CS550: Massive Data Mining and Learning
Homework 2

Due 11:59pm Friday, March 10, 2023

Submission Instructions

Assignment Submission: Include a signed agreement to the Honor Code with this assignment. Assignments are due at 11:59pm. All students must submit their homework via Canvas. Students can typeset or scan their homework. These questions require thought but do not require long answers. Please be as concise as possible. You should submit your answers as a writeup in PDF format, for those questions that require coding, write your code for a question in a single source code file, and name the file as the question number (e.g., question_1.java or question_1.py), finally, put your PDF answer file and all the code files in a folder named as your Name and NetID (i.e., Firstname-Lastname-NetID.pdf), compress the folder as a zip file (e.g., Firstname-Lastname-NetID.zip), and submit the zip file via Canvas. For the answer writeup PDF file, we have provided both a word template and a latex template for you, after you finished the writing, save the file as a PDF file, and submit both the original file (word or latex) and the PDF file.

Late Policy: The homework is due on 3/10 (Friday) at 11:59pm. We will release the solutions of the homework on Canvas on 3/15 (Wednesday) 11:59pm. If your homework is submitted to Canvas before 3/10 11:59pm, there will no late penalty. If you submit to Canvas after 3/10 11:59pm and before 3/15 11:59pm (i.e., before we release the solution), your score will be penalized by 0.9^k, where k is the number of days of late submission. For example, if you submitted on 3/13, and your original score is 80, then your final score will be $80 \times 0.9^3 = 58.32$ for $13 - 10 = 3$ days of late submission. If you submit to Canvas after 3/15 11:59pm (i.e., after we release the solution), then you will earn no score for the homework.

Honor Code: Students may discuss the homework problems with peers. However, each student must write down their solutions independently to show they understand the solution well enough in order to reconstruct it by themselves. Students should clearly mention the names of all the other students who were part of their discussion group. Using code or solutions directly obtained from the web or others is considered an honor code violation. We check all the submissions for plagiarism. We take the honor code seriously and expect students to do the same.

Discussion Group (People with whom you discussed ideas used in your answers): Abhishek Nayak, Anjali Menghwani

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

I acknowledge and accept the Honor Code.

(Signed)Vikram Sahai Saxena

If you are not printing this document out, please type your initials above.

Answer to Question 1(a)

We have defined a matrix M (of size p × q).

1. $(MM^T)^T = (M^T)^T M^T = MM^T$

   and

   $(M^T M)^T = M^T (M^T)^T = M^T M$

   So, the matrices MM^T and $M^T M$ are symmetric.

2. The matrix M^T will have dimension of q x p. So, matrix MM^T will have a dimension of p x p, and matrix $M^T M$ will have a dimension of q x q. As a result, the matrices MM^T and MM^T are square.

3. Assuming the data points for the matrix M are real, determining matrices MM^T and $M^T M$ will involve only transpose and multiplication operations, which will still yield only real values in the resultant matrices. Hence, the matrices MM^T and $M^T M$ are real.

Answer to Question 1(b)

Let $\mu \neq 0$ be an eigenvalue of $M^T M$.

$\Rightarrow det(M^T M - \mu I) = 0$
$\Rightarrow det(I + \frac{-1}{\mu} M^T M) = 0$
$\Rightarrow det(I + M \frac{-1}{\mu} M^T) = 0$
$\Rightarrow det(MM^T - \mu I) = 0$
$\Rightarrow \mu \neq 0$ is an eigenvalue of matrix MM^T

So, the eigenvalues of MM^T are the same as that of MM^T. However, on evaluating the eigenvectors of MM^T and MM^T, they will be different.

Answer to Question 1(c)

According to the definition of eigenvalue decomposition provided in the beginning of the question, for a real, symmetric and square matrix B, we have:

$B = Q \wedge Q^T$

From Question 1(a), we showed that the matrix $M^T M$ is a real, symmetric and square matrix. So, the eigenvalue decomposition for the matrix $M^T M$ can be written as:

$M^T M = Q \wedge Q^T$

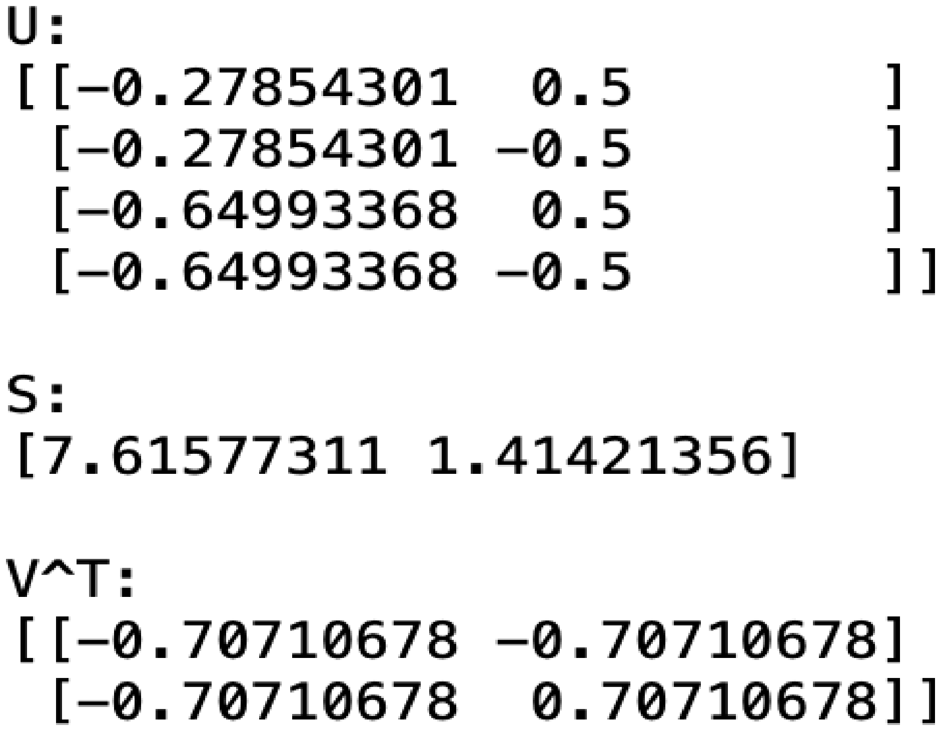
Answer to Question 1(d)

We are given:

$M = U \sum V^T$

$\Rightarrow M^T M = (U \sum V^T)^T U \sum V^T$
$\Rightarrow M^T M = V \sum U^T U \sum V^T$
$\Rightarrow M^T M = V \sum I \sum V^T$
$\Rightarrow M^T M = V (\sum)^2 V^T$

Answer to Question 1(e)(a)

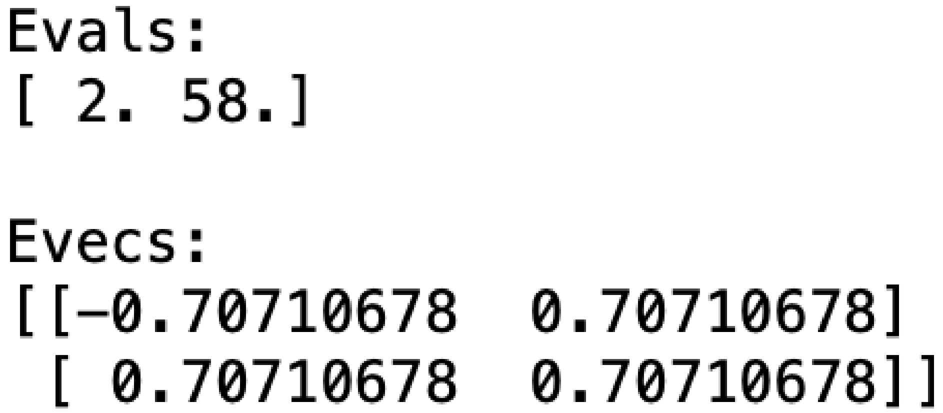


```
U:
[[-0.27854301  0.5        ]
 [-0.27854301 -0.5        ]
 [-0.64993368  0.5        ]
 [-0.64993368 -0.5        ]]

S:
[7.61577311 1.41421356]

V^T:
[[-0.70710678 -0.70710678]
 [-0.70710678  0.70710678]]
```

Answer to Question 1(e)(b)



```
Evals:
[ 2. 58.]

Evecs:
[[-0.70710678  0.70710678]
 [ 0.70710678  0.70710678]]
```

Answer to Question 1(e)(c)

After rearranging the columns, the Evecs generated in eigen value decomposition and V in SVD are the same.

Answer to Question 1(e)(d)

From Question 1(c) and Question 1(d), we get:

$\wedge = (\sum)^2$

Eigenvalue of $M^T M$ are square of singular values of matrix M.

Answer to Question 2(a)

We are given that the web has no dead ends.

$$w(r') = \sum_{i=1}^{n} r'_i$$
$$\Rightarrow w(r') = \sum_{i=1}^{n} \sum_{j=1}^{n} M_{ij} r_j$$
$$\Rightarrow w(r') = \sum_{j=1}^{n} (\sum_{i=1}^{n} M_{ij}) r_j$$

Since the web has no dead cells, $\sum_{i=1}^{n} M_{ij} = 1$

Hence, $w(r') = \sum_{j=1}^{n} r_j = w(r)$

Answer to Question 2(b)

We are given:

$$r'_i = \beta \sum_{j=1}^{n} M_{ij} r_j + \frac{1-\beta}{n}$$

Also, $w(r') = \sum_{j=1}^{n} r_j$

$$\Rightarrow w(r') = \sum_{i=1}^{n} (\beta \sum_{j=1}^{n} M_{ij} r_j + \frac{1-\beta}{n})$$
$$\Rightarrow w(r') = \sum_{i=1}^{n} \beta \sum_{j=1}^{n} M_{ij} r_j + \sum_{i=1}^{n} \frac{1-\beta}{n}$$
$$\Rightarrow w(r') = \sum_{j=1}^{n} \beta \sum_{i=1}^{n} M_{ij} r_j + n \frac{1-\beta}{n}$$
$$\Rightarrow w(r') = \beta \sum_{j=1}^{n} r_j + (1-\beta) = \beta w(r) + (1-\beta)$$
$$Considering, w(r') = w(r)$$
$$\Rightarrow w(r) = \beta w(r) + (1-\beta)$$
$$\Rightarrow w(r) = \frac{1-\beta}{1-\beta} = 1$$

Hence, $w(r') = w(r)$ when both values are equal to 1.

Answer to Question 2(c)(a)

The equation for r'_i in terms of β, M, and r will be:

$$r'_i = \beta \sum_{j=1}^{n} M_{ij} r_j + \frac{1-\beta}{n} + \frac{\beta}{n} \sum_{j \, \epsilon \, dead} r_j$$

Answer to Question 2(c)(b)

We have:

$$w(r'_i) = \sum_{i=1}^{n} (\beta \sum_{j=1}^{n} M_{ij} r_j + \frac{1-\beta}{n} + \frac{\beta}{n} \sum_{j \, \epsilon \, dead} r_j)$$
$$\Rightarrow w(r'_i) = \sum_{i=1}^{n} \beta \sum_{j=1}^{n} M_{ij} r_j + \sum_{i=1}^{n} \frac{1-\beta}{n} + \sum_{i=1}^{n} \frac{\beta}{n} \sum_{j \, \epsilon \, dead} r_j$$

$$\Rightarrow w(r'_i) = \beta \sum_{i=1}^{n} \sum_{j=1}^{n} M_{ij} r_j + n * \frac{1-\beta}{n} + n * \frac{\beta}{n} \sum_{j\,\epsilon\,dead} r_j$$

$$\Rightarrow w(r'_i) = \beta \sum_{i=1}^{n} \sum_{j=1}^{n} M_{ij} r_j + (1-\beta) + \beta \sum_{j\,\epsilon\,dead} r_j$$

Now $\forall j \,\epsilon\, live$, we have:

$$\sum_{i=1}^{n} M_{ij} = 1$$

Similarly, $\forall j \,\epsilon\, dead$, we have:

$$\sum_{i=1}^{n} M_{ij} = 0$$

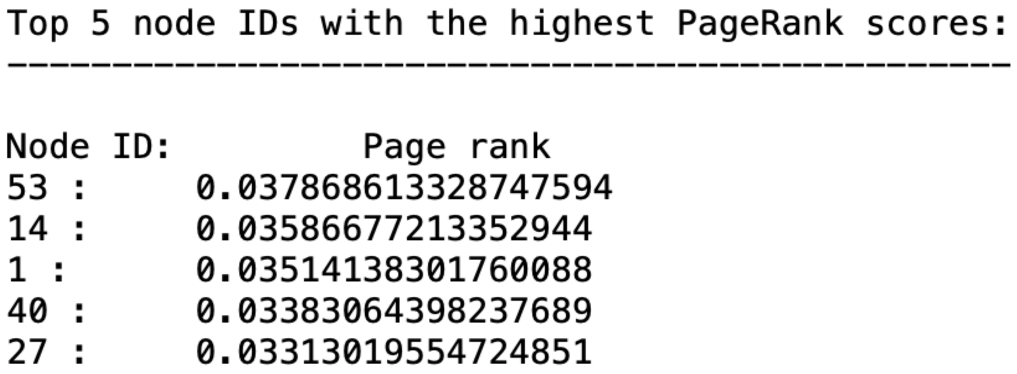
Substituting these two values in the equation for $w(r'_i)$, we get:

$$w(r'_i) = \beta \sum_{j\,\epsilon\,live} (1) r_j + (1-\beta) + \beta \sum_{j\,\epsilon\,dead} r_j$$

$$\Rightarrow w(r'_i) = \beta(\sum_{j\,\epsilon\,live} r_j + \sum_{j\,\epsilon\,dead} r_j) + (1-\beta) = \beta \sum_{j=1}^{n} r_j + (1-\beta)$$

$$\Rightarrow w(r'_i) = \beta w(r) + (1-\beta) = \beta + 1 - \beta = 1$$

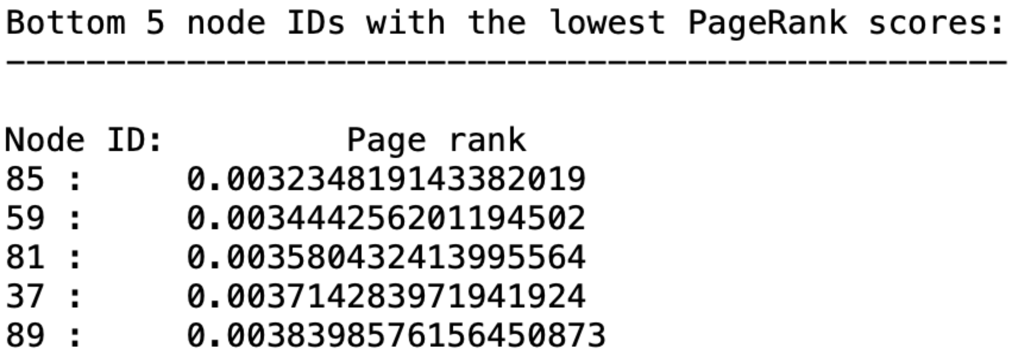
Answer to Question 3(a)



```
Top 5 node IDs with the highest PageRank scores:
------------------------------------------------

Node ID:          Page rank
53 :     0.037868613328747594
14 :     0.03586677213352944
1 :      0.03514138301760088
40 :     0.03383064398237689
27 :     0.03313019554724851
```

Answer to Question 3(b)



```
Bottom 5 node IDs with the lowest PageRank scores:
--------------------------------------------------

Node ID:          Page rank
85 :     0.003234819143382019
59 :     0.003444256201194502
81 :     0.003580432413995564
37 :     0.003714283971941924
89 :     0.0038398576156450873
```

Answer to Question 4(a)

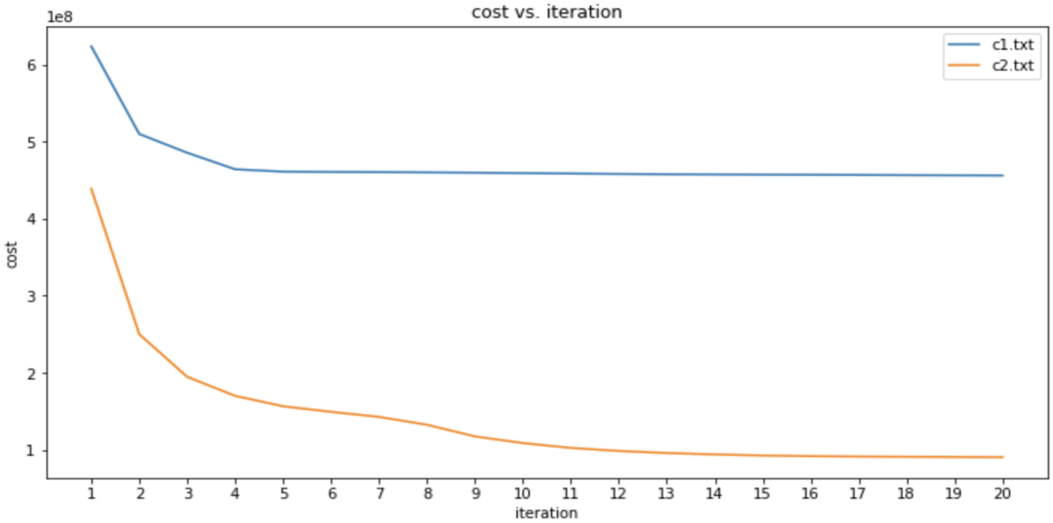
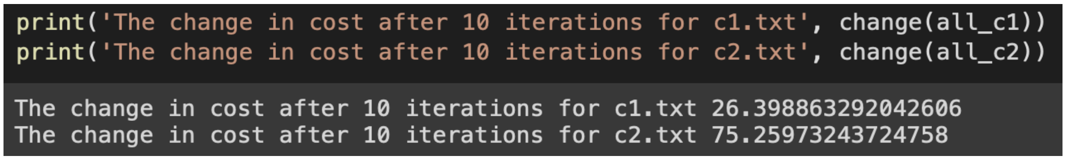


Figure 1: Cost vs Iteration of the two initialization methods

Answer to Question 4(b)



```
print('The change in cost after 10 iterations for c1.txt', change(all_c1))
print('The change in cost after 10 iterations for c2.txt', change(all_c2))
```

```
The change in cost after 10 iterations for c1.txt 26.398863292042606
The change in cost after 10 iterations for c2.txt 75.25973243724758
```

From the above image, we can see that after 10 iterations, c1 improves by 26% and c2 improves by 75%.

Reason for improvement:

The clusters are initialized randomly in the c1.txt file. In c2.txt file, the clusters are initialized as far apart as possible. So, the cluster initialization from c2.txt is much better as the clusters are as far as possible and many of the points are classified correctly from the initial start itself. Hence, the algorithm requires less number of iterations to converge when compared to the random initialization method.