Module 6 Notes

Justin Millar

2018-10-04

Download notes as a PDF

Upcoming Assignments/Quizzes

Assignments	Open Time	Due Time
Module 6 Data Quiz	October 5th (1:00 am EST)	October 7th (11:55 pm EST)
Module 6 Conceptual Quiz	October 5th (1:00 am EST)	October 7th (11:55 pm EST)

Notes from Discussion Board/Office Hours

Group summary statistics with the `aggregate()` function

Getting a summary statistics based on groups (e.g. a categorical variable) is a common activity in data analysis. One way to do this is in R is to make individual subsets of the dataframe for each level of the group using the subsetting methods we have used in previous modules. However, this can become arguous for groups that have many different categroies, and can end up filling up your memory if you're working with big data. A better approach is to use the `aggregate()` function. Here's an example using the `mtcars` dataset, where we calculate the mean `hp` for each type of `cyl`:

```
# Average horsepower for each cylinder type
aggregate(hp ~ cyl, data = mtcars, mean)
```

```
##   cyl        hp
## 1   4  82.63636
## 2   6 122.28571
## 3   8 209.21429
```

We can also save this output into a new object, and subset parts of the new dataframe to make comparisons:

```
# Average horsepower for each cylinder type
avg_hp_cyl <- aggregate(hp ~ cyl, data = mtcars, mean)

# On average, how much more horsepower do 6 cylinders than 4 cylinders?
avg_hp_cyl[2,2] - avg_hp_cyl[1,2]
```

```
## [1] 39.64935
```

Note that this is not only limited to calculating means, we can use other functions like `sum`, `min`, and `max`.

Group summary statistics with the `dplyr` package

Another way to do the same calculation is to us the `group_by()` and `summarize()` functions in the `dplyr` package. This approach is nice because we can use the pipe operator `%>%`, and it also works faster for large datasets than the base R approach:

```
library(dplyr, quietly = T)

mtcars %>%
```

```
  group_by(cyl) %>%
  summarize(avg_hp = mean(hp))
```

```
## # A tibble: 3 x 2
##     cyl avg_hp
##   <dbl>  <dbl>
## 1     4   82.6
## 2     6  122.
## 3     8  209.
```

Other notes

- Information on the final exam schedule will be coming shortly.
- If the Console pane in RStudio showing a `+` that means that R is expecting more information, which typically means that there is a missing `"` or `)`.
- Use the `==` (which reads as "is equal to") when subsetting, not the `=` (which is using for assignment).
- Don't worry if you're having trouble defining p-values, many scientists and researchers do too!