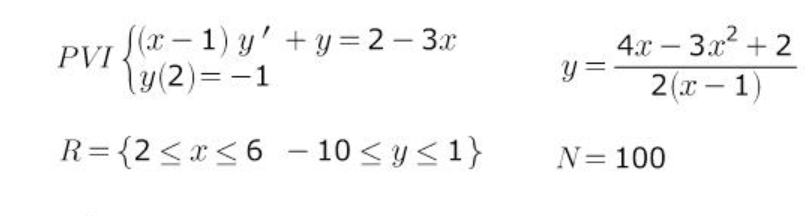
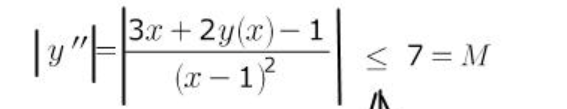
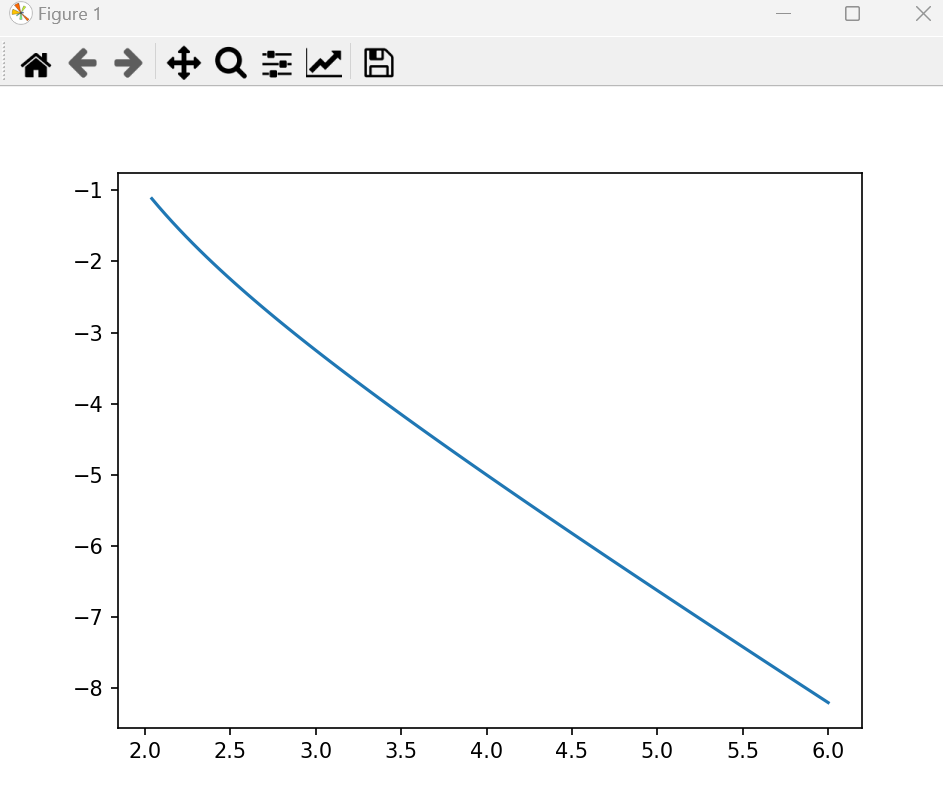
Método de Taylor

Ejericicio1:

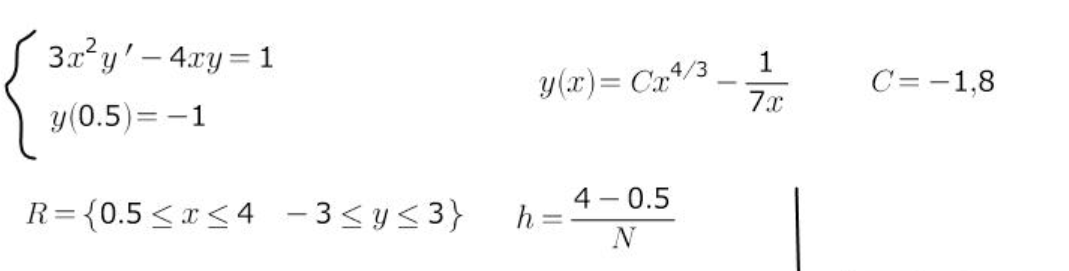


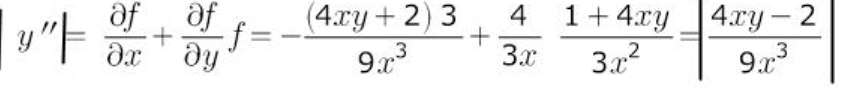


Tras realizar el código tenemos, solución real es y(6) = -8,20 y la solución aproximada es w100 = -8.1997 --> tenemos un error de 0.0003.

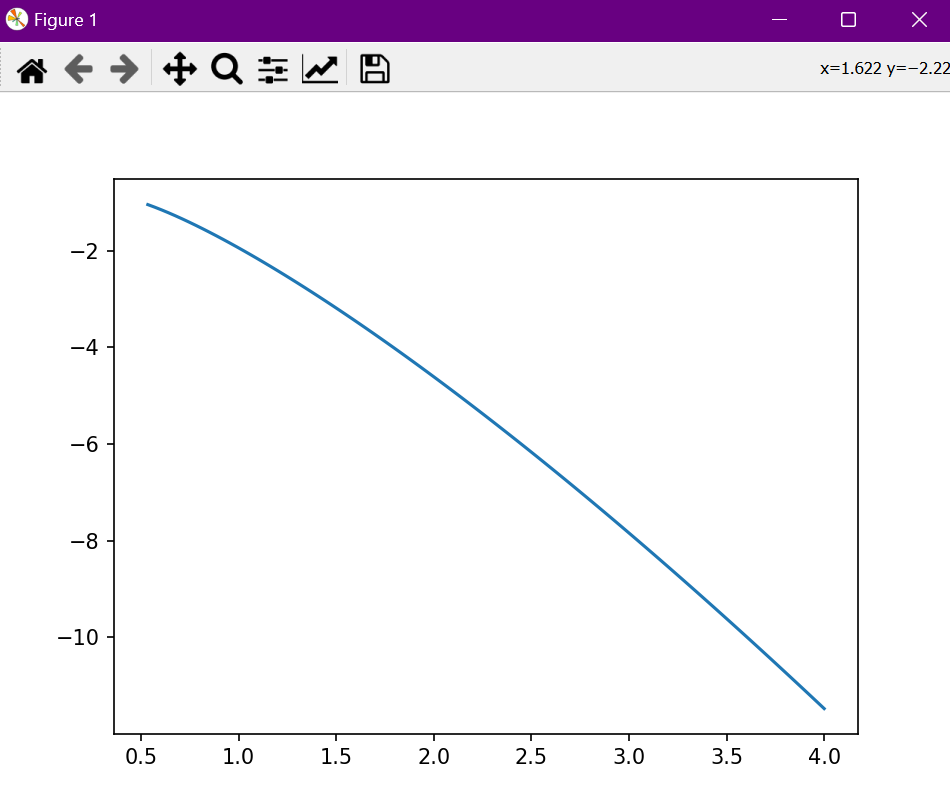


Ejercicio2:





Tras realizar el código tenemos, solución real es y(4)= -11,46 y la solución aproximada es w100 = -11.47088 --> tenemos un error de 0.01088



Código:

*import matplotlib.pyplot as plt*

*import numpy as np*

*def funcion(x,y):*

*return (1 + 4\*x\*y)/(3\*x\*\*2)*

*def segunda\_derivada(x, y):*

*return (4\*x\*y -2)/(9\*x\*\*3)*

*def iterar(x, y, f, d2):*

*'''Itera la función'''*

*while x <= xf:*

*xn = x + h*

*yn = y + h \* f (x,y) + (h\*\*2/2) \* d2(x,y)*

*iterar(xn, yn, f, d2)*

*puntos.append((xn, yn))*

*print(xn, yn)*

*return xn, yn*

*def pintar(puntos):*

*'''Pinta la gráfica'''*

*x = []*

*y = []*

*for i in puntos:*

*x.append(i[0])*

*y.append(i[1])*

*plt.plot(x, y)*

*plt.show()*

*#main*

*x0 = float(input('Introduce la x inicial: ')) #punto inicial*

*y0 = float(input('Introduce la y inicial: ')) #punto inicial*

*xf = float(input('Introduce el extremo final: ')) #punto final*

*n = int(input('Número de divisiones: '))*

*h = (xf - x0)/n #intervalo pequeño*

*puntos = [] #lista de puntos*

*iterar(x0, y0, funcion, segunda\_derivada)*

*pintar(puntos)*