Homework 2

Riyo Moore

2/16/2021

library(tidyverse)

## ── Attaching packages ─────────────────────────────────────── tidyverse 1.3.0 ──

## ✓ ggplot2 3.3.3 ✓ purrr 0.3.4  
## ✓ tibble 3.0.5 ✓ dplyr 1.0.3  
## ✓ tidyr 1.1.2 ✓ stringr 1.4.0  
## ✓ readr 1.4.0 ✓ forcats 0.5.0

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

library(dplyr)

Problem 1

data("USArrests")

This data table contains the arrests per 100,000 residents for assault, murder, and rape in the United States. Also included, is the percent of the population that lives in urban areas.

Problem 2

pmap\_dbl(USArrests, max)

## [1] 236 263 294 190 276 204 110 238 335 211 83 120 249 113 57 115 109 249 83  
## [20] 300 149 255 72 259 178 109 102 252 57 159 285 254 337 45 120 151 159 106  
## [39] 174 279 86 188 201 120 48 156 145 81 66 161

pmap\_dbl(list(USArrests$Assault), max)

## [1] 236 263 294 190 276 204 110 238 335 211 46 120 249 113 56 115 109 249 83  
## [20] 300 149 255 72 259 178 109 102 252 57 159 285 254 337 45 120 151 159 106  
## [39] 174 279 86 188 201 120 48 156 145 81 53 161

max(USArrests$Assault)

## [1] 337

It appears that the state with the largest number of Assaults is North Carolina with 337 arrests

Problem 3

library(nycflights13)  
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>

Problem 4

nrow(flights)

## [1] 336776

ncol(flights)

## [1] 19

class(flights)

## [1] "tbl\_df" "tbl" "data.frame"

tibble(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>

Tibbles are data frames that have been made easier to look at by displaying the columns and only 10 rows. It also displays the type of the columns that are present, whether that be integer or character, and the number of other rows and columns that are not present in the tibble.

Problem 5

flights %>% map(typeof)

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

Problem 6

data("iris")  
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

model1 <- iris %>%  
 split(.$Species) %>%  
 map(~lm(Sepal.Length ~ Sepal.Width, data = .))  
model1

## $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

Problem 7

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"

length(V)

## [1] 7

V[!is.na(V)]

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

V[3]

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

V[c(2,3,4,5,6)]

## [[1]]  
## [1] 22  
##   
## [[2]]  
## [1] 27  
##   
## [[3]]  
## [1] 31.5  
##   
## [[4]]  
## [1] NA  
##   
## [[5]]  
## [1] 39

str(V[7])

## List of 1  
## $ : chr "east"