

SNSAlib: A python library for analyzing signed network

Ai-Wen Li(李艾纹), Jun-Lin Lu(陆俊霖), Ying Fan(樊瑛), and Xiao-Ke Xu(许小可)

Citation: Chin. Phys. B, 2025, 34 (**3**): 038902. DOI: 10.1088/1674-1056/ada439
Journal homepage: http://cpb.iphy.ac.cn; http://iopscience.iop.org/cpb

What follows is a list of articles you may be interested in

Source localization in signed networks with effective distance

Zhi-Wei Ma(马志伟), Lei Sun(孙蕾), Zhi-Guo Ding(丁智国), Yi-Zhen Huang(黄宜真), and Zhao-Long Hu(胡兆龙)
Chin. Phys. B, 2024, 33 (**2**): 028902. DOI: 10.1088/1674-1056/ad1482

Constructing refined null models for statistical analysis of signed networks

Ai-Wen Li(李艾纹), Jing Xiao(肖婧, and Xiao-Ke Xu(许小可)
Chin. Phys. B, 2021, 30 (**3**): 038901. DOI: 10.1088/1674-1056/abc2c4

Uncovering offline event similarity of online friends by constructing null models

Wenkuo Cui(崔文阔), Jing Xiao(肖婧), Ting Li(李婷), Xiaoke Xu(许小可)
Chin. Phys. B, 2019, 28 (**6**): 068901. DOI: 10.1088/1674-1056/28/6/068901

Community detection in signed networks based on discrete-time model

Jianrui Chen(陈建芮), Li Zhang(张莉), Weiwei Liu(刘维维), Zaizai Yan(闫在在)
Chin. Phys. B, 2017, 26 (**1**): 018901. DOI: 10.1088/1674-1056/26/1/018901

An improved genetic algorithm with dynamic topology

Kai-Quan Cai(蔡开泉), Yan-Wu Tang(唐焱武), Xue-Jun Zhang(张学军), Xiang-Min Guan(管祥民)
Chin. Phys. B, 2016, 25 (**12**): 128904. DOI: 10.1088/1674-1056/25/12/128904

SPECIAL TOPIC — Computational programs in complex systems

SNSAlib: A python library for analyzing signed network

Ai-Wen Li(李艾纹)[1], Jun-Lin Lu(陆俊霖)[2], Ying Fan(樊瑛)[1,†], and Xiao-Ke Xu(许小可)[2,‡]

[1] *School of Systems Science, Beijing Normal University, Beijing* 100875, *China*
[2] *School of Journalism and Communication, Beijing Normal University, Beijing* 100875, *China*

(Received 5 November 2024; revised manuscript received 12 December 2024; accepted manuscript online 31 December 2024)

The unique structure of signed networks, characterized by positive and negative edges, poses significant challenges for analyzing network topology. In recent years, various statistical algorithms have been developed to address this issue. However, there remains a lack of a unified framework to uncover the nontrivial properties inherent in signed network structures. To support developers, researchers, and practitioners in this field, we introduce a Python library named SNSAlib (Signed Network Structure Analysis), specifically designed to meet these analytical requirements. This library encompasses empirical signed network datasets, signed null model algorithms, signed statistics algorithms, and evaluation indicators. The primary objective of SNSAlib is to facilitate the systematic analysis of micro- and meso-structure features within signed networks, including node popularity, clustering, assortativity, embeddedness, and community structure by employing more accurate signed null models. Ultimately, it provides a robust paradigm for structure analysis of signed networks that enhances our understanding and application of signed networks.

Keywords: signed networks, null models, topology structure, statistic analysis

PACS: 89.75.Fb, 87.23.Ge, 05.10.–a

DOI: 10.1088/1674-1056/ada439 **CSTR:** 32038.14.CPB.ada439

1. Introduction

The signed networks represent a distinct category of networks with both positive and negative edges,[1] in which the positive edge typically indicates positive relationships denoted by the sign "+", whereas the negative edge indicates negative relationships denoted by the sign "−". It originally comes from Heider's balance theory,[2] which explores how interpersonal relationships tend to move towards a balanced state. Building on this, Cartwright and Harary provide a formal framework by applying graph theory,[3] making signed networks an effective medium and tool for analyzing opposing relationships within complex systems.[4] For example, in the society field, community members may publicly express friendship or hostility toward other members;[5] in the biological field, neurons can transmit both excitatory and inhibitory signals through synapses;[6] in the information field, semantic information of agreement or rejection is common in webpages.[7]

The analysis of signed networks highlights how the interplay of positive and negative edges affects network topology. It not only provides a robust theoretical foundation for understanding network evolution and dynamics within complex systems but also presents practical applications in areas such as personalized recommendations,[8] sentiment prediction,[9] opinion diffusion,[10] source tracing,[11] etc. It can be approached at three distinct levels: micro-level analysis emphasizes individual nodes or edges;[12,13] meso-level analysis focuses on signed community detection;[14,15] and macro-level analysis investigates the structure balance of signed networks.[16,17] Our research only focuses on the micro- and meso-levels algorithm integration of signed networks, because macro-level research is relatively mature and uses standardized and uniform algorithms.

Besides, to ensure the accuracy of structure analysis, a reasonable approach is to utilize the null model as a reference network for comparing statistic measures with those of the original network to obtain the relative values of statistics.[18,19] The null model represents a type of random network achieved by randomizing the edge positions or some specific elements within the original networks.[20] Employing null models can assist researchers in producing more robust findings in the field of network science. The signed network null models are usually constructed by changing the sign of edges, the edge position, or the community structure.[21] Notable applications include confirming social balance theory,[22] analyzing protein network structures,[23] and cross-validating the accuracy of community detection,[24] among others.

Despite the availability of various structural statistic indicators and null models specifically designed for signed networks, a notable gap persists in the form of a robust software framework capable of effectively integrating these diverse algorithms. Currently, there are

†Corresponding author. E-mail: yfan@bnu.edu.cn
‡Corresponding author. E-mail: xuxiaoke@foxmail.com

© 2025 Chinese Physical Society and IOP Publishing Ltd. All rights, including for text and data mining, AI training, and similar technologies, are reserved.

numerous mature tools and open-source libraries available for analyzing network structures, including NetworkX (https://networkx.org/), SNAP (http://snap.stanford.edu/), and EasyGraph (https://github.com/easy-graph/Easy-Graph), etc. However, their applicability is predominantly limited to general networks characterized by homogeneous edges, rendering them unsuitable for directly analyzing signed networks with both positive and negative edges. Therefore, there is an urgent need for specialized open-source libraries tailored to the structure analysis of signed networks.

To address these issues, SNSAlib provides a systematic and standardized platform that enables users to easily select micro- and meso-statistic methods for analyzing signed network datasets. It comprises ten empirical signed networks, seven signed null models (including five micro null models and two meso null models), eight statistic functions (consisting of six micro statistics and two meso statistics), and five evaluation indicators. SNSAlib is accessible at https://github.com/py-LIJI/signed-network-null-model. This platform allows for the acquisition of relative results based on various null models, thereby accurately revealing the unique structure characteristics of signed networks. The performance of SNSAlib is experimentally assessed through the analysis of node property, clustering, assortativity, embeddedness, and community structure across multiple datasets, thus highlighting the advantages of SNSAlib in examining signed networks.

2. SNSAlib: Signed network structure analysis library

In order to systematically, accurately, and conveniently measure the structure characteristics of signed networks, we have developed SNSAlib — "(S) Signed (N) Network (S) Structure (A) Analysis library". All algorithms within SNSAlib are implemented in Python and utilize the PyCharm integrated development environment. The algorithm library is designed for deployment on a general-purpose x86 server equipped with 8 cores and 16 threads, operating on the CentOS system with 16 GB of physical memory and a 512 GB hard drive. To facilitate flexible usage of the SNSAlib by researchers, various interfaces are provided for third-party library development on the PyCharm platform. Additionally, Docker encapsulation is employed to package both the algorithm library code and its runtime environments.

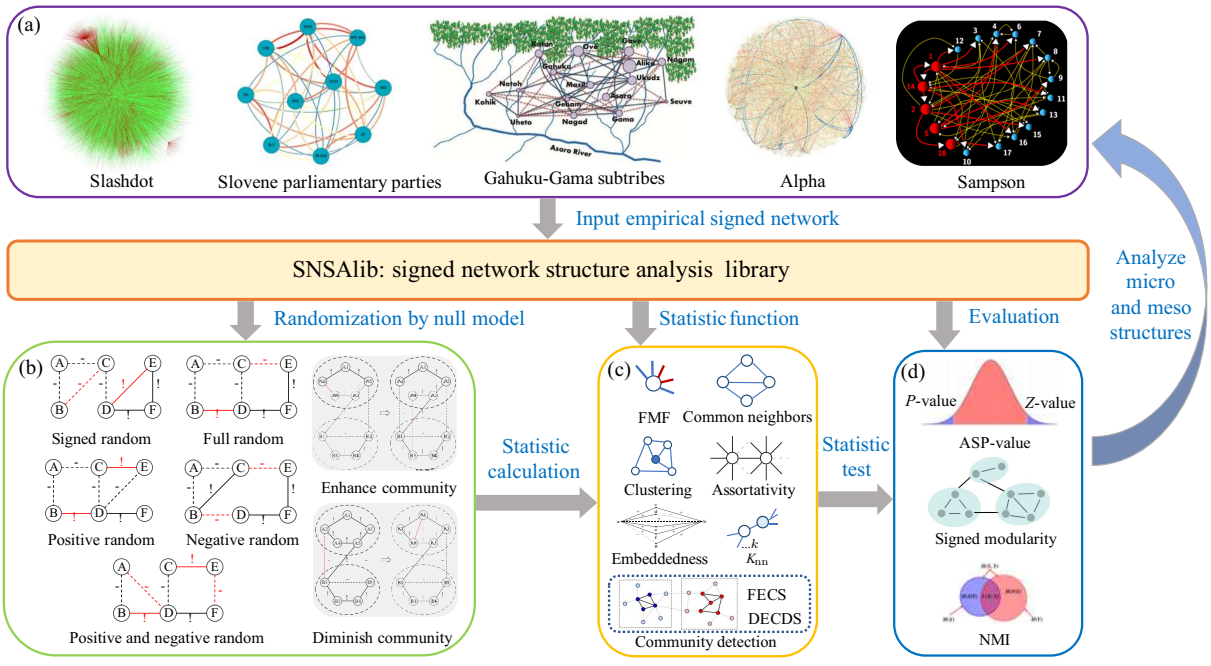


Fig. 1. Framework of SNSAlib: (a) empirical signed networks, (b) signed null models, (c) signed statistic functions, and (d) evaluation methods.

The construction of SNSAlib comprises four components: signed network datasets, null model algorithm suite, statistic function algorithm suite, and evaluation indicators. This library primarily relies on various signed null models as foundational algorithms, combined with multiple structure statistic calculation methods to analyze the structural characteristics of the original signed network at both micro and meso scales. An overview of the SNSAlib framework is illustrated in Fig. 1. Initially, several empirical signed networks are depicted in Fig. 1(a), which are input into the SNSAlib, and the dataset is processed. Through a conversion program, the dataset is entered as a triple, where $+1$ and -1 represent the positive and negative edges. Subsequently, seven types of null models within the SNSAlib algorithm suite are employed to generate random signed networks corresponding to the original networks, as shown in Fig. 1(b). Following this step, the statistic functions from the algorithm suite depicted in Fig. 1(c) are invoked to derive micro- and meso-statistics

for both the original signed network and its associated null models. Finally, the evaluation method from the SNSAlib illustrated in Fig. 1(d) is utilized to analyze the statistical significance of structure at micro- and meso-levels. This ultimately yields insights into general structure features present within empirical signed networks.

2.1. Signed network datasets

This library involves 10 empirical signed network datasets: SPP, GGS, Wiring, Sampson, Teams, Alpha, OTC, Wiki, Slashdot, and Epinions. The first five datasets are sourced from offline real-world social networks, and the latter five are obtained from online internet platforms.[25] The details of these datasets are as follows:

SPP (Slovene Parliamentary Parties 1994) is derived from the network of relationships between parliamentary parties in Slovenia in 1994.[26] In this network, the nodes represent the political parties in the Slovenian parliament, and the edges represent the estimated relationships between each pair of parties, with weights ranging from -3 to $+3$.

GGS (Gahuku-Gama subtribes) is a relationship network concerning the highland tribal groups. It was initially collected by Read in 1954 among the Gama subtribes in eastern New Guinea[27] and was later applied by Hage and Harary in 1983 as a network representing the relationships among these tribal groups.[28]

Wiring refers to the network of social relationships among 14 workshop workers at the Hawthorne plant of the Western Electric Company.[29,30] Investigators observed the interactions between workers, assigning positive values to denote friendships and negative values to indicate hostile relationships. In total, a comprehensive analysis revealed 32 distinct relationships among the workers.

Sampson is a social relationship network among monks recorded by Sampson.[30] The relationships between monks are quantified on a scale where -3 indicates strong dislike, -2 indicates moderate dislike, -1 indicates slight dislike, 1 indicates slight liking, 2 indicates moderate liking, and 3 indicates strong liking.

Teams is a social relationship network among pilots at the US Air Force Flight School.[31,32] This dataset was derived from social tests conducted by flight instructors on 48 pilots. In these assessments, each trainee was asked to identify their preferred and non-preferred flying partners. A willingness to serve as a flight partner is denoted by 1, while an unwillingness is represented by -1.

Alpha and OTC both originate from Bitcoin online trading platforms.[33] The former is derived from the Alpha platform, and the latter from the OTC platform. To mitigate the risk of ordinary users unknowingly engaging in transactions with fraudulent individuals during Bitcoin exchanges, these platforms enable users to label one another. Users can categorize their trading partners as either normal or fraudulent based on their experiences. Subsequently, other users can choose their trading partners based on these public labels, helping them avoid fraudulent transactions.

Wiki originates from the Wikipedia platform and refers to the network generated during the election process for Wikipedia administrators.[34] On Wikipedia, each entry is overseen by an administrator. The selection process for administrators commences with users self-nominating and submitting their candidacies. Subsequently, users or other administrator candidates vote, either in favor or against the candidate. The final determination of administrators is based on the aggregate count of supporting and opposing votes.

Slashdot originates from the technology news website Slashdot.com.[35] On this platform, each user can post news content, and other users can express their opinions by marking the content as liked or disliked. These different opinions among users form the Slashdot signed network.

Epinions is derived from the product review website Epinions.com,[35] in which each user can share their experiences with products, and other users can evaluate these reviews, either in favor or against. These different opinions among users form the Epinions signed network.

Empirical signed networks use varied weight intervals to denote user relationships. Thus, preprocessing is needed to standardize numerical values and formats. The values $+1$ and -1 are used to standardize positive and negative edges in the dataset. Then, the data format is transformed from a matrix list to a triplet representation stored in text files (.txt). These datasets are subsequently integrated into SNSAlib, allowing for direct access from the "signed network data" file within SNSAlib as original experimental data. An overview of the dataset information is summarized in Table 1. The scale of the ten networks in the table increases successively. Compared with the network with a small number of nodes, the positive edge ratio of the network with a large scale is generally larger.

Table 1. Overview of dataset information.

Network	SPP	GGS	Wiring	Sampson	Teams	Alpha	OTC	Wiki	Slashdot	Epinions
Nodes	10	14	16	18	48	3783	5881	10835	82140	131828
Edges	45	32	58	103	353	24186	35592	171800	549202	841372
Positive ratio	40.0%	40.6%	50.0%	54.4%	64.3%	93.6%	90.0%	77.4%	77.4%	85.3%
Negative ratio	60.0%	59.4%	50.0%	45.6%	35.7%	6.4%	10.0%	22.6%	22.6%	14.7%

2.2. Algorithm suite

The algorithm suite serves as the core component of SNSAlib, primarily consisting of signed null model algorithms and statistic function algorithms. First, Table 2 presents a collection of seven signed null model algorithms. These include five micro-level null models:[20,21] sign randomized null model, full-edge randomized null model, positive-edge randomized null model, negative-edge randomized null model, and positive- and negative-edges randomized null model. Besides, there are two meso-level null models:[36] enhanced community structure null model and diminished community structure null model. For a more intuitive understanding, the table not only provides a detailed description but also illustrates the generation process from the original network to its corresponding null model network by a schematic diagram. Additionally, Table 3 presents a collection of eight statistic function algorithms. These consist of six micro-level statistics: fans minus freaks (FMF), common neighbors, clustering coefficient, excess average degree, family of assortativity coefficients, and embeddedness level. Besides, there are two meso-level community detection algorithms: finding and extracting community in signed networks (FECS) and differential evolution community detection in signed networks (DECDS). The table not only provides a detailed description but also comes with formulas that illustrate the calculation process.

Table 2. Interface definitions of signed null models.

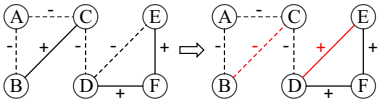
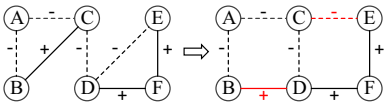
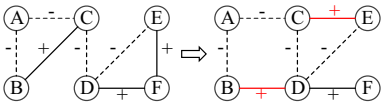
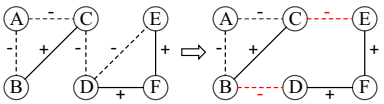
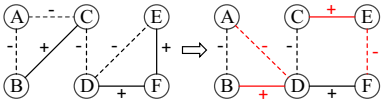
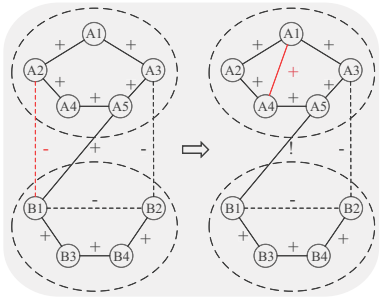
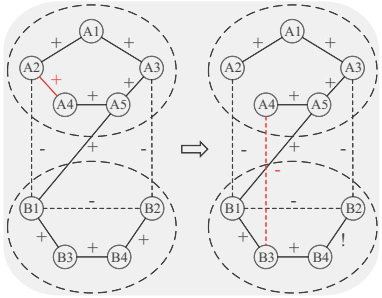
Name	Description	Schematic diagram
sign_network_sign_swap ($G0, nswap, max_tries$)	Sign randomized null model: modifies only the signs of edges, preserving edge positions and node degree sequences. This method assesses how edge signs affect network properties.	
sign_network_full_swap ($G0, nswap, max_tries$)	Full-edge randomized null model: randomizes both the edge positions and the signs of the network, resulting in a higher degree of randomness and preserving less structure information from the original networks.	
sign_network_positive_swap ($G0, nswap, max_tries$)	Positive-edges randomized null model: modifies only the positive edge topology of the original network, leaving the negative edge topology unchanged. It serves to analyze the effects of positive edge topology.	
sign_network_negative_swap ($G0, nswap, max_tries$)	Negative-edges randomized null model: modifies only the negative edge topology of the original network, while preserving the positive edge topology. It is useful for examining the impacts of negative edge topology.	
sign_network_part_swap ($G0, nswap, max_tries$)	Positive- and negative-edges randomized null model: modifies the positive and negative topologies successively. It has a stronger random degree than solely randomizing either topology but is weaker than full-edge randomized due to the lack of mutual random between positive and negative topologies.	
sign_enhance_swap ($G0, T, nswap, max_tries$)	Enhanced community structure null model: increases positive edges within communities and removes negative edges between them, making the signed network's community structure more distinct and easier to detect.	
sign_diminish_swap ($G0, T, nswap, max_tries$)	Diminished community structure null model: increases negative edges within communities while removing positive edges between them, making the signed network's community structure more obscure and harder to detect.	

Table 3. Interface definitions of statistic functions.

Name	Description	Formula
degree_difference (G)	Fans minus freaks (FMF): quantifies a node's popularity in signed networks.[37] Fans denote positive relationships, while freaks denote negative ones. A higher FMF value signifies greater node popularity.	$\mathrm{FMF}_i = k_i^+ - k_i^-$
common_neighbors (G)	Signed common neighbors (SCN): measures common neighbors considering positive and negative edges.[38] N_i^+/N_i^- and N_j^+/N_j^- represent the set of nodes whose neighbors of node i or j are positive/negative edges. A larger value of SCN_{ij} indicates stronger network connectivity.	$\mathrm{SCN}_{ij} = \left(\lvert N_i^+ \cap N_j^+\rvert + \lvert N_i^- \cap N_j^-\rvert\right) - \left(\lvert N_i^+ \cap N_j^-\rvert + \lvert N_i^- \cap N_j^+\rvert\right)$
clustering_coefficient (G)	Signed clustering coefficient C_s: quantifies the tendency of nodes to form small clusters. C_s is calculated by converting the signed network into an adjacency matrix.[37] A higher value of C_s indicates a higher network tightness.	$C_\mathrm{s} = \frac{\lVert A \circ A^2 \rVert_+}{\lVert \tilde{A}^2 \rVert_+}$
excess_average_degree (G)	Excess average degree $K_{\mathrm{nn}}(k)$: describes the number of nearest neighbors of nodes with degree k.[39] If it increases with k, the network is assortative; if it decreases, the network is disassortative; and if independent, there are no degree correlations.	$K_{\mathrm{nn}}(k) = \sum_{k'} k' \frac{kp(k,k')}{kp(k)} = \frac{\langle k^2 \rangle}{\langle k \rangle}$
matching_coefficient _family ($G, member$)	Family of assortativity coefficients of signed networks: is designed to measure the degree correlations in signed networks.[40,41] It includes six types of degree-degree mixing patterns with *member* representing an identification parameter. - If $member = 0$, calculate $r^+(+,+)$; - If $member = 1$, calculate $r^+(+,-)$; - If $member = 2$, calculate $r^+(-,-)$; - If $member = 3$, calculate $r^-(+,+)$; - If $member = 4$, calculate $r^-(+,-)$; - If $member = 5$, calculate $r^-(-,-)$.	$r^+(+,+) = \frac{M_+^{-1}\sum_i j_i^+ l_i^+ - \left[M_+^{-1}\sum_i \frac{1}{2}(j_i^+ + l_i^+)\right]^2}{M_+^{-1}\sum_i \frac{1}{2}\left((j_i^+)^2 + (l_i^+)^2\right) - \left[M_+^{-1}\sum_i \frac{1}{2}(j_i^+ + l_i^+)\right]^2}$ $r^+(+,-) = \frac{M_+^{-1}\sum_i j_i^+ l_i^- - \left[M_+^{-1}\sum_i \frac{1}{2}(j_i^+ + l_i^-)\right]^2}{M_+^{-1}\sum_i \frac{1}{2}\left((j_i^+)^2 + (l_i^-)^2\right) - \left[M_+^{-1}\sum_i \frac{1}{2}(j_i^+ + l_i^-)\right]^2}$ $r^+(-,-) = \frac{M_+^{-1}\sum_i j_i^- l_i^- - \left[M_+^{-1}\sum_i \frac{1}{2}(j_i^- + l_i^-)\right]^2}{M_+^{-1}\sum_i \frac{1}{2}\left((j_i^-)^2 + (l_i^-)^2\right) - \left[M_+^{-1}\sum_i \frac{1}{2}(j_i^- + l_i^-)\right]^2}$ $r^-(+,+) = \frac{M_-^{-1}\sum_i j_i^+ l_i^+ - \left[M_-^{-1}\sum_i \frac{1}{2}(j_i^+ + l_i^+)\right]^2}{M_-^{-1}\sum_i \frac{1}{2}\left((j_i^+)^2 + (l_i^+)^2\right) - \left[M_-^{-1}\sum_i \frac{1}{2}(j_i^+ + l_i^+)\right]^2}$ $r^-(+,-) = \frac{M_-^{-1}\sum_i j_i^+ l_i^- - \left[M_-^{-1}\sum_i \frac{1}{2}(j_i^+ + l_i^-)\right]^2}{M_-^{-1}\sum_i \frac{1}{2}\left((j_i^+)^2 + (l_i^-)^2\right) - \left[M_-^{-1}\sum_i \frac{1}{2}(j_i^+ + l_i^-)\right]^2}$ $r^-(-,-) = \frac{M_-^{-1}\sum_i j_i^- l_i^- - \left[M_-^{-1}\sum_i \frac{1}{2}(j_i^- + l_i^-)\right]^2}{M_-^{-1}\sum_i \frac{1}{2}\left((j_i^-)^2 + (l_i^-)^2\right) - \left[M_-^{-1}\sum_i \frac{1}{2}(j_i^- + l_i^-)\right]^2}$
embeddedness (G, n)	Embedding level: is measured by common neighbors,[35] classified into positive and negative embeddedness. n indicates the total number of networks to be calculated.	$CN_{ij}^+ = \left\lvert \Gamma_i^+ \cap \Gamma_j^+ \right\rvert$ $CN_{ij}^- = \left\lvert \Gamma_i^- \cap \Gamma_j^- \right\rvert$
sign_fec (G)	Finding and extracting community in signed networks (FECS): is used to detect communities in signed networks.[42] The cut-off function $P(c)$ is applied recursively until all potential communities are found.	$P(C) = \sum_k \sum_{i,j \in C_k} \max(0, -w_{ij}) + \sum_{r \neq s} \sum_{i \in C_r, j \in C_s} \max(0, w_{ij})$
sign_decd (G)	Differential evolution community detection in signed networks (DECDS): aims to detect communities in signed networks.[43] Its core principle is individual error correction based on community differences.	$CV(i) = \frac{\sum_{(i,j) \in E} f(i,j)}{k_i}$

Listing 1: Example of the execution of SNSAlib algorithm.

```
from SNSAlib import null_model
from SNSAlib import statistic_features
import networkx as nx

# Input original network
G0 = nx.read_edgelist(r'..\signed_network_data\1_SPP.txt', nodetype=int, data=(('weight',
    float), ))

# Output null model network
G = null_model.sign_network_sign_swap(G0, 400, 2000)

# FMF value and its corresponding the fraction of nodes are calculated
x,y=statistic_features.minus_dis(G0/G)
```

The above code shows an example of executing the SNSAlib algorithm. In lines 1–3, all required modules are imported; in line 6, the "SPP network dataset" is invoked as the original network using NetworkX; in line 9, the "sign randomized null model" algorithm is executed to obtain the randomized reference network (the other six null models can be called in the same way to generate the desired reference network by selecting the appropriate interface as shown in Table 2); in line 11, the "FMF" function is executed to obtain the FMF value (the other seven statistic functions can be called in the same way to analyze various statistic features of signed networks by selecting the appropriate interface as shown in Table 3).

2.3. Evaluation indicators

The evaluation of structure analysis can be conducted at both micro- and meso-levels. The micro-level evaluations primarily focus on determining whether there are significant differences between the original network and null model networks. They can provide insights into how various types of null models influence the topology of the original signed network. The meso-level evaluations primarily involve generating random networks based on null models to establish the controlled benchmark. They can assess the significance of the community structure in the actual signed network relative to random networks.

The detailed descriptions and formulas of three micro- and two meso-evaluation indicators from SNSAlib are presented in Table 4. At the micro-level, it includes three statistic tests: Z-score, P-value, and ASP-value, facilitating an analysis of its micro-level structure features. At the meso-level, it includes two indicators for community detection: signed modularity (Q_s) and normalized mutual information (NMI), facilitating an analysis of its meso-level structure features.

Table 4. Evaluation indicators of micro- and meso-levels.

Name	Description	Formula
Z-score test	It calculates the Z-score to evaluate the significance of differences between the statistic means of null model networks and the original network.[44]	$Z(j)=\frac{N(j)-\langle N_r(j)\rangle}{\sigma_r(j)}$
P-value test	It calculates the P-value to evaluate the significance of differences between the statistic means of null model networks and the original network.[45]	$P=1-\Phi(Z)$
ASP-value test	It calculates the ASP-value to evaluate the relative strength of the original network and null model network regarding assortativity and disassortativity. The ASP value is derived from normalizing the Z value.[46]	$\mathrm{ASP}(\alpha,\beta)=\frac{Z(\alpha,\beta)}{\left(\sum_{\alpha,\beta}Z(\alpha,\beta)^2\right)^{1/2}}$
Signed modularity (Q_s)	It calculates the Q_s value to evaluate the ability of community detection algorithms by comparing the difference in community structure between the actual network and random networks.[47]	$Q_s=\frac{1}{2M^+ +2M^-}\sum_{ij}\left[A_{ij}-\left(\frac{k_i^+k_j^+}{2M^+}-\frac{k_i^-k_j^-}{2M^-}\right)\right]\delta(C_i,C_j)$
Normalization mutual information (NMI)	It calculates NMI value to evaluate the ability of community detection algorithms by comparing the similarity of community structure between the actual network and random networks.[48]	$\mathrm{NMI}(A,B)=\frac{-2\sum_{i=1}^{C_A}\sum_{j=1}^{C_B}N_{ij}\log\left(\frac{N_{ij}N}{N_{i\cdot}N_{\cdot j}}\right)}{\sum_{i=1}^{C_A}N_{i\cdot}\log\left(\frac{N_{i\cdot}}{N}\right)+\sum_{j=1}^{C_B}N_{\cdot j}\log\left(\frac{N_{\cdot j}}{N}\right)}$

3. Performance evaluation of SNSAlib

Combined with various statistic functions provided by the algorithm library, the network structure analysis results of SNSAlib are presented under different datasets. At the micro level, the analysis encompasses node property, network clustering, network assortativity, and embeddedness. At the macro level, it focuses on community detection. This comprehensive evaluation demonstrates the advantages of SNSAlib in terms of functional systematization and structure analysis accuracy.

3.1. Node property

The node property, specifically the node popularity, is measured using the FMF statistic. The experiment utilizes the Epinions empirical signed network dataset. The SNSAlib operates as follows: First, the null model algorithm suite within SNSAlib is invoked to generate five different null models. Then, the FMF statistics of the original network and five null models are calculated. Finally, all FMF values are aggregated to determine the fraction of nodes exhibiting each FMF value relative to the total number of nodes.

The result is shown in Fig. 2. Overall, the original signed network and its five null models reveal a common trend: obeying the power-law distribution. Specifically, as the FMF value increases, the fraction of nodes decreases. This indicates that the node property of signed networks exhibits the rich club phenomenon similar to that in unsigned networks. That is, only a limited number of nodes within the network possess a high quantity of positive edges while maintaining a low quantity of negative edges. Besides, the results of the positive-edge randomized null model, the negative-edge randomized null model, and the positive- and negative-edge randomized null model overlap the original signed network. This is attributed to their retention of the positive and negative degree

sequences during randomization. In contrast, the results of the sign randomized null model and full-edge randomized null model diverge notably from the original network, because they randomize both positive and negative degree sequences, resulting in a higher level of randomness.

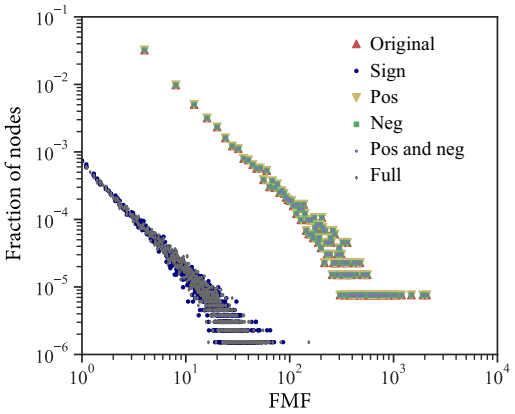


Fig. 2. The frequency distribution of FMF statistic in the original Epinion network and five null models. "Original", "Sign", "Pos", "Neg", "Pos and neg" and "Full" correspond to the original network, the sign randomized null model, the positive-edge randomized null model, the negative-edge randomized null model, the positive- and negative-edge randomized null model, and the full-edge randomized null model, respectively. This nomenclature applies similarly to terms used in subsequent figures.

3.2. Clustering

The clustering of the signed network structure, known as the tightness degree, can be measured using the common neighbors and clustering coefficient algorithms provided in SNSAlib.

3.2.1. Common neighbors

The clustering is measured using the common neighbor statistic, and the experiment utilizes the Alpha empirical signed network dataset. The SNSAlib operates as follows: First, the null model algorithm suite of SNSAlib is invoked to generate five different null models. Then, the common neighbors of the original network and five null models are calculated. Finally, the values of common neighbors are aggregated to calculate the fraction of nodes corresponding to each value of common neighbors relative to the total number of nodes.

The result is presented in Fig. 3. Overall, the original signed network and its five null models reveal a common trend: obeying the bell-shaped distribution. This pattern, characterized by high values in the middle and low values at both extremes, indicates that there are very few edges connecting most pairs of nodes, with only a limited number of nodes possessing a large number of connected edges. This feature is similar to the results obtained by using the FMF statistics. Besides, because five null models all change neighbor node sets of each node in the original network, the results of these null models do not overlap the original signed network.

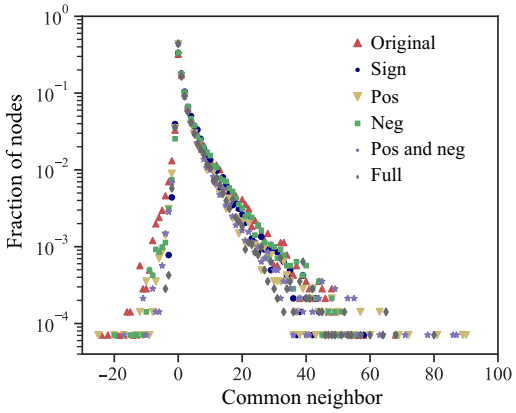


Fig. 3. The frequency distribution of common neighbors statistic in the original Alpha network and five null models.

3.2.2. Clustering coefficient

Then, the clustering is measured by the clustering coefficient, and the experiment uses eight empirical signed network datasets: GGS, Wiring, Sampson, Teams, Alpha, OTC, Wiki, and Slashdot. The SNSAlib operates as follows: First, the null model algorithm suite of SNSAlib is invoked to generate five different null models. Then, the clustering coefficient statistics of the original network and five null models are calculated. It should be noted that the empirical signed networks have low clustering coefficients, resulting in high error sensitivity. To reduce the effects of randomization errors, all null models are generated 50 times. Finally, the mean and standard deviation are calculated, and the results are compared with the original network.

The experimental results are presented in Table 5. Although the size and type of the networks are different, the clustering coefficients of the original signed networks are generally higher than those of null models, indicating stronger

Table 5. The clustering coefficient of original signed networks and the difference between the five null models compared with original networks.

Network	Original	Signed	Full	Pos	Neg	Pos and neg
GGS	0.388	-0.006 ± 0.065	-0.002 ± 0.067	0.034 ± 0.049	0.338 ± 0.031	-0.010 ± 0.064
Wiring	0.279	0.001 ± 0.102	0.014 ± 0.083	0.104 ± 0.042	0.126 ± 0.065	-0.055 ± 0.075
Sampson	0.086	0.000 ± 0.053	0.010 ± 0.046	0.011 ± 0.041	0.019 ± 0.031	-0.037 ± 0.040
Teams	0.141	-0.001 ± 0.017	0.003 ± 0.017	0.027 ± 0.014	0.106 ± 0.011	0.007 ± 0.015
Alpha	0.054	0.041 ± 0.001	0.030 ± 0.001	0.034 ± 0.01	0.052 ± 0.000	0.033 ± 0.001
OTC	0.042	0.020 ± 0.001	0.016 ± 0.001	0.025 ± 0.001	0.036 ± 0.000	0.023 ± 0.001
Wiki	0.058	0.022 ± 0.000	0.013 ± 0.000	0.023 ± 0.000	0.067 ± 0.000	0.022 ± 0.000
Slashdot	0.062	0.022 ± 0.001	0.028 ± 0.001	0.057 ± 0.001	0.061 ± 0.000	0.053 ± 0.001

clustering in the original networks. This suggests that the null model reduces the clustering of the original signed network by varying degrees. Among them, the clustering coefficients of the positive-edge randomized null model and the negative-edge randomized null model are greater than those of the positive- and negative-edge randomized null model, because the latter has a stronger randomization degree. In addition, the clustering coefficient of the negative-edge randomized null model is generally larger than that of the positive-edge randomized null model, because the ratio of positive edges is generally much larger than that of negative edges in empirical signed networks, resulting in the randomization degree of the former is lower than that of the latter. Therefore, it can be concluded that the impact of the correlation between positive and negative edges on the micro-topological structure of signed networks depends on the fraction of positive and negative edges within the network.

3.3. Assortativity

The assortativity of the signed network structure, described as degree–degree correlation, can be measured using the excess average degree and the family of assortativity coefficients provided in SNSAlib. The excess average degree is more focused on theoretical analysis, and the family of assortativity coefficients offers a more precise mathematical measurement.

3.3.1. Excess average degree

The assortativity is measured by the excess average degree, and the experiment uses the Epinions empirical signed network dataset. The SNSAlib operates as follows: First, the original network is divided into positive and negative subnetworks, and the null model algorithm suite of SNSAlib is invoked to generate five different null models. Then, the excess average degree statistic ($K_{\rm nn}$) of the original subnetworks and their five null models are calculated. Finally, the variation trend of $K_{\rm nn}$ with degree value k is observed in positive and negative subnetworks.

The experimental results are presented in Fig. 4. In Fig. 4(a), it can be observed that the excess average degree $K_{\rm nn}(k_+)$ of the original positive subnetwork (equivalent to the negative-edge randomized null model) increases with k, indicating assortativity in the network. The sign randomized null model also shows this property. However, the $K_{\rm nn}(k)$ of the positive-edge randomized null model (equivalent to the positive- and negative-edge randomized null model) and the full-edge randomized null model show no correlation with k, indicating that these three null models have no degree correlation. Furthermore, in Fig. 4(b), it can be seen that the excess average degree $K_{\rm nn}(k_-)$ of the original negative subnetwork (equivalent to the positive-edge randomized null model) tends to decrease as k increases, demonstrating disassortativity. The negative-edge randomized null model (equivalent to the positive- and negative-edge randomized null model) also exhibits the same property. However, the result of the sign randomized null model and the full-edge randomized null model is independent of k, showing no degree correlation.

Overall, it can be concluded that both assortativity and disassortativity coexist in signed networks. For the positive edge topology, nodes are assortative, meaning nodes with many positive edges tend to link with other nodes that also have many positive edges. For the negative edge topology, nodes are disassortative, where nodes with many negative edges tend to link with nodes that have fewer negative edges.

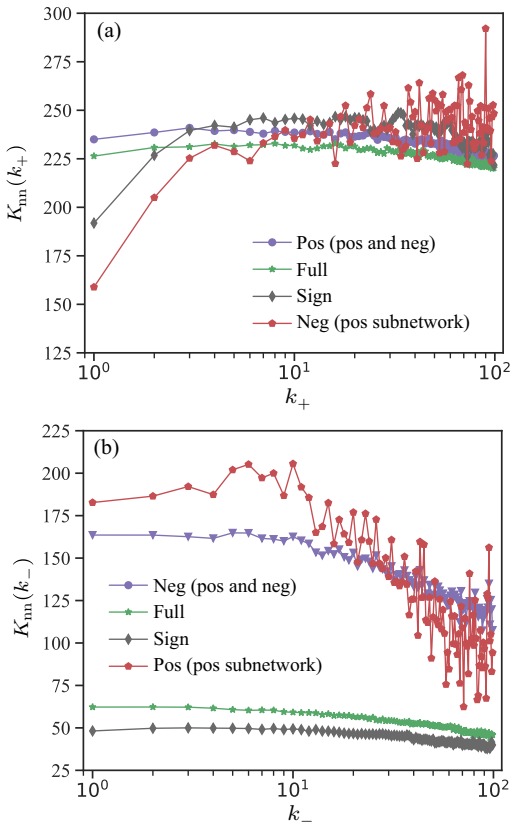


Fig. 4. The variation trend of excess average degree ($K_{\rm nn}(k)$) with the degree (k) in original Epinions empirical network and five null models. (a) The result of $K_{\rm nn}(k_+)$ in the Epinions positive subnetwork, (b) the result of $K_{\rm nn}(k_-)$ in the Epinions negative subnetwork.

3.3.2. The family of assortativity coefficients

In order to achieve a more accurate numerical analysis of assortativity, the family of assortativity coefficients can be calculated. The experiment utilizes four empirical signed network datasets: Alpha, OTC, Wiki, and Slashdot. The SNSAlib operates as follows: First, the null model algorithm suite of SNSAlib is invoked to generate the sign randomized null model and the full-edge randomized null model. Then, the family of assortativity coefficients statistic from the algorithm suite is employed to calculate the assortativity coefficients of the original network and each type of null model. To mini-

mize the impact of randomization errors, all assortativity coefficient values are independently calculated 50 times. Finally, the mean and standard deviation are computed as the final experimental results and evaluated by Z-score, P-value, and ASP-value.

The results are presented in Table 6. Z_{f} and P_{f} represent the Z-score and P-value corresponding to the full-edge randomized null model, and Z_{s} and P_{s} correspond to the sign randomization null model. In order to demonstrate the validity of the family of assortativity coefficients, that is, whether the difference in assortativity between the original network and null models can be statistically reflected, the hypothesis testing method is introduced to test the difference significance. Based on the two-tailed test, under the premise that the significance level is 0.05, the rejection regions for the Z-score are $(-\infty, 1.96)$ and $(1.96, +\infty)$, and for the P-value, the rejection region is $(0, 0.05)$. As shown in Table 6, the Z-score and P-value for both the full-edge randomized null model and the sign randomization null model fall within the rejection regions, leading to the rejection of the null hypothesis. This indicates that all six assortativity metrics are statistically significant and effectively reflect the differences in assortativity properties between the original network and null models.

Besides, because the Z-score is influenced by the network size, it is difficult to compare the Z-score across networks of different scales. Therefore, it is necessary to map the statistical results of different networks onto a unified scale. The Z-score from different networks is normalized to obtain the assortativity significance profile (ASP).

Table 6. The family of assortativity coefficients of original networks and null model networks.

Measures	Network	Original network	Sign randomized	Z_{f}	P_{f}	Full randomized	Z_{s}	P_{s}
$r^{+}(+,+)$	Alpha	–0.153	-0.128 ± 0.004	6.250	< 0.001	-0.168 ± 0.002	7.500	< 0.001
	OTC	–0.144	-0.125 ± 0.002	9.500	< 0.001	-0.165 ± 0.001	21.000	< 0.001
	Wiki	–0.356	-0.184 ± 0.004	43.000	< 0.001	-0.401 ± 0.003	15.000	< 0.001
	Slashdot	–0.190	-0.173 ± 0.001	17.000	< 0.001	-0.254 ± 0.001	64.000	< 0.001
$r^{-}(-,-)$	Alpha	–0.234	-0.126 ± 0.020	–5.400	< 0.001	-0.157 ± 0.015	–5.133	< 0.001
	OTC	–0.225	-0.129 ± 0.009	–10.667	< 0.001	-0.161 ± 0.008	–8.000	< 0.001
	Wiki	–0.393	-0.153 ± 0.012	–20.000	< 0.001	-0.387 ± 0.015	–0.400	0.001
	Slashdot	–0.264	-0.173 ± 0.002	–45.500	< 0.001	-0.254 ± 0.003	–3.333	0.001
$r^{+}(-,-)$	Alpha	–0.021	-0.122 ± 0.006	16.833	< 0.001	-0.158 ± 0.008	17.125	< 0.001
	OTC	–0.025	-0.128 ± 0.005	20.600	< 0.001	-0.163 ± 0.006	23.000	< 0.001
	Wiki	–0.075	-0.147 ± 0.005	14.400	< 0.001	-0.385 ± 0.008	38.750	< 0.001
	Slashdot	–0.002	-0.137 ± 0.002	67.500	< 0.001	-0.253 ± 0.003	83.667	< 0.001
$r^{+}(+,-)$	Alpha	–0.228	-0.236 ± 0.002	4.000	< 0.001	-0.255 ± 0.003	9.000	< 0.001
	OTC	–0.169	-0.148 ± 0.002	–10.500	< 0.001	-0.197 ± 0.002	14.000	< 0.001
	Wiki	–0.479	-0.386 ± 0.002	–46.500	< 0.001	-0.503 ± 0.002	12.000	< 0.001
	Slashdot	–0.180	-0.209 ± 0.001	29.000	< 0.001	-0.269 ± 0.001	89.000	< 0.001
$r^{-}(+,+)$	Alpha	–0.241	-0.132 ± 0.017	–6.412	< 0.001	-0.166 ± 0.015	–5.000	< 0.001
	OTC	–0.209	-0.131 ± 0.009	–8.667	< 0.001	-0.164 ± 0.007	–6.429	< 0.001
	Wiki	–0.281	-0.168 ± 0.012	–9.417	< 0.001	-0.405 ± 0.009	13.778	< 0.001
	Slashdot	–0.354	-0.221 ± 0.003	–44.333	< 0.001	-0.254 ± 0.003	–33.333	< 0.001
$r^{-}(+,-)$	Alpha	–0.349	-0.245 ± 0.008	–13.000	< 0.001	-0.247 ± 0.005	–20.400	< 0.001
	OTC	–0.263	-0.154 ± 0.003	–36.333	< 0.001	-0.191 ± 0.004	–18.000	< 0.001
	Wiki	–0.322	-0.349 ± 0.006	4.500	< 0.001	-0.497 ± 0.005	35.000	< 0.001
	Slashdot	–0.280	-0.261 ± 0.002	–9.500	< 0.001	-0.266 ± 0.002	–7.000	< 0.001

Table 7. The ASP-value of the family of assortativity coefficients.

Algorithm	Network	$Z^{+}(+,+)$	$Z^{-}(+,+)$	$Z^{+}(-,-)$	$Z^{-}(-,-)$	$Z^{+}(+,-)$	$Z^{-}(+,-)$
Full	Alpha	0.131	–0.138	0.228	–0.106	0.072	–0.325
	OTC	0.200	–0.186	0.278	–0.209	–0.188	–0.908
	Wiki	0.903	–0.202	0.195	–0.391	–0.831	0.113
	Slashdot	0.357	–0.952	0.913	–0.890	0.518	–0.238
Sign	Alpha	0.108	–0.135	0.177	–0.509	0.099	–0.455
	OTC	0.303	–0.174	0.238	–0.794	0.153	–0.401
	Wiki	0.216	–0.373	0.401	–0.040	0.131	0.780
	Slashdot	0.922	–0.901	0.866	–0.331	0.974	–0.156

The experiment results of ASP-value are presented in Table 7. First, it can be observed that for the Alpha network, the ASP value of $Z^{+}(+,+)$ is positive both in the full-edge randomized null model and the sign randomization null model. The ASP value of $Z^{+}(-,-)$ is also positive, similar to $Z^{+}(+,+)$. It indicates that when nodes are connected by positive edges, the degree mixing pattern among nodes with the same sign is generally assortative. Conversely, the ASP value

of $Z^{-}(+,+)$ is negative, and the ASP value of $Z^{-}(-,-)$ is also negative. This means that when nodes are connected by negative edges, the degree mixing pattern among nodes with the same sign is generally disassortative. Similar results are represented in the other three networks (OTC, Wiki, and Slashdot). However, the ASP values for $Z^{+}(+,-)$ and $Z^{-}(+,-)$ are mixed, with both positive and negative values, showing no clear consistency. As a result, it is difficult to determine the assortativity of these two mixing patterns.

3.4. Embeddedness

The embeddedness is measured by the embeddedness level statistic, and the experiment uses the Epinions empirical signed network dataset. The SNSAlib operates as follows: First, the original network is divided into positive and negative subnetworks, and the null model algorithm suite of SNSAlib is invoked to generate five different null models. Then, the embeddedness level statistics of the original subnetworks and their five null models are calculated. To minimize the impact of randomization errors, the experiment is independently repeated 200 times. Finally, the mean values as final results reflect the variation trend of the fraction of positive edges with positive embeddedness level and negative embeddedness level.

The results are presented in Fig. 5. In Fig. 5(a), the results of the original signed network and its three null models (positive-edge randomized null model, negative-edge randomized null model, and positive- and negative-edges randomized null model) all exhibit an upward trend. It indicates a positive embeddedness feature, where the fraction of positive edges between node pairs increases as the positive embeddedness level rises. In particular, the difference in the extent of the upward trend, due to the randomization degree of the three null models, reflected the positive embeddedness level is closely related to the network's positive edge topology and the correlation between positive and negative edges. Besides, in Fig. 5(b), the results of the original signed network and its three null models (positive-edge randomized null model, negative-edge randomized null model, and positive- and negative-edges randomized null model) all show a downward trend. It indicates a negative embeddedness feature, where the fraction of positive edges between node pairs decreases as the negative embeddedness level increases. Similar to Fig. 5(a), the negative embeddedness level is also closely related to the evolution of the network's positive edge topology and the correlation between positive and negative edges. In addition, in both Figs. 5(a) and 5(b), because the sign randomized null model and the full-edge randomized null model almost completely disrupt the topology, the results are nearly parallel to the X-axis, which indicates there is no positive or negative embeddedness feature.

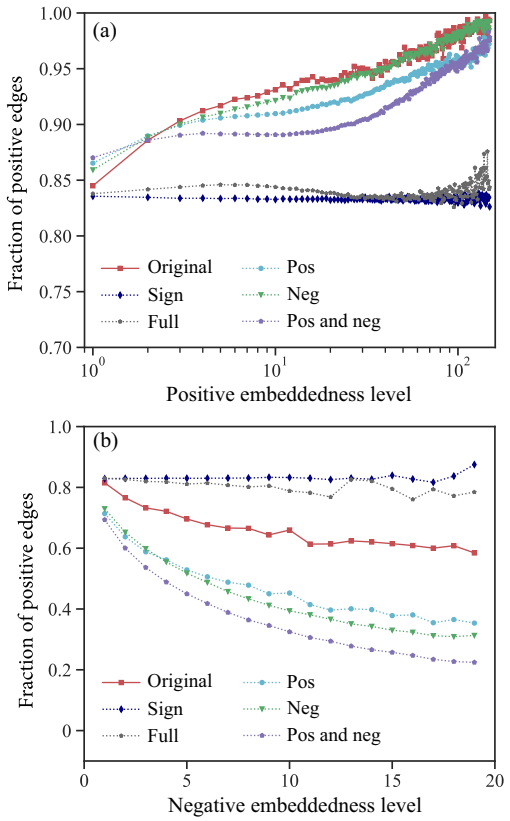


Fig. 5. The relationship between embededness and fraction of positive edges. (a) The result of positive embededness level, (b) the result of negative embededness level.

3.5. Community structure

The community structure of signed networks can be analyzed by the community detection algorithm, including finding and extracting community in signed networks (DECDS) and differential evolution community detection in signed networks (FECS) provided in SNSAlib.

Initially, the performance of community detection algorithms in signed networks is analyzed based on meso-level null models. The experiment uses the GGS empirical signed network dataset. The SNSAlib operates as follows: First, the null model algorithm suite of SNSAlib is invoked to generate the enhanced community structure null model and the diminished community structure null model. Then, the DECDS and FECS algorithms are employed to detect the community structure in the two null model networks. Finally, the results of the community detection are evaluated using normalization mutual information NMI. It should be noted that by increasing the ratio of positive edges within the community P_{+}, the community structure of signed networks is enhanced, and by increasing the ratio of negative edges within the community E_{-}, the community structure is diminished.

The results are shown in Fig. 6. In Fig. 6(a), as the ratio of positive edges in the community P_{+} increases, the NMI values of the two community detection algorithms both show an upward trend. In other words, after enhancing the community structure continuously, both algorithms can identify the com-

munity structure in the signed network more accurately. Conversely, in Fig. 6(b), as the ratio of negative edges in the community E_- increases, the NMI values of the two algorithms both show a downward trend. In other words, after diminishing the community structure continuously, the capabilities of the two algorithms decrease.

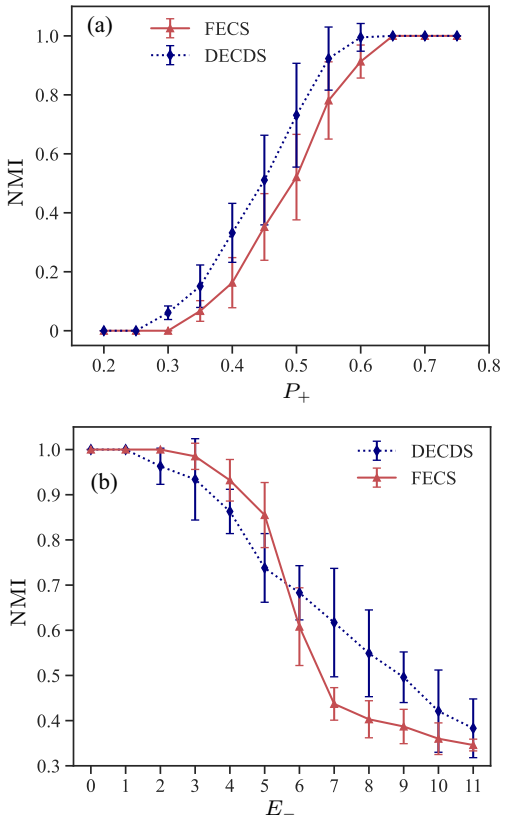


Fig. 6. The NMI of DECDS and FECS in enhanced community structure null model and diminished community structure null model. (a) The result of enhanced community structure null model, (b) the result of diminished community structure null model.

Then, the community structure feature of signed networks is analyzed based on five micro-level null models. The experiment uses five empirical signed network datasets: SPP, GGS, Wiring, Sampson, and Teams. The SNSAlib operates as follows: First, the null model algorithm suite of SNSAlib is invoked to generate five different null models. Then, the DECDS and FECS algorithms are employed to detect the community structure in both the original empirical networks and their corresponding null models. Finally, the results of the community detection are evaluated using signed modularity Q_s.

The experimental results are presented in Table 8. First, for the original SPP signed network dataset, there are higher Q_s values in both algorithms (DECDS and FECS), because it is relatively small, with only 10 nodes and 45 edges. Its actual community structure consists of just 2 communities, making the community structure quite evident. Besides, this clear structure means that except for the signed randomized null model, the other four edge randomized null models cannot change the community structure of SPP. Furthermore, for the other datasets (GGS, Wiring, Sampson, and Teams), the results show that the modularity values Q_s for the negative-edge randomized null model, positive- and negative-edge null model, full-edge randomized null model, and sign randomized null model, which all involve randomizing the negative topology, generally are lower than that of the positive-edge randomized null model. Both the DECDS and FECS algorithms generally lead to similar results. The finding suggests that the community structure of signed networks is more obscured by randomizing negative edges than positive ones, which makes community detection in these networks more challenging.

Table 8. The value of Q_s of original networks and null model networks in DECDS and FECS.

Algorithm	Network	Original	Pos	Neg	Pos and neg	Full	Sign
DECDS	SPP	0.455	0.455	0.455	0.455	0.455	0.183
	GGS	0.309	0.288	0.231	0.229	0.237	0.270
	Wring	0.301	0.221	0.196	0.157	0.325	0.146
	Sampson	0.281	0.194	0.166	0.141	0.186	0.107
	Teams	0.297	0.206	0.154	0.109	0.186	0.142
FECS	SPP	0.455	0.455	0.455	0.455	0.455	0.233
	GGS	0.289	0.278	0.228	0.231	0.235	0.256
	Wring	0.292	0.231	0.206	0.177	0.223	0.186
	Sampson	0.241	0.174	0.196	0.101	0.126	0.112
	Teams	0.307	0.214	0.254	0.139	0.118	0.132

4. Conclusion

In summary, we have developed the SNSAlib to establish a systematic framework for micro- and meso-structure analysis of signed networks. The algorithm library offers the easy-to-use various interfaces of null models and statistic functions. It supports multiple formats of the signed network dataset so that users can easily call the appropriate algorithm to analyze the empirical signed network structure directly. Furthermore, the evaluation results across different statistic features also confirm the reliability of SNSAlib for micro and meso scale analysis of signed networks and yield valuable insights.

The SNSAlib can analyze the specific type of network that incorporates both positive and negative edges,

which effectively addresses the application limitation of signed networks. At present, it has been used in several practical scenarios. For example, statics immunization strategies of disinformation,[49] sign prediction of network embedding,[50,51] link prediction based on motifs,[52] etc. However, our algorithm library only focuses on static structure analysis and does not explore the dynamic process of signed networks, which is also an important aspect of signed network analysis. In the future, the goal of improvement should be to introduce more dynamic models of signed networks, such as network structure evolution, information propagation, and game dynamics. The integration of static and dynamic perspectives allows for a thorough analysis of signed networks, thereby deepening our understanding of the vital role that opposition elements play in complex systems.

Program availability

The source code is openly available in the Science Data Bank at https://doi.org/10.57760/sciencedb.j00113.00178 and on Github at https://github.com/py-LIJI/signed-network-null-model.

Acknowledgements

Project supported by the National Natural Science Foundation of China (Grant Nos. 72371031, 62173065, and 62476045) and Fundamental Research Funds for the Central Universities (Grant No. 124330008).

References

[1] Szell M, Lambiotte R and Thurner S 2010 *Proc. Natl. Acad. Sci. USA* **107** 13636
[2] Heider F 1946 *The Journal of Psychology* **21** 107
[3] Cartwright D and Harary F 1956 *Psychological Review* **63** 277
[4] Newman M E J 2018 *Networks: An Introduction* 2nd Edn. (Oxford: Oxford University Press)
[5] Girdhar N and Bharadwaj K K 2016 *International Conference on Advances in Computing and Data Sciences* **721** 326
[6] Perfetto L, Briganti L, Calderone A, Perpetuini C A, Iannuccelli M, Langone F, Licata L, Marinkovic M, Mattioni A, Pavlidou T, Peluso D, Petrilli L L, Pirrò S, Posca D, Santonico E, Silvestri A, Spada F, Castagnoli L and Cesareni G 2016 *Nucleic Acids Research* **44** D548
[7] Lin W Y and Li B C 2024 *IEEE Transactions on Neural Networks and Learning Systems* **35** 4580
[8] Seo C, Jeong K J, Lim S and Shin W Y 2024 *IEEE Transactions on Neural Networks and Learning Systems* **35** 4729
[9] Wang H W, Zhang F Z, Hou M, Xie X, Guo M Y and Liu Q 2017 *Proceedings of the Eleventh ACM International Conference On Web Search and Data Mining (WSDM'18). ACM, New York* 592
[10] Li L B, Zeng A, Fan Y and Di Z R 2022 *Chaos* **32** 083101
[11] Ma Z W, Sun L, Ding Z G, Huang Y Z and Hu Z L 2024 *Chin. Phys. B* **33** 028902
[12] Lee W C, Lee Y C, Lee D W and Kim S K 2021 *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval* 143
[13] Ciotti V, Bianconi G, Capocci A, Colaiori F and Panzarasa P 2015 *Physica A* **422** 25
[14] José B M, Efrain C L and Rodrigo H Q 2022 *PLoS One* **17** e0278647
[15] Cheng H, He C, Liu H, Liu X, Yu P and Chen Q M 2024 *IEEE Transactions on Network Science and Engineering* **11** 1642
[16] Kirkley A, Cantwell G T and Newman M E J 2019 *Phys. Rev. E* **99** 012320
[17] Tian Y and Lambiotte R 2024 *Siam Journal On Applied Dynamical Systems* **23** 50
[18] Colizza V, Flammini A, Serrano M and Vespignani A 2006 *Nat. Phys.* **2** 110
[19] Davis J A 1977 *Social Networks* **20** 27
[20] Gjoka M, Kurant M and Markopoulou A 2013 *2013 Proceedings IEEE INFOCOM* 1968
[21] Li A W, Xiao J and Xu X K 2021 *Chin. Phys. B* **30** 038901
[22] Hao B J and István A K 2024 *Sci. Adv.* **10** eadj0104
[23] Morel-Balbi S and Peixoto P T 2020 *Phys. Rev. E* **102** 032306
[24] Paul S and Chen Y G 2022 *Sankhya-Series A-Mathematical Statistics and Probability* **84** 163
[25] Leskovec J and Krevl A 2014 *SNAP Datasets*: Stanford Large Network Dataset Collection
[26] Kropivnik S and Mrvar A 1996 *Developments in Statistics and Methodology* **12** 209
[27] Read K E 1954 *Journal of Anthropological Research* **10** 1
[28] Kronenfeld D B 2005 *Structural Models in Anthropology*, in *Encyclopedia of Social Measurement*, ed. by Kimberly Kempf-Leonard (New York: Elsevier) p. 705
[29] Homans G C 1951 *The American Journal of Psychology* **64** 463
[30] Breiger R L, Boorman S A and Arabie P 1975 *Journal of Mathematical Psychology* **12** 328
[31] Bassoul R, Moreno J L, Jennings H H, Criswell J H, Katz L, Blake R R, Mouton J S, Bonney M E, Northway M L and Loomis C P 1961 *Revue Franaise De Sociologie* **2** 331
[32] Dohleman B S 2006 *Exploratory Social Network Analysis with Pajek* (New York: Cambridge University Press) **71** 605
[33] Kumar S, Spezzano F, Subrahmanian V S and Faloutsos C 2016 *2016 IEEE 16th International Conference on Data Mining (ICDM)* 221
[34] West R, Paskov H S, Leskovec J and Potts C 2014 *Transactions of the Association for Computational Linguistics* **2** 297
[35] Leskovec J, Huttenlocher D and Kleinberg J 2010 *Conference on Human Factors in Computing Systems* (Atlanta, Georgia, USA) 1361
[36] Cui W K, Shang K K, Zhang Y J, Xiao J and X X K 2018 *Euro. Phys. J. B* **91** 1
[37] Kunegis J, Lommatzsch A and Bauckhage C 2009 *Proceedings of the 18th international conference on World wide web* (ACM Press) 741–750
[38] Yao L, Wang L N, Pan L and Yao K 2016 *Procedia Computer Science* **83** 82
[39] Vázquez A and Vespignani A 2001 *Phys. Rev. Lett.* **87** 258701
[40] Newman M E J 2002 *Phys. Rev. Lett.* **89** 89
[41] Li A W, Xiao J and Xu X K 2020 *IEEE Transactions on Computational Social Systems* **7** 1460
[42] Jia G B, Cai Z X, Musolesi M, Wang Y, Tennant D A, Weber R J M, Heath J K and He S 2012 *Revised Selected Papers of the International Conference on Learning & Intelligent Optimization* (Berlin, Heidelberg: Springer) pp. 71–85
[43] Yang B, Cheung W and Liu J M 2007 *IEEE Transactions on Knowledge & Data Engineering* **19** 1333
[44] Apollonio N, Franciosa P G and Santoni D 2022 *Scientific Reports* **12** 9757
[45] Waldorp L and Haslbeck J 2024 *Multivariate Behavioral Research* **2024** 738
[46] Foster J G, Foster D V, Grassberger P and Paczuski M 2010 *Proc. Natl. Acad. Sci. USA* **107** 10815
[47] He X C, Zhang R C and Zhu B 2023 *Complexity* **2023** 8767131
[48] Newman M E J, Cantwell G T and Young J G 2020 *Phys. Rev. E* **101** 042304
[49] Li A W, Xu X K and Fan Y 2022 *Chaos, Solitons and Fractals* **162**112489
[50] Liu S Y, Xiao J and Xu X K 2020 *Physica A* **593** 126966
[51] Liu S Y, Xiao J and Xu X K 2020 *Information Sciences* **541** 316
[52] Liu S Y, Xiao J and Xu X K 2019 *IEEE Transactions on Network Science and Engineering* **7** 1724