Tutorial for Jupyter Notebook and GitHub

ECS 171- Discussion 1 by Ziwen Kan
Slides adapted from Devashree Kataria

Jupyter Notebook

- Jupyter Notebook is an open-source web-based tool for creating, running, and sharing live code, equations, visualizations, and narratives all in one document.

- It supports over 40 programming languages, including popular ones such as Python, R, Julia, and Scala, making it a versatile tool for a wide range of data analysis, machine learning, and scientific computing tasks.

- Python notebooks saved with extension .ipynb

Jupyter Notebook: Interface

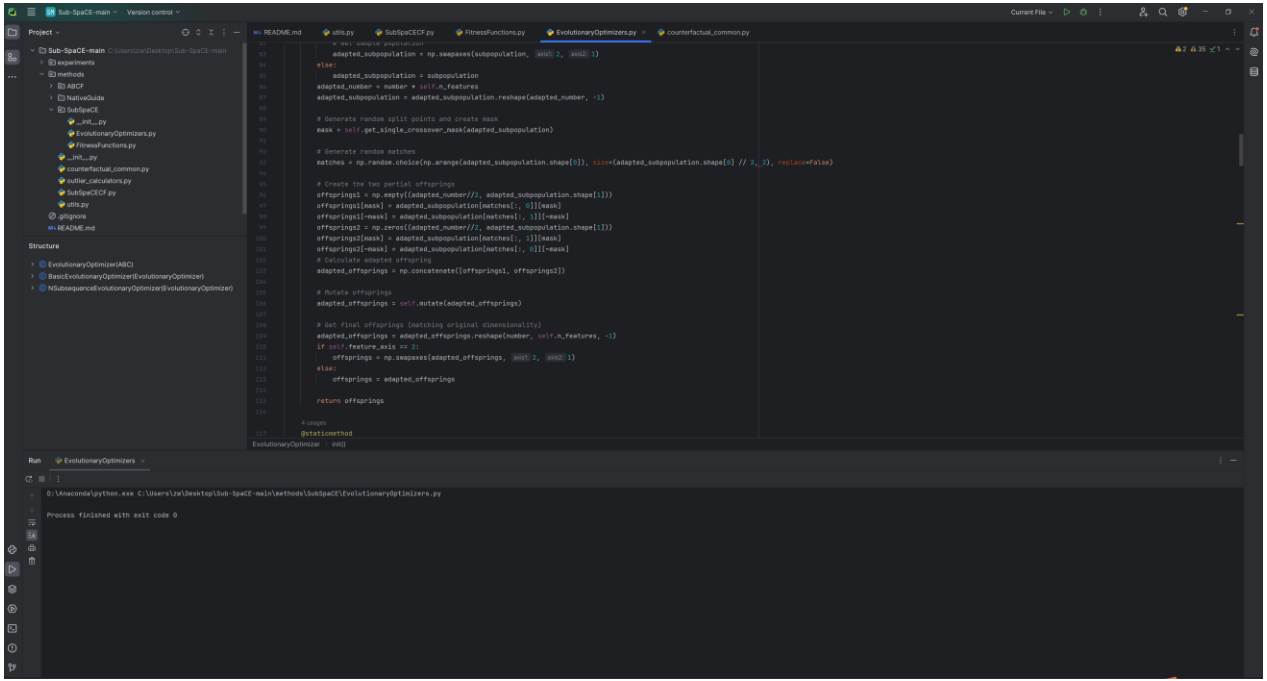
- The user interface
  - markdown cells for text-based writing
  - Primary notepad section where users can create and run code cells.

- Markdown: A powerful and flexible Markdown editor for creating formatted text, equations, and even HTML content, allowing users to create professional-looking narratives. (shortcut: esc + m)

- Code: Jupyter Notebook has a "cell" concept that allows for independent execution of each cell, making it easy to experiment and iterate on code and see the results in real time. (shortcut: esc + y)

- Notebooks can be easily saved, shared, and exported to different formats such as HTML, PDF, or slides, allowing for easy collaboration and presentation of results.

Interface of Pycharm



Interface of Jupyter Notebook

Markdown Cell Demo

Heading 1

Heading 2

Heading 3

Heading 4

I'm **bold text**.

I'm *italicized text*.

> Blockquotes are here!

Hyperlink: Canvas Page.

```
# Code highlighting
print('Hello, world')
```

Formula:

- $\sum_{x=a}^{b} f(x)$
- $\left(\int_a^b f(x)\,dx\right)$

Code Section

```
print('Hello, world!')
```

```
Hello, world!
```

```
!echo "Hello, world!"
```

```
"Hello, world!"
```

```
import pandas as pd
data = pd.read_csv('data.csv')
data
```

	cache-misses_1s	node-loads_1s	branch-misses_1s	branch-load-misses_1s	LLC-store-misses_1s	branch-loads_1s	L1-dcache-stores_1s	L1-icache-load-misses_1s	branch-instructions_1s
0	37514691	2822421	10070509	10559671	4972315	875088331	661114306	57139716	825641176
1	49901539	3999875	15983066	15905409	7812096	897853137	756436026	73999248	983561267
2	50400281	3256093	15329054	15610305	7440166	909703981	734120225	70138841	948103570
3	33857600	2281202	12144856	12142687	6133250	898523534	722133401	58023478	924673885
4	48650176	3510185	16656488	16676601	6020216	938655692	790604661	69916162	1039331799
...	...	...	...	...	...	...	...	...	...
2995	30437560	2326933	12585063	12241054	5041214	961290381	807913282	50286201	1012821501
2996	35121159	2576033	14087715	13595659	5959816	993575253	861158899	54518991	1050605365
2997	34968122	2696992	14058112	13734181	4591435	955564749	820924827	61481334	1047588424
2998	29050123	2382281	11980752	12213550	5229134	965928905	788854210	62116174	960454788
2999	31203289	2482180	12597138	12576162	4960137	1000022539	835397449	53078400	1000095628

3000 rows × 81 columns

Jupyter Notebook: Installation

- With Anaconda
  - Can use any Python library already available in Anaconda.

    Follow this for the course.

- Without Anaconda
  - You may have to install all required python libraries separately.

    Not recommended for this course

Jupyter Notebook: Installation with Anaconda

- Step 1: Download and Install Anaconda
  - To download installation file: https://www.anaconda.com/products/individual
  - For the steps further: https://docs.anaconda.com/anaconda/install/
- Step 2: Launch Anaconda Navigator
  - You can find it in your applications or search for it in your system's search bar.
- Step 3: Launch Jupyter Notebook
  - jupyter notebook will be present in the navigator with a launch button next to it.

Note: This process can also be done from the terminal or the anaconda prompt .

Jupyter Notebook: Installation without Anaconda

- Step 1: Install Python
  - https://www.python.org/
- Step 2: Install pip
  - pip --version
  - If not installed then follow the steps on https://pip.pypa.io/en/stable/installation/
- Step 3: Install Jupyter Notebook
  - pip install jupyter
- Step 4: Launch Jupyter Notebook
  - jupyter notebook
- Step 5: Create a Notebook

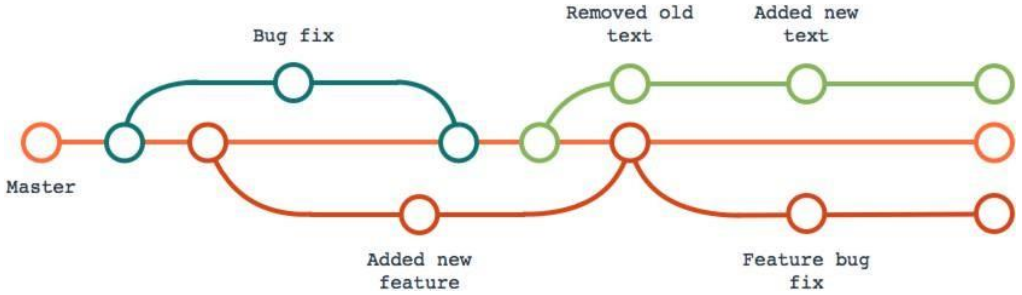
Jupyter Notebook: Basics

- Code Cells: In a Jupyter Notebook, code cells are where you create and execute programs. Simply select on a code block and hit Shift + Enter to execute it, or pick the "Run" icon in the menu.
- Markdown Cells: Markdown cells let you arrange text using the Markdown syntax. With the help of the simple markup language known as markdown, you can make headers, lists, tables, connections, and more. Markdown cells are helpful for explaining or adding notes to your code.
- Code execution: Jupyter Notebook enables you to run code units sequentially. Below the code block, the code's result is visible. For the latest findings, edit the code cells and execute them again. Additionally, Jupyter Notebook offers the ability to resume the kernel, clear output, and execute all cells simultaneously.
- Visualization: Using well-known data visualization tools like Matplotlib, Seaborn, and Plotly, Jupyter Notebook enables you to build visualizations, such as plots, charts, and graphs, immediately within the notebook. It is possible to incorporate visualizations within Markdown cells or show them as output in code cells.

Version Control

- Version control systems are a category of software tools that helps in recording changes made to files by keeping a track of modifications done in the code.
- Every contributor to the project has their own branch, and the changes aren't integrated into the primary source code until every branch has been examined and received the all-clear.
- In addition to keeping source code orderly, it increases efficiency by streamlining the development process.

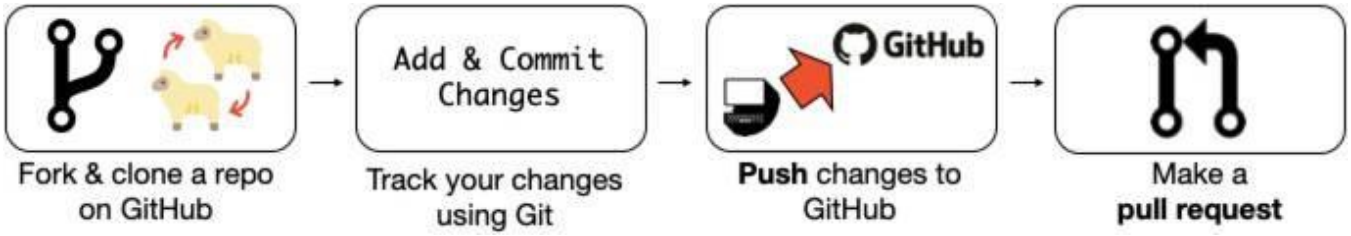
Version Control



Version control offers several benefits:

- The ability to track changes made by multiple collaborators
- Revert to previous versions of files or projects
- Resolve conflicts between different versions, and maintain a history of changes for auditing and debugging purposes.
- It also enables you to work on different branches of a project simultaneously and merge changes seamlessly.

GitHub



The most widely used distributed version control system among coders globally is called Git.

It has robust features like branching, merging, and rebasing and enables you to build and administer repositories both locally and online.

Additionally, Git is integrated with well-known code hosting services like GitHub, GitLab, and Bitbucket, which offer extra features like issue management, code review, and teamwork tools.

Installing git

https://git-scm.com/book/en/v2/Getting-Started-Installing-Git

GitHub Commands:

Git init: Creates a new.git directory in the current directory to hold the setup, information, and objects of the new Git file.

Git clone: Generates a local duplicate of a remote file. Formula: git clone

Git add: Adds files or modifications to the staging area, which serves as an interim holding location for modifications before they are committed to the repository. git add or git add is the syntax. (to add all changes)

Git status: Indicates which items in the repository have been changed, staged, or are untracked.

Git commit: Logs modifications to the source along with a commit statement outlining the modifications. git commit -m "commit message" is the syntax.

GitHub Commands:

Git merge: Combines modifications from one branch with those from another. Git merge syntax is branch_to_merge.

Git pull: Merges modifications made in a remote source into the active branch. Git pull "remote repository" "branch to pull"

Git push: Transfers modifications from a local source to a remote version. Git push syntax: remote_repository, branch_to_push

Resources:

Anaconda :

- https://docs.anaconda.com/anaconda/install/mac-os/
- https://docs.anaconda.com/anaconda/install/

Jupyter Notebook:

- https://jupyter-notebook-beginner-guide.readthedocs.io/en/latest/
- https://jupyter-notebook.readthedocs.io/en/stable/

GitHub:

- https://www.edureka.co/blog/how-to-use-github/
- An interactive interface to learn Git https://learngitbranching.js.org/?locale=en_US

Cheatsheet:

Conda Cheatsheet: https://docs.conda.io/projects/conda/en/4.6.0/_downloads/52a95608c49671267e40c689e0bc00ca/conda-cheatsheet.pdf

Jupyter notebook Cheatsheet: https://www.edureka.co/blog/wp-content/uploads/2018/10/Jupyter_Notebook_CheatSheet_Edureka.pdf

Git Cheatsheet https://education.github.com/git-cheat-sheet-education.pdf

THANK YOU!!