DS 614 Homework 2

DS 413/613 HOMEWORK 2 Instructions: For this homework assignment, produce an Rmarkdown file and a Word file as you have done for previous labs and the first homework assignment. For homework 2, however push your Word file to Github using the required commands from the Bash terminal. Create a new repository. Use your first name and your last name as a name of your new repository. When your push is complete and your Word file is registered in the new repository, email me the link (in the address bar at the top of the page) for your new repository.

Homework 2 problems 1) Enter and run USArrests. What type of information is shown in the data table USArrests? (Three or four sentences)

#USArrests

The dataset contains arrests per 100,000 residents for different crimes in each state. The arrests shown are for assault, murder, and rape in 1973. It also gives the percent of population in each state living in urban areas.

1. Use and show R coding that features a map function to show maximum values for all variables of the USArerests data frame. Which State has the largest number of Assaults according to the USAressts data frame ?

library(tidyverse)

## -- Attaching packages ------------------------------------------------------- tidyverse 1.3.0 --

## v ggplot2 3.3.2 v purrr 0.3.4  
## v tibble 3.0.3 v dplyr 1.0.2  
## v tidyr 1.1.2 v stringr 1.4.0  
## v readr 1.3.1 v forcats 0.5.0

## Warning: package 'forcats' was built under R version 4.0.3

## -- Conflicts ---------------------------------------------------------- tidyverse\_conflicts() --  
## x dplyr::filter() masks stats::filter()  
## x dplyr::lag() masks stats::lag()

map(USArrests, max)

## $Murder  
## [1] 17.4  
##   
## $Assault  
## [1] 337  
##   
## $UrbanPop  
## [1] 91  
##   
## $Rape  
## [1] 46

North Carolina had the most assaults

1. Install the nycflights13 package: install.packages(“nycflights13”) , call the following library: library(nycflights13), and then enter flights (this will produce the flights data table)

#install.packages("nycflights13")  
library(nycflights13)

## Warning: package 'nycflights13' was built under R version 4.0.3

flights

## # A tibble: 336,776 x 19  
## year month day dep\_time sched\_dep\_time dep\_delay arr\_time sched\_arr\_time  
## <int> <int> <int> <int> <int> <dbl> <int> <int>  
## 1 2013 1 1 517 515 2 830 819  
## 2 2013 1 1 533 529 4 850 830  
## 3 2013 1 1 542 540 2 923 850  
## 4 2013 1 1 544 545 -1 1004 1022  
## 5 2013 1 1 554 600 -6 812 837  
## 6 2013 1 1 554 558 -4 740 728  
## 7 2013 1 1 555 600 -5 913 854  
## 8 2013 1 1 557 600 -3 709 723  
## 9 2013 1 1 557 600 -3 838 846  
## 10 2013 1 1 558 600 -2 753 745  
## # ... with 336,766 more rows, and 11 more variables: arr\_delay <dbl>,  
## # carrier <chr>, flight <int>, tailnum <chr>, origin <chr>, dest <chr>,  
## # air\_time <dbl>, distance <dbl>, hour <dbl>, minute <dbl>, time\_hour <dttm>

1. Use and show R code that will indicate how many rows and how many columns the flights data has. Review and revisit your notes from STAT 412/612. Describe a tibble (two or three sentences). Now use and show R code that verifies that flights is a tibble

ncol(flights)

## [1] 19

nrow(flights)

## [1] 336776

is\_tibble(flights)

## [1] TRUE

#A tibble is a dataframe that \_\_\_\_\_\_\_\_\_\_\_\_\_

1. Now Use and show R code (featuring a map function) that will output the type of each column of the flights tibble.

map(flights, class)

## $year  
## [1] "integer"  
##   
## $month  
## [1] "integer"  
##   
## $day  
## [1] "integer"  
##   
## $dep\_time  
## [1] "integer"  
##   
## $sched\_dep\_time  
## [1] "integer"  
##   
## $dep\_delay  
## [1] "numeric"  
##   
## $arr\_time  
## [1] "integer"  
##   
## $sched\_arr\_time  
## [1] "integer"  
##   
## $arr\_delay  
## [1] "numeric"  
##   
## $carrier  
## [1] "character"  
##   
## $flight  
## [1] "integer"  
##   
## $tailnum  
## [1] "character"  
##   
## $origin  
## [1] "character"  
##   
## $dest  
## [1] "character"  
##   
## $air\_time  
## [1] "numeric"  
##   
## $distance  
## [1] "numeric"  
##   
## $hour  
## [1] "numeric"  
##   
## $minute  
## [1] "numeric"  
##   
## $time\_hour  
## [1] "POSIXct" "POSIXt"

1. Use and show R coding that features usage of a map function to find the slope and the intercept of models for the different levels of the Species variable of the iris data frame. For each model, Sepal.Width predicts Sepal.Length.

factor(iris$Species)

## [1] setosa setosa setosa setosa setosa setosa   
## [7] setosa setosa setosa setosa setosa setosa   
## [13] setosa setosa setosa setosa setosa setosa   
## [19] setosa setosa setosa setosa setosa setosa   
## [25] setosa setosa setosa setosa setosa setosa   
## [31] setosa setosa setosa setosa setosa setosa   
## [37] setosa setosa setosa setosa setosa setosa   
## [43] setosa setosa setosa setosa setosa setosa   
## [49] setosa setosa versicolor versicolor versicolor versicolor  
## [55] versicolor versicolor versicolor versicolor versicolor versicolor  
## [61] versicolor versicolor versicolor versicolor versicolor versicolor  
## [67] versicolor versicolor versicolor versicolor versicolor versicolor  
## [73] versicolor versicolor versicolor versicolor versicolor versicolor  
## [79] versicolor versicolor versicolor versicolor versicolor versicolor  
## [85] versicolor versicolor versicolor versicolor versicolor versicolor  
## [91] versicolor versicolor versicolor versicolor versicolor versicolor  
## [97] versicolor versicolor versicolor versicolor virginica virginica   
## [103] virginica virginica virginica virginica virginica virginica   
## [109] virginica virginica virginica virginica virginica virginica   
## [115] virginica virginica virginica virginica virginica virginica   
## [121] virginica virginica virginica virginica virginica virginica   
## [127] virginica virginica virginica virginica virginica virginica   
## [133] virginica virginica virginica virginica virginica virginica   
## [139] virginica virginica virginica virginica virginica virginica   
## [145] virginica virginica virginica virginica virginica virginica   
## Levels: setosa versicolor virginica

model <- iris %>%  
 split(.$Species) %>%  
 map(~lm(Sepal.Length ~ Sepal.Width, data = .))  
  
model

## $setosa  
##   
## Call:  
## lm(formula = Sepal.Length ~ Sepal.Width, data = .)  
##   
## Coefficients:  
## (Intercept) Sepal.Width   
## 2.6390 0.6905   
##   
##   
## $versicolor  
##   
## Call:  
## lm(formula = Sepal.Length ~ Sepal.Width, data = .)  
##   
## Coefficients:  
## (Intercept) Sepal.Width   
## 3.5397 0.8651   
##   
##   
## $virginica  
##   
## Call:  
## lm(formula = Sepal.Length ~ Sepal.Width, data = .)  
##   
## Coefficients:  
## (Intercept) Sepal.Width   
## 3.9068 0.9015

V <- list(12, 22, 27, 31.5, NA, 39, "east")  
V

## [[1]]  
## [1] 12  
##   
## [[2]]  
## [1] 22  
##   
## [[3]]  
## [1] 27  
##   
## [[4]]  
## [1] 31.5  
##   
## [[5]]  
## [1] NA  
##   
## [[6]]  
## [1] 39  
##   
## [[7]]  
## [1] "east"

For the given list above; 7a) Use and show R code to find the length of the list

length(V)

## [1] 7

7b) Use and show R code that will extract the missing value

V[is.na(V)]

## [[1]]  
## [1] NA

7c) Use and show R code that will extract the third object

V[3]

## [[1]]  
## [1] 27

7d) Use and show R code that will extract the character string and the minimum number.

V[c(1,7)]

## [[1]]  
## [1] 12  
##   
## [[2]]  
## [1] "east"

7e) Use one line of code to show that the seventh object is a character object.

map(V[7], class)

## [[1]]  
## [1] "character"