<11주차. Numpy 함수 정리>

손효지 20175074

1. **생성 함수**

-기본 행렬 생성 함수들, 행렬 생성시 자주 사용함

1) np.full()

np. full((2,2,),7) # 0 대신 다른 값을 넣는 함수

array( [[7 7],

[7 7]])

2) np.eye()

np.eye(2) # 단위행렬( 주 대각 성분이 1이고 나머진 0)생성 함수

array([[ 1, 0],

[0, 1]])

3) np.empty()

np.empty((4,2)) # ones와 zeros와 비슷하나 초기화하지 않음

array ([[ 0.00000000e+000, 6.91240343e-310]

[ 6.91240500e-310, 5.39088070e-317]

[ 5.39084907e-317, 6.91239798e-310]

[ 3.16202013e-322, 6.91239798e-310]])

1. **범용함수(올림 혹은 내림)**

- unfuncs라고 불리는 유니버설 함수는 ndarray 안에 있는 데이터 원소 별로 연산을 수행하는 함수

- 일종의 래퍼 함수로 간단하게 다른 함수에 약간의 기능을 추가하여 사용하는 함수

import numpy as np

a=np.array([-4.62, -2.19, 0, 1.57, 3.40, 4.06])

array([-4.62, -2.19, 0, 1.57, 3.40, 4.06])

np.around(a) #0.5를 기준으로 올림 혹은 내림

array([-5, -2, 0, 2, 3, 4])

1. **지수함수**

-  https://t1.daumcdn.net/cfile/tistory/226D674858C414CC0D인 함수

- NumPy의 np.exp() 함수는 밑(base)이 자연상수 e 인 지수함수 https://t1.daumcdn.net/cfile/tistory/2611623E58C41E8838 로 변환해줌

import numpy as np

x=np.array([0.00001, 1, 2, 4, 10, 100])

array([  1.00000000e-05,   1.00000000e+00,   2.00000000e+00,  
          4.00000000e+00,   1.00000000e+01,   1.00000000e+02])  
 np.exp(x)

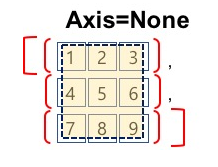
array([  1.00001000e+00,   2.71828183e+00,   7.38905610e+00,  
           5.45981500e+01,   2.20264658e+04,   2.68811714e+43])

1. **집계 함수**

* NumPy의 모든 집계 함수는 집계 함수는 AXIS를 기준으로 계산됨
* 집계함수에 AXIS를 지정하지 않으면 axis=None. axis=None, 0, 1

1. axis=None

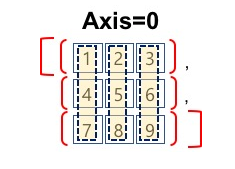
전체 행렬을 하나의 배열로 간주하고 집계 함수의 범위를 전체 행렬로 정의함



1. axis=0

행을 기준으로 각 행의 동일 인덱스의 요소를 그룹으로 함

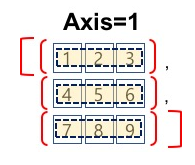
각 그룹을 집계 함수의 범위로 정의



1. axis=1

열을 기준으로 각 열의 요소를 그룹으로 함

각 그룹을 집계 함수의 범위로 정의

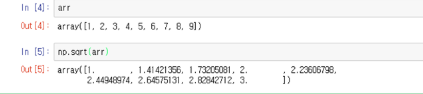


1. **특정계산 함수**

-1부터 9까지 정수 array를 만들고, np.squrt(arr)메소드를 사용

-np.arrange(n)은 0부터 n-1까지, (1,n)은 1부터 n-1까지 정수 array 생성

-np.squrt(arr) 매소드는 각 array 성분에 제곱근을 입힘

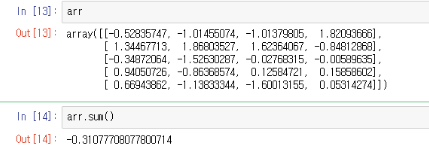


1. **통계함수**

- array 전체성분에 대한 각종 통계량을 계산할 수 있는 함수

- np.이 아닌 array.으로 시작하는 것이 특징

- arr.mean()을 이용해서 전체성분의 평균을 구함





1. **정렬함수**

- 데이터상에서 상위-% 등의 데이터를 구하는 방법

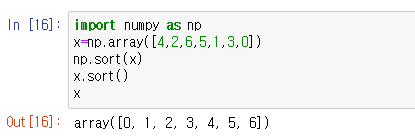
- 각 열을 오름, 내림차순으로 정렬(axis=0)하면, 서로 다른 행간에 성분들이 섞여버리게 되기 때문에, 특정 열을 기준으로 행 단위를 오름차순으로 정렬해야 함

ex) 1차원 배열 정렬 # 원래 배열은 그대로, 정렬 결과 복사본 반환

* np.sort(x)

배열 자체를 정립

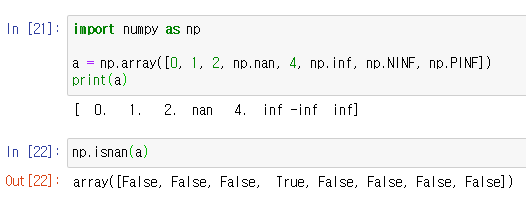
* x.sort()



1. **논리함수**

-참 또는 거짓의 Boolean 값을 변환함

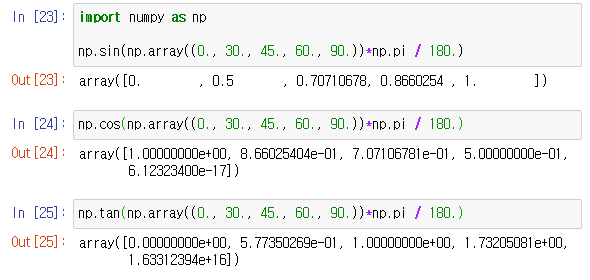
ex) 배열에 nan(Not a Number) 포함 여부 확인 함수



1. **삼각함수**

-시그널 데이터 변환, 벡터 내적 계산에 쓰임

- Python NumPy의 삼각함수는 radian을 사용하기 때문에 degree 를 radian으로 바꿔주기 위해서 degree **\* np.py/180** 을 해줌



1. **부호 판별함수**

-np.sign(x) # 1(positive), 0(zero), -1(negative) 값 반환

