HW 6 - NumPy Arrays

Due March 11th, 2026 at 11:59pm

Overview

In this homework assignment, you'll create your own GitHub branch, begin using Jupyter Notebooks and practice working with NumPy arrays.

Note: If you need help creating a branch for your `yourname` repository, please reference **Section 6: Bonus Content** in the **[Class 12]** slides on bCourses. Feel free to also look through **The GGG**.

1 Branching with GitHub

In homework 4, you practiced fetching a branch from the `course_assignments` repository. Now you'll practice making your own branch, clearing it, working in it and submitting it for this homework assignment.

WARNING: Please follow these directions carefully or ask for help during office hours. Running unfamiliar commands randomly can affect your system.

Start by navigating to your `yourname` repository.

Steps:

- Confirm that you are in `yourname` by calling `pwd`. You should see the path end with:

  `/python_decal_sp26/yourname/`

- Run `git branch`. You should see only one branch called `main` with an `*` next to it.
- Run `git branch homework6` to create a new branch.
- Run `git branch` again. Confirm you now see both `main` and `homework6`.
- Switch to the new branch by running `git checkout homework6`.
- Run `git branch` to confirm you're now on `homework6` (indicated by the `*`).
- Run `ls` to confirm it looks like a duplicate of your `main` branch.
- WARNING: Make sure you're on the `homework6` branch before continuing. Now clear the contents of this branch:

```
rm -rf *
```

- Run `ls` to confirm your directory is now empty.
- Stage the deletion: `git add .`
- Save your changes locally: `git commit -m "emptying homework6 branch"`
- WARNING: Make sure you push to `homework6`, not `main`. Run:

  ```
  git push origin homework6
  ```

- Take a screenshot showing all the commands and their outputs.
- Create a folder named `homework6` by running `mkdir homework6`.
- Name the screenshot: `hw6_branch`
- Save it inside the new `homework6` folder in your `homework6` branch.

You'll complete the rest of this homework inside this branch.

IMPORTANT: If you switch branches at any point, make sure to save your work with:

```
git add .
git commit -m "saving progress"
git push origin <branch>
```

DO NOT switch branches without committing and pushing first. You risk confusing yourself and losing your work.

2 Open Jupyter Lab

While in the `homework6` branch, launch Jupyter Lab:

```
jupyter lab
```

You should see a terminal output like this:

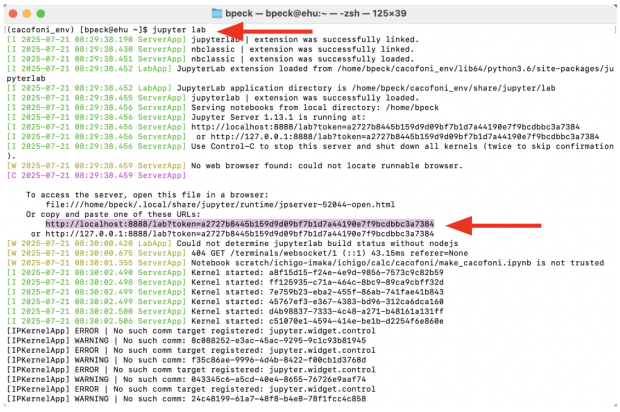

Figure 1: The first arrow shows the `jupyter lab` command. The second arrow shows the highlighted URL you need to copy.

Paste that URL into your browser. Jupyter Lab will open, starting in your `homework6` folder.

- Create a Jupyter Notebook inside of `homework6` and name it `homework6.ipynb`.

Make sure that you **run all the cells** in your Jupyter Notebook before submitting to Gradescope.

3 NumPy Arrays

At the top of your Jupyter Notebook, import NumPy:

```
import numpy as np
```

3.1 1D NumPy Arrays

Create the following list:

```
list = [20, 27, 9, 12, 31, 17, 4, 50, 39, 35]
```

Convert it to a NumPy array and call it `arr`. Then:

- Multiply each item by 4. Print the result.
- Find and print the mean.
- Find and print the median.
- Find and print the standard deviation.
- Use `np.argmin()` and `np.argmax()` to get the index of the smallest and largest numbers.
- Use those indexes to print the corresponding values from `arr`.
- Use `np.min()` and `np.max()` to print the smallest and largest values. Do they match those from the previous step?
- Create a new array with only values greater than 20 with Boolean indexing. Print it.

3.2 2D NumPy Arrays

Create the following array:

```
matrix = [
    [5, 12, 17, 9],
    [8, 21, 14, 3],
    [6, 13, 19, 11]
]
```

Now:

- Print the shape of the array. How many rows? How many columns?
- Print the third column.
- Print the second row.
- Multiply all elements by 2. Print the result.
- Find and print the mean of each column.
- Find and print the overall mean of the array.
- Find the max value in each row and the min value in each column.

- Call `.flatten()` on the array. What is the new shape? What happened?
- Use `np.argmin()` and `np.argmax()` to find the flattened index of the smallest and largest values.
- Use `np.unravel_index()` to convert those flat indices to row-column coordinates.
- Use Boolean indexing to create a new array with values greater than 15.
- Sort each row. Print the entire sorted array.
  Hint: Use np.sort().

4 Prime Star Systems

A friend claims that alien life is more likely to exist around stars whose **magnitude** is a **prime number**. Magnitude is the brightness of a celestial object. The brighter the object, the lower the number assigned as a magnitude.

You decide to test this theory using the following 2D array of star magnitudes:

```
arr = np.array([
    [2, 3, 5],
    [14, 6, 8],
    [11, 13, 17],
    [7, 10, 13]
])
```

Write a function that returns only the rows where **at least one** value is a prime number.

5 Let's Play Checkers!

Unfortunately, after combing through the data you found no evidence of alien life. You decide to take a break from your cutting-edge research to play checkers. Yikes! There are no checkerboards in sight. Looks like you will have to make one yourself.

5.1

Write a function that returns a 8x8 NumPy array of all zeros.

5.2

For the odd-numbered rows, create an alternating pattern starting with 1.

5.3

Finish the even-numbered rows so your board looks like this:

```
>>> checkerboard()
array([[1, 0, 1, 0, 1, 0, 1, 0],
       [0, 1, 0, 1, 0, 1, 0, 1],
       ...
])
```

5.4

Modify your function to return a board that starts with 0 instead:

```
>>> reverse_checkerboard()
array([[0, 1, 0, 1, 0, 1, 0, 1],
       [1, 0, 1, 0, 1, 0, 1, 0],
       ...
])
```

6 The Expanding Universe

You've been inspired by a lecture on the expanding universe. Write a function that takes a string and a number, and returns that string with the specified number of spaces between each letter.

Examples:

```
>>> universe = np.array(['galaxy', 'clusters'])

>>> expansion(universe, 1)
array(['g a l a x y', 'c l u s t e r s'])

>>> expansion(universe, 2)
array(['g  a  l  a  x  y', 'c  l  u  s  t  e  r  s'])
```

7 Second-Brightest Star

While observing with the 10-meter W. M. Keck telescope, you decide you need a **guide star** for each cluster you are studying.

Guide star = a reference star used to accurately maintain the tracking by a telescope of a celestial body.

You decide on guiding with the second brightest star and observe the brightest. Given a 2D array where each **row** is a star cluster, write a function that returns an array with the **second-brightest** value from each row.

Example:

```
>>> np.random.seed(123)
>>> stars = np.random.randint(500, 2000, (5, 5))
array([[1123, 1456, 1789, 1324, 1876],
[1567, 1987, 1678, 1405, 1589],
[1345, 1654, 1523, 1109, 1923],
[1298, 1890, 1367, 1784, 1432],
[1823, 1756, 1489, 1672, 1550]])
>>> secondDimmest(stars)
array([1789, 1678, 1654, 1784, 1756])
```

8 Submitting Your Homework

After completing everything:

Steps:

1. In your `yourname/` repository, run:

```
git add .
git commit -m "done with hw6"
git push origin homework6
```

2. Take a screenshot of the terminal output.
3. Name the screenshot: `hw6_changes`.
4. Place it inside your `homework6/` folder.

5. Ensure all screenshots are saved correctly and your code runs without errors. Your `homework6/` folder should now contain:

```
homework6/
|--- homework6.ipynb
|--- hw6_branch.png
|--- hw6_changes.png
```

6. Go to Gradescope and find the **Homework 6: NumPy Arrays** assignment.
7. Select the option to upload a GitHub repository.
8. Make sure to upload your `homework6` branch.
9. Submit your `yourname` repository.

Great job!