

Solución de Matrices Tridiagonales

Diego Barajas y Brandonn Cruz

August 2018

1 Problema

Resolver un sistema de ecuaciones, el cual puede ser representado por una matriz tridiagonal de la forma:

$$\begin{array}{cccccc|l} b_1 & c_1 & 0 & 0 & 0 & & |x_1 = d_1 \\ a_1 & b_2 & c_2 & 0 & 0 & & |x_2 = d_2 \\ & 0 & a_2 & b_3 & \ddots & 0 & |x_3 = d_3 \\ & 0 & 0 & \ddots & \ddots & c_{n-1} & |x_{n-1} = d_{n-1} \\ & 0 & 0 & 0 & a_{n-1} & b_{n-1} & |x_n = d_n \end{array}$$

2 Formalización

2.1 Entradas

Matriz tridiagonal M, representada por los vectores A, B, C y D. Tal que $|A| = n - 1$, $|B| = n$, $|C| = n - 1$ $|D| = n$. Siendo n el número de incógnitas del sistema de ecuaciones.

Cada vector tiene la forma:

$$A = [a_0, a_1, a_2, \dots, a_{n-1},]$$

$$B = [b_0, b_1, b_2, \dots, b_n]$$

$$C = [c_0, c_1, c_2, \dots, c_{n-1}]$$

$$D = [d_0, d_1, d_2, \dots, d_n]$$

Cada elemento $e \in$ a cada vector V, debe cumplir: $e \in R$.

2.2 Salidas

Solución al sistema de ecuaciones representado por la matriz tridiagonal. Es decir un vector R de tamaño n ($|R| = n$).

3 Deducción de la fórmula

La deducción del algoritmo de Thomas se da de la siguiente forma:

Se comienza con un sistema de ecuaciones de entrada de la siguiente forma:

$$\begin{aligned} b_1 x_1 + c_1 x_2 &= d_1 \\ a_2 x_1 + b_2 x_2 + c_2 x_3 &= d_2 \\ a_3 x_2 + b_3 x_3 + c_3 x_4 &= d_3 \\ a_4 x_3 + b_4 x_4 &= d_4 \end{aligned} \quad (1)$$

Los a_i se denominan subdiagonal de coeficientes, los b_i como la diagonal de coeficientes, y c_i como super diagonal de coeficientes. Los índices i indican el número de la fila. Lo que se hace a continuación es realizar operaciones de fila que permitan convertir la subdiagonal de coeficientes en 0, y transformar los coeficientes de la diagonal en 1, de la forma:

$$\begin{aligned} x_1 + c'_1 x_2 &= d'_1 \\ x_2 + c'_2 x_3 &= d'_2 \\ x_3 + c'_3 x_4 &= d'_3 \\ x_4 &= d'_4 \end{aligned} \quad (2)$$

Para convertir el sistema de ecuaciones (1) a (2) es necesario encontrar una expresión para cada c'_i y d'_i . Al comparar los sistemas de ecuaciones, es posible notar que $c'_1 = c_1/b_1$ y $d'_1 = d_1/b_1$. Para obtener la segunda fila del sistema de ecuaciones (2) se elimina x_1 entre la primera fila de (2) y la segunda fila de (1). La primera fila de (2) es multiplicada por a_2 y esta se le resta a la segunda fila de (1). Generando la ecuación:

$$(b_2 - a_2 c'_1) x_2 + c_2 x_3 = d_2 - a_2 d'_1 \quad (3)$$

Con (3) se obtiene la ecuación:

$$x_2 + (c_2 / (b_2 - a_2 c'_1)) x_3 = (d_2 - a_2 d'_1) / (b_2 - a_2 c'_1) \quad (4)$$

La ecuación (4) ahora está en la forma de la segunda fila del sistema de ecuaciones (2). Luego $c'_2 = c_2/(b_2 - a_2 c'_1)$ y $d'_2 = (d_2 - a_2 d'_1)/(b_2 - a_2 c'_1)$. Continuando con las demás filas, se identifica que la relación es:

$$(b_i - a_i c'_{i-1})$$

(5)

$$c'_i = c_i / (b_i - a_i c'_{i-1})$$

(6)

$$d'_i = (d_i - a_i d'_{i-1}) / (b_i - a_i c'_{i-1})$$

(7)

La solución del sistema es entonces la sustitución desde $x_4 = d'_4$ hacia atrás, hasta llegar a la primera fila. En general $x_n = d'_n$, donde n es el número de ecuaciones, y para cada i < n:

$$x_i = d'_i - c'_i x_{i+1}$$

(8)

Esta deducción de la fórmula ha sido parafraseada del documento del Profesor C. Espinoza (Ver el apartado de Guía de librerías usadas).

4 Pseudocódigo

Algorithm 1 Algoritmo de Thomas

```
0: procedure tridiagonales( $a, b, c, d, cifras$ )
1:  $c[1] < -c[1]/b[1]$ 
2:  $d[1] < -d[1]/b[1]$ 
3: for  $i < -2to|c|$  do
4:    $c[i] < -c[i]/(b[i] - (c[i-1] * a[i-1]))$ 
5: end for
6: for  $i < -2to|d|$  do
7:    $d[i] = (d[i] - (d[i-1] * a[i-1]))/(b[i] - (c[i-1] * a[i-1]))$ 
8: end for
9:  $x = \phi$ 
10: for  $ito|d|$  do
11:    $x = x0$ 
12: end for
13:  $x[|d|] = d[|d|]$ 
14: for  $i < -|d| - 1to - 1step - 1$  do
15:    $x[i] = (d[i] - (c[i] * x[i+1]))$ 
16:    $x[i] = round(x[i], cifras)$ 
17: end for
18: return  $x$ 
     $\equiv 0$ 
```

Algorithm 2 Algoritmo de comprobar la solución

```
0: procedure converge( $x, a, b, c, d$ )
1: for  $i < -1to|d|$  do
2:   if  $((a[i] * x[i-1]) + (b[i] * x[i]) + (c[i] * x[i-2])) \neq d[i]$  then
3:     return  $-1$ 
4:   end if
5: end for
6: return  $0$ 
     $\equiv 0$ 
```

5 Condiciones de convergencia

Para la convergencia de este método, se requiere que:

- Se reciba una matriz tridiagonal.
- Cualquier matriz hermítica puede convertirse en una matriz tridiagonal mediante una transformación ortogonal usando el algoritmo de Lanczos.

(Wikipedia, 2018)

Para saber si la respuesta arrojada por el algoritmo es correcta, se comprueba que reemplazando los valores x_i dados por el algoritmo en el sistema de ecuaciones original de como resultado d_i . Es decir, se comprueba las siguientes igualdades:

$$a_1x_{-1} + b_1x_1 + c_2x_2 = d_1$$

$$a_2x_1 + b_2x_2 + c_3x_3 = d_2$$

...

$$a_ix_{i-1} + b_ix_i + c_{i+1}x_{i+1} = d_i$$

Nótese que se hace referencia a c y d , no a c' y d' .

6 Guía de librerías usadas

Para el desarrollo de este documento no se usó ninguna librería de python, lo único que se usó fue la documentación del algoritmo de Thomas encontrada en la wikipedia y en los documentos *Algoritmo de Thomas para matrices Tridiagonales* del profesor C. Espinoza; *Álgebra lineal con aplicaciones y Python* de Ernesto Aranda.

7 Consistencia

Gracias a que se conocía la complejidad del algoritmo de thomas de antemano, siendo esta de $O(n)$ y al conocer que la forma en la que se resuelven las matrices tridiagonales es con un metodo directo, se tenía la hipotesis de que el algortimo es igual de consistente con cualquier problema bien planteado que se le presente. Al realizarse pruebas con el algortimo de thomas a diferentes tamaños de matrices tridiagonales desde matrices de tamaño 5 hasta matrices de tamaño 1000000 se concluyó que la hipotesis es veridica.

8 Aplicación en un problema

Algunas de las aplicaciones de los sistemas de ecuaciones lineales es la resolución de ecuaciones diferenciales. Por ejemplo:

La distribución de temperatura de una varilla unidimensional de longitud 1 que

no está sometida a ninguna fuente de calor, y en cuyos extremos existe una temperatura dada por los valores de las condiciones iniciales $u(0) = 0$, $u(1) = 0$. Si se añade una fuente de calor sobre la varilla, medida a través de la función $f(x) = 4\pi^2 \sin(5\pi x)$, el modelo matemático que se proporciona cómo se distribuye el calor en la varilla viene dado de la siguiente forma:

$$\begin{aligned} -u''(x) &= f(x) = 4\pi^2 \sin(5\pi x), \quad x \in [0, 1] \\ u(0) &= 0 = u(1) \\ (9) \end{aligned}$$

Para ciertas funciones es posible calcular analíticamente la solución de esta ecuación diferencial, pero en general puede ser difícil, o incluso imposible, encontrar una expresión explícita para u . En estos casos es necesario acudir al Análisis Numérico. En primer lugar hay que rebajar la intención de encontrar la solución como una función exacta, con infinitos valores en el intervalo $[0, 1]$, y buscar en su lugar un número finito de valores de la misma; esto es lo que se conoce como discretización.

Para ello se va a tomar una cantidad $n = 5$ de nodos y $h = 1/(n + 1)$, el denominado paso de discretización, quedando:

$$\begin{aligned} x_i &= ih, \quad 0 \leq i \leq n + 1 \\ x_i &= ih, \quad 0 \leq i \leq 6 \\ (10) \end{aligned}$$

Obsérvese que $x_0 = 0 < x_1 < x_2 < \dots < x_n < x_{n+1} = 1$. La intención es encontrar los valores de la solución de (9) en estos nodos. Por ejemplo, ya se sabe que $u(x_0) = 0$ y $u(x_{n+1}) = 0$, por tanto queda encontrar $u(x_i)$, $1 \leq i \leq n$, es decir, $n = 5$ valores que serán incógnitas. Por otra parte, parece evidente pensar que cuantos más puntos consideremos en la discretización mejor será la aproximación de la solución que obtengamos.

El siguiente paso consiste en escribir la ecuación $-u''(x) = f(x)$ en función de esos valores. La cuestión es cómo evaluar u'' . Es conocido que el valor de la derivada en un punto x se puede aproximar por cocientes incrementales del tipo:

$$u'(x) \approx (u(x+t) - u(x))/t \approx (u(x-t) - u(x))/t \approx (u(x+t) - u(x-t))/(2t)$$

De forma similar se puede obtener una aproximación de la derivada segunda:

$$u''(x) \approx (u(x+t) - 2u(x) + u(x-t))/t^2$$

Estas expresiones son más precisas cuanto más pequeño es t . Si ponemos $t = h$ en la última expresión y evaluamos en un nodo x_i , queda:

$$u''(x_i) \approx (u(x_{i+1}) - 2u(x_i) + u(x_{i-1}))/h^2$$

debido a la definición que se ha tomado de los nodos. Con objeto de simplificar la notación pondremos $u(x_i) = u_i$, y de forma similar $f(x_i) = f_i$.

A continuación se sustituye la ecuación en x por la evaluación de la misma en

cada uno de los nodos obteniendo el siguiente problema discreto:

$$-(u_{i+1} - 2u_i + u_{i-1})/h^2 = f_i, 1 \leq i \leq n \quad (11)$$

Dado que $u_0 = 0$, $u_6 = 0$ y $f_i, 1 \leq i \leq n$ son valores conocidos, estamos ante un sistema lineal de n ecuaciones con n incógnitas.

Si escribimos una a una las ecuaciones de (11) resulta:

$$-u_0 + 2u_1 - u_2 = h^2 f_1$$

$$-u_1 + 2u_2 - u_3 = h^2 f_2$$

$$-u_2 + 2u_3 - u_4 = h^2 f_3$$

$$-u_3 + 2u_4 - u_5 = h^2 f_4$$

$$-u_4 + 2u_5 - u_6 = h^2 f_5$$

Escribiendo en forma matricial queda:

$$\begin{array}{cccccccl} 2 & -1 & 0 & 0 & 0 & u_1 & = & 4\pi^2 \sin(5\pi 1) + 0 \\ -1 & 2 & -1 & 0 & 0 & u_2 & = & 4\pi^2 \sin(5\pi 2) \\ 0 & -1 & 2 & -1 & 0 & u_3 & = & 4\pi^2 \sin(5\pi 3) \\ 0 & 0 & -1 & 2 & -1 & u_4 & = & 4\pi^2 \sin(5\pi 4) \\ 0 & 0 & 0 & -1 & 2 & u_5 & = & 4\pi^2 \sin(5\pi 5) + 0 \end{array}$$

Que corresponde a una matriz tridiagonal que el algoritmo de Thomas puede resolver.

Este ejemplo ha sido adaptado del libro de Ernesto Aranda (Ver el apartado de Guía de librerías usadas).