CS M146, Winter 2019
Problem Set 4: Clustering and PCA
Due March 16, 2019

Tian Ye
Collaborator: Derek Chu

March 16, 2019

1 PCA and Image Reconstruction

(a) **Solution:**

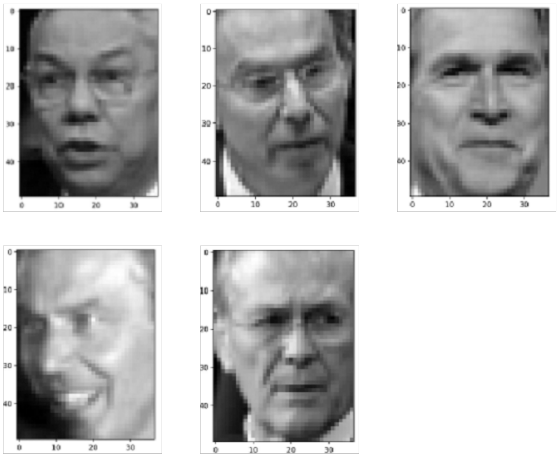

Figure 1: Example Images

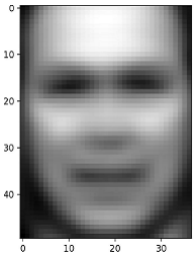

Figure 2: Average Face

Since the "average face" is the mean of all the other faces, it has no extremely distinct facial feature. Nonetheless, the average still shows signs of general features such as eyes, nose, mouth, etc.

(b) **Solution:**

Figure 3: Top 12 Eigenfaces

These twelve faces were selected as the eigenfaces as these were the ones that exhibited the most prevalent and overarching feature sets of all the faces; they represent a smaller set of base vector images which results in the specific features that arrive.

(c) **Solution:**

Figure 4: $l = 1$

Figure 5: $l = 10$

Figure 6: $l = 50$

Figure 7: $l = 100$

Figure 8: $l = 500$

Figure 9: $l = 1288$

As the number of features given to the PCA is increased, so is the facial clarity and feature distinction. This is because as more dimensions are added, more detail is added, hence resulting in improved images.

2 *K*-Means and *K*-Medoids

(a) **Solution:** Attempting to find k-clusters while varying k is a bad idea as when k is arbitrary, the algorithm will naturally take the shortest path to minimizing the error to 0. If k is variable, it will increase k until it matches the number of data points; each data point will be its own cluster which will result in 0 error. k would equal n, c would equal i, and the mean will be the individual coordinates of each individual x_i; it would result in overfitting.

(b) **Solution:** Completed.

(c) **Solution:** Completed.

(d) **Solution:**

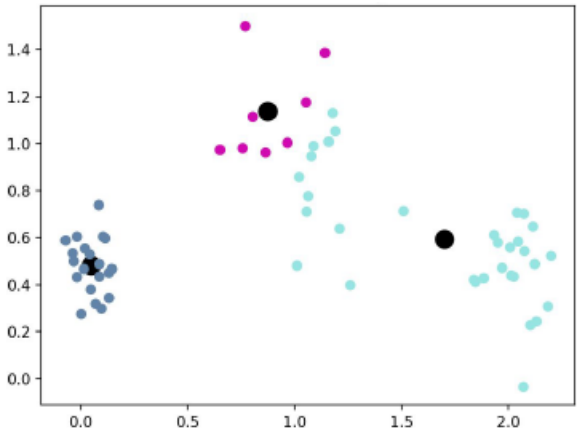

Figure 10: Iteration 1 for Random using Centroids

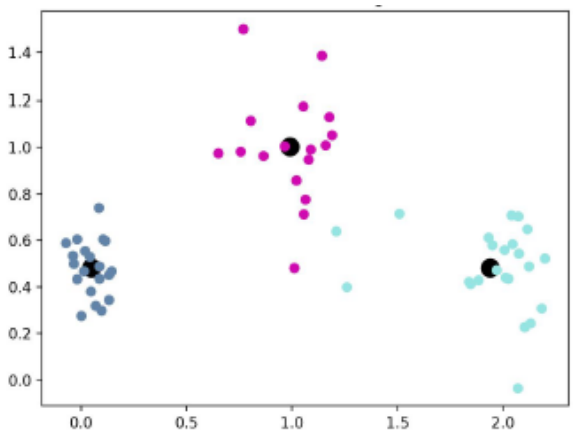

Figure 11: Iteration 2 for Random using Centroids

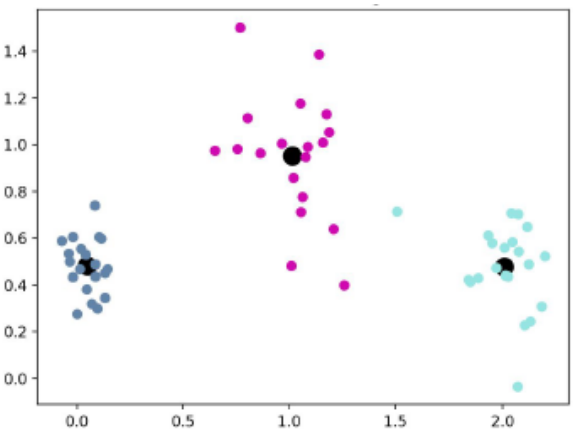

Figure 12: Iteration 3 for Random using Centroids

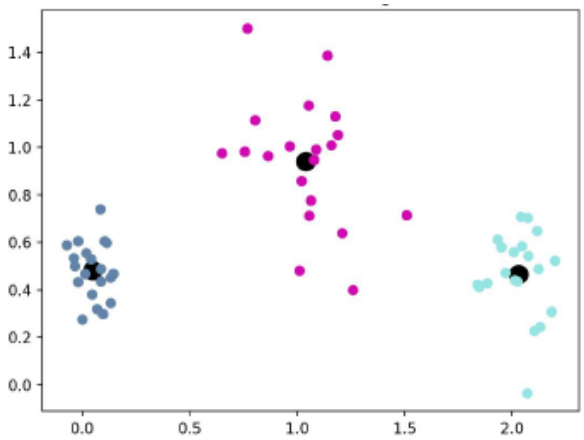

Figure 13: Iteration 4 for Random using Centroids

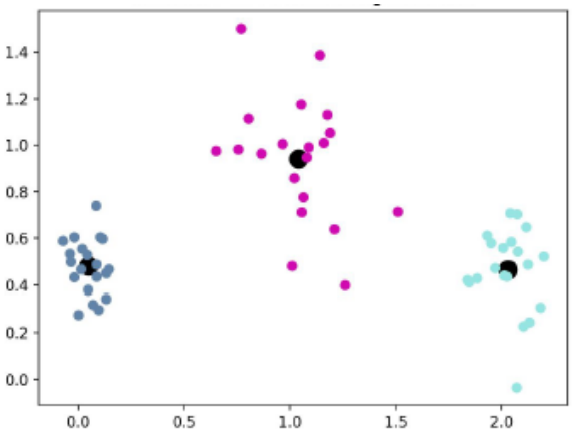

Figure 14: Iteration 5 for Random using Centroids

(e) **Solution:**

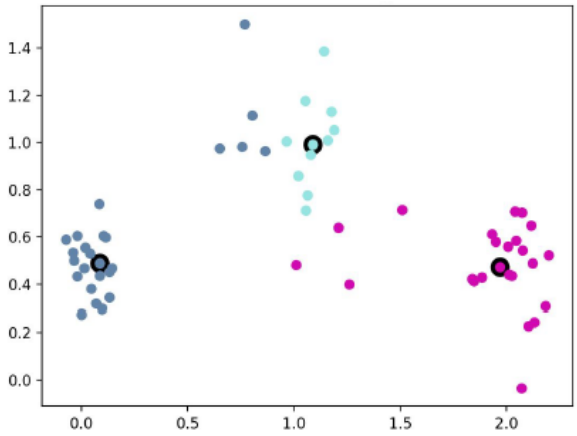

Figure 15: Iteration 1 for Random using Medoids

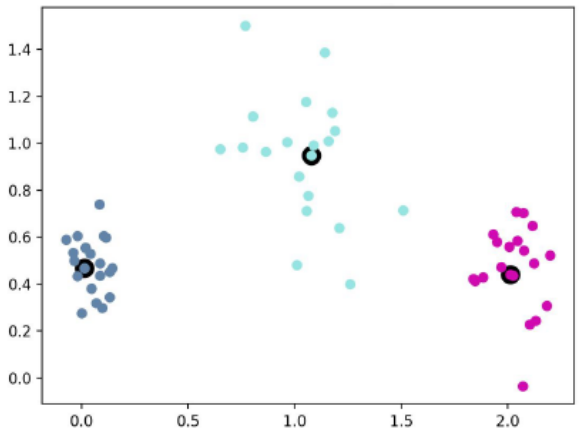

Figure 16: Iteration 2 for Random using Medoids

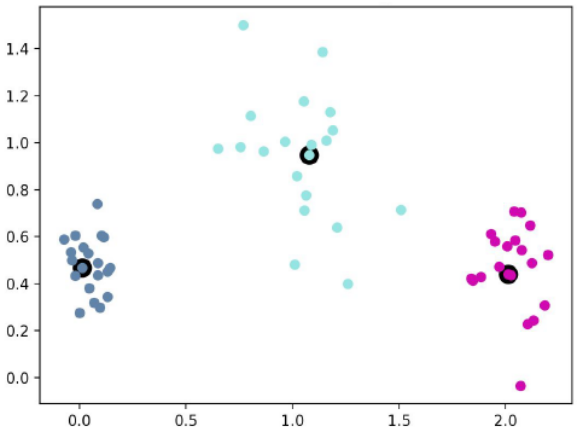

Figure 17: Iteration 3 for Random using Medoids

(f) **Solution:**

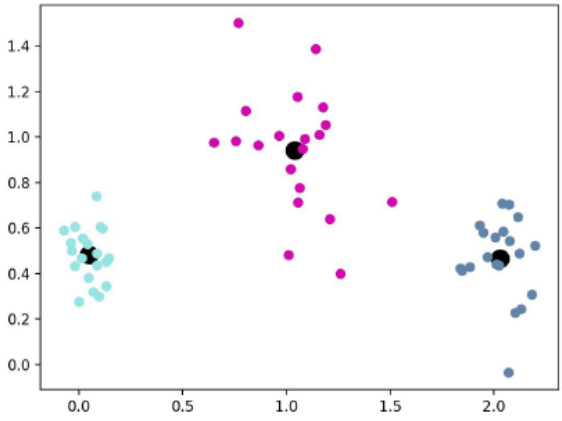

Figure 18: Iteration 1 for Cheat using Centroids

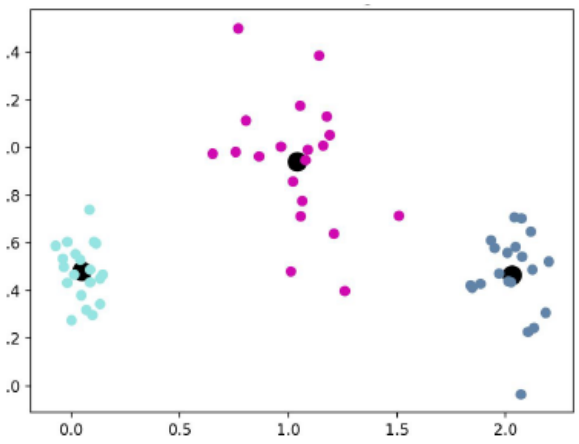

Figure 19: Iteration 2 for Cheat using Centroids

(g) **Solution:**

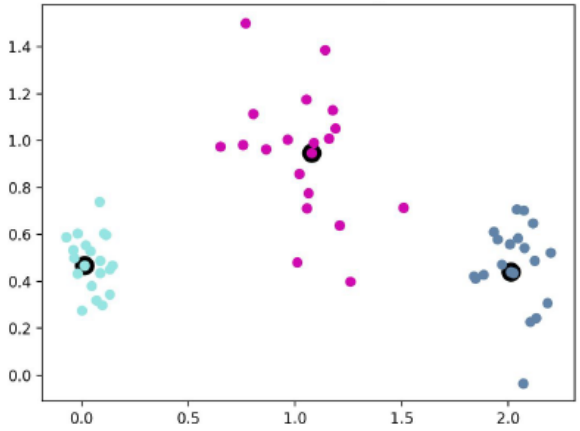

Figure 20: Iteration 1 for Cheat using Medoids

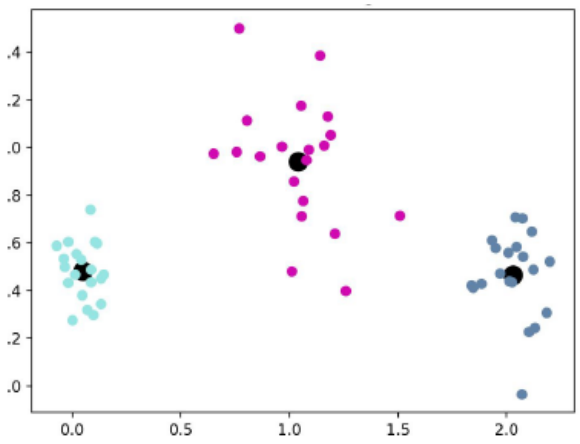

Figure 21: Iteration 2 for Cheat using Medoids

3 Clustering Faces

(a) **Solution:** The averages, minimum, and maximum change with seed; however, since we were given a default seed of 1234, that will be the seed we will use. On average, k-means has worse performance than k-medoids based on pure metric scores; k-means does have better efficiency though. k-medoids has better performance while having a slower time; this is because k-means skew in the presence of outliers while k-medoids attempts to achieve a cluster center. Comparing execution times of the two, we see that k-means executes faster than k-medoids, being ~0.16 and ~0.30 respectively.

	Average	Mininum	Maximum
K-Means	0.6175	0.550	0.775
K-Medoids	0.6325	0.575	0.725

(b) **Solution:**

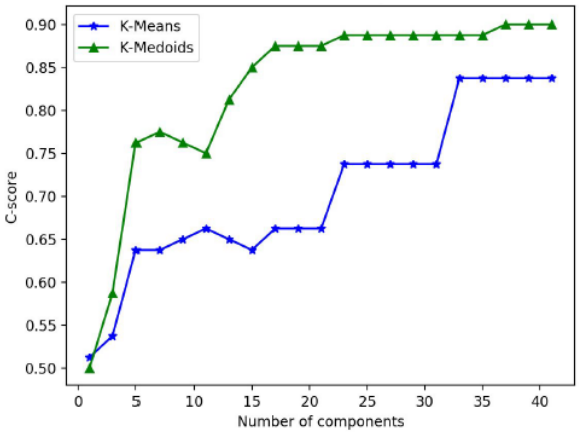

Figure 22: Cluster Score vs Components

Referring to the figure, we can see that the C-score increases proportionally to the number of components for both the k-means and k-medoid algorithms; this is similar to what we saw in the previous PCA implementations. This is because additionally components permit more feature to manifest, resulting in a higher clustering score.

(c) **Solution:** For my algorithm, I wrote a design that simply paired two faces and compared them using two clusters and the cheat optimization. It would then run that comparison over all possible combinations of faces and rank all the scores, selecting the lowest and highest scores which represented the easist and hardest faces to cluster, respectively. The faces that were ultimately selected were 4 and 5 with a score of 0.5125 as well as 9 and 16 with a score of 0.9875. Faces 4 and 5 have similar mouths, noses, and eyes as well as having the same skin tone, while 9 and 16 are distinct not only by the shape of the facial features but also by their different skin tones.

Figure 23: Faces 4 and 5

Figure 24: Faces 9 and 16