

Universidad Autónoma de Nuevo León  
Facultad de Ciencias Fisico Matemáticas

Metodos Matemáticos de la Fisica II

**Metodo numérico para obtener la matriz numerica  
inversa**

Lisette Galvan Tlapale

Raul Eduardo Santoy Flores

Fernando Martinez Diaz

10 de noviembre de 2020

## 1. Introducción

Los calculos operacionales de matrices cuadradas por lo regular suelen complicarse cuando se incrementa su dimensión. Recurrimos a solucionarlo numericamente implementando el metodo de la **adjunta** para obtener la matriz inversa y la **regla de Cramer** para obtener los determinantes de todas las matrices obtenidas. Escribimos el codigo en fortran.

## 2. Creando la matriz de $9 \times 9$

Construimos la matriz  $A$  de dimensión  $n = 9$  con los siguientes ciclos

```
n = 9
Do i = 1,n
  A(i,i) = 4
end do

Do i =1,n
  if (mod(i,3).ne.0.and.i.lt.9) then
    A(i,i+1) = -1
  end if
  if (mod(i-1,3).ne.0.and.i.gt.1) then
    A(i,i-1) = -1
  end if
  if (i.lt.7) then
    A(i,i+3) = -1
  end if
  if (i.gt.3) then
    A(i,i-3) = -1
  end if
end do
```

Como resultado obtenemos

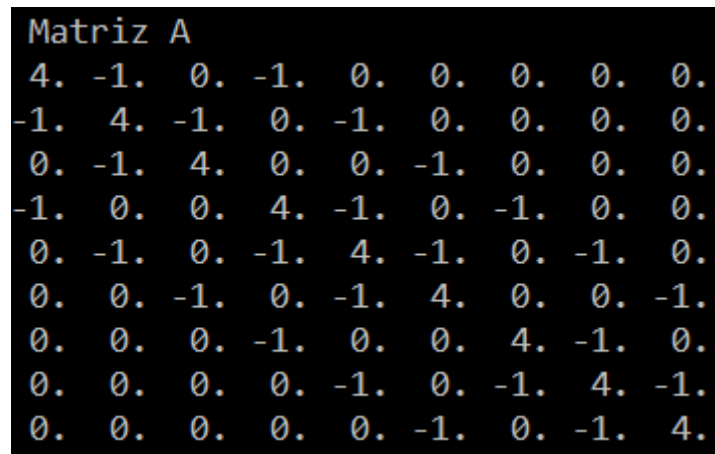


Figura 1: Contruyendo matriz  $A$  a partir de ciclos

Construimos nuestro vector  $Y(i)$

```

Y(1) = 75
Y(2) = 0
Y(3) = 50
Y(4) = 75
Y(5) = 0
Y(6) = 50
Y(7) = 175
Y(8) = 100
Y(9) = 150

```

Lo que sigue es aplicar el metodo de la adjunta para obtener la matriz inversa  $M$ , donde muchas de las variables son contadoras. Calculamos el determinante de  $A$  utilizando la subrutina TSuperior(n,A,b)

```

Call TSuperior(n,A,b)
DetA = (-1)**b
!Producto de la diagonal
Do i = 1, n
    DetA = DetA*A(i,i)
end do
Write(*,*) ' Det(A) =',DetA

```

Esta subrutina aprovecha las propiedades de los determinantes para reducir a una matriz triangular superior. La finalidad de esto es poder calcular el determinante de cualquier matriz cuadrada como el producto de los elementos en la diagonal de la matriz trangular superior que produce TSuperi-ro. La variable  $b$  es un contador de intercambio de renglones y  $n$  es la dimensión de la matriz a reducir. Las demas variales son contadoras o auxiliares.

```

!-----Reduccion a matriz triangular superior-----
Subroutine TSuperior(d,M,b)

Real M,C,R
Integer i,k,t,d,b
Dimension M(10,10),C(10),R(10)

b = 0
Do i = 1, d - 1

    !Intercambio de renglones
    k = 0
    Do While (M(i,i).eq.0.AND.i.lt.d.AND.k.lt.d-i)
        k = k + 1
        if (M(i+k,i).ne.0) then
            Do t = 1, d
                R(t) = M(i,t)
                M(i,t) = M(i+k,t)
                M(i+k,t) = R(t)
            end do
            b = b + 1
        end if
    end do

```

```

end do

!Vector columna
Do k = 1, d
    C(k) = M(k,i)
end do

!Operación entre renglones (+)(-)
Do k = i + 1, d
    t = 1
    !Proceso por eliminacion por el menor en el vector i
    Do While (C(k).ne.0.AND.t.le.d)
        M(k,t) = M(k,t) - (C(k)/C(i))*M(i,t)
        t = t + 1
    end do
end do

end do

Return
End Subroutine

```

---

El determinante de  $A$  tiene que ser distinto de cero para poder aplicar construir la matriz adjunta  $Adj$ . Iterativamente se vuelve a llamar a la subrutina  $TSuperior$  para reducir las matrices obtenidas en el procedimiento y poder calcular su determinante.

```

if (DetA.ne.0) then

Do i = 1, n
    Do j = 1, n

        !Construyendo el menor M(i,j)
        u = 1
        Do k = 1, n
            v = 0
            Do t = 1, n
                if (k.ne.i.AND.t.ne.j) then
                    v = v + 1
                    M(u,v) = A(k,t)
                end if
            end do
            if (v.eq.n-1) then
                u = u + 1
            end if
        end do

        d = n - 1
    end do
end if

```

```

!Determinante del menor M(i,j)
if (n.gt.2) then
    Call TSuperior(d,M,b)
end if
Det(i,j) = (-1)**b
Do k = 1, n - 1
    Det(i,j) = Det(i,j)*M(k,k)
end do

!Elemento (i,j) de la matriz adjunta
Adj(i,j) = (-1)**(i + j)*Det(i,j)

end do
end do

```

Por ultimo, se construye la matriz inversa IA

```

!Matriz inversa
Do i = 1, n
    Do j = 1, n
        IA(i,j) = Adj(j,i)/DetA
    end do
end do
Write(*,*) ' Matriz Inversa de A'
Write(*,*) ' '
Do i = 1, n
    Write(*,99) (IA(i,j), j = 1, n)
end do
99 Format(9f8.5,9X)

```

```

Matriz Inversa de A
0.25000 0.06667 0.01786 0.07177 0.04354 0.01946 0.02325 0.02239 0.01339
0.00000 0.26667 0.07143 0.01914 0.08989 0.04969 0.01341 0.03585 0.02679
0.00000 0.00000 0.26786 0.00478 0.02247 0.08613 0.00525 0.01652 0.03125
0.00000 0.00000 0.00000 0.26794 0.08427 0.02816 0.07957 0.05370 0.02679
0.00000 0.00000 0.00000 0.00000 0.29354 0.09317 0.02515 0.10448 0.06250
0.00000 0.00000 0.00000 0.00000 0.00000 0.29482 0.00760 0.03025 0.09821
0.00000 0.00000 0.00000 0.00000 0.00000 0.00000 0.26989 0.08795 0.03125
0.00000 0.00000 0.00000 0.00000 0.00000 0.00000 0.00000 0.29811 0.09821
0.00000 0.00000 0.00000 0.00000 0.00000 0.00000 0.00000 0.00000 0.29911

```

Figura 2: Matriz Inversa de A

```

Write(*,*) ' '
Write(*,*) 'Soluciones T(i,j) expresada en vector'
Write(*,*) ' '
Do i = 1, n
    T(i) = 0

```

```

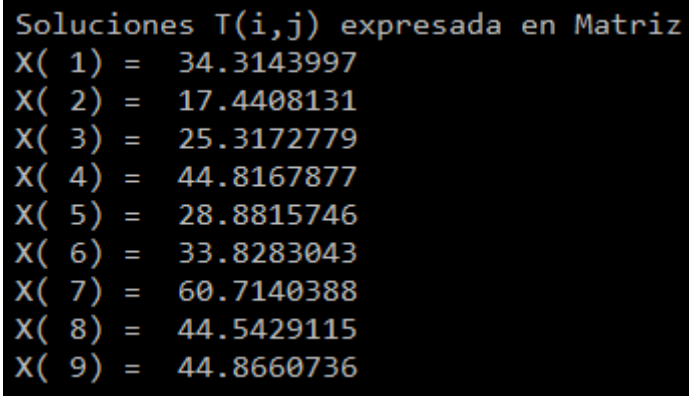
      Do j = 1, n
        T(i) = T(i) + IA(i,j)*Y(j)
      end do
      Write(*,*) 'T(',i,') =',X(i)
    end do

  end if

  PAUSE
  END PROGRAM

```

Obteniendo como resultados



```

Soluciones T(i,j) expresada en Matriz
X( 1) = 34.3143997
X( 2) = 17.4408131
X( 3) = 25.3172779
X( 4) = 44.8167877
X( 5) = 28.8815746
X( 6) = 33.8283043
X( 7) = 60.7140388
X( 8) = 44.5429115
X( 9) = 44.8660736

```

Figura 3: Soluciones  $T_{i,i}$