

How can I plot circles in Matplotlib?

Authored by
stats writer

April 22, 2024

RECOMMENDED CITATION

stats writer (2024). *How can I plot circles in Matplotlib?*. PSYCHOLOGICAL SCALES.
Retrieved from <https://scales.arabpsychology.com/?p=137891>

Matplotlib is a powerful and popular library used for data visualization in Python. One of its key features is the ability to plot circles, which can be useful in various applications such as creating scatter plots, representing data points on a map, or visualizing relationships between variables. To plot circles in Matplotlib, one can use the "circle" function and specify the center coordinates, radius, and other parameters for customization. This function can be easily incorporated into a larger plot or used individually to create standalone circles. With its user-friendly syntax and extensive customization options, Matplotlib offers a straightforward and efficient way to plot circles for data visualization.

Plot Circles in Matplotlib (With Examples)

You can quickly add circles to a plot in Matplotlib by using the function, which uses the following syntax:

```
matplotlib.patches.Circle(xy, radius=5)
```

where:

xy: The (x, y) coordinates for the circle
radius: The radius of the circle. Default is 5.

This tutorial shows several examples of how to use this function in practice:

Example 1: Create a Single Circle

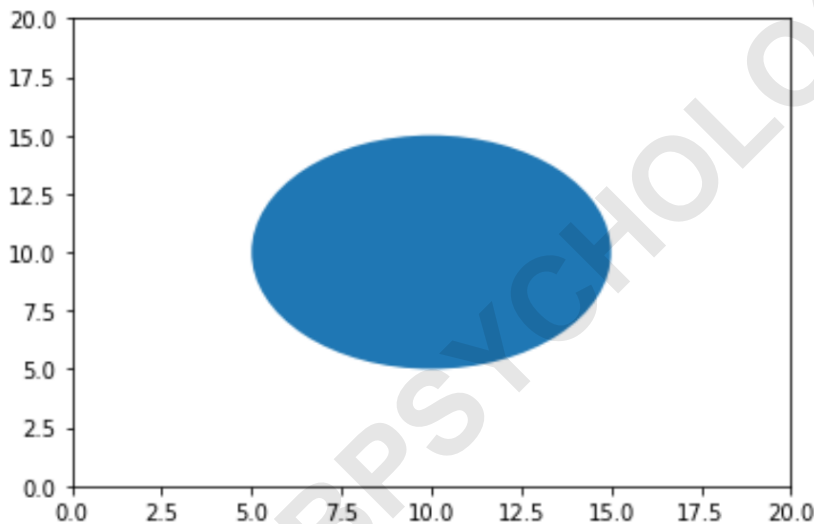
The following code shows how to create a single circle on a Matplotlib plot located at (x, y) coordinates (10,10):

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

```
#set axis limits of plot (x=0 to 20, y=0 to 20)  
plt.axis()
```

```
#create circle with (x, y) coordinates at (10, 10)  
c=plt.Circle((10, 10))
```

```
#add circle to plot (gca means "get current axis")  
plt.gca().add_artist(c)
```



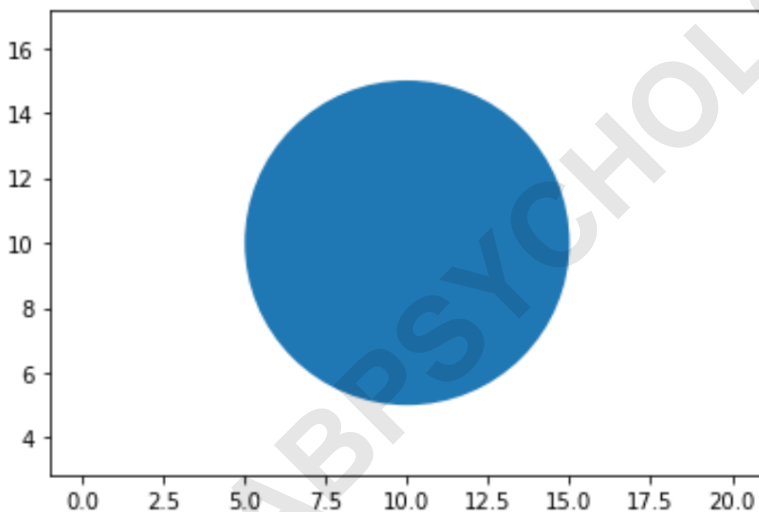
By default, one axis of a Matplotlib plot typically displays more pixels per data unit. To make a circle appear as a circle instead of an ellipse, you need to use the argument `plt.axis("equal")` as follows:

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

```
#set axis limits of plot (x=0 to 20, y=0 to 20)  
plt.axis()  
plt.axis("equal")
```

```
#create circle with (x, y) coordinates at (10, 10)  
c=plt.Circle((10, 10))
```

```
#add circle to plot (gca means "get current axis")  
plt.gca().add_artist(c)
```



Example 2: Create Multiple Circles

The following code shows how to create multiple circles on a Matplotlib plot:

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

```
#set axis limits of plot (x=0 to 20, y=0 to 20)
```

```
plt.axis()
```

```
plt.axis("equal")
```

```
#define circles
```

```
c1=plt.Circle((5, 5), radius=1)
```

```
c2=plt.Circle((10, 10), radius=2)
```

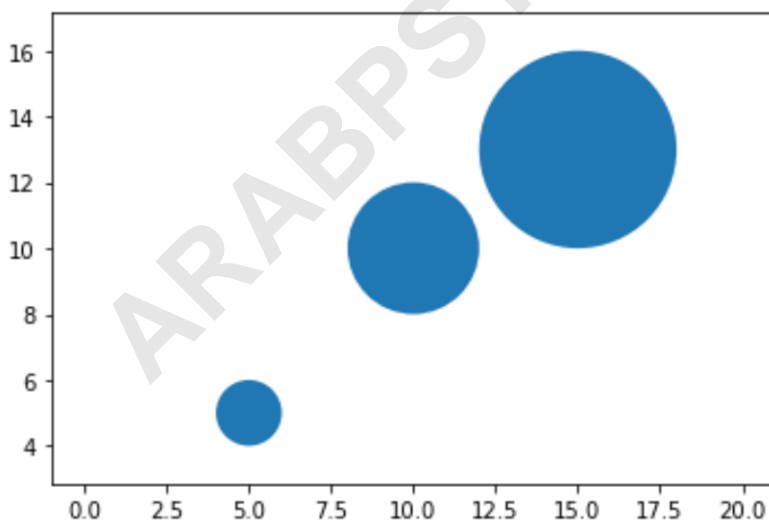
```
c3=plt.Circle((15, 13), radius=3)
```

```
#add circles to plot
```

```
plt.gca().add_artist(c1)
```

```
plt.gca().add_artist(c2)
```

```
plt.gca().add_artist(c3)
```



Example 3: Modify Circle Appearance

radius: Specify radius of circle **color:** Specify color of

circlealpha: Specify transparency of circle

The following code shows an example of how to use several of these arguments at once:

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

```
#set axis limits of plot (x=0 to 20, y=0 to 20)
```

```
plt.axis()
```

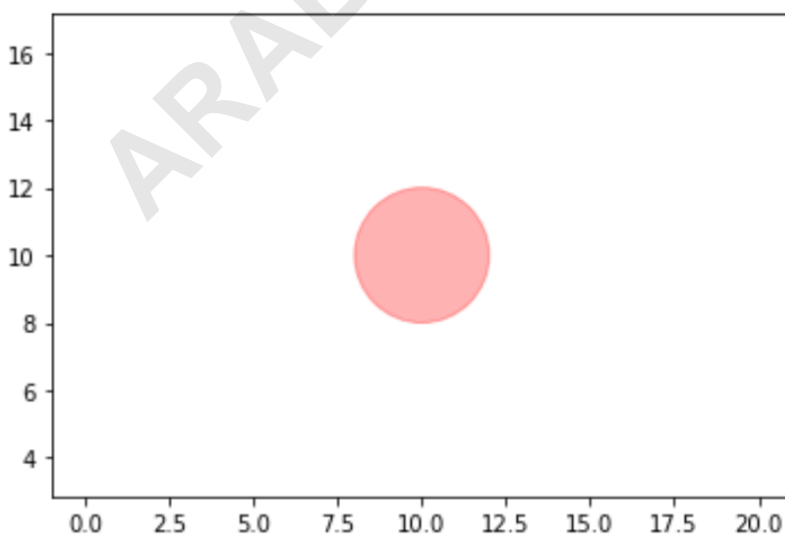
```
plt.axis("equal")
```

```
#create circle with (x, y) coordinates at (10, 10)
```

```
c=plt.Circle((10, 10), radius=2, color='red', alpha=.3)
```

```
#add circle to plot (gca means "get current axis")
```

```
plt.gca().add_artist(c)
```



Note that you can also use custom hex color codes to specify the color of circles.

ARABPSYCHOLOGY.COM