



Introduction To Data Analysis

# 데이터 분석 입문

## Lecture 18. matplotlib의 scatter 그래프

인공지능소프트웨어학과 강환수 교수

## 학습개요

- ✓ 산점도를 그리는 scatter() 함수
- ✓ 산점도의 색상과 크기 옵션
- ✓ 산점도와 색상 활용 colorbar



## 학습목표

- ✓ 산점도를 이해하고 scatter() 함수로 산점도를 그릴 수 있다.
- ✓ 색상과 크기 옵션을 활용해 산점도 그래프를 그릴 수 있다.
- ✓ 산점도와 색상 활용 colorbar를 활용해 크기나 값을 색상으로 구분할 수 있다.

LESSON 01

# scatter() 함수의 산점도



## 산점도(Scatter Plot) 개요

산점도는 가로축과 세로축을 기준으로 두 요소가 서로 어떤 관계를 맺고 있는지를 파악하기 쉽게 나타낸 그래프

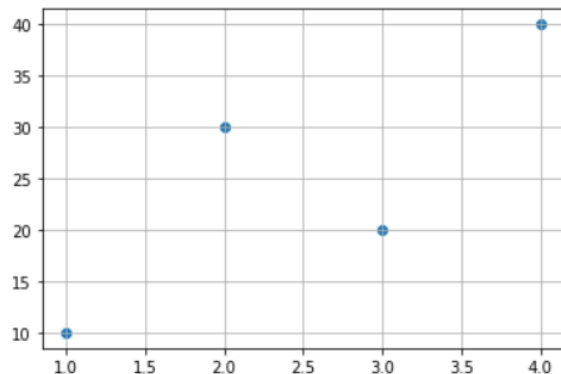
예를 들어,

- ✓ 가로축을 제주도 지역의 남성 인구수
- ✓ 세로축을 제주도 지역의 여성 인구수

## 산점도(Scatter Plot)

## plt.scatter(x, y)

```
import matplotlib.pyplot as plt  
  
plt.scatter([1, 2, 3, 4], [10, 30, 20, 40])  
plt.grid(True)  
plt.show()
```

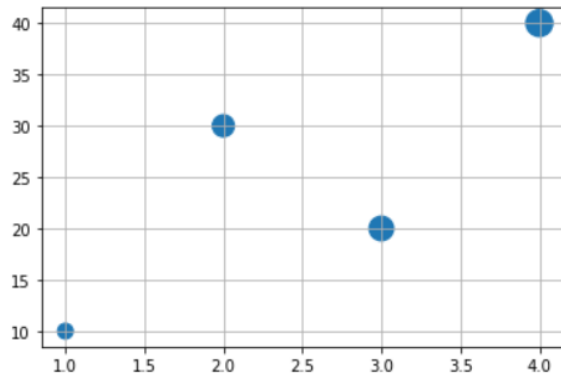


- (1, 10), (2, 30), (3, 20), (4, 40)  
좌표에 점이 찍히게 됩니다.

## 산점도(Scatter Plot)로 버블 차트(Bubble Chart) 표현

## 점(Point)의 크기를 지정: 키워드 인자 s

```
import matplotlib.pyplot as plt  
  
plt.scatter([1, 2, 3, 4], [10, 30, 20, 40], s=[100, 200, 250, 300])  
plt.grid(True)  
plt.show()
```



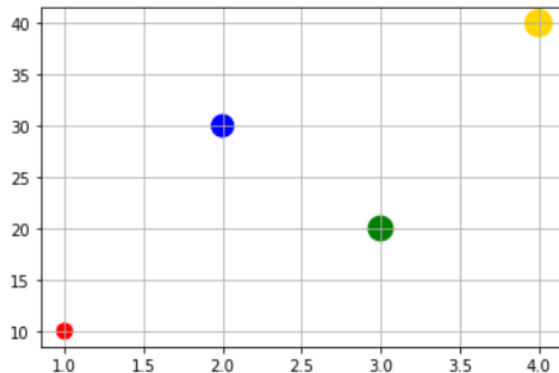
- s 속성은 size를 의미합니다.  
버블의 원하는 크기를 s  
속성에 지정해줍니다.

## 산점도(Scatter Plot)의 색상

## 버블의 색상 지정: c

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

```
plt.scatter([1, 2, 3, 4], [10, 30, 20, 40], s=[100, 200, 250, 300], c=['red', 'blue', 'green', 'gold'])  
plt.grid(True)  
plt.show()
```

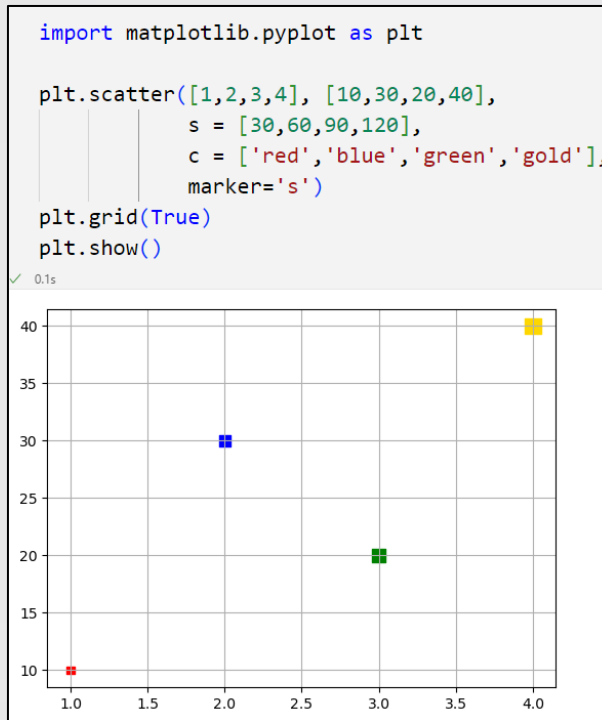


- c 속성은 표현하고 싶은 색상의 수를 의미합니다. 각 버블의 원하는 색상을 c 속성에 지정해 줄 수 있습니다.

## 산점도(Scatter Plot)의 마커

### 마커 지정: marker

"o"		octagon
"s"		square
"p"		pentagon
"P"		plus (filled)
"*"		star
"h"		hexagon1
"H"		hexagon2
"+"		plus





LESSON 02

# 산점도와 colorbar 활용

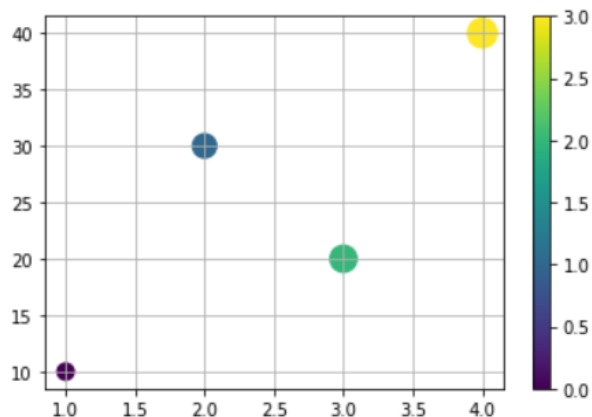


## 산점도(Scatter Plot)의 컬러바

## 컬러바(Color Bar) 추가하기

```
import matplotlib.pyplot as plt

plt.scatter([1, 2, 3, 4], [10, 30, 20, 40], s=[100, 200, 250, 300], c=range(4))
plt.grid(True)
plt.colorbar()
plt.show()
```



- `colorbar()` 함수를 사용하면 그래프 오른쪽에 컬러바가 추가됩니다.
- 4가지 색상을 사용한다는 의미로 `c=range(4)`라고 수정하였습니다.
- 각 데이터는 정해진 색상 개수에 따라 해당하는 컬러바의 색을 부여 받습니다.

## 산점도(Scatter Plot) 컬러바

 camp= 'jet'

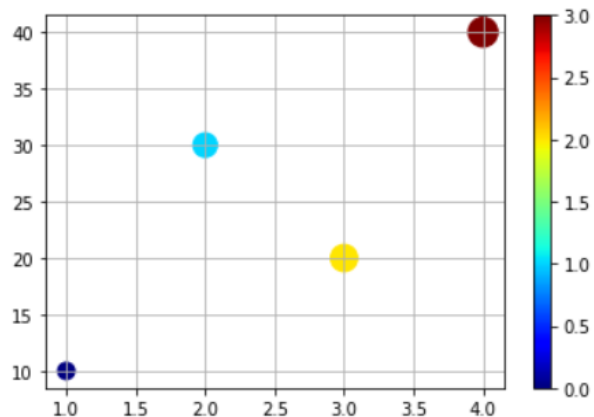
```
import matplotlib.pyplot as plt
```

```
plt.scatter([1, 2, 3, 4], [10, 30, 20, 40], s=[100, 200, 250, 300], c=range(4), cmap='jet')
```

```
plt.grid(True)
```

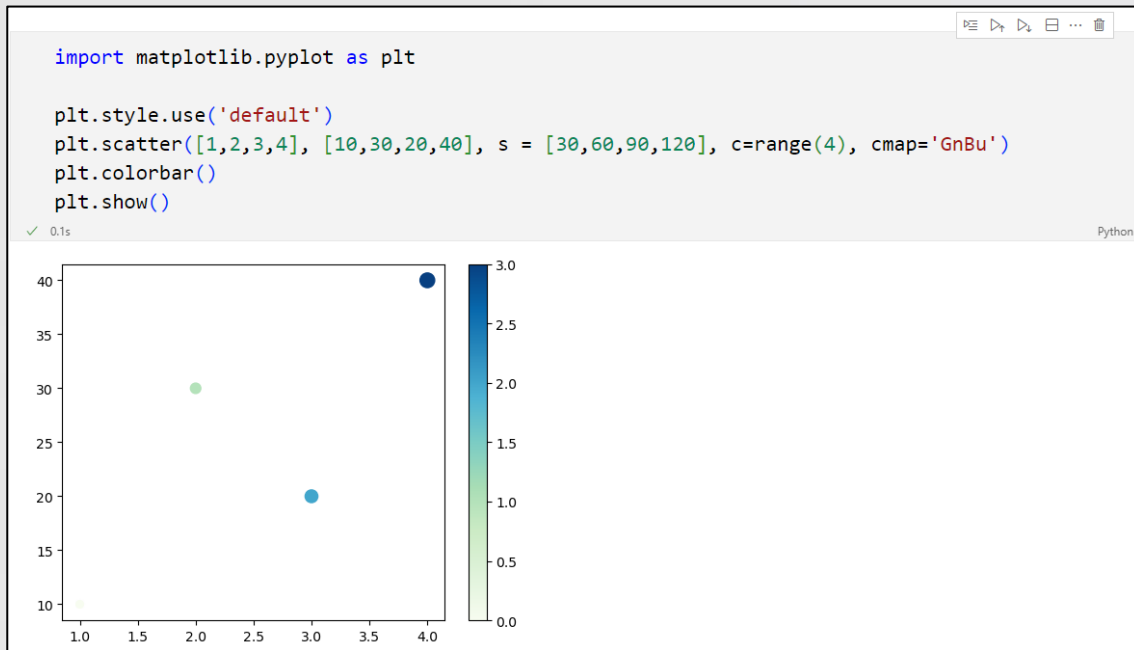
```
plt.colorbar()
```

```
plt.show()
```



- cmap 속성은 Color Map을 의미합니다. 컬러바에 사용될 색상의 종류를 지정할 수 있습니다.

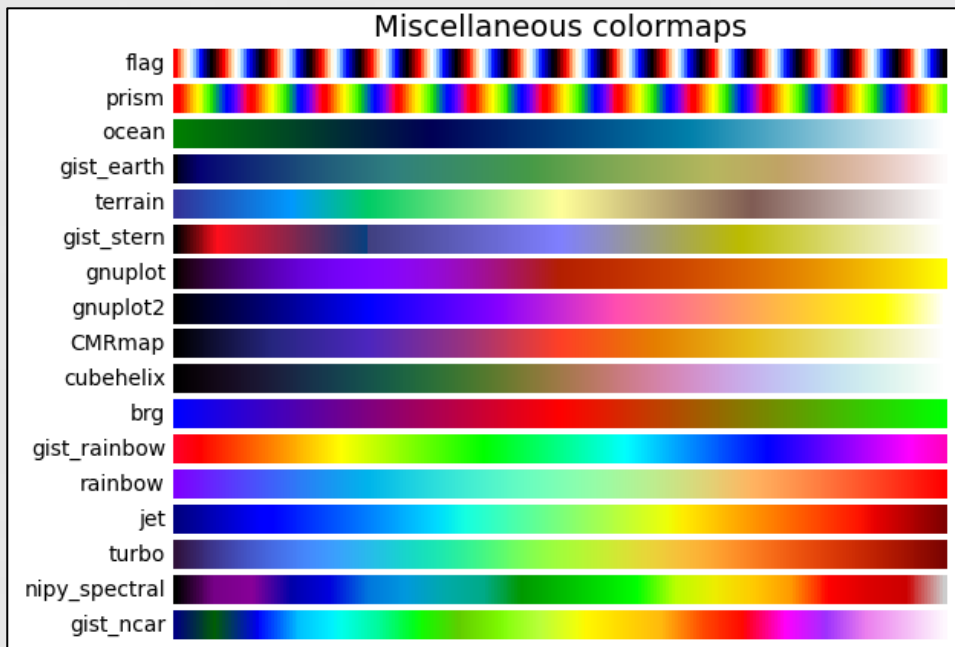
## 산점도(Scatter Plot) 컬러바

 cmap='GnBu'

## 산점도(Scatter Plot) 컬러맵 개요

matplotlib 라이브러리에서 사용할 수 있는 다양한 컬러맵의 종류

[URL] <https://matplotlib.org/stable/tutorials/colors/colormaps.html>



이외에도 다양한 컬러맵이 있습니다.

## 산점도(Scatter Plot) 컬러맵

### matplotlib 라이브러리에서 사용할 수 있는 다양한 컬러맵 목록 보기

```
from matplotlib import colormaps
list(colormaps)
```



```
['magma',
 'inferno',
 'plasma',
 'viridis',
 'cividis',
 'twilight',
 'twilight_shifted',
 'turbo',
 'Blues',
 'BrBG',
 'BuGn',
 'BuPu',
 'CMRmap',
 'GnBu',
 'Greens',
 'Greys',
 'OrRd',
 'Oranges',
 'PRGn',
 'PiYG',
 'PuBu',
 'PuBuGn',
 'PuOr',
 'PuRd',
 'Purples',
 ...
 'Set3_r',
 'tab10_r',
 'tab20_r',
 'tab20b_r',
 'tab20c_r']
```

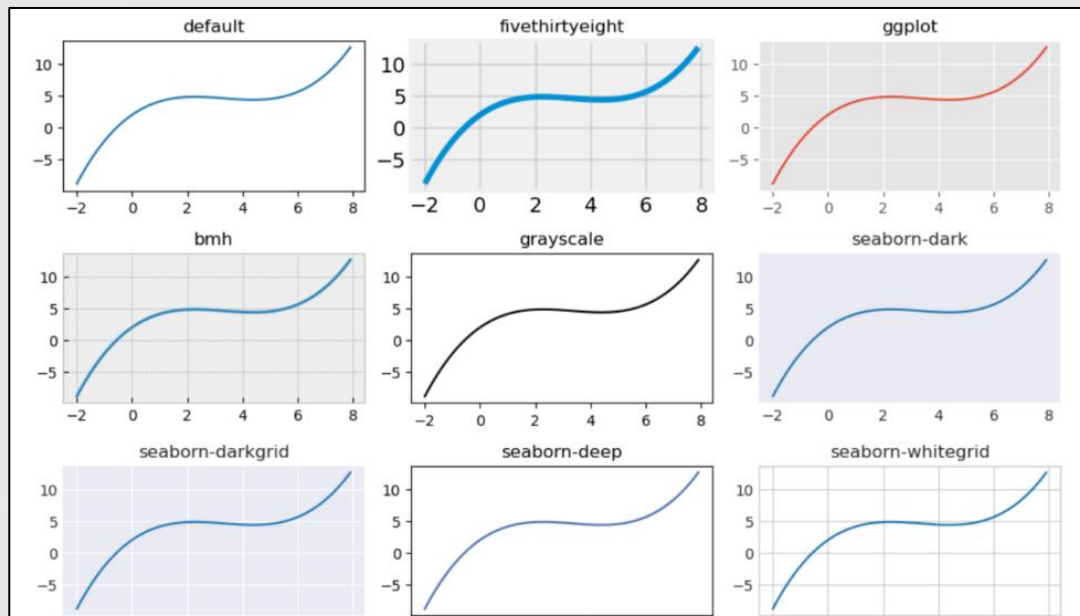
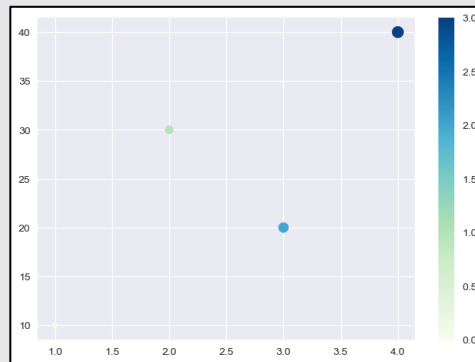
## plt.style

## 그림 스타일

✓ plt.style.use('seaborn-v0\_8')

## 한번 지정 이후 다시 돌아오려면

✓ plt.style.use('default')



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

```
plt.style.available
```

✓ 0.0s

```
['Solarize_Light2',
 '_classic_test_patch',
 '_mpl-gallery',
 '_mpl-gallery-nogrid',
 'bmh',
 'classic',
 'dark_background',
 'fast',
 'fivethirtyeight',
 'ggplot',
 'grayscale',
 'seaborn-v0_8',
 'seaborn-v0_8-bright',
 'seaborn-v0_8-colorblind',
 'seaborn-v0_8-dark',
 'seaborn-v0_8-dark-palette',
 'seaborn-v0_8-darkgrid',
 'seaborn-v0_8-deep',
 'seaborn-v0_8-muted',
 'seaborn-v0_8-notebook',
 'seaborn-v0_8-paper',
 'seaborn-v0_8-pastel',
 'seaborn-v0_8-poster',
 'seaborn-v0_8-talk',
 'seaborn-v0_8-ticks',
 'seaborn-v0_8-white',
 'seaborn-v0_8-whitegrid',
 'tableau-colorblind10']
```

## — 산점도(Scatter Plot)에서 난수 활용

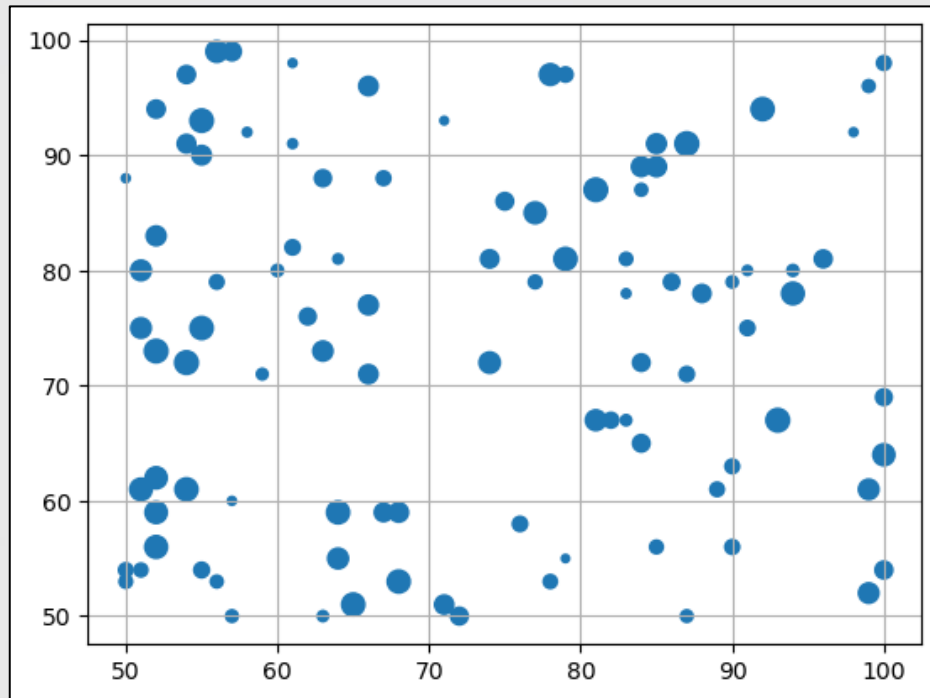
## 📊 위치, 크기가 서로 다른 100개의 점을 산점도 시각화

```
import random
import matplotlib.pyplot as plt

x = []
y = []
size = []

for i in range(100) :
    x.append(random.randint(50,100))
    y.append(random.randint(50,100))
    size.append(random.randint(10,100))

plt.style.use('default')
plt.scatter(x, y, s=size)
plt.grid(True)
plt.show()
```





## 산점도(Scatter Plot) 컬러바 추가

## 컬러맵, 컬러바 추가하기

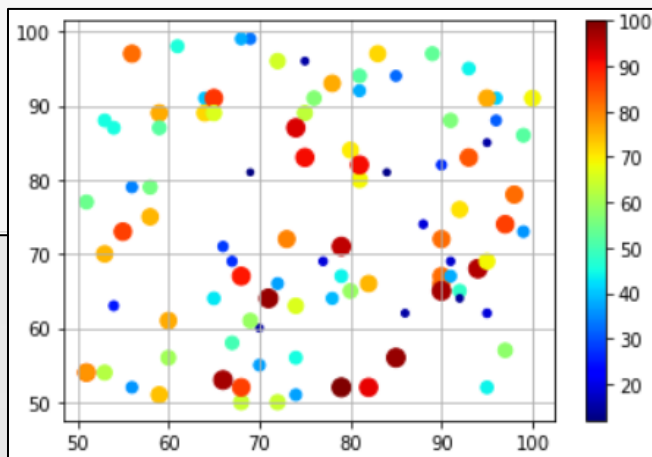
```
import matplotlib.pyplot as plt
import random

x = []
y = []
size = []

for i in range(100):
    x.append(random.randint(50, 100))
    y.append(random.randint(50, 100))
    size.append(random.randint(10, 100))

plt.scatter(x, y, s=size, c=size, cmap='jet')
plt.grid(True)
plt.colorbar()
plt.show()
```

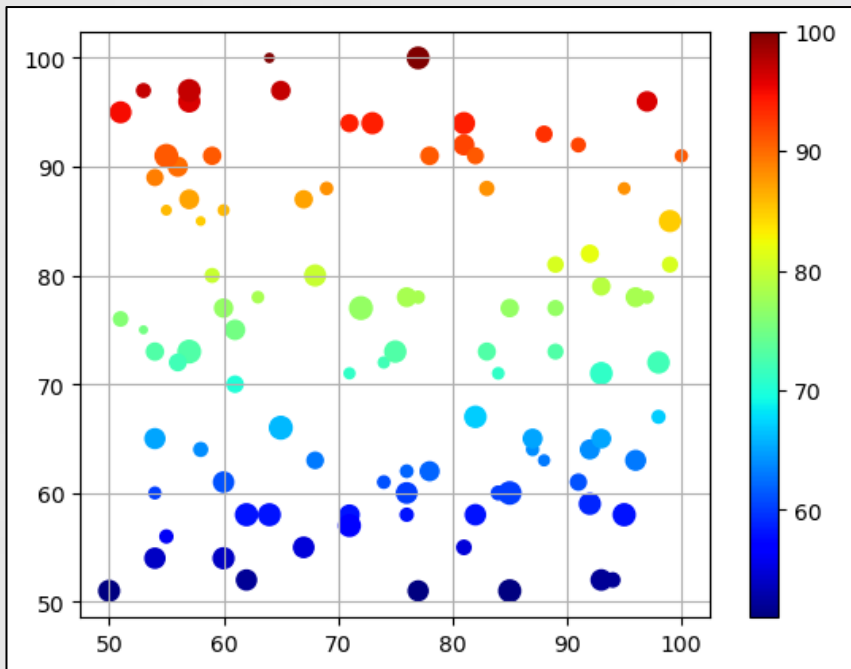
- 작은 점들이 큰 점에 가려서 잘 안 보이네요.  
어떻게 해야 할까요?



## 산점도(Scatter Plot) c=x or y

### c=size

```
plt.scatter(x, y, s=size, c=y, cmap='jet')
plt.colorbar()
```



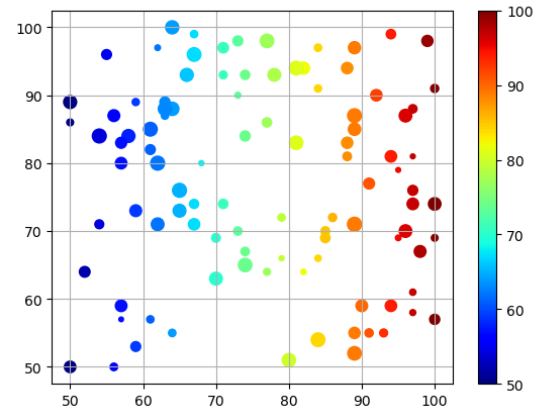
```
import random
import matplotlib.pyplot as plt

x = []
y = []
size = []

for i in range(100) :
    x.append(random.randint(50,100))
    y.append(random.randint(50,100))
    size.append(random.randint(10,100))

plt.scatter(x, y, s=size, c=x, cmap='jet')
plt.colorbar()
plt.grid(True)
plt.show()
```

✓ 0.1s



## 산점도(Scatter Plot) alpha

### 버블의 투명도 설정하기

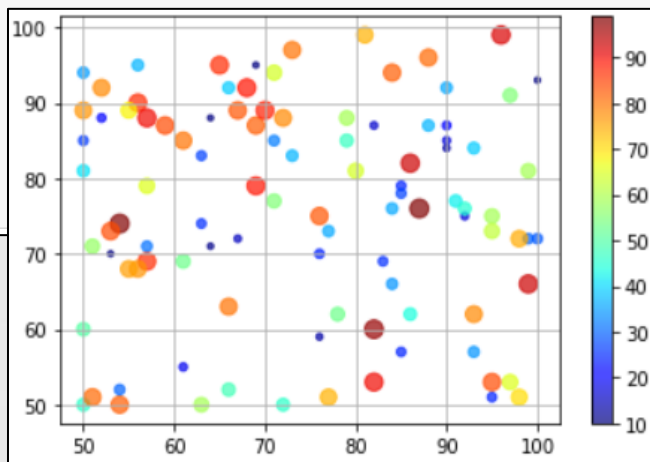
```
import matplotlib.pyplot as plt
import random

x = []
y = []
size = []

for i in range(100):
    x.append(random.randint(50, 100))
    y.append(random.randint(50, 100))
    size.append(random.randint(10, 100))

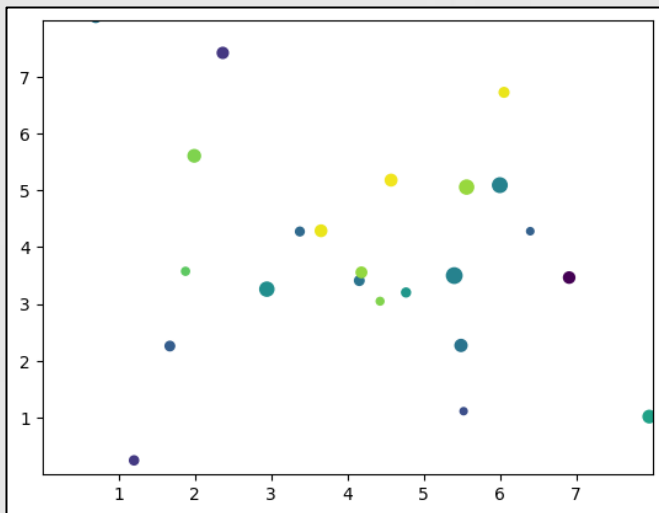
plt.scatter(x, y, s=size, c=size, cmap='jet', alpha=0.7)
plt.grid(True)
plt.colorbar()
plt.show()
```

- alpha 속성값의 범위는 0부터 1까지 입니다.  
0에 가까울 수록 투명해지고, 1에 가까울 수록 불투명합니다.



## 다양한 산점도

### 난수 활용



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

plt.style.use('default')

# make the data
# np.random.seed(3)
x = 4 + np.random.normal(0, 2, 24) # 평균 0, 표준편차 2, 난수 24개 생성
y = 4 + np.random.normal(0, 2, len(x))
# size and color:
sizes = np.random.uniform(15, 80, len(x)) # [15, 80) 사이의 난수 24개 생성
colors = np.random.uniform(15, 80, len(x))

# plot
fig, ax = plt.subplots()
ax.scatter(x, y, s=sizes, c=colors)
ax.set(xlim=(0, 8), xticks=np.arange(1, 8),
       ylim=(0, 8), yticks=np.arange(1, 8))
plt.show()
```

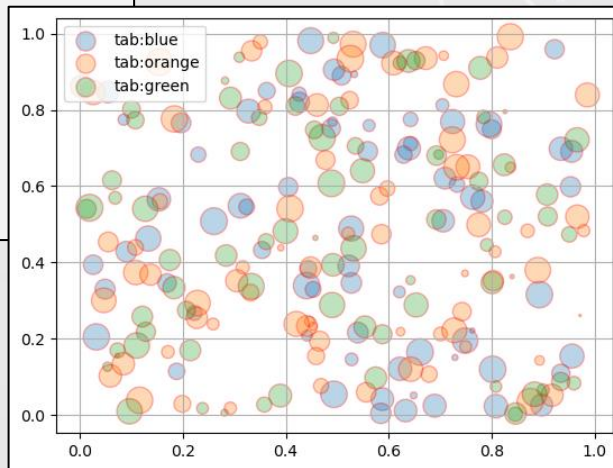
## 산점도(Scatter Plot) 범례

## 색상 범례

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

# np.random.seed(19680801)
fig, ax = plt.subplots()
for color in ['tab:blue', 'tab:orange', 'tab:green']:
    n = 75
    x, y = np.random.rand(2, n) # 2행 75열, 1행을 x, 2행을 y에 저장
    scale = 300.0 * np.random.rand(n) # [0, 1) 난수 75개
    ax.scatter(x, y, c=color, s=scale, label=color,
               alpha=0.3, edgecolors='red')

ax.legend()
ax.grid(True)
plt.show()
```



## 3차원 산점도(Scatter Plot)

 x, y, z 축

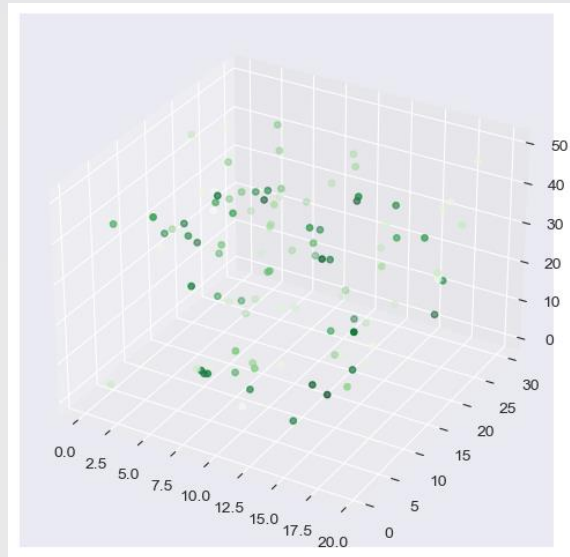
```
# from mpl_toolkits.mplot3d import Axes3D
import matplotlib.pyplot as plt
import numpy as np

n = 100
xmin, xmax, ymin, ymax, zmin, zmax = 0, 20, 0, 30, 0, 50
cmin, cmax = 0, 2

# (b - a) * random_sample() + a : [a, b), b > a` random number
xs = np.array([(xmax - xmin) * np.random.random_sample() + xmin for i in range(n)])
ys = np.array([(ymax - ymin) * np.random.random_sample() + ymin for i in range(n)])
zs = np.array([(zmax - zmin) * np.random.random_sample() + zmin for i in range(n)])
color = np.array([(cmax - cmin) * np.random.random_sample() + cmin for i in range(n)])

fig = plt.figure(figsize=(6, 6))
ax = fig.add_subplot(111, projection='3d')
ax.scatter(xs, ys, zs, c=color, marker='o', s=15, cmap='Greens')

plt.show()
```



SUMMARY

# 학습정리



## ⚙ 산점도 함수 `scatter(x, y, ...)`

- `c, s, marker`
- `label, edgecolor`

## ⚙ 산점도와 `colorbar` 활용

- `cmap`
- `colorbar()`

## ⚙ 전체 그래프의 스타일 지정

- `plt.style.use('ggplot')`
- `plt.style.use('seaborn-v0_8')`
- `plt.style.use('default')`

