

# How do I calculate an Exponential Moving Average in Pandas?

Authored by  
**stats writer**

April 18, 2024

## RECOMMENDED CITATION

stats writer (2024). *How do I calculate an Exponential Moving Average in Pandas?*. PSYCHOLOGICAL SCALES. Retrieved from <https://scales.arabpsychology.com/?p=136767>

The process of calculating an Exponential Moving Average in Pandas involves using a mathematical formula to determine the average value of a dataset over a specified time period. This is done by giving more weight to more recent data points, rather than giving equal weight to all data points. This method is commonly used in finance and stock market analysis to track trends and smooth out fluctuations in data. The Exponential Moving Average is calculated using the Pandas library in Python, which provides a user-friendly and efficient way of performing this calculation. By using this method, users can obtain a more accurate representation of the overall trend in their data.

## Calculate an Exponential Moving Average in Pandas

**In time series analysis, a moving average is simply the average value of a certain number of previous periods.**

**An exponential moving average is a type of moving average that gives more weight to recent observations, which means it's able to capture recent trends more quickly.**

**This tutorial explains how to calculate an exponential moving average for a column of values in a pandas DataFrame.**

**Example: Exponential Moving Average in Pandas**

**Suppose we have the following pandas DataFrame:**

```
import pandas as pd
```

```
#create DataFrame
```

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

```
#view DataFrame
```

```
df
```

```
period sales
```

```
0 1 25
```

```
1 2 20
```

```
2 3 14
```

```
3 4 16
```

```
4 5 27
```

```
5 6 20
```

```
6 7 12
```

```
7 8 15
```

```
8 9 14
```

```
9 10 19
```

We can use the `pandas.DataFrame.ewm()` function to calculate the exponentially weighted moving average for a certain number of previous periods.

For example, here's how to calculate the exponentially weighted moving average using the four previous periods:

```
#create new column to hold 4-day exponentially  
weighted moving average
```

```
df = df.ewm(span=4, adjust=False).mean()
```

```
#view DataFrame
```

```
df
```

```
period sales 4dayEWM
```

```
0 1 25 25.000000
```

```
1 2 20 23.000000
```

```
2 3 14 19.400000
```

```
3 4 16 18.040000
```

```
4 5 27 21.624000
```

```
5 6 20 20.974400
```

```
6 7 12 17.384640
```

```
7 8 15 16.430784
```

```
8 9 14 15.458470
```

```
9 10 19 16.875082
```

**We can also use the matplotlib library to visualize the sales compared to the 4-day exponentially weighted moving average:**

```
import matplotlib.pyplot as plt
```

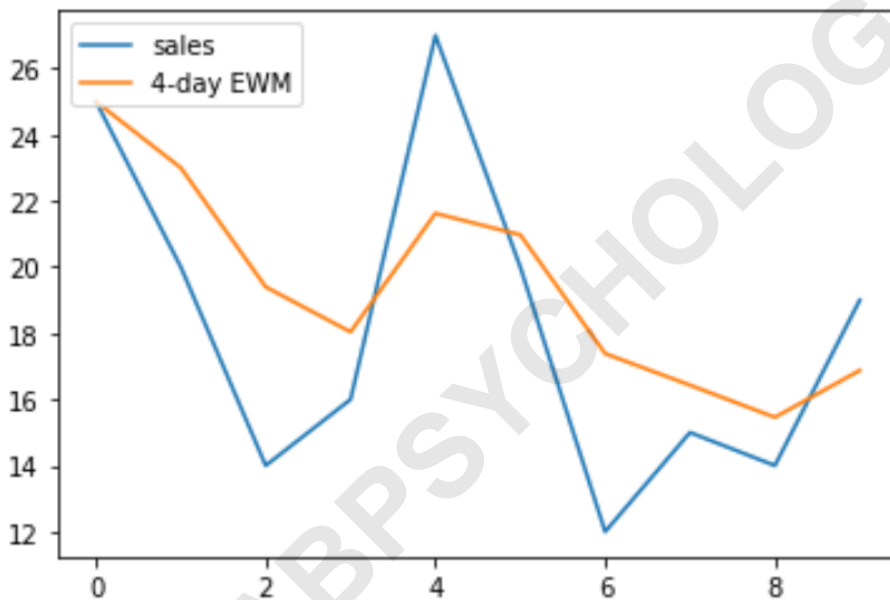
```
#plot sales and 4-day exponentially weighted moving average
```

```
plt.plot(df, label='Sales')
```

```
plt.plot(df, label='4-day EWM')
```

```
#add legend to plot
```

```
plt.legend(loc=2)
```



**[How to Calculate Moving Averages in Python](#)**

**[How to Calculate the Mean of Columns in Pandas](#)**

**[How to Calculate Autocorrelation in Python](#)**