Clustering Coefficient in Graph Theory

Difficulty Level : Basic • Last Updated : 08 Feb, 2018

In graph theory, a clustering coefficient is a measure of the degree to which nodes in a graph tend to cluster together. Evidence suggests that in most real-world networks, and in particular social networks, nodes tend to create tightly knit groups characterized by a relatively high density of ties; this likelihood tends to be greater than the average probability of a tie randomly established between two nodes (Holland and Leinhardt, 1971; Watts and Strogatz, 1998).

Two versions of this measure exist: the global and the local. The global version was designed to give an overall indication of the clustering in the network, whereas the local gives an indication of the embeddedness of single nodes.

Global clustering coefficient

The global clustering coefficient is based on triplets of nodes. A triplet consists of three connected nodes. A triangle therefore includes three closed triplets, one centered on each of the nodes (n.b. this means the three triplets in a triangle come from overlapping selections of nodes). The global clustering coefficient is the number of closed triplets (or 3 x triangles) over the total number of triplets (both open and closed). The first attempt to measure it was made by Luce and Perry (1949). This measure gives an indication of the clustering in the whole network (global), and can be applied to both undirected and directed networks.

Local clustering coefficient

A graph $G = (V, E)$ formally consists of a set of vertices V and a set of edges E between them. An edge e_{ij} connects vertex v_i with vertex v_j.

The neighborhood N_i for a vertex v_i is defined as its immediately connected neighbors as follows:

$$N_i = \{v_j : e_{ij} \in E e_{ji} \in E\}.$$

We define k_i as the number of vertices, $|N_i|$, in the neighbourhood, N_i, of a vertex.

The local clustering coefficient C_i for a vertex v_i is then given by the proportion of links between the vertices within its neighborhood divided by the number of links that could possibly exist between them. For a directed graph, e_{ij} is distinct from e_{ji}, and therefore for each neighborhood N_i there are $k_i(k_i - 1)$ links that could exist among the vertices within the neighborhood (k_i is the number of neighbors of a vertex). Thus, the local clustering coefficient for directed graphs is given as [2]

$$C_i = \frac{|\{e_{jk} : v_j, v_k \in N_i, e_{jk} \in E\}|}{k_i(k_i - 1)}.$$

An undirected graph has the property that e_{ij} and e_{ji} are considered identical. Therefore, if a vertex v_i has k_i neighbors, $\frac{k_i(k_i-1)}{2}$ edges could exist among the vertices within the neighborhood. Thus, the local clustering coefficient for undirected graphs can be defined as

$$C_i = \frac{2|\{e_{jk} : v_j, v_k \in N_i, e_{jk} \in E\}|}{k_i(k_i - 1)}.$$

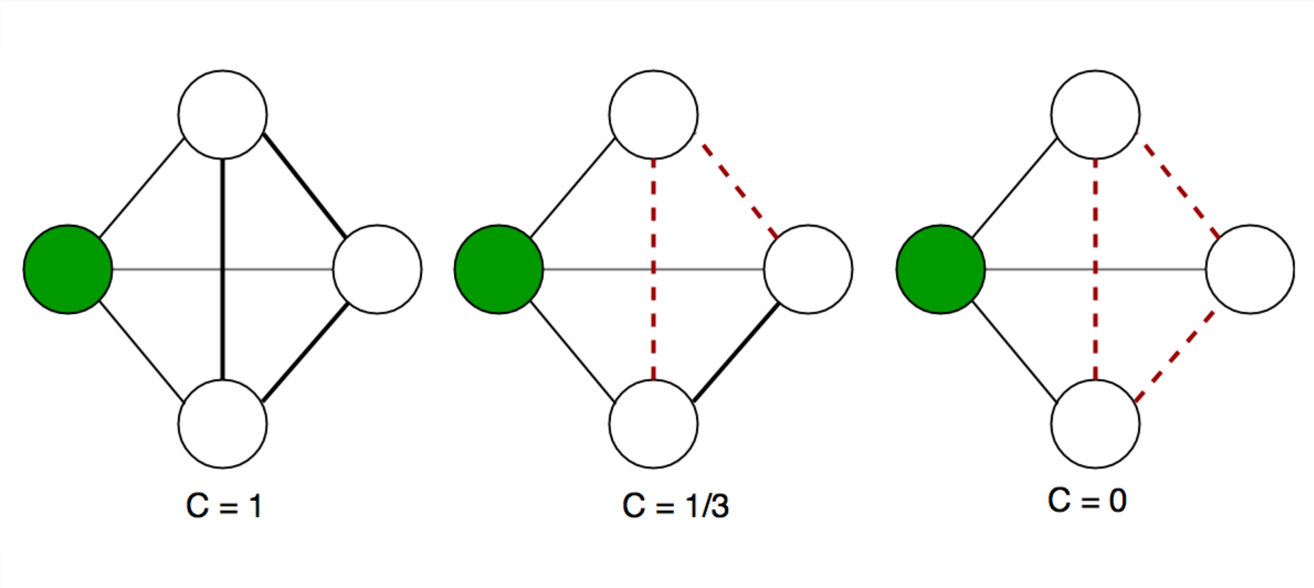
Let $\lambda_G(v)$ be the number of triangles on $v \in V(G)$ for undirected graph G. That is, $\lambda_G(v)$ is the number of sub-graphs of G with 3 edges and 3 vertices, one of which is v. Let $\tau_G(v)$ be the number of triples on $v \in G$. That is, $\tau_G(v)$ is the number of sub-graphs (not

$C_i = \frac{\lambda_G(v)}{\tau_G(v)}$.

It is simple to show that the two preceding definitions are the same, since

$\tau_G(v) = C(k_i, 2) = \frac{1}{2}k_i(k_i - 1)$.

These measures are 1 if every neighbor connected to v_i is also connected to every other vertex within the neighborhood, and 0 if no vertex that is connected to v_i



Example local clustering coefficient on an undirected graph. The local clustering coefficient of the green node is computed as the proportion of connections among its neighbours.

Here is the code to implement the above clustering coefficient in a graph. It is a part of

```
    The local clustering of each node in `G` is the
    fraction of triangles that actually exist over
    all possible triangles in its neighborhood.
    The average clustering coefficient of a graph
    `G` is the mean of local clusterings.

    This function finds an approximate average
    clustering coefficient for G by repeating `n`
    times (defined in `trials`) the following
    experiment: choose a node at random, choose
    two of its neighbors at random, and check if
    they are connected. The approximate coefficient
    is the fraction of triangles found over the
    number of trials [1]_.

    Parameters
    ----------
    G : NetworkX graph

    trials : integer
        Number of trials to perform (default 1000).

    Returns
    -------
    c : float
        Approximated average clustering coefficient.



    """
    n = len(G)
    triangles = 0
    nodes = G.nodes()
    for i in [int(random.random() * n) for i in range(trials)]:
        nbrs = list(G[nodes[i]])
        if len(nbrs) < 2:
            continue
        u, v = random.sample(nbrs, 2)
        if u in G[v]:
            triangles += 1
    return triangles / float(trials)
```

Note: The above code is valid for undirected networks and not for the directed networks.
The code below has been run on IDLE(Python IDE of windows). You would need to download the networkx library before you run this code. The part inside the curly braces represent the output. It is almost similar as Ipython(for Ububtu users).

```
>>> cc
#Output of Global CC
0.08333333333333333
>>> c=nx.clustering(G)
>>> c
# Output of local CC
{0: 0.0, 1: 0.3333333333333333, 2: 0.0, 3: 0.0, 4: 0.0, 5: 0.0, 6: 0.0,
 7: 0.3333333333333333, 8: 0.0, 9: 0.16666666666666666}
```

The above two values give us the global clustering coefficient of a network as well as local clustering coefficient of a network.

Next into this series, we will talk about another centrality measure for any given network.

References

You can read more about the same at

- https://en.wikipedia.org/wiki/Clustering_coefficient
- http://networkx.readthedocs.io/en/networkx-1.10/index.html

.

This article is contributed by Jayant Bisht. If you like GeeksforGeeks and would like to contribute, you can also write an article using contribute.geeksforgeeks.org or mail your article to contribute@geeksforgeeks.org. See your article appearing on the GeeksforGeeks main page and help other Geeks.

Please write comments if you find anything incorrect, or you want to share more information about the topic discussed above.

Attention reader! Don't stop learning now. Get hold of all the important mathematical concepts for competitive programming with the **Essential Maths for CP Course** at a student-friendly price. To complete your preparation from learning a language to DS Algo and many more, please refer **Complete Interview Preparation Course**.

Like 0

Next

Katz Centrality (Centrality Measure)

RECOMMENDED ARTICLES

Page : 1 2 3

01 **Mathematics | Graph theory practice questions**
29, Oct 17

02 **Proof that Independent Set in Graph theory is NP Complete**
26, Jun 20

03 **Maximum binomial coefficient term value**
18, Dec 17

04 **Fibonomial coefficient and Fibonomial triangle**
25, Dec 17

05 **Sum of product of r and rth Binomial Coefficient (r * nCr)**
03, Jun 18

06 **Central binomial coefficient**
26, Jun 20

07 **Eggs dropping puzzle (Binomial Coefficient and Binary Search Solution)**
03, Sep 18

08 **Program to find correlation coefficient**
31, Oct 17

Vote for difficulty

Current difficulty : Basic

Easy Normal Medium Hard Expert

Article Tags : Graph, Mathematical

Practice Tags : Mathematical, Graph

Improve Article Report Issue

Writing code in comment? Please use ide.geeksforgeeks.org, generate link and share the link here.

Load Comments

5th Floor, A-118,
Sector-136, Noida, Uttar Pradesh - 201305

feedback@geeksforgeeks.org

Web Development

Web Tutorials

HTML

CSS

JavaScript

Bootstrap

Contribute

Write an Article

Write Interview Experience

Internships

Videos