Week 2 Exercises

PJ Grant

Nov 3rd, 2024

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)

library(stringr)  
library(lubridate)

##   
## Attaching package: 'lubridate'

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

library(forcats)  
library(readr)

# Exercise 1

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

sales <- read\_delim("~/MM/DSE5002/Week\_2/Data/sales\_pipe.txt",   
 delim = "|",   
 escape\_double = FALSE,   
 trim\_ws = TRUE,   
 locale = locale(encoding = "latin1"))

## Warning: One or more parsing issues, call `problems()` on your data frame for details,  
## e.g.:  
## dat <- vroom(...)  
## problems(dat)

## Rows: 4928 Columns: 20  
## ── Column specification ────────────────────────────────────────────────────────  
## Delimiter: "|"  
## chr (16): Ship.Date, Ship.Mode, Customer.ID, Customer.Name, Segment, Country...  
## dbl (4): Order.ID, Order.Date, Product.ID, Discount  
##   
## ℹ Use `spec()` to retrieve the full column specification for this data.  
## ℹ Specify the column types or set `show\_col\_types = FALSE` to quiet this message.

##, fileEncoding = 'WINDOWS-1252'  
  
# Splitting the values in the profit column into three distinct columns  
temp\_char <- str\_split\_fixed(string=sales$Profit,pattern='\\|', n=3)  
  
#Assign the correct column headers  
colnames(temp\_char) <- c("Quantity1", "Discount1", "Profit1")  
  
#Combining the two data frames  
updated\_sales <- cbind(sales, temp\_char)   
  
# Super setting the correct columns  
updated\_sales <- updated\_sales[,2:23]  
  
#eliminating the incorrect profit column  
updated\_sales <- updated\_sales[,-19]  
  
#Renaming the columns  
new\_column\_names <- c("row\_id", "order\_id", "order\_date", "ship\_date", "ship\_mode", "customer\_id", "customer\_name", "segment", "country", "city", "state", "postal\_code", "region", "product\_id", "category", "sub\_category", "product\_name", "sales", "quantity", "discount", "profit")  
  
colnames(updated\_sales) <- new\_column\_names  
  
#updating from strings / characters to numbers  
updated\_sales$quantity = as.numeric(updated\_sales$quantity)  
updated\_sales$discount = as.numeric(updated\_sales$discount)  
updated\_sales$profit = as.numeric(updated\_sales$profit)  
  
#convert back to sales  
sales = updated\_sales  
View(sales)

# 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.**

# Included in the column name update above

# 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**

#Set as date  
sales$ship\_date <- as.Date(sales$ship\_date, format = '%B %d %Y')  
sales$order\_date <- as.Date(sales$order\_date, format = '%m/%d/%Y')  
  
#Find max and min order dates  
newest\_order = max(sales$order\_date) #The most recent order date is 2017-12-30  
oldest\_order = min(sales$order\_date) #The older order was placed on 2014-01-03  
  
#Create interval between oldest and most recent order  
date\_intervals = interval(newest\_order, oldest\_order)  
  
time\_in\_seconds = int\_length(date\_intervals) \* -1  
time\_in\_seconds

## [1] 125884800

#Convert interval from seconds to days // 1457 days occurred between the oldest and most recent order  
number\_of\_days = time\_in\_seconds / 60 / 60 / 24  
number\_of\_days

## [1] 1457

# ~ 4 years  
number\_of\_days / 365

## [1] 3.991781

# ~208 weeks  
number\_of\_days / 7

## [1] 208.1429

# Exercise 4

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

#Calculate the time to ship for each order  
sales$time\_to\_ship <- sales$ship\_date - sales$order\_date  
  
# The average time to ship for each order is 3.9 days  
mean(sales$time\_to\_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.

#Split names and create new columns with first and last name  
sales$name <- str\_split\_fixed(sales$customer\_name," ", n=2)  
sales$first\_name <- sales$name[,1]  
sales$last\_name<- sales$name[,2]  
  
# There are 37 people names bill  
length(str\_subset(sales$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**

#240 mentions of the word table in the product\_name column  
sum(str\_count(sales$product\_name,"table"))

## [1] 240

# 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.

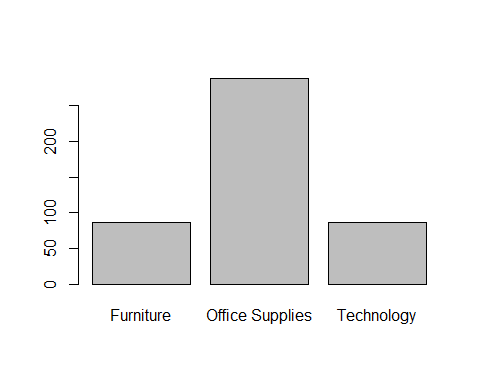
table(sales$state)

##   
## 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.

texas\_count <- sales[sales$state == "Texas",]  
barplot(table(texas\_count$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.**

#The average profit by region is as follows: Central(20.5), East(29.9), South(11.3), West(32.8)   
aggregate(sales$profit, list(sales$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.**

#Determine which year each order was placed  
sales$order\_year <- year(sales$order\_date)  
  
#Determine the average profit by year  
round(aggregate(sales$profit, list(sales$order\_year), mean),1)

## Group.1 x  
## 1 2014 32.2  
## 2 2015 21.6  
## 3 2016 30.1  
## 4 2017 21.3

#The average profit for 2014 was 32.2, 2015 was 21.6, 2016 was 30.1, 2017 was 21.3