How do I calculate a rolling mean in Pandas?

Authored by

stats writer

April 24, 2024

Calculating a rolling mean in Pandas refers to the process of finding the average value of a dataset over a specified window of time. This is commonly used to smooth out fluctuations and identify long-term trends in time series data. To calculate a rolling mean in Pandas, one can use the built-in function "rolling()" and specify the desired window size. This will create a new column with the rolling mean values, allowing for further analysis and visualization of the data.

Calculate a Rolling Mean in Pandas

A rolling mean is simply the mean of a certain number of previous periods in a time series.

To calculate the rolling mean for one or more columns in a pandas DataFrame, we can use the following syntax:

df.rolling(rolling_window).mean()

This tutorial provides several examples of how to use this function in practice.

Example: Calculate the Rolling Mean in Pandas

Suppose we have the following pandas DataFrame:

import numpy as np
import pandas as pd

#make this example reproducible

```
np.random.seed(0)

#create dataset
period = np.arange(1, 101, 1)
leads = np.random.uniform(1, 20, 100)
sales = 60 + 2*period + np.random.normal(loc=0, scale=.5*period, size=100)
df = pd.DataFrame({'period': period, 'leads': leads, 'sales': sales})

#view first 10 rows
df.head(10)
```

```
period leads sales
0 1 11.427457 61.417425
1 2 14.588598 64.900826
2 3 12.452504 66.698494
3 4 11.352780 64.927513
4 5 9.049441 73.720630
5 6 13.271988 77.687668
6 7 9.314157 78.125728
7 8 17.943687 75.280301
8 9 19.309592 73.181613
9 10 8.285389 85.272259
```

We can use the following syntax to create a new column that contains the rolling mean of 'sales' for the previous 5 periods:

```
#find rolling mean of previous 5 sales periods
df = df.rolling(5).mean()

#view first 10 rows
df.head(10)
```

```
period leads sales rolling_sales_5
0 1 11.427457 61.417425 NaN
1 2 14.588598 64.900826 NaN
2 3 12.452504 66.698494 NaN
3 4 11.352780 64.927513 NaN
4 5 9.049441 73.720630 66.332978
5 6 13.271988 77.687668 69.587026
6 7 9.314157 78.125728 72.232007
7 8 17.943687 75.280301 73.948368
8 9 19.309592 73.181613 75.599188
9 10 8.285389 85.272259 77.909514
```

We can manually verify that the rolling mean sales displayed for period 5 is the mean of the previous 5 periods:

Rolling mean at period 5: (61.417+64.900+66.698+64.927+73.720)/5 = 66.33

We can use similar syntax to calculate the rolling mean of multiple columns:

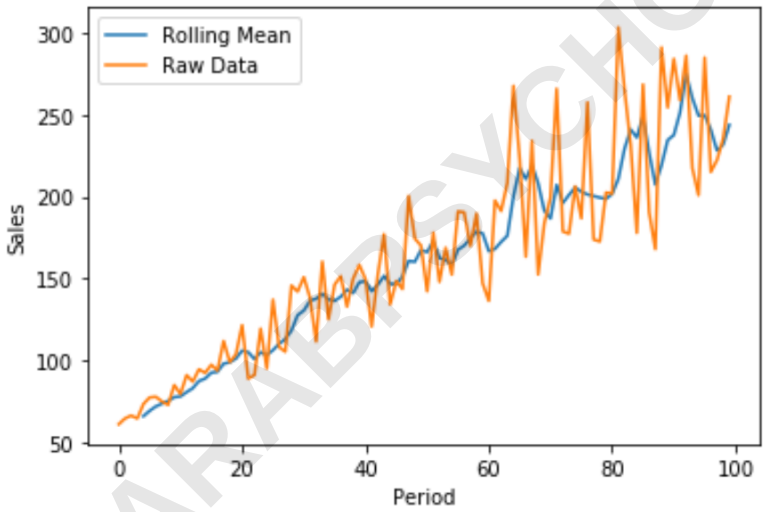
```
#find rolling mean of previous 5 leads periods
df = df.rolling(5).mean()

#find rolling mean of previous 5 leads periods
df = df.rolling(5).mean()

#view first 10 rows
df.head(10)
```

```
period leads sales rolling_sales_5 rolling_leads_5
0 1 11.427457 61.417425 NaN NaN
1 2 14.588598 64.900826 NaN NaN
2 3 12.452504 66.698494 NaN NaN
3 4 11.352780 64.927513 NaN NaN
4 5 9.049441 73.720630 66.332978 11.774156
5 6 13.271988 77.687668 69.587026 12.143062
6 7 9.314157 78.125728 72.232007 11.088174
7 8 17.943687 75.280301 73.948368 12.186411
8 9 19.309592 73.181613 75.599188 13.777773
```

9 10 8.285389 85.272259 77.909514 13.624963

We can also create a quick line plot using Matplotlib to visualize the raw sales compared to the rolling mean of sales:

import matplotlib.pyplotas pltplt.plot(df, label='Rolling Mean')plt.plot(df, label='Raw Data')plt.legend()plt.ylabel('Sales')plt.xlabel('Period')plt.show()



The blue line displays the 5-period rolling mean of sales and the orange line displays the raw sales data.

The following tutorials explain how to perform other

common tasks in pandas: