How can I calculate the difference between two times using Pandas?

Authored by

stats writer

June 26, 2024

Pandas is a popular Python library used for data analysis and manipulation. It offers a variety of tools for working with time-related data, including the ability to calculate the difference between two time values. This can be done by utilizing the built-in functions and methods provided by Pandas, such as the `pd.to_datetime()` function to convert time values into a Pandas datetime object, and the `pd.Timedelta()` function to calculate the difference between two datetime objects. By using these functions, users can easily and accurately calculate the time difference between two given time values, making it a useful tool for time-based data analysis and calculations.

Pandas: Calculate a Difference Between Two Times

You can use the following syntax to calculate a difference between two times in a pandas DataFrame:

```
#calculate time difference in hours
df = (df.end_time - df.start_time) / pd.Timedelta(hours=1)

#calculate time difference in minutes
df = (df.end_time - df.start_time) / pd.Timedelta(minutes=1)

#calculate time difference in seconds
df = (df.end_time - df.start_time) / pd.Timedelta(seconds=1)
```

This particular example calculates the difference between the times in the end_time and start_time columns of some pandas DataFrame.

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

Example: Calculate Difference Between Two Times in Pandas

Suppose we have the following pandas DataFrame:

```
import pandas as pd

#create DataFrame
df=pd.DataFrame({'start_time':pd.date_range(start='5/25/2020',periods=6,freq='15min'),
'end_time':pd.date_range(start='5/26/2020',periods=6,freq='30min')})

#view DataFrame
print(df)

start_time end_time
0 2020-05-25 00:00:00 2020-05-26 00:00:00
1 2020-05-25 00:15:00 2020-05-26 00:30:00
2 2020-05-25 00:30:00 2020-05-26 01:00:00
3 2020-05-25 00:45:00 2020-05-26 01:30:00
4 2020-05-25 01:00:00 2020-05-26 02:00:00
5 2020-05-25 01:15:00 2020-05-26 02:30:00
```

We can use the following syntax to calculate the time difference between the start_time and end_time columns in terms of hours, minutes, and seconds:

```
#calculate time difference in hours
df = (df.end_time - df.start_time) / pd.Timedelta(hours=1)

#calculate time difference in minutes
df = (df.end_time - df.start_time) / pd.Timedelta(minutes=1)

#calculate time difference in seconds
df = (df.end_time - df.start_time) / pd.Timedelta(seconds=1)

#view updated DataFrameprint(df)

start_time end_time hours_diff min_diff sec_diff
0 2020-05-25 00:00:00 2020-05-26 00:00:00 24.00 1440.0 86400.0
1 2020-05-25 00:15:00 2020-05-26 00:30:00 24.25 1455.0 87300.0
2 2020-05-25 00:30:00 2020-05-26 01:00:00 24.50 1470.0 88200.0
```

3 2020-05-25 00:45:00 2020-05-26 01:30:00 24.75 1485.0 89100.0
4 2020-05-25 01:00:00 2020-05-26 02:00:00 25.00 1500.0 90000.0
5 2020-05-25 01:15:00 2020-05-26 02:30:00 25.25 1515.0 90900.0

The new columns contain the time differences between the start_time and end_time columns in various units.

For example, consider the first row:

The difference between the start time and end time is 24 hours.The difference between the start time and end time is 1,440 minutes.The difference between the start time and end time is 86,400 seconds.

Note that in this example, the start_time and end_time columns are already formatted as datetimes.

If your time columns are instead currently formatted as strings, you can use pd.to_datetime to first convert each column to a datetime format before calculating the difference between the times:

#convert columns to datetime format
df] = df].apply(pd.to_datetime)

You can then proceed to calculate the time differences between the columns since they are both now in a datetime format that pandas can recognize.