Social and Graph Data Management: Degree Correlations

Pablo Mollá Chárlez

Contents

1 Introduction to Degree Correlations in Networks

Building upon our discussion of hubs and the preferential attachment model, which explains how new nodes are more likely to connect to existing high-degree hubs, we now turn our attention to degree correlations within networks. Specifically, seek to understand whether this tendency for hubs to attract connections is a universal behavior across different types of networks or if nodes tend to connect to others with similar degrees.

In social networks, for example, it is commonly observed that hubs (such as celebrities or influential individuals) tend to connect with other hubs, creating tightly-knit clusters of high-degree nodes. This phenomenon, often referred to as "hubs connecting to hubs", reflects the social dynamics where prominent individuals interact within their elite circles. On the other hand, in other types of networks like protein interaction networks, hubs are more likely to connect to nodes with significantly lower degrees. This indicates a different organizational principle where highly connected proteins interact with many less connected partners, possibly reflecting functional specialization.

Understanding these degree correlations is crucial as they reveal **underlying structural patterns and interaction rules that govern various real-world networks**. By examining whether hubs preferentially connect to other hubs or to low-degree nodes, we gain deeper insights into the complexity and functionality of diverse networked systems.

2 Probability of Connection in Random Networks

In a random network model, the likelihood that two nodes with degrees k and k' are connected is approximated by the formula:

$$p_{k,k'} \approx \frac{k \cdot k'}{2L}$$

where L is the total number of links in the network. This equation implies that nodes with higher degrees (hubs) are more likely to form connections with other high-degree nodes simply due to their abundance of connections. However, contrary to the random model's prediction, protein-interaction networks exhibit a different pattern. In these networks, high-degree nodes (proteins that interact with many others) are more likely to connect to low-degree nodes rather than other high-degree nodes. This indicates a preference for hubs to associate with less connected nodes, deviating from the random expectation.

3 Neutrality, Assortativity and Disassortativity in Networks

Based on how hubs connect within a network, networks can be categorized into three types:

1. **Neutral Networks** $\boxed{?}$
   - Description: The connections between nodes occur randomly.
   - Characteristics: The frequency of links between hubs aligns with the probability predicted by the random model ($p_{k,k'} \approx \frac{k \cdot k'}{2L}$).
   - **Example**: Erdős–Rényi (ER) random graph model. In an ER graph, each pair of nodes has an equal probability of being connected, independent of their degrees. This means that connections between high-degree nodes (hubs) occur purely by chance.
2. **Assortative Networks** $\boxed{=}$
   - Description: Nodes tend to connect to other nodes with similar degrees.
   - Characteristics: High-degree nodes (hubs) are more likely to connect with other high-degree nodes, forming tightly-knit clusters of similar connectivity.
   - **Example**: Social networks where celebrities or influential individuals frequently interact with one another.

3. **Disassortative Networks** $\boxed{\neq}$

- Description: Nodes tend to connect to others with different degrees.
- Characteristics: High-degree nodes preferentially connect to low-degree nodes, avoiding connections with other hubs.
- **Example**: Protein-interaction networks where highly connected proteins interact with many less connected proteins.

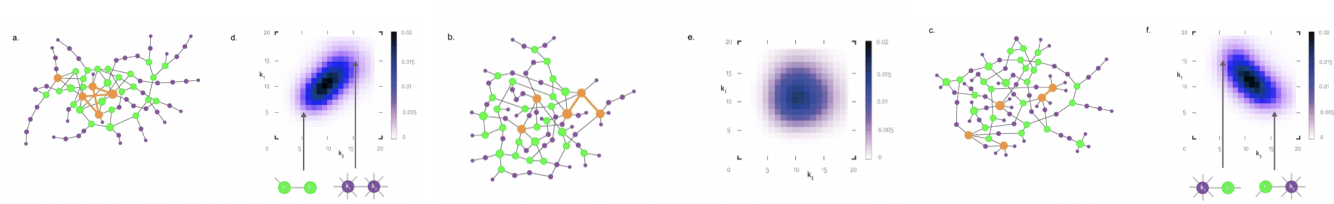

Figure 1: Assortative, Neutral and Disassortative Networks

Furthermore, we define **Perfect assortativity** when every node connects exclusively to other nodes with the same degree. In such a network, there is complete similarity in the degrees of connected nodes, meaning high-degree nodes only link to other high-degree nodes, and low-degree nodes only link to other low-degree nodes. This results in maximum positive degree correlations.

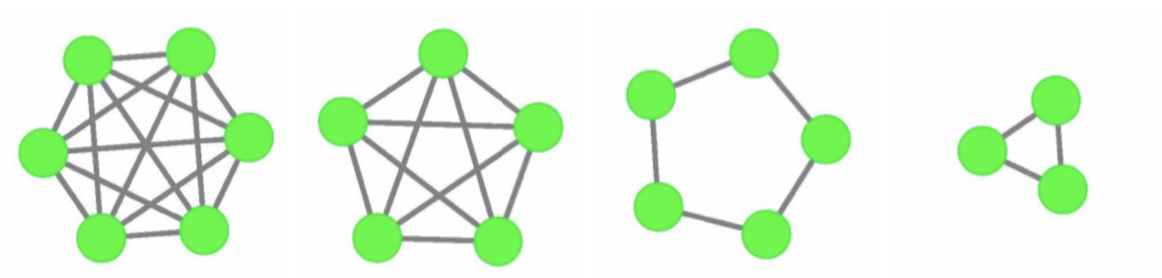

Figure 2: Perfect Assortativity

3.1 Degree Correlation Matrix

The degree correlation matrix e captures the probabilities of connections between nodes of different degrees. Specifically, e_{ij} represents the probability that a node of degree i is connected to a node of degree j. In a random network, the probability that a randomly selected link connects a node of degree k to a node of degree k' is given by:

$$e_{ij}^{random} = q_i \cdot q_j, q_k = \frac{k \cdot p_k}{\langle k \rangle}$$

where q_k is the probability that a link points to a node of degree k, p_k is the degree distribution, and $\langle k \rangle$ is the average degree. A network exhibits degree correlations if its degree correlation matrix e deviates from the random expectation $e_{ij} = q_i \cdot q_j$. This deviation indicates non-random mixing patterns, such as **assortative** or **disassortative** mixing. The actual degree correlation matrix e_{ij} quantifies the probability that a randomly selected edge in a network connects a node of degree i to a node of degree j. This matrix captures the empirical degree correlations present in the network, allowing us to analyze patterns such as assortative or disassortative mixing. For an **undirected network**, the actual degree correlation matrix e_{ij} is defined as:

$$e_{ij} = \frac{\text{Number of edges between degree } i \text{ and degree } j \text{ nodes}}{\text{Total number of edges } L}$$

3.2 Degree Correlation Function

The degree correlation function $k_{\text{nn}}(k)$ measures the average degree of the neighbors of nodes with degree k. It is defined as:

$$k_{\text{nn}}(k) = \sum_{k'} k' \cdot P(k'|k)$$

where $P(k'|k)$ is the conditional probability that a node of degree k is connected to a node of degree k'. For the 3 types of previously defined networks, we would have:

- **Neutral Networks**: Here, $k_{\text{nn}}(k)$ is independent of k, indicating no degree correlations: $k_{\text{nn}}^{random}(k) = \frac{\langle k^2 \rangle}{\langle k \rangle}$
- **Assortative Networks**: $k_{\text{nn}}(k)$ increases with k, meaning high-degree nodes tend to connect to other high-degree nodes.
- **Disassortative Networks**: $k_{\text{nn}}(k)$ decreases with k, indicating that high-degree nodes prefer connecting to low-degree nodes.

3.3 Comparing Matrix Correlation and Degree Correlation Function

While the degree correlation matrix e provides a detailed view of the connection probabilities between all pairs of degrees, it is often cumbersome to work with due to its complexity, especially in large networks. Analyzing e requires handling a potentially large matrix, making it less practical for quick assessments or large-scale studies. In contrast, the degree correlation function $k_{\text{nn}}(k)$ offers a more streamlined and intuitive measure of degree correlations by summarizing the average neighbor degree as a function of a node's degree. This makes it easier to identify trends such as assortativity or disassortativity without delving into the full matrix.

4 Numerical Example: Degree Correlation Matrix & Correlation Function

To illustrate both the degree correlation matrix and the degree correlation function, let's consider a simple graph with **4 nodes** where the **nodes** are A, B, C, D and the **edges** are $A - B, A - C, A - D, B - C$.

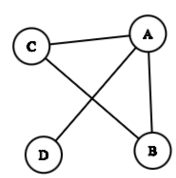

Figure 3: Simple Graph with 4 Nodes

1. $\boxed{\textbf{Random Network Expectation } e_{ij}^{random}}$ First, we determine the **Degree Distribution**: **Node** A has a degree of 3 (connected to B, C, D), **Node** B a degree of 2 (connected to A, C), **Node** C a degree of 2 (connected to A, B) and **Node** D a degree of 1 (connected to A).

Degree k	Number of Nodes	Probability p_k
1	1	$1/4 = 0.25$
2	2	$2/4 = 0.50$
3	1	$1/4 = 0.25$

Table 1: Degree Distribution

Besides, the **Average Degree** $\langle k \rangle = \frac{1+2+2+3}{4} = \frac{8}{4} = 2$. Finally, the degree correlation matrix e^{random} represents the probability that a node of degree i is connected to a node of degree j, and we apply the formula to calculate the **probability of** q_k defined as $q_k = \frac{k \cdot p_k}{\langle k \rangle}$ and compute the **Random Network Expectation** $e_{ij}^{\text{random}} = q_i \cdot q_j$:

Degree k	p_k	q_k
1	0.25	0.125
2	0.5	0.5
3	0.25	0.375

$\longrightarrow$

	Degree 1	**Degree 2**	**Degree 3**
Degree 1	0.015625	0.0625	0.046875
Degree 2	0.0625	0.25	0.1875
Degree 3	0.046875	0.1875	0.140625

Table 2: Calculation of q_k & Random Network Expectation e_{ij}^{random}

2. **Actual Degree Correlation Matrix** e_{ij} We need to count the actual connections between degrees:

- **A (Degree 3)** connects to: B (Degree 2), C (Degree 2), D (Degree 1)
- **B (Degree 2)** connects to: A (Degree 3), C (Degree 2)
- **C (Degree 2)** connects to: A (Degree 3), B (Degree 2)
- **D (Degree 1)** connects to: A (Degree 3)

The **probability of the actual degree correlation matrix** is defiend as:

$$e_{ij} = \frac{\text{Number of connections between } i \text{ and } j}{\text{Total number of edges}} = \frac{\text{Connections}}{4}$$

	Degree 1	**Degree 2**	**Degree 3**
Degree 1	0	0	1
Degree 2	0	1	2
Degree 3	1	2	0

$\longrightarrow$

	Degree 1	**Degree 2**	**Degree 3**
Degree 1	0	0	0.25
Degree 2	0	0.25	0.5
Degree 3	0.25	0.5	0

Table 3: Actual Connection Counts & Actual Degree Correlation Matrix e_{ij}

3. **Comparison of Actual vs. Random Matrices**

Degree	Actual e_{jk}	Random e_{jk}^{random}
Degree 1	$e_{11} = 0$	$e_{11}^{\text{random}} = 0.015625$
	$e_{12} = 0$	$e_{12}^{\text{random}} = 0.0625$
	$e_{13} = 0.25$	$e_{13}^{\text{random}} = 0.046875$
Degree 2	$e_{21} = 0$	$e_{21}^{\text{random}} = 0.0625$
	$e_{22} = 0.25$	$e_{22}^{\text{random}} = 0.25$
	$e_{23} = 0.5$	$e_{23}^{\text{random}} = 0.1875$
Degree 3	$e_{31} = 0.25$	$e_{31}^{\text{random}} = 0.046875$
	$e_{32} = 0.5$	$e_{32}^{\text{random}} = 0.1875$
	$e_{33} = 0$	$e_{33}^{\text{random}} = 0.140625$

The results can be easily compared after the computations:

- **Degree 1**: As the actual $e_{13} = 0.25$ and Random $e_{13}^{\text{random}} = 0.046875$, **degree 1** nodes are more likely to connect to **degree 3** nodes than expected in a random network.
- **Degree 2**: As the actual $e_{22} = 0.25$ and Random $e_{22}^{\text{random}} = 0.25$ are equal, **degree 2** nodes connect to other **degree 2** nodes as expected. Although, as the actual $e_{23} = 0.5$ and Random $e_{23}^{\text{random}} = 0.1875$, **degree 2** nodes are more likely to connect to **degree 3** nodes.

- **Degree 3**: The actual $e_{31} = 0.25$ and Random $e_{31}^{\text{random}} = 0.046875$ implies that **degree 3** nodes are more likely to connect to **degree 1** nodes, and due to the actual $e_{32} = 0.5$ and Random $e_{32}^{\text{random}} = 0.1875$, **degree 3** nodes are more likely to connect to **Degree 2** nodes.

Bear in mind that, e_{jk}^{random} serves as a baseline to understand what connection probabilities would look like without any degree correlations, and the purpose of e_{jk} is to represent the actual connection probabilities in the observed network, capturing any existing degree correlations.

In general, we observe a deviation from the random expectation, i.e. the actual e_{jk} significantly differs from e_{jk}^{random}, indicating that the network exhibits degree correlations, in particular Assortative Mixing (evident where **degree 2** nodes connect more frequently to other higher-degree nodes (**degree 2** to **degree 3**) than expected) and Disassortative Mixing (evident where **degree 3** nodes connect more frequently to lower-degree nodes (**degree 1** and **degree 2**) than expected).

4. $\boxed{\textbf{Degree Correlation Function } (k_{\mathbf{nn}}(k))}$ The formula to compute is:

$$k_{\text{nn}}(k) = \sum_{k'} k' \cdot P(k'|k)$$

Where $P(k'|k)$ is the probability that a node of degree k is connected to a node of degree k'. First we need to determine the probability $P(k'|k)$:

- **Node A (Degree 3)**: Connected to B (Degree 2), C (Degree 2), D (Degree 1)
  - Total neighbors: 3
  - $P(2|3) = \frac{2}{3} \approx 0.6667$
  - $P(1|3) = \frac{1}{3} \approx 0.3333$
- **Node B & Node C (Degree 2)**: B Connected to A (Degree 3) and C (Degree 2), C connected to A (Degree 3) and B (Degree 2)
  - Total neighbors (for Degree 2): 4 (each Degree 2 node has 2 neighbors)
  - $P(3|2) = \frac{2}{4} = 0.5$
  - $P(2|2) = \frac{2}{4} = 0.5$
- **Node D (Degree 1)**: Connected to A (Degree 3)
  - Total neighbors: 1
  - $P(3|1) = 1$

Now, we can compute $k_{\text{nn}}(k) = \sum_{k'} k' \cdot P(k'|k)$:

- **For k = 1**: $k_{\text{nn}}(1) = 3 \times 1 = 3$
- **For k = 2**: $k_{\text{nn}}(2) = 3 \times 0.5 + 2 \times 0.5 = 1.5 + 1 = 2.5$
- **For k = 3**: $k_{\text{nn}}(3) = 2 \times 0.6667 + 1 \times 0.3333 \approx 1.3334 + 0.3333 = 1.6667$

Secondly, we compute the $k_{\text{nn}}^{\text{random}} = \frac{\langle k^2 \rangle}{\langle k \rangle}$ where:

- $\langle k^2 \rangle = \frac{1^2+2^2+2^2+3^2}{4} = \frac{1+4+4+9}{4} = \frac{18}{4} = 4.5$
- $\langle k \rangle = \frac{1+2+2+3}{4} = \frac{8}{4} = 2$

Then, $k_{\text{nn}}^{\text{random}} = \frac{4.5}{2} = 2.25$. Summarizing the information in a table, we notice that:

Degree k	$k_{\text{nn}}(k)$	$k_{\text{nn}}^{\text{random}}$
1	3	2.25
2	2.5	2.25
3	1.6667	2.25

- **For k = 1**: $k_{\text{nn}}(1) = 3 > k_{\text{nn}}^{\text{random}} = 2.25$. Nodes with degree 1 have neighbors with higher average degrees than expected randomly.
- **For k = 2**: $k_{\text{nn}}(2) = 2.5 > k_{\text{nn}}^{\text{random}} = 2.25$. Nodes with degree 2 have neighbors with slightly higher average degrees than expected randomly.
- **For k = 3**: $k_{\text{nn}}(3) = 1.6667 < k_{\text{nn}}^{\text{random}} = 2.25$. Nodes with degree 3 have neighbors with lower average degrees than expected randomly.

We could conclude that the considered network is somewhat disassortative.

By representing the function as values of the node degrees k, we can appreciate the trend that allows us to determine whether the network is **neutral**, **assortative** or **disassortative**, like in the following 2 examples:

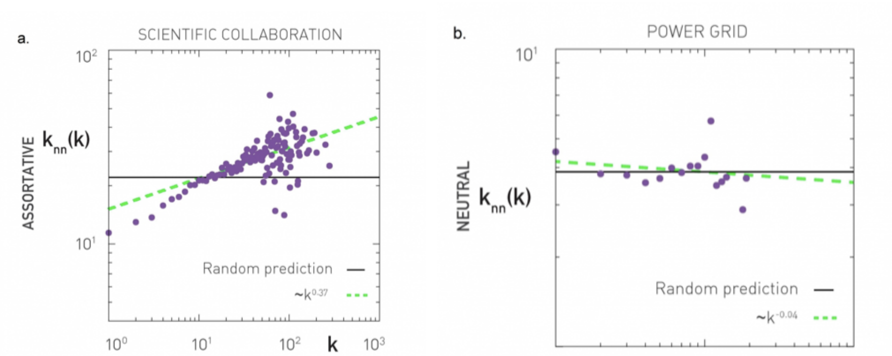

Figure 4: Scientific Collaboration & Power Grid

5 Approximating the Degree Correlation Function

The degree correlation function $k_{\text{nn}}(k)$ provides insight into how the average degree of a node's neighbors varies with the node's own degree. To simplify the analysis and identify underlying patterns, we can approximate this function using a Power-Law form:

$$k_{\text{nn}}(k) = a \cdot k^{\mu}$$

5.1 Interpretation of the Correlation Exponent μ

The exponent μ in the approximation $k_{\text{nn}}(k) = a \cdot k^{\mu}$ characterizes the type of network based on the nature of degree correlations:

- $\boxed{\mu > 0}$ **Assortative Networks**
  - Nodes with higher degrees tend to connect to other high-degree nodes.
  - Leads to clusters of hubs within the network.
- $\boxed{\mu = 0}$ **Neutral Networks**
  - No degree correlation; the average neighbor degree is independent of the node's degree.
  - Connections occur randomly with respect to node degrees.
- $\boxed{\mu < 0}$ **Disassortative Networks**
  - Nodes with higher degrees tend to connect to low-degree nodes.
  - Results in a hierarchical or star-like structure.

5.2 Numerical Example Continuation

Recall from the previous section:

Degree k	$k_{\text{nn}}(k)$	$k_{\text{nn}}^{\text{random}}$
1	3	2.25
2	2.5	2.25
3	1.6667	2.25

To approximate $k_{\text{nn}}(k)$ with the form $a \cdot k^{\mu}$, we can perform a simple regression on the available data points. Given the small size of the network (only 3 unique degree values), this approximation serves as an illustrative example rather than a statistically robust model. We assume the form $k_{\text{nn}}(k) = a \cdot k^{\mu}$ and we proceed as follows:

$$\ln(k_{\text{nn}}(k)) = \ln(a) + \mu \ln(k)$$

Setting up the equations, and solving for μ and a:

$$\begin{aligned} \ln(3) &= \ln(a) + \mu \ln(1) & & \ln(3) = \ln(a) \;\Rightarrow\; \ln(a) = \ln(3) \;\longleftrightarrow\; a = 3 \\ \ln(2.5) &= \ln(a) + \mu \ln(2) & \longleftrightarrow\; & \ln(2.5) = \ln(3) + \mu \ln(2) \;\longleftrightarrow\; \mu \approx -0.252 \\ \ln(1.6667) &= \ln(a) + \mu \ln(3) & & \ln(1.6667) = \ln(3) + \mu \ln(3) \;\longleftrightarrow\; \mu \approx -0.536 \end{aligned}$$

Considering μ as the average of both values (for illustrative purposes), we obtain:

$$k_{\text{nn}}(k) \approx 3 \cdot k^{-0.394}$$

With $\mu \approx -0.394 < 0$, the network is disassortative. This aligns with our previous observations where high-degree nodes tend to connect to low-degree nodes.

6 Degree Correlation Coefficient

While the degree correlation function $k_{\text{nn}}(k)$ provides a functional perspective on degree correlations, the degree correlation coefficient r offers a scalar measure, summarizing the overall degree correlation in the network. The degree correlation coefficient r is defined as:

$$r = \frac{\sum_{j,k} j \cdot k \cdot (e_{jk} - q_j q_k)}{\sigma^2} \quad \text{where: } \sigma^2 = \sum_{j,k} j \cdot k \cdot q_j q_k - \left(\sum_j j \cdot q_j \right)^2$$

The coefficient r captures the Pearson correlation between the degrees of nodes at either end of an edge:

- $\boxed{r > 0}$ **Assortative Networks**
  - Positive correlation between degrees of connected nodes.
  - High-degree nodes tend to connect to other high-degree nodes.
- $\boxed{r = 0}$ **Neutral Networks**
  - No correlation between degrees of connected nodes.
  - Connections are random with respect to node degrees.
- $\boxed{r < 0}$ **Disassortative Networks**
  - Negative correlation between degrees of connected nodes.
  - High-degree nodes tend to connect to low-degree nodes.

6.1 Giant Components & Path Length in Networks

6.1.1 Giant Component and Robustness

- **Assortative Networks**: Giant component forms earlier ($\langle k \rangle$ lower), enhancing robustness due to high-degree nodes clustering together.
- **Disassortative Networks**: Giant component forms later ($\langle k \rangle$ higher), making them less robust as high-degree nodes bridge low-degree ones.
- **Neutral Networks**: Intermediate behavior, similar to random networks.

6.1.2 Path Length and Stability

- **Assortative Networks**: Lower average path length and maximum path length ($d_{\max}$), reflecting tighter clustering.
- **Disassortative Networks**: Higher $d_{\max}$ and longer paths due to dispersed connectivity.

Other consequences include the stability from assortative networks which are more stable under random failures; disassortative networks are more fragile.

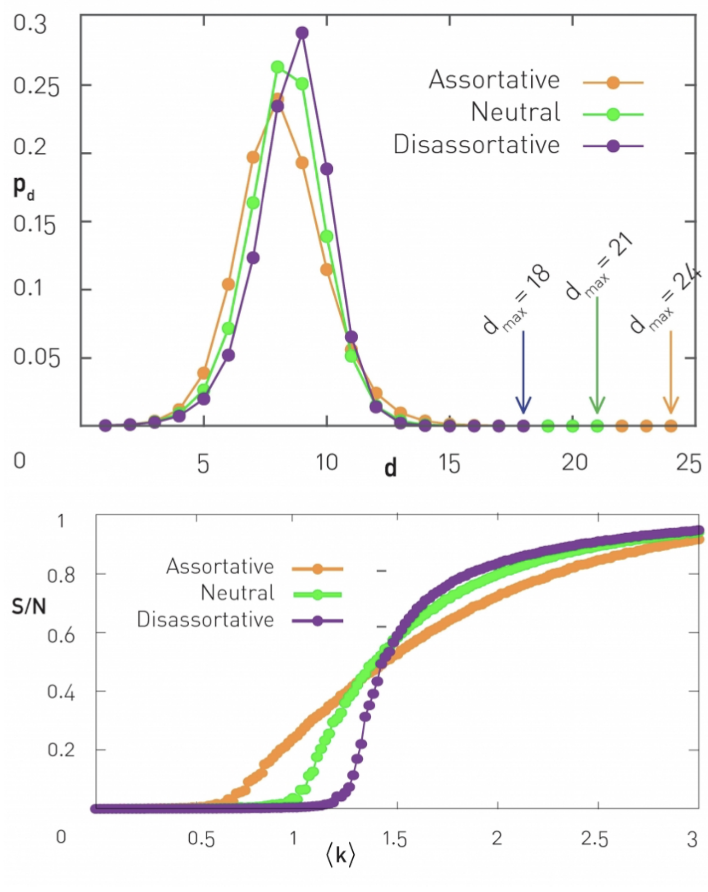

Figure 5: Average path Length & Giant Component Size **(S/N)**