Check-in 2

Code

AUTHOR
Maxim Dokukin

PUBLISHED
February 12, 2024

Remember, **follow the instructions below and use R Markdown to create a pdf document with your code and answers to the following questions on Gradescope.** You may find a template file by clicking "Code" in the top right corner of this page.

Collaborators

INSERT NAMES OF ANY COLLABORATORS

A. Flights Data

1. For this lab, we'll be using the `flights` data. You can load this data using the following code:

```r
library(tidyverse)
```

```
── Attaching core tidyverse packages ────────────────────────
tidyverse 2.0.0 ──
✔ dplyr     1.1.4     ✔ readr     2.1.5
✔ forcats   1.0.0     ✔ stringr   1.5.1
✔ ggplot2   3.4.4     ✔ tibble    3.2.1
✔ lubridate 1.9.3     ✔ tidyr     1.3.1
✔ purrr     1.0.2
── Conflicts ──────────────────────────────────────────
tidyverse_conflicts() ──
✖ dplyr::filter() masks stats::filter()
✖ dplyr::lag()    masks stats::lag()
ℹ Use the conflicted package (<http://conflicted.r-lib.org/>)
to force all conflicts to become errors
```

```r
library(nycflights13)
data(flights)
```

2. Use the `mutate()` function to turn `origin` into a factor.

```
  flights <- flights |> mutate(origin = as.factor(origin))
```

3. Compute the mean `arr_delay` for each origin airport. Which airport has the longest delays on average? You can drop missing rows.

```
mean_arr_delays_by_origin <- flights |>
  group_by(origin) |>
  summarize(mean_arr_delay = mean(arr_delay, na.rm = T)) |>
  ungroup()

print(mean_arr_delays_by_origin, n=3)
```

```
# A tibble: 3 × 2
  origin mean_arr_delay
  <fct>           <dbl>
1 EWR              9.11
2 JFK              5.55
3 LGA              5.78
```

Newark (EWR)

4. Use the `mutate()` function to turn `month` into a factor.

```
  flights <- flights |> mutate(month = as.factor(month))
```

5. Compute the mean `arr_delay` by month. What do you notice?

```
mean_arr_delays_by_months <- flights |>
  group_by(month) |>
  summarize(mean_arr_delay = mean(arr_delay, na.rm = T)) |>
  ungroup()

print(mean_arr_delays_by_months, n=12)
```

```
# A tibble: 12 × 2
   month mean_arr_delay
   <fct>          <dbl>
 1 1              6.13
 2 2              5.61
 3 3              5.81
 4 4             11.2
 5 5              3.52
 6 6              16.5
 7 7              16.7
 8 8               6.04
 9 9              -4.02
10 10             -0.167
11 11              0.461
12 12             14.9
```

Sep to Nov delays are the shortest.

6. Compute the mean `arr_delay` by month AND airport. What do you notice?

```
mean_arr_delays_by_ma <- flights |>
  group_by(month, origin) |>
  summarize(mean_arr_delay = mean(arr_delay, na.rm = T)) |>
  ungroup()
```

```
`summarise()` has grouped output by 'month'. You can override
using the
`.groups` argument.
```

```
print(mean_arr_delays_by_ma, n = 36)
```

```
# A tibble: 36 × 3
   month origin mean_arr_delay
   <fct> <fct>           <dbl>
 1 1     EWR            12.8
 2 1     JFK             1.37
 3 1     LGA             3.38
 4 2     EWR             8.78
 5 2     JFK             4.39
 6 2     LGA             3.15
 7 3     EWR            10.6
 8 3     JFK             2.58
 9 3     LGA             3.74
10 4     EWR            14.1
11 4     JFK             7.01
12 4     LGA            12.0
13 5     EWR             5.38
14 5     JFK             2.12
15 5     LGA             2.80
16 6     EWR            16.9
17 6     JFK            17.6
```

```
18 6     LGA            14.8
19 7     EWR            15.5
20 7     JFK            20.2
21 7     LGA            14.2
22 8     EWR             6.71
23 8     JFK             5.91
24 8     LGA             5.41
25 9     EWR            -4.73
26 9     JFK            -4.46
27 9     LGA            -2.83
28 10    EWR             2.60
29 10    JFK            -3.59
30 10    LGA             0.186
31 11    EWR             0.672
32 11    JFK            -0.873
33 11    LGA             1.55
34 12    EWR            19.6
35 12    JFK            12.7
36 12    LGA            12.0
```

Sep to Nov delays are the shortest.