Immersive Python Workshop Day 2

Python for:

Geospatial Data

Sentiment Analysis

Data Visualization

Managing Python Projects with GitHub

Day 2 Schedule

Time:	
1:00 PM	**Welcome to Day 2**
1:10 PM	**Instruction: Breakout topics (50 minutes)**
	Sentiment Analysis
	Geospatial Analysis
	Data Visualization
2:00 PM	**Break**
2:10 PM	**Instruction: Breakout topics continued (50 minutes)**
	Sentiment Analysis
	Geospatial Analysis
	Data Visualization
3:00 PM	**Break**
3:10 PM	**Instruction: Git and GitHub**
3:55 PM	**Post workshop assessment**
4:00 PM	**Wrap up**

Software Installation Support

★ **Python (https://www.python.org/downloads/)**

  ○ Make sure you can install new Python packages using the Terminal (MacOS) or Command Prompt (Windows)

  ○ The command on a PC should be: **python -m pip install packagename**

  ○ The command on a Mac should be: **python3 -m pip install packagename**

★ **Github Desktop (https://desktop.github.com/download/)**

★ **VSCode (https://code.visualstudio.com/download)**

  ○ Create a new file with the extension .py (to designate the file as a Python script)

  ○ Try clicking the play/run button in the upper right corner of VSCode to run the script

★ **Git (https://git-scm.com/downloads)**

Software Installation Common Issues

★ **Mac**

- ○ **ISSUE:** you cannot run a Python script in VS Code

- ○ **CAUSE:** the *Python default interpreter path* is not set correctly

- ○ **SOLUTION:** In VSCode go to *File > Preferences > Settings and search for the Python default interpreter path* setting - when you find it, update the the value to the correct Python interpreter path

★ **PC**

- ○ **ISSUE:** trying to *python* in the command prompt opens the Windows Store

- ○ **CAUSE:** the directory containing your Python interpreter is not correctly listed in your system Environment Variables' *path* listing

- ○ **SOLUTION:** use the full path to your Python interpreter in the Command Prompt

Creating a New Python File in VSCode

➜   **Open VSCode**

➜   **File > New File**

➜   **Name the file: pythonworkshopday2.py**

➜

Track 1

Topic:
Data Visualization

Location:
Data Lab

Instructor:
Michael Shensky

Track 2

Topic:
Sentiment Analysis

Location:
Project Room 5

Instructor:
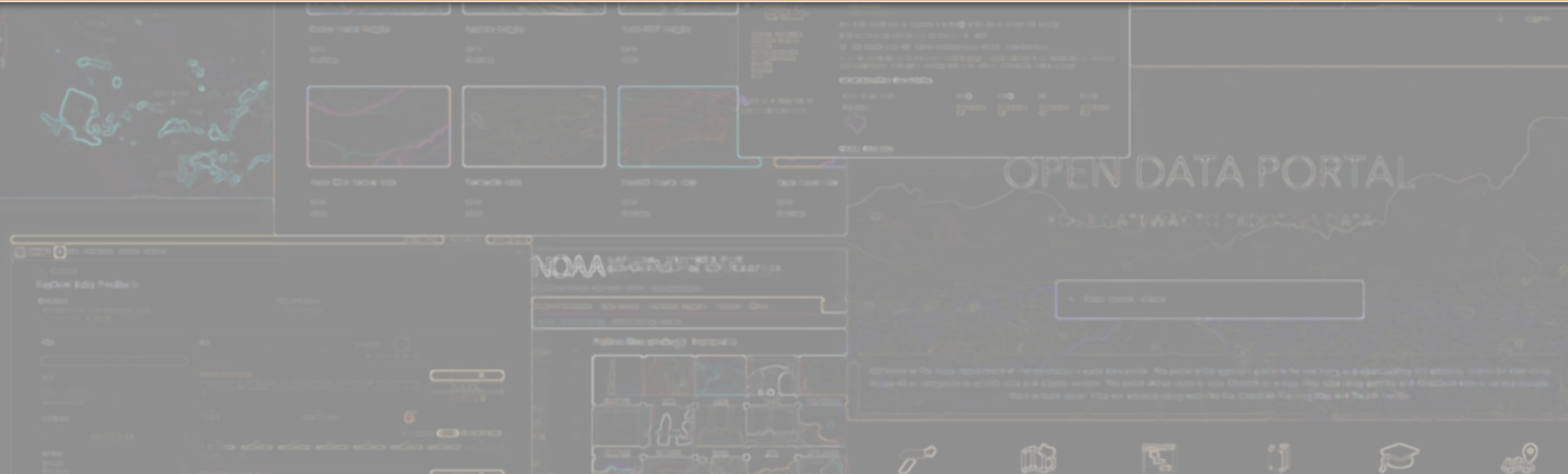Ian Goodale

Track 3

Topic:
Geospatial Data

Location:
Project Room 6

Instructor:
Alex Marden

Immersive Python Workshop Day 2 | Git & GitHub

Introduction to Git

→   https://git-scm.com/downloads

→   Distributed version control system

→   Free and open source

→   Cross platform compatible

→   Run via Terminal or Command Prompt

```
usage: git [-v | --version] [-h | --help] [-C <path>] [-c <name>=<value>]
           [--exec-path[=<path>]] [--html-path] [--man-path] [--info-path]
           [-p | --paginate | -P | --no-pager] [--no-replace-objects] [--bare]
           [--git-dir=<path>] [--work-tree=<path>] [--namespace=<name>]
           [--config-env=<name>=<envvar>] <command> [<args>]

These are common Git commands used in various situations:

start a working area (see also: git help tutorial)
   clone      Clone a repository into a new directory
   init       Create an empty Git repository or reinitialize an existing one

work on the current change (see also: git help everyday)
   add        Add file contents to the index
   mv         Move or rename a file, a directory, or a symlink
   restore    Restore working tree files
   rm         Remove files from the working tree and from the index

examine the history and state (see also: git help revisions)
   bisect     Use binary search to find the commit that introduced a bug
   diff       Show changes between commits, commit and working tree, etc
   grep       Print lines matching a pattern
   log        Show commit logs
   show       Show various types of objects
   status     Show the working tree status

grow, mark and tweak your common history
   branch     List, create, or delete branches
   commit     Record changes to the repository
   merge      Join two or more development histories together
   rebase     Reapply commits on top of another base tip
   reset      Reset current HEAD to the specified state
   switch     Switch branches
   tag        Create, list, delete or verify a tag object signed with GPG

collaborate (see also: git help workflows)
   fetch      Download objects and refs from another repository
   pull       Fetch from and integrate with another repository or a local branch
   push       Update remote refs along with associated objects

'git help -a' and 'git help -g' list available subcommands and some
concept guides. See 'git help <command>' or 'git help <concept>'
to read about a specific subcommand or concept.
See 'git help git' for an overview of the system.
```
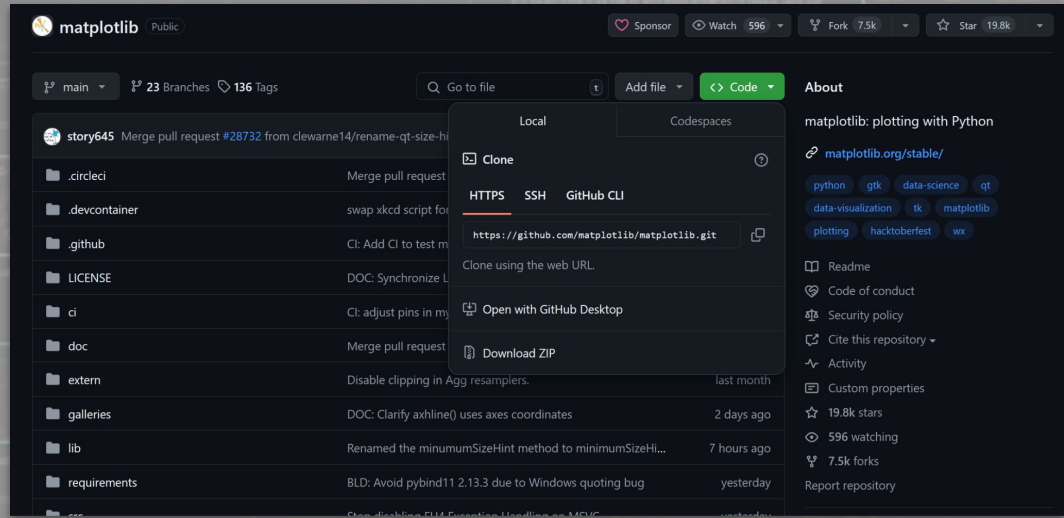
Essential Git Commands

➔ Init

➔ Clone

➔ Commit

➔ Push

➔ Pull

GitHub

➔ GitHub is a great platform for backing up, versioning, collaborating on, and sharing code

➔ It can be a great tool for researchers who want to make their work transparent and replicable

➔ **http://github.com**   **vs.**   **http://github.austin.utexas.edu/** *(for faculty & staff)*

➔ You can use GitHub to publish code for your own projects or to contribute to other software projects for which you are a collaborator or which have been released as open source

Why Push Research Code to GitHub

➔ Get recognition for the coding work that you have done

➔ Share your work with others so that your code can have a greater impact

➔ Make code backups and versioning relatively easy

➔ Potentially benefit from others finding bugs and submitting pull requests for code improvements

➔ Enterprise GitHub and private repositories allow for secure and controlled code management

➔ Help make your research workflows reproducible

GitHub Terminology to Know

➔ **Git**   Open source software used to track and manage versions of files containing text

➔ **Repository**   A container for all of a projects files, they can be open or private

➔ **Commit**   An individual change to a set of files in a repository

➔ **Versioning**   Tracking and saving records of changes to files over time

➔ **Pull request**   Proposed changes to a files in a repository that can be accepted or rejected

➔ **Git Ignore File** A file in a git repository that describes files which should **not** be shared publicly

Getting Started with GitHub

1. Create a GitHub account

2. Download and install GitHub Desktop (already on lab computers)

3. Create a new repository in GitHub

4. Connect GitHub Desktop to your personal GitHub and clone the new repository down to your computer

5. Copy your Python script file that you created earlier today and past it into the repository on your computer

6. Make an initial commit using GitHub Desktop

7. Push the code from your local repository to your GitHub account
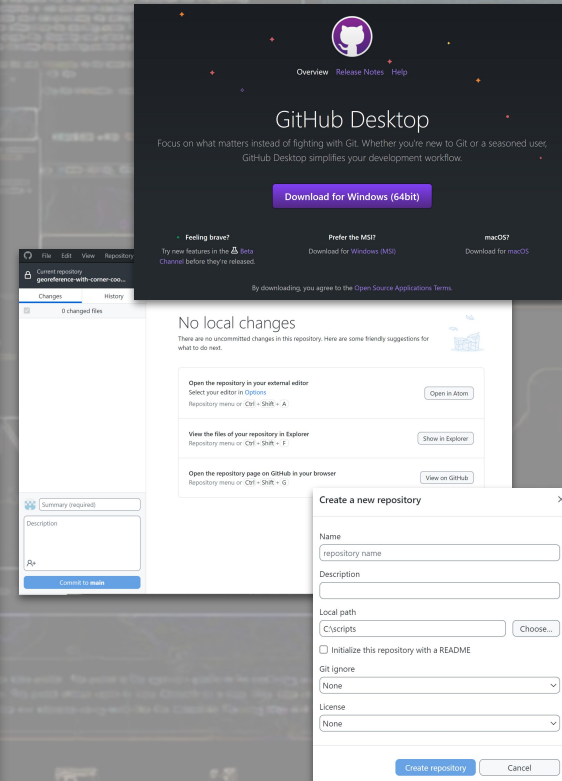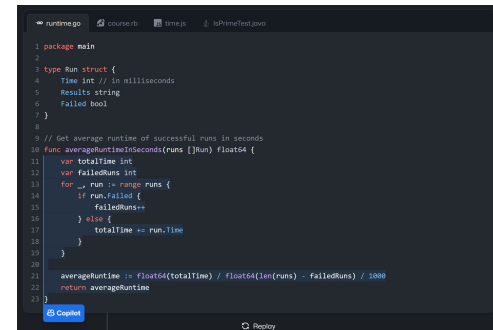
Other GitHub Skills to Practice

- Edit .gitignore file

- Fill out README file

- Fork an existing open source repository

- Submit a pull request

GitHub Copilot
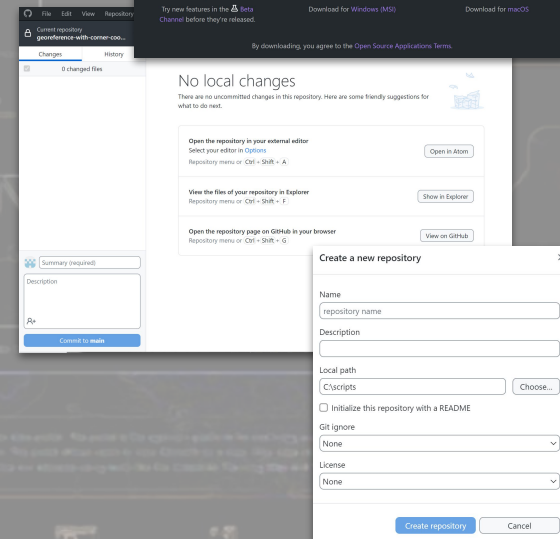
➔ https://github.com/features/copilot

➔ 10$ per month (free trial available)

➔ Good for help with adding new code snippets within a project

➔ Extension in VS Code can be used for free by students, faculty, & open source project maintainers

◆ https://docs.github.com/en/copilot/managing-copilot/managing-copilot-as-an-individual-subscriber/managing-your-copilot-subscription/getting-free-access-to-copilot-as-a-student-teacher-or-maintainer

Selecting a Software Licensing

➔ If you want others to know how they are permitted to use the code you share on GitHub, you should select a license that clearly defines associated restrictions and obligations

➔ There are many options to choose from, but some like these below are frequently chosen

◆ **MIT**

◆ **BSD 3-Clause**

◆ **GNU**

◆ **Apache**

➔ Visit https://choosealicense.com/licenses/ for detailed list of license options

UT Open Source Program Office (UT-OSPO)

→ The **UT Open Source Program Office (UT-OSPO)** is a great new resource for developers of research software

→ Established in 2023 with funding from the Alfred P. Sloan Foundation (Grant Number G-2023-20944)

→ Director: Angela Newell, Ph.D. (anewell@austin.utexas.edu)

→ The Office is a partnership between TACC, ITT, the iSchool, & UT Libraries

UT-OSPO Vision

Focus on faculty, post-docs, grad students - helping them interact with the open source communities for the tools that they use.

Engage faculty and students in an open source participation pathway advancing basic use of open source software through contribution, sharing, accepting external contributions, and ultimately developing an ecosystem of related open source projects.

Develop a community of researchers who use, contribute to or develop open source code and a program office that embraces and supports researchers at every step of the pathway from using code to developing software

18

Open Source Participation Pathway

Using
Researchers use appropriate open source software tools.

Contributing
Research teams have a deeper understanding of an external software community, participate in identifying bugs, asking for new features.

Sharing
Software from a team is made available using open source license/platform.

Accepting
An open source project receives contributions from outside the original research team.

Advancing
An open source project supports outside contributions, but is part of a larger ecosystem of related projects with up/downstream dependencies.

For more information reach out to the UT Open Source Program Office: ospo@utlists.utexas.edu

How Can the OSPO Help Researchers?

➔ Training/Events

➔ Funding Identification

➔ Best practices guides

➔ Consultation Services

How to Connect

Contact us at: ospo@utlists.utexas.edu

Visit our website: www.opensource.utexas.edu (or us QR code below)

Sign up to receive emails: https://utlists.utexas.edu/sympa/subscribe/ospo.announce
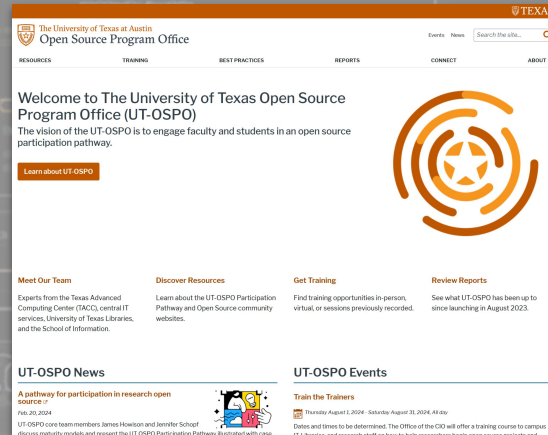
Upcoming Fall Workshops & Events

Data & Donuts

What	Workshops covering research data practices and software
When	On the following Fridays: 9/13, 9/27, 10/11, 10/25
Time	12pm - 1:15pm
Where	Zoom (all dates) / PCL Scholars Lab (select dates)
More info	https://guides.lib.utexas.edu/data-and-donuts

UT GIS Day

What	Celebrate and learn about GIS and geospatial research at UT
When	Wednesday, November 20th, time TBA
Where	PCL Scholars Lab
More info	https://guides.lib.utexas.edu/gis/ut-gis-day

UNIVERSITY OF TEXAS LIBRARIES

Scholars Lab Fellowship Program

Announcing the inaugural, paid, two-semester program for UT graduate students. Up to five Fellows will be selected to complete projects involving data, digital collections, digital media, or digital methods/platforms.

Fellows receive:

- Project support through consultation with research librarians and staff experts
- Mentorship, cohort-building and related professional development opportunities
- Stipend of $3,000 to focus on accepted proposed project work

Applications accepted:
August 12 - September 15, 2024

Learn more: tinyurl.com/scholarslabfellows

UTL Map & Geospatial Collections Explorer Fellowship

➔ Launched in 2021, this annual Fellowship incentivizes and rewards use of UT Libraries geospatial collections

➔ The 2024 call for proposals will be released August 26th and will be open through **Monday 10/7/2024**

➔ All current **UT students, faculty, & postdocs** are eligible

➔ Up to two fellowships awarded at **$1500** each

➔ Winners will be announced at a campus event on UT GIS Day on Wednesday, November 20th

➔ Visit https://guides.lib.utexas.edu/gis/utl-map-and-geospatial-collection-explorer-fellowship for the latest updates

Workshop Wrap Up

➜   Thank you for attending!

➜   Any questions?

➜   Please provide Day 2 feedback in the shared notes document