Assignment_1

Vamshee Deepak Goud Katta

10/20/2021

Reading data file

```
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
```

```
library(readr)
Retail <- read.csv("Online_Retail.csv")
```

1. Breakdown of transactions by country

```
Retail1  <- as.data.frame(table(Retail$Country))
Percentage <- Retail1$Freq/NROW(Retail) * 100
Retail1 <- cbind(Retail1, Percentage)
names(Retail1) <- c("Country","Total Transactions","Percentage")
Retail1[Retail1$Percentage > 1,]
```

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

2. Creating variable 'TransactionValue'

```
TransactionValue <- Retail$Quantity*Retail$UnitPrice
Retail2 <- cbind(Retail, TransactionValue)
head(Retail2)
```

```
##   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
```

3. Breakdown of transaction values by countries exceeding 130000

```
Retail3 <- Retail2%>%group_by(Country)%>%
  summarise(Total=sum(TransactionValue))
Retail3[Retail3$Total>130000,]
```

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

4. Golden Questions

Converting 'InvoiceDate' into a POSIXlt object

```
Retail4 <- Retail
Temp=strptime(Retail4$InvoiceDate,format='%m/%d/%Y %H:%M',tz='GMT')
Retail4$New_InvoiceDate <- as.Date(Temp)
Retail4$New_InvoiceDate[20000]- Retail4$New_InvoiceDate[10]
```

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

```
Retail4$Invoice_Day = weekdays(Retail4$New_InvoiceDate)
Retail4$Invoice_Hour = as.numeric(format(Temp, "%H"))
Retail4$Invoice_Month = as.numeric(format(Temp, "%m"))
```

a) Percentage of transactions (by numbers) by days of the week

```
Retail4%>%group_by(Invoice_Day)%>%
  summarise(count=n())%>%
  mutate(Percentage=count/nrow(Retail4)* 100)
```

```
## # A tibble: 6 x 3
##   Invoice_Day  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
```

b) Percentage of transactions (by transaction volume) by days of the week

```
Retail4%>%group_by(Invoice_Day)%>%
  summarise(Total=sum(TransactionValue))%>%
  mutate(Percentage=Total/sum(Total)*100)
```

```
## # A tibble: 6 x 3
##   Invoice_Day    Total Percentage
##   <chr>          <dbl>      <dbl>
## 1 Friday      9747748.       16.7
## 2 Monday      9747748.       16.7
## 3 Sunday      9747748.       16.7
## 4 Thursday    9747748.       16.7
## 5 Tuesday     9747748.       16.7
## 6 Wednesday   9747748.       16.7
```

c) Percentage of transactions (by transaction volume) by month of the year

```
Retail4%>%group_by(Invoice_Month)%>%
  summarise(Total=sum(TransactionValue))%>%
  mutate(Percentage=Total/sum(Total)*100)
```

```
## # A tibble: 12 x 3
##    Invoice_Month    Total Percentage
##            <dbl>    <dbl>      <dbl>
##  1             1 9747748.       8.33
```

```
##  2             2 9747748.       8.33
##  3             3 9747748.       8.33
##  4             4 9747748.       8.33
##  5             5 9747748.       8.33
##  6             6 9747748.       8.33
##  7             7 9747748.       8.33
##  8             8 9747748.       8.33
##  9             9 9747748.       8.33
## 10            10 9747748.       8.33
## 11            11 9747748.       8.33
## 12            12 9747748.       8.33
```

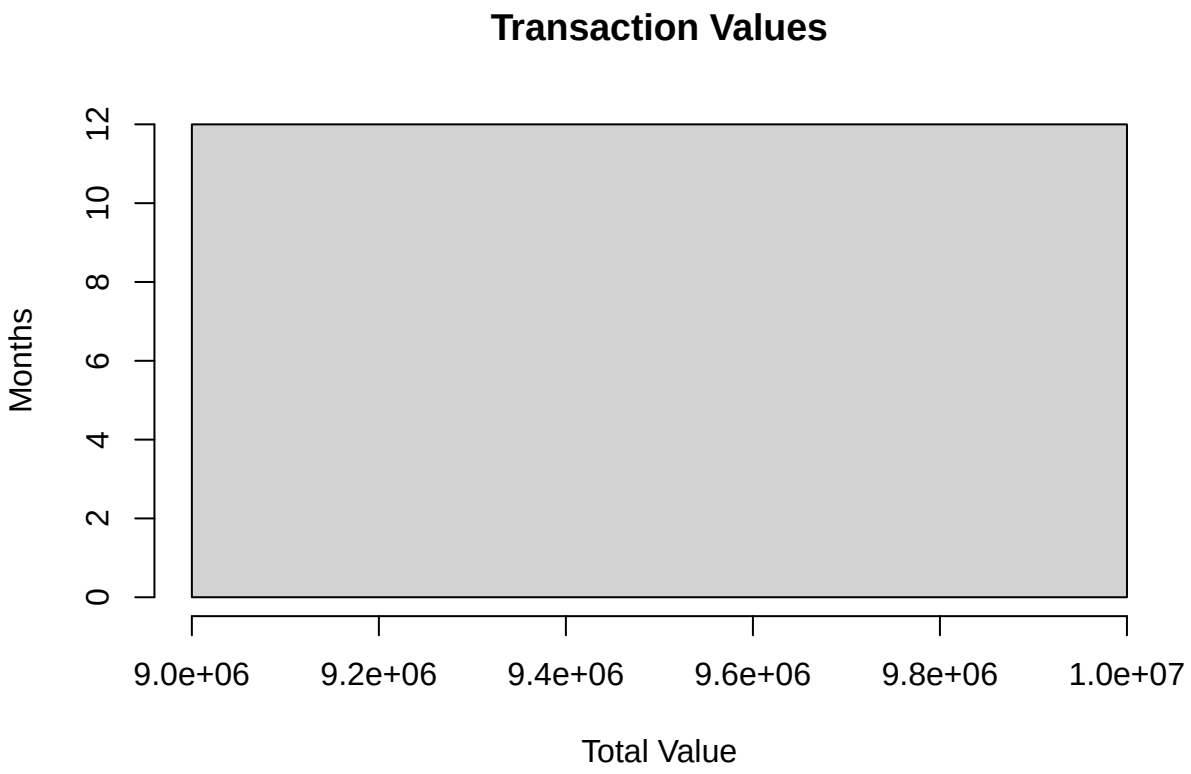
d) Date with the highest number of transactions from Australia

```
Retail4%>%
filter(Country=="Australia")%>%
group_by(New_InvoiceDate)%>%
tally(sort = TRUE)%>%
filter(n==max(n))
```

```
## # A tibble: 1 x 2
##   New_InvoiceDate     n
##   <date>          <int>
## 1 2011-06-15        139
```

5. Histogram of transaction values from Germany

```
Retail4%>%
group_by(Country)%>%
filter(Country=="Germany")%>%
group_by(Invoice_Month)%>%
summarise(Total = sum(TransactionValue))-> Germany
hist(Germany$Total, main = "Transaction Values", xlab = "Total Value", ylab = "Months")
```



6. Customer with highest number of transactions

```
Retail%>%
group_by(CustomerID)%>%
tally(sort = TRUE)%>%
filter(!is.na(CustomerID))%>%
filter(n==max(n))
```

```
## # A tibble: 1 x 2
##   CustomerID     n
##        <int> <int>
## 1      17841  7983
```

Most Valuable Customer

```
Retail2%>%
group_by(CustomerID)%>%
summarise(Total=sum(TransactionValue))%>%
filter(!is.na(CustomerID))%>%
filter(Total == max(Total))
```

```
## # A tibble: 1 x 2
##   CustomerID   Total
##        <int>   <dbl>
## 1      14646 279489.
```

7. Percentage of missing values

```
colMeans(is.na(Retail2) *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
##          0.00000
```

8. Number of transactions with missing CustomerID records by countries

```
Retail2%>%
  group_by(Country)%>%
  summarise(Total=nrow(Retail2[is.na(Retail2$CustomerID),]))
```

```
## # A tibble: 38 x 2
##    Country          Total
##    <chr>            <int>
##  1 Australia       135080
##  2 Austria         135080
##  3 Bahrain         135080
##  4 Belgium         135080
##  5 Brazil          135080
##  6 Canada          135080
##  7 Channel Islands 135080
##  8 Cyprus          135080
##  9 Czech Republic  135080
## 10 Denmark         135080
## # ... with 28 more rows
```

10. Return rate for the French customers

```
Retail10c <- Retail2%>%
  filter(Country=="France", Quantity<0)%>%
  count
Retail10t <- Retail2%>%
```

```
  filter(Country=="France")%>%
  count
Retail10 <- (Retail10c$n / Retail10t$n) * 100
Retail10
```

```
## [1] 1.741264
```

11. Product with highest revenue

```
Retail2%>%
  group_by(Description)%>%
  summarise(Total=sum(TransactionValue))%>%
  arrange(desc(Total)) %>%
  head(100)
```

```
## # A tibble: 100 x 2
##    Description                          Total
##    <chr>                                <dbl>
##  1 "DOTCOM POSTAGE"                    206245.
##  2 "REGENCY CAKESTAND 3 TIER"          164762.
##  3 "WHITE HANGING HEART T-LIGHT HOLDER"  99668.
##  4 "PARTY BUNTING"                      98303.
##  5 "JUMBO BAG RED RETROSPOT"            92356.
##  6 "RABBIT NIGHT LIGHT"                 66757.
##  7 "POSTAGE"                            66231.
##  8 "PAPER CHAIN KIT 50'S CHRISTMAS "    63792.
##  9 "ASSORTED COLOUR BIRD ORNAMENT"      58960.
## 10 "CHILLI LIGHTS"                      53768.
## # ... with 90 more rows
```

Postage is not an actual product. Hence the product with highest revenue is 'REGENCY CAKESTAND 3 TIER'

12. Unique customers in the dataset

```
Retail12 <- unique(Retail2$CustomerID, fromLast = FALSE,nmax = NA)
length(Retail12)
```

```
## [1] 4373
```

9. Average number of days between consecutive shopping