CS 132 – Spring 2020, Assignment 9

Answer A

First lets multiply the two to find Ax:

$$\boxed{\begin{bmatrix}3 & 6 & 7\\3 & 3 & 7\\5 & 6 & 5\end{bmatrix} \times \begin{bmatrix}1\\-2\\1\end{bmatrix} = \begin{bmatrix}3-12+7\\3-6+7\\5-12+5\end{bmatrix} = \begin{bmatrix}-2\\4\\-2\end{bmatrix}}$$

Hence we can see that x is in fact an eigenvector of A, and it's eigenvalue is -2.

Answer B

If we first organize ourselves to put this into $Ax = \lambda x$, then to find x we have that: $Ax - \lambda x = 0 \equiv x(A - I\lambda) = 0$
First we solve $A - I\lambda$:

$$\boxed{\begin{bmatrix}1 & 2 & 2\\3 & -2 & 1\\0 & 1 & 1\end{bmatrix} - \begin{bmatrix}3 & 0 & 0\\0 & 3 & 0\\0 & 0 & 3\end{bmatrix} = \begin{bmatrix}-2 & 2 & 2\\3 & -5 & 1\\0 & 1 & -2\end{bmatrix}}$$

Now we solve $(A - I\lambda)x = 0$:

$$\boxed{\begin{bmatrix}-2 & 2 & 2 & 0\\3 & -5 & 1 & 0\\0 & 1 & -2 & 0\end{bmatrix} \equiv \begin{bmatrix}1 & -1 & -1 & 0\\0 & 1 & -2 & 0\\0 & 0 & 0 & 0\end{bmatrix}}$$

We know that λ is a eigenvalue of A if $x(A - I\lambda) = 0$ has a non-trivial solution, which happens if it has a free variable.

Since we've already proved that already, we can easily say that it is infact a eigenvalue of A.

Now we only have to find a vector that fulfill the previously stated solution, so if we look at that previous matrix, we realize that any eigenvector must have the following form:

$$\begin{bmatrix}3x_3\\2x_3\\x_3\end{bmatrix}$$

Let $x_3 = 1$ then we have that one possible eigenvector for this is: $\begin{bmatrix}3\\2\\1\end{bmatrix}$

Answer C

If we first organize ourselves to put this into $Ax = \lambda x$, then to find x we have that: $Ax - \lambda x = 0 \equiv x(A - I\lambda) = 0$
First we solve $A - I\lambda$:

$$\boxed{\begin{bmatrix} 10 & -9 \\ 4 & -2 \end{bmatrix} - \begin{bmatrix} 4 & 0 \\ 0 & 4 \end{bmatrix} = \begin{bmatrix} 6 & -9 \\ 4 & -6 \end{bmatrix}}$$

Now we solve $(A - I\lambda)x = 0$:

$$\boxed{\begin{bmatrix} 6 & -9 & 0 \\ 4 & -6 & 0 \end{bmatrix} \equiv \begin{bmatrix} 1 & -3/2 & 0 \\ 0 & 0 & 0 \end{bmatrix}}$$

With that, we know that any eigenvector will follow the basic condition: $\begin{bmatrix} 1.5x_2 \\ x_2 \end{bmatrix}$.

$$\boxed{\text{As such, the basis for this eigenspace is: } \begin{bmatrix} 1.5 \\ 1 \end{bmatrix}}$$

Answer D

a. $\boxed{\text{True}}$
As per the definition of eigenvectors.
b. $\boxed{\text{True}}$
As if they corresponded to the same eigenvalue, due to the definition of how eigenvalues/eigenvector relate to each other, the vectors would be linearly dependent.
c. $\boxed{\text{True}}$
As the stochastic matrix to A with a a steady state vector q, satisfies the equation: $Aq = 1q$.
d. $\boxed{\text{True}}$
As per theorem 1 of this chapter, the Eigenvalues of a triangular matrix are in its main diagonal.
e. $\boxed{\text{True}}$
As the eigenspace of A will be a combination of vectors, from which we can build another matrix B for which said vectors are the nullspace of.

Answer E

Any non-invertible 2x2 matrix will only have a single eigenvalue, 0 so an example of one would be:

$$\boxed{\begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix}}$$

Answer F

The determinant of this matrix (or the characteristic polynomial) is:

$$\boxed{\text{(5-}\lambda)^2 - 3^2 = 16 - 10\lambda + \lambda^2}$$

If we solve this equation we get that $\boxed{\lambda = 2 | \lambda = 8}$

Answer G

The determinant of this matrix (or the characteristic polynomial) is:

$$\boxed{\text{(5-}\lambda) \times (3 - \lambda) - (-3)(-4) = 3 - 8\lambda + \lambda^2}$$

If we solve this equation we get that $\boxed{\lambda = -7.6 | \lambda = 7.6}$

Answer H

b. $\boxed{\text{False}}$
As per the Theorem 3, $det A^T = det A$
c. $\boxed{\text{True}}$
As per the definition above.
d. $\boxed{\text{True}}$
As no matter how many row replacements you end up doing, the eigenvalue calculation ends up with the same result.

Answer I

a.

Since v_1 is already the steady state vector of A, we first need to discover the characteristic polynomial of:

$$\begin{bmatrix} .6 & .3 \\ .4 & .7 \end{bmatrix} - \begin{bmatrix} \lambda & 0 \\ 0 & \lambda \end{bmatrix} = \begin{bmatrix} .6-\lambda & .3 \\ .4 & .7-\lambda \end{bmatrix}$$

So we get that: $(.6-\lambda) \times (.7-\lambda) - .3 \times .4 = 0 \equiv .3 - 1.3\lambda + \lambda^2 = 0$

If we solve this we get $\lambda = 1 | \lambda = 0.3$, since v_1 uses the first choice, then we must calculate the eigenvector for .3

To calculate this, we will use the equation $(A - I\lambda)x = 0$:

$$\begin{bmatrix} .3 & .3 & 0 \\ .4 & .4 & 0 \end{bmatrix} \equiv \begin{bmatrix} .3 & .3 & 0 \\ 0 & 0 & 0 \end{bmatrix} \equiv \begin{cases} x_1 = -x_2 \\ x_2 - free \end{cases}$$

$$\boxed{\text{As such, the basis for } R^2 \text{ made of } v_1 v_2\text{: } \begin{bmatrix} -1 \\ 1 \end{bmatrix}, \begin{bmatrix} 3/7 \\ 4/7 \end{bmatrix}}$$

b.

$$\boxed{\mathrm{v}_1 + cv_2 = x_0 \equiv \begin{bmatrix} 3/7 \\ 4/7 \end{bmatrix} + \begin{bmatrix} 0.5/7 \\ -0.5/7 \end{bmatrix} = \begin{bmatrix} 0.5 \\ 0.5 \end{bmatrix}}$$

c.

$$\boxed{\mathrm{x}_1 = A(x_0) = A(v_1 + cv_2) = A(v_1) + cA(v_2) = 1 \times v_1 + c \times 0.3 \times v_2}$$

$$\boxed{\mathrm{x}_2 = A(x_1) = A(v_1 + 0.3cv_2) = A(v_1) + 0.3cA(v_2) = v_1 + 0.3^2 \times c \times v_2}$$

$$\boxed{\text{In general: } x_k = v_1 + (0.3)^k \times c \times v_2}$$

$$\boxed{\text{As k increases, } x_k \text{ will tend to } v_1\text{, as } 0.3^k \text{ will tend to 0.}}$$

Answer J

As we can see, in both cases, the eigenvalues are the same, but the eigenvectors are not.

In [16]:
```
import numpy as np

A = np.random.randint(11, size=(4,4))
At = A.transpose()

(u1, v1) = np.linalg.eig(A)
(u2, v2) = np.linalg.eig(At)


print(A)
print(At)
print()
print(np.round(u1,2))
print(np.round(u2,2))
print()
print(np.round(v1,2))
print(np.round(v2,2))
```

```
[[10  8  6  6]
 [ 3  5  2  2]
 [ 8  1  4  5]
 [ 2  8  0  1]]
[[10  3  8  2]
 [ 8  5  1  8]
 [ 6  2  4  0]
 [ 6  2  5  1]]

[18.57+0.j  1.77+0.j -0.17+1.j -0.17-1.j]
[18.57+0.j  1.77+0.j -0.17+1.j -0.17-1.j]

[[-0.78+0.j   -0.37+0.j   -0.29+0.19j -0.29-0.19j]
 [-0.28+0.j    0.17+0.j   -0.04+0.05j -0.04-0.05j]
 [-0.52+0.j   -0.49+0.j   -0.26-0.43j -0.26+0.43j]
 [-0.22+0.j    0.77+0.j    0.78+0.j    0.78-0.j  ]]
[[-0.61+0.j    0.31+0.j   -0.29+0.23j -0.29-0.23j]
 [-0.61+0.j   -0.95+0.j    0.83+0.j    0.83-0.j  ]
 [-0.34+0.j    0.02+0.j    0.1 -0.3j    0.1 +0.3j ]
 [-0.37+0.j    0.07+0.j   -0.26-0.09j -0.26+0.09j]]
```

In [18]:
```
import numpy as np

A = np.random.randint(11, size=(5,5))
At = A.transpose()

(u1, v1) = np.linalg.eig(A)
(u2, v2) = np.linalg.eig(At)


print(A)
print(At)
print()
print(np.round(u1,2))
print(np.round(u2,2))
print()
print(np.round(v1,2))
print(np.round(v2,2))
```

```
[[ 0  1  7  1  6]
 [ 4  3  3 10  6]
 [ 3 10  2  6  3]
 [ 1  9  5 10  5]
 [10  2 10  3  9]]
[[ 0  4  3  1 10]
 [ 1  3 10  9  2]
 [ 7  3  2  5 10]
 [ 1 10  6 10  3]
 [ 6  6  3  5  9]]

[26.72+0.j    6.92+0.j   -3.85+0.j   -2.89+2.61j -2.89-2.61j]
[26.72+0.j    6.92+0.j   -3.85+0.j   -2.89+2.61j -2.89-2.61j]

[[-0.26+0.j   -0.29+0.j   -0.56+0.j   -0.02+0.31j -0.02-0.31j]
 [-0.45+0.j    0.12+0.j    0.3 +0.j    0.25-0.33j  0.25+0.33j]
 [-0.41+0.j    0.26+0.j   -0.26+0.j   -0.58+0.j   -0.58-0.j  ]
 [-0.54+0.j    0.53+0.j   -0.3 +0.j   -0.2 +0.22j -0.2 -0.22j]
 [-0.52+0.j   -0.74+0.j    0.66+0.j    0.54-0.14j  0.54+0.14j]]
[[ 0.32+0.j    0.44+0.j    0.65+0.j   -0.24+0.07j -0.24-0.07j]
 [ 0.44+0.j   -0.19+0.j    0.55+0.j   -0.66+0.j   -0.66-0.j  ]
 [ 0.44+0.j    0.44+0.j   -0.07+0.j   -0.02-0.43j -0.02+0.43j]
 [ 0.53+0.j   -0.69+0.j   -0.32+0.j    0.42+0.27j  0.42-0.27j]
 [ 0.48+0.j    0.31+0.j   -0.42+0.j    0.28+0.02j  0.28-0.02j]]
```

In [ ]:

Answer K

a.

$$\boxed{\begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix}}$$

b.

In order to get the eigenvalues of A, we must first find the characteristic equation of A, which would be:

$$\begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix} - \begin{bmatrix} \lambda & 0 \\ 0 & \lambda \end{bmatrix} = \begin{bmatrix} 1-\lambda & 1 \\ 1 & -\lambda \end{bmatrix}$$

So we get that: $(1-\lambda) \times (-\lambda) - 1 = \lambda^2 - \lambda - 1 = 0$

If we solve this we get that $\lambda = -0.618 | \lambda = 1.618$, now we will solve $(A - \lambda I)x = 0$, and find the eigenvectors:

$$\begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix} - \begin{bmatrix} -0.618 & 0 \\ 0 & -0.618 \end{bmatrix} = \begin{bmatrix} 1.618 & 1 \\ 1 & 0.618 \end{bmatrix} \equiv \begin{bmatrix} 0 & 0 \\ 1 & 0.618 \end{bmatrix} \equiv \begin{cases} x_1 = -0.618x_2 \\ x_2 - free \end{cases} \equiv \begin{bmatrix} -0.618 \\ 1 \end{bmatrix}$$

$$\begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix} - \begin{bmatrix} 1.618 & 0 \\ 0 & 1.618 \end{bmatrix} = \begin{bmatrix} -0.618 & 1 \\ 1 & -1.618 \end{bmatrix} \equiv \begin{bmatrix} 0 & 0 \\ 1 & -1.618 \end{bmatrix} \equiv \begin{cases} x_1 = 1.618x_2 \\ x_2 - free \end{cases} \equiv \begin{bmatrix} 1.618 \\ 1 \end{bmatrix}$$

$$\boxed{\text{Eigenvalues: } -0.6181.618}$$

$$\boxed{\text{Basis for the eigenspace: } \begin{bmatrix} 1.618 \\ 1 \end{bmatrix}, \begin{bmatrix} -0.618 \\ 1 \end{bmatrix}}$$

c.

Now that we know that the basis for the eigenspace (let it be S) is:

$$\begin{bmatrix} 1.618 & -0.618 \\ 1 & 1 \end{bmatrix}$$

We can get the diagonalization matrix easily, let it be D:

$$\begin{bmatrix} 1.618 & 0 \\ 0 & 0.618 \end{bmatrix}$$

Since we know that x_0, will be made from our S matrix multiplied by some scalar we have that: $x_0 = Sc \equiv x_0 S^{-1} = c$, so now we need S^{-1}:

$$S^{-1} = \frac{1}{det(S)} \times \begin{bmatrix} 1 & -0.618 \\ -1 & 1.618 \end{bmatrix} \equiv \begin{bmatrix} 0.447 & -0.276 \\ -0.447 & 0.724 \end{bmatrix}$$

Finally we can calculate c and get:

$$\boxed{c = \begin{bmatrix} 0.447 & -0.276 \\ -0.447 & 0.724 \end{bmatrix} \times \begin{bmatrix} 0 \\ 1 \end{bmatrix} \equiv c = \begin{bmatrix} -0.276 \\ 0.724 \end{bmatrix}}$$

d.

With all that we've gather before we know that:

$$\boxed{x_k = S \times D^k \times c}$$

e.

```
In [6]: import numpy as np

        def getFib(x):
            return np.round(((1.618 ** (x-3)) * (0.724) * (1.618)) + ((-0.618 ** (x-3)) * (0.276) * (-0.618)),4)

        print("F(40) = ", getFib(40))
        print("F(50) = ", getFib(50))
        print("F(60) = ", getFib(60))

F(40) =  63229860.3597
F(50) =  7775125279.0733
F(60) =  956076334208.8616
```

d.

If we go back to our equation, we know that: $x_k = S \times D^k \times c$, but since, when k approached infinity, D^k approaches 0, then as the sequence grows, it will tend towards Sc

Answer J

In [7]:
```python
import numpy as np

A = np.array([[0, 0, 1/10, 1/10, 1/10, 1/10],
     [1/7, 1, 0, 0, 0, 0],
     [1/10, 1/10, 0, 0, 1/10, 1/10],
     [0, 0, 1/7, 1, 0, 0],
     [1/10, 1/10, 1/10, 1/10, 0, 0],
     [0, 0, 0, 0, 1/7, 1]])
```

In [10]:
```python
#A
x, v = np.linalg.eig(A)

print(np.real(x))
```

```
[ 0.16575188  1.03424812 -0.08685593 -0.08685593  0.98685593  0.98685593]
```

In [11]:
```python
#B
a = np.amax(x)
print("The largest eigenvalue is: ", a, " which means that the array is unstable.")
```

```
The largest eigenvalue is:  1.0342481186734473  which means that the array is unstable.
```

In [12]:
```python
#C
startVal = [1, 0, 0, 0, 0, 0]
print(v.dot(startVal))
```

```
[-0.56906708  0.09744738 -0.56906708  0.09744738 -0.56906708  0.09744738]
```

In [18]:
```python
#D
b = np.real(v[1])
print("Let the eigenvalue a=", biggest, "and the vector b=", b)
print()
print("xk = b * a^k * ", v.dot(startVal)[1])
```

```
Let the eigenvalue a= 1.0342481186734473 and the vector b= [ 0.09744738 -0.56144158 -0.10640568  0.04097612 -0.81306232 -0.2854
8524]

xk = b * a^k *  0.09744738430733318
```

In [27]:
```python
#E
def getWorms(k):
    return np.round(b.dot(((a ** k) * v.dot(startVal)[1])),4)

print("F(100) = ", getWorms(100))
print("F(250) = ", getWorms(250))
print("F(500) = ", getWorms(500))
print("F(750) = ", getWorms(750))
```

```
F(100) =  [ 7.50571552e+18 -4.32440628e+19 -8.19571268e+18  3.15611459e+18
 -6.26247128e+19 -2.19890047e+19]
F(250) =  [ 1.66789878e+50 -9.60957279e+50 -1.82122799e+50  7.01342818e+49
 -1.39162858e+51 -4.88633416e+50]
F(500) =  [ 2.92953716e+102 -1.68784826e+103 -3.19884823e+102  1.23185523e+102
 -2.44428959e+103 -8.58247372e+102]
F(750) =  [ 5.14550885e+154 -2.96457689e+155 -5.61853324e+154  2.16365988e+154
 -4.29320846e+155 -1.50744613e+155]
```