

TD centrale 2 analyse et géométrie

I Fiche AN et PLOT

I.1 Résolution d'équations

Exercice 1 Soit $f : x \mapsto x^3 - 3x - 1$

1. Représenter graphiquement f sur l'intervalle $[-3, 3]$.
2. Donner des valeurs approchées des solutions de l'équation $f(x) = 0$.

Exercice 2 Utilisation de la fonction `root`

1. Donner une valeur approchée d'une solution du système $\begin{cases} x^2 + y^2 = 5 \\ x^4 + y^4 = 20 \end{cases}$
2. Déterminer une matrice U orthogonale et une matrice S symétrique vérifiant $SU = \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix}$.

Solution de l'exercice:

1:

```
import scipy.optimize as resol
def f(v) :
    return v[0]**2 + v[1]**2 - 5, v[0]**4 + v[1]**4 - 20
sol = resol.root(f, [1,1])# un False dans le resultat indique que
# le point de depart [0,0] de l'algorithme n'est pas bon
sol = resol.root(f, [1,1])
print(sol.x)
print(f(sol.x)) # verification fait presque (0,0)
```

I.2 Intégration, équations différentielles, séries entières, graphiques

Exercice 3 (centrale 2 extrait): On pose $f_p(x) = \int_0^1 \sqrt{x^2 + t^p} dt$.

1. Représenter graphiquement f_2 .
2. Représenter graphiquement les fonctions f_i sur un même graphique pour $i \in [1, 5]$.

Solution de l'exercice:

```
#Q1
def g(x):
    def f(t):
        return np.sqrt(x**2+t**2)
    return integr.quad(f,0,1)
X = np.arange(0,2, 0.01)
Y=[g(x) for x in X]
plt.plot(X,Y)
plt.show()

#Q2
X = np.arange(0,2, 0.01)
for i in range(1,6):
    def g(x):
        def f(t):
            return np.sqrt(x**2+t**i)
        return integr.quad(f,0,1)
    Y=[g(x) for x in X]
    plt.plot(X,Y)
plt.show()
```

Exercice 4 Tracer la courbe représentant la solution de $\begin{cases} y' + e^x y = 1 \\ y(0) = 1 \end{cases}$ sur l'intervalle $[0, 1]$.

Solution de l'exercice:

```
import math
def f(x, t) :
    return 1-math.exp(t)*x
T = np.arange(0, 1.01, 0.01)# condition de Cauchy en T[0]=0
X = integr.odeint(f, 1, T)# deuxieme argument 1: valeur de la solution en 0
plt.plot(T,X)
plt.show()
```

Exercice 5 (centrale 2 extrait): On considère l'équation $(1 - x)y'' = y : (\mathcal{E})$ sur $]-\infty, 1[$.

- (maths) Montrer qu'il existe une et une seule solution f de cette équation vérifiant $f(0) = 0$ et $f'(0) = 1$.
- (python) Représenter la fonction f sur l'intervalle $[-2; 0, 95]$.

3. Soit (a_n) la suite définie par $a_0 = 0, a_1 = 1$ et $\forall n \in \mathbb{N} \setminus \{0, 1\}, a_n = \frac{n-2}{n}a_{n-1} + \frac{1}{n(n-1)}a_{n-2}$.

- (python) Représenter graphiquement a_n pour $n \in [0, 100]$.
- Montrer que le rayon de convergence de $\sum a_n x^n$ est supérieur ou égal à 1.
- (python) Représenter graphiquement la fonction $x \mapsto \sum_{n=0}^{100} a_n x^n$ sur $[-2; 0, 95]$. Que constate-on?
- Démontrer le résultat.

Solution de l'exercice:

On a $(E) = (1 - x)y'' = y$ sur $]1, +\infty[\Leftrightarrow y'' - \frac{1}{(1-x)}y = 0$. D'après le théorème de Cauchy, il existe une et une seule solution f de cette équation vérifiant $f(0) = 0$ et $f'(0) = 1$.

```
""" Q2: representation de la solution f"""
def f(X,t): # fonction de l'equa diff X'=f(X,t)
    return [X[1],1/(1-t)*X[0]]
from scipy.integrate import odeint
import numpy as np
import matplotlib.pyplot as plt
# sur [0,0.95]
liste_t=np.linspace(0,0.95,100)
sol=odeint(f,[0,1],liste_t) # X[0]=[0,1]
plt.plot(liste_t,sol[:,0]) # sol array 100,2: premiere col: valeurs de y, deuxieme: valeurs de y'
# sur [-2,0]
liste_t=np.linspace(0,-2,100) # liste des temps de 0 a -2 (temps parcouru en sens inverse)
sol=odeint(f,[0,1],liste_t) # X[0]=[0,1]
plt.plot(liste_t,sol[:,0])
plt.show()
```

```
"""Q3a: representation des an"""
liste_n=[n for n in range(101)]
liste_a=[0,1]
a,b=0,1
for n in range(2,101):
    c=(n-2)/n*b+a/(n*(n-1))
    liste_a.append(c)
    a,b=b,c
plt.plot(liste_n,liste_a) # on constate 0<=an<=1
```

3b: Montrons par récurrence sur n que $0 \leq a_n \leq 1$. C'est vrai pour $n = 0$ et 1 .

Soit $n \geq 2$. Supposons $0 \leq a_{n-2} \leq 1$ et $0 \leq a_{n-1} \leq 1$.

On a $a_n = \frac{n-2}{n}a_{n-1} + \frac{1}{n(n-1)}a_{n-2} \leq \frac{n-2}{n} + \frac{1}{n(n-1)} \leq \frac{n-2}{n} + \frac{1}{n} < 1$ car $n-1 \geq 1$ et $a_n \geq 0$ donc $0 \leq a_n \leq 1$.

Posons $b_n = 1$. La série entière $\sum b_n x^n$ est de rayon 1 et $0 \leq a_n \leq b_n$ donc le le rayon de convergence de $\sum a_n x^n$ est supérieur ou égal à 1.

3c:

```
""" representation de S100(x)
def S100(x):
    S,p=0,1
    for i in range(101):
        S=S+liste_a[i]*p
        p=p*x
    return S
liste_S=[S100(t) for t in liste_t]
plt.plot(liste_t,liste_S)
plt.plot(liste_t,sol[:,0])
```

3d: Montrons que $S = f$.

Pour tout $n \geq 0$, $n(n-1)a_n - (n-2)(n-1)a_{n-1} + a_{n-2} = 0$ donc si $|x| < 1$,

$\sum_{n=2}^{+\infty} n(n-1)a_n x^{n-2} - \sum_{n=2}^{+\infty} (n-2)(n-1)a_{n-1} x^{n-2} + \sum_{n=2}^{+\infty} a_{n-2} x^{n-2} = 0$ (toutes les séries ont même rayon que $\sum a_n x^n$).

Or $\sum_{n=2}^{+\infty} n(n-1)a_n x^{n-1} = x \sum_{n=2}^{+\infty} n(n-1)a_n x^{n-2} = xS''(x)$; et $\sum_{n=2}^{+\infty} (n-2)(n-1)a_{n-1} x^{n-2} = x \sum_{n=2}^{+\infty} (n-2)(n-1)a_{n-1} x^{n-3} =$

$x \sum_{n=3}^{+\infty} n(n-1)a_n x^{n-2} = xS''(x) - a_2 = xS''(x)$ car $a_2 = 0$.

et $\sum_{n=2}^{+\infty} a_{n-2} x^{n-2} = S(x)$ donc $(1-x)S'' = S$ donc S vérifie l'équation (E) et

comme $S(0) = a_0 = 0$ et $S'(0) = a_1 = 1$, on déduit avec le théorème de Cauchy que $S = f$.

II Tracé de courbes et surfaces

Exercice 6 (centrale 2 extrait): Soit $M \in \mathcal{M}_n(\mathbb{R})$. On définit $\varphi_M : \mathbb{R}^n \rightarrow \mathbb{R}$ définie par $\varphi_M(X) = {}^t X M X$.

On suppose que $M = \begin{pmatrix} -2 & 4 \\ -1 & 2 \end{pmatrix}$ et $X = \begin{pmatrix} x \\ y \end{pmatrix}$. Tracer la surface d'équation $z = \varphi_M(x, y)$ pour $(x, y) \in [-2, 2]^2$.

Solution de l'exercice:

On a $\varphi_M(x, y) = {}^t X M X = (x, y) \begin{pmatrix} -2x + 4y \\ -x + 2y \end{pmatrix} = x(-2x + 4y) + y(-x + 2y) = (2x + y)(-x + 2y) = -2x^2 +$

$3xy + 2y^2$.

```
import matplotlib.pyplot as plt
import numpy as np
from mpl_toolkits.mplot3d import Axes3D
ax=Axes3D(plt.figure())
def f(x,y):
    return (2*x+y)*(-x+2*y)
X=np.arange(-2,2,0.02)
Y=np.arange(-2,2,0.02)
X,Y=np.meshgrid(X,Y)
Z=f(X,Y)
ax.plot_surface(X,Y,Z)
plt.show()
# On a eu un probleme mardi avec ce script (a verifier)
```

Exercice 7 Tracer la courbe paramétrée plane définie par $\begin{cases} x(t) = \cos(t) \\ y(t) = \sin(2t) \end{cases}$

Exercice 8 Tracer la courbe du plan d'équation $x^2 + y^4 = \lambda$ pour $\lambda \in \{1, 4, 10\}$.

Solution de l'exercice:

```
def f(x,y) :
    return x**2 + y**4
X = np.arange(-10, 10, 0.01)
Y = np.arange(-10, 10, 0.01)
X, Y = np.meshgrid(X, Y)
Z = f(X, Y)
plt.axis('equal')
plt.contour(X, Y, Z, [1,4,10])
```

III Enoncé complet

Exercice 9 On pose, pour tout $n \in \mathbb{N}^*$, $A_n : x \mapsto \sum_{k=1}^n \frac{x^k}{k}$.

1. Montrer que pour tout $y \in \mathbb{R}^+$, il existe un unique $x \in \mathbb{R}^+$ tel que $A_n(x) = y$. On note $f_n(y)$ cette unique solution.
2. Ecrire une fonction Python qui prend en argument n et x et qui renvoie $A_n(x)$.
3. Utiliser la fonction `fsolve` pour obtenir $f_n(y)$.
4. Tracer, pour différentes valeurs de y , les valeurs $(f_n(y))_{n \leq 100}$.
5. Montrer que la suite $(f_n(y))_n$ converge.
6. Déterminer la limite de la suite $(f_n(y))$.

Solution de l'exercice:

1 La fonction A_n est un polynôme. Elle est donc continue sur \mathbb{R}^+ , et dérivable sur \mathbb{R}^+ :

$$\forall x \geq 0, A'_n(x) = \sum_{k=1}^n x^{k-1} = \sum_{k=0}^{n-1} x^k \geq 0$$

Donc A_n est strictement croissante sur \mathbb{R}^+ .

Comme de plus $A_n(0) = 0$ et $A_n(x) \xrightarrow{x \rightarrow +\infty} +\infty$, il vient que A_n réalise une bijection de \mathbb{R}^+ dans \mathbb{R}^+ .

Et donc pour tout $y \in \mathbb{R}^+$ il existe un unique $x \in \mathbb{R}^+$ tel que $A_n(x) = y$.

2

```
def A(n,x):
    s=0
    puiss =1$.
    for k in range ( 1,n+1 ):
        puiss *= x
        s += puiss / k
    return s
```

3

```
from scipy.optimize import fsolve
def f(n,y):
    return fsolve( lambda x : A(n,x)-y , 1)[0]
```

Remarque : j'avais dans un premier temps pris 0 comme second paramètre pour l'utilisation de `fsolve` et j'avais constaté un problème j'ai changé le point de départ de l'algorithme de résolution en 1, avec un résultat probant.

4

```

import matplotlib.pyplot as plt
Y = [1,2,3,4]
couleurs = ['red','green','blue','cyan']
les_n = [n for n in range(1, 101)]
for k in range(len(Y)):
    y = Y[k]
    coul = couleurs[k]
    les_f = [f(n,y) for n in les_n]
    plt.plot(les_n,les_f,'o',color=coul,label='y'+str(y))
plt.legend()
plt.show()

```

5 On dirait bien que la suite $(f_n(y))_n$ est décroissante et convergente vers une limite $\ell(y)$, et que $y \mapsto \ell(y)$ est croissante.

Fixons un $y > 0$ et notons pour alléger la rédaction $u_n = f_n(y)$.

$$A_n(u_{n+1}) = A_{n+1}(u_{n+1}) - \frac{u_{n+1}^{n+1}}{n+1} \leq A_{n+1}(u_{n+1}) \leq y \leq A_n(u_n)$$

Et puisque la fonction A_n est croissante, il vient que $u_{n+1} \leq u_n$.

Comme d'autre part on a $u_n \geq 0$, la suite (u_n) est décroissante et minorée. Elle est donc convergente.

6 • Notons ℓ la limite de la suite (u_n) . La suite (u_n) étant décroissante, on a :

$$\forall n \in \mathbb{N}, y = \sum_{k=1}^n \frac{u_n^k}{k} \geq \sum_{k=1}^n \frac{l^k}{k} \geq 0$$

donc la série $\sum \frac{l^k}{k}$ converge donc

$$0 \leq l < 1$$

• Il existe donc un rang n_0 à partir duquel $0 \leq \frac{1+l}{2} < 1$ qui est le rayon de convergence de $\sum \frac{x^k}{k}$

donc la série entière $\sum \frac{x^k}{k}$ converge normalement donc uniformément sur $\left[0, \frac{1+l}{2}\right]$

donc si $n \geq n_0$, $|A_n(u_n) - (-\ln(1-u_n))| \leq \|R_n\|_{\infty}^{[0, \frac{1+l}{2}]}$ où R_n est le reste de Cauchy de cette série entière.

On en déduit que $\lim_{n \rightarrow +\infty} |A_n(u_n) - (-\ln(1-u_n))| = 0$. Or $A_n(u_n) = y$ et $\lim_{n \rightarrow +\infty} u_n = l$

Or $A_n(u_n) = y$ donc $\lim_{n \rightarrow +\infty} -\ln(1-u_n) = y$ et $\lim_{n \rightarrow +\infty} u_n = l$ donc $\lim_{n \rightarrow +\infty} -\ln(1-u_n) = -\ln(1-l)$.

On a donc $-\ln(1-l) = y$ donc $l = 1 - e^{-y}$.