

### Tarea 3 - Método de integración de Romberg

El método de integración de Romberg permite calcular la integral de una función para un intervalo definido. El algoritmo de solución busca, de manera óptima, partir la función en subintervalos, hacer una aproximación de la función usando la extrapolación de Richardson en el subintervalo y luego calculando el área, utilizando la regla del trapecio iterativamente, se determina el valor de la integral aproximada para el intervalo  $[a, b]$  como la suma de las áreas bajo de la curva calculadas de manera aproximada en cada subintervalo.

#### Algoritmo

Sea  $N$  un entero positivo, en el intervalo  $[a, b]$   $a, b \in \mathbb{R}$  y  $f(x)$  una función de una variable, el algoritmo de Romberg calcula de manera aproximada la solución de  $I(f) = \int_a^b f(x) dx$ , con una precisión de orden  $O(h^{2N})$ .

#### Inicio

$$h = b - a$$

**para**  $j = 1, 2, \dots, N$  **hacer**

$$R_{j,1} = \frac{h}{2} \left[ f(a) + 2 \sum_{j=1}^{2^{j-1}-1} f(a + jh) + f(b) \right] \text{ (Regla del trapecio)}$$

**para**  $k = 2, 3, \dots, j$  **hacer**

$$R_{j,k} = R_{j,k-1} + \frac{R_{j,k-1} - T_{j-1,k-1}}{4^{k-1} - 1} \text{ (Extrapolación de Richardson)}$$

**terminar**

$$h = h/2$$

**terminar**

#### Fin

El resultado del algoritmo es una matriz con los valores de la aproximación de Romberg para distintos tamaños de intervalo; se toma como base un tamaño de paso para la integración  $h = b - a$  luego, el tamaño de paso cambia según el orden de aproximación y se lleva a cabo el proceso de extrapolación.

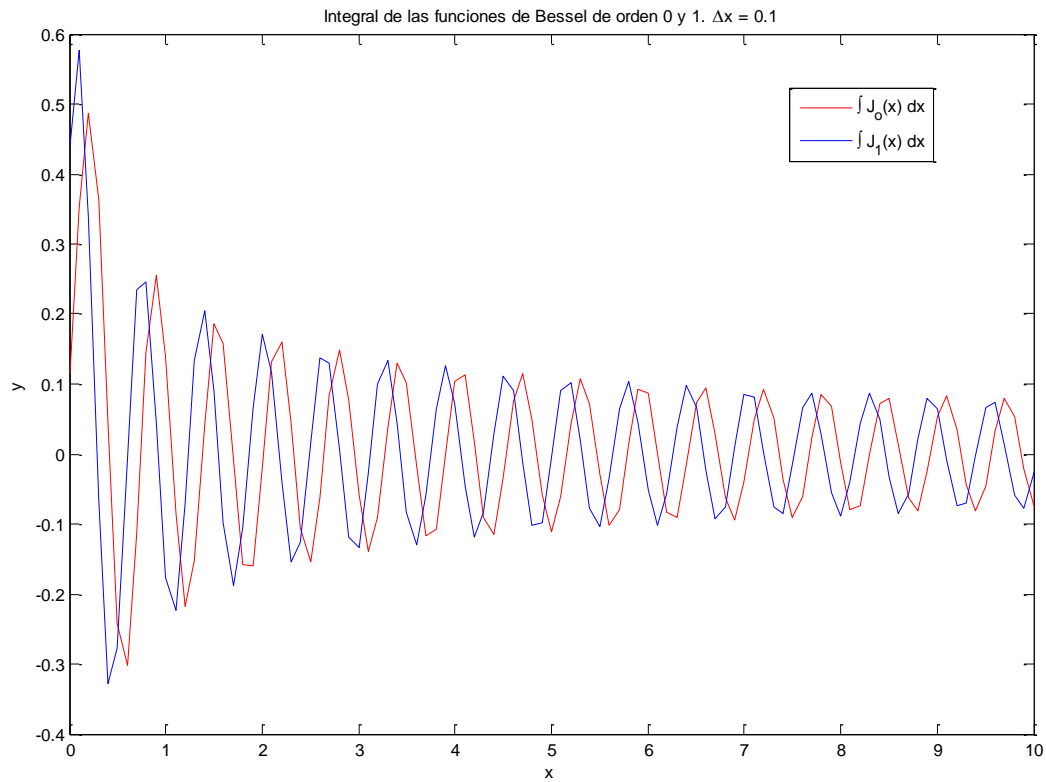


Figura 1. Resultado del algoritmo de Romberg para funciones de Bessel de orden 0 y 1.

Después de implementar el algoritmo de Romberg para una dimensión, se calculan las integrales de las funciones de Bessel de orden 0 y 1, para el intervalo  $x = [0, 1]$  con pasos  $\Delta x = 0.1$ . Se utiliza una rutina en Matlab que utiliza la función base descrita anteriormente y almacena el valor al que converge el método numérico, como se muestra a continuación:

```
% Definición de variables
x = 0:0.1:10; % Vector x con dx = 0.1
n = [0 1]; % Órdenes de la función de Bessel J_n(x)

for j=1 : length(n)
    for i=1 : length(x)
        % Definición de la función explícita de Matlab f(x)
        f = inline(sprintf('cos(0.1f*sin(x)-%d*x)/pi', i, j));
        % Cálculo de la matriz de Romberg con resultado de precisión 1e-6.
        R = romberg(f, 0, pi, 15, 1e-6)';
        % Se almacena el valor de la integral para la función de Bessel de
        % orden j, en x = x(i).
        % Se toma el último valor de la Matriz de Romberg, i.e., al que
        % converge el algoritmo.
        int_val(j,i) = R(size(R,1), size(R,2));
    end
end
plot(x, int_val(1,:)) % Gráficas
plot(x, int_val(2,:), 'r')
```

# Anexos

---

## Código en Matlab

```
% Algoritmo de Integración de Romberg
% Entradas
% f - Función explícita de Matlab. e.g., f = @(x) x/(1-x.^2)
% a,b - límites del intervalo de integración.
% n - número de aproximaciones a calcular; define la dimensión de la matriz
% de Romberg ->  $R \in \mathbb{R}^{(n \times n)}$ .
% tol - error admisible para la aproximación de Romberg.
% Salidas:
% r - Matriz de Romberg con valores aproximados de la integral

function R = romberg(f, a, b, n, tol)

    % Tamaños de intervalo desde aproximación de orden  $O(h)$  hasta la
    % aproximación de orden  $O(h^{(2n)})$ 
    h = (b - a) ./ (2.^(0:n-1));

    % Inicialización del primer elemento de la matriz de Romberg
    R(1,1) = (b - a) * (f(a) + f(b)) / 2;

    % Ciclo principal
    for j=2 : n
        % Cálculo iterativo de área con regla del trapecio
        suma_trapecio = 0;
        for i = 1:2^(j-2)
            suma_trapecio = suma_trapecio + f(a + (2 * i - 1) * h(j));
        end
        % Se guarda el valor de la integral para la aproximación j-ésima de
        % primer orden
        R(j,1) = R(j-1,1) / 2 + h(j) * suma_trapecio;

        % Extrapolación de Richardson
        % Se utiliza la extrapolación de Richardson para determinar con mayor
        % precisión y velocidad computacional el valor exacto de la integral
        for k = 2:j
            R(j,k) = (4^(k-1) * R(j,k-1) - R(j-1,k-1)) / (4^(k-1) - 1);
        end

        % Se verifica como criterio de convergencia la diferencia entre los
        % últimos valores calculados de la matriz de Romberg, como lo indica el
        % libro texto.
        if abs(R(j,j) - R(j-1, j-1)) < tol
            return
        end
    end
end
```