Vamsi Nalluri

Loading the Data Set & the required libraries and diplaying the summary stats and the top 6 rows of the data using head function.

#install.packages("dplyr)  
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

#install.packages("ggplot2")  
library(ggplot2)  
  
# clean the workspace  
rm(list = ls())  
  
Mydata = read.csv('Online\_Retail.csv')  
  
head(Mydata)

## 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

summary(Mydata)

## 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

We will use group\_by, summarise , mutate and filter functions to out put the desired result

Mydata %>% group\_by(Country) %>% summarise(n=n()) %>% mutate(per = 100\* n/sum(n)) %>% filter(per>1)

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

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

We will use mutate function from dplyr package to add another column to the dataframe Mydata and (display the dataframe)

Mydata <- Mydata %>% mutate(TransactionValue = (Quantity \* UnitPrice))  
  
#Mydata

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.

Using group\_by, summarise and filter functions to display the countries that have a total transaction value of over 130000

Mydata %>% group\_by(Country) %>% summarise(TV = sum(TransactionValue)) %>% ungroup %>% filter(TV>130000)

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

1. Bonus Point Questions:

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

Percent of Transactions (numbers) by Day of the week in Per column

Temp=strptime(Mydata$InvoiceDate,format='%m/%d/%Y %H:%M',tz='GMT')  
  
Mydata$New\_Invoice\_Date <- as.Date(Temp)  
  
Mydata$Invoice\_Day\_Week= weekdays(Mydata$New\_Invoice\_Date)  
  
Mydata$New\_Invoice\_Hour = as.numeric(format(Temp, "%H"))  
  
Mydata$New\_Invoice\_Month = as.numeric(format(Temp, "%m"))  
  
  
Mydata %>% count(Invoice\_Day\_Week , sort = TRUE) %>% mutate(per = 100\* n/sum(n))

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

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

Percent of Transactions (volume) by Day of the week in Per column

We first get the total(count) of transaction values by week day into Mydata\_TV\_DW\_Per dataframe Then we get the total transaction values by by week day into Mydata\_TV\_DW\_Per1 for each month Lastly, we append the percentage (per) column to Mydata\_TV\_DW\_Per1

Mydata\_TV\_DW\_Per <- Mydata %>% count(Invoice\_Day\_Week) %>% summarise(TVDW = sum(Mydata$TransactionValue))   
  
Mydata\_TV\_DW\_Per2 <- Mydata %>% group\_by(Invoice\_Day\_Week) %>% summarise(TVDW1 = sum(TransactionValue))  
  
Mydata\_TV\_DW\_Per2 %>% mutate(per = 100\* Mydata\_TV\_DW\_Per2$TVDW1/(Mydata\_TV\_DW\_Per[1,1]))

## # A tibble: 6 × 3  
## Invoice\_Day\_Week TVDW1 per  
## <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

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

Percent of Transactions (volume) by month of the year

We first get the total(count) of transaction values by month into Mydata\_TV\_M\_Per dataframe Then we get the total transaction values by by month into Mydata\_TV\_M\_Per1 for each month Lastly, we append the percentage (per) column to Mydata\_TV\_M\_Per1

Mydata\_TV\_M\_Per <- Mydata %>% count(New\_Invoice\_Month) %>% summarise(TVM = sum(Mydata$TransactionValue))   
  
Mydata\_TV\_M\_Per1 <- Mydata %>% group\_by(New\_Invoice\_Month) %>% summarise(TVM1 = sum(TransactionValue))  
  
Mydata\_TV\_M\_Per1 %>% mutate(per = 100\* Mydata\_TV\_M\_Per1$TVM1/(Mydata\_TV\_M\_Per[1,1]))

## # A tibble: 12 × 3  
## New\_Invoice\_Month TVM1 per  
## <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

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

We will group by and count the values , string the result in Mydata4 and then apply filter

Mydata4 <- Mydata %>% group\_by(Country) %>% count(Mydata$country,Mydata$New\_Invoice\_Date)  
  
Mydata5 <- filter(Mydata4 , Country == "Australia" & n == max(n))   
  
Mydata5

## # A tibble: 1 × 3  
## # Groups: Country [1]  
## Country `Mydata$New\_Invoice\_Date` n  
## <chr> <date> <int>  
## 1 Australia 2011-06-15 139

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

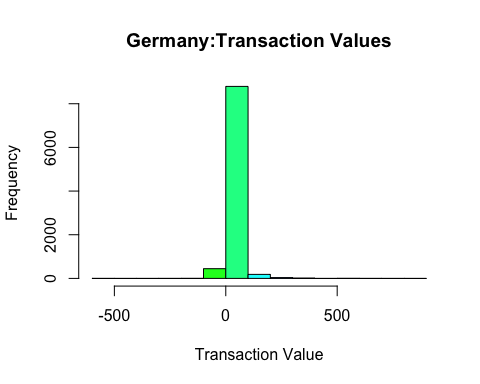
A cursory look at the result shows that it would be idle to go for the maintenance during 19 to 20 hours as the cumulative value of transcations are less during that time (though its much lesser during 6 to 7, we cant for that window as IT team would not be available at 6)

Mydata6 <- Mydata %>% group\_by(New\_Invoice\_Hour) %>% count(Mydata$New\_Invoice\_Hour , sort = TRUE)  
  
Mydata6

## # A tibble: 15 × 3  
## # Groups: New\_Invoice\_Hour [15]  
## New\_Invoice\_Hour `Mydata$New\_Invoice\_Hour` n  
## <dbl> <dbl> <int>  
## 1 12 12 78709  
## 2 15 15 77519  
## 3 13 13 72259  
## 4 14 14 67471  
## 5 11 11 57674  
## 6 16 16 54516  
## 7 10 10 49037  
## 8 9 9 34332  
## 9 17 17 28509  
## 10 8 8 8909  
## 11 18 18 7974  
## 12 19 19 3705  
## 13 20 20 871  
## 14 7 7 383  
## 15 6 6 41

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

Mydata7 <- Mydata %>% summarise(Country,TransactionValue)  
  
Mydata8 <- filter(Mydata7 , Country == "Germany")   
  
hist(Mydata8$TransactionValue,main="Germany:Transaction Values", xlab = "Transaction Value", col = rainbow(14))



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

First we will calculate the highest number of transactions by a customer and then most valuble (highest value) customer

Mydata %>% count(CustomerID) %>% filter(n == (max(n)))

## CustomerID n  
## 1 NA 135080

#since highest number is associated with NA , we will ignore the CustomerID NA and check again  
  
Mydata9 <- Mydata %>% count(CustomerID) %>% filter(CustomerID != "NA")  
  
filter(Mydata9,n == (max(n)))

## CustomerID n  
## 1 17841 7983

Mydata %>% group\_by(CustomerID) %>% summarise(TV = sum(TransactionValue)) %>% filter(TV == max(TV))

## # A tibble: 1 × 2  
## CustomerID TV  
## <int> <dbl>  
## 1 NA 1447682.

#since highest Transaction Value is associated with NA , we will ignore the CustomerID NA and check again  
  
Mydata10 <- Mydata %>% group\_by(CustomerID) %>% summarise(TV = sum(TransactionValue)) %>% filter(TV != max(TV))  
  
filter(Mydata10,TV == (max(TV)))

## # A tibble: 1 × 2  
## CustomerID TV  
## <int> <dbl>  
## 1 14646 279489.

7.Calculate the percentage of missing values for each variable in the dataset (5 marks). Hint colMeans():

summary(Mydata)

## 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   
## TransactionValue New\_Invoice\_Date Invoice\_Day\_Week New\_Invoice\_Hour  
## Min. :-168469.60 Min. :2010-12-01 Length:541909 Min. : 6.00   
## 1st Qu.: 3.40 1st Qu.:2011-03-28 Class :character 1st Qu.:11.00   
## Median : 9.75 Median :2011-07-19 Mode :character Median :13.00   
## Mean : 17.99 Mean :2011-07-04 Mean :13.08   
## 3rd Qu.: 17.40 3rd Qu.:2011-10-19 3rd Qu.:15.00   
## Max. : 168469.60 Max. :2011-12-09 Max. :20.00   
##   
## New\_Invoice\_Month  
## Min. : 1.000   
## 1st Qu.: 5.000   
## Median : 8.000   
## Mean : 7.553   
## 3rd Qu.:11.000   
## Max. :12.000   
##

# A quick glance at the summary stats reveals that only CustomerID column has got missing values(NAs). Hence , percentage of missing values for all other variables is 0.  
  
Percent\_CustID = mean(is.na(Mydata$CustomerID)) \* 100  
round(Percent\_CustID,2)

## [1] 24.93

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

Mydata %>% group\_by(Country) %>% count(CustomerID) %>% filter(is.na(CustomerID == "NA"))

## # A tibble: 9 × 3  
## # Groups: Country [9]  
## Country CustomerID n  
## <chr> <int> <int>  
## 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

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) (Optional/Golden question: 18 additional marks!) Hint: 1. A close approximation is also acceptable and you may find diff() function useful.

Sorting CustomerID and New\_Invoice\_Date columns and then mutating a new column intr with the diffrence in innvoice dates, finally calculating the average

Mydata11 <- Mydata[order(Mydata$CustomerID, Mydata$New\_Invoice\_Date),]  
Mydata12 <- Mydata11 %>% group\_by(CustomerID) %>% mutate(intr = (c(0,diff(New\_Invoice\_Date))))  
  
mean(Mydata12$intr)

## [1] 1.079535

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? (10 marks). Consider the cancelled transactions as those where the ‘Quantity’ variable has a negative value.

First we will calculate the total number of transaction from Frace and then count of cancelled transactions to calculate the Rate of Return

Mydata13 <- Mydata %>% group\_by(Country , Quantity) %>% summarise(n=n()) %>% filter(Country == "France")

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

Total\_Transactions\_Fr = sum(Mydata13$n)  
  
Mydata14 <- Mydata %>% group\_by(Country , Quantity) %>% summarise(n=n()) %>%   
 filter(Country == "France" & Quantity < 0)

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

Total\_Canl\_Transactions\_Fr = sum(Mydata14$n)  
  
RoR\_Fr = paste0((round(100\* Total\_Canl\_Transactions\_Fr/Total\_Transactions\_Fr,2 ))," %")  
  
  
RoR\_Fr

## [1] "1.74 %"

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

DOT is the product that generated the highest revenue

Mydata %>% group\_by(StockCode) %>% summarise(TV = sum(TransactionValue)) %>% filter(TV == max(TV))

## # A tibble: 1 × 2  
## StockCode TV  
## <chr> <dbl>  
## 1 DOT 206245.

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

length(unique(Mydata$CustomerID))

## [1] 4373