Community Detection And Community Optimization

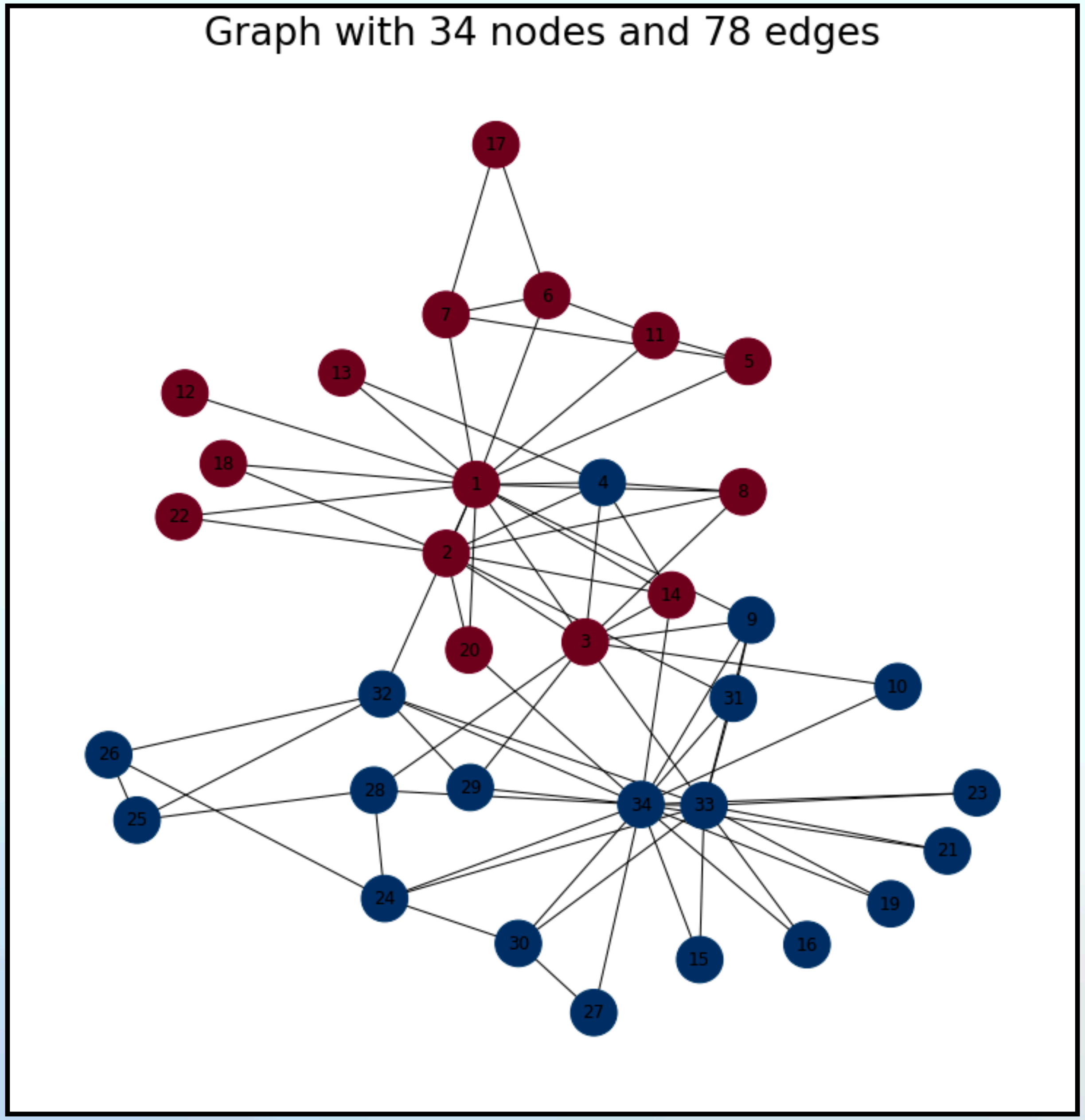
=

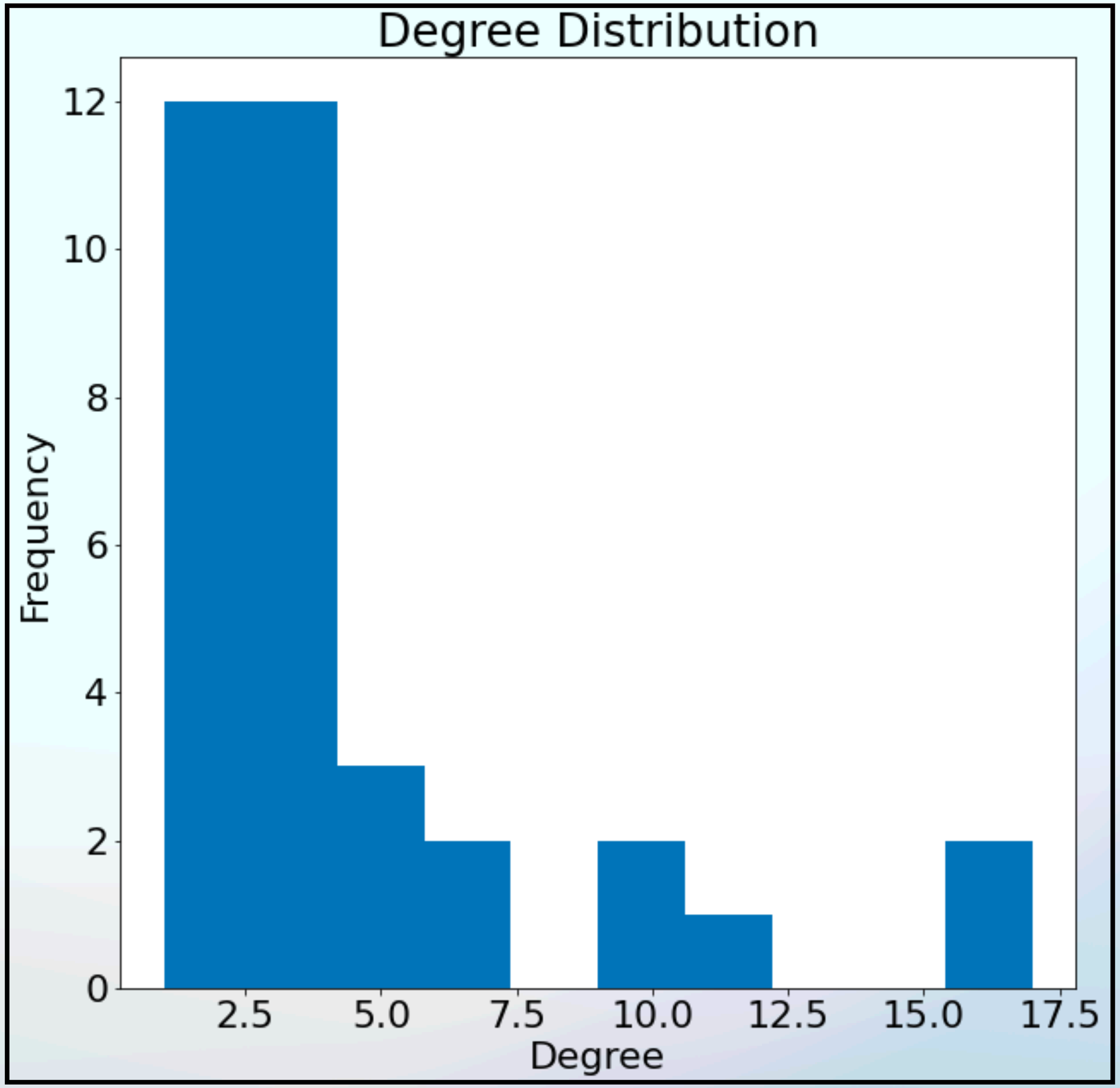
Content

- Modularity
- Modularity Maximation:
    - Newman's FastGreedy Algorithm
    - Louvain Algorithm
- Resolution Limit
- Conductance

Modularity

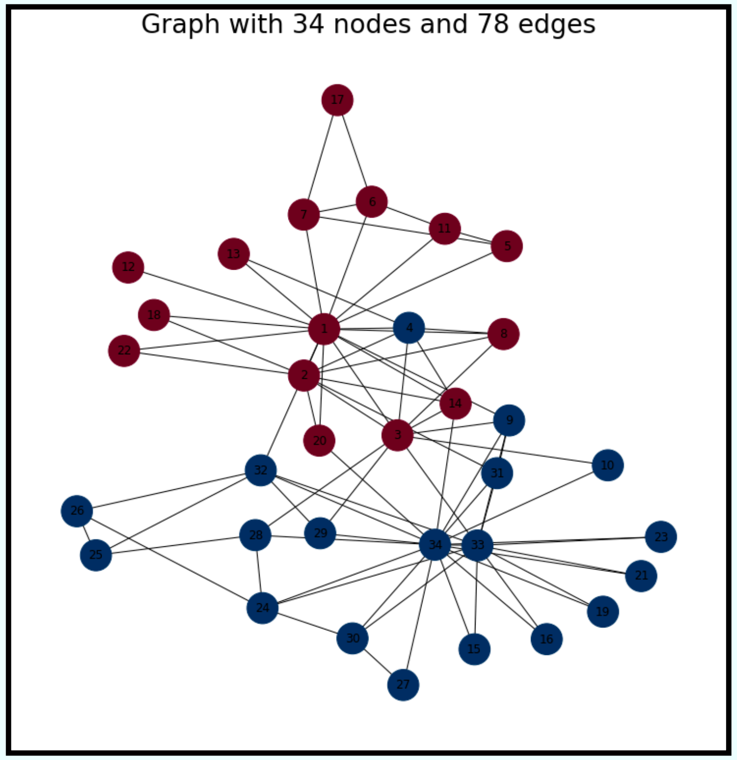
Demo: Zachary Karate graph and its ground-truth communities

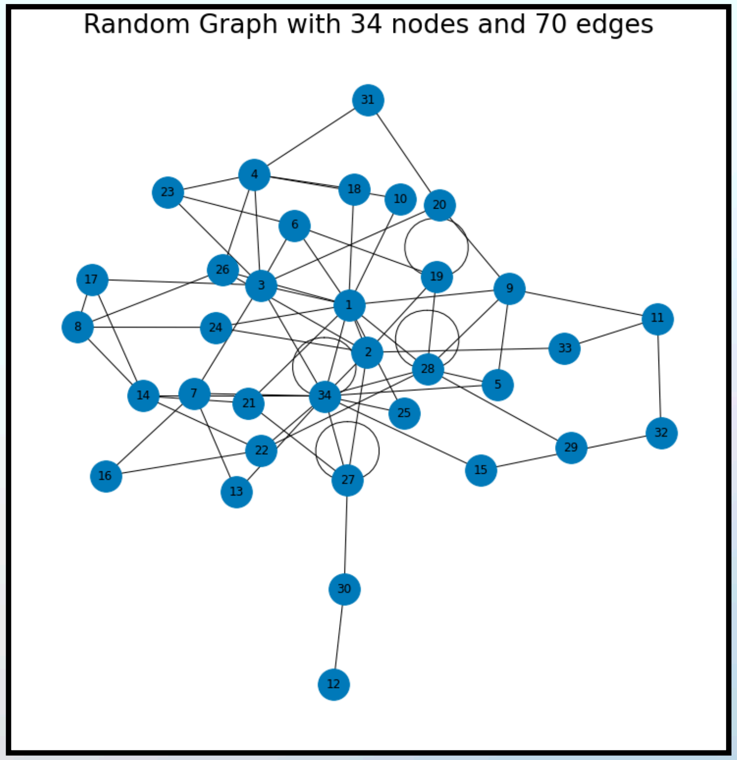




Modularity

- How do we measure modularity?
    - Compare the topology of the random network to the given network
    - how condensed is the graph?
- measures the density of links inside communities compared to links between communities
- Basic Hypothesis:
    - Random network lacks inherent community structure
    - Measure how clustered our network is relative to what's randomly expected
- Usually used for optimization or relative comparison





Modularity

Computing Modularity

$$M = \frac{1}{2m} \sum_{\forall u,v \in C} \left(A_{uv} - \frac{d(u)d(v)}{2m} \right)$$

- Where,
    - $M \rightarrow$ Modularity
    - $m \rightarrow \ |E(G)|$, number of Edges
    - $\forall u, v \in C \rightarrow$ vertex pairs in the community
    - $A \rightarrow$ Adjacency Matrix
    - $\dfrac{d(u)d(v)}{2m} \rightarrow$ expected # of edges between u and v in a random graph
- Value in between: $[-1/2, 1]$
-

Modularity Maximization

- Maximize modularity as a community detection algorithm
- Usually: Greedy Agglomerative
    - Each observation starts in its cluster, and greedily, pairs of clusters are merged as one moves up the hierarchy.
        - Newman Algorithm
        - Louvain Algorithm

Modularity Maximization

Newman's FastGreedy Algorithm

- Greedy Agglomerative Algorithm:
    - Initially: all vertices in unique communities
    - Iterate while # communities > 1:
        - Merge community pair that maximizes modularity
- Pros:
    - Encapsulates the hierarchy
- Cons:
    - Issues with modularity calculations in practice

Modularity Maximization

Louvain Algorithm

- Similar to the Newman Algorithm
    - But with explicit edge contractions
- Initially, each node in the network is assigned a community
- Edge contraction:
    - For each node, compute the difference in modularity if it is placed in its neighbors' community. Move if there is any gain.
    - Contract all the nodes within the communities to a "super-node"
- Repeat the edge contraction method until modularity does not increase

Issues with Modularity

Resolution Limit

- It cannot resolve relatively small communities!

$$M = \frac{1}{2m} \sum_{\forall u,v \in C} \left(A_{uv} - \frac{d(u)d(v)}{2m} \right)$$

  - the expected number of edges between nodes u and v in a random graph does not adequately scale for detecting smaller communities in large networks

  - become insensitive to communities smaller than a certain scale

Issues with Modularity

Resolution Limit

- How small?

- Change in modularity by combining communities A and B

$$\Delta M = \frac{l_{AB}}{m} - \frac{k_A k_B}{2m^2}$$

  - $l_{AB} \rightarrow$ edges between A and B

  - $k_A, k_B \rightarrow$ sum of degrees of vertives in A or B

- Consider:

$$\frac{l_{AB}}{m} = \frac{k_A k_B}{2m^2} \implies l_{AB} = \frac{k_A k_B}{2m}$$

Issues with Modularity

Resolution Limit

- Assuming we gain from merging A and B

$$l_{AB} > \frac{k_A k_B}{2m}, \; k_A = k_B = k, \text{ and } \; l_{AB} = 1$$

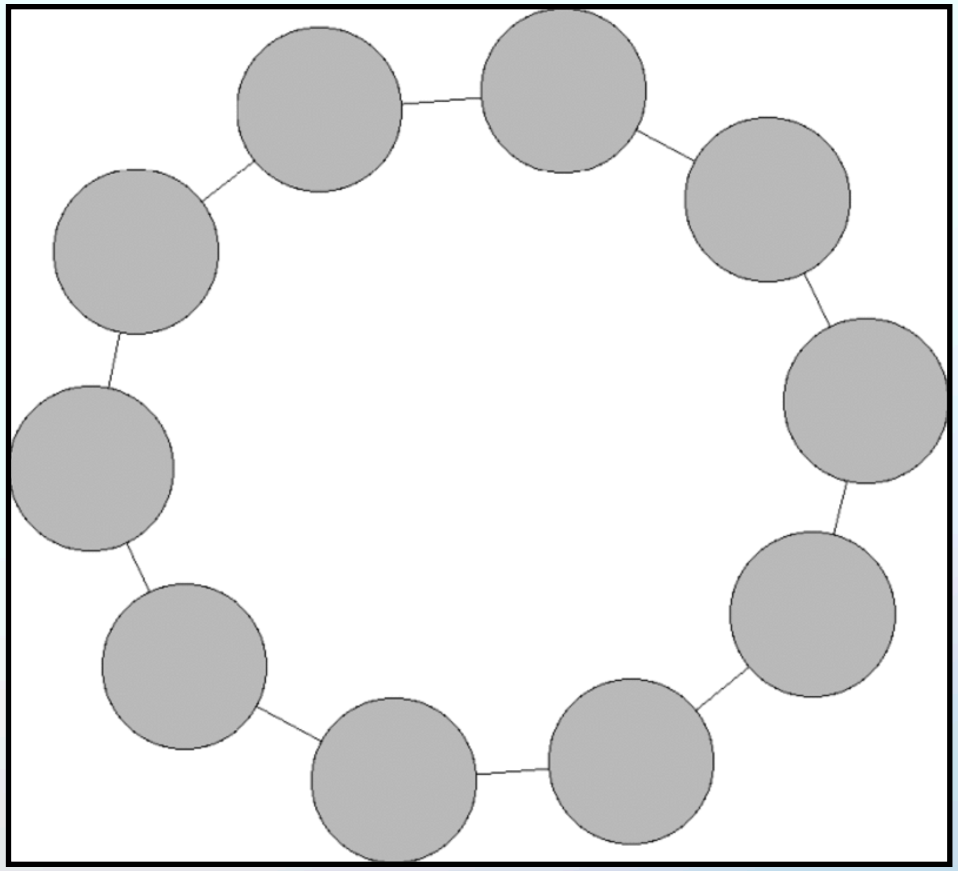
- Then,

$$1 > \frac{k^2}{2m}$$

$$2m > k^2$$

$$\sqrt{2m} > k$$

- Hence, the lower bound on the size of the community that modularity optimization can find is $k \leq \sqrt{2m}$.



Example: Ring of Cliques

Issues with Modularity

Other Issues:

- The expected density of a random network:

$$\frac{d(u)d(v)}{2m}$$

- Different types of networks have different degree distribution:
    - Dense graphs, skewed, or dense subgraphs
- Takeaways: Our modularity value can be meaningless in skewed or dense networks
- Potential Fix: Use the actual attachment probabilities

Conductance

- A measure of how quickly a random walk converges to a stationary state
- Lower Conductance $\Longrightarrow$ More defined communities
- Defined in terms of edge cut, S and $\bar{S}$:
- Conductance (S) = $\dfrac{\text{cut}(S)}{\min(K_S, K_{\bar{S}})}$
    - $\text{cut}(S) \rightarrow$ number of edges in cut
    - $K_S \rightarrow$ sum of degrees in S