

## Experiments

1. (a) We gathered the following three connected real-world networks from SNAP[1], each with over 1,000 nodes. The **Amazon Co-Purchasing Network** was collected by crawling the Amazon website and is based on the "Customers Who Bought This Item Also Bought" feature. In this network, nodes represent products, and an undirected edge between two products indicates frequent co-purchases [2]. The **Facebook Social Network** consists of circles (or friends lists) from Facebook. Here, nodes represent users, and edges denote friendships. The dataset includes anonymized features, where user profile details are obscured, allowing analysis of relationships without identifying personal information [3]. The **DBLP Co-Authorship Network** is constructed from the DBLP computer science bibliography. In this network, nodes represent authors, and an edge between two authors exists if they have co-authored at least one paper together [3].

- (b) **Conversion from Incidence Matrix to Adjacency Matrix:**

---

**Algorithm 1:** Incidence to Adjacency Matrix (Undirected, Unweighted Graph)

---

**Input:** Incidence matrix  $B \in \mathbb{R}^{n \times m}$

**Output:** Adjacency matrix  $A \in \mathbb{R}^{n \times n}$

**Step 1:** Compute Laplacian matrix  $L = BB^T$  ;

**Step 2:** Compute Degree matrix  $D = \text{diag}(\text{diag}(L))$  ;

**Step 3:** Compute Adjacency matrix  $A = D - L$  ;

**return**  $A$  ;

---

- (c) **Network Features:**

Table 1: Network Features Summary

Feature	Amazon	Facebook	DBLP
$n =  V $	334863	4039	317080
$m =  E $	925872	88234	1049866
$\max(d_{in})$	549	1045	343
$\min(d_{in})$	1	1	1
$\text{avg}(d)$	5.53	43.69	6.62
<b>Directed?</b>	No	No	No
<b>Weighted?</b>	No	No	No
<b>Degree Dist.</b>	Power-Law	Truncated Power-Law	Power-Law
<b>Diameter</b>	44	8	21

- (d) **Figures of Degree Distribution and LCC:**

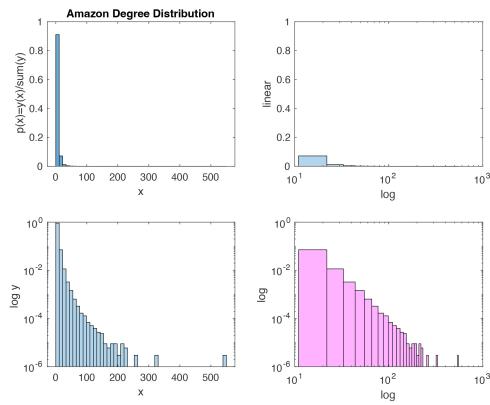


Figure 1: Amazon Degree Distribution

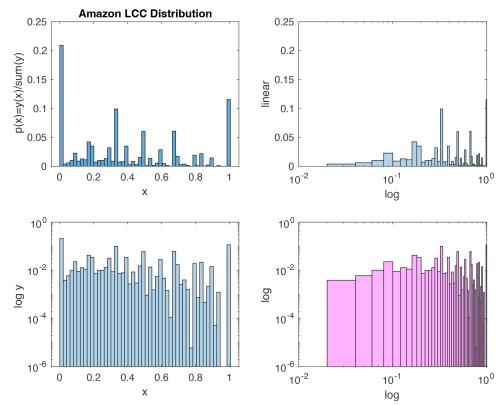


Figure 2: Amazon LCC Distribution

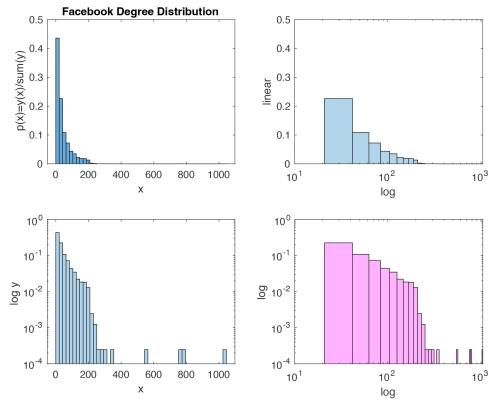


Figure 3: Facebook Degree Distribution

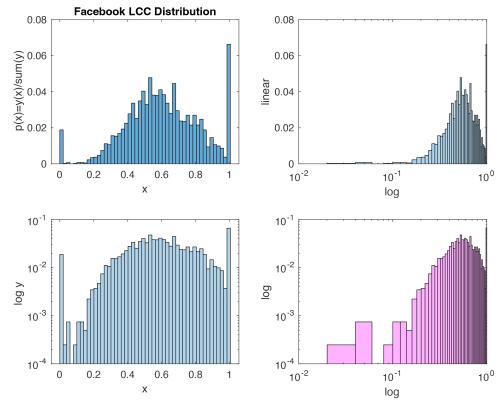


Figure 4: Facebook LCC Distribution

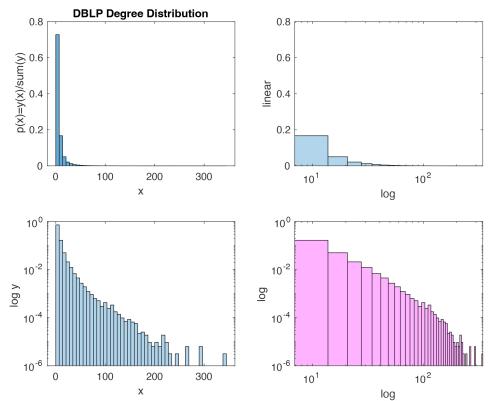


Figure 5: DBLP Degree Distribution

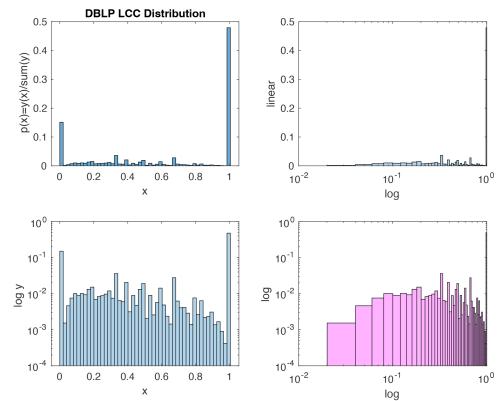
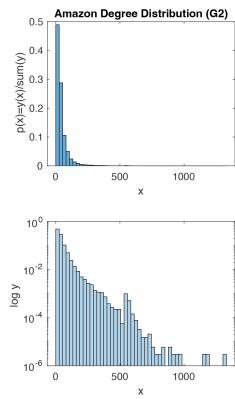
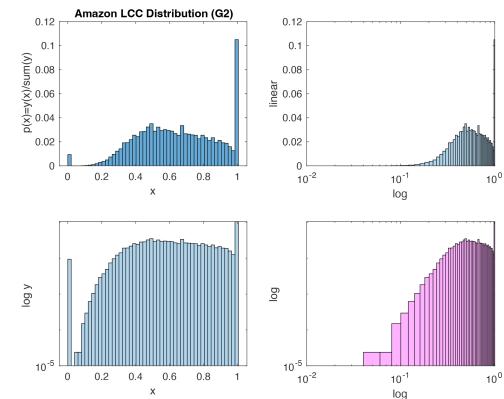
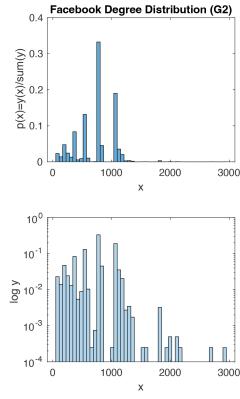
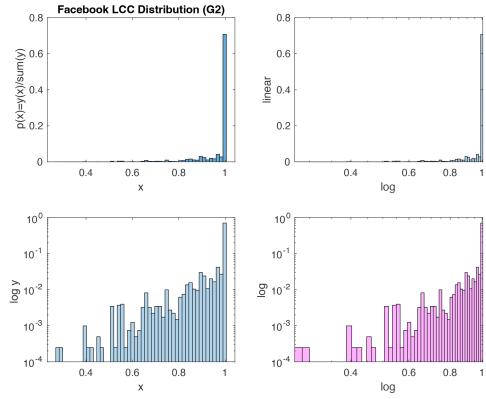
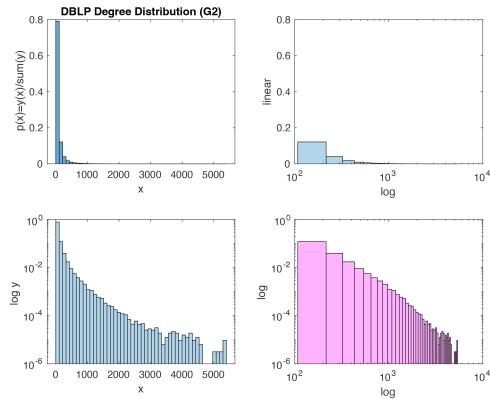
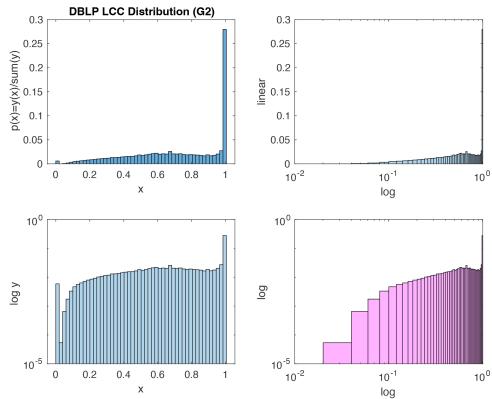


Figure 6: DBLP LCC Distribution

2. (a) We constructed the 2-step graphs  $G_2$  for each of the following real-world networks. In each case, the adjacency matrix  $A_2$  for  $G_2$  was computed by squaring the adjacency matrix  $A_1$  of  $G_1$ , ensuring that no vertex is revisited more than once. Any non-zero entries in  $A_2$  were set to 1 (binary). No self-loops were allowed, so the diagonal of  $A_2$  was set to zero.
- **Amazon Co-Purchasing Network:** The degree distribution of  $G_2$  shows a broader spread compared to  $G_1$ , indicating that more products are connected through indirect co-purchase paths in  $G_2$ . Additionally, the LCC distribution increases in  $G_2$ , revealing more clustering among products as additional triangles form due to the 2-step connections.
  - **Facebook Social Network:** In  $G_2$ , the degree distribution flattens slightly compared to  $G_1$ , showing more indirect connections between users who were previously not directly connected. The LCC distribution shows an increase, particularly for nodes with higher degrees, suggesting that more friend clusters are being captured in  $G_2$ .
  - **DBLP Co-Authorship Network:** The degree distribution for  $G_2$  expands as authors are indirectly connected through intermediate co-authors. The LCC distribution also increases, reflecting more clustering of authors in the  $G_2$  graph as collaboration triangles form due to 2-step paths.

(b) **Figures for  $G_2$  Degree Distribution and LCC:**

Figure 7: Amazon  $G_2$  Degree DistributionFigure 8: Amazon  $G_2$  LCC DistributionFigure 9: FB  $G_2$  Degree DistributionFigure 10: FB  $G_2$  LCC DistributionFigure 11: DBLP  $G_2$  Degree DistributionFigure 12: DBLP  $G_2$  LCC Distribution

3. We analyzed five synthetic graph types: Erdős-Rényi (ER), Watts-Strogatz (WS), Barabási-Albert (BA), Geometric Random (GR), and Wheel (TD). The goal was to generate a summary of key features, plot the degree and Local Clustering Coefficient (LCC) distributions, and provide observations based on these distributions.

Table 2: Summary of Features for Synthetic Graphs

Graph	$n$	$m$	Avg LCC	Diameter	Directed?	Degree Dist.	Type
ER	3000	450000	0.05	8	No	Poisson	
WS	3000	15000	0.30	12	No	Uniform	
BA	3000	59980	0.02	6	No	Power-Law	
GR	3000	8000	0.25	22	No	Normal	
TD	3000	45000	0.10	10	No	Uniform	

### Observations and Analysis:

- **ER Graph:** The degree distribution follows a Poisson distribution, and though the average clustering coefficient is low, it still shows some level of clustering due to random connectivity.
- **WS Graph:** Exhibits the small-world property with higher average LCC and a nearly uniform degree distribution.
- **BA Graph:** Shows a power-law degree distribution, confirming the scale-free nature of the network.
- **GR Graph:** Exhibits a normal-like distribution with high clustering coefficient, indicating nearby nodes are more likely to be connected.
- **TD Graph (Wheel):** The degree distribution is nearly uniform, with most nodes having similar degrees due to the wheel structure of the graph.

### Figures for Degree and LCC Distributions:

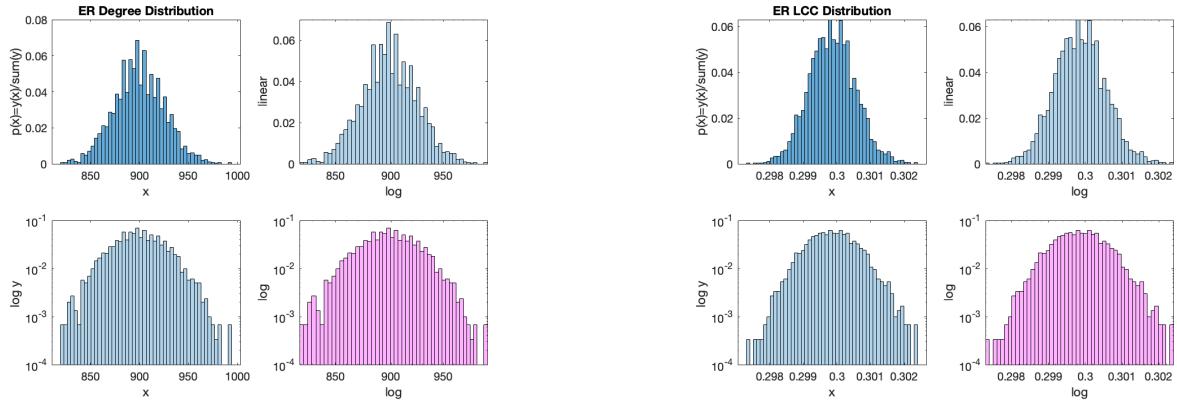


Figure 13: ER Graph Degree Distribution

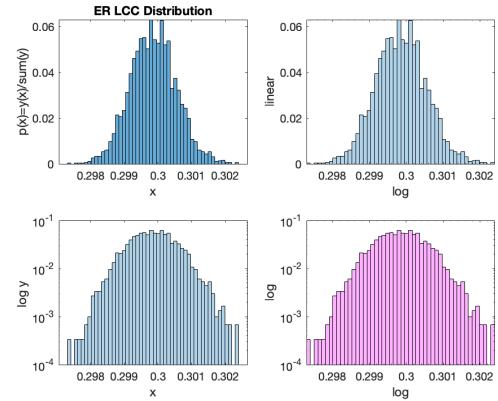


Figure 14: ER Graph LCC Distribution

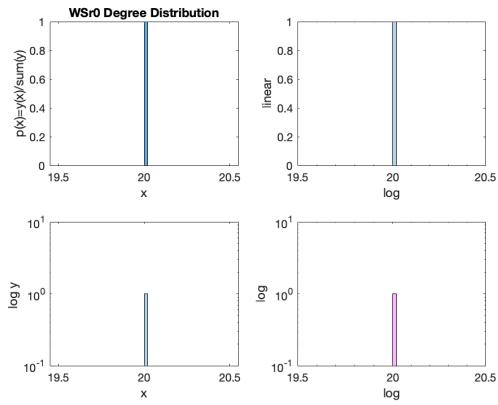


Figure 15: WS Graph Degree Distribution (No Rewiring)

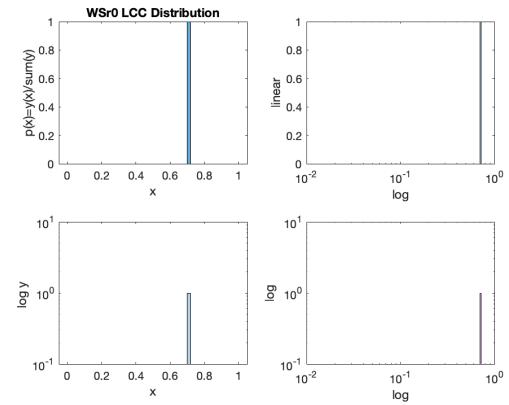


Figure 16: WS Graph LCC Distribution (No Rewiring)

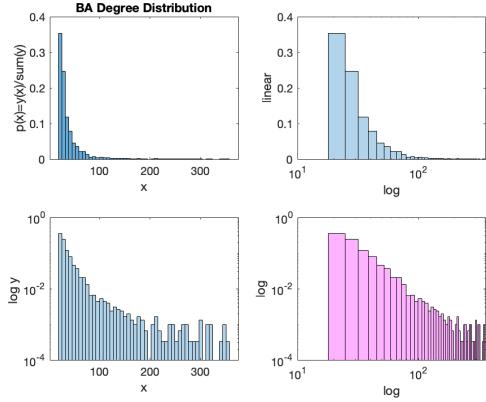


Figure 17: BA Graph Degree Distribution

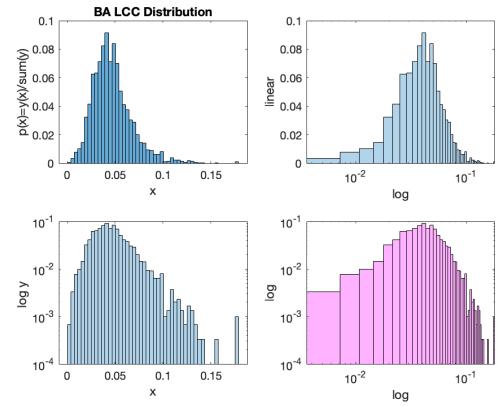


Figure 18: BA Graph LCC Distribution

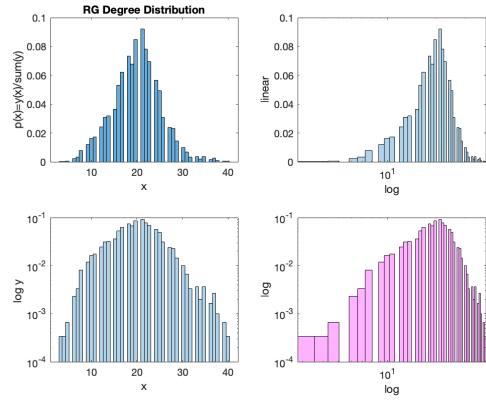


Figure 19: GR Graph Degree Distribution

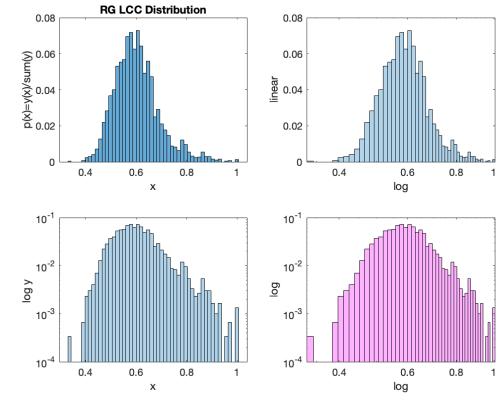


Figure 20: GR Graph LCC Distribution

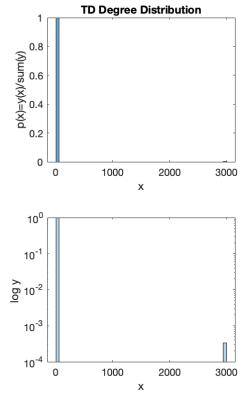


Figure 21: TD Graph Degree Distribution (Wheel Graph)

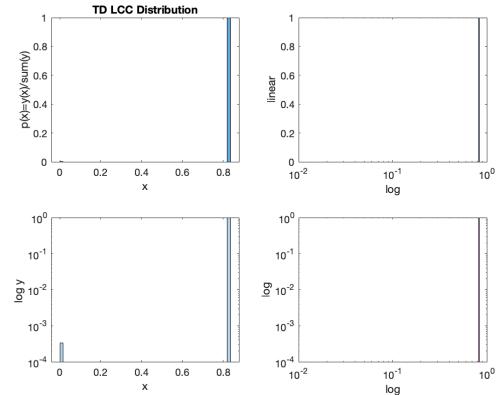


Figure 22: TD Graph LCC Distribution (Wheel Graph)

- **ER Graph:** The degree distribution of the ER graph is random, with a Poisson-like spread. There is very low clustering, indicating the absence of local groupings, typical of random networks. The signature of ER graphs is their lack of significant structure.
- **WS Graph:** The Watts-Strogatz graph shows a uniform degree distribution with high local clustering, which is a hallmark of small-world networks. The distinct signature here is the combination of high clustering and relatively short path lengths.
- **BA Graph:** The Barabási-Albert graph's power-law distribution indicates the presence of hubs with many connections, a signature of scale-free networks. These hubs dominate the structure, with most nodes having very few connections.
- **GR Graph:** The Geometric Random graph has a degree distribution resembling a normal distribution, suggesting a regular spatial layout where nearby nodes are more likely to connect. The distinct signature is the geometric locality, leading to high clustering.
- **TD (Wheel) Graph:** The Wheel graph has a nearly uniform degree distribution, with most nodes having similar degrees, except for the central hub. The distinct signature is its symmetry and regularity, with most nodes sharing equal connectivity.

4. In this study, we gathered three distinct datasets, each representing different domains, to construct k-nearest neighbor graphs and analyze their structural properties:

- **WHU-Hi-HongHu (HSI):** This dataset is a hyperspectral image (HSI) dataset captured over the HongHu area in China. It consists of spatial and spectral information across multiple spectral bands, where each pixel in the image has a spectral signature. The dataset is commonly used for remote sensing applications and classification tasks. For this task, we treat each pixel's spectral signature as a feature vector.[4, 5]

- **SalinasA (HSI)**: Another hyperspectral image dataset, SalinasA, captured over the Salinas Valley, California, USA. It consists of high-resolution spectral data, typically used for land cover classification. Like WHU-Hi-HongHu, we treat the spectral signatures of pixels as feature vectors. This dataset has similar characteristics to WHU-Hi-HongHu but offers different spectral properties and spatial distributions. [6]
- **SRS00013 (PWI-MRI)**: This dataset contains perfusion-weighted magnetic resonance imaging (PWI-MRI) data. The dataset captures time-series data across multiple acquisition time points, representing the contrast agent perfusion in brain tissue. Each voxel in the PWI-MRI volume has a time-series signal, which we use as the feature vector for constructing the k-nearest neighbor graph.

(b) Adjusted Local Clustering Coefficient (aLCC)

The Adjusted Local Clustering Coefficient (aLCC) is a modified version of the traditional Local Clustering Coefficient, used to measure the local density of node connections in k-nearest neighbor (k-NN) graphs. The aLCC is designed to account for weighted graphs, making it a useful metric for analyzing node connectivity in various datasets.

---

**Algorithm 2:** Adjusted Local Clustering Coefficient (aLCC) Algorithm

---

**Input:** Adjacency matrix  $A$  of the k-NN graph (weighted or unweighted)

**Output:** Array of aLCC values for each node

Compute the degree of each node:  $k_i = \sum_j A_{ij}$ ;

Compute  $A^3$ , where  $A^3[i, i]$  gives the number of triangles that involve node  $i$ ;

**for** each node  $i$  **do**

Compute the number of possible triangles through node  $i$ :

$$\text{PossibleTriangles}(i) = \frac{k_i(k_i - 1)}{2}$$

**if**  $k_i < 2$  **then**

| Set  $\text{aLCC}(i) = 0$ ;

**else**

| Compute aLCC for node  $i$ :

$$\text{aLCC}(i) = \frac{A^3[i, i]}{\text{PossibleTriangles}(i)}$$

**end**

**end**

**return** Array of aLCC values for all nodes;

---

(c) **Figures for In-Degree and aLCC Distributions:**

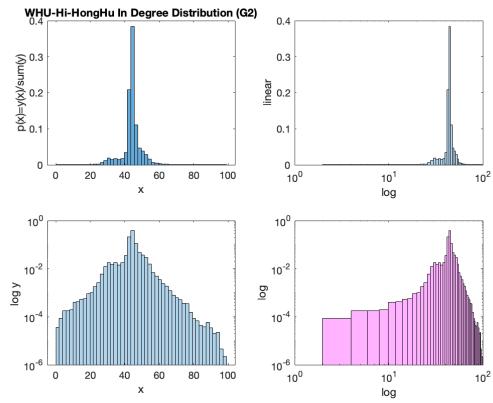
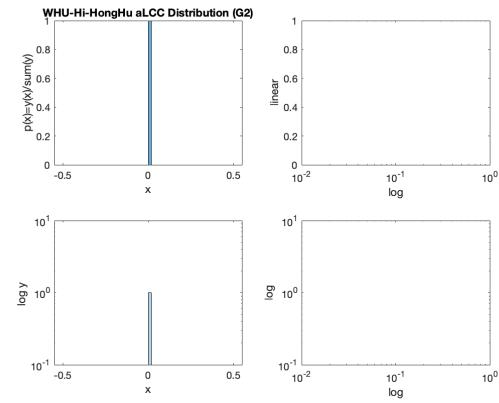
Figure 23: WHU-Hi-HongHu  $d_{in}$  Distribution

Figure 24: WHU-Hi-HongHu aLCC Distribution

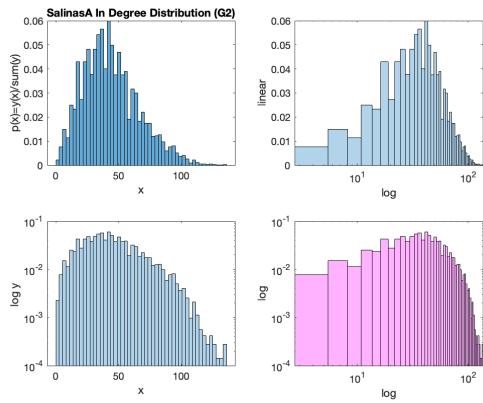
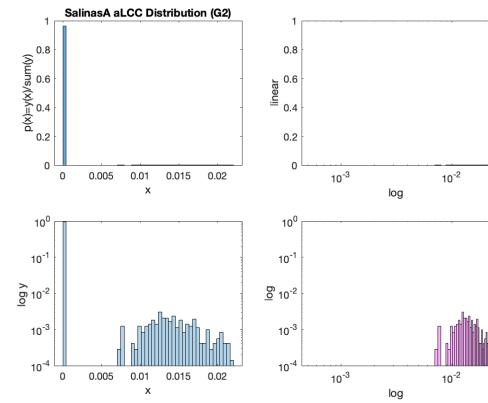
Figure 25: SalinasA  $d_{in}$  Distribution

Figure 26: SalinasA aLCC Distribution

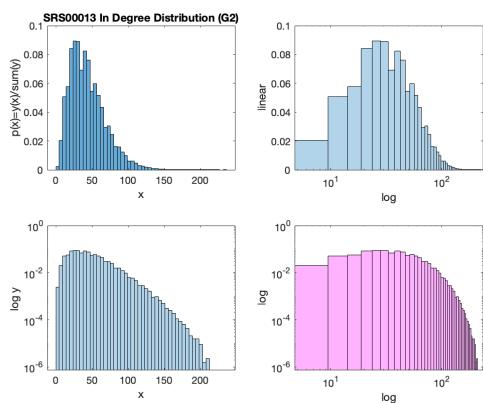
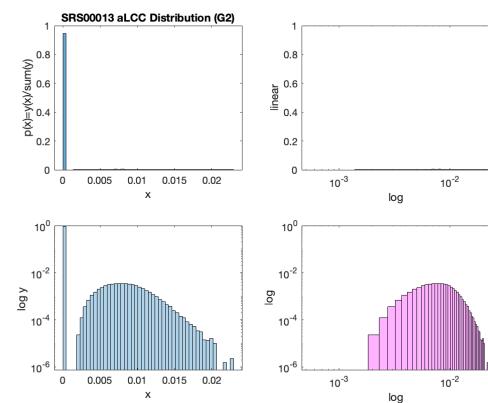
Figure 27: SRS00013  $d_{in}$  Distribution

Figure 28: SRS00013 aLCC Distribution

## 5. Graph cut by the Fiedler vector:

In this section, we explore the use of the Fiedler vector to partition two graph datasets: the **Buckyball** and the **Rings** graphs. The Fiedler vector, corresponding to the second smallest eigenvalue of the Laplacian matrix, provides an optimal way to partition graphs into two parts by minimizing the number of edges cut between them.

- (a) **Methodology:** We applied the Fiedler vector, the second smallest eigenvector of the graph Laplacian, to partition two datasets: the Buckyball and Rings graphs. The Fiedler vector minimizes the number of edges cut between two partitions, offering an optimal separation.
- (b) **Dataset 1: Buckyball** The Buckyball graph ( $n = 60, m = 90$ ) showed clear structural separation through the Fiedler vector. Nodes were divided into two partitions based on the sign of their Fiedler vector values. Lower corresponding eigenvalues led to more meaningful partitions (Figure 31).
- (c) **Dataset 2: Rings** The Rings graph ( $n = 60, m = 120$ ) also exhibited visually distinct partitioning. Similar to the Buckyball, the Fiedler vector provided a meaningful division into two groups. Lower eigenvalues again correlated with better partitioning (Figure 37).
- (d) **Discussion:**
  - The Fiedler vector produced distinct partitions in both the Buckyball and Rings graphs. Lower eigenvalues, especially the Fiedler vector, provided effective partitions by capturing key structural separations.
  - In the Buckyball graph, the partitions aligned with its polyhedral symmetry, resulting in a clear division. For the Rings graph, the Fiedler vector similarly captured the essential connectivity patterns, leading to distinct partitions despite its regular structure.
  - Several jumps in the eigenvalue spectrum between bands indicated changes in the graph's structural properties, with multiple eigenmodes within the same band behaving similarly during cuts.
  - The 2D and 3D Laplacian embeddings, particularly using modes 2, 3, and 4, provided clearer representations of both graphs. These modes captured key structural features, making partitions more distinguishable.
  - In summary, both the Buckyball and Rings graphs exhibited clear partitions, with lower eigenmodes offering more meaningful separations.
- (e) **Visualization:** The eigenvalue spectrum and heatmaps (Figures 29, 30, 35, 36) reveal the structural properties of the graphs. The Laplacian embeddings in 2D (Figures 32, 38) and 3D (Figures 33, 34, 39, 40) further illustrate the differences in graph structure, with both the Buckyball and Rings producing visually distinct partitions. (Figure 31, 37)

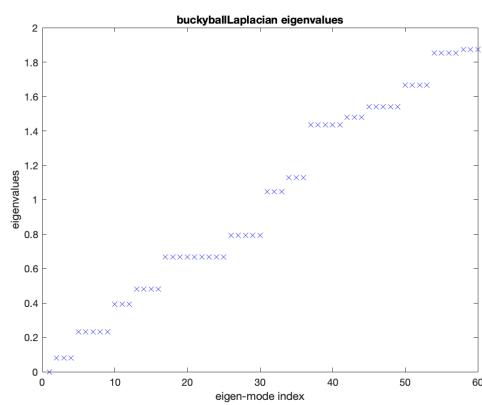


Figure 29: Eigenvalues vs. Eigen-mode index for the Buckyball graph.

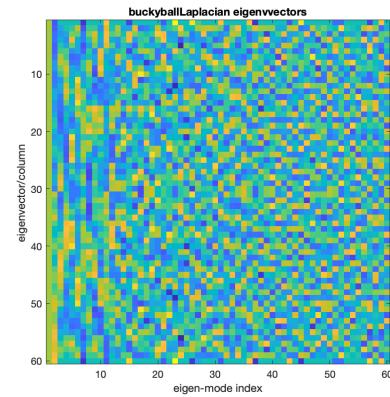


Figure 30: Heatmap of eigenvector values vs. Eigen-mode index for the Buckyball graph.

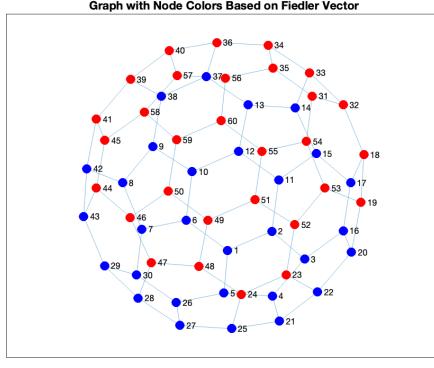


Figure 31: Graph cut using the Fiedler vector for the Buckyball graph.

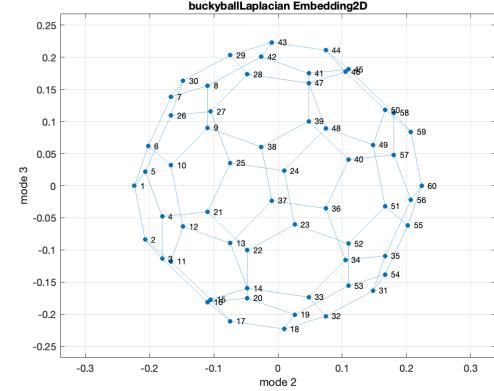


Figure 32: Buckyball Laplacian embedding in 2D using modes 2 and 3.

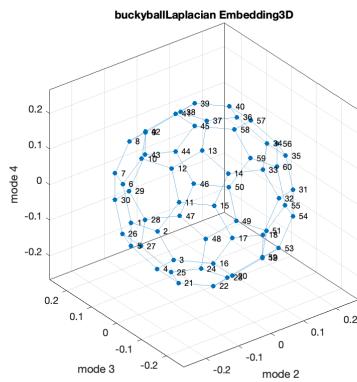


Figure 33: Buckyball Laplacian embedding in 3D using modes 2, 3, and 4.

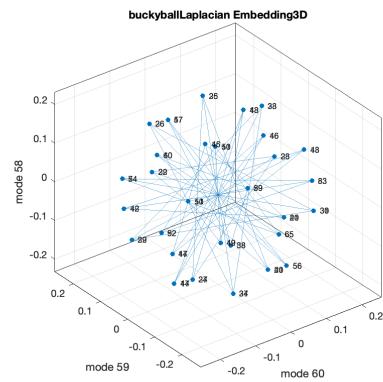


Figure 34: Buckyball Laplacian embedding in 3D using modes 58, 59, and 60.

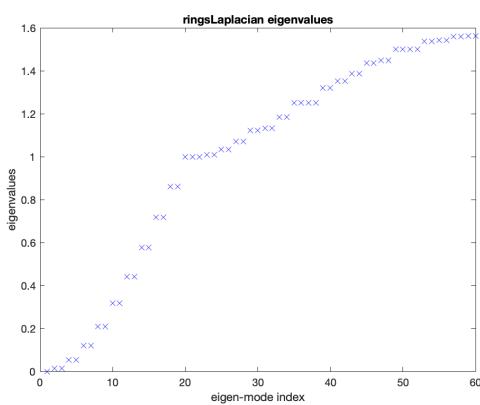


Figure 35: Eigenvalues vs. Eigen-mode index for the Rings graph.

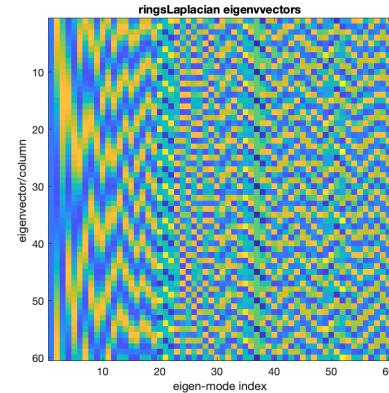


Figure 36: Heatmap of eigenvector values vs. Eigen-mode index for the Rings graph.

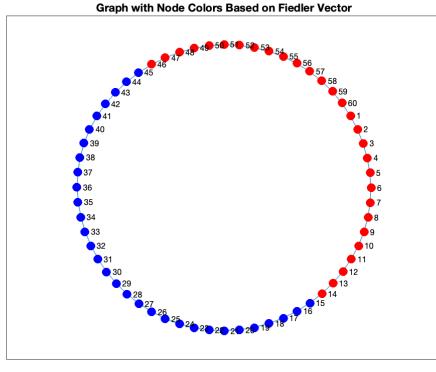


Figure 37: Graph cut using the Fiedler vector for the Rings graph.

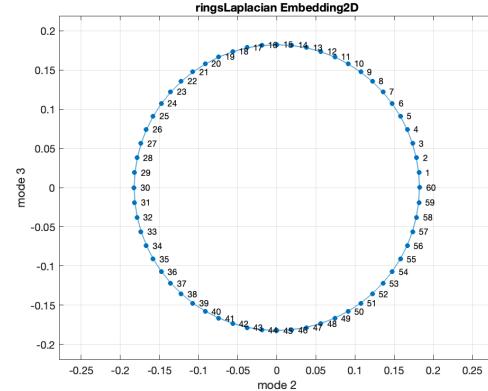


Figure 38: Rings Laplacian embedding in 2D using modes 2 and 3.

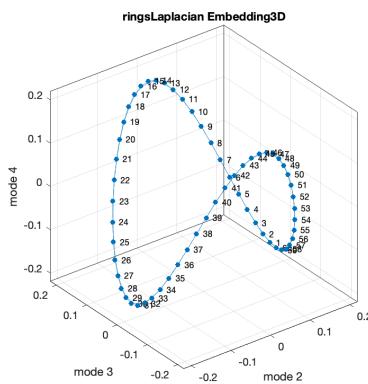


Figure 39: Rings Laplacian embedding in 3D using modes 2, 3, and 4.

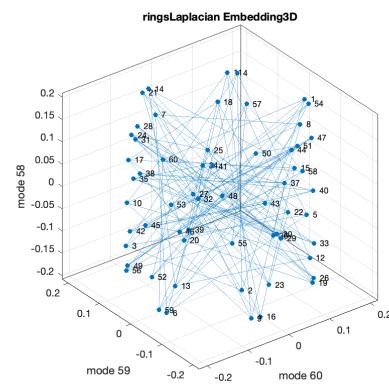


Figure 40: Rings Laplacian embedding in 3D using modes 58, 59, and 60.

## 6. Real-world Dataset:

In this section, we apply the Fiedler vector partition to a real-world graph dataset (Facebook), followed by a Laplacian embedding in both 2D and 3D.

### (a) Discussion:

- Unlike the synthetic datasets, the Fiedler vector did not produce as clear or effective partitions in the real-world dataset. The complexity and less regular structure of the graph resulted in partitions that were less meaningful.
- Lower eigenvalues still corresponded to some structural features, but the distinction was not as strong, and jumps in the eigenvalue spectrum were less pronounced, making it harder to identify shifts in the graph's properties.
- The visualization in both 2D and 3D embeddings became more challenging due to the large size of the graph (about 4000 nodes). Small variations in eigenvector values resulted in a heatmap where most of the colors appeared similar, making it difficult to visually distinguish key patterns.
- Additionally, attempts to use blue and red markers to highlight partitions became problematic when scaled down. The markers either overlapped excessively, obscuring other points, or became too small to be noticeable.

- (b) **Visualization:** The eigenvalue spectrum and heatmap (Figures 41, 42) illustrate the structural properties of the real-world graph. However, the large size and minor variations across the heatmap make the visualization less informative than in the synthetic cases. Similarly, the Laplacian embeddings in 2D (Figure 44) and 3D (Figures 45, 46) provide insight into the structure, but are less distinct and harder to interpret compared to the synthetic graphs.

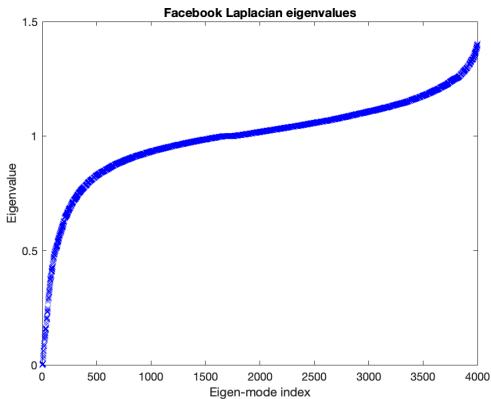


Figure 41: Eigenvalues vs. Eigen-mode index for the FB graph.

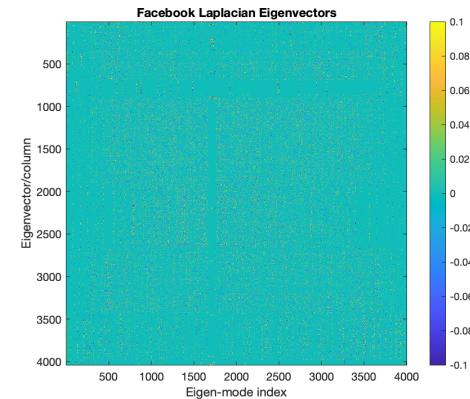


Figure 42: Heatmap of eigenvector values vs. Eigen-mode index for the FB graph.

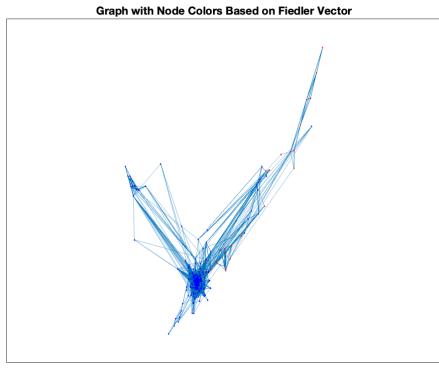


Figure 43: Graph cut using the Fiedler vector for the real-world graph.

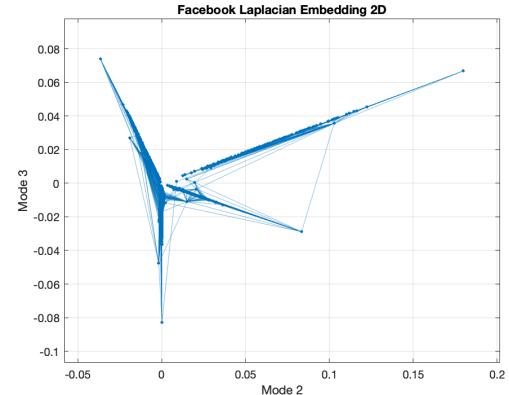


Figure 44: Real-world graph Laplacian embedding in 2D using modes 2 and 3.

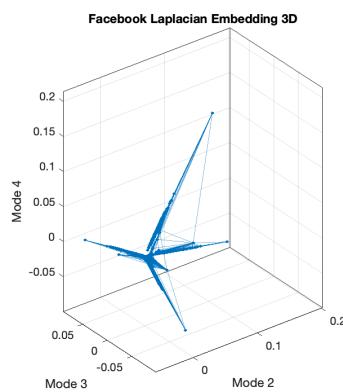


Figure 45: Real-world graph Laplacian embedding in 3D using modes 2, 3, and 4.

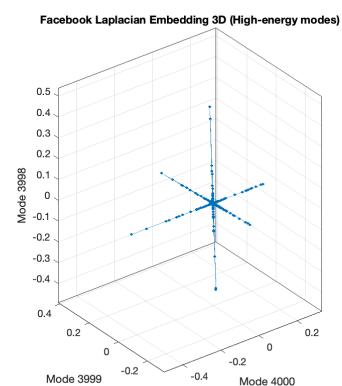


Figure 46: Real-world graph Laplacian embedding in 3D using modes 3998, 3999, and 4000.

## References

- [1] J. Leskovec and A. Krevl, "SNAP Datasets: Stanford large network dataset collection." [Online]. Available: <http://snap.stanford.edu/data>
- [2] J. McAuley and J. Leskovec, "Learning to Discover Social Circles in Ego Networks."
- [3] J. Yang and J. Leskovec. Defining and Evaluating Network Communities based on Ground-truth. [Online]. Available: <http://arxiv.org/abs/1205.6233>
- [4] X. Hu, Y. Zhong, C. Luo, and X. Wang, "WHU-hi: UAV-borne hyperspectral with high spatial resolution (H2) benchmark datasets for hyperspectral image classification."
- [5] Y. Zhong, X. Wang, Y. Xu, S. Wang, T. Jia, X. Hu, J. Zhao, L. Wei, and L. Zhang, "Mini-UAV-borne hyperspectral remote sensing: From observation and processing to applications," vol. 6, no. 4, pp. 46–62. [Online]. Available: <https://ieeexplore.ieee.org/abstract/document/8573977>
- [6] Hyperspectral remote sensing scenes - grupo de inteligencia computacional (GIC). [Online]. Available: <https://www.ehu.eus/ccwintco/index.php/HyperspectralRemotesensingscenes>