STAT 385 - Homework 5

Jia Lin Mei - jmei43

Due 11:59 PM, 3/24/2020

Contents

Dataset: 2015 Flight Delays and Cancellations Data

In this homework, we take a quick look at the 2015 Flight Delays and Cancellations Data provided by the U.S. Department of Transportation. This is a huge dataset availalbe on Kaggle. But for us, we will only take a look at flights **flying out** from O'Hare International Airport (ORD) in January, 2015.

Load data

- I have filtered out the data specific to O'Hare and stored it in `ohare_jan.csv`. This filtered data is available at the URL: https://nkha149.github.io/stat385-sp2020/files/data/ohare_jan.csv.

```
library(tidyverse)
flights <- read_csv(file = "https://nkha149.github.io/stat385-sp2020/files/data/ohare_jan.csv")
```

- Write the code to print out the number of variables (columns) and the number of observations (rows) in this dataset.

```
dim(flights)
```

```
## [1] 23484    26
```

```
#There are 26 variables and 23484 observations in this dataset
```

- Use the `View()` function to take a look at the data. (Don't add any code here)

Review Basic Functions

First, let's review some basic R functions that we learned in the first half of the course.

Use R code to answer the following questions:

- How many different airlines fly out from O'Hare?

```
newairlines <- unique(flights$AIRLINE)
length(newairlines)
```

```
## [1] 12
```

- How many different airports is O'Hare connected to? (flights coming out of O'Hare go to)

```
newdest <- unique(flights$DESTINATION_AIRPORT)
length(newdest)
```

```
## [1] 154
```

- What is the average **departing delay** of flights departing O'Hare in Jan 2015?

```
mean(flights$DEPARTURE_DELAY, na.rm=TRUE)
```

```
## [1] 19.96205
```

- What is the five summary statistics of the **taxi out time** of flights departing O'Hare in Jan 2015?

```
newtaxiout <- na.omit(flights$TAXI_OUT)
summary(newtaxiout)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    1.00   13.00   16.00   19.87   21.00  152.00
```

`filter` function

Now, we will practice the skills we recently learned in the `dplyr` package.

- Print out only the flights that are going to U of I Willard Airport, `CMI`.
- Make sure to remove `eval = FALSE` after your write the code!

```
flights %>%
  filter(DESTINATION_AIRPORT == "CMI")
```

```
## # A tibble: 177 x 26
##     YEAR MONTH   DAY DAY_OF_WEEK AIRLINE FLIGHT_NUMBER ORIGIN_AIRPORT
##    <dbl> <dbl> <dbl>       <dbl> <chr>           <dbl> <chr>
##  1  2015     1     1           4 MQ               3274 ORD
##  2  2015     1     1           4 MQ               3155 ORD
##  3  2015     1     1           4 MQ               3048 ORD
##  4  2015     1     1           4 MQ               3319 ORD
##  5  2015     1     1           4 MQ               2873 ORD
##  6  2015     1     1           4 MQ               2762 ORD
##  7  2015     1     2           5 MQ               3274 ORD
##  8  2015     1     2           5 MQ               3155 ORD
##  9  2015     1     2           5 MQ               3048 ORD
## 10  2015     1     2           5 MQ               3319 ORD
## # ... with 167 more rows, and 19 more variables: DESTINATION_AIRPORT <chr>,
## #   SCHEDULED_DEPARTURE <dbl>, DEPARTURE_TIME <dbl>, DEPARTURE_DELAY <dbl>,
## #   TAXI_OUT <dbl>, SCHEDULED_TIME <dbl>, ELAPSED_TIME <dbl>, TAXI_IN <dbl>,
## #   SCHEDULED_ARRIVAL <dbl>, ARRIVAL_TIME <dbl>, ARRIVAL_DELAY <dbl>,
## #   DIVERTED <dbl>, CANCELLED <dbl>, CANCELLATION_REASON <chr>,
## #   AIR_SYSTEM_DELAY <dbl>, SECURITY_DELAY <dbl>, AIRLINE_DELAY <dbl>,
## #   LATE_AIRCRAFT_DELAY <dbl>, WEATHER_DELAY <dbl>
```

- Print out only the flights that are going to Willard Airport on the weekends.
- Make sure to remove `eval = FALSE` after your write the code!

```
flights %>%
  filter(DAY_OF_WEEK == "6" | DAY_OF_WEEK== "7") %>%
  filter(DESTINATION_AIRPORT == "CMI")
```

```
## # A tibble: 45 x 26
##     YEAR MONTH   DAY DAY_OF_WEEK AIRLINE FLIGHT_NUMBER ORIGIN_AIRPORT
##    <dbl> <dbl> <dbl>       <dbl> <chr>           <dbl> <chr>
##  1  2015     1     3           6 MQ               3274 ORD
##  2  2015     1     3           6 MQ               3155 ORD
##  3  2015     1     3           6 MQ               3048 ORD
##  4  2015     1     3           6 MQ               3319 ORD
##  5  2015     1     3           6 MQ               2873 ORD
##  6  2015     1     4           7 MQ               3274 ORD
##  7  2015     1     4           7 MQ               3155 ORD
##  8  2015     1     4           7 MQ               3048 ORD
##  9  2015     1     4           7 MQ               3319 ORD
## 10  2015     1     4           7 MQ               2873 ORD
## # ... with 35 more rows, and 19 more variables: DESTINATION_AIRPORT <chr>,
## #   SCHEDULED_DEPARTURE <dbl>, DEPARTURE_TIME <dbl>, DEPARTURE_DELAY <dbl>,
## #   TAXI_OUT <dbl>, SCHEDULED_TIME <dbl>, ELAPSED_TIME <dbl>, TAXI_IN <dbl>,
## #   SCHEDULED_ARRIVAL <dbl>, ARRIVAL_TIME <dbl>, ARRIVAL_DELAY <dbl>,
## #   DIVERTED <dbl>, CANCELLED <dbl>, CANCELLATION_REASON <chr>,
## #   AIR_SYSTEM_DELAY <dbl>, SECURITY_DELAY <dbl>, AIRLINE_DELAY <dbl>,
## #   LATE_AIRCRAFT_DELAY <dbl>, WEATHER_DELAY <dbl>
```

- Print out only the flights that are going to Willard Airport on the weekends that **are scheduled to arrive before 8:00 PM**.
- Make sure to remove `eval = FALSE` after your write the code!

```
flights %>%
  filter(DAY_OF_WEEK == "6" | DAY_OF_WEEK== "7") %>%
  filter(DESTINATION_AIRPORT == "CMI") %>%
  filter(SCHEDULED_ARRIVAL <= 2000)
```

```
## # A tibble: 36 x 26
##     YEAR MONTH   DAY DAY_OF_WEEK AIRLINE FLIGHT_NUMBER ORIGIN_AIRPORT
##    <dbl> <dbl> <dbl>       <dbl> <chr>           <dbl> <chr>
##  1  2015     1     3           6 MQ               3274 ORD
##  2  2015     1     3           6 MQ               3155 ORD
##  3  2015     1     3           6 MQ               3048 ORD
##  4  2015     1     3           6 MQ               3319 ORD
##  5  2015     1     4           7 MQ               3274 ORD
##  6  2015     1     4           7 MQ               3155 ORD
##  7  2015     1     4           7 MQ               3048 ORD
##  8  2015     1     4           7 MQ               3319 ORD
##  9  2015     1    10           6 MQ               3546 ORD
## 10  2015     1    10           6 MQ               3155 ORD
## # ... with 26 more rows, and 19 more variables: DESTINATION_AIRPORT <chr>,
## #   SCHEDULED_DEPARTURE <dbl>, DEPARTURE_TIME <dbl>, DEPARTURE_DELAY <dbl>,
## #   TAXI_OUT <dbl>, SCHEDULED_TIME <dbl>, ELAPSED_TIME <dbl>, TAXI_IN <dbl>,
## #   SCHEDULED_ARRIVAL <dbl>, ARRIVAL_TIME <dbl>, ARRIVAL_DELAY <dbl>,
## #   DIVERTED <dbl>, CANCELLED <dbl>, CANCELLATION_REASON <chr>,
## #   AIR_SYSTEM_DELAY <dbl>, SECURITY_DELAY <dbl>, AIRLINE_DELAY <dbl>,
## #   LATE_AIRCRAFT_DELAY <dbl>, WEATHER_DELAY <dbl>
```

select function

- Of all the flights, print out only the following columns: DESTINATION, DAY_OF_WEEK, SCHEDULED_DEPARTURE, DEPARTURE_TIME, DEPARTURE_DELAY, SCHEDULED_ARRIVAL, ARRIVAL_TIME, ARRIVAL_DELAY.
- Make sure to remove eval = FALSE after your write the code!

```r
flights %>%
  select(DESTINATION_AIRPORT,DAY_OF_WEEK,SCHEDULED_DEPARTURE,DEPARTURE_TIME,
         DEPARTURE_DELAY,SCHEDULED_ARRIVAL,ARRIVAL_TIME,ARRIVAL_DELAY)
```

```
## # A tibble: 23,484 x 8
##    DESTINATION_AIR~ DAY_OF_WEEK SCHEDULED_DEPAR~ DEPARTURE_TIME DEPARTURE_DELAY
##    <chr>                  <dbl>            <dbl>          <dbl>           <dbl>
##  1 PHX                        4              500            459              -1
##  2 IAH                        4              510            514               4
##  3 FLL                        4              530            526              -4
##  4 DEN                        4              533            540               7
##  5 DTW                        4              535            550              15
##  6 BOS                        4              540            529             -11
##  7 LGA                        4              556            547              -9
##  8 ATL                        4              600            602               2
##  9 MIA                        4              600             NA              NA
## 10 MCO                        4              608            603              -5
## # ... with 23,474 more rows, and 3 more variables: SCHEDULED_ARRIVAL <dbl>,
## #   ARRIVAL_TIME <dbl>, ARRIVAL_DELAY <dbl>
```

- Of all the flights going to Willard Airport on the weekend, print out all the columns except the following ones: AIRLINE_DELAY, SECURITY_DELAY, AIR_SYSTEM_DELAY.
- Make sure to remove eval = FALSE after your write the code!

```r
flights %>%
  filter(DAY_OF_WEEK == "6" | DAY_OF_WEEK== "7") %>%
  filter(DESTINATION_AIRPORT == "CMI") %>%
  select(-AIRLINE_DELAY,-SECURITY_DELAY,-AIR_SYSTEM_DELAY)
```

```
## # A tibble: 45 x 23
##     YEAR MONTH   DAY DAY_OF_WEEK AIRLINE FLIGHT_NUMBER ORIGIN_AIRPORT
##    <dbl> <dbl> <dbl>       <dbl> <chr>           <dbl> <chr>
##  1  2015     1     3           6 MQ               3274 ORD
##  2  2015     1     3           6 MQ               3155 ORD
##  3  2015     1     3           6 MQ               3048 ORD
##  4  2015     1     3           6 MQ               3319 ORD
##  5  2015     1     3           6 MQ               2873 ORD
##  6  2015     1     4           7 MQ               3274 ORD
##  7  2015     1     4           7 MQ               3155 ORD
##  8  2015     1     4           7 MQ               3048 ORD
##  9  2015     1     4           7 MQ               3319 ORD
## 10  2015     1     4           7 MQ               2873 ORD
## # ... with 35 more rows, and 16 more variables: DESTINATION_AIRPORT <chr>,
## #   SCHEDULED_DEPARTURE <dbl>, DEPARTURE_TIME <dbl>, DEPARTURE_DELAY <dbl>,
## #   TAXI_OUT <dbl>, SCHEDULED_TIME <dbl>, ELAPSED_TIME <dbl>, TAXI_IN <dbl>,
## #   SCHEDULED_ARRIVAL <dbl>, ARRIVAL_TIME <dbl>, ARRIVAL_DELAY <dbl>,
## #   DIVERTED <dbl>, CANCELLED <dbl>, CANCELLATION_REASON <chr>,
## #   LATE_AIRCRAFT_DELAY <dbl>, WEATHER_DELAY <dbl>
```

mutate function

- Add a column that is the the ratio of the total taxing time (`TAXI_IN` and `TAXI_OUT`) and the flying time (`ELAPSED_TIME`). Name this new coumn `TAXI_RATIO`.
- Make sure to remove `eval = FALSE` after your write the code!

```
flights %>%
  mutate (TAXI_RATIO = (TAXI_IN+TAXI_OUT)/ELAPSED_TIME)
```

```
## # A tibble: 23,484 x 27
##     YEAR MONTH   DAY DAY_OF_WEEK AIRLINE FLIGHT_NUMBER ORIGIN_AIRPORT
##    <dbl> <dbl> <dbl>       <dbl> <chr>           <dbl> <chr>
##  1  2015     1     1           4 US                602 ORD
##  2  2015     1     1           4 UA               1500 ORD
##  3  2015     1     1           4 NK                409 ORD
##  4  2015     1     1           4 UA               1167 ORD
##  5  2015     1     1           4 EV               5498 ORD
##  6  2015     1     1           4 B6               1012 ORD
##  7  2015     1     1           4 NK                224 ORD
##  8  2015     1     1           4 DL                977 ORD
##  9  2015     1     1           4 F9               1256 ORD
## 10  2015     1     1           4 UA                654 ORD
## # ... with 23,474 more rows, and 20 more variables: DESTINATION_AIRPORT <chr>,
## #   SCHEDULED_DEPARTURE <dbl>, DEPARTURE_TIME <dbl>, DEPARTURE_DELAY <dbl>,
## #   TAXI_OUT <dbl>, SCHEDULED_TIME <dbl>, ELAPSED_TIME <dbl>, TAXI_IN <dbl>,
## #   SCHEDULED_ARRIVAL <dbl>, ARRIVAL_TIME <dbl>, ARRIVAL_DELAY <dbl>,
## #   DIVERTED <dbl>, CANCELLED <dbl>, CANCELLATION_REASON <chr>,
## #   AIR_SYSTEM_DELAY <dbl>, SECURITY_DELAY <dbl>, AIRLINE_DELAY <dbl>,
## #   LATE_AIRCRAFT_DELAY <dbl>, WEATHER_DELAY <dbl>, TAXI_RATIO <dbl>
```

---

groupby and summarize functions

- Find the average departure delay time by destination and day of the week.
- Make sure to remove `eval = FALSE` after your write the code!

```
flights %>%
  group_by(DESTINATION_AIRPORT,DAY_OF_WEEK) %>%
  summarize(ave_dep_delay_time = mean(DEPARTURE_DELAY, na.rm = TRUE),
            n = n())
```

```
## # A tibble: 1,027 x 4
## # Groups:   DESTINATION_AIRPORT [154]
##    DESTINATION_AIRPORT DAY_OF_WEEK ave_dep_delay_time     n
##    <chr>                     <dbl>              <dbl> <int>
##  1 ABE                           1            NaN         1
##  2 ABE                           2             -3.25      4
##  3 ABE                           3             -5         1
##  4 ABE                           4             -6.5       3
##  5 ABE                           5              2.17      6
##  6 ABE                           6             -8.5       2
##  7 ABE                           7             30.7       3
##  8 ABQ                           1             54.8       4
##  9 ABQ                           3              9.25      4
## 10 ABQ                           4             27         6
## # ... with 1,017 more rows
```

- Find the median taxi out time by airline and day of the week.
- Make sure to remove `eval = FALSE` after your write the code!

```
flights %>%
  group_by(AIRLINE,DAY_OF_WEEK) %>%
  summarize(med_taxiout_time = median(TAXI_OUT, na.rm = TRUE),
            n=n())
```

```
## # A tibble: 84 x 4
## # Groups:   AIRLINE [12]
##    AIRLINE DAY_OF_WEEK med_taxiout_time     n
##    <chr>         <dbl>            <dbl> <int>
##  1 AA                1             14     521
##  2 AA                2             15     487
##  3 AA                3             14     504
##  4 AA                4             13     641
##  5 AA                5             13     665
##  6 AA                6             13     552
##  7 AA                7             15     529
##  8 AS                1             17      12
##  9 AS                2             18.5    12
## 10 AS                3             20.5    12
## # ... with 74 more rows
```

- Find the number of canceled flights for each airline.
- Make sure to remove `eval = FALSE` after your write the code!

```
flights %>%
  group_by(AIRLINE) %>%
  summarise(canceled_flights = sum(CANCELLED == "1"))
```

```
## # A tibble: 12 x 2
##    AIRLINE canceled_flights
##    <chr>              <int>
##  1 AA                    87
##  2 AS                     0
##  3 B6                     9
##  4 DL                     2
##  5 EV                   103
##  6 F9                     3
##  7 MQ                   603
##  8 NK                     6
##  9 OO                   151
## 10 UA                   132
## 11 US                    22
## 12 VX                     0
```

- Find the ratio of canceled flights and the number of scheduled flights for each airline.
- Make sure to remove `eval = FALSE` after your write the code!

```
cancel <- flights %>%
  group_by(AIRLINE) %>%
  summarise(canceled_flights = sum(CANCELLED == "1"),
            total_flights = n())
cancelratio <- cancel %>%
  mutate(cancel_ratio = canceled_flights/total_flights)
cancelratio
```

```
## # A tibble: 12 x 4
##    AIRLINE canceled_flights total_flights cancel_ratio
##    <chr>              <int>         <int>        <dbl>
##  1 AA                    87          3899      0.0223
##  2 AS                     0           100      0
##  3 B6                     9           170      0.0529
##  4 DL                     2           569      0.00351
##  5 EV                   103          3767      0.0273
##  6 F9                     3           283      0.0106
##  7 MQ                   603          5655      0.107
##  8 NK                     6           767      0.00782
##  9 OO                   151          3181      0.0475
## 10 UA                   132          4383      0.0301
## 11 US                    22           634      0.0347
## 12 VX                     0            76      0
```

---

arrange function

- Of the airlines that have at least 1000 scheduled flights, find the airline with the best canceling ratio record.
- Make sure to remove `eval = FALSE` after your write the code!

```
cancelratio %>%
  arrange(desc(cancel_ratio))
```

```
## # A tibble: 12 x 4
##    AIRLINE canceled_flights total_flights cancel_ratio
##    <chr>              <int>         <int>        <dbl>
##  1 MQ                   603          5655      0.107
##  2 B6                     9           170      0.0529
##  3 OO                   151          3181      0.0475
##  4 US                    22           634      0.0347
##  5 UA                   132          4383      0.0301
##  6 EV                   103          3767      0.0273
##  7 AA                    87          3899      0.0223
##  8 F9                     3           283      0.0106
##  9 NK                     6           767      0.00782
## 10 DL                     2           569      0.00351
## 11 AS                     0           100      0
## 12 VX                     0            76      0
```

```
#Airline MQ has the highest (best?) cancelling ratio.
```