

5.15 and 5.16

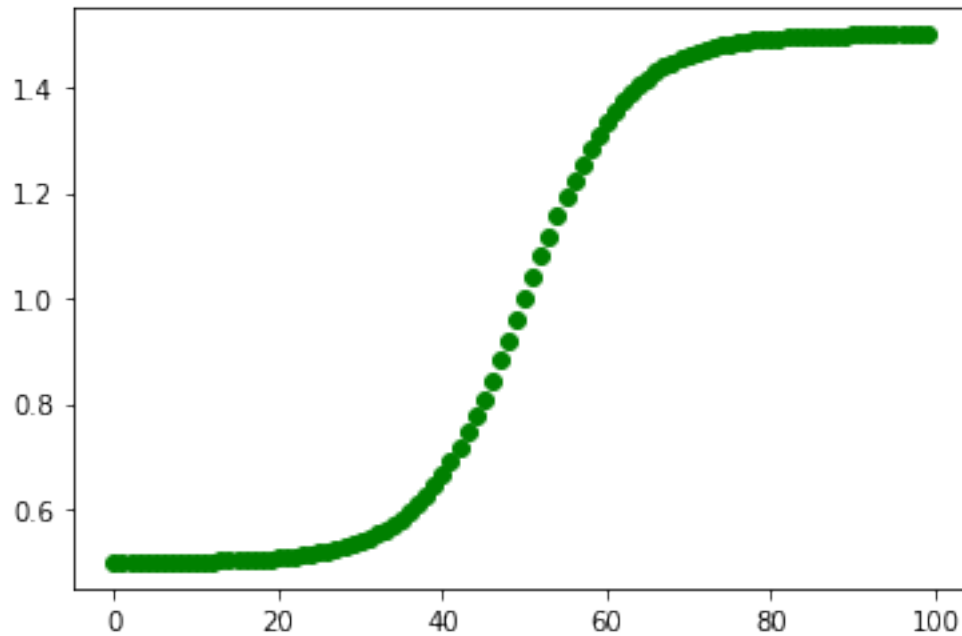
June 3, 2020

1 problema 5.15 y 5.16

```
[1]: from numpy import empty, zeros, max, cos, sin, pi, linspace
    from pylab import imshow, gray, show, □
    ↪ hot, sqrt, tanh, cosh, plot, xlabel, ylabel, legend, title
```

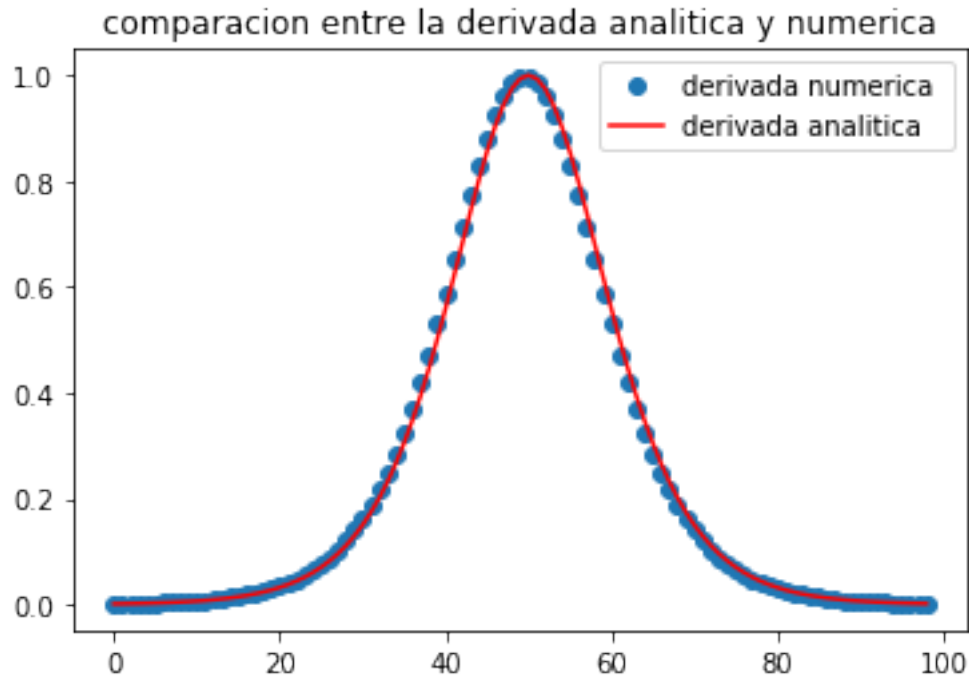
```
[2]: def f(x):
    return 1 + 0.5 * tanh(2*x)
    def dfunc(x):
    return 1-tanh(2*x)**2
```

```
[3]: #parametros y discretizacion de la funcion
    a= -2
    b=2
    N=100
    h=(b-a)/N
    fp=[]
    for i in range (N):
        fp.append(f(a+i*h))
    plot (fp, "go")
    show()
```



```
[4]: # derivada numerica por metodo de derivadas hacia adelante
dfn=[]
dfa=[]
for i in range (N-1):
    dfn.append ((fp[i+1] - (fp[i]))/h)
    dfa.append (dfunc(a+i*h))
```

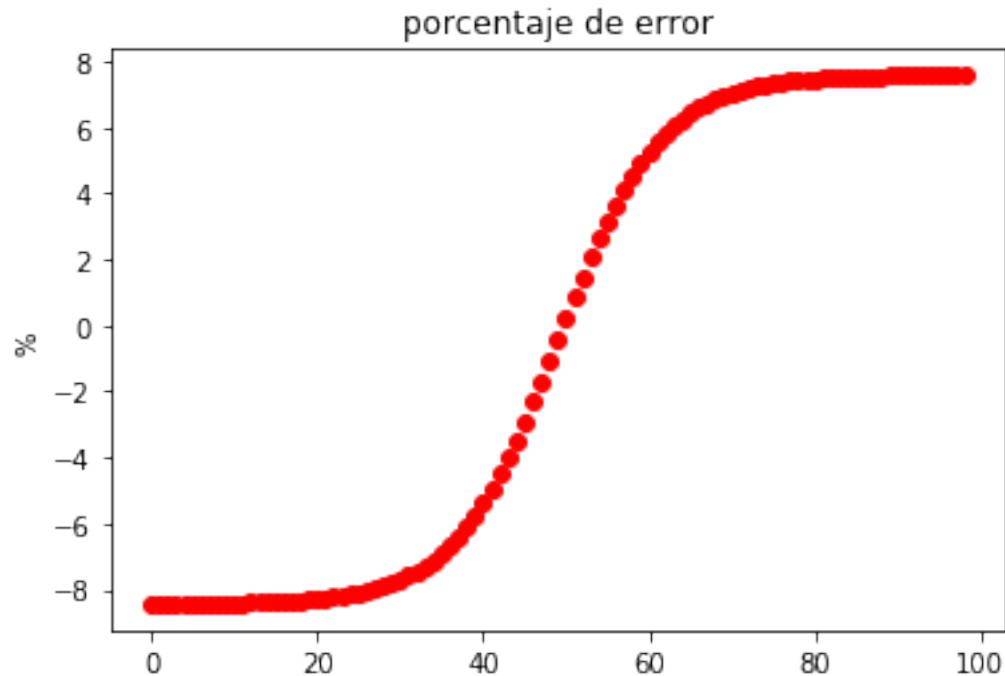
```
[5]: # comparacion entre la derivada analitica y numerica
plot(dfn,'o',label='derivada numerica ')
plot(dfa,'r',label='derivada analitica')
title('comparacion entre la derivada analitica y numerica')
legend()
show()
```



cómo se observa el método de derivadas hacia adelante, funciona muy bien, aunque los valores no son muy exactos algunos de ellos se notan que están un poco sesgados de los valores de la derivada analítica, aunque la exactitud de la derivada numérica también depende del valor de N quien a medida que este se aumenta las gráficas tienden a ser más similares.

1.0.1 calculo del error metodo de derivadas hacia adelante

```
[6]: diff=[]
for i in range(N-1):
    tmp=1- (dfn[i]/dfa[i])
    diff.append(tmp*100)
plot (diff,'ro')
title('porcentaje de error')
ylabel('%')
show()
```

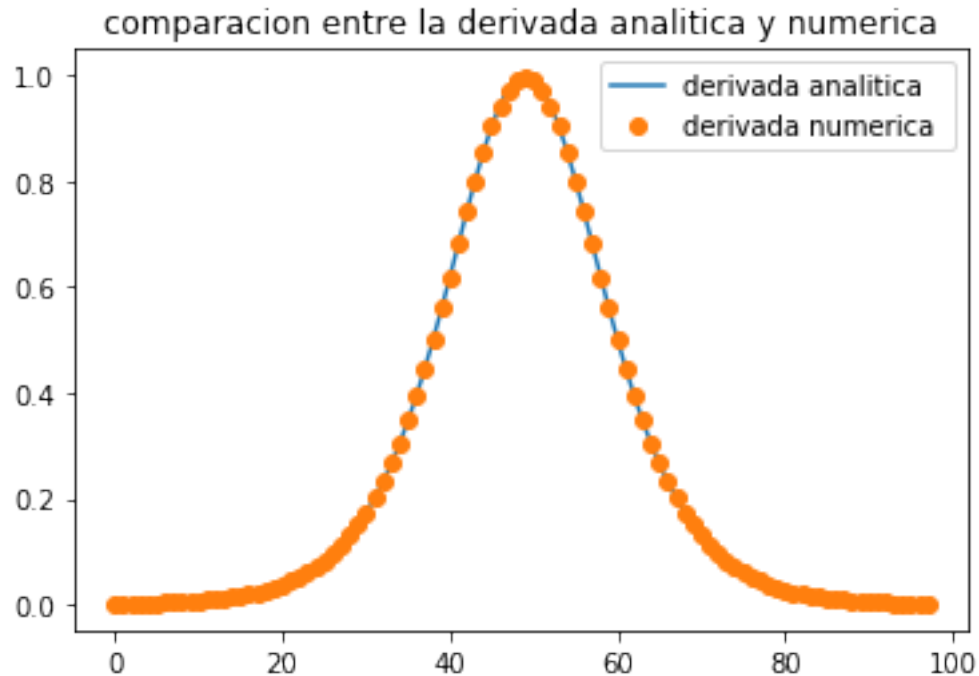


el error en este método es notable y su valor depende del valor de N como lo había dicho antes, este error puede ser considerable en este caso con un N de 100 el error está oscilando entre el valor de un dígito.

1.1 Método derivadas centrales.

```
[7]: # derivada numerica por metodo de derivadas centrales
dfn=[]
dfa=[]
for i in range (1,N-1):
    dfn.append((fp[i+1]-fp[i-1])/(2*h))
    dfa.append(dfunc(a+i*h))
```

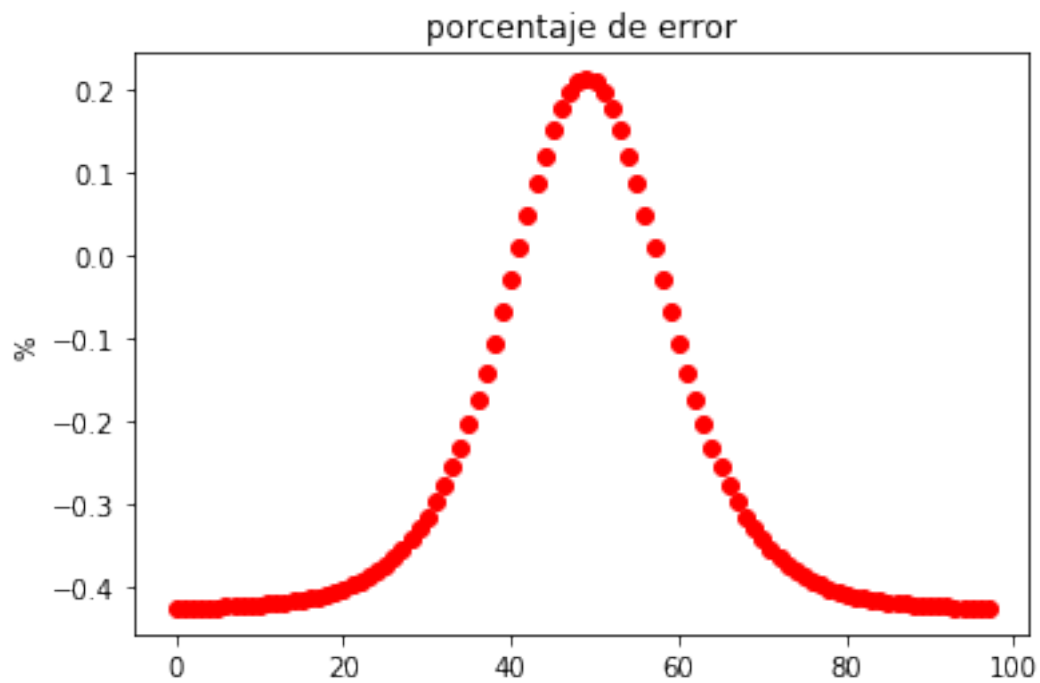
```
[8]: plot (dfa,label='derivada analitica')
plot (dfn,'o',label='derivada numerica ')
title('comparacion entre la derivada analitica y numerica')
legend()
show()
```



el método de derivadas centrales se observa que las grafica de la derivada numérica es mucho más semejante que en el método anterior lo que indica que este método es más preciso, este debido a que este método toma un promedio en los pasos que realiza.

1.1.1 calculo del error metodo de derivadas centrales

```
[9]: diff=[]
    for i in range (N-2):
        tmp=1-(dfn[i]/dfa[i])
        diff.append(tmp*100)
        show()
    plot(diff,'ro')
    title('porcentaje de error')
    ylabel('%')
    show()
```



al tratarse de un método más exacto el porcentaje de error es mucho menor que el método anterior, de manera considerable, aunque la precisión del método también está ligada al valor que se le atribuya a N