Can I use ggplot styles in matplotlib plots?

Authored by

stats writer

June 25, 2024

"Can I use ggplot styles in matplotlib plots?" is a commonly asked question among data analysts and visualization enthusiasts. The answer is yes, it is possible to use ggplot styles in matplotlib plots. ggplot is a popular plotting library in the R programming language, known for its elegant and visually appealing plots. However, with the introduction of the "style" feature in the latest versions of matplotlib, users can now replicate the same look and feel of ggplot plots in their matplotlib plots. This allows for more flexibility and customization options for users who are familiar with ggplot but want to use matplotlib for their projects. With a simple code adjustment, users can easily incorporate ggplot styles and create stunning visualizations in their matplotlib plots.

Use ggplot Styles in Matplotlib Plots

One of the most popular data visualization packages in the R programming language is ggplot2.

To apply ggplot2 styling to a plot created in Matplotlib, you can use the following syntax:

import matplotlib.pyplotas plt

plt.style.use('ggplot')

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

Example: Using ggplot Styles in Matplotlib Plots

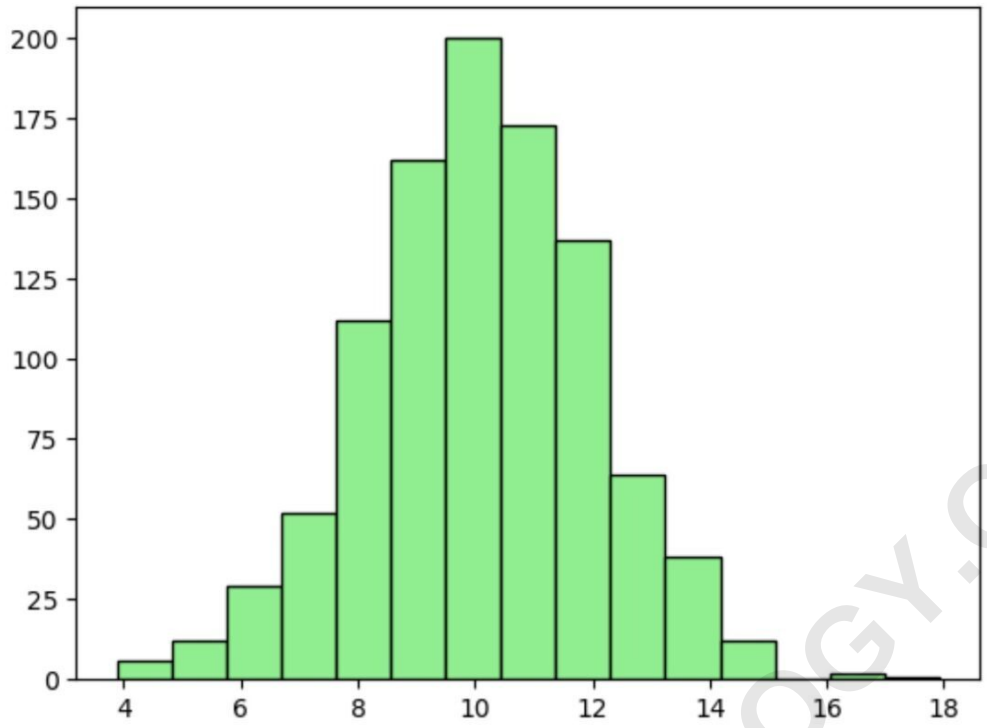
Suppose we have a NumPy array with 1,000 values:

import numpy as np

```
#make this example reproducible.
np.random.seed(1)

#create numpy array with 1000 values that follow normal dist with mean=10 and sd=2
data = np.random.normal(size=1000, loc=10, scale=2)

#view first five values
data

array()
```

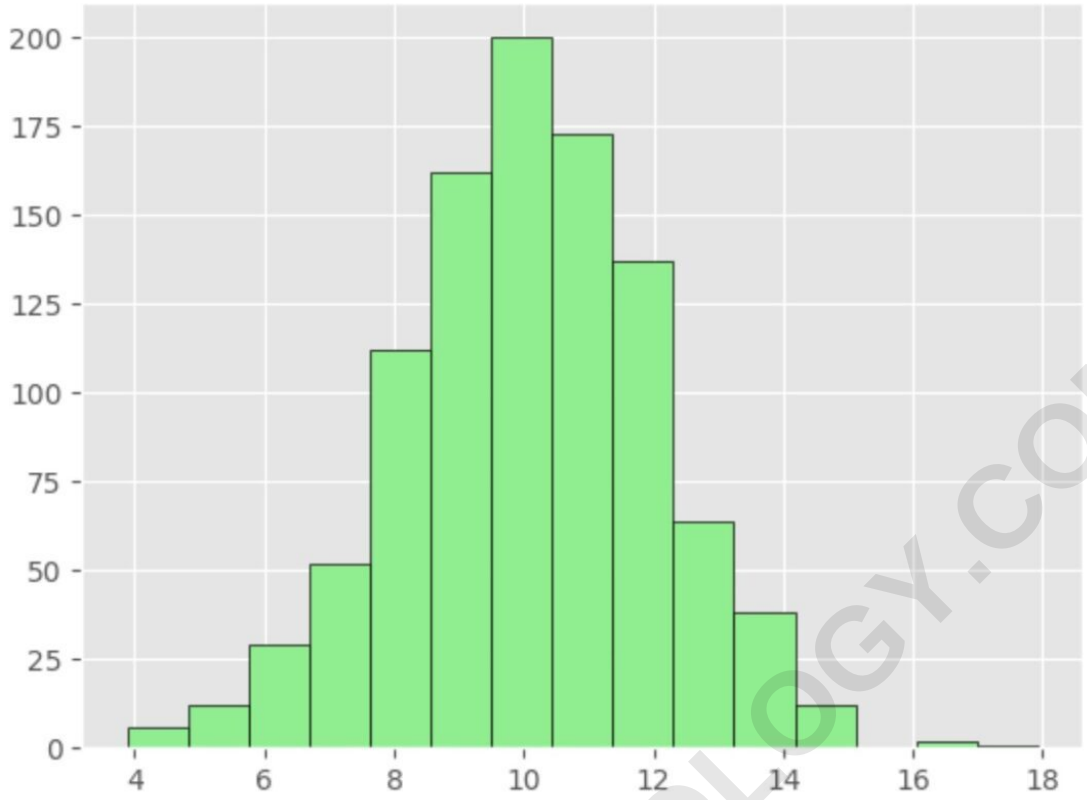
We can use the following code to create a histogram in Matplotlib to visualize the distribution of values in the NumPy array:

```
import matplotlib.pyplotas plt

#create histogram
plt.hist(data, color='lightgreen', ec='black', bins=15)
```



To apply ggplot2 styling to this histogram, we can use plt.syle.use('ggplot') as follows:

```
import matplotlib.pyplotas plt

#specify ggplot2 style
plt.style.use('ggplot')

#create histogram with ggplot2 style
plt.hist(data, color='lightgreen', ec='black', bins=15)
```



The histogram now has the style of a plot created in ggplot2.

Namely, this style adds a light grey background with white gridlines and uses slightly larger axis tick labels.

Note that we applied ggplot2 styling to a histogram, but the statement plt.style.use('ggplot') can be used to apply ggplot2 styling to any plot in Matplotlib.

Note: You can find more style sheets available to use in Matplotlib plots .

The following tutorials explain how to create other common charts in Python: