

RESEARCH PAPER

A Triad Percolation Method for Detecting Communities in Social Networks

Zhiwei Zhang¹, Lin Cui¹, Zhenggao Pan¹, Aidong Fang¹ and Haiyang Zhang²¹ School of Informatics and Engineering, Suzhou University, Anhui, CN² School of Environment Science and Spatial Informatics, Suzhou University, Anhui, CNCorresponding author: Lin Cui (jsjxcuilin@126.com)

For the purpose of detecting communities in social networks, a triad percolation method is proposed, which first locates all close-triads and open-triads from a social network, then a specified close-triad or open-triad is selected as the seed to expand by utilizing the triad percolation method, such that a community is found when this expanding process meet a particular threshold. This approach can efficiently detect communities not only from a densely social network, but also from the sparsely one. Experimental results performing on real-world social benchmark networks and artificially simulated networks give a satisfactory correspondence.

Keywords: Community detection; Network analysis; Complex network; Social sciences computing; Triad percolation; Graph theory

1. Introduction

Many real-world systems in nature and society can be described as complex networks or graphs (Cui, Wang & Li 2014; Mu et al. 2014; Shen et al. 2009; Zhang & Wang 2015), such as the Internet, World Wide Web, social networks, biological networks, in which the nodes denote the entities and the interactions between entities can be described as edges. Community structure or clustering structure is one of the most common features of complex networks, where a group of nodes exhibit dense connections, meanwhile exhibiting comparatively sparse connections with the rest of the network (Ahn et al. 2009; Mu et al. 2014; Shen et al. 2009; Zhang & Wang 2015). Understanding community structure of social networks is critical due to its broad applications such as friend recommendations, user modeling and content personalization. However, the quantitative definition of community has been widely discussed by scholars from different areas. Until now, there is still no well-accepted quantitative definition (Liu et al. 2013). And there is no unique and widely accepted goodness measure of community quality in literature. In most cases, the community structure is often dependent on the specific application scenario. For instance, communities in a social network represent the groups of people with similar interests and talking topics; communities in biological networks stand for the modules of biological tissues with similar functions; communities in a protein network elaborate the a set of proteins with similar interaction function; communities in a web network are considered to be clusters of web documents with related topics (Liu et al. 2013; Zhao et al. 2018). Informally, a good community is a densely-connected group of nodes that is sparsely connected to the rest of the network. Extensive research has been devoted to designing community detection algorithms to uncover communities with the goal to minimize or maximize structural metrics, such as modularity, triangle participation ratio, or conductance of the discovered communities (Wagenseller & Wang 2018).

Over the past decade community detection has been applied to many real-world areas such as biological networks, protein networks, web graphs, VLSI design, social networks, and task scheduling. According to the motivations, application scenarios, the criterion of whether to allow overlapping and technical principles that existing methods adopted, the representative inspirations could be divided into two big classes: overlapping community detection and disjoint community detection (Leskovec, Lang & Mahoney 2010; Xie, Kelley & Szymanski 2013; Xu et al. 2016). Disjoint community detection focuses on the division of the boundaries between communities, and the sole ownership of each node. Overlapping community detection focuses on the distribution of the membership of each node, and the entire whole network structure.

Disjoint community detection was reviewed and categorized into five research lines, researchers and scholars used numerous techniques such as spectral clustering, modularity maximization, random walks and statistical mechanics to discover a community as a set of nodes that has more links between its members than with the remainder of the network (Leskovec, Lang & Mahoney 2010). Overlapping community detection are reviewed and categorized into five categories: Clique Percolation, Line Graph and Link Partitioning, Local Expansion and Optimization, Fuzzy Detection, Agent-Based and Dynamical Algorithms (Xie, Kelley & Szymanski 2013), which are investigated based on the consensus that people in a social network are naturally characterized by multiple community memberships (Xu et al. 2016).

However, the real-world networks, especially online social networks such as Sina, Twitter, and Facebook, are highly sparse, and communities in those networks are often small, comparing dozens or even hundreds of members (Han & Tang 2015; Leskovec et al. 2008). Intuitively, the latent members usually centered on a hub-node, which often are opinion leaders or activities, but the links between members are relatively seldom, and hub-nodes often form three tuples with other members, which all lead to sparse network topological structure and community distribution. Therefore, the traditional clique-based community detection approaches cannot work efficiently.

It is important to note that different community detection algorithms often tend to perform significantly well or poorly on certain kinds of networks. Moreover, one might prefer to choose specific algorithm with respect to the target application or the characteristics of a network that to be analyzed. Thus, it is necessary to analyze the real situations and application scenarios (Zhao et al. 2018). Thereby, without considering the individual nodes and edges, each node inhabits a triad (detailed in section 3.1). Inspired by previous related studies, an efficient and functional algorithm TPM (Triad Percolation Method), a method like CPM (Palla et al. 2005), is proposed. This method is more universal than the approach presented in Ref. (Fagnan, Zaïane & Barbosa 2014), which only considers the close-triad (3-clique).

The remainder of this paper is organized as follows. Section 2 reviews related work. Section 3 introduces motivations and outlines the algorithm TPM. In section 4 purpose experimental results are reported and the final section offers concluding remarks and sheds light on future research directions.

2. Related work

In the last decade, community detection can be claimed as a remarkable achievement in theory and in practical, especially in the field of social sciences computing. Many community detection algorithms have been developed. We organize a brief discussion on related works into two parts: the first is devoted to common node-based and link based community detection algorithms, and the second concerns the methods related to the idea of this article.

The node-based and link-based community detection methods are more widely used in detecting communities from networks. The node-based methods (Conrad 2010; Cui, Wang & Li 2014; Fagnan, Zaïane & Barbosa 2014; Lancichinetti, Fortunato & Kertész 2009; Lusseau et al. 2003; McAuley & Leskovec 2012; Mu et al. 2014; Newman 2006; Palla et al. 2005; Shen et al. 2009; Wang 2011; Zhang & Wang 2015; Zachary 1977) directly split the node set of a network into different partitions. Many excellent approaches for that purpose, such as CPM (Palla et al. 2005), GCE (Conrad et al. 2010), SLPA (Xie, Szymanski & Liu 2011), are proposed. The local optimization strategy is adopted by many of those algorithms. Besides, such algorithms like EAGLE (Shen et al. 2009), LFM (Lancichinetti, Fortunato & Kertész 2009), not only can discover overlapping communities, but also can identify the hierarchical organizations from a social network. As for the main idea of link-based techniques is taking advantage of the link information rather than node information to detect communities as following described (Ahn, Bagrow & Lehmann 2010; Shi et al. 2013; Ye et al. 2011). First, the link-based algorithms cluster on links of an original network. Then, link communities are mapped into node communities by gathering nodes that incident to all links within each link community (Ahn, Bagrow & Lehmann 2010; Shi et al. 2013; Ye et al. 2011), which just like the node-based community detection progress mentioned above.

In past years, a lot of approaches, employing different types of heuristics and a wide variety of criteria to optimize, have been proposed. Detailed surveys describing these methods can be found in (Aggarwal & Subbian 2014; Amelio & Pizzuti 2014; Fortunato 2010; Fortunato & Castellano 2012; Peel, Larremore & Clauset 2017; Pizzuti 2018). Reichardt and Bornholdt (2006) developed an approach based on finding the minimum-energy state of an infinite-range Potts spin glass, authors have proved an equivalence between minimizing the Hamiltonian and finding the maximum modularity partition of a network, and next to Reichardt and Bornholdt (2006), Delvenne et al. (2010) introduced a measure called clustering stability which generalizes modularity, normalized cut objective and Fiedler's spectral clustering approach for certain values of input parameters (Delvenne, Yaliraki & Barahona 2010; Veldt, Gleich & Wirth 2018).

Significantly, Palla et al. (2005) proposed a Clique Percolation Method that identifies communities by rolling a k -clique around the network until it is unable to reach any unexplored nodes (Fagnan, Zaïane & Barbosa 2014; Palla et al. 2005). As all we know, the k -clique rarely occurs in online social networks, such as Sina, Facebook and Twitter, hence the CPM does not effectively identify communities from sparsely online social networks. Correspondingly, Fagnan et al. (2014) attempted to remedy the problem that traditional methods cannot be extended to large scale datasets by using triads and the T metric they proposed, and the method they proposed contains two steps: 1) First, initial communities are detected by applying T metric within a local community detection framework; 2) Then, an additional measurement is employed to identify outliers/hubs in the discovered initial communities. However, this triad-based method presented in Ref. (Fagnan, Zaïane & Barbosa 2014) aim at the situation that communities are tightly-knit groups of nodes that interact more frequently within the group than outside of the group, and performs badly in the sparse online social networks. Irrespective of single nodes or single edges of a network, each node must be contained in a triad, close-triad or open-triad. Thereby, we put forward the Triad Percolation Method by combining the above problems for community detection in large scale sparse online social networks, which combines the motivations of CPM (Palla et al. 2005) and the idea presented in Ref. (Fagnan, Zaïane & Barbosa 2014).

3. Triad Percolation Method

In this section, several definitions used in this paper are explained firstly. Then, we demonstrate some motivations for the TPM, and outline this algorithm. Finally, corresponding analysis and parameter tuning strategy are given to TPM.

3.1. Preliminaries of TPM

Definition 1, Triad: A 4-tuple $[v_1, v_2, v_3, flag]$ is used to represent a triad, where, v_1, v_2 and v_3 are the nodes that build a triad, while the flag indicates that the triad is close or open. $flag = -1$ illustrates that triad is a close-triad. Otherwise, the $flag$ is set to the centroid of an open-triad. As shown in **Figure 1**, $[2, 4, 5, -1]$, $[3, 4, 5, -1]$ and $[6, 8, 9, -1]$ are close-triads; however, open-triads include $[2, 4, 13, 4]$, $[3, 4, 13, 4]$ and $[6, 8, 13, 8]$. Noticing that a pair of triads are adjacent to each other if and only if they have a common edge, e.g. $[2, 4, 5, -1]$ and $[2, 4, 13, 4]$.

Definition 2, External Degree: A metric for evaluating a matched open-triad, when an open-triad matched with (i.e. adjacent to) a triad in a specific community, thereby the remaining unmatched node of an open-triad is usually at the border of community, thus we take advantage of the remaining node's degree to denote the external degree of that open-triad. For instance, $[2, 4, 13, 4]$ is adjacent to the $[2, 4, 5, -1]$, as well as $[6, 8, 13, 8]$ and $[6, 8, 9, -1]$. However, their external degree is 2, which is the degree of remaining unmatched node 13, i.e. $extDegree([2, 4, 13, 4]) = 2$, which is usually a bridge node.

3.2. Motivations of TPM

As we all know that the real-world social networks have a sparse feature and exponential-law degree distribution. Hence, the k -cliques (Palla et al. 2005) or maximal cliques (Bron & Kerbosch 1973) are not extensively resides in the real situations. Irrespective of the single nodes or single edges of a network, each node

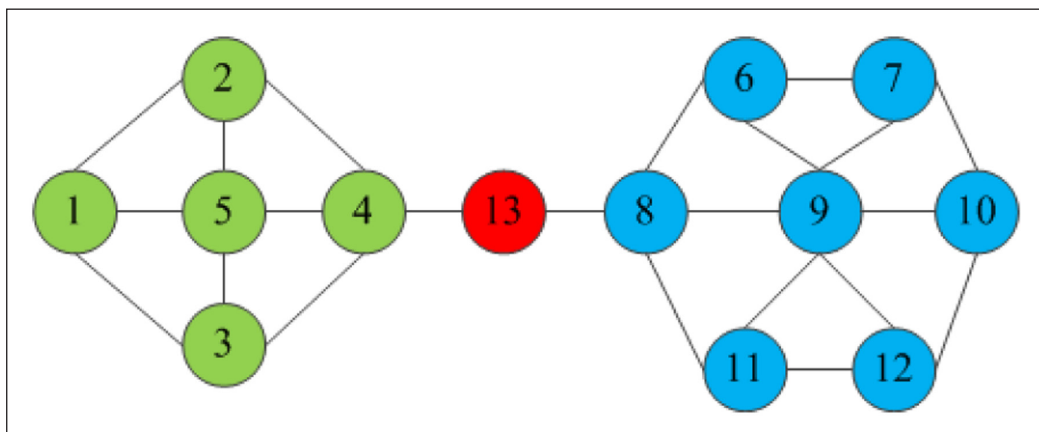


Figure 1: Example of a social network. (For interpretation of the references to color in this and following figure legend, the reader is referred to the web version of this article.).

must be contained in a triad, close-triad or open-triad. The extensive researches suggest that close-triads are often located in the center of a community, and open-triads are usually situated on the border of that community (Palla et al. 2005).

For the purpose of detecting communities from social networks, first find out all of the triads, then an initial community is formed by expanding a seed close-triad or an open-triad by adopting the strategy that similar to the CPM (Palla et al. 2005). Eventually, we also iteratively extend an initial community to a newer, larger community. A community is detected until there is no any initial community meets the specific requirement detailed in formula (4).

Under the premise that two communities share common nodes and one community contains relatively less nodes, while another community possesses more nodes, thereby these two communities will possess strong coupling strength due to the small community is usually sub-community of the bigger community (Wang 2011; Zhang & Wang 2015). Accordingly, the two communities will merge into a newer, larger community. Formula (1) and (2) therefore are adopted (Wang 2011).

$$BC_1 = \frac{|N(C_i) \cap N(C_j)|}{\min\{|N(C_i)|, |N(C_j)|\}} \quad (1)$$

$$BC_2 = \frac{|N(C_i) \cup N(C_j)|}{\max\{|N(C_i)|, |N(C_j)|\}} \quad (2)$$

Where $N(C_i)$ and $|N(C_i)|$ denote the node set and the amount of nodes for community C_i , respectively.

The more links that connect all nodes of two communities with bigger community clustering coefficient (Cui, Wang & Li 2014), the higher probability they belong to the same community, thus they will be incorporated into a larger community, which makes the formula (3) (Cui, Wang & Li 2014) comes into being.

$$BC_3 = \frac{2 * E_{ij}}{K_{ij}(K_{ij} - 1)} \quad (3)$$

Where K_{ij} indicates the total amounts of nodes for community C_i and C_j , number of links that connect K_{ij} nodes is denoted by E_{ij} , and these K_{ij} nodes will have $K_{ij}(K_{ij} - 1)/2$ links as a complete graph while they have E_{ij} links in fact.

During the emerging process, we need to select a pair of similar communities which belonging coefficient (detailed in formula (4)) meets the requirement of greater than a specific threshold α , where the belonging coefficient for evaluating the degree that C_j belongs to C_i . The TPM is a heuristic algorithm, which lacking of global community partition, so we calculate the Harmonic mean value instead of calculating the arithmetic mean value to measure whether a community C_j belongs to another C_i .

$$BC(C_i, C_j) = \begin{cases} 0, & BC_1 = 0 \text{ or } BC_2 = 0 \text{ or } BC_3 = 0 \\ \frac{3.0}{\sum_{x=1}^3 \frac{1.0}{BC_x}}, & BC_1 \neq 0 \text{ or } BC_2 \neq 0 \text{ or } BC_3 \neq 0 \end{cases} \quad (4)$$

3.3. The algorithm of TPM

Inspired by the empirical analysis and observations mentioned above, the algorithm TPM is summarized as follows **Table 1**.

3.3.1. TPM time-complexity analysis

The algorithm time-complexity is now analyzed, assuming n as the total number of nodes of a social network G , m and s as the amount of close-triads and the number of open-triads, respectively. In steps 1 and 2, we simply iterating the each node in social network G , the $O(n^2)$ operations are needed. As for step 5 to 28, the main manipulations are query and match, so when we select a triad as a seed community to expand, there needs $O(m^2)$ query and match operations for a close-triad expanding, and $O(s^2)$ corresponding manipulations for an open-triad seed expanding. Noticing the fact that many real-world social networks have the sparse

Table 1: The algorithm of Triad Percolation Method (TPM).**Algorithm 1:** TPM(G, α)**Input:** G : a social network; α : belonging coefficient threshold**Output:** a community list $comm_lst$

1. $close_triad$ = locates all close-triads from social network G
2. $open_triad$ = locates all open-triads from social network G
3. $adj_lst = \emptyset$: a list of adjacent triads that expand from a specific seed triad
4. $comm_lst = \emptyset$: a list of adj_lst , and eventually form a community list
5. **while** $close_triad \neq \emptyset$: //start to generate initial communities by expanding seeds
6. $seed_close_triad$ = select a close-triad with the largest sum of degree of three nodes from $close_triad$
7. $adj_lst.append(seed_close_triad)$
8. $close_triad.remove(seed_close_triad)$
9. **for** itm_clo **in** $close_triad$:
10. **if** itm_clo is adjacent to a triad in $comm_lst$:
11. $adj_lst.append(itm_clo)$
12. $close_triad.remove(itm_clo)$
13. **for** itm_opn **in** $open_triad$:
14. **if** itm_opn is adjacent to a triad in adj_lst **and** $extDegree(itm_opn) \leq 2$:
15. $adj_lst.append(itm_opn)$
16. $open_triad.remove(itm_opn)$
17. $comm_lst.append(adj_lst)$
18. $adj_lst = \emptyset$
19. **while** $open_triad \neq \emptyset$:
20. $seed_open_triad$ = select an open-triad from $open_triad$
21. $adj_lst.append(seed_open_triad)$
22. $open_triad.remove(seed_open_triad)$
23. **for** itm_opn **in** $open_triad$:
24. **if** itm_opn is adjacent to a triad in adj_lst **and** $extDegree(itm_opn) \leq 2$:
25. $adj_lst.append(itm_opn)$
26. $open_triad.remove(itm_opn)$
27. $comm_lst.append(adj_lst)$
28. $adj_lst = \emptyset$
29. **while** $open_triad \neq \emptyset$:
30. $seed_open_triad$ = select an open-triad from $open_triad$
31. $adj_lst.append(seed_open_triad)$
32. $open_triad.remove(seed_open_triad)$
33. **for** itm_opn **in** $open_triad$:
34. **if** itm_opn is adjacent to a triad in adj_lst **and** $extDegree(itm_opn) \leq 2$:
35. $adj_lst.append(itm_opn)$
36. $open_triad.remove(itm_opn)$
37. $comm_lst.append(adj_lst)$
38. $adj_lst = \emptyset$
39. **for** itm_opn **in** $open_triad$:
40. $comm_lst.append(itm_opn)$
41. Convert the items in $comm_lst$ into the corresponding node set // initial communities generation end here
42. A pair of initial communities with, and are selected from $comm_lst$, merged into a newer, larger community. Repeat this process until there is no any pair of communities meet the requirement of. $BC(C_i, C_j) > \alpha$.
43. Output the result communities in $comm_lst$

characteristic, hence the variables m , n and s meet the condition of $m \leq s \leq n$ that this process requires $O(n^2)$ at most. Come to step 31, this conversion obtains $O(n)$ operations. The merge manipulation of step 32 is an iterative process, which demands $O(n^2)$ operations at most. In summary, the overall time-complexity of algorithm TPM is $O(n^2)$, a time-consuming approach, and its performance will be improved in our future work.

3.3.2. The estimation of belonging coefficient threshold α

In order to making TPM adapt to different social network settings, we also proposed an approach to estimate the belonging coefficient threshold. Suppose $LSP = \{v_1, v_2, \dots, v_k\}$ is the largest shortest path of social network G , we get the average clustering coefficient $LACC$ of nodes within the LSP , i.e. $LACC = \frac{1}{k} \sum_{i=1}^k cc(v_i)$, where $CC(v_i)$ is the clustering coefficient of the node v_i ($v_i \in LSP$). And we also assume that ACC illustrates the average clustering coefficient of social network G . Finally, we use the harmonic mean value $hmcc$ of $LACC$ and ACC as the estimation of belonging coefficient threshold α , i.e. $hmcc = (2 * LACC * ACC) / (LACC + ACC)$. The reason why we use the strategy mentioned above is motivated by the idea of formula (1) to (4), which are based on local aggregation and expansion. Although this method may not provide a pretty accurate threshold, it can furnish an approximate reference value automatically.

Above all, the calculated reference threshold serves as a benchmark for fine-tuning, we can also run the application of TPM more than once, and we set the tuned threshold that maximizes the following modularity Q as the final threshold α (Newman 2006).

$$\begin{cases} Q = \frac{1}{2m} \sum_{l=1}^k \sum_{i \in C_l, j \in C_l} A_{ij} - d_i d_j / 2m \\ s.t. \max(Q) \end{cases} \quad (5)$$

4. Experiments and analysis

In this section, the performance of TPM is evaluated on four real-world social networks and an artificially simulated network, Zachary Karate Club benchmark network (Zachary 1977), Bottlenose Dolphins network (Lusseau et al. 2003), Facebook ego network (McAuley & Leskovec 2012), the Cora¹ citation network, and the LFR (Lancichinetti, Fortunato & Radicchi 2008) benchmark network, their statistical properties and specific settings are listed as following **Table 2**. Although the real social networks Karate and Dolphins are very small, they can serve as the benchmark networks due to their community structure was known beforehand, and so as to be able to evaluate the accuracy of the tested algorithm.

4.1. Zachary's Karate Club network

The Zachary's Karate Club network (Zachary 1977), the first real-world social network tested was widely used as a benchmark for evaluating the performance of community detection algorithms. This network is an un-weighted network with 34 members of a karate club as nodes and 78 links representing friendships among members of the club. However, due to the conflict between the club administrator and the club main instructor, the members were split into two different groups centering on the administrator and the instructor, respectively. As shown in **Figure 2**, the community organization is plotted, where node 1 and 33 represent the administrator and instructor, respectively.

Table 2: The properties and settings of each social network.

Network	# nodes	# edges	# comm.	α
Karate Club [21]	34	78	2	0.35
Dolphins [22]	62	159	2	0.25
Facebook [23]	333	2519	8	0.16
Cora ¹	2708	5249	293	0.29
LFR ²	500,000	— ³	500	0.32

¹ Download URL: <https://linqs.soe.ucsc.edu/data>.

² Download URL: <http://santo.fortunato.googlepages.com/benchmark.tgz>.

³ The amount of edges corresponding to different mixing parameter mu ($mu \in [0.1, 0.9]$) is 15367853, 14359862, 15116857, 11368975, 12584676, 15632564, 11354862, 12758439 and 14865724, respectively. Under the premise of the same parameters settings, however, it is important to note that the number of network edges obtained by running the LFR software multiple times is significantly different.

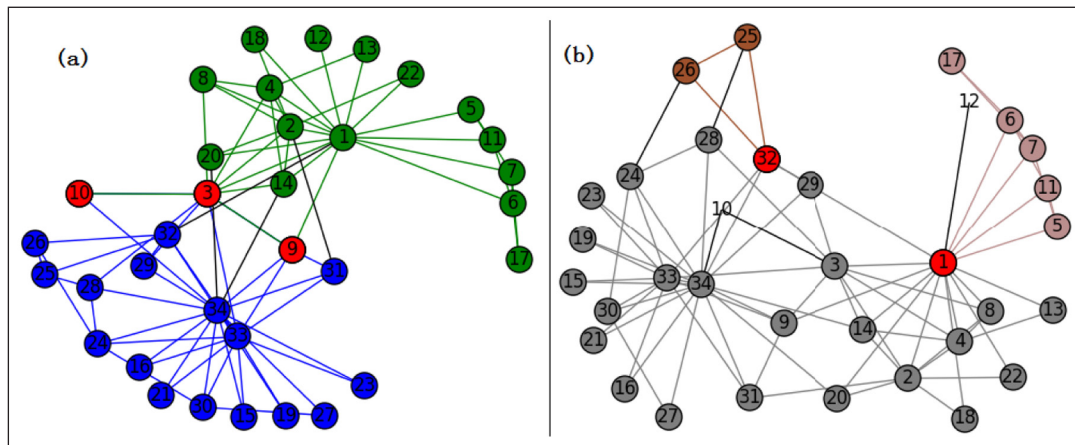


Figure 2: The Zachary's Karate club network and its community organization. Different community rendered in different color, the nodes and links that belong to same community rendered in same color, and the red nodes are overlaps. **(a)** Community structure detected by TPM. **(b)** The corresponding community organization discovered by CPM.

Here, we set the belonging coefficient threshold as $\alpha = 0.35$ for TPM. Then, this network was naturally partitioned into 2 communities, which are in accord with the actual situation, as shown in **Figure 2(a)**. Three overlaps, node 3, 9 and 10, were extracted. For node 10, it has only two edges, which connect two different communities, and the probability that they belong to each community are equal, hence node 10 as an overlap is reasonable. As for node 9, it forms three close-triads, which are $\{9, 3, 1\}$, $\{9, 33, 34\}$ and $\{9, 31, 33\}$. Then, $\{9, 33, 34\}$ and $\{9, 31, 33\}$ belong to same community, but $\{9, 3, 1\}$ belongs to another. However, so strong strength of triangle stability that makes node 9 constructs a stable structure. Therefore, node 9 plays a role of overlap is reasonable, and node 3 obtained the similar situation. The performance is better than CPM (Palla et al. 2005) and GCE (Conrad et al. 2010).

However, the communities discovered by CPM ($k = 3$), just as shown in **Figure 2(b)**, are not meet the real situation that two communities centered by node 1 and 33 respectively. Besides, the node 10 and 12, which also can be called as 'lost guys', are not contained into the detected communities. Noticing that the number of 'lost guys' will be more as the increasing of clique size k for CPM (Palla et al. 2005), especially for large, sparse social networks. But, the proposed algorithm TPM will be better able to handle those cases.

4.2. The Bottlenose Dolphins network

The second real-world benchmark network examined was the Bottlenose Dolphin Network (Lusseau et al. 2003), the 62 nodes are the bottlenose dolphins (genus *tursiops*) of a bottlenose dolphin community living off Doubtful Sound, a fjord in New Zealand (spelled fiord in New Zealand). An edge indicates a frequent association. The dolphins were observed between 1994 and 2001. This network is an un-weighted and undirected network with 62 nodes and 159 links.

For this case, we assigned the belonging coefficient threshold $\alpha = 0.25$. As shown in **Figure 3(a)**, the TPM experimental correspondence is almost perfect, the community partitions and overlapping nodes are reasonable and accord with the real situation. However, the communities detected by CPM ($k = 3$), as shown in **Figure 3(b)**, is just passable, which not only inconsistent with the actual situation, but also produced many 'lost guys', such as node 4, 11, 12, and 58. And that situation will be more serious as the increasing of clique size k for CPM. But the proposed TPM will works well for such sparse networks.

4.3. The Facebook network

This section introduces a dataset of ego-network from Facebook. This dataset adopt from Ref. (McAuley & Leskovec 2012), users in Facebook identifying friends sharing a common attribute. Generally, there are two useful sources of data that help with locating those common attribute groups. We expect that communities are formed by densely-connected sets of users. In other words, the goal is to find nested as well as overlapping communities in this network (McAuley & Leskovec 2012).

As for this Facebook network setting, the belonging coefficient threshold was set to $\alpha = 0.16$. During the experimental process, we also fine-tuned this parameter by step length of 0.05, and we acquired different community divisions, which demonstrated the clustering situation of network in different granularity.

Through the empirical analysis, a relative good community partition for this Facebook network is cleared in **Figure 4(a)**, which brought the network's distinctive community structure to light.

Meanwhile, except for the 'lost guys', the CPM ($k = 3$) also got a good network partition. However, the CPM is hard to determine the clique size, and that also lost many nodes inevitably, especially for the large, sparse real-world social networks.

4.4. The Cora citation network

The Cora¹ dataset consists of 2708 scientific publications (nodes) classified into one of seven classes and 5429 links. The Cora dataset consists of Machine Learning papers. These papers are classified into one of the following seven classes: case_based, genetic_algorithms, neural_networks, probabilistic_methods, reinforcement_learning, rule_learning and theory. The papers were selected in a way such that in the final corpus every paper cites or is cited by at least one other paper.

However, due to the large number of nodes and edges in Cora¹ network and the limited paper space, the specific community divisions are not plotted as shown in **Figures 2, 3** and **4**. Thus, we only described the statistics of the communities detected by the TPM and CPM, respectively. The CPM as a comparative method, the size k of clique is set to 3, 4 and 5, respectively. (1) When the value of k is 3, the CPM obtained 293 communities with 1469 nodes, and the remained 1239 nodes are not divided into corresponding

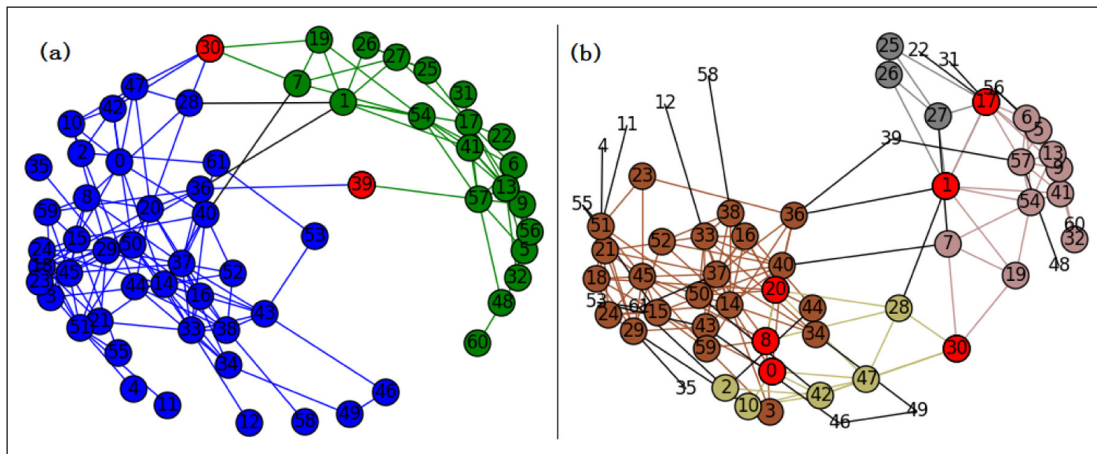


Figure 3: The bottlenose dolphins benchmark network and its community structure. Different communities rendered in different color, the nodes and links belong to same community rendered in same color. The red nodes are overlaps. **(a)** Community structure detected by TPM. **(b)** The corresponding community organization discovered by CPM.

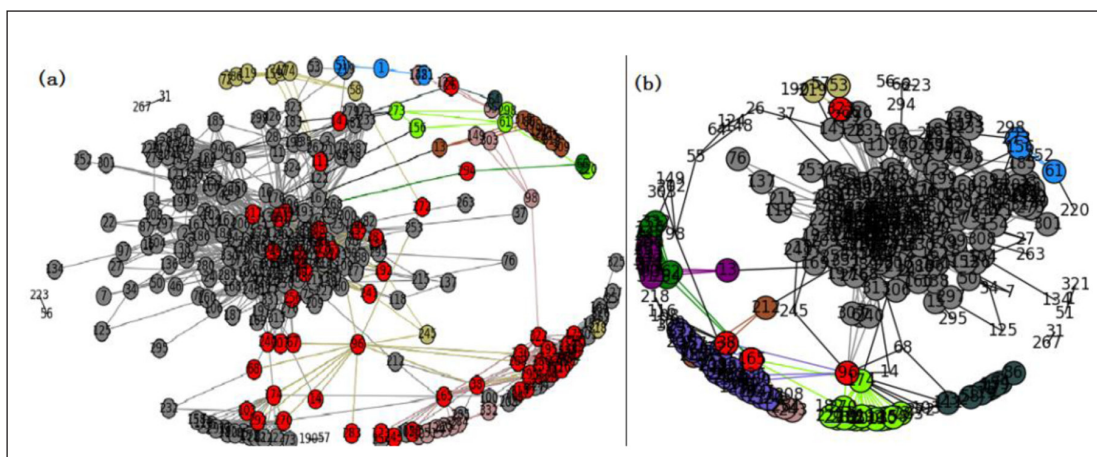


Figure 4: The Facebook ego-network and its corresponding community structure. Different communities rendered in different color, the nodes and links belong to same community rendered in same color. Red overlapping nodes are overlaps. **(a)** Community structure detected by TPM. **(b)** The corresponding community organization discovered by CPM.

communities; (2) when the k is set to 4, the CPM obtained 89 communities with 393 nodes, and the remained 2315 nodes are not included in corresponding communities; (3) when the value k is 5, the CPM only acquired 7 communities with only 36 nodes. But, the TPM also got 300 communities with 2708 nodes, that is, all nodes are classified into corresponding communities. The reason for the above correspondences is that the cliques used in the CPM are closed structures, there are edges between all nodes, and seldom in online social networks, thus only few nodes can be clustered by the CPM. Fortunately, thanks to the TPM utilized the triads, which not only contains the close-triads similar to the 3-clique, but also contains the open-triads, thus combined with the statements in section 3.2, the TPM can classified all nodes into corresponding communities and more suitable for community discovery in sparse networks.

4.5. LFR benchmark network

This section compares the performance of our algorithm with that of the algorithm CPM (Palla et al. 2005), LFM (Lancichinetti, Fortunato & Kertész 2009) and CNM (Clauset, Newman & Moore 2004). The comparative experiments were run on the benchmark networks generated by the LFR software (Lancichinetti, Fortunato & Radicchi 2008). The LFR software, released by Lancichinetti and his colleagues (2008), can generate artificial simulation networks with benchmark community and corresponding statistical information, which has been widely applied in the field of community detection algorithm evaluation. However, the parameters of LFR employed in this article were set as follows: number of nodes $N = 500,000$, average degree $k = 500$, maximum degree $maxk = 1000$, minimum community size $minc = 1000$, maximum community size $maxc = 1000$, and the mixing parameter mu from 0.1 to 0.9 was adjusted, where the mixing parameter is not the average ratio of external degree/total degree (as it used to be) but the maximum (or the minimum) of that distribution. Considerably, the greater the value of mu , the harder it is to detect corresponding communities. Thus, nine types of benchmark networks were obtained, each type generating ten networks by using the same parameters and averaging the test results. Meanwhile, we take advantage of NMI (Normalized Mutual Information) as a criterion to measure the performance of these four algorithms (Lancichinetti, Fortunato & Kertész 2009), and the NMI is widely used to evaluate the difference between the results of community detection and the benchmark community division in the field of network analysis.

The experimental consequences, as shown in **Figure 5**, indicate that TPM performance is just slightly lower than the other three algorithms when the mu is less-than-or-equal to 0.5. However, when mu is greater

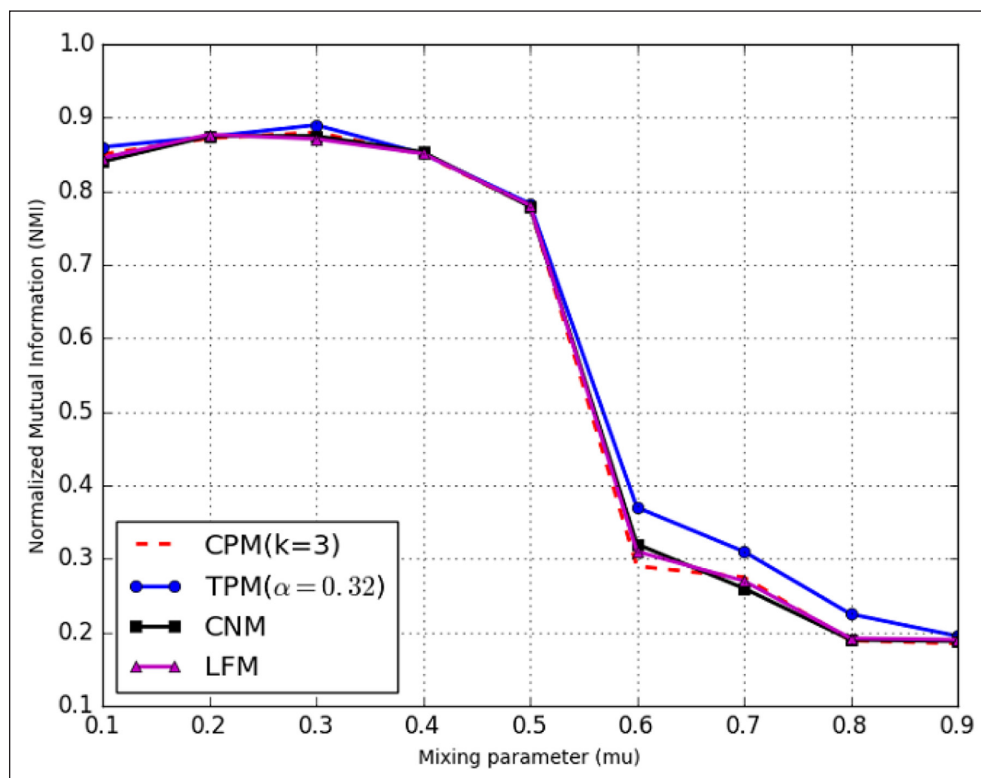


Figure 5: Comparative experiments on benchmark networks generated by LFR software.

than 0.5, the algorithm TPM performs better than the remains, even though the performance decreases with the mixing parameter μ increasing, which is due to the fact that TPM can cluster many nodes with the degree of 1, and the remaining algorithms do not have the above advantages.

5. Conclusions and future directions

In this paper, a triad-based community detection algorithm TPM was proposed. First, all triads which include close-triads and open-triads were found. A seed close-triad or open-triad was selected to expand to a basic community. Finally, the two communities with belonging coefficient greater than α were merged into a newer community, until there is no any pair of communities with higher belonging coefficient than α . The experimental results examined on different real-world networks and artificially simulated benchmark network indicate that our method is useful and effective.

The biggest advantage of the algorithm TPM is that it not only discovers communities from networks with typical community structures, but also detects communities from sparse networks with many open triads. We can obtain different granularity community structure by adjusting the belonging coefficient α , and so as to achieve the purpose of analyzing the network topology from diverse aspects. Meanwhile, due to the TPM is based on the idea of triads (triples), it can be easily implemented on a distributed graph processing platform for large scale network, such as GraphX on the Spark platform.

However, the drawback of the algorithm TPM is its large time-complexity $O(n^2)$, hence there are at least two directions in which could be extended for this work. First, due to the TPM is so extremely time-intensive that additional improvements are needed in future work, one of the approaches we adopted is to implement the TPM by using the GraphX package provided by Spark. Besides, another drawback of TPM is the poor efficiency detection of overlapping communities in large-scale LFR benchmark networks, especially in term of recall, precision and F-value of detected overlaps, however the detailed underlying reasons have been analyzed and corresponding improvements were made in overlapping community detection in our next paper. Finally, an application of the TPM with weighted and directed networks is also needed for future research issues.

Acknowledgements

This paper was supported in part by the National Natural Science Foundation of China under Grant 61702355, Key Natural Science Project of Anhui Provincial Education Department under Grant KJ2018A0448 and KJ2018A0449, Daze Scholar Project of Suzhou University under Grant 2018SZXYDZXZ01, Scientific and Technological Project of Suzhou under Grant SZ2017GG39, Coordinated Education Project of the Ministry of Education of China under Grant 201702139004, and the Major Projects of Teaching Research of Anhui Province under Grant 2016jyxm1026.

Competing Interests

The authors have no competing interests to declare.

References

- Aggarwal, C** and **Subbian, K**. 2014. Evolutionary network analysis: A survey. *ACM Computing Surveys*, 47(1): 1–36. DOI: <https://doi.org/10.1145/2601412>
- Ahn, YY, Bagrow, JP** and **Lehmann, S**. 2010. Link communities reveal multi-scale complexity in networks. *Nature*, 466: 761–764. DOI: <https://doi.org/10.1038/nature09182>
- Amelio, A** and **Pizzuti, C**. 2014. Overlapping Community Discovery Methods: A Survey. In: Gündüz-Öğüdücü, S and Etaner-Uyar, A (eds.), *Social Networks: Analysis and Case Studies. Lecture Notes in Social Networks*, 105–125. Springer, Vienna. DOI: https://doi.org/10.1007/978-3-7091-1797-2_6
- Bron, C** and **Kerberosch, J**. 1973. Algorithm 457: Finding all cliques in an undirected graph, *Community. ACM*, 16(9): 575–577. DOI: <https://doi.org/10.1145/362342.362367>
- Clauset, A, Newman, ME** and **Moore, C**. 2004. Finding community structure in very large networks. *Physical Review E*, 70(2): 066111. DOI: <https://doi.org/10.1103/PhysRevE.70.066111>
- Conrad, L, Reid, F, McDaid, A** and **Hurley, N**. 2010. Detecting highly overlapping community structure by greedy clique expansion. In: *The 4th SNA-KDD Workshop'10, Washington, DC USA*, 10. July 25. URL: <http://hdl.handle.net/10197/2516>.
- Cui, YZ, Wang, XY** and **Li, JQ**. 2014. Detecting overlapping communities in networks using the maximal sub-graph and the clustering coefficient. *Physica A: Statistical Mechanics and its Applications*, 405: 85–91. DOI: <https://doi.org/10.1016/j.physa.2014.03.027>


- Delvenne, JC, Yaliraki, SN and Barahona, M.** 2010. Stability of graph communities across time scales. *Proceedings of the National Academy of Sciences*, 29: 12755–12760. DOI: <https://doi.org/10.1073/pnas.0903215107>
- Fagnan, J, Zaïane, O and Barbosa, D.** 2014. Using Triads to Identify Local Community Structure in Social Networks. In: *2014 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM 2014)*, 108–112. Beijing, China, August 17–20. DOI: <https://doi.org/10.1109/ASONAM.2014.6921568>
- Fortunato, S.** 2010. Community detection in graphs. *Physics Reports*, 486(3): 75–174. DOI: <https://doi.org/10.1016/j.physrep.2009.11.002>
- Fortunato, S and Castellano, C.** 2012. Community structure in graphs. In: Meyers, R (ed.), *Computational Complexity*, 490–512. Springer, New York, NY. DOI: <https://doi.org/10.1007/978-1-4614-1800-9>
- Han, H and Tang, J.** 2015. Probabilistic community and role model for social networks. In: *Proceedings of the 21th ACM SIGKDD international conference on Knowledge Discovery and Data Mining*, 407–416. Sydney, NSW, Australia.
- Lancichinetti, A, Fortunato, S and Radicchi, F.** 2008. Benchmark graphs for testing community detection algorithms. *Physical Review E*, 78(4). DOI: <https://doi.org/10.1103/PhysRevE.78.046110>
- Lancichinetti, A, Fortunato, S and Skertész, J.** 2009. Detecting the overlapping and hierarchical community structure in complex networks. *New J. Phys*, 11(3). DOI: <https://doi.org/10.1088/1367-2630/11/3/033015>
- Leskovec, L, Lang, KJ, Dasgupta, A and Mahoney, MW.** 2008. Statistical properties of community structure in large social and information networks. In: *The 17th International World Wide Web Conference, WWW'08*, 695–704. Beijing, China. April 21–25. DOI: <https://doi.org/10.1145/1367497.1367591>
- Leskovec, J, Lang, KJ and Mahoney, M.** 2010. Empirical comparison of algorithms for network community detection. In: *Proceedings of the 19th International Conference on World Wide Web*, 631–640. Raleigh, North Carolina, USA. DOI: <https://doi.org/10.1145/1772690.1772755>
- Liu, D, Jin, D, He, D, Huang, J, Yang, J and Yang, B.** 2013. Community mining in complex networks. *Journal of Computer Research & Development*, 50(10): 2140–2154. URL: http://crad.ict.ac.cn/EN/abstract/article_1338.shtml.
- Lusseau, D, Schneider, K, Boisseau, OJ, Haase, P, Slooten, E and Dawson, SM.** 2003. The bottlenose dolphin community of doubtful sound features a large proportion of long-lasting associations—Can geographic isolation explain this unique trait. *Behavioral Ecology and Sociobiology*, 54(4): 396–405. DOI: <https://doi.org/10.1007/s00265-003-0651-y>
- McAuley, J and Leskovec, J.** 2012. Learning to Discover Social Circles in Ego Networks. In: *Advances in Neural Information Processing System 25 (NIPS 2012)*, curran associates, Inc., 539–547. URL: <http://papers.nips.cc/paper/4532-learning-to-discover-social-circles-in-ego-networks.pdf>.
- Mu, CH, Liu, Y, Wu, JS and Jiao, LC.** 2014. Two-stage algorithm using influence coefficient for detecting the hierarchical, non-overlapping and overlapping community structure. *Physica A: Statistical Mechanics and its Applications*, 408: 47–61. DOI: <https://doi.org/10.1016/j.physa.2014.04.023>
- Newman, MEJ.** 2006. Modularity and community structure in networks. *Proceedings of the National Academy of Sciences of the United States of America*, 103(23): 8577–8696. DOI: <https://doi.org/10.1073/pnas.0601602103>
- Palla, G, Derényi, I, Farkas, I and Vicsek, T.** 2005. Uncovering the overlapping community structure of complex networks in nature and society. *Nature*, 435(7043): 814–818. DOI: <https://doi.org/10.1038/nature03607>
- Peel, L, Larremore, DB and Clauset, A.** 2017. The ground truth about metadata and community detection in networks. *Science Advances*, 3(5): 1–8. DOI: <https://doi.org/10.1126/sciadv.1602548>
- Pizzuti, C.** 2018. Evolutionary Computation for Community Detection in Networks: A Review. *IEEE Transactions on Evolutionary Computation*, 22(3): 464–483. DOI: <https://doi.org/10.1109/TEVC.2017.2737600>
- Reichardt, J and Bornholdt, S.** 2006. Statistical mechanics of community detection. *Physical Review E*, 74: 016110. DOI: <https://doi.org/10.1103/PhysRevE.74.016110>
- Shen, HW, Cheng, XQ, Cai, K and Hu, MB.** 2009. Detect overlapping and hierarchical community structure in networks. *Physica A: Statistical Mechanics and its Applications*, 38: 1706–1712. DOI: <https://doi.org/10.1016/j.physa.2008.12.021>
- Shi, C, Cai, Y, Fu, D, Dong, Y and Wu, B.** 2013. A link clustering based overlapping community detection algorithm. *Data Knowl. Eng.*, 87: 394–404. DOI: <https://doi.org/10.1016/j.datak.2013.05.004>

- Veldt, N, Gleich, DF and Wirth, A.** 2018. A Correlation Clustering Framework for Community Detection. In: *Proceedings of the 2018 World Wide Web Conference*, 439–448. Lyon, France. April 23–27. DOI: <https://doi.org/10.1145/3178876.3186110>
- Wagenseller, P and Wang, F.** 2018. Size Matters: A Comparative Analysis of Community Detection Algorithms. *IEEE Transactions on Computational Social Systems*, 1–10. DOI: <https://doi.org/10.1109/TCSS.2018.2875626>
- Wang, L.** 2011. Using the relationship of shared neighbors to find hierarchical overlapping communities for effective connectivity in IoT. In: *The 6th International Conference on Pervasive Computing and Applications, Port Elizabeth*, 400–406. South Africa. Oct. 26–28. DOI: <https://doi.org/10.1109/ICPCA.2011.6106538>
- Xie, J, Kelley, S and Szymanski, BK.** 2013. Overlapping community detection in networks: The state-of-the-art and comparative study. *ACM Comput. Surv. (CSUR)*, 45(4): 1–35. DOI: <https://doi.org/10.1145/2501654.2501657>
- Xie, JR, Szymanski, BK and Liu, XM.** 2011. SLPA: Uncovering overlapping communities in social networks via a speaker–listener interaction dynamic process. In: *The 11th IEEE International Conference on Data Mining Workshops*, 344–349. Vancouver, BC, Canada. Dec. 11–11. DOI: <https://doi.org/10.1109/ICDMW.2011.154>
- Xu, Y, Xu, H, Zhang, D and Zhang, Y.** 2016. Finding overlapping community from social networks based on community forest model. *Knowledge-Based Systems*, 109: 238–255. DOI: <https://doi.org/10.1016/j.knsys.2016.07.007>
- Ye, Q, Wu, B, Zhao, ZX and Wang, B.** 2011. Detecting link communities in massive networks. In: *The 2011 International Conference on Advances in Social Networks Analysis and Mining*, 71–78. Kaohsiung, Taiwan. July 25–27. DOI: <https://doi.org/10.1109/ASONAM.2011.53>
- Zachary, WW.** 1977. An information flow model for conflict and fission in small groups. *Journal of Anthropological Research*, 33(4): 452–473. DOI: <https://doi.org/10.1086/jar.33.4.3629752>
- Zhang, ZW and Wang, ZY.** 2015. Mining overlapping and hierarchical communities in complex networks. *Physica A: Statistical Mechanics and its Applications*, 421: 25–33. DOI: <https://doi.org/10.1016/j.physa.2014.11.023>
- Zhao, Z, Zheng, S, Li, C, Sun, J, Chang, L and Chiclana, F.** 2018. A comparative study on community detection methods in complex networks. *Journal of Intelligent and Fuzzy Systems*, 35(1): 1–10. DOI: <https://doi.org/10.3233/JIFS-17682>

How to cite this article: Zhang, Z, Cui, L, Pan, Z, Fang, A and Zhang, H. 2018. A Triad Percolation Method for Detecting Communities in Social Networks. *Data Science Journal*, 17: 30, pp.1–12. DOI: <https://doi.org/10.5334/dsj-2018-030>

Submitted: 19 February 2018 **Accepted:** 02 November 2018 **Published:** 26 November 2018

Copyright: © 2018 The Author(s). This is an open-access article distributed under the terms of the Creative Commons Attribution 4.0 International License (CC-BY 4.0), which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited. See <http://creativecommons.org/licenses/by/4.0/>.

 *Data Science Journal* is a peer-reviewed open access journal published by Ubiquity Press.

OPEN ACCESS 