

How can I calculate a moving average by group in Pandas?

Authored by
stats writer

July 1, 2024

RECOMMENDED CITATION

stats writer (2024). *How can I calculate a moving average by group in Pandas?*.

PSYCHOLOGICAL SCALES. Retrieved from <https://scales.arabpsychology.com/?p=164884>

The process of calculating a moving average by group in Pandas involves using the `groupby` function to group the data by a specific variable, then applying the `rolling` function to calculate the moving average for each group. This allows for the calculation of the average of a specified number of consecutive values within each group, providing a better understanding of the overall trend and pattern of the data. This method is commonly used in data analysis and can be easily implemented using the powerful functions available in the Pandas library.

Pandas: Calculate a Moving Average by Group

You can use the following basic syntax to calculate a moving average by group in pandas:

```
#calculate 3-period moving average of 'values' by 'group'  
df.groupby('group').transform(lambda x: x.rolling(3, 1).mean())
```

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

Example: Calculate Moving Average by Group in Pandas

Suppose we have the following pandas DataFrame that shows the total sales made by two stores during five sales periods:

```
import pandas as pd
```

```
#create DataFrame
```

```
df = pd.DataFrame({'store': ,  
'period': ,  
'sales': })
```

```
#view DataFrame
```

```
df
```

```
store period sales
```

```
0 A 1 7
```

```
1 A 2 7
```

```
2 A 3 9
```

```
3 A 4 13
```

```
4 A 5 14
```

```
5 B 1 13
```

```
6 B 2 13
```

```
7 B 3 19
```

```
8 B 4 20
```

```
9 B 5 26
```

We can use the following code to calculate a 3-day moving average of sales for each store:

```
#calculate 3-day moving average of sales by store
```

```
df = df.groupby('store').transform(lambda x: x.rolling(3,
```

1).mean())

#view updated DataFrame

df

store period sales ma

0 A 1 7 7.000000

1 A 2 7 7.000000

2 A 3 9 7.666667

3 A 4 13 9.666667

4 A 5 14 12.000000

5 B 1 13 13.000000

6 B 2 13 13.000000

7 B 3 19 15.000000

8 B 4 20 17.333333

9 B 5 26 21.666667

Note: x.rolling(3, 1) means to calculate a 3-period moving average and require 1 as the minimum number of periods.

The 'ma' column shows the 3-day moving average of sales for each store.

To calculate a different moving average, simply change

the value in the `rolling()` function.

For example, we could calculate the 2-day moving average of sales for each store instead:

```
#calculate 2-day moving average of sales by store
df = df.groupby('store').transform(lambda x: x.rolling(2,
1).mean())
```

```
#view updated DataFrame
```

```
df
```

```
store period sales ma
```

```
0 A 1 7 7.0
```

```
1 A 2 7 7.0
```

```
2 A 3 9 8.0
```

```
3 A 4 13 11.0
```

```
4 A 5 14 13.5
```

```
5 B 1 13 13.0
```

```
6 B 2 13 13.0
```

```
7 B 3 19 16.0
```

```
8 B 4 20 19.5
```

```
9 B 5 26 23.0
```

Additional Resources

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

ARABPSYCHOLOGY.COM