

How can I change the fonts in Matplotlib?

Authored by
stats writer

May 12, 2024

RECOMMENDED CITATION

stats writer (2024). *How can I change the fonts in Matplotlib?*. PSYCHOLOGICAL SCALES.
Retrieved from <https://scales.arabpsychology.com/?p=143832>

Matplotlib is a powerful data visualization library in Python that offers a wide range of customization options. One of the key features of Matplotlib is the ability to change the fonts used in the visualizations. This can be achieved by using the "font" parameter in the rcParams module or by specifying the font directly in the plot function. Additionally, users can also change the font size, style, and weight to further customize their visualizations. With these options, users can effectively enhance the overall appearance and readability of their Matplotlib plots.

Change Fonts in Matplotlib (With Examples)

You can use one of the following methods to change the font family in Matplotlib:

Method 1: Change Font for All Text

```
import matplotlib
```

```
matplotlib.rcParams = 'monospace'
```

Method 2: Change Font for Title & Axis Labels

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

```
mono_font = {'fontname':'monospace'}
```

```
serif_font = {'fontname':'serif'}
```

```
plt.title('Title of Plot',**mono_font)
```

```
plt.xlabel('X Label', **serif_font)
```

The following examples show how to use each method in practice.

Method 1: Change Font for All Text

The following code shows how to change the font family for all text in a Matplotlib plot:

```
import matplotlib
import matplotlib.pyplot as plt

#define font family to use for all text
matplotlib.rcParams = 'monospace'

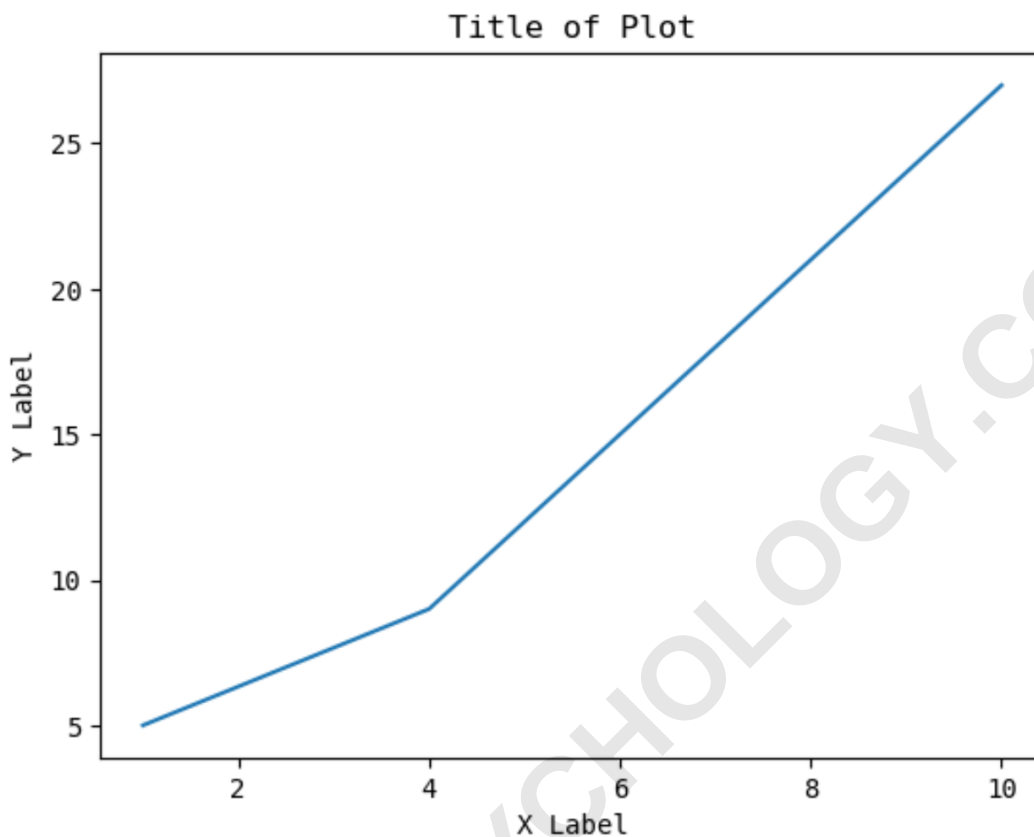
#define x and y
x =
y =

#create line plot
plt.plot(x, y)

#add title and axis labels
plt.title('Title of Plot')
plt.xlabel('X Label')
plt.ylabel('Y Label')

#display plot
```

plt.show()



Notice that the title and both axis labels have a 'monospace' font, since that is the font family we specified in the rcParams argument.

Method 2: Change Font for Title & Axis Labels

The following code shows how to specify a unique font family for both the title and the axis labels:

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

#define font families to use

mono_font = {'fontname':'monospace'}

serif_font = {'fontname':'serif'}

#define x and y

x =

y =

#create plot of x and y

plt.plot(x, y)

#specify title and axis labels with custom font families

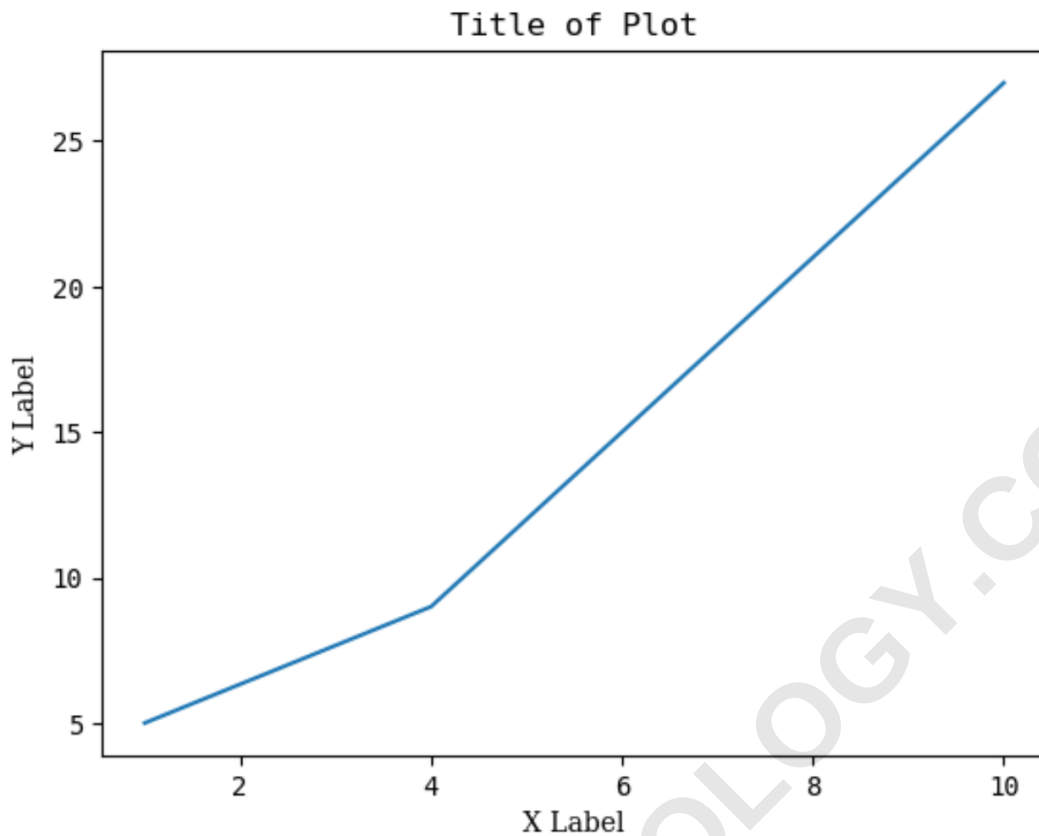
plt.title('Title of Plot', **mono_font)

plt.xlabel('X Label', **serif_font)

plt.ylabel('Y Label', **serif_font)

#display plot

plt.show()



Notice that the title uses a 'monospace' font family while the x-axis and y-axis labels use a 'serif' font family.

Note: You can find a complete list of available font families that you can use in Matplotlib .