

How can a trendline be added in Matplotlib?

Authored by
stats writer

June 29, 2024

RECOMMENDED CITATION

stats writer (2024). *How can a trendline be added in Matplotlib?*. PSYCHOLOGICAL SCALES. Retrieved from <https://scales.arabpsychology.com/?p=157788>

A trendline can be added in Matplotlib by using the "plot" function and specifying the data points to be plotted. Then, the "polyfit" function can be used to calculate the coefficients of the trendline. These coefficients can be used to create an equation for the trendline, which can be plotted using the "plot" function again. Additionally, the "grid" function can be used to add a grid to the plot, making it easier to visualize the trendline in relation to the data.

Add a Trendline in Matplotlib (With Example)

You can use the following basic syntax to add a trendline to a plot in Matplotlib:

```
#create scatterplot
```

```
plt.scatter(x, y)
```

```
#calculate equation for trendline
```

```
z = np.polyfit(x, y, 1)
```

```
p = np.poly1d(z)
```

```
#add trendline to plot
```

```
plt.plot(x, p(x))
```

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

Example 1: Create Linear Trendline in Matplotlib

The following code shows how to create a basic trendline for a scatterplot in Matplotlib:

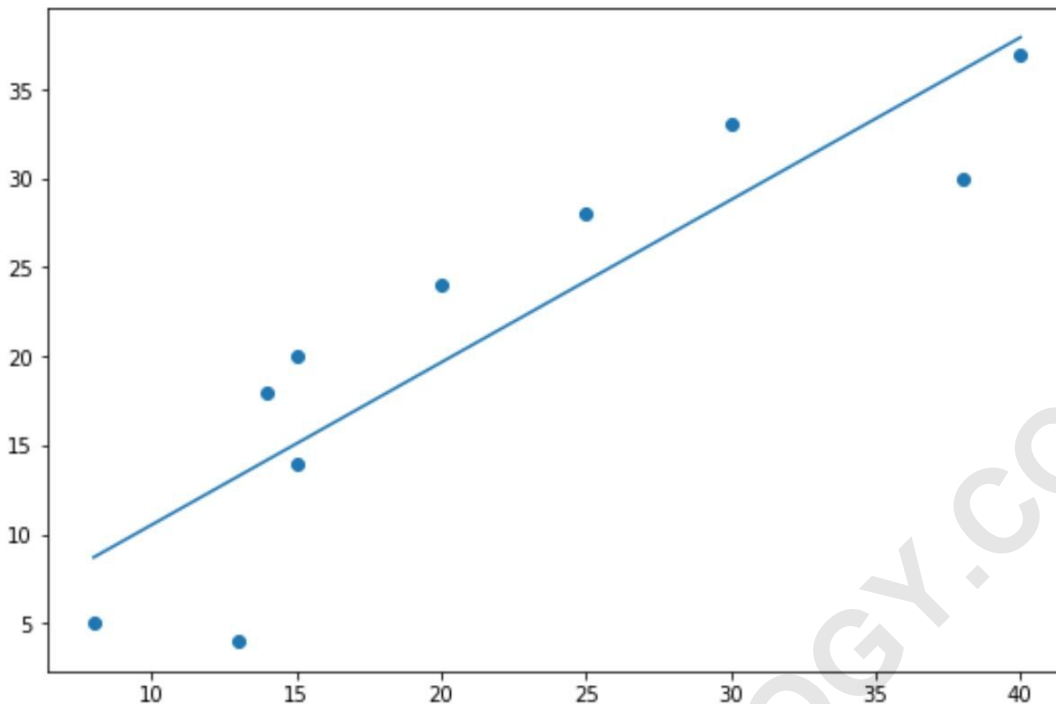
```
import numpy as np
import matplotlib.pyplot as plt

#define data
x = np.array()
y = np.array()

#create scatterplot
plt.scatter(x, y)

#calculate equation for trendline
z = np.polyfit(x, y, 1)
p = np.poly1d(z)

#add trendline to plot
plt.plot(x, p(x))
```

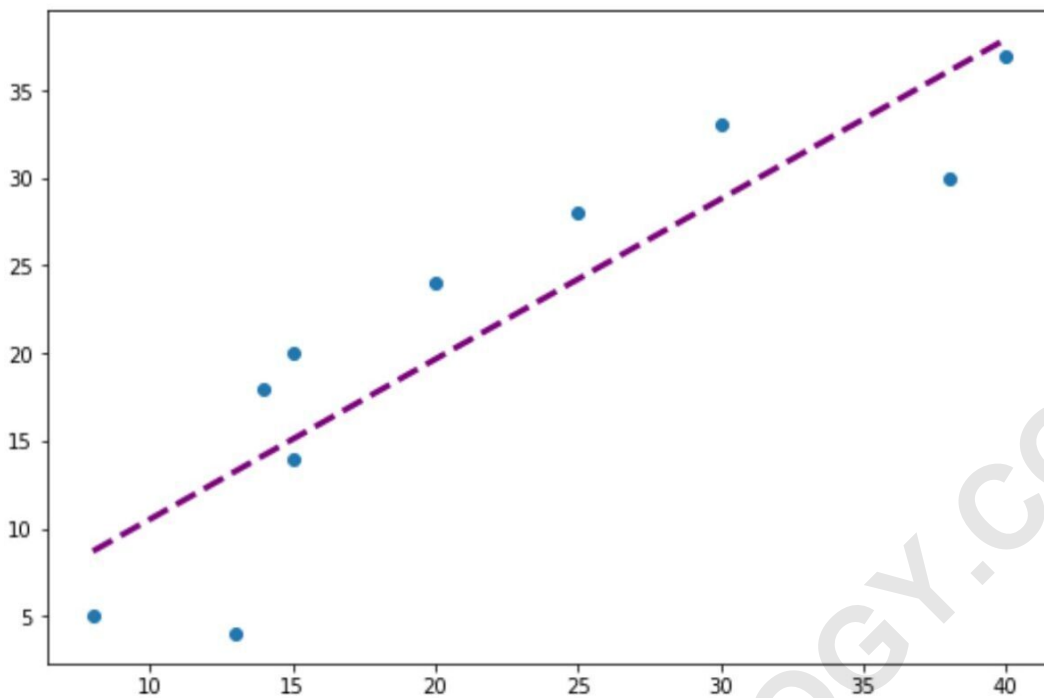


The blue dots represent the data points and the straight blue line represents the linear trendline.

Note that you can also use the color, linewidth, and linestyle arguments to modify the appearance of the trendline:

#add custom trendline to plot

```
plt.plot(x, p(x), color="purple", linewidth=3, linestyle="--")
```



Example 2: Create Polynomial Trendline in Matplotlib

To create a polynomial trendline, simply change the value in the `np.polyfit()` function.

For example, we could use a value of 2 to create a quadratic trendline:

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

```
#define data
```

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

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

```
#create scatterplot
```

```
plt.scatter(x, y)
```

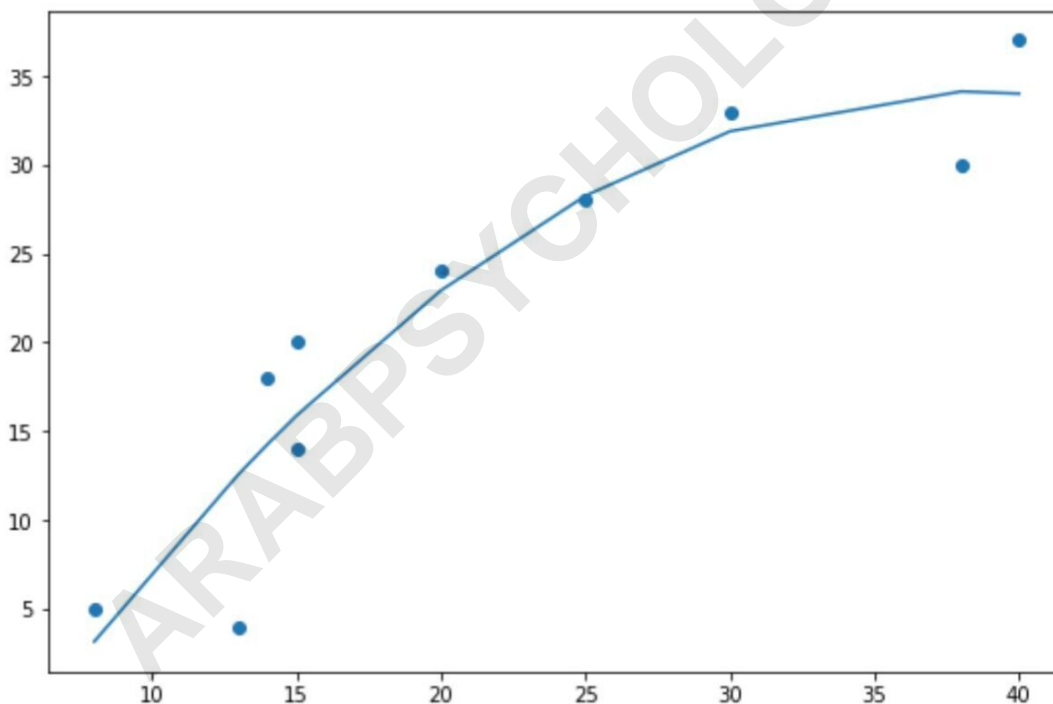
```
#calculate equation for quadratic trendline
```

```
z = np.polyfit(x, y, 2)
```

```
p = np.poly1d(z)
```

```
#add trendline to plot
```

```
plt.plot(x, p(x))
```



Notice that the trendline is now curved instead of straight.

Additional Resources

The following tutorials explain how to perform other common functions in Matplotlib:

ARABPSYCHOLOGY.COM