

What makes NumPy. Shape() different from NumPy. Size()?

I noticed that some numpy operations take an argument called shape, such as np.zeros, whereas some others take an argument called size, such as np.random.randint. To me, those arguments have the same function and the fact that they have different names is a bit confusing. Actually, size seems a bit off since it really specifies the .shape of the output.

2. In NumPy, describe the idea of broadcasting.

The term broadcasting refers to the ability of NumPy to treat arrays of different shapes during arithmetic operations. Arithmetic operations on arrays are usually done on corresponding elements. If two arrays are of exactly the same shape, then these operations are smoothly performed.

3. What makes python better than other libraries for numerical computation?

Numerical Python has a fixed-size, homogeneous (fixed-type), multi-dimensional array type and lots of functions for various array operations. The result is a dynamically typed environment for array computing similar to basic Matlab.

4. How does NumPy deal with files?

NumPy introduces a simple file format for ndarray objects. This . npy file stores data, shape, dtype and other information required to reconstruct the ndarray in a disk file such that the array is correctly retrieved even if the file is on another machine with different architecture.

5. Mention the importance of NumPy. empty().

The numpy module of Python provides a function called numpy.empty(). This function is used to create an array without initializing the entries of given shape and type. Just like numpy.zeros(), the numpy.empty() function doesn't set the array values to zero, and it is quite faster than the numpy.zeros(). This function requires the user to set all the values in the array manually and should be used with caution.