Informe Laboratorio 2 Computación Científica

Josué Peña A.

Febrero 2020

Contents

1 Introducción

En este laboratorio se realiza la implementación en Matlab de dos soluciones para resolver un problema lineal de ajuste de datos de mínimos cuadrados para matrices rectangulares de mxn, con $m > n$. Estos dos soluciones son el método de ecuaciones normales y el factorización QR mediante transformaciones Householder. Para estos dos últimos, se implementaron las funciones *normal_equations.m* y *householder.m*, respectivamente.

Así mismo, se implementó la función *gen_matrix_A.m* para generar la matriz de Vandermonte para el problema de ajuste de datos, y las funciones *gen_v.m* y *gen_H.m* para calcular el vector v_i y la matriz H para las transformaciones Householder.

También, se implementaron las funciones
suc_sust_upper.m y *suc_sust_lower.m*, para despejar sistemas lineales triangulares superiores e inferiores mediante sustitución sucesiva, respectivamente. Estas dos últimas funciones son modificaciones de la función de sustitución sucesiva utilizada en el laboratorio 1.

Por último se realizó el procedimiento *main.m* en donde se ingresan los parámetros para las dos métodos (grado del polinomio a aproximar, vector de tiempos t, y vector de observaciones y), y se calculan y grafican los resultados obtenidos de cada uno. Los valores usados para cada parámetro son los mismos del ejemplo visto en clase:

$$d = 2, \, t = \begin{bmatrix} -1.0 \\ -0.5 \\ 0 \\ 0.5 \\ 1.0 \end{bmatrix}, \, y = \begin{bmatrix} 1.0 \\ 0.5 \\ 0 \\ 0.5 \\ 2.0 \end{bmatrix}$$

2 Método ecuaciones normales

La función implementada recibe los parámetros previamente definidos, y retorna solamente el vector solución x. Dentro del procedimiento se genera la matriz A, se calculan las matrices $A^\top b$ y $A^\top A$, se calcula la descomposición de Choleski de esta última (haciendo uso de la función *chol* de Matlab), y se despejan las variables y y x mediante sustitución sucesiva inferior y superior, correspondientemente.

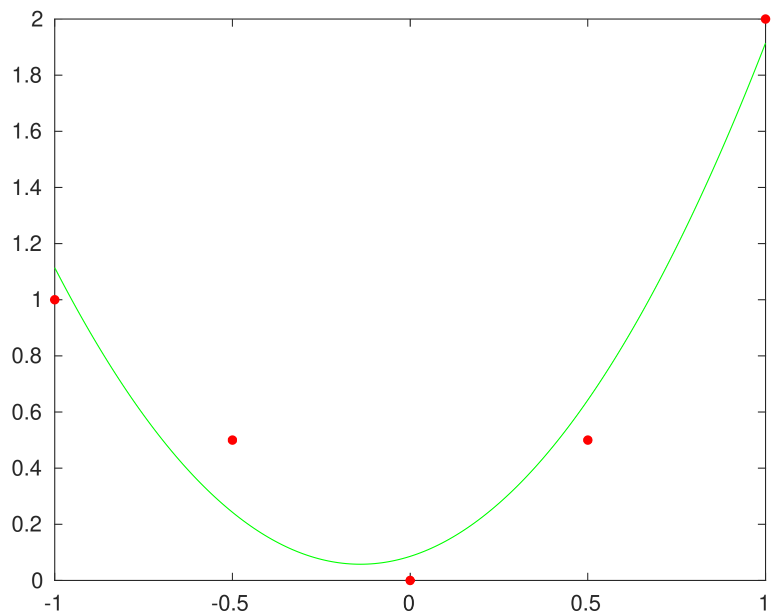

Figure 1: Polinomio aproximado mediante Ecuaciones Normales

3 Transformaciones Householder

En la función implementada, al igual que el método anterior, recibe los mismos parámetros y solo retorna el vector solución x. Se general la matriz A y se prosigue a realizar un ciclo que calcule el vector v_i y la matriz de transformación H_i para cada columna i de la matriz A.

En la función que calcula el vector v_i, se verifica adecuadamente si el pivote de la columna de la matriz A es positivo. En tal caso, se pone el valor de la norma del la columna como negativo en el vector e_i que está restando en la fórmula para calcular el vector v. Aunque, tal vector e_i no se calcula explícitamente en el algoritmo, sino que solo se modifica directamente la i-esima columna de la matriz A.

Finalmente se resuelve el sistema triangular superior $Rx = y$ realizando el despeje por sustitución sucesiva con las primeras $d + 1$ filas de la matriz A y el vector v resultantes luego de las transformaciones.

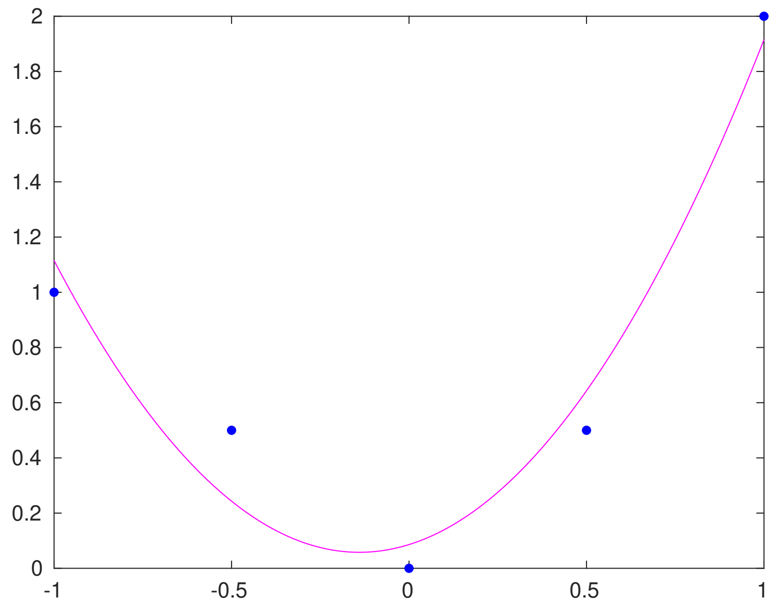

Figure 2: Polinomio aproximado mediante Householder

4 Conclusiones

Se puede concluir que ambos métodos generan los mismos resultados de aproximación para el sistema planteado. Sin embargo, en el caso de la implementación presentada en este laboratorio, aplicar el método de transformaciones Householder resulta computacionalmente más costoso que el de ecuaciones normales (tanto en complejidad temporal como en uso de memoria, debido a la constante alocación dinámica con la creación de los vectores v_i y las matrices H_i).

Esto es debido a que se calcula la matriz H explícitamente en cada iteración por cada una de las columnas de la matriz A, las cuales deben multiplicarse. Para sistemas lo suficientemente grandes, es más eficiente aplicar el método de ecuaciones normales, ya que no hay que realizar tantas multiplicaciones repetitivas entre matrices.

Aunque como se menciona en el libro guía, no podría siempre ser el más indicado ya que puede producir resultados poco precisos debido a la perdida de información que puede existir en la creación de la matriz A, ya que los números generados pueden ser tan pequeños que podrían salirse de los límites de representación del sistema de punto flotante de la máquina en la que se esté trabajando.