

How can I plot a smooth curve using Matplotlib?

Authored by
stats writer

April 19, 2024

RECOMMENDED CITATION

stats writer (2024). *How can I plot a smooth curve using Matplotlib?*. PSYCHOLOGICAL SCALES. Retrieved from <https://scales.arabpsychology.com/?p=136913>

Matplotlib is a powerful library in Python that allows for the creation of high-quality visualizations. One of its key features is the ability to plot smooth curves. To plot a smooth curve using Matplotlib, you would first need to import the library and the necessary modules. Then, you can use the "plot" function to create a basic line graph. To make the curve smoother, you can use the "smooth" function and specify the degree of smoothing. Additionally, you can customize the curve by adjusting the color, line style, and marker style. With Matplotlib, creating a smooth, visually appealing curve is simple and efficient.

Plot a Smooth Curve in Matplotlib

Often you may want to plot a smooth curve in Matplotlib for a line chart. Fortunately this is easy to do with the help of the following SciPy functions:

scipy.interpolate.make_interp_spline()**scipy.interpolate.BSpline()**

This tutorial explains how to use these functions in practice.

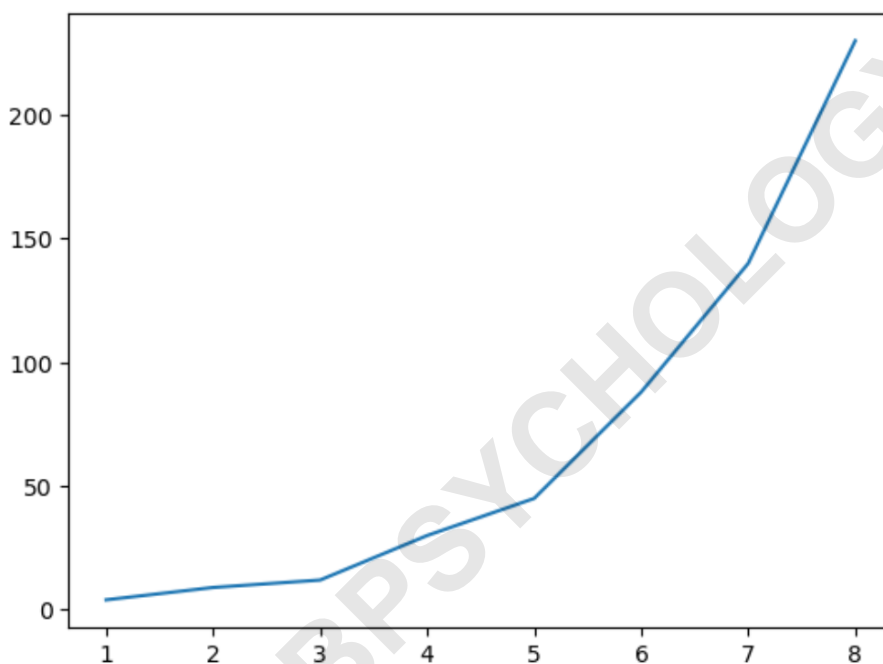
Example: Plotting a Smooth Curve in Matplotlib

The following code shows how to create a simple line chart for a dataset:

```
import numpy as np
import matplotlib.pyplot as plt

#create data
```

```
x = np.array()  
y = np.array()  
  
#create line chart  
plt.plot(x,y)  
plt.show()
```



Notice that the line chart isn't completely smooth since the underlying data doesn't follow a smooth line. We can use the following code to create a smooth curve for this dataset:

```
from scipy.interpolate import make_interp_spline,  
BSpline
```

```
#create data
```

```
x = np.array()
```

```
y = np.array()
```

```
#define x as 200 equally spaced values between the min  
and max of original x
```

```
xnew = np.linspace(x.min(), x.max(), 200)
```

```
#define spline
```

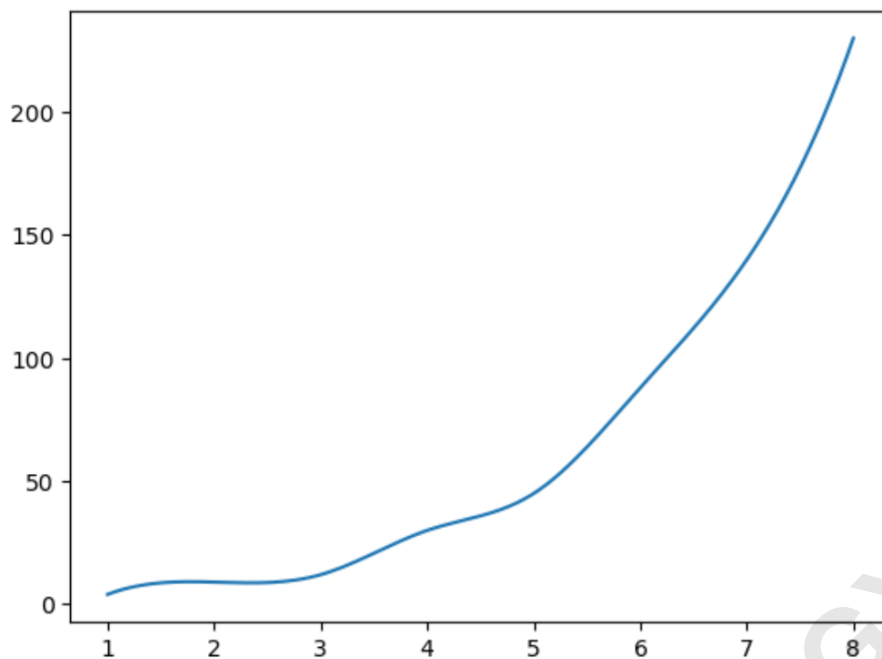
```
spl = make_interp_spline(x, y, k=3)
```

```
y_smooth = spl(xnew)
```

```
#create smooth line chart
```

```
plt.plot(xnew, y_smooth)
```

```
plt.show()
```



Note that the higher the degree you use for the `k` argument, the more "wiggly" the curve will be. For example, consider the following chart with `k=7`:

```
from scipy.interpolate import make_interp_spline,  
BSpline
```

```
#create data
```

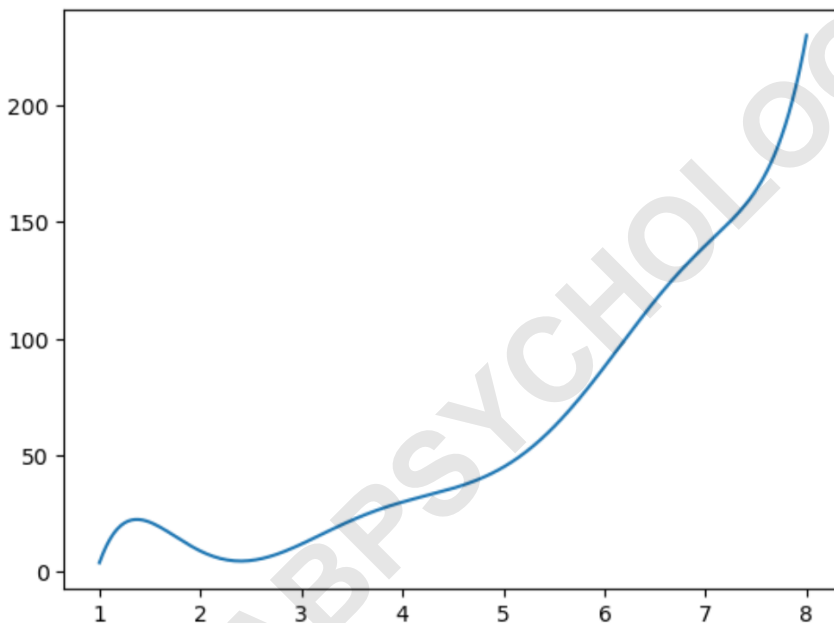
```
x = np.array()
```

```
y = np.array()
```

```
#define x as 200 equally spaced values between the min  
and max of original x
```

```
xnew = np.linspace(x.min(), x.max(), 200)
```

```
#define spline with degree k=7  
spl = make_interp_spline(x, y, k=7)  
y_smooth = spl(xnew)  
  
#create smooth line chart  
plt.plot(xnew, y_smooth)  
plt.show()
```



Depending on how curved you want the line to be, you can modify the value for k.

How to Show Gridlines on Matplotlib Plots

How to Remove Ticks from Matplotlib Plots

How to Create Matplotlib Plots with Log Scales

ARABPSYCHOLOGY.COM