

scatter-plot

March 20, 2025

1 matplotlib Scatter-Plot

```
[4]: import matplotlib.pyplot as plt
import numpy as np
import pandas as pd
```

```
[2]: !pip install matplotlib
```

```
Collecting matplotlib
  Using cached matplotlib-3.9.2-cp312-cp312-win_amd64.whl.metadata (11 kB)
Collecting contourpy>=1.0.1 (from matplotlib)
  Using cached contourpy-1.3.0-cp312-cp312-win_amd64.whl.metadata (5.4 kB)
Collecting cycler>=0.10 (from matplotlib)
  Using cached cycler-0.12.1-py3-none-any.whl.metadata (3.8 kB)
Requirement already satisfied: fonttools>=4.22.0 in
c:\users\rahul\appdata\local\programs\python\python312\lib\site-packages (from
matplotlib) (4.54.1)
Requirement already satisfied: kiwisolver>=1.3.1 in
c:\users\rahul\appdata\local\programs\python\python312\lib\site-packages (from
matplotlib) (1.4.7)
Requirement already satisfied: numpy>=1.23 in
c:\users\rahul\appdata\local\programs\python\python312\lib\site-packages (from
matplotlib) (2.1.1)
Requirement already satisfied: packaging>=20.0 in
c:\users\rahul\appdata\local\programs\python\python312\lib\site-packages (from
matplotlib) (24.1)
Requirement already satisfied: pillow>=8 in
c:\users\rahul\appdata\local\programs\python\python312\lib\site-packages (from
matplotlib) (10.4.0)
Requirement already satisfied: pyparsing>=2.3.1 in
c:\users\rahul\appdata\local\programs\python\python312\lib\site-packages (from
matplotlib) (3.1.4)
Requirement already satisfied: python-dateutil>=2.7 in
c:\users\rahul\appdata\local\programs\python\python312\lib\site-packages (from
matplotlib) (2.9.0.post0)
Requirement already satisfied: six>=1.5 in
c:\users\rahul\appdata\local\programs\python\python312\lib\site-packages (from
python-dateutil>=2.7->matplotlib) (1.16.0)
```

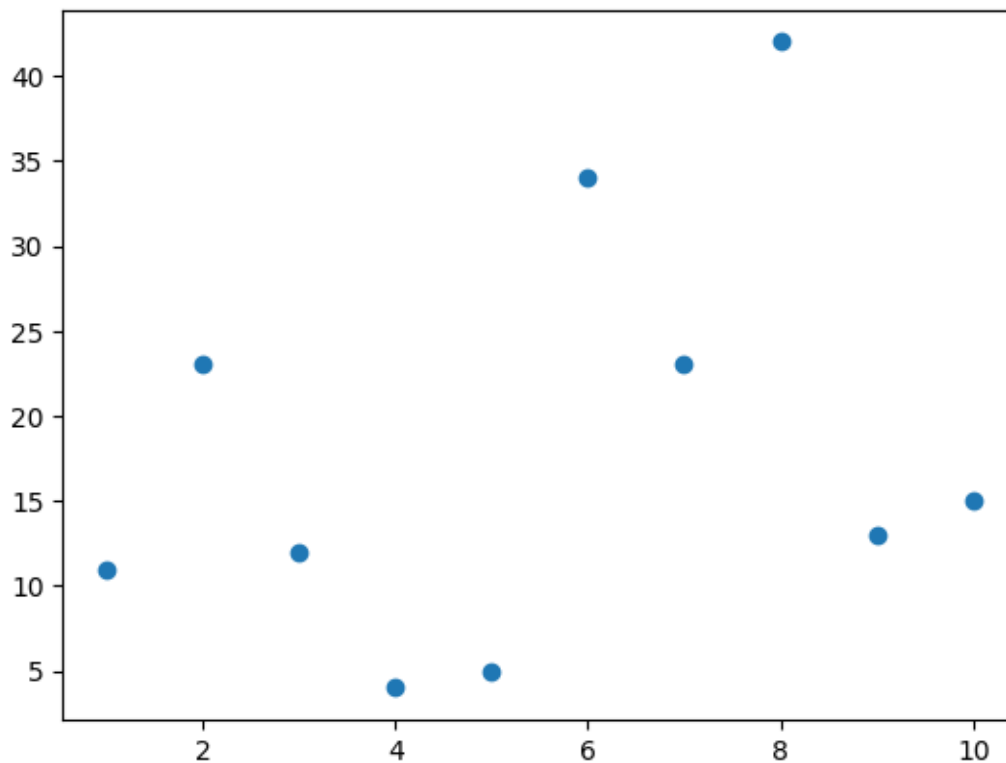
Using cached matplotlib-3.9.2-cp312-cp312-win_amd64.whl (7.8 MB)
Using cached contourpy-1.3.0-cp312-cp312-win_amd64.whl (218 kB)
Using cached cycler-0.12.1-py3-none-any.whl (8.3 kB)
Installing collected packages: cycler, contourpy, matplotlib
Successfully installed contourpy-1.3.0 cycler-0.12.1 matplotlib-3.9.2

```
[3]: print("my name is rahul Naidu")
```

my name is rahul Naidu

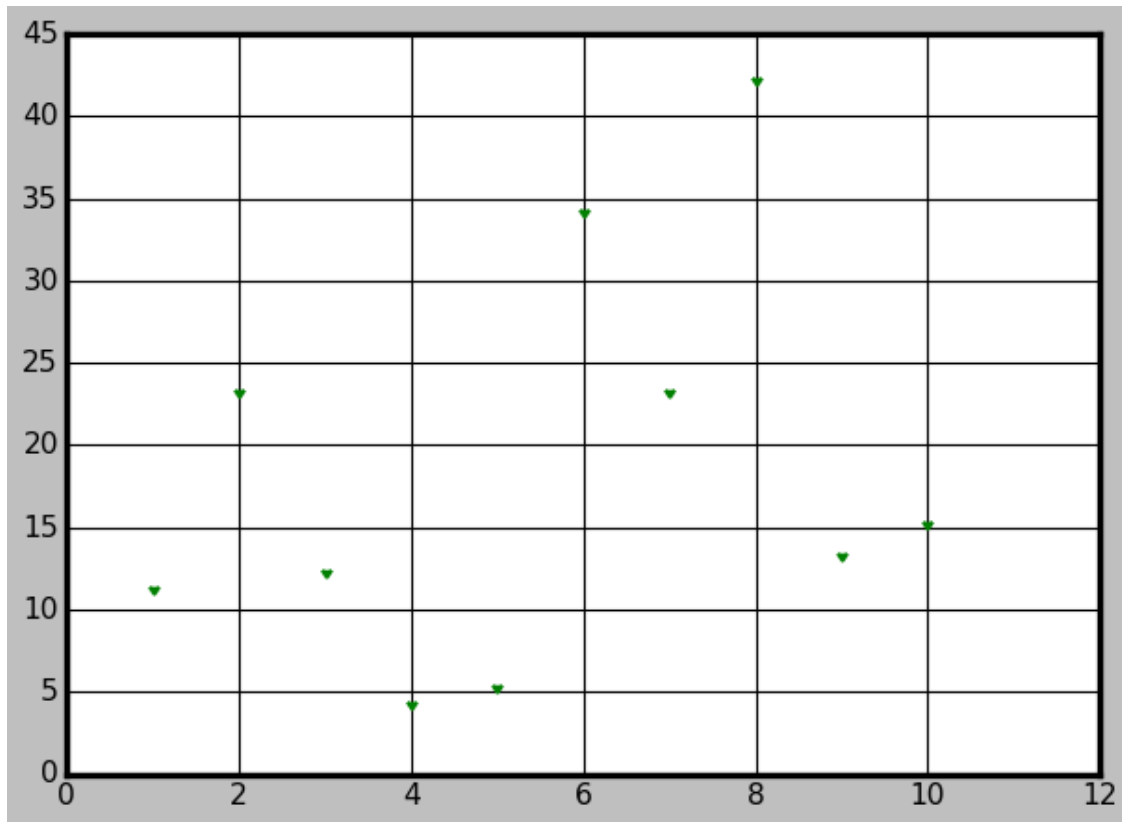
```
[7]: rollno = [1,2,3,4,5,6,7,8,9,10]  
marks = [11,23,12,4,5,34,23,42,13,15]
```

```
[8]: plt.scatter(rollno,marks)  
plt.show()
```



- 2 The `plt.style.use()` function in `matplotlib` allows you to apply different predefined styles to your plots. These styles control various aspects like background color, grid, and font. Below are some common values you can use with `plt.style.use()`:
- 3 1. Built-in Styles:
- 4 You can use these predefined styles by passing their names as strings.
- 5 ‘default’: The default Matplotlib style.
- 6 ‘classic’: The original Matplotlib style from earlier versions.
- 7 ‘ggplot’: Mimics the style of the popular R package `ggplot2`.
- 8 ‘fivethirtyeight’: A style inspired by the plots on the FiveThirtyEight website.
- 9 ‘bmh’: A simple style with a grey background, often used for business presentations.
- 10 ‘dark_background’: A style with a black background and bright lines.
- 11 ‘Solarize_Light2’: A style with a bright background and contrasting grid.
- 12 ‘grayscale’: A monochromatic, grayscale style.

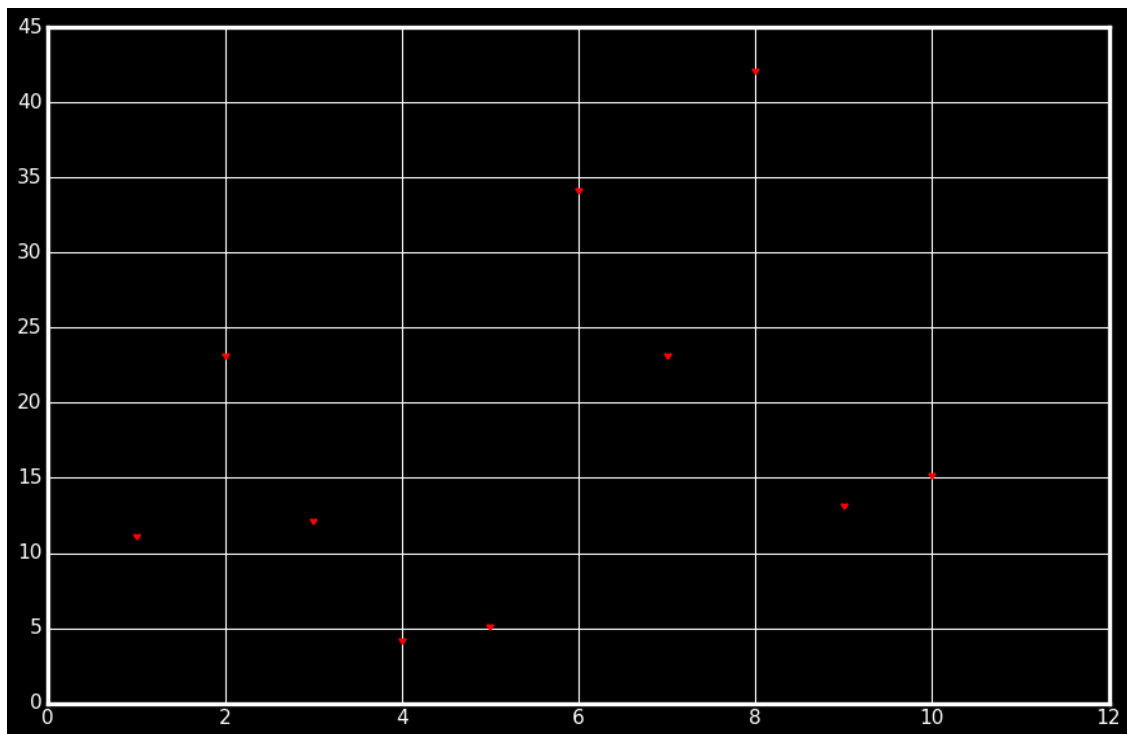
```
[27]: plt.style.use('grayscale')
      plt.scatter(rollno,marks,color='green',marker=7)
      plt.show()
```



‘.’: Point marker ‘,’: Pixel marker ‘o’: Circle marker ‘v’: Triangle down ‘^’: Triangle up ‘<’: Triangle left ‘>’: Triangle right ‘1’: Tri-down (triangular marker with only one point down) ‘2’: Tri-up (triangular marker with only one point up) ‘3’: Tri-left (triangular marker with only one point left) ‘4’: Tri-right (triangular marker with only one point right) ‘s’: Square marker ‘p’: Pentagon marker ‘*’: Star marker ‘h’: Hexagon1 (hexagon marker with horizontally flat top and bottom) ‘H’: Hexagon2 (hexagon marker with pointy top and bottom) ‘+’: Plus marker ‘x’: X marker ‘D’: Diamond marker ‘d’: Thin diamond marker ‘|’: Vertical line marker ‘_’: Horizontal line marker 2. Unfilled Markers: These markers are unfilled and are mostly the same as their filled counterparts, but without the fill:

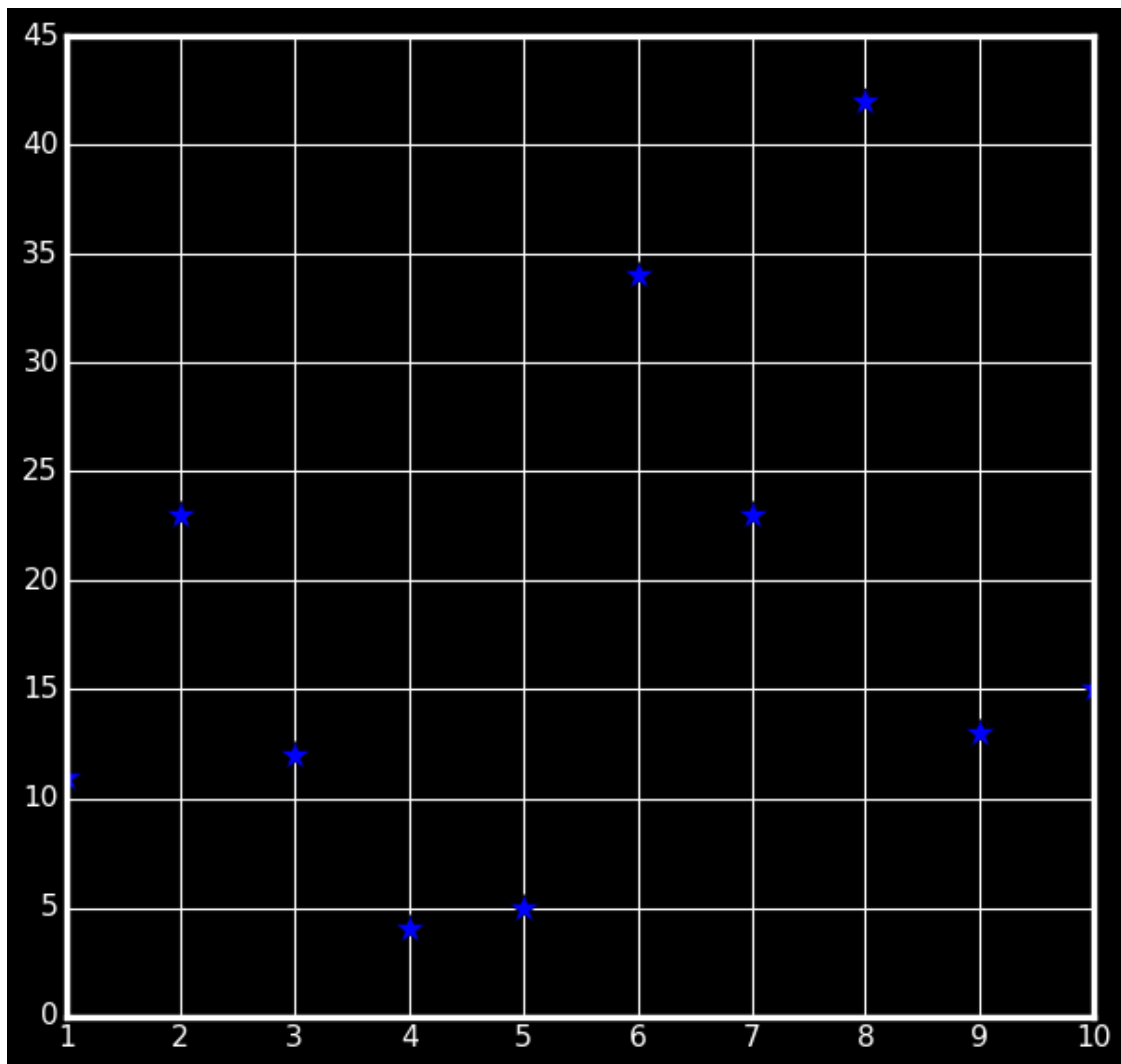
0: Tick left 1: Tick right 2: Tick up 3: Tick down 4: Caret left 5: Caret right 6: Caret up 7: Caret down

```
[42]: plt.figure(figsize = (12,8))
plt.scatter(rollno,marks,color='red',marker=7)
plt.show()
```



```
[34]: plt.figure(figsize = (8,8))  
plt.plot(rollno,marks , 'b*', markersize = 15)
```

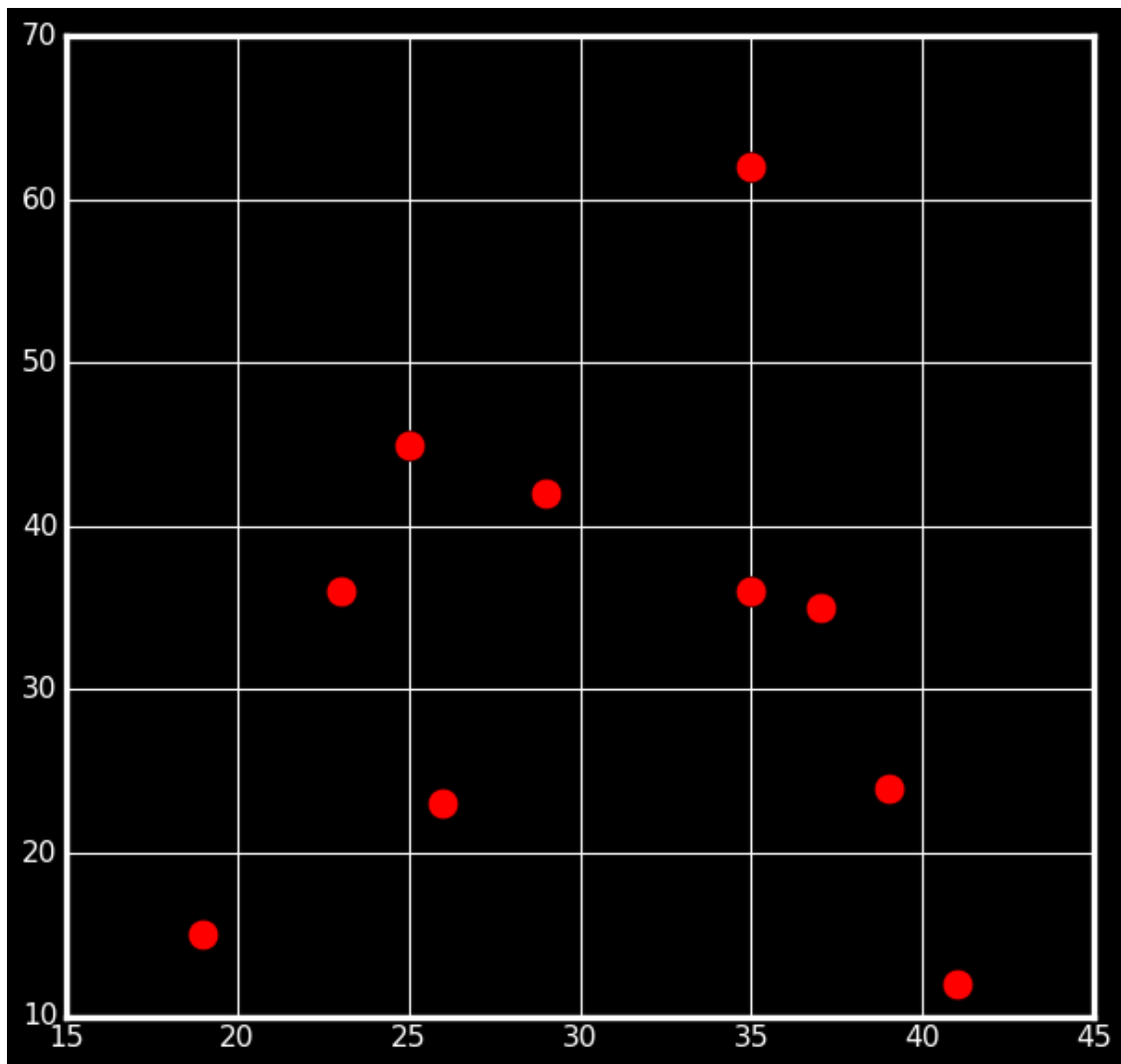
```
[34]: [<matplotlib.lines.Line2D at 0x2b3be705ee0>]
```



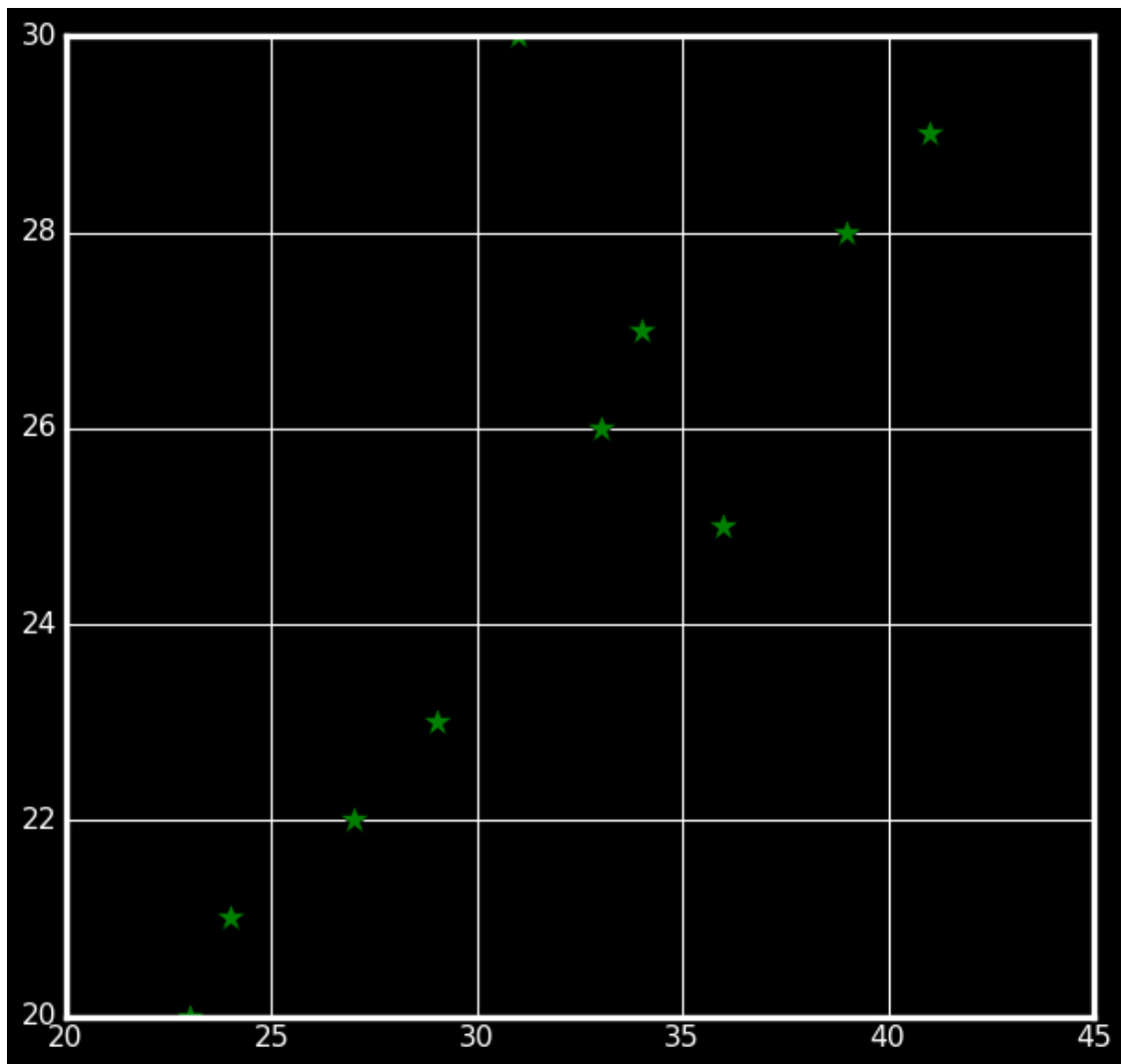
13 #Multiple plots on Same Figure

```
[35]: temp_pune = [25,35,26,19,35,29,41,37,23,39]
      humid_pune = [45,62,23,15,36,42,12,35,36,24]
      temp_bang = [23,24,29,27,34,36,33,31,39,41]
      humid_bang = [20,21,23,22,27,25,26,30,28,29]
```

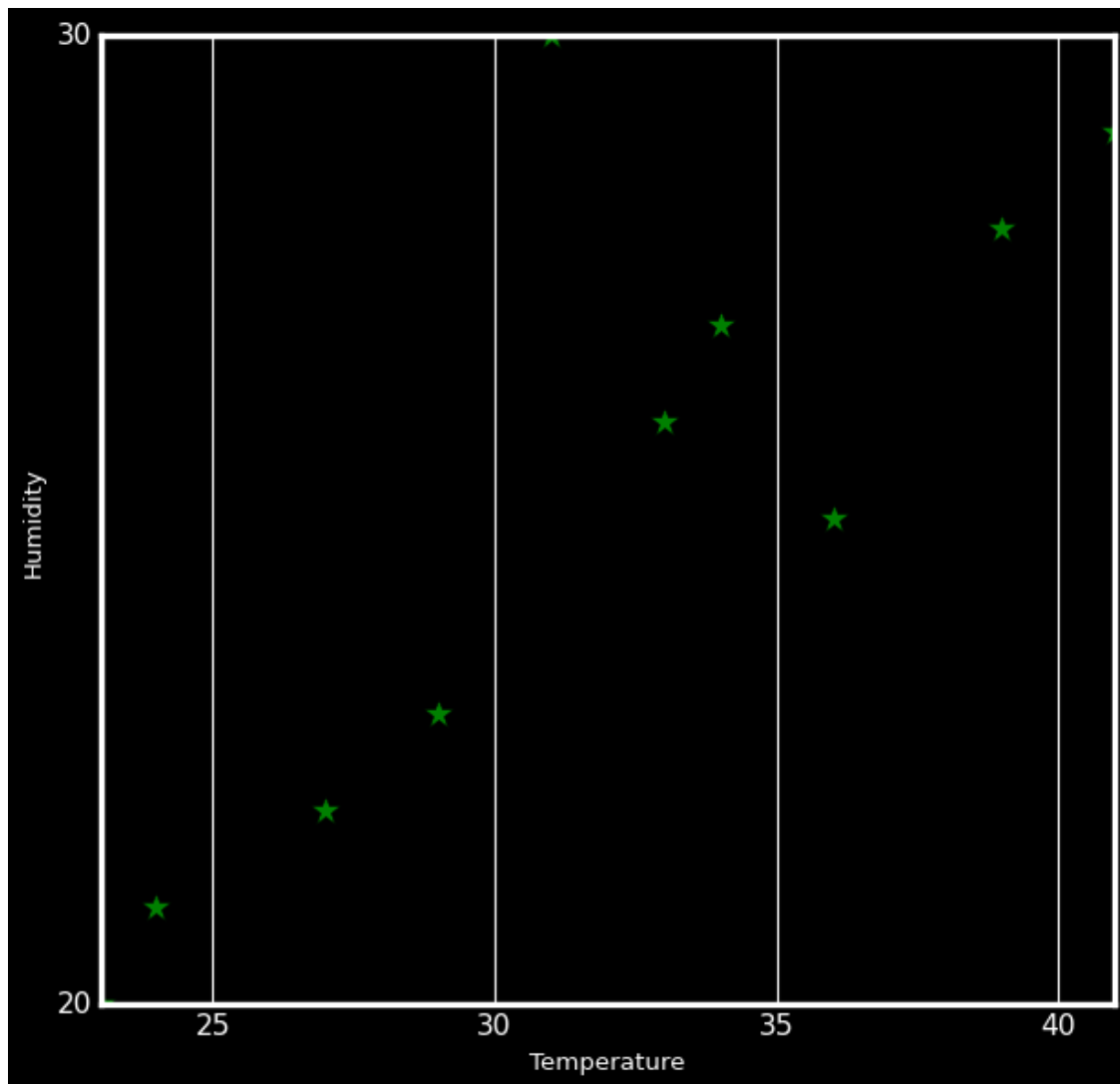
```
[37]: plt.figure(figsize=(8,8))
      plt.plot(temp_pune,humid_pune,'ro',markersize=15)
      plt.show()
```



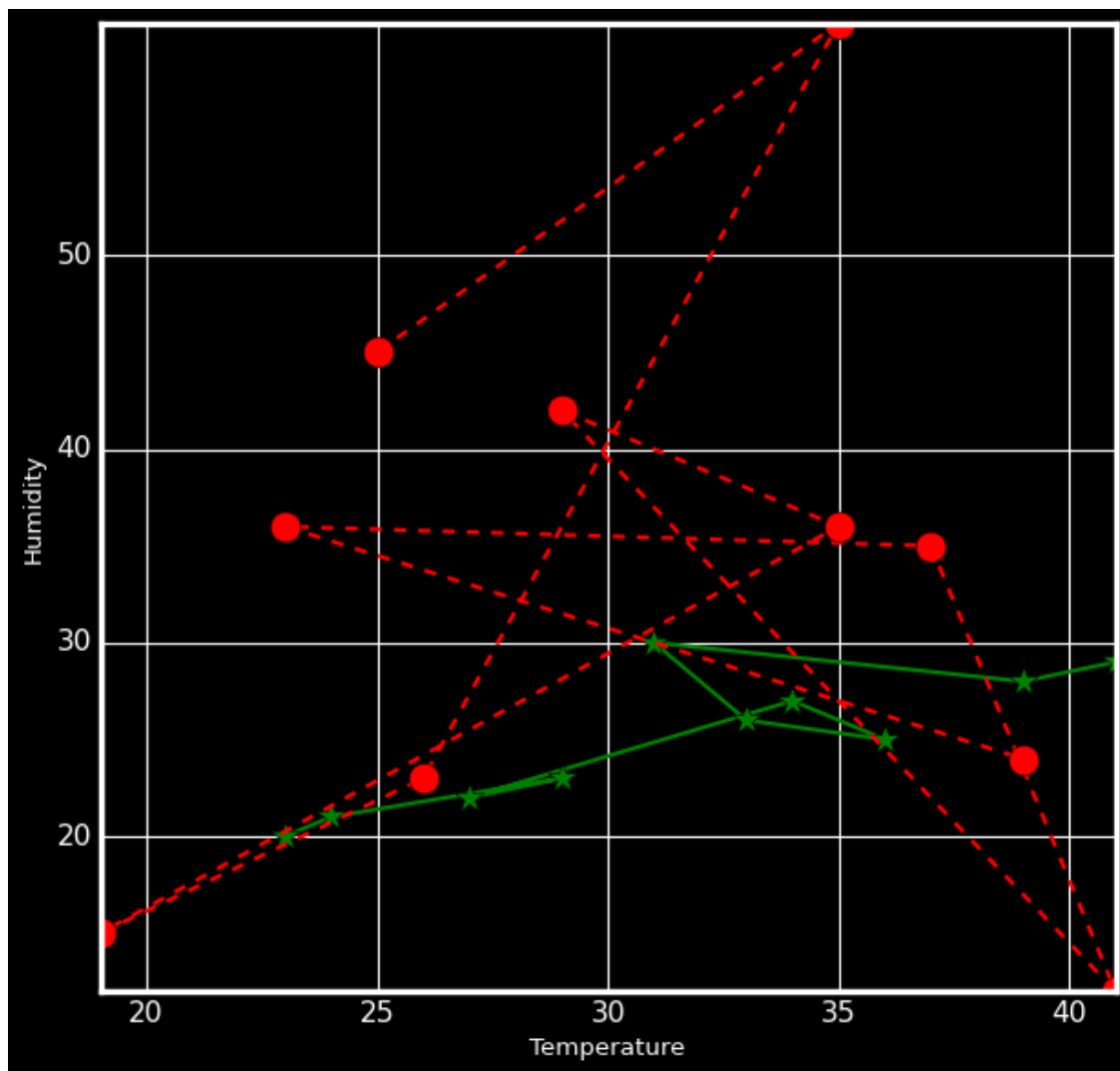
```
[38]: plt.figure(figsize=(8,8))  
plt.plot(temp_bang,humid_bang,'g*',markersize=15)  
plt.show()
```



```
[40]: plt.figure(figsize = (8,8))
plt.xticks(np.arange(0,60,5))
plt.yticks(np.arange(10,60,10))
plt.plot(temp_bang,humid_bang,'g*',markersize=15)
plt.xlabel('Temperature')
plt.ylabel('Humidity')
plt.show()
```

```
[48]: plt.figure(figsize = (8,8))
plt.xticks(np.arange(0,60,5))
plt.yticks(np.arange(10,60,10))
plt.plot(temp_bang,humid_bang,'g*',markersize=15,linestyle = '-')
plt.plot(temp_pune,humid_pune,'ro',markersize=15,linestyle="--")
plt.xlabel('Temperature')
plt.ylabel('Humidity')
plt.show()
```



[]: