nycflights

Group A

3/4/2022

For these exercises, we need to load the following packages:

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

Load dataset

```
data(flights)
glimpse(flights)
```

```
## Rows: 336,776
## Columns: 19
## $ year           <int> 2013, 2013, 2013, 2013, 2013, 2013, 2013, 2013, 2013, 2~
## $ month          <int> 1, 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, 1~
## $ dep_time       <int> 517, 533, 542, 544, 554, 554, 555, 557, 557, 558, 558, ~
## $ sched_dep_time <int> 515, 529, 540, 545, 600, 558, 600, 600, 600, 600, 600, ~
## $ dep_delay      <dbl> 2, 4, 2, -1, -6, -4, -5, -3, -3, -2, -2, -2, -2, -2, -1~
## $ arr_time       <int> 830, 850, 923, 1004, 812, 740, 913, 709, 838, 753, 849,~
## $ sched_arr_time <int> 819, 830, 850, 1022, 837, 728, 854, 723, 846, 745, 851,~
## $ arr_delay      <dbl> 11, 20, 33, -18, -25, 12, 19, -14, -8, 8, -2, -3, 7, -1~
## $ carrier        <chr> "UA", "UA", "AA", "B6", "DL", "UA", "B6", "EV", "B6", "~
## $ flight         <int> 1545, 1714, 1141, 725, 461, 1696, 507, 5708, 79, 301, 4~
## $ tailnum        <chr> "N14228", "N24211", "N619AA", "N804JB", "N668DN", "N394~
## $ origin         <chr> "EWR", "LGA", "JFK", "JFK", "LGA", "EWR", "EWR", "LGA",~
## $ dest           <chr> "IAH", "IAH", "MIA", "BQN", "ATL", "ORD", "FLL", "IAD",~
## $ air_time       <dbl> 227, 227, 160, 183, 116, 150, 158, 53, 140, 138, 149, 1~
## $ distance       <dbl> 1400, 1416, 1089, 1576, 762, 719, 1065, 229, 944, 733, ~
## $ hour           <dbl> 5, 5, 5, 5, 6, 5, 6, 6, 6, 6, 6, 6, 6, 6, 6, 5, 6, 6, 6~
## $ minute         <dbl> 15, 29, 40, 45, 0, 58, 0, 0, 0, 0, 0, 0, 0, 0, 0, 59, 0~
## $ time_hour      <dttm> 2013-01-01 05:00:00, 2013-01-01 05:00:00, 2013-01-01 0~
```

(1) Find the number of flights for three different subsets

(a) Flights that had an arrival delay of two or more hours

```
flights |>
  filter(arr_delay >= 2) |>
  nrow()
```

```
## [1] 127929
```

127929 flights had an arrival delay of two or more hours.

(b) flights that flew from JFK to Houston (IAH or HOU)

```
flights |>
  filter(origin == "JFK", (dest == "IAH" | dest == "HOU")) |>
  nrow()
```

```
## [1] 988
```

988 flights flew from JFK to Houston.

(c) departed between midnight and 6am (inclusive)

```
flights |>
  filter(dep_time == 2400 | dep_time <= 600) |>
  nrow()
```

```
## [1] 9373
```

9373 flights departed between midnight and 6am (inclusive).

(2)

(a) How many flights have a missing `dep_time`?

```
q2_a <- flights |>
  filter(is.na(dep_time)) |>
  nrow()

q2_a
```

```
## [1] 8255
```

(b) Do all of these flights also have a missing `arr_time`?

```
flights |>
  filter(is.na(dep_time), is.na(arr_time)) |>
  nrow() |>
  all.equal(q2_a)
```

```
## [1] TRUE
```

Yes, all of these flights also have a missing `arr_time`.

(c) What flights might be represented by missing `dep_time`?

Missing `dep_time` likely represents cancelled flights. They never depart therefore `dep_time` and `arr_time` would be both missing.

(3) Which ten destinations had the highest mean air time?

```
flights |>
  group_by(dest) |>
  summarise(mean_air_time = mean(air_time, na.rm = TRUE)) |>
  slice_max(mean_air_time, n = 10)
```

```
## # A tibble: 10 x 2
##    dest  mean_air_time
##    <chr>         <dbl>
##  1 HNL            617.
##  2 ANC            413.
```

```
##  3 SJC            347.
##  4 SFO            346.
##  5 OAK            345.
##  6 SMF            336.
##  7 BUR            334.
##  8 PSP            333.
##  9 PDX            330.
## 10 LGB            330.
```

(4) Which ten flights had the slowest speed?

```
flights |>
  mutate(speed = distance / (air_time / 60)) |>
  select(air_time, distance, speed, dest) |>
  slice_min(speed, n = 10)
```

```
## # A tibble: 10 x 4
##    air_time distance speed dest
##       <dbl>    <dbl> <dbl> <chr>
##  1       75       96  76.8 PHL
##  2      141      199  84.7 ACK
##  3       61       94  92.5 PHL
##  4       59       94  95.6 PHL
##  5       60       96  96   PHL
##  6       60       96  96   PHL
##  7       59       96  97.6 PHL
##  8      131      214  98.0 DCA
##  9       57       96 101.  PHL
## 10       55       94 103.  PHL
```

(5) How can we use the function `ends_with()` to select the columns for the actual and scheduled departure times?

```
flights |>
  select(ends_with("dep_time"))
```

```
## # A tibble: 336,776 x 2
##    dep_time sched_dep_time
##       <int>          <int>
##  1      517            515
##  2      533            529
##  3      542            540
##  4      544            545
##  5      554            600
##  6      554            558
##  7      555            600
##  8      557            600
##  9      557            600
## 10      558            600
## # ... with 336,766 more rows
```

(6) Is there a similar function that we can use to select actual departure time, scheduled departure time, and departure delay?

```
q6 <- flights |>
  select(contains("dep"))

q6
```

```
## # A tibble: 336,776 x 3
##    dep_time sched_dep_time dep_delay
##       <int>          <int>     <dbl>
##  1      517            515         2
##  2      533            529         4
##  3      542            540         2
##  4      544            545        -1
##  5      554            600        -6
##  6      554            558        -4
##  7      555            600        -5
##  8      557            600        -3
##  9      557            600        -3
## 10      558            600        -2
## # ... with 336,766 more rows
```

(7) Compare `dep_time`, `sched_dep_time` and `dep_delay` in the tibble created in (6)

(a) Append a column `diff_time` with the difference between `dep_time` and `sched_dep_time`.

```
q6_a <- q6 |>
  mutate(diff_time = dep_time - sched_dep_time)

q6_a |>
  head(10)
```

```
## # A tibble: 10 x 4
##    dep_time sched_dep_time dep_delay diff_time
##       <int>          <int>     <dbl>     <int>
##  1      517            515         2         2
##  2      533            529         4         4
##  3      542            540         2         2
##  4      544            545        -1        -1
##  5      554            600        -6       -46
##  6      554            558        -4        -4
##  7      555            600        -5       -45
##  8      557            600        -3       -43
##  9      557            600        -3       -43
## 10      558            600        -2       -42
```

(b) How would you expect `diff_time` and `dep_delay` to be related?

We would expect `diff_time` and `dep_delay` to be the same since they both represent the delay between scheduled and actual departure times. However, we actually see that the two are different, mainly because `diff_time` is calculated by subtracting `sched_dep_time` from `dep_time` and since these times are represented by integers, they do not take into account the fact that an hour is 60 minutes long,

not 100. So, in calculating the difference, there is always a 40 minute error when the two times are on different hours.

(c) Fix the problem.

```
calc_time <- function(dep, sch) {
  # Obtain hour from time
  dep_h <- dep %/% 100
  sch_h <- sch %/% 100
  # Obtain minutes from time
  dep_m <- dep %% 100
  sch_m <- sch %% 100
  # Convert time in terms of minutes elapsed
  d_time <- dep_h * 60 + dep_m
  s_time <- sch_h * 60 + sch_m
  # if sch_h is greater than dep_h,
  # it's either the plane left earlier than scheduled or
  # it departed on the next day
  # the earliest the plane left is about 43 minutes before
  # therefore, if sch_h - dep_h > 1,
  # we can know for sure it departed after midnight
  if_else(
    sch_h - dep_h > 1,
    d_time + 24 * 60 - s_time,
    d_time - s_time
  )
}

q7_c <- q6 |>
  mutate(diff_time = calc_time(dep_time, sched_dep_time))

# check if diff_time and dep_delay are all equal
all.equal(q7_c$diff_time, q7_c$dep_delay)
```

```
## [1] TRUE
```

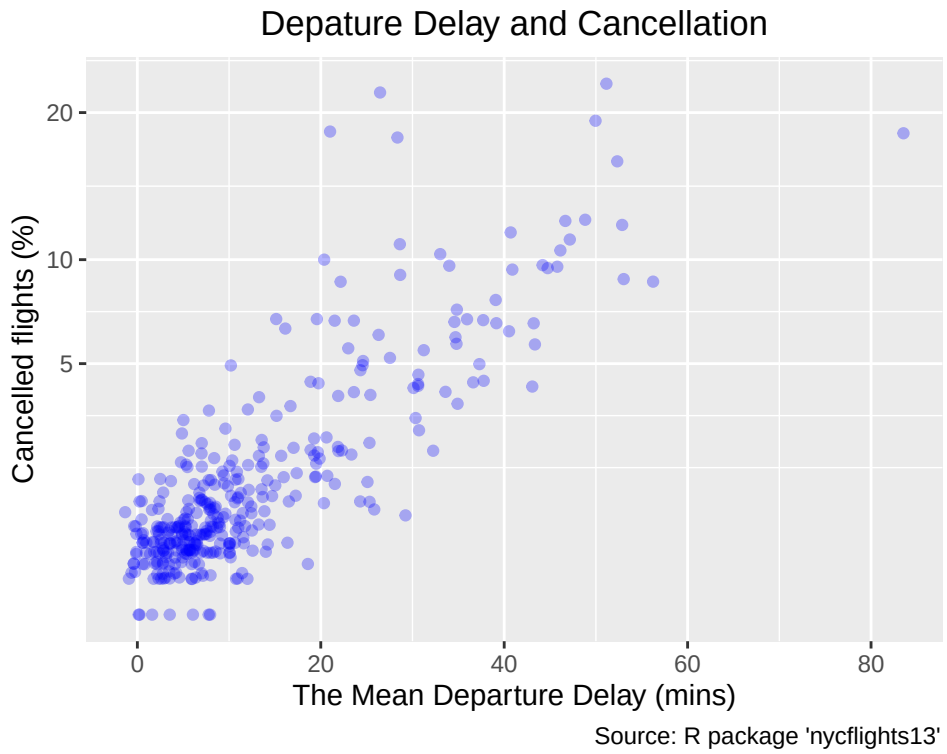
(8)

Make a scatter plot in which each point represents one day and the coordinates are:

- x: the mean departure delay (conditional on the departure delay being known).
- y: the percentage of cancelled flights. We consider a flight as cancelled if the departure time is `NA`.

```
q8 <- flights |>
  select(year, month, day, dep_time, dep_delay) |>
  group_by(year, month, day) |>
  mutate(
    mean_dep_delay = mean(dep_delay, na.rm = TRUE),
    perc = 100 * sum(is.na(dep_time)) / n()
  ) |>
  select(-dep_time, -dep_delay) |>
  ungroup() |>
  distinct() |>
  # filter out the outliers
  filter(perc < 40)
```

```
q8 |>
  ggplot(aes(mean_dep_delay, perc)) +
  geom_point(alpha = 0.3, colour = "blue", na.rm = TRUE) +
  labs(
    x = "The Mean Departure Delay (mins)",
    y = "Cancelled flights (%)",
    title = "Depature Delay and Cancellation",
    caption = "Source: R package 'nycflights13'"
  ) +
  theme(plot.title = element_text(hjust = 0.5)) +
  scale_y_continuous(breaks = c(5, 10, 20), trans = "sqrt")
```



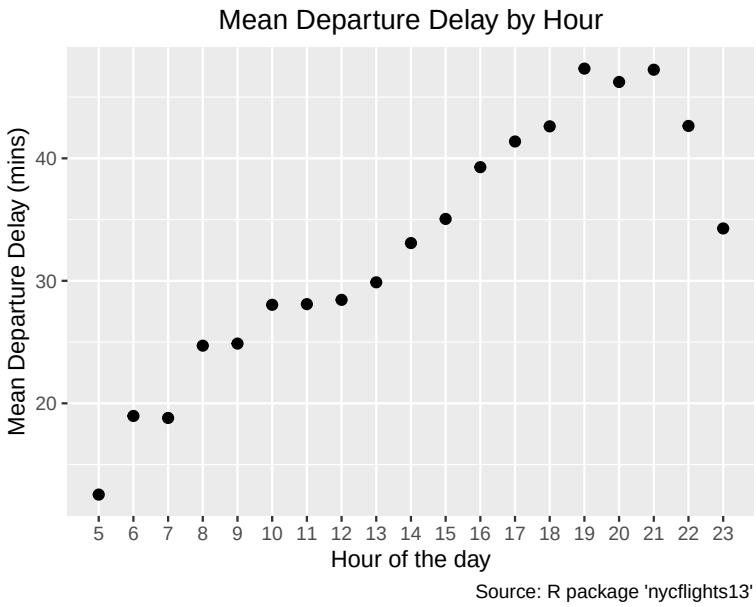
Judging from the plot, is the proportion of cancelled flights related to the mean departure delay?

Yes, it seems that the departure delay and cancellation rate are positively correlated.

(9) Make a plot that shows the mean departure delay by hour.

```
flights |>
  select(hour, dep_delay) |>
  filter(dep_delay >= 0) |>
  group_by(hour) |>
  summarise(mean_dep_delay = mean(dep_delay, na.rm = TRUE)) |>
  filter(!is.na(mean_dep_delay)) |>
  ggplot(aes(hour, mean_dep_delay)) +
  geom_point(size = 2) +
  labs(
    x = "Hour of the day",
    y = "Mean Departure Delay (mins)",
```

```
  title = "Mean Departure Delay by Hour",
  caption = "Source: R package 'nycflights13'"
) +
theme(plot.title = element_text(hjust = 0.5)) +
scale_x_continuous(
  breaks = seq(5, 24, by = 1),
  minor_breaks = NULL
)
```



What time of day should you fly if you want to avoid departure delays?

You should fly at 5 am to avoid departure delays.