Universidad de Valladolid

Facultad de Ciencias

Grado en Matemáticas

La descomposición en valores singulares

Alumna: Beatriz Salvador Mancho

Tutora: M.P. Calvo Cabrero

Índice

Introducción

El Trabajo Fin de Grado que presentamos tiene por objeto el estudio de la descomposición en valores singulares de una matriz.

En la Sección 1 se incluyen los resultados teóricos más relevantes relacionados con dicha descomposición. Cabe destacar el teorema de descomposición que establece que cualquier matriz $A \in \mathbb{R}^{m\times n}$ se puede escribir como un producto $A = U\Sigma V^T$, donde $U \in \mathbb{R}^{m\times m}$ y $V \in \mathbb{R}^{n\times n}$ son matrices ortogonales y $\Sigma \in \mathbb{R}^{m\times n}$ es una matriz cuyos elementos no nulos están situados en las primeras r posiciones de la diagonal principal, siendo r el rango de A.

En la Sección 2 se explican con detalle los algoritmos que permiten calcular en la práctica y de un modo eficiente la descomposición en valores singulares de una matriz. Dichos algoritmos han sido implementados en MATLAB y los programas se han incluído en el Apéndice.

En la Sección 3 se recogen dos aplicaciones reales de la descomposición en valores singulares. La primera relacionada con la compresión y transmisión de imágenes, ilustra el interés de uno de los teoremas de aproximación que se han estudiado en la Sección 1. La segunda corresponde a un ejemplo de Climatología, donde se estudia la covariabilidad entre las precipitaciones en la costa mediterránea española y las alturas geopotenciales en la zona de Atlántico Norte y Europa.

En el Apéndice se incluyen los programas en MATLAB que implementan los algoritmos descritos en la Sección 2 y un programa en MATLAB que permite visualizar gráficamente la actuación paso a paso de dichos algoritmos.

Agradecemos a la Doctora María Luisa Martín (Universidad de Valladolid) el habernos facilitado los datos reales para el ejemplo de Climatología.

1. La descomposición en valores singulares. Resultados teóricos

Como ya se ha indicado en la introducción, se presentan en esta sección los resultados teóricos más relevantes en relación con la descomposición en valores singulares de una matriz. En [10] se pueden encontrar algunos de ellos y en [7] se hace una revisión histórica sobre el tema.

1.1. El teorema de descomposición

Teorema 1. *Dada una matriz $A \in \mathbb{R}^{m\times n}$ de rango r, existen números reales $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_r > 0$ y bases ortonormales $\{u_1, u_2, \ldots, u_m\}$ de $\mathbb{R}^m$ y $\{v_1, v_2, \ldots, v_n\}$ de $\mathbb{R}^n$ tales que*

$$Av_i = \sigma_i u_i, \qquad i = 1, 2, \ldots, r, \qquad\qquad A^T u_i = \sigma_i v_i, \qquad i = 1, 2, \ldots, r,$$

$$Av_i = 0, \qquad i = r+1, \ldots, n, \qquad\qquad A^T u_i = 0, \qquad i = r+1, \ldots, m.$$

Estas ecuaciones implican que $v_1, v_2, \ldots, v_n$ son autovectores de A^TA, $u_1, u_2, \ldots, u_m$ son autovectores de AA^T y $\sigma_1^2, \ldots, \sigma_r^2$ son los autovalores no nulos de A^TA y AA^T.

Demostración. Como A^TA es simétrica, diagonaliza ortogonalmente y sus autovalores son reales. Sea $\{v_1, \ldots, v_n\}$ una base ortonormal de $\mathbb{R}^n$ formada por autovectores de A^TA y sean $\lambda_1, \ldots, \lambda_n$ los autovalores asociados. Como A^TA es semidefinida positiva todos sus autovalores son mayores o iguales que 0. Sin pérdida de generalidad consideramos $v_1, \ldots, v_n$ ordenados de modo que $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_n$. Por ser r el rango de A y, por tanto, el rango de A^TA, debe cumplirse que $\lambda_r > 0$ y $\lambda_{r+1} = \lambda_{r+2} = \cdots = \lambda_n = 0$. Por tanto $v_1, \ldots, v_r$ son autovectores de A^TA asociados a autovalores positivos y, si $r < n$, $v_{r+1}, \ldots, v_n$ están en el núcleo de A^TA y, por tanto, en el núcleo de A, es decir, $Av_i = 0$ para $i = r+1, \ldots, n$.

Para $1 \leq i \leq r$ definimos σ_i y u_i del siguiente modo

$$\sigma_i = \sqrt{\lambda_i}, \qquad\qquad u_i = \frac{1}{\sigma_i} Av_i. \tag{1}$$

Por la forma en que hemos definido u_i se tiene que $Av_i = \sigma_i u_i, i = 1, \ldots, r$, y por la definición de σ_i obtenemos que

$$\|u_i\| = \sqrt{u_i^T u_i} = \frac{1}{\sigma_i}\sqrt{(Av_i)^T(Av_i)} = \frac{1}{\sigma_i}\sqrt{v_i^T(A^TA)v_i} = \frac{\sigma_i}{\sigma_i}\sqrt{v_i^T v_i} = 1, \qquad i = 1, \ldots, r.$$

Por otro lado dados dos vectores ortogonales v_i, v_j, $i \neq j$, se tiene que Av_i y Av_j también son ortogonales, puesto que

$$(Av_i)^T(Av_j) = v_i^T A^T A v_j = v_i^T \sigma_j^2 v_j = \sigma_j^2 v_i^T v_j = 0.$$

Esto implica que por la forma en que hemos definido $u_i, i = 1, \ldots, r$, los vectores $u_1, \ldots, u_r$ son ortogonales y en particular ortonormales. Además $\sigma_i^2 = \lambda_i, i = 1, \ldots, r$. Ahora multiplicando por A^T en la segunda igualdad de (1) se tiene para $1 \leq i \leq r$,

$$A^T u_i = \frac{1}{\sigma_i} A^T A v_i = \frac{1}{\sigma_i} \lambda_i v_i = \sigma_i v_i.$$

Falta ahora definir $u_{r+1}, \ldots, u_m$ en el caso en que $r < m$. Los vectores $u_1, \ldots, u_r$ son autovectores de AA^T asociados a los autovalores no nulos $\lambda_1, \ldots, \lambda_r$ pues

$$AA^T u_i = A\sigma_i v_i = \sigma_i A v_i = \sigma_i^2 u_i = \lambda_i u_i.$$

Como $AA^T \in \mathbb{R}^{m \times m}$ y tiene rango r, la dimensión del subespacio $\mathrm{Ker}(AA^T)$ es $m - r$. Tomamos $u_{r+1}, \ldots, u_m$ vectores ortonormales de una base de $\mathrm{Ker}(AA^T)$. Dado que $u_{r+1}, \ldots, u_m$ son autovectores de AA^T asociados al autovalor 0 y que AA^T es simétrica, los vectores $u_{r+1}, \ldots, u_m$ son ortogonales a $u_1, \ldots, u_r$. Por tanto, $\{u_1, \ldots, u_m\}$ es una base ortonormal de $\mathbb{R}^m$ formada por autovectores de AA^T. Además, como el núcleo de AA^T es el mismo que el núcleo de A^T se tiene que $A^T u_i = 0$ para $i = r + 1, \ldots, m$.

□

En la Figura 1 incluímos una representación gráfica de la acción de A y A^T sobre las bases del Teorema 1 [10].

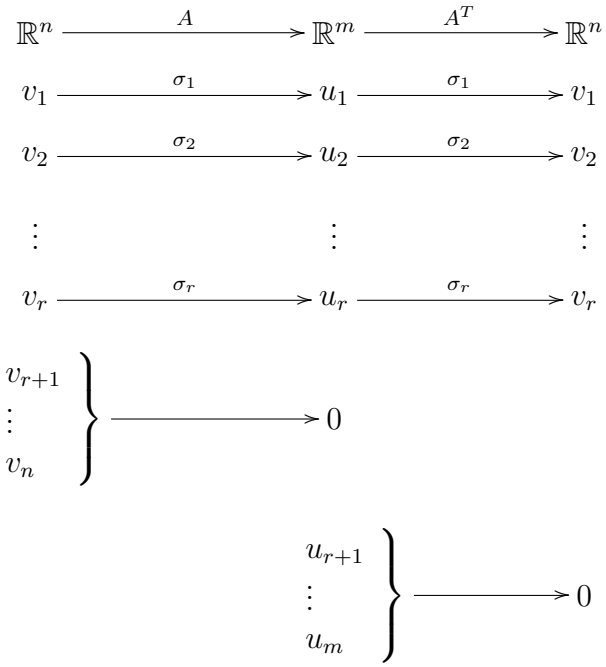


Figura 1: Acción de A y A^T sobre las bases del Teorema 1.

Es posible dar una interpretación matricial del Teorema 1.

Teorema 2. *Dada una matriz $A \in \mathbb{R}^{m\times n}$ de rango r, existen matrices ortogonales $U \in \mathbb{R}^{m\times m}$ y $V \in \mathbb{R}^{n\times n}$, y una matriz Σ de la forma*

$$\Sigma = \begin{pmatrix} \Sigma_r & O \\ O & O \end{pmatrix} = \left(\begin{array}{cccc|c} \sigma_1 & 0 & \dots & 0 & \\ 0 & \sigma_2 & \dots & 0 & \\ \vdots & \vdots & \ddots & \vdots & O \\ 0 & 0 & \dots & \sigma_r & \\ \hline \multicolumn{4}{c|}{O} & O \end{array}\right) \in \mathbb{R}^{m\times n} \tag{2}$$

con $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_r > 0$ (el símbolo O denota matrices nulas de las dimensiones adecuadas), tales que

$$A = U\Sigma V^T \tag{3}$$

Demostración. En virtud del Teorema 1, existen bases ortonormales $\{v_1, \dots, v_n\}$ de $\mathbb{R}^n$ y $\{u_1, \dots, u_m\}$ de $\mathbb{R}^m$ tales que

$$Av_i = \begin{cases} \sigma_i u_i & i = 1, \dots, r, \\ 0 & i = r+1, \dots, n. \end{cases} \tag{4}$$

Tomando U y V las matrices con columnas igual a los vectores de dichas bases ortonormales se tiene que $U = [u_1, \dots, u_m] \in \mathbb{R}^{m\times m}$ y $V = [v_1, \dots, v_n] \in \mathbb{R}^{n\times n}$ son matrices ortogonales y las ecuaciones (4) pueden ser escritas en forma matricial como

$$A[v_1, \dots, v_r | v_{r+1}, \dots, v_n] = [u_1, \dots, u_r | u_{r+1}, \dots, u_m] \left(\begin{array}{cccc|c} \sigma_1 & 0 & \dots & 0 & \\ 0 & \sigma_2 & \dots & 0 & \\ \vdots & \vdots & \ddots & \vdots & O \\ 0 & 0 & \dots & \sigma_r & \\ \hline \multicolumn{4}{c|}{O} & O \end{array}\right),$$

i.e. $AV = U\Sigma$, donde los vectores $u_{r+1}, \dots, u_m$ no intervienen expresamente al ir siempre multiplicados por ceros.

Por ser V ortogonal, $VV^T = I$ y, por tanto,

$$A = AVV^T = U\Sigma V^T.$$

□

Los σ_i, $i = 1, 2, \dots, r$, son los llamados valores singulares de A, con vectores singulares asociados por la derecha $v_1, \dots, v_r$ y por la izquierda $u_1, \dots, u_r$. Teniendo en cuenta la estructura de ceros de la matriz Σ y a la vista de (3), A se puede escribir como una suma de r matrices de rango 1

$$A = \sum_{j=1}^{r} \sigma_j u_j v_j^T. \tag{5}$$

Aunque en un sentido estricto los valores singulares de A son los elementos diagonales de Σ estrictamente positivos, en algunas secciones de este trabajo, por simplificar la notación, se ha extendido esta denominación también a los elementos nulos de dicha diagonal. En ningún caso esta extensión da lugar a confusión.

1.2. El problema lineal de mínimos cuadrados

Tomamos $A \in \mathbb{R}^{m\times n}$, de rango r y $b \in \mathbb{R}^m$ y consideramos el sistema de ecuaciones lineales

$$Ax = b,$$

con $x \in \mathbb{R}^n$ el vector de incógnitas. Si $m > n$, el sistema es sobredeterminado y podría no tener solución. En ese caso buscaremos un x tal que $\|b - Ax\|_2$ sea mínima. Hallar dicho x es lo que conocemos como resolver el problema lineal de mínimos cuadrados. En el caso en el que $m \geq n$ y rango(A)=n el problema lineal de mínimos cuadrados tiene solución única. Si rango(A)$< n$, la solución en el sentido de mínimos cuadrados no es única y existen varios $x \in \mathbb{R}^n$ para los cuales $\|b - Ax\|_2$ es mínima. Incluso en el caso $m < n$ puede ocurrir que $Ax = b$ no tenga solución.

Como la solución del problema lineal de mínimos cuadrados puede no ser única, consideramos el siguiente problema que sí va a tener solución única: *de todos los $x \in \mathbb{R}^n$ que minimizan $\|b - Ax\|_2$, encontrar aquel para el cual $\|x\|_2$ sea lo más pequeña posible.*

Supongamos A y b conocidos, y que tenemos la descomposición en valores singulares de A, $A = U\Sigma V^T$, donde $U \in \mathbb{R}^{m\times m}$ y $V \in \mathbb{R}^{n\times n}$ son ortogonales y Σ es como en (2). Como U es ortogonal,

$$\|b - Ax\|_2 = \|U^T(b - Ax)\|_2 = \|U^Tb - U^TAx\|_2 = \|U^Tb - \Sigma(V^Tx)\|_2.$$

Haciendo $c = U^Tb$ e $y = V^Tx$, se tiene

$$\|b - Ax\|_2^2 = \|c - \Sigma y\|_2^2 = \sum_{i=1}^{r} \mid c_i - \sigma_i y_i \mid^2 + \sum_{i=r+1}^{m} \mid c_i \mid^2 . \tag{6}$$

El mínimo se alcanza si, y solo si, $y_i = \dfrac{c_i}{\sigma_i}, i = 1, \ldots, r$, y vale $\displaystyle\sum_{i=r+1}^{m} \mid c_i \mid^2$.

Cuando $r < n$, $y_{r+1}, \ldots, y_n$ no aparecen explícitamente en (6) y de todas las soluciones así obtenidas tiene $\|y\|_2$ mínima la que satisface $y_{r+1} = \cdots = y_n = 0$. Como $x = Vy$ y V es ortogonal, $\|x\|_2 = \|y\|_2$. Entonces $\|x\|_2$ es mínima si, y solo si, $\|y\|_2$ lo es. Esto prueba que el problema lineal de mínimos cuadrados tiene exactamente una solución de norma mínima.

1.2.1. La pseudoinversa de una matriz

Damos ahora la definición de pseudoinversa de una matriz que adoptamos en este trabajo, que está intimamente relacionada con la resolución del problema lineal de mínimos cuadrados.

La pseudoinversa A^+ de una matriz $A \in \mathbb{R}^{m\times n}$ es la única matriz que satisface que para cada $b \in \mathbb{R}^m$, $x = A^+b$ es la solución de norma mínima del problema lineal de mínimos cuadrados $Ax = b$.

Podemos dar una expresión para A^+ en algunos casos particulares.

> Si $m = n$ y A es regular, $x = A^{-1}b$ es la única solución del sistema lineal $Ax = b$ y, por tanto, en este caso $A^+ = A^{-1}$ (la pseudoinversa coincide con la inversa).
>
> Si $m > n$ y el rango de A es n, ya hemos comentado que existe una única solución en el sentido de mínimos cuadrados del sistema lineal $Ax = b$, que no es otra que la solución de las correspondientes ecuaciones normales, que al ser A^TA regular viene dada por $x = (A^TA)^{-1}A^Tb$. En este caso, entonces, $A^+ = (A^TA)^{-1}A^T$.

En la descripción que hemos hecho del problema lineal de mínimos cuadrados hemos visto que si $c = U^Tb = \begin{pmatrix}\hat{c}\\ d\end{pmatrix}$ e $y = \begin{pmatrix}\hat{y}\\ 0\end{pmatrix}$ con $\hat{c}, \hat{y} \in \mathbb{R}^r$, la solución de norma mínima viene dada por $x = Vy$ donde además

$$Vy = V\begin{pmatrix}\hat{y}\\ 0\end{pmatrix} = V\begin{pmatrix}\Sigma_r^{-1}\hat{c}\\ 0\end{pmatrix} = V\begin{pmatrix}\Sigma_r^{-1} & O\\ O & O\end{pmatrix}\begin{pmatrix}\hat{c}\\ d\end{pmatrix} = V\begin{pmatrix}\Sigma_r^{-1} & O\\ O & O\end{pmatrix}c = V\begin{pmatrix}\Sigma_r^{-1} & O\\ O & O\end{pmatrix}U^Tb,$$

de donde, con la definición que hemos dado de pseudoinversa, se deduce que

$$A^+ = V\begin{pmatrix}\Sigma_r^{-1} & O\\ O & O\end{pmatrix}U^T.$$

De la igualdad anterior se deduce también que el rango de A^+ y el de A coinciden, que $u_1, \ldots, u_r$ y $v_1, \ldots, v_r$ son vectores singulares de A^+ por la derecha y por la izquierda, respectivamente y que $\sigma_1^{-1}, \ldots, \sigma_r^{-1}$ son los valores singulares de A^+. Además, la matriz $\begin{pmatrix}\Sigma_r^{-1} & O\\ O & O\end{pmatrix}$ no es otra que Σ^+, la pseudoinversa de Σ. Para ver esto, notemos que si nos planteamos el sistema lineal $\Sigma x = b$ o, equivalentemente,

$$\begin{aligned}\sigma_i x_i &= b_i, \quad i = 1, \ldots, r.\\ 0 &= b_i, \quad i = r+1, \ldots, m,\end{aligned}$$

su solución de norma mínima en el sentido de mínimos cuadrados viene dada por $x_i = \dfrac{b_i}{\sigma_i}$ para $i = 1, \ldots, r$, y $x_i = 0$, para $i = r+1, \ldots, n$, que puede expresarse como

$$x = \begin{pmatrix}\Sigma_r^{-1} & O\\ O & O\end{pmatrix}b \Rightarrow \Sigma^+ = \begin{pmatrix}\Sigma_r^{-1} & O\\ O & O\end{pmatrix} \in \mathbb{R}^{n\times m}.$$

Podemos escribir entonces

$$A^+ = V\Sigma^+U^T, \tag{7}$$

o incluso $A^+ = \sum_{j=1}^{r} \frac{1}{\sigma_j} v_j u_j^T$, que es la descomposición en valores singulares de A^+, análoga a (5) para A.

Veamos ahora que la pseudoinversa, tal y como se ha definido en esta subsección coincide con la inversa generalizada de Moore-Penrose

Teorema 3. *La pseudoinversa tal como se ha definido en esta sección coincide con la inversa generalizada de Moore-Penrose. Más precisamente, si $A \in \mathbb{R}^{m\times n}$ es una matriz dada y $A^+ \in \mathbb{R}^{n\times m}$ denota su pseduoinversa se satisface que:*

1. $AA^+A = A$.
2. $A^+AA^+ = A^+$.
3. $(AA^+)^T = AA^+$.
4. $(A^+A)^T = A^+A$.

Demostración. En todos los casos vamos a utilizar la descomposición en valores singulares de A y A^+ obtenidas en (3) y (7) respectivamente. Para probar la primera igualdad notemos que

$$AA^+A = (U\Sigma V^T)(V\Sigma^+U^T)(U\Sigma V^T) = U\Sigma\Sigma^+\Sigma V^T$$

Teniendo en cuenta que $\Sigma\Sigma^+$ es una matriz diagonal $m \times m$ con los elementos $(i,i) = 1, i = 1,\ldots,r$ e $(i,i) = 0, i = r+1,\ldots,m$, es fácil concluir que $\Sigma\Sigma^+\Sigma = \Sigma$ y, por tanto, que

$$AA^+A = U\Sigma\Sigma^+\Sigma V^T = U\Sigma V^T = A.$$

Para probar 2, de forma análoga a lo anterior, tenemos

$$A^+AA^+ = (V\Sigma^+U^T)(U\Sigma V^T)(V\Sigma^+U^T) = V\Sigma^+\Sigma\Sigma^+U^T.$$

Teniendo en cuenta ahora que $\Sigma^+\Sigma$ es una matriz diagonal $n \times n$ con los elementos $(i,i) = 1, i = 1,\ldots,r$ e $(i,i) = 0, i = r+1,\ldots,n$ es fácil ver que $\Sigma^+\Sigma\Sigma^+ = \Sigma^+$ y, por tanto, concluir que

$$A^+AA^+ = V\Sigma^+\Sigma\Sigma^+U^T = V\Sigma^+U^T = A^+.$$

Para la igualdad 3 se tiene que

$$AA^+ = (U\Sigma V^T)(V\Sigma^+U^T) = U\Sigma\Sigma^+U^T.$$

Trasponiendo obtenemos

$$(AA^+)^T = (U\Sigma\Sigma^+U^T)^T = U(\Sigma\Sigma^+)^TU^T = U\Sigma\Sigma^+U^T,$$

donde la última igualdad es cierta por ser $\Sigma\Sigma^+$ matriz diagonal.

Para la última igualdad razonamos de forma análoga al caso anterior,

$$A^+A = (V\Sigma^+U^T)(U\Sigma V^T) = V\Sigma^+\Sigma V^T.$$

Calculando ahora la traspuesta de A^+A se tiene que

$$(A^+A)^T = (V\Sigma^+\Sigma V^T)^T = V(\Sigma^+\Sigma)^T V^T = V\Sigma^+\Sigma V^T,$$

siendo la última igualdad cierta por ser $\Sigma^+\Sigma$ matriz diagonal. □

1.3. Norma y número de condición euclídeo de una matriz

La norma matricial de mayor interés es la deducida de la norma vectorial euclídea. Para definirla introducimos el concepto de radio espectral.

Dada una matriz $A \in \mathbb{R}^{n\times n}$ definimos el radio espectral de dicha matriz como

$$\rho(A) = \max_s |\lambda_s|, \qquad \lambda_s \text{ autovalor de } A.$$

Con esta notación, y teniendo en cuenta la definición general de norma euclídea de una matriz,

$$\|A\|_2 = \sup_{x\neq 0} \frac{\|Ax\|_2}{\|x\|_2} = \max_{\|x\|_2=1} \|Ax\|_2,$$

podemos probar que

$$\|A\|_2 = \sqrt{\rho(A^TA)}.$$

En efecto, sea $x \in \mathbb{R}^n$ no nulo. A^TA tiene una base ortonormal de autovectores $u_j, 1 \leq j \leq n$, es decir, $A^TAu_j = \lambda_j u_j$ donde los λ_j son reales no negativos. Si

$$x = \sum_{j=1}^n \alpha_j u_j$$

es la expresión de x en la base de autovectores, podemos escribir

$$\begin{aligned}
\|Ax\|_2^2 &= (Ax)^TAx = x^TA^TAx = \left(\sum_{j=1}^n \alpha_j u_j^T\right) A^TA \left(\sum_{j=1}^n \alpha_j u_j\right) \\
&= \sum_{j,k=1}^n \alpha_j\alpha_k u_j^TA^TAu_k = \sum_{j,k=1}^n \alpha_j\alpha_k u_j^T\lambda_k u_k \\
&= \sum_{j=1}^n |\alpha_j^2|\lambda_j \leq \rho(A^TA)\sum_{j=1}^n |\alpha_j^2| = \rho(A^TA)\|x\|_2^2,
\end{aligned}$$

de modo que

$$\|A\|_2^2 \leq \rho(A^TA).$$

Para probar la desigualdad contraria, sea $u \neq 0$ un autovector de A^TA tal que el módulo del autovalor correspondiente λ coincida con $\rho(A^TA)$. Tendremos

$$\|Au\|_2^2 = u^TA^TAu = u^T\lambda u = \lambda\|u\|_2^2,$$

donde $\lambda \geq 0$ por ser cociente de cantidades no negativas. Se tiene entonces

$$\|Au\|_2^2 = \rho(A^TA)\|u\|_2^2,$$

por lo que

$$\rho(A^TA) = \frac{\|Au\|_2^2}{\|u\|_2^2} \leq \|A\|_2^2.$$

Veamos ahora cómo calcular el valor de esta norma en el caso de matrices no cuadradas. Sea $A \in \mathbb{R}^{m\times n}$. Se define

$$\|A\|_2 = \sup_{x\neq 0} \frac{\|Ax\|_2}{\|x\|_2},$$

donde la norma que aparece en el numerador es la norma euclídea en $\mathbb{R}^m$ mientras que la que aparece en el denominador es la norma euclídea en $\mathbb{R}^n$.

Teorema 4. *Sea $A \in \mathbb{R}^{m\times n}$ con valores singulares $\sigma_1 \geq \sigma_2 \geq \cdots \geq 0$. Se tiene que*

$$\|A\|_2 = \sigma_1.$$

Demostración. Debemos probar que $\sup_{x\neq 0}(\|Ax\|_2/\|x\|_2) = \sigma_1$. Para ello, basta darse cuenta de que si v_1 es un vector singular por la derecha de A asociado al valor singular σ_1, se tiene

$$\frac{\|Av_1\|_2}{\|v_1\|_2} = \frac{\|\sigma_1 u_1\|_2}{\|v_1\|_2} = \sigma_1\frac{\|u_1\|_2}{\|v_1\|_2} = \sigma_1,$$

por lo que $\sup_{x\neq 0}(\|Ax\|_2/\|x\|_2) \geq \sigma_1$. Ahora debemos probar que cualquier otro vector sufre una magnificación menor o igual que la de v_1.

Sea $x \in \mathbb{R}^n$, y sea $\{v_1, \ldots, v_n\}$ una base ortonormal de autovectores de A^TA. El vector x puede expresarse como una combinación lineal $x = \alpha_1 v_1 + \cdots + \alpha_n v_n$ y $\|x\|_2^2 = |\alpha_1|^2 + \cdots + |\alpha_n|^2$. Multiplicando x por la matriz A, se tiene $Ax = \alpha_1 Av_1 + \cdots + \alpha_r Av_r + \cdots + \alpha_n Av_n = \sigma_1\alpha_1 u_1 + \cdots + \sigma_r\alpha_r u_r$, donde r es el rango de A. Como $u_1, \ldots, u_r$ son también ortonormales, $\|Ax\|_2^2 = |\sigma_1\alpha_1|^2 + \cdots + |\sigma_r\alpha_r|^2$. Por tanto, $\|Ax\|_2^2 \leq \sigma_1^2(|\alpha_1|^2 + \cdots + |\alpha_r|^2) \leq \sigma_1^2\|x\|_2^2$, de donde se obtiene $\|Ax\|_2/\|x\|_2 \leq \sigma_1$. Lo cual determina la igualdad. □

Como los valores singulares de A y A^T coinciden, tenemos el siguiente resultado.

Corolario 1. *Se tiene $\|A\|_2 = \|A^T\|_2$.*

Recordamos ahora la definición del número de condición de una matriz cuadrada invertible $A \in \mathbb{R}^{n\times n}$ como

$$\kappa_2(A) = \|A\|_2\|A^{-1}\|_2.$$

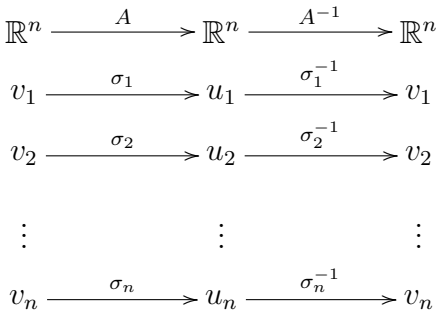


Figura 2: Acción de A y A^{-1}

Veamos cómo expresar $\kappa_2(A)$ en términos de los valores singulares de A. Como el rango de A es n, A y A^{-1} tienen n valores singulares estrictamente positivos y su acción se muestra en el diagrama de la Figura 2.

En términos matriciales tenemos $A = U\Sigma V^T$ y $A^{-1} = V\Sigma^{-1}U^T$, puesto que al ser A una matriz cuadrada su inversa coincide con la pseudoinversa y, por lo visto en la Sección 1.2.1, los valores singulares de A^{-1} son $\sigma_n^{-1} \geq \cdots \geq \sigma_1^{-1}$ siendo $\sigma_i, 1 \leq i \leq n$ los valores singulares de A. Por el Teorema 4 aplicado a A^{-1} se tiene $\|A^{-1}\|_2 = \sigma_n^{-1}$ y, por tanto, se puede escribir el número de condición euclídeo de A como

$$\kappa_2(A) = \frac{\sigma_1}{\sigma_n}. \tag{8}$$

Otra forma alternativa de expresarlo es

$$\kappa_2(A) = \frac{M(A)}{m(A)}, \tag{9}$$

donde

$$M(A) = \sup_{x \neq 0} \frac{\|Ax\|_2}{\|x\|_2} \qquad \text{y} \qquad m(A) = \operatorname{ínf}_{x \neq 0} \frac{\|Ax\|_2}{\|x\|_2}. \tag{10}$$

Si $A \in \mathbb{R}^{m \times n}$, se pueden seguir definiendo las cantidades $M(A)$ y $m(A)$ como en (10), entendiendo que las normas de los numeradores son en $\mathbb{R}^m$ mientras que las de los denominadores son en $\mathbb{R}^n$. Si el rango de A es n entonces, $m(A) > 0$ y se puede adoptar (9) como definición del número de condición de A. Si el rango de A es $r < n$, $m(A) = 0$ y, por convenio, se define $\kappa_2(A) = \infty$, puesto que el cociente (9) no tiene ningún sentido.

Enunciamos así el siguiente teorema que recoge estos resultados.

Teorema 5. *Sea $A \in \mathbb{R}^{m \times n}$. Se verifica:*

1. *$M(A) = \sigma_1$ y $m(A) = \sigma_n$.*
2. *Si $\sigma_n \neq 0$, entonces $\kappa_2(A) = \dfrac{\sigma_1}{\sigma_n}$ (como para matrices cuadradas).*
3. *Si $\sigma_n = 0$, por convenio, $\kappa_2(A) = \infty$.*

Demostración. De la expresión dada para $M(A)$ en la primera ecuación de (10) vemos que $M(A) = \|A\|_2 = \sigma_1$.

Basta ver ahora que $m(A)$ coincide con σ_n y, por la definición del número de condición dada en (9) extendida a matrices no cuadradas, habremos terminado. Sea v_n un vector singular por la derecha de A asociado al valor singular σ_n. Entonces

$$\frac{\|Av_n\|_2}{\|v_n\|_2} = \frac{\|\sigma_n u_n\|_2}{\|v_n\|_2} = \sigma_n,$$

donde la última igualdad es cierta por ser u_n y v_n unitarios. Podemos concluir, por tanto, que $\text{ínf}_{x\neq 0}\left(\|Ax\|_2/\|x\|_2\right) \leq \sigma_n$.

Para probar la desigualdad contraria, tenemos que ver que cualquier otro vector sufre una magnificación mayor o igual que la de v_n. Sea $x \in \mathbb{R}^n$, luego x se puede escribir como combinación lineal de autovectores de A^TA, $x = \alpha_1 v_1 + \cdots + \alpha_n v_n$. Como $v_1, \ldots, v_n$ son vectores ortonormales, $\|x\|_2^2 = |\alpha_1|^2 + \cdots + |\alpha_n|^2$. Multiplicando x por la matriz A, se tiene $Ax = \alpha_1 Av_1 + \cdots + \alpha_r Av_r + \cdots + \alpha_n Av_n = \sigma_1\alpha_1 u_1 + \cdots + \sigma_n\alpha_n u_n$. Como $u_1, \ldots, u_n$ son también ortonormales, $\|Ax\|_2^2 = |\sigma_1\alpha_1|^2 + \cdots + |\sigma_n\alpha_n|^2$. Por tanto, $\|Ax\|_2^2 \geq \sigma_n^2(|\alpha_1|^2 + \cdots + |\alpha_n|^2) = \sigma_n^2\|x\|_2^2$, de donde se obtiene $\|Ax\|_2/\|x\|_2 \geq \sigma_n$.

Si $\sigma_n = 0$, tomamos v_n en el núcleo de A, se tiene así que $\|Av_n\|_2 = 0$ y, por tanto, $\text{ínf}_{x\neq 0}\left(\|Ax\|_2/\|x\|_2\right) = 0$. □

1.3.1. Imagen de la esfera n-dimensional de radio unidad

Veamos ahora cómo la descomposición en valores singulares de $A \in \mathbb{R}^{n\times n}$ permite determinar cuál es la imagen de la esfera n-dimensional de radio 1, $S^n = \{x \in \mathbb{R}^n \mid \|x\|_2 = 1\}$, a través de la transformación lineal $T(x) = Ax$.

Si $A \in \mathbb{R}^{n\times n}$ es una matriz con descomposición en valores singulares $A = U\Sigma V^T$ y rango r, queremos determinar el conjunto

$$T(S^n) = \{z \in \mathbb{R}^n \mid z = Ax, \quad x \in S^n\}.$$

Para ello consideremos en primer lugar el cambio de variable $x = Vy$, donde $\|x\|_2 = \|y\|_2$ por ser V ortogonal. Se tiene entonces que

$$z \in T(S^n) \Leftrightarrow z = Ax \text{ con } \|x\|_2 = 1 \Leftrightarrow z = AVy \text{ con } \|y\|_2 = 1 \Leftrightarrow z = U\Sigma y \text{ con } \|y\|_2 = 1.$$

Teniendo en cuenta ahora la estructura de la matriz Σ y denotando por $u_1, \ldots, u_n$ a las columnas de U se tiene que $U\Sigma y = \sigma_1 y_1 u_1 + \cdots + \sigma_r y_r u_r$, si A tiene rango r.

Entonces

$$z \in T(S^n) \Leftrightarrow z = \sigma_1 y_1 u_1 + \cdots + \sigma_r y_r u_r \quad \text{con} \quad \|y\|_2 = 1.$$

Sean $z_1, \ldots, z_n$ las coordenadas de z en la base de $\mathbb{R}^n$ formada por los autovectores de

AA^T. Se tiene entonces que

$$z \in T(S^n) \Leftrightarrow \begin{cases} z_1 &= \sigma_1 y_1 \\ \vdots & \vdots \quad \vdots \\ z_r &= \sigma_r y_r \\ z_{r+1} &= 0 \\ \vdots & \vdots \quad \vdots \\ z_n &= 0 \end{cases} \quad \text{con} \quad \|y\|_2 = 1.$$

Como $y_i = z_i/\sigma_i$ para $i = 1, \ldots, r$ y $\|y\|_2 = 1$ se concluye que:

Si $r = n$,

$$z \in T(S^n) \Leftrightarrow \frac{z_1^2}{\sigma_1^2} + \cdots + \frac{z_n^2}{\sigma_n^2} = 1,$$

con lo cual $T(S^n)$ resulta ser un elipsoide n-dimensional (si $n = 1$ es un par de puntos, y si $n = 2$ es una elipse) contenido en el subespacio generado por $\{u_1, \ldots, u_n\}$, y que tiene por ejes a las rectas generadas por los vectores $u_1, \ldots, u_n$ siendo la longitud de los semiejes los valores singulares de A, $\sigma_1, \ldots, \sigma_n$. La dispersión del elipsoide con respecto a la esfera inicial viene dada por el número de condición $\kappa_2(A)$.

Si $r < n$,

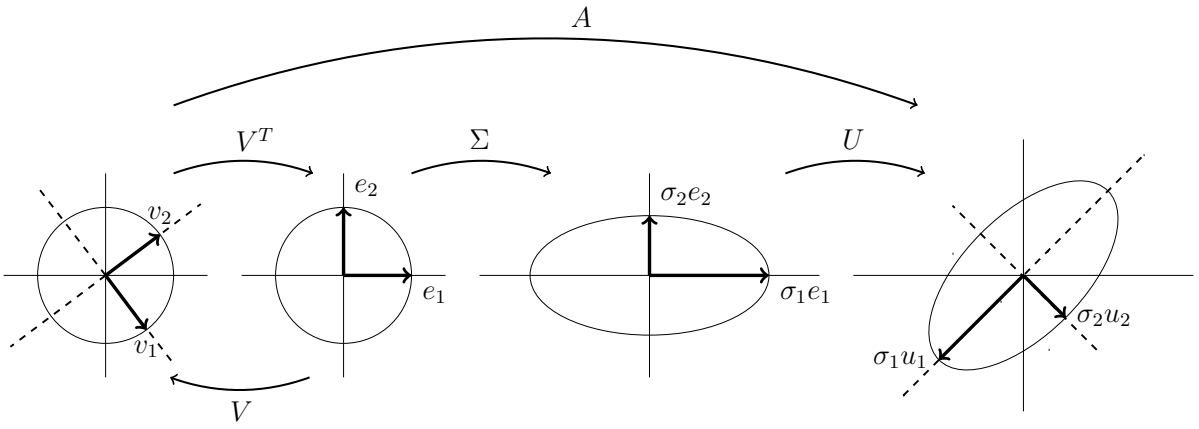
$$\frac{z_1^2}{\sigma_1^2} + \cdots + \frac{z_r^2}{\sigma_r^2} = \sum_{i=1}^{r} y_i^2 \leq 1,$$

y, por lo tanto,

$$z \in T(S^n) \Leftrightarrow \frac{z_1^2}{\sigma_1^2} + \cdots + \frac{z_r^2}{\sigma_r^2} \leq 1 \quad \text{y} \quad z_{r+1} = \cdots = z_n = 0.$$

En este caso $T(S^n)$ resulta ser un elipsoide r-dimensional (si $r = 1$ es un par de puntos, y si $r = 2$ es una elipse) contenido en el subespacio generado por $\{u_1, \ldots, u_n\}$, y que tiene por ejes a las rectas generadas por los vectores $u_1, \ldots, u_r$ siendo la longitud de los semiejes los valores singulares de A, $\sigma_1, \ldots, \sigma_r$. La dispersión del elipsoide con respecto a la esfera inicial viene dada por $\dfrac{\sigma_1}{\sigma_r}$.

Veamos esta interpretación con un ejemplo en el caso de una matriz $A \in \mathbb{R}^{2\times 2}$ donde la circunferencia de radio unidad se transforma en una elipse [8].



La matriz utilizada para la transformación ha sido la matriz $A \in \mathbb{R}^{2\times 2}$ con descomposición en valores singulares dada por

$$A = \begin{pmatrix} -2 & 11 \\ -10 & 5 \end{pmatrix} = \begin{pmatrix} -1/\sqrt{2} & -1/\sqrt{2} \\ -1/\sqrt{2} & -1/\sqrt{2} \end{pmatrix} \begin{pmatrix} 10\sqrt{2} & 0 \\ 0 & 5\sqrt{2} \end{pmatrix} \begin{pmatrix} 3/5 & 4/5 \\ -4/5 & 3/5 \end{pmatrix}^T.$$

1.4. Determinación del rango numérico de una matriz

En ausencia de errores de redondeo y de perturbaciones en la matriz, la descomposición en valores singulares de la misma, revela el rango de dicha matriz. Dada una matriz $A \in \mathbb{R}^{m\times n}$ de rango estrictamente menor que mín(m, n), si se efectúa una pequeña perturbación en sus elementos, generalmente se incrementará el rango de la matriz.

Teorema 6. *Sea $A \in \mathbb{R}^{m\times n}$ de rango $r > 0$ y sea $A = U\Sigma V^T$ su descomposición en valores singulares. Definimos para $k = 1, \ldots, r-1$, $A_k = U\Sigma_k V^T$ donde Σ_k está dada por*

$$\Sigma_k = \left(\begin{array}{cccc|c} \sigma_1 & 0 & \ldots & 0 & \\ 0 & \sigma_2 & \ldots & 0 & \\ \vdots & \vdots & \ddots & \vdots & O \\ 0 & 0 & \ldots & \sigma_k & \\ \hline \multicolumn{4}{c|}{O} & O \end{array}\right) \in \mathbb{R}^{m\times n}. \tag{11}$$

El rango de A_k es k y $\|A - A_k\|_2 = \text{mín}\{\|A - B\|_2 \mid rango(B) = k\} = \sigma_{k+1}$. Es decir, de todas las matrices de rango k, A_k es la más cercana a A, cuando la distancia entre matrices se mide utilizando la norma euclídea.

Demostración. Por la descomposición en valores singulares de A y A_k se tiene que

$$A - A_k = U\Sigma V^T - U\Sigma_k V^T = U(\Sigma V^T - \Sigma_k V^T) = U(\Sigma - \Sigma_k)V^T.$$

Teniendo en cuenta la estructura de Σ dada en (2) y la de Σ_k se tiene que

$$\Sigma - \Sigma_k = \left(\begin{array}{ccccccc|c} 0 & 0 & & \dots & & 0 & & \\ 0 & \ddots & & & & & & \\ & & 0 & & & \vdots & & O \\ \vdots & & & \sigma_{k+1} & & & & \\ & & & & \ddots & 0 & & \\ 0 & & \dots & & 0 & \sigma_r & & \\ \hline & & & O & & & & O \end{array}\right),$$

donde σ_{k+1} es el valor singular más grande de $A-A_k$ y por el Teorema 4, $\|A-A_k\|_2 = \sigma_{k+1}$. Veamos ahora que para cualquier otra matriz B de rango k, $\|A - B\|_2 \geq \sigma_{k+1}$.

Dada una matriz $B \in \mathbb{R}^{m\times n}$ de rango k, el núcleo de B tiene dimensión $n-k$. Si $v_1, \dots, v_n$ denotan las columnas de V, el subespacio vectorial generado por $v_1, \dots, v_{k+1}$ tiene dimension $k+1$. Como tanto el núcleo de B como $< v_1, \dots, v_{k+1} >$ son subespacios de $\mathbb{R}^n$, y la suma de sus dimensiones es mayor que n, la intersección de ambos subespacios es no nula. Sea $x \in Ker(B) \bigcap < v_1, \dots, v_{k+1} >$, y supongamos, sin pérdida de generalidad, que $\|x\|_2 = 1$. Como $x \in < v_1, \dots, v_{k+1} >$, existen escalares $\alpha_1, \dots, \alpha_{k+1}$ tales que $x = \alpha_1 v_1 + \dots + \alpha_{k+1} v_{k+1}$. Por la ortonormalidad de $v_1, \dots, v_{k+1}$, $\|x\|_2^2 = |\alpha_1|^2 + \dots + |\alpha_{k+1}|^2 = 1$ y por pertencer x al $Ker(B)$, $Bx = 0$.

Además

$$(A-B)x = Ax - Bx = Ax = \sum_{i=1}^{k+1} \alpha_i A v_i = \sum_{i=1}^{k+1} \sigma_i \alpha_i u_i,$$

donde $u_1, \dots, u_{k+1}$ son también ortonormales. Se tiene entonces

$$\|(A-B)x\|_2^2 = \sum_{i=1}^{k+1} |\sigma_i \alpha_i|^2 \geq \sigma_{k+1}^2 \sum_{i=1}^{k+1} |\alpha_i|^2 = \sigma_{k+1}^2.$$

Por tanto,

$$\|A - B\|_2 \geq \frac{\|(A-B)x\|_2}{\|x\|_2} \geq \sigma_{k+1}.$$

□

Corolario 2. *Sea $A \in \mathbb{R}^{m\times n}$ de rango n y sean $\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_n > 0$ sus valores singulares. Si $B \in \mathbb{R}^{m\times n}$ satisface que $\|A - B\|_2 < \sigma_n$, entonces el rango de B también es n.*

Demostración. Supongamos $rango(B) = k < n$ entonces

$$\|A - B\|_2 \geq \|A - A_k\|_2 = \sigma_{k+1} \geq \sigma_n$$

lo cual contradice el que $\|A - B\|_2 < \sigma_n$. □

Del Corolario 2 se deduce que las matrices que están suficientemente próximas respecto de la norma euclídea, tienen el mismo rango.

Para determinar el rango numérico de una matriz A, tomamos un valor ϵ positivo, que representa la magnitud de incertidumbre sobre los elementos de A. Si existe una matriz B de rango k tal que $\|A - B\|_2 < \epsilon$ y, para cada matriz C de rango $\leq k-1$ tenemos $\|A - C\|_2 \gg \epsilon$, diremos que el rango de la matriz A es k. Por el Teorema 6, sabemos que esta condición es satisfecha si, y solo si, los valores singulares de A satisfacen $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_k \gg \epsilon > \sigma_{k+1} \geq \cdots$ Por tanto el rango numérico puede determinarse examinando los valores singulares de la matriz. Una matriz que tiene k valores singulares grandes y los restantes muy pequeños, tiene rango numérico k. Sin embargo, si en el conjunto de valores singulares no hay una diferencia de tamaño destacada entre dos de ellos en comparación con el resto, puede ser imposible asignar un rango numérico significativo a la matriz.

Presentamos ahora una aplicación del Teorema 6 para el caso de matrices cuadradas.

Teorema 7. *Sea $A \in \mathbb{R}^{n\times n}$ una matriz no singular, con valores singulares $\sigma_1 \geq \cdots \geq \sigma_n > 0$ y sea A_s la matriz singular más próxima a A en el sentido de minimizar $\|A - A_s\|_2$. Entonces $\|A - A_s\|_2 = \sigma_n$, y*

$$\frac{\|A - A_s\|_2}{\|A\|_2} = \frac{1}{\kappa_2(A)}.$$

La demostración es inmediata de los Teoremas 4 y 6 y de la definición del número de condición dada en (8).

El Teorema 7 indica que la distancia de A a la matriz singular más próxima a ella es el menor valor singular de A, y la distancia relativa respecto de la matriz singular más próxima es igual al inverso del número de condición de A.

1.5. Teorema de aproximación en norma de Frobenius

Otra norma matricial que se puede ralacionar con la descomposición en valores singulares de una matriz $A \in \mathbb{R}^{m\times n}$, es la conocida como norma de Frobenius, que viene dada por

$$\|A\|_F = \left(\sum_{i=1}^{m} \sum_{j=1}^{n} |a_{ij}|^2 \right)^{1/2}.$$

Es fácil ver que en términos de los valores singulares de la matriz A podemos expresar dicha norma como

$$\|A\|_F = (\sigma_1^2 + \cdots + \sigma_r^2)^{1/2}, \tag{12}$$

si el rango de A es r. Para demostrar (12) veamos primero que $\|A\|_F^2 = tr(A^T A)$ y, en consecuencia, que si Q es una matriz ortogonal, $\|QA\|_F = \|A\|_F$.

Notemos que el elemento diagonal i-ésimo de $A^T A$ viene dado por

$$(A^T A)_{ii} = \sum_{k=1}^{m} |a_{ki}|^2 = \sum_{k=1}^{m} a_{ki}^2,$$

donde la última igualdad es cierta por ser A real, y, por tanto,

$$tr(A^TA) = \sum_{i=1}^{n}(A^TA)_{ii} = \sum_{i=1}^{n}\sum_{k=1}^{m} a_{ki}^2 = \|A\|_F^2.$$

Veamos ahora que si Q es ortogonal $\|QA\|_F = \|A\|_F$

$$\|QA\|_F = \sqrt{tr((QA)^T(QA))} = \sqrt{tr(A^TQ^TQA)} = \sqrt{tr(A^TA)} = \|A\|_F.$$

Por el Teorema 2, sabemos que $A = U\Sigma V^T$ con U, V y, en consecuencia, V^T ortogonales. Podemos escribir entonces

$$\|A\|_F = \|U\Sigma V^T\|_F = \|\Sigma V^T\|_F = \|\Sigma\|_F = \left(\sum_{i=1}^{r}|\sigma_i|^2\right)^{1/2} = (\sigma_1^2 + \cdots + \sigma_r^2)^{1/2}.$$

Como $\|A\|_F = \|A^T\|_F$, también se puede utilizar $\|A\|_F^2 = tr(AA^T)$.

De forma análoga al resultado sobre la mejor aproximación a una matriz por matrices de rango dado cuando la distancia entre matrices se mide con la norma euclídea, podemos enunciar otro resultado cuando la distancia entre matrices se mide utilizando la norma de Frobenius [3].

Teorema 8. *Sea $A \in \mathbb{R}^{m\times n}$ de rango $r > 0$ y sea $A = U\Sigma V^T$ su descomposición en valores singulares. Definimos para $k = 1, \ldots, r-1$, $A_k = U\Sigma_k V^T$ donde Σ_k está definida como en (11). Se verifica entonces que:*

$$\|A - A_k\|_F = \text{mín}\{\|A - B\|_F \mid rango(B) = k\} = \sqrt{\sigma_{k+1}^2 + \cdots + \sigma_r^2}.$$

Demostración. Como vimos en la demostración del Teorema 6, teniendo en cuenta la descomposición en valores singulares de A y A_k, se tiene

$$A - A_k = U(\Sigma - \Sigma_k)V^T,$$

donde $\Sigma - \Sigma_k$ es igual que en dicho teorema y, por las propiedades de la norma de Frobenius, se tiene que

$$\|A - A_k\|_F = \|\Sigma - \Sigma_k\|_F = \sqrt{\sigma_{k+1}^2 + \cdots + \sigma_r^2}.$$

Veamos ahora que si B es otra matriz de rango k, $\|A - B\|_F \geq \sqrt{\sigma_{k+1}^2 + \cdots + \sigma_r^2}$.

Utilizando la descomposición en valores singulares de B, se puede expresar B como

$$B = \sum_{i=1}^{k} x_i y_i^T$$

donde los vectores y_i se pueden elegir ortonormales. Los vectores x_i son ortogonales pero de norma euclídea igual a $\widetilde{\sigma}_i$ siendo $\widetilde{\sigma}_1, \ldots, \widetilde{\sigma}_k$ los valores singulares de B. Se verifica entonces que [7]

$$\begin{aligned}
\|A-B\|_F^2 &= tr\left((A-\sum_{i=1}^{k} x_i y_i^T)(A-\sum_{i=1}^{k} x_i y_i^T)^T\right) \\
&= tr\left(AA^T + \sum_{i=1}^{k}(x_i - Ay_i)(x_i - Ay_i)^T - \sum_{i=1}^{k} Ay_i y_i^T A^T\right) \\
&= \|A\|_F^2 + \sum_{i=1}^{k}\|x_i - Ay_i\|_F^2 - \sum_{i=1}^{k}\|Ay_i\|_F^2.
\end{aligned}$$

Por ser el segundo sumando ≥ 0, basta ver entonces que $\sum_{i=1}^{k}\|Ay_i\|_F^2 \leq \sum_{i=1}^{k}\sigma_i^2$.

De esta forma se tendrá que

$$\begin{aligned}
\|A-B\|_F^2 &= \|A\|_F^2 + \sum_{i=1}^{k}\|x_i - Ay_i\|_F^2 - \sum_{i=1}^{k}\|Ay_i\|_F^2 \\
&\geq \sum_{i=1}^{r}\sigma_i^2 + \sum_{i=1}^{k}\|x_i - Ay_i\|_F^2 - \sum_{i=1}^{k}\sigma_i^2 \\
&\geq \sum_{i=k+1}^{r}\sigma_i^2 = \|A-A_k\|_F^2.
\end{aligned}$$

Teniendo en cuenta la descomposición en valores singulares de la matriz A y particionando adecuadamente las matrices V y Σ de modo que $V = (V_1|V_2)$ con V_1 formada por las k primeras columnas de V se verifica que

$$\|Ay_i\|_F^2 = \|U\Sigma V^T y_i\|_F^2 = \|\Sigma V^T y_i\|_F^2 = \|\Sigma_1 V_1^T y_i\|_F^2 + \|\Sigma_2 V_2^T y_i\|_F^2.$$

Manipulando en esta expresión y teniendo en cuenta que $\sigma_k^2\|V^T y_i\|_F^2 = \sigma_k^2\|V_1^T y_i\|_F^2 + \sigma_k^2\|V_2^T y_i\|_F^2$, se obtiene que

$$\begin{aligned}
\|Ay_i\|_F^2 &= \sigma_k^2\|V^T y_i\|_F^2 + \left(\|\Sigma_1 V_1^T y_i\|_F^2 - \sigma_k^2\|V_1^T y_i\|_F^2\right) \\
&+ \left(\|\Sigma_2 V_2^T y_i\|_F^2 - \sigma_k^2\|V_2^T y_i\|_F^2\right).
\end{aligned}$$

Como los elementos diagonales de Σ_2 son $\leq \sigma_k$, la cantidad que aparece en la segunda línea de la relación anterior es ≤ 0. Por otra parte, como V es ortogonal y los vectores y_i son ortonormales, $\|V^T y_i\|_F^2 = 1$. Por lo tanto,

$$\|Ay_i\|_F^2 \leq \sigma_k^2 + \left(\|\Sigma_1 V_1 y_i\|_F^2 - \sigma_k\|V_1^T y_i\|_F^2\right) \leq \sigma_k^2 + \sum_{j=1}^{k}(\sigma_j^2 - \sigma_k^2)|v_j^T y_i|^2.$$

En consecuencia,

$$\begin{aligned}\sum_{i=1}^{k}\|Ay_i\|_F^2 &\leq k\sigma_k^2+\sum_{i=1}^{k}\left(\sum_{j=1}^{k}(\sigma_j^2-\sigma_k^2)|v_j^Ty_i|^2\right)\\ &= \sum_{j=1}^{k}\left(\sigma_k^2+(\sigma_j^2-\sigma_k^2)\sum_{i=1}^{k}|v_j^Ty_i|^2\right)\\ &\leq \sum_{j=1}^{k}(\sigma_k^2+(\sigma_j^2-\sigma_k^2)\|v_j\|_F^2)=\sum_{j=1}^{k}\sigma_j^2.\end{aligned}$$

□

1.6. Caracterización variacional de los valores singulares

Para presentar la caracterización variacional de los valores singulares, tengamos en cuenta que la caracterización variacional en el caso de los autovalores de una matriz es

$$\lambda_1=\max_{v\in\mathbb{R}^n}\left(\frac{v^TA^TAv}{v^Tv}\right)=\max_{u\in\mathbb{R}^m}\left(\frac{u^TAA^Tu}{u^Tu}\right),$$

siendo λ_1 el mayor autovalor de AA^T y de A^TA. Como los valores singulares de A son la raíz cuadrada de los autovalores de AA^T y de A^TA, estos heredan la caracterización variacional definida anteriormente, teniendo así,

$$\sigma_1=\max_{v\in\mathbb{R}^n}\left(\frac{v^TA^TAv}{v^Tv}\right)^{1/2}=\max_{u\in\mathbb{R}^m}\left(\frac{u^TAA^Tu}{u^Tu}\right)^{1/2},$$

siendo v_1 y u_1 los vectores singulares por la derecha y por la izquierda, respectivamente, donde se alcanzan estos máximos.

Los valores singulares también satisfacen una sutil propiedad variacional en la que intervienen ambos vectores singulares, por la derecha y por la izquierda, al mismo tiempo. Consideramos para ello vectores unitarios $v\in\mathbb{R}^n$ y $u\in\mathbb{R}^m$. Se tiene que

$$|u^TAv|\leq\|u\|_2\|A\|_2\|v\|_2=\sigma_1,$$

donde la desigualdad es cierta por la desigualdad de Cauchy-Schwarz y la igualdad de la derecha viene dada por la definición de norma euclídea de A y el carácter unitario de los vectores u y v. Por otro lado, si u_1 y v_1 son los vectores singulares por la derecha y por la izquierda, respectivamente, asociados al valor singular σ_1, se tiene

$$|u_1^TAv_1|=|u_1^T(\sigma_1u_1)|=\sigma_1.$$

Por tanto

$$\sigma_1=\max_{u\in\mathbb{R}^m,v\in\mathbb{R}^n}\frac{|u^TAv|}{\|u\|_2\|v\|_2}.$$

Podemos caracterizar los valores singulares menores que σ_1 de modo análogo.

Teniendo en cuenta la descomposición en valores singulares como suma de matrices de rango uno dada en (5), y restringiendo la elección de los vectores unitarios u y v a vectores ortogonales a u_1 y v_1, respectivamente, se tiene

$$u^T A v = u^T \left(\sum_{j=1}^{r} \sigma_j u_j v_j^T \right) v = \sum_{j=1}^{r} \sigma_j u^T u_j v_j^T v = u^T \left(\sum_{j=2}^{r} \sigma_j u_j v_j^T \right) v.$$

Por lo tanto

$$|u^T A v| \leq \sigma_2,$$

alcanzándose la igualdad si $u = u_2$ y $v = v_2$. Continuando el proceso podemos enunciar el siguiente teorema.

Teorema 9. *Sea $A \in \mathbb{R}^{m \times n}$ y σ_k su k-ésimo valor singular. Entonces*

$$\sigma_k = \max_{\substack{u \in < u_1, \ldots, u_{k-1} >^{\perp} \\ v \in < v_1, \ldots, v_{k-1} >^{\perp}}} \frac{|u^T A v|}{\|u\|_2 \|v\|_2}, \quad k > 1.$$

2. Cálculo numérico eficiente de la descomposición en valores singulares

La Figura 1 muestra la acción de A y A^T sobre los vectores singulares por la izquierda y por la derecha de A y A^T. Hemos visto en la Sección 1.1 cómo calcular los valores y vectores singulares de A a partir de la construcción de A^TA (o de AA^T) y del posterior cálculo de sus autovalores y autovectores. El principal inconveniente de utilizar algoritmos numéricos que implementan este procedimiento es que el cálculo de los valores singulares más pequeños puede ser erróneo.

Por ejemplo, si se conocen los coeficientes de una matriz A con 6 decimales nos gustaría poder hallar sus valores singulares con un nivel de error $\varepsilon \approx 10^{-6}$. Si dicha matriz tiene, por ejemplo, valores singulares $\sigma_1 \approx 1$ y $\sigma_r \approx 10^{-3}$, entonces aunque σ_r es muy pequeño comparado con σ_1, esperamos calcular σ_r con una precisión de 2 o 3 decimales. Sin embargo, si se calcula A^TA (o AA^T), cuyos coeficientes también se conocerán con 6 decimales correctos, los autovalores de A^TA asociados a los valores singulares σ_1 y σ_r serán $\lambda_1 \approx 1$ y $\lambda_r \approx 10^{-6}$, por lo que no es esperable obtener λ_r de forma correcta al ser del mismo tamaño que el error en los datos.

Por tanto, en general, es preciso implementar algoritmos numéricos que proporcionen aproximaciones a los valores y vectores singulares de una matriz sin tener que construir las matrices A^TA (o AA^T).

En la descomposición en valores singulares de $A = U\Sigma V^T$, las matrices U y V deben ser ortogonales. Para la obtención de esta descomposición de A, podemos aplicar transformaciones ortogonales por la derecha y por la izquierda a la matriz A intentando llevarla a una matriz con elementos nulos fuera de la diagonal principal. Veamos la implementación de este algoritmo para el caso de matrices no dispersas [4], [5], [9], [10].

Continuamos con el tratamiento de matrices $A \in \mathbb{R}^{m\times n}$ con $m \geq n$. Si $m < n$, en lugar de trabajar con A se trabaja con su matriz traspuesta.

2.1. Matrices ortogonales

En esta sección estudiamos dos tipos de matrices ortogonales que se van a utilizar para hallar la descomposición en valores singulares de una matriz.

2.1.1. Los reflectores de Householder

Sea $v \in \mathbb{R}^d$ con $v \neq 0$. Se define el reflector de Householder asociado a v como la matriz de $\mathbb{R}^{d\times d}$

$$H_v = I_d - \frac{2}{v^Tv}vv^T.$$

Si tenemos en cuenta que $v^Tv = \|v\|_2^2$ resulta que

$$H_v = I_d - 2\frac{v}{\|v\|_2}\frac{v^T}{\|v\|_2} = H_w, \quad \text{con } w = \frac{v}{\|v\|_2},$$

viendo así que la transformación depende únicamente de la dirección del vector v y no de su módulo.

Estas transformaciones son conocidas como reflectores de Householder, debido a que la aplicación de $\mathbb{R}^d$ en sí mismo

$$x \longrightarrow H_v x = (I_d - \frac{2}{v^T v} v v^T)x = x - \frac{2}{v^T v}(v^T x)v = x - \frac{v^T x}{v^T v}v - \frac{v^T x}{v^T v}v \qquad (13)$$

es de hecho una reflexión respecto del subespacio

$$v^\perp = \{u \in \mathbb{R}^d \mid v^T u = 0\}$$

de los vectores ortogonales a v, puesto que $\frac{v^T x}{v^T v}v$ es la proyección ortogonal de x sobre v, y, por tanto, $x - \frac{v^T x}{v^T v}v \in v^\perp$.

La expresión (13) nos dice que la imagen de un vector cualquiera se obtiene restándole el doble de su proyección ortogonal sobre v. En particular $H_v v = -v$ y $H_v u = u$ para cada vector $u \in v^\perp$.

Entre las propiedades de los reflectores, cabe destacar que H_v es una perturbación de rango 1 de la matriz identidad de orden d, que $H_v^2 = I_d$ y, por tanto, $H_v = H_v^{-1}$. Como H_v es simétrica, entonces H_v es ortogonal y conserva la norma euclídea. Además el determinante de H_v es -1.

Los reflectores de Householder también verifican el siguiente resultado

Teorema 10. *Si $a, b \in \mathbb{R}^d$ son tales que $\|a\|_2 = \|b\|_2 \neq 0$ resulta que*

$$H_{a-b} a = b.$$

Demostración. Busquemos un vector v tal que $b = H_v a$. Como $b = a - \frac{2}{v^T v}(v^T a)v$, debe ser $\frac{2}{v^T v}(v^T a)v = a - b$, y esta ha de ser la dirección de cualquier vector cuyo reflector transforme a en b. □

Teorema 11. *Sea $x \in \mathbb{R}^d$ arbitrario, $e_1 = (1, 0, \ldots, 0)^T \in \mathbb{R}^d$ y $\sigma = \|x\|_2$. Si se define $v = x \pm \sigma e_1$ entonces $H_v x = \mp \sigma e_1$.*

Demostración. Aplicamos el Teorema 10, tomando $b = \mp\sigma e_1$ y $a = x$. Para que H_v transforme a en b, se debe elegir entonces $v = x - (\mp\sigma e_1) = x \pm \sigma e_1$. □

En la práctica se elige v en la dirección de $x \pm \sigma e_1$ normalizado, para que su primera componente sea igual a 1, $v_1 = 1$. Esto permite almacenar el valor de las componentes $v_2, v_3, \ldots, v_d$ en las posiciones donde se van a introducir los ceros de $H_v x$. Además, el signo $\pm$ se elegirá atendiendo al signo de la primera componente de x para evitar posibles pérdidas de precisión por la resta de cantidades próximas.

La siguiente función en MATLAB toma como dato un vector x y devuelve en la salida el vector v normalizado con la primera componente igual a 1 tal que el producto $H_v x$ tiene las últimas $d-1$ componentes nulas.

```
function v = house(x);
d = length(x);
sigma = norm(x,2);
v = x;
v(1) = x(1)+sign(x(1))*sigma;
v(2:d) = v(2:d)/v(1);
v(1) = 1.0;
end
```

El costo operativo $C_1(d)$ de esta subrutina para un vector $x \in \mathbb{R}^d$ teniendo en cuenta solo las multiplicaciones y divisiones, es el siguiente

d productos en el cálculo de la norma euclídea de `x`.

$d-1$ divisiones para el cálculo de `v(2:d)`.

En total $C_1(d) = 2d-1$ multiplicaciones/divisiones.

Para multiplicar un reflector de Householder $H_v \in \mathbb{R}^{d\times d}$ por una matriz $A \in \mathbb{R}^{d\times l}$, se tiene en cuenta que

$$H_v A = (I_d - \frac{2}{v^T v} v v^T)A = A - \frac{2}{v^T v} v (A^T v)^T,$$

es decir, no es preciso construir H_v para hallar el producto $H_v A$.

La función en MATLAB que se incluye a continuación toma como dato de entrada una matriz $A \in \mathbb{R}^{d\times l}$ y un vector $v \in \mathbb{R}^d$ y devuelve en la salida el producto $H_v A$, que se sobreescribe en A. Dicho producto está calculado sin construir explícitamente la matriz H_v.

```
function A = houseproducto(A,v);
beta = -2/(v'*v);
w = beta*(A'*v);
A = A+v*w';
end
```

El costo operativo $C_2(d,l)$ de ejecutar esta función, para una matriz $A \in \mathbb{R}^{d\times l}$, contando solo multiplicaciones y divisiones, viene dado por

d productos y 1 división en el cálculo de `beta`.

$l \times (d+1)$ productos en el cálculo de `w`.

$d \times l$ productos en el cálculo de `v*w'`.

En total $C_2(d,l) = d(1+2l)+1+l$ productos/divisiones.

2.1.2. Las rotaciones de Givens

Hemos visto que los reflectores de Householder son unas transformaciones ortogonales que sirven para introducir ceros en una matriz a gran escala. Sin embargo, su uso no es eficiente cuando la introducción de ceros en una matriz se debe hacer de forma más selectiva. Para ello se introducen las transformaciones de Givens. Dados un par de índices i, k con $1 \leq i < k \leq d$ y un ángulo $\theta \in \mathbb{R}$, se define la matriz de Givens

$$G(i,k,\theta) = \begin{pmatrix} 1 & \cdots & 0 & \cdots & 0 & \cdots & 0 \\ \vdots & & \vdots & & \vdots & & \vdots \\ 0 & \cdots & c & \cdots & s & \cdots & 0 \\ \vdots & & \vdots & & \vdots & & \vdots \\ 0 & \cdots & -s & \cdots & c & \cdots & 0 \\ \vdots & & \vdots & & \vdots & & \vdots \\ 0 & \cdots & 0 & \cdots & 0 & \cdots & 1 \end{pmatrix} \in \mathbb{R}^{d\times d},$$

donde $c = \cos(\theta)$ aparece en las posiciones (i,i) y (k,k), $s = \sin(\theta)$ aparece en la posición (i,k) y $-s$ en la posición (k,i). Una rotación de Givens es una perturbación de rango 2 de la matriz identidad, a menos que θ sea un múltiplo entero de 2π en cuyo caso la rotación de Givens coincide con la identidad. Estas transformaciones son ortogonales, pues corresponden a giros en el plano (i,k). Más precisamente, si $x \in \mathbb{R}^d$, el vector $y = G(i,k,\theta)^T x$ se obtiene aplicando a x una rotación de θ radianes en el sentido contrario a las agujas del reloj en el plano (i,k). Además si $x = (x_1, \ldots, x_d)^T$ e $y = (y_1, \ldots, y_d,)^T$,

$$y_j = \begin{cases} cx_i - sx_k & si \;\; j = i, \\ sx_i + cx_k & si \;\; j = k, \\ x_j & si \;\; j \neq i, k. \end{cases} \tag{14}$$

Por tanto, dado $x \in \mathbb{R}^d$ podemos elegir c y s para que y_k sea 0. Basta tomar

$$c = \frac{x_i}{\sqrt{x_i^2 + x_k^2}}, \quad s = \frac{-x_k}{\sqrt{x_i^2 + x_k^2}}.$$

Desde el punto de vista práctico, es preferible implementar un algoritmo que, dados a y b reales calcule los valores $c = \cos(\theta)$ y $s = \sin(\theta)$ para que

$$\begin{pmatrix} c & s \\ -s & c \end{pmatrix}^T \begin{pmatrix} a \\ b \end{pmatrix} = \begin{pmatrix} r \\ 0 \end{pmatrix}. \tag{15}$$

Naturalmente, $r = \pm\sqrt{a^2 + b^2}$.

La siguiente función de MATLAB implementa la elección de c y s de un modo eficiente:

```
function [c, s] = givens(a,b);
if b==0,
  c = 1; s = 0;
```

```
elseif abs(b) > abs(a),
  tau = -a/b; s = 1/sqrt(1+tau*tau); c = s*tau;
else
  tau = -b/a; c = 1/sqrt(1+tau*tau); s = c*tau;
end
```

El costo operativo de esta función si solo se cuentan productos y divisiones es de 4 productos/divisiones.

También es posible aprovechar la estructura dispersa de las matrices de Givens a la hora de multiplicarlas por una matriz dada. Sea $A \in \mathbb{R}^{d\times l}$ y $c = \cos(\theta)$, $s = \sin(\theta)$. Al premultiplicar por $G(i,k,\theta)^T$ solo se modifican las filas i y k de la matriz A, pudiendo sobreescribir en la propia matriz A los nuevos valores como en la siguiente función de MATLAB :

```
function A = givensproducto(A,c,s,i,k);
tau1 = A(i,:);
tau2 = A(k,:);
A(i,:) = c*tau1-s*tau2;
A(k,:) = s*tau1+c*tau2;
end
```

El costo operativo de esta subrutina para una matriz $A \in \mathbb{R}^{d\times l}$ contando solo multiplicaciones y divisiones, viene dado por

$2d$ productos en el cálculo de `A(i,:)`.

$2d$ productos en el cálculo de `A(k,:)`.

En total son $4d$ productos.

Veamos ahora paso a paso, cómo obtener la matriz de valores singulares de una matriz dada.

2.2. Reducción de la matriz a forma bidiagonal

Una matriz $B \in \mathbb{R}^{m\times n}$ con $m \geq n$ es bidiagonal si $b_{i,j} = 0$ si $i > j$ o $i < j-1$. Es decir, una matriz bidiagonal B tiene la forma siguiente

$$\left(\begin{array}{ccccc} * & * & & & \\ & * & * & & O \\ & & * & \ddots & \\ & O & & \ddots & * \\ & & & & * \\ \hline & & O & & \end{array}\right).$$

Aquí y en lo que sigue $*$ denota posibles elementos no nulos de la matriz.

Teorema 12. *Sea $A \in \mathbb{R}^{m\times n}$ con $m \geq n$. Existen matrices ortogonales $\hat{U} \in \mathbb{R}^{m\times m}$ y $\hat{V} \in \mathbb{R}^{n\times n}$, ambas producto de un número finito de reflectores de Householder, y una matriz bidiagonal $B \in \mathbb{R}^{m\times n}$ tales que*

$$A = \hat{U}B\hat{V}^T.$$

La demostración del Teorema 12 es constructiva. El primer paso es crear ceros en la primera columna y fila de A. Sea $\hat{U}_1 \in \mathbb{R}^{m\times m}$ un reflector tal que

$$\hat{U}_1 \begin{pmatrix} a_{11} \\ a_{21} \\ \vdots \\ a_{m1} \end{pmatrix} = \begin{pmatrix} \hat{a}_{11} \\ 0 \\ \vdots \\ 0 \end{pmatrix}.$$

Entonces la primera columna de $\hat{U}_1 A$ está formada por ceros salvo la entrada $(1,1)$. Ahora tomamos $(\hat{a}_{11}, \hat{a}_{12}, \ldots, \hat{a}_{1n})$, la primera fila de $\hat{U}_1 A$, y sea $\hat{V}_1 \in \mathbb{R}^{n\times n}$ una matriz de la forma

$$\left(\begin{array}{c|ccc} 1 & 0 & \ldots & 0 \\ \hline 0 & & & \\ \vdots & & \overline{V}_1 & \\ 0 & & & \end{array}\right),$$

donde $\overline{V}_1$ es un reflector de $\mathbb{R}^{(n-1)\times(n-1)}$ tal que $(\hat{a}_{12}, \hat{a}_{13}, \ldots, \hat{a}_{1n})\,\overline{V}_1 = (*, 0, \ldots, 0)$.

Por ser la primera columna de $\hat{V}_1$, igual a e_1, la primera columna de $\hat{U}_1 A$ no se ve modificada al multiplicar por $\hat{V}_1$ por la derecha. Por tanto, la primera fila de $\hat{U}_1 A \hat{V}_1$ está formada por ceros salvo las dos primeras entradas, teniendo así

$$\hat{U}_1 A \hat{V}_1 = \left(\begin{array}{c|ccc} * & * & \ldots & 0 \\ \hline 0 & & & \\ \vdots & & \hat{A} & \\ 0 & & & \end{array}\right).$$

El segundo paso del algoritmo es análogo al primero pero actuando sobre la submatriz $\hat{A}$. Los reflectores usados en el segundo paso no destruyen los ceros creados en el paso anterior porque $\hat{U}_2$ se define a partir de un reflector de tamaño $(m-1)\times(m-1)$ orlado con la primera fila y columna de la identidad y $\hat{V}_2$ se define a partir de un reflector $(n-2)\times(n-2)$ orlado con las dos primera filas y columnas de la identidad. Tras los dos primeros pasos se tiene

$$\hat{U}_2\hat{U}_1 A \hat{V}_1 \hat{V}_2 = \left(\begin{array}{cc|ccc} * & * & 0 & 0 & \ldots & 0 \\ \hline 0 & * & * & 0 & \ldots & 0 \\ \hline 0 & 0 & & & & \\ \vdots & \vdots & & & \hat{\hat{A}} & \\ 0 & 0 & & & & \end{array}\right).$$

El tercer paso actúa sobre la submatriz $\hat{\hat{A}} \in \mathbb{R}^{(m-2)\times(n-2)}$ y así sucesivamente. Si $m > n$ hay que completar un total de n pasos y si $m = n$ solo se necesitan $n-1$. Tras completar el proceso se obtiene una matriz de la forma

$$\hat{U}_n \cdots \hat{U}_2 \hat{U}_1 A \hat{V}_1 \hat{V}_2 \cdots \hat{V}_{n-2} = \left(\begin{array}{ccccc} * & * & & & \\ & * & * & & O \\ & & * & \ddots & \\ & O & & \ddots & * \\ & & & & * \\ \hline & & O & & \end{array} \right) = B, \tag{16}$$

teniendo en cuenta que en los dos últimos pasos solo se aplican reflectores por la izquierda para hacer ceros en las dos últimas columnas de la matriz y que además si $m = n$, $\hat{U}_n = I_m$. Por tanto, tomando

$$\hat{U} = \hat{U}_1 \hat{U}_2 \cdots \hat{U}_n \quad \text{y} \quad \hat{V} = \hat{V}_1 \hat{V}_2 \cdots \hat{V}_{n-2}$$

y teniendo en cuenta que los reflectores de Householder son matrices simétricas se tiene $\hat{U}^T A \hat{V} = B$, o equivalentemente,

$$A = \hat{U} B \hat{V}^T.$$

En el Apéndice se ha incluído una función en MATLAB que implementa la reducción de una matriz a forma bidiagonal y una visualización gráfica de este proceso.

Costo operativo

A la vista de (16) el costo operativo de la bidiagonalización de una matriz de $\mathbb{R}^{m\times n}$ si sólo se quiere hallar B viene dado por

$$\sum_{k=1}^{n^*} C_1(m-k+1) + \sum_{k=1}^{n^*} C_2(m-k+1, n-k+1) = \sum_{k=m-n^*+1}^{m} C_1(k) + \sum_{k=m-n^*+1}^{m} C_2(k, n+k-m)$$

$$= \sum_{k=m-n^*+1}^{m} (2k-1) + \sum_{k=m-n^*+1}^{m} [k(1+2n-2(m-n)) + 1 + k - (m-n)],$$

donde $n^* = n$ si $m > n$ y $n^* = n-1$ si $m = n$, correspondiente a aplicar los reflectores de Householder por la izquierda y

$$\sum_{k=1}^{n-2} C_1(n-k) + \sum_{k=1}^{n-2} C_2(n-k, m-k) = \sum_{k=2}^{n-1} C_1(k) + \sum_{k=2}^{n-1} C_2(k, k+m-n)$$

$$= \sum_{k=2}^{n-1} (2k-1) + \sum_{k=2}^{n-1} [k(1+2(k+m-n)) + k + m - n + 1]$$

correspondiente a los reflectores aplicados por la derecha. Tras escribir el término general de los sumatorios en potencias de k y teniendo en cuenta que

$$\sum_{k=m-n^*+1}^{m} C(k) = \sum_{k=1}^{m} C(k) - \sum_{k=1}^{m-n^*} C(k) \quad \text{y} \quad \sum_{k=2}^{n-1} C(k) = \sum_{k=1}^{n-1} C(k) - C(1)$$

utilizando fórmulas bien conocidas se llega a un total de $2mnn^* - mn^{*^2} - nn^{*^2} + \frac{2}{3}n^{*^3}$ productos/divisiones correspondientes a aplicar los reflectores de Householder por la izquierda y a un total de $mn^2 - \frac{1}{3}n^3$ productos/divisiones debidos a los reflectores aplicados por la derecha.

Al final del procedimiento, se han aplicado n^* reflectores por la izquierda y $n-2$ por la derecha, siendo el costo operativo $\sim 2mn^2 - \frac{2}{3}n^3$, si $m > n$ y $\sim \frac{4}{3}n^3$ si $m = n$.

En algunas aplicaciones, como en el problema lineal de mínimos cuadrados, $A \in \mathbb{R}^{m\times n}$ con m mucho mayor que n. En este caso es más eficiente realizar la reducción a forma bidiagonal en dos pasos.

En el primer paso se realiza una factorización QR de A

$$A = QR = \left(Q_{mn} \,\middle|\, Q_{m,m-n}\right) \left(\frac{\hat{R}}{O}\right),$$

donde $Q_{mn} \in \mathbb{R}^{m\times n}$, $Q_{m,m-n} \in \mathbb{R}^{m\times(m-n)}$ y $\hat{R} \in \mathbb{R}^{n\times n}$ es triangular superior. Esto involucra multiplicaciones por reflectores solo por el lado izquierdo.

En el segundo paso $\hat{R}$ es reducida a forma bidiagonal $\hat{R} = \hat{\hat{U}}\hat{B}\hat{\hat{V}}^T$, donde todas las matrices son cuadradas de dimensión $n \times n$, teniendo así,

$$A = \left(Q_{mn} \,\middle|\, Q_{m,m-n}\right) \left(\begin{array}{c|c} \hat{\hat{U}} & O \\ \hline O & I_{m-n} \end{array}\right) \left(\frac{\hat{B}}{O}\right) \hat{\hat{V}}^T.$$

Tomando

$$\hat{U} = \left(Q_{mn} \,\middle|\, Q_{m,m-n}\right) \left(\begin{array}{c|c} \hat{\hat{U}} & O \\ \hline O & I_{m-n} \end{array}\right) = \left(Q_{mn}\hat{\hat{U}} \,\middle|\, Q_{m,m-n}\right) = \left(\hat{U}_{mn} \,\middle|\, \hat{U}_{m,m-n}\right) \in \mathbb{R}^{m\times m},$$

donde $\hat{U}_{mn} \in \mathbb{R}^{m\times n}$ está formada por las n primeras columnas de $\hat{U}$ y $\hat{U}_{m,m-n} \in \mathbb{R}^{m\times(m-n)}$ está formada por las columnas restantes de $\hat{U}$,

$$B = \left(\frac{\hat{B}}{O}\right) \in \mathbb{R}^{m\times n}$$

y

$$\hat{V} = \hat{\hat{V}} \in \mathbb{R}^{n\times n}$$

se tiene $A = \hat{U}B\hat{V}^T$. No obstante también es cierto que $A = \hat{U}_{mn}\hat{B}\hat{V}^T$.

La ventaja de este procedimiento en el caso de matrices con $m \gg n$ es que los reflectores por la derecha son aplicados a la matriz pequeña $\hat{R}$ en lugar de a la matriz A, por lo que el costo operativo es menor. La desventaja es que los reflectores destruyen la forma triangular de $\hat{R}$ y la mayor parte de las multiplicaciones por la izquierda deben ser repetidas, pero en la matriz pequeña $\hat{R}$. Aun así, si $m/n > 5/3$ el costo añadido de las multiplicaciones realizadas a mayores por la izquierda es menor que el ahorro obtenido en las multiplicaciones por la derecha, ver [9] para un análisis más detallado.

2.3. El algoritmo QR implícito para matrices bidiagonales

Una vez hallada la matriz bidiagonal cuadrada $\hat{B}$, el problema de calcular la descomposición en valores singulares de A se reduce a calcular los valores singulares de la matriz $\hat{B}$. Si $\hat{B} = \widetilde{U}\hat{\Sigma}\widetilde{V}^T$ con $\widetilde{U}, \widetilde{V} \in \mathbb{R}^{n\times n}$ ortogonales y $\hat{\Sigma} \in \mathbb{R}^{n\times n}$ diagonal, es la descomposición en valores singulares de $\hat{B}$, entonces

$$\begin{aligned} A &= \hat{U}_{mn}\hat{B}\hat{V}^T = \hat{U}_{mn}\widetilde{U}\hat{\Sigma}\widetilde{V}^T\hat{V}^T \\ &= \left(\hat{U}_{mn} \,\middle|\, \hat{U}_{m,m-n}\right) \left(\begin{array}{c|c} \widetilde{U} & O \\ \hline O & I_{m-n} \end{array}\right) \left(\frac{\hat{\Sigma}}{O}\right) \widetilde{V}^T\hat{V}^T \\ &= \left(\hat{U}_{mn}\widetilde{U} \,\middle|\, \hat{U}_{m,m-n}\right) \left(\frac{\hat{\Sigma}}{O}\right) \left(\hat{V}\widetilde{V}\right)^T, \end{aligned}$$

que es la descomposición en valores singulares de A, tal como se definió en el Teorema 2, aunque desde el punto de vista práctico es suficiente considerar

$$A = \left(\hat{U}_{mn}\widetilde{U}\right)\hat{\Sigma}\left(\hat{V}\widetilde{V}\right)^T.$$

Por conveniencia en la notación, a partir de ahora en lugar de $\hat{B}$ utilizaremos $B \in \mathbb{R}^{n\times n}$ para referirnos a la matriz bidiagonal cuadrada

$$B = \begin{pmatrix} \beta_1 & \gamma_1 & & & \\ & \beta_2 & \gamma_2 & O & \\ & & \ddots & \ddots & \\ & O & & \beta_{n-1} & \gamma_{n-1} \\ & & & & \beta_n \end{pmatrix}.$$

Diremos que B es una matriz propiamente bidiagonal si $\beta_i \neq 0$ y $\gamma_i \neq 0$ para todo i.

Si B no es una matriz propiamente bidiagonal, se puede reducir el problema de encontrar la descomposición en valores singulares de B a dos subproblemas de dimensión menor. En [10] se pueden encontrar los detalles.

Asumimos entonces, sin pérdida de generalidad, que B es propiamente bidiagonal, y pasamos a describir el algoritmo QR implícito para encontrar la descomposición en valores singulares de B.

Si $B \in \mathbb{R}^{n\times n}$ es una matriz propiamente bidiagonal, entonces BB^T y B^TB son matrices propiamente tridiagonales, y se pueden calcular sus autovalores mediante la iteración QR con desplazamiento. El algoritmo que vamos a desarrollar es equivalente al algoritmo QR, aplicado tanto a BB^T como a B^TB, pero sin la construcción explícita de las matrices producto.

Comenzamos el primer paso del algoritmo QR implícito eligiendo el desplazamiento adecuado. La submatriz inferior derecha de dimensión 2×2 de BB^T es

$$\begin{pmatrix} \beta_{n-1}^2 + \gamma_{n-1}^2 & \beta_n\gamma_{n-1} \\ \beta_n\gamma_{n-1} & \beta_n^2 \end{pmatrix}. \tag{17}$$

Se calculan los autovalores de esta submatriz y se toma como desplazamiento σ el autovalor de (17) más cercano a β_n^2, (desplazamiento de Wilkinson para BB^T). Podríamos escoger σ también a partir de B^TB pero la forma de BB^T es algo más simple.

Una iteración del algoritmo QR con desplazamiento σ aplicado a B^TB comienza hallando la factorización QR

$$B^TB - \sigma I = QR. \tag{18}$$

Para realizar una iteración implícita, necesitamos la primera columna de Q. Por ser la matriz R triangular superior, la primera columna de Q es proporcional a la primera columna de $B^TB - \sigma I$ que viene dada por

$$\begin{pmatrix} \beta_1^2 - \sigma \\ \gamma_1\beta_1 \\ 0 \\ \vdots \\ 0 \end{pmatrix}. \tag{19}$$

Tomamos entonces V_{12} una rotación de Givens en el plano $(1,2)$, cuya primera columna sea proporcional a (19). Multiplicando B por V_{12} por la derecha, modifica solo las dos primeras columnas de B y se crea una entrada no nula en la posición $(2,1)$.

Ahora buscamos una rotación U_{12}^T, en las filas $(1,2)$ tal que $U_{12}^TBV_{12}$ tenga un cero en la posición $(2,1)$. Esta operación actúa en las filas 1 y 2 y crea un nuevo valor no nulo en la posición $(1,3)$. Tomamos ahora la rotación V_{23} que actúa sobre las columnas 2 y 3 de modo que $U_{12}^TBV_{12}V_{23}$ tenga un cero en la posición $(1,3)$, pero aparece un valor no nulo en la posición $(3,2)$.

Continuando de este modo, aplicando una rotación $U_{i,i+1}^T$ por la izquierda que anula el elemento $(i+1,i)$ y genera un nuevo elemento no nulo en la posición $(i,i+2)$ seguida de una matriz $V_{i+1,i+2}$ por la derecha que anula el elemento $(i,i+2)$ y genera un elemento no nulo en la posición $(i+2,i+1)$, para $i = 2,\ldots,n-2$, se consigue que tras aplicar una última rotación $U_{n-1,n}^T$ por la izquierda se anule el elemento $(n,n-1)$ y se obtenga una matriz bidiagonal

$$\hat{B} = U_{n-1,n}^T \cdots U_{23}^T U_{12}^T B V_{12} V_{23} \cdots V_{n-1,n}. \tag{20}$$

Tomando $U = U_{12}U_{23}\cdots U_{n-1,n}$ y $V = V_{12}V_{23}\cdots V_{n-1,n}$ podemos escribir (20) como

$$\hat{B} = U^TBV. \tag{21}$$

Con esto finaliza una iteración del algoritmo QR implícito. Además tenemos $\hat{B}\hat{B}^T = U^TBB^TU$ y $\hat{B}^T\hat{B} = V^TB^TBV$, por lo que $\hat{B}\hat{B}^T$ y $\hat{B}^T\hat{B}$ son esencialmente las mismas matrices que habríamos obtenido si hubiésemos dado una iteración del algoritmo QR con desplazamiento σ partiendo de las matrices BB^T y B^TB para aproximar sus autovalores. En esta afirmación juega un papel crucial el hecho de que la matriz V y la matriz Q de (18) tengan la primera columna igual salvo posiblemente el signo. Un análisis detallado puede verse en [10].

Si tomamos como matriz B la matriz $\hat{B}$ y repetimos la iteración QR implícita, las matrices BB^T y B^TB tenderán a una forma diagonal y las entradas de la diagonal principal convergerán a los autovalores. La utilización del desplazamiento de Wilkinson hace que las entradas $(n, n-1)$ y (n, n) de BB^T y B^TB converjan la primera a 0, y la segunda a un autovalor. Por supuesto no se trabaja con BB^T ni con B^TB sino con B. La rápida convergencia de BB^T y B^TB hacia una matriz diagonal se traduce en la convergencia de γ_{n-1} a 0 y de β_n a un valor singular de B.

Una vez que γ_{n-1} sea menor que una tolerancia fijada puede considerarse como 0 y reducir el problema a uno de dimensión $(n-1)\times(n-1)$ ignorando la última fila y última columna de la matriz B. Reiterando el proceso se encuentran todos los valores singulares de B.

En el Apéndice se puede encontrar una función en MATLAB que implementa la iteración QR implícita junto con una visualización gráfica de esta implementación.

2.4. Cálculo de los vectores singulares

Si solo se necesita calcular los valores singulares, no es necesario el almacenamiento de los reflectores usados en la reducción de la matriz a forma bidiagonal, ni de las rotaciones empleadas en cada iteración del algoritmo QR implícito.

Si, por el contrario, necesitamos los vectores singulares, se irán aplicando sobre una matriz identidad I_m todas las transformaciones ortogonales que se le hagan a la matriz A por la izquierda y se aplicarán sobre una matriz identidad I_n todas las transformaciones ortogonales que se le hagan a la matriz A por la derecha. En ningún caso se construirán cada una de las matrices ortogonales que intervienen en el proceso. A pesar de evitar dicha construcción, el costo operativo que para obtener los valores singulares crece linealmente con m, pasa a crecer cuadráticamente con el número de filas de la matriz A si es necesario calcular los vectores singulares, algo que es bien conocido en el problema más simple de calcular la factorización QR de una matriz.

3. Algunas aplicaciones de la descomposición en valores singulares

3.1. Compresión y transmisión de imágenes

Una aplicación práctica de la descomposición en valores singulares de una matriz es la transmisión y compresión de imágenes [2]. El fundamento teórico de esta aplicación es el resultado de aproximación en la norma de Frobenius recogido en el Teorema 8 de esta memoria. Dicho teorema permite aproximar cada coeficiente de una matriz por el correspondiente coeficiente de la matriz que se obtiene al considerar su aproximación óptima por matrices de un rango determinado.

Dada una matriz A de datos de tamaño $m \times n$, la transmisión de dicha matriz completa supone la transmisión de mn valores reales. Si consideramos la descomposición en valores singulares de $A = \sum_{i=1}^{r} \sigma_i u_i v_i^T$, el Teorema 8 permite de igual forma escribir la matriz de rango k que mejor aproxima a A, como $A_k = \sum_{i=1}^{k} \sigma_i u_i v_i^T$. Incrementar en una unidad el rango de la matriz con la que aproximamos a la matriz original supone la transmisión de $(m+n)$ números reales adicionales, puesto que se puede multiplicar previamente cada vector singular por la derecha por el correspondiente valor singular. Si en vez de transmitir A, transmitimos la información necesaria para generar la aproximación A_k, es suficiente transmitir $k(m+n)$ datos, en lugar de los mn que forman la matriz completa.

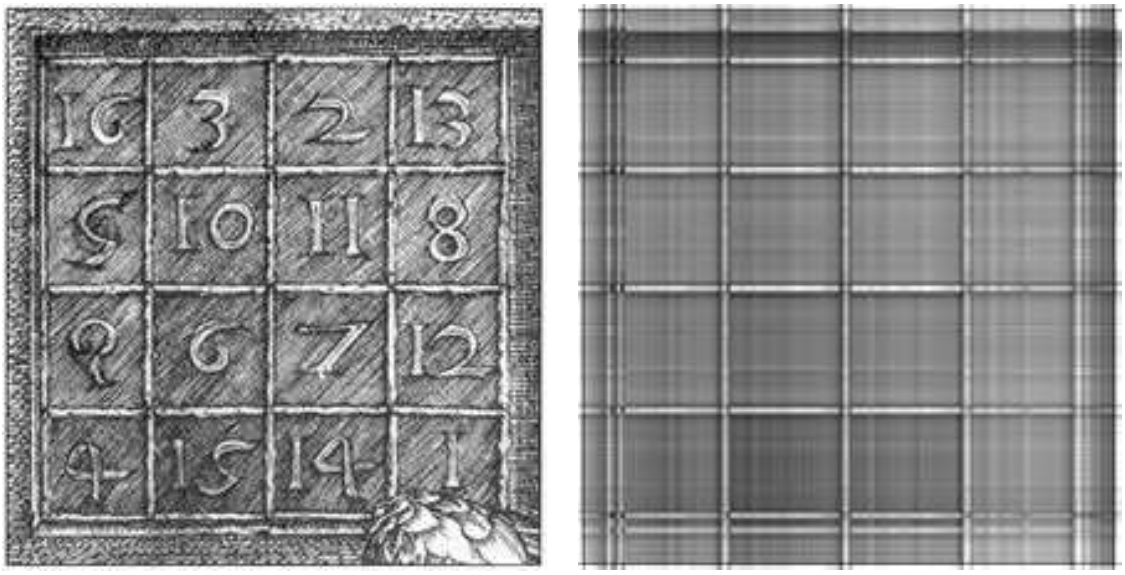


Figura 3: La figura de la izquierda muestra la imagen original con un total de 50380 datos. La figura de la derecha utiliza 898 datos, el 2 % del total.

El procedimiento tendrá interés sólo si se puede tomar $k \ll \dfrac{mn}{m+n}$. En caso contrario, además de tener que transmitir más elementos que el número de píxeles de la imagen original habríamos tenido que calcular la descomposición en valores singulares de la matriz que, como hemos visto, requiere del orden de m^2n operaciones, al ser necesarios también los vectores singulares.

El teorema de aproximación permite disponer de una primera imagen no exacta sin necesidad de disponer de los valores exactos de todos los píxeles.

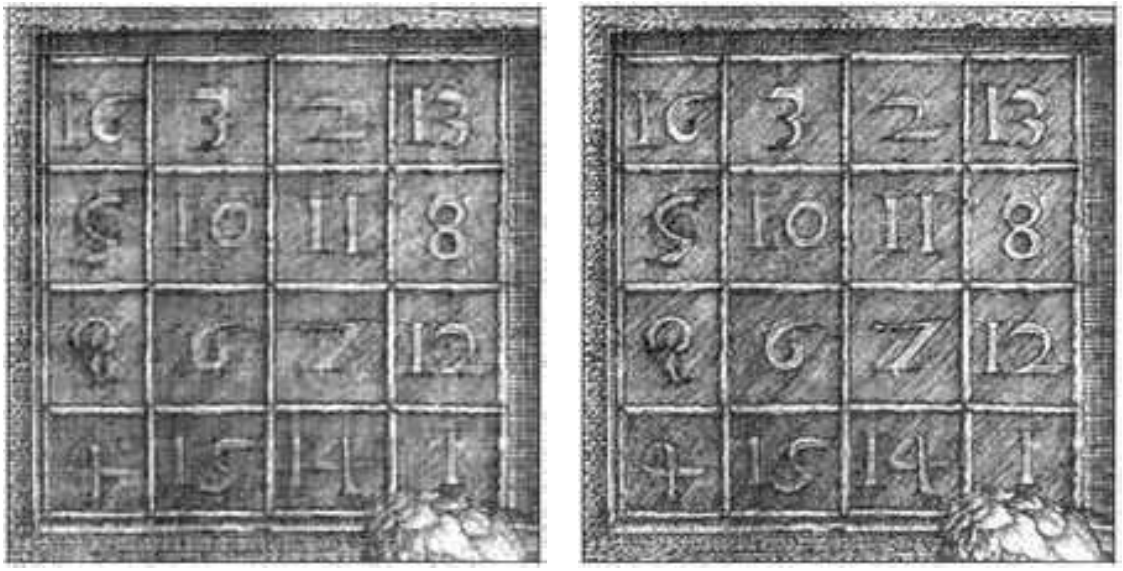


Figura 4: La figura de la izquierda muestra la imagen con 12572 datos que representa el 25 % del total. La figura de la derecha utiliza el 50 % del total de los datos, 25144.

Para mostrar un ejemplo concreto se ha tomado la imagen del cuadrado mágico del grabado Melencolia I de Alberto Durero [11], representada en la parte izquierda de la Figura 3. La matriz de datos asociada tiene dimensión 220×229, por lo que el número total de coeficientes de la matriz es algo mayor que 50000. Además dicha matriz tiene rango 220. Si se realiza su descomposición en valores singulares, cada par de vectores singulares supone el envío de 449 datos.

En el experimento se muestra en primer lugar la imagen completa al procesar la matriz A original con las funciones de MATLAB `rgb2gray` e `imshow`, y en segundo lugar se muestran las imágenes que se van obteniendo al procesar, con el mismo comando `imshow`, las aproximaciones óptimas A_k de rango k, para distintos valores de k. En las imágenes así obtenidas podemos observar que con una matriz de aproximación de rango $k = 1$ la cuadrícula del cuadrado mágico queda totalmente definida (imagen derecha de la Figura 3). Con una aproximación de rango $k = 28$, lo que corresponde a utilizar solo el 25 % de los datos, podemos intuir todos los números (imagen izquierda de la Figura 4), obteniéndose una apreciación casi perfecta de los elementos de la imagen utilizando solo la mitad de los

datos, lo que corresponde a aproximar por una matriz de rango $k = 56$ (imagen derecha de la Figura 4).

3.2. Un ejemplo de Climatología

Otro de los campos de estudio donde se utiliza la descomposición en valores singulares de una matriz es en el análisis climático. En este caso se calcula la descomposición en valores singulares de la matriz de covarianzas cruzadas entre los datos correspondientes a dos variables climatológicas espacio-temporales: el predictor y el predictando, que se conocen en los mismos instantes de tiempo [1].

Más precisamente, se parte de las matrices S, con los datos del predictor, y P con los datos del predictando de modo que cada fila contenga los datos correspondientes a un instante de tiempo y a las distintas localizaciones espaciales en las que se han tomado medidas. Asumiendo que los datos de ambas corresponden a medidas realizadas en los mismos tiempos se forma la matriz de covarianzas

$$C = S^T P,$$

donde las matrices S y P han sido previamente modificadas restando a cada columna su media.

Una vez construída C calculamos su descomposición en valores singulares obteniendo

$$C = U\Sigma V^T,$$

donde las columnas de U corresponden a la variable S y las columnas de V corresponden a la variable P. Cada pareja de vectores singulares es un modo de covariabilidad entre el predictor S y el predictando P y la primera pareja de vectores singulares da cuenta de la máxima cantidad de covarianza al cuadrado entre dichas variables.

Proyectando el campo original sobre el vector singular, se obtiene la serie temporal de coeficientes de expansión. La correlación existente entre las series temporales de coeficientes de expansión de cada variable mide la intensidad de la relación entre ellas.

Para conocer los coeficientes de expansión, es decir, las series temporales que describen como varía cada modo en el tiempo se contruyen las matrices

$$A = SU \quad \text{para } S$$

y

$$B = PV \quad \text{para } P.$$

Las columnas de las matrices A y B contienen las series temporales de cada modo y podemos reconstruir S y P haciendo $S = AU^T$ y $P = BV^T$. La matriz diagonal Σ contiene los valores singulares de C y la covarianza cuadrada total en C viene dada por la suma de los cuadrados de los valores de la diagonal de Σ. Los valores singulares proporcionan la fracción de covarianzas explicada por cada modo singular. Si σ_i es el i-ésimo valor

singular, la fracción de covarianzas correspondiente al par de vectores singulares u_i y v_i viene dada por

$$SCF_i = \frac{\sigma_i^2}{\sum \sigma_i^2}$$

Este valor indica la cantidad de información que representa sobre el total cada modo de covariabilidad. Los coeficientes de las series muestran la variación del mapa en el tiempo. Es decir, si estamos interesados en la información del predictor dado por u_i y del predictando dado por v_i, los coeficientes de expansión sobre la variabilidad vienen dados por a_i (columa i-ésima de A) y b_i (columna i-ésima de B).

En nuestro ejemplo particular los datos del predictor son valores medios mensuales del año 1967 de altura geopotencial en el nivel de 500 hPa medidos en una malla de $2.5^o \times 2.5^o$ (longitud $\times$ latitud) que comprende un dominio espacial que abarca el Atlántico Norte, el mar Mediterráneo y Europa, desde 20^o N a 85^o N de latitud, y desde 105^o W a 55^o E de longitud [6]. Esta región contiene 1755 nodos. Los datos en cada punto corresponden a la altura sobre el nivel del mar en la cual la presión es de 500 hPa. Esta altura, no se corresponde exactamente con la distancia vertical desde el mar hasta el punto donde la presión es 500 hPa pues se tiene en cuenta la acción de la gravedad. El geopotencial de 500 hPa en un punto del mapa es el trabajo necesario que habría que realizar para elevar la unidad de masa desde el nivel del mar hasta el nivel en el que la presión es de 500 hPa. Esta definición permite adoptar la altitud como algo independiente de la aceleración de la gravedad.

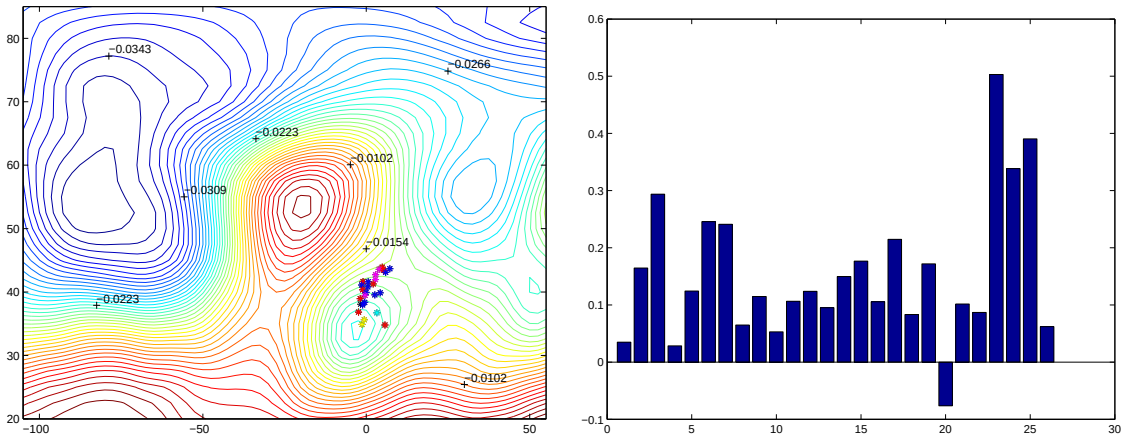


Figura 5: Representación de la altura geopotencial y las precipitaciones correspondientes al primer modo de covariablidad.

La variable regional predictando consiste en 26 series de precipitaciones mensuales acumuladas correspondientes a observatorios del sur de Francia, costa Mediterránea Peninsular y norte de África.

Tras restar las medias temporales en cada punto construímos la matriz de covarianzas $C = S^T P$ con $C \in \mathbb{R}^{1755\times 26}$ y calculamos su descomposición en valores singulares $C =$

$U\Sigma V^T$ donde $U \in \mathbb{R}^{1755\times1755}$, $\Sigma \in \mathbb{R}^{1755\times26}$ y $V \in \mathbb{R}^{26\times26}$.

Implementando el cálculo del vector formado por los SCF_i, vemos que en nuestro ejemplo el $91,9\,\%$ de la información queda explicada por el modo 1 mientras que el modo 2 aporta solo el $5,3\,\%$ de la información y todos los restantes modos menos del $3\,\%$.

De esta forma $C_1 = \sigma_1 u_1 v_1^T$ representa la mayor parte de la covarianza de los datos y con $C_2 = \sum_{i=1}^{2} \sigma_i u_i v_i^T$ se obtiene prácticamente toda la información.

Se han representado los mapas correspondientes a la información dada por u_1 y v_1 y el correspondiente peso de las series temporales de S y P.

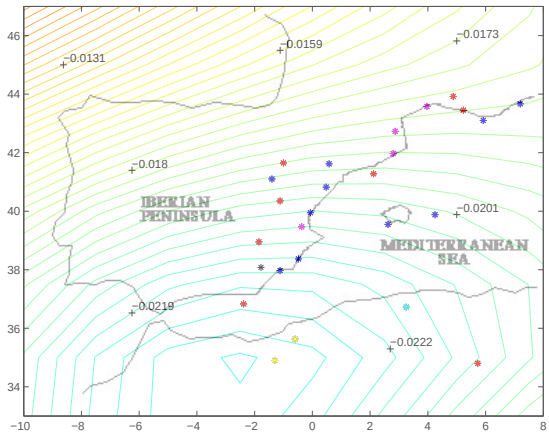


Figura 6: Detalle de las curvas de nivel de la altura geopotencial en la península ibérica.

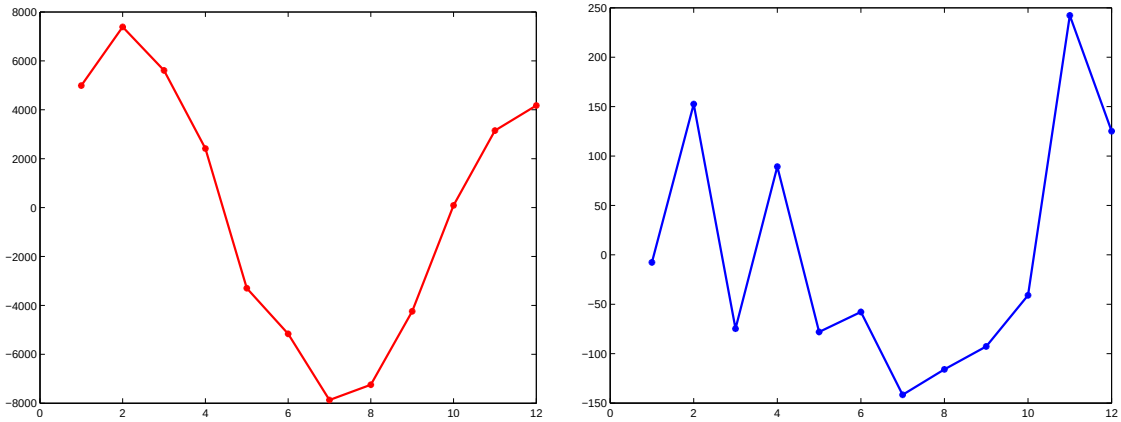


Figura 7: Representación de los coeficientes de expansión asociados a los vectores singulares u_1 y v_1.

En la gráfica izquierda de la Figura 5 se muestran las curvas de nivel para las alturas geopotenciales: las isolíneas unen puntos en los que la presión de 500 hPa se encuentra a la misma altitud. Estas líneas se denominan "isohipsas" que significa igual altura. Los

valores que acompañan a las líneas de nivel indican lo que se desvían de la altura geopotencial media las alturas geopotenciales en esos puntos. Los asteriscos que aparecen en la imagen representan la posición de las distintas estaciones meteorológicas en las que se han medido las precipitaciones, y están agrupados en distintos colores en función del nivel de precipitación marcado por cada estación.

En la Figura 6 se muestra una ampliación de esta gráfica en la zona peninsular.

En la gráfica de la derecha de la Figura 5 se han representado las anomalías de las precipitaciones en cada estación meteorológica, es decir, lo que el nivel de precipitaciones en cada estación se ha desviado de la media, estos datos corresponden al primer modo de covariabilidad, v_1.

Las gráficas de la Figura 7 muestran los coeficientes de expansión que describen la variación en el tiempo de los campos S (izquierda) y P (derecha) para el primer modo de covariabilidad. La correlación entre ellos es 0.71, lo que indica un alto grado de relación causa-efecto entre el predictor y el predictando.

Referencias

[1] H. Björnsson y S.A. Venegas, *A Manual for EOF and SVD Analyses of Climatic Data*, C^2GCR Report No. 97-1, 1997.

[2] L. Colombo y R. Lafuente, *El uso de la descomposición en valores singulares para el tratamiento de imágenes*, Matematicalia 7 (2011), pp. 1-14.

[3] C. Eckart y G. Young, *The aproximation of one matrix by another of lower rank*, Psycometrika 1 (1936), pp. 211–218.

[4] G.H. Golub y W. Kahan, *Calculating the singular values and pseudo-inverse of a matrix*, SIAM J. Numer. Anal. 2 (1965), pp. 205–224.

[5] G.H. Golub y C. Reinsch, *Singular Value Decomposition and Least Squares Solutions*, Numer. Math. 14 (1970), pp. 403–420.

[6] M.Y. Luna, A. Morata, M.L. Martín y F. Valero, *Influencia de los patrones de teleconexión del Atlántico Norte en la precipitación primaveral del Mediterráneo occidental*, Física de la Tierra 12 (2004), pp. 137–148.

[7] G.W. Stewart, *On the early history of the singular value descomposition*, SIAM Review 35 (1993), pp. 551–566.

[8] G. Strang, *The Fundamental Theorem of Linear Algebra*, The American Mathematical Monthly 100 (1993), pp. 845–855.

[9] Lloyd N. Trefethen y David Bau, *Numerical Linear Algebra*, SIAM, Philadelphia (1997).

[10] David S. Watkins, *Fundamentals of Matrix Computations*, John Wiley and Sons, Singapur (1991).

[11] I. Zaballa, *Valores singulares.¿Qué son?.¿Para qué sirven?.*, Departamento de Matemática Aplicada y EIO, Universidad del País Vasco.

Apéndice

Como ya se indicó en la Sección 2, en este apéndice se incluyen los programas en MATLAB que implementan los algoritmos allí descritos para calcular de un modo eficiente la descomposición en valores singulares de una matriz. Se incluye también un función que permite visualizar gráficamente la actuación de las distintas transformaciones ortogonales que se aplican durante este proceso.

A. Implementación del proceso de bidiagonalización de una matriz

```
function [A, U, V] = bidiagonal(A);
% [A, U, V] = bidiagonal(A) genera una matriz bidiagonal (B) del mismo
% tamaño que A y que se sobreescribe en A y dos matrices ortogonales
% U y V tales que U*A*V=B.
% Para ello se aplican reflectores de Householder por la derecha y
% por la izquierda sobre la matriz A y sobre matrices identidad de
% dimensiones adecuadas.
% El número de filas de A tiene que ser mayor o igual que el número
% de columnas.

[m,n] = size(A);

if m<n % Si m < n la función deberá ejecutarse con A'
  disp('El número de filas debe ser mayor o igual que el número de columnas');
else

  U=eye(m); % Se inicializa la matriz ortogonal izquierda
  V=eye(n); % Se inicializa la matriz ortogonal derecha

  for k=1:n-2,
    % Se aplica el reflector de Householder U_k por la izquierda a
    % las últimas m-k+1 filas y n-k+1 columnas de la matriz A y a
    % las últimas m-k+1 filas de la matriz que procede de la matriz
    % identidad de orden m
    w(k:m,1) = house(A(k:m,k));
    beta = 2/(w(k:m,1)'*w(k:m,1));
    A(k:m,k:n) = A(k:m,k:n)-w(k:m,1)*(beta*(A(k:m,k:n)'*w(k:m,1)))';
    U(k:m,:) = U(k:m,:)-w(k:m,1)*(beta*(U(k:m,:)'*w(k:m,1)))';

    % Se aplica el reflector de Householder V_k por la derecha a
    % las últimas m-k filas y n-k columnas de la matriz A y a las
    % últimas n-k columnas de la matriz que procede de la matriz
    % identidad de orden n
    z(k+1:n,1) = house(A(k,k+1:n)');
    gamma = 2/(z(k+1:n,1)'*z(k+1:n,1));
    A(k:m,k+1:n) = A(k:m,k+1:n)-gamma*(A(k:m,k+1:n)*z(k+1:n,1))*z(k+1:n,1)';
    V(:,k+1:n) = V(:,k+1:n)-gamma*(V(:,k+1:n)*z(k+1:n,1))*z(k+1:n,1)';
```

```
  end
  % Se aplica el reflector de Householder U_{n-1} por la izquierda
  k=n-1;
  w(k:m,1) = house(A(k:m,k));
  beta = 2/(w(k:m,1)'*w(k:m,1));
  A(k:m,k:n) = A(k:m,k:n)-w(k:m,1)*(beta*(A(k:m,k:n)'*w(k:m,1)))';
  U(k:m,:) = U(k:m,:)-w(k:m,1)*(beta*(U(k:m,:)'*w(k:m,1)))';
  if m>n % Se aplica el reflector de Householder U_n por la izquierda
    w(n:m,1) = house(A(n:m,n));
    beta = 2/(w(n:m,1)'*w(n:m,1));
    A(n:m,n) = A(n:m,n)-w(n:m,1)*(beta*(A(n:m,n)'*w(n:m,1)))';
    U(n:m,:) = U(n:m,:)-w(n:m,1)*(beta*(U(n:m,:)'*w(n:m,1)))';
  elseif A(n,n)<0, %Se cambia de signo el elemento (n,n) si es negativo
    A(n,n) = -A(n,n);
    U(n,:) = -U(n,:);
  end
end
end
```

B. Cálculo de la descomposición en valores singulares de una matriz bidiagonal

Los argumentos de entrada de esta función deben ser los argumentos de salida de la función `bidiagonal`. Si la matriz de partida ya fuese bidiagonal, se ejecuta directamente esta función tomando la matriz original como `B`, y `U` y `V` iguales a las matrices identidad de órdenes adecuados, sin tener que ejecutar la función `bidiagonal`.

```
function [B, U, V] = diagonal(B,U,V);
% [B, U, V] = diagonal(B,U,V) genera una matriz diagonal del mismo
% tamaño que B y que se sobreescribe en B y dos matrices ortogonales
% U y V tales U*A*V=B (A es el argumento de entrada de bidiagonal).
% Para ello se aplican rotaciones de Givens por la derecha y por la
% izquierda sobre la matriz B y sobre las matrices U y V.
% La matriz B tiene que ser bidiagonal superior y el numero de filas
% de B tiene que ser mayor o igual que el número de columnas.

tol=1.0e-4; % Tolerancia utilizada para parar la iteracion
[m,n]=size(B);

if m<n % Si m> n la función deberá ejecutarse con A'
  disp('El número de filas debe ser mayor o igual que el número de columnas');
else

  fin=n;
  B=B(1:n,:); % Se eliminan las filas nulas de B
```

```
while (norm(diag(B,1))>tol) % Se comprueba el criterio de parada
  % Se calcula el desplazamiento: el autovalor de C más cercano a
  % B(fin,fin)^2 es sigma(p)
  C=[B(fin-1,fin-1)^2+B(fin-1,fin)^2 B(fin,fin)*B(fin-1,fin);
  B(fin,fin)*B(fin-1,fin) B(fin,fin)^2];
  sigma=eig(C);
  [diferencia,p]=min(abs(C(2,2)-sigma));

  % Se aplica la rotación de Givens V_{1,2} por la derecha a B y a V
  denominador=sqrt((B(1,1)^2-sigma(p))^2+B(1,2)^2*B(1,1)^2);
  c=(B(1,1)^2-sigma(p))/denominador;
  s=-B(1,2)*B(1,1)/denominador;
  tau1=B(1:2,1);
  tau2=B(1:2,2);
  B(1:2,1)=c*tau1-s*tau2;
  B(1:2,2)=s*tau1+c*tau2;
  gamma1=V(:,1);
  gamma2=V(:,2);
  V(:,1)=c*gamma1-s*gamma2;
  V(:,2)=s*gamma1+c*gamma2;

  for i=1:fin-2
    % Se aplica la rotación de Givens U_{i,i+1} por la izquierda
    % a B y a U
    [c,s]=givens(B(i,i),B(i+1,i));
    tau1=B(i,i:i+2);
    tau2=B(i+1,i:i+2);
    B(i,i:i+2)=c*tau1-s*tau2;
    B(i+1,i:i+2)=s*tau1+c*tau2;
    gamma1=U(i,:);
    gamma2=U(i+1,:);
    U(i,:)=c*gamma1-s*gamma2;
    U(i+1,:)=s*gamma1+c*gamma2;

    % Se aplica la rotación de Givens V_{i+1,i+2} por la derecha
    % a B y a V
    [c,s]=givens(B(i,i+1),B(i,i+2));
    tau1=B(i:i+2,i+1);
    tau2=B(i:i+2,i+2);
    B(i:i+2,i+1)=c*tau1-s*tau2;
    B(i:i+2,i+2)=s*tau1+c*tau2;
    gamma1=V(:,i+1);
    gamma2=V(:,i+2);
    V(:,i+1)=c*gamma1-s*gamma2;
    V(:,i+2)=s*gamma1+c*gamma2;
```

```
    end
    % Se aplica la última rotación de Givens por la izquierda U_{n-1,n}
    [c,s]=givens(B(fin-1,fin-1),B(fin,fin-1));
    tau1=B(fin-1,fin-1:fin);
    tau2=B(fin,fin-1:fin);
    B(fin-1,fin-1:fin)=c*tau1-s*tau2;
    B(fin,fin-1:fin)=s*tau1+c*tau2;
    gamma1=U(fin-1,:);
    gamma2=U(fin,:);
    U(fin-1,:)=c*gamma1-s*gamma2;
    U(fin,:)=s*gamma1+c*gamma2;
    % Se reduce el tamaño de la matriz sobre la que se aplican las
    % rotaciones si B(fin-1,fin) es menor que la tolerancia dada
    if fin>2 && abs(B(fin-1,fin))<tol
      fin=fin-1;
    end
  end
  % Se ordenan los elementos diagonales de B y las columnas de U y V
  [x,I]=sort(diag(B),1,'descend');
  U(1:n,:)=U(I,:);
  V=V(:,I);
  B=[diag(x); zeros(m-n,n)];
end
end
```

C. Visualizacíón gráfica de la actuación del algoritmo de descomposición en valores singulares

Representamos gráficamente utilizando la función `mesh` de MATLAB la acción de los reflectores de Householder y las rotaciones de Givens sobre una matriz.

```
function [A, U, V]= dibujarSVD(A);
% [A, U, V] = dibujarmatriz(A) representa gráficamente el cálculo numérico
% eficiente de la descomposición en valores singulares mostrando la acción
% de los reflectores de Householder y de las rotaciones de Givens sobre la
% matriz A mediante la función representa. Al mismo tiempo se genera una
% matriz diagonal (D) del mismo tamaño que A que se sobreescribe en A y dos
% matrices ortogonales U y V tales que U*A*V=D.
% El numero de filas de A tiene que ser mayor o igual que el número de
% columnas.
[m,n] = size(A);
rellena(A);
if m<n % Si m> n la función deberá ejecutarse con A'
  disp('El número de filas debe ser mayor o igual que el número de columnas');
```

```
else
  U=eye(m); % Se inicializa la traspuesta de la matriz ortogonal izquierda
  V=eye(n); % Se inicializa la matriz ortogonal derecha
  for k=1:n-2,
    % Se aplica el reflector de Householder U_k por la izquierda a las
    % últimas m-k+1 filas y n-k+1 columnas de la matriz A y a las últimas
    % m-k+1 filas de la matriz que procede de la identidad de orden m
    w(k:m,1) = house(A(k:m,k));
    beta = 2/(w(k:m,1)'*w(k:m,1));
    A(k:m,k:n) = A(k:m,k:n)-w(k:m,1)*(beta*(A(k:m,k:n)'*w(k:m,1)))';
    U(k:m,:) = U(k:m,:)-w(k:m,1)*(beta*(U(k:m,:)'*w(k:m,1)))';
    representa(A);
    % Se aplica el reflector de Householder V_k por la derecha a las
    % últimas m-k filas y n-k columnas de la matriz A y a las últimas
    % n-k columnas de la matriz que procede de la identidad de orden n
    z(k+1:n,1) = house(A(k,k+1:n)');
    gamma = 2/(z(k+1:n,1)'*z(k+1:n,1));
    A(k:m,k+1:n) = A(k:m,k+1:n)-gamma*(A(k:m,k+1:n)*z(k+1:n,1))*z(k+1:n,1)';
    V(:,k+1:n) = V(:,k+1:n)-gamma*(V(:,k+1:n)*z(k+1:n,1))*z(k+1:n,1)';
    representa(A);
  end
  % Se aplica el reflector de Householder U_n-1 por la izquierda
  k=n-1;
  w(k:m,1) = house(A(k:m,k));
  beta = 2/(w(k:m,1)'*w(k:m,1));
  A(k:m,k:n) = A(k:m,k:n)-w(k:m,1)*(beta*(A(k:m,k:n)'*w(k:m,1)))';
  U(k:m,:) = U(k:m,:)-w(k:m,1)*(beta*(U(k:m,:)'*w(k:m,1)))';
  representa(A);
  if m>n % Se aplica el reflector de Householder U_n por la izquierda
    w(n:m,1) = house(A(n:m,n));
    beta = 2/(w(n:m,1)'*w(n:m,1));
    A(n:m,n) = A(n:m,n)-w(n:m,1)*(beta*(A(n:m,n)'*w(n:m,1)))';
    U(n:m,:) = U(n:m,:)-w(n:m,1)*(beta*(U(n:m,:)'*w(n:m,1)))';
    representa(A);
  elseif A(n,n)<0,
    A(n,n) = -A(n,n);
    U(n,:) = -U(n,:);
    representa(A);
  end
end
ceros=zeros(m-n,n);
tol=1.0e-4; % Tolerancia utilizada para parar la iteracion
fin=n;
```

```
A=A(1:n,:); % Se eliminan las filas nulas de B
while (norm(diag(A,1))>tol) % Se comprueba el criterio de parada
  % Se calcula el desplazamiento: el autovalor de C más cercano a
  % B(fin,fin)^2 es sigma(p)
  C=[A(fin-1,fin-1)^2+A(fin-1,fin)^2 A(fin,fin)*A(fin-1,fin);
  A(fin,fin)*A(fin-1,fin) A(fin,fin)^2];
  sigma=eig(C);
  [diferencia,p]=min(abs(C(2,2)-sigma));
  % Se aplica la rotación de Givens V_1,2 por la derecha a B y a V
  denominador=sqrt((A(1,1)^2-sigma(p))^2+A(1,2)^2*A(1,1)^2);
  c=(A(1,1)^2-sigma(p))/denominador;
  s=-A(1,2)*A(1,1)/denominador;
  tau1=A(1:2,1);
  tau2=A(1:2,2);
  A(1:2,1)=c*tau1-s*tau2;
  A(1:2,2)=s*tau1+c*tau2;
  AA=[A;ceros];
  representa(AA);
  gamma1=V(:,1);
  gamma2=V(:,2);
  V(:,1)=c*gamma1-s*gamma2;
  V(:,2)=s*gamma1+c*gamma2;
  for i=1:fin-2
    % Se aplica la rotación de Givens U_i,i+1 por la izquierda a B y a U
    [c,s]=givens(A(i,i),A(i+1,i));
    tau1=A(i,i:i+2);
    tau2=A(i+1,i:i+2);
    A(i,i:i+2)=c*tau1-s*tau2;
    A(i+1,i:i+2)=s*tau1+c*tau2;
    AA=[A;ceros];
    representa(AA);
    gamma1=U(i,:);
    gamma2=U(i+1,:);
    U(i,:)=c*gamma1-s*gamma2;
    U(i+1,:)=s*gamma1+c*gamma2;
    % Se aplica la rotación de Givens V_i+1,i+2 por la derecha a B y a V
    [c,s]=givens(A(i,i+1),A(i,i+2));
    tau1=A(i:i+2,i+1);
    tau2=A(i:i+2,i+2);
    A(i:i+2,i+1)=c*tau1-s*tau2;
    A(i:i+2,i+2)=s*tau1+c*tau2;
    AA=[A;ceros];
    representa(AA);
```

```
    gamma1=V(:,i+1);
    gamma2=V(:,i+2);
    V(:,i+1)=c*gamma1-s*gamma2;
    V(:,i+2)=s*gamma1+c*gamma2;
  end
  % Se aplica la última iteracion de Givens por la izquierda U_n-1,n
  [c,s]=givens(A(fin-1,fin-1),A(fin,fin-1));
  tau1=A(fin-1,fin-1:fin);
  tau2=A(fin,fin-1:fin);
  A(fin-1,fin-1:fin)=c*tau1-s*tau2;
  A(fin,fin-1:fin)=s*tau1+c*tau2;
  AA=[A;ceros];
  representa(AA);
  gamma1=U(fin-1,:);
  gamma2=U(fin,:);
  U(fin-1,:)=c*gamma1-s*gamma2;
  U(fin,:)=s*gamma1+c*gamma2;
  % Se reduce el tamaño de la matriz sobre la que se aplican las
  % rotaciones una vez que B(fin-1,fin) es menor que la tolerancia dada
  if fin>2 && abs(A(fin-1,fin))<tol
    fin=fin-1;
  end
end
% Se ordenan los elementos diagonales de B y las columnas de U y V
[x,I]=sort(diag(A),1,'descend');
U(1:n,:)=U(I,:);
V=V(:,I);
A=[diag(x); zeros(m-n,n)];
representa(A);
end
```

Mostramos ahora la función en MATLAB `representa` que dibuja los valores de la matriz en 3D tomando estos como altura.

```
function []=representa(A);
[m,n]=size(A);
% Se intercalan filas y columnas de ceros entre cada fila y columna de A
% obteniendo así una matriz de dimensión (2m+1)x(2n+1). De esta forma se
% obtiene una mejor visualización con la función mesh de los valores de
% la matriz.
AA=zeros(2*m+1,n); AA(2:2:2*m,:)=A;
A=zeros(2*m+1,2*n+1); A(:,2:2:2*n)=AA;
% Se representan los valores de la matriz en 3D tomando como alturas los
% valores que tiene la matriz.
```

```
figure(1)
x=0:2*n; y=0:2*m;
mesh(x,y,abs(A(2*m+1:-1:1,2*n+1:-1:1)))
axis([0 2*n 0 2*m 0 5])
axis off
pause(1)
end
```

Incluímos ahora una ilustración gráfica de algunos pasos del proceso de bidiagonalización y posterior cálculo de la descomposición en valores singulares de la matriz bidiagonal. Para ello se ha utilizado la función `mesh` de MATLAB que permite representar el tamaño de los elementos de la matriz. Más precisamente, la altura de la pirámide con vértice en la posición (i, j) refleja el tamaño del correspondiente elemento $a_{i,j}$ de la matriz. Se ha trabajado con una matriz de tamaño 10×8 que se ha generado con el comando $\texttt{A} = \texttt{rand}(10, 8)$. Sus coeficientes, por tanto, están entre 0 y 1. La escala en todas las gráficas es la misma. En los ejes x e y se ha adaptado al tamaño de la matriz y en el eje z varía entre 0 y 5.

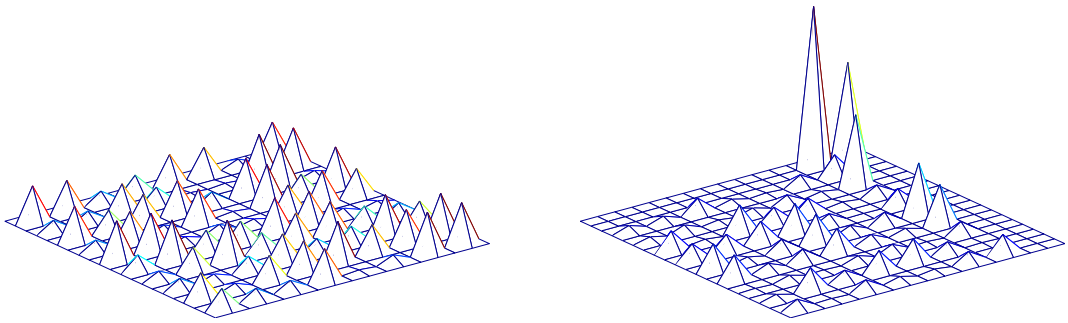


Figura 8: La figura de la izquierda muestra la matriz de partida. La figura de la derecha representa la matriz una vez que se han aplicado los reflectores de Householder $\hat{U}_1$ por la izquierda y $\hat{V}_1$ por la derecha (ver demostración del Teorema 12)

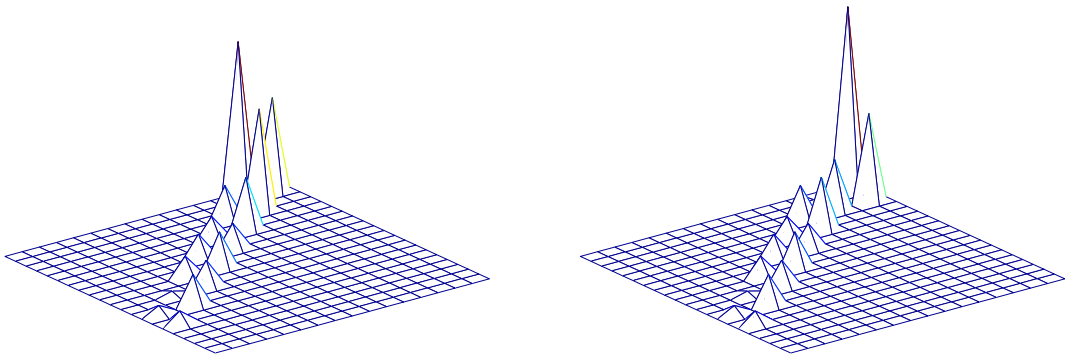


Figura 9: La figura de la izquierda muestra la matriz bidiagonal obtenida tras aplicar todos los reflectores de Householder. La figura de la derecha representa la matriz una vez que se ha aplicado la rotación de Givens V_{12} en el plano $(1, 2)$ por la derecha (ver Subsección 2.3).

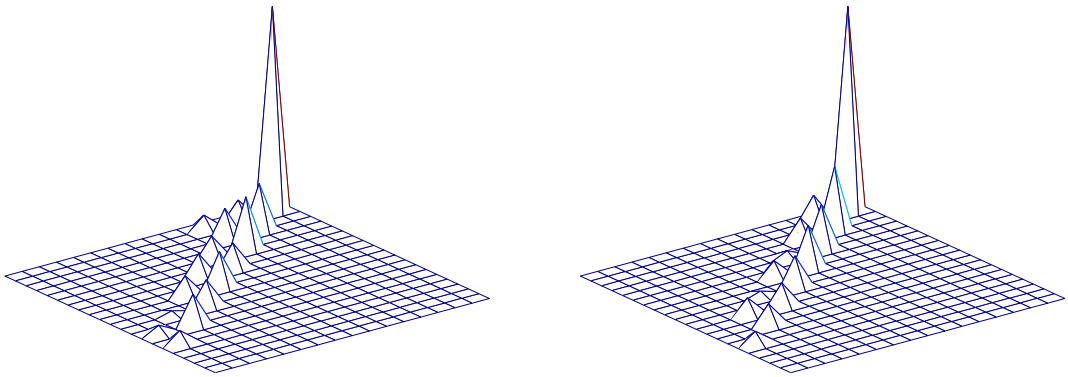


Figura 10: La figura de la izquierda representa la matriz una vez que se ha aplicado la rotación de Givens U_{12}^T en el plano $(1, 2)$ por la izquierda. La figura de la derecha muestra la matriz bidiagonal que se obtiene al terminar la primera iteración QR implícita. El último valor de la subdiagonal ya es prácticamente nulo.

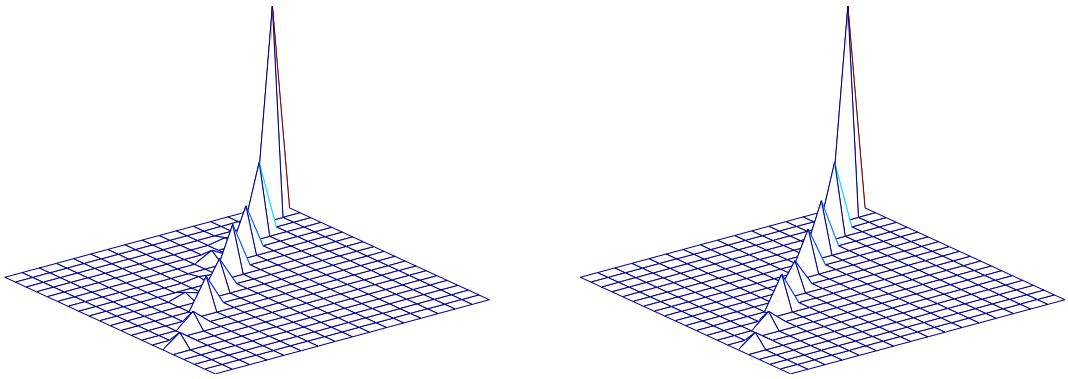


Figura 11: La figura de la izquierda muestra la matriz bidiagonal con varios elementos nulos en la subdiagonal. La figura de la derecha representa la matriz diagonal formada por los valores singulares de la matriz de partida.