

TECHMIMO

Autor: Rafael Pereira da Silva

Seguem alguns recados para ajudá-los e para contribuir com o curso:

- Fiquem à vontade para me contatar pelo LinkedIn, costumo responder por lá também:
<https://www.linkedin.com/in/rafael-pereira-da-silva-23890799/> (<https://www.linkedin.com/in/rafael-pereira-da-silva-23890799/>)
- Fiquem a vontade para compartilharem o certificado do curso no LinkedIn. Eu costumo curtir e comentar para dar mais credibilidade
- Vocês podem usar esses notebooks para resolver os exercícios e desafios
- Não se esqueçam de avaliar o curso e dar feedback, eu costumo criar conteúdos baseado nas demandas de vocês
- Se tiverem gostando do curso, recomendem aos amigos, pois isso também ajuda a impulsionar e a crescer a comunidade
- Bons estudos e grande abraços!

Seção 10 - Ferramentas Gráficas

10.1 Matplotlib

10.1.1 Pyplot e elementos gráficos

Agenda

- Uso no Jupyter notebook: **%matplotlib notebook**
- Usando o pyplot: **import matplotlib.pyplot as plt**
- **Marcadores** : https://matplotlib.org/3.3.3/api/_as_gen/matplotlib.pyplot.plot.html
(https://matplotlib.org/3.3.3/api/_as_gen/matplotlib.pyplot.plot.html)
- Elementos gráficos: Marcadores, títulos, eixos, grade e legenda

In [2]:

```
import matplotlib.pyplot as plt
%matplotlib notebook
```

In [5]:

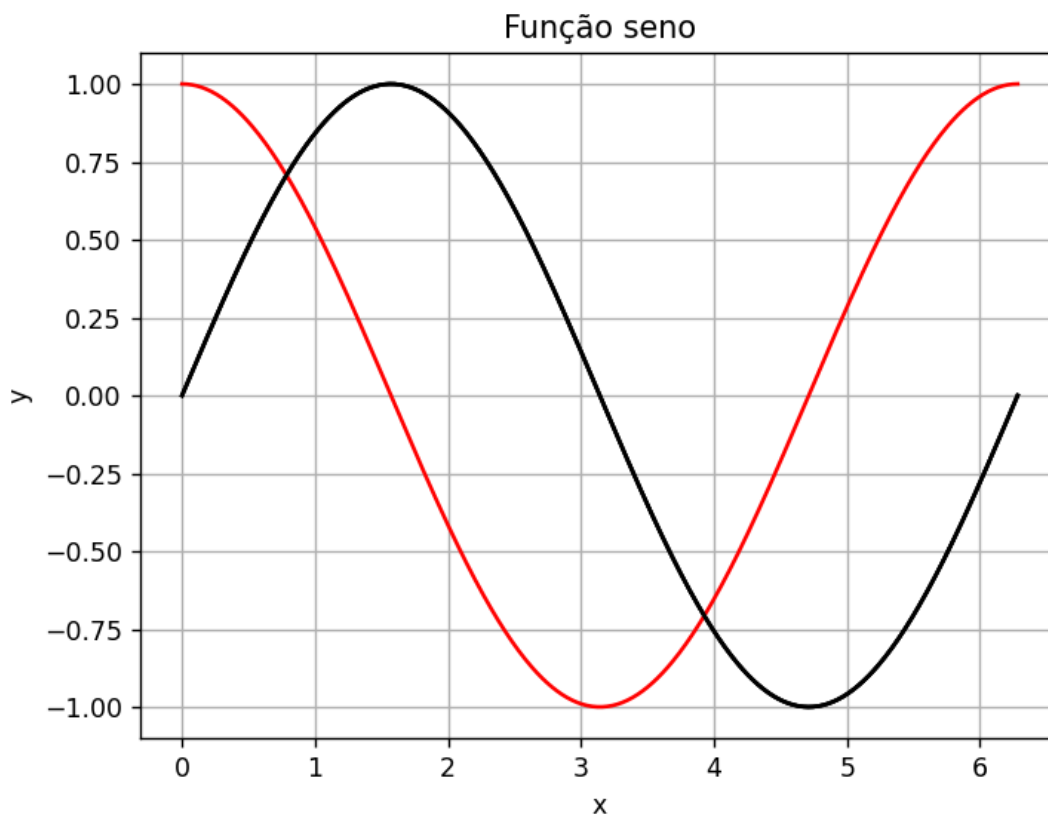
```
import numpy as np
xa_array = np.linspace(0,2*np.pi,500)
ya_array = np.sin(xa_array)

xb_array = np.linspace(0,2*np.pi,500)
yb_array = np.sin(xb_array+np.pi/2)

#grafico_obj = plt.plot()

plt.plot(xa_array,ya_array,'-k')
plt.plot(xb_array,yb_array,'-r')
plt.xlabel('x')
plt.ylabel('y')
plt.title('Função seno')

plt.grid()
```



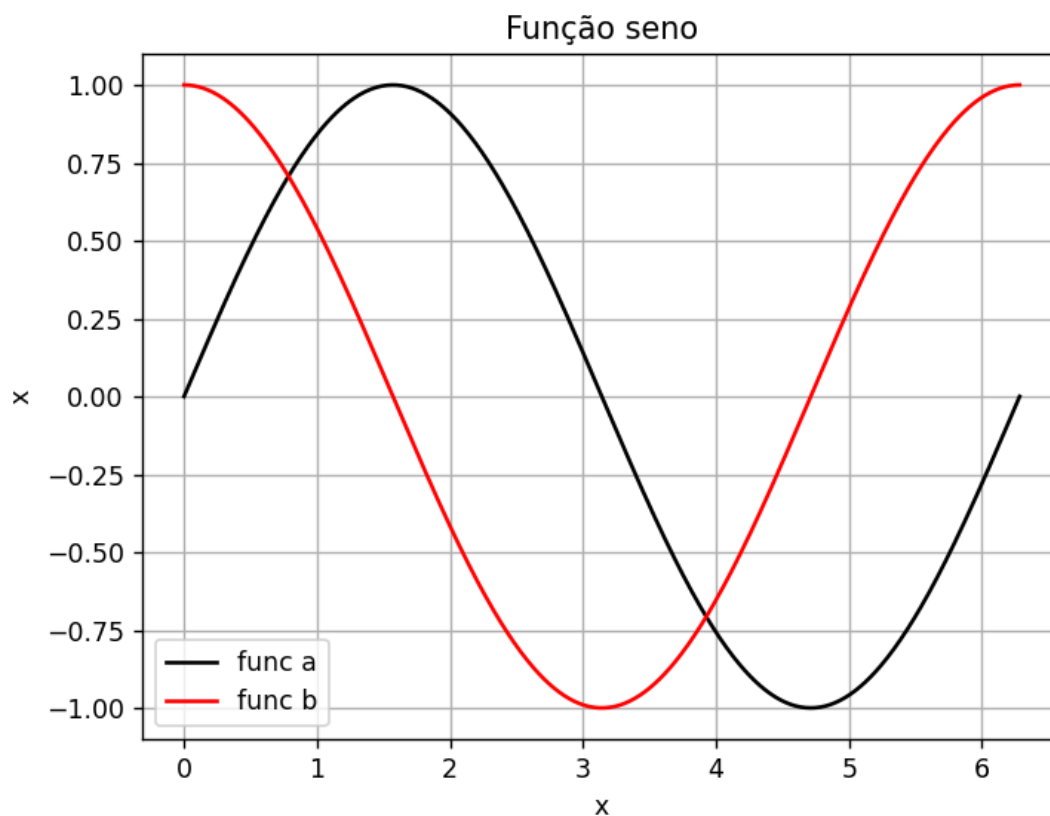
In []:

In [10]:

```
## Uma melhor prática

fig, ax = plt.subplots()
ax.plot(xa_array, ya_array, 'k-', label='func a')
ax.plot(xb_array, yb_array, 'r-', label='func b')

ax.set(xlabel='x', ylabel='x',
       title='Função seno')
ax.grid()
plt.legend()
#plt.show() Necessário para algumas IDEs
```



Out[10]:

<matplotlib.legend.Legend at 0x19d1ae3b3a0>

In []:

10.1.2 - Subplots

- Dica: subplots são armazenados em arrays numpy
- https://matplotlib.org/gallery/subplots_axes_and_figures/subplot.html
(https://matplotlib.org/gallery/subplots_axes_and_figures/subplot.html)

In []:

```
import matplotlib.pyplot as plt
%matplotlib notebook
```

In [75]:

```
xa_array = np.linspace(0,100,500)
ya_array = np.exp(xa_array)

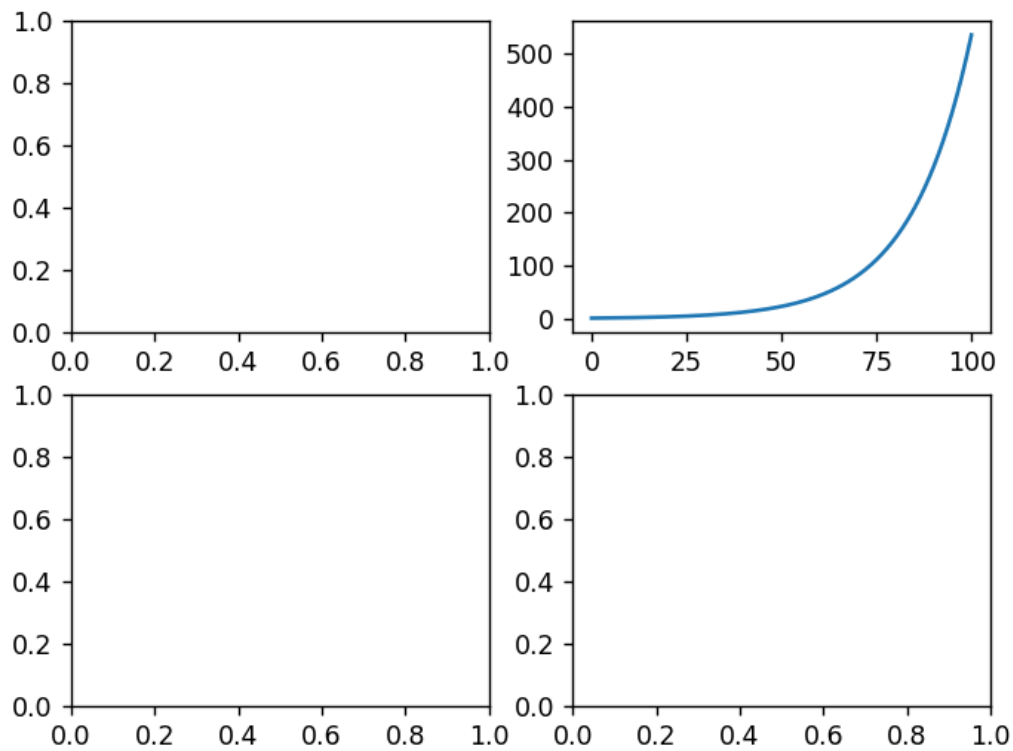
xb_array = np.linspace(0,100,500)
yb_array = np.exp(2*xb_array)

fig, (ax1, ax2) = plt.subplots(2,2)
fig.suptitle('A tale of 2 subplots')

ax1[1].plot(xa_array,ya_array)

plt.show()
```

A tale of 2 subplots

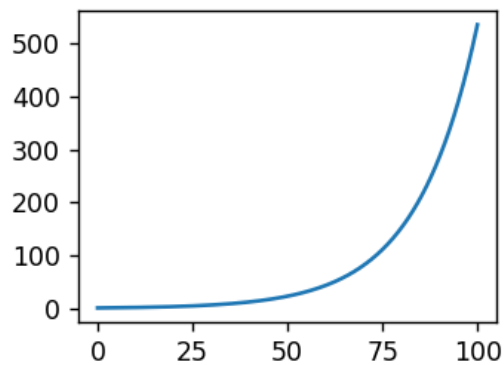
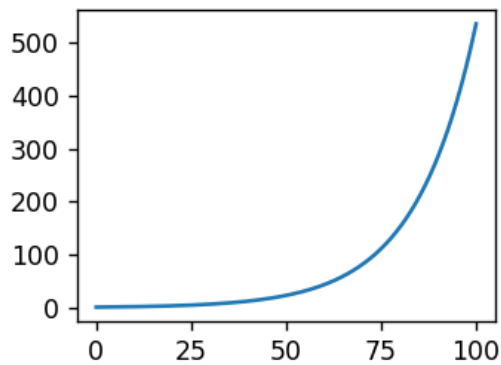


In [104]:

```
plt.figure()

plt.subplot(221)
plt.plot(xa_array,ya_array)

plt.subplot(224)
plt.plot(xa_array,ya_array)
```



Out[104]:

```
[<matplotlib.lines.Line2D at 0x17fe6e67970>]
```

10.1.3 - Escala Log

<https://matplotlib.org/tutorials/introductory/pyplot.html#sphx-glr-tutorials-introductory-pyplot-py>
(<https://matplotlib.org/tutorials/introductory/pyplot.html#sphx-glr-tutorials-introductory-pyplot-py>)

In [81]:

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

In [116]:

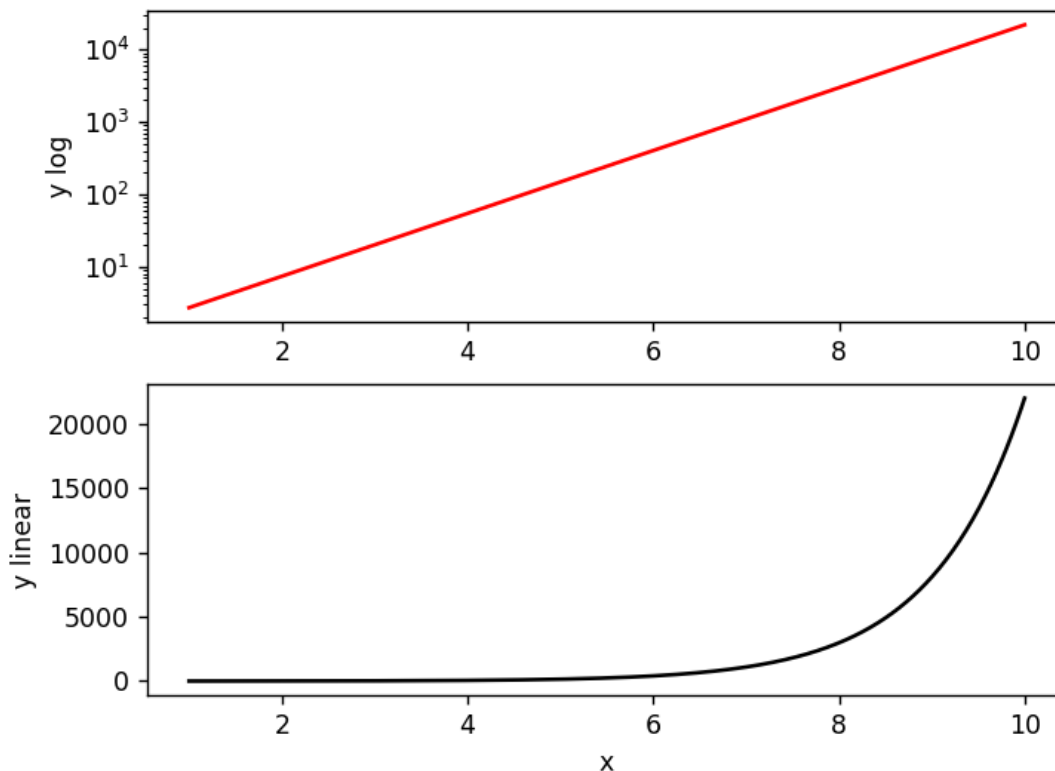
```
x_array = np.linspace(1,10,500)
y_array = np.exp(x_array)

plt.figure()
plt.suptitle('comparação escalas')

plt.subplot(211)
plt.plot(x_array,y_array,'r-')
plt.ylabel('y log')
plt.yscale('log')

plt.subplot(212)
plt.plot(x_array,y_array,'k-')
plt.xlabel('x')
plt.ylabel('y linear')
plt.yscale('linear')
```

comparação escalas



10.1.4 - Eixo secundário

https://matplotlib.org/3.1.0/gallery/subplots_axes_and_figures/secondary_axis.html
[\(https://matplotlib.org/3.1.0/gallery/subplots_axes_and_figures/secondary_axis.html\)](https://matplotlib.org/3.1.0/gallery/subplots_axes_and_figures/secondary_axis.html)

<https://stackoverflow.com/questions/14762181/adding-a-y-axis-label-to-secondary-y-axis-in-matplotlib>
<https://stackoverflow.com/questions/14762181/adding-a-y-axis-label-to-secondary-y-axis-in-matplotlib>

Uso do objeto gêmeo >>twinx()<<

In [117]:

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

In [22]:

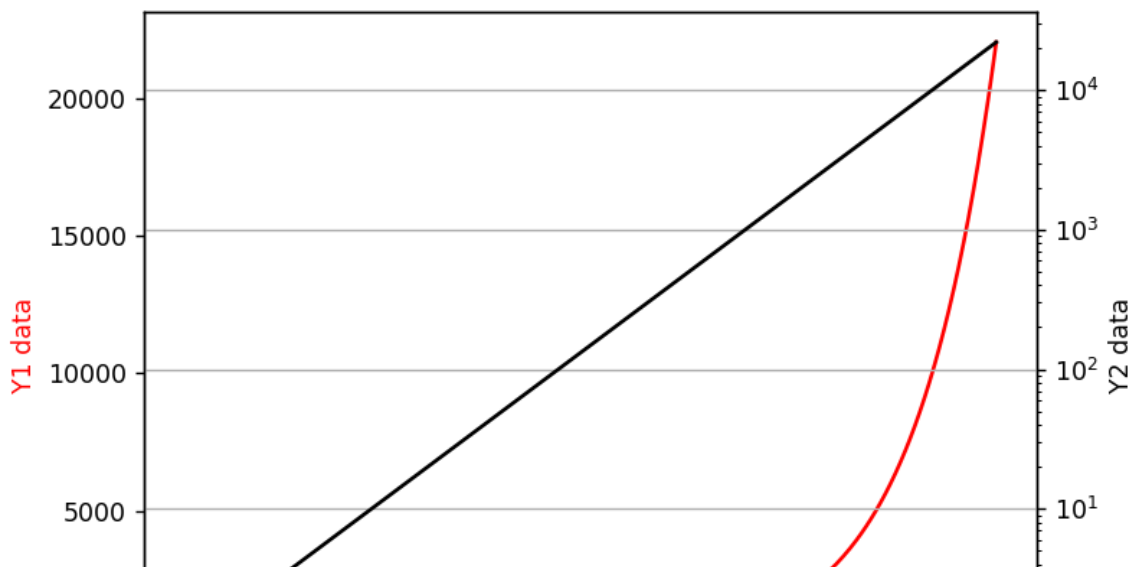
```
x_array = np.linspace(0,10,500)
y_array = np.exp(x_array)

fig,ax1 = plt.subplots()

ax2 = ax1.twinx()
ax1.plot(x_array, y_array, 'r-',label='linear')
ax2.plot(x_array, y_array, 'k-',label='log')

ax1.set_xlabel('X data')
ax1.set_ylabel('Y1 data', color='r')
ax2.set_ylabel('Y2 data', color='k')

plt.yscale('log')
plt.grid()
plt.show()
```



In []:

10.1.5 - Barras

https://matplotlib.org/3.1.1/api/_as_gen/matplotlib.pyplot.bar.html
(https://matplotlib.org/3.1.1/api/_as_gen/matplotlib.pyplot.bar.html)

`matplotlib.pyplot.bar(x, height, width=0.8, bottom=None, , align='center', data=None, *kwargs)`

In [1]:

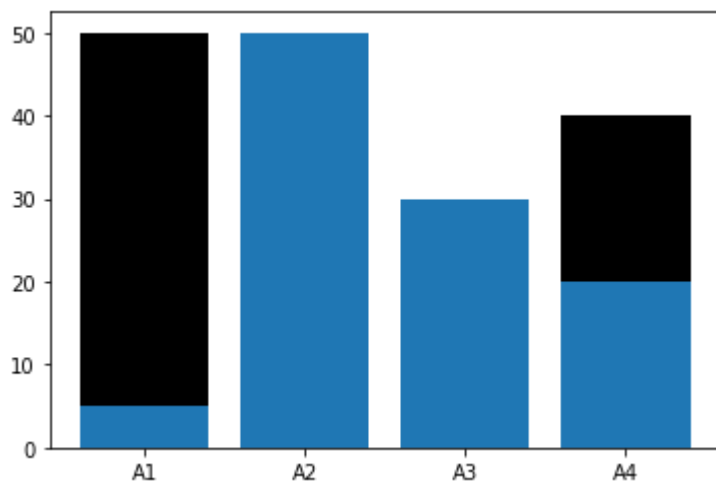
```
import matplotlib.pyplot as plt
```

In [15]:

```
nomes = ['A1', 'A2', 'A3', 'A4']  
valores = [50, 10, 20, 40]  
valores2 = [5, 50, 30, 20]  
  
plt.bar(nomes, valores, color='k')  
plt.bar(nomes, valores2)
```

Out[15]:

<BarContainer object of 4 artists>



In [31]:

```
fig, ax = plt.subplots()

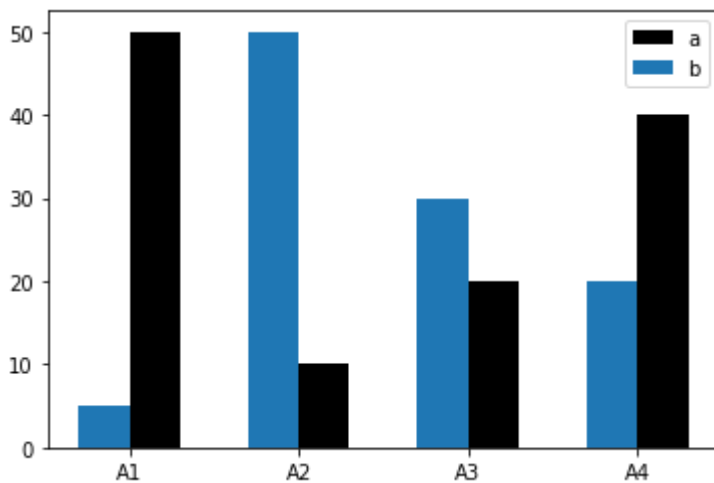
x = np.arange(4)

rect1 = ax.bar(x+largura/2, valores,color='k',width = largura,label='a')
rect2 = ax.bar(x-largura/2, valores2,width = largura,label='b')

ax.set_xticks(x)
ax.set_xticklabels(nomes)
plt.legend()
```

Out[31]:

<matplotlib.legend.Legend at 0x268d49ac0d0>



In []:

10.1.6 - Pizza

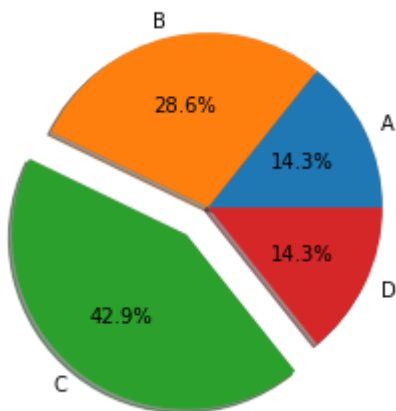
matplotlib.pyplot.pie(x, explode=None, labels=None, colors=None, autopct=None, pctdistance=0.6, shadow=False, labeldistance=1.1, startangle=None, radius=None, counterclock=True, wedgeprops=None, textprops=None, center=(0, 0), frame=False, rotatelabels=False, *, data=None)

In [45]:

```
fig, ax = plt.subplots()
nomes = ('A', 'B', 'C', 'D')
valores = [10, 20, 30, 10]
explodir = (0, 0, 0.2, 0)

ax.pie(valores, labels=nomes, autopct='%1.1f%%', shadow = True, explode=explode)

plt.show()
```



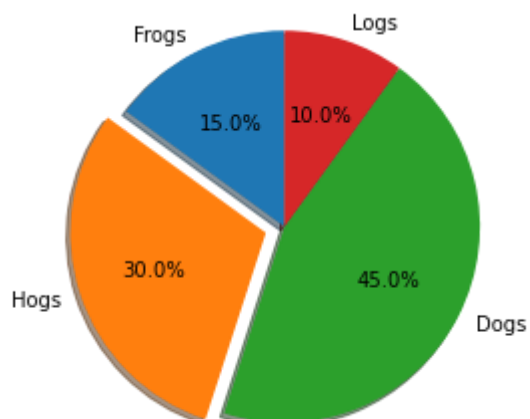
In [36]:

```
import matplotlib.pyplot as plt

# Pie chart, where the slices will be ordered and plotted counter-clockwise:
labels = 'Frogs', 'Hogs', 'Dogs', 'Logs'
sizes = [15, 30, 45, 10]
explode = (0, 0.1, 0, 0) # only "explode" the 2nd slice (i.e. 'Hogs')

fig1, ax1 = plt.subplots()
ax1.pie(sizes, explode=explode, labels=labels, autopct='%1.1f%%',
        shadow=True, startangle=90)
ax1.axis('equal') # Equal aspect ratio ensures that pie is drawn as a circle.

plt.show()
```



In []: