Week 2 Exercises

Wanida Ruangsiriluk

October 31, 2023

Please complete all exercises below. You may use stringr, lubridate, or the forcats library.

Place this at the top of your script: library(stringr) library(lubridate) library(forcats)

Exercise 1

Read the sales_pipe.txt file into an R data frame as sales.

```
library(stringr)
library(lubridate)
```

```
##
## Attaching package: 'lubridate'

## The following objects are masked from 'package:base':
##
##     date, intersect, setdiff, union
```

```
library(forcats)


setwd("..")
sales <- read.delim("Week_2/Data/sales_pipe.txt"
                     ,stringsAsFactors=FALSE
                     ,sep = "|"
 )
```

Exercise 2

You can extract a vector of columns names from a data frame using the colnames() function. Notice the first column has some odd characters. Change the column name for the FIRST column in the sales date frame to Row.ID.

Note: You will need to assign the first element of colnames to a single character.

```
colnames(sales)
```

```
##  [1] "XO..Row.ID"    "Order.ID"      "Order.Date"    "Ship.Date"
##  [5] "Ship.Mode"     "Customer.ID"   "Customer.Name" "Segment"
##  [9] "Country"       "City"          "State"         "Postal.Code"
## [13] "Region"        "Product.ID"    "Category"      "Sub.Category"
## [17] "Product.Name"  "Sales"         "Quantity"      "Discount"
## [21] "Profit"
```

```r
colnames(sales) [1] <- "Row.ID"
colnames(sales)
```

```
##  [1] "Row.ID"        "Order.ID"      "Order.Date"    "Ship.Date"
##  [5] "Ship.Mode"     "Customer.ID"   "Customer.Name" "Segment"
##  [9] "Country"       "City"          "State"         "Postal.Code"
## [13] "Region"        "Product.ID"    "Category"      "Sub.Category"
## [17] "Product.Name"  "Sales"         "Quantity"      "Discount"
## [21] "Profit"
```

Exercise 3

Convert both Ship.Date and Order.Date to date vectors within the sales data frame. What is the number of days between the most recent order and the oldest order? How many years is that? How many weeks?

Note: Use lubridate

```r
# Your code here

sales$Order.Date <- as.Date(sales$Order.Date
                            ,format='%m/%d/%Y')
sales$Ship.Date <- as.Date(sales$Ship.Date
                          ,format = '%B %d %Y',
                          optional = FALSE)
oldest_ord <- min(sales$Order.Date)
recent_ord <- max(sales$Order.Date)
num_orddays <- recent_ord - oldest_ord
num_orddays <- difftime(recent_ord, oldest_ord, units = "days")

num_ordwks <- difftime(recent_ord, oldest_ord, units = "weeks")

num_ordyrs <- as.duration(num_orddays)

print(num_orddays)
```

```
## Time difference of 1457 days
```

```r
print(num_ordwks)
```

```
## Time difference of 208.1429 weeks
```

```r
print(num_ordyrs)
```

```
## [1] "125884800s (~3.99 years)"
```

Exercise 4

What is the average number of days it takes to ship an order?

```
# Your code here
avg_ship <- mean(difftime(sales$Ship.Date, sales$Order.Date
                          , units = "days"))
print(avg_ship)
```

```
## Time difference of 3.908482 days
```

Exercise 5

How many customers have the first name Bill? You will need to split the customer name into first and last name segments and then use a regular expression to match the first name bill. Use the length() function to determine the number of customers with the first name Bill in the sales data.

```
# Your code here

cust_first_last <- str_split_fixed(string = sales$Customer.Name,
                                   pattern = " ", n = 2)

bill_only <- str_subset(string = cust_first_last,
                        pattern = "Bill", negate = FALSE)
length(bill_only)
```

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

Exercise 6

How many mentions of the word 'table' are there in the Product.Name column? **Note you can do this in one line of code**

```
# Your code here

length(str_subset(sales$Product.Name,
                  pattern = "Tables",
                  negate = FALSE))
```

```
## [1] 151
```

Exercise 7

Create a table of counts for each state in the sales data. The counts table should be ordered alphabetically from A to Z.

```r
# Your code here

sales$State <- factor(sales$State)
sales_state_table <- table(sales$State)
print(sales_state_table)
```

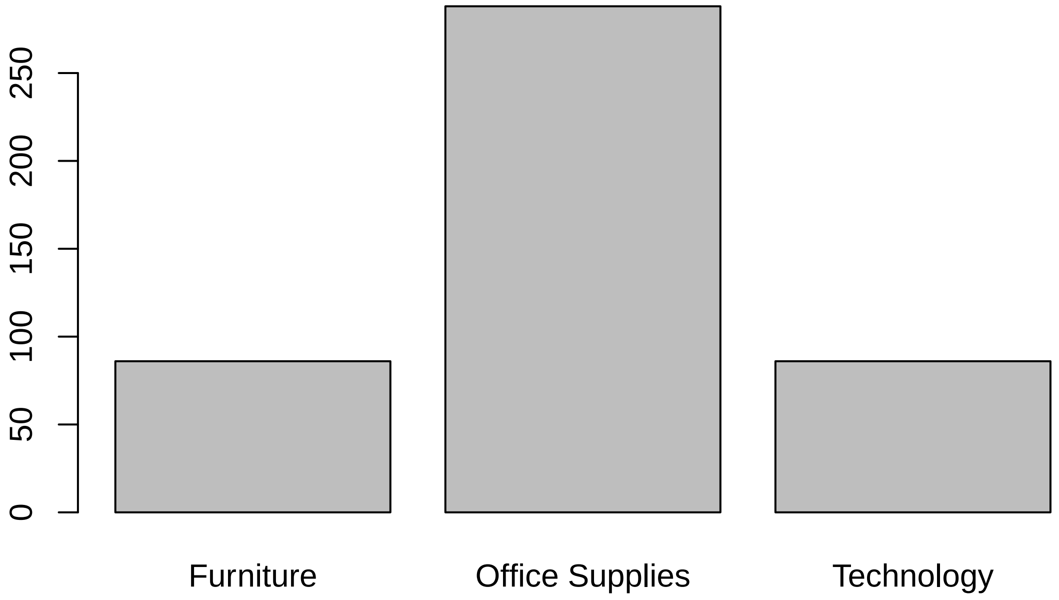
```
## 
##              Alabama              Arizona             Arkansas 
##                   28                  119                   22 
##           California             Colorado          Connecticut 
##                  993                   90                   50 
##             Delaware District of Columbia              Florida 
##                   47                    1                  186 
##              Georgia                Idaho             Illinois 
##                   79                    9                  286 
##              Indiana                 Iowa               Kansas 
##                   74                   11                   16 
##             Kentucky            Louisiana                Maine 
##                   64                   18                    4 
##             Maryland        Massachusetts             Michigan 
##                   63                   71                  142 
##            Minnesota          Mississippi             Missouri 
##                   41                   27                   37 
##              Montana             Nebraska               Nevada 
##                    2                   26                   24 
##        New Hampshire           New Jersey           New Mexico 
##                    9                   58                   11 
##             New York       North Carolina         North Dakota 
##                  555                  117                    7 
##                 Ohio             Oklahoma               Oregon 
##                  211                   38                   56 
##         Pennsylvania         Rhode Island       South Carolina 
##                  312                   25                   28 
##         South Dakota            Tennessee                Texas 
##                    9                   88                  460 
##                 Utah              Vermont             Virginia 
##                   27                   10                   80 
##           Washington        West Virginia            Wisconsin 
##                  254                    4                   38 
##              Wyoming 
##                    1
```

Exercise 8

Create an alphabetically ordered barplot for each sales Category in the State of Texas.

```r
# Your code here

sales_tx_df = sales[sales$State == "Texas", ]
barplot(table(sales_tx_df$Category))
```



Exercise 9

Find the average profit by region. **Note: You will need to use the aggregate() function to do this. To understand how the function works type ?aggregate in the console.**

```r
# Your code here

prof_reg <- aggregate(x = sales$Profit, by = list(sales$Region), FUN = "mean")
setNames(prof_reg, c("Region", "Avg.Profit"))
```

```
##    Region Avg.Profit
## 1 Central   20.46822
## 2    East   29.91937
## 3   South   11.27720
## 4    West   32.77000
```

Exercise 10

Find the average profit by order year. **Note: You will need to use the aggregate() function to do this. To understand how the function works type ?aggregate in the console.**

```r
# Your code here
ord_yr <- str_split_fixed(string = sales$Order.Date,
                          pattern = "-", n=3)
sales$Order.Year <- ord_yr[ , 1]

prof_yr <- aggregate(x = sales$Profit, by = list(sales$Order.Year), FUN = "mean")
setNames(prof_yr, c("Year", "Avg.Profit"))
```

```
##   Year Avg.Profit
## 1 2014   32.24582
## 2 2015   21.58676
## 3 2016   30.10960
## 4 2017   21.31825
```