

Assignment-Online Retail Analytics

```
library(dplyr)
Online_Retail<-read.csv("./Online_Retail.csv")
```

1. Breakdown of the number of transactions by countries in both percentage and count, and showing the countries accounting for more than 1% of the total transactions:

```
trans_countries<-Online_Retail %>% group_by(Country) %>% summarise(cnt = n()) %>% mutate(perc =round((cnt/sum(cnt))*100))
head(trans_countries)
```

```
## # A tibble: 4 x 3
##   Country      cnt  perc
##   <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. Creating a new variable TransactionValue and adding it to the dataframe:

```
TransactionValue<-Online_Retail$Quantity*Online_Retail$UnitPrice
#creating a dataframe and adding TransactionValue to it
Online_Retail_new<-data.frame(InvoiceNo=Online_Retail$InvoiceNo,StockCode= Online_Retail$StockCode,Desc=Online_Retail$Description,TransactionValue=TransactionValue)
```

3. Showing the breakdown of transaction values by countries in total sum of transaction value. Displaying countries with total transaction exceeding 13000

```
Trans_value_countries<- Online_Retail_new %>% group_by(Country) %>% summarise(sum_TransactionValue = sum(TransactionValue))
head(Trans_value_countries)
```

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

4. Optional question

```
Temp=strptime(Online_Retail_new$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$New_Invoice_Date<- as.Date(Temp)
```

```
Online_Retail_new$New_Invoice_Date[20000]- Online_Retail_new$New_Invoice_Date[10]
```

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

```
#Invoice_Week
```

```
Online_Retail_new$Invoice_Day_Week= weekdays(Online_Retail_new$New_Invoice_Date)
```

```
#Invoice_Hour
```

```
Online_Retail_new$New_Invoice_Hour = as.numeric(format(Temp, "%H"))
```

```
#Invoice_month
```

```
Online_Retail_new$New_Invoice_Month = as.numeric(format(Temp, "%m"))
```

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

```
perc_transc<- Online_Retail_new %>% group_by(Invoice_Day_Week) %>% summarise(count=n()) %>% mutate(perc=
head(perc_transc)
```

```
## # A tibble: 6 x 3
##   Invoice_Day_Week  count  perc
##   <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

```
perc_trans_week<- Online_Retail_new %>% group_by(Invoice_Day_Week) %>% summarise(Total=sum(TransactionV
head(perc_trans_week)
```

```
## # A tibble: 6 x 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

```
perc_trans_month<- Online_Retail_new %>% group_by(New_Invoice_Month) %>% summarise(Total=sum(TransactionValue))
head(perc_trans_month)
```

```
## # A tibble: 6 x 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
```

4(d). The date with the highest number of transactions from Australia

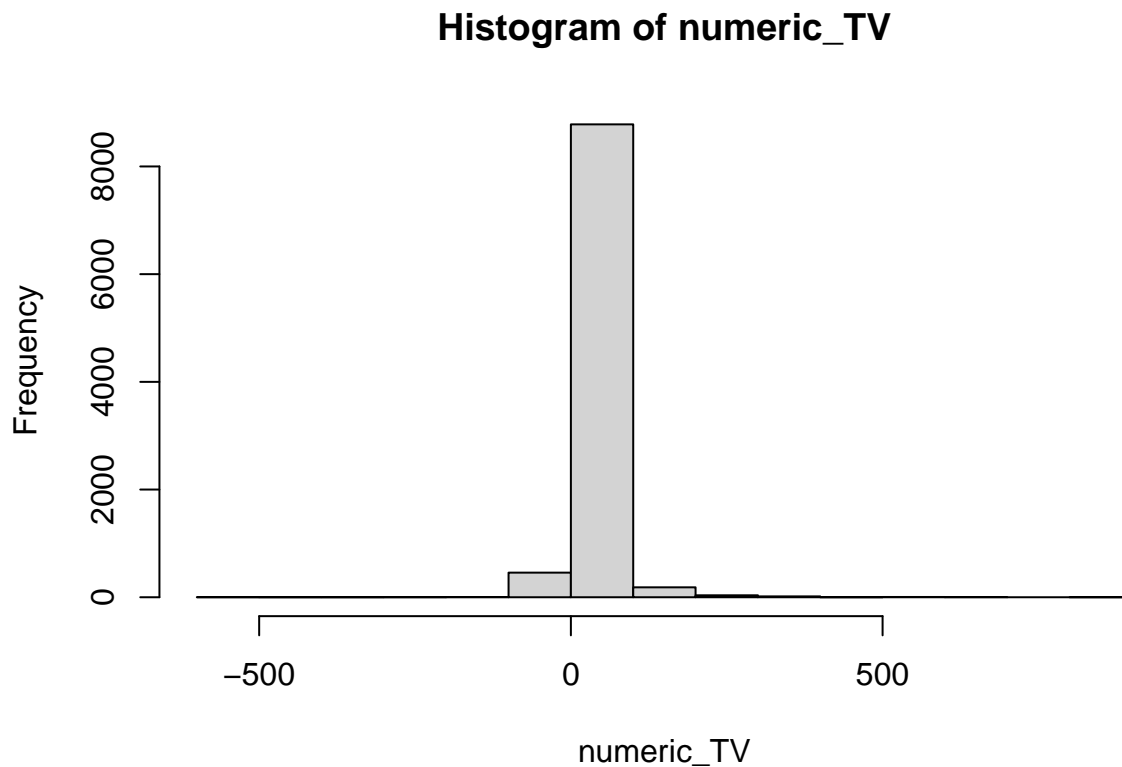
```
date_trans<- Online_Retail_new %>% filter(Country == 'Australia') %>% group_by(New_Invoice_Date) %>% summarise(count=n())
head(date_trans)
```

```
## # A tibble: 6 x 2
##   New_Invoice_Date 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
```

```
#Australia recorded highest number of transactions on 2011-06-15
```

5. Histogram of transaction Values from Germany

```
histogram<- Online_Retail_new %>% filter(Country == 'Germany')
numeric_TV<- as.integer(histogram$TransactionValue)
hist(numeric_TV)
```



6. Identifying the customer with highest number of transactions and finding the most valuable customer

```
cust_count<-Online_Retail_new %>% group_by(CustomerID) %>% summarise(cntt = n()) %>% arrange(desc(cntt))
head(cust_count)
```

```
## # A tibble: 6 x 2
##   CustomerID cntt
##   <int> <int>
## 1      NA 135080
## 2   17841  7983
## 3   14911  5903
## 4   14096  5128
## 5   12748  4642
## 6   14606  2782
```

#Customer 17841 has the highest number of transactions.

```
cust_sum<-Online_Retail_new %>% group_by(CustomerID) %>% summarise(sum_cnt =sum(TransactionValue)) %>% 
head(cust_sum)
```

```
## # A tibble: 6 x 2
##   CustomerID sum_cnt
##   <int> <dbl>
```

```
## 1      NA 1447682.
## 2    14646 279489.
## 3    18102 256438.
## 4    17450 187482.
## 5    14911 132573.
## 6    12415 123725.
```

#Customer 14646 is the most valuable

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

```
missing_values<- (colMeans(is.na(Online_Retail_new))*100)
```

```
missing_values
```

```
##      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
```

8. Number of transactions with missing CustomerID records by countries

```
missing<-Online_Retail_new %>% filter(is.na(CustomerID)) %>% group_by(Country) %>% summarise(count_by_c)
```

```
View(missing)
```

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

```
difference_days<-Online_Retail_new %>% select(CustomerID,New_Invoice_Date) %>% group_by(CustomerID) %>%
View(difference_days)
mean(difference_days$days)
```

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

On an average,customers come back after 38 days to the website for their next shopping.

10. Return rate for the French customers

```
cancelled_customers <- Online_Retail_new %>% filter(Country=='France',Quantity<0) %>% summarise(count =
Total_customers<- Online_Retail_new %>% filter(Country=='France') %>% count()
```

```
return_rate_french_cust=((cancelled_customers/Total_customers)*100)
```

```
head(return_rate_french_cust)
```

```
##      count
## 1 1.741264
```

11. Product that has generated the highest Revenue for the retailer

```
item_sum<-Online_Retail_new %>% group_by(Description) %>% summarise(sum_cnt = sum(TransactionValue)) %>%  
head(item_sum)
```

```
## # A tibble: 6 x 2  
##   Description          sum_cnt  
##   <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.
```

#DOTCOM POSTAGE generates highest revenue for the retailer

12. Unique customers in the dataset

```
unique_cust<- Online_Retail_new %>% distinct(CustomerID) %>% summarise(ncount = n())  
head(unique_cust)
```

```
##   ncount  
## 1    4373
```

#There are 4373 unique customers in the dataset