

## Simulations en terminale

Proportion d'échantillons de taille  $n$  tels que  $\left|S_n - \frac{n}{6}\right| > \sqrt{n}$ .

```
1 from random import*
2 from math import*
3 from matplotlib import*
4 from matplotlib.pyplot import*
5 def segment(x0,y0,x1,y1):
6     lx,ly=[x0,x1],[y0,y1]
7     plot(lx,ly,'b-')
8
9 def échantillon(n):
10    Nombre=0
11    for i in range(n):
12        if randint(1,6)==6:
13            Nombre=Nombre+1
14    return Nombre
15
16 def écart(n,N):
17    Compt=0
18    for i in range(N):
19        if abs(échantillon(n)-n/6)>sqrt(n):
20            Compt=Compt+1
21    return Compt/N
22
23 def graphe(n,N,E):
24    L=[]
25    for k in range(E):
26        L.append(écart(n,N))
27    axis([0,E,0.0,0.2])
28    xlabel("Nombre d'échantillons")
29    ylabel("proportions observées")
30    title("Programme terminale")
31    grid(True)
32    plot(L,'b-')
33    segment(0,5./36,100,5./36)
34    show()
35 graphe(20,100,100)
36
```

Moyenne  $m$  des valeurs d'un échantillon telle que  $|m - 1030| \leq 4$  ou 8 ou 12

```
1 from math import*
2 from numpy import*
3 from random import*
4 from matplotlib import*
5 from matplotlib.pyplot import*
6
7 def moyenne(n):
8     L1=[]
9     for j in range(n):
10        P=random()
11        if P<=0.6:
12            P=1000
13        elif P<=0.8:
14            P=1050
15        else:
16            P=1100
17        L1.append(P)
18    return(mean(L1))
19
20 def graph(n,E):
21    for i in range(E):
22        plot(i,moyenne(n),'+',color='red')
23    axis([0,100,1010,1050])
24    plot([0,100],[1022,1022],color='blue')
25    plot([0,100],[1038,1038],color='blue')
26    plot([0,100],[1026,1026],color='red')
27    plot([0,100],[1034,1034],color='red')
28    plot([0,100],[1018,1018],color='green')
29    plot([0,100],[1042,1042],color='green')
30    show()
31 graph(50,100)
32
```

Proportion des échantillons dont la moyenne  $m$  des valeurs vérifie  $|m - 1030| \leq 4$  ou 8 ou 12

```

1 from math import*
2 from numpy import*
3 from random import*
4 from matplotlib import*
5 from matplotlib.pyplot import*
6
7 def moyenne(n):
8     L1=[]
9     for j in range(n):
10        P=random()
11        if P<=0.6:
12            P=1000
13        elif P<=0.8:
14            P=1050
15        else:
16            P=1100
17        L1.append(P)
18    return(mean(L1))
19
20 def écart(n,N,s):
21    Compt=0
22    for i in range(N):
23        if abs(moyenne(n)-1030)<=s:
24            Compt=Compt+1
25    return Compt/N
26
27 def graphe(n,N,s,E):
28    L=[]
29    T=[]
30    Q=[]
31    for k in range(E):
32        L.append(écart(n,N,s))
33        T.append(écart(n,N,2*s))
34        Q.append(écart(n,N,3*s))
35    axis([0,E,0.0,1])
36    xlabel("Nombre d'échantillons")
37    ylabel("Proportions observées")
38    title("Programme terminale")
39    grid(True)
40    plot(L,'r-',label='|m - $\mu$|<4')
41    plot(T,'b-',label='|m - $\mu$|<8')
42    plot(Q,'g-',label='|m - $\mu$|<12')
43    legend()
44    show()
45 graphe(100,100,4,100)

```

Proportion d'échantillons tels que  $|S_n - \mu| > 2\sigma$ .

```

1 from math import*
2 from numpy import*
3 from random import*
4 from matplotlib import*
5 from matplotlib.pyplot import*
6
7 def segment(x0,y0,x1,y1):
8     lx,ly=[x0,x1],[y0,y1]
9     plot(lx,ly,'b-')
10
11 def echantillon(n):
12     Nombre=0
13     for j in range(n):
14         if randint(1,6)==6:
15             Nombre=Nombre+1
16     return Nombre
17
18 def écart(n,N):
19     Compt=0
20     for i in range(N):
21         if abs(echantillon(n)-n/6)>=2*sqrt(5*n/36):
22             Compt=Compt+1
23     return Compt/N
24
25 def graph(n,N,E):
26     L=[]
27     for i in range(E):
28         L.append(écart(n,N))
29     axis([0,E,0,0.5])
30     xlabel("N Echantillons")
31     ylabel("Fréquences observées")
32     grid(True)
33     plot(L,'b-')
34     segment(0,0.25,E,0.25)
35     show()
36
37 graph(500,100,100)
38
39

```

## Recherche d'une taille d'échantillon pour majorer une probabilité.

```
• 1 from math import*
• 2 from numpy import*
• 3 from random import*
  4
  5 def echan(n,N):
• 6     L=[]
• 7     p=0
• 8     for i in range(N):
• 9         c=0
•10         for j in range(n):
•11             if randint(1,6)==6
•12                 c=c+1
•13             if abs(c/n-1/6)>0.01:
•14                 p=p+1
•15     f=float(p)/N
•16     L.append(f)
•17     M=mean(L)
•18     print(L)
•19     print(M)
•20 echan(5000,1000)
  21
```