

El uso de la descomposición en valores singulares para el tratamiento de imágenes

Leonardo Colombo

Instituto de Ciencias Matemáticas CSIC-UAM-UC3M-UCM, Madrid

e-mail: leo.colombo@icmat.es

página web: <http://www.icmat.es/leocolombo>

Ramiro Lafuente

Universidad Nacional de Córdoba, Argentina

FaMAF & CIEM

e-mail: ramlaf@gmail.com

1. Introducción

Hace ya algunos siglos, el uso de los valores singulares y la descomposición en valores singulares han sido objeto de estudio de diferentes disciplinas matemáticas como el análisis matricial, el análisis funcional y las ecuaciones en derivadas parciales, entre otras; pero sin duda, el ámbito en que mayor impacto tuvo y continúa teniendo es el del *álgebra lineal numérica*.

Dichos valores tienen una naturaleza geométrica, dada a conocer en primer lugar por el matemático italiano Eugene Beltrami [1, 7]. En el siglo XIX Beltrami, junto con otros geómetras diferenciales, buscaban reducir una forma cuadrática a una forma diagonal mediante transformaciones ortogonales. Fue Beltrami quien dio comienzo a este estudio, y al problema de la descomposición en valores singulares.

Beltrami resolvió un caso particular del problema, pero clave e indispensable para los estudios que siguieron en años posteriores. Demostró que existe tal reducción cuando la matriz A de la forma bilineal es real, no singular y cuadrada, encontrando una descomposición del tipo

$$U^T AV = S = \text{diag}(\sigma_1, \sigma_2, \dots, \sigma_n),$$

donde U y V son matrices ortogonales. Más aún, dichos valores σ_i son las raíces cuadradas positivas de AA^T (o de $A^T A$, ya que son matrices semejantes).

Otro estudio sobre formas bilineales llevó a Camille Jordan a obtener resultados similares y más generales que los obtenidos por Beltrami. Los estudios de Jordan [4] fueron luego continuados por Wielandt y Lanczos.

Independientemente, John Sylvester propuso un algoritmo para reducir una forma cuadrática a otra diagonal en [6], donde también mencionó la existencia de una iteración similar para diagonalizar una forma bilineal.

Geoméricamente, los valores singulares de una matriz A se pueden definir como las longitudes de los semiejes de la hiperelipse que se transforma mediante A en la esfera unidad; pero en este

trabajo utilizaremos la definición habitual dada en los libros de álgebra lineal numérica, al igual que las técnicas usuales en dicha disciplina.

Vale la pena mencionar que el tratamiento de imágenes tuvo un gran impacto en diversas áreas de la matemática hace algunos años, prolongándose hasta el día de hoy. Al igual que sucede con la descomposición en valores singulares, muchas disciplinas científicas están interesadas en la relación de las matemáticas con el tratamiento (compresión, eliminación de ruido, desenfoque, etc.) de imágenes. La estadística, la teoría de procesos estocásticos, el análisis armónico y la teoría del muestreo, la teoría de la información, el análisis numérico y la optimización, y, en general, la física (sin olvidar algunas ramas de la ingeniería y la informática) son algunas de las áreas que en la actualidad ponen su interés en estos problemas.

Para que el lector se haga una idea del uso actual del procesamiento de imágenes en la tecnología, vamos a hacer referencia a los satélites Landsat [5]. Estos son una serie de satélites que orbitan circularmente alrededor de la Tierra, grabando imágenes del terreno y de las costas, de modo que cualquier lugar se pueda testear con imágenes cada 8 días. Las imágenes obtenidas por estos satélites son útiles para estudiar el ritmo y la dirección del crecimiento urbano, por ejemplo. La comunidad agropecuaria las utiliza para analizar la humedad del suelo y clasificar la vegetación. En cuanto a los gobiernos, éstos pueden detectar y estimar los daños provocados por desastres naturales. Y los organismos de protección del medio ambiente las utilizan para identificar la contaminación realizada por chimeneas y medir la temperatura del agua de ríos y lagos cercanos a plantas de energía. Los sensores que están incorporados en los satélites toman siete imágenes simultáneamente de la región de la Tierra que se quiera estudiar. Cada imagen se digitaliza y se guarda en una matriz rectangular, donde cada entrada es un número que indica la intensidad de señal de un punto (o *pixel*) de la imagen.

Dichas imágenes suelen tener mucha información redundante, la cual ocupa espacio innecesariamente, y uno trata de comprimirla para que esto no ocurra. Pero tal compresión puede provocar que la imagen resulte ilegible. En este trabajo indagaremos en la relación que hay entre la descomposición en valores singulares y la compresión de imágenes.

Luego de introducir el tema mediante una revisión de los resultados principales que usaremos sobre la descomposición en valores singulares, veremos una aplicación de la misma a la compresión de imágenes. Finalmente, estudiaremos una aplicación del método en la cual se resolverá un problema de mínimos cuadrados cuya solución será una imagen codificada en forma de vector.

2. Preliminares y notación

2.1. Acerca de la Descomposición en Valores Singulares (SVD)

Los valores singulares de una matriz $A \in R^{m \times n}$ se definen como las raíces cuadradas no negativas de los autovalores de la matriz simétrica $A^t A \in R^{m \times n}$, que siempre es definida no negativa (o semidefinida positiva). Estos se ordenan en sentido decreciente:

$$\sigma_1(A) \geq \sigma_2(A) \geq \dots \geq \sigma_p(A) \geq 0,$$

donde $p = \min(m, n)$. Si el contexto lo permite, se omite la matriz A de su expresión y se denotan simplemente por: $\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_p \geq 0$.

Si $A \in R^{m \times n}$, con $m \geq n$, entonces existen matrices U, V, S tales que

$$A = USV^t, \tag{1}$$

donde U es una matriz $m \times m$ que satisface $U^t U = I$, V es una matriz $n \times n$ que satisface $V^t V = I$, y S es una matriz $m \times n$ con un bloque superior diagonal $n \times n$ cuya diagonal es $\sigma_1, \dots, \sigma_n$ (siendo $\sigma_1 \geq \dots \geq \sigma_n \geq 0$), y un bloque inferior $(m-n) \times n$ de ceros. Es decir:

$$S = \begin{bmatrix} \text{diag}(\sigma_1, \dots, \sigma_n) \\ 0_{(m-n) \times n} \end{bmatrix}.$$

Esta descomposición de A como producto de tres matrices cumpliendo las condiciones mencionadas se llama *Descomposición en Valores Singulares*, o SVD por sus siglas en inglés ("*Singular Value Decomposition*").

Si $r = \text{rg}A$, entonces

$$\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_r > \sigma_{r+1} = \dots = \sigma_n = 0.$$

O sea, que en caso de que A sea de rango deficiente ($r < n$), A posee $n-r$ valores singulares nulos. Las columnas de las matrices $U = [u_1, \dots, u_n]$ y $V = [v_1, \dots, v_n]$ contienen los vectores singulares a izquierda y derecha, respectivamente, de A ; a diferencia de los valores singulares, éstos no están unívocamente determinados. Se sobrentenderá en adelante que las afirmaciones que se hagan sobre ellos son independientes de los vectores elegidos. Pero hay que tener cierto cuidado, ya que los vectores singulares a derecha v_i de A son vectores propios de $A^t A$ y los vectores singulares a izquierda u_i de A son vectores propios de AA^t . Sin embargo, esto no significa que cualesquiera bases ortonormales de vectores propios de $A^t A$ y AA^t puedan ser utilizadas como vectores singulares a derecha e izquierda, respectivamente, de A . Las columnas v_i de cualquier matriz ortogonal V que diagonalice a $A^t A$ pueden servir como vectores singulares por la derecha de A , pero los vectores singulares por la izquierda correspondientes u_i vienen dados por las fórmulas $Av_i = \sigma_i u_i$ si $i = 1, 2, \dots, r$, es decir,

$$u_i = \frac{Av_i}{\sigma_i} = \frac{Av_i}{\|Av_i\|_2}, i = 1, 2, \dots, r;$$

y para $\{u_{r+1}, u_{r+2}, \dots, u_m\}$ se elige cualquier base ortonormal de $\ker A^t$. También se cumple:

$$A^t u_i = \sigma_i v_i, \quad i = 1, \dots, r.$$

Además, para los subespacios núcleo e imagen de A , se tiene que

$$\ker A = \text{span}\{v_{r+1}, \dots, v_n\}, \quad \text{Im } A = \text{span}\{u_1, \dots, u_r\},$$

donde span denota el subespacio generado por dichos vectores.

Con las notaciones

$$U_k = [u_1, \dots, u_k] \in \mathbb{R}^{m \times k}, \quad V_k = [v_1, \dots, v_k] \in \mathbb{R}^{n \times k},$$

se tiene

$$\begin{aligned}\operatorname{Im} A &= \operatorname{Im} U_r, \\ \operatorname{Im} A^t &= \operatorname{Im} V_r,\end{aligned}$$

y las relaciones importantes

$$\begin{aligned}\ker A \oplus \operatorname{Im} A^t &= \mathbb{R}^{n \times 1}, \\ \operatorname{Im} A \oplus \ker A^t &= \mathbb{R}^{m \times 1},\end{aligned}$$

donde \oplus denota la suma directa ortogonal.

2.2. Descomposición diádica

La descomposición diádica de la matriz A se obtiene directamente de la descomposición en valores singulares mostrada en (1), y es la siguiente:

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

Se llama diada a toda matriz de rango 1, de ahí el nombre de la expresión anterior. Esta terminología proviene de la Física, donde se habla de tensores diádicos. Usaremos la norma espectral de matrices $\|A\|_2$, i.e. la norma inducida por las normas euclídeas en $\mathbb{R}^{m \times 1}$ y $\mathbb{R}^{n \times 1}$,

$$\|A\|_2 = \max_{x \in \mathbb{R}^{n \times 1}} \frac{\|Ax\|_2}{\|x\|_2};$$

Se puede demostrar que $\|A\|_2 = \sigma_1(A)$.

2.3. Teorema de Eckart y Young

Siguiendo en el contexto anterior, dado un entero k , $0 \leq k < r$, sea

$$A_k = \sum_{i=1}^k \sigma_i u_i v_i^T.$$

Entonces, el teorema de Eckart y Young asegura que

$$\min_{\operatorname{rg} X \leq k} \|X - A\|_2 = \|A_k - A\|_2 = \sigma_{k+1}(A).$$

O sea, la matriz de rango menor o igual a k que minimiza la distancia a A (en norma 2) es precisamente A_k . En particular,

$$\min_{\operatorname{rg} X < r} \|X - A\|_2 = \|A_{r-1} - A\|_2 = \sigma_r(A).$$

2.4. Resolución de problemas de cuadrados mínimos por medio de la SVD

Estudiaremos sistemas de ecuaciones lineales del tipo

$$Ax = b,$$

donde $A \in \mathbb{R}^{m \times n}$ es la matriz del sistema, $x \in \mathbb{R}^{n \times 1}$ es la incógnita, y $b \in \mathbb{R}^{m \times 1}$ es un vector arbitrario de escalares.

En el caso de un sistema con igual cantidad de incógnitas que ecuaciones ($m = n$), se pueden hallar las soluciones mediante métodos directos, como por ejemplo eliminación gaussiana, descomposición LU, Cholesky, etc.¹. En el caso de que estos métodos no puedan ser aplicados (por problemas de memoria o tiempos de ejecución muy grandes) se pueden utilizar métodos iterativos, los cuales aproximan la solución x progresivamente para luego, al cabo de cierta cantidad de iteraciones, obtener una solución *acceptable*.

Cuando el sistema es *incompatible*, se intenta buscar un x de modo tal que Ax se aproxime a b tanto como sea posible. O sea, se busca minimizar la expresión

$$\|Ax - b\|_2^2. \quad (2)$$

En general, cuando $m > n$ no hay un x que satisfaga $Ax = b$. Es en estos casos donde se resuelve el problema de minimizar la expresión expuesta anteriormente, también conocido como *problema de mínimos cuadrados*. Es posible demostrar que el x que minimiza dicha expresión es la solución del sistema lineal

$$A^t(Ax) = A^tb.$$

Para resolver este tipo de sistemas, en los cuales la matriz es cuadrada ($A^tA \in \mathbb{R}^{n \times n}$), debemos considerar dos casos: cuando el rango por columnas de A es completo, y cuando no lo es (para más información, ver [2, 3]).

Si $rg(A) = n$, entonces A^tA es definida positiva, y podemos resolver el sistema utilizando la descomposición de Cholesky. Sin embargo, cuando $rg(A) < n$ no puede ser utilizado este método, ya que algunos autovalores de la matriz A^tA son nulos. Es en estos casos donde se puede aplicar la Descomposición en Valores Singulares (SVD). Estudiaremos el caso en el que el rango de la matriz A sea deficiente, esto es, menor a la cantidad de columnas.

Dada la matriz del problema $A \in \mathbb{R}^{m \times n}$ ($m \geq n$), supongamos que $rg(A) = r < n$, o sea, que A es de rango deficiente. En este caso, no existe un único x que minimiza (2), sino que en realidad el conjunto de tales vectores es de dimensión $n - r$. Intentaremos encontrar dichos x , y considerar como solución al problema el x de norma mínima. Como vimos anteriormente, existen exactamente r valores singulares de A que son no nulos, y estos son los únicos que influyen en la SVD de A . Luego, escribamos la SVD de A como:

$$A = USV^t = [U_1, U_2] \begin{bmatrix} S_1 & 0 \\ 0 & 0 \end{bmatrix} [V_1, V_2]^t = U_1 S_1 V_1^t. \quad (3)$$

¹ Para la aplicación de dichos métodos se debe suponer que el sistema es compatible determinado.

donde $S_1 \in \mathbb{R}^{r \times r}$ es una matriz diagonal (submatriz de S), en cuya diagonal se encuentran los valores singulares no nulos de A en orden decreciente, y U_1, V_1 representan las primeras r columnas de las matrices U, V , respectivamente. Dada la descomposición en (3), es posible demostrar que todas las soluciones al problema de mínimos cuadrados son de la forma

$$x = V_1 S_1^{-1} U_1^t b + V_2 z,$$

con z un vector arbitrario de dimensiones adecuadas. Como las columnas de V_1 y V_2 son mutuamente ortogonales, tenemos por el teorema de Pitágoras que

$$\|x\|_2^2 = \|V_1 S_1^{-1} U_1^t b\|_2^2 + \|V_2 z\|_2^2,$$

lo cual claramente se minimiza tomando $z = 0$ (vector nulo).

De esta forma, conociendo la SVD de nuestra matriz A , podemos obtener el x de norma mínima que minimiza (2), mediante la fórmula

$$x = V_1 S_1^{-1} U_1^t b = A^+ b, \quad (4)$$

donde A^+ es la matriz pseudo inversa de Moore-Penrose de A (ver por ejemplo [3]).

3. Compresión de imágenes mediante el uso de la SVD

En esta sección presentamos una aplicación de la descomposición en valores singulares de una matriz, relacionada con la compresión de imágenes (representadas por matrices) de una manera más económica.

3.1. Levantamiento de imágenes a partir de una matriz

¿A qué nos referimos cuando hablamos de *imágenes representadas por matrices*? Si imaginamos la imagen como una tabla de píxeles, cada uno con una *densidad* determinada, lo que obtenemos es en realidad una matriz, cuyas entradas indican el color (o valor de gris, si fuese en blanco y negro) de cada píxel de la imagen en cuestión. De esta forma, podemos trabajar con las imágenes como si fueran matrices numéricas cualesquiera.

Por ejemplo, la matriz identidad (de una determinada dimensión) representa, en términos de imágenes, un cuadrado blanco con su diagonal descendente pintada de negro (los números entre 0 y 1 nos dan la densidad de color gris que debe *pintarse* en cada píxel, asumiendo que el 0 representa al color blanco puro y el 1 al negro).

3.2. Compresión de la imagen

Si tuviéramos la necesidad de enviar una imagen en forma de matriz, digamos por correo electrónico, podría surgir el problema de que las dimensiones de la matriz fuesen exageradamente grandes (por ejemplo, una matriz de 10.000 x 10.000), y en consecuencia la cantidad de memoria necesaria para guardar los datos que contiene sería extremadamente grande. Mostraremos un método mediante el cual, enviando sólo información *parcial* de la matriz (es decir, *no* la matriz completa), logramos que se pierda la menor cantidad de información posible sobre la imagen.

Dada la matriz $A \in \mathbb{R}^{m \times n}$ que representa a la imagen en cuestión lo que hacemos en primer lugar es, a partir de su descomposición en valores singulares $A = USV^t$, obtener su descomposición diádica

$$A = \sum_{i=1}^n \sigma_i u_i v_i^T, \quad (5)$$

donde u_i, v_i son los vectores columna de U, V respectivamente, y σ_i son los valores singulares de la matriz A . El resultado principal que fundamenta este método es el teorema de Eckart y Young, el cual asegura que si sólo conservamos los primeros k términos ($k < n$) de la expresión en (5), entonces la distancia entre la matriz obtenida y A es exactamente σ_{k+1} . Además, el mencionado teorema nos asegura que la matriz $A_k = \sum_{i=1}^k \sigma_i u_i v_i^T$ es la más próxima a A entre todas las matrices de rango k . Como los valores singulares $\sigma_1, \dots, \sigma_k, \sigma_{k+1}, \dots$ se encuentran en orden decreciente (llegando incluso a ser nulos si $\text{rg}A < n$), entonces con la matriz A_k tenemos una buena aproximación de nuestra matriz inicial A .

Lo interesante de este resultado es que para transmitir la matriz A necesitamos enviar las n columnas de U y de V además de los n valores singulares de A , mientras que para conocer A_k sólo necesitamos conocer k de los n elementos antes mencionados.

Vale la pena señalar que, sorprendentemente, no hacen falta demasiados valores singulares para obtener una aproximación aceptable de la imagen. Como veremos en las imágenes de ejemplo, con enviar solo un cuarto, o incluso menos valores singulares de nuestra matriz, podemos reconstruir la imagen casi a la perfección.

3.3. Algoritmos de Matlab

A continuación presentaremos los comandos de Matlab² necesarios para obtener la descomposición en valores singulares y para levantar imágenes correspondientes a las matrices A_k mencionadas anteriormente.

Matlab presenta una importante librería de imágenes, codificadas en forma de matrices, las cuales se pueden utilizar para ilustrar lo expuesto en esta sección. Nosotros hemos elegido la imagen *durer.mat*. Esta es una matriz de 630 x 509. Para trabajar con ella, basta con cargarla a la memoria de Matlab de la siguiente manera:

```
load durer.m
```

Y la matriz correspondiente a la imagen *durer.m* queda guardada en la variable X . Ahora buscamos las matrices U, S y V de la SVD de x , mediante la función `svd` de Matlab:

```
[U, S, V]=svd(X);
```

Finalmente, para mostrar la imagen obtenida al quedarnos con sólo k valores (y vectores) singulares, lo que debemos hacer es darle un valor a k , y escribir el código necesario para levantar la imagen correspondiente:

```
colormap(map)
k=...
image(U(:,1:k)*S(1:k,1:k)*V(:,1:k)')
```

² Matlab® es una marca registrada de Math Works.

3.4. Imágenes para ilustrar la idea

Luego de aplicar repetidas veces el método en Matlab como mostramos antes, obtenemos las imágenes correspondientes a los diferentes valores de k . Observamos que la imagen se encuentra perfectamente distinguible con $k = 40$ valores singulares, y que ya con 150 valores obtenemos una calidad casi idéntica a la conseguida con la totalidad de la matriz (Figuras 1 y 2).

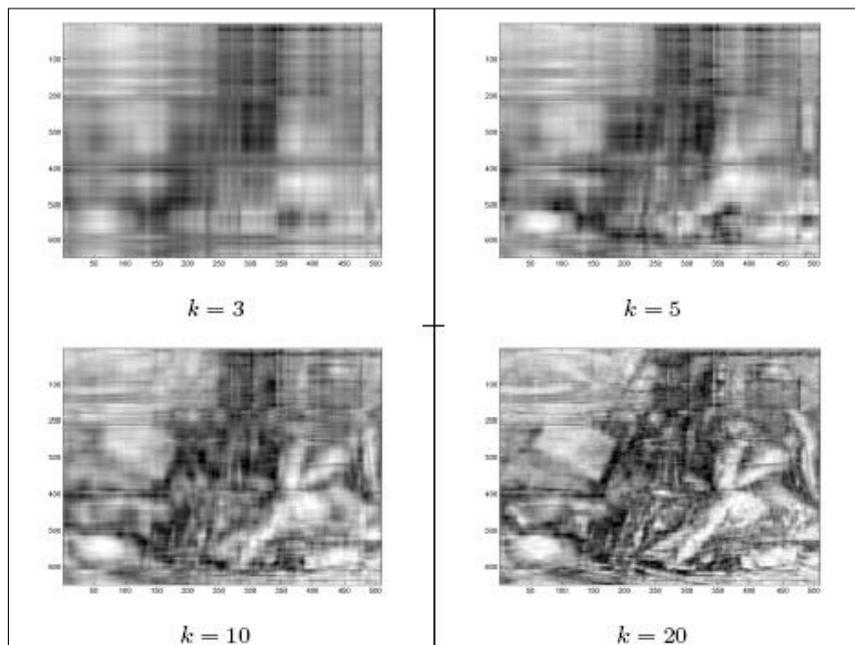


Figura 1. Reconstrucción de la imagen usando la SVD.

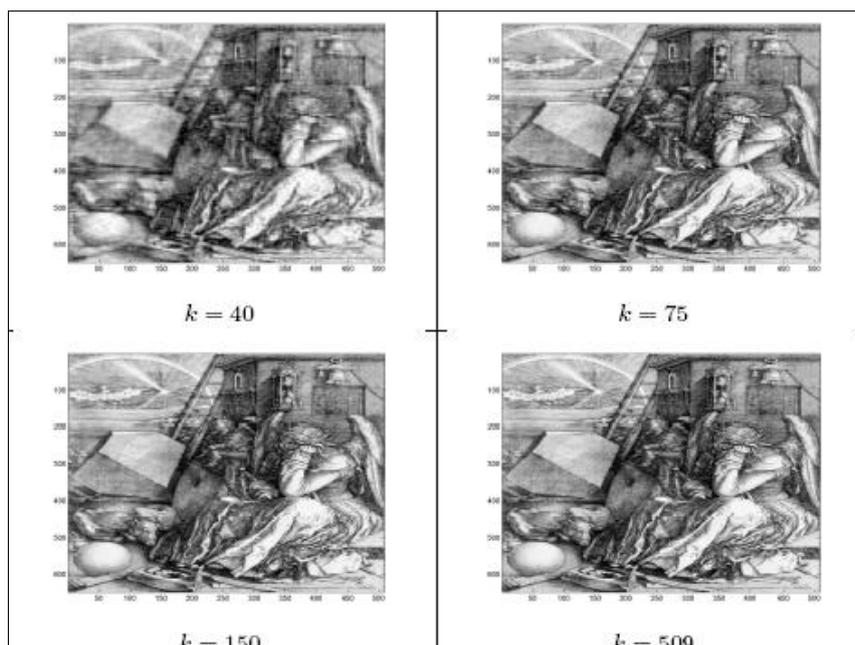


Figura 2. Reconstrucción de la imagen usando la SVD.

3.5. Acerca de la imagen Durer.m

La imagen que hemos escogido para este ejemplo, *durer.m* (Figura 3), representa el grabado *Melancolía I* del pintor renacentista alemán Albrecht Dürer, y se considera su obra más misteriosa por la geometría y cuestiones matemáticas que aparecen en ella.



Figura 3. *Melancolía I*, de A. Dürer.

Dürer manifiesta su interés hacia la matemática en esta obra. En ella se pueden observar semicircunferencias, poliedros, esferas y, lo más llamativo, el cuadrado mágico que aparece en la parte superior derecha.

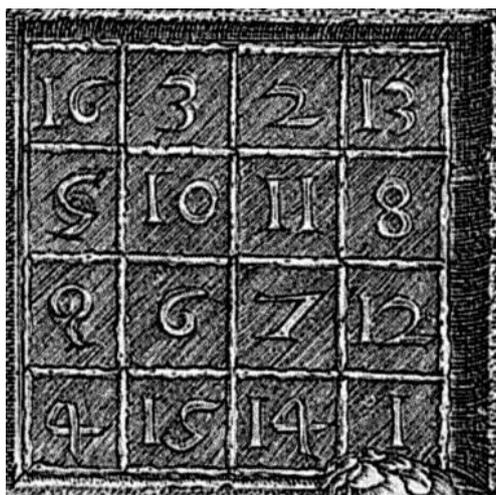


Figura 4. Detalle del cuadrado mágico en *Melancolía I* de A. Dürer.

Observemos la particularidad de este cuadrado mágico (Figura 4): todas las filas, columnas, diagonales principales y submatrices de 2×2 suman el número mágico, 34.

4. Aplicación a la resolución de un problema de cuadrados mínimos

A continuación veremos una aplicación del método de descomposición en valores singulares para la resolución de un sistema de ecuaciones particular. En dicho sistema, lo que obtendremos como solución será una imagen (a diferencia de la sección anterior, la imagen estará codificada en el vector x y no en la matriz A). Este problema surge en la práctica, cuando queremos reconstruir una imagen a partir de la emisión de rayos X en una tomografía computerizada.

4.1. Reconstrucción de imágenes por emisión de rayos X en una tomografía computerizada

Una *tomografía computerizada* brinda una imagen digitalizada de una región plana R , con el objetivo de conseguir una reconstrucción en blanco y negro de la mencionada región R . Para obtener la imagen digitalizada, debe procederse de la siguiente manera:

- i) Enmarcar en un rectángulo la zona R a reproducir.
- ii) Dividir el rectángulo en pequeños cuadrados, los denominados *píxeles*.
- iii) Colorear cada cuadradito (píxel) asignándole un color en la gama blanco-negro, es decir, colorear de gris con una intensidad predeterminada.

Si denotamos n a la cantidad de píxeles obtenidos entonces, cuanto mayor sea n , mejor será la calidad de la imagen reproducida. El mismo procedimiento se puede aplicar si lo que se desea reconstruir es un cuerpo con volumen.

En las imágenes reproducidas por un tomógrafo computerizado la intensidad del color asignado a cada píxel de la malla depende de la densidad del tejido cuya imagen se desea reproducir. Considerando que, *a priori*, la densidad del cuerpo a reconstruir es desconocida (es la incógnita del problema a resolver), el tomógrafo deberá procesar información que le permita inferir la densidad del cuerpo en cada uno de los píxeles de la malla. En lo que sigue haremos una breve descripción del modo en que un tomógrafo computerizado opera cuando intenta reconstruir imágenes planas:

- i) Una fuente externa al cuerpo a reconstruir emite un rayo i que atraviesa al mencionado cuerpo.
- ii) Al atravesar el cuerpo el rayo i tiene una pérdida de intensidad, la que se supone que es directamente proporcional a la densidad del cuerpo atravesado.
- iii) Un receptor externo recibe al rayo y mide la atenuación total sufrida por el rayo, a la que denotamos b_i .

Supongamos que la región a reconstruir es atravesada por m rayos, numerados de 1 a m , y que la malla posee n píxeles numerados de 1 a n . Obsérvese que a partir del conocimiento del ángulo que forma el rayo i con la fuente emisora y del tamaño de la malla, se puede determinar la longitud del tramo de rayo que atraviesa cada píxel³. Denotamos a_{ij} a la longitud del rayo i que atraviesa el píxel j , y b_i a la atenuación del rayo i medida en el receptor. Si suponemos que la densidad del cuerpo es constante en cada píxel de la malla, y que el factor de atenuación es proporcional a la longitud a_{ij} del tramo del rayo i que atraviesa el píxel j , la contribución a la atenuación total del rayo i debida al paso por el píxel j será entonces $a_{ij}x_j$, para cierto x_j (que depende únicamente del píxel j y no del rayo).

³ Obviamente el rayo no atraviesa algunos píxeles, en cuyo caso la longitud del tramo asociada a tal píxel es cero para ese rayo.

Así, la atenuación total en el rayo i provendrá de sumar la atenuación sufrida a lo largo de todos los píxeles que el rayo atraviesa; es decir,

$$\sum_{j=1}^n a_{ij} x_j$$

debe coincidir con la atenuación b_i medida por el receptor. En otras palabras, las incógnitas x_j deben satisfacer

$$\sum_{j=1}^n a_{ij} x_j = b_i, \quad i = 1, \dots, m. \quad (6)$$

Para poder inferir la densidad del tejido del píxel j , debemos hallar la atenuación x_j ocurrida al atravesar el píxel j . Teniendo en cuenta que un rayo sólo atraviesa algunos píxeles, se deduce que para reconstruir toda la imagen se debe atravesar la región por varios rayos y procesar en forma conjunta la información obtenida por todos ellos. Luego, debemos hallar los valores de x_j que satisfacen el sistema de ecuaciones (6), siendo m el número de rayos emitidos por las fuentes. Usando notación matricial, reconstruir la imagen equivale a resolver el sistema

$$Ax = b.$$

Cada fila de A corresponde a un rayo, y los elementos de la fila i -ésima de A son

$$a_{i1}, a_{i2}, \dots, a_{ij}, \dots, a_{in},$$

siendo $a_{ij} = 0$ si el rayo i no pasa por el píxel j .

Cabe destacar que:

- i) Para conseguir una buena reconstrucción de la imagen debemos dividir la región en un número grande de píxeles (n grande) y usar muchos rayos para atravesarla (m grande)⁴.
- ii) Un rayo puede atravesar sólo unos pocos píxeles; por lo tanto, la ecuación asociada a cada rayo tiene muchos coeficientes a_{ij} nulos. Así, la matriz A es dispersa (*sparse* en inglés).
- iii) Obsérvese que si la cantidad de rayos emitidos es menor que la cantidad de píxeles, $m < n$, entonces el sistema tendrá infinitas soluciones. En este caso, quizás se desee hallar aquella solución que da la menor atenuación de píxeles. Tal solución es la de mínima norma.
- iv) Si la cantidad de rayos emitidos es mayor que la cantidad de píxeles, $m > n$, entonces el sistema podría no tener solución. Lo mismo puede ocurrir si por errores de redondeo en las mediciones realizadas para cualquier a_{ij} , el sistema resulta *incompatible*.

⁴ Nótese que el sistema es generalmente de gran escala.

Además, la atenuación de la energía es una integral de línea a lo largo de la trayectoria del rayo i , y b_i es el valor de esa integral, mientras que cada ecuación planteada es una suma de Riemann. Eso dice que es altamente probable que el sistema planteado sea inconsistente. En tal situación, una posibilidad interesante es hallar valores para x_j que satisfagan que la diferencia $Ax - b$ sea tan pequeña como sea posible. Es decir, buscamos la solución al problema de mínimos cuadrados del sistema $Ax = b$.

4.2. Presentación y resolución del problema mediante SVD

Resolveremos el problema de mínimos cuadrados

$$\min_{x \in \mathbb{R}^n} \|Ax - b\|_2^2,$$

donde A es la matriz que representa las longitudes de los diferentes rayos que atraviesan la región, b representa la atenuación de cada rayo medida en el receptor, y x , la incógnita, son las densidades de la región en cada píxel.

En nuestro caso particular, la matriz A es de dimensiones 144×144 y de rango aproximadamente 125. Como $rg(A) < n$, no podemos utilizar la ecuación normal y luego la descomposición de Cholesky. Es por eso que resolveremos la cuestión haciendo uso de la SVD de la matriz A :

$$A = USV^t.$$

Como ya hemos visto, la incógnita x de menor norma que resuelve el problema viene dada por la fórmula (4):

$$x = V_1 S_1^{-1} U_1^t b = A^+ b.$$

Para hallar la solución x en la práctica, se puede proceder en Matlab como explicaremos a continuación.

4.2.1. Algoritmos de Matlab para la resolución del problema

Primero debemos leer los datos de los archivos correspondientes. Supongamos que tenemos los datos de A y b guardados en los archivos $A.dat$, $b.dat$ respectivamente. Entonces, los leemos en Matlab con el siguiente código:

```
fid1=fopen('A.dat');  
A=fscanf(fid1,'%f %f',[144 144]);  
fclose(fid1)  
fid2=fopen('b.dat');  
b=fscanf(fid2,'%f ',[144,1]);  
fclose(fid2)
```

Para hallar la matriz pseudoinversa de A basta con escribir

```
C=pinv(A).
```

Luego, la solución al problema de cuadrados mínimos que estamos buscando es

```
x=C*b.
```

Recordemos que dicha solución está todavía en forma de vector (de 144×1). Como representa una imagen, debemos traducir x a una matriz cuadrada Y , de 12×12 , usando el comando de Matlab

```
Y = reshape(x,12,12);
```

Ahora sólo resta comparar este resultado con la verdadera solución del problema.

4.3. Estudio de los resultados obtenidos

Nos interesa saber en cuánto difiere nuestro resultado de la solución real al problema. Para comparar las imágenes, debemos convertir las matrices que tenemos (el vector solución obtenido, y el que sabemos que es solución real al problema) en imágenes, y observar qué nivel de similitud encontramos.

4.3.1. Algoritmo de Matlab para levantar imágenes

A continuación presentamos el algoritmo para convertir una matriz en una imagen en blanco y negro. En Matlab, debemos escribir:

```
figure(a)  
imagesc(Y)  
colormap(gray)
```

donde a es el número de imagen que queremos asociarle, e Y es la matriz a levantar.

4.3.2. Comparación con la solución "verdadera"

Sólo resta ver las imágenes que hemos obtenido con los algoritmos ya expuestos y con la ayuda de Matlab. Observamos en la [Figura 5](#) que las imágenes no son completamente iguales; sin embargo, la solución obtenida se aproxima muchísimo (y en varios lugares coincide exactamente) con la verdadera.

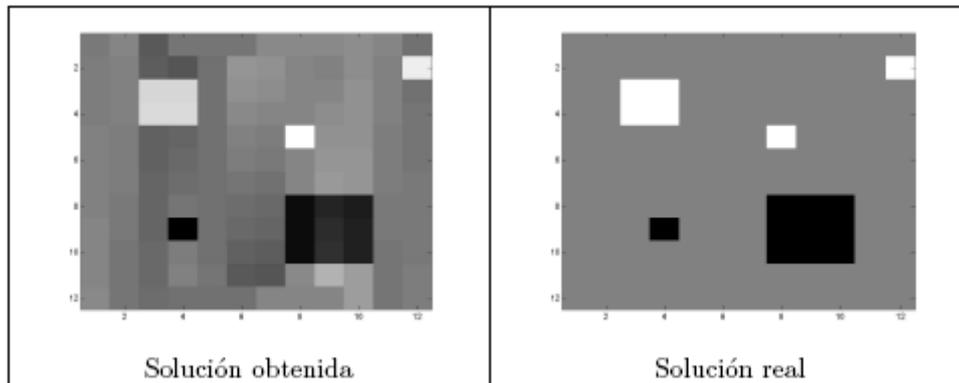


Figura 5. Solución obtenida vs. solución real.

5. Conclusiones

A lo largo del trabajo hemos visto la utilización de la Descomposición en Valores Singulares para distintas aplicaciones.

Con respecto a la compresión de imágenes, vale la pena mencionar que es un ejemplo perfecto para mostrar el poder de algunas herramientas del Álgebra Lineal Numérica (y en particular de la SVD).

Hemos podido observar los cambios en las imágenes a medida que k aumentaba, y llegamos a la conclusión de que lo primordial para recuperar la información de la matriz A fueron sólo unos pocos valores singulares (con sus respectivos vectores u_i, v_i): los mayores.

Además, vimos cómo levantar una imagen (motivados por la reconstrucción de imágenes por emisión de rayos X) resolviendo un sistema de ecuaciones con infinitas soluciones, planteando el problema como uno de mínimos cuadrados, pues estábamos trabajando con matrices de rango deficiente. Hemos visto cómo la solución obtenida (una imagen codificada en forma de vector) se aproxima mucho a la solución real, mostrando en cierta manera la eficiencia del método. Si bien la resolución del problema de mínimos cuadrados se podría haber realizado mediante otro método, las dimensiones de nuestro sistema no eran *demasiado* grandes, por lo que no tuvimos inconvenientes al aplicar el método de la SVD, mediante el cual se obtuvo la solución casi inmediatamente.

6. Agradecimientos

Los autores quieren agradecer a Nélida Echebest (Universidad Nacional de La Plata-Argentina) sus críticas y recomendaciones. L. Colombo agradece al programa predoctoral JAE-Pre del CSIC, y R. Lafuente al CONICET-Argentina.

Referencias

- [1]  E. Beltrami: Sulle funzioni biliniari. *Giornale di Matematiche ad Uso degli Studenti Delle Università* 11 (1873), 98-106.
- [2]  J. Demmel: *Applied numerical linear algebra* (1st ed.). SIAM, 1997.
- [3]  R. Horn, Ch. Johnson: *Topics in matrix analysis*. Cambridge University Press, 1991.
- [4]  C. Jordan: Sur la réduction des formes bilinéaires. *Comptes Rendues de l'Académie des Sciences* 78 (1874), 614-617.
- [5]  NASA: *Landsat Program*, <http://landsat.gsfc.nasa.gov>.
- [6]  J. Sylvester: A new proof that a general quadratic may be reduced to its canonical form (that is, a linear function of squares) by means of a real orthogonal substitution. *The Messenger of Mathematics* 19 (1889), 1-5.
- [7]  I. Zaballa Tejada: *Valores singulares. ¿Qué son? ¿Para qué sirven?*, http://www.ehu.es/izaballa/Cursos/valores_singulares.pdf

Sobre los autores



Leonardo Colombo (i) es licenciado en Matemática por la Universidad Nacional de La Plata (Argentina, 2009). Actualmente es becario JAE-Pre del Instituto de Ciencias Matemáticas (ICMAT) en Madrid, donde realiza su trabajo de tesis doctoral titulado sobre el tema *Métodos geométricos y numéricos para teoría de control óptimo* bajo la dirección del Dr. David Martín de Diego.

Ramiro Lafuente es licenciado en Matemática por la Universidad Nacional de La Plata (Argentina, 2009). Actualmente es becario de CONICET en FaMAF (Universidad Nacional de Córdoba, Argentina), para realizar su tesis doctoral bajo la dirección del Dr. Jorge Lauret sobre el tema *El flujo de Ricci y las métricas solitones en variedades homogéneas*.

