

Métodos Numéricos II 2023

Lista 03

29.agosto.2023

1. Implementar el método de las potencias y el método QR , para calcular todos los autovalores y autovectores de una matriz simétrica $A \in \mathbb{R}^{n \times n}$. Aplicarlo a las matrices siguientes:

$$A = \begin{pmatrix} 4 & -1 & 0 & 0 \\ -1 & 4 & -1 & 0 \\ 0 & -1 & 4 & -1 \\ 0 & 0 & -1 & 4 \end{pmatrix} \quad B = \begin{pmatrix} 1 & \frac{1}{2} & \frac{1}{3} & \frac{1}{4} \\ \frac{1}{2} & \frac{1}{3} & \frac{1}{4} & \frac{1}{5} \\ \frac{1}{3} & \frac{1}{4} & \frac{1}{5} & \frac{1}{6} \\ \frac{1}{4} & \frac{1}{5} & \frac{1}{6} & \frac{1}{7} \end{pmatrix}.$$

En cada una de las matrices, elabore una gráfica del error de convergencia $\|\ell_k - \ell_{k-1}\|$, en función del número de iteración k , donde ℓ es el vector que contiene a los autovalores de la matriz.

Comparar ambos métodos y discutir cuál tiene mejor desempeño.

2. Leer el ejemplo 7.6.1 (páginas 559-566) de libro de C. Meyer *Matrix Analysis and Applied Linear Algebra*.

Aplicar el método QR anterior, para calcular todos los autovalores de una matriz tridiagonal A como en la ecuación (7.6.2), de tamaño 1000×1000 (esta matriz resultaría al hacer un mallado con 1000 nodos para resolver el sistema de EDO (7.6.2) de forma numérica).

A partir de éstos, hallar y graficar los diez primeros modos de vibración de la viga (Figura 7.6.3). Por simplicidad, asuma $T = mL$. Elabore una tabla que relacione cada autovalor λ_i , $i = 1, 2, \dots, 10$, con su respectiva frecuencia de vibración y el período de oscilación correspondiente a los primeros 10 modos de vibración.

3. Implementar un algoritmo para convertir una matriz $A \in \mathbb{R}^{m \times n}$ en alguna representación rara. El input de su algoritmo debe ser un archivo `.npy` (que es un archivo de Python para guardar matrices en formato denso). Además, usted debe indicar como argumento, en cuál formato raro debe construir la matriz. La idea es elegir cualquiera de los siguientes formatos:
 - COO (formato coordenado),
 - CSR (*compressed sparse row*),
 - CSC (*compressed sparse column*).

La salida de su algoritmo debe ser un archivo `.npy`, `.csv` ó `.txt` donde se indique la matriz en el formato raro elegido.

Mostrar en su tarea el resultado del algoritmo aplicado a dos matrices diferentes (mostrar los tres tipos de salida para cada matriz).

4. Implementar en Python el algoritmo para sumar dos matrices $A, B \in \mathbb{R}^{m \times n}$ en formato disperso COO. En este caso, se debe implementar el algoritmo desde cero. No está permitido usar las funciones de `scipy.linalg.sparse`, salvo para verificar su resultado.
