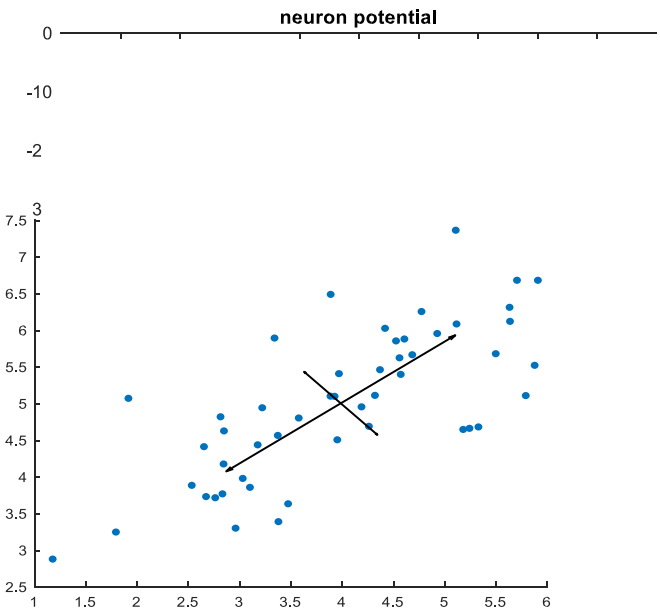
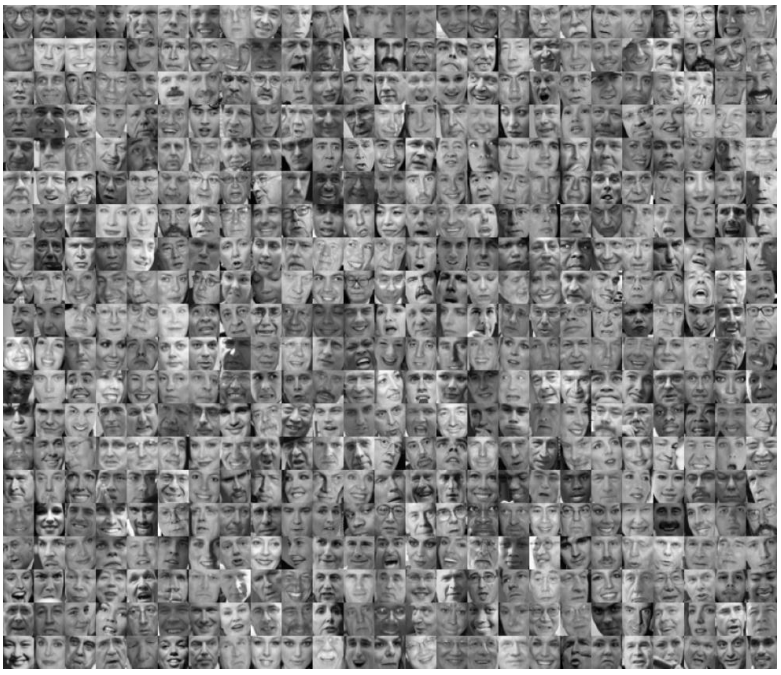
3. PCA

3.1



3.2

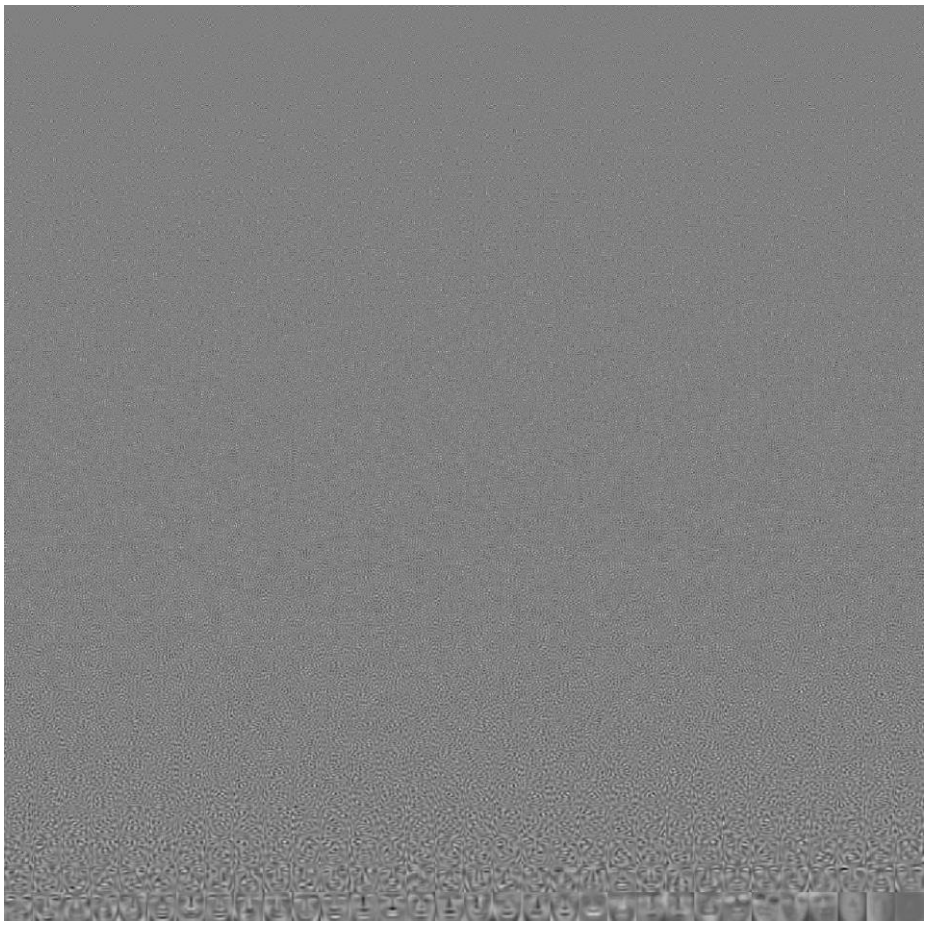
The initial faces:



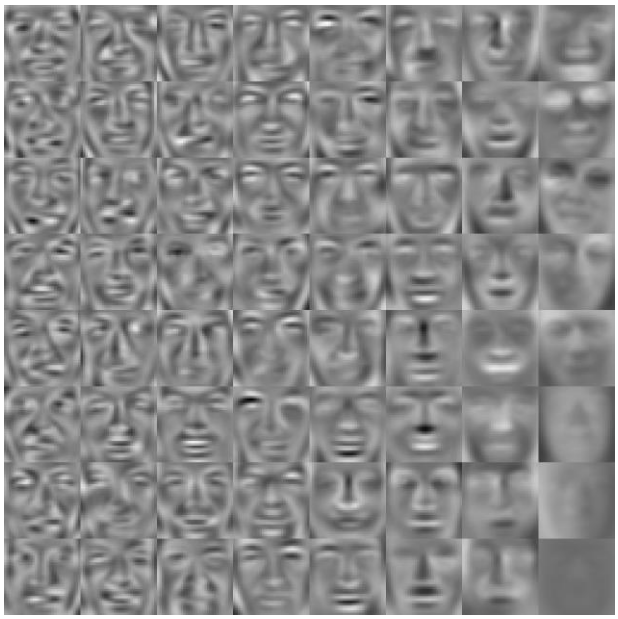
The five largest eigenvalues:

0.5915
0.6942
1.1201
2.3158
5.1156

Image of all the eigen faces:

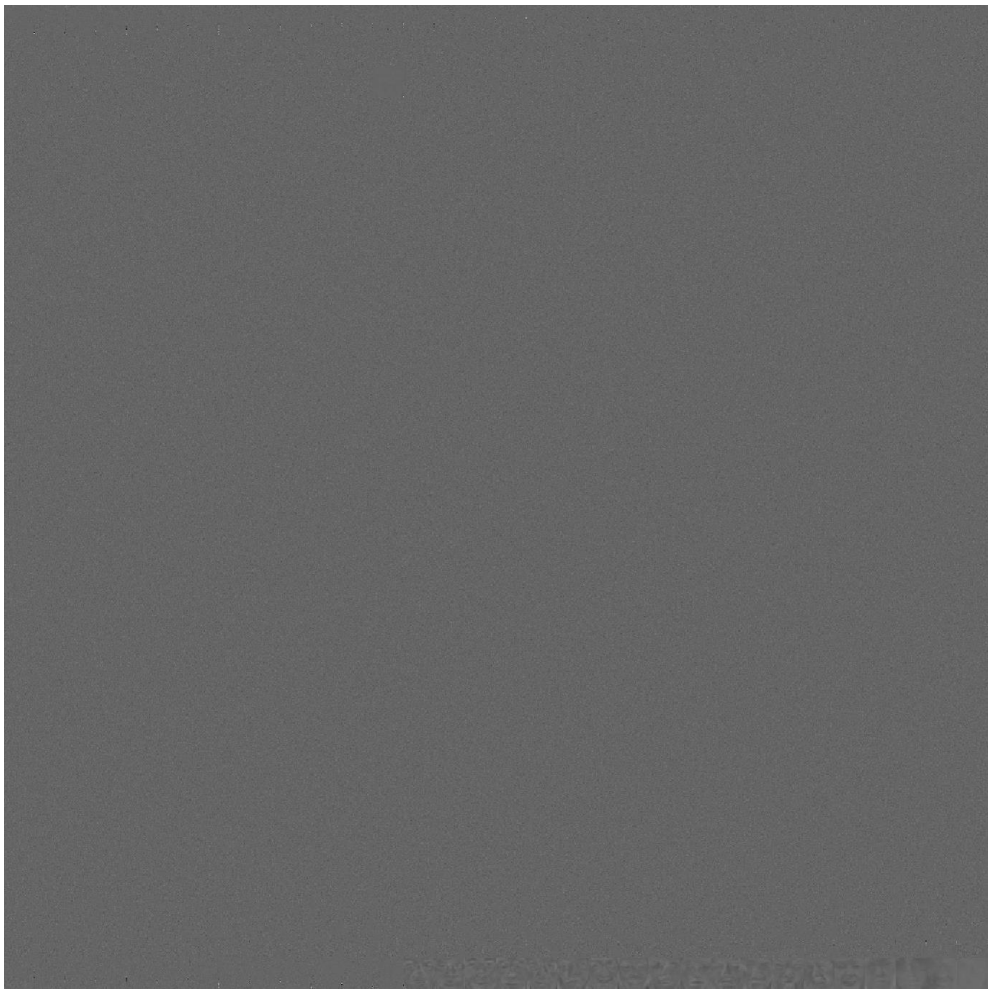


The last 64 eigen faces:

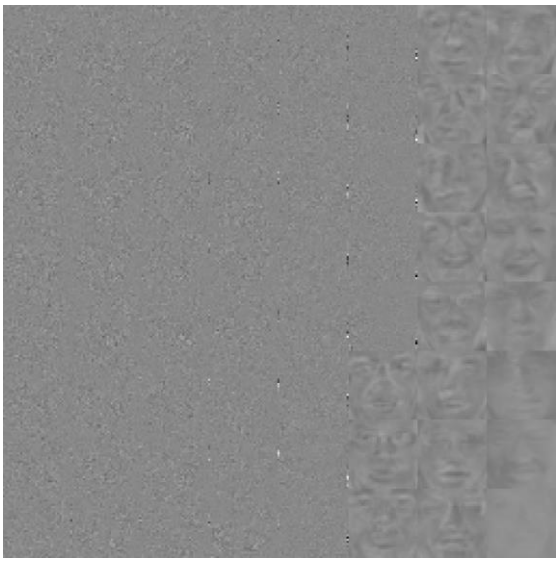


Now if just the first 20 faces are used for data training:

Image of all eigen faces:



The last 64 eigen faces:



4. K-Means

4.1

In both of this techniques the data will be categorized into k clusters, and since each cluster's representative is calculated by finding the average of its members, the result of both algorithms will be the same if and only if the clusters be the same.

Imagine a situation where element (i) is in the first cluster considering k-mean algorithm and the second cluster by applying competitive learning. It is possible only if element (i) has the same Euclidean distance with both of the clusters' centers.

4.2

In each cycle that the algorithm is running each element's cluster will change if it leads to smaller distance, thus, the sum of Euclidean distances will become smaller after each cycle if the end has not been reached yet, and will stay constant if the algorithm is finished.

4.3

The E-step of the EM algorithm computes the expected value of $l(\theta; X, Y)$ given the observed data, X , and the current parameter estimate, θ_{old} say.

$$(1)\quad Q(\theta;\, \theta_{old}) = E[l(\theta; X, Y)|X, \theta_{old}] = \int l(\theta; X, y)p(y|X,\, \theta_{old})dy$$

where $p(.\,|X, \theta_{old})$ is the conditional density of Y given the observed data, X , and assuming $\theta = \theta_{old}$.

The M-step consists of maximizing over θ the expectation computed in (1). That is, we set

$$\theta_{new} = \max_{\theta} Q(\theta;\, \theta_{old})$$

We then set $\theta_{new} = \theta_{old}$.

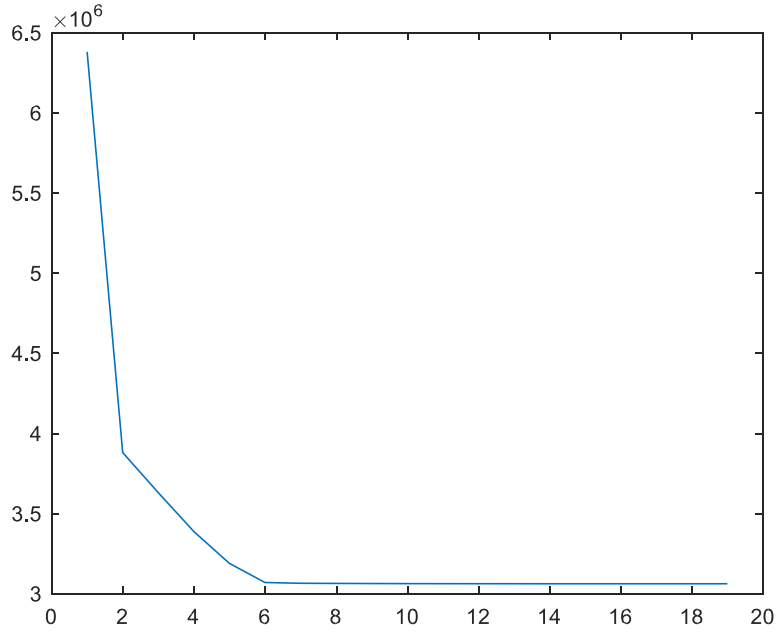
By the above definitions the second step of k-mean algorithm is the same as E-step because in both steps we are calculating parameters based on the data we have. In the k-mean's third step and m-step we are trying to optimize the parameter. In one minimize the cost function and in other one maximize the parameter computed in the previous step.

4.4 (4.9)

The original picture:

4.6 (4.11)

This plot is for k = 4:



4.7 (4.12)

K = 4

K = 6

K = 8

K = 10

K = 12