

INFORME DE LABORATORIO 1 DE ALGORITMOS NUMÉRICOS:
MÉTODOS NUMÉRICOS

CATALINA MORALES ROJAS

Profesor: Óscar Rojas Díaz.

Santiago - Chile
10 de mayo de 2018

TABLA DE CONTENIDOS

ÍNDICE DE FIGURAS.....	vi
ÍNDICE DE CUADROS	vii
CAPÍTULO 1. INTRODUCCIÓN	9
CAPÍTULO 2. MÉTODOS DE RESOLUCIÓN DE ECUACIONES NO LI- NEALES.....	11
2.1 MÉTODO DE LA BISECCIÓN	11
2.2 MÉTODO DE NEWTON-RHAPSON	12
2.3 MÉTODO DE LA SECANTE	12
2.4 MÉTODO REGULA FALSI	13
2.5 MÉTODO DE NEWTON-RHAPSON MULTIVARIABLES	14
CAPÍTULO 3. MÉTODOS DE RESOLUCIÓN DE SISTEMAS DE ECUA- CIONES LINEALES	15
3.1 MÉTODOS ITERATIVOS	15
3.1.1 Método de Gauss-Jacobi	15
3.1.2 Método de Gauss-Seidel	15
3.2 MÉTODOS DIRECTOS	16
3.2.1 Factorización LU	16
3.2.2 Método Cholesky	17
3.2.3 Factorización QR	18
3.2.4 Método de Householder	18

3.2.5	Método de Givens	19
CAPÍTULO 4.	FUNCIONES.....	21
4.0.1	Regla de Fourier	21
CAPÍTULO 5.	RESULTADOS.....	25
5.1	RESULTADOS FUNCIÓN 1	25
5.1.1	Gráfico iteraciones	25
5.2	ITERACIONES FUNCIÓN 2	26
5.3	ITERACIONES FUNCIÓN 3	27
5.4	SISTEMA DE ECUACIONES CON NEWTON-RAPHSON	29
5.5	RESULTADOS RANKING MÉTODOS DE SOLUCIÓN DE SISTE- MAS DE ECUACIONES	30
5.5.1	Tiempo de ejecución	30
5.5.2	Error	30
5.5.3	Costos espaciales	30
5.5.4	Gráfico Tiempos Métodos	31
5.5.5	Gráfico Error Métodos	32
5.5.6	Gráfico Ranking	32
5.5.7	Gráfico Error Métodos	34
5.5.8	Gráfico Ranking	34
5.5.9	Gráfico Error Métodos	36
5.5.10	Gráfico Ranking	37
CAPÍTULO 6.	ANÁLISIS DE RESULTADOS.....	39
6.1	ANÁLISIS MÉTODOS DE RESOLUCIÓN DE ECUACIONES NO LI- NEALES	39
6.2	ANÁLISIS MÉTODO NEWTON-RAPHSON MULTIVARIABLE . . .	40
6.3	RANKING	40
CAPÍTULO 7.	CONCLUSIONES	43

CAPÍTULO 8. BIBLIOGRAFÍA	45
---------------------------------------	-----------

ÍNDICE DE FIGURAS

Figura 4-1: Regla de Fourier.	22
Figura 5-1: Iteraciones función 1.	25
Figura 5-2: Iteraciones función 1.	26
Figura 5-3: Iteraciones función 3.1	27
Figura 5-4: Iteraciones función 3.2	27
Figura 5-5: Tiempos matriz 289	32
Figura 5-6: Error matriz 289	32
Figura 5-7: Ranking sistema 289	33
Figura 5-8: Tiempos sistema 1089	34
Figura 5-9: Error sistema 1089	34
Figura 5-10: Ranking sistema 1089	35
Figura 5-11: Tiempo de ejecución Givens para sistema 1089	35
Figura 5-12: Tiempos sistema 4225	36
Figura 5-13: Error sistema 4225	37
Figura 5-14: Ranking sistema 4225	38

ÍNDICE DE CUADROS

Tabla 4.1: Valores iniciales para función 1.	22
Tabla 4.2: Valores iniciales para función 2.	23
Tabla 4.3: Valores iniciales para función 2.	23
Tabla 5.1: Tabla de resultados función 1	26
Tabla 5.2: Tabla de resultados función 2	26
Tabla 5.3: Tabla de resultados función 3	28
Tabla 5.4: Tabla de resultados sistema de ecuaciones 1	29
Tabla 5.5: Tabla de resultados sistema de ecuaciones 2	30
Tabla 5.6: Tabla de tiempos, errores y costos espaciales matriz 289x289	31
Tabla 5.7: Tabla de tiempos, errores y costos espaciales sistema 1089x1089 . .	33
Tabla 5.8: Tabla de tiempos,errores y costos espaciales sistema 4225x4225 . . .	36

CAPÍTULO 1. INTRODUCCIÓN

En la asignatura de algoritmos numéricos es posible estudiar diferentes métodos que permiten obtener soluciones numéricas a diversos problemas matemáticos. Un algoritmo corresponde a una serie de pasos ordenados que se utilizan para llegar a un resultado, los cuales pueden ser aplicados a la computación.

A continuación se presentan los fundamentos de los métodos que permiten resolver ecuaciones con una y varias variables (método de la bisección, regula falsi, secante, Newton Raphson, Newton Raphson multivariable, entre otros) y métodos que permiten encontrar las soluciones de un sistema de ecuaciones (LU, Cholesky, Gauss-seidel, Givens, Householder, Jacobi y QR). Cada uno de los métodos mencionados, cuenta con ventajas y desventajas, las cuales se pueden ver representadas por una serie de gráficos y tablas con sus costos temporales, espaciales, el error y el número de iteraciones. El objetivo del laboratorio e informe es entender el funcionamiento que tiene cada uno de los métodos, realizar una correcta implementación y comparar con respecto a una serie de criterios cuales son los mejores.

CAPÍTULO 2. MÉTODOS DE RESOLUCIÓN DE ECUACIONES NO LINEALES

Como se mencionó anteriormente, cada método cuenta tanto con ventajas como con desventajas, así como también similitudes entre ellos. En este capítulo se presenta una breve descripción del funcionamiento que tiene cada uno de los métodos utilizados durante el desarrollo del laboratorio.

Cada uno de los algoritmos cuenta con un error asociado, el cual corresponde a la diferencia existente entre el valor obtenido y el valor esperado. Es importante señalar que los algoritmos implementados terminan de iterar cuando el error del valor encontrado es menor a una tolerancia definida o cuando se llega al número máximo de iteraciones.

Dentro de los métodos de resolución de ecuaciones no lineales existen dos grandes grupos: Los métodos cerrados y los métodos abiertos. Los primeros contienen dentro de su intervalo de definición a la o las raíces, por lo cual siempre que el intervalo sea correcto, los métodos convergen, en cambio los métodos abiertos aproximan la raíz en cada iteración, por lo cual no es siempre probable su convergencia.

2.1 MÉTODO DE LA BISECCIÓN

El método de la bisección se encuentra basado en el teorema del valor medio, el cual indica que si una función es continua en el intervalo $[a, b]$ y el signo de $f(a)$ es distinto a $f(b)$ entonces existe un r perteneciente al intervalo tal que $f(r) = 0$. Es un método cerrado, lo que implica que si se cumple con lo postulado anteriormente, el método debe converger, es necesario contar con dos puntos para realizar el cálculo de la raíz. La fórmula para obtener el punto medio es:

$$r = \frac{(a + b)}{2} \quad (2.1)$$

Los pasos a seguir por este método son:

1. Verificar que $f(a) * f(b) < 0$, ya que esto implica que existe raíz.
2. Se calcula el punto medio del intervalo y se evalúa en la función, si el resultado es 0, se ha encontrado la raíz, de lo contrario se deben definir nuevamente los intervalos de la siguiente manera:

- El nuevo intervalo es $[a, r]$ si el signo de $f(a)$ es distinto al signo de $f(r)$.
 - El nuevo intervalo es $[r, b]$ si el signo de $f(b)$ es distinto al signo de $f(r)$.
3. Si r no corresponde a la raíz, se modifica el intervalo y se realiza nuevamente el proceso, hasta encontrar la raíz o que el error obtenido sea menor a la tolerancia $e < \text{tolerancia}$.

El método solo entrega una raíz del intervalo (aunque existan más), su convergencia es lineal, por lo cual se puede denominar como un método lento.

2.2 MÉTODO DE NEWTON-RHAPSON

El método de Newton-Rhapson es conocido por su eficacia y por ser un método abierto. Se necesita un punto para comenzar a iterar, el cual debe ser cercano a la raíz. La fórmula de iteración para este método es la siguiente:

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)} \quad (2.2)$$

Los pasos a seguir por este método son:

1. Se obtiene un punto mediante la ecuación presentada anteriormente.
2. Se evalúa el punto obtenido en la función $f(x_{n+1})$, si el resultado es cero significa que se ha encontrado la raíz, de lo contrario con el valor de x obtenido, se realiza nuevamente el algoritmo.

2.3 MÉTODO DE LA SECANTE

El método de la secante corresponde a una variación del método de Newton-Rhapson, ya que la función $f'(x)$ es cambiada por la definición de derivada, tal como se muestra en la siguiente ecuación de recurrencia:

$$x_{n+1} = x_n - \frac{x_n - x_{n-1}}{f(x_n) - f(x_{n-1})} * f(x_n) \quad (2.3)$$

Este método es un algoritmo de aproximación que usa las raíces de las líneas secantes para aproximar mejor la raíz de la función f . Se necesitan dos puntos para iniciar el algoritmo.

Los pasos a seguir por el método son:

1. Obtener ecuación de la recta que pasa por $(x_{n-1}, f(x_{n-1}))$ y $(x_n, f(x_n))$.
2. Se calcula el siguiente elemento con la función de recurrencia x_{n+1} , que es la intersección de la secante con el eje de las abscisas.
3. Se obtiene el nuevo valor de la fórmula y se repite el proceso hasta encontrar la raíz o que el error obtenido sea menor a la tolerancia $e < \text{tolerancia}$.

2.4 MÉTODO REGULA FALSI

Regula falsi corresponde a un método iterativo para encontrar la solución de ecuaciones no lineales. Combina el Método de la Bisección y el Método de la Secante.

Corresponde a un método cerrado, por lo cual utiliza dos puntos iniciales $[a, b]$, para que el método converja se debe cumplir que la multiplicación de los puntos evaluados en la función debe ser menor a cero $f(a) * f(b) < 0$.

En cada iteración se calcula c_n con la fórmula mostrada a continuación.

$$c_n = \frac{f(b_n) * a_n - f(a_n) * b_n}{f(b_n) - f(a_n)} \quad (2.4)$$

Los pasos a seguir por el método son:

1. Seleccionar el intervalo inicial $[a_0, b_0]$.
2. Calcular c_n con la fórmula presentada anteriormente.
3. Evaluar $f(c_n)$. Si es suficientemente pequeño, c_n es la raíz buscada.
4. Si no, el próximo intervalo $[a_{n+1}, b_{n+1}]$ será:
 - a) $[a_n, c_n]$ si $f(a_n)$ y $f(c_n)$ tienen signos opuestos.
 - b) $[c_n, b_n]$ si $f(b_n)$ y $f(c_n)$ tienen signos opuestos.
5. Se repite el algoritmo hasta encontrar la raíz o que el error obtenido sea menor a la tolerancia $e < \text{tolerancia}$.

2.5 MÉTODO DE NEWTON-RHAPSON MULTIVARIABLES

El método de Newton Raphson multivariantes permite obtener las soluciones de un sistema de ecuaciones. Para que este método sea convergente debe cumplir con:

1. El determinante de la matriz jacobiana generada por el sistema de ecuaciones debe ser distinto de cero: $\det(J(f_1, f_2)) \neq 0$
2. Las derivadas parciales segundas deben ser continuas.

La fórmula utilizada para por el método es:

$$\begin{pmatrix} x_1^{k+1} \\ x_2^{k+1} \end{pmatrix} = \begin{pmatrix} x_1^k \\ x_2^k \end{pmatrix} - \begin{pmatrix} \frac{df_1}{dx_1}(x_1^k, x_2^k) & \frac{df_1}{dx_2}(x_1^k, x_2^k) \\ \frac{df_2}{dx_1}(x_1^k, x_2^k) & \frac{df_2}{dx_2}(x_1^k, x_2^k) \end{pmatrix}^{-1} \begin{pmatrix} f_1(x_1^k, x_2^k) \\ f_2(x_1^k, x_2^k) \end{pmatrix} \quad (2.5)$$

CAPÍTULO 3. MÉTODOS DE RESOLUCIÓN DE SISTEMAS DE ECUACIONES LINEALES

Los métodos que se presentan a continuación entregan soluciones para sistemas de ecuaciones con la forma $Ax = b$ y se subdividen en dos grandes grupos: los métodos iterados y los métodos directos.

3.1 MÉTODOS ITERATIVOS

Los métodos iterativos se caracterizan por tener un comportamiento en relación a un vector de aproximaciones iniciales, el cual, tal como dice su nombre, comienza a iterar hasta llegar al resultado.

3.1.1 Método de Gauss-Jacobi

El método de Jacobi utiliza los valores iniciales para la primera aproximación, luego los resultados obtenidos para la segunda y así sucesivamente hasta encontrar su resultado. Para comenzar, el método necesita la matriz de coeficientes, las aproximaciones iniciales, el vector de términos independientes, la tolerancia y el número máximo de iteraciones. La fórmula iterativa utilizada por el método es la siguiente:

$$x_i = \frac{b_i - \sum_{j=1}^{i-1} a_{ij}x_j^s - \sum_{j=i+1}^n a_{ij}x_{ji}^s}{a_{ii}} \quad (3.1)$$

Los pasos a seguir por este método son:

1. Se toman los valores iniciales y se evalúan en la fórmula, luego los resultados se vuelven a ingresar de manera iterativa.
2. El método termina cuando el error encontrado es menor a la tolerancia (el error corresponde a la norma infinita) o cuando se llega al número máximo de iteraciones.

3.1.2 Método de Gauss-Seidel

El método de Gauss-Seidel tiene un funcionamiento similar al de Jacobi, pero utiliza los valores calculados en cada uno de los pasos para encontrar los nuevos valores. La fórmula utilizada por el método es:

$$x^{s+1} = \frac{b_i - \sum_{j=1}^{i-1} a_{ij}x_j^{s+1} - \sum_{j=i+1}^n a_{ij}x_j^s}{a_{ii}} \quad (3.2)$$

Los pasos a seguir por este método son:

1. Se toman los valores iniciales y se evalúan en la fórmula, luego los resultados se vuelven a ingresar de manera iterativa.
2. El método termina cuando el error encontrado es menor a la tolerancia (el error corresponde a la norma infinita) o cuando se llega al número máximo de iteraciones.

3.2 MÉTODOS DIRECTOS

Los métodos directos se caracterizan por tener un número finito de pasos para encontrar la solución, se basa principalmente en el uso de operaciones de filas en la matriz, para transformar el sistema de ecuaciones en algo más simple de resolver.

3.2.1 Factorización LU

La factorización LU es un método de descomposición de matrices que permite generar a partir de una matriz A una matriz triangular inferior y otra triangular superior. Como se utiliza un sistema de ecuaciones $Ax = b$, realizando la transformación, se tienen las siguientes propiedades:

- $LUx = b$

- $Ux = Y$

- $LY = b$

Dentro de los métodos LU se encuentra el método de Crout y Doolittle, en donde su principal diferencia es que en el primero la diagonal con unos corresponde a la matriz U , y en el segundo a la matriz L . Para este método se necesitan los valores de la matriz de coeficientes y el vector de valores, además la matriz de los coeficientes debe ser cuadrada. Los pasos a seguir por este método para encontrar las soluciones son los siguientes:

1. Con la matriz de coeficientes se calculan la matriz L y U respectivamente.

$$\begin{pmatrix} a_{11} & a_{12} & a_{13} & \dots & a_{1n} \\ a_{21} & a_{22} & a_{23} & \dots & a_{2n} \\ \dots & \dots & \dots & \dots & \dots \\ a_{n1} & a_{n2} & a_{n3} & \dots & a_{nn} \end{pmatrix} = \begin{pmatrix} l_{11} & 0 & 0 & \dots & 0 \\ l_{21} & l_{22} & 0 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots \\ l_{n1} & l_{n2} & \dots & l_{nn-1} & l_{nn} \end{pmatrix} * \begin{pmatrix} u_{11} & u_{12} & \dots & u_{1n-1} & u_{1n} \\ 0 & u_{22} & \dots & u_{2n-1} & u_{2n} \\ \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & \dots & u_{nn} \end{pmatrix} \quad (3.3)$$

2. Se calcula Y despejando el vector de la ecuación $LY = b$.

$$\begin{pmatrix} l_{11} & 0 & 0 & \dots & 0 \\ l_{21} & l_{22} & 0 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots \\ l_{n1} & l_{n2} & \dots & l_{nn-1} & l_{nn} \end{pmatrix} * \begin{pmatrix} y_1 \\ y_2 \\ \dots \\ y_{n-1} \\ y_n \end{pmatrix} = \begin{pmatrix} b_1 \\ b_2 \\ \dots \\ b_{n-1} \\ b_n \end{pmatrix} \quad (3.4)$$

3. Se calcula el vector de resultados x despejando los valores de la ecuación $Ux = Y$.

$$\begin{pmatrix} u_{11} & u_{12} & \dots & u_{1n-1} & u_{1n} \\ 0 & u_{22} & \dots & u_{2n-1} & u_{2n} \\ \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & \dots & u_{nn} \end{pmatrix} * \begin{pmatrix} x_1 \\ x_2 \\ \dots \\ x_{n-1} \\ x_n \end{pmatrix} = \begin{pmatrix} y_1 \\ y_2 \\ \dots \\ y_{n-1} \\ y_n \end{pmatrix} \quad (3.5)$$

3.2.2 Método Cholesky

El método de Cholesky utiliza la factorización LU , pero en lugar de poseer una diagonal con unos, las diagonales de ambas matrices son idénticas. Para poder utilizar este métodos, las matrices deben ser definidas positivas y simétricas. Los pasos a seguir por el método son:

1. Con la matriz de coeficientes se calculan la matriz L y L^t respectivamente.

$$\begin{pmatrix} a_{11} & a_{12} & a_{13} & \dots & a_{1n} \\ a_{21} & a_{22} & a_{23} & \dots & a_{2n} \\ \dots & \dots & \dots & \dots & \dots \\ a_{n1} & a_{n2} & a_{n3} & \dots & a_{nn} \end{pmatrix} = \begin{pmatrix} l_{11} & 0 & 0 & \dots & 0 \\ l_{21} & l_{22} & 0 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots \\ l_{n1} & l_{n2} & \dots & l_{nn-1} & l_{nn} \end{pmatrix} * \begin{pmatrix} l_{11} & l_{12} & \dots & l_{1n-1} & l_{1n} \\ 0 & l_{22} & \dots & l_{2n-1} & l_{2n} \\ \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & \dots & l_{nn} \end{pmatrix} \quad (3.6)$$

2. Se calcula Y despejando el vector de la ecuación $LY = b$.

$$\begin{pmatrix} l_{11} & 0 & 0 & \dots & 0 \\ l_{21} & l_{22} & 0 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots \\ l_{n1} & l_{n2} & \dots & l_{nn-1} & l_{nn} \end{pmatrix} * \begin{pmatrix} y_1 \\ y_2 \\ \dots \\ y_{n-1} \\ y_n \end{pmatrix} = \begin{pmatrix} b_1 \\ b_2 \\ \dots \\ b_{n-1} \\ b_n \end{pmatrix} \quad (3.7)$$

3. Se calcula el vector de resultados x despejando los valores de la ecuación $L^t x = Y$.

$$\begin{pmatrix} l_{11} & l_{12} & \dots & l_{1n-1} & l_{1n} \\ 0 & l_{22} & \dots & l_{2n-1} & l_{2n} \\ \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & \dots & l_{nn} \end{pmatrix} * \begin{pmatrix} x_1 \\ x_2 \\ \dots \\ x_{n-1} \\ x_n \end{pmatrix} = \begin{pmatrix} y_1 \\ y_2 \\ \dots \\ y_{n-1} \\ y_n \end{pmatrix} \quad (3.8)$$

3.2.3 Factorización QR

La factorización QR corresponde a la descomposición de la matriz A en una matriz ortogonal y otra triangular superior. La ortogonalización es realizada mediante el método de Gram-Schmidt.

3.2.4 Método de Householder

El método de Householder es una variación del método QR, donde se reemplaza la matriz Q por una matriz H , donde esta matriz es ortogonal y simétrica. Para calcular esta matriz se usa la siguiente formula.

$$H = I - \frac{2}{v^t v} v v^t \quad (3.9)$$

En la ecuación (2.11) la variable v es un vector no nulo de la matriz A . Además, en esta ecuación $v v^t$ es una matriz cuadrada ($n \times n$) y $v v^t$ es un número.

Los pasos a seguir en este método son.

1. Calcular la matriz H mediante la ecuación (2.11)
2. Mediante la ecuación $R = H^t A$ se calcula la matriz R
3. Se obtiene los valores de las raíces resolviendo la ecuación $Rx = H^t b$

Este método tiene varias ventajas, en concreto, la matriz H al ser igual a su propia inversa ahorra numerosos cálculos ya que estas no se deben invertir y al ser ortogonales las hace idóneas para el calculo de matrices semejantes.

3.2.5 Método de Givens

El método de Givens o rotaciones de Givens permiten calcular la descomposición QR de la matriz A . En este, cada rotación anula un elemento de la subdiagonal de la matriz, así formando la matriz R .

Una rotación de givens tiene la forma genérica siguiente.

$$G = \begin{pmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{pmatrix} \quad (3.10)$$

Donde se tienen las siguientes ecuaciones para calcular los elementos de la matriz G .

$$\sin(\theta) = \frac{x_j}{\sqrt{x_i^2 + x_j^2}} \quad (3.11)$$

$$\cos(\theta) = \frac{x_i}{\sqrt{x_i^2 + x_j^2}} \quad (3.12)$$

CAPÍTULO 4. FUNCIONES

Las funciones que se utilizarán para el análisis de los métodos no lineales son:

1. $f(x) = x - 2^{-x}$
2. $f(x) = e^{-x} - x^2 + 3x - 2$
3. $f(x) = x^3 - 7x^2 + 14\text{sen}(2x\pi)$

Las funciones para el método de Newton-Raphson multivariantes son:

$$\begin{cases} x_1^2 - 10x_1 + x_2^2 + 8 = 0 \\ x_1x_2^2 - x_1 - 10x_2 + 8 = 0 \\ X_{(0)} = (0, 0)^T \end{cases} \quad (4.1)$$

$$\begin{cases} x_1^2 + x_2 - 37 = 0 \\ x_1 - x_2 - 5 = 0 \\ x_1 + x_2 + x_3 - 3 = 0 \\ X_{(0)} = (0, 0, 0)^T \end{cases} \quad (4.2)$$

4.0.1 Regla de Fourier

Para la elección de los puntos que se utilizarán al momento de comenzar las iteraciones, es de vital importancia restringir el dominio de estos, debido a que existen casos en donde el punto inicial implica si el método converge o no (en Newton-Raphson el punto elegido debe ser cercano a la raíz), métodos en donde si la raíz no se encuentra dentro del dominio, esta no será encontrada (métodos cerrados) y métodos en donde si la función tiene más de una raíz (como en el caso de funciones trigonométricas), los algoritmos recorrerán la función, pero sin encontrar las raíces, por lo cual entrarán en un ciclo infinito. La regla de Fourier indica que si $f(x)$ es una función continua y dos veces derivable en el intervalo $[a, b]$ y si los signos de las funciones evaluadas en los puntos son distintos $\text{sig}(f(a)) = \text{sig}(f(b))$ y sus dos primeras derivadas $f'(x)$ y $f''(x)$ no se anulan, entonces existe una única raíz en el intervalo, tal como se aprecia en la siguiente imagen.

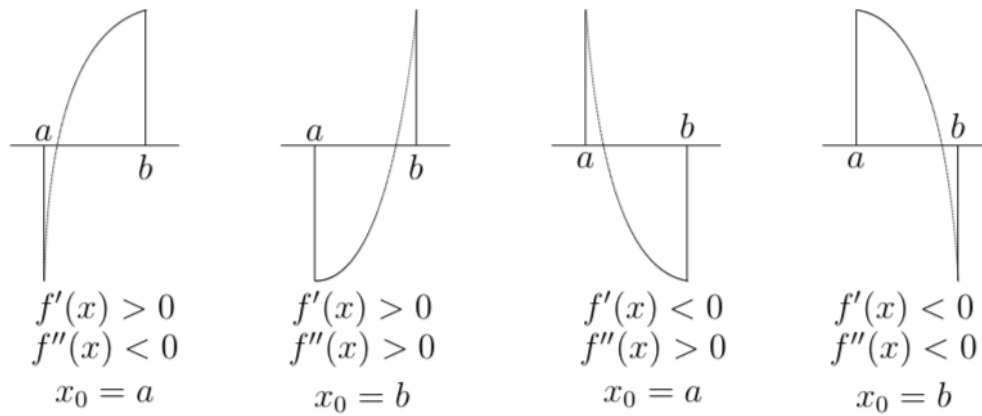


Figura 4-1: Regla de Fourier.

Para comenzar las iteraciones, y mediante el uso de la regla de Fourier se tiene que:

$$f(x) = x - 2^{-x}$$

$$f'(x) = 1 + \ln(2) * 2^{-x}$$

$$f''(x) = -\ln(2) * 2^{-x}$$

En el intervalo $[0, 1]$, por lo que se toma a 0 como x_0 para los casos en donde solo se necesita un valor inicial, y el intervalo para el resto, tal como muestra la tabla a continuación:

Tabla 4.1: Valores iniciales para función 1.

Método	x0	x1
Bisección	0	1
Secante	0	1
Regula Falsi	0	1
Newton-Rhapson	0	-

Para la función 2, mediante la regla de Fourier se obtienen los siguientes puntos:

$$f(x) = e^{-x} - x^2 + 3x - 2$$

$$f'(x) = -e^{-x} - 2x + 3$$

$$f''(x) = e^{-x} - 2$$

Tabla 4.2: Valores iniciales para función 2.

Método	x0	x1
Bisección	0.5	1
Secante	0.5	1
Regula Falsi	0.5	1
Newton-Rhapson	0.5	-

Para la función 3 es importante analizar cual será el dominio que tendrá la función, esto se debe a que presenta múltiples raíces, por lo tanto es posible que no todos los métodos converjan al mismo valor, lo cual impide que se realice la comparación entre ellos de manera correcta. Según la regla de Fourier y lo detallado anteriormente, se obtienen los siguientes puntos:

$$f(x) = x^3 - 7x^2 + 14\text{sen}(2x\pi)$$

$$f'(x) = 3x^2 - 14x + 28\pi\cos(2x\pi)$$

$$f''(x) = 6x - 14 + 56\pi^2\text{sen}(2x\pi)$$

Tabla 4.3: Valores iniciales para función 2.

Método	x0	x1
Bisección	-0.5	0.3
Secante	-0.5	0.3
Regula Falsi	-0.5	0.3
Newton-Rhapson	0	-

Es importante señalar que la tolerancia soportada por los métodos es de 10^{-12} .

CAPÍTULO 5. RESULTADOS

A continuación se presentan los gráficos obtenidos para cada una de las funciones con los métodos de la bisección, regula falsi, Newton Raphson y secante, además de una tabla resumen con el valor obtenido por cada método, los costos temporales, espaciales y el número de iteraciones.

5.1 RESULTADOS FUNCIÓN 1

5.1.1 Gráfico iteraciones

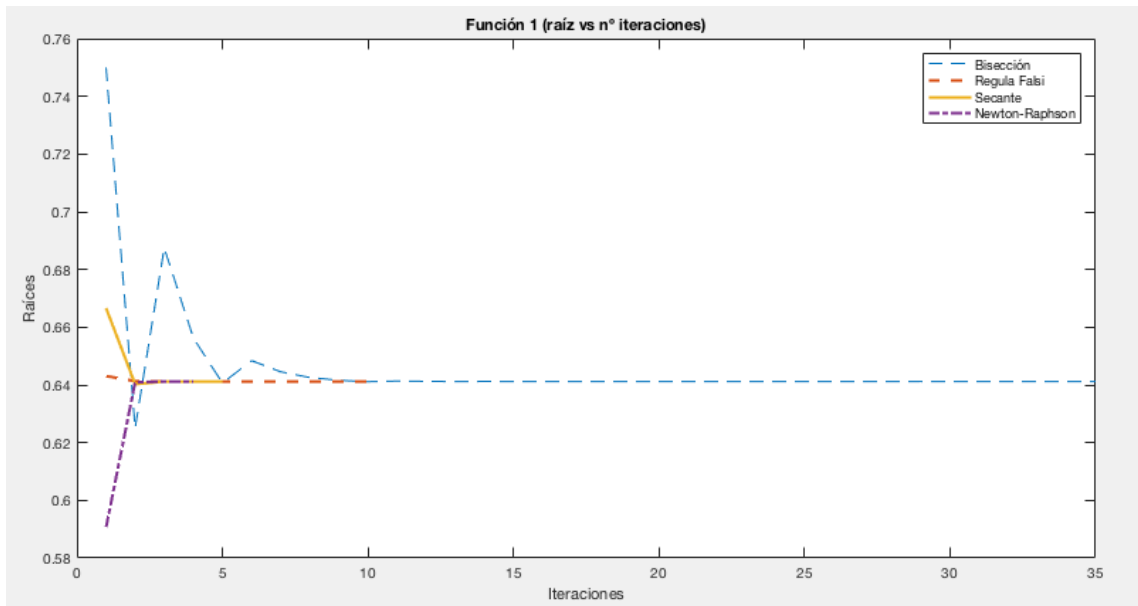


Figura 5-1: Iteraciones función 1.

Tabla 5.1: Tabla de resultados función 1

Método	Aproximación	Iteraciones	Costos temporales	Error
Bisección	0.641185744505492	36	0.014748158	1.455191522836685e-11
Secante	0.641185744504986	5	0.001903701	2.491323813913482e-10
Regula Falsi	0.641185744506646	11	0.003243284	1.536992755291067e-12
NR	0.641185744504986	4	0.003514544	8.131175732728479e-09

5.2 ITERACIONES FUNCIÓN 2

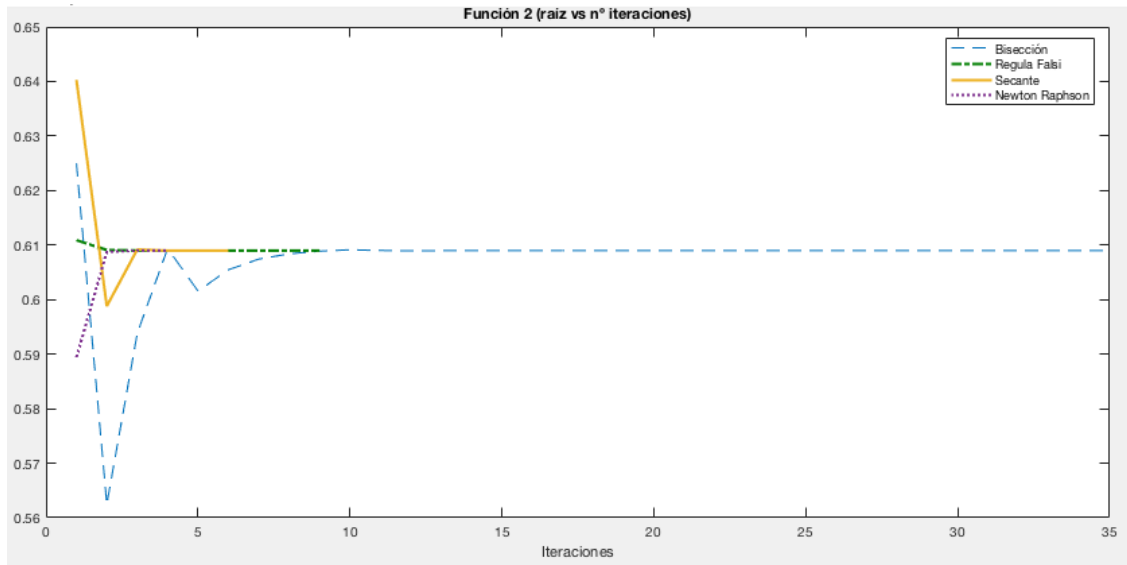


Figura 5-2: Iteraciones función 1.

Tabla 5.2: Tabla de resultados función 2

Método	Aproximación	Iteraciones	Costos temporales	Error
Bisección	0.608989103006024	36	0.009813449	7.275957614183426e-12
Secante	0.608989103010166	6	6.74561e-04	1.264714999393846e-10
Regula Falsi	0.608989103006024	10	0.002154467	4.690914323646211e-12
NR	0.608989103010165	4	0.003710529	2.847689384299912e-08

5.3 ITERACIONES FUNCIÓN 3

Para la función 3, en el caso de Newton-Raphson fueron elegidos dos valores para comenzar la iteración. Esto se debe a que de lo contrario la función converge a un valor diferente al del resto de los métodos y no sería posible realizar la comparación entre ellos, debido a que la ecuación posee variadas raíces.

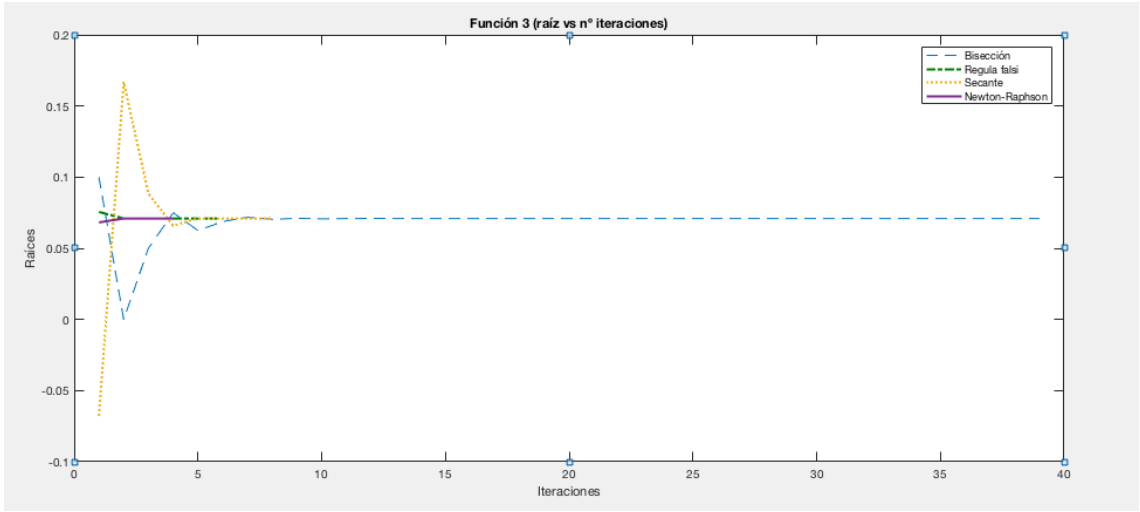


Figura 5-3: Iteraciones función 3.1

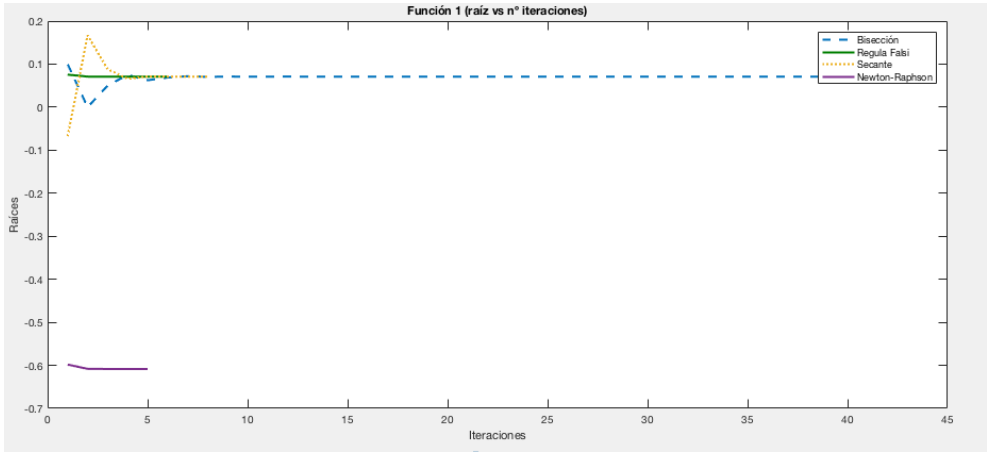


Figura 5-4: Iteraciones función 3.2

Tabla 5.3: Tabla de resultados función 3

Método	Aproximación	Iteraciones	Costos temporales	Error
Bisección	0.070930401236228	44	0.006261908	4.547751064620797e-14
Secante	0.070930401236217	8	0.001569783	2.995264591909574e-10
Regula Falsi	0.070930401236225	7	0.001255891	1.883507239064386e-12
NR	0.070930401236217	4	0.00265239	2.132995724490883e-10

5.4 SISTEMA DE ECUACIONES CON NEWTON-RAPHSON

Se sintetizan los resultados obtenidos con el método de Newton-Raphson multivariantes en la siguiente tabla, para los sistemas de dos y tres incógnitas presentados anteriormente, y con una tolerancia de 10^{-2} . A pesar de que el resultado que se puede obtener es exacto (con una tolerancia de cero), fue realizado con un error mayor para poder ilustrar en menos iteraciones como varía el método durante las iteraciones.

Tabla 5.4: Tabla de resultados sistema de ecuaciones 1

Tiempo ejecución: 0.012985983

Iteración	x	y	Error
0	0	0	-
1	0.804214219173303	0.804214219173303	0.804214219173303
2	0.890985065683538	0.890985065683538	0.086770846510235
3	0.920789591138063	0.920789591138063	0.029804525454525
4	0.937051244644437	0.937051244644437	0.016261653506375
5	0.986969763943797	0.986969763943797	0.049918519299359
6	0.995065374989048	0.995065374989048	0.008095611045252

Tabla 5.5: Tabla de resultados sistema de ecuaciones 2

Tiempo ejecución: 0.031217.

Iteración	x	y	z	Error
0	0	0	0	-
1	5.000000030387355	36.999996938606529	-38.999996950761471	53.990735648999731
2	4.350877013813784	18.491231469141432	-19.842108482955219	26.646208931838931
3	5.363824815192837	9.255453970976651	-11.619278786169486	12.407279212876080
4	5.696055456650742	4.665329966963007	-7.361385423613749	6.269710635925870
5	5.882822620623157	2.427280142481938	-5.310102945295597	3.041629299611164
6	5.966094750867120	1.412647731713453	-4.378742565301984	1.379799019451414
7	5.995188109884477	1.058565996691071	-4.053754106575548	0.481495375804127
8	5.999871861950288	1.001559580217726	-4.001431442168013	0.077519870118892
9	5.999999904179513	1.000001166362872	-4.000001070542385	0.002119200684898

5.5 RESULTADOS RANKING MÉTODOS DE SOLUCIÓN DE SISTEMAS DE ECUACIONES

Para lograr realizar el ranking, es necesario tomar en cuenta factores como el tiempo de ejecución de cada método de solución de sistemas de ecuaciones, el costo espacial y el error. Se entiende que entre menor es el error, tiempo de ejecución y costos espaciales, mejor es el algoritmo.

5.5.1 Tiempo de ejecución

El tiempo de ejecución es medido en segundos y trabajado en porcentaje tanto en el gráfico como en la tabla.

5.5.2 Error

El error de cada método es obtenido mediante la fórmula $Ax = b$, y al igual que el tiempo es medido en porcentajes.

5.5.3 Costos espaciales

Los costos espaciales corresponde a la cantidad de operaciones realizadas por cada algoritmo, para este caso son calculados mediante la fórmula $f(x, y) = x + 3 * y$, en donde x corresponde a las operaciones de suma y resta e y a las multiplicaciones y divisiones, que son multiplicadas por 3 debido al costo computacional adicional en relación a las

operaciones aditivas. Al igual que en los factores anteriores, los costos espaciales son trabajados en porcentajes.

El ranking fue calculado utilizando el porcentaje de eficacia (que depende del error) con un cincuenta por-ciento y el porcentaje de eficiencia (dependiendo del tiempo y los costos espaciales) también con un cincuenta por-ciento .

A continuación se presentan los resultados obtenidos al ingresar los métodos a la función ranking, con una matriz de 289 filas por 289 columnas.

Tabla 5.6: Tabla de tiempos, errores y costos espaciales matriz 289x289

Método	Tiempo	Error	NºOperaciones
LU	0.02884284	0.102067315787698	582624
Cholesky	0.26807619	0.070512126359375	1672732
Jacobi	8.656166394	0.032690875559471	62473895
Gauss-Seidel	4.862459446	0.053964238030386	32995130
QR	0.186713111	0.210149836752013	1669264
Householder	0.901104543	0.345140830798859	22752
Givens	53.544947888000003	0.203285451525137	48444203

5.5.4 Gráfico Tiempos Métodos

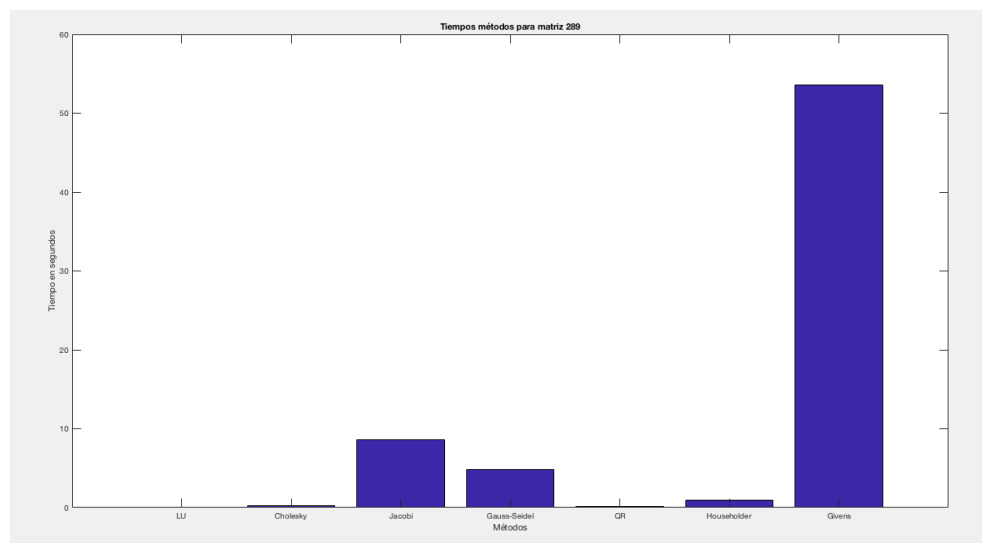


Figura 5-5: Tiempos matriz 289

5.5.5 Gráfico Error Métodos

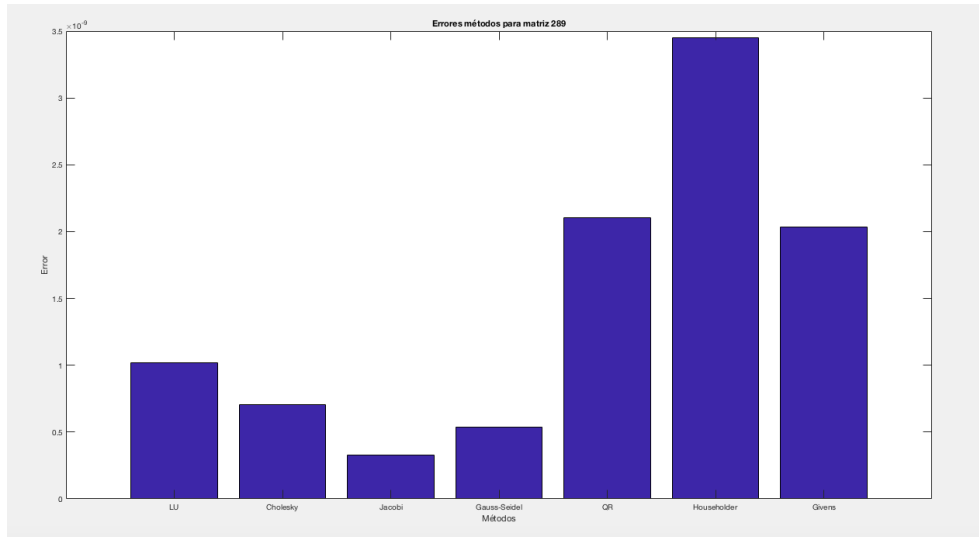


Figura 5-6: Error matriz 289

5.5.6 Gráfico Ranking

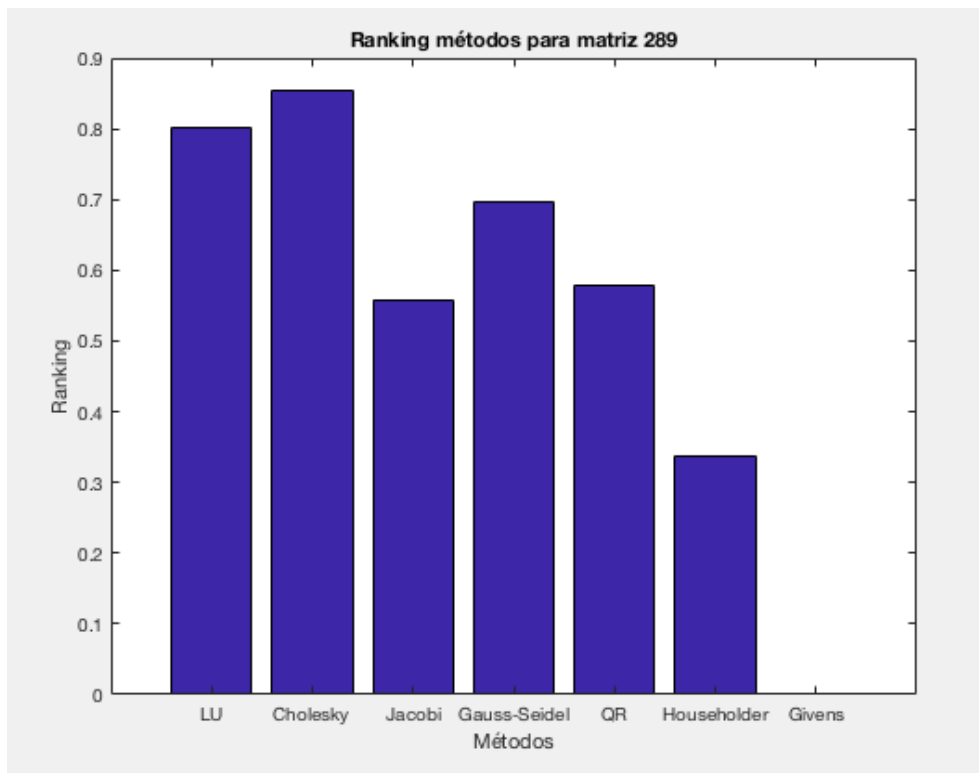


Figura 5-7: Ranking sistema 289

El ranking fue calculado utilizando el porcentaje de eficacia (que depende del error) con un cincuenta por-ciento y el porcentaje de eficiencia (dependiendo del tiempo y los costos espaciales) también con un cincuenta por-ciento . Por lo tanto, entre mayor es el valor presente en el ranking, mejor será el método.

Tabla 5.7: Tabla de tiempos, errores y costos espaciales sistema 1089x1089

Método	Tiempo	Error	NºOperaciones
LU	1.20111153	1.380592326511123e-09	8293824
Cholesky	6.452391944	1.119813936685487e-09	23727132
Jacobi	4.30326173604e+02	5.680841829683319e-10	3.34904161e+09
Gauss-Seidel	1.94125130834e+02	4.866653597895957e-10	1.677980205e+09
QR	4.208470635	4.440864134553134e-09	23714064
Householder	1.17748924417e+02	1.085598494837195e-08	85952
Givens	Sin información	Sin información	Sin información

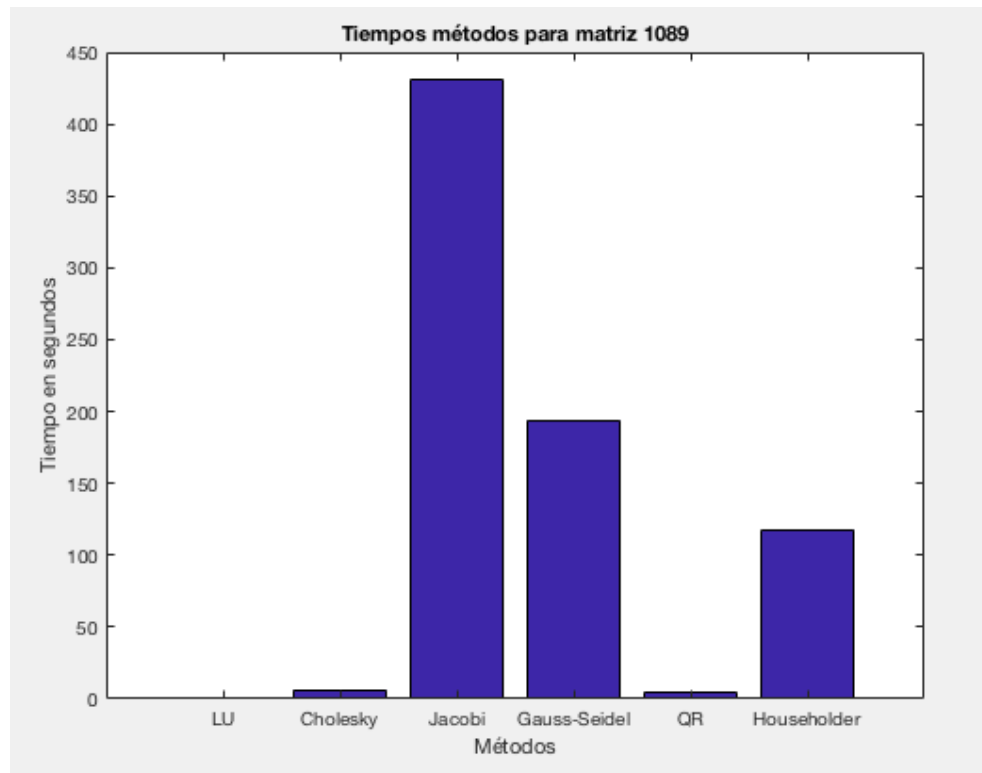


Figura 5-8: Tiempos sistema 1089

5.5.7 Gráfico Error Métodos

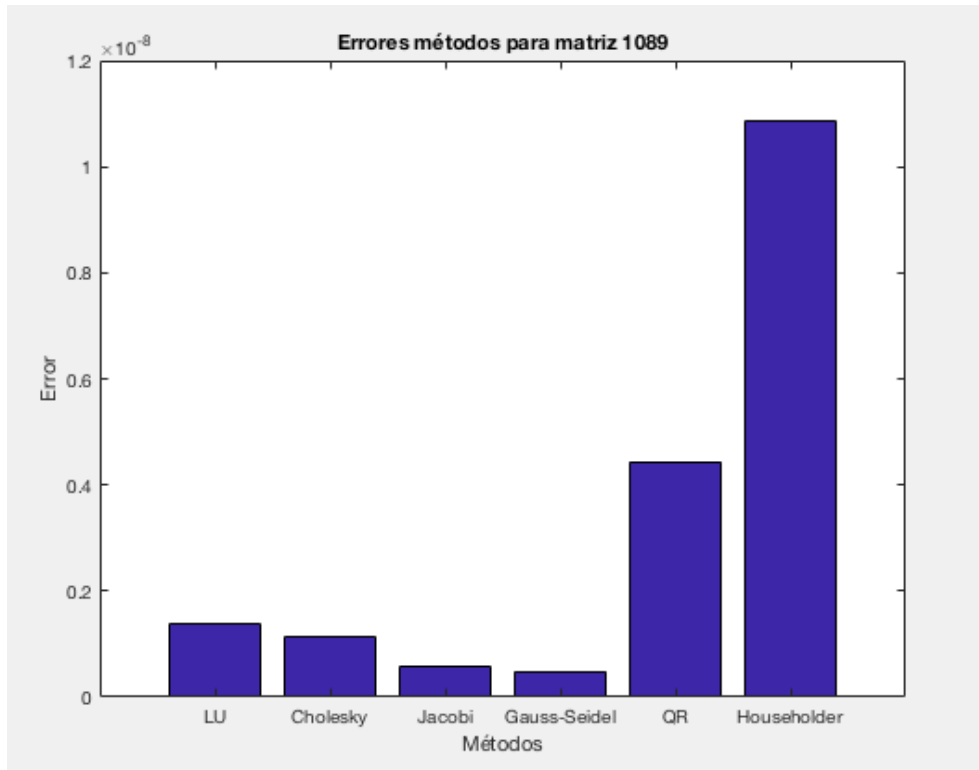


Figura 5-9: Error sistema 1089

5.5.8 Gráfico Ranking

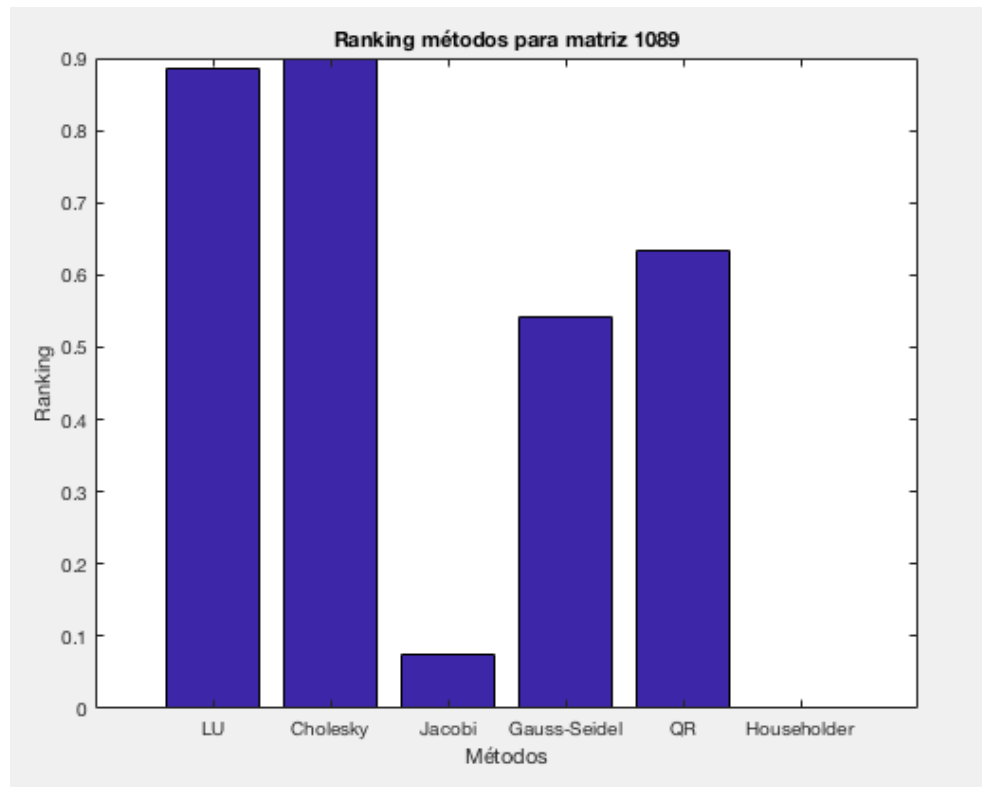


Figura 5-10: Ranking sistema 1089

Para la matriz de 1089 no fue posible graficar el método de Givens, debido a el exceso de tiempo que presenta el método, tal como se muestra en la figura a continuación.

Nombre del proceso	% CPU	Tiempo CPU	Subprocesos
MATLAB	194,2	8:54:42,57	69

Figura 5-11: Tiempo de ejecución Givens para sistema 1089

Tabla 5.8: Tabla de tiempos, errores y costos espaciales sistema 4225x4225

Método	Tiempo	Error	NºOperaciones
Cholesky	1.0301433020000000e+03	2.267615200428556e-09	357046300
Gauss-Seidel	9.5483503270000000e+03	1.139259084566553e-09	9.6444708750000000e+10
LU	1.0885894400000000e+02	2.863327166852555e-09	299975
QR	2.1138418800000000e+02	1.169990537776360e-08	223126475

Para el caso del sistema de 4225 no se consideraron los métodos de Householder, Givens y Jacobi, debido al excesivo tiempo que tomaba el método para encontrar los resultados, por lo cuál se deduce que al momento de generar el ranking e ingresando cualquier tipo de porcentaje (de eficacia y eficiencia) para cada uno, nunca se encontrarán entre los mejores para desarrollar los algoritmos.

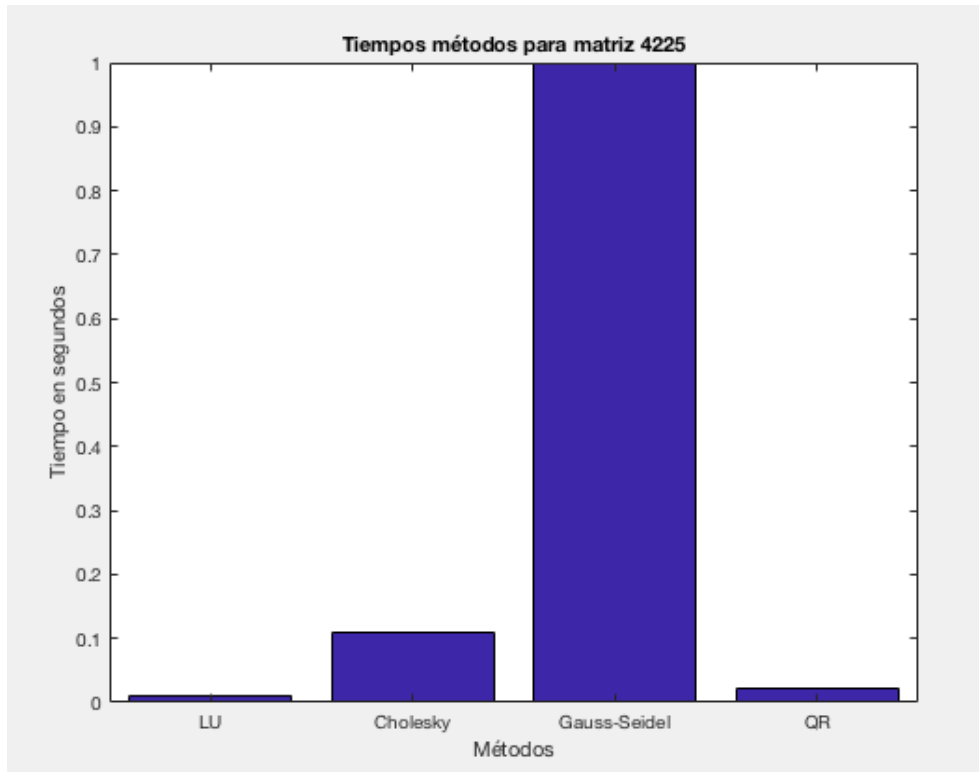


Figura 5-12: Tiempos sistema 4225

5.5.9 Gráfico Error Métodos

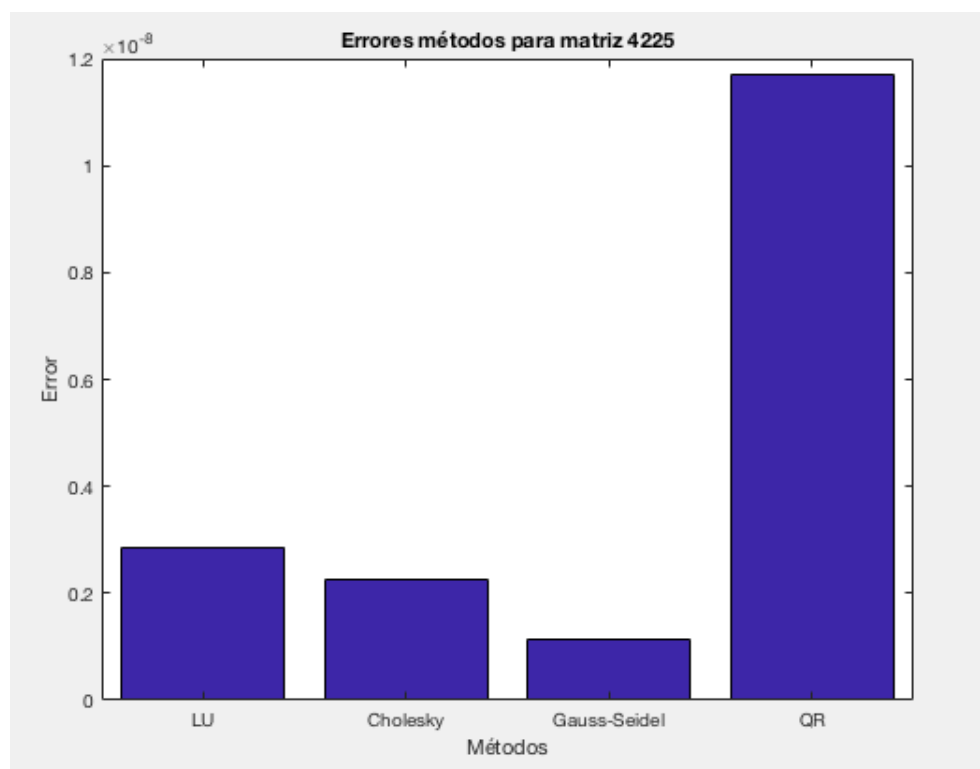


Figura 5-13: Error sistema 4225

5.5.10 Gráfico Ranking

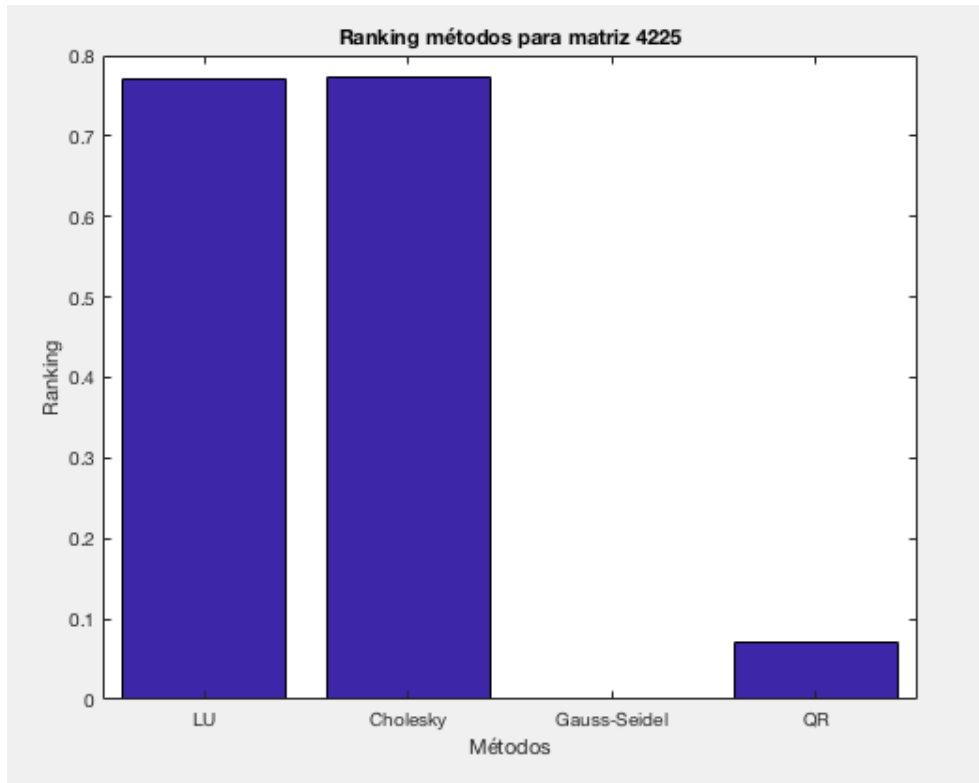


Figura 5-14: Ranking sistema 4225

CAPÍTULO 6. ANÁLISIS DE RESULTADOS

6.1 ANÁLISIS MÉTODOS DE RESOLUCIÓN DE ECUACIONES NO LINEALES

A partir de los resultados de las funciones 1,2 y 3, se puede apreciar que la principal diferencia entre los métodos de resolución de ecuaciones no lineales es el número de iteraciones que necesita cada uno para llegar al error. El método de la bisección es quien presenta un mayor número de iteraciones, esto es debido a que calcula siempre el valor medio, sin tomar en consideración que la raíz buscada se puede encontrar cerca del punto obtenido, por lo cual presenta una convergencia lineal. Entre sus ventajas se puede destacar que si el intervalo en el que itera contiene la raíz, el método siempre converge.

El resto de los métodos presentan un comportamiento similar con respecto al número de iteraciones, especialmente Newton-Raphson y el método de la secante, esto se debe a que la secante calcula la pendiente para obtener el siguiente valor, tal como Newton calcula la derivada.

A medida que se aumenta el número de iteraciones de todos los métodos, se puede notar que el error que presentan disminuye.

En los tres casos los métodos que convergen de manera más rápida son secante y Newton-Raphson, además este último lo hace con el menor número de iteraciones, esto se debe a que calcula el siguiente valor mediante la tangente a la curva que se está evaluando, por lo tanto, es posible confirmar lo planteado al inicio del informe, con respecto a que este método es el más eficiente de los desarrollados. Una de las desventajas que presenta es debido a que es un método abierto, por lo tanto, no siempre converge, y para hacerlo, necesita iniciar su iteración con un valor cercano a la raíz, lo cual es un dato que no siempre es posible obtener antes de la primera iteración.

En la función número tres, se presentan dos gráficos, en donde es posible ilustrar que a partir del punto de inicio de la iteración, Newton-Raphson no siempre convergerá al mismo valor que el resto de los métodos (en caso de que exista más de una raíz), esto implica que en algunos casos (como en las funciones trigonométricas), el método comience a calcular

tangentes, pero sin acercarse a la raíz, por lo cual entrará en un ciclo infinito.

6.2 ANÁLISIS MÉTODO NEWTON-RAPHSON MULTIVARIABLE

En el método de Newton-Raphson multivariantes, se puede apreciar que existe un aumento en el tiempo con respecto a los sistemas de ecuaciones de dos y tres variables, esto se debe a que el cálculo de la matriz inversa implica grandes costos computacionales a medida que aumentan las variables en los sistemas de ecuaciones.

En estos casos, si los sistemas tienen solución, es posible obtener un error igual o muy cercano a cero en un número pequeño de iteraciones. En un inicio, los errores son valores muy altos, por lo tanto es necesario realizar varias iteraciones para obtener un resultado más exacto. Como se muestra en la tabla, el primer sistema converge a un error de 10^{-2} en 6 iteraciones, mientras el segundo sistema lo hace en 9.

6.3 RANKING

A partir de los datos obtenidos en las tablas y gráficos de tiempos, errores y ranking. Para el sistema de la 289, se puede apreciar que los costos temporales tienen una gran variación unos con otros, ya que la mayoría logra entregar el resultado en menos de un segundo, siendo el método LU el que presenta el menor tiempo, seguido por Cholesky, QR y Householder. Dentro de los métodos que presentan los mayores costos temporales, Givens destaca sobre el resto, teniendo una ejecución de más de cincuenta segundos, esto se debe a que los métodos iterativos deben realizar un mayor número de operaciones para alcanzar el error requerido, por lo que incluso en algunos casos se puede presentar una iteración infinita.

Con respecto al error presente en el sistema, en todos los casos se obtienen cifras menores a uno, por lo cual todos los métodos entregan una precisión aceptable, no obstante, destaca con un menor costo Jacobi.

Finalmente, según el número de operaciones realizadas, calculadas ocupando el criterio antes mencionado, los métodos que presentan los menores costos son también aquellos que tienen los menores tiempos, por lo cual existe una relación directa entre ambos factores, lo cual implica que el cálculo de la eficacia sea realizado a partir de ambos datos. Todos estos valores antes mencionados, generan el ranking de métodos, en donde el mejor

corresponde a Cholesky, esto se debe a que el método utiliza la matriz transpuesta para realizar el cálculo, por lo que al momento de evaluar son menos las operaciones que se deben realizar, lo contrario pasa con los métodos iterativos, en donde las operaciones que se realizan lo hacen de manera constante, solo se modifican los valores que se ingresan a la evaluación. Es importante destacar que el ranking fue realizado con un cincuenta por ciento de eficacia y eficiencia.

Para el caso del segundo sistema de ecuaciones, se tiene en consideración que Givens es el método que presenta el mayor tiempo, debido a esto no fue posible obtener su resultado y se detuvo su ejecución luego de ocho horas, por lo cuál sus datos no son ingresados en los resultados de ranking. En este caso, Jacobi pasa a ser el método con el mayor tiempo (sin contar a Givens), continuando con lo sucedido en la matriz de 289, en donde fue el que logró un mayor valor en relación al promedio. Esto se debe a que los métodos iterados sólo varían el valor de sus constantes, no la cantidad de operaciones que realizan, por lo que al obtener sistemas con más valores, estos métodos aumentarán significativamente su costo temporal. A medida que aumentan los sistemas de ecuaciones, disminuye el error asociado a cada uno de los métodos, pero continúan siendo los métodos iterados los que presentan el menor, siendo Jacobi el que posee mayor precisión. En el caso del número de operaciones, los valores se mantienen en proporción al sistema anterior. Finalmente, se obtuvo un ranking en donde nuevamente Cholesky es quien figura como el mejor método en términos de eficacia y eficiencia y Givens como el peor. Agrupando los métodos entre iterativos y directos, son los primeros aquellos que presentan un peor algoritmo cuando mayor son los sistemas de ecuaciones.

Para el tercer sistema de ecuaciones analizado, se descarta la implementación de los métodos de Givens, Householder y Jacobi, debido a que su cálculo demora demasiado tiempo, lo cual implica que en ninguna circunstancia para una matriz de mayor tamaño estos serán considerados como los mejores. Cholesky continua siendo el mejor método con respecto al ranking, seguido de cerca por LU. Los valores tanto de los errores, tiempos de ejecución y costos espaciales se siguen comportando como en el sistema de 289 y 1089, por lo cual se puede determinar que los valores son correctos.

CAPÍTULO 7. CONCLUSIONES

A partir del análisis realizado se puede concluir que existe una gran variedad de métodos para ecuaciones no lineales que permiten obtener la raíz, tanto para una o más variables. Es determinante conocer en un inicio el dominio que presenta la función que se quiere analizar, ya que para los métodos cerrados como bisección y regula falsi, si el intervalo que se ingresa en un inicio no contiene a la raíz, el método no podrá encontrar nunca la solución. En el caso de Newton Raphson, sucede algo similar, ya que si el valor entregado al comenzar la iteración no es cercano a la raíz, se pueden producir tres casos: el primero es donde el método no converge, el segundo caso es cuando se encuentra una solución con valor cero, la cual es distinta al valor real de la raíz, y el tercer caso (que también es el más extraño y difícil que ocurra) es donde la solución es periódica, por lo tanto el algoritmo itera siempre entre dos números. La regla de Fourier es una buena práctica para evitar que sucedan los errores mencionados anteriormente, ya que obliga a que los valores ingresados en la primera iteración, entreguen la solución esperada.

Otro factor importante a considerar es el nivel de error o tolerancia que se asigna al momento de comenzar las iteraciones. A medida que disminuye la tolerancia que tendrá el método, el error por ende será menor y el cálculo entregado al finalizar el algoritmo será más exacto, pero también implica un mayor número de tiempo de ejecución e iteraciones. Con respecto a los métodos desarrollados en el laboratorio, debido a que los errores son bajos en todos los casos, el principal criterio para obtener el mejor método es según el número de iteraciones y los costos temporales, ya que estos valores se relacionan de manera directa, por lo tanto el mejor algoritmo con respecto a tiempo es la secante, y con respecto al número de iteraciones es Newton-raphson, esto se debe a que a pesar de que Newton tiene menos iteraciones, su costo espacial es mayor (mayor número de operaciones por iteración). Algunos de los métodos presentan similitudes en sus resultados, esto se debe a que unos son variaciones de los otros, como por ejemplo el método de la bisección con regula falsi y el método de la secante con Newton-Raphson.

Con respecto al método de Newton-Raphson multivariable, es un método eficiente para resolver sistemas de ecuaciones, debido a que entrega resultados con errores igual a cero.

En el caso del ejemplo entregado, se buscaba explicar como disminuye significativamente el error a medida que avanzan las iteraciones y que continua con ellas hasta encontrar la tolerancia buscada, y para representar esto fue tomado un error mayor a cero. Además, las iteraciones y tiempos aumentan a medida que aumenta las variables buscadas en los sistemas de ecuaciones.

El ranking es una buena herramienta para determinar el método a utilizar dependiendo de los requerimientos que tiene la persona que implementará el algoritmo. Debido a que los tres sistemas entregan resultados similares con respecto a los costos temporales, espaciales y al error, se puede determinar que el mejor método implementado en los tres casos de matrices estudiado fue Cholesky, esto se debe a que a medida que recorre la matriz de valores y encuentra los resultados, los reemplaza de manera inmediata, disminuyendo así el número de operaciones realizadas, al contrario de lo sucedido con los métodos iterados, en donde a medida que avanza, siempre debe realizar el mismo número de operaciones, lo cual genera que sean los "peores" métodos en términos temporales, de costos espaciales y su error. A pesar de ser Cholesky el mejor de los métodos, si un usuario no se encuentra interesado en el error que entregará el resultado final y sólo quiere bajos costos espaciales y temporales, los métodos entregados por el ranking serán los métodos LU y Gauss-Seidel. Cada método presenta ventajas y desventajas, por lo que depende de el uso que se le quiere dar el método a elegir. Por lo tanto, es posible afirmar que el objetivo del laboratorio fue cumplido, ya que se implementaron todos los métodos requeridos, junto con sus tablas de datos y gráficos, que permitieron realizar las comparaciones y entender el funcionamiento de cada uno.

Finalmente, se puede concluir que Matlab es un buen software para el análisis de datos, algoritmos y modelos matemáticos, ya que cuenta con las herramientas necesarias, tales como la capacidad de implementar funciones y matrices, junto con sus respectivos gráficos, para llevar a cabo todas las pruebas que fueron requeridas en este laboratorio.

CAPÍTULO 8. BIBLIOGRAFÍA

Cobos, F. (2002). *Apuntes de Cálculo Numérico*. Universidad de Sevilla.

Plaza, S. (2007). *Métodos Numéricos*. Depto. de Matemática, Facultad de Ciencias, Universidad de Santiago de Chile.