Modelação de Sistemas Físicos - Aula Prática nº1

Nome: José Coutinho

email: `jose.coutinho@ua.pt`

Gabinete: 13.2.2.1 (Departamento de Física)

Avaliação Prática:

- Mostrar e explicar o trabalho feito durante a aula, 12 valores;
- Responder às perguntas escritas no ficheiro do trabalho, 8 valores;

Software:

- Spyder IDE, https://www.spyder-ide.org/
- Jupyter Notebook (Install on windows using Anaconda), https://www.xda-developers.com/how-install-jupyter-notebook-windows/

Ambiente Conda:

```
$ conda create --name msf
$ conda activate msf
(msf) $ conda install python=3.8 ipykernel numpy matplotlib sympy
(msf) $ python -m ipykernel install --user --name msf --display-name
'MSF'
(msf) $ conda deactivate
```

Exercício 1

Foram medidos dois comprimentos:

$P = 18.2 \pm 0.1$ cm

$Q = 10.7 \pm 0.2$ cm

a) Calcule a soma das duas quantidades, $S = P + Q$

b) Calcule a diferença das duas quantidades, $D = P - Q$

c) Calcule o produto das duas quantidades, $M = P \times Q$

Solução:

a) $S = 18.2 + 10.7 \pm (|0.1| + |0.2|)$ cm

$$S = 28.9 \pm 0.3 \text{ cm}$$

b) $P - Q = 18.2 - 10.7 \pm (|0.2| + |0.2|)$ cm

$$P - Q = 7.5 \pm 0.3 \text{ cm}$$

c) $P \times Q = 18.2 \times 10.7 \pm (0.1/18.2 + 0.2/10.7) \times (18.2 \times 10.7) \text{ cm}^2$

$$P \times Q = 194.7 \pm 4.7 \text{ cm}^2$$

Exercício 2

Reescreva corretamente as seguintes quantidades, tendo em conta o número de algarismos significativos:

a) 4.10 ± 0.02 m

b) 4.10 ± 0.2 m

c) 141.5 ± 2 s

d) 23.30 ± 0.25 kg

Solução

a) 4.10 ± 0.02 m

b) 4.1 ± 0.2 m

c) 142 ± 2 s

d) 23.30 ± 0.25 kg

Experiência numérica

Vamos simular a medição de valores de duas variáveis, X e Y, e explorar a incerteza ao utiliza-las em operações algébricas.

1. Verificar as bibliotecas

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

2. Simular medições com incerteza

Use o seguinte código para gerar $N = 10$ valores de X com media esperada $\bar{X} = 4.5$ e desvio padrão $\sigma_X = 0.5$, e calcular a sua media e incerteza:

In [13]:
```
# número de medições
N = 10

# gerar N valores de X com média 4.5 e desvio padrão 0.5
X = np.random.normal(4.5, 0.5, size = N)
Xmedia = np.mean(X)
Xerro = np.std(X)/np.sqrt(N)

print("X = ", ["{:.1f}".format(Xi) for Xi in X])
print("Xmedia = {:.1f}".format(Xmedia))
print("Xerro = {:.1f}".format(Xerro))
```

```
X =  ['4.9', '4.7', '4.5', '4.3', '3.9', '4.1', '4.8', '4.1', '4.1', '4.5']
Xmedia = 4.4
Xerro = 0.1
```

Adapte o Código para gerar outro conjunto de $N = 10$ valores, Y, com media $\bar{Y} = 10.0$ e desvio padrão $\sigma_Y = 1.0$,

In [14]:
```
# gerar n valores de Y com média 12 e desvio padrão 0.7\
Y = np.random.normal(10.0, 1.0, size = N)
Ymedia = np.mean(Y)
Yerro = np.std(Y)/np.sqrt(N)

print("Y = ", ["{:.1f}".format(Yi) for Yi in Y])
print("Ymedia = {:.1f}".format(Ymedia))
print("Yerro = {:.1f}".format(Yerro))
```

```
Y =  ['12.1', '10.1', '10.0', '9.5', '10.7', '9.6', '8.8', '9.7', '11.5', '9.1']
Ymedia = 10.1
Yerro = 0.3
```

3. Calcular somas e erros associados

Crie um novo conjunto de $N = 10$ valores, Z, que seja a soma elementar de X e Y ($Z_i = X_i + Y_i$) calcule a média dos valores de Z:

In [16]:
```
# soma de X e Y elemento-a-elemento
Z = X + Y
Zmedia = np.mean(Z)
print("Zmedia = {:.1f}".format(Zmedia))
```

```
Zmedia = 14.5
```

Estima a incerteza em Z de duas maneiras:

i) Calcule a incerteza diretamente do desvio padrão dos valores de Z

In [17]:
```
# Calcular a incerteza na média de Z, diretamente do desvio padrão dos valores
Zerro = np.std(Z)/np.sqrt(N)
print("Zerro = {:.1f}".format(Zerro))
```

```
Zerro = 0.3
```

ii) Use as incertezas de X e Y, previamente calculadas, para estimar a incerteza em Z

In [19]:
```
# Calcular incerteza na média de Z por via da propagação de erros e comparar
Zerro_frm = Xerro + Yerro
print("Zerro_frm = {:.1f}".format(Zerro_frm))
```

```
Zerro_frm = 0.4
```

4. Calcular o produto e o erro associado

In [20]:
```
# produto de X e Y elemento-a-elemento
W = X * Y
Wmedia = np.mean(W)
print("Wmedia = {:.1f}".format(Wmedia))
```

```
Wmedia = 44.2
```

Estima a incerteza em W das mesmas duas maneiras.

Incerteza diretamente obtida do desvio padrão dos valores de W

In [23]:
```
# Calcular a incerteza na média de W, usando o desvio padrão dos valores
Werro = np.std(W)/np.sqrt(N)
print("Werro = {:.1f}".format(Werro))
```

```
Werro = 1.8
```

Incerteza de W com base nas incertezas de X e Y, previamente calculadas (usando a fórmula da propagação dos erros)

In [25]:
```
# Calcular incerteza na média de W com fórmula e comparar
W_frm = Xmedia * Ymedia
Werro_frm = (np.abs(Xerro / Xmedia) + np.abs(Yerro / Ymedia)) * W_frm
print("W_frm = {:.1f}".format(W_frm))
print("Werro_frm = {:.1f}".format(Werro_frm))
```

```
W_frm = 44.2
Werro_frm = 2.4
```

Pergunta 1:

Os erros obtidos pelas fórmulas de propagação (ii) concordam com os resultados encontrado pelo método direto (i)? Comente.

- A fórmula de propagação dos erros **majora** o erro obtido pelo desvio padrão.

$$\Delta Z = \Delta X + \Delta Y > \sigma_Z / \sqrt{N}$$

em que $Z = \bar{X} + \bar{Y}$

$$\Delta W = |W| \left| \frac{\Delta X}{X} \right| + |W| \left| \frac{\Delta Y}{Y} \right| > \sigma_W / \sqrt{N},$$

em que $W = \bar{X} \times \bar{Y}$

Exercício 4

Agora vamos verificar se a incerteza na média de N medições diminui realmente como $1/\sqrt{N}$.

1. Escreva código para simular N medições com uma determinada media e desvio padrão esperada (como no

exercício anterior), repetindo-se o procedimento para valores de N múltiplos 10, de 10 até 10000.

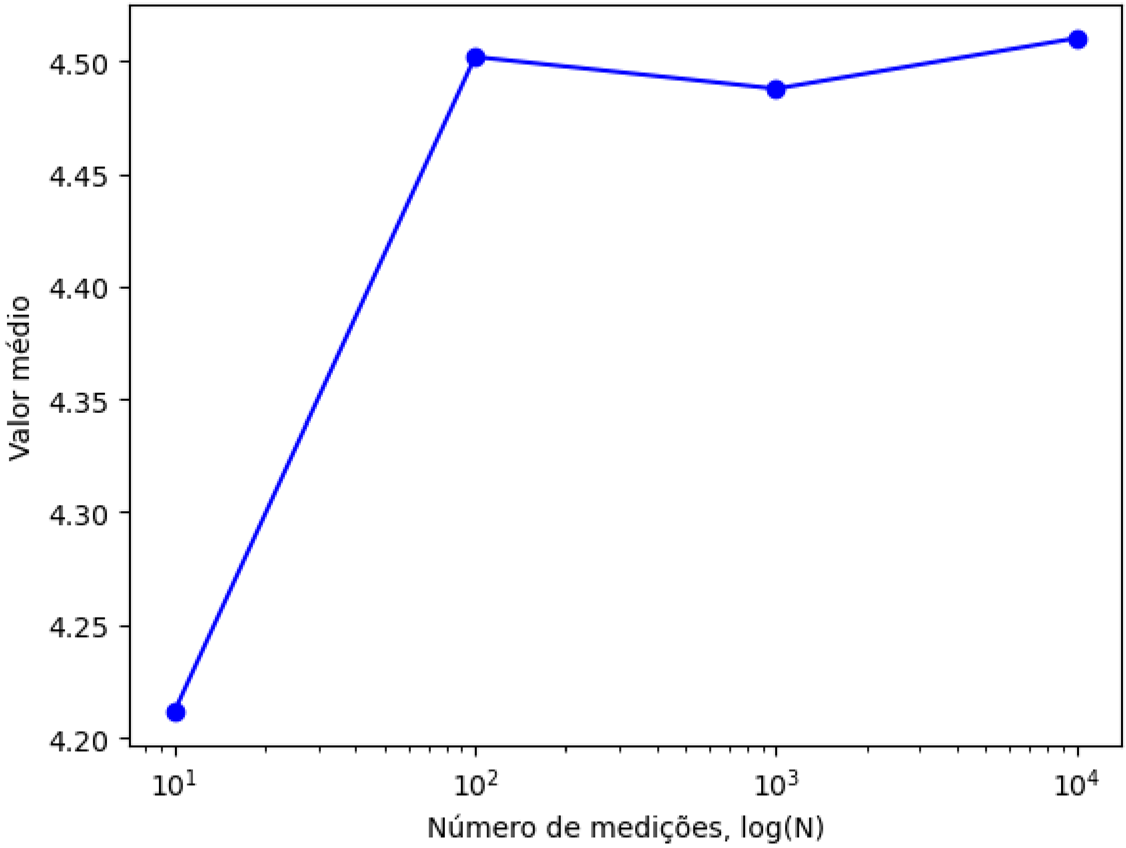
Dica: Pode ser util a função `np.logspace()`

In [52]:
```
# número de medições
#
# Gerar uma sequencia de num=4 numeros inteiros (dtype=int) em
# escala logarítmica_base10, com inicio em 10^start e fim em 10^stop
N = np.logspace(start=1, stop=4, num=4, dtype=int)
print("N = ", N)

Xmedia = np.zeros(4)
Xerro = np.zeros(4)
i = 0
# gerar N valores de X com média 4.5 e desvio padrão 0.5
for N_i in N:
    X = np.random.normal(4.500, 0.500, size = N_i)
    Xmedia[i] = np.mean(X)
    Xerro[i] = np.std(X)/np.sqrt(N_i)
    print("N_i = {:.3f}, Xmedia = {:.3f}, Xerro = {:.3f}".format(N_i, Xmedia[i], Xerr
    i = i + 1
```

```
N =  [   10   100  1000 10000]
N_i = 10.000, Xmedia = 4.212, Xerro = 0.202
N_i = 100.000, Xmedia = 4.502, Xerro = 0.049
N_i = 1000.000, Xmedia = 4.488, Xerro = 0.016
N_i = 10000.000, Xmedia = 4.510, Xerro = 0.005
```

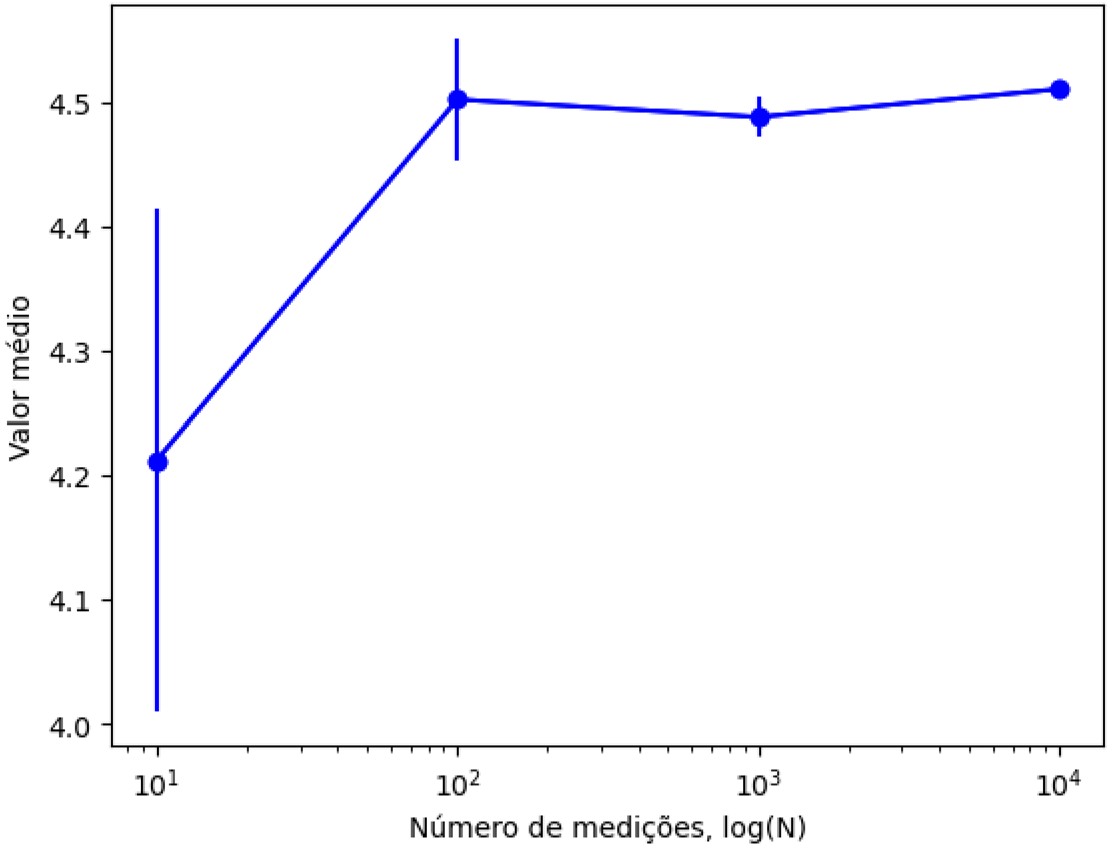
2. Para cada N, calcule a media dos valores gerados, e faça o gráfico das medias em função de N. Use uma escala logaritmica para os valores de N.

In [57]:
```
plt.semilogx(N, Xmedia, 'b-o')

plt.xlabel("Número de medições, log(N)")
plt.ylabel("Valor médio")
plt.show()
```



3. Adicione ao gráfico uma linha que represente o valor da média, e os limites de variação esperados, de acordo com a formula $\sigma/\sqrt{N}$

In [62]:
```
plt.semilogx(N, Xmedia, 'b-o')
plt.errorbar(N, Xmedia, yerr=Xerro, fmt='b-o')
plt.xlabel("Número de medições, log(N)")
plt.ylabel("Valor médio")
plt.show()
```



Pergunta 2:

O que acontece à média $\bar{X}$ à medida que N aumenta? Os valores caem sempre dentro dos limites traçados em 3.? porquê?

O erro calculado a partir do desvio padrão é sempre subestimado.