# title: “BA\_Asssignment 2”

# output: “Yaswanth\_Golla”

# date: “2023-10-14”

library(ggplot2)  
library(dplyr)

##   
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':  
##   
## filter, lag

## The following objects are masked from 'package:base':  
##   
## intersect, setdiff, setequal, union

Retail<-read.csv("~/Documents/BA/BA Assignment 2/Online\_Retail.csv")  
summary(Retail)

## 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

#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.

Countries\_counts = Retail %>% group\_by(Country) %>% count(Country)  
Countries\_percentage = Retail %>% group\_by(Country) %>% summarise(percent = 100\* n()/nrow(Retail))  
Filtered\_Country\_percentage = filter(Countries\_percentage, percent>1)

# view the countries counts  
Countries\_counts

## # A tibble: 38 × 2  
## # Groups: Country [38]  
## Country n  
## <chr> <int>  
## 1 Australia 1259  
## 2 Austria 401  
## 3 Bahrain 19  
## 4 Belgium 2069  
## 5 Brazil 32  
## 6 Canada 151  
## 7 Channel Islands 758  
## 8 Cyprus 622  
## 9 Czech Republic 30  
## 10 Denmark 389  
## # ℹ 28 more rows

# view the transactions greater than 1%  
Filtered\_Country\_percentage

## # A tibble: 4 × 2  
## Country percent  
## <chr> <dbl>  
## 1 EIRE 1.51  
## 2 France 1.58  
## 3 Germany 1.75  
## 4 United Kingdom 91.4

#Q2 Create a new variable ‘TransactionValue’ that is the product of the exising ‘Quantity’ and ‘UnitPrice’ variables. Add this variable to the dataframe.

Online\_Retail <- cbind(Retail, TransactionValues = Retail$Quantity \* Retail$UnitPrice)  
head(Online\_Retail)

## 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 TransactionValues  
## 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

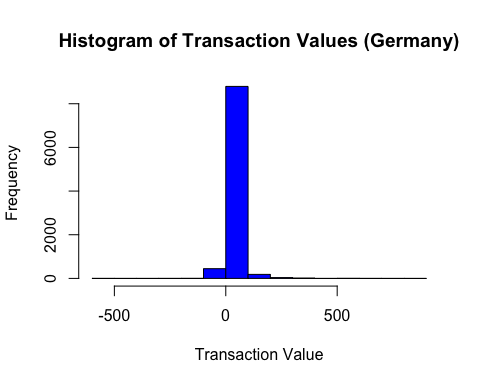
#Q3 Using the newly created variable, TransactionValue, show the breakdown of transaction values by countries i.e. how much money in total has been spent each country. Show this in total sum of transaction values. Show only countries with total transaction exceeding 130,000 British Pound.

Online\_Retail %>%   
 group\_by(Country) %>%  
 summarise(Total\_Spend = sum(TransactionValues)) %>%  
 filter(Total\_Spend > 130000) %>%   
 arrange(desc(Total\_Spend))

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

#Q5)Plot the histogram of transaction values from Germany. Use the hist() function to plot.

GermanyTransactions <- subset(Online\_Retail, Country == "Germany")  
hist(GermanyTransactions$TransactionValues, main = "Histogram of Transaction Values (Germany)", xlab = "Transaction Value", ylab = "Frequency",col = "blue")

 # Q6)Which customer had the highest number of transactions? Which customer is most valuable (i.e.highest total sum of transactions)?

Online\_Retail %>%  
 group\_by(CustomerID) %>%  
 summarise(transactions = n()) %>%  
 top\_n(2) %>%  
 arrange(desc(transactions))

## Selecting by transactions

## # A tibble: 2 × 2  
## CustomerID transactions  
## <int> <int>  
## 1 NA 135080  
## 2 17841 7983

Online\_Retail %>%  
 group\_by(CustomerID) %>%  
 summarise(transaction\_sum = sum(TransactionValues)) %>%  
 top\_n(2) %>%  
 arrange(desc(transaction\_sum))

## Selecting by transaction\_sum

## # A tibble: 2 × 2  
## CustomerID transaction\_sum  
## <int> <dbl>  
## 1 NA 1447682.  
## 2 14646 279489.

#Q7)Calculate the percentage of missing values for each variable in the dataset

missingvalues= colMeans(is.na(Online\_Retail))\*100  
missingvalues

## InvoiceNo StockCode Description Quantity   
## 0.00000 0.00000 0.00000 0.00000   
## InvoiceDate UnitPrice CustomerID Country   
## 0.00000 0.00000 24.92669 0.00000   
## TransactionValues   
## 0.00000

#Q8)What are the number of transactions with missing CustomerID records by countries?

missingCustomer = Online\_Retail[is.na(Online\_Retail$CustomerID),]  
table(missingCustomer$Country)

##   
## Bahrain EIRE France Hong Kong Israel   
## 2 711 66 288 47   
## Portugal Switzerland United Kingdom Unspecified   
## 39 125 133600 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)

# Assuming 'Invoice Date' is in a date format  
Online\_Retail$InvoiceDate <- as.Date(Online\_Retail$InvoiceDate)  
  
# Sort the data by CustomerID and InvoiceDate  
Online\_Retail <- Online\_Retail %>%  
 arrange(CustomerID, InvoiceDate)  
  
# Calculate the time difference between consecutive transactions for each customer  
time\_diff <- Online\_Retail %>%  
 group\_by(CustomerID) %>%  
 mutate(DaysBetween = as.numeric(difftime(InvoiceDate, lag(InvoiceDate), units = "days")))  
  
# Remove the first row for each customer since there is no previous transaction  
time\_diff <- time\_diff %>%  
 filter(!is.na(DaysBetween))  
  
# Calculate the average number of days between consecutive shopping trips  
average\_days\_between\_shopping <- mean(time\_diff$DaysBetween, na.rm = TRUE)  
  
# Display the result  
print(average\_days\_between\_shopping)

## [1] 14.98301

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

Francetransactions\_Cancelled <- subset(Online\_Retail, Country == "France" & Quantity < 0)  
Francetransactions <- subset(Online\_Retail, Country == "France")  
France\_Returnrate <- 100\*(nrow(Francetransactions\_Cancelled) / nrow(Francetransactions))  
France\_Returnrate

## [1] 1.741264

#Q11)What is the product that has generated the highest revenue for the retailer? (i.e. item with the highest total sum of ‘TransactionValue’).

Product\_revenue= Online\_Retail %>% group\_by(StockCode) %>% summarise(Sum\_transactionvalue = sum(TransactionValues))  
Product\_revenue[which.max(Product\_revenue$Sum\_transactionvalue),]

## # A tibble: 1 × 2  
## StockCode Sum\_transactionvalue  
## <chr> <dbl>  
## 1 DOT 206245.

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

uniquecustomers <- unique(Online\_Retail$CustomerID)  
number\_of\_uniquecustomers <- length(uniquecustomers)  
print(number\_of\_uniquecustomers)

## [1] 4373