



TP Python 7 : Représentation graphique de fonctions



Problèmes d'entraînement

Correction

Entrée[2]:

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

Exercice 1 :

Construire une fonction permettant de représenter le graphe de la fonction f_N d'expression

$$f_N(x) = \sum_{k=0}^N \frac{\cos(3x)}{2^k}.$$

Entrée[7]:

```
1 def somme_fonc(N):
2     # On choisit l'intervalle de notre choix : puisqu'on travaille
3     # privilégions un intervalle centré en 0
4     X = np.linspace(-3*np.pi,3*np.pi,100)
5     Y = []
6
7     # Pour chaque valeur de x dans la liste X, on calcule l'image
8     # On parcourt la liste X
9     for i in range(len(X)):
10        # On fixe la valeur de l'antécédent
11        x = X[i]
12        # On calcule la somme
13        s = np.cos(3*x)
14        for k in range(N+1):
15            s = s + np.cos(3*x)/(2**k)
16
17        # On a terminé de calculer s=f_N(x), on l'ajoute donc à
18        Y.append(s)
19
20    # On a bouclé sur tous les antécédents.
21    # On plot.
22    plt.clf()
23    plt.plot(X,Y)
24    plt.xlabel('x')
25    plt.ylabel('y')
26    # La fonction title ne marche pas comme la fonction print, on
27    # la variable entière (int) N en une chaîne de caractère (str)
28    plt.title('Graphe de f_'+str(N))
29    plt.show()
30
31    # La fonction somme_fonc ne renvoie rien!
32    return
33
34 # Tests. Les graphes sont similaires, mais les pics ne sont pas
35 somme_fonc(1)
36 somme_fonc(5)
```

Exercice 2 [Cardioïde]

Construire la courbe paramétrée d'équation :

$$\begin{cases} x = \cos(t)(1 - \cos t) \\ y = \sin(t)(1 - \cos t), \end{cases} \quad \text{pour } t \in [0, 2\pi].$$

Entrée[11]:

```
1 T = np.linspace(0,2*np.pi,1000)
2 X = []
3 Y = []
4
5 for k in range(len(T)):
6     X.append(np.cos(T[k])*(1-np.cos(T[k])))
7     Y.append(np.sin(T[k])*(1-np.cos(T[k])))
8
9 plt.clf()
10 plt.plot(X,Y, 'pink')
11 plt.xlabel('cos(t)(1-cos(t)) pour t dans [0,2pi]')
12 plt.ylabel('sin(t)(1-cos(t)) pour t dans [0,2pi]')
13 plt.title('Une cardioïde')
14 plt.show()
```

Exercice 3 [Potentiel de Lennard-Jones]

Le potentiel de Lennard-Jones est une énergie représentant l'interaction de deux atomes dans un gaz parfait monoatomique. On note r la distance entre deux atomes. Le potentiel est donné par :

$$\forall r \in \mathcal{D}_V, \quad V(r) := 4\varepsilon \left[\left(\frac{\sigma}{r} \right)^{12} - \left(\frac{\sigma}{r} \right)^6 \right],$$

où $\varepsilon > 0$ est une énergie de référence, et σ est une distance de référence, appelée \emph{distance d'équilibre}.

1. **Étude préliminaire** → À faire sur une copie.

- Quelle est la variable de la fonction V ? Quels sont le ou les paramètres de la fonction V ?
- Donner le domaine de définition \mathcal{D}_V de la fonction V .
- Pour quelle(s) valeur(s) de r la fonction V s'annule-t-elle?
- Établir le tableau de variations de V .
- La fonction V admet-elle un minimum? Un maximum? Si oui, donner la valeur maximale et la valeur minimale de V .
- Quelles sont les limites de V en 0 et $+\infty$? Donner une interprétation physique de ces limites.

2. **Python**

- Écrire une fonction `V_LJ` qui prend en argument les paramètres ε et σ ainsi que la variable r , et renvoie $V(r)$.
- Tracer le graphe de la fonction V pour un gaz parfait d'Argon, où $\varepsilon = 1,66 \times 10^{-21}$ J et $\sigma = 3.405$ Å.

→ On se placera sur l'intervalle $[2.7, 10.2]$ et on jouera sur les limites de l'axe des ordonnées à l'aide de la commande non exigible `plt.ylim(-6e-21, 6e21)` pour pouvoir observer les variations de V .

Étude préliminaire

- La variable de V est r , les paramètres (quantités fixées) sont ε et σ .
- Une distance est toujours positive, et r ne peut être nulle. Donc $\mathcal{D}_V = \mathbb{R}_+^*$
- On résoud l'équation suivante, pour tout $r > 0$:

$$\begin{aligned}
V_{LJ}(r) = 0 &\Leftrightarrow 4\varepsilon \left[\left(\frac{\sigma}{r}\right)^{12} - \left(\frac{\sigma}{r}\right)^6 \right] = 0 \\
&\Leftrightarrow \left(\frac{\sigma}{r}\right)^{12} - \left(\frac{\sigma}{r}\right)^6 = 0 \\
&\Leftrightarrow \left(\frac{\sigma}{r}\right)^6 \left[\left(\frac{\sigma}{r}\right)^6 - 1 \right] = 0 \\
&\Leftrightarrow \left(\frac{\sigma}{r}\right)^6 - 1 = 0 \quad \text{car } \left(\frac{\sigma}{r}\right)^6 \neq 0 \\
&\Leftrightarrow r = \sigma
\end{aligned}$$

La fonction V s'annule en $r = \sigma$.

4. Afin d'obtenir le tableau de variations de la fonction V , nous devons la dériver. C'est possible car elle est la composée de fonctions polynômiales et de la fonction inverse, toutes dérivables sur $]0, +\infty[$. On obtient, pour tout $r > 0$:

$$V'(r) = 4\varepsilon \left(6 \frac{\sigma^6}{r^7} - 12 \frac{\sigma^{12}}{r^{13}} \right),$$

de sorte que, pour tout $r > 0$,

$$\begin{aligned}
V'(r) = 0 &\Leftrightarrow \left(6 \times \frac{\sigma^6}{r^7} - 12 \times \frac{\sigma^{12}}{r^{13}} \right) = 0 \quad \text{car } \varepsilon \neq 0 \\
&\Leftrightarrow \left(1 - 2 \times \frac{\sigma^6}{r^6} \right) = 0 \quad \text{car } 6 \times \frac{\sigma^6}{r^7} \neq 0 \\
&\Leftrightarrow r^6 = 2\sigma^6 \\
&\Leftrightarrow r = 2^{1/6} \sigma := r^*.
\end{aligned}$$

La fonction V atteint un extremum en $r^* = 2^{1/6} \sigma$, et il vient

$$V(r^*) = 4\varepsilon \left(\frac{\sigma^{12}}{2^{12/6} \sigma^{12}} - \frac{\sigma^6}{2^{6/6} \sigma^6} \right) = 4\varepsilon \left(\frac{1}{4} - \frac{1}{2} \right) = -\varepsilon.$$

En observant que $\lim_{r \rightarrow 0^+} V(r) = +\infty$, on peut enfin dresser le tableau de variation de V .

5. La fonction V admet un minimum, atteint en r^* . La valeur minimale de V vaut $-\varepsilon$ (en J). Elle n'admet pas de maximum.
6. Lorsque la distance interatomique est proche de zéro, l'énergie V_{LJ} tends vers $+\infty$: quand deux atomes sont très proches l'un de l'autre, les forces de répulsions se font de plus en plus grandes. Il n'est pas possible d'avoir deux atomes qui se superposent! Lorsque la distance interatomique devient très grande, les deux atomes ne se voient presque plus, et l'énergie d'interaction devient alors quasi nulle.

Entrée[56]:

```
1 def V_LJ(r,eps,sig):
2
3     if r <=0 :
4         print('ERROR : une distance est positive, et deux atomes
5             return
6
7         frac = (sig/r)**6
8         diff = frac**2 - frac
9         y = 4*eps*diff
10
11     return y
12
13 e = 1.66e-21
14 s = 3.405
15 R = np.linspace(2.7,10.2,1000)
16 Y = []
17
18 for k in range(len(R)):
19     Y.append(V_LJ(R[k],e,s))
20
21 plt.clf()
22 plt.plot(R,Y,'g')
23 plt.ylim([-6e-21,6e-21])
24 plt.xlabel('Distance interatomique')
25 plt.ylabel('Énergie')
26 plt.title('Énergie')
27 plt.show()
```