hw01: Set-up

Kenji Sato[*]

30 March, 2017

1 Overview

Purpose

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. 2016), **knitr** (Xie 2016b) and **bookdown** (Xie 2016a) to demonstrate that you can write full academic papers with Rmarkdown. You don't need to understand all the details behind the scene now. After a bit of struggle with R, Rmarkdown 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.

[*] Kobe University. Email: mail@kenjisato.jp

2 Prerequisite

GitHub Account

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

- GitHub

Softwares

Please install the following softwares (they are free)

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

When you finish installation, follow the instructions below.

3 Instructions

Step 1. Clone the assignment repository

1. Click the invitation link you've got from the lecturer or TA.
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.

Make a new branch with a descriptive name by clicking the button with the branch-shaped picture. Let's name the new branch as "`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

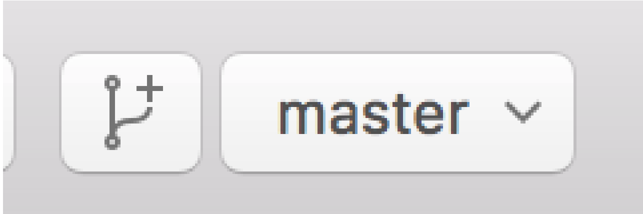


Figure 1: master branch

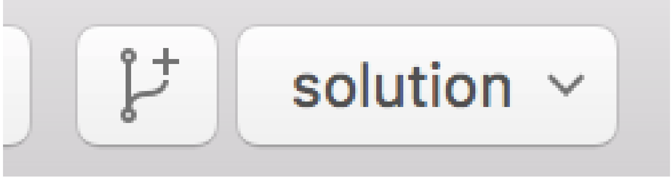


Figure 2: solution branch

experimental branch without affecting the `master` branch. If they are successful, on the other hand, you can make the `master` branch `merge` the experimental branch to be a part of the main line of development. Branches are particularly useful when you work on a team. You can share your experiments, without affecting other members' work.

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

Step 3. Open the assignment directory by 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 of this instruction manual.

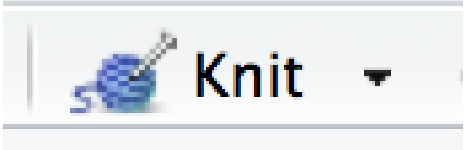


Figure 3: knit button

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.

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 ask google about that. If you still can't fix, ask the TA or lecturer.

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 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 package called **tidyverse** (Wickham 2017) and **bookdown** (Xie 2016a). Please install them by running the following command in the console pane.

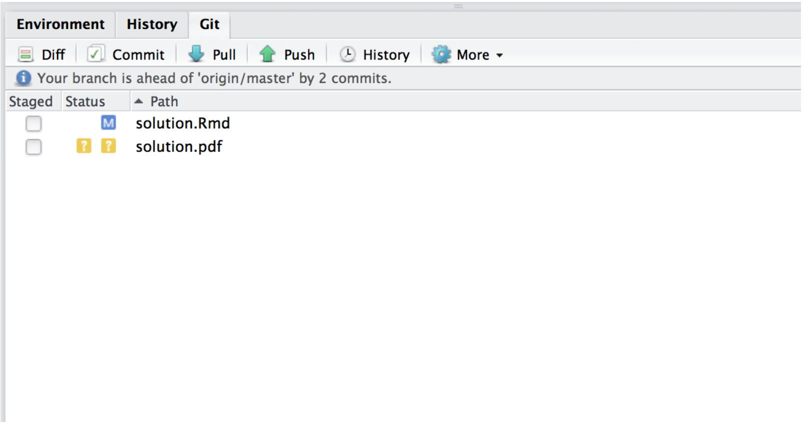
```r
install.packages("bookdown")
install.packages("tidyverse")
```

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

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.



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

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

Figure 4: Git pane

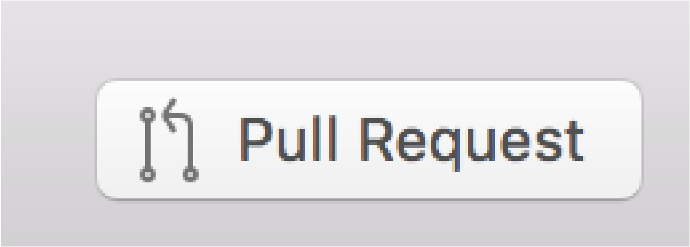


Figure 5: Pull Request

Step 7. Commit

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

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]

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.

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).

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

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

Congratulations, you are done!

In the rest of the handout, I show you 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 × 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")
```

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

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:

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

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, and Rob Hyndman. 2016. *Rmarkdown: Dynamic Documents for R*. https://CRAN.R-project.org/package=rmarkdown.

Wickham, Hadley. 2017. *Tidyverse: Easily Install and Load 'Tidyverse' Packages*. https:

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

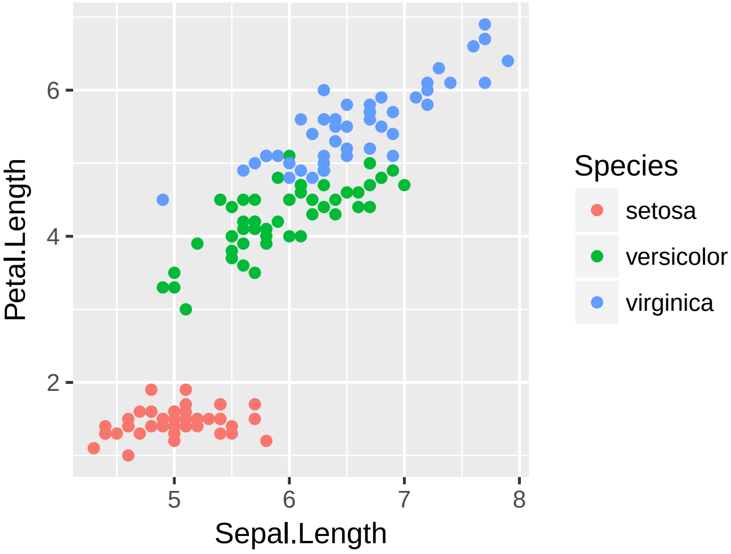


Figure 6: Iris Data

//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.