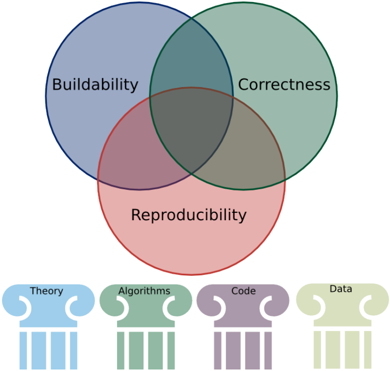
Reproducibility, Correctness, and Buildability: the Three Principles for Ethical Public Dissemination of Computer Science and Engineering Research

Kristin Y. Rozier
NASA Ames Research Center

Eric W.D. Rozier
University of Chicago/University of Miami

ETHICS•2014

Principles for Ethical Public Dissemination



Reproducibility: capability to reproduce fundamental results from released details

Correctness: ability of an independent reviewer to verify and validate the results of a paper

Buildability: ability of other researchers to use the published research as a foundation for their own new work, including utility, usability, extensibility, suitability as a foundation

Motivation

Publication is about helping the advancement of humankind

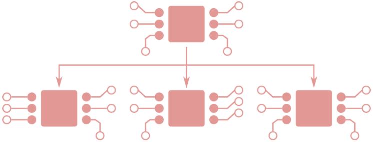
Papers presenting potentially useful novel ideas regularly appear without a comparison to the state of the art, without appropriate benchmarks, without any mention of limitations, and without sufficient detail to reproduce the experiments. This hampers scientific progress and perpetuates the cycle.[1]

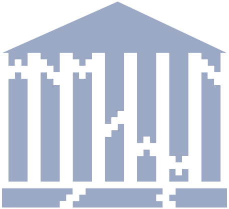
Often authors are made to prove that their work is novel or interesting, rather than making them prove that their findings are true.

[1] J. Vitek and T. Kalibera, Repeatability, reproducibility, and rigor in systems research, EMSOFT 11.

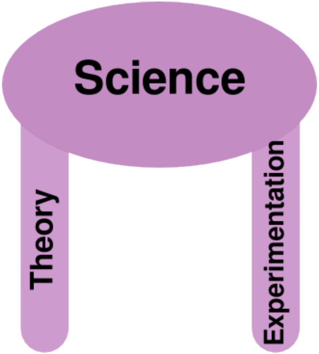
Thesis Statement

Given that it is possible for all publicly disseminated research to uphold these three principles and deviation from them is determined by factors that restrict public dissemination, such as intellectual property, secure/classified information, and time, it is unethical to publicly disseminate work that does not adhere to at least one of the principles of reproducibility, correctness, and buildability.





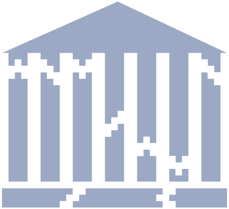
Theory: One of the Two Legs of Science



Reproducibility & Correctness:

- State all theorems explicitly, in main text
- Previously published theorems: cite source in theorem statement
- Full Proofs! or at least Proof arguments!
- Evidence of problem applicability
- Enumerate all assumptions!
- Automated theorem provers!
- Accompanying implementation with rigorous experimental evaluation

Theory as a Foundation



Buildability:

- ~~“The proof is obvious.”~~
- ~~“Left as an exercise for the reader.”~~
- Celebrate automated theorem provers!
- Online scientific notebooks, paper websites with full proofs

Algorithm Reproducibility

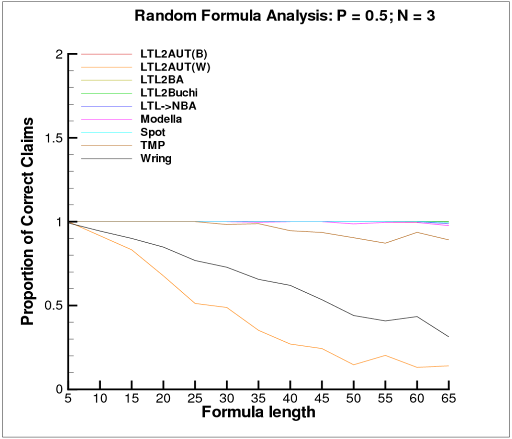
- List the algorithm *in its entirety*
- Accompany formal semantics with intuitive English explanation
- Comments! (They are not just for code!)
- Accompanying reference implementation
- Utilize multimedia (i.e. post a video demonstration)

Address both algorithms as main contributions and algorithms for reproducing important figures and statistics!

Separating the algorithm from its implementation aids in both *reproducibility* and *correctness*.

Oxford Algorithms Graphic: `http://www.cs.ox.ac.uk/research/algorithms/`

Correctness Study[2]



Benchmarking via satisfiability checking of peer-reviewed algorithms for encoding Linear Temporal Logic formulas as automata:

- Every publicly available algorithm was wrong sometimes.
- Wrong = reporting SAT when a formula is UNSAT and vice versa.

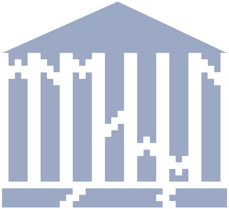
Correctness should be explicitly demonstrated before publication!

[2] K.Y. Rozier and M.Y. Vardi. “LTL Satisfiability Checking.” STTT, 2010.

Buildable Algorithms

Buildability:

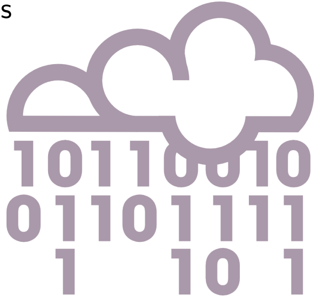
- Completeness: types, inputs, data structures
- Labels:
  - clever tricks
  - dependencies, sensitivities to ordering
  - opportunities for parallelization, optimization
- Documentation! Comments! Examples!



Code Reproducibility

Release (as much as possible) the code!

- List the code *in its entirety*: online, in an appendix, or both
- Use online code databases and repositories
- Accompany code with algorithms, precompiled binaries, test cases
- Documentation!
  - Why the code is useful
  - What the code does
  - Operating systems, platforms, hardware
  - Illustrative example
  - Function descriptions
  - Suggestions for reuse



Address both code as a main contribution and code for reproducing important figures and statistics!

Code Correctness: An Active Research Area

- Automated verification tools
- Compare outputs to alternative implementations
- Release the source
- Automated tests
- Online software development pages/bug tracking systems
- Sanity checks!

When comparing run times against another tool, did both tools return the same answer for each of those run times?

Sanity checks:

- Does the tool always have the same high-level behavior?
- Does the output fall into the expected region of answer-space?

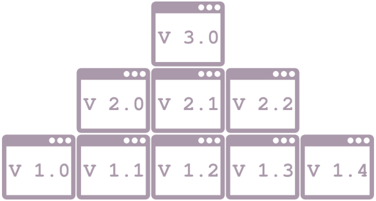
Code to Build Upon

Buildability:

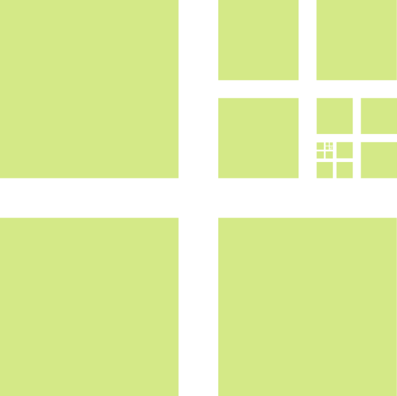
- dependencies with version numbers
- license, instructions for use
- automatic code checks
- *literate programming* combines code and documentation
- listing of flags/optional arguments

Buildability for non-releasable code:

- ability to separately execute functions
- configurable, parameterized front end
- use cases, examples



Ethics and Data



Research Ethics

- **Reproducibility** - Enable the confirmation of results from an empirical study. Builds trust during peer review.
- **Correctness** - Document data as well as methods. Provide data sets as part of proofs of correctness.
- **Buildability** - Need long term solutions for storing and hosting data for longitudinal studies and buildable work.

Ethics and Data

Professional Ethics

- **Privacy** - Customer data, private records, etc.
- **Compliance** - HIPAA, Sarbannes-Oxley, etc.
- **Confidentiality** - Sensitive industrial data, pre-patent information, proprietary data.

Ethics and Data

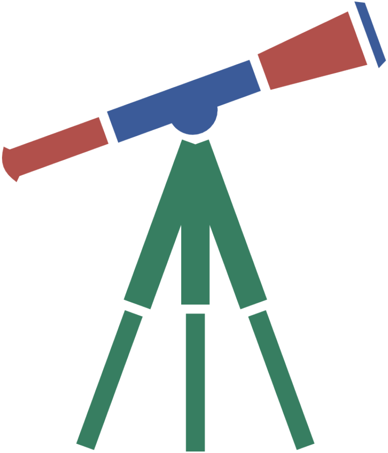
Striking an Ethical Balance

- k-anonymity and deidentification
- data integrity preserving transformations (data condensation, scrambling, swapping, association rule mining)
- synthetic data generation

Conclusions

The idea is to try to give all of the information to help others to judge the value of your contribution; not just the information that leads to judgment in one particular direction or another

- Feynman



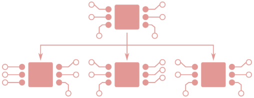
Outlook

Calls for papers should specify which principles, or which combination, they are expecting!

- increase quality, consistency, and objectivity of reviews
- Example: *best* papers in the three-way intersection
- Example: workshop call for reproducibility and partial correctness but not yet full correctness or buildability
- Example: tool paper focusing on buildability and correctness

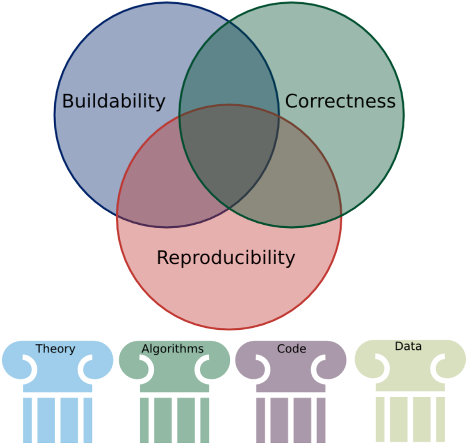
The vast majority of published results in CS/ECE are *positive*

- need support for publishing more impactful negative, but reproducible results
- encourage replication studies



Reproducibility, correctness, and buildability should be included as standard numerical ratings on peer review forms.

Conclusions & Outlook


Buildability
Correctness
Reproducibility
Theory
Algorithms
Code
Data