Addition (add):

python

Copy code

import numpy as np

a = np.array([1, 2, 3])

b = np.array([4, 5, 6])

result = np.add(a, b)

print(result) # Output: [5 7 9]

Subtraction (subtract):

python

Copy code

import numpy as np

a = np.array([1, 2, 3])

b = np.array([4, 5, 6])

result = np.subtract(a, b)

print(result) # Output: [-3 -3 -3]

Logarithm (log):

python

Copy code

import numpy as np

a = np.array([1, 10, 100])

result = np.log(a)

print(result) # Output: [0. 2.30258509 4.60517019]

Absolute value (absolute):

python

Copy code

import numpy as np

a = np.array([-1, -2, 3, -4])

result = np.absolute(a)

print(result) # Output: [1 2 3 4]

Sine (sin):

python

Copy code

import numpy as np

a = np.array([0, np.pi/2, np.pi])

result = np.sin(a)

print(result) # Output: [0.00000000e+00 1.00000000e+00 1.22464680e-16]

Cosine (cos):

python

Copy code

import numpy as np

a = np.array([0, np.pi/2, np.pi])

result = np.cos(a)

print(result) # Output: [ 1.000000e+00 6.123234e-17 -1.000000e+00]

Tangent (tan):

python

Copy code

import numpy as np

a = np.array([0, np.pi/4, np.pi/2])

result = np.tan(a)

print(result) # Output: [ 0.00000000e+00 1.00000000e+00 1.63312394e+16]

Greater than comparison (greater):

python

Copy code

import numpy as np

a = np.array([1, 2, 3])

b = np.array([2, 2, 2])

result = np.greater(a, b)

print(result) # Output: [False False True]

Greater than or equal to comparison (greater\_equal):

python

Copy code

import numpy as np

a = np.array([1, 2, 3])

b = np.array([2, 2, 2])

result = np.greater\_equal(a, b)

print(result) # Output: [False True True]

Less than comparison (less):

python

Copy code

import numpy as np

a = np.array([1, 2, 3])

b = np.array([2, 2, 2])

result = np.less(a, b)

print(result) # Output: [ True False False]

Not equal to comparison (not\_equal):

python

Copy code

import numpy as np

a = np.array([1, 2, 3])

b = np.array([2, 2, 2])

result = np.not\_equal(a, b)

print(result) # Output: [ True False True]

Remainder (fmod):

python

Copy code

import numpy as np

a = np.array([10, 8, 15])

b = np.array([3, 5, 6])

result = np.fmod(a, b)

print(result) # Output: [ 1 3 3]

Floor (floor):

python

Copy code

import numpy as np

a = np.array([1.2, 2.7, 3.9])

result = np.floor(a)

print(result) # Output: [1. 2. 3.]

Ceiling (ceil):

python

Copy code

import numpy as np

a = np.array([1.2, 2.7, 3.9])

result = np.ceil(a)

print(result) # Output: [2. 3. 4.]

Truncation (trunc):

python

Copy code

import numpy as np

a = np.array([1.2, 2.7, 3.9])

result = np.trunc(a)

print(result) # Output: [1. 2. 3.]

Mean (mean):

python

Copy code

import numpy as np

a = np.array([1, 2, 3, 4, 5])

result = np.mean(a)

print(result) # Output: 3.0

Median (median):

python

Copy code

import numpy as np

a = np.array([1, 2, 3, 4, 5])

result = np.median(a)

print(result) # Output: 3.0

Mode (mode):

python

Copy code

import numpy as np

a = np.array([1, 2, 2, 3, 4, 4, 4])

result = np.mode(a)

print(result) # Output: [4]

Standard deviation (std):

python

Copy code

import numpy as np

a = np.array([1, 2, 3, 4, 5])

result = np.std(a)

print(result) # Output: 1.4142135623730951

Variance (var):

python

Copy code

import numpy as np

a = np.array([1, 2, 3, 4, 5])

result = np.var(a)

print(result) # Output: 2.0

These are simple examples to demonstrate the usage of each operation in NumPy. You can apply these operations to more complex arrays and use them in various mathematical and statistical calculations.

User

simplay explain all the above

ChatGPT

Certainly! Here's a simplified explanation of each operation mentioned:

Addition (add): Adds corresponding elements of two arrays together.

Subtraction (subtract): Subtracts corresponding elements of one array from another.

Logarithm (log): Computes the natural logarithm (base e) of each element in the array.

Absolute value (absolute): Returns the absolute (positive) value of each element in the array.

Sine (sin): Computes the sine of each element in the array.

Cosine (cos): Computes the cosine of each element in the array.

Tangent (tan): Computes the tangent of each element in the array.

Greater than comparison (greater): Compares corresponding elements of two arrays and returns True if the first array element is greater than the second element.

Greater than or equal to comparison (greater\_equal): Compares corresponding elements of two arrays and returns True if the first array element is greater than or equal to the second element.

Less than comparison (less): Compares corresponding elements of two arrays and returns True if the first array element is less than the second element.

Not equal to comparison (not\_equal): Compares corresponding elements of two arrays and returns True if the elements are not equal.

Remainder (fmod): Computes the remainder of division element-wise between two arrays.

Floor (floor): Rounds down each element in the array to the nearest integer.

Ceiling (ceil): Rounds up each element in the array to the nearest integer.

Truncation (trunc): Truncates each element in the array towards zero.

Mean (mean): Computes the arithmetic mean (average) of the array's elements.

Median (median): Computes the median (middle value) of the array's elements.

Mode (mode): Computes the mode (most frequent value) of the array's elements.

Standard deviation (std): Computes the standard deviation of the array's elements, a measure of the spread of data.

Variance (var): Computes the variance of the array's elements, a measure of the average squared deviation from the mean.