MATEMÁTICAS III

CÁLCULO NUMÉRICO

GRADO INGENIERÍA QUÍMICA

Departamento de Matemática Aplicada

Universidad de Salamanca

Mabel Asensio Sevilla

Julio-2011
Revisado Septiembre 2015

Índice general

Capítulo 0

Errores

0.1. Introducción

Un método numérico es un método "aproximado" para la resolución de un problema matemático, éste, a su vez, puede representar una modelización matemática de un problema físico, químico o del mundo real. En la práctica, la solución al problema real que nosotros conoceremos será la que nos proporcione el método numérico, que en general no va a coincidir con la solución exacta del problema real, ya que va a estar afectada de diversos tipos de errores:

- **Experimentales**: la presencia de errores puede comenzar en la misma formulación del problema real, pues los datos se pueden haber obtenido de ciertas mediciones u otras observaciones experimentales, siempre susceptibles de errores.
- **De modelización**: debidos a la aproximación de la realidad del modelo matemático elegido.
- **De discretización o de truncamiento**: debidos a la propia naturaleza del método numérico elegido para resolver el problema matemático.
- **De redondeo**: debidos a las restricciones aritméticas de los ordenadores y la limitada capacidad humana, frente a la infinidad de cifras decimales de los números reales. Es necesario delimitar su acumulación, ya que es habitual llevar a cabo un elevado número de operaciones en la resolución de los métodos numéricos.

0.2. Error absoluto y relativo

Sea x el valor exacto de un número real y x_0 un valor aproximado. Definimos:

- **Error absoluto de x:** $\epsilon(x) = |x - x_0|$.

- **Error relativo de x:** $e(x) = \frac{\epsilon(x)}{|x|}$**.**

El error absoluto da una referencia *cuantitativa* de la bondad de la aproximación, medida por la distancia que separa el valor exacto del aproximado. El error relativo proporciona una referencia *cualitativa*, en tanto en cuanto refleja la proporción del error absoluto con respecto a la magnitud que se trata de aproximar: en este sentido, no es lo mismo un error de una unidad cuando se aproxima el valor exacto de $\pi = 3.14159\ldots$ que cuando se aproxima el valor exacto del número de Avogadro (aproximadamente igual a $6.022 \cdot 10^{23}$).

Ejemplo 0.1 Comparar los errores absolutos y relativos en las aproximaciones 3.1 de 3 y 3099 de 3000.

Diremos que la aproximación x_0 tiene p cifras decimales exactas si $\epsilon \leq 10^{-p}$. Obsérvese que esto no indica que hayan de coincidir las p primeras cifras decimales de x y x_0. Por ejemplo, si $x = 1$ y $x_0 = 0.9999$ se tiene que $\epsilon \leq 10^{-4}$ y, por tanto, 0.9999 aproxima a 1 con las cuatro cifras decimales exactas (aunque no coincida ninguno de los decimales de ambas cifras).

0.3. Errores de redondeo

Dado un número real x expresado en su forma decimal

$$x = a_n a_{n-1} \ldots a_0 . a_{-1} a_{-2} \ldots a_{-k} a_{-k-1} \ldots, \quad 0 \leq a_k \leq 9, k \in \mathbb{Z}$$

se llama parte decimal de x a la secuencia $a_{-1}a_{-2}\ldots a_{-k}a_{-k-1}\ldots$. Por ejemplo, 17.352 tiene por parte decimal a 352.

Si efectuamos los nuestros cálculos en una máquina que puede representar números con k cifras decimales, esta representación se puede hacer de dos formas: por *truncamiento*, cortando la parte decimal para dejarla en k cifras;

$$x_t = a_n a_{n-1} \ldots a_0 . a_{-1} a_{-2} \ldots a_{-k}$$

o por *redondeo*, si la cifra a_{-k-1} es menor que 5, entonces el resultado es el mismo que por truncamiento, y si la cifra a_{-k-1} es igual o mayor que 5, entonces se añade 1 a la cifra k-ésima y se trunca el número restante.

Observar que $|x - x_t| \leq 10^{-k}$ mientras que $|x - x_r| \leq \frac{1}{2}10^{-k}$

Capítulo 1

Ecuaciones y sistemas de ecuaciones no lineales

En este tema repasaremos uno de los problemas básicos del cálculo numérico:

Dada una función f real de variable real, hallar los valores de la variable x que satisfagan la ecuación $f(x) = 0$.

La función f puede ser polinómica, transcendente o incluso puede que no dispongamos de una expresión explícita de la misma, por ejemplo, si es la solución de una ecuación diferencial. Los valores que buscamos son los valores $\bar{x}$ que anulan dicha función. A estos valores se les denomina *raíces* o *soluciones* de la ecuación, o también *ceros* de la función $f(x)$. Geométricamente representan las abscisas de los puntos de corte de la gráfica $y = f(x)$ con el eje OX.

Definición 1.1 Multiplicidad de una raíz.

Una raíz $\bar{x}$ de la ecuación $f(x) = 0$ se dice que tiene *multiplicidad* n si

$$f(\bar{x}) = f'(\bar{x}) = f''(\bar{x}) = \ldots = f^{n-1)}(\bar{x}) = 0 \text{ y } f^{n)}(\bar{x}) \neq 0$$

Si la multiplicidad de una raíz es 1, diremos que es una raíz *simple*.

En general, las raíces de una ecuación no lineal no se pueden calcular de forma exacta, sino que se recurre a métodos numéricos que permiten obtener aproximaciones numéricas de las mismas. El objetivo de este capítulo es presentar algunos de estos métodos, pero antes veremos algunos resultados que nos permitirán localizar y separar previamente las raíces de una ecuación. Posteriormente veremos algunos de los métodos más clásicos para el cálculo de raíces de ecuaciones, como el método de la bisección, el método de punto fijo y el método de Newton, y algunas de sus modificaciones. Por último veremos como adaptar alguno de estos métodos al caso de sistemas no lineales, como el método de punto fijo y los métodos Quasi-Newton.

El problema de hallar las raíces de una ecuación, $f(x) = 0$, aparece frecuentemente en ingeniería. Por ejemplo, para calcular el volumen V de un gas de van der Waals como función de la temperatura absoluta T, la presión P, el número de moles n y los parámetros de Van der Walls a, b, la ecuación de estado es,

$$\Big(P + \frac{a}{V^2}\Big)(V - b) = nRT,$$

que conduce a una ecuación polinómica de grado 3 en V,

$$PV^3 - (Pb + nRT)V^2 + aV - ab = 0.$$

En teoría de la difracción de la luz necesitamos las raíces de la equación

$$x - \tan x = 0.$$

En el cálculo de órbitas planetarias necesitamos las raíces de la ecuación de Kepler

$$x - a \sin x = b, \tag{1.1}$$

para varios valores de a y b. En teoría de la combustión

$$x = \delta e^{\gamma x}, \tag{1.2}$$

para varios valores de γ y δ.

1.1. Localización y separación de raíces de una ecuación.

El proceso de localización y separación de raíces de una ecuación es una tarea previa a la aplicación de un método numérico para el cálculo de estas raíces. Consiste en obtener información de las zonas donde se encuentran las raíces reales de la ecuación, para posteriormente buscar intervalos $[a_1, b_1], [a_2, b_2], \ldots$ que contengan una y sólo una de estas raíces.

Dada una ecuación no lineal $f(x) = 0$ con n raíces reales distintas, $\bar{x}_1, \ldots, \bar{x}_n$, se pretende hallar n intervalos disjuntos $I_i = [a_i, b_i]$ para $i = 1, \ldots, n$ de modo que $\bar{x}_i \in I_i$, $i = 1, \ldots, n$.

A veces puede obtenerse algún tipo de información gráfica si se transforma la ecuación $f(x) = 0$ en otra del tipo $g(x) = h(x)$ y se cotejan los puntos de corte de las gráficas de $g(x)$ y $h(x)$. Esto sólo da una idea gráfica de donde están los ceros, pero no puede servir como prueba de localización y separación de las raíces de una ecuación, ya que en algunos casos la información gráfica que proporciona un ordenador puede no ajustarse a la realidad.

Es el estudio analítico de la función $f(x)$ el que puede aportarnos la información necesaria, abordando diversos aspectos:

(a) *Crecimiento.* Estudio de los intervalos de crecimiento de $f(x)$. Si $[a, b]$ es un intervalo de crecimiento (resp. decrecimiento) monótono de $f(x)$, entonces a lo más habrá una única raíz de $f(x) = 0$ en ese intervalo. El estudio de los intervalos de crecimiento de una función supone hallar los ceros de su derivada, lo que en ocasiones puede ser tanto o más complejo que el problema de partida.

(b) *Teorema de Bolzano.* Se trata de aplicar el toerema de Bolzano a cada uno de los intervalos en los que se sospecha que hay una raíz. Esto require que se satisfagan las hipótesis de este teorema, lo cual no siempre ocurre.

(c) *Sucesiones de Sturm.* No siempre se conoce como determinar una sucesión de Sturm para una función dada. Estudiaremos el caso de las funciones polinómicas.

Ecuaciones polinómicas

Dado un polinomio

$$P(x) = a_0x^n + a_1x^{n-1} + \ldots + a_{n-1}x + a_n$$

donde $a_i \in \mathbb{R}$ para $i = 0, 1, \ldots, n$ diremos que $P(x) = 0$ es una *ecuación polinómica.*

El *teorema fundamental del Álgebra* nos dice que la ecuación polinómica $P(x) = 0$ con coeficientes reales tiene n raíces reales y complejas contadas con sus multiplicidades. Las raíces complejas aparecen en pares conjugados (si $a + bi$ es raíz, entonces $a - bi$ también lo es).

Veamos algunos resultados que permiten localizar ceros de un polinomio.

Proposición 1.1 Si $\bar{x}$ es una raíz de $P(x) = 0$ entonces:

$$\frac{1}{1 + \frac{A}{|a_n|}} < |\bar{x}| < 1 + \frac{A}{|a_0|} \text{ siendo } A = max_{i \geq 1}|a_i|$$

Proposición 1.2 (Regla de Laguerre) Dado $c \in \mathbb{R}^+$ podemos escribir $P(x) = (x - c)C(x) + r$ con $C(x) = b_0x^{n-1} + \ldots + b_{n-2}x + b_{n-1}$ y $r \in \mathbb{R}$. Si $r \geq 0$ y $b_i \geq 0$ para $i = 0, 1, \ldots, n-1$ ó $r \leq 0$ y $b_i \leq 0$ para $i = 0, 1, \ldots, n-1$, entonces el número real c es una cota superior de las raíces positivas de la ecuación.

Proposición 1.3 Sea $R(x) = a_nx^n + a_{n-1}x^{n-1} + \ldots + a_0$, es decir, $R(x) = x^nP(\frac{1}{x})$ para $x \neq 0$. Por tanto $P(\bar{x}) = 0 \Leftrightarrow R(\frac{1}{\bar{x}}) = 0$. Esto nos permite obtener una cota inferior de las raíces positivas de $P(x) = 0$ puesto que si c' una cota superior de las raíces positivas de $R(x) = 0$ obtenida mendiante la regla de Laguerre, entonces $\frac{1}{c'}$ es una cota inferior de las raíces positivas de $P(x) = 0$.

Proposición 1.4 Sea $H(x) = P(-x)$, entonces $P(\bar{x}) = 0 \Leftrightarrow H(-\bar{x}) = 0$, esto es, si $\bar{x}$ es una raíz negativa de $P(x) = 0$, entonces $-\bar{x}$ es una raíz positiva de $H(x) = 0$. Esto nos permite obtener cotas inferiores y superiores de las raíces negativas de $P(x) = 0$: si c y c' son cotas superior e inferior de las raíces positivas de $H(x) = 0$, respectivamente, entonces $-c'$ es cota superior de las raíces negativas de $P(x) = 0$ y $-c$ cota inferior de las raíces negativas.

Ejemplo 1.1 Dado el polinomio $P(x) = x^4 + 2x^3 - 3x^2 - 4x - 1$, acotar las raíces de $P(x) = 0$ tanto como se pueda.

Definición 1.2 Una *sucesión de Sturm* para una función $f(x)$ en $[a, b]$ es un conjunto $f_0(x) = f(x)$, $f_1(x), \ldots, f_n(x)$ de funciones continuas en dicho intervalo tales que:

(a) $f_n(x) \neq 0 \forall x \in [a, b]$, es decir, el signo de $f_n(x)$ permanece constante en $[a, b]$

(b) Si $f_i(c) = 0$ con $c \in [a, b]$ entonces $f_{i-1}(c) \cdot f_{i+1}(c) < 0$, es decir, tienen signos opuestos y no se anulan en c.

(c) Si $f_0(c) = 0$ con $c \in [a, b]$ entonces $\frac{f_0(x)}{f_1(x)}$ pasa de negativa a positiva en c

La importancia de las sucesiones de Sturm radica en el resultado siguiente:

Teorema 1.1 (Teorema de Sturm) Sea $f_0(x), f_1(x), \ldots, f_n(x)$ una sucesión de Sturm para $f(x) = f_0(x)$ en el intervalo $[a, b]$. Consideremos las siguientes sucesiones en las que $sig(d)$ denota el signo de d (indistintamente $\pm$ cuando $d = 0$)

$$sig(f_0(a)), sig(f_1(a)), \ldots, sig(f_n(a))$$
$$sig(f_0(b)), sig(f_1(b)), \ldots, sig(f_n(b)).$$

Denotemos por N_1 el número de cambios de signo en la primera sucesión, y por N_2 el número de cambios de signo en la segunda (siempre ha de ser $N_1 \geq N_2$). Entonces el número de raíces de la ecuación $f_0(x) = 0$ en el intervalo $[a, b]$ viene dado por $N_1 - N_2$.

Por tanto, si conocemos una sucesión de Sturm para una función $f(x)$, podremos separar todos sus ceros reales. Lamentablemente, no hay procedimientos sistemáticos para formar sucesiones de Sturm para cualesquiera funciones dadas, salvo contadas excepciones, como es el caso de los polinomios, para los que la sucesión de Sturm se construte de la siguiente forma:

$$f_0(x) = P(x), \quad f_1(x) = P'(x), \quad f_{i+1}(x) = -r_i(x)$$

donde $r_i(x)$ es el resto de dividir f_{i-1} entre f_i, es decir, $f_{i-1}(x) = c_i(x) \cdot f_i(x) + r_i(x)$.

Ejemplo 1.2 Dado el polinomio $P(x) = x^4 + 2x^3 - 3x^2 - 4x - 1$, construir una sucesión de Sturm para este polinomio y separar las raíces de $P(x) = 0$.

1.2. Ecuaciones no lineales

1.2.1. Método de bisección.

Se basa en la aplicación reiterada del **teorema de Bolzano**: *Si f es una función continua definida sobre un intervalo cerrado $[a, b]$ tal que $f(a).f(b) < 0$ entonces f debe tener un cero en (a, b).*

El método de la bisección explota esta propiedad de la siguiente manera:

(a) Tomamos $c = \frac{a+b}{2}$

(b)
- Si $f(a).f(c) < 0$, entonces f tiene un cero en (a, c) y hacemos $b \longleftarrow c$
- Si $f(a).f(c) > 0$, entonces $f(c).f(b) < 0$ y f tiene un cero en (c, b) y hacemos $a \longleftarrow c$
- Si $f(a).f(c) = 0$, está claro que $f(c) = 0$ y ya hemos encontrado un cero.

En las dos primeras situaciones del punto 2, hemos reducido el problema a la búsqueda de ceros en un intervalo de longitud la mitad que la del intervalo original y repetimos el proceso.

La situación $f(c) = 0$ es poco probable que se dé en la práctica, debido a los errores de redondeo. Así, el criterio para concluir no debe depender de que $f(c) = 0$, sino que permitiremos una tolerancia razonable, tal como $|f(c)| < \varepsilon$, para cierto ε suficientemente pequeño.

Pseudo-código del algoritmo de la bisección

- entrada $a, b, M, \delta, \varepsilon$
- $u \longleftarrow f(a)$, $v \longleftarrow f(b)$, $e \longleftarrow b - a$
- si $sign(u) = sign(v)$ entonces parar
- para $k = 1, ..., M$ hacer
  - $e \longleftarrow \frac{e}{2}$, $c \longleftarrow a + e$, $w \longleftarrow f(c)$
  - salida k, c, w, e
  - si $|e| < \delta$ or $|w| < \varepsilon$ entonces parar
  - si $sign(w) \neq sign(u)$ entonces $b \longleftarrow c$, $v \longleftarrow w$
  - sino $a \longleftarrow c$, $u \longleftarrow w$
  - fin condicional
- fin bucle

Varias de las partes de este pseudo-código necesitan una explicación adicional. En primer lugar, el punto medio c se calcula como $c \longleftarrow a + \frac{b-a}{2}$ en lugar de $c \longleftarrow \frac{a+b}{2}$. Al hacerlo así se sigue la estrategia general de que, al efectuar cálculos numéricos, es mejor calcular una cantidad añadiendo un pequeño término de corrección a una aproximación obtenida previamente. En segundo lugar, es mejor determinar si la función cambia de signo en el intervalo recurriendo a que $sign(w) \neq sign(u)$ en lugar de utilizar $w.u < 0$ ya que esta última requiere una multiplicación innecesaria. Por otra parte e corresponde al cálculo de la cota del error que se establece más adelante.

En el algoritmo hay tres criterios que pueden detener la ejecución:

- M, señala el máximo número de iteraciones, un algoritmo correctamente diseñado tiene que ser finito.
- Por otra parte la ejecución del programa se puede detener, ya sea cuando el error es suficientemente pequeño o cuando lo es el valor de $f(c)$. Los parámetros δ y ε controlan esta situación. Se pueden dar ejemplos en los que se satisface uno de los dos criterios sin que el otro se satisfaga.

Teorema 1.2 : Análisis del error

Sea f continua en $[a,b] = [a_0, b_0]$ con $f(a).f(b) < 0$. Sean $[a_0, b_0], [a_1, b_1], ..., [a_n, b_n]$ los intervalos sucesivos generados por el método de la bisección. Entonces los límites $\lim_{n\to\infty} a_n$, $\lim_{n\to\infty} b_n$ existen, son iguales y representan un cero de f. Si $r = \lim_{n\to\infty} c_n$ con $c_n = \frac{a_n+b_n}{2}$, entonces

$$|r - c_n| \leq \frac{b_0 - a_0}{2^{n+1}}$$

Demostración:

Por la propia construcción del algoritmo, tenemos,

$$a_0 \leq a_1 \leq ... \leq b_0$$

$$b_0 \geq b_1 \geq ... \geq a_0$$

$$b_{n+1} - a_{n+1} = \frac{b_n - a_n}{2}, \quad n \geq 0$$

La sucesión $\{a_n\}$ converge debido a que es creciente y está acotada superiormente.

La sucesión $\{b_n\}$ converge por ser decreciente y estar acotada inferiormente.

Además, se tiene,

$$b_n - a_n = \frac{b_{n-1} - a_{n-1}}{2} = ... = \frac{b_0 - a_0}{2^n}$$

Así

$$\lim_{n\to\infty} b_n - \lim_{n\to\infty} a_n = \lim_{n\to\infty} \frac{b_0 - a_0}{2^n} = 0$$

Si escribimos $r = \lim a_n = \lim b_n$, tomando límites en la desigualdad $f(a_n).f(b_n) < 0$, resulta $f(r)^2 = f(r).f(r) \leq 0$, es decir $f(r) = 0$.

Finalmente, en la etapa en la que se ha construido el intervalo $[a_n, b_n]$, si se detiene en este momento el algoritmo, sabemos que la raíz de la ecuación se encuentra en ese intervalo. La mejor estimación para

esa raíz será el punto medio $c_n = \frac{a_n+b_n}{2}$ y el error cometido verificará

$$|r - c_n| \leq \frac{b_n - a_n}{2} \leq \frac{1}{2}\frac{b_0 - a_0}{2^n} = \frac{b_0 - a_0}{2^{n+1}}$$

■

1.2.2. El método de punto fijo

Utilizaremos este método para resolver ecuaciones de la forma $x = g(x)$. Observemos que si queremos hallar las raíces de una ecuación $f(x) = 0$, podemos ponerla de la forma anterior, por ejemplo, haciendo $g(x) = x - f(x)$ o más generalmente $g(x) = x - \rho(x)f(x)$ donde $\rho(x) \neq 0$, es una función adecuadamente elegida, que puede ser constante o no.

De manera más precisa el problema planteado es el siguiente:

Dada $g : [a, b] \longrightarrow [a, b]$ *función continua, hallar* $x \in [a, b]$ *tal que* $x = g(x)$.

Teorema 1.3 : Existencia del punto fijo.

Sea $g : [a, b] \longrightarrow [a, b]$ continua, entonces existe al menos un $x \in [a, b]$ tal que $x = g(x)$.

Demostración:

Si $a = g(a)$ o $b = g(b)$ entonces a o b es una solución. Supongamos pues que $a \neq g(a)$ y que $b \neq g(b)$.

Pongamos $f(x) = x - g(x)$, tendremos, $f(a) = a - g(a) < 0$ y $f(b) = b - g(b) > 0$. Por el teorema de Bolzano existe al menos $\bar{x} \in (a, b)$ tal que $f(\bar{x}) = 0$, es decir, $\bar{x} = g(\bar{x})$.
■

Teorema 1.4 : Unicidad del punto fijo.

Sea $g : [a, b] \longrightarrow [a, b]$ continua y contractiva, es decir, existe $k < 1$ tal que $|g(x) - g(y)| \leq k|x - y|, \forall x, y \in [a, b]$, entonces el punto fijo $\bar{x}$ es único.

Demostración:

Sean $\bar{x_1}$ y $\bar{x_2}$ dos puntos fijos de g, $\bar{x_1} \neq \bar{x_2}$, es decir, $\bar{x_1}, \bar{x_2} \in [a, b]$, $\bar{x_1} = g(\bar{x_1})$ y $\bar{x_2} = g(\bar{x_2})$.

$$|\bar{x_1} - \bar{x_2}| = |g(\bar{x_1}) - g(\bar{x_2})| \leq k|\bar{x_1} - \bar{x_2}| < |\bar{x_1} - \bar{x_2}|$$

■

Observación: Si g es diferenciable y existe un número $k < 1$ tal que $|g'(x)| < k$ para todo $x \in [a, b]$, entonces para $\xi \in [a, b]$, resulta $|g(x) - g(y)| = |g'(\xi)||x - y| \leq k|x - y|$.

El algoritmo de punto fijo o iteración funcional es:

- Dado un $x_0 \in [a, b]$,
- calculado x_n, obtenemos $x_{n+1} = g(x_n)$

Pseudo-código del algoritmo de punto fijo

- entrada $x_0,\ M,\ \varepsilon$
- $x \leftarrow x_0$
- Para $k = 1, ..., M$ hacer
  - $x_1 \leftarrow x$, $x \leftarrow g(x)$, $e \leftarrow |x - x_1|$
  - salida $k,\ x,\ e$
  - si $e < \varepsilon$ entonces parar
- fin bucle

Teorema 1.5 : Teorema de convergencia y análisis del error.

Sea $g : [a, b] \longrightarrow [a, b]$ continua y contractiva, es decir, tal que

$$|g(x) - g(y)| < k|x - y| \quad \forall x, y \in [a, b], \quad k < 1$$

entonces la sucesión x_n generada por el algoritmo de punto fijo verifica

$$\lim_{n\to\infty} x_n = \bar{x}$$

siendo $\bar{x}$ el único punto fijo de g, y además,

$$|x_n - \bar{x}| \leq \frac{k^n}{1 - k}|x_1 - x_0|$$

Demostración:

$$|x_{n+1} - \bar{x}| = |g(x_n) - g(\bar{x})| \leq k|x_n - \bar{x}| \leq ... \leq k^n|x_0 - \bar{x}|$$

de donde

$$\lim_{n\to\infty} |x_n - \bar{x}| \leq |x_0 - \bar{x}| \lim_{n\to\infty} k^n = 0$$

pues $k < 1$.

Por otro lado,

$$|x_{n+1} - x_n| = |g(x_n) - g(x_{n-1})| \leq k|x_n - x_{n-1}| \leq ... \leq k^n|x_1 - x_0|$$

Para $m > n \geq 1$ tendremos,

$$\begin{aligned} |x_m - x_n| &= |x_m - x_{m-1} + x_{m-1} - x_{m-2} + x_{m-2} - ... + x_{n+1} - x_n| \\ &\leq |x_m - x_{m-1}| + |x_{m-1} - x_{m-2}| + ... + |x_{n+1} - x_n| \\ &\leq (k^{m-1} + k^{m-2} + ... + k^n)|x_1 - x_0| \\ &\leq k^n(1 + k + ... + k^{m-n-1})|x_1 - x_0| \end{aligned}$$

Pasando al límite cuando $m \to \infty$ se obtiene

$$|x_n - \bar{x}| \leq \frac{k^n}{1-k}|x_1 - x_0|$$

■

Definición 1.3 Orden de convergencia, convergencia lineal, cuadrática y orden α.

Supongamos que $\{x_n\}_{n=1}^{\infty}$ es una sucesión convergente cuyo límite es p. Sea $e_n = x_n - p$. Si existen dos constantes $\lambda > 0$ y $\alpha > 0$ tales que

$$lim_{n\to\infty} \frac{|e_{n+1}|}{|e_n|^\alpha} = \lambda$$

diremos que $\{x_n\}$ converge hacia p, con orden α. En particular:

- Si $\alpha = 1$, diremos que la convergencia es lineal.
- Si $\alpha = 2$, diremos que la convergencia es cuadrática.
- Si $1 < \alpha < 2$, diremos que la convergencia es superlineal.

Orden de convergencia del método de punto fijo

El método de punto fijo tiene convergencia lineal si g' es continua y $g'(\bar{x}) \neq 0$ siendo $\bar{x}$ el punto fijo de g. En efecto,

$$e_{n+1} = x_{n+1} - \bar{x} = g(x_n) - g(\bar{x}) = g'(\xi_n)(x_n - \bar{x}) = g'(\xi_n)e_n$$

donde $\xi_n \in [x_n, \bar{x}]$, finalmente

$$lim_{n\to\infty} \frac{|e_{n+1}|}{|e_n|} = lim_{n\to\infty}|g'(\xi_n)| = |g'(\bar{x})| = \lambda > 0$$

1.2.3. El método de Newton

Consideremos de nuevo el problema de buscar las raíces de una ecuación del tipo $f(x) = 0$. Si $f(x)$, $f'(x)$ y $f''(x)$ son continuas cerca de una raíz $\bar{x}$, esta información adicional sobre la naturaleza de $f(x)$ puede usarse para desarrollar algoritmos que produzcan sucesiones $\{x_k\}$ que converjan a $\bar{x}$ más rápidamente que el método de bisección o de punto fijo. El método de Newton-Raphson, o simplemente de Newton, que descansa en la continuidad de $f'(x)$ y $f''(x)$, es uno de los algoritmos más útiles y mejor conocidos.

Supongamos que $\bar{x}$ es una raíz de la ecuación anterior y supongamos además que f es dos veces derivable con continuidad. Si x es una aproximación de $\bar{x}$, usando el desarrollo de Taylor, podemos escribir,

$$0 = f(\bar{x}) = f(x) + f'(x)(\bar{x} - x) + \frac{1}{2}f''(\xi)(\bar{x} - x)^2$$

Si x está cerca de $\bar{x}$, $(\bar{x} - x)^2$ es un número pequeño y podremos despreciar el último término frente a los otros, y $\bar{x}$ vendrá dado aproximadamente por

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

Como hemos despreciado el término cuadrático este valor no será exactamente $\bar{x}$, pero es de esperar que será una mejor aproximación que el valor x de partida. De aquí se obtiene el algoritmo de Newton:

- x_0, valor cercano a $\bar{x}$.
- Calculado x_n, obtenemos x_{n+1},

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)} \text{ Fórmula de Newton-Raphson}$$

El método de Newton también es conocido como método de la tangente, ya que si trazamos la tangente a la curva $y = f(x)$ en el punto $(x_n, f(x_n))$ obtenemos la recta $y = f(x_n) + f'(x_n)(x - x_n)$ que corta al eje $y = 0$ en el punto de abscisa $x = x_n - \frac{f(x_n)}{f'(x_n)}$, que es precisamente el valor de x_{n+1} en la fórmula de Newton-Raphson.

El método de Newton se puede interpretar como un método de punto fijo, pues buscamos el punto fijo de la función $x - f(x)/f'(x)$.

Pseudocódigo del algoritmo de Newton

- entrada $x_0, M, \delta, \varepsilon$

- $v \leftarrow f(x_0)$
- salida $0, x_0, v$
- si $|v| < \varepsilon$ entonces parar
- para $k = 1, ..., M$ hacer
  - $x_1 \leftarrow x_0 - \frac{v}{f'(x_0)}$
  - $v \leftarrow f(x_1)$
  - salida k, x_1, v
  - si $|x_1 - x_0| < \delta$ o $|v| < \varepsilon$ entonces parar
  - $x_0 \leftarrow x_1$
- fin bucle

Un inconveniente de la sucesión generada por la fórmula de Newton-Raphson es que no siempre se tiene asegurada la convergencia hacia $\bar{x}$, incluso tomando x_0 próximo a $\bar{x}$. Nos preguntamos que condiciones hay que exigir a x_0 y a f para que la sucesión $\{x_n\}$ generada por la fórmula de Newton-Raphson sea convergente a $\bar{x}$.

El resultado más general de convergencia del método de Newton es el siguiente:

Teorema 1.6 : Convergencia del método de Newton

Supongamos que $f \in C^2[a, b]$ y que $\overline{x} \in [a, b]$ es una raíz simple de f, es decir, $f(\overline{x}) = 0$ y $f'(\overline{x}) \neq 0$. Entonces existe una constante $\delta > 0$ tal que la sucesión $\{x_n\}_0^\infty$ generada por el método de Newton converge a $\overline{x}$ cualquiera que sea la aproximación inicial $x_0 \in [\overline{x} - \delta, \overline{x} + \delta]$, y además la convergencia es cuadrática, es decir, existe una contante $C > 0$ tal que

$$|x_{n+1} - \overline{x}| \leq C|x_n - \overline{x}|^2, \quad n \geq 0.$$

Existe un resultado que, partiendo de un intervalo inicial adecuado, cuando este existe, nos permite asegurar la convergencia del método de Newton y nos indica el valor incical con el que comenzar la iteración.

Teorema 1.7 (Regla de Fourier) Sea $f(x) : [a, b] \to \mathbb{R}$, continua y dos veces diferenciable con continuidad en $[a, b]$ y tal que verifica:

- $f(a) \cdot f(b) < 0$, es decir $sigf(a) \neq sigf(b)$,
- $f'(x) \neq 0, \forall x \in [a, b]$,
- $f''(x) \neq 0, \forall x \in [a, b]$.

Entonces el método de Newton converge si tomamos $x_0 = a$ o $x_0 = b$ de tal forma que $f(x_0) \cdot f''(x_0) > 0$, es decir, tomando como valor inicial x_0 el extremo del intervalo en el que la función y su segunda derivada tienen el mismo signo.

Ejemplo 1.3 Aproximar la raíz cuadrada de 3 con el método de Newton partiendo del intervalo $[1, 2]$.

1.2.4. Modificaciones del método de Newton.

El método de Newton presenta problemas cuando $\bar{x}$, la raíz de $f(x) = 0$ que se busca es múltiple. Esta situación se detecta porque la convergencia del método se hace especialmente lenta. La fórmula de Newton-Raphson puede modificarse para adpatarse a este caso:

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

donde k representa la multiplicidad de $\bar{x}$.

En la práctica, el problema es que no conocemos k, pero a esto nos puede ayudar el comportamiento de f y sus derivadas al aplicar el método.

Ejemplo 1.4 La ecuación $x - \sin x = 0$ tiene una raíz triple en $x = 0$. Aplicar el método de Newton y su modificación a este ejemplo partiendo del intervalo $[-1, 1]$.

1.2.5. Método de la secante.

En muchas aplicaciones, $f(x)$ no viene dada por una fórmula explícita, por ejemplo si $f(x)$ es el resultado de algún algoritmo numérico o de un proceso experimental. Como $f'(x)$ no estará en consecuencia disponible, el método de Newton deberá modificarse de modo que únicamente requiera valores de $f(x)$.

Cuando $f'(x)$ no está disponible, podemos reemplazarlo por una aproximación suya, por ejemplo, tomando la pendiente de la secante formada a partir de dos puntos sobre la gráfica de la función, es decir, aproximamos la derivada en un punto x_n mediante

$$f'(x_n) \approx a_n = \frac{f(x_n + h_n) - f(x_n)}{h_n}$$

Una manera de aproximar $f'(x_n)$ es utilizar los valores de f en x_n y x_{n-1}, es decir,

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

obtenemos así el llamado método de la secante,

- x_0, x_1
- $x_{n+1} = x_n - f(x_n)\frac{x_n - x_{n-1}}{f(x_n) - f(x_{n-1})}$

y que necesita de una sola evaluación de la función en cada iteración.

Pseudocódigo del algoritmo de la secante

- entrada $x_0, x_1, M, \delta, \varepsilon$
- $v_0 \leftarrow f(x_0), \quad v_1 \leftarrow f(x_1)$
- salida $0, x_0, v_0, 1, x_1, v_1$
- si $|v_1| < \varepsilon$ entonces parar
- para $k = 1, ..., M$ hacer
  - $x_2 \leftarrow x_1 - v_1 \frac{x_1 - x_0}{v_1 - v_0}$
  - $v_2 \leftarrow f(x_2)$
  - salida k, x_2, v_2
  - si $|x_2 - x_1| < \delta$ o $|v_2| < \varepsilon$ entonces parar
  - $x_0 \leftarrow x_1, v_0 \leftarrow v_1$
  - $x_1 \leftarrow x_2, v_1 \leftarrow v_2$
- fin bucle

Si hay convergencia en el método de la secante, esta es superlineal. El orden de convergencia es el número áureo.

$$\alpha = \frac{1 + \sqrt{5}}{2} \approx 1.62$$

1.3. Sistemas de ecuaciones no lineales.

Nos ocupamos ahora del problema del cálculo numérico de los ceros de funciones vectoriales de varias variables reales que tienen la forma general $F(x) = 0$, donde $F : \Re^n \to \Re^n$ viene definida por sus n componentes $f_i : \Re^n \to \Re$ para $i = 0, \ldots, n$, esto es, un sistema de n ecuaciones no lineales con n incógnitas:

$$\begin{cases} f_1(x_1, \ldots, x_n) & = 0 \\ f_2(x_1, \ldots, x_n) & = 0 \\ \ldots\ldots & \ldots \\ f_n(x_1, \ldots, x_n) & = 0 \end{cases}$$

El salto de una a varias variables conlleva la introducción de nuevos conceptos que generalicen los habituales en $\Re$. El concepto de norma como distancia, compatible con las operaciones de la estructura de espacio vectorial, generaliza el concepto de valor absoluto en $\Re$, y con él es fácil expresar el análisis en varias variables manteniendo una semejanza casi total con el caso de una variable. La dificultad es mayor, pero las herramientas son las mismas generalizando lo que hemos estudiado en una variable.

Recordemos algunos resultados del análisis en varias variables que necesitaremos.

- Sea D un conjunto cerrado de $\Re^n$ y $f : D \subset \Re^n \to \Re$, entonces,
  - f tiene límite l en x_0 ($\lim_{x\to x_0} f(x) = l$), si $\forall \epsilon > 0\ \exists \delta > 0$ tal que $|f(x) - l| < \epsilon\ \forall x \in D$ con $0 < \|x - x_0\| < \delta$.
  - f es continua en $x_0 \in D$ si $\exists \lim_{x\to x_0} f(x)$ y $\lim_{x\to x_0} f(x) = f(x_0)$.
  - f es continua en D si lo es en cada punto de D.
  - Sea $x_0 \in D$, si $\exists \delta > 0$, $K > 0$ con $|\frac{\partial f(x)}{\partial x_j}| \leq K$ para cada $j = 1, \ldots, n$ siempre que $\|x - x_0\| < \delta$ y $x \in D \Rightarrow f$ es continua en x_0.
- Sea $F : D \subset \Re^n \to \Re^n$, con $F = (f_1, \ldots, f_n)^t$ entonces,
  - $\lim_{x\to x_0} F(x) = L = (l_1, \ldots, l_n)^t \Leftrightarrow \lim_{x\to x_0} f_i(x) = l_i$ para cada $i = 1, \ldots, n$.
  - F es continua en $x_0 \in D$ si $\exists \lim_{x\to x_0} F(x)$ y $\lim_{x\to x_0} F(x) = F(x_0)$.
  - F es continua en D si lo es en cada punto de D.

1.3.1. Método de punto fijo en varias variables.

Sea $G : D \subset \Re^n \to \Re^n$, decimos que tiene un punto fijo en $p \in D$ si $G(p) = p$.

Como en el caso de una variable, cuando tenemos un sistema de ecuaciones no lineales, $F(x) = 0$, podemos escribirlo en la forma $G(x) = x$, de varias formas, por ejemplo haciendo $G(x) = x - F(x)$, y transformar el problema de calcular una raíz de F en calcular un punto fijo de G.

Pseudo-código del algoritmo de punto fijo en varias variables

- entrada $x_0,\ M,\ \varepsilon$
- $x \leftarrow x_0$
- Para $k = 1, ..., M$ hacer
  - $x_1 \leftarrow x$, $x \leftarrow G(x)$, $e \leftarrow \|x - x_1\|$

- salida $k,\ x,\ e$
- si $e < \varepsilon$ entonces parar

▪ fin bucle

Tenemos el siguiente resultado que nos da la convergencia del método de punto fijo en varias variables. Observar que la convergencia depende de las propiedades de G, por tanto, la elección de esta función a la hora de escribir el sistema que queremos resolver $F(x) = 0$, en la forma $G(x) = x$, es crucial.

Teorema 1.8 :

Sea $D = \{(x_1, \ldots, x_n)^t / a_i \leq x_i \leq b_i, i = 1, \ldots, n\}$ y $G : D \to \Re^n$ tal que sea continua y $G(x) \in D$, $\forall x \in D$. Entonces G tiene al menos un punto fijo $p \in D$. Si además, G tiene derivadas parciales primeras continuas y $\exists K < 1$ tal que

$$\left|\frac{\partial g_i(x)}{\partial x_j}\right| \leq \frac{K}{n} \quad \forall x \in D, \quad i, j = 1, \ldots, n$$

entonces la sucesión $\{x^{(m)}\}_{m=0}^{\infty}$ definida por la iteración funcional $x^{(m)} = G(x^{(m-1)})$ para $m \geq 1$, partiendo de un $x^{(0)} \in D$ arbitrario, converge a dicho punto fijo $p \in D$ y

$$\|x^{(m)} - p\|_\infty \leq \frac{K^m}{1-K}\|x^{(1)} - x^{(0)}\|_\infty$$

1.3.2. Método de Newton en varias variables.

Supongamos que $x^{(0)}$ es un valor próximo a $\overline{x}$ solución de $F(x) = 0$, es decir, $x^{(0)} = \overline{x} + h$, haciendo el desarrollo de Taylor,

$$0 = F(\overline{x}) = F(x^{(0)} - h) = F(x^{(0)}) - DF(x^{(0)})(h) + \text{Resto}$$

Despreciando el resto,

$$DF(x^{(0)})(h) \approx F(x^{(0)})$$

es decir,

$$h \approx DF(x^{(0)})^{-1} F(x^{(0)})$$

de donde,

$$x^{(1)} = x^{(0)} - h = x^{(0)} - DF(x^{(0)})^{-1} F(x^{(0)})$$

será una mejor aproximación de $\overline{x}$ que $x^{(0)}$, donde $DF(x^{(0)})$ viene dado por la matriz Jacobiana,

$$J(x) = \begin{pmatrix} \frac{\partial f_1(x)}{\partial x_1} & \frac{\partial f_1(x)}{\partial x_2} & \cdots & \frac{\partial f_1(x)}{\partial x_n} \\ \frac{\partial f_1(x)}{\partial x_1} & \frac{\partial f_1(x)}{\partial x_2} & \cdots & \frac{\partial f_1(x)}{\partial x_n} \\ \cdots & \cdots & \cdots & \cdots \\ \frac{\partial f_1(x)}{\partial x_1} & \frac{\partial f_1(x)}{\partial x_2} & \cdots & \frac{\partial f_1(x)}{\partial x_n} \end{pmatrix}$$

De aquí se obtiene el algoritmo de Newton:

- $x^{(0)}$, valor cercano a $\overline{x}$.
- Calculado $x^{(n)}$, obtenemos $x^{(n+1)}$,

$$x^{(n+1)} = x^{(n)} - J(x^{(n)})^{-1}F(x^{(n)})$$

Este algoritmo plantea la dificultad de tener que calcular la inversa de la matriz Jacobiana en cada iteración, en la práctica el método se realiza en don pasos,

- se resuelve el sistema $J(x^{(n)})y = -F(x^{(n)})$
- se calcula $x^{(n+1)} = x^{(n)} + y$,

Pseudocódigo del algoritmo de Newton

- entrada $x^{(0)}, M, \varepsilon$
- para $k = 1, ..., M$ hacer
  - calcular $F(x)$ y $J(x)$
  - resolver el sistema lineal $n \times n$ $J(x)y = -F(x)$
  - hacer $x = x + y$
  - salida k, x
  - si $\|y\| < \varepsilon$ entonces parar
  - sino hacer $k = k + 1$
- fin bucle

En el método de Newton en una variable se podía interpretar como un método de punto fijo donde tratábamos de encontrar una función $\rho(x)$ tal que la iteración funcional de $g(x) = x - \rho(x)f(x)$ diera convergencia cuadrática al punto fijo p de $g(x)$, escogiendo $\rho(x) = 1/f'(x)$. Para el caso n-dimensional, buscamos una matriz $n \times n$, $A(x) = (a_{ij}(x))$, donde cada componente es una función $a_{ij}(x) : \Re^n \to \Re$ tal que,

$$G(x) = x - A(x)^{-1}F(x)$$

de convergencia cuadrática a la solución de $F(x) = 0$, siempre que, desde luego, $A(x)$ sea no singular en el punto fijo de $G(x)$. Esta matriz va a ser precisamente la matriz Jacobiana.

Tenemos el siguiente resultado que nos da la convergencia del método de Newton en varias variables, de nuevo, el valor incial debe ser próximo a la solución buscada.

Teorema 1.9 :

Sea $\mathbf{F} = (f_1, \ldots, f_n)^t : D \in \Re^n \to \Re^n$ con D un abierto convexo y $\mathbf{F}$ diferenciable con continuidad en D. Supongamos que existe un punto $\bar{\mathbf{x}} \in D$, dos constantes $r, \beta > 0$ y otra $\gamma \geq 0$ tales que:

$$\begin{array}{l} \mathbf{F}(\bar{\mathbf{x}}) = \mathbf{0} \\ \|J(\bar{\mathbf{x}})^{-1}\| \leq \beta \\ \|J(\mathbf{x}) - J(\mathbf{y})\| \leq \gamma \|\mathbf{x} - \mathbf{y}\| \; \forall \mathbf{x}, \mathbf{y} \in B(\bar{\mathbf{x}}, r) \end{array}$$

Entonces para $\epsilon = min\{r, 1/2\gamma\beta\}$, y $\forall \mathbf{x}^{(0)} \in B(\bar{\mathbf{x}}, \epsilon)$ la sucesión generada por el método de Newton, converge a $\bar{\mathbf{x}}$ y además

$$\|\mathbf{x}^{(m+1)} - \bar{\mathbf{x}}\| \leq \beta\gamma \|\mathbf{x}^{(m)} - \bar{\mathbf{x}}\|^2$$

.

Capítulo 2

Sistemas de ecuaciones lineales

2.1. Generalidades sobre matrices y vectores

Tipos de matrices. Notación

Denotaremos $\mathcal{M}_{n,n}$ el espacio vectorial de matrices cuadradas (n filas, n columnas) con coeficientes reales. Sea $A = (a_{ij}) \in \mathcal{M}_{n,n}$, $A^t = (a_{ji})$ la matriz traspuesta, A^{-1} la matriz inversa.

- A es singular si $detA = 0$, A es regular si $detA \neq 0$.
- A es simétrica si $A^t = A$, A es ortogonal si $A^{-1} = A^t$.
- A es definido positiva si es simétrica y $x^t Ax > 0 \forall x \neq 0$.
- A es semidefinido positiva si es simétrica y $x^t Ax \geq 0 \forall x \neq 0$.
- A es diagonal dominante si $|a_{ii}| \geq \sum_{j \neq i} |a_{ij}|, i = 1, \ldots, n$
- A es estrictamente diagonal dominante si $|a_{ii}| > \sum_{j \neq i} |a_{ij}|, i = 1, \ldots, n$
- A y B son matrices semejantes si existe una matriz regular C tal que $B = C^{-1}AC$.
- A es digonalizable si es semejante a una matriz diagonal.
- A es banda p, q si $a_{ij} = 0$ para $i \geq j + p$ y $j \geq i + q$ (triangular superior:p=1,q=n; triangular inferior:p=n,q=1; Hessemberg superior:p=2,q=n; Hessemberg inferior:p=n,q=2; diagonal:p=q=1; tridiagonal:p=q=2; etc.)
- Sea L una matriz triangular, entonces $detL = \prod_{i=1}^{n} l_{ij}$.

- El producto de dos matrices triangulares superiores (resp. inferiores) es una matriz triangular superior (resp. inferior), y los elementos de la diagonal son el producto de los elementos de las diagonales.
- La inversa de una matriz triangular superior (resp. inferior) es una matriz triangular superior (resp. inferior), y los elementos de la diagonal son los inversos de los elementos de la diagonal.

Propiedades

- Sea A una matriz cuadrada $n \times n$ con coeficientes reales. Las siguientes afirmaciones son equivalentes.
  - A^{-1} existe.
  - $detA \neq 0$.
  - El sistema lineal $Ax = 0$ tiene solamente la solución $x = 0$.
  - Para cualquier vector b, el sistema lineal $Ax = b$ tiene solución única.
  - Las filas y las columnas de A son linealmente independientes.
  - El rango de la matriz A es n.
- $(A^t)^t = A$, $(A + B)^t = A^t + B^t$, $(\alpha A)^t = \alpha A^t$.
- $(A^t)^{-1} = (A^{-1})^t$, $(AB)^t = B^t A^t$.
- $detId = 1$, $detA^t = detA$, $det(\alpha A) = \alpha^n detA$.
- $det(AB) = detAdetB$, $detA^{-1} = 1/detA$ si A^{-1} existe.
- $detB = det(C^{-1}AC) = detC^{-1}detAdetC = detA$
- Una matriz cuadrada tiene inversa $\Leftrightarrow$ es regular.
- **Lema de Schur**: Toda matriz es semejante a una matriz triangular superior mediante una transformación de semejanza con una matriz ortogonal.
- **Criterio de Sylvester**: Una matriz simétrica es definido positiva $\Leftrightarrow$ todos los determinantes principales son estrictamente positivos.
- Las submatrices principales de una matriz definido positiva son definido positivas.
- Los elementos diagonales de una matriz definido positiva son estrictamente positivos.
- Si Q_1 y Q_2 son ortogonales $\Rightarrow$ $Q_1 Q_2$ es ortogonal.
- Si Q es ortogonal $\Rightarrow$ $|detQ| = 1$.
- Si Q es ortogonal $\Rightarrow$ $\|Qx\|_2 = \|x\|_2$.

Valores y vectores propios

- Dada una matriz cuadrada A, diremos que un vector $v \neq 0$ es un *vector propio* de A de *valor propio* λ cuando $Av = \lambda v$.
- Los valores propios de A son las raíces del polinomio característico $p_A(\lambda) = det(A - \lambda Id)$.
- El espectro de A, $Esp(A)$ = conjunto de valores propios de A.
- El radio espectral de A, $\rho(A) = \text{máx}_{\lambda_i \in Esp(A)} |\lambda_i|$.
- Los vectores propios de A y A^t se llaman vectores propios por la derecha y vectores propios por la izquierda de A (respect.).
- Las matrices A y A^t tienen los mismos valores propios. Los vectores propios por la derecha son ortogonales a los vectores propios por la izquierda de valores propios diferentes.
- A es regular $\Leftrightarrow$ tiene todos los valores propios diferentes de cero. Los valores propios de A^{-1} son los inversos de los valores propios de A. v es un vector propio de valor propio λ de $A \Leftrightarrow v$ es un vector propio de valor propio $1/\lambda$ de A^{-1}.
- Los valores propios de una matriz simétrica son reales.
- Una matriz simétrica es definido positiva $\Leftrightarrow$todos los valores propios son estrictamente positivos.
- Los espectros de dos matrices semejantes son iguales. Si v es un vector propio de valor propio λ de A y $B = CAC^{-1}$, entonces $C^{-1}v$ es vector propio de valor propio λ de B.
- Se llaman *valores singulares* μ de A a las raíces cuadradas positivas de los valores propios de A^tA.

Normas vectoriales

- Sea E un espacio vectorial, una norma en E es una aplicación

$$\begin{array}{rl} \| \quad \| : & E \to \Re^+ \\ & x \to \|x\| \end{array}$$

que cumple:

- $\|x\| = 0 \Leftrightarrow x = 0$,
- $\|cx\| = |c|\|x\| \forall$ escalar $c, \forall x \in E$,
- $\|x + y\| \leq \|x\| + \|y\| \forall x, y \in E$.

- Las normas vectoriales son las que están definidas sobre espacios vectoriales de la forma $E = \Re^n$ o $E = \mathbb{C}^n$.
- Normas Hölder:

$$\|x\|_p = (\sum_{i=1}^{n} |x_i|^p)^{1/p}, p \geq 1$$

- Norma suma de módulos:

$$\|x\|_1 = \sum_{i=1}^{n} |x_i|$$

- Norma euclídea:

$$\|x\|_2 = (\sum_{i=1}^{n} |x_i|^2)^{1/2}$$

- Norma infinito:

$$\|x\|_\infty = \max_{i=1 \div n} |x_i|$$

Normas matriciales

Una norma matricial es una norma en el espacio de las matrices cuadradas $\mathcal{M}_{n,n}$ que sea multiplicativa, es decir, que cumpla

$$\|AB\| \leq \|A\|\|B\| \quad \forall A, B \in \mathcal{M}_{n,n}.$$

Una norma matricial $\|\quad\|$ es *consistente* con una norma vectorial (que denotaremos igual) si y sólo si

$$\|Ax\| \leq \|A\|\|x\| \quad \forall A \in \mathcal{M}_{n,n}, \forall x \in \mathcal{R}^n.$$

Dada una norma vectorial $\|\quad\|$, siempre se puede definir una norma matricial consistente con ella, llamada *norma matricial subordinada*, mediante

$$\|A\| = \max_{x \neq 0} \frac{\|Ax\|}{\|x\|} = \max_{\|x\|=1} \|Ax\|$$

por ejemplo, las normas subordinadas a las normas Hölder,

$$\|A\|_p = \max_{\|x\|_p=1} \|Ax\|_p$$

Se verifican las siguientes propiedades,

- Norma 1:

$$\|A\|_1 = \max_{1 \leq j \leq n} \sum_{i=1}^{n} |a_{ij}|$$

- Norma euclídea:

$$\|A\|_2 = \sqrt{\rho(A^t A)} = \mu_{max}$$

- Norma infinito:

$$\|A\|_\infty = \max_{1 \leq i \leq n} \sum_{j=1}^{n} |a_{ij}|$$

- Si A es simétrica $\Rightarrow \rho(A) = \|A\|_2$

Propiedades

- $\rho(A) \leq \|A\|$ para una norma matricial cualquiera.
- $\|A\| \leq \rho(A) + \varepsilon$ para al menos una norma matricial inducida.
- Las siguientes condiciones son equivalentes:
  - $\lim_{k\to\infty} A^k = 0$
  - $\lim_{k\to\infty} A^k x = 0 \quad \forall x \in \mathcal{R}^n$
  - $\rho(A) < 1$
  - $\|A\| < 1$ para alguna norma matricial inducida.

Condicionamiento de una matriz

Sea A una matriz cuadrada regular, y $\|\quad\|$ una norma matricial, se define el condicionamiento de A asociado a la norma $\|\quad\|$, como

$$cond(A) = \|A\|\|A^{-1}\|$$

- $cond(\alpha A) = cond(A), \forall \alpha \neq 0, \forall A \in \mathcal{M}_{n,n}$ regular.
- $cond(A) \geq 1$ si se calcula con una norma inducida.
- $cond_2(A) = \frac{\mu_{max}}{\mu_{min}}$, donde μ_{max}, μ_{min} son los valores singulares de A más grande y más pequeño.
- Una matriz está bien condicionada si su número de condicionamiento es próximo a 1.
- Las matrices mejor condicionadas son las matrices ortogonales.

En este capítulo consideraremos el problema de resolución de sistemas de ecuaciones lineales, unos de los problemas numéricos que en la práctica aparece con mayor frecuencia, por sí mismo o como parte de un problema mayor.

Los métodos que utilizaremos se clasifican en dos grupos: métodos directos y métodos iterativos. En teoría, los métodos directos permiten calcular la solución exacta con un número finito de operaciones aritméticas, en la práctica debido a la finitud de las cifras con que podemos representar los números reales en un ordenador, la acumulación de errores de redondeo produce soluciones aproximadas. En los métodos iterativos la solución se define como límite de una sucesión infinita de vectores, en la práctica se puede obtener una solución aproximada fijando un número finito de iteraciones.

Un sistema de n ecuaciones lineales con n incógnitas se puede escribir,

$$\begin{cases} a_{11}x_1 + a_{12}x_2 + \ldots + a_{1n}x_n = b_1 \\ a_{21}x_1 + a_{22}x_2 + \ldots + a_{2n}x_n = b_2 \\ \ldots \\ a_{n1}x_1 + a_{n2}x_2 + \ldots + a_{nn}x_n = b_n \end{cases}$$

donde los coeficientes a_{ij} $(i,j = 1,2,\ldots,n)$ y los términos independientes b_i $(i = 1,2,\ldots,n)$ son constantes dadas. El problema es determinar las incógnitas del sistema $x_1, x_2, \ldots, x_n$ de forma que se satisfagan las n ecuaciones simultaneamente.

El sistema en forma matricial se escribe,

$$Ax = b,$$

siendo $A = (a_{ij})$ la matriz de coeficientes, $b = (b_i)$ el vector de términos de independientes, y $x = (x_i)$ el vector de incógnitas. A partir de ahora, nos centraremos en el caso de sistemas determinados, es decir, con solución única.

2.2. Métodos directos de resolución de sistemas de ecuaciones lineales

2.2.1. Matrices triangulares

Comenzaremos presentando dos algoritmos de resolución muy sencillos para sistemas con matriz de coeficientes triangular. Tengamos en cuenta que para que un sistema de este tipo sea determinado los coeficientes de la diagonal principal deben ser no nulos.

Consideremos el siguiente sistema de n ecuaciones con n incógnitas, cuya matriz de coeficientes es triangular superior invertible,

$$\left\{\begin{array}{rl} a_{11}x_1 + a_{12}x_2 + \ldots\ldots + a_{1n}x_n = & b_1 \\ a_{22}x_2 + \ldots\ldots + a_{2n}x_n = & b_2 \\ \ldots\ldots & \\ a_{n-1,n-1}x_{n-1} + a_{n-1,n}x_n = & b_{n-1} \\ a_{nn}x_n = & b_n \end{array}\right.$$

Teniendo en cuenta que $a_{ii} \neq 0$, para $i = 1,\ldots,n$, de la última ecuación podemos despejar,

$$x_n = b_n/a_{nn},$$

sustituyendo este valor en la penúltima ecuación, tenemos

$$x_{n-1} = (b_{n-1} - a_{n-1,n}x_n)/a_{n-1,n-1},$$

y así sucesivamente hasta llegar a

$$x_1 = (b_1 - a_{1,2}x_2 - \ldots - a_{1n}x_n)/a_{1,1}.$$

Este método se llama *método de sustitución inversa.*

Algoritmo 2.1 : Método de sustitución inversa.

Para resolver $Ax = b$ siendo A triangular superior invertible

$$\begin{array}{l} x_n = b_n/a_{nn}, \\ \text{para } i \text{ desde } n-1 \text{ hasta } 1 \\ \quad x_i = (b_i - \sum_{j=i+1}^{n} a_{ij}x_j)/a_{ii}, \end{array}$$

Algoritmo 2.2 : Método de sustitución directa.

Para resolver $Ax = b$ siendo A triangular inferior invertible

$$\begin{array}{l} x_1 = b_1/a_{11}, \\ \text{para } i \text{ desde } 2 \text{ hasta } n \\ \quad x_i = (b_i - \sum_{j=1}^{i-1} a_{ij}x_j)/a_{ii}, \end{array}$$

Es importante tener una idea del costo computacional de estos algoritmos para poder hacer comparaciones entre distintos métodos. Para algoritmos de álgebra lineal, una forma de estimar este coste es contar el número de operaciones algebraicas del algoritmo. En casi todos los ordenadores las sumas y restas llevan aproximadamente el mismo tiempo de ejecución, al igual que productos y divisiones, así, normalmente para contar el número de operaciones de un algoritmo se cuentan el número de sumas/restas y el número de productos/divisiones.

En los algoritmos anteriores, el número de operaciones realizadas es,

$$\sum_{i=1}^{n-1} i = \frac{(n-1)n}{2} = \frac{n^2}{2} - \frac{n}{2} \text{ sumas/restas}$$

$$\sum_{i=1}^{n} i = \frac{n(n+1)}{2} = \frac{n^2}{2} + \frac{n}{2} \text{ productos/divisiones}$$

2.2.2. Eliminación gaussiana

Entre los métodos directos de resolución de sistemas de ecuaciones lineales es el más popular. También se usa para calcular determinantes e inversas de matrices.

La idea básica consiste en transforma (reducir) el sistema inicial en uno equivalente (misma solución) cuya matriz de coeficientes sea triangular superior. El nuevo sistema se puede resolver fácilmente mediante el método de sustitución inversa. La reducción se realiza mediante transformaciones elementales sobre las ecuaciones del sistema: permutar dos ecuaciones y sustituir una ecuación por su suma con otra multiplicada por una constante.

Veamos como es posible transformar un sistema determinado $Ax = b$ de n ecuaciones con n incógnitas, realizando un número finito de transformaciones elementales, en un sistema equivalente con matriz de coeficientes triangular superior. La reducción se realiza en $n-1$ pasos.

Para simplificar la notación usaremos la matriz ampliada del sistema $\widetilde{A}$, donde cada fila representa una ecuación.

$$\widetilde{A} = \widetilde{A}_1 = \left(\begin{array}{cccc|c} a_{11} & a_{12} & \dots & a_{1n} & b_1 \\ a_{21} & a_{22} & \dots & a_{2n} & b_2 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ a_{n1} & a_{n2} & \dots & a_{nn} & b_n \end{array}\right) = \left(\begin{array}{cccc|c} a_{11}^{(1)} & a_{12}^{(1)} & \dots & a_{1n}^{(1)} & b_1^{(1)} \\ a_{21}^{(1)} & a_{22}^{(1)} & \dots & a_{2n}^{(1)} & b_2^{(1)} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ a_{n1}^{(1)} & a_{n2}^{(1)} & \dots & a_{nn}^{(1)} & b_n^{(1)} \end{array}\right)$$

Primer paso de la reducción: si $a_{11}^{(1)} \neq 0$ cada fila i, $(i = 2, 3, \dots, n)$ se sustituye por ella misma menos la primera fila multiplicada por $l_{i1} = a_{i1}^{(1)}/a_{11}^{(1)}$, obteniéndose la siguiente matriz ampliada,

$$\widetilde{A}_2 = \left(\begin{array}{cccc|c} a_{11}^{(1)} & a_{12}^{(1)} & \dots & a_{1n}^{(1)} & b_1^{(1)} \\ 0 & a_{22}^{(2)} & \dots & a_{2n}^{(2)} & b_2^{(2)} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & a_{n2}^{(2)} & \dots & a_{nn}^{(2)} & b_n^{(2)} \end{array}\right)$$

siendo

$$\begin{array}{l} a_{ij}^{(2)} = a_{ij}^{(1)} - l_{i1}a_{1j}^{(1)} \quad (i = 2, \dots, n; j = 2, \dots, n), \\ b_i^{(2)} = b_i^{(1)} - l_{i1}b_1^{(1)} \quad (i = 2, \dots, n). \end{array}$$

Segundo paso de la reducción: si $a_{22}^{(2)} \neq 0$ cada fila i, $(i = 3, 4, \dots, n)$ se sustituye por ella misma menos la segunda fila multiplicada por $l_{i2} = a_{i2}^{(2)}/a_{22}^{(2)}$, obteniéndose la siguiente matriz ampliada,

$$\widetilde{A}_3 = \left(\begin{array}{ccccc|c} a_{11}^{(1)} & a_{12}^{(1)} & a_{13}^{(1)} & \dots & a_{1n}^{(1)} & b_1^{(1)} \\ 0 & a_{22}^{(2)} & a_{23}^{(2)} & \dots & a_{2n}^{(2)} & b_2^{(2)} \\ 0 & 0 & a_{33}^{(3)} & \dots & a_{3n}^{(3)} & b_3^{(3)} \\ \vdots & \vdots & \ddots & \vdots & \vdots & \\ 0 & 0 & a_{n2}^{(3)} & \dots & a_{nn}^{(3)} & b_n^{(3)} \end{array}\right)$$

siendo

$$\begin{array}{l} a_{ij}^{(3)} = a_{ij}^{(2)} - l_{i2}a_{2j}^{(2)} \quad (i = 3, \dots, n; j = 3, \dots, n), \\ b_i^{(3)} = b_i^{(2)} - l_{i2}b_2^{(2)} \quad (i = 3, \dots, n). \end{array}$$

y así sucesivamente hasta llegar al paso $n-1$ en el que se obtiene la matriz ampliada,

$$\widetilde{A}_n = \left(\begin{array}{cccc|c} a_{11}^{(1)} & a_{12}^{(1)} & \dots & a_{1n}^{(1)} & b_1^{(1)} \\ 0 & a_{22}^{(2)} & \dots & a_{2n}^{(2)} & b_2^{(2)} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \dots & a_{nn}^{(n)} & b_n^{(n)} \end{array}\right)$$

de un sistema equivalente al primero triangular superior que se resuelve por sustitución inversa.

Los números $a_{11}^{(1)}, a_{22}^{(2)}, \ldots, a_{nn}^{(n)}$ se llaman pivotes. Los números l_{ij} se llaman multiplicadores.

Si en el paso correspondiente se tiene que $a_{kk}^{(k)} = 0$, por ser el sistema determinado, siempre podemos encontrar $a_{jk}^{(k)} \neq 0, (j \geq k)$, permutar las filas j y k y continuar el proceso, esto se denomina pivotaje. El proceso de eliminación gaussiana se puede realizar sin pivotaje, si y sólo si todos los determinantes principales de la matriz de coeficientes A son no nulos.

Ejemplo 2.1 Resolver el siguiente sistema de ecuaciones lineales mediante el método de eliminación gaussiana,

$$\left\{\begin{array}{rcrcrcrcr} x_1 & - & x_2 & + & 2x_3 & - & x_4 & = & -8 \\ 2x_1 & - & 2x_2 & + & 3x_3 & - & 3x_4 & = & -20 \\ x_1 & + & x_2 & + & x_3 & & & = & -2 \\ x_1 & - & x_2 & + & 4x_3 & + & 3x_4 & = & 4 \end{array}\right.$$

Algoritmo 2.3 Método de eliminación gaussiana

Para resolver el sistema determinado $Ax = b$ siendo A de orden n

para k desde 1 hasta n
  para i desde $k+1$ hasta n
    $l = a_{ik}/a_{kk}$
    para j desde $k+1$ hasta n
      $a_{ij} = a_{ij} - la_{kj}$,
    $b_i = b_i - lb_k$,
$x_n = b_n/a_{nn}$,
para i desde $n-1$ hasta 1
  $x_i = (b_i - \sum_{j=i+1}^{n} a_{ij}x_j)/a_{ii}$,

El número de operaciones aritméticas realizadas en el proceso de reducción es,

$$\sum_{k=1}^{n-1}(n-k)(n-k+1) = \sum_{m=1}^{n-1} m(m+1) = \frac{1}{6}(n-1)n(2n-1) + \frac{1}{2}n(n-1) \text{ sumas/restas}$$

$$\sum_{k=1}^{n-1}(n-k)(n-k+2) = \sum_{m=1}^{n-1} m(m+2) = \frac{1}{6}(n-1)n(2n-1) + n(n-1) \text{ productos/divisiones}$$

Junto con el coste operacional del proceso de sustitución inversa, tenemos un total de,

$$\frac{n^3}{3} + \frac{n^2}{2} - \frac{5n}{6} \text{ sumas/restas}$$

$$\frac{n^3}{3} + n^2 - \frac{n}{3} \text{ productos/divisiones}$$

En la siguiente tabla podemos ver como aumenta el coste operacional con el tamaño del sistema.

n	$+/-$	$*/\div$
3	17	11
10	430	375
50	44150	42875
100	343300	338250

Una variante del método de eliminación gaussiana es el **Método de Gauss-Jordan**. Consiste en reducir a 0 no sólo los elementos por debajo de la diagonal, sino también por encima, es decir, en el paso k de la reducción, se transforman en 0 todos los elementos de la columna k fuera de la diagonal, obteniendo al final un sistema equivalente con matriz de coeficientes diagonal, que se resuelva fácilmente dividiendo los términos independientes por el correspondiente elemento de la diagonal. El coste operacional de este método es superior al de eliminación gaussiana.

$$\frac{n^3}{2} - \frac{n}{2} \text{ sumas/restas}$$

$$\frac{n^3}{2} + n^2 - \frac{n}{2} \text{ productos/divisiones}$$

2.2.3. Técnicas de pivotaje

Si en algún paso del proceso de eliminación gaussiana el pivote es no nulo pero muy pequeño, aunque podemos aplicar el método, éste es muy inestable numéricamente en el sentido de que los errores de la solución, propagados a partir de los errores de los datos, pueden aumentar notablemente. Conviene en estos casos modificar el método de eliminación gaussiana con alguna técnica de pivotaje, por ejemplo:

Pivotaje parcial o maximal por columnas

En el paso k-ésimo, en lugar de tomar como pivote $a_{kk}^{(k)}$, se toma el elemento de máximo valor absoluto entre los elementos $a_{ik}^{(k)}$ para $i = k, \ldots, n$, es decir, si $|a_{pk}^{(k)}| = \text{máx}_{k \leq i \leq n} |a_{ik}^{(k)}|$, se permutan las filas p y k.

También es recomendable utilizar alguna técnica de pivotaje cuando las magnitudes entre las distintas filas (distintas ecuaciones) es muy diferente, en este caso se recomienda:

Pivotaje escalado

:

Calculamos para cada fila un factor escalar $s_i = \text{máx}_{1\leq j\leq n} |a_{ij}|$, es decir, elegimos en cada fila el elemento de máximo valor absoluto. Si $detA \neq 0$ entonces $s_i \neq 0 \quad \forall i = 1, \ldots, n$. En cada paso k, permutamos la fila k por la p si

$$\frac{|a_{pk}^{(k)}|}{s_p} = \max_{k\leq i\leq n} \frac{|a_{ik}^{(k)}|}{s_i}$$

De este modo conseguimos eliminar la diferencia de magnitud relativa entre los elementos de distintas filas. Los factores escalares se eligen una sola vez al principio del proceso y no en cada paso, pues esto elevaría en exceso el coste operacional.

2.2.4. Factorización LU

El método de factorización LU produce la factorización de la matriz del sistema A en el producto de una matriz triangular inferior con unos en la diagonal L, y otra triangular superior U. Esta factorización $A = LU$ se puede realizar de manera única siempre que todos los determinantes principales de A sean no nulos, la misma condición que nos permite realizar el proceso de eliminación gaussiana sin pivotaje. En caso de que no se cumpla esta condicón y siempre que A sea regular, será posible permutar las ecuaciones de manera que la nueva matriz PA admita una tal factorización, $PA = LU$.

Una vez realizada la factorización LU, la solución del sistema $Ax = b$, (o bien $PAx = Pb$ si ha sido necesario hacer alguna permutación), se realiza en dos pasos, resolviendo sucesivamente dos sistemas triangulares,

$$Ax = LUx = b \Rightarrow \begin{cases} 1) & Ly = b \Rightarrow y \\ 2) & Ux = y \Rightarrow x \end{cases}$$

$$PAx = LUx = Pb \Rightarrow \begin{cases} 1) & Ly = Pb \Rightarrow y \\ 2) & Ux = y \Rightarrow x \end{cases}$$

El modo de calcular los coeficientes de las matrices L y U nos lo da el proceso de eliminación gaussiana sin más que interpretarlo en forma matricial.

Permutar las filas p y k de una matriz A consiste en multiplicar dicha matriz por la matriz de permutación P correspondiente, que no es mas que la matriz identidad en la que se han permutado las filas p y k. Observar que las matrices de permutación verifican $P^{-1} = P$. Observar también que el determinate de una matriz a la que se han permutado dos filas es el mismo cambiado de signo.

Así, si en el primer paso del proceso de eliminación gaussiana tenemos que permutar las filas 1 y p, estamos multiplicando el sistema $Ax = b$ por la matriz de permutación P_1, que es la matriz identidad donde se han permutado las filas 1 y p. Después, al hacer ceros por debajo del pivote estamos multiplicando por una matriz L_1 que se obtiene a partir de los multiplicadores, a saber,

$$P_1 A_1 = L_1 A_2$$

donde,

$$L_1 = \begin{pmatrix} 1 & 0 & 0 & \dots & 0 \\ l_{21} & 1 & 0 & \dots & 0 \\ l_{31} & 0 & 1 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ l_{31} & 0 & 0 & \dots & 1 \end{pmatrix}$$

Entonces, multiplicando por P_1, tenemos,

$$A = A_1 = P_1 L_1 A_2$$

En el segundo paso,

$$P_2 A_2 = L_2 A_3$$

donde,

$$L_2 = \begin{pmatrix} 1 & 0 & 0 & \dots & 0 \\ 0 & 1 & 0 & \dots & 0 \\ 0 & l_{32} & 1 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & l_{n2} & 0 & \dots & 1 \end{pmatrix}$$

de modo que,

$$A = A_1 = P_1 L_1 A_2 = P_1 L_1 P_2 L_2 A_3$$

Al llegar al último paso,

$$A = P_1 L_1 P_2 L_2 \dots P_{n-1} L_{n-1} A_n$$

donde $A_n = U$ es la matriz triangular superior buscada. Si no hiciera falta ninguna permutación tendríamos $A = L_1 L_2 \dots L_{n-1} U = LU$ donde $L = L_1 L_2 \dots L_{n-1}$ es la matriz triangular inferior con unos en la diagonal construida con los multiplicadores l_{ij},

$$L_= \begin{pmatrix} 1 & 0 & 0 & \dots & 0 \\ l_{21} & 1 & 0 & \dots & 0 \\ l_{31} & l_{32} & 1 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ l_{n1} & l_{n2} & l_{n3} & \dots & 1 \end{pmatrix}$$

En el caso de ser necesarias las permutaciones, como el producto de matrices no es conmutativo, tenemos,

$$A = P_1 L_1 P_2 L_2 \dots P_{n-1} L_{n-1} A_n = P_1 P_2 \dots P_{n-1} L'_1 L'_2 \dots L'_{n-1} U = PLU$$

donde,

$$\begin{aligned} P &= P_1 P_2 \dots P_{n-1} \\ L &= L'_1 L'_2 \dots L'_{n-1} \\ L'_k &= P_{n-1} P_{n-2} \dots P_{k+1} L_k P_{k+1} \dots P_{n-2} P_{n-1} \end{aligned}$$

de modo que multiplicando por la matriz de permutación P tenemos, $PA = LU$.

Ejemplo 2.2 Resolver el sistema de ecuaciones lineales del ejemplo anterior mediante el método de factorización LU.

2.2.5. Matrices especiales: factorización LDL^t, Cholesky

Si la matriz A es simétrica y admite una factorización $A = LU$, podemos escribir $U = DR$ con D diagonal y R triangular superior con unos en la diagonal. Al ser A simétrica, $A = A^t$, por lo que $LDR = R^t DL^t$, y como la factorización es única, entonces $R = L^t$, de donde se puede escribir,

$$A = LDL^t$$

Si además A es definido positiva, entonces los elementos diagonales de D son positivos y podemos calcular su raíz cuadrada. Sea $D^{1/2}$ la matriz diagonal cuyos coeficientes son las raíces cuadradas de los de D. Entonces,

$$A = LDL^t = LD^{1/2}D^{1/2}L^t = \mathcal{L}\mathcal{L}^t$$

donde $\mathcal{L} = LD^{1/2}$ que es la factorización de Cholesky para matrices simétricas definido positivas.

Ejemplo 2.3 Calcular la factorización LDL^t y la de Cholesky para la siguiente matriz,

$$\begin{pmatrix} 13 & 11 & 11 \\ 11 & 13 & 11 \\ 11 & 11 & 13 \end{pmatrix}$$

2.2.6. Aplicaciones

Cálculo del determinante de una matriz.

Si en una matriz se permutan dos filas o dos columnas, el valor absoluto del determinante no varía, pero si cambia de signo. Si en una matriz a los elementos de una fila o columna se les suman los de otra multiplicados por un escalar, el determinante no varía. Estas son las transformaciones que se realizan en el proceso de eliminación gaussiana, por tanto,

$$detA = \varepsilon detU = \varepsilon \prod_{i=1}^{n} u_{ii}$$

ya que U es triangular, siendo $\varepsilon = 1$ si el número de permutaciones de filas realizadas es par, y $\varepsilon = -1$ si es impar.

El número de operaciones realizadas al calcular el determinate de este modo es del orden de $\vartheta(\frac{n^3}{3})$, mucho menor que el número de operaciones realizadas al usar la regla de Laplace para el cálculo del determinante,

$$detA = \sum_{\sigma} (-1)^{sign\sigma} a_{1,\sigma(1)} \dots a_{n,\sigma(n)}$$

que son $n! - 1$ sumas/restas, y $n!(n-1)$ productos/divisiones.

Cálculo de la inversa de una matriz.

Si A es una matriz regular, y X es su inversa, entonces su producto es la matriz identidad $AX = Id$. Separando las columnas de esta ecuación, si $x^{(k)}$ es la k-ésima columna de X, y $e^{(k)}$ la k-ésima columna de la matriz identidad, esto es, el k-ésimo vector de la base canónica, tenemos

$$Ax^{(k)} = e^{(k)}$$

Entonces el cálculo de X se reduce a la resolución de n sistemas de ecuaciones lineales de dimensión n con la misma matriz de coeficientes A y distintos términos independientes, las columnas de la matriz identidad. Entonces basta con aplicar el método de eliminación gaussiana la la siguiente matriz ampliada,

$$(A|Id) = \left(\begin{array}{ccc|ccc} a_{11} & \dots & a_{1n} & 1 & \dots & 0 \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ a_{n1} & \dots & a_{nn} & 0 & \dots & 1 \end{array}\right)$$

Ejemplo 2.4 Invertir las siguiente matrices,

$$\begin{pmatrix} 1 & 2 & -1 \\ 2 & 1 & 0 \\ -1 & 1 & 2 \end{pmatrix}$$

$$\begin{pmatrix} 1 & 2 & 3 & 4 \\ 1 & 4 & 9 & 16 \\ 1 & 8 & 27 & 64 \\ 1 & 16 & 81 & 256 \end{pmatrix}$$

2.3. Métodos iterativos de resolución de sistemas de ecuaciones lineales

Los métodos directos son eficaces para sistemas de tamaño moderado, por ejemplo $n \approx 1000$ o en el caso de matrices huecas $n \approx 5000$, 10000. Para valores significativamente mayores los métodos directos pierden eficacia, no sólo porque el número de operaciones necesario crece desmesuradamente sino también porque la acumulación de errores de redondeo puede desvirtuar el resultado.

En el caso de grandes sistemas de ecuaciones, los llamados métodos iterativos, resultan más convenientes. De forma genérica: Para resolver un sistema $Ax = b$, se transforma en otro equivalente (es decir, con la misma solución) que tenga la forma

$$x = Bx + c$$

expresión que sugiere el siguiente método iterativo

$$\left\{\begin{array}{c} x^{(0)}, \quad \text{arbitrario} \\ x^{(k+1)} = Bx^{(k)} + c \end{array}\right.$$

Diremos que el método es convergente, si

$$\lim_{k\to\infty} x^{(k)} = x$$

cualquiera que sea el valor inicial $x^{(0)}$ elegido.

Teorema 2.1 El método iterativo anterior es convergente si $\rho(B) < 1$, o de forma equivalente si $\|B\| < 1$ para al menos una norma matricial (que podemos elegir subordinada).

Demostración:

$$x = Bx + c$$
$$x^{(k+1)} = Bx^{(k)} + c$$

restando

$$x^{(k+1)} - x = B(x^{(k)} - x)$$

Llamando $e^{(0)} = x^{(0)} - x$ al error inicial y $e^{(k)} = x^{(k)} - x$ al error en la iteración k, resulta $e^{(k+1)} = Be^{(k)}$ y también

$$e^{(k)} = Be^{(k-1)} = ... = B^k e^{(0)}$$

de donde

$$\|e^{(k)}\| = \|B^k e^{(0)}\| \leq ||B^k|| \|e^{(0)}\| \leq \|B\|^k \|e^{(0)}\|$$

Si $\|B\| < 1$ entonces

$$\lim_{k\to\infty} \|e^{(k)}\| = 0$$

es decir,

$$\lim_{k\to\infty} x^{(k)} = x$$

■

Hemos obtenido,

$$\|e^{(k)}\| \leq \|B^k\| \|e^{(0)}\|$$

esto se puede interpretar como que en cada una de las k primeras iteraciones el error se ha reducido en un factor de $\|B^k\|^{1/k}$ y, en consecuencia, se puede estimar que para que el error se reduzca en un factor de 10^{-m} se deben realizar N iteraciones cumpliéndose,

$$(\|B^k\|^{1/k})^N \leq 10^{-m}, \text{ o bien } N \geq \frac{m}{-log_{10}(\|B^k\|^{1/k}))}.$$

Al número $-log_{10}(\|B^k\|^{1/k}))$ se le llama velocidad media de convergencia en k iteraciones.

Se puede demostrar que $\rho(B) = \lim_{k\to\infty} \|B^k\|^{1/k}$. Al número $-log_{10}(\rho(B))$ se le llama velocidad asintótica de convergencia.

Teorema 2.2 El método iterativo anterior se verifica,

$$\|e^{(k)}\| = \|x^{(k)} - x\| \leq \frac{\|B\|^k}{1 - \|B\|}\|x^{(1)} - x^{(0)}\|$$

Demostración: En efecto, tomando normas en la siguiente expresión,

$$x^{(k)} - x = x^{(k)} - x^{(k+1)} + x^{(k+1)} - x$$

resulta

$$\|x^{(k)} - x\| \leq \|x^{(k)} - x^{(k+1)}\| + \|x^{(k+1)} - x\| \leq \|x^{(k+1)} - x^{(k)}\| + \|B\|\|x^{(k)} - x\|$$

es decir,

$$(1 - \|B\|)\|x^{(k)} - x\| \leq \|x^{(k+1)} - x^{(k)}\| \leq \|B\|\|x^{(k)} - x^{(k-1)}\| \leq \ldots \leq \|B\|^k\|x^{(1)} - x^{(0)}\|$$

de donde se deduce la desigualdad buscada. ■

2.3.1. Método de Jacobi

Supongamos que queremos resolver el sistema

$$\begin{array}{lll} a_{11}x_1 + & \ldots & + a_{1n}x_n = b_1 \\ a_{21}x_1 + & \ldots & + a_{2n}x_n = b_2 \\ & \ldots & \\ a_{n1}x_1 + & \ldots & + a_{nn}x_n = b_n \end{array}$$

que podemos escribir

$$\begin{aligned} a_{11}x_1 &= b_1 - \sum_{j \neq 1} a_{1j}x_j \\ a_{22}x_2 &= b_2 - \sum_{j \neq 2} a_{2j}x_j \\ &\ldots \\ a_{nn}x_n &= b_n - \sum_{j \neq n} a_{nj}x_j \end{aligned}$$

El algoritmo de Jacobi se escribe: Dado $x^{(0)} \in \mathcal{R}^n$ arbitrario, una vez calculada una aproximación $x^{(k)}$, calculamos $x^{(k+1)}$ de la manera siguiente,

$$\begin{aligned} x_1^{(k+1)} &= \frac{b_1 - \sum_{j \neq 1} a_{1j}x_j^{(k)}}{a_{11}} \\ x_2^{(k+1)} &= \frac{b_2 - \sum_{j \neq 2} a_{2j}x_j^{(k)}}{a_{22}} \\ &\ldots \\ x_n^{(k+1)} &= \frac{b_n - \sum_{j \neq n} a_{nj}x_j^{(k)}}{a_{nn}} \end{aligned}$$

Este método está definido sólo si $a_{ii} \neq 0$ para $i = 1, \ldots, n$. La ecuación i-ésima

$$\boxed{x_i^{(k+1)} = \frac{1}{a_{ii}}\Big(b_i - \sum_{j \neq i} a_{ij} x_j^{(k)}\Big)}$$

se puede escribir también, restando en los dos miembros $x_i^{(k)}$ así:

$$x_i^{(k+1)} - x_i^{(k)} = \frac{1}{a_{ii}}\Big(b_i - \sum_{j=1}^{n} a_{ij} x_j^{(k)}\Big) = \frac{r_i^{(k)}}{a_{ii}}$$

donde hemos designado mediante $r^{(k)}$ al vector residuo

$$r^{(k)} = b - Ax^{(k)}$$

correspondiente al valor $x^{(k)}$.

Vamos a escribir el método anterior en forma matricial. Pondremos

$$A = D - E - F$$

donde

- D es la parte diagonal de A, $D_{ii} = a_{ii}, \quad i = 1, \ldots, n$
- $-E$ es la parte estrictamente triangular inferior

$$\begin{cases} (-E)_{ij} = a_{ij} & i > j \\ (-E)_{ij} = 0 & i \leq j \end{cases}$$

- $-F$ es la parte estrictamente triangular superior

$$\begin{cases} (-F)_{ij} = a_{ij} & i < j \\ (-F)_{ij} = 0 & i \geq j \end{cases}$$

Entonces la iteración de Jacobi se escribe

$$\boxed{x^{(k+1)} = D^{-1}(E + F)x^{(k)} + D^{-1}b}$$

o bien,

$$\boxed{x^{(k+1)} = (I - D^{-1}A)x^{(k)} + D^{-1}b}$$

es pues de la forma general $x = Bx + c$, con $B = J = I - D^{-1}A$ y $c = D^{-1}b$.

2.3.2. Método de Gauss-Seidel

Si observamos con atención la expresión general de una iteración del algoritmo de Jacobi, podemos ver que si procedemos en el orden natural, $i = 1, 2, \ldots, n$, al calcular $x_i^{(k+1)}$, los valores $x_1^{(k+1)}, x_2^{(k+1)}, \ldots, x_{i-1}^{(k+1)}$ ya los hemos obtenido. Si el método es convergente, tenemos la esperanza que estos $i-1$ valores estén más cerca de la solución que los anteriores $x_1^{(k)}, x_2^{(k)}, \ldots, x_{i-1}^{(k)}$. Por lo tanto podemos utilizarlos en lugar de estos en la expresión que sirve para calcular $x_i^{(k+1)}$, quedando

$$\boxed{x_i^{(k+1)} = \frac{1}{a_{ii}}\Big(b_i - \sum_{j=1}^{i-1} a_{ij}x_j^{(k+1)} - \sum_{j=i+1}^{n} a_{ij}x_j^{(k)}\Big)}$$

Obtenemos así el llamado método de Gauss-Seidel, que podemos escribir de la forma

$$\sum_{j=1}^{i} a_{ij}x_j^{(k+1)} = b_i - \sum_{j=i+1}^{n} a_{ij}x_j^{(k)}$$

o en forma matricial

$$(D - E)x^{(k+1)} = b + Fx^{(k)}$$

y también

$$\boxed{x^{(k+1)} = (D - E)^{-1}Fx^{(k)} + (D - E)^{-1}b}$$

que es de la forma general $x = Bx + c$, con $B = \mathcal{L}_1 = (D - E)^{-1}F$ y $c = (D - E)^{-1}b$.

En el método de Gauss-Seidel no aparece el residuo explícitamente, sino

$$\widetilde{r}_i = b_i - \sum_{j=1}^{i-1} a_{ij}x_j^{(k+1)} - \sum_{j=i}^{n} a_{ij}x_j^{(k)}$$

entonces,

$$x_i^{(k+1)} = x_i^{(k)} + \frac{\widetilde{r}_i}{a_{ii}}$$

2.3.3. Métodos de relajación

Se pueden generalizar los dos métodos anteriores de Jacobi y Gauss-Seidel, introduciendo un parámetro $\omega > 0$. Sea $x_i^{(k)}$ ya calculado y $\hat{x}_i^{(k+1)}$ obtenido a partir de $x_i^{(k)}$ por uno de los dos métodos anteriores. Se define entonces la combinación lineal

$$x_i^{(k+1)} = \omega\hat{x}_i^{(k+1)} + (1 - \omega)x_i^{(k)}$$

Si el método de partida es el de Jacobi, obtenemos para $i = 1, \ldots, n$

$$\boxed{x_i^{(k+1)} = \frac{\omega}{a_{ii}}(b_i - \sum_{j \neq i} a_{ij} x_j^{(k)}) + (1-\omega) x_i^{(k)}}$$

o bien multiplicando por a_{ii}

$$a_{ii} x_i^{(k+1)} = \omega(b_i - \sum_{j \neq i} a_{ij} x_j^{(k)}) + (1-\omega) a_{ii} x_i^{(k)}$$

y con notación matricial

$$Dx^{(k+1)} = (1-\omega)Dx^{(k)} + \omega b + \omega(E+F)x^{(k)}$$

y también

$$\boxed{x^{(k+1)} = (I - \omega D^{-1}A)x^{(k)} + \omega D^{-1}b}$$

En el caso del método de Gauss-Seidel, el correspondiente método de relajación se llama S.O.R. (Succesive Over Relaxation) y se escribe

$$\boxed{x_i^{(k+1)} = \frac{\omega}{a_{ii}}(b_i - \sum_{j=1}^{i-1} a_{ij} x_j^{(k+1)} - \sum_{j=i+1}^{n} a_{ij} x_j^{(k)}) + (1-\omega) x_i^{(k)}}$$

y con notación matricial

$$(D - \omega E)x^{(k+1)} = \omega b + ((1-\omega)D + \omega F)x^{(k)}$$

es decir

$$\boxed{x^{(k+1)} = (D - \omega E)^{-1}\omega b + (D - \omega E)^{-1}((1-\omega)D + \omega F)x^{(k)}}$$

que es de la forma general $x = Bx + c$, con $B = \mathcal{L}_\omega = (D - \omega E)^{-1}((1-\omega)D + \omega F)$ y $c = (D - \omega E)^{-1}\omega b$

2.3.4. Control de parada de las iteraciones

Designemos mediante $r^{(k)}$ al vector residuo correspondiente a la iteración k-ésima, es decir,

$$r^{(k)} = b - Ax^{(k)}$$

Un posible control de parada consiste en parar en la k-ésima iteración si

$$\frac{\|r^{(k)}\|}{\|b\|} \leq \varepsilon$$

para ε elegido convenientemente pequeño.

Esta relación implica que el error $e^{(k)} = x - x^{(k)}$ verifica

$$\frac{\|e^{(k)}\|}{\|x\|} \leq \varepsilon cond(A)$$

siendo x la solución exacta de $Ax = b$. En efecto, como

$$\|e^{(k)}\| = \|A^{-1}r^{(k)}\| \leq \|A^{-1}\|\|r^{(k)}\|$$

entonces

$$\|e^{(k)}\| \leq \varepsilon\|A^{-1}\|.\|b\| \leq \varepsilon\|A^{-1}\|.\|Ax\| \leq \varepsilon cond(A)\|x\|$$

y de ahí la afirmación realizada.

En los métodos de Gauss-Seidel y S.O.R. no aparece el residuo explícitamente, sino

$$\widetilde{r}^k = b_i - \sum_{j=1}^{i-1} a_{ij}x_j^{(k+1)} - \sum_{j=i}^{n} a_{ij}x_j^{(k)},$$

entonces el criterio de podría ser,

$$\frac{||\widetilde{r}^{(k)}||}{||b||} \leq \varepsilon$$

lo que evita cálculos suplementarios.

Otro posible criterio de parada consiste en interrumpir las iteraciones cuando,

$$\frac{\|x^{(k)} - x^{(k-1)}\|}{\|x^{(k)}\|} \leq \varepsilon$$

que es un control cómodo desde el punto de vista del cálculo. Presenta sin embargo el inconveniente que podría darse en ciertos casos en los que se verificase el control sin que $x^{(k)}$ estuviese cerca de la solución x. Por ejemplo si para algún k resulta $x^{(k)} = 0$ sin ser ésta la solución buscada.

2.3.5. Resultados de convergencia

Los métodos anteriores son de la forma general

$$x^{(k+1)} = Bx^{(k)} + c$$

La condición necesaria y suficiente de convergencia es

$$\rho(B) < 1$$

Para el método de Jacobi

$$B = J = D^{-1}(E + F) = Id - D^{-1}A$$

Para el método de Gauss-Seidel

$$B = \mathcal{L}_1 = (D - E)^{-1}F$$

Para el método S.O.R.

$$B = \mathcal{L}_\omega = (D - \omega E)^{-1}((1 - \omega)D + \omega F)$$

Estas matrices se pueden expresar en función de $L = D^{-1}E$ y de $U = D^{-1}F$ que son respectivamente dos matrices triangulares inferior y superior con diagonal nula

$$L = \begin{bmatrix} 0 & 0 & \dots & 0 \\ -\frac{a_{21}}{a_{22}} & 0 & \dots & 0 \\ & \dots & \dots & \\ -\frac{a_{n1}}{a_{nn}} & -\frac{a_{n2}}{a_{nn}} & \dots & 0 \end{bmatrix} \quad U = \begin{bmatrix} 0 & -\frac{a_{12}}{a_{11}} & \dots & -\frac{a_{1n}}{a_{11}} \\ 0 & 0 & \dots & -\frac{a_{2n}}{a_{22}} \\ & \dots & \dots & \\ 0 & 0 & \dots & 0 \end{bmatrix}$$

Teniendo en cuenta que $D^{-1}A = D^{-1}(D - E - F) = I - L - U$, podemos escribir fácilmente, para el método de Jacobi

$$J = D^{-1}(E + F) = D^{-1}E + D^{-1}F = L + U$$

para el método de Gauss-Seidel

$$\mathcal{L}_1 = (D - E)^{-1}F = (I - D^{-1}E)D^{-1}F = (I - L)^{-1}U$$

y para el método S.O.R.

$$\begin{aligned} \mathcal{L}_\omega &= (D - \omega E)^{-1}((1 - \omega)D + \omega F) = (I - \omega D^{-1}E)^{-1}((1 - \omega)I + \omega D^{-1}F) \\ &= (I - \omega L)^{-1}((1 - \omega)I + \omega U) \end{aligned}$$

Vamos a ver una condición necesaria para que el radio espectral de la matriz del método S.O.R. sea menor que la unidad.

Teorema 2.3 Para toda matriz A, el radio espectral de la matriz del método de relajación S.O.R. es superior o igual a $|\omega - 1|$ en consecuencia una condición necesaria para que el método sea convergente es $0 < \omega < 2$.

Demostración: Los valores propios de la matriz $\mathcal{L}_\omega$ del método de relajación verifican la relación

$$\Pi_{i=1}^n \lambda_i(\mathcal{L}_\omega) = det(\mathcal{L}_\omega) = \frac{det(\frac{1-\omega}{\omega}D + F)}{det(\frac{D}{\omega} - E)} = \frac{(\frac{1-\omega}{\omega})^n \Pi a_{ii}}{(\frac{1}{\omega})^n \Pi a_{ii}} = (1 - \omega)^n$$

y como por otra parte

$$\rho(\mathcal{L}_\omega) \geq |\lambda_i|$$

lo que implica

$$\rho^n(\mathcal{L}_\omega) \geq \Pi_{i=1}^n |\lambda_i| = |\omega - 1|^n$$

resulta finalmente

$$\rho(\mathcal{L}_\omega) \geq |\omega - 1|$$

■

Corolario 2.1 Para toda matriz A, una condición necesaria de convergencia del método de S.O.R. es

$$0 < \omega < 2$$

Matrices diagonal dominantes

Definición 2.1 Una matriz A cuadrada de orden n se dice que es estrictamente diagonal dominante si

$$|a_{ii}| > \sum_{\substack{j=1 \\ j\neq i}}^{n} |a_{ij}| \quad \text{para} \quad i = 1, \ldots, n$$

Teorema 2.4 Si A es una matriz de orden n estrictamente diagonal dominante entonces es no singular.

Demostración: Consideremos el sistema de ecuaciones

$$Ax = 0$$

y veamos que tiene como única solución $x = 0$.

Por reducción al absurdo, supongamos que $x = [x_1, \ldots, x_n]^t$ es una solución distinta de cero. En este caso para algún k, $0 < |x_k| = \text{máx}_{1\leq j\leq n} |x_j|$

Como $\sum_{j=1}^{n} a_{ij}x_j = 0$ para todo $i = 1, \ldots, n$, tomando $i = k$ resulta

$$a_{kk}x_k = -\sum_{\substack{j=1 \\ j\neq k}}^{n} a_{kj}x_j$$

de donde

$$|a_{kk}||x_k| \leq \sum_{\substack{j=1 \\ j\neq k}}^{n} |a_{kj}||x_j|$$

es decir

$$|a_{kk}| \leq \sum_{\substack{j=1 \\ j\neq k}}^{n} |a_{kj}| \frac{|x_j|}{|x_k|} \leq \sum_{\substack{j=1 \\ j\neq k}}^{n} |a_{kj}|$$

en contradición con la propiedad de A de ser estrictamente diagonal dominante. ■

Teorema 2.5 Sea A, matriz cuadrada de orden n estrictamente diagonal dominante. Entonces el método de Jacobi para resolver un sistema de ecuaciones lineales asociado a dicha matriz es convergente.

Demostración: La matriz de iteración para el método de Jacobi es $J = D^{-1}(E+F) = L+U$. Vamos a demostrar que $||J||_\infty < 1$. En efecto,

$$J = L + U = \begin{bmatrix} 0 & -\frac{a_{12}}{a_{11}} & \ldots & \ldots & -\frac{a_{1n}}{a_{11}} \\ -\frac{a_{21}}{a_{22}} & 0 & \ldots & \ldots & -\frac{a_{2n}}{a_{22}} \\ & \ldots & \ldots & \ldots & \\ -\frac{a_{n1}}{a_{nn}} & -\frac{a_{n2}}{a_{nn}} & \ldots & \ldots & 0 \end{bmatrix}$$

de donde

$$||J||_\infty = \max_{1\leq i\leq n} \sum_{\substack{j=1\\ j\neq i}}^{n} |\frac{a_{ij}}{a_{ii}}| = \max_{1\leq i\leq n} \frac{\sum_{j\neq i} |a_{ij}|}{|a_{ii}|} < 1$$

pues A es estrictamente diagonal dominante. ■

Teorema 2.6 Sea A una matriz estrictamente diagonal dominante, entonces el método de Gauss-Seidel para resolver un sistema de ecuaciones lineales asociado a dicha matriz es convergente.

Demostración: La matriz asociada a la iteración de Gauss-Seidel es

$$\mathcal{L}_1 = (D-E)^{-1}F = (I-L)^{-1}U$$

Para determinar el radio espectral de $\mathcal{L}_1$, calcularemos primero los valores propios, es decir, las raíces del polinomio característico

$$p(\lambda) = det(\lambda I - \mathcal{L}_1) = 0$$

Observando que $det(I-L) = 1$ resulta

$$\begin{aligned} p(\lambda) &= det(I-L)det(\lambda I - \mathcal{L}_1) \\ &= det(I-L)det(\lambda I - (I-L)^{-1}U) \\ &= det(\lambda(I-L) - U) \\ &= det(\lambda(I-L-\frac{U}{\lambda})) \\ &= \lambda^n det(I-L-\frac{U}{\lambda}) \end{aligned}$$

de donde $p(\lambda) = 0$ si $\lambda = 0$ o bien si $det(I-L-\frac{U}{\lambda}) = 0$.

Queremos demostrar que todas las raíces de $p(\lambda) = 0$ verifican $|\lambda| < 1$. Supongamos por reducción al absurdo que existe al menos una raiz λ tal que $|\lambda| \geq 1$. Entonces por una parte $det(I - l - \frac{U}{\lambda}) = 0$ y por otra parte como $A = D - E - F$ es estrictamente diagonal dominante, también lo es $I - L - U$ y lo será también $I - L - \frac{U}{\lambda}$ si $|\lambda| \geq 1$. Por lo tanto $I - L - \frac{U}{\lambda}$ es no singular en contradicción con $det(I - L - \frac{U}{\lambda}) = 0$. ■

Matrices simétricas y definidas postivas

En esta sección vamos a ver algunos resultados interesantes que relacionan la convergencia de los métodos iterativos cuando la matriz del sistema es definido positiva, caso que aparece en muchas aplicaciones prácticas, pero no nos detendremos en los detalles de las demostraciones.

Teorema 2.7 Sea A una matriz simétrica no singular descompuesta en la forma $A = M - N$ donde M es no singular. Sea $B = M^{-1}N = Id - M^{-1}A$ la matriz de iteración. Supongamos que $M^t + N$ (que es simétrica) es definido positiva. Entonces si A es definido positiva $\Rightarrow \rho(B) < 1$.

Teorema 2.8 Si A es simétrica y definido positiva $\Rightarrow$ el método SOR es convergente si $0 < w < 2$. En particular, Si A es simétrica y definido positiva $\Rightarrow$ el método de Gauss-Seidel es convergente ($w = 1$).

Teorema 2.9 Si A es simétrica, definido positiva y $2D - A$ es definido positiva $\Rightarrow$ el método de Jacobi es convergente.

Comparación de los métodos de Jacobi y Gauss-Seidel. Búsqueda del parámetro de relajación óptimo en el método S.O.R.

Teorema 2.10 Si A es una matriz tridiagonal $\Rightarrow \rho(\mathcal{L}_1) = (\rho(J))^2$. Entonces los dos métodos Jacobi y Gauss-Seidel convergen o divergen simultáneamente, y si convergen, Gauss-Seidel lo hace más rápidamente.

Teorema 2.11 Si A es definido positiva y tridiagonal, y $0 < w < 2 \Rightarrow$ los tres métodos Jacobi, Gauss-Seidel y SOR convergen y la elección óptima del parámetro es

$$w_{op} = \frac{2}{1 + \sqrt{1 - \rho(J)^2}}$$

siendo $\rho(\mathcal{L}_{w_{op}}) = w - 1$.

Capítulo 3

Interpolación

En este tema y el siguiente intentaremos dar respuesta a una situación bastante habitual en el ámbito científico: investigamos un fenómeno físico/químico que se está desarrollando ante nuestros ojos, podemos tomar muestras experimentales y a partir de estas mediciones obtener más información. Para ello podemos intentar recrear/reconstruir el fenómeno en su totalidad (en un *dominio continuo* del espacio/tiempo o cualquier otra magnitud) con una función que represente *lo mejor posible* esos datos.

Las técnicas que utilizan funciones continuas y que vamos a estudiar en este y el próximo capítulo son de dos tipos:

- Interpolación: cálculo de funciones que pasan (*interpolan* es el término matemático) exactamente por los puntos señalados por los datos.
- Aproximación: cálculo de funciones que aproximan los datos en un cierto sentido (para una determinada forma de medir el error).

En este capítulo trataremos el problema de la interpolación, que además tiene mucha utilidad al tratar la derivación y la integración numérica.

Problema de interpolación: sean (x_i, y_i) para $i = 0, \ldots, n$, pares de valores reales (puntos del plano) tales que $x_i \neq x_j$ para $i \neq j$, buscamos una función $p(x)$ de un determinado tipo, tal que $p(x_i) = y_i$ para $i = 0, \ldots, n$.

Los datos a interpolar pueden proceder de mediciones experimentales como hemos mencionado antes: conocida experimentalmente la respuesta y_i obtenida bajo condiciones x_i, nos interesa encontrar el resultado y que obtendríamos al tomar condiciones x no experimentales. Pero también podemos pensar que los puntos dados forman parte de la gráfica de una función f que queremos conocer al menos aproximadamente y de la que únicamente sabemos su valor en ciertos puntos x_i. A partir de ahora, para una mayor generalidad, hablaremos del problema de interpolación de funciones.

Al plantearse un problema de interpolación, uno debe contestar a tres preguntas:

- ¿De qué tipo debe ser la función $p(x)$ buscada? Polinomial, trigonométrica, racional, exponencial, etc. El comportamiento de los datos a interpolar nos puede orientar sobre el tipo de función interpoladora a elegir: si f tiene un comportamiento periódico, elegiremos funciones trigonométricas; si sospechamos que f puede tener asíntotas, convendrá que p sea racional; si f responde a un comportamiento polinómico, buscaremos p entre las funciones polinómicas. En este capítulo sólo trataremos este último caso, el de la interpolación polinómica.
- Una vez elegido el conjunto de funciones en el que debemos buscar p, ¿existe la función buscada?, y si existe, ¿es única?.
- ¿Es la función p una buena aproximación de la función f fuera de los puntos de interpolación?.

3.1. Interpolación polinómica.

3.1.1. Planteamiento del problema

Dados $n+1$ *puntos de interpolación* (x_i, y_i) para $i = 0, \ldots, n$, con $x_i \neq x_j$ para $i \neq j$, llamamos *interpolación polinomial* a la determinación de un polinomio $p(x)$ de grado $\leq N$ tal que

$$p(x_i) = y_i \quad i = 0, \ldots, n$$

Si y_i es el valor de una función f en x_i para $i = 0, \ldots, n$, hablaremos de la *interpolación polinomial de la función* f en las *abscisas de interpolación* o *nodos* x_i, $i = 0, \ldots, n$.

3.1.2. Tipo de función interpoladora

La función p buscada formará parte del conjunto de polinomios de grado $\leq N$, para un cierto N que determinaremos más adelante, es decir, será de la forma,

$$p(x) = a_N x^N + a_{N-1} x^{N-1} + \ldots + a_1 x + a_0$$

y para determinarla habrá que encontrar los $N+1$ coeficientes $a_0, a_1 \ldots, a_N$.

3.1.3. Existencia y unicidad del polinomio interpolador

Teorema 3.1 Dados $x_0, x_1, \ldots, x_n$ $n+1$ valores reales distintos, para cada conjunto de $n+1$ valores arbitrarios $y_0, y_1, y_2, \ldots, y_n$ existe un único polinomio $p_n(x)$ de grado a lo más n tal que $p(x_i) = y_i$ para $i = 0, 1, \ldots, n$.

Demostración:

Demostremos en primer lugar la *unicidad*: supongamos que hubiera dos polinomios $p_n(x)$ y $q_n(x)$ verificando las condiciones del teorema. Por tanto, $p_n(x) - q_n(x)$ es un polinomio de grado a lo más n verificando $(p_n - q_n)(x_i) = 0$ para $i = 0, 1, \ldots, n$, es decir, tiene $n+1$ raíces pero es de grado n, por lo tanto, $p_n - q_n \equiv 0$, es decir, $p_n \equiv q_n$.

La *existencia* la demostraremos por inducción sobre n: para $n = 0$, si $p_0(x_0) = y_0$, se trata de la función constante $p_0(x) = y_0$. Supongamos que el teorema es cierto para $n \leq k-1$, demostrémoslo para $n = k$. Por hipótesis de inducción, existe un polinomio p_{k-1} de grado a lo más $k-1$ tal que $p_{k-1}(x_i) = y_i$ para $i = 0, 1, \ldots, k-1$. Tratemos de construir p_k de la siguiente forma,

$$p_k(x) = p_{k-1}(x) + c(x - x_0)(x - x_1)\ldots(x - x_{k-1})$$

que es un polinomio de grado a lo más k verificando $p_k(x_i) = y_i$ para $i = 0, 1, \ldots, k-1$. Para determinar p_k basta calcular el valor de c despejando de

$$p_{k-1}(x_k) + c(x_k - x_0)(x_k - x_1)\ldots(x_k - x_{k-1}) = y_k$$

posible puesto que todos los nodos x_i son distintos. ■

3.1.4. Métodos de cálculo del polinomio interpolador.

Podemos dar otra demostración del Teorema 4.1. que nos permite calcular el polinomio interpolador $p(x) = a_0 + a_1 x + \ldots + a_n x^n$ simplemente imponiendo las $n+1$ condiciones que debe cumplir,

$$\left.\begin{array}{l} a_0 + a_1 x_0 + \ldots + a_n x_0^n = y_0 \\ a_0 + a_1 x_1 + \ldots + a_n x_1^n = y_1 \\ \ldots \\ a_0 + a_1 x_n + \ldots + a_n x_n^n = y_n \end{array}\right\}$$

que es un sistema lineal de $n+1$ ecuaciones con $n+1$ incógnitas que son los coeficientes $a_0, a_1, \ldots, a_n$ del polinomio p_n. El determinante de la matriz de coeficientes es el *determinante de Vandermonde* que tiene la forma,

$$\left|\begin{array}{cccc} 1 & x_0 & \ldots & x_0^n \\ 1 & x_1 & \ldots & x_1^n \\ . & . & \ldots & . \\ 1 & x_n & \ldots & x_n^n \end{array}\right| = \prod_{k>i}(x_k - x_i)$$

y es no nulo ya que $x_k \neq x_i$ si $k \neq i$. Por tanto el sistema es compatible determinado y tiene solución única.

Este proceso para calcular p_n es excesivamente laborioso cuando n no es pequeño. Hay que entender que el polinomio interpolador es único pero se puede expresar de muy diversas formas y llegar hasta él a través de diferentes algoritmos. Estudiaremos dos métodos para calcular el polinomio interpolador.

Método de Lagrange

Se toma como expresión del polinomio interpolador la fórmula de interpolación de Lagrange,

$$p_n(x) = \sum_{i=0}^{n} y_i l_i(x), \quad l_i(x) = \frac{\prod_{j\neq i}(x - x_j)}{\prod_{j\neq i}(x_i - x_j)}, \quad i = 0, 1, \ldots, n$$

donde $l_i(x)$ son los llamados polinomios de Lagrange, que son de grado n y verifican $l_i(x_j) = \delta_{ij}$ para $i, j = 0, 1, \ldots, n$, por tanto $p_n(x_i) = y_i$ para $i = 0, 1, \ldots, n$ como se deseaba.

Ejemplo 3.1 Encontrar el polinomio interpolador de la siguiente tabla de datos, mediante el método de Lagrange.

x	1	2	4	5
y	0	2	12	21

Ejemplo 3.2 Encontrar el polinomio interpolador de la siguiente tabla de datos, mediante el método de Lagrange. Observar que es la ecuación de la recta que pasa por los puntos del plano (x_0, y_0) y (x_1, y_1).

x	x_0	x_1
y	y_0	y_1

Método de diferencias divididas de Newton

Expresamos el polinomio interpolador de la siguiente forma,

$$p_n(x) = c_0 + c_1(x - x_0) + c_2(x - x_0)(x - x_1) + \ldots + c_n(x - x_0)(x - x_1)\ldots(x - x_{n-1})$$

El método de las diferencias divididas de Newton permite calcular los coeficientes c_j para $j = 0, 1, \ldots, n$, mediante la construcción de las llamadas diferencias divididas:

$$\begin{aligned} f[x_i] &= y_i, \quad (i = 0, \ldots, n) \\ f[x_i, x_{i+1}, \ldots, x_{i+j}, x_{i+j+1}] &= \frac{f[x_{i+1}, \ldots, x_{i+j+1}] - f[x_i, \ldots, x_{i+j}]}{x_{i+j+1} - x_i}, \\ & \quad (i = 0, \ldots, n-j), \quad (j = 0, \ldots, n-1) \end{aligned}$$

de forma que $c_j = f[x_0, x_1, \ldots, x_j], \quad (j = 0, \ldots, n)$, es decir, el polinomio interpolador viene dado por la siguiente *fórmula de interpolación de Newton*:

$$\begin{aligned} p_n(x) = \quad & f[x_0] + f[x_0, x_1](x - x_0) + f[x_0, x_1, x_2](x - x_0)(x - x_1) + \\ & + \ldots + f[x_0, \ldots, x_n](x - x_0)(x - x_1)\ldots(x - x_{n-1}) \end{aligned}$$

Ejemplo 3.3 Veamos el esquema de construcción de las diferencias divididas de Newton para $n = 2$,

$$\begin{array}{c|c} x_0 & f[x_0] = y_0 \\ & \searrow \\ & \qquad f[x_0, x_1] = \frac{f[x_1]-f[x_0]}{x_1-x_0} \\ & \nearrow \qquad\qquad\qquad \searrow \\ x_1 & f[x_1] = y_1 \qquad\qquad\qquad f[x_0, x_1, x_2] = \frac{f[x_1,x_2]-f[x_0,x_1]}{x_2-x_0} \\ & \searrow \qquad\qquad\qquad \nearrow \\ & \qquad f[x_1, x_2] = \frac{f[x_2]-f[x_1]}{x_2-x_1} \\ & \nearrow \\ x_2 & f[x_2] = y_2 \end{array}$$

El método de diferencias divididas de Newton tiene la ventaja de que cuando se añaden puntos de interpolación puede aprovecharse todo el trabajo hecho, basta con continuar el esquema de construcción de diferencias divididas y calcular las nuevos coeficientes $c_{n+1}, c_{n+2}, \ldots$, aprovechando así todos los cálculos previos.

Ejemplo 3.4 Con la siguiente tabla de datos,

x	1	2	4	5
$f(x)$	0	2	12	21

mediante el método de diferencias divididas de Newton, aproximar el valor de $f(3)$, usando

- el polinomio interpolador de grado 2 calculado usando los tres primeros nodos de la tabla,
- el polinomio interpolador de grado 2 calculado usando los tres últimos nodos de la tabla,
- el polinomio interpolador de grado 3 calculado usando todos los datos de la tabla.

3.1.5. Error de interpolación

Nos interesa tener un criterio para medir la proximidad del polinomio p_n a la función f fuera de los puntos de interpolación x_k.

Teorema 3.2 Sea $f \in C^{n+1}(a, b)$ y sea p_n un polinomio de grado a lo más n que interpola a f en $n + 1$ puntos distintos $x_0, x_1, \ldots, x_n$ del intervalo (a, b). Entonces para cada $x \in (a, b)$ existe un $\xi_x \in (a, b)$ tal que

$$f(x) - p_n(x) = \frac{1}{(n+1)!} f^{n+1)}(\xi_x) \prod_{i=0}^{n} (x - x_i)$$

Si analizamos el error podemos observar tres términos diferentes:

- $\frac{1}{(n+1)!} \xrightarrow[n\to\infty]{} 0$,
- $f^{n+1)}(\xi_x)$ que depende de si la derivada $n+1$-ésima de la función a interpolar está acotada,
- $\prod_{i=0}^{n}(x-x_i)$ que depende de la colocación de los nodos de interpolación.

Una pregunta natural en este contexto es la siguiente: Supongamos que dado un intervalo (a,b) lo vamos subdividiendo en más puntos, concretamente, $x_j = a+jh$ para $j = 0,1,2,...,n$, donde $h = (b-a)/n$ y supongamos que construimos con estos puntos el polinomio de interpolación $p_n(x)$ para una función dada f, esto es, que $p_n(x_i) = f(x_i)$, para estos n puntos. La pregunta es, ¿tenderá a 0 el error a medida que crece en número de nodos de interpolación, es decir, el grado del polinomio interpolador?. La respuesta es NO.

Ejemplo 3.5 Comparar lo que sucede con el error de interpolación al aumentar el número de nodos de interpolación para las siguientes funciones:

- $Sen(\pi x)$ en el intervalo $(0, 1.5)$,
- $\frac{1}{1+25x^2}$ en el intervalo $(-1,1)$.

Lo que ocurre al aproximar la función $\frac{1}{1+25x^2}$ en el intervalo $(-1,1)$ con polinomios de grado alto es lo que se conoce como el efecto Runge. La aproximación es mala en los extremos del intervalo, así que una idea para mejorar dicha aproximación es la de olvidarnos de tomar nodos igualmente espaciados y tomar nodos que se concentren más cerca de los extremos. De este modo al obligar al polinomio $p_n(x)$ a pasar por estos puntos quizás se mejore la aproximación. Por supuesto que tiene que haber un equilibrio en la disposición de los nodos x_i , pues si ponemos pocos puntos en la región central del intervalo quizás perderíamos allí. Estas ideas son las que llevan a una teoría de aproximación muy bonita, donde resulta que los nodos a usar son los ceros de los llamados **polinomios de Chebyshev** $T_n(x)$, dados por,

$$x_k = \cos\frac{(2k+1)\pi}{2(n+1)}, \quad k = 0,1,\ldots,n$$

Otra posibilidad para reducir el error de interpolación debido al uso de polinomios interpoladores de grado alto es el uso de la interpolación polinómica a trozos o **splines**. Consiste en trazar una serie de puntos que uniremos por pedazos de curvas cúbicas. Esto es, tomamos un polinomio de grado 3 distinto que une cada par de puntos consecutivos a interpolar. Los coeficientes de cada polinomio se tienen que tomar adecuadamente para que hasta las segundas derivadas coincidan en los puntos de enganche. El resultado es una curva suave agradable a la vista.

3.2. Interpolación de Hermite.

El término *interpolación de Hermite* hace referencia a la interpolación de una función y de algunas de sus derivadas en un conjunto de nodos.

3.2.1. Ejemplo sencillo

Sean x_0, x_1 dos puntos donde conocemos el valor de una función f y también de su primera derivada f'. Buscamos el polinomio p de menor grado que verifique,

$$\begin{array}{ll} p(x_0) = f(x_0), & p(x_1) = f(x_1), \\ p'(x_0) = f'(x_0), & p'(x_1) = f'(x_1). \end{array}$$

En vista de que hay cuatro condiciones, parece lógico buscar p en el espacio de polinomios de grado ≤ 3, escribámoslo de la siguiente forma,

$$p(x) = a + b(x - x_0) + c(x - x_0)^2 + d(x - x_0)^2(x - x_1),$$

cuya derivada se escribe,

$$p'(x) = b + 2c(x - x_0) + 2d(x - x_0)(x - x_1) + d(x - x_0)^2.$$

Imponiendo las condiciones y denotando $h = x_1 - x_0$, obtenemos,

$$\begin{array}{ll} f(x_0) = a & f(x_1) = a + bh + ch^2, \\ f'(x_0) = b & f'(x_1) = b + 2ch + dh^2. \end{array}$$

Por tanto, despejando,

$$\begin{array}{ll} a = f(x_0) & c = (f(x_1) - a - bh)/h^2, \\ b = f'(x_0) & d = (f'(x_1) - b - 2ch)/h^2. \end{array}$$

3.2.2. Problema de Hermite generalizado

La interpolación de Hermite puede generalizarse al caso en que conocemos la función f en una serie de nodos x_i para $i = 0, 1, \ldots, n$ y sus respectivas derivadas hasta un cierto orden que puede ser distinto en cada nodo.

Teorema 3.3 Dados $x_0, x_1, \ldots, x_n$ $n + 1$ nodos distintos dos a dos, y los valores de la función f y derivadas sucesivas en esos nodos,

$$f^{j)}(x_i), \quad j = 0, 1, \ldots, k_i - 1, \quad i = 0, 1, \ldots, n,$$

entonces existe un único polinomio p_N de grado a lo más N con $N + 1 = k_0 + k_1 + \ldots + k_n$ verificando las condiciones de interpolación

$$p_N^{j)}(x_i) = f^{j)}(x_i), \quad j = 0, 1, \ldots, k_i - 1, \quad i = 0, 1, \ldots, n.$$

Demostración:

Buscamos un polinomio de grado a lo más N, que tiene $N+1$ coeficientes, e imponemos $N+1$ condiciones. Por tanto, tenemos que resolver un sistema lineal de $N+1$ ecuaciones con $N+1$ incógnitas y deseamos asegurarnos de que la matriz de coeficientes es no singular para que exista una solución única. Para demostrar que una matriz cuadrada es no singular basta con demostrar que el correspondiente sistema homogéneo tiene como única solución la idénticamente nula. En nuestro caso esto se correspondería con encontrar un polinomio q de grado a lo más N verificando,

$$q^{j)}(x_i) = 0, \quad j = 0, 1, \ldots, k_i - 1, \quad i = 0, 1, \ldots, n.$$

es decir, buscamos un polinomio q de grado a lo más N que tiene un cero con multiplicidad k_i en x_i para $i = 0, 1, \ldots, n$, y por tanto debe ser múltiplo de $\prod_{i=0}^{n}(x - x_i)^{k_i}$ que es de grado $N+1$, imposible a no ser que $q \equiv 0$ como deseábamos.

■

En lo que se refiere al error en este caso, puede decirse que si $f \in C^{N+1}(a, b)$ y $x_i, \in (a, b)$ para $i = 0, 1, \ldots, n$, entonces para cada $x \in (a, b)$ existe un $\xi_x \in (a, b)$ tal que

$$f(x) - p_N(x) = \frac{1}{(N+1)!} f^{N+1)}(\xi_x)(x - x_0)^{k_0}(x - x_1)^{k_1} \ldots (x - x_n)^{k_n}$$

3.2.3. Caso particular: el polinomio de Taylor

El caso de interpolación de Hermite en un solo nodo se trata del conocido polinomio interpolador de Taylor: sea $f \in C^{n+1}(a, b)$, para cada $x_0 \in (a, b)$, existe un único polinomio p_n de grado a lo más n tal que $p_n^{j)}(x_0) = f^{j)}(x_0)$ para $j = 0, 1, \ldots, n$, que es el polinomio de Taylor,

$$p_n(x) = f(x_0) + f'(x_0)(x - x_0) + \frac{f''(x_0)}{2!}(x - x_0)^2 + \ldots + \frac{f^{n)}(x_0)}{n!}(x - x_0)^n,$$

junto con la fórmula del error de la interpolación de Taylor,

$$f(x) - p_n(x) = \frac{1}{(n+1)!} f^{n+1)}(\xi_x)(x - x_0)^{n+1},$$

para cierto $\xi_x \in (a, b)$.

3.2.4. Método de las diferencias divididas de Newton generalizado

Para calcular el polinomio interpolador de Hermite se usa una generalización del método de diferencias divididas de Newton en la que el esquema triangular se construye de la siguiente manera: en la primera columna se coloca cada nodo repetido tantas veces como condiciones haya sobre él; en la segunda columna

los respectivos valores de la función a interpolar en los nodos correspondientes, es decir, $f[x_i] = f(x_i)$ tantas veces como condiciones sobre el nodo i tengamos, para $i = 0, 1, \ldots, n$; en la tercera columna cuando aparezcan dos nodos iguales, tendremos en cuanta que,

$$f'(x_i) = \lim_{x \to x_i} \frac{f(x) - f(x_i)}{x - x_i} = \lim_{x \to x_i} f[x_i, x] = f[x_i, x_i],$$

y en general,

$$f[x_i, \underbrace{x_i, \ldots, x_i}_{j}] = \frac{f^{j)}(x_i)}{j!},$$

de modo que el polinomio interpolador sería,

$$\begin{aligned} p_N(x) = \quad & f[x_0] + f[x_0, x_0](x - x_0) + \ldots + f[\underbrace{x_0, x_0, \ldots, x_0}_{k_0}](x - x_0)^{k_0 - 1} + \ldots \\ & + f[\underbrace{x_0, x_0, \ldots, x_0}_{k_0}, x_1](x - x_0)^{k_0} + \ldots \\ & + f[\underbrace{x_0, \ldots, x_0}_{k_0}, \underbrace{x_1, \ldots, x_1}_{k_1}](x - x_0)^{k_0}(x - x_1)^{k_1 - 1} + \ldots \\ & + \ldots \\ & f[\underbrace{x_0, \ldots, x_0}_{k_0}, \ldots, \underbrace{x_n, \ldots, x_n}_{k_n}](x - x_0)^{k_0} \ldots (x - x_n)^{k_n - 1} \end{aligned}$$

3.2.5. Ejemplo sencillo

El triángulo de diferencias divididas de Newton que deberíamos construir para el ejemplo sencillo propuesto antes, en el que conocemos el valor de la función y su primera derivada en dos nodos, sería,

$$\begin{array}{c|ccccccc} x_0 & f[x_0] = f(x_0) & & & & & & \\ & & \searrow & & & & & \\ & & & f[x_0, x_0] = f'(x_0) & & & & \\ & & \nearrow & & \searrow & & & \\ x_0 & f[x_0] = f(x_0) & & & & f[x_0, x_0, x_1] & & \\ & & \searrow & & \nearrow & & \searrow & \\ & & & f[x_0, x_1] & & & & f[x_0, x_0, x_1, x_1] \\ & & \nearrow & & \searrow & & \nearrow & \\ x_1 & f[x_1] = f(x_1) & & & & f[x_0, x_1, x_1] & & \\ & & \searrow & & \nearrow & & & \\ & & & f[x_1, x_1] = f'(x_1) & & & & \\ & & \nearrow & & & & & \\ x_1 & f[x_1] = f(x_1) & & & & & & \end{array}$$

y el correspondiente polinomio interpolador,

$$\begin{aligned} p(x) = \quad & f[x_0] + f[x_0, x_0](x - x_0) + f[x_0, x_0, x_1](x - x_0)^2 + \\ & + f[x_0, x_0, x_1, x_1](x - x_0)^2(x - x_1). \end{aligned}$$

Ejemplo 3.6 Siendo $f(x) = x^{12}$, hallar el polinomio $p_{11}(x)$ verificando,

$$\begin{array}{ll} p_{11}^{i)}(-1) = f^{i)}(-1), & i = 0, 1, 2, 3, \\ p_{11}^{i)}(0) = f^{i)}(0), & i = 0, 1, 2, \\ p_{11}^{i)}(1) = f^{i)}(1), & i = 0, 1, 2, 3, 4. \end{array}$$

Capítulo 4

Aproximación numérica.

4.1. Introducción.

En el capítulo anterior hemos hablado de aproximación de funciones mediante interpolación que tiene muchas ventajas: el polinomio interpolador es fácil de calcular y se dispone de una fórmula explícita para el error de interpolación; la interpolación es muy útil para generar fórmulas de derivación e integración numérica; la interpolación es especialmente apropiada para el cálculo de funciones dadas por tablas, es decir, para funciones bien conocidas sobre conjuntos discretos de abscisas donde el error de redondeo de los valores es menor que el error propio de interpolación.

Sin embargo, la interpolación presenta ciertos problemas en otros casos, por ejemplo: si tenemos un conjunto discreto de valores (x_k, y_k) $(k = 0, 1, \ldots, m)$ que tienen errores de redondeo apreciables, no es conveniente utilizar el polinomio interpolador que interpole exactamente esos datos ya que su carácter oscilante puede provocar que el error fuera de los puntos de interpolación sea muy grande; otra situación en la que tampoco es conveniente la interpolación es cuando conocemos una función f en todo un intervalo I, generar una tabla de valores y buscar el polinomio interpolador no es la manera más eficiente de aproximar dicha función.

Si nos encontramos con el *caso discreto*, es decir, un conjunto discreto de valores (x_k, y_k) $(k = 0, 1, \ldots, m)$, podemos pensar en buscar un polinomio p_n de grado $n \leq m$ tal que los errores, $e_k = y_k - p_n(x_k)$ $(k = 0, 1, \ldots, m)$ sean lo más pequeños posibles en un sentido que determinaremos más adelante. Este es el proceso de aproximación polinomial. Podemos aproximar estos datos con otro tipo de función, $f_n(x) = a_0\varphi_0(x) + \ldots + a_n\varphi_n(x)$, donde debemos encontrar los parámetros $a_0, \ldots, a_n$ de forma que los errores $e_k = y_k - f_n(x_k)$ $(k = 0, 1, \ldots, m)$ sean lo más pequeños posibles en un cierto sentido. Este es el problema de *aproximación discreta*.

En el *caso continuo*, cuando conocemos la función a aproximar f en todo un intervalo I, buscamos una función f_n de forma que la función de error de aproximación $e_n(x) = f(x) - f_n(x)$ sea lo más pequeña

posible sobre el intervalo I en algún sentido que se determina a priori. La función aproximadora puede ser polinómica o de forma general, $f_n(x) = a_0\varphi_0(x) + \ldots + a_n\varphi_n(x)$, donde $\varphi_0, \ldots, \varphi_n$ son funciones dadas, fácilmente calculables, y el problema se reduce a calcular los parámetros $a_0, \ldots, a_n$.

Por tanto, el *problema general de aproximación es*: dado un conjunto I de abscisas de aproximación, y unas funciones básicas φ_j $(j = 0, 1, \ldots, n)$ definidas sobre I, para cada función f definida sobre I, buscamos los coeficientes $a_0, \ldots, a_n$, de forma que $f_n(x) = a_0\varphi_0(x) + \ldots + a_n\varphi_n(x)$ haga que la magnitud del error de aproximación $e_n(x) = f(x) - f_n(x)$ sea lo más pequeña posible.

Por tanto, para determinar totalmente un problema de aproximación es necesario especificar: el conjunto I de abscisas de aproximación, las funciones básicas y la forma de medir la magnitud del error.

4.1.1. Conjunto de abscisas de aproximación

Si I es finito $(I = x_0, \ldots, x_m)$ hablaremos de *aproximación discreta*; si I es un intervalo de extremos a y b $(a < b)$, hablaremos de *aproximación continua*. Dar una función f sobre un conjunto finito $I = x_0, \ldots, x_m$ equivale a dar $y_k = f(x_k)$ $(k = 0, \ldots, m)$.

4.1.2. Funciones básicas

Las funciones $\varphi_0, \ldots, \varphi_n$, definidas sobre I, pueden escogerse de diversas formas dependiendo del comportamiento de la función f a aproximar. Si f es periódica, elegiremos las funciones básicas entre las funciones trigonométricas, por ejemplo: $\varphi_0(x) = 1, \varphi_1(x) = senx, \varphi_2(x) = cosx, \ldots, \varphi_{2s-1}(x) = sen2sx, \varphi_{2s}(x) = cos2sx$, donde $n = 2s$, que llamaremos *aproximación trigonométrica*; Si f responde a un comportamiento polinómico, elegiremos cada $\varphi_j(x) = p_j(x)$ entre los polinomios de grado j $(j = 0, \ldots, n$ (por ejemplo, $\varphi_j(x) = x^j$ aunque esta no será siempre la elección más adecuada) y hablaremos de *aproximación polinomial*.

Observar que nos estamos limitando al caso de *aproximación lineal*, es decir, buscamos la función aproximadora $f_n(x)$ en el espacio vectorial generado por las funciones básicas,

$$\mathcal{E}_n = \langle \varphi_0, \ldots, \varphi_n \rangle,$$

esto es,

$$f_n(x) = \sum_{j=0}^{n} a_j\varphi_j(x), \quad x \in I.$$

4.1.3. Medida de la magnitud del error: normas funcionales

La magnitud del error de aproximación se puede medir de diferentes maneras según sea el caso discreto o el caso continuo, y según la norma que utilicemos.

Caso discreto

El error de aproximación es un vector de $m+1$ valores,

$$e_k = f(x_k) - f_n(x_k), \quad (k = 0, \ldots, m)$$

por tanto, para medirlo utilizaremos una norma vectorial. Las dos normas más usadas son:

- la *norma euclídea*
$$\|e\|_2 = \Big(\sum_{k=0}^{m} |e_k|^2\Big)^{\frac{1}{2}}$$
- la *norma del máximo*
$$\|e\|_\infty = \max_{k=0\div m} |e_k|$$

Cuando quiere darse una importancia diferente a los distintos términos del error se usan normas ponderadas introduciendo coeficientes positivos llamados pesos $w = \{w_k\}_{k=0\div m}$,

$$\|e\|_{2,w} = \Big(\sum_{k=0}^{m} w_k|e_k|^2\Big)^{\frac{1}{2}}, \quad \|e\|_{\infty,w} = \max_{k=0\div n} w_k|e_k|.$$

Caso continuo

El error de aproximación es una función definida en el intervalo $I = [a, b]$, definimos:

- la *norma euclídea*
$$\|e\|_2 = \Big(\int_a^b |e(x)|^2 dx\Big)^{\frac{1}{2}}$$
- la *norma del máximo*
$$\|e\|_\infty = \max_{x\in I} |e(x)|$$

Se puede probar que estas definiciones cumplen las propiedades de norma sobre el conjunto $\mathcal{C}([a, b])$ de funciones continuas sobre el interval $[a, b]$.

Como en el caso discreto, se pueden definir las correspondientes normas ponderadas introduciendo una *función peso* $w \in \mathcal{C}([a, b])$ positiva ($w(x) > 0$ sobre I),

$$\|e\|_{2,w} = \Big(\int_a^b w(x)|e(x)|^2 dx\Big)^{\frac{1}{2}}, \quad \|e\|_\infty = \max_{x\in I} |e(x)|w(x).$$

Tanto en el caso discreto como en el continuo, si se elige la norma euclídea hablaremos de *aproximación por mínimos cuadrados*, si se elige la norma del máximo, hablaremos de *aproximación minimax*. En este capítulo nos centraremos en la aproximación por mínimos cuadrados.

4.2. Aproximación por mínimos cuadrados.

4.2.1. Definición del problema

Consideremos un conjunto de abscisas I, ya sea continuo o discreto, unas funciones básicas φ_j ($j = 0 \div n$), y el espacio vectorial que generan $\mathcal{E}_n$. Para cada función f definida sobre I, buscamos una función $f_n^* \in \mathcal{E}_n$ tal que $\|f - f_n^*\|_2$ sea mínima en $\mathcal{E}_n$, es decir,

$$\|f - f_n^*\|_2 = \min_{f_n \in \mathcal{E}_n} \|f - f_n\|_2,$$

donde $\| \quad \|_2$ representa aquí cualquiera de las normas euclídeas, ponderada o no, tanto en el caso continuo como discreto.

- en el caso discreto, $I = x_0, \ldots, x_m$ y si $e = (e_0, \ldots, e_m)$,

$$\|e\|_{2,w} = \Big(\sum_{k=0}^{m} w_k |e_k|^2 \Big)^{\frac{1}{2}}$$

  donde $w = w_0, \ldots, w_m$ es una colección de pesos positivos;

- en el caso continuo, I es un intervalo de la recta real de extremos a y b,

$$\|e\|_{2,w} = \Big(\int_a^b w(x)|e(x)|^2 dx \Big)^{\frac{1}{2}}$$

  donde $w(x) > 0$ es una función peso sobre I.

4.2.2. Productos escalares asociados

La propiedad fundamental de las normas euclídeas es que provienen de sendos productos escalares:

- en el caso discreto,

$$(u, v) = \sum_{k=0}^{m} w_k u_k v_k,$$

- en el caso continuo,

$$(u, v) = \int_a^b w(x) u(x) v(x) dx$$

en el sentido que se cumple, en ambos casos,

$$\|e\|_2^2 = (e, e).$$

Estos productos escalares cumplen las propiedades de definición de producto escalar:

- $(u,u) \geq 0$ y $(u,u) = 0$ si y sólo si $u = 0$,
- $(u,v) = (v,u)$,
- $(a_1u_1 + a_2u_2, v) = a_1(u_1,v) + a_2(u_2,v)$, para funciones u_1, u_2, v sobre I y números reales a_1, a_2 cualesquiera.

4.2.3. Ecuaciones normales.

Sea f_n^* una función sobre I tal que,

$$(f - f_n^*, f_n) = 0, \quad \forall f_n \in \mathcal{E}_n,$$

entonces tenemos,

$$\begin{aligned}\|f - f_n\|_2^2 &= (f - f_n, f - f_n) = (f - f_n^* + f_n^* - f_n, f - f_n^* + f_n^* - f_n) = \\ &= (f - f_n^*, f - f_n^*) + 2(f_n - f_n^*, f_n^* - f) + (f_n^* - f_n, f_n^* - f_n) = \\ &= \|f - f_n^*\|_2^2 + \|f_n^* - f_n\|_2^2,\end{aligned}$$

por tanto,

$$\|f - f_n\|_2^2 = \|f - f_n^*\|_2^2 + \|f_n^* - f_n\|_2^2, \quad \forall f_n \in \mathcal{E}_n.$$

En particular,

$$\|f - f_n\|_2 \geq \|f - f_n^*\|_2, \quad \forall f_n \neq f_n^* \in \mathcal{E}_n,$$

es decir, f_n^* es la única función de $\mathcal{E}_n$ que satisface la condición de aproximación por mínimos cuadrados,

$$\|f - f_n^*\|_2 = \min_{f_n \in \mathcal{E}_n} \|f - f_n\|_2.$$

Dado que $\mathcal{E}_n$ está generado por las funciones básicas φ_i $(i = 0 \div n)$, y $f_n^* \in \mathcal{E}_n$, podemos escribir,

$$f_n^*(x) = \sum_{j=0}^{n} a_j^* \varphi_j(x),$$

y la condición anterior equivale a encontrar los coeficientes a_j^* $(j = 0 \div n)$, tales que satisfagan las llamadas *ecuaciones normales*,

$$\sum_{j=0}^{n} (\varphi_i, \varphi_j) a_j^* = (\varphi_i, f) \quad (i = \div n).$$

Este sistema puede escribirse en forma matricial,

$$Aa^* = b,$$

donde $A = ((\varphi_i, \varphi_j))_{i,j=0\div n}$, $a^* = (a_j^*)_{j=0\div n}$ y $b = ((\varphi_i, f))_{i=0\div n}$.

La matriz A es semidefinido positiva, es decir, simétrica y para cualquier vector $a = (a_0, a_1, \ldots, a_n)^t$, se tiene,

$$a^t A a = \Big(\sum_{j=0}^{n} a_j \varphi_j, \sum_{i=0}^{n} a_i \varphi_i \Big) = \| \sum_{j=0}^{n} a_j \varphi_j \|_2^2 = \|f_n\|_2^2 \geq 0$$

donde f_n viene dada por, $f_n(x) = \sum_{j=0}^{n} a_j \varphi_j(x)$.

Esta relación nos muestra además que las funciones básicas φ_j son linealmente independientes si y sólo si $detA \neq 0$, y que las ecuaciones normales tienen solución única para cualquier f si y sólo si las funciones básicas son linealmente independientes.

4.2.4. Un ejemplo sencillo: la recta de regresión

Tenemos un conjunto de puntos del plano (x_k, y_k) $(k = 0 \div m)$, con $m > 2$, y buscamos una recta $y = a_0 + a_1 x$ que los aproxime de modo que minimice $\sum_{k=0}^{m} d_k^2$ la suma de los cuadrados de las desviaciones $d_k = y_k - a_0 - a_1 x$ $(k = 0 \div m)$.

Este no es más que un problema de aproximación discreta por mínimos cuadrados con $I = x_0, x_1, \ldots, x_m$, $\varphi_0(x) = 1$, $\varphi_1(x) = x$, todos los pesos iguales a 1, y el producto escalar $(u, v) = \sum_{k=0}^{m} u_k v_k$.

Las correspondientes ecuaciones normales,

$$\begin{pmatrix} (\varphi_0, \varphi_0) & (\varphi_0, \varphi_1) \\ (\varphi_1, \varphi_0) & (\varphi_1, \varphi_1) \end{pmatrix} \begin{pmatrix} a_0^* \\ a_1 * \end{pmatrix} = \begin{pmatrix} (\varphi_0, f) \\ (\varphi_1, f) \end{pmatrix}$$

forman el siguiente sistema lineal de dos ecuaciones con dos incógnitas,

$$\begin{pmatrix} m+1 & \sum_{k=0}^{m} x_k \\ \sum_{k=0}^{m} x_k & \sum_{k=0}^{m} x_k^2 \end{pmatrix} \begin{pmatrix} a_0^* \\ a_1 * \end{pmatrix} = \begin{pmatrix} \sum_{k=0}^{m} y_k \\ \sum_{k=0}^{m} x_k y_k \end{pmatrix}$$

cuya solución es:

$$a_1^* = \frac{\overline{xy} - \overline{x} \cdot \overline{y}}{\overline{x^2} - \overline{x}^2},$$
$$a_0^* = \overline{y} - a_1^* \overline{x},$$

donde la barra indica la media, es decir,

$$\overline{x} = \frac{1}{m+1} \sum_{k=0}^{m} x_k \qquad \overline{y} = \frac{1}{m+1} \sum_{k=0}^{m} y_k$$
$$\overline{x^2} = \frac{1}{m+1} \sum_{k=0}^{m} x_k^2 \qquad \overline{xy} = \frac{1}{m+1} \sum_{k=0}^{m} x_k y_k$$

4.3. Ortogonalización.

Una vez reducido el problema de aproximación por mínimos cuadrados, es necesario resolver el sistema de ecuaciones normales asociadas,

$$Aa^* = b,$$

donde $A = ((\varphi_i, \varphi_j))_{i,j=0\div n}$, $a^* = (a_j^*)_{j=0\div n}$ y $b = ((\varphi_i, f))_{i=0\div n}$.

Como la matriz A es semidefinido positiva siempre que las funciones básicas sean linealmente independientes, un método especialmente adecuado es el de Cholesky que requiere $\frac{n^3}{6} + \vartheta(n^2)$ operaciones.

El trabajo preliminar de construcción de las ecuaciones normales requiere generalmente $\frac{1}{2}p(n+1)(n+4)$ operaciones, donde p es el número de operaciones necesarias para cada producto escalar, que normalmente será mayor que $n+1$: en el caso discreto, $p = m+1 \geq n+1$, si I consta de $m+1$ elementos; en el caso continuo hay que calcular las correspondientes integrales. Por tanto, la mayor parte del cálculo corresponde a la formación de las ecuaciones normales. Ahora bien, esta parte del cálculo está fuertemente condicionada por la elección de las funciones básicas de $\mathcal{E}_n$. En general, debemos calcular todos los productos escalares, es decir, todos los coeficientes de la matriz A. También debemos tener en cuenta que la matriz A puede estar mal condicionada si las funciones básicas son "poco independientes"desde el punto de vista numérico. Todo esto nos lleva a pensar que una buena elección de de una base de funciones de $\mathcal{E}_n$ reduce considerablemente los cálculos, por ejemplo con una base de funciones ortogonales ψ_j $(j = 0 \div n)$ respecto al producto escalar, es decir $(\psi_i, \psi_j) = 0, \forall i \neq j$ y $(\psi_i, \psi_i) > 0$ $(i = 0 \div n)$, el sistema es diagonal y la solución es inmediata:

$$f_n^*(x) = \sum_{j=0}^{n} c_j^* \psi_j(x), \quad c_j^* = \frac{(\psi_j, f)}{(\psi_j, \psi_j)}.$$

Dada la simplicidad de estas expresiones, los métodos estándar de resolución de las ecuaciones normales, están basados en la ortogonalización de las funciones básicas, es decir, en la expresión de las ecuaciones normales en una base de funciones ortogonales.

4.3.1. Ortogonalización de Gram-Schmidt

Consideramos una base de funciones $\varphi_i(x)$ $(i = 0, \div n)$ de nuestro espacio vectorial $\mathcal{E}_n$ que está dotado del correspondiente producto escalar $(\cdot, \cdot)$. Buscamos otra base de funciones de $\mathcal{E}_n$, $\psi_i(x)$ $(i = 0, \div n)$, que sean ortogonales respecto a ese producto escalar, es decir,

$$\begin{array}{ll} (\psi_i(x), \psi_j(x)) = 0 & \forall i \neq j \\ (\psi_i(x), \psi_i(x)) > 0 & \forall i = 0, \ldots, n \end{array}$$

El proceso es

$$\begin{array}{l} \psi_0(x) = \varphi_0(x) \\ \psi_i(x) = \varphi_i(x) - \sum_{j=1}^{i-1} \alpha_{ij} \psi_j(x)\,, i = 1, \ldots, n \\ \text{con } \alpha_{ij} = \dfrac{(\varphi_i(x), \psi_j(x))}{(\psi_j(x), \psi_j(x))} \end{array}$$

Cuando el espacio de funciones es el espacio de polinomios de grados $\leq n$, partiendo de la base de polinomios $\varphi_i(x) = x^i$ $(i = 0, \div n)$, tenemos la siguiente recurrencia para calcular una base de polinomios

ortogonales respecto a un producto escalar $(\cdot,\cdot)$ determinado.

$$
\begin{aligned}
&p_0(x) = 1\\
&p_1(x) = x - a_1\\
&p_j(x) = (x - a_j)p_{j-1}(x) - b_j p_{j-2}(x) \quad j \geq 2\\
&\text{con } a_j = \frac{(xp_{j-1}(x), p_{j-1}(x))}{(p_{j-1}(x), p_{j-1}(x))}\\
&\qquad b_j = \frac{(xp_{j-1}(x), p_{j-2}(x))}{(p_{j-2}(x), p_{j-2}(x))}
\end{aligned}
$$

Capítulo 5

Integración y derivación numéricas

5.1. Integración numérica.

La integración numérica es el proceso por medio del cual se genera un valor numérico que aproxima el valor de la integral definida de una función que no posee una primitiva fácil de calcular. Para calcular,

$$\int_a^b f(x)dx,$$

buscamos primero una primitiva, es decir, una función F tal que $F' = f$, y entonces

$$\int_a^b f(x)dx = F(b) - F(a).$$

Pero existen muchas funciones elementales que no poseen primitivas sencillas, por ejemplo, $f(x) = e^{x^2}$. Una primitiva de esta función es,

$$F(x) = \sum_{k=0}^{\infty} \frac{x^{2k+1}}{(2k+1)k!}.$$

Una estrategia muy poderosa para calcular el valor numérico de la integral $\int_a^b f(x)dx$, consiste en reemplazar f por otra función g que aproxime f de manera adecuada en el intervalo de integración y que sea fácil de integrar. Entonces,

$$f \approx g \Rightarrow \int_a^b f(x)dx \approx \int_a^b g(x)dx.$$

Por ejemplo, g puede ser un polinomio que interpole a f en un conjunto de nodos o una serie de Taylor. En el ejemplo anterior,

$$\int_0^1 e^{x^2}dx \approx \int_0^1 \sum_{k=0}^{n} \frac{x^{2k}}{k!}dx \approx \sum_{k=0}^{n} \frac{1}{(2k+1)k!}$$

5.1.1. Integración vía interpolación. Fórmulas de Newton-Cotes

Deseamos calcular,

$$\int_a^b f(x)dx.$$

Elegimos los nodos $x_1, x_1, \ldots, x_n$ en $[a, b]$, e iniciamos un proceso de interpolación polinómica de Lagrange,

$$p(x) = \sum_{i=0}^{n} f(x_i) l_i(x), \quad l_i(x) = \frac{\prod_{j \neq i}(x - x_j)}{\prod_{j \neq i}(x_i - x_j)}, \quad i = 0, 1, \ldots, n.$$

Aproximamos,

$$\int_a^b f(x)dx \approx \int_a^b p(x)dx = \sum_{k=0}^{n} f(x_i) \int_a^b l_i(x)dx.$$

Entonces, para cualquier función $f(x)$ tenemos,

$$\boxed{\int_a^b f(x)dx \approx \sum_{k=0}^{n} A_i f(x_i), \text{ donde } A_i = \int_a^b l_i(x)dx.}$$

llamadas *Fórmulas de Newton-Cotes.*

Regla del trapecio: $n = 1$

El ejemplo más sencillo de una fórmula de Newton-Cotes es la *regla del trapecio* que se obtiene para $n = 1$, es decir, dos nodos que son los extremos del intervalo de integración, $x_0 = a, x_1 = b$. Por tanto, los correspondientes polinomios de Lagrange son,

$$l_0(x) = \frac{b - x}{b - a}, \quad l_1(x) = \frac{x - a}{b - a},$$

e integrando,

$$A_0 = \int_a^b l_0(x)dx = \frac{1}{2}(b - a) = \int_a^b l_1(x)dx = A_1,$$

obteniéndose la conocida *regla del trapecio*,

$$\boxed{\int_a^b f(x)dx \approx \frac{b - a}{2}\big(f(a) + f(b)\big)}$$

Si en el intervalo $[a, b]$ se hace una partición como la siguiente,

$$a = x_0 < x_1 < \ldots < x_n = b,$$

aplicando la regla del trapecio en cada uno de los subintervalos, obtenemos la *regla del trapecio compuesta*,

$$\boxed{\int_a^b f(x)dx = \sum_{i=1}^{n} \int_{x_{i-1}}^{x_i} f(x)dx \approx \frac{1}{2}\sum_{i=1}^{n}(x_i - x_{i-1})(f(x_{i-1}) + f(x_i)).}$$

Los nodos no tiene porque estar espaciados uniformemente, pero si lo están, es decir, tomando $h = \frac{b-a}{n}$, definiendo $x_i = a + hi$ para $i = 0, 1, \ldots, n$, la regla del trapecio compuesta se escribe,

$$\boxed{\int_a^b f(x)dx \approx \frac{h}{2}\Big[f(a) + 2\sum_{i=1}^{n-1} f(a+hi) + f(b)\Big]}$$

Regla de Simpson: $n = 2$

Un ejemplo más complicado de fórmula de Newton-Cotes es la *regla de Simpson* que se obtiene para $n = 2$, es decir, con tres nodos, que son los extremos del intervalo de integración y el punto medio, $x_0 = a, x_1 = \frac{a+b}{2}, x_2 = b$. Procediendo como en el caso anterior,

$$\begin{aligned}
A_0 &= \int_a^b l_0(x)dx = \int_a^b \frac{(x - \frac{a+b}{2})(x-b)}{(a - \frac{a+b}{2})(a-b)}dx = \frac{b-a}{6}, \\
A_1 &= \int_a^b l_1(x)dx = \int_a^b \frac{(x-a)(x-b)}{(\frac{a+b}{2} - a)(\frac{a+b}{2} - b)}dx = 4\frac{b-a}{6}, \\
A_2 &= \int_a^b l_2(x)dx = \int_a^b \frac{(x-a)(x - \frac{a+b}{2})}{(b-a)(b - \frac{a+b}{2})}dx = \frac{b-a}{6},
\end{aligned}$$

obtenemos la regla de Simpson

$$\boxed{\int_a^b f(x)dx \approx \frac{b-a}{6}\big(f(a) + 4f(\frac{a+b}{2}) + f(b)\big)}$$

Si en el intervalo $[a, b]$ se hace una partición con un número par de intervalos, es decir, eligiendo n un número par, definimos $x_i = a + hi$ para $i = 0, 1, \ldots, n$ con $h = \frac{b-a}{n}$, y aplicamos la regla de Simpson a cada par de intervalos, obtenemos la *regla de Simpson compuesta*,

$$\boxed{\int_a^b f(x)dx \approx \frac{h}{3}\sum_{i=1}^{n/2}\Big[f(x_{2i-2}) + 4f(x_{2i-1}) + f(x_{2i})\Big],}$$

reordenando,

$$\boxed{\int_a^b f(x)dx \approx \frac{h}{3}\Big[f(x_0) + 2\sum_{i=2}^{n/2} f(x_{2i-2}) + 4\sum_{i=1}^{n/2} f(x_{2i-1}) + f(x_n)\Big].}$$

Las fórmulas de Newton-Cotes nos llevan a otras fórmulas de integración más generales del tipo,

$$\boxed{\int_a^b f(x)w(x)dx \approx \sum_{k=0}^{n} A_i f(x_i), \text{ donde } A_i = \int_a^b l_i(x)w(x)dx,}$$

donde $w(x)$ es cualquier función de peso.

5.1.2. Método de los coeficientes indeterminados

A medida que elegimos más nodos en las fórmulas de Newton-Cotes, las integrales que tenemos que calcular para obtener los coeficientes A_i se complican. Hay otro procedimiento para calcular el valor de estos coeficientes, el llamado método de los coeficientes indeterminados, que consiste en imponer en la fórmula de integración correspondiente, las condiciones que debe cumplir, es decir, que sea exacta para polinomios del grado correspondiente.

Veamos cómo se obtiene la regla de Simpson con este método. La regla de Simpson es una expresión del tipo,

$$\int_a^b f(x)dx \approx A_0 f(a) + A_1 f(\frac{a+b}{2}) + A_2 f(b),$$

donde tenemos que calcular los coeficientes A_0, A_1 y A_2 de forma que dicha fórmula sea exacta para polinomios de grado más alto posible, como tenemos tres grados de libertad, podemos imponer que esta fórmula sea exacta para polinomios de grado ≤ 2. Basta imponer que la fórmula de integración sea exacta para $f(x) = 1, x, x^2$, obteniendo el siguiente sistema de ecuaciones,

$$\begin{aligned} b-a &= \int_a^b 1dx = A_0 + A_1 + A_2 \\ \frac{b^2-a^2}{2} &= \int_a^b xdx = A_0 a + A_1 \frac{a+b}{2} + A_2 b \\ \frac{b^3-a^3}{3} &= \int_a^b x^2 dx = A_0 a^2 + A_1 \big(\frac{a+b}{2}\big)^2 + A_2 b^2 \end{aligned}$$

de donde podemos despejar $A_0 = A_2 = \frac{b-a}{6}$ y $A_1 = 4\frac{b-a}{6}$, como corresponde.

5.1.3. Cambio de intervalo

A partir de una fórmula de integración numérica en un intervalo de integración determinado, podemos deducir la correspondiente fórmula de integración numérica para cualquier otro intervalo de integración mediante un cambio de variable lineal. Si la primera fórmula es exacta para polinomios de un cierto grado, lo mismo será cierto para la segunda fórmula. Veamos cómo se lleva a cabo.

Supongamos que contamos con la siguiente fórmula de integración numérica,

$$\int_c^d f(t)dt \approx \sum_{i=0}^{n} A_i f(t_i)$$

No nos importa el origen de esta fórmula, sin embargo, supongamos que es exacta para todos los polinomios de grado $\leq m$. Si necesitamos esta fórmula para algún otro intervaalo, $[a, b]$, definimos primero una función lineal $\lambda(t)$ tal que, si t recorre $[c, d]$, entonces $\lambda(t)$ recorre $[a, b]$. La expresión explícita de $\lambda(t)$ es,

$$\lambda(t) = \frac{b-a}{d-c}t + \frac{ad-bc}{d-c}.$$

Por tanto, para el cambio de variable en la integral tenemos $x = \lambda(t) \Rightarrow dx = \lambda'(t)dt = \frac{b-a}{d-c}dt$, de donde,

$$\int_a^b f(x)dx = \frac{b-a}{d-c}\int_{\lambda^{-1}(a)=c}^{\lambda^{-1}(b)=d} f(\lambda(t))dt \approx \frac{b-a}{d-c}\sum_{i=0}^{n} A_i f(\lambda(t_i)),$$

fórmula que seguirá siendo exacta para todos los polinomios de grado $\leq m$.

Ejemplo 5.1 Deducir la fórmula de Simpson para el intervalo $[0, 1]$ por el método de los coeficientes indeterminados, es muy sencillo, después utilizando este cambio de variable podremos tener la fórmula de Simpson para cualquier otro intervalo.

5.1.4. Cuadratura gaussiana.

En la sección anterior hemos visto cómo generar fórmulas de integración numérica, también llamadas fórmulas de cuadratura, del tipo,

$$\int_a^b f(x)dx \approx \sum_{k=0}^{n} A_i f(x_i),$$

que son exactas para polinomios de grado $\leq n$. En estas fórmulas, la elección de los nodos $x_0, x_1, \ldots, x_n$ se hace a priori, y una vez fijados los nodos, los coeficientes A_i se determinan de manera unívoca imponiendo la igualdad en la fórmula de cuadratura para todos los polinomios de grado $\leq n$. Nos preguntamos ahora si una elección de nodos puede ser mejor que otra, por ejemplo, nos preguntamos si podría haber un conjunto particular de nodos para los que todos los coeficientes A_i fueran todos iguales, simplificando así la fórmula de cuadratura.

Partiendo de las fórmulas de cuadratura más generales, a saber,

$$\int_a^b f(x)w(x)dx \approx \sum_{k=0}^{n} A_i f(x_i),$$

donde $w(x)$ es una función de peso positiva, sabemos que esta fórmula es exacta para polinomios de grado $\leq n$ si y sólo si,

$$A_i = \int_a^b w(x) \prod_{j=0, j\neq i}^{j=n} \frac{x-x_j}{x_i-x_j}dx.$$

En vista de que se cuenta con $n+1$ coeficientes A_i y $n+1$ nodos x_i, sin que exista ninguna restricción a priori sobre estos últimos, sospechamos que se pueden encontrar fórmulas de cuadratura que sean exactas para polinomios de grado $\leq 2n+1$. El siguiente resultado nos indica dónde colocar los nodos para que esto sea posible, obteniendo las llamadas *fórmulas de cuadratura gaussianas.*

Teorema 5.1 Dada una función de peso positiva w, y un polinomio q no nulo de grado $n+1$ que sea w-ortogonal a π_n, espacio de polinomios de grado $\leq n$, en el sentido de que para cualquier $p \in \pi_n$ se tiene,

$$\int_a^b q(x)p(x)w(x)dx = 0,$$

entonces, si $x_0, x_1, \ldots, x_n$ son las raíces de q, la fórmula de cuadratura,

$$\int_a^b f(x)w(x)dx \approx \sum_{k=0}^{n} A_i f(x_i), \quad A_i = \int_a^b w(x) \prod_{j=0, j\neq i}^{j=n} \frac{x - x_j}{x_i - x_j} dx,$$

será exacta para todo polinomio de grado $\leq 2n+1$.

Demostración:

Sea $f \in \pi_{2n+1}$, dividimos f entre q obteniendo un cociente p y un resto r, $f = pq + r$. Por tanto $p, r \in \pi_n$ y $f(x_i) = r(x_i)$, para $i = 0, 1, \ldots, n$. Integrando,

$$\int_a^b f(x)w(x)dx = \underbrace{\int_a^b q(x)p(x)w(x)dx}_{=0} + \int_a^b r(x)w(x)dx = \sum_{k=0}^{n} A_i r(x_i) = \sum_{k=0}^{n} A_i f(x_i).$$

como queríamos demostrar. ■

Para poder aplicar la fórmula de integración en ese conjunto de nodos que son las raíces de q, es necesario que éstas sean reales y simples. Esto se deduce de forma inmediata del siguiente resultado.

Teorema 5.2 Sea w una función de peso positiva en $\mathcal{C}[a,b]$. Sea q un elemento no nulo de $\mathcal{C}[a,b]$ que sea w-ortogonal a π_n. Entonces q cambia de signo en (a,b) al menos $n+1$ veces.

Demostración:

Como $1 \in \pi_n$, entonces $\int_a^b q(x)w(x)dx = 0$, mostrando que q cambia de signo al menos una vez en (a,b) ya que la función de peso w es positiva.

Supongamos que q cambia de signo en sólo r ocasiones, con $r \leq n$. Escogemos puntos t_i de manera que $a = t_0 < t_1 < \ldots < t_r < t_{r+1} = b$, y tal que q sólo tiene un signo en cada intervalo definido por estos puntos. Entonces el polinomio $p(x) = \prod_{i=1}^{r}(x - t_i)$ tiene la misma propiedad respecto al signo que q y por lo tanto $\int_a^b q(x)p(x)w(x)dx \neq 0$, pero esto es una contradicción puesto que $p \in \pi_n$, a no ser que $r = n+1$ como queríamos demostrar. ■

El cálculo de los coeficientes A_i en las fórmulas de cuadratura gaussianas, se realiza del mismo modo que en el caso de las fórmulas anteriores no gaussianas, una vez determinados los nodos x_i. Podemos calcular directamente su valor mediante las integrales de los correspondientes polinomios de Lagrange, o mediante el método de los coeficientes indeterminados.

A su vez, los nodos son las raíces de un cierto polinomio q_{n+1} que queda univocamente determinado mediante dos condiciones:

- q_{n+1} es un polinomio mónico de grado $n+1$, es decir, el coeficiente de x^{n+1} es la unidad.
- q_{n+1} es w-ortogonal a π_n, es decir, $\int_a^b q_{n+1}(x)w(x)p(x)dx = 0, \quad \forall p \in \pi_n$

Estos son los llamados polinomios ortogonales que podemos calcular con la fórmula recurrente vista en el Tema 5,

$$\begin{aligned} &p_0(x) = 1 \\ &p_1(x) = x - a_1 \\ &p_j(x) = (x - a_j)p_{j-1}(x) - b_j p_{j-2}(x) \quad j \geq 2 \\ &\text{con } a_j = \frac{(xp_{j-1}(x), p_{j-1}(x))}{(p_{j-1}(x), p_{j-1}(x))} \\ &\qquad b_j = \frac{(xp_{j-1}(x), p_{j-2}(x))}{(p_{j-2}(x), p_{j-2}(x))} \end{aligned}$$

de forma que el polinomio p_{n+1} calculado con esta fórmula será ortogonal a π_n en el sentido del producto escalar usado en la misma, que en nuestro caso debe ser,

$$(p, q) = \int_a^b p(x)w(x)q(x)dx.$$

Ejemplo 5.2 Encontrar la fórmula de cuadratura gaussiana para $[a, b] = [-1, 1]$, $w(x) = 1$ y $n = 1$.

5.2. Derivación numérica.

Aunque haya reglas bien conocidas para derivar las funciones más usuales, no siempre pueden ser utilizadas (por ejemplo, en funciones dadas por tablas de valores), o no es conveniente (por ejemplo, en funciones con expresiones analíticas muy complicadas). En estos casos debemos recurrir a técnicas numéricas que, partiendo de los valores de la función en diversas abscisas, nos permitirá calcular una aproximación al valor de alguna de sus derivadas en una abscisa próxima.

5.2.1. Derivadas primeras.

La derivada de una función f en un punto x_0 es,

$$f'(x_0) = \lim_{h \to 0} \frac{f(x_0 + h) - f(x_0)}{h}$$

lo que nos da una forma obvia de generar una aproximación de $f'(x_0)$,

$$f'(x_0) \approx \frac{f(x_0+h)-f(x_0)}{h}$$

para valores pequeños de h. Aunque esto parezca muy evidente no es demasiado útil debido a los errores de redondeo, pero es un buen punto de partida.

Fórmulas de derivación interpolatoria.

Para conocer mejor el error que se comete con este tipo de aproximaciones vamos a utilizar las fórmulas de interpolación polinómica de las que conocemos el error.

Sean $x_0, x_1, \ldots, x_n$, $n+1$ puntos distintos de un intervalo I en el que $f \in C^{n+1}(I)$, por las fórmulas de interpolación polinómica sabemos que, para algún $\xi_x \in I$,

$$f(x) = \sum_{k=0}^{n} f(x_k)L_k(x) + f^{n+1)}(\xi_x)\frac{\prod_{k=0}^{n}(x-x_k)}{(n+1)!}$$

donde $L_k(x)$ es el k-ésimo polinomio de Lagrange de los nodos $x_0, x_1, \ldots, x_n$, es decir,

$$L_k(x) = \frac{\prod_{i\neq k}(x-x_i)}{\prod_{i\neq k}(x_k-x_i)}$$

Si derivamos esta expresión, obtenemos,

$$f'(x) = \sum_{k=0}^{n} f(x_k)L_k'(x) + D_x(f^{n+1)}(\xi_x))\frac{\prod_{k=0}^{n}(x-x_k)}{(n+1)!} + f^{n+1)}(\xi_x)D_x\Big(\frac{\prod_{k=0}^{n}(x-x_k)}{(n+1)!}\Big)$$

que en el caso en que x sea una de los nodos x_j, se reduce a,

$$f'(x_j) = \sum_{k=0}^{n} f(x_k)L_k'(x_j) + \frac{f^{n+1)}(\xi_j)}{(n+1)!}\prod_{\substack{k=0\\k\neq j}}^{n}(x_j-x_k)$$

llamada **fórmula de derivación de $n+1$ puntos** para aproximar $f'(x_j)$.

En términos generales, la utilización de más puntos produce una mayor exactitud aunque no es conveniente dada la cantidad de evaluaciones funcionales y el aumento del error de redondeo. Las fórmulas más comunes son las de 2, 3 y 5 puntos, que veremos con más detenimiento.

Fórmulas de 2 puntos: $n = 1$

Supongamos que $x_0 \in (a,b)$, donde $f \in \mathcal{C}^2[a,b]$, y que $x_1 = x_0 + h$ para algún $h \neq 0$ suficientemente pequeño para asegurarnos que $x_1 \in [a,b]$. Construimos el primer polinomio de Lagrange para f

determinado por x_0 y x_1 con su término de error,

$$f(x) = f(x_0)\frac{x-x_1}{x_0-x_1} + f(x_1)\frac{x-x_0}{x_1-x_0} + f''(\xi_x)\frac{(x-x_0)(x-x_1)}{2}$$

para cierto $\xi_x \in [a,b]$. Sustituyendo $x_1 = x_0 + h$,

$$f(x) = f(x_0)\frac{x-x_0-h}{-h} + f(x_0+h)\frac{x-x_0}{h} + f''(\xi_x)\frac{(x-x_0)(x-x_0-h)}{2}$$

Al diferenciar, obtenemos,

$$f'(x) = f(x_0)\frac{-1}{h} + f(x_0+h)\frac{1}{h} + D_x\big(f''(\xi_x)\big)\frac{(x-x_0)(x-x_0-h)}{2} + f''(\xi_x)\frac{2(x-x_0)-h}{2}$$

de donde, tomando $x = x_0$ tenemos,

$$\boxed{f'(x_0) = \frac{f(x_0+h)-f(x_0)}{h} - \frac{h}{2}f''(\xi_{x_0})}$$

Para valores pequeños de h podemos utilizar $(f(x_0+h)-f(x_0))/h$ para aproximar $f'(x_0)$ con un error acotado por $M|h|/2$ donde M es una cota de $|f''(x)|$ en $[a,b]$. Esta fórmula se llama *fórmula de la diferencia progresiva* si $h > 0$, y *fórmula de la diferencia regresiva* si $h < 0$.

Fórmulas de 3 puntos: $n = 2$

Supongamos que $x_0 \in (a,b)$, donde $f \in \mathcal{C}^3[a,b]$, $x_1 = x_0 + h$ y $x_2 = x_0 + 2h$ para algún $h \neq 0$ suficientemente pequeño para asegurarnos que $x_1, x_2 \in [a,b]$. Construimos el primer polinomio de Lagrange para f determinada por x_0, x_1 y x_2 con su término de error,

$$f(x) = f(x_0)L_0(x) + f(x_1)L_1(x) + f(x_2)L_2(x) + f'''(\xi_x)\frac{(x-x_0)(x-x_1)(x-x_2)}{6}$$

para cierto $\xi_x \in [a,b]$, donde,

$$\begin{aligned}
L_0(x) &= \frac{(x-x_1)(x-x_2)}{(x_0-x_1)(x_0-x_2)} \quad \Rightarrow \quad L_0'(x) = \frac{2x-x_1-x_2}{(x_0-x_1)(x_0-x_2)} \\
L_1(x) &= \frac{(x-x_0)(x-x_2)}{(x_1-x_0)(x_1-x_2)} \quad \Rightarrow \quad L_1'(x) = \frac{2x-x_0-x_2}{(x_1-x_0)(x_1-x_2)} \\
L_2(x) &= \frac{(x-x_0)(x-x_1)}{(x_2-x_0)(x_2-x_1)} \quad \Rightarrow \quad L_2'(x) = \frac{2x-x_0-x_1}{(x_2-x_0)(x_2-x_1)}
\end{aligned}$$

de modo que para $x = x_j$ para $j = 0,1,2$, tenemos,

$$\begin{aligned}
f'(x_j) = \quad & f(x_0)\frac{2x-x_1-x_2}{(x_0-x_1)(x_0-x_2)} + f(x_1)\frac{2x-x_0-x_2}{(x_1-x_0)(x_1-x_2)} + f(x_2)\frac{2x-x_0-x_1}{(x_2-x_0)(x_2-x_1)} \\
& + \frac{1}{6}f'''(\xi_x)\prod_{\substack{k=0 \\ k\neq j}}^{2}(x_j - x_k)
\end{aligned}$$

Tomando ahora $x_1 = x_0 + h$ y $x_2 = x_0 + 2h$, la fórmula anterior queda:

- para $x_j = x_0$,

$$f'(x_0) = \frac{1}{h}\Big[-\frac{3}{2}f(x_0) + 2f(x_0+h) - \frac{1}{2}f(x_0+2h)\Big] + \frac{h^2}{3}f'''(\xi_0)$$

- para $x_j = x_1 = x_0 + h$,

$$f'(x_0+h) = \frac{1}{h}\Big[-\frac{1}{2}f(x_0) + \frac{1}{2}f(x_0+2h)\Big] - \frac{h^2}{6}f'''(\xi_1)$$

- para $x_j = x_2 = x_0 + 2h$,

$$f'(x_0+2h) = \frac{1}{h}\Big[\frac{1}{2}f(x_0) - 2f(x_0+h) + \frac{3}{2}f(x_0+2h)\Big] + \frac{h^2}{3}f'''(\xi_2)$$

Por razones de comodidad, podemos sustituir en la segunda fórmula x_0 por $x_0 + h$ y en la tercera fórmula x_0 por $x_0 + 2h$, obteniendo,

$$\begin{aligned}
f'(x_0) = &\quad \frac{1}{2h}\Big[-3f(x_0) + 4f(x_0+h) - f(x_0+2h)\Big] + \frac{h^2}{3}f'''(\xi_0)\\
f'(x_0) = &\quad \frac{1}{2h}\Big[-f(x_0-h) + f(x_0+h)\Big] - \frac{h^2}{6}f'''(\xi_1)\\
f'(x_0) = &\quad \frac{1}{2h}\Big[f(x_0-2h) - 4f(x_0-h) + 3f(x_0)\Big] + \frac{h^2}{3}f'''(\xi_2)
\end{aligned}$$

donde la primera y última fórmula son iguales sin más que sustituir h por $-h$. Por tanto, en realidad hay dos fórmulas de 3 puntos, la *fórmula de diferencias finitas centrada*:

$$\boxed{f'(x_0) = \frac{1}{2h}\Big[-f(x_0-h) + f(x_0+h)\Big] - \frac{h^2}{6}f'''(\xi_0)},$$

que emplea datos a ambos lados de x_0 y por ello tiene un error aproximadamente la mitad que la otra fórmula, ya sea para $h > 0$ o para $h < 0$, que emplea únicamente datos a un lado de x_0:

$$\boxed{f'(x_0) = \frac{1}{2h}\Big[-3f(x_0) + 4f(x_0+h) - f(x_0+2h)\Big] + \frac{h^2}{3}f'''(\xi_1)}$$

Fórmulas de 5 puntos: $n = 4$

Para obtener las fórmulas de 5 puntos se evalúa la función en otros dos puntos más, por ejemplo, $x_0 - 2h$, $x_0 - h$, x_0, $x_0 + h$ y $x_0 + 2h$, pero cuyo término de error tiene la forma $\theta(4)$. Una de estas fórmulas es,

$$f'(x_0) = \frac{1}{12h}\Big[f(x_0-2h) - 8f(x_0-h) + 8f(x_0+h) - f(x_0+2h)\Big] + \frac{h^4}{30}f^v(\xi)$$

donde ξ está entre $x_0 - 2h$ y $x_0 + 2h$.

5.2.2. Derivadas de orden superior.

Se pueden obtener fórmulas para aproximar derivadas de orden superior de una función en un punto x_0 utilizando exclusivamente los valores de la función en varios puntos. La obtención de estas fórmulas por el procedimiento anterior es muy laboriosa, pero usando desarrollos de Taylor alrededor de un punto se pueden obtener dichas fórmulas de modo más sencillo.

Veamos un ejemplo: hagamos el desarrollo de Taylor de grado 3 de una funcióm f en un entorno de x_0 y evaluemos en $x_0 - h$ y $x_0 + h$.

$$f(x_0 + h) = f(x_0) + f'(x_0)h + \frac{1}{2}f''(x_0)h^2 + \frac{1}{6}f'''(x_0)h^3 + \frac{1}{24}f^{iv}(\xi_1)h^4$$

$$f(x_0 - h) = f(x_0) - f'(x_0)h + \frac{1}{2}f''(x_0)h^2 - \frac{1}{6}f'''(x_0)h^3 + \frac{1}{24}f^{iv}(\xi_{-1})h^4$$

donde $x_0 - h < \xi_{-1} < x_0 < \xi_1 < x_0 + h$.

Sumando,

$$f(x_0 + h) + f(x_0 - h) = 2f(x_0) + f''(x_0)h^2 + \frac{h^4}{24}\big(f^{iv}(\xi_1) + f^{iv}(\xi_{-1})\big)$$

y despejando $f''(x_0)$,

$$f''(x_0) = \frac{1}{h^2}\big[f(x_0 - h) - 2f(x_0) + f(x_0 + h)\big] - \frac{h^2}{24}\big(f^{iv}(\xi_1) + f^{iv}(\xi_{-1})\big)$$

Suponiendo que f^{iv} es continua en $[x_0 - h, x_0 + h]$, por el teorema del valor intermedio, existe un ξ entre ξ_{-1} y ξ_1 con

$$f^{iv}(\xi) = \frac{1}{2}(f^{iv}(\xi_{-1}) + f^{iv}(\xi_1))$$

por tanto,

$$\boxed{f''(x_0) = \frac{1}{h^2}\big[f(x_0 - h) - 2f(x_0) + f(x_0 + h)\big] - \frac{h^2}{12}f^{iv}(\xi)}$$

para ξ entre ξ_{-1} y ξ_1.