# CS550: Massive Data Mining and Learning
# Homework 2

Due 11:59pm Monday, March 21, 2022

# 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 3/21 (Monday) at 11:59pm. We will release the solutions of the homework on Canvas on 3/25 (Friday) 11:59pm. If your homework is submitted to Canvas before 3/21 11:59pm, there will no late penalty. If you submit to Canvas after 3/21 11:59pm and before 3/25 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 3/24, and your original score is 80, then your final score will be $80 \times 0.9^3 = 58.32$ for $24 - 21 = 3$ days of late submission. If you submit to Canvas after 3/25 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):

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

I acknowledge and accept the Honor Code.

*(Signed)*SK_____

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

## Answer to Question 1(a)

Both the matrices $MM^T$ and $M^TM$ are symmetric, square and real.

- **Symmetric** $(M^TM)^T = M^TM$ and $(MM^T)^T = MM^T$.
  Thus, both matrices are symmetric

- **Square** M is a matrix of size p * q. If we multiply matrix size p * q with its transpose, we will get $MM^T$ is a size p * p. Similarly, $M^TM$ is size q * q.

- **Real** As M is real, $M^T$ will be real and their multiplication will be real.

## Answer to Question 1(b)

Let v be the eigenvector and $\lambda$ be the eigenvalue of $M^T M$.

$$M^T M(v) = \lambda(v)$$
$$MM^T M(v) = \lambda M(v)$$
$$MM^T(Mv) = \lambda(Mv)$$

As we can see, eigenvalue of $MM^T$ is $\lambda$ and eigenvector is $Mv$.
Therefore, $MM^T$ and $M^T M$ have same eigenvalues but different eigenvectors.

## Answer to Question 1(c)

We proved above that $M^T M$ is a real, symmetric and square matrix. We can write it's eigenvalue decomposition as $M^T M = Q \Lambda Q^T$

## Answer to Question 1(d)

$$
\begin{aligned}
M &= U \sum V^T \\
M^T M &= (U \sum V^T)^T (U \sum V^T) \\
&= (V(\sum)^T U^T)(U \sum V^T) \\
&= V(\sum)^T (U^T U) \sum V^T \\
&= V(\sum)^T I \sum V^T \\
&= V(\sum)^2 V^T
\end{aligned}
$$

## Answer to Question 1(e)(a)

Code is written in question_1e.ipynb and results are as follows:

```
U =  [[-0.27854301  0.5       ]
 [-0.27854301 -0.5       ]
 [-0.64993368  0.5       ]
 [-0.64993368 -0.5       ]]
Sigma= [7.61577311 1.41421356]
V_transpose=  [[-0.70710678 -0.70710678]
 [-0.70710678  0.70710678]]
```

## Answer to Question 1(e)(b)

Code is written in question_1e.ipynb and results are as follows:

```
Evals =
 [58.  2.]
Evecs =
 [[ 0.70710678 -0.70710678]
 [ 0.70710678  0.70710678]]
```

## Answer to Question 1(e)(c)

Matrix V produced by SVD is equivalent to matrix of eigenvectors if we reorder the columns based on ordering of singular values.

## Answer to Question 1(e)(d)

Code is written in question_1e.ipynb and results are as follows:

```
Eigen values of M_Transpose_M =  [58.  2.]
Singular values of M =  [7.61577311 1.41421356]
sqrt MtM = [7.61577311 1.41421356]
```

Singular values of M are square roots of eigenvalues of $M^T M$

## Answer to Question 2(a)

We know that Web has no dead ends.

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

The term in the bracket corresponds to column sum for each column of matrix M and it will be 1 as there are no dead ends.

$$\begin{aligned} w(r') &= \sum_{j=1}^{n} r_j \\ w(r') &= w(r) \end{aligned}$$

## Answer to Question 2(b)

We have,

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

(1 - $\beta$) is teleportation probability. So,

$$\begin{aligned} w(r') &= \sum_{i=1}^{n} r'_j \\ &= \sum_{i=1}^{n} (\beta \sum_{j=1}^{n} M_{ij} r_j + \frac{(1-\beta)}{n}) \\ &= \beta \sum_{i=1}^{n} \sum_{j=1}^{n} M_{ij} r_j + \sum_{i=1}^{n} \frac{(1-\beta)}{n} \\ &= \beta \sum_{j=1}^{n} r_j + \frac{(1-\beta)}{n} n \\ &= \beta w(r) + (1-\beta) \end{aligned}$$

For w(r') = w(r),

$$\begin{aligned} w(r) &= \beta w(r) + (1-\beta) \\ 1 &= \beta + \frac{(1-\beta)}{w(r)} \\ (1-\beta) w(r) &= (1-\beta) \\ w(r) &= 1 \end{aligned}$$

Therefore, w(r) = w(r') = 1

## Answer to Question 2(c)(a)

$$
\begin{aligned}
r'_i &= \beta \sum_{j=1}^{n} M_{ij} r_j + \frac{(1-\beta)}{n} \sum_{j \in live} r_j + \frac{1}{n} \sum_{j \in dead} r_j \\
&= \beta \sum_{j=1}^{n} M_{ij} r_j + \frac{(1-\beta)}{n} \sum_{j \in live} r_j + \frac{(1-\beta)+\beta}{n} \sum_{j \in dead} r_j \\
&= \beta \sum_{j=1}^{n} M_{ij} r_j + \frac{(1-\beta)}{n} \sum_{j \in live} r_j + \frac{(1-\beta)}{n} \sum_{j \in dead} r_j + \frac{\beta}{n} \sum_{j \in dead} r_j \\
&= \beta \sum_{j=1}^{n} M_{ij} r_j + \frac{(1-\beta)}{n} \sum_{j} r_j + \frac{\beta}{n} \sum_{j \in dead} r_j
\end{aligned}
$$

We already know from above that w(r) = 1, i.e., $\sum_{j=1}^{n} r_j = 1$
So,

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

## Answer to Question 2(c)(b)

$$w(r') = \sum_{i=1}^{n} r'_j$$

We already know $r'_j$ from above, So

$$\begin{aligned} w(r') &= \sum_{i=1}^{n}(\beta \sum_{j=1}^{n} M_{ij} r_j + \frac{(1-\beta)}{n} + \frac{\beta}{n} \sum_{j \in dead} r_j) \\ &= \beta \sum_{i=1}^{n} \sum_{j=1}^{n} M_{ij} r_j + \sum_{i=1}^{n} \frac{(1-\beta)}{n} + \sum_{i=1}^{n} \frac{\beta}{n} \sum_{j \in dead} r_j \\ &= \beta \sum_{j=1}^{n} \sum_{i=1}^{n} M_{ij} r_j + \frac{(1-\beta)}{n} n + \beta \sum_{j \in dead} r_j \\ &= \beta \sum_{j=1}^{n} \sum_{i=1}^{n} M_{ij} r_j + (1-\beta) + \beta \sum_{j \in dead} r_j \end{aligned}$$

We know that $\sum_{j=1}^{n} M_{ij} = 1$ for all j $\in$ live and $\sum_{j=1}^{n} M_{ij} = 0$ for all j $\in$ dead.

$$\begin{aligned} w(r') &= \beta \sum_{j \in live}^{n} (1) r_j + (1-\beta) + \beta \sum_{j \in dead} r_j \\ &= \beta (\sum_{j \in live}^{n} r_j + \sum_{j \in dead} r_j) + (1-\beta) \\ &= \beta (\sum_{j=1}^{n} r_j) + (1-\beta) \\ &= \beta w(r) + (1-\beta) \\ &= \beta + (1-\beta) \\ &= 1 \end{aligned}$$

Therefore, w(r') = 1

## Answer to Question 3(a)

Code is written in question_3.ipynb

Top 5 node IDs with the highest PageRank scores:-

| node ID | Page rank |
|---|---|
| 53 | 0.037868613328747594 |
| 14 | 0.03586677213352943 |
| 1 | 0.03514138301760088 |
| 40 | 0.03383064398237689 |
| 27 | 0.033130195547248505 |

## Answer to Question 3(b)

Code is written in question_3.ipynb

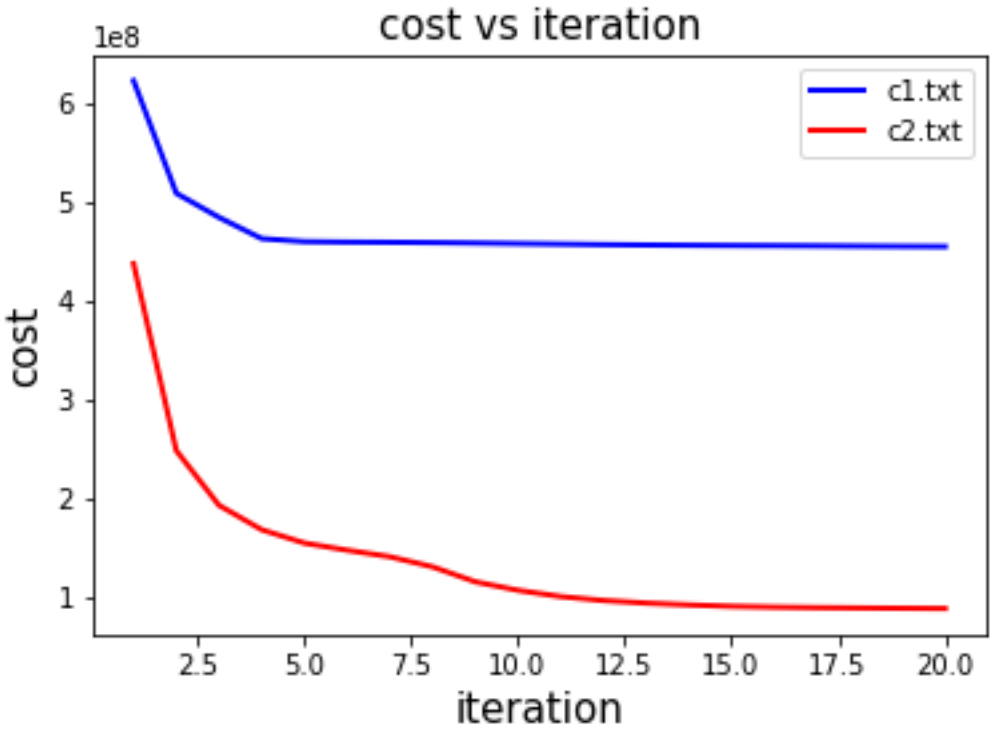
Top 5 node IDs with the lowest PageRank scores:-

| Node ID | Page rank |
|---|---|
| 85 | 0.0032348191433820 19 |
| 59 | 0.0034442562011945 02 |
| 81 | 0.0035804324139955 64 |
| 37 | 0.0037142839719419 24 |
| 89 | 0.0038398576156450873 |

## Answer to Question 4(a)

Code is written in question_4.ipynb

Plot of cost vs. iteration for two initialization strategies:



## Answer to Question 4(b)

Percentage change in cost after 10 iterations of the k-Means algorithm when the cluster centroids are initialized using **c1.txt = 26.48%**

Percentage change in cost after 10 iterations of the k-Means algorithm when the cluster centroids are initialized using **c2.txt = 76.70%**

Initialization using c2.txt is better than random initialization using c1.txt and this we can see in the above graph. Cluster centroids are chosen randomly in c1.txt and initial cluster centroids are as far away as possible in c2.txt. As clusters are spread out in vector space, every point in data will be mapped to nearby cluster from 1st iteration itself. Therefore, initialization using c2.txt is converging faster than initialization using c1.txt.