

```
RStudio
File Edit Code View Plots Session Build Debug Profile Tools Help
Go to file/function
Comcast Telecom Complaints data.R*
Source on Save Run Source
4 #Comcast is an American global telecommunication company. The firm
5 #The existing database will serve as a repository of public custom
6 #It will help to pin down what is wrong with Comcast's customer se
7
8
9 # Installing Packages
10 install.packages('ggplot2')
11 install.packages('ggpubr')
12 install.packages('dplyr')
13 install.packages('lubridate')
14 install.packages('stringi')
15 install.packages('data.table')
16 install.packages('plyr')
17
18 # Adding Libraries
19 library(ggplot2)
20 library(ggpubr)
21 library(dplyr)
22 library(lubridate)
23 library(stringi)
24 library(data.table)
25 library(plyr)
26
27 # Analysis Task
28 # 01 - Import data into R environment.
29 comcast_data<- read.csv("D:\\Data Science\\R Programming\\Projects
30 comcast_data
31
32
```

Console Terminal x

R 4.2.2 ~ /

```
> TRUE)
> comcast_data
Ticket..
1 250635
2 223441
3 242732
4 277946
5 307175
6 338519
7 361148
8 359792
9 318072
10 371214
11 255938
12 276409
13 339282
14 360178
15 376268
16 370137
17 363695
18 238694
19 230876
20 318725
21 327657
22 328742
23 328165
24 370538
25 370363
26 270163
27 355976
28 260651
```

```
Comcast Telecom Complaints data.R*
# All with help to put down what is wrong with Comcast's customer service

# Installing Packages
install.packages('ggplot2')
install.packages('ggpubr')
install.packages('dplyr')
install.packages('lubridate')
install.packages('stringi')
install.packages('data.table')
install.packages('plyr')

# Adding Libraries
library(ggplot2)
library(ggpubr)
library(dplyr)
library(lubridate)
library(stringi)
library(data.table)
library(plyr)

# Analysis Task
# 01 - Import data into R environment.
comcast_data <- read.csv("D:\\Data Science\\R Programming\\Projects\\P
comcast_data

summary(comcast_data)
```

32:22 (Top Level)

R Script

Console Terminal

```
R 4.2.2 ~/\r\n94                                     No\r\n95                                     No\r\n96                                     No\r\n97                                     No\r\n98                                     No\r\n99                                     No\r\n100                                    Yes\r\n  [ reached 'max' / getOption("max.print") -- omitted 2124 rows ]\r\n> summary(comcast_data)\r\n      Ticket..      Customer.Complaint      Date\r\nLength:2224      Length:2224      Length:2224\r\nClass :character  Class :character  Class :character\r\nMode  :character  Mode  :character  Mode  :character\r\n\r\n      Time      Received.Via      City\r\nLength:2224      Length:2224      Length:2224\r\nClass :character  Class :character  Class :character\r\nMode  :character  Mode  :character  Mode  :character\r\n\r\n      State      Zip.code      Status\r\nLength:2224      Min.   : 1075      Length:2224\r\nClass :character  1st Qu.:30057      Class :character\r\nMode  :character  Median :37211      Mode  :character\r\n                        Mean  :47994\r\n                        3rd Qu.:77059\r\n                        ..      : 88222
```

Files Plots Packages Help Viewer Presentation


```
Comcast Telecom Complaints data.R* Comcast_data*
summary(comcast_data)
View(comcast_data)

# Extracting monthly and daily count
comcast_data$Date<-dmy(comcast_data$Date)
comcast_data$datemonth<-months(as.Date(comcast_data$Date))

comcast_data$datemonth

Months_counts <- table(comcast_data$datemonth)

#Months_counts
months_counts_df<-as.data.frame(Months_counts)
names(months_counts_df)[1]<-"Months"
names(months_counts_df)[2]<-"Count"

#months_counts_df
comcast_data$date_of_each_month<-day(as.Date(comcast_data$Date))

#comcast_data$date_of_each_month
Date_counts <- table(comcast_data$date_of_each_month)

#Date_counts
Date_counts_df<-as.data.frame(Date_counts)
names(Date_counts_df)[1]<-"date_of_each_month"
names(Date_counts_df)[2]<-"Count"

#Date_counts_df
```

61:1 (Top Level)

R Script

```
Console Terminal
R 4.2.2 ~ /
[949] June May March August
[953] "September" "April" "April" "April"
[957] "April" "April" "July" "April"
[961] "April" "April" "January" "June"
[965] "August" "October" "October" "May"
[969] "April" "June" "June" "June"
[973] "June" "June" "June" "June"
[977] "June" "June" "June" "June"
[981] "May" "May" "June" "July"
[985] "May" "January" "January" "June"
[989] "June" "June" "June" "April"
[993] "April" "April" "July" "June"
[997] "April" "April" "May" "April"
[ reached getOption("max.print") -- omitted 1224 entries ]
> Months_counts <- table(comcast_data$datemonth)
> #Months_counts
> months_counts_df<-as.data.frame(Months_counts)
> names(months_counts_df)[1]<-"Months"
> names(months_counts_df)[2]<-"Count"
> #months_counts_df
> comcast_data$date_of_each_month<-day(as.Date(comcast_data
$Date))
> #comcast_data$date_of_each_month
> Date_counts <- table(comcast_data$date_of_each_month)
> #Date_counts
> Date_counts_df<-as.data.frame(Date_counts)
> names(Date_counts_df)[1]<-"date_of_each_month"
> names(Date_counts_df)[2]<-"Count"
>
> |
```


RStudio

File Edit Code View Plots Session Build Debug Profile Tools Help

Go to file/function Addins

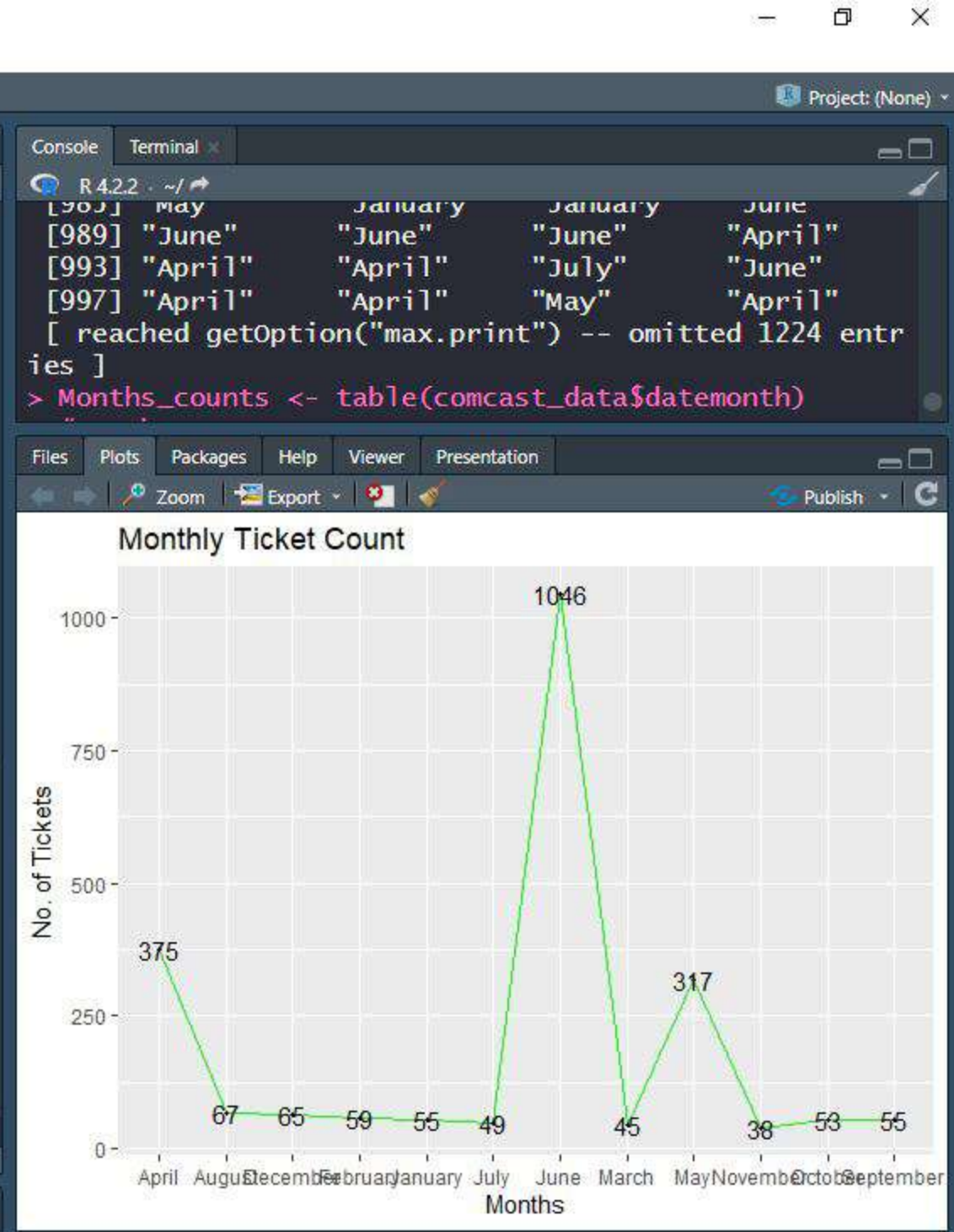
Comcast Telecom Complaints data.R* comcast_data

Source on Save Run

```
44 #Months_counts
45 months_counts_df<-as.data.frame(Months_counts)
46 names(months_counts_df)[1]<-"Months"
47 names(months_counts_df)[2]<-"Count"
48
49 #months_counts_df
50 comcast_data$date_of_each_month<-day(as.Date(comcast_data$Date))
51
52 #comcast_data$date_of_each_month
53 Date_counts <- table(comcast_data$date_of_each_month)
54
55 #Date_counts
56 Date_counts_df<-as.data.frame(Date_counts)
57
58 names(Date_counts_df)[1]<-"date_of_each_month"
59 names(Date_counts_df)[2]<-"Count"
60 #Date_counts_df
61
62
63 #Task02-Provide the trend chart for the number of complaints at monthly an
64
65 #Trend Chart for Monthly count
66 ggplot(data=months_counts_df, aes(x=Months, y=Count, label=Count, group=1))
67   geom_line(color="green")+
68   geom_point(size = 0.8)+geom_text()+
69   labs(title = "Monthly Ticket Count",x= "Months",y = "No. of Tickets")
70 |
71
```

70:1 (Top Level) R Script

Environment History Connections Tutorial



Comcast Telecom Complaints data.R*

comcast_data

```

53 Date_counts <- table(comcast_data$date_of_each_month)
54
55 #Date_counts
56 Date_counts_df <- as.data.frame(Date_counts)
57
58 names(Date_counts_df)[1] <- "date_of_each_month"
59 names(Date_counts_df)[2] <- "Count"
60 #Date_counts_df
61
62
63 #Task02-Provide the trend chart for the number of complaints at monthly and daily granularity levels.
64
65 #Trend Chart for Monthly count
66 ggplot(data=months_counts_df, aes(x=Months, y=Count, label=Count, group=1)) +
67   geom_line(color="green")+
68   geom_point(size = 0.8)+geom_text()+
69   labs(title = "Monthly Ticket Count",x= "Months",y = "No. of Tickets")
70
71 #Trend Chart for Daily count
72 ggplot(data=Date_counts_df, aes(x=date_of_each_month, y=Count,label=Count, group=1)) +
73   geom_line(color="blue")+
74   geom_point(size = 0.8)+
75   geom_text()+labs(title = "Daily Ticket Count",x= "Dates of Each Month",y = "No. of Tickets")
76
77 #Task03-Provide a table with the frequency of complaint types.
78 comcast_data$Customer.Complaint
79 complaints_count <- table(comcast_data$Customer.Complaint)
80
81
82 network_tickets <- data.frame(contains(comcast_data$Customer.Complaint,match = 'network',ignore.case = T))
83 internet_tickets <- data.frame(contains(comcast_data$Customer.Complaint,match = 'internet',ignore.case = T))
84 billing_tickets <- data.frame(contains(comcast_data$Customer.Complaint,match = 'bill',ignore.case = T))
85 datacap_tickets <- data.frame(contains(comcast_data$Customer.Complaint,match = 'data cap',ignore.case = T))
86 customerservice_tickets <- data.frame(contains(comcast_data$Customer.Complaint,match = 'customer service',ignore
87
88 nrow(network_tickets)
89 nrow(internet_tickets)
90 nrow(billing_tickets)
91 nrow(datacap_tickets)
92 nrow(customerservice_tickets)
93

```

Console Terminal

R422: ~/

```

5 maps and customer service representative hung up call repeatedly"
[987] "Long term billing issue with rude customer service caused depression and trauma"
[988] "Comcast incorrect and confusing billing"
[989] "Data caps & Monthly prices"
[990] "Installation Request - New Subdivision - Jesup, GA"
[991] "Comcast Grievance"
[992] "Comcast -Exfinity customer service errors, lies and waste d time"
[993] "Comcast -Exfinity customer service errors, lies and waste d time"
[994] "Comcast Very Bad /Rude customer service"
[995] "Billing Dispute"
[996] "Comcast Cable"
[997] "LIED TO!!! Now I'm suffering?!?! And at a loss!!!"
[998] "cyber bullying"
[999] "Comcast Service"
[1000] "Comcast Business"
[ reached getOption("max.print") -- omitted 1224 entries ]
> complaints_count <- table(comcast_data$Customer.Complaint)
> network_tickets <- data.frame(contains(comcast_data$Customer.Complaint,match = 'network',ignore.case = T))
> internet_tickets <- data.frame(contains(comcast_data$Customer.Complaint,match = 'internet',ignore.case = T))
> billing_tickets <- data.frame(contains(comcast_data$Customer.Complaint,match = 'bill',ignore.case = T))
> datacap_tickets <- data.frame(contains(comcast_data$Customer.Complaint,match = 'data cap',ignore.case = T))
> customerservice_tickets <- data.frame(contains(comcast_data$Customer.Complaint,match = 'customer service',ignore.case = T))
> nrow(network_tickets)
[1] 2
> nrow(internet_tickets)
[1] 532
> nrow(billing_tickets)
[1] 379
> nrow(datacap_tickets)
[1] 150
> nrow(customerservice_tickets)
[1] 78
>

```



```
Comcast Telecom Complaints data.R* comcast_data
74 geom_point(size = 0.8)+
75 geom_text()+labs(title = "Daily Ticket Count",x= "Dates of Each Month",y ="No. of Tickets")
76
77 #Task03-Provide a table with the frequency of complaint types.
78 comcast_data$Customer.Complaint
79 complaints_count<-table(comcast_data$Customer.Complaint)
80
81
82 network_tickets<- data.frame(contains(comcast_data$Customer.Complaint,match = 'network',ignore.case = T))
83 internet_tickets<- data.frame(contains(comcast_data$Customer.Complaint,match = 'internet',ignore.case = T))
84 billing_tickets<- data.frame(contains(comcast_data$Customer.Complaint,match = 'bill',ignore.case = T))
85 datacap_tickets<- data.frame(contains(comcast_data$Customer.Complaint,match = 'data cap',ignore.case = T))
86 customerservice_tickets<- data.frame(contains(comcast_data$Customer.Complaint,match = 'customer service',ignore.case = T))
87
88 nrow(network_tickets)
89 nrow(internet_tickets)
90 nrow(billing_tickets)
91 nrow(datacap_tickets)
92 nrow(customerservice_tickets)
93
94 #Task04-Which complaint types are maximum i.e., around internet, network issues, or across any other domains.
95 #Create a new categorical variable with value as Open and Closed. Open & Pending is to be categorized as Open.
96 #Provide state wise status of complaints in a stacked bar chart. Use the categorized variable from Q3. Provide
97
98
99 #Complaint Type that has maximum Tickets
100 if(nrow(network_tickets) > nrow(internet_tickets)){
101   print("Network Issues")
102 } else if(nrow(internet_tickets) > nrow(billing_tickets)){
103   print("Internet Issues")
104 } else if(nrow(billing_tickets) > nrow(datacap_tickets)){
105   print("Billing Issues")
106 } else if(nrow(datacap_tickets) > nrow(customerservice_tickets)){
107   print("Data cap Issues")
108 } else {
109   print("Customer Service Issues")
110 }
111 |
112
113
114
115
111.1 (Top Level) :- R Script
```

```
Console Terminal
R 4.2.2 - ~/
[996] Comcast Cable
[997] "LIED TO!!! Now I'm suffering?!?! And at a loss!!!"
[998] "cyber bullying"
[999] "Comcast Service"
[1000] "Comcast Business"
[ reached getOption("max.print") -- omitted 1224 entries ]
> complaints_count<-table(comcast_data$Customer.Complaint)
> network_tickets<- data.frame(contains(comcast_data$Customer.Complaint,match = 'network',ignore.case = T))
> internet_tickets<- data.frame(contains(comcast_data$Customer.Complaint,match = 'internet',ignore.case = T))
> billing_tickets<- data.frame(contains(comcast_data$Customer.Complaint,match = 'bill',ignore.case = T))
> datacap_tickets<- data.frame(contains(comcast_data$Customer.Complaint,match = 'data cap',ignore.case = T))
> customerservice_tickets<- data.frame(contains(comcast_data$Customer.Complaint,match = 'customer service',ignore.case = T))
> nrow(network_tickets)
[1] 2
> nrow(internet_tickets)
[1] 532
> nrow(billing_tickets)
[1] 379
> nrow(datacap_tickets)
[1] 150
> nrow(customerservice_tickets)
[1] 78
> #Complaint Type that has maximum Tickets
> if(nrow(network_tickets) > nrow(internet_tickets)){
+   print("Network Issues")
+ } else if(nrow(internet_tickets) > nrow(billing_tickets)){
+   print("Internet Issues")
+ } else if(nrow(billing_tickets) > nrow(datacap_tickets)){
+   print("Billing Issues")
+ } else if(nrow(datacap_tickets) > nrow(customerservice_tickets)){
+   print("Data cap Issues")
+ } else {
+   print("Customer Service Issues")
+ }
[1] "Internet Issues"
>
```



```

Comcast Telecom Complaints data.R comcast_data
Source on Save
85 datacap_tickets<- data.frame(contains(comcast_data$Customer.Complaint,match = 'data cap',ignore.case = T))
86 customerservice_tickets<- data.frame(contains(comcast_data$Customer.Complaint,match = 'customer service',ignore
87
88 nrow(network_tickets)
89 nrow(internet_tickets)
90 nrow(billing_tickets)
91 nrow(datacap_tickets)
92 nrow(customerservice_tickets)
93
94 #Task04-Which complaint types are maximum i.e., around internet, network issues, or across any other domains.
95 #Create a new categorical variable with value as Open and Closed. Open & Pending is to be categorized as Open
96 #Provide state wise status of complaints in a stacked bar chart. Use the categorized variable from Q3. Provide
97
98
99 #Complaint Type that has maximum Tickets
100 if(nrow(network_tickets) > nrow(internet_tickets)){
101   print("Network Issues")
102 } else if(nrow(internet_tickets) > nrow(billing_tickets)){
103   print("Internet Issues")
104 } else if(nrow(billing_tickets) > nrow(datacap_tickets)){
105   print("Billing Issues")
106 } else if(nrow(datacap_tickets) > nrow(customerservice_tickets)){
107   print("Data cap Issues")
108 } else {
109   print("Customer Service Issues")
110 }
111
112
113 #Open and Pending Statuses are considered as "Open"
114 my_data <- as_tibble(comcast_data$Status)
115 Open_complaints <- my_data %>% filter(value=="Open" | value=="Pending")
116 comcast_data<-subset(comcast_data,select=-c(ComplaintStatus))
117
118 comcast_data$Status<-gsub('Pending', 'Open', comcast_data$Status)
119
120 Open_complaints<-(comcast_data$Status == "Open")
121 comcast_data$Complaint_Status[Open_complaints]<-"Open"
122 |
123
124
125
126

```

```

R 4.2.2 - /
> billing_tickets<- data.frame(contains(comcast_data$Customer.Com
plaint,match = 'bill',ignore.case = T))
> datacap_tickets<- data.frame(contains(comcast_data$Customer.Com
plaint,match = 'data cap',ignore.case = T))
> customerservice_tickets<- data.frame(contains(comcast_data$Cust
omer.Complaint,match = 'customer service',ignore.case = T))
> nrow(network_tickets)
[1] 2
> nrow(internet_tickets)
[1] 532
> nrow(billing_tickets)
[1] 379
> nrow(datacap_tickets)
[1] 150
> nrow(customerservice_tickets)
[1] 78
> #Complaint Type that has maximum Tickets
> if(nrow(network_tickets) > nrow(internet_tickets)){
+   print("Network Issues")
+ } else if(nrow(internet_tickets) > nrow(billing_tickets)){
+   print("Internet Issues")
+ } else if(nrow(billing_tickets) > nrow(datacap_tickets)){
+   print("Billing Issues")
+ } else if(nrow(datacap_tickets) > nrow(customerservice_ticket
s)){
+   print("Data cap Issues")
+ } else {
+   print("Customer Service Issues")
+ }
[1] "Internet Issues"
> #Open and Pending Statuses are considered as "Open"
> my_data <- as_tibble(comcast_data$Status)
> Open_complaints <- my_data %>% filter(value=="Open" | value=="P
ending")
> comcast_data<-subset(comcast_data,select=-c(ComplaintStatus))
Error in eval(substitute(select), nl, parent.frame()) :
  object 'ComplaintStatus' not found
> comcast_data$Status<-gsub('Pending', 'Open', comcast_data$Statu
s)
> Open_complaints<-(comcast_data$Status == "Open")
> comcast_data$Complaint_Status[Open_complaints]<-"Open"
>

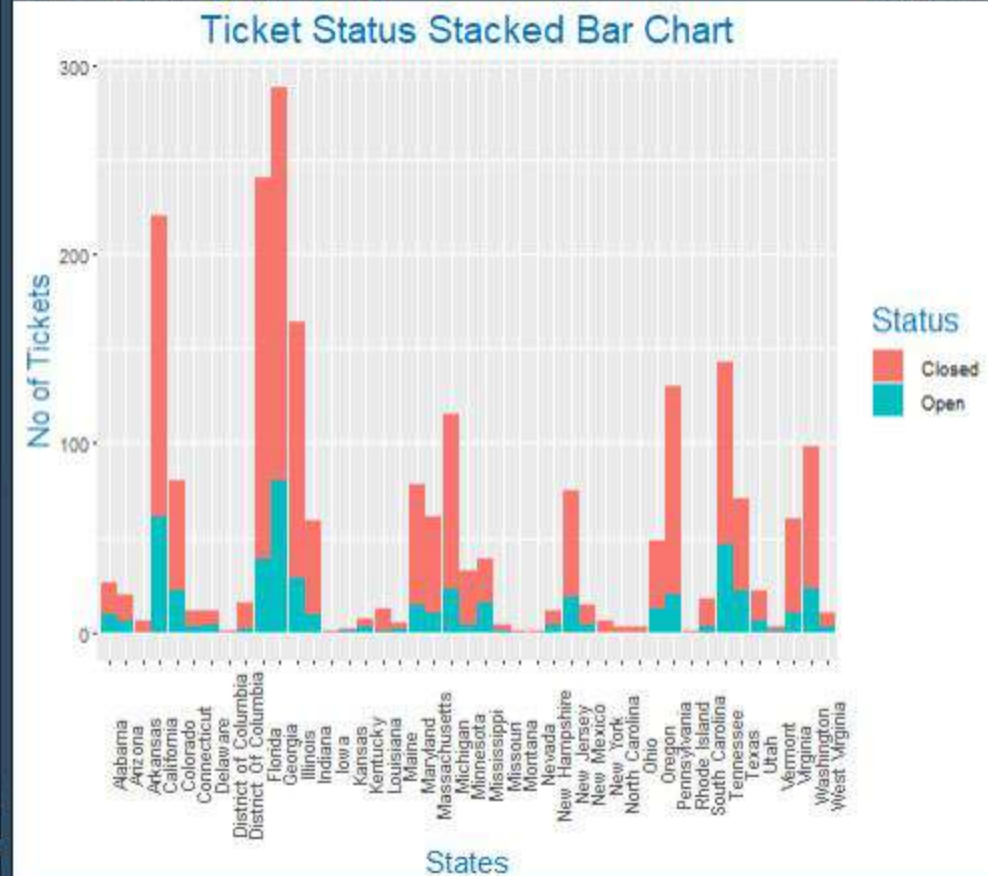
```



```
> #Stacked Bar chart for Open and Closed Complaints
> comcast_data<- group_by(comcast_data,State,Complaint_Status)
> chart_data<- dplyr::summarise(comcast_data,Count = n())
`summarise()` has grouped output by 'State'. You can override
using the `groups` argument.
> ggplot(as.data.frame(chart_data) ,mapping = aes(State,Count))+
+   geom_col(aes(fill = Complaint_Status),width = 0.95)+
+   theme(axis.text.x = element_text(angle = 90),
+         axis.title.y = element_text(size = 15),
+         axis.title.x = element_text(size = 15),
```

Files Plots Packages Help Viewer Presentation

Zoom Export Publish




```

116 comcast_data<-subset(comcast_data,select=~c(ComplaintStatus))
117
118 comcast_data$Status<-gsub('Pending', 'Open', comcast_data$Status)
119
120 Open_complaints<-(comcast_data$Status == "Open")
121 comcast_data$Complaint_Status[Open_complaints]<-"Open"
122
123 #Solved and Closed Statuses are considered as "Closed"
124 comcast_data$Status<-gsub('Solved','Closed', comcast_data$Status)
125
126
127 Closed_complaints<-(comcast_data$Status == "Closed")
128 comcast_data$Complaint_Status[Closed_complaints]<-"Closed"
129
130
131 #Stacked Bar chart for Open and Closed Complaints
132 comcast_data<- group_by(comcast_data,State,Complaint_Status)
133 chart_data<- dplyr::summarise(comcast_data,Count = n())
134
135 ggplot(as.data.frame(chart_data) ,mapping = aes(State,Count))+
136   geom_col(aes(fill = Complaint_Status),width = 0.95)+
137   theme(axis.text.x = element_text(angle = 90),
138         axis.title.y = element_text(size = 15),
139         axis.title.x = element_text(size = 15),
140         title = element_text(size = 16,colour = "#0073C2FF"),
141         plot.title = element_text(hjust = 0.5))+
142   labs(title = "Ticket Status Stacked Bar Chart",
143        x = "States",y = "No of Tickets",
144        fill= "Status")
145
146
147 #Task05|- Which state has the maximum complaints
148 chart_data%>% filter(Complaint_Status=="Open")-> Open_complaints
149
150 max(Open_complaints$Count)
151 (Open_complaints)[1]
152 Open_complaints[Open_complaints$Count == max(Open_complaints$Count),c(1,3)]
153
154
155
156

```

summarise() has grouped output by State. You can override using the '.groups' argument.

```

> ggplot(as.data.frame(chart_data) ,mapping = aes(State,Count))+
+   geom_col(aes(fill = Complaint_Status),width = 0.95)+
+   theme(axis.text.x = element_text(angle = 90),
+         axis.title.y = element_text(size = 15),
+         axis.title.x = element_text(size = 15),
+         title = element_text(size = 16,colour = "#0073C2FF"),
+         plot.title = element_text(hjust = 0.5))+
+   labs(title = "Ticket Status Stacked Bar Chart",
+        x = "States",y = "No of Tickets",
+        fill= "Status")
> #State that has the maximum complaints
> chart_data%>% filter(Complaint_Status=="Open")-> Open_complaints
> max(Open_complaints$Count)
[1] 80
> (Open_complaints)[1]
# A tibble: 34 x 1
# Groups:   State [34]
  State
  <chr>
1 Alabama
2 Arizona
3 California
4 Colorado
5 Connecticut
6 Delaware
7 District Of Columbia
8 Florida
9 Georgia
10 Illinois
# ... with 24 more rows
# i Use 'print(n = ...)' to see more rows
> Open_complaints[Open_complaints$Count == max(Open_complaints$Count),c(1,3)]
# A tibble: 1 x 2
# Groups:   State [1]
  State Count
  <chr>   <int>
1 Georgia    80

```



```

131 #Stacked Bar chart for Open and Closed Complaints
132 comcast_data<- group_by(comcast_data,State,Complaint_Status)
133 chart_data<- dplyr::summarise(comcast_data,Count = n())
134
135 ggplot(as.data.frame(chart_data) ,mapping = aes(State,Count))+
136   geom_col(aes(fill = Complaint_Status),width = 0.95)+
137   theme(axis.text.x = element_text(angle = 90),
138         axis.title.y = element_text(size = 15),
139         axis.title.x = element_text(size = 15),
140         title = element_text(size = 16,colour = "#0073C2FF"),
141         plot.title = element_text(hjust = 0.5))+
142   labs(title = "Ticket Status Stacked Bar Chart",
143        x = "States",y = "No of Tickets",
144        fill= "Status")
145
146
147 #Task05-Which state has the maximum complaints
148 chart_data%>% filter(Complaint_Status=="Open")-> Open_complaints
149
150 max(Open_complaints$Count)
151 (Open_complaints)[1]
152 Open_complaints[Open_complaints$Count == max(Open_complaints$Count),c(1,3)]
153
154 #Complaints which were received through the Internet and customer care calls.
155 #Task06-Which state has the highest percentage of unresolved complaints
156 # -Provide the percentage of complaints resolved till date, which were received through the
157 # -Internet and customer care calls.
158
159 Resolved_data<-group_by(comcast_data,Complaint_Status)
160 Total_resolved<-dplyr::summarise(Resolved_data ,percentage = (n()/nrow(Resolved_data)))
161 #Total_resolved
162 Resolved_data1 <- group_by(comcast_data,Received.Via,Complaint_Status)
163 Category_resolved<-dplyr::summarise(Resolved_data1,percentage =(n()/nrow(Resolved_data)))
164 #Category_resolved
165
166
167
168
169
170
171

```

```

R422 - /
+ plot.title = element_text(hjust = 0.5))
+   labs(title = "Ticket Status Stacked Bar Chart",
+        x = "States",y = "No of Tickets",
+        fill= "Status")
> #State that has the maximum complaints
> chart_data%>% filter(Complaint_Status=="Open")-> Open_complaints
> max(Open_complaints$Count)
[1] 80
> (Open_complaints)[1]
# A tibble: 34 x 1
# Groups:   State [34]
  State
  <chr>
1 Alabama
2 Arizona
3 California
4 Colorado
5 Connecticut
6 Delaware
7 District Of Columbia
8 Florida
9 Georgia
10 Illinois
# ... with 24 more rows
# i Use `print(n = ...)` to see more rows
> Open_complaints[Open_complaints$Count == max(Open_complaints$Count),c(1,
3)]
# A tibble: 1 x 2
# Groups:   State [1]
  State Count
  <chr>   <int>
1 Georgia    80
> Resolved_data<-group_by(comcast_data,Complaint_Status)
> Total_resolved<-dplyr::summarise(Resolved_data ,percentage = (n()/nrow(R
esolved_data)))
> #Total_resolved
> Resolved_data1 <- group_by(comcast_data,Received.Via,Complaint_Status)
> Category_resolved<-dplyr::summarise(Resolved_data1,percentage =(n()/nrow
(Resolved_data)))
`summarise()` has grouped output by 'Received.Via'. You can
override using the `.groups` argument.
>

```


RStudio

File Edit Code View Plots Session Build Debug Profile Tools Help

Go to file/function Addins

Comcast Telecom Complaints data.R comcast_data

```
160 Total_resolved<-dplyr::summarise(Resolved_data ,percentage = (n()/nrow(Resolved_data)))
161 #Total_resolved
162 Resolved_data1 <- group_by(comcast_data,Received.Via,Complaint_Status)
163 Category_resolved<-dplyr::summarise(Resolved_data1,percentage =(n()/nrow(Resolved_data)))
164 #Category_resolved
165
166
167 #Pie Chart for Category wise Ticket Status
168 #*****
169
170 par(mfrow = c(1,2))
171 total<-ggplot(data=Total_resolved,
172             aes(x= "",y =percentage,fill = Complaint_Status))+
173   geom_bar(stat = "identity",width = 1)+
174   coord_polar("y",start = 0)+
175   geom_text(aes(label = paste0(round(percentage*100),"%")),
176           position = position_stack(vjust = 0.5))+
177   labs(title = "Pie Chart based on Ticket Status",x = NULL,y = NULL,fill = NULL)+
178   theme_classic()+theme(axis.line = element_blank(),
179                         axis.text = element_blank(),
180                         axis.ticks = element_blank())
181
182
183 category<-ggplot(data=Category_resolved,
184                 aes(x= "",y =percentage,fill = Complaint_Status))+
185   geom_bar(stat = "identity",width = 1)+
186   coord_polar("y",start = 0)+
187   geom_text(aes(label = paste0(Received.Via,"",round(percentage*100),"%")),
188           position = position_stack(vjust = 0.5))+
189   labs(title = "Pie Chart for Category wise Ticket Status",x = NULL,y = NULL,fill = NULL)+
190   theme_classic()+theme(axis.line = element_blank(),
191                         axis.text = element_blank(),
192                         axis.ticks = element_blank())
193
194 ggarrange(total,category,nrow = 1, ncol = 2)
195
196
197
198
199
200
201
```

194:45 (Top Level) R Script

Environment History Connections Tutorial

Project: (None)

R422

```
> Resolved_data<-group_by(comcast_data,Complaint_Status)
> Total_resolved<-dplyr::summarise(Resolved_data ,percentage = (n()/nrow(Resolved_data)))
> #Total_resolved
> Resolved_data1 <- group_by(comcast_data,Received.Via,Complaint_Status)
> Category_resolved<-dplyr::summarise(Resolved_data1,percentage =(n()/nrow(Resolved_data)))
```

Files Plots Packages Help Viewer Presentation

Zoom Export Publish

Pie Chart based on Ticket Status

Status	Count
Closed	77
Open	23

Pie Chart for Category wise Ticket Status

Received.Via	Status	Count
Customer Care Call	Closed	39
Customer Care Call	Open	11
Internet	Closed	38
Internet	Open	12