

# Software Environment Setup

Machine Learning in Water and Environmental Modeling  
Pre-Workshop Setup Meeting, May 1, 2025

**Brad Tom** and Peyman Namadi  
DWR Delta Modeling Section

# Part A: Local Computer

# Prerequisites

- Laptop or Desktop with Windows\*
- 6GB free space
- Administrative privileges

\* The ML code will also run on Mac OS, Linux, and Google Colab

# Overview

- Install Miniconda<sup>1</sup>
- Create a conda environment<sup>2</sup>
- Download ANN code and input data files
- Run the Jupyter Notebook<sup>3</sup> ANN code

1. Anaconda is a free, open-source platform that allows you to write and execute code in the Python programming language. Miniconda is a lightweight version of Anaconda. We recommend Miniconda because installation and setup take less time.

2. A conda environment is a folder containing a version of Python, and a specific set of python packages

3. Jupyter Notebook is a web application for creating and sharing computational documents

# Install Miniconda/Anaconda


- Miniconda or Anaconda:  
<https://www.anaconda.com/download/success>

# Downloading the Miniconda Installer

https://www.anaconda.com/download/success

**ANACONDA** | Products Solutions Resources Partners Company [Sign Up](#)


## Miniconda Installers



### Windows

Python 3.12


↓ 64-Bit Graphical Installer



### Mac

Python 3.12

- ↓ 64-Bit (Apple silicon) Graphical Installer
- ↓ 64-Bit (Apple silicon) Command Line Installer
- ↓ 64-Bit (Intel chip) Graphical Installer
- ↓ 64-Bit (Intel chip) Command Line Installer



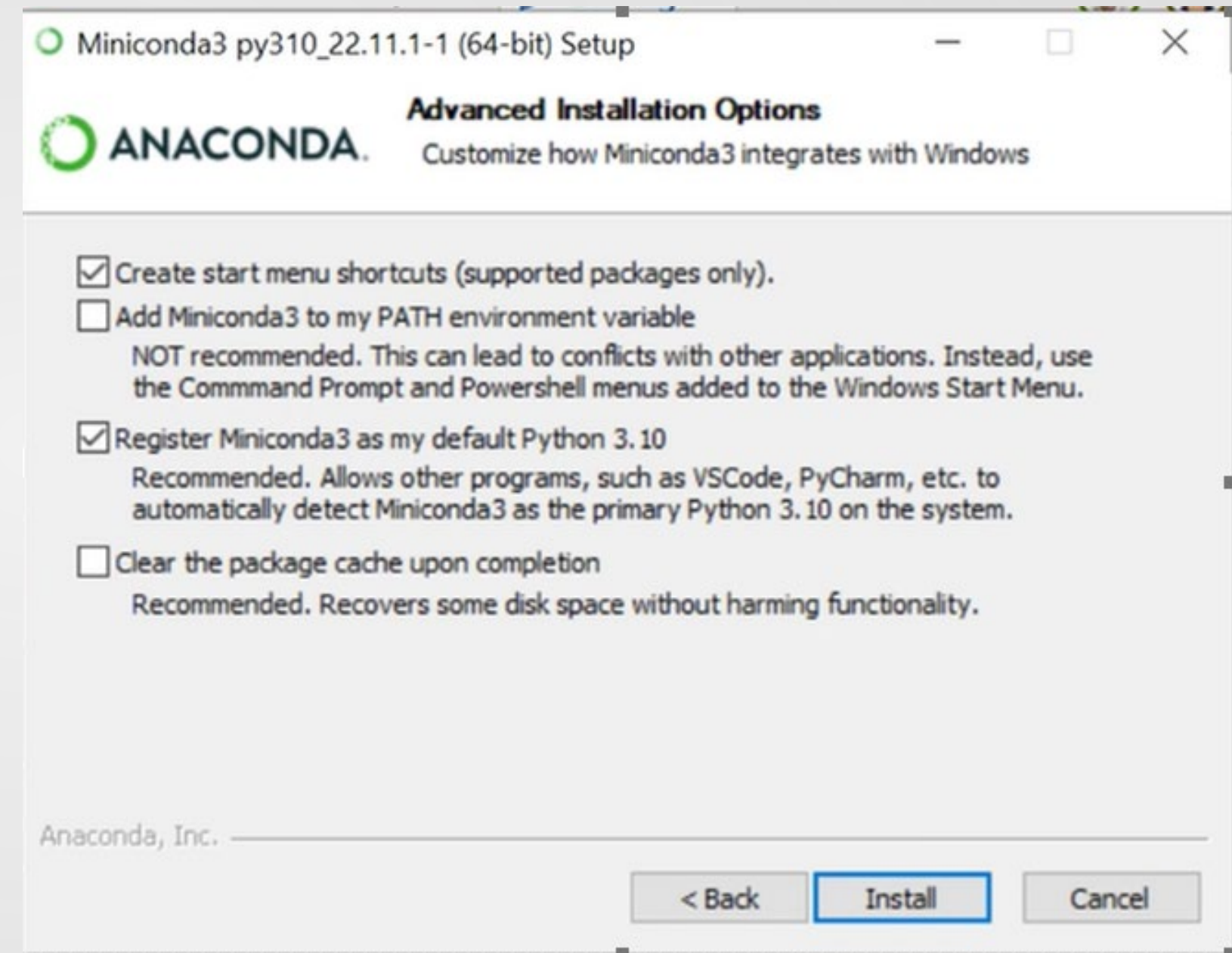
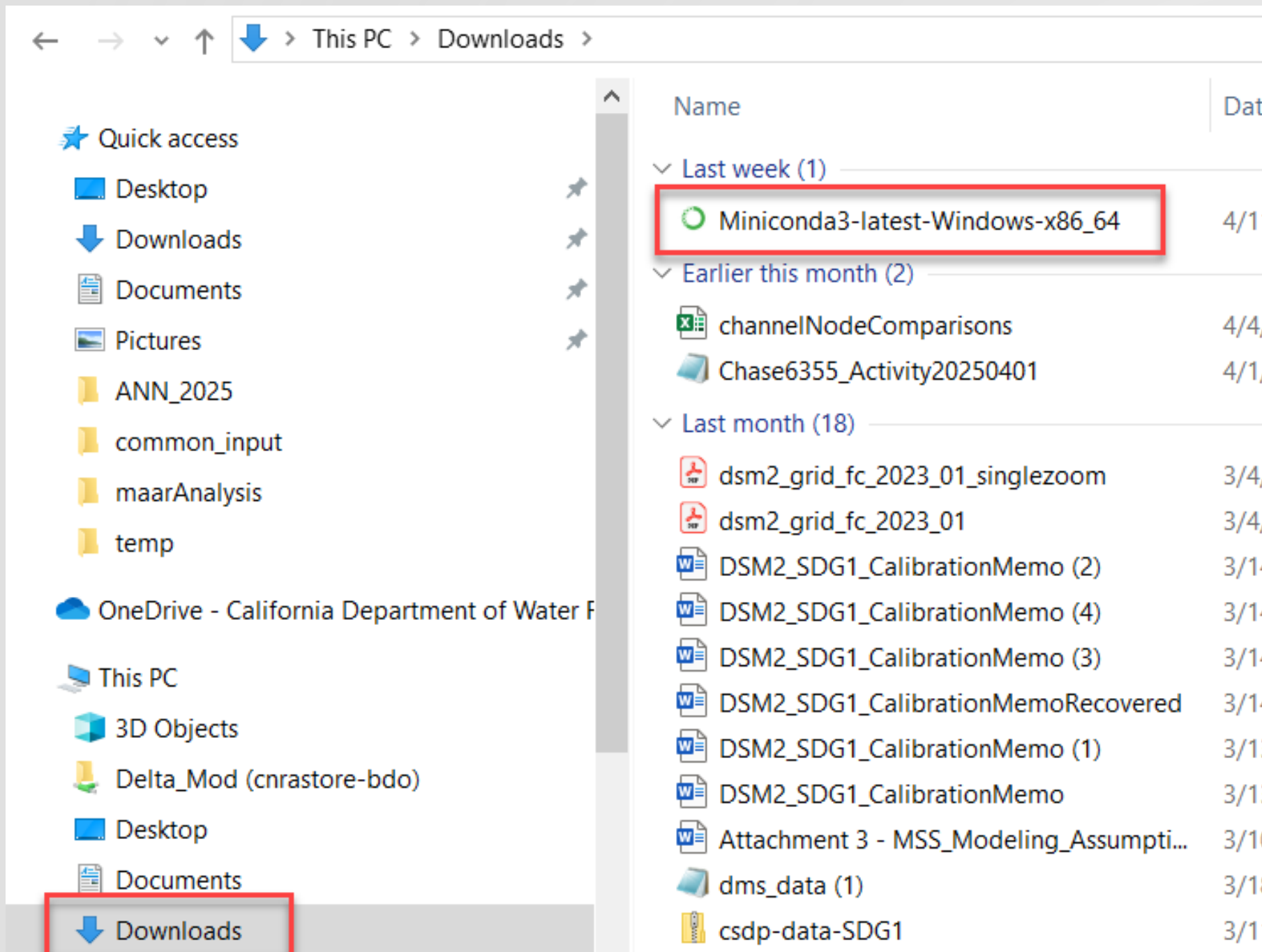
### Linux

Python 3.12

- ↓ 64-Bit (x86) Installer
- ↓ 64-Bit (AWS Graviton2 / ARM64) Installer
- ↓ 64-bit (Linux on IBM Z & LinuxONE) Installer



# Install Miniconda



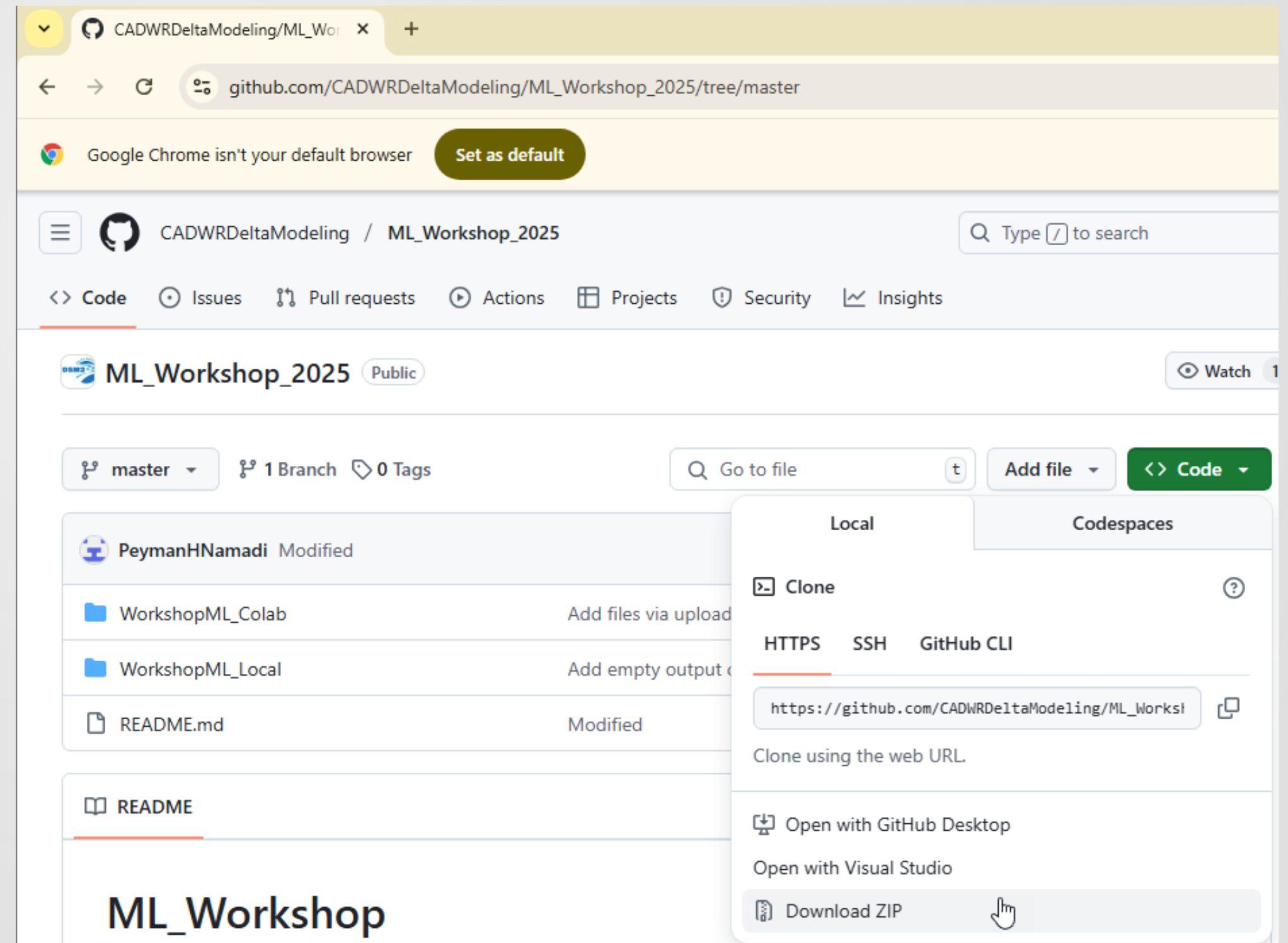
# Download ML Code/Data

a) If you don't have Git\* installed

- [https://github.com/CADWRDeltaModeling/ML\\_Workshop\\_2025/](https://github.com/CADWRDeltaModeling/ML_Workshop_2025/)

Click “Code”, then “Download ZIP”

Git\* is a free and open source distributed version control system

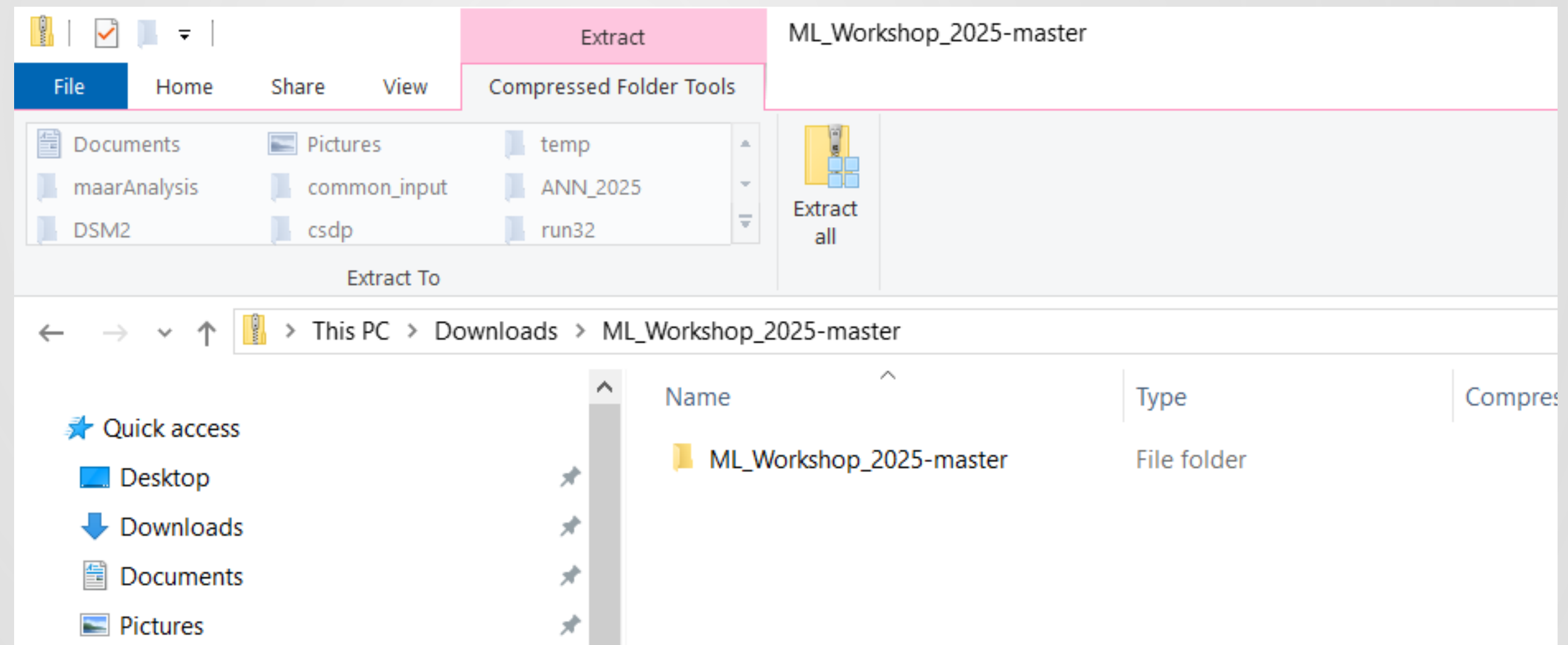




# Download ML Code/Data

a) If you don't have Git installed (cont)

- Double click the zip file
- Copy and paste the folder inside to another location

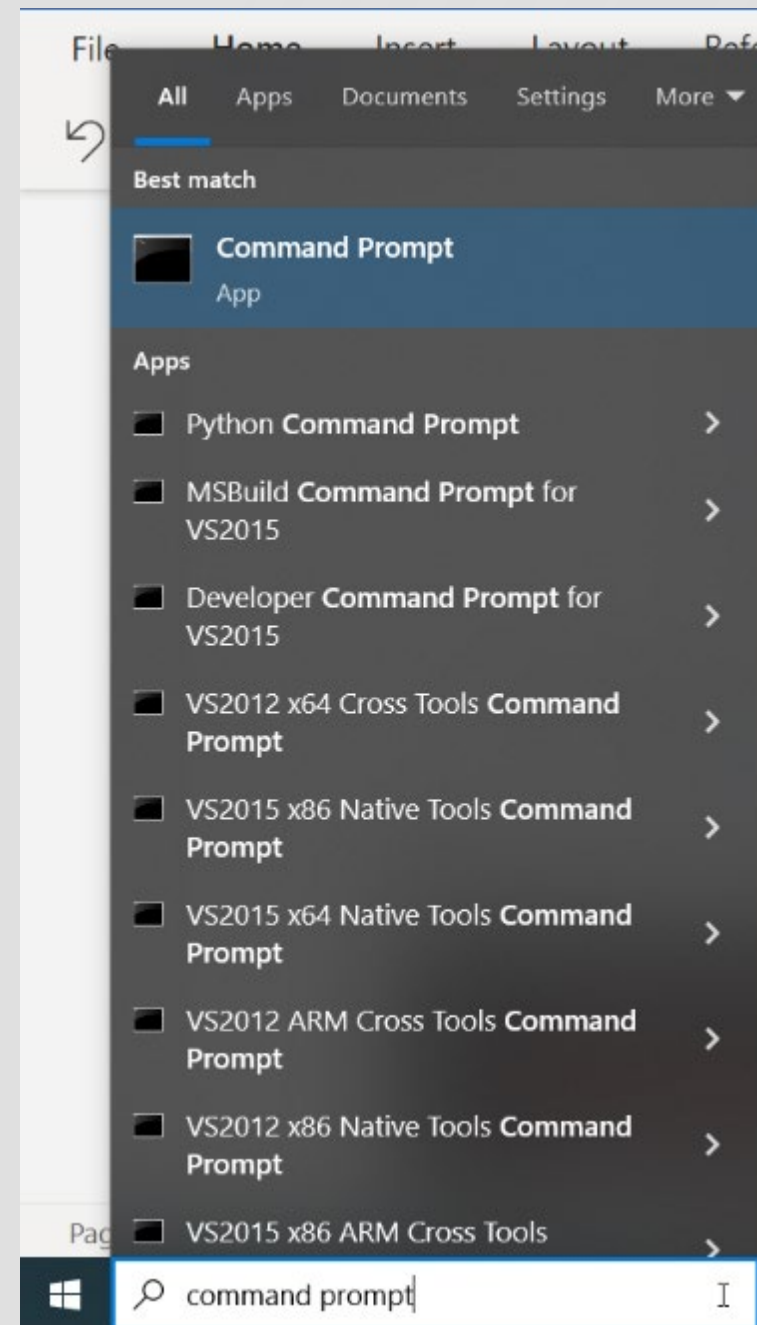


# Download ML Code/Data

b) If you have Git installed

Open a command prompt window

Navigate to the location where you want to save files



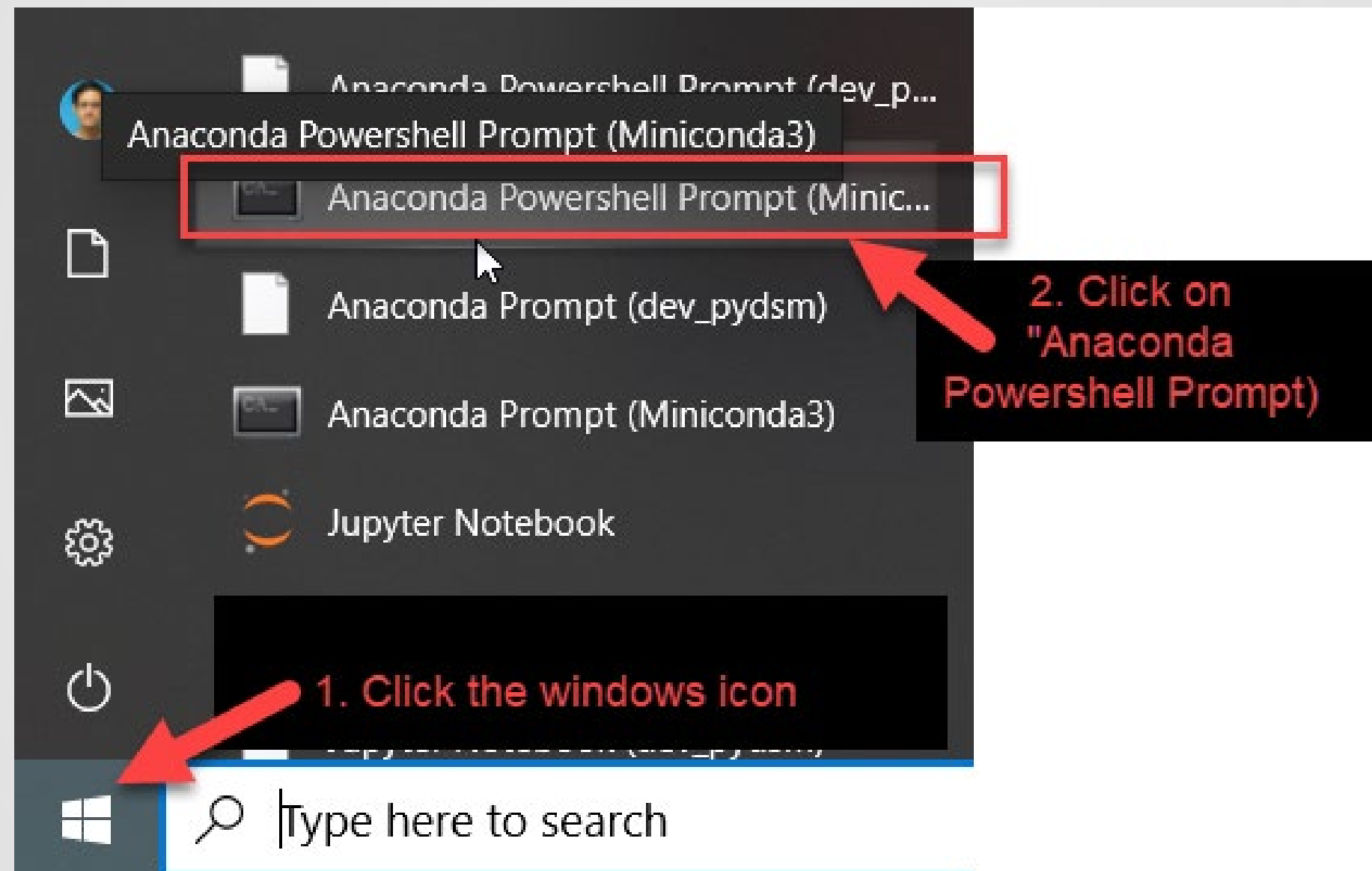
```
Select Anaconda Prompt (miniconda3)

(base) D:\temp>git clone https://github.com/CADWRDeltaModeling/ML_Workshop_2025
Cloning into 'ML_Workshop_2025'...
remote: Enumerating objects: 51, done.
remote: Counting objects: 100% (51/51), done.
remote: Compressing objects: 100% (28/28), done.
remote: Total 51 (delta 20), reused 51 (delta 20), pack-reused 0 (from 0)
Receiving objects: 100% (51/51), 1.73 MiB | 19.91 MiB/s, done.
Resolving deltas: 100% (20/20), done.

(base) D:\temp>
```

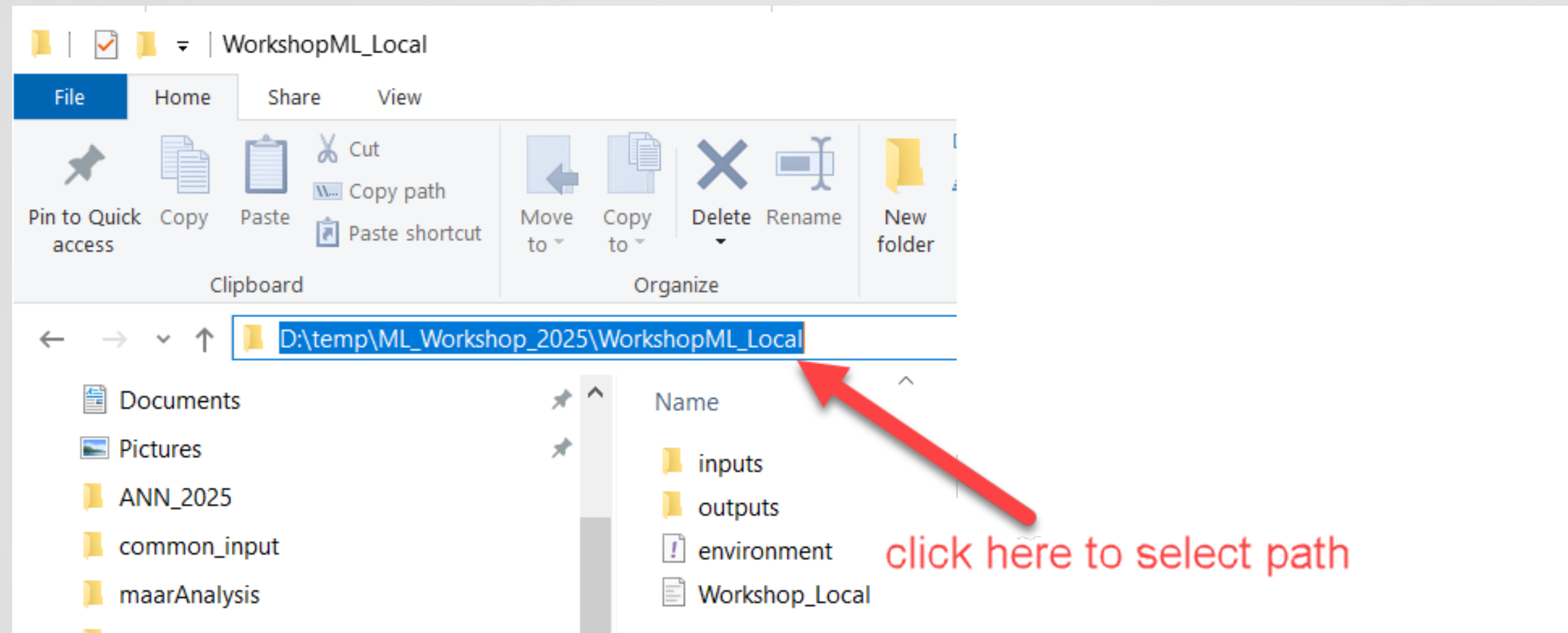
```
git clone https://github.com/CADWRDeltaModeling/ML_Workshop_2025
```

# Open an Anaconda Powershell prompt



# Create a conda environment

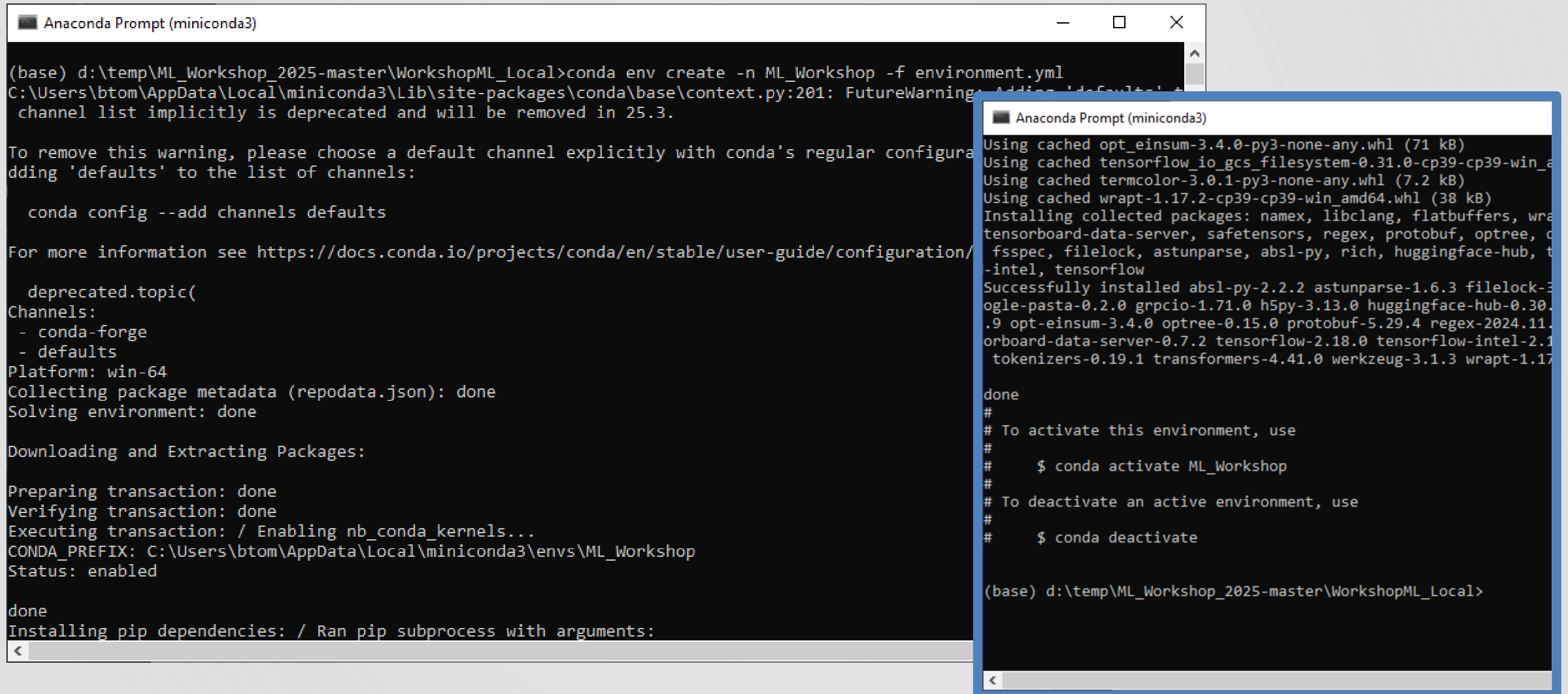
- Use the “cd” command to navigate to the folder containing your code/data



```
Anaconda Prompt (miniconda3)
(base) D:\temp>cd d:\temp\ML_Workshop_2025-master\WorkshopML_Local
(base) d:\temp\ML_Workshop_2025-master\WorkshopML_Local>_
```

# Create a conda environment

- conda env create -n ML\_Workshop -f environment.yml



The image displays two overlapping screenshots of an Anaconda Prompt terminal window. The background window shows the execution of the command `conda env create -n ML_Workshop -f environment.yml`. It includes a warning about the deprecated `channel list` and instructions on how to remove it by adding 'defaults' to the channels. The terminal output shows the environment being created, with various packages being installed and dependencies being resolved. The foreground window, which is highlighted with a blue border, shows the output of the `conda env create` command, listing the packages being installed and the final status of the environment.

```
Anaconda Prompt (miniconda3)

(base) d:\temp\ML_Workshop_2025-master\WorkshopML_Local>conda env create -n ML_Workshop -f environment.yml
C:\Users\btom\AppData\Local\miniconda3\Lib\site-packages\conda\base\context.py:201: FutureWarning: Adding 'defaults' to the
channel list implicitly is deprecated and will be removed in 25.3.
  channel list implicitly is deprecated and will be removed in 25.3.

To remove this warning, please choose a default channel explicitly with conda's regular configuration
adding 'defaults' to the list of channels:

    conda config --add channels defaults

For more information see https://docs.conda.io/projects/conda/en/stable/user-guide/configuration/
deprecated.topic(
Channels:
- conda-forge
- defaults
Platform: win-64
Collecting package metadata (repodata.json): done
Solving environment: done

Downloading and Extracting Packages:

Preparing transaction: done
Verifying transaction: done
Executing transaction: / Enabling nb_conda_kernels...
CONDA_PREFIX: C:\Users\btom\AppData\Local\miniconda3\envs\ML_Workshop
Status: enabled

done
Installing pip dependencies: / Ran pip subprocess with arguments:

Anaconda Prompt (miniconda3)

Using cached opt_einsum-3.4.0-py3-none-any.whl (71 kB)
Using cached tensorflow_io_gcs_filesystem-0.31.0-cp39-cp39-win_a
Using cached termcolor-3.0.1-py3-none-any.whl (7.2 kB)
Using cached wrapt-1.17.2-cp39-cp39-win_amd64.whl (38 kB)
Installing collected packages: namex, libclang, flatbuffers, wr
tensorflow-data-server, safetensors, regex, protobuf, optree, c
fsspec, filelock, astunparse, absl-py, rich, huggingface-hub, t
-intel, tensorflow
Successfully installed absl-py-2.2.2 astunparse-1.6.3 filelock-3
ogle-pasta-0.2.0 grpcio-1.71.0 h5py-3.13.0 huggingface-hub-0.30.
.9 opt-einsum-3.4.0 optree-0.15.0 protobuf-5.29.4 regex-2024.11.
orboard-data-server-0.7.2 tensorflow-2.18.0 tensorflow-intel-2.1
tokenizers-0.19.1 transformers-4.41.0 werkzeug-3.1.3 wrapt-1.17

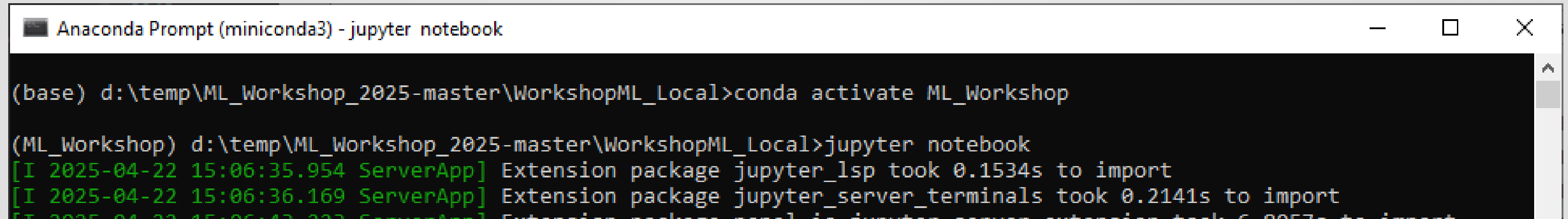
done
#
# To activate this environment, use
#
#     $ conda activate ML_Workshop
#
# To deactivate an active environment, use
#
#     $ conda deactivate
#

(base) d:\temp\ML_Workshop_2025-master\WorkshopML_Local>
```



# Open Jupyter Notebook

```
conda activate ML_Workshop  
jupyter notebook
```

A screenshot of an Anaconda Prompt window titled "Anaconda Prompt (miniconda3) - jupyter notebook". The window shows a terminal with the following text:

```
(base) d:\temp\ML_Workshop_2025-master\WorkshopML_Local>conda activate ML_Workshop  
(ML_Workshop) d:\temp\ML_Workshop_2025-master\WorkshopML_Local>jupyter notebook  
[I 2025-04-22 15:06:35.954 ServerApp] Extension package jupyter_lsp took 0.1534s to import  
[I 2025-04-22 15:06:36.169 ServerApp] Extension package jupyter_server_terminals took 0.2141s to import  
[I 2025-04-22 15:06:37.227 ServerApp] Extension package jupyter_server_extensions took 0.8057s to import
```

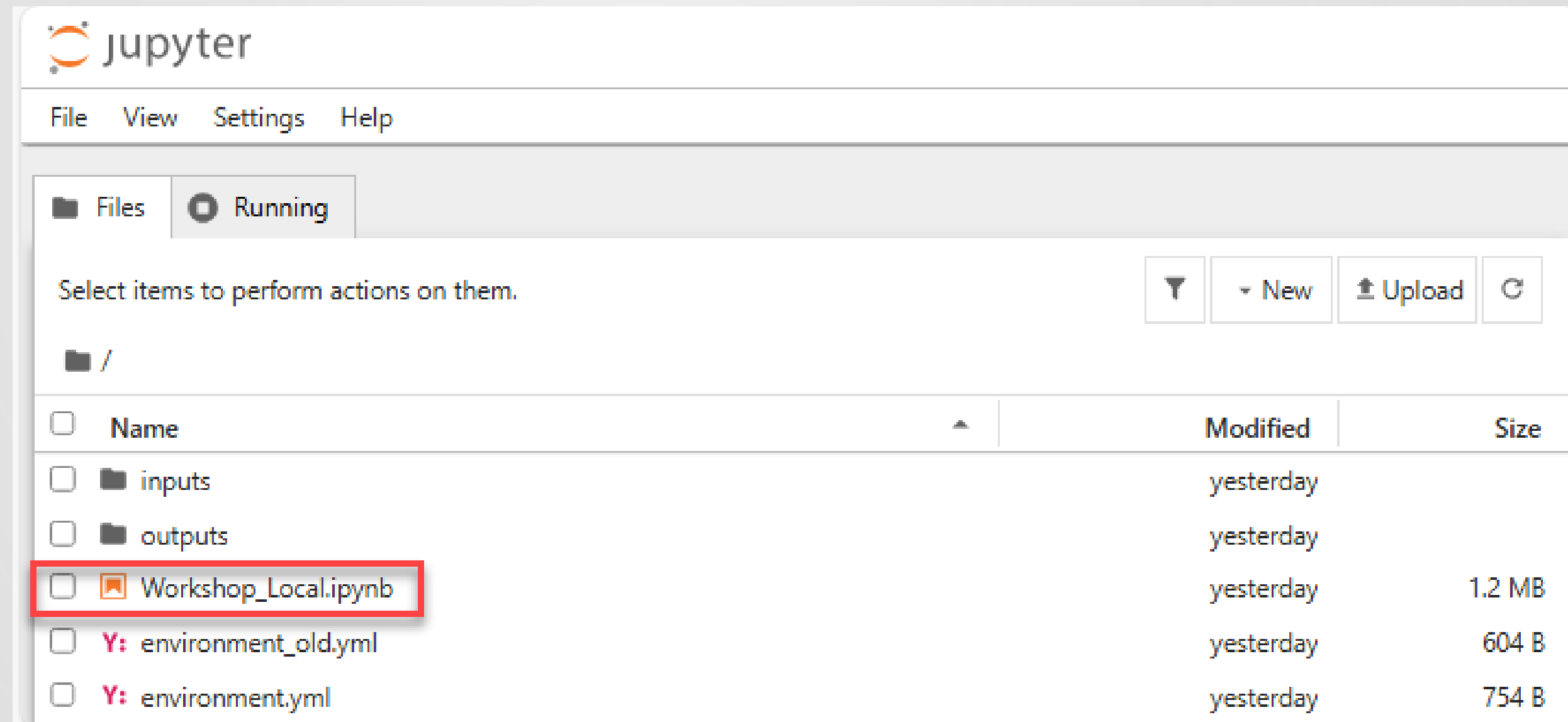
**If the above command fails, try these two commands:**

```
call conda activate jupyter  
start jupyter notebook
```

# Open the ANN Jupyter Notebook

click on

**Workshop\_Local.ipynb**



# Select notebook kernel

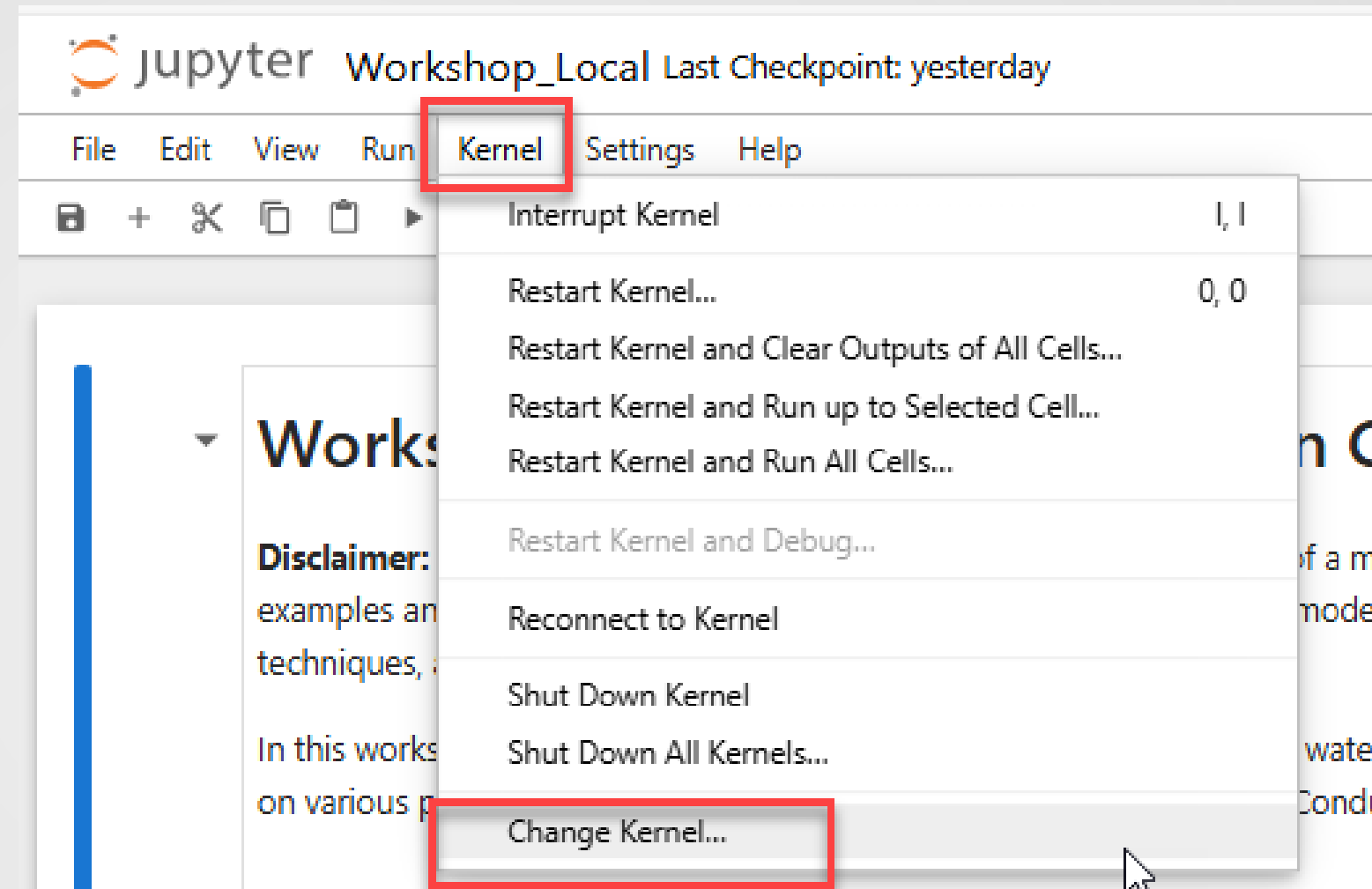
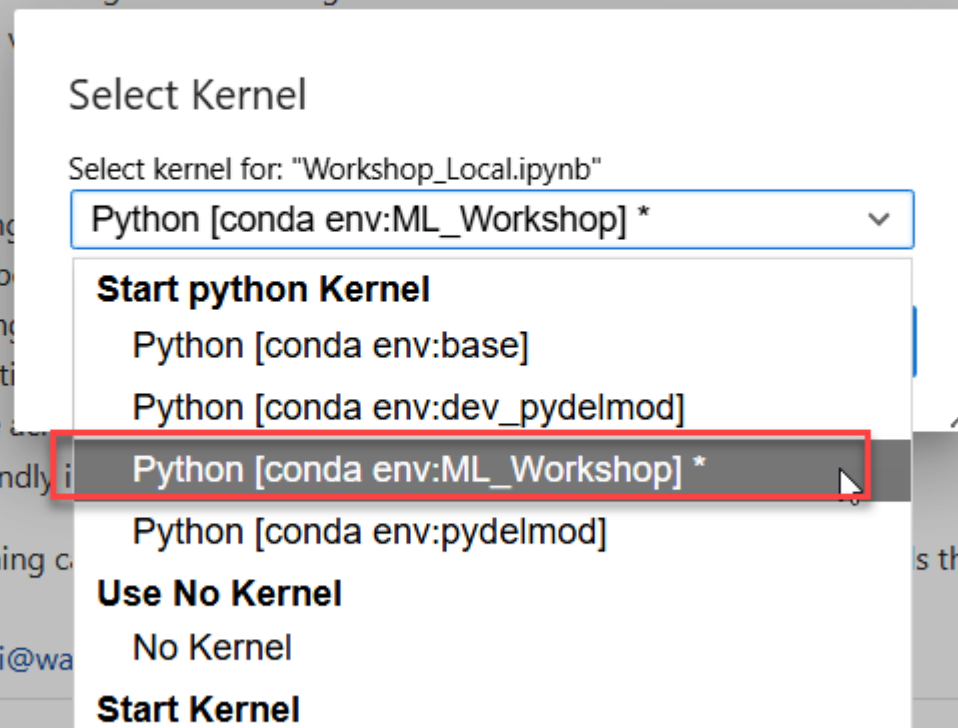
created for educational purposes as part of a machine learning workshop. The models and met  
optimized for production use. The actual models used in production environments may incorp  
ve validation.

rough a machine learning workflow using water ion data from the California Delta. We'll focus  
g both numerical v

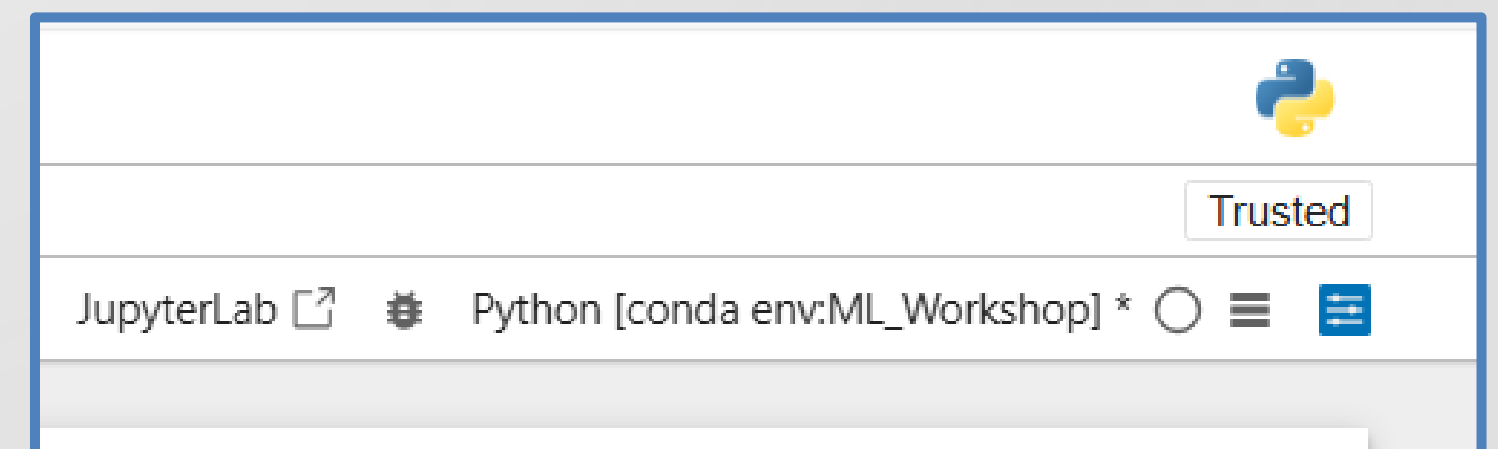
d initial processing  
ore relationships b  
engineering, scaling  
ent and train multi  
bare performance  
Create a user-friendly i

ow machine learning c  
**se contact:**  
seinzadehnamadi@wa

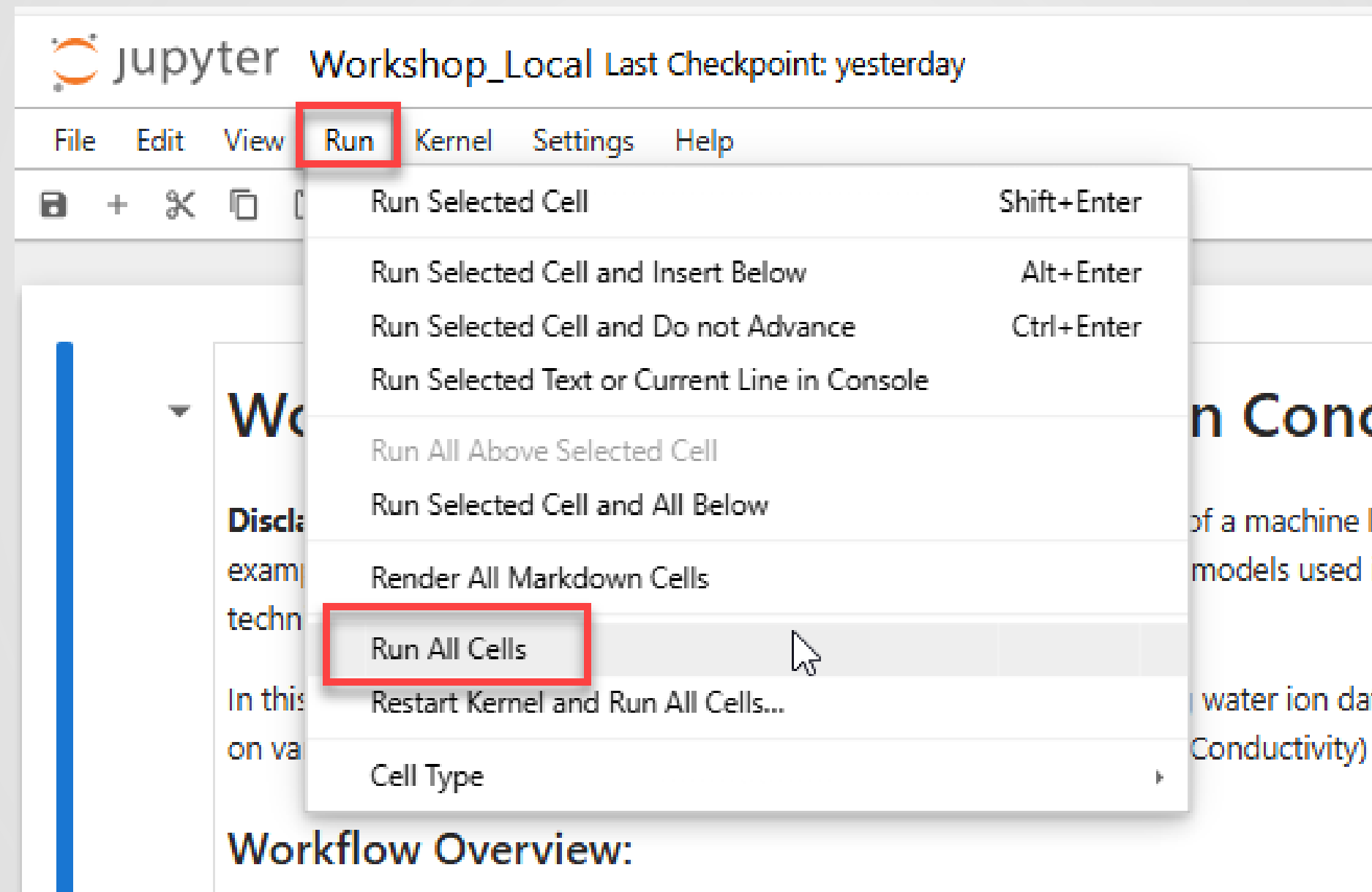
and Library Imports



A notebook kernel is a “computational engine” that executes the code contained in a Notebook document.



# Run the notebook code



# Run the notebook code

## Br Concentration Prediction Dashboard

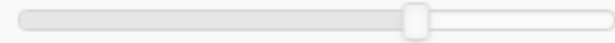
This dashboard uses three machine learning models to predict Br concentration based on environmental parameters. Adjust the sliders and dropdown menus to see how different conditions affect predicted Br levels.

### Input Parameters

Electrical Conductivity (EC) [ $\mu\text{S}/\text{cm}$ ]: 1000



Sacramento X2: 80



Location

OMR

Water Year Type (WYT)

AN

Month

Sep

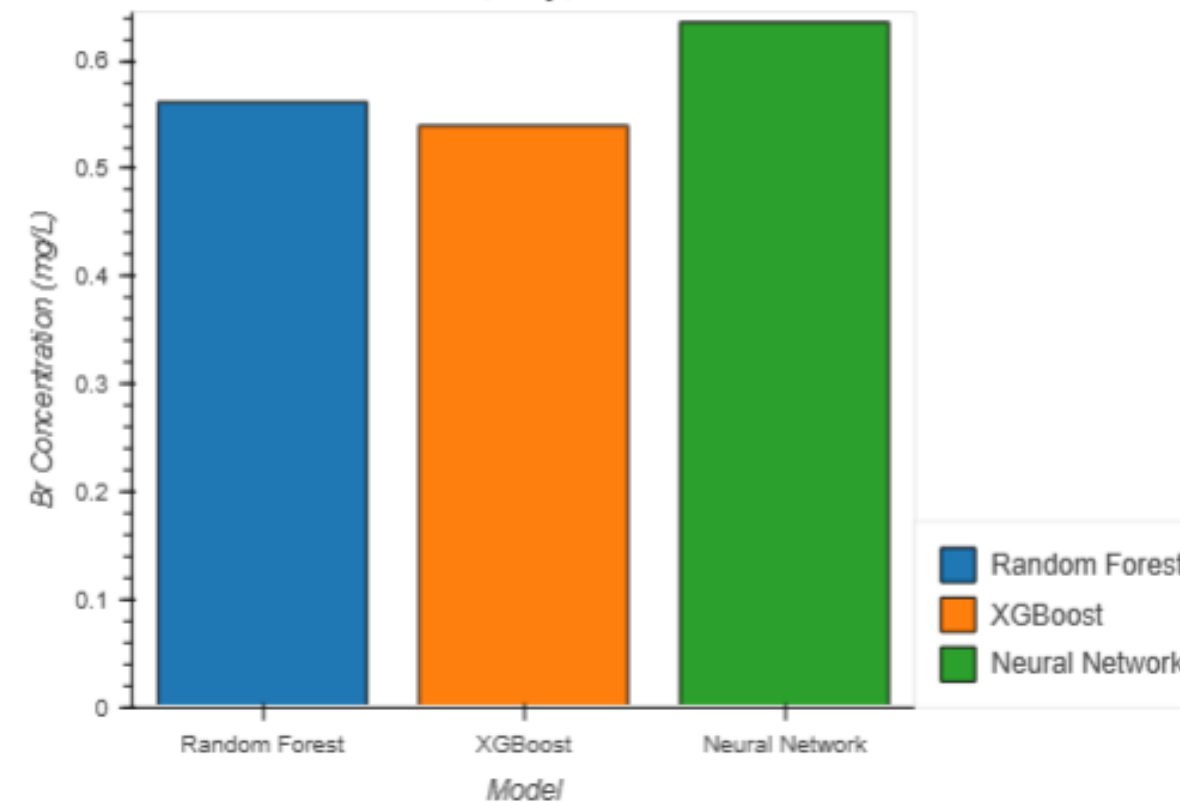
Predictions updated

### Prediction Results

#### Predicted Br Concentration

Model	Br (mg/L)
Random Forest	0.56
XGBoost	0.54
Neural Network	0.64

Predicted Br for OMR, Sep, WYT: AN





# Completed cell vs running cell

## Section 0: Setup and Library Imports

```
[1]: # ===== SECTION 0: IMPORT LIBRARIES =====  
# Core data handling and analysis  
import pandas as pd  
import numpy as np  
import os  
import joblib  
from datetime import datetime
```

## 1. Data Loading

```
[*]: import pandas as pd  
import os  
  
# Define the path to the inputs folder  
input_folder = 'inputs'  
  
# List of ion files to load
```

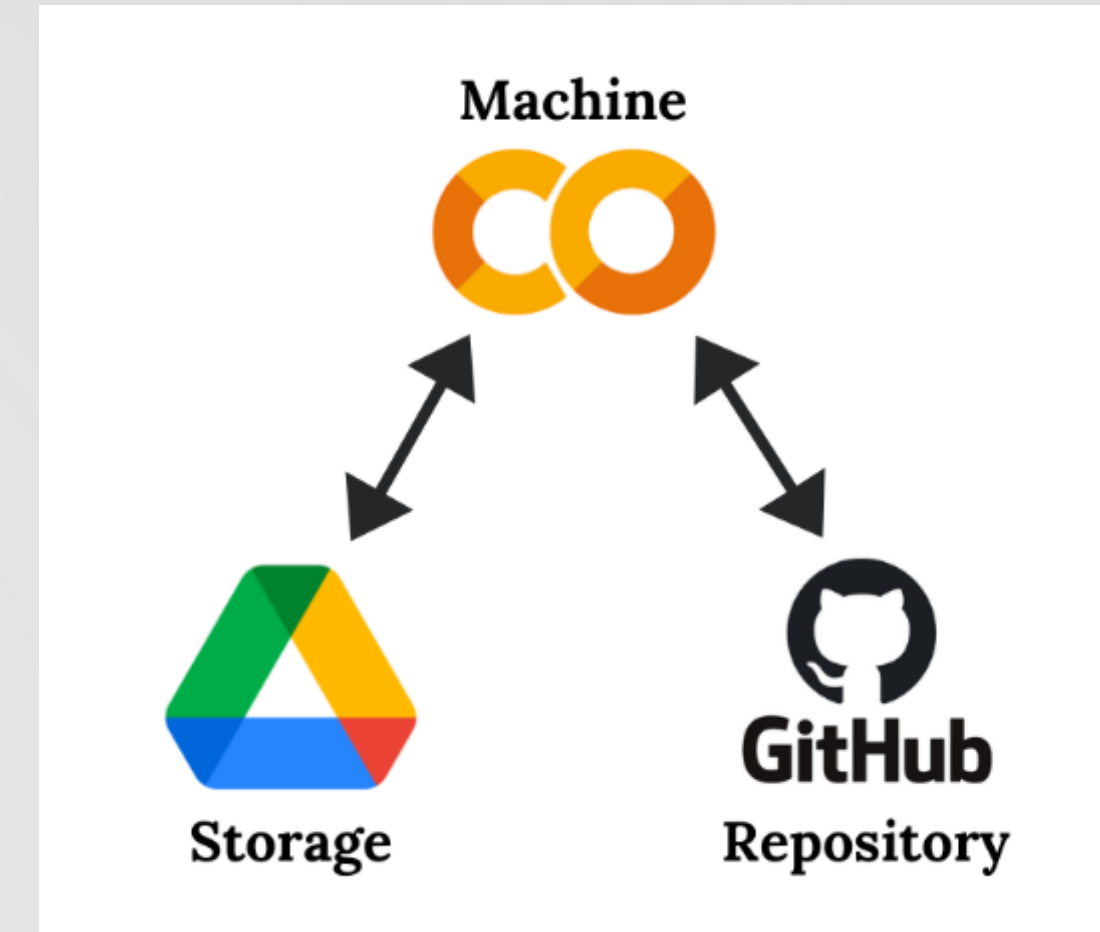
# Part B:

Google

colab

## Advantages of Google Colab:

- Free GPU/TPU access (to a certain limit)
- No setup required on your computer
- Real-time collaboration

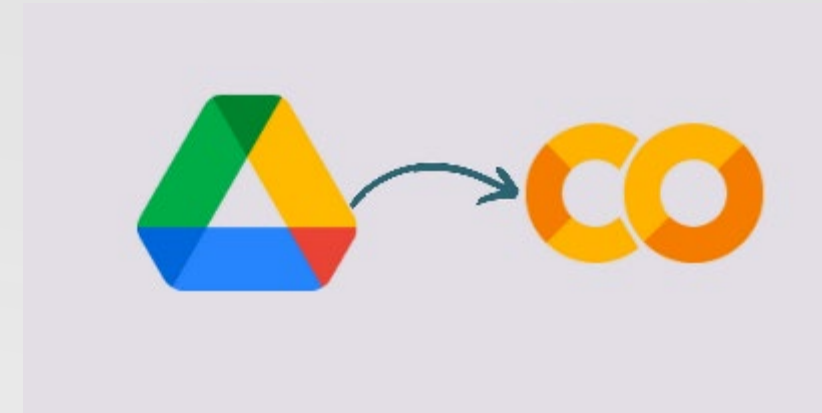


## Limitations of Google Colab:

- **Session time limits:** Free → 12-hour maximum runtime; paid tiers → 24 hours
- **Limited persistent storage:** Files not saved to Google Drive are deleted when sessions end
- **Inconsistent performance:** GPU/TPU availability and performance vary based on usage demand
- **Resource constraints:** Limited RAM (typically 12-25GB) & storage capacity, especially on free tier



**Step 1:** Create a **Google account** or use an existing Google account and log in to your **Google Drive** at <https://drive.google.com/drive/my-drive>

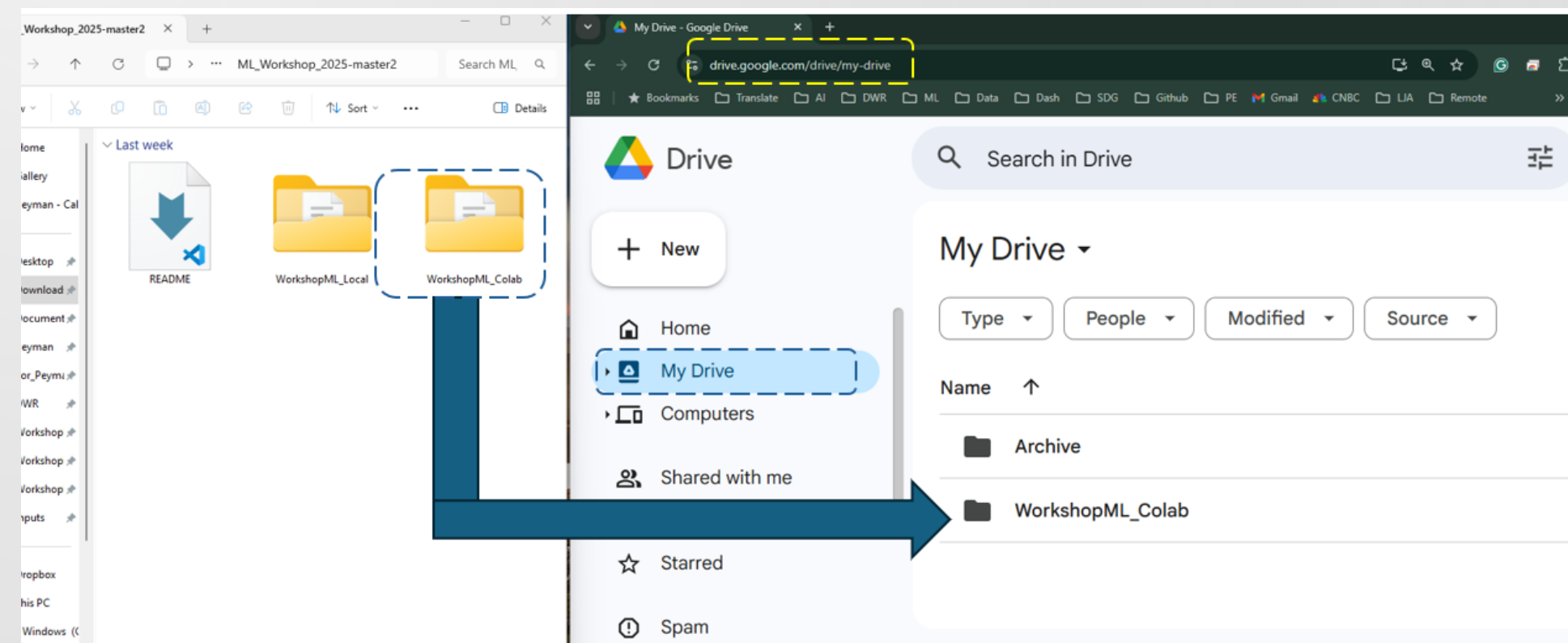


**Step 2:** Use this link to download the folder containing data and code:  
[https://github.com/CADWRDeltaModeling/ML\\_Workshop\\_2025/tree/master](https://github.com/CADWRDeltaModeling/ML_Workshop_2025/tree/master)



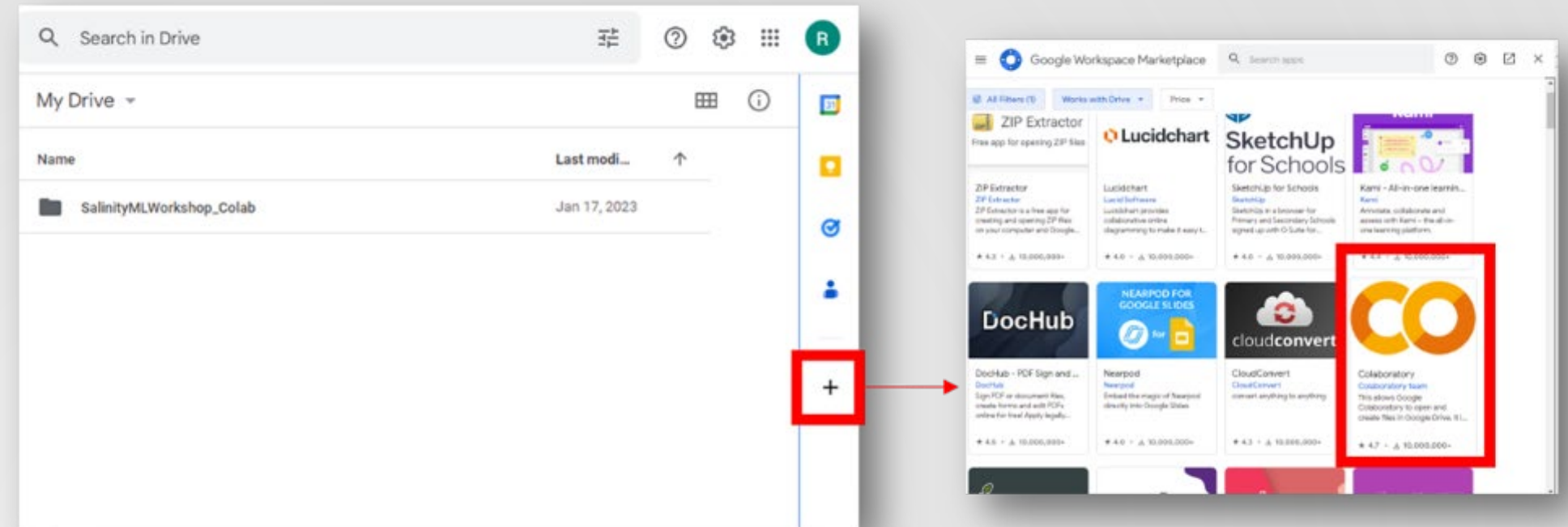
**Step 3:** Navigate to your download location and **unzip** the folder

**Step 4:** After unzipping the file, you'll see two folders: "WorkshopML\_Local" and "WorkshopML\_Colab". Take the "**WorkshopML\_Colab**" folder and upload it directly to the main area of your Google Drive (the "My Drive" section). Make sure you're uploading the actual folder with the workshop materials, not a parent folder that contains another folder with the same name.

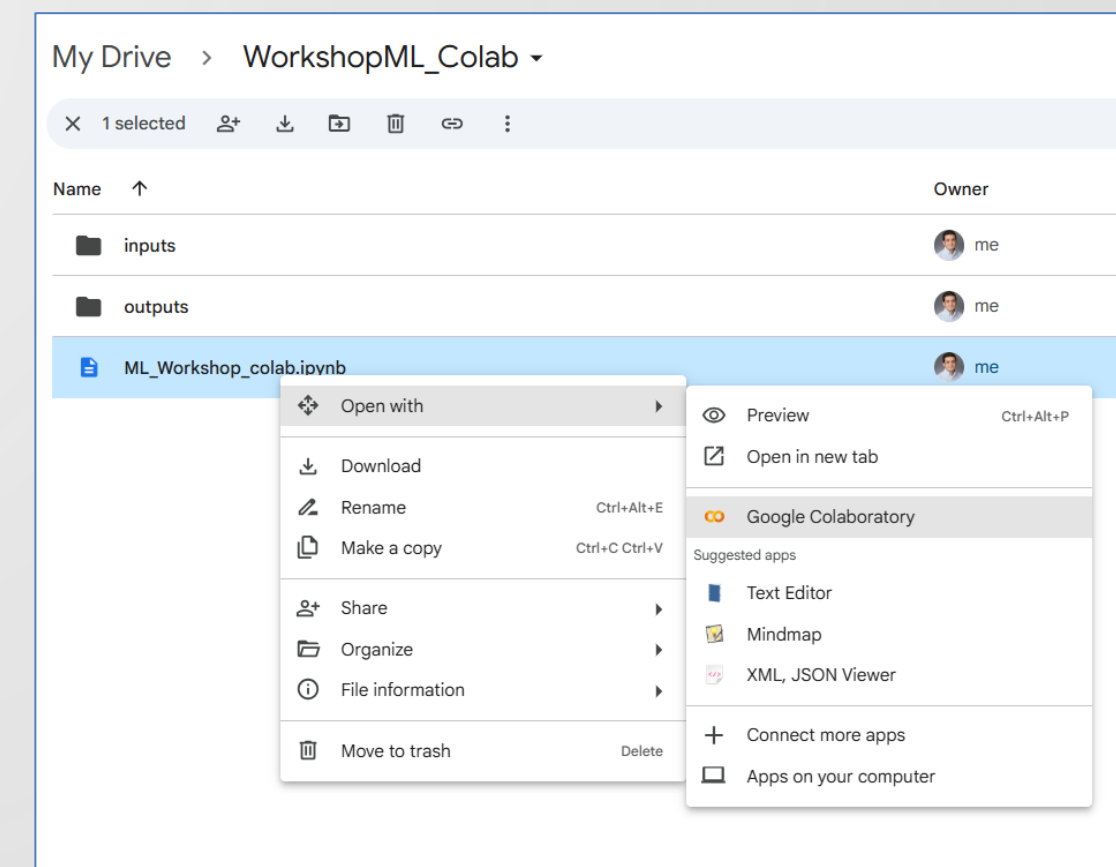




**Step 5:** Link the **Google Colab** Add-On to your Google Drive. See screenshot below. When complete, refresh your browser for the changes to take effect.

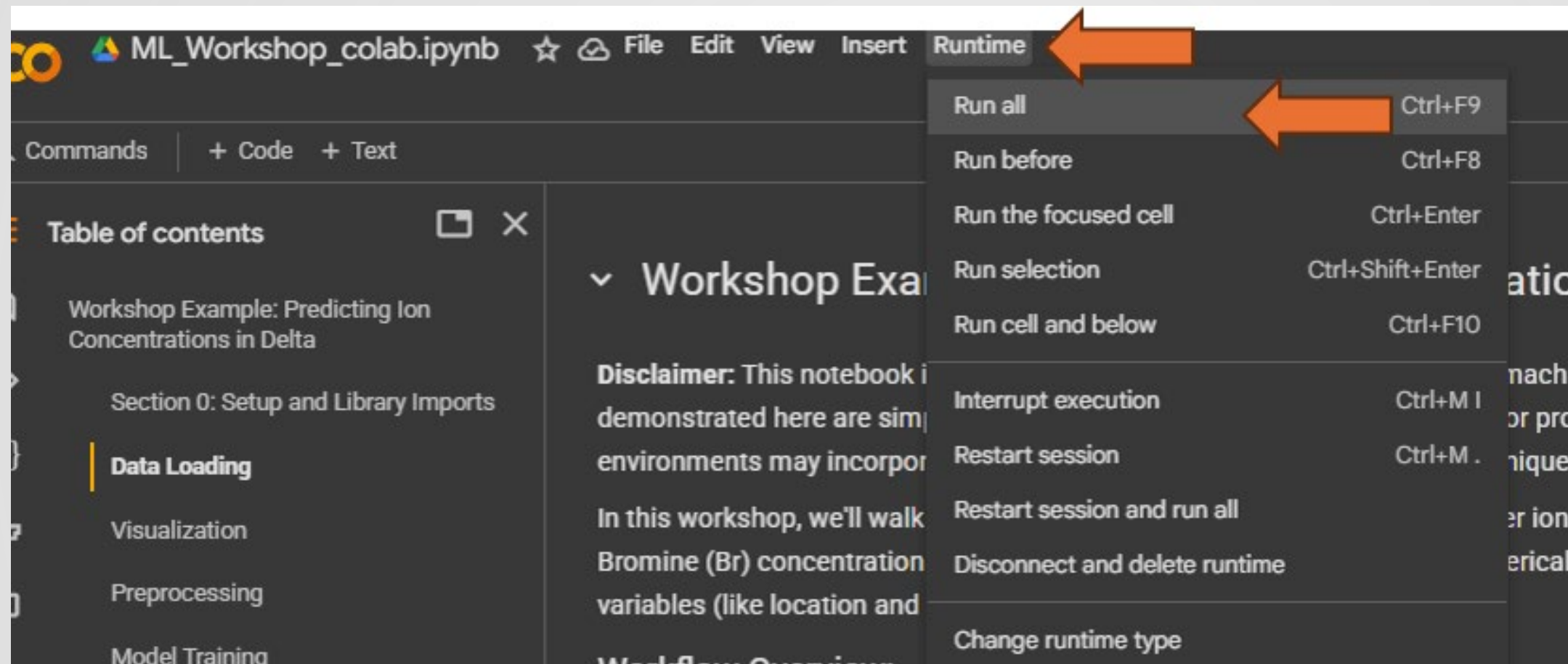


**Step 6:** Go into the “WorkshopML\_Colab” folder you uploaded and open *ML\_Workshop\_colab.ipynb*. Right-click and Open With the Google Colaboratory.





**Step 7:** With the script now open in Colab, run the script by clicking *Runtime* and *Run All*. The script takes about 2 minutes to complete. Note: upon launching the code, Google Colab will request permission to access the folder – follow the prompts to allow access.



# Questions?

