CMPT 843 Project

Marten Heidemeyer
mheideme@sfu.ca

April 8, 2015

1 Motivation and Application Background

0.5 pages My own application of the method. Also talk about the extra work I did (reproducing the experiments and adding extra measures).

2 Background

Triangles can be a more or less important structure of a graph. If many nodes are reachable from those nodes that are forming a triangle this triangle is important to connect the remaining nodes. In this case the triangle is serving as a hub for the nodes around it. Removing the nodes that form the triangle or the edges between them would disconnect a large part of the graph. The importance of one triangle can be calculated as the number of nodes that are adjacent to it. Let this sum be defined as the *connectivity* of a triangle. Figure 1 shows a graph with three triangles with *connectivity* 15, 14 and 10. The average *connectivity* of triangles in this graph is $\frac{15+14+10}{3} = 13$. If we want to find the average *connectivity* of triangles in a large scale graph it makes sense to sample a number of triangles and observe the *connectivity* of the sampled triangles. In this report I document the application of the *Neighborhood Sampling* method as introduced in the paper "Counting and Sampling Triangles from a Graph Stream" to estimate the average connectivity of the overall population of triangles in the graph. Here I will limit the experiments to triangles for which the adjacent nodes are distributed evenly across the three nodes of the triangle as in the graph of Figure 1.

3 Solution and Analysis

To test how well the presented method is suitable to find the average connectivity of triangles in a graph I will first generate a graph with τ triangles. For each of the τ triangles I draw a random number r between 1 and a and assign each node of the triangle r adjacent nodes (thus this triangle gets $3 \times r$ adjacent nodes). All in all the generated graph consists of only triangles, where each triangle gets a random number of adjacent nodes that are equally distributed among the triangle nodes. Next I shuffle the order of the edges to generate a graph stream in which the edges appear in random order. Now I use the *Neighborhood Sampling* method to sample triangles from this stream where I use e estimators. As described in the paper each estimator can either hold a triangle or not. For the case that an estimator is holding a triangle we take this triangle into our sample with probability $\frac{c}{2\Delta}$ (c is the number of edges the estimator observed that are adjacent to the first edge

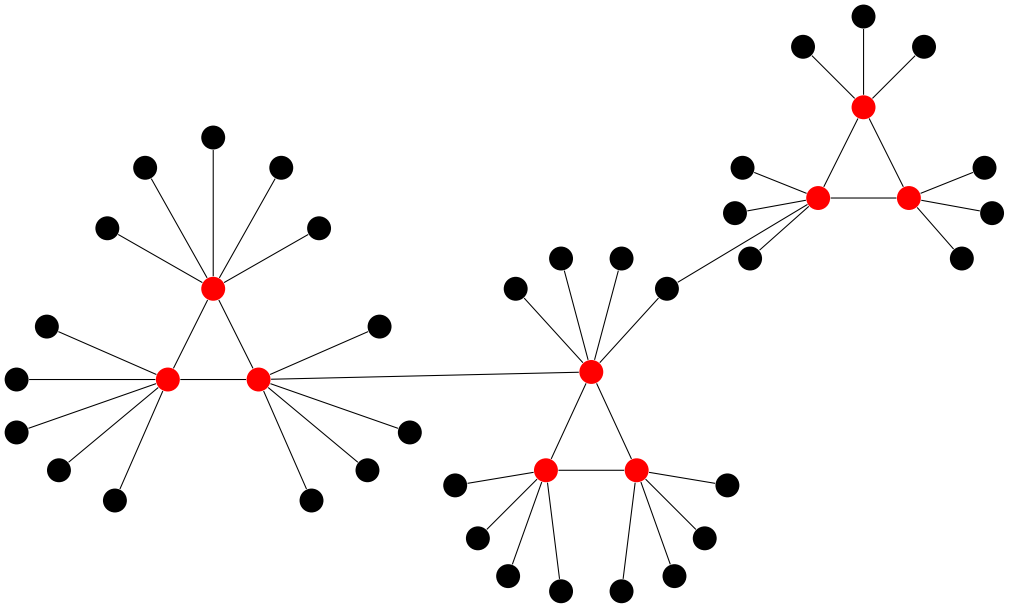

Figure 1: all nodes in the graph are adjacent to one of three triangles

of the triangle that arrived in the stream and Δ is the maximum degree of a node in the graph). For each triangle that I get in my sample I estimate the total number of adjacent nodes to it (the *connectivity* of the triangle) as described in 3.1. Finally I compare the mean and variance of the *connectivity* that I got from my sample with the actual mean and variance of the triangle population.

The sampling method is simple random sampling with replacement. Therefore I expect that I get an unbiased estimator for the overall *connectivity* of the triangles. The size of the samples that the method allows me to draw depends both on the number of estimators and the maximum degree Δ in the generated graph. My estimation should get more accurate the higher I choose the number of estimators e as more estimators return a larger sample. At the same time my estimation should become less accurate, the larger I choose a because a larger a results in a larger value Δ which decreases the size of the sample I can draw from the estimators.

3.1 Estimating the number of adjacent nodes of a triangle

The estimators from which the triangles are being sampled are first sampling a random edge r_1 from the stream, then they sample a random edge r_2 from the substream of edges that arrive after r_1 and are adjacent to it. Finally

they sample an edge r_3 that closes the wedge $r_1 r_2$. Each estimator keeps track of the number of edges it observes that are adjacent to r_1 and arrive after it in the stream. This number is kept in a counter c. Thus c keeps track of the number of edges that arrive after r_1 and which are adjacent to either of two nodes of the triangle (the two nodes of the triangle that are connected by r_1). In the *Neighborhood Sampling* method the counter c is used to decide weather or not an estimator samples an edge as the new r_2 edge (an edge that is adjacent to r_1 is sampled as the new r_2 edge with probability $\frac{1}{c}$ and each time we observe an edge that is adjacent to r_1 c is increased).

In my experiments I use the counter c to estimate the *connectivity* of a triangle as follows: For each estimator that is returning a triangle into my sample I estimate the total number of adjacent nodes to this triangle as $((c-2)\frac{4}{3}) \times \frac{3}{2}$. The term $(c-2)$ is the number of r_1-adjacent edges that were observed in the stream after r_1 excluding the two edges that belong to the triangle (r_2 and r_3). The term $(c-2)\frac{4}{3}$ represents the $(c-2)$ edges that are adjacent to r_1 and arrived after r_1 in the stream *plus* the expected number of edges that arrived in the stream before r_1 (the expected number of r_1 adjacent edges that arrived before r_1 is $\frac{1}{3}(c-2)$). Thus the term $((c-2)\frac{4}{3})$ represents the number of edges that are adjacent to the r_1 edge of the triangle. The edge r_1 represents two nodes of the triangle. My estimate for the number of edges that are adjacent to the third node of the triangle is $\frac{1}{2}((c-2)\frac{4}{3})$, because I am assuming that the number of adjacent nodes are equally distributed among the three triangle nodes. The total number of adjacent edges to this triangle is then $((c-2)\frac{4}{3}) \times \frac{3}{2}$.

4 Experimental Results

4.1 Method Implementation

First I implemented the method from the paper and verified that my implementation is correct by reproducing the experiments from the paper with the same datasets which I downloaded from https://snap.stanford.edu/data. To verify my implementation I downloaded the amazon, dblp and livejournal graphs and got similar results as the authors of the paper. Further I downloaded a Protein-Protein-Interaction Network from http://thebiogrid.org/ and ran the method on this dataset. In addition to the number of triangles that was estimated by the method, I noted the number of triangles that I was able to sample from the chosen number of estimators as listed in Table 1.

dataset, true triangle count	$e = 1000$	$e = 128000$	$e = 1000000$
Amazon, $\tau = 667129$	$\tau_{est} = 710143$, $s = 1$	$\tau_{est} = 653875$, $s = 91$	$\tau_{est} = 669659$, $s = 584$
DBLP, $\tau = 2224385$	$\tau_{est} = 2392644$, $s = 3$	$\tau_{est} = 2221672$, $s = 374$	$\tau_{est} = 2220412$, $s = 2863$

Table 1

$\tau, a \Rightarrow \mu, v$	$e = 10000$	$e = 100000$	$e = 1000000$
500, 10 ⇒ 16, 115	$s = 21, \mu_{est} = 16, var_{est} = 119$	$s = 204, \mu_{est} = 14, var_{est} = 99$	$s = 2273, \mu_{est} = 16, var_{est} = 105$
500, 30 ⇒ 46, 962	$s = 2, \mu_{est} = 20, var_{est} = 360$	$s = 35, \mu_{est} = 46, var_{est} = 936$	$s = 302, \mu_{est} = 44, var_{est} = 939$
500, 50 ⇒ 76, 2633	$s = 1, \mu_{est} = 91, var_{est} = 0$	$s = 11, \mu_{est} = 57, var_{est} = 2129$	$s = 124, \mu_{est} = 79, var_{est} = 2841$
500, 100 ⇒ 152, 10576	$s = 0, \mu_{est} = \emptyset, var_{est} = \emptyset$	$s = 2, \mu_{est} = 112, var_{est} = 6210$	$s = 30, \mu_{est} = 158, var_{est} = 9290$
1000, 10 ⇒ 16, 116	$s = 18, \mu_{est} = 15, var_{est} = 72$	$s = 196, \mu_{est} = 15, var_{est} = 90$	$s = 2157, \mu_{est} = 15, var_{est} = 100$
1000, 30 ⇒ 45, 959	$s = 5, \mu_{est} = 68, var_{est} = 647$	$s = 31, \mu_{est} = 46, var_{est} = 766$	$s = 311, \mu_{est} = 43, var_{est} = 776$
1000, 50 ⇒ 76, 2757	$s = 1, \mu_{est} = 19, var_{est} = 0$	$s = 15, \mu_{est} = 78, var_{est} = 1903$	$s = 133, \mu_{est} = 74, var_{est} = 2388$
1000, 100 ⇒ 155, 11478	$s = 0, \mu_{est} = \emptyset, var_{est} = \emptyset$	$s = 2, \mu_{est} = 323, var_{est} = 573$	$s = 36, \mu_{est} = 150, var_{est} = 9374$
2000, 10 ⇒ 16, 118	$s = 27, \mu_{est} = 19, var_{est} = 75$	$s = 223, \mu_{est} = 16, var_{est} = 106$	$s = 2078, \mu_{est} = 15, var_{est} = 99$
2000, 30 ⇒ 46, 996	$s = 2, \mu_{est} = 55, var_{est} = 1437$	$s = 29, \mu_{est} = 39, var_{est} = 940$	$s = 323, \mu_{est} = 42, var_{est} = 762$
2000, 50 ⇒ 76, 2766	$s = 1, \mu_{est} = 7, var_{est} = 0$	$s = 16, \mu_{est} = 82, var_{est} = 1710$	$s = 131, \mu_{est} = 79, var_{est} = 2614$
2000, 100 ⇒ 150, 10605	$s = 0, \mu_{est} = \emptyset, var_{est} = \emptyset$	$s = 2, \mu_{est} = 69, var_{est} = 255$	$s = 28, \mu_{est} = 155, var_{est} = 7854$

Table 2: This table lists my experiment results. It can be read as follows: The first column lists the propoerties of the generated graphs. The column entry $1000, 50 \Rightarrow 76, 2757$ means I generated a graph that consists of 1000 triangles where for each triangle t I draw a number r between 1 and 50 and assigned each node of t r adjacent nodes. The mean *connectivity* of the triangles in this graph was 76 and the variance of the *connectivity* was 2757. Then I used 10^3,10^4 and 10^5 estimators to sample triangles from this graph. For 10^4 estimators for example I could draw $s = 15$ triangles into my sample. For each of these triangles I estimated the total number of adjacent nodes as explained in 3.1. Finally I computed the *connectivity* mean and variance of my sample as μ_{est} and var_{est}.

4.2 Testing of the method on simulated graphs

I tested the method for triangle sampling with the following parameters for τ (number of generated triangles), a (maximum number of adjacent nodes that can get assigned to a triangle node) and e (number of estimators from which we sample the triangles). In Table 2, the first column lists the mean *connectivity* (μ) of the generated triangles and the *connectivity*-variance of the generated triangles (v) for the chosen parameters τ and a (so you read the first column as $\tau, a \Rightarrow \mu, v$). In the table cells, s lists the sample size that I got from the e estimators and μ_{est}, var_{est} list the mean and variance of *connectivity* I got from my sample.

5 Conclusion and Discussion

0.5 page