R4DS Chapters 5 and 13

Nick Sun

May 8, 2019

5.7.1 Problem 2

Which plane in the `flights` dataset has the worst on-time record?

```
colnames(flights)
```

```
##  [1] "year"           "month"          "day"            "dep_time"
##  [5] "sched_dep_time" "dep_delay"      "arr_time"       "sched_arr_time"
##  [9] "arr_delay"      "carrier"        "flight"         "tailnum"
## [13] "origin"         "dest"           "air_time"       "distance"
## [17] "hour"           "minute"         "time_hour"
```

```
flights %>%
  filter(!is.na(tailnum) & !is.na(arr_delay)) %>%
  mutate(ontime = arr_delay <= 0) %>%
  group_by(tailnum) %>%
  summarise(
    on_time_prop = mean(ontime),
    flights = n()
  ) %>%
  filter(flights > 9) %>%
  arrange(on_time_prop) %>%
  head
```

```
## # A tibble: 6 x 3
##   tailnum on_time_prop flights
##   <chr>          <dbl>   <int>
## 1 N168AT        0.0588      17
## 2 N337AT        0.0769      13
## 3 N169AT        0.0909      11
## 4 N290AT        0.125       16
## 5 N273AT        0.154       13
## 6 N326AT        0.176       17
```

The plane with the worst on time record (minimum number of flights is 10) is N168AT with a on time proportion on around 5.8%.

5.7.1 Problem 4

For each destination, compute the total minutes of delay. For each flight, compute the total delay for its destination.

For the first part of this question, we can do:

```r
colnames(flights)
```

```
##  [1] "year"           "month"          "day"            "dep_time"
##  [5] "sched_dep_time" "dep_delay"      "arr_time"       "sched_arr_time"
##  [9] "arr_delay"      "carrier"        "flight"         "tailnum"
## [13] "origin"         "dest"           "air_time"       "distance"
## [17] "hour"           "minute"         "time_hour"
```

```r
flights %>%
  filter(arr_delay >= 0) %>%
  group_by(dest) %>%
  summarise(minutes_delayed = sum(arr_delay)) %>%
  arrange(desc(minutes_delayed)) %>%
  head
```

```
## # A tibble: 6 x 2
##   dest  minutes_delayed
##   <chr>           <dbl>
## 1 ATL            300299
## 2 ORD            283046
## 3 CLT            207441
## 4 MCO            206119
## 5 SFO            205406
## 6 LAX            203226
```

Atlanta has some serious delays.

For the second part we can do:

```r
flights %>%
  filter(arr_delay > 0) %>%
  group_by(dest, carrier) %>%
  summarise(
    total_arr_delay = sum(arr_delay)
  ) %>%
  group_by(dest) %>%
  mutate(
    arr_delay_prop = total_arr_delay / sum(total_arr_delay)
  ) %>%
  arrange(dest, desc(arr_delay_prop))
```

```
## # A tibble: 293 x 4
## # Groups:   dest [103]
##    dest  carrier total_arr_delay arr_delay_prop
##    <chr> <chr>             <dbl>          <dbl>
##  1 ABQ   B6                 4487        1
##  2 ACK   B6                 2974        1
##  3 ALB   EV                 9580        1
##  4 ANC   UA                   62        1
##  5 ATL   DL               157428        0.524
##  6 ATL   FL                56000        0.186
##  7 ATL   EV                42086        0.140
```

```
##  8 ATL   MQ                 41864        0.139
##  9 ATL   UA                  1982        0.00660
## 10 ATL   WN                   533        0.00177
## # ... with 283 more rows
```

5.7.1 Problem 6

Look at each destination. Can you find flights that are suspiciously fast? Compute the air time for a flight relative to the shortest flight to that destination. Which flights were most delayed in the air?

We can compute the average mean air times and identify unusual flights using the following code:

```
flights %>%
  group_by(origin, dest) %>%
  mutate(
    mean_air_time = mean(air_time, na.rm = TRUE)
  ) %>%
  group_by(flight) %>%
  mutate(
    flight_time_ratio = air_time / mean_air_time
  ) %>%
  select(
    origin, dest, flight, flight_time_ratio, air_time, mean_air_time
  ) %>%
  arrange(flight_time_ratio, desc(mean_air_time))
```

```
## # A tibble: 336,776 x 6
## # Groups:   flight [3,844]
##    origin dest  flight flight_time_ratio air_time mean_air_time
##    <chr>  <chr>  <int>             <dbl>    <dbl>         <dbl>
##  1 LGA    BOS     2132             0.555       21          37.9
##  2 LGA    ATL     1499             0.572       65         114.
##  3 EWR    GSP     4292             0.590       55          93.2
##  4 LGA    BOS     2142             0.608       23          37.9
##  5 EWR    BNA     3805             0.611       70         115.
##  6 EWR    MSP     4667             0.617       93         151.
##  7 EWR    CVG     4687             0.645       62          96.1
##  8 EWR    RIC     3830             0.654       35          53.5
##  9 JFK    BUF     2002             0.665       38          57.1
## 10 JFK    ROC       30             0.675       35          51.9
## # ... with 336,766 more rows
```

```
# %>%
#   select(origin, dest, mean_air_time) %>%
#   distinct %>%
#   arrange(mean_air_time)
```

Funkily short flights include flight 2132 (from LGA to BOS) and flight 1499 (from LGA to ATL) which had an air time of 65 minutes while the average air time is 113 minutes.

We can compare all flights to the shortest flight in their trip to identify just how delayed some flights were in the air.

```r
flights %>%
  group_by(origin, dest) %>%
  mutate(
    shortest_flight_time = min(air_time, na.rm = TRUE),
    air_time_ratio = air_time / shortest_flight_time
  ) %>%
  select(origin, dest, flight, air_time, shortest_flight_time, air_time_ratio) %>%
  filter(air_time_ratio != 1) %>%
  arrange(desc(air_time_ratio)) %>%
  head
```

```
## # A tibble: 6 x 6
## # Groups:   origin, dest [5]
##   origin dest  flight air_time shortest_flight_time air_time_ratio
##   <chr>  <chr>  <int>    <dbl>                <dbl>          <dbl>
## 1 LGA    BOS     2136      107                   21           5.10
## 2 LGA    DCA     2175      131                   32           4.09
## 3 JFK    ACK     1491      141                   35           4.03
## 4 EWR    BOS     1703      112                   30           3.73
## 5 JFK    BOS     1750       96                   26           3.69
## 6 LGA    BOS     2132       77                   21           3.67
```

13.4.6 Problem 1

Compute the average delay by destination then join on the airports dataframe so you can show the spatial distribution of delays

First we can grab the average delay by destination:

```r
flights %>%
  filter(arr_delay > 0) %>%
  group_by(dest) %>%
  summarise(
    average_delay = mean(arr_delay)
  ) %>%
  select(dest, average_delay) -> delay_by_dest

colnames(delay_by_dest) <- c("faa", "average_delay")

# airports %>%
#   left_join(delay_by_dest, by = "faa") %>%
#   ggplot(aes(lon, lat)) +
#     borders("state") +
#     geom_point(aes(fill = average_delay)) +
#     coord_quickmap()

delay_by_dest %>%
  left_join(airports, by = "faa") %>%
  select(faa, average_delay, name, lat, lon) %>%
  ggplot(aes(x = lon, y = lat)) +
    borders("state") +
    labs(title = "Eastern Seaboard has a ton of delays",
         subtitle = "Cherry Capital Airport in Michigan though takes the cake for longest average delay
```

```r
       x = "Longitude",
       y = "Latitude") +
  geom_point(aes(color = average_delay),
                size = 2,
                alpha = .8) +
  scale_color_continuous("Average Delay",
                         low = "#ffff99", high = "#e60000") +
  theme(
    legend.title = element_text(face = "bold",
                                size = 10)
  ) +
  theme_minimal() +
  coord_quickmap()
```

```
## Warning: Removed 4 rows containing missing values (geom_point).
```