



PyQB

Monga

NumPy
ndarr
ay
Creation

Programming in Python¹

Mattia Monga

Dip. di Informatica
Università degli Studi di Milano, Italia
mattia.monga@unimi.it

Academic year 2025/26, I semester

¹ © 2025 M. Monga. Creative Commons Attribution — Condividi allo stesso modo 4.0 Internazionale. <http://creativecommons.org/licenses/by-sa/4.0/deed.it>



PyQB

Monga

NumPy
ndarr
ay
Creation

Lecture XIII: NumPy arrays



PyQB

Monga

NumPy
ndarr
ay
Creation

NumPy

NumPy is a third-party library very popular for scientific/numerical programming (<https://numpy.org/>).

- Features familiar to matlab, R, Julia programmers
- The key data structure is the array
 - 1-dimension arrays: vectors
 - 2-dimension arrays: matrices
 - n-dimension arrays

In some languages array is more or less synonym of list: Python distinguishes: lists (mutable, arbitrary elements), arrays (mutable, all elements have the same type), tuples (immutable, fixed length, arbitrary elements).



PyQB

Monga

NumPy
ndarr
ay
Creation

NumPy arrays

The most important data structure in NumPy is ndarray: a (usually fixed-size) sequence of same type elements, organized in one or more dimensions.

<https://numpy.org/doc/stable/reference/arrays.ndarray.html>

Implementation is based on byte arrays: accessing an element (all of the same byte-size) is virtually just the computation of an 'address'.

Why?



PyQB

Monga

NumPy

ndarray
Creation

- using NumPy arrays is often more compact, especially when there's more than one dimension
- faster than lists when the operation can be vectorized
- (slower than lists when you append elements to the end)
- can be used with element of different types but this is less efficient

97

ndarray



PyQB

Monga

NumPy

ndarray
Creation

A ndarray has a dtype (the type of elements) and a shape (the length of the array on each dimensional axis). (Note the jargon: slightly different from linear algebra)

- Since appending is costly, normally they are pre-allocated (zeros, ones, arange, linspace, ...)
- vectorized operations can simplify code (no need for loops) and they are faster with big arrays
- vector indexing syntax (similar to R): very convenient (but you need to learn something new)

98

All the elements must have the same size



PyQB

Monga

NumPy

ndarray
Creation

This is actually a big limitation: the faster access comes with a price in flexibility.

```
>>> np.array(['', '', ''])
array(['', '', ''], dtype='<U1')
>>> np.array(['a', 'bb', 'ccc'])
array(['a', 'bb', 'ccc'], dtype='<U3')
>>> np.array(['a', 'bb', 'cccccccccccccccccccc'])
array(['a', 'bb', 'cccccccccccccccccccc'], dtype='<U21')
```

99