**Project Submission Guidelines for Data Science with R**

Hi Learner,

You need to Submit any 1 project out of following -

Insurance, Healthcare, Internet, Retail, E-commerce, Education

Kindly follow the below steps to draft the Project Report :

1. Project Report has to be created in a “single” word document/ PPT / PDF

2. Business Scenario

3. Expectation /Goals

4. Code

5. Output Screenshot

6. Analysis

Prepare the document with your own code, output screenshots and analysis (a small description on your analysis based on the output obtained).

Once done you are required to submit the same file following the path mentioned below:  
LMS > Course Icon > Project > Submit.

\*Note – Submit the same single file (word doc, PDF or PPT) in all the 3 fields i.e.   
 **Writeup, Screenshots, Source Code**

All the very best for your Project and Course Completion.

COMCAST TELECOM CONSUMER COMPLAINTS

Name: Luke Nonyane

Date: August 29, 2021

Business Scenario

* Comcast is an American global telecommunication company. The firm has been providing terrible customer service. In October 2016 the authority fined them $2.3 million, after receiving over 1000 customer complaints. Pin down what is wrong with Comcast’s customer service by finding out the complaint types and how to reduce prevalence.

Expectation/Goals

* Perform analysis tasks to uncover the trends hidden in the data in order to provide actionable insights to the decision makers.

Code

* Code is at the end.

Output Screenshot

* Import data into R environment.
  + customer\_complaints = read.csv("Comcast Telecom Complaints data1.csv", sep=",")
* Provide the trend chart for the number of complaints at monthly and daily granularity levels.
  + Chart, line chart

    Description automatically generated
  + Chart, line chart

    Description automatically generated
* Provide a table with the frequency of complaint types.
  + Table has 1841 distinct complaints. Will provide clip of sample view. The rest can be seen when the script is run.
  + We can see that the month of June recorded the highest customer complaints for the year 2015. One assumption to the spike can be that this is peak moving season so a lot of accounts need to be moved or are due for renewal.
  + Table

    Description automatically generated
* 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.
  + Graphical user interface, table

    Description automatically generated
* Which state has the maximum complaints?
  + The state of GA has 288 complaints.
  + Table

    Description automatically generated with medium confidence
* Which state has the highest percentage of unresolved complaints
  + Graphical user interface

    Description automatically generated with medium confidence
  + The state of Kentucky has the highest percentage of unresolved cases.
* Provide the percentage of complaints resolved till date, which were received through the Internet and customer care calls.
* Graphical user interface, text, application

  Description automatically generated
* About 77% of the complaints resolved till date where received through ‘Internet’ and ‘Customer Care Calls’.

Summary

* As we can see from the analysis tasks performed the month of June experienced an unorthodox amount of complaints; over 1000. One assumption that could have contributed to this large number is the season of the year. During the summer is when most apartment leases are due for renewal. Tenants either choose to renew or look for another property to rent. If they are moving they will need to transfer their service to another property else renew current terms with comcast. We can use the data to address peak times of calls by having more staff work during the season. Clients will be able to get their complaints heard without having to be on hold for long durations of times. The services offered by Comcast need to improve as well; The infrastructure around performance complaints needs upgrades from laying down new cables (fibre) to the modems and routers used to access the network. If the same complaints keep coming back customers will switch to competitors and comcast will keep getting hit fines.

SCRIPT

getwd()

customer\_complaints = read.csv("Comcast Telecom Complaints data1.csv", sep=",")

View(customer\_complaints)

str(customer\_complaints)

# convert Status, Received.Via, Filing.on.Behalf.of.Someone to factor from char

install.packages("plyr")

library(plyr)

customer\_complaints$Received.Via <- sapply(customer\_complaints$Received.Via, factor)

customer\_complaints$Status <- sapply(customer\_complaints$Status, factor)

customer\_complaints$Filing.on.Behalf.of.Someone <- sapply(customer\_complaints$Filing.on.Behalf.of.Someone, factor)

# Monthly trend chart for number of complaints y

# https://r-coder.com/plot-r/

# x=date y=complaints

# Format date column to be the same. https://www.stat.berkeley.edu/~s133/dates.html

customer\_complaints$Date

# https://www.rdocumentation.org/packages/lubridate/versions/1.7.10

install.packages("lubridate")

library(lubridate)

v1 <- as.POSIXlt((parse\_date\_time(customer\_complaints$Date, c('dmy'))))

complaint\_months <- month(v1)

complaint\_months

# Set complaint\_months to factor (effectively binning)

#complaint\_months <- sapply(complaint\_months, factor)

#complaint\_months

# Create function that records month of complaint and assigns to correct month.

# We can see the distinct months in our records are 1-12 Jan-Dec.

complaint\_months\_to\_char <- sapply(complaint\_months, character)

complaint\_months\_to\_char

count\_of\_Jan <- 0

count\_of\_Feb <- 0

count\_of\_Mar <- 0

count\_of\_Apr <- 0

count\_of\_May <- 0

count\_of\_Jun <- 0

count\_of\_Jul <- 0

count\_of\_Aug <- 0

count\_of\_Sep <- 0

count\_of\_Oct <- 0

count\_of\_Nov <- 0

count\_of\_Dec <- 0

for(item in complaint\_months){

if(item == "1") count\_of\_Jan <- count\_of\_Jan + 1

if(item == "2") count\_of\_Feb <- count\_of\_Feb + 1

if(item == "3") count\_of\_Mar <- count\_of\_Mar + 1

if(item == "4") count\_of\_Apr <- count\_of\_Apr + 1

if(item == "5") count\_of\_May <- count\_of\_May + 1

if(item == "6") count\_of\_Jun <- count\_of\_Jun + 1

if(item == "7") count\_of\_Jul <- count\_of\_Jul + 1

if(item == "8") count\_of\_Aug <- count\_of\_Aug + 1

if(item == "9") count\_of\_Sep <- count\_of\_Sep + 1

if(item == "10") count\_of\_Oct <- count\_of\_Oct + 1

if(item == "11") count\_of\_Nov <- count\_of\_Nov + 1

if(item == "12") count\_of\_Dec <- count\_of\_Dec + 1

}

count\_of\_Jan

count\_of\_Feb

count\_of\_Mar

count\_of\_Apr

count\_of\_May

count\_of\_Jun

count\_of\_Jul

count\_of\_Aug

count\_of\_Sep

count\_of\_Oct

count\_of\_Nov

count\_of\_Dec

# Create month name bank to be used for the plot

month\_name\_bank <- c("Jan", "Feb", "Mar", "Apr", "May", "Jun", "Jul", "Aug", "Sep", "Oct", "Nov", "Dec")

# Provide the trend chart for the number of complaints at monthly granularity levels.

#x <- c("Jan", "Feb", "Mar", "Apr", "May", "Jun", "Jul", "Aug", "Sep", "Oct", "Nov", "Dec")

x <- c(1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12)

x

y <- c(count\_of\_Jan, count\_of\_Feb, count\_of\_Mar, count\_of\_Apr, count\_of\_May, count\_of\_Jun, count\_of\_Jul, count\_of\_Aug, count\_of\_Sep, count\_of\_Oct, count\_of\_Nov, count\_of\_Dec)

y

plot(x, y, xlab="Months", ylab="Number of complaints", type="b", main="Monthly Trend Chart of Customer Complaints", xlim=c(1,12), ylim=c(0,1500))

# https://stackoverflow.com/questions/13229546/how-can-i-label-points-in-this-scatterplot

text(x, y, labels = month\_name\_bank, cex = 0.7, pos=3)

# Provide the trend chart for the number of complaints at daily granularity levels

v1$wday

# We can see our days of the week range from 0-6 (Sun-Sat)

# Store days of week in variable days\_of\_week

days\_of\_week <- v1$wday

days\_of\_week

# Iterate through days\_of\_week and sum the distinct occurrence of week day

# Define count variables of each day of the week and assign 0 as the value.

#Sun-Sat:0-6

count\_Sun <- 0

count\_Mon <- 0

count\_Tue <- 0

count\_Wed <- 0

count\_Thur <- 0

count\_Fri <- 0

count\_Sat <- 0

for(item in days\_of\_week){

if(item == 0) count\_Sun <- count\_Sun + 1

if(item == 1) count\_Mon <- count\_Mon + 1

if(item == 2) count\_Tue <- count\_Tue + 1

if(item == 3) count\_Wed <- count\_Wed + 1

if(item == 4) count\_Thur <- count\_Thur + 1

if(item == 5) count\_Fri <- count\_Fri + 1

if(item == 6) count\_Sat <- count\_Sat + 1

}

count\_Sun

count\_Mon

count\_Tue

count\_Wed

count\_Thur

count\_Fri

count\_Sat

name\_of\_days <- c("Sun", "Mon", "Tue", "Wed", "Thur", "Fri", "Sat")

# Plot x(Days of the week from 0-6) y(count of each day of the week)

x1 <- c(0 ,1, 2, 3, 4, 5, 6)

y1 <- c(count\_Sun, count\_Mon, count\_Tue, count\_Wed, count\_Thur, count\_Fri, count\_Sat)

plot(x1, y1, xlab="Days of week", ylab="Number of complaints", type="b", main="Daily Trend Chart of Customer Complaints", xlim=c(0,6), ylim=c(0,500))

text(x1, y1, labels = name\_of\_days, cex = 0.7, pos=3)

############################################################################

# Provide a table with the frequency of complaint types.

# https://www.statology.org/create-table-in-r/

complaint\_types <- sapply(customer\_complaints$Customer.Complaint, factor)

plyr::count(complaint\_types) # These are the distinct complaint types; 1841

complaint\_types\_table <- plyr::count(complaint\_types)

View(complaint\_types\_table)

############################################################################

# 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.

trial\_data = read.csv("Comcast Telecom Complaints data1.csv", sep=",")

trial\_df <- tibble(trial\_data)

head(trial\_df)

install.packages("dplyr")

library(dplyr)

trial\_df <- trial\_df %>%

mutate(condensed\_status = case\_when(Status == 'Open' | Status == 'Pending' ~ 'Open',

Status == 'Solved' | Status == 'Closed' ~ 'Closed'))

View(trial\_df)

############################################################################

## Provide state wise status of complaints in a stacked bar chart. Use the categorized variable from Q3##

# How many different states do we have?

sapply(customer\_complaints$State, factor)

# Colors for the legend

# showCols(bg="gray20",cl=colors()[1:60], rot=30, cex=0.9)

# showCols(cl= colors(), bg="gray33", rot=30, cex=0.75)

# View(showCols(cl= colors(), bg="gray33", rot=30, cex=0.75))

# colors <- showCols(cl = colors(), bg = "gray33", rot = 30, cex = 0.75)[["label"]]

statewise\_complaints <- table(trial\_df$State,trial\_df$condensed\_status)

barplot(statewise\_complaints,

main="Statewise complaints",

col=c("grey", "cornflowerblue", "aliceblue", "bisque2", "blueviolet", "brown4", "burlywood4", "cadetblue4",

"chartreuse4", "chocolate4", "coral4", "cornsilk3", "cyan3", "darkgoldenrod1", "darkgreen", "darkolivegreen1",

"darkorange1", "darkorchid1", "darksalmon", "darkseagreen4", "darkslategray3", "deeppink", "deepskyblue",

"dimgray", "dodgerblue3", "firebrick3", "gold4", "goldenrod4", "gray3", "honeydew4", "hotpink4", "indianred4",

"ivory4", "khaki4", "lavenderblush3", "lemonchiffon2", "lightblue2", "lightcyan1", "lightgoldenrod1",

"lightyellow3", "magenta1", "maroon1", "mediumblue", "mediumorchid4"),

legend.text = TRUE,

args.legend = list(x = "topright",

inset = c(-0.009, -0.1),

cex=0.265))

############################################################################

# Which state has the maximum complaints

open\_complaints <- statewise\_complaints[,"Open"]

closed\_complaints <- statewise\_complaints[,"Closed"]

View(open\_complaints)

View(closed\_complaints)

states <- unique(trial\_df$State)

# For loop to sum up each of the states' open + closed tickets

i <- 1

for (i in states) {

print(i)

print(as.numeric(open\_complaints[i]) + as.numeric(closed\_complaints[i]))

}

# The state of Georgia has the maximum complaints

############################################################################

## Which state has the highest percentage of unresolved complaints ##

# For loop to find the percentages of unresolved complaints

e <- 1

for (e in states) {

print(e)

print("Total complaints")

print(as.numeric(open\_complaints[e]) + as.numeric(closed\_complaints[e]))

print("Percent of unresolved complaints:")

print( as.numeric(open\_complaints[e]) / (as.numeric(open\_complaints[e]) + as.numeric(closed\_complaints[e])))

}

# The state of Kentucky has the highest percentage of unresolved complaints at 0.4285714 ~ 43%

############################################################################

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

# Use trial\_df columns condensed\_status & received.via

# Collect all complaints received through Internet & Customer care calls

calls\_received\_through\_internet\_and\_ccarecalls <- table(trial\_df$Received.Via, trial\_df$condensed\_status)

View(calls\_received\_through\_internet\_and\_ccarecalls)

# From the view above we can see the closed tickets for customer care calls = 864 and internet = 843

calls\_received\_through\_internet\_and\_ccarecalls[,"Closed"]

ccarecalls\_internet\_closed <- calls\_received\_through\_internet\_and\_ccarecalls[,"Closed"]

ccarecalls\_internet\_open <- calls\_received\_through\_internet\_and\_ccarecalls[,"Open"]

ccarecalls\_internet\_closed

ccarecalls\_internet\_open

ccarecall\_closed <- ccarecalls\_internet\_closed["Customer Care Call"]

ccarecall\_closed

internet\_closed <- ccarecalls\_internet\_closed["Internet"]

internet\_closed

ccarecall\_open <- ccarecalls\_internet\_open["Customer Care Call"]

ccarecall\_open

internet\_open <- ccarecalls\_internet\_open["Internet"]

internet\_open

total\_tickets <- as.numeric(ccarecall\_closed) + as.numeric(internet\_closed) + as.numeric(ccarecall\_open) + as.numeric(internet\_open)

total\_tickets

total\_resolved\_through\_ccarecall\_internet <- as.numeric(ccarecall\_closed) + as.numeric(internet\_closed)

total\_resolved\_through\_ccarecall\_internet

percentage\_resolved\_through\_ccarecall\_internet <- total\_resolved\_through\_ccarecall\_internet / total\_tickets

percentage\_resolved\_through\_ccarecall\_internet

############################################################################