0 INTRODUCCION

0.1 Métodos numéricos computacionales

El objetivo de la asignatura es dar una introducción a los métodos numéricos, esto es, la resolución de problemas mediante técnicas numéricas como alternativa a las técnicas analíticas (exactas).

La mayoría de problemas del mundo real, requieren para su solución varias fases, en todas ellas se introducen errores.

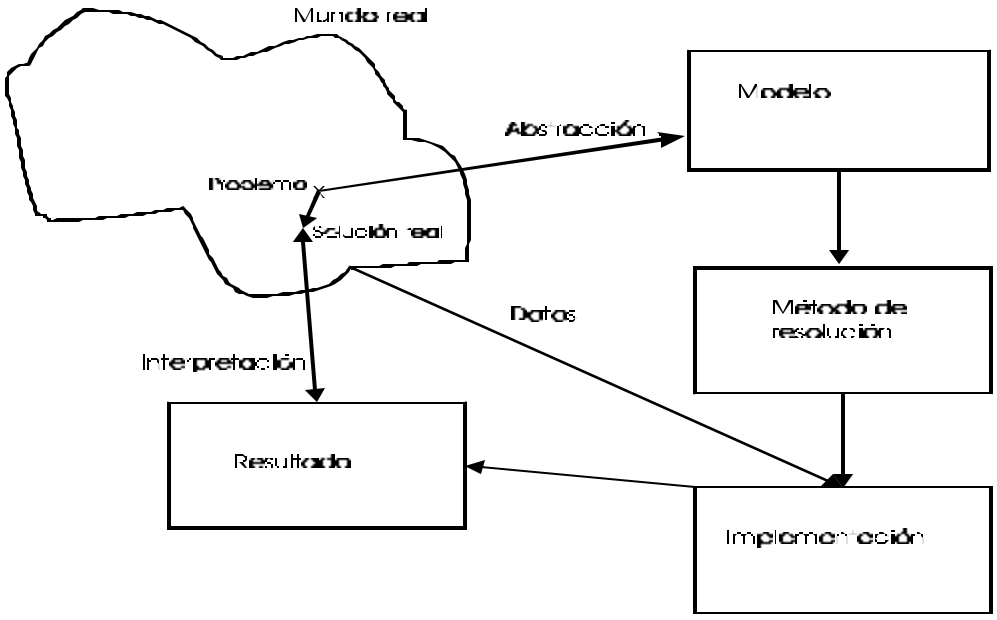

Figura 1: Fases de la resolución de un problema

- Modelo: Se realiza una abstracción desde el problema real considerando solo las variables más importantes, las relaciones entre ellas, etc,...

  El modelo será una representación simplificada del mundo real y por tanto contendrá errores. Por ejemplo en la construcción de un puente se considerará el grosor, coeficiente de dilatación de las vigas, etc,..., pro no se considera la atracción lunar, el peso y el grado de reflectancia de color de la capa de pintura, etc,...

  Un buen modelo no debe ser exhaustivo y solo debe considerar las variables importantes.

- Método de solución: Se busca un algoritmo que resuelva el modelo de forma general. Los algoritmos pueden ser métodos iterativos que necesiten infinitos pasos para llegar a la solución exacta, por lo que pueden tener errores.

- Implementación: Al incorporar al algoritmo anterior los datos concretos del problema (desde el mundo real, se obtendrá un resultado. Los errores pueden ser introducidos tanto en la introducción de datos, como en el cálculo con ellos.

- Resultado: El resultado deberá ser comparado con el mundo real. Para validar el modelo debemos comparar si los resultados obtenidos coinciden con los reales.

Los métodos numericos tratan con el problema de operar con datos (implementación), pero también con el de realizar algoritmos que produzcan poco error al trabajar con ellos.

0.2 Sistema aritmético de punto flotante. Errores de representación

La representación usual de un número real en cálculo científico es en punto flotante. En esta representación el número se descompone en mantisa (m) y exponente (exp), donde la mantisa se representa en módulo y signo (RMS) y el exponente en exceso 2^{t-1}.

Los n bits dedicados a almacenar el número real se descompone en s para representar la mantisa (uno de ellos para el signo) y t para la mantisa (otro para el signo).

Normalización: Para ahorrar espacio el número se normaliza al almacenarlo, esto es la mantisa se representa de forma que $\frac{1}{\beta} \leq m < 1$ (donde β es la base de numeración).

0.2.1 Características de la representación en flotante

- Una representación en flotante queda determinada por 3 parámetros β, s y t. itemA cada $x \in \mathcal{R}$ le corresponde una única representación $x' \in \mathcal{F}(\beta, s, t)$.
- A cada $x' \in \mathcal{F}(\beta, s, t)$, le corresponde un intervalo $I(x') \subset \mathcal{R}$, que se representa por él.
- Existen números reales no representables (OVERFLOW).
- Existe un intervalo de centro 0 que se representa por 0 (UNDERFLOW).

0.2.2 Errores relacionados

Al operar con numeros existen diversas fuentes de errores:

- **Error de redondeo:** Es el error que se comete al representar un número en puntop flotante. Si el numero real x es representado mediante x', entonces el error absoluto de redondeo es $\Delta_x = |x - x'|$ y el relativo es $\delta_x = \frac{|\Delta_x}{|x|}$

  El error relativo de redondeo se encuentra acotado superiormente por el epsilon de la máquina ϵ.

- **Errores aritméticos:** Al operar en la ALU se introducen errores, por ejemplo al multiplicar dos números de forma exacta con 16 dígitos significativos, resulta usualmente otro con 32 que no podrá ser representado en el sistema, pero ya la ALU habrá cometido errores al realizar desplazamientos y sumas.

  Excepto en el caso de **sustracciones cancelativas**, los errores aritméticos de las sumas, restas, multiplicaciones y divisiones se encuentran acotados por 4ϵ.

  La sustracción cancelativa es una de las mayores fuentes de error y debe evitarse o cometerse en los últimos cálculos, a ser posible.

- **Error propagado:** Es el error que existiendo en los datos se propaga al resultado. Por ejemplo en la suma $x = x' + \Delta_x$ y $y = y' + \Delta_y$, resulta que $x + y = x' + y' + \Delta_x + \Delta_y$ y por tanto $\Delta_{x+y} = \Delta_x + \Delta_y$ luego $|\Delta_{x+y} \leq |\Delta_x| + |\Delta_y|$

  Si $y = f(x_1, x_2, \ldots, x_n)$ y x_i tienen errores entonces: $|\Delta y| \leq \sum_{i=1}^{n} |\frac{\partial y}{\partial x_i}| \, |\Delta_{x_i}\|$

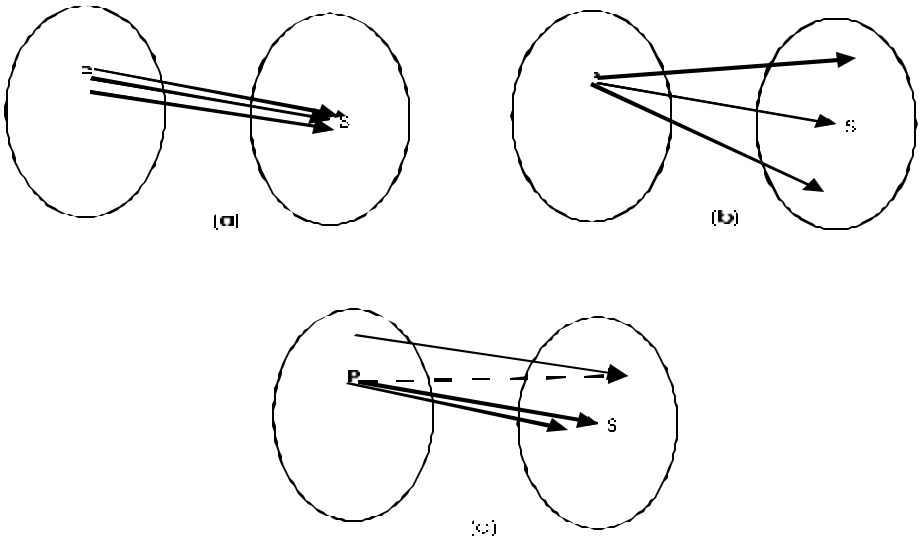

Figura 2: (a) *Problema estable:* Problemas parecidas tienen soluciones exactas parecidas. (b) *Problema inestable:* Problemas parecidos tienen soluciones muy diferentes. (c) *Método inestable:* A pesar de ser estable el problema, la solución computada es muy diferente a la verdadera y solo problemas muy diferentes del propuesto poseen esa solución.

0.3 Inestabilidad de problemas y métodos numéricos

Problema bien condicionado o estable: Si el problema real tiene una solución exacta y además problemas similares producen soluciones cercanas. Ejemplos: Al variar poco una de las resistencias de un circuito electrico las intensidades varian poco, dilatación de una viga, fuerza y elongación de un muelle, lanzamiento de proyectiles, intensidad de corriente y calor desprendido.

Problema mal condicionado o inestable: Si problemas similares producen resultados muy diferentes. Ejemplos: El lanzamiento de un dado, diodo en el punto de corte, interruptores (pequeñas fuerzas producen grandes efectos), encontrar el punto de corte de dos rectas casi paralelas. Un método numérico será un proceso de cálculo que lleve a la resolución de un problema, evidentemente los errores como hemos visto ocurren en todas las fases de dicha resolución y pueden llevar a proporcionar soluciones no válidas.

Método mal condicionado o inestable: Puede ocurrir que un problema esté bien condicionado, pero estamos usando un método de cálculo que no lo está. Ejemplo: En la ecuación de segundo grado $ax^2 + bx + c = 0$ cuando $4ac \ll b^2$ resulta que la formula $x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$, produce al calcular una de las raices una sustracción cancelativa y resulta inestable. Sin embargo si calculamos primero $x_1 = \frac{-b - sgn(b)\sqrt{b^2 - 4ac}}{2a}$ y la otra raiz mediante $x_2 = \frac{c}{ax_1}$, el método resultante es estable.

1 ALGEBRA LINEAL NUMÉRICA

1.1 Sistemas de ecuaciones lineales

Un sistema lineal puede expresarse por:

$$\begin{array}{rl} a_{1,1}x_1 + a_{1,2}x_2 + a_{1,3}x_3 + \ldots + a_{1,n}x_n & = b_1 \\ a_{2,1}x_1 + a_{2,2}x_2 + a_{2,3}x_3 + \ldots + a_{2,n}x_n & = b_2 \\ a_{3,1}x_1 + a_{3,2}x_2 + a_{3,3}x_3 + \ldots + a_{3,n}x_n & = b_3 \\ \ldots & \ldots \\ a_{m,1}x_1 + a_{m,2}x_2 + a_{m,3}x_3 + \ldots + a_{m,n}x_n & = b_m \end{array}$$

que se expresa matricialmente por $Ax = b$, donde $A = (a_{ij})$, $b = (b_i)$ y $x = (x_i)$.

1.1.1 Conocimientos previos

- Clasificación de los sistemas: Según el número de soluciones de un sistema lineal, éstos pueden ser:
  - Sistema compatible determinado si tiene una única solución (SCD).
  - Sistema compatible indeterminado si tiene infinitas soluciones (SCI).
  - Sistema imposible o incompatible si no tiene soluciones (SI).
- Rango de una matriz: Número máximo de vectores filas o columnas linealmente independiente, o también el orden del mayor menor distinto de cero.
- Regla de Cramer: Si A es cuadrada y $|A| \neq 0$, se trata de un SCD, y su solución es: $x_i = \frac{\Delta_i}{|A|}$
- Teorema de Rouché-Frobernius: Si el rango de A, coincide con el de la ampliada A^+, el sistema es compatible, en caso contrario es SI. Si además coincide con el número de incógnitas, entonces será SCD, en caso contrario será SCI.
- Cálculo de la matriz inversa. Matrices inversibles. Una matriz cuadrada es inversible (posee matriz inversa) si y solo si $|A| \neq 0$, calculándose $A^{-1} = \frac{Adj(A^t)}{|A|}$. A las matrices no inversibles se les denomina tambien *matrices singulares*.
- Matriz diagonal: Matriz que solo tiene componentes diferentes de cero en la diagonal principal.
- Matriz triangular superior: Matriz con las componentes por debajo de la diagonal principal iguales a cero. Análogamente una matriz será *triangular inferior* si sus componentes sobre la diagonal principal son todos ceros.
- Matriz traspuesta: Dada una matriz A, se denomina traspuesta de A (A') a una matriz que tiene por filas las columnas de A.
- Matriz simétrica: Una matriz lo es si verifica ($A = A'$).
- Matriz antisimétrica: Una matriz lo es si cumple ($A = -A'$).

- Matriz simétrica definida positiva: Una matriz simétrica lo es si y solo si $\forall x \neq 0 \Leftrightarrow x'Ax > 0$. Es equivalente a que todos sus autovalores sean positivos y también es equivalente a que los menores principales sucesivos sean todos positivos.
- Matriz simétrica semidefinida positiva: Una matriz simétrica lo es si y solo si $\forall x \neq 0 \Leftrightarrow x'Ax \geq 0$. Es equivalente a que todos sus autovalores sean no negativos y también es equivalente a que los menores principales sucesivos sean todos no negativos.
- Valores propios o autovalores de una matriz: Son números λ_i, (reales o complejos), que verifican $A\bar{x} = \lambda\bar{x}$. Los vectores $\bar{x}$ que verifican la ecuación anterior se llaman **vectores propios o autovectores**.
- Matrices equivalentes: Dos matrices se llaman **equivalentes** si tienen los mismos autovalores.
- Propiedad: Una matriz simétrica tiene todos sus autovalores reales.

1.1.2 Solución de Ax = b por el método de Cramer

El método de Cramer es extremadamente ineficiente para la resolución del sistema.

Si A es cuadrada y $|A| \neq 0$, entonces existe la inversa A^{-1} y la solución del sistema es $x = A^{-1}b$, que puede ser calculada mediante $\mathbf{x_i} = \frac{\Delta_i}{\Delta}$.

Su cálculo necesita $n(n+1)! - 1$ operaciones desglosadas en n divisiones, $(n+1)(n!-1)$ sumas y (n+1)n!(n-1) productos.

1.1.3 Normas de vectores y de matrices

Se llama *norma vectorial* a una aplicación $\| \ \|$ del espacio vectorial V, sobre su cuerpo (usualmente $\mathcal{R}$) que verifica:

1) $\|x\| \geq 0$ para todo $x \in V$
2) $\|x\| = 0$, si y solo si $x = (0, 0, \ldots, 0)^t$
3) $\|\alpha x\| = |\alpha|\|x\|, \quad \forall x \in V, \alpha \in \mathcal{R}$
4) $\|x + y\| \leq \|x\| + \|y\| \quad \forall x, y \in V$.

La norma de un vector es una medida del tamaño del vector y puede considerarse como la distancia del punto x, al origen. Como extensión la distancia entre dos vectores podrá considerarse como la norma del vector diferencia entre ellos.

Ejemplos de normas vectoriales:

Norma k

Dado un vector $\bar{x} = (x_1, x_2, x_3, \ldots, x_n)^t$, se define su norma k mediante: $\|x\|_k = \sqrt[k]{(\sum_{i=1}^{n} |x_i|^k)}$

Son especialmente importantes la norma 2 y la norma 1 de expresiones:

$$\|x\|_1 = \sum_{i=1}^{n} |x_i|, \qquad \|x\|_2 = \sqrt{\sum_{i=1}^{n} (x_i)^2}$$

A la norma 2 se le llama también **norma euclídea**.

Norma infinito

Dado un vector $\bar{x} = (x_1, x_2, x_3, \dots, x_n)^t$, se define su norma infinito (o también llamada norma del máximo), mediante: $\|x\|_\infty = \max_i |x_i|$

NORMA MATRICIAL

Es una aplicación de las matrices cuadradas (orden n) en $\mathcal{R}$, tal que se verifican las mismas propiedades que para la norma vectorial y además:
5) $\|AB\| \leq \|A\| \, \|B\|$

Concepto de norma matricial inducida:
Para toda norma vectorial, existe una norma matricial natural o inducida, definida mediante:

$$\|A\|_N = \max_{\|\bar{x}\|=1} \|A\bar{x}\|$$

Las normas matriciales inducidas por una norma vectorial verifican: $\|I\| = 1$.

Ejemplos de normas matriciales:

Norma 1

$\|A\|_1 = \max_j \sum_i |a_{ij}|$ (Es inducida por la norma 1 vectorial).

Norma infinito

$\|A\|_\infty = \max_i \sum_j |a_{ij}|$ (Inducida por la norma infinito).

Norma 2 o espectral:

$\|A\|_2 = (\rho(A^t A))^{1/2}$, donde ρ es el radio espectral. Es inducida por la norma 2 vectorial.
Donde el **radio espectral** se define como el módulo del autovalor con módulo mayor: $\rho(B) = \max|\lambda_i|$ donde λ_i, son los valores propios de B.

Norma euclídea o de Frobernius:

$\|A\|_E = \left(\sum_i \sum_j |a_{ij}|^2\right)^{1/2}$
No es una norma inducida, de hecho: $\|A\| = \sqrt{n}$, donde n es el orden de la matriz.

1.1.4 Propiedades de las normas

- Toda norma matricial verifica: $\rho(A) = \inf \|A\|$, es decir:

  1) $\rho(A) \leq \|A\|$ 2) $\forall \epsilon > 0$ existe una norma $\|A\| < \rho(A) + \epsilon$

- La sucesión de las potencias de una matriz $A^n \to 0$ si y solo si $\rho(A) < 1$.

1.1.5 Definiciones

- **Definición de límite de una sucesión matricial:** Una sucesión de matrices $\{A_n\}$ tiene por límite otra matriz A ($\{A_n\} \to A$), si y solo si $\lim_{n\to\infty} \|A_n - A\| = 0$.
- **Matriz de diagonal dominante por filas:** Una matriz es de diagonal dominante por filas si y solo si: $\forall i,\ |a_{i,i}| > \sum_{j\neq i} |a_{i,j}|$
- **Matriz de diagonal dominante por columnas:** Una matriz es de diagonal dominante por columnas si y solo si: $\forall i,\ |a_{i,i}| > \sum_{j\neq i} |a_{j,i}|$

1.2 Métodos directos para la solución de sistemas lineales

Se denominan **métodos directos** a aquellos que proporcionarían la solución exacta si trabajamos en aritmética exacta, como alternativa los **métodos iterativos** pretenden en cada iteración producir un vector más cercano al vector solución.

1.2.1 Casos especiales

Cuando la matriz tiene alguna propiedad puede servir para resolver el sistema de forma simple:

- Cuando la matriz A es diagonal

$$\left.\begin{array}{rcl} a_{1,1}x_1 & & = b_1 \\ & a_{2,2}x_1 & = b_2 \\ & a_{3,3}x_1 & = b_3 \\ & \ddots & \dots \\ & a_{n,n}x_n & = b_n \end{array}\right\} \quad \Rightarrow \quad x_i = \frac{b_i}{a_{i,i}} \quad \forall i$$

  La solución del sistema se obtiene en n operaciones.

  El determinante de A vale: $|A| = \prod_{i=1}^{n} a_{i,i}$

- Cuando la matriz A es triangular superior

  Se resuelve mediante sustitución regresiva en n^2 operaciones.

$$\left.\begin{array}{rrrrrr} a_{1,1}x_1 & +a_{1,2}x^2 & +a_{1,3}x_3 & +\dots & +a_{1,n}x_n & = b_1 \\ & +a_{2,2}x^2 & +a_{2,3}x_3 & +\dots & +a_{2,n}x_n & = b_2 \\ & & +a_{3,3}x_3 & +\dots & +a_{3,n}x_n & = b_3 \\ & & & \ddots & & \dots \\ & & & & a_{n,n}x_n & = b_n \end{array}\right\} \quad \Rightarrow$$

  $x_i = \frac{b_i - \sum_{k=i+1}^{n} a_{i,k}x_k}{a_{i,i}} \quad \forall i.$

  El determinante $|A| = \prod_{i=1}^{n} a_{i,i}$

- Cuando A es triangular inferior

  Se resuelve mediante sustitución progresiva en n^2 operaciones.

$$\left.\begin{array}{lllllll} a_{1,1}x_1 & & & & & & = b_1 \\ a_{2,1}x_1 & +a_{2,2}x^2 & & & & & = b_2 \\ a_{3,1}x_1 & +a_{3,2}x^2 & +a_{3,3}x_3 & & & & = b_3 \\ & \vdots & \vdots & & \ddots & & \vdots \\ a_{n,1}x_1 & +a_{n,2}x^2 & +a_{n,3}x_3 & +\ldots & & +a_{n,n}x_n & = b_n \end{array}\right\} \Rightarrow$$

  $x_i = \frac{b_i - \sum_{k=1}^{k-1} a_{i,k}x_k}{a_{i,i}} \quad \forall i.$

  El determinante $|A| = \prod_{i=1}^{n} a_{i,i}$

1.2.2 Método de reducción de Gauss

Dado el sistema Ax=b, con A no singular, se tratará de encontrar un sistema equivalente $Rx = b'$ con R triangular superior. Teóricamente consiste en encontrar una matriz S tal que SA=R, por lo que $SAx = Sb = b' \Rightarrow Rx = b'$.

En la práctica consiste en hacer operaciones por filas que hagan la matriz A triangular superior. Las operaciones por filas empleadas hacen ceros en una columna por debajo de la diagonal, pero a veces también es necesario el intercambio de filas.

NOTA: El intercambio de filas no es computacionalmente necesaria, pues basta controlarlo mediante punteros o registro de índices, debiendo el algoritmo llevar la cuenta de los intercambios.

La condición necesaria y suficiente para que no sea necesario intercambiar filas es que todos los menores principales de orden creciente sean distintos de cero, sin considerar el último $|A|$.

Esta propiedad es verificada si:

1) A es definida positiva.
2) A es de diagonal dominante por filas.
3) A es de diagonal dominante por columnas.

El $|A|$, podrá calcularse como el producto de los términos diagonales una vez diagonalizada, multiplicada por $(-1)^t$, donde t=número de intercambios.

Las operaciones necesarias serán: $\frac{4n^3+9n^2-7n}{6}$ (menor que n^3 si $n > 1$) incluyendo las n^2 operaciones necesarias para la sustitución regresiva.

Una vez encontrado el pivote en el paso k-ésimo (se supone situado en $a_{k,k}$, se hacen ceros en la columna k-ésima mediante la fórmula:

$$a_{i,j}^{k+1} = a_{i,j}^{k} - \frac{a_{i,k}^{k}}{a_{k,k}^{k}} a_{k,j} \quad \left\{ \begin{array}{l} i = k+1, k+2, \ldots, n \\ j = k+1, k+2, \ldots, n, n+1 \end{array} \right\}$$

donde la columna n+1 inicial es el vector b.

Técnicas de pivotaje:

Si en el paso k-ésimo, el término diagonal $a_{k,k} = 0$ tenemos que intercambiar filas, pero si $a_{k,k} \approx 0$, entonces lo que ocurre es que se incrementan los errores de cálculo al dividir por un número pequeño. Para evitarlo se recurre al pivotaje parcial o total.

Parcial:

Tomamos como pivote del paso j, el $a_{i,j}$ de mayor valor absoluto, para i variando entre j y n. Es decir en el paso j-ésimo tomamos como pivote el de mayor valor absoluto de la j-ésima columna entre las filas j a la n (ambas inclusive).

Total:

Tomamos como pivote del paso j, el $a_{i,k}$ de mayor valor absoluto para i, k variando desde j hasta n. Deberemos controlar especialmente las columnas intercambiadas para obtener el resultado correcto al realizar la sustitución regresiva. Es decir ahora tomamos el de mayor valor absoluto de la matriz formada por las filas j a la n y las mismas columnas.

Técnica de normalización (scaling):

Al utilizar pivotaje, no se realizaría una misma elección si una ecuación aparece afectada por un factor de escala.

Ejemplo:

$$\left.\begin{array}{rrr} 3x+ & 2y & =8 \\ 0.1x+ & 0.0005y & =0.3 \end{array}\right\} \quad \text{y} \quad \left.\begin{array}{rrr} 0.03x+ & 0.02y & =0.08 \\ 10x+ & 0.05y & =30 \end{array}\right\}$$

donde en el primer caso el pivote sería el $a_{1,1} = 3$ y en el segundo el $a_{2,1} = 10$.

Para evitarlo se normaliza cada ecuación dividiendo por el coeficiente mayor, o cualquier otra técnica de balanceo.

Comentario El método de Gauss, con pivotaje total y normalización es el que más reduce los errores, pero a costa de complicar el algoritmo.

Generalmente basta con emplear pivotaje parcial.

1.2.3 Método de Gauss-Jordan

Se busca ahora obtener un sistema $Dx = b'$ equivalente al original, con D matriz diagonal.

Se computa mediante:

$$a_{i,j}^{k+1} = a_{i,j}^{k} - \frac{a_{i,k}^{k}}{a_{k,k}^{k}} a_{k,j} \quad \left\{ \begin{array}{l} i = 1, 2, \ldots, k-1, k+1, k+2, \ldots, n \\ j = k+1, k+2, \ldots, n, n+1 \end{array} \right\}$$

posteriormente basta hacer $x_k = \frac{b_k}{a_{k,k}}$

El número de operaciones es: $n^3 + n^2 - n$

El determinante de A, se obtiene como producto de los términos de la diagonal.($|A| = \prod_i d_{i,i}$).

Conclusiones

- Es más fácil de programar que Gauss. (Evita sustitución regresiva)
- Tiene más operaciones y como consecuencia acumula más errores que Gauss.
- Permite paralelismo (Si tenemos n-1 procesadores, cada uno puede hacer cero en una fila diferente).
- Es el método directo usual para calcular la matriz inversa.

1.2.4 Métodos de descomposición LU: Crout y Doolittle

El objetivo es descomponer la matriz A en el producto de dos matrices triangulares L y U, que posteriormente faciliten la resolución.

Definición: Una matriz cuadrada P es de permutación si en cada fila y/o columna tan solo existe un 1 siendo las restantes componentes 0. Se llama así porque al realizar una multiplicación de una matriz A por ella produce una permutación de las filas o columnas de A.

Propiedades:

- Dada una matriz cuadrada A, siempre existe una matriz de permutación P tal que PA=LU, donde L es triangular inferior y U superior.
- Si A es no singular $\Rightarrow$ L y U lo son.
- Si A es no singular, P es la identidad si y solo si los sucesivos menores principales de A son distintos de cero.
- La descomposición de PA en LU no es única, si hacemos la diagonal de U que sean unos ($U_{i,i} = 1$) se llamará método de Crout y si forzamos que ($L_{i,i} = 1$) se llamará método de Doolittle. Ambos tienen eficiencia computacional similar.

DESCOMPOSICIÓN PARA EL MÉTODO DE CROUT:

$$A = \begin{pmatrix} a_{1,1} & a_{1,2} & a_{1,3} & \dots & a_{1,n} \\ a_{2,1} & a_{2,2} & a_{2,3} & \dots & a_{2,n} \\ a_{3,1} & a_{3,2} & a_{3,3} & \dots & a_{3,n} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ a_{n,1} & a_{n,2} & a_{n,3} & \dots & a_{n,n} \end{pmatrix} = \begin{pmatrix} l_{1,1} & 0 & 0 & \dots & 0 \\ l_{2,1} & l_{2,2} & 0 & \dots & 0 \\ l_{3,1} & l_{3,2} & l_{3,3} & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ l_{n,1} & l_{n,2} & l_{n,3} & \dots & l_{n,n} \end{pmatrix} \begin{pmatrix} 1 & u_{1,2} & u_{1,3} & \dots & u_{1,n} \\ 0 & 1 & u_{2,3} & \dots & u_{2,n} \\ 0 & 0 & 1 & \dots & u_{3,n} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & 1 \end{pmatrix} =$$

$$= \begin{pmatrix} l_{1,1} & l_{1,1}u_{1,2} & l_{1,1}u_{1,3} & \dots & l_{1,1}u_{1,n} \\ l_{2,1} & l_{2,1}u_{1,2} + l_{2,2} & l_{2,1}u_{1,3} + l_{2,2}u_{2,3} & \dots & l_{2,1}u_{1,n} + l_{2,2}u_{2,n} \\ l_{3,1} & l_{3,1}u_{1,2} + l_{3,2} & l_{3,1}u_{1,3} + l_{3,2}u_{2,3} + l_{3,3} & \dots & l_{3,1}u_{1,n} + l_{3,2}u_{2,n} + l_{3,3}u_{3,n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ l_{n,1} & l_{n,1}u_{1,2} + l_{n,2} & l_{n,1}u_{1,3} + l_{n,2}u_{2,3} + l_{n,3} & \dots & l_{n,1}l_{1,n} + l_{n,2}u_{2,n} + l_{n,3}u_{3,n} + \dots + l_{n,n} \end{pmatrix}$$

Identificando coeficientes:

$$l_{i,1} = a_{i,1} \qquad (i = 1 \text{ hasta } n)$$

$$u_{1,j} = \frac{a_{1,j}}{l_{1,1}} (j = 2 \text{ hasta } n)$$

$$l_{i,k} = a_{i,k} - \sum_{m=1}^{m=k-1} l_{i,m}u_{m,k} \qquad (k = 2 \text{ hasta } n,\ i = k \text{ hasta } n)$$

$$u_{k,j} = \frac{a_{k,j} - \sum_{m=1}^{m=k-1} l_{k,m}u_{m,j}}{l_{k,k}} \qquad (k = 2..n, \quad j = k+1, n)$$

Resolución numérica

Una vez obtenida L y U tenemos Ax=b $\Rightarrow$ LUx=b $\Rightarrow$ (llamando y=Ux) $\Rightarrow$ Ly=b (que se resuelve por sustitución progresiva) $\Rightarrow$ Ux=y (por sustitución regresiva).

Determinante
El determinante de A será: $|A| = \prod_{i=1}^{n} l_{i,i}$ (si no hay intercambios, es decir P=I, en otro caso habría que multiplicar por $|P| = (-1)^p$, donde p es el número de intercambios)

Comentarios
La resolución por Crout tiene la misma eficiencia que Gauss, pues realmente se hacen los mismos cálculos, pero conservando la matriz L de operaciones que convierten A en triangular superior. Por tanto la matriz P de permutación es la identidad en los mismos casos que el método de Gauss no necesita intercambios.

Permite algo de paralelismo, pues muchos coeficientes de la descomposición pueden calcularse independientemente.

En el caso general deben considerarse los intercambios (igual que en Gauss), pudiendo hacerse pivotaje y normalización.

Reduce el cálculo necesario para la resolución de otros sistemas con la misma A y diferentes b, pues el cálculo más complejo es la descomposición (del orden de n^3 operaciones), mientras que las dos sustituciones solo necesitan $2n^2$ operaciones.

Permite empaquetamiento: Es decir la información de la descomposición (matrices L y U), pueden ser almacenadas en una única matriz, donde en la diagonal y debajo de ella están los $l_{i,j}$, mientras sobre la diagonal están los $u_{i,j}$.

1.2.5 Método de Cholesky o de la raiz cuadrada

Es una variante del LU, para matrices simétricas definidas positivas, conseguiéndose algunas ventajas.

Si A es simétrica y definida positiva, resulta que A puede descomponerse en el producto de TT^t donde T es triangular inferior.

DESCOMPOSICIÓN:

$$t_{1,1} = \sqrt{a_{1,1}} \qquad t_{j,1} = \frac{a_{j,1}}{t_{1,1}} \quad (j > 1)$$

$$t_{2,2} = \sqrt{a_{2,2} - t_{2,1}^2} \qquad t_{j,2} = \frac{a_{j,2} - t_{2,1}t_{j,1}}{t_{2,2}} \quad (j > 2)$$

$$t_{i,i} = \sqrt{a_{i,i} - \sum_{k=1}^{i-1} t_{i,k}^2} \qquad t_{j,i} = \frac{a_{j,i} - \sum_{k=1}^{i-1} t_{j,k}ti,k}{t_{i,i}} \quad (i > 2, j > i)$$

El número de operaciones es de $n^3/6 + n$, menor que en Gauss y Gauss-Jordan.

Resolución Dado el sistema Ax=b $\Rightarrow TT^t x = b \Rightarrow$ (llamando $T^t x = y$) $\Rightarrow$ Ty=b (sustitución progresiva.) y $T^t x = y$ (sustitución regresiva).
Determinante: El determinante de A es: $|A| = \prod_i t_{i,i}^2$

Consideraciones
Tiene la ventaja de reducir cálculo para sucesivas b, como en el caso LU.

Reduce espacio de almacenamiento y cálculo respecto a la LU.

Aunque para toda A (no singular) resulte que A^tA es simétrica y definida positiva, el sistema $A^tAx = A^tb$ resulta estar mal condicionado, por lo que es desaconsejable aplicarle Cholesky.

La condición necesaria y suficiente para que una matriz A simétrica admita descomposición en TT^t en los números reales, es que los sucesivos menores principales sean positivos, excepto el último que debe ser no negativo.

Para poder resolver por Cholesky es necesario y suficiente que la matriz A sea simétrica definida positiva, es decir todos los menores principales sucesivos deben ser positivos.

1.3 Métodos de ortogonalización. Sistemas sobredeterminados

Una matriz Q se dice **ortogonal** si y solo si $QQ^t = I$, o lo que es lo mismo $Q^t = Q^{-1}$

El método de factorización QR, consiste en descomponer la matriz A en QR donde Q es ortogonal y R triangular superior.

Resolución: Una vez obtenida la descomposición A=QR:

$$Ax = b \Rightarrow QRx = b \Rightarrow Q^tQRx = Q^tb \Rightarrow Rx = Q^tb$$

que se resolverán por sustitución regresiva.

Los métodos QR, en general necesitan más operaciones que los LU o Gauss, pero tienen en su favor que tienden a mantener el condicionamiento de la matriz A.

Una aplicación del método QR es a la resolución de sistemas sobredeterminados.

Sistema sobredeterminado: Un sistema sobredeterminado es un sistema imposible, donde se tomará como mejor solución la que minimice la norma 2 del vector residuo (r=Ax-b).

1.3.1 Método de ortogonalización de Househölder

Es un método para descomponer una matriz A en QR. Es preferido cuando la matriz A es densa (existe otro método debido a Givens, que es más eficiente cuando la matriz A es esparcida. Este en cada paso consigue ceros en todas las posiciones debajo de la diagonal en determinada columna, mientras que Givens solo consigue un nuevo cero.

Utiliza matrices ortogonales de la forma: $H = I - 2uu^t/(u^tu)$ donde los H_k se toman de forma que hagan 0 todos los términos de H_kA debajo de la diagonal principal en la columna k-ésima.

La elección más conveniente de u en el paso k-ésimo será: $u = \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ a_{k,k} + sgn(a_{k,k})\sqrt{\sum_{j=k}^{n} a_{j,k}^2} \\ a_{k+1,k} \\ a_{k+2,k} \\ \vdots \\ a_{n,k} \end{pmatrix}$

La descomposición se consigue en el orden de $4n^3/3$ operaciones, multiplicando sucesivamente por matrices H_i hasta llevar la matriz a triangular: $H_{n-1}H_{n-2}\dots H_2H_1A = R$, por lo que A=QR donde $Q = H_1^t H_2^t \dots H_{n-1}^t$

Ejemplo: Resolver por Housseholder:
$$\left.\begin{array}{rrrr} x & +2y & & = 2 \\ x & +3y & +2z & = 5 \\ 2x & & +z & = 7 \end{array}\right\}$$

Formo la matriz ampliada $(A|b)$ y tomo la primera columna de A y formo el vector:

$$u_1 = \begin{pmatrix} 1+\sqrt{1^2+1^2+2^2} \\ 1 \\ 2 \end{pmatrix} = \begin{pmatrix} 3.4495 \\ 1 \\ 2 \end{pmatrix}$$

Construyo a partir de él, la primera matriz ortogonal:

$$H1 = I - 2\frac{uu^t}{u^tu} = \begin{pmatrix} -0.4082 & -0.4082 & -0.8165 \\ -0.4082 & 0.8816 & -0.2367 \\ -0.8165 & -0.2367 & 0.5266 \end{pmatrix}$$

Multiplico $H1 * (A|b)$, resultando el sistema equivalente:

$$\begin{pmatrix} -2.4495 & -2.0412 & -1.6330 & -8.5732 \\ 0.0000 & 1.8285 & 1.5266 & 1.9348 \\ 0.0000 & -2.3431 & 0.0532 & 0.8697 \end{pmatrix}$$

Construyo un nuevo vector $u_2 = \begin{pmatrix} 0 \\ 1.8285+\sqrt{1.8285^2+(-2.3431)^2} \\ -2.3431 \end{pmatrix} = \begin{pmatrix} 0 \\ 4.8005 \\ -2.3431 \end{pmatrix}$, resultando:

$$H2 = I - 2\frac{u_2u_2^t}{u_2^tu_2} = \begin{pmatrix} 1.0000 & 0 & 0 \\ 0 & -0.6152 & 0.7884 \\ 0 & 0.7884 & 0.6152 \end{pmatrix}$$

Multiplicando de nuevo H2 por la matriz anterior obteniendo H2*H1*A=R

$$\begin{pmatrix} -2.4495 & -2.0412 & -1.6330 & -8.5732 \\ 0.0000 & -2.9721 & -0.8972 & -0.5047 \\ 0.0000 & 0 & 1.2362 & 2.0604 \end{pmatrix}$$

La solución se obtiene por sustitución regresiva: $z = \frac{2.0604}{1.2362} = 1.6667$

$$y = \frac{-0.5047+0.8972z}{-2.9721} = -0.3333,\ x = \frac{-8.5732+2.0412*y+1.6330*z}{-2.4495} = 2.6667$$

1.3.2 Aplicación a la resolución de sistemas sobredeterminados

Resolver un sistema sobredeterminado o imposible, consiste en obtener el vector $\bar{x}$, tal que minimice alguna norma del vector residuo: $\bar{r} = A\bar{x} - b$.

El método QR proporciona una herramienta para obtener la solución de un sistema sobredeterminado por el criterio de los mínimos cuadrados o norma 2 vectorial.

Con este criterio $\bar{x_0}$ es la solución si y solo si $\|r\|_2^2 = \|A\bar{x_0} - b\|_2^2 = \sum_i r_i^2$ se hace mínimo para ese $\bar{x_0}$. ($\|A\bar{x_0} - b\| \leq \|Ax - b\|, \quad \forall x$).

Ejemplo:

Resolver el sistema: $\left.\begin{array}{r} 2x + 3y = 5 \\ 2x + y = 3 \\ x - y = 1 \end{array}\right\}$

Evidentemente es un sistema sobredeterminado pues rango(A)=2 y rango(A^+)=3.

Se calcula u=$\begin{pmatrix} 2 + \sqrt{2^2 + 2^2 + 1^2} \\ 2 \\ 1 \end{pmatrix} = \begin{pmatrix} 5 \\ 2 \\ 1 \end{pmatrix}$

Se calcula $H = I - 2\frac{uu^t}{u^t u} = \begin{pmatrix} -0.6667 & -0.6667 & -0.3333 \\ -0.6667 & 0.7333 & -0.1333 \\ -0.3333 & -0.1333 & 0.9333 \end{pmatrix}$

Multiplicamos la matriz H por la ampliada de A $(A|b)$, obteniendo

$$H(A|b) = \left(\begin{array}{rr|r} -3.0000 & -2.3333 & -5.0000 \\ 0.0000 & -1.1333 & -2.0000 \\ 0 & -2.0667 & -1.0000 \end{array}\right)$$

Construimos de nuevo el vector u: $u2 = \begin{pmatrix} 0 \\ -1.1333 - \sqrt{(-1.1333)^2 + (-2.0667)^2} \\ -2.0667 \end{pmatrix} = \begin{pmatrix} 0 \\ -3.4904 \\ -2.0667 \end{pmatrix}$

Formo la matriz $H2 = I - 2\frac{u2u2^t}{u2^t u2} = \begin{pmatrix} 1.0000 & 0 & 0 \\ 0 & -0.4808 & -0.8768 \\ 0 & -0.8768 & 0.4808 \end{pmatrix}$

Multiplico H2*HA obteniendo: $\left(\begin{array}{rr|r} -3.0000 & -2.3333 & -5.6667 \\ 0.0000 & 2.3570 & 1.6028 \\ 0.0000 & 0.0000 & 0.5657 \end{array}\right)$

Sustituyendo hacia atrás con las 2 primeras ecuaciones:

$$y = \frac{1.6028}{2.3570} = 0.6800, \qquad x = \frac{-5.6667 + 2.3333 * y}{-3} = 1.3600$$

El vector residuo se obtiene como $r = A\begin{pmatrix} 1.36 \\ 0.68 \end{pmatrix} - b = \begin{pmatrix} -0.24 \\ 0.40 \\ -0.32 \end{pmatrix}$

Obteniéndose $\sum_i r_i^2 = 0.3200, \Rightarrow \|r\|_2 = \sqrt{\sum_i r_i^2} = 0.5657$

1.4 Métodos iterativos para sistemas lineales

Para grandes sistemas los métodos directos son ineficaces debido a la propagación de errores de cálculo, mientras que los métodos iterativos, tras realizar una iteración, toman el vector

obtenido como si fuese el vector inicial, por lo que no se propaga el error. Otro posible motivo para usar métodos iterativos es la necesidad de mantener la matriz A en memoria en los métodos directos (problema de almacenamiento) que los hace inviables.

Es fácil resolver un sistema mediante un método iterativo, sin necesidad de almacenar la matriz en memoria. En casos extremadamente grandes podría leerse una ecuación desde un fichero, posteriormente otra, ...

Además en la mayoría de los sistemas grandes (2500 ecuaciones o más), se trata usualmente de matrices esparcidas, por lo que aunque pueden programarse métodos directos especiales para este tipo de matrices es más fácil resolverlo por técnicas iterativas.

Si $n \geq 50$, los métodos directos se consideran inservibles y mucho antes si la matriz A está mal condicionada.

Los métodos iterativos tendrán los problemas usuales de estos: convergencia, velocidad de convergencia, obtención de un valor inicial.

Generalidades:

Buscaremos una sucesión de vectores $\{x^{(m)}\}$ que converja hacia x^*, tal que $Ax^* = b$. Usualmente el vector siguiente se calcula a partir del anterior: $x^{(k+1)} = F(x^{(k)})$ donde para sistemas lineales la función F será de la forma: $F(x^{(k+1)}) = Bx^{(k)}+$ C.

1.4.1 Condiciones de convergencia

1) Un método iterativo del tipo: $x^{(m+1)} = Bx^{(m)} + C$ es convergente si existe $L \in [0, 1)$ tal que $\|Bv\| \leq L\|v\|$ para todo vector v. Si para alguna norma ocurre la propiedad, es convergente.

2) Un método iterativo es convergente a la solución de un sistema $Ax = b$ si y solo si:

$$a)\rho(B) < 1 \qquad b)C = (I - B)A^{-1}b$$

1.4.2 Métodos iterativos usuales

Los que vamos a ver son métodos de *partición regular* que consisten en descomponer la matriz A en diferncia de otras dos (A=M-N), resultando:

$$Ax = b \Rightarrow (M - N)x = b \Rightarrow Mx = Nx - b \Rightarrow x = M^{-1}Nx - M^{-1}b$$

que determina el algoritmo iterativo: $x^{k+1)} = M^{-1}Nx^{k)} - M^{-1}b$

Es necesario que M sea inversible y para que sea convergente que $\rho(M^{-1}N) < 1$.

1.4.3 Método de Jacobi

Dado el sistema lineal:

$$\left.\begin{array}{rl} a_{1,1}x_1 + a_{1,2}x_2 + a_{1,3}x_3 + \ldots + a_{1,n}x_n & = b_1 \\ a_{2,1}x_1 + a_{2,2}x_2 + a_{2,3}x_3 + \ldots + a_{2,n}x_n & = b_2 \\ a_{3,1}x_1 + a_{3,2}x_2 + a_{3,3}x_3 + \ldots + a_{3,n}x_n & = b_3 \\ \ldots & \ldots \\ a_{n,1}x_1 + a_{n,2}x_2 + a_{n,3}x_3 + \ldots + a_{n,n}x_n & = b_n \end{array}\right\}$$

podemos despejar x_i de la ecuación i-ésima:

$$\left.\begin{array}{rl} x_1 = & \frac{1}{a_{1,1}}(b_1 - a_{1,2}x_2 - a_{1,3}x_3 - \ldots - a_{1,n}x_n) \\ x_2 = & \frac{1}{a_{2,2}}(b_2 - a_{2,1}x_1 - a_{2,3}x_3 - \ldots - a_{2,n}x_n) \\ x_3 = & \frac{1}{a_{3,3}}(b_3 - a_{3,1}x_1 - a_{3,2}x_2 - \ldots - a_{3,n}x_n) \\ \ldots & \ldots \\ x_n = & \frac{1}{a_{n,n}}(b_n - a_{n,1}x_1 - a_{n,2}x_2 - a_{n,3}x_3 - \ldots - a_{n,n-1}x_{n-1}) \end{array}\right\}$$

El método de Jacobi itera mediante:

$$\left.\begin{array}{rl} x_1^{(r+1)} = & \frac{1}{a_{1,1}}(b_1 - a_{1,2}x_2^{(r)} - a_{1,3}x_3^{(r)} - \ldots - a_{1,n}x_n^{(r)}) \\ x_2^{(r+1)} = & \frac{1}{a_{2,2}}(b_2 - a_{2,1}x_1^{(r)} - a_{2,3}x_3^{(r)} - \ldots - a_{2,n}x_n^{(r)}) \\ x_3^{(r+1)} = & \frac{1}{a_{3,3}}(b_3 - a_{3,1}x_1^{(r)} - a_{3,2}x_2^{(r)} - \ldots - a_{3,n}x_n^{(r)}) \\ \ldots & \ldots \\ x_n^{(r+1)} = & \frac{1}{a_{n,n}}(b_n - a_{n,1}x_1^{(r)} - a_{n,2}x_2^{(r)} - a_{n,3}x_3^{(r)} - \ldots - a_{n,n-1}x_{n-1}^{(r)}) \end{array}\right\}$$

En el método de Jacobi siempre se obtiene, iterando según la fórmula anterior, cada nueva componente del nuevo vector $x^{(r+1)}$ a partir de las componentes del vector anterior $x^{(r)}$, sin emplear las componentes actualizadas en el paso actual.

Matricialmente: Descomponiendo la matriz A:

$$A = L + D + R \Rightarrow Ax = b \Rightarrow (D + L + R)x = b \Rightarrow$$

$$Dx = -(L + R)x + b \Rightarrow x = -D^{-1}(L + R)x + D^{-1}b$$

Luego $x^{(r+1)} = F(x^{(r)}) = Bx^{(r)} + C$ con:

$$C = D^{-1}b, \quad B = -D^{-1}(L + R) = -D^{-1}(L + R + D - D) = -D^{-1}(A - D) = I - D^{-1}A$$

Además de las características propias de todos los métodos iterativos, éste permite paralelismo, pues cada componente $x_i^{(r+1)}$, pude calcularse por un procesador diferente.

1.4.4 Método de Gauss-Seidel

En las ecuaciones anteriores cuando calculamos $x_i^{(r+1)}$ en función de los $x_i^{(r)}$, podemos emplear algunos $x_j^{(r+1)}$, aquellos con $j < i$, que ya han sido calculados, con lo que aceleramos la convergencia. En esto consiste el método de Gauss-Seidel.

Las ecuaciones por la que iteramos son ahora:

$$\left.\begin{array}{rl} x_1^{(r+1)} = & \frac{1}{a_{1,1}}(b_1 - a_{1,2}x_2^{(r)} - a_{1,3}x_3^{(r)} - \ldots - a_{1,n}x_n^{(r)}) \\ x_2^{(r+1)} = & \frac{1}{a_{2,2}}(b_2 - a_{2,1}x_1^{(r+1)} - a_{2,3}x_3^{(r)} - \ldots - a_{2,n}x_n^{(r)}) \\ x_3^{(r+1)} = & \frac{1}{a_{3,3}}(b_3 - a_{3,1}x_1^{(r+1)} - a_{3,2}x_2^{(r+1)} - \ldots - a_{3,n}x_n^{(r)}) \\ \ldots & \ldots \\ x_n^{(r+1)} = & \frac{1}{a_{n,n}}(b_n - a_{n,1}x_1^{(r+1)} - a_{n,2}x_2^{(r+1)} - a_{n,3}x_3^{(r+1)} - \ldots - a_{n,n-1}x_{n-1}^{(r+1)}) \end{array}\right\}$$

Matricialmente:

$$Ax = b \Leftrightarrow (D+L+R)x = b \Leftrightarrow (D+L)x = -Rx + b \Leftrightarrow x = -(D+L)^{-1}Rx + (D+L)^{-1}b$$

Luego $C = (D+L)^{-1}b, \quad B = -(D+L)^{-1}R = -(D+L+R-R)^{-1}R = -(A-R)^{-1}R$

Casi siempre es más rápido que el de Jacobi, aunque no aprovecha el paralelismo como él.

1.4.5 Método de Sobre-relajación (SOR)

Consiste en poner un parámetro w de relajación, que sirva para acelerar la convergencia del método de Gauss-Seidel. Si una componente vale $x_i^{(r)}$ y por Gauss-Seidel pasaría a valer $x_{i,GS}^{(r+1)}$, el incremento para Gauss-Seidel sería: $\Delta_{GS} = x_{i,GS}^{(r+1)} - x_i^{(r)}$, el método SOR le da un incremento generalmente mayor ($w > 1$) haciendo:

$$\Delta_{SOR} = w\Delta_{GS} = w(x_{i,GS}^{(r+1)} - x_i^{(r)})$$

Nótese que si w=1, sigue siendo el método de Gauss-Seidel.
La fórmula iterativa queda:

$$x_i^{(r+1)} = x_i^{(r)} + w(x_{i,GS}^{(r+1)} - x_i^{(r)}) = (1-w)x_i^{(r)} + w(x_{i,GS}^{(r+1)})$$

Es decir:

$$\left.\begin{array}{ll}
x_1^{(r+1)} = (1-w)x_1^{(r)} & +\frac{w}{a_{1,1}}(b_1 - a_{1,2}x_2^{(r)} - a_{1,3}x_3^{(r)} - \ldots - a_{1,n}x_n^{(r)}) \\
x_2^{(r+1)} = (1-w)x_2^{(r)} & +\frac{w}{a_{2,2}}(b_2 - a_{2,1}x_1^{(r+1)} - a_{2,3}x_3^{(r)} - \ldots - a_{2,n}x_n^{(r)}) \\
x_3^{(r+1)} = (1-w)x_3^{(r)} & +\frac{w}{a_{3,3}}(b_3 - a_{3,1}x_1^{(r+1)} - a_{3,2}x_2^{(r+1)} - \ldots - a_{3,n}x_n^{(r)}) \\
\ldots & \ldots \\
x_n^{(r+1)} = (1-w)x_n^{(r)} & +\frac{w}{a_{n,n}}(b_n - a_{n,1}x_1^{(r+1)} - a_{n,2}x_2^{(r+1)} - a_{n,3}x_3^{(r+1)} - \ldots - a_{n,n-1}x_{n-1}^{(r+1)})
\end{array}\right\}$$

El valor de w se toma usualmente ligeramente superior a 1 (1.2, 1.25, 1.1, ... son valores típicos) para que la convergencia sea acelerada (métodos de sobre-relajación), pero a veces, cuando Gauss-Seidel no es convergente, se toma $0 < w < 1$, para conseguir que resulte convergente (métodos de sub-relajación).

Matricialmente:

$$Ax = b \Rightarrow wAx = wb \Rightarrow w(D+L+R) = wb \Rightarrow (1-w)Dx + w(D+L+R)x = (1-w)Dx + wb \Rightarrow$$

$$(D + wL + wR)x = (1-w)Dx + wb \Rightarrow (D+wL)x = -wRx + (1-w)Dx + wb \Rightarrow$$

$$(D+wL)x = [(1-w)D - wR]x + wb \Rightarrow x = (D+wL)^{-1}[(1-w)D - wR]x + (D+wL)^{-1}wb$$

1.4.6 Condiciones de convergencia de los métodos iterativos usuales

- **La condición necesaria y suficiente para que un método sea convergente es que el radio espectral de B sea estrictamente menor que 1.**
- Jacobi converge si todas las soluciones de $|\lambda D + L + R| = 0$ son en módulo menores de 1. (Son los autovalores de B_j). (Son los autovalores de B_J).

- Gauss-Seidel converge si todas las soluciones de $|\lambda D + \lambda L + R| = 0$ sean en módulo menores de 1. (Son los autovalores de B_{GS}).
- Si A es tridiagonal y simétrica definida positiva, convergen Jacobi y Gauss-Seidel, siendo el valor óptimo de w para el método SOR: w=$\frac{2}{1+\sqrt{1-\rho_J^2}}$, valor para el que $\rho(B_{SOR}) = w-1$ y se verifica además $\rho(B_J) = (\rho B_{GS})^2 < 1$
- Si A es simétrica definida positiva y $0 < w < 2$, converge el método SOR (w=1 es Gauss-Seidel).
- Si D-L-R es simétrica definida positiva entonces converge Jacobi.
- Si A es de diagonal estrictamente dominante por filas o por columnas, entonces convergen Jacobi, Gauss-Seidel y SOR ($0 < w \leq 1$).

1.5 Error en la solución de un sistema lineal. Condición de una matriz

Al resolver un sistema, aparecen errores de redondeo al efectuar cualquier operación aritmética, que posteriormente se propagan al efectuar nuevos cálculos. Por otra parte aunque los métodos iterativos son más insensibles a esta propagación del error, necesitarían de un proceso de infinitas iteraciones para obtener la solución exacta, por lo que cuando terminamos de iterar tendremos en cualquier caso una solución aproximada.

1.5.1 Acotación del error en los métodos iterativos

Sea el método iterativo general $x^{(r+1)} = Bx^{(r)} + C$, llamaremos x^* a la solución exacta que verificará $x^* = Bx^* + C$.

El error tras m+1 iteraciones será: $\|x^{(m+1)} - x^*\| = \|Bx^{(m)} + C - (Bx^* + C)\| = \|B(x^{(m)} - x^*)\| \leq \|B\|\|x^{(m)} - x^*\|$

Luego en cada iteración el error se reduce al menos en el factor $\|B\|$, lo que ocurre para cualquier norma empleada.

Llevando la expresión al error inicial:

$$\|x^{(m+1)} - x^*\| \leq \|B\|\|x^{(m)} - x^*\| \leq \|B\|^2\|x^{(m-1)} - x^*\| \leq \ldots \leq \|B\|^{m+1}\|x^{(0)} - x^*\|$$

Si calculamos ahora:

$$\|x^{(m+1)} - x^{(m)}\| = \|Bx^{(m)} + C - Bx^{(m-1)} - C\| \leq \|B\|\|x^{(m)} - x^{(m-1)}\| \leq \ldots \leq \|B\|^m\|x^{(1)} - x^{(0)}\|$$

que nos sirve para obtener:

$$\|x^{(m)} - x^*\| \leq \|x^{(m+1)} - x^{(m)}\| + \|x^{(m+2)} - x^{(m+1)}\| + \|x^{(m+3)} - x^{(m+2)}\| + \ldots \leq$$

$$\leq \|B\|^m\|x^{(1)} - x^{(0)}\| + \|B\|^{m+1}\|x^{(1)} - x^{(0)}\| + \|B\|^{m+2}\|x^{(1)} - x^{(0)}\| + \ldots = \frac{\|B\|^m}{1 - \|B\|}\|x^{(1)} - x^{(0)}\|$$

que expresa el error tras m iteraciones en función de la norma de la matriz B y la norma de la diferencia entre el vector inicial y la primera iteración.

1.5.2 Estimación del error cometido al resolver un sistema

Intentamos resolver $Ax = b$ por algún método, pero obtenemos $x' \backslash Ax' = b + r$, siendo la verdadera solución $x^* \,/\, Ax^* = b$.

Llamamos **vector de errores** a $\Delta x = x' - x^*$ y **vector residuo** a $r = Ax' - b$.

Entonces:

$$r = Ax' - b = Ax' - Ax^* = A\Delta x \Rightarrow \|r\| \leq \|A\|\|\Delta x\| \Rightarrow \|\Delta x\| \geq \frac{\|r\|}{\|A\|}$$

$$r = A\Delta x \Rightarrow \Delta x = A^{-1}r \Rightarrow \|\Delta x\| \leq \|A^{-1}\|\|r\|$$

Hemos obtenido una cota inferior y otra superior para la norma del vector error:

$$\frac{\|r\|}{\|A\|} \leq \|\Delta x\| \leq \|A^{-1}\|\|r\|$$

A partir de Ax=b y $x = A^{-1}b$ se obtiene: $\frac{\|b\|}{\|A\|} \leq \frac{\|\Delta x\|}{\|x\|} \leq \|A^{-1}\|\|b\|$

De ambas se obtiene:

$$\frac{1}{c(A)}\frac{\|r\|}{\|b\|} \leq \frac{\|\Delta x\|}{\|x\|} \leq c(A)\frac{\|r\|}{\|b\|}$$

Donde el **número de condición de la matriz A**, viene expresado por c(A), y tiene por valor: $c(A) = \|A\|\|A^{-1}\|$, y será (para las normas inducidas) siempre mayor que 1 pues:

$$c(A) = \|A\|\|A^{-1}\| \geq \|AA^{-1}\| = \|I\| = 1$$

1.5.3 Caso de existir errores en los datos

Suponemos ahora que los coeficientes a_{ij} y b_i son aproximados y queremos ver como puede afectar al resultado. Es decir, estamos resolviendo $(A + \Delta A)x = b + \Delta b$, del que obtenemos la solución exacta x', mientras el sistema que de verdad queremos resolver es Ax=b, que tiene la solución exacta x^*.

El error será: $\Delta x = x' - x^* \Rightarrow (A+\Delta A)(x^*+\Delta x) = b+\Delta b$ que desarrollando y despreciando el término $\Delta A\Delta x$, se obtiene: $A\Delta x + \Delta Ax^* = \Delta b \Rightarrow$

$$\Delta x \approx A^{-1}(\Delta b - \Delta Ax^*)$$

TEOREMA:

Llamando $rel(x) = \frac{\|\Delta x\|}{\|x\|}$, $\quad rel(A) = \frac{\|\Delta A\|}{\|A\|}$, $\quad rel(b) = \frac{\|\Delta b\|}{\|b\|}$, entonces:

$$rel(x) \leq c(A)\frac{rel(A) + rel(b)}{1 - c(A)rel(A)}$$

1.6 Anexo (Cálculo de determinantes, matriz inversa y autovalores

1.6.1 Cálculo de determinantes

- Si la matriz es diagonal o triangular su determinante es el producto de los términos diagonales.
- Si triangularizamos por Gauss o diagonalizamos por Gauss-Jordan, será el producto de los términos de la diagonal principal, pero debemos tener en cuenta que si se realiza un intercambio, el determinante cambia de signo.
- Se podría calcular mediante los métodos LU, teniendo en cuenta $|A| = |L||U|$. Tambien deberemos considerar los intercambios.

Es de notar que en esas operaciones se introducen errores de redondeo que son propagados, por lo que es difícil asegurar numéricamente si un determinante es 0 o no lo es.

1.6.2 Cálculo de la matriz inversa

Para n elevado, el método para calcular A^{-1} como la adjunta de la traspuesta partido el $|A|$, no es conveniente por extremadamente ineficiente.

Calculo por vectores columnas

Calcular la inversa de A, será hallar una matriz $V = (v_{ij})$ tal que al multiplicarla por A, nos dé la identidad.

Llamando $\bar{v}_i$ a la columna i-exima de la matriz inversa y $\bar{e}_i$ al vector con todas sus componentes cero excepto la i-ésima donde tiene un 1, tenemos que se debe verificar: $A\bar{v}_i = \bar{e}_i$. Luego resolviendo n sistemas (uno para cada columna de la inversa), obtenemos la matriz inversa.

Como en los n sistemas Ax=b, solo cambia el vector b, resultan convenientes los métodos de factorización. En otros métodos (Gauss, Gauss-Jordan), existen técnicas para realizar cálculos comunes para la resolución de los n sistemas simultáneamente.

Aunque no resulten los mejores a veces se necesita usar los métodos iterativos para resolver los n sistemas.

Inversión por Gauss-Jordan

Si A es no singular, formo $(A|I)$ y mediante transformaciones por filas, transformamos la primera submatriz en la unidad, resultando $(I|A^{-1})$.

El cálculo realizado será equivalente a multiplicar por A^{-1}, pues $A^{-1}(A|I) = (I|A^{-1})$

EJEMPLO: Calcular la inversa de A= $\begin{pmatrix} 4 & 2 & 1 \\ 2 & 3 & 1 \\ 0 & 1 & 4 \end{pmatrix}$

Formo: $(A|I) = \left(\begin{array}{ccc|ccc} 4 & 2 & 1 & 1 & 0 & 0 \\ 2 & 3 & 1 & 0 & 1 & 0 \\ 0 & 1 & 4 & 0 & 0 & 1 \end{array}\right) \rightarrow \left(\begin{array}{ccc|ccc} 1 & 0.5 & 0.25 & 0.25 & 0 & 0 \\ 0 & 2 & 0.5 & -0.5 & 1 & 0 \\ 0 & 1 & 4 & 0 & 0 & 1 \end{array}\right) \rightarrow$

$\left(\begin{array}{ccc|ccc} 1 & 0 & 0.125 & 0.375 & -0.25 & 0 \\ 0 & 1 & .25 & -0.25 & 0.5 & 0 \\ 0 & 0 & 3.75 & 0.25 & -0.5 & 1 \end{array}\right) \rightarrow \left(\begin{array}{ccc|ccc} 1 & 0 & 0 & 0.36666 & -0.26666 & -0.03333 \\ 0 & 1 & 0 & -2.3333 & 0.46666 & -0.06666 \\ 0 & 0 & 1 & 0.06666 & -0.13333 & 0.26666 \end{array}\right)$

1.6.3 Métodos numéricos para el cálculo de autovalores y autovectores

Un número λ (real o complejo) es un autovalor (valor propio) de A, si y solo si existen vectores x tales que $Ax = \lambda x$ (los vectores se llaman vectores propios o autovectores).

Los valores propios de una matriz A son los números (reales o complejos) λ_i, tales que $|A - \lambda_i I| = 0$.

Para cada valor propio λ_i, el sistema $(A - \lambda_i I)x = 0$, tiene soluciones diferentes de la trivial, los vectores x que son solución de dicho sistema serán los vectores propios de la matriz. (Asociados al valor propio λ_i).

Puede hablarse de autovectores por la derecha $(Ax = \lambda x)$,y por la izquierda.$(xA = \lambda x)$, los valores propios resultan ser los mismos, pero no los vectores propios.

No existen métodos directos pues no es posible calcular directamente las raíces de un polinomio. De todas formas se denominan directos los métodos que calculan los coeficientes del polinomio característico de forma directa aunque necesiten posteriormente algún método iterativo para su resolución.

Los métodos iterativos se clasifican según busquen todos los autovalores, solo el valor propio dominante, busquen obtener también los autovectores, servir solo para determinado tipo de matrices: (simétricas, tridiagonales, hermitianas, ...).

Propiedades de los valores propios

1) Si λ_i, es un valor propio de A, entonces λ_i^n lo es de A^n.

2) Si λ_i y λ_j, son dos autovalores distintos de una matriz A, entonces los vectores propios asociados son ortogonales.

3) Si A es simétrica entonces:

- Los valores característicos son reales.
- Existen n vectores característicos que forman un sistema ortonormal.
- Existe una matriz P (ortogonal) tal que $D = P^t AP$ con D diagonal y de términos d_{ii} los valores característicos de A.

4) A^t posee los mismos valores propios que A, pero no siempre los mismos vectores propios.

5) Si existe A^{-1}, los valores propios de A^{-1} son los inversos de los de A.

Acotación (Teorema de los circulos de Gerschgorin: Si denotamos por R_i el círculo del plano complejo de centro a_{ii} y radio $\sum_{j \neq i} |a_{i,j}|$, entonces en la región $R = \bigcup_i R_i$ se encuentran todos los autovalores de A.

Método de Leverrier (Cálculo del polinomio característico): Como método directo de cálculo del polinomio característico está el siguiente:

Si $\lambda^n + p_1\lambda^{n-1} + p_2\lambda^{n-2} + \ldots + p_{n-1}\lambda + p_n$ es el polinomio característico buscado, llamando $s_k = \sum_i \lambda_i^k = tr(A^k)$, entonces $p_k = \frac{-1}{k}\left[s_k + \sum_{i=1}^{n-1} p_i s_{k-i}\right]$

Método de la potencia: Este método busca el autovalor dominante.

Sea y_0 un vector arbitrario, construimos $y_1 = Ay_0; \quad y_2 = Ay_1; \ldots ; y_p = Ay_{p-1}$, entonces si la componente i-ésima de y_{p-1} es distinta de cero, el cociente $\frac{(y_p)_i}{|(y_{p-1})_i|} \to \lambda_1$

Donde λ_1, es el autovalor dominante, es decir $|\lambda_1| \geq |\lambda_i|$, $\forall i$.

Si existen diferentes autovalores con el mayor módulo el método puede tener problemas

NOTAS: Si aplicamos el método de la potencia a la matriz A^{-1} entonces el autovalor encontrado es $\frac{1}{\lambda}$ con λ el autovalor de menor módulo de A. Si se lo aplicamos a $(A - kI)^{-1}$

resultando un valor ϵ, resulta que $k - \frac{1}{\epsilon}$ es el autovalor de A más cercano a k (Permite depurar un autovalor).

Algoritmo del método de la potencia:

Paso 1) Tomo un vector normalizado. ($\max \mid y_0^i \mid = 1$)

Paso 2) Repetir desde j=0 hasta criterio de parada. $|k_{j+1} - k_j| < \epsilon$

Calculo $z_{j+1} = Ay_j$ y $k_{j+1} = \max_i |z_{j+1}^i|$,

Calculamos $y_{j+1} = \frac{z_j}{k_{j+1}}$ (Normalización)

Paso 3) El valor propio dominante $\approx k_{j+1}$ y el vector propio asociado $\approx y_{j+1}$

Método QR Es el método básico para el cálculo de todos los autovalores y autovectores y consiste en repetir el algoritmo:

Paso 1: Llamo $A_0 = A$, y realizo su descomposición QR: $A_0 = Q_0 R_0$.

Paso 2: Obtengo $A_1 = R_0 Q_0$

Paso 3: Repetir hasta criterio de parada: $A_i = Q_i R_i, \qquad A_{i+1} = R_i Q_i$

Se demuestra que la sucesión de matrices $\{A_i\}$ converge hacia una matriz cuasitriangular con los valores propios reales en la diagonal.

La complejidad y velocidad de convergencia es mejorada reduciendo previamente mediante transformaciones de Householder o Givens:

- Si A es simétrica a tridiagonal simétrica.
- Si A no es simétrica a Hessemberg superior. ($A_{i,j} = 0$ si $i > j + 1$)

NOTA: Es fácil observar que las matrices A_i son equivalentes entre sí, pues por su construcción $R_i = Q_i^{-1} A_i = Q_i^t A_i$, por lo que: $A_{i+1} = R_i Q_i = Q_i^t A_i Q_i$, luego A_{i+1} y A_i son equivalentes y tienen los mismos autovalores.

Ejemplo: Calcular los valores propios de: $A = \begin{pmatrix} 1 & -1 & 2 & 4 \\ 3 & 2 & 5 & 7 \\ 0 & 1 & 4 & 5 \\ 2 & 1 & 3 & 5 \end{pmatrix}$

Por el método de Leverrier:

$$A^2 = \begin{pmatrix} 6 & 3 & 17 & 27 \\ 23 & 13 & 57 & 86 \\ 13 & 11 & 36 & 52 \\ 15 & 8 & 36 & 55 \end{pmatrix} \quad A^3 = \begin{pmatrix} 69 & 44 & 176 & 265 \\ 234 & 146 & 597 & 898 \\ 150 & 97 & 381 & 569 \\ 149 & 92 & 379 & 571 \end{pmatrix} \quad A^4 = \begin{pmatrix} 731 & 460 & 1857 & 2789 \\ 2468 & 1553 & 6280 & 9433 \\ 1579 & 994 & 4016 & 6029 \\ 1567 & 985 & 3987 & 5990 \end{pmatrix}$$

$$s1 = tr(A) = 12 \qquad s_2 = tr(A^2) = 110 \qquad s_3 = tr(A^3) = 1167 \qquad s_4 = tr(A^4) = 12290$$

$p_1 = -s_1 = -12 \qquad p_2 = -(s_2 + s_1 p_1)/2 = 17 \qquad p_3 = -(s_3 + s_2 p_1 + s_1 p_2)/3 = -17 p_4 = -(s_4 + s_3 p_1 + s_2 p_2 + s_1 p_3)/4 = 12 \Leftrightarrow p(\lambda) = \lambda^4 - 12\lambda^3 + 17\lambda^2 - 17\lambda + 12$

Que resolviendo da: $\lambda_1 = 10.5284 \quad \lambda_2 = 1.0631 \quad \lambda_{3,4} = 0.2043 \pm 1.0151i$

Por el método de la potencia: Tomo un vector arbitrario: $y = (1\,2\,3\,4)' \Leftrightarrow y_1 = Ay = (21\,50\,34\,33)'$

$$y_2 = Ay_1 = (171\,564\,351\,359)' \quad y_3 = (1745\,5909\,3763\,3754)'$$

$y_4 = (18378\,62146\,39731\,39458)'$ y dividiendo término a término y_4 entre y_3 resulta: (10.5318 10.5172 10.5583 10.5109) donde cualquiera de sus componentes son aproximaciones al valor propio dominante.

Existen métodos de deflacción para encontrar una matriz 4x4 que posea los autovalores restantes y el 0.

Por el método QR: Tras descomponer sucesivas veces $A_0 = A; A_i = Q_i R_i; A_{i+1} = R_i Q_i$ converge a:

$$Q = \begin{pmatrix} -1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & -0.1365 & 0.9906 \\ 0 & 0 & 0.9906 & 0.1365 \end{pmatrix} A = \begin{pmatrix} 10.5284 & \neq 0 & \neq 0 & \neq 0 \\ 0 & 1.0627 & \neq 0 & \neq 0 \\ 0 & 0 & 0.3489 & -2.4356 \\ 0 & 0 & 0.4316 & 0.0595 \end{pmatrix}$$

que nos indica que existen dos autovalores reales y dos complejos, siendo estos últimos los autovalores de la submatriz $\begin{pmatrix} 0.3489 & -2.4356 \\ 0.4316 & 0.0595 \end{pmatrix}$

2 INTERPOLACION Y APROXIMACION.

2.1 Interpolación polinomial clásica. Fórmulas de Lagrange y en diferencias. Error

Dados un conjunto de valores experimentales $\{(x_i, y_i)i \in I\}$, buscamos la función de la que provienen. Interpolar será encontrar una función que pase exactamente por los puntos anteriores.

Dados $\{(x_i, y_i)i \in I\}$, obtenidos de una función desconocida y un subconjunto $\mathcal{M} \subset \mathcal{F}(\mathcal{R}, \mathcal{R})$, la interpolación consiste en encontrar un elemento $g \in \mathcal{F}$ que verifique $g(x_i) = y_i, \forall i$.

Generalmente $\mathcal{M}$ coincide con las funciones polinomicas, pero podrán ser splines, trigonométricas, etc, ...

Además podrán existir otro tipo de condiciones sobre $f'(x_i)$, $f''(x_i), \ldots$

Otra generalización será a funciones entre espacios vectoriales ($R^m \to R^n$), donde si n=1, las funciones serían formas lineales.

En general podrá ocurrir que no exista la función interpoladora en $\mathcal{M}$, que existan muchas o solamente una. Veamos condiciones de existencia y unicidad:

Si queremos interpolar a los puntos $\{(x_i, y_i)i \in I\}$, una función y=F(x) de una familia $\mathcal{M}$ de funciones de las que conocemos una base $B = \{\varphi_0, \varphi_1, \ldots, \varphi_n\}$, entonces debe verificarse: $\forall i, F(x) = a_0\varphi_0(x) + a_1\varphi_1(x) + \ldots + a_n\varphi_n(x)$ y debe encontrarse una solución del sistema:

$$\left.\begin{array}{ccccccccccc} a_0\varphi_0(x_0) & + & a_1\varphi_1(x_0) & + & a_2\varphi_2(x_0) & + & \ldots & + & a_n\varphi_n(x_0) & = & y_0 \\ a_0\varphi_0(x_1) & + & a_1\varphi_1(x_1) & + & a_2\varphi_2(x_1) & + & \ldots & + & a_n\varphi_n(x_1) & = & y_1 \\ a_0\varphi_0(x_2) & + & a_1\varphi_1(x_2) & + & a_2\varphi_2(x_2) & + & \ldots & + & a_n\varphi_n(x_2) & = & y_2 \\ \vdots & & \vdots & & \vdots & & \vdots & & \vdots & & \vdots \\ a_0\varphi_0(x_n) & + & a_1\varphi_1(x_n) & + & a_2\varphi_2(x_n) & + & \ldots & + & a_n\varphi_n(x_n) & = & y_n \end{array}\right\}$$

Resolviendo el sistema, obtenemos los a_i, y la función interpoladora:

$$F(x) = a_0\varphi_0(x) + a_1\varphi_1(x) + \ldots + a_n\varphi_n(x)$$

Para que el sistema tenga solución única, su determinante deberá ser distinto de cero.

$$\Delta = \left|\begin{array}{ccccc} \varphi_0(x_0) & \varphi_1(x_0) & \varphi_2(x_0) & \ldots & \varphi_n(x_0) \\ \varphi_0(x_1) & \varphi_1(x_1) & \varphi_2(x_1) & \ldots & \varphi_n(x_1) \\ \varphi_0(x_2) & \varphi_1(x_2) & \varphi_2(x_2) & \ldots & \varphi_n(x_2) \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \varphi_0(x_n) & \varphi_1(x_n) & \varphi_2(x_n) & \ldots & \varphi_n(x_n) \end{array}\right| \neq 0 \Leftrightarrow \textbf{Solución única}$$

Luego en el caso general la existencia de solución y su unicidad, no está garantizada.

2.1.1 Problema clásico

Dados unos puntos $x_0, x_1, x_2, \ldots, x_n$ y sus aplicados: $f(x_0)$, $f(x_1)$, $f(x_2)$, ..., $f(x_n)$, buscamos encontrar una función polinómica del menor grado posible (inferior o igual a n), que pase por esos n+1 puntos. Queremos que:

$$\left.\begin{array}{ll} P(x_0) & = y_0 \\ P(x_1) & = y_1 \\ P(x_2) & = y_2 \\ \vdots & \vdots \\ P(x_{n-1}) & = y_{n-1} \\ P(x_n) & = y_n \end{array}\right\} \Rightarrow \left.\begin{array}{lllll} a_0 & + a_1 x_0 & + a_2 x_0^2 & + \ldots + a_n x_0^n & = y_0 \\ a_0 & + a_1 x_1 & + a_2 x_1^2 & + \ldots + a_n x_1^n & = y_1 \\ a_0 & + a_1 x_2 & + a_2 x_2^2 & + \ldots + a_n x_2^n & = y_2 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ a_0 & + a_1 x_{n-1} & + a_2 x_{n-1}^2 & + \ldots + a_n x_{n-1}^n & = y_{n-1} \\ a_0 & + a_1 x_n & + a_2 x_n^2 & + \ldots + a_n x_n^n & = y_n \end{array}\right\}$$

Como primer método (muy poco eficiente) para calcular el polinomio interpolador, puede considerarse la solución del sistema.

Para que la solución sea única, el determinante del sistema debe ser distinto de cero.

$$\Delta = \begin{vmatrix} 1 & x_0 & x_0^2 & \ldots & x_0^n \\ 1 & x_1 & x_1^2 & \ldots & x_1^n \\ 1 & x_2 & x_2^2 & \ldots & x_2^n \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & x_n & x_n^2 & \ldots & x_n^n \end{vmatrix} = \{\text{Al ser de Vandermonde}\} = \prod_{j<i} (x_i - x_j) \neq 0$$

Luego si los puntos son distintos, resulta que **la solución es única**, en el caso polinomico. Es decir, **existe un único polinomio interpolador de grado n, a n+1 puntos** (x_i, y_i), si los x_i son diferentes entre si.

Veamos ahora métodos eficientes de interpolación polinómica.

2.1.2 Método de Lagrange

Buscamos un polinómio de grado menor o igual a n, que verifique $P(x_i) = y_i$, $0 \leq i \leq n$.

Buscaremos primero un polinomio $P_l(x)$, de grado como máximo n, tal que valga 0 en todos los x_i excepto en el x_l en el que vale 1. Estos son los **Polinomios de Lagrange**, en particular éste será el l-ésimo. Su fórmula es conocida y vale:

$$\boldsymbol{P_l(x) = \prod_{i=0, i\neq l}^{i=n} \frac{(x - x_i)}{(x_l - x_i)}}$$

Conocidos los $\boldsymbol{P_l}$, el polinomio interpolador pedido valdrá:

$$\boldsymbol{P(x) = \sum_{l=0}^{n} P_l(x) y_l}$$

Ejemplo:

Hallar un polinomio de grado 3 que verifique $P(0) = 1$, $P(1) = 2$, $P(2) = 5$, $P(3) = 1$.

$P(x) = \frac{(x-1)(x-2)(x-3)}{(0-1)(0-2)(0-3)}1 + \frac{(x-0)(x-2)(x-3)}{(1-0)(1-2)(1-3)}2 + \frac{(x-0)(x-1)(x-3)}{(2-0)(2-1)(2-3)}5 + \frac{(x-0)(x-1)(x-2)}{(3-0)(3-1)(3-2)}1$

2.1.3 Interpolación recurrente

Generalmente no se conoce el grado del polinomio a interpolar, el método de Lagrange no permite incrementar el grado sin repetir todos los cálculos.

La interpolación recurrente busca realizar la interpolación a m puntos, a partir de la interpolación realizada anteriormente a m-1 puntos.

Diferencias divididas

Sea P_{m-1} el polinomio de interpolación (colocación) de $x_0, x_1, x_2, \ldots, x_{m-1}$ y sea P_m el de interpolación de $x_0, x_1, x_2, \ldots x_{m-1}, x_m$. Llamamos:

$$q_m(x) = p_m(x) - p_{m-1}(x) = \Delta_m(x - x_0)(x - x_1)\ldots(x - x_{m-1}) = \Delta_m \prod_{k=0}^{k=m-1} (x - x_k)$$

A los $\Delta_i = f[x_0, x_1, x_2, \ldots, x_i]$ se les llama diferencias divididas.

La recurrencia se comienza mediante: $f[x_0] = y_0$ y el polinomio interpolador se obtiene mediante:

$$P(x) = \sum_{i=0}^{m} f[x_0, x_1, x_2, \ldots, x_i] \prod_{j=0}^{j=i-1} (x - x_j) =$$

$$= f[x_0] + f[x_0, x_1](x - x_0) + \ldots + f[x_0, x_1, \ldots, x_m](x - x_0)(x - x_1)\ldots(x - x_{m-1})$$

PROPIEDADES:

Las diferencias divididas son simétricas:

$$f[x_0, x_1, \ldots, x_i, \ldots, x_j, \ldots, x_m] = f[x_0, x_1, \ldots, x_j, \ldots, x_i, \ldots, x_m]$$

La diferencia dividida para k+1 puntos, puede calcularse a partir de las de k puntos, mediante la fórmula:

$$f[x_0, x_1, \ldots, x_k] = \frac{f[x_0, x_1, \ldots, x_{k-1}] - f[x_1, x_2, \ldots, x_k]}{x_0 - x_k} \qquad (k \geq 1)$$

DISPOSICION DE CALCULOS:

Los cálculos de las sucesivas diferencias divididas se disponen usualmente en una tabla, donde las dos primeras columnas son los puntos y las restantes son las diferencias divididas, dispuestas en orden creciente.

$$\begin{array}{llllll}
x_0 & y_0 & & & & \\
 & & f[x_0, x_1] & & & \\
x_1 & y_1 & & f[x_0, x_1, x_2] & & \\
 & & f[x_1, x_2] & & f[x_0, x_1, x_2, x_3] & \\
x_2 & y_2 & & f[x_1, x_2, x_3] & & \ldots \\
 & & f[x_2, x_3] & & \ldots & \\
x_3 & y_3 & & \ldots & & \\
 & & \ldots & & & \\
\ldots & \ldots & & & &
\end{array}$$

Si se añade un punto más, basta añadirlo a la tabla y calcular una diferencia dividida más de cada orden.

EJEMPLO:

Hallar el polinomio que pasa por (2,1), (3,8) (4,5) (7,0).

$$\begin{array}{llll} 2 & 1 & & \\ & & 7=\frac{8-1}{3-2} & \\ 3 & 8 & & -5=\frac{-3-7}{4-2} \\ & & -3=\frac{5-8}{4-3} & \\ 4 & 5 & & \frac{1}{3}=\frac{-\frac{5}{3}+3}{7-3} \\ & & -5/3=\frac{0-5}{7-4} & \\ 7 & 0 & & \end{array} \qquad \begin{array}{l} \\ \\ \frac{16}{15}=\frac{\frac{1}{3}+5}{7-2} \\ \\ \\ \end{array}$$

Luego el polinomio interpolador resulta: $P(x) = 1 + 7(x-2) - 5(x-2)(x-3) + 16/15(x-2)(x-3)(x-4)$

EVALUACION DEL POLINOMIO DE INTERPOLACION:

Un algoritmo eficiente es el de **Horner generalizado**, que consta de los siguientes pasos:
1) Calcular las diferencias divididas: $\Delta_i = f[x_0, x_1, \ldots, x_n]$ y conocidos los x_i.
2) Hacemos: $b_n := \Delta_n$
3) Para k:=n-1 hasta 0 hacer: $b_k := (x - x_k) * b_{k+1} + \Delta_k$
4) El polinomio evaluado en el punto x: $P(x) := b_0$

NOTA: Si en el paso usamos un x concreto el resultado será el aplicado del punto x, mientras que si empleamos un x genérico el resultado será el polinomio interpolador, que como sabemos es único, es decir debe resultar el mismo valor que si se calcula por Lagrange, etc.

INTERPOLACION MEDIANTE DIFERENCIAS FINITAS:

En el caso particular de que los x_i estén en progresión aritmética, pueden usarse técnicas más eficientes:

Llamamos **diferencia progresiva**: $\Delta f_k = y_{k+1} - y_k; \ldots; \Delta^{i+1} f_k = \Delta(\Delta^i f_k)$

Llamamos **diferencia regresiva**: $\nabla f_k = y_k - y_{k-1}; \ldots; \nabla^{i+1} f_k = \nabla(\nabla^i f_k)$

Obtniendose entre otras, las siguientes fórmulas interpolatorias:

Fórmula de Newton progresiva: $P(x) = \sum_{i=0}^{n} \frac{\Delta^i f_0}{i! h^i} \prod_{j=0}^{i-1}(x - x_j)$

Fórmula de Newton regresiva: $P(x) = \sum_{i=0}^{n} \frac{\nabla^i f_0}{i! h^i} \prod_{j=0}^{i-1}(x - x_{n-j})$

Ejemplo:

Hallar el polinomio de interpolación a los puntos: (0,2), (1,3), (2,5) y (3,9)

$$\begin{array}{llll} x_i & y_i & \Delta y & \Delta^2 y \quad\quad \Delta^3 y \\ 0 & \mathbf{2} & & \\ & & \mathbf{1} = 3-2 & \\ 1 & 3 & & \mathbf{1} = 2-1 \\ & & 2 = 5-3 & \mathbf{1} = 2-1 \quad \Rightarrow \\ 2 & 5 & & 2 = 4-2 \\ & & 4 = 9-5 & \\ 3 & 9 & & \end{array}$$

$P(x) = 2 + 1(x-0) + \frac{1}{2!1^2}(x-0)(x-1) + \frac{1}{3!1^3}(x-0)(x-1)(x-2)$

Podríamos seguir otro orden para obtener P(x):

$P(x) = 2 + 1(x-3) + \frac{2}{2!1^2}(x-3)(x-2) + \frac{1}{3!1^3}(x-3)(x-2)(x-1)$

En ambos casos su desarrollo conduce a: $P(x) = \frac{1}{6}x^3 + \frac{5}{6}x + 2$

2.1.4 Error de interpolación

Sean $(x_i, f(x_i))$, $\forall i \in \{0, 1, \dots, n\}$ distintos entre sí y P(x) el polinomio interpolador a esos puntos, entonces:

$$E(x) = f(x) - P(x) = f[x_0, x_1, x_2, \dots, x_n, x] \prod_{i=0}^{n} (x - x_i)$$

Teorema:

Si $f \in \mathcal{C}^{n+1)}[a, b]$, (intervalo cerrado que contiene a los puntos x_i, y al punto a evaluar x), y x_i, $\forall i \in \{0, 1, \dots, n\}$, entonces:

$$E(x) = \frac{f^{n+1}(\xi)}{(n+1)!}(x - x_0)(x - x_1)\dots(x - x_n) \text{ con } \xi \in (x_0, x_n)$$

Corolario:

Si $[a, b]$ es un intervalo cerrado que contiene a los puntos x_i y al punto a evaluar x, y si M=$\max_{\xi \in [a,b]} |f^{n+1}(\xi)|$ entonces:

$$|E(x)| \leq \frac{M}{(n+1)!} \max_t |\prod_i (t - x_i)|, \text{ con } t \in [a, b]$$

NOTAS:

- Si los x_i están igualmente espaciados, el error de interpolación aumenta mucho hacia los extremos del intervalo de interpolación, empeorandose la situación, si aumentamos el grado del polinomio, introduciendo nuevos puntos, por lo que a partir de determinado grado (pequeño), no conviene este tipo de interpolación.
- Si podemos elegir los puntos x_i donde realizar las observaciones, debemos optar por puntos que reduzcan el valor de $\max_t |\prod_i (t - x_i)|$, ya que el otro factor en la expresión del error depende de la función a interpolar y no es controlable.
- La mejor situación de los puntos es un problema que está estudiado y su situación coincide en $[-1, 1]$ con los ceros de los polinómios de Tshebysheff.
- En el caso general de un intervalo $[a, b]$, la situación óptima de los x_i es:

$$x_i = \frac{a+b}{2} + \frac{b-a}{2} \cos\left(\frac{2i-1}{2n}\pi\right) \qquad \{k = 1, .., n\}$$

Extrapolación e intepolación inversa

La **extrapolación** consiste en estimar el valor de la función f(x) cuando el punto $x \notin [x_0, x_n]$, aunque para ello se emplean las mismas fórmulas que para interpolación.

El principal inconveniente es que el error aumenta, pues el factor $\prod_i (x - x_i)$ se incrementa rapidamente al alejarnos de los puntos de interpolación, invalidando los resultados.

Dada una tabla de valores $\{(x_i, y_i)\}$, la **interpolación inversa** consiste en hallar valores de x tales que verifiquen $f(x) = \alpha$, para un α dado.

Método 1: Si los y_i, son distintos puedo interpolar una función (polinómica o no) de la forma x=g(y) a (y_i, x_i) y calcular x para $y = \alpha$ $(x = g(\alpha))$.

Método 2: Interpolar una función y=f(x) a los valores dados y luego calcular $f^{-1}(\alpha)$. (Generalmente resulta imposible evaluar la inversa).

Método 3: Interpolamos una función (polinómica o no) y=f(x) a los valores dados y luego resolvemos la ecuación $f(x) - \alpha = 0$

Método 4: Empleamos cualquier método de resolución numérica (bipartición, newton, secante, ...), estimando el aplicado de x por la función desconocida mediante algún método de interpolación.

2.2 Interpolación osculatoria: Métodos de Hermite y Hermite cúbico

Hemos visto que añadiendo más puntos de interpolación (**subtabulación**), no siempre reducimos el error cometido, sino al contrario.

Para reducir errores debemos usar otras técnicas, como añadir otro tipo de condiciones (usualmente sobre derivadas en los x_i), o emplear otros tipos de funciones no polinómicas.

Se llama **interpolación osculatoria** cuando se le imponen condiciones a la función no solo de pasar por unos puntos, sino de tener determinados valores en las derivadas en esos puntos

2.2.1 Método de diferencias divididas

Se calculan mediante diferencias divididas, considerendo que un punto se encuentra repetido tantas veces, como derivadas conocidas en él.

Se tiene también en cuenta:
$f[x_i, x_i, \overset{(k}{\ldots}, x_i] = \frac{f^{(k-1}(x_i)}{(k-1)!}$
Se demuestra para k=2, mediante:

$$f[x_i, x_i] = \lim_{h \to 0} f[x_i, x_i + h] = \lim_{h \to 0} \frac{f(x_i + h) - f(x_i)}{h} = f'(x_i)$$

Y en el caso general comparando las 2 expresiones del error:

$$E(x) = \frac{f^{n+1}(\xi)}{(n+1)!}(x - x_0)(x - x_1)\ldots(x - x_n)$$

$$E(x) = f[x_0, x_1, x_2, \ldots, x_n, x] \prod_{i=0}^{n} (x - x_i)$$

considerando que los puntos son todos iguales.

Los calculos se disponen de igual forma que para diferencias divididas.

Ejemplo. Ajustar un polinomio a $F(0) = 0,\ F'(0) = 2,\ F''(0) = 2,\ F(1) = 1,\ F'(1) = 0$

$$
\begin{array}{cccccc}
x_i & y_i & \Delta y & \Delta^2 y & \Delta^3 y & \\
\mathbf{0} & \mathbf{0} & & & & \\
 & & \mathbf{2} & & & \\
\mathbf{0} & \mathbf{0} & & \mathbf{1} & & \\
 & & \mathbf{2} & & -2 & \\
\mathbf{0} & \mathbf{0} & & -1 & & 2 \\
 & & 1 & & 0 & \\
\mathbf{1} & \mathbf{1} & & -1 & & \\
 & & \mathbf{0} & & & \\
\mathbf{0} & \mathbf{0} & & & &
\end{array}
\Rightarrow P(x) = 0 + 2x + 1x^2 - 2x^3 + 2x^3(x-1)
$$

2.2.2 Método de Hermite:

Es un caso particular de interpolación osculatoria mediante polinomios, cuando en todos los puntos de interpolación, deben cumplirse las condiciones de pasar por ellos y tener un valor prefijado en la primera derivada, sin existir otro tipo de condiciones.

2.2.3 Interpolación no polinómica

Se trata ahora de disminuir el error de interpolación mediante el empleo de funciones no polinómicas.

Al comienzo vimos como podíamos interpolar funciones en un espacio vectorial general, conocida un base $B = \{\varphi_0, \varphi_1, \dots, \varphi_n\}$.

Si $B = \{1, sen(x), cos(x), sen(2x), cos(2x), sen(3x), cos(3x), \dots\}$, será un tipo importante de interpolación trigonométrica, pero podrá ser también la base $B = \{1, e^x, e^{2x}, \dots\}$, o cualquier otra. El método de resolución del sistema no es eficiente, pero a veces es el único existente.

Sabemos que la interpolación mediante polinomios de alto grado, produce oscilaciones que inutilizan el resultado por lo que podemos recurrir a funciones pseudo-polinomicas, es decir que coinciden con un polinomio en un intervalo, pero que cambian de polinomio al cambiar de intervalo. La más simple es la interpolación lineal en una tabla, que consiste en encontrar el intervalo $[x_i, x_{i+1}]$, en el que se encuentra el punto x a evaluar, realizando con esos dos puntos una interpolación lineal.

Exceptuando la interpolación lineal por su simplicidad, las más empleadas son las cúbicas, en cada intervalo se calcula el aplicado de x, mediante la evaluación de un polinomio de tercer grado.

2.2.4 Interpolación Hermite cúbica

Al igual que antes con Hermite, la información de que se dispone son los puntos por los que debe pasar y las derivadas primeras en dichos puntos, pero ahora en vez de un polinomio de alto grado que cumpla todas las condiciones, se busca una función definida a intervalos donde los polinomios que la forman tienen como máximo grado 3. Se calcula empleando interpolación osculatoria en cada intervalo.

Ejemplo: Interpolar un polinomio a los datos: $f(0) = 1$, $f'(0) = 1$, $f(2) = 0$, $f'(2) = 1$, $f(3) = 1$, $f'(3) = 4$.

$$\begin{array}{cccccc} 0 & 1 & & & & \\ & & 1 & & & \\ 0 & 1 & & -0.75 & & \\ & & -0.5 & & 0.75 & \\ 2 & 0 & & 0.75 & & \\ & & 1 & & & \\ 2 & 0 & & & & \end{array}$$

Primer intervalo [0,2]: $\Rightarrow P_1(x) = 1+x-0.75x^2+0.75x^2(x-2)$

$$\begin{array}{cccccc} 2 & 0 & & & & \\ & & 1 & & & \\ 2 & 0 & & 0 & & \\ & & 1 & & 3 & \\ 3 & 1 & & 3 & & \\ & & 4 & & & \\ 3 & 1 & & & & \end{array}$$

Segundo intervalo [2,3]: $\Rightarrow P_2(x) = 0+1(x-2)+0(x-2)^2+3(x-2)^2(x-3)$

La función resultante será:

$$H_3(x) = \begin{cases} 1+x-0.75x^2+0.75x^2(x-2) & \text{si } x \in [0,2] \\ 0+1(x-2)+0(x-2)^2+3(x-2)^2(x-3) & \text{si } x \in [2,3] \end{cases}$$

2.3 INTERPOLACION MEDIANTE SPLINES

Se denomina **spline de grado k**, a funciones (pseudo-polinómicas) definidas a intervalos, k-1 veces derivables y coincidentes en cada intervalo con polinomios de grado máximo k.

Si k=3 los splines se llaman splines cúbicos y son los más usados por consideraciones de tipo físico. Debido a su importancia veremos fórmulas simplificadas (eficientes) para su cálculo, mientras que para los otros se podrían calcular mediante las condiciones de spline.

INTERPOLACION POR SPLINES CÚBICOS:

Dados n+1 puntos $\{x_0, x_1, \ldots, x_n\}$ un spline cubico de interpolación será de la forma:

$$s(x) = \begin{cases} a_0+b_0(x-x_0)+c_0(x-x_0)^2+d_0(x-x_0)^3 & x \in [x_0,x_1) \\ a_1+b_1(x-x_1)+c_1(x-x_1)^2+d_1(x-x_1)^3 & x \in [x_1,x_2) \\ a_2+b_2(x-x_2)+c_2(x-x_2)^2+d_2(x-x_2)^3 & x \in [x_2,x_3) \\ \ldots & \ldots \\ a_{n-1}+b_{n-1}(x-x_{n-1})+c_{n-1}(x-x_{n-1})^2+d_{n-1}(x-x_{n-1})^3 & x \in [x_{n-1},x_n] \end{cases}$$

TIPOS DE SPLINES CÚBICOS

Como vemos en la expresión anterior, dados n+1 puntos, el spline consta de 4n coeficientes. Sin embargo las condiciones a verificar son:

n+1: por pasar por los n+1 puntos de interpolación, que se denominan **nodos**.

n-1: por la continuidad en los n-1 nodos intermedios.

n-1: por la derivabilidad en los n-1 nodos intermedios.

n-1: por la existencia de la segunda derivada en los n-1 nodos intermedios.

Luego resultan 4n-2 condiciones y 4n coeficientes, por lo que existen 2 grados de libertad (2 condiciones adicionales que podemos poner libremente). De la forma que pongamos estas condiciones llamadas **condiciones de frontera**, tendremos el tipo de spline utilizado.

2.3.1 Spline forzado

Las condiciones frontera son la segunda derivada en los extremos: $f''(x_0) = \alpha$ y $f''(x_n) = \beta$. En el caso de que $f''(x_0) = f''(x_n) = 0$ se llamará **spline natural** o de frontera libre.

Para calular los coeficientes del spline forzado se resuelve el sistema Ac=k, de n+1 ecuaciones con n+1 incognitas:

$$A = \begin{pmatrix} 1 & 0 & 0 & 0 & \dots & 0 & 0 \\ h_0 & 2(h_0+h_1) & h_1 & 0 & \dots & 0 & 0 \\ 0 & h_1 & 2(h_1+h_2) & h_2 & \dots & 0 & 0 \\ 0 & 0 & h_2 & 2(h_2+h_3) & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \dots & 2(h_{n-2}+h_{n-1}) & h_{n-1} \\ 0 & 0 & 0 & 0 & \dots & 0 & 1 \end{pmatrix}$$

$$k = \begin{pmatrix} \frac{f''(x_0)}{2} \\ 3(f[x_1,x_2] - f[x_0,x_1]) \\ 3(f[x_2,x_3] - f[x_1,x_2]) \\ \vdots \\ 3(f[x_{n-2},x_{n-1}] - f[x_{n-3},x_{n-2}]) \\ 3(f[x_{n-1},x_n] - f[x_{n-2},x_{n-1}]) \\ \frac{f''(x_n)}{2} \end{pmatrix}$$

donde $h_i = x_{i+1} - x_i$ y $f[x_i, x_j]$ indica la primera diferencia dividida de x_i y x_j.

Después se calculan los restantes coeficientes mediante:

$$a_i = f(x_i) \qquad d_i = \frac{c_{i+1} - c_i}{3h_i} \qquad b_i = f[x_i, x_{i+1}] - \frac{h_i}{3}(2c_i - c_{i+1})$$

$i = \{0, 1, \dots, n-1\}$

2.3.2 Spline completo o de frontera sujeta

Las condiciones de frontera serán valores conocidos para la primera derivada en los puntos extremos ($f'(x_0) = \alpha$ y $f'(x_n) = \beta$). Se resuelve de forma similar $Ac = k$ con:

$$A = \begin{pmatrix} 2h_0 & h_0 & 0 & 0 & \dots & 0 & 0 \\ h_0 & 2(h_0+h_1) & h_1 & 0 & \dots & 0 & 0 \\ 0 & h_1 & 2(h_1+h_2) & h_2 & \dots & 0 & 0 \\ 0 & 0 & h_2 & 2(h_2+h_3) & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \dots & 2(h_{n-2}+h_{n-1}) & h_{n-1} \\ 0 & 0 & 0 & 0 & \dots & h_{n-1} & 2h_n \end{pmatrix}$$

$$k = \begin{pmatrix} 3(f[x_0, x_1] - f'(x_0)) \\ 3(f[x_1, x_2] - f[x_0, x_1]) \\ 3(f[x_2, x_3] - f[x_1, x_2]) \\ \vdots \\ 3(f[x_{n-2}, x_{n-1}] - f[x_{n-3}, x_{n-2}]) \\ 3(f[x_{n-1}, x_n] - f[x_{n-2}, x_{n-1}]) \\ 3(f'(x_n) - f[x_{n-1}, x_n]) \end{pmatrix}$$

calculandose los a_i, b_i y d_i, por las mismas fórmulas del spline forzado.

2.3.3 Otros tipos de splines cúbicos

Otras condiciones de frontera empleadas son:

a) Conocer las derivadas primera y segunda en uno de los extremos.

b) Existencia de la derivada tercera en x_1 y en x_{n-1}. Equivale a quitar 2 nodos.

c) Peridicidad: Las derivadas primeras y segunda coinciden en los extremos. $f'(x_0) = f'(x_n)$ y $f''(x_0) = f''(x_n)$.

2.4 Aproximación por mínimos cuadrados

El objetivo de la aproximación es encontrar una función más simple que las obtenidas por interpolación, que se ajuste a unos datos, pero ahora las condiciones no tienen que verificarse estrictamente

Dada una función f de un conjunto E (Ej. funciones reales) y un subconjunto $U \subset E$, (Ej. funciones polinómicas grado acotado), la aproximación intenta determinar el elemento de U que sea el más próximo a f.

1. Todo espacio normado es métrico. Pues $d(x, y) = \|x - y\|$.
2. Si U es compacto entonces existe al menos una mejor aproximación para todo elemento de E. En R, R^n los compactos son los subconjuntos cerrados y acotados.
3. Si U es un subespacio vectorial de E (dimensión finita), entonces existe la mejor aproximación.
4. La condición necesaria y suficiente para que $u \in U$ sea la mejor aproximación a $f \in E$ (U subespacio de dimensión finita y E dotado de un producto escalar), es que para todo $v \in U, < f - u, v >= 0$. (Donde $< f, g >$ indica el producto escalar de las funciones f y g).

2.4.1 Cálculo de la mejor aproximación

Sea $\{v_i\}$ una base de U, entonces $u = \sum_i \lambda_i v_i \Rightarrow < f - \sum_i \lambda_i v_i, v_j >= 0 \forall j$, luego:

$$\sum_i \lambda_i < v_i, v_j >=< f, v_j > \qquad \forall j$$

NOTA: Si la base de U fuese ortogonal $< v_i, v_j >= 0$ si $i \neq j$ quedando:

$$< v_i, v_i > \lambda_i =< f, v_i > \qquad \forall i$$

Como generalmente U es el conjunto de las funciones polinómicas de grado inferior o igual a n, se usan bases de polinomios ortogonales, que se encuentran precalculadas.

OTRA FORMA: Dada la base $\{v_i\}$ de U, todo vector $u \in U$ puede expresarse como: $u = \sum_i \lambda_i v_i$, y la distancia entre f y u dependerá de los λ_i, $d(f,u) = \|f-u\| = \sqrt{< f-u, f-u >} = F(\lambda_1, \ldots, \lambda_n)$ y para minimizarla la condición necesaria es que: $\frac{\partial F}{\partial \lambda_i} = 0$, $\forall i$, aunque para ello se necesita que la función distancia sea derivable.

2.4.2 Aproximación mínimo cuadrática

El producto escalar que conduce a la norma cuadrática es:

CASO CONTINUO: $< f, g >= \int_a^b w(x) f(x) g(x) dx$

CASO DISCRETO: $< f, g >= \sum_i w(x_i) f(x_i) g(x_i)$

Donde w(x) es llamada la **función peso** y debe ser positiva en [a,b] (caso continuo) y para todo x_i, en el caso discreto.

Ejemplo 1:

Hallar el polinomio de segundo grado mejor aproximación a $f(x) = 1 + e^x$ en [0,2].

En este caso w(x)=1. $u = a + bx + cx^2$

Las ecuaciones normales son:

$$\left.\begin{array}{llll} a\int_0^2 11dx & +\int_0^2 x1dx & +\int_0^2 x^2 1dx & =\int^2(1+e^x)1dx \\ a\int_0^2 1xdx & +\int_0^2 xxdx & +\int_0^2 x^2 xdx & =\int^2(1+e^x)xdx \\ a\int_0^2 1x^2dx & +\int_0^2 xx^2dx & +\int_0^2 x^2x^2dx & =\int^2(1+e^x)x^2dx \end{array}\right\}$$

Ecuaciones que nos proporcionan a, b y c.

El segundo método consiste en expresar $F = \int_0^2((1+e^x) - (a+bx+cx^2))^2 dx$ y plantear el sistema resultante de calcular: $\frac{\partial F}{\partial a} = 0$, $\frac{\partial F}{\partial b} = 0$ y $\frac{\partial F}{\partial c} = 0$,

Ejemplo 2: Encontrar la parabola $y = a + bx + cx^2$ que mejor se aproxima a los puntos (0,1), (2,3) (3,3) (5,5) (6,4).

$$\left.\begin{array}{llll} a\sum_i 1\,1 & +b\sum_i x_i\,1 & +c\sum_i x_i^2\,1 & =\sum_i y_i\,1 \\ a\sum_i 1\,x & +b\sum_i x_i\,x_i & +c\sum_i x_i^2\,x_i & =\sum_i y_i\,x_i \\ a\sum_i 1\,x_i^2 & +b\sum_i x_i\,x_i^2 & +c\sum_i x_i^2\,xi^2 & =\sum_i y_i\,x_i^2 \end{array}\right\} \Rightarrow \left.\begin{array}{rcrcrcr} 5a & + & 16b & + & 74c & = & 16 \\ 16a & + & 74b & + & 376c & = & 64 \\ 74a & + & 376b & + & 2018c & = & 308 \end{array}\right\}$$