Part IV

GIT

TABLE OF CONTENTS

GIT FUNDAMENTALS

What Git does

Git is a *distributed version control system*: it records the history of a project as a sequence of snapshots (i.e history-bricks).

```
* 8f3a91c   (HEAD -> main, origin/main) Add transformer slides
* 4c21b7a   Fix Flow Matching objective notation
* 7ab13cd   Merge branch 'feature/attention-diagram'
|\
| * c6f8e21   Draw attention block with TikZ
| * a13bd42   Add QKV explanation
|/
* 2e91fa0   Initial Beamer structure
```

Some vocabulary

Each commit stores a coherent state of the code, making it possible to inspect, compare, and recover previous versions.

- *Repository*: the project history, including commits, branches, tags, and metadata.
- *Commit*: an immutable snapshot (a history-brick) of the project, linked to previous commits.
- *Branch*: a separate line of development (i.e a separate history line), used to work on a feature or experiment without changing the main version of the project.
- *Merge / rebase*: mechanisms to integrate changes from different branches.

GIT IN VS CODE

Integrated workflow

VS Code provides a graphical interface for common Git operations.

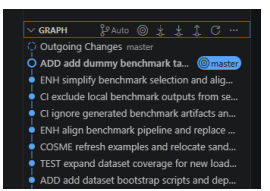

Figure. Git history (on the left panel) in Visual Code Studio.

Commands

- *Stage changes*: select which modifications should be included in the next commit.
- *Commit*: save a coherent snapshot with a clear message.
- *Push / pull*: synchronize local commits with the remote repository.
- *Branch management*: create, switch, and merge branches directly from the editor.

Git vs. GitHub

Git is the *version control system*; GitHub is a *collaboration platform* built around Git repositories. Git tracks local history, while GitHub hosts the shared remote.

Commands

- *Remote repository*: central shared copy, usually called `origin`, used to synchronize work with `push` and `pull`.
- *Feature branches*: isolate new work from the stable `main` branch.
- *Pull requests*: propose changes, discuss code, run automatic checks, and merge only after validation.
- *Good practice*: small commits, clear messages, frequent pulls, and protected `main`.