Assignment 2

Kiran Kour

2022-10-18

# Importing dataset

OnlineRetail<- read.csv("Online\_Retail.csv")

# Importing required libraries

#install.packages("tidyverse")

library(tidyverse)

## ── Attaching packages ─────────────────────────────────────── tidyverse 1.3.2 ──  
## ✔ ggplot2 3.3.6 ✔ purrr 0.3.4  
## ✔ tibble 3.1.8 ✔ dplyr 1.0.9  
## ✔ tidyr 1.2.1 ✔ stringr 1.4.1  
## ✔ readr 2.1.2 ✔ forcats 0.5.2  
## ── Conflicts ────────────────────────────────────────── tidyverse\_conflicts() ──  
## ✖ dplyr::filter() masks stats::filter()  
## ✖ dplyr::lag() masks stats::lag()

library(readr)  
library(dplyr)

# The first 6 columns of the dataset

head(OnlineRetail)

## InvoiceNo StockCode Description Quantity  
## 1 536365 85123A WHITE HANGING HEART T-LIGHT HOLDER 6  
## 2 536365 71053 WHITE METAL LANTERN 6  
## 3 536365 84406B CREAM CUPID HEARTS COAT HANGER 8  
## 4 536365 84029G KNITTED UNION FLAG HOT WATER BOTTLE 6  
## 5 536365 84029E RED WOOLLY HOTTIE WHITE HEART. 6  
## 6 536365 22752 SET 7 BABUSHKA NESTING BOXES 2  
## InvoiceDate UnitPrice CustomerID Country  
## 1 12/1/2010 8:26 2.55 17850 United Kingdom  
## 2 12/1/2010 8:26 3.39 17850 United Kingdom  
## 3 12/1/2010 8:26 2.75 17850 United Kingdom  
## 4 12/1/2010 8:26 3.39 17850 United Kingdom  
## 5 12/1/2010 8:26 3.39 17850 United Kingdom  
## 6 12/1/2010 8:26 7.65 17850 United Kingdom

# Data Exploration

# Getting the descriptive statistics  
  
summary(OnlineRetail)

## InvoiceNo StockCode Description Quantity   
## Length:541909 Length:541909 Length:541909 Min. :-80995.00   
## Class :character Class :character Class :character 1st Qu.: 1.00   
## Mode :character Mode :character Mode :character Median : 3.00   
## Mean : 9.55   
## 3rd Qu.: 10.00   
## Max. : 80995.00   
##   
## InvoiceDate UnitPrice CustomerID Country   
## Length:541909 Min. :-11062.06 Min. :12346 Length:541909   
## Class :character 1st Qu.: 1.25 1st Qu.:13953 Class :character   
## Mode :character Median : 2.08 Median :15152 Mode :character   
## Mean : 4.61 Mean :15288   
## 3rd Qu.: 4.13 3rd Qu.:16791   
## Max. : 38970.00 Max. :18287   
## NA's :135080

**Question 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.**

# Number of transactions by countries  
  
Transactions<- table(OnlineRetail$Country)  
head(Transactions)

##   
## Australia Austria Bahrain Belgium Brazil Canada   
## 1259 401 19 2069 32 151

# Countries accounting for more than 1% of the total transactions.  
  
Trans\_Countries<- OnlineRetail %>% group\_by(Country)%>% summarise(Total\_Trans= n(), Total\_Perc= sum(n()/length(OnlineRetail$Country)\*100)) %>% filter(Total\_Perc >1)

# Dataframe for the Number of countries with more than 1% of the total transactions  
head(Trans\_Countries)

## # A tibble: 4 × 3  
## Country Total\_Trans Total\_Perc  
## <chr> <int> <dbl>  
## 1 EIRE 8196 1.51  
## 2 France 8557 1.58  
## 3 Germany 9495 1.75  
## 4 United Kingdom 495478 91.4

*EIRE, FRANCE, GERMANY, and UNITED KINGDOM are the countries with more than 1% of the total transactions.*

**Question 2:**

**Create a new variable ‘TransactionValue’ that is the product of the exising ‘Quantity’ and ‘UnitPrice’ variables. Add this variable to the dataframe.**

# Creation of new variable 'TransactionValue'.  
  
OnlineRetail <- OnlineRetail %>% mutate(TransactionValue= Quantity \* UnitPrice)  
  
# Rows and columns of the dataset  
  
head(OnlineRetail)

## InvoiceNo StockCode Description Quantity  
## 1 536365 85123A WHITE HANGING HEART T-LIGHT HOLDER 6  
## 2 536365 71053 WHITE METAL LANTERN 6  
## 3 536365 84406B CREAM CUPID HEARTS COAT HANGER 8  
## 4 536365 84029G KNITTED UNION FLAG HOT WATER BOTTLE 6  
## 5 536365 84029E RED WOOLLY HOTTIE WHITE HEART. 6  
## 6 536365 22752 SET 7 BABUSHKA NESTING BOXES 2  
## InvoiceDate UnitPrice CustomerID Country TransactionValue  
## 1 12/1/2010 8:26 2.55 17850 United Kingdom 15.30  
## 2 12/1/2010 8:26 3.39 17850 United Kingdom 20.34  
## 3 12/1/2010 8:26 2.75 17850 United Kingdom 22.00  
## 4 12/1/2010 8:26 3.39 17850 United Kingdom 20.34  
## 5 12/1/2010 8:26 3.39 17850 United Kingdom 20.34  
## 6 12/1/2010 8:26 7.65 17850 United Kingdom 15.30

**Question 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.**

# Countries with total transaction exceeding 130,000 British Pound  
  
BritishPound <- OnlineRetail %>% select(Country, TransactionValue)%>% group\_by(Country) %>% summarise(Transactions= sum(TransactionValue))%>% filter(Transactions >130000)

as.data.frame(BritishPound)

## Country Transactions  
## 1 Australia 137077.3  
## 2 EIRE 263276.8  
## 3 France 197403.9  
## 4 Germany 221698.2  
## 5 Netherlands 284661.5  
## 6 United Kingdom 8187806.4

*There are in total 6 countries whose transactions exceed 130,000 British Pound out of which United Kingdom has the highest transaction.*

**Question 4:**

**Converting InvoiceDate variable to Date variable**

# First let’s convert ‘InvoiceDate’ into a POSIXlt object:  
  
Temp=strptime(OnlineRetail$InvoiceDate,format='%m/%d/%Y %H:%M',tz='GMT')   
  
# Checking the variable  
  
head(Temp)

## [1] "2010-12-01 08:26:00 GMT" "2010-12-01 08:26:00 GMT"  
## [3] "2010-12-01 08:26:00 GMT" "2010-12-01 08:26:00 GMT"  
## [5] "2010-12-01 08:26:00 GMT" "2010-12-01 08:26:00 GMT"

#Now, let’s separate date, day of the week and hour components dataframe with names as New\_Invoice\_Date,Invoice\_Day\_Week and New\_Invoice\_Hour:   
  
OnlineRetail$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.   
  
OnlineRetail$New\_Invoice\_Date[20000]-OnlineRetail$New\_Invoice\_Date[10]

## Time difference of 8 days

# Converting dates to days of the week. Let’s define a new variable for that   
  
OnlineRetail$Invoice\_Day\_Week= weekdays(OnlineRetail$New\_Invoice\_Date)

# For the Hour, let’s just take the hour (ignore the minute) and convert into a normal numerical value:   
  
OnlineRetail$New\_Invoice\_Hour = as.numeric(format(Temp, "%H"))

# Finally, lets define the month as a separate numeric variable too:   
  
OnlineRetail$New\_Invoice\_Month = as.numeric(format(Temp, "%m"))

# Dataset with new columns  
  
OnlineRetail[1:6,10:13]

## New\_Invoice\_Date Invoice\_Day\_Week New\_Invoice\_Hour New\_Invoice\_Month  
## 1 2010-12-01 Wednesday 8 12  
## 2 2010-12-01 Wednesday 8 12  
## 3 2010-12-01 Wednesday 8 12  
## 4 2010-12-01 Wednesday 8 12  
## 5 2010-12-01 Wednesday 8 12  
## 6 2010-12-01 Wednesday 8 12

**Now answer the following questions.**

1. **Show the percentage of transactions (by numbers) by days of the week.**

# Getting the total no.of day transactions and its percentage  
  
Day\_Percent <- OnlineRetail %>% group\_by(Invoice\_Day\_Week) %>% summarise(Trans\_Number= n(), Percent= sum(n()/length(OnlineRetail$Invoice\_Day\_Week)\*100))

#Show the dataframe  
  
as.data.frame(Day\_Percent)

## Invoice\_Day\_Week Trans\_Number Percent  
## 1 Friday 82193 15.16731  
## 2 Monday 95111 17.55110  
## 3 Sunday 64375 11.87930  
## 4 Thursday 103857 19.16503  
## 5 Tuesday 101808 18.78692  
## 6 Wednesday 94565 17.45035

1. **Show the percentage of transactions (by transaction volume) by days of the week.**

# Getting the total volume of transactions by week and it's percentage  
  
Totalday\_percent <- OnlineRetail%>% group\_by(Invoice\_Day\_Week)%>% summarise(Total\_trans= sum(TransactionValue))%>% mutate(Percent= Total\_trans/sum(Total\_trans)\*100)

as.data.frame(Totalday\_percent)

## Invoice\_Day\_Week Total\_trans Percent  
## 1 Friday 1540610.8 15.804787  
## 2 Monday 1588609.4 16.297194  
## 3 Sunday 805678.9 8.265282  
## 4 Thursday 2112519.0 21.671867  
## 5 Tuesday 1966182.8 20.170636  
## 6 Wednesday 1734147.0 17.790232

1. **Show the percentage of transactions (by transaction volume) by month of the year.**

# Getting total transaction value by months and it's percent  
  
Totalmonth\_percent <- OnlineRetail%>% group\_by(New\_Invoice\_Month)%>% summarise(Total\_trans= sum(TransactionValue))%>% mutate(Percent= Total\_trans/sum(Total\_trans)\*100)

as.data.frame((Totalmonth\_percent))

## New\_Invoice\_Month Total\_trans Percent  
## 1 1 560000.3 5.744919  
## 2 2 498062.6 5.109515  
## 3 3 683267.1 7.009487  
## 4 4 493207.1 5.059703  
## 5 5 723333.5 7.420519  
## 6 6 691123.1 7.090080  
## 7 7 681300.1 6.989308  
## 8 8 682680.5 7.003469  
## 9 9 1019687.6 10.460751  
## 10 10 1070704.7 10.984123  
## 11 11 1461756.2 14.995836  
## 12 12 1182625.0 12.132290

1. **What was the date with the highest number of transactions from Australia?**

# Selecting the date with highest number of transactions from Australia  
  
Highest\_num<- OnlineRetail%>% filter(OnlineRetail$Country == "Australia")%>% group\_by(New\_Invoice\_Date)%>% summarise(Aus\_TransactionValue= n())%>% top\_n(1, Aus\_TransactionValue)

as.data.frame(Highest\_num)

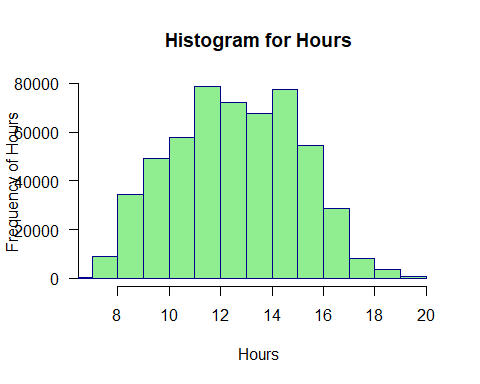
## New\_Invoice\_Date Aus\_TransactionValue  
## 1 2011-06-15 139

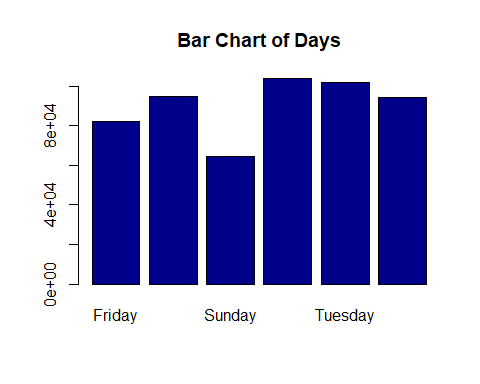
*On 2011-06-15 Australia recorded the highest number of transactions i.e., 139.*

1. **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? The responsible IT team is available from 7:00 to 20:00 every day.**

# Histogram for hours  
  
hist(OnlineRetail$New\_Invoice\_Hour, main= "Histogram for Hours",xlab= "Hours", ylab= "Frequency of Hours",border= "Dark blue",col= "Light green", las=1, xlim=c(7,20), breaks= 12)

# Bar Chart to identify   
  
barplot(table(OnlineRetail$Invoice\_Day\_Week), main="Bar Chart of Days",col=" Dark Blue")



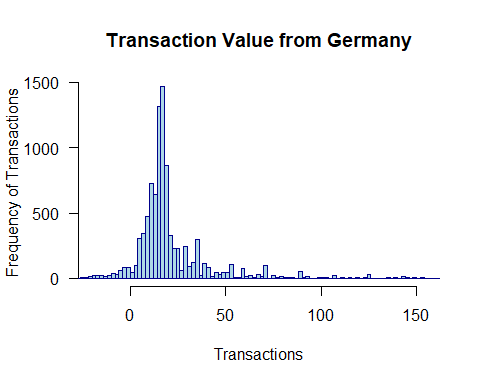


*From the Histogram and the Bar chart we can interpret that the best hours to do the maintenance of the company’s website are between 18:00- 20:00. Moreover, Sunday would be the great day to do the maintenance.*

1. **Plot the histogram of transaction values from Germany. Use the hist() function to plot.**

# Getting the transaction values from Germany  
  
Germany <- select(OnlineRetail,TransactionValue, Country)%>% filter(OnlineRetail$Country == "Germany")

# Histogram for transaction values from Germany  
  
hist(Germany$TransactionValue,xlab= "Transactions",ylab= "Frequency of Transactions",xlim=c(-20,155),las=1, breaks= 600,col= "light blue",border="dark blue", main="Transaction Value from Germany")



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

# Customer with the highest number of transactions  
  
Valuable\_customer <- OnlineRetail %>% na.omit()%>% group\_by(CustomerID)%>% summarise(Num\_highest = n())%>% top\_n(1,Num\_highest)

as.data.frame(Valuable\_customer)

## CustomerID Num\_highest  
## 1 17841 7983

*The customer with CustomerID 17841 had the highest number of transactions amongst the others with a total of 7983.*

# Valuable customer with the highest Volume of transactions  
  
Valuable\_customer <- OnlineRetail%>% na.omit()%>% group\_by(CustomerID)%>% summarise(High\_transaction= sum(TransactionValue))%>% top\_n(1,High\_transaction)

as.data.frame(Valuable\_customer)

## CustomerID High\_transaction  
## 1 14646 279489

*The customer with CustomerID 14646 is the valuable customer with the highest transaction sum of 279489 British Sterling Pound.*

1. **Calculate the percentage of missing values for each variable in the dataset. Hint colMeans():**

percent\_missing <- colMeans(is.na(OnlineRetail))

as.data.frame(percent\_missing)

## percent\_missing  
## InvoiceNo 0.0000000  
## StockCode 0.0000000  
## Description 0.0000000  
## Quantity 0.0000000  
## InvoiceDate 0.0000000  
## UnitPrice 0.0000000  
## CustomerID 0.2492669  
## Country 0.0000000  
## TransactionValue 0.0000000  
## New\_Invoice\_Date 0.0000000  
## Invoice\_Day\_Week 0.0000000  
## New\_Invoice\_Hour 0.0000000  
## New\_Invoice\_Month 0.0000000

*Out of all the Variables in the dataset CustomerID is the only attribute with 24.92669% of NAs.*

1. **What are the number of transactions with missing CustomerID records by countries?**

# Number of Transactions with missing CustomerID records by countries  
  
ID\_missing<- OnlineRetail%>% group\_by(Country, CustomerID)%>% filter(is.na(CustomerID)) %>% summarise(Num\_trans= n())

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

as.data.frame(ID\_missing)

## Country CustomerID Num\_trans  
## 1 Bahrain NA 2  
## 2 EIRE NA 711  
## 3 France NA 66  
## 4 Hong Kong NA 288  
## 5 Israel NA 47  
## 6 Portugal NA 39  
## 7 Switzerland NA 125  
## 8 United Kingdom NA 133600  
## 9 Unspecified NA 202

*There are 9 countries with missing CustomerID records, out of which the United Kingdom is the highest with 133600 missing values.*

1. **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)**

# Days average between consecutive shopping  
  
Days\_Avg <- OnlineRetail %>% select(CustomerID, New\_Invoice\_Date) %>% group\_by(CustomerID) %>% mutate(Days\_diff = as.numeric(c(diff(New\_Invoice\_Date),0))) %>% summarise(Days\_time = sum(Days\_diff),  
Days\_Avg = sum(Days\_diff)/sum(n()))

head(as.data.frame(Days\_Avg))

## CustomerID Days\_time Days\_Avg  
## 1 12346 0 0.000000  
## 2 12347 365 2.005495  
## 3 12348 283 9.129032  
## 4 12349 0 0.000000  
## 5 12350 0 0.000000  
## 6 12352 260 2.736842

1. **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? Consider the cancelled transactions as those where the ‘Quantity’ variable has a negative value.**

# Return rate for the French customers  
  
numerator <- OnlineRetail %>% select(Quantity, TransactionValue, Country) %>% filter(Country == "France" & Quantity < 0)  
denominator <- OnlineRetail %>% select(Quantity, TransactionValue, Country) %>% filter(Country == "France")  
Ratio <- count(numerator) / count(denominator)  
  
  
as.data.frame(Ratio)

## n  
## 1 0.01741264

*The return rate for the French customers is 1.741264%*

1. **What is the product that has generated the highest revenue for the retailer? (i.e. item with the highest total sum of ‘TransactionValue’).**

# Highest revenue for the retailer  
  
Rev\_Highest <- OnlineRetail %>% group\_by(Description) %>%  
summarise(Trans\_highest = sum(TransactionValue)) %>%  
 top\_n(1)

## Selecting by Trans\_highest

as.data.frame((Rev\_Highest))

## Description Trans\_highest  
## 1 DOTCOM POSTAGE 206245.5

*The product generating the highest revenue for the retailer is DOTCOM POSTAGE i.e., 206245.5 British Sterling Pound.*

1. **How many unique customers are represented in the dataset? You can use unique() and length() functions.**

#Showing the number of unique customers  
  
length(unique(OnlineRetail$CustomerID))

## [1] 4373

*There are total of 4373 unique customers*