Reproducible & Automated Workflow

Hyoungchul Kim

The Wharton School, University of Pennsylvania

September 9, 2024

Motivation

Reproducible workflow is becoming more important in (Empirical) Economics research

- Empirical research is becoming more intense and complex.
- More prone to making fatal error.
- It also gives more control to the researcher.
- It makes workflow more efficient (e.g. small change in code script).

$\Longrightarrow$ I plan to discuss about some tools I have been looking into during last few weeks.

Caveat

- First off, I don't use Stata. I use R.
- Some of these reproducible tools can be more inclined toward R user.
- But still, it will give nice tools you can adjust to use on your language (Stata, Python, etc).

R-package version: RENV package

Problem

- Just like other programming languages, you use many useful packages (or libraries) in R.
- But some packages become outdated or inconsistent over time.
- This could lead to severe consequences where your code will not work after update (e.g. You update your package and it starts to not work).

Solution: Use Renv package in R!

- Renv package store all the packages you use in separate library.
- It stores all info of the package (including version) in the renv.lock file.
- Allows reproducibility since other ppl can use the lock file to restore the package version.

Caveat of Renv

1. You can only use it in R (but you can use Poetry on Python).
2. It does not track the version of R itself.
3. It does not track the version of dependenceis of the packages.

Conda (mamba) environment

Solution to problem in Renv: Use conda environment!

- Conda provides separate virtual environment to store packages.
- You can also track version of R and Python itself.
- It works on other languages such as Python, etc.
- Caveat: Cannot still track dependencies of the dependencies (e.g. pip).
- Only install through conda.

It works on my computer!: Docker

Unfortunately, just tracking versions of packages is not enough.

- How packages work also depends on the overall system dependencies and OS you use (linux, mac, windows).
- In order to fully incorporate these dependencies, we need to ship our computer to others.
- This is impossible! ... Or is it?

Solution: Use Docker!

- Docker basically helps you to ship your computer to other ppl.
- Using Dockerfile it gives an instruction on how to build your computer inside their computer.
- Almost like a virtual environment (with just the necessary dependencies).

Make

- Research workflow is a DAG process.
- As the workflow becomes complicated, it is not easy to manually check this process.
- e.g. What do I need to re-run if I am just changing one part of the script?
- Make does this for you by re-running the necessary processes whenever there is a change in the dependencies.

Conclusion

For more . . .

- Check Prof. Dingel's website (nice place to start).
- You can also try some resources I've put in my repository.
- You also need to learn some command line (bash, terminal) to do this (but the learning curve is not that steep).