Integración Numérica
Método del Trapecio

Carlos Muñoz De La Toba

November 2018

1 Introduction

A veces cuando se tiene un problema por ejemplo una integral como la siguiente

$$I = \int_0^3 \frac{x^4 e^x}{(e^x - 1)^2} \tag{1}$$

La cual desde luego es una integral que talvez analíticamente no sea nada sencilla calcular.

Para estos casos podemos hacer uso de diferentes métodos numéricos. En ésta ocación presentaremos el Método del Trapecio para integrales definidas.

2 Método del Trapecio

Este es un método numérico para integración que consiste basicamente en aproximar el valor de la integral de una función $f(x)$ en un intervalo definido dividiendolo en pequeños trapecios, que se subdividen a la vez en en rectángulos y triángulos, a lo largo de la curva

Empezamos definiendo la base de los rectángulos definiendo un número de puntos "n" en el que se dividirá el segmento $[x_i, x_f]$, en este caso, $[a, b]$ como se muestra en la figura 1 esto nos da los Δx. Empezando con el trapecio comprendido entre $[a, x_1]$,al que llamaremos a_1 el cual tiene $base = x_1 - a$ y la altura del rectángulo de $f(a)$. Después calculando el triángulo que nos sobra tenemos que su base es la misma que la del rectángulo pero ahora su altura es $f(x_1) - f(a)$. Así sumando las áreas obtenidas tenemos que

$$a_1 = \frac{1}{2}(x_1 - a)(f(a) + f(x_1)) \tag{2}$$

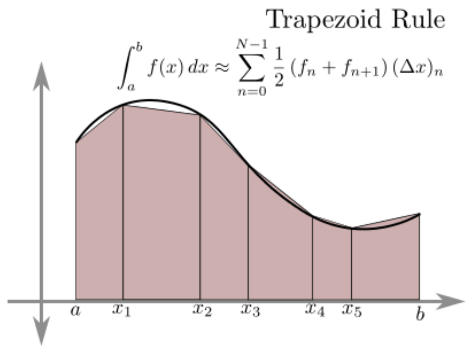

Figure 1: Caption

Asaí repitiendo para cada uno de los trapecios obtendremos una suma de la forma

$$\int_a^b f(x)dx \approx \frac{\Delta x}{2}(f(x_0) + 2(f(x_1) + f(x_2) + \ldots + f(x_{n-1})) + (f(x_n)) \quad (3)$$

Donde $\Delta x = \frac{b-a}{n}$ y $x_i = \alpha + i\Delta x$

Entonces arreglando

$$\int_a^b f(x)dx \approx \sum_{k=1}^{N} \frac{f(x_{k-1}) + f(x_k)}{2} \Delta x_k \quad (4)$$

Código del Método de Trapezoide en Fortran

```
  program trapezoide

implicit none
integer :: n
real :: u,a,b,s,f,t
real,allocatable :: r(:)
integer :: i,k

OPEN (UNIT=11, FILE="resultadostrap.txt",status = "unknown")

write(*,*)"Ingresa el numero de puntos"
```

```
read(*,*) n
allocate(r(n))

write(*,*)"Ingresa el limite inferior"

read(*,*) a

write(*,*)"Ingresa limite superior"

read(*,*) b

write(*,*) "-----------------------------"


if ( a <= 0) then
   a=0.000001
end if

   s = f(a)
   t = f(b)

   print*,"-----------------------"

   write(*,*) 'La funcion evaluada en el limite inferior es', s

   print*,"-----------------------"

    write(*,*) 'La funcion evaluada en el limite superior es', t

    u = ((b-a)/n)

      print*,"-----------------------"

      write(*,*) 'El tamaño de las divisiones es', u


      print*,"-----------------------"

do i =1,n

r(i) = ((u/2)*(2*f(((((b-a)*i)/n) + a))))

write(*,*) "el trapecio ", i, "tiene area", r(i)

write(11,*) r(i)
```

```
  end do

  close(unit=11)

  print*,"-----------------------"

  print*, "El area es",sum(r)


 print*,

  end program trapezoide


 real function f(x)
  implicit none
  real :: x
  f = (x**4)*EXP(X)/((EXP(X)-1.0)**2)

end function f
```