Informe Tarea 6

Anderson Daniel Grajales Alzate

15 de octubre de 2018

1. Solución numérica de sistemas de ecuaciones lineales

Muchos problemas del mundo real se formulan como sistemas de ecuaciones de n variables y m incógnitas que bajo condiciones ideales(n y m no son valores muy grandes), se pueden resolver de manera analítica. Sin embargo, cuando n y m tienden a ser valores muy grandes la solución analítica a estos problemas es muy difícil de calcular ya que requiere de mucho tiempo y claramente no es la forma más eficiente hacerlo. Debido a esto, desde el campo del *Análisis Numérico* se plantean diversas formas computacionales, ya sean algoritmos u otras técnicas que nos permitan resolver estos sistemas rápidamente, teniendo en cuenta que hay una ***propagación de error*** en cada cálculo dependiendo de la capacidad de la computadora donde se ejecuten estos. En este documento se presentan algunos algoritmos numéricos que son de gran ayuda a la hora de resolver sistemas de ecuaciones lineales.

1.1. Determinantes

Definición 1.1. *Sea $A = [a_{ij}]$ una matriz de tamaño $n \times n$. El cofactor C_{ij} de a_{ij} se define como $(-1)^{i+j} \det M_{ij}$, donde M_{ij} es la matriz de tamaño $(n-1) \times (n-1)$, que se obtiene al eliminar la fila i y la columna j de la matriz.[1]*

Teorema 1.1. *Sea $A = [a_{ij}]$ una matriz de tamaño $n \times n$. [1]*

- *Para cada $1 \leq i \leq n$ se cumple que:* $\det A = a_{i1}C_{i1} + a_{i2}C_{i2} + ... + a_{in}C_{in}$

- *Para cada* $1 \leq j \leq n$ *se cumple que:* $\det A = a_{1j}C_{1j}+a_{2j}C_{2j}+...+a_{nj}C_{nj}$

De acuerdo al teorema 1.1 se puede definir una ecuación de recurrencia para encontrar el determinante de una matriz $A = [a_{ij}]$ de la siguiente manera:

$$det(A,n)_{1\leq i\leq n} = \begin{cases} a_{11}, & \text{if } n = 1. \\ (-1)^{i+1} \times a_{1i} \times det(A', n-1), & \text{if } n > 1. \end{cases} \quad (1)$$

Donde A' es la matriz que se obtiene al eliminar la columna i y la fila n de A. De esta manera, para una matriz B de $n \times n$, la solución se entrega de la forma: $det(B, n)$.

1.2. Multiplicación de matrices

Definición 1.2. *Dadas las matrices* $A \in M_{m\times n}$ *y* $B \in M_{n\times p}$*, entonces el producto de* A *con* B*, denotado* AB*, es una matriz* $C \in M_{m\times p}$*, dada por: [1]*

$$c_{ij} = a_{i1}b_{1j} + a_{i2}b_{2j} + a_{i3}b_{3j} + \cdots + a_{in}b_{nj} = \sum_{k=1}^{n} a_{ik}b_{kj}$$

con $i = 1, ..., m$ *y* $j = 1, ..., p$

1.3. Escalonamiento de matrices

Definición 1.3. *Sea* $A = [a_{ij}]$ *una matriz de* $n \times n$*. Decimos que* A *está escalonada si* $\forall i, j, 1 \leq i \leq j \leq n$*,* $a_{ij} = 0$*.*

Leer A, b
si $A \notin \Re^{n\times n}$ ***ó*** $b \notin \Re^{n}$ **entonces**
  A debe ser cuadrada y b debe ser un arreglo de n posiciones
si no, si $\det A = 0$ **entonces**
  A debe ser invertible
en otro caso
  para $k = 1$ **a** $n-1$ **hacer**
    si $A_{kk} = 0$ **entonces**
      $j \leftarrow k+1$
      mientras $j < n$ ***y*** $A_{jk} = 0$ **hacer**
        $j \leftarrow j+1$
      fin
      si $j < n$ **entonces**
        para $l = k$ **a** n **hacer**
          $A_{kl} \leftarrow A_{kl} + A_{jl}$
        fin
        $b_k \leftarrow b_k + b_j$
      fin
    fin
    para $i = k+1$ **a** n **hacer**
      si $A_{ki} \neq 0$ **entonces**
        $m \leftarrow \frac{A_{ik}}{A_{kk}}$
        para $l = k$ **a** n **hacer**
          $A_{il} \leftarrow A_{il} - m \times A_{kl}$
        fin
        $b_i \leftarrow b_i - m \times b_k$
      fin
    fin
  fin
fin
La solución (A, b)

Algoritmo 1: Algoritmo para escalonar matrices

1.4. Eliminación Gaussiana Simple

Algoritmo 2: Algoritmo de Eliminación Gaussiana Simple

1.5. Sustitución Regresiva

Leer A, b
$marks \leftarrow NULL$
$n \leftarrow Len(A) - 1$
$x \leftarrow 0$
$x_n \leftarrow \frac{b_n}{A_{nn}}$
para $i \leftarrow n - 1$ **a** -1 **hacer**
  $sumatoria \leftarrow 0$
  para $p \leftarrow i + 1$ **a** $n + 1$ **hacer**
    $sumatoria \leftarrow sumatoria + A_{ip} * x_p$
  fin
  $x_i \leftarrow b_i$
fin
si $marks \neq NULL$ **entonces**
  $marcas[Len(A)] \leftarrow 0$
  para $i \leftarrow 0$ **a** $Len(A)$ **hacer**
    $marcas_{marks_i} \leftarrow x_i$
  fin
  Retornar $marcas$
Retornar x

Algoritmo 3: Algoritmo de Sustitución Regresiva

1.6. Sustitución Progresiva

Leer A, b
$n \leftarrow Len(A) - 1$
$x \leftarrow 0$
$x_0 \leftarrow \frac{b_0}{A_0 0}$
para $i \leftarrow 1$ **a** $n + 1$ **hacer**
  $sumatoria \leftarrow 0$
  para $p \leftarrow 0$ **a** i **hacer**
    $sumatoria \leftarrow sumatoria + A_{ip} * x_p$
  fin
  $x_i \leftarrow \frac{b_i - sumatoria}{A_{ii}}$
fin
Retornar x

Algoritmo 4: Algoritmo de Sustitución Progresiva

1.7. Eliminacion Gaussiana con Pivoteo Parcial

Leer A, b, n, k
$mayor \leftarrow |A_{kk}|$
$filaMayor \leftarrow k$
para $s \leftarrow k + 1$ **a** n **hacer**
  si $|A_{sk}| > mayor$ **entonces**
    $mayor \leftarrow |A_{sk}|$
    $filaMayor \leftarrow s$
fin
si $mayor = 0$ **entonces**
  El sistema no tiene solucion unica
si no, si $filaMayor \neq k$ **entonces**
  $temp \leftarrow A_{filaMayor}$
  $A_{filaMayor} \leftarrow temp$
  $A_k \leftarrow A_{filaMayor}$
  $temp \leftarrow b_{filaMayor}$
  $b_{filaMayor} \leftarrow b_k$
  $b_k \leftarrow b_{filaMayor}$
Retornar(A, b)

Algoritmo 5: Algoritmo de Pivoteo Parcial

1.8. Eliminacion Gaussiana con Pivoteo Total

Leer A, b, n, k
$mayor \leftarrow |A_{kk}|$
$filaMayor \leftarrow k$
$columnaMayor \leftarrow k$
para $r \leftarrow k$ **a** n **hacer**
  para $s \leftarrow k$ **a** n **hacer**
    si $|A_{rs}| > mayor$ **entonces**
      $mayor \leftarrow |A_{rs}|$
      $filaMayor \leftarrow r$
      $columnaMayor \leftarrow s$
  fin
fin
si $mayor = 0$ **entonces**
  El sistema no tiene solucion unica
en otro caso
  si $filaMayor \neq k$ **entonces**
    $temp \leftarrow A_k$
    $A_k \leftarrow A_{filaMayor}$
    $A_{filaMayor} \leftarrow temp$
    $temp \leftarrow b_k$
    $b_k \leftarrow b_{filaMayor}$
    $b_{filaMayor} \leftarrow temp$
  si $columnaMayor \neq k$ **entonces**
    $temp \leftarrow A_{0columnaMayor}$
    $A_{0columnaMayor} \leftarrow A_{0k}$
    $A_{0k} \leftarrow temp$
    $temp \leftarrow b_{0columnaMayor}$
    $b_{0columnaMayor} \leftarrow b_{0k}$
    $b_{0k} \leftarrow temp$
    $temp \leftarrow marcas_k$
    $marcas_k \leftarrow marcas_{columnaMayor}$
    $marcas_{columnaMayor \leftarrow temp}$
  Retornar(A, b)

Algoritmo 6: Algoritmo de Pivoteo Total

2. Factorización LU para la solución de sistemas de ecuaciones lineales

Una forma práctica de resolver sistemas de ecuaciones lineales de manera eficiente es definiendo una matriz $A \in \Re^{n \times n}$ como el producto de dos matrices $L, U \in \Re^{n \times n}$, esto es, $A = LU$, donde U es una matriz superior diagonal y L es una matriz inferior diagonal. Dado lo anterior, podemos resolver sistemas de ecuaciones lineales de la siguiente manera:

$$
\begin{gathered}
Ab = x \\
x = LUb \\
Ab = LUb \\
z = Ub \\
Ab = Lz
\end{gathered}
$$

2.1. Eliminación gaussiana simple

Leer A, b
// Escalonar(A, b) devuelve la matriz inferior, superior y el nuevo
// vector b que resulta de hacer el proceso de escalonamiento de A
$L, U, b' = Escalonar(A, b)$
$z = SustiituionProgresiva(L, b')$
$x = SustitucionRegresiva(U, z)$
Retornar x

Algoritmo 7: Algoritmo de Gauss-LU

2.2. Método de Cholesky

Leer $A \in \Re^{n\times n}$
$L, U = 0^{n\times n}$
para $k = 1$ **a** n **hacer**
    $s^{(1)} \leftarrow 0$
    para $p = 1$ **a** k **hacer**
        $s^{(1)} \leftarrow s^{(1)} + L_{k,p} \times U_{p,k}$
    fin
    $L_{k,k} \leftarrow \sqrt{A_{k,k} - s^{(1)}}$
    $U_{k,k} \leftarrow L_{k,k}$
    para $i = k$ **a** n **hacer**
        $s^{(2)} \leftarrow 0$
        para $p = 1$ **a** k **hacer**
            $s^{(2)} \leftarrow s^{(2)} + L_{i,p} \times U_{p,k}$
        fin
        $L_{i,k} \leftarrow \frac{A_{i,k} - s^{(2)}}{U_{k,k}}$
    fin
    para $j = k + 1$ **a** n **hacer**
        $s^{(3)} \leftarrow 0$
        para $p = 1$ **a** k **hacer**
            $s^{(3)} \leftarrow s^{(3)} + L_{k,p} \times U_{p,j}$
        fin
        $U_{k,j} \leftarrow \frac{A_{k,j} - s^{(3)}}{L_{k,k}}$
    fin
fin
Retorna L, U

Algoritmo 8: Algoritmo de Cholesky

Referencias

[1] Orlando García Jaimes, Jairo A. Villegas Gutiérrez, Jorge Iván. *Álgebra Lineal*. Editorial EAFIT, Medellín 2012.