Data Mining Homework - 3

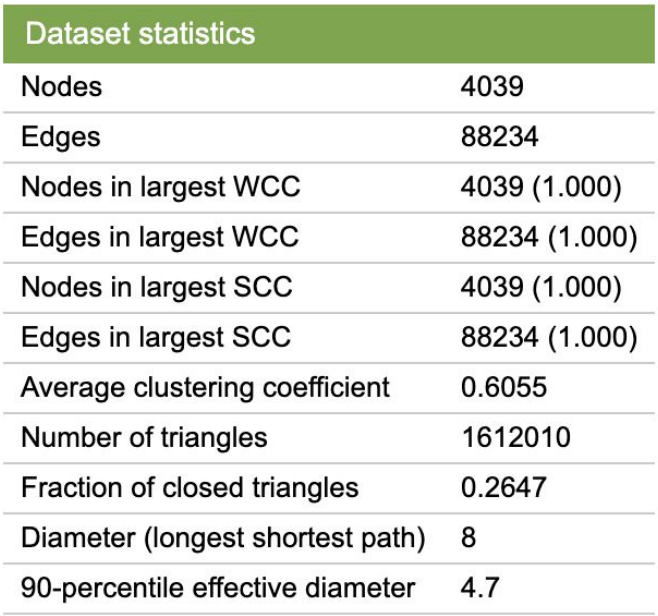
Mining Data Streams

Group – 7

Xing Zeng <xingzeng@kth.se> Sevket Melih Zenciroglu <smzen@kth.se>

1. Dataset

We use graph dataset: Social circles: Facebook Dataset. This dataset consists of 'circles' (or 'friends lists') from Facebook, which includes node features (profiles), circles, and ego networks.



Dataset statistics	
Nodes	4039
Edges	88234
Nodes in largest WCC	4039 (1.000)
Edges in largest WCC	88234 (1.000)
Nodes in largest SCC	4039 (1.000)
Edges in largest SCC	88234 (1.000)
Average clustering coefficient	0.6055
Number of triangles	1612010
Fraction of closed triangles	0.2647
Diameter (longest shortest path)	8
90-percentile effective diameter	4.7

2. Solution

The problem is to sample graph stream data and estimate the number of triangles based on sampling data. We solve this problem by TRIÈST algorithm [2] which is to count Local and Global Triangles in Fully-Dynamic Streams with

Fixed Memory Size.

The idea is based on reservoir sampling which is an algorithm to deal with sampling in a stream and select each element in equal probability. We implement two algorithms in paper, one is Trièst-base. The second algorithm is Trièst-impr which is based on Trièst-base but improves the performance.

- Trièst-base

Trièst-base works on insertion- only streams and uses standard reservoir sampling to maintain the edge sample S. If t(represent one stream) < M(fixed memory size), we keep adding it into S until Fixed Memory is full. Then we select random number x (0<x<1), if x<= M/t, then we randomly choose one sample in S and replace it with this new stream data. Otherwise, discard the new data.

- Trièst-base

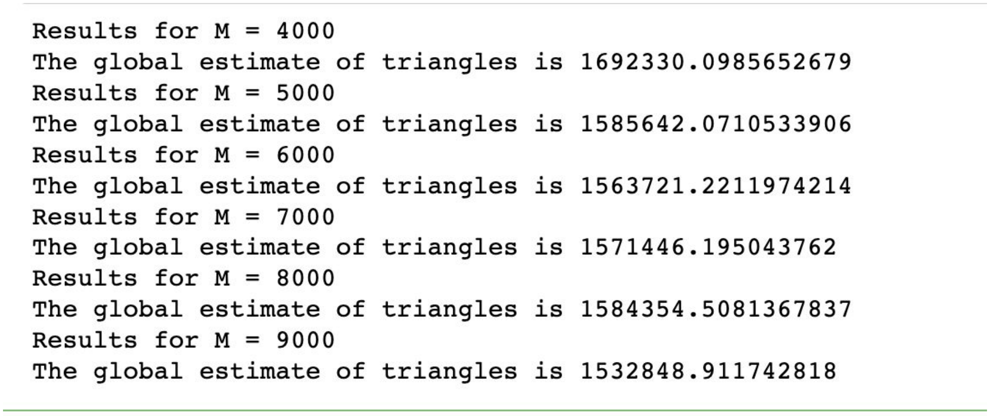
Trièst-impr is a variant of trièst-base with small modifications that result in higher-quality (i.e., lower variance) estimations. These changes are:

- UpdateCounters is called unconditionally for each element on the stream.
- Never decrements the counters when an edge is removed from S.
- UpdateCounters performs a weighted increase of the counters.

3. Result

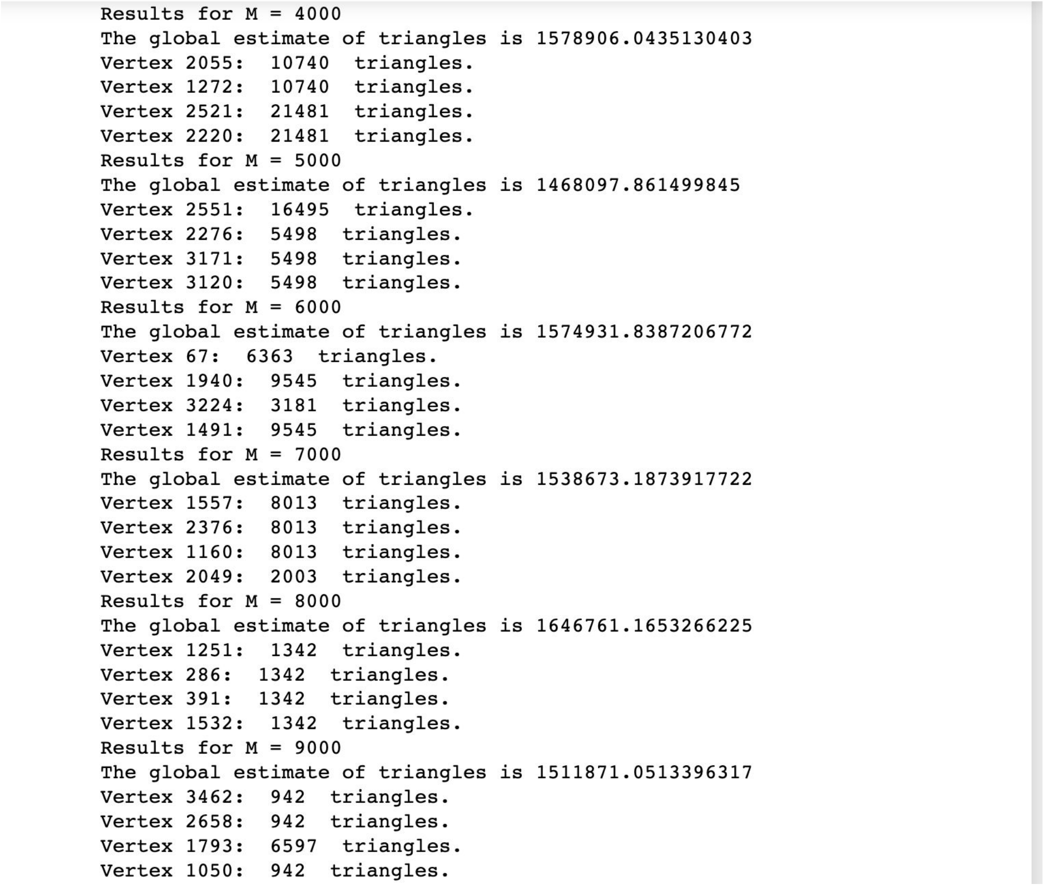
Actual number of triangles is 1612010

Result of Trièst-impr:



```
Results for M = 4000
The global estimate of triangles is 1692330.0985652679
Results for M = 5000
The global estimate of triangles is 1585642.0710533906
Results for M = 6000
The global estimate of triangles is 1563721.2211974214
Results for M = 7000
The global estimate of triangles is 1571446.195043762
Results for M = 8000
The global estimate of triangles is 1584354.5081367837
Results for M = 9000
The global estimate of triangles is 1532848.911742818
```

Result of Trièst-base:



```
Results for M = 4000
The global estimate of triangles is 1578906.0435130403
Vertex 2055:  10740  triangles.
Vertex 1272:  10740  triangles.
Vertex 2521:  21481  triangles.
Vertex 2220:  21481  triangles.
Results for M = 5000
The global estimate of triangles is 1468097.861499845
Vertex 2551:  16495  triangles.
Vertex 2276:  5498  triangles.
Vertex 3171:  5498  triangles.
Vertex 3120:  5498  triangles.
Results for M = 6000
The global estimate of triangles is 1574931.8387206772
Vertex 67:  6363  triangles.
Vertex 1940:  9545  triangles.
Vertex 3224:  3181  triangles.
Vertex 1491:  9545  triangles.
Results for M = 7000
The global estimate of triangles is 1538673.1873917722
Vertex 1557:  8013  triangles.
Vertex 2376:  8013  triangles.
Vertex 1160:  8013  triangles.
Vertex 2049:  2003  triangles.
Results for M = 8000
The global estimate of triangles is 1646761.1653266225
Vertex 1251:  1342  triangles.
Vertex 286:  1342  triangles.
Vertex 391:  1342  triangles.
Vertex 1532:  1342  triangles.
Results for M = 9000
The global estimate of triangles is 1511871.0513396317
Vertex 3462:  942  triangles.
Vertex 2658:  942  triangles.
Vertex 1793:  6597  triangles.
Vertex 1050:  942  triangles.
```

4. Optional task for extra bonus

1. What were the challenges you have faced when implementing the algorithm?
   Ans: I didn't know how to simulate streaming. I use each line as one coming streaming data.

2. Can the algorithm be easily parallelized? If yes, how? If not, why? Explain.
   Ans: Yes, we can partition the graph, and count each part of the graph in parallel.

3. Does the algorithm work for unbounded graph streams? Explain.
   Ans: Yes, since it requires only O(M) space.

4. Does the algorithm support edge deletions? If not, what modification would it need? Explain.
   Ans: No. Edge deletions based on random pairing (RP), a sampling scheme that extends reservoir sampling and can handle deletions. The idea behind the RP scheme is that edge deletions seen on the stream will be "compensated" by future edge insertions.
   In order to realize it, keeping a counter di (resp. do) to keep track of the number of uncompensated edge deletions involving an edge e that was (resp. was not) in S at the time the deletion for e was on the stream.

5. Reference

[1] https://snap.stanford.edu/data/ego-Facebook.html

[2] Lorenzo De Stefani, Alessandro Epasto, Matteo Riondato, and Eli Upfal. 2017. TRIÈST: Counting Local and Global Triangles in Fully Dynamic Streams with Fixed Memory Size. ACM Trans. Knowl. Discov. Data 11, 4, Article 43 (August 2017), 50 pages. DOI:https://doi.org/10.1145/3059194