- DANL 310: Data Visualization and Presentation -

2. Getting Started with GitHub website and R Markdown

Byeong-Hak Choe

SUNY Geneseo

February 1, 2022

Team Project

- Each team consists of two or three students by midnight February 24, 2022.
- Each team must send web links for data, metadata (simple description of data) and 1-page proposal for the team project to Byeong-Hak Choe by midnight March 3, 2022 via email.
- The data for the team project must be approved by Byeong-Hak Choe.
- The write-up for the team project must include machine learning methods (e.g., linear regression, clustering) and several visualizations.
- The document for the team project must be published in each member's website.
- The document for the team project must be identical for both team members.

Grading

Team Project

The team project score is calculated as follows:

$$\begin{aligned}
&\text{(Team Project Score)} \\
= \quad & 0.15 \times \text{(Byeong-Hak's evaluation on Midterm Presentation)} \\
&+ 0.25 \times \text{(Byeong-Hak's evaluation on Final Presentation)} \\
&+ 0.50 \times \text{(Byeong-Hak's evaluation on Team Project Web-page)} \\
&+ 0.02 \times \text{(Average of peer evaluation on Midterm Presentation)} \\
&+ 0.03 \times \text{(Average of peer evaluation on Final Presentation)} \\
&+ 0.03 \times \text{(Average of peer evaluation on Team Project Web-page)} \\
&+ 0.02 \times \min \{ \text{(Self-evaluation on your contribution} \\
&\qquad\qquad\qquad \text{to the Team Project)}, \\
&\qquad\qquad\quad \text{(Your team member's evaluation on} \\
&\qquad\qquad\qquad \text{your contribution to the Team Project)} \} \,.
\end{aligned}$$

Grading

Team Project

- The highest and lowest peer evaluation scores from the other teams on the midterm presentation will be dropped when calculating the average of your peer evaluation score on the midterm presentation.
- The highest and lowest peer evaluation scores from the other teams on the final presentation will be dropped when calculating the average of your peer evaluation score on the final presentation.
- The highest and lowest peer evaluation scores from the other teams on the team project's web-page will be dropped when calculating the average of your peer evaluation score on the team project's web-page.
- If you do not submit your peer evaluation for the team projects' web-pages to Byeong-Hak before the due, you will receive a 10% of deduction on your team project score.

Personal website on GitHub

- We have a class website on GitHub (https://bcdanl.github.io/index.html).
- This website is built by R Markdown with R Studio.

Personal website on GitHub

Step 1

- Step 1.

    1. Create your GitHib account and login to it.
    2. From https://github.com, click the plus (+) icon in the upper right corner and select “New repository”.
    3. Name this repo `YOUR_GITHUB_NAME.github.io`, which will be the domain for your website.
    4. Copy the address of your git repo

       ```
       https://github.com/YOUR_GITHUB_NAME/
       YOUR_GITHUB_NAME.github.io
       ```

Personal website on GitHub

Step 2

- Step 2.
  1. Open R Studio.
  2. Select “New Project” under the “File” menu.
  3. Click “Version Control”.
  4. Click “Git”.
  5. Paste the address of your git repo in the “Repository URL” dialogue box: `https://github.com/YOUR_GITHUB_NAME/YOUR_GITHUB_NAME.github.io`
  6. Choose the “Project directory name” as `YOUR_GITHUB_NAME.github.io`.
  7. Select a location of the files for your personal website in your laptop and copy its path name to “Create project as subdirectory of:”.
  8. Click “Create Project”.

Personal website on GitHub

Step 3

- Step 3.
  1. Let's update our rmarkdown package to make sure we actually have the version that supports R Markdown websites.
     - `install.packages("rmarkdown", type = "source")`.
  2. Create a couple empty files inside your repository folder in your laptop.
     - Under "File" select "New File" then "R Markdown" - save this file as "`index.Rmd`".
     - Under "File" select "New File" then "R Markdown" - save this file as "`about.Rmd`".
     - Under "File" select "New File" then "Text File" - save this file as "`_site.yml`".

Personal website on GitHub

Step 4

- Step 4.
  - Fill out _site.yml as follows:

```
name: "personal-website"
output_dir: "."
navbar:
  title: "Personal Website"
  left:
    - text: "Home"
      href: index.html
    - text: "About Me"
      href: about.html
```

Personal website on GitHub

Step 4

- Step 4.
  - Fill out `index.Rmd` as follows:

```
---
title: "Personal Website"
---

Hello, World!
```

  - Fill out `about.Rmd` as follows:

```
---
title: "About Me"
---

Why I am awesome.
```

Personal website on GitHub

Step 5

- Step 5.
  1. Run the following within your R Studio project.

     ```
     rmarkdown::render_site()
     ```

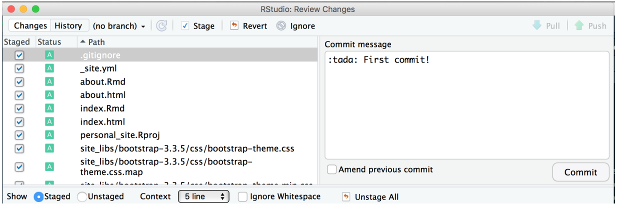
     - Now if everything has gone according to plan, you should get a bunch of output followed by the message : `Output created: index.html`.
  2. Open the repository folder with Finder (Mac) or File Explorer (Windows) in your laptop.
  3. Open the file, `index.html`.

Personal website on GitHub

Step 6

- Step 6: Now we just have to add commit and push everything to GitHub.

1. Click the “Git” tab in the upper right corner pane of RStudio.
2. Click the “Commit” button.
3. Click all of the files to stage them, type any commit message, and then click “Commit”.



Staged	Status	Path
☑	A	.gitignore
☑	A	_site.yml
☑	A	about.Rmd
☑	A	about.html
☑	A	index.Rmd
☑	A	index.html
☑	A	personal_site.Rproj
☑	A	site_libs/bootstrap-3.3.5/css/bootstrap-theme.css
☑	A	site_libs/bootstrap-3.3.5/css/bootstrap-theme.css.map
☑	A	site_libs/bootstrap-3.3.5/css/bootstrap-theme.min.css

Commit message:

```
:tada: First commit!
```

□ Amend previous commit

Show ● Staged ○ Unstaged Context 5 line □ Ignore Whitespace

4. Click the green up arrow to “push” the commit to GitHub.

Personal website on GitHub

Step 7

- Step 7: Return to your GitHub repository in the browser.
  1. Click the “Setting” tab in the menu.
  2. Scroll down to the “GitHub Pages” header. Under “source” select “master branch” and then click “Save”.
  3. Open the new tab in your browser. Go to your personal website:

     `https://YOUR_GITHUB_NAME.github.io`

Personal website on GitHub

Modifying your website

1. Modify `about.Rmd` as follows:

```
---
title: "About Me"
---

- __Name:__ YOUR_NAME
- __Ocupation:__ "Student"
- __Hobbies:__ Learning data analytics.

Here is a super cool photo.

![](FILE_NAME_OF_YOUR_PHOTO_IN_THE_REPO_FOLDER.jpg){
    width=50%}
```

2. Rebuild your site by running `rmarkdown::render_site()` in console.
3. Open `index.html` again to see if it worked.

Personal website on GitHub

Modifying your website

1. Create a new “Text file” in R Studio named “`style.css`” and add the following lines:

```
img {
   width: 85%;
   display: block;
   margin: 0 auto;
}
```

- Cascading Style Sheets (CSS) is used to format the layout of a webpage (color, font, text size, background, display, etc.).
  - HTML will format the architecture of the house;
  - CSS will be the carpet and walls to decorate the house;
  - JavaScript adds interactive elements in the house, such as opening doors and lighting.

Personal website on GitHub

Modifying your website

2. Add the following lines to your `_site.yml` file to apply this css to your site.

```
name: "YOUR_NAMEs-website"
output_dir: "."
navbar:
  title: "YOUR_NAME's Website"
  left:
    - text: "Home"
      href: index.html
    - text: "About Me"
      href: about.html
output:
  html_document:
    theme: default
    css: style.css
```

3. Run `rmarkdown::render_site()` to checkout how things have changed.

Personal website on GitHub

Adding projects/other links

1. Edit the `_site.yml` file as follows:

```
name: "YOUR_NAME's-website"
output_dir: "."
navbar:
  title: "YOUR_NAME's Website"
  left:
    - text: "Home"
      href: index.html
    - text: "About Me"
      href: about.html
    - text: "Project"      ##### the new
      href: project.html   ##### stuff
output:
  html_document:
    theme: default
    css: style.css
```

Personal website on GitHub

Adding projects/other links

1. Create another file called `project.Rmd` with the following text in the repository folder in your laptop.

```
---
title: "Team Project"
---

This is an example of the team project web-page.

&nbsp;

## US county map of climate change belief in 2018

&nbsp;
__Estimated percentage who think that global warming is
    caused mostly by human activities__

‘‘‘{r, echo = TRUE, eval = TRUE,  fig.align='center'}

‘‘‘
```

Personal website on GitHub

Adding projects/other links

2. Inside the chunk of “‘r, ... “‘ in the `project.Rmd` file, insert the following `ggplot` code:

```
cc_belief_county <- read.table(
  'https://bcecon.github.io/cc_belief_county.csv',
  sep = ',',
  header = TRUE, stringsAsFactor = TRUE
)
# US county map of climate change opinion
legend_title <- ""
ggplot(data = cc_belief_county) +
  geom_polygon(aes(long, lat, group = group,
                   fill = human2018),
               color = "grey", size = 0.1) +
  coord_map("bonne", parameters = 41.6) +
  scale_fill_gradient(legend_title, low='lightblue',
      high='hotpink') +
  theme_map() + theme(legend.position="right") +
  labs(caption = "Data source: Yale Climate Change
      Communication")
```

3. Run `rmarkdown::render_site()`.