# Course 2 Section 1.7 - Relational data

Jiaying Wu

```
library(tidyverse)
```

### Step 1: Open RStudio and install nycflights13

If you haven't already, open RStudio on your computer, and then install and load the package nycflights13.

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

### Step 2: Determine the amount of variables and observations

Then, look up the help file for the flights data.

- How many variables and observations are there?
- What do the columns mean?

```
glimpse(flights)
```

```
## Rows: 336,776
## Columns: 19
## $ year           <int> 2013, 2013, 2013, 2013, 2013, 2013, 2013, 2013, 2013...
## $ month          <int> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1...
## $ day            <int> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1...
## $ dep_time       <int> 517, 533, 542, 544, 554, 554, 555, 557, 557, 558, 55...
## $ sched_dep_time <int> 515, 529, 540, 545, 600, 558, 600, 600, 600, 600, 60...
## $ dep_delay      <dbl> 2, 4, 2, -1, -6, -4, -5, -3, -3, -2, -2, -2, -2, -2,...
## $ arr_time       <int> 830, 850, 923, 1004, 812, 740, 913, 709, 838, 753, 8...
## $ sched_arr_time <int> 819, 830, 850, 1022, 837, 728, 854, 723, 846, 745, 8...
## $ arr_delay      <dbl> 11, 20, 33, -18, -25, 12, 19, -14, -8, 8, -2, -3, 7,...
## $ carrier        <chr> "UA", "UA", "AA", "B6", "DL", "UA", "B6", "EV", "B6"...
## $ flight         <int> 1545, 1714, 1141, 725, 461, 1696, 507, 5708, 79, 301...
## $ tailnum        <chr> "N14228", "N24211", "N619AA", "N804JB", "N668DN", "N...
## $ origin         <chr> "EWR", "LGA", "JFK", "JFK", "LGA", "EWR", "EWR", "LG...
## $ dest           <chr> "IAH", "IAH", "MIA", "BQN", "ATL", "ORD", "FLL", "IA...
## $ air_time       <dbl> 227, 227, 160, 183, 116, 150, 158, 53, 140, 138, 149...
## $ distance       <dbl> 1400, 1416, 1089, 1576, 762, 719, 1065, 229, 944, 73...
## $ hour           <dbl> 5, 5, 5, 5, 6, 5, 6, 6, 6, 6, 6, 6, 6, 6, 6, 5, 6, 6...
## $ minute         <dbl> 15, 29, 40, 45, 0, 58, 0, 0, 0, 0, 0, 0, 0, 0, 0, 59...
## $ time_hour      <dttm> 2013-01-01 05:00:00, 2013-01-01 05:00:00, 2013-01-0...
```

```
?flights
```

- **year**, **month**, **day**: Date of departure.
- **dep_time**, **arr_time**: Actual departure and arrival times (format HHMM or HMM), local tz.
- **sched_dep_time**, **sched_arr_time**: Scheduled departure and arrival times (format HHMM or HMM), local tz.
- **dep_delay**, **arr_delay**: Departure and arrival delays, in minutes. Negative times represent early departures/arrivals.
- **carrier**: Two letter carrier abbreviation. See airlines to get name.
- **flight**: Flight number.
- **tailnum**: Plane tail number. See planes for additional metadata.
- **origin**, **dest**: Origin and destination. See airports for additional metadata.
- **air_time**: Amount of time spent in the air, in minutes.
- **distance**: Distance between airports, in miles.
- **hour**, **minute**: Time of scheduled departure broken into hour and minutes.
- **time_hour**: Scheduled date and hour of the flight as a POSIXct date.

### Step 3: Find out how many flights depart from JFK

Use the wrangling verbs filter(), count() and summarise() on the flights data to answer the following: What is the average number of flights that United Airlines (UA) flies out of JFK between 8.00 am and 9.00 am by day of the month?

```
flights %>%
  filter(origin == "JFK") %>%
  count()
```

```
## # A tibble: 1 x 1
##        n
##    <int>
## 1 111279
```

The total flights depart from JFK is 111279.

The average number of flights that United Airlines (UA) flies out of JFK between 8.00 am and 9.00 am by day of the month:

```
flights %>%
  filter(origin == "JFK",
         carrier == "UA",
         dep_time >= 800 & dep_time <= 900) %>%
  group_by(day) %>%
  count() %>%
  summarise(avg_flights = mean(n))
```

```
## # A tibble: 31 x 2
##      day avg_flights
##    <int>       <dbl>
##  1     1          18
##  2     2          19
##  3     3          18
##  4     4          17
##  5     5          18
##  6     6          18
##  7     7          19
##  8     8          21
##  9     9          18
## 10    10          19
## # ... with 21 more rows
```

**Step 4: Identify the plane responsible for delays**

Use the wrangling verbs, group_by(), summarise() and arrange() to find the plane that has the highest total arrival delay (minutes).

```
flights %>%
  group_by(tailnum) %>%
  summarise(sum_arr_delay = sum(arr_delay,na.rm = TRUE)) %>%
  arrange(desc(sum_arr_delay))
```

```
## # A tibble: 4,044 x 2
##    tailnum sum_arr_delay
##    <chr>           <dbl>
##  1 N15910           7317
##  2 N15980           7134
##  3 N16919           6904
##  4 N228JB           6778
##  5 N14998           6087
##  6 N192JB           5810
##  7 N292JB           5804
##  8 N12921           5788
##  9 N13958           5620
## 10 N10575           5566
## # ... with 4,034 more rows
```

The plane with the highest total arrival delay of 7317 minutes was caused by plane with tail number N15910.