# BA assignment 1

Nemin Dholakia

10/21/2021

```
library(dplyr)
library(zoo)
set.seed(120)
library(readr)
library(tinytex)
Online_Retail <-O_R <- read_csv("D:/MSBA/Business
Analytics/Online_Retail.csv")
```

##1. Showing the breakdown of the number oftransactions by countries i.e. how many transactions are in the dataset for each country (considering all records including cancelled transactions). Showing this in total number and also in percentage.Showing only countries accounting for more than 1% of the total transactions.

```
O_R %>%
  group_by(Country)%>%
  summarise(transactions = n())%>%
  mutate(percentage= (transactions/541909)*100)%>%
  arrange(desc(transactions))%>%
  filter(data <- percentage > 1)

## # A tibble: 4 x 3
##   Country        transactions percentage
##   <chr>                 <int>      <dbl>
## 1 United Kingdom       495478      91.4
## 2 Germany                9495       1.75
## 3 France                 8557       1.58
## 4 EIRE                   8196       1.51
```

##2. Creating a new variable ‘Transaction Value’ that is the product of the exising ‘Quantity’ and ‘UnitPrice’ variables. Add this variable to the dataframe.

```
O_R<- mutate(O_R, "TransactionValue"=TransactionValue<- O_R$Quantity *
O_R$UnitPrice)
colnames(O_R)

## [1] "InvoiceNo"         "StockCode"         "Description"       "Quantity"
## [5] "InvoiceDate"       "UnitPrice"         "CustomerID"        "Country"
## [9] "TransactionValue"
```

##3. Will Use the newly created variable,TransactionValue, will show the breakdown of transactionvaluesby countries i.e. how much money in total has been spent each country.

Will Show this in total sum of transaction values. Show only countries with total transaction exceeding 130,000 British Pound.

```
O_R%>%
  group_by(Country)%>%
  summarise(total.sum.of.transaction.values = sum(TransactionValue))%>%
  arrange(desc(total.sum.of.transaction.values))%>%
  filter(total.sum.of.transaction.values>130000)
```

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

##4.This is an optional question which carries additional marks (golden questions). In this question, we are dealing with the InvoiceDate variable. The variable is read as a categorical when you read data from the file. Now we need to explicitly instruct R to interpret this as a Date variable. “POSIXlt” and “POSIXct”are two powerful object classesin R to deal with date and time. Click herefor more information. First let’s convert ‘InvoiceDate’ into a POSIXltobject:Temp=strptime(O_R$InvoiceDate, format = '$New_Invoice_Date<-as.Date(Temp)The Date objects have a lot of flexible functions. For example knowing two date values, the object allows you to know the difference between the two dates in terms of the number days. Try this:O_R$New_Invoice_Date[20000] - O_R$New_Invoice_Date[10]Also we can convert dates to days of the week. Let’s define a new variable for thatO_R$Invoice_Day_Week = weekdays(O_R$New_Invoice_Date) Page 3For the Hour, let’s just take the hour (ignore the minute) and convert into a normal numerical value:O_R$New_Invoice_Hour =as.numeric(format(Temp,"%H"))Finally, lets define the month as a separate numeric variable too:O_R$New_Invoice_Month = as.numeric(format(Temp, “%m”))

```
#Now,let’s convert ‘InvoiceDate’ into a POSIXltobject:
Temp=strptime(O_R$InvoiceDate,format='%m/%d/%Y %H:%M',tz='GMT')
#Now, let’s separate date,  day  of  the  week  and  hour components
dataframe with names as
#New_Invoice_Date,Invoice_Day_Weekand New_Invoice_Hour:
O_R$New_Invoice_Date<-as.Date(Temp)
#knowing two date values,the object allows you to know the difference between
the two dates in terms of the number days.
O_R$New_Invoice_Date[20000]-O_R$New_Invoice_Date[10]
```

```
## Time difference of 8 days
```

```
#Also we can convert dates to days of the week. Let’s define a new variable
for that
O_R$Invoice_Day_Week=weekdays(O_R$New_Invoice_Date)
```

```
#Now, let’s just take the hour (ignore the minute)  and  convert  into  a
normal  numerical value:
O_R$New_Invoice_Hour =as.numeric(format(Temp,"%H"))
#Now, lets define the month as a separate numeric variable too:
O_R$New_Invoice_Month = as.numeric(format(Temp, "%m"))
```

## Answering the following questions:

##4.a)Will show the percentage of transactions (by numbers) by days of the week

```
O_R%>%
  group_by(Invoice_Day_Week)%>%
  summarise(Number.of.transaction=(n()))%>%

mutate(Number.of.transaction,'percent'=(Number.of.transaction*100)/sum(Number
.of.transaction))

## # A tibble: 6 x 3
##   Invoice_Day_Week Number.of.transaction percent
##   <chr>                            <int>   <dbl>
## 1 Friday                           82193    15.2
## 2 Monday                           95111    17.6
## 3 Sunday                           64375    11.9
## 4 Thursday                        103857    19.2
## 5 Tuesday                         101808    18.8
## 6 Wednesday                        94565    17.5
```

##4.b)Will show the percentage of transactions (by transaction volume) bydays of the week

```
O_R%>%
  group_by(Invoice_Day_Week)%>%
  summarise(Volume.of.transaction=(sum(TransactionValue)))%>%

mutate(Volume.of.transaction,'percent'=(Volume.of.transaction*100)/sum(Volume
.of.transaction))

## # A tibble: 6 x 3
##   Invoice_Day_Week Volume.of.transaction percent
##   <chr>                            <dbl>   <dbl>
## 1 Friday                        1540611.   15.8
## 2 Monday                        1588609.   16.3
## 3 Sunday                         805679.    8.27
## 4 Thursday                      2112519    21.7
## 5 Tuesday                       1966183.   20.2
## 6 Wednesday                     1734147.   17.8
```

##4.c)Will show the percentage of transactions (by transaction volume) by month of the year

```
O_R%>%
  group_by(New_Invoice_Month)%>%
  summarise(Volume.By.Month=sum(TransactionValue))%>%

mutate(Volume.By.Month,'Percent'=(Volume.By.Month*100)/sum(Volume.By.Month))

## # A tibble: 12 x 3
##    New_Invoice_Month Volume.By.Month Percent
##                <dbl>           <dbl>   <dbl>
##  1                 1         560000.    5.74
##  2                 2         498063.    5.11
##  3                 3         683267.    7.01
##  4                 4         493207.    5.06
##  5                 5         723334.    7.42
##  6                 6         691123.    7.09
##  7                 7         681300.    6.99
##  8                 8         682681.    7.00
##  9                 9        1019688.   10.5
## 10                10        1070705.   11.0
## 11                11        1461756.   15.0
## 12                12        1182625.   12.1
```

##4.d) The date with the highest number of transactions from Australia

```
c<-O_R%>%
  group_by(New_Invoice_Date,Country)%>%
  filter(Country=='Australia')%>%
  summarise(Number=sum(Quantity),amount=sum(TransactionValue))%>%
  arrange(desc(Number))
c<-c[c['Number']==max(c['Number']),]
print(paste('The date with the highest number of transactions from Australia
is', c['New_Invoice_Date'],'which is',c['amount'],'$'))

## [1] "The date with the highest number of transactions from Australia is
15140 which is 23426.81 $"
```

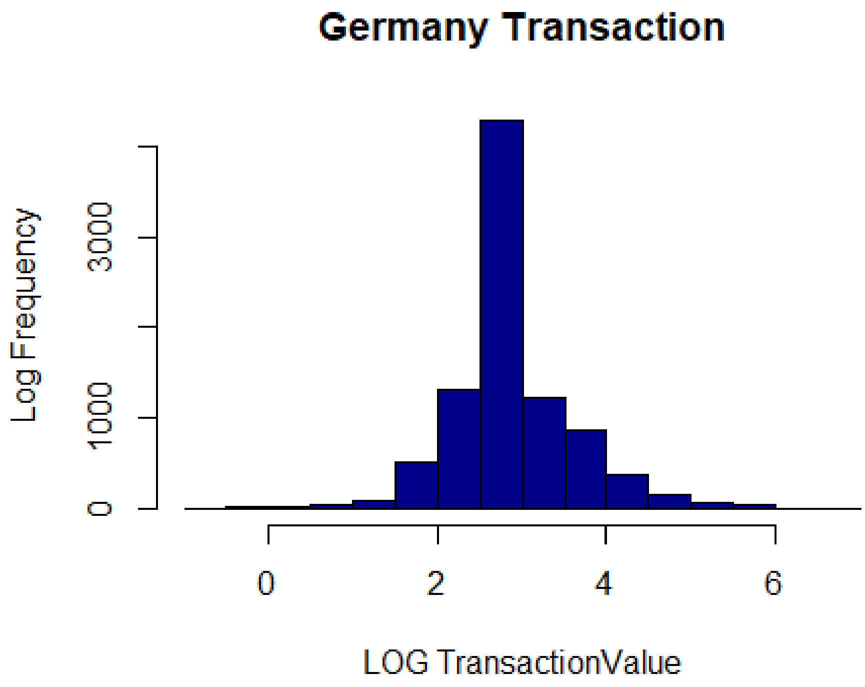
##4.e)The company needs to shut down the website for two consecutivehours for maintenance. What would be the hour of the day to start this so that the distribution is at minimum for the customers? The responsible IT team is available from 7:00 to 20:00 every day.

```
d=O_R%>%
  group_by(New_Invoice_Hour)%>%
  summarise(Total.transaction= n())
e<-rollapply(d['Total.transaction'],2,sum)
index(min(e))

## [1] 1

print('As per the data, in the morning between 7 to 9 is the best time for
shut  down the  website  for two consecutivehours for maintenance')
```

```
## [1] "As per the data, in the morning between 7 to 9 is the best time for
shut  down the  website  for two consecutivehours for maintenance"
```

##5.Plotting the histogramo f transaction values from Germany.Usethe hist() function to plot.

```
hist(x=log(O_R$TransactionValue[O_R$Country=="Germany"]),xlab = "LOG
TransactionValue",col = 'dark blue' ,main = 'Germany Transaction',ylab = 'Log
Frequency')
```



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

```
data<- O_R %>%
  group_by(CustomerID)%>%
  summarise(CustomerTransaction = n())%>%
  filter(CustomerID != "NA")%>%
  filter(CustomerTransaction ==max(CustomerTransaction) )
print(paste('The customerID had the highest number of transactions
is',data$CustomerID,'with max transaction of ',data$CustomerTransaction))

## [1] "The customerID had the highest number of transactions is 17841 with
max transaction of  7983"

data2<- O_R%>%
  group_by(CustomerID)%>%
  summarise(total.transaction.by.each.customer = sum(TransactionValue))%>%
  arrange(desc(total.transaction.by.each.customer))%>%
  filter(CustomerID != "NA")%>%
  filter(total.transaction.by.each.customer
==max(total.transaction.by.each.customer) )
print(paste('Most valuable customerID is',data2$CustomerID,'with total
transaction Amount $',data2$total.transaction.by.each.customer))

## [1] "Most valuable customerID is 14646 with total transaction Amount $
279489.02"
```

##7.Calculating the percentage of missing values for each variable in the dataset

```
NullValue<-colMeans(is.na(O_R))
print(paste('Online customerID column has missing values in dataset and
i.e.',NullValue['CustomerID']*100,'% of whole data'))

## [1] "Online customerID column has missing values in dataset and  i.e.
24.9266943342886 % of whole data"
```

##8.What are the number oftransactions withmissing CustomerID recordsby countries?

```
O_R%>%
  group_by(Country)%>%
  filter(is.na(CustomerID))%>%
  summarise(No.of.missing.CustomerID=n())

## # A tibble: 9 x 2
##   Country        No.of.missing.CustomerID
##   <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
```

##9.On average, how often the costumers comeback to the website for their next shopping? (i.e. what is the average number of days between consecutive shopping)

```
aa<-O_R%>%
  group_by(CustomerID)%>%
  summarise(difference.in.consecutivedays= diff(New_Invoice_Date))%>%
  filter(difference.in.consecutivedays>0)
print(paste('the average  number  of  days  between  consecutive  shopping
is',mean(aa$difference.in.consecutivedays)))

## [1] "the average  number  of  days  between  consecutive  shopping is
38.4875"
```

##10.In the retail sector, it is very important to understand the return rate of the goods purchased by customers. In this example, we will be defining this quantity, simply,as the ratio of the numberof 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? Considering the cancelled transactions as those where the ‘Quantity’ variable hasa negative value.

```
return_val<-nrow(O_R%>%
  group_by(CustomerID)%>%
  filter((Country=='France')&(TransactionValue<0)&(CustomerID != 'Na')))
total_french_customer<-nrow(O_R%>%
  group_by(CustomerID)%>%
  filter((Country=='France')&(CustomerID != 'Na')))



print(paste('Return rate for french customer is given
as',((return_val)/(total_french_customer))*100,'Percent'))

## [1] "Return rate for french customer is given as 1.75479919915204 Percent"
```

##11.The product that has generated the highest revenue for the retailer

```
Total_customer1<-O_R%>%
  group_by(Description,StockCode)%>%
  summarise(n=sum(TransactionValue))%>%
  arrange(desc(n))
a<- Total_customer1[Total_customer1['n']==max(Total_customer1['n']),]
print(paste('The product generated the highest revenue is',
a$Description,'with stock code',a$StockCode))

## [1] "The product generated the highest revenue is DOTCOM POSTAGE with
stock code DOT"
```

##12. Unique customers represented in the dataset. Will use unique() and length() functions.

```
print(paste('Total no. of customers with valid customer id are
',length(unique(O_R$CustomerID))-1,'. This does not include null
CustomerID'))

## [1] "Total no. of customers with valid customer id are  4372 . This does
not include null CustomerID"
```