Taller 2 – Sistemas de ecuaciones lineales

Heyling Burgos Algarin
Ingeniería de sistemas
Bogotá, Colombia
burgosaheyling@javeriana.edu.co

Andrés Fabián Olarte Vargas
Ingeniería de sistemas
Bogotá, Colombia
olarte_fabian@javeriana.edu.co

Johan Mateo Rosero Quenguan
Ingeniería de sistemas
Bogotá, Colombia
roseroq-j@javeriana.edu.co

Andrés Felipe Vásquez Rendón
Ingeniería de sistemas
Bogotá, Colombia
af.vasquezr@javeriana.edu.co

Introducción

En el presente documento se evaluarán tres puntos distintos sobre el tema de sistemas de ecuaciones lineales en las que se hallarán las soluciones más optimas de los mismos. De igual forma, se hará un estudio del análisis de los resultados y los diferentes cambios que se pueden hacer en cada uno.

Palabras Clave

→ **Gauss-Seidel**: consiste en hacer iteraciones, desde un valor inicial, para encontrar los valores a las incógnitas hasta llegar a una tolerancia deseada (López, 2016).
→ **Gauss con pivoteo parcial**: propone la eliminación progresiva de variables por medio de transformaciones elementales en el sistema de ecuaciones, hasta obtener solo una ecuación con una incógnita (Granada, 2015).
→ **Gauss**: consiste en transformar un sistema de ecuaciones lineal en otro pero escalonado (YoSoyTuProfe, 2016).
→ **Cramer:** la regla de Cramer es una fórmula para la solución de un sistema de ecuaciones lineales con tantas ecuaciones como incógnitas (Marta, 2021).
→ **Factorización LU** de una matriz A es el producto de una matriz triangular inferior L y una matriz triangular superior U cuyo resultado es igual a A (Rodriguez, 2019).
→ **Gradiente conjugado**: algoritmo cuyo fin es resolver sistemas de ecuaciones lineales, donde las matrices son simétricas (ZAPATA, 2018).

Punto 3

Suponga que en el siguiente modelo $f(x)$ describe la cantidad de personas que son afectadas por un virus, en donde t es el tiempo en días

$f(t) = k_1 t + k_2 t^2 + k_3 e^{0.15}$ Se conocen los siguientes datos: $f(10) = 25; f(15) = 130; f(20) = 650$

Determine de forma aproximada el día más cercano donde la cantidad de personas infectadas supera los 1500; 1800; 2000.

Para el desarrollo de este punto se usó el método de Gauss-Seidel que permitió encontrar los coeficientes de una función para un instante *t*, donde además se trabajó con una tolerancia de 10^-16 para encontrar la solución más precisa. Del mismo modo, se logró determinar una matriz A de 3x3 gracias a las funciones dadas. Esta matriz se desarrolló con las *x* evaluadas cambiando las t en la función original y los resultados fueron de gran ayuda para determinar la B. La *X* son las *k* respectivas de cada término de la función.

Luego de encontrar cada coeficiente, se realizó la función de Cant_infectados que determina el día más cercano superando una cantidad de personas establecidas. Su funcionamiento se consigue gracias a la *X* obtenida por el método de Gauss-Seidel que halla los coeficientes de la función real para luego reemplazarlas y obtener la matriz A. El arreglo B fue obtenido con las diferentes cantidades dadas por el enunciado y la *X* son días que superan dichas cantidades. Los diferentes resultados fueron:

Cantidad a superar	**Resultado de días más cercanos**
1500	25
1800	26
2000	26

Como se puede observar en la anterior tabla el resultado de días más cercano a la cantidad de 1500 es de 25 ya que el resultado que la iteración más próxima que lo alcanza a superar es de 1742.65243381890, donde notamos que no es suficiente para ser una cantidad más grande que 1800, por eso necesita una iteración más que es el 26 logrando un resultado de 2173.78011216999 abarcando las dos cantidades de 1800 y 2000.

Punto 8

Dados los sistemas del punto 1 (*figura 1*), evaluar el error hacia atrás, hacia delante y el número de condición cuando el sistema se soluciona por el método de: Gauss con pivoteo parcial, Gauss, Cramer, Factorización LU.

i.
$$\begin{aligned} u - 8v - 2w &= 1 \\ u + v + 5w &= 4 \\ 3u - v + w &= -2 \end{aligned}$$

ii.
$$\begin{aligned} u + 4v &= 5 \\ v + w &= 2 \\ 2u + 3w &= 0 \end{aligned}$$

iii.
$$\begin{aligned} u + 3v - w &= 18 \\ 4u - v + w &= 27.34 \\ u + v + 7w &= 16.2 \end{aligned}$$

Figura 1. Sistema de ecuaciones.

Para el desarrollo de este punto se tuvieron que implementar los 4 algoritmos descritos anteriormente, todo esto con el fin de obtener la solución, con la cual se logró implementar el error hacia atrás, el error hacia adelante y el número de condición. Cabe resaltar que la implementación de los 4 métodos se realizó con métodos directos, algunos con ayudas de librerías para realizar las operaciones como multiplicación de matrices, determinante, entre otras. A excepción de la factorización LU, en la cual se utilizó la librería scipy linalg, todo con el fin de obtener la matriz U y así poder realizar sustitución hacia atrás. Por lo cual no se usó una tolerancia especifica, además con la ayuda de algunas librerías se logró realizar el error hacia adelante, en el cual se debía saber el valor exacto de la solución. Una vez explicada la implementación se mostrarán los resultados que se obtuvieron.

Resultados.

- Matriz 1.

$$u - 8v - 2w = 1$$
$$u + v + 5w = 4$$
$$3u - v + w = -2$$

1) Gauss con pivoteo.

```
------- Metodo Gauss con pivoteo parcial ---------

Matriz inicial

 [[ 1. -8. -2.  1.]
 [ 1.  1.  5.  4.]
 [ 3. -1.  1. -2.]]

Matriz con pivoteo parcial

 [[ 3. -1.  1. -2.]
 [ 1. -8. -2.  1.]
 [ 1.  1.  5.  4.]]

Resultados:

 [[-1.2448979591836735], [-0.5714285714285715], [1.1632653061224492]]

Error relativo hacia atras:  9.930136612989092e-16

Error relativo hacia adelante es:  0.0

Numero de condicion:  5.277592057545204
```

2) Gauss.

```
------ Metodo Gauss  ---------

Matriz inicial

 [[ 1. -8. -2.  1.]
 [ 1.  1.  5.  4.]
 [ 3. -1.  1. -2.]]

Gauss:

 [[  1.          -8.          -2.           1.        ]
 [  0.          9.           7.           3.        ]
 [  0.          0.         -10.88888889 -12.66666667]]

Resultados:

 [[-1.2448979591836746], [-0.5714285714285716], [1.1632653061224492]]

Error relativo hacia atras:  3.1401849173675502e-15

Error relativo hacia adelante es:  1.1157603309187458e-15

Numero de condicion:  5.277592057545204
```

3) Cramer.

```
------ Metodo crammer ---------

Matriz inicial

 [[ 1. -8. -2.  1.]
 [ 1.  1.  5.  4.]
 [ 3. -1.  1. -2.]]

Resultados

 [[-1.244897959183672], [-0.5714285714285714], [1.1632653061224487]]

Error relativo hacia atras:  4.463041323674983e-15

Error relativo hacia adelante es:  1.6203171603697468e-15

Numero de condicion:  5.277592057545204
```

4) Factorización LU.

```
------ Metodo Factorización LU ---------

Matriz inicial

 [[ 1. -8. -2.  1.]
 [ 1.  1.  5.  4.]
 [ 3. -1.  1. -2.]]

Matriz L

 [[ 1.          0.          0.        ]
 [ 0.33333333  1.          0.        ]
 [ 0.33333333 -0.17391304  1.        ]]

Matriz U

 [[ 3.         -1.          1.         -2.        ]
 [ 0.         -7.66666667 -2.33333333  1.66666667]
 [ 0.          0.          4.26086957  4.95652174]]

Matriz LxU

 [[ 3. -1.  1. -2.]
 [ 1. -8. -2.  1.]
 [ 1.  1.  5.  4.]]

Resultados

 [[-1.2448979591836735], [-0.5714285714285715], [1.1632653061224492]]

Error relativo hacia atras:  9.930136612989092e-16
```

- **Matriz 2.**

$$\text{ii.}\quad \begin{aligned} u + 4v &= 5 \\ v + w &= 2 \\ 2u + 3w &= 0 \end{aligned}$$

1) Gauss con pivoteo.

```
------- Metodo Gauss con pivoteo parcial ---------

Matriz inicial

 [[1. 4. 0. 5.]
 [0. 1. 1. 2.]
 [2. 0. 3. 0.]]

Matriz con pivoteo parcial

 [[2. 0. 3. 0.]
 [1. 4. 0. 5.]
 [0. 1. 1. 2.]]

Resultados:

 [[-0.8181818181818181], [1.4545454545454546], [0.5454545454545454]]

Error relativo hacia atras:  0.0

Error relativo hacia adelante es:  0.0

Numero de condicion:  8.16361611659769
```

2) Gauss.

```
------ Metodo Gauss  ---------

Matriz inicial

 [[1. 4. 0. 5.]
 [0. 1. 1. 2.]
 [2. 0. 3. 0.]]

Gauss:

 [[ 1.   4.   0.   5.]
 [ 0.   1.   1.   2.]
 [ 0.   0. 11.   6.]]

Resultados:

 [[-0.8181818181818183], [1.4545454545454546], [0.5454545454545454]]

Error relativo hacia atras:  4.440892098500626e-16

Error relativo hacia adelante es:  2.220446049250313e-16

Numero de condicion:  8.16361611659769
```

3) Cramer.

```
------ Metodo crammer ---------

Matriz inicial

 [[1. 4. 0. 5.]
 [0. 1. 1. 2.]
 [2. 0. 3. 0.]]

Resultados

 [[-0.8181818181818186], [1.4545454545454548], [0.5454545454545456]]

Error relativo hacia atras:  1.0175362097255202e-15

Error relativo hacia adelante es:  5.438959822042073e-16

Numero de condicion:  8.16361611659769
```

4) Factorización LU.

```
------ Metodo Factorización LU ---------

Matriz inicial

 [[1. 4. 0. 5.]
 [0. 1. 1. 2.]
 [2. 0. 3. 0.]]

Matriz L

 [[1.   0.   0.  ]
 [0.5  1.   0.  ]
 [0.   0.25 1.  ]]

Matriz U

 [[ 2.     0.     3.     0.   ]
 [ 0.     4.    -1.5    5.   ]
 [ 0.     0.     1.375  0.75 ]]

Matriz LxU

 [[2. 0. 3. 0.]
 [1. 4. 0. 5.]
 [0. 1. 1. 2.]]

Resultados

 [[-0.8181818181818181], [1.4545454545454546], [0.5454545454545454]]

Error relativo hacia atras:  0.0

Error relativo hacia adelante es:  0.0

Numero de condicion:  8.16361611659769
```

- **Matriz 3.**

$$u + 3v - w = 18$$
iii. $4u - v + w = 27.34$
$$u + v + 7w = 16.2$$

1) Gauss con pivoteo.

```
------- Metodo Gauss con pivoteo parcial ---------

Matriz inicial

 [[ 1.    3.   -1.   18.  ]
 [ 4.   -1.    1.   27.34]
 [ 1.    1.    7.   16.2 ]]

Matriz con pivoteo parcial

 [[ 4.   -1.    1.   27.34]
 [ 1.    3.   -1.   18.  ]
 [ 1.    1.    7.   16.2 ]]

Resultados:

 [[7.58595744680851], [3.7051063829787227], [0.7012765957446806]]

Error relativo hacia atras:  6.153480596427404e-15

Error relativo hacia adelante es:  8.881784197001252e-16

Numero de condicion:  3.7678201908389206
```

2) Gauss.

```
------ Metodo Gauss  ---------

Matriz inicial

 [[ 1.    3.   -1.   18.  ]
 [ 4.   -1.    1.   27.34]
 [ 1.    1.    7.   16.2 ]]

Gauss:

 [[  1.           3.          -1.          18.        ]
 [  0.         -13.           5.         -44.66      ]
 [  0.           0.           7.23076923   5.07076923]]

Resultados:

 [[7.585957446808513], [3.7051063829787227], [0.7012765957446807]]

Error relativo hacia atras:  1.1234667099445444e-14

Error relativo hacia adelante es:  2.6668472207145996e-15

Numero de condicion:  3.76782019083892
```

3) Cramer.

```
------ Metodo crammer ---------

Matriz inicial

 [[ 1.     3.    -1.    18.   ]
 [ 4.    -1.     1.    27.34]
 [ 1.     1.     7.    16.2 ]]

Resultados

 [[7.510638297872339], [3.7234042553191506], [0.6808510638297 87]]

Error relativo hacia atras:  0.39446165846632386

Error relativo hacia adelante es:  0.08015602711376163

Numero de condicion:  3.76782019083892
```

4) Factorización LU.

```
Matriz inicial

 [[ 1.    3.   -1.   18.  ]
 [ 4.   -1.    1.   27.34]
 [ 1.    1.    7.   16.2 ]]

Matriz L

 [[1.         0.         0.        ]
 [0.25       1.         0.        ]
 [0.25       0.38461538 1.        ]]

Matriz U

 [[ 4.         -1.          1.         27.34      ]
 [ 0.          3.25       -1.25       11.165     ]
 [ 0.          0.          7.23076923  5.07076923]]

Matriz LxU

 [[ 4.   -1.    1.   27.34]
 [ 1.    3.   -1.   18.  ]
 [ 1.    1.    7.   16.2 ]]

Resultados

 [[7.58595744680851], [3.7051063829787227], [0.7012765957446806]]

Error relativo hacia atras:  6.153480596427404e-15

Error relativo hacia adelante es:  8.881784197001252e-16

Numero de condicion:  3.76782019083892
```

Punto 9

Dado un sistema cualquiera que está asociada a una matriz dispersa con n=10000 implemente el método del gradiente conjugado para resolver el problema.

Para la solución de este punto, se empleó una búsqueda sobre el significado de una matriz dispersa, la cual consiste en una matriz n x n, en la cual la mayoría de sus datos son ceros. Ya con el significado se pasó al desarrollo de método del gradiente conjugado, el cual permite la solución de matrices simétricas y de grandes tamaños, además de también ser útil para matrices dispersas.

Con esto en mente, se desarrolló en primera instancia un algoritmo el cual permite la creación de una matriz dispersa de n x n datos, en los cuales los datos que no son cero son valores asignados aleatoriamente entre 1 y 100, además, la matriz de valores resultantes también es creada de forma aleatoria entre estos mismos valores. Ya con la matriz dispersa creada, esta es enviada a una función que realiza propiamente el método del gradiente conjugado, el cual permite hallar solución de esta matriz.

```
Solucion: 92 : 79.77918763003944

Solucion: 93 : 80.04288106300044

Solucion: 94 : 80.45728974022526

Solucion: 95 : 79.6473470904134

Solucion: 96 : 80.77754249201323

Solucion: 97 : 81.09781953926237

Solucion: 98 : 79.81685711197615

Solucion: 99 : 81.36159535215178

Solucion: 100 : 80.89057867550129
```

Como se logra observar en la imagen anterior, se desarrolló el ejercicio para una matriz dispersa de 100x100, esto debido a que el método del gradiente conjugado es un método complejo, el cual requiere de muchas operaciones entre matrices, requiriendo así de muchos recursos computacionales, lo cual para matrices de 1000x1000 o incluso 500X500 implicaba en largos tiempos de ejecución antes de poder presentar la solución a la matriz; es por eso que se decidió realizar la prueba del método con esta matriz 100x100.

Referencias

Granada, S. (2015). *Procesos Numéricos.* Obtenido de Eliminación Gaussiana con Pivoteo Parcial: https://sites.google.com/site/trabajoprocesos20151/1-2-eliminacion-gaussiana-con-pivoteo-parcial

López, D. (30 de abril de 2016). *MÉTODO DE GAUSS - SEIDEL.* Obtenido de https://aprendeenlinea.udea.edu.co/lms/moodle/mod/page/view.php?id=24491

Marta. (4 de marzo de 2021). *Solución de sistemas de Cramer*. Obtenido de https://www.superprof.es/apuntes/escolar/matematicas/algebralineal/sistemas/regla-de-cramer.html

Rodriguez, A. (9 de enero de 2019). *El problema de la semana - Factorización LU*. Obtenido de https://www.nibcode.com/es/blog/1174/el-problema-de-la-semana-factorizacion-lu

YoSoyTuProfe. (1 de noviembre de 2016). *Método de Gauss | Teoría y ejercicios.* Obtenido de https://yosoytuprofe.20minutos.es/2016/11/01/metodo-de-gauss/

ZAPATA, E. A. (15 de mayo de 2018). *GRADIENTE CONJUGADO.* Obtenido de https://www.medellin.unal.edu.co/~cemejia/doc/nla/proyectos/METODO%20GRADIENTE%20CONJUGADO.pdf