Tarea 6 - Métodos numéricos
Giovanni Gamaliel López Padilla

Índice

1. Introducción

Una matriz A de m×n es un ordenamiento rectangular de m por n númros distribuidos en un orden definido de m filas y n columnas.

$$A = \begin{pmatrix} a_{11} & a_{12} & a_{13} & \cdots & a_{1n} \\ a_{21} & a_{22} & a_{23} & \cdots & a_{2n} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ a_{m1} & a_{m2} & a_{m3} & \cdots & a_{mn} \end{pmatrix} \tag{1}$$

donde a_{ij} es el i,j-ésimo elemento de A. Se define una matriz cuadrada si y solo si $m = n$. Una matriz diagonal es una matriz cuadrada, en la que los elementos a_{ij} son iguales a 0 para $i \neq j$ y es un escalar para $i = j$. Se define a una matriz triangular superior aquellas matrices cuadradas que sus elementos a_{ij} son iguales a cero para $i > j$. Para una matriz triangular inferior es el caso contrario, sus elementos a_{ij} son iguales a cero para $i < j$.
Una ecuación lineal es una expresión del tipo:

$$a_1x_1 + a_2x_2 + a_3x_3 + \cdots + a_nx_n = b \tag{2}$$

donde $x_{1,2,\ldots,n}$ se denominan como variables, $a_{1,2,\ldots,n}$ son los coeficientes de cada termino y b es un término constante. Un sistema de ecuaciones lineales es un conjunto de m ecuaciones lineales en las cuales contienen n variables.

$$\begin{cases} a_{11}x_{11} + a_{12}x_{12} + \cdots + a_{1n}x_{1n} = b_1 \\ a_{21}x_{21} + a_{22}x_{22} + \cdots + a_{2n}x_{2n} = b_2 \\ \vdots \\ a_{m1}x_{m1} + a_{m2}x_{m2} + \cdots + a_{mn}x_{mn} = b_m \end{cases} \tag{3}$$

Las operaciones definidas sobre las matrices y sobre las ecuaciones son idénticas, por lo tanto, es posible realizar la escritura de un sistema de ecuaciones en forma matricial. Cada elemento de la matriz A (ecuación 1) correspondera a un coeficiente del sistema de ecuaciones (ecuación 3), donde cada fila representará a una ecuación lineal.

$$\begin{pmatrix} a_{11}x_{11} + a_{12}x_{12} + \cdots + a_{1n}x_{1n} = b_1 \\ a_{21}x_{21} + a_{22}x_{22} + \cdots + a_{2n}x_{2n} = b_2 \\ \vdots \\ a_{m1}x_{m1} + a_{m2}x_{m2} + \cdots + a_{mn}x_{mn} = b_m \end{pmatrix} \rightarrow \begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \vdots & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_m \end{pmatrix} = \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_m \end{pmatrix} \tag{4}$$

La parte derecha de la ecuación 4 puede escribirse como $AX = B$, esta es llamada la ecuación matricial del sistema.

2. Métodos

2.1. Matriz diagonal

Suponiendo que se tiene un sistema de ecuaciones de n ecuaciones donde los coeficientes son iguales a cero excepto cuando el la posición del coeficiente coincide con el número de la ecuación. Entonces representando el sistema de ecuaciones antes descrito en una ecuación matricial se obtiene lo siguiente:

$$\begin{pmatrix} a_{11} & 0 & \cdots & 0 \\ 0 & a_{22} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & a_{nn} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix} = \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{pmatrix}$$

realizando la multiplicación de matrices se obtiene que:

$$\begin{pmatrix} a_{11}x_1 \\ a_{22}x_2 \\ \vdots \\ a_{nn}x_n \end{pmatrix} = \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{pmatrix}$$

por igualación de términos se obtiene que las soluciones del sistema de ecuaciones son descritas como:

$$x_i = \frac{b_i}{a_{ii}}$$

Se dice que el sistema no tiene solución cuando alguno de los elementos de la diagonal (a_{ii}) es igual a cero. El algoritmo planteado para realizar la solución a este tipo de sistema de ecuaciones es el siguiente:

```
// input: matriz, vector_b
// output: solutions
for(i = 1; i <= n; i++)
{
    valid_solution(matriz[i][i])
    solutions[i] = vector_b[i] / matriz[i][i]
}
```

La impremertación de este problema se encuentra en la carpeta Problema_1. La función solve_diagonal_matrix esta contenido en el archivo solution.h

2.2. Matriz triangular superior

Suponiendo que se tiene un sistema de ecuaciones de n ecuaciones donde los coeficientes son iguales a cero cuando la posición del coeficiente es menor a el número de la ecuación. Entonces representando el sistema de ecuaciones antes descrito en una ecuación matricial se obtiene lo siguiente:

$$\begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ 0 & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & a_{nn} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix} = \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{pmatrix}$$

realizando la multiplicación de matrices de obtiene que:

$$\begin{pmatrix} a_{11}x_1 + a_{12}x_2 + \cdots + a_{1n}x_n \\ a_{22}x_2 + a_{23}x_3 + \cdots + a_{2n}x_n \\ \vdots \\ a_{nn}x_n \end{pmatrix} = \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{pmatrix}$$

reduciendo esto a una expresión generalizada se obtiene que la solución de la i-ésima variable es:

$$x_i = \frac{b_i - \sum\limits_{j=i+1}^{n} a_{ij}x_j}{a_{ii}}$$

Se dice que el sistema de ecuaciones no tiene solución cuando algún elemento de la diagonal (a_{ii}) es igual a cero. El algoritmo planteado para realizar la solución a este tipo de ecuacioes es el siguiente:

```
// input: matriz, vector_b
// output: solutions
for(i = n; i <= 1; i--)
{
    sum_i = 0
    valid_solution(matriz[i][i])
    for(j = n; j <= i+1; j--)
    {
        sum_i += matriz[i][j]*solutions[j]
    }
    solutions[i] = (vector_b[i] - sum_i) / matriz[i][i]
}
```

La impremertación de este problema se encuentra en la carpeta Problema_2. La función solve_triangular_superior_matrix esta contenido en el archivo solution.h

2.3. Matriz triangular inferior

Suponiendo que se tiene un sistema de ecuaciones de n ecuaciones donde los coeficientes son iguales a cero cuando la posición del coeficiente es mayor a el número de la ecuación. Entonces representando el sistema de ecuaciones antes descrito en una ecuación matricial se obtiene lo siguiente:

$$\begin{pmatrix} a_{11} & 0 & \cdots & 0 \\ a_{21} & a_{22} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix} = \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{pmatrix}$$

realizando la multiplicación de matrices de obtiene que:

$$\begin{pmatrix} a_{11}x_1 \\ a_{21}x_1 + a_{22}x_2 \\ \vdots \\ a_{n1}x_1 + a_{n2}x_2 + \cdots + a_{nn}x_n \end{pmatrix} = \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{pmatrix}$$

reduciendo esto a una expresión generalizada se obtiene que la solución de la i-ésima variable es:

$$x_i = \frac{b_i - \sum\limits_{j=1}^{i-1} a_{ij}x_j}{a_{ii}}$$

Se dice que el sistema de ecuaciones no tiene solución cuando algún elemento de la diagonal (a_{ii}) es igual a cero. El algoritmo planteado para realizar la solución a este tipo de ecuacioes es el siguiente:

```
// input: matriz, vector_b
// output: solutions
for(i = 1; i <= n; i++)
{
    sum_i = 0
    valid_solution(matriz[i][i])
    for(j = 1; j <= i-1; j++)
    {
        sum_i += matriz[i][j]*solutions[j]
    }
    solutions[i] = (vector_b[i] - sum_i) / matriz[i][i]
}
```

La imprementación de este problema se encuentra en la carpeta Problema_3. La función solve_triangular_inferior_matrix esta contenido en el archivo solution.h

2.4. Eliminación Gaussiana

Suponiendo que se tiene un sistema de ecuaciones de n ecuaciones con n coeficientes, donde cada coeficiente es un número real. Entonces representando el sistema de ecuaciones como una ecuación matricial se obtiene lo siguiente:

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

Llamaremos a la matriz expandida aquella matriz de tamaño n×n+1, tal que:

$$\left(\begin{array}{cccc|c} a_{11} & a_{12} & \cdots & a_{1n} & b_1 \\ a_{21} & a_{22} & \cdots & a_{2n} & b_2 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} & b_n \end{array}\right) \tag{5}$$

Se realizaran operaciones entre renglones de la matriz extendida (5) de tal manera que se llegue a una matriz triangular superior. Se obtendra un término que llamaremos r_ij, el cual esta definido como en la ecuación 6.

$$r_{ji} = \frac{a_{ii}}{a_{ji}} \tag{6}$$

Donde la posición j, es el número de fila que será restada por la fila i. Por ejemplo, para iniciar se tomara a $i = 1$, y se restara la fila 2 ($j = 2$), esto para obtener que el elemento a_{21} sea igual a 0. Aplicando este proceso a la matriz 5 se obtiene el siguiente resultado:

$$\left(\begin{array}{cccc|c} a_{11} & a_{12} & \cdots & a_{1n} & b_1 r_{21} \\ a_{21}r_{21} & a_{22}r_{21} & \cdots & a_{2n}r_{21} & b_2 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} & b_n \end{array}\right)$$

Restando la linea 2 con la linea 1 se obtiene lo siguiente:

$$\left(\begin{array}{cccc|c} a_{11} & a_{12} & \cdots & a_{1n} & b_1 \\ a_{21}r_{21} - a_{11} & a_{22}r_{21} - a_{12} & \cdots & a_{2n}r_{21} - a_{1n} & b_2 r_{21} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} & b_n \end{array}\right)$$

Donde el termino $a_{21}r_{21} - a_{11} = 0$, renombrando a cada termino de la fila 2, como a^*_{2i}, se obtiene que:

$$\left(\begin{array}{cccc|c} a_{11} & a_{12} & \cdots & a_{1n} & b_1 \\ 0 & a^*_{22} & \cdots & a^*_{2n} & b^*_2 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} & b_n \end{array}\right)$$

repitiendo este proceso con las demás filas se llega a la siguiente matriz:

$$\left(\begin{array}{cccc|c} a_{11} & a_{12} & \cdots & a_{1n} & b_1 \\ 0 & a^*_{22} & \cdots & a^*_{2n} & b^*_2 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & a^*_{n2} & \cdots & a^*_{nn} & b^*_n \end{array}\right)$$

El proceso se repetira tomando el valor de cada diagonal de la siguiente fila y restando las filas que estan por debajo de esta, para asi obtener una matriz triangular superior y aplicar el método de solución antes descrito para este sistema de ecuaciones.
El algoritmo planteado para realizar la solución a este tipo de sistemas de ecuaciones es el siguiente:

```
// inputs: matriz, vector_b
// output: solutions
for(i = 1; i <= n-1; i++)
{
    a_ii = matriz[i][i]
    b_i = vector_b[i]
    for(j = i + 1; j <= n; j++)
    {
        a_ij = matriz[i][j]
        ratio_ji = a_ii / a_ij
        b_j = b_j * ratio_ji - b_i
        for(k = i; k < n,k++)
        {
            matriz_ik = matriz[i][k]
            matriz_jk = matriz_jk * ratio_ji - matriz_ik
        }
    }
}
solve_triangular_superior_matrix(matrix,vector_b)
```

La imprementación de este problema se encuentra en la carpeta Problema_4. La función solve_gaussian_matrix esta contenido en el archivo solution.h

3. Resultados

3.1. Matriz diagonal

El sistema de ecuaciones planteado en los archivos de prueba (M_DIAG.txt y V_DIAG.txt) corresponde a la siguiente ecuación matricial:

$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 \\ 0 & 0 & 3 & 0 \\ 0 & 0 & 0 & 4 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} = \begin{pmatrix} 1 \\ 2 \\ 3 \\ 4 \end{pmatrix}$$

Ejecutando el programa escrito, se obtiene los siguientes resultados:

```
--------------------------------------------------
Soluciones del sistema

x_1 = 1.000000
x_2 = 1.000000
x_3 = 1.000000
x_4 = 1.000000
```

3.2. Matriz triangular superior

El sistema de ecuaciones planteado en los archivos de prueba (M_TSUP.txt y V_TSUP.txt) corresponde a la siguiente ecuación matricial:

$$\begin{pmatrix} 1 & 2 & 3 & 4 \\ 0 & 5 & 6 & 7 \\ 0 & 0 & 8 & 9 \\ 0 & 0 & 0 & 10 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} = \begin{pmatrix} 1 \\ 2 \\ 3 \\ 4 \end{pmatrix}$$

Ejecutando el programa escrito, se obtiene los siguientes resultados:

```
 ---------------------------------------------------------------
  Soluciones del sistema

  x_1 = -0.141667
  x_2 = -0.116667
  x_3 = -0.075000
  x_4 = 0.400000
```

3.3. Matriz triangular inferior

El sistema de ecuaciones planteado en los archivos de prueba (M_TINF.txt y V_TINF.txt) corresponde a la siguiente ecuación matricial:

$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 2 & 3 & 0 & 0 \\ 4 & 5 & 6 & 0 \\ 7 & 8 & 9 & 10 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} = \begin{pmatrix} 1 \\ 2 \\ 3 \\ 4 \end{pmatrix}$$

Ejecutando el programa escrito, se obtiene los siguientes resultados:

```
 ---------------------------------------------------------------
  Soluciones del sistema

  x_1 = 1.000000
  x_2 = 0.000000
  x_3 = -0.166667
  x_4 = -0.150000
```

3.4. Matriz Eliminación Gaussiana

La solución del sistema de ecuaciones planteado en los archivos de prueba (M_LARGE.txt y V_LARGE.txt) es extenso para ser escrito en este reporte, es por ello que se encuentra en el archivo Solution_LARGE.txt, el cual esta contenido en la carpeta Problema_4.
Se probo con otro sistema de ecuaciones de menor tamaño para comprobar su solución. Este se encuentra en los archivos test_matrix.txt y test_result.txt, el cual es el siguiente:

$$\begin{pmatrix} 2 & 1 & -3 \\ 5 & -4 & 1 \\ 1 & -1 & -4 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} = \begin{pmatrix} 7 \\ -19 \\ 4 \end{pmatrix}$$

Ejecutando el programa escrito, se obtiene los siguientes resultados:

```
 ---------------------------------------------------------------
  Soluciones del sistema

  x_1 = -1.000000
  x_2 = 3.000000
  x_3 = -2.000000
```

4. Conclusiones

Los métodos de solución de las ecuaciones lineales en estos casos son de gran importancia, ya que problemas complicados pueden reducirse a alguno de los resueltos anteriormente. La implementación haciendo uso de apuntadores fue de vital importancia ya que sin ellos estos problemas serian complicados de implementarse como funciones dentro del programa. El método de eliminación gaussiana puede tener cambios, esto por medio de un cierto orden para llegar a una matriz triangular inferior o una matriz diagonal.

5. Compilación y comandos de ejecucción

Los programas contenidos en cada carpeta de los problemas fue compilado usando el siguiente comando:

```
gcc -Wall -Wextra -Werror -pedantic -ansi -o main.out main.c -std=c11
```

La forma de ejecutarse es con el siguiente comando:

```
./main.out matriz vector
```

donde matriz es el nombre de archivo que contiene la matriz y vector contiene el vector columna. Por ejemplo, para la matriz diagonal el comando seria:

```
./main.out M_DIAG.txt V_DIAG.txt
```

6. Referencias

José Arias. *Matrices y sistemas de ecuaciones lineales.* Universidad de Medellín, Medellín, Colombia, 2010. ISBN 9789588348612.

Stanley Grossman. *Álgebra lineal.* McGraw Hill, Ciudad de México, 2019. ISBN 978-1456272128.