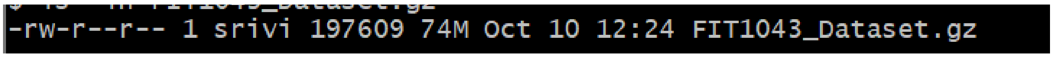
# A1.

1. 74MB

ls -lh FIT1043_Dataset.gz

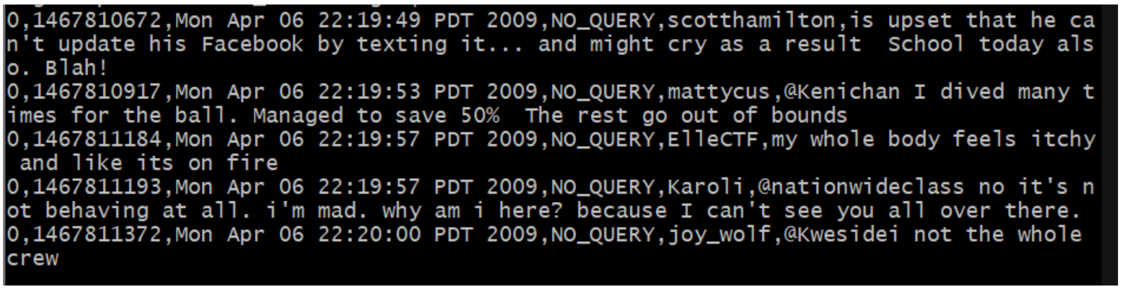
The command `ls -lh FIT1043_Dataset.gz` was used to determine the size of the file. The `ls` command lists the details of files, and the `-lh` option provides a long listing format and displays the size in a human-readable format, showing MB instead of just bytes. This command showed that the file is 74MB, helping assess the dataset's storage requirements.



```
-rw-r--r-- 1 srivi 197609 74M Oct 10 12:24 FIT1043_Dataset.gz
```

2. $ gunzip -c FIT1043_Dataset.gz |head -5

To identify the delimiter used in the dataset, the command `gunzip -c FIT1043_Dataset.gz | head -5` was executed. Here, `gunzip -c` decompresses the file and streams its contents to the terminal without permanently unzipping it. The `head -5` command then displays the first five lines of the file. From this output, it became clear that the dataset columns were separated by commas, indicating it is in a CSV format.

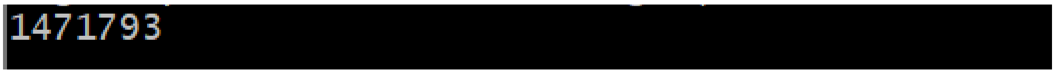


```
0,1467810672,Mon Apr 06 22:19:49 PDT 2009,NO_QUERY,scotthamilton,is upset that he ca
n't update his Facebook by texting it... and might cry as a result  School today als
o. Blah!
0,1467810917,Mon Apr 06 22:19:53 PDT 2009,NO_QUERY,mattycus,@Kenichan I dived many t
imes for the ball. Managed to save 50%  The rest go out of bounds
0,1467811184,Mon Apr 06 22:19:57 PDT 2009,NO_QUERY,ElleCTF,my whole body feels itchy
 and like its on fire
0,1467811193,Mon Apr 06 22:19:57 PDT 2009,NO_QUERY,Karoli,@nationwideclass no it's n
ot behaving at all. i'm mad. why am i here? because I can't see you all over there.
0,1467811372,Mon Apr 06 22:20:00 PDT 2009,NO_QUERY,joy_wolf,@Kwesidei not the whole
crew
```

3. 1471793 lines

$ gunzip -c FIT1043_Dataset.gz |wc -l

To count the number of lines in the dataset, the command `gunzip -c FIT1043_Dataset.gz | wc -l` was used. The `gunzip -c` decompresses the file, and the `wc -l` command counts the number of lines in the output. This combination showed that the dataset contains 1,471,793 lines, giving an idea of the dataset's size and the number of data entries available for processing.



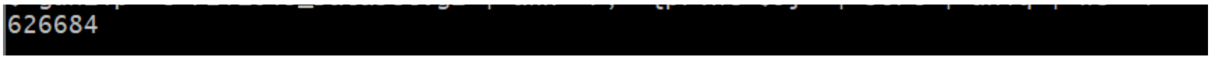
```
1471793
```

# A2.

1. 626684

$ gunzip -c FIT1043_Dataset.gz | awk -F, '{print $5}' | sort | uniq | wc -l

To calculate the number of unique users in the dataset, the command `gunzip -c FIT1043_Dataset.gz | awk -F, '{print $5}' | sort | uniq | wc -l` was employed. The `gunzip -c` decompresses the file and passes it to `awk`, which uses the `-F,` option to specify a comma as the field separator, then extracts the 5th column- which is the username column. The `sort` command organises the user IDs, and `uniq` eliminates duplicates. Finally, `wc -l` counts the unique user IDs, providing the number of distinct users.

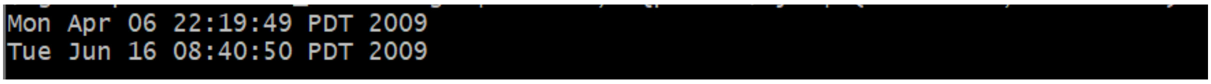


```
626684
```

2. Monday, April 06, 22:19:49, 2009 to Tuesday, June 16, 08:40:50, 2009

$ gunzip -c FIT1043_Dataset.gz | awk -F, '{print $3}' | (head -n 1; tail -n 1)

To determine the date range of the dataset, the command `gunzip -c FIT1043_Dataset.gz | awk -F, '{print $3}' | (head -n 1; tail -n 1)` was used. The `gunzip -c` decompresses the file, and `awk -F, '{print $3}'` extracts the third column, which contains the dates. The `head -n 1` outputs the first line, and `tail -n 1` outputs the last line. This provides the earliest and latest dates in the dataset, which helps establish the chronological range of the data.

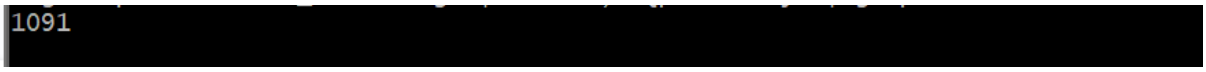


```
Mon Apr 06 22:19:49 PDT 2009
Tue Jun 16 08:40:50 PDT 2009
```

3a. 1091

gunzip -c FIT1043_Dataset.gz | awk -F, '{print $6}' | grep -i -c " france "

To find how many tweets mentioned the word "France" in any case combination, the command `gunzip -c FIT1043_Dataset.gz | awk -F, '{print $6}' | grep -i -c " france "` was used. Here, `awk -F, '{print $6}'` extracts the sixth column, which contains the tweet text. The `grep -i` command searches for the word "france" in a case-insensitive manner, while the `-c` option counts the matches. The inclusion of spaces around "france" ensures that only standalone occurrences are matched, excluding words like "Francesca." This command revealed that there are 1,091 tweets mentioning "France". However, this means that any punctuation attached to France will not be counted. My tutor had confirmed that this was an appropriate input.

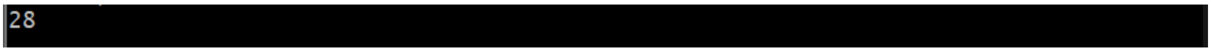


```
1091
```

3b. 28

$ gunzip -c FIT1043_Dataset.gz | grep -i " france " | grep -v -e " france " -e " France " | wc -l

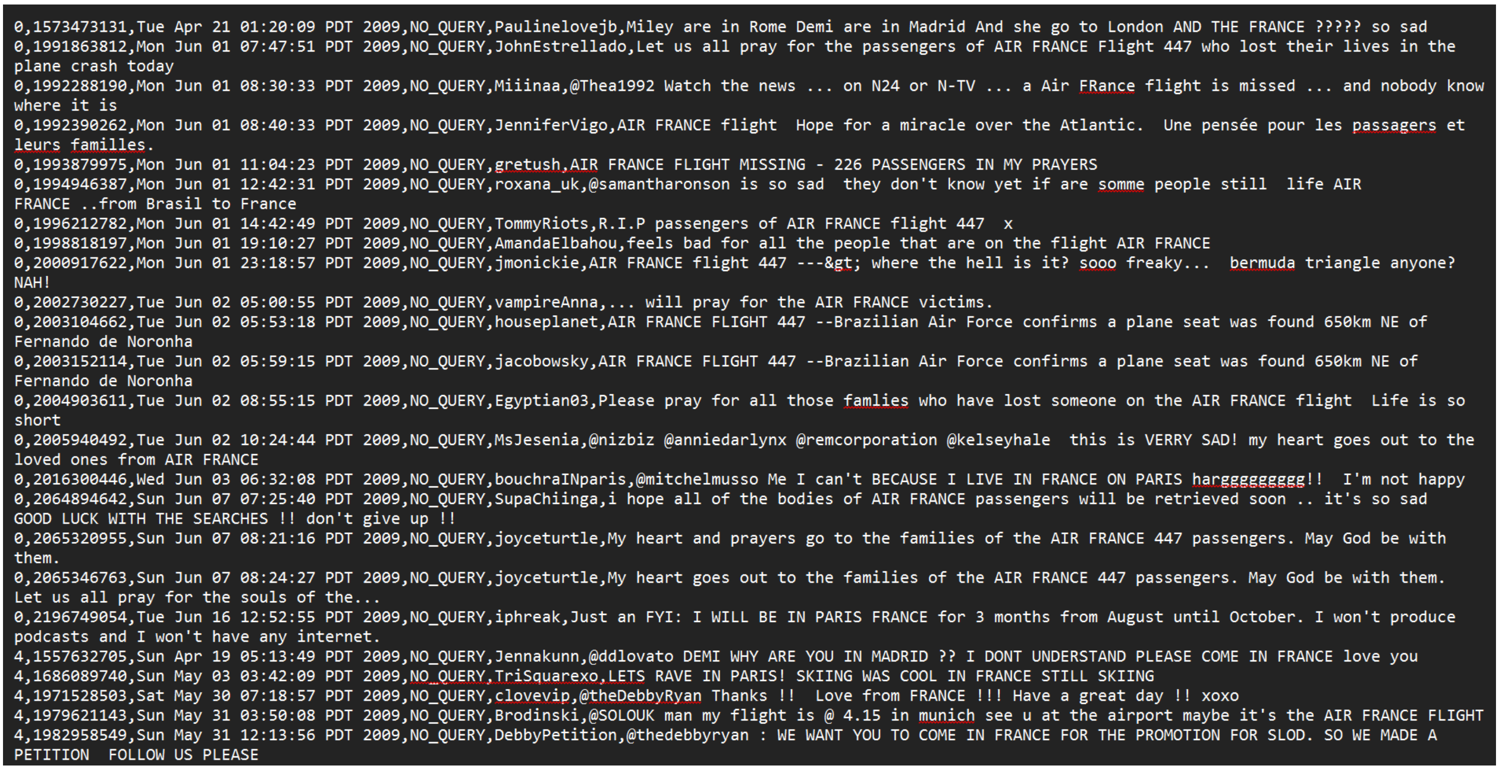
To find variations in the spelling of "France" (like FRance, francE), the command `gunzip -c FIT1043_Dataset.gz | grep -i " france " | grep -v -e " france " -e " France " | wc -l` was used. The `grep -i " france "` first identifies all case-insensitive matches of "france," then `grep -v -e " france " -e " France "` filters out tweets that contain the regular spellings "france" or "France." The `wc -l` counts the remaining tweets, revealing that 28 tweets use an unconventional casing of "France." There is a space before and after the word France when searching for these occurrences again.



```
28
```

c.
gunzip -c FIT1043_Dataset.gz | grep -i " france " | grep -v -e " france " -e " France " > myText.txt

Finally, to output these lines to a text file, the command `gunzip -c FIT1043_Dataset.gz | grep -i " france " | grep -v -e " france " -e " France " > myText.txt` was used. This command saves all lines containing non-standard versions of "France" to the file `myText.txt` for further analysis.



```
0,1573473131,Tue Apr 21 01:20:09 PDT 2009,NO_QUERY,Paulinelovejb,Miley are in Rome Demi are in Madrid And she go to London AND THE FRANCE ????? so sad
0,1991863812,Mon Jun 01 07:47:51 PDT 2009,NO_QUERY,JohnEstrellado,Let us all pray for the passengers of AIR FRANCE Flight 447 who lost their lives in the
plane crash today
0,1992288190,Mon Jun 01 08:30:33 PDT 2009,NO_QUERY,Miiinaa,@Thea1992 Watch the news ... on N24 or N-TV ... a Air FRance flight is missed ... and nobody know
where it is
0,1992390262,Mon Jun 01 08:40:33 PDT 2009,NO_QUERY,JenniferVigo,AIR FRANCE flight  Hope for a miracle over the Atlantic.  Une pensée pour les passagers et
leurs familles.
0,1993879975,Mon Jun 01 11:04:23 PDT 2009,NO_QUERY,gretush,AIR FRANCE FLIGHT MISSING - 226 PASSENGERS IN MY PRAYERS
0,1994946387,Mon Jun 01 12:42:31 PDT 2009,NO_QUERY,roxana_uk,@samantharonson is so sad  they don't know yet if are somme people still  life AIR
FRANCE ..from Brasil to France
0,1996212782,Mon Jun 01 14:42:49 PDT 2009,NO_QUERY,TommyRiots,R.I.P passengers of AIR FRANCE flight 447  x
0,1998818197,Mon Jun 01 19:10:27 PDT 2009,NO_QUERY,AmandaElbahou,feels bad for all the people that are on the flight AIR FRANCE
0,2000917622,Mon Jun 01 23:18:57 PDT 2009,NO_QUERY,jmonickie,AIR FRANCE flight 447 ---&gt; where the hell is it? sooo freaky...  bermuda triangle anyone?
NAH!
0,2002730227,Tue Jun 02 05:00:55 PDT 2009,NO_QUERY,vampireAnna,... will pray for the AIR FRANCE victims.
0,2003104662,Tue Jun 02 05:53:18 PDT 2009,NO_QUERY,houseplanet,AIR FRANCE FLIGHT 447 --Brazilian Air Force confirms a plane seat was found 650km NE of
Fernando de Noronha
0,2003152114,Tue Jun 02 05:59:15 PDT 2009,NO_QUERY,jacobowsky,AIR FRANCE FLIGHT 447 --Brazilian Air Force confirms a plane seat was found 650km NE of
Fernando de Noronha
0,2004903611,Tue Jun 02 08:55:15 PDT 2009,NO_QUERY,Egyptian03,Please pray for all those famlies who have lost someone on the AIR FRANCE flight  Life is so
short
0,2005940492,Tue Jun 02 10:24:44 PDT 2009,NO_QUERY,MsJesenia,@nizbiz @anniedarlynx @remcorporation @kelseyhale  this is VERRY SAD! my heart goes out to the
loved ones from AIR FRANCE
0,2016300446,Wed Jun 03 06:32:08 PDT 2009,NO_QUERY,bouchraINparis,@mitchelmusso Me I can't BECAUSE I LIVE IN FRANCE ON PARIS harggggggggg!!  I'm not happy
0,2064894642,Sun Jun 07 07:25:40 PDT 2009,NO_QUERY,SupaChiinga,i hope all of the bodies of AIR FRANCE passengers will be retrieved soon .. it's so sad
GOOD LUCK WITH THE SEARCHES !! don't give up !!
0,2065320955,Sun Jun 07 08:21:16 PDT 2009,NO_QUERY,joyceturtle,My heart and prayers go to the families of the AIR FRANCE 447 passengers. May God be with
them.
0,2065346763,Sun Jun 07 08:24:27 PDT 2009,NO_QUERY,joyceturtle,My heart goes out to the families of the AIR FRANCE 447 passengers. May God be with them.
Let us all pray for the souls of the...
0,2196749054,Tue Jun 16 12:52:55 PDT 2009,NO_QUERY,iphreak,Just an FYI: I WILL BE IN PARIS FRANCE for 3 months from August until October. I won't produce
podcasts and I won't have any internet.
4,1557632705,Sun Apr 19 05:13:49 PDT 2009,NO_QUERY,Jennakunn,@ddlovato DEMI WHY ARE YOU IN MADRID ?? I DONT UNDERSTAND PLEASE COME IN FRANCE love you
4,1686089740,Sun May 03 03:42:09 PDT 2009,NO_QUERY,TriSquarexo,LETS RAVE IN PARIS! SKIING WAS COOL IN FRANCE STILL SKIING
4,1971528503,Sat May 30 07:18:57 PDT 2009,NO_QUERY,clovevip,@theDebbyRyan Thanks !!  Love from FRANCE !!! Have a great day !! xoxo
4,1979621143,Sun May 31 03:50:08 PDT 2009,NO_QUERY,Brodinski,@SOLOUK man my flight is @ 4.15 in munich see u at the airport maybe it's the AIR FRANCE FLIGHT
4,1982958549,Sun May 31 12:13:56 PDT 2009,NO_QUERY,DebbyPetition,@thedebbyryan : WE WANT YOU TO COME IN FRANCE FOR THE PROMOTION FOR SLOD. SO WE MADE A
PETITION  FOLLOW US PLEASE
```

# A3.

1.

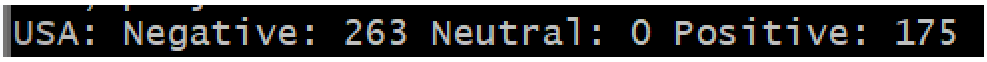
```
$ zgrep -i " USA " FIT1043_Dataset.gz | awk -F, '{neg+=($1=="0"); neu+=($1=="2"); pos+=($1=="4")} END {print "USA:", "Negative:", neg, "Neutral:", neu, "Positive:", pos}'
```

```
$ zgrep -i " Canada " FIT1043_Dataset.gz | awk -F, '{neg+=($1=="0"); neu+=($1=="2"); pos+=($1=="4")} END {print "Canada:", "Negative:", neg, "Neutral:", neu, "Positive:", pos}'
```

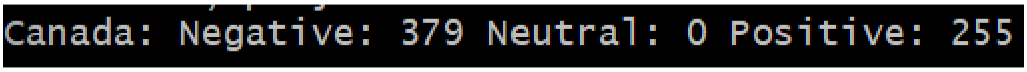
USA
Negative 263
Neutral 0
Positive 175

Canada
Negative 379
Neutral 0
Positive 255

The command starts with `zgrep -i " USA " FIT1043_Dataset.gz`, where `zgrep` searches within the compressed file `FIT1043_Dataset.gz` for lines containing "USA". The `-i` option makes the search case-insensitive, and the spaces around "USA" ensure it is treated as a standalone word, avoiding matches within other words (like "RussianUSA"). The result of this search is piped (`|`) to `awk`, which processes each line. `awk -F,` tells `awk` to use a comma as the field separator. In the `{neg+=($1=="0"); neu+=($1=="2"); pos+=($1=="4")}` part, `awk` looks at the first field in each line, where it checks if the value is "0" (negative sentiment), "2" (neutral), or "4" (positive), incrementing the corresponding counters (`neg`, `neu`, or `pos`). Finally, the `END` block is executed after processing all lines, where it prints the total sentiment counts for the USA: "Negative", "Neutral", and "Positive". A similar command is used for "Canada", with spaces around "Canada" to ensure it's treated as a standalone word and not part of other words.



```
USA: Negative: 263 Neutral: 0 Positive: 175
```

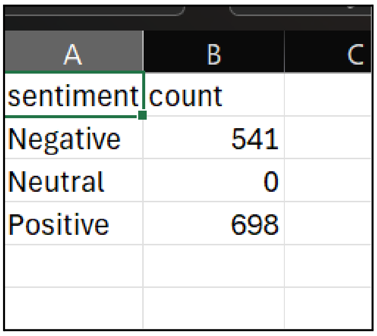


```
Canada: Negative: 379 Neutral: 0 Positive: 255
```

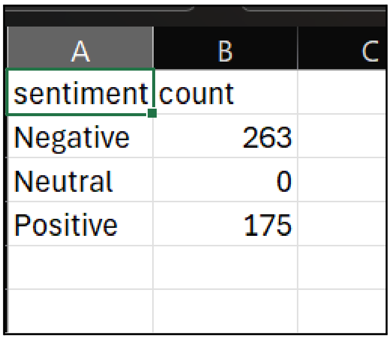
2.

```
echo -e "sentiment,count\nNegative,541\nNeutral,0\nPositive,698" > sentiment-canada.csv
echo -e "sentiment,count\nNegative,263\nNeutral,0\nPositive,175" > sentiment-USA.csv
```

This code creates two CSV files that store sentiment data for Canada and the USA. The first line of each command defines the column headers, "sentiment" and "count," which categorise the data. The next lines provide the sentiment categories (Negative, Neutral, and Positive) and their corresponding counts for each country. These files are saved as sentiment-canada.csv and sentiment-USA.csv, respectively, allowing the data to be easily imported into spreadsheet programs.



| A | B | C |
|---|---|---|
| sentiment | count | |
| Negative | 541 | |
| Neutral | 0 | |
| Positive | 698 | |

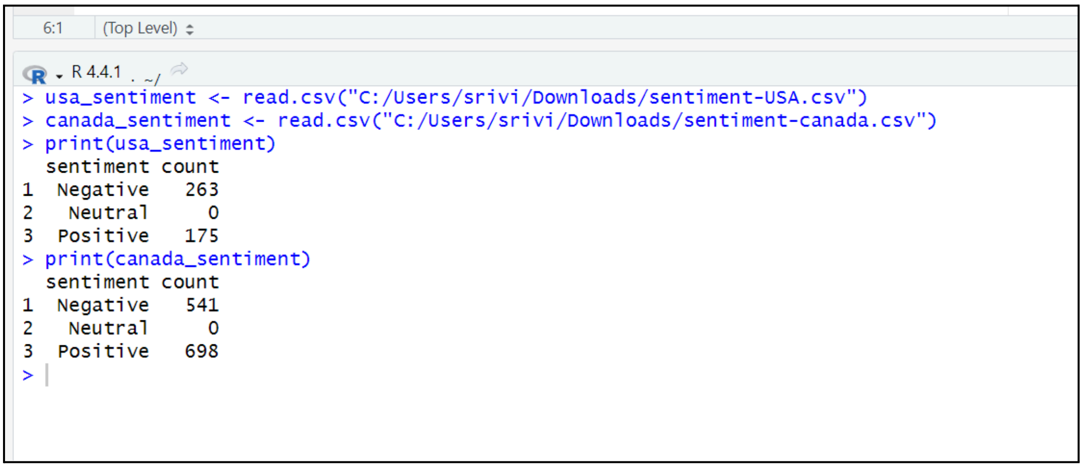


| A | B | C |
|---|---|---|
| sentiment | count | |
| Negative | 263 | |
| Neutral | 0 | |
| Positive | 175 | |

3.
usa_sentiment <- read.csv("C:/Users/srivi/Downloads/sentiment-USA.csv")
canada_sentiment <- read.csv("C:/Users/srivi/Downloads/sentiment-canada.csv")

print(usa_sentiment)
print(canada_sentiment)

In R, the sentiment data from both CSV files was read using the `read.csv()` function, like `usa_sentiment <- read.csv("sentiment-USA.csv")`. This imports the CSV files into R, making them available for data manipulation and visualisation. Then the print statements output the data so we can see it.



```
> usa_sentiment <- read.csv("C:/Users/srivi/Downloads/sentiment-USA.csv")
> canada_sentiment <- read.csv("C:/Users/srivi/Downloads/sentiment-canada.csv")
> print(usa_sentiment)
  sentiment count
1  Negative   263
2   Neutral     0
3  Positive   175
> print(canada_sentiment)
  sentiment count
1  Negative   541
2   Neutral     0
3  Positive   698
> 
```

4.
library(ggplot2)

usa_sentiment <- read.csv("C:/Users/srivi/Downloads/sentiment-USA.csv")
canada_sentiment <- read.csv("C:/Users/srivi/Downloads/sentiment-canada.csv")

usa_sentiment$Country <- "USA"
canada_sentiment$Country <- "Canada"

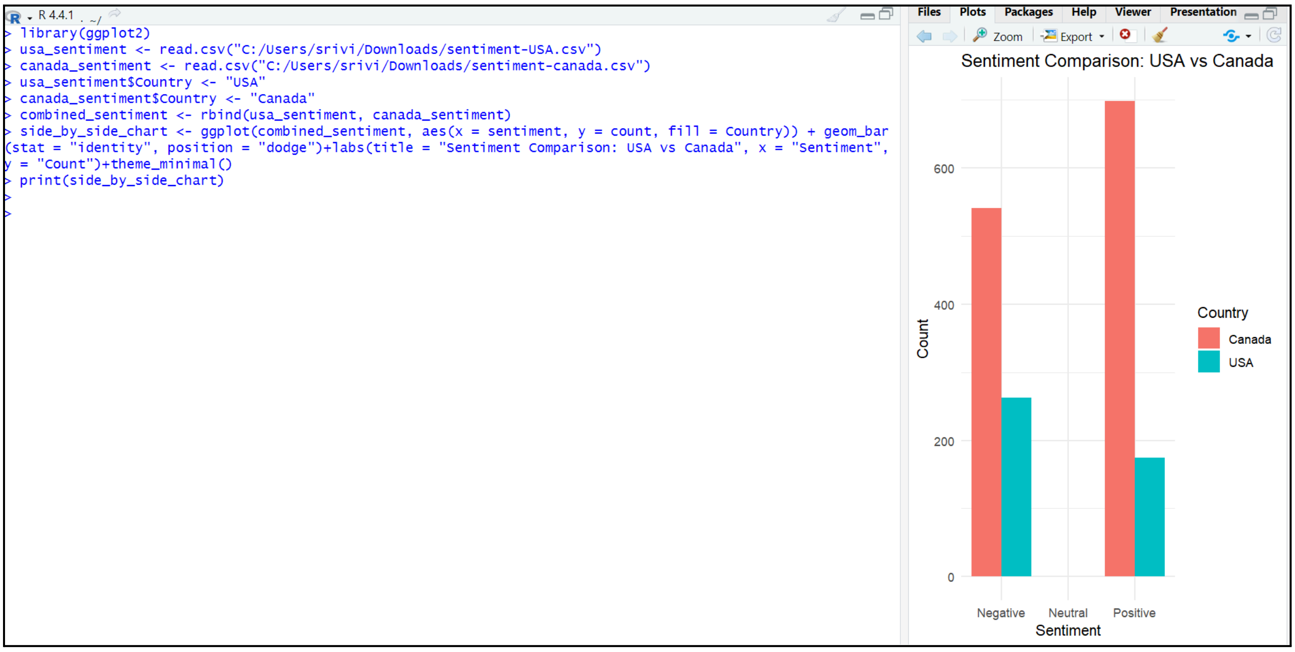
combined_sentiment <- rbind(usa_sentiment, canada_sentiment)

side_by_side_chart <- ggplot(combined_sentiment, aes(x = Sentiment, y = Count, fill = Country)) + geom_bar(stat = "identity", position = "dodge")+labs(title = "Sentiment Comparison: USA vs Canada", x = "Sentiment", y = "Count")+theme_minimal()

print(side_by_side_chart)

This R code begins by loading the ggplot2 library, which is widely used for creating data visualisations. It then imports sentiment data from two CSV files that were saved in the previous question, sentiment-USA.csv and sentiment-canada.csv, which contain information about the number of negative, neutral, and positive tweets related to each country. The data is read using read.csv(), which converts the CSV files into data frames in R. Once loaded, the script adds a new column called Country to each dataset, labelling the USA dataset as "USA" and the Canada dataset as "Canada". This step is important to distinguish between the two when visualising the combined data.

The rbind() function is then used to merge both datasets, stacking the rows from the USA and Canada datasets into a single data frame. After combining, the code creates a side-by-side bar chart using ggplot(). It sets Sentiment as the x-axis, Count as the y-axis, and uses the fill aesthetic to colour the bars according to the country. The geom_bar(stat = "identity", position = "dodge") ensures that bars for the USA and Canada are positioned next to each other, making it easy to compare their sentiment distributions. The chart is given a title, "Sentiment Comparison: USA vs Canada", and both axes are labelled accordingly. The theme_minimal() function applies a clean and simple visual style. The chart is then displayed with print(side_by_side_chart).



```
> library(ggplot2)
> usa_sentiment <- read.csv("C:/Users/srivi/Downloads/sentiment-USA.csv")
> canada_sentiment <- read.csv("C:/Users/srivi/Downloads/sentiment-canada.csv")
> usa_sentiment$Country <- "USA"
> canada_sentiment$Country <- "Canada"
> combined_sentiment <- rbind(usa_sentiment, canada_sentiment)
> side_by_side_chart <- ggplot(combined_sentiment, aes(x = sentiment, y = count, fill = Country)) + geom_bar
(stat = "identity", position = "dodge")+labs(title = "Sentiment Comparison: USA vs Canada", x = "Sentiment",
y = "Count")+theme_minimal()
> print(side_by_side_chart)
>
>
```

5.

The data shows that tweets mentioning the USA are divided into 263 negative tweets, 0 neutral tweets, and 175 positive tweets, resulting in a total of 438 tweets. For Canada, there are 379 negative tweets, 0 neutral tweets, and 255 positive tweets, with a total of 634 tweets.

Observations:
Neither the USA nor Canada received any neutral tweets, indicating that all the tweets are distinctly polarised– either positive or negative, with no in-between sentiment present. This suggests that the discourse surrounding both countries is more definitive in terms of opinion, with no ambiguity in the analysed tweets.

In terms of positive sentiment, the USA recorded 175 positive tweets, which is lower than Canada's 255 positive tweets. Looking at the proportion of positive sentiment, it represents approximately 39.9% of total tweets for the USA, while for Canada, positive tweets make up about 40.2% of its total. Although the difference is slight, Canada enjoys a marginally higher proportion of positive sentiment, suggesting a slightly more favourable view of the country compared to the USA.

Regarding negative sentiment, the USA recorded 263 negative tweets, while Canada had 379. Negative sentiment accounts for 60.1% of total tweets about the USA, whereas in Canada, it makes up 59.8% of the total tweets. Both countries show a similar balance between positive and negative sentiment, though the USA has a marginally higher percentage of negative tweets, pointing towards a slightly more critical discourse.

Key Differences:
There are some key differences between the two countries in terms of tweet volume and sentiment distribution. Canada has a higher total number of tweets with 634 mentions compared to the USA's 438. This is somewhat unexpected, as the USA is typically a more prominent topic in global discussions. However, in this dataset, Canada is mentioned more frequently.

In terms of sentiment, both countries show a similar pattern, with no neutral tweets and a slightly higher proportion of negative sentiment. However, Canada has a more balanced sentiment distribution, with a slightly larger share of positive tweets both in raw numbers and as a percentage of its total tweet volume. This suggests that public perception of Canada is a little more positive compared to the USA.

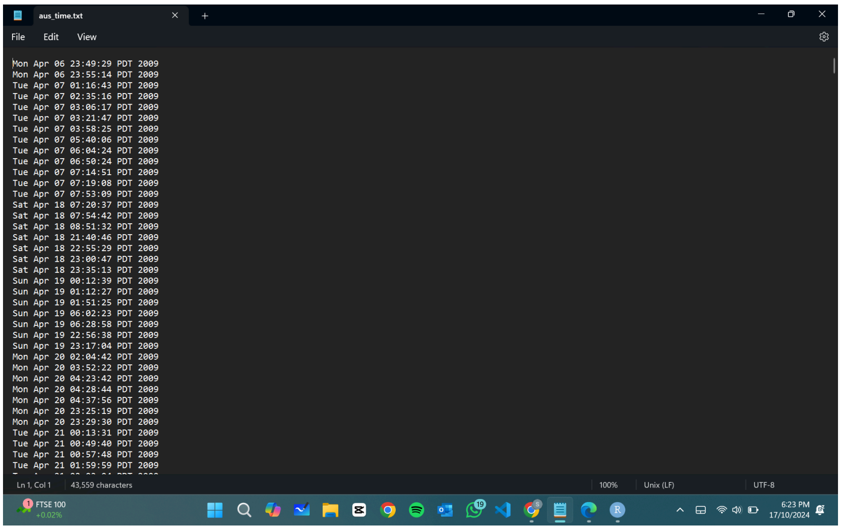
When it comes to public perception, both the USA and Canada are subject to significant negative sentiment. However, Canada's sentiment is more balanced, with a smaller gap between positive and negative tweets. This suggests that while both countries face a critical discourse, Canada is viewed in a somewhat more favourable light. The USA, on the other hand, appears to evoke more polarised opinions, with a higher percentage of negative tweets indicating a more divided perception.

In conclusion, this sentiment analysis highlights some clear differences between how the USA and Canada are perceived on social media. While Canada is discussed more frequently within this dataset, the sentiment is more balanced, leaning slightly towards the positive. The USA, in contrast, experiences more polarised sentiment, with a larger share of negative tweets. Despite this, the overall sentiment towards both countries shows a degree of criticism, with Canada being viewed somewhat more favourably than the USA.

# A4.

1. $ gunzip -c FIT1043_Dataset.gz | grep -i " Australia" | awk -F',' '{print $3}' > aus_time.txt

The command `gunzip -c FIT1043_Dataset.gz | grep -i "Australia" | awk -F',' '{print $3}' > aus_time.txt` was used to extract the timestamps of tweets mentioning "Australia." The `grep -i "Australia"` searches for case-insensitive mentions of "Australia," and `awk -F, '{print $3}'` extracts the third column, which contain timestamps, saving them to the file `aus_time.txt`.

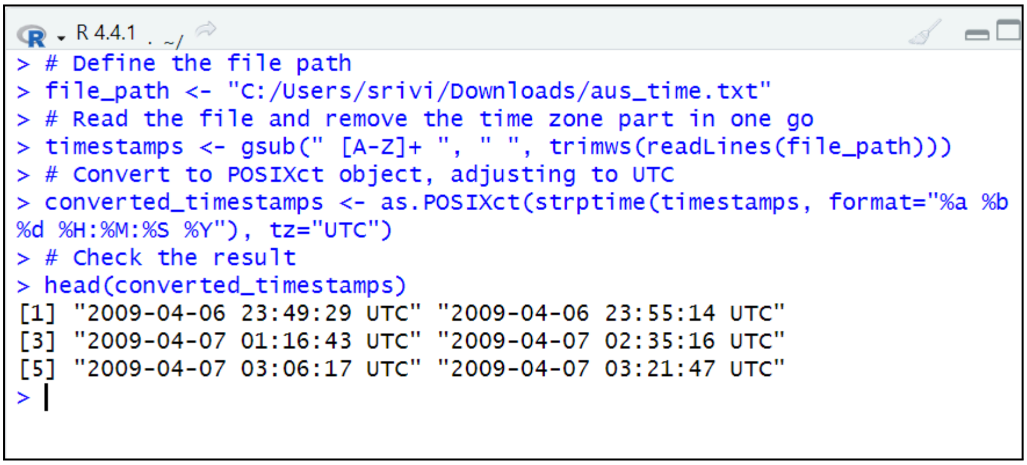


```
Mon Apr 06 23:49:29 PDT 2009
Mon Apr 06 23:55:14 PDT 2009
Tue Apr 07 01:16:43 PDT 2009
Tue Apr 07 02:35:16 PDT 2009
Tue Apr 07 03:06:17 PDT 2009
Tue Apr 07 03:21:47 PDT 2009
Tue Apr 07 03:58:25 PDT 2009
Tue Apr 07 05:40:06 PDT 2009
Tue Apr 07 06:04:24 PDT 2009
Tue Apr 07 06:50:24 PDT 2009
Tue Apr 07 07:14:51 PDT 2009
Tue Apr 07 07:19:08 PDT 2009
Tue Apr 07 07:53:09 PDT 2009
Sat Apr 18 07:20:37 PDT 2009
Sat Apr 18 07:54:42 PDT 2009
Sat Apr 18 08:51:32 PDT 2009
Sat Apr 18 21:40:46 PDT 2009
Sat Apr 18 22:55:29 PDT 2009
Sat Apr 18 23:00:47 PDT 2009
Sat Apr 18 23:35:13 PDT 2009
Sun Apr 19 00:12:39 PDT 2009
Sun Apr 19 01:12:27 PDT 2009
Sun Apr 19 01:51:25 PDT 2009
Sun Apr 19 06:02:23 PDT 2009
Sun Apr 19 06:28:58 PDT 2009
Sun Apr 19 22:56:38 PDT 2009
Sun Apr 19 23:17:04 PDT 2009
Mon Apr 20 02:04:42 PDT 2009
Mon Apr 20 03:52:22 PDT 2009
Mon Apr 20 04:23:42 PDT 2009
Mon Apr 20 04:28:44 PDT 2009
Mon Apr 20 04:37:56 PDT 2009
Mon Apr 20 23:25:19 PDT 2009
Mon Apr 20 23:29:30 PDT 2009
Tue Apr 21 00:13:31 PDT 2009
Tue Apr 21 00:49:40 PDT 2009
Tue Apr 21 00:57:48 PDT 2009
Tue Apr 21 01:59:59 PDT 2009
```

2.

```
file_path <- "C:/Users/srivi/Downloads/aus_time.txt"

timestamps <- gsub(" [A-Z]+ ", " ", trimws(readLines(file_path)))

converted_timestamps <- as.POSIXct(strptime(timestamps, format="%a %b %d %H:%M:%S %Y"),
tz="UTC")

head(converted_timestamps)
```

The R code reads timestamp data from the file located at "C:/Users/srivi/Downloads/aus_time.txt" using readLines() and cleans the timestamps by removing unnecessary uppercase letters (such as time zones) using gsub() with a regular expression. The gsub() function replaces sequences of uppercase letters with a space, and trimws() removes any extra whitespace around the text. After cleaning the timestamps, the strptime() function converts the text strings into structured date-time objects by matching the format "%a %b %d %H:%M:%S %Y" (which includes abbreviated weekday and month names, day, time, and year). These are further converted to the POSIXct format using as.POSIXct() with the time zone set to UTC, enabling easy manipulation of the timestamps. The head() function then displays the first few converted timestamps to verify if the needed changes are made



```
> # Define the file path
> file_path <- "C:/Users/srivi/Downloads/aus_time.txt"
> # Read the file and remove the time zone part in one go
> timestamps <- gsub(" [A-Z]+ ", " ", trimws(readLines(file_path)))
> # Convert to POSIXct object, adjusting to UTC
> converted_timestamps <- as.POSIXct(strptime(timestamps, format="%a %b
%d %H:%M:%S %Y"), tz="UTC")
> # Check the result
> head(converted_timestamps)
[1] "2009-04-06 23:49:29 UTC" "2009-04-06 23:55:14 UTC"
[3] "2009-04-07 01:16:43 UTC" "2009-04-07 02:35:16 UTC"
[5] "2009-04-07 03:06:17 UTC" "2009-04-07 03:21:47 UTC"
> 
```

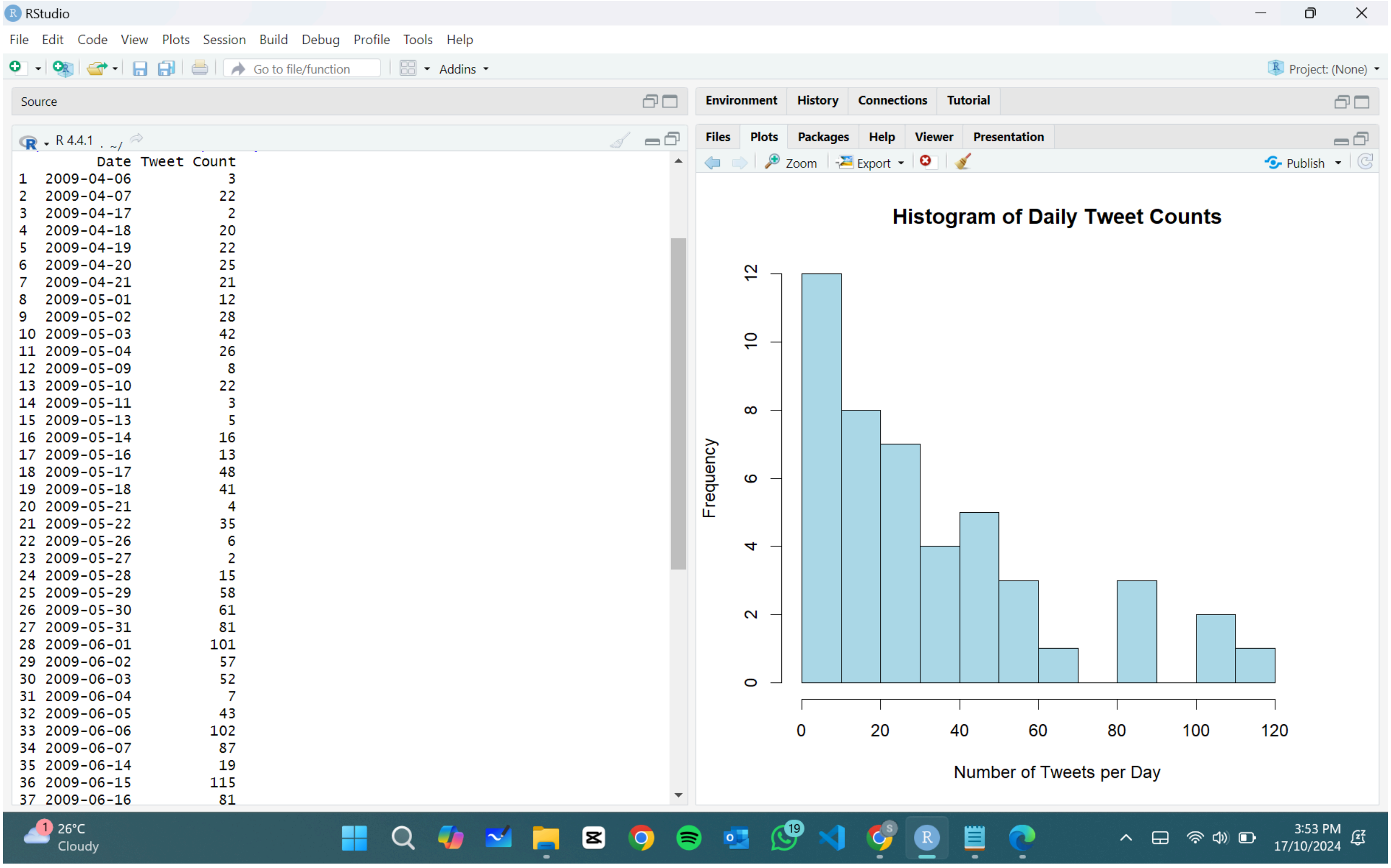
3.

```
tweet_dates <- as.Date(converted_timestamps)

tweet_counts_per_day <- as.data.frame(table(tweet_dates))

colnames(tweet_counts_per_day) <- c("Date", "Tweet Count")

print(tweet_counts_per_day)

hist(as.numeric(tweet_counts_per_day$`Tweet Count`),
    breaks =15,
    main = "Histogram of Daily Tweet Counts",
    xlab = "Number of Tweets per Day",
    ylab = "Frequency",
    col = "lightblue",
    border = "black",
    xlim = c(0, 130))
```

This R code processes tweet timestamps by first converting them into date objects using as.Date(), which extracts the date portion from each timestamp, ignoring the time. It then calculates the number of tweets per day by creating a frequency table with table(), which is converted into a data frame with columns renamed as "Date" and "Tweet Count" for clarity. After printing the daily tweet counts, the code generates a histogram using the hist() function to visualise the distribution of tweet counts per day. The histogram shows the frequency of different daily tweet counts, with 15 bins, light blue bars, black borders, and the x-axis limited to 0-130 tweets per day. This provides a clear picture of the variation in tweet activity across different days.

Discussion of the Histogram Results:
The histogram of daily tweet counts reveals a right-skewed distribution, indicating that the majority of days experience relatively low tweet activity, while a smaller number of days show significantly higher tweet counts. The most frequent observation is that in about 12 days, fewer than 20 tweets were made, suggesting that low tweet activity is common. As the number of tweets increases, the frequency of days with that level of activity declines sharply, with very few days exceeding 40 tweets. However, there are noticeable outliers, with a few days recording between 80 and 120 tweets, which stand out against the general trend.

This distribution suggests that most days are characterised by limited interaction or events of lesser significance, but on some specific days, there were significant spikes in tweet activity. These spikes could potentially be tied to major events or periods of heightened engagement, reflecting an irregular pattern in the daily tweet counts. The sharp decrease in the number of days with higher tweet counts, along with the occasional extreme values, points to a dynamic where typical days see low to moderate tweeting, with only rare instances of high-volume days.


RStudio
File Edit Code View Plots Session Build Debug Profile Tools Help
Go to file/function
Addins
Project: (None)
Source
R 4.4.1 · ~/
Environment History Connections Tutorial
Files Plots Packages Help Viewer Presentation
Zoom
Export
Publish
Histogram of Daily Tweet Counts
Frequency
0
2
4
6
8
10
12
0
20
40
60
80
100
120
Number of Tweets per Day
26°C
Cloudy
3:53 PM
17/10/2024

```
         Date Tweet Count
1  2009-04-06           3
2  2009-04-07          22
3  2009-04-17           2
4  2009-04-18          20
5  2009-04-19          22
6  2009-04-20          25
7  2009-04-21          21
8  2009-05-01          12
9  2009-05-02          28
10 2009-05-03          42
11 2009-05-04          26
12 2009-05-09           8
13 2009-05-10          22
14 2009-05-11           3
15 2009-05-13           5
16 2009-05-14          16
17 2009-05-16          13
18 2009-05-17          48
19 2009-05-18          41
20 2009-05-21           4
21 2009-05-22          35
22 2009-05-26           6
23 2009-05-27           2
24 2009-05-28          15
25 2009-05-29          58
26 2009-05-30          61
27 2009-05-31          81
28 2009-06-01         101
29 2009-06-02          57
30 2009-06-03          52
31 2009-06-04           7
32 2009-06-05          43
33 2009-06-06         102
34 2009-06-07          87
35 2009-06-14          19
36 2009-06-15         115
37 2009-06-16          81
```