인공지능을 위한

파이썬 코딩 기초

파이썬 모듈 넘파이

김루진 강사

소속

인공지능을 위한

파이썬 코딩 기초

학습 내용

1. 넘파이
2. 넘파이 연산
3. 넘파이 활용
4. 이미지 처리

넘파이(NumPY)

» 파이썬에서 배열과 행렬들을 효율적으로 다룰 수 있게 해주는 라이브러리

파이썬 넘파이로 배열 및 행렬 만들기

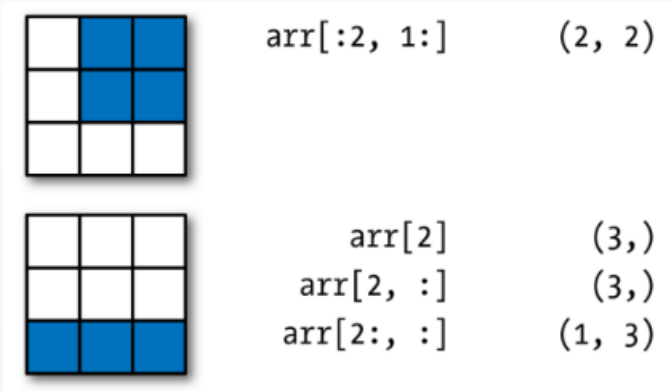
```
import numpy as np
A= np.array([1,3],[2,4])
#1. ndim - 배열의 차원
A.ndim # = 2

#2. shape - 배열 크기
A.shape # = (2,2)

#3. dtype - 원소 자료형
A.dtype # = dtype('int32')

#4. max mean min sum - 최댓값 평균값 최솟값 합계
A.max(),A.mean(),A.min(),A.sum() # = 4, 2.5, 1, 10

#5. flatten() - 1차원 배열 형태로 바꿈
A.flatten() # = array([1,3,2,4])

#6. transpose() - 요소 위치를 주대각선을 기준으로 뒤바꿈
A.transpose() # = array([1,2],[3,4])

#7. dot() - 두 배열의 내적 곱
A.dot(A) # [[7 15],[10 22]]
```

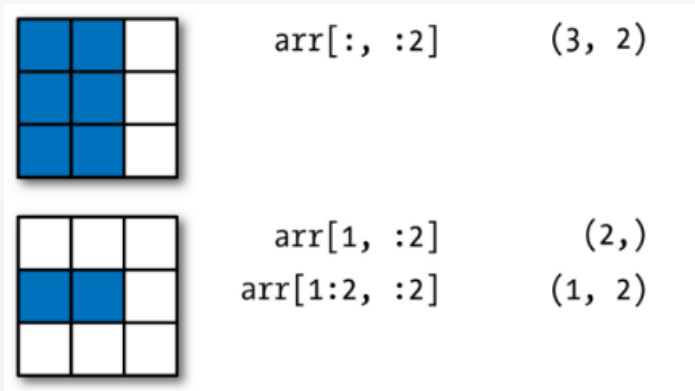
넘파이 – 생성, 슬라이싱

과학계산 전용, 행렬/배열 처리 및 연산, 선형대수(벡터연산, 역함수)

```
import numpy as np
lst = [ [1, 2, 3], [4, 5, 6], [7, 8, 9] ]
arr = np.array(lst)
a = arr[0:2, 0:2]
print(a)
```

출력결과: [[1 2]
[4 5]]





넘파이 배열 연산

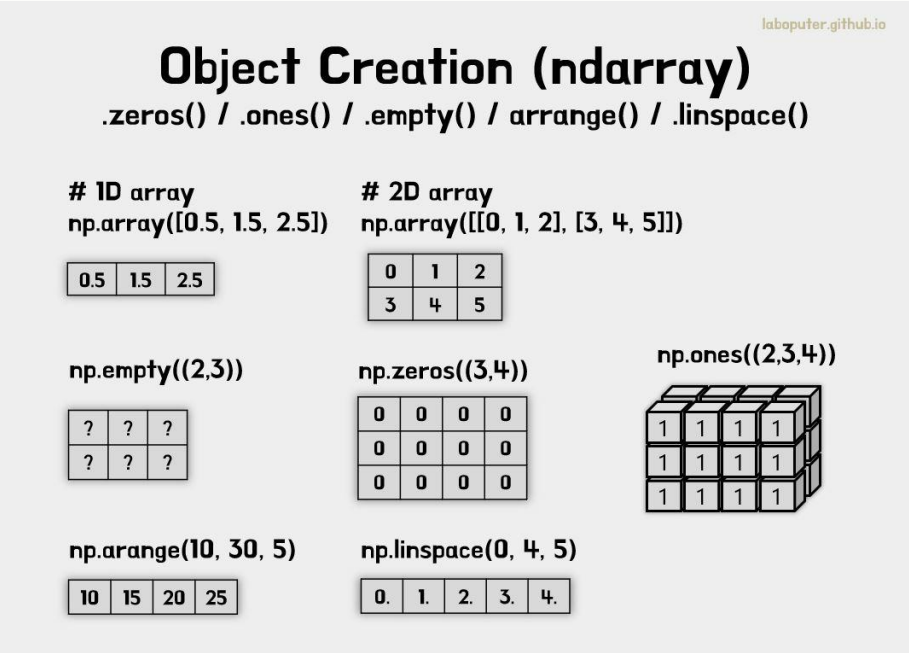
- 넘파이 배열 간의 연산은 반복문을 사용하지 않아도 가능
  - 내부에서 연산이 가능
  - 기본적으로 성분끼리 연산을 하는 데 이러한 연산을 벡터화 계산이라고 한

```
import numpy as np
arr = np.array([[1, 2, 3], [4, 5, 6]])
print(arr)

출력결과: [[1 2 3]
           [4 5 6]]

arr *= arr
print(arr)

출력결과: [[ 1  4  9]
           [16 25 36]]
```

```
arr -= arr
print(arr)

출력결과: [[0 0 0]
           [0 0 0]]

arr2 = np.array([[0, 4, 9], [3, 2, 7]])
print(arr2 > arr)

출력결과: [[False  True  Ture]
           [False False  True]]
(비교연산 가능)
```

넘파이 – 생성

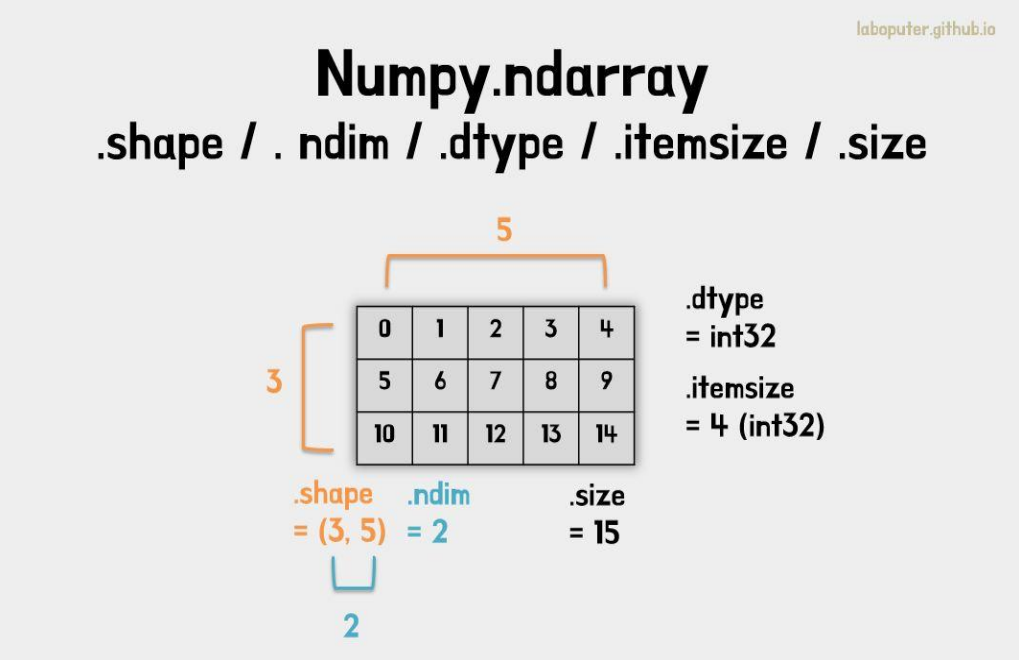
- np.array()를 이용하여 Python에서 사용하는 Tuple(튜플)이나 List(리스트)를 입력으로 numpy.ndarray 생성



출처 https://laboputer.github.io/machine-learning/2020/04/25/numpy-quickstart/

넘파이 – ndarray

numpy.ndarray의 대표적인 속성값

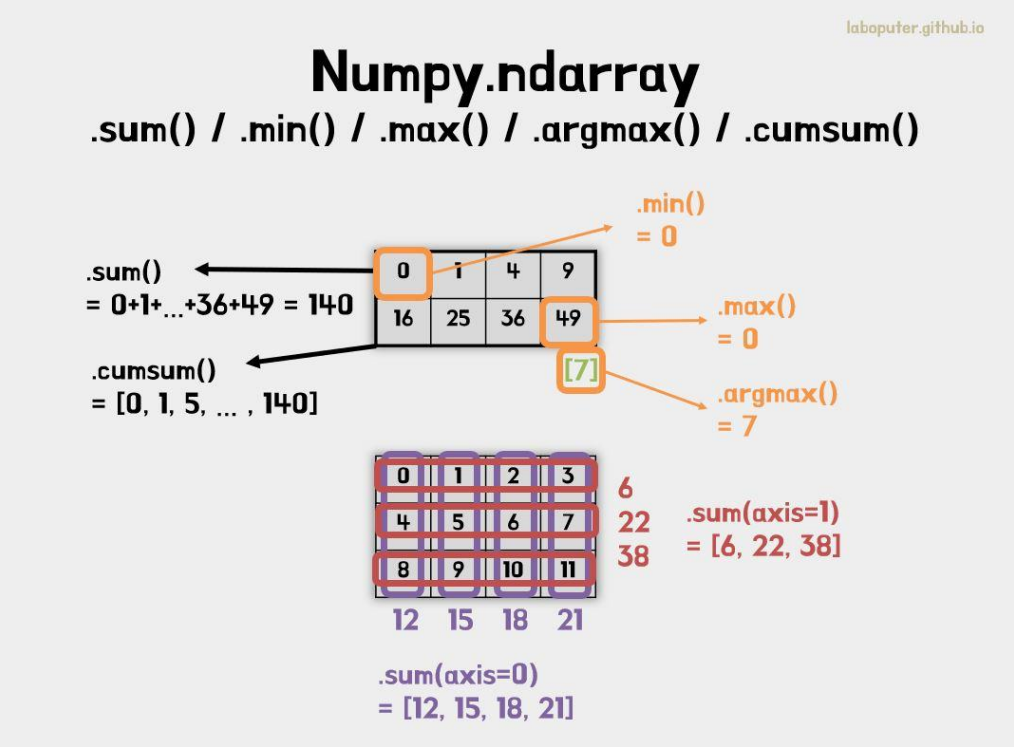


- ndarray.shape : 배열의 각 축(axis)의 크기
- ndarray.ndim : 축의 개수(Dimension)
- ndarray.dtype : 각 요소(Element)의 타입
- ndarray.itemsize : 각 요소(Element)의 타입의 bytes 크기
- ndarray.size : 전체 요소(Element)의 개수

출처 https://laboputer.github.io/machine-learning/2020/04/25/numpy-quickstart/

넘파이 내부연산

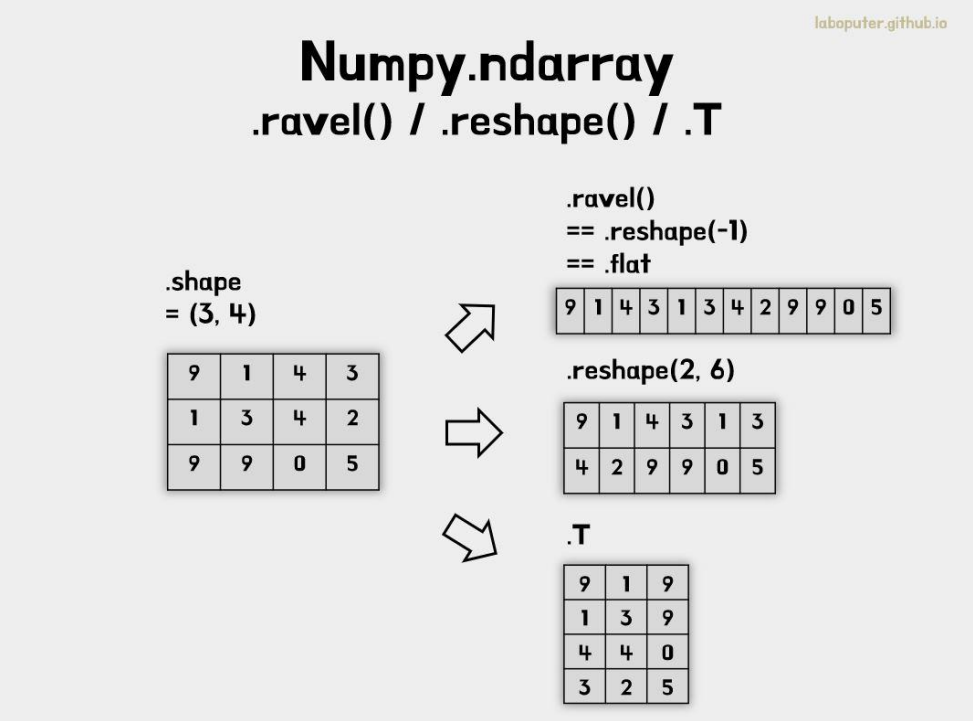
- .sum(), .min(), .max(), .argmax(), .cumsum()와 같은 연산



출처

넘파이 내부연산

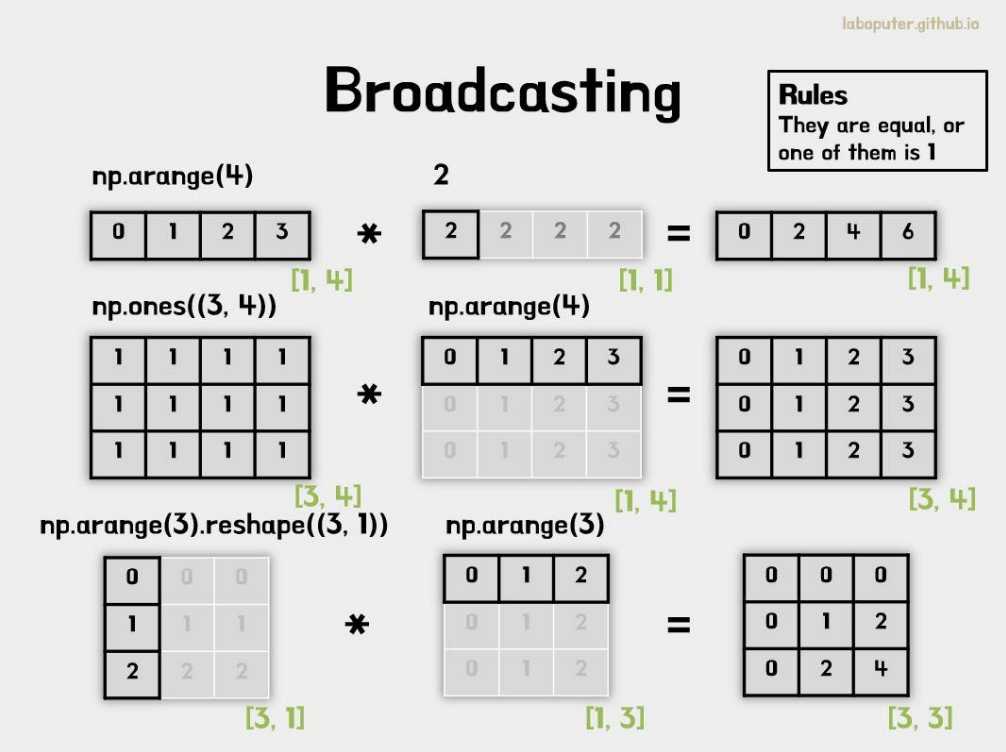
Shape 변경 (Shape Manipulation)



출처

넘파이 브로드캐스팅

이것이 없다면 Shape를 맞춰야 하는 번거로움이 생김



출처

넘파이 이미지 처리

- 이것이 없다면 Shape를 맞춰야 하는 번거로움이 생김

출처

감사합니다.