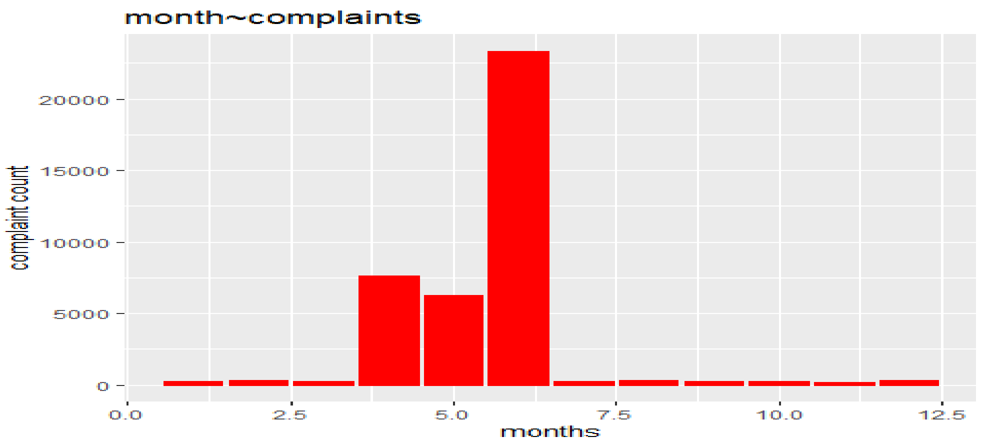
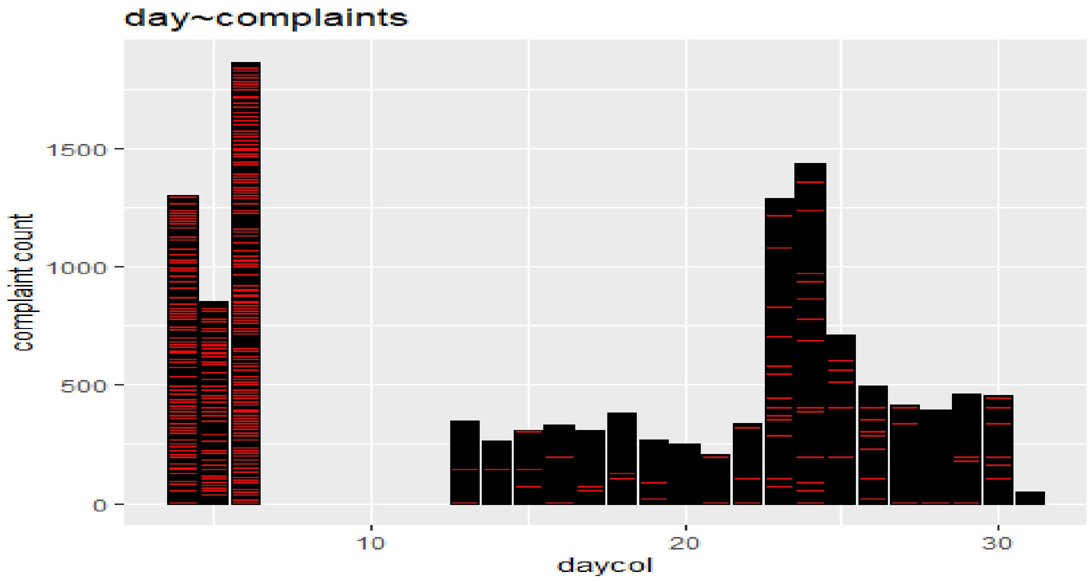
```
#Name: Priyaranjan Bisoyi
#submitted:30aug2021
#project name: Comcast Telecom Consumer Complaints .
#status:completed


if (!require = dplyr) {
  install.packages('dplyr')
  library(dplyr)
}
library(dplyr)
library(magrittr)
library(data.table)
library(plotrix)
library(ggplot2)
library(leaflet)
library(tidyr)
library(base)
library(lubridate)
install.packages('stringr')
library(stringr)

cat('\f')


###############read the input data on to Interface
comcast <-
  read.csv(
    "C:/Users/priya/OneDrive/Documents/SIMPLILEARN/R
PROGRAMMING/INPUT/Comcast Telecom Complaints data.csv",
    sep = ',',
    header = T
  )
head(comcast)#limit datarows
names(comcast)#find names in a dataset to extract columns as per
requirement
View(comcast)

#################convert date column to proper date format
comcstdt <- dmy(as.character.Date(comcast$Date))
View(comcstdt)
help("as.character.Date")

Fix(comcast_)

cat('\f')
###############drop the other date column with in proper format

comcast_ <- select(comcast, -Date)

##############Generate month,day and rename comcstdt column to date
column
comcast1 <- transform(
  comcast_,
  Date = comcstdt,
  monthcol = month(comcstdt),
  daycol = day(comcstdt)
)
```

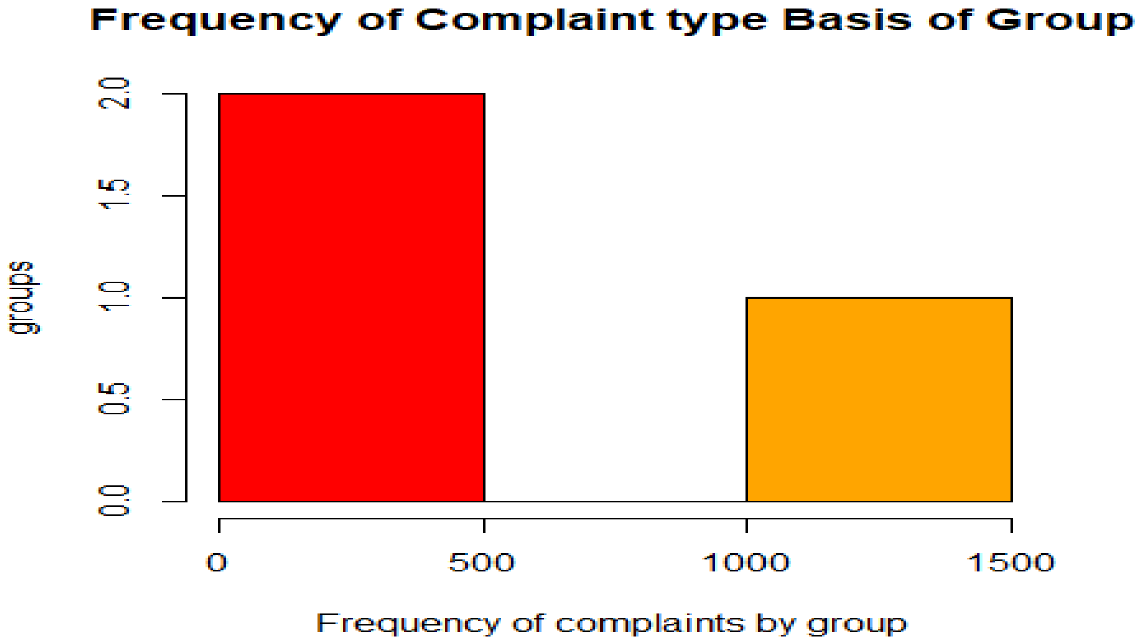
```
View(comcast1)
##############Grouping of complaints on the basis of month and day

comcastf <-
  arrange(comcast1,
          Customer.Complaint,
          monthcol,
          desc(daycol),
          group_by = T)
comcastf
help("group_by")

cat('\f')

############plot chart for customer complaints with month and days and
find out the frequency


comcastcomp <-
  select(comcastf, Customer.Complaint, monthcol, daycol)
fix(comcastcomp)
#no of complaints as per month count
plot_month_com <-
  ggplot(comcastcomp, aes(x = monthcol, y = daycol)) + geom_bar(color =
                                                               'red',
fill = "green", stat =

"identity")
plot_month_com + labs(title = 'month~complaints', x = 'months', y =
'complaint count')
```



```
#no of complaints per day in a month
plot_day_com <-
  ggplot(comcastcomp, aes(x = daycol, y = monthcol)) + geom_bar(color =

"black", fill = "red", stat =

"identity")
plot_day_com + labs(title = 'day~complaints', x = 'daycol', y =
'complaint count')
```

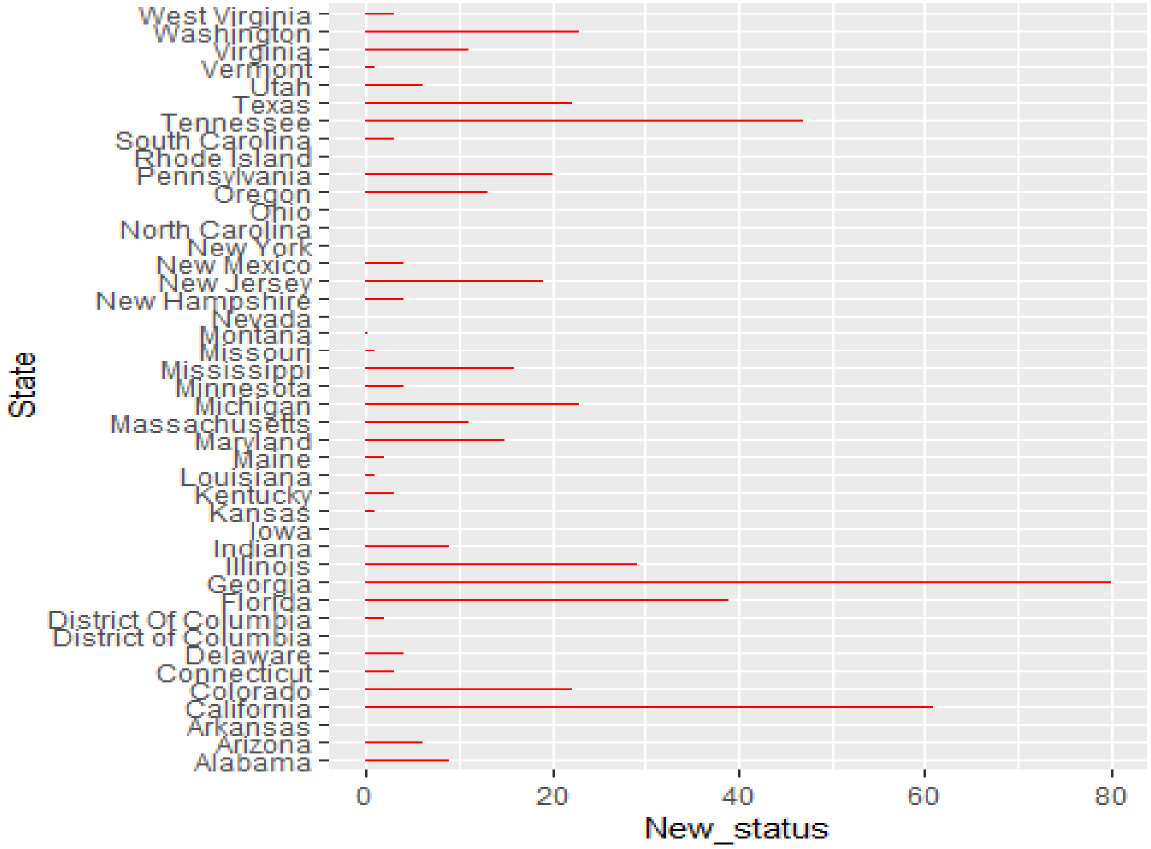


```
help(geom_bar)

##############providing data for knowing the frequency of complaints type

data(comcastf)

frq_complaint <- data.frame(table(comcastf$Customer.Complaint))
fix(frq_complaint)
frq_complaint_t <-
  transform(frq_complaint, complaint.type = toupper(Var1)) %>%
rename(Customer.Complaint =

Var1) %>% select(complaint.type)
View(frq_complaint_t)

frq_complaint_t_ <- arrange(rename(data.frame(table(
  frq_complaint_t$complaint.type
)), Customer.Complaint = Var1), desc(Freq))#differenet frquency of
complaint type

##As per problem statement group it by network, internet and other type
complaints

frq_complaint_tf <- transform(frq_complaint_t_, group_comp = ifelse(
  grepl('INTERNET', Customer.Complaint),
  'GROUP A',
  ifelse(grepl('NETWORK', Customer.Complaint), 'GROUP B', 'GROUP C')
))
frq_complaint_tf

final_frq_table <-
  data.frame(table(select(
    frq_complaint_tf,-Freq,-Customer.Complaint
  )))
final_frq_table

# Var1 Freq
# 1 GROUP A  426 = INTERNET
# 2 GROUP B    2 = NETWORK
# 3 GROUP C 1312 = OTHER TYPE
#draw histogram for visual representation
hist(
  final_frq_table$Freq,
  col = c('red', 'green', 'orange'),
  xlab = 'Frequency of complaints by group',
  ylab = 'groups',
  main = 'Frequency of Complaint type Basis of Group'
)
```



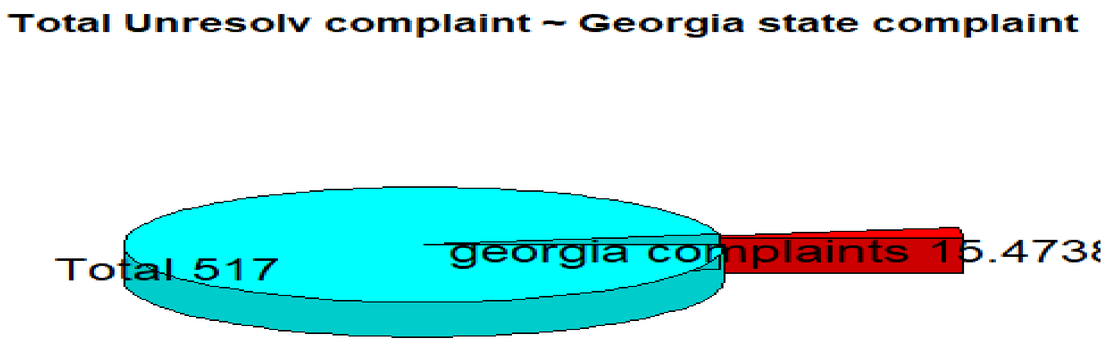
```
# - Create a new categorical variable with value as Open and Closed. Open
& Pending is to be categorized as Open
# and Closed & Solved is to be categorized as Closed.


comcastf_ <-
  transform(comcastf, status_new = ifelse(
    Status == 'Open',
    'Open',
    ifelse(
      Status == 'Pending',
      'Open',
      ifelse(
        Status == 'Closed',
        'Closed',
        ifelse(Status == 'Solved', 'Closed', 'None')
      )
    )
  ))

#assign close values as 0 and open values as 1
comcastf_1 <-
  transform(comcastf_, New_status = ifelse(status_new == 'Open', 1, 0))
View(comcastf_1)
```

```
# - Provide state wise status of complaints in a stacked bar chart. Use
the categorized variable from Q3.
# Provide insights on:

s_fr <-
  select(arrange(data.frame(table(comcastf$State)), desc(Freq)) %>%
rename(state_freq =

Freq),
          state_freq,
          everything())#state frequency
View(s_fr)

c_fr <- select(arrange(data.frame(table(
  comcastf_1$New_status
)), desc(Freq)) %>% rename(compl_freq = Freq),
compl_freq)#open or close complaint freq
View(c_fr)

#without formula just plot the bar for frquency check in barplot
ggplot(comcastf_1, aes(x = New_status, y = State, color = 'New_status'))
+
  geom_bar(
    color = 'red',
    fill = "green",
    stat =
      "identity",
    width = .0009
  )
```



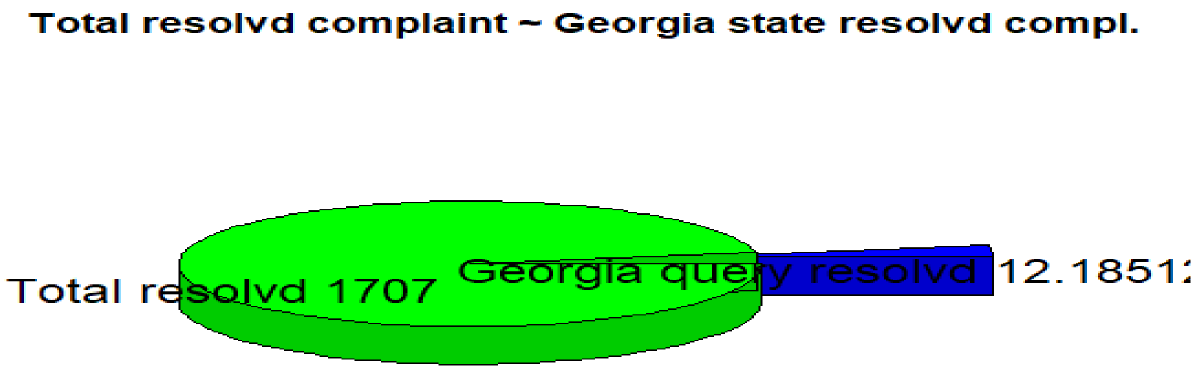
```
##Which state has the maximum complaints
```

```
max_compl_state <-
  arrange(select(data.frame(table(comcastf$State)), Var1, Freq),
desc(Freq)) %>%
  rename(complaint_total_freq = Freq, State_names = Var1)
View(max_compl_state)

# Which state has the highest percentage of unresolved complaints

View(comcastf)

#find only open queries

open_comp_tot <-
  data.frame(table(select(
    filter(comcastf_1, New_status == 1), New_status, State
  ))) %>% arrange(Freq)
View(highest_unres_comp_state <- arrange(open_comp_tot, desc(Freq)))
View(tot_unres_complaints <- sum(open_comp_tot$Freq, na.rm = T))

#####Draw pie 3d chart for visualisation

pct <-
  (max(highest_unres_comp_state$Freq) / tot_unres_complaints) * 100
##state complaint % highest

Tot <-
  tot_unres_complaints ##Total compalints registered whic are  open

fin <-
  cbind(pct, Tot) #matrix for total complaint and sate haiving highest
unresolved complaints
fin

labelur <-
  paste(c("Georgia complaints", "Total"), " ", fin, sep = "")

pie3D(
  fin,
  labels = labelur,
  explode = 0.4,
  main = "Total Unresolv complaint ~ Georgia state complaint",
  col = rainbow(2)
)
```



```
##Ans= Georgia is having highest open queries


# - Provide the percentage of complaints resolved till date, which were
received
#through the Internet and customer care calls.

#find closed complaints
close_comp_tot <-
  data.frame(table(select(
    filter(
      comcastf_1,
      New_status == 0 &
        State == 'Georgia' &
        Received.Via %in% c("Internet", "Customer Care Call")
    ),
    New_status,
    Received.Via,
    State
  ))) %>% arrange(Freq)

View(highest_res_comp_state <- arrange(close_comp_tot, desc(Freq)))
View(tot_res_complaints <- group_by(arrange(data.frame(table(
  select(
    filter(
      comcastf_1,
      New_status == 0 &
        Received.Via %in% c("Internet", "Customer Care Call")
    ),
    New_status,
    Received.Via,
    State
  )
))), State))
all_state_resol_complaint_tot <-
  sum(tot_res_complaints$Freq, na.rm = T)
all_state_resol_complaint_tot
#select now both internet and customer resolved in
whole,Received.Via,Freq
```

```
resolved_complaints_state <-
  data.frame(filter(select(tot_res_complaints, State, Freq), State ==
'Georgia'))
state_resol_complaint_tot <-
  sum(resolved_complaints_state$Freq, na.rm = T)

####find percentage of total reolved complaints of sate with correspond
to all resolved complaints of all state

pctr <-
  (state_resol_complaint_tot / all_state_resol_complaint_tot) * 100
pctr

finres <- cbind(pctr, all_state_resol_complaint_tot)
finres

labelsr <-
  paste(c("Georgia query resolvd", "Total resolvd"), " ", finres, sep =
"")

pie3D(
  fin,
  labels = labelsr,
  explode = 0.4,
  main = "Total resolvd complaint ~ Georgia state resolvd compl.",
  col = c('blue', 'Green')
)
```



```
#help(ggplot)
```