Matrices tridiagonales

Este archivo es un ejemplo de solución de la primera tarea de matrices especiales.

Explicación de la estructura de las matrices tridiagonales

Sean $a \in \mathbb{R}^5$, $b, c \in \mathbb{R}^4$. Entonces la *matriz tridiagonal* generada por a, b, c es

$$\begin{bmatrix} a_1 & b_1 & 0 & 0 & 0 \\ c_1 & a_2 & b_2 & 0 & 0 \\ 0 & c_2 & a_3 & b_3 & 0 \\ 0 & 0 & c_3 & a_4 & b_4 \\ 0 & 0 & 0 & c_4 & a_5 \end{bmatrix}.$$

En general, suponemos que $a \in \mathbb{R}^n$, $b, c \in \mathbb{R}^{n-1}$. Entonces la matriz generada por a, b, c es de tamaño $n \times n$. Denotemos esta matriz por T. Para cualesquiera $j, k \in \{1, \ldots, n\}$, la entrada (j, k) de la matriz A es

$$T_{j,k} = \begin{cases} a_j, & j = k; \\ b_j, & j + 1 = k; \\ c_{j-1}, & j = k + 1; \\ 0, & \text{en otros casos.} \end{cases}$$

Para muchas otras clases interesantes de matrices, sus entradas no se pueden describir por fórmulas tan simples, y es suficiente explicar su estructura con uno o dos ejemplos.

Notamos que la matriz se determina por las entradas de los vectores a, b, c, y el tamaño total de estos datos iniciales es

$$D(n) = n + 2(n - 1) = 3n - 2.$$

El tamaño de los datos iniciales es pequeño en comparación con n^2:

$$\lim_{n \to \infty} \frac{D(n)}{n^2} = 0.$$

Función que construye matrices tridiagonales a partir de los datos iniciales

```
function [T] = tridiagonal_full(a, b, c),
   n = length(a);
   T = zeros(n);
   for j = 1 : n,
      T(j, j) = a(j);
   end
   for j = 1 : n - 1,
      T(j, j + 1) = b_j;
      T(j + 1, j) = c_j;
   end
end
```

Podemos generar la misma matriz de manera más eficiente, usando la función `diag` del lenguaje MATLAB:

```
function [T] = tridiagonal_full_2(a, b, c),
   T = diag(a) + diag(b, 1) + diag(c, -1);
end
```

Una prueba:

```
function [] = test_tridiagonal_full(),
   n = 5;
   a = 2 * rand(n, 1) - ones(n, 1);
   b = 2 * rand(n - 1, 1) - ones(n - 1, 1);
   c = 2 * rand(n - 1, 1) - ones(n - 1, 1);
   T = tridiagonal_full(a, b, c);
   T1 = tridiagonal_full_2(a, b, c);
   display(T);
   display(T1);
end
```

Matrices tridiagonales positivas definidas

Esta parte es optativa y se puede hacer después.
Luego vamos a resolver sistemas de ecuaciones lineales por el método de gradiente conjugado, el cual se aplica solamente a matrices simétricas positivas definidas. Encontremos una condición suficiente para que nuestras matrices sean simétricas y positivas definidas.

1 Proposición. *Sean $a \in \mathbb{R}^n$, $b, c \in \mathbb{R}^{n-1}$. Denotemos por T a la matriz tridiagonal generada por a, b, c.*

1. *T es simétrica si, y sólo si, $b = c$.*
2. *Definimos $c_0 = b_n = 0$ y supongamos que $b = c$ y*
$$\forall j \in \{1, \ldots, n\} \qquad a_j > |b_j| + |c_{j-1}|. \tag{1}$$
Entonces la matriz T es positiva definida.

Demostración. La primera afirmación es obvia. Supongamos que se cumplen las hipótesis del inciso 2. Para cada $j \in \{1, \ldots, n\}$ denotemos por R_j a la suma de los valores absolutos de las entradas en el j-ésimo renglón de la matriz, fuera de la diagonal principal:
$$R_j = \sum_{k \in \{1,\ldots,n\} \setminus \{j\}} |T_{j,k}| = |b_j| + |c_{j-1}|.$$
Por el teorema de Gershgorin, el espectro de la matriz está contenido en la unión de los discos cerrados con centros a_j y radios R_j. Como la matriz es simétrica, sus valores propios son reales, y en vez de los discos podemos hablar de los segmentos
$$[a_j - R_j, a_j + R_j], \qquad j = 1, \ldots, n.$$
Pero la condición (1) garantiza que $a_j - R_j > 0$, así que todos los valores propios de T son estrictamente positivos. □

La siguiente función muestra como generar de manera pseudoaleatoria matrices tridiagonales reales simétricas positivas definidas.

```
function [] = test_tridiagonal_positive(),
   n = 5;
   a = 2 * rand(n, 1) - ones(n, 1);
   b = 2 * rand(n - 1, 1) - ones(n - 1, 1);
   c = 2 * rand(n - 1, 1) - ones(n - 1, 1);
   bext = [b; 0];
   cext = [0; c];
   a = abs(bext) + abs(cext) + ones(n, 1) + rand(n, 1);
   T = tridiagonal_full(a, b, c);
   display(T);
   display(eig(T));
end
```