CSE 547: Machine Learning for Big Data
Homework 2

Academic Integrity We take https://www.cs.washington.edu/academics/misconductacademic integrity extremely seriously. We strongly encourage students to form study groups. Students may discuss and work on homework problems in groups. However, each student must write down the solutions and the code independently. In addition, each student should write down the set of people whom they interacted with.

Discussion Group (People with whom you discussed ideas used in your answers):

On-line or hard copy documents used as part of your answers:

I acknowledge and accept the Academic Integrity clause.

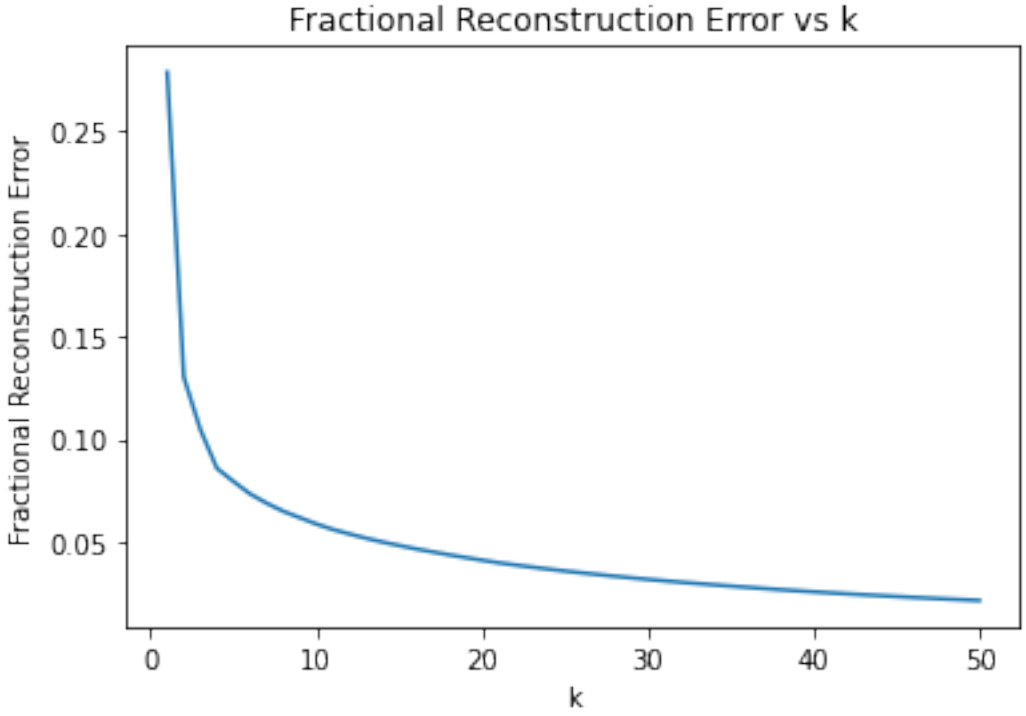
(Dongyang Wang)

Answer to Question 1(a)

1.
Since the matrix $\Sigma = \frac{1}{n}X^TX$, it is symmetric and can be decomposed into $\Sigma = UDU^T$, where D is a diagonal matrix with the eigenvalues of Σ along the diagonal and U is an orthogonal matrix, whose columns are the eigenvectors of Σ. $Tr(\Sigma) = Tr(UDU^T) = Tr(DU^TU) = Tr(D)$, which are the eigenvalues of Σ. Therefore, $Tr(\Sigma) = \sum_{i=1}^{d} \lambda_i$.
Also, $Tr(\Sigma) = Tr(\frac{1}{n}X^TX) = \frac{1}{n}Tr(X^TX) = \frac{1}{n}Tr(XX^T)$, by definition of X, $Tr(\Sigma) = \frac{1}{n}Tr(XX^T) = \frac{1}{n}\sum_{i=1}^{n} ||x_i||_2^2$.
2.
For the third part, since each column of P^* is a the corresponding eigenvector λ_i and X to XP is a linear projection, $Tr(\frac{1}{n}P^{*T}X^TXP^*) = Tr(\frac{1}{n}X^TX)$, with the result from above, we know $Tr(\frac{1}{n}P^{*T}X^TXP^*) = \sum_{i=1}^{k} \lambda_i$.

Answer to Question 1(b)

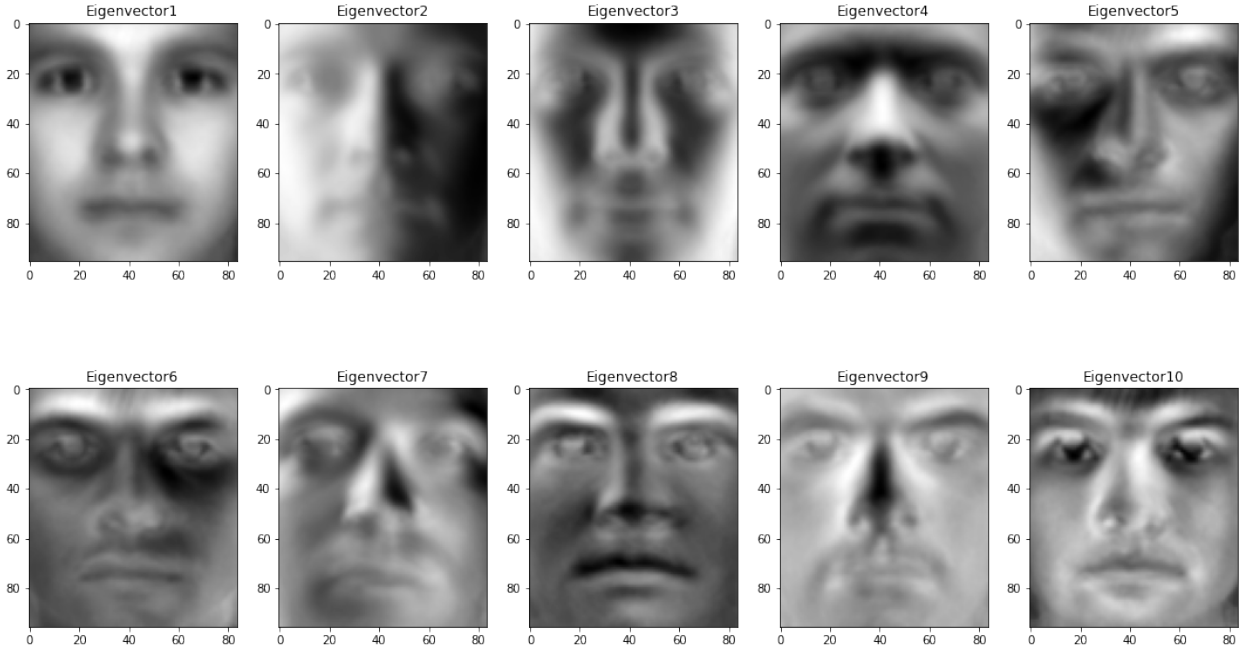
1.
The 1st Eigenvalue is (781.8126992600016+0j)
The 2nd Eigenvalue is (161.15157496732695+0j)
The 10th Eigenvalue is (3.339586754887828+0j)
The 30th Eigenvalue is (0.809087790377721+0j)
The 50th Eigenvalue is (0.38957773951814434+0j)
The trace of the sigma matrix is 1084.2074349947673
2.



3.
The principal eigenvalue (largest one) is the one that captures the first dimension and largest dimension of the variation in the variables. Since all the data points will have at least some kind of variation, the first one can be significantly larger than the rest because later variations can be more subtle given the first dimension (the biggest possible variation) is fixed.

Answer to Question 1(c)

Here is a list of image results for top 10 eigenvectors obtained from PCA.

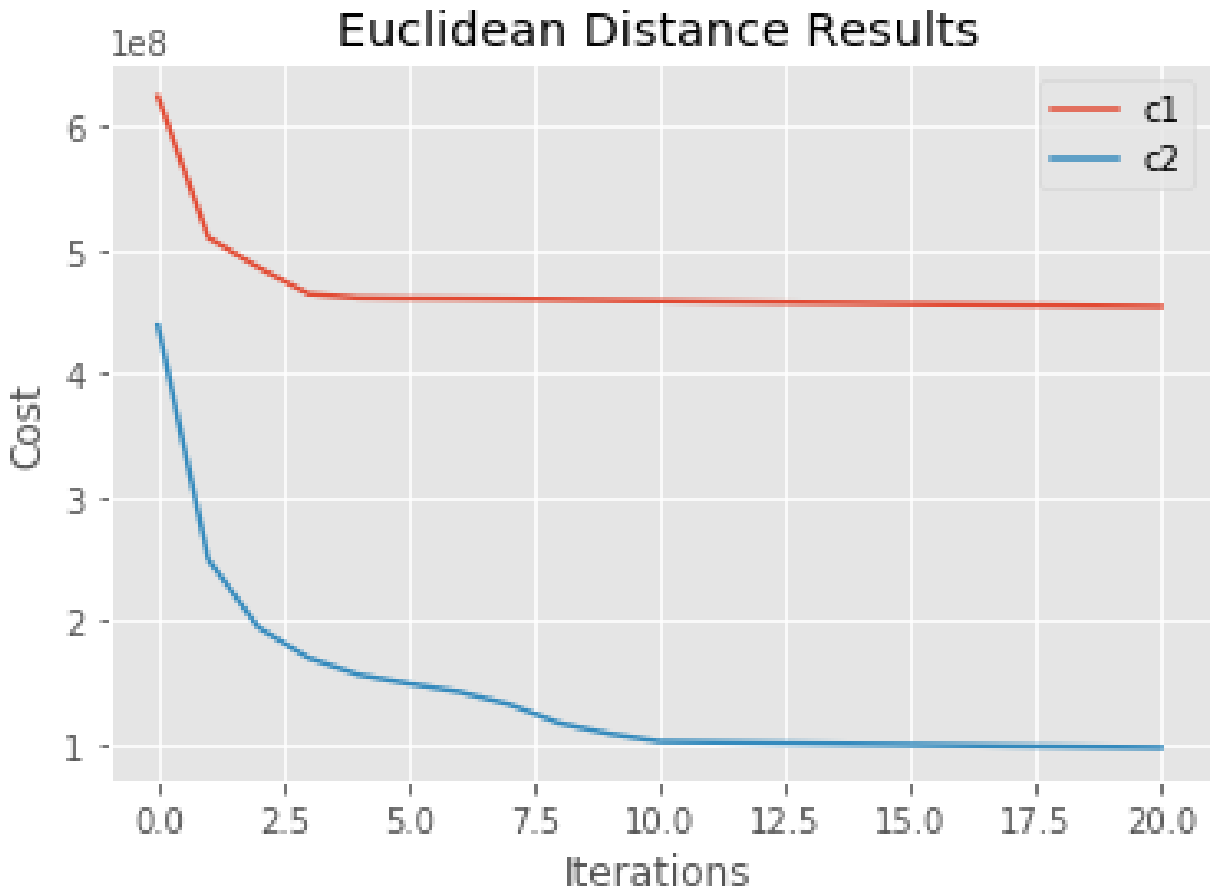


Based on the graphics above, it seems that the eigenvectors have captured the basic shapes of human faces, although the capture is a bit rough. Different eigenvectors have different focuses, and the eigenvectors with higher eigenvalues tend to have better resemblance of human faces. For example, eigenvector 1 has a good overall capture of the human face, while eigenvector 8 is good on eyebrows specifically. For the others, eigenvector 2 seems to compare left and right faces; eigenvector 3 is about the peripheral shapes of the faces; eigenvector 4 is about the nose mostly; eigenvector 5 is more about the right part of the image, or left face; eigenvector 6 focuses on forehead; eigenvector 7 is more on top left and lower right corners of the image of a face; eigenvector 9 is a bit abstract and more on face parts other than the nose; eigenvector 10 is on mouth and nose more or less.

Answer to Question 1(d)

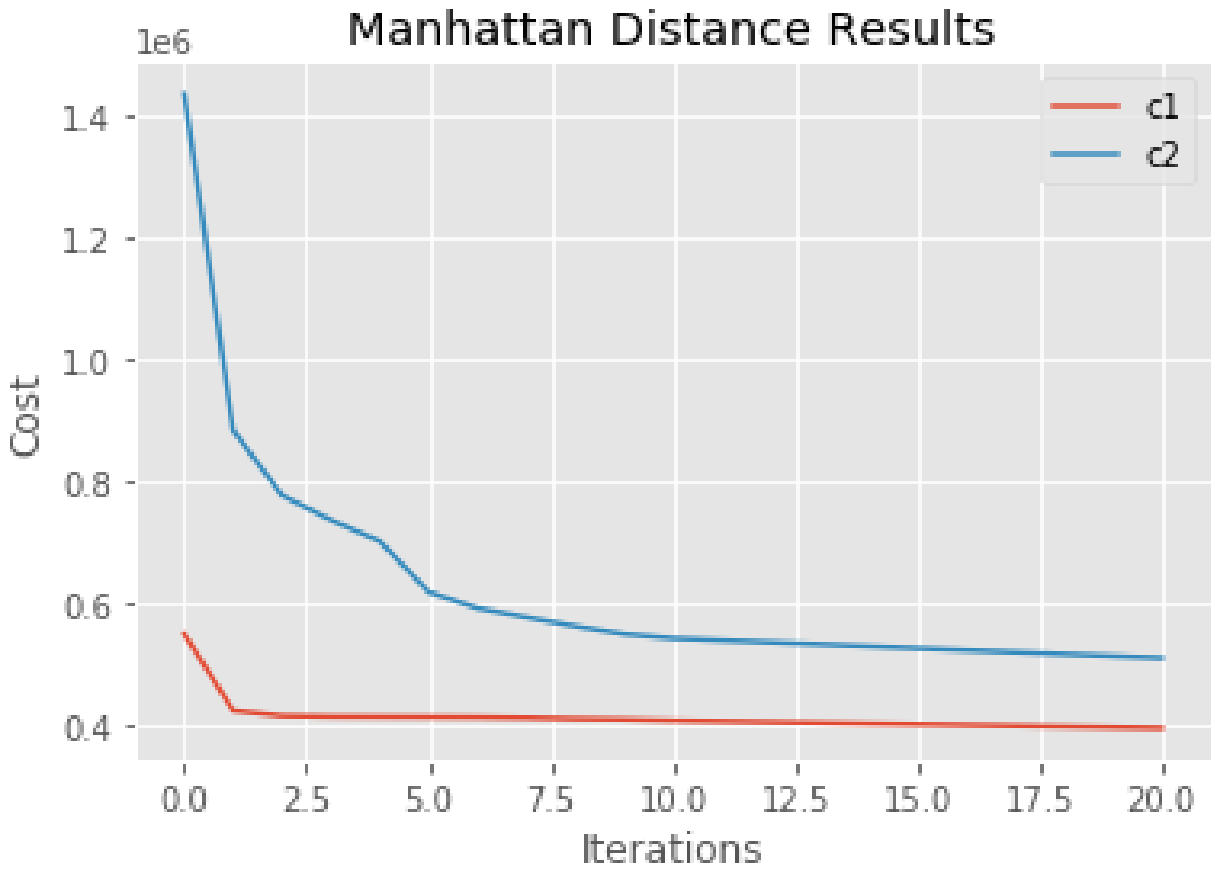
Based on the results from above, the approximation gets better as we include more eigenvectors. The quality of image in terms of clarity and contrast and lighting all increases as more eigenvectors all included. Also, the pictures are very similar in terms of face shapes (but not lighting) when the number of eigenvectors is small. The eigenvectors in c together can make the picture clearer, has more detailed shapes, and better lighting that more resembles the original piece. From k =1 to k =2, for example, there second image(23) shows significant improvement in quality because the contrast between the left and right face can now be captured. Still in the same row of image, the forehead and eyebrows get clearer and more alike the original as we increase k so eigenvectors 6 and 8 are in use.

Answer to Question 2(a)



The percentage of error reduction for c1 is: 26.48391714456061 %.
The percentage of error reduction for c2 is: 76.69795594605938 %.
It seems that c2 has significantly better reduction in errors as well as performance. The intuition is that with cluster centroids more spread out, it's more likely to classify points into distinct clusters with different means. The percentage is so high because as the estimated centroids get closer to the true centroids a lot faster with centroids being spread out.

Answer to Question 2(b)



The percentage of error reduction for c1 is: 25.771956039138043 %.
The percentage of error reduction for c2 is: 62.13918226180144 %.
Different from the mean case, the median can better be approximated by c1, although the reduction in error is greater in c2. As noted before, the fast reduction in error is due potentially to the vast change in the centroids since each iteration updates the centroids greater for c2 initialized centroids. On the other hand, the median is better approximated by a random generation of points (c1), since the points that are far out (c2) are absolutely not medians.

Answer to Question 3(a)

We want to place additional weight on the cases where the user actually interacted with the item. Therefore, in cases where the user doesn't, we want to simply use 1 to indicate that no effect has been measured.

Answer to Question 3(b)

Taking the derivative for C wrt X, converting the sums, we obtain

$$C' = 2(Y^T C_u Y X - Y^T C_u p_u) + 2\lambda X$$

. Setting it equal to 0, we have

$$Y^T C_u p_u = Y^T C_u Y X + 2\lambda X$$

, which is equivalent to

$$Y^T C_u p_u = (C_u Y + 2\lambda) X$$

and simple transformation gives us

$$x_u = (C_u Y + 2\lambda I)^{-1} Y^T C_u p_u$$

.

Answer to Question 3(c)

The number of nonzero entries is the number of observations where $r_{ui} > 0$, or where the confidence weight is not zero, or where the user at least interacted with the item somehow. The complexity can be easily derived then. Since $O(Y^T C_u Y)$ now concerns only the second part of decomposition, i.e., $O(Y^T(C_u - I)Y)$ and the core term is captured by $O(C_u - I)$ as discussed earlier, then it is equivalent to $O(n_u f^2)$. It means that it has f^2 entries and each entry requires n_u times summation.

Answer to Question 3(d)

The sparsity ratio is 0.011265055687640536.
Before any iteration the cost is