Homework 0: Setting Up Development Environment CPE 490/590 ST

Instructor: Rahul Bhadani

Due: January 13, 2025, 11:59 PM
100 points

You are allowed to use a generative model-based AI tool for your assignment. However, you must submit an accompanying reflection report detailing how you used the AI tool, the specific query you made, and how it improved your understanding of the subject. You are also required to submit screenshots of your conversation with any large language model (LLM) or equivalent conversational AI, clearly showing the prompts and your login avatar. Some conversational AIs provide a way to share a conversation link, and such a link is desirable for authenticity. Failure to do so may result in actions taken in compliance with the plagiarism policy.

Additionally, you must include your thoughts on how you would approach the assignment if such a tool were not available. Failure to provide a reflection report for every assignment where an AI tool is used may result in a penalty, and subsequent actions will be taken in line with the plagiarism policy.

Submission instruction:

Upload a .pdf on Canvas with the format {firstname.lastname}_CPE 490/590 ST_hw00.pdf. For example, if your name is Sam Wells, your file name should be sam.wells_CPE 490/590 ST_hw00.pdf. If there is a programming assignment, then you should include your source code along with your PDF files in a zip file {firstname.lastname}_CPE 490/590 ST_hw00.zip. Your submission must contain your name, and UAH Charger ID or the UAH email address. Please number your pages as well.

---

1 Working with Google Colab (20 Points)

Start your screen recorder, either using Zoom, or anything equivalent, can capture the following tasks:

Create a Google colab notebook on https://colab.research.google.com/, name it `hw00.ipynb`.

1. Create a markdown (text) cell, and enter your name, and email as follows:

```
# First Name Last Name
## email@uah.edu
```

2. Install some python packages as follows by creating new code cell and execute:

```
%reload_ext autoreload
%autoreload 2
! pip install pandas matplotlib torch seaborn scikit-learn pygame
```

   Please describe, what does

```
%reload_ext autoreload
%autoreload 2
```

   do.

3. Next, create another new code cell, and execute the following

```
import torch
import numpy as np
import matplotlib.pyplot as plt

print("Hello, World!")
```

Finally, as a part of your submission, provide a public URL to your Google Colab notebook. Further, upload your screen recorded video to YouTube with visibility set to *unlisted* or *public*, and provide its url in the submission.

2 Install Visual Studio Code and Important Extensions, and Prepare for Python Coding (20 Points)

Note: the following assignment is meant for MacOS, or Ubuntu Linux. If you have Windows OS, please modify the following instructions accordingly.

Start your screen recorder, either using Zoom, or anything equivalent, can capture the following tasks:

1. Download and install Visual Studio Code.
2. Install Anaconda, create an Anaconda virtual environment, and jupyter kernel as described in Lecture 01 slides.
3. Activate your conda environment in the command line.
4. Install the following packages from command line after activating your conda environment:

```
pip install pandas matplotlib torch seaborn scikit-learn pygame
```

5. Open Visual Studio Code and click on the Extensions icon in the left sidebar (or press Ctrl+Shift+X).
6. Search for and install the following extensions:
   (a) Python (by Microsoft)
   (b) Pylance (by Microsoft)
   (c) Jupyter (by Microsoft)
   (d) Jupyter Notebook (by Microsoft)
7. Create a new file and save it as `vscodehw00.ipynb`. On the top right corner, click **Select Kernel** and choose the appropriate kernel you created earlier.
8. Enter following code to your notebook and execute them

```
import torch
import numpy as np
import matplotlib.pyplot as plt

print("Hello, Jupyter!")
```

   Make sure the notebook is executed successfully.

Finally, as a part of your submission, upload your screen recorded video to YouTube with visibility set to *unlisted* or *public*, and provide its url in the submission.

3 Working with GitHub (20 points)

Start your screen recorder, either using Zoom, or anything equivalent, can capture the following tasks:

1. Create a GitHub account if you don't have one already.
2. Install Git on your computer if it's not already installed.
3. Configure Git with your name and email:
4. ```
   git config --global user.name "Your Name"
   git config --global user.email "your.email@example.com"
   ```

   Replace the values as per your preferance.
5. Create a new repository on GitHub named "CPE490590hw00". Make sure that the repository you have created is a public repository.
6. Clone it locally to your computer:

   ```
   git clone https://github.com/yourusername/CPE490590hw00
   ```

   Place your `vscodehw00.ipynb` in `CPE490590hw00` folder.

   You should replace repo URL accordingly. Change your current working directory to `CPE490590hw00` folder using command line.
7. Create a new branch named "hw00":

   ```
   git checkout -b hw00
   ```

8. Add your `vscodehw00.ipynb` file to this branch.

   ```
   git add vscodehw00.ipynb
   git commit -m "Add hw 00 Jupyter notebook"
   ```

9. Create a file `.gitignore`. What purpose does `.gitignore` serve?

   Add following content to the `.gitignore` file:
   ```

```
# --------------------------
# PYTHON
#------------------------------
.ipynb_checkpoints/
dist/
*.egg-info/
## ------------------------------
# Visual Studio Code
#------------------------------
.vscode/
```

Explain what would adding the above content to the `.gitignore` file do?

Next, add `.gitignore` file to git and commit it:

```
git add .gitignore
git commit -m "Added gitignore file"
```

10. Push your branch to GitHub:

```
git push origin hw00
```

Finally, as a part of your submission, upload your screen recorded video to YouTube with visibility set to *unlisted* or *public*, and provide its url in the submission.

4 Working with UV (40 points)

In this section, you are required to learn UV python project manager.

Start your screen recorder, either using Zoom, or anything equivalent, can capture the following tasks:

Go to https://docs.astral.sh/uv/

and follow the tutorials listed in following sections of the webpage:

- Getting started

  - Installation
  - First steps
  - Features
  - Getting help
- Guides
  - Installing Python
  - Running scripts
  - Using tools
  - Working on projects

Finally, as a part of your submission, upload your screen recorded video to YouTube with visibility set to *unlisted* or *public*, and provide its url in the submission.

Additionally, go through the Concepts at https://docs.astral.sh/uv/concepts/ based on which some quiz questions will be asked.