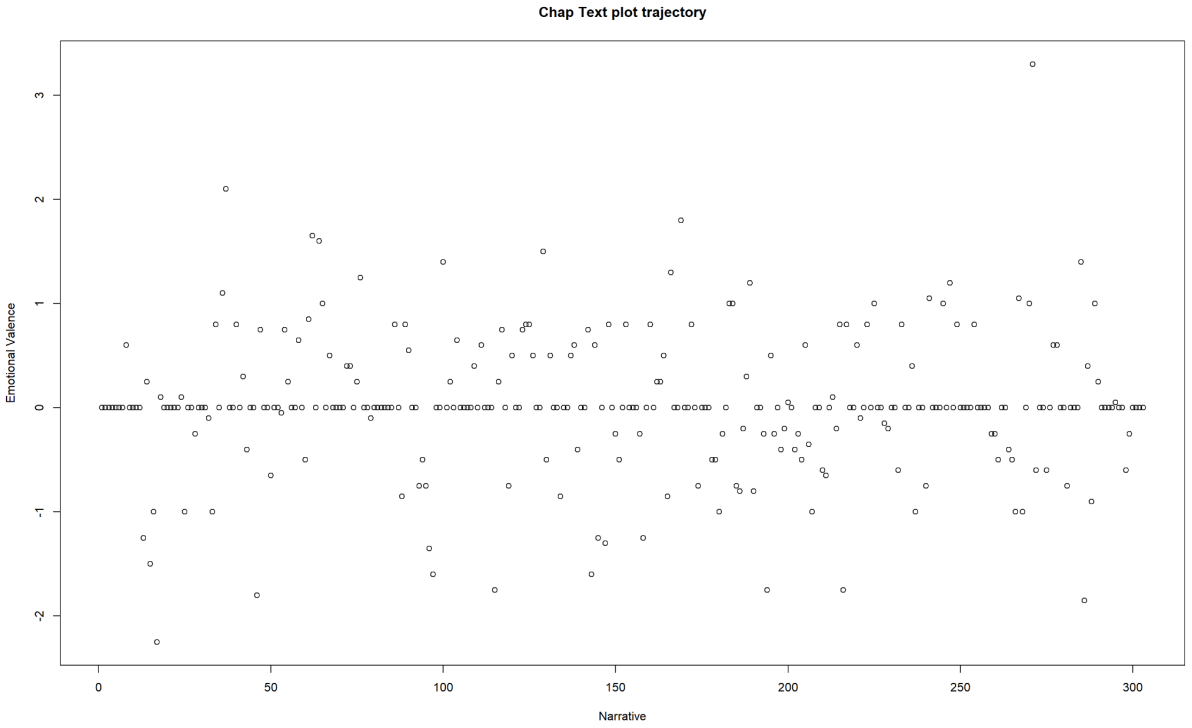
The print function is used to display the chapter number and the corresponding sentiment score for each chapter.

Finally, the plot function is used to visualize the sentiment trajectory of each chapter. The x-axis represents the narrative, and the y-axis represents the emotional valence.

```
> # chapter wise sentimental analysis
> for (i in 1:length(book_corpus)) {
+   doc <- content(book_corpus[i])
+   chapSentences <- get_sentences(doc[[1]]$content)
+   chapSent <- get_sentiment(chapSentences, method = "syuzhet")
+   print(paste("Chapter ", i))
+   print(chapSent)
+   plot(chapSent, main = "Chap Text plot trajectory", xlab = "Narrative", ylab = "Emotional Valence")
+ }
[1] "Chapter  1"
  [1]  0.00  0.00  0.00  0.00  0.00  0.00  0.00  0.60  0.00  0.00  0.00  0.00 -1.25  0.25 -1.50 -1.00 -2.25  0.10  0.00  0.00  0.00  0.00
 [23]  0.00  0.10 -1.00  0.00  0.00 -0.25  0.00  0.00  0.00 -0.10 -1.00  0.80  0.00  1.10  2.10  0.00  0.00  0.80  0.00  0.30 -0.40  0.00
```

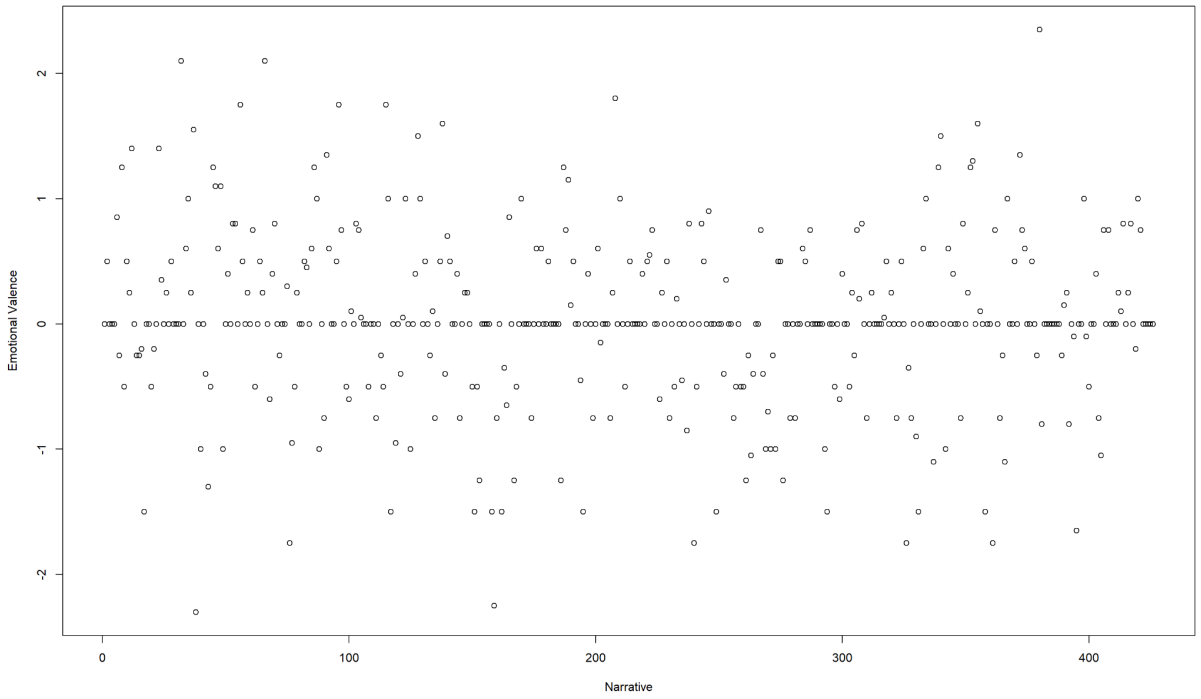
- **Chapter 1**

```
[1] "Chapter  1"
  [1]  0.00  0.00  0.00  0.00  0.00  0.00  0.60  0.00  0.00  0.00  0.00 -1.25  0.25 -1.50 -1.00 -2.25  0.10  0.00  0.00  0.00  0.00
 [23]  0.00  0.10 -1.00  0.00  0.00 -0.25  0.00  0.00  0.00 -0.10 -1.00  0.80  0.00  1.10  2.10  0.00  0.00  0.80  0.00  0.30 -0.40  0.00
 [45]  0.00 -1.80  0.75  0.00  0.00 -0.65  0.00  0.00 -0.05  0.75  0.25  0.00  0.00  0.65  0.00 -0.50  0.85  1.65  0.00  1.60  1.00  0.00
 [67]  0.50  0.00  0.00  0.00  0.00  0.40  0.40  0.00  0.25  1.25  0.00  0.00 -0.10  0.00  0.00  0.00  0.00  0.00  0.80  0.00 -0.85
 [89]  0.80  0.55  0.00  0.00 -0.75 -0.50 -0.75 -1.35 -1.60  0.00  0.00  1.40  0.00  0.25  0.00  0.65  0.00  0.00  0.00  0.00  0.40  0.00
[111]  0.60  0.00  0.00  0.00 -1.75  0.25  0.75  0.00 -0.75  0.50  0.00  0.00  0.75  0.80  0.80  0.50  0.00  0.00  1.50 -0.50  0.50  0.00
[133]  0.00 -0.85  0.00  0.00  0.50  0.60 -0.40  0.00  0.00  0.75 -1.60  0.60 -1.25  0.00 -1.30  0.80  0.00 -0.25 -0.50  0.00  0.80  0.00
[155]  0.00  0.00 -0.25 -1.25  0.00  0.80  0.00  0.25  0.25  0.50 -0.85  1.30  0.00  0.00  1.80  0.00  0.00  0.80  0.00 -0.75  0.00  0.00
[177]  0.00 -0.50 -0.50 -1.00 -0.25  0.00  1.00  1.00 -0.75 -0.80 -0.20  0.30  1.20 -0.80  0.00  0.00 -0.25 -1.75  0.50 -0.25  0.00 -0.40
[199] -0.20  0.05  0.00 -0.40 -0.25 -0.50  0.60 -0.35 -1.00  0.00  0.00 -0.60 -0.65  0.00  0.10 -0.20  0.80 -1.75  0.80  0.00  0.00  0.60
[221] -0.10  0.00  0.80  0.00  1.00  0.00  0.00 -0.15 -0.20  0.00  0.00 -0.60  0.80  0.00  0.00  0.40 -1.00  0.00  0.00 -0.75  1.05  0.00
[243]  0.00  0.00  1.00  0.00  1.20  0.00  0.80  0.00  0.00  0.00  0.00  0.80  0.00  0.00  0.00  0.00 -0.25 -0.25 -0.50  0.00  0.00 -0.40
[265] -0.50 -1.00  1.05 -1.00  0.00  1.00  3.30 -0.60  0.00  0.00 -0.60  0.00  0.60  0.60  0.00  0.00 -0.75  0.00  0.00  0.00  1.40 -1.85
[287]  0.40 -0.90  1.00  0.25  0.00  0.00  0.00  0.00  0.05  0.00  0.00 -0.60 -0.25  0.00  0.00  0.00  0.00
```



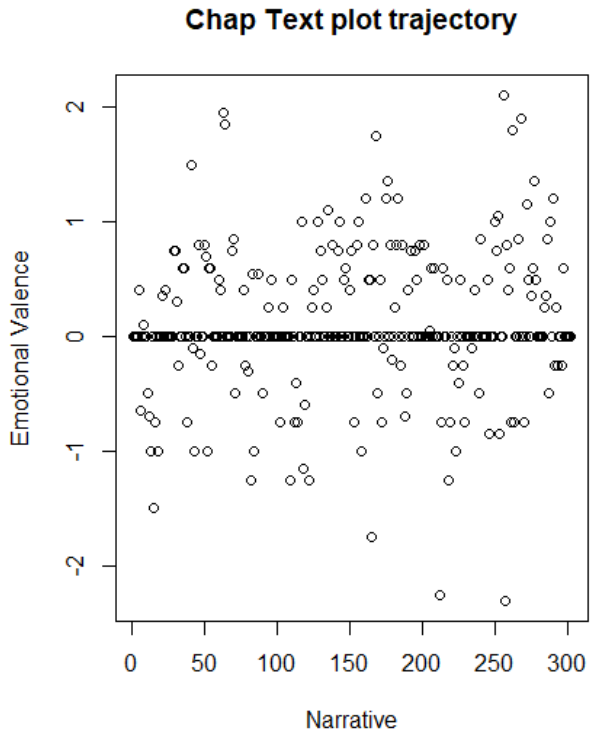
- **Chapter 2**

```
[1] "Chapter  2"
  [1]  0.00  0.50  0.00  0.00  0.00  0.85 -0.25  1.25 -0.50  0.50  0.25  1.40  0.00 -0.25 -0.25 -0.20 -1.50  0.00  0.00 -0.50 -0.20  0.00
 [23]  1.40  0.35  0.00  0.25  0.00  0.50  0.00  0.00  0.00  2.10  0.00  0.60  1.00  0.25  1.55 -2.30  0.00 -1.00  0.00 -0.40 -1.30 -0.50
 [45]  1.25  1.10  0.60  1.10 -1.00  0.00  0.40  0.00  0.80  0.80  0.00  1.75  0.50  0.00  0.25  0.00  0.75 -0.50  0.00  0.50  0.25  2.10
 [67]  0.00 -0.60  0.40  0.80  0.00 -0.25  0.00  0.00  0.30 -1.75 -0.95 -0.50  0.25  0.00  0.00  0.50  0.45  0.00  0.60  1.25  1.00 -1.00
 [89]  0.00 -0.75  1.35  0.60  0.00  0.00  0.50  1.75  0.75  0.00 -0.50 -0.60  0.10  0.00  0.80  0.75  0.05  0.00  0.00 -0.50  0.00  0.00
[111] -0.75  0.00 -0.25 -0.50  1.75  1.00 -1.50  0.00 -0.95  0.00 -0.40  0.05  1.00  0.00 -1.00  0.00  0.40  1.50  1.00  0.00  0.50  0.00
[133] -0.25  0.10 -0.75  0.00  0.50  1.60 -0.40  0.70  0.50  0.00  0.00  0.40 -0.75  0.00  0.25  0.25  0.00 -0.50 -1.50 -0.50 -1.25  0.00
[155]  0.00  0.00  0.00 -1.50 -2.25 -0.75  0.00 -1.50 -0.35 -0.65  0.85  0.00 -1.25 -0.50  0.00  1.00  0.00  0.00  0.00 -0.75  0.00  0.60
[177]  0.00  0.60  0.00  0.00  0.50  0.00  0.00  0.00  0.00 -1.25  1.25  0.75  1.15  0.15  0.50  0.00  0.00 -0.45 -1.50  0.00  0.40  0.00
[199] -0.75  0.00  0.60 -0.15  0.00  0.00  0.00 -0.75  0.25  1.80  0.00  1.00  0.00 -0.50  0.00  0.50  0.00  0.00  0.00  0.00  0.40  0.00
[221]  0.50  0.55  0.75  0.00  0.00 -0.60  0.25  0.00  0.50 -0.75  0.00 -0.50  0.20  0.00 -0.45  0.00 -0.85  0.80  0.00 -1.75 -0.50  0.00
[243]  0.80  0.50  0.00  0.90  0.00  0.00 -1.50  0.00  0.00 -0.40  0.35  0.00  0.00 -0.75 -0.50  0.00 -0.50 -0.50 -1.25 -0.25 -1.05 -0.40
[265]  0.00  0.00  0.75 -0.40 -1.00 -0.70 -1.00 -0.25 -1.00  0.50  0.50 -1.25  0.00  0.00 -0.75  0.00 -0.75  0.00  0.00  0.60  0.50  0.00
[287]  0.75  0.00  0.00  0.00  0.00  0.00 -1.00 -1.50  0.00  0.00 -0.50  0.00 -0.60  0.40  0.00  0.00 -0.50  0.25 -0.25  0.75  0.20  0.80
[309]  0.00 -0.75  0.00  0.25  0.00  0.00  0.00  0.00  0.05  0.50  0.00  0.25  0.00 -0.75  0.00  0.50  0.00 -1.75 -0.35 -0.75  0.00 -0.90
[331] -1.50  0.00  0.60  1.00  0.00  0.00 -1.10  0.00  1.25  1.50  0.00 -1.00  0.60  0.00  0.40  0.00  0.00 -0.75  0.80  0.00  0.25  1.25
[353]  1.30  0.00  1.60  0.10  0.00 -1.50  0.00  0.00 -1.75  0.75  0.00 -0.75 -0.25 -1.10  1.00  0.00  0.00  0.50  0.00  1.35  0.75  0.60
[375]  0.00  0.00  0.50  0.00 -0.25  2.35 -0.80  0.00  0.00  0.00  0.00  0.00  0.00 -0.25  0.15  0.25 -0.80  0.00 -0.10 -1.65  0.00
[397]  0.00  1.00 -0.10 -0.50  0.00  0.00  0.40 -0.75 -1.05  0.75  0.00  0.75  0.00  0.00  0.00  0.25  0.10  0.80  0.00  0.25  0.80  0.00
[419] -0.20  1.00  0.75  0.00  0.00  0.00  0.00  0.00
```



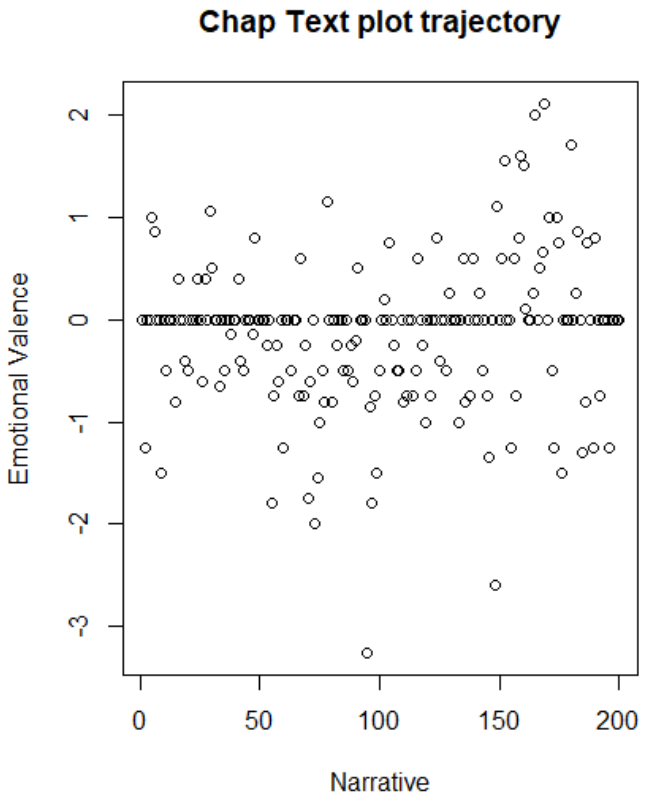
- **Chapter 3**

```
[1] "Chapter  3"
  [1]  0.00  0.00  0.00  0.00  0.40 -0.65  0.00  0.10  0.00  0.00 -0.50 -0.70 -1.00  0.00 -1.50 -0.75  0.00 -1.00  0.00  0.00  0.35  0.00
 [23]  0.40  0.00  0.00  0.00  0.00  0.00  0.75  0.75  0.30 -0.25  0.00  0.00  0.60  0.60  0.00 -0.75  0.00  0.00  1.50 -0.10 -1.00  0.00
 [45]  0.00  0.80 -0.15  0.00  0.00  0.80  0.70 -1.00  0.60  0.60 -0.25  0.00  0.00  0.00  0.00  0.50  0.40  0.00  1.95  1.85  0.00  0.00
 [67]  0.00  0.00  0.75  0.85 -0.50  0.00  0.00  0.00  0.00  0.00  0.40 -0.25  0.00 -0.30  0.00 -1.25  0.55 -1.00  0.00  0.00  0.55  0.00
 [89]  0.00 -0.50  0.00  0.00  0.00  0.25  0.00  0.50  0.00  0.00  0.00  0.00  0.00 -0.75  0.00  0.25  0.00  0.00  0.00  0.00 -1.25  0.50
[111]  0.00 -0.75 -0.40 -0.75  0.00  0.00  1.00 -1.15 -0.60  0.00  0.00 -1.25  0.00  0.25  0.40  0.00  0.00  1.00  0.00  0.75  0.50  0.00
[133]  0.00  0.25  1.10  0.00  0.00  0.80  0.00  0.00  0.00  0.75  1.00  0.00  0.00  0.50  0.60  0.00  0.00  0.40  0.75  0.00 -0.75  0.00
[155]  0.80  1.00  0.00 -1.00  0.00  0.00  1.20  0.00  0.50  0.50 -1.75  0.80  0.00  1.75 -0.50  0.00  0.50 -0.75 -0.10  0.00  1.20  1.35
[177]  0.00  0.80 -0.20  0.00  0.25  0.80  1.20  0.00 -0.25  0.80  0.00 -0.70 -0.50  0.40  0.00  0.75  0.00  0.00  0.75  0.50  0.00  0.80
[199]  0.00  0.00  0.80  0.00  0.00  0.00  0.05  0.60  0.00  0.60  0.00  0.00  0.00 -2.25 -0.75  0.60  0.00  0.00  0.50 -1.25 -0.75  0.00
[221] -0.25 -0.10 -1.00  0.00 -0.40  0.50  0.00 -0.25 -0.75  0.00  0.00  0.00  0.00 -0.10  0.00  0.40  0.00  0.00 -0.50  0.85  0.00  0.00
[243]  0.00  0.00  0.50 -0.85  0.00  0.00  0.00  1.00  0.75  1.05 -0.85  0.00  0.00  2.10 -2.30  0.80  0.40  0.60 -0.75  1.80 -0.75  0.00
[265]  0.00  0.85  0.00  1.90  0.00 -0.75  0.00  1.15  0.50  0.00  0.35  0.60  1.35  0.50  0.00  0.00  0.00  0.00  0.00  0.25  0.35  0.85
[287] -0.50  1.00  0.00  1.20 -0.25  0.25 -0.25  0.00  0.00 -0.25  0.60  0.00  0.00  0.00  0.00  0.00
[1] "Chapter  4"
```



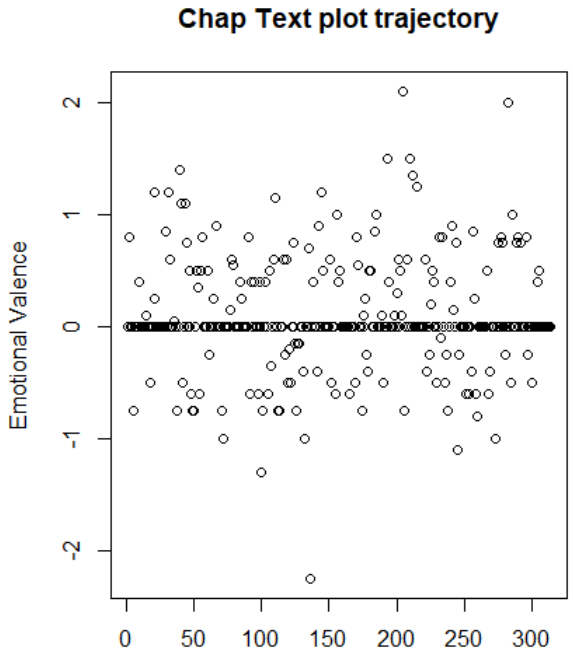
- **Chapter 4**

```
  [1]  0.00 -1.25  0.00  0.00  1.00  0.85  0.00  0.00 -1.50  0.00 -0.50  0.00  0.00  0.00 -0.80  0.40  0.00  0.00 -0.40 -0.50  0.00  0.00
 [23]  0.00  0.40  0.00 -0.60  0.40  0.00  1.05  0.50  0.00  0.00 -0.65  0.00 -0.50  0.00  0.00 -0.15  0.00  0.00  0.40 -0.40 -0.50  0.00
 [45]  0.00  0.00 -0.15  0.80  0.00  0.00  0.00  0.00 -0.25  0.00 -1.80 -0.75 -0.25 -0.60  0.00 -1.25  0.00  0.00 -0.50  0.00  0.00 -0.75
 [67]  0.60 -0.75 -0.25 -1.75 -0.60  0.00 -2.00 -1.55 -1.00 -0.50 -0.80  1.15  0.00 -0.80  0.00 -0.25  0.00  0.00 -0.50  0.00 -0.50 -0.25
 [89] -0.60 -0.20  0.50  0.00  0.00  0.00 -3.25 -0.85 -1.80 -0.75 -1.50 -0.50  0.00  0.20  0.00  0.75  0.00 -0.25 -0.50 -0.50  0.00 -0.80
[111] -0.75  0.00  0.00 -0.75 -0.50  0.60  0.00 -0.25 -1.00  0.00 -0.75  0.00  0.00  0.80 -0.40  0.00  0.00 -0.50  0.25  0.00  0.00  0.00
[133] -1.00  0.00  0.60 -0.80  0.00 -0.75  0.60  0.00  0.00  0.25 -0.50  0.00 -0.75 -1.35  0.00 -2.60  1.10  0.00  0.60  1.55  0.00  0.00
[155] -1.25  0.60 -0.75  0.80  1.60  1.50  0.10  0.00  0.00  0.25  2.00  0.00  0.50  0.65  2.10  0.00  1.00 -0.50 -1.25  1.00  0.75 -1.50
[177]  0.00  0.00  0.00  1.70  0.00  0.25  0.85  0.00 -1.30 -0.80  0.75  0.00 -1.25  0.80  0.00 -0.75  0.00  0.00  0.00 -1.25  0.00  0.00
[199]  0.00  0.00
```



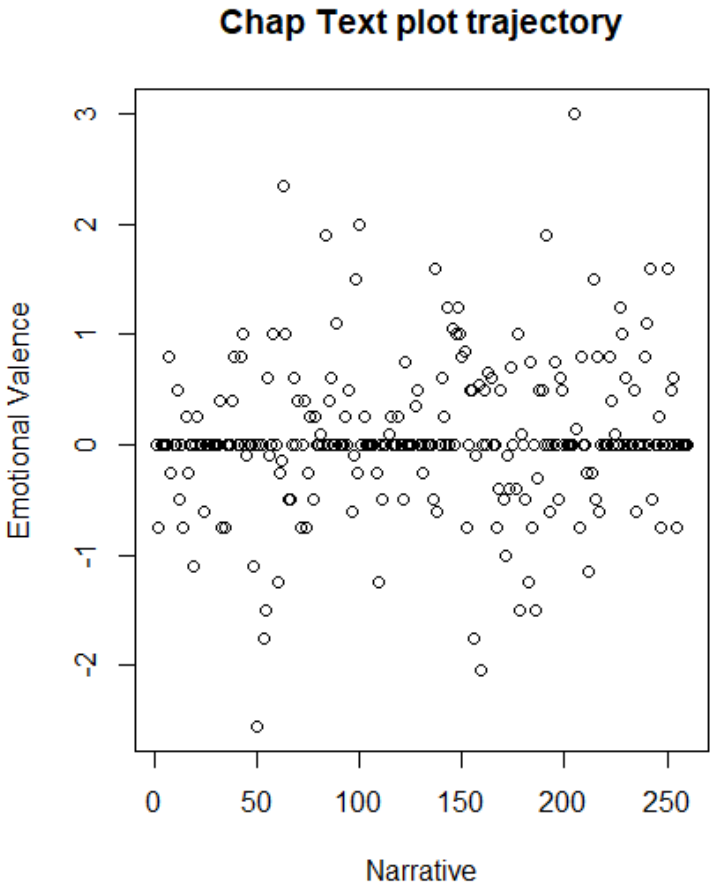
- **Chapter 5**

```
[1] "Chapter  5"
  [1]  0.00  0.80  0.00  0.00 -0.75  0.00  0.00  0.00  0.40  0.00  0.00  0.00  0.00  0.10  0.00  0.00 -0.50  0.00  0.00  1.20  0.25  0.00
 [23]  0.00  0.00  0.00  0.00  0.00  0.00  0.85  0.00  1.20  0.60  0.00  0.00  0.05  0.00 -0.75  0.00  1.40  1.10 -0.50  0.00  1.10  0.75
 [45]  0.00  0.00  0.50 -0.60 -0.75 -0.75  0.00  0.50  0.35 -0.60  0.50  0.80  0.00  0.00  0.00  0.50 -0.25  0.00  0.00  0.25  0.00  0.90
 [67]  0.00  0.00  0.00 -0.75 -1.00  0.00  0.00  0.00  0.00  0.00  0.15  0.60  0.55  0.00  0.00  0.00  0.00  0.40  0.25  0.00  0.00  0.00
 [89]  0.00  0.80 -0.60  0.40  0.00  0.40  0.00  0.00 -0.60  0.40  0.00 -1.30 -0.75  0.00  0.40  0.00 -0.60  0.50 -0.35  0.00  0.60  1.15
[111]  0.00 -0.75 -0.75  0.00  0.00  0.60 -0.25  0.60 -0.50 -0.20 -0.50  0.00  0.75  0.00 -0.15 -0.75 -0.15 -0.15  0.00  0.00 -0.40 -1.00
[133]  0.00  0.00  0.70 -2.25  0.00  0.40  0.00  0.00 -0.40  0.90  0.00  1.20  0.50  0.00  0.00  0.00  0.00  0.00  0.60 -0.50  0.00  0.00
[155] -0.60  1.00  0.40  0.50  0.00  0.00  0.00  0.00  0.00  0.00 -0.60  0.00  0.00  0.00 -0.50  0.80  0.55  0.00  0.00 -0.75  0.10  0.00
[177]  0.25 -0.25 -0.40  0.50  0.50  0.00  0.00  0.85  1.00  0.00  0.00  0.00  0.10 -0.50  0.00  0.00  1.50  0.40  0.00  0.00  0.00  0.10
[199]  0.00  0.30  0.60  0.00  0.50  0.10  2.10 -0.75  0.00  0.60  0.00  1.50  0.00  1.35  0.00  0.00  1.25  0.00  0.00  0.00  0.00  0.00
[221]  0.60 -0.40  0.00 -0.25  0.20  0.50  0.00  0.40  0.00 -0.50  0.00  0.80 -0.10  0.80  0.00 -0.50 -0.25 -0.75  0.00  0.40  0.90  0.15
[243]  0.00  0.75 -1.10 -0.25  0.00  0.00  0.00  0.00 -0.60  0.00  0.00 -0.60  0.00 -0.40  0.85  0.25 -0.60 -0.80  0.00  0.00  0.00  0.00
[265]  0.00  0.00  0.50 -0.60 -0.40  0.00  0.00  0.00 -1.00  0.00  0.75  0.00  0.80  0.75  0.00  0.00 -0.25  0.00  2.00  0.00 -0.50  1.00
[287]  0.00  0.00  0.75  0.80  0.00  0.75  0.00  0.00  0.00  0.80 -0.25  0.00  0.00 -0.50  0.00  0.00  0.00  0.40  0.00  0.50  0.00  0.00
[309]  0.00  0.00  0.00  0.00  0.00  0.00
```



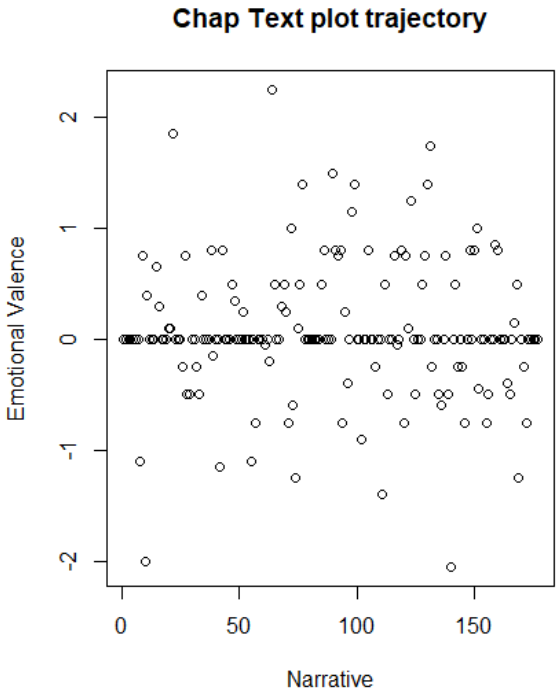
- **Chapter 6**

```
[1] "Chapter  6"
  [1]  0.00 -0.75  0.00  0.00  0.00  0.00  0.80 -0.25  0.00  0.00  0.50 -0.50  0.00 -0.75  0.25 -0.25  0.00  0.00 -1.10  0.00  0.25  0.00
 [23]  0.00 -0.60  0.00  0.00  0.00  0.00  0.00  0.00  0.00  0.40 -0.75 -0.75  0.00  0.00  0.00  0.40  0.80  0.00  0.00  0.80  1.00  0.00
 [45] -0.10  0.00  0.00 -1.10  0.00 -2.55  0.00  0.00 -1.75 -1.50  0.60 -0.10  0.00  1.00  0.00 -1.25 -0.25 -0.15  2.35  1.00 -0.50 -0.50
 [67]  0.00  0.60  0.00  0.40 -0.75  0.00  0.40 -0.75 -0.25  0.25 -0.50  0.25  0.00  0.00  0.10  0.00  1.90  0.00  0.40  0.60  0.00  0.00
 [89]  1.10  0.00  0.00  0.00  0.25  0.00  0.50 -0.60 -0.10  1.50 -0.25  2.00  0.00  0.25  0.00  0.00  0.00  0.00  0.00 -0.25 -1.25  0.00
[111] -0.50  0.00  0.00  0.10  0.25  0.00  0.00  0.00  0.25  0.00 -0.50  0.75  0.00  0.00  0.00  0.00  0.35  0.50  0.00  0.00 -0.25  0.00
[133]  0.00  0.00  0.00 -0.50  1.60 -0.60  0.00  0.60  0.25  0.00  1.25  0.00  1.05  0.00  1.00  1.25  1.00  0.80  0.85 -0.75  0.00  0.50
[155]  0.50 -1.75 -0.10  0.55 -2.05  0.00  0.50  0.00  0.65  0.60  0.00  0.00 -0.75 -0.40  0.50 -0.50 -1.00 -0.10 -0.40  0.70  0.00 -0.40
[177]  1.00 -1.50  0.10  0.00 -0.50 -1.25  0.75 -0.75  0.00 -1.50 -0.30  0.50  0.50  0.00  1.90  0.00 -0.60  0.00  0.75  0.00 -0.50  0.60
[199]  0.50  0.00  0.00  0.00  0.00  0.00  3.00  0.15 -0.75  0.80  0.00  0.00 -0.25 -1.15 -0.25  1.50 -0.50  0.80 -0.60  0.00  0.00  0.00
[221]  0.00  0.80  0.40  0.00  0.10  0.00  1.25  1.00  0.00  0.60  0.00  0.00  0.00  0.50 -0.60  0.00  0.00  0.00  0.80  1.10  0.00  1.60
[243] -0.50  0.00  0.00  0.25 -0.75  0.00  0.00  1.60  0.00  0.50  0.60  0.00 -0.75  0.00  0.00  0.00  0.00  0.00
```



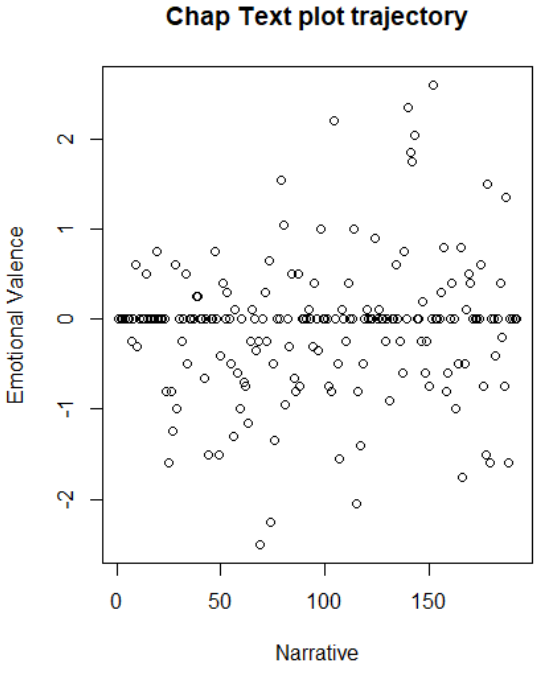
- **Chapter 7**

```
[1] "Chapter  7"
  [1]  0.00  0.00  0.00  0.00  0.00  0.00 -1.10  0.75 -2.00  0.40  0.00  0.00  0.00  0.65  0.30  0.00  0.00  0.00  0.10  0.10  1.85
 [23]  0.00  0.00  0.00 -0.25  0.75 -0.50 -0.50  0.00  0.00 -0.25 -0.50  0.40  0.00  0.00  0.00  0.80 -0.15  0.00  0.00 -1.15  0.80  0.00
 [45]  0.00  0.00  0.50  0.35  0.00  0.00  0.00  0.25  0.00  0.00 -1.10  0.00 -0.75  0.00  0.00  0.00 -0.05  0.00 -0.20  2.25  0.50  0.00
 [67]  0.00  0.30  0.50  0.25 -0.75  1.00 -0.60 -1.25  0.10  0.50  1.40  0.00  0.00  0.00  0.00  0.00  0.00  0.00  0.50  0.80  0.00  0.00
 [89]  0.00  1.50  0.80  0.75  0.80 -0.75  0.25 -0.40  0.00  1.15  1.40  0.00  0.00 -0.90  0.00  0.00  0.80  0.00  0.00 -0.25  0.00  0.00
[111] -1.40  0.50 -0.50  0.00  0.00  0.75 -0.05  0.00  0.80 -0.75  0.75  0.10  1.25  0.00 -0.50  0.00  0.00  0.50  0.75  1.40  1.75 -0.25
[133]  0.00  0.00 -0.50 -0.60  0.00  0.75 -0.50 -2.05  0.00  0.50 -0.25  0.00 -0.25 -0.75  0.00  0.80  0.00  0.80  1.00 -0.45  0.00  0.00
[155] -0.75 -0.50  0.00  0.00  0.85  0.80  0.00  0.00  0.00 -0.40 -0.50  0.00  0.15  0.50 -1.25  0.00 -0.25 -0.75  0.00  0.00  0.00  0.00
[177]  0.00
```



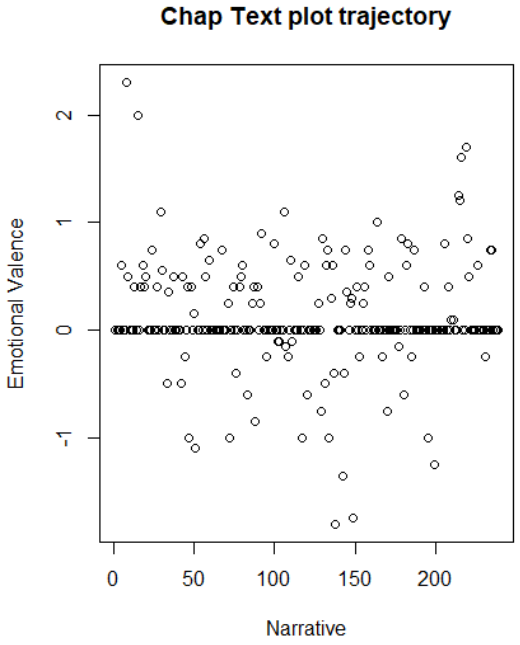
- **Chapter 8**

```
[1] "Chapter  8"
  [1]  0.00  0.00  0.00  0.00  0.00 -0.25  0.00  0.60 -0.30  0.00  0.00  0.00  0.50  0.00  0.00  0.00  0.00  0.75  0.00  0.00  0.00
 [23]  0.00 -0.80 -1.60 -0.80 -1.25  0.60 -1.00  0.00 -0.25  0.00  0.50 -0.50  0.00  0.00  0.00  0.25  0.25  0.00  0.00 -0.65  0.00 -1.50
 [45]  0.00  0.00  0.75  0.00 -1.50 -0.40  0.40  0.00  0.30  0.00 -0.50 -1.30  0.10 -0.60 -1.00  0.00 -0.70 -0.75 -1.15 -0.25  0.10  0.00
 [67] -0.35 -0.25 -2.50  0.00  0.30 -0.25  0.65 -2.25 -0.50 -1.35  0.00  0.00  1.55  1.05 -0.95  0.00 -0.30  0.50 -0.65 -0.80  0.50 -0.75
 [89]  0.00  0.00  0.00  0.10  0.00 -0.30  0.40  0.00 -0.35  1.00  0.00  0.00  0.00 -0.75 -0.80  2.20  0.00 -0.50 -1.55  0.10  0.00 -0.25
[111]  0.40  0.00  0.00  1.00 -2.05 -0.80 -1.40 -0.50  0.00  0.10  0.00  0.00  0.00  0.90  0.00  0.10  0.00  0.00 -0.25  0.00 -0.90  0.00
[133]  0.00  0.60  0.00 -0.25 -0.60  0.75  0.00  2.35  1.85  1.75  2.05  0.00  0.00 -0.25  0.20 -0.60 -0.25 -0.75  0.00  2.60  0.00  0.00
[155]  0.00  0.30  0.80 -0.80 -0.60  0.00  0.40  0.00 -1.00 -0.50  0.80 -1.75 -0.50  0.10  0.50  0.40  0.00  0.00  0.00  0.00  0.60 -0.75
[177] -1.50  1.50 -1.60  0.00  0.00 -0.40  0.00  0.40 -0.20 -0.75  1.35 -1.60  0.00  0.00  0.00  0.00
```



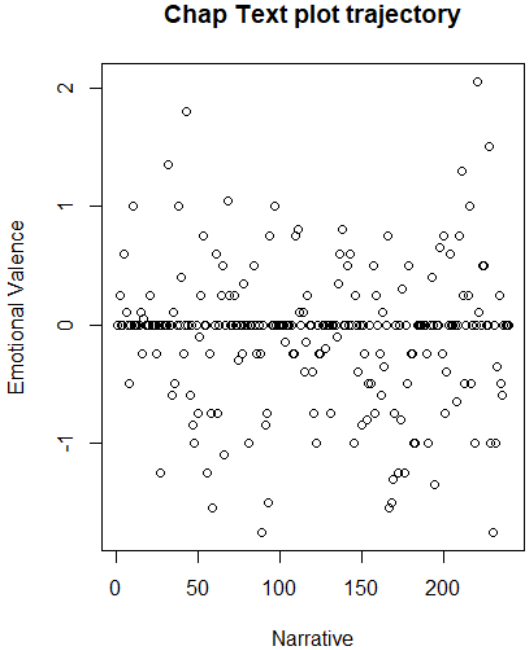
- **Chapter 9**

```
[1] "Chapter  9"
  [1]  0.00  0.00  0.00  0.00  0.60  0.00  0.00  2.30  0.50  0.00  0.00  0.00  0.40  0.00  2.00  0.00  0.40  0.60  0.40  0.50  0.00  0.00
 [23]  0.00  0.75  0.00  0.00  0.40  0.00  1.10  0.55  0.00  0.00 -0.50  0.35  0.00  0.00  0.50  0.00  0.00  0.00  0.00 -0.50  0.50 -0.25
 [45]  0.00  0.40 -1.00  0.40  0.00  0.15 -1.10  0.00  0.00  0.80  0.00  0.85  0.50  0.00  0.65  0.00  0.00  0.00  0.00  0.00  0.00  0.00
 [67]  0.75  0.00  0.00  0.00  0.25 -1.00  0.00  0.40  0.00 -0.40  0.00  0.40  0.50  0.60  0.00  0.00 -0.60  0.00  0.00  0.25  0.40 -0.85
 [89]  0.40  0.00  0.25  0.90  0.00  0.00 -0.25  0.00  0.00  0.00  0.00  0.80  0.00 -0.10 -0.10  0.00  0.00  1.10 -0.15 -0.25  0.00  0.65
[111] -0.10  0.00  0.00  0.00  0.50  0.00 -1.00  0.00  0.60 -0.60  0.00  0.00  0.00  0.00  0.00  0.00  0.25  0.00 -0.75  0.85 -0.50  0.60
[133]  0.75 -1.00  0.30  0.60 -0.40 -1.80  0.00  0.00  0.00 -1.35 -0.40  0.75  0.35  0.00  0.25  0.30 -1.75  0.00  0.40  0.00 -0.25  0.00
[155]  0.25  0.40  0.00  0.75  0.60  0.00  0.00  0.00  0.00  1.00  0.00  0.00 -0.25  0.00  0.00 -0.75  0.50  0.00  0.00  0.00  0.00  0.00
[177] -0.15  0.00  0.85 -0.60  0.00  0.60  0.80  0.00 -0.25  0.00  0.75  0.00  0.00  0.00  0.00  0.00  0.40  0.00 -1.00  0.00  0.00  0.00
[199] -1.25  0.00  0.00  0.00  0.00  0.00  0.00  0.80  0.00  0.40  0.00  0.10  0.10  0.00  0.00  1.25  1.20  1.60  0.00  0.00  1.70  0.85
[221]  0.50  0.00  0.00  0.00  0.00  0.60  0.00  0.00  0.00  0.00 -0.25  0.00  0.00  0.75  0.75  0.00  0.00  0.00  0.00
```



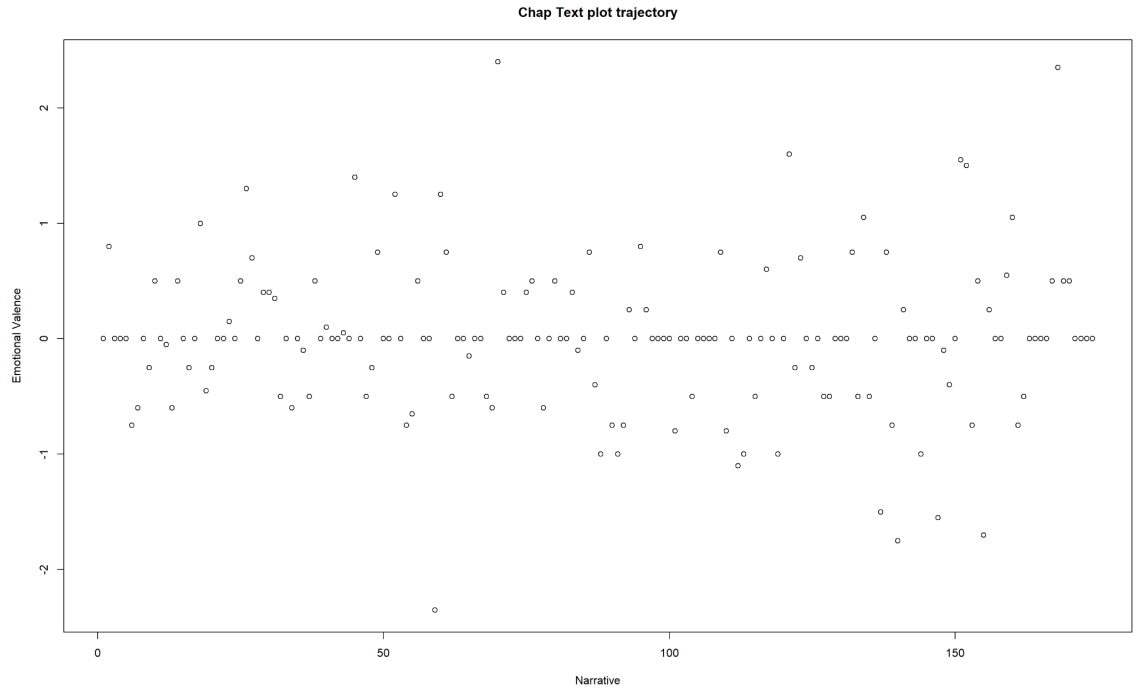
- **Chapter 10**

```
[1] "Chapter  10"
  [1]  0.00  0.25  0.00  0.00  0.60  0.10  0.00 -0.50  0.00  1.00  0.00  0.00  0.00  0.00  0.10 -0.25  0.05  0.00  0.00  0.00  0.25  0.00
 [23]  0.00  0.00 -0.25  0.00 -1.25  0.00  0.00  0.00  0.00  1.35  0.00 -0.60  0.10 -0.50  0.00  1.00  0.00  0.40 -0.25  0.00  1.80  0.00
 [45] -0.60  0.00 -0.85 -1.00  0.00 -0.75 -0.10  0.25  0.75  0.00  0.00 -1.25 -0.25 -0.75 -1.55  0.00  0.60 -0.75  0.00  0.25  0.50 -1.10
 [67]  0.00  1.05  0.25  0.00  0.00  0.25  0.00  0.00 -0.30  0.00 -0.25  0.35  0.00  0.00 -1.00  0.00  0.00  0.50  0.00 -0.25  0.00 -0.25
 [89] -1.75  0.00 -0.85 -0.75 -1.50  0.75  0.00  0.00  1.00  0.00  0.00  0.00  0.00  0.00 -0.15  0.00  0.00  0.00  0.00 -0.25 -0.25  0.75
[111]  0.80  0.10  0.00  0.10 -0.40 -0.15  0.25  0.00  0.00 -0.40 -0.75 -1.00  0.00 -0.25 -0.25  0.00  0.00 -0.20  0.00  0.00 -0.75  0.00
[133]  0.00  0.00 -0.10  0.35  0.60  0.80  0.00  0.00  0.50  0.00  0.60  0.00 -1.00  0.25  0.00 -0.40  0.00 -0.85  0.00 -0.25 -0.80 -0.50
[155]  0.00 -0.50  0.50 -0.75  0.25  0.00 -0.25 -0.60  0.10 -0.35  0.00  0.75 -1.55 -1.50 -1.30 -0.75  0.00 -1.25  0.00 -0.80  0.30 -1.25
[177]  0.00 -0.50  0.50 -0.25 -0.25 -1.00 -1.00  0.00  0.00  0.00  0.00  0.00  0.00 -0.25 -1.00  0.00  0.40  0.00 -1.35  0.00  0.00  0.65
[199] -0.25  0.75 -0.75 -0.40  0.00  0.60  0.00  0.00  0.00 -0.65  0.00  0.75  1.30  0.25 -0.50  0.00  0.25  1.00 -0.50  0.00 -1.00  0.00
[221]  2.05  0.10  0.00  0.50  0.50  0.00  0.00  1.50 -1.00 -1.75  0.00 -1.00 -0.35  0.25 -0.50 -0.60  0.00  0.00  0.00  0.00
```



- **Chapter 11**

```
[1] "Chapter  11"
  [1]  0.00  0.80  0.00  0.00  0.00 -0.75 -0.60  0.00 -0.25  0.50  0.00 -0.05 -0.60  0.50  0.00 -0.25  0.00  1.00 -0.45 -0.25  0.00  0.00
 [23]  0.15  0.00  0.50  1.30  0.70  0.00  0.40  0.40  0.35 -0.50  0.00 -0.60  0.00 -0.10 -0.50  0.50  0.00  0.10  0.00  0.00  0.05  0.00
 [45]  1.40  0.00 -0.50 -0.25  0.75  0.00  0.00  1.25  0.00 -0.75 -0.65  0.50  0.00  0.00 -2.35  1.25  0.75 -0.50  0.00  0.00 -0.15  0.00
 [67]  0.00 -0.50 -0.60  2.40  0.40  0.00  0.00  0.00  0.40  0.50  0.00 -0.60  0.00  0.50  0.00  0.00  0.40 -0.10  0.00  0.75 -0.40 -1.00
 [89]  0.00 -0.75 -1.00 -0.75  0.25  0.00  0.80  0.25  0.00  0.00  0.00  0.00 -0.80  0.00  0.00 -0.50  0.00  0.00  0.00  0.00  0.75 -0.80
[111]  0.00 -1.10 -1.00  0.00 -0.50  0.00  0.60  0.00 -1.00  0.00  1.60 -0.25  0.70  0.00 -0.25  0.00 -0.50 -0.50  0.00  0.00  0.00  0.75
[133] -0.50  1.05 -0.50  0.00 -1.50  0.75 -0.75 -1.75  0.25  0.00  0.00 -1.00  0.00  0.00 -1.55 -0.10 -0.40  0.00  1.55  1.50 -0.75  0.50
[155] -1.70  0.25  0.00  0.00  0.55  1.05 -0.75 -0.50  0.00  0.00  0.00  0.00  0.50  2.35  0.50  0.50  0.00  0.00  0.00  0.00
> |
```



stringi package

Stringi is a R package that provides a comprehensive range of string processing methods for working with Unicode strings. It is based on the ICU library (International Components for Unicode), which is a collection of C/C++ libraries that enable Unicode and globalization. Because it supports more than 150 character sets and can handle both normal ASCII and Unicode strings, the stringi package is very handy for dealing with multilingual material. Regular expressions, string searching, pattern matching, case conversions, text segmentation, collation, transliteration, and many other functions are available in the package to manipulate and analyze strings.

stri_split_fixed(): This function splits a string into pieces using a fixed pattern and returns the pieces as a character vector.
stri_sort(): This function sorts a character vector in ascending order.
stri_count_regex(): This function counts the number of times a regular expression occurs in a string.

```
> # 3 functions of each package
> # stringi package
> sentence2 <- bookcl[[1]]$content[2]
> sentence2
[1] "ON THE ARIZONA HILLS"
> strings <- stringi::stri_split_fixed(sentence2, "THE")
> stringi::stri_sort(strings[[1]])
[1] " ARIZONA HILLS" "ON "
> stringi::stri_count_regex(sentence2, "T.")
[1] 1
> sentence2
[1] "ON THE ARIZONA HILLS"
> stringi::stri_trans_tolower(sentence2)
[1] "on the arizona hills"
```

quanteda package

tokens(): This function tokenizes a character vector, splitting it into individual words or other units of text.
kwic(): This function returns keyword-in-context (KWIC) lines for a specific keyword or regular expression pattern.
dfm(): This function creates a document-feature matrix (DFM) from a set of tokens, representing each document as a row and each unique token as a column.

```
> # quanteda
> bookTokens1 <- quanteda::tokens(bookText$content)
> kwic(bookTokens1, pattern = "remarkable")
Keyword-in-context with 2 matches.
  [text45, 6] privations we located the most | remarkable | goldbearing quartz vein
 [text197, 8]           owe my life and the | remarkable | experiences and adventures which

> kwic(bookTokens1, pattern = "independent")
Keyword-in-context with 0 matches.
> dfm1 <- dfm(bookTokens)
> dfm1
Document-feature matrix of: 10 documents, 63 features (87.30% sparse) and 0 docvars.
       features
docs    on the arizona hills i am a very old man
  text1  0   0       0     0 0  0 0    0   0   0
```

Conclusion

We learned the importance of cleaning up text by deleting special characters and stop words through this project, which had a significant impact on the correctness of our DTM and TDM matrices. We also noticed that seemingly small features like punctuation and word length can have a major impact on the outcomes of our analysis. This experience has taught us a lot about data exploration in the context of text and NLP functions.

There are two types of text analysis: syntactic analysis and semantic analysis. Syntactic studies, such as part-of-speech tagging, word frequency distribution, and n-grams, concentrate on the document's grammatical structure rather than its meaning. Semantic analyses, on the other hand, such as document categorization, word clouds, and sentiment analysis, concentrate on the themes and concepts expressed in the document. Word clouds are based on frequency of occurrence, whereas sentiment analysis is based on machine learning methods. In this assignment, we learned how to use the stringi and quanteda packages to perform string manipulations and quantitative analysis on textual data.