RESEARCHOPS: A PRINCIPLED FRAMEWORK AND GUIDE TO COMPUTATIONAL REPRODUCIBILITY

AARON WILLCOX | ELLIOT GOULD

2021-07-12

@aaron_willcox | | *@Elliot_Gould_*

RESEARCH CODE

- Source code generated each year grows by about 20% (L. Hatton & M. van Genuchten, 2019).
- Sharing policy increase: 15% in 2015 to 75% in 2020 (Culina et al., 2018).
- Data handling and processing often informally transmitted (Maer-Matei et al., 2019).
- Lack of formal training for researchers (Koehler Leman et al., 2020).

COMPUTATIONAL REPRODUCIBILITY

The ability to produce equivalent analytical outcomes from the same data set using the same code and software as the original study (Fidler et al., 2017).

CHALLENGES IN ECOLOGY

Challenge	Cause or mechanism	Examples	Consequences	Solutions	Source
Regularly Updated Data	Requires active data management, continual data entry, data processing and integration and error checking because data are continually changing.	long-term observational studies, experiments with repeated sampling, use of automatic sensors, ongoing literature mining, iterative near-term forecasting, adaptive management	Large burden on small teams without rapid and automated protocols. Data analysis prone to errors without QA/QC protocols. Reproducibility difficult to achieve without pipeline workflows.	version-control, automated testing, continuous integration and analysis	Yenni et al., (2019)
'Data 'freshness' or the time between data collection and data use.	Data freshness is difficult to track due to variation in reporting practices. This difficulty is increased when many data sources are combined. Unknown data freshness or stale data may increase uncertainty and decrease accuracy in conclusions reached.	SDM's where predictor variables do not capture recent environmental changes, such as rapid habitat loss, or where occurrence records do not coincide with period from which predictor variable captured	Poor model performance, reduced accuracy of predictions in areas of rapid environmental change, increased risk of negative outcomes of conservation decisions	Good metadata that includes temporal aspects of original data collection.	Murray et al., (2021)
Integrating and synthesising independently collected data from many sources	Ecological data often context specific, with many nuances and details in the study-system being poorly documented. Methods section limits are too small to capture full suite of details.	Complex modelling studies, conservation-decision-making studies, model transfers	Data are easily misinterpreted, biases unknown, and may pose statistical issues when integrating across multiple dimensions and sources.	use of FAIR data principles (FAIR: Findable, Accessible, Interoperable and Reusable), use of TRUST principles: Transparency, Responsibility, User focus, Sustainability and Technology, data archiving practices that adheres to these principles.	Culina et al., (2018)
Manual / hard-copy data entry	data collected on data sheets in the field or lab. Data structure not enforced by hand-recording, mistakes in data entry.	Hard-copy, free-hand field-data recording. Experimental protocols and results recorded by hand in lab-notebooks.	Errors in data entry may result in serious errors in conclusions, especially if systematic bias in recording errors.	Digital data recording with the use of schemas to enforce required data structure. Automated testing or QA/QC upon data entry.	Yenni et al., (2019)
Bio-logging and automated sensors	Ongoing QA/QC and data processing necessary, no standards for archiving data, most data are undiscoverable and inaccessible.	Camera traps, weather data, geo-location tracking, remote sensing or drone data, bio-logging data	Burden on researchers wanting to either share or reuse bio-logging data, datasets unable to be merged.	FAIR, TRUST principles, standardised templates and metadata, workflows for producing archive-quality data files/td>	(Sequeira et al., 2021)

@aaron_willcox | | @Elliot_Gould_

HAVE YOU REPRODUCED LATELY?

- Archmiller et al., (2020) Found 74 suitable for CR of the 19 obtained 13 were able to mostly or fully reproduce.
- Obels et al., (2020) 62 articles identified, 41 had data available and 37 had analysis scripts Ran scripts for 31 analysis and reproduced main results for 21 articles.
    - Increase Code Sharing.
    - Organization and Documentation and Training.
    - Good Research Practices.


ins
hom
}
]
del
BEST
practice
Shift

SOURCE OF IRREPRODUCIBLE RESULTS

- Lack of a workflow framework.
- Missing software dependencies.
- Excluded data manipulation steps (Leipzig et al., 2020).
- **Irreproducibility and a lack of transparency can be overcome by borrowing a set of tools and practices from software engineering, called DevOps**

DEVOPS

- **Version Control**: Historical context of data and code changes.
- **Containers**: System environmental configuration.
- **Continuous Practices (CI/CD)**: Quality assurance and automation.
- **Testing**: Expected constraints at output.

MODERN SCIENTIFIC RESEARCH

- No differences between researchers from computer science (Yasmin AlNoamany & John A. Borghi, 2018).
- Computational reproducibility best approached by focusing on software as a product (Hocquet & Wieber, 2021).
- More easily achieve computational reproducibility.
- “*Product*” is the reproducible outcome built around a scientific workflow.

RESEARCHOPS

The Case for DevOps in Scientific Applications (de Bayser et al., 2015)

- Aid in computational reproducibility and transparency of their work (Beaulieu-Jones & Greene, 2017; Wittman & Aukema, 2020)
- Increase scientific productivity (Peikert & Brandmaier, 2019)
- Collaborate effectively within and between researchers (Díaz et al., 2019)

WORKFKFLOWS, PIPELINES & COMPONENTS! OH MY!

- **Scientific Workflow**: Overall scope of the research project.
- **Pipeline**: Execution of each process or stages of the scientific workflow.
- **Components**: Tools and/or software adopted to execute the pipeline to deliver research outcomes.

RESEARCHOPS FRAMEWORK

Project Management

• **Design & Infrastructure**	Identify inputs and outputs, programming langauge and environmental infrastrucutre.
• **Data Standardization**	Identify key data assets and file nomenclature, directory and file structure.
• **Operating Procedure**	Identify how to collaborate & communicate on code and issues.
• **Documentation**	Maintain documentation through a Wiki to preserve previous steps.

Computational Reproducibility

DevOps		ResearchOps	
Components	**Application**	**Purpose**	**Reproducible Outcome**
Continuous Integration	Github Actions Travis-CI Gitlab	Testing & Quality Assurance Automation and delivery	Review any irreproducible results
Version Control System	git subversion	Development History	Historical Context of decisions and changes
Containers	Docker Reprozip Kubernettes	Maintain environmental software dependencies	Execute run time environment of research pipeline

@aaron_willcox | | @Elliot_Gould_

PROJECT SCOPE

Project Management	
• **Design & Infrastructure**	Identify inputs and outputs, programming langauge and environmental infrastrucutre.
• **Data Standardization**	Identify key data assets and file nomenclature, directory and file structure.
• **Operating Procedure**	Identify how to collaborate & communicate on code and issues.
• **Documentation**	Maintain documentation through a Wiki to preserve previous steps.

DevOps		ResearchOps	
Components	**Application**	**Purpose**	**Reproducible Outcome**
Continuous Integration	Github Actions Travis-CI Gitlab	Testing & Quality Assurance Automation and delivery	Review any irreproducible results
Version Control System	git subversion	Development History	Historical Context of decisions and changes
Containers	Docker Reprozip Kubernettes	Maintain environmental software dependencies	Execute run time environment of research pipeline

Computational Reproducibility

@aaron_willcox | | @Elliot_Gould_

PIPELINE

Project Management	
• Design & Infrastructure	Identify inputs and outputs, programming langauge and environmental infrastrucutre.
• Data Standardization	Identify key data assets and file nomenclature, directory and file structure.
• Operating Procedure	Identify how to collaborate & communicate on code and issues.
• Documentation	Maintain documentation through a Wiki to preserve previous steps.

DevOps		ResearchOps	
Components	**Application**	**Purpose**	**Reproducible Outcome**
Continuous Integration	Github Actions Travis-CI Gitlab	Testing & Quality Assurance Automation and delivery	Review any irreproducible results
Version Control System	git subversion	Development History	Historical Context of decisions and changes
Containers	Docker Reprozip Kubernettes	Maintain environmental software dependencies	Execute run time environment of research pipeline
Pipeline			

@aaron_willcox | | @Elliot_Gould_

RESEARCH OUTCOME

• **Design & Infrastructure**	Identify inputs and outputs, programming langauge and environmental infrastrucutre.
• **Data Standardization**	Identify key data assets and file nomenclature, directory and file structure.
• **Operating Procedure**	Identify how to collaborate & communicate on code and issues.
• **Documentation**	Maintain documentation through a Wiki to preserve previous steps.

Project Management

DevOps			ResearchOps
Components	**Application**	**Purpose**	**Reproducible Outcome**
Continuous Integration	Github Actions Travis-CI Gitlab	Testing & Quality Assurance Automation and delivery	Review any irreproducible results
Version Control System	git subversion	Development History	Historical Context of decisions and changes
Containers	Docker Reprozip Kubernettes	Maintain environmental software dependencies	Execute run time environment of research pipeline

Computational Reproducibility

@aaron_willcox | | @Elliot_Gould_

RESEARCHOPS FRAMEWORK

• **Design & Infrastructure**	Identify inputs and outputs, programming langauge and environmental infrastrucutre.
• **Data Standardization**	Identify key data assets and file nomenclature, directory and file structure.
• **Operating Procedure**	Identify how to collaborate & communicate on code and issues.
• **Documentation**	Maintain documentation through a Wiki to preserve previous steps.

Project Management

DevOps		**ResearchOps**	
Components	**Application**	**Purpose**	**Reproducible Outcome**
Continuous Integration	Github Actions Travis-CI Gitlab	Testing & Quality Assurance Automation and delivery	Review any irreproducible results
Version Control System	git subversion	Development History	Historical Context of decisions and changes
Containers	Docker Reprozip Kubernettes	Maintain environmental software dependencies	Execute run time environment of research pipeline

Computational Reproducibility

@aaron_willcox | | @Elliot_Gould_

THANK YOU!

@aaron_willcox

*@Elliot_Gould_*

References

AlNoamany, Yasmin, and John A. Borghi. 2018. "Towards Computational Reproducibility: Researcher Perspectives on the Use and Sharing of Software." Article. *PEERJ COMPUTER SCIENCE*, September. https://doi.org/ghkb9j (https://doi.org/ghkb9j).

Archmiller, Althea A., Andrew D. Johnson, Jane Nolan, Margaret Edwards, Lisa H. Elliott, Jake M. Ferguson, Fabiola Iannarilli, et al. 2020. "Computational Reproducibility in the Wildlife Society's Flagship Journals." Article. *JOURNAL OF WILDLIFE MANAGEMENT* 84 (5): 1012–17. https://doi.org/gg66q7 (https://doi.org/gg66q7).

Beaulieu-Jones, Brett K, and Casey S Greene. 2017. "Reproducibility of Computational Workflows Is Automated Using Continuous Analysis." *Nature Biotechnology*, no. 4: 342.

Catlin, Ann Christine, Chandima HewaNadungodage, and Andres Bejarano. 2019. "Lifecycle Support for Scientific Investigations: Integrating Data, Computing, and Workflows." Article. *COMPUTING IN SCIENCE & ENGINEERING* 21 (4): 49–61. https://doi.org/gjt725 (https://doi.org/gjt725).

Culina, Antica et al. 2018. "Navigating the Unfolding Open Data Landscape in Ecology and Evolution." *Nature Ecology & Evolution*, 1–7.

de Bayser, Maximilien, Leonardo G. Azevedo, and Renato Cerqueira. 2015. "ResearchOps: The Case for DevOps in Scientific Applications." In *2015 IFIP/IEEE International Symposium on Integrated Network Management (IM)*, 1398–1404. Ottawa, ON, Canada: IEEE. https://doi.org/ghkb9s (https://doi.org/ghkb9s).

Díaz, Jessica, Jorge Pérez-Martínez, Agustin Yague, Andrea Villegas, and Antonio Antona. 2019. *DevOps in Practice – A Preliminary Analysis of Two Multinational Companies*.

Hatton, L., and M. Van Genuchten. 2019. "Computational Reproducibility: The Elephant in the Room." *IEEE Software* 36 (2): 137–44. https://doi.org/ggkvtr (https://doi.org/ggkvtr).

Hocquet, Alexandre, and Frederic Wieber. 2021a. "Epistemic Issues in Computational Reproducibility: Software as the Elephant in the Room." Article. *EUROPEAN JOURNAL FOR PHILOSOPHY OF SCIENCE* 11 (2). https://doi.org/gkm94m (https://doi.org/gkm94m).

Hocquet, Alexandre, and Frédéric Wieber. 2021b. "Epistemic Issues in Computational Reproducibility: Software as the Elephant in the Room." *European Journal for Philosophy of Science* 11 (2). https://doi.org/gkm94m (https://doi.org/gkm94m).

Koehler Leman, Julia, Brian D. Weitzner, P. Douglas Renfrew, Steven M. Lewis, Rocco Moretti, Andrew M. Watkins, Vikram Khipple Mulligan, et al. 2020. "Better Together: Elements of Successful Scientific Software Development in a Distributed Collaborative Community." *PLoS Computational Biology* 16 (5): e1007507. https://doi.org/ggt62r (https://doi.org/ggt62r).

Leipzig, Jeremy, Daniel Nüst, Charles Tapley Hoyt, Stian Soiland-Reyes, Karthik Ram, and Jane Greenberg. 2020. "The Role of Metadata in Reproducible Computational Research."

Maer-Matei, Monica Mihaela, Tiberiu Marian Georgescu, Cristina Mocanu, and Ana-Maria Zamfir. 2019/01/01////. "Skill Needs for Early Career Researchers." *Sustainability* 11 (10). https://doi.org/ghkb9m (https://doi.org/ghkb9m).

Murray, NJ et al. 2021. "Data Freshness in Ecology and Conservation." *Trends Ecol Evol* 36 (6): 485–87.

Obels, Pepijn, Daniël Lakens, Nicholas Coles, Jaroslav Gottfried, and Seth Green. 2020. "Analysis of Open Data and Computational Reproducibility in Registered Reports in Psychology." *Advances in Methods and Practices in Psychological Science* 3 (May): 251524592091887. https://doi.org/gg4vw4 (https://doi.org/gg4vw4).

Peikert, Aaron, and Andreas Markus Brandmaier. 2019. "A Reproducible Data Analysis Workflow with R Markdown, Git, Make, and Docker." PsyArXiv. https://doi.org/10.31234/osf.io/8xzqy (https://doi.org/10.31234/osf.io/8xzqy).