

numpy

Dr. Ilkay Altintas

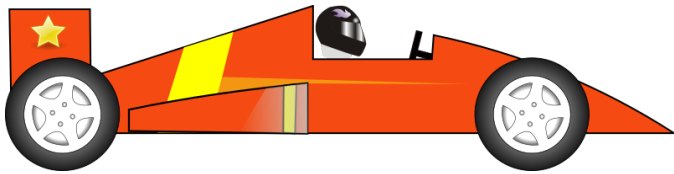
What are the key features of numpy?

- Multi-dimensional arrays
- Built-in array operations
- Simplified, but powerful array interactions → broadcasting
- Integration of other languages (Fortran, C, C++)

$$\begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{bmatrix}$$

Why numpy for data science?

- Speed



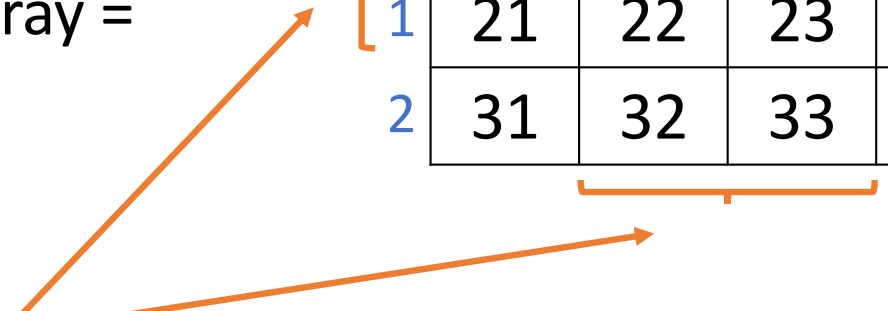
- Functionality



Many packages are built on Numpy!

numpy: ndarray Indexing

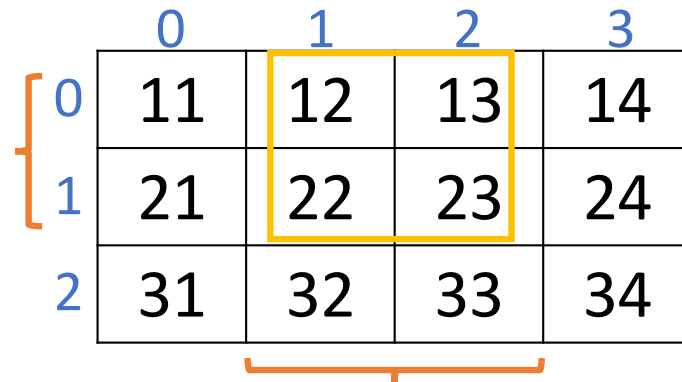
an_array =



	0	1	2	3
0	11	12	13	14
1	21	22	23	24
2	31	32	33	34

a_slice = an_array[:2, 1:3]

an_array =



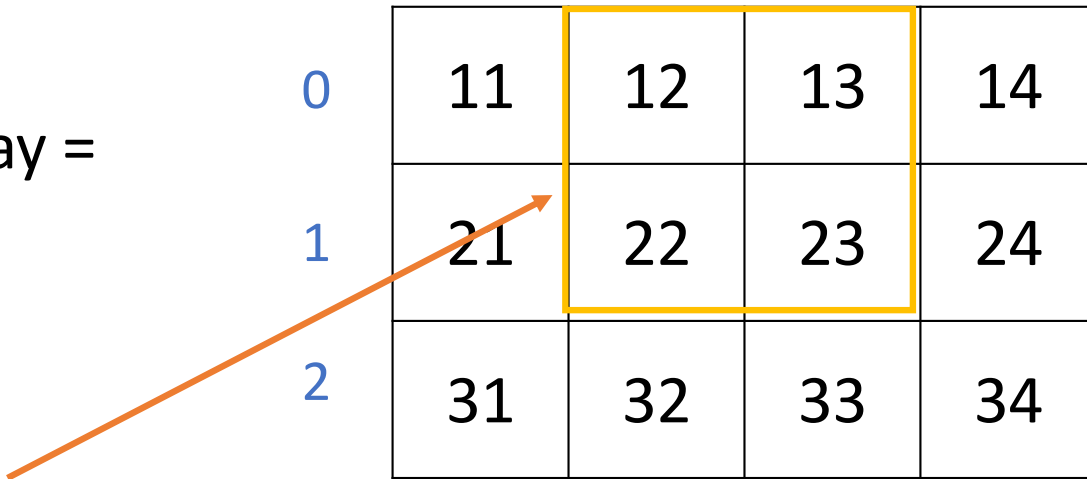
	0	1	2	3
0	11	12	13	14
1	21	22	23	24
2	31	32	33	34

a_slice = an_array[:2, 1:3]

an_array =

a_slice

	0	1	2	3
0	11	12	13	14
1	21	22	23	24
2	31	32	33	34



an_array =

a_slice

	0	1	2	3
0	11	12	13	14
1	21	22	23	24
2	31	32	33	34

	0	1	2	3
		0	1	
an_array[0,1]	0	12	13	14
	1	21	22	23
a_slice[0,0]	2	31	32	33
				34

numpy: Broadcasting

- employ broadcasting to perform operations on different size ndarrays

	0	1	2	3
0				
1				
2				

A

0	1	2	3

B

	0	1	2	3
0	1	2	3	4
1	5	6	7	8
2	9	10	11	12

A

+

0	1	2	3
0	1	0	2

B

	0	1	2	3		0	1	2	3			
0	1	2	3	4	+	0	0	1	0	2	B	
1	5	6	7	8		1	0	1	0	2		B
2	9	10	11	12		2	0	1	0	2		
						A						

	0	1	2	3
0	1	3	3	6
1	5	7	7	10
2	9	11	11	14

Result

Broadcasting rules

“When operating on two arrays, NumPy compares their shapes element-wise. It starts with the trailing dimensions, and works its way forward. **Two dimensions are compatible when**

1. **they are equal, or**
2. **one of them is 1”**

<https://docs.scipy.org/doc/numpy/user/basics.broadcasting.html>