

**Universidad Nacional del Altiplano**  
**Facultad de Ingeniería Estadística e Informática**

**Alumna:** Leydy Griselda Aguilar Ccopa

**Docente:** Ing. Torres Cruz Fred

**Trabajo N° 6 – Método de la Secante**

El método de la secante es un procedimiento numérico usado para encontrar raíces de una ecuación no lineal  $f(x)=0$ .

Es parecido al método de Newton-Raphson, pero no necesita derivadas, lo que lo hace más práctico en muchos casos.

En lugar de usar la tangente (como hace Newton-Raphson), este método usa una línea secante que pasa por dos puntos del gráfico de la función  $f(x)$ .

Esa recta corta el eje X en una posición que se toma como una mejor aproximación de la raíz.

Si tenemos dos aproximaciones iniciales  $x_0$  y  $x_1$ , calculamos las siguientes iteraciones usando:

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

donde:

- $x_{n-1}$  y  $x_n$  son las dos últimas aproximaciones conocidas,
- $x_{n+1}$  es la nueva aproximación de la raíz.

CODIGO:

# Método de la Secante

```
def secante(f, x0, x1, tol=1e-6, max_iter=100):
```

```
    """
```

```
    Encuentra la raíz usando el método de la secante
```

```
    f: función
```

```
    x0, x1: dos valores iniciales
```

```
    tol: tolerancia
```

```
    max_iter: número máximo de iteraciones
```

```
"""
```

```
print("\n=== MÉTODO DE LA SECANTE ===")
```

```
print(f"{'Iter':<6} {'x_n-1':<15} {'x_n':<15} {'x_n+1':<15} {'f(x_n+1)':<15} {'Error':<15}")
```

```
print("-" * 95)
```

```
for i in range(max_iter):
```

```
    fx0 = f(x0)
```

```
    fx1 = f(x1)
```

```
    # Evitar división por cero
```

```
    if abs(fx1 - fx0) < 1e-12:
```

```
        print("\nError: División por cero")
```

```
        return x1
```

```
    # Fórmula de la secante
```

```
    x2 = x1 - fx1 * (x1 - x0) / (fx1 - fx0)
```

```
    fx2 = f(x2)
```

```
    error = abs(x2 - x1)
```

```
    print(f"{'i+1':<6} {'x0':<15.8f} {'x1':<15.8f} {'x2':<15.8f} {'fx2':<15.8e} {'error':<15.8e}")
```

```
    if error < tol or abs(fx2) < tol:
```

```
        print(f"\nConvergió en {i+1} iteraciones")
```

```
        print(f"Raíz aproximada: x = {x2:.8f}")
```

```
return x2
```

```
# Actualizar valores
```

```
x0 = x1
```

```
x1 = x2
```

```
print(f"\nNo convergió en {max_iter} iteraciones")
```

```
return x2
```

```
# Ejemplo de uso: Encontrar raíz de  $x^2 - 2 = 0$ 
```

```
def f(x):
```

```
    return x**2 - 2
```

```
# Ejecutar
```

```
raiz = secante(f, 1, 2)
```

```
----- RESIARI: C:/USERS/USER/DOCUMENTS/PROGRAMACION NUMERICA.

=== MÉTODO DE LA SECANTE ===
Iter  x_n-1      x_n      x_n+1      f(x_n+1)      Error
-----
1      1.00000000  2.00000000  1.33333333  -2.22222222e-01  6.66666667e-01
2      2.00000000  1.33333333  1.40000000  -4.00000000e-02  6.66666667e-02
3      1.33333333  1.40000000  1.41463415  1.18976800e-03  1.46341463e-02
4      1.40000000  1.41463415  1.41421144  -6.00728684e-06  4.22707867e-04
5      1.41463415  1.41421144  1.41421356  -8.93145558e-10  2.12358245e-06

Convergió en 5 iteraciones
Raiz aproximada: x = 1.41421356
>
```