Pre-parcial Análisis Numérico

Santiago Chaparro y Briam Agudelo

August 2018

1 Formalización

1.1 Entradas

Para un sistema de ecuaciones de forma matricial de la forma $AX = B$:
A: matriz cuadrada (nxn) de números decimales, representa los índices del sistema de ecuaciones.
B: Vector de tamaño de (nx1) de números decimales, representa el vector resultado del sistema de ecuaciones.
X_0**:** Vector de tamaño de (nx1) de números decimales, representa la aproximación a la solución del sistema descrito anteriormente.

1.2 Salidas

Teniendo en cuenta la ecuación descrita anteriormente:
X**:** vector que representa el resultado de la ecuación.

2 Diseño

2.1 Fundamentos lógico matemáticos

Para el método iterativo de Jacobi con el que se pretende hallar una solución a un sistema de ecuaciones de la forma $AX = B$(forma matricial).
En donde de forma extendida tendríamos:

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} & \dots & a_{1n} \\ a_{21} & a_{22} & a_{23} & \dots & a_{2n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & a_{n3} & \dots & a_{nn} \end{bmatrix} * \begin{bmatrix} x_0 \\ x_1 \\ \ddots \\ x_n \end{bmatrix} = \begin{bmatrix} b_0 \\ b_1 \\ \ddots \\ b_n \end{bmatrix}$$

El método parte de un X_0 el cual es una solución aproximada(semilla del método), de esta forma si el sistema de ecuaciones converge(este punto se tratará más adelante) se debe escoger un número n de iteraciones para hallar la solución más aproximada. Este proceso iterativo tiene la siguiente ecuación en donde se calcula el vector solución del sistema de ecuaciones de la siguiente iteración a partir del calculado en la iteración anterior.

$$x_i^{(k+1)} = \frac{1}{a_{ii}}(b_i - \sum_{j \neq i} a_{ij} x_j^{(k)}), i = 1, 2, 3, ...$$

O de forma matricial tenemos:

$$X^{(k+1)} = TX^{(k)} + C$$

Donde $T = D^{-1}(D - A)$ y $C = D^{-1}B$

NOTA: Para la implementación del método se hizo uso de esta última ecuación(2) ya que R facilita las operaciones entre matrices.

2.2 Deducción de la ecuación implementada

Para llegar a la ecuación (2) partimos de $AX = B$:

Si descomponemos A como:

$A = (D + R)$

Donde D=Matriz diagonal de A y R=L+U, donde L=Matriz triangular inferior y U=Matriz triangular superior, obtenemos.

$(D + R)X = B$
$DX + RX = B$
$DX = B - RX$

Multiplicamos por la inversa de de la matriz diagonal para despejar X(Vector solución).

$D^{-1}DX = D^{-1}B - D^{-1}RX$
$X = D^{-1}B - D^{-1}RX$

Sabiendo que A=D+R, tenemos que R=A-D y reemplazando en la ecuación anterior tenemos que:

$X = D^{-1}B - D^{-1}(A - D)X$
$X = D^{-1}B + D^{-1}(D - A)X$
$X = D^{-1}(D - A)X + D^{-1}B$

Si decimos que $C = D^{-1}B$ y $T = D^{-1}(D - A)$ llegamos a la ecuación que pretendíamos deducir.

2.3 Fundamentos e implementación del criterio de convergencia

Teniendo en cuenta que D, corresponde a la matriz diagonal de A, que D^-1 es la inversa de esta misma matriz diagonal; y que R es la diferencia entre la diagonal y la matriz original ($R = D - A$). Se puede utilizar la siguiente formula para comprobar la convergencia del método con una matriz A en particular: $p(D^-1 * R) < 1$. Una vez realizado $D^-1 * R$, se debe hallar el radio espectral (p). Esto se hace sacando el máximo de los valores propios de $D^-1 * R$, luego se revisa si este valor corresponde a un numero menor a uno. Si satisface esta ultima condición, se puede realizar el método de Jacobi.

3 Pseudocódigo

Algorithm 1 Función que devuelve una solución cercana a las ecuación $Ax = b$

procedure JACOBI($t, x0, b, aDiagonal, iteraciones$)
    $cont \leftarrow 0$
    $diagonalInversa \leftarrow hallarInversa(aDiagonal)$
    while $cont < iteraciones$ **do**
        $r \leftarrow t * x0$
        $c \leftarrow diagonalInversa * b$
        $x \leftarrow r + c$
        $x0 \leftarrow x$
        $cont \leftarrow cont + 1$
        return (x0)

Algorithm 2 Función que verifica si la matriz converge y luego halla una solucion x que satisfaga la ecuacion $Ax = b$

procedure ALGORITMOPRINCIPAL()

$x0 \leftarrow [1, 1]$

$b \leftarrow [11, 13]$

$a \leftarrow \begin{bmatrix} 2 & 1 \\ 5 & 7 \end{bmatrix}$

$aDiagonal \leftarrow matrizDiagonal(a)$

$t \leftarrow hallarInversa(aDiagonal) * (aDiagonal - a)$

$valoresPropios \leftarrow hallarInversa(t)$

if $valorMaximo(valoresPropios) < 1$ **and** $valorMaximo(valoresPropios * (-1)) < 1$ **then**

$iteraciones \leftarrow 25$

$x \leftarrow jacobi(t, x0, b, aDiagonal, iteraciones)$

$imprimir("ELMÉTODOCONVERGEALVECTORSOLUCIÓN...")$

$imprimir(jacobi(t, x0, b, aDiagonal, iteraciones))$

else

$imprimir("ELMÉTODONOCONVERGEPARAESTESISTEMADEECUACIONES")$