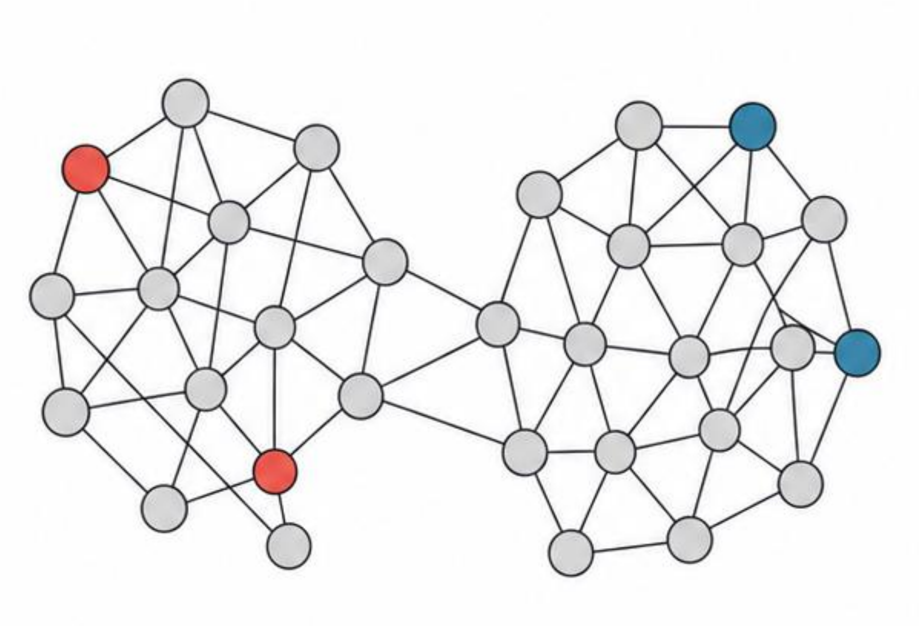
ESE 5291 · Graph Signal Processing and Graph Machine Learning



Community Detection and Spectral Clustering

September 15

Zhiyang Wang · Electrical & Systems Engineering · WashU

Class 7 · Community Structures Live in the Dominant Eigenspaces.

Three questions frame the lecture

01 **WHAT GENERATED THE GRAPH?**

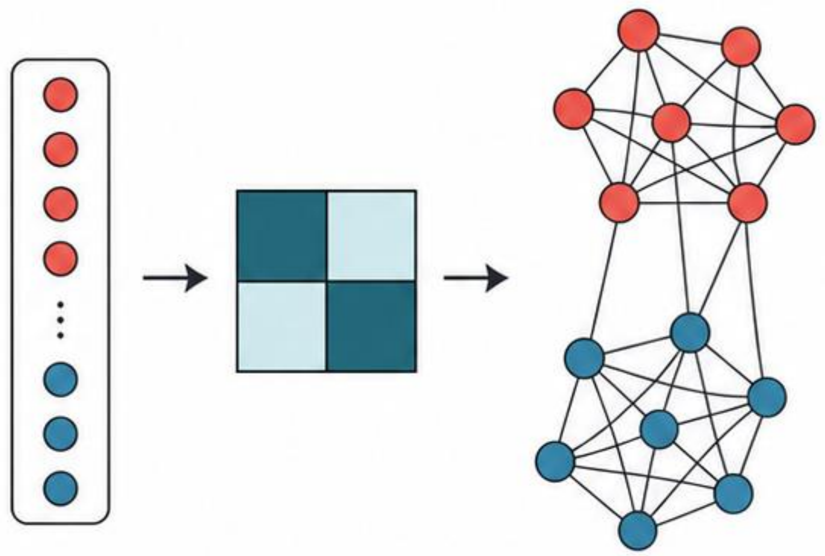
Use a Stochastic Block Model (SBM) to turn latent labels into edge probabilities.

02 **WHY DO EIGENVECTORS HELP?**

The informative eigenspace separates community signal from random edges.

03 **WHEN IS RECOVERY POSSIBLE?**

When the density gap between communities is larger than the random fluctuation — signal beats noise.



An SBM turns hidden labels into edge probabilities

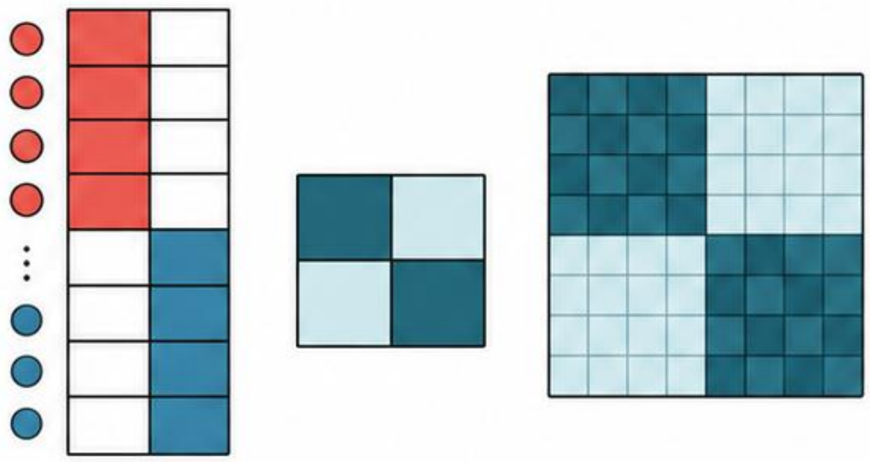
$$\mathbf{Y} \in \{0,1\}^{n\times C}, \qquad \mathbf{B} \in [0,1]^{C\times C}$$

Each row of $\boldsymbol{Y}$ is a one-hot community label;
$\boldsymbol{B}$ stores within- and between-group connection rates.

GENERATIVE MODEL

$$\mathbf{P} = \mathbf{YBY}^{\top}, \quad A_{ij} \sim Bernoulli\left(P_{ij}\right) \quad P_{ij} = [\mathbf{YBY}^{\top}]_{ij} = B_{c_i c_j}$$

Input: one adjacency matrix $\boldsymbol{A}$.
Output: an estimated label $\hat{y}_i$ for every node.



Community detection inverts a latent-variable graph model from one noisy sample.

The balanced two-block SBM uses only p and q

SAME COMMUNITY

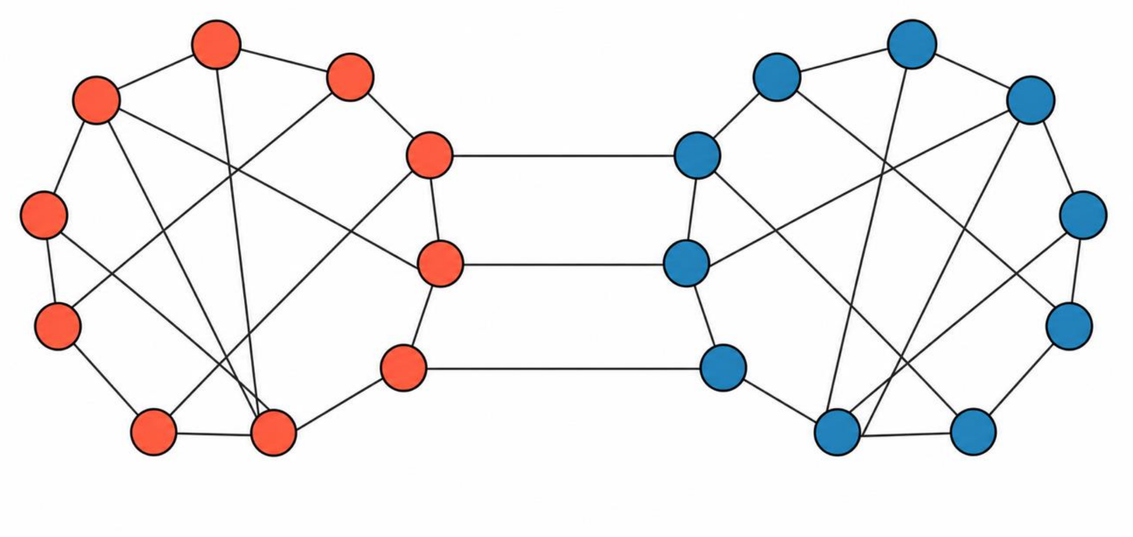
$$Pr(A_{ij} = 1) = p$$

Each community contains $n/2$ nodes.

DIFFERENT COMMUNITIES

$$Pr(A_{ij} = 1) = q$$

Assortative structure requires $p > q$.



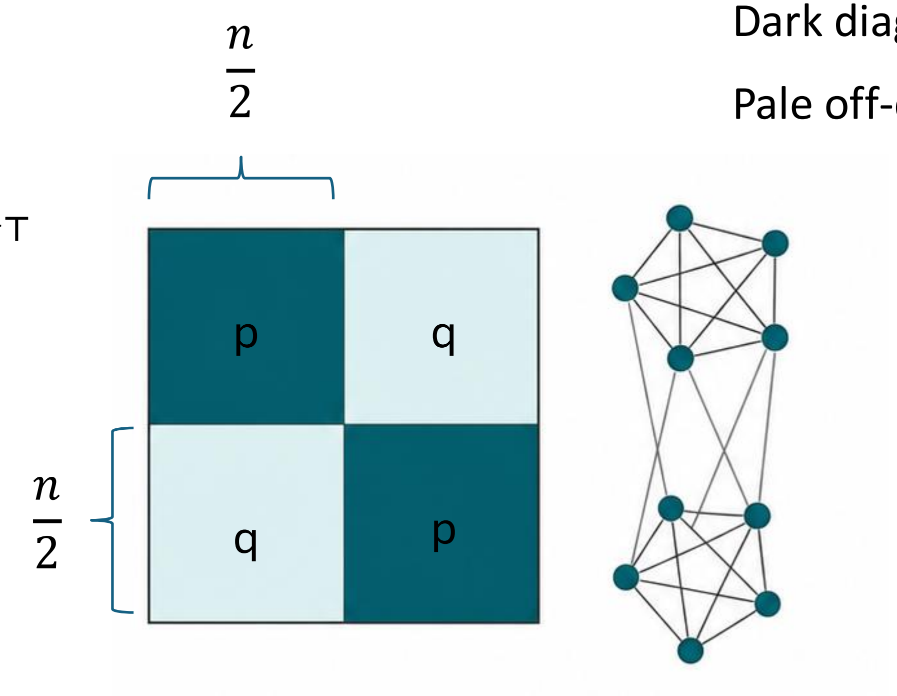
The gap $p - q$ is community signal, $p + q$ controls density and edge noise.

Expected adjacency reveals the block pattern

BLOCK MODEL

$$\mathbf{B} = \begin{pmatrix} p & q \\ q & p \end{pmatrix}$$

$$\mathbf{P} = \mathbb{E}[\mathbf{A}] = \mathbf{Y}\mathbf{B}\mathbf{Y}^\top$$



INTERPRETATION

Dark diagonals: within-group edges.

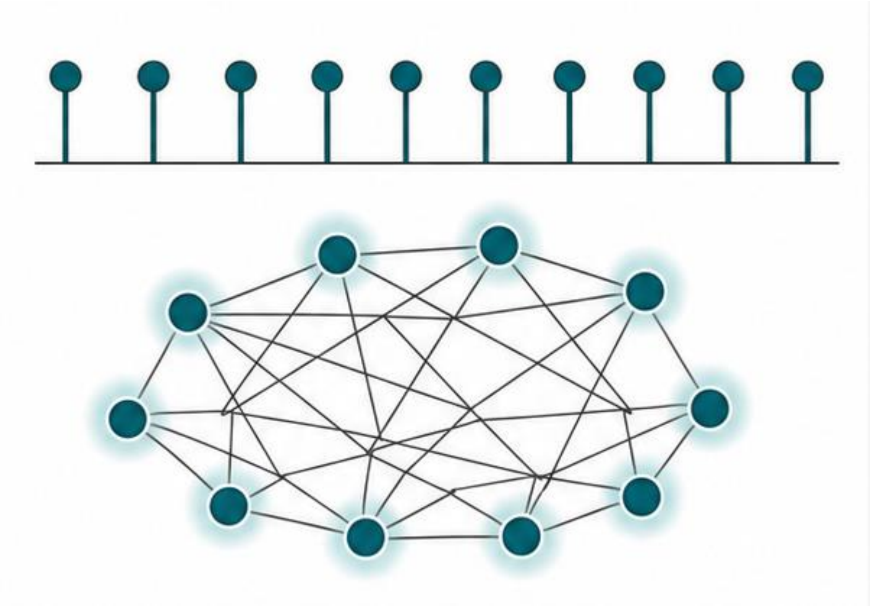
Pale off-diagonals: cross-group edges.

In expectation, every node in a community has the same connection profile.

The first eigenvector measures graph density

For a balanced two-block SBM, every row of $\boldsymbol{P}$ has the same sum. $\boldsymbol{P}\boldsymbol{v}_{\mathbf{1}} = \lambda_1 \boldsymbol{v}_{\mathbf{1}}$

$$\mathbf{v}_1 = \frac{1}{\sqrt{n}}\mathbf{1}, \qquad \lambda_1 = \frac{n}{2}(p+q)$$



The leading mode says how dense the graph is.

The second eigenvector reveals the partition

$$v_2 = \frac{1}{\sqrt{n}}[1, \dots, 1, -1, \dots, -1]^\top$$

Positive entries identify one block; negative entries identify the other.

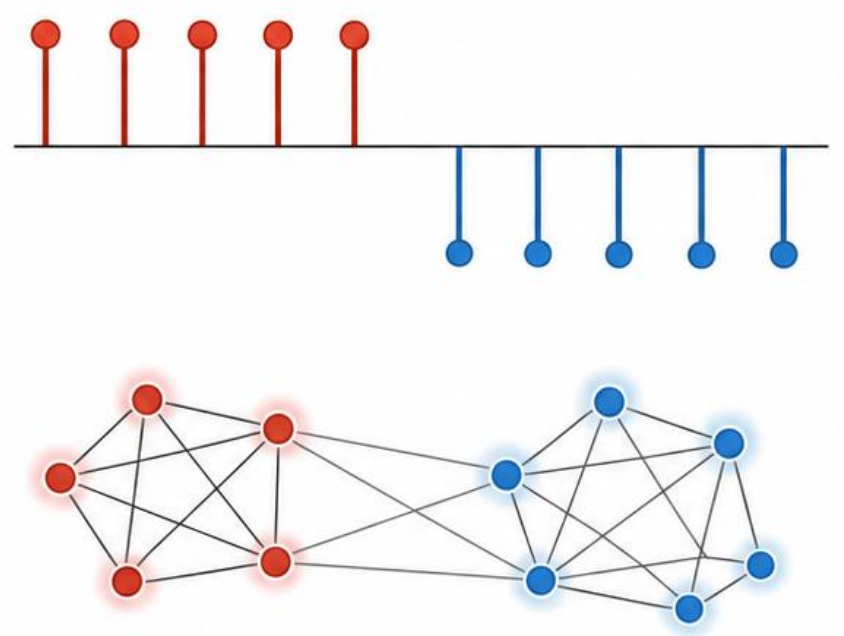
COMMUNITY EIGENVALUE

$$\lambda_2 = \frac{n}{2}(p - q)$$

A larger $|p - q|$ separates the informative mode more strongly from random fluctuations.

AND NOTHING ELSE — P IS EXACTLY RANK TWO

$$\mathbf{P} = \lambda_1 v_1 v_1^\top + \lambda_2 v_2 v_2^\top$$



For $C = 2$, community detection reduces to recovering the sign of v_2 -- spectral clustering.

Observed adjacency equals signal plus noise

SIGNAL

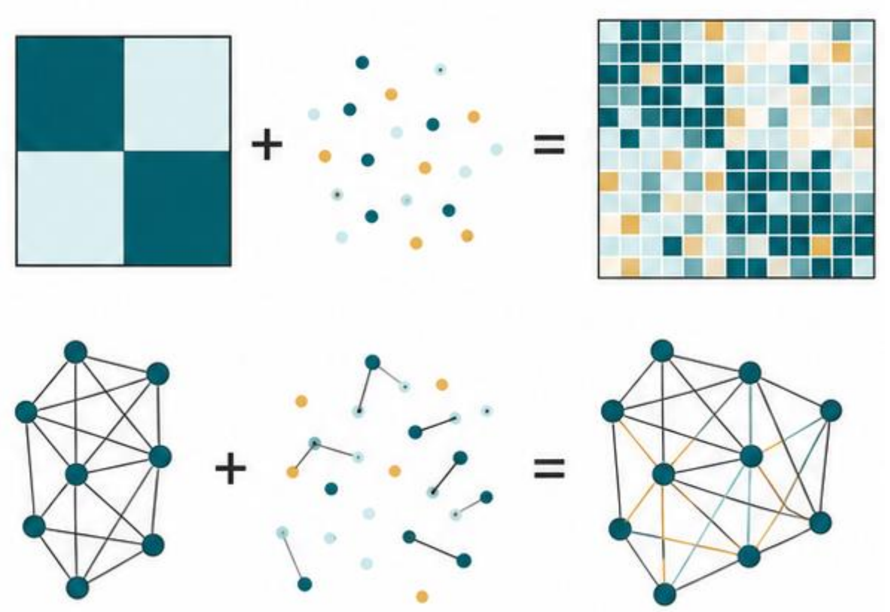
$$\mathbf{P} = \mathbb{E}[\mathbf{A}] = \mathbf{Y}\mathbf{B}\mathbf{Y}^\top$$

Low rank; aligned with the partition.

NOISE

$$\mathbf{E} = \mathbf{A} - \mathbf{P}, \qquad \mathbb{E}[\mathbf{E}] = 0$$

Random edges rotate eigenvectors.



Spectral clustering succeeds when the informative eigenspace survives the perturbation E.

A spectral gap stabilizes the community subspace

POPULATION SEPARATION

Signal eigenvalues: λ_1, λ_2.

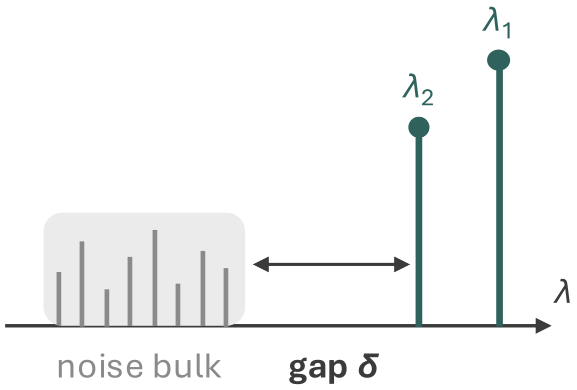
Noise occupies a lower spectral bulk.

PERTURBATION VIEW

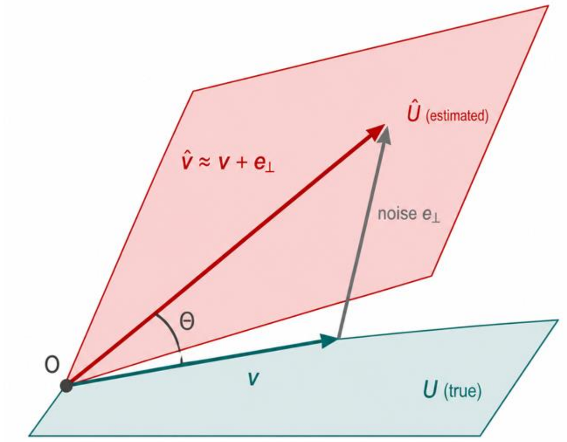
If $\|\boldsymbol{E}\|$ is below the eigengap,

the empirical eigenspace stays close (Davis–Kahan).

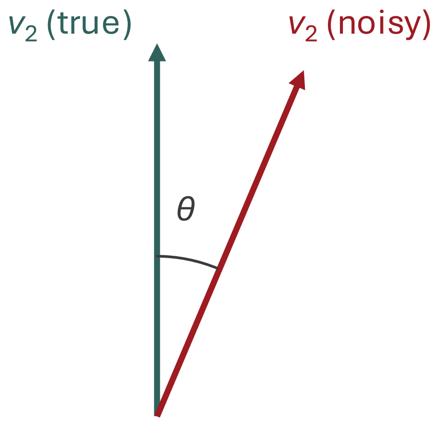
Eigenvalue spectrum



Signal subspace vs. noise



Subspace tilt under noise



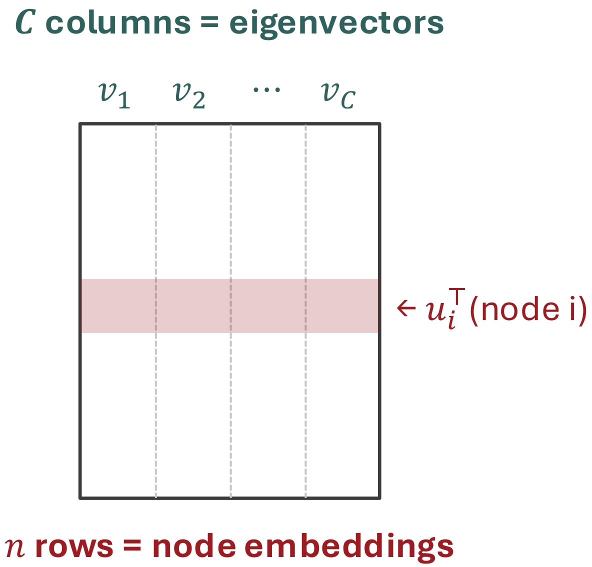
$sin\theta \leq \|E\|/\delta$

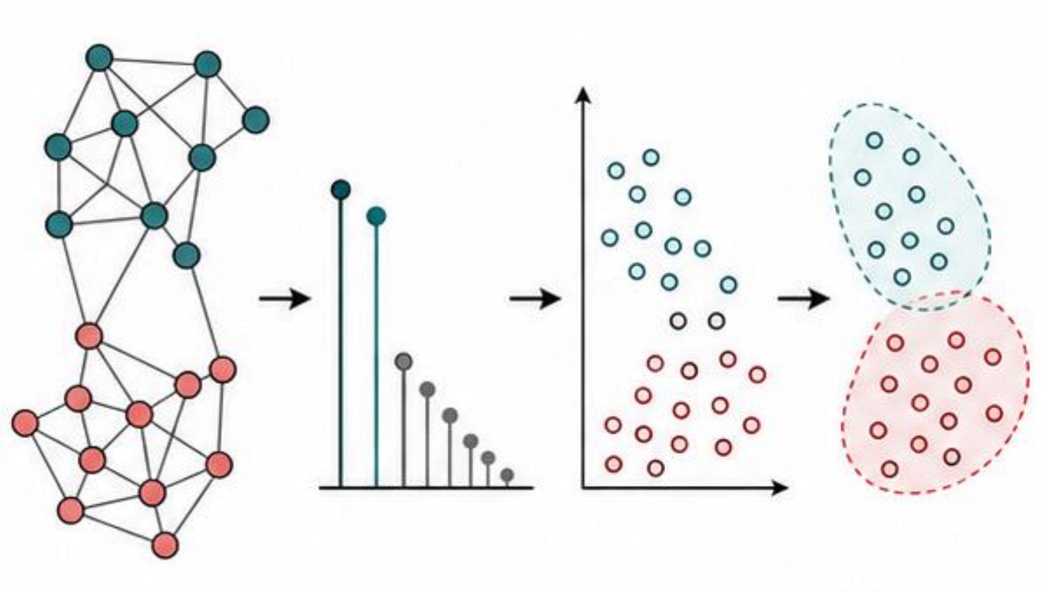
An eigenvalue matters through its separation from competing modes.

Spectral clustering turns eigenvectors into labels

$$\mathbf{A} = \mathbf{V}\Lambda\mathbf{V}^\top \rightarrow \mathbf{U} = [v_1, \dots, v_C] \rightarrow \text{cluster the rows } u_i$$

Choose an operator → extract C informative eigenvectors → embed every node by a row of $\boldsymbol{U}$ → then cluster the row vectors.



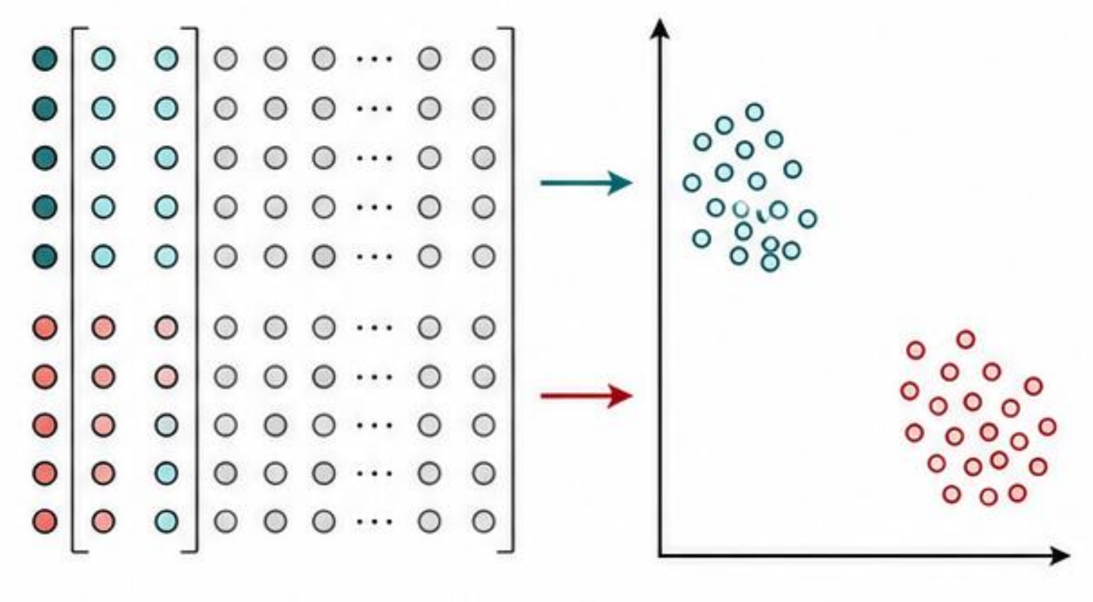


The spectral step converts a graph partition into ordinary clustering in $\mathbb{R}^C$.

Rows of U are node embeddings

$\boldsymbol{U} \in \mathbb{R}^{\mathrm{nxc}}$; row $\boldsymbol{u}_{\mathrm{i}}$ contains node i's coordinates in the informative subspace.

$$\mathbf{u}_i = [v_1(i), \dots, v_C(i)] \in \mathbb{R}^C$$

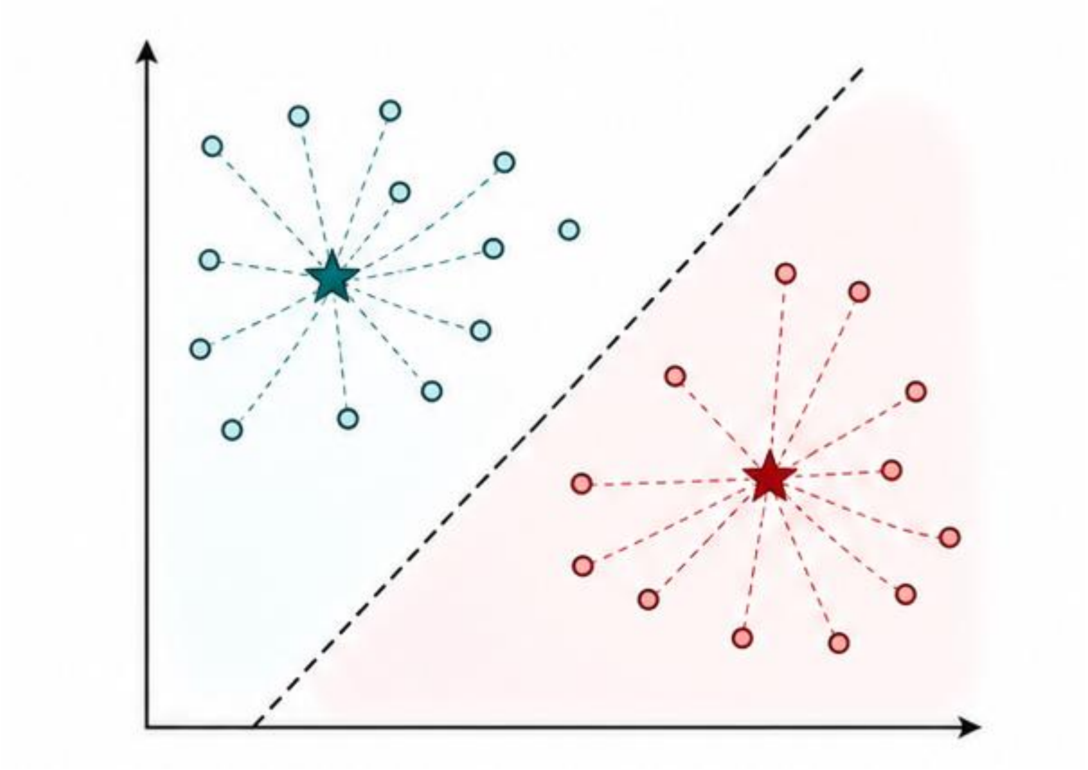


Similar global connection profiles become nearby points in embedding space.

K-means turns embeddings into community labels

Given node embeddings $\{u_i\}$, choose C centroids and assign each node to its nearest centroid.

$$min_{\mu_1,\dots,\mu_C} \sum_i min_c \|u_i - \mu_c\|^2$$



Output: one cluster index $\hat{y}_i$ per node; cluster names remain arbitrary.

Operator choice controls sensitivity to node degree

ADJACENCY

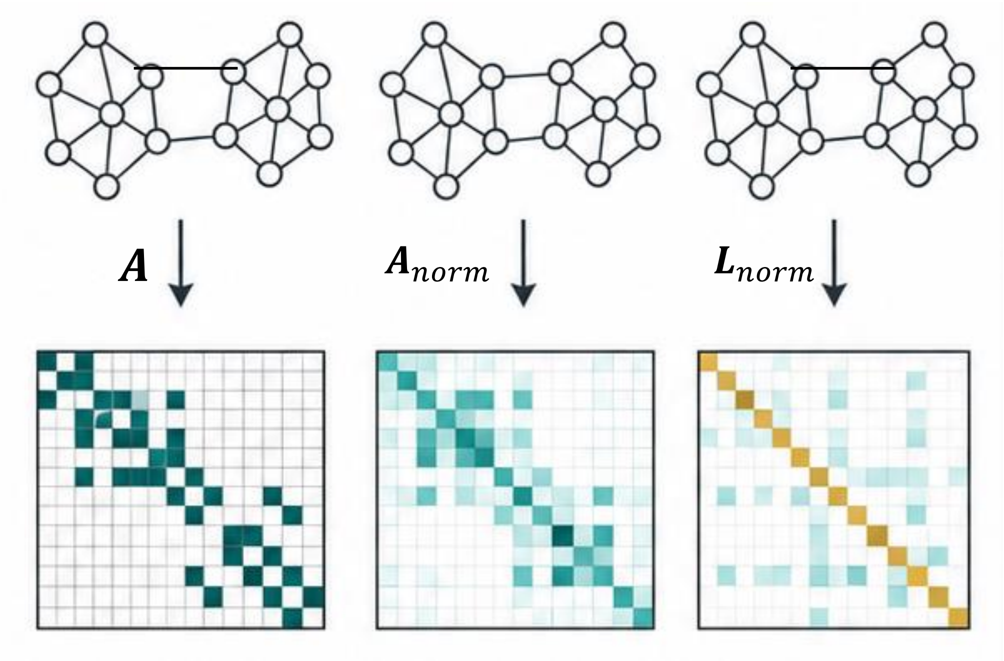
Use top eigenvectors of $\boldsymbol{A}$.

Simple, but hubs can dominate.

NORMALIZED OR LAPLACIAN

$\boldsymbol{A}_{norm} = \mathbf{D}^{-1/2}\mathbf{A}\mathbf{D}^{-1/2}$ or $\boldsymbol{L}_{norm} = \boldsymbol{I} - \boldsymbol{D}^{-1/2}\boldsymbol{A}\boldsymbol{D}^{-1/2}$

Balances unequal degrees.



Normalize when degree variation is nuisance rather than community signal.

Hubs can mimic community eigenvectors

FAILURE MODE

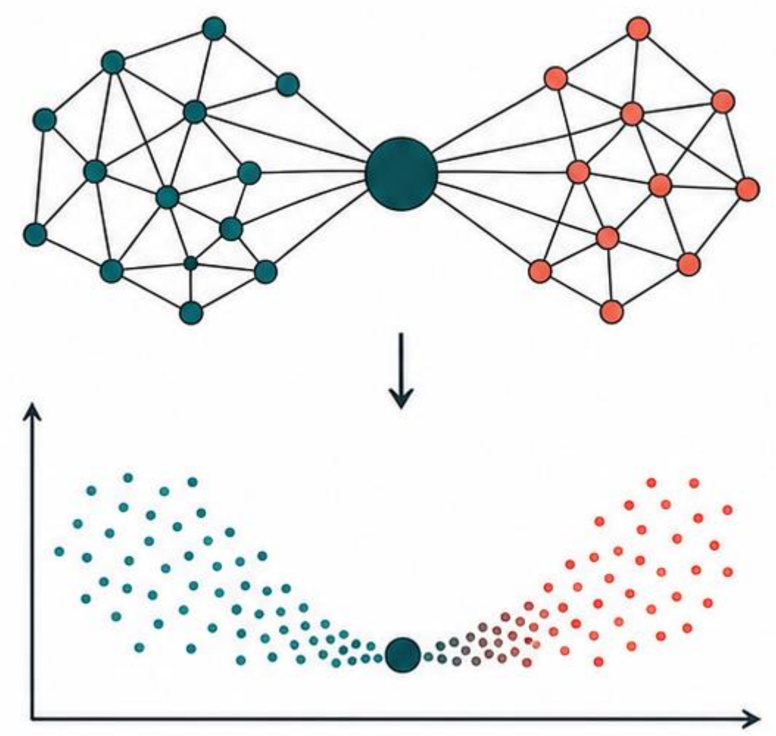
A hub can dominate $\boldsymbol{A}$'s leading modes.

It bends the embedding.

MITIGATION

Use normalization, regularization,

or degree-corrected block models.



Inspect degree distributions before interpreting spectral clusters.

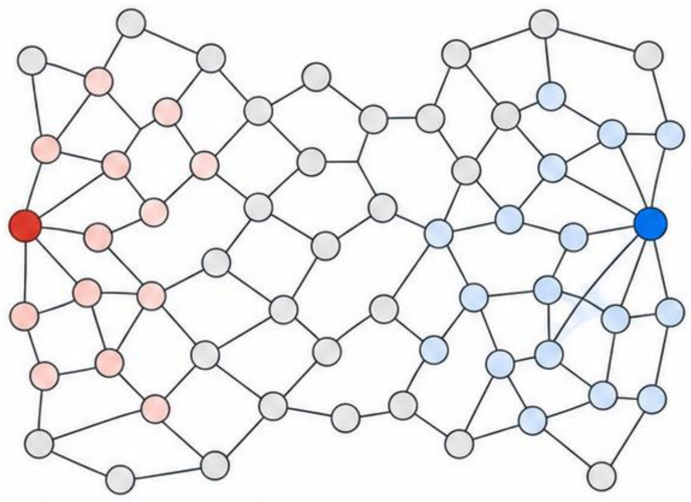
A few labels turn clustering into node classification

Known labels on $J \subset V$ anchor the community names

$$min_f \sum_{i \in J} \ell\left(f(\mathbf{A})_i, y_i\right)$$

- Input: $\boldsymbol{A}$ plus labels on subset J.
- Output: class probabilities for all nodes.

Unlabeled nodes affect embeddings through A.
ℓ = cross-entropy: the trainable surrogate of the 0–1 loss.



Supervision resolves label identity and improves ambiguous regions.

A linear head makes spectral embeddings trainable

FIXED GRAPH REPRESENTATION

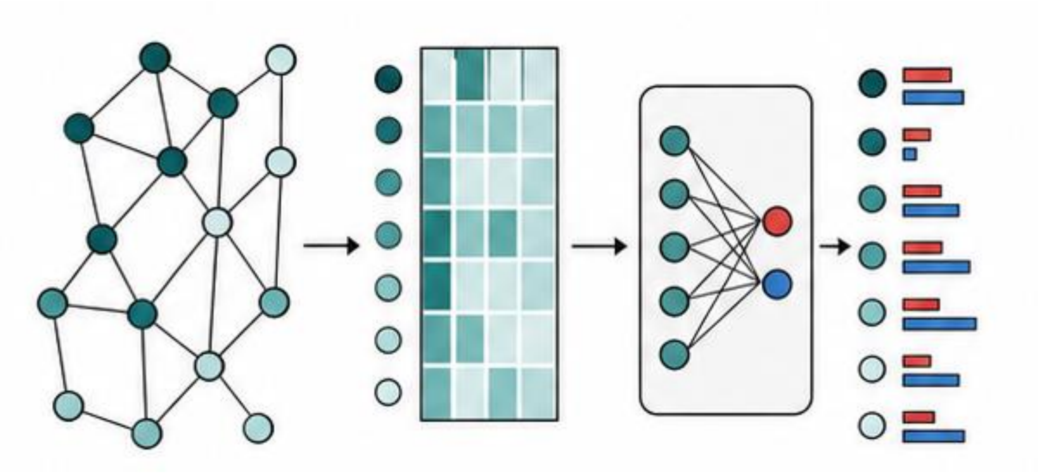
Use the top C eigenvectors.

Each row u_i represents node i.

TRAINABLE CLASSIFIER

$\boldsymbol{Z} = \mathrm{softmax}(\boldsymbol{UW})$

$\boldsymbol{W}$ maps embeddings to class logits.



The graph supplies features; labeled nodes learn the decision boundary.

Cross-entropy trains on labeled nodes only

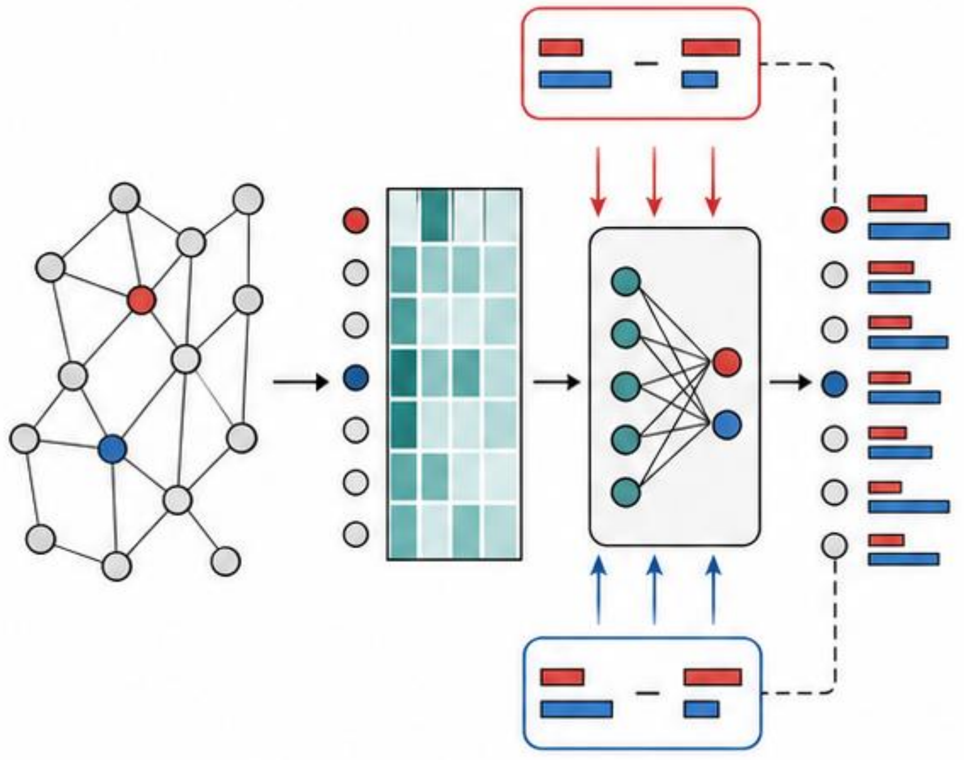
PREDICTION

$z_i = softmax(u_i\mathbf{W})$ for every node i.

SUPERVISED LOSS

$$L = -\sum_{i\in J}\sum_{c} y_{ic} log z_{ic}$$

Gradients update the shared $\boldsymbol{W}$.



One shared classifier transfers supervision to unlabeled embeddings.

Spectral embeddings are graph layers

01

Graph-dependent input

$U = V_C(A)$ encodes global structure.

02

Shared linear map

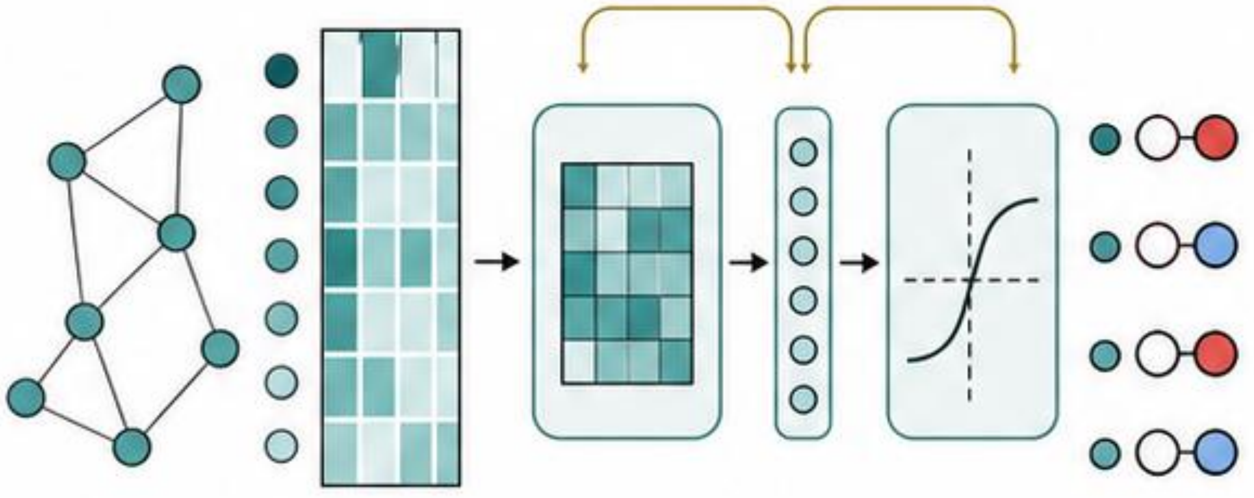
Every node applies the same W to u_i.

03

Pointwise nonlinearity

Softmax or σ produces node outputs.

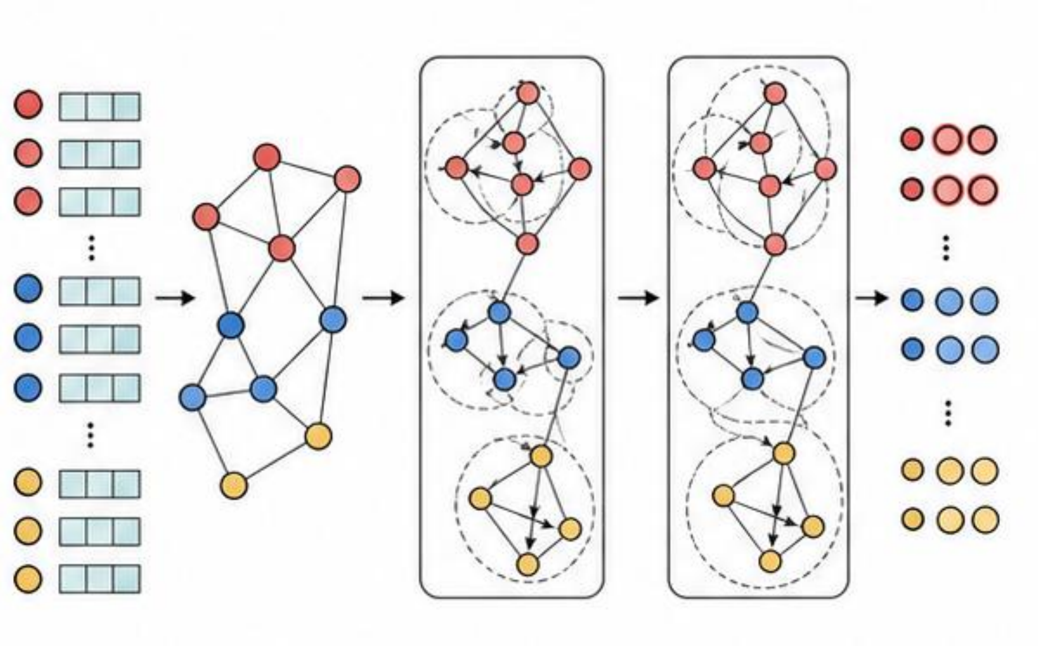
$$\mathbf{Z} = \sigma(\mathbf{V}_C(\mathbf{A})\mathbf{W}), \mathbf{W} \in \mathbb{R}^{C \times C}$$



A GCN learns embeddings by message passing

Input: $\boldsymbol{A}$, node features $\boldsymbol{X}$, labels on J; use $\widehat{\boldsymbol{S}} = \widetilde{\boldsymbol{D}}^{-1/2}\widetilde{\boldsymbol{A}}\widetilde{\boldsymbol{D}}^{-1/2}$, $\widetilde{\boldsymbol{A}} = \boldsymbol{I} + \boldsymbol{A}$.

$$\boldsymbol{Y}^{(\ell+1)} = \sigma\left(\boldsymbol{H}^{(\ell)}(\widehat{\boldsymbol{S}})\boldsymbol{Y}^{(\ell)}\right),$$



GCNs learn community-sensitive features without explicitly storing eigenvectors.

Contextual SBM combines edges and node features

GRAPH EVIDENCE

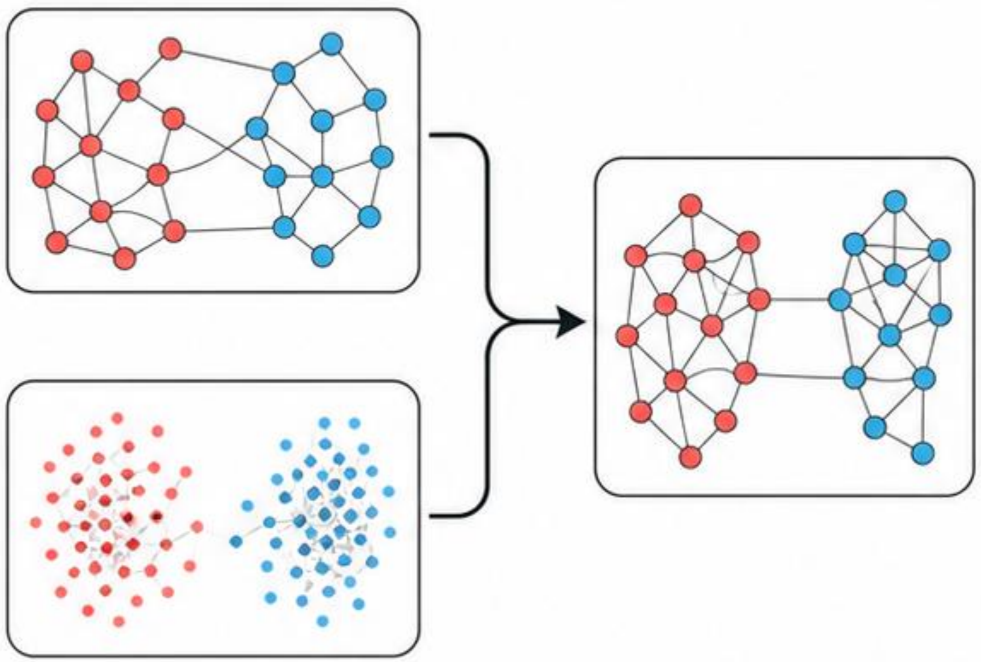
$\mathbf{A}$ follows an SBM.

Edges help when $p - q$ is large.

FEATURE EVIDENCE

x_i is correlated with y_i.

Attributes help when topology is weak.



$$\log P(\mathbf{A}, \mathbf{X} \mid \mathbf{y}) = \log P(\mathbf{A} \mid \mathbf{y}) + \log P(\mathbf{X} \mid \mathbf{y}).$$

Combine both signals when neither topology nor features are sufficient alone.

When $p = q$, topology carries no community signal

NO-SIGNAL CASE

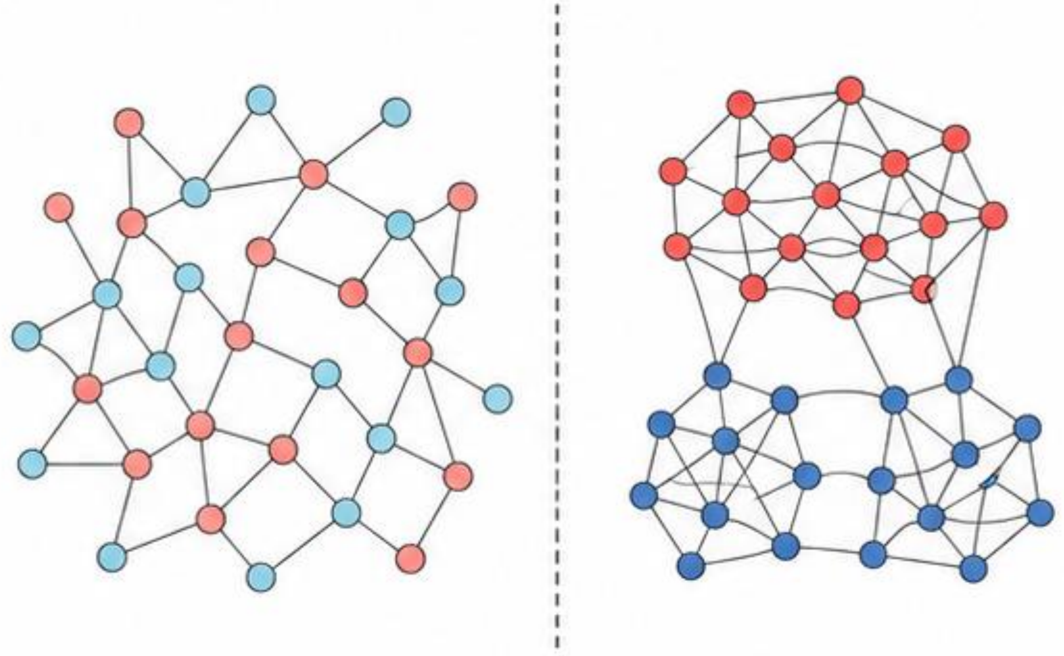
$p = q \Rightarrow$ identical connection rates.

The graph collapses to Erdős–Rényi.

DETECTABLE CASE

$p > q$ creates an informative mode.

Its strength grows with $p - q$.



When the topology is independent of the labels, no algorithm can recover them.

Sparse scaling keeps average degree finite

EDGE PROBABILITIES

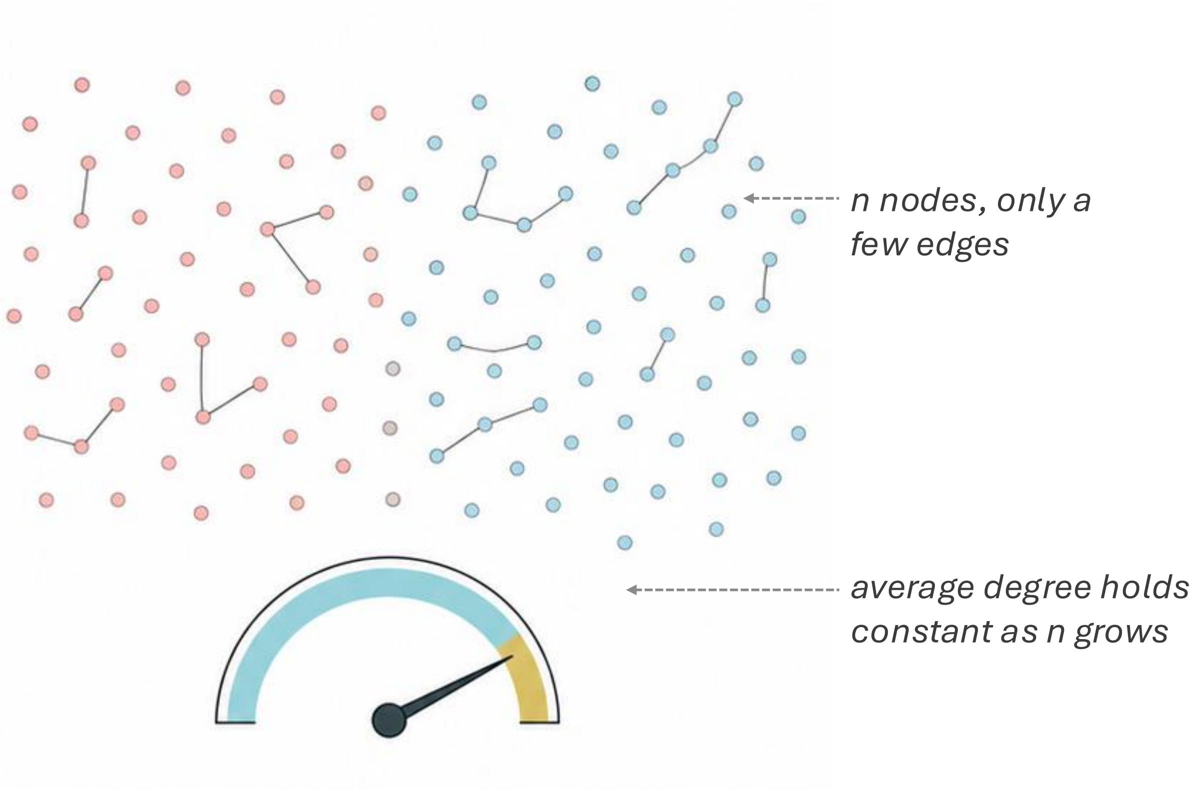
$$p = \frac{a}{n}, \qquad q = \frac{b}{n}$$

Each potential edge becomes rare.

AVERAGE DEGREE

$$\bar{d} \approx \frac{a+b}{2}$$

The graph remains sparse as n grows.



In sparse graphs, structural signal and random fluctuations are comparable.

Detectability has a sharp signal-to-noise threshold

NORMALIZED SIGNAL

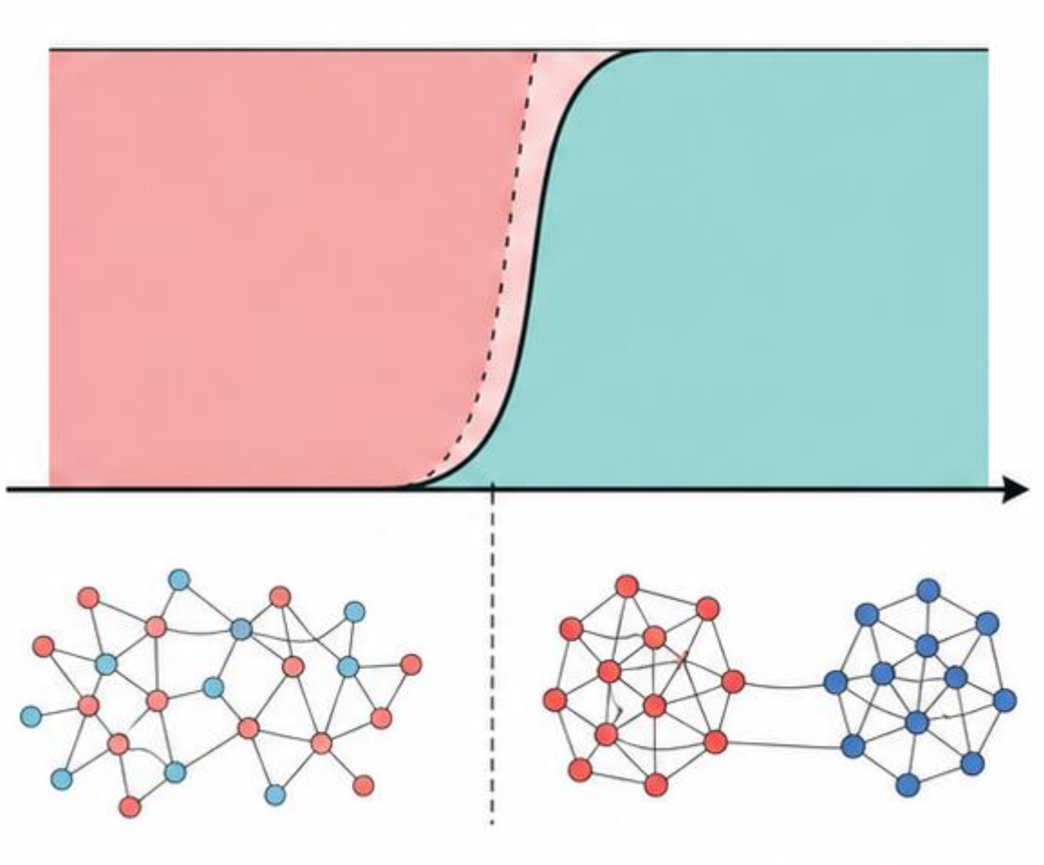
$$SNR = \frac{(p-q)^2}{2(p+q)}$$

Compare n·SNR with 1.

SPARSE TWO-BLOCK FORM

$$p = \frac{a}{n}, q = \frac{b}{n}$$

$$n \cdot SNR = \frac{(a-b)^2}{2(a+b)}$$



Weak recovery needs $(a-b)^2 > 2(a+b)$; below it, no algorithm beats chance.

A Proof Of The Block Model Threshold Conjecture, Mossel, Neeman, Sly, 2015

Estimate C with eigengaps

01 **SPECTRUM**

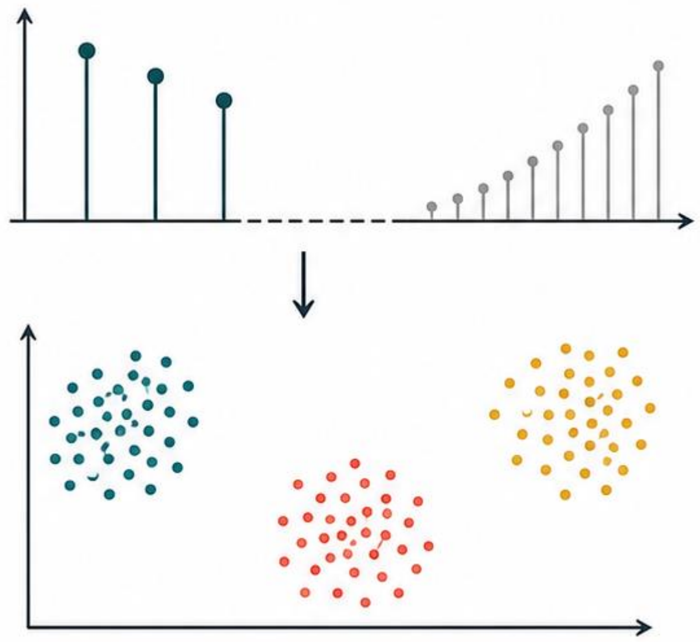
Look for a stable gap after the informative eigenvalues.

02 **EMBEDDING**

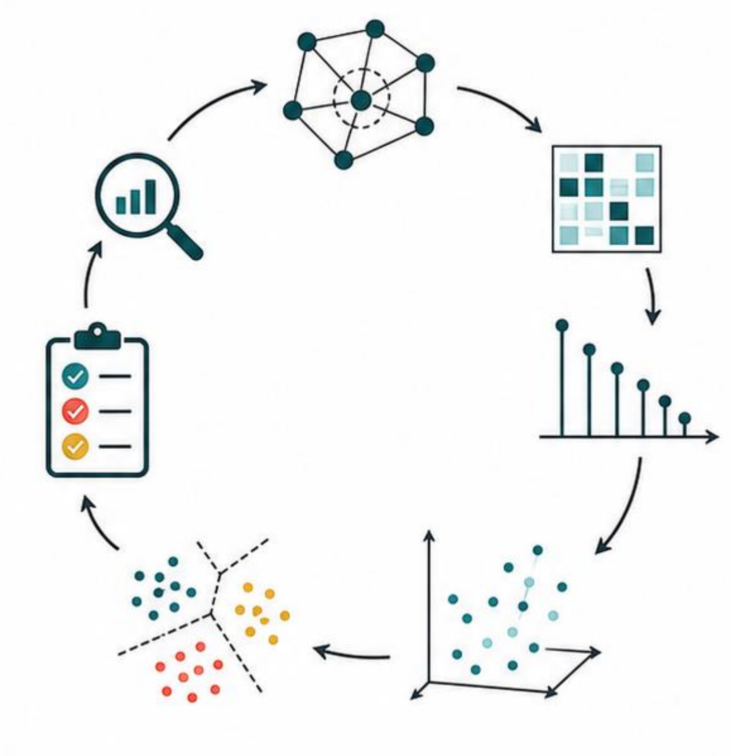
Verify that row embeddings form coherent, separated groups.

03 **STABILITY**

Recluster across operators, seeds, and graph perturbations.



Model the graph. Inspect the spectrum.



01 SBM defines what community signal means

02 Eigenvectors reveal low-rank block structure

03 Labels make embeddings trainable

04 Thresholds expose impossible regimes

Community detection is estimation: choose the model and operator, then test whether signal survives noise.