CENTRO DE INVESTIGACIÓN E INNOVACIÓN EN TIC
MAESTRIA EN CIENCIAS DE DATOS E INFORMACIÓN
TAREA 4

07 de marzo de 2023

Docente: Dr. Juliho Castillo Colmenares

Autores:
David Aarón Ramirez Olmeda
Miguel Sánchez Domínguez
Ricardo López Cruz

En esta actividad se implementará el método de Newton para resolver dos ecuaciones no lineales: $f(x) = x^2 - 4$ y $g(x) = x^2$. Se calcularán las soluciones aproximadas de cada ecuación utilizando el método de Newton con una tolerancia de 10^{-9} y se determinará el número de iteraciones necesarias para alcanzar la tolerancia deseada. Además, se creará una visualización de los resultados obtenidos para cada ecuación. Finalmente, se comparará el número de iteraciones necesarias y la velocidad de convergencia en ambos casos. El objetivo de esta actividad es familiarizarse con el método de Newton y su aplicación a ecuaciones no lineales, así como comprender la importancia de la elección de la función y del punto de inicio en la eficacia del método.

Inciso A

Se implementará el método de Newton para resolver la ecuación $f(x) = 0$ con $f(x) = x^2 - 4$, utilizando $x_0 = 3$. El método de Newton se define por la siguiente iteración:

$$x_{n+1} = x_n - [f(x_n)/f'(x_n)]$$

In [1]:
```python
def newton_method(f, f_prime, x0, tol):
    x = x0
    iterations = 0
    print("x" + str(iterations) + " = " + str(x))
    while abs(f(x)) > tol:
        x = x - f(x)/f_prime(x)
        iterations += 1
        print("x" + str(iterations) + " = " + str(x))
    return x, iterations
```

In [2]:
```python
def f(x):
    return x**2 - 4

def f_prime(x):
    return 2*x

x0 = 3
tol = 1e-9

solution, iterations = newton_method(f, f_prime, x0, tol)

print("La solución es x = " + str(solution))
print("Se necesitaron " + str(iterations) + " iteraciones.")
```

```
x0 = 3
x1 = 2.1666666666666665
x2 = 2.0064102564102564
x3 = 2.0000102400262145
x4 = 2.000000000026214
La solución es x = 2.000000000026214
Se necesitaron 4 iteraciones.
```

Inciso B

Se realizará una visualización de los resultados obtenidos en el inciso A.

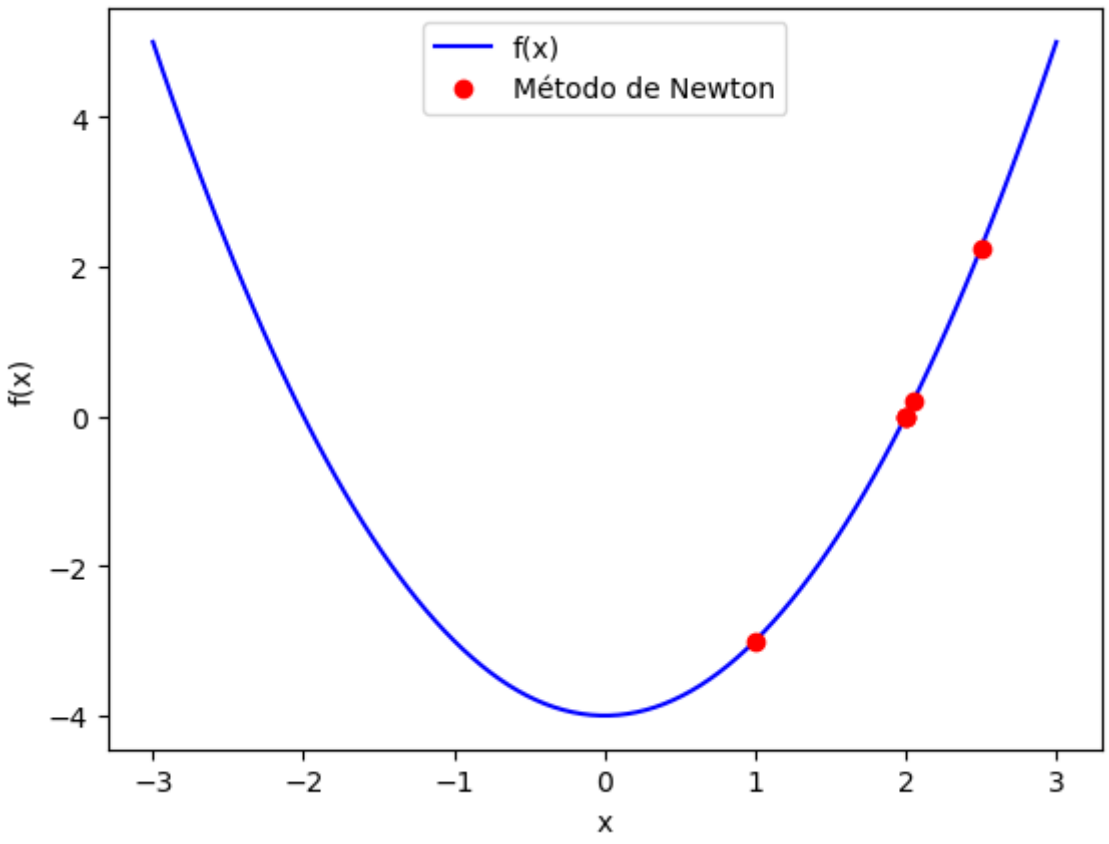
In [3]:
```python
import numpy as np
import matplotlib.pyplot as plt

def newton_method_g(f, f_prime, x0, tol):
    x = x0
    iterations = 0
    x_list = [x]
    while abs(f(x)) > tol:
        x = x - f(x)/f_prime(x)
        iterations += 1
        x_list.append(x)
    return x_list, iterations

x0 = 1
tol = 1e-9

x_list, iterations = newton_method_g(f, f_prime, x0, tol)

x = np.linspace(-3, 3, 100)
y = f(x)
plt.plot(x, y, 'b')
y_newton = f(np.array(x_list))
plt.plot(x_list, y_newton, 'ro')
plt.xlabel("x")
plt.ylabel("f(x)")
plt.legend(["f(x)", "Método de Newton"])
plt.show()
```



Este código generará un gráfico que muestra la función f(x) y las iteraciones del método de Newton en rojo:

Podemos ver que el método de Newton converge rápidamente a la solución exacta en x = 2. A medida que se realizan más iteraciones, los puntos rojos se acercan cada vez más a la intersección de la función con el eje x.

C

El método de Newton se utiliza para aproximar soluciones de ecuaciones de la forma f(x) = 0. En este caso, queremos aproximar la solución de g(x) = 0, donde g(x) = x^2.

Podemos encontrar la solución aproximada utilizando el método de Newton de la siguiente manera:

In [4]:
```
def g(x):
    return x**2

def g_prime(x):
    return 2*x

x0 = 1
tol = 1e-9

solution, iterations = newton_method(g, g_prime, x0, tol)

print("La solución es x = " + str(solution))
print("Se necesitaron " + str(iterations) + " iteraciones.")
```

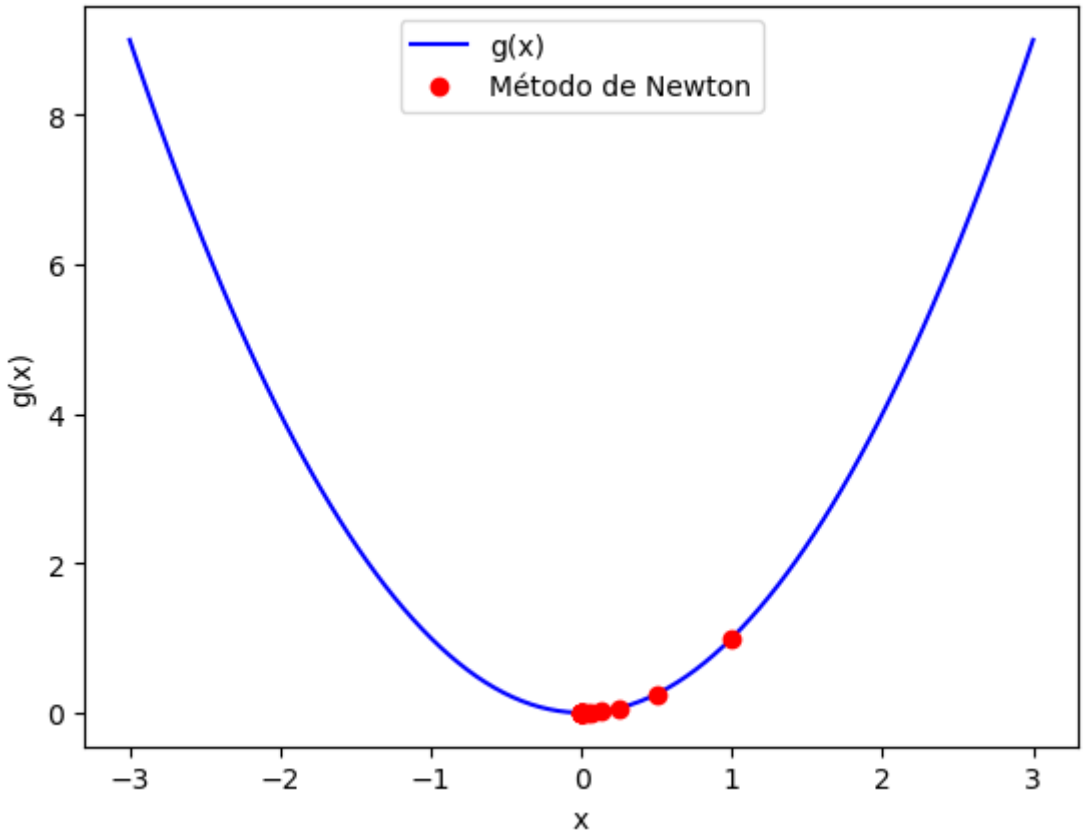
```
x0 = 1
x1 = 0.5
x2 = 0.25
x3 = 0.125
x4 = 0.0625
x5 = 0.03125
x6 = 0.015625
x7 = 0.0078125
x8 = 0.00390625
x9 = 0.001953125
x10 = 0.0009765625
x11 = 0.00048828125
x12 = 0.000244140625
x13 = 0.0001220703125
x14 = 6.103515625e-05
x15 = 3.0517578125e-05
La solución es x = 3.0517578125e-05
Se necesitaron 15 iteraciones.
```

D

In [5]:
```
x0 = 1

x_list, iterations = newton_method_g(g, g_prime, x0, tol)

y = g(x)
plt.plot(x, y, 'b')
y_newton = g(np.array(x_list))
plt.plot(x_list, y_newton, 'ro')
plt.xlabel("x")
plt.ylabel("g(x)")
plt.legend(["g(x)", "Método de Newton"])
plt.show()
```



E

La razón por la cual el método de Newton converge más rápidamente para la función $f(x)$ que para la función $g(x)$ es que la función $f(x)$ tiene una raíz doble en $x = 2$ y $x = -2$, lo que hace que el método de Newton converja rápidamente a la raíz. Por otro lado, la función $g(x)$ tiene una única raíz en $x = 0$, pero en la primera iteración el método de Newton se acerca a la raíz muy lentamente debido a la forma de la función y la elección del punto inicial x_0.

Por lo tanto, podemos concluir que el método de Newton converge en menos iteraciones para la función $f(x) = x^2 - 4$ que para la función $g(x) = x^2$, lo que demuestra que la velocidad de convergencia del método de Newton depende no solo de la forma de la función, sino también de

la elección del punto inicial x_0.