DATA 227

Using Quarto for Reproducible Research

2022-10-13

Replication vs. Reproducibility

- To replicate an experiment, we re-perform the entire experiment, collect new data, and re-run the analysis.
- "In order for a scientific study to be replicated [...] the method of statistical analysis must be entirely **reproducible** [1]

[1] Stevens, J. R. (2017). Replicability and reproducibility in comparative psychology. Frontiers in psychology, 8, 862.

Reproducibility

- In order to something to be reproducible, you would need to...
  - Have the original dataset/file.
  - Know which data cleaning procedures were used.
  - Know which statistical test was applied.
  - Apply the statistical test making the exact same decisions (what parts of the data were used, which outliers were removed, etc.).

Reproducible Workflows

- A typical workflow has at least two components [2]:
  - A statistical software package for performing the data analysis, and
  - Software for presenting the results.

There are programs that combine these two components into one program!
Therefore, there is no loss of information from one component to another.

[2] Baumer, B., Cetinkaya-Rundel, M., Bray, A., Loi, L., & Horton, N. J. (2014). R Markdown: Integrating a reproducible analysis tool into introductory statistics. arXiv preprint arXiv:1402.1894.

Introduction to Quarto

- Quarto is an "open-source scientific and technical publishing system" that allows for "[weaving] together narrative text and code to produce elegantly formatted output." [3]
- Quarto works for...
  - VS Code
  - **RStudio**
  - **Jupyter**
  - Editor

In this lecture, we will go over some Quarto basics, but we will also make sure that everyone is able to produce assignments with publication-quality graphics for their next HW submission.

[3] https://quarto.org

Installing Quarto

Follow the Quarto Installation Link for instructions.

Make sure you install additional tools if you are using RStudio or VS Code.

Opening Documents

- **RStudio**
  - File -> New File -> Quarto Document (Fill in fields as necessary)
- **Jupyter**
  - File -> New -> Notebook
    - `quarto preview filename.ipynb` for a preview of your document.

Quarto Content

A Quarto file contains three types of content:

- a YAML header,
- code chunks, and
- markdown text.

YAML Headers 1

YAML looks something like this:

```
title: "Homework X"
author: "Your Name"
date: "The Due Date"
```

- In your document, it should be located in between two lines with three dashes each (---).
  - In RStudio, it's at the very top of your document.
  - In a Jupyter notebook, it's at the top in a "raw" cell.
- One of the most important parts of the YAML header is specifying the output of the document–we'll review that later in the lecture.

YAML Headers 2

- Notice that some of the information you entered when you created a new document is included in the header!
- For the most part, you will not need to worry about editing the YAML in this class.
  - You can copy and paste the header from the previous page and edit it as needed.
  - You can also use the point and click interface on RStudio, etc.
  - That being said, if you want to get really fancy with your reports, it's good to read up on Quarto's documentation for formatting. Note that the documentation you're reviewing will depend on your choice of output.

Code Chunks and Cells

This is where a lot of your focus will be directed (both when completing homework and if you choose to use Quarto for report writing in the future).

- R Chunk: Click on the green "+C" button, or type:

````
```{r}
# Type some R code here!
```
````

- Python Cell: Click on the black "+" button, and make sure it is a "Code" cell.

Writing Code

Once you insert a code chunk or cell into your document, you can insert whatever code you like. In your final document, the code you write will be evaluated and the output will be shown immediately after the chunk.

````
```{r}
2+2
5*4
```
````

```
[1] 4
[1] 20
```

This includes code to write graphics, which is our class's biggest need!

Running Code Chunks and Cells

Sometimes, you will want to run code while you are working.

- In RStudio...
  - The green triangle button in the upper right hand corner will run your current chunk.
  - The gray triangle/green bar button in the upper right hand corner will run all cells before your current chunk.
- In Python...
  - The gray triangle button in the toolbar will run your current cell.
  - The double gray triangle button will restart the terminal and run all cells.

Code Chunk and Cell Options

- To change the "options" in a cell, you can use the prefixed comment (`#|`).
- There is a list of all options you could use in RStudio and Jupyter. Be sure to check out what you can do with figures.
- Here is a list of the options you might use...
  - `eval`: If true, the chunk/cell will be evaluated.
  - `echo`: If true, the chunk/cell content will be repeated in the document. It's probably good to keep this on in your homework, but you could turn it off for a more formal report.
  - `warning`/`error`: If true, any warning or error messages will be included in your report. I would not mess with this for now, because you don't want to turn in a submission with errors.

Markdown

Quarto accepts markdown syntax! Just note that in Python, you need to insert a Markdown cell. Here are some options you may want to use:

- Section Headings: Use #, ##, ### depending on what size heading you want.
- Bold: Use two asterisks (**) on each side of the word.
- Italic: Use one asterisk (*) on each side of the word.
- Bullets: Use a single asterisk (*) for a bulleted list. Use an indented plus sign (+) for the second level of the list, and a double-indented minus sign (-) for the third.
- Links: Use square brackets around the text for the link, immediately followed by parentheses with the link inside.

Note! If you are using the "Visual" view in RStudio, you can use all these features as if you were working with a word processor.

Rendering in RStudio

Rendering will create the readable version of your document, with all of the code, graphics, and plain text in the same place. It is equivalent to knitting in RStudio.

1. Save your document with a helpful title (consider Lastname_HWX) in a helpful place (consider making a directory with all of the files for this class). Make sure your document has a .qmd extension.
2. Click the blue arrow button labeled "Render".
3. Find your browser. You should be able to see your document there!

Rendering a Jupyter Notebook

1. Save your document.
2. From the Terminal, use the command `quarto render filename.ipynb --to html` (eventually you may change this to something else, more on that later).
3. Find your browser. You should be able to see your document there!

HW Specifications

Your homework should:

- Be submitted as a PDF, and
- Include "publication ready" graphics–that is, the images should be saved as .svg or .eps, or as .png with high resolution (300dpi).

Rendering to PDF 1

RStudio:

In order to be able to render your document as a PDF in RStudio, you will need a LaTeX installation. If you don't already have LaTeX installed, run this line:

Jupyter and Editor do not need any additional tools to be installed.

Note that installation of a LaTeX engine allows you to use math type if you want! Use a dollar sign ($) on each side of the math type for inline symbols, or two dollar signs on each side ($$) for centered equations.

Rendering to PDF 2

- Now, change the output in your YAML header to "pdf_document" (you can also specify that you want a PDF document when you open a new file).
- Click on "Render".
- Navigate to the directory where your original .qmd file was saved. There should be a .pdf file with the same name and all of your rendered output!

Changing Image Type

This can also be changed in your YAML header with `fig-format` (options `retina`, `png`, `jpeg`, `svg`, or `pdf`).

Image Resolution

You can also change the image resolution with different options in your code chunks.

- `fig.height`, `fig.width`
- `out.height`, `out.width`
- `dpi` (default is 96).

These can also be changed in your YAML header with `fig-height`, `fig-width`, and `fig-dpi`.

Options

The easiest way to incorporate your settings into every homework is to use the following YAML header:

```yaml
title: "Homework X"
author: "Your Name"
date: "The Due Date"
format:
  pdf:
    fig-dpi: 300
```

Don't forget it goes in between the two lines with three dashes (`---`) each!

Quarto Resources

- Quarto Installation Guide
- Getting started with RStudio or Jupyter. Make sure to check out the following Computation and Authoring tutorials!
- PDF Options