Integrantes	Clave única
Ricardo Alonso Mendiola	174759
Marcela Alejandra Carrillo Gamez	155145
Enrique Ampudia González	164581

1er proyecto de Cálculo Numérico

Ejercicio 1: Método de la Potencia

Sea

$$A = \begin{pmatrix} 1 & -2 & 0 \\ 1 & 17 & -2 \\ 5 & 7 & 7 \end{pmatrix}$$

a) Usando su programa, apliquen el método de la potencia a la matriz A, con el vector inicial $q0 = (1, 1, 1)^T$. Hagan exactamente 10 iteraciones.

b) Usen el comando `[V, D] = eig(A)` de MATLAB para obtener los valores y vectores propios "exactos" de A. Comparen el valor propio dominante de `D` con su resultado del apartado 1.

c) Sea v el vector propio dominante en `V` (dado por `eig` arriba). Ese está normalizado y tiene la norma euclidiana uno. Dividan v por su elemento de mayor magnitud, con el fin de poder compararlo fácilmente con los vectores de su iteración.
Calculen las razones

$$\frac{\|q_{j+1} - v\|_\infty}{\|q_j - v\|_\infty}, \quad j = 1, 2, 3, \ldots$$

d) Usando `D`, calculen la razón $|\lambda 2/\lambda 1|$ y compárenlo con las razones del apartado 3.

Resultados

Al aplicar el método de la potencia a la matriz A obtenemos el valor propio $\lambda_1 = 15.3424$ con el vector propio

$$v_1 = \begin{bmatrix} -0.1105 \\ 0.7924 \\ 0.5999 \end{bmatrix}$$

Asimismo aplicando `eigs(A)` obetenemos las siguientes matices:

$$\mathtt{V} = \begin{bmatrix} -0.8163 & -0.0631 & 0.1106 \\ 0.1239 & 0.2316 & -0.5987 \\ 0.5642 & 0.9708 & -0.5987 \end{bmatrix}, \ \mathtt{D} = \begin{bmatrix} 1.3036 & 0 & 0 \\ 0 & 8.3453 & 0 \\ 0 & 0 & 15.3511 \end{bmatrix}$$

Donde v_1 es el tercer vector columna de la matriz `V`. En teoría, el método de la potencia debería de converger al vector propio correspondiente al valor propio dominante. En este caso, esto sucede porque los valores propios son: 15.3511, 8.3453 y 1.3036. Asimismo, calculando el cociente

$$\frac{\|q_{j+1} - v\|_\infty}{\|q_j - v\|_\infty}$$

donde v es el resultado del vector propio asociado a λ_1 y q_j es cada iteración en el método de la potencia, obtuvimos los siguientes resultados:

	1	2	3	4	5	6	7	8	9
1	1.0787	1.0165	1	1	1	1	1	1	1

Es decir, a partir de la tercera iteración llegamos al resultado debido a que la aproximación de la siguiente iteración es la mismo que la anterior.

Ejercicio 2: Método de la Potencia inversa con shift

Repitan el Ej. 1 con el método de la potencia inversa con *shift*:

- $\rho = 0$: Recuerden la razón exacta en este caso es $|\lambda_n/\lambda_{n-1}|$.
- $\rho = 8.5$: La razón exacta para ese caso se deja calcular usando `eig`.

Resultados

Con el shift $\rho = 0$, el vector obtenido se aproxima al vector propio correspondiente al valor propio 1.3036 el cual es

$$v_2 = \begin{bmatrix} 0.8163 \\ -0.1239 \\ -0.5642 \end{bmatrix}$$

Asimismo, el radio de convergencia es $|\frac{\lambda_n}{\lambda_{n-1}}| = 0.1562$, y obtuvimos los siguientes resultados para el cociente $\frac{\|q_{j+1}-v\|_\infty}{\|q_j - v\|_\infty}$

	1	2	3	4	5
1	0.6279	0.6279	0.6279	0.6279	0.6279

Podemos concluir que el método de la potencia inversas con shift aproxima de manera constante al vector propio que eberiamos de obtener debido a que el cociente entre iteraciones es constante y menor que 1. Además, el vector propio obtenido difiere en un signo con respecto al primer vector columna de la matriz V (la cual se encuentra en el ejercicio 1). Por otro lado, con el shift $\rho = 8.5$ obtuvimos las siguientes aproximaciones para valor propio más cercano a ρ

$$\lambda_3 = 8.3453, v_3 = \begin{bmatrix} -0.0631 \\ 0.2316 \\ 0.9708 \end{bmatrix}$$

Además, para el cociente $\frac{\|q_{j+1}-v\|_\infty}{\|q_j - v\|_\infty}$ obtuvimos lo siguiente

	1	2	3	4	5	6	7	8	9
1	0.4813	0.0071	0.4813	0.0071	0.4813	0.0071	0.4813	0.0071	0.4813

Al ver el resultado, hay una intercalación entre 0.0071 y 0.4813. Esto sucede debido a que hay un cambio de signo en cada iteración, lo cual podemos observar en la matriz de iteraciones

	1	2	3	4	5	6	7	8	9	10
1	0.0411	-0.0645	0.0650	-0.0650	0.0650	-0.0650	0.0650	-0.0650	0.0650	-0.0650
2	-0.2237	0.2389	-0.2386	0.2386	-0.2386	0.2386	-0.2386	0.2386	-0.2386	0.2386
3	-1	1	-1	1	-1	1	-1	1	-1	1

A partir de la tercera iteración, la aproximación del método converge a una dirección en específico.

Análisis

En teoría en el método de la potencia inversa con shift, el eigenpar obtenido debería de ser el correspondiente al valor propio más cercano al shift. En el ejercicio anterior se cumple en ambos casos porque cuando $\rho = 0$ obtenemos $\lambda = 1.3036$ y su vector asociado, y cuando $\rho = 8.5$ obtenemos $\lambda = 8.3453$. Es decir, si obtenemos, en ambos casos, el eigenpar correspondiente al valor propio más cercano al shift.

Ejercicio 3: Método de la Potencia inversa con cociente de Rayleigh

Usen la matriz del Ej. 1 y su programa de potencia inversa con cociente de Rayleigh con varios vectores iniciales q_0 para calcular tres vectores propios linealmente independientes, y sus valores propios. Además, verifiquen para alguno de los tres vectores q_0 y su límite (un eigenvector v_j) que la convergencia es cuadrática.

Resultados

Para este ejercicio utilizamos el comando `rand(3,1)` para obtener 3 vectores aleatorios que resultaran en los 3 eigenvectores linealmente independientes, los q_0 utilizados fueron:

```
[0.816500398514693; -0.123817283647833; -0.563911677034108]
[0.0461713906311539;0.0971317812358475;0.823457828327293]
[-0.110552520687031; 0.793274489615348; 0.598743454490530]
```

Lo cual nos arrojó los siguientes eigenvalores con sus correspondientes eigenvectores:

$$\lambda_1 = \texttt{15.35108877470099} \; v_1 = \begin{bmatrix} \texttt{0.110552523708467} \\ \texttt{-0.793274541003725} \\ \texttt{-0.598743385848244} \end{bmatrix}$$

$$\lambda_2 = \texttt{8.345345380481284} \; v_2 = \begin{bmatrix} \texttt{-0.063065623050813} \\ \texttt{0.231619391471733} \\ \texttt{0.970760106660485} \end{bmatrix}$$

$$\lambda_3 = \texttt{1.303565844821029} \; v_3 = \begin{bmatrix} \texttt{-0.816270640261197} \\ \texttt{0.123895943256746} \\ \texttt{0.564226937581056} \end{bmatrix}$$

De este último λ_3 calculamos la convergencia al eigenvector obtenido por la función de MATLAB `eig(A)`, por medio de la norma infinito, esto nos arrojo los siguientes resultados:

$$\texttt{1.0e-10} * \begin{bmatrix} \texttt{0.324278381924614} & \texttt{0.000102140518266} & \texttt{0.000001110223025} & \texttt{0.000001110223025} \end{bmatrix}$$

Donde se puede observar que la convergencia es cuadtrática.

Análisis

Como podemos ver por medio de los resultados anteriores, la convergencia y los resultados del método dependen únicamente del vector de entrada q_0, pues fue justamente por medio de variar éste, que logramos llegar a 3 eigenvectores linealmente independientes.

Además, al calcular la convergencia del eigenvector λ_3, con su respectivo eigenvector, podemos notar que es cuadrática, como era de esperarse, pues sabemos que aunque este método no siempre converge, si lo hace, esta convergencia es cuadrática, pues existe C > 0, tal que

$$\lim_{k\to\infty} \frac{\|q_{j+1} - v\|_\infty}{\|q_j - v\|_\infty^2} = C$$

Ejercicio 4: Método de la Potencia

Expliquen porqué el método de la potencia no funciona para la matriz siguiente:

$$A = \begin{pmatrix} 8 & 10 & 6 \\ -24 & -7 & -25 \\ 40 & 9 & 23 \end{pmatrix}$$

Resultados

Para esta matriz utilizamos el comando `eig(A)` de MATLAB, para conocer los eigenvectores y eigenvalores de la matriz A e identificar los problemas que esta pueda presentar. Los valores obtenidos fueron:

$$\lambda_1 = \texttt{16 - 8i}\sqrt[2]{3}\ v_1 = \begin{bmatrix} \texttt{0.028289929799334 - 0.195998383020010i} \\ \texttt{-0.664813350284338 - 0.195998383020010i} \\ \texttt{0.693103280083672} \end{bmatrix}$$

$$\lambda_2 = \texttt{16 + 8i}\sqrt[2]{3}\ v_2 = \begin{bmatrix} \texttt{0.028289929799334 + 0.195998383020010i} \\ \texttt{-0.664813350284338 + 0.195998383020010i} \\ \texttt{0.693103280083672} \end{bmatrix}$$

$$\lambda_3 = \texttt{-8}\ v_3 = \begin{bmatrix} \texttt{0.577350269189626} \\ \texttt{-0.577350269189626} \\ \texttt{-0.577350269189626} \end{bmatrix}$$

Análisis

Como podemos ver esta matriz presenta dos problemas por los cuales el método de la potencia no funciona:

1. El primero es que A tiene dos eigenvectores linealmente dependientes (v_1, v_2), y si bien, esto no siempre es un problema, no se garantiza que este método converja.

2. El segundo y mas importante, es que las magnitudes de los eigenvalores de A son de la forma $|\lambda_1| = |\lambda_2| > |\lambda_1|$. Como podemos ver, A tiene dos eigenvalores dominantes $(|\lambda_1|, |\lambda_2|)$ y además son complejos conjugados, por lo que el método no converge.

Por estas dos razones, el método de la potencia no converge a menos que el vector inicial sea de al forma $k * v_3,\ k \in \mathbb{R}$.

Ejercicio 5: Método QR simple y QR con shifts dinámicos

Comparen los resultados de su implementación del método QR simple y los del QR con *shift* dinámico con el resultado del comando MATLAB `eig`. (Comparen los *errores relativos* de los eigenvalores.) Para ello usen matrices dadas por el siguiente comando de MATLAB:

```
gallery( 'lehmer', 25 )
```

y apliquen los métodos como sigue:

- `MQRsimple( gallery( 'lehmer', 25 ), 1000, 1E-10)`
- `MQRdynamic( gallery( 'lehmer', 25 ), 20, 1E-10)`
- `eigs( gallery( 'lehmer', 25 ), 25)`

Además cuenten las iteraciones requeridas por su implementación del método QR simple *vs* QR con *shift* dinámico.

Resultados

En 3 vectores guardamos los valores propios de la matriz especificada, en orden descendente. Cada vector correspondió a uno de los 3 métodos. Calculamos los errores del método QR simple y QR con shifts dinámicos restando cada uno de sus vectores con el vector del método eigs de MATLAB. Por último, guardamos esta resta en dos vectores. Al calcular el error del método QR simple (primera imagen) y el error del método QR con `shifts` dinámicos (segunda imagen) obtuvimos los siguientes resultados:

1.0e-14 *	1.0e-14 *
0.177635683940025	0.177635683940025
0	0
0.044408920985006	0.222044604925031
0.088817841970013	0.133226762955019
0.133226762955019	0.011102230246252
0.088817841970013	0.044408920985006
0.072164496600635	0.044408920985006
0.094368957093138	0.033306690738755
0.016653345369377	0.027755575615629
0.041633363423443	0.019428902930940
0.013877787807814	0.011102230246252
0.041633363423443	0.013877787807814
0.011102230246252	0.027755575615629
0.008326672684689	0.002775557561563
0.012490009027033	0.036082248300318
0.012490009027033	0.030531133177192
0.012490009027033	0.006938893903907
0.002775557561563	0.006938893903907
0.009020562075079	0.024286128663675
0.014571677198205	0.019428902930940
0.006245004513517	0.009714451465470
0.067307270867900	0.013877787807814
0.023939183968480	0.002775557561563
0.031571967262778	0.006245004513517
0.042327252813834	0

Asimismo, con el siguiente algoritmo, en el cual se compara la norma 1 de cada uno de los errores, podemos concluir que el método QR con `shifts` dinámicos es más preciso

```
%Comparamos los errores
if (norm(ErrorSimple,1) > norm(ErrorDynamic,1))
    disp('Es mas preciso el dinámico')
else
    if (norm(ErrorSimple,1) < norm(ErrorDynamic,1))
        disp('Es más preciso el simple')
    else
        disp('Es lo mismo')
    end
end
```

Es decir, se cumplió la primera condición del algoritmo.

Análisis

Como muestran los vectores, para cada uno de los eigenvalores, el error es mayor utilizando el método QR simple. Por otro lado, el método QR con `shift` dinámico requiere menos iteraciones para llegar al resultado. Por lo tanto, el método QR con `shift` dinámico es más eficiente respecto al simple tanto en exactitud como número de iteraciones.

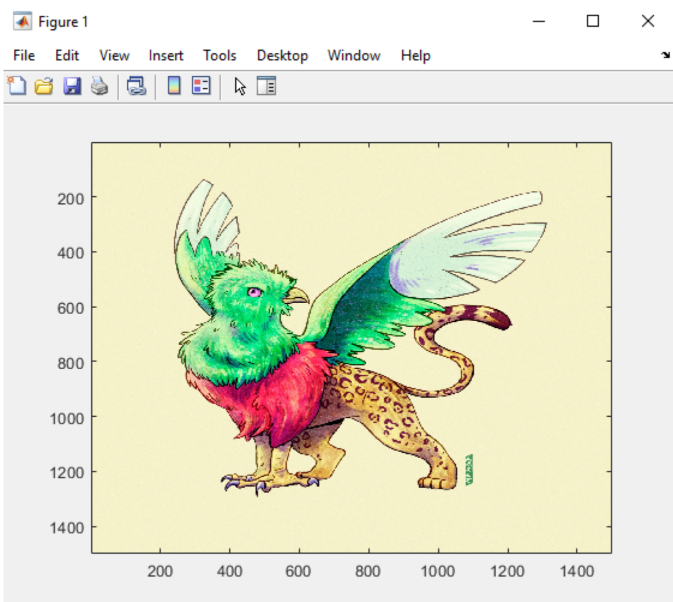
Ejercicio 6: Método SVD economizada

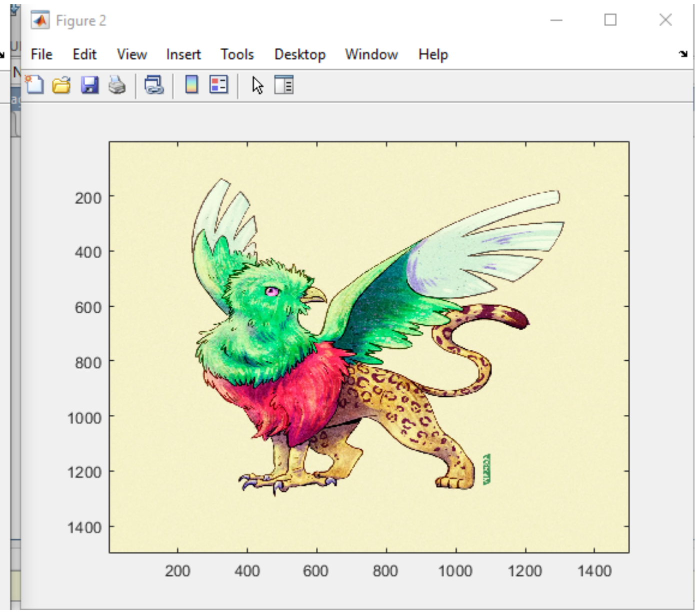
Encuentren dos archivos referentes a la práctica: `imagecompressionmSvdEcon.m` y `quetzalbalam.jpg`. Estos deben ayudar a ver si su implementación del método SVD funciona (es correcta). Comparen el rango máximo de la imagen original contra el `r` que recuperan del método. Digan cuál es la economía de los datos en el sistema al usar "A_r" en lugar de A.

Respondan de modo subjetivo: ¿Les parece satisfactoria la resolución de la compresión? ¿Cuál les parece un `r` adecuado?

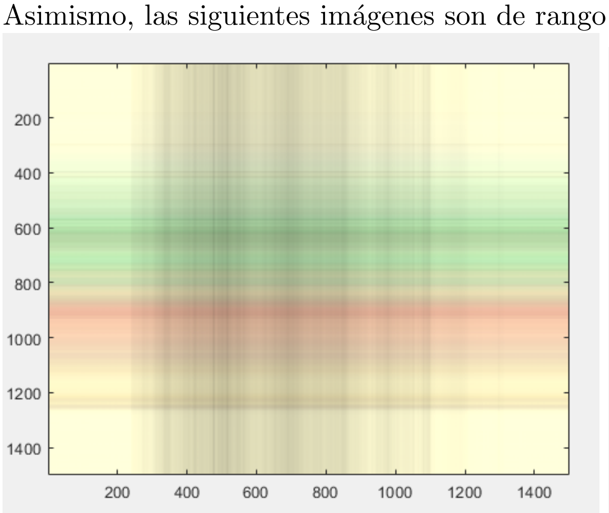
Resultados

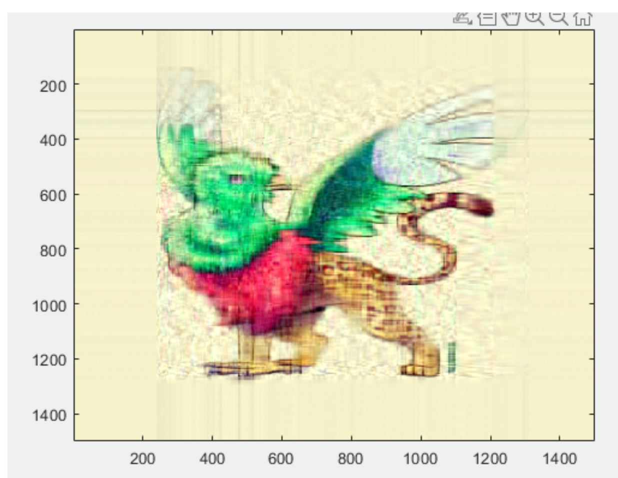
El **r** recuperado del método es de 1500, comparando ambas imágenes vemos que el método funciona. La imagen del lado izquierdo es la imagen original, y la del lado derecho es la imagen con el método utilizado

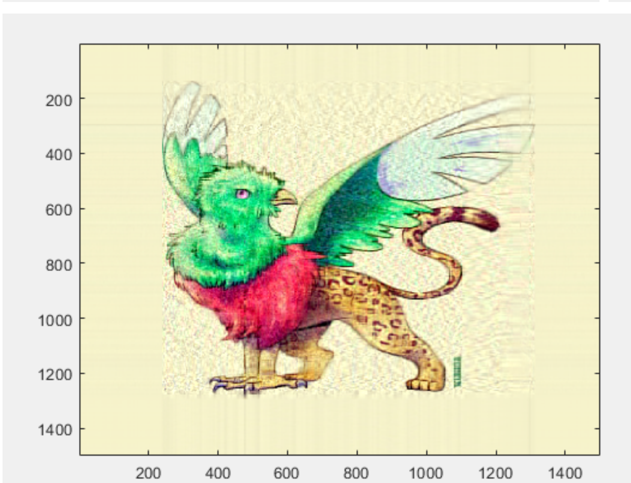


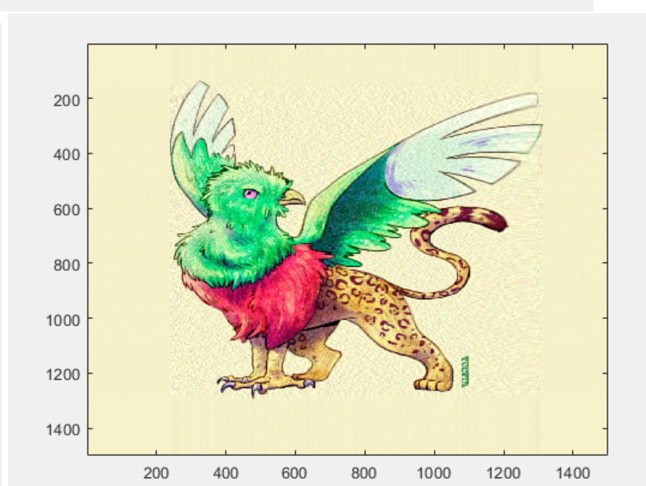


Asimismo, las siguientes imágenes son de rango 1, 25, 50 y 100 respectivamente.









Se podría decir que a partir del rango 100 tenemos una buena resolución de la imagen.