HW2

Chao Chen Yu

2/14/2021

1. It's a data frame. There are four variables. Murder and Rape are doubles. Assault and UrbanPop are integers. And the data set have 50 observations.

```
data(USArrests)
str(USArrests)
```

```
## 'data.frame':    50 obs. of  4 variables:
##  $ Murder  : num  13.2 10 8.1 8.8 9 7.9 3.3 5.9 15.4 17.4 ...
##  $ Assault : int  236 263 294 190 276 204 110 238 335 211 ...
##  $ UrbanPop: int  58 48 80 50 91 78 77 72 80 60 ...
##  $ Rape    : num  21.2 44.5 31 19.5 40.6 38.7 11.1 15.8 31.9 25.8 ...
```

2.

```
USArrests %>%
  map_dbl(max)
```

```
##   Murder  Assault UrbanPop     Rape
##     17.4    337.0     91.0     46.0
```

```
row.names(USArrests)[map_dbl(USArrests,which.max)]
```

```
## [1] "Georgia"        "North Carolina" "California"     "Nevada"
```

```
cat("North Carolina has the largest number of Assaults")
```

```
## North Carolina has the largest number of Assaults
```

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>
```

4.

```
str(flights) #tibble [336,776 × 19]
```

```
## tibble [336,776 x 19] (S3: tbl_df/tbl/data.frame)
##  $ year          : int [1:336776] 2013 2013 2013 2013 2013 2013 2013 2013 2013 2013 ...
##  $ month         : int [1:336776] 1 1 1 1 1 1 1 1 1 1 ...
##  $ day           : int [1:336776] 1 1 1 1 1 1 1 1 1 1 ...
##  $ dep_time      : int [1:336776] 517 533 542 544 554 554 555 557 557 558 ...
##  $ sched_dep_time: int [1:336776] 515 529 540 545 600 558 600 600 600 600 ...
##  $ dep_delay     : num [1:336776] 2 4 2 -1 -6 -4 -5 -3 -3 -2 ...
##  $ arr_time      : int [1:336776] 830 850 923 1004 812 740 913 709 838 753 ...
##  $ sched_arr_time: int [1:336776] 819 830 850 1022 837 728 854 723 846 745 ...
##  $ arr_delay     : num [1:336776] 11 20 33 -18 -25 12 19 -14 -8 8 ...
##  $ carrier       : chr [1:336776] "UA" "UA" "AA" "B6" ...
##  $ flight        : int [1:336776] 1545 1714 1141 725 461 1696 507 5708 79 301 ...
##  $ tailnum       : chr [1:336776] "N14228" "N24211" "N619AA" "N804JB" ...
##  $ origin        : chr [1:336776] "EWR" "LGA" "JFK" "JFK" ...
##  $ dest          : chr [1:336776] "IAH" "IAH" "MIA" "BQN" ...
##  $ air_time      : num [1:336776] 227 227 160 183 116 150 158 53 140 138 ...
##  $ distance      : num [1:336776] 1400 1416 1089 1576 762 ...
##  $ hour          : num [1:336776] 5 5 5 5 6 5 6 6 6 6 ...
##  $ minute        : num [1:336776] 15 29 40 45 0 58 0 0 0 0 ...
##  $ time_hour     : POSIXct[1:336776], format: "2013-01-01 05:00:00" "2013-01-01 05:00:00" ...
```

```
typeof(flights)
```

```
## [1] "list"
```

```
class(flights) #  It's a tibble.
```

```
## [1] "tbl_df"     "tbl"        "data.frame"
```

The flights contains 336,766 rows and 19 columns. It provides the actual departure and arrival times and scheduled departure and arrival times. Comparing to data.frame, tibble is more easily to browse the data set.

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"
```

6.

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

models
```

```
## $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
```

7.

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

length(V) #7a
```

```
## [1] 7
```

```r
V[is.na(V)] #7b
```

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

```r
V[[3]] #7c
```

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

```r
V[c(which.min(V[map(V,class)=="numeric"]),which(map(V,class)=="character"))] #7d
```

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

```r
class(V[[7]]) #7e
```

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