Data Science Assignment

Jorge Portugal

May 25, 2021

Initial Thoughts

```
data <- read.csv("C:/Users/J/Desktop/New folder/data.csv.csv")

#The following code is  just me brainstorming ideas as to how to approach the problem

colnames(data)
```

```
##  [1] "Region"              "Country"             "Item.Type"
##  [4] "Fiscal.Year"         "Sales.Channel"       "Order.Priority"
##  [7] "Order.Date"          "Order.ID"            "Ship.Date"
## [10] "Units.Sold"          "Unit.Price"          "Unit.Cost"
## [13] "Total.Revenue"       "Total.Cost"          "Total.Profit"
## [16] "Profit.as...of.Cost"
```

```
unique(data$Fiscal.Year)
```

```
## [1] 2010 2011 2012 2016 2017 2013 2014 2015
```

```
dat<-subset(data, Fiscal.Year==2017)
#I initially thought question wanted only the latest year's data to make decisions for the next year.
#At this point, I didn't realize the Profit values were the same across every year
unique(dat$Item.Type)
```

```
##  [1] "Beverages"       "Office Supplies" "Vegetables"      "Fruits"
##  [5] "Cosmetics"       "Cereal"          "Meat"            "Personal Care"
##  [9] "Baby Food"       "Snacks"          "Household"       "Clothes"
```

```
unique(dat$Profit.as...of.Cost)
```

```
##  [1] 0.4926077 0.2404945 0.6942703 0.3482659 0.6602742 0.7564683 0.1568455
##  [8] 0.4422093 0.6013047 0.5658867 0.3297847 2.0491071
```

```
#I looked at the values of Item Type and Profit, and noticed there were only 12 different profit values
#It was straightforward from there, I then found the minimum profit value along with its
#corresponding item type, and that is the least profitable Item Type
mins<-min(dat[,"Profit.as...of.Cost"])
mins2<-dat[dat$Profit.as...of.Cost == mins, "Item.Type"]
mins2[1]
```

```
## [1] "Meat"
```

```r
#Did the same for the max, which results in the discovery of the most profitable item type
max<-max(dat[,"Profit.as...of.Cost"])
max2<-dat[dat$Profit.as...of.Cost == max, "Item.Type"]
max2[1]
```

```
## [1] "Clothes"
```

Problem 1

After this brainstorimg session, I came up with:

The following function, which takes in a dataset and a year as parameters , and returns a table containing each Item with its corresponding Average Profit value in descending order.

```r
myfunc<-function(data,year){
  sub<-subset(data, Fiscal.Year==year)
  Avg.Profit<-c(mean(sub$Profit.as...of.Cost[sub$Item.Type=="Beverages"]),
               mean(sub$Profit.as...of.Cost[sub$Item.Type=="Office Supplies"]),
               mean(sub$Profit.as...of.Cost[sub$Item.Type=="Vegetables"]),
               mean(sub$Profit.as...of.Cost[sub$Item.Type=="Fruits"]),
               mean(sub$Profit.as...of.Cost[sub$Item.Type=="Cosmetics"]),
               mean(sub$Profit.as...of.Cost[sub$Item.Type=="Cereal"]),
               mean(sub$Profit.as...of.Cost[sub$Item.Type=="Meat"]),
               mean(sub$Profit.as...of.Cost[sub$Item.Type=="Personal Care"]),
               mean(sub$Profit.as...of.Cost[sub$Item.Type=="Baby Food"]),
               mean(sub$Profit.as...of.Cost[sub$Item.Type=="Snacks"]),
               mean(sub$Profit.as...of.Cost[sub$Item.Type=="Household"]),
               mean(sub$Profit.as...of.Cost[sub$Item.Type=="Clothes"]))
  Avg.Profit<-round(Avg.Profit,5)
  Item.Type<-unique(sub$Item.Type)

  dat<-data.frame(Item.Type,Avg.Profit)
  dat<-dat[order(-Avg.Profit),]
  return(dat)
  }

myfunc(data, 2017)
```

```
##          Item.Type Avg.Profit
## 12         Clothes    2.04911
## 6           Cereal    0.75647
## 3       Vegetables    0.69427
## 5        Cosmetics    0.66027
## 9        Baby Food    0.60130
## 10          Snacks    0.56589
## 1        Beverages    0.49261
## 8    Personal Care    0.44221
## 4           Fruits    0.34827
## 11       Household    0.32978
## 2  Office Supplies    0.24049
## 7             Meat    0.15685
```

Results indicated Clothes and Cereal are the most profitable and should be sold more, while Office Supplies and Meat were the least profitable and should be dropped.

This function is flexible and builds upon my initial work. It can check for profitability per year (there were few unique values per Year in this specific dataset), as well as calculate the Average Profit Value per item. (were were also few unique Profit values)

```
myfunc2<-function(data,year){
  sub<-subset(data, Fiscal.Year==year)
  Avg.Profit<-c(mean(sub$Profit.as...of.Cost[sub$Region=="Asia"]),
                mean(sub$Profit.as...of.Cost[sub$Region=="Europe"]),
                mean(sub$Profit.as...of.Cost[sub$Region=="Sub-Saharan Africa"]),
                mean(sub$Profit.as...of.Cost[sub$Region=="North America"]),
                mean(sub$Profit.as...of.Cost[sub$Region=="Middle East and North Africa"]),
                mean(sub$Profit.as...of.Cost[sub$Region=="Central America and the Caribbean"]),
                mean(sub$Profit.as...of.Cost[sub$Region=="Australia and Oceania"]))
  Avg.Profit<-round(Avg.Profit,5)
  Region<-unique(sub$Region)

  dat<-cbind(Region,Avg.Profit)
  dat<-dat[order(-Avg.Profit),]
  return(dat)
}

myfunc2(data, 2017)
```

```
##      Region                              Avg.Profit
## [1,] "North America"                     "0.63522"
## [2,] "Sub-Saharan Africa"                "0.63256"
## [3,] "Australia and Oceania"             "0.62177"
## [4,] "Asia"                              "0.61619"
## [5,] "Middle East and North Africa"      "0.60995"
## [6,] "Central America and the Caribbean" "0.60254"
## [7,] "Europe"                            "0.60241"
```

Similar approach to Item Type profitability.

The results indicate North America and Sub-Saharan Africa should be given preference over the other Regions, as they have the highest average profit.

Problem 2

The way I approached this question is to create a function that would calculate the number of days between Order Date and Ship Date for every item (called Diff), obtain the mean Diff per item, and return a list containing each item type with its corresponding Diff in ascending order.

```
Asia<-subset(data, Region=="Asia")
Europe<-subset(data, Region=="Europe")
Sub.Saharan.Africa<-subset(data, Region=="Sub-Saharan Africa")
North.America<-subset(data, Region=="North America")
MiddleEast.NorthAfrica<-subset(data, Region=="Middle East and North Africa")
CentralAmerica.Caribbean<-subset(data, Region=="Central America and the Caribbean")
Australia.Oceania<-subset(data, Region=="Australia and Oceania")
```

```r
myfunc4<-function(sub){
  sub$Order.Date<-as.Date(sub$Order.Date,format = "%m/%d/%y")
  sub$Ship.Date<-as.Date(sub$Ship.Date,format = "%m/%d/%y")
  sub$Diff<-difftime(sub$Ship.Date,sub$Order.Date, units="days")

  Time<-c(mean(sub$Diff[sub$Item.Type=="Beverages"]),
              mean(sub$Diff[sub$Item.Type=="Office Supplies"]),
              mean(sub$Diff[sub$Item.Type=="Vegetables"]),
              mean(sub$Diff[sub$Item.Type=="Fruits"]),
              mean(sub$Diff[sub$Item.Type=="Cosmetics"]),
              mean(sub$Diff[sub$Item.Type=="Cereal"]),
              mean(sub$Diff[sub$Item.Type=="Meat"]),
              mean(sub$Diff[sub$Item.Type=="Personal Care"]),
              mean(sub$Diff[sub$Item.Type=="Baby Food"]),
              mean(sub$Diff[sub$Item.Type=="Snacks"]),
              mean(sub$Diff[sub$Item.Type=="Household"]),
              mean(sub$Diff[sub$Item.Type=="Clothes"]))
  Time<-round(Time,2)
  df<-data.frame(Item.Type=unique(sub$Item.Type),Time=Time)
  df<-df[order(Time),]
  return(df)
}


myfunc4(Asia)
```

```
##          Item.Type       Time
## 1        Beverages 24.57 days
## 3        Cosmetics 24.62 days
## 6             Meat 24.74 days
## 11       Household 25.09 days
## 4           Fruits 25.17 days
## 2  Office Supplies 25.20 days
## 5    Personal Care 25.36 days
## 10      Vegetables 25.51 days
## 7          Clothes 25.59 days
## 8           Cereal 25.68 days
## 9        Baby Food 25.91 days
## 12          Snacks 25.96 days
```

```r
myfunc4(Europe)
```

```
##          Item.Type       Time
## 10         Clothes 24.54 days
## 7           Snacks 24.71 days
## 4       Vegetables 24.88 days
## 5           Cereal 24.89 days
## 12          Fruits 24.95 days
## 9        Household 24.97 days
## 8             Meat 25.11 days
## 1  Office Supplies 25.13 days
```

```
## 6        Baby Food 25.23 days
## 3    Personal Care 25.28 days
## 11       Beverages 25.38 days
## 2        Cosmetics 26.17 days
```

```r
myfunc4(Sub.Saharan.Africa)
```

```
##          Item.Type       Time
## 7           Cereal 24.28 days
## 10      Vegetables 24.31 days
## 1           Snacks 24.42 days
## 5          Clothes 24.44 days
## 9        Baby Food 24.44 days
## 6    Personal Care 25.01 days
## 3        Household 25.03 days
## 12          Fruits 25.08 days
## 11 Office Supplies 25.14 days
## 8        Cosmetics 25.25 days
## 2        Beverages 25.30 days
## 4             Meat 25.74 days
```

```r
myfunc4(North.America)
```

```
##          Item.Type       Time
## 6             Meat 22.95 days
## 10       Baby Food 24.59 days
## 5        Beverages 24.69 days
## 11         Clothes 24.73 days
## 1           Fruits 24.86 days
## 2           Cereal 25.29 days
## 4  Office Supplies 25.38 days
## 12   Personal Care 25.62 days
## 3       Vegetables 25.87 days
## 8        Cosmetics 26.24 days
## 9           Snacks 26.43 days
## 7        Household 27.09 days
```

```r
myfunc4(MiddleEast.NorthAfrica)
```

```
##          Item.Type       Time
## 2           Fruits 24.09 days
## 1          Clothes 24.36 days
## 5        Cosmetics 24.79 days
## 6    Personal Care 25.06 days
## 10       Household 25.13 days
## 11          Snacks 25.17 days
## 12       Baby Food 25.20 days
## 3        Beverages 25.30 days
## 8           Cereal 25.30 days
## 9             Meat 25.36 days
## 7       Vegetables 25.56 days
## 4  Office Supplies 25.65 days
```

```r
myfunc4(CentralAmerica.Caribbean)
```

```
##          Item.Type       Time
## 5        Cosmetics 23.97 days
## 7           Fruits 24.15 days
## 9           Cereal 24.41 days
## 3        Baby Food 24.56 days
## 6  Office Supplies 24.67 days
## 10       Beverages 25.13 days
## 12          Snacks 25.24 days
## 11   Personal Care 25.28 days
## 1          Clothes 25.34 days
## 8       Vegetables 25.34 days
## 2             Meat 25.46 days
## 4        Household 26.14 days
```

```r
myfunc4(Australia.Oceania)
```

```
##          Item.Type       Time
## 4        Baby Food 22.93 days
## 5  Office Supplies 24.00 days
## 7           Fruits 24.45 days
## 10          Snacks 24.46 days
## 6    Personal Care 24.49 days
## 12       Beverages 24.88 days
## 8        Household 24.89 days
## 11            Meat 25.02 days
## 1          Clothes 25.31 days
## 9        Cosmetics 25.36 days
## 2       Vegetables 25.57 days
## 3           Cereal 26.16 days
```

This function has been run 7 times, once for each Region. The items at the top are the easiest to sell, while those at the bottom take the longest to sell.

```r
#sub<-data
#for(i in unique(dat$Region)){
#    nxt[i]<-mean(dat$Profit.as...of.Cost[dat$Region==i])
#  }
#nxt
```