

Métodos Numéricos II 2021

Lista 03

15.agosto.2021

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. **Segmentación Espectral (Spectral Clustering).**

En esta tarea vamos a implementar un método para binarizar (segmentar en dos componentes) una imagen RGB.

Dada una imagen a colores (por ejemplo, en formato RGB), en general la segmentación consiste en clasificar cada píxel de la imagen en k grupos, atendiendo a características comunes como color, textura y posición espacial. En particular, la segmentación binaria consiste en clasificar los píxeles en dos grupos: 0 = fondo (*background*), y 1 = objeto de interés (*foreground*).

Una técnica para segmentación binaria consiste en la segmentación espectral.

Dada una imagen I de $h \times w$ píxeles (h = número de filas o altura de la imagen, w = número de columnas o ancho de la imagen), el método de segmentación espectral construye un grafo G con hw vértices, uno por cada píxel de la imagen.

Cada píxel es conectado con sus cuatro vecinos (norte, sur, este y oeste), y las aristas de conexión se ponderan de acuerdo a la similitud de los píxeles vecinos que se unen, mediante alguna métrica de afinidad. Por ejemplo, una de las métricas de afinidad más comunes se construye mediante un kernel gaussiano

$$\text{similarity}(p_i, p_j) = \exp \left(-\frac{1}{2\sigma^2} (\alpha \|color(i) - color(j)\| + \beta \|position(i) - position(j)\|) \right).$$

Aquí, $color(i)$ y $color(j)$ corresponden al vector de colores RGB de los píxeles i y j , respectivamente. Similarmente, $position(i)$ y $position(j)$ son las coordenadas (x, y) de los píxeles i y j , en la matriz $h \times w$ de la imagen I .

Así, $\|color(i) - color(j)\|$ mide la distancia de color entre los píxeles i y j , mientras que $\|position(i) - position(j)\|$ mide la distancia espacial entre estos píxeles. $\alpha > 0$ y $\beta > 0$ son coeficientes que ponderan la diferencia de color y de distancia espacial, y $\sigma > 0$ es un parámetro de escala dependiendo de la imagen.

- (a) El primer paso de la segmentación espectral es construir una matriz de afinidad W , que representa al grafo de conexiones G , pesado mediante la métrica de afinidad anterior. Esta matriz $W \in \mathbb{R}^{hw \times hw}$ es simétrica, y usualmente es de dimensiones muy grandes, con pocas conexiones. Así que W se trabaja como una matriz rara.

- (b) Construida la matriz de afinidad $W = (w_{ij})$, se construye la matriz Laplaciana del grafo G . Para ello, en cada fila de la matriz W se calcula

$$d_i = \sum_{j=1}^{hw} w_{ij}.$$

Esto es, d_i es la suma de afinidades en la fila i (d_i se llama usualmente el **grado** del vértice i). Luego, se construye la matriz diagonal

$$D = \text{diag}(d_1, d_2, \dots, d_{hw}).$$

El Laplaciano del grafo G se define como la matriz

$$L = W - D.$$

Sin embargo, en lugar de L , se utiliza el llamado *Laplaciano normalizado* de G , definido como

$$\mathcal{L} = D^{-1/2} L D^{-1/2} = D^{-1/2} (W - D) D^{-1/2}.$$

- (c) Con el Laplaciano normalizado \mathcal{L} , calculamos el autovector asociado al segundo menor autovalor de \mathcal{L} (el menor autovalor siempre es 0). Este autovalor \mathbf{v}_{n-1} es llamado el **vector de Fiedler**.
- (d) Finalmente, los signos del vector de Fiedler \mathbf{v} producen la segmentación binaria requerida. Esto es, si la i -ésima entrada v_i del vector de Fiedler satisface $v_i > 0$, el píxel i de la imagen se asigna al grupo 1, mientras que si $v_i < 0$, el píxel i se asigna al grupo 0. Esto se hace para cada uno de los píxeles de la imagen. Al final de este paso, se obtiene una segmentación *background-foreground* de la imagen I .



Su trabajo consiste en implementar un método que haga una segmentación espectral a partir de la matriz de afinidad W . Para simplificar el trabajo, se da un código en Python que dada una imagen pequeña I (de 80×120 píxeles) produce una matriz W en formato sparse. A partir de W , obtener la segmentación binaria de la imagen I :

- Calcular la laplaciana L y la laplaciana normalizada \mathcal{L} en formato sparse.
- Hallar el segundo menor autovalor de \mathcal{L} y el vector de Fiedler \mathbf{v}_{n-1} .
- Calcular los signos de las entradas de \mathbf{v} y construir una máscara binaria en función de estos signos.
- Mostra la imagen original, y su segmentación binaria respectiva.

Repetir esto para las diferentes imágenes proporcionadas.

http://www.tml.cs.uni-tuebingen.de/team/luxburg/publications/Luxburg07_tutorial.pdf
<https://ranger.uta.edu/~chqding/Spectral/>