

Métodos numéricos para la determinación de autovalores

Pablo J. Santamaría

Introducción

Muchos problemas de interés conducen al cálculo, o por lo menos a la estimación, de los valores característicos o *autovalores* de una matriz asociada con un sistema lineal de ecuaciones. Algebraicamente el problema consiste en, dada una matriz real $n \times n$ \mathbf{A} , encontrar los escalares (generalmente complejos) λ (los *autovalores* de \mathbf{A}) y los vectores *no nulos* \mathbf{x} (los *autovectores* de \mathbf{A} asociados a λ) tales que

$$\mathbf{A}\mathbf{x} = \lambda\mathbf{x}, \quad \mathbf{x} \neq \mathbf{0}$$

Sabemos que tal matriz \mathbf{A} tiene precisamente n autovalores, no necesariamente distintos, que son las raíces del *polinomio característico* de grado n

$$p(\lambda) = \det(\mathbf{A} - \lambda\mathbf{I})$$

Así pues, formalmente, los autovalores de \mathbf{A} se pueden obtener encontrando las n raíces de $p(\lambda)$. En la práctica esto puede llevarse a cabo con matrices o bien de pequeño tamaño o bien de formas particulares. En el caso general el polinomio característico es difícil de obtener y, de cualquier modo, sabemos que la determinación de las raíces de un polinomio de grado n -ésimo es también un problema difícil, puesto que, excepto para valores pequeños de n , no es un problema *cerrado* (esto es, no hay fórmulas explícitas). En consecuencia es necesario considerar algoritmos que permitan determinar sistemáticamente *todos* los autovalores de una matriz dada en una forma *eficiente*, esto es, con el menor número de operaciones posibles, y además, dado que operar con grandes matrices involucra muchas sustracciones aritméticas, tales algoritmos deben ser *estables* de manera que los resultados numéricos no sean en realidad números aleatorios en lugar de los autovalores deseados.

La estrategia práctica a considerar es dividir el problema en dos partes: en primer lugar se reduce la matriz original \mathbf{A} a una matriz $\mathbf{B} = (b_{i,j})$ de una estructura más simple (en el sentido de que posea la mayor cantidad de elementos nulos posibles) pero con los *mismos* autovalores; y luego se calculan los autovalores de esta nueva matriz. Típicamente, las matrices buscadas son ya sea una matriz *tridiagonal*

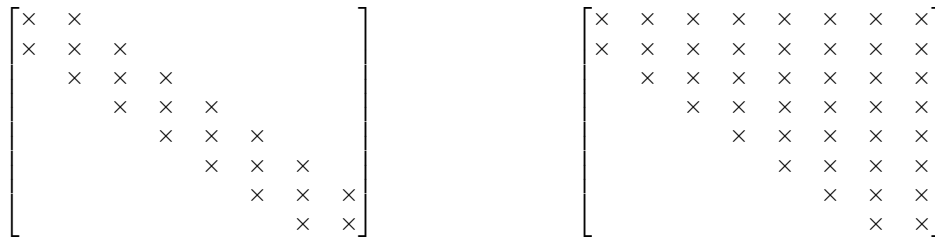
$$b_{ij} = 0, \quad \text{para } |i - j| > 1$$

o una *matriz de Hessenberg (superior)*

$$b_{ij} = 0, \quad \text{para } i \leq j + 2$$

cuyas formas se ilustran en la fig. 0.1. Una matriz tridiagonal tiene elementos nulos fuera de las diagonales principal, superior e inferior. Una matriz de Hessenberg tiene elementos nulos debajo de la diagonal inferior.

La utilidad de esta estrategia dual resulta clara si se considera la cantidad de cómputo realizada en la manipulación de una matriz grande (digamos $n = 100$ o 200), tal como la multiplicación por



(a) Una matriz tridiagonal

(b) Una matriz de Hessenberg

Figura 0.1.

otra matriz. Para una matriz *densa* (esto es, con pocos elementos nulos) el número de operaciones requeridas es del orden de n^3 , en tanto que para una matriz de Hessenberg tal cálculo requiere en realidad del orden de n^2 operaciones y para una matriz tridiagonal el número requerido es de orden n . Así para una matriz grande la diferencia entre n^2 y n^3 es apreciable (¡ sin mencionar n versus n^3 !) y por lo tanto la búsqueda de los autovalores de una matriz simplificada resultará más eficiente. Dado que podemos reducir una matriz dada a la forma de Hessenberg (o a la forma tridiagonal en ciertos casos) mediante un número *finito* de pasos aritméticos, la estrategia diseñada resulta de suma utilidad práctica.

Como las dos etapas de nuestra estrategia son (casi) independientes vamos a discutir las separadamente. Así trataremos en primer lugar la posibilidad de reducir una matriz dada a una forma más simple. Aquí es donde las cuestiones de la forma de proceder y estabilidad de los métodos resultan cruciales.

Reducción de matrices a una forma más simple

Los métodos para reducir una matriz general dada a una forma más simple se basan en las *transformaciones de similitud o semejanza*.

DEFINICIÓN

Dos matrices $n \times n$ \mathbf{A} y \mathbf{B} se dicen *semejantes* si existe una matriz \mathbf{T} $n \times n$ no singular tal que

$$\mathbf{B} = \mathbf{T}^{-1}\mathbf{A}\mathbf{T}$$

La transformación de similitud tiene la propiedad de preservar los autovalores. En efecto, a partir de la ecuación de autovalores

$$\mathbf{A}\mathbf{x} = \lambda\mathbf{x}$$

premultiplicando por \mathbf{T}^{-1} tenemos que

$$\mathbf{T}^{-1}\mathbf{A}\mathbf{x} = \lambda\mathbf{T}^{-1}\mathbf{x}$$

Si definimos

$$\mathbf{T}^{-1}\mathbf{x} = \mathbf{y}, \quad \text{con lo cual } \mathbf{x} = \mathbf{T}\mathbf{y}$$

obtenemos

$$\mathbf{T}^{-1}\mathbf{A}\mathbf{T}\mathbf{y} = \lambda\mathbf{T}^{-1}\mathbf{T}\mathbf{y} = \lambda\mathbf{y}$$

es decir tenemos el nuevo problema de autovalores

$$\mathbf{B}\mathbf{y} = \lambda\mathbf{y}$$

para la matriz \mathbf{B} que es semejante a la matriz \mathbf{A} , donde los nuevos autovectores \mathbf{y} están relacionados en forma simple con los de \mathbf{A} , pero los autovalores *son los mismos*. Este resultado se sigue también del hecho de que matrices semejantes tienen el mismo polinomio característico. En efecto,

$$\begin{aligned} p_{\mathbf{B}}(\lambda) &= \det(\mathbf{B} - \lambda\mathbf{I}) \\ &= \det(\mathbf{T}^{-1}\mathbf{A}\mathbf{T} - \lambda\mathbf{T}^{-1}\mathbf{T}) \\ &= \det(\mathbf{T}^{-1}(\mathbf{A} - \lambda\mathbf{I})\mathbf{T}) \\ &= \det\mathbf{T}^{-1} \det(\mathbf{A} - \lambda\mathbf{I}) \det\mathbf{T} \\ &= \det(\mathbf{A} - \lambda\mathbf{I}) = p_{\mathbf{A}}(\lambda) \end{aligned}$$

La clase de transformaciones de similitud es muy amplia. Dentro de ella está incluida un tipo de transformación mucho más restrictivo, la *transformación ortogonal*. Aquí la matriz de transformación es *ortogonal*, esto es, su inversa es su traspuesta

$$\mathbf{T}^{-1} = \mathbf{T}^t$$

Una transformación ortogonal no solo preserva los autovalores sino también la simetría si ésta existe. En efecto, si \mathbf{A} es una matriz *simétrica* y \mathbf{B} es semejante a \mathbf{A} por una transformación ortogonal, la traspuesta de \mathbf{B} es

$$\mathbf{B}^t = (\mathbf{T}^t\mathbf{A}\mathbf{T})^t = \mathbf{T}^t\mathbf{A}^t\mathbf{T} = \mathbf{T}^t\mathbf{A}\mathbf{T} = \mathbf{B}$$

es decir, \mathbf{B} es también simétrica. Recuérdese también que si una matriz real es simétrica sus autovalores serán necesariamente números reales y no complejos como en el caso general.

La estrategia general de los métodos para calcular los autovalores de una matriz \mathbf{A} es realizar primero una sucesión de transformaciones de semejanza

$$\begin{aligned} \mathbf{A}^{(0)} &= \mathbf{A} \\ \mathbf{A}^{(i)} &= \mathbf{T}_i^{-1}\mathbf{A}^{(i-1)}\mathbf{T}_i, \quad i = 1, 2, \dots, m \end{aligned}$$

para gradualmente transformar la matriz \mathbf{A} en una matriz \mathbf{B} de forma más simple

$$\mathbf{B} = \mathbf{A}_m = \mathbf{T}^{-1}\mathbf{A}\mathbf{T}, \quad \mathbf{T} = \mathbf{T}_1\mathbf{T}_2 \dots \mathbf{T}_m$$

y entonces determinar los autovalores λ de \mathbf{B} (que serán necesariamente los de \mathbf{A}). La matriz \mathbf{B} es escogida de manera tal que

- La determinación de los autovalores de \mathbf{B} sea lo más simple posible y
- El problema de autovalores para \mathbf{B} no esté (sustancialmente) peor condicionado que el de \mathbf{A} (esto es, que pequeños cambios en la matriz \mathbf{B} no empeoren los autovalores más que de lo que lo hacen pequeños cambios en \mathbf{A}).

Si $\Delta\mathbf{A}$ y $\Delta\mathbf{B}$ son las matrices que contienen a las perturbaciones en los coeficientes de \mathbf{A} y \mathbf{B} , dado que

$$\begin{aligned} \mathbf{B} &= \mathbf{T}^{-1}\mathbf{A}\mathbf{T} \\ \mathbf{B} + \Delta\mathbf{B} &= \mathbf{T}^{-1}(\mathbf{A} + \Delta\mathbf{A})\mathbf{T} \end{aligned}$$

tenemos que

$$\Delta\mathbf{B} = \mathbf{T}^{-1}\Delta\mathbf{A}\mathbf{T}, \quad \text{y por lo tanto } \Delta\mathbf{A} = \mathbf{T}\Delta\mathbf{B}\mathbf{T}^{-1}$$

Así para una norma vectorial $\|\cdot\|$ y su correspondiente norma matricial subordinada obtenemos las siguientes estimaciones

$$\begin{aligned} \|\mathbf{B}\| &\leq \text{cond } \mathbf{T} \|\mathbf{A}\| \\ \|\Delta\mathbf{A}\| &\leq \text{cond } \mathbf{T} \|\Delta\mathbf{B}\| \end{aligned}$$

donde $\text{cond } \mathbf{T} = \|\mathbf{T}\| \|\mathbf{T}^{-1}\|$ es el *número de condición* de la matriz \mathbf{T} . Así

$$\frac{\|\Delta \mathbf{A}\|}{\|\mathbf{A}\|} \leq (\text{cond } \mathbf{T})^2 \frac{\|\Delta \mathbf{B}\|}{\|\mathbf{B}\|}$$

Entonces es de esperar que para $\text{cond } \mathbf{T} \gg 1$ el problema para \mathbf{B} estará peor condicionando que el problema para \mathbf{A} . Puesto que

$$\text{cond } \mathbf{T} = \text{cond}(\mathbf{T}_1 \dots \mathbf{T}_m) \leq \text{cond } \mathbf{T}_1 \dots \text{cond } \mathbf{T}_m$$

un buen condicionamiento estará asegurado si escogemos matrices de transformación \mathbf{T}_i tales que $\text{cond } T_i$ no sea demasiado grande ¹. Este es el caso, en particular, para la norma máxima $\|x\|_\infty = \max_{1 \leq i \leq n} |x_i|$ ² y las matrices de *eliminación* de la forma

$$\mathbf{T}_i = \mathbf{G}_j = \begin{bmatrix} 1 & & & & 0 \\ & \ddots & & & \\ & & 1 & & \\ & & l_{j+1,j} & \ddots & \\ & & \vdots & \ddots & \\ 0 & & l_{nj} & & 0 & 1 \end{bmatrix}, |l_{kj}| \leq 1$$

la cual es una matriz triangular inferior que difiere de la identidad en la columna j -ésima y cuya inversa es

$$\mathbf{G}_j^{-1} = \begin{bmatrix} 1 & & & & 0 \\ & \ddots & & & \\ & & 1 & & \\ & & -l_{j+1,j} & \ddots & \\ & & \vdots & \ddots & \\ 0 & & -l_{nj} & & 0 & 1 \end{bmatrix}$$

Para estas matrices

$$\text{cond}_\infty \mathbf{T}_i \leq 4$$

Para la norma euclídeana $\|x\|_2 = \sqrt{x^t x}$ ³ un ejemplo son las matrices ortogonales \mathbf{T}_i para las cuales $\text{cond}_2 \mathbf{T}_i = 1$ ⁴. De acuerdo a lo anterior, es de esperar que algoritmos basados en transformaciones ortogonales sean muy estables (como contrapartida estos algoritmos requieren más aritmética que aquellos basados en transformaciones de semejanza generales).

¹Recuérdese que $\text{cond } \mathbf{A} \leq 1$ para cualquier matriz no singular \mathbf{A} . En efecto, $1 = \|\mathbf{I}\| = \|\mathbf{A}\mathbf{A}^{-1}\| \leq \|\mathbf{A}\| \|\mathbf{A}^{-1}\| = \text{cond } \mathbf{A}$.

²Cuya norma matricial subordinada es $\|\mathbf{A}\| = \max_{1 \leq i \leq n} \sum_{j=1}^n |a_{ij}|$.

³Cuya norma matricial subordinada es $\|\mathbf{A}\|_2 = \mu_1$ donde μ_1 es el máximo valor de las raíces cuadradas no negativas de los autovalores (necesariamente reales y no negativos) de la matriz simétrica semidefinida positiva $\mathbf{A}^t \mathbf{A}$.

⁴En efecto, si \mathbf{T} es ortogonal $\mathbf{T}\mathbf{T}^t = \mathbf{I}$ y como los autovalores de la matriz identidad son 1 (con multiplicidad n), $\text{cond}_2 T = \|\mathbf{T}\| \|\mathbf{T}^t\| = 1 \times 1 = 1$.

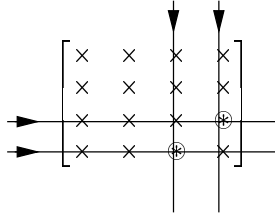


Figura 0.2.

la matriz $A^{(i)}$ converge a una matriz diagonal, cuyos elementos son precisamente los autovalores buscados. Dado que este proceso es infinito, en una implementación práctica uno debería iterar hasta una matriz $A^{(i)}$ cuya suma de los módulos de los elementos fuera de la diagonal

$$\sum_{i=1}^n \sum_{\substack{j=1 \\ (j \neq i)}}^n |a_{ij}^{(i)}|$$

sea suficientemente pequeña dentro de la precisión de los cálculos. Los autovalores de \mathbf{A} pueden aproximarse entonces por las componentes diagonales de \mathbf{A} .

Sin embargo, este es un proceso generalmente lento; en la práctica se requieren de alrededor de $6n^3$ multiplicaciones antes de considerar los elementos fuera de la diagonal suficientemente pequeños, y con matrices grandes esto es mucho trabajo. Otras estrategias que consideraremos que pronto consideraremos realizan una mejor labor.

Mencionemos también que el método de Jacobi puede ser implementado de manera de llevar una matriz no simétrica a una matriz *triangular*, proceso que, en general, requerirá de alrededor de $12n^3$ multiplicaciones antes de que converga.

Reducción de Givens

Aunque la transformación de Jacobi no es útil en la forma presentada, una pequeña modificación permite una estrategia *finita* para reducir una matriz simétrica \mathbf{A} a una forma tridiagonal. En lugar de producir un cero en las “esquinas” del cuadrado que defina la rotación en el plano, se escoge el parámetro c de manera de llevar a cero un elemento que *no* esté localizado en una de estas cuatro esquinas, esto es, que no estén en las posiciones a_{jj}, a_{jk}, a_{kk} . Específicamente, tomamos primero

$$\mathbf{T}_1 = \mathbf{\Omega}_{23}$$

de manera de anular el elemento en la posición a_{31} (si la matriz es simétrica esto anulará también a a_{13}). Después elegimos $\mathbf{T}_1 = \mathbf{\Omega}_{23}$ de manera de anular el elemento en la posición a_{41} , etc. En general, elegimos la sucesión de transformaciones \mathbf{T}_i dados por

$$\begin{array}{cccc} \mathbf{\Omega}_{23}, & \mathbf{\Omega}_{24}, & \dots, & \mathbf{\Omega}_{2n} \\ & \mathbf{\Omega}_{34}, & \dots, & \mathbf{\Omega}_{3n} \\ & & & \vdots \\ & & & \mathbf{\Omega}_{n-1,n} \end{array}$$

de manera tal que $\mathbf{\Omega}_{jk}, j = 2, \dots, n-1, k = j+1, j+2, \dots, n$ anule el elemento de la posición $a_{k,j-1}$. En la figura 0.3 se ilustra el procedimiento para una matriz 4×4 .

El método funciona debido a que los elementos en las posiciones fuera del cuadrado que define la rotación $\mathbf{\Omega}_{jk}$ (esto es, a_{rj} y a_{rk} con $r \neq j, r \neq k$) son simples combinaciones lineales de los valores

anteriores situados en la misma posición ⁶. Es esta simple combinación lineal la que preserva los ceros obtenidos en el paso anterior, ya que si estos elementos eran ceros, permanecerán nulos en la siguiente transformación.

Así pues, después de $(n-1)(n-2)/2 \sim n^2/2$ rotaciones, una matriz simétrica es reducida a una forma tridiagonal. Dado que el método involucra solo matrices ortogonales es estable, y requiere de aproximadamente $4n^3/3$ multiplicaciones. Esta técnica es, pues, eficiente, la única desventaja es que existe una técnica de estrategia similar que realiza la misma tarea con la *mitad* de trabajo. Tal método será discutido en la siguiente sección.

Finalmente mencionemos que la técnica de Givens puede ser implementada para llevar una matriz no simétrica a la forma de Hessenberg. En este caso el número de multiplicaciones realizadas en el proceso es alrededor de $10n^3/3$.

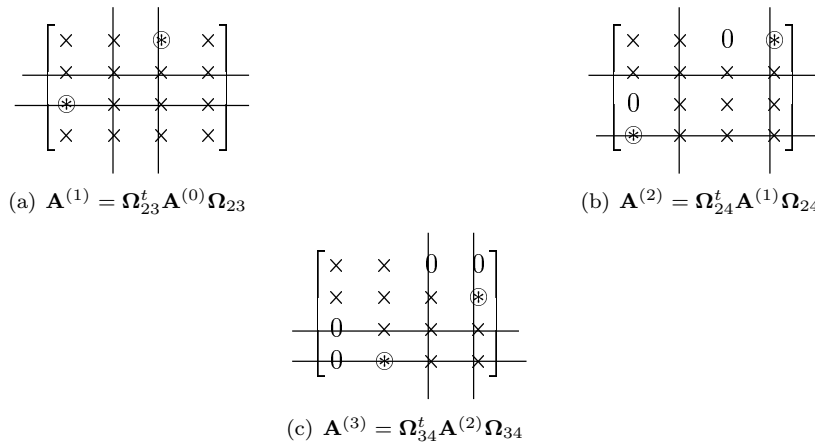


Figura 0.3.

Reducción de Householder

Al igual que la técnica de Givens, el algoritmo de Householder produce ceros a partir de transformaciones ortogonales. Sin embargo, distinto de Givens, cada transformación está diseñada para llevar a cero parte de toda una columna (y su correspondiente fila si la matriz es simétrica) de una vez.

El ingrediente básico del algoritmo es una matriz ortogonal simétrica \mathbf{P} , llamada *matriz de Householder*, la cual tiene la forma

$$\mathbf{P} = \mathbf{I} - 2\mathbf{w}\mathbf{w}^t$$

donde \mathbf{w} es un vector real con $\|\mathbf{w}\|_2^2 = \mathbf{w}^t\mathbf{w} = 1$ ⁷. Que \mathbf{P} es simétrica resulta claro de inmediato

$$\begin{aligned} \mathbf{P}^t &= (\mathbf{I} - 2\mathbf{w}\mathbf{w}^t)^t \\ &= \mathbf{I} - 2(\mathbf{w}\mathbf{w}^t)^t \\ &= \mathbf{I} - 2(\mathbf{w}^t)^t\mathbf{w}^t \\ &= \mathbf{I} - 2\mathbf{w}\mathbf{w}^t = \\ &= \mathbf{P} \end{aligned}$$

⁶Sólo los cuatro elementos que se encuentran en las esquinas del cuadrado tienen fórmulas más complicadas, pero esto no es importante aquí.

⁷Nótese que, en \mathbf{P} , $\mathbf{w}\mathbf{w}^t$ es una matriz cuadrada, en tanto que $\mathbf{w}^t\mathbf{w}$ es un escalar

La ortogonalidad se deduce como sigue

$$\begin{aligned}\mathbf{P}^2 &= (\mathbf{I} - 2\mathbf{w}\mathbf{w}^t)(\mathbf{I} - 2\mathbf{w}\mathbf{w}^t) \\ &= \mathbf{I} - 4\mathbf{w}\mathbf{w}^t + 4\mathbf{w}(\mathbf{w}^t\mathbf{w})\mathbf{w}^t \\ &= \mathbf{I}\end{aligned}$$

Así $\mathbf{P}^{-1} = \mathbf{P}$, pero como $\mathbf{P}^t\mathbf{P}$, tenemos que $\mathbf{P}^{-1} = \mathbf{P}^t$, lo que prueba la ortogonalidad.

El vector \mathbf{w} se escoge de manera tal que para un vector \mathbf{x} dado

$$\mathbf{P}\mathbf{x} = k\mathbf{e}_1$$

donde \mathbf{e}_1 es el vector $[1, 0, \dots, 0]^t$; es decir la matriz \mathbf{P} actuando sobre un dado vector \mathbf{x} lleva a cero todos sus componentes excepto el primero. Nótese que el *módulo* del escalar k es necesariamente la norma euclídeana de \mathbf{x} ⁸.

Para reducir una matriz simétrica \mathbf{A} a la forma tridiagonal construimos la matriz de transformación \mathbf{T}_1 como una matriz particionada que consiste de un 1 en la posición $(1, 1)$, una matriz \mathbf{P}_1 de Householder de orden $(n-1) \times (n-1)$ y ceros en el resto como se muestra a continuación

$$\mathbf{T}_1 = \left[\begin{array}{c|cccc} 1 & 0 & 0 & \cdots & 0 \\ \hline 0 & & & & \\ 0 & & & & \\ \vdots & & & & \\ 0 & & & & \mathbf{P}_1 \end{array} \right]$$

Nótese que esta matriz así construida es ortogonal y simétrica, con lo cual $\mathbf{T}_1^{-1} = \mathbf{T}_1^t = \mathbf{T}_1$.

Si tomamos como vector \mathbf{x} los $(n-1)$ elementos inferiores de la *primer* columna de \mathbf{A} , la *premultiplicación* por $\mathbf{T}_1^{-1} = \mathbf{T}_1$ llevará los $(n-2)$ elementos inferiores de dicha columna a cero

$$\begin{aligned}\mathbf{T}_1\mathbf{A} &= \left[\begin{array}{c|cccc} 1 & 0 & 0 & \cdots & 0 \\ \hline 0 & & & & \\ 0 & & & & \\ \vdots & & & & \\ 0 & & & & \mathbf{P}_1 \end{array} \right] \left[\begin{array}{c|cccc} a_{11} & a_{12} & a_{13} & \cdots & a_{1n} \\ \hline a_{21} & & & & \\ a_{31} & & & & \\ \vdots & & & & \\ a_{n1} & & & & \end{array} \right] \\ &= \left[\begin{array}{c|cccc} a_{11} & a_{12} & a_{13} & \cdots & a_{1n} \\ \hline k & & & & \\ 0 & & & & \\ \vdots & & & & \\ 0 & & & & \end{array} \right] \begin{array}{l} \\ \\ \\ \text{irrelevante} \\ \end{array}\end{aligned}$$

y la *postmultiplicación* por \mathbf{P} no destruye los ceros ya que cualquier postmultiplicación por \mathbf{T} no cambia la primer columna, de la misma manera que cualquier premultiplicación no cambia la primer fila. Además, si la matriz \mathbf{A} es simétrica, dado que la simetría debe preservarse por la transformación ortogonal, tendremos los correspondientes ceros en la primera fila de la transformación completa

$$\mathbf{T}_1\mathbf{A}\mathbf{T}_1 = \left[\begin{array}{c|cccc} a_{11} & k & 0 & \cdots & 0 \\ \hline k & & & & \\ 0 & & & & \\ \vdots & & & & \\ 0 & & & & \end{array} \right] \begin{array}{l} \\ \\ \\ \text{irrelevante} \\ \end{array}$$

⁸En efecto, siendo \mathbf{P} ortogonal, $\|\mathbf{x}\|_2^2 = \mathbf{x}^t\mathbf{x} = \mathbf{x}\mathbf{P}^t\mathbf{P}\mathbf{x} = (\mathbf{P}\mathbf{x})^t(\mathbf{P}\mathbf{x}) = k^2\mathbf{e}_1^t\mathbf{e}_1 = k^2$, de donde $\|\mathbf{x}\|_2 = |k|$.

3. Para $j = i + 2, i + 3, \dots, n$ se calculan los multiplicadores

$$l_{j,i+1} = \frac{a_{ji}}{a_{i+1,j}}$$

y se *sustraen* $l_{j,i+1}$ veces la fila $i+1$ de la fila j . Para hacer esta eliminación una transformación de semejanza, también se *suma* $l_{j,i+1}$ veces la columna j a la columna $i+1$. Así en este paso hemos calculado la matriz

$$\mathbf{G}_{i+1}^{-1} \mathbf{A}' \mathbf{G}_{i+1}$$

Como resultado de los pasos anteriores la matriz obtenida

$$\begin{aligned} \mathbf{A}^{(i)} &= \mathbf{G}_{i+1}^{-1} \mathbf{A}' \mathbf{G}_{i+1} \\ &= \mathbf{G}_{i+1}^{-1} \mathbf{P}_{rs}^{-1} \mathbf{A}^{(i)} \mathbf{P}_{rs} \mathbf{G}_{i+1} \\ &= \mathbf{T}_i^{-1} \mathbf{A}^{(i-1)} = \mathbf{T}_i, \quad \mathbf{T}_i = \mathbf{P}_{rs} \mathbf{G}_{i+1} \end{aligned}$$

tiene sus primeras i columnas en la forma de Hessenberg. Repitiendo este proceso $n - 2$ veces obtenemos la matriz de Hessenberg $\mathbf{B} = \mathbf{A}_{n-2}$ deseada.

El número de multiplicaciones realizadas por este algoritmo es del orden de $5n^3/6$, la mitad de operaciones necesarias para llevar a cabo la misma tarea con el algoritmo de Householder. Ya que en la mayoría de los casos prácticos este método no es esencialmente menos estable que el método de Householder, resulta ser el preferido en la práctica para reducir una matriz no simétrica a la matriz de Hessenberg.

Métodos para determinar los autovalores

Aunque, formalmente, los autovalores de una matriz $n \times n$ son las raíces su polinomio característico, encontrar la forma *explícita* de este polinomio y, recién entonces, determinar sus raíces es un procedimiento demasiado costoso para ser llevado a cabo en una matriz grande. Una estrategia más razonable a la construcción explícita, es *evaluar* el polinomio característico directamente para algún valor de prueba. Si nuestro valor de prueba es próximo a un autovalor de la matriz el valor del polinomio será pequeño, y nulo si nuestra estimación es exacta (salvo por errores de redondeo en la evaluación aritmética). Para una matriz dada la evaluación del polinomio característico requiere del cómputo de un determinante de orden n . Para una matriz *densa* el cómputo de tal determinante a través de su definición recursiva requiere del orden de $n!$ operaciones —por supuesto, tal cálculo es tremendamente ineficiente— En su lugar el uso de un procedimiento de eliminación gaussiana permite reducir el número de operaciones a aproximadamente n^3 ⁹. Ahora bien, para una matriz de Hessenberg tal cálculo requiere en realidad de n^2 operaciones y para una matriz tridiagonal el número es proporcional a n . De aquí nuestro interés en algoritmos que permitan evaluar los polinomios característicos de estas matrices especiales.

⁹El método se basa en que el algoritmo de eliminación gaussiana produce una factorización $\mathbf{A} = \mathbf{PLU}$ de una matriz \mathbf{A} donde \mathbf{P} es una matriz de permutación determinada por la estrategia de pivoteo, \mathbf{L} una matriz triangular inferior con elementos de su diagonal iguales a la unidad y $\mathbf{U} = (u_{ij})$ una matriz triangular superior. Entonces, como el determinante de una matriz triangular (inferior o superior) es el producto de los elementos de su diagonal y el determinante de una matriz de permutación (la cual intercambia filas de la matriz identidad) es 1 o -1 si la permutación es par o impar, tenemos que

$$\det \mathbf{A} = (-1)^p \prod_{i=1}^n u_{i,i}$$

donde p es el número de intercambios de filas realizados durante el algoritmo de eliminación.

Evaluación del polinomio característico de una matriz tridiagonal (simétrica)

Consideremos una matriz tridiagonal simétrica

$$\mathbf{A} = \begin{bmatrix} \alpha_1 & \beta_2 & & & \\ \beta_2 & \ddots & \ddots & & \\ & \ddots & \ddots & \beta_n & \\ & & & \beta_n & \alpha_n \end{bmatrix}$$

Sin pérdida de generalidad podemos asumir que \mathbf{A} es una matriz tridiagonal *irreducible*, esto es, $\beta_i \neq 0$ para todo i . De lo contrario, \mathbf{A} puede escribirse como una matriz triangular por bloques, cuyos bloques son matrices tridiagonales irreducibles $\mathbf{A}^{(i)}$, $i = 1, \dots, k$

$$\mathbf{A} = \begin{bmatrix} \mathbf{A}^{(1)} & & & & \\ & \mathbf{A}^{(2)} & & & \\ & & \ddots & & \\ & & & \ddots & \\ & & & & \mathbf{A}^{(k)} \end{bmatrix}$$

Como el polinomio característico de \mathbf{A} es el producto de los polinomios característicos de los bloques $\mathbf{A}^{(i)}$, los autovalores de \mathbf{A} serán los autovalores de $\mathbf{A}^{(i)}$, $i = 1, \dots, k$. Así, pues, es suficiente considerar matrices irreducibles \mathbf{A} .

Escribamos el polinomio característico $p(\lambda)$ de \mathbf{A} como el determinante que lo define

$$p(\lambda) = \begin{vmatrix} \alpha_1 - \lambda & \beta_2 & & & & \\ \beta_2 & \alpha_2 - \lambda & \beta_3 & & & \\ & \beta_3 & \alpha_3 - \lambda & & & \\ & & \beta_4 & \ddots & & \\ & & & \ddots & \ddots & \\ & & & & \alpha_{n-1} - \lambda & \beta_n \\ & & & & \beta_n & \alpha_n - \lambda \end{vmatrix}$$

Desarrollando el determinante por la última columna resulta

$$p(\lambda) = (\alpha_n - \lambda) \begin{vmatrix} \alpha_1 - \lambda & \beta_2 & & & \\ \beta_2 & \alpha_2 - \lambda & \beta_3 & & \\ & \beta_3 & \alpha_3 - \lambda & & \\ & & \ddots & \ddots & \\ & & & \beta_{n-1} & \\ & & & \alpha_{n-1} - \lambda & \end{vmatrix} - \beta_n \begin{vmatrix} \alpha_1 - \lambda & \beta_2 & & & \\ \beta_2 & \alpha_2 - \lambda & \beta_3 & & \\ & \beta_3 & \alpha_3 - \lambda & & \\ & & \ddots & \ddots & \\ & & & \beta_{n-2} & \\ & & & \alpha_{n-2} - \lambda & \beta_{n-1} \\ & & & \beta_n & \end{vmatrix}$$

Tenemos así un término que es el producto de $(\alpha_n - \lambda)$ por el polinomio característico $p_{n-1}(\lambda)$ de la submatriz tridiagonal de dimensiones $(n-1) \times (n-1)$, y otro término que es el producto de β_n por otro determinante de orden $(n-1)$. Desarrollando éste por su última fila, la cual contiene solo a β_n obtenemos la fórmula

$$p(\lambda) = (\alpha_n - \lambda) p_{n-1}(\lambda) - \beta_n^2 p_{n-2}(\lambda)$$

Notando que

$$p_1(\lambda) = (\alpha_1 - \lambda), \quad p_2(\lambda) = (\alpha_2 - \lambda)p_1(\lambda) - \beta_2^2$$

y definiendo

$$p_0(\lambda) = 1$$

podemos calcular $p(\lambda)$ de \mathbf{A} en forma recursiva. En efecto, si

$$p_i(\lambda) = \det(\mathbf{A}_i - \lambda\mathbf{I}), \quad \mathbf{A}_i = \begin{bmatrix} \alpha_1 & \beta_2 & & & & & \\ & \beta_2 & \ddots & & & & \\ & & \ddots & \ddots & & & \\ & & & \ddots & \ddots & & \\ & & & & \beta_i & & \\ & & & & \beta_i & \alpha_i & \end{bmatrix}$$

calculamos

$$\begin{aligned} p_0(\lambda) &= 1 \\ p_1(\lambda) &= (\alpha_1 - \lambda) \\ p_i(\lambda) &= (\alpha_i - \lambda)p_{i-1}(\lambda) - \beta_i^2 p_{i-2}(\lambda), \quad i = 2, 3, \dots, n \end{aligned}$$

con lo cual $p(\lambda) = p_n(\lambda)$.

Para un valor dado de λ este algoritmo proporciona una forma razonable de evaluar $p(\lambda)$. Podemos entonces utilizar los métodos numéricos clásicos para determinar las raíces de un polinomio. Así utilizando un cierto paso en los valores de prueba λ podemos evaluar $p(\lambda)$ hasta obtener un cambio de signo y entonces encerrar la raíz dividiendo el intervalo a la mitad —este es el *método dicotómico*—. O bien, habiendo obtenido una aproximación a la raíz, utilizar un método como el de la *secante* o el *método de Newton*. En este último caso es necesario el conocimiento de la derivada del polinomio $p(\lambda)$, pero ésta puede ser calculada recursivamente por el algoritmo que resulta de diferenciar la fórmula recursiva que permite el cálculo del polinomio. Esto conduce a

$$\begin{aligned} p'_0(\lambda) &= 0 \\ p'_1(\lambda) &= -1 \\ p'_i(\lambda) &= (\alpha_i - \lambda)p'_{i-1}(\lambda) - \beta_i^2 p'_{i-2}(\lambda) - p'_{i-1}(\lambda), \quad i = 2, 3, \dots, n \end{aligned}$$

con lo cual $p'(\lambda) = p'_n(\lambda)$.

Evaluación del polinomio característico de una matriz de Hessenberg

Podemos evaluar el determinante del polinomio característico de una matriz de Hessenberg para un valor de prueba λ por un método similar al usado con una matriz tridiagonal. Deseamos encontrar el valor numérico de

$$p(\lambda) = \begin{vmatrix} a_{11} - \lambda & a_{12} & a_{13} & a_{14} & \dots & a_{1,n-1} & a_{1n} \\ b_1 & a_{22} - \lambda & a_{23} & a_{24} & & a_{2,n-1} & a_{2n} \\ & b_2 & a_{33} - \lambda & a_{34} & & a_{3,n-1} & a_{3n} \\ & & b_3 & a_{44} - \lambda & & a_{4,n-1} & a_{4n} \\ & & & b_4 & & \vdots & \vdots \\ & & & & & a_{n-1,n-1} - \lambda & a_{n-1,n} \\ & & & & \dots & b_{n-1} & a_{nn} - \lambda \end{vmatrix}$$

donde todas las cantidades en el determinante son números conocidos. Nuestra estrategia consiste en encontrar una combinación lineal de las $(n-1)$ primeras *columnas* que, cuando es sumada a

la última columna, reduzca a ésta a ceros excepto la primer posición. Esto es, queremos encontrar un conjunto de escalares β_i tales que

$$\mathbf{C}_n + \sum_{i=1}^{n-1} \beta_i \mathbf{C}_i = k(\lambda) \mathbf{e}_1 \quad (0.1)$$

donde los \mathbf{C}_i son las columnas de nuestro determinante. Si reemplazamos la ultima columna del determinante por esta combinación lineal de columnas, el valor numérico del determinante no cambia, y éste tiene ahora la forma

$$p(\lambda) = \begin{bmatrix} a_{11} - \lambda & a_{12} & a_{13} & a_{14} & \dots & a_{1,n-1} & k \\ b_1 & a_{22} - \lambda & a_{23} & a_{24} & & a_{2,n-1} & 0 \\ & b_2 & a_{33} - \lambda & a_{34} & & a_{3,n-1} & 0 \\ & & b_3 & a_{44} - \lambda & & a_{4,n-1} & 0 \\ & & & b_4 & & \vdots & \vdots \\ & & & & & a_{n-1,n-1} - \lambda & 0 \\ & & & & \dots & b_{n-1} & 0 \end{bmatrix}$$

Desarrollando este determinante por la última columna obtenemos

$$p(\lambda) = (-1)^{n+1} k(\lambda) \begin{bmatrix} b_1 & a_{22} - \lambda & a_{23} & \dots & a_{2,n-1} \\ & b_2 & a_{33} - \lambda & & a_{3,n-1} \\ & & b_3 & & a_{4,n-1} \\ & & & \ddots & \vdots \\ & & & & a_{n-1,n-1} - \lambda \\ & & & & b_{n-1} \end{bmatrix}$$

el cual es el determinante de una matriz triangular y por lo tanto es el producto de los elementos de su diagonal. Así

$$p(\lambda) = (-1)^{n+1} k(\lambda) \prod_{i=1}^{n-1} b_{ii} \quad (0.2)$$

Podemos encontrar los β_i en forma sucesiva a partir de las ecuaciones (0.1)

$$\begin{aligned} \beta_{n-1} b_{n-1} + (a_{nn} - \lambda) &= 0 \\ \beta_{n-2} b_{n-2} + \beta_{n-1} (a_{n-1,n-1} - \lambda) + a_{n-1,n} &= 0 \\ &\vdots \\ \beta_1 b_1 + \beta_2 (a_{22} - \lambda) + \beta_3 a_{23} + \dots + \beta_{n-1} a_{2,n-1} + a_{2n} &= 0 \end{aligned}$$

y por lo tanto obtenemos k de

$$\beta_1 (a_{11} - \lambda) + \beta_2 a_{12} + \dots + \beta_{n-1} a_{1,n-1} + a_{1n} = k$$

y ahora podemos evaluar $p(\lambda)$ de (0.2).

Algoritmos de factorización

Para finalizar vamos a presentar un conjunto de algoritmos *iterativos* para calcular los autovalores de una matriz \mathbf{A} $n \times n$ que si bien no convergen exactamente en número *finito* de pasos resultan de suma importancia. Estos algoritmos son conocidos como *métodos de factorización* ya que se basan en la suposición de que la matriz \mathbf{A} puede ser factorizada en una matriz izquierda \mathbf{F}_L y una matriz derecha \mathbf{F}_R

$$\mathbf{A} = \mathbf{F}_L \mathbf{F}_R$$

Si ahora invertimos el orden de los factores obtenemos una matriz

$$\mathbf{A}' = \mathbf{F}_R \mathbf{F}_L = \mathbf{F}_L^{-1} \mathbf{A} \mathbf{F}_L$$

(dado que $\mathbf{F}_L^{-1} \mathbf{A} = \mathbf{F}_R$) que es semejante a la matriz \mathbf{A} a través de la transformación \mathbf{F}_L . Así, *cualquier* factorización en dos matrices al ser multiplicadas en orden inverso constituyen una transformación de semejanza. Los métodos de factorización explotan esta idea, pero para que el algoritmo sea práctico la factorización de ser tal que el método converja, sea estable y eficiente.

El primero de los dos métodos que consideraremos es el llamado *método LR*. Aquí se genera una sucesión de matrices comenzando con $\mathbf{A}^{(1)} = \mathbf{A}$, y se utiliza el procedimiento de eliminación gaussiana para representar la matriz $\mathbf{A}^{(i)}$ como el producto de una matriz triangular inferior $\mathbf{L}_i = (l_{jk})$ con $l_{jj} = 1$ y una matriz triangular superior \mathbf{R}_i

$$\mathbf{A}^{(i)} = \mathbf{L}_i \mathbf{R}_i, \quad \mathbf{L}_i = \begin{bmatrix} 1 & & 0 \\ \vdots & \ddots & \\ * & \cdots & 1 \end{bmatrix}, \quad \mathbf{R}_i = \begin{bmatrix} * & \cdots & * \\ & \ddots & \vdots \\ 0 & & * \end{bmatrix}$$

y entonces se toma

$$\mathbf{A}^{(i+1)} = \mathbf{R}_i \mathbf{L}_i = \mathbf{L}_{i+1} \mathbf{R}_{i+1}, \quad i = 1, 2, \dots$$

Ahora bien, sabemos que para una matriz arbitraria $\mathbf{A}^{(i+1)}$ no siempre existe tal factorización *sin intercambio de filas*. Sin embargo, asumiendo que $\mathbf{A}^{(i)}$ puede ser factorizada, es posible asegurar que, bajo ciertas condiciones, la sucesión de matrices $\mathbf{A}^{(i)}, i = 1, \dots$ converge a una matriz triangular superior cuyos elementos de la diagonal son los autovalores de \mathbf{A} .

Para una matriz densa $n \times n$, la factorización requiere del orden de $n^3/3$ multiplicaciones, y la multiplicación matricial en orden inverso de otras $n^3/3$. Así, en cada paso del algoritmo (necesariamente infinito) tenemos del orden de n^3 operaciones. Por lo tanto, el algoritmo *LR* no es práctico para matrices generales. Ahora bien, puede verificarse fácilmente que las formas tridiagonales y de Hessenberg se preservan bajo el algoritmo *LR*. Como la factorización de matrices tridiagonales requiere del orden de n multiplicaciones, y n^2 para matrices de Hessenberg, vemos que el algoritmo *LR* puede ser efectivo para encontrar los autovalores de estas matrices particulares. Para una matriz general solo debe reducirse primero a una de las formas anteriores por los métodos ya discutidos.

Sin embargo el método *LR* adolece de problemas. En primer lugar las iteraciones se detendrán si alguna de las matrices $\mathbf{A}^{(i)}$ no tiene la descomposición triangular, y aún, si esta descomposición existe el problema de calcular $\mathbf{L}^{(i)}$ y \mathbf{R}_i puede estar mal condicionado al no permitir el intercambio de filas con el fin de evitar elementos de pivote pequeños. Así el método puede ser inestable.

Para evitar estas dificultades, y hacer el método numéricamente estable, uno podría realizar la descomposición con pivote parcial ¹⁰

$$\mathbf{P}_{(i)} \mathbf{A}^{(i)} = \mathbf{L}_i \mathbf{R}_i, \quad \mathbf{P}_i \text{ matriz de permutación, } \mathbf{P}_i^{-1} = \mathbf{P}_i^t \\ \mathbf{A}^{(i+1)} = \mathbf{R}_i \mathbf{P}_i^t \mathbf{L}_i = \mathbf{L}_i^{-1} (\mathbf{P}_i \mathbf{A}^{(i)} \mathbf{P}_i^t) \mathbf{L}_i$$

¹⁰La cual siempre existe.

Sin embargo, hay ejemplos para los cuales este proceso *no* converge. Por ejemplo, sea

$$\begin{aligned}\mathbf{A}^{(1)} &= \begin{bmatrix} 1 & 3 \\ 2 & 0 \end{bmatrix}, \quad \lambda_1 = 3, \lambda_2 = -2 \\ \mathbf{P}_1 &= \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}, \quad \mathbf{P}_1 \mathbf{A}^{(1)} = \begin{bmatrix} 2 & 0 \\ 1 & 3 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 1/2 & 1 \end{bmatrix} \begin{bmatrix} 2 & 0 \\ 0 & 3 \end{bmatrix} = \mathbf{L}_1 \mathbf{R}_1 \\ \mathbf{A}^{(2)} &= \mathbf{R}_1 \mathbf{P}_1^t \mathbf{L}_1 = \begin{bmatrix} 1 & 2 \\ 3 & 0 \end{bmatrix} \\ \mathbf{P}_2 &= \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}, \quad \mathbf{P}_2 \mathbf{A}^{(2)} = \begin{bmatrix} 3 & 0 \\ 1 & 2 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 1/3 & 1 \end{bmatrix} \begin{bmatrix} 3 & 0 \\ 0 & 2 \end{bmatrix} = \mathbf{L}_2 \mathbf{R}_2 \\ \mathbf{A}^{(3)} &= \mathbf{R}_2 \mathbf{P}_2^t \mathbf{L}_2 = \begin{bmatrix} 1 & 3 \\ 2 & 0 \end{bmatrix} = \mathbf{A}^{(1)}\end{aligned}$$

Estas dificultades del método *LR* desaparecen en el llamado *método QR*. Formalmente, el método comienza con $\mathbf{A}^{(1)} = \mathbf{A}$ y genera matrices \mathbf{Q}_i y \mathbf{R}_i tales que

$$\mathbf{A}^{(i)} = \mathbf{Q}_i \mathbf{R}_i$$

donde \mathbf{Q}_i es una matriz *ortogonal*, $\mathbf{Q}_i^t \mathbf{Q}_i = \mathbf{I}$ y \mathbf{R}_i es una matriz triangular superior

$$\mathbf{R}_i = \begin{bmatrix} * & \cdots & * \\ & \ddots & \vdots \\ 0 & & * \end{bmatrix}$$

Puede mostrarse que una factorización de este tipo siempre existe y puede ser calculada en una manera estable a través de matrices de Householder que anulen los elementos de las sucesivas columnas $\mathbf{A}^{(i)}$ debajo de la diagonal.

El método toma ahora como siguiente matriz de la sucesión

$$\mathbf{A}^{(i+1)} = \mathbf{R}_i \mathbf{Q}_i$$

La transformación $\mathbf{A}^{(i)} \rightarrow \mathbf{A}^{(i+1)}$ no solo es de semejanza, sino que también es ortogonal. En efecto, ya que \mathbf{Q} es ortogonal, $\mathbf{R}_i = \mathbf{Q}_i^{-1} \mathbf{A}^{(i)} = \mathbf{Q}_i^t \mathbf{A}^{(i)}$ y por lo tanto

$$\mathbf{A}^{(i+1)} = \mathbf{R}_i \mathbf{Q}_i = \mathbf{Q}_i^t \mathbf{A}^{(i)} \mathbf{Q}_i$$

Puede mostrarse que, bajo ciertas condiciones —en particular que los autovalores λ_i de \mathbf{A} tengan módulos distintos

$$|\lambda_1| > |\lambda_2| > \cdots > |\lambda_n|$$

— entonces la sucesión de matrices $\mathbf{A}^{(i)}$ converge a una matriz triangular superior siendo sus elementos diagonales los autovalores de \mathbf{A} . Además la velocidad de convergencia con que las componentes debajo de la diagonal principal tienden a cero está determinada por los cocientes

$$\left| \frac{\lambda_j}{\lambda_k} \right|, \quad j > k$$

Así la convergencia puede resultar muy lenta si algunos de estos cocientes son cercanos a la unidad. Sin embargo, con una modificación del procedimiento se puede acelerar la convergencia.

Como en el método *LR*, un paso en el método *QR* para una matriz densa $n \times n$ requiere del orden de n^3 operaciones. Debido a esto, el método *QR* se aplica en la práctica solo a matrices simples, como son las matrices de Hessenberg, o en la caso de matrices simétricas, las matrices tridiagonales (simétricas). Así una matriz cualquiera debe ser reducida previamente a una de estas formas por los métodos ya discutidos. Por supuesto, para que este procedimiento tenga sentido

debe mostrarse que estas formas especiales son invariantes por una transformación QR ; esto es, si $\mathbf{A}^{(i)}$ es una matriz de Hessenberg (posiblemente simétrica, en cuyo caso es tridiagonal), entonces también lo es $\mathbf{A}^{(i+1)}$. Esta invarianza puede ser establecida fácilmente. En primer lugar, si $\mathbf{A}^{(i)}$ es simétrica, $\mathbf{A}^{(i+1)}$ también lo es debido que la transformación QR es ortogonal. Supongamos ahora que $\mathbf{A}^{(i)}$ es una matriz de Hessenberg $n \times n$, como ya mencionamos $\mathbf{A}^{(i+1)}$ puede ser calculada a partir de una sucesión de transformaciones de Householder, sin embargo, en el caso de matrices de Hessenberg, resulta más eficiente utilizar transformaciones de Givens. Primero, reducimos los elementos de la subdiagonal $\mathbf{A}^{(i)}$ a cero a través de matrices de Givens adecuadas del tipo $\Omega_{21}, \dots, \Omega_{n-1,n}$

$$\Omega_{n-1,n} \cdots \Omega_{2,3} \Omega_{1,2} \mathbf{A}^{(i)} = \mathbf{R}_i = \begin{bmatrix} * & \cdots & * \\ & \ddots & \vdots \\ 0 & & * \end{bmatrix}$$

$$\mathbf{A}^{(i)} = \mathbf{Q}_i \mathbf{R}_i, \quad \mathbf{Q}_i = \Omega_{12}^t \Omega_{23}^t \cdots \Omega_{n-1,n}^t$$

y entonces calculamos $\mathbf{A}^{(i+1)}$

$$\mathbf{A}^{(i+1)} = \mathbf{R}_i \mathbf{Q}_i = \mathbf{R}_i \Omega_{12}^t \Omega_{23}^t \cdots \Omega_{n-1,n}^t$$

Debido a la estructura especial de $\Omega_{j,j+1}$, la matriz triangular superior \mathbf{R}_i es transformada por la postmultiplicación con $\Omega_{j,j+1}^t$ a una matriz $\mathbf{A}^{(i+1)}$ de la forma de Hessenberg. Nótese que $\mathbf{A}^{(i+1)}$ puede ser calculada de $\mathbf{A}^{(i)}$ muy eficientemente si las multiplicaciones matriciales son realizadas en el siguiente orden

$$\mathbf{A}^{(i+1)} = (\Omega_{n-1,n} \cdots (\Omega_{23} ((\Omega_{12} \mathbf{A}^{(i)}) \Omega_{12}^t)) \Omega_{23}^t \cdots) \Omega_{n-1,n}^t$$

Llevada a cabo de esta manera, un paso en la transformación de una matriz de Hessenberg $n \times n$ requiere solo de n^2 multiplicaciones y solo n operaciones en el caso simétrico, esto es, en el caso de una matriz tridiagonal simétrica.

Aplicado de esta forma, el método QR resulta ser uno de los mejores métodos conocidos para determinar *todos* los autovalores de una matriz.

Referencias

- [1] F. S. Acton: *Numerical Methods that Work*.
- [2] J. Stoer and R. Bulirsch: *Introduction to Numerical Analysis*.
- [3] *Numerical Recipes in FORTRAN 77: The art of scientific computing*.