

Does Our Theory Work In Practice? Advising Undergraduate Researchers on Statistical Simulation Studies

Jerzy Wieczorek
Colby College
jerzy.wieczorek@colby.edu

August 4, 2026
Joint Statistical Meetings
Boston, Massachusetts

Today's talk

- ▶ Pros and cons of statistical simulation studies with undergraduate research assistants
- ▶ Case studies
- ▶ Lessons learned

Context: Statistics at Colby College



- ▶ Liberal arts college of around 2,400 students in Waterville, in central Maine
- ▶ First US liberal arts college to have its own standalone Department of Statistics (2021)
- ▶ Added a Data Science major (2023), jointly administered by Statistics, Mathematics, and Computer Science departments
- ▶ Around 250 students take our Intro Stats course each year
- ▶ Around 10-15 Statistics majors and 10-15 Data Science majors in each class year
- ▶ Around 5 full-time, tenure-track/tenured Statistics faculty with backgrounds including biostatistics, epidemiology, machine learning, and survey statistics

What do I mean by “statistical simulation studies”?

- ▶ Students try to simulate their own datasets that would illustrate the effectiveness (or deficiencies) of our proposed new statistical methods
 - ▶ Assess a method's statistical properties: bias, variance, MSE, CI coverage, etc.
 - ▶ Compare the timing of different algorithms on equivalent datasets
 - ▶ Debug code by making sure it runs for a variety of data values and structures
- ▶ Students learn about statistical practice, conceptual theory, and hands-on programming
- ▶ Also a useful skill for their own classes, e.g. checking your MathStats proof by simulating data in R

Benefits

- ▶ Simulations can help to convince ourselves, referees, and journal readers that our methods work as advertised
- ▶ Students learn how to understand a new proposed method well enough to judge its effectiveness
- ▶ Students internalize the meaning of Frequentist metrics “under repeated sampling” (bias, CI coverage, etc.)
- ▶ Coding up a complex simulation requires students to think in new ways about data distributions, beyond usual coursework
- ▶ Students learn to use GitHub and best practices for reproducible computing, version control, bug tracking, and coordination among multiple programmers
- ▶ Simulated datasets don't require funding and don't run into privacy/confidentiality disclosure concerns

Challenges

- ▶ Installing Git & teaching GitHub can be tricky (Jenny Bryan's "Happy Git With R" has been a lifesaver)
- ▶ Students often don't know where to start, and I've found it hard to teach explicitly – you have to build up some intuition for it
- ▶ Our curriculum has historically pushed Probability & MathStats to senior year, so earlier students aren't fluent with some standard distributions until late in their studies
- ▶ Simulating new data isn't covered in most of our Stats courses

Case study 1: CIPerm with Tupaj



Emily Tupaj '21

“Does this software implementation work the way we’d expect?”

Case study 1: the statistical problem

Consider testing for a difference in 2 means: $\mu_Y - \mu_X = \Delta$

- ▶ You can run a permutation test of $H_0 : \Delta = 0$ with one set of permutations
- ▶ To get a permutation confidence interval, the simple aka “naive” approach is: run the permutation test many times at many different null Δ values

In an unpublished 2009 paper, my grad school classmate Minh Nguyen and advisor Mara Tableman came up with a clever way to get CIs from **just one** set of permutations. This is much faster, especially for large datasets!

Nguyen's proof was clear, but his code had not been published.

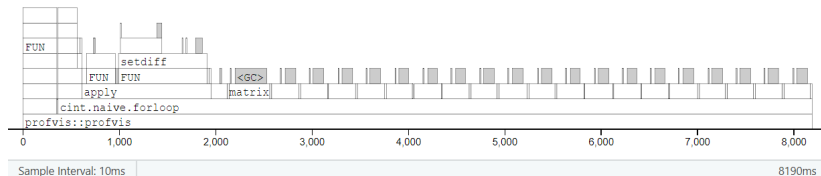
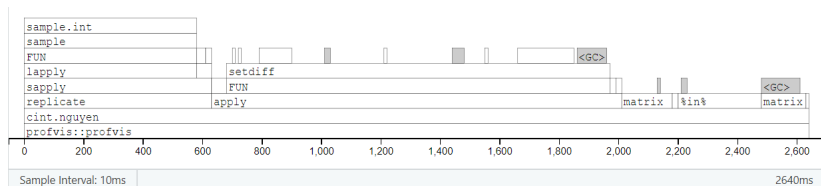
Does it work as advertised?

Case study 1: Tupaj's contributions

Emily Tupaj '21 helped me clean up and test Nguyen's R code:

- ▶ Double-check Nguyen's small examples by hand
- ▶ Implement, test, and debug the R code from Nguyen's manuscript; document it in reproducible RMarkdown docs
- ▶ Also implement "naive" permutation CIs to validate results of Nguyen's method
- ▶ Simulate larger datasets; validate that the naive method and Nguyen method give same results; & record repeated run-times to show Nguyen's method is substantially faster
- ▶ Make a R package, `CIPerm`, to disseminate this code (with Nguyen's approval)

Case study 1: example of results



Case study 1: lessons learned

- ▶ Undergrads can make genuine research contributions by cleaning up “completed” yet unpublished work:
 - ▶ Turning pseudocode into runnable R functions
 - ▶ Making them public on GitHub and/or via R packages
 - ▶ Providing reproducible and public RMarkdown docs and/or R package vignettes
- ▶ Timing/benchmarking studies can be a good intro to sim studies, as details of how data are simulated can be less crucial

See Tupaj's work at:

<https://github.com/ColbyStatSvyRsch/CIPerm>

and:

Tupaj and Wieczorek (2022),

“CIPerm: Computationally-Efficient Confidence Intervals for Mean Shift from Permutation Methods,”

R package version 0.2.3, *CRAN*

Case study 2: surveyCV with Guerin and McMahon



Cole Guerin '21



Thomas McMahon '21

“Does my theoretical method work the way we’d expect?”

Case study 2: the statistical problem

Consider cross-validation (CV) for model selection

- ▶ Standard CV assumes the data were sampled iid. You partition the data into folds at random, hold out one fold, train a model on the others, and test it on the held-out fold. Repeat for each fold.
- ▶ But what if the data were collected through stratified sampling? Or cluster sampling? Or with different probabilities, such as probability proportional to size (PPS) sampling?

I worked on ways to think about what CV is doing, without assuming iid sampling; and ways to create CV folds and CV estimates of test performance under complex sampling designs.

Would my methods work as expected?

Unlike in Tupaj's project, here we didn't have pre-existing code and examples to test.

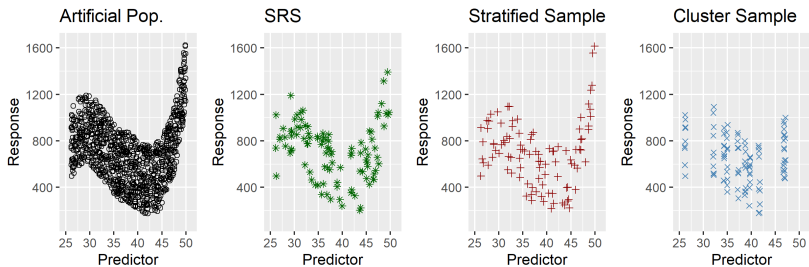
Case study 2: Guerin and McMahon's contributions

Cole Guerin '21 & Thomas McMahon '21 helped me validate my methods and make my code accessible:

- ▶ Find real survey datasets with various complex sampling designs, and use them to test out our ideas
- ▶ Simulate data to highlight the value of our methods:
 - ▶ Devise a finite population with a nonlinear trend
 - ▶ Write R code to sample from that pop in different ways (SRS, stratified, clustered, PPS)
 - ▶ Compare CV MSEs and model selection outcomes across repeated samples, sampling designs, and CV methods
- ▶ Make a R package, `surveyCV`, to disseminate our methods
 - ▶ Include the sims as vignettes
 - ▶ Allow use of R objects from popular survey package
- ▶ Collaborate on a publication about our methods & results

Case study 2: example of results

Simulated Data and Examples of How It Was Sampled



Case study 2: lessons learned

- ▶ Simulating new data checks students' understanding:
 - ▶ Can you generate new data that follows certain assumptions?
 - ▶ Do you understand what our theory predicts should happen?
 - ▶ Combine these skills to simulate extreme examples, where the method should work especially well or especially poorly
- ▶ It can also help convince reviewers, readers, and R package users that the methods work as advertised
 - ▶ Unlike most real-data studies, a sim study can show a method's performance in a situation where "truth" is known
- ▶ Teaching a sampling course let me recruit prepared students

See their work at:

<https://github.com/ColbyStatSvyRsch/surveyCV>

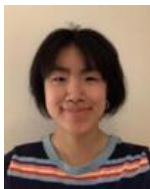
and:

Wieczorek, Guerin, and McMahon (2022),
"K-fold cross-validation for complex sample surveys,"
Stat, 11 (1)

Case study 3: kbaabb with Cody, Tan, Chistolini, and Rui



Zach Cody '23



Emily Tan '23



Jackie Chistolini '24J



Leonor Rui '24

“How can we convince others to adopt our sim-study approach?”

Case study 3: the statistical problem

Consider the use of simulation studies to choose models for a specific applied problem

- ▶ Simulating abstract data, e.g. $\epsilon \sim N(0, 1)$, may be unrealistic for your particular applied problem/context
- ▶ Repeated sampling from an artificial population may be better, **if** we can justify why the artificial population resembles the real one in relevant ways

We developed a new way to create realistic artificial populations for comparing Small Area Estimation (SAE) models, suitable for the structure of US Forest Service data.

What demos might help convince forestry researchers to adopt our method?

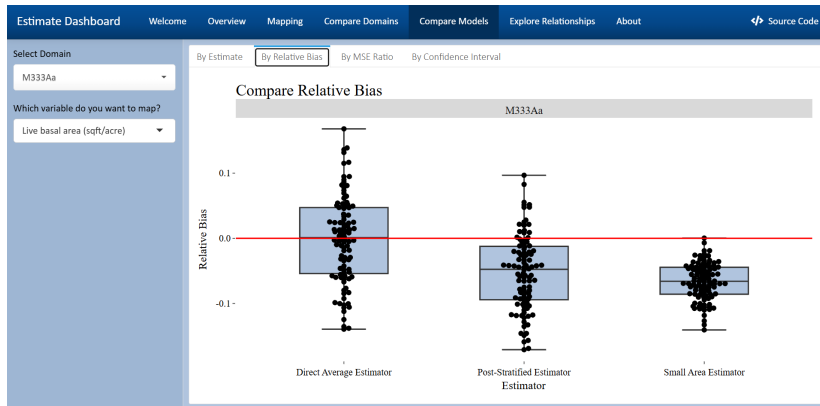
This time, sim studies were not a result, but the method itself.

Case study 3: Cody, Tan, Chistolini, & Rui's contributions

Cody '23, Tan '23, Chistolini '24J, & Rui '24 helped me demonstrate, document, and visualize the process of using our sim-study approach – making this more of a case study:

- ▶ Using repeated samples from my artificial population, try out a variety of models: EDA, model development, model fitting, and performance metrics (bias, MSE, CI coverage. . .)
- ▶ Compare results to the artificial population's “truth” to improve individual models or choose a “best” one
- ▶ Document this process in reproducible RMarkdown scripts, to help forestry researchers see how it works
- ▶ Develop a R Shiny app to visualize the simulated datasets & model performance metrics, showing how the results vary by model, by metric, or by subdomain

Case study 3: example of results



Case study 3: lessons learned

- ▶ Small Area Estimation was an extra challenge for the students: not covered in our classes, few intro-level resources
- ▶ But by learning to develop and document a good analysis process, students internalized best-practices themselves, sharpening Stat Modeling & Machine Learning course skills
- ▶ Students with data visualization and cartography experience made a better Shiny app than I could!

See their work at:

<https://github.com/ColbyStatSvyRsch/FIA-simpop-app>

and:

White, Wieczorek, et al. (2026),

“A method for empirically assessing small area estimators via bootstrap-weighted k-nearest-neighbor artificial populations, with applications to forest inventory,”

Forestry, 99 (2)

Case study 4: surveyConformal with Day



Andy Day '25

“It was obvious how to implement it, until we actually tried!”

Case study 4: the statistical problem

Consider conformal prediction, a general-purpose approach to prediction intervals (PIs)

- ▶ Standard conformal inference methods assume that the data were sampled iid.
- ▶ As with the surveyCV case study: what if sampling was stratified, clustered, PPS, etc.?

I worked on extending conformal prediction methods to complex survey sampling designs in a design-based, finite-population setting.

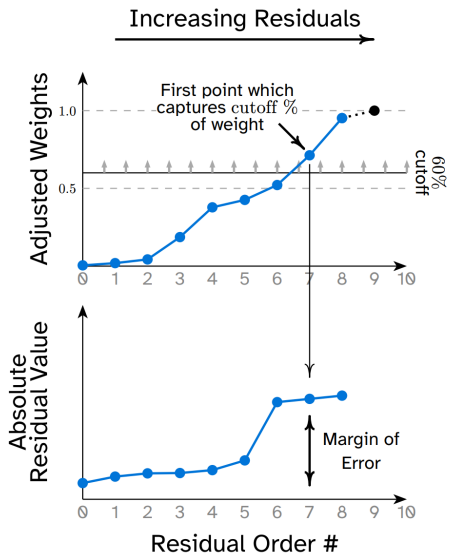
Was it clear how to implement my methods for each of the most common designs?

Case study 4: Day's contributions

Andy Day '25 helped me implement broader versions of my methods and find gaps in the theory:

- ▶ Rewrite my one-off analysis R code into general-purpose, efficient C++/Rcpp
- ▶ Simulate new data with different structures than the specific dataset used as a case study in my paper
- ▶ Run simulation studies to confirm PI coverage... or to notice possible gaps in the methods
 - ▶ Found that although cluster-sampling method works alone, and PPS method works alone, their combo does not work for sampling with clustering *and* weights

Case study 4: example of results



Case study 4: lessons learned

- ▶ By going beyond the data example in my paper, to generating more general datasets, we found gaps that require new theory before new code
- ▶ Student working in C++/Rcpp left well-documented codebase, but hard for me to maintain as a C++ novice
- ▶ Use of GitHub and version control helped me review where old versions of the code had needed improvements
- ▶ Setting GitHub repos to private vs. public, we can refine work-in-progress without sharing buggy code with the world

For further details, see:

Wieczorek (2023),

“Design-based conformal prediction,”

Survey Methodology, 49 (2)

Thank you!

Please reach out if you are interested in these case studies,
or any statistical or pedagogical collaborations!

Contact: jerzy.wieczorek@colby.edu