Assignment1

Reading the csv file

```
Retail_Management<-read.csv("Online_Retail.csv")
head(Retail_Management)
```

```
##   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
```

```
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(ISLR)
```

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.

```
Retail_Management %>%
group_by(Country) %>%
tally(sort=TRUE) %>% summarise(Country, counts=n,percent= n/sum(n)*100) %>% filter (p
ercent > 1)
```

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

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

```
Retail_Management<- mutate(Retail_Management,TransactionValue = Quantity * UnitPrice)
head(Retail_Management[,9])
```

```
## [1] 15.30 20.34 22.00 20.34 20.34 15.30
```

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

Retail_Management %>%
group_by(Country) %>%
  summarise(TransValuesum =sum (TransactionValue)) %>% filter(TransValuesum > 130000)
%>% arrange(desc(TransValuesum))
```

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

#4. This is an optional question which carries additional marks (golden questions). I n this question, we are dealing with the InvoiceDate variable. The variable is read a s a categorical when you read data from the file. Now we need to explicitly instruct R to interpret this as a Date variable. "POSIXlt" and "POSIXct" are two powerful obje ct classes in R to deal with date and time

```
Temp<- strptime(Retail_Management$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"
```

```
Retail_Management$New_Invoice_Date<-as.Date(Temp)

Retail_Management$Invoice_Day_week <- weekdays(Retail_Management$New_Invoice_Date)
Retail_Management$New_Invoice_Hour <-as.numeric (format(Temp,"%H"))
Retail_Management$New_Invoice_Month <- as.numeric(format(Temp, "%m"))
head(Retail_Management)
```

```
##   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
##   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
```

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

Retail_Management %>%
group_by(Invoice_Day_week) %>%
tally(sort=TRUE) %>%
summarise(Invoice_Day_week,TransactionCounts = n ,percent = n/sum(n)*100) %>%
arrange(desc(TransactionCounts))
```

```
## # A tibble: 6 × 3
##   Invoice_Day_week TransactionCounts percent
##   <chr>                        <int>   <dbl>
## 1 Thursday                    103857    19.2
## 2 Tuesday                     101808    18.8
## 3 Monday                       95111    17.6
## 4 Wednesday                    94565    17.5
## 5 Friday                       82193    15.2
## 6 Sunday                       64375    11.9
```

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

Retail_Management %>%
group_by(Invoice_Day_week) %>%
summarise(TransValueSum = sum(TransactionValue)) %>%
mutate(TransValuepercent= TransValueSum/sum(TransValueSum))%>%
arrange(desc(TransValueSum))
```

```
## # A tibble: 6 × 3
##   Invoice_Day_week TransValueSum TransValuepercent
##   <chr>                    <dbl>             <dbl>
## 1 Thursday              2112519             0.217
## 2 Tuesday               1966183.            0.202
## 3 Wednesday             1734147.            0.178
## 4 Monday                1588609.            0.163
## 5 Friday                1540611.            0.158
## 6 Sunday                 805679.            0.0827
```

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

Retail_Management %>%
group_by(New_Invoice_Month) %>%
summarise(TransValueSum = sum(TransactionValue)) %>%
mutate(TransValuePercent=TransValueSum/sum(TransValueSum)) %>%
arrange(desc(TransValuePercent))
```

```
## # A tibble: 12 × 3
##    New_Invoice_Month TransValueSum TransValuePercent
##                <dbl>         <dbl>             <dbl>
##  1                11      1461756.            0.150
##  2                12      1182625.            0.121
##  3                10      1070705.            0.110
##  4                 9      1019688.            0.105
##  5                 5       723334.            0.0742
##  6                 6       691123.            0.0709
##  7                 3       683267.            0.0701
##  8                 8       682681.            0.0700
##  9                 7       681300.            0.0699
## 10                 1       560000.            0.0574
## 11                 2       498063.            0.0511
## 12                 4       493207.            0.0506
```

```
#d) What was the date with the highest number of transactions from Australia?

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

```
## # A tibble: 1 × 2
##   InvoiceDate         n
##   <chr>           <int>
## 1 6/15/2011 13:37   139
```

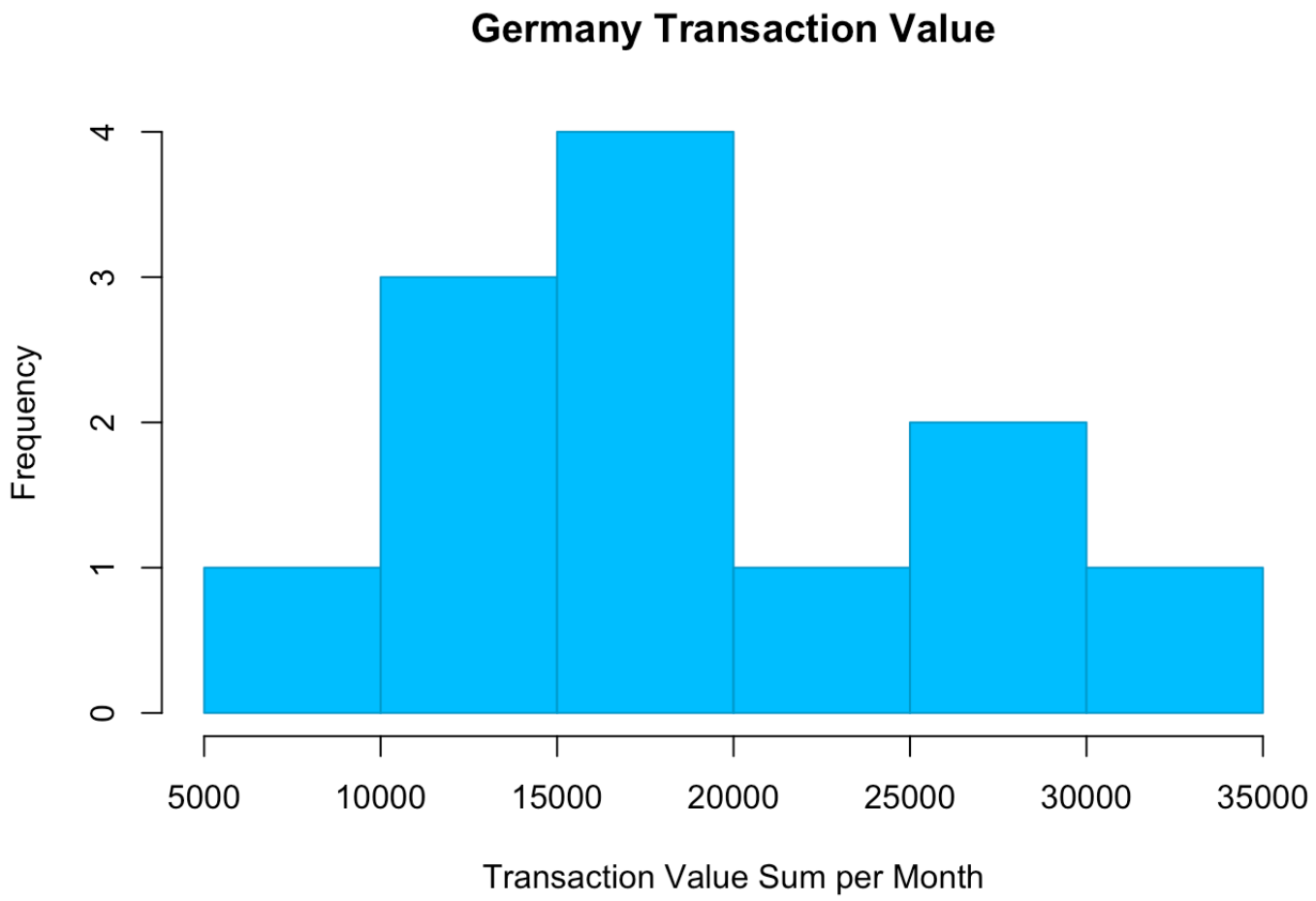
```
# e) The company needs to shut down the website for two consecutive hours for mainten
ance. What would be the hour of the day to start this so that the distribution is at
minimum for the customers? The responsible IT team is available from 7:00 to 20:00 ev
ery day.

Retail_Management %>%
group_by(New_Invoice_Hour) %>%
tally(sort=TRUE) %>%
filter(New_Invoice_Hour>=7 & New_Invoice_Hour<=20) %>%
arrange(n) %>%
head(5)
```

```
## # A tibble: 5 × 2
##   New_Invoice_Hour     n
##              <dbl> <int>
## 1                7   383
## 2               20   871
## 3               19  3705
## 4               18  7974
## 5                8  8909
```

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

Retail_Management %>%
group_by(Country) %>%
filter(Country == "Germany") %>%
group_by(New_Invoice_Month) %>%
summarise(TransValueSum = sum(TransactionValue)) -> Germany
hist(Germany$TransValueSum, border = "deepskyblue3", main = "Germany Transaction Value", xlab = "Transaction Value Sum per Month", ylab ="Frequency", col ="deepskyblue")
```



```
#6. Which customer had the highest number of transactions? Which customer is most val
uable (i.e. highest total sum of transactions)?

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

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

```
Retail_Management %>%
  group_by(CustomerID) %>%
  summarise(TransValueSum = sum(TransactionValue)) %>%
  filter(is.na(CustomerID)) %>%
  filter(TransValueSum == max(TransValueSum))
```

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

```
# 7. Calculate the percentage of missing values for each variable in the dataset (5 m
arks). Hint colMeans():

colMeans(is.na(Retail_Management))
```

```
##         InvoiceNo         StockCode       Description          Quantity
##         0.0000000         0.0000000         0.0000000         0.0000000
##       InvoiceDate         UnitPrice        CustomerID           Country
##         0.0000000         0.0000000         0.2492669         0.0000000
##  TransactionValue  New_Invoice_Date  Invoice_Day_week  New_Invoice_Hour
##         0.0000000         0.0000000         0.0000000         0.0000000
## New_Invoice_Month
##         0.0000000
```

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

Retail_Management %>%
filter(is.na(CustomerID)) %>%
group_by(Country)%>%
summarise(CustomerID) %>%
tally(sort=TRUE)
```

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

```
## # A tibble: 9 × 2
##   Country             n
##   <chr>           <int>
## 1 United Kingdom 133600
## 2 EIRE              711
## 3 Hong Kong         288
## 4 Unspecified       202
## 5 Switzerland       125
## 6 France             66
## 7 Israel             47
## 8 Portugal           39
## 9 Bahrain             2
```

```
#9. On average, how often the costumers comeback to the website for their next shoppi
ng?

Retail_Management %>%
select(CustomerID, New_Invoice_Date) %>%
group_by(CustomerID) %>%
distinct(New_Invoice_Date) %>%
arrange(desc(CustomerID)) %>%
mutate(DaysBetween = New_Invoice_Date - lag(New_Invoice_Date))->
custDaysBtwVisit

custDaysBtwVisit %>%
filter(!is.na(DaysBetween)) -> RetcustDaysBtwVisits
mean(RetcustDaysBtwVisits$DaysBetween)
```

```
## Time difference of 38.4875 days
```

```
# 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, a s the ratio of the number of transactions cancelled (regardless of the transaction va lue) 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 tho se where the 'Quantity' variable has a negative value.

Retail_Management %>%
group_by(Country) %>%
filter(Country=="France") %>%
select(Country,Quantity) %>%
filter(Quantity < 0 ) -> FrenchReturns
Retail_Management %>%
group_by(Country) %>%
filter(Country== "France") %>%
select(Quantity, Country) %>%
filter(Quantity > 0 ) ->FrenchPurchases
FRReturns<- sum(FrenchReturns$Quantity)
FRTransactions<-sum(FrenchPurchases$Quantity)
FRReturns/FRTransactions *100
```

```
## [1] -1.448655
```

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

Retail_Management %>%
group_by(StockCode) %>%
summarise(Trans_Value_Tot=sum(TransactionValue)) %>%
arrange(desc(Trans_Value_Tot)) %>%
filter(StockCode !="Dot") %>%
filter(Trans_Value_Tot == max(Trans_Value_Tot))
```

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

```
# 12. How many unique customers are represented in the dataset? You can use unique() and length() functions.
Retail_Management %>%
group_by(CustomerID) %>%
distinct(CustomerID) -> UniqueCustomers
length(UniqueCustomers$CustomerID)
```

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