MÉTODO DE NEWTON–RAPHSON

Reporte Académico

Autor: Job Edward Apaza Curtihuanca

1. Introducción

El método de Newton–Raphson es uno de los métodos numéricos más eficientes para la aproximación de raíces de ecuaciones no lineales. Se basa en el uso de la derivada de la función para construir una sucesión de aproximaciones que converge rápidamente hacia la raíz buscada.

Gracias a su convergencia cuadrática, este método es ampliamente utilizado en problemas de ingeniería, ciencias aplicadas y análisis computacional, siempre que se disponga de una buena aproximación inicial.

2. Objetivos

2.1. Objetivo general

Aplicar el método de Newton–Raphson para aproximar raíces de ecuaciones no lineales mediante un proceso iterativo eficiente.

2.2. Objetivos específicos

- Analizar el fundamento matemático del método.
- Describir el algoritmo iterativo.
- Evaluar la convergencia del método.
- Resolver un problema práctico.

3. Fundamentación teórica

Sea una función $f(x)$ derivable en un intervalo que contiene la raíz α. El método de Newton–Raphson se basa en la aproximación lineal de la función mediante su recta tangente en un punto inicial x_0.

La intersección de la recta tangente con el eje x proporciona una mejor aproximación de la raíz.

4. Fórmula iterativa

La fórmula general del método de Newton–Raphson está dada por:

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)} \tag{1}$$

Este proceso se repite hasta que se cumpla el criterio de parada.

5. Algoritmo del método

1. Elegir un valor inicial x_0 cercano a la raíz.
2. Calcular $f(x_0)$ y $f'(x_0)$.
3. Aplicar la fórmula iterativa.
4. Verificar el criterio de parada.
5. Repetir el proceso.

6. Criterios de parada

El método se detiene cuando se cumple al menos uno de los siguientes criterios:

- $|x_{n+1} - x_n| < \varepsilon$
- $|f(x_{n+1})| < \varepsilon$
- Se alcanza el número máximo de iteraciones.

7. Convergencia

El método de Newton–Raphson presenta convergencia cuadrática, es decir:

$$|x_{n+1} - \alpha| \approx C|x_n - \alpha|^2 \tag{2}$$

donde α es la raíz exacta y C es una constante positiva.

8. Ventajas y desventajas

8.1. Ventajas

- Convergencia rápida.

- Alta precisión.
- Amplio uso en aplicaciones científicas.

8.2. Desventajas

- Requiere el cálculo de derivadas.
- Sensible al valor inicial.
- Puede divergir si $f'(x_n) = 0$.

9. Ejemplo de aplicación

Sea la función:

$$f(x) = x^3 - x - 2 \tag{3}$$

Su derivada es:

$$f'(x) = 3x^2 - 1 \tag{4}$$

Con un valor inicial $x_0 = 1{,}5$, se obtiene la siguiente aproximación iterativa:

Iteración	x_n	$f(x_n)$	x_{n+1}
1	1.500	-0.125	1.521
2	1.521	0.002	1.521

La raíz aproximada es:

$$x \approx 1{,}521$$

10. Implementación computacional

A continuación se deja el espacio reservado para la implementación del método de Newton–Raphson en un lenguaje de programación (por ejemplo, Python):

```
# M todo de Newton-Raphson donde el usuario escribe la funci n

print("=== M TODO DE NEWTON-RAPHSON ===")
print("Ejemplo: para x^3 - 4*x + 1 escribe: x**3 - 4*x + 1\n")


funcion_str = input("Ingresa la funci n f(x): ")
derivada_str = input("Ingresa la derivada f'(x): ")
```

```python
# Convierte las funciones escritas en funciones reales
def f(x):
    return eval(funcion_str)

def f_derivada(x):
    return eval(derivada_str)

# Valor inicial
x0 = float(input("Ingresa el valor inicial (x0): "))

# Par metros del m todo
tolerancia = 0.0001
max_iter = 20
iteracion = 0

print("\nIteraci n |       x       |     f(x)     ")
print("----------------------------------------")

while iteracion < max_iter:
    fx = f(x0)
    fpx = f_derivada(x0)

    # Evitar divisi n entre cero
    if fpx == 0:
        print("Error: derivada igual a cero, no se puede continuar.")
        break

    # F rmula de Newton-Raphson
    x1 = x0 - fx / fpx

    print(f"{iteracion+1:9d} | {x1:12.4f} | {fx:12.4f}")

    # Verificar si ya se alcanz  la tolerancia
    if abs(x1 - x0) < tolerancia:
        print("\nRa z aproximada encontrada: {:.4f}".format(x1))
        print("Iteraciones totales:", iteracion + 1)
        break

```

```
    x0 = x1
    iteracion += 1

else:
    print("\nNo se encontr  la ra z dentro del n mero m ximo de iteraciones.")
```

11. Aplicaciones

El método de Newton–Raphson se utiliza ampliamente en ingeniería, física, economía, optimización y análisis numérico para resolver ecuaciones no lineales de alta complejidad.

12. Conclusiones

El método de Newton–Raphson es una de las técnicas más eficientes para la resolución de ecuaciones no lineales. Su convergencia cuadrática permite obtener soluciones precisas en pocas iteraciones; sin embargo, requiere una adecuada selección del valor inicial y el cálculo de derivadas para garantizar su correcta aplicación.