**SENTIMENT ANALYSIS**

**Submitted as part of DATA SCIENCE AND BIG DATA ANALYTICS**

**Course Requirement**

**By**

**Muskan Singhal (RA1711008010070)**

**Department of Information and Technology**

**SRM Institute of Science and Technology**

**Kattankulathur, Chennai**



**SRM INSTITUTE OF SCIENCE AND TECHNOLOGY**

**FACULTY OF ENGINEERING AND TECHNOLOGY**

**DEPARTMENT OF INFORMATION TECHNOLOGY**

**ACKNOWLEDEGMENT**

It is a matter of great pleasure and privilege for us to present this report about our Sentiment Analysis on dataset based on the books by Jane Austen. Through this report, we would like to thank our teacher whose consistent support and guidance has been the standing pillar in  
architecture of this report.

Our sincere thanks to **Mr. L. Anand**, Associate Professor for guiding us.  
throughout the work and for teaching and valuable suggestions that helped us to make this project.

Next, we would like to express our sincere gratitude to him to experience a sustained enthusiastic and involved interest from his side.  
We would like to thank him for positive attitude he showed for our work.

**TABLE OF CONTENTS**

|  |  |  |
| --- | --- | --- |
| S.NO. | Description | Page No. |
| 1. | Aim of the project | 5 |
| 2. | What is sentimental analysis | 6 |
| 3. | Intro to RStudio | 7 |
| 4. | Developing sentimental analysis | 8 |
| 5. | Use of inner join | 10 |
| 6. | Summary and findings | 17 |

**AIM OF THE PROJECT**

In sentiment analysis, task is commonly defined as classifying a given text into one of two classes: objective or subjective. This problem can sometimes be more difficult than polarity classification. The subjectivity of words and phrases may depend on their context and an objective document may contain subjective sentences. Sentiment analysis ascertains the mood on social media pertaining to products, service activities, campaigns and companies. Where opinions are predominantly negative, the company can analyse the reasons and react.

The aim of this project is to build a sentiment analysis model which will allow us to categorize words based on their sentiments, that is whether they are positive, negative and also the magnitude of it.

We will be using multiple packages and functions, to obtain analysis for the given project, like:

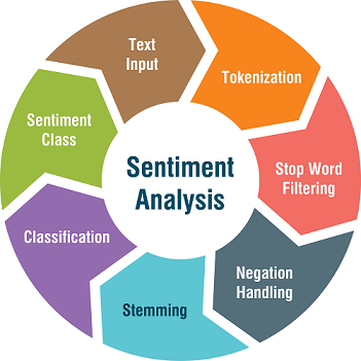
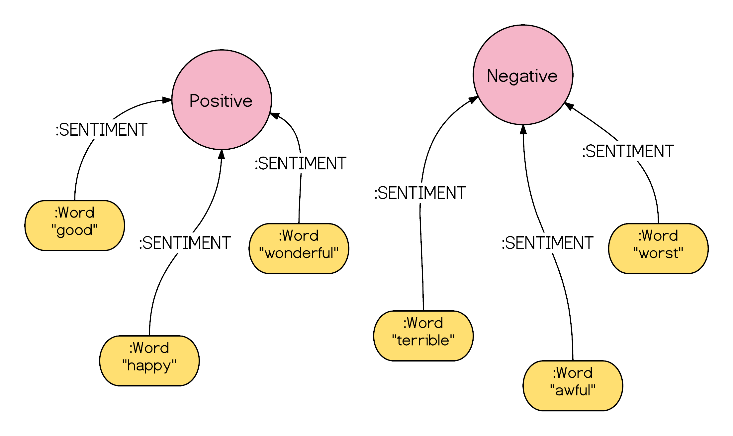
* ggplot2
* tidytext
* janeaustenr
* tidyr
* plyr
* dplyr
* stringr

These packages will provide us access to different functions to perform different operations.

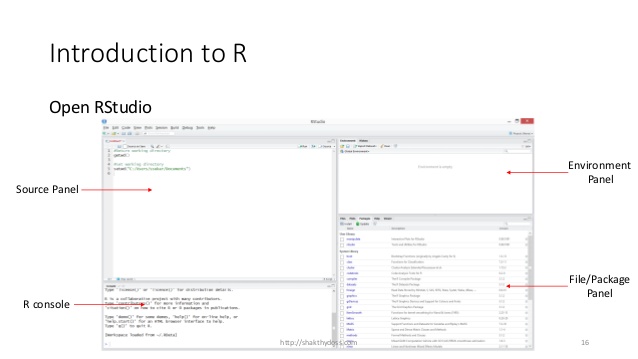
**WHAT IS SENTIMENT ANALYSIS?**

Sentiment Analysis is a process of extracting opinions that have different polarities. By polarities, we mean positive, negative or neutral. It is also known as opinion mining and polarity detection. With the help of sentiment analysis, you can find out the nature of opinion that is reflected in documents, websites, social media feed, etc. Sentiment Analysis is a type of classification where the data is classified into different classes. These classes can be binary in nature (positive or negative) or, they can have multiple classes (happy, sad, angry, etc.).

In marketing, sentiment analysis predominantly falls under the realm of social media monitoring. Here the term “social media” doesn’t refer only to Facebook, Twitter and the others, but also to YouTube, product and service reviews in shops and portals, as well as forum posts. Today, opinions are very quickly formed and spread through these channels. So those who know how their own products are being talked about in comments and posts can react accordingly. Sentiment analysis isn’t limited to just texts though. Videos, images and even podcasts are closely examined as well.

**R AND RSTUDIO**

****

The panel in the upper right contains your workspace as well as a history of the commands that you’ve previously entered.

Any plots that you generate will show up in the panel in the lower right corner. The panel on the left is where the action happens. It’s called the console. Everytime you launch RStudio, it will have the same text at the top of the console telling you the version of R that you’re running.

Below that information is the prompt. As its name suggests, this prompt is really a request, a request for a command.

We will carry out sentiment analysis with R in this project. The dataset that we will use will be provided by the R package ‘janeaustenR’.

In order to build our project on sentiment analysis, we will make use of the tidytext package that comprises of sentiment lexicons that are present in the dataset of ‘sentiments’.

**DEVELOPING OUR SENTIMENT ANALYSIS MODEL IN R**

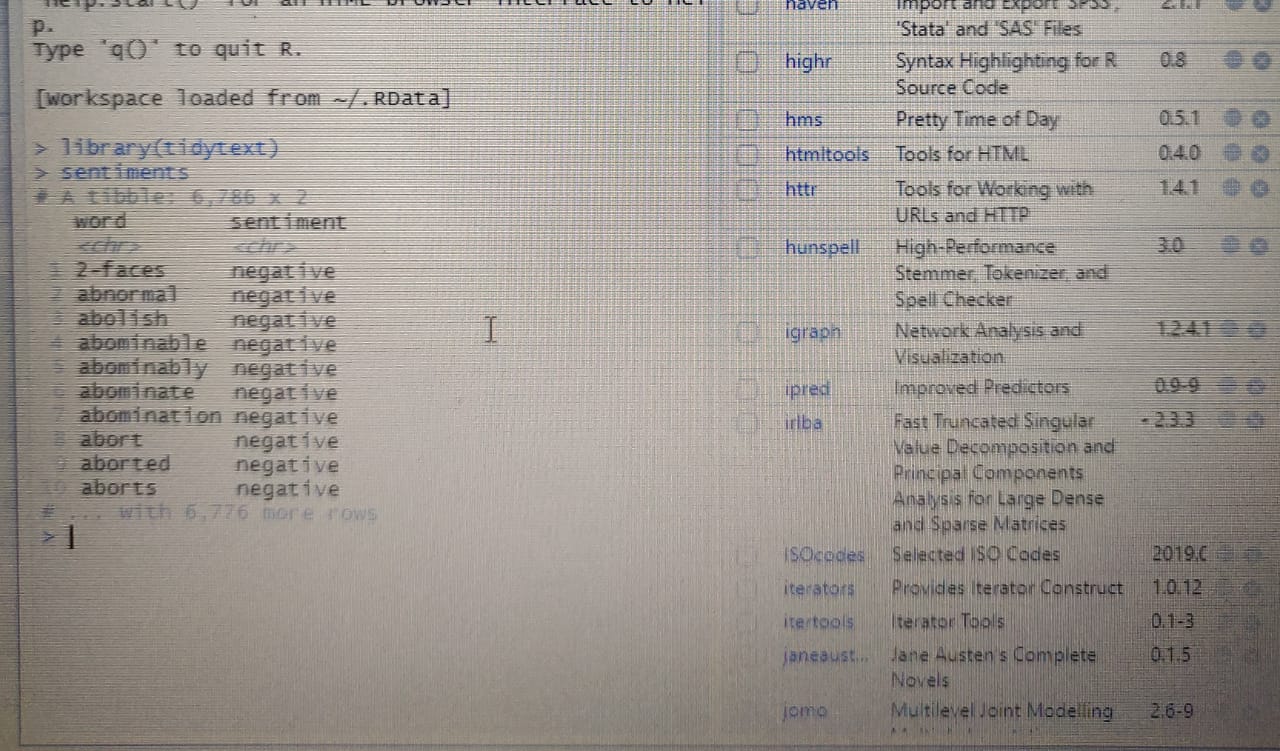
We will carry out the sentiment analysis with R in this project. The dataset that we will use will be provided by the R package ‘janeaustenR’.

In order to build our project on sentiment analysis, we will make use of the tidytext package that comprises of sentiment lexicons that are present in the dataset of ‘sentiments’.

**Syntax:**

**library(tidytext)**

**sentiments**



We will make use of three general purpose lexicons like –

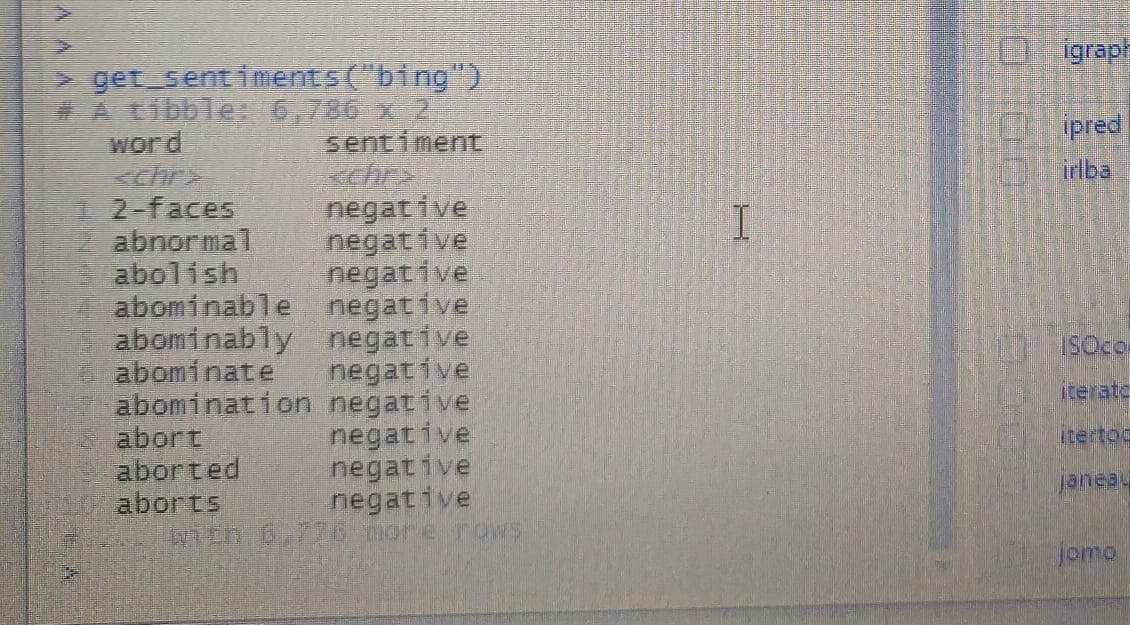
* AFINN
* bing
* loughran

These three lexicons make use of the unigrams. Unigrams are a type of n-gram model that consists of a sequence of 1 item, that is, a word collected from a given textual data. In the AFINN lexicon model scores the words in a range from -5 to 5. The increase in negativity corresponds the negative sentiment whereas an increase in positivity corresponds the positive one. The bing lexicon model on the other hand, classifies the sentiment into a binary category of negative or positive. And finally, the loughran model that performs analysis of the shareholder’s reports. In this project, we will make use of the bing lexicons to extract the sentiments out of our data.

We can retrieve these lexicons using the get\_sentiments() function. We can implement this as follows –

syntax:

**get\_sentiments(“bing”)**



**PERFORMING SENTIMENTAL ANALYSIS USING INNER JOIN**

In this step, we will import our libraries ‘janeaustenr’, ‘stringr’ as well as ‘tidytext’. The janeaustenr package will provide us with the textual data in the form of books authored by the novelist [Jane Austen](https://en.wikipedia.org/wiki/Jane_Austen). Tidytext will allow us to perform efficient text analysis on our data. We will convert the text of our books into a tidy format using unnest\_tokens() function.

**syntax:**

**library(janeaustenr)**

**library(stringr)**

**library(tidytext)**

**tidy\_data <- austen\_books() %>%**

**group\_by(book) %>%**

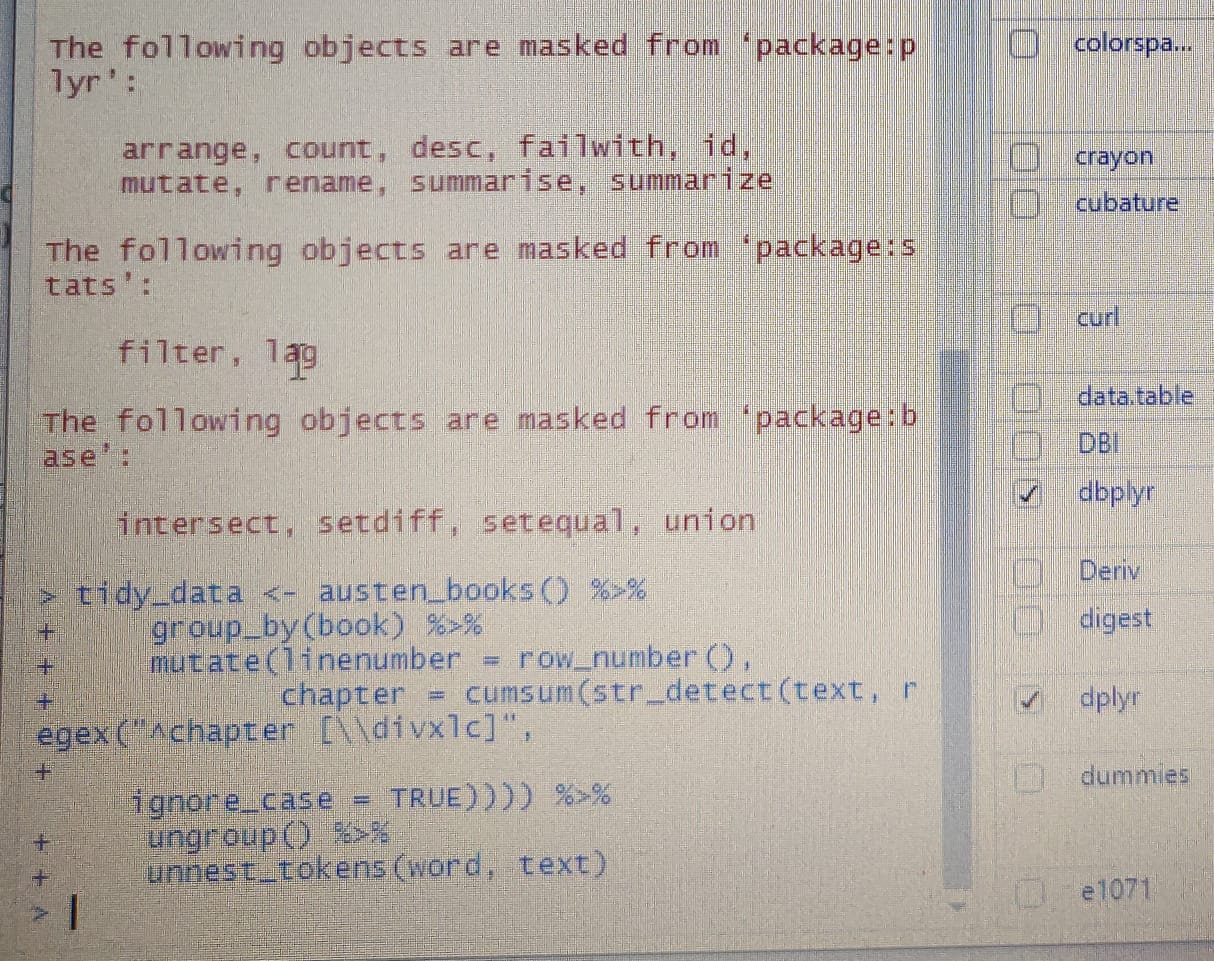
**mutate(linenumber = row\_number(),**

**chapter = cumsum(str\_detect(text, regex("^chapter [\\divxlc]",**

**ignore\_case = TRUE)))) %>%**

**ungroup() %>%**

**unnest\_tokens(word, text)**



We have performed the tidy operation on our text such that each row contains a single word. We will now make use of the “bing” lexicon to and implement filter() over the words that correspond to joy. We will use the book Sense and Sensibility and derive its words to implement out sentiment analysis model.

**Syntax to perfom it is :**

**positive\_senti <- get\_sentiments("bing") %>%**

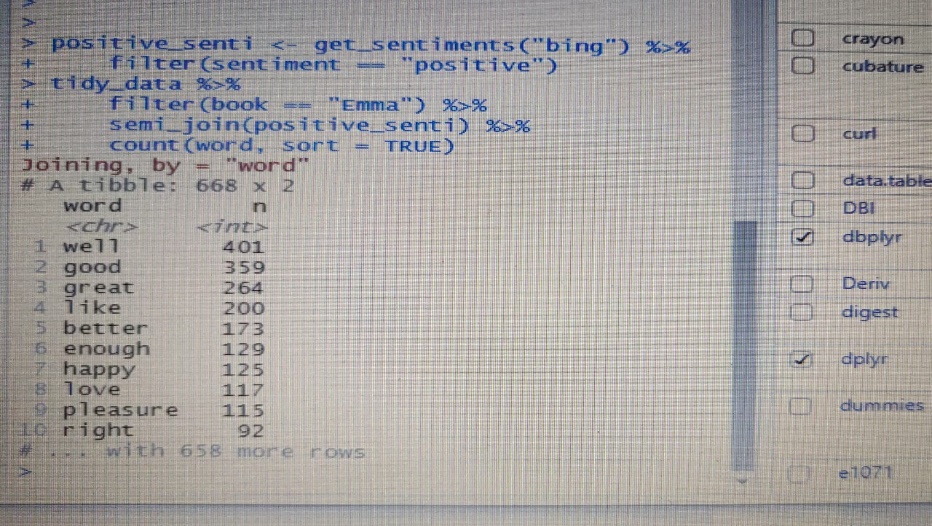
**filter(sentiment == "positive")**

**tidy\_data %>%**

**filter(book == "Emma") %>%**

**semi\_join(positive\_senti) %>%**

**count(word, sort = TRUE)**



From our above result, we observe many positive words like “good”, “happy”, “love” etc. In the next step, we will use spread() function to segregate our data into separate columns of positive and negative sentiments. We will then use the mutate() function to calculate the total sentiment, that is, the difference between positive and negative sentiment.

**Syntax :**

**library(tidyr)**

**bing <- get\_sentiments("bing")**

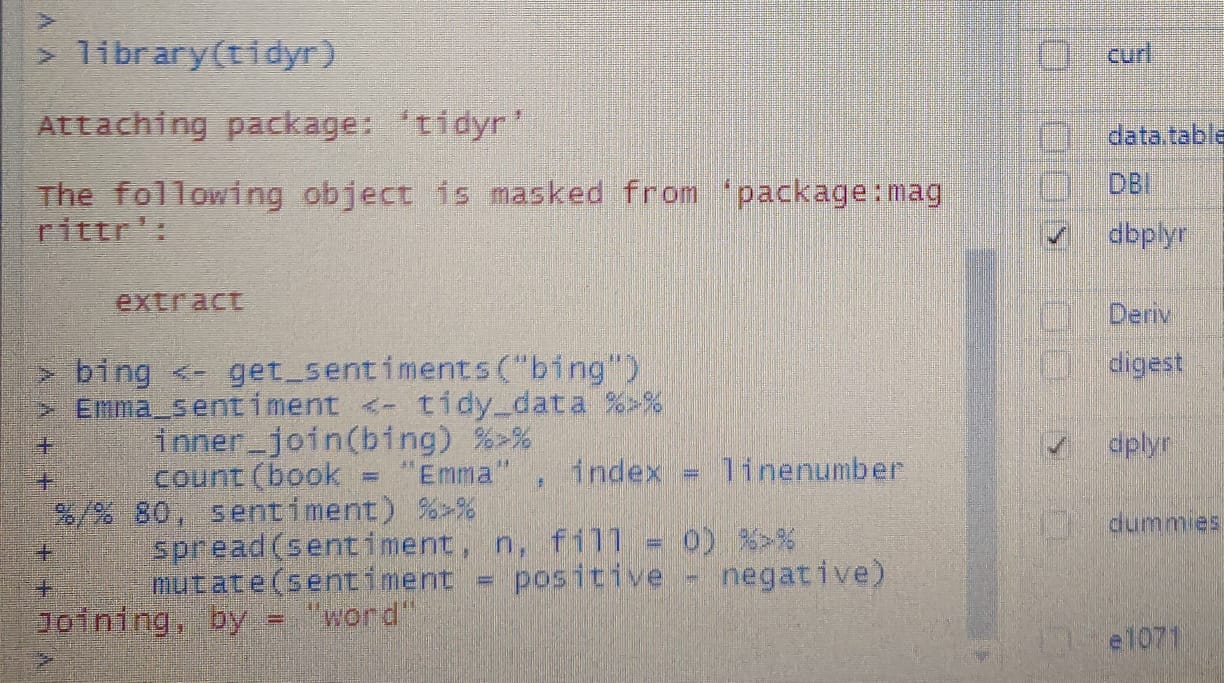
**Emma\_sentiment <- tidy\_data %>%**

**inner\_join(bing) %>%**

**count(book = "Emma" , index = linenumber %/% 80, sentiment) %>%**

**spread(sentiment, n, fill = 0) %>%**

**mutate(sentiment = positive - negative)**



In the next step, we will visualize the words present in the book “Emma” based on their corrosponding positive and negative scores.

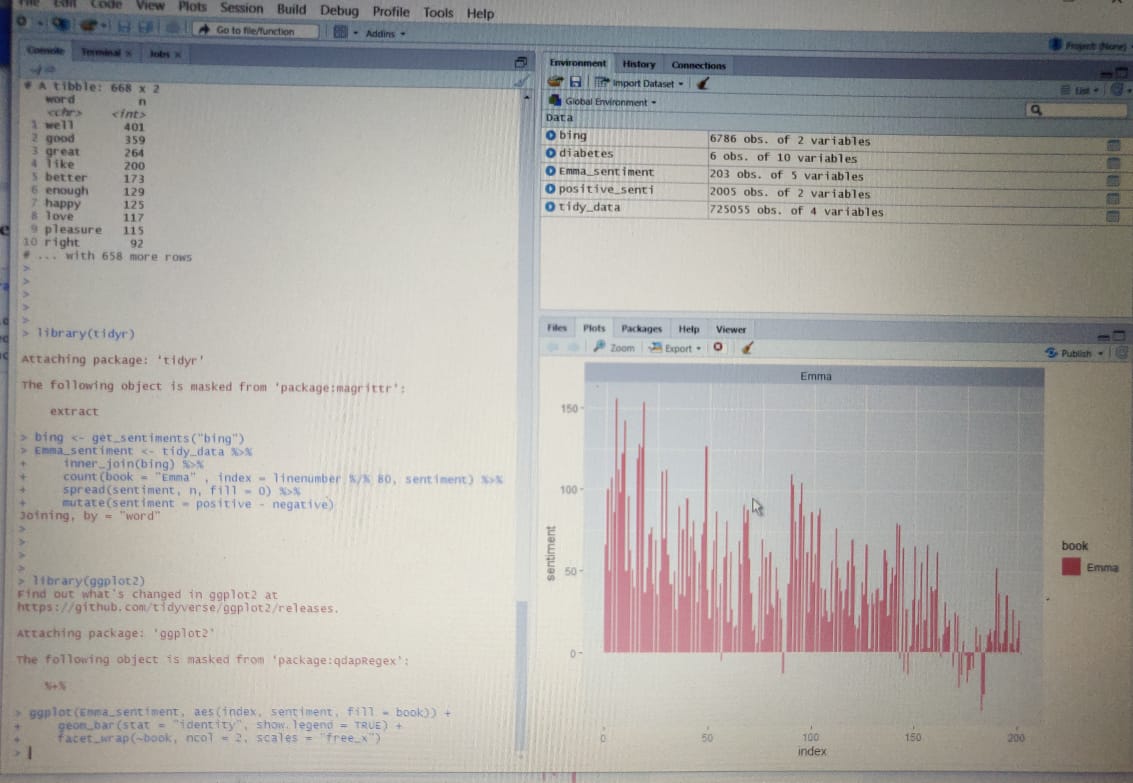
**syntax:**

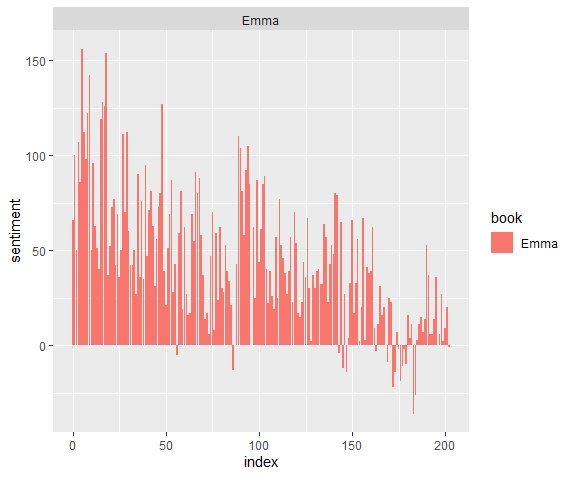
**library(ggplot2)**

**ggplot(Emma\_sentiment, aes(index, sentiment, fill = book)) +**

**geom\_bar(stat = "identity", show.legend = TRUE) +**

**facet\_wrap(~book, ncol = 2, scales = "free\_x")**





Let us now proceed towards counting the most common positive and negative words that are present in the novel.

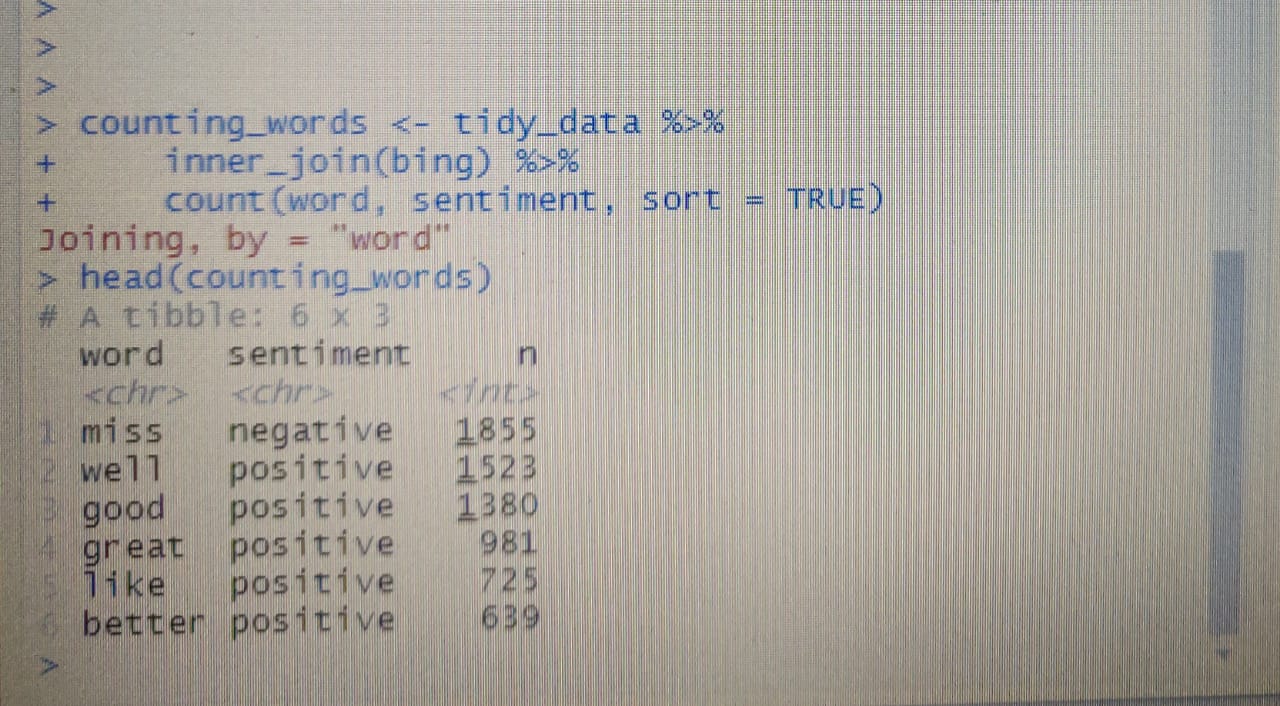
**syntax :**

**counting\_words <- tidy\_data %>%**

**inner\_join(bing) %>%**

**count(word, sentiment, sort = TRUE)**

**head(counting\_words)**



In the next step, we will perform visualization of our sentiment score. We will plot the scores along the axis that is labeled with both positive as well as negative words. We will use ggplot() function to visualize our data based on their scores.

**syntax :**

**counting\_words %>%**

**filter(n > 150) %>%**

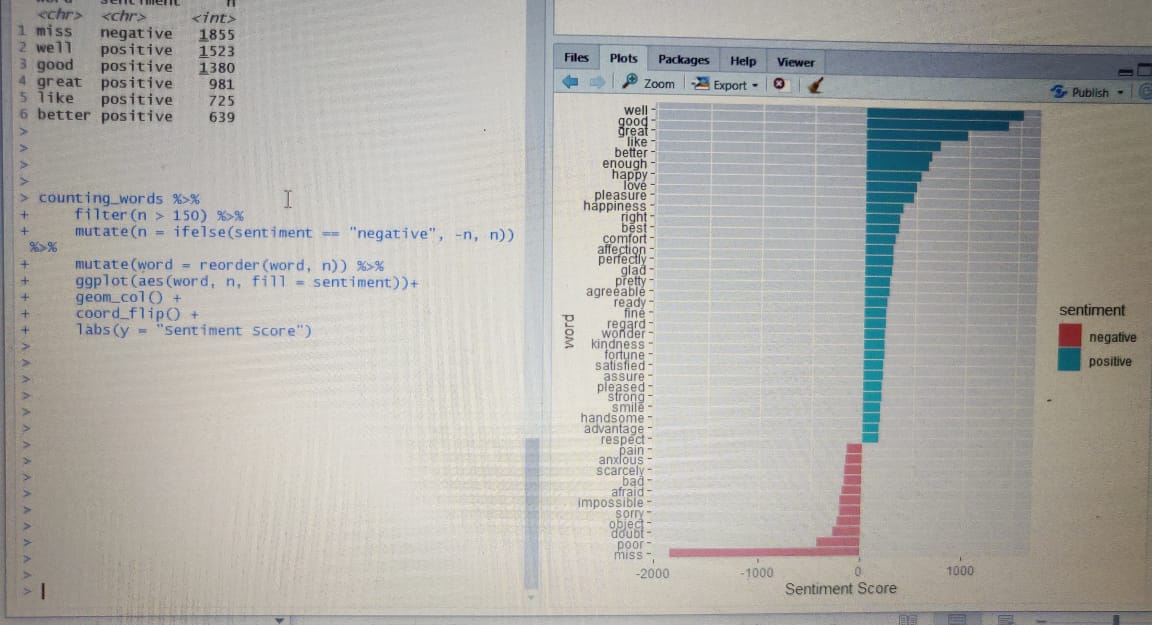
**mutate(n = ifelse(sentiment == "negative", -n, n)) %>%**

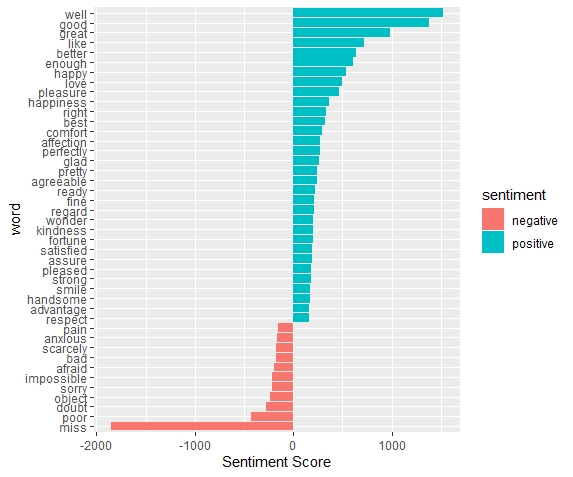
**mutate(word = reorder(word, n)) %>%**

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

**geom\_col() +**

**coord\_flip() + labs(y = "Sentiment Score")**





In the final visualization, let us create a wordcloud that will delineate the most recurring positive and negative words. In particular, we will use the comparision.cloud() function to plot both negative and positive words in a single wordcloud as follows:

**syntax :**

**library(reshape2)**

**library(wordcloud)**

**tidy\_data %>%**

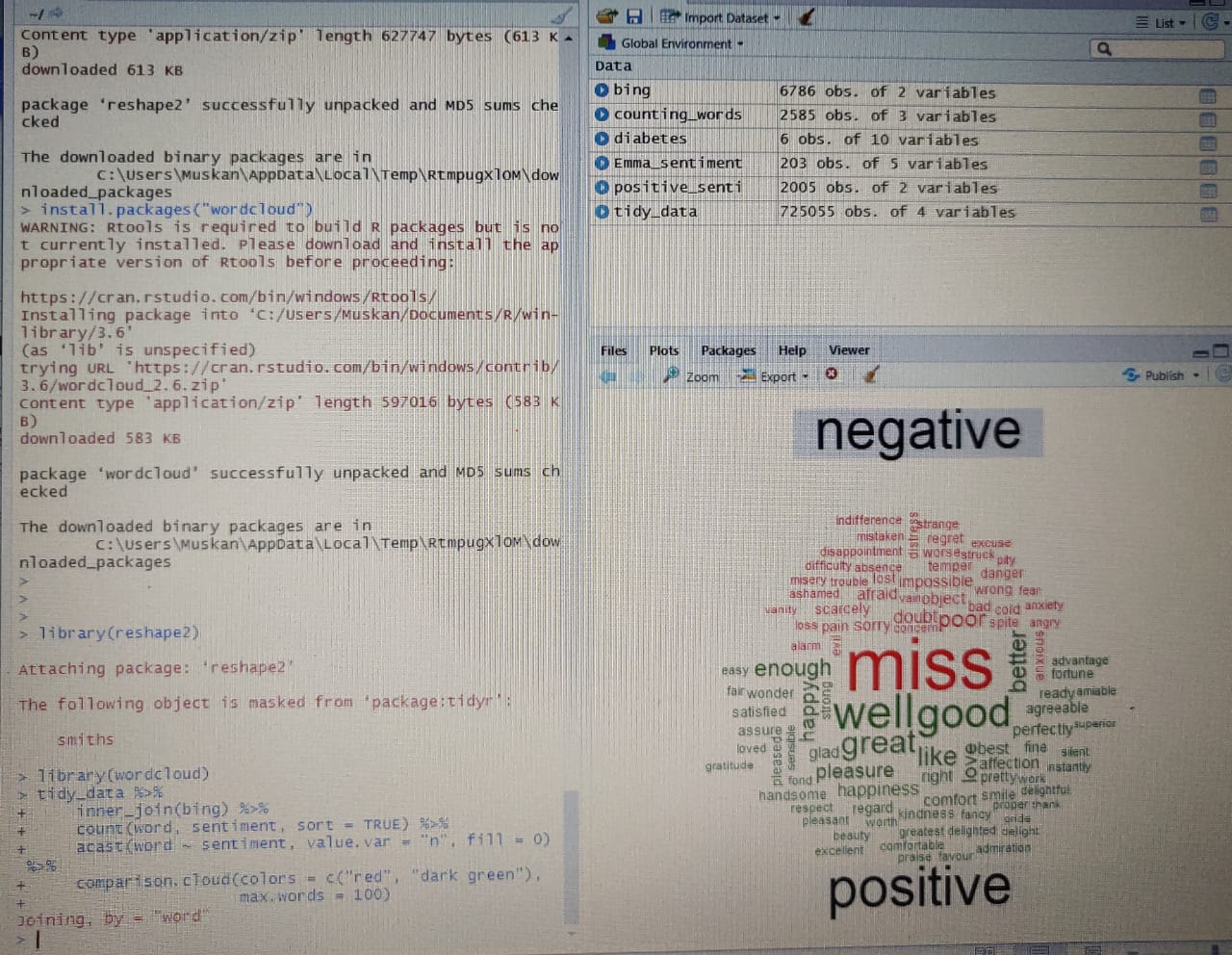
**inner\_join(bing) %>%**

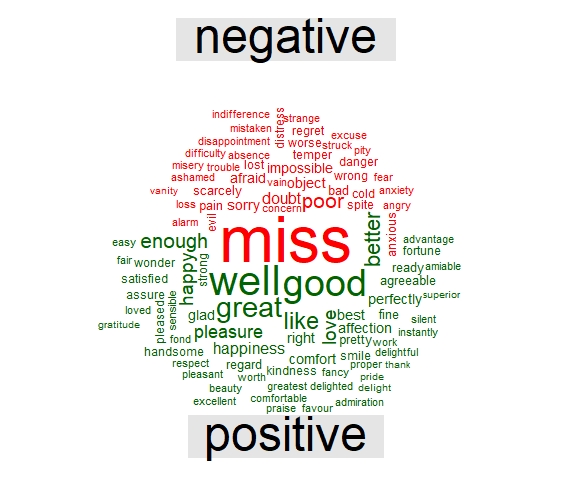
**count(word, sentiment, sort = TRUE) %>%**

**acast(word ~ sentiment, value.var = "n", fill = 0) %>%**

**comparison.cloud(colors = c("red", "dark green"),**

**max.words = 100)**





This word cloud will enable us to efficiently visualize the negative as well as positive groups of data. Therefore, we are now able to see the different groups of data based on their corresponding sentiments.

**SUMMARY AND FINDINGS**

In this project we learnt about the concept of sentiment analysis and implemented it over the dataset of Jane Austen’s books. We were able to delineate it through various visualizations after we performed data wrangling on our data. We used a lexical analyser – ‘Bing’ in this instance of our project. We were able to explore and gain knowledge of multiple packages like tidyr, tidytext, ggplot2, dplyr, plyr, which are a very important part of this project. Furthermore, we also represented the sentiment score through a plot and also made a visual report of wordcloud effectively visualizing both negative and positive sides of data.