1. Introducción

El análisis numérico provee métodos computacionales para el estudio y la solución de problemas matemáticos. Debido a que los cálculos se realizan en computadoras digitales, debemos conocer las implicancias que esto tiene en la implementación de métodos numéricos. El estudio del error es de primordial importancia en el análisis numérico. La mayoría de los métodos numéricos obtienen soluciones que son sólo una aproximación de la solución verdadera, y es importante, de ser posible, poder estimar o acotar el error resultante. En esta Segunda Unidad, nos enfocamos en el estudio de los errores que se generan en los cálculos, debido a la representación computacional de números en punto flotante.

2. Sistemas de Numeración Posicionales

Los *sistemas de numeración* son *posicionales* cuando el valor de cada dígito del número depende de la posición en la que se encuentra. Ejemplos de sistemas posicionales: binario, decimal, octal y hexadecimal. Un ejemplo de sistema de numeración *no posicional* es el sistema romano. El número de símbolos permitidos en un sistema de numeración posicional se conoce como *base* del sistema de numeración. Si un sistema de numeración posicional tiene base β significa que disponemos de β símbolos diferentes para escribir los números. La tabla 1 presenta un listado de los distintos sistemas de numeración posicional.

Tabla 1: Sistemas de numeración posicionales

Sistema	Base	Cifras que utiliza
Binario	2	0, 1
Ternario	3	0, 1, 2
Cuaternario	4	0, 1, 2, 3
Quinario	5	0, 1, 2, 3, 4
Senario	6	0, 1, 2, 3, 4, 5
Septario o Hectal	7	0, 1, 2, 3, 4, 5, 6
Octal	8	0, 1, 2, 3, 4, 5, 6, 7
Nonario	9	0, 1, 2, 3, 4, 5, 6, 7, 8
Decimal	10	0, 1, 2, 3, 4, 5, 6, 7, 8, 9
Undecimal	11	0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A
...	...	...
Hexadecimal	16	0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F

En general, un número con parte entera finita se representa en la base β como

$$(-1)^{\sigma}(a_n a_{n-1} \dots a_1 a_0, a_{-1} a_{-2} \dots)_{\beta}$$

donde los coeficientes a_i son los valores (posición) de los dígitos en el sistema con base β, es decir, enteros positivos tales que $0 \leq a_i \leq \beta - 1$, y σ es una variable binaria que representa el signo del número ($\sigma = 0$ si el número es positivo y $\sigma = 1$ si es negativo).

La fórmula general para construir un número real x en un sistema de numeración posicional de base β es la siguiente:

$$x = (-1)^{\sigma} \left(a_n \beta^n + a_{n-1} \beta^{n-1} + \dots + a_1 \beta^1 + a_0 \beta^0 + a_{-1} \beta^{-1} + a_{-2} \beta^{-2} + \dots \right) \tag{1}$$

Por ejemplo, usando el sistema decimal, el número 213,58 se puede construir como:

$$213,58 = (-1)^0 \times \left(2 \times 10^2 + 1 \times 10^1 + 3 \times 10^0 + 5 \times 10^{-1} + 8 \times 10^{-2} \right).$$

2.1. Sistema Binario

En el *sistema binario*, de base $\beta = 2$, los números se representan utilizando solamente dos cifras: cero (0) y uno (1). Un dígito binario, o *bit*, puede representar uno de estos dos valores. El sistema binario es ampliamente utilizado en las computadoras ya que los procesadores se fabrican con transistores en su interior que no son sino pequeños interruptores que dejan pasar o no dejan pasar la electricidad, representando con ello los unos y los ceros respectivamente.

Un número binario se representa utilizando el subíndice $\beta = 2$, como por ejemplo el siguiente número con parte entera y fraccionaria:

$$(1\ 0\ 1\ 0\ 1,\ 1\ 1\ 0\ 1)_2$$

2.2. Conversión entre Decimal y Binario

Conversión de binario a decimal

Para la conversión de binario a decimal empleamos la fórmula (1) con $\beta = 2$. Por ejemplo, $(10101{,}1101)_2$ es la representación binaria del número 21,8125, puesto que,

$$(10101{,}1101)_2 = 2^4 + 2^2 + 2^0 + 2^{-1} + 2^{-2} + 2^{-4} = 21,8125$$

Otros ejemplos:

- Entero binario con m unos:

$$x = (\underbrace{1\ 1\ 1\ \dots\ 1}_{m\ \text{unos}})_2 = 2^{m-1} + 2^{m-2} + \cdots + 2^1 + 1 = \sum_{k=0}^{m-1} 2^k,$$

  lo cual representa una suma parcial de una serie geométrica de razón 2. Luego,

$$x = \frac{1 - 2^m}{1 - 2} = 2^m - 1$$

- El binario periódico $(0{,}01010101\dots)_2$.

$$\begin{aligned} x = (0{,}01010101\dots)_2 &= 2^{-2} + 2^{-4} + 2^{-6} + \dots \\ &= 2^{-2}\left(1 + 2^{-2} + 2^{-4} + \dots\right) = \frac{1}{4}\sum_{n=0}^{\infty}\left(\frac{1}{4}\right)^n, \end{aligned}$$

  lo cual representa una serie geométrica de razón $0,25 < 1$. Sabemos que dicha serie es convergente, y conocemos la expresión de su suma,

$$x = \frac{1}{4}\sum_{n=0}^{\infty}\left(\frac{1}{4}\right)^n = \frac{1}{4}\frac{1}{1 - \frac{1}{4}} = \frac{1}{4 - 1} = \frac{1}{3} = 0,333\dots$$

- El binario periódico $(0{,}110011001100\dots)_2$.

$$\begin{aligned} x = (0{,}110011001100\dots)_2 &= 2^{-1} + 2^{-2} + 2^{-5} + 2^{-6} + \dots \\ &= \sum_{n=1}^{\infty}\left(\frac{1}{2}\right)^{4n-2} + \sum_{n=1}^{\infty}\left(\frac{1}{2}\right)^{4n-3} = 4\sum_{n=1}^{\infty}\left(\frac{1}{16}\right)^n + 8\sum_{n=1}^{\infty}\left(\frac{1}{16}\right)^n \\ &= 12\frac{\frac{1}{16}}{1 - \frac{1}{16}} = \frac{12}{15} = 0,8 \end{aligned}$$

Conversión de decimal a binario

Un número entero x se convierte a binario dividiendo sucesivamente por dos hasta que el cociente sea 0, y registrando el valor de los restos, de acuerdo al siguiente procedimiento:

dividir x por 2, llamar al cociente x_1, el resto es a_0
dividir x_1 por 2, llamar al cociente x_2, el resto es a_1
dividir x_2 por 2, llamar al cociente x_3, el resto es a_2
$\vdots$

Por ejemplo, para convertir el número $(11)_{10}$ a binario, tenemos,

$$\begin{aligned} 11 = 2 \times 5 + 1 &\longrightarrow a_0 = 1 \\ 5 = 2 \times 2 + 1 &\longrightarrow a_1 = 1 \\ 2 = 2 \times 1 + 0 &\longrightarrow a_2 = 0 \\ 1 = 2 \times 0 + 1 &\longrightarrow a_3 = 1 \end{aligned}$$

Luego, $(11)_{10} = (1011)_2$.

Un número fraccionario x se convierte a binario multiplicando sucesivamente por dos, y registrando la parte entera del número resultante, de acuerdo al siguiente procedimiento:

multiplicar x por 2. La parte entera es a_{-1} y la parte fraccionaria es x_1.
multiplicar x_1 por 2. La parte entera es a_{-2} y la parte fraccionaria es x_2.
multiplicar x_2 por 2. La parte entera es a_{-3} y la parte fraccionaria es x_3.
$\vdots$

Por ejemplo, para convertir el número $x = (0,2)_{10}$ a binario, tenemos,

$$\begin{aligned} 2 \times x = 0,4 &\longrightarrow a_{-1} = 0,\ x_1 = 0,4 \\ 2 \times x_1 = 0,8 &\longrightarrow a_{-2} = 0,\ x_2 = 0,8 \\ 2 \times x_2 = 1,6 &\longrightarrow a_{-3} = 1,\ x_3 = 0,6 \\ 2 \times x_3 = 1,2 &\longrightarrow a_{-2} = 1,\ x_4 = 0,2 \\ &\vdots \end{aligned}$$

Luego, $(0,2)_{10} = (0,00110011001100\ldots)_2$. Notar que se obtiene una fracción binaria periódica.

3. Representación Computacional de Números en Punto Flotante

Las computadoras son el principal medio de cálculo en análisis numérico y por ello es importante conocer como operan. La aritmética que realiza una computadora es distinta de la aritmética de nuestros cursos de álgebra o cálculo. La computadora opera con números binarios y cada número se almacena con un número finito de dígitos binarios. Debido a esto, los números irracionales, los binarios periódicos, y muchos otros números, no se pueden representar con exactitud. La representación de números en punto flotante permite representar un número muy elevado (pero finito) de números reales sobre un amplio rango de valores, a pesar de emplear un número finito de dígitos. La notación en punto flotante está relacionada con la notación científica.

3.1. Representación General

Sea β la base del sistema de numeración empleado en la computadora. La mayoría de las computadoras emplean $\beta = 2$, o también $\beta = 8$ o 16. Un número x se representa en la computadora como un número en punto flotante, $fl(x)$, de la forma:

$$fl(x) = (-1)^{\sigma} \, (,a_1 \; a_2 \; \ldots \; a_n)_{\beta} \times \beta^{E-s} \tag{2}$$

La mantisa m es la parte fraccional del número, definida por los n dígitos a_i, $i = 1, \ldots, n$, como:

$$m = (,a_1 \; a_2 \; \ldots \; a_n)_{\beta} = \frac{a_1}{\beta^1} + \frac{a_2}{\beta^2} + \cdots + \frac{a_n}{\beta^n}$$

donde a_i, $i = 1, \ldots, n$, son números naturales tales que $0 \leq a_i \leq \beta - 1$, con $a_1 \neq 0$ para números en punto flotante normalizados.

El exponente E es un número entero positivo, definido por los t dígitos c_j, $j = 1, \ldots, t$, como:

$$E = (c_1 \; c_2 \; \ldots \; c_t)_{\beta} = c_1\beta^1 + c_2\beta^2 + \cdots + c_t\beta^t$$

El signo del número está definido por la variable binaria σ, introducida previamente, de forma que $\sigma = 0$ si el número es positivo y $\sigma = 1$ si es negativo.

El uso exclusivo de enteros positivos para el exponente no permitiría una representación adecuada de números con magnitud pequeña. Para garantizar que estos números también sean representables, se resta el sesgo s del exponente, el cual es una constante para una representación dada.

Ejemplos de números decimales en notación de punto flotante con una mantisa de 10 dígitos:

- $fl(3{,}333\ldots) = (,3333333333)_{10} \times 10^1$
- $fl(0{,}000777\ldots) = (,7777777778)_{10} \times 10^{-3}$
- $fl(100{,}02) = (,1000200000)_{10} \times 10^3$

El cero no puede ser representado como punto flotante normalizado y se representa como caso particular. Para una representación dada, existen límites en los exponentes que se pueden representar:

$$L \leq E - s \leq U$$

con $L < 0$ y $U > 0$. Si el exponente de un número x viola la cota inferior, es decir, si $E - s < L$, ocurre un desbordamiento a cero o *underflow*. En este caso, $fl(x) = 0$ y los cálculos continuan. Si el exponente de x viola la cota superior, es decir, si $E - s > U$, luego x no se puede representar como $fl(x)$ y ocurre un desbordamiento u *overflow*. Esto representa un error fatal y el cálculo (programa) se interrumpe.

3.2. Norma IEEE para Números en Punto Flotante

La norma IEEE 754 define distintos formatos estándar para números binarios en punto flotante.[1] Consideraremos solamente la *precisión simple* y la *precisión doble*.

Precisión simple: En precisión simple, los números se representan con 32 bits y un sesgo de 127. Se emplea 1 bit para el signo, 8 para el exponente, y 23 para la mantisa.

Bits	1	8	23	Total: 32 bits = 4 bytes
	σ	exponente	mantisa	

[1] IEEE son las siglas en inglés del Institute of Electrical and Electronics Engineers.

El flotante de un número x en precisión simple está dado por:

$$fl(x) = (-1)^{\sigma} \, (1, a_1 \; a_2 \; \ldots \; a_{23})_2 \times 2^{E-127} \tag{3}$$

Precisión doble: En precisión doble, los números se representan con 64 bits y un sesgo de 1023. Se emplea 1 bit para el signo, 11 para el exponente, y 52 para la mantisa.

Bits $\underbrace{1}_{\sigma}$ $\underbrace{11}_{\text{exponente}}$ $\underbrace{52}_{\text{mantisa}}$ Total: 64 bits = 8 bytes

El flotante de un número x en precisión doble está dado por:

$$fl(x) = (-1)^{\sigma} \, (1, a_1 \; a_2 \; \ldots \; a_{52})_2 \times 2^{E-1023} \tag{4}$$

Notar que en el estándar IEEE 754, el 1 anterior a la coma en (3) y (4) es implícito, y no se almacena ya que se asume se presencia. Por otra parte, el dígito a_1 puede ser igual a cero.

Llamaremos *significante* al número $\xi = (1, a_1 \; a_2 \; \ldots \; a_n)_2$, para distinguirlo de la mantisa. Se desprende inmediatamente que el significante satisface $1 \leq \xi < 2$. Analizaremos cuales son los límites del exponente $E - s$ en precisión simple.

El máximo valor del exponente E con precisión simple está dado por,

$$(11111111)_2 = 2^8 - 1 = 255$$

Para números normalizados se representan números enteros del 1 al 254. El mínimo (0) y el máximo (255) se reservan para otros fines. Utilizando un sesgo de 127, se pueden representar exponentes en el rango,

$$-126 \leq E - s \leq 127$$

Es decir, los límites del exponente son:

$$L = -126 = 1 - 127$$
$$U = 127 = 254 - 127$$

Ejemplo. Representación del número $(40)_{10}$ en precisión simple.

$$\underbrace{0}\;\underbrace{10000100}\;\underbrace{01000000000000000000000}$$

Exponente: $E = 2^7 + 2^2 = 132$
Significante: $\xi = 2^0 + 2^{-2} = 1{,}25$

Verificación: $1{,}25 \times 2^{132-127} = 1{,}25 \times 2^5 = 40$

3.3. Truncamiento y Redondeo

La mayoría de los números reales no se pueden representar en forma exacta en la representación en punto flotante introducida previamente. Por lo tanto deben aproximarse por un número cercano que sea representable. Dado un número real arbitrario x, existen dos maneras principales de generar $fl(x)$ a partir de x: el truncamiento y el redondeo.

Cualquier número real x se puede escribir como:

$$x = (-1)^{\sigma} \, (, a_1 \; a_2 \; \ldots \; a_n \; a_{n+1} \; \ldots)_{\beta} \times \beta^{E-s}$$

con $a_1 \neq 0$.

Truncamiento: Consiste en cortar los números a_{n+1}, a_{n+2}, ...

$$fl(x) = (-1)^{\sigma} \, (, a_1 \; a_2 \; \ldots \; a_n)_{\beta} \times \beta^{E-s} \tag{5}$$

Redondeo: En el caso de un número redondeado, tenemos:

$$fl(x) = \begin{cases} (-1)^{\sigma} \, (, a_1 \; a_2 \; \ldots \; a_{n-1} \; a_n)_{\beta} \times \beta^{E-s} & 0 \leq a_{n+1} < \dfrac{\beta}{2} \\ \\ (-1)^{\sigma} \, [(, a_1 \; a_2 \; \ldots \; a_{n-1} \; a_n)_{\beta} + (, 0 \; 0 \; \ldots \; 0 \; 1)_{\beta}] \times \beta^{E-s} & \dfrac{\beta}{2} \leq a_{n+1} < \beta \end{cases} \tag{6}$$

En ocasiones, se emplea una variante de la definición dada por (6) a fin de obtener un redondeo no sesgado. En el caso particular en que:

$$(1) \; a_{n+1} = \frac{\beta}{2} \quad \text{y} \quad (2) \; a_j = 0 \text{ para } j \geq n+2,$$

se redondea hacia arriba si a_n es impar y hacia abajo si a_n es par.

Redondeo en decimal: Notar que la definición (6) concuerda con la definición clásica de redondeo que conocemos para el sistema decimal:

$$fl(x) = \begin{cases} (-1)^{\sigma} \, (, a_1 \; a_2 \; \ldots \; a_{n-1} \; a_n)_{10} \times 10^{E-s} & 0 \leq a_{n+1} < 5 \\ \\ (-1)^{\sigma} \, [(, a_1 \; a_2 \; \ldots \; a_{n-1} \; a_n)_{10} + (, 0 \; 0 \; \ldots \; 0 \; 1)_{10}] \times 10^{E-s} & 5 \leq a_{n+1} < 10 \end{cases}$$

Redondeo en binario: Incluyendo el significante empleado en la norma IEEE, el redondeo en binario está dado por:

$$fl(x) = \begin{cases} (-1)^{\sigma} \, (1, a_1 \; a_2 \; \ldots \; a_{n-1} \; a_n)_2 \times 2^{E-s} & a_{n+1} = 0 \\ \\ (-1)^{\sigma} \, [(1, a_1 \; a_2 \; \ldots \; a_{n-1} \; a_n)_2 + (, 0 \; 0 \; \ldots \; 0 \; 1)_2] \times 2^{E-s} & a_{n+1} = 1 \end{cases}$$

Algunos programas como Matlab y Scilab utilizan la variante mencionada anteriormente, es decir, en el caso particular en que:

$$(1) \; a_{n+1} = 1 \quad \text{y} \quad (2) \; a_j = 0 \text{ para } j \geq n+2,$$

se redondea hacia arriba si $a_n = 1$ y hacia abajo si $a_n = 0$.

3.4. Medidas de Precisión de la Representación en Punto Flotante

Introduciremos ahora algunas medidas que nos darán una idea de la precisón posible de la representación con punto flotante.

Epsilon de la máquina

Sea y el menor número representable "en la máquina"que es mayor a 1. El epsilon de la máquina es una medida de precisión dada por: $\varepsilon = y - 1$.

Empleando la norma IEEE se tiene:

$$1 = (1, 0 \; 0 \; \ldots \; 0 \; 0)_2 \times 2^0$$
$$y = (1, 0 \; 0 \; \ldots \; 0 \; 1)_2 \times 2^0 = 1 + 2^{-n} > 1$$

Luego, $\varepsilon = 2^{-n}$. En precisión simple, $\varepsilon = 2^{-23} \approx 1{,}19 \times 10^{-7}$.

Unidad de redondeo

La unidad de redondeo de un computador es un número δ que satisface:

1) es un número positivo en punto flotante.

2) es el menor número tal que $fl(1+\delta) > 1$.

Luego, para cualquier otro número positivo en punto flotante $\hat{\delta} < \delta$, se tiene que $fl(1+\hat{\delta}) = 1$, y así, $1+\hat{\delta}$ es identico a 1 en la aritmética del computador. Notar que δ mide el "ancho del cero" en la representación de punto flotante.

No es difícil derivar el valor de δ. Empleando la norma IEEE tenemos,

$$(1,0\ 0\ \ldots\ 0\ 1\ 0\ 0\ \ldots)_2 \times 2^0 = 1 + 2^{-n-1}$$
$$\uparrow$$
$$\text{posición } n+1$$

Utilizando redondeo en binario (sin la variante que utiliza Scilab):

$$fl(1+2^{-n-1}) = (1,0\ 0\ \ldots\ 0\ 1)_2 \times 2^0 = 1 + 2^{-n} > 1$$
$$\uparrow$$
$$\text{posición } n$$

Ahora, si $\hat{\delta} < 2^{-n-1}$ entonces $1+\hat{\delta}$ tiene un cero en la posición $n+1$ de la mantisa, y por definición se tiene entonces que $fl(1+\hat{\delta}) = 1$. Luego, $\delta = 2^{-n-1}$. En precisión simple, $\delta \approx 5{,}96 \times 10^{-8}$.

Mayor entero positivo representable en forma exacta.

Otra medida de precisión relacionada con el número de bits del significante consiste en hallar el mayor entero M tal que todo entero x que satisface $0 \leq x \leq M$, se puede representar en forma exacta en punto flotante. Es decir, se trata de hallar $M \in \mathbb{Z}^+$ tal que:

1) $0 < x \leq M$, $x \in \mathbb{Z}^+$, implica $fl(x) = x$

2) $fl(M+1) \neq M+1$

En precisión simple, tenemos:

$$(1,\underbrace{1\ 1\ 1\ \ldots\ 1}_{23\text{ unos}})_2 \times 2^{23} = 2^{23} + 2^{22} + \cdots + 2^1 + 2^0 = 2^{24} - 1$$

Además, 2^{24} se almacena en forma exacta,

$$2^{24} = (1,\underbrace{0\ 0\ 0\ \ldots\ 0}_{23\text{ ceros}})_2 \times 2^{24}$$

Sin embargo, $2^{24}+1$ no se almacena en forma exacta ya que esto requeriría una mantisa de 24 bits:

$$(1,\underbrace{0\ 0\ 0\ \ldots\ 0}_{23\text{ ceros}}\ 1)_2 \times 2^{24}$$

Luego, $M = 2^{24} = 16777216$.

4. Errores Numéricos

4.1. Error Absoluto y Relativo

Al resover un problema, buscamos obtener la solución exacta o verdadera, que denotamos x_v. Sin embargo, aplicando métodos numéricos se obtiene por lo general una solución approximada x_a. Definimos el *error* en x_a como:

$$\text{Error} = x_v - x_a$$

Definimos el *error absoluto* y el *error relativo* en x_a como:

$$\text{Error absoluto} = |\text{Error}| = |x_v - x_a|$$

$$\text{Error relativo} = \frac{\text{error absoluto}}{|\text{valor verdadero}|} = \frac{|x_v - x_a|}{|x_v|}$$

4.2. Error de Truncamiento y Redondeo

Si $x \neq fl(x)$ y se utiliza truncamiento, luego $fl(x) < x$ y el error $x - fl(x)$ es siempre positivo. Esto trae consecuencias en el cálculo numérico, ya que no hay posibilidad de cancelación de errores y la propagación de errores es mayor. Con el redondeo, el error $x - fl(x)$ es negativo para la mitad de los valores de x y positivo para la otra mitad de los valores posibles de x. Además, el peor error posible por redondeo es la mitad que en el caso de truncamiento.

A menudo se representa el error relativo como

$$\frac{x - fl(x)}{x} = -\varepsilon, \quad \text{si } x \neq 0$$

de donde

$$fl(x) = (1 + \varepsilon)x$$

Luego, $fl(x)$ puede verse como un valor perturbado de x. La siguiente proposición provee cotas sobre el error relativo ε.

Proposición 1 *Sea $x \in \mathbb{R}$, con $x \neq 0$. Las siguientes cotas sobre el error relativo ε son válidas empleando las fórmulas de truncamiento y redondeo dadas por* (5) *y* (6)*, respectivamente.*

$$i) \qquad -\beta^{-n+1} \leq \varepsilon \leq 0 \qquad\qquad fl(x) \textit{ truncado}$$

$$ii) \quad -\frac{1}{2}\beta^{-n+1} \leq \varepsilon \leq \frac{1}{2}\beta^{-n+1} \quad fl(x) \textit{ redondeado}$$

Demostración. Veremos la demostración del item (i) solamente. Supondremos $\sigma = 0$ (el caso $\sigma = 1$ no cambia el signo de ε). En el caso de truncamiento, tenemos:

$$x - fl(x) = (, 0\ 0\ \ldots\ 0\ a_{n+1}\ a_{n+2}\ \ldots)_\beta \times \beta^e, \quad \text{con } e = E - s$$

Sea $\gamma = \beta - 1$,

$$0 \leq x - fl(x) \leq (, 0\ 0\ \ldots\ 0\ \gamma\ \gamma\ \ldots)_\beta \times \beta^e =$$

$$= \gamma \left[\sum_{i=1}^{\infty} \frac{1}{\beta^{n+i}} \right] \beta^e = \frac{\gamma}{\beta^n} \left[\sum_{i=1}^{\infty} \left(\frac{1}{\beta} \right)^i \right] \beta^e = \frac{\gamma}{\beta^n} \frac{\frac{1}{\beta}}{1 - \frac{1}{\beta}} \beta^e = \frac{\gamma}{\beta^n} \frac{1}{\gamma} \beta^e = \frac{\beta^e}{\beta^n} = \beta^{-n+e}$$

Dividiendo por x la desigualdad anterior, tenemos

$$0 \leq \frac{x - fl(x)}{x} \leq \frac{\beta^{-n+e}}{(, a_1\ a_2\ \ldots)_\beta \times \beta^e} \tag{7}$$

Luego,

$$0 \leq -\varepsilon \leq \frac{\beta^{-n}}{(, 1\ 0\ 0\ 0\ \ldots)_\beta} = \beta^{-n+1}$$

con lo cual queda demostrado el item (i). □

4.3. Cifras Significativas

En un trabajo científico, se considera que las cifras significativas (o dígitos significativos) de un número son aquellas que tienen un significado real o aportan alguna información. Las cifras significativas de un número vienen determinadas por su incertidumbre. Por ejemplo, consideremos una medida de longitud que arroja un valor de 4325,3528 metros con un error de 0,8 metros. Puesto que el error es del orden de décimas de metro, es evidente que todas las cifras del número que ocupan una posición menor que las décimas no aportan ninguna información. No tiene sentido dar el número con una exactitud de diez milésimas, si afirmamos que el error es de casi un metro. Cuando se expresa un número debe evitarse siempre la utilización de cifras no significativas.

Cifras significativas de un número

Para conocer el número de cifras significativas de un número decimal, se siguen las siguientes reglas:

- Cualquier dígito distinto de cero es significativo. Por ejemplo, 438 tiene tres cifras significativas.
- Los ceros situados en medio de números diferentes de cero son significativos. Por ejemplo, 402 tiene tres cifras significativas, y 30002 tiene cinco cifras significativas.
- Los ceros a la izquierda del primer número distinto de cero no son significativos. Por ejemplo, 0,0023 tiene dos cifras significativas.
- Los ceros que se encuentran después de la coma y después de un dígito distinto de cero, son significativos. Por ejemplo 10,00 tiene 4 cifras significativas, y 0,0030 tiene dos cifras significativas.
- En los números enteros, los ceros situados después de un dígito distinto de cero pueden ser o no significativos. Por ejemplo, 600 puede tener una cifra significativa (6), dos (60), o tres (600). Para conocer el número correcto de cifras significativas necesitamos conocer más información acerca de cómo fué generado el número (por ejemplo, si el número es una medición, necesitamos conocer la precisión del instrumento de medición empleado). También podemos conocer el número correcto de cifras significativas si expresamos el número en notación científica. Por ejemplo, 6×10^2 tiene una cifra significativa, $6{,}0 \times 10^2$ tiene dos cifras significativas, y $6{,}00 \times 10^2$ tiene tres cifras significativas.

Cifras significativas de un valor aproximado con respecto a un valor verdadero

Sea x_v el valor verdadero de un número y x_a un valor aproximado.

Definición. Decimos que x_a tiene m cifras significativas con respecto a x_v si el error $|x_v - x_a|$ tiene una magnitud menor o igual a cinco unidades en el dígito $(m+1)$ de x_v contando de izquierda a derecha desde el primer dígito distinto de cero en x_v.

Ejemplos

(a) $x_v = 1/3 \quad x_a = 0,333 \quad |x_v - x_a| \doteq 0{,}000333$

Decimos que x_a tiene tres cifras significativas con respecto a x_v.

(b) $x_v = 23,496 \quad x_a = 23,494 \quad |x_v - x_a| = 0{,}002$

Decimos que x_a tiene cuatro cifras significativas con respecto a x_v.

(c) $x_v = 0{,}02144 \quad x_a = 0{,}02138 \quad |x_v - x_a| = 0{,}00006$

Decimos que x_a tiene dos cifras significativas (y no tres) con respecto a x_v.

Para medir el número de cifras significativas de un valor aproximado se suele emplear la siguiente desigualdad. Si

$$\left|\frac{x_v - x_a}{x_v}\right| \leq 5 \times 10^{-m-1}, \tag{8}$$

luego x_a tiene m cifras significativas con respecto a x_v. Para demostrar esto, consideremos primero el caso en que $0{,}1 \leq x_v < 1$. Luego (8) implica

$$|x_v - x_a| \leq 5 \times 10^{-m-1}|x_v| < 5 \times 10^{-m-1}.$$

Como $0{,}1 \leq x_v < 1$, esto implica que x_v tiene m cifras significativas. Para un x_v general la demostración es la misma, haciendo $x_v = \hat{x}_v \times 10^E$, con $0{,}1 \leq \hat{x}_v < 1$, y E un número entero.

Nota: Notar que (8) es una condición suficiente, pero no necesaria, para que x_a tenga m cifras significativas con respecto a x_v. Los ejemplos (a) y (b) dados anteriormente tienen un mayor número de cifras significativas que las indicadas por la condición (8).

Redondeo a m cifras significativas

Redondear un número decimal x a m cifras significativas (o a m dígitos) es equivalente a redondear el número utilizando en notación de punto flotante una mantisa de m dígitos. Para ello, primero se escribe el número en la forma $x = \hat{x} \times 10^E$, con $0{,}1 \leq \hat{x} < 1$, y E un número entero. Luego se procede a redondear $\hat{x}$ con m dígitos después de la coma. El número redondeado es $\text{rn}(x) = \bar{x} \times 10^E$, con $\bar{x} = 0, a_1\ a_2\ \cdots\ a_m$. Puesto que $a_1 \neq 0$ y todos los dígitos se encuentran después de la coma, $\text{rn}(x)$ tiene m cifras significativas. Además, el valor aproximado que se obtiene $x_a = \text{rn}(x)$ tiene m cifras significativas con respecto al valor original $x_v = x$, puesto que al redondear un número se cumple la definición vista anteriormente.

Ejemplos

(a) Redondeo con 5 cifras significativas

$x_v = 1,123456 \quad x_a = 1,1235 \quad |x_v - x_a| = 0,000044$

Luego x_a tiene cinco cifras significativas con respecto a x_v.

(b) Redondeo con 2 cifras significativas

$x_v = 0,20004 \quad x_a = 0,20 \quad |x_v - x_a| = 0{,}00004$

Luego x_a tiene dos cifras significativas (y no cuatro) con respecto a x_v.

(c) Redondeo con 4 cifras significativas

$x_v = 0,20005 \quad x_a = 0,2001 \quad |x_v - x_a| = 0{,}00005$

Luego x_a tiene cuatro cifras significativas con respecto a x_v.

4.4. Propagación de Errores

Consideraremos el efecto de realizar cálculos con números sujetos a error.

Error propagado

Sea ω una de las operaciones aritméticas $+$, $-$, $\times$, $/$; y sea $\hat{\omega}$ la versión computacional de la misma operación, la cual incluye redondeo o truncamiento. Sean x_a e y_a números usados en los cálculos, y suponga que ya presentan error, siendo sus valores verdaderos

$$x_v = x_a + \epsilon, \qquad y_v = y_a + \eta.$$

Luego, $x_a \hat{\omega} y_a$ es el número calculado, y su error está dado por:

$$x_v \omega y_v - x_a \hat{\omega} y_a = [x_v \omega y_v - x_a \omega y_a] + [x_a \omega y_a - x_a \hat{\omega} y_a] \tag{9}$$

La primera cantidad entre corchetes es llamada *error propagado*, mientras que la segunda cantidad es el error de redondeo o de truncamiento. Supondremos en los sucesivo que se emplea redondeo. Para esta segunda cantidad, usualmente tenemos que

$$x_a \hat{\omega} y_a = fl(x_a \omega y_a) \tag{10}$$

lo cual significa que $x_a \omega y_a$ se calcula con exactitud y luego se redondea. Aplicando la cota (ii) de la Proposición 1,

$$|x_a \omega y_a - x_a \hat{\omega} y_a| \leq \frac{\beta^{-n+1}}{2} |x_a \omega y_a|$$

Para el error propagado examinaremos los casos particulares.

Caso (a). Multiplicación. Para el error en $x_a y_a$ tenemos,

$$\begin{aligned} x_v y_v - x_a y_a &= x_v y_v - (x_v - \epsilon)(y_v - \eta) \\ &= x_v \eta + y_v \epsilon - \epsilon \eta \end{aligned}$$

Definiendo el error relativo, $\mathrm{Rel}(x_a) \equiv \epsilon / x_v$, tenemos

$$\begin{aligned} \mathrm{Rel}(x_a y_a) = \frac{x_v y_v - x_a y_a}{x_v y_v} &= \frac{\eta}{y_v} + \frac{\epsilon}{x_v} - \frac{\epsilon}{x_v}\frac{\eta}{y_v} \\ &= \mathrm{Rel}(x_a) + \mathrm{Rel}(y_a) - \mathrm{Rel}(x_a)\mathrm{Rel}(y_a) \end{aligned}$$

Para $|\mathrm{Rel}(x_a)|, |\mathrm{Rel}(y_a)| \ll 1$,

$$\mathrm{Rel}(x_a y_a) \approx \mathrm{Rel}(x_a) + \mathrm{Rel}(y_a)$$

El símbolo "$\ll$" significa "mucho menor que."

Caso (b). División. Usando argumentos similares,

$$\mathrm{Rel}\left(\frac{x_a}{y_a}\right) = \frac{\mathrm{Rel}(x_a) - \mathrm{Rel}(y_a)}{1 - \mathrm{Rel}(y_a)}$$

Para $|\mathrm{Rel}(y_a)| \ll 1$,

$$\mathrm{Rel}\left(\frac{x_a}{y_a}\right) \approx \mathrm{Rel}(x_a) - \mathrm{Rel}(y_a)$$

Tanto para la multiplicación como para la división los errores relativos no se propagan rápidamente.

Caso (c). Suma y resta.

$$(x_v \pm y_v) - (x_a \pm y_a) = (x_v - x_a) \pm (y_v - y_a) = \epsilon \pm \eta,$$

por lo tanto,

$$\text{Error}(x_a \pm y_a) = \text{Error}(x_a) \pm \text{Error}(y_a)$$

Esto puede parecer bueno y razonable, pero es engañoso. El error relativo $\text{Rel}(x_a \pm y_a)$ puede ser bastante pobre comparado con $\text{Rel}(x_a)$ y $\text{Rel}(y_a)$.

Ejemplo. Sea $x_v = \pi$, $x_a = 3{,}1416$, $y_v = \frac{22}{7}$, $y_a = 3{,}1429$. Luego

$$\begin{aligned} x_v - x_a &\approx -7{,}35 \times 10^{-6} \qquad \text{Rel}(x_a) \approx -2{,}34 \times 10^{-6} \\ y_v - y_a &\approx -4{,}29 \times 10^{-5} \qquad \text{Rel}(y_a) \approx -1{,}36 \times 10^{-5} \\ (x_v - y_v) - (x_a - y_a) &\approx -0{,}0012645 - (-0{,}0013) \approx 3{,}55 \times 10^{-5} \\ \text{Rel}(x_a - y_a) &\approx -0{,}028 \end{aligned}$$

Aunque el error en $x_a - y_a$ es bastante pequeño, el error relativo $\text{Rel}(x_a - y_a)$ es mucho mayor que $\text{Rel}(x_a)$ y $\text{Rel}(y_a)$. Esta pérdida de precisión al sustraer cantidades similares se examinará con mayor detalle a continuación.

Error por supresión de cifras significativas

Cuando restamos dos números muy cercanos, ocurre por lo general un error de supresión de dígitos significativos. Son problemas difíciles de detectar, e incluso cuando se detectan pueden ser difíciles de resolver. Analizaremos como se producen estos errores mediante el siguiente ejemplo.

Evaluar la función

$$f(x) = x\left(\sqrt{x+1} - \sqrt{x}\right)$$

en una calculadora decimal de 6 dígitos, es decir, empleando una representación de números decimales en punto flotante con una mantisa de 6 dígitos. En la siguiente tabla se muestra el valor de $f(x)$ que se obtienen con la calculadora para valores crecientes de x, y el valor real de $f(x)$, redondeado correctamenta a 6 dígitos.

x	$f(x)$ calculadora	$f(x)$ real
1	,414210	,414214
10	1,54340	1,54347
100	4,99000	4,98756
1000	15,8000	15,8074
10000	50,0000	49,9988
100000	100,000	158,113

Vemos como para valores elevados de x el error en la evaluación de $f(x)$ aumenta considerablemente. Para ver lo que está sucediendo analizaremos el caso de $x = 100$. Tenemos:

$$\sqrt{101} = 10{,}0499 \qquad \sqrt{100} = 10{,}0000$$

Luego, en la aritmética de la calculadora tenemos

$$\sqrt{101} - \sqrt{100} = 0{,}0499000$$

Mientras que el valor real es $\sqrt{101} - \sqrt{100} = 0{,}0498756$. Comparando ambos números vemos que hubo una supresión de tres cifras significativas.

En este ejemplo en particular, es posible evitar la supresión de cifras significativas reformulando $f(x)$:

$$f(x) = x\frac{(\sqrt{x+1} - \sqrt{x})}{1}\frac{(\sqrt{x+1} + \sqrt{x})}{(\sqrt{x+1} + \sqrt{x})} = \frac{x}{\sqrt{x+1} + \sqrt{x}}$$

En este caso, si evaluamos $f(100)$ con una calculadora decimal de 6 dígitos, obtenemos $f(100) = 4{,}98756$, lo cual es igual al valor real de $f(100)$ correctamente redondeado.

Error propagado en la evaluación de funciones

Supongamos que queremos evaluar la función $f(x)$ en la computadora. El resultado por lo general no será el valor de $f(x)$ sino una aproximación de dicho valor que denotamos $\hat{f}(x)$. Por otra parte, por lo general queremos evaluar la función para un valor exacto x_v, pero en lugar de ello, la evaluamos para un valor aproximado x_a. El error resultante en la evaluación de la función estará dado por:

$$f(x_v) - \hat{f}(x_a) = [f(x_v) - f(x_a)] + [f(x_a) - \hat{f}(x_a)] \tag{11}$$

La primera cantidad entre corchetes es el error propagado, y es el error que resulta de aplicar aritmética exacta en la evaluación de la función. La segunda cantidad entre corchetes es el error que resulta de evaluar $f(x_a)$ en la computadora. Este segundo error puede verse como una variable aleatoria de pequeña magnitud que resulta de la acumulación de los errores de redondeo asociados a las operaciones aritméticas que definen a la función $f(x)$.

Error en sumatorias

Veremos cómo se propaga el error al realizar una sumatoria de números empleando aritmética de punto flotante. Sea la suma

$$S = \sum_{j=1}^{m} x_j$$

donde $x_1, \ldots, x_m$ son números en punto flotante. Definimos

$$S_2 \equiv fl(x_1 + x_2) = (x_1 + x_2)(1 + \varepsilon_2)$$

Recursivamente, definimos

$$S_{r+1} \equiv fl(S_r + x_{r+1}) = (S_r + x_{r+1})(1 + \varepsilon_{r+1}), \qquad r = 2, \ldots, m-1$$

Expandiendo las primeras tres sumas obtenemos:

$$\begin{aligned}
S_2 - (x_1 + x_2) &= (x_1 + x_2)\varepsilon_2 \\
S_3 - (x_1 + x_2 + x_3) &= (x_1 + x_2)\varepsilon_2 + (x_1 + x_2 + x_3)\varepsilon_3 + (x_1 + x_2)\varepsilon_2\varepsilon_3 \\
S_4 - (x_1 + x_2 + x_3 + x_4) &= (x_1 + x_2)\varepsilon_2 + (x_1 + x_2 + x_3)\varepsilon_3 + (x_1 + x_2 + x_3 + x_4)\varepsilon_4 + \\
&\quad + (x_1 + x_2)(\varepsilon_2\varepsilon_3 + \varepsilon_2\varepsilon_4 + \varepsilon_2\varepsilon_3\varepsilon_4) + (x_1 + x_2 + x_3)\varepsilon_3\varepsilon_4
\end{aligned}$$

Despreciando los productos de errores relativos $\varepsilon_i\varepsilon_j$, debido a su pequeña magnitud, obtenemos por inducción

$$\begin{aligned} S_m - \sum_{j=1}^{m} &\approx (x_1+x_2)\varepsilon_2 + (x_1+x_2+x_3)\varepsilon_3 + \cdots + (x_1+x_2+x_3+\cdots+x_m)\varepsilon_m \\ &= (x_1+x_2)(\varepsilon_2+\varepsilon_3+\cdots+\varepsilon_m) + x_3(\varepsilon_3+\varepsilon_4+\cdots+\varepsilon_m) + \\ &+ x_4(\varepsilon_4+\cdots+\varepsilon_m) + \cdots + x_m\varepsilon_m \end{aligned}$$

Observando esta fórmula, vemos que el mayor número de errores ε_j multiplica a x_1 y x_2, mientras que solo ε_m multiplica a x_m. Si queremos minimizar el error $|S - S_m|$ deducimos que la mejor estrategia es sumar los números del menor al mayor, es decir, ordenando los términos antes de sumarlos de tal modo que $0 \leq |x_1| \leq |x_2| \leq \cdots \leq |x_m|$. Por supuesto, existen contraejemplos, pero para sumatorias grandes esta estrategia por lo general minimiza la propagación de errores.

4.5. Fuentes de Error en Problemas Matemáticos

La resolución de un problema matemático de ingeniería o de ciencia computacional está sujeto a las siguientes fuentes de error.

1) Error de modelado matemático. En física y en ciencias aplicadas (lo cual incluye todas las ingenierías), un modelo matemático es una representación simplificada, a través de ecuaciones, funciones o fórmulas matemáticas, de un fenómeno o de la relación entre dos o más variables. Se podría decir que un modelo matemático es una traducción de la realidad física de un sistema físico en términos matemáticos, es decir, una forma de representar matemáticamente cada uno de los tipos de entidades que intervienen en un cierto proceso.

Las relaciones matemáticas formales entre los objetos del modelo, deben representar de alguna manera las relaciones reales existentes entre las diferentes entidades o aspectos del sistema físico. Así, una vez "traducido" o "representado" cierto problema en forma de modelo matemático, se pueden aplicar el cálculo, el álgebra y otras herramientas matemáticas para deducir el comportamiento del sistema bajo estudio.

Como ejemplo, consideremos la Ley de Gases Ideales, dada por

$$PV = nRT$$

donde P es la presión del gas, V es el volumen que ocupa, n es el número de moles del gas, lo cual está relacionado con su masa, R es la constante universal de los gases ideales, y T es la temperatura absoluta del gas. Esta ley describe el estado de un gas hipotético formado por moléculas puntuales que no se atraen o repelen entre sí. No existen gases que sean exactamente ideales, pero muchos de ellos se aproximan al comportamiento ideal para temperaturas cercanas a la temperatura ambiente y presiones cercanas a la presión atmosférica, de tal modo que aproximarlos por un gas ideal es muy útil en numerosas situaciones. Por supuesto, existen ecuaciones de estado mas precisas. van der Waals introdujo correcciones que tenían en cuenta el volumen de las moléculas y las fuerzas atractivas que una molécula ejerce sobre otra a distancias muy cercanas entre ellas, lo cual le valió el premio Nobel en 1910. La ecuación de van der Waals está dada por

$$\left(P + \frac{an^2}{V^2}\right)(V - nb) = nRT$$

donde las constantes a y b son características de cada gas.

Debido a la complejidad de la realidad física, un modelo matemático es siempre una aproximación de la realidad. La complejidad del modelo puede variar dependiendo de las simplificaciones que se realicen para que el modelo sea más manejable. Como resultado, la precisión del modelo es limitada, y estas limitaciones pueden o no representar un problema dependiendo del uso que quiera hacerse del modelo.

Si un modelo es adecuado para un determinado propósito, luego desearíamos emplear un método numérico que preserve dicha precisión. Si por el contrario el modelo no es suficientemente preciso, luego el análisis numérico no puede mejorar dicha precisión (salvo por casualidad). Por otra parte, no es una buena idea crear un modelo que sea más complicado de lo necesario. Un modelo más complicado puede introducir mayores dificultades en el análisis numérico sin que ello redunde en un mayor beneficio en relación al propósito para el cual el modelo fue creado.

2) Incertidumbre en datos físicos. La mayoría de los datos obtenidos a partir de un experimento físico están sujetos a errores de medición o incertidumbre. Esto afecta la precisión de los cálculos realizados en base a dichos datos. El efecto de dichos errores en los cálculos es similar al efecto de los errores de redondeo, si bien, el error en los datos físicos es por lo general mucho mayor que el error de redondeo.

3) Equivocaciones. Existen diversos errores que podemos cometer al programar un método numérico para resolver un problema matemático. Por ejemplo, podemos escribir incorrectamente una ecuación o ingresar incorrectamente el valor de un parámetro del modelo, o en forma más general, podemos cometer errores de programación. En general, para programas largos y complejos los errores de programación pueden ser difíciles de detectar. Un pequeño error sutil puede producir una gran diferencia en los resultados numéricos. Por ello es importante emplear técnicas para la revisión sistemática del código fuente y contar con medios computacionales para la detección de errores (depuradores). Es importante verificar que el programa esté devolviendo la respuesta que esperamos de él, para lo cual se aconseja correr el programa utilizando ejemplos para los que se conoce la respuesta exacta. Para evitar errores es importante emplear buenas prácticas en programación.

4) Errores de truncamiento o redondeo. Los errores de truncamiento o redondeo son inevitables cuando se utiliza representación en punto flotante. En esta Unidad hemos estudiado los efectos que producen dichos errores.

5) Error de aproximación matemática. Es la principal fuente de error de la que se ocupa el análisis numérico, y analizaremos este error para muchos de los métodos numéricos que se estudian en este curso. El error de aproximación matemática es el error que ocurre cuando reemplazamos un problema computacionalmente difícil de resolver (o irresoluble) por un problema semejante que es más fácil de resolver. Los siguientes ejemplos permiten precisar la idea:

- Aproximación polinomial de Taylor.

$$e^x \approx 1 + x + \frac{1}{2}x^2$$

- Integración numérica.

$$\int_0^1 f(x)dx \approx \frac{1}{n}\sum_{j=1}^{n} f\left(\frac{j}{n}\right)$$

- Diferenciación numérica.

$$f'(x) \approx \frac{f(x+h) - f(x-h)}{2h}$$