Department of Computer Scince and Engineering

Scilab

LINEAR ALGEBRA AND ITS APPLICATIONS -UE19MA251

Session: Jan-May 2021

NAME OF THE STUDENT : SUJAY S AMBEKAR

SRN : PES1UG19CS516

BRANCH : COMPUETR SCIENCE AND ENGINEERING

SEMESTER & SECTION : IV ‘H2’

INDEX

Sl.No.	TOPIC	Page No.
1	Gaussian Elimination	3
2	The LU Decomposition	5
3	Inverse of a Matrix by the Gauss- Jordan Method	6
4	The Span of Column Space of a Matrix	7
5	The Four Fundamental Subspaces	9
6	Projections by Least Squares	11
7	The Gram-Schmidt Orthogonalization	12
8	Eigen values and Eigen Vectors of a Matrix	13

PROBLEM 1

Solve the following system of equations by Gaussian Elimination. Identify the pivots in each case.

$2x + 5y + z = 0$, $4x + 8y + z = 2$, $y - z = 3$

Code:

P1 - Gaussian Elimination.sce (C:\Users\Sujay\Desktop\LA\Linear-Algebra-Scilab\P1 - Gaussian Elimination.sce) - SciNotes

P1 - Gaussian Elimination.sce

```
clc;clear;
A = [2,5,1;4,8,1;0,1,-1], b = [0;2;3];
disp("Matrix before Gaussian Elimination: ")
disp(A);
Ab = [A b];
a = Ab;
n = 3;
for i = 2:n
    for j=2:n+1
        a(i,j) = a(i,j) - a(1,j)*a(i,1)/a(1,1);
    end
    a(i,1) = 0;
end
for i=3:n
    for j=3:n+1
        a(i,j) = a(i,j)-a(2,j)*a(i,2)/a(2,2);
    end
    a(i,2) = 0;
end

x(n) = a(n,n+1)/a(n,n);
for i=n-1:-1:1
    sumk = 0;
    for k=i+1:n
        sumk = sumk+a(i,k)*x(k);
    end
    x(i) = (a(i,n+1) - sumk)/a(i,i);
end

disp("Values of x,y,z:")
disp(x);
```

```
for i=n-1:-1:1
    sumk = 0;
    for k=i+1:n
        sumk = sumk+a(i,k)*x(k);
    end
    x(i) = (a(i,n+1) - sumk)/a(i,i);
end

disp("Values of x,y,z:")
disp(x);
disp("Matrix after Gaussian Elimination: ")
disp(a);
disp("The pivots are: ");
disp(a(3,3),a(2,2),a(1,1));

```

Output:

```
Scilab 6.0.2 Console

Matrix before Gaussian Elimination:

   2.   5.   1.
   4.   8.   1.
   0.   1.  -1.

Values of x,y,z:

   0.5
   0.3333333
  -2.6666667

Matrix after Gaussian Elimination:

   2.   5.   1.    0.
   0.  -2.  -1.    2.
   0.   0.  -1.5   4.

The pivots are:

   2.

  -2.

  -1.5

-->
```

PROBLEM 2

Factorize the following matrix as A = LU

A=[2 , 3 , 1 ; 4 , 7 , 5 ; 1 , -2 , 2]

Code:

P2 - LU Decomposition.sce (C:\Users\Sujay\Desktop\LA\Linear-Algebra-Scilab\P2 - LU Decomposition.sce) - SciNotes

P2 - LU Decomposition.sce

```
clear;clc;;
A = [2 3 1;4 7 5; 1 -2 2];
U = A;
disp(A, "The given matrix is:");
m = det(U(1,1));
n = det(U(2,1));
a = n/m;
U(2,:)= U(2,:) - U(1,:)/(m/n);
n = det(U(3,1));
b = n/m;
U(3,:) = U(3,:) - U(1,:)/(m/n);
m = det(U(2,2));
n = det(U(3,2));
c = n/m;
U(3,:) = U(3,:) - U(2,:)/(m/n);
disp(U,"The upper triangular matrix is:");
L = [1,0,0;a,1,0;b,c,1];
disp(L,"The lower triangular matrix is:");
```

Output:

Scilab 6.0.2 Console

```
 The given matrix is:

   2.   3.   1.
   4.   7.   5.
   1.  -2.   2.

 The upper triangular matrix is:

   2.   3.   1.
   0.   1.   3.
   0.   0.   12.

 The lower triangular matrix is:

   1.     0.     0.
   2.     1.     0.
   0.5   -3.5    1.

-->
```

PROBLEM 3

Find the inverse of the matrix

A = [1, 0, 0; 1, 1, 1; 0, 0, 1] by Gauss – Jordan method.

Code:

P3 - Gauss Jordan Inverse.sce (C:\Users\Sujay\Desktop\LA\Linear-Algebra-Scilab\P3 - Gauss Jordan Inverse.sce) - SciNotes

P3 - Gauss Jordan Inverse.sce

```
clc;clear;
A = [1 0 0;1 1 1;0 0 1];
n = length(A(1,:));
Aug = [A,eye(n,n)];
//Forward Elimination
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
//Backward Elimination
for j = n:-1:2
    Aug(1:j-1,:) = Aug(1:j-1,:)-Aug(1:j-1,j)/Aug(j,j)*Aug(j,:);
end
//Diagonal Normalization
for j=1:n
    Aug(j,:) = Aug(j,:)/Aug(j,j);
end
B = Aug(:,n+1:2*n);
disp("The inverse of A is:");
disp(B);
```

Output:

Scilab 6.0.2 Console

```
 The inverse of A is:

   1.   0.   0.
  -1.   1.  -1.
   0.   0.   1.

-->
```

PROBLEM 4

Identify the columns that are in the column space of A where A = [2, 4, 6, 4; 2, 5, 7, 6; 2, 3, 5, 2]

[

Code:

P4 - Span of Column Space.sce (C:\Users\Sujay\Desktop\LA\Linear-Algebra-Scilab\P4 - Span of Column Space.sce) - SciNotes

P4 - Span of Column Space.sce

```
clc;clear;
a = [2 4 6 4; 2 5 7 6; 2 3 5 2];
disp("The given matrix is:");
disp(a);
a(2,:) = a(2,:)-(a(2,1)/a(1,1))*a(1,:);
a(3,:) = a(3,:)-(a(3,1)/a(1,1))*a(1,:);
disp(a);
a(3,:) = a(3,:)-(a(3,2)/a(2,2))*a(2,:);
disp(a);
a(1,:) = a(1,:)/a(1,1);
a(2,:) = a(2,:)/a(2,2);
disp(a);
for i=1:3
    for j=i:4
        if(a(i,j)<>0)
            disp("is a pivot column",j,"column");
            break;
        end
    end
end
```

Output:

```
Scilab 6.0.2 Console

The given matrix is:


   2.   4.   6.   4.
   2.   5.   7.   6.
   2.   3.   5.   2.

   2.   4.   6.   4.
   0.   1.   1.   2.
   0.  -1.  -1.  -2.

   2.   4.   6.   4.
   0.   1.   1.   2.
   0.   0.   0.   0.

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

column

   1.

is a pivot column

column

   2.

is a pivot column
```

PROBLEM 5

Find the four fundamental subspaces of

A = [1, 3, 3, 2; 2, 6, 9, 7; -1, -3, 3, 4]

Code:

P5 - Fundamental Subspaces.sce (C:\Users\Sujay\Desktop\LA\Linear-Algebra-Scilab\P5 - Fundamental Subspaces.sce) - SciNotes

P5 - Fundamental Subspaces.sce

```
clc;clear;
A = [1 3 3 2;2 6 9 7;-1 -3 3 4];
disp("The given matrix is:");
disp(A);
[m,n] = size(A);
disp(m,"m = ");
disp(n,"n = ");
[v,pivot] = rref(A);
disp(rref(A),"Row Reduced Echelon Form: ");
r = length(pivot);
disp(r,"Rank: ");
colspace = A(:,pivot);
disp(colspace,"Column Space: ");
nullspace = kernel(A);
disp(nullspace,"Null Space: ");
rowspace = v(1:r,:)';
disp(rowspace,"Row Space: ");
leftnullspace = kernel(A');
disp(leftnullspace,"Left Null Space: ");
```

Output:

```
Scilab 6.0.2 Console

The given matrix is:

  1.   3.   3.   2.
  2.   6.   9.   7.
 -1.  -3.   3.   4.

m =

  3.

n =

  4.

Row Reduced Echelon Form:

  1.   3.   0.  -1.
  0.   0.   1.   1.
  0.   0.   0.   0.

Rank:

  2.

Column Space:

  1.   3.
  2.   9.
 -1.   3.
```

```
Null Space:

 -0.0160107   0.951055
  0.2344393  -0.2964715
 -0.687307   -0.0616403
  0.687307    0.0616403

Row Space:

  1.   0.
  3.   0.
  0.   1.
 -1.   1.

Left Null Space:

  0.9128709
 -0.3651484
  0.1825742
```

PROBLEM 6

Solve Ax = b by least squares where

$$A=\begin{pmatrix}1 & 0\\ 0 & 1\\ 1 & 1\end{pmatrix} \quad b=\begin{pmatrix}1\\ 1\\ 0\end{pmatrix}$$

Code:

P6 - Least Squares Projection.sce (C:\Users\Sujay\Desktop\LA\Linear-Algebra-Scilab\P6 - Least Squares Projection.sce) - SciNotes

P6 - Least Squares Projection.sce

```
clc;clear;
A = [1 0;0 1;1 1];
b = [1;1;0];
disp("The given matrix A is:")
disp(A);
disp(b, "b: ");
x = (A'*A)\(A'*b)
C = x(1,1);
D = x(2,1);
disp(C,"C: ");
disp(D,"D: ");
disp("The best fit line is b = C+Dt")

```

Output:

Scilab 6.0.2 Console

```
The given matrix A is:

  1.   0.
  0.   1.
  1.   1.

b:

  1.
  1.
  0.

C:

  0.3333333

D:

  0.3333333

The best fit line is b = C+Dt

-->
```

PROBLEM 7

Apply the Gram – Schmidt process to the vectors (1, 1, 0) , (1, 0, 1,) and (0, 1, 1) to produce a set of orthonormal vectors.

Code:

P7 - Gram Schmidt Process.sce (C:\Users\Sujay\Desktop\LA\Linear-Algebra-Scilab\P7 - Gram Schmidt Process.sce) - SciNotes

P7 - Gram Schmidt Process.sce

```
clc;clear;
A= [1 1 0;1 0 1;0 1 1];
disp(A, "The given matrix A is:");
[m,n] = size(A);
for k=1:n
    V(:,k) = A(:,k);
    for j=1:k-1
        R(j,k) = V(:,j)'*A(:,k);
        V(:,k) = V(:,k)-R(j,k)*V(:,j);
    end
    R(k,k) = norm(V(:,k));
    V(:,k) = V(:,k)/R(k,k);
end
disp(V,"Q: ");
```

Output:

Scilab 6.0.2 Console

```
 The given matrix A is:

   1.   1.   0.
   1.   0.   1.
   0.   1.   1.

 Q:

   0.7071068   0.4082483  -0.5773503
   0.7071068  -0.4082483   0.5773503
   0.          0.8164966   0.5773503

-->
```

PROBLEM 8

Find the Eigen values and the corresponding Eigen vectors of

A = [2, 2, 1; 1, 3, 1; 1, 2, 2]

Code:

P8 - Eigen Values and Eigen Vectors.sce (C:\Users\Sujay\Desktop\LA\Linear-Algebra-Scilab\P8 - Eigen Values and Eigen Vectors.sce) - SciNotes

P8 - Eigen Values and Eigen Vectors.sce

```
clc;clear;
A = [2 2 1;1 3 1;1 2 2];
disp(A,"The given matrix A is: ")
lam = poly(0,"lam");
charMat = A-lam*eye(3,3);
disp(charMat,"The Characteristic Matrix is: ");
charPoly = poly(A,"lam");
disp(charPoly,"The Characteristic Polynomial is:");
lam = spec(A);
disp(lam,"Eigen  Values: ");
function[x,lam] = eigenvectors(A)
    [n,m] = size(A);
    lam = spec(A)';
    x = [];
    for k=1:3
        B = A-lam(k)*eye(3,3);
        C = B(1:n-1,1:n-1);
        b = -B(1:n-1,n);
        y = C\b;
        y = [y;1];
        y = y/norm(y);
        x = [x y];
    end
endfunction
[x,lam] = eigenvectors(A);
disp(x,"Eigen Vectors of A: ");
```

Output:

```
Scilab 6.0.2 Console

The given matrix A is:

  2.   2.   1.
  1.   3.   1.
  1.   2.   2.

The Characteristic Matrix is:


  2 -lam   2         1


  1        3 -lam    1


  1        2         2 -lam


The Characteristic Polynomial is:

                2      3
 -5 +11lam -7lam  +lam

Eigen  Values:

  1.
  5.
  1.
```

```
Eigen Vectors of A:

 -0.7071068   0.5773503  -0.7071068
  0.          0.5773503   1.570D-16
  0.7071068   0.5773503   0.7071068

-->
```