Final Progress Report

Graph Mining with the Configuration Model and Community Detection in Diverse Networks

Name of Fellow: Tvisha Elley
Applicaton No.: MATS698
Name of Guide: Dr. Ganesh Viswanathan
Host Institute: IIT Bombay
Department: Chemical Engineering

Summer Research Fellowship Programme Report submitted to the Indian Academy of Sciences, Indian National Science Academy, and The National Academy of Sciences, India

Abstract

This project explored the structural properties of networks and their analysis using graph mining techniques, with a focus on comparing real-world networks to their randomised counterparts generated by the configuration model. The configuration model network is primarily used to generate random networks with a specified degree distribution, serving as a null model for network analysis to understand how network structure, especially node degrees, influences emergent properties. In particular, the model was used to enable a systematic study of how node centrality distributions vary between real and random structures. To quantify these differences, information-theoretic measures such as the Jensen-Shannon Divergence were applied.

In addition, community detection methods were applied specifically to the TNF-α signaling network, with an emphasis on comparing approaches such as modularity-based Louvain and Leiden algorithms with the Constant Potts Model (CPM). This comparison highlights differences in the number and resolution of communities detected by each method and serves as a basis for understanding how the choice of algorithm and resolution parameter can affect the inferred community structure.

The study examined four data sets: a proximity network, the US power grid network, a scientific collaboration network of authors, and a protein-protein interaction network involved in TNF-α signaling. The results highlight key differences between real and randomised network structures, offering a deeper understanding of how topology influences functional organisation.

Contents

Chapter 1

Introduction

Networks provide a powerful mathematical framework for representing and analysing complex systems across disciplines, from social interactions and infrastructure to biological processes and scientific collaborations. In this framework, entities are represented as nodes and their relationships as edges, enabling the study of how structure influences function.

This report focuses on graph mining, the analysis of networks to identify patterns, important nodes, and structural features. By examining properties such as degree distributions, centrality measures, and community structure, we can assess node roles, network efficiency, and resilience. Here, graph mining techniques are applied to both simple graphs and directed networks, using real-world datasets from multiple domains.

Real-world networks often display patterns that are different from random networks. Comparing them with degree-preserving random counterparts generated by the configuration model helps distinguish structural features from those arising by chance. For instance, differences in centrality distributions can highlight how topology shapes node importance, which is one of the main focuses of this study.

1.1 Datasets Used in This Study

This study analyses four real-world networks that span social, infrastructure, academic, biological, and e-commerce domains.

- **Proximity Network:** A human contact network capturing face-to-face proximity interactions. [13]
- **U.S. Power Grid Network:** Represents the electrical infrastructure of the Western United States including generators, substations, and transmission lines. [12]
- **Coauthorship Network:** A collaboration network of authors submitting to the arXiv's General Relativity and Quantum Cosmology category between 1993 and 2003. [6]
- **Protein–Protein Interaction (PPI) Network:** A biological network representing interactions in the TNF-α signaling pathway. This network was curated in the laboratory of Prof. Ganesh Viswanathan, who served as my project guide.

Dataset	**Nodes**	**Edges**	**Type**	**Directed?**
Proximity (Infect-Dublin)	410	2,765	Social/Proximity	No
U.S. Power Grid	4,941	6,594	Infrastructure	No
Coauthorship Network	5,242	14,496	Scientific Collaboration	No
TNF-α PPI Network	341	487	Biological	Yes

Table 1.1: Summary of datasets used in the study.

Chapter 2

Methodology

This chapter outlines the methods used to analyze the selected networks, generate comparable random counterparts, and evaluate structural features.

2.1 Configuration Model

To distinguish intrinsic structural patterns from those arising by chance, each network was compared with a degree-preserving random counterpart generated using the *configuration model* [1, 10]. The configuration model preserves the degree sequence of the original network while rewiring edges randomly, producing a network with the same degree distribution but otherwise randomized connections.

Instead of using the default `networkx` implementation, which may produce self-loops and multi-edges, this study follows the formulation described by Maslov and Sneppen [8]. This approach ensures the generation of simple graphs (no self-loops or multiple edges) while preserving the degree sequence. The method proceeds by repeatedly selecting two edges at random and swapping their endpoints, subject to the constraint that the resulting graph remains simple. Self-loops and multiple edges are excluded since they lack meaningful interpretation in the studied networks (e.g., a node interacting with itself or duplicate connections between the same pair of nodes). Their presence would artificially inflate degree, bias centrality measures, and distort comparisons with real-world networks that are typically represented as simple graphs.

2.1.1 Configuration Model Implementation

The following pseudocode outlines the implementation of a configuration model that preserves the degree sequence while avoiding self-loops and multi-edges:

Algorithm 1 Non-Multi-Edge Configuration Model

Require: Graph G

edge_list ← list of all edges in G
num_trials ← number of edges in G
for trial = 1 **to** num_trials **do**
  if length of edge_list ¡ 2 **then**
    break
  end if
  Randomly select edge1 and edge2 from edge_list
  if edge1 = edge2 **then**
    continue
  end if
  Let edge1 = (a, b), edge2 = (c, d)
  new_edges ← [(a, d), (c, b)]
  if a = d OR c = b **then**
    continue {avoid self-loops}
  end if
  if G already has (a, d) OR (c, b) **then**
    continue {avoid multi-edges}
  end if
  Remove edge1 and edge2 from G
  Add new_edges to G
  Remove edge1 and edge2 from edge_list
end for
return G

2.2 Centrality Measures

Centrality measures quantify the importance of nodes in a network, based on criteria such as connectivity, proximity to others, or control over communication paths. Several centrality measures were computed for both the original and configuration-model networks to quantify node importance and compare their structural roles. Definitions and formulae follow Newman [10].

- **Eigenvector Centrality:** Eigenvector centrality assigns relative scores to nodes such that connections to high-scoring nodes contribute more than connections to low-scoring ones. Formally, if $\mathbf{A}$ is the adjacency matrix and x_i is the centrality of node i, then

$$x_i = \frac{1}{\lambda} \sum_j A_{ij} x_j,$$

  which in vector form is

$$\mathbf{A}\mathbf{x} = \lambda \mathbf{x},$$

  where λ is the largest eigenvalue of $\mathbf{A}$ and $\mathbf{x}$ is the corresponding eigenvector. The elements of this eigenvector correspond to the centralities of each node.

- **Katz Centrality:** Katz centrality addresses key limitations of eigenvector centrality. In eigenvector centrality, a node's score depends entirely on the centrality of its

neighbors; as a result, nodes with zero in-degree (or those connected only to such nodes) receive a score of zero. This issue can propagate, leading to entire regions of directed networks being undervalued.

Katz centrality resolves this by introducing a damping factor α and a constant offset β, which ensure that every node receives some baseline contribution independent of its neighbors. This makes Katz centrality particularly suitable for directed networks where sinks or sources would otherwise collapse to zero under eigenvector centrality. Formally, it is defined as:

$$x_i = \alpha \sum_j A_{ij} x_j + \beta,$$

where α controls the influence of neighbors and β ensures non-zero centrality for all nodes.

In this study, we used $\alpha = 0.85$ and $\beta = 1$.

- **PageRank Centrality:** As defined by Newman [10], PageRank is a variation of Katz centrality where the centrality a node derives from it's neighbours is proportional to their centrality divided by their out degree. The PageRank score p_i of node i satisfies:

$$p_i = \alpha \sum_j \frac{A_{ij}}{k_j^{\text{out}}} p_j + \beta$$

where n is the number of nodes, A_{ij} is the adjacency matrix element from i to j, and k_j^{out} is the out-degree of node j. In this study, we set $\alpha = 0.85$ and used $\beta = 1$.

- **Closeness Centrality:** Closeness centrality of a node i is defined as the reciprocal of the sum of the shortest-path distances from i to all other nodes that are reachable from it. Formally:

$$C_C(i) = \frac{n-1}{\sum_{j \neq i} d(i,j)}$$

where $d(i,j)$ is the length of the shortest path between i and j, and n is the number of nodes in the connected component containing i. Nodes with high closeness centrality have, on average, short distances to all other nodes, making them effective spreaders of information or influence.

- **Betweenness Centrality:** Betweenness centrality of a node i quantifies the number of times i acts as a bridge along the shortest paths between pairs of other nodes. It is given by:

$$C_B(i) = \sum_{s \neq i \neq t} \frac{\sigma_{st}(i)}{\sigma_{st}}$$

where σ_{st} is the total number of shortest paths between s and t, and $\sigma_{st}(i)$ is the number of those paths passing through i. This measure highlights nodes that control communication flow or act as critical intermediaries within the network.

2.3 Evaluation Metrics

To quantitatively compare the structural properties of the original networks and their configuration-model counterparts, the following metrics were used:

- **Jensen–Shannon (JS) Divergence** [7]: A symmetric measure of similarity between two probability distributions P and Q, based on the Kullback–Leibler divergence. The JS divergence is calculated as:

$$JS(P \parallel Q) = \frac{1}{2}KL(P \parallel M) + \frac{1}{2}KL(Q \parallel M)$$

where $M = \frac{1}{2}(P + Q)$ and KL denotes the Kullback–Leibler divergence:

$$KL(P \parallel Q) = \sum_i P(i) \log \frac{P(i)}{Q(i)}$$

The JS divergence is bounded between 0 and 1 (when using base-2 logarithms), with 0 indicating identical distributions and 1 indicating non-overlapping distributions. In this study, histograms used to estimate P and Q were binned using the **Freedman–Diaconis rule** [5] to ensure optimal bin width and avoid bias in the JS computation.

- **Kolmogorov–Smirnov (KS) Test** [9]: A non-parametric test used to determine whether or not two samples are drawn from the same distribution. If $F_n(x)$ and $G_m(x)$ are the empirical cumulative distribution functions (ECDFs) of the two samples of sizes n and m, the KS statistic is defined as:

$$D_{n,m} = \sup_x |F_n(x) - G_m(x)|$$

where $\sup_x$ denotes the supremum over all x. The KS statistic ranges from 0 to 1, with larger values indicating greater divergence between the distributions.

- **Coefficient of Determination (R^2):** A statistical measure used to quantify how well observed outcomes are replicated by a model. In this study, R^2 was used to evaluate the agreement between node-level centrality values of the original network and the average values from configuration-model networks. It is defined as:

$$R^2 = 1 - \frac{\sum_i (y_i - \hat{y}_i)^2}{\sum_i (y_i - \bar{y})^2},$$

where y_i are the observed values (original network centralities), $\hat{y}_i$ are the predicted values (averaged random network centralities), and $\bar{y}$ is the mean of the observed values. An R^2 score approaching 1 indicates a good fit, while a score near 0 denotes poor correspondence between the two sets of values.

2.4 Community Detection

Community detection aims to partition a network into groups of nodes (communities) such that nodes within the same group are more densely connected to each other than to the rest of the network. Identifying such structures helps reveal organisational structures such as functional modules in biological systems, clusters in social networks, or subnetworks in infrastructure systems.

2.4.1 Quality Functions

A central idea in community detection is the use of **quality functions**—numerical measures that quantify how good a given partition is. The algorithm then seeks to maximize (or minimize) this function.

Modularity. Modularity [11] is the most widely used quality function for community detection. It measures the fraction of edges that fall within communities minus the expected fraction in a degree-preserving random graph (the configuration model). For a partition of the network into communities, modularity Q is given by:

$$Q = \frac{1}{2m} \sum_{ij} \left[A_{ij} - \frac{k_i k_j}{2m} \right] \delta(c_i, c_j)$$

where:

- A_{ij} is the adjacency matrix.
- k_i and k_j are degrees of nodes i and j.
- m is the total number of edges.
- $\delta(c_i, c_j)$ is 1 if i and j are in the same community, 0 otherwise.

Higher Q values indicate a better separation of communities from a random baseline.

Constant Potts Model (CPM). The Constant Potts Model [14] is an alternative quality function designed to overcome the resolution limit problem [4] of modularity. CPM defines the quality function as:

$$\mathcal{H} = \sum_{C} \left[E_C - \gamma \binom{n_C}{2} \right]$$

where:

- E_C is the number of edges inside community C.
- n_C is the number of nodes in community C.
- γ is the **resolution parameter** controlling the size of detected communities.

By tuning γ, one can detect communities at different scales.

2.4.2 Algorithms for Community Detection

Community detection was performed using the Leiden algorithm [15], which builds upon the Louvain method [2, 3]. The Louvain algorithm is a fast, hierarchical, greedy optimization approach that maximizes modularity. It proceeds in two main phases:

1. **Local moving phase:** Each node is moved to the community of its neighbor that provides the greatest increase in modularity.

2. **Aggregation phase:** Communities are aggregated into super-nodes, producing a smaller network on which the process is repeated.

This multi-level process yields a hierarchical decomposition of the network.

The Leiden algorithm improves upon Louvain in three key ways:

1. Guarantees that all communities are **well-connected**, avoiding fragmented clusters.

2. Enhances the speed, stability, and robustness of the results.

3. Allows optimization of both modularity and the Constant Potts Model (CPM), overcoming the resolution-limit problem [4, 14].

Leiden introduces a **refinement phase** between local moving and aggregation, ensuring that poorly connected communities are split before aggregation. In this study, Leiden was used as the primary method for community detection due to its improved guarantees and flexibility.

Chapter 3

Results and Analysis

This chapter presents the work completed during the study, including network generation, structural analysis, and community detection outcomes. The analysis proceeds in three stages. First, networks are introduced along with their structural properties such as degree distributions and centrality measures. Second, these properties are compared with corresponding random networks generated using the configuration model, with statistical tests used to quantify similarities and differences. Finally, community detection methods are applied to the protein interaction network to uncover structural groupings, with validation against biological or functional classifications highlighted as an important direction for future work.

3.1 Degree Distributions and Centrality Correlations

This section presents the degree distributions and centrality measures of the networks, along with comparisons to their randomized counterparts. Scatter plots of centrality versus degree are used to visualize dependencies, while statistical metrics—including the Kolmogorov–Smirnov test, and Jensen–Shannon divergence—quantify the relationships and similarities between the observed and random networks.

3.1.1 Proximity Network

The proximity network [13] used in the study represents individuals as nodes, with edges denoting close physical interactions. It is an undirected and unweighted social network that captures human contact patterns, making it particularly relevant for studying information or disease spreading dynamics.

Degree and Centrality Distributions

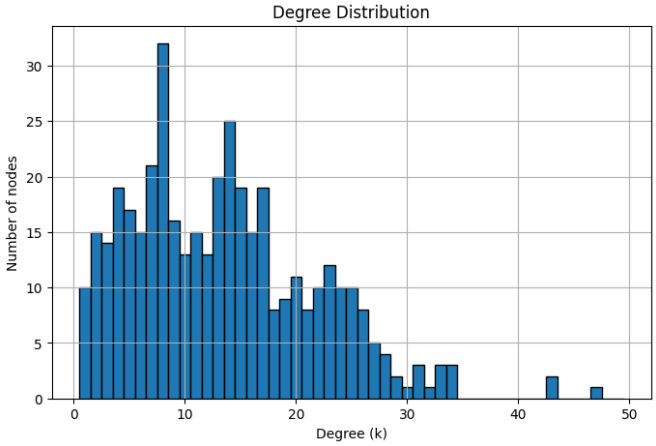

Figure 3.1: Degree distribution for the Proximity network.

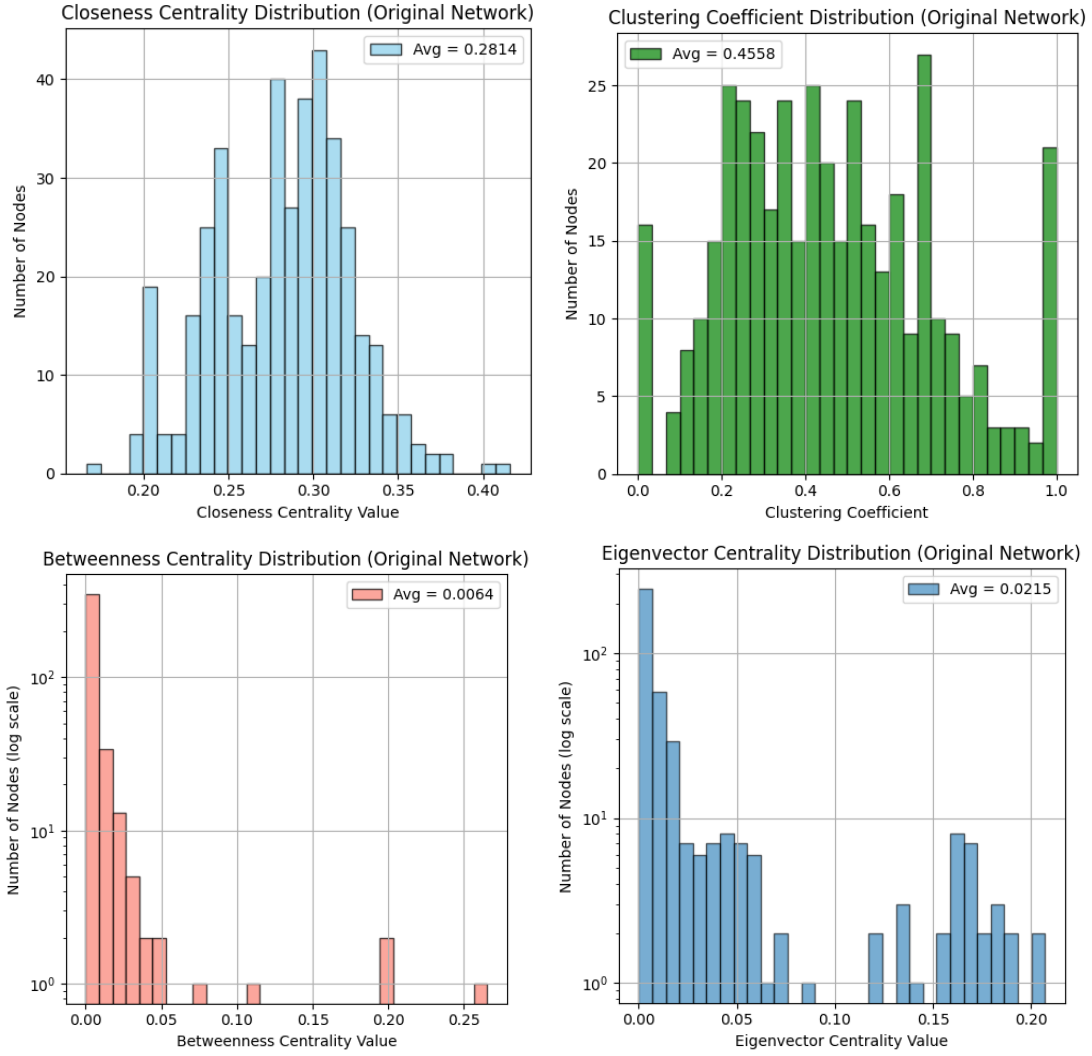

Figure 3.2: Histograms of centrality measures for the Proximity network.

Centrality Measure	Average Value
Closeness	0.2814
Betweenness	0.0064
Clustering Coefficient	0.4558

Table 3.1: Average centrality values for the Proximity network.

Configuration-Model Results

A total of 100 random graphs were generated using the configuration model for the statistics given below.

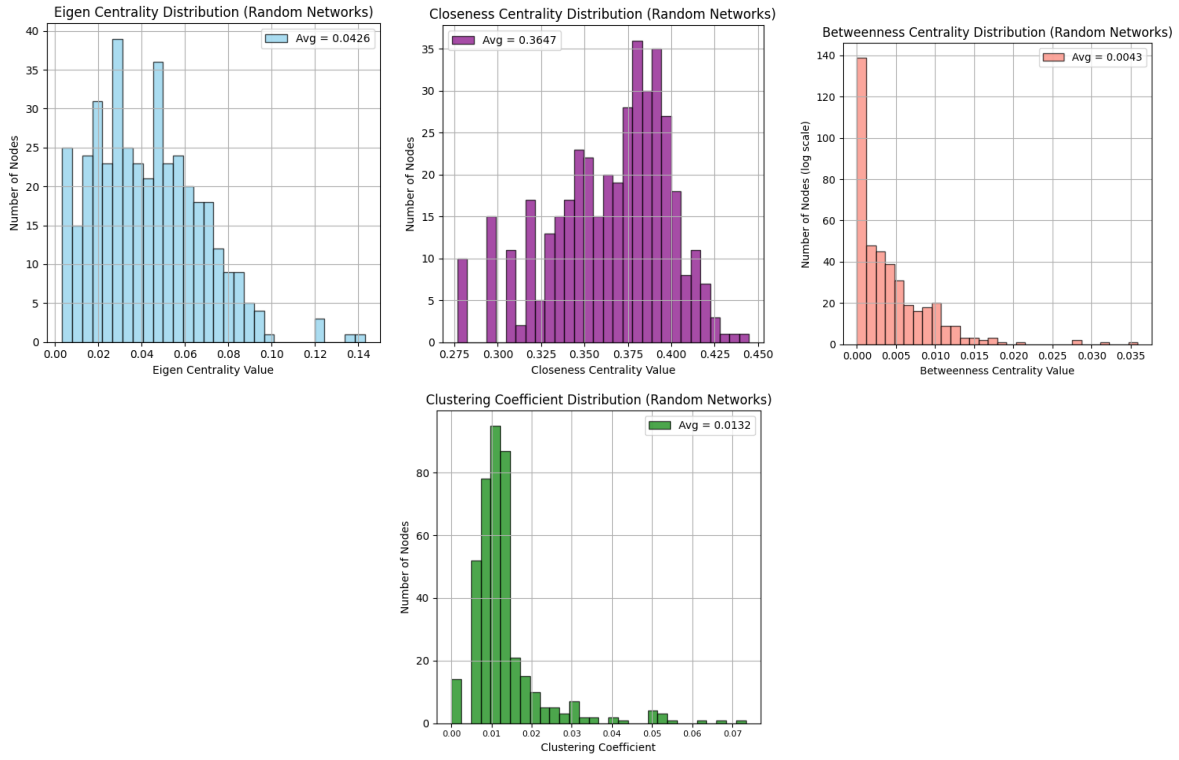

Figure 3.3: Histograms of mean centrality measures across random networks generated from the Proximity network.

Centrality Measure	Average Value
Closeness	0.3647
Betweenness	0.0043
Clustering Coefficient	0.0132

Table 3.2: Average centrality values for the random-averaged Proximity network.

Comparison with Orignal Network

Centrality Measure	Avg. KS Statistic	Avg. JSD
Eigenvector Centrality	0.6330	0.6670
Closeness Centrality	0.7447	0.7420
Betweenness Centrality	0.1195	0.2102
Clustering Coefficient	0.9408	0.9508

Table 3.3: Average KS and JSD results for configuration model comparisons with the original network across centrality distributions.

3.1.2 US Power Grid

The Power Grid network used in this study is an undirected, unweighted network representing the topology of the Western States Power Grid of the United States. Nodes correspond to generators, transformers, and substations, while edges represent transmission lines. This dataset was compiled by Watts and Strogatz and made publicly available [12]. It serves as a canonical example of a real-world infrastructure network and has been widely used in the study of network robustness and structural properties.

Degree and Centrality Distributions

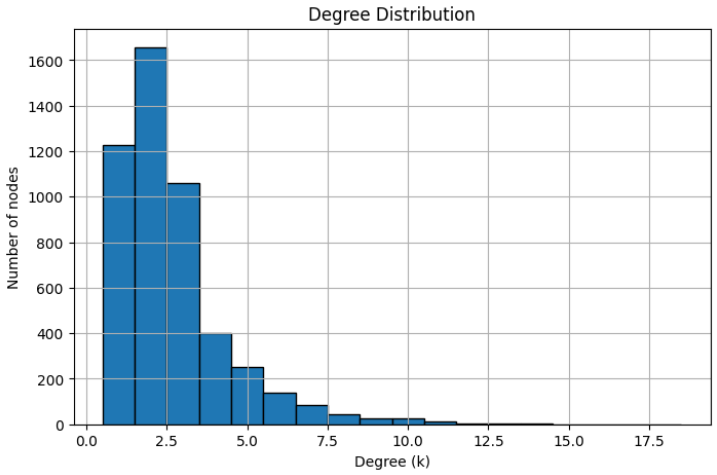

Figure 3.4: Degree distribution for the US Power Grid network.

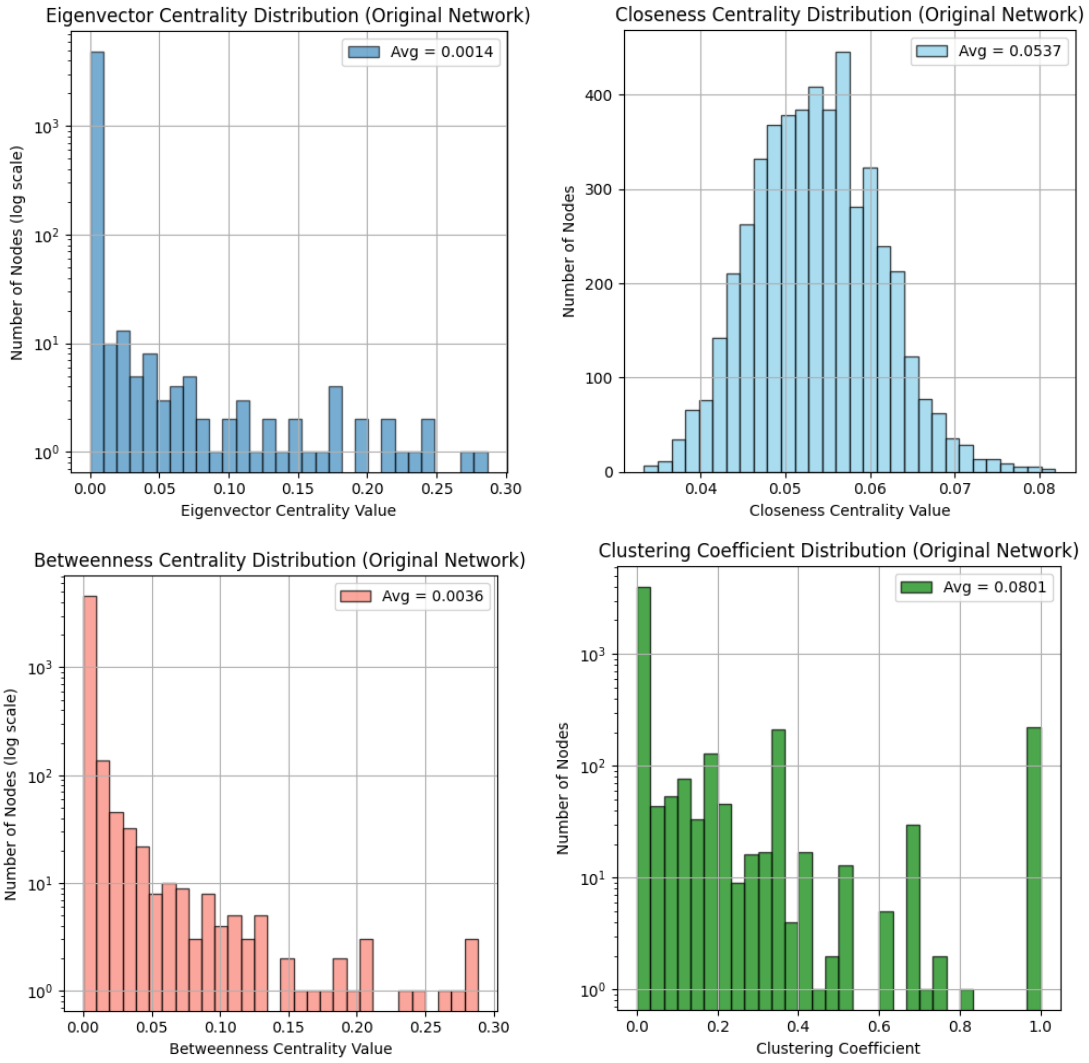

Figure 3.5: Histograms of centrality measures for the US Power Grid network.

Centrality Measure	Average Value
Closeness	0.0537
Betweenness	0.0036
Clustering Coefficient	0.0801

Table 3.4: Average centrality values for the US Power Grid network.

Configuration-Model Results

A total of 20 graphs were generated using the power grid network for the statistics below.

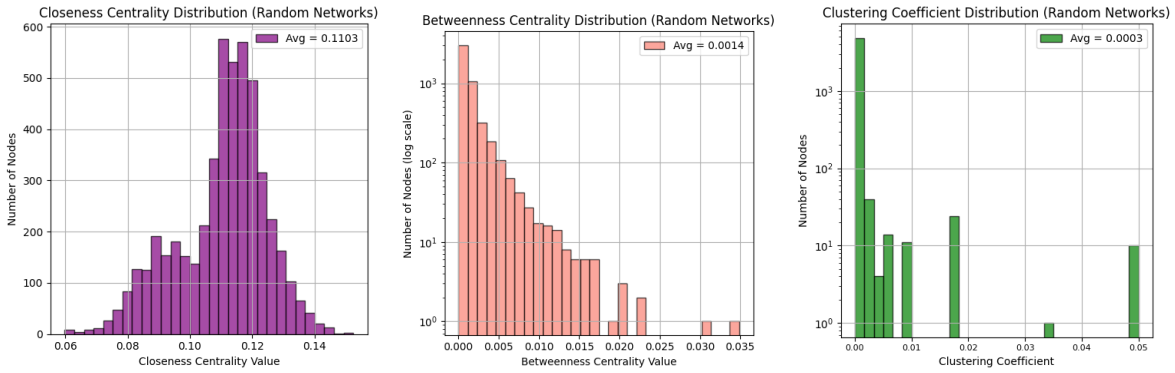

Figure 3.6: Histograms of mean centrality measures across random networks generated from the Power Grid network.

Centrality Measure	Average Value
Closeness	0.1103
Betweenness	0.0014
Clustering Coefficient	0.0003

Table 3.5: Average centrality values for the random-averaged Power Grid network.

Comparison with Orignal Network

Centrality Measure	**Avg. KS Statistic**	**Avg. JSD**
Closeness Centrality	0.9532	0.9931
Betweenness Centrality	0.1397	0.2823
Clustering Coefficient	0.1911	0.7282

Table 3.6: Average KS and JSD results for configuration model comparisons with the original network across centrality distributions.

3.1.3 GR-QC Collaboration Network

The General Relativity and Quantum Cosmology (GR-QC) collaboration network is derived from the arXiv e-print archive. Nodes represent authors of scientific papers in the GR-QC category, and an undirected edge is placed between two authors if they co-authored at least one paper. A paper with k authors thus generates a fully connected subgraph on k nodes.

The dataset spans January 1993 to April 2003 (124 months), covering essentially the entire history of the GR-QC section since shortly after the inception of the arXiv. It provides a representative example of a large-scale scientific collaboration network, commonly studied to investigate community structure, small-world effects, and the growth of scientific partnerships.

Degree and Centrality Distributions

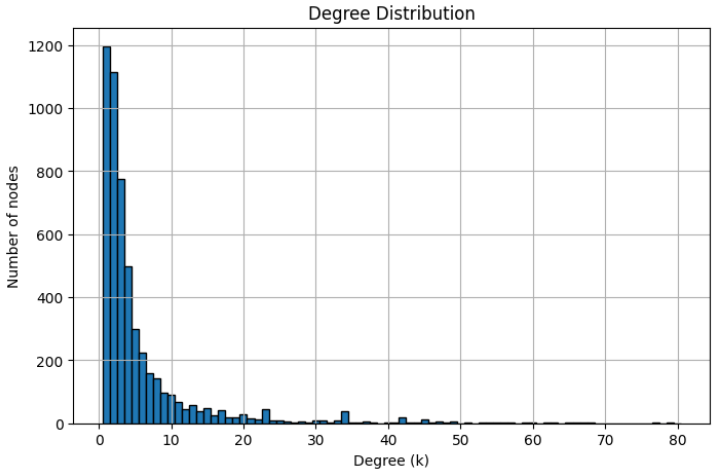

Figure 3.7: Degree distribution for the Collaboration network.

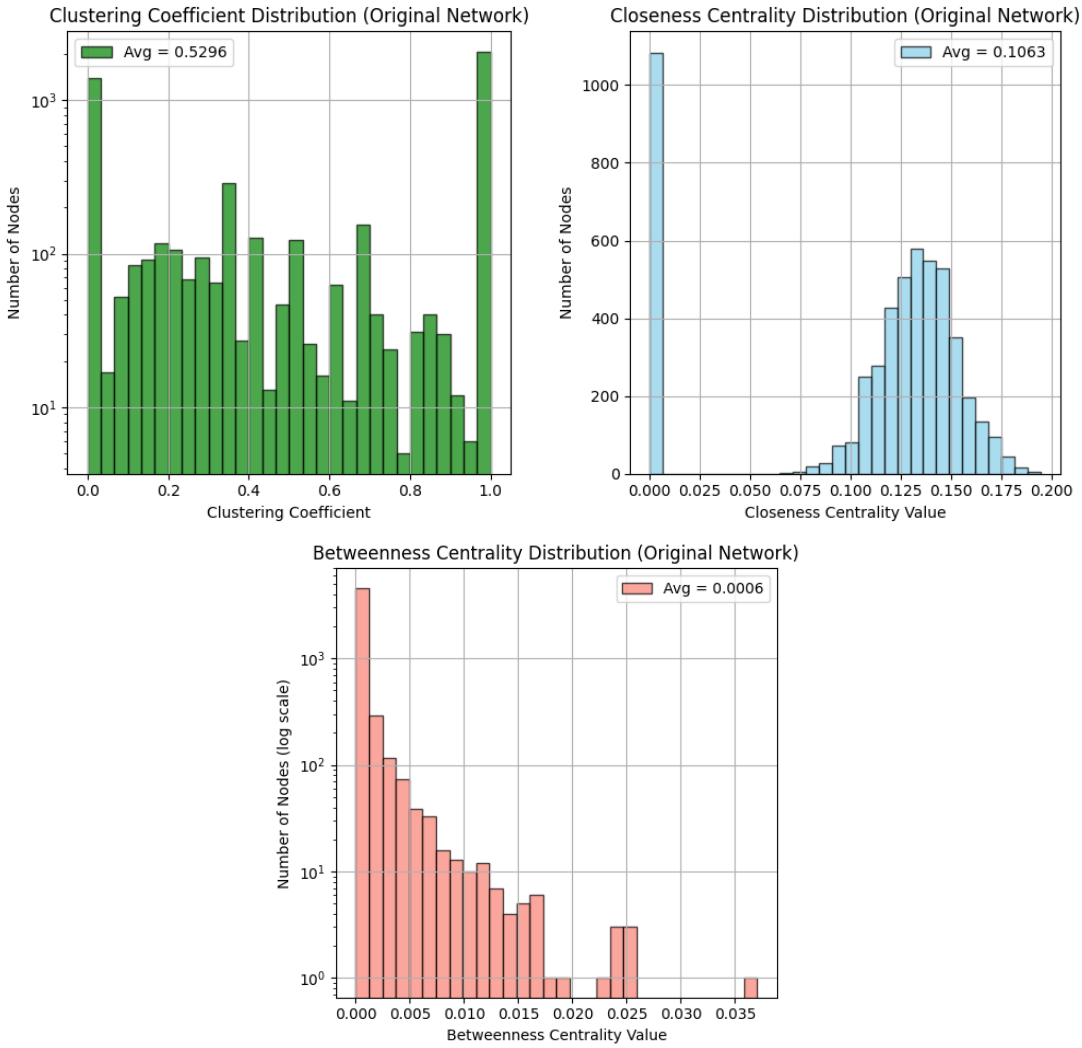

Figure 3.8: Histograms of centrality measures for the Collaboration network.

Centrality Measure	Average Value
Closeness	0.1063
Betweenness	0.0006
Clustering Coefficient	0.5296

Table 3.7: Average centrality values for the collaboration network.

Configuration-Model Results

Due to the large number of nodes in this network, only 10 graphs were generated for the statistics given below.

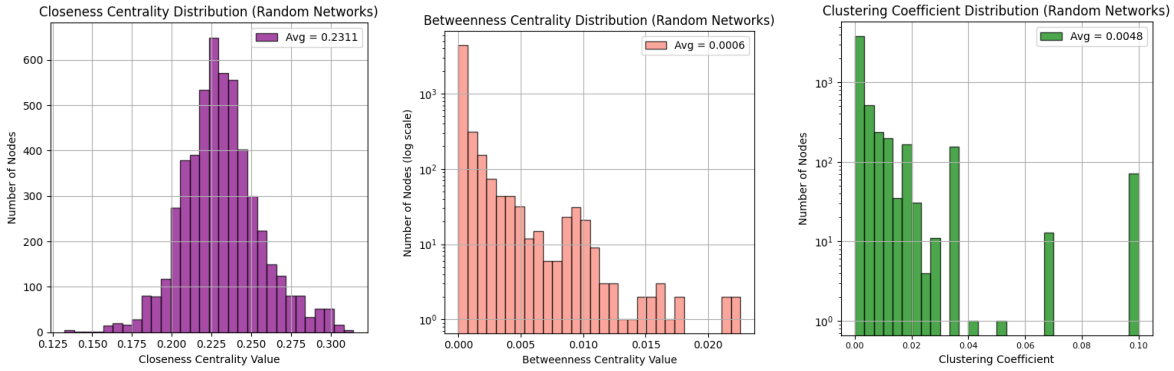

Figure 3.9: Histograms of mean centrality measures across random networks generated from the Collaboration network.

Centrality Measure	Average Value
Closeness	0.3647
Betweenness	0.0043
Clustering Coefficient	0.0132

Table 3.8: Average centrality values for the random-averaged Power Grid network.

Comparison with Orignal Network

Centrality Measure	**Avg. KS Statistic**	**Avg. JSD**
Closeness Centrality	0.9678	0.9634
Betweenness Centrality	0.4294	0.3644
Clustering Coefficient	0.7176	0.8572

Table 3.9: Average KS and JSD results for configuration model comparisons with the original network across centrality distributions.

3.1.4 TNF-α Signaling (PPI) Network

The TNF-α signaling network analyzed in this study was curated in the lab of Prof. Ganesh Viswanathan. It is a directed protein–protein interaction network representing the molecular interactions involved in TNF-α signaling. The network includes the corresponding protein identifiers for all nodes, which can be used in future work to link network structure with biological information or external datasets.

Degree and Centrality Distributions

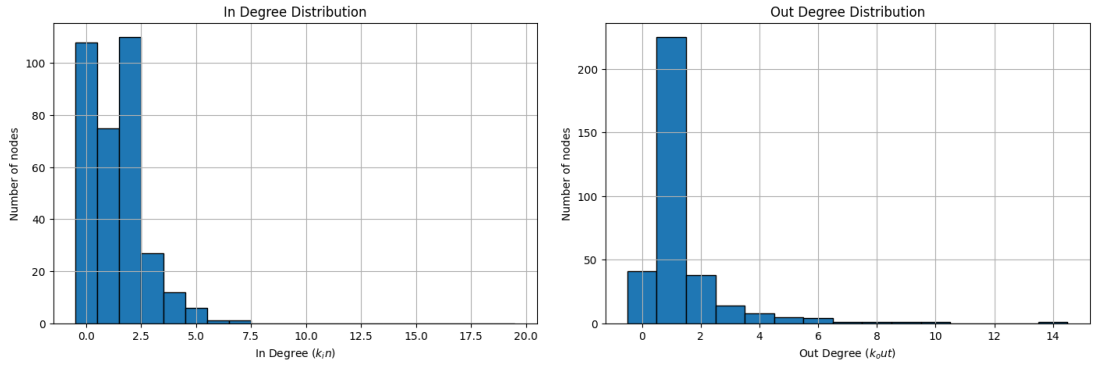

Figure 3.10: In-degree (left) and out-degree (right) distributions for the TNF-α PPI network.

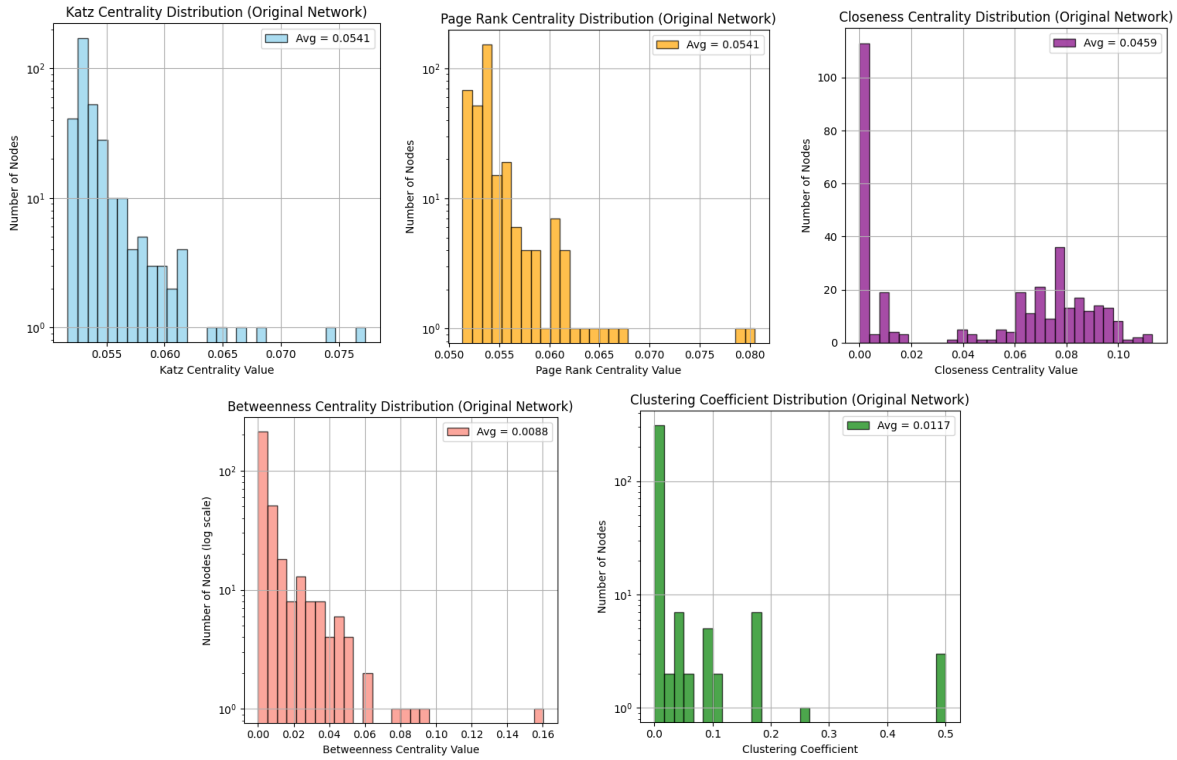

Figure 3.11: Histograms of centrality measures for the TNF-α PPI network.

Centrality Measure	Average Value
Closeness	0.0459
Betweenness	0.0088
Clustering Coefficient	0.0117

Table 3.10: Average centrality values for the TNF-α network.

Configuration-Model Results

As this network has lesser nodes compared to the others, a total of 100 random graphs were generated for the statistics given below.

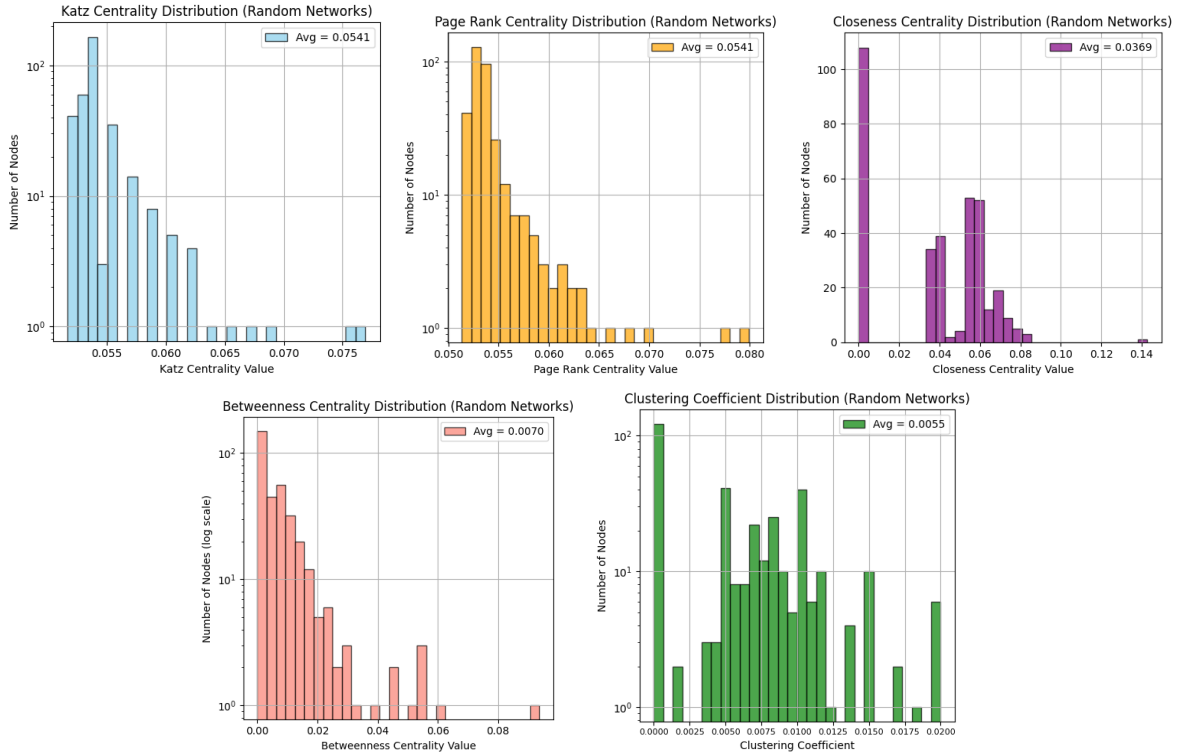

Figure 3.12: Histograms of mean centrality measures across random networks generated from the TNF-α PPI network.

Centrality Measure	Average Value
Closeness	0.0369
Betweenness	0.0070
Clustering Coefficient	0.0055

Table 3.11: Average centrality values for the random-averaged TNF-α network.

Comparison with Orignal Network

Centrality Measure	**Avg. KS Statistic**	**Avg. JSD**
Katz Centrality	0.2027	0.2738
Page Rank Centrality	0.1487	0.2072
Closeness Centrality	0.2386	0.3446
Betweenness Centrality	0.1254	0.3329
Clustering Coefficient	0.0443	0.6039

Table 3.12: Average KS and JSD results for configuration model comparisons with the original network across centrality distributions.

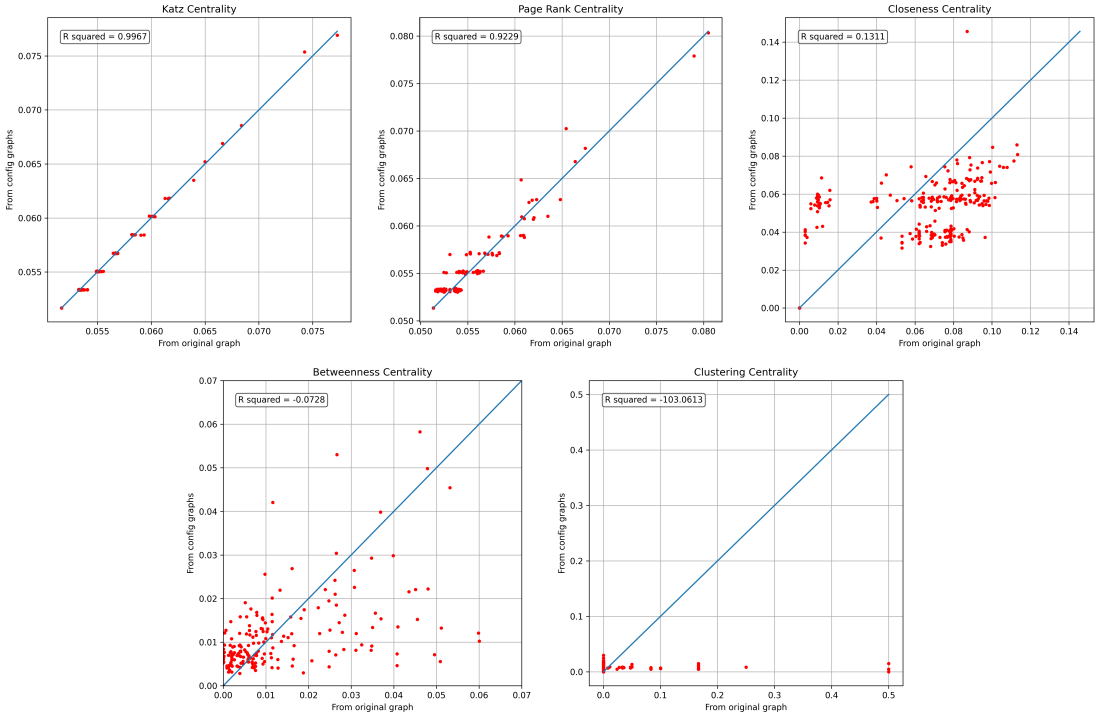

Figure 3.13: Comparison of node centrality values from the original network (x-axis) with the averaged centrality values from randomized networks (y-axis). Each point represents a node. The closer the points lie to the $y = x$ line, the stronger the agreement.

Centrality Measure	R^2 **Value**
Katz Centrality	0.9967
PageRank	0.9229
Closeness Centrality	0.1311
Betweenness Centrality	-0.0728
Clustering Centrality	-103.0613

Table 3.13: Coefficient of determination (R^2) values for the comparison between centrality values of the original network and the average centrality values of the corresponding random networks. Each R^2 is calculated against the $y = x$ line, indicating how well the original network's centrality distribution aligns with that of the randomized ensemble.

3.1.5 Analysis

Across all four networks, the comparison between the original networks and their random counterparts reveals clear differences in how structural properties emerge beyond degree sequence alone.
Overall, these findings demonstrate that:

- **Betweenness centrality** tends to be well-explained by degree sequence across all networks, with relatively low KS and JSD values.
- **Clustering coefficients** are consistently much higher in real networks than in their randomized counterparts, highlighting the tendency of real-world systems to form closely interconnected groups.

- **Closeness centrality** shows some of the strongest deviations, particularly in the Power Grid and GR-QC networks, suggesting that real-world networks organize paths differently than degree-constrained random graphs.
- **Eigenvector- and Katz-type measures** highlight system-dependent behaviors: in social and infrastructural networks they diverge strongly, whereas in the smaller TNF-α network they are largely captured by degree effects.

These results confirm that while configuration models can account for some centrality distributions—especially those heavily tied to degree, such as betweenness—they fail to reproduce features like clustering and other network patterns that emerge from real-world interactions. This highlights the value of centrality-based analysis in distinguishing degree-driven effects from structural patterns that carry real significance in the original networks.

3.2 Community Detection

We applied two popular community detection algorithms, the *Leiden* and *Louvain* methods, to the protein interaction network. Both methods partition the network into groups of nodes (communities) such that nodes are more densely connected within a community than between communities.

3.2.1 TNF-α Signaling (PPI) Network

Algorithm used	No. of Communities	Quality Function	Quality Function Value	Compilation Time(s)
Louvain	12	Modularity	0.7273	0.014
Leiden	13	Modularity	0.7279	0.002
Leiden	45	Constant Potts, $\gamma = 0.01$	324.63	0.001
Leiden	47	Constant Potts, $\gamma = 0.05$	269	0.001

Table 3.14: Community detection results for the protein interaction network.

From the results in the table, it is evident that the Leiden method has a significantly shorter computation time compared to the Louvain method for community detection. Regarding the resolution issue in modularity-based methods, the number of communities detected is much lower than that found using the CPM-based Leiden algorithm, even for very small values of the resolution parameter. This indicates that modularity tends to merge smaller communities together and may overlook finer community structure. In contrast, CPM allows for tuning of the resolution parameter γ, where increasing γ leads to the detection of a larger number of smaller communities. This flexibility makes CPM particularly valuable for uncovering multi-scale structure in networks.

However, a limitation of CPM is that it requires selecting an appropriate value for γ, and in many real-world cases, there may not be a clear estimate for this parameter. In such situations, modularity-based methods can provide a useful starting point by giving an initial partition without requiring explicit parameter tuning.

Furthermore, when both Louvain and Leiden are applied with modularity as the quality function, they yield a comparable number of communities and nearly identical modularity scores. This similarity arises because both methods optimize the same objective function (modularity), differing primarily in how they refine partitions.

Chapter 4

Summary and Future work

4.1 Conclusion

This study compared real-world networks with their degree-preserving configuration model counterparts to assess which structural features can be attributed to degree sequence alone. Our results show that some measures, such as betweenness centrality, are largely explained by degree constraints, while others, including clustering, closeness centrality, and eigenvector-based measures, reveal differences that point to additional organizing principles in the real networks.

The analysis highlights the usefulness of combining centrality measures with null models to gain a clearer picture of network structure. The configuration model, by preserving the degree distribution while randomizing other connections, provides a meaningful baseline against which to interpret centrality distributions and other metrics. Using this approach helps identify where networks deviate from degree-driven expectations and where additional structure plays a significant role.

We also observed practical differences in community detection approaches. The Constant Potts Model (CPM) offers control over the number and size of communities through its resolution parameter, which allows finer exploration of mesoscopic structure. However, it requires careful choice of this parameter, which is not always straightforward in real applications. In such cases, modularity-based approaches may serve as a reasonable first estimate of community organization before refining the resolution further with CPM.

Taken together, these results show the value of centrality-based and community-level analysis when used alongside appropriate null models. This combination not only quantifies the role of degree distribution but also points to structural differences that may be functionally relevant, laying the groundwork for more targeted investigations into network behavior.

4.2 Future Directions

While this study highlights key differences between real networks and their degree-preserving null models, several directions remain open for further investigation.

First, the centrality-based analysis can be expanded by applying the same methodology across additional network types, or by focusing more deeply on individual networks to explore how specific structural features influence dynamical processes such as diffusion, synchronization, or epidemic spreading. This would strengthen the connection between

structural observations and functional implications.

Second, the comparison of centrality distributions relied on Jensen–Shannon divergence (JSD) using bins determined by the Freedman–Diaconis rule. Since JSD is sensitive to binning, exploring alternative binning strategies could provide more robust or consistent measures of similarity between distributions.

For community detection, the current analysis showed that the protein interaction network contains strong modular organization but lacked external benchmarks for validation. Future work could compare the detected communities with known biological or functional groupings to evaluate their relevance and refine detection parameters. Similar analyses could be extended to the other networks studied here, although this was not attempted in the present work due to the lack of available ground-truth data. Such comparisons would help clarify how modular organization in these networks relates to their real-world function.

Finally, exploring a wider range of values for the Constant Potts Model (CPM) resolution parameter could give a better understanding of community structure at different scales. Developing automated or data-driven ways to choose this parameter would make CPM easier to use for networks where no estimate is available, and allow a more systematic comparison with modularity-based methods.

Acknowledgements

I would like to express my sincere gratitude to my guide, Dr. Ganesh A. Viswanathan, for his invaluable guidance, support, and encouragement throughout this project. I also thank his lab group for curating the protein–protein interaction network and providing insights that were instrumental for this study. Finally, I extend my thanks to all colleagues and peers who offered feedback and assistance during the course of this work and in the making of this report.

References

[1] Edward A. Bender and E. Rodney Canfield. The asymptotic number of labeled graphs with given degree sequences. *Journal of Combinatorial Theory, Series A*, 24(3):296–307, 1978. doi:10.1016/0097-3165(78)90059-6.

[2] Vincent D. Blondel, Jean-Loup Guillaume, Renaud Lambiotte, and Etienne Lefebvre. Fast unfolding of communities in large networks. *Journal of Statistical Mechanics: Theory and Experiment*, (10):P10008, 2008. doi:10.1088/1742-5468/2008/10/P10008.

[3] Vincent D. Blondel, Jean-Loup Guillaume, and Renaud Lambiotte. Fast unfolding of communities in large networks: 15 years later. *arXiv preprint arXiv:2311.06047*, 2023. URL: https://arxiv.org/abs/2311.06047.

[4] Santo Fortunato and Marc Barthélemy. Resolution limit in community detection. *Proceedings of the National Academy of Sciences*, 104(1):36–41, 2007. doi:10.1073/pnas.0605965104.

[5] David Freedman and Persi Diaconis. Histograms: Estimation and visualization of distributions. *Statistics: A Journal of Theoretical and Applied Statistics*, 12(1):75–84, 1981.

[6] Jure Leskovec, Jon Kleinberg, and Christos Faloutsos. Collaboration network of arxiv general relativity category. https://snap.stanford.edu/data/ca-GrQc.html, 2007.

[7] J. Lin. Divergence measures based on the shannon entropy. In *IEEE International Conference on Information Theory*, pages 1–8, 1991. doi:10.1109/ITW.1991.168764.

[8] Sergei Maslov and Kim Sneppen. Specificity and stability in topology of protein networks. *Science*, 296(5569):910–913, 2002. doi:10.1126/science.1065103.

[9] Frank J. Massey. The kolmogorov-smirnov test for goodness of fit. *Journal of the American Statistical Association*, 46(253):68–78, 1951. doi:10.1080/01621459.1951.10500769.

[10] M. E. J. Newman. *Networks: An Introduction*. Oxford University Press, Oxford, UK, 2010.

[11] M. E. J. Newman and Michelle Girvan. Finding and evaluating community structure in networks. *Physical Review E*, 69(2):026113, 2004. doi:10.1103/PhysRevE.69.026113.

[12] Mark Newman. Us power grid network data. https://public.websites.umich.edu/~mejn/netdata/.

[13] Ryan A. Rossi and Nesreen K. Ahmed. Infectious contact network - dublin. https://networkrepository.com/infect-dublin.php, 2015.

[14] Vincent A Traag, Paul Van Dooren, and Yurii Nesterov. Narrow scope for resolution-limit-free community detection. *Physical Review E*, 84(1):016114, 2011. doi:10.1103/PhysRevE.84.016114.

[15] Vincent A. Traag, Ludo Waltman, and Nees Jan van Eck. From louvain to leiden: guaranteeing well-connected communities. *Scientific Reports*, 9(1):5233, 2019. doi:10.1038/s41598-019-41695-z.