STATISTICAL ANALYSIS OF INFECTIOUS DISEASE DATA ON NETWORKS

By

Xuan Li

BS, Bioengineering, Huazhong Agricultural University, China, 2011

Submitted to the Graduate Faculty of

Biostatistics

Graduate School of Public Health in partial fulfillment

of the requirements for the degree of

Master of Science

University of Pittsburgh

2015

UNIVERSITY OF PITTSBURGH

GRADUATE SCHOOL OF PUBLIC HEALTH

This thesis was presented

by

Xuan Li

It was defended on

June 29, 2015

and approved by

Committee Chair
Gary Marsh, PhD, Professor, Department of Biostatistics, Graduate School of Public Health, University of Pittsburgh

Committee Member
John J. Grefenstette, PhD, Professor, Health Policy and Management, Graduate School of Public Health, University of Pittsburgh

Committee Member
Hasan Guclu, PhD, Assistant Professor, Department of Health Policy and Management, Graduate School of Public Health, University of Pittsburgh

Committee Member
Supriya Kumar, PhD, Visiting Research Assistant Professor, Department of Behavioral and Community Health Sciences, Graduate School of Public Health, University of Pittsburgh

Gary Marsh, PhD

STATISTICAL ANALYSIS OF INFECTIOUS DISEASE DATA ON NETWORKS

Xuan Li, MS

University of Pittsburgh, 2015

ABSTRACT

Purpose

Infectious disease modeling has a long history in helping researchers to understand the complex spread pattern of infectious disease. Social contact networks and agent-based models can be used to conceptualize social contact pattern and spread process of infectious disease. The goal of this research is to investigate the relationship between network measurements and individual infection risk using statistical analysis.

Public Health significance

This research will help in gaining a better understanding of the important factors of infection risk in a population. Identification of central people may be used to inform building an efficient surveillance and prevention program.

Methods

Three social contact network models were used in this thesis, Erdos-Renyi network, Barabasi-Albert network and Jefferson County contact network using FRED platform. We simulated mild and severe epidemic outbreaks on them and calculated infection risk and infection speed of each individual. Network measurements, degree, betweenness centrality, closeness centrality, eigenvector centrality, PageRank, and clustering coefficient were measured on the ability to

identify groups of different infection risk level and infection speed. Random Forest and variable importance were used to estimate the most important factors in predicting infection risk

Results

For Barabasi-Albert and Erdos-Renyi networks, centrality measurements are critical factors in identifying infection risk. Degree is the most important factor in Barabasi-Albert network while closeness and degree are the most important in the mild outbreak and severe outbreak respectively in the Erdos-Renyi network. Results of Jefferson County contact network in FRED find out the importance of location sizes. The highly clustered structure of location-based model makes betweenness centrality and clustering coefficient important in predicting infection risk.

Conclusion

Different network structures and characteristics of the disease will influence the importance of network measurements. Network structures also influence the correlations between network measurements. Random forest is a powerful tool for classifying infection risk. Centrality network measurements may help in identifying high infection risk people.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

PREFACE

A special thanks goes to Dr. Marsh, the committee chair for supervising on my thesis, and also my academic advisor for the reviewing of my thesis. He provided enormous encouragement and support for my master study. I am very grateful to have him as my advisor and enjoy every talk with him.

I wish to express my sincere thanks to Dr. Guclu for his excellent guidance on my thesis. I was fortunate to have the opportunity to work with him. I greatly benefited from his insights.

I am grateful to Dr. Grefenstette for his help and participation in my thesis committee. Thanks for introducing FRED to me and giving me great support.

I also wish to take this opportunity to thank Dr.Kumar. She gave a lot of valuable suggestions and really helped me in improving my results.

1.0 INTRODUCTION

Infectious disease is one of the leading factors of illness and death worldwide. Even after many notable successes in prevention and control, it continues to be a threat to public health. In recent years, the increasing movements of people facilitate the spread of infectious disease. Moreover, the adaptation and evolution of agents lead to the emergence of new infectious disease and the reemergence of some existing infectious disease [1]. Several epidemics surprised the global community, including HIV, SARS, H1N1 pandemic influenza, and Ebola. The complex transmission and spread route among the human population challenge the efforts to prevent and control infectious disease. One of the major control methods is through the development of strong surveillance systems, which need a clear understanding of the transmission pattern of infectious disease on human population [2].

Faced with the complexity of these infectious diseases, mathematical and computational models offer valuable tools for understanding the transmission of infections and evaluating the potential impact and to control. In recent years, infectious disease modeling studies were conducted to characterize the population and to estimate individuals' risk of infection using network analysis. Studies in Human Immunodeficiency Virus (HIV) found that the virus transmits through a contact network of sexual contact and intravenous drug use [3]. Complex heterogeneous contact networks are of great importance in understanding the transmission pattern in these studies.

They can be conceptualized by using simple social contact network models. Epidemic outbreaks can be described using agent-based models.

Finding the target group with high infection risks in an outbreak will help to improve the efficiency of targeted surveillance and prevention programs for efficiency. Social contact network measurements provide a way to determine the role of an individual within one population in one outbreak. The concept of "centrality" in network analysis describes the relative importance of one individual in one population according to some criteria. In a study of Syphilis in human population, centrality measurements were used to identify the important individuals in infection spread process. Studies also suggested that many network parameters have linear relationships with infection risk [4].

Several studies have been done on exploring the factors of individual infection risk. Christley [5] used random and small-world network models to explore the relationship between network measurements and infection risk. Their results showed that centrality measurements including degree, betweenness centrality and closeness centrality were associated with infection risk. However, the limitations of this paper are that it generated networks with only 100 nodes and conducted limited statistical analyses.

Some other studies are focused on exploring the relationships among network measurements. Chang-Yong Lee did research on some complex networks in social science and found out that degree and betweenness centrality are highly correlated [6]. Valente studied 58 different real network datasets and found out that the centrality measurements are correlated with different Pearson correlation coefficients [7].

In this thesis, our first goal is to investigate the relationship between network measurements and infection risk of each individual in theoretical social contact network models. Our aim is

finding the important network measurements that could help in classifying the high-risk group. Erdos-Renyi network, Barabasi-Albert network and Jefferson County contact network were used to represent three kinds of social contact patterns. Our second goal is to further research the vital factors in predicting infection risk of each individual. Besides the network measurements, other geographic characteristics are also included as predictors. For this part of the thesis, we used a large-scale epidemic simulation system FRED that was developed at Pitt Public Health.

2.0 METHOD

2.1 SOCIAL CONTACT NETWORK

A social contact network consists of nodes and edges, where nodes represent individuals that we want to study and edges represent the contacts between individuals. For simplicity, we assumed that the edges of networks are undirected (the contacts are mutual) and have no weights (all are equivalent). We generated Erdos-Renyi and Barabasi Albert networks using the iGraph package in Python [10].

2.1.1 Erdos-Renyi network

Erdos-Renyi network (ER network) [8] network is also referred as the random network. It is constituted by N node connected by M edges, which are chosen randomly from all possible edges. The probability that two nodes are connected is independent of nodes' degree. The degree distribution of ER network follows Poisson distribution. We define the expected value of degree as the average degree K, thus M= KN/2, see Fig 1.

ER network is considered as a benchmark network for its pure random structure. Complex networks with complicated topology and principles often appear random. It is a simple but helpful tool in studying complex networks.

2.1.2 Barabasi–Albert network

Barabasi–Albert network (BA network) [9] is a kind of scale-free network in which the degree distribution has no scale. Empirical studies showed that many large complex networks are scale-free. Barabasi and Albert argued that the scale-free nature has two mechanisms which are common in many real complex networks [9]. First, small clusters of nodes are firstly formed and then the network expands to some size. Second, the likelihood of connecting incoming nodes to nodes is done with the probability proportional to their degrees, which represent a phenomenon known as preferential attachment.

This attachment mechanism creates a power-law degree distribution. Thus, BA network is also considered as a type of "power-law network". BA networks have some high-degree nodes that are absent in ER networks in which the degree distribution is centered around the average value (see Fig. 1). Rather than focus on topology like a random network, Barabasi-Albert network emphasizes on capturing the network dynamics.

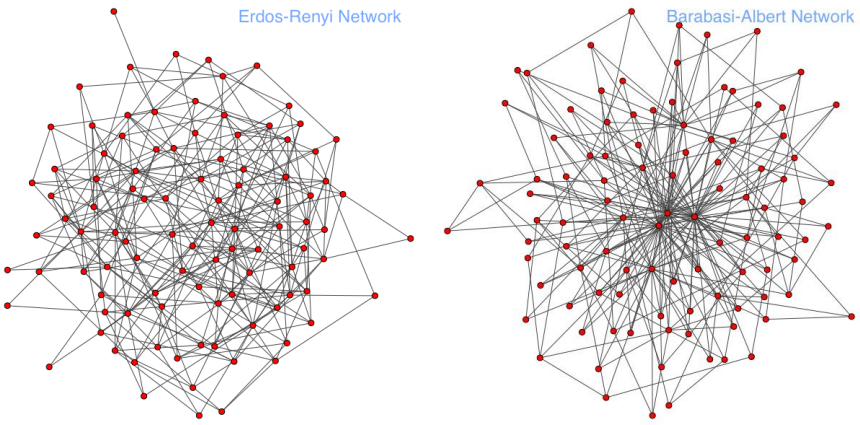

Figure 1: The visualizations of ER network and BA network. Both networks have 100 nodes and average degree K=6.

2.2 SOCIAL CONTACT NETWORK MEASUREMENTS

Network measurements were calculated using the iGraph package in Python[10].

2.2.1 Centrality measurements

The concept of centrality represents the relative importance of a node inside a network. There are many different definitions of centrality to describe the different structural importance of nodes. Degree, for example, pays more attention to "local" patterns while betweenness centrality, closeness centrality, eigenvector centrality, and page rank are more focused on "global" or "overall" structure of the network. Below are the centrality measurements that were studied in this thesis.

2.2.2 Degree

Degree is short for 'the degree of connectedness'. The degree of a node i is equal to the directly connected nodes' number. Degree distribution is defined as the distribution of probability that a randomly chosen node has a degree k.

A node with high degree is an important node that in a large number of interactions. In the Erdos-Renyi network, degree follows Poisson distribution, while, degree distribution follows the power law in the Barabasi-Albert network. Christley's study suggested that the degree is associated with infection risk [5].

2.2.3 Closeness centrality

Closeness centrality is based on geodesic distance. It is defined as the inverse of the sum of the shortest path lengths from node i to all other nodes in the network [11]. The closeness centrality of node i is:

$$C(i) = \frac{1}{\sum_{j=1}^{N} d(i,j)}$$

Where $d(i,j)$ =the shortest path length between node i and node j.

This measurement regards a node as central if this node has a shorter distance to all other nodes in networks. A node with higher closeness centrality value can communicate more quickly with the other nodes in the network.

To make closeness centrality comparable in models of different node sizes, closeness centrality is normalized by the number of connected nodes excluding the vertex itself.

2.2.4 Betweenness centrality

The betweenness centrality [12] of a node i is defined as the proportion of the shortest paths passing through it. Betweenness centrality describes the importance of nodes as the proportion of paths between other nodes in the network. It measures the information flow between different nodes based on the assumption that information flow follows the shortest path route. Christley's study suggested betweenness centrality is also an important factor for infection risk [5].

To make betweenness centrality comparable in models with different node sizes, betweenness centrality values are normalized by (N-1) *(N-2)/2 (N is the total number of nodes).

(N-1) *(N-2)/2 is the number of pairs of vertices excluding the vertex itself when every node is connected with each other.

2.2.5 Eigenvector Centrality

Eigenvector centrality is also known as Bonacich centrality. Its assumption is based on the concept that central nodes have more connections to high-scoring nodes. It is calculated based on eigenvectors of adjacency matrices. A is adjacency matrix with a unique all positive eigenvector v satisfying Av = λv. v will be the eigenvector corresponding to the largest eigenvalue λ. The eigenvector centrality value for the i^{th} node will be the i^{th} element of v, satisfying the recursive equation. j is a set of the neighbors of node i.

$$C_E(i) = \frac{1}{\lambda}\sum_j A_{ij} C_E(j)$$

2.2.6 PageRank

PageRank [14] is a variant of eigenvector centrality. Compared to eigenvector centrality, it adds a scaling factor and an attenuation factor. The formula for PageRank is:

$$C_E(i) = \alpha \sum_j \frac{A_{ji} C_E(j)}{L(j)} + \frac{1-\alpha}{N}$$

α is an attenuation factor with a range from 0 to 1. L(j) is a scaling factor which equal the number of neighbors of node j. It was developed by Larry Page and Sergey Brin in Stanford University and used by Google as an algorithm to rank the websites. The basic idea of PageRank is quite

similar to eigenvector centrality that "more important nodes are likely to receive more connection from other nodes". It is calculated by summing up the neighbor nodes' number and quality of connections. It is widely used as a famous rank algorithm in information technology.

2.2.7 Clustering coefficient

We also consider some other popular network measurements in our study.

Clustering coefficient [15] is defined as the ratio between the number of loops of length three and the number of connected triples. In other words, it shows how clustered the nodes are. Clustering coefficient is a purely local network measurement but useful in quantifying the local "strength" of connectivity.

2.3 AGENT-BASED SIMULATION OF AN SIR EPIDEMIC ON A NETWORK

In order to simulate an epidemic outbreak such as influenza, we need to model the natural history of the infection. For one epidemic outbreak, the basic assumption is that the population of interest is divided into several compartments based on their infection status (e.g. susceptible S, infectious I, or recovered R) and ignore any other demographic process (e.g birth, death, migrations) for simplicity. In the simplest SIR model, susceptible individuals become infectious immediately upon infection, thus start infecting others and recover after some time period [16]. In this thesis, we assume that the infectious period is 4 days for each individual. Once the individual recovers, he/she is immune to the disease. The process can be described as below:

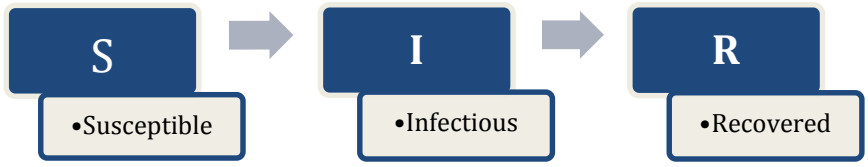

Figure 2: The SIR model of infectious disease

In a differential equation model of SIR, β is the rate of infection per effective contact (transmission or infection probability in our agent-based simulations) and γ is the recovery rate, which determines the average infectious period (for an infectious period of four days γ=0.25). There is a threshold that whether an infectious disease will be an outbreak or not. If the rate of new infections appears is greater than the rate of infected individuals recovers, then there is likely to have an outbreak. In order to describe this characteristic, a number called "reproduction number" (R_0= β/γ) is introduced, i.e., if R_0>1 an outbreak almost surely happens, otherwise the disease will die out before causing an outbreak. The reproduction number is also considered as a measure of severity of the epidemic.

In agent-based simulations of SIR model on networks, the reproduction number cannot be calculated exactly but an approximate formula is used, i.e., R_0=βKδ, where β is the transmission probability, K is the average degree and δ is the length of the infectious period.

Another important measure for the severity of the epidemic is the proportion of infected people after a disease outbreak, also known as "attack rate". In one outbreak, the total number of people that got infected divided by the total number of people in the network is the attack rate. In our study, we simulated two levels of epidemic outbreaks according to the attack rate.

2.4 INFECTION RISK AND SPEED

Infection risk is defined as the probability of infection for an individual in an epidemic outbreak. In order to calculate the infection risk, we simulated an infectious disease outbreak on a social network numerous times up to ten thousand starting in each of them with a random index case. Then we calculated the risk of infection of an individual as the ratio of the number of runs that his/her got infected to the total number of runs.

Another measurement which is related to the risk of infection is the speed of infection for the individual measured as the average of the inverse of the numbers of steps passed for infection to reach that individual since the beginning of the outbreak with an index case.

2.5 FRED

FRED (A Framework for Reconstructing Epidemic Dynamics) is an open source platform developed by the University of Pittsburgh's National Center of Excellence for the Models of Infectious Disease Agent Study (MIDAS)[17]. It is an agent-based simulation model with the purpose of facilitating infectious disease research.

The populations of FRED use realistic data and are based on the US Census Bureau’s Public Use Microdata files (PUMs) and Census aggregated data. Some socio-demographic characteristics and daily behaviors are recorded for each agent in FRED. For a specific geographic location, house, school, neighborhood, and workplace are also reflected with the actual distance.

Each individual in FRED also has health information recorded (e.g., current health status, date of infection, and susceptibility) [17].

In FRED, the neighborhood is defined by dividing the whole area into a 1km*1km square. School data is real survey data while classroom size is artificially chosen. First we separate all students to separate age groups and each age group is divided into classroom groups of up to 40 students. Students interact with the students assigned to the same classroom and the same school with different probabilities. Same with workplace and offices, offices are defined by dividing up all the workers in given workplace groups of up to 50 workers.

FRED is an agent-based model that can be used to simulate the daily activities' interactions of millions of target agents in a specific geographical region during an epidemic and measure the effects of intervention strategies for the infectious disease. Within one day, each agent interacts only with the other agents that share the same locations and have a possibility of transmission of disease. For each simulated epidemic in FRED, the spread of disease is tracked during a period of time, usually several months or years.

The transmission model in FRED is an agent-based model. Two sets of numeric parameters determine the spread of infection: the number of contacts per infectious person per day, and the probability that a contact transmits an infection. Each place has different values of parameters.

The transmission probability for a given place is determined by the agent's age and health status. For the contacts number of infectious person per day, it has the following formula:

Number of contacts(i) = Trans(D) * CR(P) * Inf(i) * S(P)/N(P)

where: Trans(D) = the transmissibility factor for disease D,

CR(P) = the contact rate for place P, the number of potentially infective daily contacts

Inf(i) = the infectivity of agent i,

S(P) = the number of susceptible agents visiting place P,

N(P) = the number of total agents who usually visit place P

The Pitt MIDAS group has used FRED to evaluate responses to influenza pandemics, including vaccination policies [18], school closure [19], and the recent publication discussing vaccine coverage [20].

In this paper, the contact network of Jefferson County is constructed based on location. Only household, classroom and office are considered as contacts in our network in order to to control the density of the network. For example, if one agent has a probability of staying at home and school, this agent has contact with all the agents that located in the same house and the same classroom (not same school).

We conducted epidemic simulations in FRED using Jefferson County data in Pennsylvania. Demographic characteristic (age, gender and race), the number of people in places (households, neighborhoods, school, classrooms, workplaces, and offices) and network measurements are considered in this thesis

2.6 STATISTICAL ANALYSIS

2.6.1 Spearman's rank correlation coefficient

Spearman's rank correlation coefficient is a non-parametric measure of dependency between two variables. It is defined as Pearson correlation coefficient by rank. It can assess whether there is a relationship between two variables using monotone functions.

To calculate Spearman's rank correlation coefficient between the variable of x and y, we first rank x and y by ascending order and calculate the differences d between the two ranks.

$$\rho = 1 - \frac{6 \sum d_i^2}{n(n^2 - 1)}$$

The sign of Spearman's rank means the direction of their association between two variables. It has a range of -1 to 1. Compared to Pearson correlation coefficient, Spearman correlation coefficient can better assess whether the two variables are monotonically related especially when their relationship is not linear.

Spearman's rank correlation coefficient was calculated using the "psych" package of version 1.5.1 in R

2.6.2 Classification

The whole population is split into three groups (low risk, medium risk and high risk) based on the value of infection risk. We split the group based on Quantile. Our goal is to classify three groups with network measurements using Random Forest[22h].

Random forest evolves from the classification tree and attempts to solve the classification tree's problems of high variance and high bias and find a natural balance between the two extremes. It has been proven to outperform the bagging classification tree and boosting classification tree [22]. To better understand Random forest, we first start from how to grow a classification tree.

2.6.3 Classification tree

Classification tree is a condensed classification statement of the scheme, and can analyze those models which contain a categorical dependent variable. It starts with a root node, and then finds the most informatics binary split. Taking each of the resulting new nodes and repeating the process, the recursion will end until there is only one observation at each node. This classification tree is constructed using the Gini measure of impurity [23]. For node τ, Gini index is used as the following equation,

$$i(\tau) \coloneqq \sum_{j \neq k} p_j p_k = 1 - \sum_{j=1}^{k} p_j^2$$

For each variable j and at each node τ, any split s would find a partition for

$$\tau = \tau_L \cup \tau_R$$

The goodness-of-split is measured by the reduction in impurity caused by the split s on the j^{th} variable, from node τ to two separated τ_L andτ_R:

$$\Delta i(\tau, j, s) \coloneqq i(\tau) - [p_L i(\tau_L) + p_R i(\tau_R)]$$

Since i(τ) is independent of the split that is yet to happen, to maximize $\Delta i(\tau, j, s)$ is to minimize $p_L i(\tau_L) + p_R i(\tau_R)$.

2.6.4 Random forests

Random forest [22] grow many classification trees and try to reduce bias by bootstrap aggregation and randomly selecting variables in individual tree level. To classify a new observation, we put the input data down the forest model. Each tree in the forest has a result group, and the final group is the one receiving the most votes

2.6.5 Out of bag (OOB)

To get tree model, we first need to bootstrap original data and build models based on each bootstrap data. The observations which are left out of the bootstrap sample will not be used in the tree model building. They are called out of bag observations. They will be used as a test set to be put on the tree model and get prediction of class. The proportion of whose predicted class is not equal to the true class of all the out of bag observations is the OOB error estimate. In random forests, there is no need for external cross-validation because the out of bag error in Random Forest has proven to be unbiased in many tests [22].

2.6.6 Variable Importance

To get the variable important scores of variable V, we first randomly permuted the values of variable V and put all the OOB observations to the forest model. Then we subtracted the OOB error estimate in the variable V permuted OOB data from the OOB error estimate in the original OOB data. We averaged all trees in the forest and get the variable importance value of variable V. According to several studies of variable importance, the calculation of variable importance by

permutation performs better than other tests such as Gini importance. It has lower bias even with multi-valued attributes [24] [25].

2.6.7 Random Forest Algorithm

The algorithm of random forest with n trees and m input variable is:

Step1: Bootstrap the original dataset and generate n bootstrap sample.

Step2: For each tree, randomly select m input variables from M total variables and use the bootstrap sample as training data to fit a tree model without prune.

Step3: For each tree, put down the left out cases and count the number of trees with the wrong class. Calculate the misclassification rate as out of bag error rate.

Step4: For each tree, permute one variable value each time and keep other variables fixed, refit the tree and calculate the out of bag errors. Deducting from the original OOB error, the increased OOB error is the variable importance.

Step5: The final classification results are class votes by most trees. Aggregate out of bag error and importance measures from all tree to calculate the overall out of bag error rate and variable importance.

According to Breiman's paper [22], two characteristics will affect the overall performance of Random Forest. One is strength, which measures the prediction power of the model, and the other is the correlations between each tree. High strength and lower correlation will improve the performance and minimize the generalization error. Using bootstrap aggregation (bagging) is able to enhance accuracy and improve unstable procedures [22]. In the randomForest package, parameter "ntree" means how many tree models will be build. More ntrees will increase the

strength without consideration of overfitting problems, but the large number of tree models demand high computational cost. There is another important parameter in random forest model which make it outperforms bagging Classification tree, it is “mtry”. Randomly selecting variable in the tree model building process will decrease the correlation but also decrease the strength. An optimized mtry will be determined by 5-fold cross-validation results with minimized OOB error rate.

All the random forest analyses were done using the randomForest package version 4.6-10 in R [26].

3.0 RESULTS

3.1 NETWORK AND ATTACK RATE

In this section we will present results for network characteristics. We worked on two different network models: ER and BA networks. We constructed ER and BA networks with 10,000 nodes and an average degree K=10. The density of two networks are 0.01 and the total number of edges are 50000.

According to the results in Table 1, with the same network density, there are differences between network measurements of BA network and ER Network. This indicates the different structure of the two networks. The BA network is highly clustered with larger average clustering coefficient (mean: 0.0055) compared to the ER network (mean: 0.0011). There are some hubs in the BA network which are located in the core and have a high degree. Hubs can act as bridges and help other nodes more easily reach each other. Thus, the BA network has higher average closeness centrality value than the ER network. Hubs in the BA network have extremely high values of almost all centrality measurements. Besides hubs, The BA network also have some nodes located in the periphery and are not easily reached because of low degree. Those nodes have relatively low centrality measurements value. Because of those special characteristics, the BA network has broader distributions of centrality measurements than the ER network.

.

Table 1: Summary of network measurements in the Barabasi-Albert and Erdos-Renyi network

Variable	Barabasi-Albert Network				Erdos-Renyi Network			
	Mean	Median	Min-Max	SD	Mean	Median	Min-Max	SD
Degree	10	7	5-273	11.6	10	10	0-25	3.17
Clustering Coefficient	0.0055	0	0-0.2	0.019	0.0011	0	0-0.33	0.006
Betweenness Centrality	0.00027	0.000068	0.00065-0.057	0.0015	0.00032	0.00029	0-0.0017	0.0002
Closeness Centrality	0.27	0.27	0.22-0.40	0.019	0.19	0.19	0.0001-0.2072	0.006
Eigenvector Centrality	0.021	0.013	0.001-1	0.034	0.3509	0.33	0-0.43	0.013
PageRank	0.0001	0.000075	0.000052-0.0023	0.000098	0.0001	0.0001	0.00001-0.00025	0.000027

To explore the relationship between network measurements, Spearman's correlation matrix is constructed. The Histogram plot and density distribution of network measurements were plotted on diagonal panels. Spearman's correlation coefficient and the scatterplots between two network measurements are in upper and lower diagonal panels respectively.

In the Barabasi-Albert network, degree, betweenness centrality, eigenvector centrality and PageRank have a similar distribution of power law. Degree and PageRank, closeness centrality and eigenvector centrality are perfectly correlated with Spearman's correlation coefficient of 0.99. Betweenness centrality is highly correlated with other four network measurements with Spearman's correlation coefficients from 0.8 to 0.88.

In an Erdos-Renyi network, degree, betweenness centrality, eigenvector centrality and PageRank have distributions similar to a Poisson. All network measurements are highly correlated with each other with Spearman correlation coefficients range from 0.9 to 1.0. Two pairs, degree and PageRank, closeness and eigenvector are perfectly correlated with coefficients equal to 1.0.

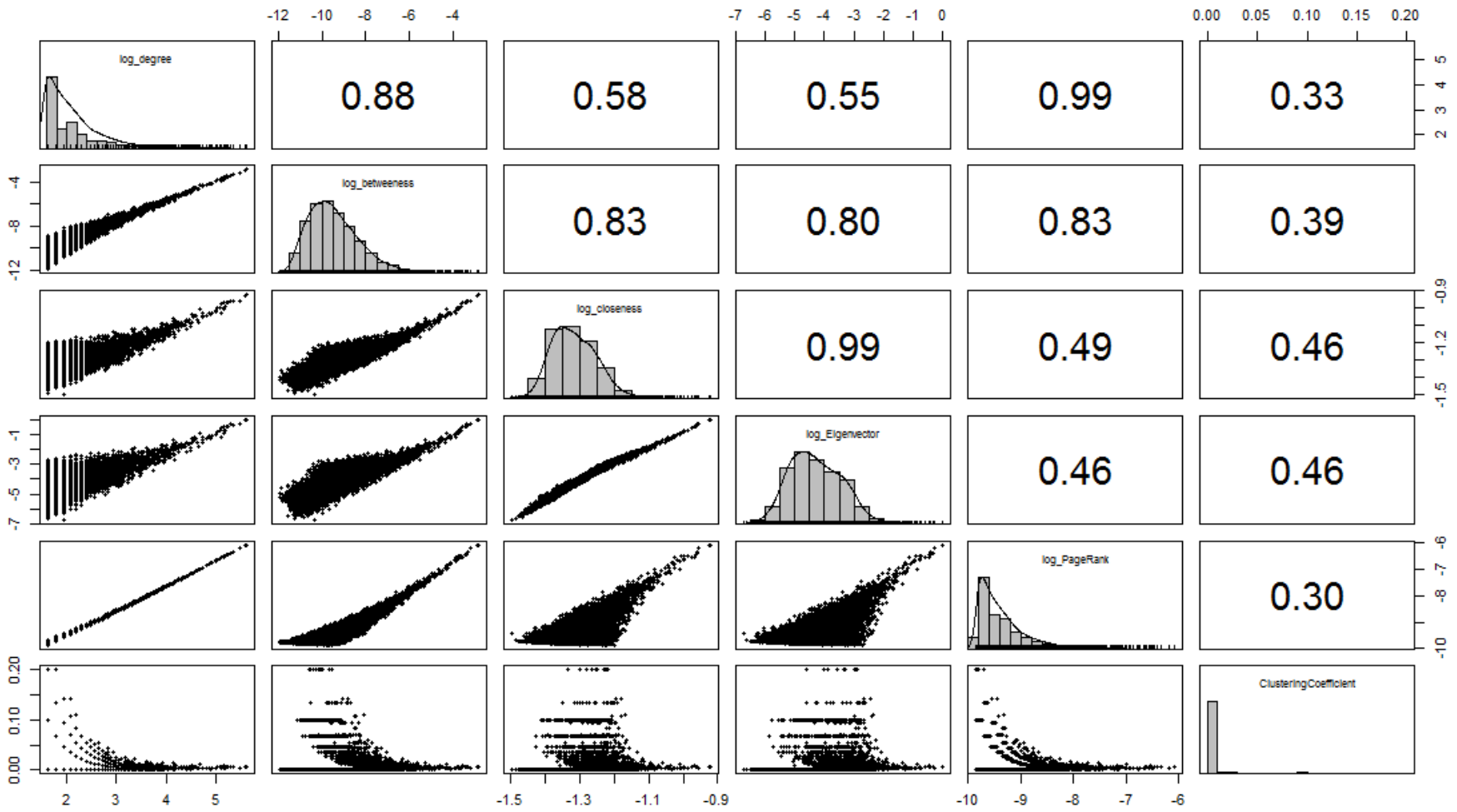

Figure 3: The Spearman's correlation matrix of the Barabasi-Albert network. Degree, betweenness, closeness, eigenvector and PageRank are log-transformed. Upper-matrix shows the Spearman's correlation coefficients and lower matrix shows the scatterplots between network measurements.

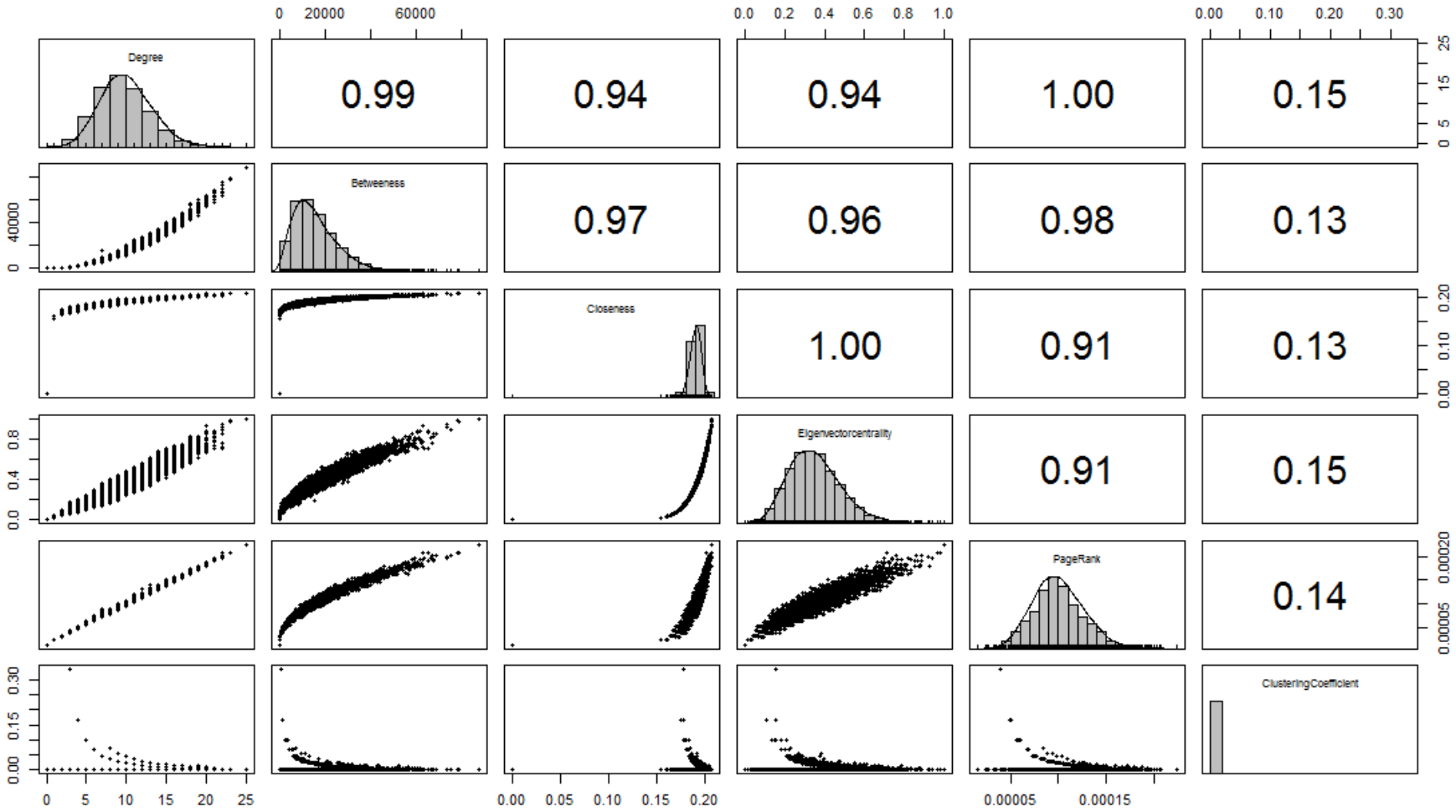

Figure 4: The Spearman's correlation matrix of the Erdos-Renyi network

Network structure influences the Spearman correlation coefficient between centrality measurements. The ER network is a random graph structure that thus all centrality measurements are closely correlated with each other. This suggests that nodes that have more connection in the ER network tend to be closer to the center of information flow, have shorter distances with other nodes and connect with more important neighbors. Degree and PageRank, closeness centrality and eigenvector centrality are highly correlated in both networks. Many studies have been done on approximating PageRank using degree by mathematic analysis and experiments [29]. The two measurements are only different in the multiplicity factor under some assumptions. In a recently published paper of José Ricardo [30], closeness centrality and eigenvector centrality were also shown to be highly correlated in the ER and BA network. This suggests that in both network model, the nodes that are closer to their neighbors are more likely to have important neighbors.

Clustering coefficient have very low Spearman correlation correlation with all other centrality measurements. However, Spearman correlation coefficients between clustering coefficient and centrality measurements are higher in the BA network than ER network. This may be explained by the BA network structure. Nodes in the core of the BA network are more clustered with high clustering coefficient. Those nodes also have higher centrality measurements.

3.2 NETWORK MEASUREMENTS

10,000 times simulation with two different infection probability β values 0.035 and 0.05 were done on the BA and ER network, corresponding to mild and severe outbreaks. The attack rate is defined as the proportion of people infected in each outbreak. We took an average of the attack rate on

10,000 times simulations and get the average attack rate. Kernel density plots showed the distribution of all the attack rate values.

Table 2: The average attack rate for the Erdos-Renyi and Barabasi-Albert network

	Erdos-Renyi		Barabasi–Albert	
	K=10 β =0.035	K=10 β =0.05	K=10 β =0.035	K=10 β =0.05
MEAN	0.20	0.56	0.25	0.48
SD	0.22	0.33	0.25	0.32

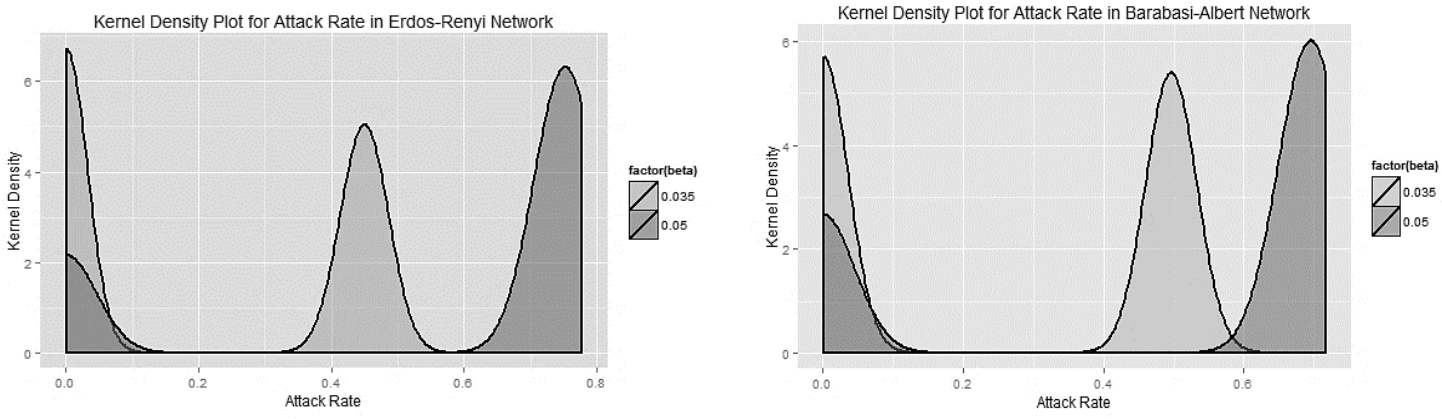

Figure 5: The kernel density plots of attack rate in the ER and BA network with β=0.035(Gray) and β=0.05(Black).

In both ER and BA network, there are two peaks in attack rate distribution. There is one small peak on the interval (0, 0.1) which shows there is no global epidemic in this network. This may because that when index cases happen in the periphery area, they have very limited connections to induce an epidemic.

3.3 DESCRIPTIVE ANALYSIS

In this section we present the results of infection risk and speed of infection and their relationship with network measurements. We worked on two different network models: ER and BA networks with an average degree K=10 and two different infection probability β values 0.035 and 0.05, corresponding to mild and severe outbreaks.

3.3.1 Barabasi-Albert Network

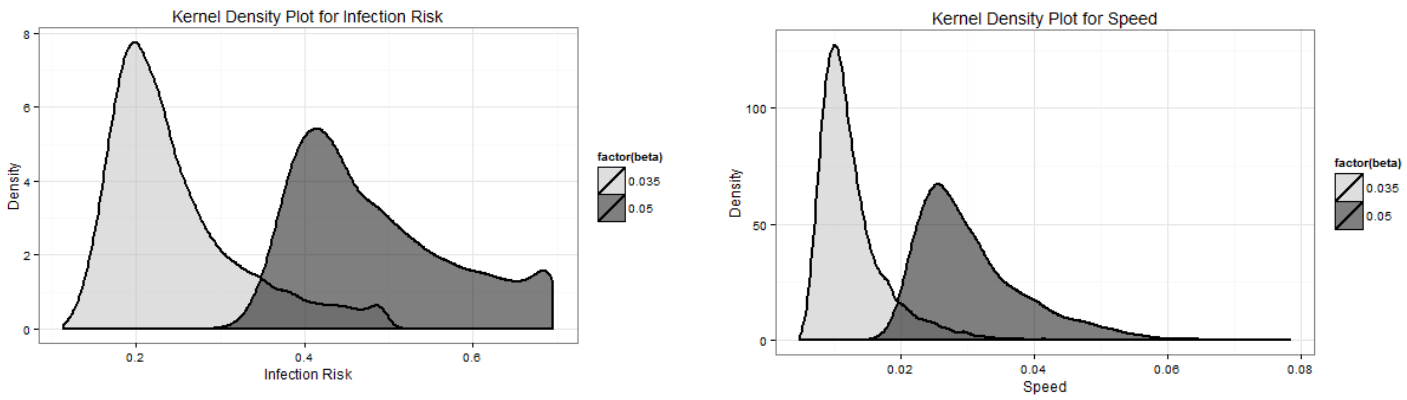

Figure 6: Kernel density plot of infection risk and speed in the BA network for β=0.035(Gray) and β=0.05(Black)

In the BA network, there are two peaks in infection risk distribution. Several nodes which have very high infection risk are the hub nodes. When β=0.035, the infection risk distribution has a sharper peak and fatter tail, while the infection risk distribution with β=0.05 has a more rounded peak and thinner tail.

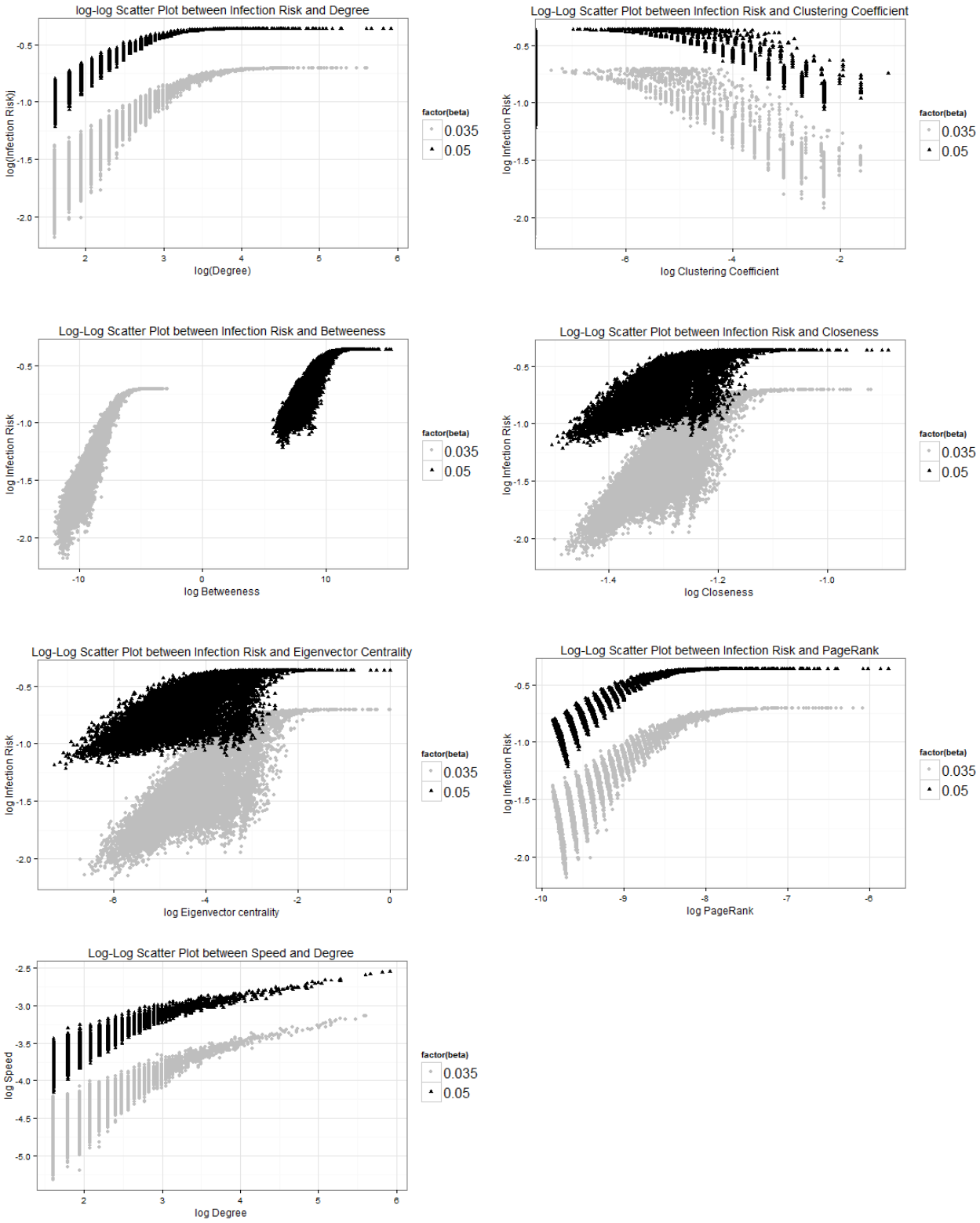

Figure 7: Log-log transformed scatterplots between network measurements and infection risk or speed in the BA network (β =0.035(Grey) β =0.05(Black)

From the log-log scatterplot of Barabasi-Albert network measurements, degree, betweenness centrality, and closeness centrality have exponential relationships with infection risk, and then the increasing rate decrease to zero. Degree and speed have an exponential relationship with infection risk with a changing point in the middle. The shapes of the curves do not change much from β=0.035 to β=0.05. When outbreaks become more severe, for the nodes with the same network measurements, the infection risk increases.

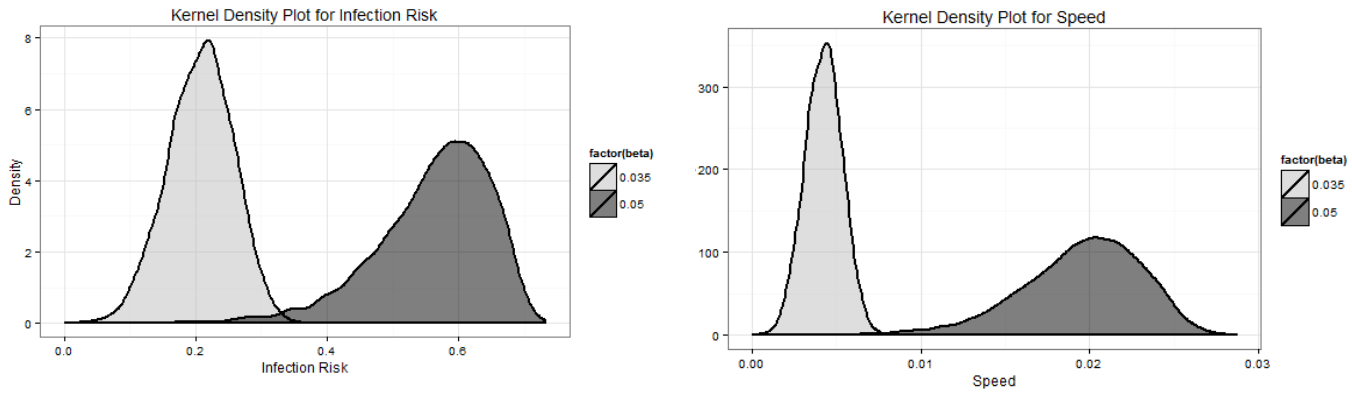

Figure 8: The kernel density plots of infection risk in the ER network for β =0.035 and β =0.05

3.3.2 Erdos-Renyi Network

In the ER network, when β=0.035, the infection risk distribution has a sharper peak and fatter tail, while the infection risk distribution under β=0.05 has a more rounded peak and thinner tail.

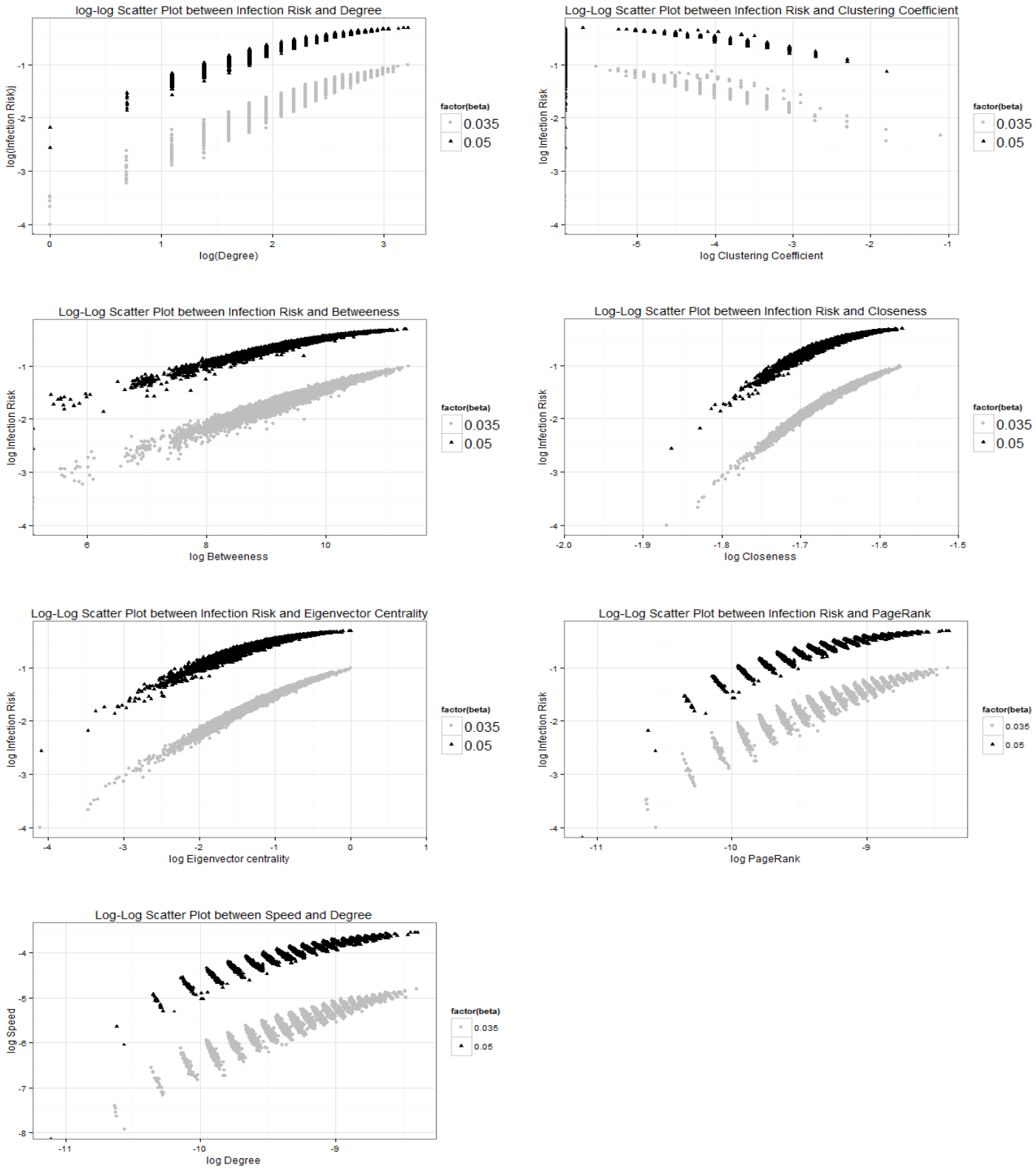

Figure 9: Log-log transformed scatterplots between network measurements and infection risk or speed in the BA network with β =0.035 and β =0.05.

In log-log transformed scatterplot of Erdos-Renyi network measurements, degree has an exponential relationship with infection risk, same between degree and speed. When the outbreaks become more severe, for the nodes with same network measurements, the infection risk increases.

3.4 CLASSIFICATION

Since the infection risk has only one peak, we sorted the whole population in ascending order by infection risk and separated into three groups with equal numbers. The first one third people were the low-risk group, the last one third people were the high-risk group, and others were the medium-risk group.

Random forest was used for the classification of infection risk. We calculated the variable importance of each variable for the ER network and the BA network with $\beta=0.035$ and $\beta=0.05$. To eliminate and the variation of variable importance estimation, we repeated the classification procedure 50 times and then took an average to get final results.

Table 3: The mean raw important scores of network measurements in the ER network with β=0.035. OOB error =1.96%

Variable	Mean Raw Important Score			
	Low-Risk Group	Medium-Risk Group	High-Risk Group	Overall
Degree	0.090	0.081	0.22	0.13
PageRank	0.02	0.08	0.09	0.06
Betweenness Centrality	0.084	0.058	0.14	0.095
Closeness Centrality	0.22	0.15	0.28	0.22
Eigenvector Centrality	0.17	0.14	0.23	0.18
Clustering Coefficient	<0.0001	<0.0001	<0.0001	<0.0001

Table 4: The mean raw important scores of network measurements in the ER network with β=0.05. OOB error =1.49%

Variable	Mean Raw Important Score			
	Low-Risk Group	Medium-Risk Group	High-Risk Group	Overall
Degree	0.26	0.23	0.37	0.28
PageRank	0.07	0.15	0.17	0.13
Betweenness Centrality	0.13	0.12	0.19	0.15
Closeness Centrality	0.12	0.11	0.18	0.13
Eigenvector Centrality	0.07	0.08	0.10	0.08
Clustering Coefficient	<0.0001	<0.0001	<0.0001	<0.0001

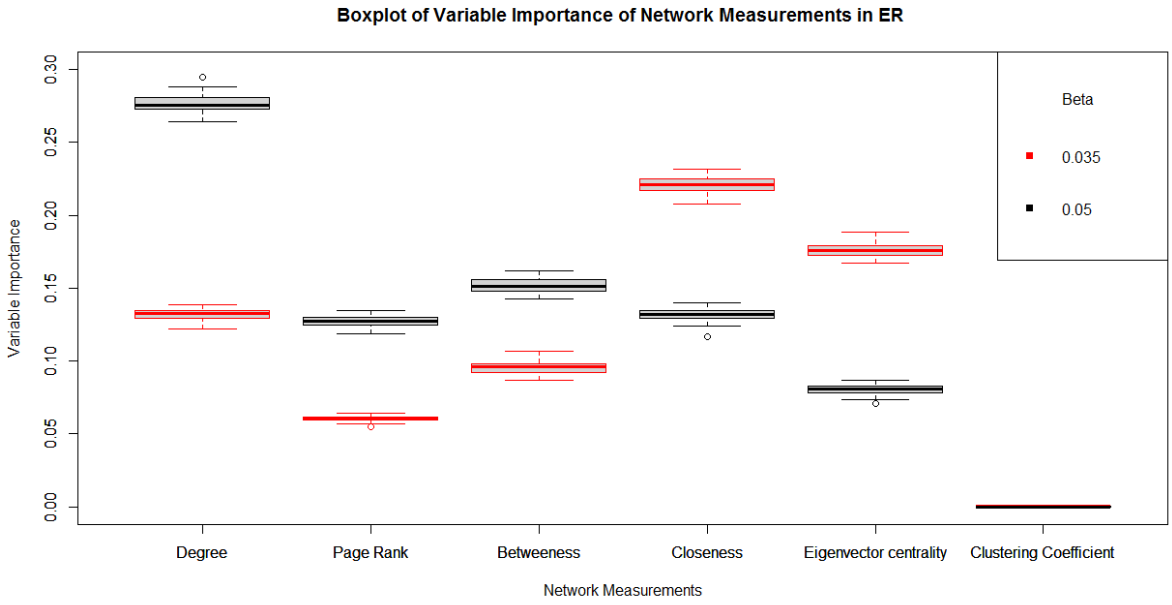

Figure 10: The boxplot of mean raw important score of network measurements in the ER network with β=0.035(Red) and β=0.05(Black)

The mean raw important score is defined as the average of the decrease in accuracy when permuting current variable and fix other variables. It shows the importance of the variable in classifying groups in the random forest model. In both mild outbreaks and severe outbreaks random forest models, all centrality measurements have a relatively high mean raw important score but different in levels. The mean raw important scores of clustering coefficient are extremely low (<0.0001).

In mild outbreaks, closeness centrality is the most important variables (See Table 3). Eigenvector centrality and degree are the second and third most important variables respectively. They play important a role, especially in classifying the high-risk group. It is interesting to find that clustering coefficient have little effect at all in classifying the groups. Scatterplots in Fig 7 shows some negative exponential relationship with infection risk. The main reason is because

clustering coefficient have too many zero values (94%) in the ER network. Compared to the mild outbreaks (See Fig.10), degree becomes the most important variables in severe outbreaks. Betweenness and PageRank become more important and ranked as the second and third important variables compared to mild outbreaks.

Higher Closeness means the node is closer to other nodes in the network. When the outbreak happens, nodes with higher closeness may be quickly reached because it has a shorter shortest path with infected nodes. Because ER is a random graph network with no hubs, infection cases will grow slowly within small areas when β=0.035. Only 20% of nodes are infected in average. Thus, those nodes that can be quickly reached will have a higher risk. However, when β=0.05, more than half of the nodes are infected in the ER network. In this case, closeness centrality becomes less important because outbreak spreads in larger areas while degree becomes more important because nodes with a higher degree will have a higher probability of contacting infected nodes.

Table 5: The mean raw important scores of network measurements in the BA network withβ=0.035. OOB error =1.99%

	Mean Raw Important Score			
Variable	Low-Risk Group	Medium-Risk Group	High-Risk Group	Overall
Degree	0.32	0.35	0.52	0.40
PageRank	0.16	0.39	0.20	0.25
Betweenness Centrality	0.053	0.082	0.14	0.087
Closeness Centrality	0.12	0.10	0.12	0.11
Eigenvector Centrality	0.036	0.051	0.043	0.045
Clustering Coefficient	0.001	0.001	0.001	0.001

Table 6: The mean raw important score of network measurements in the BA network with β=0.05. OOB error =4.24%

	Mean Raw Important Score			
Variable	Low-Risk Group	Medium-Risk Group	High-Risk Group	Overall
Degree	0.40	0.43	0.54	0.46
PageRank	0.23	0.33	0.21	0.26
Betweenness Centrality	0.040	0.046	0.09	0.064
Closeness Centrality	0.072	0.059	0.064	0.063
Eigenvector Centrality	0.032	0.041	0.015	0.03
Clustering Coefficient	0.001	0.002	0.001	0.0017

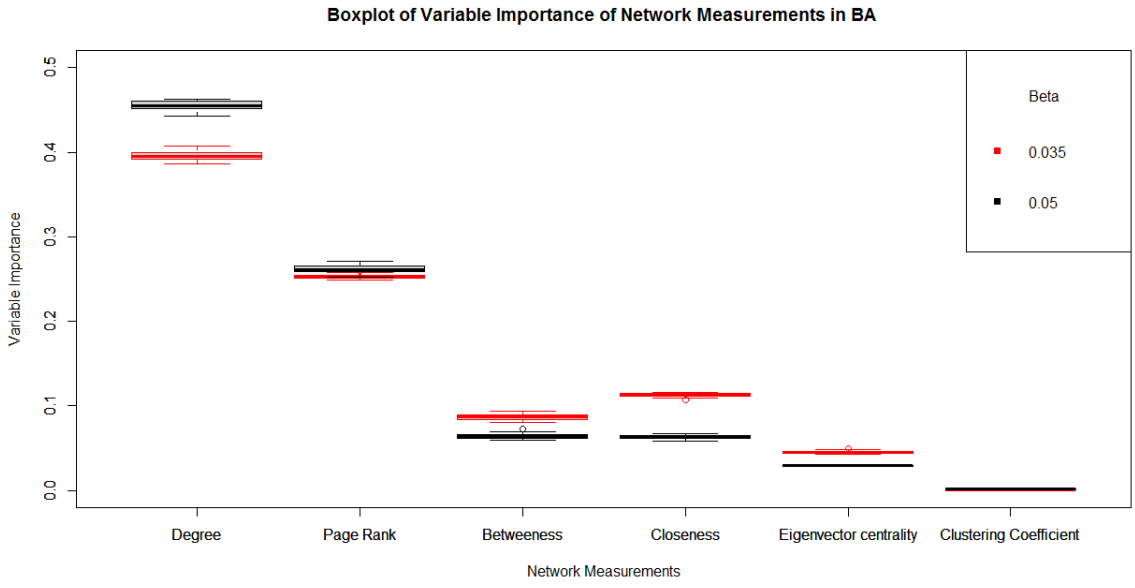

Figure 11: The boxplot of mean raw important score of network measurements in the BA network with β=0.035(Red) and β=0.05(Black)

Similar to the ER network, centrality measurements are important variables in random forest models of the BA network. Clustering coefficient has limited effect in classifying groups because 82% nodes have a clustering coefficient value of zero.

Based on the boxplot of variable importance in the BA network (see Fig.10), the rank of variable importance does not change much between the mild outbreak and the severe outbreak. The BA network has hubs that can act as bridges and facilitate the outbreak spread. Whether in mild outbreaks or severe outbreaks, the infectious disease starts from index nodes, quickly reaches hub nodes in the center and then spread to other parts of the network.

In the BA network RF models, degree is the most important variable in classification (mean raw importance score is 0.40 for β =0.035 and 0.45 for β =0.5), while PageRank is the second most important. Moreover, degree and PageRank become more important in classification when the epidemic becomes severe. Degree and PageRank are highly correlated thus both of them are most important variables and have similar trends when the epidemic becomes severe.

In the BA network (see Fig.3), degree and PageRank have broader power law distribution and are highly skewed with a thinner peak than closeness centrality, eigenvector centrality and betweenness centrality. The hubs in the BA network shorten the paths between nodes and narrow down the value range of closeness centrality, eigenvector centrality and betweenness centrality. In this case, degree and PageRank are more important in classifying different infection risk groups.

The boxplots show that variable importance measures are stable within 50 run times in both ER and BA network (see Fig.10 and Fig.11). The OOB error rate is 1.99% and 1.46% in the ER network corresponding to the mild outbreak and severe outbreak respectively, while in the BA network, the OOB error rate is 1.99% and 4.24% for mild outbreak and a severe outbreak. The error rates are very low and indicate the great prediction power of Random Forest in our datasets.

3.5 NETWORK SIZE AND STABLITIZATION

Further exploration of network size was conducted using the random forest. We generated five networks with the number of nodes range from 100 to 10,000 and used random forests to get the mean raw important scores. The stack plot of mean raw important scores in the BA network shows that how the ratios between network measurements change with an increasing number of nodes. Degree and PageRank become more important with an increasing number of nodes. The ratio between any two network measurements become stable when the number of nodes is larger than 5,000. We finally chose 10,000 nodes as an optimum network size for the purposes of removing small network effects and decreasing computation cost.

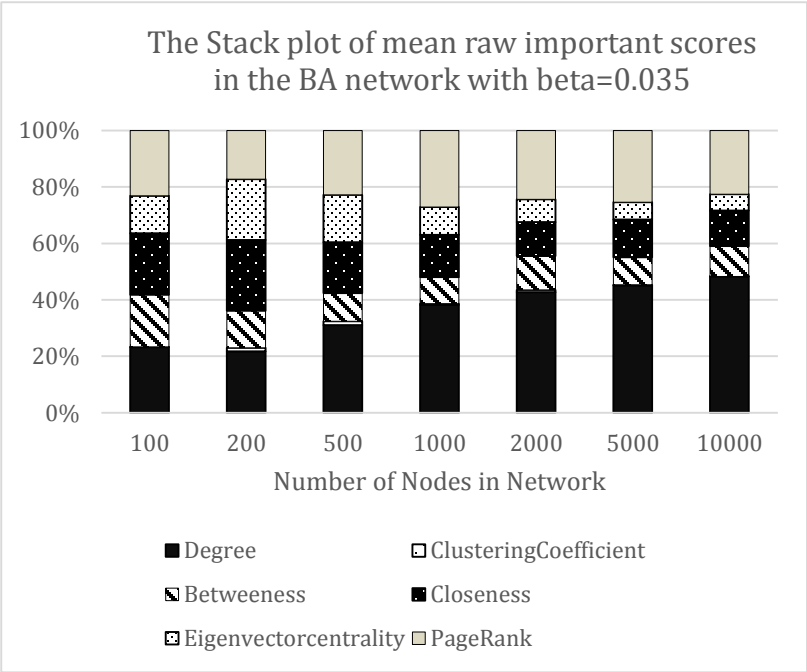

Figure 12: The stack plot of the mean raw important scores of network measurements in the BA network with different nodes number. β=0.035

To further test the stability of random forest, we varied the simulation times from 1,000 to 100,000. The stack plot shows that, although there is a small fluctuation, the rank of importance of network measurements do not change. However, increasing simulation times can help decrease OOB error in both ER and BA network (see Fig.13). For example, in the BA network with beta=0.035, OOB error decreases from 7.44% in 1000 runs nodes to 1.99% in 100,000 runs. 10,000 runs is our final choice with consideration of both the low OOB error rate and low computational cost.

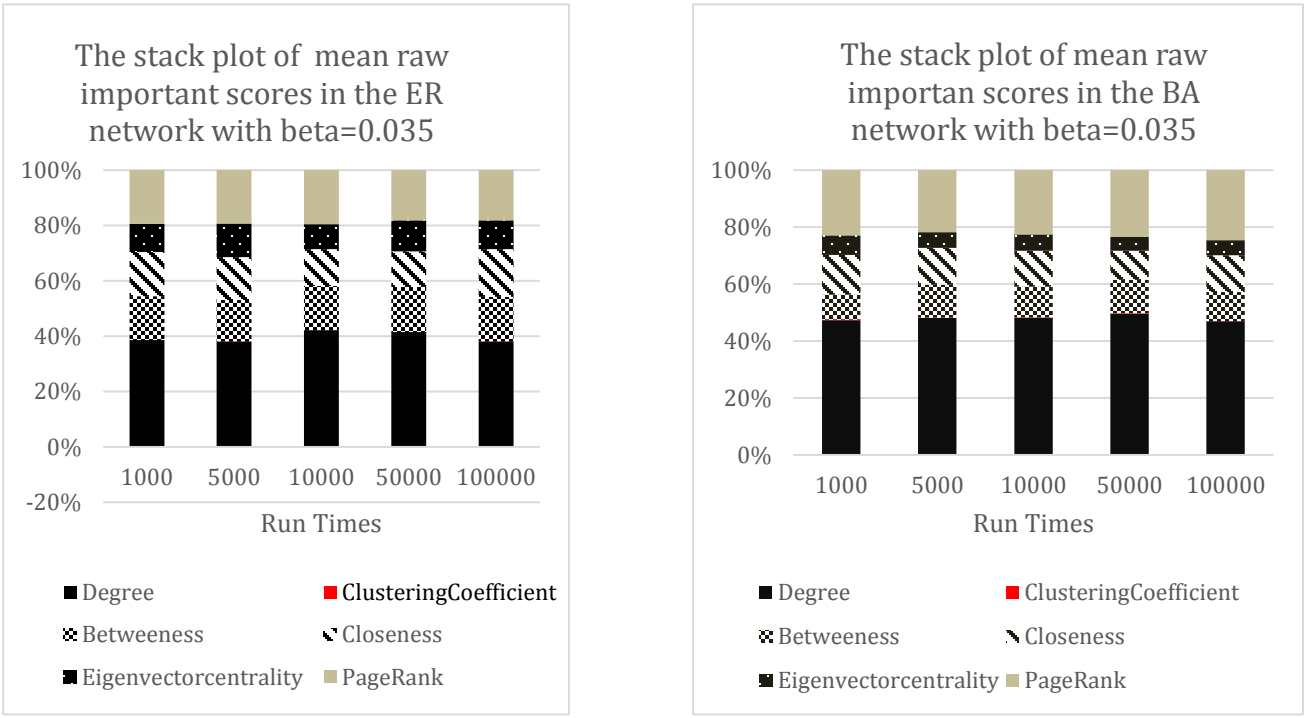

Figure 13: The stack plot of mean row important scores of network measurements with different run times

3.6 CONTACT NETWORK IN FRED

We used Jefferson County data in FRED as our research object. There are totally 45,318 people in Jefferson County synthetic population (See Table 7), 7,746 individuals at schools with 7662 students (age range from 0 to 20) and 84 teachers (age from 40 to 50), 19,927 at workplaces. Other than that, there are 2094 children (age from 0 to 5) and elderly people (age larger than 60). Gender is balanced in all subgroups. 97% of the population is white and most of the surveys are filled by the head of the household. There are 3 people in one house and 114 people on average in the neighborhood area (1km*1km). There are 26 schools in Jefferson County with an average size of 300. A total of 4,140 workplaces is located in Jefferson County with an average of 5+-SD people in each place.

Table 7: Summary of Jefferson County Synthetic Population Characteristics

Variable	Type	Number	Mean/Quantile	Comment
Age	Continuous	45316	Mean:41.14 SD:27.3	
Sex	Binary	45316	Female:51% Male:49%	
Race	Category	45316	97% White 3% Other	10 races
Relationship	Category	45316	42% Header 58% Other	14 types
Household Size	Continuous	18987	Mean:2.38 SD:1.36	
Neighborhood Size	Continuous	870	Mean:52 SD: 110	
School Size	Continuous	26	Mean:298 SD:314.4	7746 people
Classroom Size	Continuous	291	Mean: 26.46 SD:11.28	7662 students
Workplace Size	Continuous	4140	Mean:5 SD:23.96	19927 workers
Office Size	Continuous	4262	Mean:4.68 SD:9.25	19927 workers

Table 8: Summary of Subgroups in the the Jefferson County

	Students	Staff In School	Workers	Others
Number	7662	84	19926	17912
Age	Mean: 11.31 Range:0-18	Mean:43.18	Mean:43.56	0-5: 2094
		Range:20-71	Range:16-100	>60:8384
				Others:7484
Gender	50% Female	43% Female	47.5% Female	55% Female
	50% Male	57% Male	52.5% Male	45% Male

Contact network of Jefferson County was constructed based on location. Everyone in the same location in the household, classroom and office was connected with each other. We didn't consider neighborhood connection because it will make extremely high and unrealistic degree value. Network measurements were calculated and summarized below.

Table 9: Network measurements of Jefferson County Contact network

Network Measurements	Jefferson County			
Variables	Mean	Median	Min-Max	SD
Degree	432	261	0-2952	597.9
Clustering Coefficient	0.87	0.98	0-1	0.18
Betweenness Centrality	0.013	0.00055	0-1	0.031
Closeness Centrality	0.013	0.013	0.00002-0.13	0.0005
Eigenvector Centrality	0.051	0.00094	0-1	0.199
Page Rank	0.000022	0.000021	0.0000003-0.00007	0.00001

The contact network in Jefferson County is dense and highly clustered. Average clustering coefficient (mean: 0.87) and degree (mean: 600) are much higher than the BA and ER network model (See Table 9). 44% nodes have clustering coefficient and betweenness centrality equal to 1. In this case, the nodes are fully connected. 89% of those people don't have a workplace or go to school. They form a small group in their household.

The special structure of the Jefferson County contact network contributes to the special results above. When generating the network, we considered that all individuals sharing a location are connected to each other. It is a location-based network.

We constructed Spearman correlation matrix to explore the correlation between network measurements in the Jefferson County contact network. Histogram and kernel density distribution are listed in diagonal panels. The upper panels and lower panels contain Spearman correlation coefficients and scatterplot between measurements respectively.

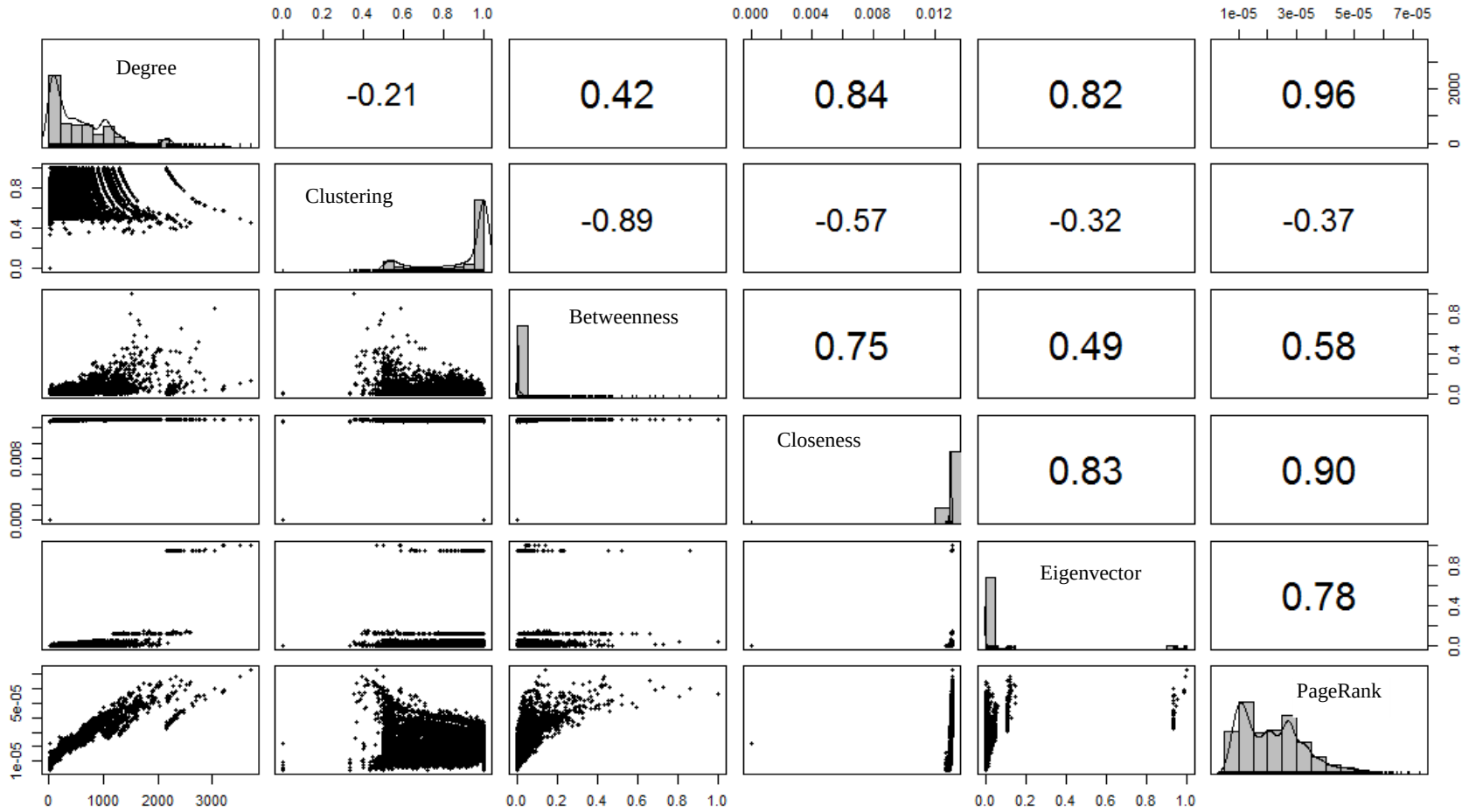

Figure 14: Spearman's correlation matrix of network measurements in Jefferson County contact network

In the Jefferson County contact network, degree and PageRank are highly correlation with Spearman's correlation coefficient of 0.96(See Fig.14). This result is similar to Barabasi-Albert and Erdos-Renyi networks. Three pairs, degree and betweenness centrality, closeness centrality and eigenvector centrality, closeness centrality and PageRank, are also highly correlated with each other with Spearman's correlation coefficients 0.84, 0.83 and 0.90, respectively.

Different from BA and ER network, clustering coefficient is negatively correlated with all centrality measurements, especially betweenness centrality (Spearman's correlation coefficients equal to -0.84) in the Jefferson County contact network. Nodes in the contact network are connected by location and form many small clusters in the network. Nodes with high clustering coefficient are fully connected and have limited outgoing connection. Betweenness centrality measures the paths that going through, thus highly clustered nodes will have lower betweenness centrality.

3.7 ATTACK RATE AND INFECTION RISK IN JEFFERSON

Two kinds of outbreaks with different transmission parameters were simulated using FRED on Jefferson County population data. Mild outbreaks (transmission parameters of 0.6) have an average attack rate of 0.2, while the average attack rate of the severe outbreaks (transmission parameter of 1.0) is 0.5.

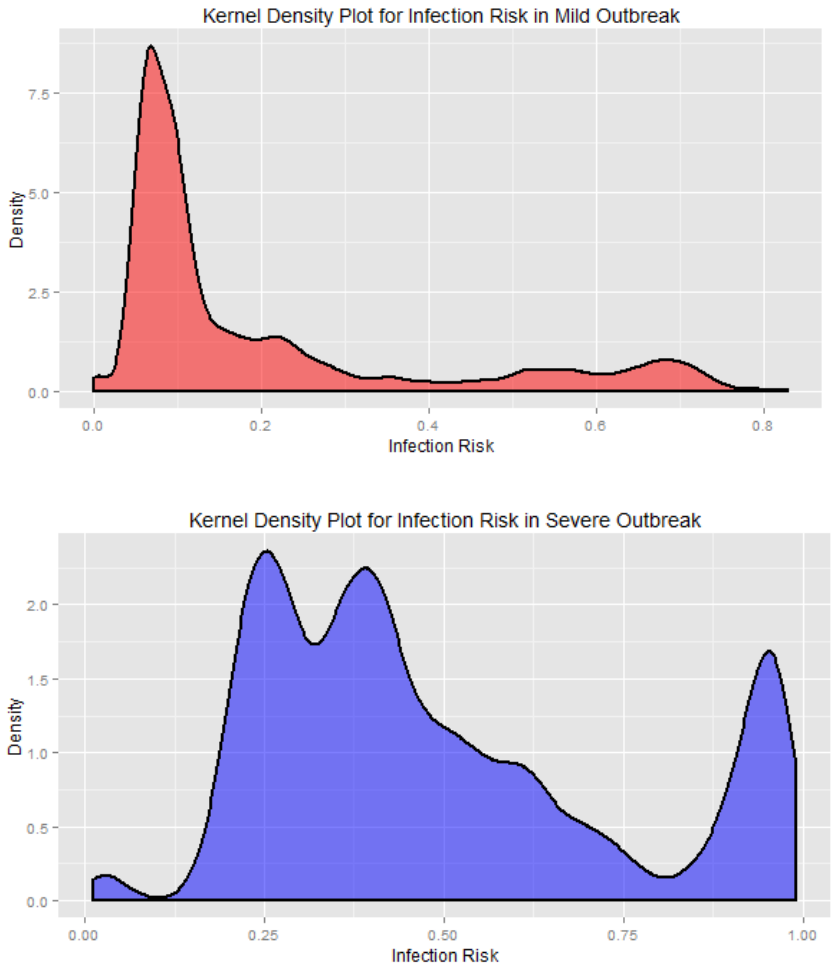

Figure 15: Kernel Density plot of infection risk in the two outbreaks of Jefferson County contact network

The infection risk distributions of mild and severe outbreaks are different both in shape and range (shown in Fig. 15). In mild outbreaks, there is one main sharp peak around 0.7 and smaller peaks in the long tail on around 0.22, 0.55 and 0.58. Whereas in severe outbreaks, there are three main peaks, around 0.25, 0.38 and 0.95, and a small peak around 0.4.

Further research on the high-risk population and low-risk groups gave us more interesting findings. We ranked the individual according to the infection risk and picked the top 7,000 cases

with the highest infection risk. Most of the cases overlap between the mild outbreak and severe outbreak.

Among the overlapped people, 96% of the high-risk people are students, 98% are within the age 0-18 years old. More interestingly, outbreak. 97% of these people are students. We ploted a histogram of these people's age (see Fig.16). There are 9 staffs in the school, 55 workers and 27 unemployed other than students.

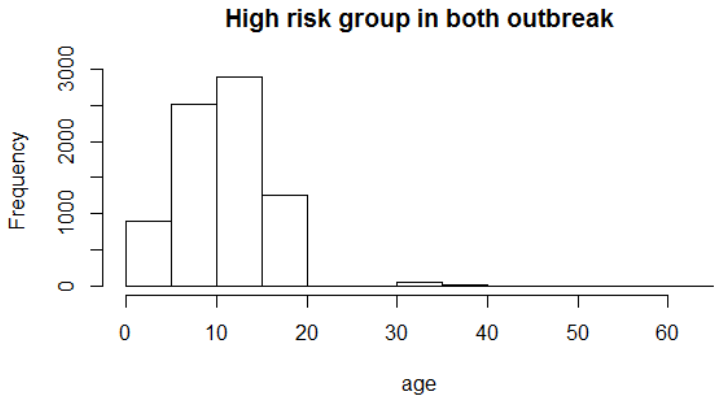

Figure 16: Histogram of age in the top 7,000 infection risk people in both mild and high outbreaks.

3.8 CLASSIFICATION RESULTS OF JEFFERSON COUNTY CONTACT NETWORK

Same grouping methods were used on mild outbreaks and severe outbreaks data of Jefferson County contact network. We sorted the whole population in ascending order by infection risk and separated into three groups with equal numbers. The first one third were the low-risk group, the last one third were the high-risk group, and others were in the medium-risk group.

We ran random forest classification algorithm on data and got the mean raw importance scores for each variable and three classification groups. We also plotted a boxplot comparing the mild break and severe outbreak with 50 running times.

Table 10: The mean raw importance score of variables in mild outbreak, Jefferson County OOB error 12%

	Mean Raw Important Score			
Variables	Low-Risk Group	Medium-Risk Group	High-Risk Group	Overall
Household Size	0.3271	0.1807	0.2176	0.2419
School Size	0.0528	0.0601	0.1445	0.0858
Betweenness Centrality	0.1404	0.0693	0.0369	0.0823
Workplace Size	0.0823	0.1463	0.0108	0.0798
Office Size	0.0899	0.1400	0.0082	0.0793
Clustering Coefficient	0.0966	0.0614	0.0313	0.0632
Age	0.0190	0.0191	0.0949	0.0443
Classroom Size	0.0308	0.0283	0.0556	0.0383
Closeness Centrality	0.0323	0.0511	0.0164	0.0333
Relationship	0.0082	0.0154	0.0463	0.0233
Degree	0.0190	0.0282	0.0130	0.0201
Eigenvector Centrality	0.0163	0.0338	0.0097	0.0199
PageRank	0.0147	0.0265	0.0122	0.0178
Neighborhood Size	0.0151	0.0214	0.0067	0.0144
Race	0.0162	0.0034	0.0009	0.0068
Sex	0.0006	0.0051	0.0034	0.0030

Table 11: The mean raw important score of variables in severe outbreaks. Jefferson County, OOB error 9.9%

	Mean Raw Important Score			
Variables	Low-risk Group	Medium-risk Group	High-risk Group	Overall
Household Size	0.3532	0.2517	0.2523	0.2858
Betweenness Centrality	0.1660	0.0850	0.0532	0.1015
Workplace Size	0.0933	0.1612	0.0163	0.0904
Office Size	0.0803	0.1425	0.0134	0.0788
School Size	0.0580	0.0602	0.1100	0.0760
Clustering Coefficient	0.0799	0.0561	0.0316	0.0559
Age	0.0184	0.0201	0.0692	0.0359
Closeness Centrality	0.0266	0.0596	0.0164	0.0342
Classroom Size	0.0307	0.0384	0.0245	0.0312
Degree	0.0182	0.0351	0.0132	0.0221
Eigenvector Centrality	0.0122	0.0305	0.0083	0.0170
Relationship	0.0110	0.0100	0.0292	0.0167
Neighborhood Size	0.0142	0.0286	0.0072	0.0166
PageRank	0.0098	0.0264	0.0111	0.0157
Race	0.0153	0.0025	0.0007	0.0062
Sex	0.0003	0.0039	0.0019	0.0020

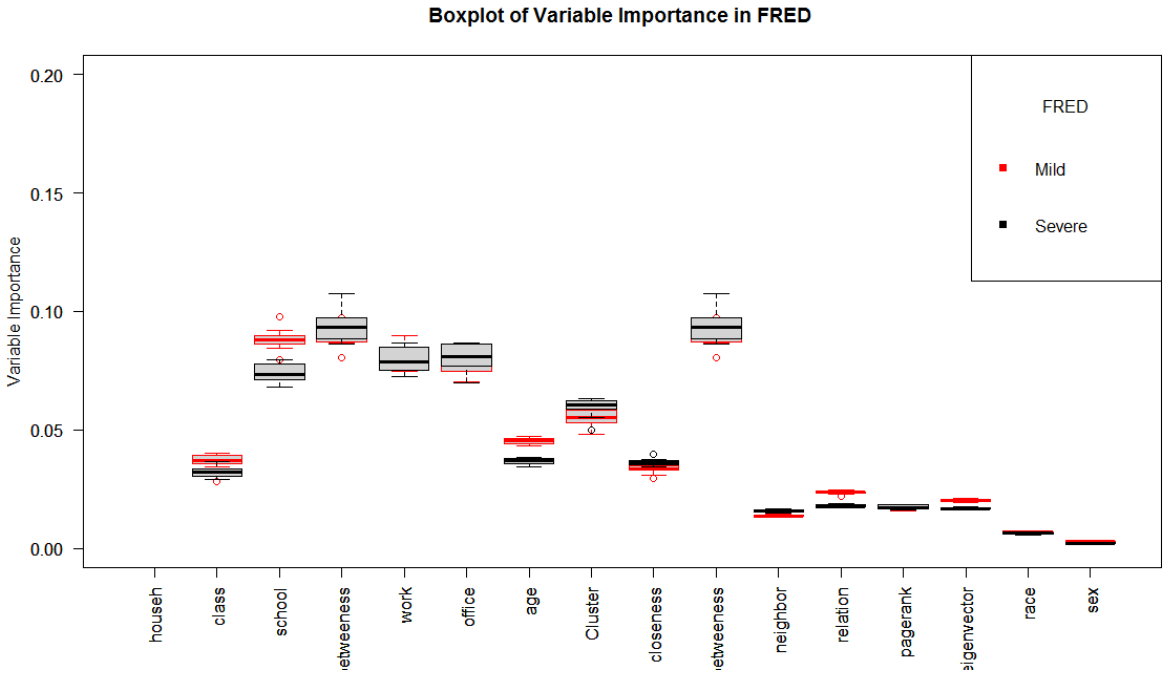

Figure 17: Boxplots of the mean raw important scores in mild and severe outbreaks of Jefferson County.

10 repeated runs.

Based on the results of mean raw important score on both mild outbreaks and severe outbreaks RF model, household size, school size, classroom size, office size and workplace size rank are important variables in RF model.

The infectious disease model in FRED is a location-based agent model. The infection spreads when people share the same location. According to the paper of Cooley [31], contact patterns of n house, class and work are different. FRED set different contact rate and infectivity for house, neighborhood, class, school, workplace and school. Infectivity also varies based on age. Default parameters are calculated using Allegheny County as an example (See Table 12). The neighborhood has the highest contact rate, but the lowest infectivity. People encounter a lot of neighbor every day but only have short time contact which makes the infectivity extremely low. Not only because limited numbers in the house, but also tuning of parameters, the household has lowest contact rate. However, the household has the highest infectivity since they have more frequently interactions.

The results in severe outbreaks are similar to those in mild outbreaks. The boxplot also shows the mean raw important scores are stable (see Fig.17). Betweenness, clustering coefficient and closeness centrality are relatively important than other network measurements in the Jefferson County contact network. This is very different from results in the BA and ER network where degree is very important and clustering coefficient has the lowest score. The main reason is that the spatial and location-based structure of contact network constructed in FRED is very different from the BA and ER network models which are not spatial nor location-based.

Table 12: Contact rate and infectivity parameters in FRED for different places.

Places	Contact Rate	Infectivity
Household	0.198	0.3-0.6
Neighborhood	42.4	0.0048
Classroom	28.64	0.0315-0.0575
School	14.32	0.0315-0.0575
Office	3	0.0575
Workplace	1.5	0.0575

When an outbreak happens, the disease will spread in small clusters and also spread to other locations by some connections. Nodes in fully connected clusters have extreme high clustering coefficient and extreme low betweenness centrality while nodes with high betweenness and low clustering coefficient will have more connections to other locations. If an outbreak happens in clusters, typically nodes in a fully connected cluster will have a higher chance to be infected. However, if an outbreak happens outside clusters, fully connected clusters will hardly be reached because they have limited connections with nodes in other locations. Nodes with high betweenness centrality will be more likely reached in this case. However, the form of relationship between infection risk and betweenness centrality or clustering coefficient is complex (See Fig 18).

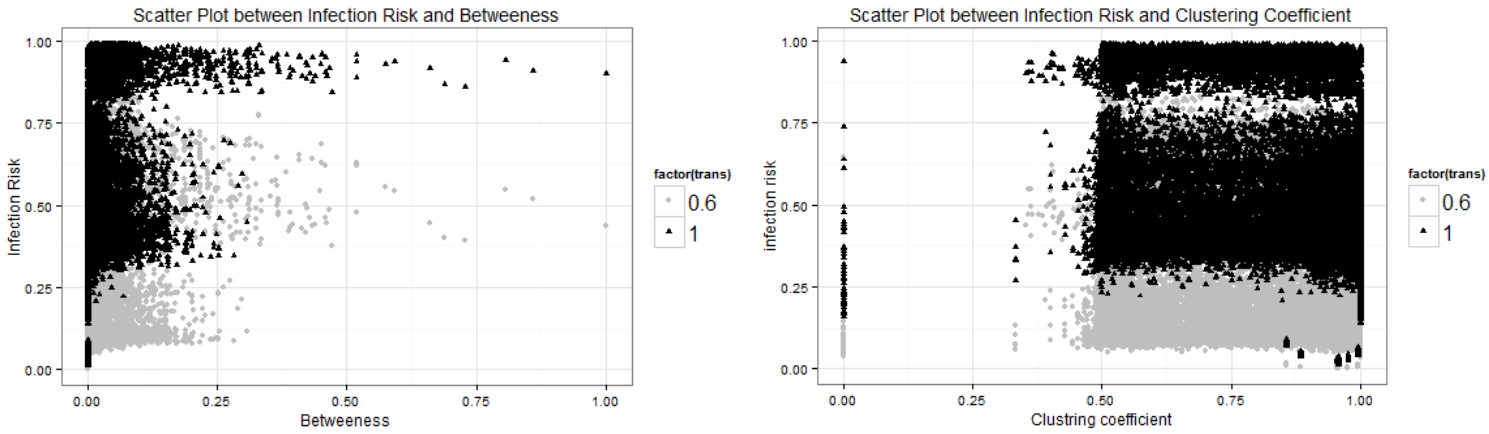

Figure 18: Scatterplot between infection risk and betweenness centrality/clustering coefficient

4.0 CONCLUSION AND DISCUSSION

With years of development in infectious disease modeling, many sophisticated mathematical models have been used in understanding the spread of infectious disease. Although we have learned a lot about disease dynamics, the use of social contact network to predict infection risk has received limited attention. It is easy to understand that more central individuals are at the greatest risk of infection during epidemic outbreaks. However, there are many different measurements of "centrality" and it might be difficult to determine which one is the most important one?

The results of BA, ER and FRED network all prove centrality measurements are of great importance in predicting infection risk. However, the structure of contact network and characteristics of a disease have an effect on the importance of network measurements in predicting infection risk. BA and ER network models are representatives of random and power-law network respectively while FRED network is location based network model. The ER network is a combination of graph theory and probability theory. It is easy to research with quite mature underlying mathematical theory and simple calculation. Some complex network, for example, ad hoc sensor network, share similar properties with the ER network. The BA network incorporates the dynamic into graph theory. The scale-free nature of the BA network broadens its application in real life. These two models have been studied extensively for their application in epidemiology. For the random network and scale-free network, random forest results show that degree is the most important variable in the Barabasi-Albert network for both mild and severe outbreak whereas in the Erdos-Renyi network it becomes important when outbreak becomes severe. It is quite intuitionally that people who have a lot of contacts will have a higher chance to be infected because the disease has more ways to directly transmit to him/her. When the mild outbreak happens on the

Erdos-Renyi network, closeness centrality is the most important variable. This result indicates that when the disease can only spread in the limited area, people who can be quickly reached will have a higher risk of being infected.

The results also suggest important applications in real life. Degree measures the direct contacts of each individual and very convenient to calculate in real life. Since degree has a monotonically increasing relationship with infection risk and plays a vital important role in classifying high-risk group in both ER and BA network, it gives a hint that if the population has a similar contact structure with the ER and BA network we can target on the people with lots of contacts as high-risk group. Unlike the "local" measurement degree, other centrality measurements, for example, betweenness centrality, closeness centrality, eigenvector centrality and PageRank are more concentrate on "global" structure. They are very difficult to calculate in real life. They require a clear and complete picture of the real life contact network which is hard to construct a contact network without tracking all the people's daily activities. Moreover, the calculation algorithm is very computational cost and time-consuming for a complex network. However, we still can get some information from the definition of those centrality measurements. Take the betweenness centrality as an example, the nodes with large betweenness usually hold the vital positions in the pathways between pairs of nodes. Hubs in the BA networks and the nodes connecting two separate communities also have large betweenness centrality. Such nodes, although not necessarily having a high degree, play the role of bridges which are connecting the nodes in two communities. Some examples can be found in real life which sharing similar characteristics with those large betweenness centrality nodes, such as public transportation. Consequently, results of betweenness centrality indicate that transportation need more attention on disease prevention for the disease spreading through direct people to people contact.

There is a different story in FRED network. The structure of the Jefferson County contact network limited the influences of degree, but betweenness centrality and clustering coefficient become more important. Although the relationship forms between betweenness centrality and infection risk, closeness centrality and infection risk, are not monatomic and very complex, the role of betweenness centrality and clustering coefficient in disease spreading in of great importance.

The disease model of FRED considers lots of factors, for example, the transmissivity of disease, contact rate and infectivity of places, age and health status. The network of FRED has many small and large clusters due to the classroom, school, office, workplace and household share. The results of RF model shows that location size is important in classifying infection risk in the Jefferson County contact network. Thus in real life, a larger company and school are under higher risk and need more attention on the disease prevention program.

Correlations among network measurements are also influenced by the network structure. In the Erdos-Renyi network, i.e., all measurements are highly correlated, but this is not the case in the Barabasi-Albert network. Also, degree and PageRank are highly correlated with similar distributions in all three network models. In the paper of Jose, they came up with an idea to collect correlations between network measurements and build profiles of different types of networks [30]. The similarity and differences of correlations between network measurements may give some clues on the topology of networks. Besides, for the complex network and some real life network with unclear organizing rules, profiles of different types of network can give some important information about their type and structure.

Public Health Impact

The important variables that we find in this thesis will help build an efficient surveillance system. Rather than conducting a prevention program that covers all the susceptive people, a targeted prevention program can focus more on the high-risk population with some certain characteristics. Degree suggests targeting people with more contacts while betweenness centrality focuses on the people who act as connecting bridges. Results in FRED suggests school, workplace and house with larger size are under higher risk. The Larger company and school should receive more attention on disease prevention.

5.0 LIMITATION AND FUTURE WORK

5.1.1 Highly Correlated Variable

The highly-correlated variable is a difficult subject to handle in statistics. According to Robin's studies [27], variable importance in the random forest may have a preference of highly correlated variables. An improved random forest method called "conditional inference tree" using the cforest package has been developed to deal with the highly correlated variable issue and has been proven effective by several studies. However, it requires enormous computational resources and cannot handle the large dataset we study in this thesis.

5.1.2 Statistical Inference of variable importance

Variable importance has been widely used in many fields for variable selection, such as genomic analysis and pattern recognition. Some researchers realize the importance of interpreting variable importance using statistic languages and make some contributions [28]. However, limited researches have been done by statisticians.

5.1.3 FRED network

It is very difficult to get a real epidemic disease and contact network data, thus FRED is a valuable tool which provides real population information and use computers to simulate the epidemic. However, in our studies, it is challenging to construct a network using FRED. The simple

assumption that everyone is connected with each other if they share a location is not easy to make in real life, depending on the size and nature of the interaction of individuals. Weighted networks with degree proportional to the contact rate in different places may be a solution in solving large degree problems.

APPENDIX: R CODE

I. BA and ER Network

```
####Attack Rate Summarize####
library(ggplot2)
BA.0.035$β<-0.035
BA0.05$β<-0.05
BA0.07$β<-0.07
BA<-rbind(BA.0.035,BA0.05)
m1<-ggplot(BA,aes(x=V1,group=β))
m1+geom_density(aes(fill=factor(β)), size=0.8,alpha=0.5)+xlab("Attack Rate") +ylab("Kernel Density") +ggtitle("Kernel Density Plot for Attack Rate in Barabasi-Albert Network")

ER0.035$β<-0.035
ER0.05$β<-0.05
ER<-rbind(ER0.035,ER0.05)

m2<-ggplot(ER,aes(x=V1,group=β))
m2+geom_density(aes(fill=factor(β)), size=0.8,alpha=0.5)+xlab("Attack Rate") +ylab("Kernel Density") +ggtitle("Kernel Density Plot for Attack Rate in Erdos-Renyi Network")
##### Network Characteristics
summary(BA0.035)
apply(BA0.035,2,sd)
summary(ER0.035)
apply(ER0.035,2,sd)



#####Decriptive Analysis
BA0.035$β<-0.035
BA0.05$β<-0.05
BA0.05<-BA0.05[,c(1:10,12,13)]
BA<-rbind(BA0.035,BA0.05)

ER0.035$β<-0.035
ER0.05$β<-0.05
ER20.05<-ER0.05[,c(1:10,14,15)]
ER<-rbind(ER0.035,ER20.05)

library(ggplot2)
m1<-ggplot(BA,aes(x=V1,group=β))
m1+geom_density(aes(fill=factor(β)), size=0.8,alpha=0.5)+xlab("Infection Risk") +ylab("Density") +ggtitle("Kernel Density Plot for Infection Risk")

m2<-ggplot(BA,aes(x=V2,group=β))
m2+geom_density(aes(fill=factor(β)), size=0.8,alpha=0.5)+xlab("Speed") +ylab("Density") +ggtitle("Kernel Density Plot for Speed")

p<-ggplot(BA,aes(x=Degree,y=InfectionRisk,group=β))
p+ geom_point(aes(colour = factor(β),shape = factor(β)))+theme(legend.text=element_text(size=16))+xlab("log(Degree)") +ylab("log(Infection Risk)j") +ggtitle("log-log Scatter Plot between Infection Risk and Degree")

p1<-ggplot(BA,aes(x=log(ClusteringCoefficient),y=log(InfectionRisk),group=β))
p1+ geom_point(aes(colour = factor(β),shape = factor(β)))+theme(legend.text=element_text(size=16))+xlab("log Clustering Coefficient") +ylab("log Infection Risk") +ggtitle("Log-Log Scatter Plot between Infection Risk and Clustering Coefficient")

p2<-ggplot(BA,aes(x=log(Betweenness),y=log(InfectionRisk),group=β))
p2+ geom_point(aes(colour = factor(β),shape = factor(β)))+theme(legend.text=element_text(size=16))+xlab("log Betweenness") +ylab("log Infection Risk") +ggtitle("Log-Log Scatter Plot between Infection Risk and Betweenness")

p3<-ggplot(BA,aes(x=log(Closeness),y=log(InfectionRisk),group=β))
p3+ geom_point(aes(colour = factor(β),shape = factor(β)))+theme(legend.text=element_text(size=16))+xlab("log Closeness") +ylab("log Infection Risk") +ggtitle("Log-Log Scatter Plot between Infection Risk and Closeness")

p4<-ggplot(BA,aes(x=log(Eigenvectorcentrality),y=log(InfectionRisk),group=β))
p4+ geom_point(aes(colour = factor(β),shape = factor(β)))+theme(legend.text=element_text(size=16))+xlab("log Eigenvector centrality") +ylab("log Infection Risk") +ggtitle("Log-Log Scatter Plot between Infection Risk and Eigenvector Centrality")

p5<-ggplot(BA,aes(x=log(PageRank),y=log(InfectionRisk),group=β))
p5+ geom_point(aes(colour = factor(β),shape = factor(β)))+theme(legend.text=element_text(size=16))+xlab("log PageRank") +ylab("log Infection Risk") +ggtitle("Log-Log Scatter Plot between Infection Risk and PageRank")

p6<-ggplot(BA,aes(x=log(PageRank),y=log(Speed),group=β))
```

```
p6+ geom_point(aes(colour = factor(β),shape = factor(β)))+theme(legend.text=element_text(size=16))+xlab("log Degree") +ylab("log Speed") +ggtitle("Log-Log Scatter Plot between Speed and Degree")


m1<-ggplot(ER,aes(x=V1,group=β))
m1+geom_density(aes(fill=factor(β)), size=0.8,alpha=0.5)+xlab("Infection Risk") +ylab("Density") +ggtitle("Kernel Density Plot for Infection Risk")

m2<-ggplot(ER,aes(x=V2,group=β))
m2+geom_density(aes(fill=factor(β)), size=0.8,alpha=0.5)+xlab("Speed") +ylab("Density") +ggtitle("Kernel Density Plot for Speed")

p<-ggplot(ER,aes(x=log(Degree),y=log(InfectionRisk),group=β))
p+ geom_point(aes(colour = factor(β),shape = factor(β)))+theme(legend.text=element_text(size=16))+xlab("log(Degree)") +ylab("log(Infection Risk)j") +ggtitle("log-log Scatter Plot between Infection Risk and Degree")

p1<-ggplot(ER,aes(x=log(ClusteringCoefficient),y=log(InfectionRisk),group=β))
p1+ geom_point(aes(colour = factor(β),shape = factor(β)))+theme(legend.text=element_text(size=16))+xlab("log Clustering Coefficient") +ylab("log Infection Risk") +ggtitle("Log-Log Scatter Plot between Infection Risk and Clustering Coefficient")

p2<-ggplot(ER,aes(x=log(Betweenness),y=log(InfectionRisk),group=β))
p2+ geom_point(aes(colour = factor(β),shape = factor(β)))+theme(legend.text=element_text(size=16))+xlab("log Betweenness") +ylab("log Infection Risk") +ggtitle("Log-Log Scatter Plot between Infection Risk and Betweenness")

p3<-ggplot(ER,aes(x=log(Closeness),y=log(InfectionRisk),group=β))
p3+coord_cartesian(xlim = c(-2.0, -1.5)) + geom_point(aes(colour = factor(β),shape = factor(β)))+theme(legend.text=element_text(size=16))+xlab("log Closeness") +ylab("log Infection Risk") +ggtitle("Log-Log Scatter Plot between Infection Risk and Closeness")

p4<-ggplot(ER,aes(x=log(Eigenvectorcentrality),y=log(InfectionRisk),group=β))
p4+ coord_cartesian(xlim = c(-4.2, 1)) +geom_point(aes(colour = factor(β),shape = factor(β)))+theme(legend.text=element_text(size=16))+xlab("log Eigenvector centrality") +ylab("log Infection Risk") +ggtitle("Log-Log Scatter Plot between Infection Risk and Eigenvector Centrality")

p5<-ggplot(ER,aes(x=log(PageRank),y=log(InfectionRisk),group=β))
p5+ geom_point(aes(colour = factor(β),shape = factor(β)))+theme(legend.text=element_text(size=16))+xlab("log PageRank") +ylab("log Infection Risk") +ggtitle("Log-Log Scatter Plot between Infection Risk and PageRank")

p6<-ggplot(ER,aes(x=log(PageRank),y=log(Speed),group=β))
p6+ geom_point(aes(colour = factor(β),shape = factor(β)))+theme(legend.text=element_text(size=16))+xlab("log Degree") +ylab("log Speed") +ggtitle("Log-Log Scatter Plot between Speed and Degree")

###############Network Measurements degree betweenness cc, closeness, eigenvectore centrality and PageRank

library(psych)

pairs.panels(BA0.035[,c(3,5,6,7,8)], smooth=FALSE,density=TRUE,ellipses=FALSE,digits =2,method="spearman", pch = 20, cor=TRUE,hist.col="Grey")
pairs.panels(ER0.035[,c(3,5,6,7,8)], smooth=FALSE,density=TRUE,ellipses=FALSE,digits =2,method="spearman", pch = 20, cor=TRUE,hist.col="Grey")


#####Network
colnames(BA0.035)[1]<-"InfectionRisk"
colnames(BA0.035)[2]<-"Speed"
colnames(BA0.035)[3]<-"Degree"
colnames(BA0.035)[4]<-"ClusteringCoefficient"
colnames(BA0.035)[5]<-"Betweenness"
colnames(BA0.035)[6]<-"Closeness"
colnames(BA0.035)[7]<-"Eigenvectorcentrality"
colnames(BA0.035)[8]<-"PageRank"
colnames(BA0.035)[9]<-"Kcore"
colnames(BA0.035)[10]<-"Knn"


colnames(BA0.05)[1]<-"InfectionRisk"
colnames(BA0.05)[2]<-"Speed"
colnames(BA0.05)[3]<-"Degree"
colnames(BA0.05)[4]<-"ClusteringCoefficient"
colnames(BA0.05)[5]<-"Betweenness"
colnames(BA0.05)[6]<-"Closeness"
colnames(BA0.05)[7]<-"Eigenvectorcentrality"
colnames(BA0.05)[8]<-"PageRank"
colnames(BA0.05)[9]<-"Kcore"
colnames(BA0.05)[10]<-"Knn"

colnames(BA0.07)[1]<-"InfectionRisk"
colnames(BA0.07)[2]<-"Speed"
colnames(BA0.07)[3]<-"Degree"
colnames(BA0.07)[4]<-"ClusteringCoefficient"
colnames(BA0.07)[5]<-"Betweenness"
colnames(BA0.07)[6]<-"Closeness"
colnames(BA0.07)[7]<-"Eigenvectorcentrality"
colnames(BA0.07)[8]<-"PageRank"
```

```
colnames(BA0.07)[9]<-"Kcore"
colnames(BA0.07)[10]<-"Knn"

colnames(ER0.035)[1]<-"InfectionRisk"
colnames(ER0.035)[2]<-"Speed"
colnames(ER0.035)[3]<-"Degree"
colnames(ER0.035)[4]<-"ClusteringCoefficient"
colnames(ER0.035)[5]<-"Betweenness"
colnames(ER0.035)[6]<-"Closeness"
colnames(ER0.035)[7]<-"Eigenvectorcentrality"
colnames(ER0.035)[8]<-"PageRank"
colnames(ER0.035)[9]<-"Kcore"
colnames(ER0.035)[10]<-"Knn"


colnames(ER0.05)[1]<-"InfectionRisk"
colnames(ER0.05)[2]<-"Speed"
colnames(ER0.05)[3]<-"Degree"
colnames(ER0.05)[4]<-"ClusteringCoefficient"
colnames(ER0.05)[5]<-"Betweenness"
colnames(ER0.05)[6]<-"Closeness"
colnames(ER0.05)[7]<-"Eigenvectorcentrality"
colnames(ER0.05)[8]<-"PageRank"
colnames(ER0.05)[9]<-"Kcore"
colnames(ER0.05)[10]<-"Knn"

colnames(ER0.07)[1]<-"InfectionRisk"
colnames(ER0.07)[2]<-"Speed"
colnames(ER0.07)[3]<-"Degree"
colnames(ER0.07)[4]<-"ClusteringCoefficient"
colnames(ER0.07)[5]<-"Betweenness"
colnames(ER0.07)[6]<-"Closeness"
colnames(ER0.07)[7]<-"Eigenvectorcentrality"
colnames(ER0.07)[8]<-"PageRank"
colnames(ER0.07)[9]<-"Kcore"
colnames(ER0.07)[10]<-"Knn"

library(party)
library(randomForest)
#####Parameters optimization
rf_model<-train(class~.,data=BA0.05.tree,method="rf",trControl=trainControl(method="cv",number=5), prox=TRUE,allowParallel=TRUE)
print(rf_model)

#mtry select


#####function for 50 runs time


rfbox<-function(dataset,runs=50,mtr=4,ntr=501){
  myVarsum=matrix(,ncol=50,nrow=6)
  confusion<-matrix(,ncol=)
  errors<-dim(runs)
  for(i in 1:runs){
    set.seed(i+50)
    rf<-randomForest(formula=class~.,data=dataset,ntree=ntr,replace=T,mtry=mtr, importance=T,na.action = na.omit)
    pro<-predict(rf)
    obs<-dataset$class
    confusionmatrix<-
    error<-mean(rf$err.rate[,1])
    errors[i] = error
    set.seed(i+50)
    myVar<-importance(rf,scale=F,type=1)
    myVarsum[,i]=myVar
  }
  myVarsum=t(myVarsum)
  err<pmean(errors)
  print(err)
  return(myVarsum)
}

###categorized the infection risk
BA0.035$class<-cut(BA0.035$InfectionRisk,quantile(BA0.035$InfectionRisk,probs = c(0,1,2,3)/3),label=c("low","medium","high"))
BA0.05$class<-cut(BA0.05$InfectionRisk,quantile(BA0.05$InfectionRisk,probs = c(0,1,2,3)/3),label=c("low","medium","high"))
BA0.07$class<-cut(BA0.07$InfectionRisk,quantile(BA0.07$InfectionRisk,probs = c(0,1,2,3)/3),label=c("low","medium","high"))
ER0.035$class<-cut(ER0.035$InfectionRisk,quantile(ER0.035$InfectionRisk,probs = c(0,1,2,3)/3),label=c("low","medium","high"))
ER0.05$class<-cut(ER0.05$InfectionRisk,quantile(ER0.05$InfectionRisk,probs = c(0,1,2,3)/3),label=c("low","medium","high"))
ER0.07$class<-cut(ER0.07$InfectionRisk,quantile(ER0.07$InfectionRisk,probs = c(0,1,2,3)/3),label=c("low","medium","high"))
```

```
###Select interest features
BA0.035.tree<-BA0.035[,c(3:8,11)]
BA0.05.tree<-BA0.05[,c(3:8,14)]
BA0.07.tree<-BA0.07[,c(3:8,14)]
ER0.035.tree<-ER0.035[,c(3:8,11)]
ER0.05.tree<-ER0.05[,c(3:8,14)]
ER0.07.tree<-ER0.07[,c(3:8,14)]

#####run the random forest function

rf1<-rfbox(dataset=BA0.035.tree)
rf2<-rfbox(dataset=BA0.05.tree)
rf3<-rfbox(dataset=BA0.07.tree)
rf4<-rfbox(dataset=ER0.035.tree)
rf5<-rfbox(dataset=ER0.05.tree)
rf6<-rfbox(dataset=ER0.07.tree)

#####graph reuslts
colnames(rf1)<-c("Degree","Clustering Coefficient","Betweenness","Closeness","Eigenvector centrality","Page Rank")
colnames(rf2)<-c("Degree","Clustering Coefficient","Betweenness","Closeness","Eigenvector centrality","Page Rank")
colnames(rf3)<-c("Degree","Clustering Coefficient","Betweenness","Closeness","Eigenvector centrality","Page Rank")
colnames(rf4)<-c("Degree","Clustering Coefficient","Betweenness","Closeness","Eigenvector centrality","Page Rank")
colnames(rf5)<-c("Degree","Clustering Coefficient","Betweenness","Closeness","Eigenvector centrality","Page Rank")
colnames(rf6)<-c("Degree","Clustering Coefficient","Betweenness","Closeness","Eigenvector centrality","Page Rank")
rf1<-rf1[,c(1,6,3,4,5,2)]
rf2<-rf2[,c(1,6,3,4,5,2)]
rf5<-rf5[,c(1,6,3,4,5,2)]
rf4<-rf4[,c(1,6,3,4,5,2)]

library(ggplot2)
rf1_means <- (colMeans(rf1, na.rm = TRUE))
boxplot(rf1,xlab="Network Measurements",ylim=c(0,0.5),ylab="Variable Importance",main="Boxplot of Variable Importance of Network Measurements
in BA ",col="light grey", border="red")
par(new=TRUE)
rf2_means <- (colMeans(rf2, na.rm = TRUE))
boxplot(rf2,col="light grey", border="black",ylim=c(0,0.5),)
legend("topright", title="B  ",c("0.035","0.05"), cex=1,pch=c(15,15),col=c("RED","BLACK"), horiz=FALSE)

rf4_means <- (colMeans(rf4, na.rm = TRUE))
boxplot(rf4,xlab="Network Measurements",ylim=c(0,0.3),ylab="Variable Importance",main="Boxplot of Variable Importance of Network Measurements
in ER ",col="light grey", border="red")
par(new=TRUE)
rf5_means <- (colMeans(rf5, na.rm = TRUE))
boxplot(rf5, col="light grey", border="black",ylim=c(0,0.3),)
legend("topright", title="B  ",c("0.035","0.05"), cex=1,pch=c(15,15),col=c("RED","BLACK"), horiz=FALSE)
}
```

Network Size/Run times/ B

```
###########R code#Code for final thesis
library(party)
library(randomForest)
N100$class<-cut(N100$InfectionRisk,quantile(N100$InfectionRisk,probs = c(0,1,2,3)/3),label=c("low","medium","high"))
N200$class<-cut(N200$InfectionRisk,quantile(N200$InfectionRisk,probs = c(0,1,2,3)/3),label=c("low","medium","high"))
N500$class<-cut(N500$InfectionRisk,quantile(N500$InfectionRisk,probs = c(0,1,2,3)/3),label=c("low","medium","high"))
N1000$class<-cut(N1000$InfectionRisk,quantile(N1000$InfectionRisk,probs = c(0,1,2,3)/3),label=c("low","medium","high"))
N2000$class<-cut(N2000$InfectionRisk,quantile(N2000$InfectionRisk,probs = c(0,1,2,3)/3),label=c("low","medium","high"))
N5000$class<-cut(N5000$InfectionRisk,quantile(N5000$InfectionRisk,probs = c(0,1,2,3)/3),label=c("low","medium","high"))
N10000$class<-cut(N10000$InfectionRisk,quantile(N10000$InfectionRisk,probs = c(0,1,2,3)/3),label=c("low","medium","high"))

rfav<-function(dataset,runs=10,mtr=2,ntr=501){
  myVarsum=c(0,0,0,0,0,0)
  errors<-dim(runs)
  for(i in 1:runs){
    set.seed(i)
    rf<-randomForest(formula=class~.,data=dataset,ntree=ntr,replace=T,mtry=mtr, importance=T,na.action = na.omit)
    error<-mean(rf$err.rate[,1])
    errors[i] = error
    set.seed(i)
    myVar<-importance(rf,scale=F,type=1)
    myVarsum=myVarsum+myVar
  }
  myVarav=myVarsum/runs
  err<-mean(errors)
  print(myVarav)
  print(err)
}
```

```
rfbox<-function(dataset,runs=50,mtr=2,ntr=501){
  myVarsum=matrix(,ncol=50,nrow=6)
  errors<-dim(runs)
  for(i in 1:runs){
    rf<-randomForest(formula=class~.,data=dataset,ntree=ntr,replace=T,mtry=mtr, importance=T,na.action = na.omit)
    error<-mean(rf$err.rate[,1])
    errors[i] = error
    myVar<-importance(rf,scale=F,type=1)
    myVarsum[,i]=myVar
  }
  myVarsum=T(myVarsum)
  return(myVarsum)
}

#bootstap random forest
N100.tree<-N100[,c(3:8,11)]
N200.tree<-N200[,c(3:8,11)]
N500.tree<-N500[,c(3:8,11)]
N1000.tree<-N1000[,c(3:8,11)]
N2000.tree<-N2000[,c(3:8,11)]
N5000.tree<-N5000[,c(3:8,11)]
N10000.tree<-N10000[,c(3:8,11)]


dfit<-rpart(formula=class~.,data=ba3.tree,method="class",control=rpart.control(cp=0,xval=10))
dfit<-rpart(formula=class~.,data=ba5.tree,method="class",control=rpart.control(cp=0,xval=10))

plot(dfit) #main="Complete Tree for binary outcome"
text(dfit,use.n=T,xpd = TRUE)
printcp(dfit)   ###complexity parameter
plotcp(dfit)


####  ABOVE is the full tree#####


#### Below is Pruning the tree#####
fit3<-prune(dfit,cp= 2.4180e-03 )
plot(fit3) #,main="Optimized Tree for Binary outcome"
text(fit3,use.n=T,xpd = TRUE)


##### Imprtance of Variable on bootstrap samples#####

rf1<-rfav(dataset=N100.tree)
rf2<-rfav(dataset=N200.tree)
rf3<-rfav(dataset=N500.tree)
rf4<-rfav(dataset=N1000.tree)
rf5<-rfav(dataset=N2000.tree)
rf6<-rfav(dataset=N5000.tree)
rf7<-rfav(dataset=N10000.tree)

######K

k6$class<-cut(k6$InfectionRisk,quantile(k6$InfectionRisk,probs = c(0,1,2,3)/3),label=c("low","medium","high"))
k10$class<-cut(k10$InfectionRisk,quantile(k10$InfectionRisk,probs = c(0,1,2,3)/3),label=c("low","medium","high"))
k20$class<-cut(k20$InfectionRisk,quantile(k20$InfectionRisk,probs = c(0,1,2,3)/3),label=c("low","medium","high"))
k40$class<-cut(k40$InfectionRisk,quantile(k40$InfectionRisk,probs = c(0,1,2,3)/3),label=c("low","medium","high"))
k100$class<-cut(k100$InfectionRisk,quantile(k100$InfectionRisk,probs = c(0,1,2,3)/3),label=c("low","medium","high"))
k200$class<-cut(k200$InfectionRisk,quantile(k200$InfectionRisk,probs = c(0,1,2,3)/3),label=c("low","medium","high"))


#bootstap random forest
k6.tree<-k6[,c(3:8,12)]
k10.tree<-k10[,c(3:8,12)]
k20.tree<-k20[,c(3:8,12)]
k40.tree<-k40[,c(3:8,12)]
k100.tree<-k100[,c(3:8,12)]
k200.tree<-k200[,c(3:8,12)]

rf1<-rfav(dataset=k6.tree)
rf2<-rfav(dataset=k10.tree)
rf3<-rfav(dataset=k20.tree)
rf4<-rfav(dataset=k40.tree)
rf5<-rfav(dataset=k100.tree)
rf6<-rfav(dataset=k200.tree)

###B
```

```
b0.035$class<-cut(b0.035$InfectionRisk,quantile(b0.035$InfectionRisk,probs = c(0,1,2,3)/3),label=c("low","medium","high"))
b0.03$class<-cut(b0.03$InfectionRisk,quantile(b0.03$InfectionRisk,probs = c(0,1,2,3)/3),label=c("low","medium","high"))
b0.05$class<-cut(b0.05$InfectionRisk,quantile(b0.05$InfectionRisk,probs = c(0,1,2,3)/3),label=c("low","medium","high"))
b0.07$class<-cut(b0.07$InfectionRisk,quantile(b0.07$InfectionRisk,probs = c(0,1,2,3)/3),label=c("low","medium","high"))
b0.02$class<-cut(b0.02$InfectionRisk,quantile(b0.02$InfectionRisk,probs = c(0,1,2,3)/3),label=c("low","medium","high"))


#bootstap random forest
b0.02.tree<-b0.02[,c(3:8,14)]
b0.03.tree<-b0.03[,c(3:8,11)]
b0.035.tree<-b0.035[,c(3:8,11)]
b0.05.tree<-b0.05[,c(3:8,14)]
b0.07.tree<-b0.07[,c(3:8,14)]

rf1<-rfav(dataset=b0.02.tree)
rf2<-rfav(dataset=b0.03.tree)
rf3<-rfav(dataset=b0.035.tree)
rf4<-rfav(dataset=b0.05.tree)
rf5<-rfav(dataset=b0.07.tree)


###N β=0.035 runs varies

n1000$class<-cut(n1000$InfectionRisk,quantile(n1000$InfectionRisk,probs = c(0,1,2,3)/3),label=c("low","medium","high"))
n5000$class<-cut(n5000$InfectionRisk,quantile(n5000$InfectionRisk,probs = c(0,1,2,3)/3),label=c("low","medium","high"))
n10000$class<-cut(n10000$InfectionRisk,quantile(n10000$InfectionRisk,probs = c(0,1,2,3)/3),label=c("low","medium","high"))
n50000$class<-cut(n50000$InfectionRisk,quantile(n50000$InfectionRisk,probs = c(0,1,2,3)/3),label=c("low","medium","high"))
n100000$class<-cut(n100000$InfectionRisk,quantile(n100000$InfectionRisk,probs = c(0,1,2,3)/3),label=c("low","medium","high"))


n1000.tree<-n1000[,c(3:8,11)]
n5000.tree<-n5000[,c(3:8,11)]
n10000.tree<-n10000[,c(3:8,11)]
n50000.tree<-n50000[,c(3:8,11)]
n100000.tree<-n100000[,c(3:8,11)]

rf1<-rfav(dataset=n1000.tree)
rf2<-rfav(dataset=n5000.tree)
rf3<-rfav(dataset=n10000.tree)
rf4<-rfav(dataset=n50000.tree)
rf5<-rfav(dataset=n100000.tree)


;####β=0.07


r1000$class<-cut(r1000$InfectionRisk,quantile(r1000$InfectionRisk,probs = c(0,1,2,3)/3),label=c("low","medium","high"))
r5000$class<-cut(r5000$InfectionRisk,quantile(r5000$InfectionRisk,probs = c(0,1,2,3)/3),label=c("low","medium","high"))
r10000$class<-cut(r10000$InfectionRisk,quantile(r10000$InfectionRisk,probs = c(0,1,2,3)/3),label=c("low","medium","high"))
r50000$class<-cut(r50000$InfectionRisk,quantile(r50000$InfectionRisk,probs = c(0,1,2,3)/3),label=c("low","medium","high"))
r100000$class<-cut(r100000$InfectionRisk,quantile(r100000$InfectionRisk,probs = c(0,1,2,3)/3),label=c("low","medium","high"))


r1000.tree<-r1000[,c(3:8,11)]
r5000.tree<-r5000[,c(3:8,11)]
r10000.tree<-r10000[,c(3:8,14)]
r50000.tree<-r50000[,c(3:8,11)]
r100000.tree<-r100000[,c(3:8,11)]

rf1<-rfav(dataset=r1000.tree)
rf2<-rfav(dataset=r5000.tree)
rf3<-rfav(dataset=r10000.tree)
rf4<-rfav(dataset=r50000.tree)
rf5<-rfav(dataset=r100000.tree)
```

FRED

```
##########R code for FRED
jeff1.0 <- read.table("F:/Files/Project/Thesis/fred/inf_risk_jefferson_tr1.0.dat", quote="\"",fill = TRUE)
jeff0.6 <- read.table("F:/Files/Project/Thesis/fred/inf_risk_jefferson_tr0.6.dat", quote="\"",fill = TRUE)
jeff1.5 <- read.table("F:/Files/Project/Thesis/fred/inf_risk_jefferson_tr1.5.dat", quote="\"",fill = TRUE)
jeff0.6<-na.omit(jeff0.6)
jeff1.0<-na.omit(jeff1.0)
jeff1.5<-na.omit(jeff1.5)
```

```
for (i in 5 :10) {
  jeff0.6[,i]<-as.numeric(levels(jeff0.6[,i])[jeff0.6[,i]])
}
for (i in 5 :10) {
  jeff1.0[,i]<-as.numeric(levels(jeff1.0[,i])[jeff1.0[,i]])
}


colnames(jeff0.6)[1]<-"id"
colnames(jeff0.6)[2]<-"infectionrisk"
colnames(jeff0.6)[3]<-"age"
colnames(jeff0.6)[4]<-"sex"
colnames(jeff0.6)[5]<-"race"
colnames(jeff0.6)[6]<-"relationship"
colnames(jeff0.6)[7]<-"degree"
colnames(jeff0.6)[8]<-"household"
colnames(jeff0.6)[9]<-"neighborhood"
colnames(jeff0.6)[10]<-"school"
colnames(jeff0.6)[11]<-"classroom"
colnames(jeff0.6)[12]<-"workplace"
colnames(jeff0.6)[13]<-"office"

colnames(jeff1.0)[1]<-"id"
colnames(jeff1.0)[2]<-"infectionrisk"
colnames(jeff1.0)[3]<-"age"
colnames(jeff1.0)[4]<-"sex"
colnames(jeff1.0)[5]<-"race"
colnames(jeff1.0)[6]<-"relationship"
colnames(jeff1.0)[7]<-"degree"
colnames(jeff1.0)[8]<-"household"
colnames(jeff1.0)[9]<-"neighborhood"
colnames(jeff1.0)[10]<-"school"
colnames(jeff1.0)[11]<-"classroom"
colnames(jeff1.0)[12]<-"workplace"
colnames(jeff1.0)[13]<-"office"

colnames(jeff1.5)[1]<-"id"
colnames(jeff1.5)[2]<-"infection risk"
colnames(jeff1.5)[3]<-"age"
colnames(jeff1.5)[4]<-"sex "
colnames(jeff1.5)[5]<-"race"
colnames(jeff1.5)[6]<-"relationship "
colnames(jeff1.5)[7]<-"degree "
colnames(jeff1.5)[8]<-"household"
colnames(jeff1.5)[9]<-"neighborhood"
colnames(jeff1.5)[10]<-"school"
colnames(jeff1.5)[11]<-"classroom"
colnames(jeff1.5)[12]<-"workplace"
colnames(jeff1.5)[13]<-"office"

####Descriptive Statistics
###Student

class<-jeff0.6[which(jeff0.6$classroom>0),]
summary(class$classroom)
sd(class$classroom)
school<-jeff0.6[which(jeff0.6$school>0),]
sd(school$school)
summary(school$school)
id<-class$id
classnot<-school[which(!(school$id %in% id)),]

###Work
work<-jeff0.6[which(jeff0.6$workplace>0),]
summary(work$workplace)
office<-jeff0.6[which(jeff0.6$office>0),]

summary(office$office)
###Home
mean(jeff0.6$household)
sd(jeff0.6$household)
summary(jeff0.6$neighborhood)
sd(jeff0.6$neighborhood)

####Network
jeff0.6net <- read.table("F:/Files/Project/Thesis/fred/jefferson-tr0.6-inf-risk-net-measurements.dat", quote="\"")
jeff1.0net <- read.table("F:/Files/Project/Thesis/fred/jefferson-tr1.0-inf-risk-net-measurements.dat", quote="\"")
colnames(jeff0.6net)[1]<-"id"
colnames(jeff0.6net)[2]<-"infectionrisk"
```

```
colnames(jeff0.6net)[3]<-"degree.2"
colnames(jeff0.6net)[4]<-"clusteringcoefficient"
colnames(jeff0.6net)[5]<-"betweenness"
colnames(jeff0.6net)[6]<-"closeness"
colnames(jeff0.6net)[7]<-"eigenvector"
colnames(jeff0.6net)[8]<-"PageRank"
colnames(jeff0.6net)[9]<-"coreness"
colnames(jeff0.6net)[10]<-"knn"

colnames(jeff1.0net)[1]<-"id"
colnames(jeff1.0net)[2]<-"infectionrisk"
colnames(jeff1.0net)[3]<-"degree.2"
colnames(jeff1.0net)[4]<-"clusteringcoefficient"
colnames(jeff1.0net)[5]<-"betweenness"
colnames(jeff1.0net)[6]<-"closeness"
colnames(jeff1.0net)[7]<-"eigenvector"
colnames(jeff1.0net)[8]<-"PageRank"
colnames(jeff1.0net)[9]<-"coreness"
colnames(jeff1.0net)[10]<-"knn"

jeff0.6net<-jeff0.6net[,-2]
jeff1.0net<-jeff1.0net[,-2]

summary(jeff0.6net)
apply(jeff0.6net,2,sd)



###Merge two dataset
jeff6<-merge(jeff0.6,jeff0.6net,by="id")
jeff6<-na.omit(jeff6)

jeff10<-merge(jeff1.0,jeff1.0net,by="id")
jeff10<-na.omit(jeff10)

jeff6$class<-cut(jeff6$infectionrisk,breaks=c(0,0.08,0.4,1),label=c("low","medium","high"))
jeff10$class<-cut(jeff10$infectionrisk,breaks=c(0,0.3,0.8,1),label=c("low","medium","high"))


jeff6$race<-factor(jeff6$race)
jeff6$relationship<-factor(jeff6$relationship)
jeff10$race<-factor(jeff10$race)
jeff10$relationship<-factor(jeff10$relationship)
library(plyr)
jeff6$sex<-revalue(jeff6$sex, c("F"=0, "M"=1,"is"=2))
jeff10$sex<-revalue(jeff10$sex, c("F"=0, "M"=1,"is"=2))

###descriptive graph
###Find each sub population

school<-jeff6[which(jeff6$school>0),]
work<-jeff6[which(jeff6$workplace>0),]
student<-jeff6[which(jeff6$classroom>0),]
nooffice<-jeff6[which(jeff6$office==0),]
nothing<-office[which(office$school==0),]
teacher<-school[which(school$classroom==0),]

schoolsize<-unique(school$school)
schoolsize[26]<-5
summary(schoolsize)

##worksize
worksize<-unique(work$workplace)
table<-table(work$workplace)
a<-as.vector(table)
name<-as.numeric(names(table))
size<-NULL
b<-0
for(i in 1:90 ){
  j<-a[i]/name[i]
  size[b:(b+j)]<-name[i]
  b<-b+j
}

##classroom size
classsize<-unique(student$classroom)
table<-table(student$classroom)
a<-as.vector(table)
name<-as.numeric(names(table))
size<-NULL
```

```
b<-0
for(i in 1:length(classsize) ){
  j<-a[i]/name[i]
  size[b:(b+j)]<-name[i]
  b<-b+j
}
##office size
officesize<-unique(work$office)
table<-table(work$office)
a<-as.vector(table)
name<-as.numeric(names(table))
size<-NULL
b<-0
for(i in 1:length(officesize) ){
  j<-a[i]/name[i]
  size[b:(b+j)]<-name[i]
  b<-b+j
}
mean(size)
sd(size)
####
##neighborhood size
neisize<-unique(jeff6$neighborhood)
table<-table(jeff6$neighborhood)
a<-as.vector(table)
name<-as.numeric(names(table))
size<-NULL
b<-0
for(i in 1:length(neisize) ){
  j<-a[i]/name[i]
  size[b:(b+j)]<-name[i]
  b<-b+j
}
mean(size)
sd(size)
####

##neighborhood size
housesize<-unique(jeff6$household)
table<-table(jeff6$household)
a<-as.vector(table)
name<-as.numeric(names(table))
size<-NULL
b<-0
for(i in 1:length(housesize) ){
  j<-a[i]/name[i]
  size[b:(b+j)]<-name[i]
  b<-b+j
}
mean(size)
sd(size)
####
teacher<-
g6<-jeff6
g6$β<-0.6
g10<-jeff10
g10$β<-1.0
g<-rbind(g6,g10)
m<-ggplot(g6,aes(x=infectionrisk))
m+geom_density(fill=(col="red"),size=0.8,alpha=0.5)+xlab("Infection Risk") +ylab("Density") +ggtitle("Kernel Density Plot for Infection Risk in Mild Outbreak")

n<-ggplot(g10,aes(x=infectionrisk))
n+geom_density(fill=(col="blue"),size=0.8,alpha=0.5)+xlab("Infection Risk") +ylab("Density") +ggtitle("Kernel Density Plot for Infection Risk in Severe Outbreak")
library(psych)
pairs.panels(jeff6.tree[,c(12:17)], smooth=FALSE,density=TRUE,ellipses=FALSE,digits =2,method="spearman", pch = 20, cor=TRUE,hist.col="Grey")


####random forest parameter optimization
library(party)
library(caret)
library(randomForest)

jeff6.tree<-jeff6[,-c(1,2)]
jeff6.tree$sex<-factor(jeff6.tree$sex)
jeff6.tree<-na.omit(jeff6.tree)

jeff10.tree<-jeff10[,-c(1,2)]
jeff10.tree$sex<-factor(jeff10.tree$sex)
```

```
jeff10.tree<-na.omit(jeff10.tree)

rf1<-randomForest(formula=class~.,data=jeff6.tree,ntree=201,replace=T,mtry=5, importance=T,na.action = na.omit)
importance(rf1,scale=F)

rf2<-randomForest(formula=class~.,data=jeff10.tree,ntree=201,replace=T,mtry=5, importance=T,na.action = na.omit)
importance(rf2,scale=F)

rf.rep<-function(dataset,runs=10,mtr=5,ntr=201){
  myVarsum<-matrix(,ncol=4,nrow=19)
  myVar<-matrix(,ncol=19,nrow=4)
  confusions<-matrix(,ncol=3,nrow=3)
  importancematrix<-matrix(,nrow=19,ncol=10)
  errors<-dim(runs)
  for(i in 1:runs){
    set.seed(i+450)
    rf<-randomForest(formula=class~.,data=dataset,ntree=ntr,replace='T,mtry=mtr, importance=T,na.action = na.omit)
    pro<-predict(rf,type="response")
    obs<-dataset$class
    #confusion<-table(obs,pro)
    #confusions<-c(confusions,confusion
    error<-mean(rf$err.rate[,1])
    errors[i] = error
    set.seed(i+450)
    #myVar<-importance(rf,scale=F)[,1:4]
    importancematrix[,i]<-importance(rf,scale=F,type=1)
    #myVarsum=myVarsum+myVar
   }

  #myVarsum=myVarsum/10
  err<-mean(errors)
  #confusions<-confusions/10
  cat('error',errors,'\n', 'average error',err,'\n')
  #cat(myVarsum)
  #print(confusions)
  return(importancematrix)

}

results6<-rf.rep(jeff6.tree)
results.10<-rf.rep(jeff10.tree)
####Plot the random forest results

colnames(rf1)<-c("Degree","Clustering Coefficient","Betweenness","Closeness","Eigenvector centrality","Page Rank")
colnames(rf2)<-c("Degree","Clustering Coefficient","Betweenness","Closeness","Eigenvector centrality","Page Rank")
rf1<-rf1[,c(1,6,3,4,5,2)]
rf2<-rf2[,c(1,6,3,4,5,2)]
library(ggplot2)
rf1_means <- (colMeans(rf1, na.rm = TRUE))
boxplot(rf1,xlab="Network Measurements",ylim=c(0,0.5),ylab="Variable Importance",main="Boxplot of Variable Importance of Network Measurements
in BA ",col="light grey", border="red")
par(new=TRUE)
rf2_means <- (colMeans(rf2, na.rm = TRUE))
boxplot(rf2,col="light grey", border="black",ylim=c(0,0.5),)
legend("topright", title="B  ",c("0.035","0.05"), cex=1,pch=c(15,15),col=c("RED","BLACK"), horiz=FALSE)
```

BIBLIOGRAPHY

1. Levins, R., Eckardt, I., Awerbuch, T., Brinkmann, U., Epstein, P., & Makhoul, N. (1994). The Emergence of New Diseases. American Scientist, 52-60.

2. WHO | Global infectious disease surveillance. (n.d.). Retrieved July 2015, from http://www.who.int/mediacentre/factsheets/fs200/en/

3. Klovdahl, A. S., Potterat, J. J., Woodhouse, D. E., Muth, J. B., Muth, S. Q., & Darrow, W. W. (1994). Social networks and infectious disease: The Colorado Springs study. *Social Science & Medicine, 38*, 79-99. doi:10.1016/0277-9536(94)90302-6

4. Bell, D. C., Atkinson, J. S., & Carlson, J. W. (1999). Centrality measures for disease transmission networks. *Social Networks, 21*, 1-21. doi:10.1016/S0378-8733(98)00010-0

5. Christley, R. M., Pinchbeck, G. L., Bowers, R. G., Clancy, D., French, N. P., Bennett, R., & Turner, J. (2005). Infection in Social Networks: Using Network Analysis to Identify High-Risk Individuals. *American Journal of Epidemiology*, *162*(10), 1024-1031. doi:10.1093/aje/kwi308

6. Lee, C. (2006, May 20). Correlations among centrality measures in complex networks. Retrieved April 18, 2015, from http://arxiv.org/archive/physics

7. Scott, W.J., et al., Treatment of non-small cell lung cancer stage I and stage II: ACCP evi Valente, T. (2008). How Correlated Are Network Centrality Measure. Connect (Tor), 28(1), 16-26.

8. Erdos, P., & Renyi, A. (1960). On the evolution of random graphs. *Publ. Math. Inst. Hungar. Acad. Sci*, *5*, 17-61.

9. Barabási, A. L., & Albert, R. (1999). Emergence of scaling in random networks. *science*, *286*(5439), 509-512..

10. Csardi G, Nepusz T: The igraph software package for complex network research, InterJournal, Complex Systems 1695. 2006. http://igraph.org.

11. Freeman, L. C. (1979). Centrality in social networks: Conceptual clarification. *Social Networks, 1.3*, 215-239. doi:10.1016/0378-8733(78)90021-7.

12. Freeman, L. C. (1977). A set of measuring centrality based on betweenness. *Sociometry ,40*, 35-41. doi:10.2307/3033543

13. Newman, M. E. (2008). The mathematics of networks. *The new palgrave encyclopedia of economics*, *2*(2008), 1-12

14. Page, L., Brin, S., Motwani, R., & Winograd, T. (1998). The PageRank Citation Ranking: Bringing Order to the Web.

15. Watts, D. J., & Strogatz, S. H. (1998). Collective dynamics of' Small-world. *Nature*, *393*, 440.

16. Keeling, M. J., & Rohani, P. (2008). *Modeling infectious diseases in humans and animals*. Princeton University Press.

17. Grefenstette, J. J., Brown, S. T., Rosenfeld, R., DePasse, J., Stone, N. T., Cooley, P. C., ... & Burke, D. S. (2013). FRED (A Framework for Reconstructing Epidemic Dynamics): an open-source software system for modeling infectious diseases and control strategies using census-based populations. *BMC public health*, *13*(1), 940..

18. Lee, B. Y., Brown, S. T., Korch, G. W., Cooley, P. C., Zimmerman, R. K., Wheaton, W. D., ... & Burke, D. S. (2010). A computer simulation of vaccine prioritization, allocation, and rationing during the 2009 H1N1 influenza pandemic. *Vaccine*, *28*(31), 4875-4879.

19. Lee, B. Y., Brown, S. T., Cooley, P., Potter, M. A., Wheaton, W. D., Voorhees, R. E., ... & Burke, D. S. (2010). Simulating school closure strategies to mitigate an influenza epidemic. *Journal of public health management and practice: JPHMP*, *16*(3), 252.

20. Liu, F., Enanoria, W. T., Zipprich, J., Blumberg, S., Harriman, K., Ackley, S. F., ... & Porco, T. C. (2015). The role of vaccination coverage, individual behaviors, and the public health response in the control of measles epidemics: an agent-based simulation for California. *BMC public health*,*15*(1), 447.

21. Lehmann, E. L., & D'Abrera, H. J. (2006). *Nonparametrics: statistical methods based on ranks* (p. 464). New York: Springer.

22. Breiman, L. (2001). Random forests. *Machine learning*, *45*(1), 5-32.

23. Rokach, L., & Maimon, O. (2005). Top-down induction of decision trees classifiers-a survey. *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on*, *35*(4), 476-487.

24 Deng, H., Runger, G., & Tuv, E. (2011). Bias of importance measures for multi-valued attributes and solutions. *Artificial Neural Networks and Machine Learning–ICANN 2011*, 293-300.

25. Strobl, C., Boulesteix, A. L., & Augustin, T. (2007). Unbiased split selection for classification trees based on the Gini index. *Computational Statistics & Data Analysis*, *52*(1), 483-501.

26. Liaw, A., & Wiener, M. (2002). Classification and regression by randomForest. *R news*, *2*(3), 18-22..

27. Genuer, R., Poggi, J. M., & Tuleau-Malot, C. (2010). Variable selection using random forests. *Pattern Recognition Letters*, *31*(14), 2225-2236.

28. van der Laan, M. J. (2006). Statistical inference for variable importance. *The International Journal of Biostatistics*, *2*(1).

29. Litvak, N., Scheinhardt, W. R., & Volkovich, Y. (2007). In-degree and PageRank: Why do they follow similar power laws?. *Internet mathematics*,*4*(2-3), 175-198.

30. Ronqui, J. R. F., & Travieso, G. (2015). Analyzing complex networks through correlations in centrality measurements. *Journal of Statistical Mechanics: Theory and Experiment*, *2015*(5), P05030.

31. Cooley, P., Brown, S., Cajka, J., Chasteen, B., Ganapathi, L., Grefenstette, J., ... & Wagener, D. K. (2011). The role of subway travel in an influenza epidemic: a New York City simulation. *Journal of Urban Health*, *88*(5), 982-995.