

TP 8 : Suites et probabilités

Suites

Que fait la fonction suivante ?

```
def mystere(n):
    H=1
    k=1
    indices=[1]
    for i in range(2,n+1):
        while H<i:
            k=k+1
            H=H+1./k
        print(k,H)
        indices.append(k)
    return(indices)
```

Et que constate-t-on sur le tracé suivant ?

```
n=8
plt.plot(range(1,n+1),mystere(n))
t=np.linspace(1,n,1000)
plt.plot(t,np.exp(t))
plt.show()
```

Probabilités

```
import random as rd
import matplotlib.pyplot as plt
import numpy as np
```

1. Qu'observez-vous dans chacun des exemples ci-dessous ? Relancez les blocs d'instruction à plusieurs reprises !

```
L=[rd.random() for k in range(2000)]
Bornes=np.linspace(0,1,11)
plt.hist(L,Bornes)
plt.show()
```

```
L=[int(6*rd.random()+1) for k in range(2000)]
Frequencies=[L.count(k)/1000 for k in range(1,7)]
plt.bar(np.arange(1,7),Frequencies)
plt.show()
```

```
p=0.7
L=[rd.random()<p for k in range(2000)]
Frequencies=[L.count(k)/2000 for k in [True,False]]
plt.bar(np.arange(1,3),Frequencies)
plt.show()
```

2. Utiliser les codes précédents pour simuler les exercices de la feuille de TD, par exemple 15, 16, 17...

3. Une suite à l'exercice 18.

On appelle valeur de dépistage d'une maladie la probabilité qu'un individu soit malade sachant que le test est positif. On note f la proportion de malades au sein de la population. Commenter le code suivant et le simplifier si possible.

```
nb_individus_total=1000
L=[]
for f in range(0,101):
    nb_positifs=0
    nb_malades_positifs=0
    for k in range(1,nb_individus_total):
        Individu_malade=False
        Test_positif=False
        if 100*rd.random()<f:
            Individu_malade=True
            if rd.random()<0.99:
                Test_positif=True
        else:
            if rd.random()<0.002:
                Test_positif=True
        if Test_positif and Individu_malade:
            nb_malades_positifs = nb_malades_positifs+1
        if Test_positif:
            nb_positifs = nb_positifs+1
    L.append(nb_malades_positifs/nb_positifs)
A=np.arange(101)
plt.plot(A,L)
plt.show()
```