

# Métodos Numéricos

## ¿Qué son los métodos numéricos?

Son técnicas que permiten **aproximar soluciones** de problemas matemáticos que no se pueden resolver de forma exacta o cuya solución exacta sería demasiado complicada.

El objetivo es obtener una respuesta **suficientemente buena** mediante **iteraciones**.

Se aplican en:

Raíces de ecuaciones

Sistemas de ecuaciones

Integración aproximada

Derivación numérica

Interpolación

Ajuste de curvas

Ecuaciones diferenciales

Los métodos numéricos son procedimientos sistemáticos que permiten obtener **soluciones aproximadas** a problemas matemáticos cuya solución exacta es difícil, imposible o poco práctica de calcular. Su fortaleza radica en que trabajan por **iteraciones**, refinando la aproximación hasta obtener un resultado con la precisión deseada.

En la vida real, muchos fenómenos no tienen soluciones exactas, pero sí se pueden resolver numéricamente. Por ejemplo: encontrar raíces de ecuaciones complicadas, integrar funciones que no poseen integral elemental, derivar datos experimentales, y estimar valores intermedios entre puntos conocidos.

Cuando usamos métodos numéricos, no buscamos una solución “perfecta”, sino una solución **suficientemente buena**.

Para entender qué tan buena es esa solución, necesitamos dos ideas fundamentales: **exactitud** y **precisión**. Aunque parecen lo mismo, no lo son. Entender la diferencia evita errores de interpretación y ayuda a evaluar la calidad de cualquier resultado numérico.

## ¿Qué es la exactitud?

La exactitud indica **qué tan cerca está un resultado del valor verdadero**.

Si el valor real es 10 y nuestro método produce 9.99, el resultado es **muy exacto**.

Si obtenemos 8, el resultado es **poco exacto**.

La exactitud compara directamente contra la realidad.

Un resultado puede ser muy preciso pero poco exacto si está sistemáticamente desplazado del valor real.

## Ejemplo cotidiano:

Un reloj que siempre marca 5 minutos adelantado **es preciso**, porque avanza de forma uniforme, pero **no es exacto**, porque nunca marca la hora correcta.

## 2. ¿Qué es la precisión?

La precisión indica **la consistencia de los resultados**, es decir, qué tan parecidos son entre sí.

Si repetimos un cálculo y obtenemos 9.80, 9.81, 9.79, 9.82, tenemos **alta precisión**.

Si obtenemos 9.80, 12.4, 6.7, 15.0, tenemos **baja precisión**, aunque uno de esos valores pudiera coincidir con el correcto por casualidad.

La precisión describe la “dispersión” o variación entre resultados repetidos.

### La diferencia clave

**Exactitud** → cercanía al valor verdadero.

**Precisión** → repetibilidad del resultado.

Un método numérico puede tener:

Alta exactitud y alta precisión → objetivo ideal.

Alta precisión pero baja exactitud → resultados muy parecidos entre sí, pero lejos del valor real.

Baja precisión pero alta exactitud → a veces acierta por suerte, pero es inconsistente.

Baja precisión y baja exactitud → resultados dispersos y alejados del valor real.

### Analogía didáctica: tirando flechas a un blanco

Imagina un blanco

Las flechas caen **todas cerca del centro**

→ alta exactitud y alta precisión.

Las flechas caen **agrupadas pero alejadas del centro**

→ alta precisión, baja exactitud.

Las flechas **están cerca del centro pero muy dispersas**

→ alta exactitud, baja precisión.

Las flechas **dispersas y además lejos del centro**

→ baja exactitud y baja precisión.

Esta analogía ayuda a visualizar inmediatamente la diferencia.

### ¿Por qué importa esto en métodos numéricos?

Porque cada resultado numérico está afectado por:

errores de redondeo,  
truncamientos,  
iteraciones finitas,  
modelos aproximados,  
datos experimentales imperfectos.

Saber distinguir entre exactitud y precisión permite:

evaluar si un método converge correctamente,  
detectar resultados engañosos,  
decidir si necesitamos más iteraciones,  
estimar cuántas cifras significativas son confiables,  
evitar conclusiones incorrectas por interpretaciones superficiales.

La exactitud y la precisión son la base para comprender la calidad de cualquier cálculo numérico.

Un resultado puede parecer “bueno” a primera vista, pero solo comparando su **cercanía al valor real** (exactitud) y su **consistencia interna** (precisión), podemos juzgar si realmente es confiable.

### Gráficas didácticas: Exactitud y Precisión

A continuación se muestran cuatro ilustraciones que explican de manera visual las combinaciones posibles entre *exactitud* y *precisión*.

#### 1. Alta exactitud y alta precisión

(Flechas agrupadas cerca del centro)



#### Interpretación:

Los resultados caen prácticamente en el valor real y además son consistentes entre sí.  
→ Es el comportamiento ideal de un método numérico.

#### ¿Por qué fue necesario desarrollar los métodos numéricos?

Los métodos numéricos surgieron por una razón muy simple: **muchos problemas reales no tienen solución exacta, o bien su solución exacta es tan complicada que resulta inútil en la práctica.**

La matemática pura proporciona fórmulas exactas para algunos casos, pero la vida real plantea ecuaciones, modelos y datos que exceden esas posibilidades.

Por eso se necesitó una forma de obtener **soluciones aproximadas**, confiables y rápidas, aunque no fueran perfectas.

A continuación las razones fundamentales:

### 1. Existen ecuaciones que **NO** pueden resolverse de forma algebraica

Muchas ecuaciones no tienen solución cerrada. Ejemplos clásicos:

$$x = \cos x$$

$$e^{-x} = x$$

$$x^5 - 3x + 1 = 0$$

O incluso ecuaciones sencillas de física, economía o ingeniería que involucran funciones trascendentes, combinaciones no lineales o modelos complejos.

El desarrollo de métodos numéricos permitió **resolverlas sin depender de fórmulas**.

### 2. Grandes sistemas de ecuaciones serían imposibles de resolver a mano

En ingeniería y física aparecen sistemas con decenas, cientos o miles de ecuaciones:

análisis estructural

redes eléctricas

modelos de fluidos

optimización

ecuaciones diferenciales acopladas

Resolverlos manualmente es impráctico; incluso con métodos exactos se necesitarían horas o días.

Los métodos numéricos permiten soluciones **rápidas y escalables**.

### 3. Muchas integrales **NO** tienen antiderivada conocida

Por ejemplo:

$$\int e^{-x^2} dx$$

$$\int \ln(1 + x^2) dx$$

Integrales definidas de funciones medidas experimentalmente

En estos casos se necesitó un modo de obtener el área bajo la curva **sin encontrar la función primitiva**.

Por eso surgieron métodos como trapecio, Simpson, etc.

#### 4. Los datos reales son aproximados e incompletos

En mediciones:

hay ruido,  
límites instrumentales,  
errores inevitables,  
datos faltantes o irregulares.

Los métodos numéricos permiten:

suavizar,  
interpolar,  
estimar,  
predecir,  
ajustar curvas a los datos.

Sin ellos, la ciencia experimental no podría transformar datos imperfectos en conclusiones válidas.

#### 5. Las computadoras requieren algoritmos, no fórmulas exactas

Una computadora no “entiende” una solución simbólica; solo puede ejecutar **procedimientos repetitivos**.

Los métodos numéricos se adaptan perfectamente:

operaciones simples  
miles de iteraciones  
cálculos rápidos  
resultados controlados por tolerancias de error

Por eso, cuando aparecieron las computadoras, los métodos numéricos se volvieron indispensables.

#### 6. Permiten resolver problemas demasiado grandes para la matemática exacta

Ejemplos:

simulación del clima  
trayectorias espaciales  
diseño de aviones  
procesamiento de señales (FFT, filtrado, etc.)  
modelos financieros  
simulación de epidemias  
análisis estructural tridimensional

Ninguno de estos problemas puede resolverse con lápiz y papel.  
Necesitan aproximaciones **rápidas, estables y eficientes**.

### **Conclusión didáctica**

Los métodos numéricos no surgieron como un lujo académico, sino como una **necesidad práctica**.

La realidad es demasiado compleja para depender solo de soluciones exactas.

Los métodos numéricos permiten:

aproximar lo que no puede resolverse simbólicamente,  
manejar grandes volúmenes de información,  
controlar el error,  
usar la potencia de las computadoras,  
resolver problemas reales con precisión suficiente.

En resumen:

**Los métodos numéricos existen porque el mundo no es exacto, pero sí necesita respuestas.**

### **Conclusión**

Los métodos numéricos no surgieron como un lujo académico, sino como una **necesidad práctica**.

La realidad es demasiado compleja para depender solo de soluciones exactas.

Los métodos numéricos permiten:

aproximar lo que no puede resolverse simbólicamente,  
manejar grandes volúmenes de información,  
controlar el error,  
usar la potencia de las computadoras,  
resolver problemas reales con precisión suficiente.

En resumen:

**Los métodos numéricos existen porque el mundo no es exacto, pero sí necesita respuestas.**

## ¿Qué es el error en métodos numéricos?

En métodos numéricos, el **error** es la diferencia entre el **valor verdadero** y el **valor aproximado** que obtenemos mediante un método.

$$\text{Error} = \text{Valor verdadero} - \text{Valor aproximado}$$

El error es inevitable porque:

usamos números con precisión limitada,  
hacemos aproximaciones paso a paso,  
algunos cálculos requieren truncar series o expresiones infinitas,  
y muchas veces los datos iniciales no son exactos.

El objetivo de un buen método numérico no es eliminar el error (imposible), sino **controlarlo** y **mantenerlo lo suficientemente pequeño**.

### 1. Error de Truncamiento

El **error de truncamiento** ocurre cuando se **acorta un proceso matemático infinito** para obtener una forma más simple y manejable.

Ejemplos típicos:

#### Series infinitas

$$e^x = 1 + x + \frac{x^2}{2!} + \dots$$

Si la cortamos después de unos términos, se pierde información → **error de truncamiento**.

#### Derivación e integración numérica

Las fórmulas son aproximaciones basadas en **expansiones de Taylor truncadas**.

Ejemplo básico:

$$f'(x) \approx \frac{f(x+h) - f(x)}{h}$$

Esa aproximación omite términos de orden superior → **truncamiento**.

#### Métodos iterativos

En bisección, Newton, secante, etc., la raíz exacta no se obtiene; se detiene el proceso → **error por truncar la iteración**.

### Resumen:

El error de truncamiento aparece cuando simplificamos una expresión o un proceso para hacerlo utilizable.

## 2. Error de Redondeo

El **error de redondeo** surge porque la computadora **no puede representar todos los números reales exactamente**.

Solo maneja un número limitado de cifras significativas.

Ejemplos claros:

El número  $\pi$  se guarda como 3.14159265... (cifras limitadas).

Fracciones como  $1/3$  nunca se representan exactamente.

Operaciones repetidas amplifican el error acumulado.

Este error aparece en:

sumas y restas con números muy grandes o muy pequeños,  
multiplicaciones largas,  
conversión decimal  $\leftrightarrow$  binario,  
cualquier cálculo realizado en hardware digital.

### Resumen:

El error de redondeo proviene de la limitada capacidad de representación numérica en las computadoras.

## 3. Error en los Datos (Error de medición)

Ocurre cuando los **datos de entrada ya contienen errores** desde su origen, ya sea por:  
imprecisión de instrumentos,  
ruido,  
errores humanos,  
redondeos previos,  
promedios mal calculados.

Ejemplo:

El sensor marca 2.47 cm, pero el valor real es 2.468 cm  $\rightarrow$  ese error pasa directo al cálculo numérico. Los métodos numéricos **no pueden corregir errores de entrada**, solo trabajar con ellos.

### Resumen:

Si los datos no son exactos, los resultados tampoco lo serán. Un método numérico solo puede ser tan bueno como los datos que recibe.

## Error Absoluto

El **error absoluto** mide *cuánto nos equivocamos en valor absoluto*, sin importar la dirección.

$$\text{Error absoluto} = \text{Valor verdadero} - \text{Valor aproximado}$$

### Ejemplo con recta numérica

Supongamos que el valor real es **10 cm**, pero el instrumento marca **9.6 cm**.

$$E_{\text{abs}} = |10 - 9.6| = 0.4 \text{ cm}$$

### Interpretación:

La medición se equivocó **0.4 cm**, independientemente de que el instrumento midió de menos.

## Error Relativo

El **error relativo** indica qué tanto representa ese error absoluto comparado con el valor verdadero. Es un indicador de **proporción del error**.

$$\text{Error relativo} = \frac{\text{Error absoluto}}{\text{Valor verdadero}}$$

Se suele expresar en **decimal** o **porcentaje**.

### Ejemplo con recta numérica

Usando el caso anterior:

Valor real: 10 cm

Valor medido: 9.6 cm

Error absoluto: 0.4 cm

$$E_{\text{abs}} = |10 - 9.6| = 0.4 \text{ cm}$$

### Interpretación:

La medición tuvo un error del **4%** respecto al valor verdadero.

### Ejemplo práctico

**Error relativo:**

$$10/250 = 0.04 = 4\%$$

### Conclusión:

El error es pequeño comparado con el tamaño de la cantidad real.

#### 4. ¿Por qué es importante distinguir ambos errores?

Error absoluto

Mide **cuánto** nos equivocamos.

Se mide en mismas unidades del fenómeno.

Útil cuando las magnitudes son similares.

Error relativo

Mide **qué tan grande** es el error en relación al valor real.

Se expresa en porcentaje o número adimensional.

Útil cuando se comparan valores grandes o pequeños.

Ejemplo típico:

Un error de **1 cm** es enorme si mides un insecto, pero insignificante si mides un edificio.

Por eso el **error relativo** es tan importante.

#### MÉTODOS PARA ENCONTRAR RAÍCES DE ECUACIONES

Los siguientes métodos se utilizan para resolver ecuaciones de la forma:

$$f(x)=0$$

Cada uno tiene fortalezas y limitaciones. No existe “el mejor”, sino el más adecuado según el problema.

##### 1. Método de Bisección

**Idea principal:**

Divide un intervalo en dos y selecciona el subintervalo donde cambia el signo de la función.

**Fórmula:**

$$m = \frac{a + b}{2}$$

**Ventajas:**

Siempre converge si la función es continua.  
Método muy seguro.

**Desventajas:**

Lento.  
Requiere conocer un intervalo donde haya cambio de signo.

**Se usa cuando:**

Necesitas certeza de convergencia, aunque tome más iteraciones.

## Ejemplo

### Planteamiento del problema

Queremos resolver la ecuación:

$$f(x) = e^x - \sin(3x) - 2$$

Es decir, encontrar el valor (o valores) de  $x$  tales que:

$$f(x) = e^x - \sin(3x) - 2 = 0$$

En general no es posible despejar  $x$  de forma algebraica, así que usamos **métodos numéricos**.

### Análisis inicial y elección del intervalo

Evaluamos la función en algunos puntos sencillos para ver dónde cambia de signo:

En  $x = 0$ :

$$f(0) = e^0 - \sin(0) - 2 = 1 - 0 - 2 = -1$$

En  $x = 0.5$ :

$$f(0.5) = e^{0.5} - \sin(1.5) - 2 \approx 1.6487 - 0.9975 - 2 \approx -1.3488$$

En  $x = 1$ :

$$f(1) = e^1 - \sin(3) - 2 \approx 2.7183 - 0.1411 - 2 \approx 0.5772$$

### Observación:

$$f(0.5) \approx -1.3488 < 0$$

$$f(1.0) \approx 0.5772 > 0$$

Por lo tanto, **en el intervalo [0.5,1.0] hay al menos una raíz**, porque la función pasa de negativa a positiva (Teorema del Valor Intermedio).

Tomaremos este intervalo para el método de bisección y un valor cercano (por ejemplo  $x_0 = 1$ ) para el método de Newton.

## Método de Bisección

### Idea del método

Dado un intervalo  $[a,b]$  con:

$$f(a) \cdot f(b) < 0$$

sabemos que hay una raíz dentro. El método de bisección:

Calcula el punto medio:

$$m = \frac{a + b}{2}$$

Evalúa  $f(m)$ .

Si  $f(a) \cdot f(m) < 0$ , la raíz está en  $[a, m]$ , así que ponemos  **$b = m$** .

Si  $f(m) \cdot f(b) < 0$ , la raíz está en  $[m, b]$ , así que ponemos  **$a = m$** .

Repetimos hasta que el intervalo sea muy pequeño o  **$|f(m)|$**  sea menor que una tolerancia.

### Iteraciones de bisección para $[a,b] = [0.5,1.0]$

Usamos una tolerancia, por ejemplo  $\epsilon = 10^{-4}$ .

A continuación se muestra el desarrollo de las primeras iteraciones (valores redondeados a 6 decimales):

## 2. Método de Newton–Raphson

### Idea principal:

Utiliza la recta tangente en el punto actual para aproximar la raíz.

### Fórmula:

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

### Se usa cuando:

La función es suave y se conoce una buena aproximación inicial.

## 3. Método de la Secante

### Idea principal:

Aproxima la derivada usando dos puntos consecutivos.

### Fórmula:

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

### Ventajas:

No necesita derivada.

Rápido, más que bisección.

### Desventajas:

Puede fallar si el denominador se acerca a cero.

No siempre es tan estable como Newton.

### Se usa cuando:

No es posible calcular la derivada, pero sí evaluar la función.

It	$a$	$b$	$m = (a + b)/2$	$f(a)$	$f(b)$	$f(m)$
1	0.500000	1.000000	0.750000	-1.348774	0.577162	-0.661073
2	0.750000	1.000000	0.875000	-0.661073	0.577162	-0.095045
3	0.875000	1.000000	0.937500	-0.095045	0.577162	0.230405
4	0.875000	0.937500	0.906250	-0.095045	0.230405	0.064669
5	0.875000	0.906250	0.890625	-0.095045	0.064669	-0.015982
6	0.890625	0.906250	0.898438	-0.015982	0.064669	0.024150
7	0.890625	0.898438	0.894531	-0.015982	0.024150	0.004035
8	0.890625	0.894531	0.892578	-0.015982	0.004035	-0.005986
9	0.892578	0.894531	0.893555	-0.005986	0.004035	-0.000979
10	0.893555	0.894531	0.894043	-0.000979	0.004035	0.001527
11	0.893555	0.894043	0.893799	-0.000979	0.001527	0.000274
12	0.893555	0.893799	0.893677	-0.000979	0.000274	-0.000352

En la iteración 12:

$$m_{12} \approx 0.893555$$

$f(m_{12}) \approx -0.000352$ , que ya es **muy pequeño** en valor absoluto.

Podemos tomar como aproximación de la raíz por bisección:

$$X \approx 0.8937(\text{bisección}, \varepsilon \approx 10^{-4})$$

## 4. Método de Falsa Posición (Regla Falsa o *Regula Falsi*)

### Idea principal:

Como bisección, pero en vez del punto medio, usa una recta secante que cruza por el eje x.

### Fórmula:

$$x = b - f(b) \frac{b - a}{f(b) - f(a)}$$

### Ventajas:

Más rápido que bisección.

Conserva la seguridad del intervalo con cambio de signo.

### Desventajas:

Puede quedarse "pegado" en un extremo del intervalo.

### Se usa cuando:

Se requiere seguridad de bisección, pero con mayor velocidad.

## 5. Método del Punto Fijo

### Idea principal:

Reescribe la ecuación como:

$$x = g(x)$$

y se itera:

$$x_{n+1} = g(x_n)$$

### Ventajas:

Conceptualmente simple.

### Desventajas:

Muy sensible a la elección de  $g(x)$ .

Puede divergir fácilmente.

### Se usa cuando:

La ecuación permite una transformación estable.

## 6. Método de Müller

### Idea principal:

Utiliza un polinomio cuadrático que pasa por tres puntos para aproximar la raíz.

### Ventajas:

Funciona con funciones complejas.

No necesita derivadas.

### Desventajas:

Más pesado computacionalmente.

Puede producir raíces complejas aunque no existan en el intervalo.

### Se usa cuando:

La función es complicada y se requiere robustez sin derivadas.

## 7. Métodos basados en interpolación (como *Brent*)

### Idea principal:

Combinan lo mejor de bisección, secante y falsa posición para lograr rapidez y seguridad.

### Ventajas:

De los métodos más confiables y rápidos.

Estándar en muchas librerías científicas.

### Desventajas:

Más difícil de explicar matemáticamente.

### Se usa cuando:

Se necesita un método profesional, híbrido y muy seguro.

## Método de Newton–Raphson

### Idea geométrica

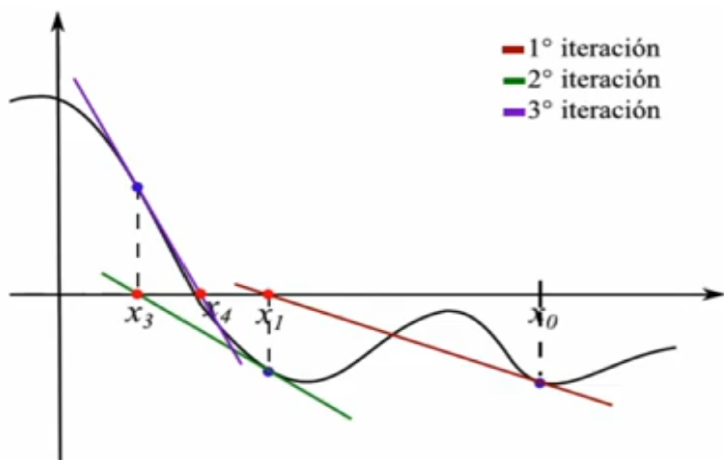
Partes de un valor inicial  $x_0$ .

Dibujas la **recta tangente** a la curva  $y = f(x)$  en el punto  $(x_0, f(x_0))$ .

Ves dónde esa tangente corta al eje  $x$ ; ese punto es el siguiente valor  $x_1$ .

Repites el proceso: con  $x_1$  haces otra tangente, obtienes  $x_2$ , etc.

Cada vez “bajas” por la tangente hasta acercarte más a la raíz.



### De ahí sale la fórmula

La ecuación de la recta tangente en  $x_n$ :

$$y = f(x_n) + f'(x_n)(x - x_n)$$

La intersección con el eje  $x$  se da cuando  $y = 0$ :

$$0 = f(x_n) + f'(x_n)(x_{n+1} - x_n)$$

Despejando  $x_{n+1}$ :

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

Esa es la fórmula de Newton–Raphson.

### Algoritmo paso a paso

Elegir valor inicial  $x_0$ .

Repetir:

Calcular  $f(x_n)$ .

Calcular  $f'(x_n)$ .

Actualizar:

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

Parar cuando  $|x_{n+1} - x_n|$  o  $|f(x_{n+1})|$  sea menor que una tolerancia.

**Ejemplo detallado con  $f(x) = x^2 - 2$**

Función:

$$f(x) = x^2 - 2$$

Derivada:

$$f'(x) = 2x$$

Tomamos  $x_0 = 1$ .

**Iteración 0 → 1**

$$f(1) = 1^2 - 2 = -1$$

$$f'(1) = 2 \cdot 1 = 2$$

$$x_1 = 1 - \frac{-1}{2} = 1 + 0.5 = 1.5$$

**Iteración 1 → 2**

$$f(1.5) = 1.5^2 - 2 = 2.25 - 2 = 0.25$$

$$f'(1.5) = 2 \cdot 1.5 = 3$$

$$x_2 = 1.5 - \frac{0.25}{3} \approx 1.5 - 0.083333 = 1.416666 \dots$$

**Iteración 2 → 3**

$$f(1.416666 \dots) \approx 1.416666^2 - 2 \approx 0.006944$$

$$f'(1.416666 \dots) \approx 2 \cdot 1.416666 = 2.833333$$

$$x_3 \approx 1.416666 - \frac{0.006944}{2.833333} \approx 1.4142157$$

**Iteración 3 → 4**

$$f(1.4142157) \approx 0.000006$$

$$f'(1.4142157) \approx 2.828431$$

$$x_4 \approx 1.4142157 - \frac{0.000006}{2.828431} \approx 1.41421356$$

Ya estamos prácticamente en  $\sqrt{2}$ .

En **4 iteraciones** pasamos de 1 a

1.41421356 con una precisión altísima.

**Ventajas:**

Convergencia **muy rápida** (cuadrática cerca de la raíz).

Implementación sencilla si tienes la derivada.

**Desventajas:**

Necesita  $f'(x)$ .

Si eliges mal  $x_0$ , puede divergir o irse a otra raíz.

Si  $f'(x_n) \approx 0$ , la fórmula explota numéricamente.

### Ejemplo del Método de Newton–Raphson aplicado a:

$$f(x) = e^x - \sin(3x) - 2$$

#### Fórmula general

Para una función  $f(x)$  derivable, la fórmula de Newton–Raphson es:

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

Necesitamos la derivada de  $f(x)$ :

$$f(x) = e^x - \sin(3x) - 2$$

La derivada de  $e^x$ .

La derivada de  $-\sin(3x)$  es  $-3\cos(3x)$ .

La derivada de  $-2$  es  $0$ .

Entonces:

$$f'(x) = e^x - 3\cos(3x)$$

#### Elección de $x_0$

Tomamos como valor inicial:

$$x_0 = 1$$

porque ya vimos que  $f(1) > 0$  y está cerca de la raíz hallada por bisección.

#### Desarrollo iterativo de Newton

Vamos a mostrar las primeras iteraciones (redondeando a 6 decimales).

En cada paso calculamos:

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

**Iteración 1**

$$x_0 = 1.000000$$

Calculamos:

$$f(1) = e^1 - \sin(3) - 2 \approx 0.577162$$

$$f'(1) = e^1 - 3 \cos(3) \approx 5.688259$$

Entonces:

$$x_1 = x_0 - \frac{f(x_0)}{f'(x_0)} = 1 - \frac{0.577162}{5.688259} \approx 1 - 0.101465 \approx 0.898535$$

**Iteración 2**

$$x_1 \approx 0.898535$$

---

Calculamos:

$$f(x_1) \approx 0.024651$$

$$f'(x_1) \approx 5.162555$$

Entonces:

$$x_2 = x_1 - \frac{f(x_1)}{f'(x_1)} \approx 0.898535 - \frac{0.024651}{5.162555} \approx 0.898535 - 0.004775 \approx 0.893760$$

**Iteración 3**

$$x_2 \approx 0.893760$$

---

Calculamos:

$$f(x_2) \approx 0.00007265$$

$$f'(x_2) \approx 5.132041$$

Entonces:

$$x_3 = x_2 - \frac{f(x_2)}{f'(x_2)} \approx 0.893760 - \frac{0.00007265}{5.132041} \approx 0.893760 - 0.00001456 \approx 0.893745$$

**Iteración 4 (ya prácticamente no cambia)**

$$x_3 \approx 0.893745$$

$$f(x_3) \approx 0.00000000065 \quad (\approx 6.5 \times 10^{-10})$$

El valor de  $x$  apenas cambia en más decimales, así que podemos considerar que la raíz aproximada es:

$$x \approx 0.893745 \quad (\text{Newton-Raphson})$$

**Resumen de Newton (tabla corta)**

It	$x_n$	$f(x_n)$	$f'(x_n)$	$ x_n - x_{n-1} $ aprox
0	1.000000	0.577162	5.688259	—
1	0.898535	0.024651	5.162555	0.101465
2	0.893760	0.00007265	5.132041	0.004775
3	0.893745	0.00000000065	5.131950	0.000015

Se ve la **convergencia muy rápida** comparado con bisección: en solo 3–4 iteraciones Newton llega a un error prácticamente nulo (en esta escala de decimales).

## Método de Brent (híbrido, robusto y muy usado)

Newton es ideal para enseñar, pero en la práctica profesional casi siempre se usa un método **híbrido** que combine:

la seguridad de **bisección** (nunca se sale del intervalo y siempre converge si hay cambio de signo),  
la rapidez de **secante**,  
y mejoras adicionales que eviten quedarte “estancado”.

Eso es lo que hace **Brent**.

### Idea general de Brent

Trabaja siempre con un **intervalo [a,b]** donde se cumple:

$$f(a) \cdot f(b) < 0$$

es decir, hay cambio de signo, por lo que existe al menos una raíz dentro.

En cada paso:

Intenta usar una aproximación rápida (tipo **secante** o **interpolación cuadrática inversa**) para proponer un nuevo punto candidato  $x$ .

Verifica condiciones de seguridad:

- ¿El candidato está dentro del intervalo?
- ¿Está reduciendo suficientemente el tamaño del intervalo?
- ¿No es un paso demasiado pequeño o inestable?

Si el candidato no es confiable, se **regresa a un paso de bisección**:

$$m = \frac{a + b}{2}$$

Actualiza el intervalo manteniendo siempre un cambio de signo.

Resultado:

Cuando la función se porta “bonita”, el método corre casi tan rápido como Newton/secante.  
Cuando la función se complica, **cae de regreso a bisección**, que es lenta pero segura.

Por eso es tan usado: **es difícil que falle**.

## Pasos conceptuales de Brent (simplificados)

Iniciar con **a** y **b** tales que  $f(a) \cdot f(b) < 0$ .

Elegir un punto **c** dentro de  $[a,b]$  (por ejemplo, el mejor aproximado anterior).

Según la historia de los últimos puntos, calcula un candidato usando:  
método de la **secante**, o **interpolación cuadrática inversa** (tres puntos).

Verificar:

que el punto esté dentro de  $[a,b]$ ,  
que reduzca de forma razonable el intervalo,  
que no genere pasos demasiado pequeños que bloqueen la convergencia.

Si las condiciones fallan → se usa **bisección**.

Actualizar el intervalo con el nuevo punto, manteniendo siempre un cambio de signo.

Repetir hasta que el intervalo sea muy pequeño o  $f(x)$  sea menor que la tolerancia.

## Ejemplo con la misma función $f(x) = x^2 - 2$

No voy a detallar todas las fórmulas internas (porque Brent es largo de escribir a mano), sino la **idea de cómo se movería el intervalo**. Tomemos:

$$a_0 = 1 \rightarrow f(1) = -1$$

$$b_0 = 2 \rightarrow f(2) = 2$$

Claramente  $f(a_0) \cdot f(b_0) < 0$ , hay raíz entre 1 y 2.

**Paso 1: tipo secante / regula falsi**

Punto de secante (similar a falsa posición):

$$x_1 = b_0 - f(b_0) \frac{b_0 - a_0}{f(b_0) - f(a_0)} = 2 - 2 \cdot \frac{2 - 1}{2 - (-1)} = 2 - \frac{2}{3} = 1.\bar{3}$$

$$x_1 \approx 1.3333$$

$$f(1.3333) \approx 1.7777 - 2 = -0.2222$$

El cambio de signo ahora está entre  $x_1$  y 2:

$$f(1.3333) < 0, f(2) > 0$$

Nuevo intervalo:  $[1.3333, 2]$ .

**Paso 2: nuevo intento rápido**

Con los nuevos puntos, Brent vuelve a intentar un paso tipo secante/interpolación.

Si el candidato nuevo está bien situado y mejora el intervalo, se acepta.

Si no, se toma el **punto medio**:

$$m = \frac{1.3333 + 2}{2} = 1.66665$$

y se continúa como bisección.

Con estos pasos sucesivos el intervalo se va cerrando:

[1.3333, 2]

luego algo como [1.3333, 1.6666]

luego [1.3333, 1.5]

etc.

Y la raíz se aproxima rápidamente a 1.41421356.

En la práctica:

Brent suele tardar **muy pocas iteraciones** (parecido a Newton),

pero con la **garantía** de que, mientras siempre haya cambio de signo, no se va a “volver loco” ni a salir del intervalo.

## Ventajas y desventajas de Brent (resumen)

### Ventajas:

**Muy robusto:** mantiene siempre un intervalo con cambio de signo.

**Rápido:** cuando la función se deja, se comporta casi tan bien como secante/Newton.

Es el método que implementan muchas librerías científicas modernas para  $f(x)=0$ .

### Desventajas:

Difícil de explicar y programar “a mano” en comparación con Newton o secante.

No es el primer método ideal para enseñar teoría; es más bien “herramienta profesional”.

## Comparación y conclusión para el ejemplo

Para la ecuación:

$$e^x - \sin(3x) - 2 = 0$$

con los métodos usados:

**Bisección** (intervalo inicial [0.5,1.0],  $\epsilon \approx 10^{-4}$ :

$X \approx 0.8937$

**Newton–Raphson** (inicial  $x_0 = 1$ ):

$X \approx 0.893745$

Ambos métodos coinciden en la misma raíz, solo que:

Bisección es **más lento pero muy seguro** (solo necesita que haya cambio de signo).

Newton–Raphson es **mucho más rápido**, pero requiere:

conocer  $f'(x)$ ,

un buen valor inicial  $x_0$ ,

y que la derivada no sea cero ni muy pequeña cerca de la raíz.