

[DM2025] Lab 1 – Environment Setup

Hi everyone,

We will have our first lab session (for Q&A) on **September 15 (Monday)** at **9:00 AM** via our [YouTube Stream/Classroom](#). Please be on time.

The Lab tutorial video will be launched on [YouTube Channel](#) before **September 15**.

We strongly recommend setting up the environment on your personal laptop before the lab session so that you can follow along smoothly.

This document provides detailed instructions for setting up the environment required for the course.

System Requirements

Please make sure you have the following installed:

- Python 3.11.0 (recommended)
 - uv (Python virtual environment manager)
 - Git
 - GitHub account
 - Jupyter Notebook
 - VS Code (optional)
-

Setup Instructions

1. Install Python 3.11.0 (If you use your own version, take your own risk of dependency issues)

Download and install Python 3.11.0 (recommended): <https://www.python.org/downloads/release/python-3110/>

During installation, check "**Add Python to PATH**".

Verify installation:

```
python --version
```

Expected output: Python 3.11.0

2. Create a GitHub Account and Install Git

Sign up for GitHub: <https://github.com/>

Install Git:

Windows: <https://gitforwindows.org/>

Linux:

```
sudo apt install git-all
```

macOS:

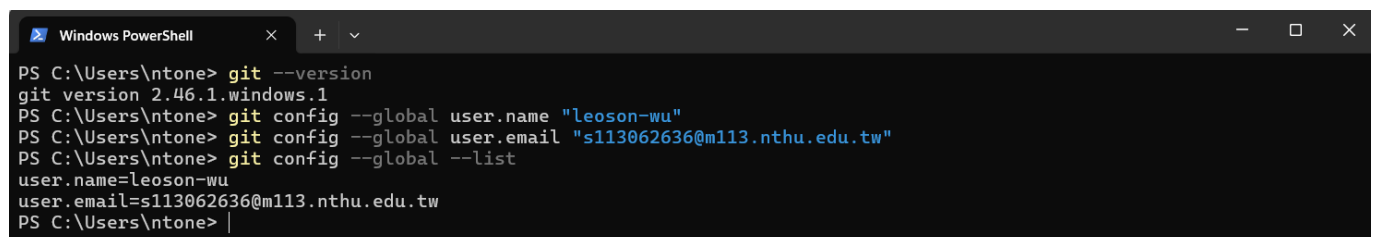
```
brew install git
```

Verify installation:

```
git --version
```

Configure Git (replace with your GitHub username and email):

```
git config --global user.name "YOUR_USERNAME"  
git config --global user.email "your_email@example.com"
```



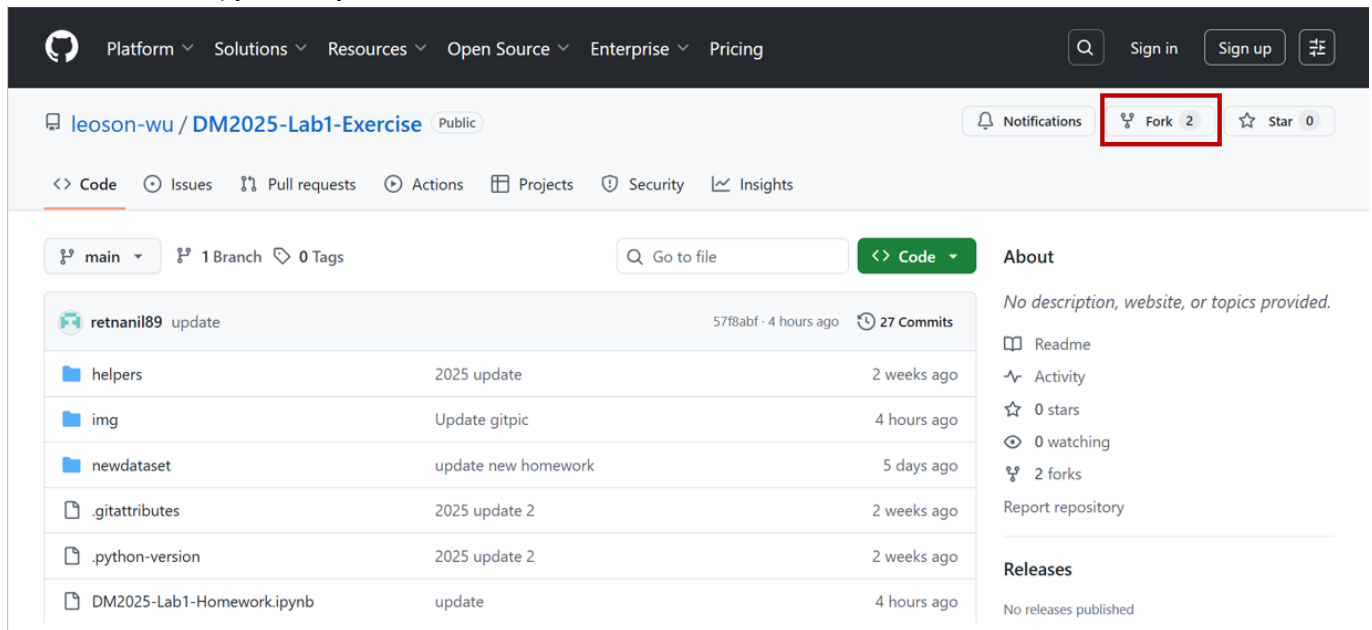
```
Windows PowerShell  
PS C:\Users\ntone> git --version  
git version 2.46.1.windows.1  
PS C:\Users\ntone> git config --global user.name "leoson-wu"  
PS C:\Users\ntone> git config --global user.email "s113062636@m113.nthu.edu.tw"  
PS C:\Users\ntone> git config --global --list  
user.name=leoson-wu  
user.email=s113062636@m113.nthu.edu.tw  
PS C:\Users\ntone> |
```

3. Fork the Repository to your GitHub Account

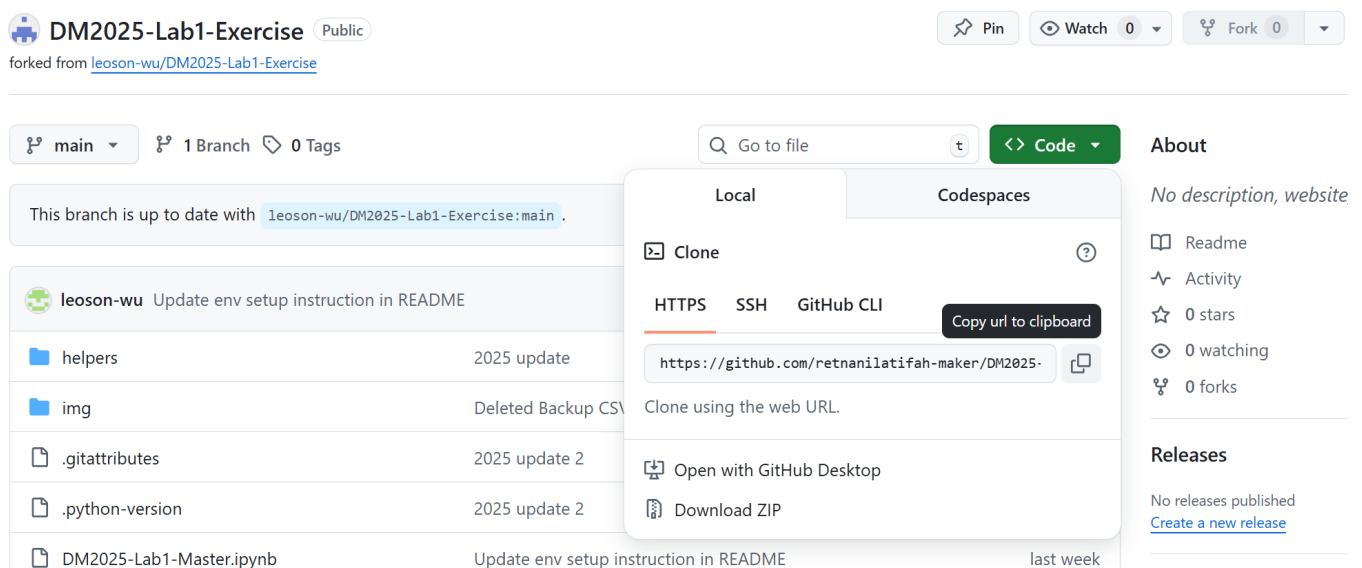
Go to: [DM Lab 1](#) in GitHub,

Sign in to your GitHub account

Click "Fork" to copy it into your own GitHub account.



And it will redirect you to a "copy" of the repository in your own account. Once in your account (check that your name shows up at the top left corner), click the green button "Code", and the clipboard button beside the **link** that pops up.



4. Create a Project Folder and Clone the Repository from your GitHub to your Project Directory

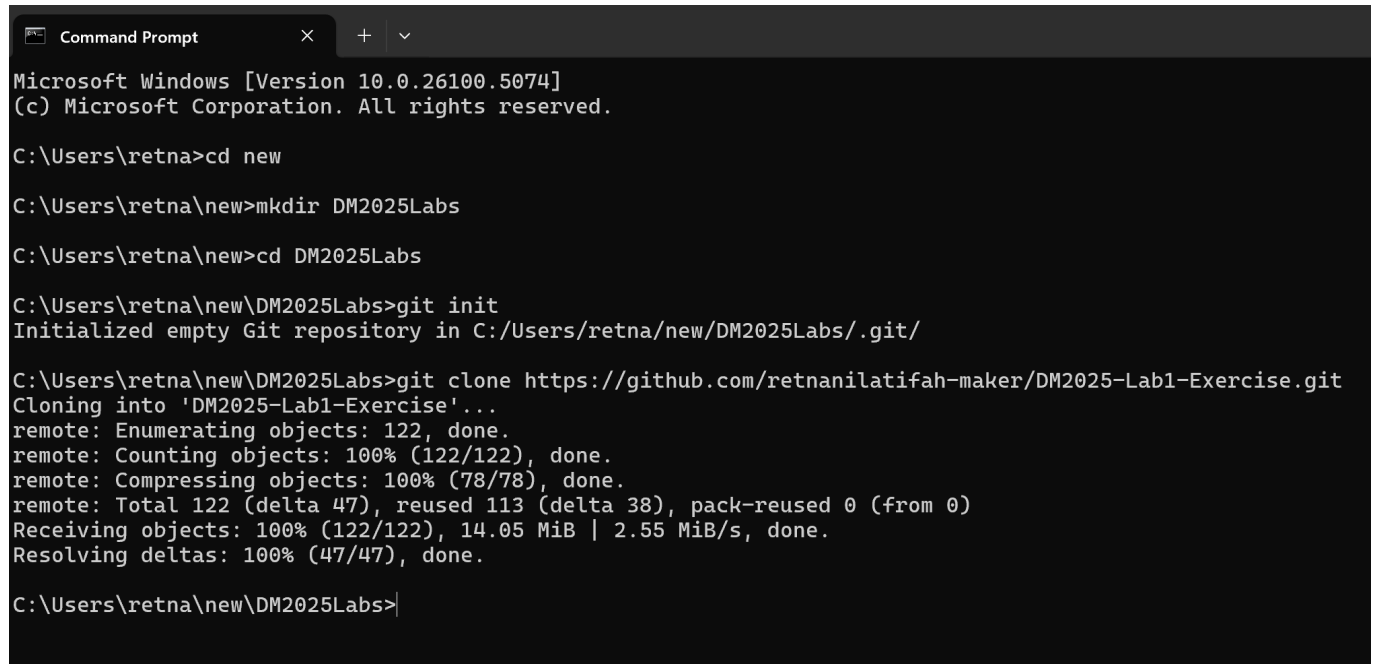
Choose a location for your labs and create a directory:

Open a "Command Prompt" window in Windows or a "Terminal" window in macOS/Linux. Type the following commands, followed by the Enter key for each line:

```
cd <yourpath>
mkdir DM2025Labs
cd DM2025Labs
git clone <link you copied in the previous step>
```

Replace <yourpath> by the path where you're going to store your documents. Below is an example, where I store my Lab in the "new" folder.

Replace <link you copied in the previous step> with the URL link to your own fork of the repository in your GitHub account (not the TAs one)



```
Command Prompt
Microsoft Windows [Version 10.0.26100.5074]
(c) Microsoft Corporation. All rights reserved.

C:\Users\retna>cd new

C:\Users\retna\new>mkdir DM2025Labs

C:\Users\retna\new>cd DM2025Labs

C:\Users\retna\new\DM2025Labs>git init
Initialized empty Git repository in C:/Users/retna/new/DM2025Labs/.git/

C:\Users\retna\new\DM2025Labs>git clone https://github.com/retnanilatifah-maker/DM2025-Lab1-Exercise.git
Cloning into 'DM2025-Lab1-Exercise'...
remote: Enumerating objects: 122, done.
remote: Counting objects: 100% (122/122), done.
remote: Compressing objects: 100% (78/78), done.
remote: Total 122 (delta 47), reused 113 (delta 38), pack-reused 0 (from 0)
Receiving objects: 100% (122/122), 14.05 MiB | 2.55 MiB/s, done.
Resolving deltas: 100% (47/47), done.

C:\Users\retna\new\DM2025Labs>
```

5. Install uv

In terminal or PowerShell:

```
pip3 install uv
uv --version
```

6. Create a Virtual Environment with uv

Navigate to the project folder: **DM2025-Lab1-Exercise** and create Virtual Environment

The Virtual Environment must be created under the project folder: **DM2025-Lab1-Exercise**

```
cd <your path to the DM2025-Lab1-Exercise>
uv venv
```

This creates a .venv folder inside the project.

7. Install the Dependent Libraries

Under project folder: **DM2025-Lab1-Exercise**

Install Libraries

```
uv add jupyter numpy pandas matplotlib plotly nltk scikit-learn seaborn pami umap-learn
```

This installs all required Python packages: jupyter, numpy, pandas, matplotlib, plotly, nltk, scikit-learn, seaborn, pami, umap-learn

You can also install extra libraries if needed.

```
uv add <library_name>
```

(Alternative) If you are using Kaggle or Colab

Install the library via `pip3` instead of `uv`

```
pip3 install jupyter numpy pandas matplotlib plotly nltk scikit-learn seaborn pami umap-learn
```

Note: In Kaggle/Colab, Python version may differ (e.g., 3.10). Some packages could behave differently.

8. Register Jupyter Kernel

Under project folder: `DM2025-Lab1-Exercise`

```
uv run python -m ipykernel install --user --name=dm2025lab --display-name "Python (dm2025lab)"
```

9. Run Notebook

Run in VS Code

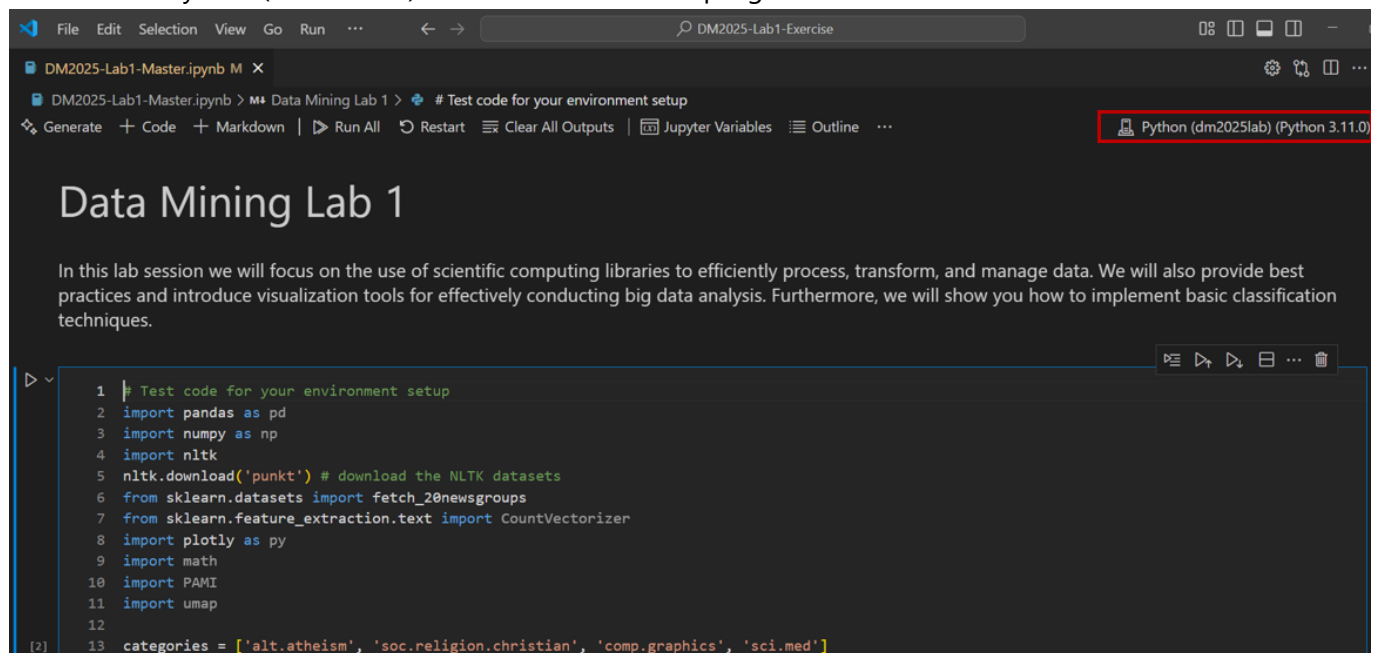
If using VS Code:

Open your terminal/PowerShell

```
cd <your path to the DM2025-Lab1-Exercise>  
code
```

Open the `DM2025-Lab1-Master.ipynb`

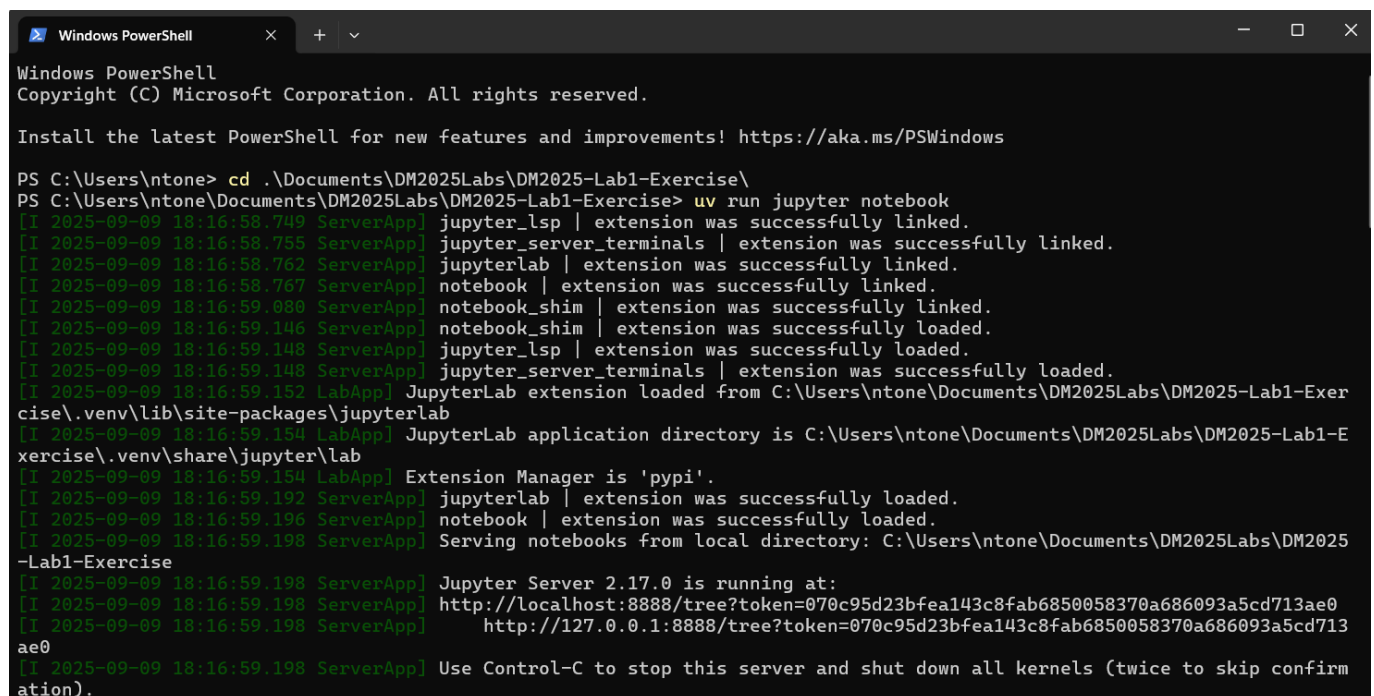
Then select "Python (dm2025lab)" as the kernel in the top-right corner.



(Alternative) Run Jupyter Notebook in Browser

Start Jupyter:

```
cd <your path to the DM2025-Lab1-Exercise>
uv run jupyter notebook
```



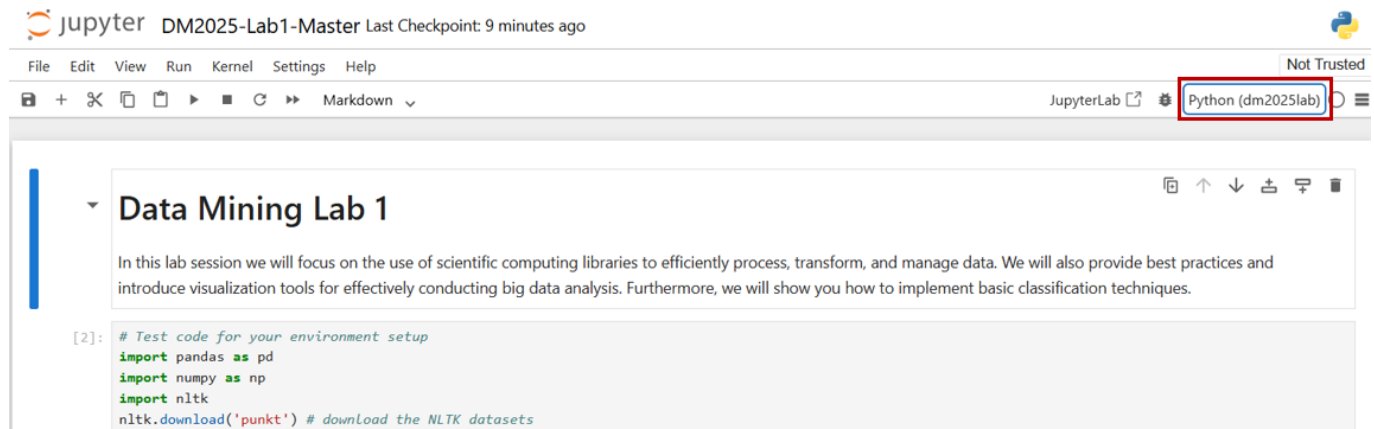
If error occurs:

```
python -m notebook
```

A browser window will open.

Open the **DM2025-Lab1-Master.ipynb**

Select **Python (dm2025lab)** as the notebook kernel on the top-right corner.



(Alternative) Run in Kaggle

If you cannot set up Python locally:

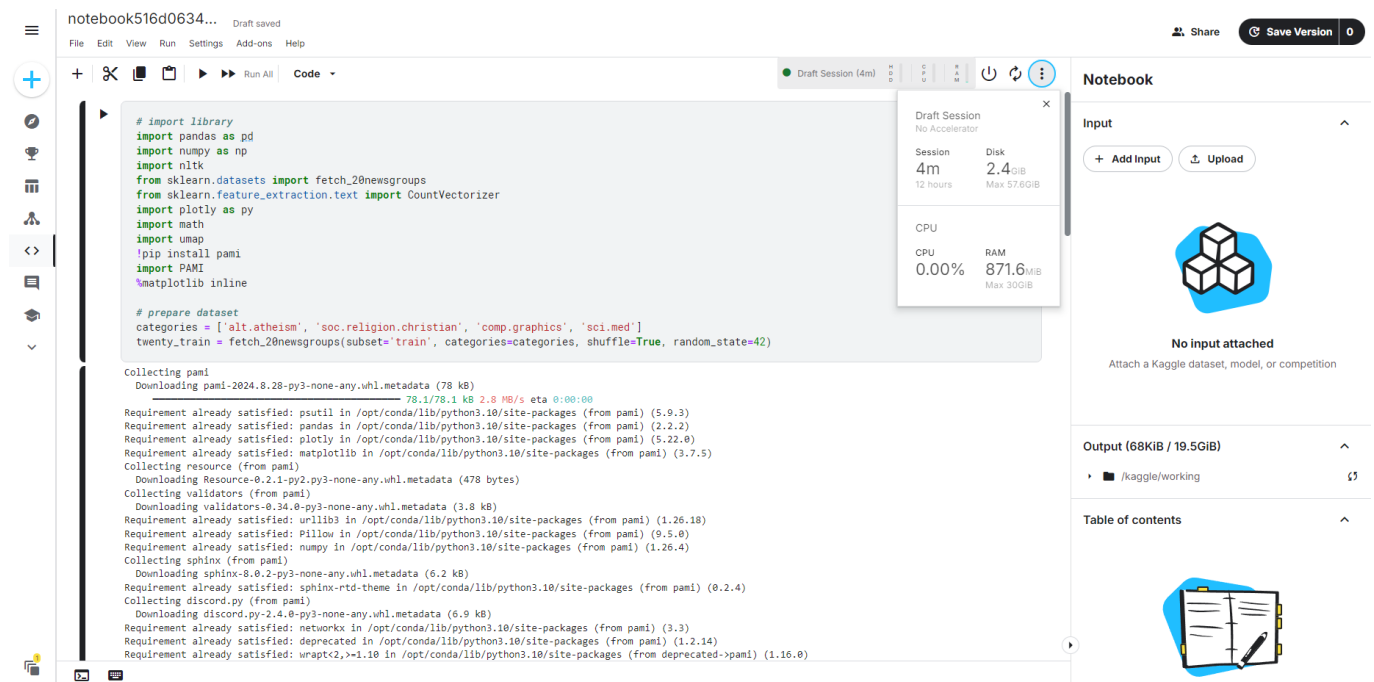
Create an account: <https://www.kaggle.com/>

Go to: <https://www.kaggle.com/kernels> → "New Notebook"

Run the test script.

For missing libraries (e.g., PAMI), install manually:

```
!pip3 install pami
```



10. Test Your Environment

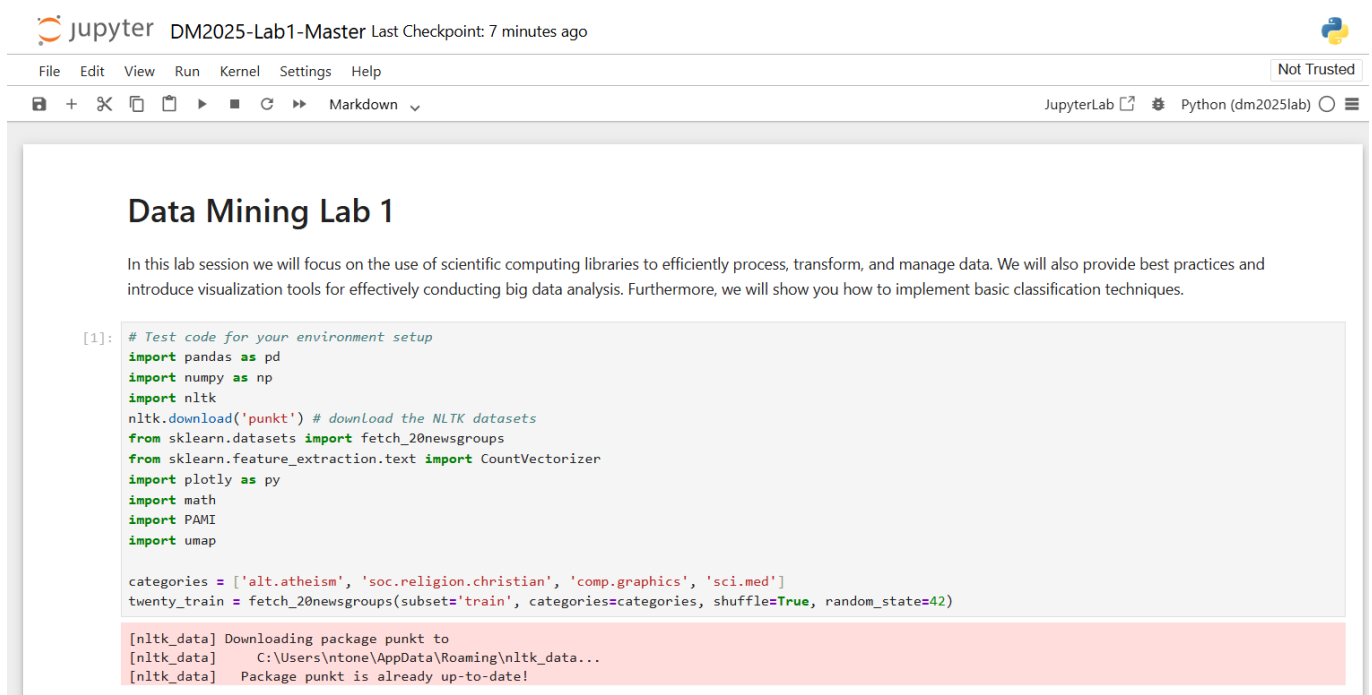
Open the **DM2025-Lab1-Master.ipynb**

Paste the script below into a notebook cell and run it:

```
# test code for environment setup
import pandas as pd
import numpy as np
import nltk
nltk.download('punkt') # download the NLTK datasets
from sklearn.datasets import fetch_20newsgroups
from sklearn.feature_extraction.text import CountVectorizer
import plotly as py
import math

# If you get "ModuleNotFoundError: No module named 'PAMI'"
# run the following in a new Jupyter cell:
# !pip3 install PAMI
import PAMI
import umap

categories = ['alt.atheism', 'soc.religion.christian', 'comp.graphics', 'sci.med']
twenty_train = fetch_20newsgroups(subset='train', categories=categories,
shuffle=True, random_state=42)
```



The screenshot shows a JupyterLab window titled "DM2025-Lab1-Master" with a last checkpoint of 7 minutes ago. The interface includes a menu bar (File, Edit, View, Run, Kernel, Settings, Help) and a toolbar with icons for file operations and execution. The main area displays a code cell with the following content:

```
[1]: # Test code for your environment setup
import pandas as pd
import numpy as np
import nltk
nltk.download('punkt') # download the NLTK datasets
from sklearn.datasets import fetch_20newsgroups
from sklearn.feature_extraction.text import CountVectorizer
import plotly as py
import math
import PAMI
import umap

categories = ['alt.atheism', 'soc.religion.christian', 'comp.graphics', 'sci.med']
twenty_train = fetch_20newsgroups(subset='train', categories=categories, shuffle=True, random_state=42)
```

The output of the code cell is displayed in a pink-shaded box:

```
[nltk_data] Downloading package punkt to
[nltk_data] C:\Users\ntone\AppData\Roaming\nltk_data...
[nltk_data] Package punkt is already up-to-date!
```

If no errors occur, your environment is ready.

Troubleshooting

Ask classmates or TAs for help before the lab if you encounter installation issues.

If you prefer a GUI for Git, use GitHub Desktop: <https://desktop.github.com/>

Good luck with the setup and see you on Monday, Sept 15!

Best regards, The TAs

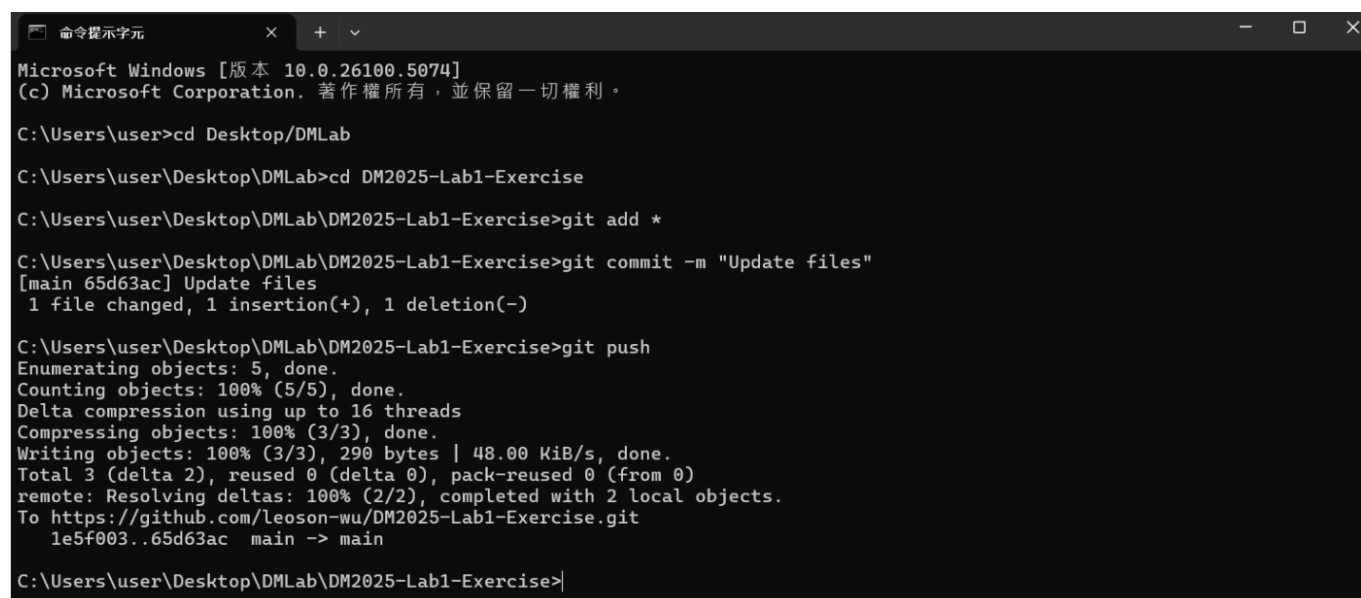
If everything is OK, you can start to do the lab ...

Save your Progress by Push

Remember to save your notebooks. You will also have to "Push" the changes you've made in your computer to the internet. To do this, open a "Command Prompt" window in Windows or a "Terminal" window in macOS/Linux. Type the following commands followed by the Enter key:

```
cd <your path to the DM2025-Lab1-Exercise>
git add .
git commit -m "yourmessage"
git push
```

You can replace "yourmessage" with something like "Finished Ex1 and Ex2. Added graph for Ex. 6" . You can save and commit as often as you like. Below is an example:



```
Microsoft Windows [版本 10.0.26100.5074]
(c) Microsoft Corporation. 著作權所有，並保留一切權利。
C:\Users\user>cd Desktop/DMLab
C:\Users\user\Desktop\DMLab>cd DM2025-Lab1-Exercise
C:\Users\user\Desktop\DMLab\DM2025-Lab1-Exercise>git add *
C:\Users\user\Desktop\DMLab\DM2025-Lab1-Exercise>git commit -m "Update files"
[main 65d63ac] Update files
1 file changed, 1 insertion(+), 1 deletion(-)
C:\Users\user\Desktop\DMLab\DM2025-Lab1-Exercise>git push
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 16 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 290 bytes | 48.00 KiB/s, done.
Total 3 (delta 2), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (2/2), completed with 2 local objects.
To https://github.com/leoson-wu/DM2025-Lab1-Exercise.git
1e5f003..65d63ac main -> main
C:\Users\user\Desktop\DMLab\DM2025-Lab1-Exercise>
```

Submission Guidelines and Deadline

Make sure to commit and push your changes to your GitHub repository **BEFORE the deadline for each phase (September 28th and October 19th, 11:59 pm, Sunday)**. During the second phase, the answers from the first phase will not be considered if they can not be pushed on time.

Make sure your repository must contains 2 notebooks, including:

1. [DM2025-Lab1-Master.ipynb](#) from [DM Lab 1 Master](#)
2. [DM2025-Lab1-Homework.ipynb](#) from [DM Lab 1 Homework](#)

Deadline:
September 28th 11:59 pm

- | | | | |
|-------------------------------|---|---------|--------|
| 1. DM2025-Lab1-Master.ipynb | : | Phase1, | Phase2 |
| 2. DM2025-Lab1-Homework.ipynb | : | Phase1, | Phase2 |

Deadline:
October 19th 11:59 pm

When you're done (or at any moment), find your repository link. Open the assignment page on our [NTU COOL platform](#). Make a submission by pasting the link to your git repository (or the link to your Kaggle kernel) to **Lab 1 section**.

The screenshot shows the NTU COOL platform interface. The top navigation bar includes a hamburger menu, the course title '114-1_TAICA_Data Mining : Concepts, Techniques, and Ap...', and a link to 'Assignments'. The sidebar on the left contains various navigation options: Account, Dashboard, Courses, Calendar, Inbox, Registration, FAQ, and Settings. The main content area displays a list of assignments under the heading '114-1_TAICA_Data Mining : Concepts, Techniques, and Ap... > Assignments'. The assignments are: Lab 1 (100 pts, 10% of Total), Lab 2 (100 pts, 10% of Total), Final Project (100 pts, 25% of Total), and Class Participation (10% of Total). Each assignment has a status icon and a plus sign for more options. At the bottom, it says 'No assignments in this group'.

You can find your repository link by logging into [Github](#), clicking on your profile icon on the upper right corner, selecting "Your repositories", and clicking on the name of your repository. Then copy the link in your browser.

Again, **we will not consider pushes made after the deadline.**

That's it! We wish you Good luck!