

Complex Systems Workflow

Fitz Koch

December 10th, 2025

Outline

- 1 Philosophy: Modular, Replicable, Automated
- 2 What I Do: Learning Tools
- 3 What I Do: Coding, Paths, Figures, Latex

The Whole Point

Learn

- Memorize
- Categorize
- Synthesize
- Connect
- Expand

Create

- Literature
- Explore
- Hypothesize
- Test
- Replicate
- Analyze

Communicate

- Visualize
- Stabilize
- Consolidate
- Format
- Share

Find → Read → Notes → Ideas → Experiment Design → Code →
Data → Figures → Writing → Publishing → Archiving

How to Get There

Complex Systems = Data Science + Physics + Epistemology

- **Modular:** Playful, expandable, iterable
- **Replicable:** Collaboration without subscription, fixing, expansion
- **Automated:** Systems require computation

Consolidated Pipeline of Open Source Local Software

Moral and Virtue Arguments

- **Moral Thought Experiment:** AI-assisted automation VS automation with AI. Consider environmental, social, personal ramifications. [3]
- **Virtue:** Better to understand the things you are using well.
- **Moral:** Better to live in world where skill and understanding are valuable and personal rather than auctioned off for false convenience.
- **Both:** How you do what you do should be a decision you make, not one that corporations make for you. [2]

Technical and Self-Interested Arguments

- **Venal Self Interest:**
Pipelining as a skillset (the age of the ipynb is fading)
- **Version Control**
- **Regular Control**
- **Ongoing Access**
- ***Clearer opportunities for improvement***
- **Fun!**



Class Notes

- In Class: pen and paper
- Later: transferred, not transcribed, to Obsidian
- Clarified with wikipedia, textbooks, youtube, AI
- Categorize, Connect, Store, Label
- Importance of not being a freak about it.

Research Notes

- Zotero first
- Annotate and highlight digitally or on paper (if hard to focus on)
- Larger ideas and concepts in citation page in zotero
- Ideas integrated into conceptual 'evergreen' notes

Flash Cards

- Obsidian → ANKI
- Flashcards of formulas, phrases, ideas that I think will anchor my memory for things that are important to me
- Importance of not being a freak about it, seriously, we mean it.

Live Demo

Live Demo

Create

The Pipeline: experiment ideas in Obsidian turn into code in `src` uses data in `data` to make figures in `latex/figures` which `latex/project.tex`, possibly drafted in Obsidian, complies to share results.

Computational Experiments

A whole slide deck in itself....

Coding Practices

- **Mise en Place:** Clean Cutting Board, Clean Mind. Build a recipe, then prepare ingredients, then put together.
- **DRY:** Be kind to your future self! Like getting a glass of water after going out drinking.
- **Modular:** Organized functionality = iterable functionality.
- **Paths:** Never raw strings, (python: Pathlib) and as limited absolute paths as possible
- **Reproducibility:** if in python, use conda/uv or at least a regularly pip list.
- Make stuff so that you can use it in later projects!
- Tests, Types, **Dataclasses**
- **Doing everything like you want to be better at doing it next time.**

Data Management

- Don't put it on github
- Consider scale before you start
- Standardize, Log, and *save a record of all transformations*, ideally in a replicable script

Latex

- Local is better than overleaf: version control, directly piping in figures and results from code, potential for more automated writing.
- Write your own snippets and macros
- Latex extensions in vs code make it really easy
- Exporting full zotero bib allows for really quick citations.
- Live Demo!

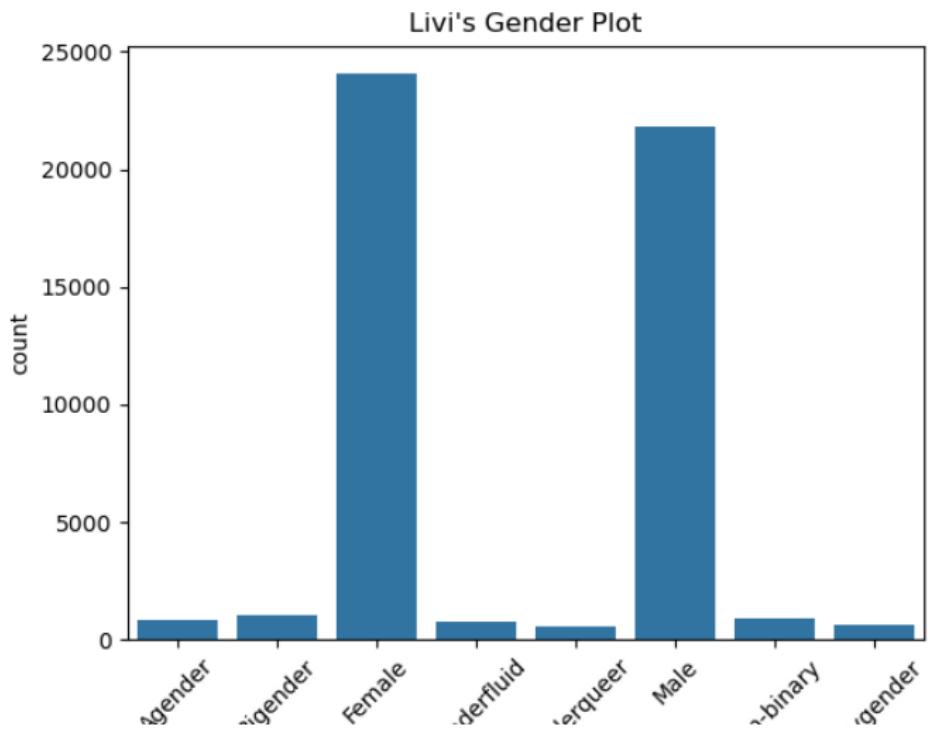


Figure: Gender distribution, which we will here claim has something to do with R^2 [1]

Version Control: GIT

- Commit when you might break something; push when you've changed or fixed something; PR when you've finished a defined subproject
- Easy to link an employer a cool github.
- Easy to bring in collaborators.
- Test: guess the GOODNESS levels of these three options:
 - VS Code extension
 - Terminal
 - Github GUI

References I

- [1] “Coefficient of Determination”. In: *Wikipedia* (Dec. 2025). (Visited on 12/10/2025).
- [2] *How Google Took Over the Classroom - The New York Times.* <https://www.nytimes.com/2017/05/13/technology/google-education-chromebooks-schools.html>. (Visited on 12/07/2025).
- [3] Laura Rodríguez Salamanca. *Microsoft, Google Say Their Data Centers Create Thousands of Jobs. Their Permit Filings Say Otherwise.* <https://restofworld.org/2025/data-centers-jobs-microsoft-google-chile/>. Nov. 2025. (Visited on 12/07/2025).