Cómputo Científico
Tarea 4
Cálculo de eigenvalores

Iván Irving Rosas Domínguez

11 de octubre de 2023

1. Dado el siguiente:

Teorema 1 (Gershgorin)**.** Dada una matriz $A = (a_{ij})$ de $m \times m$, cada eigenvalor de A está en al menos uno de los discos en el plano complejo con centro en a_{ii} y radio $\sum_{j \neq i} |a_{ij}|$. Además, si n de estos discos forman un dominio conexo, disjunto de los otros $m - n$ discos, entonces hay exactamente n eigenvalores en ese dominio.

Deduce estimaciones de los eigenvalores de

$$\begin{pmatrix} 8 & 1 & 0 \\ 1 & 4 & \epsilon \\ 0 & \epsilon & 1 \end{pmatrix}$$

con $|\epsilon| < 1$.

Solución: dado el teorema anterior, procedemos a calcular los discos de Gershgorin para $i = 1, 2, 3$.

a) $D_1 = \{z \in \mathbb{C} : |z - 8| \leq 1 + 0\} = D_1(8)$, el disco complejo de radio 1 con centro en 8.

b) $D_2 = \{z \in \mathbb{C} : |z - 4| \leq 1 + \epsilon\} = D_{1+\epsilon}(4)$, el disco complejo de radio 1 con centro en 4.

c) $D_3 = \{z \in \mathbb{C} : |z - 1| \leq \epsilon\} = D_\epsilon(1)$, el disco complejo de radio ϵ con centro en 1.

No obstante, dado que la matriz es simétrica y suponiendo que $\varepsilon \in \mathbb{R}$, entonces tiene entradas reales, así que sus eigenvalores son reales, por lo que podemos descartar los discos complejos y hablar de intervalos en la recta real. Luego, los Discos de Gershgorin están dados por

a) $D_1 = [7, 9]$, la bola cerrada con centro en 8 y radio 1,

b) $D_2 = [3 - \epsilon, 5 + \epsilon]$, la bola cerrada con centro en 4 y radio $1 + \epsilon$,

c) $D_3 = [1 - \varepsilon, 1 + \varepsilon]$, la bola cerrada con centro en 1 y radio ϵ,

2. Implementa la iteración QR con shift. Aplícala a la matriz A del Ejercicio 1 con $\epsilon = 10^{-N}$ para $N = 1, 3, 4, 5$.

Solución: para este ejercicio recurrimos al script de Python llamado *Ejercicio 2-5*. En él, se construye el algoritmo *QRshift*, el cual implementa justamente el algoritmo QR con shift para llevar a una matriz A a una matriz cuya diagonal posea los eigenvalores de la misma (siempre que esta sea diagonalizable). La función tiene dos argumentos, siendo el primero de ellos la matriz a la cual se le van a extraer los eigenvalores, y el segundo es el número de iteraciones que se quieren realizar en el algoritmo.

Dado que las entradas de la matriz debajo de la diagonal convergen a 0, un alto número de iteraciones puede producir un error, ya que se excede el tamaño mínimo manejable por la computadora.

El algoritmo tiene la posibilidad de regresar 3 objetos. El primero de ellos contiene a la matriz A que justamente resulta del algoritmo. El segundo regresa la matriz Q de la última iteración y que teóricamente converge a la matriz de eigenvectores de A. El tercero corresponde al vector de eigenvalores de A.

```
>>> A1=QRshift(B1,10)[2]
>>> A3=QRshift(B3,10)[2]
>>> A4=QRshift(B4,10)[2]
>>> A5=QRshift(B5,10)[2]
>>> A1
array([8.23614095, 3.76735469, 0.99650436])
>>> A3
array([8.23606798, 3.76393237, 0.99999965])
>>> A4
array([8.23606798, 3.76393203, 1.        ])
>>> A5
array([8.23606798, 3.76393202, 1.        ])
>>>
```

Figura 1: Vector de eigenvalores de cada una de las cuatro matrices correspondientes a $\varepsilon = 10^{-N}, N = 1, 3, 4, 5$.

Luego de cargar las cuatro matrices del ejercicio 1 e implementar el algoritmo con cada una, se obtienen los resultados de la figura 1. Observamos que en efecto conforme ε se hace más pequeño, el vector de los eigenvalores converge a (8,23606798, 3,76393202, 1.), los cuales son justamente los eigenvalores de la matriz

$$X = \begin{pmatrix} 8 & 1 & 0 \\ 1 & 4 & 0 \\ 0 & 0 & 1 \end{pmatrix},$$

como se puede corroborar en la figura 2 que contiene los eigenvalores de la matriz anterior y que fueron hallados usando la función *scipy.linalg.eigvals*.

```
array([8.23614095, 3.76735469, 0.99650436])
>>> A3
array([8.23606798, 3.76393237, 0.99999965])
>>> A4
array([8.23606798, 3.76393203, 1.        ])
>>> A5
array([8.23606798, 3.76393202, 1.        ])
>>> eigvals(X)
array([8.23606798+0.j, 3.76393202+0.j, 1.        +0.j])
>>>
```

Figura 2: Vector de eigenvalores de la matriz X. Los resultados coinciden con los del algoritmo QR haciendo ε pequeño, y con las estimaciones de los discos de Gershgorin

3. Determina todos los eigenvalores y eigenvectores de una matriz de Householder. Aseguramos que los eigenvalores de una matriz de Householder son solo 1 y -1.

Demostración. Por definición, una matriz de Householder está dada por $P = I_n - 2vv^*$, donde v es un vector de norma 1 que es normal a un hiperplano generado por $n-1$ vectores $v_n, ..., v_n$ en $\mathbb{R}^n$. Notemos que si realizamos $Pv_i, i \in \{2, ..., n\}$, se tiene que

$$Pv_i = I_n v_i - 2vv^* v_i = v_i - 2v(0) = v_i,$$

ya que $v^* v_i = 0$ pues v es ortogonal a v_i, con $i \in \{2, ..., n\}$. Por lo tanto, tenemos que para cualquier $u \in < v_2, ..., v_n >$, $Pu = u$, ya que justamente la propiedad anterior se cumple para la base. Luego, 1 es un eigenvalor de P y además,

$v_2, ..., v_n$ son eigenvectores asociados al eigenvalor 1. Nótese ahora que

$$Pv = I_n v - 2vv^* v = v - 2v(v^* v) = v - 2v(1) = -v,$$

ya que v es ortonormal y por lo tanto $v^* v = \|v\|^2 = 1$. Luego, para cualquier $u \in < v >$, se tiene que $Pu = -u$ por la misma propiedad universal de las bases. Concluimos que -1 también es un eigenvalor de P y que v se puede considerar un eigenvector asociado a dicho valor.

Denotando por $v_1 := v$, tenemos que 1 y -1 son los eigenvalores de una matriz de Householder asociados a los vectores $v_1, ..., v_n$ que forman una base de R^n. ■

4. Demuestra que no es posible construir la transformación de similaridad del Teorema de Schur con un número finito de transformaciones de similaridad de Householder.

Solución: el teorema de Schur nos dice que si tenemos una matriz $A \in M_n(\mathbb{C})$, entonces existen Q matriz unitaria y U una matriz triangular superior tales que

$$A = Q^* U Q,$$

y además, las entradas de la diagonal de U son los eigenvalores de la matriz A. Por otro lado, sabemos que si $n \geq 5$, entonces

$$det(A - \lambda I_n),$$

el polinomio característico de A que tiene por raíces a los eigenvalores de la matriz A no tiene una fórmula para resolverse en términos de operaciones que involucren sumas, multiplicaciones y potencias, lo cual se deduce del célebre teorema de Abel-Ruffini.

Finalmente, las transformaciones de similaridad de Householder son transformaciones lineales, las cuales están descritas justamente por las matrices de Householder. Por lo tanto, si en un número finito de transformaciones de Householder se pudiera obtener la descomposición del teorema de Schur, entonces tendríamos una fórmula para solución de la ecuación de grado 5 o superior, lo cual es imposible.

Dicho de manera más precisa, si $P_1, ..., P_m$ son una sucesión finita de transformaciones de Householder codificadas en las m matrices (de Householder) anteriores, de tal forma que

$$P_m \cdot P_{m-1} \cdot ... \cdot P_1 A = Q^* U^* Q,$$

donde $Q^* U Q$ es la descomposición de A del teorema de Schur, entonces se tendría una fórmula para hallar las soluciones del polinomio característico, de grado mayor o igual a 5, de A. A saber, la fórmula estaría dada por

$$\lambda_k = U_{kk} = (QP_m \cdot P_{m-1} \cdot ... \cdot P_1 A Q^*) = \sum_{k,j,i_1...,i_{m-1},i_m,l,k}^{n} Q_{kj} P_{m,ki_1} P_{m-1,i_1 i_2} \cdot ... \cdot P_{1,i_{m-1} i_m} A_{i_m l} (Q^*)_{lk},$$

pero esto contradice lo que hemos dicho. La afirmación del ejercicio se sigue.

5. ¿Qué pasa si aplicas la iteración QR sin shift a una matriz ortogonal?

Solución: sabemos que la descomposición QR es única salvo cambios de signo (ver Trefethen, Bau), por lo que si una matriz A es ortogonal, entonces podemos verla como $A = A \cdot I_n = Q \cdot R$, donde $A = Q$ y $I_n = R$. Luego, al aplicar el algoritmo QR a la matriz A ortogonal, entonces $A = QR = A \cdot I_n \implies RQ = I_n \dot{A} = A$, de manera que la matriz se queda inalterada. Esto se puede apreciar computacionalmente en la figura 3. El código está al final del script *Ejercicio 2-5*. La implementación es exactamente la misma que la hecha en el ejercicio 2 salvo que no se utiliza un shift en el algoritmo.

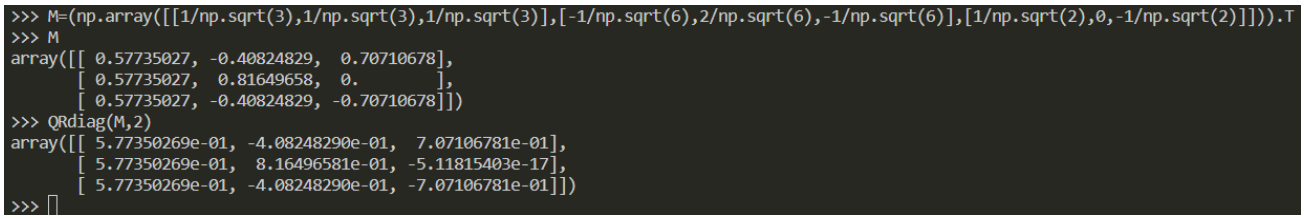

Figura 3: Corroboración de que una matriz ortogonal queda inalterada bajo el algoritmo QR.