Bài giảng R, số 2
-Khám phá phân tích dữ liệu với R, p3-

TS.Tô Đức Khánh

04/03/2024

1 Thống kê mô tả cho biến định tính

1.1 Bảng tần số, bảng tỷ số

Để miêu tả tổng hợp cho một biến định tính (có thể là định danh hoặc thứ bậc), ta có thể sử dụng bảng tần số (contingency table) hoặc bảng tỷ số (proportional table). Để tạo bảng tần số cho một biến định tính, ta sử dụng hàm `count()`. Dữ liệu `flights`, bao hàm thông tin các chuyến bay cất cánh từ thành phố New York trong năm 2013. Dữ liệu này được tổng hợp bởi US Bureau of Transportation Statistics, và được cung cấp trong thư viện `nycflights13`. Ta dùng hàm `data()` để nhập liệu được cung cấp trong một thư viện.

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

```
## # A tibble: 336,776 x 19
##    year month   day dep_time sched_dep_time dep_delay arr_time sched_arr_time
##   <int> <int> <int>    <int>          <int>     <dbl>    <int>          <int>
## 1  2013     1     1      517            515         2      830            819
## 2  2013     1     1      533            529         4      850            830
## 3  2013     1     1      542            540         2      923            850
## 4  2013     1     1      544            545        -1     1004           1022
## 5  2013     1     1      554            600        -6      812            837
## # i 336,771 more rows
## # i 11 more variables: arr_delay <dbl>, carrier <chr>, flight <int>, tailnum <chr>,
## #   origin <chr>, dest <chr>, air_time <dbl>, distance <dbl>, hour <dbl>,
## #   minute <dbl>, time_hour <dttm>
```

```
glimpse(flights)
```

```
## Rows: 336,776
## Columns: 19
## $ year           <int> 2013, 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, 1, 1, ~
## $ day            <int> 1, 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, 558, ~
## $ sched_dep_time <int> 515, 529, 540, 545, 600, 558, 600, 600, 600, 600, 600, 600, ~
## $ dep_delay      <dbl> 2, 4, 2, -1, -6, -4, -5, -3, -3, -2, -2, -2, -2, -2, -1, 0, ~
## $ arr_time       <int> 830, 850, 923, 1004, 812, 740, 913, 709, 838, 753, 849, 853,~
## $ sched_arr_time <int> 819, 830, 850, 1022, 837, 728, 854, 723, 846, 745, 851, 856,~
## $ arr_delay      <dbl> 11, 20, 33, -18, -25, 12, 19, -14, -8, 8, -2, -3, 7, -14, 31~
## $ carrier        <chr> "UA", "UA", "AA", "B6", "DL", "UA", "B6", "EV", "B6", "AA", ~
## $ flight         <int> 1545, 1714, 1141, 725, 461, 1696, 507, 5708, 79, 301, 49, 71~
```

```
## $ tailnum        <chr> "N14228", "N24211", "N619AA", "N804JB", "N668DN", "N39463", ~
## $ origin         <chr> "EWR", "LGA", "JFK", "JFK", "LGA", "EWR", "EWR", "LGA", "JFK~
## $ dest           <chr> "IAH", "IAH", "MIA", "BQN", "ATL", "ORD", "FLL", "IAD", "MCO~
## $ air_time       <dbl> 227, 227, 160, 183, 116, 150, 158, 53, 140, 138, 149, 158, 3~
## $ distance       <dbl> 1400, 1416, 1089, 1576, 762, 719, 1065, 229, 944, 733, 1028,~
## $ hour           <dbl> 5, 5, 5, 5, 6, 5, 6, 6, 6, 6, 6, 6, 6, 6, 6, 5, 6, 6, 6, 6, ~
## $ minute         <dbl> 15, 29, 40, 45, 0, 58, 0, 0, 0, 0, 0, 0, 0, 0, 0, 59, 0, 0, ~
## $ time_hour      <dttm> 2013-01-01 05:00:00, 2013-01-01 05:00:00, 2013-01-01 05:00:~
```

Ta lập bảng tần số cho biến `carrier` để xem có bao nhiêu hãng hàng không:

```
flights |> count(carrier)
```

```
## # A tibble: 16 x 2
##   carrier     n
##   <chr>   <int>
## 1 9E      18460
## 2 AA      32729
## 3 AS        714
## 4 B6      54635
## 5 DL      48110
## # i 11 more rows
```

Ở đây, ta đang áp dụng hàm `count()` lên biến `carrier` trong dữ liệu `flights`, kết quả thu được là bảng tần số cho số chuyến bay của các hãng hàng không. Bảng kết quả này, về mặt kỹ thuật trong `R` là một bảng dữ liệu với 2 cột, tương ứng, tên thể loại và tần số. Do đó, ta có thể tạo thêm cột biến mới chứ thông tin về tỷ số của các thể loại.

```
flights |> count(carrier) |> mutate(prop = n/sum(n))
```

```
## # A tibble: 16 x 3
##   carrier     n       prop
##   <chr>   <int>      <dbl>
## 1 9E      18460 0.0548139
## 2 AA      32729 0.0971833
## 3 AS        714 0.00212010
## 4 B6      54635 0.162229
## 5 DL      48110 0.142855
## # i 11 more rows
```

Đoạn lệnh ở trên tương đương với việc ta lưu bảng tần số và áp dụng `mutate()`:

```
table_flight <- flights |> count(carrier)
table_flight |> mutate(prop = n/sum(n))
```

Ngoài ra, ta cũng có thể tạo bảng tần số chéo cho 2 hoặc nhiều biến định tính. Ví dụ, ta muốn biết số chuyến máy bay xuất phát từ 3 sân bay: `EWR`, `JFK`, `LGA`:

```
flights |> count(origin, carrier)
```

```
## # A tibble: 35 x 3
##   origin carrier     n
##   <chr>  <chr>   <int>
## 1 EWR    9E       1268
## 2 EWR    AA       3487
## 3 EWR    AS        714
## 4 EWR    B6       6557
## 5 EWR    DL       4342
```

```
## # i 30 more rows
```

Kết quả trả ra là một bảng dữ liệu, do đó, ta có thể trình bày lại theo dạng bảng tần số hai chiều, bằng cách kết hợp với hàm `spread()`, ví dụ:

```
flights |> count(origin, carrier) |> spread(origin, n)
```

```
## # A tibble: 16 x 4
##   carrier   EWR   JFK   LGA
##   <chr>   <int> <int> <int>
## 1 9E       1268 14651  2541
## 2 AA       3487 13783 15459
## 3 AS        714    NA    NA
## 4 B6       6557 42076  6002
## 5 DL       4342 20701 23067
## # i 11 more rows
```

Ở dòng lệnh trên, ta tạo bảng tần số (`n`) với cột là các phân loại của `race`.

Một cách khác, có phần tiện lợi hơn, đó là sử dụng hàm `tabyl()` trong thư viện `janitor`, hàm này sẽ trả về kết quả là một bảng tần số cho 1 biến định tính, bảng tần số chéo của 2 biến định tính, và có thể là 3 biến định tính.

```
library(janitor)
flights |> tabyl(carrier, origin)
```

```
##  carrier   EWR   JFK   LGA
##       9E  1268 14651  2541
##       AA  3487 13783 15459
##       AS   714     0     0
##       B6  6557 42076  6002
##       DL  4342 20701 23067
##       EV 43939  1408  8826
##       F9     0     0   685
##       FL     0     0  3260
##       HA     0   342     0
##       MQ  2276  7193 16928
##       OO     6     0    26
##       UA 46087  4534  8044
##       US  4405  2995 13136
##       VX  1566  3596     0
##       WN  6188     0  6087
##       YV     0     0   601
```

Chú ý, biến nào viết trước sẽ là biến xuất hiện ở dòng của bảng tần số chéo. Cũng bằng sử dụng `tabyl()`, ta có thể tạo ra bảng tỷ số chéo của 2 biến định tính, bằng cách ghép thêm hàm `adorn_percentages()`, với đối số chuyền vào là `"row"` nếu ta muốn tính tỷ số theo dòng, hoặc `"col"` nếu ta muốn tính theo cột của bảng, ví dụ:

```
flights |> tabyl(carrier, origin) |> adorn_percentages("col")
```

```
##  carrier          EWR         JFK          LGA
##       9E 1.049365e-02 0.131660062 0.0242781525
##       AA 2.885753e-02 0.123859848 0.1477040378
##       AS 5.908884e-03 0.000000000 0.0000000000
##       B6 5.426408e-02 0.378112672 0.0573465059
##       DL 3.593330e-02 0.186027912 0.2203951769
##       EV 3.636281e-01 0.012652881 0.0843286006
```

```
##       F9 0.000000e+00 0.000000000 0.0065448778
##       FL 0.000000e+00 0.000000000 0.0311478856
##       HA 0.000000e+00 0.003073356 0.0000000000
##       MQ 1.883560e-02 0.064639330 0.1617396954
##       OO 4.965449e-05 0.000000000 0.0002484187
##       UA 3.814044e-01 0.040744435 0.0768569299
##       US 3.645467e-02 0.026914332 0.1255087806
##       VX 1.295982e-02 0.032315172 0.0000000000
##       WN 5.121033e-02 0.000000000 0.0581586440
##       YV 0.000000e+00 0.000000000 0.0057422942
```

Thực hành:

- Sử dụng `tabyl()` để tạo bảng tần số cho 1 biến định tính trong bảng dữ liệu. Nhận xét được gì từ bảng kết quả.
- Sử dụng `tabyl()` để tạo bảng tần số cho 2 biến định tính trong bảng dữ liệu. Nhận xét được gì từ bảng kết quả.
- Tạo ra bảng tỷ số chéo của 2 biến định tính, với tỷ số tính theo dòng.

Tìm hiểu thêm về cách dùng hàm `tabyl()` tại trang web https://cran.r-project.org/web/packages/janitor/vignettes/tabyls.html