

```

1 import numpy as np
2 import matplotlib.pyplot as plt

4 t = np.array([0,10,20,30,40,50]) # minutes
5 H0 = np.array([10,6.5,4.9,3.9,3.2,2.8])*1e-3 # mol.L-1

7 ### 2. Méthode différentielle
8 v = [-(H0[i+1]-H0[i-1])/(t[i+1]-t[i-1]) for i in range(1,len(t)-1)] #
    liste définie en compréhension

10 x = np.log(H0)[1:len(t)-1] # on enlève les deux éléments extrêmes pour
    lesquels la vitesse ne peut pas être calculée
11 y = np.log(v)

13 plt.figure()
14 plt.plot(x,y,'+')
15 plt.xlabel('$\\log([H_0^-])$ (concentration en $\\mathrm{mol.L}^{-1}$)')
16 plt.ylabel('$\\log(v)$ (vitesse en $\\mathrm{mol.L}^{-1}\\mathrm{s}^{-1}$)')

18 a,b=np.polyfit(x,y,1)
19 plt.plot(x,a*x+b) # droite de régression

21 print("estimation de l'ordre : p =", a)

23 ### 3. Méthode intégrale
24 y2=1/H0-1/H0[0]

26 plt.figure()
27 plt.plot(t,y2,'+')
28 plt.ylabel('$\\frac{1}{[H_0^-](t)} - \\frac{1}{[H_0^-](0)}$ ($\\mathrm{L.mol}^{-1}$)')
29 plt.xlabel('$t$ (minutes)')

31 k=np.mean(y2[1:]/t[1:]) # on enlève t[0] pour ne pas diviser par 0

33 plt.plot(t,k*t) # droite de régression

35 print("constante de vitesse : k =",k, "L.mol-1.min-1")

```