Learning from Networks

Graph Analytics: Clustering Coefficient

Fabio Vandin

October 30^{th}, 2024

Clustering Coefficient

Let $G = (V, E)$ be a weighted/unweighted graph with $|V| = n$.

Definition

The *clustering coefficient* $cc(G)$ of G is:

$$cc(G) = \frac{|\{(u, v, z) : (u, v) \in E, (v, z) \in E, (z, u) \in E\}|}{6\binom{n}{3}}$$

Clustering Coefficient

Sometimes the *average local clustering coefficient* is used as well:

Definition

The *average local clustering coefficient* $avg_{lcc}(G)$ of G is:

$$avg_{lcc}(G) = \frac{1}{n} \sum_{v \in V} cc(v)$$

Computing the Clustering Coefficient

Given a graph $G = (V, E)$, how do we compute its clustering coefficient $cc(G)$?

Clustering Coefficient: Naïve Algorithm

Idea: for each triplet u, v, z of vertices (with $u \neq v \neq z \neq u$), check it is a triangle

Algorithm `NaïveCC`(G)
Input: graph $G = (V, E)$ with $|V| = n$ and $|E| = m$
Output: clustering coefficient of G

```
num_t ← 0;
forall u ∈ V do
    forall v ∈ V do
        forall z ∈ V do
            if u ≠ v and u ≠ z and v ≠ z then
                if (u, v) ∈ E and (u, z) ∈ E and (v, z) ∈ E then
                    num_t ← num_t + 1;
return num_t / (6 * binom(n, 3));
```

$num_t \leftarrow 0$;
forall $u \in V$ **do**
  forall $v \in V$ **do**
    forall $z \in V$ **do**
      if $u \neq v$ *and* $u \neq z$ *and* $v \neq z$ **then**
        if $(u, v) \in E$ *and* $(u, z) \in E$ *and* $(v, z) \in E$ **then**
          $num_t \leftarrow num_t + 1$;

return $\frac{num_t}{6\binom{n}{3}}$;

Complexity?

Clustering Coefficient: Better Algorithm

Idea:

- instead of considering all triplets of *nodes*, starts from *edges*
- consider all edges incident to the same vertex v one after the other

Note: this is equivalent to the exact algorithm to compute $cc(v)$ for all $v \in V$!

Clustering Coefficient: Better Algorithm (continue)

Algorithm `BetterCC`(G)
Input: graph $G = (V, E)$ with $|V| = n$ and $|E| = m$
Output: clustering coefficient of G

```
num_t ← 0;
forall u ∈ V do
    forall v ∈ N(u) do
        forall z ∈ N(u) do
            if v ≠ z and (v, z) ∈ E then
                num_t ← num_t + 1;
return num_t / (6·C(n,3));
```

$num_t \leftarrow 0$;
forall $u \in V$ **do**
  forall $v \in \mathcal{N}(u)$ **do**
    forall $z \in \mathcal{N}(u)$ **do**
      if $v \neq z$ *and* $(v, z) \in E$ **then**
        $num_t \leftarrow num_t + 1$;
return $\frac{num_t}{6\binom{n}{3}}$;

Complexity?

Complexity is too high for large networks!

Clustering Coefficient: Approximation with Sampling Algorithm

Use the one pass streaming algorithm based on reservoir sampling developed to estimate the local clustering coefficients!

Exercise

Write the pseudocode and prove that the expectation of the returned estimate is the clustering coefficient of G.

Clustering Coefficient: Better Approximation

Buriol, L. S., Frahling, G., Leonardi, S., Marchetti-Spaccamela, A., and Sohler, C. (2006). *Counting triangles in data streams*. ACM PODS.

Provides several algorithms for the *semi-streaming* model

We are going to look at one specific algorithm with the following assumption:

- all edges incident to a vertex v are stored *subsequently*

The paper presents also algorithms without such assumption.

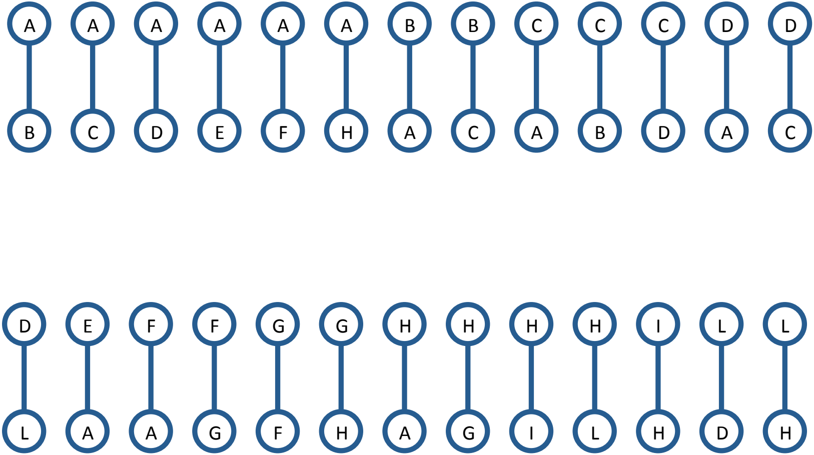
Approximating the Clustering Coefficient: Algorithm

Idea: sample a path $u - v - z$ of length 2 uniformly at random, and then check if the edge (z, u) closes a triangle

How do we sample a path $u - v - z$ (of length 2) uniformly at random?

- for a vertex v with degree $deg(v)$, the number of *distinct* paths of the type $u - v - z$ is $\frac{deg(v)}{2}(deg(v) - 1)$
- use the fact above to count the total number P of paths of length 2
- pick a value uniformly at random in $\{1, \ldots, P\}$
- scan the edges to find the vertex v in the middle of the path, and pick the "correct" path

Example


A B
A C
A D
A E
A F
A H
B A
B C
C A
C B
C D
D A
D C
D L
E A
F A
F G
G F
G H
H A
H G
H I
H L
I H
L D
L H

Approximating the Clustering Coefficient: Algorithm

Algorithm `ApproximateCC`(G, k)

Input: graph $G = (V, E)$ with $|V| = n$ and $|E| = m$; $k \in \mathbb{N}^+$

Output: approximation of clustering coefficient of G

$P \leftarrow 0$;

forall $u \in V$ **do**

  $d_u \leftarrow |\mathcal{N}(u)|$;

  $P \leftarrow P + \frac{d_u}{2}(d_u - 1)$;

forall $i \leftarrow 1$ *to* k **do**

  $(u, v, z) \leftarrow$ path of length 2 chosen uniformly at random;

  if $(u, z) \in E$ **then**

    $\beta_i \leftarrow 1$;

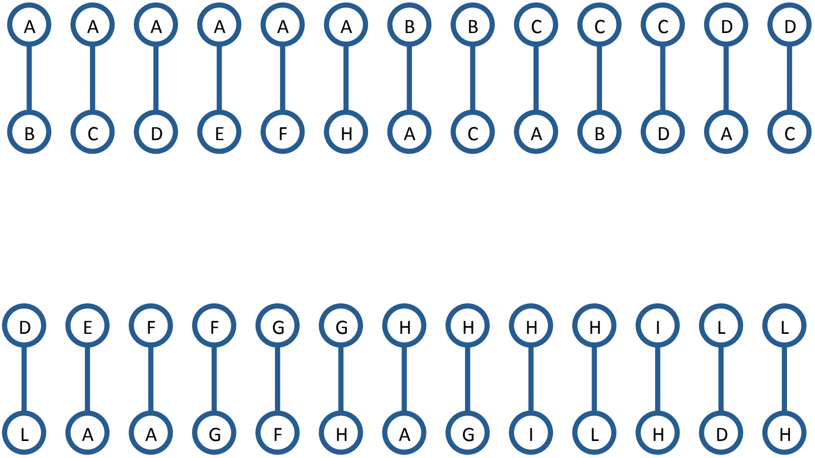
  else

    $\beta_i \leftarrow 0$;

$num_t \leftarrow \frac{1}{k}\left(\sum_{i=1}^{k} \beta_i\right)\left(\underbrace{\frac{\sum_{v \in V} d_v(d_v - 1)}{6}}_{= P/3}\right)$;

return $\frac{num_t}{\binom{n}{3}}$;

Example


A A A A A A B B C C C D D
B C D E F H A C A B D A C
D E F F G G H H H H I L L
L A A G F H A G I L H D H

Approximating the Clustering Coefficient: Analysis

Let T_i = set of subsets of 3 nodes having *exactly* i edges among them.

Proposition

For each $i = 1, \ldots, k$:

$$\mathbb{E}[\beta_i] = \frac{3|T_3|}{|T_2| + 3|T_3|}$$

Approximating the Clustering Coefficient: Analysis

Proposition

$\mathbb{E}[num_t] = |T_3|$

Approximating the Clustering Coefficient: Analysis

Proposition

Let $\varepsilon > 0, \delta \in (0,1)$ be constants. If

$$k \geq \frac{1}{\varepsilon^2} \frac{|T_2| + 3|T_3|}{|T_3|} \ln \frac{2}{\delta}$$

then

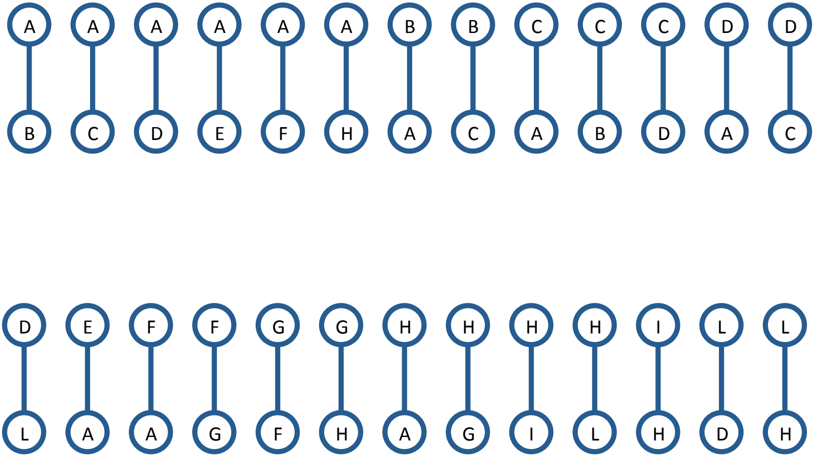
$$\mathbb{P}\left[(1-\varepsilon)\,|T_3| \leq num_t \leq (1+\varepsilon)\,|T_3|\right] \geq 1 - \delta.$$

Approximating the Clustering Coefficient: Analysis

Proposition

`ApproximateCC`(G, k) can be implemented with 3 passes on the data.

Example


A B
A C
A D
A E
A F
A H
B A
B C
C A
C B
C D
D A
D C
D L
E A
F A
F G
G F
G H
H A
H G
H I
H L
I H
L D
L H

Experimental Evaluation

Graph	r=10,000			r=100,000			r=1,000,000		
	$\tilde{T}_3$	Qlt(%)	Time	$\tilde{T}_3$	Qlt(%)	Time	$\tilde{T}_3$	Qlt(%)	Time
webgraph	7,991,057,264	-	153.78	7,541,370,749	-	393.78	7,993,479,298	-	490.56
	6,461,924,928	-	153.55	7,384,193,673	-	392.20	8,097,287,808	-	490.00
	9,977,868,646	-	153.69	8,337,706,066	-	393.92	7,591,170,489	-	491.28
actor2004	1,127,610,593	-4.16	12.29	1,155,564,261	-1.79	33.28	1,181,693,982	0.43	35.84
	1,111,095,851	-5.57	12.52	1,192,599,566	1.36	20.28	1,177,782,402	0.10	35.11
	1,177,449,181	0.07	12.12	1,175,270,762	-0.11	20.30	1,178,307,250	0.14	85.48
google-2002	43,353	-1.22	0.28	45,489	3.65	1.20	44,765	2.00	4.97
	45,293	3.20	0.28	45,435	3.52	1.00	43,704	-0.42	4.85
	37,346	-14.91	0.27	42,420	-3.34	0.99	44,208	0.73	7.55
actor2002	344,973,896	-0.53	6.70	345,817,151	-0.29	11.93	347,151,238	0.10	24.36
	351,507,109	1.35	6.59	347,683,085	0.25	12.03	345,810,766	-0.29	24.38
	330,775,554	-4.62	6.62	344,359,433	-0.71	12.00	347,532,178	0.21	55.16
authors	1,636,611	-1.73	0.43	1,665,394	-0.01	1.21	1,670,148	0.28	4.47
	1,586,971	-4.71	0.44	1,648,484	-1.02	1.19	1,665,792	0.02	4.45
	1,633,188	-1.94	0.44	1,650,487	-0.90	1.20	1,664,291	-0.07	6.86
itdk0304	458,517	0.76	0.33	449,558	-1.21	1.24	457,604	0.56	4.58
	399,317	-12.25	0.34	458,260	0.70	1.11	451,481	-0.79	4.44
	438,002	-3.75	0.34	453,440	-0.36	1.11	451,358	-0.81	6.40
wikiEN	21,099,883	7.35	2.19	20,693,869	5.29	5.34	19,938,256	1.44	16.73
	17,713,801	-9.87	2.21	20,206,714	2.81	4.78	19,894,603	1.22	16.78
	20,695,192	5.30	2.19	17,977,501	-8.53	4.78	19,414,246	-1.22	26.72
wikiDE	7,524,028	-6.87	0.91	8,265,424	2.31	3.24	8,120,882	0.52	10.54
	8,327,148	3.07	0.89	8,213,376	1.66	2.44	8,080,158	0.01	10.54
	8,114,584	0.44	0.94	8,162,754	1.04	2.45	8,024,967	-0.67	16.43
wikiFR	3,060,821	-3.23	0.34	3,255,383	2.92	1.45	3,125,790	-1.18	7.67
	3,476,882	9.92	0.34	3,199,530	1.15	1.29	3,125,613	-1.18	7.61
	3,447,016	8.98	0.34	3,206,780	1.38	1.28	3,138,100	-0.79	10.63
wikiES	863,765	8.45	0.18	782,798	-1.72	0.94	793,282	-0.40	5.09
	791,437	-0.63	0.18	774,447	-2.76	0.90	800,619	0.52	5.09
	768,999	-3.45	0.18	827,132	3.85	0.87	803,774	0.92	6.85
wikiIT	339,404	3.39	0.12	313,241	-4.58	0.75	337,843	2.92	4.16
	318,664	-2.92	0.12	308,480	-6.03	0.74	330,290	0.62	4.11
	305,763	-6.85	0.12	339,498	3.42	0.73	322,894	-1.64	5.53
wikiPT	70,699	0.94	0.07	70,443	0.57	0.53	70,942	1.28	2.63
	62,620	-10.60	0.07	71,136	1.56	0.53	72,329	3.26	2.58
	80,752	15.29	0.07	69,568	-0.68	0.53	69,203	-1.20	3.32