CSE 547: Machine Learning for Big Data
Homework 2

Answer to Question 1(a)

1. proof: we only need to prove that for any square matrix X and Y, we have tr(XY)=tr(YX). This can be done by using the definition of trace. tr(XY)=$\sum_{i=1}^{n}(XY)_{ii} = \sum_{i=1}\sum_{k=1} X_{ik}Y_{ki} = \sum_{k=1}\sum_{i=1} X_{ki}Y_{ik} = \sum_{k=1}^{n}(YX)_{kk} = tr(YX)$

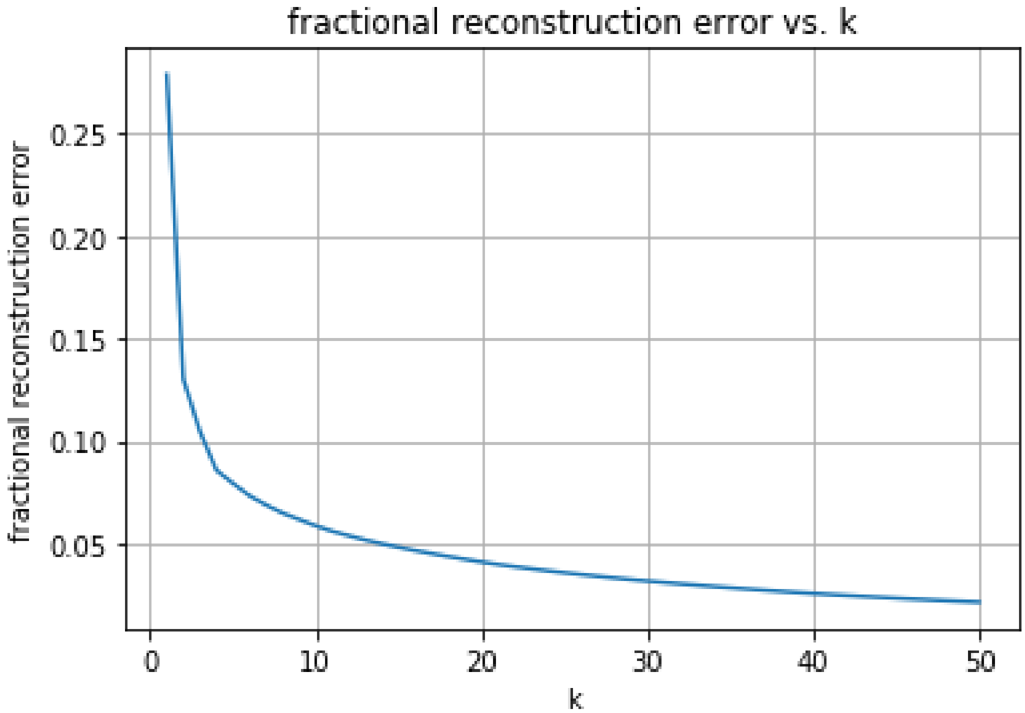
2. (a) Since Σ is a symmetric matrix, it can be decomposed as $Z\Lambda Z^T$, where Z is orthogonal and Λ is diagonal matrix containing all the eigenvalues of Σ. From the above question, we have $tr(Z\Lambda Z^T) = tr(Z^T Z\Lambda) = tr(\Lambda) = \sum_i \lambda_i$.

   (b) tr(Σ) $= 1/\mathrm{n}\sum_{i=1}(X^T X)_{ii} = 1/n\sum_i(X_i^T X_i) = 1/n\sum_i \|X_i\|^2$

Answer to Question 1(b)

1. $\lambda_1 = 781.8126992600016, \lambda_2 = 161.15157496732675, \lambda_3 = 3.3395867548878746, \lambda_4 = 0.8090877903777226, \lambda_{50} = 0.3895777395181458$. From the previous question, the sum is the trace of Σ, that is, 1084.2074349947673

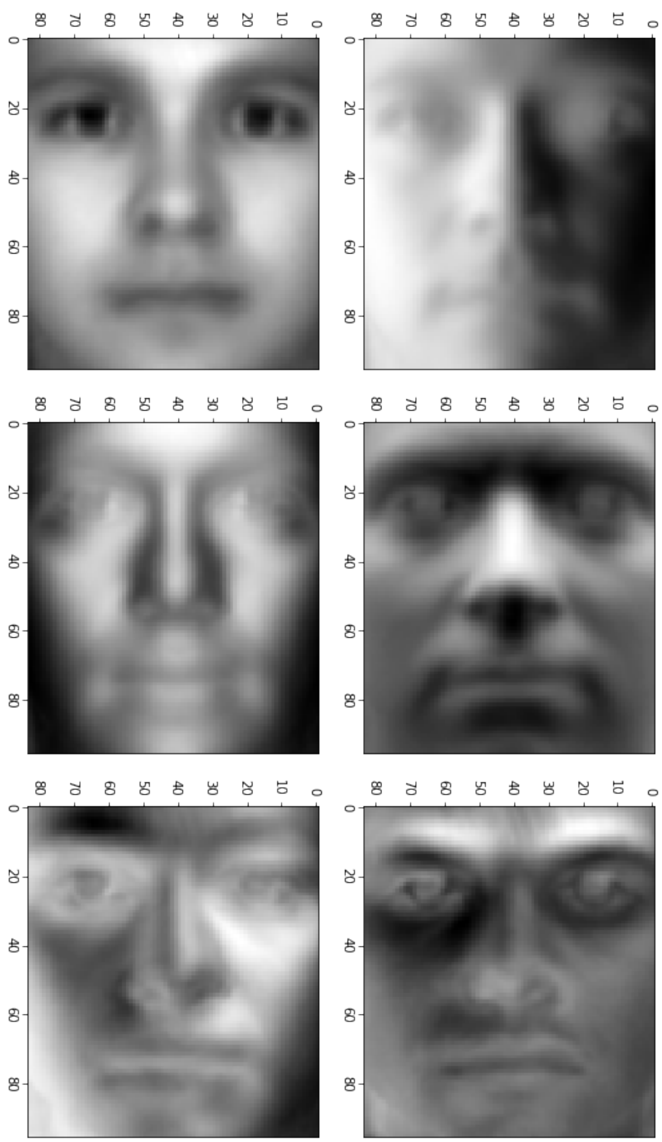
2. The fractional reconstruction error vs k plot is as follows:



3. What we actually do here is that we conduct the orthogonal transformation on Σ and let the after-transformed matrix $\Sigma' = \text{P}\Sigma P^T$ be a diagonal matrix (actually variance matrix of Y = PX, which stores the information of Y). Also the diagonal elements are actually the eigenvalues of Σ. So eigenvalues of Σ store information of Y. And the first eigenvalue captures the largest variance, the most information of Y.

Answer to Question 1(c)

1. Below are plots of the first 10 eigenvectors:



2. The eigenvectors also capture the information, the variance of the data. Degree of dispersion of X along the direction of eigenvectors represents the variance along this direction. The more spread out the projected points along the eigenvector are, the more variance/information X has along this direction.

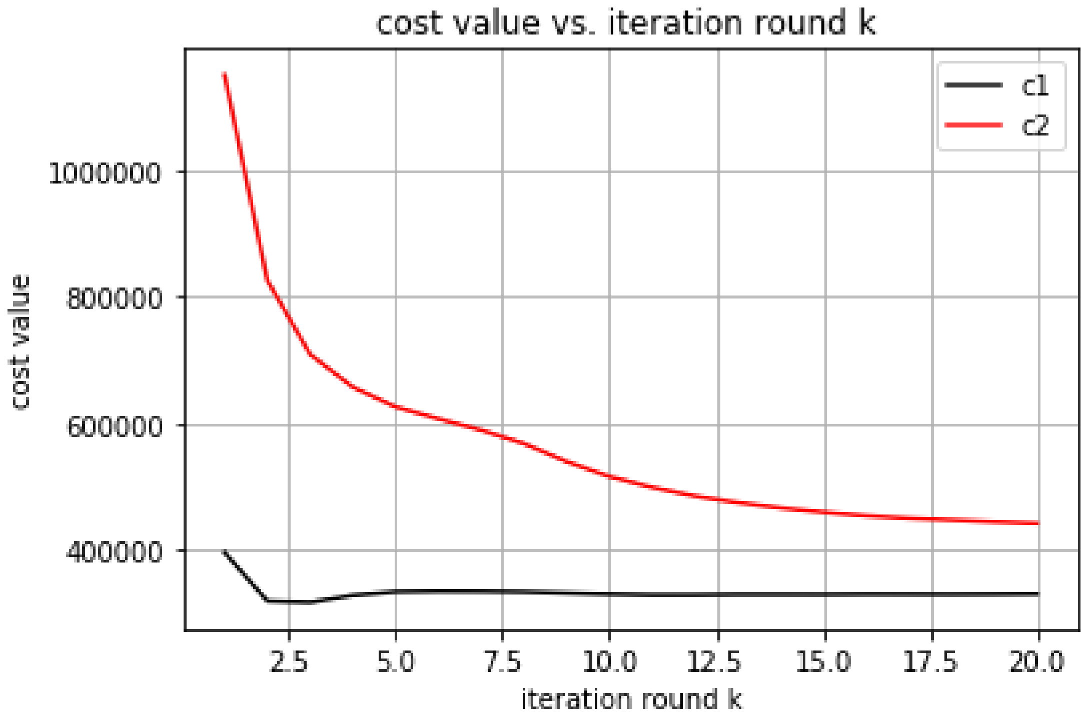
Answer to Question 1(d)

1. The 5×6 plot table is shown below:

2. As we can see, the image are becoming more and more similar to the original images when we increase k from 1 to 50. Back to the reconstruction error plot of question 1(b), the error rate reduces significantly when we increase k. The plots table also reflects this point, and indicates that reconstruction matrix stores more information when we increase k. And when k is increased to d=8064, the reconstruction matrix is identity matrix and thus will contain all information of the original matrix.

Answer to Question 2(a)

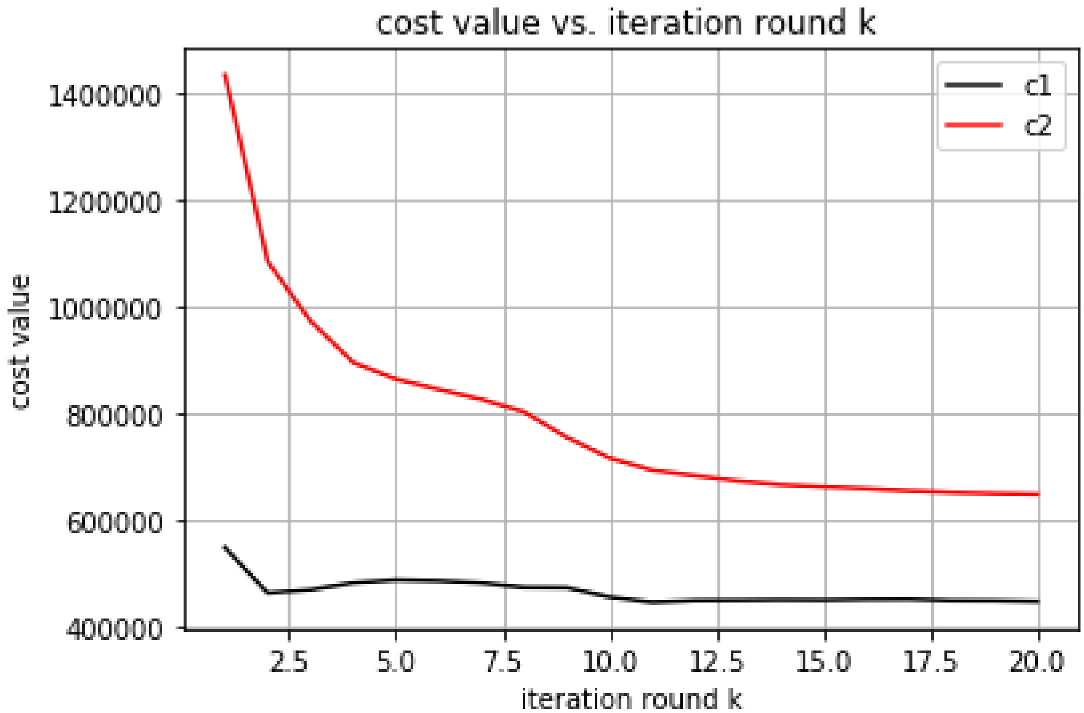
1. For cost function $\Phi(C) = \sum_x \min_{\forall c \in C} \|x - c\|_2$, the plot is as follows:



2. c1 improves 16.9% after 10-round iterations while c2 improves 56.7%, which suggests that c2 is better than c1. I think this is because c2 puts the initial points as far as possible, leading to less overlap and thus better selection of clusters.

Answer to Question 2(b)

1. For cost function $\Phi(C) = \sum_x \min_{\forall c \in C} \|x - c\|_1$, the plot is as follows:



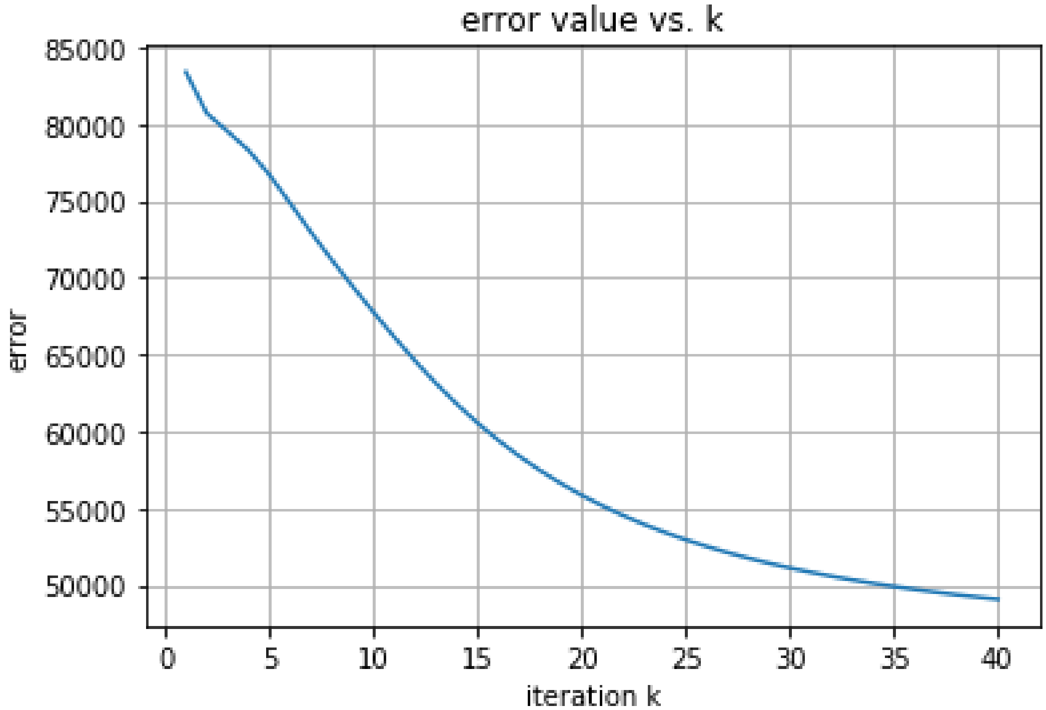
2. c1 improves 18.7% after 10-round iterations while c2 improves 51.6%, which suggests that c2 is better than c1 due to the more spread out initial clusters.

Answer to Question 3(a)

$\epsilon_{iu} = 2(R_{iu} - q_i p_u^T)$
$q_i = q_i + \eta(\epsilon_{xi} p_u - 2\lambda q_i)$
$p_u = p_u + \eta(\epsilon_{xi} q_i - 2\lambda p_x)$

Answer to Question 3(b)

Plot of E vs. number of iterations is shown below:



Here we use $\eta = 0.01$

Answer to Question 4(a)

T_{ii} represents the number of nodes of user i.
T_{ij} represents the number of items user i and user j both like, or equivalently, represents the path of length 2 bewteen i and j.

Answer to Question 4(b)

First of all, cos-sim(i,j) = $\frac{i \cdot j}{\|i\| \|j\|}$ and,
$\|i\| = \sqrt{number\,of\,users\,who\,like\,item\,i} = \sqrt{Q_{ii}}$
$\|j\| = \sqrt{number\,of\,users\,who\,like\,item\,j} = \sqrt{Q_{jj}}$
Thus we have cos-sim(i,j) = $\frac{R_{\cdot i} R_{\cdot j}}{\sqrt{Q_{ii}}\sqrt{Q_{jj}}}$ where $R_{\cdot i}$ and $R_{\cdot j}$ represent the i-th and j-th column of R, respectively. And,
$\frac{R_{\cdot i} R_{\cdot j}}{\sqrt{Q_{ii}}\sqrt{Q_{jj}}} = \sum_k \frac{R_{ki} Rkj}{\sqrt{Q_{ii}}\sqrt{Q_{jj}}} = \sum_k Q_{ii}^{-1/2} R_{ki} R_{kj} Q_{jj}^{-1/2}$
Since Q is a diagonal matrix, and thus $Q^{-1/2}$ is also diagonal. Thus $Q_{ip}^{-1/2} = 0$ for i≠p. Using this property, we have
$\sum_k Q_{ii}^{-1/2} R_{ki} R_{kj} Q_{jj}^{-1/2} = \sum_{p,k,q} Q_{ip}^{-1/2} R_{pk}^T R_{kq} Q_{qj}^{-1/2} = (Q^{-1/2} R^T R Q^{-1/2})_{ij}$.
Thus, $S_I = Q^{-1/2} R^T R Q^{-1/2}$
Similarly, we could have $S_U = P^{-1/2} R R^T P^{-1/2}$. The procedure is almost the same as that of obtaining S_U.

cos-sim(i,j) = $\frac{R_{\cdot i} R_{\cdot j}}{\sqrt{P_{ii}}\sqrt{P_{jj}}} = \sum_k \frac{R_{ki} Rkj}{\sqrt{P_{ii}}\sqrt{P_{jj}}} = \sum_k P_{ii}^{-1/2} R_{ki} R_{kj} P_{jj}^{-1/2} = \sum_{p,k,q} P_{ip}^{-1/2} R_{pk}^T R_{kq} P_{qj}^{-1/2} = (P^{-1/2} R R^T P^{-1/2})_{ij}$.
Thus, $S_I = P^{-1/2} R R^T P^{-1/2}$

Answer to Question 4(c)

For user-user collaborative filtering method, for user i and item j, we have
$\Gamma(i,j) = r_{i,j} = \sum_{x \in users} cossim(x,i) R_{xj} = \sum_x (S_u)_{x,i} R_{xj} = \sum_x (S_u)_{i,x} R_{xj} = (S_u R)_{ij} = (P^{-1/2} R R^T P^{-1/2} R)_{ij}$.
Similarly, for user-user collaborative filtering method,
$\Gamma(i,j) = r_{i,j} = \sum_{x \in items} R_{ix} cossim(x,j) = \sum_x R_{ix} (S_I)_{xj} = (RS_I)_{ij} = (RQ^{-1/2} R^T R Q^{-1/2})$

Answer to Question 4(d)

For user-user:
"FOX 28 News at 10pm" 908.480053
"Family Guy" 861.175999
"2009 NCAA Basketball Tournament" 827.601295
"NBC 4 at Eleven" 784.781959
"Two and a Half Men" 757.601118

For item-item:
"FOX 28 News at 10pm" 31.364702
"Family Guy" 30.001142
"NBC 4 at Eleven" 29.396798
"2009 NCAA Basketball Tournament" 29.227002
"Access Hollywood" 28.971278

Submission Instructions

Assignment Submission All students should submit their assignments electronically via GradeScope. Students may typeset or scan their **neatly written** homeworks (points **will** be deducted for illegible submissions). Simply sign up on the Gradescope website and use the course code 97EWEW. Please use your UW NetID if possible.

For the non-coding component of the homework, you should upload a PDF rather than submitting as images. We will use Gradescope for the submission of code as well. Please make sure to tag each part correctly on Gradescope so it is easier for us to grade. There will be a small point deduction for each mistagged page and for each question that includes code. Put all the code for a single question into a single file and upload it. Only files in text format (e.g. .txt, .py, .java) will be accepted. **There will be no credit for coding questions without submitted code on Gradescope, or for submitting it after the deadline**, so please remember to submit your code.

Late Day Policy All students will be given two no-questions-asked late periods, but only one late period can be used per homework and cannot be used for project deliverables. A late-period lasts 48 hours from the original deadline (so if an assignment is due on Thursday at 11:59 pm, the late period goes to the Saturday at 11:59pm Pacific Time).

Academic Integrity We take academic integrity extremely seriously:
(`https://www.cs.washington.edu/academics/misconduct`).
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 hardcopy documents used as part of your answers:

I acknowledge and accept the Academic Integrity clause.

*(Signed) RUOJIN HE*________________________________