assignment\_2

sai sree pulimamidi

2022-10-26

***Loading all the required packages***

library("VIM")

## Loading required package: colorspace

## Loading required package: grid

## VIM is ready to use.

## Suggestions and bug-reports can be submitted at: https://github.com/statistikat/VIM/issues

##   
## Attaching package: 'VIM'

## The following object is masked from 'package:datasets':  
##   
## sleep

library("ISLR")  
library("caret")

## Loading required package: ggplot2

## Loading required package: lattice

library("class")  
library("e1071")  
library("ggplot2")  
library("corrplot")

## corrplot 0.92 loaded

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

***Setting up working directory and loading data***

online\_retail <- read.csv("C:/Users/sudhakar/Downloads/Online\_Retail (1).csv")

\*\*\*1

data\_country <- as.data.frame(table(online\_retail$Country))  
  
data\_country$Percentage <- data\_country$Freq/nrow(online\_retail) \* 100  
  
colnames(data\_country) <- c("Country", "Count", "Percentage")  
  
data\_country[data\_country$Percentage > 1,]

## Country Count Percentage  
## 11 EIRE 8196 1.512431  
## 14 France 8557 1.579047  
## 15 Germany 9495 1.752139  
## 36 United Kingdom 495478 91.431956

*Countries accounting for more than 1% of the total transactions are EIRE, France, Germany and United Kingdom.* /vspace{1mm}/newline

***2. Adding new attribute “TransactionValue” which is the product of Quantity and UnitPrice***

online\_retail$TransactionValue <- online\_retail$Quantity \* online\_retail$UnitPrice

*By adding this new attribute we can now calculate the value of the transactions based on our requirement.* /vspace{1mm}/newline

***3. Using the newly created variable, TransactionValue, showing the breakdown of transaction values by countries with total transaction exceeding 130,000 British Pound***

online\_retail %>% select(TransactionValue,Country) %>% group\_by(Country) %>% summarise(Total = sum(TransactionValue)) %>% filter(Total >= 130000) %>% arrange((desc(Total)))

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

*There are total 6 countries where the transaction value exceeds 130,000 British Pound and the highest among them is “United Kingdom”.* /vspace{1mm}/newline

***4. Converting Invoice Date into a POSIXlt object***

Temp=strptime(online\_retail$InvoiceDate,format='%m/%d/%Y %H:%M',tz='GMT')  
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"

#New\_Invoice\_Date  
online\_retail$New\_Invoice\_Date <- as.Date(Temp)  
  
online\_retail$New\_Invoice\_Date[20000]- online\_retail$New\_Invoice\_Date[10]

## Time difference of 8 days

#Invoice\_Day\_Week  
online\_retail$Invoice\_Day\_Week= weekdays(online\_retail$New\_Invoice\_Date)  
  
#New\_Invoice\_Hour  
online\_retail$New\_Invoice\_Hour = as.numeric(format(Temp, "%H"))  
  
#New\_Invoice\_Month  
online\_retail$New\_Invoice\_Month = as.numeric(format(Temp, "%m"))

***4(a). Percentage of transactions (by numbers) by days of the week***

online\_retail %>% group\_by(Invoice\_Day\_Week) %>% summarise(count=n()) %>% mutate(percentage=count/nrow(online\_retail)\*100)

## # A tibble: 6 × 3  
## Invoice\_Day\_Week count percentage  
## <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). Percentage of transactions (by transaction volume) by days of the week***

online\_retail %>% group\_by(Invoice\_Day\_Week) %>% summarise(Total = sum(TransactionValue)) %>% mutate(Percentage = Total/sum(Total)\*100)

## # A tibble: 6 × 3  
## Invoice\_Day\_Week Total Percentage  
## <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). Percentage of transactions (by transaction volume) by month of the year***

online\_retail %>% group\_by(New\_Invoice\_Month) %>% summarise(Total = sum(TransactionValue)) %>% mutate(Percentage = Total/sum(Total) \* 100)

## # A tibble: 12 × 3  
## New\_Invoice\_Month Total Percentage  
## <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***

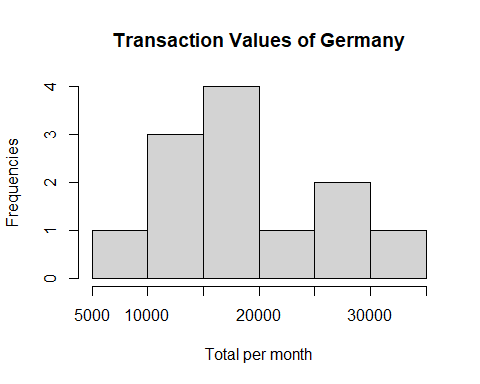
online\_retail %>% filter(Country =="Australia") %>% group\_by(New\_Invoice\_Date) %>% summarise(Total\_Count = n()) %>% arrange((desc(Total\_Count)))

## # A tibble: 49 × 2  
## New\_Invoice\_Date Total\_Count  
## <date> <int>  
## 1 2011-06-15 139  
## 2 2011-07-19 137  
## 3 2011-08-18 97  
## 4 2011-03-03 84  
## 5 2011-10-05 82  
## 6 2011-05-17 73  
## 7 2011-02-15 69  
## 8 2011-01-06 48  
## 9 2011-07-14 35  
## 10 2011-09-16 34  
## # … with 39 more rows

*As we can see from above on 2011-06-15 Australia has recorded the highest number of transactions i.e. 139 Transactions.* /vspace{1mm}/newline

***5. Plot the histogram of transaction values from Germany***

Germany <- online\_retail %>% filter(Country == "Germany") %>% group\_by(New\_Invoice\_Month) %>% summarise(Total=sum(TransactionValue))   
  
hist(Germany$Total, main = "Transaction Values of Germany", xlab="Total per month", ylab="Frequencies")



***6(a). Customer who had highest number of transactions***

online\_retail %>% group\_by(CustomerID) %>% select(CustomerID) %>% filter(!is.na(CustomerID)) %>% summarise(n\_count = n()) %>% arrange((desc(n\_count)))

## # A tibble: 4,372 × 2  
## CustomerID n\_count  
## <int> <int>  
## 1 17841 7983  
## 2 14911 5903  
## 3 14096 5128  
## 4 12748 4642  
## 5 14606 2782  
## 6 15311 2491  
## 7 14646 2085  
## 8 13089 1857  
## 9 13263 1677  
## 10 14298 1640  
## # … with 4,362 more rows

*The CustomerID 17841 had the highest number of transactions amongst others with a total of 7983 transactions.* /vspace{1mm}/newline

***6(b). Most valuable customer with the highest total sum of transactions***

online\_retail %>% group\_by(CustomerID) %>% select(CustomerID, TransactionValue) %>% filter(!is.na(CustomerID)) %>% summarise(Spending\_max = sum(TransactionValue)) %>% arrange((desc(Spending\_max)))

## # A tibble: 4,372 × 2  
## CustomerID Spending\_max  
## <int> <dbl>  
## 1 14646 279489.  
## 2 18102 256438.  
## 3 17450 187482.  
## 4 14911 132573.  
## 5 12415 123725.  
## 6 14156 113384.  
## 7 17511 88125.  
## 8 16684 65892.  
## 9 13694 62653.  
## 10 15311 59419.  
## # … with 4,362 more rows

*The CustomerID 14646 is the most valuable customer with the highest spending sum of 279,489.020 British Sterling Pound.* /vspace{1mm}/newline

***7. Percentage of missing values for each variable in the dataset***

colMeans(is.na(online\_retail)\*100)

## InvoiceNo StockCode Description Quantity   
## 0.00000 0.00000 0.00000 0.00000   
## InvoiceDate UnitPrice CustomerID Country   
## 0.00000 0.00000 24.92669 0.00000   
## TransactionValue New\_Invoice\_Date Invoice\_Day\_Week New\_Invoice\_Hour   
## 0.00000 0.00000 0.00000 0.00000   
## New\_Invoice\_Month   
## 0.00000

*We can observe that CustomerID is the only attribute with 24.9266% of NAs in the entire dataset.* /vspace{1mm}/newline

***8. The number of transactions with missing CustomerID records by Countries***

online\_retail %>% filter(is.na(CustomerID)) %>% group\_by(Country) %>% count()

## # A tibble: 9 × 2  
## # Groups: Country [9]  
## Country n  
## <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

*There are in total 8 countries and 1 unspecified country in the entire dataset which has NA values in them amongst these United Kingdom is the country with highest NA records of 133,600 rows.* /vspace{1mm}/newline

***9. On average, how often the costumers comeback to the website for their next shopping?***

Diff\_Days <- online\_retail %>% select(CustomerID,New\_Invoice\_Date) %>% group\_by(CustomerID) %>% distinct(New\_Invoice\_Date) %>% arrange(desc(CustomerID)) %>% mutate(Days\_Between = New\_Invoice\_Date-lag(New\_Invoice\_Date)) %>% filter(!is.na(Days\_Between))   
  
Diff\_Days

## # A tibble: 15,200 × 3  
## # Groups: CustomerID [2,992]  
## CustomerID New\_Invoice\_Date Days\_Between  
## <int> <date> <drtn>   
## 1 18287 2011-10-12 143 days   
## 2 18287 2011-10-28 16 days   
## 3 18283 2011-01-23 17 days   
## 4 18283 2011-02-28 36 days   
## 5 18283 2011-04-21 52 days   
## 6 18283 2011-05-23 32 days   
## 7 18283 2011-06-14 22 days   
## 8 18283 2011-06-23 9 days   
## 9 18283 2011-07-14 21 days   
## 10 18283 2011-09-05 53 days   
## # … with 15,190 more rows

mean(Diff\_Days$Days\_Between)

## Time difference of 38.4875 days

*On an average approximately for every 38 days customers come back to the website for their next shopping.* /vspace{1mm}/newline

***10. Return rate of goods purchased by the customers from France***

F\_Cancel <- online\_retail %>% filter(Country=="France",Quantity<0) %>% count()  
  
F\_Total <- online\_retail %>% filter(Country=="France") %>% count()  
  
Return\_Percentage\_France <- F\_Cancel/F\_Total\*100  
Return\_Percentage\_France

## n  
## 1 1.741264

*The return rate of customers who made purchases in France is 1.741264%.* /vspace{1mm}/newline

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

online\_retail %>% select(StockCode,TransactionValue) %>% group\_by(StockCode) %>% summarise(Total=sum(TransactionValue)) %>% arrange((desc(Total)))

## # A tibble: 4,070 × 2  
## StockCode Total  
## <chr> <dbl>  
## 1 DOT 206245.  
## 2 22423 164762.  
## 3 47566 98303.  
## 4 85123A 97894.  
## 5 85099B 92356.  
## 6 23084 66757.  
## 7 POST 66231.  
## 8 22086 63792.  
## 9 84879 58960.  
## 10 79321 53768.  
## # … with 4,060 more rows

*The product with the StockCode as “DOT” is the one which has generated highest revenue to the retailer i.e. 206,245.48 British Sterling Pound.* /vspace{1mm}/newline

***12. Unique Customers in the dataset***

online\_retail %>% select(CustomerID) %>% unique() %>% count()

## n  
## 1 4373

*In total there are 4,373 unique customers in the dataset.*