Mouhamadou Moustapha Gueye
Lynda Hedir

Institut Galilée
UNIVERSITÉ PARIS 13

Compte rendu du TP2

Introduction :

Dans ce TP, le but est de représenter quelques méthodes numériques pour calculer, approximativement l'intégrale d'une fonction sur un intervalle compact (borné, fermé).

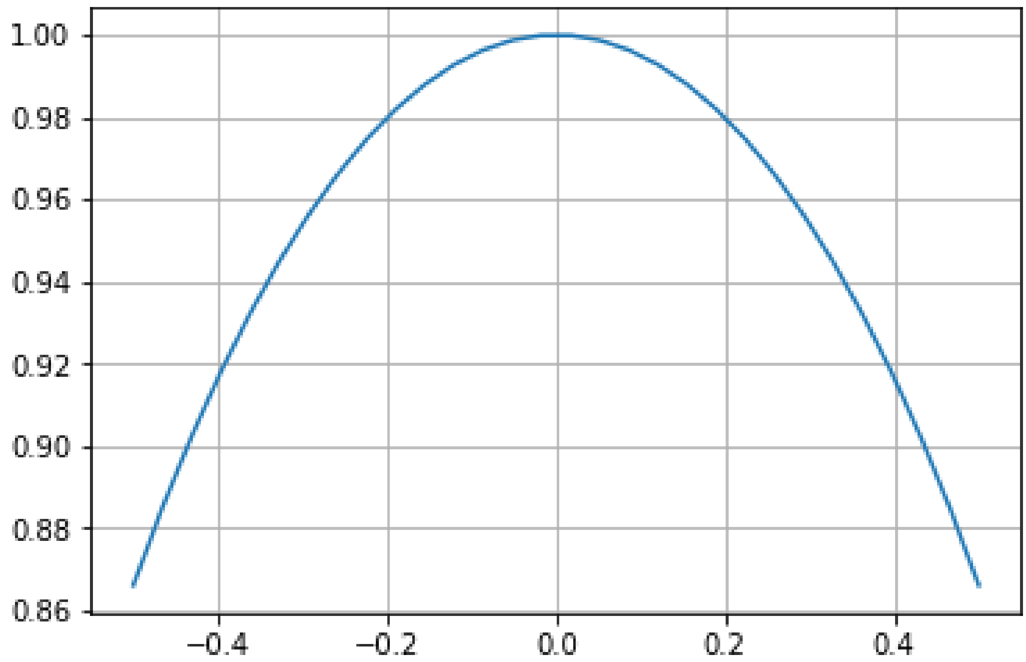
Exercice 1:

Dans cette partie, on procède à la création d'une fonction, à une représentation graphique et à un calcul exact de l'intégrale.

- *Création de la fonction* $f(x)=\sqrt{(1-x^2)}$ *en langage python:*

```
# -*- coding: utf-8 -*-
"""
Éditeur de Spyder

Ceci est un script temporaire.
"""
import math
import matplotlib.pyplot as plt
import numpy as np

#Exercice 1
#a) Création de la fonction f(x)

def f(x):
        return math. sqrt(1-x**2)

#Representation graphique de f(x)

x=np.linspace(-0.5,0.5,30) #création de l'abscice x
y=[f(i) for i in x] #creation de l'ordonnee y
plt.plot(x,y)
plt.grid()
plt.show
```

- *Représentation graphique de la fonction*



Mouhamadou Moustapha Gueye
Lynda Hedir

- *Calcul de l'intégrale à la main*

* Calculons l'intégrale à la main:

$f(x) = \sqrt{1-x^2}$ et $\int_{-0,5}^{0,5} \sqrt{1-x^2}\,.dx$

Posons $x = \sin\theta$, $dx = -\cos\theta . d\theta$, si : $x = \frac{1}{2} \Rightarrow \theta = \frac{\pi}{6}$

si $x = -\frac{1}{2} \Rightarrow \theta = -\frac{\pi}{6}$.

$$\int_{-\frac{1}{2}}^{\frac{1}{2}} \sqrt{1-x^2}\,.dx = \int_{-\frac{\pi}{6}}^{\frac{\pi}{6}} \cos\theta \sqrt{1-(\sin\theta)^2}\,.d\theta = \int_{-\frac{\pi}{6}}^{\frac{\pi}{6}}$$

or $\cos 2\theta = 2\cos^2\theta - 1 \Rightarrow \cos^2\theta = \frac{1+\cos 2\theta}{2}$.

$$\Rightarrow \int_{-\frac{\pi}{6}}^{\frac{\pi}{6}} \cos^2\theta . d\theta = \int_{-\frac{\pi}{6}}^{\frac{\pi}{6}} \left(\frac{1+\cos 2\theta}{2}\right) . d\theta.$$

$$= \int_{-\frac{\pi}{6}}^{\frac{\pi}{6}} \frac{1}{2}\, d\theta + \int_{-\frac{\pi}{6}}^{\frac{\pi}{6}} \frac{\cos 2\theta}{\theta} . d\theta.$$

$$= \int_{-\frac{\pi}{6}}^{\frac{\pi}{6}} \left[\frac{1}{2}\theta\right]_{-\frac{\pi}{6}}^{\frac{\pi}{6}} + \left[\frac{\sin 2\theta}{4}\right]_{-\frac{\pi}{6}}^{\frac{\pi}{6}}$$

$$= \frac{\pi}{6} + \left[\frac{2}{4}\left(\frac{\sqrt{3}}{2}\right)\right]$$

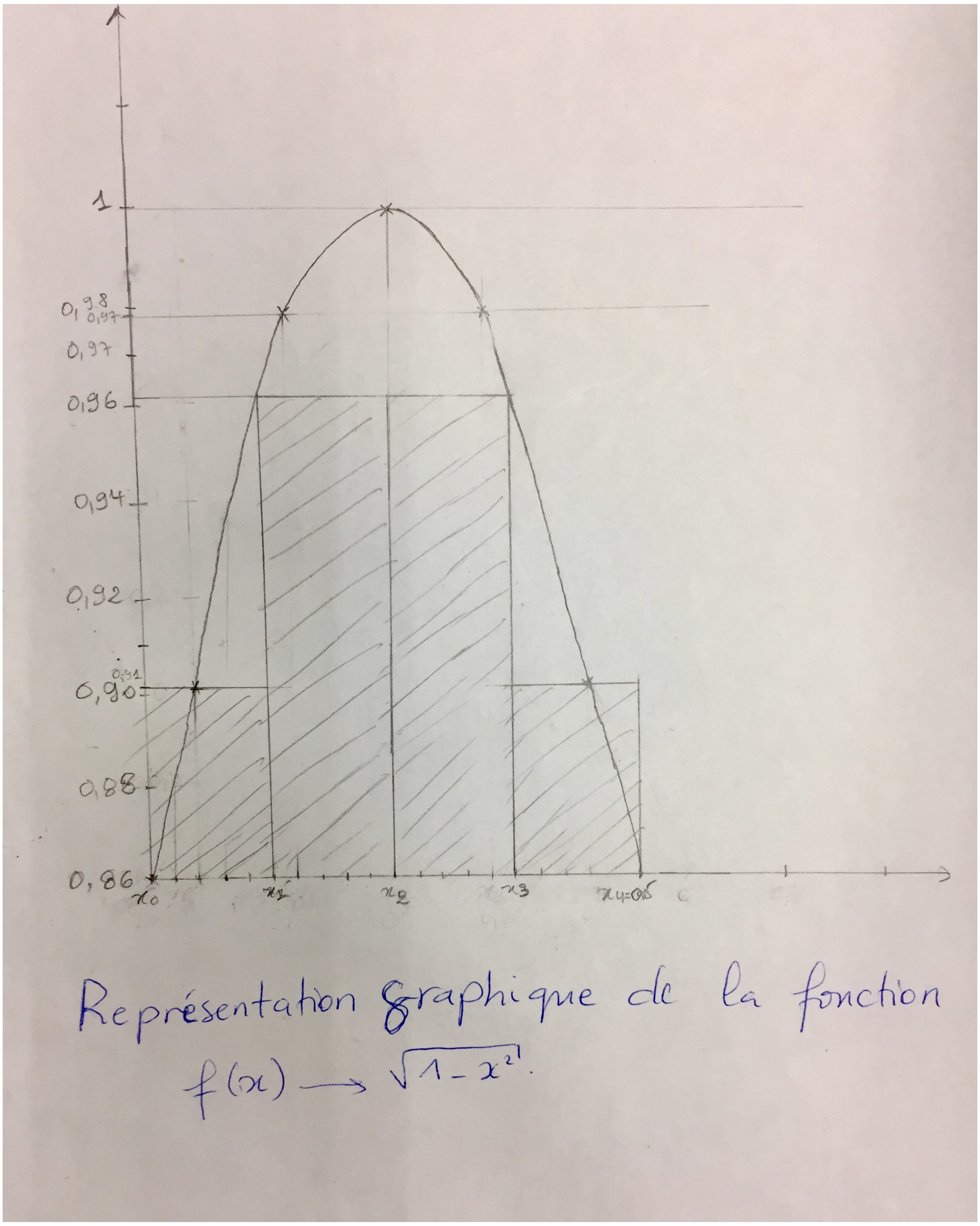
$$= \frac{\pi}{6} + \frac{\sqrt{3}}{4}.$$

Ainsi: $\boxed{\int_{-\frac{1}{2}}^{\frac{1}{2}} \sqrt{1-x^2}\,.dx = \frac{\pi}{6} + \frac{\sqrt{3}}{4}}$

Mouhamadou Moustapha Gueye
Lynda Hedir

Exercice 2 :

a) *Représentation graphique de la fonction* $f(x)=\sqrt{(1-x^2)}$



Mouhamadou Moustapha Gueye
Lynda Hedir

b) *Calcul approché de l'intégrale I au moyen de la méthode du point milieu*

- *Création des valeurs de la liste (x0, x1,….,x5)*

```
for i in range(5):
        x=-0.5 + 0.25*i
        X.append(x)
```

- *Détermination de milieu de chaque subdivision*

```
for i in range(4):

        c=(X[i]+X[i+1])/2
        C.append(c)
```

- *Calcul des aires de petits rectangles sous la courbe*

```
for i in range(4):

        a=(f(C[i])*(0.25))
        A.append(a)
```

- *Calcul de l'aire totale sous la courbe (en faisant la somme des aires des petits rectangles)*

```
s=0
for i in range(4):
        s=A[i]+s
```

Mouhamadou Moustapha Gueye
Lynda Hedir

c) *Calcul de l'erreur commise*

Erreur = S4 – I

Erreur = 0.0029792992680897O8

d) *Mesure du temps de calcul de l'intégrale*

```
depart = time.clock()
X=[]
C=[]
A=[]

#Création des valaurs de la liste (x0, x1,....,x5), append permet
#de regrouper tous les x


for i in range(5):
        x=-0.5 + 0.25*i
        X.append(x)

#On determine le milieu de chaque subdivision


for i in range(4):

        c=(X[i]+X[i+1])/2
        C.append(c)

#Calcul des aires f(c[i]) represente la hauteur


for i in range(4):

        a=(f(C[i])*(0.25))
        A.append(a)

#Calcul de l'aire totale sous la courbe en faisant la somme des
#aires des petits rectangles


s=0
for i in range(4):
        s=A[i]+s


arrivee=time.clock()

temps_ecoule= arrivee -depart

print(temps_ecoule)
```

Mouhamadou Moustapha Gueye
Lynda Hedir

e) *Calcul de l'intégrale I avec la **fonction python point_milieu***

```
def point_milieu(f,d,b,n):
    depart = time.clock()
    X, C, A = [], [], []

    for i in range(n):

            #On definit les subdivisions
            x=d + ((b-d)/(n-1))*i
            X.append(x)

    #On determine le milieu de chaque subdivision
    for i in range(n-1):
            c=(X[i]+X[i+1])/2
            C.append(c)

    #Calcul des aires f(c[i]) represente la hauteur
    for i in range(n-1):
            a=(f(C[i])*((b-d)/(n-1)))
            A.append(a)

    '''
    Calcul de l'aire totale sous la courbe en faisant
    la somme des aires des petits rectangles
    '''
    s = 0
    for i in range(n-1):
            s=A[i]+s
    arrivee = time.clock()
    return s, arrivee-depart
```

f) Test de la fonction point_milieu

$f(x)=\sqrt{(1-x^2)}$

$a=-0{,}5$

$b=0{,}5$

$n=10^k$ (k variant de 1 à 6)

n	erreur	temps
10	5.928544e-04	2.600000000e-05
100	4.909353e-06	8.500000000e-05
1000	4.869940e-08	7.650000000e-04
10000	9.717425e-10	7.328000000e-03
100000	4.953200e-10	6.587800000e-02
1000000	4.905648e-10	7.658270000e-01

Mouhamadou Moustapha Gueye
Lynda Hedir

Exercice 3 :

- *Calcul de l'intégrale I avec la* ***méthode du trapèze***

```
def trapeze(f,a,b,n):
    depart = time.clock()
    h = (b - a)/n
    S = 0

    for i in range(n):
        xg = a + h*i
        xd = a + (i+1)*h
        s = h*((f(xd)+f(xg)))/2
        S = S + s
    arrivee = time.clock()

    return S, arrivee-depart
```

Exercice 4 :

- *Calcul de l'intégrale I avec la* **méthode de Simpson**

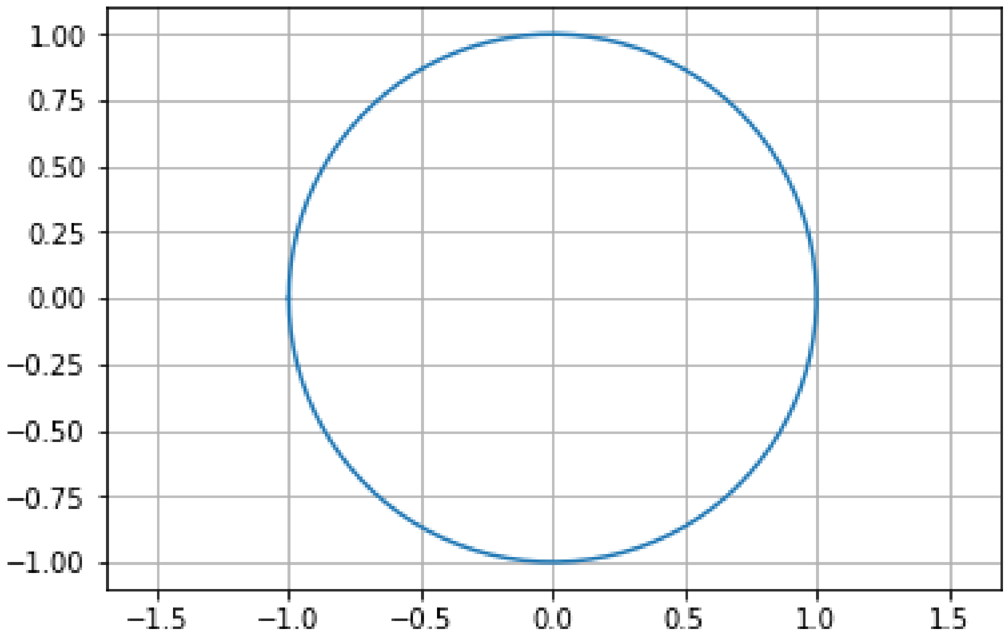
```
def point_simpson (f,a,b,n):
    m = (b - a)/n
    xi = a
    xj = a
    i = 1
    S = 0
    while i <= n:
                xi = a + m*i
        xj = a + m*(i - 1)
        si = (f(xi) + 4*f((xi + xj)/2) +f(xj))/6*m
        S = S + si
        i = i + 1
    return S
```

Exercice 5 :

- *Calcul de l'intégrale I avec la* **méthode de Monte-Carlo**

a) Cercle unité

```
from math import *
import matplotlib.pyplot as plt
from time import *
from fonctiontp2 import *
import numpy as np


t= np.linspace(-pi, pi, 100)

x= cos(t)
y= sin(t)

plt.plot(x,y)
plt.grid('on')
plt.axis('equal')

```

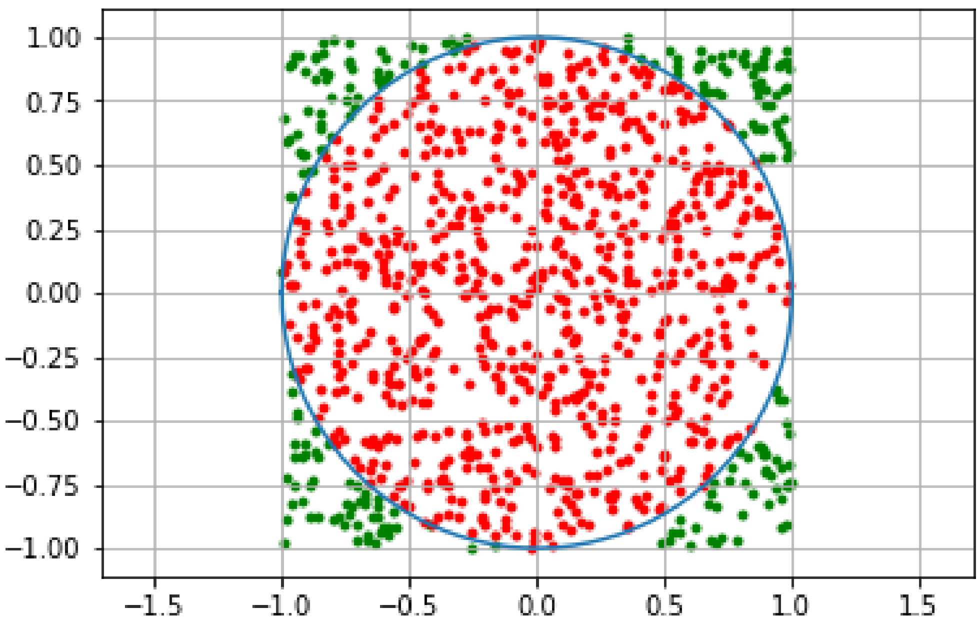


La surface du cercle est πr^2 , or le rayon est égale à 1, donc la surface du cercle unité est égale à π .

Mouhamadou Moustapha Gueye
Lynda Hedir

c) *On génère N = 1000 points suivant la distribution uniforme sur le carré $[-1,1]^2$ et on les place sur le graphique.*

```
from numpy import *
t = linspace(-pi, pi, 100)

x = cos(t)
y = sin(t)

plt.plot(x,y)
plt.grid('on')
plt.axis('equal')

for i in range(1000):
     xpoint = 2*random.rand()-1
     ypoint = 2*random.rand()-1
     if (xpoint**2+ypoint**2)<1:
          plt.scatter(xpoint,ypoint, color='red', marker='.')
     else:
          plt.scatter(xpoint,ypoint, color='green', marker='.')
plt.savefig('Cercle monte_carlo')
```



f) Test de la fonction Monte Carlo avec $N=10^k$, k variant de 1 à 6.

```
Tableau avec la methode monte_carlo:
       n |            erreur       |    temps (sec.)
----------------------------------------
      10|            0.0224073  |   0.003231
     100|            0.0455927  |   0.003137
    1000|            0.0504073  |   0.003129
   10000|            0.0215927  |   0.003149
  100000|            0.0575927  |   0.003112
   1e+06|            0.0264073  |   0.003092
   1e+07|           0.00240735  |   0.003096
```