hw01: Set-up

Kenji Sato[*]
Kobe University

2017/11/27

1 Overview

Purpose

In this course, you will perform data analysis and simulation in the R programming language (R Core Team 2017), write your solution to homework assignments in R Markdown, and submit them through GitHub.

Sounds unfamiliar? That's why I designed this homework. The purpose of this assignment is twofold.

1. To check if your environment for computation and writing is up and running; and
2. To become familiar with GitHub workflow.

In this document, I will use a lot of functionalities of **rmarkdown** (Allaire et al. 2017), **knitr** (Xie 2016b) and **bookdown** (Xie 2016a) packages to demonstrate that you can write full academic papers with R Markdown. You don't need to understand all the details behind the scene now. After a bit of struggle with R, R Markdown and LaTeX, you will be able to tweak this source file to write your own paper.

In a nutshell

In this assignment, you will

- clone the assignment repository;
- make a working branch;
- replicate the present PDF file with your name on the author line; and
- commit the change and open a Pull Request.

2 Prerequisite

GitHub Account

Make a github account if you don't have one.

- GitHub

[*] Email: mail@kenjisato.jp

Software

Please install the following applications (they are free)

- R
- RStudio
- LaTeX system (texlive for Windows/Linux, MacTeX for macOS)
- GitHub Desktop
- Git for Windows for Windows

When you finish installation, follow the instructions below.

If you plan to use a computer at the computer lab, follow the instructions in https://github.com/rokko-ma17q2/hw-portal/wiki#computer-lab

3 Instructions

Here is the workflow for our assignments. Steps 1-3 and 7-11 are common to all the assignments and optional exercises.

Step 1. Clone the assignment repository

1. Click the invitation link you can find on https://github.com/rokko-ma17q2/hw-portal
2. Accept the invitation and then your assignment repository is automatically created.
3. Go to the repository that GitHub Classroom has made for you.
4. Click the green button saying "Clone or download" and then click "Open in Desktop".

Then GitHub Desktop will open.

Step 2. Make a new branch with a descriptive name

DO NOT start working on the problem yet! In GitHub Desktop, you should see something like Figure 1. It says that you are on the master branch, which is the default branch.

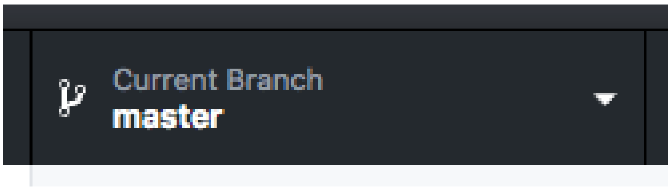


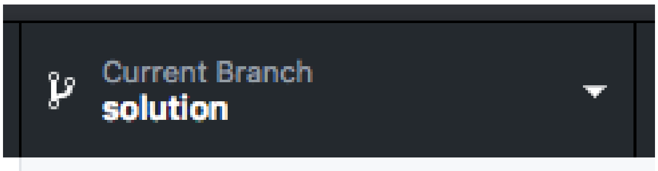
Figure 1: master branch

Make a new branch with a descriptive name by clicking the "Current Branch" tab. You'll find "New" button; clicking it will cause "Create a Branch" dialog to pop up. Enter the name for the new branch, say "`solution`".

Wait a minute. What is a branch? A branch corresponds to a "line of development." You may want to divert from the main line of development (`master` branch) to do some experiments. If the experiments turn out to be unsuccessful, you can discard the experimental branch without affecting the `master` branch. If they are successful, on the other hand, you might want the master branch import the updates made in the experimental branch. In Git terminology, this is called merge.

Branches are particularly useful when you work on a team. You can share your experiments, without affecting other members' work. Other members can review what you have done in your branch by checking the difference between merging and merged branches. In this course, the lecturers use this **diff** feature to check if the solutions seem correct.

When you are on a branch you create, you should see something like Figure 2 on the GitHub Desktop application.



Current Branch
solution

Figure 2: solution branch

Step 3. Open the assignment directory with RStudio

Then open the folder (hw01) in which the repository resides, by

- [Windows] Clicking the gear button and then "Open in Explorer"
- [Mac] Right-Clicking (Two-finger click) on the repository name in the left area, and then clicking "Open in Finder"

Double click hw01.Rproj file. RStudio will open.

Step 4. Open this assignment file

Look at the "Files" pane on RStudio, where you can find "solution.Rmd". Click it and now you see the source file (basically that's a trimmed version of this document).

Step 5. Knit (and become friends with errors)

Click "knit" button above the editor pane (Figure 3). If this is the first time you try to knit Rmd file, you are prompted to install and/or update dependencies.

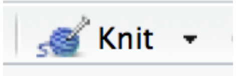


Figure 3: knit button

Success? Now, do you see a PDF file named solution.pdf? Everything seems to be okay? Congratulations, you are good to go!

If you don't see the PDF, hang in there. Since this is a common case, don't be ashamed of having caused an error. As you learn how to code, you'll encounter even more error messages (not less!!).

What you need to do is read the error message carefully. Try to figure out what caused it. After a careful reading of error message, you sometimes can fix it by yourself. But not always. If you still don't know what to do, copy the error message and search for an answer on Google. If you still can't fix, ask the TA or lecturer on the slack team. https://rokkoecon-slack-invite.herokuapp.com/

IMPORTANT!!! Don't ask us an "I can't compile... something went wrong" kind of vague question. Be an error handling machine and ask this way, "I did ... expecting to get ... but I got an error message instead saying (this and that)... I will send the whole error message and the result of `sessionInfo()` to you. Do you know how to get rid of it?"

Your problem was probably caused by missing packages? As a matter of fact, this Rmd depends on not-preinstalled packages called **tidyverse** (Wickham 2017) and **bookdown** (Xie 2016a). Please install them by running the following command in the console pane.

```r
install.packages("bookdown")
install.packages("tidyverse")
```

`install.packages("package_name")` is a typical way of installing packages on R. Please remember it.

Step 6. Make modification

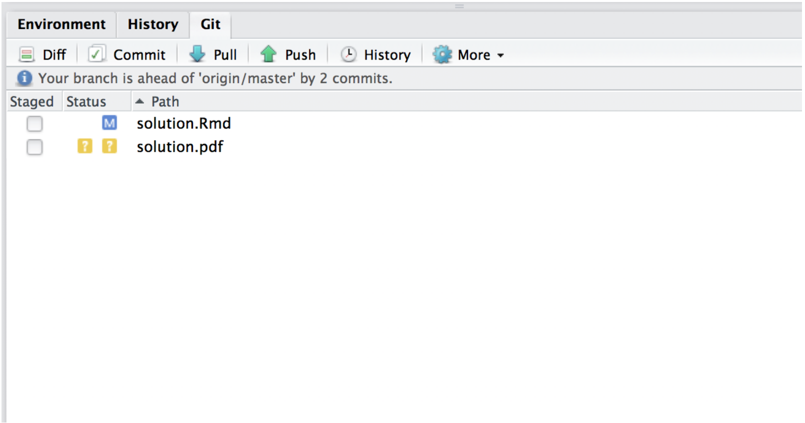
You may want to modify the source because the output PDF file has a strange author name on it: "Your name". Find the way to change the author name. Put your name there.

Save the file and knit again.

Now you know how to create a professional-quality paper (in appearance) with code, tables, graphs and mathematical expressions and references! (See Example sections below.)

Step 7. Commit

Find the Git pane. After the modification and knitting, it should look like Figure 4.



Your branch is ahead of 'origin/master' by 2 commits.

Staged	Status	Path
☐	M	solution.Rmd
☐	? ?	solution.pdf

Figure 4: Git pane

Stage `solution.Rmd` and `solution.pdf` by checking the box on the left of the file names. This corresponds to the git command `git-add`.[1]

[1] See https://www.atlassian.com/git/tutorials/saving-changes

Then click "Commit button." A new window will open and you will be requested to write a commit message. Write a short description of what you did and why. Click "Commit" button when finished.

Think of commit as journaling. When you've done something important, always make a commit to record the change. (Be generous when judging "importance.")

Step 8. Open a Pull Request

Let's go back to GitHub Desktop application. Find the homework repository on the left column. The History tab shows what you did. Check that every change you are supposed to make is highlighted by green (addition) and red (deletion).

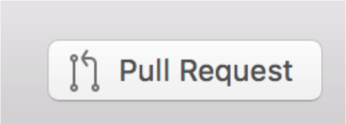


Figure 5: Pull Request

Click "Pull Request" button (Figure 5), write a short nice description and click "Send Pull Request."

Congratulations, you are done!

Step 9. Update the submission

We make mistakes. Sometimes, we discover them after submitting the report. Although you cannot erase the history, you can update the submission.

On RStudio, go to the Git pane. Check the modified file and commit. Then click **push**.

Step 10. Receive comments, update, and merge

Git is used for recording. GitHub is a communication platform based on Git. The lectuer exploits this communication feature to make your solutions better.

After you commit, the lecturer will

- request modifications if the solutions contain major mistakes,
- make comments and merge if there are small but important mistakes,
- merge without comments if there are no outstanding mistakes.

Please do as requested. After modification on the same branch, you can simply commit and push on RStudio. The Pull Request will be updated automatically.

Step 11. Receive suggested solution

Those who submit a homework assignment will be invited to a repository of mine or teaching assistant Beck's. The repository will contain a suggested solution to the assignment.

NB: You must show your efforts to receive the suggested solution. If the submission shows no trace of effort, we won't share the suggested solution.

In the rest of the handout, I show you a small portion of what you can do with R and Rmarkdown.

4 Example: R code and output in a document

```
library(tidyverse)
```

```
## Loading tidyverse: ggplot2
## Loading tidyverse: tibble
## Loading tidyverse: tidyr
## Loading tidyverse: readr
## Loading tidyverse: purrr
## Loading tidyverse: dplyr
```

```
## Conflicts with tidy packages ----------------------------------------------
```

```
## filter(): dplyr, stats
## lag():    dplyr, stats
```

```
(iris_tbl <- as_data_frame(iris))
```

```
## # A tibble: 150 x 5
##    Sepal.Length Sepal.Width Petal.Length Petal.Width Species
##           <dbl>       <dbl>        <dbl>       <dbl>  <fctr>
## 1           5.1         3.5          1.4         0.2  setosa
## 2           4.9         3.0          1.4         0.2  setosa
## 3           4.7         3.2          1.3         0.2  setosa
## 4           4.6         3.1          1.5         0.2  setosa
## 5           5.0         3.6          1.4         0.2  setosa
## 6           5.4         3.9          1.7         0.4  setosa
## 7           4.6         3.4          1.4         0.3  setosa
## 8           5.0         3.4          1.5         0.2  setosa
## 9           4.4         2.9          1.4         0.2  setosa
## 10          4.9         3.1          1.5         0.1  setosa
## # ... with 140 more rows
```

The raw output of R is not very beautiful for PDF. You probably want to use `knitr::kable()` function to print a table. Table 1 is produced by the following code.

```
knitr::kable(head(iris_tbl, 10), caption = "Iris dataset")
```

The following code generates and print Figure 6.[2]

```
ggplot(iris_tbl) +
  geom_point(aes(x = Sepal.Length, y = Petal.Length, color = Species))
```

5 Example: Mathematical expressions using LaTeX

You can produce mathematical equations using LaTeX syntax:

[2] See http://stackoverflow.com/questions/38861041/knitr-rmarkdown-latex-how-to-cross-reference-figures-and-tables/38884378#38884378

Table 1: Iris dataset

Sepal.Length	Sepal.Width	Petal.Length	Petal.Width	Species
5.1	3.5	1.4	0.2	setosa
4.9	3.0	1.4	0.2	setosa
4.7	3.2	1.3	0.2	setosa
4.6	3.1	1.5	0.2	setosa
5.0	3.6	1.4	0.2	setosa
5.4	3.9	1.7	0.4	setosa
4.6	3.4	1.4	0.3	setosa
5.0	3.4	1.5	0.2	setosa
4.4	2.9	1.4	0.2	setosa
4.9	3.1	1.5	0.1	setosa

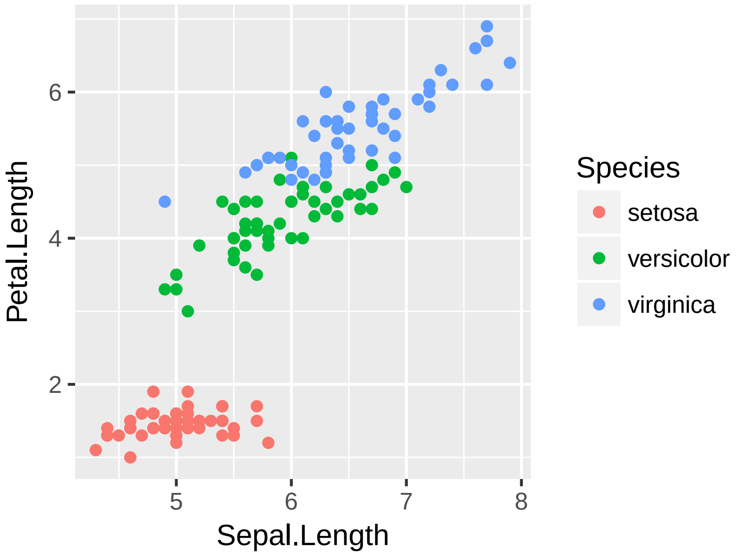


Figure 6: Iris Data

$$f(x) = f(0) + \int_0^x f'(y)dy. \tag{1}$$

Cross reference works like Equation (1) but the syntactic rule is different from LaTeX. See https://bookdown.org/yihui/bookdown/markdown-extensions-by-bookdown.html#equations for more detail.

References

Allaire, JJ, Joe Cheng, Yihui Xie, Jonathan McPherson, Winston Chang, Jeff Allen, Hadley Wickham, Aron Atkins, Rob Hyndman, and Ruben Arslan. 2017. *Rmarkdown: Dynamic Documents for R*. https://CRAN.R-project.org/package=rmarkdown.

R Core Team. 2017. *R: A Language and Environment for Statistical Computing*. Vienna, Austria: R Foundation for Statistical Computing. https://www.R-project.org/.

Wickham, Hadley. 2017. *Tidyverse: Easily Install and Load 'Tidyverse' Packages*. https://CRAN.R-project.org/package=tidyverse.

Xie, Yihui. 2016a. *Bookdown: Authoring Books and Technical Documents with R Markdown*. https://github.com/rstudio/bookdown.

———. 2016b. *Knitr: A General-Purpose Package for Dynamic Report Generation in R*. https://CRAN.R-project.org/package=knitr.