How do you calculate the dot product using NumPy?

Authored by

stats writer

May 3, 2024

The dot product is a mathematical operation that is used to calculate the sum of the products of corresponding elements in two vectors. In order to calculate the dot product using NumPy, one can use the dot() method provided by the NumPy library. This method takes in two arrays as input and returns the dot product as a single scalar value. The arrays must have the same dimensions in order to perform the calculation. This method is efficient and convenient for performing dot product calculations in scientific and data analysis applications.

Calculate Dot Product Using NumPy

Given vector *a* = and vector *b* = , the dot product of the vectors, denoted as a · b, is given by:

a · b = a1 * b1 + a2 * b2 + a3 * b3

For example, if *a* = and *b* = , then the dot product of *a* and *b* would be equal to:

a · b = 2*4 + 5*3 + 6*2

a · b = 8 + 15 + 12

a · b = 35

Simply put, the dot product is the sum of the products of the corresponding entries in two vectors.

In Python, you can use the numpy.dot() function to quickly calculate the dot product between two vectors:

import numpy as np
np.dot(a, b)

The following examples show how to use this function in practice.

Example 1: Calculate Dot Product Between Two Vectors

The following code shows how to use numpy.dot() to calculate the dot product between two vectors:

import numpy as np

#define vectors
a =
b =

#calculate dot product between vectors
np.dot(a, b)

33

Here is how this value was calculated:

a · b = 7*1 + 2*4 + 2*9a · b = 7 + 8 + 18a · b = 33

Example 2: Calculate Dot Product Between Two Columns

The following code shows how to use numpy.dot() to calculate the dot product between two columns in a pandas DataFrame:

```
import pandas as pd
import numpy as np

#create DataFrame
df = pd.DataFrame({'A': ,
'B': ,
'C': })

#view DataFrame
df

A B C
0 4 5 11
1 6 7 8
2 7 7 9
3 7 2 6
4 9 2 1

#calculate dot product between column A and column C
np.dot(df.A, df.C)
```

206

Here is how this value was calculated:

A · C = 4*11 + 6*8 + 7*9 + 7*6 + 9*1A · C = 44 + 48 + 63 + 42 + 9A · C = 206

Note: Keep in mind that Python will throw an error if the two vectors you're calculating the dot product for have different lengths.