Proyecto Final Álgebra Lineal OT23

Rolando Rivas Dávalos 594276

Instrucciones

1. Siga los lineamientos de la rúbrica de evaluación. Los equipos pueden ser hasta de 4 integrantes. No se reciben trabajos individuales.
2. Resuelva cada uno de los problemas planteados. Utilice MATLAB, incluya los gráficos que considere necesarios para justificar sus respuestas. Responda a todas las preguntas utilizando texto.
3. Es la sección de Conclusiones siga las instrucciones dadas.
4. Una vez completada la actividad, exporte a PDF y envíelo a través de la bandeja de entrega en MRooms. **Asegúrese de que el archivo contenga todo de manera legible y ordenada.**

Objetivos

- Conocer sobre aplicaciones del álgebra lineal más allá de lo visto en el curso.
- Identificar los conceptos aprendidos en el curso aplicables a la resolución de los problemas.
- Interpretar los resultados para dar una conclusión.

Problema a resolver

- Investigue sobre el tema 'Descomposición en valores singulares' y sus aplicaciones. Elabore un resumen con los aspectos más importantes. Asegúrese de incluir información sobre 'aproximaciones de rango bajo' y compresión de imágenes utilizando la SVD.

Descomposición en Valores Singulares

La descomposición en valores singulares es una factorización de cualquier matriz A en 3 nuevas matrices U, S y V que relevan información importante de la matriz A. U es la resultante de multiplicar A*A´ (transpuesta de A), V resulta de multiplicar A´*A y S es una matriz diagonal de eigenvalores de A. Tiene aplicaciones en áreas como compresión de imagenes, de reducción de ruido, sistemas de recomendaciones y resolucion de ecuaciones lineales.

La aproximación de rango bajo k se refiere a una aproximación de alguna matriz A en la que al descomponer A en valores singulares U, S y V se seleccionan k columnas de cada una de las tres nuevas matrices para obtener una nueva matriz aproximada de A al hacerla el producto matricial. Esa nueva matriz es una aproximación de A en cuanto a que se usó menos información para reconstruir A.

- Utilice la función svd() en MATLAB para calcular A_1, la aproximación de rango 1 de la matriz

$$A = \begin{bmatrix} 1 & 2 & 3 \\ 3 & 3 & 4 \\ 5 & 6 & 7 \end{bmatrix}.$$

```
A = [1 2 3; 3 3 4; 5 6 7];
[U, S, V] = svd(A)
```

```
U = 3×3
   -0.2904    0.9504   -0.1114
   -0.4644   -0.2418   -0.8520
   -0.8367   -0.1957    0.5115
S = 3×3
   12.5318         0         0
         0    0.9122         0
         0         0    0.3499
V = 3×3
   -0.4682   -0.8261   -0.3136
   -0.5581    0.0012    0.8298
   -0.6851    0.5635   -0.4616
```

```
k = 1;
U1 = U(:, 1:k);
S1 = S(1:k, 1:k);
V1 = V(:, 1:k);
A1 = U1 * S1 * V1'
```

```
A1 = 3×3
    1.7039    2.0313    2.4935
    2.7243    3.2477    3.9867
    4.9087    5.8517    7.1832
```

- Utilice la función svd() en MATLAB para calcular A_2, la aproximación de rango 2 de la matriz A. ¿Qué aproximación es mejor, A_1 o A_2? Explique por qué.

```
k = 2;
U2 = U(:, 1:k);
S2 = S(1:k, 1:k);
V2 = V(:, 1:k);
A2 = U2 * S2 * V2'
```

```
A2 = 3×3
    0.9878    2.0324    2.9820
    2.9065    3.2474    3.8624
    5.0561    5.8515    7.0826
```

- Para la matriz A, la descomposición en valores singulares es $A = USV^T$, donde $U = [\mathbf{u_1}\ \mathbf{u_2}\ \mathbf{u_3}]$. Calcule el producto punto de $\mathbf{u_1}$ y $\mathbf{u_2}$, guárdelo en una variable d_1. Calcule el producto cruz de $\mathbf{u_1}$ y $\mathbf{u_2}$ y guarde el resultado en una variable c. Calcule el producto punto de $\mathbf{c}$ y $\mathbf{u_3}$. ¿Tienen sentido sus resultados? Explique por qué.

```
d1 = dot(U(:,1), U(:,2))
```

```
d1 = 1.6653e-16
```

```
c = cross(U(:,1), U(:,2))
```

```
c = 3×1
   -0.1114
   -0.8520
    0.5115
```

```
dot(U(:,3), c)
```

```
ans = 1.0000
```

```
%La matriz U es ortogonal, podemos demostrar esto con un producto punto
%entre ambos vectores el cual da 0.
%El tercer vector tiene que ser ortogonal tanto a u1 como a u2 y se
%comprueba esto con un producto cruz, creando otro vector ortogonal a u1 y
%u2. Por ultimo comprobamos que la matriz es otronormal al multipliar dos
%vectores iguales por el producto punto y dar de resultado 1.
```

- Determine sI las columnas de la matriz U generan a $\mathbb{R}^3$. Explique su respuesta.

```
syms a b c
x = [a;b;c];
r = [U x];
rref(r)
```

ans =

$$\left(\begin{array}{cccc} 1 & 0 & 0 & -\frac{343004116523404461767736090136329120146668912 64\,a}{1181011424105780413767716563923932856216164612 13}-\frac{180973330625750391542431}{389733769954907536543334} \\ 0 & 1 & 0 & \frac{374138749638638025843438674967035518764921651 20\,a}{393670474701926804589238854641310952072054870 71}-\frac{31412068668712115893722}{1299112566516358455144} \\ 0 & 0 & 1 & \frac{664545684302479960135793510068724849578694672384\,c}{1299112566516358455144488220316326141837781073343}-\frac{33205511970613650487363}{389733769954907536543334} \end{array}\right.$$

```
% Debido a que el sistema de ecuaciones r tiene solución infinita se
% concluye que las columnas de U generan a R3
```

- Cargue el archivo imagen_grayscale.mat, disponible en MRooms, y utilice la función de MATLAB imshow() mostrar la imagen almacenada. Para esta imagen, encuentre el valor de k que generará un cociente de compresión (compression ratio) de $CR \approx 2$. Para ese valor de k (redondeado al entero más cercano), encuentre la aproximación de rango k para esta imagen. Deberá aplicar el comando `uint8()` a la matriz aproximante antes de usar el comando imshow() para que se muestre la imagen que resulta.

```
Image = importdata('imagen_grayscale.mat')
```

```
Image = 866×1280 uint8 matrix
    93    93    93    93    93    93    92    94    97    98    97    96    95 ···
    93    93    93    93    93    93    92    94    97    99   100   100   100
    94    94    94    94    94    94    92    95    97   101   104   104   103
    94    94    95    95    95    97    98    96    99   101   105   105   105
```

```
    94    95    95    96    96    98    99    97   100   100   101   101   101
    96    96    97    97    97    99    98    98    98    98    98    98    98
    97    97    97    98    98    98    97    96    97    96    96    96    96
    97    97    97    99    99    98    96    95    96    96    96    96    96
    99    99    99    99    99    98    97    96    96    96    96    96    96
    97    97    98    98    98    97    96    96    97    97    96    96    95
     ⋮
```

```
imshow(uint8(Image))
```

```
Cols = size(Image, 2);
k = round(Cols/2)
```

```
k = 640
```

```
[U, S, V] = svd(double(Image));
U1 = U(:, 1:k);
S1 = S(1:k, 1:k);
V1 = V(:, 1:k);
A1 = U1 * S1 * V1';
imshow(uint8(A1))
```

- Repita el apartado anterior pero ahora para $CR \approx 5$, $CR \approx 10$ y $CR \approx 25$. Explique la tendencia que se observa en la imagen conforme CR aumenta. Diga qué CR considera mejor y por qué.

```
% CR = 5
k = round(Cols/5)
```

```
k = 256
```

```
U1 = U(:, 1:k);
S1 = S(1:k, 1:k);
V1 = V(:, 1:k);
A1 = U1 * S1 * V1';
imshow(uint8(A1))
```

```
% CR = 10
k = round(Cols/10)
```

```
k = 128
```

```
U1 = U(:, 1:k);
S1 = S(1:k, 1:k);
V1 = V(:, 1:k);
A1 = U1 * S1 * V1';
imshow(uint8(A1))
```

```
% CR = 25
k = round(Cols/25)
```

```
k = 51
```

```
U1 = U(:, 1:k);
S1 = S(1:k, 1:k);
V1 = V(:, 1:k);
A1 = U1 * S1 * V1';
imshow(uint8(A1))
```

```
% Conforme aumenta el cociente de compresion la imagen se vuelve cada vez
% menos nítida. Esto debido a que menos columnas de U, S y V se toman en
% cuenta, menos información hay en la imagen y menos precisa se vuelve.

% Cuando el CR se aproxima a 10 la imagen aun sigue siendo nítida para la
% mayoria de dispositivos. Cuando el CR es 25 la imagen ya no es nitida y
% se nota la perdida de calidad.
%
% El cociente de compresion ideal para la mayoria de situaciones en las que
% se necesita la mayor calidad con el menor peso en memoria sería un CR =
% 10 para transmicion en internet e impreso en papel.
```

Conclusiones

El álgebra lineal es un área de las matemáticas que nos da muchas posibilidades y herramientas para resolver problemas. La compresion de imagenes para su transportación es un problema resuelto con la manipulación de matrices. Veo la importancia que esta técnica tiene en el mundo de la tecnología, podemos comprimir la información de una imagen al igual que mantener su calidad con el proceso antes mostrado. Tambien esta herramienta amplió nuestro conocimiento de matlab y sus comandos que son

utilizados para realizar operaciones.

"Doy mi palabra que he realizado esta actividad con integridad académica"

Referencias

- Kolman, B., & Hill, D. (2006). Álgebra Lineal (8a). PEARSON Education.