Supplementary

In the table below we summarize the main differences and the key features one can notice when comparing both types of graphs, especially if they have the same number of nodes and a similar number of edges:

	Erdős-Rényi	Barabási-Albert
Modeling approach	This model generates graphs by randomly connecting pairs of nodes with a given probability. In the $G(n, p)$ version, each pair of n nodes is connected with a probability p, resulting in an approximately uniform distribution of edges across the graph.	This model uses a "preferential attachment" mechanism, where new nodes are more likely to connect to already well-connected (high-degree) nodes. This leads to a scale-free structure, meaning that the node degrees follow a power-law distribution, with a few highly connected nodes (hubs) and many nodes with low degree.
Degree distribution	The degree distribution is approximately Poisson (for large n) in the $G(n, p)$ model. This means that most nodes have a similar number of connections, close to the average degree. The likelihood of finding nodes with significantly more or fewer edges than the average is low. Degree distribution is relatively narrow.	The degree distribution follows a power-law. This implies a few nodes (hubs) with very high degrees, while most nodes have very few connections. Degree distribution is very broad, with a long tail (high-degree nodes are rare but possible).
Visual differences	The network will look fairly homogeneous. The nodes will appear to have a roughly equal number of connections, and there will be no obvious hubs or nodes with significantly higher degrees than others. The graph will appear more random and even, with edges more uniformly distributed across the nodes.	The graph will have an uneven, "hub-and-spoke" structure, with some nodes (the hubs) having many more connections than the average node. The most obvious visual feature is the presence of a few highly connected hubs, around which many other nodes are clustered with fewer edges. The rest of the network will consist of nodes with only a few connections, forming a "sprawling" structure from the hubs.
Clustering coefficient	The clustering coefficient (how likely two neighbors of a node are to be connected) tends to be quite low in ER graphs because edges are randomly distributed.	BA graphs tend to have a higher clustering coefficient, especially around the hubs, as new nodes often connect to already well-connected nodes, forming local clusters.
Shortest path length	The average shortest path length between any two nodes in an ER graph is relatively short, but not as extreme as in BA graphs. It scales logarithmically with the number of nodes, i.e., $\log(n)$.	The average shortest path length tends to be very short (the small-world effect) due to the presence of hubs, which act as shortcuts between otherwise distant nodes. The path length scales with $\log(\log(n))$.
First noticeable patterns	Nodes have similar degrees, with no clear dominant hubs, and the graph looks more evenly spread out.	A few dominant hubs with many connections, while the majority of nodes have very few edges, forming a more clustered, uneven structure.

Centrality definitions

Edge betweenness centrality Edge betweenness centrality and weighted edge betweenness centrality are measures in network analysis that assess the importance of edges in a graph. They quantify how often an edge lies on the shortest paths between pairs of vertices. The edge betweenness centrality of an edge is the sum of the fraction of all-pairs shortest paths that pass through the edge. For each pair of vertices (s, t), the fraction of shortest paths between s and t that go through a particular edge is calculated. The edge betweenness centrality is then the sum of that fractions over all pairs (s, t).

Formally, $C^{edge}_{betweenness}(e) = \sum\limits_{s \neq t \in V} \frac{\sigma_{st}(e)}{\sigma_{st}}$, where σ_{st} is the total number of shortest paths from vertex s to vertex t and $\sigma_{st}(e)$ is the number of those paths that pass through edge e.

In a weighted graph, the weighted edge betweenness centrality takes into account the weights when calculating the shortest paths. The idea remains the same, it measures how often an edge is on the shortest paths, but "shortest" now refers to the path with the minimum total weight, not just the fewest number of edges. The graph's weights are used to determine the shortest paths (typically using algorithms like Dijkstra's). The centrality is then calculated similarly to the unweighted version but with weighted shortest paths. The formula remains the same: $C^{edge}_{betweenness}(e) = \sum\limits_{s \neq t \in V} \frac{\sigma_{st}(e)}{\sigma_{st}}$. But here, σ_{st} and $\sigma_{st}(e)$ are based on the shortest paths determined by edge weights.

Edges with high betweenness centrality are crucial for maintaining the shortest paths within the graph. Such edges can be considered as "bottlenecks" or "bridges" in the network. Edges with high weighted edge betweenness centrality are critical in terms of the weighted shortest paths in the network. For instance, in a transportation network, an edge with high weighted edge betweenness might represent a critical route that minimizes travel time or cost. Both measures are used to identify crucial connections within a network that, if removed, could significantly impact the flow or connectivity within the graph.

Edge closeness centrality Edge closeness centrality and weighted edge closeness centrality are measures that extend the concept of closeness centrality from nodes to edges in a graph. These measures quantify the importance of an edge based on its role in facilitating efficient communication or connectivity across the network. While the definition of node closeness centrality is generally uniformly accepted, there are more than one definitions of edge closeness centrality, very similar to each other. We define the (weighted/unweighted) edge closeness centrality of an edge as the sum of (weighted/unweighted) node closeness centralities of its end vertices.

Node closeness centrality quantifies how close a node is to all other nodes in the graph. It is based on the idea that a node is more important or influential if it can quickly interact with other nodes. Closeness centrality of a node is defined as the reciprocal of the sum of the shortest path distances from that node to all other nodes in the network. The same holds for weighted node closeness centrality with the difference that the distances are calculated based on the weighted shortest pathes. Mathematically, it's expressed as:

$C^{node}_{closeness}(v) = \frac{1}{\sum\limits_{v \neq u \in V} d(v,u)}$, where $d(v, u)$ is the (unweighted or weighted) distance between v and u.

A node with high closeness centrality can reach other nodes in the network more quickly on average, meaning it is "close" to many other nodes. Closeness centrality is particularly useful when the speed of information or resource flow through a network is critical. An edge with high closeness centrality is one that contributes significantly to keeping the average shortest path length in the graph low. It plays a crucial role in maintaining efficient connectivity across the network. An edge with high weighted edge closeness centrality is essential for maintaining low average weighted shortest path lengths across the graph. It is particularly significant in networks where the edge weights represent important factors like time, cost, or capacity. Both measures, unweighted and weighted edge closeness centrality, help in identifying edges that are vital for ensuring efficient connectivity or communication within the network.