Business\_Analytics\_Assignment1

Kruthi Tatavarthy

3/13/2022

## R Markdown

This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. For more details on using R Markdown see <http://rmarkdown.rstudio.com>.

When you click the **Knit** button a document will be generated that includes both content as well as the output of any embedded R code chunks within the document. You can embed an R code chunk like this:

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)  
library(magrittr)  
library(reshape2)  
  
Online\_Retail\_data <- read.csv("C:/Users/Kruthi Tatavarthy/Documents/Masters in Business Analytics/Sem-1/Business Analytics/Online\_Retail.csv")  
  
head(Online\_Retail\_data)

## 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

#Question 1:   
  
attach(Online\_Retail\_data)  
  
glimpse(Online\_Retail\_data)

## Rows: 541,909  
## Columns: 8  
## $ InvoiceNo <chr> "536365", "536365", "536365", "536365", "536365", "536365"~  
## $ StockCode <chr> "85123A", "71053", "84406B", "84029G", "84029E", "22752", ~  
## $ Description <chr> "WHITE HANGING HEART T-LIGHT HOLDER", "WHITE METAL LANTERN~  
## $ Quantity <int> 6, 6, 8, 6, 6, 2, 6, 6, 6, 32, 6, 6, 8, 6, 6, 3, 2, 3, 3, ~  
## $ InvoiceDate <chr> "12/1/2010 8:26", "12/1/2010 8:26", "12/1/2010 8:26", "12/~  
## $ UnitPrice <dbl> 2.55, 3.39, 2.75, 3.39, 3.39, 7.65, 4.25, 1.85, 1.85, 1.69~  
## $ CustomerID <int> 17850, 17850, 17850, 17850, 17850, 17850, 17850, 17850, 17~  
## $ Country <chr> "United Kingdom", "United Kingdom", "United Kingdom", "Uni~

dim(Online\_Retail\_data)

## [1] 541909 8

summary(Online\_Retail\_data)

## 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 mutate, n(), filter   
  
count\_countries <- Online\_Retail\_data %>% group\_by(Country) %>% summarise(country\_txn = n())  
total\_txns <- nrow(Online\_Retail\_data) #sum(count\_countries$country\_txn)  
percent <- round(100\*(count\_countries$country\_txn/total\_txns), digits=2)  
print(subset(cbind(count\_countries,percent),percent > 1))

## Country country\_txn percent  
## 11 EIRE 8196 1.51  
## 14 France 8557 1.58  
## 15 Germany 9495 1.75  
## 36 United Kingdom 495478 91.43

#Question 2:  
  
  
Online\_Retail\_data1 <- Online\_Retail\_data %>% mutate(TransactionValue = Quantity \* UnitPrice)  
summarise(Online\_Retail\_data1)

## data frame with 0 columns and 1 row

glimpse(Online\_Retail\_data1)

## Rows: 541,909  
## Columns: 9  
## $ InvoiceNo <chr> "536365", "536365", "536365", "536365", "536365", "53~  
## $ StockCode <chr> "85123A", "71053", "84406B", "84029G", "84029E", "227~  
## $ Description <chr> "WHITE HANGING HEART T-LIGHT HOLDER", "WHITE METAL LA~  
## $ Quantity <int> 6, 6, 8, 6, 6, 2, 6, 6, 6, 32, 6, 6, 8, 6, 6, 3, 2, 3~  
## $ InvoiceDate <chr> "12/1/2010 8:26", "12/1/2010 8:26", "12/1/2010 8:26",~  
## $ UnitPrice <dbl> 2.55, 3.39, 2.75, 3.39, 3.39, 7.65, 4.25, 1.85, 1.85,~  
## $ CustomerID <int> 17850, 17850, 17850, 17850, 17850, 17850, 17850, 1785~  
## $ Country <chr> "United Kingdom", "United Kingdom", "United Kingdom",~  
## $ TransactionValue <dbl> 15.30, 20.34, 22.00, 20.34, 20.34, 15.30, 25.50, 11.1~

#Question 3:   
  
  
attach(Online\_Retail\_data1)

## The following objects are masked from Online\_Retail\_data:  
##   
## Country, CustomerID, Description, InvoiceDate, InvoiceNo, Quantity,  
## StockCode, UnitPrice

new\_dataframe <- summarise(group\_by(Online\_Retail\_data1, Online\_Retail\_data1$Country), sum(TransactionValue))  
Transaction\_data <- filter(new\_dataframe, new\_dataframe$`sum(TransactionValue)` >130000)  
glimpse(Transaction\_data)

## Rows: 6  
## Columns: 2  
## $ `Online\_Retail\_data1$Country` <chr> "Australia", "EIRE", "France", "Germany"~  
## $ `sum(TransactionValue)` <dbl> 137077.3, 263276.8, 197403.9, 221698.2, ~

#Question 4:  
  
Temp=strptime(Online\_Retail\_data1$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"

Online\_Retail\_data1$New\_Invoice\_Date <- as.Date(Temp)  
Online\_Retail\_data1$New\_Invoice\_Date[20000]-Online\_Retail\_data1$New\_Invoice\_Date[10]

## Time difference of 8 days

Online\_Retail\_data1$Invoice\_Day\_Week = weekdays(Online\_Retail\_data1$New\_Invoice\_Date)  
Online\_Retail\_data1$New\_Invoice\_Hour = as.numeric(format(Temp, "%H"))  
Online\_Retail\_data1$New\_Invoice\_Month = as.numeric(format(Temp, "%m"))  
  
# a) Show the percentage of transactions (by numbers) by days of the week   
  
a\_no<-summarise(group\_by(Online\_Retail\_data1,Invoice\_Day\_Week),Transaction\_Value=n\_distinct(InvoiceNo))  
a\_no1<-mutate(a\_no, transaction\_percent=(Transaction\_Value/sum(Transaction\_Value))\*100)  
a\_no1

## # A tibble: 6 x 3  
## Invoice\_Day\_Week Transaction\_Value transaction\_percent  
## <chr> <int> <dbl>  
## 1 Friday 4184 16.2   
## 2 Monday 4138 16.0   
## 3 Sunday 2381 9.19  
## 4 Thursday 5660 21.9   
## 5 Tuesday 4722 18.2   
## 6 Wednesday 4815 18.6

# b) Show the percentage of transactions (by transaction volume) by days of the week  
  
b\_no <- summarise(group\_by(Online\_Retail\_data1,Invoice\_Day\_Week),Transaction\_Volume=sum(TransactionValue))  
b\_no1 <- mutate(b\_no,percentage=(Transaction\_Volume/sum(Transaction\_Volume))\*100)  
b\_no1

## # A tibble: 6 x 3  
## Invoice\_Day\_Week Transaction\_Volume 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

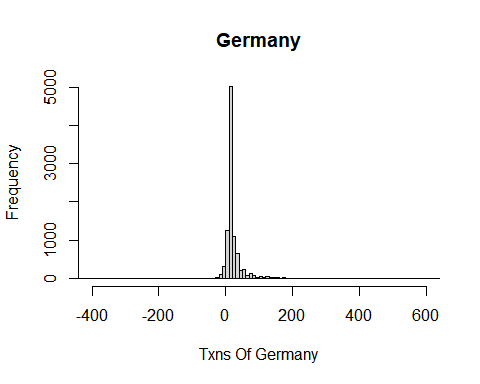
# c) Show the percentage of transactions (by transaction volume) by month of the year  
  
c\_no<-summarise(group\_by(Online\_Retail\_data1,New\_Invoice\_Month),Transaction\_Volume=sum(TransactionValue))  
c\_no1<-mutate(c\_no,percentage=(Transaction\_Volume/sum(Transaction\_Volume))\*100)  
c\_no1

## # A tibble: 12 x 3  
## New\_Invoice\_Month Transaction\_Volume 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

#d)What was the date with the highest number of transactions from Australia?   
  
Online\_Retail\_data1 %>% filter(Country == 'Australia') %>% group\_by(New\_Invoice\_Date) %>% summarise(max=max(TransactionValue))

## # A tibble: 49 x 2  
## New\_Invoice\_Date max  
## <date> <dbl>  
## 1 2010-12-01 51   
## 2 2010-12-08 71.4   
## 3 2010-12-14 -6.25  
## 4 2010-12-17 148.   
## 5 2011-01-06 1020   
## 6 2011-01-10 81.6   
## 7 2011-01-11 35.4   
## 8 2011-01-14 142.   
## 9 2011-01-17 47.4   
## 10 2011-01-19 38.2   
## # ... with 39 more rows

# Question 5:   
  
germany\_dataset <- subset(Online\_Retail\_data1$TransactionValue, Online\_Retail\_data1$Country == "Germany")  
hist(germany\_dataset, xlim=c(-400, 600), breaks = 150, xlab="Txns Of Germany", main="Germany")



# Question 6:  
  
Online\_Retail\_data2 <- na.omit(Online\_Retail\_data1)  
res <- summarise(group\_by(Online\_Retail\_data2,CustomerID), sum2= sum(TransactionValue))  
res[which.max(res$sum2),]

## # A tibble: 1 x 2  
## CustomerID sum2  
## <int> <dbl>  
## 1 14646 279489.

#14646 has highest number of transactions 279489  
data1 <- as.data.frame(table(Online\_Retail\_data1$CustomerID))  
data1[which.max(data1$Freq),]

## Var1 Freq  
## 4043 17841 7983

#17841 custId has highest nof txns  
  
# Question 7:  
  
colMeans(is.na(Online\_Retail\_data)\*100)

## InvoiceNo StockCode Description Quantity InvoiceDate UnitPrice   
## 0.00000 0.00000 0.00000 0.00000 0.00000 0.00000   
## CustomerID Country   
## 24.92669 0.00000

# Question 8:   
  
intial\_result <- Online\_Retail\_data %>% filter(is.na(CustomerID)) %>% group\_by(Country)  
glimpse(intial\_result$Country)

## chr [1:135080] "United Kingdom" "United Kingdom" "United Kingdom" ...

summary(intial\_result$Country)

## Length Class Mode   
## 135080 character character

# Question 9:   
  
visit <- Online\_Retail\_data1 %>%  
 group\_by(InvoiceNo, CustomerID, Country, New\_Invoice\_Date, New\_Invoice\_Month, New\_Invoice\_Hour, Invoice\_Day\_Week) %>%  
 summarise(orderVal = sum(TransactionValue)) %>%  
 mutate(recent = Sys.Date() - New\_Invoice\_Date) %>%  
 ungroup()

## `summarise()` has grouped output by 'InvoiceNo', 'CustomerID', 'Country',  
## 'New\_Invoice\_Date', 'New\_Invoice\_Month', 'New\_Invoice\_Hour'. You can override  
## using the `.groups` argument.

visit$recent <- as.character(visit$recent)  
visit$recentDays <- sapply(visit$recent, FUN = function(x) {strsplit(x, split = '[ ]')[[1]][1]})  
visit$recentDays <- as.integer(visit$recentDays)  
head(visit, n = 5)

## # A tibble: 5 x 10  
## InvoiceNo CustomerID Country New\_Invoice\_Date New\_Invoice\_Month  
## <chr> <int> <chr> <date> <dbl>  
## 1 536365 17850 United Kingdom 2010-12-01 12  
## 2 536366 17850 United Kingdom 2010-12-01 12  
## 3 536367 13047 United Kingdom 2010-12-01 12  
## 4 536368 13047 United Kingdom 2010-12-01 12  
## 5 536369 13047 United Kingdom 2010-12-01 12  
## # ... with 5 more variables: New\_Invoice\_Hour <dbl>, Invoice\_Day\_Week <chr>,  
## # orderVal <dbl>, recent <chr>, recentDays <int>

attach(visit)

## The following objects are masked from Online\_Retail\_data1:  
##   
## Country, CustomerID, InvoiceNo

## The following objects are masked from Online\_Retail\_data:  
##   
## Country, CustomerID, InvoiceNo

custVisit <- Online\_Retail\_data1 %>%  
 group\_by(CustomerID, Country) %>%  
 summarise(orders = n\_distinct(InvoiceNo), revenue = sum(TransactionValue),   
 mostDay = names(which.max(table(Invoice\_Day\_Week))), mostHour = names(which.max(table(New\_Invoice\_Hour))),  
 recency = min(recentDays))%>%  
 ungroup()

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

head(custVisit)

## # A tibble: 6 x 7  
## CustomerID Country orders revenue mostDay mostHour recency  
## <int> <chr> <int> <dbl> <chr> <chr> <int>  
## 1 12346 United Kingdom 2 0 Tuesday 10 3747  
## 2 12347 Iceland 7 4310 Tuesday 14 3747  
## 3 12348 Finland 4 1797. Thursday 19 3747  
## 4 12349 Italy 1 1758. Monday 9 3747  
## 5 12350 Norway 1 334. Wednesday 16 3747  
## 6 12352 Norway 11 1545. Tuesday 14 3747

# Question 10:   
france\_dataframe <- (filter(Online\_Retail\_data1,Online\_Retail\_data1$Country=="France"))  
cancelled\_df <- nrow(subset(france\_dataframe, Online\_Retail\_data1$TransactionValue < 0))  
cancelled\_df

## [1] 9290

not\_cancelled\_df <- nrow(Online\_Retail\_data1) - cancelled\_df  
(cancelled\_df/not\_cancelled\_df)

## [1] 0.01744211