CS550: Massive Data Mining and Learning Spring 2019
Problem Set 3
Due 11:59pm Saturday, Apr 20, 2019

Only one late period is allowed for this homework (11:59pm Sunday Apr 21)

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:

None

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)

Let $Bij = Aij - \frac{kikj}{2m}$. $Q = \frac{1}{4m} s^T Bs$.

We have $s_i = 1$ when $i \leq 4$, and $s_i = -1$ when $i \geq 5$.

Therefore, we can use `question_1.py` to compute Q

If we ignore edge (A, G):
Here is the result:

```
A:
[[0 1 1 1 0 0 0 0]
 [1 0 1 1 0 0 0 0]
 [1 1 0 1 0 0 0 0]
 [1 1 1 0 0 0 0 0]
 [0 0 0 0 0 1 1 0]
 [0 0 0 0 1 0 1 0]
 [0 0 0 0 1 1 0 1]
 [0 0 0 0 0 0 1 0]]
Q:
[[0.48]]
```

If we count edge (A, G):
Here is the result:

```
A:
[[0 1 1 1 0 0 1 0]
 [1 0 1 1 0 0 0 0]
 [1 1 0 1 0 0 0 0]
 [1 1 1 0 0 0 0 0]
 [0 0 0 0 0 1 1 0]
 [0 0 0 0 1 0 1 0]
 [1 0 0 0 1 1 0 1]
 [0 0 0 0 0 0 1 0]]
Q:
[[0.39256198]]
```

Answer to Question 1(b)

Here is the result:

```
A:
[[0 1 1 1 0 0 1 0]
 [1 0 1 1 0 0 0 0]
 [1 1 0 1 0 0 0 0]
 [1 1 1 0 0 0 0 0]
 [0 0 0 0 0 1 1 1]
 [0 0 0 0 1 0 1 0]
 [1 0 0 0 1 1 0 1]
 [0 0 0 0 1 0 1 0]]
Q:
[[0.41319444]]
```

Compared with `0.39256198`, the modularity went up because edge (E, H) is in the second community. It makes the second community more tight. Or more specifically, it increases "the number of edges within the second group" more than "the expected number of that".

Therefore, the modularity increased.

Answer to Question 1(c)

Here is the result:

```
A:
[[0 1 1 1 0 1 1 0]
 [1 0 1 1 0 0 0 0]
 [1 1 0 1 0 0 0 0]
 [1 1 1 0 0 0 0 0]
 [0 0 0 0 0 1 1 0]
 [1 0 0 0 1 0 1 0]
 [1 0 0 0 1 1 0 1]
 [0 0 0 0 0 0 1 0]]
Q:
[[0.31944444]]
```

Compared with `0.39256198`, the modularity went down because edge (A, F) is between 2 communities. It makes both community comparatively less tight. Or more specifically, it increases "the expected number of edges within each group" but does not increase "the real number of that".

Therefore, the modularity decreased.

Answer to Question 2(a)

Using `Qa()` from `question_2.py`, we can compute the degree matrix and Laplacian matrix using adjacency matrix.

Here is the result:

```
A:
[[0 1 1 1 0 0 1 0]
 [1 0 1 1 0 0 0 0]
 [1 1 0 1 0 0 0 0]
 [1 1 1 0 0 0 0 0]
 [0 0 0 0 0 1 1 0]
 [0 0 0 0 1 0 1 0]
 [1 0 0 0 1 1 0 1]
 [0 0 0 0 0 0 1 0]]
D:
[[4 0 0 0 0 0 0 0]
 [0 3 0 0 0 0 0 0]
 [0 0 3 0 0 0 0 0]
 [0 0 0 3 0 0 0 0]
 [0 0 0 0 2 0 0 0]
 [0 0 0 0 0 2 0 0]
 [0 0 0 0 0 0 4 0]
 [0 0 0 0 0 0 0 1]]
L:
[[ 4 -1 -1 -1  0  0 -1  0]
 [-1  3 -1 -1  0  0  0  0]
 [-1 -1  3 -1  0  0  0  0]
 [-1 -1 -1  3  0  0  0  0]
 [ 0  0  0  0  2 -1 -1  0]
 [ 0  0  0  0 -1  2 -1  0]
 [-1  0  0  0 -1 -1  4 -1]
 [ 0  0  0  0  0  0 -1  1]]
```

Answer to Question 2(b)

Here is the result:
Each row is a eigenvector.
(Note: because of the precision, the smallest eigenvalue is not perfect zero.)

```
eigenvalues:
[-1.70863910e-16  3.54248689e-01  1.00000000e+00  3.00000000e+00
  4.00000000e+00  4.00000000e+00  4.00000000e+00  5.64575131e+00]
eigenvectors
[[ 3.53553391e-01  3.53553391e-01  3.53553391e-01  3.53553391e-01
   3.53553391e-01  3.53553391e-01  3.53553391e-01  3.53553391e-01]
 [-2.47017739e-01 -3.82527662e-01 -3.82527662e-01 -3.82527662e-01
   3.82527662e-01  3.82527662e-01  2.47017739e-01  3.82527662e-01]
 [ 5.05409087e-19 -1.57998933e-16  3.09183799e-17  3.29325364e-17
   4.08248290e-01  4.08248290e-01  1.27488329e-16 -8.16496581e-01]
 [ 1.27836110e-17 -5.74672591e-16 -6.43678830e-16  1.22633375e-15
  -7.07106781e-01  7.07106781e-01 -7.98233130e-18  2.20669260e-16]
 [-1.27260608e-02 -5.73441158e-01 -2.06411808e-01  7.92579027e-01
   4.24202027e-03  4.24202027e-03 -1.27260608e-02  4.24202027e-03]
 [ 5.77200260e-01 -3.96085495e-01  6.65116150e-02 -2.47626381e-01
  -1.92400087e-01 -1.92400087e-01  5.77200260e-01 -1.92400087e-01]
 [-2.04151676e-01 -4.71820784e-01  8.13205819e-01 -1.37233359e-01
   6.80505586e-02  6.80505586e-02 -2.04151676e-01  6.80505586e-02]
 [-6.62557346e-01  1.42615758e-01  1.42615758e-01  1.42615758e-01
  -1.42615758e-01 -1.42615758e-01  6.62557346e-01 -1.42615758e-01]]
```

Answer to Question 2(c)

Here we have the “second smallest” eigenvector:

```
[-2.47017739e-01 -3.82527662e-01 -3.82527662e-01 -3.82527662e-01
  3.82527662e-01  3.82527662e-01  2.47017739e-01  3.82527662e-0
1]
```

Therefore, the first 4 vertices (A, B, C, D) belongs to a community, and the others (E, F, G, H) belongs to another one.

Answer to Question 3(a)

Here we have $C_i = \{n \mid n = 0 \ mod \ i\} \cap V$. Namely, all vertices shares the common factor i. Therefore, for each vertex pair in C_i, there is an edge between them. Namely, it is a clique.

Answer to Question 3(b)

C_i is a maximal clique if and only if i is a prime number.

When i is a prime number, $\forall v \in V - C_i$, we must have there is no edge between v and i. Because the only factor of i other than 1 is itself, and once v have the factor i, it must be divisible by i, which means $v \in C_i$. Contradiction!

Therefore, we cannot add any more vertices to C_i if i is a prime number . Namely, it is already a maximal clique.

If i is not a prime number, say $i = pq$, where at least q is a prime number, we must have $q \notin C_i$ because $q < i$, and a smaller number cannot be divisible by a larger number. Notice that $\{n \mid n = 0 \ mod \ i\} \subset \{n \mid n = 0 \ mod \ p\}$. Namely, $\forall v \in C_i, (v, \ p) \in E$. Therefore, we can add p into the clique.

Therefore, we still can add more vertices to C_i if i is not a prime number. Namely, it is not a maximal clique.

Answer to Question 3(c)

Notice that $|C_i| = \lfloor \frac{1000000}{i} \rfloor$. Therefore, we have C_2 is the largest clique because 2 is the smallest possible number. Since 2 is a prime number, C_2 is a maximal clique.

Therefore, it is the unique maximal clique.

Some extra thought:

Here is a nontrivial case: S is a clique. Namely, each 2 numbers in S share a common factor. However, there is no common factor over the whole set. Here is an example: $S = \{6, 10, 15\}$.

We can show that there are only 2 kinds of clique: the trivial one, C_i, and the above nontrivial one. (Just by contradiction, assuming there is a set not belonging to them, we can find a vertex pair that is not connected.)

This kind of nontrivial set can be generated by a prime number set. (If there is a nonprime number, we can factorize it to several prime numbers.) Say P is the prime number set. Each element in S can be factorized into at least $\left\lfloor \frac{|P|}{2} \right\rfloor + 1$ prime numbers in P. In this case, each 2 numbers in S have at least $\left\lfloor \frac{|P|}{2} \right\rfloor \times 2 + 2 \geq |P| + 1$ factors in total. Namely, they must share a common factor.

Also we can show that any these nontrivial set can be factorized into these prime number set P. (Or we can show it is not a maximal clique.)

Notice that $2 \times 3 \times 5 \times 7 \times 9 \times 11 \times 13 \times 17 \times 19 = 9699690 < 1000000, 2 \times 3 \times 5 \times 7 \times 9 \times 11 \times 13 \times 17 \times 19 \times 23 = 223092870 > 1000000$. Therefore, the element in S can be at most factorized into 8 prime numbers. Namely, $|P| \leq 15$. Therefore, $|S| < 2^{|P|} = 32768$.

Therefore, any this nontrivial clique is smaller that C_2.