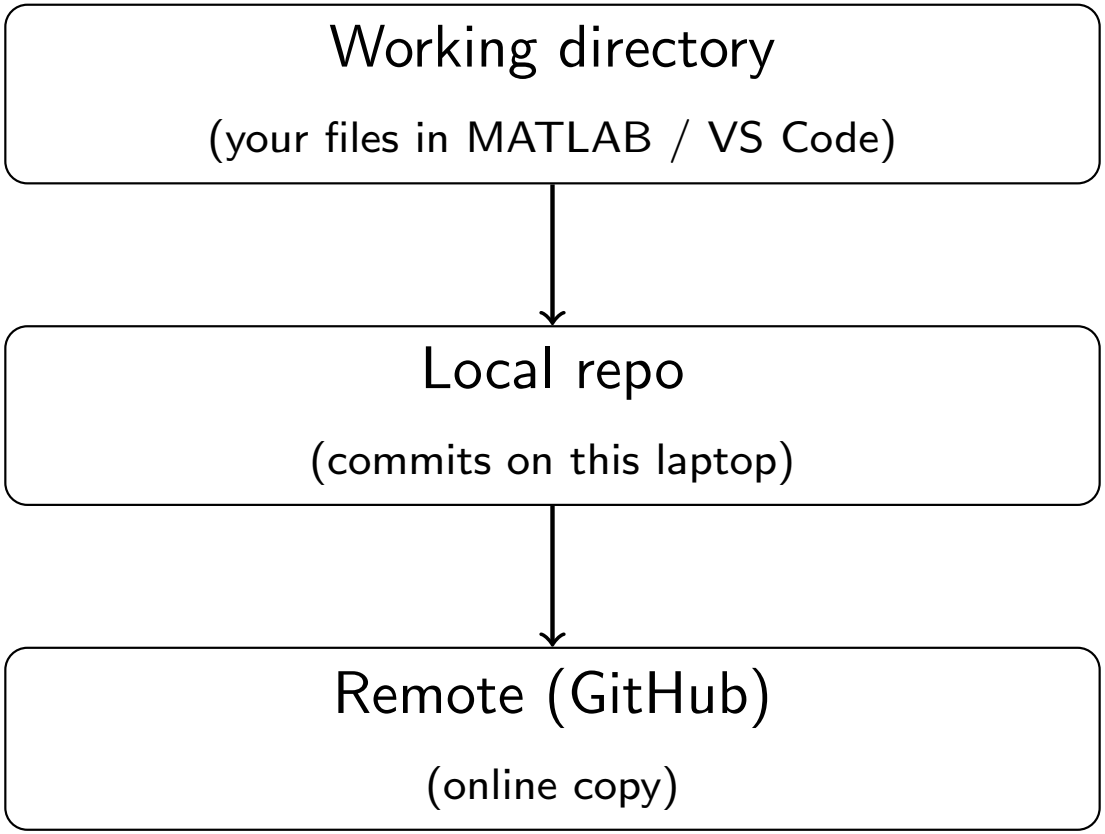
CAMP – Intro to Git with RL_ConnectK (MATLAB)

Becca Thomases

January 14, 2026

Git mental model: three layers



- **Working dir:** edited files (may be unsaved / untracked).
- **Local repo:** snapshots with messages = commits (`git add`, `git commit`).
- **Remote (GitHub):** shared copy online (`git push`, `git pull`).

RL_ConnectK structure (MATLAB)

- `connectK_train.m`
  - Sets up a 4×4, $K = 3$ ConnectK game.
  - Trains a **REINFORCE** (policy-gradient) agent.
  - Calls `save_connectk_agent` to create a saved agent in `Saved Agents/`.
- Tournament + baselines
  - `run_connectk_tournament.m`: plays many games between two agents (function handles).
  - `Run_tournament.m`: example script (trained agent vs `random_agent`).
  - `random_agent.m`: baseline that picks a random legal column.
- Saved agents
  - `save_connectk_agent.m`: saves `policy` + `metadata` and auto-generates `Saved Agents/agent_NAME.m`.
  - `make_saved_agent_handle.m`: given a name like `'agent_1_4x4_k3'` returns `@agent_1_4x4_k3`.
  - `Saved Agents/`: contains `.mat` file + wrapper function for each trained agent.

Today: using git with RL_ConnectK

1. **Get the code**
   - Clone the repo from GitHub with `git clone`.
   - Open the folder in MATLAB.
2. **Run the existing pipeline**
   - `connectK_train` $\Rightarrow$ trains and saves `agent_1_4x4_k3`.
   - `Run_tournament` $\Rightarrow$ runs trained agent vs `random_agent`.
3. **Make one small code change**
   - Example: change `N_EPISODES` or add a clarifying comment.
4. **Record it with git**
   - `git status` (see what changed)
   - `git add` (stage the file)
   - `git commit -m "..."` (save a snapshot locally)

Forks vs the main repo (GitHub)

- The instructor repo lives at, e.g.:

  `github.com/beccathomases/RL_ConnectK`

- A **fork** is your own copy of that repo under your account:

  `github.com/`*`your-username`*`/RL_ConnectK`

- Typical student workflow:
  1. Fork the instructor repo on GitHub.
  2. Clone *your* fork to your laptop.
  3. Make commits locally as you experiment.
  4. `git push` $\Rightarrow$ your changes appear in your fork.
- Later: you can open a **pull request** from your fork back to the main repo if we want to share improvements.