



UNIVERSIDAD DEL PAPALOAPAN

Campus Loma Bonita

INGENIERÍA EN COMPUTACIÓN

**“DESARROLLO DE UN SISTEMA PARA
LA APROXIMACIÓN DE LAS INVERSAS
DE MOORE-PENROSE Y DRAZIN”**

TESIS

QUE PARA OBTENER EL TÍTULO DE:
INGENIERO EN COMPUTACIÓN

PRESENTA:

FABIOLA ARMAS RAMÍREZ

DIRECTOR DE TESIS:

DR. VÍCTOR MANUEL MÉNDEZ SALINAS

LOMA BONITA, OAXACA, MÉXICO

2018

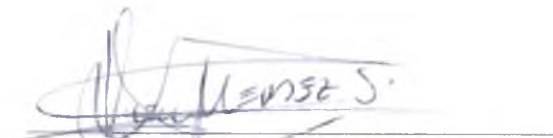


UNIVERSIDAD DEL PAPALOAPAN

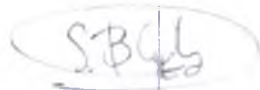
Campus Loma Bonita

INGENIERÍA EN COMPUTACIÓN

LA PRESENTE TESIS TITULADA “DESARROLLO DE UN SISTEMA PARA LA APROXIMACIÓN DE LAS INVERSAS DE MOORE-PENROSE Y DRAZIN” PRESENTADA POR LA SUSTENTANTE DE LA LICENCIATURA C. FABIOLA ARMAS RAMÍREZ BAJO LA DIRECCIÓN DEL DR. VÍCTOR MANUEL MÉNDEZ SALINAS, HA SIDO REVISADA Y ACEPTADA POR EL COMITÉ EXAMINADOR PARA SER DEFENDIDA EN EL EXAMEN PROFESIONAL Y OBTENER EL TÍTULO DE INGENIERO EN COMPUTACIÓN.



DR. VÍCTOR MANUEL MÉNDEZ
SALINAS
DIRECTOR



DRA. BEATRIZ CARELY
LUNA
REVISOR



DR. EDUARDO SÁNCHEZ
SOTO
REVISOR

Dedicado a mi familia

Agradecimientos

Le agradezco a mi madre Juana que siempre ha visto por mis hermanas Flerida, Viviana y por mí, gracias por los valores enseñados y el apoyo incondicional en todo momento, también a Hermenegildo por apoyarnos desde que empezó a ser parte de nuestra familia.

A mi abuela Patricia y mi abuelo Lauriano que me han dado tanto amor como les ha sido posible, por sus consejos de vida y por preocuparse por mí.

A keyri por hacerme reír y a veces enojar con sus travesuras.

A mi asesor de tesis el Dr. Victor Manuel Méndez Salinas por la colaboración y la dirección de esta tesis.

A los profesores que fueron parte de mi formación académica compartiendo sus conocimientos con mis compañeros y conmigo.

Al jefe de carrera el profesor Ariel por preocuparse siempre por los alumnos y ayudarnos en lo que le es posible.

Al profesor Eduardo Soto por ser mi tutor durante casi toda la carrera y brindarme su confianza y consejos durante todo este tiempo.

A Carlos por apoyarme y motivarme en momentos de desesperación durante la elaboración de mi tesis y sobre todo por hacerme parte de su familia.

A mis amigos Belem, Itzamary, Heriberto y Nora que conocí durante mi estancia en la universidad, gracias por darme su amistad, tiempo, cariño y motivación durante todo este tiempo.

Fabiola Armas Ramírez

Abstract

In the last one hundred years, one of the subjects that has taken on a great importance within linear algebra is the study of methods for the solution of systems of linear equations and in specific the matrices that are invertible.

This work focuses especially on cases in which the matrices do not have an associated inverse, this is when need arises to obtain some kind of partial or total inverse of a matrix that is singular and even rectangular. The first concept related to the generalized inverse was in 1903, this concept was coined by Fredholm [18] who called the class of generalized inverse of an integral value as "*pseudoinverse*".

During the following years others authors investigated this subject: Myller in 1906, Westfall and Bounitzky[14] in 1909, Elliott in 1928, and Reid [13] in 1931, just to name a few. The class of all pseudoinverses was characterized in 1912 by Hurwitz[20].

The theory of generalized inverses is used to calculate the inverse of a matrix that is not invertible among other things, this theory also complies with the basic properties of a usual matrix, this inverse matrix is called generalized inverse, although it is not always unique, that is to say, a single matrix can have several generalized inverses determined by using different methods, the inverse most recognized is that of Moore-Penrose.

The generalized inverses of a matrix have played a very important role in areas of mathematics, robotics and statistics, among others. In these applications the use of a computer facilitates the calculation of these inverses. When handling real numbers computer programs do not always represent real numbers in an exact manner because, the number of digits with which the real number is represented is limited by the word size of each computer, therefore, the coomputer restricts the accuracy and precision of real numbers, which leads generating approximation errors.

Glosario de Anotaciones

- A^\dagger – Inversa de Moore-Penrose de A . 11
- $N(T)$ – Espacio nulo de T . 17
- $R(T)$ – Rango de T . 17
- $A_{\{i,j,\dots,k\}}$ – $A_{\{i,j,\dots,k\}}$ -inversas de A . 12
- A^* – Matriz conjugada de A . 8
- λ – Escalar. 9
- $P_{L,S}$ – Proyector en L a lo largo de S . 25
- $A^\#$ – La grupo inversa. 24
- $A^{(i,j,\dots,k)}$ – Una $A^{(i,j,\dots,k)}$ -inversa de A . 12
- H_A – Forma de Hermite de A . 12
- $\text{Ind}(A)$ – índice de A . 21
- $L \oplus S$ – Suma directa de L, S . 25
- λ^\dagger – Inversa de Moore-Penrose de un escalar. 15
- $\mathcal{L}(U, V)$ – Transformación lineal de U a V . 17
- $\mathbb{C}^{m \times n}$ – matrices complejas $m \times n$. 12
- $\mathbb{C}_r^{m \times n}$ – matrices complejas $m \times n$ con $\text{rango}(r)$. 13
- $\text{rango}(A)$ – Rango de A . 15
- A^{-1} – Inversa de una matriz. 8
- A^T – Matriz transpuesta de A . 8

Índice de figuras

2.1. (i, j, k) -inversas mas usuales. Fuente : [15]	18
4.1. Interfaz gráfica del sistema de aproximaciones de inversas generalizadas . . .	52
5.1. Matriz A de prueba para inversa de Moore-Penrose	55
5.2. Carga de la matriz A al sistema	56
5.3. Inversa de Moore-Penrose de la matriz A	57
5.4. Prueba 1 de la inversa de Moore-Penrose utilizando la ecuación (1)	57
5.5. Prueba 2 de la inversa de Moore-Penrose utilizando la ecuación (2)	58
5.6. Prueba 3 de la inversa de Moore-Penrose utilizando la ecuación (3)	58
5.7. Prueba 4 de la inversa de Moore-Penrose utilizando la ecuación (4)	59
5.8. Archivo de texto original y el archivo generado por el sistema	59
5.9. Verificación de singularidad de la matriz A	60
5.10. Matriz A de prueba para inversa de Drazin	61
5.11. Inversa de Drazin de la matriz A (Figura 5.10)	61
5.12. Prueba 1 de la inversa de Drazin utilizando la ecuación (1^k)	62
5.13. Prueba 2 de la inversa de Drazin utilizando la ecuación (5)	62
5.14. Prueba 3 de la inversa de Drazin utilizando la ecuación (5^k)	63
5.15. Prueba 4 de la inversa de Drazin utilizando la ecuación (6^k)	63

Índice general

Agradecimientos	III
Abstract	IV
Glosario de Anotaciones	V
Índice de Figuras	VI
1. Introducción	1
1. Motivación	1
2. Antecedentes	2
3. Planteamiento del problema	4
4. Hipótesis	4
5. Objetivo de la tesis	4
5.1. Objetivo General	5
5.2. Objetivos Específicos	5
6. Alcances y limitaciones	5
7. Justificación	6
2. Preliminares	8
1. Inversa de una matriz	8
2. Inversa generalizada de una matriz	9
2.1. Resolución de sistemas lineales	9
2.2. Ecuaciones de Penrose	11
3. Existencia y construcción de inversas generalizadas	12
3.1. Existencia y construcción de $\{1\}$ -inversas	13
3.2. Propiedades de la $\{1\}$ -inversa	14
3.3. Existencia y construcción de la $\{1,2\}$ -inversas	15

3.4.	Existencia y construcción de las $\{1, 2, 3\}$, $\{1, 2, 4\}$ y $\{1, 2, 3, 4\}$ -inversas	16
3.5.	Fórmula explícita para A^\dagger	18
3.6.	Cálculo de la inversa de Moore-Penrose	19
4.	Inversas generalizadas espectrales	20
4.1.	Índice de una matriz	21
4.2.	Inversa espectral de una matriz diagonal	23
4.3.	La grupo inversa	24
4.4.	Propiedades de la grupo inversa	27
5.	Inversa de Drazin	28
5.1.	Cálculo de la inversa de Drazin	30
5.2.	A^D como un polinomio en A	33
3.	Aspectos computacionales de las inversas generalizadas	37
1.	Introducción	37
2.	Cálculo de las $\{1\}$ y $\{1, 2\}$ -inversas sin restricciones	38
3.	Cálculo de la $\{1, 3\}$ -inversa sin restricciones	39
4.	Desarrollo	41
1.	Alcances	41
2.	Limitaciones	41
3.	Aproximación de la inversa de Moore-Penrose	42
4.	Aproximación de la inversa de Drazin	46
5.	Aproximación a A^D como polinomio en A	49
6.	Interfaz gráfica	51
5.	Pruebas	54
1.	Introducción	54
2.	Inversa de Moore-Penrose	54
3.	Inversa de Drazin	59
6.	Conclusiones	65
	Bibliografía	68

Capítulo 1

Introducción

En este capítulo se muestran los antecedentes de las inversas generalizadas, así como el planteamiento del problema, hipótesis, objetivos, alcances y limitaciones que se establecieron para el desarrollo del sistema de aproximaciones de inversas.

1. Motivación

En los últimos cien años uno de los temas que ha tomado importancia dentro del álgebra lineal es el estudio de matrices para solución de sistemas de ecuaciones lineales y en específico las matrices que son invertibles. Este trabajo se enfoca en los casos en que las matrices no cuentan con una inversa asociada, es aquí cuando surge la necesidad de obtener algún tipo de inversa parcial o total de una matriz que sea singular e incluso rectangular. En 1903 aparece el primer concepto relacionado con la inversa generalizada, este concepto fue acuñado por Fredholm [18] quien llamó a la clase de inversa generalizada de un valor integral como "*pseudoinversa*". Durante los años siguientes más autores investigaron sobre este tema: Myller en 1906, Westfall y Bounitzky [14] en 1909, Elliott en 1928, y Reid [13] en 1931, sólo por nombrar algunos. La clase de todas las pseudoinversas fue caracterizada en 1912 por Hurwitz [20].

La teoría de inversas generalizadas se utiliza entre otras cosas para calcular la inversa de una matriz que no es invertible, que además cumpla con las propiedades básicas de una matriz usual, esta matriz inversa se llama inversa generalizada, y en ocasiones no es única, es decir, una sola matriz puede tener varias inversas generalizadas determinadas por el uso de diferentes métodos, la inversa más conocida es la de Moore-Penrose.

Las inversas generalizadas han jugado un papel muy importante en diversas áreas de las

matemáticas, la robótica y la estadística, entre otras. En estas aplicaciones el uso de la computadora facilita el cálculo de las inversas. Los programas de computadora al manejar números reales no siempre lo hacen de manera exacta, debido a que el número de dígitos con el que se representa el número real está limitado por el tamaño de palabra de cada computadora, por ende, la computadora restringe la precisión y exactitud de los números reales, lo que conlleva a generar errores de aproximación.

En esta tesis se implementarán dos algoritmos para obtener aproximaciones a la inversa Moore-Penrose y la inversa de Drazin respectivamente. Para realizar el sistema se ocupará Matlab [6], debido a que nos facilita el manejo de matrices, además de ser compatible con cualquier sistema operativo. Las inversas generalizadas de matrices existen tanto para números reales como números complejos, en este caso la implementación de los algoritmos se limitará a trabajar con matrices de números reales.

2. Antecedentes

El concepto de inversa generalizada fue mencionado por primera vez en 1903 por Fredholm, en [18] dio una inversa generalizada (llamada por él "pseudoinversa") de un operador integral. La clase de todas las pseudoinversas fue caracterizada en 1912 por Hurwitz quien utilizó la dimensión finita de los espacios nulos de los operadores de Fredholm para dar una construcción algebraica simple. Las inversas generalizadas de los operadores diferenciales mencionados en el trabajo de Hilbert sobre inversas de funciones de Green en 1904 fueron estudiadas por numerosos autores en particular Myller en 1906, Westfall y Bounitzky en 1909, Elliot en 1928 y Reid en 1931.

E. H. Moore [22] definió una inversa única (llamada por él "*recíproco general*") para cada matriz finita (cuadrada o rectangular), esta inversa se incluye en las inversas generalizadas de operadores diferenciales e integrales que anteceden a las inversas generalizadas de las matrices. Aunque su primera publicación sobre el tema fue un resumen de la charla dada en una reunión de la American Mathematical Society en 1920, no se le dio importancia al descubrimiento de Moore durante los primeros 30 años después de su publicación, tiempo durante el cual surgieron trabajos sobre inversas generalizadas aplicadas a matrices por Siegel [26] en 1937 y aplicaciones para operadores por Tseng [27], Murray y Von Neumann en 1936, Atkinson en 1952 y 1953.

El interés sobre las propiedades de los mínimos cuadrados de ciertas inversas generalizadas, hizo que se diera importancia a este tema durante los años cincuenta. Estas propiedades

fueron reconocidas por Bjerhammar quien reavivó el interés sobre la inversa de Moore y también observó la relación de las inversas generalizadas con soluciones de sistemas lineales. En 1955 Penrose [23] extendió los resultados de Bjerhammar en sistemas lineales y mostró que la inversa de Moore para una matriz dada A es la única matriz X que satisface las ecuaciones:

$$AXA = A, \tag{1}$$

$$XAX = X, \tag{2}$$

$$(AX)^* = AX, \tag{3}$$

$$(XA)^* = XA, \tag{4}$$

donde $(XA)^*$ es la matriz conjugada de la multiplicación de XA . Este descubrimiento ha sido muy importante, a esta inversa única algunos autores la llaman inversa de Moore-Penrose. Esta inversa es reflexiva en el sentido de que si A es inversa generalizada de X , entonces X es inversa generalizada de A . Para saber si A es externamente invertible, se debe tener un operador C distinto de cero tal que $C = CAC$, entonces decimos que C es inversa externa para A y que A es externamente invertible. La inversa externa no es única. Algunas inversas generalizadas tienen propiedades espectrales, la inversa de Drazin [16] es una de ellas, que se centra en matrices cuadradas singulares, debido a que, poseen vectores y valores propios, para estas inversas se toman las ecuaciones de Moore-Penrose, complementándolas con las siguientes ecuaciones. La inversa de Drazin sirve de base para las inversas en álgebras de Banach.

$$A^k X A = A^k, \tag{1^k}$$

$$AX = XA, \tag{5}$$

$$A^k X = X A^k, \tag{5^k}$$

$$AX^k = X^k A, \tag{6^k}$$

El conjunto de estas ecuaciones nos permiten construir inversas de Moore-Penrose y Drazin respectivamente. Uno de los temas actuales en esta área son los inversos espectrales, que vienen ligados directamente con las inversas de Drazin y la grupo inversa.

La inversa de Drazin tiene aplicaciones en resolución de sistemas lineales diferenciales, así como en cadenas de Markov, sólo por nombrar algunas. Diferentes autores han propuesto algoritmos para el cálculo de la inversa de Drazin. En [21] se muestra un algoritmo para determinar la inversa de Drazin de una matriz polinomial de una variable mediante la traza, mientras que en [17] se muestra un algoritmo para determinar la inversa de Drazin de matrices polinomiales de dos variables, un artículo que nos muestra un algoritmo básico es [25] donde se ocupa el algoritmo de Greville.

3. Planteamiento del problema

La investigación de las inversas generalizadas en los últimos años ha tomado gran importancia, no sólo para el área de matemáticas sino en algunas ramas de la física, la mecánica, la robótica, la psicología, la estadística, entre otras. Para facilitar el cálculo de inversas generalizadas, que en muchas ocasiones se hace manualmente, se propone el desarrollo de un sistema que entregue aproximaciones a las inversas generalizadas, en particular la inversa generalizada de Moore-Penrose y la inversa generalizada de Drazin, este sistema será de gran ayuda particularmente al departamento de matemáticas de la Universidad del Papaloapan campus Loma Bonita, que tiene investigaciones relacionadas con inversas generalizadas espectrales.

4. Hipótesis

Mediante el desarrollo de un software que entregue aproximaciones de inversas generalizadas, en particular la inversa de Moore-Penrose y la inversa de Drazin, se espera reducir en gran porcentaje los cálculos manuales.

5. Objetivo de la tesis

A manera de mostrar detalladamente el objetivo de la tesis, se describe el objetivo general seguido de los objetivos específicos de la tesis.

5.1. Objetivo General

Desarrollar un sistema en la herramienta de software matemático Matlab que implemente los algoritmos correspondientes para generar aproximaciones a la inversa generalizada de Moore-Penrose y la inversa generalizada de Drazin.

5.2. Objetivos Específicos

- Estudiar la inversa generalizada de Moore-Penrose.
- Estudiar las inversas generalizadas con propiedades espectrales.
- Estudiar la inversa generalizada de Drazin.
- Instalar el entorno de programación.
- Analizar los algoritmos de la inversa de Moore-Penrose y la inversa de Drazin.
- Implementar los algoritmos de la inversa de Moore-Penrose y la inversa de Drazin.
- Verificar resultados de inversas generalizadas aplicando las propiedades pertinentes para cada una de las inversas.

6. Alcances y limitaciones

Esta tesis busca desarrollar un software que entregue aproximaciones a la inversa de Moore-Penrose y la inversa de Drazin, dentro de los cálculos que hará la computadora nos podemos encontrar con algunos problemas como es la precisión de los resultados de las inversas, debido a que las computadoras en su nivel más básico sólo pueden registrar si fluye electricidad o no por cierta parte de su circuito con unos o ceros, por ende, el sistema binario es el que ocupan para su funcionamiento y representación de los números de cualquier sistema numérico, por lo que no todo número puede ser representado en la computadora.

La representación de los números en la computadora está limitado por el tamaño de palabra, por lo que en algunas ocasiones se generan aproximaciones que conllevan errores, estos errores podrían generar incompatibilidad en los sistemas, así que para poder manejar este tipos de errores el Instituto de Ingeniería Eléctrica y Electrónica (IEEE por sus siglas en inglés) generó un estándar para la representación de los números flotantes en la computadora en 1985, que fue aceptado en 1989 como el estándar internacional IEC559, bajo este estándar

los sistemas generan notaciones científicas que ayudan a minimizar el error, además de ser compatible con todo sistema, en [24] se detalla más a fondo esta problemática.

El sistema se desarrollará en Matlab, y para la precisión de la aproximación se ocupará la función `vpa()` [11], que cumple con los lineamientos establecidos por la IEEE, con este sistema se espera ayudar en la investigación y desarrollo de una teoría espectral sobre inversos de anillos y de álgebras debido a que no hay textos exhaustivos que traten estos temas por ser nuevos, además de reducir el tiempo y esfuerzo destinados a cálculos de inversas generalizadas, las dos inversas que entregará el sistema son las más usuales con respecto a las inversas generalizadas, siendo la base para próximas investigaciones.

7. Justificación

La investigación sobre inversas generalizadas va en constante crecimiento, sus aplicaciones son variadas dependiendo de la inversa que se calcule, estas aplicaciones abarcan diferentes áreas de matemáticas, así como ramas de la física, la mecánica, la estadística, la psicología, entre otras. Algunas de las más conocidas son las siguientes:

- Redes eléctricas
- Proyecciones ortogonales
- Modelo estadístico lineal
- Los métodos de Newton para resolver sistemas de ecuaciones no lineales, sin requerir la no singularidad de la matriz Jacobiana
- La grupo inversa a las cadenas de Markov finitas
- La inversa de Drazin para resolver ecuaciones diferenciales
- Distribuciones de probabilidad

En particular, todas las inversas tienen una aplicación compartida: el modelo estadístico lineal.

Aunque existen varios tipos de inversas como la inversa de Boot-Duffin y la inversa de Djorjevic-Wei($p - q$), en esta tesis sólo se abordarán las inversas de Moore-Penrose y de Drazin, en [15] se profundiza la aplicación de la inversa de Drazin en cadenas de Markov, sistemas de ecuaciones diferenciales, entre otros, además de ejemplificar algunas de estas

aplicaciones. Hasta ahora sólo existen librerías para aproximación de la inversa de Moore-Penrose, por lo que en este trabajo se desarrollará un software que entregue aproximaciones a las inversas mencionadas. Las inversas generalizadas pueden ser aplicadas a matrices con números reales o números complejos, pero el sistema se limitará a las inversas de matrices de números reales. Para la implementación de los algoritmos se buscó una plataforma que fuera compatible con cualquier sistema operativo, en este caso Matlab que es una herramienta de software matemático.

Capítulo 2

Preliminares

En este capítulo se abordarán conceptos básicos sobre inversas generalizadas, como lo son la existencia y construcción de inversas, la grupo inversa, inversas espectrales, inversa de Moore-Penrose e inversa de Drazin así como las propiedades de cada una de ellas.

1. Inversa de una matriz

Una matriz A tiene una inversa si y sólo sí, es cuadrada y si además es no singular, o en otras palabras, si sus columnas o filas son linealmente independientes. Esta inversa única A^{-1} cumple:

$$AA^{-1} = A^{-1}A = I, \quad (1)$$

donde I es la matriz identidad. A continuación se mencionan algunas propiedades de una matriz inversa [12]:

$$\begin{aligned} (A^{-1})^{-1} &= A, \\ (A^T)^{-1} &= (A^{-1})^T, \\ (A^*)^{-1} &= (A^{-1})^*, \\ (AB)^{-1} &= B^{-1}A^{-1}, \end{aligned}$$

Donde A^T es la matriz transpuesta y A^* es la matriz transpuesta conjugada de A . Un número complejo o real λ es llamado un valor propio de una matriz cuadrada, y un vector no nulo x

es llamado vector propio de A correspondiente a λ si:

$$Ax = \lambda x.$$

Otra propiedad de la inversa A^{-1} es que sus valores propios son recíprocos a la matriz A .

2. Inversa generalizada de una matriz

En el siglo pasado en numerosas áreas de matemáticas se intentaba encontrar una inversa parcial de una matriz que fuera singular o rectangular. Por la inversa generalizada de una matriz A nos referimos a una matriz X asociada de alguna manera a A tal que:

- I) exista para una clase de matrices mayor que la clase de matrices no singulares;
- II) tenga algunas de las propiedades de la inversa usual;
- III) se reduzca a la inversa usual cuando A sea no singular.

El último inciso se puede ilustrar de la siguiente manera:

$$AXA = A \tag{2}$$

Si A fuera no singular el valor de X sería A^{-1} , ya que

$$AA^{-1}A = A.$$

2.1. Resolución de sistemas lineales

La aplicación más conocida y usada de matrices es la resolución de sistemas de ecuaciones lineales de la forma:

$$Ax = b, \tag{3}$$

donde b es un vector conocido y x es un vector desconocido. Si A es no singular, entonces tenemos una única solución para el vector x :

$$x = A^{-1}b.$$

Por otro lado, cuando A puede ser singular o rectangular, a veces no hay solución o multiplicidad de soluciones.

La existencia de un vector x que satisface (3) es equivalente a decir que b es alguna combinación de columnas de A . Si A es $m \times n$ y el rango de la matriz es menor que m , puede que no sea así. Si no se cumple con lo anterior, entonces hay algún vector h tal que:

$$b = Ah.$$

Ahora, si X es alguna matriz que satisface (2), y si tomamos: $x = Xb$ tenemos que:

$$Ax = AXb = AXAh = Ah = b,$$

y por tanto x satisface (3).

En el caso general cuando (3) tiene muchas soluciones se puede decidir por una caracterización de todas ellas. Se ha demostrado que si X es cualquier matriz que satisface $AXA = A$, entonces $Ax = b$ tiene una solución sí y solo sí:

$$AXb = b.$$

En este caso la solución general es:

$$x = Xb + (I - XA)y, \tag{4}$$

donde y es arbitraria.

Este hecho se puede ilustrar con el siguiente ejemplo:

Ejemplo 2.1 Encuentre todas las soluciones al sistema:

$$\begin{bmatrix} 5 & 2 & -1 & 2 \\ 2 & 2 & 3 & 1 \\ 1 & 1 & 4 & -1 \\ 2 & -1 & -3 & -1 \\ 3 & 0 & 1 & -2 \end{bmatrix} x = \begin{bmatrix} 7 \\ 9 \\ 5 \\ -6 \\ -1 \end{bmatrix}$$

Solución: Determinando una inversa generalizada de la matriz de coeficientes tenemos:

$$G = \frac{1}{15} \begin{bmatrix} 5 & -9 & 8 & 0 & 0 \\ -5 & 21 & -17 & 0 & 0 \\ 0 & -3 & 6 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Por lo tanto al sustituir en la fórmula de todas las soluciones

$$x = Xb + (I - XA)y$$

obtenemos:

$$x = \frac{1}{15} \begin{bmatrix} 5 & -9 & 8 & 0 & 0 \\ -5 & 21 & -17 & 0 & 0 \\ 0 & -3 & 6 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} 7 \\ 9 \\ 5 \\ -6 \\ -1 \end{bmatrix} + \begin{bmatrix} 0 & 0 & 0 & \frac{7}{15} \\ 0 & 0 & 0 & -\frac{28}{15} \\ 0 & 0 & 0 & \frac{9}{15} \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} y1 \\ y2 \\ y3 \\ y4 \end{bmatrix}$$

$$x = \frac{1}{15} \begin{bmatrix} -6 + 7y4 \\ 69 - 28y4 \\ 3 + 9y4 \\ 15y4 \end{bmatrix}$$

2.2. Ecuaciones de Penrose

En 1955, Penrose demostró que, para toda matriz finita A (cuadrada o rectangular) de elementos reales o complejos, existe una matriz única X que satisface las siguientes cuatro ecuaciones, llamadas ecuaciones de Penrose [12].

$$AXA = A, \tag{1}$$

$$XAX = X, \tag{2}$$

$$(AX)^* = AX, \tag{3}$$

$$(XA)^* = XA \tag{4}$$

Donde A^* denota la matriz transpuesta conjugada de A . Esta inversa (única) fue estudiada por E.H. Moore denotando A^\dagger como la inversa generalizada de Moore-Penrose. Si A es no singular, entonces $X = A^{-1}$ satisface trivialmente las cuatro ecuaciones. Se deduce que la inversa de Moore-Penrose de una matriz no singular es la misma que la inversa ordinaria. Con base en lo anterior podemos establecer la siguiente definición:

Definición 2.1 Para cualquier $A \in \mathbb{C}^{m \times n}$, $A\{i, j, \dots, k\}$ denotará el conjunto de matrices $X \in \mathbb{C}^{m \times n}$ que satisface las ecuaciones (i), (j), ..., (k) de las ecuaciones (1) - (4). Una matriz $X \in A\{i, j, \dots, k\}$ es llamada un $\{i, j, \dots, k\}$ - inversa de A , y también es denotada por $A^{(i, j, \dots, k)}$.

3. Existencia y construcción de inversas generalizadas

Antes de iniciar la construcción de alguna inversa es importante conocer la construcción de la forma normal de Hermite de cualquier matriz.

Definición 2.2 Una matriz $H \in \mathbb{C}^{n \times n}$ se dice que tiene la forma de Hermite si sus elementos h_{ij} satisfacen las condiciones siguientes:

- (I) H es triangular superior (esto es $h_{ij} = 0$ cuando $i > j$).
- (II) h_{ii} es 0 o 1.
- (III) Si $h_{ii} = 0$ entonces $h_{ik} = 0$ para cada k , $1 \leq k \leq n$.
- (IV) Si $h_{ii} = 1$ entonces $h_{ki} = 0$ para cada k , $k \neq i$.

Por ejemplo, la matriz:

$$H = \begin{bmatrix} 1 & 2 & 0 & 3 & 5 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & -2 & 4 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 3 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

tiene la forma de Hermite.

A continuación se muestran algunos datos sobre la forma de Hermite:

- (H1) Una matriz A puede ser reducida a la forma de Hermite. Si A se reduce a su forma escalonada siempre se puede realizar una permutación de las filas para obtener la forma de Hermite.
- (H2) Para una matriz A , su forma de Hermite H_A es única si es obtenida por reducción de fila.

(H3) $H_A^2 = H_A$ (por lo tanto H_A es una proyección).

(H4) $N(A) = N(H_A) = R(I - H_A)$ y una base para $N(A)$ es el conjunto de columnas distintas de cero de $I - H_A$

Esta definición es muy importante, debido a que gran parte de los cálculos de inversas, utilizan la forma reducida de Hermite.

3.1. Existencia y construcción de $\{1\}$ -inversas

Es fácil construir una $\{1\}$ -inversa de la matriz $R \in \mathbb{C}_r^{m \times n}$, $r > 0$ dada por

$$R = \begin{bmatrix} I_r & K \\ O & O \end{bmatrix}. \quad (5)$$

Para cualquier $L \in \mathbb{C}^{(n-r) \times (m-r)}$, la matriz $n \times m$

$$S = \begin{bmatrix} I_r & O \\ O & L \end{bmatrix},$$

es una $\{1\}$ -inversa de (5). Si R es de rango columna (fila) completo, las dos submatrices inferiores (de la derecha) se interpretan como nulas. La construcción de $\{1\}$ -inversas para $A \in \mathbb{C}^{m \times n}$ se simplifica transformando A a su forma normal de Hermite, como se muestra en el teorema siguiente:

Teorema 1 Si tenemos $A \in \mathbb{C}_r^{m \times n}$, $E \in \mathbb{C}_m^{m \times m}$ y $P \in \mathbb{C}_n^{n \times n}$, tal que:

$$EAP = \begin{bmatrix} I_r & K \\ O & O \end{bmatrix}. \quad (6)$$

entonces, para cualquier $L \in \mathbb{C}^{(n-r) \times (m-r)}$, la matriz $n \times m$:

$$X = P \begin{bmatrix} I_r & O \\ O & L \end{bmatrix} E \quad (7)$$

es una $\{1\}$ -inversa de A . La matriz particionada en (6) y (7) debe interpretarse adecuadamente en el caso $r = m$ o $r = n$.

3.2. Propiedades de la $\{1\}$ -inversa

Para una matriz A , denotaremos cualquier $\{1\}$ -inversa por $A^{(1)}$. Nótese que en general $A^{(1)}$ no es única, para ejemplificar este caso tomaremos la propiedad $AXA = A$.

Ejemplo 2.2 Si tenemos :

$$A = \begin{bmatrix} 1 & 1 \\ 0 & 0 \end{bmatrix}$$

la matriz inversa X asociada a A se define como:

$$X = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix}$$

de esta manera al resolver la multiplicación obtenemos:

$$\begin{aligned} (AX)A &= \left(\begin{bmatrix} 1 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \right) \begin{bmatrix} 1 & 1 \\ 0 & 0 \end{bmatrix} \\ &= \begin{bmatrix} a_{11} + a_{21} & a_{11} + a_{21} \\ 0 & 0 \end{bmatrix} \begin{bmatrix} 1 & 1 \\ 0 & 0 \end{bmatrix} \\ &= \begin{bmatrix} a_{11} + a_{21} & a_{11} + a_{21} \\ 0 & 0 \end{bmatrix}. \end{aligned}$$

Podemos observar que el resultado nos pide que la suma de $a_{11} + a_{21} = 1$, para que el resultado de la multiplicación sea A , mientras que a_{12} y a_{22} son iguales a cero. Este es un claro ejemplo donde la matriz X no es única, podemos construir varias matrices de 2×2 que satisfagan $a_{11} + a_{21} = 1$, y de esta manera cumplen la condición $AXA = A$:

Algunas matrices X serían:

$$X = \begin{bmatrix} \frac{1}{5} & 0 \\ \frac{4}{5} & 0 \end{bmatrix} \quad \text{o} \quad X = \begin{bmatrix} \frac{1}{2} & 0 \\ \frac{1}{2} & 0 \end{bmatrix}.$$

Para cualquier escalar λ se define λ^\dagger :

$$\lambda^\dagger = \begin{cases} \lambda^{-1} & \text{si } \lambda \neq 0 \\ 0 & \text{si } \lambda = 0 \end{cases} \quad (8)$$

Las propiedades más importantes de las $\{1\}$ -inversas son dados en el siguiente lema:

Lema 1 Si tenemos $A \in \mathbb{C}_r^{m \times n}$, $\lambda \in \mathbb{C}$. Entonces:

(a) $(A^{(1)})^* \in A^* \{1\}$.

(b) Si A es no singular, $A^{(1)} = A^{-1}$ (de manera única).

(c) $\lambda^\dagger A^{(1)} \in (\lambda A) \{1\}$.

(d) $\text{rango}(A^{(1)}) \geq \text{rango}(A)$.

(e) Si S y T son no singulares, $T^{-1}A^{(1)}S^{-1} \in SAT\{1\}$.

(f) $AA^{(1)}$ y $A^{(1)}A$ son idempotentes y tienen el mismo el rango que A .

Si una matriz A $m \times n$ es rango columna completo, su $\{1\}$ -inversa coincide con su inversa izquierda. Si es rango fila completo entonces su $\{1\}$ -inversa es su inversa derecha.

Lema 2 Si tenemos $A \in \mathbb{C}_r^{m \times n}$, entonces:

(a) $A^{(1)}A = I_n$, sí y sólo si $r = n$.

(b) $AA^{(1)} = I_m$, sí y sólo si $r = m$.

3.3. Existencia y construcción de la $\{1, 2\}$ -inversas

Bjerhammar fue el primero en señalar que la existencia de $\{1\}$ -inversa de una matriz A implica la existencia de $\{1, 2\}$ -inversa. Esto es fácil de verificar con el siguiente lema:

Lema 3 Si $Y, Z \in A\{1\}$ y se cumple

$$X = YAZ,$$

entonces $X \in A\{1, 2\}$.

Como las matrices A y X ocurren simétricamente en (1) y (2), entonces $X \in A\{1, 2\}$ es equivalente a $A \in X\{1, 2\}$, y en cualquier caso podemos decir que A y X son $\{1, 2\}$ -inversas entre sí.

De las ecuaciones (1) y (2), y del hecho de que el rango del producto de las matrices no excede el rango de ningún factor, se tiene que si A y X son $\{1, 2\}$ -inversas entre sí, entonces tienen el mismo rango. Bjerhammar demostró que si X es $\{1\}$ -inversa de A y del mismo rango que A entonces es una $\{1, 2\}$ -inversa de A .

Teorema 2 (Bjerhammar). Dada A y $X \in A\{1\}$, $X \in A\{1, 2\}$ si y sólo si el rango de X es igual al rango de A .

Para ejemplificar este teorema podemos tomar las matrices A y X del ejemplo 2.2:

$$A = \begin{bmatrix} 1 & 1 \\ 0 & 0 \end{bmatrix} \quad X = \begin{bmatrix} \frac{1}{5} & 0 \\ \frac{4}{5} & 0 \end{bmatrix}$$

El teorema nos dice que para que $X \in A\{1\}$ implique $X \in A\{1, 2\}$ se debe cumplir lo siguiente:

$$\text{rango}(X) = \text{rango}(A)$$

y tenemos que $\text{rango}(A) = 1$ y el $\text{rango}(X) = 1$, por lo tanto X es $\{1\}$ y $\{1, 2\}$ -inversa.

3.4. Existencia y construcción de las $\{1, 2, 3\}$, $\{1, 2, 4\}$ y $\{1, 2, 3, 4\}$ -inversas

Bjerhammar mostró que la existencia de $\{1\}$ -inversa implica la existencia de $\{1, 2\}$ -inversas, Urquhart demostró que la existencia de $\{1\}$ -inversa para cada matriz finita con elementos en \mathbb{C} implica la existencia de $\{1, 2, 3\}$ -inversas y $\{1, 2, 4\}$ -inversas de cada matriz similar.

Para la demostración se utiliza el siguiente lema:

Lema 4 Para cualquier matriz finita A .

$$\text{rango}(AA^*) = \text{rango}(A) = \text{rango}(A^*A).$$

Teorema 3 (Urquhart) Para cualquier matriz finita A con elementos complejos,

$$Y = (A^*A)^{(1)}A^* \in A\{1, 2, 3\} \tag{9}$$

y

$$Z = A^*(AA^*)^{(1)} \in A\{1, 2, 4\}. \tag{10}$$

Una $\{1, 2\}$ -inversa de una matriz A , es por supuesto, una $\{2\}$ -inversa y de manera similar una $\{1, 2, 3\}$ -inversa es también $\{1, 3\}$ -inversa y una $\{2, 3\}$ -inversa. Así, podemos decir que al establecer la existencia de una $\{1, 2, 3, 4\}$ -inversa podemos demostrar la existencia de una $\{i, j, \dots, k\}$ -inversa para todas las posibles selecciones de uno, dos o tres enteros i, j, \dots, k del conjunto $\{1, 2, 3, 4\}$.

Teorema 4 (Urquhart) Para cualquier matriz finita A de elementos complejos,

$$A^{(1,4)}AA^{(1,3)} = A^\dagger. \quad (11)$$

Dentro de la construcción de las inversas, es necesario tener claro lo que es el rango de la matriz y el rango del espacio. El rango de una matriz denotado como $\text{rango}(A)$ es el número de columnas(filas) que son linealmente independientes, mientras que para el rango del espacio se necesita una transformación lineal:

Si tenemos dos espacios vectoriales U, V sobre un campo \mathbb{F} y un mapeo $T : U \rightarrow V$ podemos decir que T es lineal o una transformación lineal, si $T(\alpha x + y) = \alpha Tx + Ty$, para toda $\alpha \in \mathbb{F}$ y $x, y \in U$. El conjunto de transformaciones lineales de U a V son denotadas por $\mathcal{L}(U, V)$. Sea $T \in \mathcal{L}(U, V)$. Para cualquier $u \in U$, el punto Tu en V es llamado la imagen de u (sobre T). El rango de T se denota por $R(T)$, para evitar confusión nos referiremos a el como la imagen de T :

$$R(T) = \{v \in V : v = Tu \text{ para cualquier } u \in U\}.$$

En particular el espacio nulo de T , denotado por $N(T)$, es la imagen inversa del vector $0 \in V$.

$$N(T) = \{u \in U : Tu = 0\}.$$

La información anterior se resume en el cuadro siguiente, donde se muestran éstas inversas usuales junto con sus propiedades más significativas.

Tipo de inversa	Terminología	Propiedades
{1}-inversa	Inversa que resuelve ecuaciones (Algunas veces es llamada una g -inversa)	$G \in A\{1\}$ si y sólo si Gb es una solución de $Ax = b$ para cada $b \in R(A)$
{1, 2}-inversa	Inversa con rango y espacio nulo fijo (La (N, R) -inversa) Algunos autores la llaman inversa reflexiva.	Si $G \in A\{1, 2\}$ entonces $N(A) + R(G) = \mathbb{C}^n$ y $R(A) + N(G) = \mathbb{C}^m$. Esto es, cada $(1, 2)$ -inversa define un subespacio complementario para $N(A)$ así como para $R(A)$. Recíprocamente cada par (N, R) , donde N y R son complementos de $N(R)$ y $R(A)$ respectivamente, determina de forma única a $(1, 2)$ -inversa, $G_{N,R}$ con $R(G_{N,R}) = N$ y $N(G_{N,R}) = R$.
{1, 3}-inversa	Inversa de mínimos cuadrados.	$G \in A\{1, 3\}$ si y sólo si Gb es una solución de mínimos cuadrados de $Ax = b$ para cada $b \in \mathbb{C}^m$
{1, 4}-inversa	Inversa de norma mínima.	$G \in A\{1, 4\}$ si y sólo si Gb es una solución de norma mínima de $Ax = b$ para cada $b \in R(A)$
{1, 2, 3, 4}-inversa	Inversa de Moore-Penrose A^\dagger .	$\{1, 2, 3, 4\}$ contiene exactamente un elemento, A^\dagger , que es $(R(A^*), N(A^*))$ -inversa para A . $A^\dagger b$ es la solución por mínimos cuadrados de norma mínima de $Ax = b$. Si $b \in R(A)$, entonces $A^\dagger b$ es la solución de norma mínima.

Figura 2.1: (i, j, k) -inversas mas usuales. Fuente : [15]

Las (i, j, k) -inversas son también llamadas inversas que resuelven ecuaciones lineales, debido a que, una de sus características fundamentales es que proporcionan algún tipo de solución, o solución por mínimos cuadrados, para sistemas lineales de ecuaciones algebraicas.

3.5. Fórmula explícita para A^\dagger

C.C MacDuffee aparentemente fue el primero que señaló en 1959, que una factorización de una matriz A de rango completo lleva a la fórmula explícita para la inversa de Moore-Penrose, A^\dagger .

Teorema 5 (MacDuffee). Si $A \in \mathbb{C}_r^{m \times n}$, $r > 0$, tiene una factorización de rango completo de la forma

$$A = FG, \tag{12}$$

entonces:

$$A^\dagger = G^*(F^*AG^*)^{-1}F^*. \tag{13}$$

3.6. Cálculo de la inversa de Moore-Penrose

Tomamos la ecuación (13) en donde tenemos que:

$$A^\dagger = G^*(F^*AG^*)^{-1}F^*$$

Donde F y G es la factorización de A . Basados en el algoritmo que se encuentra en [15] se hace el cálculo de la inversa de Moore-Penrose:

Algoritmo 1 Para obtener la inversa generalizada de una matriz cuadrada $A \in \mathbb{C}^{n \times n}$:

- (I) Se reduce A a su forma Hermitiana H_A .
- (II) Se construye F con las filas diferentes de cero de H_A .
- (III) Se construye G tomando como referencia las columnas pivotes de H_A , y tomando estas mismas columnas de la matriz A .
- (IV) Teniendo estas matrices ahora se determina $A^\dagger = G^*(F^*AG^*)^{-1}F^*$.

Para la notación $()^*$ denota las matrices transpuestas.

Ejemplo 2.3 Sea

$$A = \begin{bmatrix} 1 & 0 & -1 & 1 \\ 0 & 2 & 2 & 2 \\ -1 & 4 & 5 & 3 \end{bmatrix}$$

Al obtener la forma reducida de hermite de la matriz A queda:

$$H_A = \begin{bmatrix} 1 & 0 & -1 & 1 \\ 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix}.$$

Mediante la matriz H_A podemos obtener la matriz G y la matriz F , donde G se obtiene eliminando de H_A los renglones de ceros, mientras que F es el resultado de la extracción de las columnas de A que coinciden con las columnas pivotes de H_A . Por tanto,

$$G = \begin{bmatrix} 1 & 0 & -1 & 1 \\ 0 & 1 & 1 & 1 \end{bmatrix}$$

$$F = \begin{bmatrix} 1 & 0 \\ 0 & 2 \\ -1 & 4 \end{bmatrix},$$

$$F^*AG^* = \begin{bmatrix} 6 & -12 \\ -12 & 60 \end{bmatrix},$$

donde:

$$A^\dagger = G^*(F^*AG^*)^{-1}F^* = \begin{bmatrix} \frac{5}{18} & \frac{1}{9} & -\frac{1}{18} \\ \frac{1}{18} & \frac{1}{18} & \frac{1}{18} \\ -\frac{2}{9} & -\frac{1}{18} & \frac{1}{9} \\ -\frac{1}{3} & -\frac{1}{6} & 0 \end{bmatrix}.$$

4. Inversas generalizadas espectrales

Algunas inversas generalizadas de matrices no singulares tienen propiedades espectrales, éstas propiedades están relacionadas con sus valores propios. Sólo las matrices que son cuadradas se consideran, debido a que, son las que tienen valores y vectores propios.

Si A es no singular, es fácil ver que cada vector propio de A asociado con el valor propio λ , es también un vector propio de A^{-1} asociado con el valor propio λ^{-1} . (Recordemos que una matriz no singular no tiene un 0 como un valor propio).

Una matriz $A \in \mathbb{C}_r^{n \times n}$ que no es diagonal no tiene n vectores propios linealmente independientes, sin embargo, tiene n vectores principales linealmente independientes.

No hay dificultad en mostrar esto, si A es no singular, un vector x es un λ^{-1} -vector de A^{-1} de grado p si y sólo si es un λ -vector de A de grado p . Existen inversas generalizadas de matrices singulares que comparten propiedades espectrales. Para mostrar estas propiedades recordaremos las ecuaciones de Penrose (1) - (4), que se complementarán con las siguientes ecuaciones aplicadas únicamente a matrices cuadradas.

$$A^k X A = A^k, \tag{1^k}$$

$$A X = X A, \tag{5}$$

$$A^k X = X A^k, \tag{5^k}$$

$$AX^k = X^k A, \quad (6^k)$$

En estas ecuaciones k es un entero positivo que es el índice de A , se tiene entonces $\{1^k, 2, 5\}$ -inversa de A , esta inversa es llamada inversa de Drazin y tiene propiedades espectrales que la hacen extremadamente importante en muchas aplicaciones.

4.1. Índice de una matriz

El conjunto de las tres ecuaciones (1^k), (2) y (5), son equivalentes al siguiente conjunto de ecuaciones:

$$AX = XA \quad (5)$$

$$A^{k+1}X = A^k \quad (14)$$

$$AX^2 = X. \quad (15)$$

Esto se puede demostrar si se toma (14) para algún entero positivo k , entonces se aplica para cualquier entero $\ell > k$:

$$\text{rango}(A^k) = \text{rango}(A^{(k+1)}). \quad (16)$$

Por lo tanto, una solución X para (14) (y consecuentemente del conjunto de (5), (14), (15)) existe sólo si se tiene (16). En esta conexión, la siguiente definición es de gran utilidad.

Definición 2.3 *El menor entero positivo k que cumple (16) es llamado el índice de A y se denota $\text{Ind}(A)$.*

El siguiente lema agrupa algunas propiedades del índice de la matriz que se necesita:

Lema 5 *Si tenemos $A \in \mathbb{C}^{n \times n}$ e $\text{Ind}(A) = k$. Entonces:*

- (a) *Todas las matrices $\{A^\ell : \ell \geq k\}$ tienen el mismo rango, el mismo rango(imagen) y el mismo espacio nulo.*
- (b) *Las matrices transpuestas de $\{(A^\ell)^T : \ell \geq k\}$ todos tienen el mismo rango(imagen), el mismo rango y el mismo espacio nulo.*
- (c) *Las matrices transpuestas conjugadas $\{(A^\ell)^* : \ell \geq k\}$ todas tienen el mismo rango, el mismo rango(imagen) y el mismo espacio nulo.*

(d) Además, para ningún ℓ más pequeño que k , A^ℓ y cualquier potencia mayor de A (o sus transpuestas o transpuestas conjugadas) tienen el mismo rango(imagen) o el mismo espacio nulo.

Para probar el lema anterior:

(a) Se sigue de la ecuación (16) que

$$R(A^{k+1}) = R(A^k) \quad (17)$$

Por lo tanto, se cumple (16) para cualquier X , y multiplicándolo por la izquierda por $A^{\ell-k}$ tenemos

$$A^\ell = A^{\ell+1}X \quad (\ell \geq k). \quad (18)$$

Se deduce de (18) que todas las matrices $\{A^\ell : \ell \geq k\}$ tienen el mismo rango(imagen) y el mismo espacio nulo.

(b) y (c) Se obtienen al aplicar (a) a A^T y A^* observando que $(A^\ell)^T = (A^T)^\ell$ y $(A^\ell)^* = (A^*)^\ell$.

(d) Si se cumple (d), debe existir alguna $\ell < k$ tal que A^ℓ o su transpuesta o la transpuesta conjugada tengan el mismo rango que el de la matriz correspondiente con el exponente $\ell + 1$. Pero esto implica que $\text{rango}(A^\ell) = \text{rango}(A^{\ell+1})$ y k no sería el índice de A . De igual forma pasa con la igualdad de los espacios nulos, esto implicaría que A^ℓ y $A^{\ell+1}$ tienen la misma nulidad y por lo tanto el mismo rango(imagen).

Teorema 6 Sea $A \in \mathbb{C}^{n \times n}$. Entonces las siguientes afirmaciones son equivalentes:

(a) $\text{Ind}(A) = k$.

(b) El exponente positivo más pequeño para el cual (14) se mantiene es k .

(c) Si A es singular y $m(\lambda)$ es su polinomio mínimo, k es la multiplicidad de $\lambda = 0$ como cero de $m(\lambda)$.

(d) Si A es singular, k es el grado máximo de 0-vectores de A .

Probar que (a) \iff (b) Claramente (18) implica:

$$\text{rango}(A^{\ell+1}) = \text{rango}(A^\ell), \quad (19)$$

Y con base en (19) se deduce:

$$R(A^{\ell+1}) = R(A^\ell),$$

Por lo tanto (18) se cumple. Así (19) y (18) son equivalentes, demostrando (a).
 (b) \iff (c) Sea:

$$m(\lambda) = \lambda^\ell p(\lambda)$$

donde $p(0) \neq 0$. Sea k como en (b), y ahora se puede mostrar que $k = \ell$. Tenemos

$$p(A)A^\ell = O$$

Si $\ell > k$, entonces

$$O = p(A)A^\ell X = p(A)A^{\ell-1},$$

donde $\lambda^{\ell-1}p(\lambda)$ es de menor grado que $m(\lambda)$ contrario a la definición del polinomio mínimo. Puesto que $p(0) \neq 0$ podemos escribir

$$m(\lambda) = c\lambda^\ell(1 - \lambda q(\lambda)), \quad (20)$$

donde $c \neq 0$ y q es un polinomio. Resultando

$$A^{\ell+1}q(A) = A^\ell. \quad (21)$$

Si $\ell < k$, esto sería una contradicción con (b).

(a) \iff (d) Sea k el índice de A y h el grado máximo de los 0-vectores de A . Podemos demostrar que $h = k$. La definición de h implica que $N(A^\ell) = N(A^h)$ para todo $\ell \geq h$, pero $N(A^{h-1})$ es un subespacio propio de $A(A^h)$. Con esto se deduce a partir del lema (5) que $h = k$.

4.2. Inversa espectral de una matriz diagonal

Las inversas de matrices diagonales nos permiten entender mejor las inversas generalizadas de matrices singulares, debido a que, son más fáciles de tratar. Para poder analizar las inversas de matrices diagonales se tiene que hacer una extensión de las propiedades espectrales de una matriz, porque la matriz singular tiene un 0 como vector propio.

Dada una matriz diagonal $A \in \mathbb{C}^{n \times n}$ se busca una matriz X tal que, cada vector propio de A esta asociado con un valor propio λ (para cada λ en el espectro de A) que es también un vector propio de X asociado con un valor propio λ^\dagger , donde λ^\dagger está definida en (8).

Dado que A tiene n vectores propios linealmente independientes, existe una matriz no singular P , que tiene un conjunto de vectores propios como columnas, tal que

$$AP = PJ, \quad (22)$$

donde

$$J = \text{diag}(\lambda_1, \lambda_2, \dots, \lambda_n)$$

es una forma de Jordan de A . Se necesitará una matriz diagonal obtenida de J reemplazando cada elemento diagonal λ_i por λ_i^\dagger . De esta manera podemos tener la inversa de Moore-Penrose J^\dagger ,

$$J^\dagger = \text{diag}(\lambda_1^\dagger, \lambda_2^\dagger, \dots, \lambda_n^\dagger).$$

Debido a la exigencia espectral en X , debemos tener

$$XP = PJ^\dagger. \quad (23)$$

Resolviendo (22) y (23) para A y X se obtiene

$$A = PJP^{-1}, \quad X = PJ^\dagger P^{-1}. \quad (24)$$

Donde J y J^\dagger si son diagonales y conmutan entre sí. Así, de (24) se sigue que $X \in A\{1, 2, 5\}$.

4.3. La grupo inversa

La grupo inversa contiene $\{1, 2, 5\}$ -inversas de A y se denota como $A^\#$, la grupo inversa es única y no está restringida a matrices diagonales; sin embargo, no existe para todas las matrices cuadradas. Tal inversa existe, si y sólo si $R(A)$ y $N(A)$ son subespacios complementarios, esto es equivalente a que el índice de A sea 1. Se tiene por tanto, el siguiente teorema:

Teorema 7 *Una matriz cuadrada A tiene una grupo de inversas si y sólo si $\text{Ind}(A) = 1$, es decir:*

$$\text{rango}(A) = \text{rango}(A)^2 \quad (25)$$

Cuando la grupo inversa existe es única.

El teorema siguiente proporciona una condición alternativa para la existencia de $A^\#$ y a su vez una fórmula explícita para su cálculo.

Teorema 8 *Sea una matriz cuadrada A con factorización de rango completo*

$$A = FG. \quad (26)$$

Entonces A tiene una grupo inversa sí y sólo si GF es no singular, en cuyo caso:

$$A^\# = F(GF)^{-2}G. \quad (27)$$

Para una clase importante de matrices la grupo inversa y la inversa de Moore-Penrose son las mismas. Diremos que una matriz cuadrada A es de rango Hermitiano (tal matriz también es llamada EP_r o EP) si:

$$R(A^*) = R(A) \quad (28)$$

o equivalente, si

$$N(A^*) = N(A). \quad (29)$$

De manera más explícita:

Teorema 9 Sea $A \in \mathbb{C}_r^{m \times n}$, $R(A) = L$, $N(A) = M$, $L \oplus S = \mathbb{C}^m$, y $M \oplus T = \mathbb{C}^n$. Entonces:

(a) X es $\{1\}$ -inversa de A tal que $N(AX) = S$ y $R(XA) = T$ sí y sólo sí

$$AX = P_{L,S}, \quad XA = P_{T,M}. \quad (30)$$

(b) La solución general de (30) es

$$X = P_{T,M}A^{(1)}P_{L,S} \times (I_n - A^{(1)}A)Y(I_m - AA^{(1)}), \quad (31)$$

donde $A^{(1)}$ es un elemento fijo (pero arbitrario) de $A\{1\}$ y Y es una elemento arbitrario de $\mathbb{C}^{n \times m}$.

(c) $A_{T,S}^{(1,2)} = P_{T,M}A^{(1)}P_{L,S}$ es la única $\{1, 2\}$ -inversa de A teniendo rango(imagen) T y espacio nulo S .

Se propone otro teorema en base al anterior y teniendo en cuenta que:

$$A = A_{R(A)N(A)}^{(1,2)}$$

y

$$A^\dagger = A_{R(A^*)N(A^*)}^{(1,2)}$$

son iguales, si y sólo si $R(A) = R(A^*)$ y $N(A) = N(A^*)$.

Teorema 10 $A^\# = A^\dagger$ sí y sólo si A es de rango Hermitiano.

Ejemplo 2.4 Sea

$$A = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix}$$

determinar $A^\dagger = G^*(F^*AG^*)^{-1}F^*$, la factorización de A es:

$$G = \begin{bmatrix} 1 & 0 & -1 \\ 0 & 1 & 2 \end{bmatrix}, \quad F = \begin{bmatrix} 1 & 2 \\ 4 & 5 \\ 7 & 8 \end{bmatrix},$$

Ahora solo hacemos las multiplicaciones matriciales.

$$A^\dagger = \left(\begin{bmatrix} 1 & 0 \\ 0 & 1 \\ -1 & 2 \end{bmatrix} \right) \left(\begin{bmatrix} 1 & 4 & 7 \\ 2 & 5 & 8 \end{bmatrix} \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ -1 & 2 \end{bmatrix} \right)^{-1} \left(\begin{bmatrix} 1 & 4 & 7 \\ 2 & 5 & 8 \end{bmatrix} \right),$$

y obtenemos la inversa de Moore-Penrose

$$A^\dagger = \begin{bmatrix} \frac{-23}{36} & \frac{-1}{6} & \frac{11}{36} \\ \frac{-1}{18} & 0 & \frac{1}{18} \\ \frac{19}{36} & \frac{1}{6} & \frac{-7}{36} \end{bmatrix}.$$

Para determinar la grupo inversa de A , aplicamos la ecuación (27) y tomamos las matrices F y G del cálculo de la inversa de Moore-Penrose :

$$A^\# = \begin{bmatrix} 1 & 2 \\ 4 & 5 \\ 7 & 8 \end{bmatrix} \left(\begin{bmatrix} 1 & 0 & -1 \\ 0 & 1 & 2 \end{bmatrix} \begin{bmatrix} 1 & 2 \\ 4 & 5 \\ 7 & 8 \end{bmatrix} \right)^{-2} \begin{bmatrix} 1 & 0 & -1 \\ 0 & 1 & 2 \end{bmatrix}$$

$$A^\# = \begin{bmatrix} \frac{-23}{36} & \frac{-1}{6} & \frac{11}{36} \\ \frac{-1}{18} & 0 & \frac{1}{18} \\ \frac{19}{36} & \frac{1}{6} & \frac{-7}{36} \end{bmatrix}.$$

Como podemos observar la inversa de Moore-Penrose y la grupo inversa de la matriz A son iguales, según el teorema (10) puesto que $N(A) = R(A^*) = 2$.

La aproximación de (24) se puede extender de matrices diagonales a todas las matrices cuadradas de índice 1. Para ello necesitamos el siguiente lema:

Lema 6 Sea J una matriz cuadrada en forma de Jordan. Entonces J es de rango Hermitiano, sí y sólo sí tiene índice 1.

Para las matrices cuadradas con índice 1 se tiene la siguiente forma explícita para la grupo inversa.

Teorema 11 *Sea A de índice 1 y*

$$A = PJP^{-1}$$

Donde P es no singular y J es una forma normal de Jordan de A . Entonces

$$A^\# = PJ^\dagger P^{-1}. \quad (32)$$

4.4. Propiedades de la grupo inversa

La grupo inversa tiene propiedades espectrales comparables a las de la inversa no singular, incluso cuando A es una matriz no diagonal. Sin embargo, en este caso $A^\#$ no es la única matriz que tiene tales propiedades, también las matrices no singulares cuentan con estas propiedades.

Se observa que si una matriz cuadrada A tiene índice 1, y sus 0-vectores son todos de grado 1, es decir, vectores nulos de A . Esto se deduce del hecho que (7) implica $N(A^2) = N(A)$.

Los dos lemas siguientes nos ayudan a establecer las propiedades espectrales del grupo de inversas.

Lema 7 *Sea x un λ -vector de A con $\lambda \neq 0$. Entonces $x \in R(A^\ell)$ donde ℓ es un entero positivo arbitrario.*

Para probar el lema anterior se tiene

$$(A - \lambda I)^p x = 0,$$

para cualquier entero positivo p . Extendiendo el elemento izquierdo por el teorema del binomio, trasponiendo el último término, y dividiendo por su coeficiente $(-\lambda)^{p-1} \neq 0$ tenemos

$$x = c_1 Ax + c_2 A^2 x + \dots + c_p A^p x, \quad (33)$$

donde

$$C_i = (-1)^{i-1} \lambda^{-1} \binom{p}{i}.$$

La multiplicación sucesiva de (33) por A da como resultado:

$$\begin{aligned}
Ax &= c_1A^2x + c_2A^3x + \dots + c_pA^{p+1}x, \\
A^2x &= c_1A^3x + c_2A^4x + \dots + c_pA^{p+2}x, \\
&\dots = \dots\dots\dots \\
A^{\ell-1}x &= c_1A^{\ell}x + c_2A^{\ell+1}x + \dots + c_pA^{p+\ell-1}x
\end{aligned} \tag{34}$$

Y la sustitución sucesiva de las ecuaciones de (34) en la parte derecha de la ecuación (33), eventualmente nos da

$$x = A^{\ell}q(A)x,$$

donde q es algún polinomio.

Lema 8 Sea A una matriz cuadrada y sea

$$XA^{\ell+1} = A^{\ell} \tag{35}$$

para algún entero positivo ℓ . Entonces cada λ -vector de A de grado p para $\lambda \neq 0$ es un vector $\lambda - 1$ de X de grado p .

Este lema se expresa con mayor generalidad, debido a que, se utiliza en la conexión con inversas generalizadas espectrales distintos a la grupo inversa.

El siguiente teorema muestra que para cada matriz A de índice 1, la grupo inversa es la única matriz en $A\{1\}$ o $A\{2\}$ que tiene propiedades espectrales comparables a las de la inversa de una matriz no singular.

Definición 2.4 X es una S -inversa de A (o A y X son S -inversos entre sí) si comparten la propiedad de que, para cada $\lambda \in \mathbb{C}$ y cada vector x , x es un λ -vector de A de grado p , sí y sólo sí este es un λ^{\dagger} -vector de X de grado p .

Teorema 12 Sea $A \in \mathbb{C}_{n \times n}$ de índice 1. Entonces $A^{\#}$ es la única S -inversa de A en $A\{1\} \cup A\{2\}$. Si A es una matriz diagonal, $A^{\#}$ es la única S -inversa de A .

5. Inversa de Drazin

La grupo inversa no existe para todas las matrices cuadradas, sólo para matrices con índice 1. Cada matriz tiene una única $\{1^k, 2, 5\}$ -inversa, donde k es el índice. Esta inversa es llamada inversa de Drazin, fue estudiada por Drazin [16].

Si $A \in \mathbb{C}^{n \times n}$ tiene índice k . Entonces $R(A^k)$ y $N(A^k)$ son subespacios complementarios, y la restricción $A_{[R(A^k)]}$ de A a $R(A^k)$ es invertible. Si $X \in \mathbb{C}^{n \times n}$ está definida por:

$$Xu = \begin{cases} A_{[R(A^k)]}^{-1}u & \text{si } u \in R(A^k) \\ 0 & \text{si } u \in N(A^k). \end{cases} \quad (36)$$

Entonces las relaciones entre $AXu = XAu$ y $XAXu = Xu$ se cumplen para $u \in R(A^k)$ y para $u \in N(A^k)$, y por ende para todo \mathbb{C}^n . La matriz X es hasta ahora $\{2, 5\}$ -inversa de A .

La definición (36) dice que AX es la identidad en $R(A^k)$, es decir, $AXA^kx = A^kx$ para todo $x \in \mathbb{C}^n$, permitiendo cero en ambos lados (si $x \in N(A^k)$). Por lo tanto X es también una $\{1^k\}$ -inversa de A . La matriz X es así una $\{1^k, 2, 5\}$ -inversa de A . En el teorema (13) se muestra que esta inversa es única, X es llamada la inversa de Drazin de A , y es denotada por A^D .

El lema siguiente es necesario para probar la existencia y unicidad de la inversa de Drazin.

Lema 9 Si Y es una $\{1^\ell, 5\}$ -inversa de la matriz cuadrada A , entonces

$$X = A^\ell Y^{\ell+1}$$

es una $\{1^\ell, 2, 5\}$ -inversa

Teorema 13 Sea $A \in \mathbb{C}^{n \times n}$ con índice k . Entonces A tiene una única $\{1^k, 2, 5\}$ -inversa, la cual se puede expresar como un polinomio en A , y es también la única $\{1^\ell, 2, 5\}$ -inversa para cada $\ell \geq k$.

La grupo inversa es el caso particular de la inversa de Drazin para matrices de índice 1. La inversa de Drazin tiene una representación simple en términos de la forma de Jordan:

Teorema 14 Si $A \in \mathbb{C}^{n \times n}$ tiene la forma de Jordan

$$A = XJX^{-1} = X \begin{bmatrix} J_1 & 0 \\ 0 & J_0 \end{bmatrix} X^{-1}, \quad (37)$$

donde J_0 y J_1 son las partes de J correspondiente a los valores propios cero distintos de cero respectivamente. Entonces

$$A^D = X \begin{bmatrix} J_1^{-1} & 0 \\ 0 & 0 \end{bmatrix} X^{-1} \quad (38)$$

Ejemplo 2.5 La matriz

$$A = \begin{bmatrix} 0 & -2 & 0 & -5 & 2 \\ 0 & -1 & 0 & -2 & 0 \\ 0 & 0 & 0 & -2 & 0 \\ 0 & 1 & 0 & 2 & 0 \\ -2 & -2 & 1 & -8 & -4 \end{bmatrix}$$

tiene la forma de Jordan

$$A = X \begin{bmatrix} J_1(1) & & \\ & J_2(2) & \\ & & J_2(0) \end{bmatrix} X^{-1} = X \begin{bmatrix} 1 & & & & \\ & 2 & 1 & & \\ & 0 & 2 & & \\ & & & 0 & 1 \\ & & & 0 & 0 \end{bmatrix} X^{-1}.$$

Las propiedades espectrales de la inversa de Drazin son las siguientes:

- $A^D A A^D = A^D$
- $A A^D = A^D A$
- $A^k A^D A = A^k$, $k = \text{Ind}(A)$
- si $\text{Ind}(A) \leq 1$, entonces: $A A^D A = A$
- $((A^D)^D)^D = A^D$

5.1. Cálculo de la inversa de Drazin

Para hacer el cálculo de la inversa de Drazin se tomará el algoritmo que se encuentra en [15].

Algoritmo 2 Cálculo de A^D donde $A \in \mathbb{C}^{n \times n}$ y $\text{Ind}(A) = k$

- I. Sea p un entero tal que $p \geq k$ (p puede ser igual a n si no se puede determinar un valor menor). Si $A^p = 0$, entonces $A^D = 0$. Por tanto se asume $A^p \neq 0$.
- II. Reducir las filas de A^p a su forma escalonada de Hermite, H_{A^p} .
- III. En base a los elementos diferentes de cero o no nulos de la diagonal de H_{A^p} se seleccionan las columnas de A^p y las llamamos v_1, v_2, \dots, v_r (esto es una base para $R(A^k)$).
- IV. Se forma la matriz $I - H_{A^p}$, y seleccionamos las columnas distintas de cero. Las llamamos $v_{r+1}, v_{r+2}, \dots, v_n$. (Esto es una base para $N(A^k)$).

V. Construimos la matriz no singular $P = [v_1 | v_r | v_{r+1} | \dots | v_n]$

VI. Se calcula P^{-1}

VII. Formamos el producto $P^{-1}AP$. Esta matriz tendrá la forma:

$$P^{-1}AP = \begin{bmatrix} C & 0 \\ 0 & N \end{bmatrix}$$

donde C es no singular y N es nilpotente.

VIII. Calculamos C^{-1} .

IX. Se calcula A^D formando el producto $A^D = P \begin{bmatrix} C^{-1} & 0 \\ 0 & 0 \end{bmatrix} P^{-1}$

Ejemplo 2.6 Siguiendo el algoritmo anterior, resolvemos la inversa de Drazin de la matriz:

$$A = \begin{bmatrix} 2 & 0 & 0 \\ -1 & 1 & 1 \\ -1 & -1 & -1 \end{bmatrix}$$

Encontrar A^D

I) Debido a que no sabemos el $\text{Ind}(A)$ podemos asignar el valor a p con respecto al tamaño de nuestra matriz, en este caso el tamaño de A es de 3×3 , por tanto $p = 3$

$$A^p = A^3 = \begin{bmatrix} 8 & 0 & 0 \\ -8 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}.$$

II) Para obtener la forma Hermite de A^p (H_{A^3}), tenemos que hacer reducción mediante Gauss Jordan a A^3

$$H_{A^3} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}.$$

III) Confirmamos que $A^p \neq 0$, además de observar que la primera fila de H_{A^3} es independiente:

$$\begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}$$

Por lo tanto de A^p , se obtiene: $v_1 = \begin{bmatrix} 8 \\ -8 \\ 0 \end{bmatrix}$ que es base para $R(A^k)$.

Si se encuentran más filas independientes, se crearán los vectores correspondientes tomados de la matriz A^p .

IV) Ahora hacemos la resta de $I - H_A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$, de esta resta se obtiene:

$$v_2 = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}, v_3 = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$$

Estos vectores forman una base para $N(A^k)$

V) Con estos vectores formamos $P = [v_1 | v_r | v_{r+1} | \dots | v_n]$

$$P = \begin{bmatrix} 8 & 0 & 0 \\ -8 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix},$$

VI) Obtenemos $P^{-1} = \frac{1}{8} \begin{bmatrix} 1 & 0 & 0 \\ 8 & 8 & 0 \\ 0 & 0 & 8 \end{bmatrix}$, y

VII) Calculamos $P^{-1}AP = \begin{bmatrix} 2 & 0 & 0 \\ 0 & 1 & 1 \\ 0 & -1 & -1 \end{bmatrix}$

VIII) En esta matriz podemos identificar C y N , donde $C = [2]$ y $N = \begin{bmatrix} 1 & 1 \\ -1 & -1 \end{bmatrix}$, ahora se debe determinar la inversa de C y construir una matriz con las mismas dimensiones que la A .

$C^{-1} = \left[\frac{1}{2}\right]$, por lo tanto se obtiene:

IX) $A^D = P \begin{bmatrix} \frac{1}{2} & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} P^{-1} = \frac{1}{2} \begin{bmatrix} 1 & 0 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$.

5.2. A^D como un polinomio en A

Si A es una matriz no singular, entonces es fácil mostrar que A^{-1} puede ser expresado como un polinomio de A . Esta propiedad no se encuentra en las (i, j, k) -inversas mostradas en la Figura 3.4. En particular si A es cuadrada, entonces puede que no exista un polinomio $p(x)$ tal que $A^\dagger = p(A)$. Sin embargo, la inversa de Drazin puede ser expresada como un polinomio en A . Como se puede observar en el siguiente teorema.

Teorema 15 Si $A \in \mathbb{C}^{n \times n}$, entonces existe un polinomio $p(x)$ tal que $A^D = p(A)$.

Teorema 16 [25] Sea $A \in \mathbb{C}^{n \times n}$. Supongamos que $\{\lambda_0, \lambda_1, \lambda_2, \dots, \lambda_t\}$ son los distintos valores propios de A y $\lambda_0 = 0$. Sea m_i la notación para la multiplicidad algebraica de λ_i y sea $m = n - m_0 = m_1 + m_2 + \dots + m_t$. Sea $p(x)$ el polinomio de grado $n - 1$ tal que $p(x) = x^{m_0}(\alpha_0 + \alpha_1 x + \dots + \alpha_{m-1} x^{m-1})$ cuyos coeficientes son las únicas soluciones del siguiente sistema lineal de ecuaciones $m \times m$. ($(\cdot)^i$ denota la i -ésima derivada con respecto a x .)

$$\begin{aligned} \frac{1}{\lambda_i} &= p(\lambda_i), \\ -\frac{1}{\lambda_i^2} &= p'(\lambda_i), \\ &\vdots \\ \frac{(-1)^{m_i-1}(m_i-1)!}{\lambda_i^{m_i}} &= p^{(m_i-1)}(\lambda_i) \end{aligned} \quad \text{para } i = 1, 2, \dots, t,$$

Entonces $p(A) = A^D$.

Con base en estos teoremas se puede determinar un algoritmo para calcular la inversa de Drazin como polinomio de A . Mediante el teorema 16 también se puede obtener la inversa de Drazin. Este teorema se puede aplicar sólo si m_0 es tan grande como n . A continuación se ejemplifica el teorema anterior.

Ejemplo 2.7 Sea

$$A = \begin{bmatrix} 10 & -8 & 6 & -3 \\ 12 & -10 & 8 & -4 \\ 1 & -1 & 1 & 0 \\ -2 & 2 & -2 & 2 \end{bmatrix}$$

se usará el teorema 16 para el cálculo de A^D . El primer paso (y generalmente el más difícil) es calcular los valores propios de A . En este caso los valores propios $\sigma(A) = \{0, 0, 1, 1\}$. Por lo tanto $m_0 = 2$ y $m_1 = 2$ por lo tanto el teorema 16 implica que A^D puede ser expresada como $A^2(\alpha_0 I + \alpha_1 A)$ porque $p(x) = x^2(\alpha_0 + \alpha_1 x)$. Ahora α_0 y α_1 son soluciones para el sistema:

$$\begin{aligned} 1 &= p(1) = \alpha_0 + \alpha_1 \\ -1 &= p'(1)2\alpha_0 + 3\alpha_1 \end{aligned}$$

Por lo tanto $\alpha_0 = 4$ y $\alpha_1 = -3$, y

$$A^D = A^2(4I - 3A) = \begin{bmatrix} 3 & -1 & 2 & 2 \\ 2 & 1 & 3 & 3 \\ -1 & 0 & -1 & -1 \\ -1 & 0 & -1 & -1 \end{bmatrix}.$$

A continuación se muestra un algoritmo para calcular esta inversa, este algoritmo se conoce también como el algoritmo de Greville [19], este algoritmo no hace uso del teorema anterior, el cálculo de la inversa de Drazin está basado en la traza de la matriz y no en los valores propios.

Algoritmo 3 Calcular A^D para $A \in \mathbb{C}^{n \times n}$

- I) Definir $S_0 = I$ y calcular recursivamente $S_j = AS_{j-1} + \beta_{n-j}I$, $\beta_{n-j} = -\frac{1}{j}Tr(AS_{j-1})$, hasta que alguna $S_t = 0$, pero $S_{t-1} \neq 0$, donde $TR(.)$ es la traza de la matriz.
- II) Sea u un número tal que $\beta_{n-u} \neq 0$ y $\beta_{n-u-1} = \beta_{n-u-2} = \dots = \beta_{n-t-1} = 0$. (Notar que $n - u = m_0$, es la multiplicidad algebraica del valor propio 0.)
- III) Sea $l = n - u$ y calcular $S_{n-m_0-1}^{l+1} = S_{u-1}^{l+1}$.
- IV) Calcular A^D como $A^D = -\frac{1}{\beta_{n-u}^{l+1}} A^l S_{u-1}^{l+1}$.

La traza de una matriz es la suma de los elementos de su diagonal y es denotada como $TR(.)$.

Con base en el algoritmo anterior calcular la inversa de Drazin de

Ejemplo 2.8

$$A = \begin{bmatrix} 10 & -8 & 6 & -3 \\ 12 & -10 & 8 & -4 \\ 1 & -1 & 1 & 0 \\ -2 & 2 & -2 & 2 \end{bmatrix}$$

I) Mediante cálculos sucesivos se obtiene:

$$S_0 = I \quad \beta_3 = -Tr(AS_0) = -Tr(A) = -3$$

$$S_1 = AS_0 - 3I = \begin{bmatrix} 7 & -8 & 6 & -3 \\ 12 & -13 & 8 & -4 \\ 1 & -1 & -2 & 0 \\ -2 & 2 & -2 & -1 \end{bmatrix}$$

$$AS_1 = \begin{bmatrix} -14 & 12 & -10 & 5 \\ -20 & 18 & -16 & 8 \\ -4 & 4 & -4 & 1 \\ 4 & -4 & 4 & -4 \end{bmatrix}, \quad \beta_2 = -\frac{1}{2}Tr(AS_1) = 2$$

$$S_2 = AS_1 + 2I = \begin{bmatrix} -12 & 12 & -10 & 5 \\ -20 & 20 & -16 & 8 \\ -4 & 4 & -2 & 1 \\ 4 & -4 & 4 & -2 \end{bmatrix},$$

$$AS_2 = \begin{bmatrix} 4 & -4 & -4 & -2 \\ 8 & -8 & 8 & -4 \\ 4 & -4 & -4 & -2 \\ 0 & 0 & 0 & 0 \end{bmatrix}, \quad \beta_1 = -\frac{1}{3}Tr(AS_2) = 0$$

$$S_3 = AS_2$$

$$AS_3 = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}, \quad \beta_0 = -\frac{1}{4}Tr(AS) = 0,$$

y

$$S_4 = AS_3 = 0$$

Por lo tanto, en este ejemplo la multiplicidad algebraica del valor propio 0 es $m_0 = 2$.

II) Establecer $u = 2$.

III) Establecer $l = 2$.

IV) Calcular $A^D = -\frac{1}{\beta_2^3} A^2 S_1^3 = -\frac{1}{8} A^2 S_1^3$ como sigue. Sabemos que $S_2 = AS_1 + \beta_2 I$, y tenemos que $AS_2 S_1^2 = A^2 S_1^3 + \beta_2 AS_1^2$. Esto se puede escribir como $A^2 S_1^3 = [(AS_2)S_1 - \beta_2(AS_1)] S_1$. AS_2 y AS_1 ya están calculadas, sólo dos multiplicaciones de matrices son necesarias. Esto es más eficiente que formar el producto $A^2 S_1^3$ directamente. En general siempre se puede hacer algo así cuando se usa este algoritmo. Ahora

$$(AS_2)S_1 = \begin{bmatrix} -12 & 12 & -12 & 6 \\ -24 & 24 & -24 & 12 \\ -12 & 12 & -12 & 6 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

$$[(AS_2)S_1 - 2(AS_1)] = \begin{bmatrix} 16 & -12 & 8 & -4 \\ 16 & -12 & 8 & -4 \\ -4 & 4 & -4 & 4 \\ -8 & 8 & -8 & 8 \end{bmatrix}$$

y

$$A^2 S_1^3 = [(AS_2)S_1 - 2(AS_1)] S_1 = \begin{bmatrix} -16 & 12 & -8 & 4 \\ -16 & 12 & -8 & 4 \\ 8 & -8 & 8 & -8 \\ 16 & -16 & 16 & -16 \end{bmatrix}$$

Por lo tanto,

$$A^D = -\frac{1}{8} A^2 S_1^3 = \begin{bmatrix} 2 & -\frac{3}{2} & 1 & -\frac{1}{2} \\ 2 & -\frac{3}{2} & 1 & -\frac{1}{2} \\ -1 & 1 & -1 & 1 \\ -2 & 2 & -2 & 2 \end{bmatrix}.$$

Capítulo 3

Aspectos computacionales de las inversas generalizadas

1. Introducción

Existen tres situaciones principales en las que se requiere obtener numéricamente una inversa generalizada de una matriz:

- (I) En el caso en que cualquier $\{1\}$ -inversa, sea suficiente;
- (II) Los casos en los que cualquier $\{1, 3\}$ -inversa o cualquier $\{1, 4\}$ -inversa sea suficiente;
- (III) El caso en el que se requiere una $\{2\}$ -inversa con un rango(imagen) específico y espacio nulo.

La inversa deseada en el caso (III) es, en la mayoría de los casos, la inversa de Moore-Penrose, que es la única $\{2\}$ -inversa de la matriz dada A que tiene el mismo rango y espacio nulo que A^* . La inversa de Drazin también se puede incluir en este caso, siendo la única $\{2\}$ -inversa que tiene el mismo espacio nulo y rango que A^ℓ , donde ℓ es cualquier número entero no menor al índice de A . Cuando $\ell = 1$ se tiene la grupo inversa.

Las inversas generalizadas están estrechamente asociadas con ecuaciones lineales, ortonormalizaciones, soluciones de mínimos cuadrados, valores singulares y diversas factorizaciones matriciales. En particular la factorización QR y la descomposición de valores singulares(SVD) figuran de forma destacada en el cómputo de la inversa de Moore-Penrose.

La factorización QR está implícita en el algoritmo principal para SVD, de Golub y Kahan, donde la matriz en cuestión se transforma primero, por rotaciones, en una forma bidiago-

nal superior. Estos temas han sido ampliamente estudiados por Björck, Golub y Van Loan, Lawson y Hanson, sólo por nombrar algunos. En este capítulo se enumeran algunos métodos computacionales para la inversa generalizada.

2. Cálculo de las $\{1\}$ y $\{1, 2\}$ -inversas sin restricciones

Sea A una matriz para la cual se desea una $\{1\}$ -inversa, cuando cualquier $\{1\}$ -inversa sea suficiente. Si sucede que A es de tal estructura o ha crecido de tal manera, que se conoce una submatriz no singular de orden máximo, podemos escribir

$$PAQ = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} \quad (1)$$

donde A_{11} es no singular, P y Q son matrices de permutación para llevar la submatriz no singular a la posición superior izquierda. (Si A es de rango completo (columna o fila), algunas de las submatrices en (1) estarán ausentes). Dado que el rango de A es el orden de A_{11} , esto implica que

$$A_{22} = A_{21}A_{11}^{-1}A_{12} \quad (2)$$

y una $\{1, 2\}$ -inversa de A es

$$A^{(1,2)} = Q \begin{bmatrix} A_{11}^{-1} & O \\ O & O \end{bmatrix} P. \quad (3)$$

En el caso más usual en el que no se conoce una submatriz no singular de orden máximo y así mismo, se desconoce el rango de A , quizás el método más simple sea usando la eliminación gaussiana para llevar A a la forma normal de Hermite,

$$EAP = \begin{bmatrix} I_r & K \\ O & O \end{bmatrix} \quad (4)$$

(Con modificaciones en el caso de que A sea de rango completo), donde E es no singular y P es una matriz de permutación, entonces:

$$A^{(1)} = P \begin{bmatrix} I_r & O \\ O & L \end{bmatrix} E \quad (5)$$

es una $\{1\}$ -inversa de A para una L arbitraria. Por supuesto, la opción más simple es $L = O$ que da la $\{1, 2\}$ -inversa

$$A^{(1,2)} = P \begin{bmatrix} I_r & O \\ O & O \end{bmatrix} E. \quad (6)$$

Por otro lado, cuando A es cuadrado, una $\{1\}$ -inversa a veces puede obtenerse tomando L como no singular en (5). La elección más simple para L es una matriz unitaria, que da

$$A^{(1)} = PE.$$

En las aplicaciones que implican ecuaciones lineales, a menudo ocurre que una solución particular no es suficiente. Los resultados anteriores pueden adaptarse fácilmente para obtener dicha solución, siempre que exista.

Teorema 17 Dado $A \in \mathbb{C}_r^{m \times n}$ con $r < m$, y $b \in \mathbb{C}^m$, consideremos la ecuación lineal

$$Ax = b \quad (7)$$

Sea E una matriz no singular, P una matriz de permutación, tal que

$$EAP = \begin{bmatrix} I_r & K \\ O & O \end{bmatrix} \quad (8)$$

y sea \widehat{b} y z definidos por

$$E[A \quad b] \begin{bmatrix} P & O \\ O^T & 1 \end{bmatrix} = \begin{bmatrix} I_r & K & \widehat{b} \\ O & O & z \end{bmatrix}. \quad (9)$$

Entonces ((7)) tiene una solución si y solo si $z = 0$, en tal caso una solución particular es

$$x = A^{(1,2)}b = P\widehat{b}. \quad (10)$$

3. Cálculo de la $\{1, 3\}$ -inversa sin restricciones

Sea $A \in \mathbb{C}_r^{m \times n}$ y

$$A = FG \quad (11)$$

una factorización de rango completo. Entonces:

$$X = G^{(1)}F^\dagger, \quad (12)$$

donde $G^{(1)}$, es una $\{1, 2, 3\}$ -inversa de A .

$$A = \tilde{Q}\tilde{R} \quad (13)$$

Donde las columnas de $\tilde{Q} = [q_1 \dots q_r]$ son ortonormales, y \tilde{R} es triangular superior.

De (12) obtenemos:

$$A^{(1,2,3)} = \tilde{R}^{(1)}\tilde{Q}^\dagger. \quad (14)$$

Esto es útil ya que la inversa de Moore-Penrose de \tilde{Q} es simplemente:

$$\tilde{Q}^\dagger = \begin{bmatrix} q_1^* \\ q_2^* \\ \dots \\ q_r^* \end{bmatrix} \quad (15)$$

y una $\{1\}$ -inversa de \tilde{R} también se obtiene fácilmente: \tilde{R} es de rango completo por filas y se puede construir (permutando columnas según sea necesario) como:

$$\tilde{R}P = [T \quad K]$$

donde T es no singular y triangular superior, y P es una matriz de permutación.

Así

$$\tilde{R}^{(1)} = P \begin{bmatrix} T^{-1} \\ O \end{bmatrix}, \quad (16)$$

donde T^{-1} es una matriz triangular superior y se obtiene de T por sustitución hacia atrás.

Capítulo 4

Desarrollo

En este capítulo se presenta el proceso que se llevó a cabo para el desarrollo del sistema de aproximaciones para las inversas de Moore-Penrose y de Drazin escogiendo e implementando los algoritmos adecuados para el cálculo de cada una de ellas.

1. Alcances

En el desarrollo de cualquier sistema se deben de especificar los alcances y limitaciones, en este caso los alcances se establecieron teniendo en cuenta el tamaño de las matrices con las que se trabajará, el entorno de programación que se utilizará, además de diseñar el sistema de forma amigable para el usuario. A continuación se muestra un listado de estas limitaciones:

- El sistema se desarrollará en Matlab 2015a.
- Se debe introducir la matriz al sistema mediante un archivo .txt, y se escribirá el directorio de donde se encuentra este archivo.
- El sistema sólo entregará aproximaciones de inversas de matrices con números reales.

2. Limitaciones

El sistema que se desarrolla cuenta con las siguientes limitaciones:

- El sistema no permite matrices de números complejos.

- Las matrices de tamaño mayor a 15×15 quedarán excluidas para el cálculo de alguna inversa.

3. Aproximación de la inversa de Moore-Penrose

La implementación del algoritmo para la aproximación de la inversa de Moore-Penrose está basada en la información mostrada en la sección 3.6 del capítulo 2, donde se muestra de manera explícita la siguiente fórmula para el cálculo de la inversa de Moore-Penrose:

$$A^\dagger = G^*(F^*AG^*)^{-1}F^*.$$

Las matrices G y F son una factorización de A , y para obtenerlas es necesario encontrar la forma de Hermite de la matriz A , esta matriz se puede generar mediante la reducción de Gauss-Jordan, respetando el orden de las filas con respecto al pivote de cada columna.

Matlab es una herramienta que nos ofrece diversas librerías y funciones para uso matemático, entre las funciones que ofrece y que podemos ocupar se encuentra la reducción mediante Gauss-Jordan, pero para las operaciones que realizaremos es necesario crear una función independiente, debido a que, la función que nos proporciona Matlab hace la reducción moviendo las filas, este movimiento es impredecible, y para poder hacer la factorización de A es necesario que las filas se muevan pero con respecto a la columna pivote.

La función que se desarrolló para hacer la forma reducida de Hermite, consta de dos partes, la primera es la reducción de la matriz mediante Gauss-Jordan, ésta reducción se hace sin alterar el orden de las filas y en la segunda parte se realiza el movimiento de las filas con respecto al pivote de cada columna.

```

1 function A=Gauss_Jordan(A)
2 % Funcion para realizar la reduccion de una matriz mediante Gauss
  Jordan,
3 % obtiene como entrada una matriz A y como salida obtenemos la matriz
4 % reducida y en su forma de Hermite
5 %
6 %
7 % Autor: Fabiola Armas Ramirez
8 %
9 [filas columnas]=size(A);
10 controlFila=1;
11 pivote=0;
12 for k=1:columnas
13 if(controlFila<=filas && pivote~=columnas )
14 if(a(controlFila ,k)~=0)
15     pivote=controlFila;
16     controlFila=controlFila+1;
17 else
18     %utilizamos la funcion de regresarPivote() que nos retorna la
19     %posicion del pivote
20     pivote=regresarPivote(A,k,controlFila);
21 end
22 if(pivote~-=-1 && pivote~=0)
23     A(pivote ,:)=A(pivote ,:)/A(pivote ,k);
24     for j=pivote-1:filas
25         A(j ,:)=A(j ,:)-A(pivote ,:)*A(j ,k);
26         j=j+1;
27     end
28     k=k-1;
29 end
30 end
31 end
32
33 for g=1:filas
34     indice=0;
35     for pivote=filas:-1:1
36         if(g<=columnas && A(pivote ,g)==1 && pivote>=g)
37             indice=pivote;
38             break
39         end
40         indice=-1;
41     end
42     if indice~-=-1 %reduccion hacia arriba, tenemos pivote y columna
43         for j=pivote-1:-1:1
44             A(j ,:)=A(j ,:)-A(pivote ,:)*A(j ,g);
45         end
46     end
47 end
48 % convertimos la matriz en su forma de Hermite
49 A=convertirHermite(A);
50 end

```

Listing 4.1: Función de Gauss-Jordan

La función mostrada en el Listing 4.1 genera la reducción de una matriz mediante Gauss-Jordan, ésta función realiza primero la reducción triangular inferior y posteriormente la reducción triangular superior, para localizar la posición del pivote de cada columna y hacer la reducción correcta es necesario la creación de otra función que nos devuelva este índice.

```

1 function ind= regresarPivote(matriz ,columnaP ,filaP)
2 % Funcion que tiene como parametros de entrada la matriz, la columna y
3 % fila de donde va a empezar a buscar el pivote.
4 % y como salida se obtiene la posicion del pivote
5 %
6 % Autor:Fabiola Armas Ramirez
7 ind=0;
8 |filas columnas|=size(matriz);
9 for fi=filaP:filas
10     if(matriz(fi ,columnaP)~=0)
11         ind=fi;
12         break;
13     end
14
15 end
16 end

```

Listing 4.2: Función para obtener el pivote

En el Listing anterior se muestra la función que nos devuelve la posición del pivote, los parámetros que tiene como entrada son la matriz, la columna de la que se deberá encontrar el pivote y el número de fila en el que se empieza a buscar el pivote, el parámetro de salida es el número de la fila en donde se encuentra el pivote.

Para determinar la aproximación de la inversa de Moore-Penrose de una matriz A , es necesario obtener la forma de Hermite de A , por lo que al terminar de hacer la reducción mediante Gauss-Jordan procedemos a convertir la matriz resultante en su forma de Hermite.

```

1 function [H_A]= convertirHermite(A)
2 %
3 % Esta funcion sirve para generar la forma reducida de Hermite
4 % de una matriz que paso por la funcion de Gauss-Jordan
5 %
6 % Autor: Fabiola Armas Ramirez
7
8 |filas columnas|=size(A);
9 for c=1:columnas
10     if(A(c,c)~=1)
11         for f=c+1:filas
12             if(A(f,c)==1)
13                 vector=A(f,:);
14                 vector2=A(c,:);
15                 A(f,:)=vector2;
16                 A(c,:)=vector;
17             end
18         end
19     end
20 end
21 H_A=A;
22 end

```

Listing 4.3: Código forma reducida de Hermite

Al aplicar el conjunto de funciones mostradas en los Listing 4.1, 4.2 y 4.3 obtenemos la forma de Hermite de una matriz A , con este resultado podemos seguir el proceso de la aproximación a la inversa de Moore-Penrose y generar las matrices F y G para la factorización de A .

```

1
2 function [G]= crearMatrizRangoFila(A,filas)
3 % Funcion que crea una matriz en obteniendo solo algunas filas de la
4 % matriz original. en este caso la matriz original es nuestro parametro
5 % de entrada A, y las filas que se tomara de esta matriz vienen en el
6 % parametro de entrada 'filas' que contiene las filas que se van a
7 % extraer de A
8 %
9 % Autor : Fabiola Armas Ramirez
10
11 G=[];
12 for re=1:length(filas)
13     G=[G; A(filas(re),:)] ;
14 end
15 end
16
17 function [F]= crearMatrizRangocolumna(A,columnas)
18 % Funcion que crea una matriz en obteniendo solo algunas columnas de la
19 % matriz original. en este caso la matriz original es nuestro parametro
20 % de entrada A, y las columnas que se tomara de esta matriz vienen en el
21 % parametro de entrada 'columnas' que contiene las filas que se van a
22 % extraer de A
23 %
24 % Autor : Fabiola Armas Ramirez
25 F=[];
26 for re=1:length(columnas)
27     F =[F A(:,columnas(re))] ;
28 end
29 end

```

Listing 4.4: Funciones para construir las matrices F Y G

Como recordaremos las matrices F y G se construyen con base en la matriz A reducida en su forma de Hermite (H_A), la matriz F se construye con las filas que son diferentes de cero de H_A . Mientras que G corresponde a las columnas linealmente independientes de A que coinciden con las columnas pivotes de la matriz H_A .

Las funciones mostradas en el Listing 4.4 nos ayudan a obtener las matrices F y G , con estas dos matrices construidas podemos implementar la fórmula para determinar A^\dagger , para esto debemos asegurarnos que la matriz resultante de la multiplicación F^*AG^* tenga determinante diferente de cero, si el determinante es diferente de cero se calcula la inversa de Moore-Penrose mediante la fórmula explícita A^\dagger , de otra manera no se podrá determinar la inversa, podemos verificar que el resultado de la multiplicación (F^*AG^*) tiene inversa con el uso de la función `inv()` [7] que viene incorporada en Matlab.

Dentro de Matlab encontramos la función `vpa()` [11] que utiliza aritmética de punto flotante y tiene una precisión de hasta 32 dígitos significativos. Generalmente `vpa()` evalúa las operaciones con precisión doble, y convierte expresiones numéricas en expresiones simbólicas. Esta función se utilizó al momento de hacer las operaciones para determinar la aproximación de la inversa, de esta manera obtenemos una mayor precisión en los resultados.

4. Aproximación de la inversa de Drazin

El algoritmo que se implementó para determinar una aproximación a la inversa de Drazin fue el algoritmo 2 que se encuentra en la sección 5.1 del capítulo 2:

Cálculo de A^D donde $A \in \mathbb{C}^{n \times n}$ y $Ind(A) = k$

- I. Sea p un entero tal que $p \geq k$ (p puede ser igual a n si no se puede determinar un valor menor). Si $A^D = 0$, entonces $A^D = 0$. Por tanto se asume $A^p \neq 0$. Entonces para saber el valor de p , se debe saber el tamaño de la matriz, en caso de que el tamaño de la matriz fuera de 2×2 el valor de p sería 2.
- II. Reducir las filas de A^p a su forma escalonada de Hermite, H_{A^p} . En este paso hacemos uso de las tres funciones mostradas en el Listing 4.1, 4.2 y 4.3.
- III. En base a los elementos no-ceros de la diagonal de H_{A^p} se seleccionan las columnas de A^p y las llamamos v_1, v_2, \dots, v_r (Esto es básicamente $R(A^p)$). Por lo tanto se seleccionan las columnas base de la matriz reducida Hermite del paso anterior y tomamos las columnas de A^p que correspondan a estos vectores base.

```
1 function [vectoresBase]= vectoresBase(matriz)
2 % Funcion que determina cuales columnas de una matriz son vectores base
3 % En vectoresBase guarda el numero de las columnas que son base
4
5 % Autor: Fabiola Armas Ramirez.
6 %
7 vectoresBase=[];
8 [filas ,columnas]=size(matriz);
9 for c=1:columnas
10     unos=0;
11     ceros=0;
12     for f=1:filas
13         if(matriz(f,c)==1)
14             unos=unos+1;
15         elseif (matriz(f,c)==0)
16             ceros=ceros+1;
17         end
18     end
19     if(unos==1 && (unos+ceros)==filas)
20         vectoresBase=[vectoresBase c];
21     end
22 end
23 end
```

Listing 4.5: Función para encontrar los vectores base en una matriz

El código mostrado en el Listing anterior, nos sirve para identificar las columnas que son vectores base de alguna matriz, en este caso la función recibe como parámetro la matriz H_{A^p} , éstos vectores base son los primeros con los que se formará la matriz P .

- IV. Se forma la matriz $I - H_{A^p}$, y seleccionamos las columnas distintas de cero. Las llamamos $v_{r+1}, v_{r+2}, \dots, v_n$. (Esto es básicamente $N(A^p)$).

```

1  for c=1:filas
2      for f=1:columnas
3          if I_H(f,c)~=0
4              % Insertamos en P los vectores diferentes de cero de la
5              % matriz I_H
6              P(:,indice+1)=I_H(:,c);
7              indice=indice+1;
8              break;
9          end
10     end
11 end

```

Listing 4.6: Código para extraer columnas diferentes de cero

El código anterior es el que nos ayudará a extraer las columnas diferentes de cero de la matriz $I - H_{A^p}$, estas columnas son los vectores que nos hacen falta para terminar de formar la matriz P , por lo que hace de manera directa la extracción e inserción de estas columnas de la matriz resultante de la resta a la matriz P .

- V. Construimos la matriz no singular $P = [v_1 | v_r | v_{r+1} | \dots | v_n]$. El código mostrado en el Listing 4.6, hace las últimas inserciones a la matriz P , para su construcción total.
- VI. Se calcula P^{-1} .

Para determinar la inversa de P utilizamos la función de Matlab `inv()`[7].

- VII. Formamos el producto $P^{-1}AP$. Esta matriz tendrá la forma:

$$P^{-1}AP = \begin{bmatrix} C & 0 \\ 0 & N \end{bmatrix}$$

donde C es no singular y N es nilpotente.

```

1 function [C N]=determinarC_N(A)
2 % Funcion que determina las matrices C y N de una matriz A.
3 % Autor:Fabiola Armas Ramirez
4 %
5
6 N=0;
7 [filas ,columnas]=size(A);
8 C=zeros(1,1);
9 % Se inicializa la matriz C y se empieza su extraccion de la
10 matriz A
11 C(1,1)=A(1,1);
12 for f=2:filas
13     if A(f,f)~=0
14         bandera=0;
15         for g=1:(f-1)
16             if A(g,f)~=0 || A(f,g)~=0
17                 bandera=1;
18                 break;
19             end
20         end
21     elseif A(f,f)==0
22         for g=1:(f)
23             if A(g,f)~=0 || A(f,g)~=0
24                 bandera=1;
25                 break;
26             end
27         end
28     if bandera==1
29         for g=1:f
30             C(g,f)=A(g,f);
31             C(f,g)=A(f,g);
32         end
33         bandera=0;
34     else
35         break;
36     end
37 end
38 n=length(C);
39 l=length(A);
40 inicio=n+1;
41 N=zeros(1-n,1-n);
42 indice=1;
43 fff=1;
44 % Comienza codigo para extraer la matriz N
45 for c=inicio:l
46     for fi=inicio:l
47         N(indice, fff)=A(fi,c);
48         indice=indice+1;
49     end
50     fff=fff+1;
51     indice=1;
52 end
53 end

```

Listing 4.7: Código para la construcción de las matrices C y N

Las submatrices C y N no tienen tamaño fijo, estas varían de dimensiones con respecto a la matriz A de la que se quiere saber su inversa, por ende en el Listing 4.7 se muestra la función que nos devuelve dos matrices, en este caso C y N , donde C es no singular y N es nilpotente.

VIII. Calculamos C^{-1} . Para calcular la inversa de C , utilizamos la función `inv()`[7] de la librería de Matlab.

IX. Se calcula A^D formando el producto $A^D = P \begin{bmatrix} C^{-1} & 0 \\ 0 & 0 \end{bmatrix} P^{-1}$.

Con esta multiplicación obtenemos la inversa de Drazin y al igual que en la aproximación de la inversa de Moore-Penrose, en las multiplicaciones de matrices se ocupa la función `vpa()`[11].

5. Aproximación a A^D como polinomio en A

Algunas veces la inversa de Drazin puede ser expresada como un polinomio, en la sección 5.2 se muestra el teorema 16, éste teorema es el que se aplica para calcular la aproximación de una inversa de Drazin como polinomio. Para determinar el tamaño del polinomio que se formará a partir de una matriz A necesitamos conocer sus valores propios:

```
1 % eig extrae los valores propios de una matriz
2 V=eig(A);
3 % se obtiene solo la parte real de los valores propios
4 % y se extrae solo el numero entero del valor propio
5 parteReal=round(real(V));
```

Listing 4.8: Código para obtener valores propios

El código del Listing 4.8 muestra el uso de la función `eig()`[5] de la librería de Matlab, esta función devuelve los valores propios de una matriz con la parte real y la parte imaginaria, en este caso sólo necesitamos la parte real del valor propio por lo que utilizamos las funciones `real()`[9] y `round()` [10] para obtener la parte real y en número entero del valor propio respectivamente, como resultado tenemos un vector con todos los valores propios de la matriz A , con base en este vector se debe calcular la multiplicidad de cada valor propio.

La función que se expone en el Listing 4.9 primero ordena el arreglo donde se encuentran los valores propios, posteriormente hace una búsqueda de cada valor propio dentro de este arreglo, con este proceso determina cual es la multiplicidad de cada valor propio. Los resultados que se obtienen de esta función son dos arreglos, el primero contiene los valores propios sin repetición y en el segundo se incluye la multiplicidad de cada valor propio.

```

1 function [valores mult]=multiplicidad(valoresPropios)
2 % Funcion que encuentra la multiplicidad de los valores propios, los
3 % parametros de salida son valores que es un arreglo ordenado de los
4 % valores propios, y mult que es el arreglo donde se encuentra la
5 % multiplicidad de los valores propios
6 ordenado=sort(valoresPropios)
7 valores=[];
8 mult=[];
9 for r=1: length(valoresPropios)
10     if (isempty(valores)==1 || isempty(find(valores==ordenado(r)))-1)
11         ordenado(r);
12         valores=[valores ordenado(r)];
13         mult=[mult length(find(ordenado==valores(length(valores))))];
14     end
15 end
16 end

```

Listing 4.9: Código para obtener la multiplicidad de los valores propios

Con base en los resultados que se obtienen del Listing 4.9 podemos determinar el polinomio de la matriz A , para construir este polinomio se necesita el uso de la librería *syms* [3] que nos ayuda a crear variables simbólicas.

Siguiendo el Teorema 16 la variable simbólica que se debe utilizar es α , pero por practicidad se creó la variable simbólica c , a continuación se debe especificar cuantas c 's se requieren, por lo que la resta de $n - m_0$, donde m_0 es la multiplicidad del valor propio cero y n es el tamaño de la matriz nos entregará el número de c 's que se crearán.

La secuencia de variables simbólicas que se guardarán en un arreglo es c_0, c_1, \dots, c_j , donde j es el resultado de la resta anterior y con la función de Matlab *poly2sym()* [2] crearemos el polinomio de la matriz A , ésta función recibe como parámetro el arreglo de las variables simbólicas y nos devuelve el polinomio con base en este mismo arreglo.

Con el polinomio construido se procede a derivar y evaluar cada derivada, para estos cálculos empleamos la función *diff()* [4] que será la encargada de derivar el polinomio recibiendo como parámetros el polinomio y el número de derivadas que queremos obtener, éstas derivadas deben ser declaradas como funciones para que puedan ser evaluadas, en el Listing 4.10 se observa como se declara el polinomio como $p(x)$ [3] y sus derivadas como $k(x)$, para evaluar estas funciones se tiene que sustituir x con el valor que se quiere evaluar, por viabilidad se simplifica cada función evaluada con *simplify()* [1]. Después de simplificar cada función obtenemos como resultado un sistema de ecuaciones el cual debe ser resuelto para obtener el valor de cada variable simbólica c , y con base en estos resultados calcular la inversa de Drazin (ver Ejemplo 2.7).

```

1 % establecer las variables simbolicas
2 syms x c0 x2
3 c = [];
4 S = [];
5 b = [];
6 if (vectorPropio(1)==0)
7     valor=1;
8     m0=mult(1);
9     % se determinan el valor de m
10    m=filas -m0;
11    s=m-1;
12    % con base en el valor de m, se crean las demas variables que se van a
13    % ocupar en el polinomio
14    c=[c0 sym('c',[1 s])];
15    c=fliplr(c);
16    % crear una funcion con el polnomio que se hace con la funcion poly2sym
17    (c)
18    p(x)=[x^m0*(poly2sym(c))];
19    for g=1:m
20        numerador=((-1)^(g-1))*(factorial(g-1));
21        denominador=vectorPropio(2)^g;
22        b=[b; numerador/denominador];
23        k(x)=simplify(diff(p(x),g-1))
24        segunda=k(vectorPropio(2));
25        S=[S; coeffs(segunda)];
26    end
27 Resultado=S\b
end

```

Listing 4.10: Código para obtener el polinomio de A

Los códigos mostrados en cada uno de los Listings de este capítulo son los encargados de crear las aproximaciones a cada una de las inversas, en el sistema de aproximaciones a inversas se determina la inversa de Moore-Penrose mediante la fórmula explícita de A^\dagger , mientras que para determinar la inversa de Drazin se utilizan dos métodos, el primero de estos métodos es localizar el polinomio de la matriz A y si este no excede dos α 's se resuelve la inversa mediante el polinomio, en caso contrario se genera la aproximación mediante el índice de la matriz.

6. Interfaz gráfica

El diseño de la interfaz gráfica del sistema de aproximaciones de inversas se realizó en Matlab mediante su herramienta "GUIDE Quick Start"[8], herramienta que nos permite generar interfaces amigables para el usuario.

La interfaz es la que permitirá de una forma fácil para el usuario introducir la información necesaria para que el sistema pueda generar la inversa generalizada de su elección.



Figura 4.1: Interfaz gráfica del sistema de aproximaciones de inversas generalizadas

La figura 4.1 muestra la interfaz gráfica que se diseñó para el sistema de aproximaciones de inversas. En algunas zonas de la figura 4.1 se observan algunos números que nos permitirán explicar la funcionalidad de cada elemento de la interfaz.

- En (1) se coloca la ruta del archivo .txt donde se encuentra la matriz de la que queremos saber su inversa. Ejemplo: $C : users \ carpeta \ matrizA.txt$, de esta manera el sistema buscará en esta ruta el archivo, y en esa misma ruta guardará el resultado pero de la siguiente manera $C : users \ carpeta \ matrizA_InversaGeneralizada.txt$, este proceso lo hace sin importar de que tipo de inversa sea el resultado, el sistema creará un archivo .txt concatenando el nombre del archivo original con $_InversaGeneralizada$ en el mismo directorio.
- En las cajas de texto (2) y (3) se debe especificar el tamaño de la matriz, colocando el número de filas y número de columnas de la matriz.
- El botón (4) cargará al sistema la matriz con base en el directorio y el tamaño de la matriz especificados en los puntos (1), (2) y (3).

- El botón (5) se selecciona cuando la inversa generalizada que buscamos corresponde a la inversa de Moore-Penrose.
- El botón (6) lo seleccionamos cuando la inversa generalizada que buscamos corresponde a la inversa de Drazin.
- El área de texto (7) es donde se visualizará la aproximación de la inversa especificada, además en este punto se crea un archivo .txt donde se guarda el resultado.

Capítulo 5

Pruebas

1. Introducción

La etapa de pruebas es importante en cualquier desarrollo de sistema de información, sirve para lograr una buena calidad del producto, debido a que si encuentra anomalías busca junto con el desarrollador o desarrolladores una solución. En este trabajo nos ayudará a probar el funcionamiento del sistema y de asegurarse que realice bien las tareas para las que fue creado.

2. Inversa de Moore-Penrose

En el periodo de pruebas de la aproximación para la inversa de Moore-Penrose, se utilizaron las ecuaciones de Moore-Penrose descritas en la sección 2.2 del capítulo 2:

$$AXA = A, \tag{1}$$

$$XAX = X, \tag{2}$$

$$(AX)^* = AX, \tag{3}$$

$$(XA)^* = XA \tag{4}$$

Con estas ecuaciones se puede saber si la inversa que nos entrega el sistema de la matriz que introdujo el usuario es la inversa de Moore-Penrose.

El primer paso para poder introducir la matriz al sistema es guardarla en un archivo de texto, el formato con el que se debe guardar esta matriz para definir las columnas es el siguiente: los números deben estar separados por un espacio, y para definir las filas se debe introducir un salto de línea entre cada fila.

A continuación se muestra el proceso de prueba para encontrar la inversa de Moore-Penrose de la matriz:

$$A = \begin{bmatrix} -6 & 2 & -2 \\ 3 & -1 & 1 \end{bmatrix}.$$

Como se puede observar esta matriz es rectangular, y por lo tanto no se puede obtener su inversa usual.

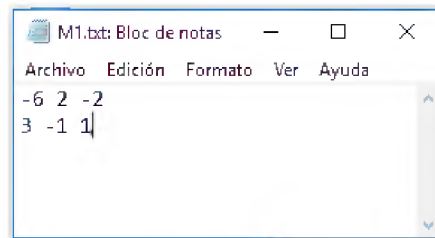


Figura 5.1: Matriz A de prueba para inversa de Moore-Penrose

En la figura 5.1 se observa la matriz A de tamaño 2×3 en un archivo de texto siguiendo el formato de separación de columnas y filas, esta matriz será introducida al sistema para posteriormente calcular su inversa de Moore-Penrose.



Figura 5.2: Carga de la matriz A al sistema

En la figura 5.2 se especifican los pasos para cargar al sistema la matriz que se encuentra en el archivo de texto:

- En (1) es donde va la ruta junto con el nombre y la extensión .txt del archivo donde se encuentra la matriz.
- En (2) y (3) colocamos el tamaño de la matriz.
- Para que el sistema pueda cargar la matriz a la memoria tenemos que seleccionar (4).
- En (5) el sistema mostrará la matriz cargada en memoria.

Con lo anterior la matriz quedará lista para que el sistema pueda generar la aproximación a la inversa de elección del usuario.



Figura 5.3: Inversa de Moore-Penrose de la matriz A

La aproximación a la inversa de Moore-Penrose se debe hacer con la selección de **(1)** como se muestra en la Figura 5.3 de esta manera el sistema calcula la inversa de Moore-Penrose y el resultado lo muestra de dos formas, la primera es en **(2)** donde el usuario tendrá una vista rápida del resultado de la inversa y la otra forma será mediante un archivo de texto, donde se guardará la inversa de Moore-Penrose, este archivo se guardará en la misma ruta donde se encuentre el archivo original.

```

A =
  -6     2    -2
   3    -1     1

X =
-0.109090909090909  0.054545454545455
 0.036363636363636 -0.018181818181818
-0.036363636363636  0.018181818181818

AXA =
[ -6.0,  2.0, -2.0]
[  3.0, -1.0,  1.0]
  
```

Figura 5.4: Prueba 1 de la inversa de Moore-Penrose utilizando la ecuación (1)

Antes de mostrar el resultado al usuario, el sistema verifica que la matriz generada sea la inversa de Moore-Penrose, en este proceso se ocupan las cuatro ecuaciones de Penrose, la primer ecuación a verificar es $AXA = A$, en la figura anterior se observan tres matrices, la matriz A que corresponde a la matriz de prueba de la figura 5.1, X corresponde a la matriz inversa de Moore-Penrose mientras que AXA es la multiplicación de ambas matrices, en esta prueba se observa que los resultados obtenidos cumplen con la ecuación $AXA = A$.

```

X =
[ -0.109090909090909090909090909091,  0.0545454545454545454545454545455]
[  0.0363636363636363636363636363636, -0.0181818181818181818181818181818]
[ -0.0363636363636363636363636363636,  0.0181818181818181818181818181818]

XAX =
[ -0.109090909090909090909090909091,  0.0545454545454545454545454545455]
[  0.0363636363636363636363636363636, -0.0181818181818181818181818181818]
[ -0.0363636363636363636363636363636,  0.0181818181818181818181818181818]

```

Figura 5.5: Prueba 2 de la inversa de Moore-Penrose utilizando la ecuación (2)

En la segunda verificación de la inversa de Moore-Penrose se aplica la ecuación (2), y en la figura 5.5 observamos que el resultado de la multiplicación XAX es la matriz X por lo que la ecuación (2) se cumple.

```

AXT =
[  0.8, -0.4]
[ -0.4,  0.2]

AX =
[  0.8, -0.4]
[ -0.4,  0.2]

```

Figura 5.6: Prueba 3 de la inversa de Moore-Penrose utilizando la ecuación (3)

La tercer prueba consiste en resolver la ecuación (3), en donde se transpone el producto de la multiplicación AX y se iguala al producto de la multiplicación AX , estos resultados se

$$A^k X = X A^k, \quad (5^k)$$

$$A X^k = X^k A, \quad (6^k)$$

Las pruebas que se realizaron para la inversa de Drazin son muy similares a las pruebas de la inversa de Moore-Penrose. A continuación se determinará la inversa de Drazin de la siguiente matriz:

$$A = \begin{bmatrix} 2 & 4 & 6 & 5 \\ 1 & 4 & 5 & 4 \\ 0 & -1 & -1 & 0 \\ -1 & -2 & -3 & -3 \end{bmatrix}$$

esta matriz es cuadrada pero no es invertible, como prueba se ocupó la función `inv()` de la librería de Matlab:

```
>> A=[2 4 6 5;1 4 5 4;0 -1 -1 0;-1 -2 -3 -3]

A =

     2     4     6     5
     1     4     5     4
     0    -1    -1     0
    -1    -2    -3    -3

>> inv(A)
Warning: Matrix is singular to working precision.

ans =

    Inf    Inf    Inf    Inf
    Inf    Inf    Inf    Inf
    Inf    Inf    Inf    Inf
    Inf    Inf    Inf    Inf
```

Figura 5.9: Verificación de singularidad de la matriz A

En la figura 5.9 se observa el uso de la función `inv()` para verificar que la matriz es no invertible, como resultado obtenemos que la matriz es singular por lo tanto no se puede invertir, con este resultado procedemos a generar la aproximación a su inversa de Drazin.

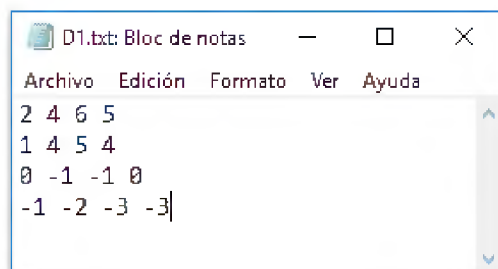


Figura 5.10: Matriz A de prueba para inversa de Drazin

La figura anterior muestra la matriz de prueba que se ocupará para determinar la inversa de Drazin, esta matriz debe ser cargada al sistema y el procedimiento para hacerlo es el mismo que se ocupa para la inversa de Moore-Penrose (ver Figura 5.2).

Cuando la matriz esté cargada en el sistema, el usuario podrá seleccionar la opción de la inversa de Drazin, en la figura 5.11 se observa la selección de **(1)** indicando que la inversa que se desea obtener es la inversa de Drazin y en **(2)** aparece la inversa de Drazin.

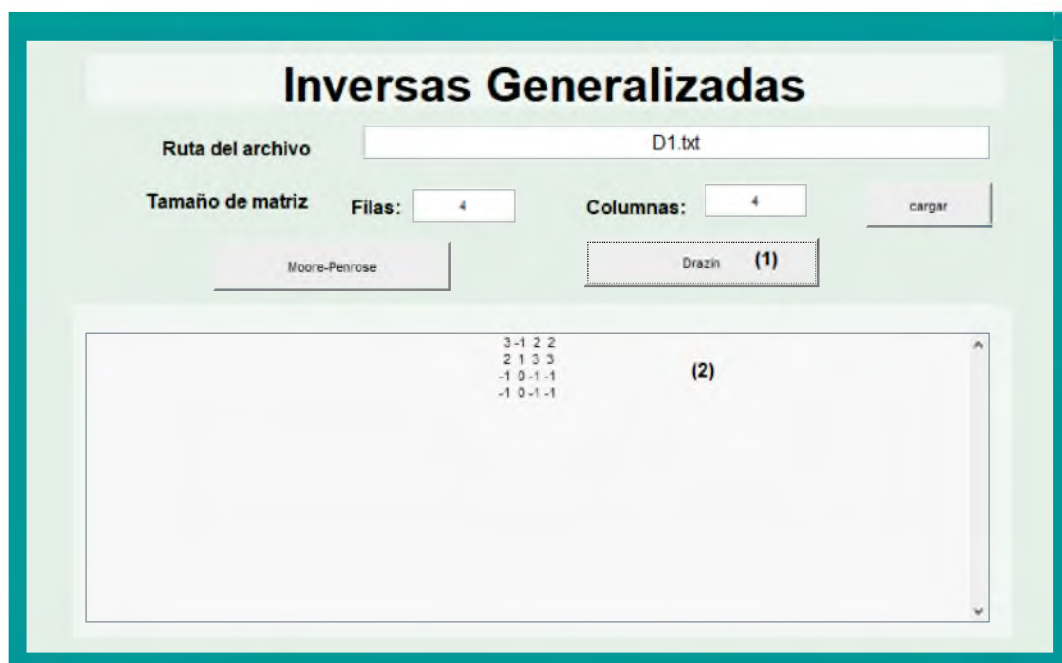


Figura 5.11: Inversa de Drazin de la matriz A (Figura 5.10)

Antes de que el sistema entregue el resultado como se muestra en la Figura 5.11, se verifican las ecuaciones mostradas al inicio de esta sección:

```

AkXA =
[ 3.0, 14.0, 17.0, 17.0]
[ 2.0, 11.0, 13.0, 13.0]
[ -1.0, -5.0, -6.0, -6.0]
[ -1.0, -5.0, -6.0, -6.0]

Ak =
[ 3.0, 14.0, 17.0, 17.0]
[ 2.0, 11.0, 13.0, 13.0]
[ -1.0, -5.0, -6.0, -6.0]
[ -1.0, -5.0, -6.0, -6.0]

```

Figura 5.12: Prueba 1 de la inversa de Drazin utilizando la ecuación (1^k)

En la Figura 5.12 se observan los resultados de la ecuación (1^k) , en el caso de la multiplicación $A^k X A$ esta representado como $AkXA$ en la imagen, donde k es el índice de la matriz, por lo que A^k es la matriz A elevada a la k potencia, y A^k esta representada como Ak en la imagen, con estos resultados se comprueba que la ecuación (1^k) se cumple para esta matriz.

```

AX =
    3    2    5    5
    2    3    5    5
   -1   -1   -2   -2
   -1   -1   -2   -2

XA =
    3    2    5    5
    2    3    5    5
   -1   -1   -2   -2
   -1   -1   -2   -2

```

Figura 5.13: Prueba 2 de la inversa de Drazin utilizando la ecuación (5)

En la Figura 5.14 se muestran los resultados de aplicar la ecuación (5), donde X es la inversa de Drazin asociada a la matriz A , en esta figura se observan las multiplicaciones XA y AX concluyendo así que la ecuación (5) se cumple.

```

AkX =
[ 3.0, 11.0, 14.0, 14.0]
[ 2.0, 9.0, 11.0, 11.0]
[ -1.0, -4.0, -5.0, -5.0]
[ -1.0, -4.0, -5.0, -5.0]

XAk =
[ 3.0, 11.0, 14.0, 14.0]
[ 2.0, 9.0, 11.0, 11.0]
[ -1.0, -4.0, -5.0, -5.0]
[ -1.0, -4.0, -5.0, -5.0]

```

Figura 5.14: Prueba 3 de la inversa de Drazin utilizando la ecuación (5^k)

En la figura anterior se hace la verificación de la inversa mediante la ecuación (5^k) , donde AkX corresponde a la multiplicación A^kX y XAk corresponde a la multiplicación XA^k , teniendo ambos resultados podemos observar que la ecuación (5^k) se cumple, por lo que ahora sólo falta verificar que se cumpla la ecuación (6^k) para la matriz de prueba A .

```

AXk =
  3   -7   -4   -4
  2   -3   -1   -1
 -1    2    1    1
 -1    2    1    1

XkA =
  3   -7   -4   -4
  2   -3   -1   -1
 -1    2    1    1
 -1    2    1    1

```

Figura 5.15: Prueba 4 de la inversa de Drazin utilizando la ecuación (6^k)

En la Figura 5.15 se observa la última verificación de la inversa que nos entrega el sistema, en donde se aplica la ecuación (6^k) , observando que la igualdad en la ecuación se cumple con los resultados obtenidos. La etapa de pruebas de la inversa de Drazin queda concluida con la verificación de las cuatro ecuaciones para matrices con propiedades espectrales, cabe señalar que el uso de la función $vpa()$ ayudó en gran medida a que se lograra este resultado, parte

del proceso para verificar que dichos resultados fueran los correctos fue aplicar las ecuaciones de Penrose en el caso de la inversa de Moore-Penrose y las ecuaciones complementarias para las inversas espectrales en el caso de la inversa de Drazin.

Capítulo 6

Conclusiones

En esta tesis se recopiló información de las dos inversas generalizadas más conocidas y aplicadas, que son la inversa de Moore-Penrose y la inversa de Drazin. También se abordaron temas como la grupo inversa, construcción y restricciones de $\{1\}$, $\{1, 2\}$, $\{1, 2, 3\}$ y $\{1, 2, 4\}$ -inversas y los aspectos computacionales de las inversas generalizadas en el capítulo 3.

Después de haber investigado sobre estas inversas se aplicaron algoritmos que se encontraron dentro de las referencias consultadas para facilitar el cálculo de las inversas, estos algoritmos fueron implementados en Matlab complementándolos con una interfaz gráfica que fuera amigable para el usuario, el tiempo que se lleva el sistema en calcular las inversas generalizadas es mucho menor que si se calcularan manualmente.

El desarrollo del sistema se muestra en el capítulo 4, donde se expone la implementación de los algoritmos mostrados en el capítulo 2 para el cálculo de la aproximación de cada inversa, además de las funciones que se crearon y las que se ocuparon de la librería de Matlab. El capítulo 5 está orientado a las pruebas que se elaboraron sobre el sistema, cabe señalar que fueron varias pruebas las que se hicieron al sistema pero sólo se expone una prueba por cada inversa generalizada. El sistema de aproximación de inversas expuesto en esta tesis será ocupado en el departamento de Matemáticas específicamente para el estudio de inversas generalizadas e inversas espectrales.

En un trabajo posterior, se puede ampliar el sistema para que no sólo ayude en el cálculo de las inversas de Drazin y las inversas de Moore-Penrose sino también en el cálculo de otras inversas generalizadas, e incluir matrices de números complejos.

Bibliografía

- [1] Algebraic simplification - matlab simplify - mathworks. <https://la.mathworks.com/help/symbolic/simplify.html>.
- [2] Create symbolic polynomial from vector of coefficients - matlab. <https://la.mathworks.com/help/symbolic/poly2sym.html>.
- [3] Create symbolic variables and functions - matlab syms. <https://la.mathworks.com/help/symbolic/syms.html>.
- [4] Differences and approximate derivatives - matlab diff. <https://la.mathworks.com/help/matlab/ref/diff.html>.
- [5] Eigenvalues and eigenvectors - matlab eig - mathworks. <https://www.mathworks.com/help/matlab/ref/eig.html>.
- [6] *Matlab*. <https://www.mathworks.com/products/matlab.html>.
- [7] Matrix inverse - matlab inv. <https://la.mathworks.com/help/matlab/ref/inv.html>.
- [8] Open guide - matlab guide. <https://la.mathworks.com/help/matlab/ref/guide.html>.
- [9] Real part of complex number - matlab real - mathworks. <https://la.mathworks.com/help/matlab/ref/real.html>.
- [10] Round to nearest decimal or integer - matlab round. <https://la.mathworks.com/help/matlab/ref/round.html>.
- [11] *Variable-precision arithmetic - MATLAB vpa*. <https://la.mathworks.com/help/symbolic/vpa.html?requestedDomain=true>.

- [12] BEN-ISRAEL, A., AND GREVILLE, T. N. E. *Generalized Inverses Theory and Applications*, second ed. Springer-Verlag, New York, 2003.
- [13] BOULLION, T. L., AND ODELL, P. L., Eds. *Proceedings of the Symposium on Theory and Applications of Generalized Inverses of Matrices* (Lubbock, Texas, 1968), Tech. Press.
- [14] BOUNITZKY, E. Sur la fonction de greendes équations differentielles linéaires ordinaires. *J. Math Pures Appl.* 5, no. 6 (1909), 65–125.
- [15] CAMPBELL, S. L., AND MEYER, C. D. *Generalized Inverses of Linear Transformations*, first edition ed. Dover, New York, 1979.
- [16] DRAZIN, M. P. Pseudo inverses in associative rings and semigroups. *The American Mathematical Monthly* 65 (Aug. - Sep 1958), 506–514.
- [17] FANBIN BU, Y. W. The algorith for computing the drazin inverses of two-variable polynomial matrices. *Applied Mathematics and Computation* (2004).
- [18] FREDHOLM, I. Sur une classe d'équations fonctionnelles. *Acta Math* 27 (1903), 365–390.
- [19] GREVILLE, T. N. E. The souriau-frame algorithm and the drazin pseudoinverse*. *Linear Algebra and Its Aplications* 6 (1973), 205–208.
- [20] HURWITZ, W. On the pseudo-resolvent to the kernel of an integral equation. *Trans.Amer.Math.Soc.* 13 (1912), 405–418.
- [21] JI, J. A finite algorithm for the drazin inverse of a polynomial matrix. *Applied Mathematics and Computation* (2002).
- [22] MOORE, E. H. On the reciprocal of the general algebraic matrix. *Amer. Math. Soc.* 26, pp. 406–413.
- [23] PENROSE, R. A generalized inverse for matrices. In *Proc. Cambridge* (1955), Philos. Soc, pp. 406–413.
- [24] QUARTERONI, A., SACCO, R., AND SALERI, F. *Numerical Mathematics*. Springer, 2000.
- [25] ROSE, N. J. A note on computing the drazin inverse. *Linear Alg. Applic.* (1976), 95–98.
- [26] SIEGEL, V. C. L. über die analytische teorie der quadratischen formen. *Annals of Mathematics* (1935), 527–606.

- [27] TSENG, Y.-Y. *The characteristics Value Problem of Hermitian Functional Operators in a Non-Hilbert Space*. Ph.d. in mathematics (published by the university of chicago libraries, 1936), University of Chicago, Chicago, 1933.