CS550: Massive Data Mining and Learning Spring 2019
Problem Set 2
Due 11:59pm Saturday, March 23, 2019

Only one late period is allowed for this homework (11:59pm Sunday 3/24)

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 Sakai. Students can typeset or scan their homework. Students also need to include their code in the final submission zip file. Put all the code for a single question into a single file.

Late Day Policy: Each student will have a total of **two** free late days, and for each homework only one late day can be used. If a late day is used, the due date is 11:59pm on the next day.

Honor Code: Students may discuss and work on homework problems in groups. This is encouraged. 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 obtained from the web 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):

None

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

I acknowledge and accept the Honor Code.

(Signed)_Haoyang Zhang_____

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

Answer to Question 1(a)

Symmetric:

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

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

Square:

Size of MM^T: $[p \times q][q \times p] \Rightarrow [p \times p]$

Size of $M^T M$: $[q \times p][p \times q] \Rightarrow [q \times q]$

Real:

Since M is real, $M^T M$ and MM^T are real.

Answer to Question 1(b)

Say $\lambda \neq 0$ is an eigenvalue of $M^T M$.

We have $M^T Mx = \lambda x$, for $x \neq 0$.

Therefore, $MM^T(Mx) = \lambda(Mx)$. Here we have $Mx \neq 0$ because $\lambda x = M^T(Mx) \neq 0$.

Namely, (λ, Mx) is a pair of eigenvalues and eigenvectors of MM^T.

Similarly, we have $M^T M(M^T x) = \lambda(M^T x)$.

Therefore, all non-zero eigenvalues of $\mathrm{M^T M}$and MM^T are the same, but the eigenvectors are different. (Because it is probable that $Mx \neq x, M^T x \neq x$.)

Answer to Question 1(c)

$M^T M = Q \Lambda Q^T$

Answer to Question 1(d)

$M^T M = (U\Sigma V)^T U\Sigma V = V^T \Sigma^2 V$

Answer to Question 1(e)(a)

```
U:
[[-0.27854301  0.5       ]
 [-0.27854301 -0.5       ]
 [-0.64993368  0.5       ]
 [-0.64993368 -0.5       ]]
Sigma:
[7.61577311 1.41421356]
V^T:
[[-0.70710678 -0.70710678]
 [-0.70710678  0.70710678]]
```

Answer to Question 1(e)(b)

```
Evals:
[58.  2.]
Evces:
[[ 0.70710678  0.70710678]
 [-0.70710678  0.70710678]]
```

Answer to Question 1(e)(c)

```
V:
[[-0.70710678 -0.70710678]
 [-0.70710678  0.70710678]]
Evces
[[ 0.70710678  0.70710678]
 [-0.70710678  0.70710678]]
```

Notice that the sign of v_1 is different, but it does not matter. Therefore, $V = Evecs$.

Answer to Question 1(e)(d)

```
Sigma^2:
[58.  2.]
Evals:
[58.  2.]
```

$\Sigma^2 = Evals$

Answer to Question 2(a)

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

Answer to Question 2(b)

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

Let $w(r') = w(r)$:

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

Answer to Question 2(c)(a)

Notice that if r_j is dead, we have $\sum_{j=1}^{n} M_{ij} = 0$.

$$r_i' = \beta \sum_{j=1}^{n} M_{ij} r_j + (1-\beta)\frac{1}{n} + \sum_{l=1}^{n} \beta \left(1 - \sum_{k=1}^{n} M_{kl}\right) \frac{r_l}{n}$$

Answer to Question 2(c)(b)

Notice that $1 - \sum_{k=1}^{n} M_{kl}$ is the indicator that node l is dead.

$$w(r') = \beta \sum_{live} w(r) + (1-\beta) + \beta \sum_{dead} w(r) = \beta w(r) + (1-\beta) = 1$$

Answer to Question 3(a)

If we consider multiple edges between a pair of nodes as only 1 effective edge:

```
53: 0.03373
14: 0.03217
40: 0.03163
1: 0.02801
27: 0.02772
```

If we consider that all multiple edges are effective:

```
53: 0.03587
14: 0.03387
1: 0.03314
40: 0.03183
27: 0.03113
```

Answer to Question 3(b)

If we consider multiple edges between a pair of nodes as only 1 effective edge:

```
85: 0.00141
59: 0.00167
81: 0.00170
37: 0.00181
89: 0.00192
```

If we consider that all multiple edges are effective:

```
85: 0.00123
59: 0.00144
81: 0.00158
37: 0.00171
89: 0.00184
```

Answer to Question 4(a)

Here is the result: (Fig.1)

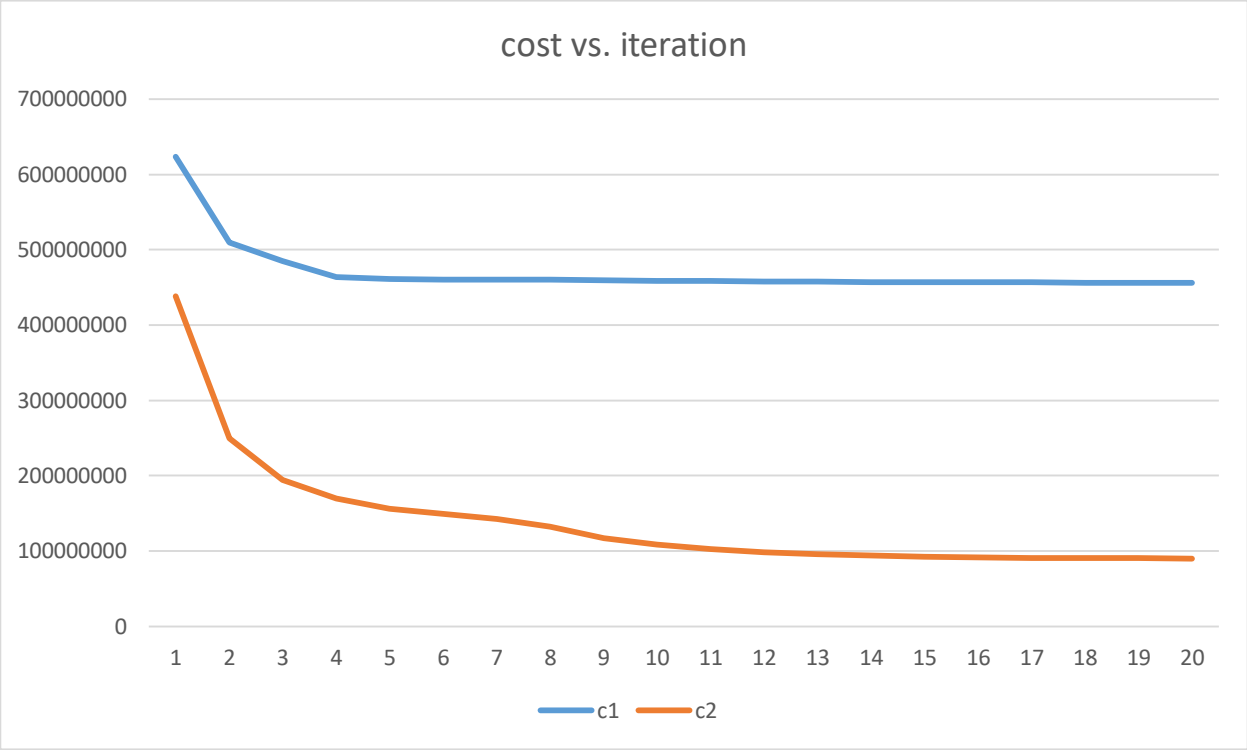

Fig.1: cost vs. iteration of using c1.txt and c2.txt

Answer to Question 4(b)

If we define percentage change in cost after 10 iterations as $\frac{\phi_1 - \phi_{11}}{\phi_1}$, we have:

Using c1.txt: $\frac{623660345.306 - 458490656.192}{623660345.306} = 26.484\%$

Using c2.txt: $\frac{438747790.028 - 102237203.318}{438747790.028} = 76.698\%$

(If not, I attached the cost of each iteration on next page to calculate the percentage change.)

Using c2.txt is better that c1.txt in terms of cost.

Notice that if we choose centroids as far as possible, it will initially be a not bad cluster. (It is showed by ϕ_1 of c2.txt is much less than c1.txt.)

Also, it will be much less likely that a cluster is stuck in the middle of other clusters. Consider this situation. (Fig. 2)

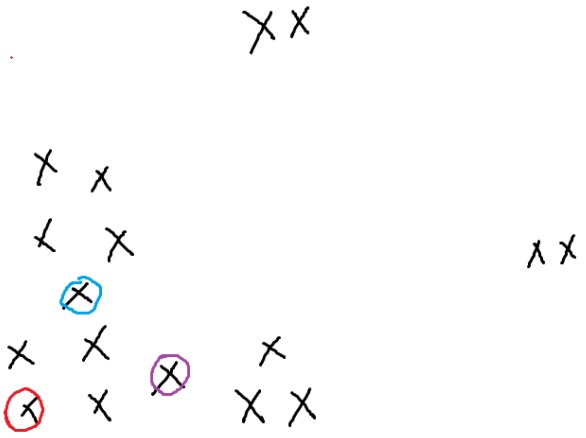

Fig.2 a cluster is stuck in the middle of other clusters.

If we initialize 3 centroids with this 3 colored points, we will immediately find the red cluster is stuck. Only a small number of nodes will be classified to red, and then it will not move a lot. It will take a long time to move the blue and purple away, (and it could be even longer if these 2 are also stuck in the middle of some other clusters,) so that the red centroid will be "freed". It is caused by choosing centroids too closed, and therefore, we can avoid it by choosing them as far as possible.

Cost using c1.txt:

```
1: 623660345.306
2: 509862908.298
3: 485480681.872
4: 463997011.685
5: 460969266.573
6: 460537847.983
7: 460313099.654
8: 460003523.889
9: 459570539.318
10: 459021103.342
11: 458490656.192
12: 457944232.588
13: 457558005.199
14: 457290136.352
15: 457050555.060
16: 456892235.615
17: 456703630.737
18: 456404203.019
19: 456177800.542
20: 455986871.027
```

Cost using c2.txt:

```
1: 438747790.028
2: 249803933.626
3: 194494814.406
4: 169804841.452
5: 156295748.806
6: 149094208.109
7: 142508531.620
8: 132303869.407
9: 117170969.837
10: 108547377.179
11: 102237203.318
12: 98278015.750
13: 95630226.122
14: 93793314.051
15: 92377131.968
16: 91541606.254
17: 91045573.830
18: 90752240.101
19: 90470170.181
20: 90216416.176
```