Proceedings of the 4th International Conference on Applied Physics and Mathematical Modeling
DOI: 10.54254/2753-8818/2026.36468

Spectral Graph Methods for Community Detection in Complex Networks

Ji Li

School of Mathematical and Physical Sciences, University of Sheffield, Sheffield, UK
jli391@sheffield.ac.uk

Abstract. Community detection is a key problem in complex-network analysis: densely connected groups may correspond to social circles, scientific fields, biological modules, or functional subsystems. This review considers how spectral graph methods translate a network into matrix form and then use eigenvalues and eigenvectors to uncover this structure. After introducing complex networks, graph matrices, community structure, and evaluation criteria, it develops the main ideas behind unnormalized and normalized graph Laplacians, the Fiedler vector, spectral embedding, normalized cut, and modularity-based eigenvector methods. Evidence from benchmark and real-world studies is then used to compare these classical techniques with regularized, non-backtracking, local, overlapping, neural-embedding, and randomized multi-layer extensions. Spectral methods remain competitive and mathematically interpretable. Their performance, however, depends on sparsity, degree heterogeneity, community overlap, network scale, and the choice of evaluation criteria. Future work is likely to combine sparse linear algebra and randomized eigensolvers with dynamic, multi-layer, and interpretable graph-learning models. By connecting core linear algebra with practical structure discovery, the review clarifies both the continuing value and the limits of the spectral perspective.

Keywords: spectral clustering, community detection, graph Laplacian, complex networks, spectral embedding

1. Introduction

1.1. Networked systems and the importance of community structure

Complex networks represent entities as nodes and their relationships as edges. This provides a common language for describing social media users and their friendships, scientific papers and their citations, proteins and their interactions, cities and their transportation networks, or routers and their Internet connections. This perspective is important because the collective organization of a system often cannot be reconstructed from a list of isolated entities alone. Community structure lies between individual nodes and the network as a whole, providing a mesoscale description. In the conventional sense, nodes within a community are more densely connected to one another than to the rest of the network. Depending on the application, these groups may represent social circles, research topics, protein modules, or regions through which information tends to flow [1].

DOI: 10.54254/2753-8818/2026.36468

Although useful, this density-based definition is not universal. Communities may instead be defined through flow, directed edges, attributes, hierarchy, or overlap, depending on the question being asked. Available labels do not necessarily provide a definitive ground truth because they may reflect administrative classifications or node attributes rather than structural communities. Accordingly, the literature cautions against assuming that every real network has a single correct partition [2]. The practical challenge, then, is to clarify what kind of structure is being sought, select methods consistent with that hypothesis, and test whether the resulting clusters are stable and useful. Once a network becomes too large for visual inspection, matrix-based methods provide a reproducible framework for analyzing its structure at scale.

1.2. Research progress and competing method families

Several major method families are commonly used for community detection. Spectral clustering treats informative eigenvectors of a graph operator as coordinates for partitioning nodes. Modularity methods compare the observed number of within-community edges with the number expected under a null model. Newman showed that this optimization can be formulated using a modularity matrix [3]. Label propagation repeatedly spreads local labels until a stable assignment emerges. It is attractive because its computational complexity is close to linear, although random node-update orders and label-selection rules can reduce the stability of the final partition [4]. Probabilistic block models take a different view: communities are latent groups that generate edges. Recent spectral analysis under extensions of these models provides explicit misclustering bounds and recovery conditions for networks with higher-order structure [5].

Within the spectral family, the choice of graph operator has substantive consequences rather than being a merely technical decision. The graph Laplacian supports both normalized and unnormalized embeddings [6], while normalized cut balances separation against the overall connectivity of the resulting groups [7]. Degree-corrected spectral clustering can improve community recovery under stated spectral conditions [8]. In low-mean-degree inhomogeneous random graphs, moreover, leading non-backtracking eigenvalues can track population-level information even when the adjacency spectrum is distorted by sparsity [9]. Benchmark rankings also vary with degree distribution, community size, mixing parameter, and evaluation metric [10, 11]. Recent work has broadened the range of spectral approaches: LEMON begins with local seed nodes [12], regularized mixed-membership spectral models accommodate overlapping communities [13], neural network embeddings may preserve geometric structures associated with the normalized Laplacian [14], and stochastic projection enables the analysis of multilayer networks containing millions of nodes [15]. These approaches are related but not interchangeable because they rely on different assumptions and involve distinct trade-offs.

1.3. Aim, scope, and paper organization

This review asks how adjacency, degree, Laplacian, modularity, and related operators turn community detection into eigenvalue, embedding, and clustering problems. The comparison considers matrix construction, scale, sparsity, direction, overlap, the required community count, post-processing, evaluation metrics, and theoretical setting. Each reported advantage is evaluated in relation to the data-generating process, parameter range, and ground-truth convention used in the corresponding study. Section 2 introduces the necessary graph matrices and evaluation concepts; Section 3 develops the main spectral pipelines and sparse-network corrections. Benchmark, local,

DOI: 10.54254/2753-8818/2026.36468

overlapping, neural, and multi-layer studies are compared in Section 4. Section 5 then considers current limitations and future directions, before Section 6 concludes.

2. Basic concepts of complex networks, community structure, and graph matrices

2.1. Networks and communities

A graph contains a set of nodes and a set of edges. Its order is the number of nodes, while its size is the number of edges. An undirected edge represents a symmetric relation. A directed edge has an origin and a destination. A weight can express frequency, capacity, similarity, or another interaction strength. A path is a sequence of adjacent nodes, and a connected component is a maximal set joined by paths. The degree of a node counts incident edges, or sums their weights in a weighted graph. Real networks are commonly sparse because the number of observed edges is far smaller than the number of possible undirected edges.

From an operational perspective, a community is a group of nodes that are relatively densely connected internally and sparsely connected to the rest of the network [1]. This definition leaves important choices open. Directed links can create source, sink, or flow communities, while weighted links make connection strength relevant. Nodes may also belong to multiple groups, and nested groups may exist at several scales. A defensible analysis must therefore connect the chosen definition to the substantive question rather than treating a partition as an intrinsic label [2].

2.2. Matrix representation and spectral quantities

For an undirected unweighted graph, the entry A_{ij} equals 1 when nodes i and j are connected and 0 otherwise. In a weighted graph, A_{ij} stores the edge weight. Each diagonal entry of the degree matrix D gives the sum of the edge values associated with the corresponding node. Together, A and D define the unnormalized graph Laplacian and its two common normalized forms. Each form treats node degree differently. Standard notation uses I, x, and λ. Here, I denotes the identity matrix, x denotes an eigenvector, and λ denotes the corresponding eigenvalue.

For an undirected graph, the unnormalized and symmetric normalized Laplacians are symmetric. Their eigenvalues are real and non-negative. The multiplicity of the eigenvalue 0 equals the number of connected components. In a connected graph, the eigenvector associated with the second-smallest eigenvalue is called the Fiedler vector. It captures the lowest-frequency non-trivial direction and often separates two weakly connected regions. Several such eigenvectors form a low-dimensional embedding for multiple communities [6].

Normalization matters. It reduces the influence of large or highly connected communities. However, the choice is not neutral. Different relaxation conditions can still produce different partitions [7].

2.3. Data and evaluation concepts

Relational data may come from friendships, citations, co-authorship, communication, hyperlinks, or molecular interactions. Internal measures judge a partition from network structure: modularity measures excess within-group connection relative to a null model, while conductance and normalized cut penalize weakly separated groups. External measures compare a partition with labels; Normalized Mutual Information (NMI) and Adjusted Rand Index (ARI) are common examples. Community balance, runtime, memory, and stability across random seeds are also

DOI: 10.54254/2753-8818/2026.36468

relevant. No single score covers structural quality, substantive meaning, and computational feasibility.

In practice, the chosen metric can change the ranking. A partition with high modularity may agree poorly with planted labels, while different versions of NMI react differently when methods return unequal numbers or sizes of communities. Conductance focuses on the boundary, but may prefer one small, well-isolated set. Ordinary partition metrics are also unsuitable for overlapping outputs unless they are replaced by explicitly overlapping variants. Evaluation criteria should therefore be specified before the results are inspected. If conclusions change across defensible metrics, that sensitivity is evidence about the underlying problem definition—not an inconvenience to be concealed.

3. Spectral graph methods for community detection

3.1. The standard spectral clustering pipeline

A standard workflow constructs A and D before selecting the unnormalized Laplacian, the symmetric normalized Laplacian, the random-walk normalized Laplacian, a normalized adjacency matrix, or another operator. Next, k informative eigenvectors are arranged as columns of a matrix, so each node is represented by one row in a lower-dimensional space. Normalized spectral clustering may then rescale each row to unit length before applying k-means [6]. Only after this continuous relaxation and post-clustering step is a discrete partition obtained.

Each stage can materially affect the resulting community partition. Graph construction decides which relationships exist; normalization sets the influence of degree; eigenvector selection determines the retained scale; and k-means adds sensitivity to initialization. The number k is often supplied in advance or estimated from an eigengap. When that gap is weak or noisy, however, it offers little guidance. Degree-corrected spectral clustering instead modifies the graph Laplacian to reduce degree effects. Under stated spectral conditions, this procedure can provide a good approximation to the target clustering [8].

3.2. Normalized cut, Fiedler partitioning, and modularity spectra

Fiedler partitioning divides nodes according to the second-smallest Laplacian eigenvector, either by its sign or by a threshold chosen to improve a cut objective. Multiple eigenvectors extend the representation beyond two groups. Shi and Malik's normalized cut divides the total edge weight crossing a partition by the total association on each side. Its generalized eigenvalue relaxation avoids the tendency of an unnormalized cut to isolate a small set; the final threshold, however, remains an approximation to a discrete problem [7].

The modularity matrix, by contrast, uses a different baseline. Its entries compare observed adjacency with the expected edge weight under a degree-preserving null model. A positive leading eigenvalue identifies a direction along which a split can increase modularity, after which recursive splitting can produce further communities [3]. Thus, normalized cut favors groups with a small boundary relative to volume, whereas modularity favors groups containing more internal edges than expected. Neither objective is neutral: normalized cut can favor balanced volumes, while modularity inherits the null model and scale limitations of its objective.

DOI: 10.54254/2753-8818/2026.36468

3.3. Sparse-network corrections and non-spectral comparators

In sparse graphs, however, the ordinary spectral picture becomes less reliable. Before the community signal is clearly visible, a few high-degree nodes—or even small local structures—may already dominate the eigenvectors. One response is degree-corrected spectral clustering, which modifies the Laplacian to reduce the influence of degree heterogeneity [8]. Another is the non-backtracking operator, which follows directed edges without immediately reversing direction. For weighted inhomogeneous random graphs with low mean degree, leading non-backtracking eigenvalues remain close to population eigenvalues while the rest are confined to a spectral bulk [9]. These advantages remain conditional on the assumed network model.

Viewed alongside non-spectral comparators, these trade-offs become clearer. Label propagation relies on inexpensive local-majority updates, although randomness in node-update order and label selection can reduce stability [4]. Direct modularity heuristics, meanwhile, avoid eigendecomposition but retain the limitations of the modularity objective. Random-walk and diffusion approaches lie between the two categories: their transition matrices are spectral objects, yet a local implementation may inspect only a small part of the graph. The choice, then, is not simply between spectral and non-spectral algorithms. It is a choice among global geometry, local speed, stability, and assumptions.

4. State-of-the-art studies of spectral community detection in complex networks

4.1. Benchmark evidence and method selection

Benchmarks make controlled comparison possible; at the same time, they shape what counts as success. The Lancichinetti-Fortunato-Radicchi (LFR) benchmark improves on equal-sized block models by introducing heterogeneous degree and community-size distributions and varying a mixing parameter. The LFR benchmark nevertheless represents only one controlled generative setting, so performance rankings obtained from it should not be treated as universal. Even so, LFR omits features such as realistic triangle patterns and deep hierarchy. Success on it is therefore evidence for a controlled regime, not proof of general real-world superiority.

A similar caution emerges from Brusco et al.'s simulation comparison of spectral clustering and walktrap. When the correct number of communities was supplied, clustering the normalized-Laplacian embedding generally recovered communities better; when that number had to be estimated, walktrap with a modularity criterion outperformed the eigengap heuristic [11]. Because the study focused on undirected psychometric networks and a defined simulation design, the resulting ranking is necessarily conditional. Read together, the two studies support a diagnostic decision process rather than a static ranking. A sparse, heterogeneous graph calls for inspection of degree and mixing; a large graph makes runtime and memory part of the result. Overlap or hierarchy, meanwhile, requires a benchmark that actually contains those features. High NMI under one generator may therefore indicate a close fit to that generator's assumptions, not the recovery of a universal community concept.

4.2. Local and overlapping spectral communities

Rather than embedding the whole network, LEMON builds a local spectral subspace from short random walks that begin at seed nodes. Within that subspace, it seeks a sparse vector concentrated around the target community. Because the search remains local, its computational cost can depend

DOI: 10.54254/2753-8818/2026.36468

more on community size than on the size of the full network; indeed, the reported experiments include graphs with millions of nodes [12]. The efficiency comes at a clear cost: supervision. Seeds are required; their quality affects recovery, and local sampling uses information about the likely target scale. For this reason, LEMON is well suited to finding a particular overlapping group, but it is not a direct substitute for a complete network partition.

Qing and Wang, by contrast, address overlap through a mixed-membership stochastic block model. Their simplex and cone variants use a regularized Laplacian to estimate per-node membership vectors, with consistency results under stated model conditions and tests on synthetic and empirical networks [13]. No seed set is required, and overlap forms part of the model. Still, the method assumes undirected, unweighted input and does not model degree heterogeneity. The comparison with LEMON reveals an important distinction in claims about scalability: fast local expansion around trusted seeds and a global mixed-membership model solve different problems.

4.3. Spectral embeddings, neural embeddings, and multi-layer scale

Neural embeddings are often presented as a modern replacement for classical spectral coordinates. The boundary, however, is less clear than that contrast suggests. Under stated planted-partition conditions, Kojaku et al. show that a shallow node2vec-type embedding aligns with eigenvectors of the symmetric normalized Laplacian [14]. Their experiments on sparse, degree-heterogeneous networks connect neural performance near the detectability limit to familiar spectral geometry. Even here, the equivalence concerns the representation rather than every stage of the pipeline. Random-walk sampling, embedding dimension, prior knowledge of k, and post-embedding k-means may all alter recovery, especially when community sizes are unequal.

Multi-layer networks present a different scaling problem: several edge types must be combined. Su et al. address it by sampling and randomly projecting a sum of squared layer adjacencies before clustering, thereby avoiding storage and eigendecomposition of a dense n by n matrix. Their experiments include YouTube with 3,223,589 nodes, seven layers, and 10,313,509 edges. On the reported hardware, the procedure therefore reaches data sizes inaccessible to a dense baseline [15]. At the same time, the analysis obtains comparable misclassification behavior under a multilayer stochastic block model with conditions including balanced communities and spectral separation. The result is best read as an important accuracy-efficiency trade-off, not as a claim that randomization is lossless for every multilayer network.

4.4. Cross-study comparison framework

A reproducible comparison should record, at a minimum, the network source; whether data are synthetic or real; nodes, edges, and layers; degree and community-size distributions; directed, weighted, attributed, or overlapping structure; operator and regularization; eigenvector selection; whether k is known; and post-embedding clustering. Results should combine NMI or ARI with modularity, conductance, runtime, memory, and variation across seeds. Theoretical guarantees, moreover, should be reported with their model assumptions. These common dimensions separate a method's mechanism from the setting in which it appeared successful.

DOI: 10.54254/2753-8818/2026.36468

5. Limitations and prospects

5.1. Computational cost and scale

At scale, computation itself becomes a central constraint. In its basic form, dense eigendecomposition requires cubic time in n and quadratic storage. For massive graphs, that is infeasible. Sparse Lanczos or related iterative methods compute only selected eigenvectors; nevertheless, convergence can slow when eigenvalues are poorly separated, and the embedding itself still stores n times k values. Global spectral algorithms may therefore become memory-bound before reaching their formal arithmetic limit. Sparse matrix storage, warm starts, sparsification, and distributed multiplication can help, provided that they preserve the operator's informative subspace.

Not every form of acceleration reduces the same burden. Local diffusion avoids processing the full graph when the question concerns one community [12]. Random sampling and projection, by contrast, reduce the multi-layer burden [15]. Yet the statistical problem changes along with the computation: local algorithms depend on seeds, whereas randomized algorithms introduce approximation and tuning dimensions. For that reason, acceleration should be evaluated through accuracy, stability, peak memory, and wall-clock time on the same hardware, with claims limited to the network models and parameter ranges tested.

5.2. Sparsity, degree heterogeneity, noise, and missing data

Sparsity is not only a computational concern; it is also a statistical one. When average degree is small, empirical matrices fluctuate strongly around their population counterparts. Hubs, leaves, or small dense motifs may localize eigenvectors, while degree variation may be mistaken for community structure. Degree-corrected spectral clustering can mitigate degree heterogeneity under stated assumptions [8]; non-backtracking spectra, meanwhile, can preserve leading population-level information in low-mean-degree random graphs [9]. Neither protection is universal. The former relies on a chosen degree-correction scheme, whereas the latter uses a different operator and is analyzed under random-graph assumptions.

Real observations introduce a different difficulty: edges may be missing or spurious, weights may be uncertain, and the sampling process itself may be biased. Any of these perturbations can alter a spectral gap and rotate the embedding. The risk is greatest when several eigenvalues are close. For that reason, robust practice should vary graph-construction thresholds, resample or perturb the network, and inspect degree-aware diagnostics. A single static partition, by itself, reveals little about whether the pattern persists; it may reflect stable structure or merely one fragile preprocessing decision.

5.3. Community number, overlap, dynamics, and multiple layers

Choosing k presents another unresolved problem, since many pipelines require it before eigendecomposition or k-means. An eigengap can suggest a value, but hierarchy, weak separation, and unequal group sizes often blur the choice. The difficulty does not end there: real nodes may belong to several communities, memberships may change over time, and network layers may contain either compatible or conflicting structure. Existing work addresses only parts of this problem. Regularized mixed-membership spectral models accommodate non-exclusive membership [13], while randomized multi-layer clustering makes large consensus calculations feasible [15].

DOI: 10.54254/2753-8818/2026.36468

Future methods should estimate uncertainty in k and membership, track change without imposing artificial temporal continuity, and allow shared and layer-specific structure to coexist.

5.4. Evaluation validity and reproducibility

Correct computation does not, by itself, guarantee valid evaluation. The objective and the application may still disagree. Modularity, for example, may reward structurally neat groups that have little substantive value, while metadata may describe roles or attributes rather than edge-based communities [2]. Recovery metrics have a precise generative meaning under a stochastic block model [5], but a real network rarely certifies that model. No single measure is therefore sufficient. NMI or ARI can be used where labels are defensible, conductance or modularity can assess internal structure, and runtime, memory, and stability capture practical use. Comparisons become more reproducible, moreover, when they include tests across random seeds, sensitivity analyses, transparent preprocessing, public code, and both synthetic and real data.

Nor does reproducibility follow from the algorithm name alone. Two studies using the same named method may nevertheless differ in their similarity graphs, Laplacian normalizations, eigensolver tolerances, eigenvector conventions, row normalizations, k-means restarts, or stopping rules. Those choices, together with the random seeds, should be published. Otherwise, an apparent improvement attributed to a new spectral operator may actually have entered later, during embedding or discretization.

5.5. Future research directions

Future progress, then, will require more than a larger eigensolver. Sparse and randomized linear algebra should work alongside robust operators, distributed computation, and explicit uncertainty. Dynamic and multilayer models, meanwhile, need scalable ways to distinguish persistent, transient, shared, and layer-specific communities. Nor need interpretability be abandoned: neural methods can learn sampling or representation choices, yet their relationship to normalized-Laplacian geometry [14] remains visible. Evaluation must become more diagnostic as well. Gains from the operator, embedding, and post-clustering stages should be separated, then tested across changes in generator, density, degree distribution, overlap, and labelling convention.

6. Conclusion

Taken together, the evidence reviewed here shows that spectral community detection is not a single algorithm but a sequence of explicit choices involving graph construction, operator selection, eigensolver design, embedding, and post-clustering. These choices determine which structural patterns are emphasized and therefore shape both the resulting partition and its interpretation. The comparison also shows that no method should be judged independently of network sparsity, degree heterogeneity, overlap, scale, and evaluation protocol. For practical applications, computational efficiency, stability, and sensitivity to preprocessing should be reported alongside structural accuracy. An operator is constructed, informative eigenvectors are extracted, nodes are embedded, and the resulting geometry is converted into groups. The graph Laplacian and normalized cut identify weak boundaries; the modularity matrix, by contrast, measures departure from a null model. In sparse or degree-heterogeneous regimes, regularization and non-backtracking operators can improve recovery, but only under particular conditions. No operator dominates across dense, sparse, overlapping, dynamic, and multi-layer settings. Each encodes a different view of scale, degree, and

DOI: 10.54254/2753-8818/2026.36468

membership. For that reason, benchmark results are most informative when the generator, evaluation metric, runtime, and meaning of ground truth are reported together. Recent work does not discard the linear-algebraic foundation; it extends it. Local diffusion reduces global cost, randomized projection enables very large multi-layer calculations, and neural embeddings may recover familiar spectral geometry. The next step is to combine these gains with uncertainty analysis and transparent evaluation. The spectral view remains valuable precisely because its assumptions can be inspected and connected to the structure an application is actually seeking.

References

[1] Diboune, A., Slimani, H., Nacer, H., & Beghdad Bey, K. (2024). A comprehensive survey on community detection methods and applications in complex information networks. *Social Network Analysis and Mining, 14*, Article 93. https://doi.org/10.1007/s13278-024-01246-5

[2] Peixoto, T. P. (2023). Descriptive vs. inferential community detection in networks: Pitfalls, myths and half-truths. *Cambridge University Press*. https://doi.org/10.1017/9781009118897

[3] Newman, M. E. J. (2006). Finding community structure in networks using the eigenvectors of matrices. *Physical Review E, 74*(3), Article 036104. https://doi.org/10.1103/PhysRevE.74.036104

[4] Zhang, S., Yu, S., E, X., Huo, R., & Sui, Z. (2022). Label propagation algorithm joint multilayer neighborhood overlap and historic label similarity for community detection. *IEEE Systems Journal, 16*(2), 2626-2634. https://doi.org/10.1109/JSYST.2021.3113826

[5] Paul, S., Milenkovic, O., & Chen, Y. (2023). Higher-order spectral clustering under superimposed stochastic block models. *Journal of Machine Learning Research, 24*(320), 1-58.

[6] Mondal, R., Ignatova, E., Walke, D., Broneske, D., Saake, G., & Heyer, R. (2024). Clustering graph data: The roadmap to spectral techniques. *Discover Artificial Intelligence, 4*, Article 7. https://doi.org/10.1007/s44163-024-00102-x

[7] Shi, J., & Malik, J. (2000). Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 22*(8), 888-905. https://doi.org/10.1109/34.868688

[8] Liu, F., Li, W., & Zhong, Y. (2022). A further study on the degree-corrected spectral clustering under spectral graph theory. *Symmetry, 14*(11), Article 2428. https://doi.org/10.3390/sym14112428

[9] Stephan, L., & Massoulié, L. (2022). Non-backtracking spectra of weighted inhomogeneous random graphs. *Mathematical Statistics and Learning, 5*(3-4), 201-271. https://doi.org/10.4171/MSL/34

[10] Gamermann, D., & Pellizzaro, J. A. (2022). An algorithm for network community structure determination by surprise. *Physica A: Statistical Mechanics and its Applications, 595*, Article 127063. https://doi.org/10.1016/j.physa.2022.127063

[11] Brusco, M. J., Steinley, D., & Watts, A. L. (2024). A comparison of spectral clustering and the walktrap algorithm for community detection in network psychometrics. *Psychological Methods, 29*(4), 704-722. https://doi.org/10.1037/met0000509

[12] Li, Y., He, K., Kloster, K., Bindel, D., & Hopcroft, J. (2018). Local spectral clustering for overlapping community detection. *ACM Transactions on Knowledge Discovery from Data, 12*(2), Article 17, 1-27. https://doi.org/10.1145/3106370

[13] Qing, H., & Wang, J. (2023). Regularized spectral clustering under the mixed membership stochastic block model. *Neurocomputing, 550*, Article 126490. https://doi.org/10.1016/j.neucom.2023.126490

[14] Kojaku, S., Radicchi, F., Ahn, Y.-Y., & Fortunato, S. (2024). Network community detection via neural embeddings. *Nature Communications, 15*, Article 9446. https://doi.org/10.1038/s41467-024-52355-w

[15] Su, W., Guo, X., Chang, X., & Yang, Y. (2025). Randomized spectral clustering for large-scale multi-layer networks. *Statistics and Computing, 35*(6), Article 190. https://doi.org/10.1007/s11222-025-10723-6