Universidad de Valladolid

FACULTAD DE CIENCIAS

TRABAJO FIN DE GRADO

Grado en Matemáticas

Métodos de Krylov para el problema de autovalores de matrices dispersas.

Autora: Paula Heras Ballesteros

Tutor: Luis María Abia Llera

2024/25

Resumen

El estudio de los problemas de autovalores y autovectores de una matriz es fundamental en el ámbito del álgebra lineal numérica, especialmente en el contexto donde las matrices involucradas son de gran tamaño o presentan una estructura dispersa. El presente trabajo se centra en el análisis de los principales métodos numéricos empleados para abordar estos problemas. Inicialmente se revisarán conceptos clave como subespacios de Krylov y se desarrollarán técnicas clásicas como el algoritmo QR. Una parte central del trabajo se dedica al proceso de Lanczos, método diseñado para matrices simétricas, destacando su formulación, propiedades, limitaciones y variantes, como la reortogonalización completa. Asimismo, se abordan también métodos diseñados para matrices no simétricas, entre ellos el proceso de Arnoldi y el algoritmo Krylov - Schur.

Palabras clave: Subespacios de Krylov; algoritmo QR caso simétrico y caso no simétrico; proceso de Lanczos y proceso de Arnoldi.

Abstract

The study of eigenvalue and eigenvector problems of a matrix is fundamental in the field of numerical linear algebra, especially in contexts where the matrices involved are large or have a sparse structure. This Degree Thesis focuses on the analysis of the main numerical methods used to address these problems. Initially, key concepts such as Krylov subspaces are reviewed, and classical techniques such as the QR algorithm are developed. A central part of the work is dedicated to the Lanczos process, a method designed for symmetric matrices, highlighting its formulation, properties, limitations, and variants such as full reorthogonalization. Methods designed for non-symmetric matrices are also covered, including the Arnoldi process and the Krylov-Schur algorithm.

Key words: Krylov subspaces; QR algorithm in the symmetric and non-symmetric case; Lanczos process and Arnoldi process.

A mi familia, en especial a mi madre, por apoyarme en cada etapa de mi vida con paciencia y amor, por ayudarme a levantarme cada vez que me he caído y por recordarme que todo llega.

A mi hermano, por enseñarme el significado de la palabra amor.

A mis amigos del instituto y del pueblo, por estar siempre ahí y hacerme sentir en casa.

A mis amigos de la carrera, por acompañarme estos años, por compartir lloros, risas y desayunos y por darle ese toque de humor tan necesario a los días de biblioteca.

A mis amigas del Sicue, por compartir el año más especial de todos.

A mi tutor Luis, por aceptar dirigirme este trabajo.

A mi tío Jose, espero que allí donde estés celebres con una sonrisa cada logro que alcancemos.

A mí misma, por no rendirme y por demostrarme de todo lo que soy capaz.

Índice general

Introducción IX

1. Conceptos básicos y fundamentales 1
- 1.1. Matrices relevantes . . . 2
- 1.2. Autovalores y autovectores. . . . 4
- 1.3. Subespacios de Krylov . . . 7

2. Problema de autovalores 9
- 2.1. Propiedades y descomposiciones . . . 10
  - 2.1.1. Autovalores y autovectores . . . 10
- 2.2. Iteraciones de subespacios . . . 12
  - 2.2.1. El método de la potencia . . . 13
- 2.3. Algoritmo QR: caso simétrico . . . 14
  - 2.3.1. Propiedades de la descomposición tridiagonal . . . 15
  - 2.3.2. Versión implícita del algoritmo QR con desplazamiento (caso simétrico) . . . 17
- 2.4. Algoritmo QR: caso no simétrico . . . 23
  - 2.4.1. Deflación . . . 23
  - 2.4.2. Propiedades de las matrices de Hessenberg . . . 24
  - 2.4.3. Estrategia de doble desplazamiento implícito: el paso QR de Francis . . . 28
  - 2.4.4. Procedimiento general . . . 30

3. Problemas de autovalores en matrices grandes y dispersas 33
- 3.1. El proceso de Lanczos simétrico . . . 33
  - 3.1.1. Subespacios de Krylov . . . 34
  - 3.1.2. Tridiagonalización . . . 36

3.1.3. Terminación y cotas de error 39
3.1.4. Aproximaciones de Ritz 41
3.1.5. Teoría de la convergencia 45
3.2. Procedimientos prácticos de Lanczos 46
3.2.1. Almacenamiento y trabajo requeridos 47
3.2.2. Propiedades de redondeo 47
3.2.3. Lanczos con reortogonalización completa 49

4. Métodos para problemas no simétricos **53**
4.1. El proceso de Arnoldi básico 54
4.2. Arnoldi con rearranque 56
4.3. Rearranque implícito 57
4.4. El algoritmo de Krylov - Schur 60
4.5. Tridiagonalización de Lanczos no simétrica 61
4.6. La idea de mirar hacia adelante 64

Introducción

La presente memoria tiene por objetivo el estudio técnico de los principales algoritmos numéricos para la aproximación de los autovalores/autovectores de grandes matrices dispersas. En los estudios del Grado de Matemáticas se estudian los métodos básicos para el problema de autovalores de matrices llenas, que culminan con la presentación del algoritmo QR para matrices generales.

En las aplicaciones, es común el interés en aproximar unos pocos autovalores de grandes matrices dispersas que provienen de la discretización de problemas de autovalores de ecuaciones en derivadas parciales. El gran tamaño de estas matrices impide un tratamiento numérico del problema basado en transformaciones ortogonales de la matriz del problema, dado que estas transformaciones destruyen el carácter disperso de las mismas. Es por ello de interés el uso de métodos que se basen exclusivamente en el cálculo del producto matriz por vector, que puede hacerse muy eficiente si se explotan los elementos cero de la matriz.

La idea básica para el problema de autovalores de una matriz simétrica es el llamado proceso de tridiagonalización de Lanczos, que se generaliza al algoritmo de Arnoldi cuando la matriz A es general. Esta memoria culmina con la introducción de estos procesos y el estudio (no completo) de algunas de las técnicas que deben observarse en su implementación.

Los capítulos 1 y 2 recogen algunos resultados generales sobre el problema de autovalores. En particular, presentamos la teoría minimax de Courant-Fischer para la caracterización de los autovalores de una matriz simétrica, que servirá posteriormente para motivar el proceso simétrico de Lanczos. El capítulo también presenta el algoritmo QR, tanto el caso simétrico como no simétrico, como punto de partida para los algoritmos de los capítulos 3 y 4.

En el capítulo 3 se introduce de forma motivada el proceso de Lanczos simétrico para matrices simétricas. La importancia del caso simétrico es la disponibilidad de una teoría de convergencia, debida a Kaniel, Saad y Page, que no desarrollamos por una complejidad técnica pero que enfatizamos dando los resultados principales. Terminamos el capítulo con la consideración de algunos detalles de interés para la implementación práctica.

Por último, el capítulo 4 se centra en el algoritmo de Arnoldi, fundamento del software ARPACK para el problema de autovalores para matrices no simétricas, y que se implementa por ejemplo en la función `eigs` de Matlab. La principal técnica del software ARPACK es el rearranque del proceso de Arnoldi cuando los vectores generados van perdiendo su ortgonalidad.

Todos los resultados presentados se basan en Golub - Van Loan ([2]) y en ARPACK ([3]), aunque se han revisado otras referencias como Saad ([9]) o Van der Vorst ([10]). El área de investigación es demasiado amplio para los objetivos de esta memoria y nos hemos conformado con una selección de los resultados más importantes.

Capítulo 1

Conceptos básicos y fundamentales

En este capítulo se presentan los conceptos esenciales necesarios para abordar el estudio de los métodos de espacios de Krylov aplicados a la aproximación de autovalores de matrices. Estos fundamentos teóricos y matemáticos forman la base conceptual sobre la que se desarrolla este trabajo de fin de grado.

Dado que los métodos de espacios de Krylov se sitúan en la intersección del álgebra lineal numérica y la resolución eficiente de problemas de valores propios, es crucial establecer un marco teórico claro. Por ello, comenzaremos revisando nociones básicas de álgebra lineal, como autovalores, autovectores y sus propiedades asociadas, ya estudiadas en el Grado, con el objetivo de ir fijando las notaciones que utilizaremos. Además, introduciremos el concepto de subespacios de Krylov, que constituye el núcleo de los métodos tratados en este estudio.

El capítulo también incluirá una discusión sobre las propiedades de las matrices, tales como simetría, diagonalización y factorización, que son especialmente relevantes para el desarrollo de los algoritmos basados en espacios de Krylov. Asimismo, se abordarán aspectos teóricos como la convergencia de las aproximaciones y la importancia de la estructura espectral de una matriz en el diseño y análisis de los métodos numéricos.

Al proporcionar esta base teórica, se busca dotar al lector de las herra-

mientas necesarias para comprender en profundidad los métodos desarrollados en los capítulos posteriores, estableciendo así un puente entre los principios fundamentales y su aplicación en la resolución de problemas prácticos.

1.1. Matrices relevantes

Definición 1.1. Sea $A \in \mathbb{R}^{m\times n}$. Se dice que A es una *matriz dispersa* si la mayoría de sus elementos son cero, es decir, si la proporción de elementos no nulos es significativamente menor que el número total de elementos. Usualmente se asume que el número de elementos no nulos de A es $\mathcal{O}(n)$ ó $\mathcal{O}(m)$.

Definición 1.2. Se dice que $Q \in \mathbb{R}^{m\times n}$ es una *matriz ortogonal* cuando

$$Q^TQ = I_n, \quad QQ^T = I_m$$

siendo I_n e I_m las matrices identidad de dimensiones $n \times n$ y $m \times m$ respectivamente.

Si $Q = [\ q_1 \mid \cdots \mid q_n\ ]$ es una matriz ortogonal $m \times n$, con $m > n$, entonces las columnas q_i forman un sistema ortonormal de vectores que siempre es posible extender a una base ortonormal completa $\{q_1, \dots, q_n, q_{n+1}, \dots, q_m\}$ de $\mathbb{R}^m$.

Definición 1.3. Una matriz $A \in \mathbb{R}^{n\times n}$ es *simétrica* si

$$A^T = A.$$

Son matrices con propiedades espectrales especiales como que todos sus autovalores son reales y los correspondientes autovectores son ortogonales y forman una base de $\mathbb{R}^n$.
Nótese que si $A \in \mathbb{R}^{n\times n}$ es una matriz simétrica, entonces

$$\lambda_n(A) \leq \cdots \leq \lambda_1(A)$$

denota el conjunto de autovalores de A ordenado de manera creciente e incluyendo un autovalor tantas veces como su multiplicidad.

Definición 1.4. Sea $\tilde{\mathbf{v}} \in \mathbb{R}^n$ un vector no nulo. Se denomina *matriz de Householder* a la matriz $P \in \mathbb{R}^{n\times n}$ de la forma

$$P = I - \beta \tilde{\mathbf{v}}\tilde{\mathbf{v}}^T, \quad \text{con } \beta = \frac{2}{v^T v}.$$

También se conoce a esta matriz P como *reflector de Householder* o *transformación de Householder*.

Definición 1.5. Dados un par de índices i, k, con $1 \leq i < k \leq n$ y un ángulo $\theta \in \mathbb{R}$, se define la *matriz de Givens* como

$$G(i,k,\theta) = \begin{bmatrix} 1 & \cdots & 0 & \cdots & 0 & \cdots & 0 \\ \vdots & \ddots & \vdots & & \vdots & & \vdots \\ 0 & \cdots & c & \cdots & s & \cdots & 0 \\ \vdots & & \vdots & \ddots & \vdots & & \vdots \\ 0 & \cdots & -s & \cdots & c & \cdots & 0 \\ \vdots & & \vdots & & \vdots & \ddots & \vdots \\ 0 & \cdots & 0 & \cdots & 0 & \cdots & 1 \end{bmatrix},$$

donde $c = \cos(\theta)$ y $s = \sin(\theta)$.

Una *rotación de Givens* es una perturbación de rango 2 de la matriz identidad, salvo que $\theta = 0$, en cuyo caso la rotación de Givens coincide con la identidad.

Estas transformaciones son ortogonales

Definición 1.6. Se dice que una matriz $H \in \mathbb{R}^{n\times n}$ tiene *forma de Hessenberg superior* si $h_{ij} = 0$ siempre que $i > j + 1$.

NOTACIÓN: Una forma de especificar una fila o columna de una matriz es con la notación de "punto y coma" como se muestra a continuación:
Sea $A \in \mathbb{R}^{m\times n}$, entonces $A(k,:)$ y $A(:,k)$ designan respectivamente la k-ésima fila y columna de la matriz A, es decir,

$$A(k,:) = [\, a_{k1}, ..., a_{kn} \,] \quad \text{y } A(:,k) = \begin{bmatrix} a_{1k} \\ \vdots \\ a_{mk} \end{bmatrix}.$$

1.2. Autovalores y autovectores.

Definición 1.7. Sea $A \in \mathbb{R}^{n\times n}$. Los *autovalores de A* son los ceros del *polinomio característico*

$$p(x) = \det(A - xI).$$

Al conjunto de todos los autovalores λ de A se le denomina *espectro de A* y se le denota por

$$\lambda(A) = \{\lambda \in \mathbb{R} \mid \det(A - \lambda I) = 0\}.$$

Dado $\lambda \in \lambda(A)$, a los vectores no nulos $\tilde{\mathbf{x}}$ tales que $A\tilde{\mathbf{x}} = \lambda\tilde{\mathbf{x}}$ se les denomina *autovectores* de A asociados a λ.

Definición 1.8. Si $A \in \mathbb{R}^{n\times n}$ tiene n autovectores independientes $x_1, ..., x_n$ y $Ax_i = \lambda_i x_i$, para cada $i \in \{1, ..., n\}$, entonces se dice que A es *diagonalizable*. En forma matricial,

$$X = [\ x_1 \mid \cdots \mid x_n\ ],$$

es invertible, y además

$$X^{-1}AX = \operatorname{diag}(\lambda_1, ..., \lambda_n).$$

En el caso general, podemos esperar en el mejor de los casos una reducción a una forma tridiagonal superior utilizando transformaciones unitarias. Es lo que llamamos la *descomposición de Schur* de una matriz.
Si $Q = [\ q_1 \mid \cdots \mid q_n\ ]$ en la descomposición de Schur, a los vectores q_i, con $i = 1, ..., n$, se les denominan *vectores de Schur*.

A continuación enunciamos un lema necesario para la prueba de resultados posteriores.

NOTACIÓN:

- $\operatorname{ran}(Q_k) = \operatorname{span}\{q_1, ..., q_n\}$ denota el subespacio generado por las columnas de Q_k.
- $\operatorname{rank}(Q_k) = \operatorname{rg}(Q_k)$.

Lema 1.9. *Sean $A \in \mathbb{C}^{n\times n}, B \in \mathbb{C}^{p\times p}$ y $X \in \mathbb{C}^{n\times p}$ tales que*

$$AX = XB, \quad \textit{rank}(X) = p.$$

Entonces, existe una matriz unitaria $Q \in \mathbb{C}^{n\times n}$ tal que

$$Q^H AQ = T = \begin{array}{c} \left[\begin{array}{c|c} T_{11} & 0 \\ \hline 0 & A_1 \end{array}\right] \begin{array}{l} p \\ n-p \end{array} \\ \begin{array}{cc} p & n-p \end{array} \end{array}$$

y $\lambda(T_{11}) = \lambda(A) \cap \lambda(B)$.

Teorema 1.10 (**Descomposición de Schur**). *Sea $A \in \mathbb{C}^{n\times n}$, entonces existe una matriz unitaria $Q \in \mathbb{C}^{n\times n}$ tal que*

$$Q^H AQ = T = D + N,$$

donde $D = diag(\lambda_1, ..., \lambda_n)$ y $N \in \mathbb{C}^{n\times n}$ es estrictamente triangular superior. Además, Q puede elegirse de manera que los autovalores λ_i aparezcan en cualquier orden a lo largo de la diagonal.

Demostración. Razonamos por inducción sobre n.

- Para n = 1, tenemos que $A \in \mathbb{C}^{1\times 1} \Rightarrow A = \lambda$.
  De este modo, cualquier matriz unitaria $Q \in \mathbb{C}^{1\times 1}$, es decir, cualquier número complejo de valor absoluto 1, verifica que

$$Q^H AQ = \lambda$$

- Lo suponemos cierto para $n-1$, es decir, para $A \in \mathbb{C}^{(n-1)\times(n-1)}$ existe una matriz unitaria $Q \in \mathbb{C}^{(n-1)\times(n-1)}$ tal que

$$Q^H AQ = T, \quad \text{con } T \text{ triangular superior.} \tag{H.I}$$

- Lo probamos para n. Como $A \in \mathbb{C}^{n\times n}$, por el Teorema Fundamental del Álgebra sabemos que A tiene al menos un autovalor λ y existe un autovector x, con $x \neq 0$, tal que $Ax = \lambda x$.
  Aplicando el Lema 1.9 con $B = (\lambda)$, podemos garantizar que existe una matriz unitaria $U \in \mathbb{C}^{n\times n}$ tal que

$$U^H AU = \begin{array}{c} \left[\begin{array}{c|c} \lambda & w^H \\ \hline 0 & C \end{array}\right] \begin{array}{l} 1 \\ n-1 \end{array} \\ \begin{array}{cc} 1 & n-1 \end{array} \end{array} \tag{1.1}$$

es una matriz triangular superior.
Como $C \in \mathbb{C}^{(n-1)\times(n-1)}$, por (H.I) sabemos que existe una matriz unitaria $\tilde{U} \in \mathbb{C}^{(n-1)\times(n-1)}$ tal que $\tilde{U}^T C \tilde{U}$ es una matriz triangular superior.
Construyamos ahora, a partir de U y $\tilde{U}$, la matriz unitaria Q.
Definamos Q como

$$Q = U \cdot \text{diag}(1, \tilde{U}),$$

donde $\text{diag}(1, \tilde{U})$ es la matriz diagonal por bloques que tiene en el primer bloque un 1 y en el segundo bloque a $\tilde{U}$.
Observamos que Q es unitaria por ser producto de matrices unitarias.
Finalmente, veamos que $Q^H A Q$ es triangular superior.

$$\begin{aligned} Q^H A Q &= \left(U \cdot \text{diag}(1, \tilde{U})\right)^H A \left(U \cdot \text{diag}(1, \tilde{U})\right) = \\ &= \text{diag}(1, \tilde{U})^H \left(U^H A U\right) \text{diag}(1, \tilde{U}). \end{aligned}$$

Entonces, teniendo en cuenta (1.1)

$$Q^H A Q = \left[\begin{array}{c|c} 1 & 0 \\ \hline 0 & \tilde{U}^H \end{array}\right] \cdot \left[\begin{array}{c|c} \lambda & w^H \\ \hline 0 & C \end{array}\right] \cdot \left[\begin{array}{c|c} 1 & 0 \\ \hline 0 & \tilde{U} \end{array}\right] = \left[\begin{array}{c|c} \lambda & w^H \tilde{U} \\ \hline 0 & \tilde{U}^H C \tilde{U} \end{array}\right],$$

y hemos visto ya que $\tilde{U}^H C \tilde{U}$ es una matriz triangular superior, luego $Q^H A Q = T$ triangular superior con $Q \in \mathbb{C}^{n\times n}$.

□

Corolario 1.11 (**Descomposición de Schur Simétrica**). *Si $A \in \mathbb{R}^{n\times n}$ es simétrica, entonces existe un matriz ortogonal real Q tal que*

$$Q^T A Q = \Lambda = diag(\lambda_1, ..., \lambda_n).$$

Demostración. Sean $\lambda_1 \in \lambda(A)$ y $x \in \mathbb{C}^n$ el autovector unitario asociado a λ_1, es decir, $Ax = \lambda_1 x$, con $\|x\|_2 = 1$.
Sabemos que $\lambda_1 = x^H A x$, siendo x^H el vector traspuesto conjugado de x.
Además, como A es simétrica (y real) $A = A^T$, luego

$$\lambda_1 = x^H A x = x^T A x,$$

lo que nos garantiza que $\lambda_1 \in \mathbb{R}$, y podemos asumir que $x \in \mathbb{R}^n$.
Sea $P_1 \in \mathbb{R}^{n\times n}$ la matriz de Householder tal que $P_1^T x = e_1 = I_n(:, 1)$.
Entonces, como $Ax = \lambda_1 x$, se tiene que

$$(P_1^T A P_1)(P_1^T x) = \lambda_1 (P_1^T x) \Longrightarrow (P_1^T A P_1) e_1 = \lambda_1 e_1$$

Esto lo que nos dice es que la primera columna de la matriz $P_1^T A P_1$ es múltiplo de e_1.
Como A es simétrica y P_1 es ortogonal, $P_1^T AP$ es simétrica, luego tiene que ser de la forma

$$P_1^T AP = \left[\begin{array}{c|c} \alpha_1 & 0 \\ \hline 0 & A_1 \end{array}\right], \quad \text{con } A_1 \in \mathbb{R}^{(n-1)\times(n-1)}.$$

Además, la matriz A_1 sigue siendo simétrica por ser A simétrica.
Razonando por inducción, podemos asumir que existe una matriz ortogonal $Q \in \mathbb{R}^{(n-1)\times(n-1)}$ tal que $Q_1^T A_1 Q_1 = \Lambda_1$ es diagonal.
Por lo tanto, al combinarlo con lo anterior se tiene que

$$Q^{-1}AQ = \Lambda, \quad \text{siendo } Q = \begin{bmatrix} 1 & 0 \\ 0 & Q_1 \end{bmatrix} \text{ y } \Lambda = \begin{bmatrix} \lambda_1 & 0 \\ 0 & \Lambda_1 \end{bmatrix}$$

□

La simetría nos garantiza que todos los autovalores de una matriz A son reales y que existe una base ortonormal de autovectores reales.

1.3. Subespacios de Krylov

Definición 1.12. Dada una matriz $A \in \mathbb{R}^{n\times n}$ y un vector $v \in \mathbb{R}^n$, se llama *espacio de Krylov* de dimensión k al subespacio generado por los primeros k vectores generados por la sucesión $v, Av, A^2v, ...$, y se denota por

$$\mathcal{K}(A, v, k) = \text{span}\{v, A^2v, \ ..., A^{k-1}v\}.$$

Propiedades 1.13. *Veamos a continuación algunas propiedades importantes de los subespacios de Krylov:*

- *Dado que la matriz A es cuadrada de orden n, la dimensión del espacio de Krylov no puede superar n, es decir,* $\dim(\mathcal{K}(A, v, k)) \leq n$.
- *Si los vectores que se generan son linealmente dependientes, el espacio de Krylov se estabiliza, es decir, $\mathcal{K}(A, v, k) = \mathcal{K}(A, v, k+1)$, para algún $k \leq n$.*

- *El menor entero k para el cual los vectores generados se vuelven linealmente dependientes es* $\deg(p_A(v)) = k$*, siendo* $p_A(v)$ *el polinomio mínimo de* v*.*

Los subespacios de Krylov no son más que los espacios generados por las columnas de matrices de Krylov, que se definen a continuación.

Definición 1.14. Dada una matriz $A \in \mathbb{R}^{n\times n}$ y un vector $v \in \mathbb{R}^n$, se denominan *matrices de Krylov* a las matrices de la forma

$$K(A, v, k) = \left[\ v \mid Av \mid \cdots \mid A^{k-1}v\ \right] \in \mathbb{R}^{n\times k}.$$

Propiedades 1.15. *Si* $k \leq n$ *y los vectores que generan el subespacio de Krylov son linealmente independientes, entonces* $ran(K(A, v, k)) = k$*. Si por el contrario* $k > n$*, las columnas de la matriz son linealmente dependientes.*

Capítulo 2

Problema de autovalores

El problema de autovalores, en particular el caso simétrico, es uno de los problemas computacionales más tratables y elegantes en álgebra lineal numérica. En este capítulo se abordan distintos métodos y algoritmos destinados a su resolución. Comenzamos con una breve discusión de las propiedades matemáticas que subyacen a los algoritmos que siguen.

En las secciones 2.1 y 2.2 se introducen las propiedades fundamentales junto con métodos iterativos clásicos como el método de la potencia. A partir de estas bases, nos centramos en el algoritmo QR, distinguiendo entre el caso simétrico y el no simétrico.

La sección 2.3 está dedicada al caso simétrico del algoritmo QR. Se profundiza en las propiedades de la descomposición tridiagonal, apoyándose en resultados teóricos clave, y se presentará también una versión implícita del algoritmo QR con desplazamiento, que permite mejorar la eficiencia numérica en matrices simétricas.

Por último, la sección 2.4 aborda el caso no simétrico del algoritmo QR, donde el problema es más complejo y se requieren herramientas más específicas. Se introducen conceptos como la deflación y la reducción a forma de Hessenberg, y se desarrolla de forma detallada la estrategia de doble desplazamiento implícito mediante el paso QR de Francis, finalizando con la exposición del procedimiento general para obtener la forma de Schur real.

2.1. Propiedades y descomposiciones

En esta sección resumimos las matemáticas necesarias para descubrir y analizar algoritmos para el problema de autovalores simétricos.

2.1.1. Autovalores y autovectores

Definición 2.1. Sean $A \in \mathbb{R}^{n\times n}$ una matriz simétrica y $\tilde{\mathbf{x}} \in \mathbb{R}^n$ un vector no nulo. Se denomina *cociente de Rayleigh* de x al escalar

$$r(x) = \frac{x^T Ax}{x^T x}.$$

Propiedades 2.2. *Veamos a continuación algunas propiedades del cociente de Rayleigh:*

- *Dado q_i autovector de A asociado a λ_i, se tiene que*

$$r(q_i) = \frac{q_i^T Aq_i}{q_i^T q_i} = \lambda_i.$$

- *Para cualquier otro vector y el valor del cociente de Rayleigh está acotado por los autovalores de A. Más en concreto se tiene que*

$$\lambda_n(A) \leq r(y) \leq \lambda_1(A),$$

siendo respectivamente λ_n y λ_1 el menor y el mayor autovalor de A, puesto que A es simétrica.

El mayor y el menor autovalor de una matriz simétrica satisfacen

$$\lambda_{\max} = \max_{x\neq 0} \frac{x^T Ax}{x^T x} \tag{2.1}$$

y

$$\lambda_{\min} = \min_{x\neq 0} \frac{x^T Ax}{x^T x}. \tag{2.2}$$

El teorema siguiente nos dice que el k-ésimo autovalor de A se obtiene tomando el máximo de los mínimos cocientes de Rayleigh sobre todos los subespacios de dimensión k.

Teorema 2.3 (**Teorema Minimax de Courant - Fischer**). *Si $A \in \mathbb{R}^{n\times n}$ es simétrica, entonces*

$$\lambda_k = \max_{dim(S)=k} \min_{0\neq y\in S} \frac{y^T Ay}{y^T y}, \quad para\ i \in \{1, ..., n\}.$$

Demostración. Sea $A \in \mathbb{R}^{n\times n}$ simétrica.
Hemos visto en el Corolario **Descomposición de Schur Simétrica** que existe una matriz ortogonal $Q = [\ q_1\ |\cdots|\ q_n\ ]$ tal que $\Lambda = Q^T AQ = \text{diag}(\lambda_1, ..., \lambda_n)$ es la descomposición de Schur de la matriz A, siendo $\lambda_n(A) \leq \cdots \leq \lambda_1(A)$ los autovalores de A.
Definamos el subespacio invariante asociado a $\lambda_1, ..., \lambda_k$ como sigue:

$$S_k = \text{span}\{q_1, ..., q_k\}.$$

Veamos en primer lugar que

$$\min_{0\neq y\in S_k} \frac{y^T Ay}{y^T y} = \lambda_k.$$

Tomamos $y \in S_k$, por lo tanto $y = \alpha_1 q_1 + \cdots + \alpha_k q_k$, con $\alpha_i \in \mathbb{R}, \forall i = 1 : n$. Entonces,

$$\frac{y^T Ay}{y^T y} = \frac{\left(\sum_{i=1}^k \alpha_i q_i\right)^T A \left(\sum_{i=1}^k \alpha_i q_i\right)}{\left(\sum_{i=1}^k \alpha_i q_i\right)^T \left(\sum_{i=1}^k \alpha_i q_i\right)}.$$

Recordemos que, $\forall i = 1 : n, q_i$ es el autovector de A asociado al autovalor λ_i, luego $Aq_i = \lambda_i q_i$, y además q_i es ortonormal, luego se tiene que

$$q_i^T q_j = \delta_{ij} = \begin{cases} 1, & \text{si } i = j, \\ 0, & \text{si } i \neq j. \end{cases}$$

Esto implica que el cociente de Rayleigh del vector y alcanza su mínimo en $y = q_k$, y en este caso se tiene que

$$\min_{0\neq y\in S} \frac{y^T Ay}{y^T y} = \frac{q_k^T Aq_k}{q_k^T q_k} = q_k^T Aq_k = \lambda_k,$$

como queríamos probar.

Veamos ahora que

$$\max_{\dim(S)=k} \min_{0\neq y\in S} \frac{y^T Ay}{y^T y} \geq \lambda_k.$$

Sea $S = S_k$, luego $\dim(S) = k$, entonces

$$\max_{\dim(S)=k} \min_{0\neq y\in S} \frac{y^T Ay}{y^T y} = \min_{0\neq y\in S} \frac{y^T Ay}{y^T y} \geq \lambda_k.$$

Finalmente, veamos que

$$\max_{\dim(S)=k} \min_{0\neq y\in S} \frac{y^T Ay}{y^t y} \leq \lambda_k.$$

Sea S un subespacio arbitrario tal que $\dim(S) = k$. Notemos que S debe tener intersección no trivial con $\text{span}\{q_k, ..., q_n\}$, subespacio de dimensión $n - k + 1$. Sea $y_* = \alpha_k q_k + \cdots + \alpha_n q_n$ en la intersección. Para y_* se tiene que

$$\frac{y_*^T Ay_*}{y_*^T y_*} \leq \lambda_k,$$

puesto que y_* tiene componentes en $q_k, q_{k+1}, ..., q_n$, y λ_k es el mayor autovalor de A asociado a dichos autovectores. Luego,

$$\min_{0\neq y\in S} \frac{y^T Ay}{y^T y} \leq \lambda_k.$$

Como lo anterior es cierto para todo S subespacio con $\dim(S) = k$,

$$\max_{\dim(S)=k} \min_{0\neq y\in S} \frac{y^T Ay}{y^T y} \leq \lambda_k.$$

Con esto se completa la demostración del teorema. □

2.2. Iteraciones de subespacios

Asumamos que $A \in \mathbb{R}^{n\times n}$ es simétrica y que $U_0 \in \mathbb{R}^{n\times n}$ es ortogonal. Consideramos la siguiente iteración QR:

$$
\begin{aligned}
&T_0 = U_0^T A U_0 \\
&\textbf{for } k = 1, 2, ... \\
&\quad T_{k-1} = U_k R_k \quad \text{(factorización QR)} \\
&\quad T_k = R_k U_k \\
&\textbf{end}
\end{aligned}
\tag{2.3}
$$

Dado que $T_k = R_k U_k = U_k^T (U_k R_k) U_k = U_k^T T_{k-1} U_k$ por inducción se tiene que

$$T_k = (U_0 U_1 \cdots U_k)^T A (U_0 U_1 \cdots U_k). \tag{2.4}$$

Por lo tanto, para cada k se tiene que T_k es ortogonalmente semejante a A. Además, T_k casi siempre converge a una forma diagonal, y por ello se puede decir que (2.4) casi siempre converge a una descomposición de Schur de A. Para establecer este resultado, consideramos primero el método de la potencia y el método de la iteración ortogonal.

2.2.1. El método de la potencia

Dado $q^{(0)} \in \mathbb{R}^n$, con $\|q^{(0)}\|_2 = 1$, el *método de la potencia* produce una secuencia de vectores $q^{(k)}$ como se sigue:

$$
\begin{aligned}
&\textbf{for } k = 1, 2, ... \\
&\quad z^{(k)} = A q^{(k-1)} \\
&\quad q^{(k)} = z^{(k)} / \|z^{(k)}\|_2 \\
&\quad \lambda^{(k)} = [q^{(k)}]^T A q^{(k)} \\
&\textbf{end}
\end{aligned}
\tag{2.5}
$$

Si $q^{(0)}$ no es "deficiente" y el autovalor de A de módulo máximo es único, entonces $q^{(k)}$ converge a un autovector.

Teorema 2.4. *Sea $A \in \mathbb{R}^{n \times n}$ una matriz simétrica, y sea*

$$Q^T A Q = diag(\lambda_1, ..., \lambda_n)$$

la descomposición de Schur de A, siendo $Q = [\, q_1 \mid \cdots \mid q_n \,]$ una matriz ortogonal y $|\lambda_1| > |\lambda_2| \geq \cdots \geq |\lambda_n|$. Sean $q^{(k)}$ los vectores especificados en (2.5)

y definamos $\theta_k \in [0, \pi/2]$ *como*

$$\cos(\theta_k) = |q_1^T q^{(k)}|.$$

Si $\cos(\theta_k) \neq 0$*, entonces para* $k = 0, 1, ...$ *tenemos que*

$$|\sin(\theta_k)| \leq \tan(\theta_0) \left|\frac{\lambda_2}{\lambda_1}\right|^k,$$

$$|\lambda^{(k)} - \lambda_1| \leq \max_{2 \leq i \leq n} |\lambda_1 - \lambda_i| \tan(\theta_0)^2 \left|\frac{\lambda_2}{\lambda_1}\right|^{2k}.$$

Demostración. Véase Golub - Van Loan ([2]) para ver en detalle la demostración. □

2.3. Algoritmo QR: caso simétrico

Sea $A \in \mathbb{R}^{n \times n}$ una matriz simétrica previamente reducida mediante transformaciones ortogonales de Householder a una forma tridiagonal T_0, y sea $U_0 \in \mathbb{R}^{n \times n}$ una matriz ortogonal. Presentamos a continuación una iteración QR:

$T_0 = U_0^T A U_0$
for $k = 1, 2, ...$
  $T_{k-1} = U_k R_k$ (Factorización QR)
  $T_k = R_k U_k$
end

Analicemos la iteración anterior.
Partimos de una matriz simétrica A y construimos mediante transformaciones de Householder una matriz ortogonal U_0 tal que $T_0 = U_0^T A U_0$ es tridiagonal. La matriz $T_0 \in \mathbb{R}^{n \times n}$ es simétrica por serlo A y conserva los autovalores de A, es decir, es semejante a A.
En el bucle, definimos una forma de la iteración QR.
Primero, factorizamos T_{k-1} en un producto de una matriz ortogonal U_k y una triangular superior R_k para, en el paso siguiente, invertir el orden de dichas matrices y definir T_k (factorización QR).

Veamos que esta última matriz T_k es semejante a A.
Para cada k, $T_k = R_k U_k$. Como $T_{k-1} = U_k R_k \Rightarrow U_k^T T_{k-1} = U_k^T U_k R_k = R_k$, luego, $T_k = R_k U_k = U_k^T T_{k-1} U_k$. Entonces, como $T_{k-1} = U_{k-1} R_{k-1}$, razonando por inducción se tiene que

$$T_k = (U_0 U_1 \cdots U_k)^T A (U_0 U_1 \cdots U_k).$$

Por lo tango, T_k es semejante a A.
Observamos que cada matriz T_k generada en la iteración es más diagonal que la anterior, y en muchos casos converge a una forma diagonal, es decir, a una forma de Schur.
Podemos entonces concluir que esta iteración QR converge casi siempre a una descomposición de Schur de A.

2.3.1. Propiedades de la descomposición tridiagonal

Vamos a probar a continuación dos teoremas sobre la descomposición tridiagonal. El primer resultado conecta la reducción de una matriz simétrica dada a una forma tridiagonal con la factorización QR de una cierta matriz de Krylov. El segundo de ellos, nos muestra que la matriz ortogonal Q necesaria para dicha tridiagonalización es única.

Teorema 2.5. *Sean $A \in \mathbb{R}^{n\times n}$ una matriz simétrica y $Q^T A Q = T$ su descomposición tridiagonal, donde $Q \in \mathbb{R}^{n\times n}$ es una matriz ortogonal.*
Entonces, $Q^T K(A, Q(:,1), n) = R$ es una matriz triangular superior.
Si R es no singular, entonces T es irreducible. Si por el contrario R es singular y k es el menor entero tal que $r_{kk} = 0$, entonces k es también el menor entero tal que $t_{k,k-1} = 0$.

Demostración. Veamos en primer lugar que $Q^T K(A, Q(:,1), n) = R$ es triangular superior.
Sabemos que $Q = [\, q_1 \mid \cdots \mid q_n \,]$ es ortogonal, luego Q^T también es ortogonal, y sabemos que $Q(:,1) = q_1$. Por hipótesis sabemos que $Q^T A Q = T$ tridiagonal, y observamos que para cada $k \in \{1, ..., n-1\}$ se tiene que

$$Q^T A^k q_1 = (Q^T A^k Q)(Q^T q_1) = T^k e_1,$$

puesto que, al ser Q una matriz cuyas columnas son vectores ortonormales, $Q^T q_1 = e_1$. T es una matriz tridiagonal y por lo tanto sus potencias van a

ser matrices con estructura triangular superior. Por lo tanto,

$$\begin{aligned} Q^T K(A, q_1, n) &= \left[\, Q^T q_1 \mid Q^T A q_1 \mid \cdots \mid Q^T A^{n-1} q_1 \,\right] = \\ &= \left[\, Q^T q_1 \mid (Q^T A Q)(Q^T q_1) \mid \cdots \mid (Q^T A^{k-1} Q)(Q^T q_1) \,\right] = \\ &= \left[\, e_1 \mid T e_1 \mid \cdots \mid T^{n-1} e_1 \,\right] = R \end{aligned}$$

es una matriz triangular superior.

Veamos ahora que si R es no singular, entonces T es irreducible, mientras que si R es singular, entonces T es reducible.
Supongamos primero que R es no singular, por lo tanto, R tiene rango máximo, es decir, todos los elementos de la diagonal principal son no nulos. Por construcción, $Q^T K(A, q_1, n) = R \Rightarrow QR = K(A, q_1, n)$, luego el rango de $K(A, q_1, n)$ es máximo y sabemos que la dimensión de $\mathcal{K}(A, q_1, n)$ es máxima, y T conserva toda la información de A, luego T es irreducible.

Ahora, supongamos lo contrario, que R es singular. En este caso R no tiene rango máximo, luego tiene algún elemento de la diagonal nulo. Supongamos que k es el menor entero tal que $r_{kk} = 0$. En este caso, el rango de $K(A, q_1, n)$ no es máximo y en consecuencia la dimensión de $\mathcal{K}(A, q_1, n)$ tampoco es máxima, por lo que no podemos separar T por bloques, y en este caso, $t_{k,k-1} = 0$, luego T es reducible. □

Teorema 2.6 (**Teorema de la Q implícita - caso simétrico -**). *Sea $A \in \mathbb{R}^{n\times n}$ una matriz simétrica, y supongamos que $Q = [\, q_1 \mid \cdots \mid q_n \,]$ y $V = [\, v_1 \mid \cdots \mid v_n \,]$ son matrices ortogonales tales que $Q^T A Q = T$ y $V^T A V = S$, siendo T y S ambas matrices tridiagonales.*
Sea, además, k el menor entero positivo para el cual $t_{k+1,k} = 0$, con la convención de que $k = n$ en el caso en el que T es irreducible.
Entonces, si $v_1 = q_1$, se tiene que $v_i = \pm q_i$ y $|t_{i,i-1}| = |s_{i,i-1}|$, para $2 \le i \le k$. Además, si $k < n$ se tiene que $s_{k+1,k} = 0$.

Demostración. Supongamos que $k = n$, y definamos en primer lugar la matriz ortogonal $W = Q^T V$. Entonces,

$$W^T T W = (Q^T V)^T (Q^T A Q)(Q^T V) = V^T Q Q^T A Q Q^T V = V^T A V = S. \quad (2.6)$$

Por hipótesis, sabemos que tanto Q como V son matrices ortogonales y que $v_1 = q_1$, luego $w_1 = Q^T v_1 = Q^T q_1 = e_1$.

Por el Teorema 2.5 sabemos que $W^T K(T, e_1, k)$ es una matriz triangular superior de rango máximo.
$K(T, e_1, k)$ es una matriz diagonal superior puesto que se construye con potencias sucesivas de la matriz tridiagonal T aplicada a e_1. De este modo, como W es ortogonal, $W^T K(T, e_1, n)$ es triangular superior, y como hemos visto en (2.6), se tiene que

$$W^T K(T, e_1, k) = K(S, e_1, k).$$

Luego, $K(S, e_1, k)$ es una matriz triangular superior y ortogonal. Esto implica que

$$W(:, 1:k) = I_n(:, 1:k) \cdot \text{diag}(\pm 1, ..., \pm 1),$$

es decir, las k primeras columnas de W son vectores canónicos multiplicados por ± 1. En consecuencia, como $W = Q^T V$,

$$Q(:, i) = \pm V(:, i), \quad 1 \leq i \leq k.$$

Analizamos ahora la subdiagonal de las matrices tridiagonales. Para cada i, con $1 \leq i \leq k$ se tiene que

$$t_{i+1,i} = Q(:, i+1)^T A Q(:, i)$$
$$s_{i+1,i} = V(:, i+1)^T A V(:, i)$$

pero como $Q(:, i) = \pm V(:, i)$, entonces,

$$t_{i+1,i} = \pm s_{i+1,i} \Longrightarrow |t_{i+1,i}| = |s_{i+1,i}|$$

□

2.3.2. Versión implícita del algoritmo QR con desplazamiento (caso simétrico)

Sea $A \in \mathbb{R}^{n \times n}$ una matriz simétrica previamente reducida mediante ortogonalizaciones de Householder. Consideramos $Q \in \mathbb{R}^{n \times n}$ tal que $Q^T A Q = T$ tridiagonal y simétrica, por serlo A.

Nos interesamos ahora por mejorar la convergencia del método. Por eso en esta sección abordamos el estudio de la versión implícita del algoritmo QR

con desplazamiento. Para ello, introduzcamos en primer lugar esta versión del algoritmo.

Sea $U_0 \in \mathbb{R}^{n\times n}$ ortogonal.

$T = U_0^T A U_0 \quad$ (tridiagonal)
for $k = 0, 1, ...$
  Elegimos un desplazamiento real μ.
  $T - \mu I = UR \quad$ (factorización QR)
  $T = RU + \mu I$
end

Generalmente, el desplazamiento μ elegido es próximo a un autovalor de T. La matriz T resultante en el bucle anterior es similar a la matriz original $T = U_0 A U_0$ y conserva sus autovalores.

Veamos cuál es el desplazamiento ideal.
Supongamos que

$$T = \begin{bmatrix} a_1 & b_1 & 0 & \cdots & 0 \\ b_1 & a_2 & b_2 & \cdots & 0 \\ 0 & \ddots & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & b_{n-1} \\ 0 & \cdots & 0 & b_{n-1} & a_n \end{bmatrix}.$$

Al desplazamiento más efectivo se conoce como *desplazamiento de Wilkonson*, y viene dado por

$$\mu = a_n + d - \operatorname{sign}(d)\sqrt{d^2 + b_{n-1}^2}, \quad \text{con } d = \frac{a_{n-1} - a_n}{2}.$$

Wilkonson (1968) demostró que la elección de este μ es más estable y práctico que tomar $\mu = a_n$, pese a que ambos convergen cúbicamente.

Nos centramos ahora en la transición de T a $T_+ = RU + \mu I = U^T T U$ pero sin calcular explícitamente la matriz $T - \mu I = UR$. Para ello, vamos a aplicar directamente transformaciones, llevándonos esto a la versión implícita del método.
Queremos aplicar *rotaciones de Givens* a la matriz T.

Definamos $c = \cos(\theta)$ y $s = \sin(\theta)$ tales que

$$\begin{bmatrix} c & s \\ -s & c \end{bmatrix}^T \begin{bmatrix} a_1 - \mu \\ b_1 \end{bmatrix} = \begin{bmatrix} \times \\ 0 \end{bmatrix}$$

Fijamos la matriz de Givens $G_1 = G(1, 2, \theta)$.
Observamos que G_1 actúa igual que el primer bloque de la matriz U, por ello, $Ge_1 = Ue_1$.
Definamos ahora

$$T \leftarrow G_1^T T G_1,$$

matriz tridiagonal que tiene en las posiciones $(1, 3)$ y $(3, 1)$ entradas no nulas que queremos eliminar.
Los autovalores de T no varían ya que se trata de una similitud ortogonal. Con esto lo que estamos consiguiendo es insertar un nuevo elemento no nulo fuera de la banda tridiagonal. Ahora lo que queremos es "mover hacia abajo" este elemento insertado hasta hacerlo desaparecer para restaurar así la tridiagonalidad de la matriz. A esta técnica se le conoce como *"zero - chasing"*. Esto lo vamos a hacer aplicando más rotaciones de Givens.
Cada rotación de Givens que aplicamos es de la forma

$$G_i = G(i, i + 1, \theta), \quad \text{con } i = 2 : (n - 1).$$

El **Teorema de la Q implícita - caso simétrico -** nos garantiza que la matriz resultante de aplicar estas rotaciones, $Z = G_1 \cdot G_2 \cdots G_{n-1}$ cumple que $Ze_1 = G_1 e_1 = Ue_1$ y que $Z^T T Z$ es tridiagonal.
Además, también nos dice que la matriz tridiagonal resultante, $Z^T T Z$, es esencialmente la misma que la matriz tridiagonal T que se obtiene con el método explícito.

Nótese que en cualquier etapa del "zero - chasing" solamente hay una entrada no nula fuera de la tridiagonal. Veámoslo con un ejemplo.

Supongamos que estamos aplicando la k-ésima rotación de Givens a la matriz T como se sigue $T \leftarrow G_k^T T G_k$:

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & c & s & 0 \\ 0 & -s & c & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}^T \begin{bmatrix} a_k & b_k & z_k & 0 \\ b_k & a_p & b_p & 0 \\ z_k & b_p & a_q & b_q \\ 0 & 0 & b_q & a_r \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & c & s & 0 \\ 0 & -s & c & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} a_k & b_k & 0 & 0 \\ b_k & a_p & b_p & z_p \\ 0 & b_p & a_q & b_q \\ 0 & z_p & b_q & a_r \end{bmatrix}$$

Observamos que el valor z_k que estaba fuera de la tridiagonal (posición $(k+1, k+3)$) ha sido desplazado a la posición $(k+2, k+4)$ (renombrado como z_p).
Este proceso se va repitiendo hasta que dicho valor llega al borde de la matriz y se anula por redondeo o convergencia. Los valores c y s se eligen de manera que anulen el valor z_k, resolviendo la ecuación $b_k s + z_k c = 0$. En general, se obtiene:

Algorithm 1 Algoritmo QR simétrico implícito con desplazamiento de Wilkonson

Dada una matriz simétrica, irreducible y diagonal $T \in \mathbb{R}^{n\times n}$, el siguiente algoritmo reescribe T con $Z^T T Z$, donde $Z = G_1 \cdots G_{n-1}$ es producto de rotaciones de Givens con la propiedad de que $Z^T(T-\mu I)$ es una matriz triangular superior, siendo μ el autovalor de la submatriz principal de dimensión 2×2 de T más próximo a t_{nn}.

1: $d = (t_{n-1,n-1} - t_{nn})/2$
2: $\mu = t_{nn} - t_{n,n-1}^2 / \left(d + \text{sign}(d)\sqrt{d^2 + t_{n,n-1}^2}\right)$
3: $x = t_{11} - \mu$
4: $z = t_{21}$
5: **for** $k = 1 : (n-1)$ **do**
6: $\quad [c, s] = \text{givens}(x, z)$
7: $\quad T = G_k^T T G_k$, donde $G_k = G(k, k+1, \theta)$
8: $\quad$ **if** $k < (n-1)$ **then**
9: $\quad\quad x = t_{k+1,k}$
10: $\quad\quad z = t_{k+2,k}$
11: $\quad$ **end if**
12: **end for**

Aplicando el algoritmo QR a $T - \mu I$ en lugar de a T lo que estamos haciendo es acelerando la convergencia de la iteración QR.

Computacionalmente se requieren aproximadamente $30n$ operaciones de coma flotante (flops) y n raíces cuadradas. Además, en el caso en el que la matriz Q se fuera actualizando durante el proceso, se necesitarían $6n^2$ flops adicionales. En la práctica es habitual almacenar la matriz tridiagonal T como dos vectores en lugar de como una matriz completa.

Este algoritmo es la base del algoritmo QR simétrico (medio habitual para el cálculo de la descomposición de Schur de una matriz simétrica densa).

Algorithm 2 Algoritmo QR Simétrico

Dada una matriz simétrica $A \in \mathbb{R}^{n\times n}$ y una tolerancia tol mayor que el error de redondeo de la máquina, este algoritmo calcula una aproximación simétrica de la descomposición de Schur $Q^T AQ = D$, donde A se reescribe con la descomposición tridiagonal.
Por medio de transformaciones de Householder se calcula $T = (P_1 \cdots P_{n-2})^T A(P_1 \cdots P_{n-2})$.

1: $D := T$ (si se desea calcular $Q, Q = P_1 \cdots P_{n-2}$)
2: **repeat**
3: **for** $i = 1 : (n-1)$ **do**
4: $|d_{i+1,i}| = |d_{i,i+1}| \leq \text{tol}\,(|d_{ii}| + |d_{i+1,i+1}|)$
5: **end for**
6: Encuéntrese el mayor q y el menor p tales que si
7:
$$\begin{bmatrix} D_{11} & 0 & 0 \\ 0 & D_{22} & 0 \\ 0 & 0 & D_{33} \end{bmatrix} \begin{matrix} p \\ n-p-q \\ q \end{matrix}$$
$$\begin{matrix} p & n-p-q & q \end{matrix}$$
8: entonces D_{33} es diagonal y D_{22} es irreducible.
9: **if** $q < n$ **then**
10: Aplicamos Algoritmo QR simétrico implícito con desplazamiento de Wilkonson a D_{22}:
11: $D = \text{diag}(I_p, Z, I_q)^T \cdot D \cdot \text{diag}(I_p, Z, I_q)$.
12: (Si se desea calcular $Q, Q = Q \cdot \text{diag}(I_p, Z, I_q)$
13: **end if**
14: **until** q = n

Este algoritmo requiere aproximadamente $4n^3/3$ flops en el caso en el que Q no se acumule. Si por el contrario se acumula Q, se requieren aproximadamente $9n^3$ flops.

En el Algoritmo QR Simétrico los autovalores $\hat{\lambda}_i$ que se calculan corresponden a una matriz próxima a A. El error absoluto que se comete en el cálculo de cada autovalor es pequeño, del orden $|\hat{\lambda}_i - \lambda_i| \approx \mathbf{u}\|A\|_2$, donde λ_i es autovalor de A y $\mathbf{u}$ es la precisión de la máquina.

Nótese que si se quieren calcular todos los autovalores pero solo unos pocos autovectores, no es eficiente acumular la matriz Q.

2.4. Algoritmo QR: caso no simétrico

Sea $A \in \mathbb{R}^{n\times n}$ una matriz no simétrica previamente reducida a una matriz de Hessenberg superior H_0, y sea $U_0 \in \mathbb{R}^{n\times n}$ una matriz ortogonal. Presentamos a continuación una iteración QR:

$$\begin{aligned} &H_0 = U_0^T A U_0 \\ &\textbf{for } k = 1, 2, ... \\ &\quad H_{k-1} = U_k R_k \quad \text{(Factorización QR)} \\ &\quad H_k = R_k U_k \\ &\textbf{end} \end{aligned} \tag{2.7}$$

Analizamos a continuación la iteración anterior.
Partimos de una matriz A y construimos mediante rotaciones de Givens una matriz ortogonal U_0 tal que $H_0 = U_0^T A U_0$ es de Hessenberg superior. Esta matriz $H_0 \in \mathbb{R}^{n\times n}$ es semejante a A.
En el bucle, se define una forma de la iteración QR.
En primer lugar, descomponemos H_k en un producto de una matriz ortogonal U_k y una matriz triangular superior R_k para, en el paso siguiente, invertir el orden de dichas matrices y definir H_k (factorización QR).
Al igual que hicimos con el caso simétrico, veamos que la matriz resultante H_k es semejante a A.
Para cada $k, H_k = R_k U_k$. Como $H_{k-1} = U_k R_k \Rightarrow U_k^T H_{k-1} = U_k^T U_k R_k = R_k$, luego, $H_k = R_k U_k = U_k^T H_{k-1} U_k$. Entonces, como $H_{k-1} = U - k - 1 R_{k-1}$, razonando por inducción se tiene que

$$H_k = (U_0 U_1 \cdots U_k)^T A (U_0 U_1 \cdots U_k).$$

Por lo tanto, H_k es semejante a A.

2.4.1. Deflación

Introducimos en esta sección la *deflación*, una técnica que se emplea para simplificar el problema del cálculo de autovalores de una matriz grande, reduciéndolo para ello a problemas más pequeños.

Sea $H \in \mathbb{R}^{n\times n}$ una matriz de Hessenberg superior e irreducible. Supongamos, sin pérdida de generalidad, que tomamos H de (2.7). En cualquier otro caso, tenemos, con $1 \leq p < n$

$$H = \begin{array}{c} \left[\begin{array}{c|c} H_{11} & H_{12} \\ \hline 0 & H_{22} \end{array}\right] \begin{array}{l} p \\ n-p \end{array} \\ \begin{array}{cc} p & n-p \end{array} \end{array}$$

El proceso o problema de deflación ocurre cuando se puede dividir el problema en dos "subproblemas" independientes empleando esta partición.

Esto sucede cuando se encuentra un valor en la subdiagonal que es muy pequeño. En este caso, el problema de autovalores de H se lleva a cabo resolviendo de manera independiente los problemas de autovalores de H_{11} y de H_{22}, problemas que son más fáciles y eficientes computacionalmente hablando. De ahí el hecho de decir que se desacopla.

Por ejemplo, supongamos que, para c una constante pequeña

$$|h_{p+1,p}| \leq c\tilde{\mathbf{u}} \left(|h_{pp}| + |h_{p+1,p+1}|\right),$$

Esto lo que nos viene a decir es que si el valor de la subdiagonal $h_{p+1,p}$ es muy pequeño en comparación con los elementos de la subdiagonal, entonces podremos considerar dicha entrada como cero sin afectar a la exactitud puesto que los errores de redondeo del orden de $\tilde{\mathbf{u}}\|H\|$ siguen estando presentes en la matriz de cualquier manera.

2.4.2. Propiedades de las matrices de Hessenberg

La descomposición de Hessenberg no es única, por lo que existen muchas formas de escribir $A \in \mathbb{R}^{n\times n}$ como $Z^T AZ = H$ dependiendo de la matriz ortogonal $Z \in \mathbb{R}^{n\times n}$ que se use.
Supongamos A no simétrica, y descomponemos $Z^T AZ$ a una matriz de Hessenberg superior empleando para ello reflectores de Householder. Obtenemos como resultado $H = Q^T AQ$ matriz de Hessenberg superior, con $Q = ZU_0$, siendo U_0 una matriz ortogonal producto de matrices de Householder.

Veamos que el primer vector de Q determina H.
Si especificamos la primera columna de Q, entonces H es única. Esto se debe

a que

$$Qe_1 = (ZU_0)e_1 = Z(U_0e_1) = Ze_1,$$

eligiendo U_0 de manera que su primera columna sea e_1, es decir, $U_0e_1 = e_1$. De este modo, observamos que la primera columna de Q solamente depende de Z. Esto es esencialmente así siempre que H no tenga ceros en la subdiagonal. A las matrices de Hessenberg con esta propiedad se las llama *no reducidas*.

A continuación se prueban dos teoremas que nos aclaran esto.

Teorema 2.7 (**Teorema de la Q implícita - caso no simétrico -**). *Sea $A \in \mathbb{R}^{n\times n}$, y supongamos que $Q = [\, q_1 \,|\cdots|\, q_n \,]$ y $V = [\, v_1 \,|\cdots|\, v_n \,]$ son matrices ortogonales tales que $Q^TAQ = H$ y $V^TAV = G$, siendo H y G ambas matrices de Hessenberg superior.*
Sea, además, k el menor entero positivo para el cual $h_{k+1,k} = 0$, con la convención de que $k = n$ en el caso en el que H es irreducible.
Entonces, si $q_1 = v_1$, se tiene que $q_i = \pm v_i$ y $|h_{i,i-1}| = |g_{i,i-1}|$, para $2 \le i \le k$. Además, si $k < n$ se tiene que $g_{k+1,k} = 0$.

Demostración. Supongamos que $k \le n$.
Definamos en primer lugar la matriz ortogonal $W = V^TQ$.
Observemos que

$$W^TGW = \left(V^TQ\right)^T \left(V^TAV\right)\left(V^TQ\right) = Q^TVV^TAVV^TQ = H \Longrightarrow$$
$$\Longrightarrow WW^TGW = WH \Longrightarrow GW = WH.$$

Comparando las columnas $i-1$, para $i = 2 : k$ observamos que

$$(GW)(:, i-1) = Gw_{i-1} \tag{2.8}$$

$$(WH)(:, i-1) = \sum_{j=1}^{n} w_j h_{j,i-1}$$

Pero H es una matriz de Hessenberg superior por lo tanto $h_{j,i-1} = 0, \forall j > i$, luego

$$(WH)(:, i-1) = \sum_{j=1}^{i} h_{j,i-1} w_j \tag{2.9}$$

Igualando las ecuaciones (2.8) y (2.9) tenemos que

$$\begin{aligned} Gw_{i-1} = \sum_{j=1}^{i} h_{j,i-1}w_j = h_{i,i-1}w_i + \sum_{j=1}^{i-1} h_{j,i-1}w_j \iff \\ \iff h_{i,i-1}w_i = Gw_{i-1} - \sum_{j=1}^{i-1} h_{j,i-1}w_j \end{aligned} \tag{2.10}$$

Por hipótesis, sabemos que tanto Q como V son matrices ortogonales y que $q_1 = v_1$, luego $w_1 = V^T q_1 = V^T v_1 = e_1$.

Veamos ahora que W es triangular superior.
Como $W = V^T Q \Rightarrow w_{ij} = v_i^T q_j$. Q y V son matrices ortogonales construidas siguiendo una estructura de Hessenberg, y como $q_1 = v_1$, las siguientes q_i siempre podemos escribirlos como combinación lineal de los i primeros vectores v_i, es decir, $q_i \in \text{span}\{v_1, ..., v_i\}$, por lo tanto q_i es ortogonal a $v_{i+1}, v_{i+2}, ...$, por lo tanto $w_{ij} = v_i T q_j =$ si $i > j$, por lo tanto W es triangular superior.

Probemos ahora que $w_i = \pm I_n(:, i)$. Para ello razonamos por inducción sobre i.

- Para $i = 1$. Ya hemos visto que $w_1 = e_1$.
- Para $j < i$ lo suponemos cierto:

$$w_j = \pm e_j, \quad \forall j < i. \tag{H.I}$$

- Lo probamos para i. Vamos a emplear para ello la igualdad (2.10).
  Por (H.I) sabemos que $\forall j < i, w_j = \pm e_i$, entonces el sumatorio

$$\sum_{j=1}^{i-1} j_{j,j-1}w_j \in \text{span}\{e_1, ..., e_{i-1}\},$$

  es decir, es un vector que solamente tiene entradas en las posiciones $1, ..., i-1$.
  Por el otro lado de la igualdad, la matriz G es de Hessenberg superior, por lo tanto $Gw_{i-1} \in \text{span}\{e_1, ..., e_i\}$.
  Por lo tanto,

$$h_{i,i-1}w_i \in \text{span}\{e_1, ..., e_i\} \Longrightarrow w_i \in \text{span}\{e_1, ..., e_i\}.$$

Además, el vector w_i debe ser ortogonal a $w_1, ..., w_{i-1}$, y por (H.I) sabemos que estos vectores son, respectivamente, $\pm e_1, ..., \pm e_i$.
Por lo tanto, el último vector ortonormal en $\text{span}\{e_1, ..., e_i\}$ que es a su vez ortogonal a los vectores anteriores es $\pm e_i$, luego $w_i = \pm e_i$, como queríamos probar.

Sabemos que $\forall i, w_i = V^T q_i$, por lo tanto, $q_i = V w_i$, y como acabamos de probar que $w_i = \pm e_i$ entonces se tiene que $q_i = V(\pm e_i) = \pm v_i$.

Nos fijamos de nuevo en la igualdad (2.10) prestando atención a la i-ésima entrada de dicha columna, teniendo en cuenta que $w_i = \pm e_i$.
Observamos que $h_{i,i-1} w_i$ es un vector con todas las entradas nulas salvo la i-ésima posición, donde vale $\pm h_{i,i-1}$. Lo mismo ocurre con $G w_{i-1} \sum_{j=1}^{i-1} h_{j,i-1} w_j$, donde en la i-ésima posición vale $\pm g_{i,i-1}$.
Por lo tanto, $\pm h_{i,i-1} = \pm g_{i,i-1}$.
Tomando valor absoluto a ambos lados de la igualdad anterior se tiene lo buscado,

$$|h_{i,i-1}| = |g_{i,i-1}|.$$

Supongamos ahora que $h_{k+1,k} = 0$, siendo $k < n$. Veamos que $g_{k+1,k} = 0$.
Teniendo en cuenta que $e_k = \pm w_k = \pm W e_k$ y que $GW = WH$ se tiene que

$$g_{k+1,k} = e_{k+1}^T G e_k = \pm e_{k+1}^T GW e_k = \pm e_{k+1}^T WH e_k =$$
$$= \pm e_{k+1}^T \sum_{i=1}^{k} h_{ik} W e_i = \pm \sum_{i=1}^{k} h_{ik} e_{k+1}^T e_i = 0$$

□

Teorema 2.8. *Consideramos las matrices $A \in \mathbb{R}^{n\times n}$ y $Q \in \mathbb{R}^{n\times n}$, siendo esta última ortogonal. Entonces, $H = Q^T AQ$ es una matriz de Hessenberg superior irreducible si y solamente si $Q^T K(A, Q(:,1), n) = R$ es una matriz triangular superior no singular.*

Demostración. ($\Rightarrow$) Supongamos que $H = Q^T AQ$ es una matriz de Hessenberg superior irreducible. Podemos escribir

$$AQ = QH \Rightarrow AQ(:,1) = QHe_1 \Rightarrow A^2 Q(:,1) = A(QHe_1) = QH^2 e_1 \Rightarrow$$
$$\Rightarrow \cdots \Rightarrow A^{n-1} Q(:,1) = QH^{n-1} e_1.$$

Consideramos entonces la siguiente matriz de Krylov:

$$K(A, Q(:,1), n) = \left[\, Qe_1 \mid QH^2e_1 \mid \cdots \mid QH^{n-1}e_1 \right] = Q \left[\, e_1 \mid He_1 \mid \cdots \mid H^{n-1}e_1 \right],$$

por lo tanto

$$Q^T K(A, Q(:,1), n) = \left[\, e_1 \mid He_1 \mid \cdots \mid H^{n-1}e_1 \right] = R,$$

que es una matriz triangular superior puesto que H es una matriz de Hessenberg superior y lo mismo van a ser sus potencias.
Por otro lado, como H es irreducible tiene todas las entradas de la subdiagonal no nulas, por lo tanto los vectores $H^k e_1$ son todos linealmente independientes y por tanto R es no singular.

($\Leftarrow$) Supongamos ahora que $Q^T K(A, Q(:,1), n) = R$ es una matriz triangular superior no singular.
Las columnas de R son linealmente independientes, y como $R(:, k+1) = HR(:,1)$ se tiene que $H(:,k) \in \text{span}\{e_1, ..., e_{k+1}\}$, luego H es una matriz de Hessenberg superior, y como R es no singular se tiene que $r_{nn} \neq 0$, por lo tanto H es irreducible. □

2.4.3. Estrategia de doble desplazamiento implícito: el paso QR de Francis

La estrategia del doble desplazamiento implícito, también conocida como paso QR de Francis, es fundamental en el algoritmo QR para matrices no simétricas. Este procedimiento acelera la convergencia de la iteración QR mientras preserva la estructura de Hessenberg, permitiendo así un tratamiento más eficiente de matrices no simétricas.

El primero en introducir este método fue Francis (1961), quien describió cómo transformar una matriz de Hessenberg superior no reducida en otra de la misma forma mediante transformaciones de Householder, simulando un paso del algoritmo QR. El procedimiento que describe reduce de manera progresiva los elementos subdiagonales de la matriz hasta obtener bloques triangulares (o casi triangulares), lo que facilita la obtención de los autovalores de la matriz.

A continuación, se detalla el algoritmo de Francis para la implementación del paso QR con doble desplazamiento:

Algorithm 3 Paso QR de Francis

Dada una matriz $A \in \mathbb{R}^{n\times n}$ y una tolerancia tol mayor que el error de redondeo de la máquina, este algoritmo calcula la forma canónica real de Schur $Q^T AQ = T$.
Si se desean calcular tanto Q como T, entonces la matriz T se almacena en H.
Si solamente se quieren calcular los autovalores, entonces los bloques diagonales de T se almacenan en las posiciones correspondientes a H.

1: Sea $H = U_0^T AU_0$ la matriz de Hessenberg superior no reducible de la matriz $A \in \mathbb{R}^{n\times n}$, cuya submatriz principal de dimensión 2×2 del extremo inferior tienes autovalores a_1 y a_2.
2: Describimos cómo sobrescribir la matriz H con $Z^T HZ$, siendo Z producto de matrices de Householder y $Z^T (H - a_1 I)(H - a_2 I)$.
3: $m = n - 1$
4: Calculamos la primera columna de la matriz $(H - a_1 I)(H - a_2 I)$
5: $s = H(m, m) + H(n, n)$
6: $t = H(m, m) \cdot H(n, n) - H(m, n) \cdot H(n, m)$
7: $x = H(1, 1) \cdot H(1, 1) - H(1, 2) \cdot H(2, 1) - s \cdot H(1, 1) + t$
8: $y = H(2, 1) \cdot (H(1, 1) - H(2, 2) - s)$
9: $z = H(2, 1) \cdot H(3, 2)$
10: **for** $k = 0 : n - 3$ **do**
11: $[v, \beta] = \text{house}([x\ y\ z])^T$
12: $q = \text{máx}\{1, k\}$
13: $H(k + 1 : k + 3, q : n) = H(1 : r, k + 1, k + 3) \cdot H(k + 1 : k + 3, q : n)$
14: $r = \text{mín}\{1, k\}$
15: $H(1 : r, k + 1 : k + 3) = H(1 : r, k + 1, k + 3) \cdot (I - \beta vv^T)$
16: $x = H(k + 2, k + 1)$
17: $y = H(k + 3, k + 1)$
18: **if** $k < (n - 3)$ **then**
19: z = H(k + 4, k + 1)
20: **end if**
21: **end for**
22: $[v, \beta] = \text{house}([x\ y]^T)$
23: $H(n - 1 : n, n - 2 : n) = (I - \beta vv^T) \cdot H(n - 1 : n, n - 2 : n)$
24: $H(1 : n, n - 1 : n) = H(1 : n, n - 1 : n) \cdot (I - \beta vv^T)$

El algoritmo descrito aplica una secuencia controlada de transformaciones a bloques 3×3 y 2×2, y cada transformación consiste en anular subdiagonales sucesivas hasta que se restaura la forma de Hessenberg. El algoritmo está diseñado de tal forma que cada transformación solamente afecte a tres filas y columnas consecutivas, lo cual garantiza la eficiencia computacional. Además, se emplea el **Teorema de la Q implícita - caso no simétrico -**, que garantiza que el producto de transformaciones de Householder preserva la forma de Hessenberg. El coste computacional aproximado de la iteración descrita es de $10n^2$ flops.

2.4.4. Procedimiento general

Vamos a describir ahora el proceso completo para reducir una matriz real $A \in \mathbb{R}^{n\times n}$ a su forma de Schur real, paso esencial en el cálculo de autovalores de matrices no simétricas. Para ello, el primer paso consiste en reducir la matriz A a su forma de Hessenberg $H = U_0^T A U_0$, donde $U_0 = P_1 \cdots P_{n-2}$ es el producto de rotaciones de Householder. Durante este proceso iterativo, es fundamental observar los elementos subdiagonales de la matriz H ya que su comportamiento permite detectar posibles desacoplamientos.

Describimos a continuación cómo se lleva a cabo este procedimiento mediante el siguiente algoritmo:

Algorithm 4 Algoritmo QR No Simétrico

Dada una matriz $A \in \mathbb{R}^{n\times n}$ y una tolerancia tol mayor que el error de redondeo de la máquina, este algoritmo calcula la forma canónica real de Schur $Q^T AQ = T$.
Si se desean calcular tanto Q como T, entonces la matriz T se almacena en H.
Si solamente se quieren calcular los autovalores, entonces los bloques diagonales de T se almacenan en las posiciones correspondientes a H.

1: Sea $H = U_0^T AU_0$ la reducción de Hessenberg de la matriz A, siendo $U_0 = P_1 \cdots P_{n-2}$.
2: Si se desea Q es de la forma $Q = P_1 \cdots P_{n-2}$
3: **repeat**
4: Anulamos todos los elementos subdiagonales que cumplan
5: $|h_{i,i-1}| \leq \text{tol} \cdot (|h_{ii}| + |h_{i-1,i-1}|)$.
6: Encuéntrese el mayor q y el menor p tales que si
7: $$H = \begin{bmatrix} H_{11} & H_{12} & H_{13} \\ 0 & H_{22} & H_{23} \\ 0 & 0 & H_{33} \end{bmatrix} \begin{matrix} p \\ n-p-q \\ q \end{matrix}$$
$$\begin{matrix} p & n-p-q & q \end{matrix}$$
8: siendo H_{33} triangular superior y H_{22} no reducida.
9: **if** $q < n$ **then**
10: Aplicamos el Paso QR de Francis a H_{22}: $H_{22} = Z^T H_{22} Z$.
11: **if** Q es necesaria **then**
12: $Q = Q \cdot \text{diag}(I_p, Z, I_q)$.
13: $H_{12} = H_{12} Z$
14: $H_{23} = Z^T H_{23}$
15: **end if**
16: **end if**
17: **until** q = n
18: Triangularizamos superiormente todos los bloques diagonales de dimensión 2×2 en H que tengan autovalores reales (si es necesario se almacenan las transformaciones).

Cabe señalar que este algoritmo requiere, aproximadamente, en el caso en el que se desean calcular tanto la matriz ortogonal Q como la forma de Schur T, $25n^3$ operaciones en coma flotante (flops). Por el contrario, cuan-

do solamente se necesitan calcular los autovalores el coste computacional se reduce a unos $10n^3$ flops. Estas cifras se basan en observaciones empíricas, considerando que de media solamente se necesitan dos pasos QR de Francis antes de que se produzca el desacoplamiento, ya sea en bloques 1×1 o 2×2, por lo tanto representan una media orientativa.

Este algoritmo mantiene buenas propiedades de estabilidad puesto que la forma de Schur real calculada T es ortogonalmente similar a una matriz ligeramente perturbada de la matriz original A. Es decir, existe una matriz ortogonal Q tal que $Q^T(A + E)Q = T$. La matriz $A + E$ representa a la matriz ligeramente perturbada de la matriz A, donde el error E verifica que $\|E\|_2 \approx \mathbf{u}\|A\|_2$, siendo $\mathbf{u}$ la precisión de la máquina.

Capítulo 3

Problemas de autovalores en matrices grandes y dispersas

En este capítulo, desarrollamos el proceso de Lanczos, que calcula una secuencia de tridiagonalizaciones parciales ortogonalmente relacionadas con una matriz simétrica A dada. Es un método muy interesante cuando A es grande y dispersa ya que se basa en productos matriz-vector en lugar de ir actualizando la propia matriz durante el proceso. También es importante que durante las primeras iteraciones el método ya obtiene información sobre los autovalores extremos de A, lo cual dota de gran utilidad al método en las situaciones en las que solamente se desean calcular algunos de los autovalores extremos de A junto con sus autovectores correspondientes.

En 3.1 se presentan la derivación y los atributos aritméticos exactos del proceso simétrico de Lanczos junto con sus extraordinarias propiedades de convergencia. En 3.2 se describen varios "ajustes" prácticos para la implementación del método. La principal estrategia que describiremos será la denominada reortogonalización, tanto completa como selectiva.

3.1. El proceso de Lanczos simétrico

Sea $A \in \mathbb{R}^{n\times n}$ una matriz grande, dispersa y simétrica. Supongamos que solamente se desean calcular unos pocos de sus autovalores extremos

(mayores y/o menores). Este problema se resuelve con un método atribuido a Lanczos (1950), que genera una sucesión de matrices tridiagonales $\{T_n\}$ con la propiedad de que los autovalores extremos de $T_k \in \mathbb{R}^{k\times k}$ son estimaciones progresivamente mejores de los autovalores extremos de A.

En esta sección, derivamos la técnica e investigamos algunas de sus propiedades usando aritmética exacta. Una manera de motivar la idea de Lanczos es recordando las limitaciones del método de la potencia, estudiado en 2.2.1. Este método puede usarse para encontrar el autovalor dominante λ_1 y su correspondiente autovector asociado x_1. Sin embargo, la velocidad de convergencia viene dada por $|\frac{\lambda_2}{\lambda_1}|^k$, siendo λ_2 el segundo autovalor más grande en valor absoluto. A menos que exista una brecha significativa entre estos dos autovalores, este método es muy lento. Además, no se tiene en cuenta la "experiencia previa", ya que después de k pasos con el vector inicial $v^{(0)}$, disponemos de las direcciones definidas por los vectores $Av^{(0)}, A^2v^{(0)}, ..., A^kv^{(0)}$. Sin embargo, en lugar de buscar una estimación optima de x_1 en el espacio generado por estos vectores, se utiliza $A^kv^{(0)}$. Por otro lado, el método de iteración ortogonal con aceleración de Ritz aborda alguna de estas preocupaciones, pero también tiene una cierta falta de aprovechamiento de las iteraciones anteriores.

Necesitamos de esta forma un método que "aprenda de la experiencia" y que aproveche todos los productos matriz - vector previamente calculados, y el método de Lanczos cumple con ello.

3.1.1. Subespacios de Krylov

Para motivar el proceso de Lanczos y su relación con los subespacios de Krylov de una matriz partimos del cociente de Rayleigh.

$$r(x) = \frac{x^TAx}{x^Tx}, \quad x \neq 0.$$

Recordemos del Teorema 2.3 que los valores máximo y mínimo de $r(x)$ son respectivamente $\lambda_1(A)$ y $\lambda_n(A)$. Supongamos ahora que $q_i \in \mathbb{R}^n, i = 1, ..., k$ es una secuencia de vectores ortonormales con $Q = [\ q_1\ |\cdots|\ q_k\ ]$, y definamos los escalares M_k y m_k como:

$$M_k = \lambda_1(Q_k^TAQ_k) = \max_{y\neq 0} \frac{y^T(Q_k^TAQ_k)y}{y^Ty} = \max_{\|y\|_2=1} r(Q_ky) \leq \lambda_1(A),$$

$$m_k = \lambda_k(Q_k^T A Q_k) = \min_{y \neq 0} \frac{y^T (Q_k^T A Q_k) y}{y^T y} = \min_{\|y\|_2 = 1} r(Q_k y) \geq \lambda_n(A),$$

Como

$$\text{ran}(Q_1) \subset \text{ran}(Q_2) \subset \cdots \subset \text{ran}(Q_n) = \mathbb{R}^n$$

se sigue que

$$M_1 \leq M_2 \leq \ ... \leq M_n = \lambda_1(A),$$
$$m_1 \geq m_2 \geq \ ... \geq m_n = \lambda_n(A).$$

Por lo tanto, el proceso de optimización descrito eventualmente convergerá a los autovalores $\lambda_1(A)$ y $\lambda_n(A)$ respectivamente. El objetivo consiste en elegir los vectores q_i de tal modo que M_k y m_k sean estimaciones de alta calidad de $\lambda_1(A)$ y $\lambda_n(A)$ mucho antes de que k se iguale a n.

Para motivar la elección de los vectores q_i, consideremos el gradiente del cociente de Rayleigh,

$$\nabla r(x) = \frac{2}{x^T x} (Ax - r(x)x)$$

Supongamos que $u_k \in \text{span}\{q_1, ..., q_k\}$ satisface que $M_k = r(u_k)$.

Observamos que si $\nabla r(u_k) = 0$, entonces $(r(u_k), u_k)$ es un par propio de A, es decir, $r(u_k)$ es un autovalor de A y u_k es el correspondiente autovector asociado, ya que

$$\nabla r(u_k) = 0 \iff \frac{2}{u_k^T u_k} (Au_k - r(u_k)u_k) = 0 \iff Au_k - r(u_k)u_k = 0$$

En caso contrario, tiene sentido considerar, desde el punto de vista de hacer M_{k+1} lo más grande posible, que el siguiente vector pueba q_{k+1} sea tal que

$$\nabla r(u_k) \in \text{span}\{q_1, \ ..., q_{k+1}\} \tag{3.1}$$

Esto se debe a que $r(x)$ aumenta más rápidamente en la dirección del gradiente $\nabla r(x)$. La estrategia nos garantizará que $M_{k+1} > M_k$, y con suerte por una cantidad significativa. Del mismo modo, si $v_k \in \text{span}\{q_1, ..., q_k\}$ satisface $r(v_k) = m_k$, tiene sentido tomar

$$\nabla r(v_k) \in \text{span}\{q_1, ..., q_{k+1}\} \tag{3.2}$$

puesto que $r(x)$ decrece más rápido en la dirección de $-\nabla r(x)$. Nótese que para cualquier $x \in \mathbb{R}^n$ se tiene que

$$\nabla r(x) \in \text{span}\{x, Ax\}.$$

Dado que los vectores u_k y v_k pertenecen ambos al mismo conjunto $\text{span}\{q_1, \ ..., q_k\}$, se deduce que las inclusiones (3.1) y (3.2) se satisfacen si

$$\text{span}\{q_1, \ ..., q_k\} = \text{span}\{q_1, Aq_1, \ ..., A^{k-1}q_1\}$$

Esto nos sugiere elegir q_{k+1} de manera que

$$\text{span}\{q_1, \ ..., q_{k+1}\} = \text{span}\{q_1, Aq_1, \ ..., A^{k-1}q_1, A^k q_1\}$$

y por lo tanto llegamos al problema de calcular una base ortogonal del subespacio de Krylov

$$\mathcal{K}(A, q_1, k) = \text{span}\{q_1, A_1, \ ..., A^{k-1}q_1\}.$$

Nótese que $\mathcal{K}(A, q_1, k)$ es precisamente el subespacio que el método de la potencia "pasa por alto" ya que solamente busca en la dirección de $A^{k-1}q_1$.

3.1.2. Tridiagonalización

Vamos a aprovechar la conexión entre la tridiagonalización de A y la factorización QR de la matriz de Krylov $K(A, q_1, n)$ para generar una base ortonormal para dicho subespacio de Krylov.

En el **Teorema de la Q implícita - caso simétrico -** hemos visto que si $Q^T AQ = T$ es tridiagonal, siendo $Q^T Q = I_n$, entonces

$$K(A, q_1, n) = QQ^T K(A, q_1, n) = Q \left[\ e_1 \mid Te_1 \mid T^2 e_1 \mid ... \mid T^{n-1} e_1 \ \right]$$

es la factorización QR de $K(A, q_1, n)$, siendo e_1 y q_1 las primeras columnas de I_n y Q respectivamente. Observamos entonces que podemos generar de forma efectiva las columnas de Q tridiagonalizando A con una matriz ortogonal que tenga como primera columna a q_1.

Podemos adaptar la tridiagonalización de Householder con este fin, aunque es un enfoque poco práctico cuando se tiene una matriz A grande y

dispersa ya que las actualizaciones del método destruyen el carácter disperso de la matriz A. Se obtienen como resultado matrices densas y de gran tamaño que no podemos aceptar.
Como alternativa intentamos calcular directamente los elementos de T.
Designemos a las columnas de la matriz Q como

$$Q = [\ q_1 \mid \cdots \mid q_n\ ]$$

siendo $q_i = [q_{i1}, ..., q_{in}]^T$, y a la matriz tridiagonal T como

$$T = \begin{bmatrix} \alpha_1 & \beta_1 & 0 & \cdots & 0 \\ \beta_1 & \alpha_2 & \beta_2 & \cdots & 0 \\ 0 & \ddots & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \beta_{n-1} \\ 0 & \cdots & 0 & \beta_{n-1} & \alpha_n \end{bmatrix}.$$

Sabemos que $Q^T AQ = T$, luego $AQ = QT$, siendo

$$QT = \begin{bmatrix} q_{11} & q_{21} & \cdots & q_{n1} \\ q_{12} & q_{22} & \cdots & q_{n2} \\ \vdots & \vdots & & \vdots \\ q_{1n} & q_{2n} & \cdots & q_{nn} \end{bmatrix} \begin{bmatrix} \alpha_1 & \beta_1 & 0 & \cdots & 0 \\ \beta_1 & \alpha_2 & \beta_2 & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & \beta_{n-1} \\ 0 & \cdots & 0 & \beta_{n-1} & \alpha_n \end{bmatrix} =$$

$$= \begin{bmatrix} q_{11}\alpha_1 + q_{21}\beta_1 & q_{11}\beta_1 + q_{21}\alpha_2 + q_{31}\beta_2 & \cdots & q_{n-1,1}\beta_{n-1} + q_{n1}\alpha_n \\ q_{12}\alpha_1 + q_{22}\beta_1 & q_{12}\beta_1 + q_{22}\alpha_2 + q_{32}\beta_2 & \cdots & q_{n-1,2}\beta_{n-1} + q_{n2}\alpha_n \\ \vdots & \vdots & & \vdots \\ q_{1n}\alpha_1 + q_{2n}\beta_1 & q_{1n}\beta_1 + q_{2n}\alpha_2 + q_{3n}\beta_2 & \cdots & q_{n-1,n}\beta_{n-1} + q_{nn}\alpha_n \end{bmatrix}$$

Entonces, igualando columnas en $AQ = QT$ se tiene que:

$$Aq_k = q_{k-1}\beta_{k-1} + q_k\alpha_k + q_{k+1}\beta_k, \tag{3.3}$$

con $\beta_0 q_0 \equiv 0,\ k \in \{1, ..., n-1\}$. Observamos así que las columnas de AQ son combinación lineal de q_{k-1}, q_k y q_{k+1}, con $k \in \{1, ..., n-1\}$.

Tomando producto escalar con q_k en ambos lados de la igualdad (3.3) se tiene

$$\begin{aligned} q_k^T Aq_k &= q_k^T(q_{k-1}\beta_{k-1} + q_k\alpha_k + q_{k+1}\beta_k) = \\ &= q_k^T q_{k-1}\beta_{k-1} + q_k^T q_k\alpha_k + q_k^T q_{k+1}\beta_k. \end{aligned}$$

Como las columnas de Q son vectores ortonormales, sabemos que

$$q_i^T q_j = \delta_{ij} = \begin{cases} 1 & \text{si} \quad i = j \\ 0 & \text{si} \quad i \neq j \end{cases}$$

luego,

$$q_k^T A q_k = \alpha_k.$$

Este valor α_k es la llamada *proyección de Aq_k sobre q_k*, aunque también es conocido como el "valor esperado" de A en dirección de q_k.

Vamos a definir ahora el vector residual r_k. Para ello vamos a despejar $\beta_{k-1}q_{k+1}$ en (3.3):

$$r_k = \beta_k q_{k+1} = Aq_k - q_{k+1}\beta_{k-1} - q_k\alpha_k = (A - \alpha_k I)q_k - q_{k-1}\beta_{k-1}. \tag{3.4}$$

Recordemos que nuestro objetivo es construir una base ortonormal.
Vamos a distinguir dos casos:

- Si $\underline{r_k \neq 0}$. En este caso podemos escribir $q_{k+1} = r_k/\beta_k$.
  Buscamos q_{k+1} unitario, es decir,

  $$\|q_{k+1}\|_2 = 1 \Longleftrightarrow \left\| \frac{r_k}{\beta_k} \right\|_2 = 1 \Longleftrightarrow \frac{\|r_k\|_2}{|\beta_k|} = 1,$$

  luego necesariamente debe cumplirse que $|\beta_k| = \|r_k\|_2 \Rightarrow \beta_k = \pm\|r_k\|_2$, sin haber pérdida de generalidad al tomar β_k positivo.

- Si $\underline{r_k = 0}$. La iteración se interrumpe y no se obtiene información relevante sobre el subespacio invariante. Observamos en este caso que Aq_k está completamente contenido en el subespacio generado por los vectores $q_1, ..., q_k$ puesto que $Aq_k = \beta_{k-1}q_{k+1} + \alpha_k q_k$.

Secuenciando de manera adecuada las fórmulas anteriores y asumiendo que $q_1 \in \mathbb{R}^n$ es un vector unitario dado, obtenemos una primera versión de la iteración de Lanczos en la que se generan los elementos de la base ortonormal.

Algorithm 5 Tridiagonalización de Lanczos

Dados $A \in \mathbb{R}^{n\times n}$ simétrica y $q_1 \in \mathbb{R}^n$ un vector unitario (norma 2), calculamos una matriz ortogonal $Q_k = [\, q_1 \mid \cdots \mid q_n \,]$ y una matriz tridiagonal simétrica $T_k \in \mathbb{R}^{n\times n}$ tales que $AQ_k = Q_k T_k$.
Las entradas de la diagonal principal y la diagonal superior de T_k son $\alpha_1, \ldots, \alpha_k$ y $\beta_1, \ldots, \beta_k$ respectivamente, con $1 \leq k \leq n$.

1: $k = 0, \beta_0 = 1, q_0 = 0, r_0 = q_1$
2: **while** $k = 0$ **or** $\beta_k \neq 0$ **do**
3: $\quad q_{k+1} = r_k/\beta_k$
4: $\quad k = k + 1$
5: $\quad \alpha_k = q_k^T A q_k$
6: $\quad r_k = (A - \alpha_k I)q_k - \beta_{k-1} q_{k-1}$
7: $\quad \beta_k = \|r_k\|_2$
8: **end while**

A los vectores q_k se les denominan *vectores de Lanczos*. En la sección 3.2 se estudian formas numéricas mejores que el proceso de tridiagonalización de Lanczos para calcular los vectores anteriores.

3.1.3. Terminación y cotas de error

Acabamos de ver en la sección anterior que la iteración de Lanczos se detiene antes de la tridiagonalización completa si q_1 está contenido en un subespacio propio invariante. En el siguiente teorema resumimos varias propiedades del método, esta entre otras.

Teorema 3.1. *La Tridiagonalización de Lanczos se ejecuta hasta que $k = m$, siendo*

$$m = rank(K(A, q_1, n)).$$

Además, cuando $k = 1 : m$ se tiene que

$$AQ_k = Q_k T_k + r_k e_k^T$$

donde $Q_k = [\, q_1 \mid \cdots \mid q_k \,]$ tiene columnas ortonormales que expanden $\mathcal{K}(A, q_1, n)$, $e_k =$

$I_n(:,k)$ y

$$T = \begin{bmatrix} \alpha_1 & \beta_1 & 0 & \cdots & 0 \\ \beta_1 & \alpha_2 & \beta_2 & \cdots & 0 \\ 0 & \ddots & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \beta_{n-1} \\ 0 & \cdots & 0 & \beta_{n-1} & \alpha_n \end{bmatrix}. \tag{3.5}$$

Demostración. Veamos que $\text{ran}(Q_k) = \mathcal{K}(A, q_1, n)$, $\forall k \leq m$, siendo $m = \text{rank}(K(A, q_1, n))$.

Para ello, razonamos por inducción sobre k.

- Supongamos que $k = 1$.

  Entonces $Q_1 = [\ q_1\ ]$, y es evidente que $\mathcal{K}(A, q_1, 1) = \text{span}\{q_1\}$.

- Suponemos ahora que para un cierto $k > 1$ la iteración de Lanczos ha producido $Q_k = [\ q_1 \mid \cdots \mid q_k\ ]$ con columnas ortonormales tales que

$$\text{ran}(Q_k) = \mathcal{K}(A, q_1, k) \tag{H.I}$$

- Lo probamos para $k+1$. Veamos que $\text{ran}(Q_{k+1}) = \mathcal{K}(A, q_1, k+1)$.

  De la Tridiagonalización de Lanczos se tiene que

$$Aq_k = q_{k-1}\beta_{k-1} + q_k\alpha_k + q_{k+1}\beta_k.$$

  Como q_k y q_{k-1} son respectivamente la última y la penúltima columna de la matriz Q_k, es evidente que $q_k, q_{k-1} \in \mathcal{K}(A, q_1, k) \subseteq \mathcal{K}(A, q_1, k+1)$.

  Por lo tanto,

$$q_{k-1}\beta_{k-1} + \alpha_k q_k \in \text{ran}(Q_{k+1}).$$

  Recordemos que en (3.4) definimos el vector residual r_k como

$$r_k = \beta_k q_{k+1} = Aq_k - (q_{k-1}\beta_k + q_k\alpha_k).$$

  Distinguimos dos casos:

  - Si $r_k \neq 0$. Hemos visto ya en la sección anterior que $q_{k+1} = r_k/\|r_k\|_2$, siendo $\beta_k = \pm\|r_k\|_2$.

En este caso, q_{k+1} es ortogonal a $q_1, ..., q_k$, luego q_{k+1} no puede ser combinación lineal de $q_1, ..., q_k$ por lo que $q_{k+1} \notin \text{span}\{q_1, ..., q_k\} = \text{ran}(Q_k)$. Por lo tanto, q_{k+1} aporta una nueva dimensión a $\text{ran}(Q_k)$ y

$$q_{k+1} \in \mathcal{K}(A, q_1, k) \subseteq \mathcal{K}(A, q_1, k+1).$$

Entonces,

$$\text{ran}(Q_{k+1}) = \mathcal{K}(A, q_1, k+1).$$

- Si $\underline{r_k = 0}$. Reescribimos (3.3) en forma matricial ayudándonos de (3.4), y se tiene que

$$AQ_k = Q_k T_k + \beta_k q_{k+1} e_k^T = Q_k T_k + r_k e_k^T. \tag{3.6}$$

En este caso, $AQ_k = Q_k T_k$, y se tiene que $\text{ran}(Q_k) = \mathcal{K}(A, q_1, k)$ es invariante para A. En consecuencia, $k = m = \text{rank}(K(A, q_1, n))$.

□

Nos interesa encontrar un $\beta_k = 0$ en la Tridiagonalización de Lanczos ya que nos indica el cálculo de un subespacio invariante exacto, aunque información valiosa sobre el subespacio invariante suele aparecer mucho antes de encontrarse un valor de β pequeño.

Aparentemente, puede extraerse más información de T_k y del subespacio de Krylov generado por las columnas de Q_k.

3.1.4. Aproximaciones de Ritz

Definición 3.2. Sean S un subespacio de $\mathbb{R}^n$ y $A \in \mathbb{R}^{n\times n}$ simétrica. Decimos que (θ, y) es un *par de Ritz* para A respecto de S si $w^T(Ay - \theta y) = 0,\ \forall w \in S$. Respectivamente, θ e y se denominan *valor de Ritz* y *vector de Ritz* de A respecto a S.

Cuando $S = \mathcal{K}(A, q_1, k)$, el proceso de Lanczos puede emplearse para calcular los valores y vectores y de Ritz de a A asociados a S. Veámoslo.

Sean $A \in \mathbb{R}^{n\times n}$ simétrica y $Q^T AQ = T$ su descomposición triangular, con $Q^T Q = I_n$.

Supongamos que

$$S_k^T T_k S_k = \Theta_k = \text{diag}(\theta_1, ..., \theta_k)$$

es una descomposición de Schur de la matriz tridiagonal T_k.

Definimos la matriz $Y_k \in \mathbb{R}^{n\times k}$ como una combinación de las k primeras columnas de Q y de S_k, es decir,

$$Y_k = [\ y_1 \mid \cdots \mid y_k\ ] = Q_k S_k.$$

Veamos ahora que para cada i, con $1 \leq i \leq k, (\theta_i, y_i)$ es un par de Ritz de A con respecto a S. Veámoslo de forma matricial.

Probar que $w^T(Ay_i - \theta_i y_i) = 0$ es lo mismo que ver que el residuo $(Ay_i - \theta_i y_i)$ es ortogonal a todo el subespacio S, es decir, a cada vector $q_j \in S$.

Esto es equivalente a ver que $Q_k^T(AY_k - Y_k\Theta_k) = 0$, siendo $AY_k - Y_k\Theta_k$ la matriz que tiene en cada columna el residuo de cada par de Ritz, es decir,

$$AY_k - Y_k\Theta_k = [\ Ay_1 - \theta_1 y_1 \mid \cdots \mid Ay_k - \theta_k y_k\ ]$$

Luego,

$$\begin{aligned} Q_k^T(AY_k - Y_k\Theta_k) &= Q_k^T(AQ_kS_k - Q_kS_kS_k^TT_kS_k) = \\ &= Q_k^TAQ_kS_k - Q_k^TQ_kT_kS_k = T_kS_k - T_kS_k = 0 \end{aligned}$$

Como $Q_k \neq 0 \Rightarrow AY_k - Y_k\Theta_k = 0$, luego hemos probado que (θ_i, y_i) es un par de Ritz para cada i, con $1 \leq i \leq k$.

A continuación se estudian dos teoremas sobre la aproximación de Ritz, que son interesantes en el contexto de Lanczos.

El primero de ellos nos lleva a ver que los θ_i son los autovalores de una "matriz óptima" tridiagonal mientras que el segundo puede emplearse para proporcionar una cota para $\|Ay_i - \theta_i y_i\|_2$.

Teorema 3.3. *Si $A \in \mathbb{R}^{n\times n}$ es simétrica y $Q_1 \in \mathbb{R}^{n\times r}$ es tal que tiene columnas ortonormales, entonces*

$$\min_{S\in\mathbb{R}^{r\times r}} \|AQ_1 - Q_1S\|_F = \|(I - Q_1Q_1^T)AQ_1\|_F$$

y $S = Q_1^TAQ_1$ es el minimizador.

Demostración. Sea $Q_2 \in \mathbb{R}^{n\times(n-k)}$ tal que $Q = [\, Q_1 \mid Q_2 \,]$ es ortogonal. De esta forma extendemos Q_1 a una base ortonormal completa de $\mathbb{R}^n$, es decir, $Q \in \mathbb{R}^{n\times n}$ y cumple que $Q^TQ = I_n$.

Sea $S \in \mathbb{R}^{r\times r}$. Teniendo en cuenta que Q es ortogonal y que la norma de Frobenius es invariante para el producto de matrices ortogonales, se tiene que

$$\|AQ_1 - Q_1S\|_F^2 = \|Q^T(AQ_1 - Q_1S)\|_F^2 = \|Q^TAQ_1 - Q^TQ_1S\|_F^2.$$

Analicemos cómo son estas matrices.

Observemos que $Q = [\, Q_1 \mid Q_2 \,] \Rightarrow Q^T = \begin{bmatrix} Q_1^T \\ Q_2^T \end{bmatrix}$, luego

$$Q^TAQ_1 = \begin{bmatrix} Q_1^T \\ Q_2^T \end{bmatrix} AQ_1 = \begin{bmatrix} Q_1^TAQ_1 \\ Q_2^TAQ_1 \end{bmatrix}$$

y

$$Q^TQ_1 = \begin{bmatrix} Q_1^T \\ Q_2^T \end{bmatrix} Q_1 = \begin{bmatrix} Q_1^TQ_1 \\ Q_2^TQ_1 \end{bmatrix} = \begin{bmatrix} I_k \\ 0 \end{bmatrix},$$

luego

$$Q^TQ_1S = \begin{bmatrix} I_k \\ 0 \end{bmatrix} S = \begin{bmatrix} S \\ 0 \end{bmatrix}.$$

Por lo tanto,

$$Q^T(AQ_1 - Q_1S) = \begin{bmatrix} Q_1^TAQ_1 - S \\ Q_1^TAQ_2 \end{bmatrix} \Longrightarrow$$
$$\Longrightarrow \|Q^T(AQ_1 - Q_1S)\|_F^2 = \|Q_1^TAQ_1 - S\|_F^2 \; + \; \|Q_1^TAQ_2\|_F^2 \; .$$

Observamos que $Q_1^TAQ_2$ no depende de S, por lo tanto, cuando minimizamos solamente afecta al primer término de la última igualdad, es decir, a $\|Q_1^TAQ_1 - S\|_F^2$.

Es evidente que el mínimo S que puede minimizar la norma es $S = Q_1^TAQ_1$ ya que en este caso la norma vale 0 (norma mínima), como se quería ver. □

Teorema 3.4. *Sean $A \in \mathbb{R}^{n\times n}$ simétrica y $Q_1 \in \mathbb{R}^{n\times r}$ tal que $Q_1^TQ_1 = I_r$. Si*

$$Z^T(Q_1^TAQ_1)Z = diag(\theta_1, ..., \theta_r) = \Theta$$

es la descomposición de Schur de la matriz $Q_1^T A Q_1$ *y* $Q_1 Z = [\, y_1 \mid \cdots \mid y_r \,]$, *entonces*

$$\|Ay_k - \theta_k y_k\|_2 = \|(I - Q_1 Q_1^T) A Q_1 Z e_k\|_2 \leq \|(I - Q_1 Q_1^T) A Q_1\|_2,$$

para $1 \leq k \leq r$.

Demostración. Por hipótesis sabemos que, para cada k, $y_k = Q_1 z_k$, siendo $z_k = Z e_k$ la k-ésima columna de la matriz Z.

Entonces,

$$Ay_k - \theta_k y_k = A Q_1 Z_k - \theta_k Q_1 z_k.$$

Sabemos, también por hipótesis, que para cada k, θ_k es un autovalor de la matriz $Q_1^T A Q_1$ asociado a z_k.

Luego,

$$Q_1^T A Q_1 = \theta_k z_k \Longrightarrow Q_1 Q_1^T A Q_1 z_k = Q_1 \theta_k z_k = \theta_k Q_1 z_k.$$

Por tanto,

$$A Q_1 z_k - \theta_k Q_1 z_k = A Q_1 z_k - Q_1 Q_1^T A Q_1 z_k = (I - Q_1 Q_1^T) A Q_1 z_k.$$

Tomando normas en la igualdad anterior se tiene que

$$\|Ay_k - \theta_k y_k\|_2 = \|(I - Q_1 Q_1^T) A Q_1 z_k\|_2 \leq \|(I - Q_1 Q_1^T) A Q_1\|_2 \, \|z_k\|_2.$$

Z es una matriz ortogonal, luego $\forall k, z_k$ es un vector ortonormal y se tiene que $\|z_k\|_2 = 1$.

Por lo tanto, queda demostrado que

$$\|Ay_k - \theta_k y_k\|_2 = \|(I - Q_1 Q_1^T) A Q_1 z_k\|_2 \leq \|(I - Q_1 Q_1^T) A Q_1\|_2.$$

□

Realmente esto podemos hacerlo mejor. Utilizando (3.6) tenemos

$$Ay_i - \theta_i y_i = (A Q_k - Q_k T_k) S_k e_i = r_k (e_k^T S_k e_i).$$

Tomando módulos en la igualdad anterior, y recordando que $\|r_k\| = |\beta_k|$, se obtiene que

$$\|Ay_i - \theta_i y_i\| = \|r_k (e_k^T S_k e_i)\| = \|r_k\| \|e_k^T S_k e_i\| = |\beta_k| \cdot |s_{ki}| < |\beta_k| \qquad (3.7)$$

Hemos tenido en cuenta que S_k es ortogonal y por lo tanto $|s_{ki}| < 1$.

Golub y Van Loan ([2]) emplearon la desigualdad recién vista (3.7) para obtener un límite de error computable.

3.1.5. Teoría de la convergencia

La discusión anterior muestra cómo el proceso de Lanczos permite obtener estimaciones de autovalores. Sin embargo, aún no se ha tratado la cuestión sobre cómo de precisa ha de ser la estimación de los autovalores de las matrices T_k en función de k. A continuación se exponen varios resultados teóricos sobre el tema, cuyas demostraciones no serán estudiadas, desarrollados por Kaniel, Saad y Paige entre otros.

Teorema 3.5. *Sean A una matriz simétrica y $Z^T AZ = \Lambda = diag(\lambda_1, ..., \lambda_n)$ su correspondiente descomposición de Schur.*
Supongamos que se realizan k pasos de la Tridiagonalización de Lanczos, y sea T_k la matriz tridiagonal descrita en (3.5).
Si $\theta_1 = \lambda_1(T_k)$, entonces

$$\lambda_1 \geq \theta_1 \geq \lambda_1 - (\lambda_1 - \lambda_n)\left(\frac{\tan(\phi_1)}{c_{k-1}(1+2\rho_1)}\right)^2,$$

donde

$$\cos(\phi_1) = |q_1^T z_1| \quad y \quad \rho_1 = \frac{\lambda_1 - \lambda_2}{\lambda_2 - \lambda_n},$$

siendo $c_{k-1}(x)$ el polinomio de Chebyshev de grado $k-1$.

Demostración. Véase Golub - Van Loan ([2]) para más detalles de la demostración. □

El siguiente corolario es un resultado análogo al teorema anterior pero para el autovalor más pequeños de T_k.

Corolario 3.6. *Empleando la notación del Teorema 3.5, si $\theta_k = \lambda_k(T_k)$, entonces*

$$\lambda_n \leq \theta_k \leq \lambda_n + (\lambda_1 - \lambda_n)\left(\frac{\tan(\phi_1)}{c_{k-1}(1+2\rho_n)}\right)^2,$$

donde

$$\rho_n = \frac{\lambda_{n-1} - \lambda_n}{\lambda_1 - \lambda_{n-1}} \quad y \quad \cos(\phi_n) = q_1^T z_n.$$

El resultado principal de la sección es el siguiente:

Teorema 3.7. *Empleando la notación del Teorema 3.5, si* $1 \leq i \leq k$ *y* $\theta_i = \lambda_i(T_k)$*, entonces*

$$\lambda_i \geq \theta_i \geq \lambda_i - (\lambda_1 - \lambda_n)\left(\frac{k_i \tan(\phi_i)}{c_{k-i}(1+2\rho_i)}\right)^2,$$

donde

$$\rho_i = \frac{\lambda_i - \lambda_{i+1}}{\lambda_{i+1} - \lambda_n}, \quad k_i = \prod_{j=1}^{i-1} \frac{\theta_j - \lambda_n}{\theta_j - \lambda_i} \quad y \quad \cos(\phi_i) = |q_1^T z_i|.$$

Demostración. Véase Saad ([9]) para ver en detalle la demostración. □

Obsérvese que en este caso las cotas se deterioran según aumenta i debido a k_i y al grado del polinomio de Chebyshev, que es menor en comparación con los anteriores.

3.2. Procedimientos prácticos de Lanczos

El comportamiento de la iteración de Lanczos se ve afectado por los errores de redondeo. El problema principal es la pérdida de la ortogonalidad de los vectores de Lanczos, que dejan de ser ortogonales entre si, lo cual complica saber cuándo terminar el proceso y cómo se relacionan los autovalores de la matriz A con los de T_k. Esta dificultad, junto con la aparición de métodos de tridiagonalización más estables, hizo que durante un tiempo se prefirieran otras alternativas.

Householder explicó por qué muchos analistas numéricos no usaron el algoritmo de Lanczos durante los años 50 y 60, pero la situación cambió cuando surgió la necesidad de resolver problemas con matrices grandes y dispersas, lo cual fue posible gracias a Paige. El método de Lanczos se popularizó porque se necesitan muchas menos iteraciones para conseguir una buena aproximación del autovalor mayor.

En esta sección explicamos varias ideas y métodos que se han desarrollado para que la aplicación del método de Lanczos sea realmente efectiva.

3.2.1. Almacenamiento y trabajo requeridos

Con una cuidadosa reescritura de la Tridiagonalización de Lanczos y empleando la fórmula

$$\alpha_k = q_k^T(Aq_k - \beta_{k-1}q_{k-1}),$$

se puede llevar a cabo todo el proceso de Lanczos usando solo un par de vectores de dimensión n como se sigue:

$w = q_1,\ v = Aw,\ \alpha_1 = w^T v,\ v = v - \alpha_1 w,\ \beta_1 = \|v\|_2,\ k = 1$
while $\beta_k \neq 0$
  for $i = 1 : n$
    $t = w_i,\ w_i = v_i/\beta_k,\ v_i = -\beta_k t$
  end
  $v = v + Aw$
  $k = k + 1,\ \alpha_k = w^T v,\ v = v - \alpha_k w,\ \beta_k = \|v\|_2$
end

Al finalizar este bucle, el vector w contiene a q_k y v al residuo $r_k = Aq_k - \alpha_k q_k - \beta_{k-1}q_{k-1}$.

Un detalle importante es que la matriz A no se modifica durante el proceso, lo cual hace que sea un método muy útil para matrices largas y dispersas.

Al finalizar el proceso, los autovalores de T_k pueden calcularse con el algoritmo QR simétrico, discutido en 2.3, o con otros métodos especializados. Además, los vectores de Lanczos se van generando y almacenando en el vector w. Si se requirieran los autovectores de los autovalores (valores de Ritz) calculados sería necesario ir almacenando todos los vectores de Lanczos.

3.2.2. Propiedades de redondeo

El descubrimiento de un modelo de tridiagonalización de Lanczos con rearranque facilitó mucho el análisis teórico del método, el cual fue desarrollado por Paige ([4], [5] [6]), quien también propuso varias mejoras y versiones prácticas del algoritmo.

Después de j pasos del algoritmo, se puede construir una matriz con los vectores de Lanczos computados $\hat{Q} = [\ \hat{q}_1 \mid \cdots \mid \hat{q}_k\ ]$, y una matriz tridiagonal asociada de la forma

$$\hat{T}_k = \begin{bmatrix} \hat{\alpha}_1 & \hat{\beta}_1 & & \cdots & 0 \\ \hat{\beta}_1 & \hat{\alpha}_2 & \ddots & & \vdots \\ & \ddots & \ddots & \ddots & \\ \vdots & & \ddots & \ddots & \hat{\beta}_{k-1} \\ 0 & \cdots & & \hat{\beta}_{k-1} & \hat{\alpha}_k \end{bmatrix}.$$

Paige demostró que si $\hat{r}_k$ es el residuo computado en el paso k, entonces se cumple la relación

$$A\hat{Q}_k = \hat{Q}_k\hat{T}_k + \hat{r}_k e_k^T + E_k,$$

con

$$\|E_k\|_2 \approx \tilde{\mathbf{u}}\|A\|_2,$$

siendo $\tilde{\mathbf{u}}$ el error de redondeo de la máquina.
Con esto, Paige quiso enseñar que la fórmula del método de Lanczos se cumple con buena precisión, dentro del error esperado por la aritmética coma flotante.

Desafortunadamente, en la práctica los vectores $\hat{q}_i$ dejan de ser ortogonales después de varios pasos, lo cual provoca que los cálculos pierdan calidad. Como consecuencia de esto, los resultados que se obtienen pueden ser imprecisos o incorrectos.
Por ejemplo, si las normas de los residuos $\hat{\beta}_i$ y los cocientes $\hat{f}_i/\hat{\beta}_i$ son pequeños, se puede demostrar que

$$\hat{\beta}_i + \hat{q}_{i+1} \approx r_i + \mathbf{u}_i,$$

siendo $\mathbf{u}_i$ un error pequeño.

Esto implica que el siguiente vector de Lanczos ya no es ortogonal a los anteriores vectores, afectando esto a la precisión del método puesto que si se pierde la ortogonalidad entonces el subespacio generado por Lanczos ya no es ideal y se acumulan los errores.

Cabe destacar que la pérdida de la ortogonalidad es algo que siempre ocurre en la práctica y con ella, un aparente deterioro en la calidad de los autovalores de $\hat{T}_k$.

Esto nos lleva directamente a un procedimiento "práctico" de Lanczos, puesto que pueden emplearse técnicas para re-ortogonalizar o reiniciar el algoritmo, aunque tiene un coste computacional.

3.2.3. Lanczos con reortogonalización completa

Como solución al problema de la pérdida de ortogonalización se propone el método de Lanczos con reortogonalización completa, que busca mantener la ortogonalidad entre los vectores generados durante la iteración. Para ello, se emplean transformaciones de Householder, que generan vectores ortogonales de manera precisa.
Veamos el proceso, el cual consiste en proyectar los nuevos vectores de Lanczos al subespacio generado por los anteriores.

Sean $r_0, ..., r_{k-1} \in \mathbb{R}^n$ unos vectores dados, y supongamos que $H_0, ..., H_{k-1}$ son matrices de Householder tales que la matriz $(H_0 \cdots H_{k-1})^T [\ r_0 \mid \cdots \mid r_{k-1}\ ]$ es triangular superior. Denotemos por $[\ q_1 \mid \cdots \mid q_k\ ]$ a las primeras k columnas del producto matrices de Householder $(H_0 \cdots H_{k-1})$.
Supongamos ahora que nos dan un vector $r_k \in \mathbb{R}^n$ y queremos calcular un vector unidad q_{k+1} en la dirección del vector

$$w = r_k - \sum_{i=1}^{k} \left(q_i^T r_k\right) q_i \in \text{span}\{q_1, ..., q_k\}^{\perp}.$$

Si la matriz de Householder H_k es tal que $(H_0 \cdots H_k)^T [\ r_0 \mid \cdots \mid r_k\ ]$ es una matriz triangular superior, entonces resulta que la columna $k+1$ de la matriz $(H_0 \cdots H_k)$ es el vector unitario deseado.

Incorporando estos cálculos al proceso de Lanczos podemos procesar vectores de Lanczos ortogonales. Veamos un ejemplo de lo que se conoce como

reortogonalización de Lanczos completa:

$r_0 = q_1$ (vector unidad dado)
Determinar la matriz de Householder H_0 tal que $H_0 r_0 = e_1$.
for $k = 1 : (n-1)$
$\alpha_k = q_k^T A q_k$
$r_k = (A - \alpha_k I) q_k - \beta_{k-1} q_{k-1}, \qquad (\beta_0 q_0 \equiv 0)$
$w = (H_{k-1} \cdots H_0) r_k$
Determinar la matriz de Householder H_k tal que $H_k w = [w_1, ..., w_k, \beta_k, 0, ..., 0]^T$.
$Q_{k+1} = H_0 \cdots H_k e_{k+1}$
end

(3.8)

La idea de emplear matrices de Householder para hacer cumplir la ortogonalidad fue desarrollada por Golub y Wilkonson entre otros ([1]).

De las propiedades de redondeo de las matrices de Householder se deduce que los vectores $\hat{q}_i$ calculados en el algoritmo recién presentado en (3.8) son ortogonales al error de precisión de la máquina.

Nótese que, tal y como está definido el vector $\hat{q}_{k+1}$, la cosa no cambia si $\beta_k = 0$. Gracias a esto, el algoritmo funciona con total seguridad hasta que $k = (n-1)$, aunque en la práctica el algoritmo termina para un valor de k mucho más pequeño.

Para cualquier otra implementación de (3.8) nunca se construye de forma explícita el producto matricial correspondiente, lo que se hace es ir almacenando los vectores de Householder v_k. Además, no es necesario calcular los k primeros componentes del vector w ya que no se usan, por ello la situación idónea sería que fueran cero.

Aunque el método de Lanczos con reortogonalización completa mejora la estabilidad y precisión del método de Lanczos en los casos más precisos, resulta ser un método más costoso, no solamente en términos de tiempo sino también de memoria de almacenamiento debido a los cálculos de Householder, que incrementan de forma significativa el trabajo en el k-ésimo paso del método propuesto en $\mathcal{O}(kn)$ operaciones de coma flotante.

Cabe mencionar que se pueden tomar cursos de acción más eficaces que

no se tratan en este trabajo puesto que requieren una mayor compresión acerca de la pérdida de la ortogonalidad de los vectores.

Uno de estos métodos de acción es la reortogonalización selectiva. Si el vector de Lanczos $\hat{q}_{k+1}$ calculado más recientemente tiene una componente no trivial y no deseada en la dirección de cualquier vector de Ritz convergente ya calculado, se puede ortogonalizar $\hat{q}_{k+1}$ solo respecto a los vectores de Ritz convergentes ya calculados. En consecuencia, en lugar de ortogonalizar $\hat{q}_{k+1}$ en contra de todos los vectores de Lanczos previamente calculados, podemos lograr el mismo efecto ortogonalizándolos contra un conjunto mucho más pequeño de vectores de Ritz convergentes.

En su esquema, conocido como *reortogonalización selectiva*, a un par de Ritz calculado $\{\hat{\theta}, \hat{y}\}$ se le denomina "bueno" si satisface

$$\|A\hat{y} - \hat{\theta}\hat{y}\|_2 \leq \sqrt{\mathbf{u}}\|A\|_2.$$

Tan pronto como se calcula $\hat{q}_{k+1}$, se ortogonaliza respecto de cada buen vector de Ritz. Esto es mucho menos costoso que completar la reortogonalización, desde que, al menos al principio, hay muchos menos buenos vectores de Ritz que vectores de Lanczos.

Capítulo 4

Métodos para problemas no simétricos

En este capítulo abordamos el estudio de métodos clásicos para resolver problemas propios en matrices no simétricas, centrándonos principalmente en el proceso de Arnoldi y sus variantes. Mientras que algoritmos como el de Lanczos resultan eficientes para matrices simétricas, su extensión al caso no simétrico requiere herramientas matemáticas y algoritmos específicos.

El proceso de Arnoldi, introducido en 4.1 constituye una pieza fundamental en la reducción de matrices no simétricas a su forma de Hessenberg, lo que facilita la aproximación de los autovalores de la matriz original y, por tanto, su análisis espectral.

Se presentan también estrategias de rearranque del proceso de Arnoldi, tanto en su versión explícita como implícita, en las secciones 4.2 y 4.3 respectivamente. Estas técnicas resultan esenciales para reducir el coste computacional en problemas de gran dimensión, pues permiten reiniciar el algoritmo enfocando el cálculo en las partes más relevantes del espectro.

Finalmente, se introducen métodos más avanzados en las secciones 4.4, 4.5 y 4.6, como el algoritmo de Krylov–Schur y la tridiagonalización de Lanczos no simétrica. Estas variantes perfeccionan la eficiencia y precisión, permitiendo un cálculo más efectivo de los autovalores dominantes o deseados.

4.1. El proceso de Arnoldi básico

Arnoldi (1951) expuso una forma de extender el proceso de Lanczos a matrices no simétricas. La idea principal en la que se basaba este proceso es en la reducción de una matriz de Hessenberg.

Supongamos que $A \in \mathbb{R}^{n\times n}$ es una matriz no simétrica, siendo $H = Q^T AQ$ una matriz de Hessenberg reducida, donde $Q = [\, q_1 \,|\cdots|\, q_n \,]$. Busquemos una cierta analogía con el proceso de Lanczos simétrico estudiado en el capítulo anterior.

Comparando las columnas de las matrices en la igualdad $AQ = QH$, obtenemos

$$Aq_k = \sum_{i=1}^{k+1} h_{ik} q_i, \qquad 1 \leq k \leq n-1.$$

Aislando entonces el último término del sumatorio se tiene que

$$h_{k+1,k} q_{k+1} = Aq_k - \sum_{i=1}^{k} h_{ik} q_i \equiv r_k$$

donde $h_{ik} = q_i^T A q_k$, para $i \in \{1, ..., k\}$. En el caso en el que $r_k \neq 0$ se sigue que

$$q_{k+1} = \frac{r_k}{h_{k+1,k}},$$

donde $h_{k+1,k} = \|r_k\|_2$.

Estas ecuaciones definen el proceso de Arnoldi y observamos que se asemejan con las ecuaciones del proceso de Lanczos simétrico estudiado en el capítulo anterior (Tridiagonalización de Lanczos).

Obtenemos de esta forma el algoritmo siguiente:

Algorithm 6 Proceso de Arnoldi

Dados $A \in \mathbb{R}^{n\times n}$ y $q_1 \in \mathbb{R}^n$ un vector unitario (norma 2), calculamos una matriz ortogonal $Q_t = [\, q_1 \mid \cdots \mid q_t \,]$ y una matriz de Hessenberg superior $H_t \in \mathbb{R}^{t\times t}$ tales que $AQ_t = Q_t H_t$, donde $1 \leq t \leq n$.

$k = 0, r_0 = q_1, h_{10} = 1$
while $(h_{k+1,k} \neq 0)$ **do**
    $q_{k+1} = r_k / h_{k+1,k}$
    $k = k + 1$
    $r_k = Aq_k$
    for $i = 1 : k$ **do**
        $h_{ik} = q_i^T r_k$
        $r_k = r_k - h_{ik} q_i$
    end for
    $h_{k+1,k} = \|r_k\|_2$
end while
$t = k$

A los vectores q_k se les denomina *vectores de Arnoldi*. Estos vectores definen una base ortonormal para el subespacio de Krylov $\mathcal{K}(A, q_1, k)$.

La situación después de que se ejecuten k pasos del algoritmo recién definido viene resumida en la ecuación

$$AQ_k = Q_k H_k + r_k e_k^T, \tag{4.1}$$

donde $Q_k = [\, q_1 \mid \cdots \mid q_k \,], e_k = I_k(:, k)$, y

$$H_k = \begin{bmatrix} h_{11} & h_{12} & \cdots & \cdots & h_{1k} \\ h_{21} & h_{22} & \cdots & \cdots & h_{2k} \\ 0 & h_{32} & \ddots & \ddots & \vdots \\ \vdots & & \ddots & \ddots & \vdots \\ 0 & \cdots & \cdots & h_{k,k-1} & h_{kk} \end{bmatrix}.$$

Si $Q_k \in \mathbb{R}^{n\times k}$ tiene columnas ortonormales, $H_k \in \mathbb{R}^{k\times k}$ es de Hessenberg superior y $Q_k^T r_k = 0$, a cualquier descomposición de la forma de la ecuación definida en (4.1) se le denomina *k-paso de la descomposición de Arnoldi*.

En el caso en el que $y \in \mathbb{R}^k$ es un vector unitario de norma 2 para la matriz H_k y se tiene además que $H_k y = \lambda y$, se obtiene entonces de la ecuación (4.1) que

$$(A - \lambda I)x = (e_k^T y)r_k,$$

siendo $x = Q_k y$.

Dado que $r_k \in \mathcal{K}(A, q_1, k)^\perp$, se tiene que (λ, x) es un par de Ritz para la matriz A con respecto al subespacio de Krylov $\mathcal{K}(A, q_1, k)$. Nótese que si $v = (e_k^T y)r_k$, entonces

$$(A + E)x = \lambda x,$$

donde $E = -vx^T$, con $\|E\|_2 = |y_k|\|r_k\|_2$

Una característica que distingue el proceso de Arnoldi del proceso simétrico de Lanczos es que los vectores de Arnoldi $q_1, ..., q_k$ deben tenerse en cuenta en el k-ésimo paso. Además, el cálculo del vector q_{k+1} conlleva $\mathcal{O}(kn)$ operaciones de coma flotante (excluyendo las que corresponden al producto matricial Aq_k), lo que provoca que se generen secuencias de Arnoldi largas.

En al sección siguiente se estudia el método de Arnoldi con rearranque, donde se eligen cuidadosamente los reinicios y se controla una iteración máxima.

4.2. Arnoldi con rearranque

Ahora vamos a considerar ejecutar Arnoldi para m pasos para, a continuación, reiniciar la iteración con un nuevo vector inicial diferente. Este nuevo vector inicial, denotado por q_+ para diferenciarlo, se elige del espacio de los vectores de Arnoldi $q_1, ..., q_m$.
El vector q_+ es de la forma

$$q_+ = p(A)q_1,$$

para algún polinomio de grado $(m-1)$. Esto se debe a la conexión que tienen estos vectores $q_1, ..., q_m$ con el subespacio de Krylov $\mathcal{K}(A, q_1, k)$.

Con el fin de simplificar el estudio, vamos a considerar $A \in \mathbb{R}^{n\times n}$ diagonalizable y $Az_i = \lambda_i z_i$, para $i = 1 : n$.
Supongamos que el autovector q_1 tiene como expansión

$$q_1 = a_1 z_1 + \cdots + a_n z_n.$$

Entonces en este caso el vector de reinicio q_+ es un escalar múltiplo de

$$z = a_1 p(\lambda_1) z_1 + \cdots + a_n p(\lambda_n) z_n.$$

Eligiendo de manera cuidadosa al polinomio $p(\lambda)$ se puede diseñar el vector q_+ de tal forma que su componente en una cierta dirección del autovalor se enfatice mientras que en las direcciones de cualquier otro autovalor se desenfatice.

Veamos un ejemplo. Supongamos que

$$p(\lambda) = c \cdot (\lambda - \mu_1)(\lambda - \mu_2) \cdots (\lambda - \mu_p), \tag{4.2}$$

siendo c una constante.
En este caso, el vector q_+ es unitario en la dirección de

$$z = c \cdot \sum_{k=1}^{n} a_k \left(\prod_{i=1}^{p} (\lambda_k - \mu_i) \right) z_k.$$

Por lo tanto, el hecho de elegir un buen vector de reinicio q_+ de $\mathcal{K}(A, q_1, m)$ consiste en elegir un polinomio filtro $p(\lambda)$ que elimine aquellas partes del espectro no deseadas. Saad ([7], [8], [9]) desarrolló varias heurísticas basadas en vectores de Ritz ya calculados para realizar esta filtración.

4.3. Rearranque implícito

En esta sección describimos un proceso de rearranque de Arnoldi para calcular de manera implícita el polinomio filtro descrito en (4.2).

Para ello, supongamos que $H_c \in \mathbb{R}^{m \times m}$ es una matriz de Hessenberg superior, $\mu_1, ..., \mu_p$ son escalares y que la matriz H_+ se obtiene por medio de la iteración QR con cambios descrita como:

$$\begin{aligned}
&H^{(0)} = H_c \\
&\quad \textbf{for } i = 0 : p \\
&\qquad H^{(i-1)} - \mu_i I = V_i R_i \quad \text{(Dado QR)} \\
&\qquad H^{(i)} = R_i V_i + \mu_i I \\
&\quad \textbf{end} \\
&\quad H_+ = H^{(p)}
\end{aligned} \tag{4.3}$$

donde cada matriz $H^{(i)}$ es a su vez una matriz de Hessenberg superior.

En el caso en el que la matriz V viene determinada como

$$V = V_1 \cdots V_p, \tag{4.4}$$

entonces

$$H_+ = V^T H_c V. \tag{4.5}$$

A continuación estudiamos un teorema en el cual se muestra que el polinomio filtro (4.2) guarda relación con la iteración QR descrita en (4.3).

Teorema 4.1. *Si las matrices $V = V_1 \cdots V_p$ y $R = R_p \cdots R_1$ están definidas mediante (4.3), entonces*

$$VR = (H_c - \mu_1 I) \cdots (H_c - \mu_p I). \tag{4.6}$$

Demostración. Razonamos por inducción sobre p.

- Para $p = 1$ se tiene que

  $$VR = V_1 R_1 = (H_c - \mu_1 I),$$

  igualdad que es cierta por definición de las matrices V_1 y R_1.

- Supongamoslo cierto para $p - 1$, es decir, supongamos que

  $$\tilde{V}\tilde{R} = (H_c - \mu_1 I) \cdots (H_c - \mu_p I), \tag{H.I}$$

  siendo $\tilde{V} = V_1 \cdots V_{p-1}$ y $\tilde{R} = R_{p-1} \cdots R_1$.

- Veámoslo para p.

  $$\begin{aligned} VR = \tilde{V}(V_p R_p)\tilde{R} = \tilde{V}(H^{(p-1)} - \mu_p I)\tilde{R} = \tilde{V}(\tilde{V}^T H_c \tilde{V} - \mu_p I)\tilde{R} = \\ = (H_c - \mu_p I)\tilde{V}\tilde{R} = (H_c - \mu_p I)(H_c - \mu_1 I) \cdots (H_c - \mu_{p-1} I), \end{aligned}$$

  donde hemos usado que $H^{(p-1)} = \tilde{V}^T H_c \tilde{V}$.

□

Nótese que la matriz R en (4.6) es triangular superior por lo tanto

$$V(:,1) = p(H_c)e_1$$

donde $p(\lambda)$ es el polinomio filtro definido mediante la ecuación (4.2) y con constante $c = 1/R(1,1)$.

Supongamos ahora que se han realizado m pasos de la iteración de Arnoldi con rearranque del vector q_1.
La factorización de Arnoldi (4.1) dice que tenemos una matriz de Hessenberg superior $H_c \in \mathbb{R}^{m\times m}$ y una matriz $Q_c \in \mathbb{R}^{n\times m}$ con columnas ortonormales tales que

$$AQ_c = Q_c H_c + r_c e_m^T. \tag{4.7}$$

Observemos que $Q_c(:,1) = q_1$ y $r_c \in \mathbb{R}^n$ son tales que $Q_c^T r_c = 0$. Si aplicamos la factorización QR (4.3) a la matriz H_c, y empleamos las ecuaciones (4.4) y (4.5) se transforma la factorización de Arnoldi anterior en la siguiente

$$AQ_+ = Q_+ H_+ + r_c e_m^T V \tag{4.8}$$

donde

$$Q_+ = Q_c V.$$

Si suponemos que el vector de rearranque q_+ es la primera columna de esta matriz Q_+, entonces se tiene que

$$q_+ = Q_+(:,1) = Q_c V(:,1) = c \cdot Q_c(H_c - \mu_1 I)\cdots(H_c - \mu_p I)e_1.$$

La ecuación (4.8) implica que

$$(A - \mu I)Q_c e_1 = Q_c(H_c - \mu I)e_1$$

para cualquier $\mu \in \mathbb{R}$ y en consecuencia

$$q_+ = c(A - \mu I)\cdots(A - \mu_p I)Q_c e_1 = p(A)q_1.$$

Esto sugiere el siguiente marco para reinicios repetidos:

Repetir:

Con el vector inicial q_1, realizamos m pasos de la iteración de Arnoldi obteniendo así $Q_c \in \mathbb{R}^{n\times m}$ y $H_c \in \mathbb{R}^{m\times m}$.

Determinar los valores filtro $\mu_1, ..., \mu_p$.

Realizados p pasos de la iteración QR desplazada (4.3) obtenemos la matriz de Hessenberg H_+ y la matriz ortogonal V.

Reemplazar q_1 con la primera columna de $Q_c V$.

(4.9)

Vamos a tratar de hacer esto mejor. Cada una de las matrices ortogonales $V_1, ..., V_p$ que surgen en el algoritmo descrito en (4.2) son de Hessenberg superior. De este modo, V tiene menor ancho de banda p, y lo mismo ocurre con $V(m, 1 : m - p - 1) = 0$.
Si $j = m - p$ se sigue de (4.8) que

$$AQ_+(:, 1 : j) = Q_+(:, 1 : j)H_+(1 : j, 1 : j) + v_{mj}r_c e_j$$

es un j-ésimo paso de la descomposición de Arnoldi, es decir, podemos ya realizar el j+1 paso de la iteración de Arnoldi empleando el vector inicial q_+, sin necesidad de lanzar el rearranque desde el primer paso. Esto nos lleva a modificar (4.9) de la forma:

Con un vector inicial q_1, realizamos m pasos de la iteración de Arnoldi obteniendo $Q_c \in \mathbb{R}^{n\times m}, H_c \in \mathbb{R}^{m\times m}$ y $r_c \in \mathbb{R}^n$ de manera que $AQ_c = Q_c H_c + r_c e_m^T$.

Repetir:

Determinar los valores filtro $\mu_1, ..., \mu_p$.

Realizados p pasos de la iteración QR desplazada (4.3) aplicada a H_c obteniendo $H_+ \in \mathbb{R}^{m\times m}$ y $V = (v_{ij}) \in \mathbb{R}^{m\times m}$.

Reemplazar Q_c con las primeras j columnas de $Q_c V$.

Reemplazar r_c con $v_{mj}r_c$.

Empezando con $AQ_c = Q_c H_c + r_c e_j^T$, realizamos $j + 1, ..., j + p = m$ pasos de la iteración de Arnoldi, obteniendo$AQ_m = Q_m H_m + r_m e_m^T$.

Fijamos $Q_c = Q_m, H_c = H_m$ y $r_c = r_m$.

Los valores filtro $\mu_1, ..., \mu_p$ deben elegirse en la proximidad de los autovalores "no deseados" de A, haciendo que sea posible formular heurísticas útiles basadas en los autovalores de la matriz de Hessenberg $H_+ \in \mathbb{R}^{m\times m}$.

En ARPACK ([3]) se detalla una implementación más profunda del método.

4.4. El algoritmo de Krylov - Schur

Vamos a introducir ahora un método de rearranque alternativo al estudiado en la sección anterior. La idea principal en la que se basa es en la descomposición de Schur de una matriz de Hessenberg de manera ordenada.

Supongamos que H_m es la matriz de Hessenberg a descomponer, generada después de m pasos de la iteración de Arnoldi, y supongamos que hemos calculado el producto

$$AQ_m = Q_m H_m + r_m e_m^T,$$

con $m = j + p$, siendo j el número de autovalores de A que se desea calcular. Sea

$$U^T H_m U = \begin{bmatrix} T_{11} & T_{12} \\ 0 & T_{22} \end{bmatrix}$$

la descomposición de Schur de la matriz A, y asumamos que los autovalores se piden de manera que nos interesan los autovalores de $T_{11} \in \mathbb{R}^{j \times j}$ mientras que los de la matriz $T_{22} \in \mathbb{R}^{p \times p}$ no.

En este caso, transformamos la descomposición de Arnoldi arriba descrita en

$$AQ_+ = Q_+ T + r_c e_m^T U$$

donde $Q_+ = Q_m U$. Se sigue que

$$AQ_+(:, 1:j) = Q_+(:, 1:j) T_{11} + r_m u^T$$

donde $U^T = U(m, 1:j)$.

Se puede determinar una matriz ortogonal $Z \in \mathbb{R}^{j \times j}$ tal que $Z^T T_{11} Z$ es de Hessenberg superior y $Z^T u = \tau e_j$. Entonces

$$A(Q_+ Z) = (Q_+ Z)(Z^T T_{11} Z) + r_c (Z^T u)^T$$

es un j-ésimo paso de la factorización de Arnoldi.

Entonces, establecemos Q_j, H_j y r_j como $Q_+ Z, Z^T T_{11} Z$ y τr_m respectivamente y ejecutamos de $j + 1$ a $j + p = m$ pasos de Arnoldi.

4.5. Tridiagonalización de Lanczos no simétrica

Otra manera de extender el proceso simétrico de Lanczos es reduciendo A a una forma tridiagonal empleando para ello una trasformación similar.

Supongamos que $A \in \mathbb{R}^{n\times n}$ y que Q es una matriz existente no singular tal que

$$Q^{-1}AQ = T = \begin{bmatrix} \alpha_1 & \gamma_1 & & \cdots & 0 \\ \beta_1 & \alpha_2 & \ddots & & \vdots \\ & \ddots & \ddots & \ddots & \\ \vdots & & \ddots & \ddots & \gamma_{n-1} \\ 0 & \cdots & & \beta_{n-1} & \alpha_n \end{bmatrix},$$

siendo

$$Q = [\, q_1 \,|\cdots|\, q_n \,],$$
$$Q^{-T} = \tilde{Q} = [\, \tilde{q}_1 \,|\cdots|\, \tilde{q}_n].$$

Comparando columnas en $AQ = QT$ y $A^T\tilde{Q} = \tilde{Q}T^T$, se tiene que

$$\begin{aligned} Aq_k &= \gamma_{k-1}q_{k-1} + \alpha_k q_k + \beta_k q_{k+1}, & \gamma_0 q_0 &\equiv 0, \\ A^T\tilde{q}_k &= \beta_{k-1}\tilde{q}_{k-1} + \alpha_k\tilde{q}_k + \gamma_k\tilde{q}_{k+1}, & \beta_0\tilde{q}_0 &\equiv 0, \end{aligned}$$

para $k = 1 : (n-1)$.
Estas ecuaciones junto con la *condición de biortogonalidad*

$$\tilde{Q}^T Q = I_n$$

implican que

$$\alpha_k = \tilde{q}_k^T A q_k$$

y

$$\begin{aligned} \beta_k q_{k+1} \equiv r_k &= (A - \alpha_k I)q_k - \gamma_{k-1}q_{k-1}, \\ \gamma_k\tilde{q}_{k+1} \equiv \tilde{r}_k &= (A - \alpha_k I)^T\tilde{q}_k - \beta_{k-1}\tilde{q}_{k-1}. \end{aligned}$$

Los factores β_k y γ_k se pueden elegir con una cierta flexibilidad. Nótese que

$$1 = \tilde{q}_{k+1}^T q_{k+1} = (\tilde{r}_k/\gamma_k)^T(r_k/\beta_k).$$

Podemos entonces afirmar que una vez esté β_k especificado, γ_k viene dado por

$$\gamma_k = \tilde{r}_k^T r_k/\beta_k.$$

Con la elección "canónica" $\beta_k = \|r_k\|_2$ obtenemos el siguiente algoritmo de tridiagonalización de Lanczos para matrices no simétricas

$q_1, \tilde{q}_1$ vectores unitarios de norma 2 dados, con $\tilde{q}_1^T 1_1 \neq 0$
$k = 0, q_0 = 0, r_0 = q_1, \tilde{q}_0 = 0, s_0 = \tilde{q}_1$
Repetir:
while $(r_k \neq 0)$ y $(\tilde{r}_k \neq 0)$ y $(\tilde{r}_k^T r_k \neq 0)$
$\beta_k = \|r_k\|_2$
$\gamma_k = \tilde{r}_k^T r_k / \beta_k$
$q_{k+1} = r_k / \beta_k$ (4.10)
$\tilde{q}_{k+1} = \tilde{r}_k / \gamma_k$
$k = k + 1$
$\alpha_k = \tilde{q}_k^T A q_k$
$r_k = (A - \alpha_k I) q_k - \gamma_{k-1} q_{k-1}$
$\tilde{r}_k = (A - \alpha_k I)^T \tilde{q}_k - \beta_{k-1} \tilde{q}_{k-1}$
end

Si por ejemplo la matriz inicial T_k ya es de la forma

$$T_k = \begin{bmatrix} \alpha_1 & \gamma_1 & & \cdots & 0 \\ \beta_1 & \alpha_2 & \ddots & & \vdots \\ & \ddots & \ddots & \ddots & \\ \vdots & & \ddots & \ddots & \gamma_{k-1} \\ 0 & \cdots & & \beta_{k-1} & \alpha_k \end{bmatrix},$$

la situación al principio del bucle se resume en las siguientes ecuaciones

$$A \,[\, q_1 \,|\cdots|\, q_k \,] = [\, q_1 \,|\cdots|\, q_k \,]\, T_k + r_k e_k^T, \tag{4.11}$$

$$A^T \,[\, \tilde{q}_1 \,|\cdots|\, \tilde{q}_k \,] = [\, \tilde{q}_1 \,|\cdots|\, \tilde{q}_k \,]\, T_k^T + \tilde{r}_k e_k^T. \tag{4.12}$$

Tenemos en esta situación tres casos.

- Si $\underline{r_k = 0}$, entonces la iteración termina y $\text{span}\{q_1, ..., q_k\}$ es un subespacio invariante para A.
- Si $\underline{\tilde{r}_k = 0}$, entonces la iteración también termina y $\text{span}\{\tilde{q}_1, ..., \tilde{q}_k\}$ es un subespacio invariante para A^T.

- Si por el contrario ninguna de estas condiciones es cierta y $\tilde{r}_k^T r_k = 0$, entonces el proceso de tridiagonalización termina sin ninguna información sobre ningún subespacio invariante. A este se le conoce como *avería grave*.

4.6. La idea de mirar hacia adelante

Examinemos ahora el problema de la avería grave que se produce en (4.10).

Vamos a asumir que $A \in \mathbb{R}^{n\times n}$ con $n = rp$, y consideremos la factorización en la cual queremos $\tilde{Q}^T Q = I_n$:

$$\tilde{Q}^T A Q = \begin{bmatrix} M_1 & C_1^T & & \cdots & 0 \\ B_1 & M_2 & \ddots & & \vdots \\ & \ddots & \ddots & \ddots & \\ \vdots & & \ddots & \ddots & C_{r-1}^T \\ 0 & \cdots & & B_{r-1} & M_r \end{bmatrix}, \tag{4.13}$$

donde todos los bloques tienen dimensión $p \times p$.
Sean $Q = [\, Q_1 \,|\cdots|\, Q_r \,]$ y $\tilde{Q} = \left[\, \tilde{Q}_1 \,|\cdots|\, \tilde{Q}_r \right]$ particiones de Q y $\tilde{Q}$ respectivamente.
Comparando por bloques las columnas en las igualdades $AQ = QT$ y $A^T\tilde{Q} = \tilde{Q}T^T$, obtenemos las ecuaciones

$$\begin{aligned} Q_{k+1}B_k &= AQ_k - Q_k M_k - Q_{k-1}C_{k-1}^T \equiv R_k, \\ \tilde{Q}_{k+1}C_k &= A^T\tilde{Q}_k - \tilde{Q}_k M_k^T - \tilde{Q}_{k-1}B_{k-1}^T \equiv S_k. \end{aligned}$$

Nótese que

$$M_k = \tilde{Q}_k^T A Q_k.$$

Si la matriz S_k dada por $S_k^T R_k = C_k^T \tilde{Q}_{k+1}^T Q_{k+1} B_k \in \mathbb{R}^{p\times p}$ es no singular y calculamos $B_k, C_k \in \mathbb{R}^{p\times p}$ tales que

$$C_k^T B_k = S_k^T R_k,$$

entonces

$$Q_{k+1} = R_k B_k^{-1}, \tag{4.14}$$

$$\tilde{Q}_{k+1} = S_k C_k^{-1} \tag{4.15}$$

satisfacen que $\tilde{Q}_{k+1}^T Q_{k+1} = I_p$.

En el entorno que acabamos de describir se asocia una avería grave con la presencia de un $S_k^T R_k$ singular.

El problema de ruptura en (4.10) puede resolverse con una factorización de la forma (4.13) en la cual los tamaños de los bloques se determinan de forma dinámica. De forma genérica, las matrices Q_{k+1} y $\tilde{Q}_{k+1}$ se construyen columna a columna con recursiones especiales que culminan en la producción de la matriz no singular $\tilde{Q}_{k+1}^T Q_{k+1}$. Los cálculos están organizados de manera que las condiciones de ortogonalidad $\tilde{Q}_i^T Q_{k+1} = 0$ y $Q_i^T \tilde{Q}_{k+1} = 0$ se cumplen para $i = 1 : k$.

Los métodos de esta forma pertenecen a la familia de los *métodos de Lanczos de anticipación*.

Bibliografía

[1] G. H. Golub, R. Underwood, and J. H. Wilkinson. The lanczos algorithm for the symmetric $Ax = \lambda Bx$ problem. Technical Report STAN-CS-72-270, Department of Computer Science, Stanford University, Stanford, CA, 1972.

[2] Gene H. Golub and Charles F. Van Loan. *Matrix Computations*. Johns Hopkins University Press, Baltimore, 4th edition, 2013.

[3] Richard B. Lehoucq, Danny C. Sorensen, and Chao Yang. *ARPACK Users' Guide: Solution of Large-Scale Eigenvalue Problems with Implicitly Restarted Arnoldi Methods*. Society for Industrial and Applied Mathematics, Philadelphia, 1998.

[4] C. C. Paige. Computational variants of the lanczos method for the eigenproblem. *Journal of the Institute of Mathematics and Its Applications*, 10:373–381, 1972.

[5] C. C. Paige. Error analysis of the lanczos algorithm for tridiagonalizing a symmetric matrix. *Journal of the Institute of Mathematics and Its Applications*, 18:341–349, 1976.

[6] C. C. Paige. Accuracy and effectiveness of the lanczos algorithm for the symmetric eigenproblem. *Linear Algebra and its Applications*, 34:235–258, 1980.

[7] Yousef Saad. Variations of arnoldi's method for computing eigenelements of large unsymmetric matrices. *Linear Algebra and its Applications*, 34:269–295, 1980.

[8] Yousef Saad. Chebyshev acceleration techniques for solving nonsymmetric eigenvalue problems. *Mathematics of Computation*, 42:567–588, 1984.

[9] Yousef Saad. *Numerical Methods for Large Eigenvalue Problems*. Algorithms and Architectures for Advanced Scientific Computing. Manchester University Press, Manchester, 1992.

[10] Henk A. van der Vorst. *Iterative Krylov Methods for Large Linear Systems*. Cambridge University Press, Cambridge, 2003.