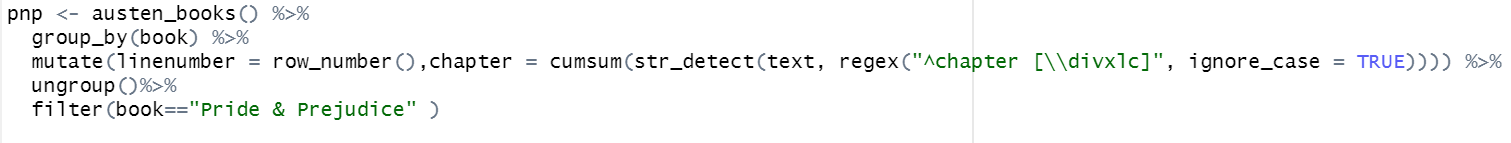
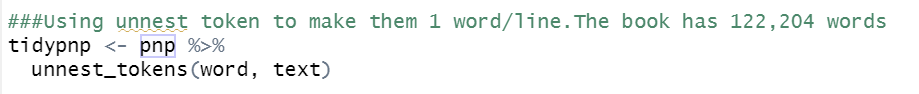
**Comparison cloud of sentiments**

(Please check R script for comments)

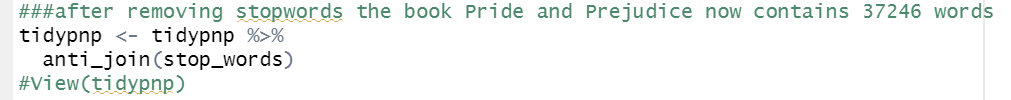
**Step 1: Installing required libraries for tasks ahead.**

**Step 2: Grouping and ungrouping by books, while filtering “Pride & Prejudice”. Using mutate functions we have added 2 more columns which shows the line number and chapter number.**

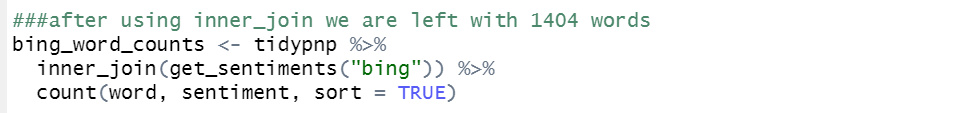


**Step 3: Using Unnest token to make data 1 word per line.** 

**Step 4: Using stop words to remove unwanted words from data.**



**Step 5: Using inner join for joining the sentiment words from Bing lexicon and words in our data.**



**Step 6: Using barcharts from ggplot to find out top 25 and bottom 25 words, where we found the word miss was used too much and could be removed, as ‘miss’ can have many meanings ranging from missing(feeling) to suffix.**

For bar graph containing top 25 words with **HIGHEST** frequency

Text

Description automatically generated Chart

Description automatically generated

For bar graph containing top 25 words with **LOWEST** frequency

Text

Description automatically generated Chart

Description automatically generated

**Step 7: Removing the word miss from data**

Text

Description automatically generated with medium confidence

**Step 8 Forming the comparison cloud**

A picture containing scatter chart

Description automatically generated

A picture containing text, newspaper, battery

Description automatically generated

**The codes above are expanded for better understanding of the process and data. However we can create the comparison cloud with very minimal code as below:**

**###Small code for comparison cloud of sentiments**

pnp <- austen\_books() %>%

group\_by(book) %>%

mutate(linenumber = row\_number()) %>%

ungroup()%>%

filter(book=="Pride & Prejudice" )%>%

unnest\_tokens(word, text)%>%

anti\_join(stop\_words)%>%

inner\_join(get\_sentiments("bing")) %>%

count(word, sentiment, sort = TRUE) %>%

acast(word ~ sentiment, value.var = "n", fill = -0.5) %>%

comparison.cloud(title.colors=c("dark red","green"),colors = c(" black", "blue"),

max.words = 500)

**### actual code for comparison cloud assignment**

###Installing required Libraries

library(janeaustenr)

library(dplyr)

library(stringr)

library(tidytext)

library(reshape2)

library(ggplot2)

library(wordcloud)

###Grouping the sentences by books and filtering the Pride and Prejudice(pnp) book using 'Filter() function'

###Using Mutate function we have added 2 colums for line number and chapter number

pnp <- austen\_books() %>%

group\_by(book) %>%

mutate(linenumber = row\_number(),chapter = cumsum(str\_detect(text, regex("^chapter [\\divxlc]", ignore\_case = TRUE)))) %>% ## line and chapter number added just to understand data better

ungroup()%>%

filter(book=="Pride & Prejudice" )

#View(pnp)

###Using unnest token to make them 1 word/line.The book has 122,204 words

tidypnp <- pnp %>%

unnest\_tokens(word, text)

###after removing stopwords the book Pride and Prejudice now contains 37246 words

tidypnp <- tidypnp %>%

anti\_join(stop\_words)

#View(tidypnp)

###after using inner\_join we are left with 1404 'unique words'

bing\_word\_counts <- tidypnp %>%

inner\_join(get\_sentiments("bing")) %>%

count(word, sentiment, sort = TRUE) ##count function will give frequency of words

#View(bing\_word\_counts)

###By plotting the bar chart we saw that the frequency of word 'miss' is unusually high and can have multiple meaning so we can remove the word miss

bing\_word\_counts %>%

group\_by(sentiment) %>%

top\_n(25) %>%

ungroup() %>%

mutate(word = reorder(word, n)) %>% ## Mutate and reorder is used so that graph bars are sorted

ggplot(aes(word, n, fill = sentiment)) +

geom\_col(show.legend = FALSE) +

facet\_wrap(~sentiment, scales = "free\_y") + # using facet wrap we get free y axis for different sentiments

labs(y = "Contribution to sentiment",

x = NULL) +

coord\_flip()

###Plotting bar chart for "bottom n" words using top\_n() function

bing\_word\_counts %>%

group\_by(sentiment) %>%

top\_n(-25,word) %>%

ungroup() %>%

mutate(word = reorder(word, n)) %>%

ggplot(aes(word, n, fill = sentiment)) +

geom\_col(show.legend = FALSE) +

facet\_wrap(~sentiment, scales = "free\_y") +

labs(y = "Contribution to sentiment",

x = NULL) +

coord\_flip()

class(bing\_word\_counts)

##removing word miss from the data bing\_word\_counts

finaldata <- bing\_word\_counts[-c(1),] # we have removed 1st row which contains 'miss'

#Comparison cloud

finaldata%>%

acast(word ~ sentiment, value.var = "n", fill = -0.5) %>%

comparison.cloud(title.colors=c("dark red","green"),colors = c(" black", "blue"),

max.words = 1500)