A BRIEF OVERVIEW OF GRAPH THEORY: ERDOS-RENYI RANDOM GRAPH MODEL AND SMALL WORLD PHENOMENON

JIATONG LI (LOGEN)

ABSTRACT. This paper briefly introduces graph theory, Erdos-Renyi random graph model, small world phenomenon, and its application in the engineering industry as supplemental material. We begin with basic definitions and notations in graph theory, then move to graph networks' properties. Then, finally, we utilize the definitions and theorem we introduced to analyze the Erdos-Renyi random graph model and the small world phenomenon. At the end of the paper, we also point out some further applications of graph theory.

CONTENTS

1. INTRODUCTION

A graph is a structure which contains a set of objects - vertices and edges. A basic example is when we connect two vertices with a line, we create a simple graph. However, such a simple network can quickly become very complex to analyze as the number of vertices and edges increases. To simulate what's happening, we use a random graph as a model in this paper. We will also explore some extensions of random graph model such as small world phenomenon.

We begin with basic definitions and notations of graph theory. Section 3 will introduce four important properties of graph networks: degree distribution, path length, clustering coefficient, and connected component. Then we apply the knowledge from Section 2 and Section 3 to analyze Erdos-Renyi random graph. Next, we discuss the small world phenomenon - the principle that we are all connected by our nearby neighbors. Finally, we zoom out and briefly mention the application of graph theory in other subjects such as engineering.

2. Basic Definitions and Notations

Definition 2.1. Let $V(G)$ and $E(G)$ denote the vertex and the edge set of simple graph G, respectively.

Definition 2.2. The order of a graph G is the cardinality of the vertex set, denoted as $|V(G)|$. The size of G is defined as the cardinality of the edge set, denoted as $|E(G)|$.

Example 2.3. In Figure 1, $V(G) = \{1, 2, 3, 4\}$ and $E(G) = \{13, 23, 12, 24\}$. Based on Definition 2.2, the graph has an order of 4 and a size of 4.

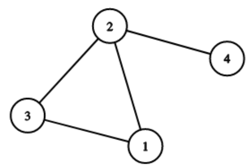

Figure 1

Definition 2.4. The degree of a vertex is the number of edges that are connected to that vertex. When the degree of v is 5, we write as $deg(v) = 5$. The minimum degree of G is symbolized by $\delta(G)$, and the maximum degree is symbolized by $\Delta(G)$.

Definition 2.5. The degree sequence of G is a list of degrees of all vertices. By convention, we list them in a nonincreasing order.

Theorem 2.6. *The sum of the degrees of all vertices is twice the size of the graph.*

$$\sum \deg(v_i) = 2|E(G)|$$

Lemma 2.7. *(Pigeonhole Principle)*
If we put n pigeons in less than n pigeonhole, at least one pigeonhole is going to contain more than one pigeons.

Definition 2.8. If an edge e joins two vertices v_1 and v_2, we say the two vertices are adjacent. If an edge e joins two vertices v_1 and v_2, we say edge e is incident upon v_1 and v_2.

Definition 2.9. A simple graph is a graph that doesn't have multiple edges and loops. A loop is created when the starting vertex and ending vertex are the same.

Definition 2.10. A complete graph is a graph in which every pair of vertices is connected by a unique edge.

Example 2.11. Figure 2 is a complete connected graph, whereas Figure 1 is a simple graph.

Lemma 2.12. *The maximum number of edges in a connected graph H equals to* $E_{\max} = \binom{N}{2}$.

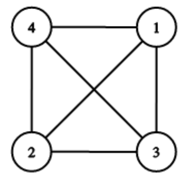

FIGURE 2

Proof. The proof is relatively intuitive. The maximum number equals to "N choose 2" (choose 2 vertices among N vertices). □

Definition 2.13. A walk is a finite number of sequences in the form of v_0v_1, v_1v_2, $\cdots$, $v_{m-1}v_m$. One can also write $v_0 \to v_1 \to v_2 \to \cdots$. By definition, v_0 is called the initial vertex, and v_m is called the final vertex. The number of edges in the walk is called length.

Definition 2.14. In Definition 2.13, if the vertices are distinct, then the walk can be called a path. Moreover, when $v_0 = v_m$, the walk is a cycle.

Definition 2.15. An undirected graph means the edge is bidirectional. By contrast, if the edge of a graph points to a specific direction, it is called directed graph.

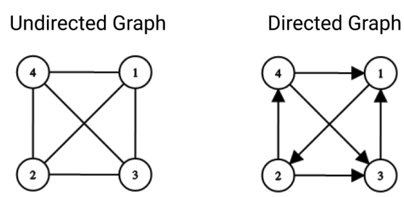

FIGURE 3

Definition 2.16. The adjacency matrix A of a graph G is $n \times n$, where $n = |V(G)|$, and is defined as:

$$A_{i,j} = \begin{cases} 1 & v_i \sim v_j \\ 0 & \text{otherwise} \end{cases}$$

3. Properties of Graph Networks

Definition 3.1. We define the degree distribution $P(k)$ of a graph as the probability of a randomly chosen node has degree k. If a graph has N_k vertices of degree k, then $P(k) = \frac{N_k}{|V(G)|}$.

Example 3.2. Figure 4 is a simple graph and we plot its degree distribution bar plot in Figure 5. For example, the degree distribution of $k = 1$ equals to $\frac{1}{6}$ since there is only one node with degree 1 among the six nodes.

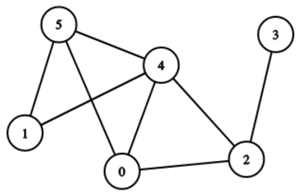

FIGURE 4. A simple graph

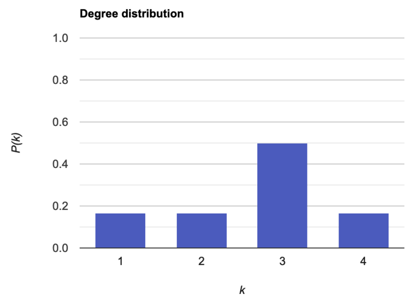

FIGURE 5. Degree distribution bar plot of Figure 4

Definition 3.3. The distance between a pair of nodes is defined as the shortest path connecting the nodes. If two nodes are not connected, the distance is defined as infinite.

Example 3.4. In Figure 4, $h_{12} = 2$ and $h_{13} = 3$.

Definition 3.5. We define the clustering coefficient of a graph as $C_i = \dfrac{2e_i}{k_i\,(k_i - 1)}$ where k_i is a degree of node i and e_i is the number of edges among the adjacent vertices of node i.

Remark 3.6. In Definition 3.5, we should notice that $\dfrac{k_i(k_i-1)}{2}$ is the maximum number of edges among the adjacent vertices of node i. Note that $C_i \in [0, 1]$.

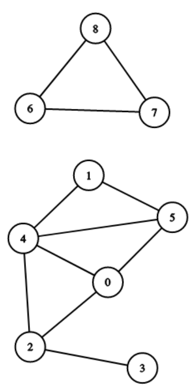

FIGURE 6

Example 3.7. In Figure 6, the vertex set of the graph $G(V, E)$ is $V(G) = \{0, 1, 2, 3, 4, 5, 6, 7, 8\}$. The graph G contains two subgraphs G_1 and G_2 where $V(G_1) = \{0, 1, 2, 3, 4, 5\}$ and $V(G_2) = \{6, 7, 8\}$. Since G_2 contains more elements, it is the largest component of graph G.

Remark 3.8. In computer science, there is a special type of algorithm called Depth-first search (DFS) which can calculate the number of connected components in a given graph.

Algorithm 1: Finding the number of connected components using DFS

```
input  : An undirected graph G(V, E)
output: An int value which indicates the number of connected components
count = 0;
for each vertex k ∈ V do
 | Visited[k] = false;
end
/* Initialize all nodes to status "unvisited"                                  */
for each vertex k ∈ V do
 |  Visited[k] = false;
 |  if Visited[k]==false then
 |  |  DFS(V, k);
 |  |  count = count + 1;
 |  end
end
Print count;
/* Recursively enumerate all nodes and count the number of connected component
   */
DFS(V, k) function: Visited[k] = true;
for each vertex q ∈ adjacent − neighbor − V[k] do
 |  if Visited[q] == false then
 |  |  DFS(V, q);
 |  end
end
```

4. Random Graph Model

Suppose $\mathcal{T}_{n,N}$ has n possible vertices and N edges, then $\left(\begin{array}{c} \binom{n}{2} \\ N \end{array} \right)$ possible graphs can be formed. We define the following variable N_c to simplify the computation.

$$N_c = \frac{1}{2} n \log n + cn$$

We define a graph is type A if it has k isolated vertices and $n-k$ effective vertices. Everything that is not type A is type $\bar{A}$.

Theorem 4.1. *Let $P_0(n, N_c)$ denote the probability of $\mathcal{T}_{n,N}$ being connected. Then*

$$\lim_{n \to +\infty} P_0\left(n, N_c\right) = e^{-e^{-2c}}$$

Proof. Let $N(n, N_c)$ denote the number of connected graph, which is equal to the number of graphs of type A without isolated points.

$$N(n, N_c) = \sum_{k=0}^{n} (-1)^k \binom{n}{k} \left(\begin{array}{c} \binom{n-k}{2} \\ N_c \end{array} \right)$$

Thus we obtain

$$\lim_{n\to\infty} \frac{N(n, N_c)}{\left(\begin{array}{c} \binom{n}{2} \\ N_c \end{array}\right)} = \sum_{k=0}^{\infty} \frac{(-1)^k e^{-2kc}}{k!} = e^{-e^{-2c}}$$

□

Theorem 4.2. *Define $P_k(n, N_c)$ as the probability that the greatest connected component of $\mathcal{T}_{n,N}$ which contains exactly $n-k$ effective points. Then we have*

$$\lim_{n\to\infty} P_k(n, N_c) = \frac{\left(e^{-2c}\right)^k e^{-e^{-2c}}}{k!}$$

Proof. Suppose the graph has k isolated points and $n-k$ effective points. It follows that

$$P_k(n, N_c) \sim \binom{n}{k} \frac{\binom{n-k}{2}}{\left(\begin{array}{c} \binom{n}{2} \\ N_c \end{array}\right)} P_0(n-k, N_c).$$

□

For a fixed value k, we have

$$\lim_{n\to\infty} \frac{N_c - \frac{1}{2}(n-k)\log(n-k)}{n-k} = c$$

Combined with Theorem 1, the theorem is proved.

Theorem 4.3. *Let $\Pi_k(n, N_c)$ denote the probability of $\mathcal{T}_{n,N_c}$ consisting exactly $k+1$ connected components. Then*

$$\lim_{n\to+\infty} \Pi_k(n, N_c) = \frac{\left(e^{-2c}\right)^k e^{-e^{2c}}}{k!}$$

Proof. The proof of the theorem is out of scope of this paper. Detailed proof of Theorem 4.3 can be found in [1]. □

5. Small World Phenomenon

5.0.1. *Six Degrees of Separation.*

Before we dive into the generalized network model, I want to provide a little background on the origin of small world phenomenon, also known as "six degrees of separation". The idea is the distance between you and another person on the planet, on average, is exactly six people away. Initially, this conclusion seems unrealistic, but we can analyze the argument in this way: suppose you have 100 connections, and each person you know also has 100 connections. After two steps, there are $100 \times 100 = 10,000$ people. After five steps, there are 10 billion connections in total. Mathematically, there is nothing wrong with our analysis. However, the conclusion can barely help us to understand what a real social network is like. This is why we need to introduce the concept of analyzing decentralized search.

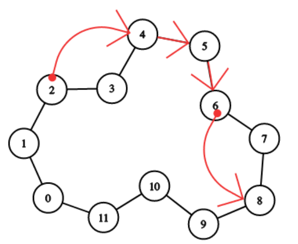

FIGURE 7. In myopic search, the message carrier will choose the contact that is closest to the target

5.0.2. *Analysis of Decentralized Search - Myopic Search.*
In Figure 7, we have a series of nodes arranged into an irregular ring shape. The ring indicates that for a particular node, it is connected to its closest two neighbors. For example, 3 is connected to both 2 and 4. The red arrow implies the path if person 2 wants to find person 8.

The principle of *myopic search* is: when a specific node v has an information, it passes to the contact that lies as close to the target node as possible. For example, if we choose 2 as the initial node and 8 as the target node.

(1) Node 2 first send message to node 4 since compared to node 3, node 4 lies closer to node 8.
(2) Node 4 passes to node 5 since there is no other way.
(3) Node 6 finally reaches node 8 which is our target node.

It is worth mentioning that myopic search is not the shortest path. Since we lack the information of the overall structure of the diagram, it is possible to find other more convenient paths. For example, if 2 is connected to 7, then the path will be 2-7-8 instead of 2-4-5-6-8.

Suppose the number of steps by myopic search is X. Define the expected value of X is $E[X]$. When a message starts from the initial node and head to the target node, we say the node is in *phase* j if its distance from the target node is between 2^j and 2^{j+1}. Notice the maximum number of phase is $log_2 n$. Therefore, we have:

$$X_{fullsearch} = X_1 + X_2 + \cdots + X_{\log n}$$

$$E[X] = E\left[X_1 + X_2 + \cdots + X_{\log n}\right] = E\left[X_1\right] + E\left[X_2\right] + \cdots + E\left[X_{\log n}\right]$$

5.0.3. *Find the Normalizing Constant.*
Let u be the initial node and v be the target node. We say the probability of u link to v is proportional to $d(u,v)^{-1}$. The proof process is based on empirical experiment and we will not discuss the detail in our paper. Like any other probability problem, there is a missing constant $1/Z$ in our equation.

To find the value of Z we can think in this way: if $n = 6$ (6 nodes in total),

$$Z \leq 2\left(1+\frac{1}{2}+\frac{1}{3}\right)$$

Therefore, assuming n is even,

$$Z \leq 2\left(1+\frac{1}{2}+\frac{1}{3}+\frac{1}{4}+\cdots+\frac{1}{n/2}\right) \leq 2\left(1+\int_1^k \frac{1}{x}dx\right) = 2\left(1+\ln k\right)$$

We know since $\ln x \leq \log_2 x$,

$$Z \leq 2\log_2 n$$

Therefore, the probability equals to:

$$\frac{1}{Z}d(v,w)^{-1} \geq \frac{1}{2\log n}d(v,w)^{-1}$$

6. Application

6.1. Analyzing Degree Distribution of Random Graph Model.

We already introduced the definition of degree distribution in Definition 3.1. Now we can use it to analyze the degree distribution of a random graph. Let $P(k)$ represents the fraction of nodes with degree k. Then

$$P(k) = \binom{n-1}{k} p^k (1-p)^{n-1-k}$$

Notice the degree distribution is binomial. We can come up with the mean and variance of a binomial distribution easily as follows.

$$\bar{k} = p(n-1)$$
$$\sigma^2 = p(1-p)(n-1)$$

6.2. Analyzing Clustering Coefficient of Random Graph Model.

For a specific random graph

$$E[e_i] = p\frac{k_i\left(k_i-1\right)}{2}$$

Plug in e_i into the equation in Definition 3.5, we have

$$E[C_i] = \frac{p \cdot k_i\left(k_i-1\right)}{k_i\left(k_i-1\right)} = p$$

6.3. Application of Graph Theory in Computer Science.

Since graph theory can perfectly visualize the concept of "relationship" in a complex network, computer scientists have integrated graph structure in a variety of fields such as graph database management, data mining, and graph machine learning. A famous product is the initial Google PageRank algorithm which treats web pages as nodes and web links as edges. By computing the credibility of website, Google assigns weight to each edges. Clearly, the application of graph theory have drastically increased the accuracy of early day search engine.

7. Acknowledgements

It is my great pleasure to thank my advisor Sehyun Ji for all the help during the REU program. Sehyun is extremely helpful in helping me understand the weekly challenge problems and graph theory. I also want to thank Peter May for organizing the REU program.

References

[1] P. Erdos, A. Renyi, On Random Graphs I, Publ. Math. Debrecen, 1959.
[2] David Easley, Jon Kleinberg, Networks, Crowds, and Markets: Reasoning about a Highly Connected World, Cambridge University Press, 2010.
[3] Robin J. Wilson, Introduction to Graph Theory, Prentice Hall, 1996.
[4] https://www.baeldung.com/cs/graph-connected-components
[5] snap.stanford.edu/class/cs224w-2019/