

How can I plot a time series using Pandas, with an example?

Authored by
stats writer

June 25, 2024

RECOMMENDED CITATION

stats writer (2024). *How can I plot a time series using Pandas, with an example?*.

PSYCHOLOGICAL SCALES. Retrieved from <https://scales.arabpsychology.com/?p=151560>

Pandas is a popular data analysis library in Python that provides powerful tools for working with structured data, including time series data. To plot a time series using Pandas, you can use the built-in function "plot" which utilizes the Matplotlib library. This function allows you to visualize the patterns and trends of your time series data in a clear and concise manner. An example of plotting a time series using Pandas would involve importing a dataset containing time series data, converting it into a Pandas dataframe, and then using the "plot" function to create a line graph. This would enable you to easily identify any changes or patterns in the data over time.

Plot a Time Series in Pandas (With Example)

You can use the following syntax to plot a time series in pandas:

```
df.plot(x='date', y='sales')
```

This particular example creates a time series plot using the column called date for the x-axis and the column called sales for the y-axis.

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

Example: How to Plot a Time Series in Pandas

Suppose we have the following pandas DataFrame that shows the total sales made by some store on various days:

```
import pandas as pd
```

```
#create DataFrame
```

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

```
#convert date column to datetime format
```

```
df = pd.to_datetime(df)
```

```
#view DataFrame
```

```
print(df)
```

```
date sales
```

```
0 2023-10-01 99
```

```
1 2023-10-02 104
```

```
2 2023-10-03 110
```

```
3 2023-10-04 140
```

```
4 2023-10-05 130
```

```
5 2023-10-06 122
```

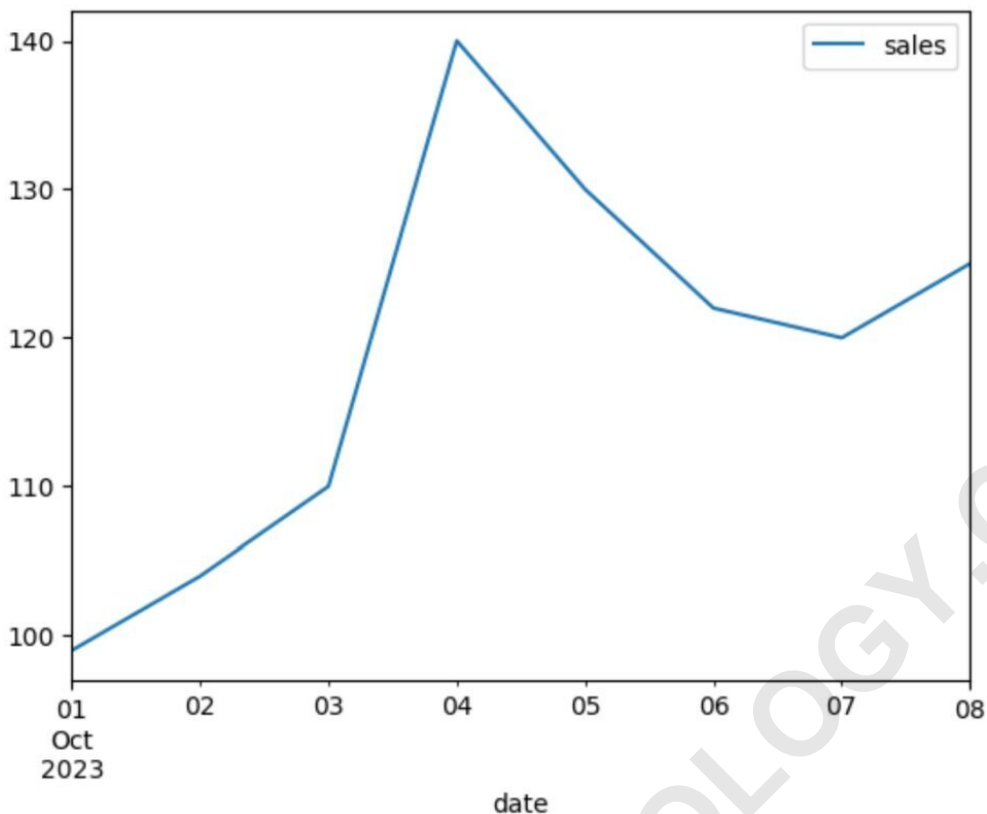
```
6 2023-10-07 120
```

```
7 2023-10-08 125
```

We can use the following syntax to create a time series plot to visualize the total sales by day:

```
#create time series plot
```

```
df.plot(x='date', y='sales')
```



The x-axis shows the date and the y-axis shows the total sales.

We can also use the following arguments in the plot() function to customize the appearance of the line in the plot:

linewidth: The width of the line

color: The color of the line

linestyle: The style of the line

legend: Whether or not to display the legend in the plot

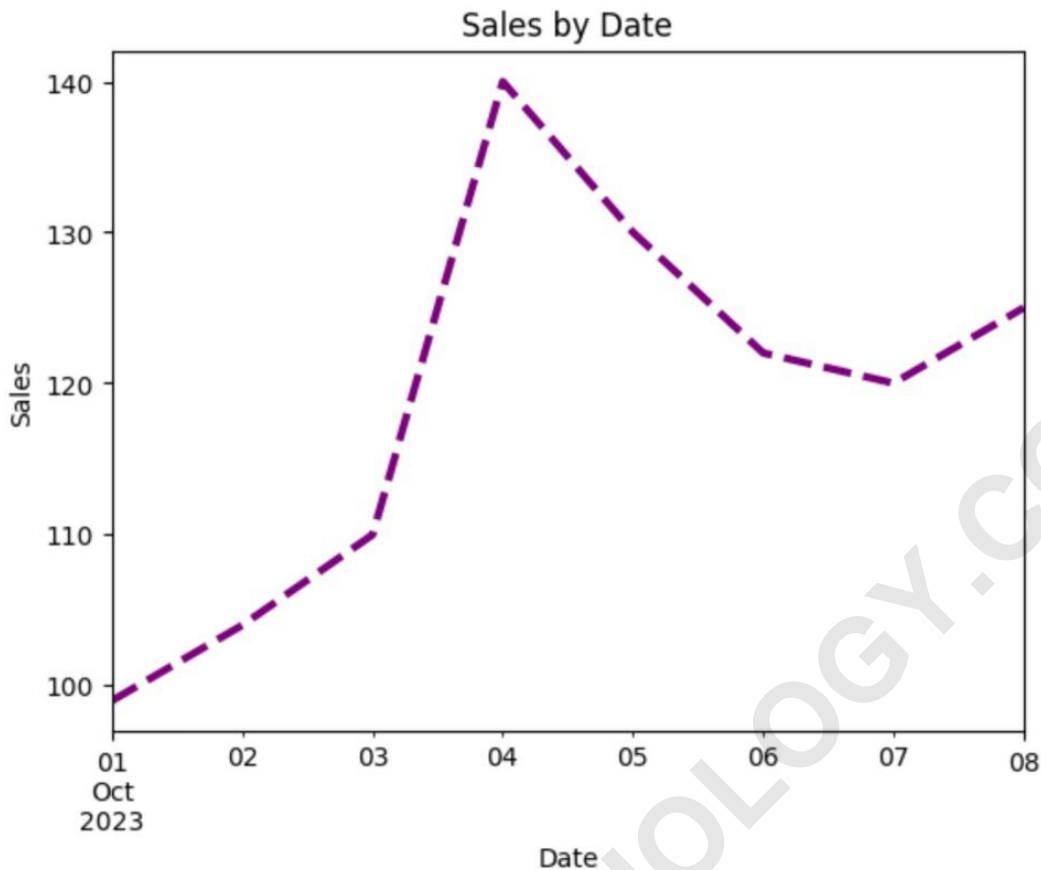
We can also use the `title()`, `xlabel()` and `ylabel()` functions from `matplotlib` to add a title and axis-labels to the plot.

The following code shows how to do so:

```
import matplotlib.pyplot as plt

#create time series plot with custom line
df.plot(x='date', y='sales',
linewidth=3, color='purple', linestyle='dashed',
legend=False)

#add title and axis labels to plot
plt.title('Sales by Date')
plt.xlabel('Date')
plt.ylabel('Sales')
```



Feel free to play around with the various arguments to make the time series plot appear however you'd like.

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