

# MTH 402 CAPSTONE

## COMMUNITY DETECTION IN NETWORKS

Cason Konzer  
University of Michigan - Flint  
*casonk@umich.edu*

October 23, 2022

### Abstract

Network analysis is a growing field of study in Mathematics, Computer and Data Science. In the past decades many advances have been made in algorithm development and computational power. Applications of network analysis spread wide across many fields, some interesting examples include supply chain, social media, author citation and web page networks. In this paper we start by reviewing motivations for community detection in networks, and what defines well selected communities. To do so we discuss papers in theory, and in application, while supplementing network basics required. We conclude by reviewing the development of community detection algorithms, and an in depth view of select a few.

### Keywords

Communities, Networks, Social Media

## 1 Introduction

Within mathematics, there are certain esteemed historical problems widely taught and known by those studying the field. To reminisce for a moment, I was introduced to two of these problems in my discrete mathematics class while in my second semester of undergraduate education. The seven bridges of Königsberg and the travelling salesman problem, both problems are prominent in graph theory. Both of these problems can be easily visualized, land masses connected by bridges and cities by roads, map reading is instilled within common human culture. In a similar manner a graph represented by nodes connected by edges is intuitive while scale is small. We can represent so much in this format, infrastructure, relationships, information flow, etc. Once networks become large, visualization and intuition fall off, and abstracts allow for greater understanding.

In a brief overview of the study of networks [16], Newman focuses on real world network types, properties of networks, null graph models and

processes taking place on networks. The study of networks in general, has drawn initial attention by mathematicians, but in the recent century, and moreover the recent decades, computer and data scientists alike have been having a field day with their applications. Within the even more recent years, the study of (social) media networks has grown rapidly. Individuals are being persuaded by bots, Trump is banned from twitter while president, and ideologies are becoming seemingly more radicalized. Community detection and clustering is an interesting approach to tackle these issues, what if we could identify bot nets, or predict attacks by leveraging media networks?

In 2008 the Louvain algorithm [2] for detecting communities within networks was published, and has become increasingly popular due to its advantageous run time. Additionally, the algorithm was shortly after extended to directed networks, akin to many media networks, and to include a ‘resolution’ parameter [11, 12] providing the ability to tune for community size/density. Lesser known, Clauset, Newman, and Moore laid

the underlying mathematics in 2004 [6], Newman then extended this to weighted networks soon after [17]. Additional community detections methods have emerged since, for example the Leiden [22] and BigCLAM [23] algorithms. What is common among many of these algorithms is their way of scoring the goodness of community partitions. In 2003 Newman proposed the metric so called modularity leveraging the number of inter-community edges in comparison to the expectation [18]. No matter the method, detecting communities is only an initial step to any research effort in media networks. To be of relevance, communities must be labeled in a meaningful way, and further their characteristics must be reviewed.

This type of study has been of persistent interest within the social sciences. In the late 80s Krackhardt and Stern [10] leveraged their university positions to study group dynamics of organizations via student assignments. A brief summary of their results finds that organizations which extend their reach externally performed superiorly. The so called ‘EI-Index’ was developed as a key metric measuring the relationship between internal and external links. ‘Echocambers’ and ‘Filter Bubbles’ have become common words within the literature, and studies on media networks such as Twitter, Facebook, and Reddit [3, 5, 9] are now commonplace. The effects of community structure and internet use in general has sparked similar research in sociology [4, 19, 20, 21], and become of an increased political interest with rising concerns of impacts on today’s youth. A thorough understanding of the structure of networks, community detection algorithms, and community analysis make for a optimistic outlook on solving societal problems stemming from media networks.

## 2 Network Basics

For ease of future discussion, let us set now some nomenclature and consider some of the first key features of networks. Common within literature there are two ways to refer to each of the two main ingredients of a network. Nodes, or vertices, are the crucial element of networks. Within the World Wide Web these would be web pages, within food webs these would be species, and within transit networks these may be thought of as intersections.

On the other hand we have edges, or links, which are the glue holding nodes together and explaining their interaction. Within the World Wide Web these are hyperlinks, or pointers from webpage to webpage, within food webs these are these represent the predator prey relationship, and within transit networks these are the roads themselves.

Following standard, we denote  $n$  to be the number of nodes in the network, and  $m$  the number of edges. Similarly, we will denote  $k_i$  as the degree of node  $i$ . Edges can be mutual, or undirected, such that an edge between node  $i$  and  $j$  is just that. Also edges can be weighted, one may think of weighted edges in the example of food webs as the proportion of a species diet, or within roads as the number of lanes. Last edges can be directed, for example a web-link from page  $i$  to page  $j$ , or a one-way street. In the study of real world networks, one will find common all of these edge attributes within a single graph.

It is useful now to introduce the configuration model, of which is described in depth by Newman in [15]. Under this model each node has a know degree, as is often the case in network data set. The degree of a node is equal to the number of edges incident on it. As each edge is incident upon two nodes, it follows closely that summing degree across all nodes should be twice that the number of edges within the network :  $\sum_i^n k_i = 2m$ . For here a useful visualization is provided in Figure 1, such that we view each node as depicted with spokes representing their edge connections. For the remainder of this paper we will denote  $s$  as the number of spokes, or twice the number of edges, within a network.

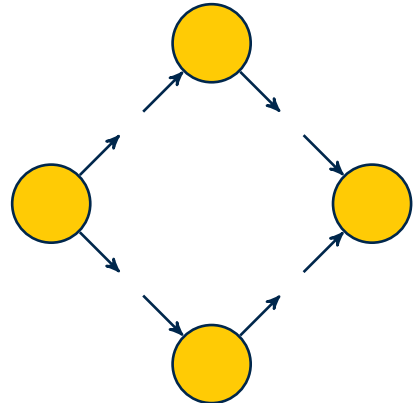


Figure 1: Spoke Representation

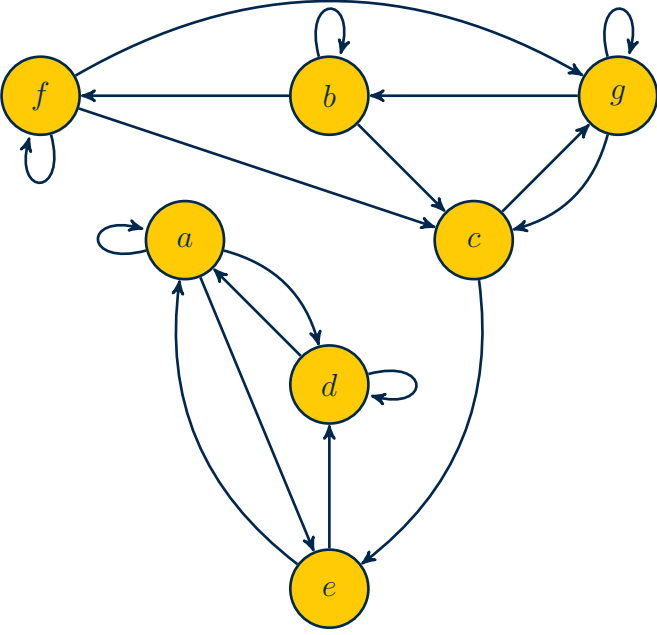


Figure 2: Example Network

A standard representation of networks is via an adjacency matrix, denoted  $A$ . The entry  $A_{i,j}$  represents the number of edges from node  $i$  to node  $j$ . A practical way to represent weighted networks is let the weights represent the number of edges, in the case of non-integer weights, a simple scaling can be applied to transform the matrix into one with integer only entries. In undirected networks, adjacency matrices exhibit symmetry across the diagonal. For the given network in Figure 2, we see the following directed adjacency matrix :

$A$	$a$	$b$	$c$	$d$	$e$	$f$	$g$
$a$	1	0	0	1	1	0	0
$b$	0	1	1	0	0	1	0
$c$	0	0	0	0	1	0	1
$d$	1	0	0	1	0	0	0
$e$	1	0	0	1	0	0	0
$f$	0	0	1	0	0	1	1
$g$	0	1	1	0	0	0	1

Similarly if we consider Figure 2 as an undirected graph, we have the adjacency matrix :

$A$	$a$	$b$	$c$	$d$	$e$	$f$	$g$
$a$	1	0	0	2	1	0	0
$b$	0	1	1	0	0	1	1
$c$	0	1	0	0	1	1	2
$d$	2	0	0	1	1	0	0
$e$	1	0	1	1	0	0	0
$f$	0	1	1	0	0	1	1
$g$	0	1	2	0	0	1	1

### 3 Good Communities

With the fundamentals of networks we can now dive into the underlying mechanics of many community detection algorithms, modularity. One underlying assumption of judging community fit is that the information to perform such judgment is provided within the network itself [1]. Of course in some real world networks the information may be available explicitly, such as groups formed on social media platforms. With this said such additional information can be used to baseline the metric here forward defined [18]. The form in which communities exist imposes an additional constraint, in that we will require communities to be disjoint. Of course this is again not always the case, as nodes may exist embedded in multiple communities.

The next assumption for good communities is that they have a proportionally higher number of internal links compared to that of external links. As the sum of these two proportions encompasses all of the links attached to nodes within a community it is sufficient to consider only one of the two figures. Good communities would then either have a maximized number of internal links, or otherwise a minimized number of external links. To clarify to the reader, internal links are edges connected to nodes of which both exist within the community. External links consist of strictly one node within the community, and edges connecting two nodes outside a community are disregarded for the figure. Following, it is such that analysis is conducted on a per community basis, and the goodness of the graph partition is evaluated as a sum of modularity values across all communities within the graph. With these assumption there is one issue persistent, always, the trivial case where the whole graph itself is selected as one giant community, would maximize modularity as outlined above [6].

To improve modularity such that it is a useful measure, we may consider the expectation of internal and external links as well. If we find that the communities detected within a graph have a higher number of internal links than the expected number of internal links, we are able to mitigate the issue with finding giant communities. In addition, it is often a question as to how many communities an algorithm should be tuned to find,

under the modularity maximization approach the number is discovered and corresponds to the number of communities found when modularity is at it's maximum.

Let us now examine the expected number internal edges. For initial analysis, let us consider the undirected network case. One piece of information we know is the degree of each node, and thus can leveraged analysis from the configuration model. Let us first consider the expected number of edges between any two nodes, denoted  $\mathbb{E}(i, j)$  for given nodes  $i$  and  $j$ . For two distinct nodes, we have  $k_i k_j$  ways to connected the spokes between them. For any one spoke, there exists  $s - 1$  other spokes to which it can connect, as a spoke cannot connect to itself. It follows that the probability one spoke connects with any other spoke is  $\frac{1}{s-1}$ . Summing of all possibly ways to connect two distinct nodes, we find then  $\mathbb{E}(i, j) = \frac{k_i k_j}{s-1}$ . In the case that the nodes are not distinct, e.g. the case for self loops, the node has  $k_i$  spokes, each of which could connect to  $k_i - 1$  other spokes attached to the same node. The the expected number of self edges for a given node is  $\mathbb{E}(i, i) = \frac{k_i(k_i-1)}{s-1}$ . A quick check summing over all node pairs, should yield the total number of spokes, as each edge is connected to two nodes.

$$\begin{aligned}
\sum_{i,j} \mathbb{E}(i, j) &= \sum_{i,j \neq i} \mathbb{E}(i, j) + \sum_i \mathbb{E}(i, i) \\
&= \sum_{i,j \neq i} \frac{k_i k_j}{s-1} + \sum_i \frac{k_i(k_i-1)}{s-1} \\
&= \frac{1}{s-1} \left[ \sum_i k_i \sum_{j \neq i} k_j + \sum_i k_i \sum_i k_i - \sum_i k_i \right] \\
&= \frac{1}{s-1} \left[ \sum_i k_i \sum_j k_j - s \right] \\
&= \frac{s^2 - s}{s-1} = \frac{s(s-1)}{s-1} = s
\end{aligned}$$

Knowing now the expected number of edges between two nodes, let us consider all nodes in a community, for all communities within the graph. Modularity is represent forward as the normalized value  $Q = C_{\% \text{ internal}} - C_{\% \mathbb{E} \text{ internal}}$ . Here  $C_{\% \text{ internal}}$  is the percentage of the internal community links within the graph and  $C_{\% \mathbb{E} \text{ internal}}$  is the expected percentage of the internal

community links within the graph respectively. The metric bounded by  $-1$  and  $1$ , in the case where our found internal links math exactly that of the expected number by random linking of nodes, we find  $Q = 0$ , which now corresponds to the trivial case where we find one giant community.

To formalize this method it is easiest to think in terms of spokes rather than edges. Letting  $c_i$  represent the community in which node  $i$  belongs, we can create an indicator function  $\delta(c_i, c_j)$  evaluating to  $1$  given nodes  $i$  and  $j$  are in the same community, and  $0$  otherwise. This provides an easy way to now identify the true percentage of internal community link found within the graph after detected.

$$C_{\% \text{ internal}} = \frac{1}{s} \left[ \sum_{i,j} A_{i,j} \delta(c_i, c_j) \right]$$

The indicator function additionally provides an easy way to calculate the expected percentage of internal links.

$$C_{\% \mathbb{E} \text{ internal}} = \frac{1}{s} \left[ \sum_{i,j \neq i} \frac{k_i k_j}{s-1} \delta(c_i, c_j) + \sum_i \frac{k_i(k_i-1)}{s-1} \right]$$

Form here we can now formalize modularity.

$$\begin{aligned}
Q &= \frac{1}{s} \left[ \sum_{i,j \neq i} \left( A_{i,j} - \frac{k_i k_j}{s-1} \right) \delta(c_i, c_j) \right. \\
&\quad \left. + \sum_i \left( A_{i,i} - \frac{k_i(k_i-1)}{s-1} \right) \right]
\end{aligned}$$

The traditional formulation of modularity makes two additional simplifications based on large scale real world networks. First, as networks grow large the subtraction of  $1$  from the number of possible spoke to spoke connections becomes negligible. Formally,  $\lim_{s \rightarrow \infty} \frac{k_i k_j}{s-1} = \frac{k_i k_j}{s}$ . Second, the number of self edges is proportionally small compared to the total number of edges, and can be treated as if the repeated nodes are distinct. Applying these two heuristics we come to the traditional definition of modularity.

$$Q = \frac{1}{s} \left[ \sum_{i,j} \left( A_{i,j} - \frac{k_i k_j}{s} \right) \delta(c_i, c_j) \right]$$

The definition calculates the modularity for the graph in whole by leveraging our indicator function. After detecting communities a simple evaluation of  $Q$  can quantify the goodness of the graph's partition.

Without any alteration to the definition, the definition holds for weighted graphs given the perspective shift from edge weightings to edge counts achieved by converting a weighted graph to a multigraph [17]. With a few adjustments, we can extend our definition to directed graphs as well [12]. First we must notice that each node has two degree types, indegree and outdegree, denoted  $k_i^{in}$  and  $k_i^{out}$  respectively for node  $i$ . We now have two equations  $\sum_i^n k_i^{in} = m$  and  $\sum_i^n k_i^{out} = m$ , such that  $\sum_i^n (k_i^{in} + k_i^{out}) = s$ . As we now count edges only once in our adjacency matrix, we divide our summation by  $m$  rather than  $s$ . Similarly in our expectation, the probability of an outward spoke connecting to any inward spoke becomes  $\frac{1}{m}$ . Within this case it is no longer necessary for the subtraction of the initial spoke as it is not one of the total inward spokes. Thus summing over all possible ways to have a edges going from node  $i$  to node  $j$  is  $\frac{k_i^{in} k_j^{out}}{m}$ . It follows nicely that our directed modularity  $Q_{dir}$  is similarly defined.

$$Q_{dir} = \frac{1}{m} \left[ \sum_{i,j} \left( A_{i,j} - \frac{k_i^{in} k_j^{out}}{m} \right) \delta(c_i, c_j) \right]$$

We now have the ability to judge graph community partition and additionally have a metric on which we wish to optimize within community detection algorithms. From experimental data, it is accepted that a modularity value greater than or equal to 0.3 represents significant community structure [6].

For an illustrative example consider Figure 3, given the selected communities identified by node color. We have the prior directed adjacency matrix, and the following expectation matrix :

$\mathbb{E}$	$a$	$b$	$c$	$d$	$e$	$f$	$g$
$a$	$\frac{9}{15}$	$\frac{9}{15}$	$\frac{3}{15}$	$\frac{6}{15}$	$\frac{6}{15}$	$\frac{9}{15}$	$\frac{9}{15}$
$b$	$\frac{15}{6}$	$\frac{15}{6}$	$\frac{15}{2}$	$\frac{15}{4}$	$\frac{15}{4}$	$\frac{15}{6}$	$\frac{15}{9}$
$c$	$\frac{15}{9}$	$\frac{15}{9}$	$\frac{3}{6}$	$\frac{15}{6}$	$\frac{15}{6}$	$\frac{15}{9}$	$\frac{15}{9}$
$d$	$\frac{15}{9}$	$\frac{15}{9}$	$\frac{3}{6}$	$\frac{15}{6}$	$\frac{15}{6}$	$\frac{15}{9}$	$\frac{15}{9}$
$e$	$\frac{15}{6}$	$\frac{15}{6}$	$\frac{15}{2}$	$\frac{15}{4}$	$\frac{15}{4}$	$\frac{15}{6}$	$\frac{15}{6}$
$f$	$\frac{15}{6}$	$\frac{15}{6}$	$\frac{15}{2}$	$\frac{15}{4}$	$\frac{15}{4}$	$\frac{15}{6}$	$\frac{15}{6}$
$g$	$\frac{15}{9}$	$\frac{15}{9}$	$\frac{3}{6}$	$\frac{15}{6}$	$\frac{15}{6}$	$\frac{15}{9}$	$\frac{15}{9}$

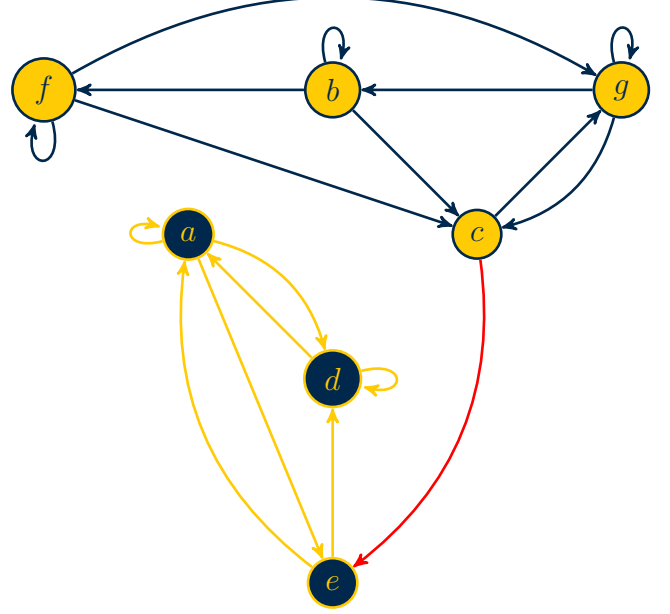


Figure 3: Example Community Partition

similarly  $A_{i,j} - \frac{k_i^{in} k_j^{out}}{m}$  becomes :

$A_{i,j} - \mathbb{E}$	$a$	$b$	$c$	$d$	$e$	$f$	$g$
$a$	$\frac{6}{15}$	$\frac{-9}{15}$	$\frac{-3}{15}$	$\frac{9}{15}$	$\frac{9}{15}$	$\frac{-9}{15}$	$\frac{-9}{15}$
$b$	$\frac{-6}{15}$	$\frac{9}{15}$	$\frac{13}{15}$	$\frac{-4}{15}$	$\frac{-4}{15}$	$\frac{9}{15}$	$\frac{-9}{15}$
$c$	$\frac{-9}{15}$	$\frac{-9}{15}$	$\frac{-3}{15}$	$\frac{-6}{15}$	$\frac{9}{15}$	$\frac{-9}{15}$	$\frac{6}{15}$
$d$	$\frac{15}{6}$	$\frac{15}{9}$	$\frac{-3}{15}$	$\frac{9}{15}$	$\frac{-6}{15}$	$\frac{-9}{15}$	$\frac{-9}{15}$
$e$	$\frac{15}{9}$	$\frac{15}{6}$	$\frac{15}{2}$	$\frac{11}{15}$	$\frac{15}{4}$	$\frac{15}{6}$	$\frac{15}{6}$
$f$	$\frac{15}{6}$	$\frac{15}{6}$	$\frac{15}{13}$	$\frac{15}{4}$	$\frac{15}{4}$	$\frac{15}{9}$	$\frac{15}{9}$
$g$	$\frac{-9}{15}$	$\frac{15}{6}$	$\frac{12}{15}$	$\frac{-6}{15}$	$\frac{15}{15}$	$\frac{-9}{15}$	$\frac{6}{15}$

$\delta(c_i, c_j)$  is :

$\delta(c_i, c_j)$	$a$	$b$	$c$	$d$	$e$	$f$	$g$
$a$	1	0	0	1	1	0	0
$b$	0	1	1	0	0	1	1
$c$	0	1	1	0	0	1	1
$d$	1	0	0	1	1	0	0
$e$	1	0	0	1	1	0	0
$f$	0	1	1	0	0	1	1
$g$	0	1	1	0	0	1	1

Last denoting  $||M||$  as the sum of all entries in the matrix  $M$ ,  $Q = \frac{1}{m} ||(A_{i,j} - \mathbb{E})\delta(c_i, c_j)|| = 0.426$ . We would thus consider that this graph exhibits strong community structure with the given partition. Due to the small scale of this graph it is easy to identify well a good partition, but as networks scale visualization becomes increasingly challenging, and thus such an approach as we have outlined is required for analysis.

## 4 Evolution of Algorithms

Algorithms are most commonly judged on complexity, but there is a clear tradeoff with accuracy. With modularity there is a proxy metric to judge the accuracy of a community detection algorithm. Traditionally, accuracy is defined with a ground truth, such that the true community structure is known within the network. Computational complexity is a good reference for scalability of an algorithm, and the upper limit of size of useable networks. Additionally as time goes on, as Moore’s Law realizes, a phenomena that the size of usable networks with fixed complexity increases dependent on the temporal increase in computational power. With that said for a given dataset, one may be able to execute a more complex algorithm with increased accuracy, in the same run time as a simpler algorithm in the same ran years prior. The evolution of algorithms plays on both of these characteristics.

The first proposal of the modularity to the best of our knowledge comes from Newman in [18]. The application was applied to the simplest of networks, undirected and unweighted. Modularity was simply used to evaluate goodness of community detection, and impose a stopping condition of the detection algorithm. Two detection techniques were utilized, both of which unutilized edge removal based on a betweenness measure. The simpler of the two methods, was to calculate shortest-path betweenness. The second method focused on random-walk, or otherwise equivalently current-flow betweenness. In both cases, the assumption is that many nodes must flow thorough community-to-community edges. In the case of removing these community connecting edges one expects to find the underlying communities. In general, the algorithm would be ran until no edges exist. After each edge removal, the graphs modularity is calculated. The best partition is then selected, such that modularity is maximized.

It is noticed that there are two forms of which modularity optimization can occur. Modularity can be single peaked, such that there is both one global and local maxima. Similarly modularity can be multi peaked, such that there is one global but multiple local maxima. In either case, one could halt the algorithm after finding the first local

maxima. In the multi peaked case the shall give a ‘good’ partition but not necessarily the best, introducing a probability for reduced accuracy. In the single peaked case this technique provides the optimal partition. Clearly, this will reduce the runtime of the algorithm, and is a tradeoff considered by the implemented dependent upon the size of their network.

Historically (pre 2000) algorithms were agglomerative (bottom up), finding communities by the addition of edges rather than removal, such as the well known k-means clustering [13, 14]. Newman’s work shifted the state of the are to decisive (top down) algorithms [8]. Soon after Newman’s initial work he extended his algorithm to weighted networks [17], thereafter developing an algorithm for modularity maximization with reduced complexity [6]. Later work by Newman extended the algorithm to directed networks [12] for increased accuracy, while in parallel Blondel, Guillaume, Lambiotte, and Lefebvre made significant strides in complexity reduction allowing for analysis of massive networks in the so-called Louvain algorithm [2]. Continuing later the Louvain algorithm was extend to directed networkes by Dugué and Perez to account for the additional information provided within the links [7]. Coming to some of the most recent developments the Louvain algorithm was transformed to the Leiden algorithm with an additional reduction in complexity by Traag, Waltman, and van Eck in 2019 [22].

## 5 Conclusion

In this disussion we have identified basics of networks, reviewed the mechanics of disjoint community detection within network, and discussed the history of community detection algorithms. Future work shall include an analysis of the algorithms aforementioned, providing deeper insights into the relevance of historical advancements. Additional work permits review of overlapping community detection algorithms, along with their sequential development. Worth while is to weight the pros and cons of disjoint vs. overlapping algorithms. As computational power and datasets grow, an understanding of where we come from is crucial to proper implementation and selection of cutting edge network algorithms.



# References

- [1] A.L. Barabási. *Network Science*. Cambridge University Press, 2016.
- [2] V. D. Blondel, J-L. Guillaume, R. Lambiotte, and E. Lefebvre. Fast unfolding of communities in large networks. *Journal of Statistical Mechanics: Theory and Experiment*, Oct 2008.
- [3] A. Bruns. Echo chamber? what echo chamber? reviewing the evidence. September 2017.
- [4] J. Cheng, C. Danescu-Niculescu-Mizil, and J. Leskovec. Antisocial behavior in online discussion communities. *Proceedings of the International AAAI Conference on Web and Social Media*, 9(1):61–70, Aug. 2021.
- [5] M. Cinelli, G. D. F. Morales, A. Galeazzi, W. Quattrociocchi, and M. Starnini. The echo chamber effect on social media. *Proceedings of the National Academy of Sciences*, 118(9):e2023301118, 2021.
- [6] A. Clauset, M. E. J. Newman, and C. Moore. Finding community structure in very large networks. *Physical Review E*, 70(6), dec 2004.
- [7] N. Dugué and A. Perez. Directed Louvain : maximizing modularity in directed networks. Research report, Université d’Orléans, November 2015. Please cite the following published version: <https://doi.org/10.1016/j.physa.2022.127798> rather than this one.
- [8] M. Girvan and M.E.J. Newman. Community structure in social and biological networks. *Proceedings of the National Academy of Sciences*, 99(12):7821–7826, 2002.
- [9] E. Guest. *Echoing Within and Between: Quantifying Echo Chamber Behaviours on Reddit*. University of Manchester, 2021.
- [10] D. Krackhardt and R. N. Stern. Informal networks and organizational crises: An experimental simulation. *Social Psychology Quarterly*, 51(2):123–140, 1988.
- [11] R. Lambiotte, J-C. Delvenne, and M. Barahona. Laplacian dynamics and multiscale modular structure in networks. *arXiv*, 1, 12 2008.
- [12] E. A. Leicht and M. E. J. Newman. Community structure in directed networks. *Physical Review Letters*, 100(11), mar 2008.
- [13] S. Lloyd. Least squares quantization in pcm. *IEEE Transactions on Information Theory*, 28(2):129–137, 1982.
- [14] J. MacQueen. Some methods for classification and analysis of multivariate observations. 1967.
- [15] M. Newman. *Networks*. Oxford University Press, 07 2018.
- [16] M. E. J. Newman. The structure and function of complex networks. *SIAM Review*, 45(2):167–256, 2003.
- [17] M. E. J. Newman. Analysis of weighted networks. *Physical Review E*, 70(5), nov 2004.
- [18] M. E. J. Newman and M. Girvan. Finding and evaluating community structure in networks. *Physical review. E, Statistical, nonlinear, and soft matter physics*, 69 2 Pt 2:026113, 2004.
- [19] C. T. Nguyen. Echo chambers and epistemic bubbles. *Episteme*, 17(2):141–161, 2020.
- [20] M. Samory and T. Mitra. Conspiracies online: User discussions in a conspiracy community following dramatic events. *Proceedings of the International AAAI Conference on Web and Social Media*, 12(1), Jun. 2018.
- [21] N. Seaver. Captivating algorithms: Recommender systems as traps. *Journal of Material Culture*, 24(4):421–436, 2019.
- [22] V. A. Traag, L. Waltman, and N. J. van Eck. From louvain to leiden: guaranteeing well-connected communities. *Scientific Reports*, 9(1):5233, Mar 2019.
- [23] J. Yang and J. Leskovec. Overlapping community detection at scale: A nonnegative matrix factorization approach. In *Proceedings of the Sixth ACM International Conference on Web Search and Data Mining*, WSDM ’13, page 587–596, New York, NY, USA, 2013. Association for Computing Machinery.