MCIS6273 Data Mining (Prof. Maull) / Fall 2025 / HW0

Points Possible	Due Date	Time Commitment (estimated)
10	Monday September 1 @ Midnight	*up to* 12 hours

- **GRADING:** Grading will be aligned with the completeness of the objectives.
- **INDEPENDENT WORK:** Copying, cheating, plagiarism and academic dishonesty *are not tolerated* by University or course policy. Please see the syllabus for the full departmental and University statement on the academic code of honor.

OBJECTIVES

- Familiarize yourself with Github and basic git
- Familiarize yourself with the JupyterLab environment, Markdown and Python
- Explore JupyterHub Linux console integrating what you learned in the prior parts of this homework
- Perform basic Python coding using Light Pollution Dataset
- Complete the online assessment

WHAT TO TURN IN

You are being encouraged to turn the assignment in using the provided Jupyter Notebook. To do so, make a directory in your Lab environment called `homework/hw0`. Put all of your files in that directory. Then zip or tar that directory, rename it with your name as the first part of the filename (e.g. `maull_hw0_files.zip`, `maull_hw0_files.tar.gz`), then download it to your local machine, then upload the `.zip` to Blackboard.

If you do not know how to do this, please ask, or visit one of the many tutorials out there on the basics of using zip in Linux.

If you choose not to use the provided notebook, you will still need to turn in a `.ipynb` Jupyter Notebook and corresponding files according to the instructions in this homework.

ASSIGNMENT TASKS

(0%) Familiarize yourself with Github and basic git

Github (https://github.com) is the *de facto* platform for open source software in the world based on the very popular git (https://git-scm.org) version control system. Git has a sophisticated set of tools for version control based on the concept of local repositories for fast commits and remote repositories only when collaboration and remote synchronization is necessary. Github enhances git by providing tools and online hosting of public and private repositories to encourage and promote sharing and collaboration. Github hosts some of the world's most widely used open source software.

If you are already familiar with git and Github, then this part will be very easy!

§ Task: Create a Zotero account.

Learn about Zotero and if you haven't already, create a free account:

- https://zotero.org

§ Task: Create a public Github repo named `"mcis6273-f25-datamining"` and place a `README.md` file in it.

Create your firxt `file calledREADME.md`' at the top level of the repository.

Please put your Zotero username in the file. Aside from that you can put whatever text you like in the file (If you like, use something like lorem ipsum to generate random sentences to place in the file.). Please include the link to **your** Github repository that now includes the minimal `README.md`. You don't have to have anything elaborate in that file or the repo.

§ Task: Fork the course repository.

Learn to use Github workflows and fork the class repo:

- https://github.com/kmsaumcis/mcis6273_f25_datamining/

(0%) Familiarize yourself with the JupyterLab environment, Markdown and Python

As stated in the course announcement Jupyter (https://jupyter.org) is the core platform we will be using in this course and is a popular platform for data scientists around the world. We have a JupyterLab setup for this course so that we can operate in a cloud-hosted environment, free from some of the resource constraints of running Jupyter on your local machine (though you are free to set it up on your own and seek my advice if you desire).

You have been given the information about the Jupyter environment we have setup for our course, and the underlying Python environment will be using is the Anaconda (https://anaconda.com) distribution. It is not necessary for this assignment, but you are free to look at the multitude of packages installed with Anaconda, though we will not use the majority of them explicitly.

As you will soon find out, Notebooks are an incredibly effective way to mix code with narrative and you can create cells that are entirely code or entirely Markdown. Markdown (MD or `md`) is a highly readable text format that allows for easy documentation of text files, while allowing for HTML-based rendering of the text in a way that is style-independent.

We will be using Markdown frequently in this course, and you will learn that there are many different "flavors" or Markdown. We will only be using the basic flavor, but you will benefit from exploring the "Github flavored" Markdown, though you will not be responsible for using it in this course – only the "basic" flavor. Please refer to the original course announcement about Markdown.

§ Task: THERE IS NOTHING TO TURN IN FOR THIS PART.

Play with and become familiar with the basic functions of the Lab environment given to you online in the course Blackboard.

§ Task: THERE IS NOTHING TO TURN IN FOR THIS PART.

Please *create a markdown document* and read the documentation for basic Markdown here. Learn to use all of the following:

- headings (one level is fine),
- bullets,
- bold and italics

Again, the content of your documentcan be whatever you like, just learn some of the basic functionality of Markdown.

(0%) Explore JupyterHub Linux console integrating what you learned in the prior parts of this homework

The Linux console in JupyterLab is a great way to perform command-line tasks and is an essential tool for basic scripting that is part of a data scientist's toolkit. Open a console in the lab environment and familiarize yourself with your files and basic commands using git as indicated below.

1. In a new JupyterLab command line console, run the `git clone` command to clone the new repository you created in the prior part. You will want to read the documentation on this command (try here https://www.git-scm.com/docs/git-clone to get a good start).
2. Within the same console, modify your `README.md` file, check it in and push it back to your repository, using `git push`. Read the documentation about `git push`.
3. The commands `wget` and `curl` are useful for grabbing data and files from remote resources off the web. Read the documentation on each of these commands by typing `man wget` or `man curl` in the terminal. Make sure you pipe the output to a file or use the proper flags to do so.

§ Task: THERE IS NOTHING TO TURN IN FOR THIS PART.

(80%) Perform basic Python coding using Light Pollution Dataset

As we begin our journey into data mining, we will spend the vast majority of the time in Python.

As has been mentioned, Python is a wonderful language for data processing tasks, and in this assignment we will write a few warm up programs to get started with a using Python to obtain data.

You may be aware that there are a abundant datasets online. We are going to dive into a unique one that we will use for the first several assignments. The dataset is from a citizen science project called *Globe at Night* (GaN).

You can find more information about this project online:

- Globe at Night (GaN)

The primary objective of the project is to encourage individuals to measure night sky darkness and submit their data to the GaN database. This is an important goal because light pollution is a significant issue in our modern world. High levels of light pollution reduce our ability to see (and wonder about) the gloriousness of our vast universe with the naked eye like our ancestors have for millenia. More importantly, these light effects have an impact on organisms and ecosystems, and may also be a contributor to chronic illness in humans. For example, recent research is showing that unnecessary exposure to light at night interacts with circadian rhythms and thus metabolism and hormones, which may set up preconditions for chronic disease in humans.

You may follow the trail wherever it may lead, but as data scientists we want to learn the application of Python and other tools to immediate data problems.

In this assignment we will use Python to extract data from GaN and store this data in a folder.

Your code must be implemented in Jupyter as a notebook and you will be required to turn in a `.ipynb` file along with corresponding data folders.

§ Task: Use Python and BeautifulSoup to write code to extract all data files from GaN.

Many times, data we would like to explore is already in a single file ready for download and engineering.

In the GaN case, we are going to have to get the files from the website, but we will NOT be allowed to do this manually. Instead, we will need to use tools and automation in Python.

A wonderful library for extracting data from HTML pages is BeautifulSoup 4.

Your *primary* task is the following:

- use BeautifulSoup 4 to find all CSV files from the following URL:
  - https://globeatnight.org/maps-data/

The subtasks of your code must do the following:

1. extract the page using BeautifulSoup 4,
2. search for the `<li><a>` tags that contain the pattern `href="/documents/*GaN*20*.csv"`,
3. store the contents of the `href="<contents_to_store>"` patterns in a file, prepending this pattern with `https://globeatnight/` (e.g. if the `"<contents_to_store>"` content is `"/documents/123/GaN1901.csv"` then the stored string is `https://globeatnight.org/documents/123/GaN1901.csv`),
4. the strings should be on a single line and the file called `data/gan_urls.txt`,
5. use the Python library `os` to make a folder called `data/` where the data file containing the URLs will be stored.

You will need to use Python loops and variables. If you do not store the URLs in a folder called `data/` you will lose points.

DO NOT overthink this. There is not a lot of code required to complete the assignment (a solution could be done in 7 lines or less).

WHAT YOU NEED TO BE SUCCESSFUL

1. You will need to use the `requests` library to load the data URL

- do not overthink this, you will simply `import requests` and then run `.get(<URL>).content` and pass this to the BeautifulSoup object.

2. Study the basic structure of the BS object with the demo code on the main documenation site: BS Quick Start
3. Once you have the BS object, just process the tags per the documentation.
   See the documentation in the BS Quick Start, for example `soup.findall()` may be a great place to start.
4. You may consider using the `os` library to check for and create the `data` folder where your file will go (see `os.mkdir()`).

(20%) Complete the online assessment

Please ZIP the folder and subfolder for your assignment and submit it directly to Blackboard.

Once you are done with the coding part of the assignment, you will need to complete the online assessment for the final 4 points (20%) of your grade.

§ Task: Turn in your solution and complete the online HW0 assessment.