

Pie plot

Pie chart có thể biểu diễn tỉ lệ dữ liệu dạng phân loại trên tổng của toàn bộ dữ liệu. Tổng thể dữ liệu là một cái bánh, mỗi loại dữ liệu chiếm một phần của cái bánh đó.

Matplotlib cung cấp hàm `pie()` để vẽ biểu đồ pie (gọi là biểu đồ tròn hay quạt gì đó), có các thuộc tính:

- `x`: list giá trị truyền vào
- `explode`: truyền vào None hoặc list các giá trị có cùng độ dài với `x`, mỗi phần tử là khoảng cách từ tâm miếng bánh đến tâm biểu đồ
- `labels`: tên mỗi miếng bánh
- `colors`: None hoặc list các màu mỗi miếng bánh
- `shadow`: True/False, mặc định là False, lựa chọn thêm hiệu ứng bóng mờ cho biểu đồ
- `counterlock`: True/False, mặc định là True, hướng của các tỉ lệ
- `autopct`: None hoặc string, chứa format string các nhãn chú thích lên trên mỗi miếng bánh

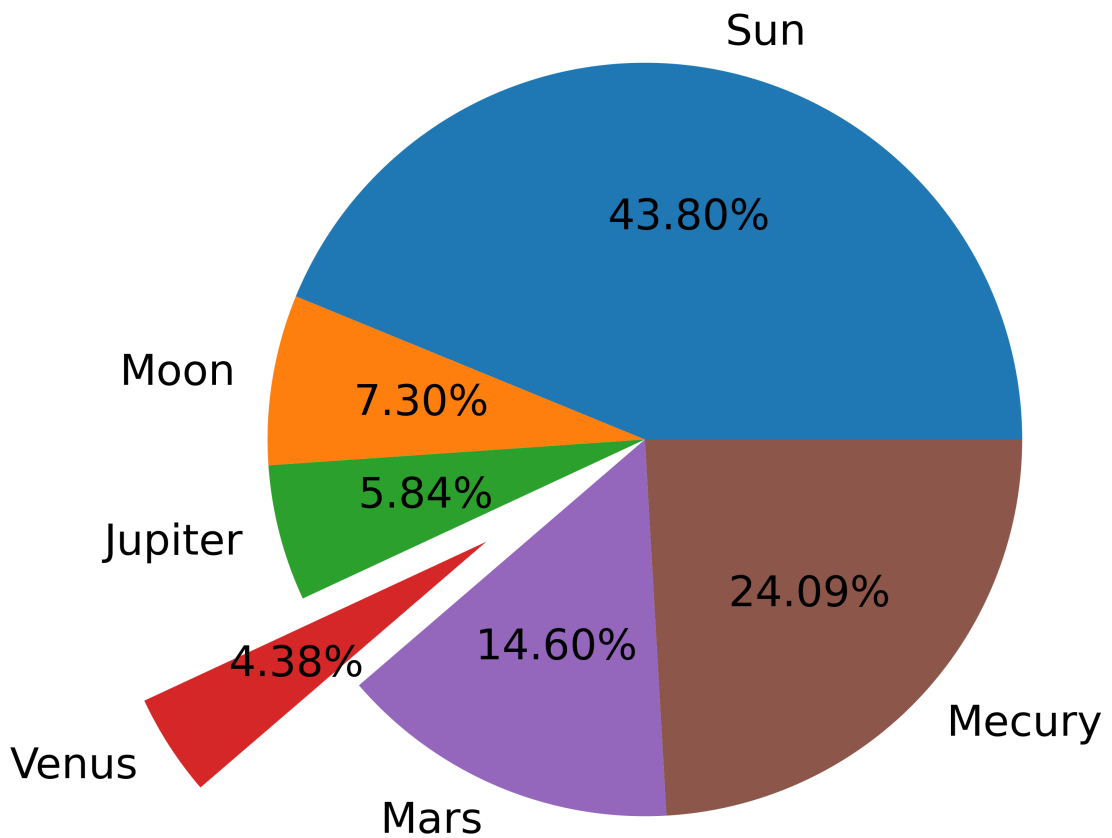
```
In [2]: import matplotlib.pyplot as plt

labels = ["Sun", "Moon", "Jupiter", "Venus", "Mars", "Mecury"]
values = [600, 100, 80, 60, 200, 330]

# Ví dụ ta muốn nhấn mạnh tỉ lệ của Mars,
# ta sẽ sửa khoảng cách từ tâm miếng Mars đến tâm biểu đồ = 0.5
explode = [0, 0, 0, 0.5, 0, 0]

fig, axes = plt.subplots(dpi=800, figsize=(4, 4))
axes.pie(values, labels=labels, autopct='%.2f%%', explode=explode)
```

```
Out[2]: ([<matplotlib.patches.Wedge at 0x7fe39751d280>,
<matplotlib.patches.Wedge at 0x7fe39751d9a0>,
<matplotlib.patches.Wedge at 0x7fe39751dfa0>,
<matplotlib.patches.Wedge at 0x7fe397533700>,
<matplotlib.patches.Wedge at 0x7fe397533e20>,
<matplotlib.patches.Wedge at 0x7fe39753e580>],
[Text(0.21305286529944611, 1.0791702722868604, 'Sun'),
Text(-1.085858860920759, 0.17581391913006217, 'Moon'),
Text(-1.0651900959848022, -0.2745360803535448, 'Jupiter'),
Text(-1.3441994118593803, -0.8678294424349154, 'Venus'),
Text(-0.42967523513105893, -1.0126100889854246, 'Mars'),
Text(0.7997903711551759, -0.7552055099159871, 'Mecury')],
[Text(0.11621065379969786, 0.5886383303382874, '43.80%'),
Text(-0.592286651411323, 0.09589850134367026, '7.30%'),
Text(-0.5810127796280738, -0.14974695292011533, '5.84%'),
Text(-0.9241370956533239, -0.5966327416740043, '4.38%'),
Text(-0.23436831007148667, -0.5523327758102315, '14.60%'),
Text(0.4362492933573686, -0.411930278135993, '24.09%')])])
```



Vẽ nhiều biểu đồ lồng nhau

Khi muốn so sánh tỉ lệ nhiều dataset với nhau, ta có thể đặt 2 hoặc nhiều pie lồng vào nhau. Để xịn xò, có nhiều hiệu ứng, đặc biệt là khi hiển thị trên website, js có thể được sử dụng kết hợp để vẽ biểu đồ dạng này. Ví dụ khi muốn mô tả tỉ lệ các ngành nghề tại một quốc gia, trong mỗi ngành nghề lại có nhiều chuyên ngành nhỏ hơn cần biểu diễn:

- IT -> (data science, SA, BE, AI engineer,...)

```
In [3]: import matplotlib.pyplot as plt

labels1 = ["Sun", "Moon", "Jupiter", "Venus", "Mars", "Mecury"]
labels2 = ["US", "UK", "JP"]
values1 = [600, 100, 80, 60, 200, 330]
values2 = [300, 100, 150]
explode = [0, 0, 0, 0.5, 0, 0]

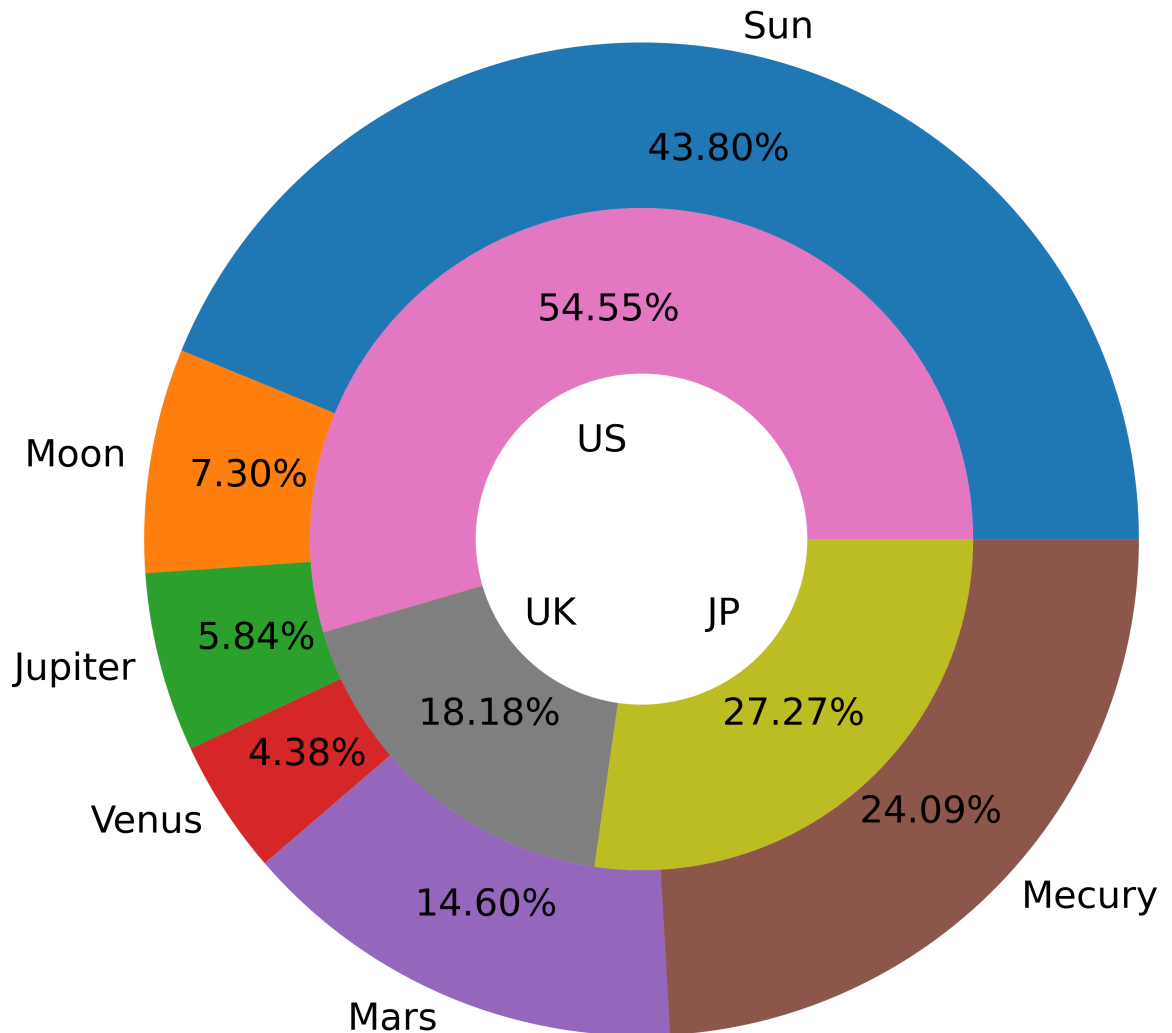
fig, axes = plt.subplots(dpi=800, figsize=(4, 4))

# outer circle
axes.pie(values1, radius=1.5, wedgeprops=dict(width=0.5),
        autopct='%.2f%%', pctdistance=0.8, labels=labels1, labeldistance=1.0)

# inner circle
axes.pie(values2, radius=1.5 - 0.5, wedgeprops=dict(width=0.5),
        autopct='%.2f%%', pctdistance=0.7, labels=labels2, labeldistance=0.3)

# ví dụ này đi copy cho nhanh :))
```

```
Out[3]: ([<matplotlib.patches.Wedge at 0x7fe37aeacd0>,
<matplotlib.patches.Wedge at 0x7fe37af93a00>,
<matplotlib.patches.Wedge at 0x7fe37afa5160>],
[Text(-0.04269446664678446, 0.2969464303839105, 'US'),
Text(-0.19645819316674754, -0.22672489571649804, 'UK'),
Text(0.19645825684928972, -0.2267248405352587, 'JP')],
[Text(-0.0996204221758304, 0.6928750042291245, '54.55%'),
Text(-0.45840245072241087, -0.5290247566718287, '18.18%'),
Text(0.4584025993150094, -0.5290246279156037, '27.27%')])
```



Đây là 2 ví dụ đơn giản về cách vẽ pie chart, để tùy chỉnh thêm cho phù hợp với sở thích và yêu cầu dữ liệu có thể tham khảo qua trang chủ của matplotlib, lựa chọn các thuộc tính phù hợp:

https://matplotlib.org/stable/gallery/pie_and_polar_charts/pie_features.html

In []: