NumPy

CAC 350

Before we start…

- Green screen?
- Did you take the reading quiz…quick, it's still open!!!
- I'll do my best to have reading quizzes posted 24 hours before class, no promises!
- By the way, I know this material is dry - just getting everyone acquainted with NumPy

Review

- What do the In and Out represent? Why do we care?
- What's NumPy?
- Can I just use it?
- How do I make an array with 1,2,3,4,5?
- How do I make an array with 10 random numbers?

Jupyter Notebook

NumPy Arrays

- Really need to understand how to manipulate arrays. Why?
- Attributes of arrays: Size, shape, memory consumption, data types
- Indexing of arrays: Getting/setting value of individual array elements
- Slicing of arrays: Getting/setting smaller subarrays within a larger array
- Reshaping of arrays: Changing the shape of a given array
- Joining and splitting of arrays: Combining multiple arrays into one, and splitting one array into many

NumPy Array Attributes

- One, Two, and Three dimensional arrays…how many dimensions can you have?

- Give me an example of what you might store in a 1D array? 2D array? 3D array?

- What does it mean to seed a random number generator?

Jupyter Notebook

Array Attributes

- ndim: number of dimensions
- shape: size of each dimension
- size: total size of the array
- Let’s look at the arrays we just created and determine these attributes for each one
- dtype: data type
- itemsize: size in bytes of each element in the array
- nbytes: total size of array in bytes

Array Indexing

- Just like in regular Python, we can access one location in an array, even negatives from the end

- How would I access the 8?

```
x = np.array([[1,4,5],
              [3,8,9],
              [7,2,0]])
```

- Arrays are mutable so we can change the value of single elements, but they have to have the same data type as the other elements in the array

Array Slicing

- Just like in regular Python

x[start:stop:step]

- Given x = np.array([2,4,6,7,2,8,1])
- What would x[:3] return?
- What would x[4:] return?
- What would x[::3] return?
- What would x[1:5:2] return?
- What would x[::-1] return?

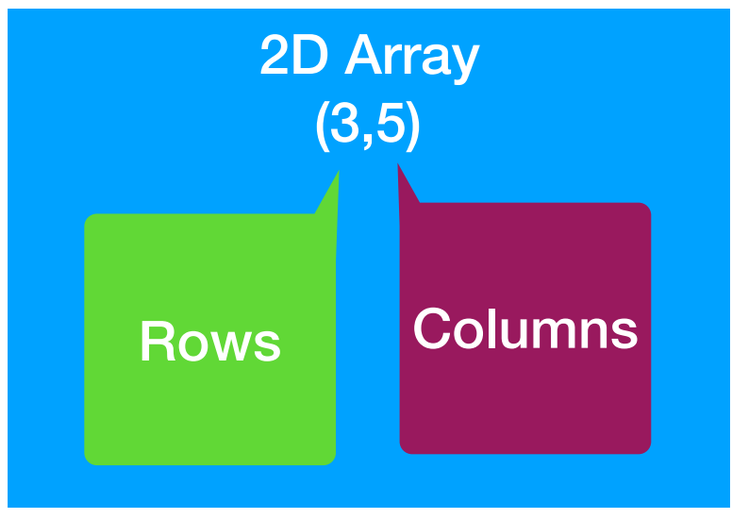
Slicing Multidimensional Arrays

- Same idea, but separate each dimension's slice by comma

- Create a 3D array (3x4x5) with random numbers

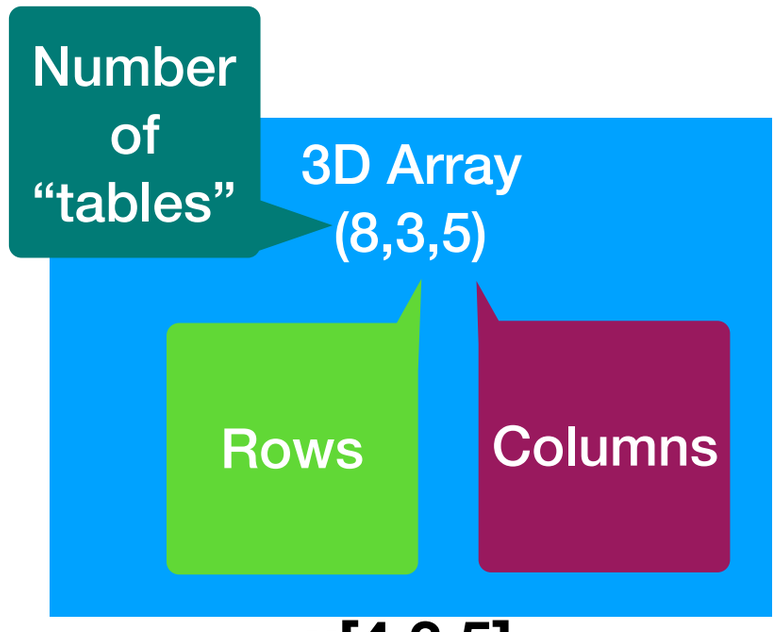
- Get just the second matrix

- Get the first column of all matrices



x[3,5]

x[:,3]



x[4,3,5]

x[2,:,3]

Copy vs. View

- Slicing returns a view of the array. What is a view?
- Why do we care? Advantage/disadvantage?
- To copy, we have to say .copy() at the end of our slice

Reshaping

- Cool trick…change the dimensions of the array as long as the original array can be manipulated into the new dimensions
  - 1D to 2D
  - Flatten 2D to 1D
  - Providing a -1 in one of the dimensions allows Python to infer what you want there based on the other information you provide (first assignment will have some practice problems on this)

Concatenation

- Can glue two arrays together just like in regular python
- np.concatenate([a1,a2])
- np.vstack/np.hstack/np.dstack advised over np.concatenate for multidimensional arrays for readability
- Jupyter Notebook

Splitting

- Break apart arrays: np.split, np.hsplit, np.vsplit, np.dsplit
- x = [1,2,3,99,99,3,2,1]
  x1,x2,x3 = np.split(x,[3,5])
- Why do I have three variables to the left of the equal sign? We usually only have one.
- Jupyter Notebook

Next Class

- Remainder of Chapter Two

- First assignment will be posted - small problems to get started