How can I use the corrwith() function in Pandas? Can you provide some examples?

Authored by

stats writer

June 26, 2024

The corrwith() function is a useful tool in the Pandas library that allows users to calculate the correlation between two variables in a dataset. This function takes in a column or series of values as its argument and returns a correlation value between that column and all other columns in the dataset. This can be used to quickly identify patterns and relationships between different variables in the data. To use the corrwith() function, simply pass in the desired column or series and specify the correlation method to be used. For example, "pearson" for Pearson correlation or "spearman" for Spearman correlation. Some examples of using the corrwith() function could be analyzing the correlation between a company's revenue and its marketing expenses, or the correlation between a student's grades and their study hours. In summary, the corrwith() function is a powerful tool for data analysis and can provide valuable insights into the relationships between variables in a dataset.

Use corrwith() in Pandas (With Examples)

You can use the corrwith() function in pandas to calculate the pairwise correlation between numerical columns *with the same name* in two different pandas DataFrames.

This function uses the following basic syntax:

df1.corrwith(df2)

Note: This function is different than the corr() function, which is used to calculate the correlation between two numerical columns within the same DataFrame.

The following example shows how to use the corrwith() function in practice.

Example: How to Use corrwith() in Pandas

Suppose we have the following two pandas DataFrames:

```
import pandas as pd

#create first DataFrame
df1 = pd.DataFrame({'team': ,
'points': ,
'assists': ,
'rebounds': })

print(df1)

team points assists rebounds
0 A 18 4 10
1 B 22 5 6
2 C 29 5 4
3 D 25 4 6
4 E 14 8 3
5 F 11 12 5

#create second DataFrame
df2 = pd.DataFrame({'team': ,
'points': ,
```

'assists': ,
'rebs': })

print(df2)

team points assists rebs
0 A 22 15 4
1 B 25 13 11
2 C 27 8 12
3 D 35 8 8
4 E 25 5 7
5 F 20 8 10

We can use the corrwith() function to calculate the correlation between the numeric columns with the same names in the two DataFrames:

#calculate correlation between numeric columns with same names in each DataFrame
df1.corrwith(df2)

points 0.677051
assists -0.478184
rebounds NaN
rebs NaN

dtype: float64

From the output we can see:

The correlation between the values in the points columns in the two DataFrames is 0.677.The correlation between the values in the assists columns in the two DataFrames is -0.478.

Since the column names rebounds and rebs didn't exist in both DataFrames, a value of NaN is returned for each of these columns.

Note # 1: By default, the corrwith() function calculates the Pearson correlation coefficient between columns, but you can also specify method='kendall' or method='spearman' to instead calculate a different type of correlation coefficient.

Note #2: You can find the complete documentation for the corrwith() function .

The following tutorials explain how to perform other common operations in pandas: