**Matplotlib**

### Matplotlib Scatter

Creating Scatter Plots

With Pyplot, you can use the scatter() function to draw a scatter plot.

The scatter() function plots one dot for each observation. It needs two arrays of the same length, one for the values of the x-axis, and one for values on the y-axis:

Example

A simple scatter plot:

import matplotlib.pyplot as plt  
import numpy as np  
  
x = np.array([5,7,8,7,2,17,2,9,4,11,12,9,6])  
y = np.array([99,86,87,88,111,86,103,87,94,78,77,85,86])  
  
plt.scatter(x, y)  
plt.show()

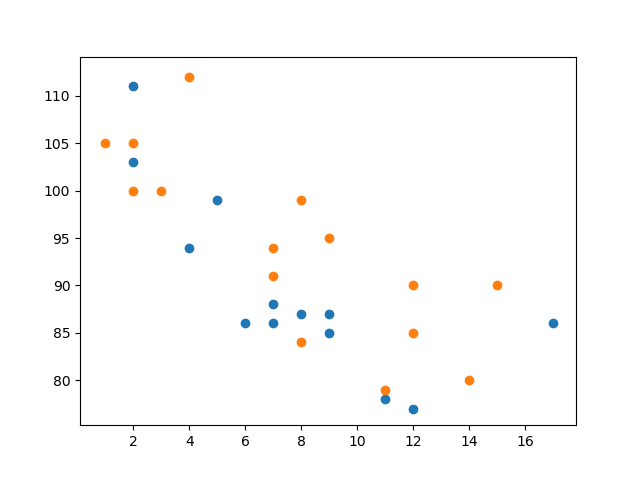


## **Compare Plots**

### Example

Draw two plots on the same figure:

import matplotlib.pyplot as plt  
import numpy as np  
  
#day one, the age and speed of 13 cars:  
x = np.array([5,7,8,7,2,17,2,9,4,11,12,9,6])  
y = np.array([99,86,87,88,111,86,103,87,94,78,77,85,86])  
plt.scatter(x, y)  
  
#day two, the age and speed of 15 cars:  
x = np.array([2,2,8,1,15,8,12,9,7,3,11,4,7,14,12])  
y = np.array([100,105,84,105,90,99,90,95,94,100,79,112,91,80,85])  
plt.scatter(x, y)  
  
plt.show()



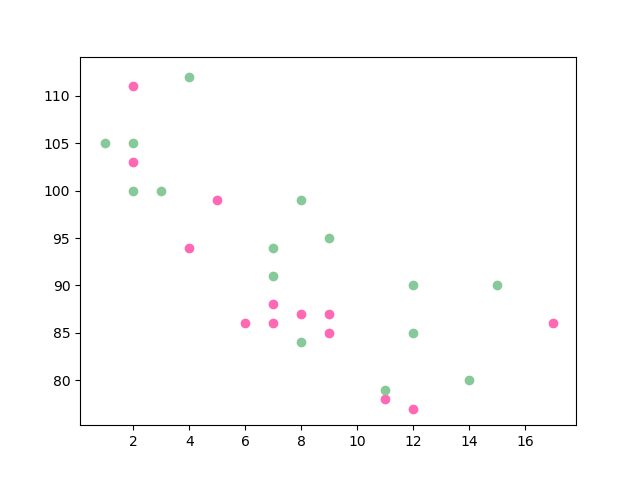
## **Colors**

You can set your own color for each scatter plot with the color or the c argument:

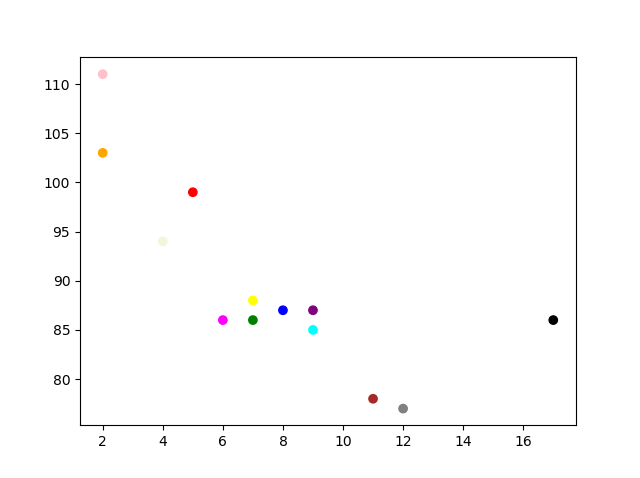
### Example

Set your own color of the markers:

x = np.array([5,7,8,7,2,17,2,9,4,11,12,9,6])  
y = np.array([99,86,87,88,111,86,103,87,94,78,77,85,86])  
plt.scatter(x, y, color = 'hotpink')  
  
x = np.array([2,2,8,1,15,8,12,9,7,3,11,4,7,14,12])  
y = np.array([100,105,84,105,90,99,90,95,94,100,79,112,91,80,85])  
plt.scatter(x, y, color = '#88c999')  
  
plt.show()



x = np.array([5,7,8,7,2,17,2,9,4,11,12,9,6])  
y = np.array([99,86,87,88,111,86,103,87,94,78,77,85,86])  
colors = np.array(["red","green","blue","yellow","pink","black","orange","purple","beige","brown","gray","cyan","magenta"])  
  
plt.scatter(x, y, c=colors)  
  
plt.show()



## **ColorMap**

The Matplotlib module has a number of available colormaps.

A colormap is like a list of colors, where each color has a value that ranges from 0 to 100.

Here is an example of a colormap:



This colormap is called 'viridis' and as you can see it ranges from 0, which is a purple color, and up to 100, which is a yellow color.

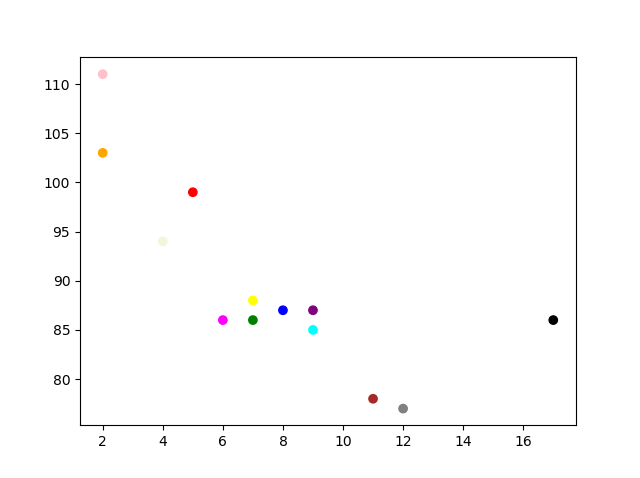
**How to Use the ColorMap**

You can specify the colormap with the keyword argument cmap with the value of the colormap, in this case 'viridis' which is one of the built-in colormaps available in Matplotlib.

In addition you have to create an array with values (from 0 to 100), one value for each of the point in the scatter plot:

### Example

### Create a color array, and specify a colormap in the scatter plot: x = np.array([5,7,8,7,2,17,2,9,4,11,12,9,6]) y = np.array([99,86,87,88,111,86,103,87,94,78,77,85,86]) colors = np.array([0, 10, 20, 30, 40, 45, 50, 55, 60, 70, 80, 90, 100]) plt.scatter(x, y, c=colors, cmap='viridis') plt.show()

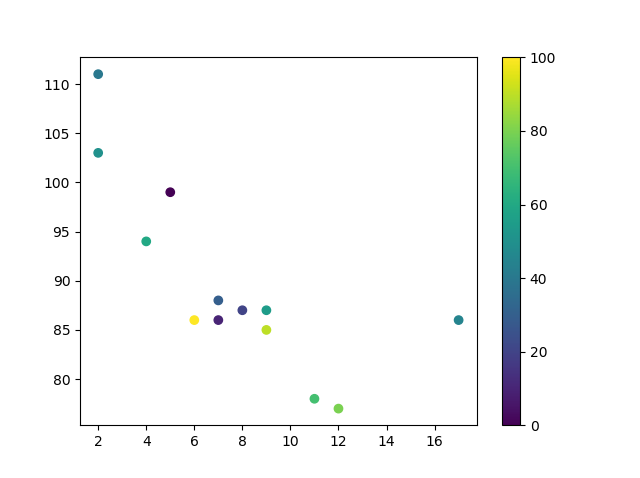


You can include the colormap in the drawing by including the plt.colorbar() statement:

### Example

Include the actual colormap:

import matplotlib.pyplot as plt  
import numpy as np  
  
x = np.array([5,7,8,7,2,17,2,9,4,11,12,9,6])  
y = np.array([99,86,87,88,111,86,103,87,94,78,77,85,86])  
colors = np.array([0, 10, 20, 30, 40, 45, 50, 55, 60, 70, 80, 90, 100])  
  
plt.scatter(x, y, c=colors, cmap='viridis')  
  
plt.colorbar()  
  
plt.show()



## **Size**

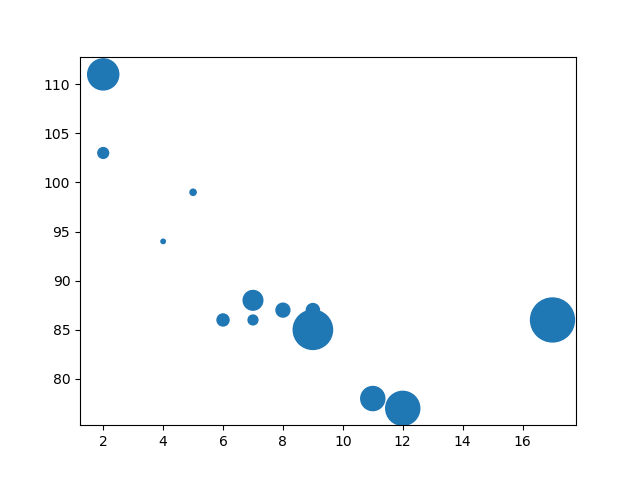
You can change the size of the dots with the s argument.

Just like colors, make sure the array for sizes has the same length as the arrays for the x- and y-axis:

### Example

Set your own size for the markers:

import matplotlib.pyplot as plt  
import numpy as np  
  
x = np.array([5,7,8,7,2,17,2,9,4,11,12,9,6])  
y = np.array([99,86,87,88,111,86,103,87,94,78,77,85,86])  
sizes = np.array([20,50,100,200,500,1000,60,90,10,300,600,800,75])  
  
plt.scatter(x, y, s=sizes)  
  
plt.show()



## **Alpha**

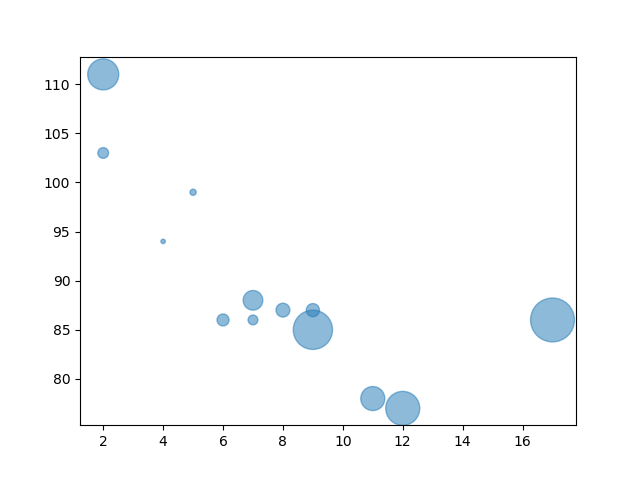
You can adjust the transparency of the dots with the alpha argument.

Just like colors, make sure the array for sizes has the same length as the arrays for the x- and y-axis:

### Example

Set your own size for the markers:

import matplotlib.pyplot as plt  
import numpy as np  
  
x = np.array([5,7,8,7,2,17,2,9,4,11,12,9,6])  
y = np.array([99,86,87,88,111,86,103,87,94,78,77,85,86])  
sizes = np.array([20,50,100,200,500,1000,60,90,10,300,600,800,75])  
  
plt.scatter(x, y, s=sizes, alpha=0.5)  
  
plt.show()



## **Combine Color Size and Alpha**

You can combine a colormap with different sizes on the dots. This is best visualized if the dots are transparent:

### Example

Create random arrays with 100 values for x-points, y-points, colors and sizes:

import matplotlib.pyplot as plt  
import numpy as np  
  
x = np.random.randint(100, size=(100))  
y = np.random.randint(100, size=(100))  
colors = np.random.randint(100, size=(100))  
sizes = 10 \* np.random.randint(100, size=(100))  
  
plt.scatter(x, y, c=colors, s=sizes, alpha=0.5, cmap='nipy\_spectral')  
  
plt.colorbar()  
  
plt.show()

