Summary

Ndarrays and the difference between ndarrays and lists

A ndarray is an array of n dimensions that has data of the same type and size. A ndarray is stored in memory like arrays common in many programming languages are stored in memory, in contiguous memory addresses. Lists, on the other hand, are arrays of pointers that point to random memory addresses of different types.

A computer code with green squares and black arrows

Description automatically generated with medium confidence

A computer memory box with green squares

Description automatically generated with medium confidence

Advantages of using numpy ndarrays over regular lists

* SIMD vector processing unit makes operations on arrays faster.
* More cache utilization, since the data in arrays is contiguous the cache won’t need to load data from the main memory as often.
* No type checking, arrays are always made up of the same data type.

Usages of Numpy

* Machine learning
* Mathematics
* Plotting
* Backend (Pandas, Digital Photography)

Ndarrays attributes:

Shape: the array’s shape is a tuple containing the dimensions of the array. An array of 2 rows and 3 column will have a shape of (2,3).

Dimensions: how many dimensions are in the array

dtype: the data type of the array.

Size: the size of the array

Itemsize: the size of the datatype of the array.

nbytes: how many bytes the array is made of , which is size\* itemsize.

A screenshot of a computer

Description automatically generated

Numpy Capabalities

Arithmetic on arrays can be done much easier. You can add elements of arrays as long as they have the same shape or the one shape could be *broadcasted* to the other.

A screenshot of a computer program

Description automatically generated

Broadcasting is an operation of matching the dimensions of differently shaped arrays in order to be able to perform further operations on those arrays.

A screenshot of a math problem

Description automatically generated

Boolean indexing is indexing through a Boolean array. a[a>2] returns all values in array a where a[i] is greater than 2.

A white paper with black text and numbers

Description automatically generated

Advanced indexing is indexing in an array through the use of lists. For example if a is an array of shape (3,3) then a[[0,1],[1,2]] will return a[0][1] and a[1][2] respectively.

A white rectangular with green and blue text

Description automatically generated

You can even index into arrays using slices, like in lists. For example, a[ : ,k] will return all elements in the kth column.

Array views and copies

A view is an object that shares the same data buffer as the original array. A copy, on the other hand, is a copy of the original array and **does not** share the same data buffer. Modifying a view’s data will cause a change in the original array’s data and vice-versa. Modifying a copy’s data will not cause any change in the original array. The difference between assigning the original array to another variable and creating a view is as follows:

A white rectangular object with a black border

Description automatically generated

A diagram of a data flow

Description automatically generated

Variable a and variable b share the same reference/pointer while variable c is a different object that points to the same data buffer. Also, views can have a different shape from the original array. Array slicing creates views of the original array.

Array functions

The numpy module has a lot of functions that support statistics, linear algebra, array comparison, and much more.

