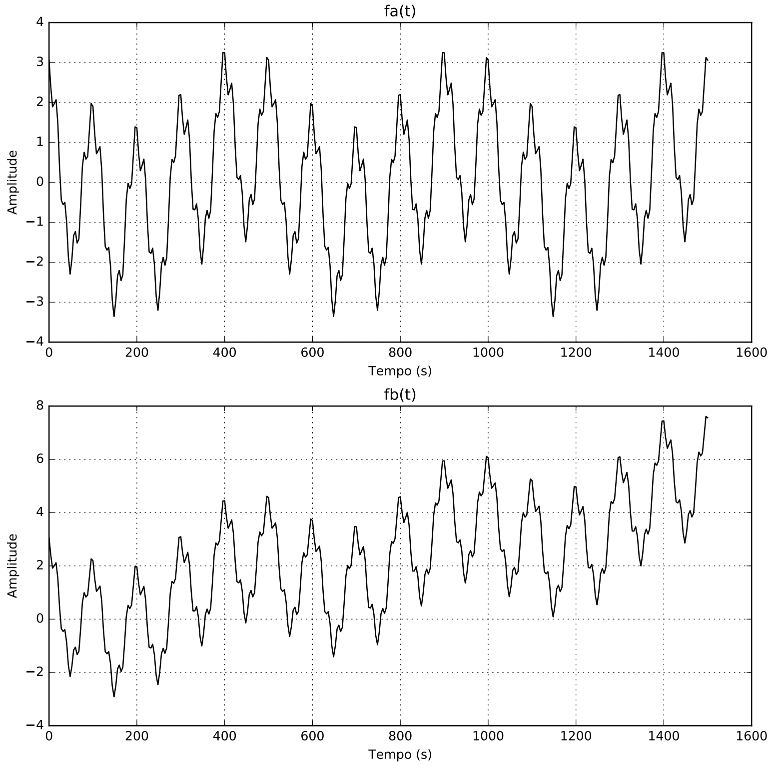
Modelação Numérica (2016/17)
Exame, 1^a Época

7/Junho/2017
L-MOG, L-EG, MI-EEA
DEGGE, FCUL

1 Exercício 1 - Análise digital de séries temporais

Observa o código seguinte, bem como o seu output:

```
#%% Bloco A
import matplotlib.pyplot as plt
import numpy as np
from numpy import pi as pi

#%% Bloco B
plt.rcParams['figure.figsize'] = 10, 10

#%% Bloco C
ti=0.       # segundos
tf=1500.    # segundos
dt=4.       # segundos
t = np.arange(ti, tf+dt, dt)

#%% Bloco D
T1=100.     # segundos
T2=500.     # segundos
fq3=.05     # Hz

#%% Bloco E
f0 = .003*t
f1 = 2. * np.cos(2*pi * t/T1)
f2 = np.cos(2*pi * t/T2 + pi/2)
f3 = .5 * np.cos(2*pi * t*fq3)

#%% Bloco F
fa=f1+f2+f3
fb=f0+f1+f2+f3

#%% Bloco G
plt.close()
plt.subplot(2,1,1)
plt.plot(t, fa, 'black')
plt.grid()
plt.xlabel(u'Tempo (s)')
plt.ylabel(u'Amplitude')
plt.title('fa(t)')

plt.subplot(2,1,2)
plt.plot(t, fb, 'black')
plt.grid()
plt.xlabel(u'Tempo (s)')
plt.ylabel(u'Amplitude')
plt.title('fb(t)')
```



1. **Comenta o código acima, indicando o significado de cada bloco de código.**

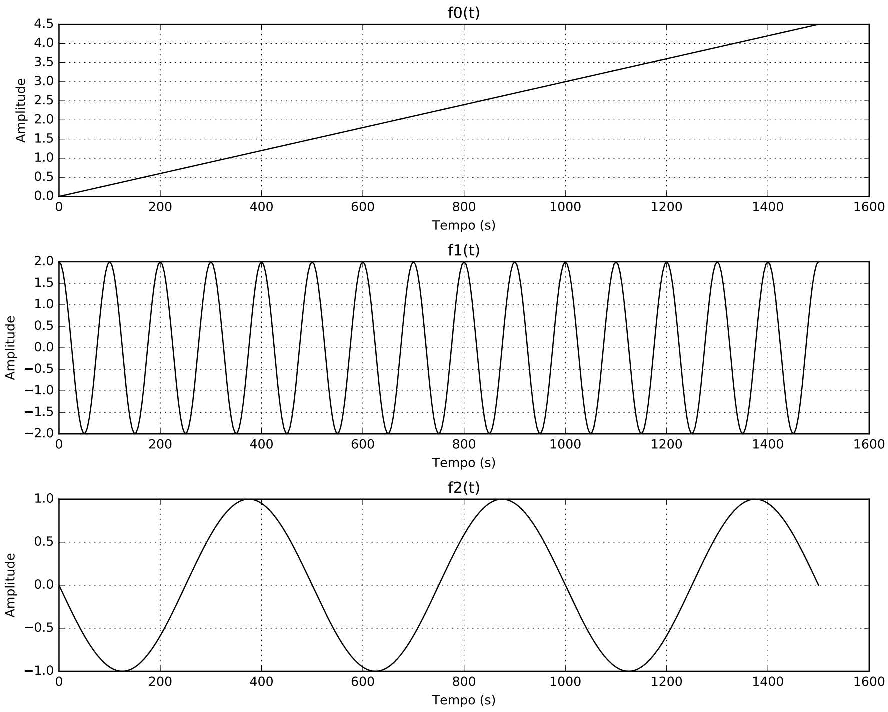
   Bloco A: Importar módulos python para fazer gráficos (matplotlib) e operações numéricas (numpy).
   Bloco B: Definir o tamanho das figuras.
   Bloco C: Definir o vector de tempo.
   Bloco D: Definir os períodos e frequências dos sinais sinusoidais.
   Bloco E: Definir as funções f_0, f_1, f_2 e f_3.
   Bloco F: Definir as funções f_a e f_b a partir das funções f_0, f_1, f_2 e f_3.
   Bloco G: Fazer os gráficos das funções f_a e f_b.

2. **Escreve a expressão analítica da função $f_b(t)$ representada no gráfico.**

$$f_b(t) = 0.003t + 2\cos\left(2\pi\frac{t}{T_1}\right) + \cos\left(2\pi\frac{t}{T_2} + \frac{\pi}{2}\right) + \frac{1}{2}\cos\left(2\pi f q_3 t\right)$$

$$T_1 = 100\,s; T_2 = 500\,s; fq_3 = 0.05\,Hz$$

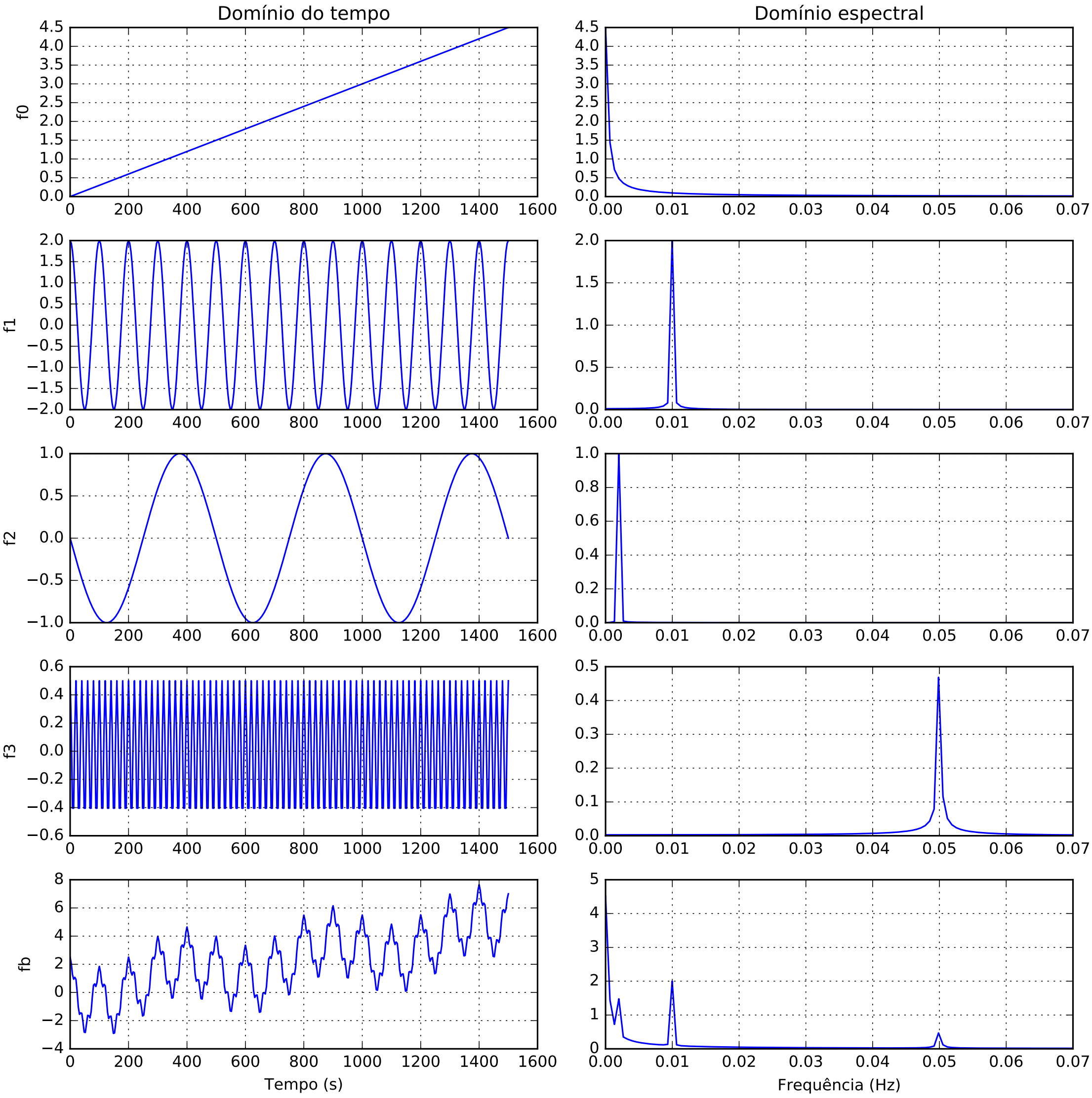
3. **Faz os gráficos das funções $f_0(t)$, $f_1(t)$ e $f_2(t)$ num intervalo de tempo à tua escolha. No caso das funções sinusoidais, representa pelo menos dois ciclos completos.**



4. **Desenha o espectro de amplitudes da função $f_b(t)$. Identifica no espectro a influência das funções $f_0(t)$, $f_1(t)$, $f_2(t)$ e $f_3(t)$.**

   As funções $f1(t)$, $f2(t)$ e $f3(t)$ são sinusoidais, por isso aparecem no espectro representadas pelas suas frequências e amplitudes. A função $f0(t)$ é uma função linear, que representa a tendência da função f_b, e vai causar no espectro um pico nas frequências perto de zero (as tendências decompoem-se como somas de funções sinusoidais, em que as de maior amplitude são as de períodos mais longos, logo frequências mais baixas).

   A figura abaixo mostra o espectro das funções $f_0(t)$, $f_1(t)$, $f_2(t)$ e $f_3(t)$ individualmente, e finalmente o espectro da função $f_b(t)$



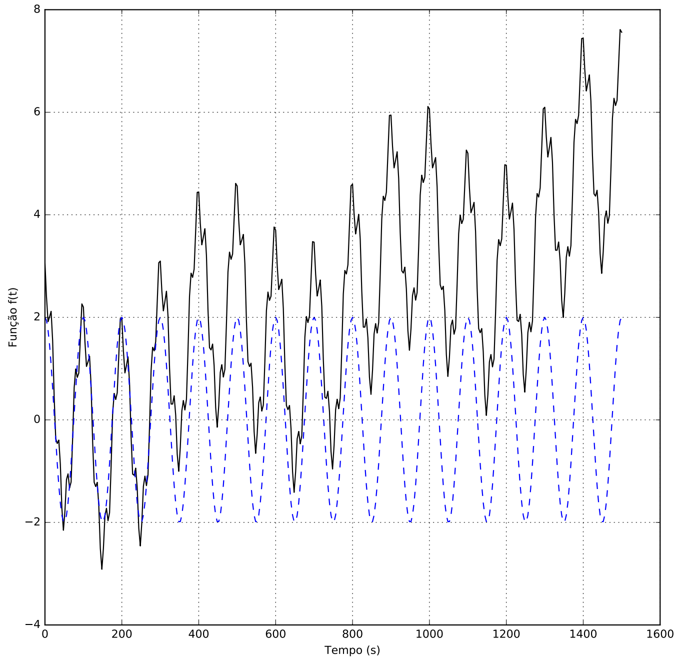
5. **Qual é a frequência de Nyquist deste sinal? Qual o significado da frequência de Nyquist?**

   A frequência de Nyquist é a frequência mais elevada que pode ser bem representada numa série dado um certo intervalo de amostragem no tempo. As séries acima estão amostradas com um intervalo de amostragem $dt = 4\,s$. Como precisamos de pelo menos dois pontos para amostrar bem um período, o período mais curto que conseguimos representar correctamente com um intervalo de amostragem $dt = 4\,s$ é $2\,dt = 8\,s$. A frequência mais alta que conseguimos representar correctamente (frequência de Nyquist) é então $1/(2\,dt) = 1/8\,Hz$.

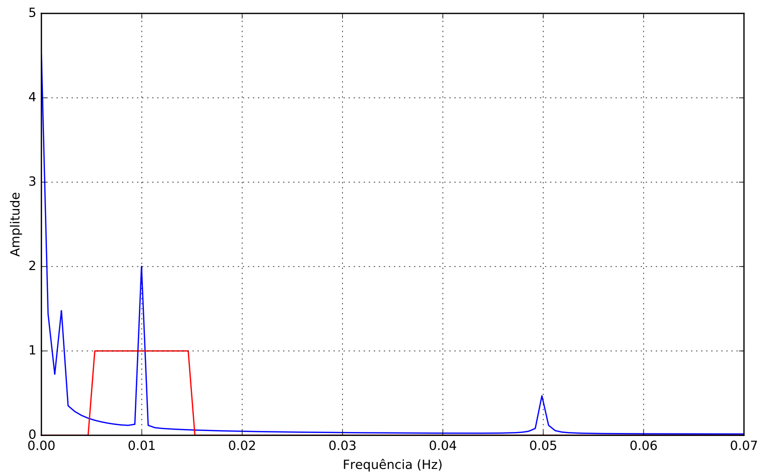
   Também podemos aplicar directamente a fórmula para calcular a frequência de Nyquist:

$$f_{Nyquist} = \frac{f_{amostragem}}{2} = \frac{1}{2\,dt} = \frac{1}{8}\,Hz$$

6. **Observa a figura da página seguinte. Desenha um filtro espectral que pudesses utilizar para obter o sinal tracejado a partir do sinal a cheio ($fb(t)$). Explica como aplicarias o filtro.**



O sinal que queremos reter é a curva sinusoidal com período de 100 s e amplitude igual a 2, ou seja, é a função $f_1(t)$. Um possível filtro espectral é o filtro passa-banda desenhado a vermelho na figura abaixo:



Iríamos aplicá-lo da seguinte forma:

(a) Calcular o espectro (ou seja, a transformada de Fourier) da função: $\mathcal{F}(f_b(t)) = F_b(f)$.

(b) Construir a função de transferência $H(f)$, de modo a deixar passar apenas a frequência desejada ($1/100\,Hz$, ver desenho acima, curva a vermelho) e eliminar as restantes frequências do sinal.

(c) Multiplicar o espectro da função pela função de transferência: $F_b(f) \times H(f)$.

(d) Passar o espectro filtrado de volta para o domínio do tempo, aplicando a transformada inversa de Fourier: $\mathcal{F}^{-1}\left(F_b(f) \times H(f)\right)$.

7. **O que é a convolução entre duas funções? (Podes explicar como preferires: utilizando a expressão matemática, utilizando gráficos, texto, etc.) Dá um exemplo da aplicação da convolução.**

   A convolução é uma operação matemática entre duas funções, que consiste em:

   (a) Inverter no tempo uma das funções.

   (b) Desfasar ponto a ponto as duas funções (a primeira e a inversa no tempo da segunda), e para cada desfasamento:

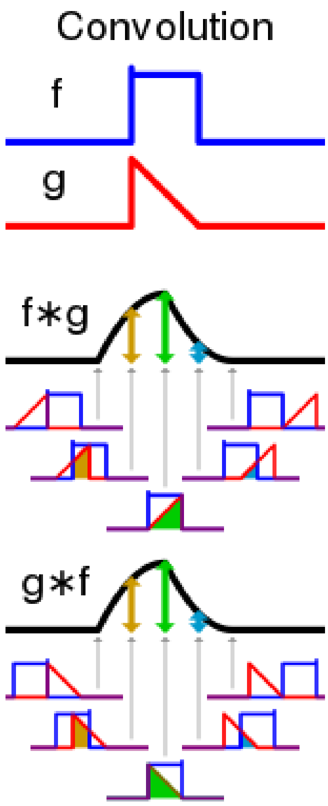
   (c) Multiplicar ponto a ponto as duas funções (sendo que a segunda estava invertida no tempo), e somar os produtos.

   (d) O valor da convolução num dado ponto no tempo, corresponde ao valor obtido multiplicando e somando as duas funções, para um dado desfasamento. A função de convolução é obtida repetindo o processo para os vários desfasamentos possíveis.

   Matematicamente, a convolução de duas funções f e g é dada por:

$$y(t) = conv(f, g) = \sum_{n=-\infty}^{+\infty} f(n)g(t-n)$$

   Um exemplo esquemático da convolução:



   As convoluções podem ser utilizadas para filtrar sinais, como no caso do filtro de média móvel, em que se aplica a convolução com uma função rectangular para filtrar os períodos inferiores ao comprimento do rectângulo.

Uma convolução no domínio temporal corresponde a uma multiplicação do domínio espectral, pelo que os filtros espectrais correspondem a convoluções no domínio do tempo.

2 Exercício 2 - Equações diferenciais parciais

Considera uma barra metálica, com 3 metros de comprimento, que se encontra à temperatura ambiente (15^oC). Uma das extremidades da barra é aquecida a 60^oC e mantida a essa temperatura. A variação da temperatura na barra é dada pela equação da condução do calor:

$$\frac{\partial T}{\partial t} = \kappa \frac{\partial^2 T}{\partial x^2}$$

Onde T é a temperatura (que vai evoluir no tempo e no espaço) e κ é a difusividade térmica do metal (que podemos assumir que é uma constante conhecida).

1. **Discretiza a equação acima, utilizando diferenças avançadas no tempo.**

$$\frac{T_i^{n+1} - T_i^n}{\Delta t} = \kappa \frac{T_{i-1}^n - 2T_i^n + T_{i+1}^n}{\Delta x^2}$$

2. **Como podes calcular a temperatura, para cada ponto ($x = x_i$), no tempo seguinte ($t = t^{n+1}$)? (Isto é, re-escreve a expressão na forma $T_i^{n+1} = ...$). Trata-se de uma esquema explícito ou implícito? Porquê?**

   Manipulando a expressão anterior, obtemos:

$$T_i^{n+1} = T_i^n + \kappa \frac{\Delta t}{\Delta x^2} \left(T_{i-1}^n - 2T_i^n + T_{i+1}^n\right)$$

   Trata-se de um método explícito, porque o ponto que queremos calcular (T_i^{n+1}) depende apenas de pontos atrás no tempo (T^n). Assim sendo, podemos calcular T_i^{n+1} explicitamente em função de pontos atrás no tempo, já conhecidos.

3. **Que tipo de condições fronteira utilizarias em cada extremo da barra para resolver este problema? Como as implementarias?**

   No extremo que é mantido a 60^oC, temos de aplicar uma condição fronteira que fixe a temperatura a 60^oC: T[0]=60.

   No extremo oposto, a barra vai aquecer ao longo do tempo, sem qualquer limitação. Assim sendo, aplica-se uma fronteira aberta: T[nx-1]=T[nx-2], onde nx é o número de pontos ao longo da barra (em x, no espaço).

4. **Esquematiza o essencial do código (pseudo-código) que utilizarias para modelar a evolução da temperatura ao longo da barra. Nota: Não é necessário indicar as linhas de código em detalhe, basta identificar os blocos de código que seriam necessários e para que servem. Podes usar linhas de código, expressões matemáticas ou esquemas que te ajudem a explicar o algoritmo.**

   (a) Dicretizar o domínio espacial: Construir um vector x, que vai de 0 a 3 m, em passos de dx. O vector terá nx pontos.

(b) Construir um vector T para guardar os valores de temperatura ao longo da barra. O vector T tem tantos pontos quanto o domínio espacial (nx), e vai ser actualizado à medida que evoluímos o problema no tempo.

(c) Inicializar o vector T. Como a barra se encontra inicialmente a 15^oC, inicialmente todos os pontos se encontram a 15^oC, excepto a extremidade que é aquecida a 60^oC: T[1:n-1]=15; T[0]=60.

(d) Evoluir o problema no tempo: Fazer um ciclo em que se avança no tempo passo a passo, e para cada passo no tempo:

   i. Para cada ponto da barra (em x), calcular a temperatura no próximo passo no tempo, utilizando a equação discretizada acima.

   ii. Aplicar as condições fronteira.

   iii. Actualizar a temperatura ao longo da barra com os novos valores calculados, ou seja, avançar no tempo a distribuição de temperaturas na barra.

   iv. Se/quando desejado, fazer a representação gráfica dos resultados.

3 Exercício 3 - Optimização de parâmetros

A trajectória de um projéctil no plano vertical segue uma parábola descrita por:

$$z = z_0 + v_{z0}t - \frac{1}{2}gt^2$$

Um radar fez N medições da posição vertical do projéctil (z). Queremos saber qual a posição e velocidade iniciais na vertical, bem como a aceleração da gravidade no local (z_0, v_{z0}, g).

1. **Identifica as observações feitas e escreve-as na forma de um vector $\vec{o}$. Faz o mesmo para os parâmetros a inverter/optimizar e escreve o vector $\vec{p}$.**

   As observações são as várias posições do projéctil na vertical $(z_1, z_2, ..., z_N)$ medidas em cada tempo $(t_1, t_2, ..., t_N)$:

$$\vec{o} = \begin{bmatrix} o_1 \\ o_2 \\ o_3 \\ ... \\ o_N \end{bmatrix} = \begin{bmatrix} z_1 \\ z_2 \\ z_3 \\ ... \\ z_N \end{bmatrix}$$

   Os parâmetros a optimizar são:

$$\vec{p} = \begin{bmatrix} p_1 \\ p_2 \\ p_3 \end{bmatrix} = \begin{bmatrix} z_0 \\ v_{z0} \\ g \end{bmatrix}$$

2. **Qual o número mínimo de observações de que precisas para conseguir constranger todos os parâmetros?**

   Como temos 3 parâmetros a optimizar, precisamos no mínimo de 3 observações independentes para os conseguir determinar.

3. **Podes resolver este problema utilizando um algoritmo linear? Se sim, escreve as equações matriciais do problema a resolver, na forma $\vec{o} = G\vec{p}$. Assume que $N = 4$.**

   Sim, a relação entre as observações e os parâmetros é linear, por isso podemos resolver o problema utilizando um algoritmo de inversão linear.

$$\begin{bmatrix} o_1 \\ o_2 \\ o_3 \\ o_4 \end{bmatrix} = \begin{bmatrix} G_{11} & G_{12} & G_{13} \\ G_{21} & G_{22} & G_{23} \\ G_{31} & G_{32} & G_{33} \\ G_{41} & G_{42} & G_{43} \end{bmatrix} \begin{bmatrix} p_1 \\ p_2 \\ p_3 \end{bmatrix}$$

$$\begin{bmatrix} z_1 \\ z_2 \\ z_3 \\ z_4 \end{bmatrix} = \begin{bmatrix} 1 & t_1 & -\frac{1}{2}t_1^2 \\ 1 & t_2 & -\frac{1}{2}t_2^2 \\ 1 & t_3 & -\frac{1}{2}t_3^2 \\ 1 & t_4 & -\frac{1}{2}t_4^2 \end{bmatrix} \begin{bmatrix} z_0 \\ v_{z0} \\ g \end{bmatrix} = \begin{bmatrix} z_0 + v_{z0}t_1 - \frac{1}{2}gt_1^2 \\ z_0 + v_{z0}t_2 - \frac{1}{2}gt_2^2 \\ z_0 + v_{z0}t_3 - \frac{1}{2}gt_3^2 \\ z_0 + v_{z0}t_4 - \frac{1}{2}gt_4^2 \end{bmatrix}$$

4. **Como farias para determinar z_0, v_{z0} e g?**

   Para determinar os parâmetros que constituem o vector $\vec{p}$, basta inverter a matriz G e multiplicá-la pelas observações $\vec{o}$:

$$\vec{o} = G\vec{p} \;\Leftrightarrow\; G^{-1}\vec{o} = G^{-1}G\vec{p} \;\Leftrightarrow\; \vec{p} = G^{-1}\vec{o}$$

   Neste caso, como a matriz G não é quadrada, não podemos invertê-la directamente. Teremos de calcular a sua inversa generalizada/pseudo-inversa.

5. **Seria possível resolver este problema utilizando um algoritmo não-linear? Se sim, qual seria a função de custo que utilizarias? Para que serve a função de custo nos algoritmos não-lineares?**

   Sim, os algoritmos não lineares podem ser utilizados para resolver tanto problemas de inversão lineares como não lineares.

   Uma possível função de custo (norma L1), C, seria:

$$\begin{aligned} C &= \sum_{i=0}^{N} |obs_i - mod_i| \\ &= \sum_{i=0}^{N} |z_i - (z_0 + v_{z0}t_i - \frac{1}{2}gt_i^2)| \end{aligned} \tag{1}$$

   A função de custo serve para avaliar a diferença entre as observações e as previsões feitas com base num dado conjunto de parâmetros, e assim guiar o algoritmo de inversão para parâmetros que gerem previsões o mais próximas possível das observações. O objectivo dos algoritmos de inversão é encontrar o mínimo da função de custo.

6. **Supõe agora que o projéctil é lançado não só com uma certa velocidade inicial na vertical, mas também com uma certa velocidade inicial na horizontal, na direcção x. Neste caso, o movimento do projéctil no plano (x, z) é dado por:**

$$z = z_0 + v_{z0}t - \frac{1}{2}gt^2$$
$$x = x_0 + v_{x0}t$$

Se o radar fizer 3 medições da posição (x, z) do projéctil, como podes encontrar as posições e velocidades inicias no plano e a aceleração da gravidade no local (x_0, z_0, v_{x0}, v_{z0}, g)? Nota: Existe um único vector de observações, que contém todas as observações feitas, e um único vector de parâmetros, que contém todos os parâmetros a determinar.

Utilizando um algoritmo linear, poderíamos resolver o problema da mesma forma descrita nas alíneas 3 e 4. Agora, as equações a resolver seriam:

$$\vec{o} = G\vec{p}$$

$$\begin{bmatrix} z_1 \\ z_2 \\ z_3 \\ x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & t_1 & -\frac{1}{2}t_1^2 \\ 0 & 1 & 0 & t_2 & -\frac{1}{2}t_2^2 \\ 0 & 1 & 0 & t_3 & -\frac{1}{2}t_3^2 \\ 1 & 0 & t_1 & 0 & 0 \\ 1 & 0 & t_2 & 0 & 0 \\ 1 & 0 & t_3 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_0 \\ z_0 \\ v_{x0} \\ v_{z0} \\ g \end{bmatrix} = \begin{bmatrix} z_0 + v_{z0}t_1 - \frac{1}{2}gt_1^2 \\ z_0 + v_{z0}t_2 - \frac{1}{2}gt_2^2 \\ z_0 + v_{z0}t_3 - \frac{1}{2}gt_3^2 \\ x_0 + v_{x0}t_1 \\ x_0 + v_{x0}t_2 \\ x_0 + v_{x0}t_3 \end{bmatrix}$$

O problema poderia também ser resolvido utilizando um algoritmo não linear. Neste caso, uma possível função de custo, C, seria:

$$\begin{aligned} C &= \sum_{i=0}^{N} |obs_i - mod_i| \\ &= \sum_{i=0}^{N} \left(|z_i - (z_0 + v_{z0}t_i - \frac{1}{2}gt_i^2)| + |x_i - (x_0 + v_{x0}t_i)| \right) \end{aligned} \tag{2}$$

Um possível algoritmo não linear:

(a) Começar por atribuir valores aleatórios aos parâmetros a determinar. Calcular as posições z e x previstas com base nos valores aleatórios, e avaliar a diferença entre as observações e as previsões. Ou seja, calcular a função de custo para os valores iniciais aleatórios.

(b) Re-calcular o valor da função de custo para vários valores aleatórios dos parâmetros, dentro do espaço de parâmetros a explorar. Começar com variações aleatórias muito grandes (dentro do domínio a explorar).

(c) Sempre que encontramos um valor da função de custo inferior ao valor mais baixo encontrado até então, guardar o conjunto de parâmetros que os gerou, e continuar a procura à volta dos melhores parâmetros.

(d) Continuar a experimentar valores aleatórios dos parâmetros, diminuindo o espaço de busca à volta dos melhores parâmetros encontrados até ao momento.