PBA Assignment 1

Mercy Angelina - 109006278

2022-10-26

Solution: The Nuts and Bolts of Data Analysis Using R

a) Find out the path to the directory containing the data and load it onto R

```
#Read the CSV file
online_retail <- read.csv("/Users/cipta/OneDrive/Documents/SCHOOL/Programming for Business
↪  Analytics/online_retail.csv", header=TRUE)
library(knitr)
kable(head(online_retail), caption = "Online Retail Data")
```

Table 1: Online Retail Data

InvoiceNo	StockCode	Description	Quantity	InvoiceDate	UnitPrice	CustomerID	Country
536365	85123A	WHITE HANGING HEART T-LIGHT HOLDER	6	12/1/10 8:26	2.55	17850	United Kingdom
536365	71053	WHITE METAL LANTERN	6	12/1/10 8:26	3.39	17850	United Kingdom
536365	84406B	CREAM CUPID HEARTS COAT HANGER	8	12/1/10 8:26	2.75	17850	United Kingdom
536365	84029G	KNITTED UNION FLAG HOT WATER BOTTLE	6	12/1/10 8:26	3.39	17850	United Kingdom
536365	84029E	RED WOOLLY HOTTIE WHITE HEART.	6	12/1/10 8:26	3.39	17850	United Kingdom
536365	22752	SET 7 BABUSHKA NESTING BOXES	2	12/1/10 8:26	7.65	17850	United Kingdom

```
#Check the data structure
str(online_retail)
```

```
## 'data.frame':    541909 obs. of  8 variables:
##  $ InvoiceNo  : chr  "536365" "536365" "536365" "536365" ...
##  $ StockCode  : chr  "85123A" "71053" "84406B" "84029G" ...
##  $ Description: chr  "WHITE HANGING HEART T-LIGHT HOLDER" "WHITE METAL LANTERN" "CREAM CUPID HEARTS COAT HANG
##  $ Quantity   : int  6 6 8 6 6 2 6 6 6 32 ...
##  $ InvoiceDate: chr  "12/1/10 8:26" "12/1/10 8:26" "12/1/10 8:26" "12/1/10 8:26" ...
##  $ UnitPrice  : num  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 17850 13047 ...
##  $ Country    : chr  "United Kingdom" "United Kingdom" "United Kingdom" "United Kingdom" ...
```

b) Convert InvoiceDate to date class and subset the data

```
online_retail$NewDate <- strptime(as.character(online_retail$InvoiceDate), "%m/%d/%y")

online_retail$NewDate[261324]
```

```
## [1] "2011-07-13 CST"
```

```
class(online_retail$NewDate)
```

```
## [1] "POSIXlt" "POSIXt"
```

```
online_retail$NewInvoiceDate <- format(online_retail$NewDate, "%Y-%m-%d")

#Check InvoiceDate new format
online_retail$NewInvoiceDate[261324]
```

```
## [1] "2011-07-13"
```

```
library(knitr)
kable(head(online_retail), caption = "Online Retail Data with new InvoiceDate format")
```

Table 2: Online Retail Data with new InvoiceDate format

InvoiceNo	StockCode	Description	Quantity	InvoiceDate	UnitPrice	CustomerID	Country	NewDate	NewInvoiceDate
536365	85123A	WHITE HANGING HEART T-LIGHT HOLDER	6	12/1/10 8:26	2.55	17850	United Kingdom	2010-12-01	2010-12-01
536365	71053	WHITE METAL LANTERN	6	12/1/10 8:26	3.39	17850	United Kingdom	2010-12-01	2010-12-01
536365	84406B	CREAM CUPID HEARTS COAT HANGER	8	12/1/10 8:26	2.75	17850	United Kingdom	2010-12-01	2010-12-01
536365	84029G	KNITTED UNION FLAG HOT WATER BOTTLE	6	12/1/10 8:26	3.39	17850	United Kingdom	2010-12-01	2010-12-01
536365	84029E	RED WOOLLY HOTTIE WHITE HEART.	6	12/1/10 8:26	3.39	17850	United Kingdom	2010-12-01	2010-12-01
536365	22752	SET 7 BABUSHKA NESTING BOXES	2	12/1/10 8:26	7.65	17850	United Kingdom	2010-12-01	2010-12-01

```
subset_df <- online_retail[online_retail$NewInvoiceDate >= "2011-07-01" & online_retail$NewInvoiceDate <=
↪  "2011-08-31", ]
subset_length <- length(subset_df$InvoiceNo)
```

- Check to see 3,664 unique transactions

```
length(unique(subset_df$InvoiceNo))
```

```
## [1] 3664
```

c) Use for-loops to

1) Compute the mean of Quantity and UnitPrice

```
quantity_average = 0
unit_average = 0
for (i in 1: subset_length) {
  quantity_average <- quantity_average + subset_df$Quantity[i]
  unit_average <- unit_average + subset_df$UnitPrice[i]
}
quantity_average = quantity_average/subset_length
unit_average = unit_average/subset_length
```

- Mean of Quantity

```
print(quantity_average)
```

```
## [1] 10.65901
```

```
print(unit_average)
```

```
## [1] 4.308608
```

2) Determine types of each column

```
for (i in 1: length(subset_df)) {
  print(class(subset_df[,i]))
}
```

```
## [1] "character"
## [1] "character"
## [1] "character"
## [1] "integer"
## [1] "character"
## [1] "numeric"
## [1] "integer"
## [1] "character"
## [1] "POSIXlt" "POSIXt"
## [1] "character"
```

3) Compute the number of unique values in each column

```
for (i in 1: length(subset_df)) {
  print(length(unique(subset_df[,i])))
}
```

```
## [1] 3664
## [1] 2982
## [1] 2953
## [1] 287
## [1] 3343
## [1] 447
## [1] 1541
## [1] 28
## [1] 52
## [1] 52
```

d) Subset the data from U.K., Netherlands, and Australia

```
country_df <- subset_df[subset_df$Country == 'United Kingdom' | subset_df$Country == 'Netherlands' |
↪  subset_df$Country == 'Australia' , ]
length(unique(country_df$Country))
```

```
## [1] 3
```

4) Report the average and standard deviation of UnitPrice

- Average

```
x = mean(country_df$UnitPrice)
format(round(x, 3), nsmall = 3)
```

```
## [1] "4.344"
```

- Standard Deviation

```
y = sd(country_df$UnitPrice)
format(round(y, 3), nsmall= 3)
```

```
## [1] "98.961"
```

5) Report the number of unique transactions in these countries

```
length(unique(country_df$InvoiceNo))
```

```
## [1] 3332
```

6) Report how many customers residing in these countries made transactions in July and August of 2011?

```
length(unique(country_df$CustomerID))-1
```

```
## [1] 1379
```

e) Do we see any customers who made a refund?

Yes, we do see customers made refunds.

7) How many customers made a refund (exclude the observations without the CustomerID)?

```
subset_df$Refund <- substr(subset_df$InvoiceNo, 1, 1)
cust_refund <- subset(subset_df, subset_df$Refund == "C")
length(unique(cust_refund$CustomerID))-1
```

```
## [1] 381
```

f) Some customers made purchases without logging into the e-commerce site. This would create records of transactions for which the CustomerID is missing (i.e. NA). Create a variable called Sales by multiplying the Quantity and the UnitPrice.

8) Calculate the total sales amount for those that are missing the CustomerID (i.e. NA)

```
subset_df$Sales <- subset_df$Quantity*subset_df$UnitPrice
custNA <- subset(subset_df, is.na(subset_df$CustomerID) == TRUE)
sum(custNA$Sales)
```

```
## [1] 173374.1
```

9) How many transactions were made without the customers logging into the e-commerce site?

```
length(unique(custNA$InvoiceNo))
```

```
## [1] 527
```

Extra Credit

EC1) Create a variable containing the monthly aggregate spending for each customer

```
#July spending
july_trans <- subset_df[subset_df$NewInvoiceDate >= "2011-07-01" & subset_df$NewInvoiceDate <=
↪  "2011-07-31", ]
#July aggregate spending for each customer
agg_cust_july <- tapply(X = july_trans$Sales, INDEX = july_trans$CustomerID, FUN = sum)

#August spending
aug_trans <- subset_df[subset_df$NewInvoiceDate >= "2011-08-01" & subset_df$NewInvoiceDate <=
↪  "2011-08-31", ]
#August aggregate spending for each customer
agg_cust_aug <- tapply(X = aug_trans$Sales, INDEX = aug_trans$CustomerID, FUN = sum)
```

EC2) Report the IDs and the monthly purchase amount of the five customers who have spent the most money in July 2011

```
agg_cust_july <- sort(-agg_cust_july)
head(agg_cust_july*(-1),5)
```

```
##    14156    18102    14911    17949    14088
## 26464.99 19889.16 13445.33 11590.58  9038.69
```