Motivation: Version Control with Git as a Learning Objective in Statistics Courses

Matthew Beckman
Penn State University

August 4, 2020
JSM Virtual Conference

Reproducibility:

- completely self-contained including. . .
  - source data
  - code book
  - all data wrangling/prep steps
  - recreate all analysis, models, visuals
  - final reporting
- easy to verify results or refresh if source data updates
- e.g., all code "just works" with no changes needed

Version control

- maintains the evolution of the project
- safely explore alternative solutions/ideas in parallel

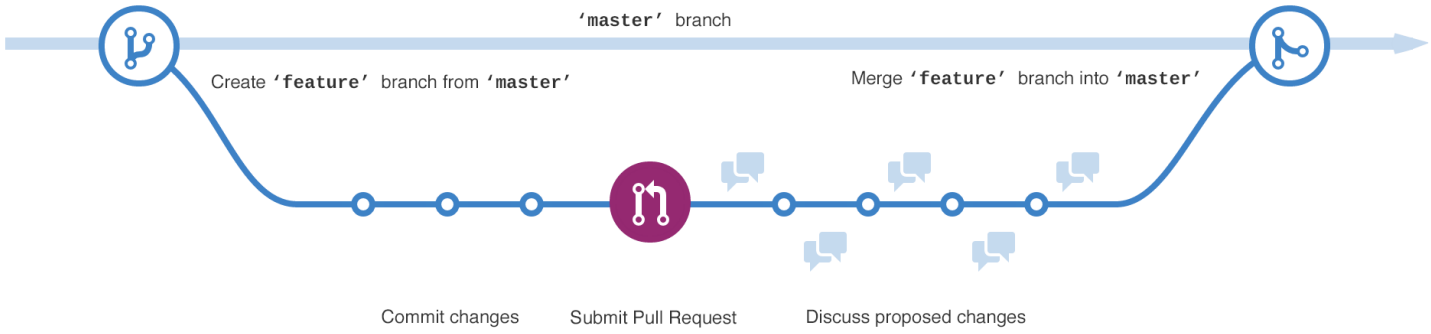

Figure 1: exploring parallel solutions
(https://guides.github.com/activities/hello-world/)

Version control

- collaboration among users
- self-collaboration–e.g., RStudio Desktop and RStudio Server

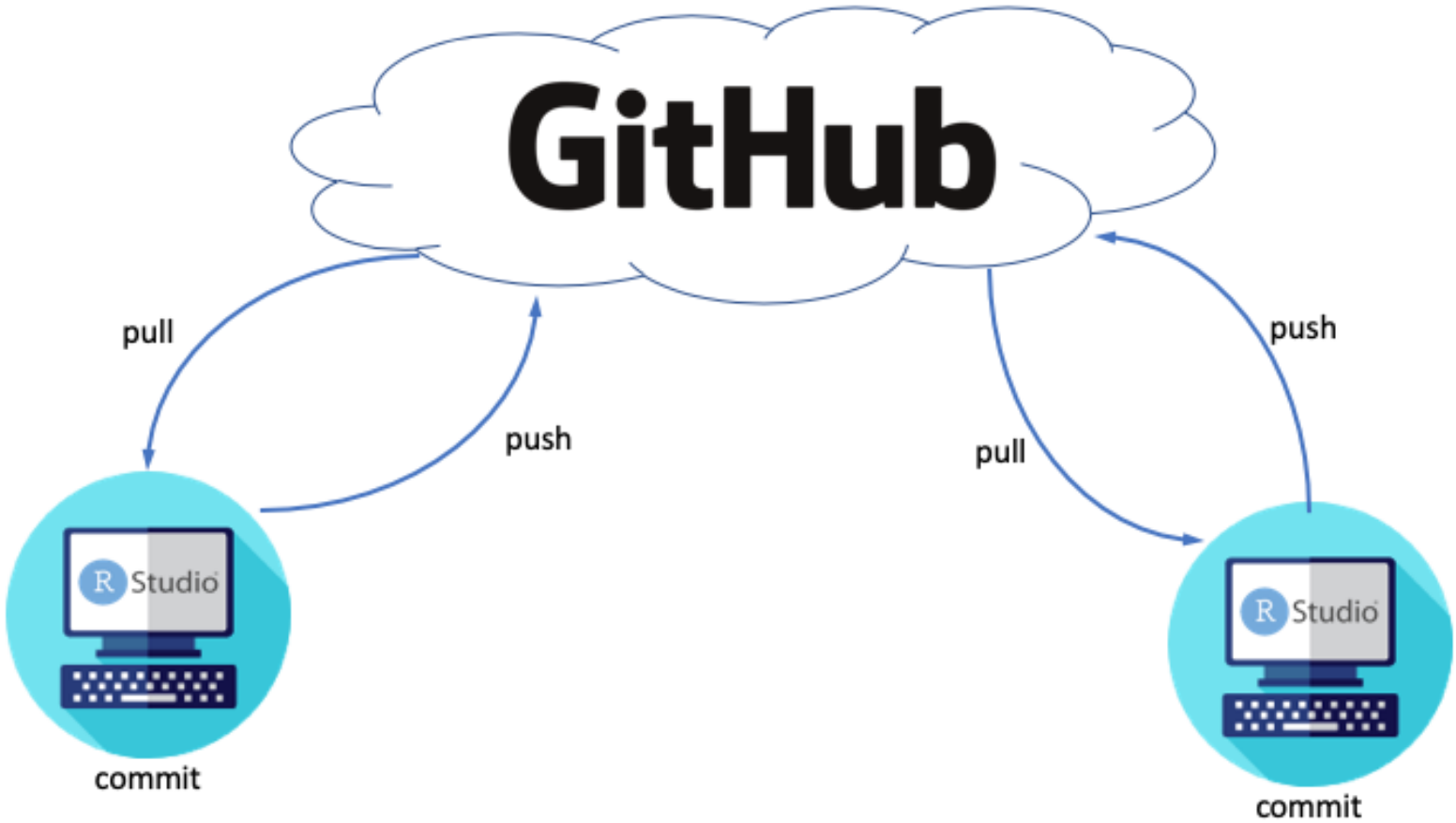

Figure 2: Collaboration schematic

Reproducibility $\neq$ Version Control

- Sometimes lumped together as if they're one in the same, and it's tempting to speak of Git(Hub) as a panacea...
- They aren't and it isn't...

Our motivation: invest in good habits with a professional workflow designed to streamline **both** virtues.

Ethical practice

- Any analysis may require hundreds of tiny decisions
- These decisions may necessarily be handled by a single person
- Work products are often intended for audience without technical expertise to scrutinize those decisions

With reproducibility & version control

- all decisions are documented
- all results can be checked
- proper scrutiny is possible (now or in future)

Industry & Academic Preparedness

```
- programming is a collaborative sport
- effective entry point for research participation
```

- Industry Preparedness
  - programming is a collaborative sport
  - quite common to refresh standard reports
- 2014 ASA Curriculum Guidelines push for reproducibility
- CS education calls for VC in the curriculum (Haaranen & Lehtinen, 2015; Zagalsky et al., 2015)
- 2017 Kaggle Study
  - almost 17,000 responses
  - over a third reported using a VC tool
  - 58% of those using VC use Git

Slide 1

- item 1
- item 2

Box

1. item
2. item

Slide 2

- Cite stuff people have already done[1]
- Link to URL's: https://en.wikipedia.org/wiki/URL

[1] American Statistical Association Undergraduate Guidelines Workgroup (2014). *Curriculum guidelines for undergraduate programs in statistical science.*

Full slide image

data. Such skills underpin strategies for assessing and ensuring data quality as part of data preparation and are a necessary precursor to many analyses[43].

- Use of one or more professional statistical software environments[44]
- Data management using software in a well-documented and reproducible way[45], data processing in different formats, and methods for addressing missing data
- Basic programming concepts (e.g., breaking a problem into modular pieces, algorithmic thinking[46], structured programming[47], debugging, and efficiency)
- Computationally intensive statistical methods (e.g., iterative methods, optimization, resampling, and simulation/Monte Carlo methods)[48]
- Use of multiple data tools[49], so graduates are not wedded to one and are better able to learn new technologies[50]

Mathematical Foundations

The study of mathematics lays the foundation for statistical theory. Undergraduate statistics majors should have a firm understanding of why and when statistical methods work. They should be able to communicate in the language of mathematics and explain the interplay between mathematical derivations and statistical applications.

- Calculus (e.g., integration and differentiation)[51]
- Linear algebra (e.g., matrix manipulations, linear transformations, projections in Euclidean space, eigenvalues/eigenvectors, and matrix decompositions)
- Probability (e.g., properties of univariate and multivariate random variables, discrete and continuous distributions)[52]
- Emphasis on connections between concepts in these mathematical foundations courses and their applications in statistics[53]

Statistical Practice

Strong communication skills complement technical knowledge and are particularly necessary for statisticians; graduates need technical skills to perform analyses and communication skills to understand clients' needs and then effectively discuss results and conclusions. Important practical skills include the following:

43 See Zhu et al. (2013) "Data acquisition and pre-processing in studies on humans: What is not taught in statistics classes," *The American Statistician*, 67(4):235–241, which includes a series of skills: (1) get to know the study; (2) assess the validity of variable coding; (3) assess data entry accuracy; (4) perform data cleaning; and (5) edit identified data errors.

44 Although we acknowledge that Microsoft Excel is a common platform for data exchange, we do not recommend it as a primary analysis environment.

45 Appropriate environments could include R, Python, and SAS, complemented by tools including shell scripts and knitr.

46 Futschek (2006) defines algorithmic thinking as a set of abilities related to constructing and understanding algorithms: (1) the ability to analyze a given problem; (2) the ability to precisely specify a problem; (3) the ability to find the basic actions that are adequate to the given problem; (4) the ability to construct a correct algorithm to a given problem using basic actions; (5) the ability to think about all possible special and normal cases of a problem; and (6) the ability to improve the efficiency of an algorithm. Futschek, G. (2006). "Algorithmic thinking: The key for understanding computer science," in R. Mittermeir (Ed.), *Informatics Education–The Bridge Between Using and Understanding Computers* (Vol. 4226, pp. 159–168). Berlin/Heidelberg: Springer. We consider this to be a necessary, but not sufficient component of "computational thinking."

47 We define structured programming as the ability to use functions and control structures (e.g., "for" loops).

48 This recommendation is consistent with the efforts of Conrad Wolfram and the Computer-Based Math initiative, *www.computerbasedmath.org* and *www.tinyurl.com/ted-wolfram*. The incorporation of these tools may be particularly valuable at the bachelor's level, since students will generally have less technical knowledge (and need to be able to simulate to generate insights and/or check analytic results).

49 Students should develop the capacity to manipulate formats such as CSV, JSON (JavaScript Object Notation, a data-interchange format that is easy to read, parse, and generate; see Nolan and Temple Lang (2014) *XML and Web Technologies for Data Sciences with R*), XML, databases (see, for example, Ripley (2001) "Using databases with R," *R News*, 1(1):18–20 and Wickham (2011) "ASA 2009 Data Expo," *Journal of Computational and Graphical Statistics*, 20(2):281–283), and text data. Because many faculty were not trained in these technologies, continuing education in this area needs to be made a priority.

50 We are not prescriptive regarding which technologies are incorporated into the curriculum, as long as they are sufficiently flexible and powerful. Many undergraduate statistics students develop expertise in environments such as R/RStudio, Python, and SAS.

51 Multivariate calculus is recommended.

52 Markov chains are a useful topic for undergraduate majors in statistics.

53 This linkage includes topics such as the delta method. In addition, many students might benefit from exposure to modeling and simulation in their mathematics courses as a way to reinforce their computational skills.

Here's a table

Col1	Col2
1	One
2	Two
3	Three
4	Four
5	Five
6	Six

Acknowledgments

- So many to thank

References

1. American Statistical Association Undergraduate Guidelines Workgroup (2014). 2014 Curriculum guidelines for undergraduate programs in statistical science. Alexandria, VA: American Statistical Association. http://www.amstat.org/education/curriculumguidelines.cfm
2. Beckman, M. D., Cetinkaya-Rundel, M., Horton, N., Rundel C., Sullivan A. J., & Tackett, M. (in review). Implementing version control with Git as a learning objective in statistics courses. Preprint URL: https://arxiv.org/pdf/2001.01988.pdf
3. Haaranen, L. & Lehtinen, T. (2015). Teaching git on the side: Version control system as a course platform, in *Proceedings of the 2015 ACM Conference on Innovation and Technology in Computer Science Education*, ITiCSE '15, ACM, New York, NY, USA, pp. 87–92. URL: http://doi.acm.org/10.1145/2729094.2742608
4. Kaggle (2017). Kaggle machine learning & data science survey 2017. URL: https://www.kaggle.com/kaggle/kaggle-survey-2017
5. Zagalsky, A., Feliciano, J., Storey, M.-A., Zhao, Y. & Wang, W. (2015). The emergence of GitHub as a collaborative platform for education, in *Proceedings of the 18th ACM Conference on Computer Supported Cooperative Work & Social Computing*, CSCW '15, ACM, New York, NY, USA, pp. 1906–1917. URL: http://doi.acm.org/10.1145/2675133.2675284

Q & A

Motivation: Version Control with Git as a Learning Objective in Statistics Courses

Matthew Beckman
Penn State University

August 4, 2020
JSM Virtual Conference

https://mdbeckman.github.io/JSM2020-Virtual/

Backup slide