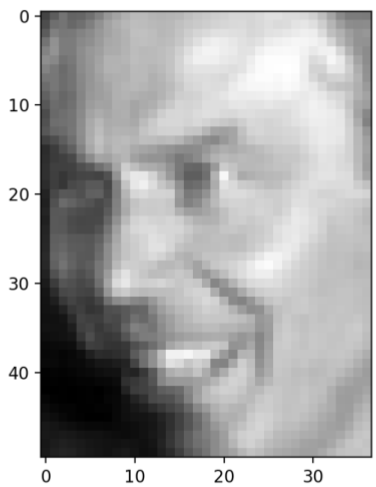
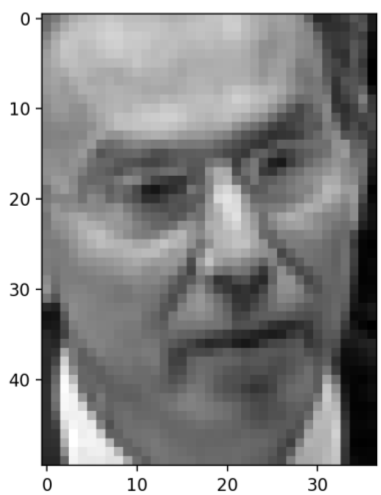
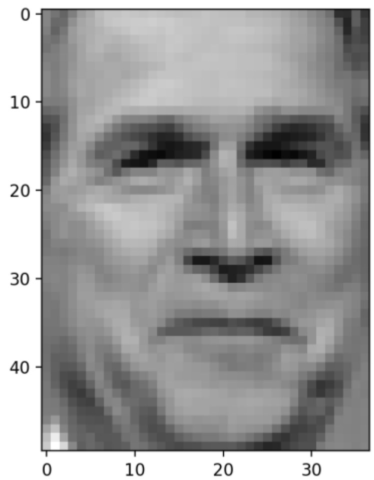
Dhruv Chakraborty
UID: 204-962-098

CM146: PSET 4

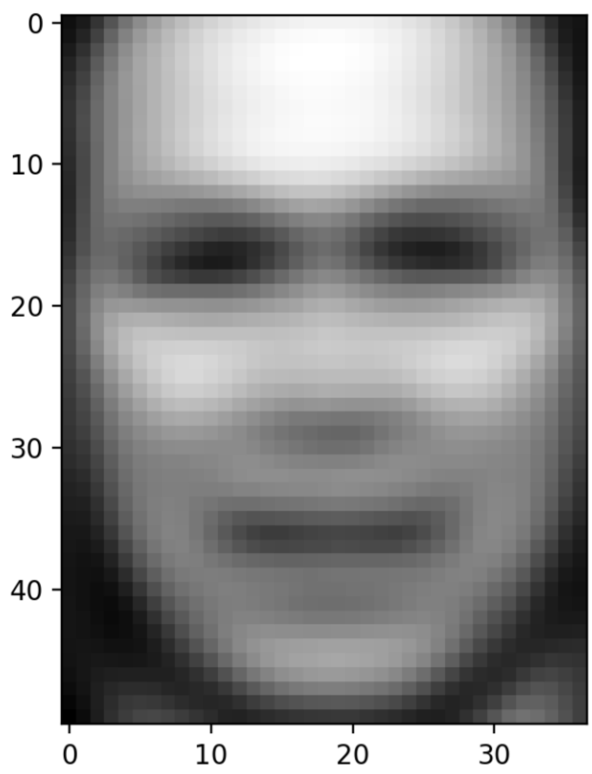
1.
(a) The following are the images of the first 3 people in the dataset:







This is what the average face in the dataset looks like:



It’s quite creepy as although it has the overall definition of a face, it does not have any specific features. This makes sense as it just averages over the pixel values of all images in the dataset, telling us where most images are dark and light. Since all images have been rescaled to face the camera, we get this featureless face as the average image.

(b) Here are the top twelve eigenfaces according to the PCA:

These eigenfaces are selected as they represent the maximum variance in the features in our dataset, so as to best recreate the original images without the use of all the features. Each eigenface amongst these 12 is effectively tasked with capturing a different set of features through which we can get as close to the original images as possible. Essentially, these faces are the best building blocks through which we can build back the original faces, while using considerably fewer features.

(c) The following are the first 12 images decomposed and reconstructed using a different dimensional value L for the PCA:

L = 1

L = 10

L = 50

L = 100

L = 500

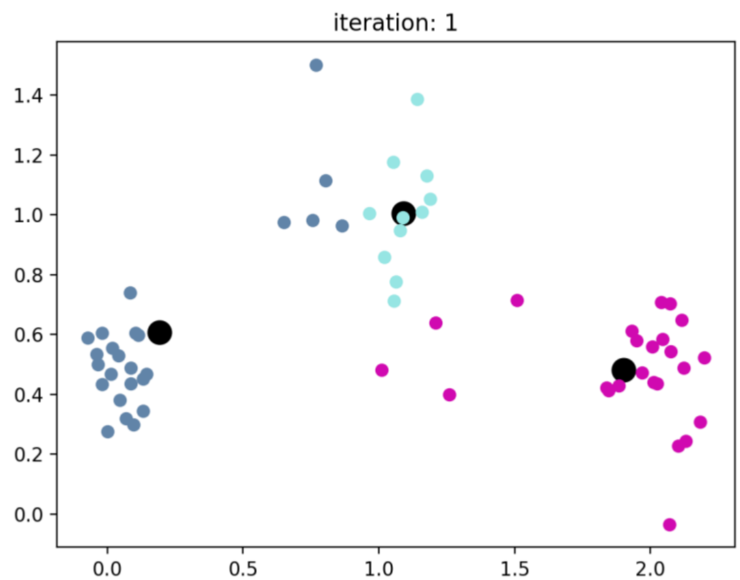
L = 1288

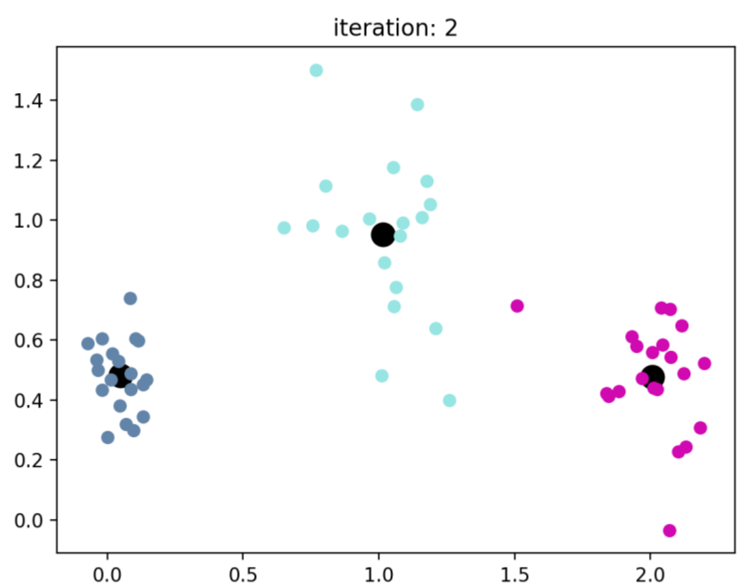
The larger the dimension i.e. L is used, the better (and closer to original) the images are. In particular, with smaller Ls such as 1, 10 and 50, the faces are very blurry and do not represent the original copies. However, L = 500 is very good, and gives us images that are effectively equivalent to the full resolution of L = 1288.

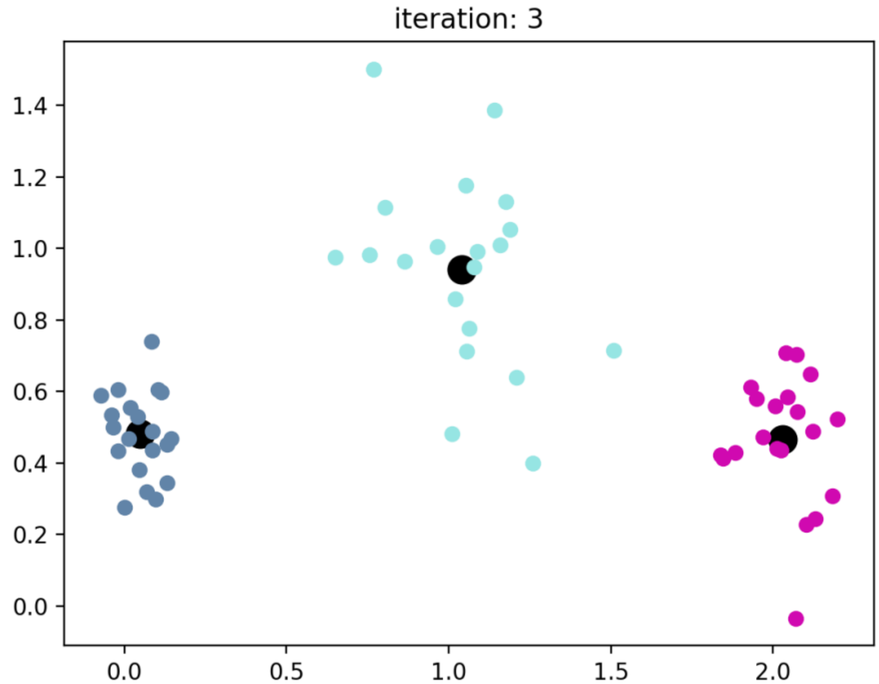
2.

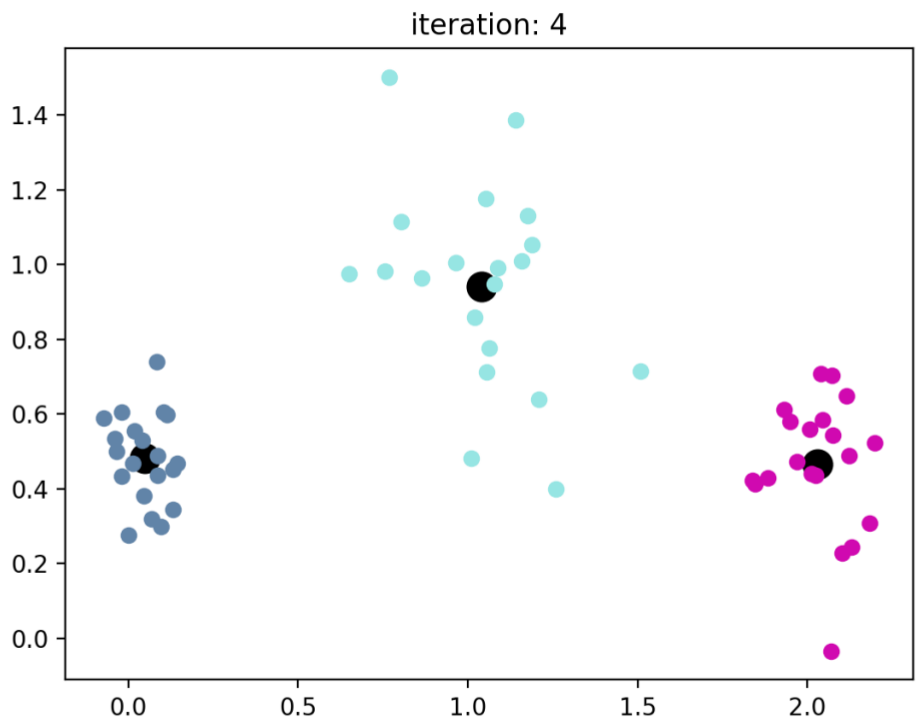
(a) The minimum value for J(mu, c, k) is 0 because we could set k to the number of training samples and have a centroid for each data point, at the same location as itself. So c(i) would be unique for each i and mu(i) = x(i) giving J(mu, c, k) = 0. This is why it would be a bad idea to minimize J(mu, c, k).

(d) The following are the plots for the toy dataset's K-means cluster assignments and corresponding cluster centers using random initialization.

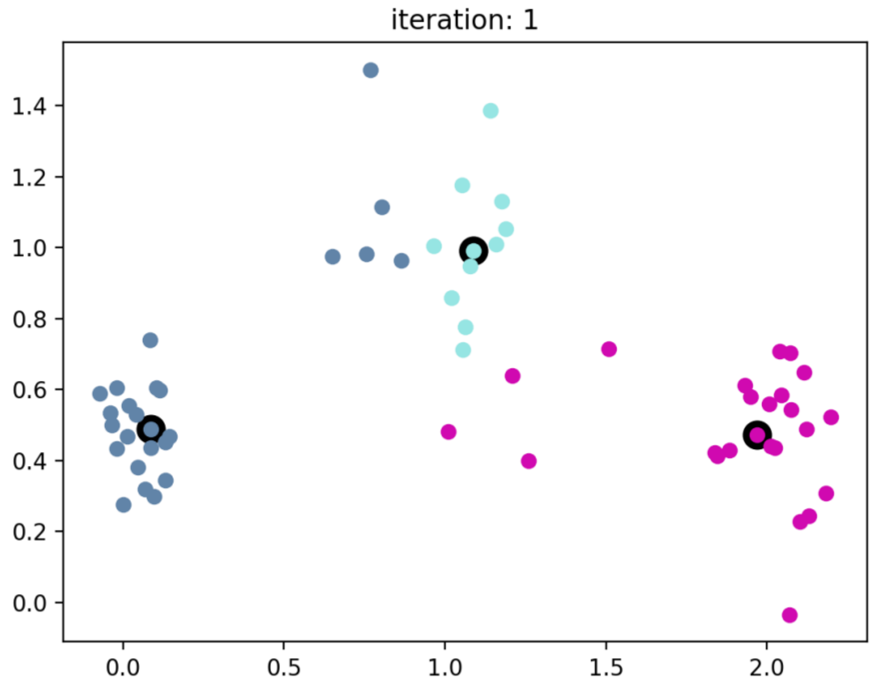


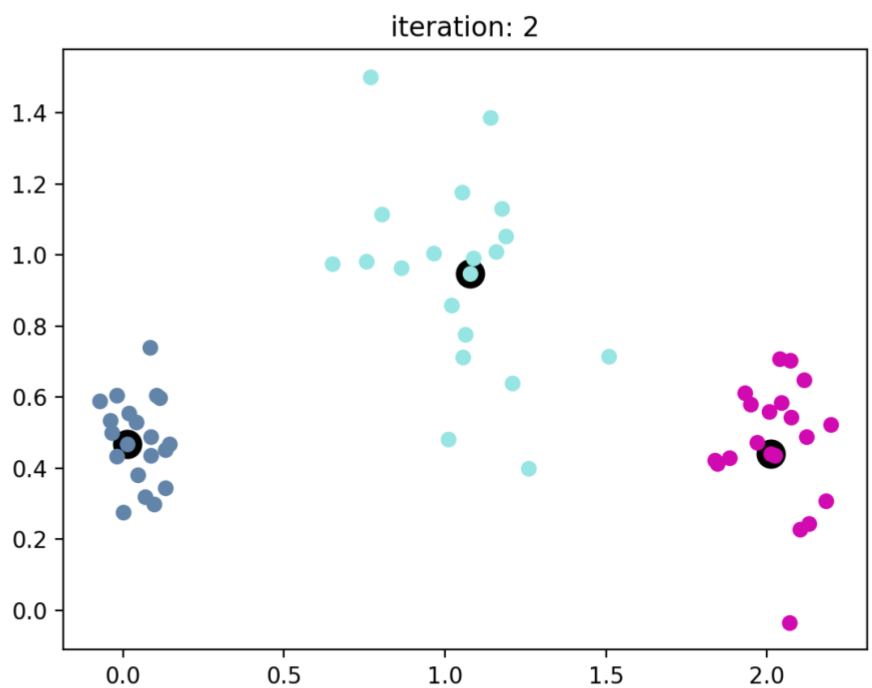


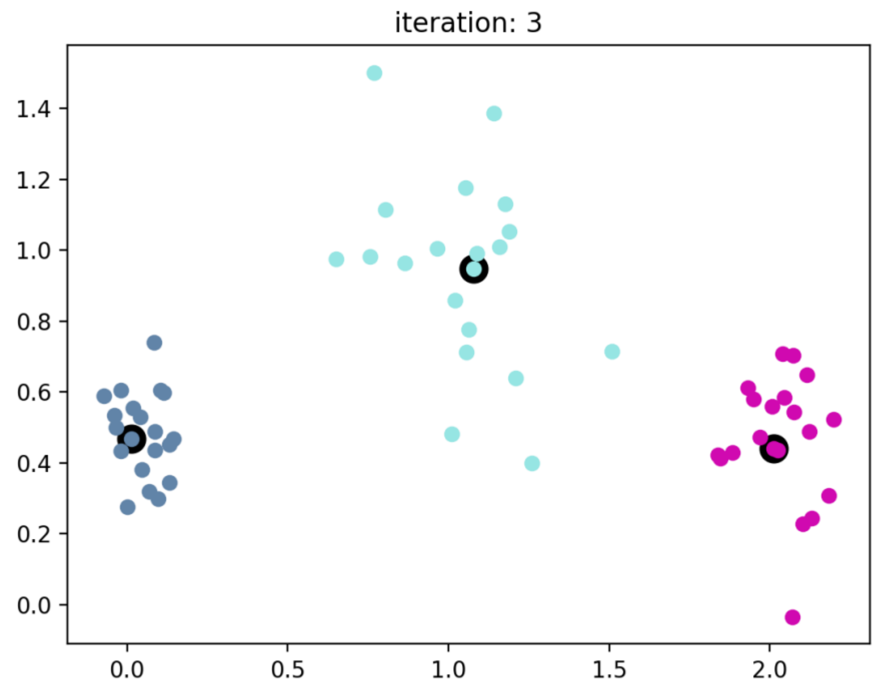

iteration: 3
1.4
1.2
1.0
0.8
0.6
0.4
0.2
0.0
0.0
0.5
1.0
1.5
2.0


iteration: 4
1.4
1.2
1.0
0.8
0.6
0.4
0.2
0.0
0.0
0.5
1.0
1.5
2.0

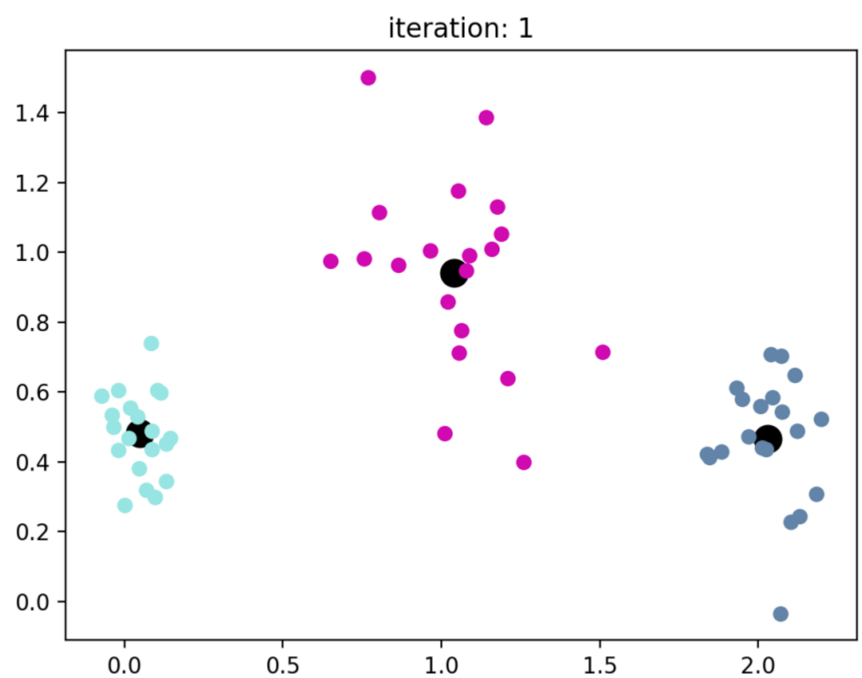
(e) Here are the plots for k-medoids clustering for each iteration using random initialization:

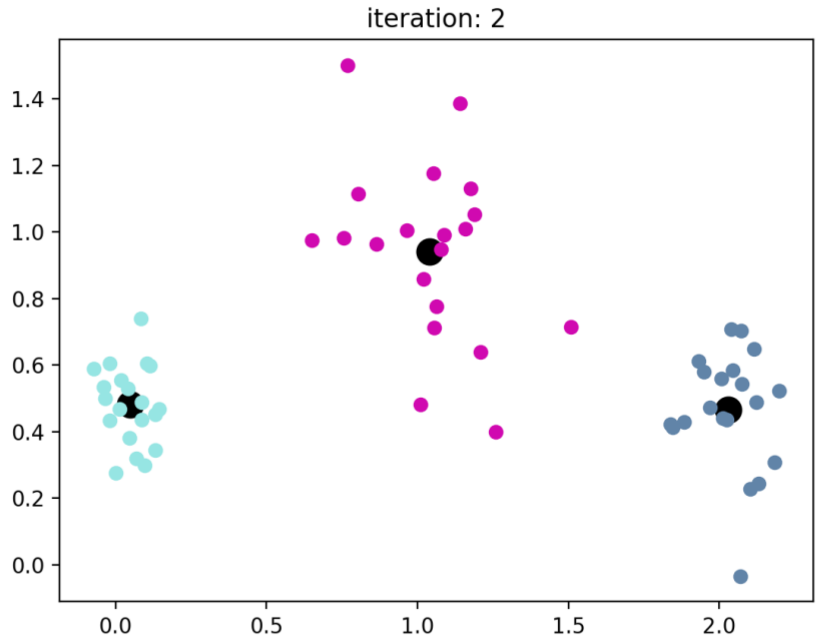




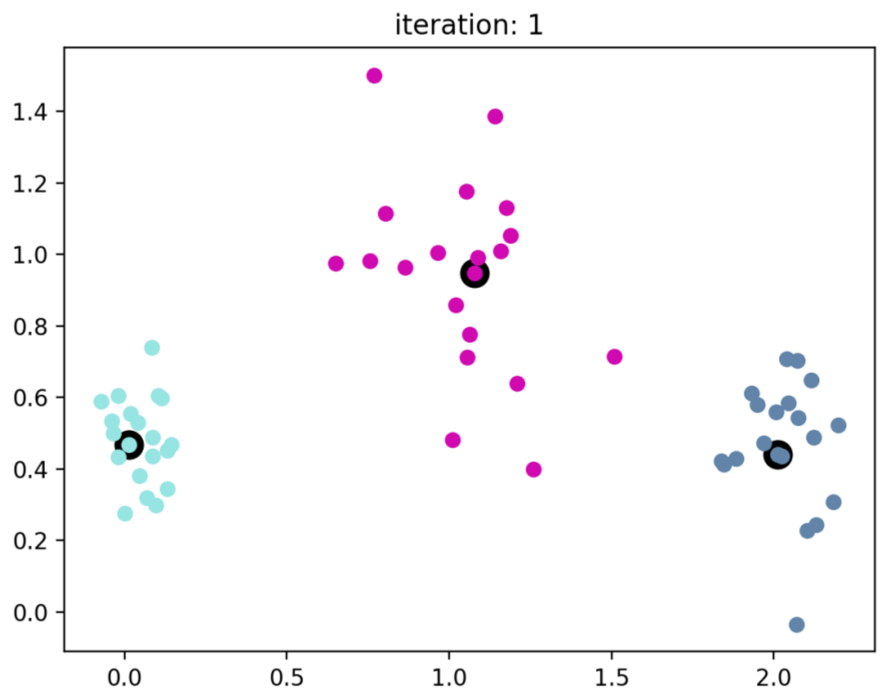


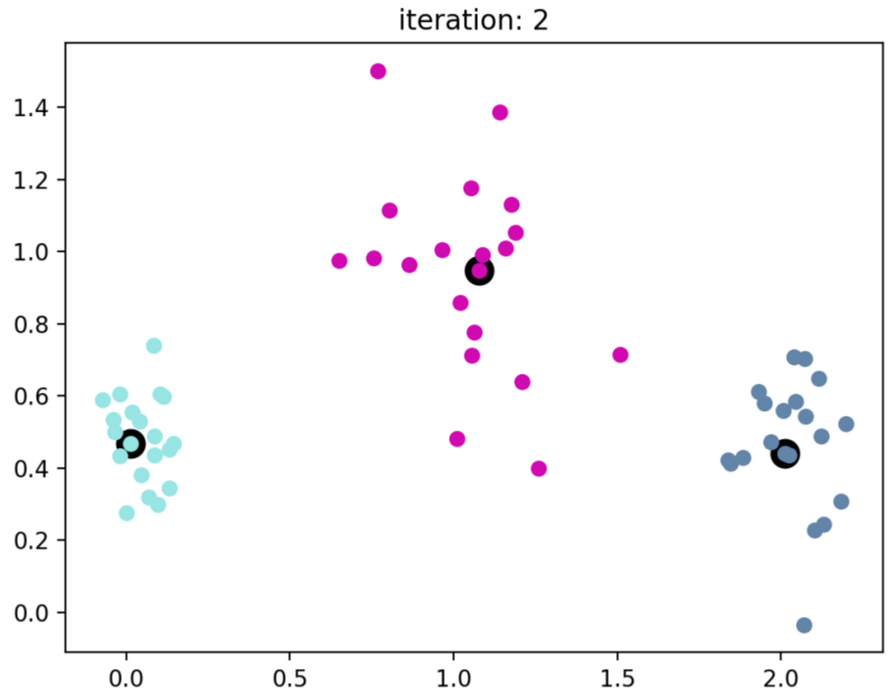
(f) K-means plots:





K-medoids plots:





Both K-Means and K-Medoids need only 1 iteration when using cheat initialization.

3.

(a) The average, min and max performance for both K-Means and K-Medoids are:

k-means average: 0.6175

k-means min: 0.55

k-means max: 0.775

k-means average runtime: 0.177487635612

k-medoids average: 0.6325

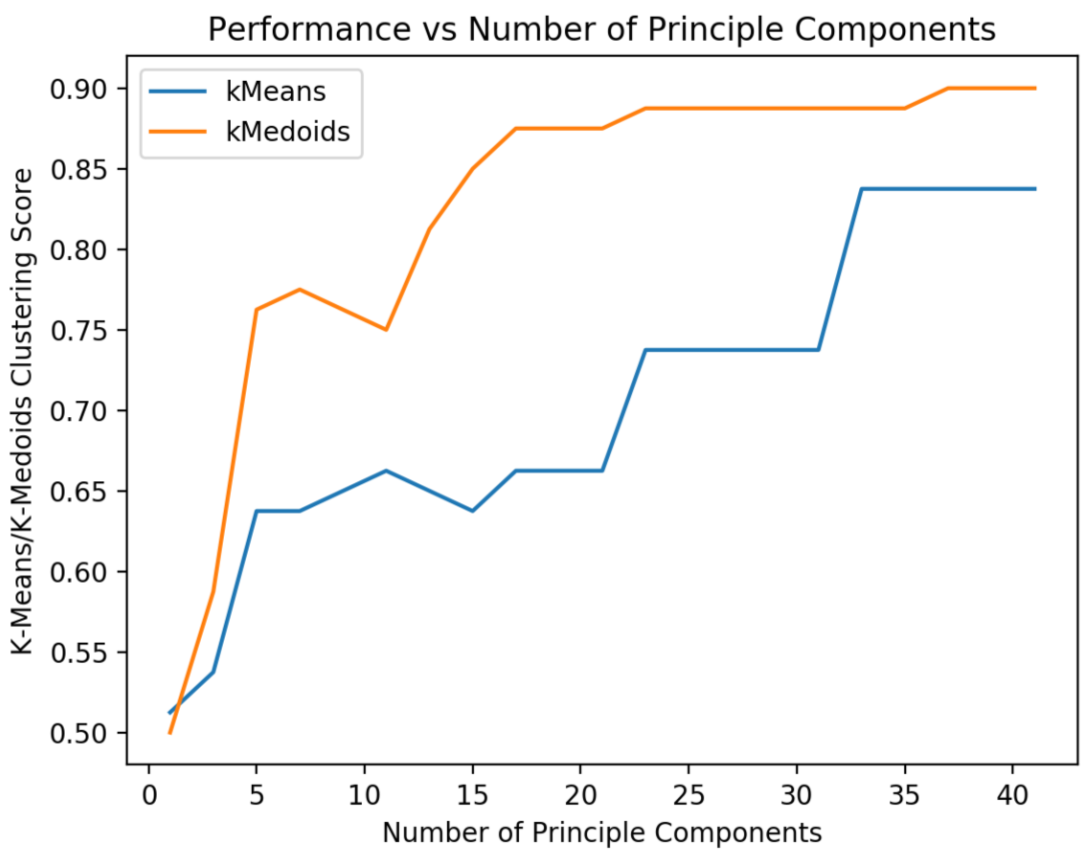
k-medoids min: 0.575

k-medoids max: 0.725

k-medoids average runtime: 0.260527205467

It seems like although both K-Means and K-Medoids have roughly similar performance (with K-Medoids having a slightly higher average and minimum score, and K-Means having a higher maximum score), K-means has a slightly better runtime.

(b) The plot for number of components against K-Means and K-Medoids clustering score is below. We can see that K-Medoids performs better than K-Means no matter how many principle components we're using. Moreover, as the number of principle components are allowed to increase, we get closer and closer to our original feature set, thus observing better performance i.e. more principle components gives a better score for both algorithms.



(c) I used K-Means to check performance over all possible pairs of faces, using the same cheat initialization for performance reasons. While doing this, I recorded the maximum and minimum performances by the KMeans algorithm for the pairs, giving us the easiest to discriminate faces and easiest to discriminate faces respectively.

Easy to discriminate:

Hard to discriminate:

KMeans performed at 98.75% for the easy to discriminate faces and at 50% for the hard to discriminate faces. Looking at these images above, this makes sense as the hard to discriminate faces are actually quite similar whereas the easy to discriminate ones are fairly different with respect to facial structure, eyes, nose, color etc.