Análisis y Modelamiento Numérico I

Representación de números en el computador

Los profesores[1]

[1]Facultad de Ciencias
Universidad Nacional de Ingeniería

2023-1

Contenidos

Épsilon de la máquina

Los números en punto flotante no están uniformemente distribuidos sobre la recta real, sino que están más próximos cerca del origen y más separados a medida que nos alejamos de él. Con mayor precisión, en un intervalo fijo $[\beta^e, \beta^{e+1}]$ los números de punto flotante presentes están igualmente espaciados con una separación igual a β^{e-t}.
Conforme e se incrementa, el espaciado entre los mismos crece también. Una medida de este espaciamento es dado por el llamado **epsilon de la máquina**

$$\varepsilon_M = \beta^{1-t} \tag{1}$$

el cual representa la distancia entre el número 1 y el número de punto flotante siguiente más próximo, es decir, es el número más pequeño en $\mathbb{F}(\beta, t, L, U)$ tal que $1 + \varepsilon_M > 1$, [1].

Representación de números reales I

Sea β un natural fijo tal que $\beta \geq 2$ y sea x un número real con un número finito de dígitos x_k con $0 \leq x_k < \beta$ para $k = -m, \ldots, n$. La notación:

$$x_\beta = (-1)^s (x_n x_{n-1} \ldots x_1 x_0 \,.\, x_{-1} x_{-2} \ldots x_{-m}), \quad x_n \neq 0$$

es llamada **representación posicional** de x con respecto a la base β.
Luego, cualquier número real puede ser aproximado por números que tienen una representación finita, es decir, fija una base β, se cumple:

$$\forall \varepsilon > 0 \, \forall \, x_\beta \in \mathbb{R}, \exists y_\beta \in \mathbb{R} \text{ tal que } |y_\beta - x_\beta| < \varepsilon, \tag{2}$$

donde y_β tiene representación posicional finita, [1].

Representación de números reales II

En efecto, dado el número real $x_\beta = x_n x_{n-1} \ldots x_0 \,.\, x_{-1} \ldots x_{-m} \ldots$ con un número de dígitos (que puede ser finito o infinito), para cualquier $r \geq 1$ se puede construir:

$$x_\beta^l = \sum_{k=0}^{r-1} x_{n-k}\beta^{n-k}, \quad x_\beta^u = x_\beta^l + \beta^{n-r+1}$$

que tienen r dígitos, tal que $x_\beta < x_\beta < x_\beta^u$ y $x_\beta^u - x_\beta^l = \beta^{n-r+1}$. Si elegimos r tal que $\beta^{n-r+1} < \varepsilon$, entonces tomando y_β igual a x_β^l o x_β^u se obtiene la desigualdad (2).

Esto justifica la representación de números reales en un computador.

Representación de números reales III

El hecho que sólo un subconjunto $\mathbb{F} \subset \mathbb{R}$ es representado en un computador trae severos problemas prácticos, en principio, la representación de cualquier número real $r \in \mathbb{R}$ mediante un elemento de $\mathbb{F}$. Además, observe que si $x, y \in \mathbb{R}$, luego, si operamos con ellos, el resultado puede no ser un elemento de $\mathbb{F}$, esto motiva definir una **arimética** sobre $\mathbb{F}$.
El modo más simple de resolver el primer problema es **redondear** $x \in \mathbb{R}$ de modo que el número redondeado pertenezca a $\mathbb{F}$. Una forma de hacerlo es como sigue: Dado $x \in \mathbb{R}$ en la notación científica, reemplazamos el valor de x por el valor $fl(x) \in \mathbb{F}$ definido como sigue:

$$fl(x) = (-1)^s(0.a_1a_2\ldots a_{t-1}\tilde{a}_t)\cdot\beta^e, \quad \tilde{a}_t = \begin{cases} a_t, & \text{si} \quad a_{t+1} < \beta/2, \\ a_t+1, & \text{si} \quad a_{t+1} \geq \beta/2 \end{cases} \tag{3}$$

Representación de números reales IV

Observe que $fl(x) = x$ para todo $x \in \mathbb{F}$. Más aún, $fl(x) \leq fl(y)$ siempre que $x \leq y$ para todo $x, y \in \mathbb{R}$. Si $x \in \mathbb{R}$ y $fl(x)$ su respectiva aproximación en punto flotante, entonces el error relativo es:

$$\delta(x) = \frac{fl(x) - x}{x}, \quad x \neq 0$$

luego:

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

¿ Es posible dar una cota para $\delta(x)$ que sea independiente de x ?

Representación de números reales V

Dentro del sistema de punto flotante el número cuyo valor absoluto es el más pequeño, es:

$$+(0.\underbrace{1000000..,0}_{t\ \text{dígitos}})_\beta \cdot \beta^L = \beta^{L-1}$$

para obtener el sucesor inmediato de β^{m-1}, debemos sumar:

$$+(0.\underbrace{0,00\ldots01}_{t\ \text{dígitos}})_\beta\,\beta^L = \beta^{L-t}$$

de lo anterior se observa que la distancia entre dos números consecutivos en el intervalo $[\beta^{L-1}, \beta^L]$ resulta igual a β^{L-t}.

Representación de números reales VI

Análogamente en el intervalo $[\beta^j, \beta^{j+1}]$ donde $L-1 \leq j \leq U-1$, la distancia entre dos números consecutivos **es siempre igual** a

$$\beta^{j+1-t}.$$

Luego, si la computadora representa a los números **por redondero** el error introducido es a lo más:

$$\frac{1}{2}\beta^{j+1-t},$$

y si representa **por truncamiento** el error introducido es a lo más:

$$\beta^{j+1-t},$$

Representación de números reales VII

obteniéndose de esta forma una medida para el error absoluto. Dado que:

$$\beta^j \leq |x| \Rightarrow \frac{1}{|x|} \leq \frac{1}{\beta^j}$$

al realizar el cociente entre β^j se obtiene que el error relativo resulta:

- Por redondeo: $|\delta(x)| \leq \frac{1}{2}\beta^{1-n}$.
- Por truncamiento: $|\delta(x)| \leq \beta^{1-n}$.

donde $\varepsilon_M = \beta^{1-n}$ es el epsilon de la máquina.

Representación de números reales VIII

Sea $\oplus$ la suma en el sistema de punto flotante $\mathbb{F}(\beta, t, L, U)$ y así resulta $1 \oplus \varepsilon_M = 1 + \beta^{1-n} > 1$. Sin embargo, si $0 < \varepsilon_1 < \varepsilon_M$ se obtiene:

$$1 \oplus \varepsilon_1 = 1$$

Proposición:

Si $x \in \mathbb{R}$ es tal que $x_{min} \leq |x| \leq x_{max}$, entonces:

$$fl(x) = x(1+\delta) \quad \text{tal que} \quad |\delta| \leq u$$

donde

$$u = \frac{1}{2}\beta^{1-t} = \frac{1}{2}\varepsilon_M$$

llamado **error de redondeo unitario**.

Representación de números reales IX

De la Proposición anterior tenemos la siguiente cota para el error relativo:

$$ER(x) = \frac{|x - fl(x)|}{|x|} \leq u$$

y para el error absoluto se tiene:

$$EA(x) = |x - fl(x)| \leq \beta^{e-t}|(a_1 \ldots a_t . a_{t+1} \ldots) - (a_1 \ldots \tilde{a}_t)|$$

de (3) resulta:

$$|(a_1 \ldots a_t . a_{t+1} \ldots) - (a_1 \ldots \tilde{a}_t)| \leq \beta^{-1}\frac{\beta}{2}$$

y así se obtiene:

$$EA(x) \leq \frac{1}{2}\beta^{e-t}.$$

Para más detalles revisar [1, 2].

Aritmética en punto flotante I

Usaremos el símbolo $\square$ para denotar una se las siguientes operaciones aritméticas: $+, -, \times, \div$.
Si $x, y \in \mathbb{F}$, es decir, números en punto flotante con t dígitos en la mantisa, entonces, en general, $x \square y \in \mathbb{F}$.
Por ejemplo, en $\mathbb{F}(10, 3, -5, 5)$ considere $x = 0{,}123 \times 10^4$ e $y = 0{,}456 \times 10^{-3}$, luego:

$$x + y = 1230 + 0{,}000456 = 1230{,}000456 \Rightarrow x + y = 0{,}1230000456 \times 10^4 \notin \mathbb{F}(10, 3, -5, 5)$$

Por tanto, en general, después de realizar una operación aritmética elemental $\square$ será necesario redondear el resultado. Esto puede resumirse en dos pasos:

- Calcular $x \square y$ con la mayor precisión posible.
- Redondear el resultado a t dígitos.

Este resultado es denotado por $fl(x \square y)$.

Aritmética en punto flotante II

Para $x, y \in \mathbb{R}$, las operaciones de suma, resta, multiplicación y división en el sistema de punto flotante $\mathbb{F}$, se definen:

1. $x + y := fl(fl(x) + fl(y))$.
2. $x - y := fl(fl(x) - fl(y))$.
3. $x \times y := fl(fl(x) \times fl(y))$.
4. $x/y := fl(fl(x) \div fl(y))$, $fl(y) \neq 0$, $y \neq 0$.

Suma/Resta en punto flotante

Sean $x, y \in \mathbb{F}(\beta, t, L, U)$. Luego $x \pm y$ se calcula como sigue:

1. **Alinear mantisas.** Tomar el número con menor exponente y desplazar su mantisa a la derecha hasta igualar los exponentes.
2. Sumar/Restar mantisas.
3. Normalizar el resultado si fuera necesario.
4. Redondear la mantisa al número de dígitos apropiado.
5. Normalizar si fuera preciso.

Ejemplo:

Considere $\mathbb{F}(10, 3, -5, 5)$ y sean $x = 0{,}433 \times 10^2 \in \mathbb{F}$ e $y = 0{,}745 \times 10^0$. Luego:

- Alineamos mantisas: $x = 0{,}433 \times 10^2$ e $y = 0{,}00745 \times 10^2$.
- Sumamos mantisas: $x + y = 0{,}44045 \times 10^2$
- No es necesario normalizar.
- Redondeamos a 3 dígitos: $x + y = 0{,}440 \times 10^2$.
- No es necesario normalizar.

Por tanto: $x + y = 0{,}440 \times 10^2$.

Ejemplo:

Cuando x e y son números reales se calcula $fl(x), fl(y)$ y se procede como el caso anterior. Por ejemplo, calcule en $\mathbb{F}(10, 3, -5, 5)$ la suma de $3\pi + 0{,}006589$. Veamos:

- $x = 3\pi = 3 \times (3{,}141592653\ldots) = 9{,}424777959\ldots = 0{,}9424777959 \times 10^1$ e $y = 0{,}00658 = 0{,}6589 \times 10^{-2}$.
- $fl(x) = 0{,}942 \times 10^1$ e $fl(y) = 0{,}659 \times 10^{-2}$.
- $fl(x) = 0{,}942 \times 10^1$ e $fl(y) = 0{,}000659 \times 10^1$.
- $fl(x) + fl(y) = 0{,}942659 \times 10^1$
- $fl(x) + fl(y) = 0{,}943 \times 10^1$
- $fl(fl(x) + fl(y)) = 0{,}943 \times 10^1$.

Así la suma $x + y$ en la máquina $\mathbb{F}(10, 3, -5, 5)$ resulta igual a $0{,}943 \times 10^1 = 9{,}43$.

Multiplicación/División en punto flotante

Sean $x, y \in \mathbb{F}(\beta, t, L, U)$. Luego $x \pm y$ se calcula como sigue:

1. Sumar/restar los exponentes.
2. Multiplicar/dividir mantisas.
3. Normalizar el resultado.
4. Redondear la mantisa al número de dígitos apropiado.
5. Normalizar si es preciso.
6. Determinar el signo del resultado.

Ejemplo:

Calcule en $\mathbb{F}(10, 3, -5, 5)$ el siguiente producto $0{,}003483 \times 3{,}159$. Veamos:

- $x = 0{,}3483 \times 10^{-2}$ e $y = 0{,}3159 \times 10^{1}$.
- $fl(x) = 0{,}348 \times 10^{-2}$ e $fl(y) = 0{,}316 \times 10^{1}$.
- $fl(x)fl(y) = 0{,}109968 \times 10^{-1}$
- $fl(fl(x)fl(y)) = 0{,}110 \times 10^{-1}$.

Así el producto xy en la máquina $\mathbb{F}(10, 3, -5, 5)$ resulta igual a $0{,}110 \times 10^{-1} = 0{,}0110$.
Si $*$ denota cualquier operación en $\mathbb{R}$, sea $\circledast$ la correspondiente operación en $\mathbb{F}(\beta, t, L, U)$. De la Proposición 4 resulta que existe δ tal que:

$$fl(x \circledast y) = (x * y)(1 + \delta), \quad \text{y} \quad |\delta| \leq u$$

Ejemplo:

Si x, y, z son números en un computador con longitud de palabra de 32 bits y $\beta = 2$, estime la cota superior que puede ser obtenida para el error relativo al calcular $z(x+y)$.

Solución:

Primero se calcular $x+y$. Esta operación aritmética da como resultado $fl(x+y)$, el cual difiere de $x+y$ por el redondeo. Por el axioma dado, se tiene que existe δ_1 tal que:

$$fl(x+y) = (x+y)(1+\delta_1), \quad |\delta_1| \leq 2^{-24}.$$

Como z es un número máquina, cuando se multiplica por el número máquina $fl(x+y)$, el resultado es el número máquina $fl(zfl(x+y))$. Estos números se diferencian del valor exacto por un δ_2 tal que:

$$fl(zfl(x+y)) = zfl(x+y)(1+\delta_2) \quad \text{donde} \quad |\delta_2| \leq 2^{-24}.$$

Ejemplo(Cont.)

De las dos ecuaciones anteriores resulta:

$$\begin{aligned} fl(zfl(x+y)) &= z(x+y)(1+\delta_1)(1+\delta_2) \\ &= z(x+y)(1+\delta_1+\delta_2+\delta_1\delta_2) \\ &\approx z(x+y)(1+\delta_1+\delta_2) \\ &= z(x+y)(1+\delta) \end{aligned}$$

donde el término $\delta_1\delta_2$ es ignorado, dado que $|\delta_1\delta_2| \leq 2^{-48}$ y $\delta = \delta_1 + \delta_2$. Observe que:

$$|\delta| \leq |\delta_1| + |\delta_2| \leq 2^{-24} + 2^{-24} = 2^{-23},$$

entonces la cota superior que se espera para el error relativo es 2^{-23}.

Representación IEEE 754

La precisión t de un sistema de números de punto flotante en una computadora estará limitada por la longitud de la palabra N disponible para representar un número. Con el fin de evitar una gran diversidad de sistemas de punto flotante incompatibles entre sí, a fines de la década de 1980 se desarrolló la norma o **stándard IEEE-754**, la cual es implementada en todas las computadoras actuales y aplicado también a otros sistemas. Esta norma define dos formatos básicos de punto flotante con base $\beta = 2$. La distribución de los N bits son en el siguiente orden:

Formato	Bits			
Precisión	Palabra (N)	Signo (s)	Exponente sesgado (E)	Mantisa (m)
Simple	32	1	8	23
Doble	64	1	11	52
Cuádruple	128	1	14	113

Precisión simple - IEEE 754

Considera el sistema de punto flotante $\mathbb{F}(2, 24, -126, 127)$ y una longitud de palabra igual a $N = 32$ bits. Dado $x \in \mathbb{R}$ en **notación científica normalizada** de la forma siguiente:

$$x = (-1)^s (d_1.d_2 \ldots d_{23} d_{24} d_{25} \ldots)_2 \times 2^e, \quad d_1 \neq 0$$

su respectiva notación en punto flotante es:

$$fl(x) = (-1)^s (d_1.d_2 \ldots d_{23} \tilde{d}_{24}) \times 2^e$$

Como en base 2 se cumple que $d_1 = 1$ siempre, entonces su representación en el computador es como sigue:

- Si tenemos n bits para el exponente, calculamos el sesgo como sigue $2^{n-1} - 1$. Para $n = 8$ el sesgo es $2^7 - 1 = 127$.
- Expresamos el exponente sesgado: $E = e + 127$ en base 2.
- Colocamos los dígitos $d_2, d_3, \ldots d_{24}$ en los bits asignados a la mantisa.
- El dígito d_1 es llamado **bit escondido**.

Ejemplo:

Representar $-31{,}125$ en precisión simple IEEE-754.

Solución:

- Número negativo: $s = 1$.
- Se tiene que $31 = 11111_2$ y $0{,}125 = 0{,}001_2$ entonces $31{,}125 = 11111{,}001_2$.
- Notación científica normalizada: $31{,}125 = 1{,}1111001_2 \times 2^4$.
- El exponentes es $e = 4$. Entonces el exponente sesgado es $E = e + 127 = 131 = 10000011_2$.
- La representación pedida es:

signo	exponente	mantisa
1	10000011	0000000000000001111001

Observación Precisión Simple IEEE-754

La representación usada para el exponente se conoce como **sesgada**, porque se calcula un nuevo exponente al sumar 127 al exponente original: $E = e + 127$. De esta forma, el exponente sesgado varía en el rango $1 \leq E \leq 254$ que pueden ser representados por un binario entero de 8 bits. Más aún, podemos incluir los valores del exponente para $L - 1 = -127(E = 0)$ y $U + 1 = 128(E = 255)$, ya que todos los enteros en el rango $0 \leq E \leq 255$ pueden ser representados como un binario entero sin signo de 8 bits. En efecto, con 8 bits tenemos $2^8 = 256$ combinaciones distintas, una para cada uno de los 256 números enteros del intervalo $[0, 255]$. El límite inferior, el " 0 ", corresponde a todos los bits igual a cero, mientras que el límite superior, el 255, corresponde a todos los dígitos igual a 1. Estos dos valores del exponente no representan números en punto flotante del sistema, pero serán usados para almacenar números especiales, como veremos a continuación.

Números especiales IEEE-754

- El **cero** es representado con ceros para el exponente y la mantisa.

0	00000000	0000000 00000000 00000000

- Los valores $+\infty$ y $-\infty$ son representados por:

0	11111111	0000000 00000000 00000000

1	11111111	0000000 00000000 00000000

- Si la secuencia de bits para el exponente está compuesta por todos los dígitos iguales a uno y la mantisa es no nula, es decir:

1	11111111	*xxxxxxx xxxxxxxx xxxxxxxx*

se tiene la ocurrencia de **NaN** (Not a Number) que representan expresiones inválidas como:

$$0 * \infty, \quad 0/0, \quad \infty/\infty, \quad \infty - \infty$$

Precisión doble - IEEE 754

Considera el sistema de punto flotante $\mathbb{F}(2, 53, -1022, 1023)$ y una longitud de palabra igual a $N = 64$ bits. Dado $x \in \mathbb{R}$ en **notación científica normalizada** de la forma siguiente:

$$x = (-1)^s (d_1.d_2 \ldots d_{63} d_{64} d_{65} \ldots)_2 \times 2^e, \quad d_1 \neq 0$$

su respectiva notación en punto flotante es:

$$fl(x) = (-1)^s (d_1.d_2 \ldots d_{63} \tilde{d}_{64}) \times 2^e$$

Como en base 2 se cumple que $d_1 = 1$ siempre, entonces su representación en el computador es como sigue:

- Si tenemos n bits para el exponente, calculamos el sesgo como sigue $2^{n-1} - 1$. Para $n = 11$ el sesgo es $2^{10} - 1 = 1023$.
- Expresamos el exponente sesgado: $E = e + 1023$ en base 2.
- Colocamos los dígitos $d_2, d_3, \ldots d_{64}$ en los bits asignados a la mantisa.
- El dígito d_1 es llamado **bit escondido**.

Errores

Error Absoluto

Sea x^* una aproximación de x. El **Error Absoluto** se define por:

$$EA(x) = |x - x^*|$$

Error Relativo

Sea x^* una aproximación de $x \neq 0$. El **Error Relativo** se define por:

$$EA(x) = \frac{|x - x^*|}{|x|}$$

Propagación de errores absolutos

Sea $\tilde{x}$ una aproximación para x e $\tilde{y}$ una aproximación para y. Luego, los errores absolutos son:

$$EA(x) = x - \tilde{x}, \quad EA(y) = y - \tilde{y}.$$

Por tanto resulta:

$$x + y = EA(x) + \tilde{x} + EA(y) + \tilde{y} \Rightarrow x + y = \tilde{x} + \tilde{y} + (EA(x) + EA(y))$$

así tenemos:

$$EA(x + y) = EA(x) + EA(y)$$

De forma análoga se obtiene:

$$EA(x - y) = EA(x) - EA(y)$$

Demuestre que:

$$EA(xy) = \tilde{x}EA(y) + \tilde{y}EA(x)$$

$$EA(x/y) = \frac{EA(x)}{\tilde{y}} - \frac{\tilde{x}EA(y)}{\tilde{y}^2}$$

Propagación de errores relativos

Para la suma y sustracción:

$$ER(x \pm y) = \frac{EA(x) \pm EA(y)}{\tilde{x} \pm \tilde{y}} = \frac{EA(x)}{\tilde{x} \pm \tilde{y}} \pm \frac{EA(y)}{\tilde{x} \pm \tilde{y}} = \frac{\tilde{x}}{\tilde{x} \pm \tilde{y}} ER(x) \pm \frac{\tilde{y}}{\tilde{x} \pm \tilde{y}} ER(y)$$

Propagación de errores relativos

Observe que para la multiplicación:

$$xy = (\tilde{x} + EA(x))(\tilde{y} + EA(y)) = \tilde{x}\tilde{y} + \tilde{x}EA(x) + \tilde{y}EA(y) + EA(x)EA(y)$$

por tanto, si $\tilde{x}$ e $\tilde{y}$ son mayores que 1 (en valor absoluto), los términos $\tilde{x}EA(x)$ e $\tilde{y}EA(y)$ indican que hay una posibilidad de que los errores originales $EA(x)$ y $EA(y)$ sean magnificados. Sin embargo, si analizamos los errores relativos se tiene una percepción más clara, pues al reordenar los términos:

$$xy - \tilde{x}\tilde{y} = \tilde{x}EA(x) + \tilde{y}EA(y) + EA(x)EA(y)$$

si x e y son no nulos, entonces podemos divir entre xy, así resulta:

$$ER(xy) = \frac{EA(xy)}{xy} = \frac{\tilde{x}EA(y) + \tilde{y}EA(x) + EA(x)EA(y)}{xy} = ER(x) + ER(y)$$

siempre que $\tilde{x}$ e $\tilde{y}$ sean buenas aproximaciones de x e y (por tanto $\tilde{x}/x \approx 1$, $\tilde{y}/y \approx 1$ y $(EA(x)/x)(EA(y)/y) \approx 0$).

Ejemplo:

Determine el valor absoluto cuando p es aproximado por p^*, donde:

1. $p = 0{,}3000 \times 10^1$ y $p^* = 0{,}3100 \times 10^1$.
2. $p = 0{,}3000 \times 10^{-3}$ y $p^* = 0{,}3100 \times 10^{-3}$.
3. $p = 0{,}3000 \times 10^4$ y $p^* = 0{,}3100 \times 10^4$.

Solución:

1. $EA(p) = 0{,}1$ y $ER(p) = 0{,}333\overline{3} \times 10^{-1}$.
2. $EA(p) = 0{,}1 \times 10^{-4}$ y $ER(p) = 0{,}333\overline{3} \times 10^{-1}$.
3. $EA(p) = 0{,}1 \times 10^{3}$ y $ER(p) = 0{,}333\overline{3} \times 10^{-1}$.

Ejemplo:

Suponga que usted recibe una calculadora super moderna como regalo de cumpleaños, capaz de almacenar 4 dígitos en la mantisa utilizando redondeo. Ansioso por usar la nueva calculadora, consideró $x = 17534$ e $y = 21178$.

1. Determine los errores relativos de x, y.
2. Después de calcular $x + y$ e xy, calcule el respectivo error relativo.

Solución:

Debido a que se usa 4 dígitos en la mantisa y redondeo, resulta:

$$x = 0{,}17534 \times 10^5 \Rightarrow x \approx 0{,}1753 \times 10^5 \Rightarrow \tilde{x} = 17530$$

$$y = 0{,}21178 \times 10^5 \Rightarrow y \approx 0{,}2118 \times 10^5 \Rightarrow \tilde{y} = 21180$$

1. Calculamos los errores absolutos:

$$EA(x) = x - \tilde{x} = 17534 - 17530 = 4, \quad EA(y) = 21178 - 21180 = -2.$$

   Los errores relativos son:

$$ER(x) = 4/17530 = 2{,}281 \times 10^{-4}, \quad ER(y) = -\frac{2}{21180} = 9{,}442 \times 10^{-5}.$$

2. Ejercicio.

Bibliografía

- A. Quarteroni, R. Sacco, and F. Saleri, *Numerical mathematics*, vol. 37. Springer Science & Business Media, 2010.
- G. Hämmerlin and K.-H. Hoffmann, *Numerical mathematics*. Springer Science & Business Media, 2012.

Otras referencias

- Numerical Analysis: Mathematics of Scientific Computing, Third Edition David Kincaid: University of Texas at Austin, Austin, TX, Ward Cheney.
- Numerical Methods Using Matlab, 4th Edition John H. Mathews, California State University, Fullerton, Kurtis K. Fink, Northwest Missouri State University
- Numerical Lineal Algebra. Lloyd N. Trefethen and David Bau, III xii+361 pages. SIAM, 1997
- Elementary Numerical Analysis, 3rd Edition Kendall Atkinson, Weimin Han