

Regla de Cramer implementada en Python

Autor Ing Sergio Adrián Martín

Introducción

La regla de Cramer es un método para resolver ecuaciones simultáneas, un poco más complejo que el método de Gauss, que implica que para un sistema de n ecuaciones con n incógnitas hay que resolver $n+1$ determinantes surgidos de el sistema de ecuaciones a continuación se explica la base de este ritmo.

Principio básico

Sí existe una matriz a formada por los coeficientes de las incógnitas en el sistema de ecuaciones , existirá un determinante de que será el determinante característico del sistema. Asimismo habrá un vector B que estará formado por los coeficientes de los términos independientes del sistema de ecuaciones, para encontrar el valor de la variable x_1 deberá generarse una matriz A1 en la que se sustituye la columna 1 de la matriz a por los valores del vector B. Sí a continuación se divide el determinante de A1 entre el determinante característico de este cociente será igual al valor de X_1 . Este procedimiento se puede repetir para encontrar el valor de X_2 , sustituyendo la columna 2 en la matriz por los valores del vector B y luego dividiendo el determinante correspondiente entre el determinante característico.

De la misma forma se puede llevar a cabo un proceso similar para encontrar x_3 , x_4 o cualquier otra incógnita dependiendo del número de incógnitas que exista en el sistema de ecuaciones. En otras palabras un proceso repetitivo qué depende por supuesto de que el determinante característico del sistema no de un valor de 0, de otra forma el sistema resulta irresoluble.

A continuación se incluye el código fuente en python para implementar la regla de cramer.

```
#resolver ecuaciones por kramer
#por Sergio Martin
import math
import numpy as np
def deter(c,m):
    if m==2:
        d=c[0][0]*c[1][1]-c[1][0]*c[0][1]
        return d
    else:
        if m==3:
            f=np.ones((2,2),float)
            d=0
            f[0][0]=c[1][1]
            f[0][1]=c[2][1]
            f[1][0]=c[1][2]
            f[1][1]=c[2][2]

            s=1
            d=c[0][0]*deter(f,2)
            s=-s
            f[0][0]=c[1][0]
            f[0][1]=c[2][0]
            f[1][0]=c[1][2]
            f[1][1]=c[2][2]
            d=d+s*c[0][1]*deter(f,2)
            s=-s
            f[0][0]=c[1][0]
```