KORREKTUR : TD PYTHON 16

Autour des séries entières

Exercice 01 : étude d'une somme de série entière

On pose : $f : x \mapsto \sum_{n=0}^{+\infty} x^{n^2}$ et $(a_n)_{n\in\mathbb{N}}$ la suite définie par $a_n = 1$ s'il existe p entier tel que $n = p^2$ et $a_n = 0$ sinon.

1. Écrire une liste d'instruction pour créer la suite (a_n) jusqu'à 1000 inclus.

2. Déterminer le domaine de définition de f.

3. Écrire une fonction `Somme` qui prend en argument x et n et renvoie $\sum_{k=0}^{n} a_k x^k$.

Tracer cette fonction pour $x \in [-1, 1]$. Que peut-on conjecturer ?

4. Écrire une fonction `NewSomme` qui prend en argument x et n et renvoie $\sum_{k=0}^{n} (-1)^k a_k x^k$.

Quel semble être sa limite en $x = 1$ quand n tend vers $+\infty$?

Solution

1. `mt.floor` désigne la partie entière et `np.sqrt` la racine carrée.

```
import math as mt ; import numpy as np
def a(n) :
    if mt.floor(np.sqrt(n)) == np.sqrt(n) :
        return 1
    else :
        return 0
```

2. On pose $u_n = x^{n^2}$. Si $x \neq 0$, $\frac{u_{n+1}}{u_n} = \frac{x^{(n+1)^2}}{x^{n^2}} = x^{2n+1}$.

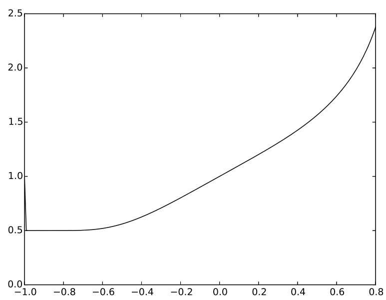
Il est clair que cette quantité tend vers 0 si et seulement si $|x| < 1$ et dans ce cas, la série $\sum u_n$ converge absolument.

Si $|x| > 1$, cette série diverge grossièrement, idem pour $x = 1$.

Pour $x = -1$, elle n'est évidemment pas absolument convergente. Par contre, $f(-1)$ peut avoir une valeur finie. On écrit pour l'instant $D_f =]-1, 1[$ et on va avec Python conjecturer plus loin le cas $x = -1$.

3. Il est clair ($Somme(1, n)$ donne le nombre d'entiers inférieurs à n qui sont des carrés) que $\sum_{n=0}^{+\infty} a_n x^n = \sum_{n=0}^{+\infty} x^{n^2}$. Donc : $\lim_{n\to+\infty} Somme(x, n) = f(x)$.

```
def Somme(x,n) :
    return(sum(a(k)*(x**k) for k in range(n+1)))
X = np.linspace(-0.98, 0.8,200)
plt.plot(X,Somme(X,2000)); plt.show()
```



On peut conjecturer que la limite de f quand x tend vers -1^+ est $1/2$ et $f(-1)$ semble être 1.

4. Encore une fois, on joue sur les intervalles car si x tend vers -1^+, `NewSomme` tend vers l'infini. Quant à la limite quand x tend vers 1^-, elle semble être $1/2$ mais monte brusquement vers 1. Il y a discontinuité en $x = 1$.

```
def NewSomme(x,n) :
    return(sum((-1)**k * a(k)*(x**k)
    for k in range(n+1)))
X = np.linspace(-0.8,1,2000)
Y = np.linspace(-1,0.8,200)
plt.plot(X,NewSomme(X,5000)); plt.show()
plt.plot(Y,NewSomme(Y,5000)); plt.show()

NewSomme(0.9999,10000) , NewSomme(1,10000)

     0.5000002 , 1
```

Exercice 02 : À propos d'une série de Taylor

Soit $f : x \mapsto \dfrac{1}{2-e^x}$ et on note pour tout $n \in \mathbb{N}$, $a(n) = \dfrac{f^{(n)}(0)}{n!}$.

1. Montrer, en utilisant $x \mapsto (2-e^x)f(x)$, que pour tout $n \in \mathbb{N}$, $a(n) = \displaystyle\sum_{k=1}^{n} \frac{a(n-k)}{k!}$.

2. Écrire avec Python une procédure pour calculer $a(n)$ pour $n \in [\![0, 10]\!]$.

3. Tracer (pour $n \in [\![0, 10]\!]$) les courbes :

$$n \mapsto (n,\, a(n))\,,\; x \mapsto \left(x,\, \frac{1}{(\ln 2)^x}\right) \text{ et } x \mapsto \left(x,\, \frac{1}{2(\ln 2)^x}\right).$$

En déduire le rayon de convergence de $\displaystyle\sum_{n\geqslant 0} a(n)x^n$. On note S sa somme.

4. Tracer l'approximation de la courbe S grâce aux sommes partielles pour $n \in [\![0, 6]\!]$.
Tracer la courbe de f sur $[0; 0.5]$. Que peut-on en déduire ?

Solution

1. On applique la **proposition 0.1** aux fonctions $x \mapsto (2-e^x)$ et $x \mapsto f(x)$. On part de $(2-e^x)f(x) = 1$. On écrit :

$$2 - e^x = 1 - \sum_{n=1}^{+\infty} \frac{x^n}{n!} = \sum_{k=0}^{+\infty} b_k x^k,$$

où $b_0 = 1$ et $b_k = -1/k!$ pour tout entier $k \geqslant 1$.

Et de même : $f(x) = \displaystyle\sum_{k=0}^{+\infty} a_k x^k$, avec pour tout $k \in \mathbb{N}$, $a_k = a(k)$. Alors :

$$(2-e^x)f(x) = \sum_{n=0}^{+\infty} c_n x^n = \sum_{n=0}^{+\infty} \left(\sum_{k=0}^{n} b_k a_{n-k}\right) x^n = 1.$$

Donc $c_0 = b_0\, a(0) = 1$. C'est cohérent !

Puis, pour tout $n \geqslant 1$, $c_n = \displaystyle\sum_{k=0}^{n} b_k a_{n-k} = 0$.

Ce qui entraîne l'implication : (En remplaçant b_k et a_k par leurs valeurs)

$$0 = a(n) - \frac{1}{n!} - \sum_{k=1}^{n-1} a(n-k) \times \frac{1}{k!} \Rightarrow a(n) = \sum_{k=1}^{n} a(n-k) \times \frac{1}{k!}.$$

2.

```
import math as mt
def a(n):
    if n == 0 :
        return 1
    else:
        return sum(a(n-k)/(mt.factorial(k)) for k in range(1,n+1))
```

Il est temps de faire fonctionner.

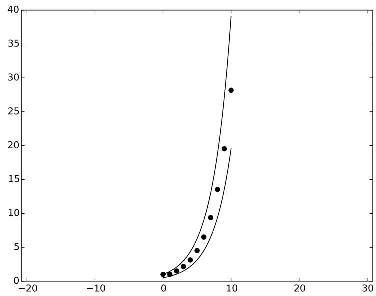
```
for i in range(11):
    print(a(i))
1
1.0
1.5
2.1666666666666665
3.1249999999999996
4.508333333333334
6.504166666666667
9.383531746031748
13.537574404761907
19.530591380070547
28.176687334656087
```

3. En suivant les notations choisies dans l'indication de cet exercice. On commence par charger les intervalles d'abscisses et d'ordonnées et les fonctions en jeu.

```
import numpy as np
X = [k for k in range(0,11)]
XX = np.linspace(0,10,500)
Y = [a(k) for k in range(0,11)]
def fsup(x):
    return 1/(mt.log(2))**x
def finf(x):
    return 1/(2*(mt.log(2)**x))
YYsup = fsup(XX); YYinf = finf(XX)
```

Puis on passe aux graphes. Par manque de place, les commandes sont représentées ligne par ligne mais vous, mettez les sur la même ligne.

```
import matplotlib.pyplot as plt
 plt.plot(X,Y,"o",color='0');
 plt.plot(XX,YYsup,color='0');
 plt.plot(XX,YYinf,color='0');
 plt.axis('equal'); plt.show()
```



On travaille sous conjecture.
On a alors : $\forall n \in \mathbb{N}$, $\dfrac{1}{2(\ln 2)^n} \leqslant a(n) \leqslant \dfrac{1}{(\ln 2)^n}$.
En multipliant par x^n, la série $\sum\limits_{n\geqslant 0} a_n x^n$ a le même rayon de convergence que $\sum\limits_{n\geqslant 0} \dfrac{x^n}{(\ln 2)^n}$, qui converge si, et seulement si, $|x| < \ln 2$.
Le rayon de convergence est $R = \ln 2$.

4. On commence par rentrer S d'après les dires de l'énoncé et son compagnon d'infortune f.

```
def S(x):
    return sum(a(n)*(x**n) for n in range(0,7))
def f(x):
    return 1/(2 - np.e**x)
```

On est incité à tracer ensemble le graphe de f et de S, sinon que remarquer ?

```
import matplotlib.pyplot as plt; import numpy as np
XXX = np.linspace(0,0.5,500)
plt.plot(XXX,f(XXX),linestyle=':',color='0');
plt.plot(XXX,S(XXX),color='0'); plt.axis('equal');
plt.show()
```

On remarque que f et S ont l'air d'être égales. Surprenant ? Pour être plus précis, les deux courbes se décalent à partir de $x = 0.4$ environ, cela est dû au fait que S est la somme partielle des 7 premiers termes seulement.