Assignment-2(BA)

Dev

2023-10-08

Setting directory for using online_retail.csv datset

```
setwd("/Users/devmarwah/Downloads")
#Reading the csv file
df=read.csv("Online_Retail.csv")
library(tidyverse)
```

```
-- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
v dplyr     1.1.3     v readr     2.1.4
v forcats   1.0.0     v stringr   1.5.0
v ggplot2   3.4.3     v tibble    3.2.1
v lubridate 1.9.2     v tidyr     1.3.0
v purrr     1.0.2
-- Conflicts ------------------------------------------ tidyverse_conflicts() --
x dplyr::filter() masks stats::filter()
x dplyr::lag()    masks stats::lag()
i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

Q-1: Show the breakdown of the number of transactions by countries i.e., how many transactions are in the dataset for each country (consider all records including cancelled transactions). Show this in total number and also in percentage. Show only countries accounting for more than 1% of the total transactions

```
#Using table command to get total transactions bycountries
total.number=(table(df$Country))
#Using prop.table and then multiplying by 100 to get values in percentages
df1=prop.table(total.number)
percentage=round(100*df1,digits = 2)
answer=cbind(total.number,percentage)
answer=as.data.frame(answer)
#Using filter to get only percentage>1
answer=answer %>%
  filter(percentage>1)
answer
```

```
               total.number percentage
EIRE                   8196       1.51
France                 8557       1.58
Germany                9495       1.75
United Kingdom       495478      91.43
```

Q-2:Create a new variable 'TransactionValue' that is the product of the exising 'Quantity' and 'UnitPrice' variables. Add this variable to the dataframe.

```
#Using mutate function to create new variable
df=df %>%
  mutate(Transactionvalue=Quantity*UnitPrice)
```

Q-3:Using the newly created variable, TransactionValue, show the breakdown of transaction values by countries i.e. how much money in total has been spent each country. Show this in total sum of transaction values. Show only countries with total transaction exceeding 130,000 British Pound

```
  df %>%
  #Grouping by countries and then summarising sum of transaction values
  group_by(Country) %>%
  summarise(Total.spending=sum(Transactionvalue)) %>%
  #Using filter command to show countries with total.spending > 130,000 Pounds
  filter(Total.spending>130000)
```

```
# A tibble: 6 x 2
  Country        Total.spending
  <chr>                   <dbl>
1 Australia             137077.
2 EIRE                  263277.
3 France                197404.
4 Germany               221698.
5 Netherlands           284662.
6 United Kingdom       8187806.
```

Doing preparations for Q-4

Checking class of invoice date variable

```
class(df$InvoiceDate)
```

```
[1] "character"
```

Converting invoice date's data type from character to POSIXIt-

```
#Using strptime command
Temp=strptime(df$InvoiceDate,format = "%m/%d/%Y %H:%M",tz='GMT' )
  new.invoice.date = as.Date(Temp,"%d")
  invoice.day=weekdays(Temp)
  invoice.hours=format(Temp,"%H")
  invoice.months=format(Temp,"%m")
```

Q-4:

a) Show the percentage of transactions (by numbers) by days of the week

```
weekday.percentage = round(100*prop.table(table(invoice.day)))
weekday.percentage
```

```
invoice.day
   Friday    Monday    Sunday  Thursday   Tuesday Wednesday
       15        18        12        19        19        17
```

b)Show the percentage of transactions (by transaction volume) by days of the week

```
#Using variable transaction value as transaction volume
volume=cbind(df,invoice.day)
volume=volume %>%
  group_by(invoice.day) %>%
  summarise(weekday.volume.percentage=100*(sum(Transactionvalue)/sum(df$Transactionvalue)))
volume
```

```
# A tibble: 6 x 2
  invoice.day weekday.volume.percentage
  <chr>                           <dbl>
1 Friday                           15.8
2 Monday                           16.3
3 Sunday                            8.27
4 Thursday                         21.7
5 Tuesday                          20.2
6 Wednesday                        17.8
```

c)Show the percentage of transactions (by transaction volume) by month of the year

```
months.name=c("January","February","March","April","May","June","July","August","September","October","
weekday.percentage.month = round(100*prop.table(table(invoice.months)))
cbind(months.name,weekday.percentage.month)
```

```
   months.name  weekday.percentage.month
01 "January"    "6"
02 "February"   "5"
03 "March"      "7"
04 "April"      "6"
05 "May"        "7"
06 "June"       "7"
07 "July"       "7"
08 "August"     "7"
09 "September"  "9"
10 "October"    "11"
11 "November"   "16"
12 "December"   "13"
```

d)What was the date with the highest number of transactions from Australia?

```
df.A=cbind(df,new.invoice.date)
df.A=df.A %>%
  filter(Country=="Australia")
df.date.A=table(df.A$new.invoice.date)
#Using max command to get the date with maximum
answer.A=which(df.date.A==max(df.date.A))
answer.A
```

```
2011-06-15
        29
```

Hence, Australia did maximum transactions on 2011-06-15

e) The company needs to shut down the website for two consecutive hours for maintenance. What would be the hour of the day to start this so that the distribution is at minimum for the customers?

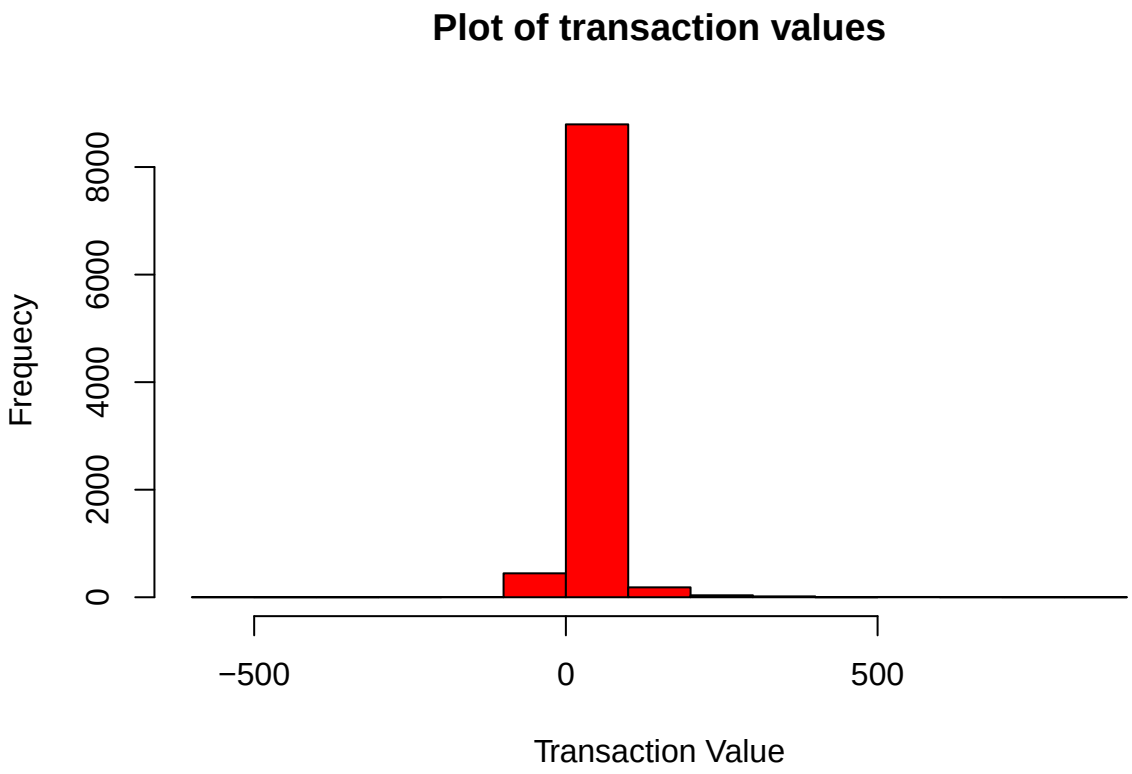
```
df.m=as.data.frame(table(invoice.hours))
df.m
```

```
   invoice.hours  Freq
1             06    41
2             07   383
3             08  8909
4             09 34332
5             10 49037
6             11 57674
7             12 78709
8             13 72259
9             14 67471
10            15 77519
11            16 54516
12            17 28509
13            18  7974
14            19  3705
15            20   871
```

We can see that the number of transactions are minimum fo 6th and 7th hour.Hence, the company can close for maintenance in these hours as it will cause least disturbance to customers.

Q-5: Plot the histogram of transaction values from Germany. Use the hist() function to plot.

```
df.5=df %>%
  #Chossig germany as country-
  filter(Country=="Germany") %>%
  mutate(Transactionvalue=as.numeric(Transactionvalue))
#Using hist function to plot histogram-
hist(df.5$Transactionvalue,
     main = "Plot of transaction values",
     xlab = "Transaction Value",
     ylab = "Frequecy",
     col = "red")
```



Q-6: Which customer had the highest number of transactions? Which customer is most valuable (i.e. highest total sum of transactions)?

```
#Using table command on customerID
table.ID=(table(df$CustomerID))
#Using max command to get maximum value of transactions
answer.customers=which.max(table.ID)
answer.customers
```

```
17841
 4043
```

Hence, CustomerID 17841 has the maximum number of transactions (4043)

```
#Using transaction value this time to get most valuable customer
most.valuable=df %>%
  filter(!is.na(CustomerID))
most.valuable=most.valuable %>%
  group_by(CustomerID) %>%
  summarise(spending=round(sum(Transactionvalue),2)) %>%
  filter(spending==max(spending))
  most.valuable
```

```
# A tibble: 1 x 2
  CustomerID spending
       <int>    <dbl>
1      14646  279489.
```

Hence,customerID 14646 is most value as it has maximum spending of 279489

Q-7: Calculate the percentage of missing values for each variable in the dataset

```
round(100*colMeans(is.na(df)))
```

```
       InvoiceNo        StockCode      Description         Quantity
               0                0                0                0
     InvoiceDate        UnitPrice       CustomerID          Country
               0                0               25                0
Transactionvalue
               0
```

Hence, only customer ID has missing values (25%)

Q-8: What are the number of transactions with missing CustomerID records by countries

Total number of missing values in CustomerID is :

```
ID.na = df %>%
  filter(is.na(CustomerID))
ID.na=ID.na %>%
  group_by(Country) %>%
  summarise(Number.of.missing.IDs=length(CustomerID))
ID.na
```

```
# A tibble: 9 x 2
  Country        Number.of.missing.IDs
  <chr>                          <int>
1 Bahrain                            2
2 EIRE                             711
3 France                            66
4 Hong Kong                        288
5 Israel                            47
6 Portugal                          39
7 Switzerland                      125
8 United Kingdom                133600
9 Unspecified                      202
```

Q-9: On average, how often the costumers comeback to the website for their next shopping?

```r
#making invoice date numeric and binding with dataset
df.consecutive=cbind(df,new.invoice.date)
df.consecutive=df.consecutive%>%
  select(CustomerID,new.invoice.date) %>%
  group_by(CustomerID) %>%
  distinct(new.invoice.date) %>%
  arrange(desc(CustomerID)) %>%
  #Making formula for difference b/w consecutive shoppings
  mutate(days.between=abs(new.invoice.date-lag(new.invoice.date))) %>%
  #Removing NA values
  filter(!is.na(days.between))
#Calculating average of days in between of consecutive shopping
mean(df.consecutive$days.between)
```

```
Time difference of 38.4875 days
```

Q-10:In the retail sector, it is very important to understand the return rate of the goods purchased by customers. In this example, we can define this quantity, simply, as the ratio of the number of transactions cancelled (regardless of the transaction value) over the total number of transactions.With this definition, what is the return rate for the French customers?

```r
df.france.cancelled=df %>%
  filter(Country=="France", Quantity<0)
df.france=df %>%
  filter(Country=="France")
#Dividing lengths of both quantities to get ratio as return rate
length(df.france.cancelled$Quantity)/length(df.france$Quantity)
```

```
[1] 0.01741264
```

Q-11:What is the product that has generated the highest revenue for the retailer?

```r
valable.product=df %>%
  group_by(Description) %>%
  summarise(revenue=round(sum(Transactionvalue),2)) %>%
  filter(revenue==max(revenue))
valable.product
```

```
# A tibble: 1 x 2
  Description    revenue
  <chr>            <dbl>
1 DOTCOM POSTAGE 206245.
```

Hence, DOTCOM POSTAGE generated highest revenue.

Q-12:How many unique customers are represented in the dataset? You can use unique() and length() functions.

```
length(unique(df$CustomerID))
```

```
[1] 4373
```

Hence, 4373 unique customerIDs are present in out dataset.
