

HW0: Environment Setup + Numpy

1 Overview

Welcome to Deep Learning! HW0 will evaluate your NumPy knowledge while introducing you to the development tools you'll use throughout the semester.

This assignments in this course are set up with uv, and can be completed on Google Colab *or* your local IDE i.e. VSCode. Most (if not all) assignments, including this one, will be paired with a corresponding Theory section that will be completed in LaTeX.

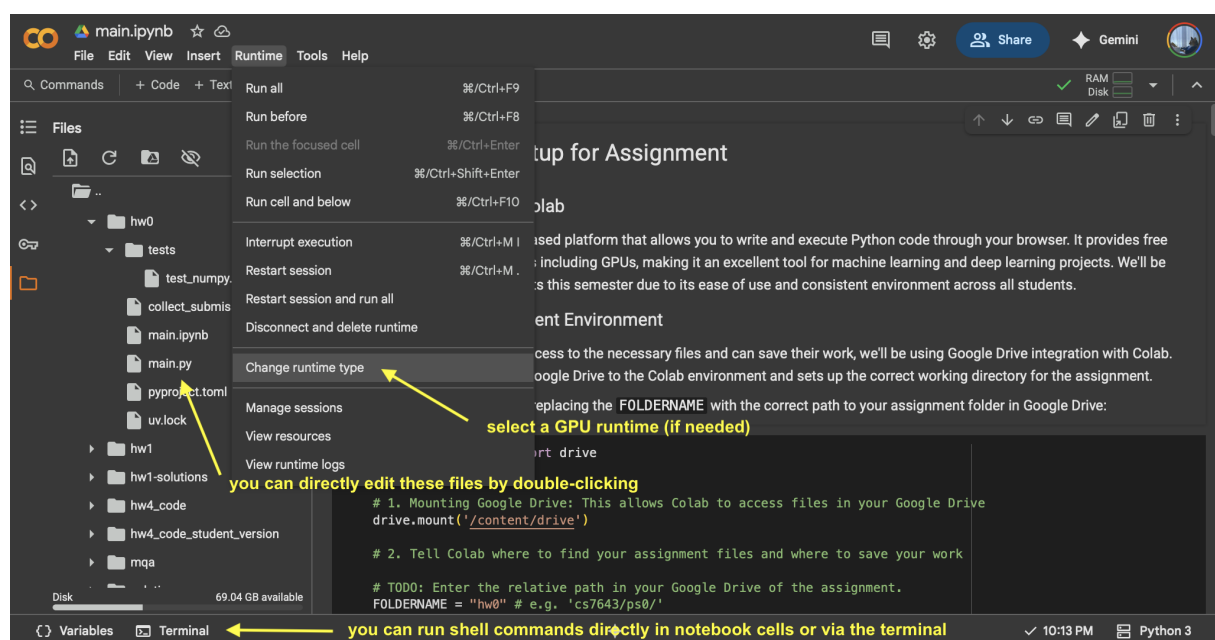
TODO:

1. Follow the instructions in [2](#) or [3](#) to set up Colab or VSCode
2. Complete the steps in `main.ipynb` to complete the NumPy exercises in `main.py`
3. Run `collect_submission.py` to generate your submission files. Make sure to save your notebook before running this command, otherwise your latest changes may not appear.
4. Submit `hw0_code_submission.zip` to **HW0 Code** Gradescope
5. Submit `hw0_notebook_submission.pdf` to **HW0 Notebook** Gradescope
6. Complete `hw0_theory.pdf` in your PDF annotator of choice and submit it to **HW0 Theory** Gradescope

2 Colab

Google Colab is a free cloud platform that allows you to write Jupyter Notebooks, run Python, and access GPU compute. You can complete all of the course's assignments entirely in Colab.

1. Login to your Google Drive and upload the `hw0` folder
2. Open `main.ipynb` with Google Colab
3. Run the cells under "Google Colab" to set up your environment



3 VSCode

Setting up VSCode requires a little more effort than Colab, but some may find that it provides a better developer experience. While your laptop/local machine may not have sufficient compute to train the bigger models in later assignments, you can still use VSCode to write and test your code, and Colab to train your models.

1. Open the assignment folder in VSCode
2. Install `Python` and `Jupyter` extensions
3. See 4 to install dependencies with `uv`
4. Press `Ctrl+Shift+P` (or `Cmd+Shift+P` on macOS) to open Command Palette
5. Type "Python: Select Interpreter" and choose the `uv`-managed Python interpreter. If it does not show up, you can set it with "Enter Interpreter Path" (the path looks like `./venv/bin/python`)
6. Open `main.ipynb`
7. Click "Select Kernel" in the top right and then choose "Select Another Kernel", "Python Environments", and then the interpreter you just selected

4 uv

`uv` is a Python package manager that replaces `pip` and `conda`. We use it to reproduce the Python environment for the assignments, described in the `pyproject.toml` and `uv.lock` files. Follow the instructions on the <https://docs.astral.sh/uv/getting-started/installation/> to install `uv`.

```
1 # install uv on Mac/Linux
2 curl -LsSf https://astral.sh/uv/install.sh | sh
3
4 # (or) install uv on Windows
5 powershell -ExecutionPolicy ByPass -c "irm https://astral.sh/uv/install.ps1 |
   iex"
```

Afterwards, `cd` into the assignment directory and create the virtual environment for the HW0.

```
1 # run this to create the virtual environment
2 uv sync
```