How can I create multiple matplotlib plots in one figure?

Authored by

stats writer

May 2, 2024

Creating multiple matplotlib plots in one figure allows you to display and compare multiple data sets or visualizations within a single image. This can be achieved by using the subplot function to divide the figure into multiple axes, each representing a separate plot. This method provides a convenient way to present complex data in a concise and organized manner, making it easier to interpret and analyze. Additionally, it allows for easier customization and editing of the overall figure, as all plots are contained within one image. Overall, creating multiple matplotlib plots in one figure is a useful technique for visualizing and understanding data in a systematic and efficient way.

Create Multiple Matplotlib Plots in One Figure

You can use the following syntax to create multiple Matplotlib plots in one figure:

```
import matplotlib.pyplotas plt

#define grid of plots
fig, axs = plt.subplots(nrows=2, ncols=1)

#add data to plots
axs.plot(variable1, variable2)
axs.plot(variable3, variable4)
```

The following examples show how to use this function in practice.

Example 1: Stack Plots Vertically

The following code shows how to create three

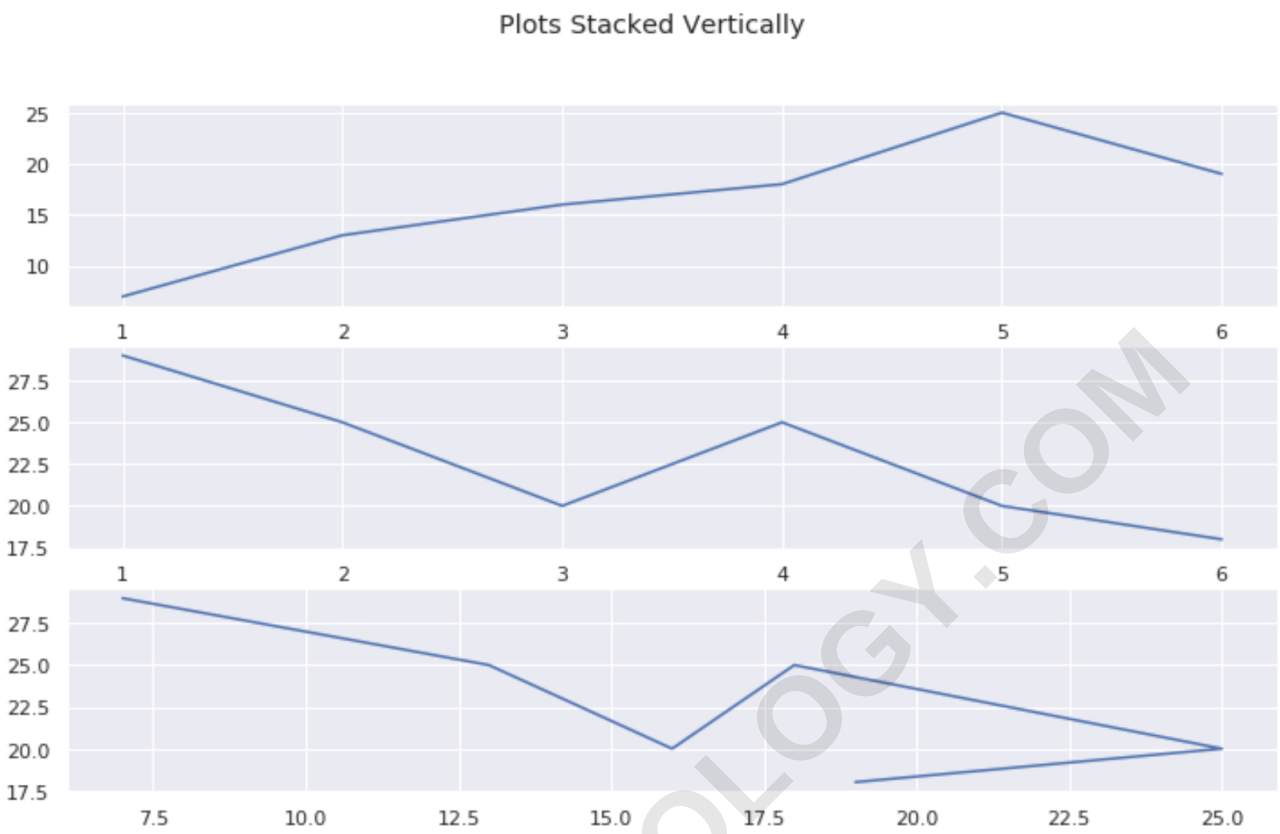
Matplotlib plots, stacked vertically:

```
#create some data
var1 =
var2 =
var3 =

#define grid of plots
fig, axs = plt.subplots(nrows=3, ncols=1)

#add title
fig.suptitle('Plots Stacked Vertically')

#add data to plots
axs.plot(var1, var2)
axs.plot(var1, var3)
axs.plot(var2, var3)
```



Example 2: Stack Plots Horizontally

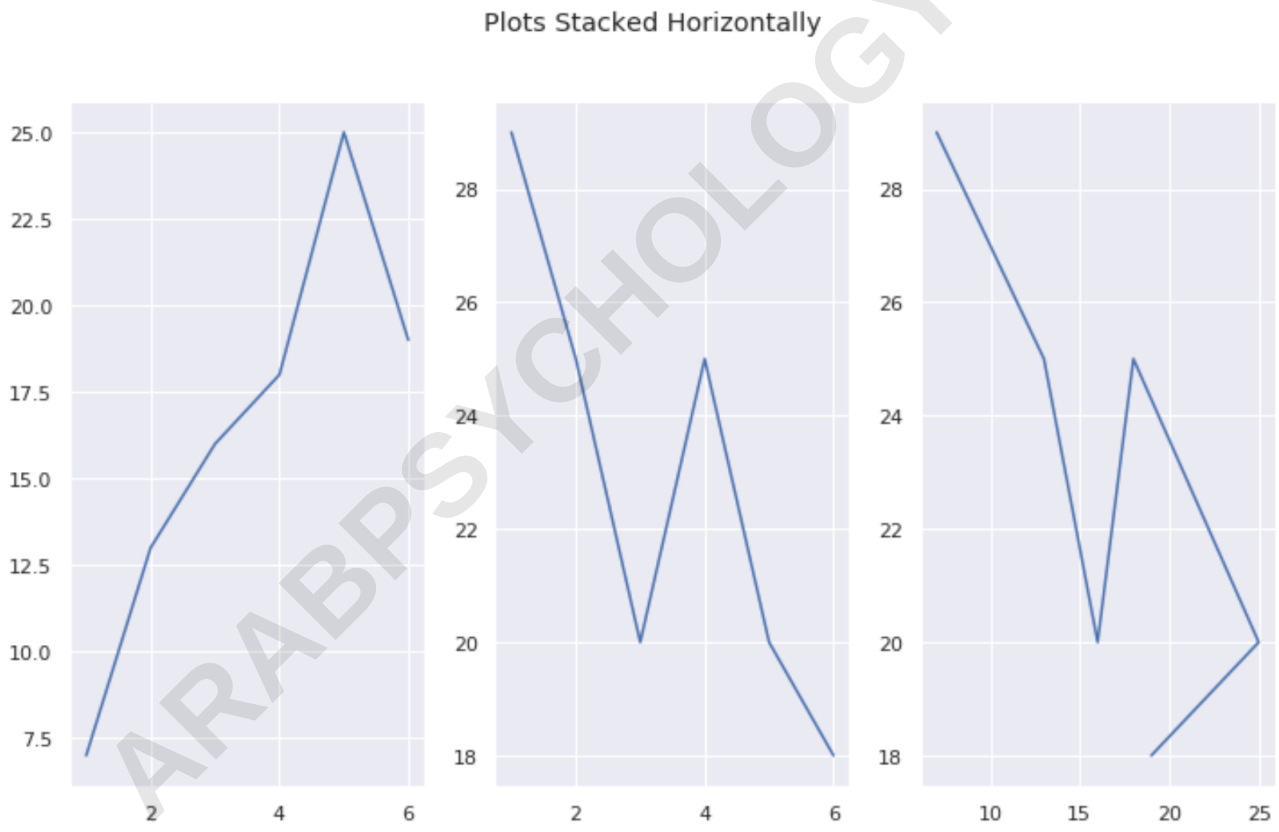
The following code shows how to create three Matplotlib plots, stacked horizontally:

```
#create some data
var1 =
var2 =
var3 =

#define grid of plots
fig, axs = plt.subplots(nrows=1, ncols=3)
```

```
#add title
fig.suptitle('Plots Stacked Horizontally')

#add data to plots
axs.plot(var1, var2)
axs.plot(var1, var3)
axs.plot(var2, var3)
```

Example 3: Create a Grid of Plots

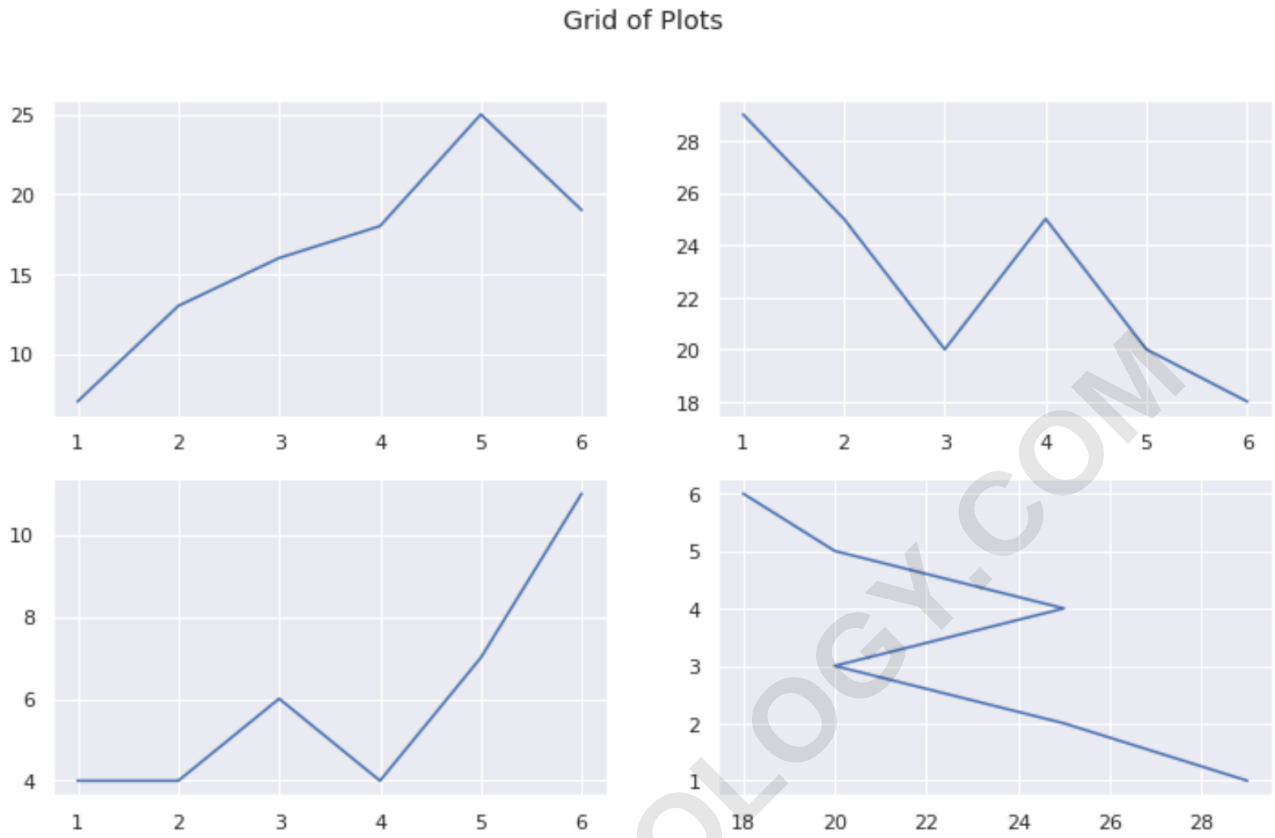
The following code shows how to create a grid of Matplotlib plots:

```
#create some data
var1 =
var2 =
var3 =
var4 =

#define grid of plots
fig, axs = plt.subplots(nrows=2, ncols=2)

#add title
fig.suptitle('Grid of Plots')

#add data to plots
axs.plot(var1, var2)
axs.plot(var1, var3)
axs.plot(var1, var4)
axs.plot(var3, var1)
```



Example 4: Share Axes Between Plots

You can use the sharex and sharey arguments to ensure that multiple plots use the same x-axis:

#create some data
var1 =
var2 =
var3 =
var4 =

#define grid of plots

```
fig, axs = plt.subplots(nrows=2, ncols=2, sharex=True, sharey=True)#add title
fig.suptitle('Grid of Plots with Same Axes')

#add data to plots
axs.plot(var1, var2)
axs.plot(var1, var3)
axs.plot(var1, var4)
axs.plot(var3, var1)
```