**Problem** **1:**

DATA 1

Y=X';

data(:,1)=Y(:,1)-mean(Y(:,1));

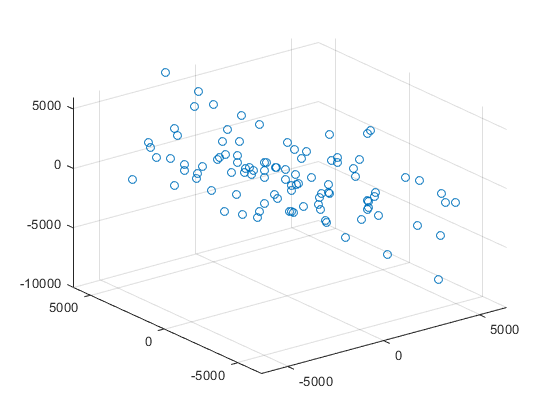
data(:,2)=Y(:,2)-mean(Y(:,2));

data(:,3)=Y(:,3)-mean(Y(:,3));

scatter plot of the Data

scatter3(X(1,:),X(2,:),X(3,:));

hold on;



PCA of the data upto 2 componets

pca(data,"NumComponents",2)

0.6277 0.3682  
 -0.5458 0.8364  
 -0.5550 -0.4061

Var=cov(data);

[U,S,V]=svd(Var);

center=mean(Y);

top two principal directions of variability overlaid onthe scatter plot of the data.

PD\_1=[center'+sqrt(S(1,1))\*U(:,1) , center'-sqrt(S(1,1))\*U(:,1)];

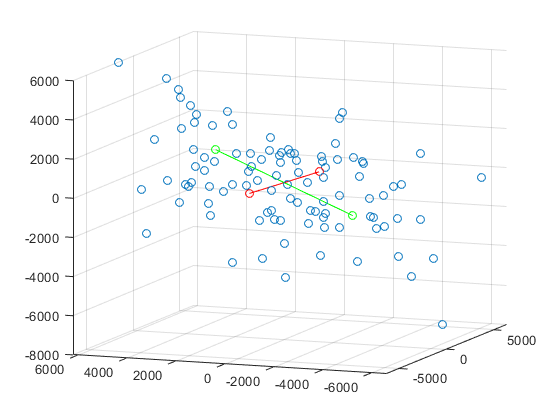
PD\_2=[center'+sqrt(S(2,2))\*U(:,2) , center'-sqrt(S(2,2))\*U(:,2)];

h1=plot3(PD\_1(1,:),PD\_1(2,:),PD\_1(3,:),'-o','Color','g');

h2=plot3(PD\_2(1,:),PD\_2(2,:),PD\_2(3,:),'-o','Color','r');

hold off;

view([-69 11])



U1 = U(:,1:2);

Z=U1'\*X;

Projection the original data to the first two principal dimensions

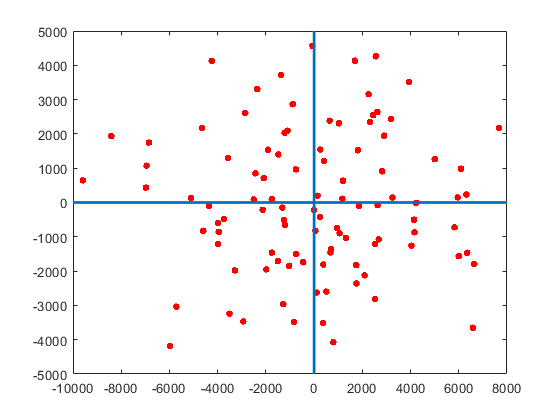
plot(Z(1,:), Z(2,:), '.', 'markersize', 15,"Color","r")

xL = xlim;

yL = ylim;

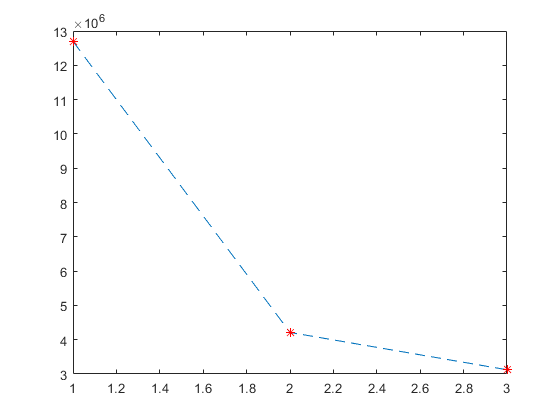
line([0 0], yL,"linewidth",2);

line(xL, [0 0],"linewidth",2);



singular value plot

plot(diag(S),"LineStyle","--","Marker","\*","MarkerEdgeColor","r")



Splot=cumsum(diag(S))/sum(diag(S));

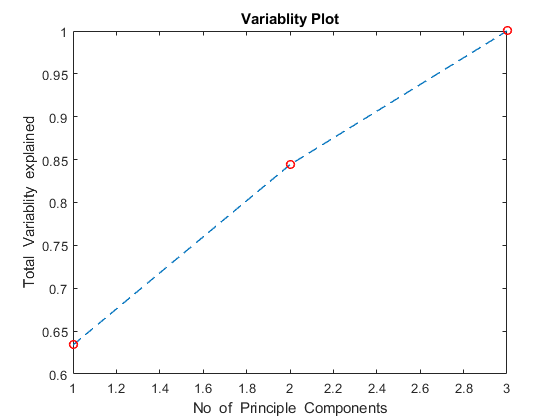
cumulative total variance plot

plot(Splot,"Marker","o","MarkerEdgeColor","r","LineWidth",1,"LineStyle","--")

title("Variablity Plot")

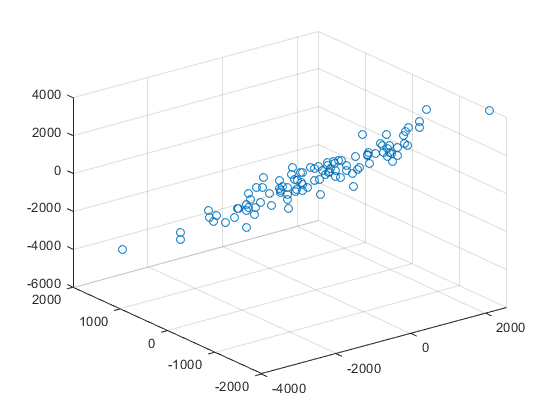
xlabel("No of Principle Components")

ylabel("Total Variablity explained")



**All 3 principal components are needed to obtain a total of 95% variability.**

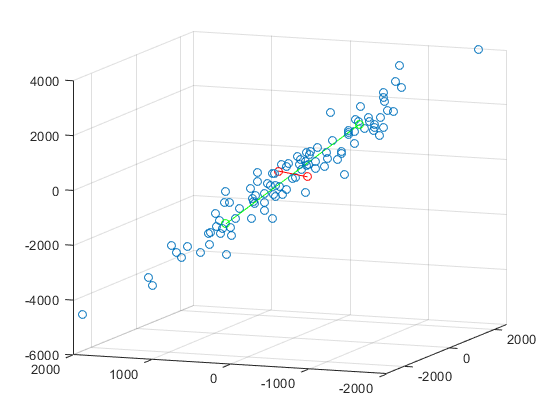
DATA 2



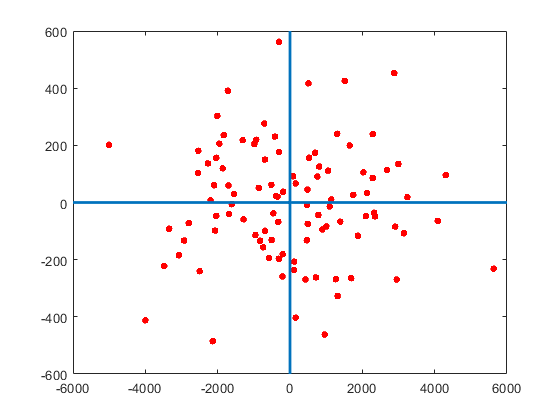
PCA of the data up to 2 components

0.4990 -0.1959  
 -0.2950 0.8775  
 0.8148 0.4377

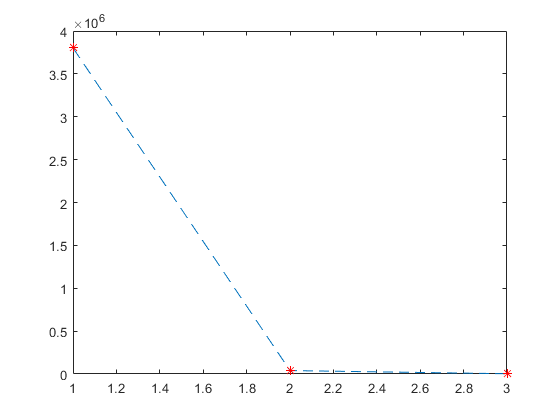
top two principal directions of variability overlaid on the scatter plot of the data.



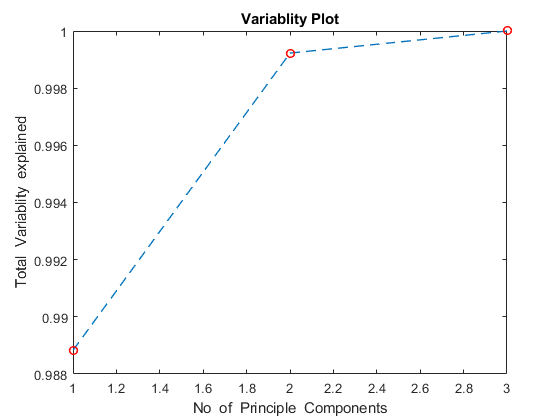
Projection the original data to the first two principal dimensions



singular value plot



cumulative total variance plot



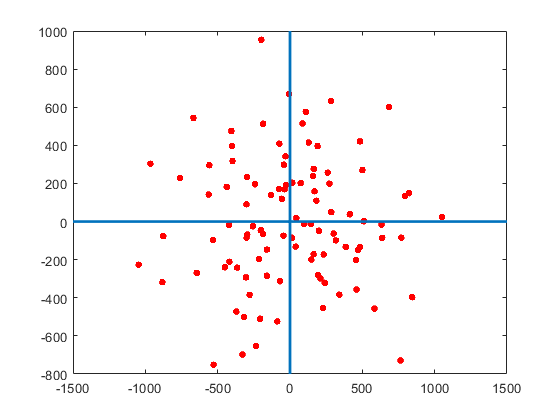
**Only 2 principal components are needed to obtain a total of 95% variability.**

Data 3

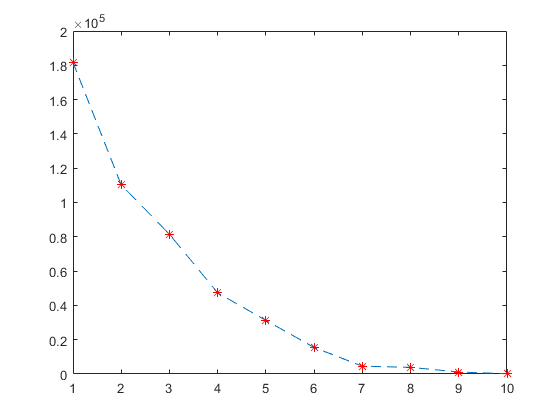
PCA of the data up to 2 components

0.0975 0.3257  
 -0.2693 -0.2891  
 0.4379 0.0955  
 0.0726 -0.1795  
 0.1976 0.5454  
 0.6304 -0.0079  
 -0.1281 -0.0883  
 -0.0183 -0.4561  
 -0.3631 0.3832  
 0.3686 -0.3299

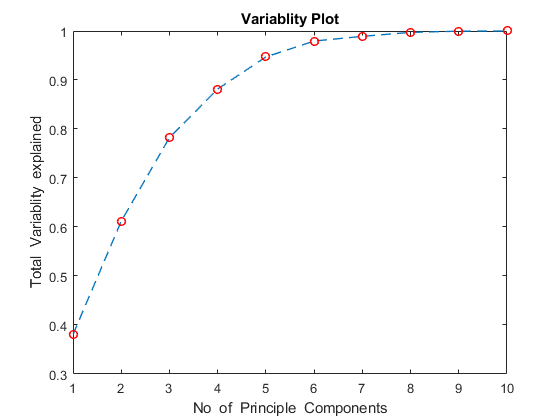
Projection the original data to the first two principal dimensions



singular value plot



cumulative total variance plot



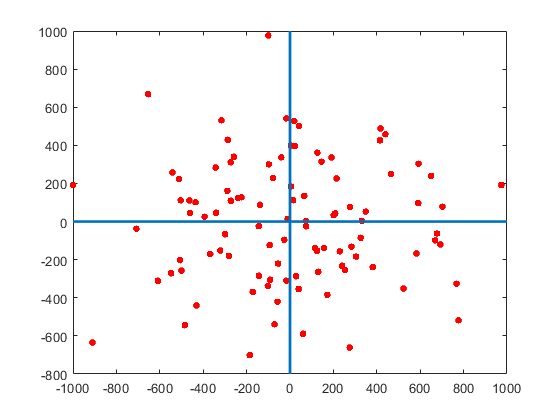
**Only 5 principal components are needed to obtain a total of 95% variability.**

Data 4

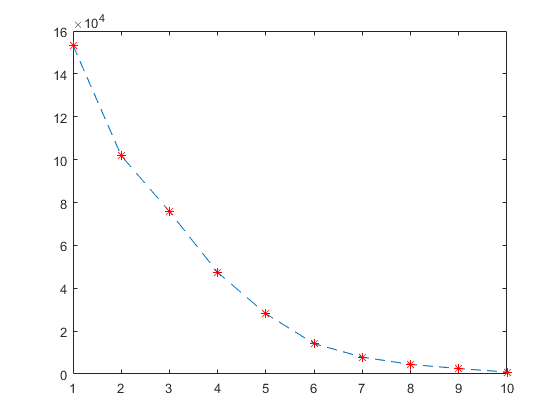
PCA of the data upto 2 componets

-0.2367 -0.4419  
 -0.3196 -0.3424  
 0.0863 -0.1074  
 -0.1825 0.1362  
 -0.1673 0.4473  
 0.1898 -0.0366  
 0.6393 0.0806  
 -0.2737 -0.3267  
 -0.4995 0.4284  
 0.0630 -0.3990

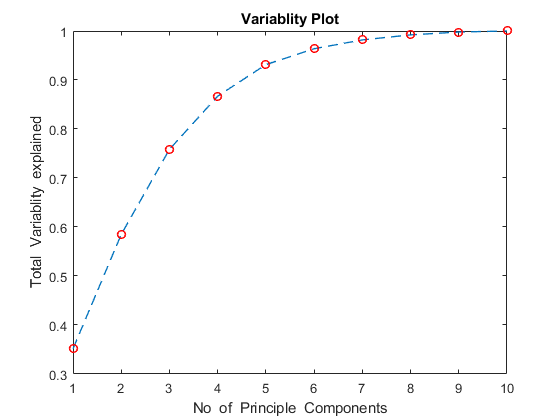
Projection the original data to the first two principal dimensions



singular value plot



cumulative total variance plot



**1st 6 principal components are needed to obtain a total of 95% variability.**

**Problem 2:**

Y=X';

for i=1:1:3584

data(:,i)=Y(:,i)-mean(Y(:,i));

end

M=mean(Y);

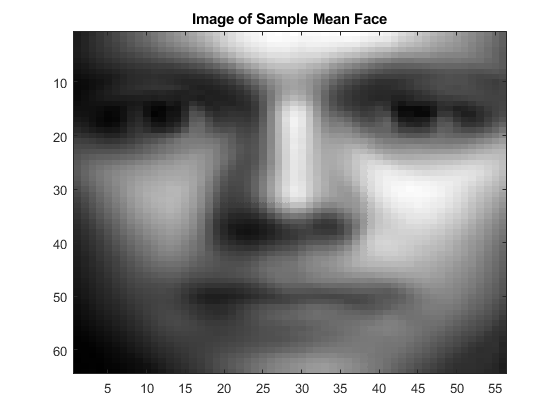
I=reshape(M,64,56);

%Image of Sample mean face

imagesc(I)

title('Image of Sample Mean Face');

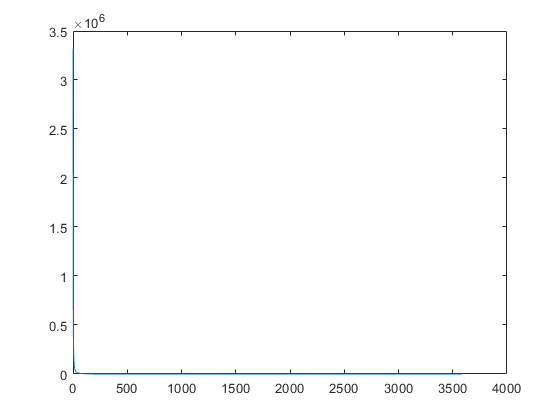
colormap(gray)



var=cov(data);

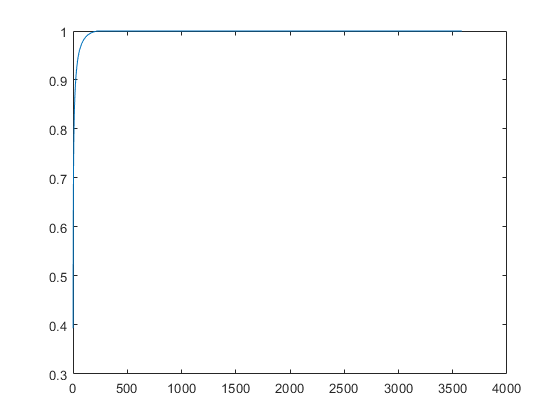
[U,S,V]=svd(var);

plot(diag(S));



Splot=cumsum(diag(S))/sum(diag(S));

plot(Splot)



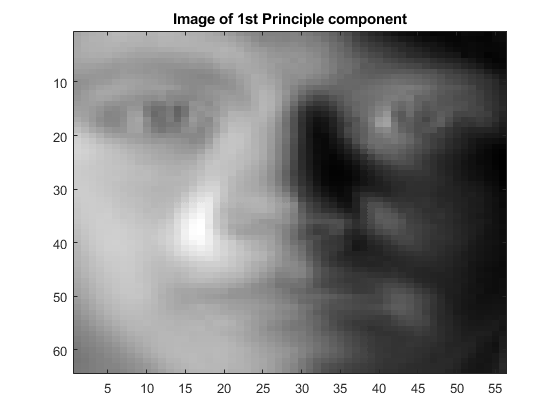
C=pca(var,"NumComponents",3);

I1=reshape(C(:,1),64,56);

imagesc(I1)

title('Image of 1st Principle Eigen vector');

colormap(gray)

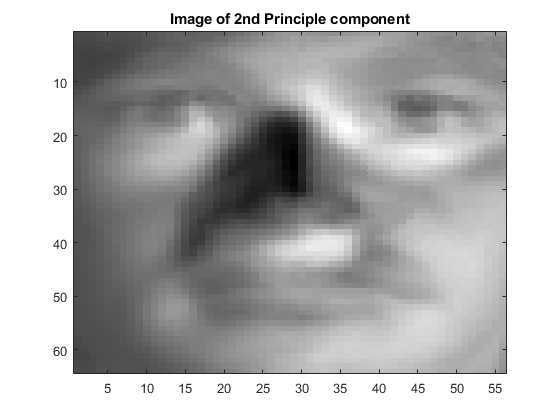


I2=reshape(C(:,2),64,56);

imagesc(I2)

title('Image of 2nd Principle Eigen vector');

colormap(gray)

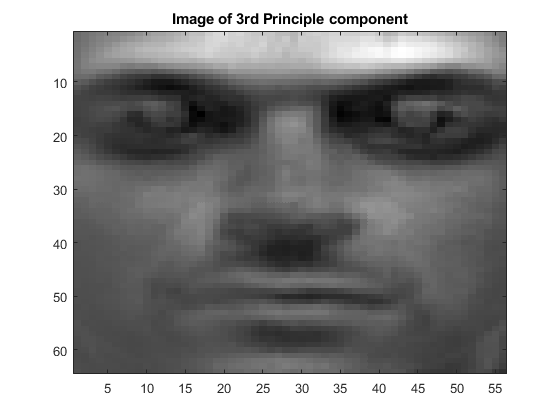


I3=reshape(C(:,3),64,56);

imagesc(I3)

title('Image of 3rd Principle Eigen vector');

colormap(gray)



reconstruction of image 50

d = 20

d = 20

P = 0;

for i = 1:1:d

P = P + U(:,i)'\*((U(:,i)')\*X(:,50));

end

Diff = abs(X(:,50) - P');

Actual = reshape(X(:,50),64,56);

Reconstructed = reshape(P,64,56);

Error = reshape(Diff,64,56);

subplot(1,3,1)

imagesc(Actual);

colormap(gray)

axis equal;

title('Actual 50');

subplot(1,3,2)

imagesc(Reconstructed);

colormap(gray)

axis equal;

title('Reconstructed 50');

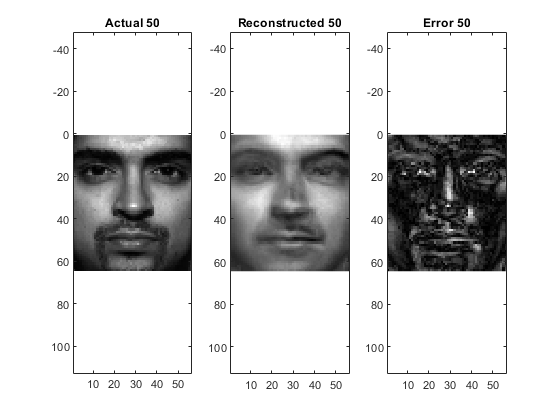
subplot(1,3,3)

imagesc(Error);

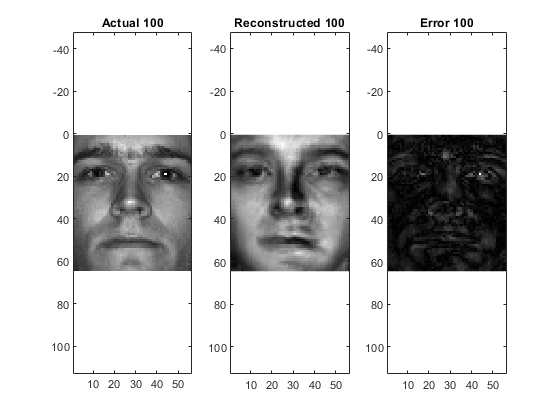
colormap(gray)

axis equal;

title('Error 50');



reconstruction of image 100



reconstruction of image 200

