

hw01j: セットアップ

佐藤健治*

2017/04/03

1 概要

この課題は 課題 hw01 に対応する日本語版の課題です。両方を提出する必要はありませんが、両方の言語でレポートや論文を作成する可能性がある人はどちらもコンパイルできる (PDF を作成できる) ことを確認しておくといでしょう。中国語その他の言語での文書作成を希望していて、自分では解決できそうにない場合は相談してください。

目的

この課題の目的は以下の 2 つです。

1. 計算・執筆環境を整える
2. GitHub のワークフローに慣れる

この文書では Rmarkdown (Allaire et al. 2017), knitr (Xie 2016b) および bookdown (Xie 2016a) の機能をたくさん使っています。みなさんが論文を書くときに参考にできるようにとの配慮です。今すぐすべてを理解する必要はありません。

サマリー

この課題の中で、

- レポジトリのクローンする
- 作業ブランチの作成する
- この PDF を著者を変更して再作成する
- 変更をコミットしてプルリクエストを送信する

ことを学びます。

* 神戸大学 mail@kenjisato.jp

2 準備

GitHub アカウント

GitHub アカウントを持っていない場合は作ってください。

- [GitHub](#)

ソフトウェア

次のソフトウェアをインストールしてください。

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

インストールが完了したら次のステップに進んでください。

3 解答の手順

Step 1. 課題レポジトリをクローン (clone) する

1. 担当教員か TA から受け取った招待用リンクをクリックする
2. 課題への招待を受け入れると、GitHub Classroom があなたのためにレポジトリを作ります
3. 2 でできたレポジトリに進んでください
4. “Clone or download” と書かれた緑色のボタンをクリックし、“Open in Desktop” をクリック します。

GitHub Desktop アプリケーションが開きます。

Step 2. 新しいブランチを作る

ここで解答をはじめないでください！ GitHub Desktop のウインドウに図 1 のような表示が見えると思います。これは、あなたが **master** ブランチにいることを示しています。**master** ブランチは、デフォルトブランチです。

新しいブランチを作り、分かりやすい名前を作りましょう。図 1 左側の枝分かれ様のボタンを押して、よい名前をつけてください。例えばここでは、“**solution**” としておきましょう。

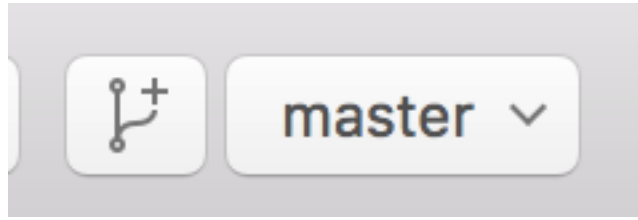


図 1: master branch

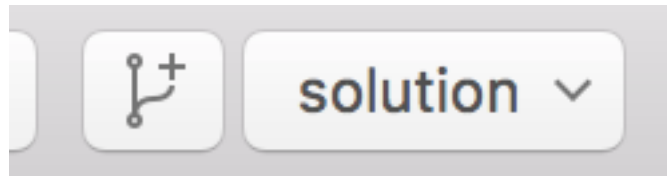


図 2: solution branch

そもそもブランチとは何かを簡単に説明しておきましょう。ブランチは「開発ライン」に対応しています。ときにはメインの開発ライン (**master**) から逸れて実験的な試みをやってみたいと思うかもしれません。そのようなときに新しいブランチを作ります。結果的に実験が失敗に終わったとしても、**master** に影響を与えることなく実験用ブランチを破棄できます。もし、実験が成功した場合には、メインの開発ラインに取り込む (**merge**) こともできます。ブランチはチームで作業するときに特に便利です。他のメンバーの開発環境を汚すことなく、自分が行った実験の成果を共有できます。

さて、図 2 のように変わったら完了です。

Step 3. 課題フォルダを RStudio で開く

フォルダ (hw01j) を OS のファイルシステムで開いてください。

- [Windows] ギアボタンをクリックして “Open in Explorer” をクリック
- [Mac] 左側に並んでいるレポジトリ名を 2 本指クリックして “Open in Finder” をクリック

ファイルシステムで hw01.Rproj ファイルをダブルクリックすると RStudio が起動します。

Step 4. ソースファイルを開く

RStudio の “Files” ペインの中に “solution.Rmd” を見つけてください。ファイル名をクリックすると、この文書のソースファイルが表示されます。



図 3: Knit ボタン

Step 5. Knit する。エラーと仲良くなる

エディターペインの上部にある“Knit”ボタン(図 3)をクリックしてください。はじめて Knit する場合には、必要なパッケージのインストールを行います。

成功しましたか？ `solution.pdf` を開いて表示に問題がないことを確認してください。問題なければ次に進むことができます。

もし、PDF が生成されていなくても諦めないでください。このようなことは日常茶飯事なので 1 つずつ解決していきましょう。心配いりません、プログラミングに熟練していくうちに会うエラーは増えます(減るのではなく！)。エラーにもすぐに慣れます。

やるべきことは、エラーメッセージを注意深く読むことです。何が原因かを考えて見てください。ときには自分で解決できることもあるでしょうが、多くの場合は他の人に頼ることも必要でしょう。エラーメッセージをコピーして Google で検索しましょう。多くの場合、あなたが経験した問題を他の誰かも経験して、インターネットのどこかに書き留めてくれています。それでも自力解決が難しい場合には TA か担当教員に相談してください。

重要!!! 「エラーがでて PDF ができないんです。どうしたらいいですか？」というような曖昧な質問はやめましょう。問題解決には問題が発生した状況を詳しく知る必要があります。「○○を期待して、××を試してみましたが、次のようなエラーが出てうまく行きません。エラー全文はこれです。解決方法はわかりますか？」という聞き方をしてください。

ひょっとしたらエラーはパッケージがインストールされていないからかもしれませんね。実はこの文書は幾つかのパッケージに依存しています。tidyverse (Wickham 2017) と bookdown (Xie 2016a) というパッケージを次のコマンドでインストールしてください。コンソールペインに次のコマンドを一行ずつ実行してください。

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

`install.packages("package_name")` は R でパッケージをインストールするための標準的な方法です。覚えておってください。

Step 6. ファイルを修正する

出力された PDF には氏名が書かれていません。Rmd ファイルのどの部分に氏名を入力すればよいかを見つけて、自分の名前書き換えてください。

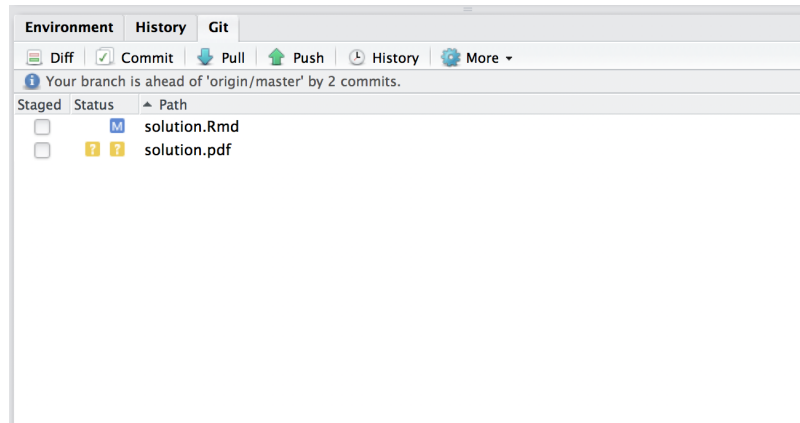


図 4: Git ペイン

できたらファイルを保存して、もう一度 Knit してください。

Step 7. コミット

RStudio で Git ペインを探してください。上で述べたとおりに修正して Knit すると、図 4 のようになっているはずです。

`solution.Rmd` と `solution.pdf` の左側にあるチェックボックスをチェックして、これらのファイルをステージングエリアに追加してください。これは `git-add` コマンドに対応しています¹。

次に “Commit” ボタンをクリックしてください。新しいウィンドウが開き、コミットメッセージを要求されます。何を変更・追加したか、その目的について簡単に記載し、“Commit” ボタンをクリックします。これで変更を記録することができました。

Step 8. Pull Request を送る

GitHub Desktop アプリケーションに戻りましょう。課題レポジトリを左側のリストから探してください。当該レポジトリを開き、History タブを表示すると、あなたの行った修正を表示することができます。行うべき修正が緑（追加）と赤（削除）でハイライトされていることを確認してください。

問題なければ “Pull Request” ボタン（図 5）をクリックし、説明を書いて “Send Pull Request” ボタンをクリックします。

おめでとうございます。これで課題の提出が完了です。

この文書の残りの部分では、R と Rmarkdown でできることの一部を紹介します。

¹<https://www.atlassian.com/git/tutorials/saving-changes> を参照

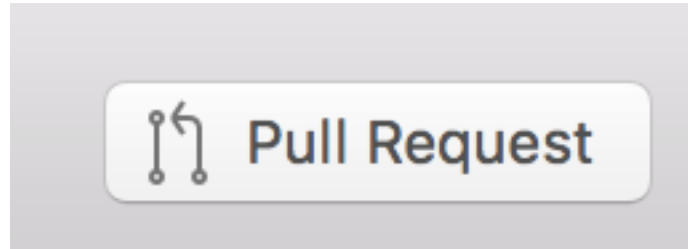


図 5: Pull Request

4 Example: R コードと出力を埋め込む

```
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
```

R のデータセットについては、生の出力はそれほど美しいものではありません。knitr::kable() 関数を使うと、少し見栄えがよくなります。表 1 は次のコードによって

表 1: Iris データセット

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

出力されたものです。

```
knitr::kable(head(iris_tbl, 10), caption = "Iris データセット")
```

次のコードは図 6 を生成します²。

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

5 Example: LaTeX を使った数式

LaTeX の構文を使って数式を書くことができます。

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

式 (1) という風に式を相互参照することもできます（これは bookdown パッケージで追加されている拡張機能です）。ただし、LaTeX とは異なる構文を使う必要があります。詳しくは、<https://bookdown.org/yihui/bookdown/markdown-extensions-by-bookdown.html#equations> を参照。

参考文献

Allaire, JJ, Joe Cheng, Yihui Xie, Jonathan McPherson, Winston Chang, Jeff Allen, Hadley Wickham, Aron Atkins, Rob Hyndman, and Ruben Arslan. 2017. Rmarkdown:

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

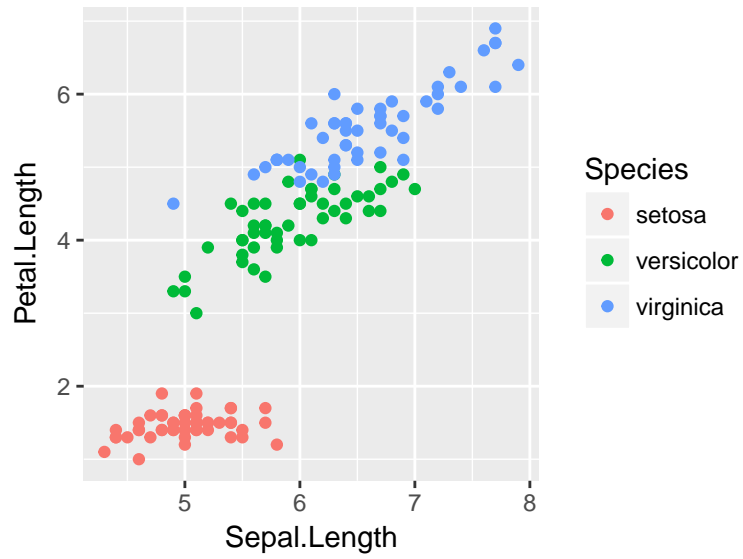


図 6: Iris データ

Dynamic Documents for R. <https://CRAN.R-project.org/package=rmarkdown>.

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>.