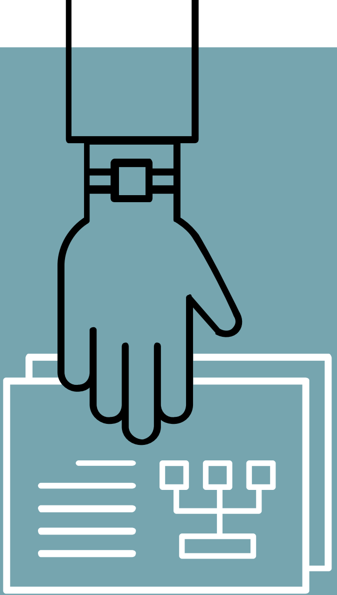
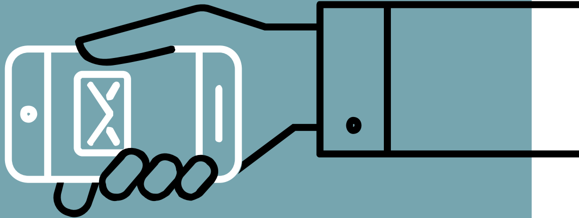


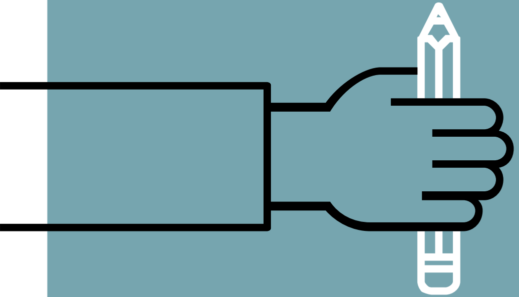


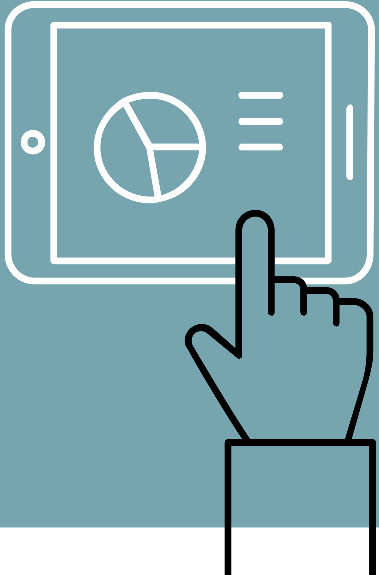
BST 270

Reproducible Data Science

Winter 2021

Session 8





Course Summary

Reproducible Research DOs:

- Start with good science
  - Garbage in, garbage out
  - Coherent, focused questions simplify many problems
  - Working with good collaborators reinforces good practices
  - Something that’s interesting to you will (hopefully) motivate good habits
- Use version control
  - Add changes in small chunks (don't just do one massive commit)
  - Track / tag snapshots; revert to old versions
  - Software like GitHub / BitBucket / SourceForge make it easy to publish results
- Teach a computer
  - If something needs to be done as part of your analysis / investigation, try to teach your computer to do it (even if you only need to do it once, like downloading a data set)
  - In order to give your computer instructions, you need to write down exactly what you mean to do and how it should be done
  - Teaching a computer almost guarantees reproducibility

Course Summary

Reproducible Research DOs:

- Keep track of your software environment
  - If you work on a complex project involving many tools / datasets, the software and computing environment can be critical for reproducing your analysis
  - Computer architecture: CPU (Intel, AMD, ARM), GPUs
  - Operating system: Windows, Mac OS, Linux / Unix
  - Software toolchain: Compilers, interpreters, command shell, programming languages (C, Perl, Python, etc.), database backends, data analysis software
  - Supporting software / infrastructure: Libraries, R packages, dependencies
  - External dependencies: Web sites, data repositories, remote databases, software repositories
  - Version numbers: Ideally, for everything (if available)

Course Summary

Reproducible Research DOs:

- Set your seed
  - Random number generators generate pseudo-random numbers based on an initial seed (usually a number or set of numbers)
    - In R you can use the set.seed()
  - Setting the seed allows for the stream of random numbers to be exactly reproducible
  - Whenever you generate random numbers for a non-trivial purpose, always set the seed
- Think about the entire pipeline
  - *Data analysis is a lengthy process; it is not just tables / figures / reports*
  - *Raw data → processed data → analysis → report*
  - *How you got the end is just as important as the end itself*
  - *The more of the data analysis pipeline you can make reproducible, the better for everyone*

Course Summary

Reproducible Research DON'Ts:

- Do things by hand
  - Editing spreadsheets of data to "clean it up"
- Removing outliers
- QA/QC
- Validating
  - Editing tables or figures (e.g. rounding, formatting)
  - Downloading data from a web site (clicking links in a web browser)
  - Moving data around your computer; splitting/reformatting data files
  - "We're just going to do this once …"
  - Things done by hand need to be precisely documented, and this is much harder than it sounds

Course Summary

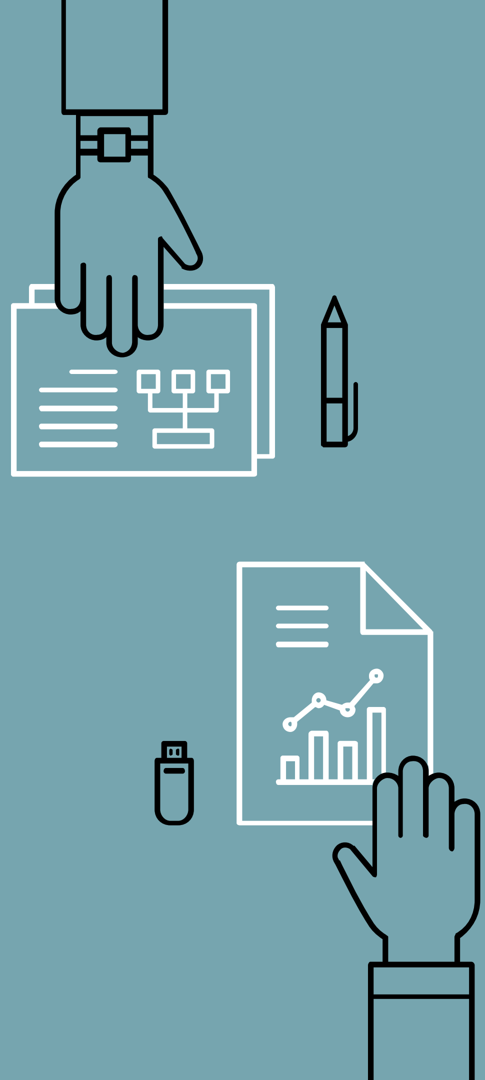
Reproducible Research DON'Ts:

- Point and click
  - Many data processing / statistical analysis packages have graphical user interfaces (GUIs)
  - GUIs are convenient / intuitive but the actions you take with a GUI can be difficult for others to reproduce
  - Some GUIs produce a log file or script which includes equivalent commands; these can be saved for later examination
  - In general, be careful with data analysis software that is highly interactive; ease of use can sometimes lead to non-reproducible analyses
  - Other interactive software, such as text editors, are usually fine

Course Summary

Reproducible Research DON’Ts:

- Save output
  - Avoid saving data analysis output (tables, figures, summaries, processed data, etc.), except perhaps temporarily for efficiency purposes.
  - If a stray output file cannot be easily connected with the means by which it was created, then it is not reproducible.
  - Save the data and code that generated the output, rather than the output itself
  - Intermediate files are okay as long as there is clear documentation of how they were created



Module 6 Part 3 Discussion

With so many platforms built for various types of sharing and collaborating, how do we decide which ones to use, especially when collaborators in non-computational fields may have no idea about any of these platforms?
It usually depends on what your team/PI/collaborators prefer or already work with. If you have the freedom to choose then it's up to you and my advice is to use whatever is easiest for you, especially if you don't have the time to learn how to use something a bit more complex. Just be sure to thoroughly document everything and showcase progress in a way that is accessible to your non-computational collaborators (via Google slides or a pdf or something similar)

We have already learned from the video that Docker could be a bit more complex for everybody to work with that, and Simon said that Code Ocean would make Docker more accessible for us. I wonder if Code Ocean would have all the functions we hope to have (version control, environmental setup, etc.) and be easy enough to use (at least much easier than Docker) for researchers with insufficient experience playing with it.
Yes, CodeOcean does have version control and does make environment setup a lot easier than just working with Docker. I haven't used it yet but it does sound a lot easier to use than a lot of other tools out there.

Module 6 Part 3 Discussion

It seems like Code Ocean is just a way for researchers using different environments to be able to run code under the same environment. Is there ever a problem when the original researcher's work requires interfacing with proprietary software that other researchers may not have access to?
There can be. Sometimes it's possible to get access to a computing cluster that has the proprietary software available through it.

I have never heard of Code Ocean before this class. I am a bit confused if it is supposed to be a replacement for GitHub or a replacement for a platform such as Jupyter notebooks? Or a combination of both?

It isn't meant to be a replacement. See my comments below about the size of a project and the number of collaborators.

Module 6 Part 3 Discussion

As data gets bigger and higher quality, should we ever be concerned about running out of cloud/physical storage space in the future?
This is a tough question because I don’t know the current state-of-the-art capabilities of storage technology. I think we’ll be fine for our lifetime because of the rapid progress we’ve made in the last decade or so. But I’m not sure if this will change in the future.

On the other hand, we know that there are advantages for keeping things simple and elegant. For example, I believe that Jupyter Notebook and/or R Markdown could be good enough in terms of reproducibility in many scenarios (e.g., reproducing statistical simulation research for biostatistical methodology papers). I wonder if we have a checklist (or a series of closed-form criteria) to determine whether we need to use further tools, such as Docker or Code Ocean, to guarantee reproducibility.
See my other comments about the size/number of collaborator on a project. I don’t know of a checklist, but the bigger the project and the more collaborators writing code, the higher the probability of using Docker or Code Ocean.

Module 6 Part 3 Discussion

Is Code Ocean currently the only tool that allows you to reproduce code on a specified computing environment, or are there other options? Problems arising from different computing environments seems to be a common issue, so I'm wondering if there are multiple alternatives to fix this issue.'
There are other options (like Docker or cloud computing platforms) but Code Ocean is the most user friendly (that I know of).

In addition, though it was mentioned that Code Ocean is now partnering with IEEE, is it commonly used among biostats researchers at the moment?
Not that I know of, but I haven't talked to every faculty member. I know JP Onnela uses AWS and I think JQ uses GCP. Curtis might use Code Ocean but I'm not sure.

Does reproducibility become harder with more and more different programming languages and softwares being published? Since there is no universal language, are there any ways to make reproducibility easier if someone uses a more obscure language, program, or software?
Yes and no. Yes, it is more difficult in the sense that there are more languages and software choices and thus more to keep track of. But no in the sense that now there are more tools to help with reproducibility - like R Markdown and Notebook, Overleaf, Jupyter Notebook, Code Ocean, GitHub, etc. As long as the documentation is really good, even using obscure languages or software should be reproducible.

Module 6 Part 3 Discussion

Is it possible to see an example of a well-composed project in Github (it could be yours or someone else's) with proper documentation?
Here is one from a team of students in my intro course last semester.

Do you recommend using Code Ocean instead of Github?
For small projects or projects where you are the only one writing code, I recommend GitHub. If you are working on a bigger code with multiple collaborators then I suggest Code Ocean.

Simon Adar mentioned the importance of giving the user the package dependencies they may need so they don't go down a rabbit hole of downloading libraries/packages. Is there an easy way of finding R/python library dependencies that are needed?
For R you can use the **available.package** function. Example input:

```
pack <- available.packages()
pack["ggplot2","Depends"]
```

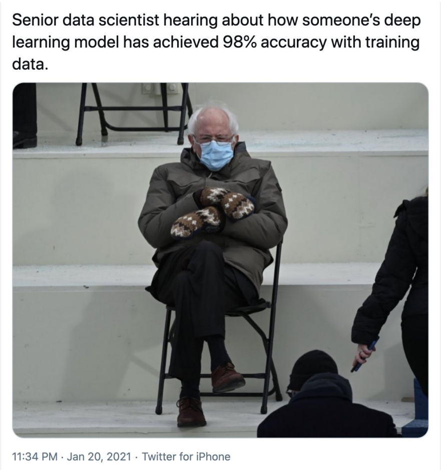
Output:

```
 [1] "R (>= 2.14), stats, methods"
```

For python: `pip show [package name`
Check out this post as well.

Module 6 Part 3 Discussion

Can you explain why this is funny?



Senior data scientist hearing about how someone's deep learning model has achieved 98% accuracy with training data.

11:34 PM · Jan 20, 2021 · Twitter for iPhone

If you test your DL model on data it was trained with, it has already seen the data and knows how to make good predictions because the model was tweaked to perform well on the training data. So stating this performance metric is meaningless and shows the person doesn't know what they are doing. They should be using a test set to gauge performance. PS - thanks for this, I'm going to use it in my slides for BST 261 :D

Module 6 Part 3 Discussion

If we take away just one thing from the course, what should it be?

Document everything and think about the reproducibility of your project at every single stage from beginning to end.

Should we be coding in R without using R Markdown?

It’s really up to you. In my opinion R Markdown makes it much easier to track your progress and have your work be reproducible. In the end, when you publish your code, having just an R script (plus other documentation) may be better (in terms of being faster) for someone to reproduce your results.

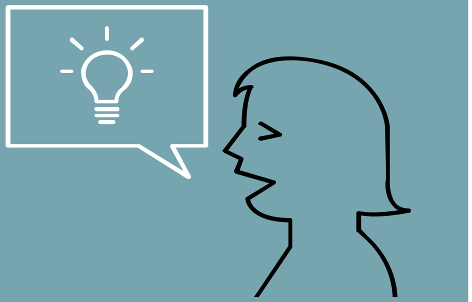
What are situations during which we would use Make? It seems like all of the work being done in Python could also be done in R so I am not sure why Make is useful?

It’s really just a preference. You do need to use other packages in R Markdown or Jupyter Notebook in order to code in both languages in one of the files, so this may be easier for some to do while others may find using Make is easier for them.

Module 6 Part 3 Discussion

This is a question about the course overall: for some really complicated papers, it may be impossible to reproduce the work just from the methods section without any additional guidance from the authors, is that in itself an issue of reproducibility or an issue of oversimplifying the methods explained?

It depends. If the methods are oversimplified in the paper because the journal has a page limit or asked the authors to cut out some details, but the authors provide more documentation somewhere else, then it's just an oversimplification of results. If the authors don't provide more details elsewhere, then it's an issue of reproducibility.



Individual Project

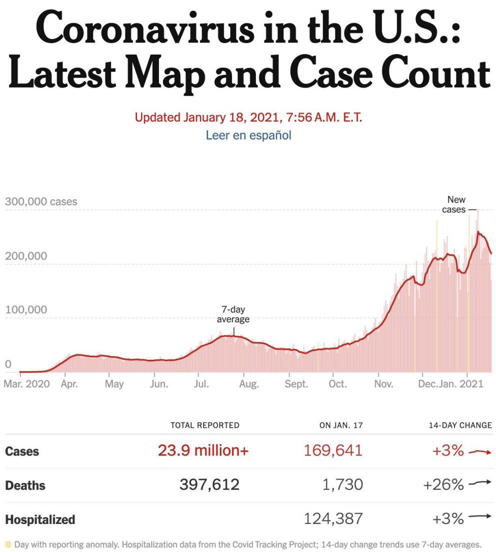
Reproducing a NYT Post

Data visualization is an incredibly powerful tool that can affect health policy decisions. Ensuring they are easy to interpret, and more importantly, showcase accurate insights from data is paramount for scientific transparency and the health of individuals. For this assignment you are tasked with reproducing COVID-19 visualizations and tables published by the New York Times.

Specifically, you will attempt to reproduce the following for January 17th, 2021:

1. New cases as a function of time with a rolling average plot - the first plot on the page (you don't need to recreate the colors or theme)

2. Table of cases, hospitalizations and deaths - the first table on the page



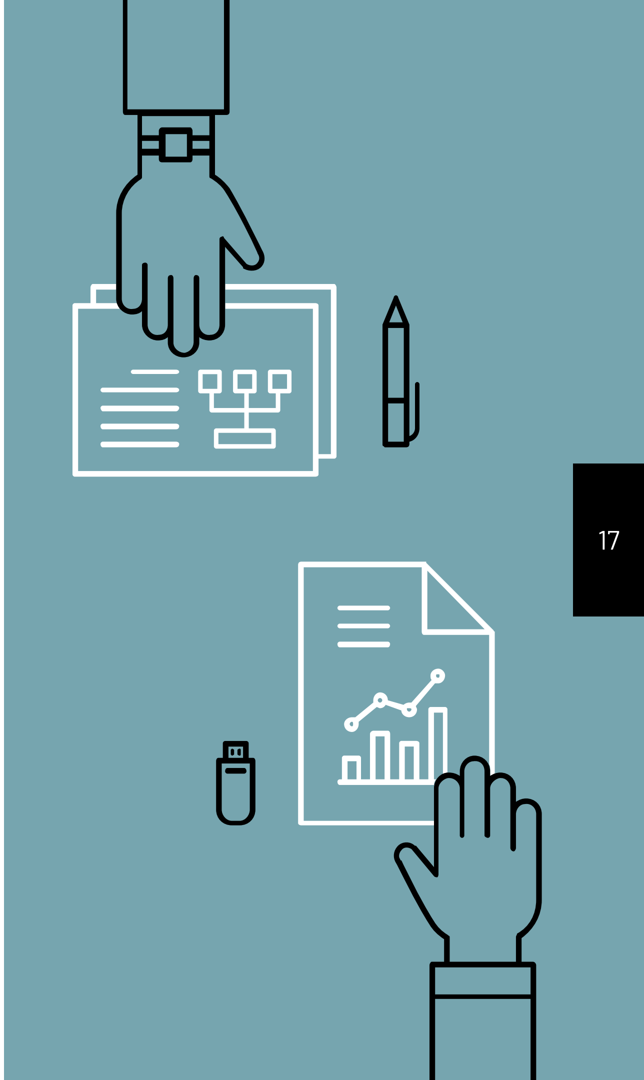
Coronavirus in the U.S.: Latest Map and Case Count

Updated January 18, 2021, 7:56 A.M. E.T.

Leer en español

	TOTAL REPORTED	ON JAN. 17	14-DAY CHANGE
Cases	23.9 million+	169,641	+3% →
Deaths	397,612	1,730	+26% →
Hospitalized		124,387	+3% →

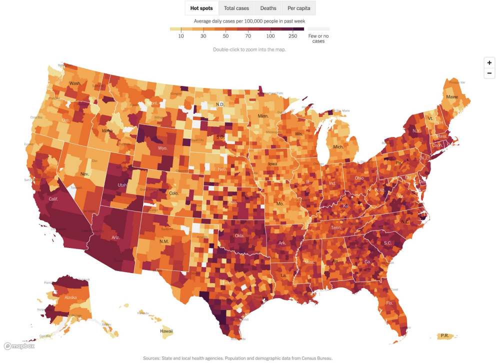
▨ Day with reporting anomaly. Hospitalization data from the Covid Tracking Project; 14-day change trends use 7-day averages.



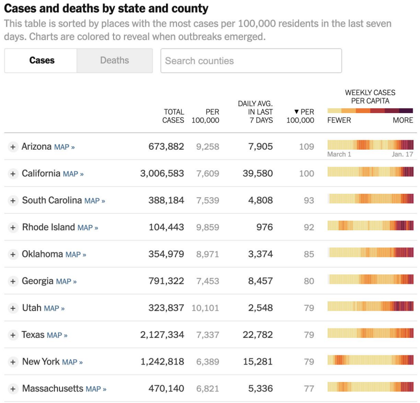
Reproducing a NYT Post

3. (Optional) The county-level map for previous week ('Hot spots') - the second plot on the page (only the 'Hot Spots' plot)

4. Table of cases by state - the second table on the page (do not need to include per 100,000 or per capita columns)



Sources: State and local health agencies. Population and demographic data from Census Bureau.



Cases and deaths by state and county

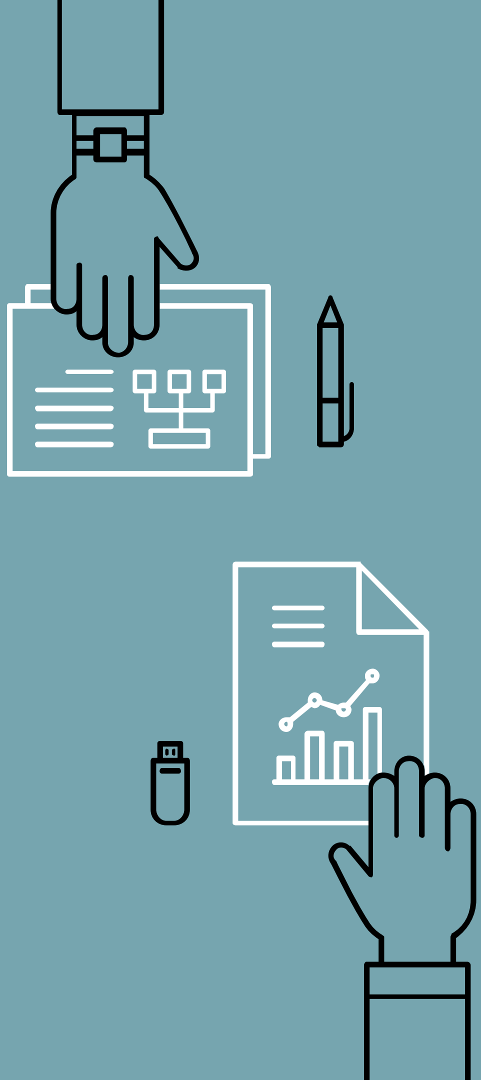
This table is sorted by places with the most cases per 100,000 residents in the last seven days. Charts are colored to reveal when outbreaks emerged.

	TOTAL CASES	PER 100,000	DAILY AVG. IN LAST 7 DAYS	▼ PER 100,000	WEEKLY CASES PER CAPITA FEWER MORE
+ Arizona MAP »	673,882	9,258	7,905	109	March 1 Jan. 17
+ California MAP »	3,006,583	7,609	39,580	100	
+ South Carolina MAP »	388,184	7,539	4,808	93	
+ Rhode Island MAP »	104,443	9,859	976	92	
+ Oklahoma MAP »	354,979	8,971	3,374	85	
+ Georgia MAP »	791,322	7,453	8,457	80	
+ Utah MAP »	323,837	10,101	2,548	79	
+ Texas MAP »	2,127,334	7,337	22,782	79	
+ New York MAP »	1,242,818	6,389	15,281	79	
+ Massachusetts MAP »	470,140	6,821	5,336	77	

Reproducing a NYT Post

Notes:

- You can download the files here. You will need to create a repository (with README file) here.
- You don't need to make the plots look exactly the same as the post - just showcase the most important information
- You don't need to make the tables look pretty, but you do need to print out at least some of the rows to check the numbers against the NYT post
- Due: Monday, January 25th by 11:59pm EST
- You will have class time today to work on it.
    - You don't have to stay on Zoom and can work on it at any time, but Matt and I will stay on Zoom until 11am if you have any questions or need help



Homework

- Submit individual project
- Submit course evaluation