Problem Set: Reproducible Research Workflows

Yasin Shafi

23 January 2026

Note on data. This assignment uses a **synthetic** voter-turnout dataset provided by `poliscitools` (via `example_data`). The goal is to practice reproducibility tooling (dependency management, logging, packaging outputs, and replication checks)—not to draw substantive inferences about real voters.

Conceptual Questions

Please write three to ten sentence explanations for each of the following questions. **You are only required to answer ONE of the two questions below.**

1. Explain what problem `renv` solves in reproducible research. In your answer, describe what information is stored in `renv.lock`, what `renv::restore()` does, and why sharing code without dependency versions can fail replication even when the analysis is "correct."

   Answer: The renv package solves the problem related to dependency. When R packages update over time, code that worked previously may produce different results or fail entirely. Packages get updated. A function may change its default behavior, get deprecated, or produce different output with a newer version.

   The renv.lock file is a JSON manifest that records the exact version of every package used in a project, including the package name, version number, source (CRAN, GitHub, etc.), and R version. This creates a snapshot of the computational environment at a specific point in time.

   When a collaborator clones the project and runs renv::restore(), the function reads renv.lock and installs the exact package versions listed there into a project-local library. This recreates the original environment regardless of what packages or versions are installed on their system.

   Even when analysis logic is correct, replication can fail because default settings change between package versions, functions get renamed or removed, statistical algorithms get updated producing slightly different estimates, and sorting or rounding behavior may differ. Without version information, a collaborator installs the latest packages and may get different results, encounter errors, or be unable to run the code at all. The analysis is "correct" but not reproducible because the computational environment wasn't preserved.

Applied Exercises

Answer: My work is documented in the following repository: `https://github.com/yasinshafi/soda501`

In the github repository, the scripts and files are loaded. In 02_reproducibility folder, the files and folders are:

- lecture
- README.md
- slides
- demo
- 'Heath, Davidson, et al. (2023) The Journal of Finance.pdf'
- 'Holzmeister,et al. (2025) Nature Human Behaviour.pdf'
- problem_set

In the `demo` folder, I produce the deliverables for the problem set of this week. Here is a list of files and folders in the `demo` folder:

- analysis_log.txt
- outputs
- renv.lock
- reproducibility.py
- data
- renv
- reproducibility.R
- requirements.txt