

Tidyverse lab

Ariane Ducellier

10/3/2023

Libraries

Load the necessary libraries.

```
library(nycflights13)
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ----- tidyverse 2.0.0 --
## v dplyr      1.1.3      v readr      2.1.4
## v forcats    1.0.0      v stringr   1.5.0
## v ggplot2    3.4.3      v tibble    3.2.1
## v lubridate  1.9.3      v tidyr     1.3.0
## v purrr      1.0.2
## -- Conflicts ----- tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()     masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

The first three exercices are done with the flights dataset from the nycflights13 package.

```
summary(flights)
```

```
##      year      month      day      dep_time      sched_dep_time
## Min.   :2013    Min.   : 1.000  Min.   : 1.00  Min.   :    1  Min.   : 106
## 1st Qu.:2013    1st Qu.: 4.000  1st Qu.: 8.00  1st Qu.: 907  1st Qu.: 906
## Median :2013    Median : 7.000  Median :16.00  Median :1401  Median :1359
## Mean   :2013    Mean   : 6.549  Mean   :15.71  Mean   :1349  Mean   :1344
## 3rd Qu.:2013    3rd Qu.:10.000  3rd Qu.:23.00  3rd Qu.:1744  3rd Qu.:1729
## Max.   :2013    Max.   :12.000  Max.   :31.00  Max.   :2400  Max.   :2359
##
##                                     NA's   :8255
##      dep_delay      arr_time      sched_arr_time      arr_delay
## Min.   : -43.00    Min.   :    1  Min.   :    1  Min.   : -86.000
## 1st Qu.:  -5.00    1st Qu.:1104  1st Qu.:1124  1st Qu.: -17.000
## Median :  -2.00    Median :1535  Median :1556  Median :  -5.000
## Mean   :   12.64    Mean   :1502  Mean   :1536  Mean   :    6.895
## 3rd Qu.:   11.00    3rd Qu.:1940  3rd Qu.:1945  3rd Qu.:   14.000
## Max.   :1301.00    Max.   :2400  Max.   :2359  Max.   :1272.000
##
##      NA's   :8255      NA's   :8713      NA's   :9430
##      carrier      flight      tailnum      origin
## Length:336776      Min.   :    1  Length:336776      Length:336776
## Class :character    1st Qu.: 553  Class :character    Class :character
## Mode  :character    Median :1496  Mode  :character    Mode  :character
##
##                                     Mean   :1972
##                                     3rd Qu.:3465
```

```
##           Max.      :8500
##
##      dest          air_time      distance      hour
## Length:336776      Min.      : 20.0      Min.      : 17      Min.      : 1.00
## Class :character    1st Qu.: 82.0      1st Qu.: 502      1st Qu.: 9.00
## Mode  :character    Median :129.0      Median : 872      Median :13.00
##                      Mean   :150.7      Mean   :1040      Mean   :13.18
##                      3rd Qu.:192.0      3rd Qu.:1389      3rd Qu.:17.00
##                      Max.    :695.0      Max.    :4983      Max.    :23.00
##                      NA's    :9430
##      minute      time_hour
## Min.      : 0.00      Min.      :2013-01-01 05:00:00.00
## 1st Qu.: 8.00      1st Qu.:2013-04-04 13:00:00.00
## Median :29.00      Median :2013-07-03 10:00:00.00
## Mean   :26.23      Mean   :2013-07-03 05:22:54.64
## 3rd Qu.:44.00      3rd Qu.:2013-10-01 07:00:00.00
## Max.    :59.00      Max.    :2013-12-31 23:00:00.00
##
```

Exercise 1

Question 1

In a single pipeline for each condition, find all flights that meet the condition: - Had an arrival delay of two or more hours. - Flew to Houston (IAH or HOU). - Were operated by United, American, or Delta. - Departed in summer (July, August, and September). - Arrived more than two hours late, but didn't leave late. - Were delayed by at least an hour, but made up more than 30 minutes in flight.

Question 2

Sort flights to find the flights with the longest departure delays. Find the flights that left earliest in the morning.

Question 3

Sort flights to find the fastest flights.

Question 4

Was there a flight on every day of 2013?

Question 5

Which flights traveled the farthest distance? Which traveled the least distance?

Question 6

Does it matter what order you used `filter()` and `arrange()` if you're using both? Why/why not? Think about the results and how much work the functions would have to do.

Exercise 2

Question 1

Compare `dep_time`, `sched_dep_time`, and `dep_delay`. How would you expect those three numbers to be related?

Question 2

Rename `air_time` to `air_time_min` to indicate units of measurement and move it to the beginning of the data frame.

Exercise 3

Question 1

Which carrier has the worst average delays? Challenge: Can you disentangle the effects of bad airports versus bad carriers? Why/why not?

Question 2

Find the flights that are most delayed upon departure from each destination.

Question 3

How do delays vary over the course of the day? Illustrate your answer with a plot.

Exercise 4

This exercise will make use of tidyverse functions for data transformation to extract and manipulate metadata of seismic stations in the Northern California Seismic Network.

We want to download seismic waveforms from a seismic data archive of specific earthquakes. We are not sure what seismic sensors (stations) are operating at that time. The list of stations available in the seismic networks has more than 6000, that's way too many! So we want to filter only the seismic stations that are relevant for the research.

This is the address of the website to download the data: NCEDC metadata

Question 1

First, you need to load the data into a tibble. You may use the following header:

```
header = c("Station", "Network", "Channel", "Location", "Rate",  
           "Start_time", "End_time", "Latitude", "Longitude",  
           "Elevation", "Depth", "Dip", "Azimuth", "Instrument")
```

Question 2

Now, we need to convert `Start_time` and `End_time` into a datetime format.

It turns out than only the following channels are relevant for the work we want to do:

- BHE, BHN, BHZ, BH1, BH2,
- EHE, EHN, EHZ, EH1, EH2,
- HHE, HHN, HHZ, HH1, HH2,
- SHE, SHN, SHZ, SH1, SH2.

That is, we want the channels that start with B, E, H or S and which second letter with an H.

Question 3

Filter the dataset to keep only the rows with the channels as defined above.

The seismic data archive that we are working on starts on 2007/07/01 and ends on 2009/07/01. We are only interested in stations that started recording before 2009/07/01 and ended recording after 2007/07/01.

Question 4

Filter the dataset to keep only stations that started recording before 2009/07/01 and ended recording after 2007/07/01.

The earthquakes we are interested in are located at latitude = 40.09N and longitude = -122.87E.

We want to keep the stations that are located less than 100 km from the earthquakes.

Question 5

Filter the dataset to keep only stations that are within 100 km from the earthquakes.

You may use this function to compute the distance from the station to the earthquakes using the latitude and the longitude:

```
get_dists <- function(df){  
  lat0 = 40.09000  
  lon0 = -122.87000  
  a = 6378.136  
  e = 0.006694470  
  dx = (pi / 180.0) * a * cos(lat0 * pi / 180.0) /  
    sqrt(1.0 - e * e * sin(lat0 * pi / 180.0) * sin(lat0 * pi / 180.0))  
  dy = (3.6 * pi / 648.0) * a * (1.0 - e * e) /  
    ((1.0 - e * e * sin(lat0 * pi / 180.0) * sin(lat0 * pi / 180.0)) ** 1.5)  
  df$x = dx * (df$Longitude - lon0)  
  df$y = dy * (df$Latitude - lat0)  
  df$Distance = sqrt((df$x, 2.0) + (df$y, 2.0), 1/2)  
  return (df$Distance)  
}
```

For the final step, we only want to keep the columns: Station, Network, Channel, Location, Latitude, Longitude, Elevation, Depth, Start_time, End_time. We want to group the stations:

Question 6

First, group the stations by Station, Network, Channel, Location, Latitude, Longitude, Elevation, Depth, and compute the minimum start time and the maximum end time.

Question 7

Second, group the stations by Station, Network, Location, Latitude, Longitude, Elevation, Depth, and concatenate all the channels for a given station in a single string, separated by a comma.