

Assignment 3

FIT1043

Timothy Wee Yan Yi
31965830

Data in the 21st century is big. With almost every major company having a database, it is important to understand how one extracts and cleans data from these large files.

Part A

The following part requires us to analyse the zipped FB_Dataset file to gather useful information. I used cygwin with bash commands to answer the questions outlined in the assignment guidelines. The following numbers correspond to the assignment questions.

1.

```
letst@LAPTOP-ND2N007H /cygdrive/c/Users/letst/Downloads
$ gunzip FB_Dataset

letst@LAPTOP-ND2N007H /cygdrive/c/Us
$ ls -l FB_Dataset
-rwx-----+ 1 letst letst 359766057
```

Code: gunzip FB_Dataset
ls -l FB_Dataset

Gunzip is used to unzip the compressed file. Once unzipped the command `ls -l 'filename'` helps us to see how big the file is. Based on the output it is 359766057 bytes or roughly 359.7 MB

2. A comma is used as the delimiter

```
letst@LAPTOP-ND2N007H /cygdrive/c/Users/letst/Desktop
$ wc -l FB_Dataset
533940 FB_Dataset
```

Code: wc -l FB_Dataset

The wc (wordcount) command is used to read some statistics about your file. There are 533940 rows in the file

3.

```
letst@LAPTOP-ND2N007H /cygdrive/c/Users/letst/Desktop
$ head -1 FB_Dataset
page_name,post_id,page_id,post_name,message,description,caption,post_type,status_type,likes_
d_count,thankful_count,angry_count,post_link,picture,posted_at
```

Code: head -1 FB_Dataset

Using the head command, we can print out the first row in the dataset giving us the column types. There are 21 columns in total separated by a comma.

4.

```
letst@LAPTOP-ND2N007H /cygdrive/c/Users/letst/Desktop
$ cut -f3 FB_Dataset | sort | uniq | wc -l
533941
```

Code: `cut -f3 FB_Dataset | sort | uniq | wc -l`

`cut -f3 Fb_Dataset` extracts the 3rd column of the dataset, which is the page id. Unique pages are the different page ids each page has therefore we can just count the unique page ids in the dataset. This can be achieved by piping it with a `sort | unique` followed by counting the number of rows. I have found 533941 unique pages.

5.

```
letst@LAPTOP-ND2N007H /cygdrive/c/Users/letst/Desktop
$ awk -F ',' '{print $21}' FB_Dataset | sort | uniq | head -5
1/1/12 0:30
1/1/12 13:09
1/1/12 14:00
1/1/12 14:15
1/1/12 15:03

letst@LAPTOP-ND2N007H /cygdrive/c/Users/letst/Desktop
$ awk -F ',' '{print $21}' FB_Dataset | sort | uniq | tail -10
9/9/16 9:46
9/9/16 9:52
9/9/16 9:55
http://nyti.ms/1F05jjF
http://nyti.ms/1G50o1c
http://nyti.ms/1Jb1Cx8
http://nyti.ms/1QVdwIR
https://external.xx.fbcdn.net/safe_image.php?d=AQCM6h0BBud5qc2y&w=1
https://external.xx.fbcdn.net/safe_image.php?d=AQDNPGKrIvxSE9SQ&w=1
635947650709027484-ThinkstockPhotos-488829993.jpg&cfs=1
posted_at

letst@LAPTOP-ND2N007H /cygdrive/c/Users/letst/Desktop
$ |
```

Code: `awk -F ',' '{print $21}' FB_Dataset | sort | uniq | head -5`
`awk -F ',' '{print $21}' FB_Dataset | sort | uniq | tail -5`

We can find the date range by sorting the dates from the smallest to the largest date. To achieve this we use the `awk` function followed by some piping `-F ','` tells the function that our delimiter is a comma and we will print the last column (date) with `print $21`. We will then sort the elements by unique values and get the head and tail of the file. From the screenshot, we can see that the date ranges from 1/1/12 0:30 to 9/9/16 9:55

6.

```
letst@LAPTOP-ND2N007H /cygdrive/c/Users/letst/Desktop
$ cut -f 1 FB_Dataset | grep "Malaysian Airlines" | head -5
abc-news,86680728811_10152262883340729,86680728811,Flight Goes Missing En Route to China,UPDATE: Malaysia Ai
as gone missing a spokeswoman has confirmed to ABC News.,abcnews.go.com,link,shared_story,2410,398,502,0,0,0
n2%2Ft31.0-8%2Fq74%2Fs720x720%2F1599401_10152267849518812_1355361430_o.jpg&cfs=1,8/3/14 2:04
abc-news,86680728811_10203375775696817,86680728811,Missing Malaysian Plane: Parallels to Doomed Air France
ysian Airlines flight draws comparisons to 2009 Air France flight.,abcnews.go.com,link,shared_story,9775,152
ernational%2Fgty_air_france_crash_tail_11_120605_wmain.jpg&cfs=1,8/3/14 22:08
abc-news,86680728811_10101236502286407,86680728811,Malaysian Flight Mystery Reveals Loophole in Passport Che
own Malaysian Airlines Flight 370 the revelation two passengers possibly used stolen European passports not
/abcn.ws/1cMyoTe,https://external.xx.fbcdn.net/safe_image.php?d=AQAu1S4zlwFtqYvu&w=130&h=130&url=http%3A%2F%
abc-news,86680728811_10101243547283177,86680728811,What We Know About Missing Malaysian Airlines Flight,What
ed_story,2221,385,185,0,0,0,0,0,0,http://abcn.ws/1gng2se,https://external.xx.fbcdn.net/safe_image.php?d=AQCx
abc-news,86680728811_10152304238235729,86680728811,Oops! Airline Runs 'Escape to Indian Ocean' Ad,Airline ap
uld-be travelers...,abcnews.go.com,link,shared_story,1754,318,636,0,0,0,0,0,0,http://abcn.ws/1dRzSMs,https://
x9_608.jpg&cfs=1,29/3/14 3:00
```

Code: `cut -f 1 FB_Dataset | grep "Malaysian Airlines" | head -5`

To find the first mention we can use the `grep` function to find rows that contain that specific keyword. This is the first message from the news source abc-news:

UPDATE: Malaysia Airlines says passengers on the missing airliner are from 13 different nationalities <http://abcn.ws/NHHeLT>

The message was regarding flight MH370, a Malaysian Airlines flight that went missing.

7.

```
letst@LAPTOP-ND2N007H /cygdrive/c/Users/letst/Desktop
$ awk -F ',' '{print $5}' FB_Dataset | head -5
message
Roberts took the unusual step of devoting the majority of his annual report to the issue of jud
Do you agree with the new law?
Some pretty cool confetti will rain down on New York City celebrators.
NULL

letst@LAPTOP-ND2N007H /cygdrive/c/Users/letst/Desktop
$ awk -F ',' '{print $5}' FB_Dataset | grep "Donald Trump" -c
3298

letst@LAPTOP-ND2N007H /cygdrive/c/Users/letst/Desktop
```

Code: `awk -F ',' '{print $5}' FB_Dataset | grep "Donald Trump" -c`

We can get the number of times Donald Trump appears in the message by using the `-c` parameter which gives the count of appearance. Donald Trump appeared 3298 times in the message field of this dataset.

8.

```

letst@LAPTOP-ND2N007H /cygdrive/c/Users/letst/Desktop
$ awk -F ',' '{print $5}' FB_Dataset | grep "Donald Trump" -c
3298

letst@LAPTOP-ND2N007H /cygdrive/c/Users/letst/Desktop
$ awk -F ',' '{print $5}' FB_Dataset | grep "Barack Obama" -c
3629

```

Code: `awk -F ',' '{print $5}' FB_Dataset | grep "Donald Trump" -c`
`awk -F ',' '{print $5}' FB_Dataset | grep "Barack Obama" -c`

We can replicate the code we used for counting the number of appearances of Donald Trump on Barack Obama. Looking at the numbers Barack Obama appeared more times in the message field thus we can infer that he is more popular as there are more news stories related to him. This could be because during the period of 2012-2016, Barack Obama was the president of the United States while Doland Trump was only a candidate thus the media attention before the 2016 presidential campaign was on Barack Obama. However, there are many other metrics that can also be used to determine popularity such as the like count or number of hearts a post gets. Therefore, a solid conclusion on which person is more popular will require further analysis.

9.

```

letst@LAPTOP-ND2N007H /cygdrive/c/Users/letst/Desktop
$ awk -F ',' 'BEGIN {FS=",";OFS=","} {if ($10 >= 1000) print $2, $5, $10}' FB_Dataset | gr
' | sort -k2 -n | head -5
Binary file (standard input) matches
post_id like_count
5550296508_10154572513421509 1000
86680728811_10154802229533812 1000
18468761129_10153650393881130 1001

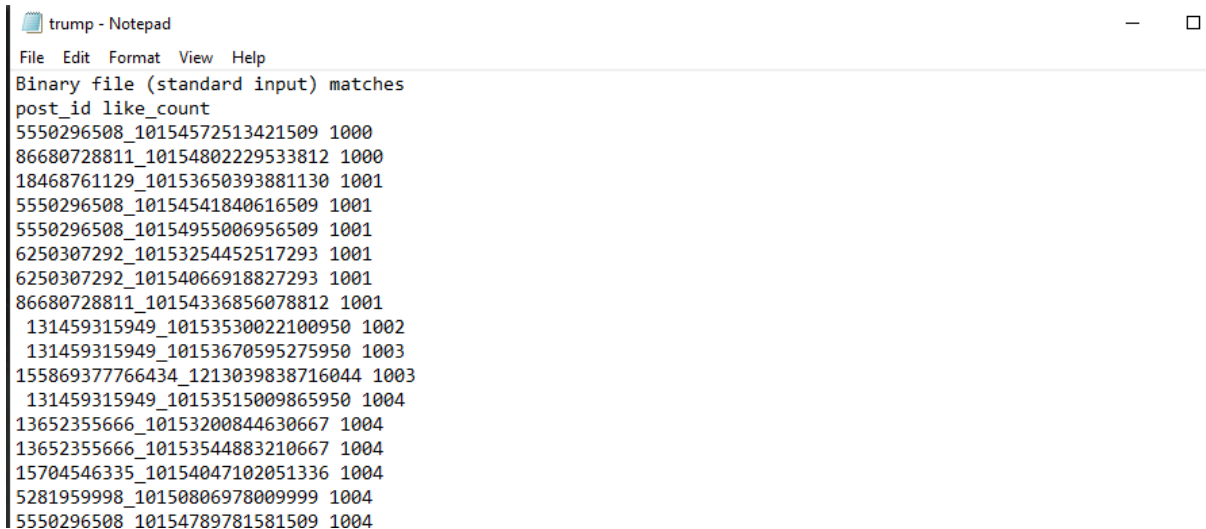
letst@LAPTOP-ND2N007H /cygdrive/c/Users/letst/Desktop
$ awk -F ',' 'BEGIN {FS=",";OFS=","} {if ($10 >= 1000) print $2, $5, $10}' FB_Dataset | gr
' | sort -k2 -n | tail -5
15704546335_10154646163096336 218748
15704546335_10154108339971336 222119
18468761129_10153524839811130 229187
5550296508_10154298504746509 248012
_22228735667216_1015396016795221722 368179

```

Code: `awk -F ',' 'BEGIN {FS=",";OFS=","} {if ($10 >= 1000) print $2, $5, $10}' FB_Dataset |`
`grep "Trump" -i | awk -F ',' '{if (NR == 1) print "post_id like_count"} {print $1, $3}' | sort -k2 -n`
`> trump.txt`

To get the like count and post id, I first begin by subsetting the dataset based on a few conditions. First our output should have a comma delimiter as it will be easier to further modify the dataset. We can use this command `BEGIN {FS=",";OFS=","}` to add a comma for

every column printed. We will now get the columns `post_id`, `message` and `likes_count`. Before that we can set a condition to only print if `likes_count` is greater or equal to 1000. Once we have this we can further subset the dataset by using the `grep` command which selects rows containing the keyword Trump. `-i` is used to ignore case. Once we have this we can remove the 2nd column (`message`). This can be done by using the `awk` command again and only printing the first (`post_id`) and third (`like count`) columns. Once we have this we can sort the 2nd column in ascending order. Before outputting into the file we will add the column header by printing the header only on the first line (*if NR == 1*).



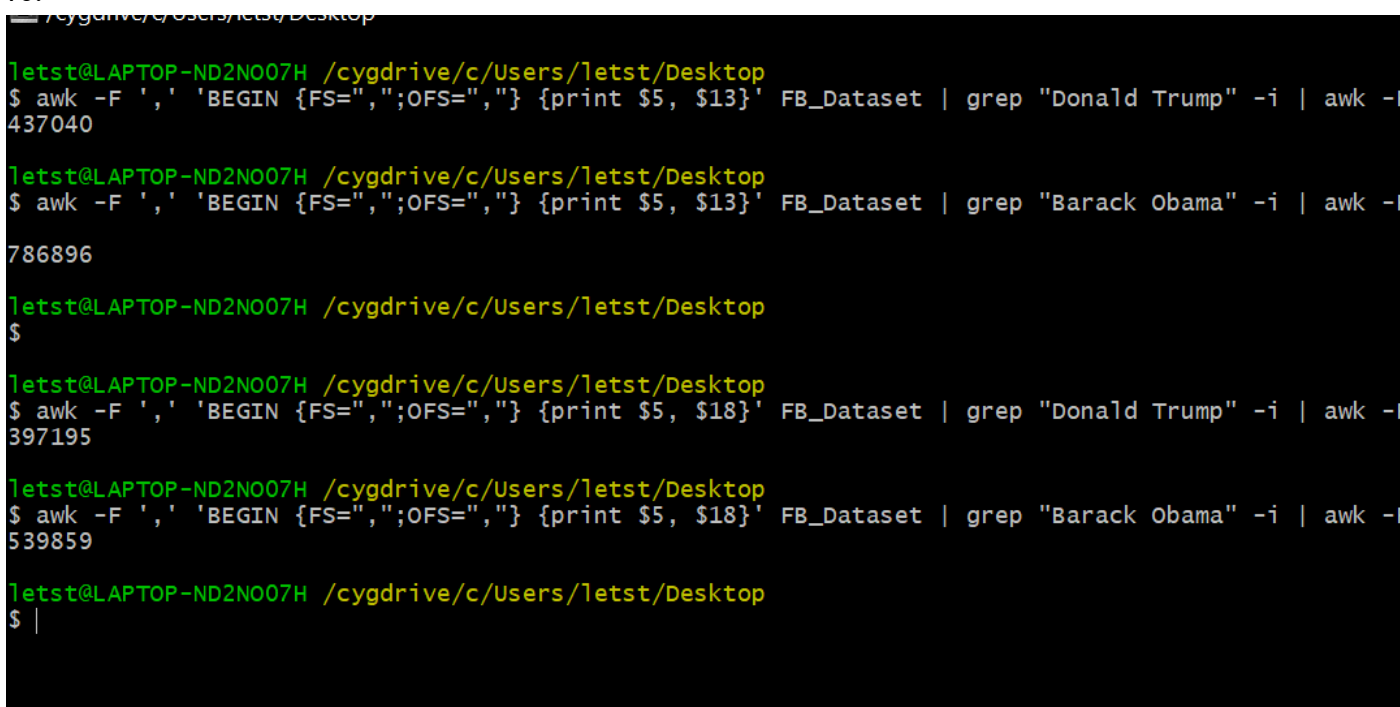
```

trump - Notepad
File Edit Format View Help
Binary file (standard input) matches
post_id like_count
5550296508_10154572513421509 1000
86680728811_10154802229533812 1000
18468761129_10153650393881130 1001
5550296508_10154541840616509 1001
5550296508_10154955006956509 1001
6250307292_10153254452517293 1001
6250307292_10154066918827293 1001
86680728811_10154336856078812 1001
131459315949_10153530022100950 1002
131459315949_10153670595275950 1003
155869377766434_1213039838716044 1003
131459315949_10153515009865950 1004
13652355666_10153200844630667 1004
13652355666_10153544883210667 1004
15704546335_10154047102051336 1004
5281959998_10150806978009999 1004
5550296508_10154789781581509 1004

```

Screenshot of the text file

10.



```

/cygdrive/c/Users/letst/Desktop
letst@LAPTOP-ND2N007H /cygdrive/c/Users/letst/Desktop
$ awk -F ' ' 'BEGIN {FS="";OFS=","} {print $5, $13}' FB_Dataset | grep "Donald Trump" -i | awk -F ' ' 'BEGIN {FS="";OFS=","} {print $5, $13}'
437040

letst@LAPTOP-ND2N007H /cygdrive/c/Users/letst/Desktop
$ awk -F ' ' 'BEGIN {FS="";OFS=","} {print $5, $13}' FB_Dataset | grep "Barack Obama" -i | awk -F ' ' 'BEGIN {FS="";OFS=","} {print $5, $13}'
786896

letst@LAPTOP-ND2N007H /cygdrive/c/Users/letst/Desktop
$

letst@LAPTOP-ND2N007H /cygdrive/c/Users/letst/Desktop
$ awk -F ' ' 'BEGIN {FS="";OFS=","} {print $5, $18}' FB_Dataset | grep "Donald Trump" -i | awk -F ' ' 'BEGIN {FS="";OFS=","} {print $5, $18}'
397195

letst@LAPTOP-ND2N007H /cygdrive/c/Users/letst/Desktop
$ awk -F ' ' 'BEGIN {FS="";OFS=","} {print $5, $18}' FB_Dataset | grep "Barack Obama" -i | awk -F ' ' 'BEGIN {FS="";OFS=","} {print $5, $18}'
539859

letst@LAPTOP-ND2N007H /cygdrive/c/Users/letst/Desktop
$ |

```

Code:

```
awk -F ' ' 'BEGIN {FS="";OFS=""} {print $5, $13}' FB_Dataset | grep "Donald Trump" -i | awk -F ' ' '{ sum += $2 } END { print sum }'
```

```
awk -F ' ' 'BEGIN {FS="";OFS=""} {print $5, $13}' FB_Dataset | grep "Barack Obama" -i | awk -F ' ' '{ sum += $2 } END { print sum }'
```

```
awk -F ' ' 'BEGIN {FS="";OFS=""} {print $5, $18}' FB_Dataset | grep "Donald Trump" -i | awk -F ' ' '{ sum += $2 } END { print sum }'
```

```
awk -F ' ' 'BEGIN {FS="";OFS=""} {print $5, $18}' FB_Dataset | grep "Barack Obama" -i | awk -F ' ' '{ sum += $2 } END { print sum }'
```

We can replicate the code above but instead of sorting the data, we will be summing it up. This can be achieved in the *sum +-\$2* parameter which sums the 2nd column of our subsetting data (the angry/love count) and we can end our command line by printing the sum.

	Barack Obama	Donald Trump
Sum love_count	786896	437040
Sum angry_count	539859	397195

Based on this table it may seem like Barack Obama has a greater positive feeling among people compared to Donald Trump due to the higher love_count. However, it may not be a fair comparison as there could be more posts related to Barack Obama thus increasing the number of love/angry comments. For another comparison we will look at the percentage of love_count based on this formula: $\text{love} / (\text{love} + \text{angry}) \times 100$

Percentage of love_count

Barack Obama	Doland Trump
59.3 %	52.4%

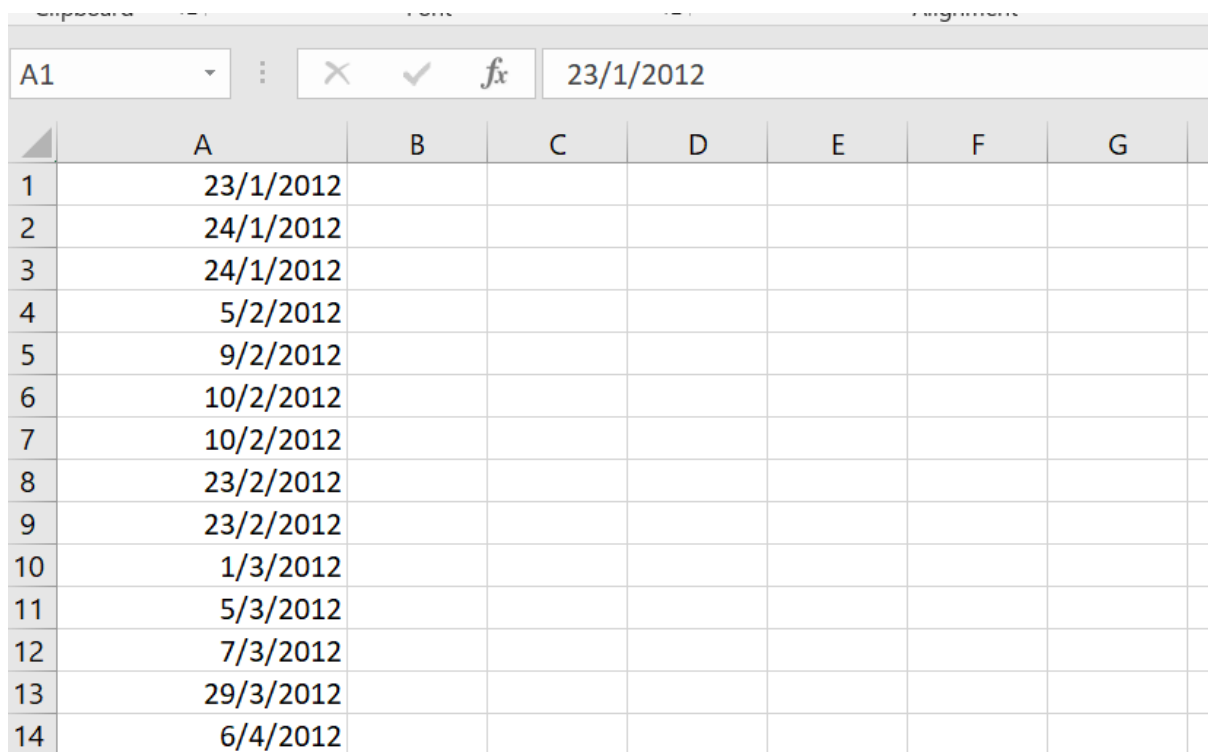
Based on the percentages, Barack Obama is more liked by only a small 6.9%. It is debatable if this is significant enough for Barack Obama to have a clear greater positive feeling among people. As both politicians represent different parties (democrats and republicans), there will be a large population of people who will agree/disagree with them which leads to many people having both positive and negative feelings towards both of them based on one's political ideology.

Task B

1. To create the histogram, I will need to extract the dates with posts related to Barack Obama. I can achieve this with the following bash command

```
awk -F ',' 'BEGIN {FS=",";OFS=","} {print $5, $21}' FB_Dataset | grep "Barack Obama" -i | awk -F ',' '{print $2}' | awk -F ' ' '{print $1}' > obama_time.csv
```

Following a similar piping format from the previous codes, we first extract rows containing "Barack Obama" with the grep command. Next, we will subset it by only extracting the posted time which is column 2 in the new subsetted data. Finally, we will remove the time posted and only take the date by printing only the date. Lastly, we will extract this into a csv file and sort it in ascending order on Excel.



	A	B	C	D	E	F	G
1	23/1/2012						
2	24/1/2012						
3	24/1/2012						
4	5/2/2012						
5	9/2/2012						
6	10/2/2012						
7	10/2/2012						
8	23/2/2012						
9	23/2/2012						
10	1/3/2012						
11	5/3/2012						
12	7/3/2012						
13	29/3/2012						
14	6/4/2012						

Screenshot of 'obama_time.csv'

We will use R to generate the histogram. We will first convert the date into a Date-Time format and create a histogram with the number of bins set to 'weeks'

R code:

```
#read the csv file
```

```
obama <- read.csv('C:\\Users\\letst/Desktop/obama_time.csv')
```

```
for (i in obama){
```

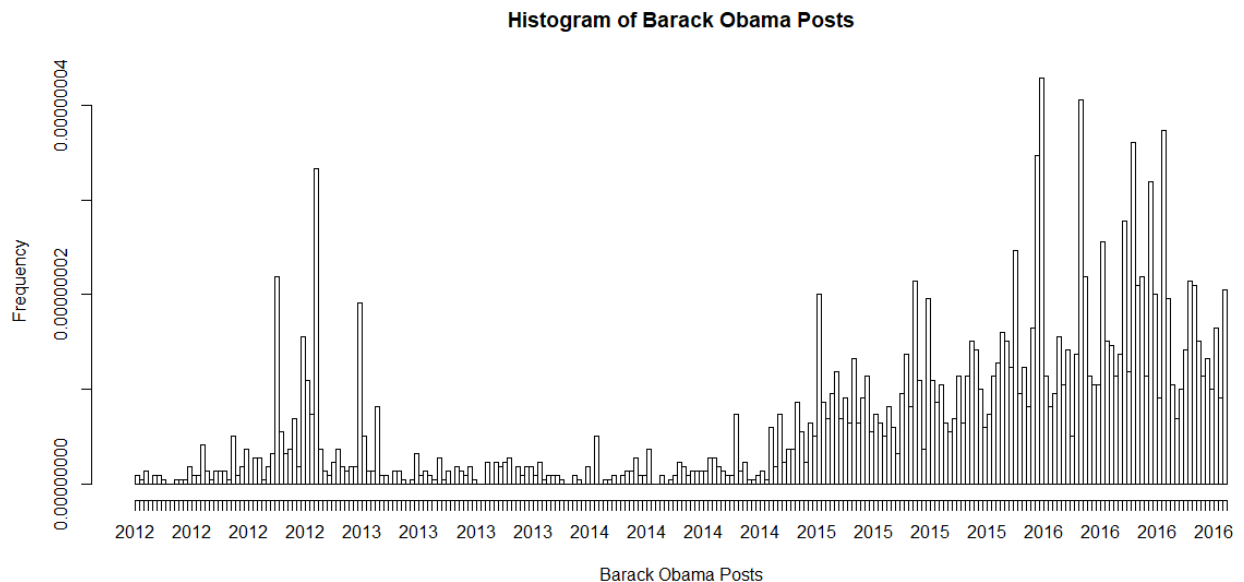
```
  x <- strptime(i, format = "%d/%m/%Y", tz = "") #format to date-time
```

```
}
```



```
options(scipen=999) #remove scientific notation

hist(x, breaks = "weeks", ylab = "Frequency", xlab = "Barack Obama Posts")
```



Based on the histogram we can notice a few different trends.

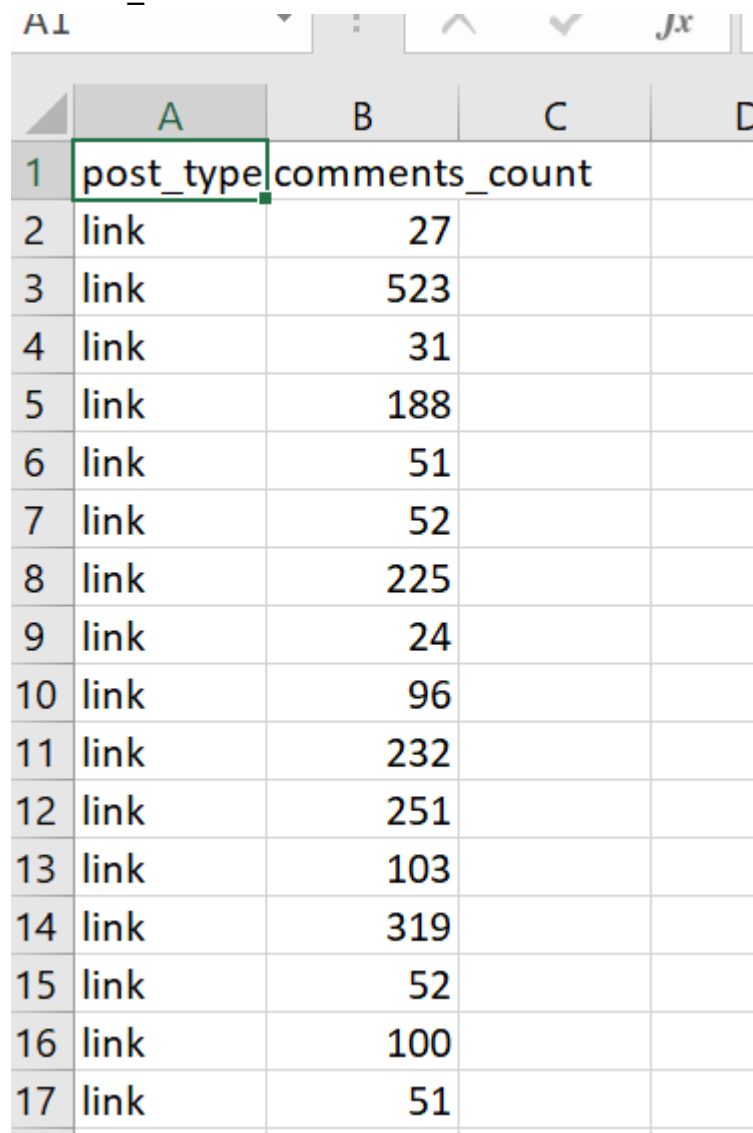
- There was a spike in posts regarding Barack Obama towards the end of 2012. This could be due to the fact that he won the 44th presidential election in 2012 (“Barack Obama | The White House,” 2021) thus there were many posts surrounding his victory.
- There were also a few spikes of posts in 2016 leading up to the 2016 presidential election. This could be due to posts surrounding the succession of Barack Obama, and overall election news coverage.
- There is an overall increase in posts related to Barack Obama from 2014 onwards. One possible explanation for this is that news agencies are posting more on Facebook compared to the earlier years. Facebook saw more and more people use its platform as it became more mainstream (“Facebook: mobile monthly active users 2016 | Statista,” 2016). This increase in users could have prompted news agencies to post more as they can reach more people which helps to grow their brand.

2. The second question asks us to find out which type of posts provide the most engagement(comment count) on the Facebook page of ABC news.

Firstly, I will extract the necessary information (post_type and comment_count) of ABC news onto an excel file with a bash command.

```
Code: awk -F ' ' 'BEGIN{FS=","; OFS = ","} {if ($1 = "abc-news") print $8, $11}' FB_Dataset > abc.csv
```

The code above checks if the page_name is “abc-news” and print out the post_type and comment_count onto a csv file.



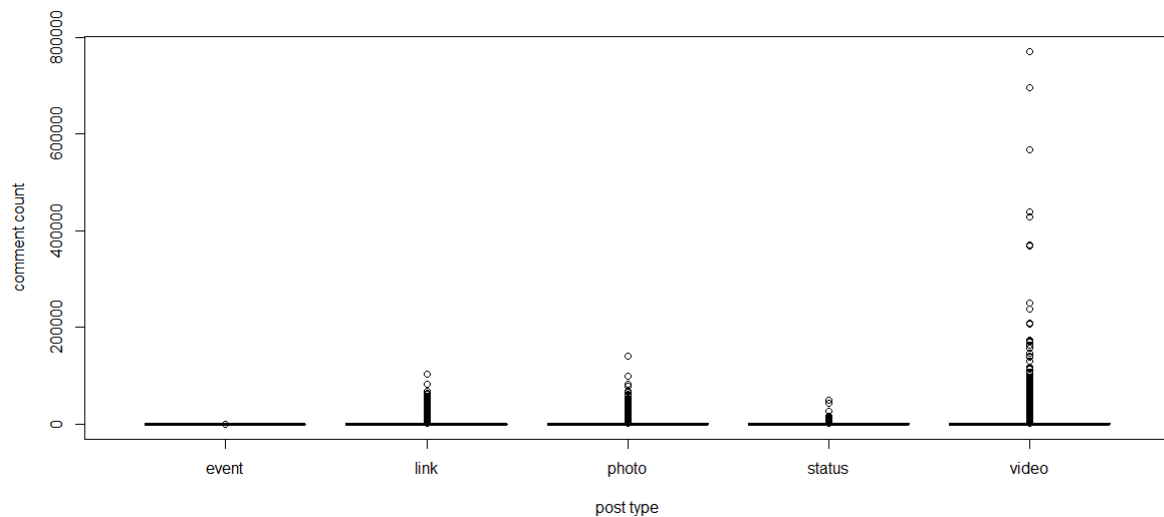
	A	B	C	D
1	post_type	comments_count		
2	link	27		
3	link	523		
4	link	31		
5	link	188		
6	link	51		
7	link	52		
8	link	225		
9	link	24		
10	link	96		
11	link	232		
12	link	251		
13	link	103		
14	link	319		
15	link	52		
16	link	100		
17	link	51		

Screenshot of 'abc.csv'

R code:

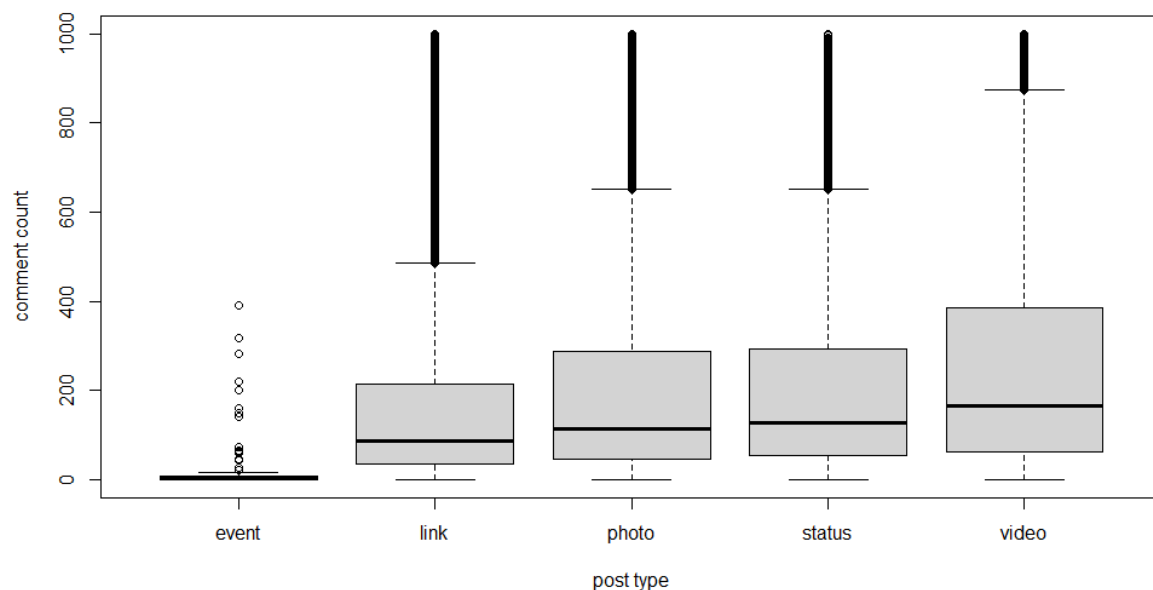
```
install.packages("dplyr")
library(dplyr)
#read the csv file
abc <- read.csv('C:\\Users\\letst/Desktop/abc.csv')
abc$comments_count = as.numeric(abc$comments_count) #change to numeric
abc$comments_count[is.na(abc$comments_count)] = 0 #remove NA values
abc <- abc %>% filter(comments_count < 1000, post_type %in% c("event", "link", "photo",
"status", "video"))
options(scipen=999) #make sure R does not convert to scientific notation
boxplot(abc$comments_count ~ abc$post_type, ylab = "comment count", xlab = "post
type")
```

Before generating the boxplot, data cleaning is needed. I will be using the dplyr library which can aid in data manipulation. The first thing is to change the comments_count to a numerical value as it was previously a character. We will also change NA values to 0 to be able to generate a boxplot. The next step is to filter the comment_count to values < 1000 to aid in visualization and choose the post_type which we want using `%in% c(...)`. Finally, we can create the boxplot with proper labeling.



Boxplot (unedited with outliers)

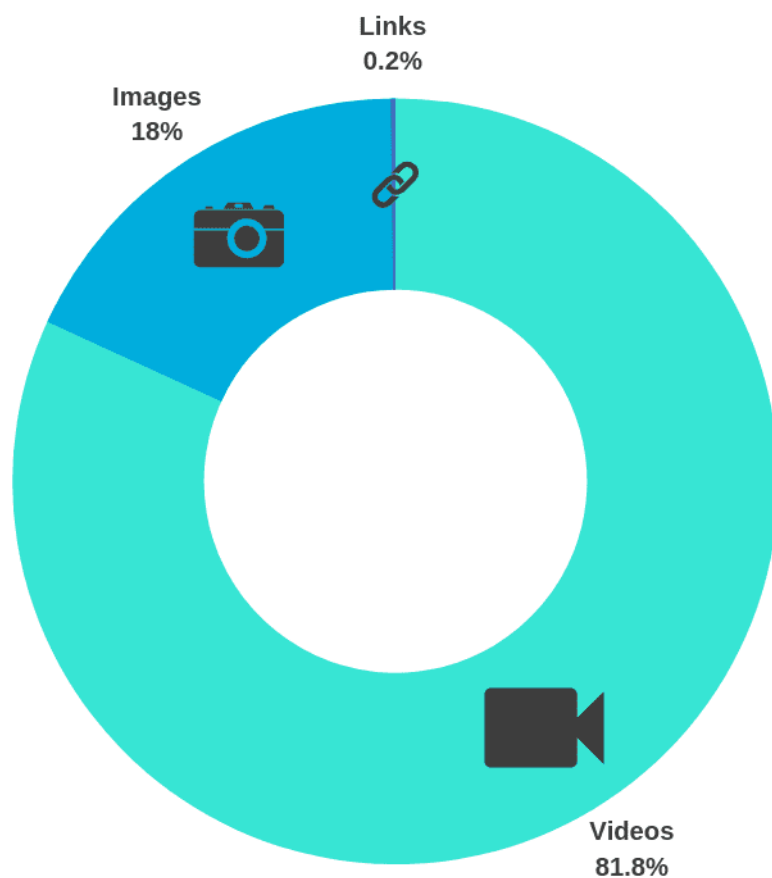
We want to remove the outliers (especially from video) to get a better visualization. The reason for these outliers could be from viral and trendy videos, for example the presidential election, and does not truly represent the engagement in general as it is content-based rather than post-type-based.



Boxplot (comments < 1000)

On average videos tend to have more comments as the median comment count is higher compared to the other post types. The nature of videos as a visual storytelling method keeps viewers engaged and hooked on the content thus they are more likely to comment their thoughts and emotions towards the video. People are also more likely to watch a video rather than read a post as it is more stimulating with colors rather than plain text. This finding is similar to another study (Peters, 2019) which shows that videos are the most popular type of Facebook post as it gives the highest engagement rate. A reason why event posts have such a low comment count could be due to the nature of the Facebook page. News pages will not have a lot of Facebook events as there is no need for them as a news agency.

Facebook Top 500 Posts of 2018 (Type)



Credit: <https://buffer.com/resources/facebook-marketing-2019/>

Conclusion

With the help of Cygwin and R, we are able to extract and clean the dataset to be able to perform very simple analytics. With these tools, the world of possibilities is endless and new data insights can be discovered if we know where to look.

References

Facebook: mobile monthly active users 2016 | Statista. (2016). Retrieved May 16, 2021, from Statista website:

<https://www.statista.com/statistics/277958/number-of-mobile-active-facebook-users-worldwide/>

Peters, B. (2019, January 10). Facebook Marketing in 2019: A Study of 777M

Facebook Posts. Retrieved May 16, 2021, from Buffer Resources website:

<https://buffer.com/resources/facebook-marketing-2019/>

dplyr package - RDocumentation. (2018). Retrieved May 16, 2021, from

Rdocumentation.org website:

<https://www.rdocumentation.org/packages/dplyr/versions/0.7.8>

Barack Obama | The White House. (2021, January 15). Retrieved May 17, 2021, from

The White House website:

<https://www.whitehouse.gov/about-the-white-house/presidents/barack-obama/>

