

# **La Descomposición en Valores Singulares**



**Carmen Jiménez Segura**  
Trabajo de fin de grado de Matemáticas  
Universidad de Zaragoza

Noviembre 2024



# Prólogo

A lo largo de la historia de la teoría de matrices, uno de los objetivos principales ha sido buscar una descomposición matricial o una descomposición en forma canónica. A lo largo del siglo XVIII algunos matemáticos estudiaron una descomposición para matrices cuadradas basada en los valores propios. Sin embargo, hasta el siglo XIX no se encontró una descomposición matricial para matrices rectangulares. Eugenio Beltrami, Camille Jordan, James Joseph, Erhard Schmidt y Hermann Weyl enunciaron la existencia de los valores singulares y desarrollaron su teoría, aplicándola a diferentes ramas de las matemáticas como el álgebra lineal o las ecuaciones integrales (ver referencia [6]).

La descomposición en valores singulares es una descomposición de una matriz  $A$  como producto de tres matrices, dos matrices ortogonales e invertibles,  $U$  y  $V$ , y una matriz diagonal  $\Sigma$  tal que sus entradas son los valores singulares de  $A$  en orden decreciente. El producto resulta de la forma  $A = U\Sigma V^T$ . Esta descomposición tiene diferentes utilidades, tanto en el ámbito de las matemáticas, como en campos más prácticos. Algunos ejemplos son la compresión de imágenes, el análisis de datos o la resolución de problemas de mínimos cuadrados.

Este trabajo está dividido en dos capítulos. En el primer capítulo vamos a estudiar la existencia de la descomposición en valores singulares y algunos resultados sobre la misma, como su relación con diferentes normas, su importancia en el problema de mínimos cuadrados o el teorema de mejor aproximación. Mientras que en el segundo capítulo vamos a presentar un método numérico para obtener la descomposición en valores singulares. Este método está dividido en dos pasos, un primer paso de bidiagonalización, seguido de un segundo paso de búsqueda de los valores singulares de la bidiagonal mediante iteraciones del algoritmo  $QR$  implícito.



# Summary

One of the central concepts in matrix theory is to find a decomposition of a matrix or its canonical form. Since the 18th century, scientists have explored a matrix decomposition for square matrices based on the eigenvalues. Some of the key contributors to this theory include Leonhard Euler, Joseph-Louis Lagrange, Augustin-Louis Cauchy and Joseph Fourier. They spent much of their lives studying different mathematical fields and relating them with the matrix decomposition based on eigenvalues. However, it was not until the 19th century that a decomposition for rectangular matrices was discovered. Five mathematicians established the existence of singular values and developed their theory. Eugenio Beltrami, Camille Jordan and James Joseph Sylvester studied the singular values in the domain of linear algebra, while Erhard Schmidt and Hermann Weyl found the singular values during the study of integral equations. We can see more about their research on [6].

The Singular Value Decomposition (SVD) is a method for decomposing any matrix  $A$  as a product of three matrices: two orthogonal matrices,  $U$  and  $V$ , and a diagonal matrix,  $\Sigma$ , whose entries are the singular values of  $A$  arranged in decreasing order,  $\sigma_1, \dots, \sigma_r$  with  $r$  being the rank of  $A$ . The decomposition has this form  $A = U\Sigma V^T$ . In the current study we work with real matrices.

The decomposition is highly versatile, with applications not only in mathematics but also in many applied fields. In mathematics, the singular value decomposition can be used to solve least squares problems, to compute different norms, to determine the spectral condition number and to find the best approximation of a given rank. These elements are the foundation for practical applications such as image compression [5], latent semantic analysis [4] and dimensionality reduction, which is particularly useful in data analysis, as we can study on [3].

This study is divided into two chapters. The first chapter is focused on establishing and proving the existence of the singular value decomposition, as well as, presenting related results. The second chapter is focused on the numerical computation of the singular values.

We are going to take a closer look at the results that we are going to prove in this study, starting by the existence of the singular value decomposition (SVD). As previously mentioned, for any matrix  $A$  in  $\mathbb{R}^{m \times n}$  with rank  $r$ , it can be decomposed as

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

where

$$\begin{aligned} U &= (u_1 \ \dots \ u_r \mid u_{r+1} \ \dots \ u_m), \\ V &= (v_1 \ \dots \ v_r \mid v_{r+1} \ \dots \ v_n), \\ \Sigma &= \text{diag}(\sigma_1, \dots, \sigma_r, 0, \dots, 0), \end{aligned}$$

with  $u_1, \dots, u_m$  orthonormal basis of  $\mathbb{R}^m$ ,  $v_1, \dots, v_n$  orthonormal basis of  $\mathbb{R}^n$  and  $\sigma_1, \dots, \sigma_r$  the singular values of  $A$ , listed in decreasing order.

The proof of this result is based on finding the eigenvalues of  $A^T A$  and  $A A^T$ , which share the same nonzero eigenvalues, denoted by  $\lambda_1, \dots, \lambda_r$ . Once these eigenvalues are computed, we define the singular values of  $A$  as  $\sigma_i = \sqrt{\lambda_i}$  for  $1 \leq i \leq r$ .

After this theorem, we will explore several results related to this decomposition. Some of them involve different norms, such as the spectral norm,  $\|A\|_2 = \sigma_1$ , or the Frobenius norm,  $\|A\|_F = \sqrt{\sigma_1^2 + \dots + \sigma_r^2}$ .

Additionally, we will examine how singular values relate to the spectral condition number, for instance,  $\text{cond}(A) = \frac{\sigma_1}{\sigma_n}$ .

In the first chapter, we will dedicate an entire section to discussing the least squares problem. A fundamental problem in many applications of linear algebra is solving linear systems of equations  $Ax = b$  with  $A$  matrix in  $\mathbb{R}^{m \times n}$ ,  $b$  a vector in  $\mathbb{R}^m$  and  $x$  unknown in  $\mathbb{R}^n$ . When  $A$  is square and non-singular, the system has a unique solution given by  $x = A^{-1}b$ . However, when the system is overdetermined ( $m > n$ ), the matrix  $A$  is rectangular and the system is often incompatible. Therefore, the best option is to find  $x$  such that the residual  $r = b - Ax$  is minimal, i.e., we want to find  $x$  such that it minimizes  $\|Ax - b\|_2$ . For this purpose, we use the pseudoinverse  $A^\dagger$ . The pseudoinverse is a matrix that satisfies the following properties,

$$AA^\dagger A = A, \quad A^\dagger AA^\dagger = A^\dagger, \quad (AA^\dagger)^T = AA^\dagger \quad \text{and} \quad (A^\dagger A)^T = A^\dagger A.$$

If we have a matrix  $A$  in  $\mathbb{R}^{m \times n}$  with full rank and its SVD is  $U\Sigma V^T$ , then its pseudoinverse is given by  $A^\dagger = V\Sigma^{-1}U^T$ . This provides the solution to the least squares problem.

To conclude this chapter, we will present and prove the low-rank approximation theorem which says that, given any matrix  $A$  in  $\mathbb{R}^{m \times n}$  with rank  $r$  and SVD of the form  $A = U\Sigma V^T$ , then, for any matrix  $B$  in  $\mathbb{R}^{m \times n}$  with rank  $p \leq r$  the inequality  $\|A - B\|_2 \leq \|A - A_p\|_2$  holds, where  $A_p = U\Sigma_p V^T$  and  $\Sigma_p = \text{diag}(\sigma_1, \dots, \sigma_p, 0, \dots, 0)$ . After proving this theorem, we will demonstrate how the singular value decomposition can be applied to a practical task such as image compression. Although this method may not be the most efficient for this purpose, it is a direct and illustrative application of the low-rank approximation theorem.

We are starting the second chapter with the introduction of two important classes of orthogonal matrices that form the basis for our numerical algorithms, since they are used to create zeros in desired components of a matrix. In particular, we will study Householder reflectors and Givens rotators.

We are going to divide the numerical computation of singular values into two steps. The first step involves computing a bidiagonal matrix associated with the original matrix  $A$ . To this end, we will use the following theorem.

**Theorem:** Let  $A$  be a matrix in  $\mathbb{R}^{m \times n}$ , then there exist orthogonal matrices  $\hat{U} \in \mathbb{R}^{m \times m}$  and  $\hat{V} \in \mathbb{R}^{n \times n}$ , both being the product of a finite number of Householder matrices, and a bidiagonal matrix  $\hat{B} \in \mathbb{R}^{m \times n}$ , such that

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

However, in practice,  $m$  is often considerably larger than  $n$ . Therefore, for efficiency, the best option is to compute the *QR* factorization of the original matrix  $A$ , and then use the previous theorem to calculate the bidiagonal form  $B$  of the upper triangular matrix  $R$ . By doing so, we obtain

$$A = Q_1 \tilde{U} B \tilde{V}^T.$$

The second step in the numerical computation is to compute the singular values of the bidiagonal matrix  $B$ . The singular value decomposition of  $B$  is related to the eigenvalue decomposition of  $B^T B$  and  $BB^T$ . This means that eigenvalue calculation algorithms for symmetric matrices will be useful for computing singular values. Additionally, we note that if  $B$  is bidiagonal, then both  $B^T B$  and  $BB^T$  are tridiagonal matrices.

One of the most common algorithms for calculating the eigenvalues of real matrices is the *QR* algorithm. However, there are two reasons that make this algorithm inefficient. The first reason is the high cost of each *QR* iteration, which is  $O(n^3)$ . The second issue is the generally slow convergence. In order to solve these problems, we are going to work with tridiagonal matrices, which reduce the cost of each iteration to  $O(n)$ , and we are going to use the shifted *QR* algorithm to improve the convergence rate.

In the current study, we will prove that we can perform an iteration of the shifted *QR* algorithm on  $BB^T$  and  $B^T B$  without directly computing these products. This way, the computational cost will be reduced. Once we have found the eigenvalues of  $BB^T$  and  $B^T B$ , we just have to calculate their square roots to obtain the singular values of the bidiagonal matrix  $B$ , and consequently, the singular values of the original matrix  $A$ .

# Índice general

<b>Prólogo</b>	<b>III</b>
<b>Summary</b>	<b>v</b>
<b>1. Descomposición en Valores Singulares</b>	<b>1</b>
1.1. Definiciones y resultados previos . . . . .	1
1.2. Descomposición en Valores Singulares . . . . .	2
1.3. Problema de mínimos cuadrados . . . . .	8
1.4. Teorema de mejor aproximación y compresión de imágenes . . . . .	10
<b>2. Cálculo numérico de la Descomposición en Valores Singulares</b>	<b>15</b>
2.1. Matrices ortogonales . . . . .	16
2.2. Reducir $A$ a forma bidiagonal . . . . .	19
2.3. Calcular los valores singulares de $B$ . . . . .	21
<b>Bibliografía</b>	<b>25</b>
<b>A. Pseudocódigos</b>	<b>29</b>



# Capítulo 1

## Descomposición en Valores Singulares

Las matrices cuadradas se pueden caracterizar por sus valores y vectores propios. Recordemos que, dada una matriz cuadrada  $A$  en  $\mathbb{R}^{nxn}$ ,  $v$  un vector no nulo en  $\mathbb{R}^n$  y  $\lambda$  un escalar, tales que  $Av = \lambda v$ , entonces decimos que  $v$  es un vector propio de  $A$  asociado al valor propio  $\lambda$ .

Esta definición es muy útil en diferentes ámbitos, como el análisis de sistemas lineales, análisis de datos, descomposición espectral e incluso en física o mecánica cuántica. Sin embargo, para el caso más general de las matrices rectangulares en  $\mathbb{R}^{mxn}$ , con  $m \neq n$ , no existen valores propios. Notemos que es imposible que la igualdad  $Av = \lambda v$  se cumpla, dado que en la parte izquierda estamos en el espacio  $\mathbb{R}^m$ , mientras que en la parte derecha estamos en  $\mathbb{R}^n$ . Por ello, es necesario buscar una definición que sirva para cualquier matriz. La descomposición en valores singulares es justamente lo que necesitamos, es una herramienta para descomponer matrices muy importante, tanto teóricamente como computacionalmente.

En este capítulo, vamos a ver cómo se define la Descomposición en Valores Singulares, SVD (del inglés Singular Value Decomposition), demostraremos que dicha descomposición siempre existe y veremos distintas formas en las que la podemos expresar (ver las páginas 56, 57, 58 de [7]). También vamos a presentar algunas utilidades que tienen los valores singulares como su relación con diferentes normas, su conexión con el número de condición espectral y su papel en el problema de mínimos cuadrados. Profundizando en el problema de mínimos cuadrados y basándonos en la descomposición en valores singulares, vamos a definir la pseudoinversa, que es una herramienta útil para resolver sistemas de ecuaciones sobredeterminados (como se explica en [2]). Para finalizar el capítulo, enunciaremos y demostraremos el teorema de mejor aproximación, que nos servirá para ver cómo los valores singulares se pueden utilizar en una actividad cotidiana como la compresión de imágenes. También, vamos a desarrollar un programa con Matlab para comprimir imágenes, basándonos en el código de la página 114 de [1].

### 1.1. Definiciones y resultados previos

Antes de introducir la descomposición en valores singulares, recordemos algunas definiciones y resultados que vamos a utilizar a lo largo de esta memoria. A partir de este momento, consideraremos matrices de  $\mathbb{R}^{mxn}$  con la condición de que  $m > n$ .

**Definición 1.1.** *Dada  $A$  una matriz en  $\mathbb{R}^{mxn}$  llamamos factorización QR a una descomposición de la forma:*

$$A = Q_1 R = [Q_1 \quad Q_2] \begin{bmatrix} R \\ 0 \end{bmatrix},$$

donde  $Q$  es una matriz ortogonal en  $\mathbb{R}^{mxm}$ , llamaremos  $Q_1$  a la matriz en  $\mathbb{R}^{mxn}$  formada por las primeras  $n$  columnas de  $Q$  y  $Q_2$  a la matriz en  $\mathbb{R}^{mx(m-n)}$  formada por las últimas  $m-n$  columnas de  $Q$ , y  $R$  es una matriz triangular superior en  $\mathbb{R}^{nxn}$ .

Cuando la matriz es de rango máximo, la factorización QR, eligiendo  $R$  con las entradas de la diagonal positivas, es única.

**Definición 1.2.** Llamamos *norma matricial*  $\|\cdot\|$  a una función que asigna a cada matriz un número real no negativo y satisface las siguientes condiciones:

1. *No negatividad*,  $\|A\| \geq 0$  y  $\|A\| = 0$  si y solo si  $A = 0$ .
2. *Homogeneidad*, sea  $\alpha$  un escalar,  $\|\alpha A\| = |\alpha| \|A\|$ .
3. *Desigualdad triangular*,  $\|A + B\| \leq \|A\| + \|B\|$ .

Existen numerosas normas que cumplen estas condiciones, vamos a ver algunas de las más utilizadas.

**Definición 1.3.** La norma definida como

$$\|A\|_2 = \max_{x \neq 0} \frac{\|Ax\|_2}{\|x\|_2} \quad \text{con } x \in \mathbb{R}^n,$$

se denomina *norma espectral*. Es la norma inducida por la norma vectorial euclídea.

**Definición 1.4.** Llamamos *norma de Frobenius* a la norma definida como,

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

La siguiente propiedad justifica la utilidad de diseñar algoritmos basados en productos de matrices ortogonales.

**Propiedad 1.5.** Las matrices ortogonales preservan la norma euclídea.

*Demostración.* Llamemos  $Q$  a una matriz ortogonal en  $\mathbb{R}^{n \times n}$  y  $x$  a un vector en  $\mathbb{R}^n$ . Veamos cómo actúan bajo la acción de la norma vectorial:

$$\|Qx\|_2 = \sqrt{(Qx)^T (Qx)} = \sqrt{x^T Q^T Q x} = \sqrt{x^T x} = \|x\|_2$$

□

## 1.2. Descomposición en Valores Singulares

Comenzamos la sección introduciendo la definición de descomposición en valores singulares.

**Definición 1.6.** Sea  $A$  una matriz en  $\mathbb{R}^{m \times n}$  y sean dos conjuntos de vectores,  $u_1, \dots, u_m$  ortonormales en  $\mathbb{R}^m$  y  $v_1, \dots, v_n$  ortonormales en  $\mathbb{R}^n$ , tales que:

$$\left\{ \begin{array}{l} Av_1 = \sigma_1 u_1, \\ \dots \\ Av_r = \sigma_r u_r, \\ Av_{r+1} = 0, \\ \dots \\ Av_n = 0, \end{array} \right. \quad (1.1)$$

donde  $r$  es el rango de la matriz  $A$ . A los valores  $\sigma_i$ , con  $1 \leq i \leq r$ , los llamamos *valores singulares de la matriz  $A$* . A los vectores  $u_i$ , con  $1 \leq i \leq m$ , y  $v_j$ , con  $i \leq j \leq n$ , los denominamos *vectores singulares*.

asociados a izquierda y a derecha, respectivamente. Trasponiendo  $A$ , vemos que tenemos una relación similar entre los vectores  $u_i$  con  $1 \leq i \leq m$  y los vectores  $v_j$  con  $1 \leq j \leq r$ :

$$\begin{cases} A^T u_1 = \sigma_1 v_1, \\ \dots \\ A^T u_r = \sigma_r v_r, \\ A^T u_{r+1} = 0, \\ \dots \\ A^T u_m = 0. \end{cases} \quad (1.2)$$

La matriz  $A$  tendrá  $r$  valores singulares positivos, que consideramos en orden descendiente  $\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_r > 0$ . Los últimos  $v_j$ , con  $r+1 \leq j \leq n$ , se encuentran en el espacio nulo de  $A$ , y los  $u_i$  tales que  $r+1 \leq i \leq m$  están en el espacio nulo de  $A^T$ .

A continuación probaremos que estos vectores singulares existen para cualquier matriz.

**Teorema 1.7.** *Dada una matriz  $A$  en  $\mathbb{R}^{m \times n}$  de rango  $r$ , se puede descomponer de la forma*

$$A = U \Sigma V^T, \quad (1.3)$$

donde

$$\begin{aligned} U &= (u_1 \ \dots \ u_r \mid u_{r+1} \ \dots \ u_m), \\ V &= (v_1 \ \dots \ v_r \mid v_{r+1} \ \dots \ v_n), \\ \Sigma &= \begin{bmatrix} \sigma_1 & & & & \\ & \ddots & & & \\ & & \sigma_r & & \\ & & & 0 & \\ & & & & \ddots \\ & & & & & 0 \end{bmatrix}, \end{aligned}$$

con  $u_1, \dots, u_m$  base ortonormal de  $\mathbb{R}^m$  y  $v_1, \dots, v_n$  base ortonormal de  $\mathbb{R}^n$ . A la factorización (1.3) la denominamos *Descomposición en Valores Singulares*, *SVD* (en inglés, *Singular Value Decomposition*).

*Demostración.* Primero vamos a ver que  $A^T A$  y  $AA^T$  tienen los mismos valores propios no nulos, puesto que esta propiedad será usada en esta demostración. Para ello, asumimos que  $\lambda \neq 0$  es valor propio de  $A^T A$ . Entonces, existe  $x \neq 0$  en  $\mathbb{R}^n$  tal que  $A^T A x = \lambda x$ , y, si multiplicamos esta igualdad por  $A$  a izquierda, tenemos que  $AA^T A x = \lambda A x$ . Es decir,  $A x$  es vector propio de  $AA^T$  asociado al valor propio  $\lambda$ . Como esto es cierto para todos los valores propios no nulos,  $A^T A$  y  $AA^T$  tienen los mismos valores propios no nulos.

Además, sabemos que  $A^T A$  y  $AA^T$  son matrices simétricas, y por tanto, diagonalizables. Sus valores propios son reales y podemos tomar vectores propios ortonormales. También son semidefinidas positivas, por lo que, todos sus valores propios son mayores o iguales a 0 y, las podemos descomponer de la siguiente manera:

$$\begin{aligned} A^T A &= V D_1 V^T, \\ AA^T &= U D_2 U^T. \end{aligned}$$

En esta descomposición  $V$  y  $U$  son las matrices ortogonales formadas por sus vectores propios respectivamente.  $D_1$  es la matriz diagonal con los valores propios reales  $\lambda_1, \dots, \lambda_n$  y  $D_2$  es la matriz diagonal con los valores propios  $\lambda_1, \dots, \lambda_m$ . Recordemos que el rango de  $A$  es  $r$ , por lo que el rango de  $A^T A$  y  $AA^T$  también es  $r$ , y, por tanto,  $\lambda_i = 0$  para  $i > r$ . Sin pérdida de generalidad, ordenamos los valores propios de forma decreciente, es decir,  $\lambda_1 \geq \dots \geq \lambda_r > 0$ . A continuación, definimos

$$\sigma_i = \sqrt{\lambda_i}, \quad \text{para cada } 1 \leq i \leq r \quad (1.4)$$

está bien definido porque cada  $\lambda_i$  es mayor estricto que 0. Luego los valores propios no nulos de  $A^T A$  y  $AA^T$  son  $\sigma_1^2, \dots, \sigma_r^2$ . Consideramos los vectores propios ortonormales  $v_1, \dots, v_r$  de  $A^T A$ , que cumplen:

$$A^T A v_k = \sigma_k^2 v_k, \quad \text{para } 1 \leq k \leq r.$$

Y ahora, definimos los vectores  $u_k$ , con  $1 \leq k \leq r$  de la siguiente forma:

$$u_k = \frac{A v_k}{\sigma_k}. \quad (1.5)$$

Así se obtienen los valores singulares y vectores singulares de A. Falta ver que  $u_1, \dots, u_r$  son vectores propios ortonormales de  $AA^T$ . Tomando  $1 \leq k \leq r$ , vemos que

$$\begin{aligned} AA^T u_k &= AA^T \left( \frac{A v_k}{\sigma_k} \right) = A \left( \frac{A^T A v_k}{\sigma_k} \right) = A \frac{\sigma_k^2 v_k}{\sigma_k} = \sigma_k^2 \frac{A v_k}{\sigma_k} = \sigma_k^2 u_k, \\ u_j^T u_k &= \left( \frac{A v_j}{\sigma_j} \right)^T \left( \frac{A v_k}{\sigma_k} \right) = \frac{v_j^T (A^T A v_k)}{\sigma_j \sigma_k} = \frac{\sigma_k v_j^T v_k}{\sigma_j} = \begin{cases} 1, & \text{si } j = k, \\ 0, & \text{si } j \neq k. \end{cases} \end{aligned}$$

Para acabar de formar la base ortonormal tenemos que elegir los  $n - r$  vectores  $v_{r+1}, \dots, v_n$  en  $\mathbb{R}^n$  y los  $m - r$  vectores  $u_{r+1}, \dots, u_m$  en  $\mathbb{R}^m$ . Los vectores  $v_j$  ( $r + 1 \leq j \leq n$ ) se encuentran en el espacio nulo de A, mientras que los vectores  $u_i$  ( $r + 1 \leq i \leq m$ ) están en el espacio nulo de  $A^T$ . Por tanto, cualquier base ortonormal de ambos espacios nulos va a ser ortogonal a los vectores singulares que ya tenemos respectivamente.

Concluimos la demostración viendo que, efectivamente, hemos obtenido una descomposición de la forma  $A = U\Sigma V^T$ . Para ello tomamos las ecuaciones de (1.1) y la ponemos de forma matricial,

$$V = \left( \begin{array}{ccc|cc} v_1 & \cdots & v_r & v_{r+1} & \cdots & v_n \end{array} \right),$$

$$U = \left( \begin{array}{ccc|cc} u_1 & \cdots & u_r & u_{r+1} & \cdots & u_m \end{array} \right),$$

$$\Sigma = \begin{bmatrix} \sigma_1 & & & & & \\ & \ddots & & & & \\ & & \sigma_r & & & \\ & & & 0 & & \\ & & & & \ddots & \\ & & & & & 0 \end{bmatrix}.$$

Así obtenemos la ecuación

$$AV = U\Sigma,$$

y como V es ortogonal, al multiplicar a ambos lados por  $V^T$ , concluimos que

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

como queríamos demostrar. □

En el siguiente resultado, mostramos que la descomposición en valores singulares también se puede escribir como la suma de r matrices de rango 1.

**Lema 1.8.** *La descomposición en valores singulares nos permite expresar A como suma de r matrices de rango 1 de la siguiente forma:*

$$A = U\Sigma V^T = \sigma_1 u_1 v_1^T + \dots + \sigma_r u_r v_r^T, \quad (1.6)$$

donde r es el rango de la matriz A,  $\sigma_i$  son los valores singulares de A y  $u_i$  y  $v_i$ , con  $1 \leq i \leq r$ , son los vectores singulares asociados a izquierda y a derecha, respectivamente.

*Demostración.* Llamemos  $B$  a la parte derecha de la igualdad,  $B = \sum_{i=1}^r \sigma_i u_i v_i^T$ , y vamos a comprobar que  $A = B$ . Para ello, vamos a ver que  $Av_j = Bv_j$  para cada  $1 \leq j \leq n$ . Primero, por la definición de valor singular sabemos que

$$\begin{cases} Av_j = \sigma_j u_j, & \text{para } 1 \leq j \leq r, \\ Av_j = 0, & \text{para } r+1 \leq j \leq n. \end{cases}$$

A continuación, estudiamos el producto  $Bv_j$ :

$$Bv_j = \left( \sum_{i=1}^r \sigma_i u_i v_i^T \right) v_j = \sum_{i=1}^r \sigma_i u_i (v_i^T v_j) .$$

Teniendo en cuenta que  $v_1, \dots, v_n$  es una base ortonormal, se cumple que  $v_i^T v_j = 0$  cuando  $i \neq j$  y  $v_i^T v_i = 1$  si  $i = j$ . Por tanto, tenemos que

$$\begin{cases} Bv_j = \sigma_j u_j, & \text{para } 1 \leq j \leq r, \\ Bv_j = 0, & \text{para } r+1 \leq j \leq n. \end{cases}$$

Hemos visto que, efectivamente,  $Av_j = Bv_j$  para cada  $1 \leq j \leq n$ . Por tanto, concluimos que

$$A = B = \sum_{i=1}^r \sigma_i u_i v_i^T .$$

□

Existe una forma reducida de la descomposición en valores singulares, que consiste en suprimir todos los vectores singulares asociados a cero.

**Teorema 1.9.** *Dada una matriz  $A$  en  $\mathbb{R}^{m \times n}$  no nula con rango  $r$ , se puede decomponer de la forma*

$$A = U_r \Sigma_r V_r^T , \quad (1.7)$$

donde  $U_r$  es una matriz en  $\mathbb{R}^{m \times r}$  formada por las primeras  $r$  columnas de  $U$  y  $V_r$  una matriz en  $\mathbb{R}^{n \times r}$  formada por las primeras  $r$  columnas de  $V$ , con  $U$  y  $V$  definidas en el Teorema 1.7. La matriz  $\Sigma_r$  es una matriz diagonal  $r \times r$  cuyos elementos diagonales son  $\sigma_1, \sigma_2, \dots, \sigma_r$ .

*Demostración.* La fórmula (1.7) es consecuencia directa del Teorema 1.7 y la factorización (1.3). □

A continuación, enunciamos varias propiedades interesantes de la descomposición en valores singulares.

**Proposición 1.10.** *Dada  $S = Q \Delta Q^T$  una matriz simétrica, definida positiva y cuyos valores propios están ordenados en orden decreciente. Entonces la descomposición en valores singulares de  $S$  es  $U \Sigma V^T = Q \Delta Q^T$ .*

*Demostración.* Basta con tomar  $U = V = Q$  ya que son matrices ortogonales. Lo que implica  $\Sigma = \Delta$ . □

**Proposición 1.11.** *Si  $S = Q \Delta Q^T$  tiene un valor propio negativo ( $Sx = -\alpha x$ ), su valor singular asociado es  $\sigma = +\alpha$  y uno de los vectores singulares será  $-x$ .*

**Proposición 1.12.** *Si  $A = Q$  es una matriz ortogonal, todos sus valores singulares son 1.*

*Demostración.* Como hemos visto antes, los valores singulares son la raíz de los valores propios de  $A^T A$  (1.4), y en este caso  $A^T A = Q^T Q = I$ . Por tanto, los valores propios de  $Q^T Q$  son 1 y  $\Sigma = I$ . □

En los siguientes resultados se muestra parte de la información que los valores singulares contienen sobre el espectro de una matriz.

**Teorema 1.13.** *Sea  $A$  una matriz en  $\mathbb{R}^{m \times n}$  cuyos valores singulares son  $\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_r$ . Entonces, se tiene que  $\|A\|_2 = \sigma_1$ .*

*Demostración.* Basándonos en la definición de norma espectral, Definición 1.3, queremos ver que

$$\|A\|_2 = \max_{x \neq 0} \frac{\|Ax\|_2}{\|x\|_2} = \sigma_1.$$

Primero, tomando  $x = v_1$  obtenemos,

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

es decir,  $\|A\|_2$  es como mínimo  $\sigma_1$ ,  $\|A\|_2 \geq \sigma_1$ .

A continuación, podemos expresar  $x \in \mathbb{R}^n$  como combinación lineal de  $v_1, \dots, v_n$  ya que es una base ortonormal del espacio,

$$x = c_1 v_1 + \dots + c_n v_n \quad \text{con} \quad c_1, \dots, c_n \in \mathbb{R}.$$

Entonces, utilizando que  $v_1, \dots, v_n$  y  $u_1, \dots, u_m$  son bases ortonormales obtenemos que

$$\|x\|_2^2 = \|c_1 v_1 + \dots + c_n v_n\|_2^2 = |c_1|^2 \|v_1\|_2^2 + \dots + |c_n|^2 \|v_n\|_2^2 = |c_1|^2 + \dots + |c_n|^2,$$

$$Ax = c_1 Av_1 + \dots + c_n Av_n = c_1 \sigma_1 u_1 + \dots + c_r \sigma_r u_r + 0 + \dots + 0,$$

$$\|Ax\|_2^2 = \|c_1 \sigma_1 u_1 + \dots + c_r \sigma_r u_r\|_2^2 = |c_1|^2 \sigma_1^2 \|u_1\|_2^2 + \dots + |c_r|^2 \sigma_r^2 \|u_r\|_2^2 = \sigma_1^2 |c_1|^2 + \dots + \sigma_r^2 |c_r|^2.$$

Por último, utilizando estas igualdades y teniendo en cuenta que  $\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_r$  podemos ver que  $\|A\|_2 \leq \sigma_1$ :

$$\|Ax\|_2^2 = \sigma_1^2 |c_1|^2 + \dots + \sigma_r^2 |c_r|^2 \leq \sigma_1^2 (|c_1|^2 + \dots + |c_r|^2) = \sigma_1^2 \|x\|_2^2,$$

$$\|A\|_2 = \max_{x \neq 0} \frac{\|Ax\|_2}{\|x\|_2} \leq \max_{x \neq 0} \frac{\sigma_1 \|x\|_2}{\|x\|_2} = \sigma_1.$$

Llegamos a la conclusión de que  $\|A\|_2 \leq \sigma_1$  y como también habíamos visto que  $\|A\|_2 \geq \sigma_1$ , entonces concluimos que  $\|A\|_2 = \sigma_1$ .  $\square$

El teorema anterior nos da una demostración inmediata del siguiente enunciado.

**Corolario 1.14.** *Sea  $A$  una matriz en  $\mathbb{R}^{m \times n}$  cuyos valores singulares son  $\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_r$ , entonces  $\|A\|_2 = \|A^T\|_2$ .*

*Demostración.* Como  $A$  y  $A^T$  tienen los mismos valores singulares  $\|A\|_2 = \|A^T\|_2 = \sigma_1$ .  $\square$

**Proposición 1.15.** *El módulo de los valores propios de una matriz cuadrada  $A$  es siempre menor o igual que su mayor valor singular  $\sigma_1$ .*

*Demostración.* Recordemos que las matrices ortogonales preservan las normas vectoriales, luego

$$\|Ax\|_2 = \|U\Sigma V^T x\|_2 = \|\Sigma V^T x\|_2 \leq \|\Sigma\|_2 \|V^T x\|_2 = \sigma_1 \|V^T x\|_2 = \sigma_1 \|x\|_2.$$

Como la igualdad anterior es cierta para cada  $x$ , sustituyendo  $\|Ax\|_2 = |\lambda| \|x\|_2$  obtenemos  $|\lambda| \leq \sigma_1$  como queríamos demostrar.  $\square$

La norma de Frobenius de una matriz también queda caracterizada por sus valores singulares.

**Teorema 1.16.** *Sea  $A$  una matriz en  $\mathbb{R}^{m \times n}$  cuyos valores singulares son  $\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_r$ , la norma de Frobenius de una matriz es de la forma:*

$$\|A\|_F = \sqrt{\sigma_1^2 + \sigma_2^2 + \dots + \sigma_r^2}.$$

*Demostración.* La norma de Frobenius de una matriz tiene la siguiente definición:

$$\|A\|_F = \sqrt{\sum_{i=1}^m \sum_{j=1}^n a_{ij}^2} = \sqrt{\text{tr}(A^T A)} .$$

Vamos a calcular  $A^T A$  usando la descomposición en valores singulares,

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

Dado que  $V$  es una matriz ortogonal, tenemos que

$$\text{tr}(A^T A) = \text{tr}(\Sigma^2) = \sigma_1^2 + \dots + \sigma_r^2 .$$

El resultado es la suma del cuadrado de los valores singulares. Así podemos concluir que  $\sqrt{\text{tr}(A^T A)} = \sqrt{\sigma_1^2 + \dots + \sigma_r^2}$ , por lo que se tiene que  $\|A\|_F = \sqrt{\sigma_1^2 + \dots + \sigma_r^2}$ .  $\square$

Los valores singulares también sirven para medir el condicionamiento de una matriz, como se muestra en los siguientes resultados.

**Definición 1.17.** Dada una matriz  $A$  en  $\mathbb{R}^{n \times n}$  invertible, consideramos el sistema lineal  $Ax = b$ , con  $b$  un vector en  $\mathbb{R}^n$ . El número de condiciónpectral mide cuánto varía la solución del sistema lineal al realizar perturbaciones en  $A$  o  $b$ . Se define como

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

**Teorema 1.18.** Sea  $A$  una matriz invertible en  $\mathbb{R}^{n \times n}$  cuyos valores singulares son  $\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_n > 0$ . El número de condiciónpectral de  $A$  es

$$\text{cond}(A) = \frac{\sigma_1}{\sigma_n} .$$

*Demostración.* Para calcular la norma euclídea de la matriz inversa hay que proceder de manera similar a la demostración del Teorema 1.13, es decir, aplicamos "la regla del sándwich" sobre

$$\|A^{-1}\|_2 = \max_{x \neq 0} \frac{\|A^{-1}x\|_2}{\|x\|_2} = \min_{x \neq 0} \frac{\|x\|_2}{\|Ax\|_2} .$$

Y llegamos a

$$\|A^{-1}\|_2 = \frac{1}{\sigma_n} .$$

Haciendo el producto de ambas normas obtenemos que

$$\text{cond}(A) = \|A\|_2 \|A^{-1}\|_2 = \sigma_1 \frac{1}{\sigma_n} = \frac{\sigma_1}{\sigma_n} .$$

$\square$

**Teorema 1.19.** Dada  $A$  una matriz en  $\mathbb{R}^{n \times n}$  invertible, entonces se tiene que  $\text{cond}(A^T A) = \text{cond}(A)^2$ .

*Demostración.* Si los valores singulares de  $A$  son  $\sigma_1, \dots, \sigma_n$ , es fácil ver que los de  $A^T A$  son  $\sigma_1^2, \dots, \sigma_n^2$ . Para ello calculamos su descomposición en valores singulares  $A^T A = (U \Sigma V^T)^T (U \Sigma V^T) = V \Sigma^T U^T U \Sigma V^T = V \Sigma^2 V^T$ . Aplicando la propiedad anterior del número de condiciónpectral a la izquierda de la igualdad obtenemos,

$$\text{cond}(A^T A) = \frac{\sigma_1^2}{\sigma_n^2} .$$

Haciendo lo mismo al lado derecho de la igualdad llegamos a,

$$\text{cond}(A)^2 = \left( \frac{\sigma_1}{\sigma_n} \right)^2 ,$$

demonstrando así la igualdad.  $\square$

### 1.3. Problema de mínimos cuadrados

Un problema fundamental en muchas aplicaciones del álgebra lineal consiste en estudiar sistemas de ecuaciones lineales  $Ax = b$  con  $A$  matriz en  $\mathbb{R}^{m \times n}$ ,  $b$  un vector en  $\mathbb{R}^m$  y  $x$  un vector indeterminado de dimensión  $n$ .

Cuando la matriz es cuadrada e invertible, el sistema tiene solución y es única,  $x = A^{-1}b$ . Pero, cuando nos encontramos con un problema sobre determinado, es decir, con más ecuaciones que incógnitas ( $m > n$ ), la matriz  $A$  es rectangular y el sistema muchas veces en la práctica es incompatible. Por ello, lo mejor a lo que podemos optar es buscar un  $x$  tal que el residuo  $r = b - Ax$  sea mínimo, es decir, buscamos  $x$  tal que minimice  $\|Ax - b\|_2$ .

**Teorema 1.20.** *Dado un subespacio  $V$  de  $\mathbb{R}^n$  y  $x$  un vector en  $\mathbb{R}^n$ . Entonces la menor distancia entre  $x$  y  $V$  es*

$$\min_{v \in V} \|x - v\|_2 = \|x - y\|_2 ,$$

donde  $y$  es la proyección ortogonal de  $x$  sobre  $V$ .

En el problema de mínimos cuadrados el subespacio  $V$  es el subespacio generado por las columnas de  $A$  y la mejor aproximación de  $b$  es su proyección ortogonal sobre el subespacio columna de  $A$ , la cual llamaremos  $A\hat{x}$ . Además, el residuo  $r = b - A\hat{x}$  tiene que ser ortogonal al subespacio columna de  $A$ , es decir,

$$A^T(b - A\hat{x}) = 0 .$$

Luego,  $A^Tb - A^TA\hat{x} = 0$  y llegamos a

$$A^TA\hat{x} = A^Tb . \quad (1.8)$$

Al sistema definido por (1.8) se le conoce como sistema de ecuaciones normales. Recordemos que es un sistema de ecuaciones lineales, donde  $A^TA$  es una matriz cuadrada de tamaño  $n \times n$ . Sabemos que cuando su rango es máximo el problema está bien definido y tiene solución única. Por tanto, asumimos que el rango de  $A$  es máximo y así el rango de  $A^TA$  también lo es. Vamos a estudiar la solución de este sistema aplicando la descomposición en valores singulares. Así, aplicamos la descomposición en valores singulares (1.3) sobre el sistema de ecuaciones normales (1.8)

$$A^TA\hat{x} = A^Tb ,$$

$$V\Sigma U^T U\Sigma V^T \hat{x} = V\Sigma U^T b ,$$

y multiplicamos a izquierda por  $V^T$

$$V^T V\Sigma^2 V^T \hat{x} = V^T V\Sigma U^T b ,$$

$$\Sigma^2 V^T \hat{x} = \Sigma U^T b .$$

Como el rango de  $A$  es máximo, ningún valor singular es nulo. Entonces, multiplicamos a ambos lados por  $(\Sigma^{-1})^2$  y después por  $V$ , obteniendo que

$$V^T \hat{x} = \Sigma^{-1} U^T b ,$$

$$\hat{x} = V\Sigma^{-1} U^T b .$$

Vemos que la solución única en sentido de mínimos cuadrados queda definida en función de la descomposición en valores singulares. Podemos ver que  $V\Sigma^{-1}U^T$  juega el mismo papel que  $A^{-1}$  en el caso de un sistema compatible determinado  $Ax = b$  con  $A$  regular. Esto nos motiva a introducir la definición de pseudoinversa para matrices más generales.

**Definición 1.21.** *Sea  $A$  una matriz en  $\mathbb{R}^{m \times n}$ . Llamamos matriz pseudoinversa de  $A$  a la matriz  $A^\dagger$  en  $\mathbb{R}^{n \times m}$ , la cual satisface:*

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

Vamos a comprobar que la pseudoinversa que hemos definido es única.

**Teorema 1.22.** *Dada una matriz  $A$  en  $\mathbb{R}^{m \times n}$  existe una única matriz pseudoinversa  $A^\dagger$ .*

*Demostración.* Supongamos que existen dos matrices  $P = A_1^\dagger$  y  $Q = A_2^\dagger$  en  $\mathbb{R}^{n \times m}$  las cuales cumplen las cuatro propiedades de la Definición de pseudoinversa 1.21, luego:

$$P = PAP = (PA)^T P = (A^T P^T)P = (AQA)^T P^T P = (A^T Q^T A^T)P^T P = (QA)^T (PA)^T P = QAPAP = QAP.$$

$$Q = QAQ = Q(AQ)^T = QQ^T A^T = QQ^T (APA)^T = QQ^T (A^T P^T A^T) = Q(AQ)^T (AP)^T = Q(AQA)P = QAP.$$

Esto demuestra que  $P = Q$ , luego la pseudoinversa es única.  $\square$

Basándonos en la deducción anterior, podemos expresar esta pseudoinversa en términos de la descomposición en valores singulares.

**Teorema 1.23.** *Sea  $A$  una matriz en  $\mathbb{R}^{m \times n}$  con rango máximo y cuya descomposición en valores singulares es  $U\Sigma V^T$ . Entonces*

$$A^\dagger = V\Sigma^{-1}U^T \quad (1.9)$$

*es su pseudoinversa.*

*Demostración.* Para demostrar que  $V\Sigma^{-1}U^T$  es una matriz pseudoinversa tenemos que comprobar que satisface las cuatro propiedades de la Definición 1.21. Todas las propiedades se pueden demostrar utilizando la descomposición en valores singulares y teniendo en cuenta que  $U$  y  $V$  son matrices ortogonales.

1.  $AA^\dagger A = (U\Sigma V^T)(V\Sigma^{-1}U^T)(U\Sigma V^T) = U\Sigma\Sigma^{-1}\Sigma V^T = U\Sigma V^T = A$ .
2.  $A^\dagger AA^\dagger = (V\Sigma^{-1}U^T)(U\Sigma V^T)(V\Sigma^{-1}U^T) = V\Sigma^{-1}\Sigma\Sigma^{-1}U^T = V\Sigma^{-1}U^T = A^\dagger$ .
3. Descomponemos la parte izquierda de la igualdad:  $(AA^\dagger)^T = ((U\Sigma V^T)(V\Sigma^{-1}U^T))^T = (U\Sigma\Sigma^{-1}U^T)^T = U(\Sigma\Sigma^{-1})^T U^T = U\Sigma\Sigma^{-1}U^T$ . Descomponemos ahora la parte derecha:  $AA^\dagger = (U\Sigma V^T)(V\Sigma^{-1}U^T) = U\Sigma\Sigma^{-1}U^T$ . Vemos que se cumple la igualdad.
4. Procedemos igual que antes. Estudiamos primero la parte izquierda:  $(A^\dagger A)^T = ((V\Sigma^{-1}U^T)(U\Sigma V^T))^T = (V\Sigma^{-1}\Sigma V^T)^T = V(\Sigma^{-1}\Sigma)^T V^T = V\Sigma^{-1}\Sigma V^T$ . Vamos ahora con el lado derecho:  $A^\dagger A = (V\Sigma^{-1}U^T)(U\Sigma V^T) = V\Sigma^{-1}\Sigma V^T$ . Se satisface la propiedad.

Como satisface las cuatro propiedades,  $V\Sigma^{-1}U^T$  es la pseudoinversa de  $A$ .  $\square$

En particular, la pseudoinversa también tiene descomposición en valores singulares.

**Teorema 1.24.** *Sea  $A$  una matriz en  $\mathbb{R}^{m \times n}$  con rango máximo y cuya descomposición en valores singulares es  $A = U\Sigma V^T$ , entonces la descomposición en valores singulares de su matriz pseudoinversa  $A^\dagger$  es*

$$A^\dagger = V\Sigma^{-1}U^T,$$

ya que,

$$\begin{cases} A^\dagger u_1 = \sigma_1^{-1} v_1, \\ \dots \\ A^\dagger u_n = \sigma_n^{-1} v_n, \\ A^\dagger u_{n+1} = 0, \\ \dots \\ A^\dagger u_m = 0. \end{cases} \quad (1.10)$$

Aunque hemos considerado el caso de  $A$  de rango máximo a la hora de resolver problemas de mínimos cuadrados, la definición de matriz pseudoinversa se puede generalizar a matrices de cualquier rango  $r$ , como mostramos a continuación.

**Definición 1.25.** *Sea  $A$  una matriz en  $\mathbb{R}^{m \times n}$  con rango  $r$  y cuya descomposición en valores singulares es  $A = U\Sigma V^T$  como en (1.3). Entonces*

$$A^\dagger = V\Sigma^\dagger U^T$$

donde  $\Sigma^\dagger$  es la pseudoinversa de la matriz diagonal  $\Sigma$ . La matriz  $\Sigma^\dagger$  tiene dimensión  $n \times m$  y se obtiene invirtiendo los valores singulares de  $A$  distintos de cero y manteniendo los ceros;

$$\Sigma^\dagger = \begin{bmatrix} \frac{1}{\sigma_1} & & & \\ & \ddots & & \\ & & \frac{1}{\sigma_r} & \\ & & & 0 \\ & & & & \ddots \\ & & & & & 0 \end{bmatrix}.$$

Observamos que  $u_1, \dots, u_m$  y  $v_1, \dots, v_n$  son los vectores singulares a derecha y a izquierda de  $A^\dagger$  y  $\sigma_1^{-1}, \dots, \sigma_r^{-1}$  sus valores singulares.

Notemos que esta definición es compatible con la definición que hemos dado previamente de pseudoinversa para matrices de rango máximo. Además, dada esta descomposición podemos deducir que el rango de  $A$  y de su pseudoinversa  $A^\dagger$  coincide.

**Corolario 1.26.** *La matriz  $A$  y  $A^\dagger$  tienen el mismo rango  $r$ .*

Todo esto nos sirve para definir la solución del problema de mínimos cuadrados. Sea  $A$  una matriz en  $\mathbb{R}^{m \times n}$  y  $b$  un vector en  $\mathbb{R}^n$ , y sea  $\hat{x}$  en  $\mathbb{R}^m$  la solución de norma mínima de

$$\|b - A\hat{x}\|_2 = \min_{x \in \mathbb{R}^m} \|b - Ax\|_2.$$

Entonces, se tiene que  $\hat{x} = A^\dagger b = V\Sigma^\dagger U^T b$ .

## 1.4. Teorema de mejor aproximación y compresión de imágenes

La descomposición en valores singulares tiene una aplicación directa en el análisis de datos y las matemáticas aplicadas, ya que sirve para encontrar aproximaciones de menor rango de una matriz, lo que se puede emplear para abaratar los costes computacionales de numerosos métodos numéricos. Podemos ver más acerca de otras aplicaciones en [5], [3] o [4].

**Teorema 1.27.** *(Teorema de mejor aproximación) Sea  $A$  una matriz en  $\mathbb{R}^{m \times n}$  con rango  $r$ , cuya descomposición en valores singulares es  $A = U\Sigma V^T$ . Entonces para cada  $B$  en  $\mathbb{R}^{m \times n}$  con rango  $p \leq r$*

$$\|A - A_p\|_2 \leq \|A - B\|_2, \quad (1.11)$$

con  $A_p = U\Sigma_p V^T$ , donde  $\Sigma_p$  es una matriz diagonal con los valores singulares de  $A$  en las primeras  $p$  entradas y 0 en las últimas,

$$\Sigma = \begin{bmatrix} \sigma_1 & & & & \\ & \ddots & & & \\ & & \sigma_p & & \\ & & & 0 & \\ & & & & \ddots \\ & & & & & 0 \end{bmatrix}.$$

*Demostración.* Comenzamos la demostración observando que  $\|A - A_p\|_2 = \sigma_{p+1}$ , ya que por el Teorema 1.13, la norma euclídea de una matriz es su mayor valor singular, y en el caso de  $A - A_p$ , éste es  $\sigma_{p+1}$ . Por otro lado, vamos a estudiar

$$\|A - B\|_2 = \max_{x \neq 0} \frac{\|(A - B)x\|_2}{\|x\|_2}.$$

Vamos a tomar un vector no nulo  $x$  tal que  $Bx = 0$  y  $x = \sum_{i=1}^{p+1} c_i v_i$ , con  $c_i$  en  $\mathbb{R}$  coeficientes y  $v_i$  los vectores singulares asociados a derecha de  $A$ . Primero, veamos que este vector existe. Para ello, vamos a ver que la intersección de los dos subespacios a los que pertenece es no nula. El espacio nulo de  $B$ ,  $\text{Null}(B)$ , tiene dimensión mayor o igual que  $n - p$ , dado que  $B$  tiene rango  $p$ . Además, el subespacio generado por los vectores asociados a derecha  $v_i$  con  $1 \leq i \leq p+1$  tiene dimensión  $p+1$ , ya que éstos son una base. Luego la intersección de ambos subespacios es no nula y existe un vector  $x$  que satisface ambas condiciones.

$$\text{Null}(B) \cap \mathbb{R} \langle v_1, \dots, v_{p+1} \rangle \neq \emptyset.$$

Como hemos adelantado, tomamos  $x$  que pertenece a dicha intersección y vemos que cumple

$$\|(A - B)x\|_2^2 = \|Ax\|_2^2 = \|A \sum_{i=1}^{p+1} c_i v_i\|_2^2 = \left\| \sum_{i=1}^{p+1} c_i \sigma_i u_i \right\|_2^2 = \sum_{i=1}^{p+1} c_i^2 \sigma_i^2 \|u_i\|_2^2 = \sum_{i=1}^{p+1} c_i^2 \sigma_i^2.$$

Recordemos que  $\|x\|_2^2 = c_1^2 + \dots + c_{p+1}^2$  y que  $\sigma_1 \geq \dots \geq \sigma_r$ , por tanto,

$$\|(A - B)x\|_2^2 = c_1^2 \sigma_1^2 + \dots + c_{p+1}^2 \sigma_{p+1}^2 \geq (c_1^2 + \dots + c_{p+1}^2) \sigma_{p+1}^2 = \sigma_{p+1}^2 \|x\|_2^2.$$

Es decir,  $\|(A - B)x\|_2 \geq \sigma_{p+1} \|x\|_2$ ,

$$\|A - B\|_2 = \max_{x \neq 0} \frac{\|(A - B)x\|_2}{\|x\|_2} \geq \max_{x \neq 0} \frac{\sigma_{p+1} \|x\|_2}{\|x\|_2} = \sigma_{p+1}.$$

Luego, juntando lo anterior tenemos que

$$\|A - A_p\|_2 = \sigma_{p+1} \leq \|A - B\|_2.$$

□

Otra forma de escribir la mejor aproximación es la siguiente.

**Corolario 1.28.** *La mejor aproximación  $A_p$  se puede expresar como suma de  $p$  matrices de rango 1,*

$$A_p = \sigma_1 u_1 v_1^T + \dots + \sigma_p u_p v_p^T.$$

*Demostración.* Es una consecuencia directa del Lema 1.8. □

Una aplicación directa de este resultado es la compresión de imágenes. Una imagen puede ser interpretada como una matriz  $m \times n$ , en la que cada componente representa la tonalidad de un píxel. Es decir, cada entrada es un valor entre 0 y 255, donde 0 es el color negro, 255 es el blanco y el resto de valores intermedios es la intensidad de gris de dicho píxel. A la hora de almacenar una imagen, en lugar de almacenar toda esta información ( $m \times n$ ), empleamos la compresión de imágenes. La compresión consiste en almacenar ciertos valores que nos permitirán reconstruir la matriz de manera aproximada. Una forma de hacer este proceso es a través de los valores singulares y el teorema de mejor aproximación (Teorema 1.27). Ésta no es la única forma ni la más óptima que existe para realizar la compresión de imágenes (ver por ejemplo [5]), pero es un ejemplo práctico y visual de la utilidad de la descomposición en valores singulares.

El teorema de mejor aproximación nos beneficia a la hora de almacenar información ya que eligiendo cierto rango  $p$  la información almacenada se reduce a  $(m + n)p$ ,  $p$  vectores de dimensión  $m$ ,  $\sigma_1 u_1, \dots, \sigma_p u_p$  y  $p$  vectores de dimensión  $n$ ,  $v_1, \dots, v_p$ .

Vamos a usar Matlab para ver cómo funciona la compresión de imágenes basándonos en este teorema. Tomamos la siguiente foto de la facultad de matemáticas, la cargamos en Matlab y la ponemos en blanco y negro para simplificar el proceso:

```
>> img=imread('FacultadMatematicas.jpg');
if size(img,3)==3
    img_gray=rgb2gray(img);
else
    img_gray= img;
end
colormap('gray');
image(img_gray);
matriz=double(img_gray);
```

Figura 1.1: Código empleado para pasar de una imagen a una matriz que recoge la tonalidad de grises



Figura 1.2: Facultad de matemáticas original y en blanco y negro

Ahora, aplicamos la descomposición en valores singulares a la matriz y, a continuación, empleamos el Teorema 1.27 para diferentes valores de  $p$  e imprimimos la imagen por pantalla.

```
[U,S,V]=svd(matriz);
image(U(:,1:p)*S(1:p,1:p)*V(:,1:p)');
```

Figura 1.3: Código empleado para ejecutar la SVD y aplicar el Teorema de mejor aproximación a distintos rangos  $p$

Las imágenes obtenidas aplicando diferentes rangos  $p$  son las siguientes

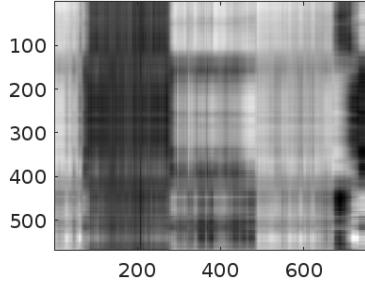


Figura 1.4:  $p = 3$

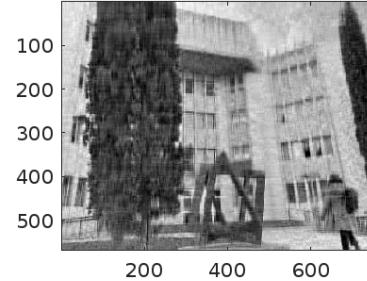


Figura 1.5:  $p = 30$

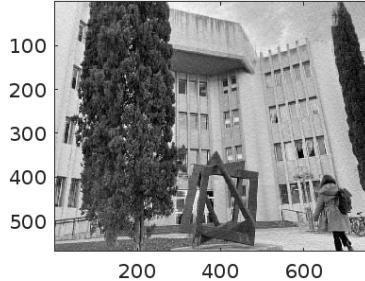


Figura 1.6:  $p = 100$

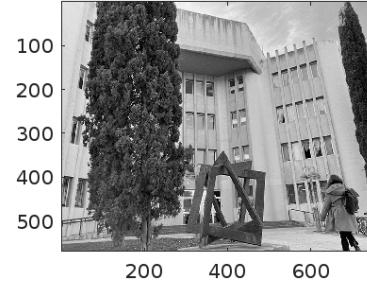


Figura 1.7:  $p = 300$

La imagen original tiene dimensión  $569 \times 759 = 431871$ , es decir, es una matriz con 431871 componentes. Con  $p = 100$ , la imagen ya se percibe muy similar a la original, y la cantidad de información que hemos necesitado es  $(569 + 759)100 = 132800$ , que es notablemente menor que la cantidad original. Observamos también que cuando el rango  $p$  es 30, la imagen ya se percibe y se reconoce. Luego parece que el rango necesario para reconocer una imagen va a ser considerablemente menor que el de la matriz original, por lo que la información que necesitamos almacenar se va a ver reducida. Esto demuestra de forma gráfica la utilidad de los valores singulares en aplicaciones de compresión de información.



## Capítulo 2

# Cálculo numérico de la Descomposición en Valores Singulares

La descomposición en valores singulares de una matriz  $A$  está íntimamente relacionada con la descomposición en valores propios de las matrices  $A^T A$  y  $AA^T$ , como hemos visto en el capítulo anterior. Por tanto, los algoritmos de cálculo de valores propios para matrices simétricas son un buen punto de partida para la obtención de algoritmos de cálculo de la descomposición en valores singulares. No obstante, la adaptación de estos algoritmos no es directa, ya que la estructura especial de la descomposición en valores singulares puede ser aprovechada para desarrollar algoritmos que sean más eficientes o más precisos (ver por ejemplo [2]).

Uno de los algoritmos más comunes para calcular los valores propios de una matriz real  $A$  es el algoritmo  $QR$ . El algoritmo  $QR$  se denomina así porque es un método numérico basado en el cálculo de sucesivas factorizaciones  $QR$ . Si llamamos  $A_0 = A$  esta secuencia se define de la siguiente forma

$$A_{m-1} = Q_m R_m, \quad R_m Q_m = A_m,$$

donde  $Q_m$  es una matriz ortogonal y  $R_m$  es triangular superior con entradas positivas en su diagonal. Recordemos que cuando  $A$  tiene rango máximo estos factores están únicamente determinados. Además, esta secuencia converge a una matriz triangular superior con los valores propios de  $A$  en su diagonal.

Hay dos razones básicas que hacen que este algoritmo sea, por lo general, ineficiente. Primero, el coste de cada iteración  $QR$  es elevado, ya que cada descomposición  $QR$  tiene un coste de  $\frac{4}{3}n^3$  y la multiplicación posterior tiene también un coste de orden  $O(n^3)$ . Este coste es elevado dado que se deben realizar varias iteraciones. El segundo problema es, que generalmente, la convergencia es lenta, es decir, se necesita un número alto de iteraciones para que  $A_m$  esté cerca de ser la matriz triangular superior con los valores propios de  $A$  en su diagonal. Entonces, necesitamos reducir el coste de cada iteración  $QR$  y, además, nos interesa acelerar la convergencia del proceso.

Cuando la matriz  $A$  sobre la que estamos haciendo el algoritmo es Hessenberg superior, el coste de cada iteración se reduce notablemente a orden  $O(n^2)$ . La situación es más favorable si la matriz es simétrica y Hessenberg superior, es decir, es una matriz tridiagonal, en este caso, el orden es de  $O(n)$ . Además, el algoritmo  $QR$  preserva la forma de Hessenberg, por tanto, el coste total del algoritmo se ve reducido considerablemente.

Por otro lado, para mejorar la velocidad de convergencia, nos interesa buscar un valor  $\rho$  próximo a algún valor propio de  $A$ . Así, al aplicar el algoritmo  $QR$  sobre  $A - \rho I$ , la convergencia es mucho más rápida. A este valor  $\rho$  se le conoce como desplazamiento (shift en inglés).

Visto esto, el algoritmo que presentaremos para calcular los valores singulares en esta sección va a ser una adaptación del algoritmo  $QR$  con desplazamiento  $\rho$  a este problema. Nuestra adaptación es una implementación del algoritmo  $QR$  implícito sobre una matriz tridiagonal.

El algoritmo que vamos a presentar para calcular los valores y vectores singulares de una matriz  $A$  se divide en dos pasos. El primer paso consiste en calcular una matriz bidiagonal cuadrada asociada a  $A$ , que llamaremos  $B$ . Después, obtendremos la descomposición en valores singulares de  $B$ .

## 2.1. Matrices ortogonales

Las matrices ortogonales son la base del algoritmo que vamos a estudiar, ya que, pueden ser empleadas para generar ceros en las componentes deseadas de una matriz con un coste computacional reducido, y además, preservan la norma euclídea. Comenzamos definiendo las matrices de Householder.

**Definición 2.1.** *Dado un vector no nulo  $v$  en  $\mathbb{R}^n$ , llamamos matriz o reflector de Householder a:*

$$H = I - 2 \frac{vv^T}{v^T v} .$$

Esta matriz se encuentra en  $\mathbb{R}^{n \times n}$ , es decir, es una matriz cuadrada.

Una transformación de Householder es una transformación lineal cuya matriz asociada es un reflector de Householder. Estas transformaciones tienen una interpretación geométrica muy interesante. El conjunto  $\mathcal{H} = \{u \in \mathbb{R}^n \mid \langle v, u \rangle = 0\}$  es un hiperplano, es decir, un subespacio vectorial de  $\mathbb{R}^n$  de dimensión  $(n-1)$ . La matriz  $H$  lleva a cada vector  $x$  en  $\mathbb{R}^n$  a su simétrico respecto al hiperplano  $\mathcal{H}$ . Veamos algunas propiedades de este tipo de matrices.

**Proposición 2.2.** *Sea  $H$  un reflector de Householder y  $v$  su vector asociado, entonces  $H$  cumple las siguientes propiedades:*

1. *Simetría,  $H^T = H$ .*
2. *Ortogonalidad,  $H^T H = I$ .*
3.  *$Hv = -v$ .*
4.  *$Hu = u$ , si  $u$  es un vector en  $\mathbb{R}^n$  tal que  $\langle v, u \rangle = 0$ .*

*Demostración.*

1. Al considerar un producto de la forma,  $vv^T$ , obtenemos una matriz simétrica. Por tanto, su traspuesta es ella misma,  $(vv^T)^T = vv^T$ . Basándonos en esta propiedad, obtenemos que

$$H^T = \left( I - 2 \frac{vv^T}{v^T v} \right)^T = I^T - 2 \left( \frac{vv^T}{v^T v} \right)^T = I - 2 \frac{vv^T}{v^T v} = H .$$

2. Para demostrar esta propiedad tenemos que observar que  $v^T v$  es un escalar. Así, vemos que

$$\begin{aligned} H^T H &= HH = \left( I - 2 \frac{vv^T}{v^T v} \right) \left( I - 2 \frac{vv^T}{v^T v} \right) = I - 2 \frac{vv^T}{v^T v} - 2 \frac{vv^T}{v^T v} + 4 \frac{(vv^T)(vv^T)}{(v^T v)^2} = \\ &= I - 4 \frac{vv^T}{v^T v} + 4 \frac{(v^T v)vv^T}{(v^T v)^2} = I - 4 \frac{vv^T}{v^T v} + 4 \frac{vv^T}{v^T v} = I . \end{aligned}$$

3. Argumentando como antes

$$Hv = \left( I - 2 \frac{vv^T}{v^T v} \right) v = v - 2 \frac{(vv^T)v}{v^T v} = v - 2 \frac{(v^T v)v}{v^T v} = v - 2v = -v .$$

4. Tomamos  $u$  tal que  $\langle v, u \rangle = v^T u = 0$ . Entonces tenemos que

$$Hu = \left( I - 2 \frac{vv^T}{v^T v} \right) u = u - 2 \frac{(vv^T)u}{v^T v} = u - 2 \frac{v}{v^T v} \langle v, u \rangle = u .$$

□

Otra propiedad importante de los reflectores de Householder es la siguiente.

**Teorema 2.3.** *Sean  $x$  e  $y$  vectores distintos en  $\mathbb{R}^n$  tales que  $\|x\|_2 = \|y\|_2$ . Entonces existe un reflector de Householder  $H$  tal que  $Hx = y$ .*

*Demostración.* Probar la existencia del reflector  $H$  del enunciado equivale a encontrar el vector  $v$  tal que

$$\left( I - 2 \frac{vv^T}{v^Tv} \right) x = y.$$

Supongamos  $v = x - y$ . Primero descomponemos  $x$  de la siguiente forma

$$x = \frac{1}{2}(x - y) + \frac{1}{2}(x + y),$$

y consideramos el producto

$$Hx = H \left( \frac{1}{2}(x - y) + \frac{1}{2}(x + y) \right) = \frac{1}{2}H(x - y) + \frac{1}{2}H(x + y).$$

Utilizamos las propiedades (3) y (4) de la proposición anterior y obtenemos que  $H(x - y) = y - x$ , ya que  $v = x - y$  y que  $H(x + y) = x + y$ , porque se cumple que

$$\langle v, x + y \rangle = \langle x - y, x + y \rangle = \langle x, x \rangle + \langle x, y \rangle - \langle y, x \rangle - \langle y, y \rangle = \|x\|_2^2 - \|y\|_2^2 = 0.$$

Para finalizar juntamos todo lo visto y observamos que

$$Hx = \frac{1}{2}H(x - y) + \frac{1}{2}H(x + y) = \frac{1}{2}(y - x) + \frac{1}{2}(x + y) = y.$$

Por tanto, tomando  $v = x - y$  se satisface la propiedad  $Hx = y$ , como queríamos demostrar.  $\square$

De este resultado podemos ver que los reflectores pueden usarse para hacer ceros en vectores y matrices.

**Corolario 2.4.** *Dado un vector no nulo  $x$  en  $\mathbb{R}^n$ . Entonces existe un reflector de Householder  $H$  tal que*

$$H \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} * \\ 0 \\ \vdots \\ 0 \end{bmatrix}.$$

*Demostración.* Tomamos  $y = [-\tau \ 0 \ \cdots \ 0]^T$  con  $\tau = \pm\|x\|_2$ . Si elegimos correctamente el signo podemos asegurar que  $x \neq y$  y, claramente,  $\|x\|_2 = \|y\|_2$ . Por lo que, utilizando el Teorema 2.3, existe un reflector  $H$  tal que  $Hx = y$ .  $\square$

Basándonos en estos resultados y sus demostraciones, podemos construir un reflector  $H$  que lleva cualquier vector  $x$  a otro cuya primera componente es distinta de cero y las demás son nulas. El reflector  $H$  es de la forma  $I - 2 \frac{vv^T}{v^Tv}$ , donde  $v$  tiene la estructura

$$v = x - y = \begin{bmatrix} x_1 + \tau \\ x_2 \\ \vdots \\ x_n \end{bmatrix}.$$

Cualquier múltiplo de  $v$  genera el mismo reflector, por lo que, generalmente, se normaliza  $v$  de manera que la primera entrada sea un 1. El vector resultante es

$$v = \frac{x-y}{x_1+\tau} = \begin{bmatrix} 1 \\ x_2/(x_1+\tau) \\ \vdots \\ x_n/(x_1+\tau) \end{bmatrix}.$$

Recordemos que la cancelación de  $x_1 + \tau$  no puede suceder, ya que hemos tomado el signo de  $\tau$  correctamente para que esto no ocurra.

Otro tipo de matriz ortogonal que vamos a utilizar en el algoritmo  $QR$  son los rotadores de Givens.

**Definición 2.5.** *Los rotadores de Givens son matrices cuadradas ortogonales que tienen la siguiente forma*

$$\begin{bmatrix} \text{col } i & \text{col } j \\ \begin{matrix} 1 & \cdots & 0 & \cdots & 0 & \cdots & 0 \\ \vdots & \ddots & \vdots & & \vdots & & \vdots \\ 0 & \cdots & c & \cdots & -s & \cdots & 0 \\ \vdots & & \vdots & \ddots & \vdots & & \vdots \\ 0 & \cdots & s & \cdots & c & \cdots & 0 \\ \vdots & & \vdots & & \vdots & \ddots & \vdots \\ 0 & \cdots & 0 & \cdots & 0 & \cdots & 1 \end{matrix} & \begin{matrix} \text{fila } i \\ \text{fila } j \end{matrix} \end{bmatrix}$$

con  $c = \cos \theta$  y  $s = \sin \theta$ .

Los rotadores de Givens se pueden interpretar geométricamente como rotaciones planas en el espacio vectorial, es decir, son rotaciones de dos dimensiones en un espacio mayor. Algunas características de estos rotadores son las siguientes.

**Proposición 2.6.** *Llamemos  $Q$  a un rotador de Givens con dimensión  $m \times m$ .*

1. *Sea  $A$  una matriz en  $\mathbb{R}^{m \times n}$ . Al aplicar  $Q$  sobre  $A$ ,  $QA$ , solo se alteran las filas  $i$  y  $j$  de  $A$ . Además, las filas  $i$  y  $j$  de  $QA$  son combinaciones lineales de las filas  $i$  y  $j$  de  $A$ .*
2. *Sea  $B$  una matriz en  $\mathbb{R}^{n \times m}$ . Al aplicar  $B$  sobre  $Q$ ,  $BQ$ , se alteran únicamente las columnas  $i$  y  $j$  de  $B$ . De manera similar a antes, las columnas  $i$  y  $j$  de  $BQ$  son combinaciones lineales de las columnas  $i, j$  de  $B$ .*
3. *Dada una matriz ortogonal, ésta se puede descomponer como producto de rotadores de Givens.*

Otra utilidad muy importante de los rotadores es crear ceros en matrices o vectores.

**Proposición 2.7.** *Dada una matriz  $A$  en  $\mathbb{R}^{m \times n}$  de la cual queremos anular la componente  $a_{ij}$ , existe un rotador de Givens  $Q$  tal que, al aplicar su traspuesta sobre la matriz  $A$ ,  $Q^T A$ , anula esa componente.*

*Demostración.* Asumimos que  $i > j$ . Tomamos la componente  $a_{jj}$  y calculamos

$$c = \frac{a_{jj}}{\sqrt{a_{jj}^2 + a_{ij}^2}} \quad \text{y} \quad s = \frac{a_{ij}}{\sqrt{a_{jj}^2 + a_{ij}^2}},$$

con estos valores formo el rotador  $Q$ , de tal manera que al hacer el producto  $Q^T A$  la componente  $(i, j)$  resultaría

$$\tilde{a}_{ij} = -sa_{jj} + ca_{ij} = -\frac{a_{ij}}{\sqrt{a_{jj}^2 + a_{ij}^2}}a_{jj} + \frac{a_{jj}}{\sqrt{a_{jj}^2 + a_{ij}^2}}a_{ij} = 0.$$

□

Un resultado fundamental para desarrollar el algoritmo  $QR$  implícito es el resultado conocido como el teorema de la  $Q$ -implícita, que obtenemos de la página 382 de [8].

**Teorema 2.8.** (Teorema de la  $Q$ -implícita) Sea  $A, \hat{A}, \tilde{A}, \hat{Q}$  y  $\tilde{Q}$  matrices en  $\mathbb{C}^{n \times n}$ . Con  $\hat{A}$  propiamente Hessenberg superior,  $\tilde{A}$  Hessenberg superior y  $\hat{Q}$  y  $\tilde{Q}$  unitarias, tales que

$$\hat{A} = \hat{Q}^{-1} A \hat{Q} \quad y \quad \tilde{A} = \tilde{Q}^{-1} A \tilde{Q}.$$

Además, la primera columna de  $\hat{Q}$  y la primera columna de  $\tilde{Q}$  son proporcionales, es decir,  $\hat{q}_1 = \tilde{q}_1 d_1$ , donde  $|d_1| = 1$ . Entonces,  $\tilde{A}$  es también propiamente Hessenberg superior y existe una matriz diagonal unitaria  $D$  tal que

$$\hat{Q} = \tilde{Q} D \quad y \quad \hat{A} = D^{-1} \tilde{A} D.$$

Ahora vamos a empezar a describir el algoritmo de cálculo de la descomposición en valores singulares. Recordemos que el primer paso consiste en reducir la matriz original a forma bidiagonal.

## 2.2. Reducir $A$ a forma bidiagonal

Definimos como matriz bidiagonal a una matriz en  $\mathbb{R}^{m \times n}$  tal que  $b_{i,j} = 0$  cuando  $i > j$  o  $i < j - 1$ . Luego es de la forma:

$$\begin{bmatrix} b_{1,1} & b_{1,2} & & & & \\ & b_{2,2} & b_{2,3} & & & \\ & & \ddots & \ddots & & \\ & & & b_{n-1,n-1} & b_{n-1,n} & \\ & & & & b_{n,n} & \end{bmatrix}.$$

Llamamos matriz propiamente bidiagonal a la matriz bidiagonal cuyas entradas en la diagonal y en la diagonal superior, es decir, los  $b_{i,j}$  con  $j-1 \leq i \leq j$ , son no nulas.

Gracias a las matrices de Householder podemos relacionar una matriz cualquiera con una matriz bidiagonal. Para ello, nos apoyaremos en el siguiente teorema.

**Teorema 2.9.** Sea  $A$  una matriz en  $\mathbb{R}^{m \times n}$ . Entonces existen matrices ortogonales  $\hat{U}$  en  $\mathbb{R}^{m \times m}$  y  $\hat{V}$  en  $\mathbb{R}^{n \times n}$ , ambas producto de un número finito de matrices de Householder, y una matriz bidiagonal  $\hat{B}$  en  $\mathbb{R}^{m \times n}$  tales que

$$A = \hat{U} \hat{B} \hat{V}^T. \quad (2.1)$$

Además, existe un algoritmo finito para calcular  $\hat{U}, \hat{V}$  y  $\hat{B}$ .

*Demostración.* Vamos a demostrar este enunciado de forma constructiva. Veremos el desarrollo para llegar a la expresión (2.1).

El primer paso consiste en hacer ceros en la primera fila y columna de  $A$ . Para ello, multiplicamos a izquierda por la matriz de Householder  $\hat{U}_1$ , de tamaño  $m \times m$ , tal que,

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

es decir, al multiplicar  $\hat{U}_1 A$  resulta una matriz con la primera columna formada por ceros excepto la posición (1,1):

$$\hat{U}_1 A = \begin{bmatrix} \hat{a}_{11} & \hat{a}_{12} & \cdots & \hat{a}_{1n} \\ 0 & & & \\ \vdots & & (\hat{U}_1 A)_1 & \\ 0 & & & \end{bmatrix}.$$

Ahora, queremos hacer los ceros en la primera fila. Para ello, tomamos la matriz de Householder  $\hat{V}_1$  de tamaño  $n \times n$  con la siguiente forma:

$$\hat{V}_1 = \begin{bmatrix} 1 & 0 & \cdots & 0 \\ 0 & & & \\ \vdots & & \tilde{V}_1 & \\ 0 & & & \end{bmatrix},$$

tal que cumpla lo siguiente

$$[\hat{a}_{12} \ \cdots \ \hat{a}_{1n}] \tilde{V}_1 = [\tilde{a}_{12} \ 0 \ \cdots \ 0].$$

Por tanto, como  $\tilde{V}_1$  tiene el primer vector de la base canónica  $e_1$  como su primera columna, hemos conseguido lo que buscábamos. Entonces, al hacer el producto  $\hat{U}_1 A \hat{V}_1$  la primera columna de  $\hat{U}_1 A$  no varía y obtenemos una matriz de la forma:

$$\hat{U}_1 A \hat{V}_1 = \begin{bmatrix} \hat{a}_{11} & \tilde{a}_{12} & 0 & \cdots & 0 \\ 0 & & & & \\ \vdots & & A_1 & & \\ 0 & & & & \end{bmatrix}.$$

El segundo paso consiste en hacer ceros en la segunda fila y columna. Vamos a proceder de la misma manera, pero esta vez sobre la submatriz  $A_1$ . Además, utilizando un razonamiento similar al que hemos visto con el vector canónico  $e_1$ , es fácil ver que los ceros de las posiciones que nos interesan no se anulan. Después de este paso obtendremos una matriz de la forma:

$$\hat{U}_2 \hat{U}_1 A \hat{V}_1 \hat{V}_2 = \begin{bmatrix} * & * & 0 & 0 & \cdots & 0 \\ 0 & * & * & 0 & \cdots & 0 \\ 0 & 0 & & & & \\ \vdots & \vdots & & A_2 & & \\ 0 & 0 & & & & \end{bmatrix}.$$

Actuando por inducción sobre la submatriz  $A_k$ , después de  $n$  pasos, la matriz resultante será de la forma:

$$\hat{U}_n \dots \hat{U}_2 \hat{U}_1 A \hat{V}_1 \hat{V}_2 \dots \hat{V}_{n-2} = \begin{bmatrix} * & * & & & & \\ * & * & & & & \\ \ddots & \ddots & & & & \\ & & * & * & & \\ & & & * & & \end{bmatrix} = \hat{B}.$$

Notemos que en los últimos dos pasos, el paso  $n$  y el paso  $n-1$ , solo necesitamos hacer el producto a izquierda.

Para acabar la demostración, llamamos  $\hat{U} = \hat{U}_1 \hat{U}_2 \dots \hat{U}_n$  y  $\hat{V} = \hat{V}_1 \hat{V}_2 \dots \hat{V}_{n-2}$ , que ya no son reflectores de Householder pero sí siguen siendo matrices ortogonales. Por tanto, hemos llegado a que

$$\hat{U}^T A \hat{V} = \hat{B}, \quad \text{es decir,} \quad A = \hat{U} \hat{B} \hat{V}^T,$$

como queríamos demostrar. □

Esta es una forma de reducir  $A$  a forma bidiagonal, pero en la práctica, a menudo  $m$  es considerablemente mayor que  $n$ , por ejemplo en el problema de mínimos cuadrados, donde podemos tener muchas

observaciones de un suceso. En estos casos, a nivel de eficiencia, es mejor calcular la matriz bidiagonal relacionada con  $A$  en dos etapas. La primera etapa consiste en calcular la factorización  $QR$  de  $A$ ,

$$A = [Q_1 \ Q_2] \begin{bmatrix} R \\ 0 \end{bmatrix}, \quad (2.2)$$

donde  $Q$  es una matriz ortogonal en  $\mathbb{R}^{m \times m}$ , dividida en  $Q_1$  una matriz formada por las primeras  $n$  columnas de  $Q$  y  $Q_2$  formada por las últimas  $m - n$  columnas de  $Q$ , y  $R$  es una matriz triangular superior cuadrada en  $\mathbb{R}^{n \times n}$ . Esta factorización reduce el número de operaciones del proceso de bidiagonalización, ya que solo tenemos que multiplicar por matrices de Householder a izquierda de  $A$ .

La segunda fase consiste en buscar la forma bidiagonal de  $R$ . Para ello, utilizamos el teorema anterior y obtenemos la factorización  $R = \tilde{U}B\tilde{V}^T$ , donde todas las matrices involucradas son  $n \times n$ . El Algoritmo 1 explica mediante pseudocódigo cómo reducir una matriz triangular superior cuadrada a forma bidiagonal paso a paso. Luego, recopilando todo lo visto hasta ahora, tenemos que

$$A = [Q_1 \ Q_2] \begin{bmatrix} \tilde{U} & 0 \\ 0 & I \end{bmatrix} \begin{bmatrix} B \\ 0 \end{bmatrix} \tilde{V}^T.$$

Si introducimos la notación

$$\hat{U} = [Q_1 \ Q_2] \begin{bmatrix} \tilde{U} & 0 \\ 0 & I \end{bmatrix} = [Q_1 \tilde{U} \ Q_2], \quad \hat{B} = \begin{bmatrix} B \\ 0 \end{bmatrix} \quad \text{y} \quad \hat{V} = \tilde{V},$$

tenemos lo que queríamos,  $A = \hat{U}\hat{B}\hat{V}^T$  como en (2.1).

Como hemos mencionado antes, realizar este proceso cuando  $m$  es significativamente mayor que  $n$  reduce el coste computacional notablemente. Esto es debido a que en la primera fase las multiplicaciones por matrices de Householder se ejecutan únicamente a izquierda. Además en la segunda fase estas multiplicaciones se ejecutan sobre la matriz reducida  $R$ , que tiene dimensión  $n \times n$  y no sobre la matriz  $A$  cuya dimensión es  $m \times n$ .

No obstante, este proceso supone una desventaja en algunos casos, ya que las multiplicaciones a derecha de la segunda fase deshacen los ceros de la matriz triangular superior, y, por tanto, hay que añadir una multiplicación a izquierda aumentando así el coste. Este aumento de coste es favorable cuando  $m$  es considerablemente mayor que  $n$ , ya que el coste adicional de las multiplicaciones a izquierda se compensará con creces por el ahorro en las multiplicaciones a derecha. En concreto, según la página 404 de [8], el cociente  $m/n$  debe ser mayor que  $5/3$  para que este proceso sea la mejor opción.

### 2.3. Calcular los valores singulares de $B$

Volviendo al problema del cálculo de los valores singulares, hemos calculado la matriz bidiagonal  $\hat{B}$  asociada a una matriz cualquiera  $A$ , de dimensión  $m \times n$ . Recordemos que  $\hat{B}$  tiene la forma

$$\hat{B} = \begin{bmatrix} B \\ 0 \end{bmatrix},$$

donde  $B$  es una matriz bidiagonal cuadrada en  $\mathbb{R}^{n \times n}$  de la forma,

$$B = \begin{bmatrix} \beta_1 & \gamma_1 & & & \\ & \beta_2 & \gamma_2 & & \\ & & \ddots & & \\ & & & \ddots & \gamma_{n-1} \\ & & & & \beta_n \end{bmatrix}. \quad (2.3)$$

Comenzamos ahora el segundo paso para obtener la descomposición en valores singulares de  $A$ . Dada la relación entre  $A$  y  $\hat{B}$  (2.1), calcular la descomposición en valores singulares de  $A$  se reduce a obtener

la descomposición en valores singulares de  $B$ . Antes de empezar a desarrollar el cálculo de los valores singulares, vamos a asumir sin pérdida de generalidad que  $B$  es propiamente bidiagonal y que sus entradas no nulas son positivas. El Comentario 1 justifica que podemos asumir que  $B$  es propiamente bidiagonal.

**Comentario 1.** *Podemos asumir que  $B$  es propiamente bidiagonal por estos motivos. Por un lado, si algún  $\gamma_k$ , con  $1 \leq k \leq n-1$ , es igual a cero, podemos dividir la matriz  $B$  en dos submatrices bidiagonales de la forma:*

$$B = \begin{bmatrix} B_1 & 0 \\ 0 & B_2 \end{bmatrix},$$

*con  $B_1$  en  $\mathbb{R}^{k \times k}$  y  $B_2$  en  $\mathbb{R}^{(n-k) \times (n-k)}$  y aplicar este algoritmo a las dos submatrices  $B_1$  y  $B_2$ . Por otro lado, si algún  $\beta_k$ , con  $1 \leq k \leq n$ , es igual a cero, podemos realizar operaciones elementales sobre las filas y las columnas para conseguir una matriz propiamente bidiagonal.*

También nos interesa asumir que las entradas no nulas de  $B$  son positivas.

**Comentario 2.** *Podemos asumir que todas las entradas no nulas, es decir,  $\beta_i$ , con  $1 \leq i \leq n$ , y  $\gamma_j$ , con  $1 \leq j \leq n-1$ , son positivas. Esto se debe a que, si no lo fuera, siempre existen dos matrices ortogonales y diagonales  $D_1$  y  $D_2$  las cuales hacen que  $D_1BD_2$  sea una matriz bidiagonal cuyas entradas no nulas son positivas.*

Visto todo esto, podemos afirmar que  $B$  es una matriz bidiagonal cuyas entradas no nulas son positivas sin pérdida de generalidad.

Como hemos visto en la demostración del Teorema 1.7, para calcular la descomposición en valores singulares de  $B$ , primero tenemos que calcular los valores propios de las matrices  $B^T B$  y  $BB^T$ . Para ello, hay varios algoritmos que podemos emplear. En este caso vamos a utilizar el algoritmo  $QR$  sin realizar explícitamente el producto de  $B^T B$  o  $BB^T$ . Antes de empezar con el desarrollo del algoritmo necesitamos presentar algunas definiciones y resultados relacionados.

**Definición 2.10.** *Una matriz tridiagonal cuadrada es una matriz cuyas entradas  $b_{i,j}$  tales que  $|i-j| > 1$ , son iguales a 0.*

$$\begin{bmatrix} b_{1,1} & b_{1,2} & & & \\ b_{2,1} & b_{2,2} & b_{2,3} & & \\ & b_{3,2} & b_{3,3} & \ddots & \\ & \ddots & \ddots & \ddots & b_{n-1} \\ & & b_{n-1,n-1} & b_{n,n} & \end{bmatrix}$$

*Una matriz es propiamente tridiagonal cuando las entradas de la diagonal principal y la diagonal superior e inferior a ésta son no nulas.*

Para nuestro resultado final vamos a utilizar la siguiente proposición.

**Proposición 2.11.** *Si  $B$  es una matriz propiamente bidiagonal, entonces  $B^T B$  y  $BB^T$  son matrices propiamente tridiagonales.*

*Demostración.* Tomamos  $B$  matriz propiamente bidiagonal, es decir, de la forma (2.3), y realizamos el producto  $B^T B$ .  $BB^T$  será análogo.

$$B^T B = \begin{bmatrix} \beta_1 & & & & \\ \gamma_1 & \beta_2 & & & \\ & \gamma_2 & \beta_3 & & \\ & & \ddots & \ddots & \\ & & & \ddots & \ddots \\ & & & & \gamma_{n-1} & \beta_n \end{bmatrix} \begin{bmatrix} \beta_1 & \gamma_1 & & & \\ & \beta_2 & \gamma_2 & & \\ & & \ddots & & \\ & & & \beta_3 & \ddots \\ & & & & \ddots & \gamma_{n-1} \\ & & & & & \beta_n \end{bmatrix} = \begin{bmatrix} \beta_1^2 & \beta_1 \gamma_1 & & & \\ \gamma_1 \beta_1 & \gamma_1^2 + \beta_2^2 & \beta_2 \gamma_2 & & \\ & \gamma_2 \beta_2 & \gamma_2^2 + \beta_3^2 & \ddots & \\ & & \ddots & \ddots & \ddots & \beta_{n-1} \gamma_{n-1} \\ & & & & & \gamma_{n-1} \beta_{n-1} & \gamma_{n-1}^2 + \beta_n^2 \end{bmatrix}.$$

□

Hemos mencionado anteriormente que para calcular los valores propios de  $B^T B$  y  $BB^T$  vamos a utilizar el algoritmo  $QR$  sin realizar explícitamente los productos  $B^T B$  y  $BB^T$ . En particular, vamos a considerar el caso más general con desplazamiento  $\rho$ , que puede servir para mejorar la velocidad de convergencia del método. Primero vamos a ver cómo sería una iteración de este algoritmo si hiciéramos los productos:

$$B^T B - \rho I = \hat{Q} R \quad \text{and} \quad BB^T - \rho I = \hat{P} S, \quad (2.4)$$

con  $\hat{Q}$  y  $\hat{P}$  matrices ortogonales, mientras que,  $R$  y  $S$  matrices triangulares superiores. Asumimos que  $\rho$  no es un valor propio de  $B^T B$  y  $BB^T$ , así las matrices  $R$  y  $S$  son invertibles. Además, podemos normalizar  $R$  y  $S$  de tal forma que las entradas de las diagonales principales sean todas positivas.

Vamos a llamar  $\hat{B}$  a

$$\hat{B} = \hat{P}^T B \hat{Q}. \quad (2.5)$$

De aquí deducimos que

$$\begin{aligned} \hat{B}^T \hat{B} &= (\hat{P}^T B \hat{Q})^T (\hat{P}^T B \hat{Q}) = \hat{Q}^T B^T \hat{P} \hat{P}^T B \hat{Q} = \hat{Q}^T B^T B \hat{Q} = \\ &= \hat{Q}^T (\hat{Q} R + \rho I) \hat{Q} = \hat{Q}^T \hat{Q} R \hat{Q} + \rho \hat{Q}^T \hat{Q} = R \hat{Q} + \rho I, \\ \hat{B} \hat{B}^T &= (\hat{P}^T B \hat{Q}) (\hat{P}^T B \hat{Q})^T = \hat{P}^T B \hat{Q} \hat{Q}^T B^T \hat{P} = \hat{P}^T B B^T \hat{P} = \\ &= \hat{P}^T (\hat{P} S + \rho I) \hat{P} = \hat{P}^T \hat{P} S \hat{P} + \rho \hat{P}^T \hat{P} = S \hat{P} + \rho I. \end{aligned}$$

Esto nos indica que  $\hat{B}^T \hat{B}$  y  $\hat{B} \hat{B}^T$  son iteraciones del algoritmo  $QR$  con desplazamiento  $\rho$  sobre  $B^T B$  y  $BB^T$ . Para calcular  $\hat{B}$  tenemos que obtener las matrices  $\hat{P}$  y  $\hat{Q}$  que se definen en (2.4). Sabemos que estas matrices se pueden escribir como productos de rotadores de Givens, ya que son ortogonales, y tienen la siguiente forma

$$\hat{P} = P_1 \cdots P_{n-1} \quad \text{y} \quad \hat{Q} = Q_1 \cdots Q_{n-1}.$$

Si obtenemos estos rotadores de Givens, los podemos aplicar de la forma adecuada a  $B$  y la transformaremos en  $\hat{B}$ . El objetivo es encontrarlos de manera que el coste del algoritmo sea lo menor posible, evitando calcular las matrices  $B^T B$  y  $BB^T$  y la factorización  $QR$  explícita.

Primero, veamos que  $\hat{B}$  hereda la forma de matriz propiamente bidiagonal de  $B$ .

**Proposición 2.12.** *Dada  $\hat{B}$  definida por (2.5) y  $R$  y  $S$  invertibles, si  $B$  es bidiagonal, entonces  $\hat{B}$  también lo es.*

*Demostración.* Probamos antes unas expresiones auxiliares que vamos a emplear.

$$\begin{aligned} (BB^T - \rho I) B &= BB^T B - \rho B = B(B^T B - \rho I), \\ (B^T B - \rho I) B^T &= B^T BB^T - \rho B^T = B^T(BB^T - \rho I). \end{aligned}$$

Por otro lado de (2.4) deducimos que

$$(B^T B - \rho I) R^{-1} = \hat{Q} \quad \text{y} \quad (BB^T - \rho I) S^{-1} = \hat{P}.$$

Ahora, podemos trabajar con  $\hat{B}$  de otra forma:

$$\hat{B} = \hat{P}^T B \hat{Q} = \hat{P}^T B (B^T B - \rho I) R^{-1} = \hat{P}^T (BB^T - \rho I) B R^{-1} = \hat{P}^T \hat{P} S B R^{-1} = S B R^{-1}, \quad (2.6)$$

$$\hat{B}^T = (\hat{P}^T B \hat{Q})^T = \hat{Q}^T B^T \hat{P} = \hat{Q}^T B^T (BB^T - \rho I) S^{-1} = \hat{Q}^T (B^T B - \rho I) B^T S^{-1} = \hat{Q}^T \hat{Q} R B^T S^{-1} = R B^T S^{-1}.$$

El primer paso es probar que, si  $B$  es triangular superior,  $\hat{B}$  también lo es. Es fácil de ver, ya que el producto de matrices triangulares superiores es una matriz triangular superior. Por hipótesis,  $S$ ,  $B$  y  $R$  son de esta forma. También sabemos que la inversa de una matriz triangular superior también lo es, por tanto  $R^{-1}$  también es triangular superior. Todas las matrices involucradas en el producto son triangulares superiores, por ello,  $\hat{B}$  también lo es.

También sabemos que al multiplicar una matriz triangular superior por una Hessenberg superior, el resultado es Hessenberg superior, luego, si  $B$  es Hessenberg,  $\hat{B}$  también lo es.

Para finalizar, recordamos que si  $B$  es bidiagonal,  $BB^T$  y  $B^TB$  son tridiagonales. Queremos llegar a que  $\hat{B}\hat{B}^T$  es tridiagonal, por tanto,  $\hat{B}$  será bidiagonal. Si  $B$  es bidiagonal, por un lado,

$$\hat{B}\hat{B}^T = SBR^{-1}RB^TS^{-1} = SBB^TS^{-1},$$

como  $BB^T$  es tridiagonal, es decir, Hessenberg,  $\hat{B}\hat{B}^T$  también es Hessenberg. Por otro lado,

$$(\hat{B}\hat{B}^T)^T = \hat{B}^T\hat{B} = RB^TS^{-1}SBR^{-1} = RB^TBR^{-1},$$

de nuevo, como  $B^TB$  es tridiagonal también es Hessenberg, por tanto  $\hat{B}^T\hat{B}$  es Hessenberg. Dado que  $\hat{B}\hat{B}^T$  y  $(\hat{B}\hat{B}^T)^T$  son ambas Hessenberg, estamos ante una matriz tridiagonal. Por tanto,  $\hat{B}$  es bidiagonal.  $\square$

**Corolario 2.13.** *Bajo la misma hipótesis que antes, si  $B$  es propiamente bidiagonal,  $\hat{B}$  también lo es.*

**Corolario 2.14.** *Cuando  $R$  y  $S$  están normalizadas de forma que las entradas de su diagonal principal sean positivas, entonces si las entradas no nulas de  $B$  son todas positivas, entonces las entradas no nulas de  $\hat{B}$  también lo son.*

A continuación, comenzamos el desarrollo del algoritmo  $QR$  implícito con desplazamiento  $\rho$ , esto es, buscamos la descomposición  $B^TB - \rho I = \hat{Q}\hat{R}$ . Sin necesidad de hacer el producto, observamos que, como  $R$  es triangular superior, la primera columna de  $\hat{Q}$  es proporcional a

$$\begin{bmatrix} \beta_1^2 - \rho \\ \gamma_1 \beta_1 \\ 0 \\ \vdots \\ 0 \end{bmatrix}. \quad (2.7)$$

Ahora utilizaremos rotadores de Givens para hacer ceros. Llamemos rotador de Givens sobre el plano  $[1, 2]$  a  $V_1$ , cuya primera columna es proporcional a (2.7) y hacemos  $BV_1$ , este producto altera únicamente las primeras dos columnas de  $B$  y crea una componente distinta de cero en la posición  $(2, 1)$ .

Para volver a la forma bidiagonal, buscamos un rotador  $U_1$  sobre el plano  $[1, 2]$  tal que al hacer  $U_1^T BV_1$  genere un cero en la posición  $(2, 1)$ . Este paso genera una nueva componente no nula en  $(1, 3)$ , la cual queremos anular mediante el rotador  $V_2$  sobre el plano  $[2, 3]$ . Ejecutamos  $U_1^T BV_1 V_2$ , esto anula la componente deseada  $(1, 3)$  y crea una nueva entrada en  $(3, 2)$ . Así, necesitamos aplicar sucesivamente rotadores  $U_2^T, V_3, U_3^T, \dots$ . De esta manera vamos anulando las componentes fuera de la bidiagonal y generando nuevas entradas no nulas hasta llegar a una matriz de forma bidiagonal:

$$\tilde{B} = U_{n-1}^T \cdots U_2^T U_1^T B V_1 V_2 \cdots V_{n-1}.$$

Si llamamos

$$\tilde{P} = U_1 U_2 \cdots U_{n-1} \quad \text{y} \quad \tilde{Q} = V_1 V_2 \cdots V_{n-1},$$

obtenemos que

$$\tilde{B} = \tilde{P}^T B \tilde{Q}. \quad (2.8)$$

En el algoritmo 2, presentado en el anexo de pseudocódigos, mostramos paso a paso cómo obtener la expresión (2.8).

Por tanto, tenemos que probar ahora que la matriz  $\hat{B}$  de (2.5) y la matriz  $\tilde{B}$  de (2.8) son “esencialmente” la misma matriz.

**Teorema 2.15.** *Dada  $B$  matriz invertible. Sea  $\tilde{B}$  matriz bidiagonal,  $\hat{B}$  matriz propiamente bidiagonal y  $\hat{P}, \hat{Q}, \tilde{P}, \tilde{Q}$  matrices ortogonales tales que*

$$\hat{B} = \hat{P}^T B \hat{Q} \quad y \quad \tilde{B} = \tilde{P}^T B \tilde{Q}.$$

*Supongamos además que  $\hat{Q}$  y  $\tilde{Q}$  tienen esencialmente la misma primera columna, es decir,  $\hat{Q}e_1 = \tilde{Q}e_1 d_1$ , con  $d_1 = \pm 1$ . Entonces, existen matrices ortogonales diagonales  $D$  y  $E$  tales que  $\hat{Q} = \tilde{Q}D$  y  $\hat{P} = \tilde{P}E$  y*

$$\hat{B} = E \tilde{B} D.$$

*Dicho de otra forma,  $\hat{B}$  y  $\tilde{B}$  son esencialmente iguales.*

*Demostración.* Notemos que

$$\hat{B}^T \hat{B} = (\hat{P}^T B \hat{Q})^T (\hat{P}^T B \hat{Q}) = \hat{Q}^T B^T \hat{P} \hat{P}^T B \hat{Q} = \hat{Q}^T (B^T B) \hat{Q} \quad y \quad \tilde{B}^T \tilde{B} = \tilde{Q}^T (B^T B) \tilde{Q}.$$

También sabemos que  $\tilde{B}^T \tilde{B}$  es una matriz tridiagonal y  $\hat{B}^T \hat{B}$  es propiamente tridiagonal. Con todo esto, podemos aplicar el teorema de la Q-implícita (Teorema 2.8). Llamamos  $A$  a  $B^T B$ ,  $\tilde{A}$  a  $\tilde{B}^T \tilde{B}$  y  $\hat{A}$  a  $\hat{B}^T \hat{B}$  y nos da como resultado la existencia de una matriz diagonal ortogonal  $D$  tal que  $\hat{Q} = \tilde{Q}D$ , es decir,  $\hat{Q}$  y  $\tilde{Q}$  son esencialmente iguales. Fijándonos en la definición de  $\hat{B}$  y  $\tilde{B}$  observamos que

$$\hat{P} \hat{B} = B \hat{Q} = B \tilde{Q} D = \tilde{P} \tilde{B} D.$$

Llamamos  $C = \hat{P} \hat{B} = \tilde{P}(\tilde{B} D)$  y observamos que ambas son descomposiciones QR de  $C$ , luego,  $\hat{P}$  y  $\tilde{P}$  son esencialmente iguales. Existe una matriz ortogonal y diagonal  $E$  tal que  $\hat{P} = \tilde{P}E$ . Juntando lo anterior llegamos a

$$\hat{B} = \hat{P}^T B \hat{Q} = (\tilde{P}E)^T B \tilde{Q} D = E^T \tilde{P}^T B \tilde{Q} D = E \tilde{B} D.$$

□

Por tanto, nuestro caso concreto queda reflejado en el siguiente corolario del teorema anterior, que utiliza la misma notación empleada anteriormente.

**Corolario 2.16.** *Si  $\hat{B}$  y  $\tilde{B}$  son propiamente bidiagonales y  $\hat{Q}e_1 = \tilde{Q}e_1$ , entonces  $\hat{Q} = \tilde{Q}$ ,  $\hat{P} = \tilde{P}$  y  $\hat{B} = \tilde{B}$ .*

Si podemos aplicar este teorema sobre nuestras matrices  $\hat{B}$  y  $\tilde{B}$ , podremos completar la iteración QR sin realizar los productos  $B^T B$  y  $BB^T$ . Para ello, tenemos que ver cómo son las primeras columnas de  $\hat{Q}$  y  $\tilde{Q}$ .

Según la notación que hemos seleccionado  $\tilde{Q} = V_1 V_2 \cdots V_{n-1}$ . Su primera columna es la primera de  $V_1$ , ya que el resto son rotadores de Givens que actúan sobre el resto de columnas. La primera columna de  $V_1$  ha sido elegida de forma que es igual a la primera columna de  $\hat{Q}$ . Es decir,  $\tilde{Q}e_1 = \hat{Q}e_1$ , luego podemos aplicar el Teorema 2.15 y deducimos que  $\hat{B}$  y  $\tilde{B}$  son esencialmente iguales.

Así hemos visto cómo calcular una iteración del algoritmo QR sin realizar el producto  $B^T B$  o  $BB^T$  puesto que la matriz  $\hat{B}$  nos sirve para definir una iteración del algoritmo QR con desplazamiento  $\rho$ . Continuando esta estrategia, podemos obtener la descomposición en valores singulares de  $B$  y gracias a la fórmula (2.1) del Teorema 2.9 obtenemos la descomposición en valores singulares de  $A$ .

El objetivo principal de este capítulo ha sido justificar paso a paso una forma de calcular los valores singulares de una matriz real cualquiera. Juntando todo lo estudiado, el procedimiento para calcular los valores singulares de  $A$  consiste en reducir la matriz original  $A$  a forma bidiagonal utilizando multiplicaciones a izquierda y a derecha por matrices de Householder. Como hemos observado, cuando  $m$  es considerablemente mayor que  $n$ , el coste computacional se ve reducido si, previamente, ejecutamos una descomposición QR de  $A$  (2.2) y, a continuación, reducimos  $R$  a forma bidiagonal. Una forma de ejecutar esta reducción es aplicar el Algoritmo 1. Una vez hemos calculado la matriz bidiagonal  $B$ , tenemos que calcular sus valores singulares y éstos se obtienen calculando los valores propios de  $B^T B$  y  $BB^T$ . Para ello, empleamos el algoritmo QR implícito sin realizar explícitamente los productos  $B^T B$  y  $BB^T$  con desplazamiento  $\rho$ . En esta sección, hemos estudiado cómo se haría una iteración justificando todos los pasos, y en Algoritmo 2 vemos claramente cómo realizar una iteración. Una vez hemos realizado todas las iteraciones y tenemos los valores propios de  $B^T B$  y  $BB^T$ , basta con tomar sus raíces cuadradas para obtener los valores singulares de  $A$ .



# Bibliografía

- [1] James W. Demmel. *Applied numerical linear algebra*. Society for Industrial y Applied Mathematics (SIAM), Philadelphia, PA, 1997, págs. xii+419.
- [2] G. Golub y W. Kahan. “Calculating the singular values and pseudo-inverse of a matrix”. En: *J. Soc. Indust. Appl. Math. Ser. B Numer. Anal.* 2 (1965), págs. 205-224.
- [3] Ben Kim. “Dimensionality and data size reduction using singular value decomposition”. En: *Issues in Information Systems* 25.3 (2024). Seattle University, págs. 231-237.
- [4] Barbara Rosario. “Latent semantic indexing: An overview”. En: *Techn. rep. INFOSYS* 240 (2000), págs. 1-16.
- [5] Chris Solomon y Toby Breckon. *Fundamentals of Digital Image Processing: A practical approach with examples in Matlab*. John Wiley & Sons, 2011.
- [6] Gilbert W Stewart. “On the early history of the singular value decomposition”. En: *SIAM review* 35.4 (1993), págs. 551-566.
- [7] Gilbert Strang. *Linear Algebra and Learning from Data*. Philadelphia, PA: Wellesley-Cambridge Press, 2019.
- [8] David S. Watkins. *Fundamentals of matrix computations*. Pure and Applied Mathematics (New York). Second editon. Wiley-Interscience [John Wiley & Sons], New York, 2002, págs. xiv+618.



## Apéndice A

# Pseudocódigos

---

### Algorithm 1 Reducir a forma bidiagonal

---

**Entradas:**  $R \in \mathbb{R}^{nxn}$  triangular superior

**Salidas:**  $B \in \mathbb{R}^{nxn}$ , bidiagonal,  $\tilde{U} \in \mathbb{R}^{nxn}$  producto de matrices Householder,  $\tilde{V} \in \mathbb{R}^{nxn}$  producto de matrices Householder,  $R = \tilde{U}B\tilde{V}^T$

**Iniciarizar**  $B = R$

**Iniciarizar**  $\tilde{U} = I_{nxn}$

**Iniciarizar**  $\tilde{V} = I_{nxn}$

**for**  $k = 1 : n$  **do**

    Buscar  $Q_k$  matriz de Householder tal que:

        Al multiplicar a izquierda por  $Q_k$  no modifique las primeras  $k - 1$  componentes

$$Q_k \begin{bmatrix} 0 \\ \vdots \\ 0 \\ b_{k-1,k} \\ b_{k,k} \\ b_{k+1,k} \\ \vdots \\ b_{m,k} \end{bmatrix} = \begin{bmatrix} 0 \\ \vdots \\ 0 \\ b_{k-1,k} \\ \tilde{b}_{k,k} \\ 0 \\ \vdots \\ 0 \end{bmatrix}$$

$$B = Q_k B$$

$$\tilde{U} = \tilde{U} Q_k$$

**if**  $k \leq n - 2$  **then**

        Buscar  $P_k$  matriz de Householder tal que:

            Al multiplicar a derecha por  $P_k$  no modifique las primeras  $k$  componentes

$$\begin{bmatrix} 0 & \cdots & 0 & b_{k,k} & b_{k,k+1} & \cdots & b_{k,n} \end{bmatrix} P_{k+1} = \begin{bmatrix} 0 & \cdots & 0 & b_{k,k} & \tilde{b}_{k,k+1} & 0 & \cdots & 0 \end{bmatrix}$$

**end if**

$$B = B P_k$$

$$\tilde{V} = \tilde{V} P_k$$

**end for**

---

---

**Algorithm 2** Calcular una iteración del algoritmo  $QR$  implícito

---

**Entradas:**  $B \in \mathbb{R}^{nxn}$ , bidiagonal**Salidas:**  $\tilde{B}, \tilde{P}, \tilde{Q} \in \mathbb{R}^{nxn}$ **Iniciar**  $\tilde{B} = B$ **Iniciar**  $\tilde{P} = I_{nxn}$ **Iniciar**  $\tilde{Q} = I_{nxn}$ **for**  $k = 1 : n - 1$  **do**    Buscar rotador de Givens  $V_k$  (en el plano  $[k, k + 1]$ ) tal que    **if**  $k = 1$  **then**

$$\text{primera columna de } V_k = \begin{bmatrix} \beta_1^2 - \rho \\ \gamma_1 \beta_1 \\ 0 \\ \vdots \\ 0 \end{bmatrix}$$

**else**         $\tilde{B}V_k$  tiene 0 en la posición  $(k - 1, k + 1)$     **end if**     $\tilde{B} = \tilde{B}V_k$      $\tilde{Q} = \tilde{Q}V_k$     Buscar rotador de Givens  $U_k$  (en el plano  $[k, k + 1]$ ) tal que  $U_k^T \tilde{B}$  tenga 0 en la componente  $(k + 1, k)$      $\tilde{B} = U_k^T \tilde{B}$      $\tilde{P} = \tilde{P}U_k$ **end for**

---