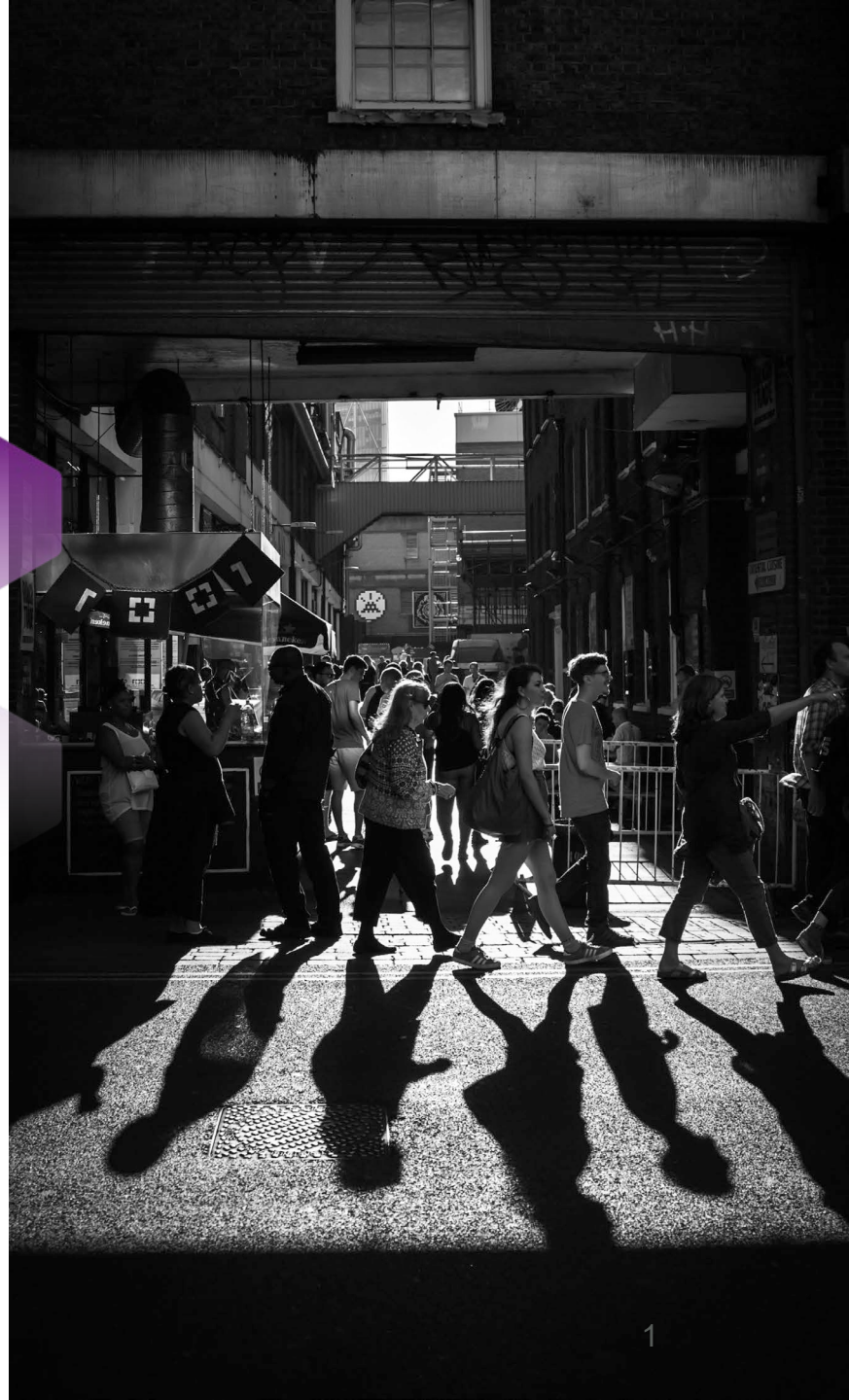
UK Data Service

Code Anxiety Club

Computational Social Science Training Team

UK Data Service



UKRI Economic and Social Research Council

Today’s topic

Date 20/01: Getting Started with Git: Version Control Basics

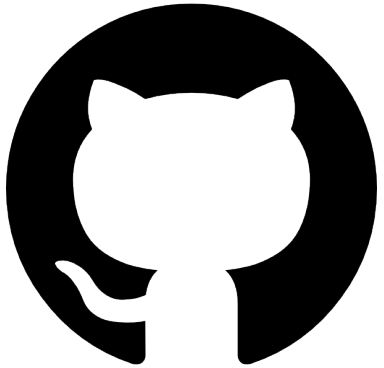
- Understand the importance of tracking changes to files over time
- Install git and master basic git commands such as: git init, git clone, git status, git add, git commit, and git push
- Practical exercises to build confidence in using git and GitHub

What is version control software? Why bother?

- A tool that records changes to files over time
- Saves snapshots (“commits”) with messages
- Rewind, compare, and recover safely
- Collaborate without overwriting each other
- Improves reproducibility

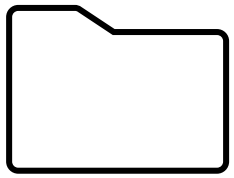
Git vs GitHub

Git: version control program on your computer

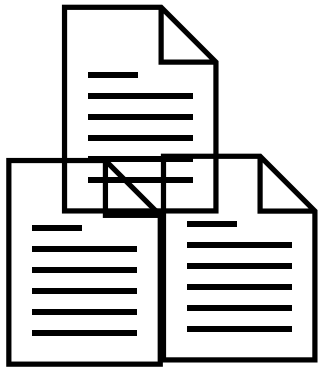


Github: website that hosts Git repos

The 5 Core Ideas



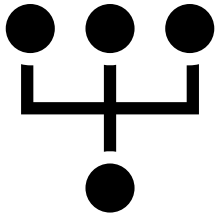
Repository



Staging area

Commit

Remote copy



Branch

Install and first-time setup

MacOS

1. Open terminal
2. Check if git is already present by running:
   git –version

If not installed:

1. Install via Homebrew by running:
   brew install git
2. Check its present by running:
   git –version

Tell Git who you are:

1. Open terminal
2. Enter the following:
   git config –global user.name “Your Name”
   git config –global user.email you@example.com

Install and first-time setup

Windows

1. Go to gitforwindows.org and download the installer
2. Run the installer. When prompted:
   - Editor: choose VS Code (if Python user) or accept default
   - Default branch name: choose main (or accept)
   - PATH: choose “Git from the command line and also from 3rd-party software.”
   - Line endings: accept recommended default
   - Credential manager: leave enabled
   - Accept remaining defaults and finish
3. To verify installation open PowerShell and run:
   git –version

Tell Git who you are:

1. Open PowerShell
2. Enter the following:
   git config --global user.name "Your Name"
   git config --global user.email "you@example.com"

Make your first Git repository

MacOS

```
mkdir -p
my_project/{data,scripts,outputs,docs}
```

PowerShell

```
mkdir my_project; cd my_project; mkdir
data,scripts,outputs,docs
```

```
project/
├ data/
├ scripts/
├ outputs/
├ docs/
└ README.md
```

- cd into your project folder
- git init
- git status
- (Optional) create .gitignore file

Stage and commit (your first snapshot)

1. Open terminal or PowerShell and create a README.md file:
   Run ‘touch README.md’ (MacOS) or ‘New-Item README.md –ItemType File’ (Windows)
2. Open the README.md file and insert a few lines about the project:
   Run ‘open –a TextEdit README.md’ (MacOS) or ‘notepad README.md’ (Windows)
3. Check that the file has indeed been created:
   git status
4. Put it in the staging area:
   git add README.md
5. Confirm the file has been staged:
   git status
6. Commit your changes, and add a description:
   git commit –m “Add README with project description”
7. Verify that the commit is recorded:
   git log --oneline

Push your project to GitHub (backup & share)

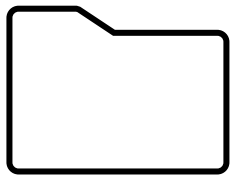
1. Create a GitHub account:
   https://github.com/signup?source=login
2. Log into GitHub and click your profile icon then > Your repositories > click ‘New’ in the top right-hand corner of the screen.
3. Copy and paste the HTTPS address, then in terminal/Powershell navigate to your project folder and run:
   git branch –M main
   git remote add origin (paste the HTTP address here)
4. Push your local changes to your new remote repository by running:
   git push –u origin main
5. If terminal or PowerShell asks for a password follow these steps:
   click your profile icon then > Settings > Developer settings > Personal access tokens > Tokens (classic) > Generate new token > Generate new token (classic) > make sure repo is ticked > click Generate token. Note it down somewhere. When prompted for the password again, paste this token in.
6. Refresh GitHub to see your changes!

Clone a repository

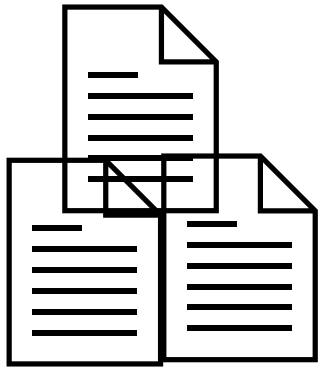
1. Locate the repository you want to clone and click on the green ‘<> Code’ button > Local > HTTPS > copy the web URL
2. Open terminal or PowerShell and navigate to the directory where you want to clone these files
3. Once there run:
   git clone https://github.com/UKDataServiceOpen/text-mining.git
4. Check its worked by navigating to the folder

Your daily Git rhythm

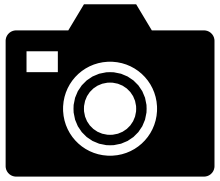
1. Edit code in scripts/ (small changes)
2. git status (see what changed)
3. git add . (stage everything you mean to save)
4. git commit –m “Clear message”
5. git push (back up/share on GitHub)



Repository

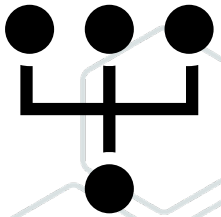


Staging area



Commit

Remote copy



Branch

Common beginner errors

“fatal: not a git repository”

- You’re not inside the project folder → cd project

Nothing happens when you push changes

- You forgot to commit → git add . → git commit –m “Enter message”

UK Data Service

Thank you.

Louise Capener

Email louise.capener@manchester.ac.uk

@CapenerLouise on Twitter/X

@UKDataService on Twitter/X