MSEL
Modelling in Science Education and Learning

Modelling in Science Education and Learning
Volume 4, No. 13, 2011.
Instituto Universitario de Matemática Pura y Aplicada

Matrices: un modelo para las fotografías digitales

María Elena Domínguez Jiménez
Universidad Politécnica de Madrid
edominguez@etsii.upm.es

Abstract

En este trabajo se resalta la relación que existe entre dos disciplinas, aparentemente muy distintas: la fotografía digital y las matemáticas. En efecto, las imágenes digitales -que todos manejamos hoy en día- se modelizan matemáticamente como matrices. Más aún, todas las manipulaciones sobre fotografías digitales se expresan mediante operaciones matriciales. En concreto, aquí se plantea un problema de compresión de imágenes digitales, basado en la DVS (descomposición en valores singulares) que los alumnos aprenden en clase de Álgebra Lineal. De esta forma, los alumnos descubren por sí mismos algunas realidades importantes que los docentes queremos transmitirles: en primer lugar, que los conocimientos teóricos adquiridos en Álgebra Lineal tienen una aplicación directa a la tecnología que les rodea diariamente en este mundo digital; y en segundo lugar, cómo la modelización matemática es una poderosa herramienta para resolver problemas prácticos reales.

In this work we emphasize the relationship between two -apparently very different- disciplines: digital photography and mathematics. In fact, digital images -that everybody handles nowadays- are mathematically modeled by matrices. Moreover, all the manipulations over digital pictures can be expressed by means of matricial operations. In particular, here we set a problem of compression of digital images, based in the SVD (singular value decomposition), a concept learnt by the students of Linear Algebra courses. In this way, the students discover, by themselves, some important issues that teachers want to transmit to them: on one hand, that the theoretical concepts of Linear Algebra are directly applied to the technology that surrounds them everyday in our digital world; on the other hand, how mathematical modelization provides a powerful tool for the resolution of real practical problems.

Keywords: Image compression, singular value decomposition (SVD).

1 Introducción

En este trabajo se propone al alumno una aplicación directa, y a la vez amena, de un resultado matricial teórico. Se trata de la descomposición en valores singulares (DVS) que los alumnos de Álgebra Lineal conocen, y de cómo a partir de ella se puede obtener un método de compresión de imágenes digitales.

2 Motivación

Toda imagen digital en blanco y negro no es más que una matriz numérica de NM píxeles, distribuidos en N píxeles por fila y M píxeles por columna. En cada píxel se asocia un número que indica su nivel de blanco (0=negro absoluto, pasando por varios niveles de gris, hasta llegar al blanco absoluto).

Por otro lado, una imagen digital en color es la superposición de 3 imágenes: R, G, B que contienen, respectivamente: nivel de rojo, verde y azul.

Así pues, toda fotografía digital es una matriz de M filas y N columnas, o bien la superposición de 3 de este tipo.

En otras palabras: *toda manipulación de fotografía digital no es más que una operación entre matrices.* Quien domine la teoría de matrices, dominará las transformaciones de imágenes digitales.

Problema: dada una imagen digital A, ¿cómo calcular otra imagen B, que contenga menor información, y tal que la "distancia" entre A y B sea "pequeña"?

Pongamos el problema matemáticamente; para ello hay que definir una distancia entre A y B. La forma más fácil es medir distancia mediante una norma, $\|A-B\|$. Las normas conocidas son la norma espectral (o norma 2) y la norma de Frobenius. En este trabajo buscaremos que el error en norma 2, es decir $\|A-B\|_2$ sea mínimo.

Por otro lado, el concepto "contener menor información" puede entenderse como que el rango de B sea menor que el rango de A, es decir $r(B) < r(A)$. Obsérvese que B sigue siendo una matriz de dimensiones $N \times M$, igual que A.

Así pues, el problema, traducido a lenguaje matemático, es

$$\text{“Dada } A, \text{ encontrar una matriz } B \text{ de rango menor que } r(A), \text{ tal que } \|A-B\|_2 \text{ sea mínimo”}$$

Pues bien, en este trabajo se explica cómo dar solución a este problema, por medio de la herramienta matricial conocida como "DVS de la matriz A".

3 Requisitos teóricos

La Descomposición en Valores Singulares de A consiste en escribir

$$A = U\Sigma V^h \tag{13.1}$$

siendo U matriz ortogonal de orden M, V matriz ortogonal de orden N, y Σ matriz diagonal con elementos diagonales positivos (llamados valores singulares de A) que se denotan por σ y suelen ordenarse de mayor a menor: $\sigma_1 \geq \sigma_2 \geq \cdots \geq 0$.

La expresión matricial (13.1) también puede reescribirse así:

$$A = \sum_{j=1}^{r} \sigma_j u_j v_j^h = \sigma_1 u_1 v_1^h + \cdots + \sigma_r u_r v_r^h \tag{13.2}$$

donde r es el rango de A, u_j denota la columna j-ésima de U y v_j denota la columna j-ésima de V.

La identidad (13.2) expresa A como suma de r matrices de rango 1. Por tanto, la información de A queda determinada por el siguiente número de datos: el mínimo entre $(N + M + 1)\, r$ y NM. Así pues, para reducir la información de A, basta que nos quedemos con menos de r sumandos, pero ¿qué sumandos elegimos, con el fin de que la "distancia" con A sea mínima?

Para resolver esta cuestión, nos valemos de este resultado que la teoría nos brinda:

TEOREMA: Sea A una matriz de rango r y sea (13.2) su DVS. De entre todas las matrices de rango $\leq k < r$, la matriz

$$A_k = \sum_{j=1}^{k} \sigma_j u_j v_j^h = \sigma_1 u_1 v_1^h + \cdots + \sigma_k u_k v_k^h \tag{13.3}$$

es la que tiene distancia mínima con A, es decir,

$$\min_{r(B)\leq k} \|A - B\|_2 = \|A - A_k\|_2 \,.$$

Además dicha distancia mínima es

$$\|A - A_k\|_2 = \sigma_{k+1}.$$

Así pues, la matriz A_k viene dada por $(N + M + 1)\, k$ con lo que necesitamos quedarnos aproximadamente sólo con el k/r por ciento de la información inicial.

La demostración de este resultado puede consultarse, por ejemplo, en [1] (páginas 72–73).

SOLUCIÓN: si se desea que A se aproxime por una matriz de forma que el error cometido sea menor que ϵ, basta elegir k tal que $\epsilon > \sigma_{k+1} \geq 0$ y entonces la correspondiente matriz (13.3) tiene rango k, el ratio de compresión es del orden de $(r - k)/r$ por ciento, y además la distancia entre la matriz A y su aproximada A_k es

$$\|A - A_k\|_2 = \sigma_{k+1} < \epsilon.$$

 ISSN 1988-3145

4 Fichero Matlab

```
function A=compresImagSVD(path,rango)
if nargin<1, path='picture.jpg'; end
A=imread(path);
B1=double(A(:,:,1));
B2=double(A(:,:,2));
B3=double(A(:,:,3));
disp('Calculando la descomposición DVS ...');
[U1,S1,V1]=svd(B1);
[U2,S2,V2]=svd(B2);
[U3,S3,V3]=svd(B3);
r=max([rank(S1),rank(S2),rank(S3)]);
dim=rango;
S1a=zeros(size(S1));
S1a(1:dim,1:dim)=S1(1:dim,1:dim);
S2a=zeros(size(S2));
S2a(1:dim,1:dim)=S2(1:dim,1:dim);
S3a=zeros(size(S3));
S3a(1:dim,1:dim)=S3(1:dim,1:dim);
Ba1=U1*S1a*V1';
Ba2=U2*S2a*V2';
Ba3=U3*S3a*V3';
Aa(:,:,1)=uint8(Ba1);
Aa(:,:,2)=uint8(Ba2);
Aa(:,:,3)=uint8(Ba3);
image(Aa);
title(['Aproximacion de rango ', int2str(dim)])
tamOr=min(prod(size(A)),(sum(size(A))-2)*3*r);
tamCompr=(sum(size(A))-2)*3*dim;
ratCompr=(tamOr-tamCompr)*100/tamOr;
disp(['Rangos:  imagen COMPRIMIDA=' num2str(dim) ' (imagen ORIGINAL=' num2str(r)
')']);
disp(['Tamaño imagen ORIGINAL: ' num2str(tamOr)]);
disp(['Tamaño imagen COMPRIMIDA: ' num2str(tamCompr)]);
```

```
disp(['RATIO de COMPRESION: ' num2str(ratCompr) ' %']);
```

5 Ejemplo

A una imagen inicial de la Figura 1, que corresponde a una matriz de rango 362, se le aplica el proceso detallado, y se obtienen varias aproximaciones, de rango 2, 4, 8, 16, y 64, respectivamente.

Se observa que la aproximación de rango 64 (figura 7) es prácticamente igual a la original, el error es mínimo y además contiene tan sólo el 30% de la información de la matriz original. Es decir, se ha obtenido una aproximación óptima con un ratio de compresión del 70%.

6 Observaciones avanzadas

- Es importante resaltar que el famoso método de compresión JPEG (basado en la Transformada Discreta del Coseno, DCT) no proporciona un ratio de compresión tan elevado como la DVS: en efecto, el método presentado para la DVS aproxima *cada imagen* por su *aproximación óptima*. En cambio, la DCT transforma todas las imágenes de la misma manera, no es una transformación optimizada a cada imagen particular, como la DVS. Sin embargo, la ventaja de la DCT es su facilidad de aplicación mediante algoritmos rápidos, junto con una eficiencia más que aceptable; por ello, el método de compresión JPEG se ha extendido mundialmente, aunque no sea óptimo.

  Para más información, visitar por ejemplo http://en.wikipedia.org/wiki/JPEG.

- Existen muchas más aplicaciones de la DVS en el campo de la Fotografía Digital. Por mencionar alguna, la DVS se utiliza para el reconocimiento de caras. A esta aplicación se le suele denominar "eigenfaces". Una explicación de la misma puede consultarse en http://www.scholarpedia.org/article/Eigenfaces.

- En [2] se estudia la DVS también para recontrucción de superficies tridimensionales, pero el contenido teórico es más avanzado, y se escapa al objetivo de este trabajo.

7 Conclusiones

En este trabajo se estudia una aplicación del Álgebra Matricial en el campo de la compresión de Imágenes Digitales. Esta aplicación de la teoría no sólo ayuda a los alumnos a entender mejor el concepto de la Descomposición en Valores Singulares (DVS), sino que también los motiva a aprenderlo, ya que observan su utilidad directa en la vida que los rodea. Adicionalmente, los alumnos pueden programar el algoritmo en Matlab (u otro programa similar); con ello, por un lado, confirmarán si han entendido bien el concepto, y por otro, podrán realizar simulaciones con otras imágenes digitales de prueba. En resumen, se trata de un ejemplo altamente pedagógico para alumnos de Álgebra Lineal.

Figura 1: Imagen original, de tamaño 564720, y rango 362.

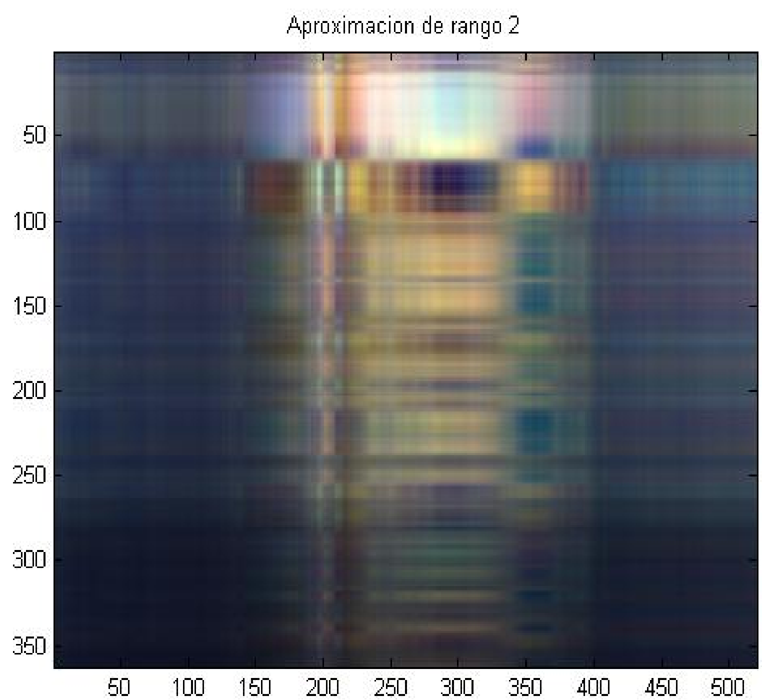

Figura 2: Aproximación de rango 2, ratio de compresión: 99 %

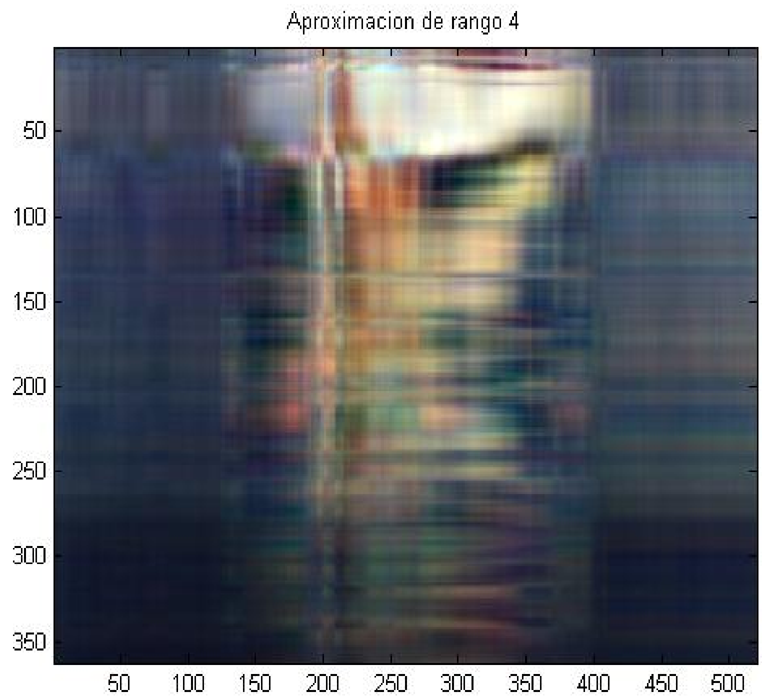

Figura 3: Aproximación de rango 4, ratio de compresión: 98 %

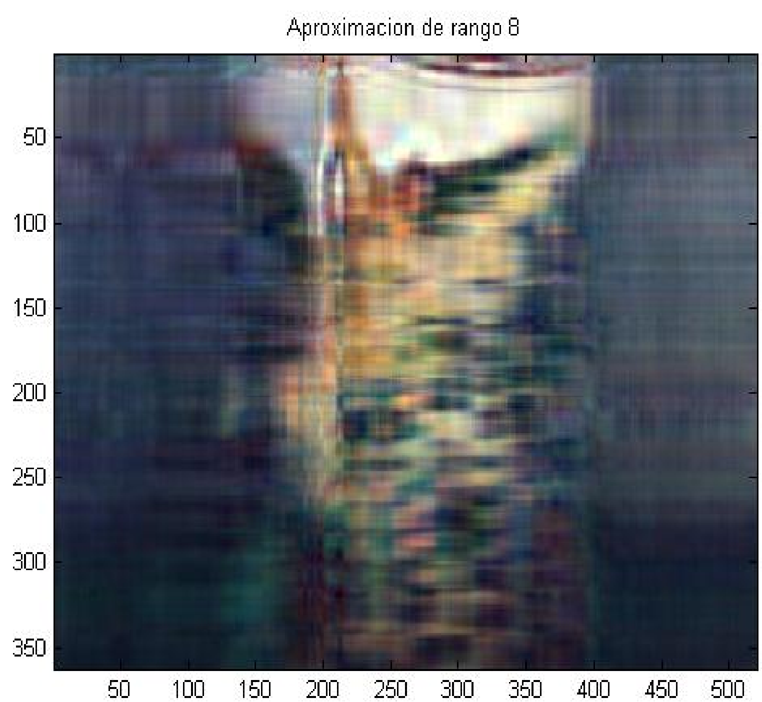

Figura 4: Aproximación de rango 8, ratio de compresión: 96 %

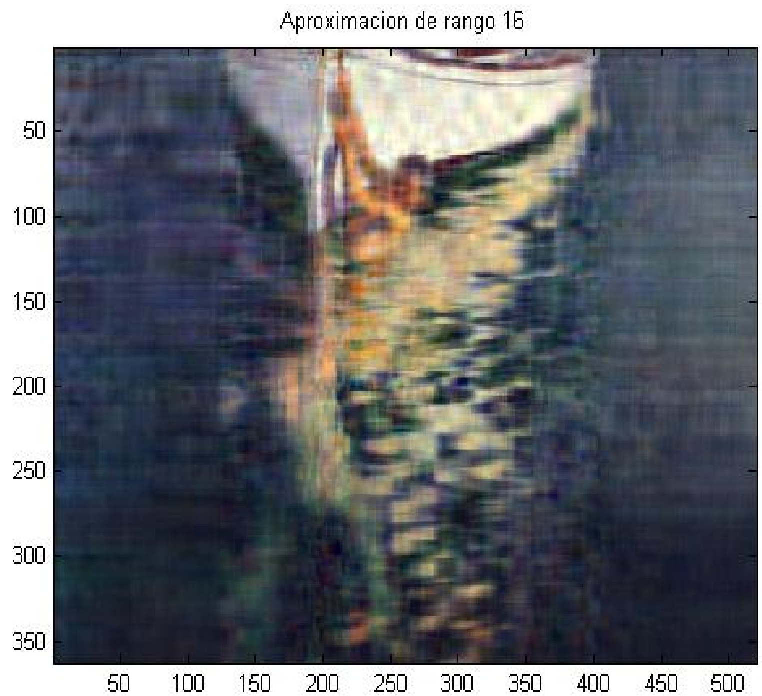

Figura 5: Aproximación de rango 16, ratio de compresión: 92 %

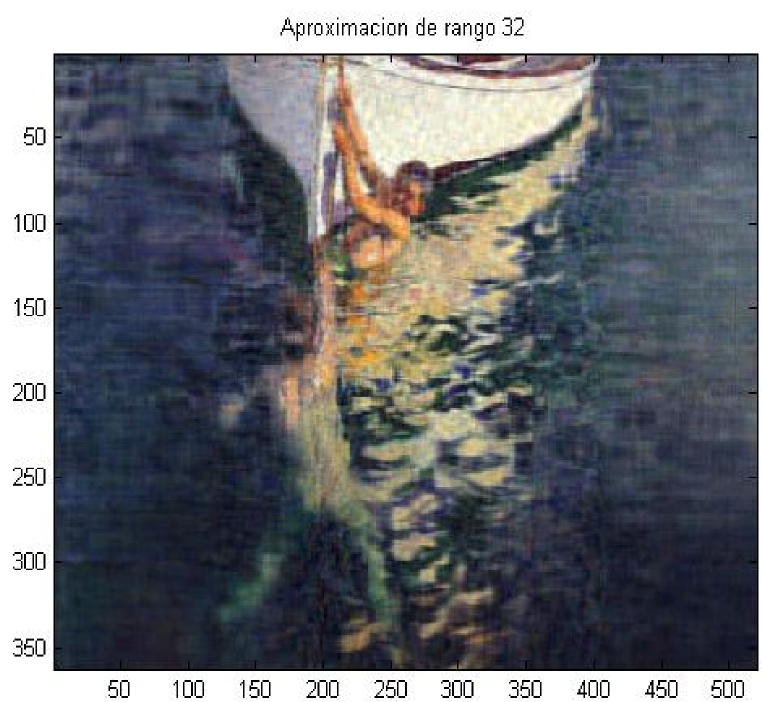

Figura 6: Aproximación de rango 32, ratio de compresión: 85 %

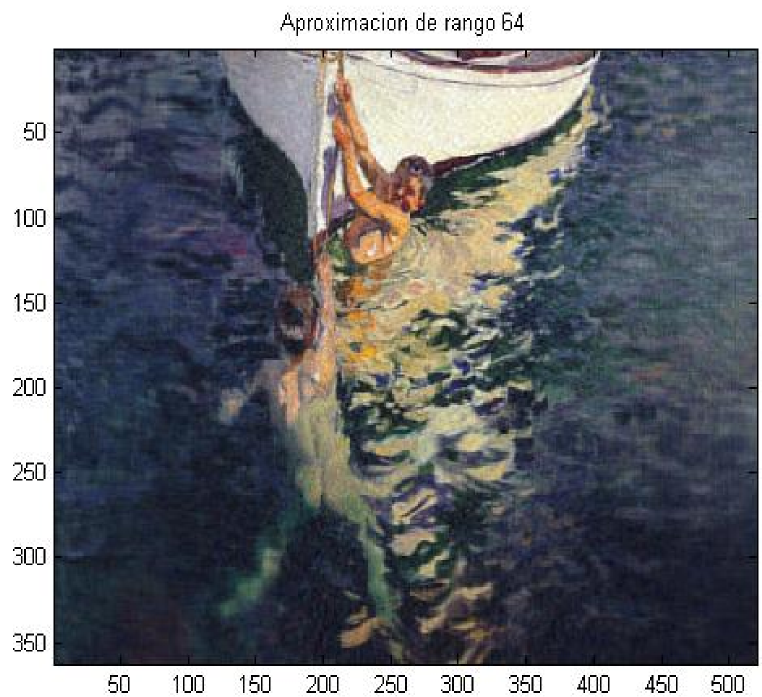

Figura 7: Aproximación de rango 64, ratio de compresión: 70 %

Referencias

[1] G. Golub, C. Van Loan, "Matrix Computations," *Ed. Johns Hopkins,* 1996 (Third Edition).

[2] N. Muller, L. Magaia, B. Herbst, "Singular Value Decomposition, Eigenfaces, and 3D Reconstructions," *SIAM Review,* Vol. 46, No. 3, pp. 518-545, 2004.