**Name: Adithya M**

**SRN: PES1UG20CS621**

**MATLAB Assignment**

Unit 1:

Question:

Solve the matrix by using gaussian elimination:

x+2y+z=3,2x+y-2z=3,-3x+y+z=-6 Code:

C = [1 2 -1; 2 1 -2; -3 1 1] b= [3 3 -6]' A = [C b]; n= size(A,1); x = zeros(n,1); for i=1:n-1

for j=i+1:n

m = A(j,i)/A(i,i)

A(j,:) = A(j,:) - m\*A(i,:) end end

x(n) = A(n,n+1)/A(n,n) for i=n-1:-1:1 summ = 0

for j=i+1:n

summ = summ + A(i,j)\*x(j,:) x(i,:) = (A(i,n+1) - summ)/A(i,i) end

end

Output:

C =

## 1 2 -1

2 1 -2

-3 1 1

b =

3

3

-6

m =

2

A =

[1 2 -1 2](#_Toc22596)

[0 -3 0 3](#_Toc22597)

[-3 1 1 5](#_Toc22598)

m =

-3

A =

1 2 -1 3

## 0 -3 0 -3

0 7 -2 3

m =

-2.3333 A =

1 2 -1 3

0 -3 0 -3

0 0 -2 -4

x =

0 0

2 summ =

0 summ =

0 x =

0

1

2

summ =

0

summ =

2 x =

# 1

1

2 summ =

0 x =

3

1

2

—------------------------------------------------------------------Practice Problems:

Question:

C = [1 1 1; 2 -6 -1; 3 4 2] b= [11 0 0]

Output:

C =

1. 1 1
2. -6 -1
3. 4 2

b =

11

0

0 m =

2

A =

1 1 1 11

0 -8 -3 -22

3 4 2 0

m =

3

A =

1 1 1 11

0 -8 -3 -22

0 1 -1 -33

m =

-0.1250

A =

1.0000 1.0000 1.0000 11.0000

0 -8.0000 -3.0000 -22.0000

0 0 -1.3750 -35.7500

x =

0

0

26 summ =

0 summ = -78 x =

0

-7 26 summ =

0

summ =

-7 x =

18

-7

26 summ =

19 x =

-8

-7

26

—------------------------------------------------------------------Question:

C = [2 1 -1; 2 5 7; 1 1 1] b= [0 52 9]

Output:

C =

2 1 -1

2 5 7

1 1 1

b =

0

52

9 m =

1

A =

2 1 -1 0

1. 4 8 52
2. 1 1 9

m =

0.5000

A =

2.0000 1.0000 -1.0000 0

0 4.0000 8.0000 52.0000

0 0.5000 1.5000 9.0000

m =

0.1250

A =

2.0000 1.0000 -1.0000 0

0 4.0000 8.0000 52.0000

0 0 0.5000 2.5000 x =

0

0

5

summ =

0 summ = 40 x =

0

3

5 summ =

0 summ =

3

x =

-1.5000

3.0000

5.0000 summ =

-2 x =

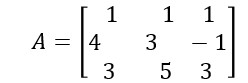
1

3

5

—------------------------------------------------------------------Question:

Find by Gauss-Jordan method:



Code:

1. = [1,1,1;4,3,-1;3,5,3];n = length(A(1,:)); Aug = [A,eye(n,n)] for j = 1:n-1 for i = j+1:n

Aug(i,j:2\*n) = Aug(i,j:2\*n) - Aug(i,j)/Aug(j,j)\*Aug(j,j:2\*n) end

end

for j = n:-1:2

Aug(1:j-1,:) = Aug(1:j-1,:) - Aug(1:j-1,j)/Aug(j,j)\*Aug(j,:) end

for j = 1:n

Aug(j,:) = Aug(j,:)/Aug(j,j) end

1. = Aug(:,n+1:2\*n)Output:

Aug =

1 1 1 1 0 0

4 3 -1 0 1 0

3 5 3 0 0 1

Aug =

1 1 1 1 0 0

* 1. -1 -5 -4 1 0

3 5 3 0 0 1

Aug =

* 1. 1 1 1 0 0

0 -1 -5 -4 1 0

* 1. 2 0 -3 0 1

Aug =

* 1. 1 1 1 0 0

0 -1 -5 -4 1 0

0 0 -10 -11 2 1

Aug =

Columns 1 through 5

1.0000 1.0000 0 -0.1000 0.2000

0 -1.0000 0 1.5000 0

0 0 -10.0000 -11.0000 2.0000 Column 6

0.1000

-0.5000

1.0000

Aug =

Columns 1 through 5

1.0000 0 0 1.4000 0.2000

0 -1.0000 0 1.5000 0

0 0 -10.0000 -11.0000 2.0000

Column 6

-0.4000

-0.5000

1.0000

Aug =

Columns 1 through 5

1.0000 0 0 1.4000 0.2000

0 -1.0000 0 1.5000 0

0 0 -10.0000 -11.0000 2.0000

Column 6

-0.4000

-0.5000

1.0000

Aug =

Columns 1 through 5

1.0000 0 0 1.4000 0.2000

0 1.0000 0 -1.5000 0

0 0 -10.0000 -11.0000 2.0000

Column 6

-0.4000

0.5000

1.0000

Aug =

Columns 1 through 5

1.0000 0 0 1.4000 0.2000

0 1.0000 0 -1.5000 0

0 0 1.0000 1.1000 -0.2000

Column 6

-0.4000

0.5000

-0.1000

B =

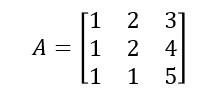
1.4000 0.2000 -0.4000

-1.5000 0 0.5000

1.1000 -0.2000 -0.1000

—------------------------------------------------------------------Practice Problems:

Question:



Output:

Aug =

1 2 3 1 0 0

1 2 4 0 1 0

1 1 5 0 0 1

Aug =

1 2 3 1 0 0

* 1. 0 1 -1 1 0
  2. 1 5 0 0 1

Aug =

1 2 3 1 0 0

0 0 1 -1 1 0

* 1. -1 2 -1 0 1

Aug =

* 1. 2 3 1 0 0

0 0 1 -1 1 0

* 1. NaN Inf -Inf Inf NaN

Aug =

* 1. NaN NaN NaN NaN NaN

0 NaN NaN NaN NaN NaN

0 NaN Inf -Inf Inf NaN

Aug =

NaN NaN NaN NaN NaN NaN

0 NaN NaN NaN NaN NaN

0 NaN Inf -Inf Inf NaN

Aug =

NaN NaN NaN NaN NaN NaN

0 NaN NaN NaN NaN NaN

0 NaN Inf -Inf Inf NaN

Aug =

NaN NaN NaN NaN NaN NaN

NaN NaN NaN NaN NaN NaN

0 NaN Inf -Inf Inf NaN

Aug =

NaN NaN NaN NaN NaN NaN

NaN NaN NaN NaN NaN NaN

0 NaN NaN NaN NaN NaN

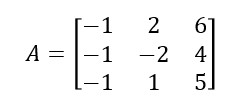
B =

NaN NaN NaN

NaN NaN NaN

NaN NaN NaN

—------------------------------------------------------------------Question:



Output:

Aug =

-1 2 6 1 0 0

-1 -2 4 0 1 0

-1 1 5 0 0 1

Aug =

-1 2 6 1 0 0

1. -4 -2 -1 1 0
2. 1 5 0 0 1

Aug =

-1 2 6 1 0 0

0 -4 -2 -1 1 0

0 -1 -1 -1 0 1

Aug =

-1.0000 2.0000 6.0000 1.0000 0 0

0 -4.0000 -2.0000 -1.0000 1.0000 0

0 0 -0.5000 -0.7500 -0.2500 1.0000

Aug =

-1.0000 2.0000 0 -8.0000 -3.0000 12.0000

0 -4.0000 0 2.0000 2.0000 -4.0000

0 0 -0.5000 -0.7500 -0.2500 1.0000

Aug =

-1.0000 0 0 -7.0000 -2.0000 10.0000

0 -4.0000 0 2.0000 2.0000 -4.0000

0 0 -0.5000 -0.7500 -0.2500 1.0000

Aug =

1.0000 0 0 7.0000 2.0000 -10.0000

0 -4.0000 0 2.0000 2.0000 -4.0000

0 0 -0.5000 -0.7500 -0.2500 1.0000

Aug =

1.0000 0 0 7.0000 2.0000 -10.0000

0 1.0000 0 -0.5000 -0.5000 1.0000

0 0 -0.5000 -0.7500 -0.2500 1.0000

Aug =

1.0000 0 0 7.0000 2.0000 -10.0000

0 1.0000 0 -0.5000 -0.5000 1.0000

0 0 1.0000 1.5000 0.5000 -2.0000

B =

7.0000 2.0000 -10.0000

-0.5000 -0.5000 1.0000

1.5000 0.5000 -2.0000

—------------------------------------------------------------------Question:

Find LU decomposition of A = [1, 1, -1; 3, 5, 6; 7, 8, 9] Code:

Ab = [1 1 -1;3 5 6;7 8 9]; n= length(A); L = eye(n); for i =2:3 alpha = Ab(i,1)/Ab(1,1); L(i,1) = alpha;

Ab(i,:) = Ab(i,:) -alpha\*Ab(1,:); end i=3;

alpha = Ab(i,2)/Ab(2,2);

L(i,2) = alpha

Ab(i,:) = Ab(i,:) -alpha\*Ab(2,:); U = Ab(1:n,1:n) Output:

L =

1.0000 0 0

3.0000 1.0000 0

7.0000 0.5000 1.0000

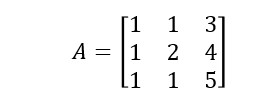
U =

1.0000 1.0000 -1.0000

0 2.0000 9.0000

0 0 11.5000

—------------------------------------------------------------------Practice Problems:



Output:

L =

1 0 0

1 1 0

1 0 1

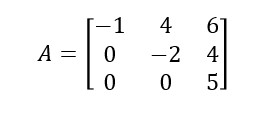
U =

1 1 3

0 1 1

0 0 2

—-------------------------------------------------------------------



Output:

L =

1 0 0

0 1 0

0 0 1

U =

-1 4 6

0 -2 4

0 0 5

—-------------------------------------------------------------------

**Unit 2:**

Find the fundamental subspaces for the matrix A = [2,3,4;4,3,8;1,3,2] Code:

A=[2,3,4;4,3,8;1,3,2] [V,pivot] = rref(A) r = length(pivot) colspace = A(:,pivot) nullspace = null(A,'r') rowspace = V(1:r,:)' leftns = null(A','r')

Output: A =

2 3 4

4 3 8

1 3 2

V =

1 0 2

1. 1 0
2. 0 0

pivot =

1. 2 r =
2. colspace =

2 3

4 3

1. 3

nullspace =

-2

0

1

rowspace = 1 0

0 1

1. 0 leftns =

-1.5000

0.5000

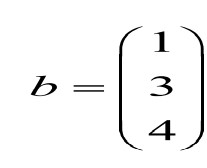
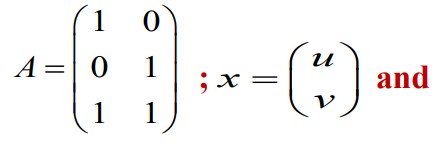
1.0000

—-------------------------------------------------------------------

**Unit 3:**

Question:

Find the projection of the matrix



Code:

A=[1,0;0,1;1,1] b=[1;3;4] x = lsqr(A,b) Output:

A =

1 0

1. 1
2. 1

b =

1

3

4

lsqr converged at iteration 2 to a solution with relative residual 6.7e-17.

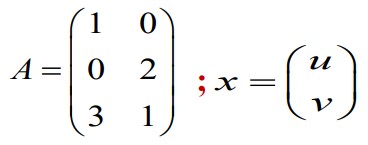
x =

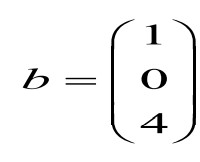
1.0000

3.0000

—------------------------------------------------------------------Question:

Find the projection for the matrix





Code:

A=[1,0;0,2;3,1] b=[1;0;4] x = lsqr(A,b) Output:

A =

1 0

0 2

3 1

b =

1

0

4

lsqr converged at iteration 2 to a solution with relative residual 0.076.

x =

1.2927

0.0244

—-------------------------------------------------------------------

Question:

Find a point on the plane x+y+z = 0 that is closest to

(2,1,0) Code:

syms c

P=[2,1,0]+c\*[1,1,-1] s=1\*(c+2)+1\*(c+1)-1\*(-c)==0 s1=solve(s,c) p=[2,1,0]+s1\*[1,1,-1]

Output:

P =

[c + 2, c + 1, -c]

s = 3\*c + 3 == 0 s1 = -1 p =

[1, 0, 1]

—-------------------------------------------------------------------

Question:

Find a point on the plane 3x+4y+z=1 that is closest to

(1,0,1) Code:

syms c

P=[1,0,1]+c\*[3,4,1] s=3\*(1+3\*c)+4\*(4\*c)+(1+c)==1 s1=solve(s,c) p=[1,0,1]+s1\*[3,4,1]

Output:

P =

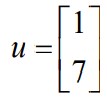
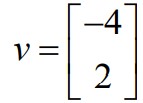
[3\*c + 1, 4\*c, c + 1] s =

26\*c + 4 == 1 s1 = -3/26

p =

[17/26, -6/13, 23/26]

—------------------------------------------------------------------Question:

Let  onto  and find P, the matrix

that will project any matrix onto the vector v. Use the result to find projection of v on u

Code:

u=[1;7] v=[-4;2]

P=(v\*transpose(v))/(transpose(v)\*v) P\*u Output: u =

1

7

v =

-4

2

P =

0.8000 -0.4000

-0.4000 0.2000 ans =

-2

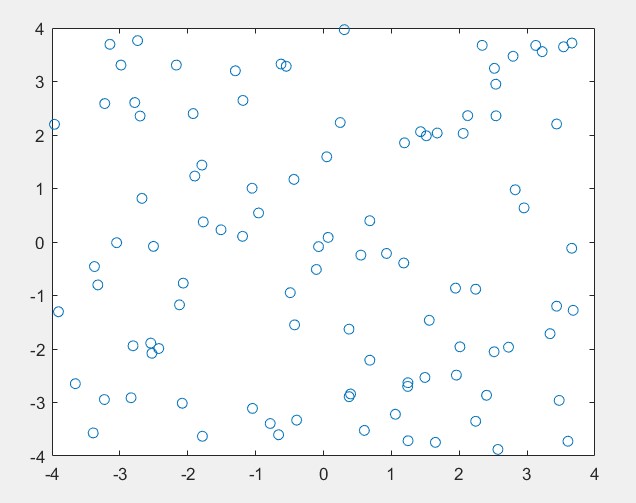
1

—------------------------------------------------------------------Question:

Projecting a lot of vectors on a single vector Code:

u=8\*rand(2,100)-4; x=u(1,: ) y=u(2,: ) plot(x,y,'o')

Output:



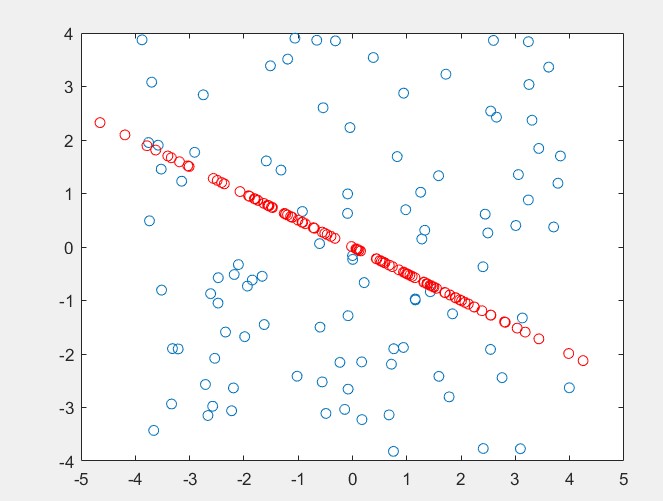
—------------------------------------------------------------------Code 2:

P=[0.8,-0.4;-0.4,0.2]

Pu=P\*u; x=Pu(1,:) y=Pu(2,:) hold on

plot(x,y,'ro')

Output:



—-------------------------------------------------------------------

Question:

Find the least square fit for the system

x + 2y = 3 3x + 2y = 5 x + y = 2.09

Code:

A=[1,2;3,2;1,1] b=[3;5;2.09] x = lsqr(A,b) Output:

A =

1 2

3 2

1 1

b =

3.0000

5.0000

2.0900

lsqr converged at iteration 2 to a solution with relative residual 0.014.

x =

1.0000

1.0100

—-------------------------------------------------------------------

Question:

Find the point on the plane 13x+4y+z=1 that is closest to

(1,-1,1) Code:

syms c

P=[1,-1,1]+c\*[13,4,1] s=13\*(13\*c+1)+4\*(4\*c-1)+1\*(c+1)==0 s1=solve(s,c) p=[1,-1,1]+s1\*[13,4,1]

Output:

P =

[13\*c + 1, 4\*c - 1, c + 1]

s =

186\*c + 10 == 0 s1 = -5/93 p =

[28/93, -113/93, 88/93]

—-------------------------------------------------------------------

Question:

Find the least square fit for this system x + 2y + z = 3

3x + 2y - 2z = 5 x + y + 7z = 21.09

Code:

A=[1,2,1;3,2,-2;1,1,7] b=[3;5;21.09] x = lsqr(A,b)

Output:

A =

1 2 1

3 2 -2

1 1 7

b =

3.0000

5.0000

21.0900

lsqr converged at iteration 3 to a solution with relative residual 4.9e-15.

x =

4.9497

-2.2914

2.6331

—------------------------------------------------------------------**Unit 4:**

Question:

Apply the Gram-Schmidt process to the vectors(1,0,1), (1,0,0) and (2,1,0) to produce a set of orthonormal vectors Code:

A=[1,1,2;0,0,1;1,0,0]

Q=zeros(3) R=zeros(3) for j=1:3 v=A(: , j)

for i=1:j-1

R(i,j)=Q(:,i)'\*A(:,j) v=v-R(i,j)\*Q(:,i)

end

R(j,j)=norm(v)

Q(:,j)=v/R(j,j) end

Output:

A =

1 1 2

1. 0 1
2. 0 0
3. =
   1. 0 0

0 0 0

0 0 0

1. =
   1. 0 0

0 0 0

0 0 0

v =

1

0

1

R =

1.4142 0 0

0 0 0

0 0 0

Q =

0.7071 0 0 0 0 0

0.7071 0 0

v =

1

0

0

R =

1.4142 0.7071 0

0 0 0

0 0 0

v =

0.5000

0

-0.5000

R =

1.4142 0.7071 0

0 0.7071 0

0 0 0

1. =

0.7071 0.7071 0

0 0 0

0.7071 -0.7071 0 v =

2

1

0

1. =

1.4142 0.7071 1.4142

0 0.7071 0

0 0 0

v =

1.0000

1.0000

-1.0000

R =

1.4142 0.7071 1.4142

0 0.7071 1.4142

0 0 0

v =

-0.0000

1.0000

0.0000

R =

1.4142 0.7071 1.4142

0 0.7071 1.4142

0 0 1.0000

Q =

0.7071 0.7071 -0.0000 0 0 1.0000

0.7071 -0.7071 0.0000

—-------------------------------------------------------------------

Question:

Apply the Gram-Schmidt process to the vectors a = (0,1,1,1), b = (1,1,-1,0) and c = (1,0,2,-1) Code:

A=[0,1,1;1,1,0;1,-1,2;1,0,-1]

Q=zeros(4,3) R=zeros(3) for j=1:3 v=A(: , j)

for i=1:j-1

R(i,j)=Q(:,i)'\*A(:,j)

v=v-R(i,j)\*Q(:,i)

end

R(j,j)=norm(v)

Q(:,j)=v/R(j,j) end

Output:

A =

1. 1 1
2. 1 0

1 -1 2

1 0 -1

1. =
   1. 0 0

0 0 0

0 0 0

0 0 0

1. =
   1. 0 0

0 0 0

0 0 0

v =

0

1

1

1

R =

1.7321 0 0

0 0 0

0 0 0

Q =

0 0 0

0.5774 0 0

0.5774 0 0

0.5774 0 0

v =

1

1

-1

0

R =

1.7321 0 0

0 0 0

0 0 0

v =

1

1

-1

0

R =

1.7321 0 0

0 1.7321 0

0 0 0

Q =

0 0.5774 0

0.5774 0.5774 0

0.5774 -0.5774 0

0.5774 0 0

v =

1

0

2

-1

R =

1.7321 0 0.5774

0 1.7321 0

0 0 0

v =

1.0000

-0.3333

1.6667

-1.3333

R =

1.7321 0 0.5774

0 1.7321 -0.5774

0 0 0

v =

1.3333 0

1.3333

-1.3333

R =

1.7321 0 0.5774

0 1.7321 -0.5774

0 0 2.3094

Q =

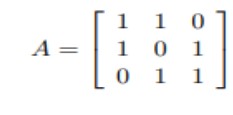
0 0.5774 0.5774

0.5774 0.5774 0

0.5774 -0.5774 0.5774

0.5774 0 -0.5774

—------------------------------------------------------------------Question:

Find QR factorization of the matrix Code:

A=[1,1,0;1,0,1;0,1,1]; [Q,R]=qr(A) Output:

Q =

-0.7071 0.4082 -0.5774

-0.7071 -0.4082 0.5774

0 0.8165 0.5774 R =

-1.4142 -0.7071 -0.7071

0 1.2247 0.4082

0 0 1.1547

—------------------------------------------------------------------Question:

QR factorization of Pascal Matrix Code:

A = sym(pascal(3)) [Q,R] = qr(A) isAlways(A == Q\*R) Output:

A =

[1, 1, 1]

[1, 2, 3]

[1, 3, 6]

Q =

[3^(1/2)/3, -2^(1/2)/2, 6^(1/2)/6]

[3^(1/2)/3, 0, -6^(1/2)/3]

[3^(1/2)/3, 2^(1/2)/2, 6^(1/2)/6] R =

[3^(1/2), 2\*3^(1/2), (10\*3^(1/2))/3]

[ 0, 2^(1/2), (5\*2^(1/2))/2]

[ 0, 0, 6^(1/2)/6]

ans =

3×3 logical array

1 1 1

1 1 1

1 1 1

—-------------------------------------------------------------------

Question:

QR decomposition to solve matrix equation of the form

Ax = b

Code:

A = sym(invhilb(5)) b = sym([1:5]')

[C,R] = qr(A,b); X = R\C isAlways(A\*X == b)

Output:

A =

[ 25, -300, 1050, -1400, 630]

[ -300, 4800, -18900, 26880, -12600]

[ 1050, -18900, 79380, -117600, 56700]

[-1400, 26880, -117600, 179200, -88200] [ 630, -12600, 56700, -88200, 44100] b =

1

2

3

4

5

X =

5

71/20

197/70

657/280

1271/630

ans =

5×1 logical array

1

1

1

1

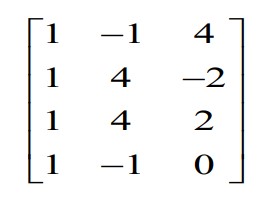
1

—------------------------------------------------------------------Practice Problems:

Question:

Find QR decomposition (Gram Schmidt Method) using

MATLAB command



Code:

A = [1,-1 4;1 4 -2;1 4 2;1 -1 0] b = sym([1:4]') [C,R] = qr(A,b); X = R\C isAlways(A\*X == b) Output:

A =

1 -1 4

1 4 -2

1 4 2

1 -1 0

b =

1

2

3

4

Warning: Solution does not exist because the system is inconsistent.

> In symengine

In sym/privBinaryOp (line 1136)

In \ (line 497)

In QR\_decomposition (line 4)

X =

Inf

Inf Inf ans =

4×1 logical array

0

0

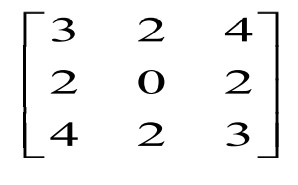
0

0

—------------------------------------------------------------------Question:

Find QR decomposition (Gram Schmidt Method) using

MATLAB command



Output:

A =

1. 2 4

2 0 2

1. 2 3

b =

1

2

3

X =

3/2

-3/4 -1/2 ans =

3×1 logical array

1

1

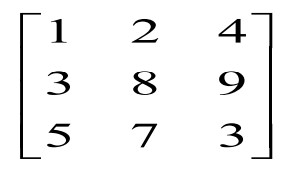
1

—-------------------------------------------------------------------

Question:

Find QR decomposition (Gram Schmidt Method) using

MATLAB command



Output:

A =

1 2 4

3 8 9

5 7 3

b =

1

2

3

X =

37/43

-11/43

7/43 ans =

3×1 logical array 1

1

1