```
 1
 2 function[U,S,V]=SelfSVD(A)
 3 tol=10^-8;
 4 [eVec,eVal]=eig(A'*A);
 5 [p,q]=size(A);
 6
 7 val=min(p,q);
 8 sigma=eig(A'*A);
 9 sigma=sqrt(abs(sigma));
10 n=length(sigma);
11 for i=1:n
12     for j=i+1:n
13         if(sigma(i)<sigma(j))
14             temp=sigma(i);
15             sigma(i)=sigma(j);
16             sigma(j)=temp;
17             eVec(:,[i,j])=eVec(:,[j,i]);
18         end
19     end
20 end
21 S=zeros(p,q);
22 for i=1:val
23     if(sigma(i)>=tol)
24         S(i,i)=sigma(i);
25     end
26 end
27
28 U=zeros(p,p);
29 V=eVec;
30 V1=A*V;
31 for i=1:val
32     if(S(i,i)~=0)
33         U(:,i)=V1(:,i)/S(i,i);
34     end
35 end
36
37
```

```
%Here I am creating a function SelfLRA which is first doing trancuated svd
%and then giving Ak

function[Ak]=SelfLRA(A,k)
%First using a predefined function we calculated U,S,V and then we are
%using concept of trancuated svd for k rank
[U,S,V]=SelfSVD(A);
V_real=V.';
for i=1:k
    Vk(i,:)=V_real(i,:);
end
for i=1:k
    Uk(:,i)=U(:,i);
end
for i=1:k
    for j=1:k
        Sk(i,j)=S(i,j);
    end
end
Ak=Uk*Sk*Vk;


```

```
A=[1 2 3 4;4 5 6 7];
[U,S,V]=SelfSVD(A);
U
S
V
disp("By inbuilt svd")
[P,Q,R]=svd(A);
P
Q
R

```

```
>> A6Q3
>> A6Q1

U =

    0.4332    0.9013
    0.9013   -0.4332


S =

   12.4434         0         0         0
         0    1.0782         0         0


V =

    0.3245   -0.7711   -0.0872   -0.5407
    0.4318   -0.3370    0.5193    0.6560
    0.5390    0.0972   -0.7770    0.3103
    0.6463    0.5313    0.3449   -0.4255

By inbuilt svd

P =

   -0.4332    0.9013
   -0.9013   -0.4332


Q =

   12.4434         0         0         0
         0    1.0782         0         0


R =

   -0.3245   -0.7711   -0.4001   -0.3741
   -0.4318   -0.3370    0.2546    0.7970
   -0.5390    0.0972    0.6910   -0.4717
   -0.6463    0.5313   -0.5455    0.0488

>>
```

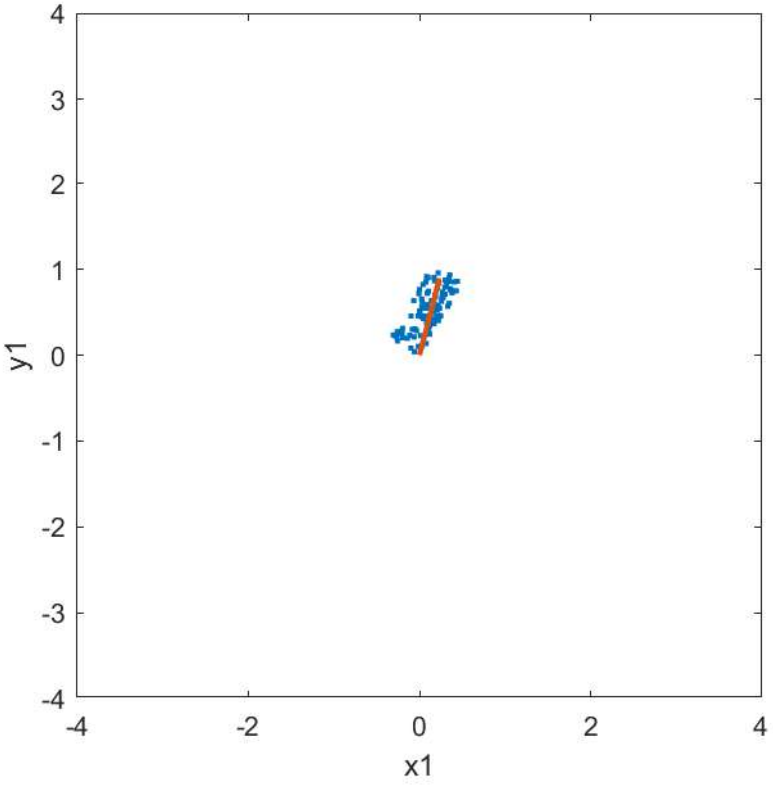
```
A=[1,0,0;0,1,0;0,0,1];
k=2;
[Ak]=SelfLRA(A,k)
Ak
disp("2-norm= ");
n1=norm(A-Ak);
n1
disp("Frobenius norm= ");
n2=norm(A-Ak,"fro");
n2
```

```
>> A6Q2

Ak =

     1     0     0
     0     1     0
     0     0     0


Ak =

     1     0     0
     0     1     0
     0     0     0

2-norm=

n1 =

     1

Frobenius norm=

n2 =

     1

>>
```

```
N=100;
x1=rand(1,N);
x2=0.4*rand(1,N);
A=[x1;x2];
Rotation = [cos(pi/3),-sin(pi/3);sin(pi/3),cos(pi/3)];
D=Rotation*A;
D
temp=D*D';
Q=(1/(N-1))*temp;
Q
[V,Di]=eig(Q);
sigma=eig(Q);
n=length(sigma);
for i=1:n
    for j=i+1:n
        if(sigma(i)<sigma(j))
            temp=sigma(i);
            sigma(i)=sigma(j);
            sigma(j)=temp;
            V(:,[i,j])=V(:,[j,i]);
        end
    end
end
plot(D(1,:),D(2,:),'.');
axis square
axis([-4 4 -4 4])

eigenVal=sigma;
u1=V(:,1);

hold on
quiver(0,0,u1(1),u1(2))
hold off
```

```
>> A6Q3

D =

  Columns 1 through 11

    0.0444   -0.0628    0.4951   -0.1800    0.0067    0.0354    0.1788   -0.0599 ↙
-0.2314   -0.0422    0.0144
    0.7032    0.4463    0.8654    0.3628    0.7495    0.6781    0.3439    0.1988 ↙
0.1626    0.5106    0.2043

  Columns 12 through 22

    0.2697   -0.0480    0.2554    0.1511    0.2036    0.0271    0.1913   -0.1330 ↙
0.1386    0.0745    0.1237
    0.6824    0.4553    0.8243    0.7606    0.5418    0.1886    0.9950    0.3832 ↙
0.9876    0.2154    0.3600

  Columns 23 through 33

    0.0094    0.1504    0.0234   -0.2878    0.3273    0.1584    0.1373   -0.0757 ↙
0.2056   -0.0512    0.3379
    0.0956    0.6523    0.1951    0.2182    0.6461    0.3097    0.6836    0.4869 ↙
0.6057    0.0545    0.8564

  Columns 34 through 44

    0.3275    0.1962    0.0926   -0.0262    0.2735    0.1524    0.2461   -0.0361 ↙
-0.0252   -0.1197   -0.1673
    0.7353    0.7480    0.8854    0.4577    0.5549    0.5766    0.4700    0.3385 ↙
0.3017    0.5906    0.3595

  Columns 45 through 55

    0.2340    0.1832   -0.0327    0.1327   -0.2777    0.0832    0.2474   -0.1807 ↙
0.3508   -0.3152    0.2987
    0.7938    1.0329    0.0535    0.5419    0.2609    0.8781    0.9993    0.1816 ↙
0.8822    0.2029    0.6172

  Columns 56 through 66

    0.1388    0.0431    0.1541    0.3115    0.0532   -0.2204   -0.0028   -0.1002 ↙
-0.3183   -0.1368    0.1938
    0.8249    0.5919    0.9334    0.8582    0.6920    0.2864    0.2531    0.2682 ↙
0.2320    0.2025    0.5999

  Columns 67 through 77

    0.2552    0.0523   -0.0567    0.3488    0.3022    0.0977    0.1572    0.2699 ↙
```

```
-0.1219    0.0752   -0.2085
    0.9377    0.3791    0.5069    0.9353    0.9174    0.7249    1.0504    0.7296 ↙
0.4592    0.7215    0.4023

  Columns 78 through 88

    0.1367    0.2165    0.0344    0.1083    0.2215    0.1289    0.0754    0.0713 ↙
0.0630    0.2394   -0.0795
    0.2623    0.6604    0.5897    0.4128    0.5680    0.7922    0.6303    0.5960 ↙
0.6375    0.4528    0.1413

  Columns 89 through 99

    0.2034    0.4146   -0.0704    0.1890    0.0757    0.1900    0.1796   -0.0831 ↙
0.0609   -0.1409    0.0276
    0.7134    0.9109    0.4500    1.0124    0.3563    0.9140    0.4213    0.5254 ↙
0.2163    0.2265    0.3408

  Column 100

    0.0836
    0.7902


Q =

    0.0349    0.0729
    0.0729    0.3757

>>
```


4
3
2
1
0
-1
-2
-3
-4
y1
-4
-2
0
2
4
x1