REPORT ON K-MEANS AND PCA

Implementation of K-means:

1. Selected dataset- Iris dataset of flowers. It consists of a set of 150 records under five attributes - sepal length, sepal width, petal length, petal width and species.

2. First I selected no of clusters K=3 and initialised the centroids randomly.

3. After this, I created a function ClosestCentroids which returns the closest centroids in an array (each row=example).

4. Once the closest centroids are achieved, I created another function to compute new centroids by computing the means of the data points which is assigned to each centroid and I got the centroids after initial finding of the closest centroid.

5. Now plotted the data points in X and maintained a progress function to plot the points with colors assigned to each centroid, It also plots the line between previous and current location of centroid.

6. Now the main step to run the k-means algorithm on matrix X. Its input parameter is initial centroids.
   K-means returns centroids, a K x n matrix of the computed centroids.
   After running k-means, for every iteration we get the updated value.

   ### Results for different values of k:

   On increasing the no of clusters, it was difficult to predict the nature of each cluster as all the clusters started appearing mixing. So results are not properly classified. But for less no of clusters, small differences were not captured. So optimum value of clusters has to be selected to group the input features.

Implementation of PCA:

1. Imported dataset for the face images with name ex7faces. This dataset has 2 variables and 300 observations.
   Select the first n rows from X, plot them as (length of image vector x length of image vector) pixel grayscale images, and combine them to one figure

2. Created display function to display images of the dataset chosen.

3. Selected and normalize the features and created function to calculate eigen values and eigen vectors. Also plotted the eigen vectors centered at the mean of the data.

4. Now to project the normalized inputs X into K dimensional space using top K eigen vectors.

5. Created a recover data function which recovers an approximate original data that has been reduced to K dimensions.

6. And finally projecting it for K=1,2,3 to check the variation in the result.

7. Also plotted the projected points using plot.scatter.

8. To have a better visualization, I also plotted the points lines connecting to projected points and original line.

Results for reconstruction error for different values of K:

As per the code, as the no of dimensions(K) increases the reconstruction becomes more difficult means error increases. Thus it becomes difficult to project back again the lower dimensions to higher dimensions.