How to Convert a NumPy Array to Pandas DataFrame

Authored by

stats writer

December 17, 2025

Numpy arrays can be easily converted to Pandas DataFrames by using the DataFrame constructor provided by Pandas. This constructor takes a Numpy array as its argument and converts it into a Pandas DataFrame. The DataFrame constructor also allows optional arguments to be passed in to customize the resulting DataFrame. This includes arguments such as column names, index names, and even data types. Once the DataFrame is created, the data can be manipulated and visualized using the Pandas library.

You can use the following syntax to convert a NumPy array into a pandas DataFrame:

#create NumPy array
data = np.array(,])

#convert NumPy array to pandas DataFrame
df = pd.DataFrame(data=data)

The following example shows how to use this syntax in practice.

Example: Convert NumPy Array to Pandas DataFrame

Suppose we have the following NumPy array:

import numpy as np

#create NumPy array
data = np.array(,])

#print class of NumPy array
type(data)

numpy.ndarray

We can use the following syntax to convert the NumPy array into a pandas DataFrame:

import pandas as pd

#convert NumPy array to pandas DataFrame
df = pd.DataFrame(data=data)

#print DataFrame
print(df)

0 1 2 3 4

0 1 7 6 5 6
1 4 4 4 3 1

#print class of DataFrame
type(df)

pandas.core.frame.DataFrame

Specify Row & Column Names for Pandas DataFrame

We can also specify row names and column names for the DataFrame by using the **index** and **columns** arguments, respectively.

#convert array to DataFrame and specify rows & columns
df = pd.DataFrame(data=data, index=, columns=)

#print the DataFrame
print(df)

A B C D E
r1 1 7 6 5 6
r2 4 4 4 3 1

How to Add a Numpy Array to a Pandas DataFrame
How to Drop the Index Column in Pandas
Pandas: Select Rows Where Value Appears in Any Column