

Apuntes de teórico de Métodos Numéricos

Matías Davyt

6 de diciembre de 2012

Índice

1. Aritmética de punto flotante y estimación de errores	3
1.1. Errores absolutos y relativos	3
1.2. Representación en punto flotante	3
1.2.1. Representación por truncamiento	3
1.2.2. Representación por redondeo	3
1.2.3. Error global y local	3
1.2.4. Epsilon de la máquina	4
1.2.5. Propiedades	4
1.3. Aproximación de derivadas mediante coeficientes incrementales	4
1.3.1. Coeficiente incremental hacia adelante	4
1.3.2. Diferencia centrada	4
1.3.3. Diferencia hacia atras	4
1.3.4. Derivada segunda	4
1.4. Extrapolación de Richardson	5
1.4.1. Teorema	5
2. Sistemas de ecuaciones lineales	6
2.1. Casos particulares de métodos directos	6
2.2. Eliminación Gaussiana	6
2.2.1. Pivoteo parcial	6
2.2.2. Proposición	7
2.3. Métodos iterativos	7
2.3.1. Métodos consistentes	7
2.3.2. Condición necesaria y suficiente de convergencia	7
2.3.3. Método de Jacobi	7
2.3.4. Método de Gauss-Seidel	7
3. Sistemas de ecuaciones no lineales	8
3.1. Método iterativo general (MIG)	8
3.1.1. Definición: orden de convergencia	8
3.1.2. Teorema	8
3.1.3. Definición: función contractiva	8

3.1.4.	Teorema	8
3.1.5.	Teorema	9
3.1.6.	Teorema	9
3.1.7.	Método de Newton-Raphson	9
3.1.8.	Teorema	9
3.1.9.	Método de Newton-Raphson para sistemas no lineales	9
3.1.10.	Método de la secante	10
3.1.11.	Método de Steffensen	10
4.	Mínimos cuadrados	11
4.1.	Problema de mínimos cuadrados lineal	11
4.1.1.	Teorema: descomposición QR	11
4.1.2.	Uso de descomposición QR para resolver el PMC lineal	11
4.1.3.	Teorema: descomposición SVD	11
4.1.4.	Uso de descomposición SVD para resolver el PMC lineal	12
4.2.	Teorema	12
4.3.	Ecuaciones normales	12
4.4.	Problema de mínimos cuadrados no lineal	12
4.4.1.	Método de Gauss-Newton	13
5.	Interpolación polinómica	13
5.1.	Ecuación de Vandermonde	13
5.2.	Estudio del error en la interpolación polinómica	13
5.3.	Interpolación de Lagrange	14
5.4.	Interpolación de Hermite	14
5.5.	Interpolación de Newton	15
5.6.	Interpolación lineal a trozos	15
5.7.	Curvas de Bezier	16
5.7.1.	Propiedad	16
5.8.	Curva paramétrica de Bezier	16
6.	Ecuaciones diferenciales ordinarias	16
6.1.	Método de Euler	17
6.2.	Notación	17
6.3.	Definición	17
6.4.	Teorema	17
6.5.	Estabilidad numérica	18
6.6.	Problema Test	18
6.7.	Región de estabilidad asociada a un método	18
6.8.	Método del trapecio	18
6.9.	Métodos consistentes	18
6.10.	Método de Euler hacia atrás	19
6.11.	Método de punto medio	19
6.12.	Métodos de Runge-Kutta	19

1. Aritmética de punto flotante y estimación de errores

1.1. Errores absolutos y relativos

Dada una magnitud $x \in \mathbb{R}^n$ incógnita y una estimación $\bar{x} \in \mathbb{R}^n$, definimos:

- Error absoluto: $\Delta x = |x - \bar{x}|$
- Error relativo: $\delta x = \frac{|x - \bar{x}|}{|x|}$

1.2. Representación en punto flotante

La representación de números reales en una cierta arquitectura se hace mediante representación en punto flotante.

Una forma de definir o representar este conjunto es la siguiente:

$$FP = \{\sigma \times (0.d_1d_2\dots d_t) \times \beta^e; 0 \leq d_i; d_1 \neq 0, I \leq e \leq U\}$$

donde:

- σ es el signo (+, -)
- $(0.d_1d_2\dots d_t) = \sum_{i=1}^t d_i \times \beta^{-i}$
- e es el exponente de la base β , el cual está acotado inferior y superiormente ($I \leq e \leq U$)

1.2.1. Representación por truncamiento

Sea FP una cierta representación de punto flotante. Dado un real x , su representación por truncamiento en FP viene dada por: $\bar{x} = \max\{y \in FP / y \geq x\}$

1.2.2. Representación por redondeo

Sea FP una cierta representación de punto flotante. Dado un real x , su representación por redondeo en FP viene dada por: $\bar{x} = \text{Avgmin}\{|y - x|; y \in FP\}$

Notación: dado $x \in \mathbb{R}$ y un cierto FP, denotamos por $FP(x)$ a su representación en punto flotante.

1.2.3. Error global y local

Dado $x \in \mathbb{R}$ y un cierto FP, sea $\bar{x} = FP(x)$. Llamaremos error global a $E_x = x - \bar{x}$. Llamaremos error local a $e_x = \frac{x - \bar{x}}{x}$

1.2.4. Epsilon de la máquina

Dado un cierto FP, definimos como epsilon de la máquina al siguiente valor:
 $\epsilon_M = \text{Min}\{x \in \mathbb{R} / FP(1+x) > 1\}$

Asumiremos de aquí en más que las estructuras FP cumplen: $\forall x \in \mathbb{R} : FP(x) = x(1 + \delta_x)$, con $|\delta_x| \leq \epsilon_M$

Se puede probar que $FP(x \text{ op } y) = (x \text{ op } y)(1 + \delta_{op})$, con $\delta_{op} \leq \epsilon_M$. Este valor dependerá de la operación

1.2.5. Propiedades

- $|E_{x+y}| \approx |E_x| + |E_y|$
- $|e_{x+y}| \approx |e_x| + |e_y|$
- $|e_{x/y}| \approx |e_x| + |e_y|$ si $y \neq 0$
- Sea $z=f(x_1, \dots, x_n)$, entonces $|E_z| \approx \sum_{i=1}^n |\frac{\partial f}{\partial x_i}(\bar{x})| \times |E_{x_i}|$

1.3. Aproximación de derivadas mediante coeficientes incrementales

1.3.1. Coeficiente incremental hacia adelante

$$f'(x) = \Delta_{f,h}(x) = \frac{f(x+h) - f(x)}{h}$$

1.3.2. Diferencia centrada

$$f'(x) = \Delta_{f,c}(x) = \frac{f(x+h) - f(x-h)}{2h}$$

1.3.3. Diferencia hacia atras

$$f'(x) = \Delta_{f,a}(x) = \frac{f(x) - f(x-h)}{h}$$

1.3.4. Derivada segunda

$$\begin{aligned} f''(x) &\approx \frac{f'(x+h) - f'(x)}{h} \approx \frac{\frac{f(x+h)-f(x)}{h} - \frac{f(x)-f(x-h)}{h}}{h} \\ &\Rightarrow f''(x) \approx \frac{f(x+h) - 2f(x) + f(x-h)}{h^2} \end{aligned}$$

1.4. Extrapolación de Richardson

Sea $\bar{f}(x, h)$ un estimador de $f(x)$, tal que $\lim_{h \rightarrow 0} \bar{f}(x, h) = f(x)$

Supongamos que se cumple la siguiente relación: $\bar{f}(x, h) = f(x) + c_p h^p + o(h^r)$ con $r > p$.

Queremos encontrar un mejor estimador de $f(x)$, intentando eliminar el término en h^p .

Sea $q > 1$. Tenemos que

$$\bar{f}(x, h) = f(x) + c_p h^p + o(h^r) \quad (1)$$

Evaluamos \bar{f} en h/q

$$\bar{f}(x, h/q) = f(x) + c_p (h/q)^p + o(h^r)$$

Multiplicando por q^p

$$q^p \bar{f}(x, h) = q^p f(x) + c_p h^p + o(h^r) \quad (2)$$

Restando (2) y (1), y luego dividiendo entre $q^p - 1$ se obtiene

$$\bar{f}(x, h/q) + \frac{\bar{f}(x, h/q) - \bar{f}(x, h)}{q^p - 1} = f(x) + o(h^r)$$

1.4.1. Teorema

Sea $\bar{f}(x, h) = f(x) + c_1 h^{p_1} + c_2 h^{p_2} + c_3 h^{p_3} + \dots$ con $0 < p_1 < p_2 < \dots$

Es decir:

$$\bar{f}(x, h) = f(x) + \sum_{i=1}^{\infty} c_i h^{p_i}$$

Se define:

- $\bar{f}_1(x, h) = \bar{f}(x, h)$
- $\bar{f}_{k+1}(x, h) = \bar{f}_k(x, h/q) + \frac{\bar{f}_k(x, h/q) - \bar{f}_k(x, h)}{q^{p_k} - 1}$

Con esta definición se cumple que

$$\bar{f}_n(x, h) = f(x) + \sum_{i=n}^{\infty} c_i^{(n)} h^{p_i}$$

2. Sistemas de ecuaciones lineales

Objetivo: Resolver sistemas de la forma $Ax=b$, con $A \in \mathbb{R}^{n \times n}$, $b \in \mathbb{R}^n$, $x \in \mathbb{R}^n$.

Nos enfocaremos en el caso en que A es no singular, es decir que es invertible.

Existen dos tipos de métodos para la resolución de sistemas lineales de esta forma:

- **Métodos directos:** en un número finito de operaciones, encuentro la solución exacta al problema
- **Métodos iterativos:** se genera una sucesión $\{x^{(k)}\}$, tal que $\lim_{k \rightarrow \infty} x^{(k)} = \bar{x}$, siendo \bar{x} la solución exacta.
Genero iterativamente los $x^{(k)}$, y detengo la iteración cuando $\|x^{(k)} - x^{(k+1)}\| < \epsilon$ y/o $\|Ax^{(k)} - b\| < \epsilon$, para un cierto epsilon.

2.1. Casos particulares de métodos directos

1. A es triangular superior

$$x_i = \frac{b_i - \sum_{k=i+1}^n A_{i,k}x_k}{A_{i,i}}$$

2. A es triangular inferior

$$x_i = \frac{b_i - \sum_{k=1}^{i-1} A_{i,k}x_k}{A_{i,i}}$$

2.2. Eliminación Gaussiana

Quiero resolver $Ax=b$ con A genérica.

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

En el paso k utilizo la fila k para eliminar x_k de las $(n-k)$ ecuaciones restantes. Para todo $i \in [k+1, n]$, reemplazo la i -ésima fila por (fila i)- m_{ik} ·(fila k), siendo $m_{ik} = \frac{a_{ik}}{a_{kk}}$

2.2.1. Pivoteo parcial

Supongamos que estamos en el k -ésimo paso de la eliminación y $a_{kk}^{(k)} = 0$.

Calculamos $|a_{rk}^{(k)}| = \max_{i=k+1 \dots n} \{|a_{ik}|\}$

Para la posición r calculada, intercambiamos las filas k y r .

2.2.2. Proposición

1. El número de operaciones necesario para resolver sistemas triangulares es del orden de $\frac{n^2}{2}$
2. El número de operaciones de la escalerización Gaussiana es del orden de $\frac{n^3}{3}$

2.3. Métodos iterativos

Para resolver un sistema lineal de la forma $Ax=b$, se genera una sucesión $\{x^{(k)}\}$, tal que $\lim_{k \rightarrow \infty} x^{(k)} = \bar{x}$, siendo \bar{x} la solución exacta.

Genero iterativamente los $x^{(k)}$, y detengo la iteración cuando $\|x^{(k)} - x^{(k+1)}\| < \epsilon$ y/o $\|Ax^{(k)} - b\| < \epsilon$, para un cierto epsilon.

Dado un $x^{(0)}$, para todo k se calcula $x^{(k+1)} = Bx^{(k)} + f$, donde:

- B es una matriz de iteración
- f es un vector obtenido a partir de b

2.3.1. Métodos consistentes

Se dice que un método iterativo es consistente si $x=Bx+f$

2.3.2. Condición necesaria y suficiente de convergencia

Dado un método consistente, la secuencia $\{x^{(k)}\}$ converge $\Leftrightarrow \rho(B) < 1$, donde $\rho(B)$ es el radio espectral de la matriz de iteración B .

2.3.3. Método de Jacobi

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

2.3.4. Método de Gauss-Seidel

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

3. Sistemas de ecuaciones no lineales

Se quieren resolver ecuaciones de la forma $f(x)=0$, con f no lineal.

3.1. Método iterativo general (MIG)

El problema $f(x)=0$ se lleva al problema de encontrar un punto fijo ¹ de una ecuación del tipo $g(x)=x$.

El MIG consiste en generar una sucesión $\{x_i\}_{i \geq 0}$ dada por $\begin{cases} x_0 \text{ dado} \\ x_{i+1} = g(x_i) \end{cases}$

Si la sucesión generada por e MIG es convergente a un punto α y g es continua, entonces α es un punto fijo de g .

3.1.1. Definición: orden de convergencia

Dada una sucesión $\{x_i\}_{i \geq 0}$ convergente a α , diremos que su orden de convergencia es p si se cumple que:

$$\lim_{i \rightarrow \infty} \frac{|x_{i+1} - \alpha|}{|x_i - \alpha|^p} = \beta \neq 0$$

A β se le llama velocidad de convergencia.

3.1.2. Teorema

Sea $g : [a, b] \rightarrow [a, b]$ continua $\Rightarrow g$ tiene al menos un punto fijo.

3.1.3. Definición: función contractiva

Una función $g : A \subset \mathbb{R}^n \rightarrow \mathbb{R}^n$ se dice que es contractiva si $\exists k \in (0, 1)$ tal que $\|g(\vec{x}_1) - g(\vec{x}_2)\| \leq k \|\vec{x}_1 - \vec{x}_2\| \forall \vec{x}_1, \vec{x}_2 \in A$

3.1.4. Teorema

Sea una función $g : [a, b] \rightarrow [a, b]$ contractiva, entonces:

1. Existe un único punto fijo $\alpha \in (a, b)$
2. La sucesión generada por el MIG converge a $\alpha \forall x_0 \in [a, b]$
3. $|x_{n+1} - \alpha| \leq k_\alpha |x_n - \alpha| < |x_n - \alpha|$

¹Se denomina punto fijo de $g(x)$ a un punto α tal que $g(\alpha) = \alpha$

3.1.5. Teorema

Sea $g : [a, b] \rightarrow \mathbb{R}$ con derivada continua y α punto fijo de g , entonces:

1. Si $|g'(\alpha)| < 1$, la sucesión generada por el MIG converge a α , tomando x_0 lo suficientemente cercano
2. Si $|g'(\alpha)| > 1$, la sucesión generada por el MIG no es convergente

3.1.6. Teorema

Sea α punto fijo de $g(x)$, $g(x)$ continua con sus derivadas primera, segunda, ..., p -ésima continuas en un entorno de α , y además $g^{(i)}(\alpha) = 0 \ \forall i = 1 \dots p-1$ y $g^{(p)}(\alpha) \neq 0$. Entonces el orden de convergencia del MIG es p , con $\beta = \frac{1}{p!}|g^{(p)}(\alpha)|$

3.1.7. Método de Newton-Raphson

Queremos resolver la ecuación $f(x)=0$. La ecuación de la recta tangente a $f(x)$ en el punto $(x_i, f(x_i))$ es $z(x) - f(x_i) = f'(x_i)(x - x_i)$. Imponiendo $z=0$, encontramos el punto de corte de dicha recta con el eje de las abscisas. A dicho punto lo llamamos x_{i+1}

$$\Rightarrow x_{i+1} = x_i - \frac{f(x_i)}{f'(x_i)}$$

3.1.8. Teorema

Sea $f(x)$ una función tal que f'' es continua en un entorno de α y $f'(\alpha) \neq 0$ (α raíz de $f(x)$).
 \Rightarrow La sucesión generada por Newton-Raphson converge a α en forma cuadrática con $\beta = \frac{f''(\alpha)}{2f'(\alpha)}$, partiendo de un x_0 cercano a α y asumiendo $f''(\alpha) \neq 0$.

3.1.9. Método de Newton-Raphson para sistemas no lineales

Sea una función $f : \mathbb{R}^n \rightarrow \mathbb{R}^n$ no lineal, tal que:

$$f(x_1, \dots, x_n) = \begin{pmatrix} f_1(x_1, \dots, x_n) \\ f_2(x_1, \dots, x_n) \\ \vdots \\ f_n(x_1, \dots, x_n) \end{pmatrix}$$

con $f_1, \dots, f_n : \mathbb{R}^n \rightarrow \mathbb{R}$ no lineales. Queremos encontrar un punto $\vec{x} = (x_1, \dots, x_n) \in \mathbb{R}^n$ tal que $f(\vec{x}) = \vec{0}$

Sea $x_i \in \mathbb{R}^n \Rightarrow f(x) = f(x_i) + J_f(x_i)(x - x_i) + o(\|x - x_i\|)$

El método de Newton-Raphson para sistemas no lineales consiste en, despreciando el infinitésimo, anular $f(x)$ en un punto $x_{i+1} \in \mathbb{R}^n$. De esta forma iteramos con:

$$\begin{cases} J_f(x_i)\vec{\delta}_i = -f(x_i) \\ x_{i+1} = x_i + \vec{\delta}_i \end{cases}$$

Despejando $\vec{\delta}_i$ en cada paso, y calculando x_{i+1}

3.1.10. Método de la secante

Dados dos puntos $(x_{i-1}, f(x_{i-1})), (x_i, f(x_i))$, calculamos la ecuación de la recta que pasa por esos dos puntos. El punto de corte de esa recta con el eje de las abscisas lo llamamos x_{i+1} .

$$\Rightarrow x_{i+1} = x_i - f(x_i) \frac{(x_i - x_{i-1})}{f(x_i) - f(x_{i-1})}$$

El orden de convergencia del método de la secante es superlineal, con $p = \frac{1+\sqrt{5}}{2}$

3.1.11. Método de Steffensen

Sabemos que el método de Newton-Raphson es de la forma $\begin{cases} J_f(x_i) \vec{\delta}_i = -f(x_i) \\ x_{i+1} = x_i + \vec{\delta}_i \end{cases}$

El método de Steffensen consiste en aproximar las derivadas parciales de la siguiente forma:

$$\frac{\partial f_i}{\partial x_j}(x^{(k)}) \approx \frac{f_i(x^{(k)} + h_j \vec{e}_j) - f_i(x^{(k)})}{h_j}$$

donde $h_j = f_j(x^{(k)})$ y $\vec{e}_j = \begin{pmatrix} 0 \\ \vdots \\ 0 \\ 1 \\ 0 \\ \vdots \\ 0 \end{pmatrix}$

4. Mínimos cuadrados

4.1. Problema de mínimos cuadrados lineal

Dada $A \in \mathbb{R}^{m \times n}$, con $m > n$ y un vector $b \in \mathbb{R}^m$, el problema de mínimos cuadrados (PMC) lineal consiste en hallar $\min_{x \in \mathbb{R}^n} \|b - Ax\|_2$

Para resolver este problema veremos

- Descomposición QR de una matriz $A \in \mathbb{R}^{m \times n}$
- Descomposición en SVD de una matriz $A \in \mathbb{R}^{m \times n}$

4.1.1. Teorema: descomposición QR

Sea $A \in \mathbb{R}^{m \times n}$, con $m \geq n$ y $\text{rango}(A)=n$.

$\Rightarrow \exists Q \in \mathbb{R}^{m \times m}$ ortogonal, $R \in \mathbb{R}^{m \times n}$ triangular superior / $A = Q \times R$

4.1.2. Uso de descomposición QR para resolver el PMC lineal

Sea $A=Q \cdot R$, queremos hallar $\min_{x \in \mathbb{R}^n} \|b - Ax\|_2$

Propiedad: Las transformaciones ortogonales no cambian $\|\cdot\|_2$ de un vector

Consideremos:

$$\|b - Ax\|_2 = \|Q^T(b - Ax)\|_2 = \|Q^Tb - Q^T Ax\|_2 = \|Q^Tb - Q^T Q R x\|_2 = \|Q^Tb - R x\|_2$$

Sea $d = Q^Tb = \begin{pmatrix} d_1 \\ d_2 \end{pmatrix}$, con $d_1 \in \mathbb{R}^n, d_2 \in \mathbb{R}^{m-n}$

$$\Rightarrow \|d - R x\|_2 = \left\| \begin{pmatrix} d_1 - R_1 x \\ d_2 \end{pmatrix} \right\|_2, R = \begin{pmatrix} R_1 \\ 0 \end{pmatrix}$$

Si planteo $\min_{x \in \mathbb{R}^n} \left\| \begin{pmatrix} d_1 - R_1 x \\ d_2 \end{pmatrix} \right\|_2$, el mínimo se da cuando $d_1 = R_1 x$

Este sistema se resuelve fácilmente al ser triangular superior

4.1.3. Teorema: descomposición SVD

Sea $A \in \mathbb{R}^{m \times n}$, con $m \geq n$ y $\text{rango}(A)=r$.

\Rightarrow Existen tres matrices

- $U \in \mathbb{R}^{m \times m}$ ortogonal
- $\Sigma \in \mathbb{R}^{m \times n}, \Sigma = \begin{pmatrix} \Sigma_r & 0 \\ 0 & 0 \end{pmatrix}, \Sigma_r \in \mathbb{R}^{r \times r}$

$$\text{donde } \Sigma_r = \begin{pmatrix} \sigma_1 & 0 & \dots & 0 \\ 0 & \sigma_2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \sigma_r \end{pmatrix}$$

$$\blacksquare V \in \mathbb{R}^{n \times n}$$

que cumplen $A = U \times \Sigma \times V^T$

4.1.4. Uso de descomposición SVD para resolver el PMC lineal

Sean $b \in \mathbb{R}^m$ y $A \in \mathbb{R}^{m \times n}$ tal que $A = U \times \Sigma \times V^T$,

$$\text{con } \Sigma = \begin{pmatrix} \Sigma_r & 0 \\ 0 & 0 \end{pmatrix}, \Sigma_r = \begin{pmatrix} \sigma_1 & 0 & \dots & 0 \\ 0 & \sigma_2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \sigma_r \end{pmatrix}, \sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_r$$

$U \in \mathbb{R}^{m \times m}$ ortogonal, $V \in \mathbb{R}^{n \times n}$ ortogonal

\Rightarrow se cumple que la única solución al PMC es $x = V \times \begin{pmatrix} \Sigma_r^{-1} & 0 \\ 0 & 0 \end{pmatrix} \times U^T \times b$

4.2. Teorema

Sea S un subespacio vectorial de \mathbb{R}^m , y $b \in \mathbb{R}^m$ un cierto vector.

La proyección ortogonal de b sobre S , denotada por $y = P_s(b)$, queda caracterizada por cualquiera de las siguientes dos condiciones:

1. $y = P_s(b)$ es el único vector de S tal que $(b-y) \in S^\perp$
2. $y = P_s(b)$ es el vector de S que minimiza la distancia euclídea de b a S , o sea que $\|b - y\|_2 = \min_{s \in S} \|b - s\|_2$

4.3. Ecuaciones normales

Si \hat{x} es solución óptima al PMC, y $R(A) = [A^{(1)}, A^{(2)}, \dots, A^{(m)}]$, entonces:

$r = (b - A\hat{x}) \in (R(A))^\perp = \text{Ker}(A^T)$. Entonces:

$$A^T \cdot (b - A\hat{x}) = 0 \iff A^T \cdot A \cdot \hat{x} = A^T \cdot b$$

A la ecuación $A^T \cdot A \cdot \hat{x} = A^T \cdot b$ se le denomina **ecuaciones normales**

$$(A^T \cdot A) \in \mathbb{R}^{n \times n}; A^T \cdot b \in \mathbb{R}^n$$

Si $\text{rango}(A) = n$, entonces $\hat{x} = (A^T \cdot A)^{-1} \cdot A^T \cdot b$

4.4. Problema de mínimos cuadrados no lineal

Sea $P = \{(y_i, t_i), i = 1 \dots n\}$ un conjunto de datos que queremos ajustar a una función $f(x, t)$, x vector de parámetros, tal que se minimiza $\sum_{i=1}^n r_i(x)^2$, donde $r_i = y_i - f(x, t_i)$ es el i -ésimo residuo.

Se llama vector de residuos a $r(x) = (r_1(x), r_2(x), \dots, r_m(x)) \in \mathbb{R}^m$, $r_i(x) : \mathbb{R}^n \rightarrow \mathbb{R} \forall i = 1 \dots n$. El PMCNL se resuelve imponiendo $r(x)=0$ y solucionando esta ecuación.

4.4.1. Método de Gauss-Newton

Sea $p_k = x - x_k$

1. Resuelvo el PMC: $\min_{p_k \in \mathbb{R}^n} \|r(x_k) + J_r(x_k)p_k\|_2$

2. $x_{k+1} = x_k + p_k$

Donde $J_r(x) \in \mathbb{R}^{m \times n}$ tal que $(J_r(x))_{i,j} = \frac{\partial r_i(x)}{\partial x_j}$

5. Interpolación polinómica

De una cierta función $f(x)$ sabemos que en un conjunto de puntos $x_i, i=0 \dots n$, se cumple que $f(x_i) = y_i, \forall i \in \{0, n\}$

Queremos encontrar un polinomio $P(x) / P(x_i) = y_i, \forall i \in \{0 \dots n\}$

5.1. Ecuación de Vandermonde

Sea $P(x) = \alpha_0 + \alpha_1 x + \alpha_2 x^2 + \dots + \alpha_n x^n$

Imponemos que $P(x_i) = y_i$

$$\Rightarrow \begin{cases} P(x_0) = \alpha_0 + \alpha_1 x_0 + \alpha_2 x_0^2 + \dots + \alpha_n x_0^n = y_0 \\ P(x_1) = \alpha_0 + \alpha_1 x_1 + \alpha_2 x_1^2 + \dots + \alpha_n x_1^n = y_1 \\ \vdots \\ P(x_n) = \alpha_0 + \alpha_1 x_n + \alpha_2 x_n^2 + \dots + \alpha_n x_n^n = y_n \end{cases}$$

Este sistema lo podemos escribir de la siguiente manera:

$$\begin{pmatrix} 1 & x_0 & x_0^2 & \dots & x_0^n \\ 1 & x_1 & x_1^2 & \dots & x_1^n \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & x_n & x_n^2 & \dots & x_n^n \end{pmatrix} \begin{pmatrix} \alpha_0 \\ \alpha_1 \\ \vdots \\ \alpha_n \end{pmatrix} = \begin{pmatrix} y_0 \\ y_1 \\ \vdots \\ y_n \end{pmatrix}$$

Matricialmente entonces tenemos el sistema lineal $X\vec{\alpha} = \vec{y}$ (ecuación de Vandermonde)

5.2. Estudio del error en la interpolación polinómica

Consideramos la diferencia $E(x) = f(x) - P(x)$

En general se cumple que $E(x) = f(x) - P(x) = (x - x_0)(x - x_1) \dots (x - x_n) \cdot g(x)$

Fijado $x \in D$, siendo D el dominio de f , definimos la siguiente función auxiliar:

$W(t) = f(t) - P(t) - (t - x_0)(t - x_1) \dots (t - x_n) \cdot g(x)$

Sabemos que $W(t)$ se anula en al menos $n+1$ puntos:

- En los $x_i, i \in \{0 \dots n\}$
- En $t=x$

Asumiendo $W(t)$ derivable hasta orden $n+1$, se cumple que

- $W'(t)$ tiene $n+1$ raíces
- $W''(t)$ tiene n raíces
- \vdots
- $W^{(n+1)}(t)$ tiene una raíz $\lambda(x)$

Se puede verificar que $W^{(n+1)}(t) = f^{(n+1)}(t) - (n+1)! \cdot g(x)$

Sabemos que $W^{(n+1)}(\lambda(x)) = 0$

$$0 = f^{(n+1)}(\lambda(x)) - (n+1)! \cdot g(x) \Rightarrow g(x) = \frac{f^{(n+1)}(\lambda(x))}{(n+1)!}$$

De esta forma tenemos que:

$$E(x) = f(x) - P(x) = \frac{f^{(n+1)}(\lambda(x))}{(n+1)!} \cdot \prod_{i=0}^n (x - x_i)$$

Si asumimos que $x, x_i, \lambda(x) \in [a, b]$ se cumple:

$$|f(x) - P(x)| < \frac{\max\{f^{(n+1)}(x); x \in [a, b]\} \cdot (b-a)^{n+1}}{(n+1)!}$$

5.3. Interpolación de Lagrange

Tengo $n+1$ puntos tales que $f(x_i) = y_i, \forall i \in \{0 \dots n\}$

Se define el j -ésimo polinomio de Lagrange como:

$$l_j(x) = \frac{(x - x_0)(x - x_1) \dots (x - x_{j-1})(x - x_{j+1}) \dots (x - x_n)}{(x_j - x_0)(x_j - x_1) \dots (x_j - x_{j-1})(x_j - x_{j+1}) \dots (x_j - x_n)}$$

Se puede ver que $l_j(x_i) = \begin{cases} 1 & \text{si } i = j \\ 0 & \text{si } i \neq j \end{cases}$

El interpolante por los $n+1$ puntos es $L(x) = \sum_{i=0}^n f(x_i) l_i(x)$

5.4. Interpolación de Hermite

Queremos encontrar $P(x)$ tal que $\begin{cases} P(x_i) = f(x_i) = y_i \\ P'(x_i) = f'(x_i) = \bar{y}_i \end{cases} \quad \forall i \in \{0 \dots n\}$ ($2n+2$ condiciones)

La interpolación de Hermite es de la forma:

$$H_{2n+1}(x) = \sum_{i=0}^n y_i h_i(x) + \sum_{i=0}^n \bar{y}_i \tilde{h}_i(x)$$

donde

$$h_i(x_j) = \begin{cases} 1 & \text{si } i = j \\ 0 & \text{si no} \end{cases}$$

$$h'_i(x_j) = 0 \quad \forall j \in \{0 \dots n\}$$

$$\tilde{h}_i(x_j) = 0 \quad \forall j \in \{0 \dots n\}$$

$$\tilde{h}'_i(x_j) = \begin{cases} 1 & \text{si } i = j \\ 0 & \text{si no} \end{cases}$$

En la práctica

$$h_i(x) = (1 - 2l'_i(x)(x - x_i))(l_i(x))^2$$

$$\tilde{h}_i(x) = (x - x_i)(l_i(x))^2$$

donde $l_i(x)$ es el i -ésimo polinomio de Lagrange definido anteriormente

5.5. Interpolación de Newton

Objetivo: encontrar un polinomio interpolante de $f(x)$ en los puntos $\{x_i, f(x_i)\}$, de forma que sea relativamente fácil incorporar nuevos puntos.

Se generan polinomios de la forma $P_k(x) = P_{k-1}(x) + q_k(x)$, tal que $P_k(x_i) = f(x_i) \quad \forall i \in \{0 \dots k\}$, $\forall k \in \{0 \dots n\}$

De esta relación inferimos que $q_k(x) = P_k(x) - P_{k-1}(x) = 0 \quad \forall i \in \{0 \dots k-1\}$

Queremos que $q_k(x_k) \neq 0$. Se busca un q_k con esta estructura $q_k(x) = a_k(x - x_0)(x - x_1) \dots (x - x_{k-1})$

$$P_k(x_k) = f(x_k) = P_{k-1}(x_k) + q_k(x_k)$$

$$\text{Tenemos que } f(x_k) = P_{k-1}(x_k) + a_k \prod_{i=0}^{k-1} (x_k - x_i)$$

$$\Rightarrow a_k = \frac{f(x_k) - P_{k-1}(x_k)}{\prod_{i=0}^{k-1} (x_k - x_i)}$$

Con lo cual

$$P_k(x) = P_{k-1}(x) + \frac{f(x_k) - P_{k-1}(x_k)}{\prod_{i=0}^{k-1} (x_k - x_i)} \times \prod_{i=0}^{k-1} (x - x_i)$$

5.6. Interpolación lineal a trozos

Sea $P = \{(x_i, y_i), i \in \{0 \dots n\}\}$ $x_0 < x_1 < \dots < x_n$

Veremos un polinomio de grado 1, $L_i(x)$, tal que $\begin{cases} L_i(x_i) = y_i \\ L_i(x_{i+1}) = y_{i+1} \end{cases}$

Voy interpolando en el intervalo $[x_i, x_{i+1}]$:

$$L_i(x) = y_i \frac{x - x_{i+1}}{x_i - x_{i+1}} + y_{i+1} \frac{x - x_i}{x_{i+1} - x_i}$$

5.7. Curvas de Bezier

Dado un conjunto de valores (denominados pesos) $\{p_i\}$, $i \in \{0 \dots n\}$, se define el polinomio de Bezier asociado a estos pesos y trabajando sobre un intervalo $[a, b]$ al polinomio dado por

$$P_n(x) = \sum_{i=0}^n p_i B_i^n(x)$$

donde $B_i^n(x)$ es el i -ésimo polinomio de Bernstein, y viene dado en forma explícita por

$$B_i^n(x) = \frac{n!}{(n-i)!i!} \times \frac{(b-x)^{n-1}(x-a)^i}{(b-a)^n}$$

En forma recursiva, los polinomios de Bernstein son:

- $B_0^0(x) = 1$
- $B_{-1}^n(x) = 0$
- $B_{n+1}^n(x) = 0$
- $B_i^n(x) = \frac{(b-x)B_{i-1}^n(x) + (x-a)B_{i-1}^{n-1}(x)}{b-a}$

Una vez fijados los $\{p_i\}$, se le llama puntos de control a los puntos de \mathbb{R}^2 dados por $\{(t_i, p_i)\}$, $i \in \{0 \dots n\}$ tales que $t_i = x_0 + i \frac{(x_n - x_0)}{n}$

5.7.1. Propiedad

El polinomio $P_n(x)$ anteriormente definido queda dentro del convexo determinado por la poligonal de Bezier, que es la poligonal que une los puntos de control y (t_0, p_0) con (t_n, p_n)

5.8. Curva paramétrica de Bezier

Teniendo como dato un conjunto de puntos de \mathbb{R}^2 , $P = \{\vec{p}_i = (x_i, y_i); i \in \{0 \dots n\}\}$, se define paramétrica de Bezier como $\vec{P}_n(t) = \sum_{i=0}^n \vec{P}_i B_i^n(t)$

6. Ecuaciones diferenciales ordinarias

Veremos métodos que resuelven numéricamente sistemas del tipo
$$\begin{cases} y' = f(x, y) \\ y(x_0) = c \end{cases} \quad x \in [a, b]$$

6.1. Método de Euler

$$\text{Dada la EDO } \begin{cases} y' = f(x, y) \\ y(x_0) = c \end{cases} \quad x \in [a, b]$$

El método de Euler genera una sucesión $\{y_n\}_{n \geq 0}$ que busca satisfacer $y_n \approx y(x_n)$, donde $y(x)$ es la solución exacta de la EDO, y los $\{x_n\}_{n \geq 0}$ son de la forma $x_n = x_0 + b \frac{n-a}{N} = x_0 + nh$ para un N dado (en general $x_0=a$). El valor de y_{n+1} se estima aproximando $y'(x_n)$ mediante $\frac{y_{n+1}-y_n}{h} \approx y'(x_n) = f(x_n, y(x_n)) \approx f(x_n, y_n)$

El método de Euler queda entonces definido de la siguiente forma:

$$\begin{cases} y_{n+1} = y_n + h \cdot f(x_n, y_n) \\ y_0 = c \end{cases}$$

6.2. Notación

Denotaremos por $y(x_{n+1}, x_0, c)$ a la solución exacta de la EDO $\begin{cases} y' = f(x, y) \\ y(x_0) = c \end{cases}$, evaluada en el punto x_{n+1}

6.3. Definición

- Se define error global en el punto x_{n+1} a la diferencia $E_{n+1} = y(x_{n+1}, x_0, c) - y_{n+1}$
- Se define error local en el punto x_{n+1} a la diferencia $\varepsilon_{n+1} = y(x_{n+1}, x_n, y_n) - y_{n+1}$, donde $y(x_{n+1}, x_n, y_n)$ es la solución exacta de la EDO $\begin{cases} y' = f(x, y) \\ y(x_n) = y_n \end{cases}$, evaluada en el punto x_{n+1}

Nota: Asumiremos de aquí en más que se satisface la siguiente propiedad:

$$l \leq \frac{\partial f}{\partial y}(x, y) \leq L, \quad \forall (x, y) \in D$$

6.4. Teorema

Si los errores locales ε_n , $n=0,1,2,\dots$ satisfacen que $\varepsilon_n \leq \varepsilon$ (cierto epsilon), entonces los errores globales E_n cumplen las siguientes desigualdades:

1. $|y(x_n) - y_n| \leq \varepsilon \cdot \frac{e^{L \cdot n \cdot h} - 1}{e^{L \cdot h} - 1}$
2. $|y(x_n) - y_n| \leq \frac{\varepsilon}{h} \cdot \frac{e^{L \cdot n \cdot h} - 1}{L} \leq c \cdot \frac{\varepsilon}{h}$
3. Si $L=0 \Rightarrow |y(x_n) - y_n| \leq n \cdot \varepsilon$

6.5. Estabilidad numérica

Sea $\{y_n\}_{n \geq 0}$ la solución numérica de una EDO.

Sea $\{\bar{y}_n\}_{n \geq 0}$ la representación en punto flotante de los $\{y_n\}_{n \geq 0}$

Sea $e_n = |y_n - \bar{y}_n|$, $\forall n \geq 0$

Diremos que la solución es numéricamente estable si los $\{e_n\}_{n \geq 0}$ permanecen acotados por un cierto ϵ

6.6. Problema Test

Se denomina problema test al problema PT:
$$\begin{cases} y' = q \cdot y \\ y(0) = 1 \end{cases}$$

Observar que la solución exacta del PT es $y(x) = e^{q \cdot x}$

6.7. Región de estabilidad asociada a un método

Dado un cierto método para resolver una EDO, la región de estabilidad asociada al método son los puntos $z=h \cdot q$ del plano complejo tales que los $\{y_n\}_{n \geq 0}$ solución de aplicar el método al problema test están acotados

6.8. Método del trapecio

El método de Euler aproxima el área debajo de la curva por:

$$\int_{x_n}^{x_{n+1}} f(x, y) dx \approx h \cdot f(x_n, y_n)$$

Otra forma es tomar el trapecio:

$$\int_{x_n}^{x_{n+1}} f(x, y) dx \approx \frac{h}{2} (f(x_{n+1}, y_{n+1}) + f(x_n, y_n))$$

El método del trapecio viene dado entonces por $y_{n+1} = y_n + \frac{h}{2} (f(x_{n+1}, y_{n+1}) + f(x_n, y_n))$

Observación: este es un método implícito. Si f es no lineal, entonces no puedo despejar y_{n+1} en función del resto de los parámetros.

Lo que se hace es:

$$\begin{cases} y_{n+1}^{(k+1)} = \frac{h}{2} \cdot f(x_{n+1}, y_{n+1}^{(k)}) + \mu_n, \text{ con } \mu_n = y_n + \frac{h}{2} \cdot f(x_n, y_n) \\ y_{n+1}^{(0)} = y_n + h \cdot f(x_n, y_n) \end{cases}$$

Partiendo de $y_{n+1}^{(0)}$, itero con la primera ecuación.

6.9. Métodos consistentes

Se dice que un método es consistente si $\lim_{h \rightarrow 0} \frac{1}{h} \cdot \max\{|\varepsilon_T(h)|\} = 0$

($\varepsilon_T(h) = \tilde{y}(x_{n+1}) - y_{n+1}$)

Diremos que un método es convergente si es consistente y estable numéricamente

6.10. Método de Euler hacia atrás

Este método surge de tomar $\frac{y_{n+1}-y_n}{h} \approx y'(x_{n+1}) = f(x_{n+1}, y(x_{n+1})) \approx f(x_{n+1}, y_{n+1})$
Imponiendo esta igualdad obtenemos:

$$\begin{cases} y_{n+1} = y_n + h \cdot f(x_{n+1}, y_{n+1}) \\ y_0 = c \end{cases}$$

6.11. Método de punto medio

Este método surge de tomar la ecuación $y'(x_n) = \frac{y(x_{n+1})-y(x_{n-1}))}{2h} + O(h^2)$
De esta ecuación podemos inferir el siguiente estimador:

$$\begin{cases} y_{n+1} = y_{n-1} + 2h \cdot f(x_n, y_n) \\ y_1 = y_0 + h \cdot f(x_0, y_0) \text{ (Euler)} \\ y_0 = c \end{cases}$$

6.12. Métodos de Runge-Kutta

Dada la EDO $\begin{cases} y' = f(x, y) \\ y(x_0) = c \end{cases} \quad x \in [a, b]$

Consideramos la siguiente aproximación:

$y(x_{n+1}) = y(x_n) + h \cdot y'(\zeta_n)$, donde $\zeta_n = x_n + \theta_n \cdot h$ y $\theta_n \in (0, 1)$

Tomando $\theta_n = \frac{1}{2}$, tenemos que $y(x_{n+1}) = y(x_n) + h \cdot f(x_n + \frac{h}{2}, y(x_n + \frac{h}{2}))$

Aproximamos $y(x_n + \frac{h}{2})$ mediante Euler hacia adelante con paso $\frac{h}{2}$:

$y(x_n + \frac{h}{2}) \approx y_n + \frac{h}{2} \cdot f(x_n, y_n)$

Un método de Runge-Kutta viene dado por

$$y_{n+1} = y_n + h \cdot f(x_n + \frac{h}{2}, y_n + \frac{h}{2} \cdot f(x_n, y_n))$$

Otro método de Runge-Kutta surge de aproximar $y'(x_n + \frac{h}{2}) \approx \frac{y'(x_n) + y'(x_{n+1}))}{2} \approx \frac{f(x_n, y_n) + f(x_{n+1}, y_n + h \cdot f(x_n, y_n))}{2}$

El segundo método viene dado entonces por

$$y_{n+1} = y_n + \frac{h}{2}(f(x_n, y_n) + f(x_{n+1}, y_n + h \cdot f(x_n, y_n)))$$