Week 2 Exercises

Annelise Lobo

July 6, 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.

```
# Your code here

#first call the libraries needed for the following code.
library(stringr)
library(lubridate)
```

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

## The following object is masked from 'package:base':
##
##     date
```

```
library(forcats)

#Using read.delim() will read the data based on the selected seperator,
#which in this case is a pipe symbol.
#readr::guess_encoding(file.choose()) will help determine the
#fileEncoding which was done before writing it into the code below

sales_pipe = read.delim("/Users/axtonpulliam/Documents/DSE5002/Week_2/sales_pipe.txt",
                        fileEncoding = "ISO-8859-1",
                        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.

```
# Your code here

# using the colnames() on the data in question and
#using box brackets will parse the data in the column number specified and
#change it to thhe desired value

colnames(sales_pipe) [1] = "Row.ID"
```

Exercise 3

Convert both Order.ID 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

#for Order.Date to be converted to date vector
#find the structure of item, find if it is a date and then convert it
#based on the format (month, day, year in this case)
str(sales_pipe$Order.Date)
```

```
##  chr [1:4928] "11/8/2016" "11/8/2016" "6/12/2016" "10/11/2015" ...
```

```r
inherits(sales_pipe$Order.Date,  c("Date"))
```

```
## [1] FALSE
```

```r
sales_pipe$Order.Date = as.Date(sales_pipe$Order.Date, format = '%m/%d/%Y')

#to convert ship.date to date vectors
#find the structure of the item, find if it is a date and then
#convert it based on the format (month as a string, day, year in this case)
str(sales_pipe$Ship.Date)
```

```
##  chr [1:4928] "November 11 2016" "November 11 2016" "June 16 2016" ...
```

```r
inherits(sales_pipe$Ship.Date, c("Date"))
```

```
## [1] FALSE
```

```r
sales_pipe$Ship.Date = as.Date(sales_pipe$Ship.Date, format = '%B %d %Y')

#days between most recent order
#find the oldest (minimum date) and the newest date (maximum value)
oldest = min(sales_pipe$Order.Date)
recent = max(sales_pipe$Order.Date)

#find number of days b/w oldest and newest order using difftime.
#Set the units to the desired output of time
day_diff = difftime(recent, oldest, units = "days")
day_diff
```

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

```r
cat("Time difference of", time_length(day_diff, "year"), "years\n")
```

```
## Time difference of 3.991781 years
```

```r
difftime(recent, oldest, units = "weeks")
```

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

Exercise 4

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

```r
# Your code here

#find the average of the time difference between ship and order date
mean(difftime(sales_pipe$Ship.Date, sales_pipe$Order.Date, units = "days"))
```

```
## 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.

```r
# Your code here

#Create two new columns for the first and last names based off the customer.name column
#Split the customer.name column into two segments, seperated by a space.
#Paste the first column of customer.name into a new column as the first names
#and the same with the second column for the last names.
temp_char=stringr::str_split_fixed(string = sales_pipe$Customer.Name, pattern = " ", n = 2)
sales_pipe$Customer.First.Name = paste(temp_char[,1], sep=" ")
sales_pipe$Customer.Last.Name = paste(temp_char[,2], sep=" ")

#find all the customers with first name Bill
#Count how many there are with the length option
length(which(sales_pipe$Customer.First.Name=='Bill'))
```

```
## [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**

```r
# Your code here
#Use grep to pattern match the  word "table".
#Use the length operation to find how many occurances of the word there are
length(grep("table", sales_pipe$Product.Name))
```

```
## [1] 197
```

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.

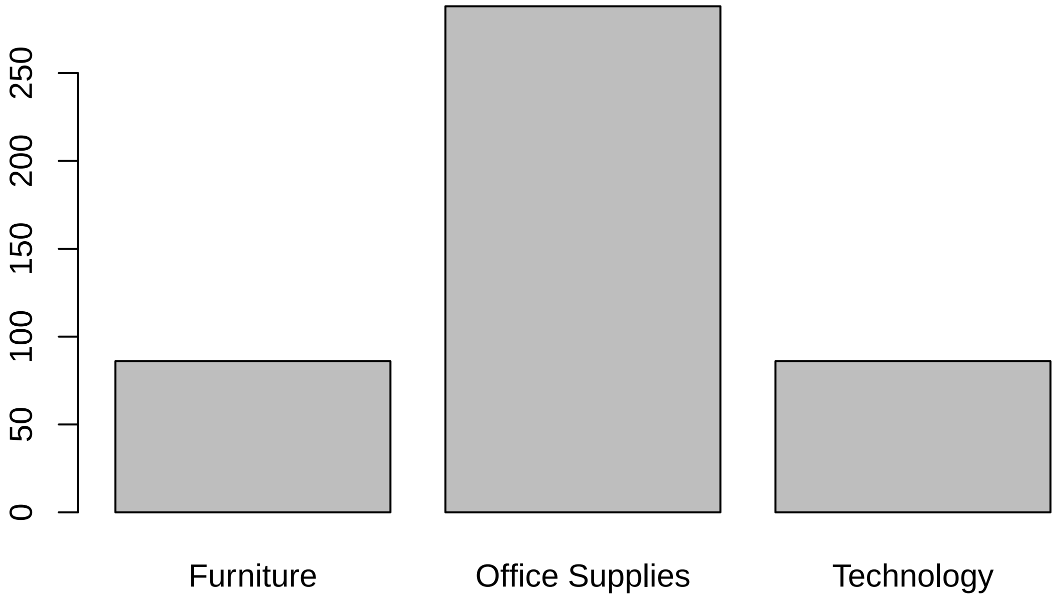
```r
# Your code here
#use the table operation to make a table of the states column.
#This will automatically be made in alphabetical order
table(sales_pipe$State)
```

```
## 
##              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.

```
# Your code here
#use the barplot operation to make a chart of the category sales
#Texas state is specified  by the box brackets
#This automatically plots in alphabetical order
barplot(table(sales_pipe$Category[sales_pipe$State  ==  "Texas"]))
```



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.**

```
# Your code here
#Use the aggregate function to provide a summary statistic of the mean of the column.
aggregate(sales_pipe$Profit, list(sales_pipe$Region), mean)
```

```
##   Group.1        x
## 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.**

```
# Your code here

#split the order date and create a new column with just the order year.
#Split the Order.Date column into three segments, seperated by a "-".
#Paste the first segment of the split column into a new column named Order.Year
temp_char  = stringr::str_split_fixed(string = sales_pipe$Order.Date, pattern = "-", n =3)
sales_pipe$Order.Year = paste(temp_char[,1], sep="-")

#Find the mean of the profits seperated by order year
aggregate(sales_pipe$Profit, list(sales_pipe$Order.Year), mean)
```

```
##   Group.1        x
## 1    2014 32.24582
## 2    2015 21.58676
## 3    2016 30.10960
## 4    2017 21.31825
```