Segment 8 Extra Credit

Britney Brown

2/8/2022

```
library(tidyverse)
library(dplyr)
```

```
original <-  read.csv("TV-Sales.csv", stringsAsFactors=FALSE)
df <- original

# change to numeric
change_to_int <- colnames(df)[-1]
for(i in 1:length(change_to_int)){
  df[,change_to_int[i]] <- as.numeric(df[,change_to_int[i]])}

# converting to datetime object
df[['Date']] <- as.POSIXct(df[['Date']],format = "%Y-%m-%d")

head(df)
```

```
##         Date S1 S2 S3 S4 S5 S6 S7 S8 S9 S10
## 1 2017-01-01 76 79 57 50 30 77 62 20 40  42
## 2 2017-01-02 64 91 42 59 46 64 61 25 35  51
## 3 2017-01-03 53 66 60 70 82 28 NA 34 49  69
## 4 2017-01-04 65 72 43 53 39 75 67 35 43  49
## 5 2017-01-05 58 69 43 71 42 64 52 28 29  43
## 6 2017-01-06 55 74 66 72 73 44 47 46 65  56
```

4.1 Which store had the highest mean sale in 2017?

```
df_dates <- df %>% dplyr::mutate(year = lubridate::year(Date),
                                 month = lubridate::month(Date),
                                 day = lubridate::day(Date))

df_2017 <- df_dates[which(df_dates$year == 2017),-c(1,12,13,14)]

means_2017 <- as.data.frame(colMeans(df_2017, na.rm = TRUE))
means_2017 <- cbind(store = rownames(means_2017), means_2017)
colnames(means_2017)[2] <- "means"
means_2017[which(means_2017$means == max(means_2017$means)),1]
```

```
## [1] "S2"
```

4.2 Which day showed the highest variance in sales across different stores?

```
library(matrixStats)
dailyvar <- rowVars(as.matrix(df[,-1]), na.rm = TRUE)
maxvar <- max(dailyvar)
df[which(dailyvar == maxvar),'Date']
```

```
## [1] "2017-11-07 PST"
```

4.3 Which year showed the highest median sale for the store S5?

```
df$Year <- format(df$Date, format="%Y")
highest_mean <- df %>% select(S5,Year) %>% group_by(Year) %>%
  summarise(median = median(S5,na.rm = TRUE))

as.numeric(highest_mean[which(highest_mean$median == max(highest_mean$median)),'Year'])
```

```
## [1] 2019
```

4.4 Which store recorded the highest number of sales for the largest number of days?

```
df2 <- original
df2[is.na(df2)] <- 0 # replace NA with 0
tbl <- table(colnames(df2[,-1])[max.col(df2[,-1],ties.method="last")])
rownames(as.data.frame(tbl[order(-tbl)][1]))
```

```
## [1] "S2"
```

4.5 Which store ranks 5th in the cumulative number of units sold over the 3-year interval?

```
stores <- as.data.frame(sapply(original[,-1],as.numeric))
cumSales <- colSums(stores, na.rm = TRUE)
rownames(as.data.frame(cumSales[order(-cumSales)][5]))
```

```
## [1] "S7"
```

4.6 Your program should create a file named repaired.csv in the directory which contains the same data as TV-Sales.csv, but with "N/A" values replaced with the median sale of that store, over the entire 3-year interval. Retain the header row found in TV-Sales.csv.

```r
medians <- as.numeric(apply(stores, 2, median, na.rm = TRUE))

for(i in 1:10){
  stores[,i][is.na(stores[,i])] <- medians[i]
}

new_data <- cbind(Date = original[,1],stores)
write.csv(new_data,"repaired.csv", row.names = FALSE)
```

4.7 After imputing the missing values, plot the cumulative sale of each store as a function of time. Your plot should have a legend, a grid along the y-axis with a cell size of 10, labeled y-ticks at each multiple of 10, and x-ticks corresponding to the first days of each quarter (3 month period). Save the plot as hw1/q4/plot.png.

```r
cumData <- cbind(Date = new_data[,1],cumsum(new_data[,-1]))

# define color palette
library(RColorBrewer)
n <- 10
qual_col_pals = brewer.pal.info[brewer.pal.info$category == 'qual',]
col_vector = unlist(mapply(brewer.pal, qual_col_pals$maxcolors, rownames(qual_col_pals)))
col=sample(col_vector, n)

jpeg(file="extra_plot.jpeg")
matplot(y = cumData[,-1], type = 'l',
        xlab = "Date",
        ylab = "Cumulative Sales",
        col=col,
        axes = FALSE)

yticks <- seq(0,80000, 10000)
axis(2,at=yticks,labels=T, las = 2, cex.axis = 0.8)

xticks <- seq(0,1100, 90)
xlabs <- c("Jan17","Apr17", "July17","Oct17",
           "Jan18","Apr18", "July18","Oct18",
           "Jan19","Apr19", "July19","Oct19","Jan20")


axis(1,at=xticks,labels=xlabs, las = 2, cex.axis = 0.8)

grid(NULL,NULL)

legend("topleft",
       colnames(cumData[,-1]),
       col=col,
       cex=0.7,
       fill=col)
dev.off()
```

5.2 Which store makes the maximum sales on Sundays?

```
df$day <- weekdays(df$Date)
df2 <- df[which(df$day == "Sunday"),-1]
df3 <- colSums(df2[,1:10], na.rm = TRUE)
rownames(as.data.frame(df3[order(-df3)][1]))
```

```
## [1] "S2"
```

5.3 Find all stores with total sales in December lower than those of S5.

```
totalsales <- df_dates[which(df_dates$month == 12),-1]
totalsales <- data.frame(means = colSums(totalsales[,1:10], na.rm = TRUE))
totalsales$store <- rownames(totalsales)
rownames(totalsales) <- NULL
orderedsales <- totalsales[order(totalsales$means),]
orderedsales[c(1:3),2]
```

```
## [1] "S8"  "S9"  "S10"
```

5.4 Which store recorded the highest number of sales for the largest number of days?

```
df2 <- original
df2[is.na(df2)] <- 0 # replace NA with 0
tbl <- table(colnames(df2[,-1])[max.col(df2[,-1],ties.method="last")])
rownames(as.data.frame(tbl[order(-tbl)][1]))
```

```
## [1] "S2"
```

5.5 What week in 2019 has the highest total sales across all the stores?

```r
df3 <- df[,1:11]
df3$Week <- as.numeric(strftime(df3$Date, format = "%V")) - 1 # to match python syntax
df3$Year <- strftime(df3$Date, format = "%Y")
df3 <- df3[which(df3$Year == "2019"),-1]
df3 <- df3[,1:11]

df3 <- df3 %>%
  group_by(Week) %>%
  summarise(S1 = sum(S1),
            S2 = sum(S2),
            S3 = sum(S3),
            S4 = sum(S4),
            S5 = sum(S5),
            S6 = sum(S6),
            S7 = sum(S7),
            S8 = sum(S8),
            S9 = sum(S9),
            S10 = sum(S10))

df3$totals <- rowSums(df3[,2:11], na.rm = TRUE)

final <- df3[which(df3$totals == max(df3$totals)),c("Week","totals")]
# corresponds to the week from 2019-09-16 to 2019-09-22, matches to sql segment
final$Week
```

```
## [1] 37
```