Graph Models

CSCI-347 Data Mining

Real-world graph properties

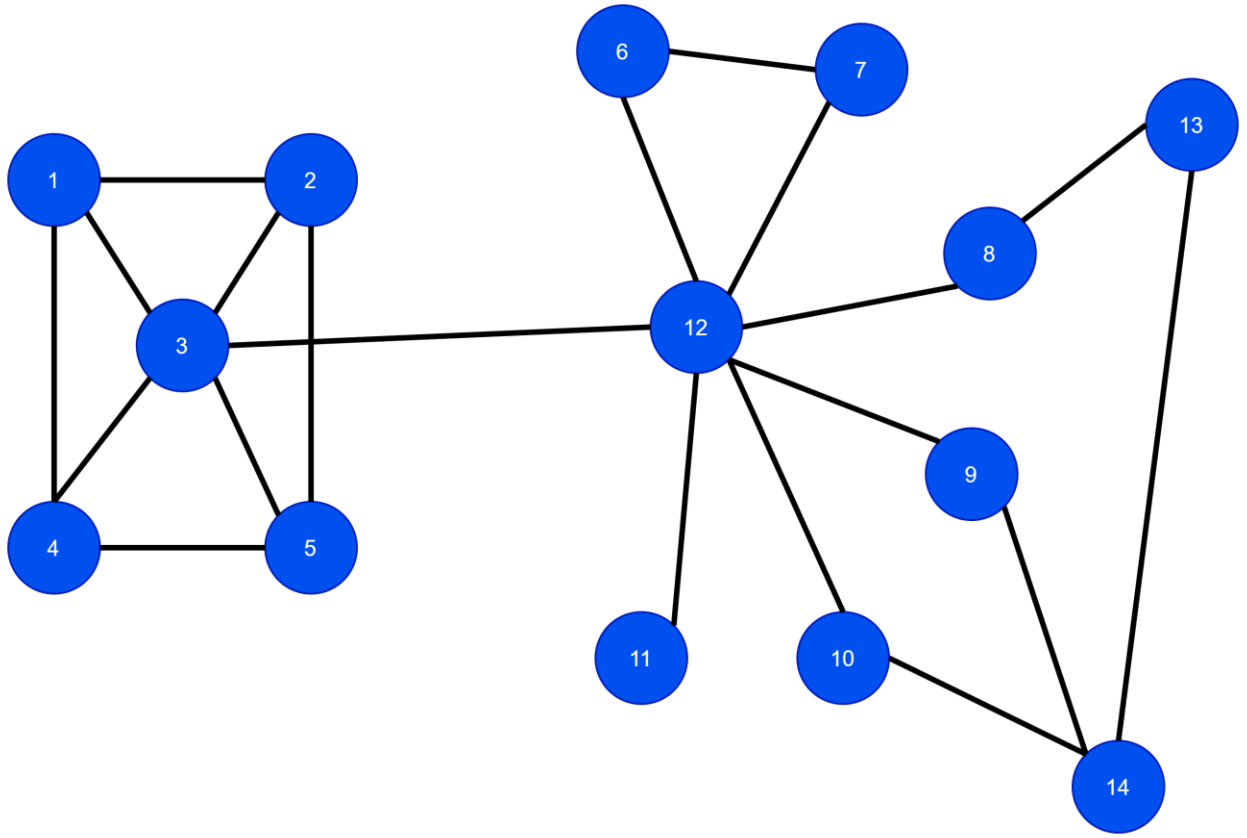
- Many real-world networks exhibit common characteristics, even though the underlying data can come from vastly different domains.
  - Social networks
  - Biological networks
  - Telecommunication networks
- Real-world networks are usually **large** and **sparse.**
- By large, we mean the order of the graph (number of nodes) is large.
- By sparse, we mean that graph size (number of edges) $m = O(n)$.

Real-world graph properties

- What kind of underlying processes enables graphs to have these properties?
- We will look at few different measures that will allow sus to compare and contrast real-world networks.

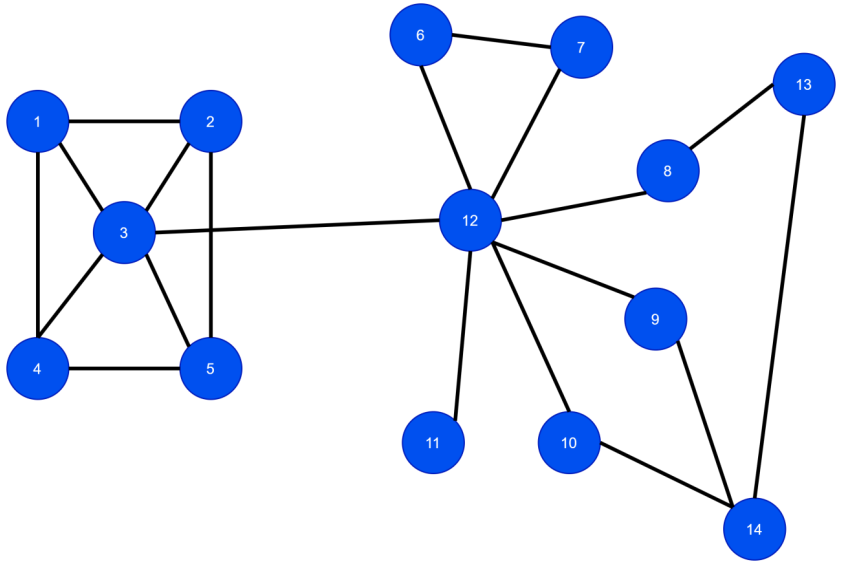
Real-world properties

- ***Small-world property***
- ***Scale-free property***
- ***Clustering effect***



Small-world property

- There is a short path between any pair of nodes.
- The graph G exhibits small-world behavior if the average path length μ_L scales logarithmically with the number of nodes in the graph, i.e.,



$$\mu_L \propto \log(n)$$

A graph is said to possess ultra-small-world property if the average path length is much smaller than $\log n$, i.e., $\mu_L \ll \log n$

MONTANA STATE UNIVERSITY

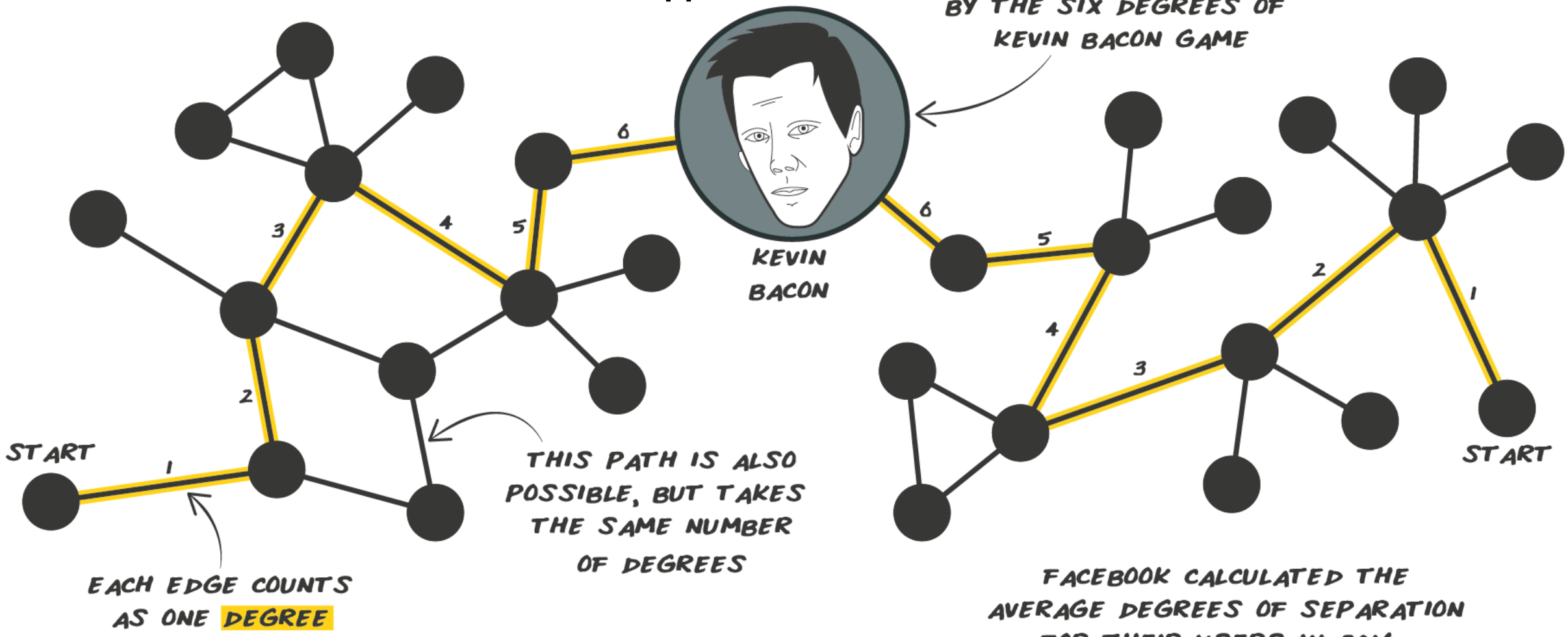
Six degrees of separation

- The Six Degrees of Separation theory suggests that any two people in the world are connected through at most six social connections (or "degrees").
- This means you can reach anyone—whether it's a celebrity, a politician, or a stranger in another country—through a chain of at most six intermediaries.
- Modern research, including studies using Facebook and LinkedIn, suggests the actual number may be closer to 4 or 5 degrees due to social media.

The Reliants Project.

SIX DEGREES OF SEPARATION

It measures how closely connected an actor is to Kevin Bacon based on shared film appearances.



https://www.reliantsproject.com/2020/06/13/concept-7-kevin-bacon-and-six-degrees-of-separation/

The Oracle of Bacon

Fun application of 6 degrees of separation

- Go to https://oracleofbacon.org/movielinks.php

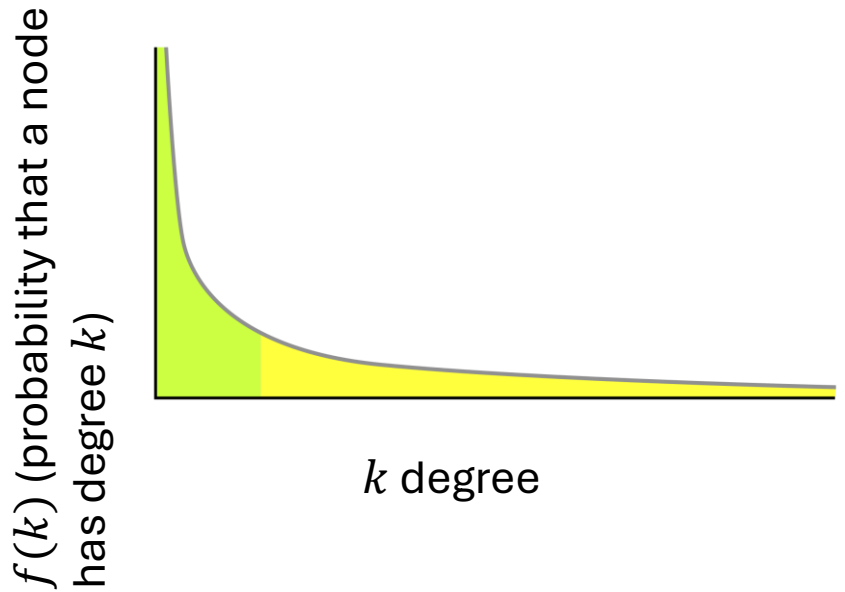
Small-world property

- The **Six Degrees of Separation** exists because of the **small-world property**.
- Small-world networks have **high clustering** coefficient and **short average path lengths**.

Scale-free property

- In many real-world graphs:
- the empirical degree distribution $f(k)$ exhibits a ***scale-free*** behavior captured by a **power-law** relationship with k, i.e., the probability that a node has degree k satisfies the condition:

$$f(k) \propto k^{-\gamma}$$



$f(k) = probability\ that\ a\ node\ has\ degree\ k$

Scale-free property

- a power-law indicates that the vast majority of nodes have very small degrees.
- Few “Hub” nodes have high degree—they connect to or interact with lot of nodes.

Why is this called scale-**free**?

- The power law relationship leads to something called **scale-free** or **scale-invariant** behavior, i.e., scaling the argument by some constant c does not change the proportionality.
- Suppose degree distribution $f(k)$ exhibits a relationship with k that can be described by power-law relationship.

$$f(k) \propto k^{-\gamma}$$

$$f(k) = \alpha k^{-\gamma}$$

$$f(ck) = \alpha(ck)^{-\gamma} = (\alpha c^{-\gamma})k^{-\gamma} \propto k^{-\gamma}$$

Why is this called scale-**free**?

$$\boldsymbol{f(k) \propto k^{-\gamma}}$$

$$f(k) = \alpha k^{-\gamma} \tag{1}$$

$$f(ck) = \alpha(ck)^{-\gamma} = (\alpha c^{-\gamma})k^{-\gamma} \propto k^{-\gamma}$$

Also taking the logarithm of both sides of (1).

$$\log f(k) = \log \alpha k^{-\gamma}$$

$$\log f(k) = -\gamma \log k + \log \alpha$$

This is basically equation of a straight line in the log-log plot of $\boldsymbol{k}$ vs $\boldsymbol{f(k)}$, with $-\boldsymbol{\gamma}$ giving the slope of the line.

How to check whether a graph exhibits scale-free behavior?

- If you are given a graph and asked to check whether it exhibits scale-free behavior then, you can perform a least-square fit of the points $(\log k\,, \log f(k))$ to a line.
- What is least square fit?
  - When you have **a set of data points** (x_i, y_i), but they don't all lie on a perfect line, least-squares fitting finds the **best straight-line approximation.**
- **How does it work?**

Least-square fit

- Given set of data points $(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)$, we assume a linear relationship.

$$y = mx + c$$

The goal is to find m and b such that total squared error is minimized:

$$\sum_{i=1}^{n} \left(y_i - (mx_i + b)\right)^2$$

$\left(y_i - (mx_i + b)\right)^2$ is the difference (or error) between the actual data point and the predicted value on the line.

Least-square fit

- Least square method calculate m and b using:

$$m = \frac{n\,\Sigma x_i y_i - \Sigma x_i \Sigma y_i}{n\Sigma x_i^2 - (\Sigma x_i)^2}$$

$$b = \frac{\Sigma y_i - m\Sigma x_i}{n}$$

These formulas find the best line that **minimizes the error**.

Determining whether a graph is scalar-free

$$\boldsymbol{f(k) \propto k^{-\gamma}}$$

$$\log f(k) = \log \alpha k^{-\gamma}$$

$$\log f(k) = -\gamma \log k + \log \alpha$$

Intercept

Slope

$$y = mx + b$$

How to check whether how well this line fit?

- One way you can do this is by using Coefficient of Determination R^2

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

Variance

Typically,

$R^2 \to 1$ Perfect fit

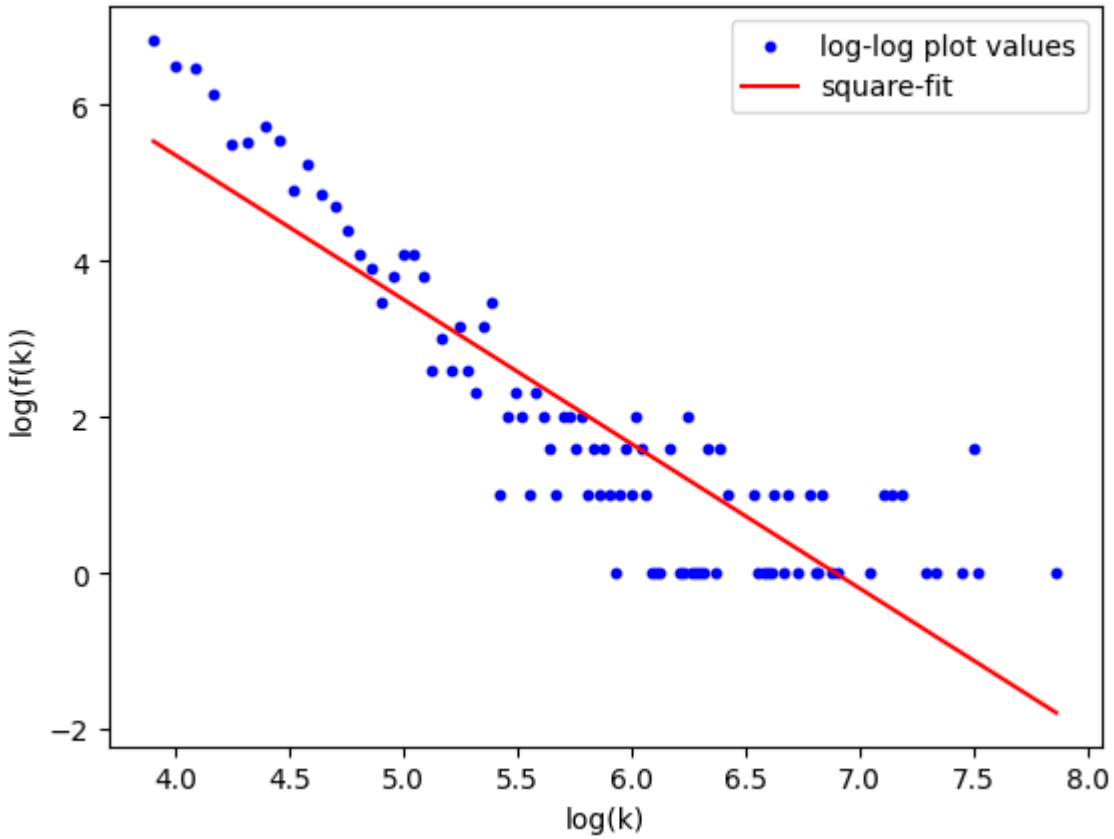
$R^2 \to 0$ →Model explains nothing

$0 < R^2 < 1 \to$ Partial explanation

If R^2 is high, then the graph follows power-law

$R^2 < 0 \to$ The fit is worse than just using the average $\bar{y}$ as the predictor

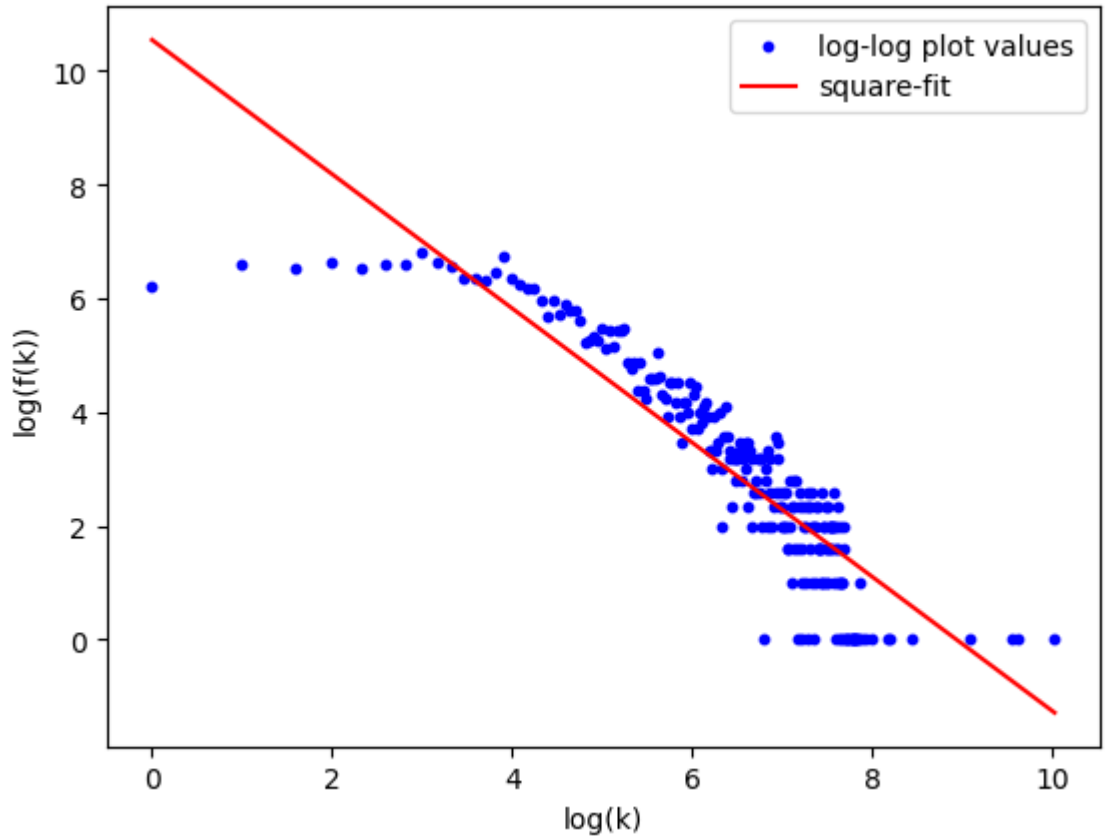
This basically quantifies the proportion of the variance in the dependent variable (your original data) that is predictable from the independent variables (or your model's predictions)


log-log plot values
square-fit
log(f(k))
log(k)
6
4
2
0
−2
4.0
4.5
5.0
5.5
6.0
6.5
7.0
7.5
8.0

In class-exercise

- Check whether the facebook_combined dataset is scale-free.
- Load the dataset and follow what we did in the class.

Example: Facebook_combined dataset



$$R^2 = 0.809$$

- In practice, one of the problems with estimating the degree distribution for a graph is the **high level of noise** for the **higher degrees. (frequency counts are lowest for higher degrees).**
- One approach:
  - Use cumulative degree distribution $F(k)$
  - Smooths out noise.
  - We can also use $F^c(k) = 1 - F(k)$, which gives the probability that a randomly chosen node has degree greater than k.
  - Let's look at some math for this case.
- **$F(k)$ is the probability that a node has degree k or less.**

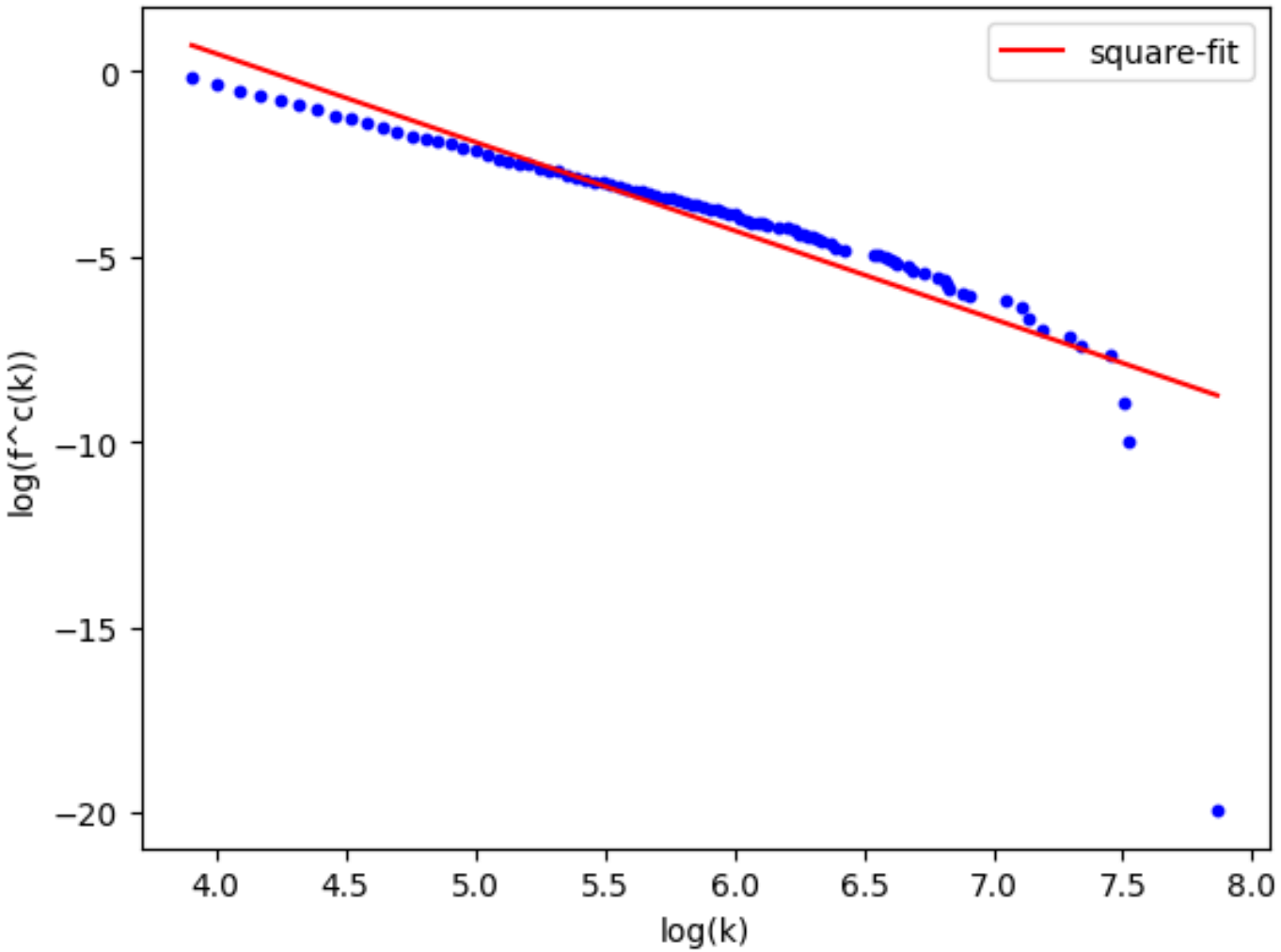
$$if\ f(k) \propto k^{-\gamma}, \gamma > 1$$

$$F^c(k) = 1 - F(k) = 1 - \sum_{x=0}^{k} f(x) = \sum_{x=k+1}^{\infty} f(x) = \sum_{x=k+1}^{\infty} x^{-\gamma}$$

$$\approx \int_{k+1}^{\infty} x^{-\gamma}\, dx = \frac{x^{-\gamma+1}}{-\gamma+1}\Bigg|_{k+1}^{\infty} \approx \frac{x^{-\gamma+1}}{-\gamma+1}\Bigg|_{k}^{\infty} = \frac{1}{\gamma-1} \cdot k^{-(\gamma-1)} \propto k^{-(\gamma-1)}$$

When k is large

In other words, log-log plot of $F^c(k)\ versus\ k$ will also be a power-law with slope $-(\gamma-1)$ as opposed to $-\gamma$.


square-fit
0
−5
−10
−15
−20
log(f^c(k))
4.0
4.5
5.0
5.5
6.0
6.5
7.0
7.5
8.0
log(k)

Clustering effect

- Two nodes are more likely to be connected if they share a common neighbor.
- $C(k)$ denote the average clustering coefficient for all nodes with degree k.
- clustering effect also manifests itself as a power-law relationship between $C(k)$ and k:

$$C(k) \propto k^{-\gamma}$$

- log-log plot of k versus $C(k)$ exhibits a straight-line behavior with negative slope $-\gamma$

Clustering effect; Why?

- High-degree nodes (hubs) connect to many nodes, but their neighbors are often not connected to each other.
  - A hub connects to many low-degree nodes from different parts of the network.
  - These low-degree nodes are unlikely to be connected to each other.
- Low-degree nodes are often in tightly connected local communities.
  - A low-degree node may connect to only a few nodes in a well-defined cluster.
  - These neighbors are more likely to be connected to each other, forming triangles.

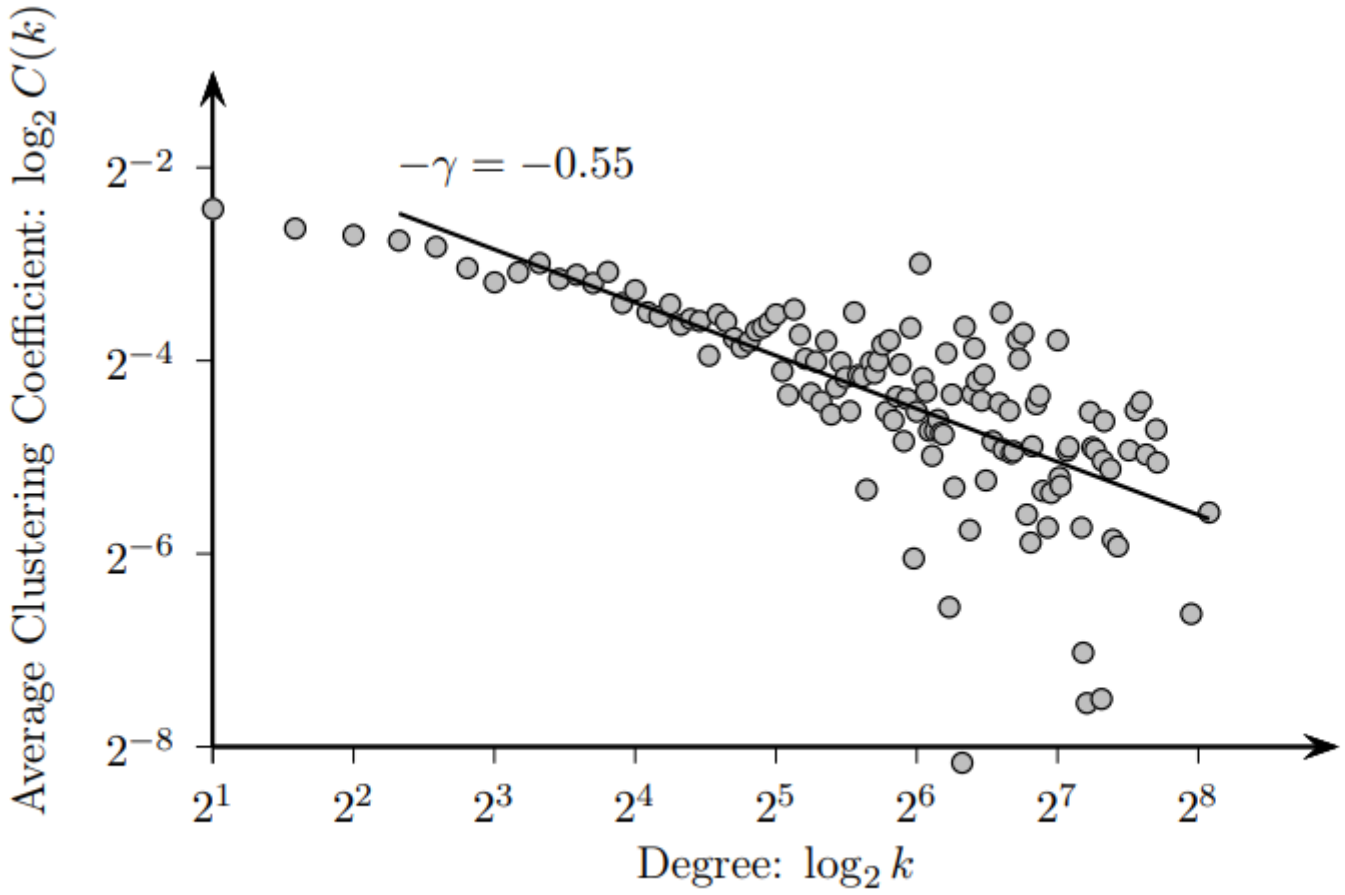


Figure 4.9: Average Clustering Coefficient Distribution

Graph models

- Now, we are going to look at different graph generation models.
- Understanding different **graph models** is crucial because real-world networks (e.g., social networks, biological systems, and communication networks) exhibit diverse structural properties. Different models help us **simulate**, **analyze**, **and** **predict** **behaviors** in these networks.
- We will learn about 3 models
  - Watts-Strogatz model.
  - Barabasi-Albert model.
  - Erdos-Renyi Random graph model.

Real-world networks have different structures

Different networks follow different connectivity patterns. By studying various models, we can match them to real-world networks:

Network type	Best model to use	Why?
Social Networks	Watts-Strogatz	Small-world property (short path lengths, high clustering)
Web networks/Internet	Barabasi-Albert (BA)	Scale-free properties (hubs, power-law degree distribution)
Scientific collaboration networks	Barabasi-Albert (BA)	A few researchers (hubs) have many collaborators
Road and Transportation Networks	Watts-Strogatz	Mostly local connections with some shortcuts
Neural networks	Watts-Stragatz	High clustering with short path lengths

Understanding How Networks Evolve

- Different models explain **how networks grow over time**:
- Barabási–Albert Model (Preferential Attachment):
  - Explains why some nodes (hubs) get disproportionately many connections.
  - Example: How popular websites (Google, Wikipedia) get more links over time.
- Watts–Strogatz Model (Small-World Effect):
  - Explains why most real-world networks have short path lengths and high clustering.
  - Example: Why social networks allow people to connect in just a few steps (Six Degrees of Separation).
- Erdős–Rényi Random Graph Model:
  - Useful for studying random connectivity patterns, such as random failures in communication networks.

Predicting Network behavior

- Graph models help in predicting how networks respond to changes.
- **Robustness and Attack Resistance**
  - Scale-free (BA) networks are robust against random failures but vulnerable to targeted attacks.
  - Random networks (Erdős–Rényi) fail gradually when removing nodes.
  - Helps in designing more resilient infrastructure (e.g., the internet, power grids).
- **Spread of Information/Disease:**
  - In small-world networks (WS model), diseases or viral content spread quickly.
  - Understanding these dynamics helps in epidemiology and marketing strategies.

Predicting Network behavior

- Community Detection & Clustering
  - Graph models help in understanding community structures, like groups in social networks.
  - Helpful in disaster response.
    - Identify most vulnerable communities in network.

ERDÖS-RÉNYI RANDOM GRAPH

- The Erdös-Rényi (ER) model generates a random graph such that any of the possible graphs with a fixed number of nodes and edges has equal probability of being chosen.
- Model has two parameters:
  - n=number of nodes
  - m=number of edges
- Let $M = \binom{n}{2} = \frac{n(n-1)}{2}$ (Maximum number of edges possible among n nodes)
- ER model specifies a collection of graphs $\mathcal{G}(n, m)$ with n nodes and m edges, such that each graph $G \in \mathcal{G}$ has equal probability of being selected.

$$P(G) = \frac{1}{\binom{M}{m}} = \binom{M}{m}^{-1}$$

$\binom{M}{m}$ is the number of possible graphs with m edges (with n nodes) corresponding the ways of choosing the m edges out of a total of M possible edges.

We will look at how this model generates such graphs.

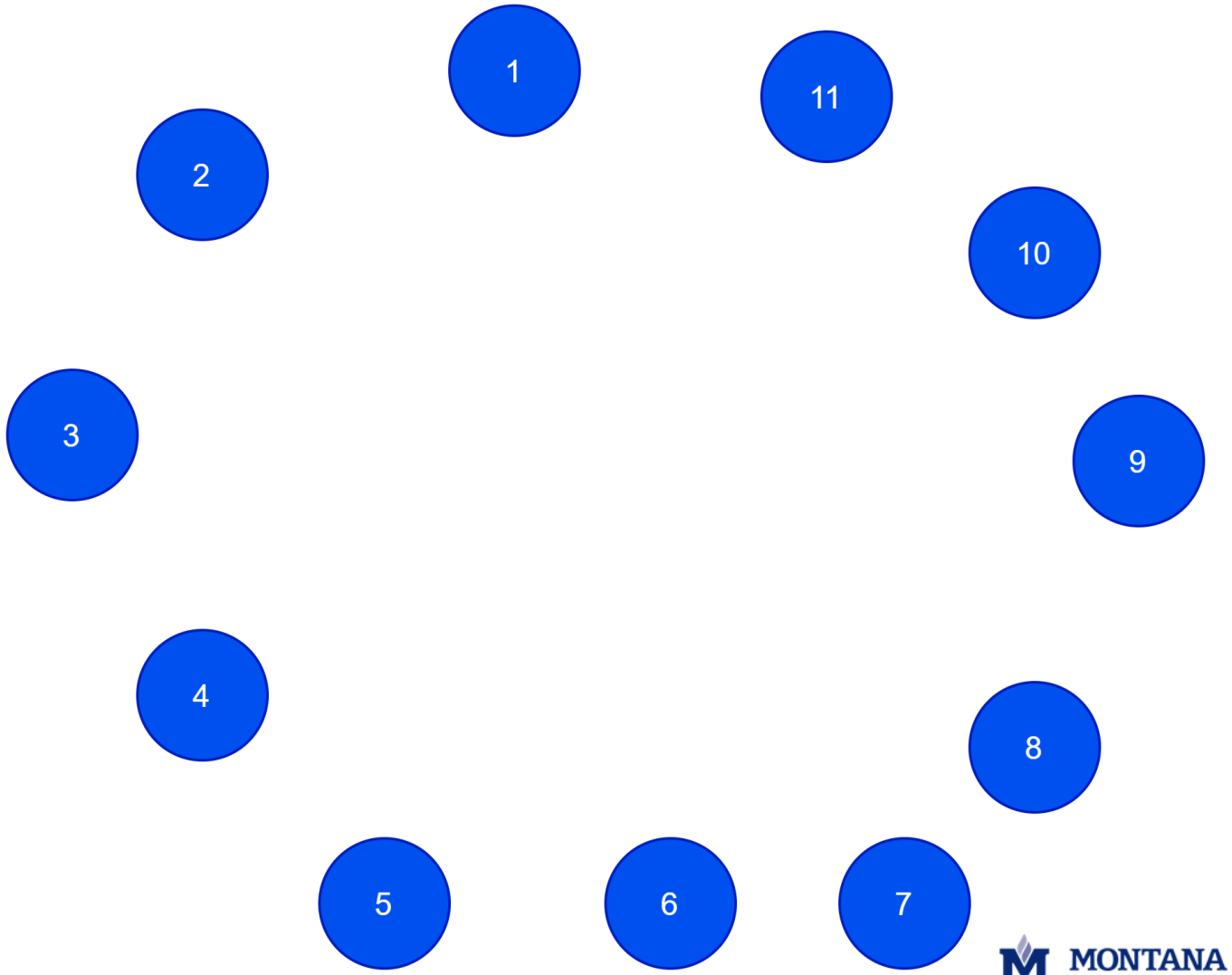
ERDÖS-RÉNYI RANDOM GRAPH generation

Process: At each step, randomly select two distinct vertices $v_i, v_j \in V$, and add edge (v_i, v_j) to E, provided that the edge is not already in the graph G. The process is repeated until exactly m edges have been added to the graph.

ERDÖS-RÉNYI RANDOM GRAPH generation

Parameters: n = 11, m = 15

- Select a pair of nodes uniformly at random.
- If an edge does not exist between the pair, add it.

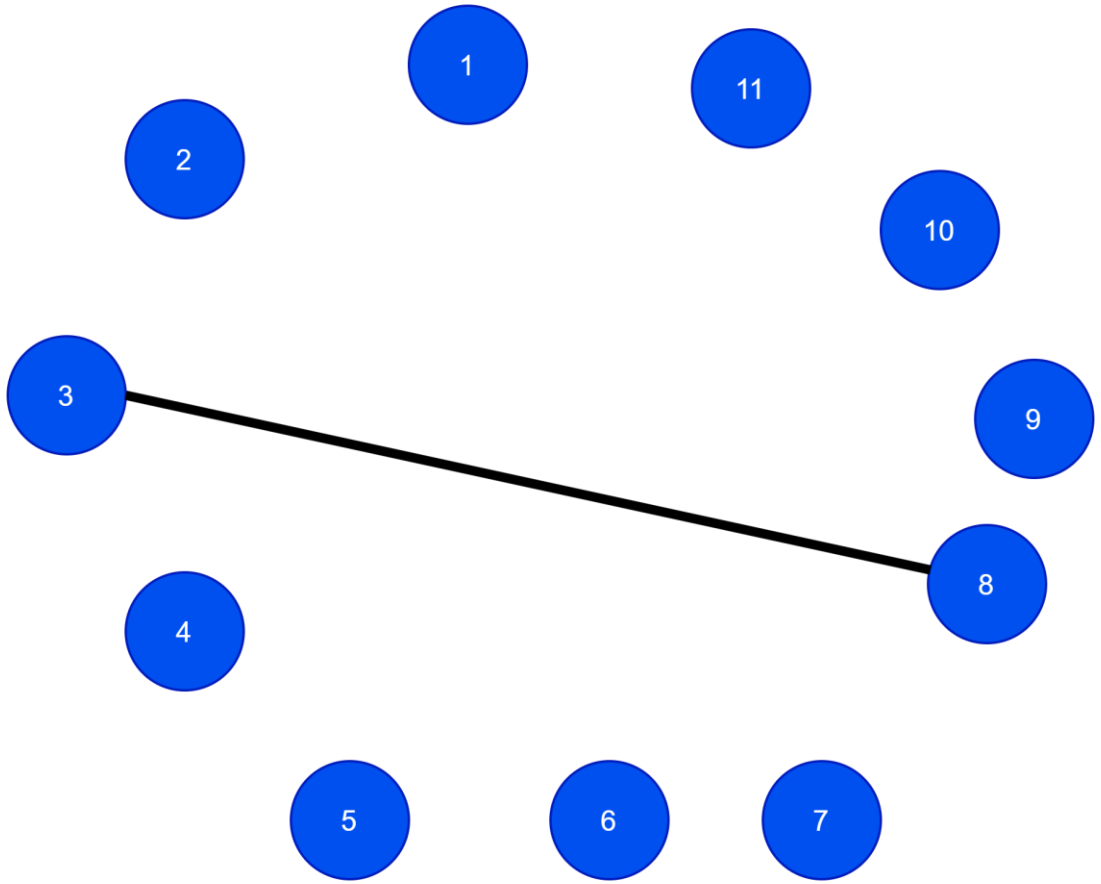


MONTANA STATE UNIVERSITY

ERDÖS-RÉNYI RANDOM GRAPH generation

Parameters: n = 11, m = 15

- Select a pair of nodes uniformly at random. (3, 8)
- If an edge does not exist between the pair, add it.

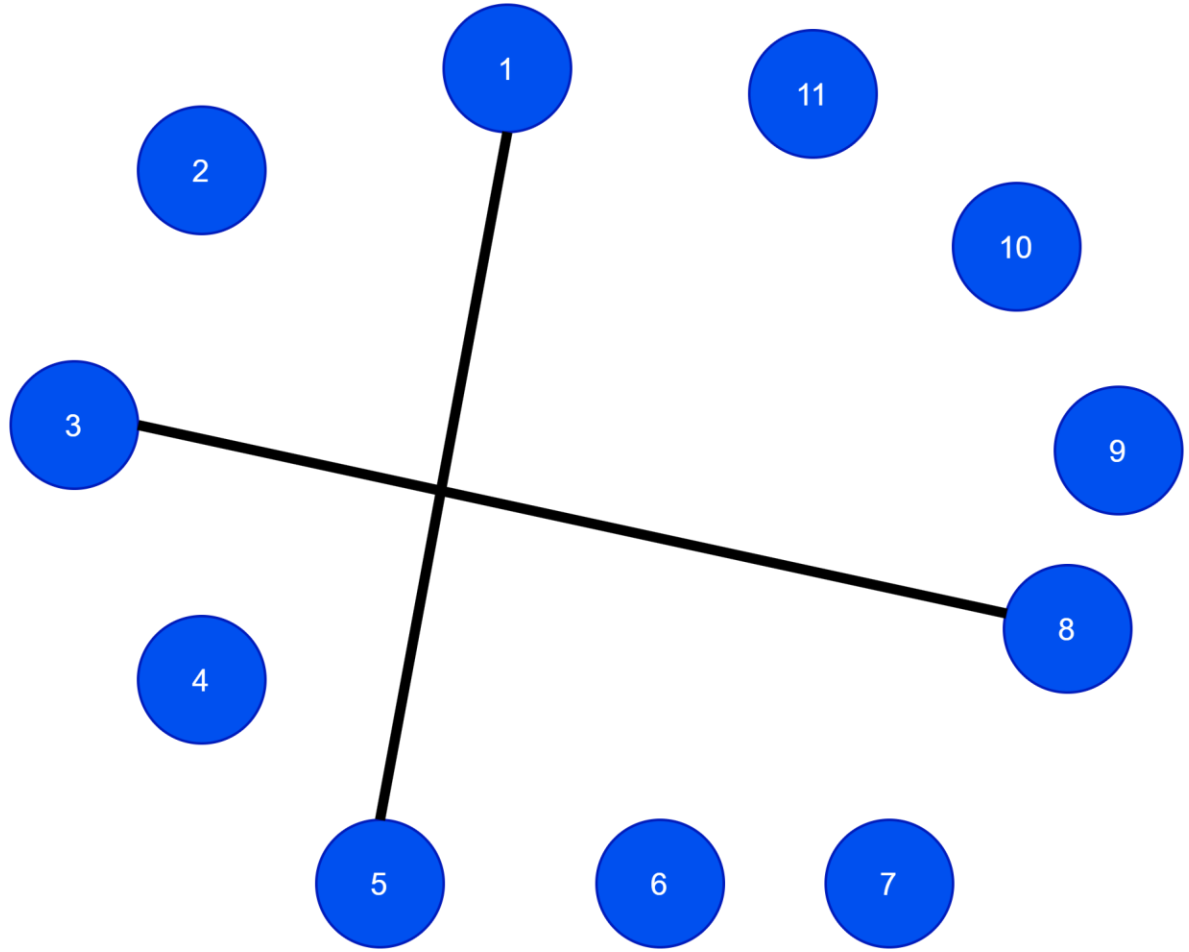


MONTANA STATE UNIVERSITY

ERDÖS-RÉNYI RANDOM GRAPH generation

Parameters: n = 11, m = 15

- Select a pair of nodes uniformly at random. (1, 5)
- If an edge does not exist between the pair, add it.

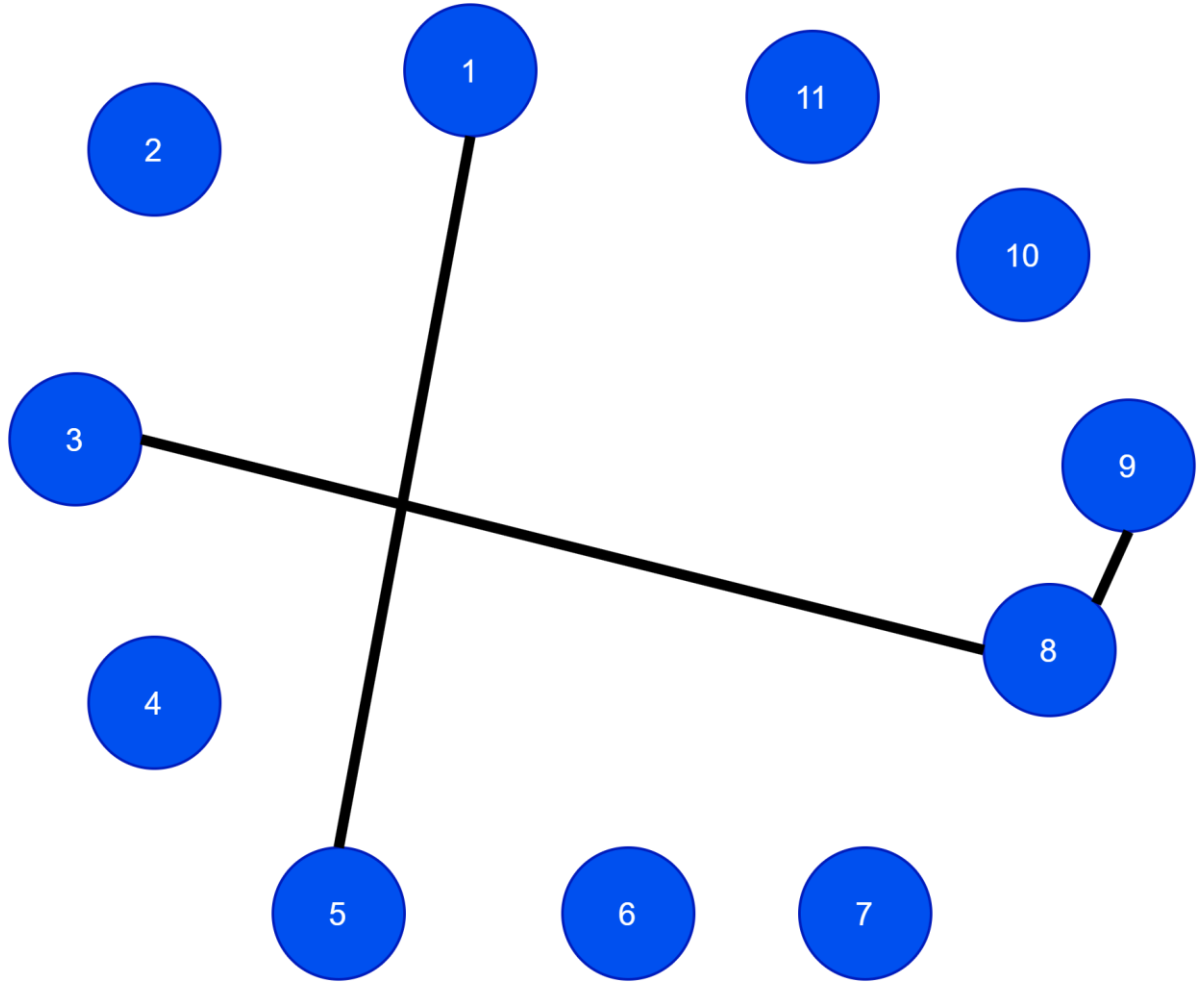


MONTANA STATE UNIVERSITY

ERDÖS-RÉNYI RANDOM GRAPH generation

Parameters: n = 11, m = 15

- Select a pair of nodes uniformly at random. (8,9)
- If an edge does not exist between the pair, add it.

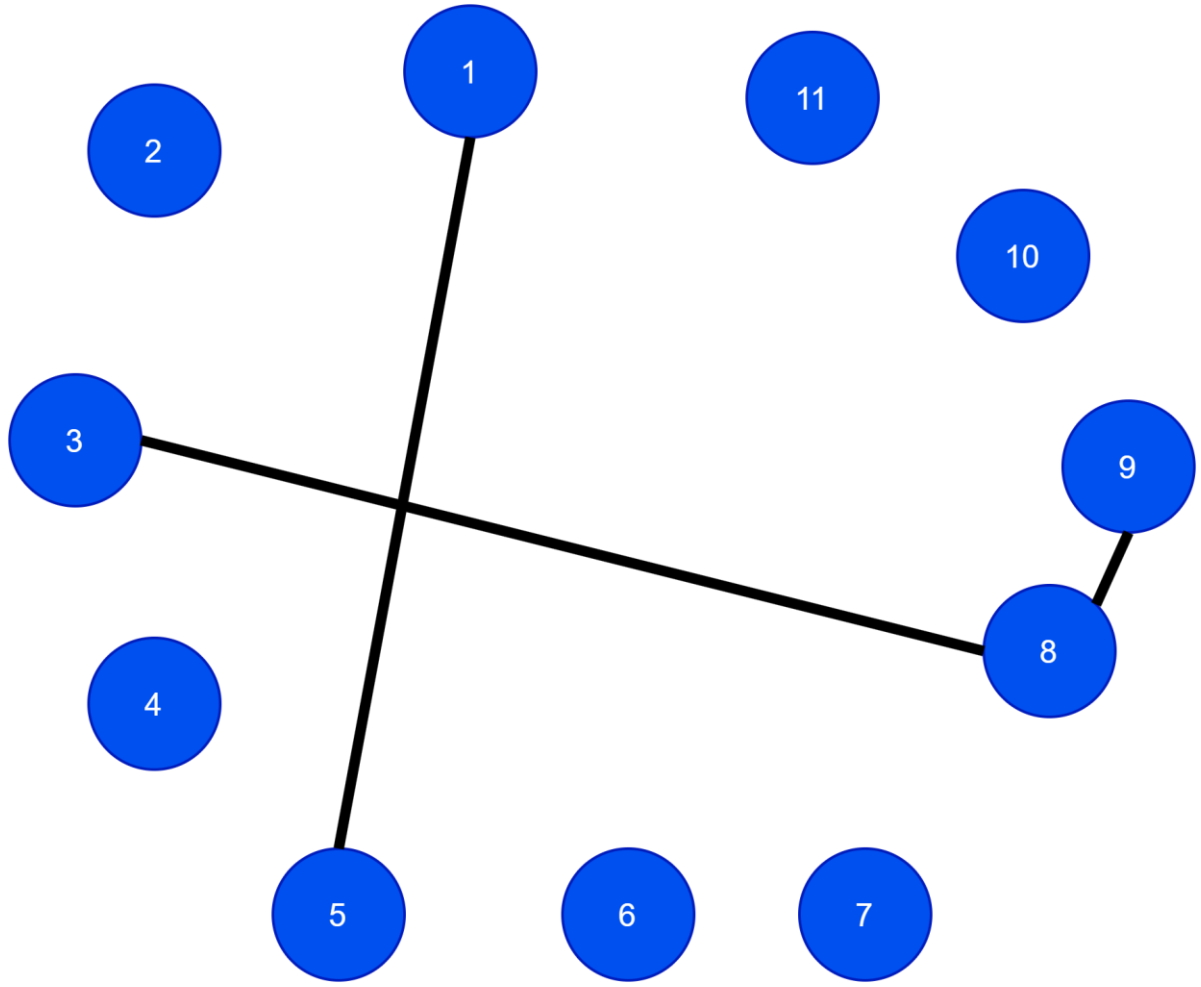


MONTANA STATE UNIVERSITY

ERDÖS-RÉNYI RANDOM GRAPH generation

Parameters: n = 11, m = 15

- Select a pair of nodes uniformly at random. (1,5)
- If an edge does not exist between the pair, add it.
- We skip it as the edge exists in the graph.

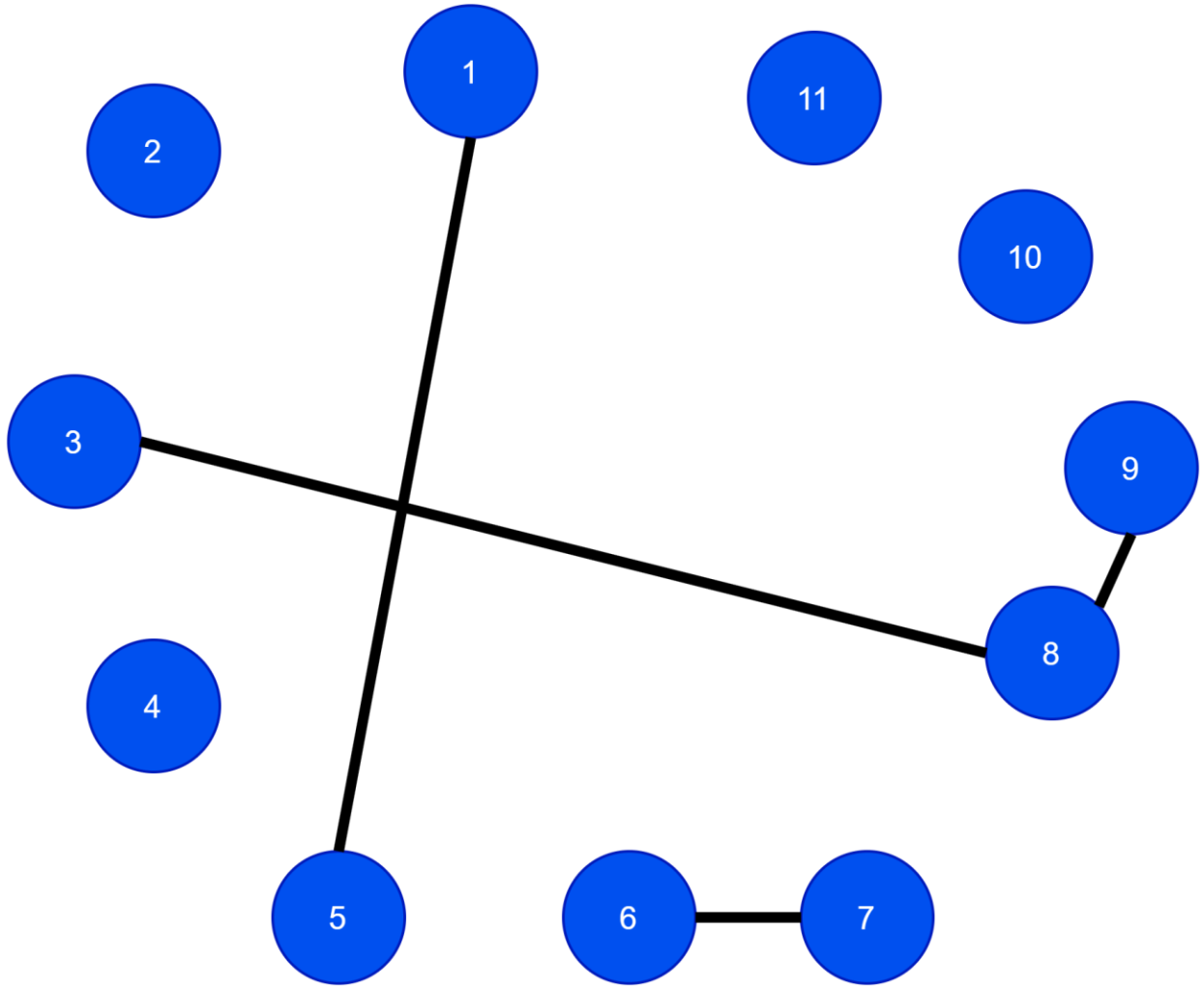


MONTANA STATE UNIVERSITY

ERDÖS-RÉNYI RANDOM GRAPH generation

Parameters: n = 11, m = 15

- Select a pair of nodes uniformly at random. (6,7)
- If an edge does not exist between the pair, add it.

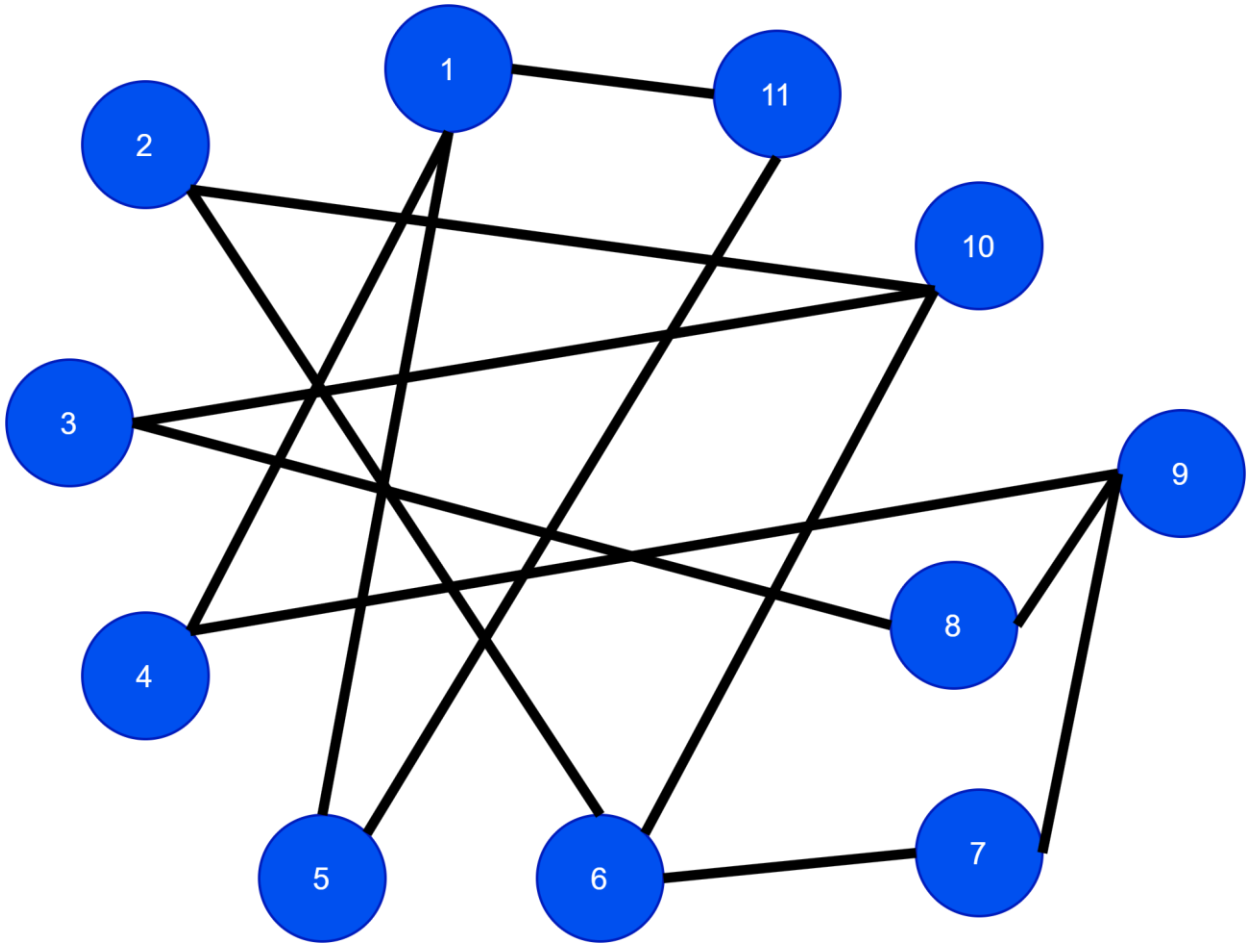


MONTANA STATE UNIVERSITY

ERDÖS-RÉNYI RANDOM GRAPH generation

Parameters: n = 11, m = 15

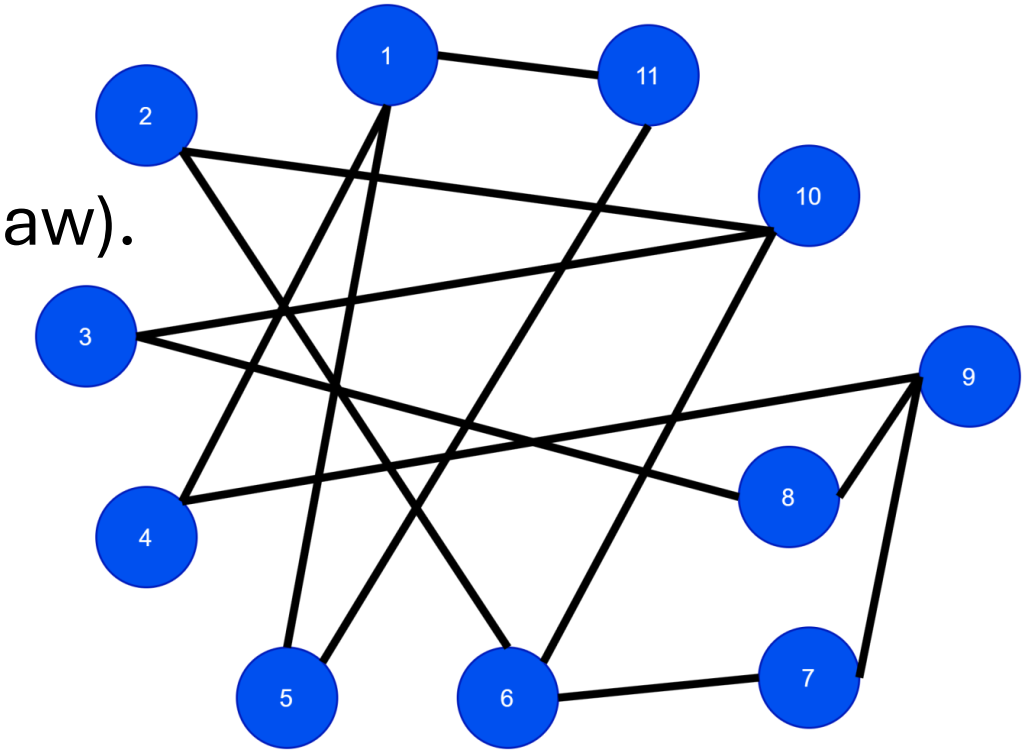
- Select a pair of nodes uniformly at random.
- If an edge does not exist between the pair, add it.
- Do this until m edges exist in the graph.



MONTANA STATE UNIVERSITY

ERDÖS-RÉNYI GRAPH properties

- Small-world behavior
- Poisson degree distribution (not-power law).
- No clustering effect.



BARABÁSI–ALBERT SCALE-FREE MODEL

- The **Barabási-Albert (BA)** model tries to capture the **scale-free degree distributions** of real-world graphs via a generative process that adds new nodes and edges at each time step.
- The edge growth is based on the concept of preferential attachment
  - Edges from the new vertex are more likely to link to nodes with higher degrees.
  - For this reason, this model is also known as ***rich get richer approach***.

BARABÁSI–ALBERT GRAPH generation

The BA model mimics a dynamically growing graph by adding new vertices and edges at each time-step $t = 1, 2, \ldots$.
Let G_t denote the graph at time t, and n_t denote the number of nodes, and m_t denote the number of edges in G_t.

Initialization: The BA model starts at time-step $t = 0$, with an initial graph G_0

with n_0 nodes and m_0 edges.

Each nodes in n_0 should have degree at least one, otherwise it will never be chosen for preferential attachment.

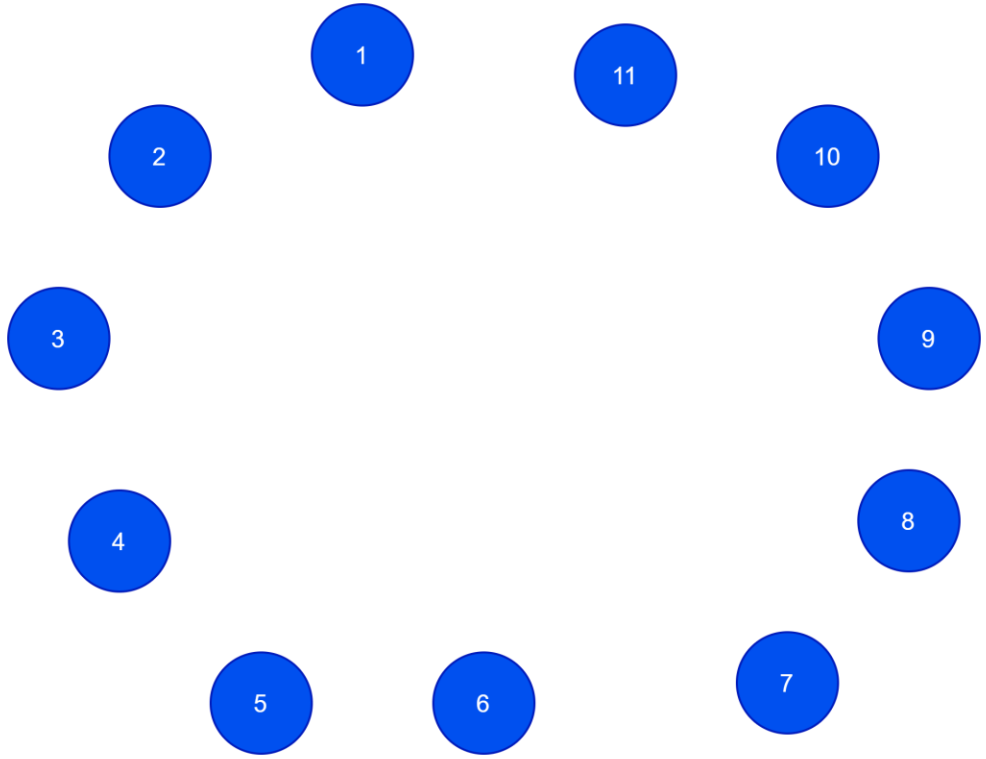
assume that each node has initial degree 2, being connected to its left and right neighbors in a circular layout.

BARABÁSI–ALBERT GRAPH generation

The BA model mimics a dynamically growing graph by adding new vertices

Parameters: $n_0 = 11, m_0 = n_0, q = 3$

- We assume that each node has initial degree 2.
- $m_0 = n_0$



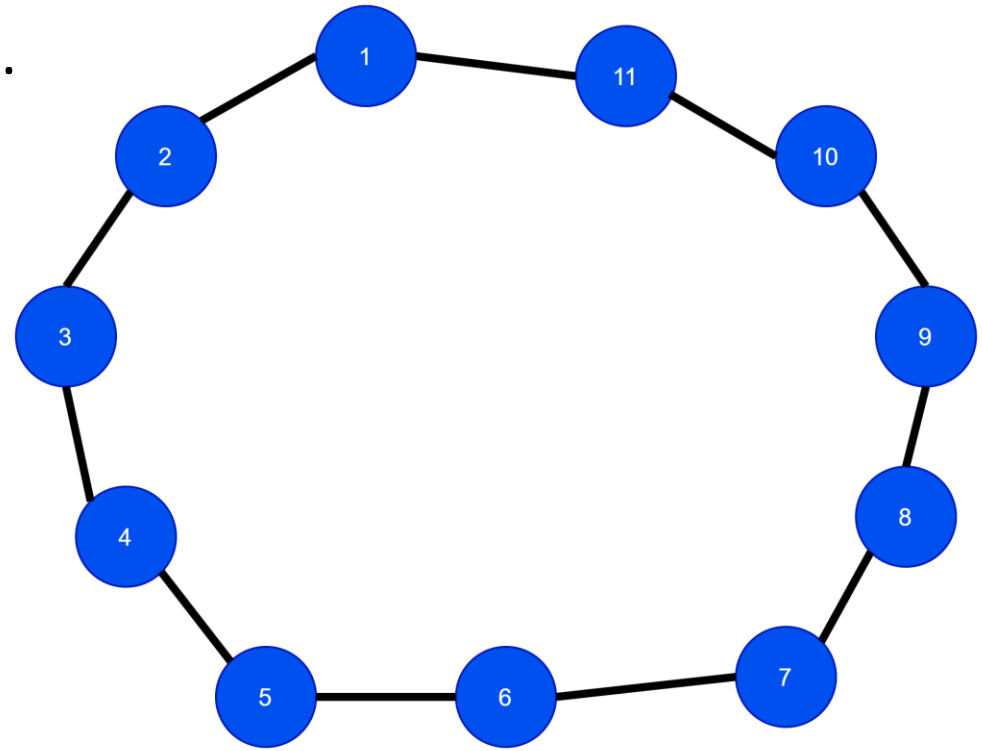
MONTANA STATE UNIVERSITY

BARABÁSI–ALBERT GRAPH generation

The BA model mimics a dynamically growing graph by adding new vertices

Parameters: $n_0 = 11, m_0 = n_0, q = 3$

- We assume that each node has initial degree 2.
- $m_0 = n_0$



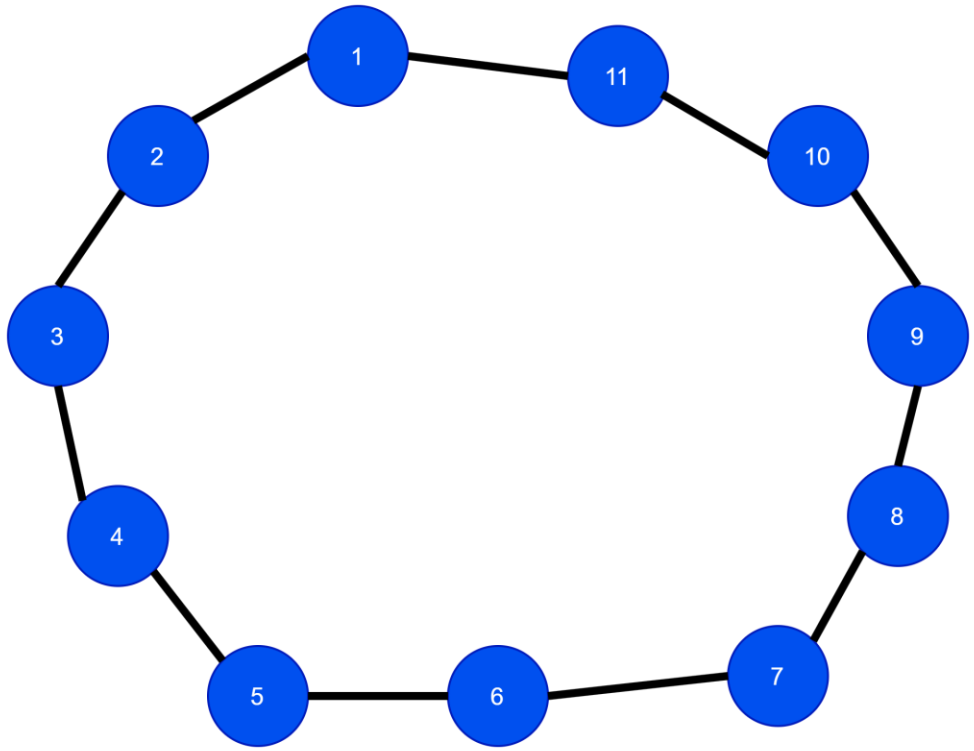
MONTANA STATE UNIVERSITY

BARABÁSI–ALBERT GRAPH generation

The BA model mimics a dynamically growing graph by adding new vertices

Parameters: $n_0 = 11, m_0 = n_0, q = 3$

- We assume that each node has initial degree 2.
- $m_0 = n_0$



The BA model derives a new graph G_{t+1} from G_t by adding exactly one new node u, and adding $q \leq n_0$ new edges from u to q distinct nodes $v_j \in G_t$, where node v_j is chosen with probability $\pi_t(v_j)$ proportional to its degree in G_t , given as:

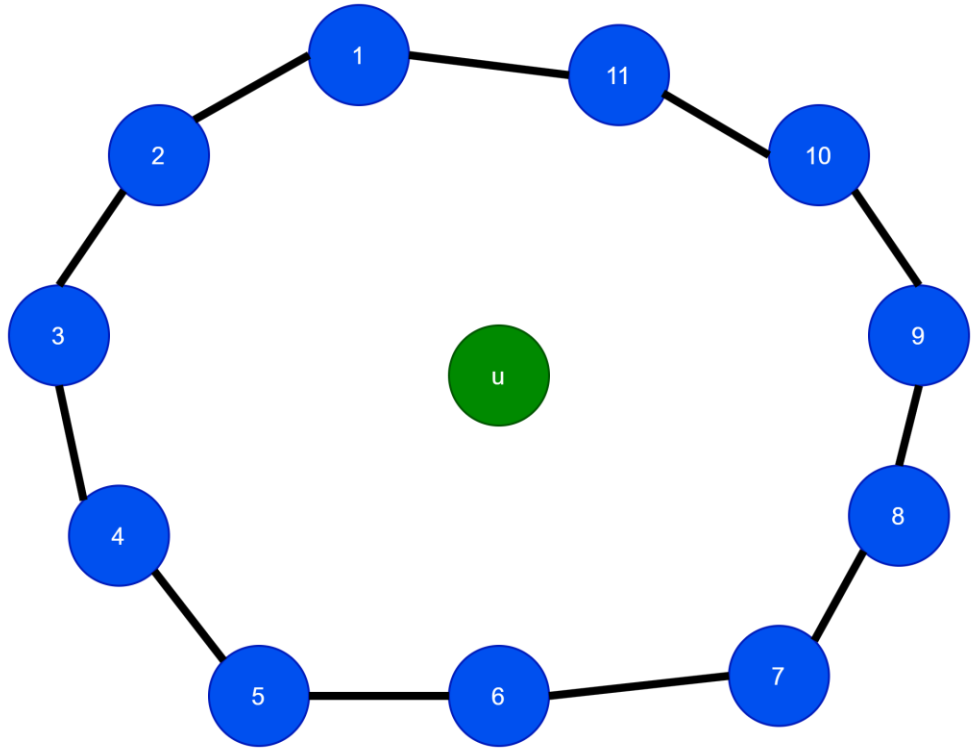
$$\pi_t(v_j) = \frac{d_j}{\sum_{v_i \in G_t} d_i}$$

MONTANA STATE UNIVERSITY

BARABÁSI–ALBERT GRAPH generation

The BA model mimics a dynamically growing graph by adding new vertices

Parameters: $n_0 = 11, m_0 = n_0, q = 3$



The BA model derives a new graph G_{t+1} from G_t by adding exactly one new node u, and adding $q \leq n_0$ new edges from u to q distinct nodes $v_j \in G_t$, where node v_j is chosen with probability $\pi_t(v_j)$ proportional to its degree in G_t , given as:

$t = 0$

$$\pi_t(v_j) = \frac{d_j}{\sum_{v_i \in G_t} d_i}$$

$$\pi_0(3) = \frac{d_3}{\sum_{v_i \in G_t} d_i} = \frac{2}{22} = \frac{1}{11}$$

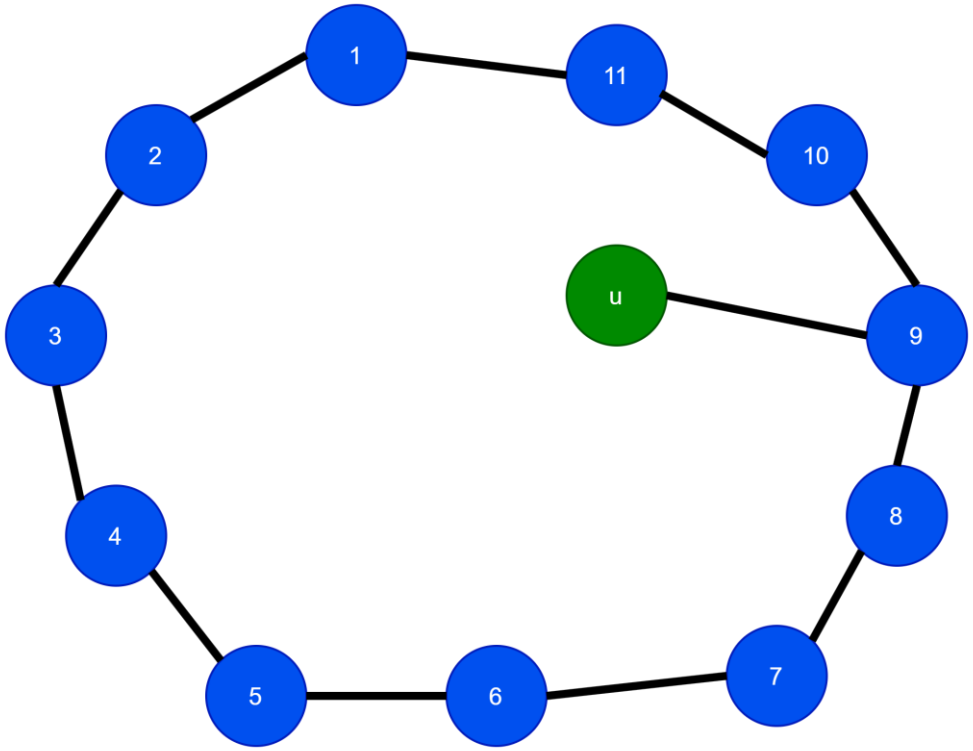
At $t = 0, \forall v_j \in V_t : \pi_t(v_j) = \frac{1}{11}$

MONTANA STATE UNIVERSITY

BARABÁSI–ALBERT GRAPH generation

The BA model mimics a dynamically growing graph by adding new vertices

Parameters: $n_0 = 11, m_0 = n_0, q = 3$



The BA model derives a new graph G_{t+1} from G_t by adding exactly one new node u, and adding $q \leq n_0$ new edges from u to q distinct nodes $v_j \in G_t$, where node v_j is chosen with probability $\pi_t(v_j)$ proportional to its degree in G_t , given as:

$t = 0$

$$\pi_t(v_j) = \frac{d_j}{\sum_{v_i \in G_t} d_i}$$

$$\pi_0(9) = \frac{d_9}{\sum_{v_i \in G_t} d_i} = \frac{3}{24}$$

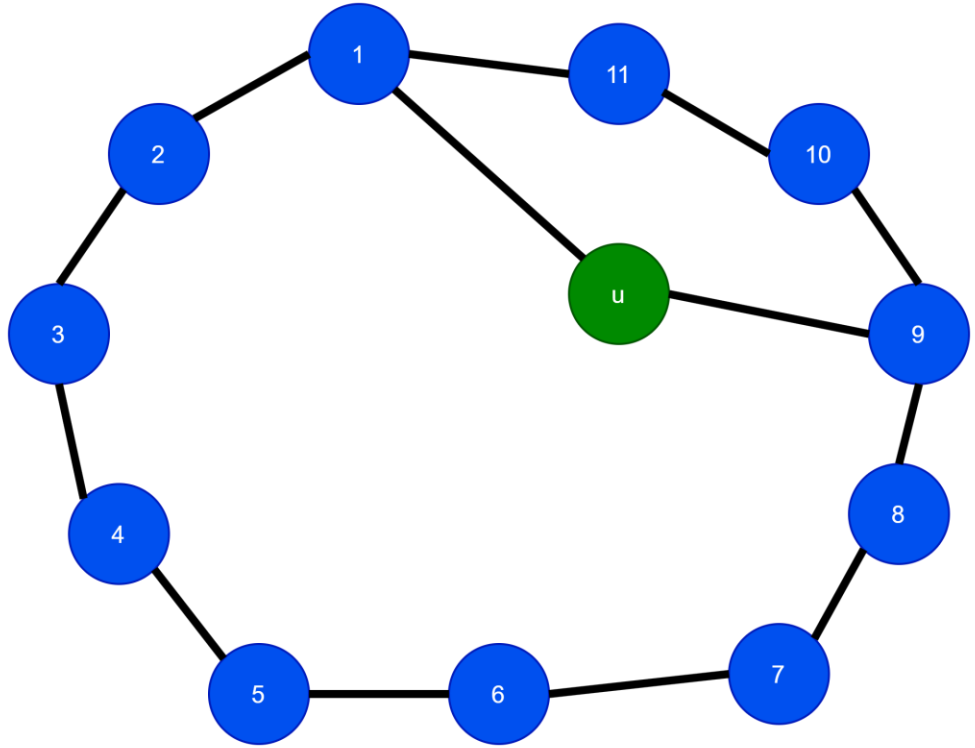
$$\pi_0(3) = \frac{d_3}{\sum_{v_i \in G_t} d_i} = \frac{2}{24}$$

MONTANA STATE UNIVERSITY

BARABÁSI–ALBERT GRAPH generation

The BA model mimics a dynamically growing graph by adding new vertices

Parameters: $n_0 = 11, m_0 = n_0, q = 3$



The BA model derives a new graph G_{t+1} from G_t by adding exactly one new node u, and adding $q \leq n_0$ new edges from u to q distinct nodes $v_j \in G_t$, where node v_j is chosen with probability $\pi_t(v_j)$ proportional to its degree in G_t , given as:

$t = 0$

$$\pi_t(v_j) = \frac{d_j}{\sum_{v_i \in G_t} d_i}$$

$$\pi_0(9) = \pi_0(1) = \frac{3}{26}$$

$$\pi_0(3) = \frac{d_3}{\sum_{v_i \in G_t} d_i} = \frac{2}{26}$$

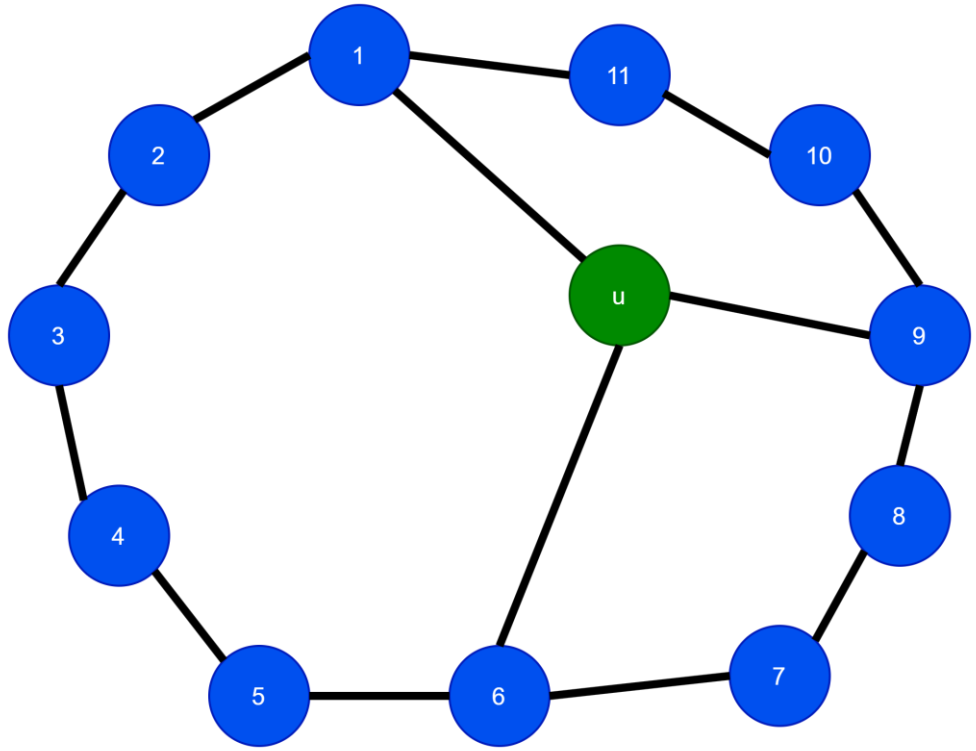
MONTANA STATE UNIVERSITY

BARABÁSI–ALBERT GRAPH generation

The BA model mimics a dynamically growing graph by adding new vertices

Parameters: $n_0 = 11, m_0 = n_0, q = 3$

The BA model derives a new graph G_{t+1} from G_t by adding exactly one new node u, and adding $q \leq n_0$ new edges from u to q distinct nodes $v_j \in G_t$, where node v_j is chosen with probability $\pi_t(v_j)$ proportional to its degree in G_t , given as:



$t = 0$

$$\pi_t(v_j) = \frac{d_j}{\sum_{v_i \in G_t} d_i}$$

$$\pi_0(3) = \frac{d_3}{\sum_{v_i \in G_t} d_i} = \frac{2}{28}$$

$$\pi_0(9) = \pi_0(1) = \pi_0(6) = \frac{3}{28}$$

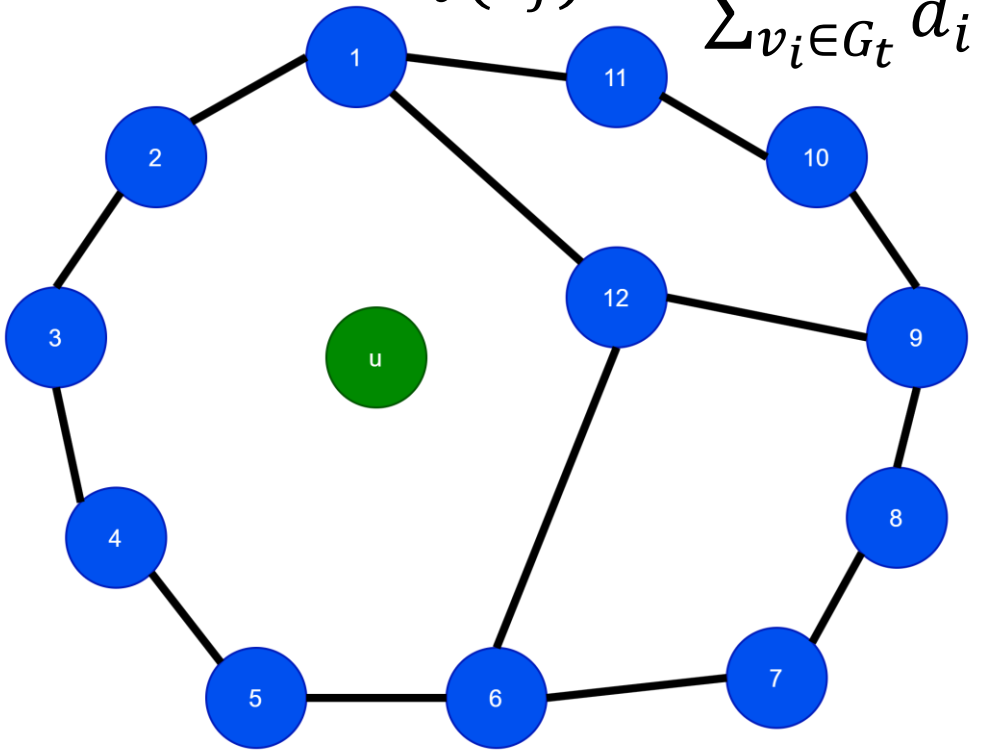
MONTANA STATE UNIVERSITY

BARABÁSI–ALBERT GRAPH generation

$$\pi_t(v_j) = \frac{d_j}{\sum_{v_i \in G_t} d_i}$$

Parameters: $n_0 = 11, m_0 = n_0, q = 3$

The BA model derives a new graph G_{t+1} from G_t by adding exactly one new node u, and adding $q \leq n_0$ new edges from u to q distinct nodes $v_j \in G_t$, where node v_j is chosen with probability $\pi_t(v_j)$ proportional to its degree in G_t , given as:



$\boldsymbol{t = 1}$

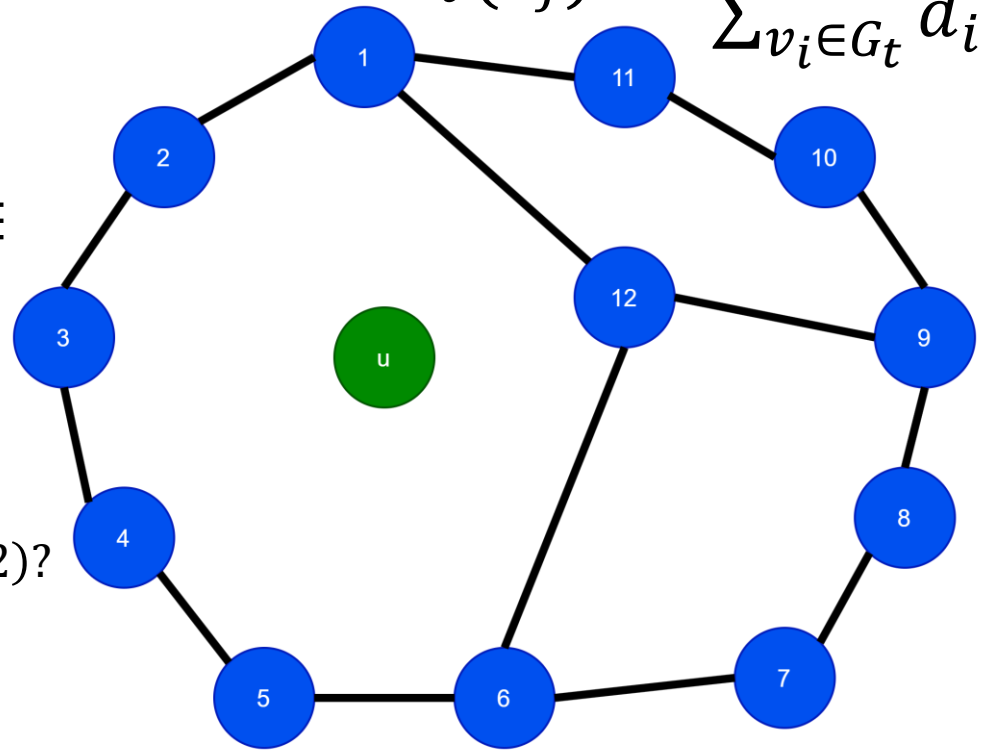
$$\sum_{v_i \in G_t} d_i \; ???$$

BARABÁSI–ALBERT GRAPH generation

Parameters: $n_0 = 11, m_0 = n_0, q = 3$

$$\pi_t(v_j) = \frac{d_j}{\sum_{v_i \in G_t} d_i}$$

The BA model derives a new graph G_{t+1} from G_t by adding exactly one new node u, and adding $q \leq n_0$ new edges from u to q distinct nodes $v_j \in G_t$, where node v_j is chosen with probability $\pi_t(v_j)$ proportional to its degree in G_t , given as:



$t = 1$

$\sum_{v_i \in G_t} d_i = 28$

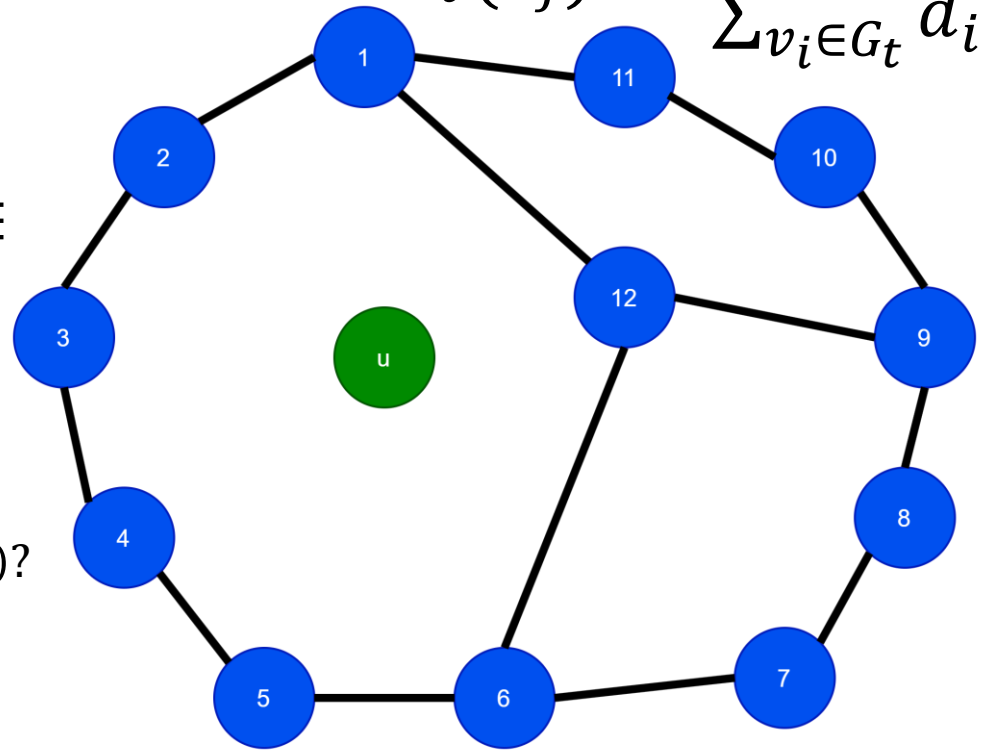
$\pi_1(1), \pi_1(2), \pi_1(3), \pi_1(4), \pi_1(5), \ldots, \pi_1(12)?$

BARABÁSI–ALBERT GRAPH generation

Parameters: $n_0 = 11, m_0 = n_0, q = 3$

$$\pi_t(v_j) = \frac{d_j}{\sum_{v_i \in G_t} d_i}$$

The BA model derives a new graph G_{t+1} from G_t by adding exactly one new node u, and adding $q \leq n_0$ new edges from u to q distinct nodes $v_j \in G_t$, where node v_j is chosen with probability $\pi_t(v_j)$ proportional to its degree in G_t , given as:



$t = 1$

$\sum_{v_i \in G_t} d_i = 28$

$\pi_1(1), \pi_1(2), \pi_1(3), \pi_1(4), \pi_1(5), \ldots, \pi_1(12)?$

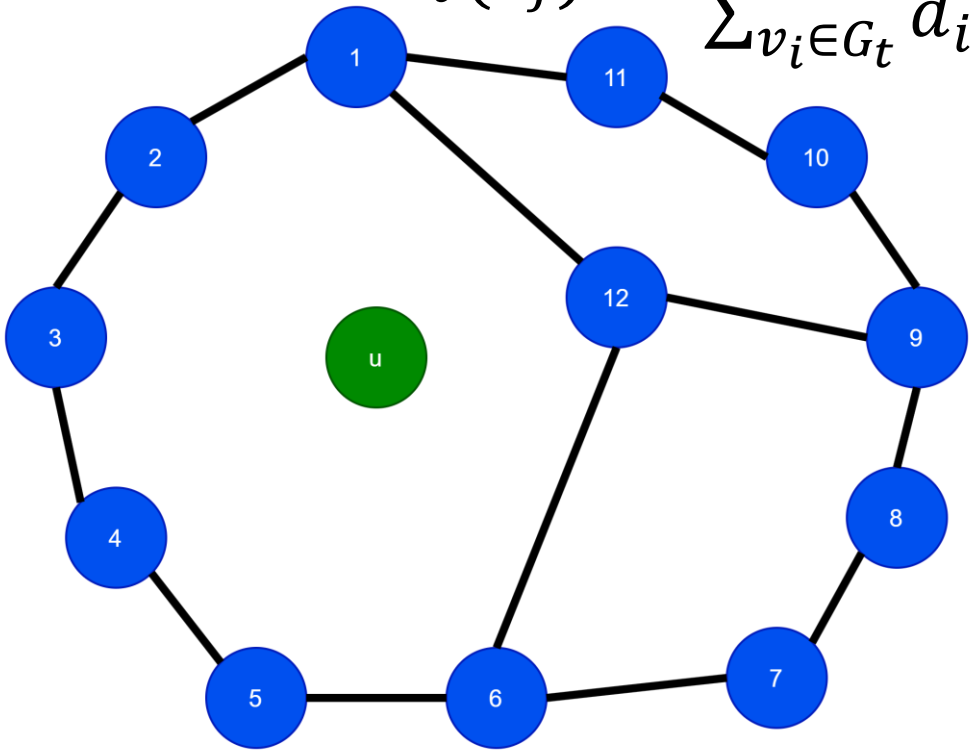
$\pi_1(2)??$

BARABÁSI–ALBERT GRAPH generation

$$\pi_t(v_j) = \frac{d_j}{\sum_{v_i \in G_t} d_i}$$

Parameters: $n_0 = 11, m_0 = n_0, q = 3$

The BA model derives a new graph G_{t+1} from G_t by adding exactly one new node u, and adding $q \leq n_0$ new edges from u to q distinct nodes $v_j \in G_t$, where node v_j is chosen with probability $\pi_t(v_j)$ proportional to its degree in G_t , given as:



$t = 1$

$\sum_{v_i \in G_t} d_i = 28$ $\pi_1(1), \pi_1(2), \pi_1(3), \pi_1(4), \pi_1(5), \ldots, \pi_1(12)$?

$$\pi_1(2) = \frac{d_2}{\sum_{v_i \in G_t} d_i} = \frac{2}{28}$$

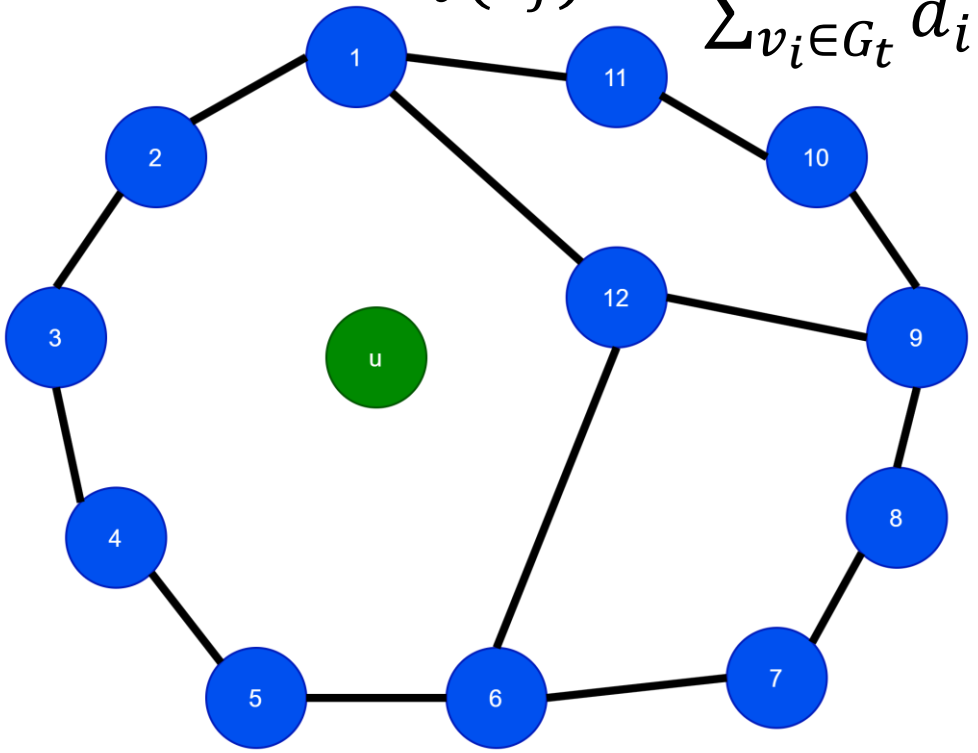
MONTANA STATE UNIVERSITY

BARABÁSI–ALBERT GRAPH generation

$$\pi_t(v_j) = \frac{d_j}{\sum_{v_i \in G_t} d_i}$$

Parameters: $n_0 = 11, m_0 = n_0, q = 3$

The BA model derives a new graph G_{t+1} from G_t by adding exactly one new node u, and adding $q \leq n_0$ new edges from u to q distinct nodes $v_j \in G_t$, where node v_j is chosen with probability $\pi_t(v_j)$ proportional to its degree in G_t , given as:



$t = 1$

$\sum_{v_i \in G_t} d_i = 28$ $\pi_1(1), \pi_1(2), \pi_1(3), \pi_1(4), \pi_1(5), \ldots, \pi_1(12)$?

$$\pi_1(2) = \frac{1}{14} \qquad \pi_1(3) = \frac{d_3}{\sum_{v_i \in G_t} d_i} = \frac{2}{28}$$

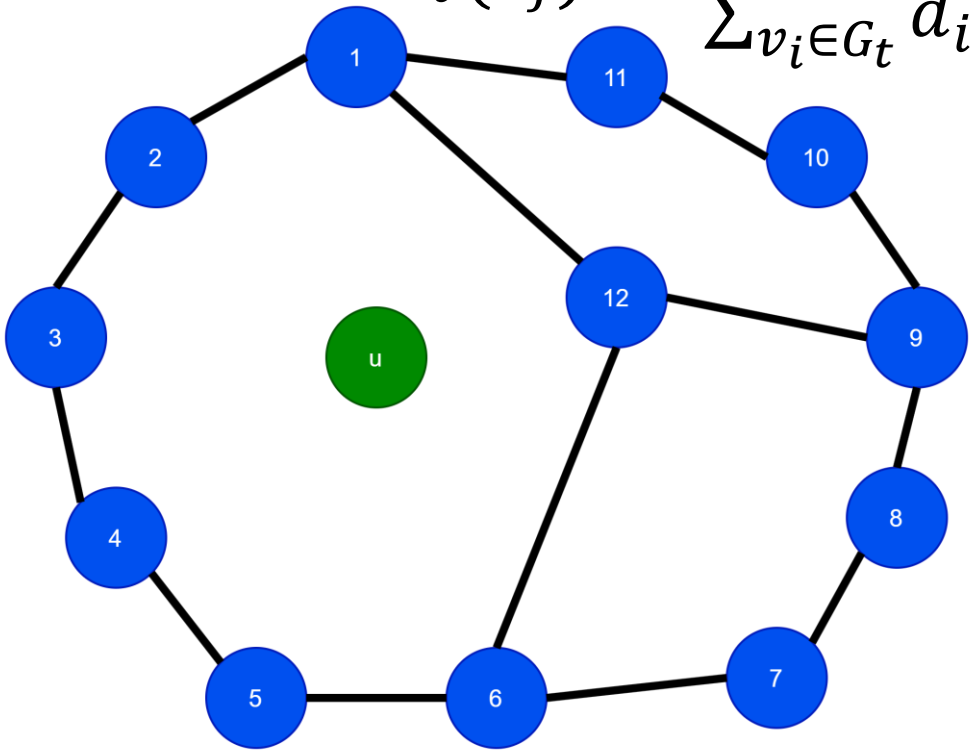
MONTANA STATE UNIVERSITY

BARABÁSI–ALBERT GRAPH generation

Parameters: $n_0 = 11, m_0 = n_0, q = 3$

$$\pi_t(v_j) = \frac{d_j}{\sum_{v_i \in G_t} d_i}$$

The BA model derives a new graph G_{t+1} from G_t by adding exactly one new node u, and adding $q \leq n_0$ new edges from u to q distinct nodes $v_j \in G_t$, where node v_j is chosen with probability $\pi_t(v_j)$ proportional to its degree in G_t , given as:



$t = 1$

$\sum_{v_i \in G_t} d_i = 28$ $\pi_1(1), \pi_1(2), \pi_1(3), \pi_1(4), \pi_1(5), \dots, \pi_1(12)$?

$\pi_1(2) = \frac{1}{14}$ $\pi_1(3) = \frac{2}{28}$ $\pi_1(12) = \frac{3}{28}$ $\pi_1(6) = \frac{3}{28}$

$\pi_1(12) = \pi_1(1) = \pi_1(9) = \pi_1(6) = \frac{3}{28}$

All the other nodes have $\frac{1}{14}$ chance to get picked

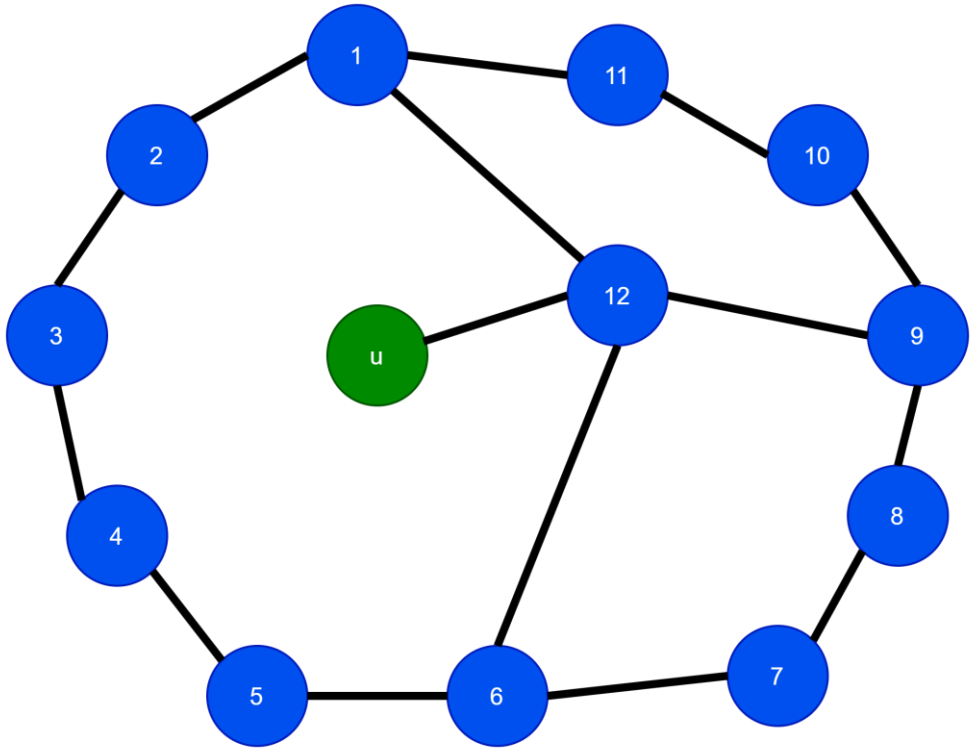
MONTANA STATE UNIVERSITY

BARABÁSI–ALBERT GRAPH generation

$$\pi_t(v_j) = \frac{d_j}{\sum_{v_i \in G_t} d_i}$$

Parameters: $n_0 = 11, m_0 = n_0, q = 3$

The BA model derives a new graph G_{t+1} from G_t by adding exactly one new node u, and adding $q \leq n_0$ new edges from u to q distinct nodes $v_j \in G_t$, where node v_j is chosen with probability $\pi_t(v_j)$ proportional to its degree in G_t , given as:



$t = 1$

$\sum_{v_i \in G_t} d_i = 30$ $\pi_1(1), \pi_1(2), \pi_1(3), \pi_1(4), \pi_1(5), \ldots, \pi_1(12)$?

$$\pi_1(2) = \frac{2}{30} \quad \pi_1(1) = \frac{3}{30} \quad \pi_1(12) = \frac{d_{12}}{\sum_{v_i \in G_t} d_i} = \frac{4}{30}$$

$$\pi_1(6) = \frac{3}{30}$$

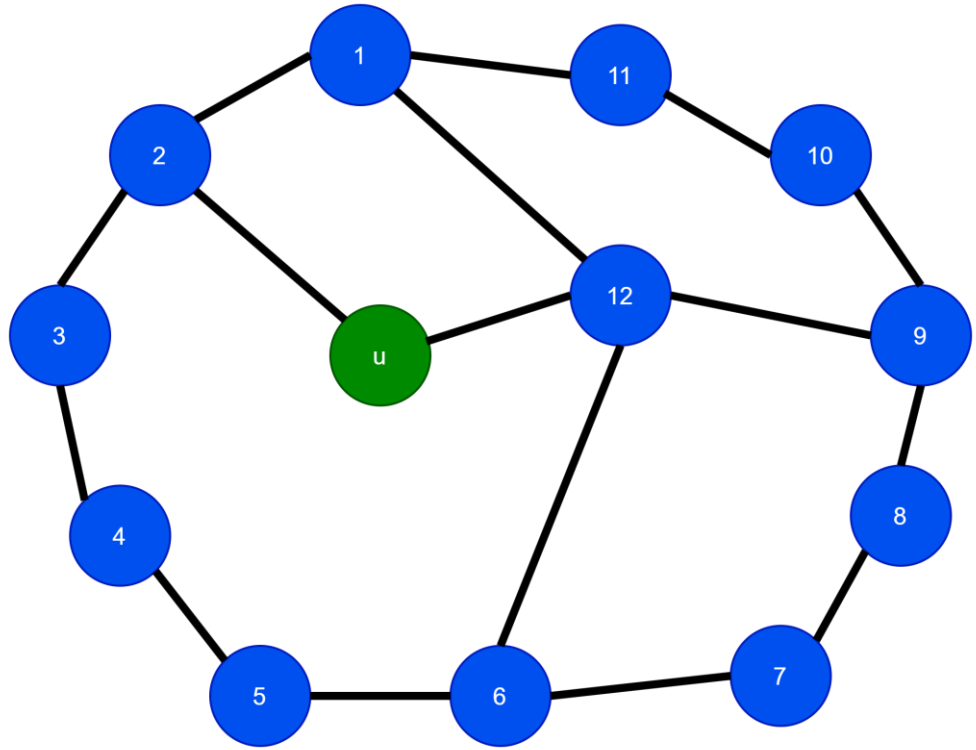
MONTANA STATE UNIVERSITY

BARABÁSI–ALBERT GRAPH generation

$$\pi_t(v_j) = \frac{d_j}{\sum_{v_i \in G_t} d_i}$$

Parameters: $n_0 = 11, m_0 = n_0, q = 3$

The BA model derives a new graph G_{t+1} from G_t by adding exactly one new node u, and adding $q \leq n_0$ new edges from u to q distinct nodes $v_j \in G_t$, where node v_j is chosen with probability $\pi_t(v_j)$ proportional to its degree in G_t , given as:



$t = 1$

$\sum_{v_i \in G_t} d_i = 32$ $\pi_1(1), \pi_1(2), \pi_1(3), \pi_1(4), \pi_1(5), \ldots, \pi_1(12)$?

$$\pi_1(2) = \frac{3}{32} \quad \pi_1(1) = \frac{3}{32} \quad \pi_1(12) = \frac{d_{12}}{\sum_{v_i \in G_t} d_i} = \frac{4}{32}$$

$$\pi_1(6) = \frac{3}{32}$$

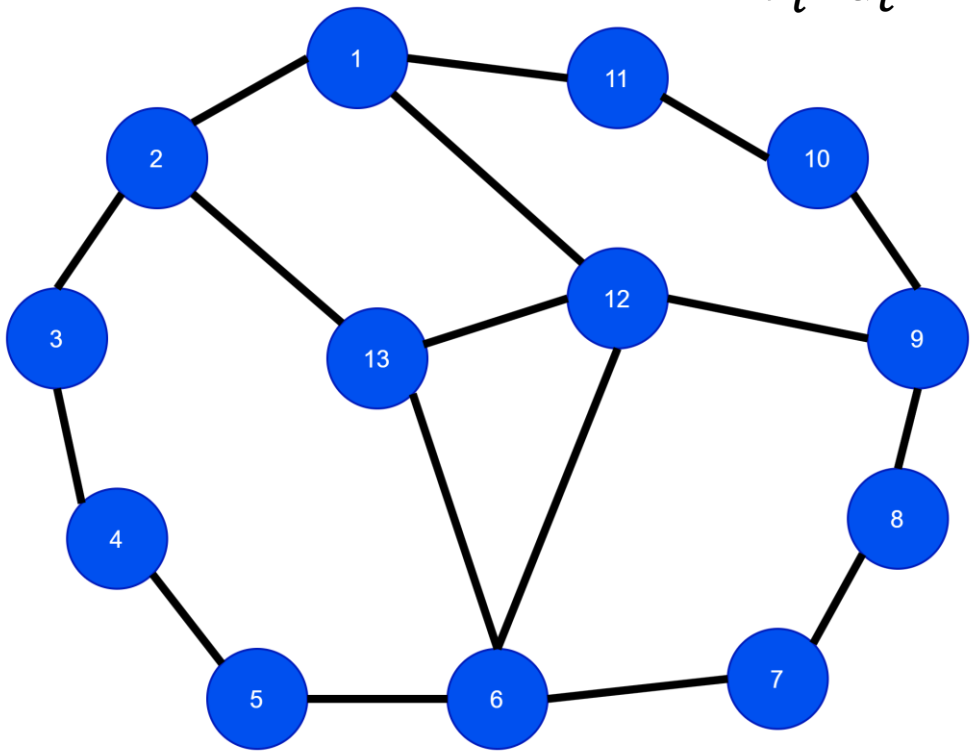
MONTANA STATE UNIVERSITY

BARABÁSI–ALBERT GRAPH generation

Parameters: $n_0 = 11, m_0 = n_0, q = 3$

$$\pi_t(v_j) = \frac{d_j}{\sum_{v_i \in G_t} d_i}$$

The BA model derives a new graph G_{t+1} from G_t by adding exactly one new node u, and adding $q \leq n_0$ new edges from u to q distinct nodes $v_j \in G_t$, where node v_j is chosen with probability $\pi_t(v_j)$ proportional to its degree in G_t , given as:



$t = 1$, step 1 completed

$$\sum_{v_i \in G_t} d_i = 34$$

$$\pi_1(2) = \frac{3}{34} \quad \pi_1(1) = \frac{3}{34} \quad \pi_1(12) = \frac{d_{12}}{\sum_{v_i \in G_t} d_i} = \frac{4}{34}$$

$$\pi_1(6) = \frac{4}{34}$$

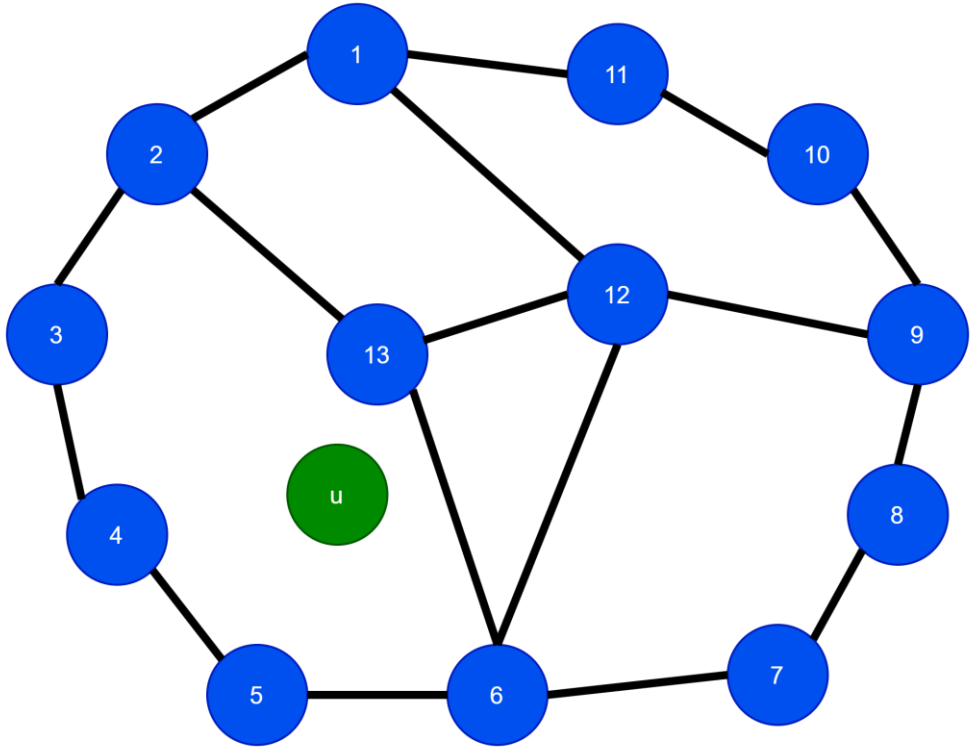
MONTANA STATE UNIVERSITY

BARABÁSI–ALBERT GRAPH generation

Parameters: $n_0 = 11, m_0 = n_0, q = 3$

$$\pi_t(v_j) = \frac{d_j}{\sum_{v_i \in G_t} d_i}$$

The BA model derives a new graph G_{t+1} from G_t by adding exactly one new node u, and adding $q \leq n_0$ new edges from u to q distinct nodes $v_j \in G_t$, where node v_j is chosen with probability $\pi_t(v_j)$ proportional to its degree in G_t , given as:



$t = 2$, we continue.....

$$\sum_{v_i \in G_t} d_i = 34$$

$\pi_2(1), \pi_2(2), \pi_2(3), \pi_2(4), \pi_2(5), \ldots, \pi_2(12)?$

$$\pi_2(2) = \frac{3}{34} \quad \pi_2(1) = \frac{3}{34} \quad \pi_2(12) = \frac{4}{34}$$

$$\pi_2(6) = \frac{4}{34}$$

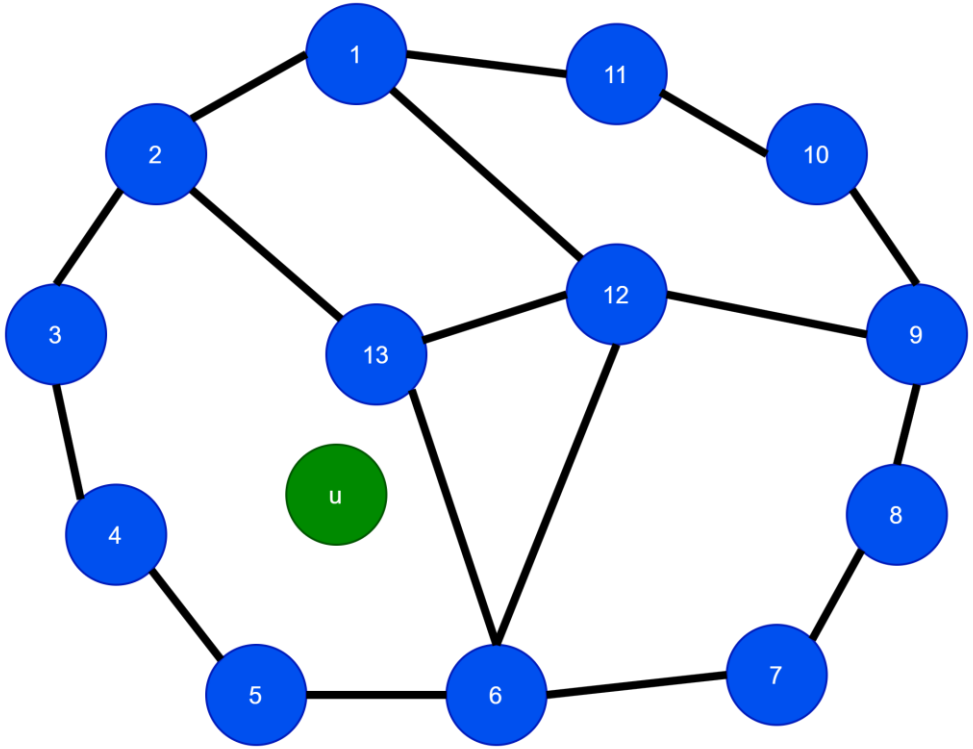
MONTANA STATE UNIVERSITY

BARABÁSI–ALBERT GRAPH generation

Parameters: $n_0 = 11, m_0 = n_0, q = 3$

$$\pi_t(v_j) = \frac{d_j}{\sum_{v_i \in G_t} d_i}$$

You can see why this model follow rich get richer principle.



MONTANA STATE UNIVERSITY

BARABÁSI–ALBERT GRAPH properties

- Ultra-Small-world behavior
- Power-law degree distribution.
- No clustering effect.

Watts-Strogatz Small-world Graph Model

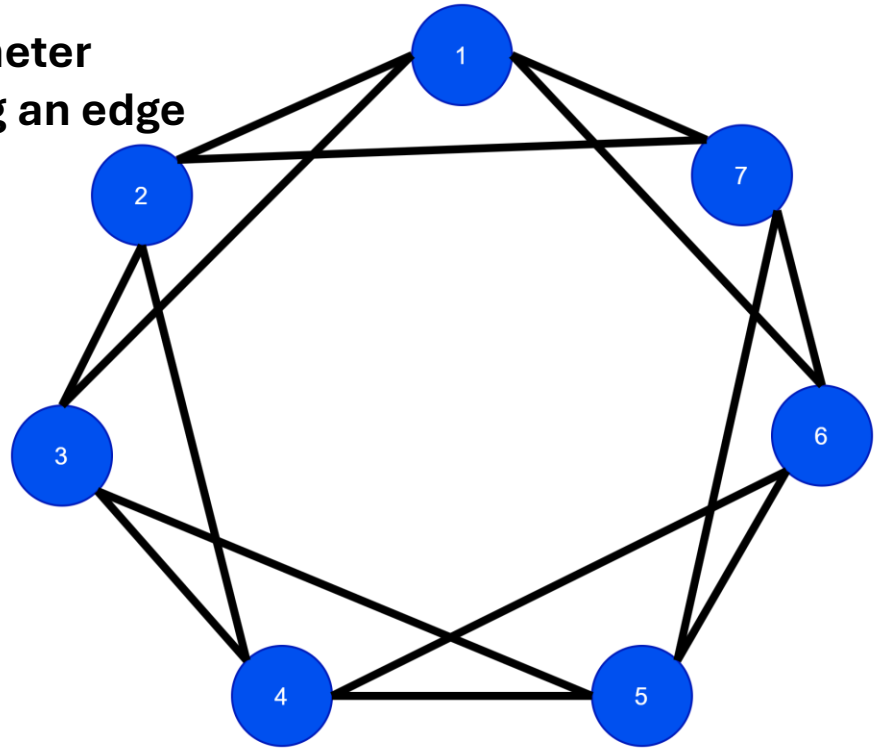
- Watts-Strogatz (WS) model tries to explicitly model high local clustering by starting with a regular network.
  - k- regular graph is a graph where each node is connected to k nodes.
- Properties
  - Degree distribution follows a binomial distribution.
  - High clustering effect

Watts-Strogatz Graph generation

- Watts-Strogatz model starts with n nodes arranged in a circular layout, with each node connected to its immediate left and right neighbors.
- The edges in the initial layout are called backbone edges.
- Each node has edges to an additional k – 1 neighbors to the left and right.
- Thus, initial graph is regular graph of degree $2k$.

Watts-Strogatz Graph generation

n = number of nodes
k = Initial topology parameter
P = probability of rewiring an edge



Initial graph with $n = 7, k = 2$.
Note that this is a 4 −regular graph

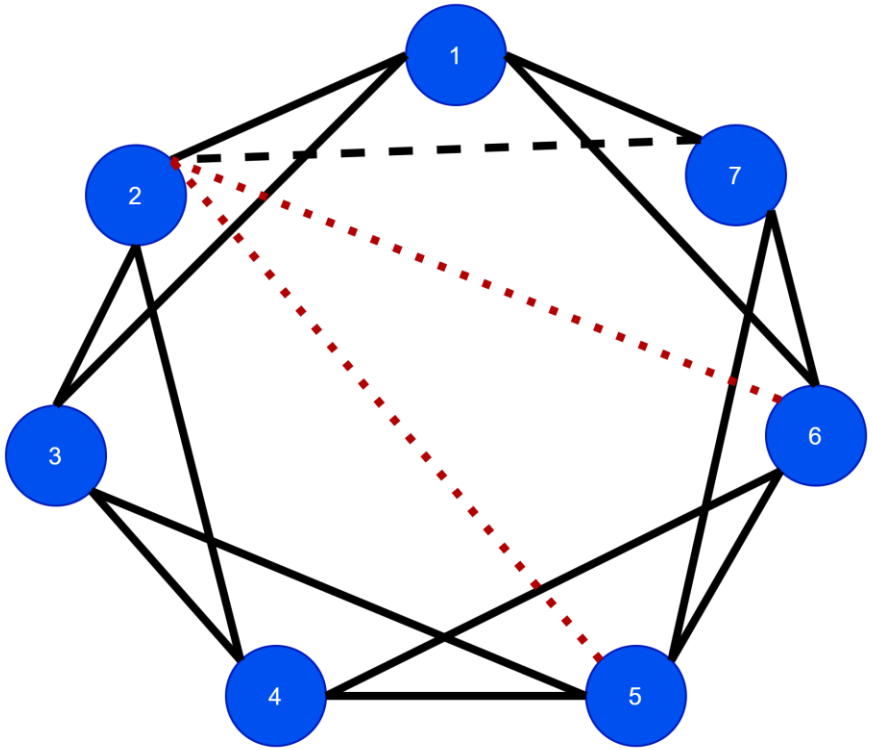
Watts-Strogatz Graph generation

Initial graph with $n = 7, k = 2$. Note that this is a 4 –regular graph

n = number of nodes
k = Initial topology parameter
p = probability of rewiring an edge

Edge Rewiring : Watts-Strogatz model perturbs the regular structure by adding some randomness to the network.

Randomly rewire edges with probability p, i.e., for each edge (u, v) in the graph, with probability p, replace v with another randomly chosen node avoiding loops (no self-loops) and duplicate edges.
In other words, remove already existing edge and rewire it somewhere else.



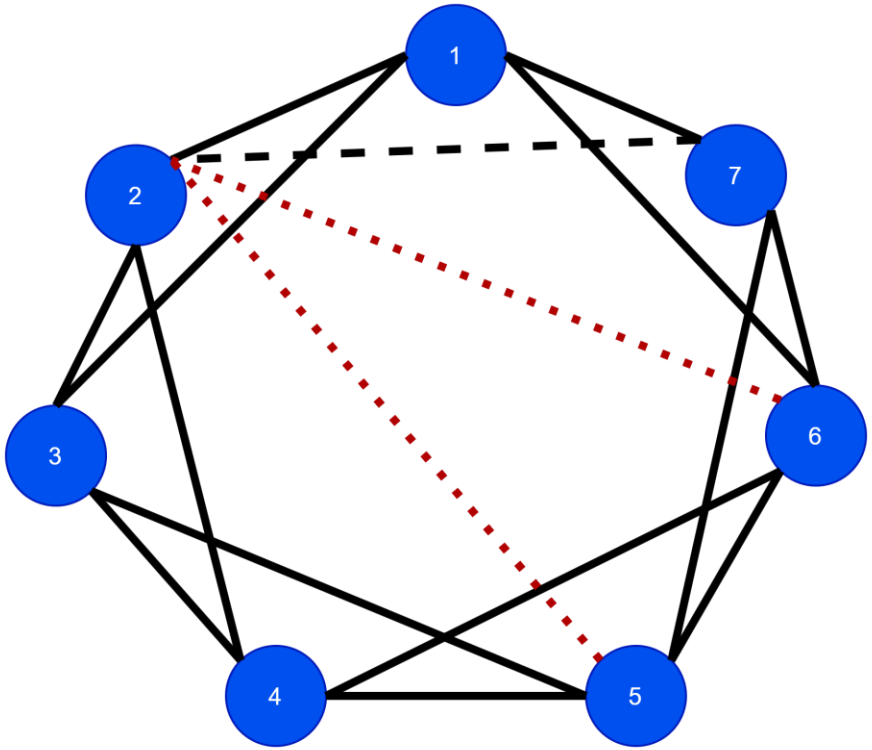
Watts-Strogatz Graph generation

Initial graph with $n = 7, k = 2$. Note that this is a 4 –regular graph

n = number of nodes
k = Initial topology parameter
p = probability of rewiring an edge

What actually happens?

- **at $p = 0$: The graphs is a regular lattice with very high clustering coefficients**
- **At $0 < p < 1$: Graph develops shortcuts, reducing the average path length while still maintaining high clustering.**
- **At $p = 1$: The graph becomes random like, resembling an Erdos-Renyi graph.**
- **We can see this in the code example.**



Watts-Strogatz Small-world Graph Model

- Properties
  - Degree distribution follows a binomial distribution. (No power-law)
  - High clustering effect
  - Small-world property