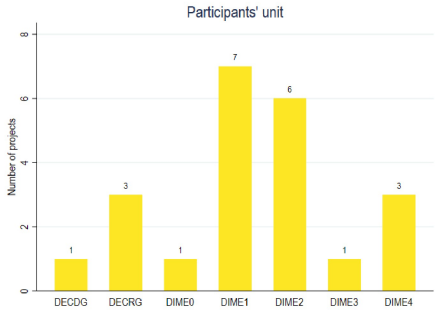
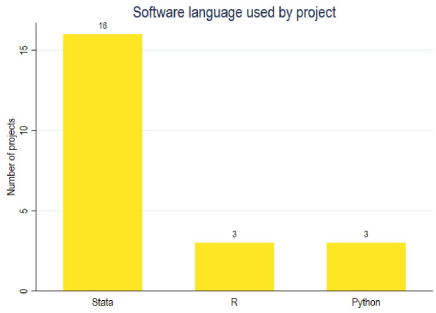
Peer Code Review Summary - FY23 Q2

DIME Analytics

A total of **22 research assistants** joined the peer code review held in the week of February 13th, 2022, and reviewed code from **20 different projects**. DIME1 and DIME2 were the most-represented units. Most projects used Stata as the main coding language.



(a) Participant Units

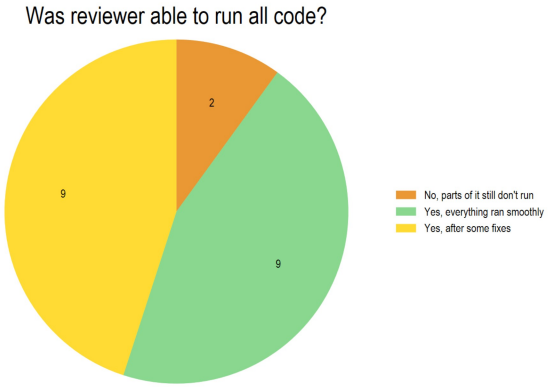


(b) Participant Software Used

Reproducibility

Out of the 22 code packages exchanged, **20 included de-identified data, and were evaluated for reproducibility.**

Of these 20, 90% were reproducible: the code file could be run by the reviewer with either minor fixes, or no changes at all. Only 2 packages could not be run by reviewers even after attempted fixes due to complicated folder structure, and inconsistent use of globals.

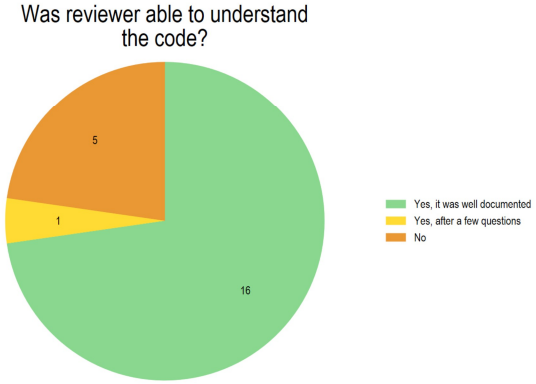


Ease of Use

72% reviewers said that the code they received was **easy to understand and well-documented**.

23% indicated that more comments in the code and additional details in the GitHub README would be helpful.

In terms of **transferability**, 41% reviewers said the material provided would be enough to take over the project without any additional communication with the original coder.



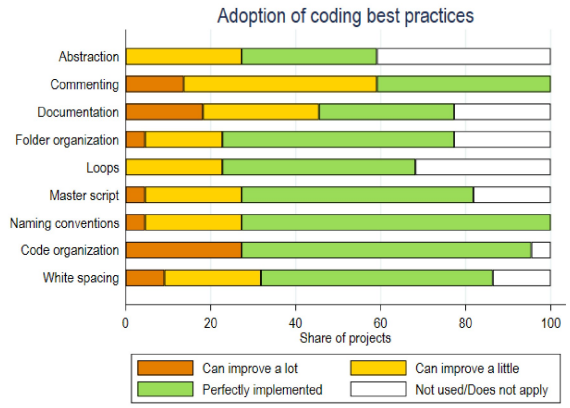
Moreover, 78% reviewers said that it would take them **1 day or less** to be able to understand the code well enough to make contributions to it. The code for 68% projects was rated **easy to maintain**. There was **only 1 project** for which making adaptations would require changes in multiple places, making it hard to build on existing code.

Adoption of Coding Best Practices

The **mean number of best practices adopted was a healthy 6.96** - out of 9 in total. Further, 41% of projects **correctly implemented** each of the best practices.

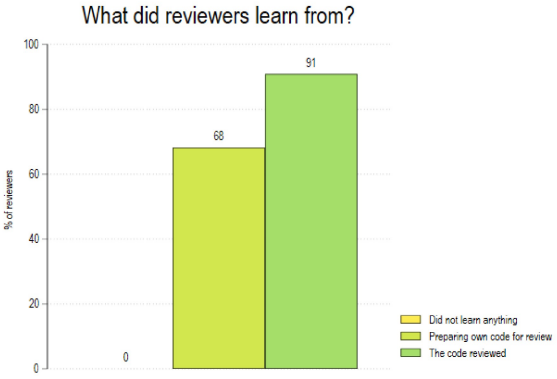
The reviewers identified the **most room for improvement** in use of comments, documentation, and white spacing.

The **most widely adopted best practices**, include naming conventions for files, code and folder organization, and use of master scripts.



Feedback and Challenges

All participants reported learning something new from the code review exercise. 17 reviewers said they learned from the code reviewed, 15 reviewers reported learning from preparing their own code for submission, and 13 reviewers reported learning from both. In addition, **participants also reported learning** the following:



- Importance of focusing on reproducibility early in the process
- How to better organize code
- New commands and functions to perform tasks more efficiently
- Use of master scripts to define basic settings

Finally, the **primary challenges** identified during the exercise include **time constraints** to work on the code review, and **lack of coordination** with partners who were absent during the work party. Participants also requested more time to review the code packages received. Based on this feedback, we aim to enforce stricter attendance requirements from participants, and provide more time to participants to submit feedback.

Participant Comments

"The team did a great job on the organization, with very clear deadlines and updates."

"This exercise allowed me to have a general framework on how I should work. The exercise taught me to understand the practices of my peers, and code with other users in mind."