CHUKWUDI ANYANWU

FIT1043 Assignment 3 Semester 2, 2019

Task A: Investigating Facebook Data Using shell commands

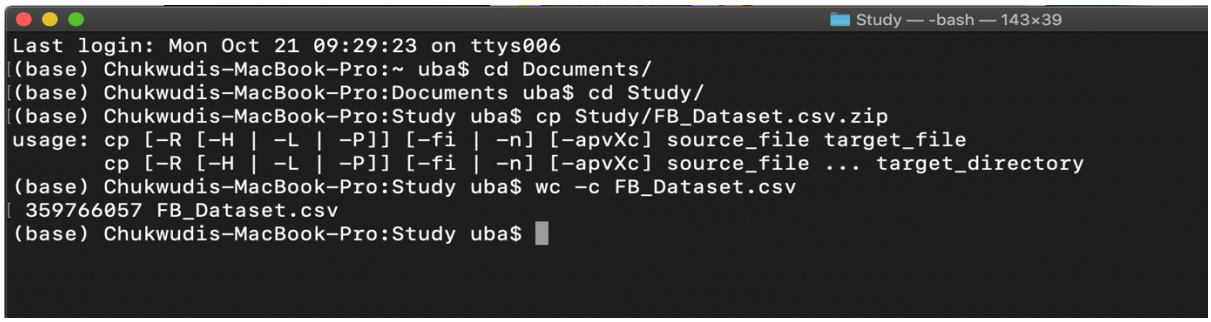
1. cp Study/FB_Dataset.csv.zip

   The above code copies the file using the cp command and decompress the file.

   wc -c FB_Dataset.csv

   The above code is used to count the bytes present in the file.

   - 359766057bytes



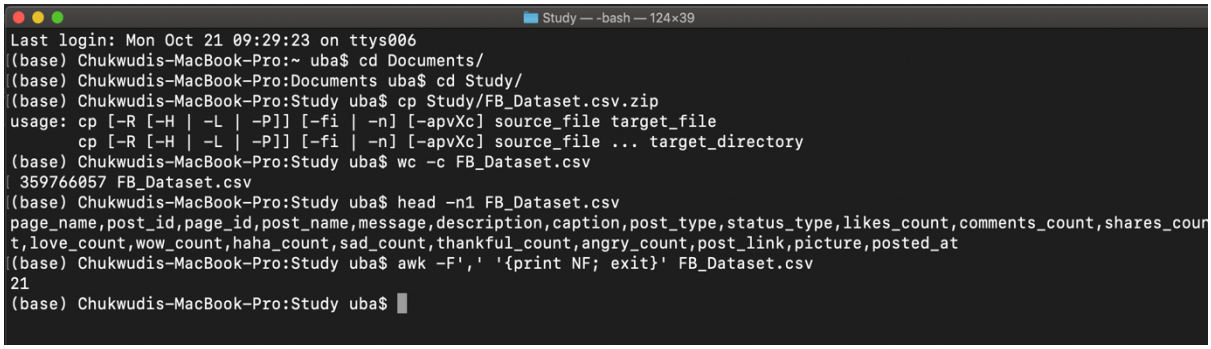
2. head -n1 FB_Dataset.csv

   The above code was used to view the first row.

   awk -F',' '{print NF; exit}' FB_Dataset.csv

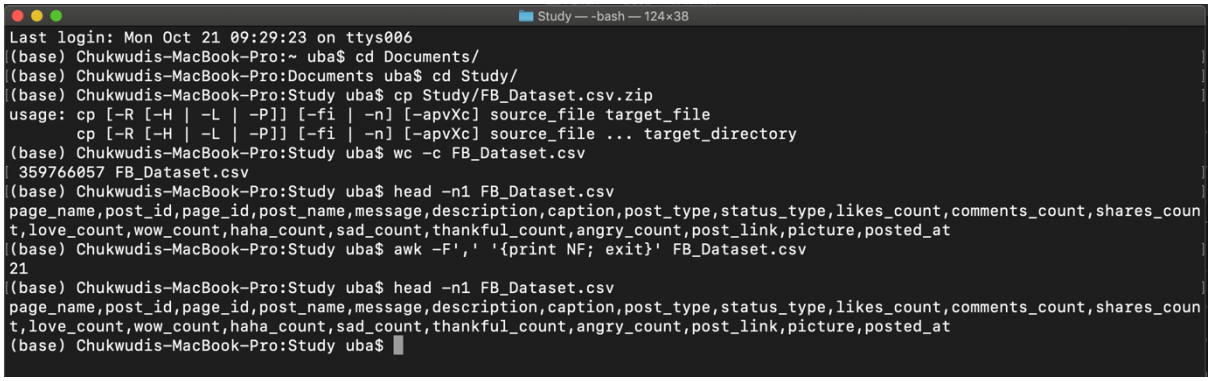
   The above code counts the number of variables after the delimiter ‘comma’ in the file.

   - 21 Columns.
   - Columns was used as delimiter and to separate each column.



3. head -n1 FB_Dataset.csv

   The above code was used to view the first row.



4. awk -F, '!seen[$3]++{c++} END{print c}' FB_Dataset.csv

The above code is used to count the unique pages (pages that hasn't been seen).

```
(base) Chukwudis-MacBook-Pro:Study uba$ awk -F, '!seen[$3]++{c++} END{print c}' FB_Dataset.csv
16
(base) Chukwudis-MacBook-Pro:Study uba$
```

- 16 unique pages

5. (head -n2; tail -n1) <FB_Dataset.csv | awk -F, '{print $21}'

The above code prints the first row after the header row in column 21 and the last row.

- The dates ranges from 1/1/12 0:30 - 7/11/16 23:45

```
(base) Chukwudis-MacBook-Pro:Study uba$ (head -n2; tail -n1) <FB_Dataset.csv | awk -F, '{print $21}'
posted_at
1/1/12 0:30
7/11/16 23:45
(base) Chukwudis-MacBook-Pro:Study uba$
```

6. grep -m 1 " Italian Dishes" FB_Dataset.csv | cut -f4,21 -d','

This code greps the first occurrence string " Italian Dishes" with the use of -m 1.

- 5 Brilliant Italian Dishes You Haven't Tried Before, 11/6/15 14:01

```
(base) Chukwudis-MacBook-Pro:Study uba$ grep -m 1 " Italian Dishes" FB_Dataset.csv | cut -f4,21 -d','
5 Brilliant Italian Dishes You Haven't Tried Before,11/6/15 14:01
(base) Chukwudis-MacBook-Pro:Study uba$
```

7. grep -o '\bDonald Trump\b' FB_Dataset.csv | wc -l

The above counted the number of times "Donald Trump" was mentioned without ignoring cases. Using the grep -o option tells grep to output each match on its online, no matter how many time the match is in the line. wc -l tells the wc utility to count the number of lines. After grep puts each match in its own line, this is the total number of occurrences of the strings in the input.

There is 13808 mention of Donald Trump

```
(base) Chukwudis-MacBook-Pro:Study uba$ grep -o '\bDonald Trump\b'  FB_Dataset.csv | wc -l
   13808
(base) Chukwudis-MacBook-Pro:Study uba$
```

8. grep -o '\bBarack Obama\b' FB_Dataset.csv | wc -l

The above counted the number of times "Barack Obama" was mentioned without ignoring cases.

- Obama was mentioned 6605 times

```
(base) Chukwudis-MacBook-Pro:Study uba$ grep -o '\bBarack Obama\b'  FB_Dataset.csv | wc -l
    6605
```

From the above two output , it appears that Donald Trump is more popular on facebook.

9. grep -i "Trump" FB_Dataset.csv | awk -F, '{ if ($10 > 100) print $2, $10 } ' | sort -k2 -n > trump.txt

The above code selected post where Trump was mentioned in the post content, ignoring the case, with number of likes greater than 100 and sorted the like_count numerical and outputted it to trump.txt.

grep -i "Trump" FB_Dataset.csv | awk -F, '{ if ($10 > 100) print $2, $10 } ' | sort -k2 -n | head -5

The above code sorted the like_count numerical and printed the first 5 data values

```
Study — -bash — 197×29
Last login: Fri Oct 25 14:58:50 on ttys003
(base) Chukwudis-MacBook-Pro:~ uba$ cd Documents
(base) Chukwudis-MacBook-Pro:Documents uba$ cd Study
(base) Chukwudis-MacBook-Pro:Study uba$ grep -i "Trump" FB_Dataset.csv | awk -F,  '{ if ($10 > 100) print $2, $10 } ' | sort  -k2 -n > trump.txt
(base) Chukwudis-MacBook-Pro:Study uba$ grep -i "Trump" FB_Dataset.csv | awk -F,  '{ if ($10 > 100) print $2, $10 } ' | sort  -k2 -n | head -5
 131459315949_10153423359555950 101
 131459315949_10153583026165950 101
 131459315949_10153707463735950 101
 131459315949_10153961477340950 101
10606591490_10153445206101491 101
(base) Chukwudis-MacBook-Pro:Study uba$
```

10. grep -i "Donald Trump" FB_Dataset.csv | awk -F, '{sum1 += $13; sum2 += $18} END {print "Donald Trump" ":" " love_count " sum1 " angry_count: " sum2} '

    The above code selected post where “Donald Trump” was mentioned in the post content, ignoring the case, and forwarded the output to awk which summed column 13 and column 18 and printed them out.

    Donald Trump: love_count: 1,566,638 angry_count: 2,197,570

```
(base) Chukwudis-MacBook-Pro:Study uba$ grep -i "Donald Trump" FB_Dataset.csv | awk -F, '{sum1 += $13; sum2 += $18} END {print "Donald Tr
ump" ":"  " love_count:  " sum1 " angry_count:  " sum2} '
Donald Trump: love_count:  1566638 angry_count:  2197570
(base) Chukwudis-MacBook-Pro:Study uba$
```

    grep -i " Barack Obama " FB_Dataset.csv | awk -F, '{sum1 += $13; sum2 += $18} END {print " Barack Obama " ":" " love_count: " sum1 " angry_count: " sum2} '

    The above code selected post where “Barack Obama” was mentioned in the post content, ignoring the case, and forwarded the output to awk which summed column 13 and column 18 and printed them out.

    Barack Obama : love_count: 596,497 angry_count: 490,842

```
(base) Chukwudis-MacBook-Pro:Study uba$ grep -i " Barack Obama " FB_Dataset.csv | awk -F, '{sum1 += $13; sum2 += $18} END {print " Barack
 Obama " ":"  " love_count:  " sum1 " angry_count:  " sum2} '
 Barack Obama : love_count:  596497 angry_count:  490842
(base) Chukwudis-MacBook-Pro:Study uba$
```

    From the above love and hate counts, Donald Trump has more angry count than love count, and Barack Obama has low love count but with a low angry count. While Trump has more love counts than Obama, when their ratios of their love count vs angry count was compared, Obama has a bigger ratio. Therefore, it can be concluded that Obama has a more positive feeling among people.

Task B: Graphing the Data in R

1. `grep "Donald Trump" FB_Dataset.csv | awk -F, '{print $21}' > trump_discussion.csv`

   The above code grep all the section that has Donald Trump mentioned and passed it to awk to print the column 21 which is the timestamp.

```
# Work in R environment #

# Increase the limit for max.print to view all the data in R.
options(max.print=999999)#

# reading data into data frame
df <- read.csv (file = 'trump_discussion.txt', header = TRUE, stringsAsFactors = FALSE)

names(df) # first print the old names

names(df)[1] <- "DateTime" # Change only the first name

head(df) # displaying the first 6 dataframe

# converting string to timestamp with format DD-MM-YYYY HH:MM
timestamp <- strptime(df$DateTime,  format = "%d/%m/%y %H:%M" )

# A frequency histogram with days as breaks
hist(timestamp, breaks = "days", freq = TRUE, include.lowest = TRUE, right = TRUE, col = "green", border = NULL, main = "Histogram for variation of discussion on Donald Trump", ylab = "Frequency", xlab = "DateTime", axes = TRUE, plot = TRUE)

# A frequency histogram with years as breaks
hist(timestamp, breaks = "years", freq = TRUE, include.lowest = TRUE, right = TRUE, col = "green", border = NULL, main = "Histogram for variation of discussion on Donald Trump", ylab = "Frequency", xlab = "DateTime", axes = TRUE, plot = TRUE)
```

```r

# Increase the limit for max.print to view all the data in R.
options(max.print=999999)

# reading data
df <- read.csv(file = 'trump_discussion.txt', header = TRUE, stringsAsFactors = FALSE)

names(df) # first print the old names

names(df)[1] <- "DateTime" # Change only the first name

head(df) # displaying the first 6 dataframe

# converting string to timestamp with format  DD-MM-YYYY HH:MM
timestamp <- strptime( df$DateTime,  format = "%d/%m/%y %H:%M" )

# A frequency histogram with days as breaks
hist(timestamp, breaks = "days",freq=TRUE, include.lowest = TRUE, right = TRUE, col = "green", border = NULL,
     main = "Histogram for variation of discussion on Donald Trump", ylab = "Frequency", xlab = "DateTime", axes = TRUE, plot = TRUE)

# A frequency histogram with years as breaks
hist(timestamp, breaks = "years",freq=TRUE, include.lowest = TRUE, right = TRUE, col = "green", border = NULL,
     main = "Histogram for variation of discussion on Donald Trump", ylab = "Frequency", xlab = "DateTime", axes = TRUE, plot = TRUE)

```

```
format = "%d/%m/%y %H:%M"
Day/Month/Year Hour: Month format was used to convert the timestamp
string into recognized and readable dateTime
```

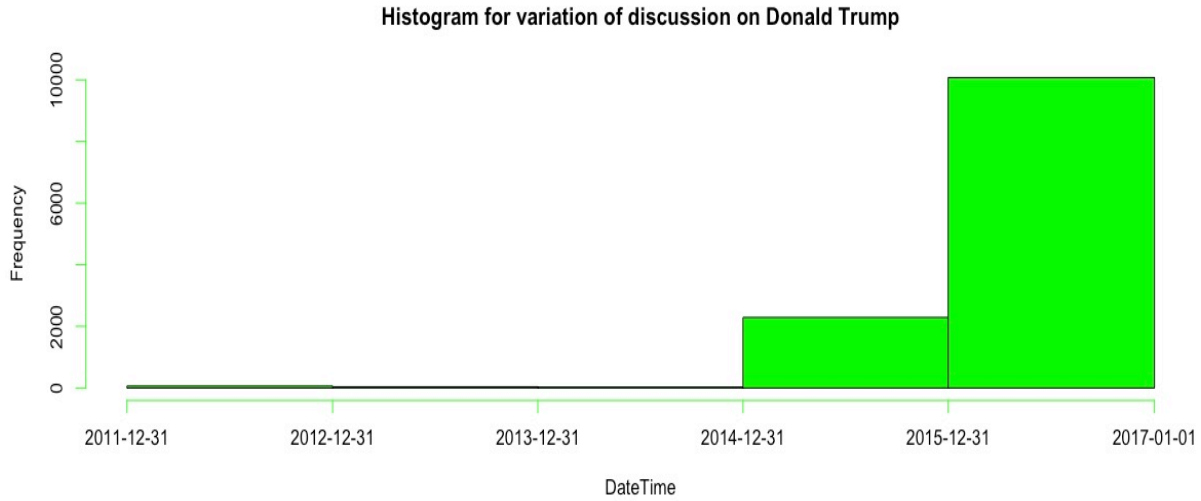


Figure above shows the date distribution with day breaks

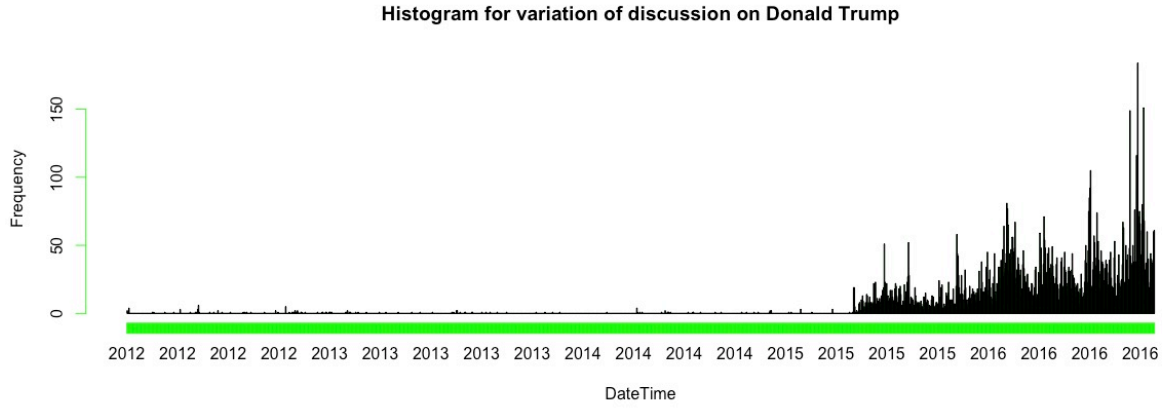


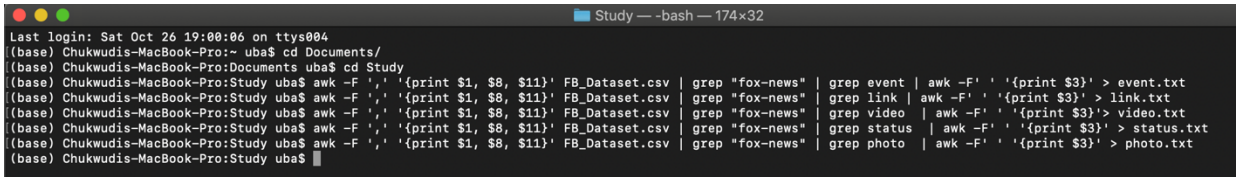
Figure above shows the date distribution with Year breaks

The above histograms show the timestamp distribution of comments regarding Donald Trump. The distribution is left skewed as all most of the data falls on the right. Both plots show distribution characterised by extreme values on the left and on that right that differ greater from other observation. On the right of the plot, there are a lot of unusual values or spikes. The problem was further investigated, and it appears that Trump's discussion rose from 2015 as he was campaigning for his presidency and it got to its highest around the year 2016 – 2017.

2.

A.
```
awk -F ',' '{print $1, $8, $11}' FB_Dataset.csv | grep "fox-news" | grep event | awk -F' ' '{print $3}' > event.txt
awk -F ',' '{print $1, $8, $11}' FB_Dataset.csv | grep "fox-news" | grep link | awk -F' ' '{print $3}' > link.txt
awk -F ',' '{print $1, $8, $11}' FB_Dataset.csv | grep "fox-news" | grep video  | awk -F' ' '{print $3}'> video.txt
awk -F ',' '{print $1, $8, $11}' FB_Dataset.csv | grep "fox-news" | grep status  | awk -F' ' '{print $3}' > status.txt
awk -F ',' '{print $1, $8, $11}' FB_Dataset.csv | grep "fox-news" | grep photo  | awk -F' ' '{print $3}' > photo.txt
```

The above codes were used to grep all the mentions of "fox-news" and grep all the post type and output the comments to txt file.

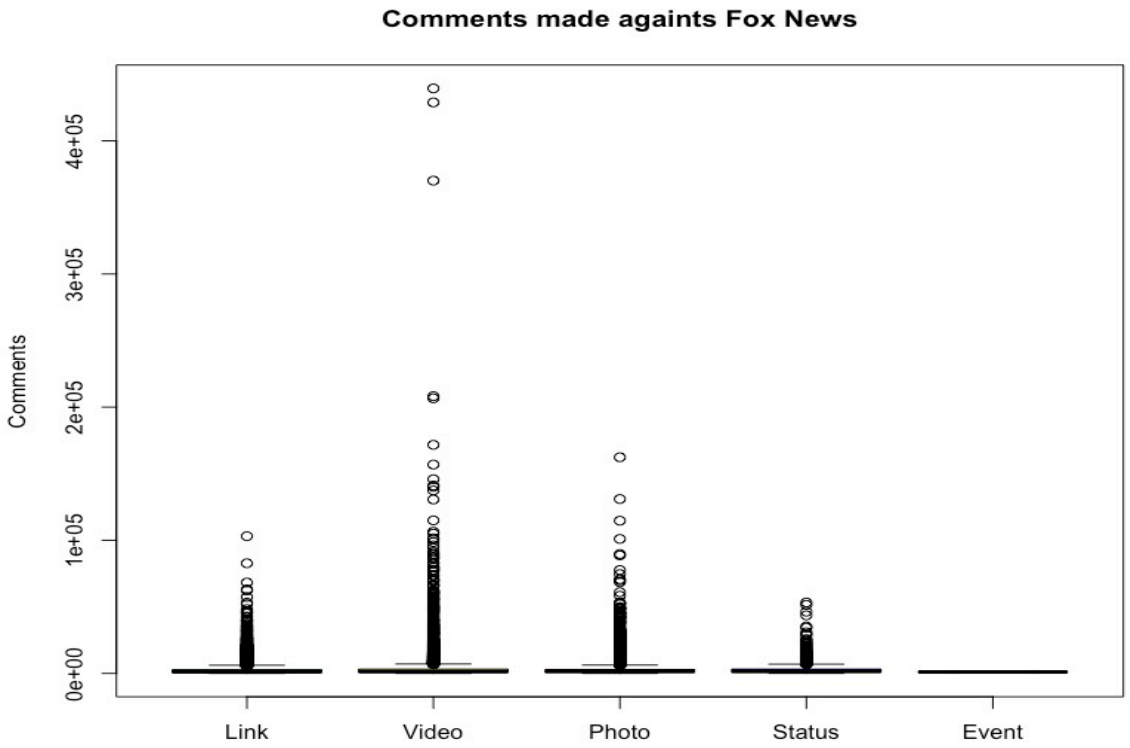



```
paste event.txt link.txt photo.txt status.txt video.txt | awk -v OFS = '\t' '{print $1, $2, $3, $4, $5}' > comments.txt
```

The above code was used to merge all the files that contain the comments from the post type and output them to comments.txt


```
# reading file into data frame
data<- read.csv(file = 'comments.txt',sep = "\t", header = TRUE, stringsAsFactors = FALSE)

# naming the columns
colnames(data) <- c("link", "video", "photo", "status", "event")

# plotting the a boxplot
boxplot(data, freq=TRUE, ylab = "Comments", col = c("green","yellow","purple","blue","brown"), main =
"Comments made againts Fox News")

# displaying the summary statistics
summary(data)
```

```
# reading file into data frame
data<- read.csv(file = 'comments.txt',sep = "\t", header = TRUE, stringsAsFactors = FALSE)

# naming the columns
colnames(data) <- c("link", "video", "photo", "status", "event")

# plotting the a boxplot
boxplot(data, freq=TRUE, ylab = "Comments", col = c("green","yellow","purple","blue","brown"),
        main = "Comments made againts Fox News")
summary(data)

# filtering data above 10000
data[data > 10000] <- NA

# boxplotting
boxplot(data, freq=TRUE, ylab = "Comments", col = c("green","yellow","purple","blue","brown"),
        main = "Comments made againts Fox News")

# displaying the summary statistics
summary(data)
```

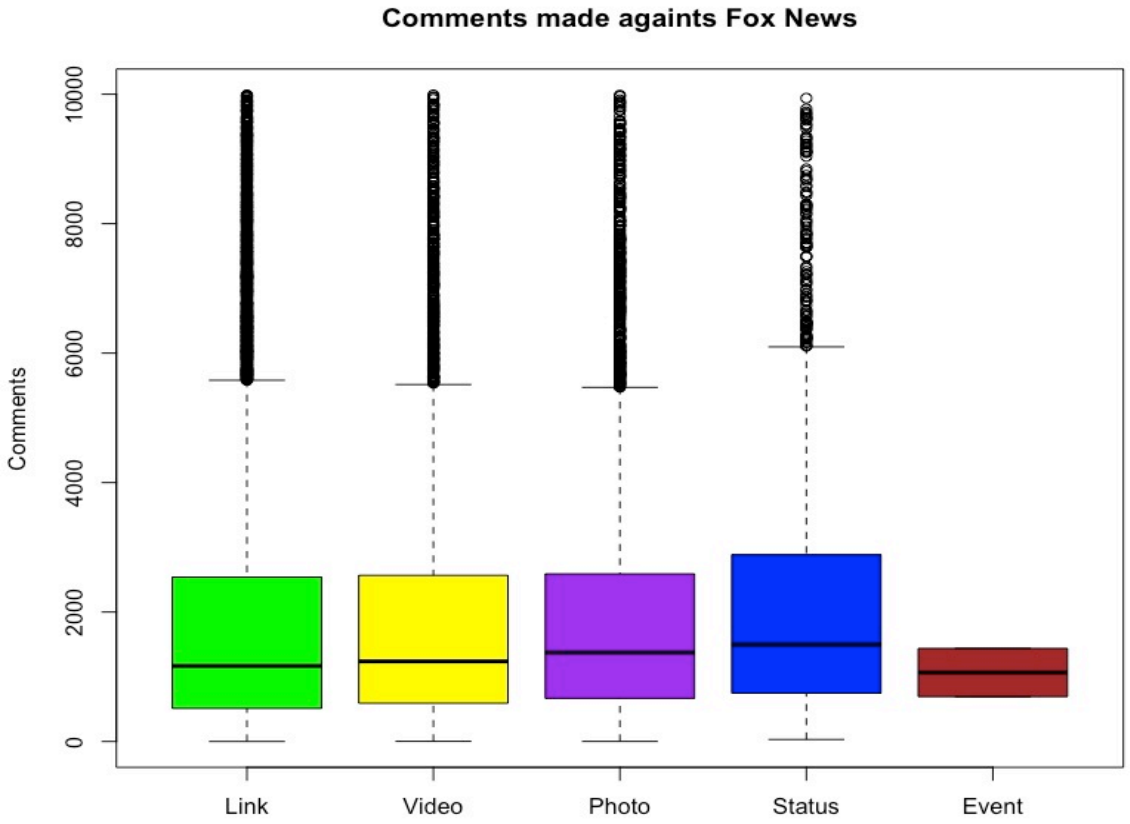



```
> summary(data)
      Link             Video            Photo             Status          Event
 Min.   :     0   Min.   :     2   Min.   :     1.0   Min.   :   31   Min.   : 692.0
 1st Qu.:   532   1st Qu.:   647   1st Qu.:   703.8   1st Qu.:  775   1st Qu.: 878.5
 Median :  1234   Median :  1378   Median :  1491.5   Median : 1556   Median :1065.0
 Mean   :  2429   Mean   :  4369   Mean   :  3295.9   Mean   : 2849   Mean   :1065.0
 3rd Qu.:  2776   3rd Qu.:  3220   3rd Qu.:  2942.5   3rd Qu.: 3236   3rd Qu.:1251.5
 Max.   :103067   Max.   :439380   Max.   :162340.0   Max.   :53263   Max.   :1438.0
                  NA's   :11556    NA's   :12267      NA's   :15590   NA's   :17309
```

Using the above graph, the range and distribution of the comments of the post type can be compared. It can be observed that most of the post type comments is skewed to the right, varies a lot with greater outliers except post type ‘Event’. Video has a wider range of outliers which indicates wider distribution with scattered data, and therefore it can be said to be the most engaging post type. And Event is the less engaging post type. Event comments is symmetric with no outliers.

B.
```
# filtering data above 10000
data[data > 10000] <- NA

# boxplotting
boxplot(data, freq=TRUE, ylab = "Comments", col = c("green","yellow","purple","blue","brown"), main =
"Comments made againts Fox News")

# displaying the summary statistics
summary(data)
```

C.




```
> summary(data)
      Link            Video           Photo           Status          Event
 Min.   :   0   Min.   :   2   Min.   :   1   Min.   :  31.0   Min.   : 692.0
 1st Qu.: 511   1st Qu.: 591   1st Qu.: 667   1st Qu.: 746.2   1st Qu.: 878.5
 Median :1167   Median :1238   Median :1374   Median :1496.5   Median :1065.0
 Mean   :1861   Mean   :1940   Mean   :1990   Mean   :2209.5   Mean   :1065.0
 3rd Qu.:2539   3rd Qu.:2567   3rd Qu.:2590   3rd Qu.:2880.5   3rd Qu.:1251.5
 Max.   :9989   Max.   :9990   Max.   :9986   Max.   :9939.0   Max.   :1438.0
 NA's   :650    NA's   :12014  NA's   :12558  NA's   :15661    NA's   :17309
>
```

In the above plot showing the distribution of comments posted against post type for fox news after filtering out 10,000 values. Looking at the median values, the post types medians ranges from 1065 to 1496. Status post type has the highest median, in other words it's has on average been the most effective for fox-news.