1. 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. 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. If the dimensions of two arrays are dissimilar, element-to-element operations are not possible. However, operations on arrays of non-similar shapes is still possible in NumPy, because of the broadcasting capability. The smaller array is broadcast to the size of the larger array so that they have compatible shapes.

   Broadcasting is possible if the following rules are satisfied –

   Array with smaller ndim than the other is prepended with '1' in its shape.

   Size in each dimension of the output shape is maximum of the input sizes in that dimension. An input can be used in calculation, if its size in a particular dimension matches the output size or its value is exactly 1.

   If an input has a dimension size of 1, the first data entry in that dimension is used for all calculations along that dimension.

   A set of arrays is said to be broadcastable if the above rules produce a valid result and one of the following is true –

   Arrays have exactly the same shape.

   Arrays have the same number of dimensions and the length of each dimension is either a common length or 1.

   Array having too few dimensions can have its shape prepended with a dimension of length 1, so that the above stated property is true.

3. SciPy is an open-source scientific computing library for the Python programming language. Since its initial release in 2001, SciPy has become a de facto standard for leveraging scientific algorithms in Python, with over 600 unique code contributors, thousands of dependent packages, over 100,000 dependent repositories and millions of downloads per year. In this work, we provide an overview of the capabilities and development practices of SciPy 1.0 and highlight some recent technical developments.

4. 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.

The ndarray objects can be saved to and loaded from the disk files. The IO functions available   are −

load() and save() functions handle /numPy binary files (with npy extension)

loadtxt() and savetxt() functions handle normal text files

The save() and load() functions accept an additional Boolean parameter allow_pickles. A pickle in Python is used to serialize and de-serialize objects before saving to or reading from a disk file.

savetxt()

The storage and retrieval of array data in simple text file format is done with savetxt() and loadtxt() functions.The savetxt() and loadtxt() functions accept additional optional parameters such as header, footer, and delimiter.

5. 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.

shape: int or tuple of ints

This parameter defines the shape of the empty array, such as (3, 2) or (3, 3).

dtype: data-type(optional)

This parameter defines the data type, which is desired for the output array.

{'C', 'F'}(optional)

This parameter defines the order in which the multi-dimensional array is going to be stored either in row-major or column-major. By default, the order parameter is set to 'C'.

Returns:

This function returns the array of uninitialized data that have the shape, dtype, and order defined in the function.