NAVAL POSTGRADUATE SCHOOL

MONTEREY, CALIFORNIA

THESIS

ASSESSING THE ROBUSTNESS OF GRAPH STATISTICS FOR NETWORK ANALYSIS UNDER INCOMPLETE INFORMATION

by

Xian Lin Penelope Chia

March 2018

Thesis Advisor:	Samuel Huddleston
Co-Advisor:	Ruriko Yoshida
Second Reader:	David L. Alderson

THIS PAGE INTENTIONALLY LEFT BLANK

REPORT DOCUMENTATION PAGE

Form Approved OMB No. 0704–0188

1. AGENCY USE ONLY ***(Leave blank)***	**2. REPORT DATE** March 2018	**3. REPORT TYPE AND DATES COVERED** Master’s thesis

4. TITLE AND SUBTITLE
ASSESSING THE ROBUSTNESS OF GRAPH STATISTICS FOR NETWORK ANALYSIS UNDER INCOMPLETE INFORMATION

5. FUNDING NUMBERS

6. AUTHOR(S) Xian Lin Penelope Chia

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Naval Postgraduate School
Monterey, CA 93943-5000

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING /MONITORING AGENCY NAME(S) AND ADDRESS(ES)
N/A

10. SPONSORING / MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government. IRB number ____N/A____.

12a. DISTRIBUTION / AVAILABILITY STATEMENT
Approved for public release. Distribution is unlimited.

12b. DISTRIBUTION CODE

13. ABSTRACT (maximum 200 words)

Due to the emergence of powerful global terrorist organizations such as Al Qaeda and ISIS over the last 15 years, social network analysis is increasingly leveraged by the Department of Defense to develop strategies to combat criminal and terrorist organizations. Understanding and correctly classifying networks improves our ability to destroy criminal and terrorist networks because we can leverage existing literature that identifies the optimal strategy for dismantling these networks based on their network structure. However, these strategies typically assume complete information about the underlying network. Due to the limited ability of an analyst to process all of the available data, our inability to detect all members of these networks, and the efforts of criminal organizations to hide their activities and structure, analysts must classify these networks and develop strategies to combat them with missing information. This thesis analyzes the performance of a variety of network statistics in the context of incomplete information by leveraging simulation to remove nodes and edges from networks and evaluating the effect this missing information has on our ability to accurately classify the underlying structure of the network. We provide recommendations to intelligence analysts about which statistics provide the most information, conditions under which it is reasonable to assert a classification, and a framework for the evaluation of network statistics for the purposes of classifying network graphs under incomplete information.

14. SUBJECT TERMS
network, graph, terrorism, intelligence, incomplete information

15. NUMBER OF PAGES
169

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT Unclassified	**18. SECURITY CLASSIFICATION OF THIS PAGE** Unclassified	**19. SECURITY CLASSIFICATION OF ABSTRACT** Unclassified	**20. LIMITATION OF ABSTRACT** UU

THIS PAGE INTENTIONALLY LEFT BLANK

ASSESSING THE ROBUSTNESS OF GRAPH STATISTICS FOR NETWORK ANALYSIS UNDER INCOMPLETE INFORMATION

Xian Lin Penelope Chia
Captain, Singapore Army, Singapore Armed Forces
B.S., University of Western Australia, 2012

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN OPERATIONS RESEARCH

from the

NAVAL POSTGRADUATE SCHOOL
March 2018

Approved by: Samuel Huddleston
Thesis Advisor

Ruriko Yoshida
Co-Advisor

David L. Alderson
Second Reader

Patricia A. Jacobs
Chair, Department of Operations Research

THIS PAGE INTENTIONALLY LEFT BLANK

ABSTRACT

Due to the emergence of powerful global terrorist organizations such as Al Qaeda and ISIS over the last 15 years, social network analysis is increasingly leveraged by the Department of Defense to develop strategies to combat criminal and terrorist organizations. Understanding and correctly classifying networks improves our ability to destroy criminal and terrorist networks because we can leverage existing literature that identifies the optimal strategy for dismantling these networks based on their network structure. However, these strategies typically assume complete information about the underlying network. Due to the limited ability of an analyst to process all of the available data, our inability to detect all members of these networks, and the efforts of criminal organizations to hide their activities and structure, analysts must classify these networks and develop strategies to combat them with missing information. This thesis analyzes the performance of a variety of network statistics in the context of incomplete information by leveraging simulation to remove nodes and edges from networks and evaluating the effect this missing information has on our ability to accurately classify the underlying structure of the network. We provide recommendations to intelligence analysts about which statistics provide the most information, conditions under which it is reasonable to assert a classification, and a framework for the evaluation of network statistics for the purposes of classifying network graphs under incomplete information.

THIS PAGE INTENTIONALLY LEFT BLANK

TABLE OF CONTENTS

LIST OF FIGURES

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF TABLES

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF ACRONYMS AND ABBREVIATIONS

BA	Barabasi-Albert
BA(O)	Barabasi-Albert (Observed) degree distribution
BA(T)	Barabasi-Albert (Theoretical) degree distribution
CART	Classification and Regression Tree
DoD	Department of Defense
DOE	Design of Experiments
ER	Erdos-Renyi
ER(O)	Erdos-Renyi (Observed) degree distribution
ER(T)	Erdos-Renyi (Theoretical) degree distribution
H	Hellinger
ISIS	Islamic State in Iraq and Syria
KL	Kullback-Leibler
NOB	Nearly Orthogonal Balanced
NOLH	Nearly Orthogonal Latin Hypercube
RF	Random Forest
SARs	Suspicious Activity Reports
SW	Small World
SW(O)	Small World (Observed) degree distribution
SW(T)	Small World (Theoretical) degree distribution
WWW	World Wide Web

THIS PAGE INTENTIONALLY LEFT BLANK

EXECUTIVE SUMMARY

A. BACKGROUND AND MOTIVATION

While September 11, 2001 (9/11) is known to many as a watershed for warfare in the modern era, even prior to the 9/11 attacks, Arquilla and Ronfeldt (2001) asserted in their publication *Networks and Netwars* that the nature of modern warfare has evolved to that of a lower intensity war against criminals and terrorists in a network-based, organizational structure instead of full intensity conflicts by state actors as featured in World War I and World War II. Recent military actions throughout the world seem to validate Arquilla and Ronfeldt's assertion. In this environment, the role and value of social network analysis in fighting terrorism (Ressler 2006), cyber crime (Yip 2008), as well as other networked criminal activities (Sparrow 1991) has been steadily increasing.

Network analysis is the science of "using mathematical properties inherent in the graphical structure to seek and uncover differing patterns in the network to determine the conditions under which the networks operate and may best be exploited" (Hopkins 2010). Many researchers agree that characterizing graphs to a high level of accuracy is an essential goal for security forces whose end-state is to curb criminal activity (Cinar et al. 2017). Understanding and correctly classifying the network allows us to effectively disrupt, destabilize or destroy these networks, and graph statistics not only allow us to classify these networks but also measure the effectiveness of destabilization strategies (Hopkins 2010).

One of the biggest problems with current approaches to network analysis in real-world practice is that most applications assume complete information (Carley et al. 2003). As Sparrow (1991) notes, "criminal network data is also inevitably incomplete; i.e., some existent links or nodes will be unobserved or unrecorded. Little research has been done on the effects of incomplete information on apparent structure" (p. 262). Analysts with incomplete information may incorrectly classify the network and hence recommend the use of a strategy meant for a one network type on another, which would not only be ineffective but a waste of resources. Inevitably, most intelligence collection will be based

on incomplete information, but there is little published work on the effects of incomplete information on network analysis (Sparrow 1991).

B. RESEARCH OBJECTIVES

This thesis addresses this gap in the existing literature by considering the following research questions ***in the context of missing information*** **(i.e., hidden edges and vertices):**

1. Which network statistics provide the most predictive power in classifying the network type?
2. What is the effect of changing the (a) edge density, (b) size (number of vertices) of the original network, (c) proportion of information loss, and (d) type of information loss (edges or vertices), (e) network type (Erdos-Renyi [ER], Small World [SW] or Barabasi-Albert [BA]) on the ability to classify a graph type correctly?
3. Can we establish a framework through which we can learn (1) and (2) for any combination of network statistics?

The key objectives of this thesis are twofold: (1) to make a recommendation to intelligence analysts as to the conditions under which it is reasonable to classify networks with incomplete information, and (2) to produce a methodology that will allow researchers to assess the performance and robustness of any graph statistic for network classification as information about the network is lost.

C. METHODOLOGY

The methodology we use to answer these research questions is illustrated in Figure E-1 and summarized as follows:

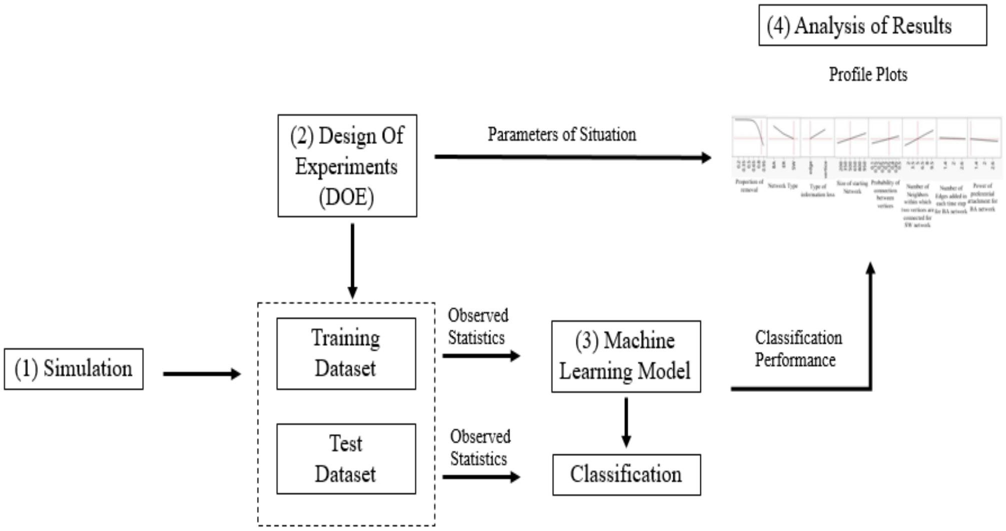
1. <u>Simulation and Descriptive Analysis (Simulation)</u>: We develop a simulation model that randomly removes edges or vertices (i.e., represents information loss) on different sizes and types of graphs and record the resulting behavior of graph statistics as information is lost; this provides the opportunity to observe the effect of information loss on individual statistics.
2. <u>Generation of Data for Machine Learning (Design of Experiments)</u>: We use a space-filling Design of Experiments (DOE) to create training and test datasets that represent networks exhibiting a variety of sizes (i.e.,

number of vertices), edge densities, number of nodes, and other parameters and then use the simulation model to build all of the needed design points.

3. Machine Learning for Network Classification (Machine Learning Model): We use Classification and Regression Trees (CART) (Breiman et al. 1984) and Random Forest (RF) classification models (Breiman 2001) to build machine learning models from the training dataset that describe the contribution of various statistics for accurate classification of networks and test the performance of these models on the out-of-sample test dataset.

4. Analysis of Results (Analysis of Results): We use logistic regression (Agresti 2012) and CART models to analyze the effect of many different situational factors (i.e., simulation parameters) on our ability to accurately distinguish between the three networks types commonly encountered in intelligence analysis applications.

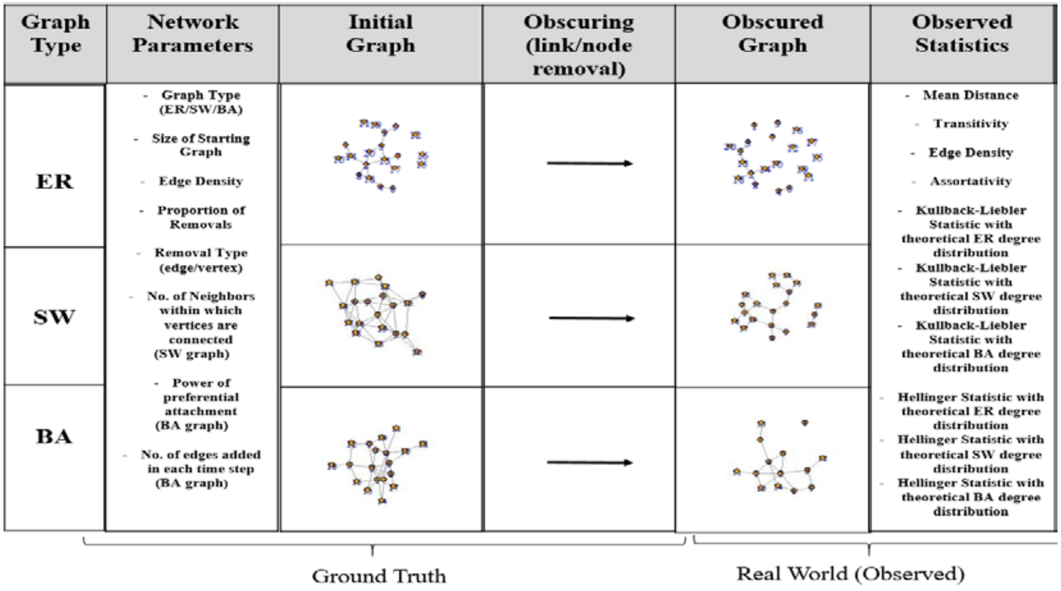
Figure E-1. Overview of Methodology


(4) Analysis of Results
Profile Plots
(2) Design Of Experiments (DOE)
Parameters of Situation
Proportion of removal
Network Type
Type of information loss
Size of starting Network
Probability of connection between vertices
Number of Neighbors within which two vertices are connected for SW network
Number of Edges added in each time step for BA network
Power of preferential attachment for BA network
(1) Simulation
Training Dataset
Test Dataset
Observed Statistics
Observed Statistics
(3) Machine Learning Model
Classification
Classification Performance

This figure illustrates our methodology. In step 1, we develop a simulation model to randomly remove nodes and edges (i.e., simulate information loss) from a variety of networks and record the resulting effect on graph statistics. In step 2, we use a space-filling NOLH DOE to design training and test datasets that represent networks under a variety of conditions and then use the simulation model to build all of the needed design points, all of which are networks in which some information has been lost. In step 3, we use the training data set to build machine learning models for classifying networks based on observed graph statistics. In step 4, we use logistic regression (illustrated with profile plots) to analyze the effect of the many studied factors on our ability to accurately classify the design points in the test dataset based on their observed graph statistics.
Detailed profile plots are in Figure E-3.

Figure E-2 provides an illustration for how the simulation model functions. The simulation model begins by creating an ER, SW, or BA network with the network parameters in the second column available for specification. Then, the simulation model iteratively and randomly removes either links or edges until a specified amount of information has been removed from the network. The result of this process is an obscured network (i.e., a network with missing information), representing the view of the true network that an intelligence analyst might see in the real world. A variety of graph statistics are calculated at each step in the removal process, providing the opportunity to observe how information loss affects these statistics. The key finding of this observational analysis is that no single statistic always distinguishes between the three graph types under the studied conditions of information loss. This finding motivates the use of machine learning models to combine the information from multiple observed statistics to improve our ability to classify with incomplete information.

Figure E-2. Illustration of Simulating Missing Information


Graph Type
Network Parameters
Initial Graph
Obscuring (link/node removal)
Obscured Graph
Observed Statistics
ER
SW
BA
- Graph Type (ER/SW/BA)
- Size of Starting Graph
- Edge Density
- Proportion of Removals
- Removal Type (edge/vertex)
- No. of Neighbors within which vertices are connected (SW graph)
- Power of preferential attachment (BA graph)
- No. of edges added in each time step (BA graph)
- Mean Distance
- Transitivity
- Edge Density
- Assortativity
- Kullback-Liebler Statistic with theoretical ER degree distribution
- Kullback-Liebler Statistic with theoretical SW degree distribution
- Kullback-Liebler Statistic with theoretical BA degree distribution
- Hellinger Statistic with theoretical ER degree distribution
- Hellinger Statistic with theoretical SW degree distribution
- Hellinger Statistic with theoretical BA degree distribution
Ground Truth
Real World (Observed)

This figure outlines the process of simulating missing information. Looking at the figure from left to right, we first generate initial graphs of each of the three graph types based on different network parameters such as size, edge density, etc. Based on a specified proportion of removal and removal type (edge/vertex), the initial graph becomes obscured. Statistics of the obscured graph are observed and recorded, and used later to assert a classification of the graph.

C. RESULTS

This section summarizes the most significant research results obtained in answering each of the three research questions posed in the introduction.

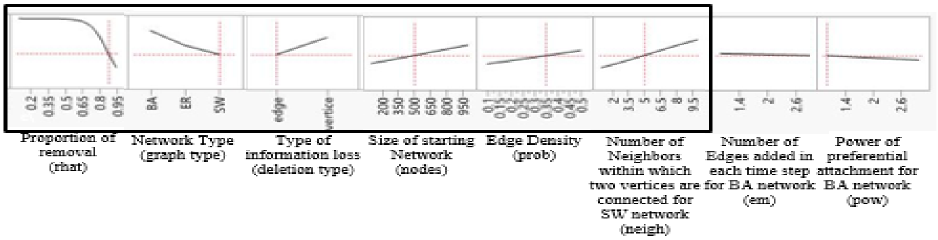
1. Which network statistics provide the most predictive power in classifying the network type?

    - We find that no single statistic is sufficient to distinguish between these three types of graphs under conditions of incomplete information. Rather, even the simplest CART model requires more than one statistic to help to classify the network. Moreover, we observe that ensemble machine learning methods such as RF models provide even more predictive power by combining models leveraging all of the available graph statistics.

2. What is the effect of changing the edge density of the original network (p), size (number of vertices) of the original network (n), proportion of information loss (rhat), and type of information loss (edges vs. vertices), and network type (ER, SW or BA) on the ability to classify a graph type correctly?

    - This analysis indicates that the proportion of removals (i.e., amount of missing information) is the most significant factor in classification performance, followed by type of missing information (edge vs. vertex) and network type. **The most significant finding is that the ability to accurately classify networks declines precipitously once more than 80% of the information about the network is missing** (as illustrated in the left-most panel in the figure). Figure E-3 provides a simple illustration depicting the general shape of the studied effects from the logistic regression analysis, with movement towards the top of the figure indicating improvement in classification performance. Factors within the bold box were statistically significant effects.

Figure E-3. Prediction Profiler for Graph Characteristics in Classifying Networks



This figure outlines the relationship between factor levels and classification performance as found in logistic regression analysis is outlined. Factors which are statistically significant are outlined in the bold box. For those with a positive slope (network size, edge density, number of neighbors within which two vertices are connected), increasing factor levels increases the probability of correct classification. A high proportion of removals, having a SW graph or hidden edges increases the probability that the graph is misclassified.

3. Can we establish a framework by which we can learn (1) and (2) for any combination of network statistics?

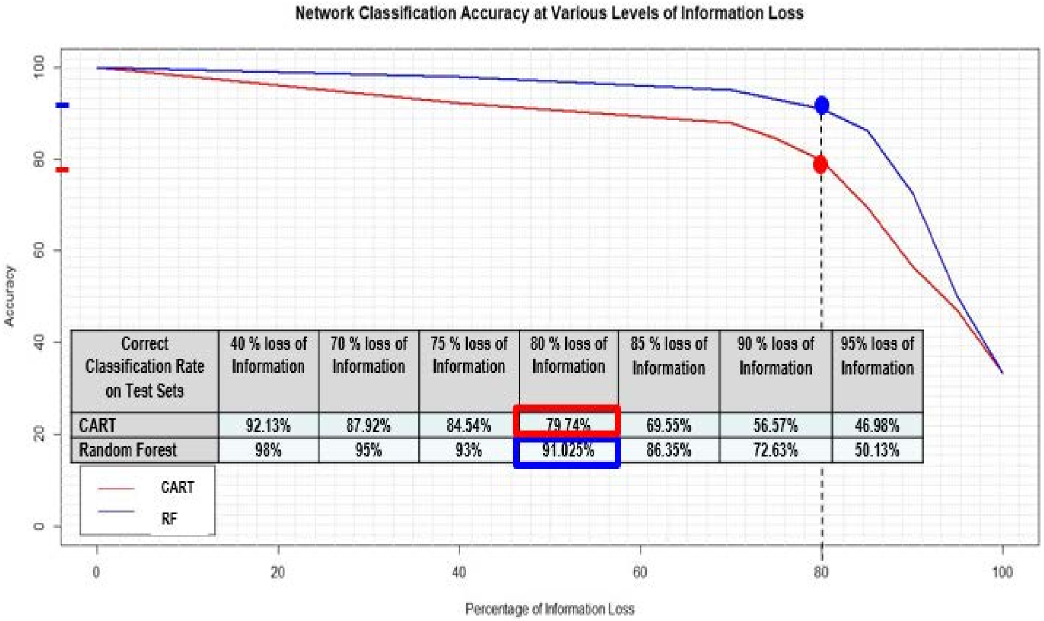
    - This thesis provides a framework for evaluating the contribution of various network statistics on our ability to classify graphs. To date, research in this area has focused on analyzing the performance of individual network statistics for classifying networks. While this thesis finds that an ensemble of statistics can classify a network with high accuracy using RF models, it also establishes a general framework with which any new statistic can be evaluated for its utility in classifying networks (in comparison to its peers) under conditions of incomplete information.

D. APPLICATION FOR INTELLIGENCE ANALYSIS

Figure E-4 provides a detailed view of the effect of information loss on our ability to classify network graphs with both CART and RF models. The amount of information missing has the most significant effect on our ability distinguish between these three network types commonly encountered in intelligence analysis. This graph provides several key insights for applications of network analysis in the intelligence domain. First, only about 20% of the information about a network is needed in order to achieve better than 90% accuracy in network classification. This means that we do not need to spend resources to completely map a network in order to accurately classify it.

Second, the significant performance improvement of the RF ensemble machine learning model in this study over the use of simple thresholds based on individual statistics suggests that this approach should be directly fielded for counter-network applications in the DoD. The classification models developed as part of this thesis, trained on a wide variety of synthetically generated networks, should provide significantly improved classification performance in practice over the current methods used, which use single statistics and assume a complete mapping of the network.

Figure E-4. Classification Accuracy vis-à-vis Information Loss



Correct Classification Rate on Test Sets	40 % loss of Information	70 % loss of Information	75 % loss of Information	80 % loss of Information	85 % loss of Information	90 % loss of Information	95% loss of Information
CART	92.13%	87.92%	84.54%	79.74%	69.55%	56.57%	46.98%
Random Forest	98%	95%	93%	91.025%	86.35%	72.63%	50.13%

This figure shows both the CART (red line) and RF (blue line) have classification accuracy that dips after information loss is beyond 80%. The RF model performs better than the CART model for all levels of information loss, but a key finding is that at 80% information loss, the CART has 79.74% of accuracy while the RF model has 91.025% accuracy.

This research also suggests that, unless we can be reasonably sure that we have sufficient information, we should be very cautious about proposing specific strategies for the dismantling of threat networks based on network classifications conducted on small samples of larger (and mostly unobserved) networks. The widespread practice of asserting a network classification based on a single statistic such as degree distribution calculated on a small observed sample of a much larger (but mostly unobserved) network is unlikely to result in accurate network classification and therefore effective strategies. Rather, a reasonable standard would require that we (1) believe we have observed at least 20% of the network, (2) have reason to believe the network is one of the three types studied in this thesis (or we have replicated this framework for additional network types), and (3) we have developed a classification model whose performance for the desired application is known and validated.

References

Agresti A (2012) *Categorical Data Analysis* 3rd ed. (John Wiley, Hoboken, NJ).

Arquilla J., Ronfeldt D (2001) Networks and netwars: The future of terror, crime, and militancy. MR-1382, RAND Corporation, Santa Monica, CA, http://www.rand.org/pubs/monograph_reports/MR1382/index.html.

Breiman L (2001) Random forests. University of California, Berkeley. January 2001. https://www.stat.berkeley.edu/~breiman/randomforest2001.pdf.

Breiman L, Friedman J, Stone, Charles J, Olshen R. (1984) *Classification and Regression Trees* (Taylor & Francis, New York).

Carley KM, Reminga J, Kamneva N (2003) Destabilizing terrorist networks. *NAACSOS Conference Proceedings* (Carnegie Mellon University Pittsburgh, PA), http://www.casos.cs.cmu.edu/publications/papers/Carley-NAACSOS-03.pdf.

Cinar MS, Genc B, Sever H, Raghavan V V. (2017) Analyzing structure of terrorist networks by using graph metrics. *2017 IEEE International Conference on Big Knowledge* (Hefei, China), 9–16.

Hopkins A (2010) Graph theory, social networks and counter terrorism. Master’s thesis, Department of Mathematics, University of Massachusetts Dartmouth, Dartmouth, MA. https://compmath.files.wordpress.com/2010/05/ahopkins_freports10.pdf.

Ressler S (2006) Social network analysis as an approach to combat terrorism: past, present, and future research. *Homeland Security Affairs* VII(2):1–10, https://www.hsaj.org/articles/171.

Sparrow MK (1991) The application of network analysis to criminal intelligence. *Social Networks* 13:251–274, https://sites.hks.harvard.edu/fs/msparrow/documents--in%20use/Application%20of%20Network%20Analysis%20to%20Criminal%20Intelligence--Social%20Networks--1991.pdf.

Yip M (2008) Social Network Analysis as a tool to study organised cybercrime [Poster]. School of Engineering and Computer Science, University Of Southampton, http://users.ecs.soton.ac.uk/lac/dtc/posters/yip.pdf.

ACKNOWLEDGMENTS

I would like to thank my advisors, LTC Samuel Huddleston and Professor Ruriko Yoshida, for inspiring me with their student-centric values and expertise in the field. This topic would not have been borne without you both. The way both of you encouraged and mentored me in the MORS-Tisdale journey—always demanding excellence but wanting me to enjoy the whole process—is a memory that I will never forget. I would also like to thank Professor Robert Koyak for his guidance in encouraging the spirit of inquiry and the discipline of troubleshooting. To Professor Alderson—who stepped up to take over Professor Koyak's role as second reader when he was unwell— thank you, Sir, for your time and effort. In addition, I would like to thank Professor Paul Sanchez and Professor Thomas Lucas for their guidance in the use of NOLH. To the staff at the Thesis Processing Office and Graduate Writing Center—in particular, Rebecca Pieken and Matthew Norton, respectively—thank you for your help with citations, editing, and formatting. To Erin Field—our beloved Ed Tech—thank you for furnishing information to all students early on thesis deadlines.

To my classmates in Operations Research (OR) and their spouses (I cannot name all but especially): MAJ Nathan Gulosh, MAJ Nicholas Freeman, MAJ Xiao Ren, LCDR Reynaldo Cabana, LCDR Patrick Saluke, LCDR John Renquist, LCDR Charles Deibler, LDCR Li-Chuan-Huan, LCDR Sean Teter, CAPT Tony Van Der Zee, CAPT Alex William Ryan, CAPT SC Lee, CAPT Coleman Strickland, CAPT Andra Deangelis, LT Kyle Belcher, LT Joy Layug, LT Carlos Cervantes, LT Devon Cobbs, LT Kimberly Cull, LT John Tanalega, and LT Rob Syle, thank you for your friendship and support. To my senior CAPT Jason Gay, thank you for your encouragement on the OR journey. To CAPT Ezra Akin, thank you for sharing with me your experience and encouraging me to be the best that I can be.

To my family—thank you for supporting me from 9000 miles away.

Last but not least, to God, thank you for seeing me through yet another challenge in my life! "For with God, nothing is impossible." (Luke 1:37)

THIS PAGE INTENTIONALLY LEFT BLANK

I. INTRODUCTION

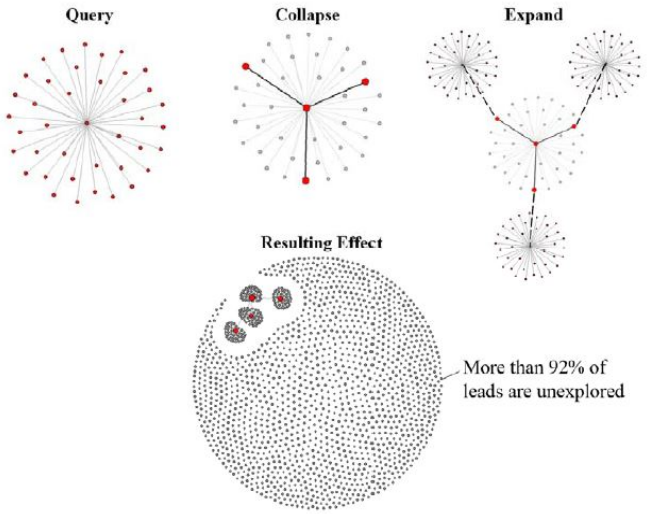
A. BACKGROUND

While September 11, 2001 (9/11) is known to many as a watershed for warfare in the modern era, even prior to the 9/11 attacks, Arquilla and Ronfeldt (2001) asserted in their publication *Networks and Netwars* that the nature of modern warfare has evolved to that of a lower intensity war against criminals and terrorists in a network-based, organizational structure instead of full intensity conflicts by state actors as featured in WWI and WWII. Recent military actions throughout the world seem to validate Arquilla and Ronfeldt's assertion. In this environment, the role and value of social network analysis in fighting terrorism (Ressler 2006), cyber crime (Yip 2008), as well as other networked criminal activities (Sparrow 1991) has been steadily increasing.

Network analysis is the science of "using mathematical properties inherent in the graphical structure to seek and uncover differing patterns in the network to determine the conditions under which the networks operate and may best be exploited" (Hopkins 2010). Many researchers agree that characterizing graphs to a high level of accuracy is an essential goal for security forces whose end-state is to curb criminal activity (Cinar et al. 2017). Understanding and correctly classifying the network allows us to effectively disrupt, destabilize or destroy these networks, and graph statistics not only allow us to classify these networks but also measure the effectiveness of destabilization strategies (Hopkins 2010). For instance, Barabasi-Albert (BA) networks are most vulnerable to targeted attacks on key vertices (nodes) (Hopkins 2010). Graph statistics, such as degree distribution, mean distance and transitivity are often used distinguish one graph type from another (Hopkins 2010).

A key to network analysis is a true and accurate mapping of the network (Huddleston et al. 2016). In the intelligence community, this mapping of the network is often done manually by analysts who build up a network through an iterative process as illustrated in Figure 1. Referencing Figure 1, this process involves studying the relationships of initial subject of interest ("Query"), selecting nodes and relationships that will be explored for further study (“Collapse”), and further study of these selected entities (“Expand”) (Huddleston et al. 2016). In real world practice, due to the limited time and resources of any analyst, this focus on only a small subset of the entities and relationships in the data inevitably results in the mapping of only a very small subset of the actual network. In the example given in Figure 1, when an analyst collapses focus onto only three of 40 possible nodes in the initial query, they immediately eliminate consideration of over 90% of the network represented in the available data.

Figure 1. How a Network Was Mapped Out with Information on an Initial Subject of Interest. Source: Huddleston et al. (2016).



This figure illustrates how intelligence analysts will never have complete information on networks. With an initial subject of interest, they “Query” all his links and relationships. They “Collapse” the network, narrowing it down to three to four nodes to continue their mapping as they are unable to explore all leads based on the resources they have. Next, they “Expand” and find out the “friends of friends” of the initial subject of interest to map the rest of the network. This process is done iteratively. However, more than 92% of the leads are unexplored due to analyst’s inability to follow up on all leads (Huddleston et al. 2016).

One of the biggest problems with current approaches in interpreting network statistics is that most of them assume complete information (Carley et al. 2003). According to Sparrow (1991), “criminal network data is also inevitably incomplete; i.e., some existent links or nodes will be unobserved or unrecorded. Little research has been done on the effects of incomplete information on apparent structure” (p. 262).

As illustrated above, intelligence information is often incomplete and much of the network is undiscovered (Huddleston et al. 2016). Analysts with incomplete information may incorrectly classify the network and hence recommend the use of a strategy meant for a one network type on another, which would not only be ineffective but also a waste of resources. Most intelligence collection will inevitably be based on incomplete information, but there is scarce work on the effects on incomplete information on network analysis (Sparrow 1991).

B. OBJECTIVES AND APPROACH

The key question for this thesis is: How robust to missing information are inferences about networks based on graph statistics? The fundamental research questions that this thesis will seek to answer is as follows. ***In the context of missing information*** **(i.e., an incomplete mapping):**

(1) Which network statistics provide the most predictive power in classifying the network type?

(2) What is the effect of changing the following parameters on the ability to classify a graph type correctly?

- edge density
- size (number of vertices) of the original network
- proportion of information loss, and
- type of information loss (edges or vertices), and
- network type (Erdos-Renyi [ER], Small World [SW], or Barabasi-Albert [BA])

(3) Can we establish a framework through which we can learn (1) and (2) for any network statistic?

We address these research questions through the following methodological steps:

- <u>Simulation and Descriptive Analysis (Simulation)</u>: We develop a simulation model that randomly removes edges or vertices (i.e., represents information loss) on different sizes and types of graphs and record the resulting behavior of graph statistics as information is lost; this provides the opportunity to observe the effect of information loss on individual statistics.

- Generation of Data for Machine Learning (Design of Experiments): We use a space-filling Design of Experiments (DOE) to create training and test datasets that represent networks exhibiting a variety of sizes (i.e., number of vertices), edge densities, number of nodes, and other parameters and then use the simulation model to build all of the needed design points.

- Machine Learning for Network Classification (Machine Learning Model): We use Classification and Regression Trees (CART) (Breiman et al. 1984) and Random Forest (RF) classification models (Breiman 2001) to build machine learning models from the training dataset that describe the contribution of various statistics for accurate classification of networks and test the performance of these models on the out-of-sample test dataset.

- Analysis of Results (Analysis of Results): We use logistic regression and CART models to analyze the effect of many different situational factors (i.e., simulation parameters) on our ability to accurately distinguish between the three networks types commonly encountered in intelligence analysis applications.

This research provides two significant and immediately applicable results: (1) it provides recommendations to intelligence analysts as to the threshold with which to trust different statistics under the condition of incomplete information, and (2) it provides a framework for evaluating the utility of network statistic for network classification.

C. SCOPE/LIMITATIONS

This thesis focuses on evaluating the performance of graph statistics for network classification, leaving consideration of node centrality measures under the loss of information for future work. We remove edges and vertices randomly to investigate the individual effects of their removal, recognizing that in reality both edges and vertices may be missing and biases may be systematic rather than random. We limit analysis and classification of networks to ER, SW and BA networks, as they largely characterize criminal and terrorist networks (elaborated in Chapter III) and are easily produced using open source software.

D. STRUCTURE OF THESIS

The remainder of this thesis is organized as follows. Chapter II (Literature Review) reviews the work done so far involving network statistics and shows a lack of investigative work done on network statistics taking into account a loss of information. Chapter III (Model Formulation) introduces key equations and outlines the descriptive and predictive portions of the model including assumptions and methodology. Chapter IV (Analysis of Results) outlines the key results and discusses their validity, while Chapter V (Conclusion) highlights key findings and recommended future work.

THIS PAGE INTENTIONALLY LEFT BLANK

II. LITERATURE REVIEW

A. THE IMPORTANCE OF NETWORK ANALYSIS

Until recent years, a large proportion of network classification was done in the field of chemical and biological data (Zhu et al. 2011). Even though it was acknowledged that there was potential usefulness of network classification in social network analysis, (Zhu et al. 2011) asserted that they were not aware of "focused study on this problem."

The events of 9/11 drove home a strong point about the dangers of networked criminal organizations, and there was increasing amounts of work done to study network properties and their meaning for counter-network operations (Ressler 2006). Many researchers now agree that characterizing networks to a high level of accuracy is an essential goal for security forces whose end-state is to curb criminal activity (Cinar et al. 2017).

B. WHY GRAPH STATISTICS ARE IMPORTANT TO NETWORK CLASSIFICATION

Network classification lends important insight as to how best to conduct counter-network operations (Hopkins 2010). For instance, Barabasi-Albert (BA) scale-free networks "are vulnerable to targeted attacks on highly connected nodes" (Faloutsos 2008), whereas Small-World (SW) networks are vulnerable to attacks on edges between key clusters or the key clusters themselves (Hopkins 2010). Network statistics like degree distribution, mean distance and transitivity are widely used to define and distinguish these networks (Hopkins 2010). For example, BA networks have a scale-free degree distribution, while ER networks have a Poisson degree distribution, and SW networks have a binomial degree distribution (Costa et al. 2005). In addition, SW networks are known to have a high transitivity and low mean distance (Xu and Chen 2008). Covert terrorist networks such as Al-Qaeda have been classified as SW networks, whereas dark networks or the World Wide Web (WWW) are BA networks (Xu and Chen 2008). By correctly classifying these networks, counter-network operations can be effectively planned in ways that minimize the use of resources by focusing on the nodes whose removal will be most damaging to the overall health of the network.

Besides classifying networks, graph statistics are also useful in that they provide some insight in the environment in which these networks operate, as well as assist in measuring the effectiveness of destabilization strategies (Carley et al. 2003). For example, networks with high clustering coefficients and low mean distance are highly efficient and can connect with other members through few mediators (Hopkins 2010). Networks with positive assortativity indicate that members are connected to others with the same characteristics (Hopkins 2010). The Al-Qaeda network is known to have positive assortativity, with "high-degree nodes…cluster[ing] together as core groups, a phenomenon evident in the ... network in which bin Laden and his closest cohorts form the core of the network and issue commands to other parts of the network" (Xu and Chen 2008). Furthermore, a high clustering coefficient is indicative that the mechanism for recruitment of new members is through a mutual friend, or transitive linking (Friemel 2011), while a high edge density would mean that the network is not easily fragmentable (Hopkins 2010). A change in the edge density, mean distance or degree distribution could indicate a measure of effectiveness in the destabilization strategy for these networks (Carley et al. 2003).

C. HOW NETWORKS ARE MAPPED

Before network analysis can commence, however, work has to be done in order to map out the network. This begins with initial subjects of interest (vertices), such as a known member of a terrorist organization (Huddleston et al. 2016). After studying connections (edges) that the subjects of interest have, more subjects (vertices) are added and a more comprehensive picture of the network is formed (Satell 2013). Not all the leads on new subjects can be explored, so a subset is chosen and a further investigation on their connections is done (Huddleston et al. 2016). Because intelligence analysts are unable to explore all leads and have to narrow their focus on some, there is almost always a problem of incomplete information. This lack of an ability to map the entire network is also illustrated in a recent study on counter-threat finance intelligence that noted the analysis was constrained to only the financial data available from Suspicious Activity Reports (SARs) rather than the full records of financial transactions involving the parties of interest (Huddleston et al. 2016). In the intelligence field, new vertices (people) are mapped in the network only when the corresponding link (relationship) is found in some way through surveillance activities or available data sources.

D. THE PROBLEM OF INCOMPLETE INFORMATION

While the importance of network analysis and graph statistics in the use of classification of networks has gained recognition and importance in recent years, ***most of these studies implicitly assume complete information***. Carley and Kim (2008) looked into interpretation of graph statistics in comparison to random graphs and approximate a distribution of these statistics. Recently, Cinar et al. (2017) computed network statistics for a number of terrorist networks such as the 9/11 Hijackers associates, the Jemaah Islamiyah Koschade and the Islamic State in Iraq and Syria (ISIS), to give an indication on various operating conditions of these networks such as density, mean distance and closeness.

In evaluating the effectiveness of destabilization strategies for terrorist networks, Carley et al. (2003) recognized that one of the most crucial problems is that in spite of the large amounts of information on such networks, most of such information is often incomplete. A large proportion of leads are unexplored (as shown in Figure 1) due to the limited ability of any analyst to process all the available data (Huddleston et al. 2016). In addition, criminals tend to make a concerted effort to erase all traces of illicit relationships and keep a low profile to avoid detection (Hopkins 2010).

While this shows that information about edges in networks is highly likely to be flawed, the same can be said about information about vertices. Determination of vertex (node) centrality (i.e., the most important person in a network), for criminal networks may be the result of who is known most completely rather than who is the most important person structurally in the actual but unobserved network (Sparrow 1991). Hence, analysts may fall into the trap of focusing on the person they have the most information on even if he may not be the ring-leader of the network (Sparrow 1991). Similarly, for classifying networks, analysts with incomplete information may misclassify a network and as a result recommend the use of a strategy meant for a different type of network, which would not only be ineffective but also a waste of resources. Most intelligence collection is inevitably based on incomplete information, but there is little work on the effects on incomplete information on network analysis (Sparrow 1991).

Thus, there is a need to develop tools to evaluate terrorist network destabilization strategies in the context of incomplete information. Sparrow (1991) had similar thoughts on criminal networked organizations, asserting “little research has been done on the effects of incomplete information on apparent structure” (p. 262). Sparrow also acknowledged that while there had been some research on the issues involved in statistical inference from networks with incomplete information, (e.g., the 1981 study by Friedkin on the effect of sampling of random edges on the structural properties of networks), he argues that biases in the real world brought about by investigative procedures do not follow a random pattern.

III. MODEL FORMULATION

A. INTRODUCTION TO NETWORK ANALYSIS AND GRAPH STATISTICS

In this section, we present definitions of terms and variables as well as explain why they are of interest to social network analysts.

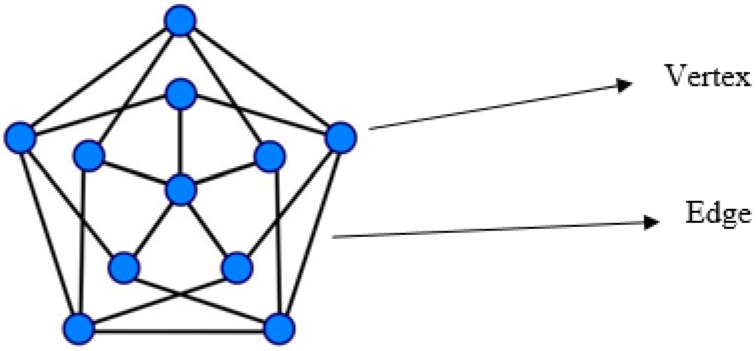
1. General Terms

We adopt definitions of terms and variables from Rodrigue and Ducruet (2017) and explain why they are of interest to social network analysts.

(1) Vertex (Node)

A vertex is “a terminal point or an intersection point in a graph” (Rodrigue and Ducruet 2017). In Figure 2, these are represented by blue circles. In the context of a social network, vertices represent people.

Figure 2. Network with Vertices (circles in blue) and Edges (black links).
Source: (Kell 2006).



(2) Edge (Link or Arc)

An edge is “a link between two vertices” (Rodrigue and Ducruet 2017). In Figure 2, these are represented by black lines. For instance, in the context of a social network, an edge represents a relationship between people.

(3) Graph

A graph is a “collection of vertices (nodes) connected by edges” (Rodrigue and Ducruet 2017). In Figure 2, the graph is the entire diagram containing both nodes and edges.

(4) Network

A network is a graph with information (attributes) (Ahuja et al. 1993). Social networks are comprised of people (nodes), relationship (edges), and specific attribute information such as type of relationship, age of person, etc.

Note that “node” and “vertex”; “edge” and “link” are synonyms. For this thesis, we use the terms “vertex,” “edge,” and “network” throughout. We use the term “graph” when it comes to computational and mathematical aspects, and the term “network” with regard to (intelligence) application aspects.

2. Network Types

Social networks are typically best approximated by one of three network types (Barabasi 2015):

- Erdos-Renyi (ER) random network
- Small-World (SW) network
- Barabasi-Albert (BA) network

Each of these networks types is described in depth in the following sections.

a. Erdos-Renyi (ER) Random Network

The Erdos-Renyi network is a random network that starts with a number of disconnected vertices and is constructed, while avoiding self-connections, by adding edges randomly with a given probability *p* (Costa et al. 2005). ER networks are characterized by a low mean average distance, low clustering coefficient, and a Poisson degree distribution (Hopkins 2010).

b. Small World (SW) Network

The small-world model originated with the "observation that most real-world graphs seem to have a low average distance between nodes…but have high clustering coefficients" (Watts and Strogatz 1998). This characterizes many real-world networks, including social networks, where only a small number of friends separate two people from knowing each other. A common saying to describe this is "6 degrees of separation [6 acquaintances] separate any two people." The small world model starts with a D-dimensional square lattice and connections are re-wired to reduce the overall mean distance (Watts and Strogatz 1998). The degree distribution of a SW network is binomial (Cinar et al. 2017). According to Alderson (2008), "the small-world model has been used to represent many types of social networks, including collaboration … trust networks … and community structure" (p.1053). Given that the SW network is characterized by clusters with weak ties, the SW network is vulnerable to disruption and fragmentation through attacks on key clusters or edges between them (Hopkins 2010).

c. Barabasi-Albert (BA) Network

Barabasi-Albert (BA) networks are also known as scale-free networks because their degree distribution follows a power law; hence, a large proportion of vertices have a small number of connections, while a small proportion of vertices have a large number of connections (Costa et al. 2005). In addition, the BA network is generated with preferential attachment; that is vertices with more existing edges are more likely to have additional edge(s) added in each time step (Faloutsos 2008). The BA network is resistant to random losses, yet in the context of counter-network operations vulnerable to targeted attacks on highly connected vertices (Faloutsos 2008).

According to Faloutsos (2008), whilst the BA model characterizes some real-world networks with its preferential attachment model, one must be careful with regard to its application to real world networks. First, the exponent of the power-law of the degree distribution is 3, but there is a proportion of real world networks do not have this property (Faloutsos 2008). Next, he suggests that the BA model has a constant average degree, "however, the average degree of some graphs … actually increases over time according to a Densification Power Law.” Nevertheless, it is still useful for real world networks with its properties.

3. Graph Statistics

The following network statistics are often used to describe and classify the topology of a network:

- edge density
- mean distance
- transitivity
- assortativity
- degree distribution
- Kullback-Leibler (KL) divergence
- Hellinger distance statistic

Each of these statistics is discussed in detail in the following sections.

a. Edge Density: The Probability of Connection between Vertices

The edge density, or the probability of connection between vertices of network is defined as the ratio of the number of edges to the number of possible edges (Rodrigue and Ducruet 2017). It is defined as follows:

$$Edge\ Density\ = \frac{Number\ of\ edges}{Number\ of\ possible\ edges}\ , \tag{1}$$

where the number of possible edges is equal to $\binom{N}{2}$**,** where N is the number of vertices. Networks with low density are easily fragmented, whereas networks with high density are resistant to fragmentation (Cinar et al. 2017).

b. Mean Distance

The mean distance of a network is the average path length of a network between nodes (Rodrigue and Ducruet 2017). This statistic indicates how far apart two nodes are on average (Costa et al. 2005). The mean distance of a network L_G is defined as:

$$L_G = \frac{1}{N(N-1)} \sum_{i \neq j} d(V_i, V_j) \ . \tag{2}$$

The mean distance of a network is computed using a summation of all the distances between vertices, (where $d(V_i, V_j)$ is the distance between vertex *i* and vertex *j*, ignoring self-connections) normalized by the total number of vertices (Costa et al. 2005). For this thesis, this statistic is computed only for connected portions of the network and unconnected portions of the network are ignored, mimicking the scenario in the intelligence analysis of networks in which unconnected entities and/or communities are not mapped to studied networks because their relationship to the network is unknown. In attempting to disrupt terrorist networks, one of the goals is often to increase the mean distance to make operations more difficult for the network. ER and SW networks are known to have low mean distances (Hopkins 2010).

c. Transitivity

The transitivity, or clustering coefficient *C* of a graph, measures the probability that the adjacent vertices of a vertex is connected (Rodrigue and Ducruet 2017), i.e., it measures the extent to which connections are defined by mutual friends, or the probability that connected triangles appear in a given network (Hopkins 2010). It is evaluated by dividing three times the number of fully connected triples (i.e., triangles in the graph) with the number of triples (Costa et al. 2005), specifically

$$C = \frac{3 \ * \ Number \ of \ fully \ connected \ triples}{Number \ of \ triples} \ . \tag{3}$$

This is also a way to gauge the connectedness of a network. Terrorist networks with high transitivity are of concern to intelligence analysts as this signifies a highly connected network. It also indicates that the recruitment mechanism for this network is through mutual friends (Ebel et al. 2003).

d. Assortativity

On assortativity, Hopkins (2010) cites Xu and Chen (2008), “in positively assortative networks, high-degree nodes tend to cluster together as core groups, a phenomenon evident in the … network in which bin Laden and his closest cohorts form the core of the network and issue commands to other parts of the network.”

Assortativity indicates a preference or an inclination for a vertex in a network to attach itself to other vertices which have similarities (such as nodes with high degree connecting to other vertices) (Hopkins 2010). Assortativity (*A*) is defined as:

$$A = \frac{\sum_{i,j} ij(e_{ij} - q_i q_j)}{\sigma_q^2} \quad . \tag{4}$$

A is computed by taking the summation of the difference between the joint probability distribution e_{ij} of the remaining degrees of vertex *i* and *j* and the product of the distribution of remaining degrees of vertex *i* (q_i) and vertex *j* (q_j), over all possible combinations of vertices *i* and *j*, divided by σ_q^2, the squared of the standard deviation in distribution of the remaining degrees (Noldus and Mieghem 2014). Social networks tend to have positive assortativity (Hopkins 2010). SW and BA networks are also known to have an assortativity value of zero (Costa et al. 2005).

e. Degree Distribution

The degree of a vertex represents the number of edges (relationships) that a vertex has with other vertices (Cinar et al. 2017). The degree distribution is a vector whose first element specifies the proportion of nodes with zero connections; second element specifies the proportion of vertices with one connection; etc.(Costa et al. 2005). Degree distributions give an indication on the proportion of people in a social network who are highly connected; as well as the proportion of people with few or no connections (Hopkins 2010).

This statistic is one of the distinguishing features between the ER, SW and BA networks: the degree distribution of a BA network is scale-free; the degree distribution of an ER network is Poisson, and the degree distributions of a SW network is binomial (Hopkins 2010). The mathematical definitions follow.

The theoretical degree distribution of an **Erdos-Renyi (ER) network**, where $P(k)$ is the probability a randomly selected vertex has degree *k*, is as follows:

$$P(k) = \frac{\mathrm{e}^{-<k>} < k >^{k}}{k!} \quad , \tag{5}$$

where the average vertex degree for the network, $< k >$ = $p(N-1)$,with *p*, the probability of connection between two random vertices and *N*, the number of all vertices in the graph (Costa et al. 2005).

The theoretical degree distribution of a **Small World (SW) network**, where $P(k)$ is the probability a randomly selected vertex has degree *k*, is as follows:

$$P(k) = \sum_{i=1}^{\min(k-\kappa,\kappa)} \binom{\kappa}{i} (1-p)^{i} p^{\kappa-i} \frac{(p\kappa)^{k-\kappa-i}}{(k-\kappa-i)!} e^{-p\kappa} \quad , \tag{6}$$

where κ represents the number of neighbors of each vertex in the initial regular network (Costa et al. 2005).

The theoretical degree distribution of a **Barabasi-Albert (BA) network**, where $P(k)$ is the probability that a randomly selected vertex has degree *k*, is as follows (Costa et al. 2005):

$$P(k) \sim k^{-3} \; . \tag{7}$$

f. *Kullback-Leibler (KL) Divergence*

The Kullback-Leibler divergence measures the distance between two probability distributions (Kullback and Leibler 1951). For two discrete probability distributions *P* and *Q*, the Kullback-Leibler divergence is defined as the expected logarithmic difference between probability distributions *P* and *Q* (Kullback and Leibler 1951):

$$D_{KL}(P \| Q) = -\sum_i P(i) \log \frac{Q(i)}{P(i)} \quad . \tag{8}$$

In this computation, note that if any element of the probability distribution $P(i)$, where $P(i)$ is the probability for a state *i* in the distribution *P*, is equal to 0, this implies that $Q(i) = 0$, where $Q(i)$ is the probability for a state *i* in the distribution *Q*, and no increment will be made to the current sum total in the Kullback-Leibler divergence statistic.

g. Hellinger Distance Statistic

The Hellinger distance statistic (Hellinger 1909) is used to quantify similarity between two probability distributions. With two discrete probability distributions *P* and *Q*, the Hellinger Distance Statistic is defined as follows:

$$1 - \sum_{j=1}^{n} \sqrt{P_j Q_j} \quad , \tag{9}$$

where P_j is the probability for a state *j* in the probability distribution *P*, and Q_j is the probability for a state *j* in the distribution *Q*. This distance is used to quantify similarity between two probability distributions (Hellinger 1909).

B. METHODOLOGY OVERVIEW

The methodology we use to answer these research questions is illustrated in Figure 3 and summarized as follows:

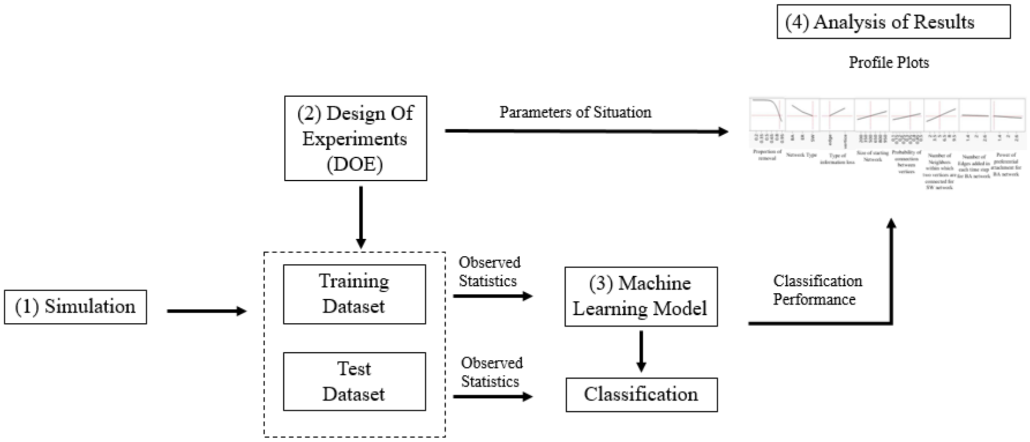
- <u>Simulation and Descriptive Analysis (Simulation)</u>: We develop a simulation model that randomly removes edges or vertices (i.e., represents information loss) on different sizes and types of graphs and record the resulting behavior of graph statistics as information is lost; this provides the opportunity to observe the effect of information loss on individual statistics.
- <u>Generation of Data for Machine Learning (Design of Experiments)</u>: We use a space-filling Design of Experiments (DOE) to create training and test datasets that represent networks exhibiting a variety of sizes (i.e., number of vertices), edge densities, number of nodes, and other

parameters and then use the simulation model to build all of the needed design points.

- Machine Learning for Network Classification (Machine Learning Model): We use Classification and Regression Trees (CART) (Breiman et al. 1984) and Random Forest (RF) classification models (Breiman 2001) to build machine learning models from the training dataset that describe the contribution of various statistics for accurate classification of networks and test the performance of these models on the out-of-sample test dataset.

- Analysis of Results (Analysis of Results): We use logistic regression and CART models to analyze the effect of many different situational factors (i.e., simulation parameters) on our ability to accurately distinguish between the three networks types commonly encountered in intelligence analysis applications.

Figure 3. Overview of Methodology

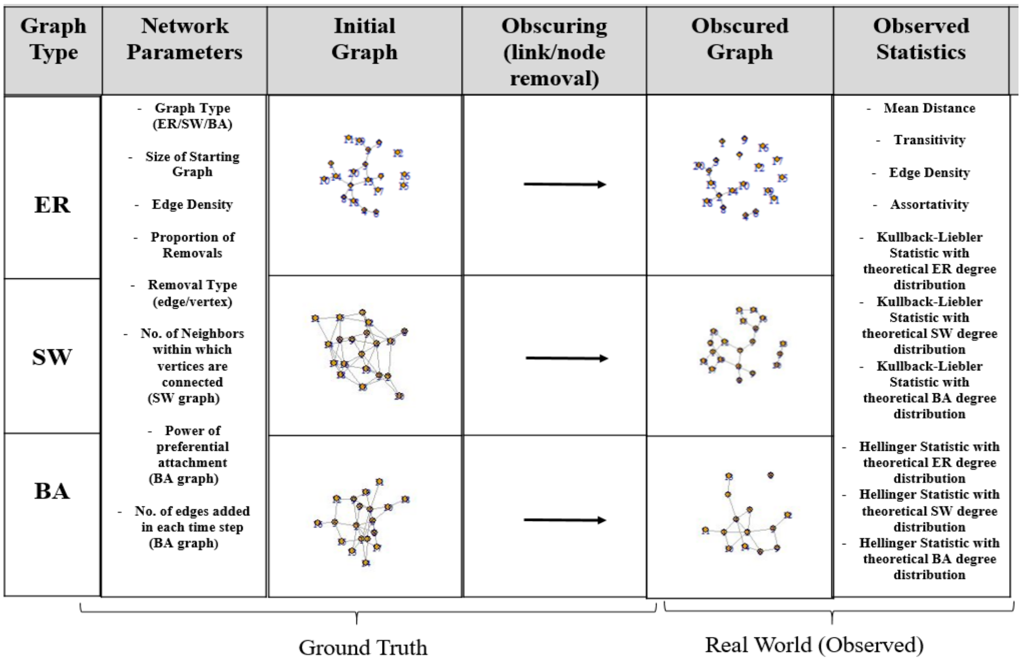

(4) Analysis of Results
Profile Plots
(2) Design Of Experiments (DOE)
Parameters of Situation
Proportion of removal
Network Type
Type of information loss
Size of starting Network
Probability of connection between vertices
Number of Neighbors within which two vertices are connected for SW network
Number of Edges added in each time step for BA network
Power of preferential attachment for BA network
(1) Simulation
Training Dataset
Test Dataset
Observed Statistics
Observed Statistics
(3) Machine Learning Model
Classification
Classification Performance

This figure illustrates our methodology. In step 1, we develop a simulation model to randomly remove nodes and edges (i.e., simulate information loss) from a variety of networks and record the resulting effect on graph statistics. In step 2, we use a space-filling NOLH DOE to design training and test datasets that represent networks under a variety of conditions and then use the simulation model to build all of the needed design points, all of which are networks in which some information has been lost. In step 3, we use the training data set to build machine learning models for classifying networks based on observed graph statistics. In step 4, we use logistic regression (illustrated with profile plots) to analyze the effect of the many studied factors on our ability to accurately classify the design points in the test dataset based on their observed graph statistics. See Figure 12 for detailed profile plots.

C. DESCRIPTIVE ANALYSIS OF STATISTICS UNDER A LOSS OF INFORMATION USING SIMULATION

This thesis considers the behavior of the various statistics that are typically used in classifying networks and/or of interest in social network analysis under conditions of incomplete information. In order to establish the behavior of these statistics under such conditions, we simulate different levels (0-80%) of missing information by randomly removing an increasing proportion of nodes and edges and computing network statistics at each point of removal. We then compare the network statistics among the three network types and check if they converged or remained distinct. We use distance measures to compare the observed data with theoretical degree distributions of the three network types in order to find out if there is any level of incomplete information at which the observed network resembles another network type. Further elaboration is provided in Figure 4.

Figure 4. Simulation of Information Loss



Graph Type	Network Parameters	Initial Graph	Obscuring (link/node removal)	Obscured Graph	Observed Statistics
ER	- Graph Type (ER/SW/BA) - Size of Starting Graph - Edge Density - Proportion of Removals - Removal Type (edge/vertex) - No. of Neighbors within which vertices are connected (SW graph) - Power of preferential attachment (BA graph) - No. of edges added in each time step (BA graph)		→		- Mean Distance - Transitivity - Edge Density - Assortativity - Kullback-Liebler Statistic with theoretical ER degree distribution - Kullback-Liebler Statistic with theoretical SW degree distribution - Kullback-Liebler Statistic with theoretical BA degree distribution - Hellinger Statistic with theoretical ER degree distribution - Hellinger Statistic with theoretical SW degree distribution - Hellinger Statistic with theoretical BA degree distribution
SW			→		
BA			→		

As the figure shows, the process of simulating missing information is outlined. Looking at the figure from left to right, we first generate initial graphs of each of the three graph types based on different network parameters such as size, edge density, etc. Based on a specified proportion of removal and removal type (edge/vertex), the initial graph becomes obscured. Statistics of the obscured graph are observed and recorded, and used later to assert a classification of the graph.

1. Graph Generation

We use the R package `igraph` (Csardi and Nepusz T 2006) to generate and manipulate networks in our study.

We use the function `erdos.renyi.game()` to generate ER graphs. This function takes the following arguments as input: number of nodes and, edge density. It produces output in the form of a graph object with characteristics as specified in the input arguments.

We use the function `sample_smallworld()` to generate SW graphs. This function takes the following arguments as input: number of dimensions, number of nodes, edge density, whether loop edges are allowed in generated graph (default = FALSE) and whether multiple edges are allowed in generated graph (default = FALSE). It produces output in the form of a graph object with characteristics as specified in the input arguments.

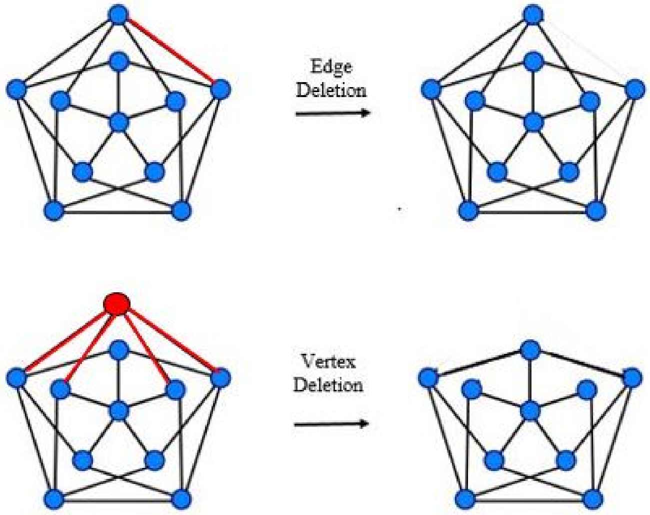
We use the function `sample_pa()` to generate BA graphs. This function takes the following arguments as input: number of nodes, power of preferential attachment (default = 1.2), number of edges added in each time step (default = NULL), distribution of edges added in each time step (default = NULL), numeric vector of number of edges added in each time step (default = FALSE), "attractiveness" of vertices with no adjacent edges (default = 1), whether to create a directed graph (default = FALSE), algorithm to use for graph generation, and starting graph (for the preferential attachment model). It produces output in the form of a graph object with characteristics as specified in the input arguments.

2. Removal of Edges and Vertices

In order to monitor the robustness of statistics for classifying graphs under incomplete information, we delete edges and vertices from the starting networks as specified in Table 1. We set the proportion of removals from 0–80% to investigate how the statistics behave under increasing removals. Referencing the edge or vertex list, depending on whichever is specified by the user, we iteratively and randomly delete an increasing number of unique edges or vertices until we obtain a specified proportion of removals.

As implemented in *R*'s `igraph` package (Csardi and Nepusz T 2006), a deletion of an edge results in one less edge, while a deletion of a vertex results in the deletion of all corresponding edges connected to that vertex (see Figure 5). This is a fair approximation to real-life effects of incomplete information. If an edge (relationship) is not known, it will not exist. If a vertex (person) is not known, his/her corresponding edges (relationships with others) are also not known. Similarly, when studying the effects of destroying networks, if a vertex (e.g., a person in the terrorist network) is destroyed, the relationships with that vertex cease to matter and hence cease to exist, while if an edge is destroyed (e.g., a relationship), that edge will no longer exist.

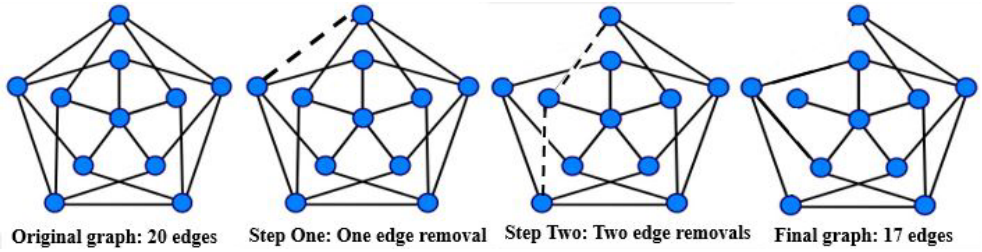
Figure 5. Edge and Vertex Deletion. Adapted from (Kell 2006).



This figure outlines the process of edge and vertex deletion. In particular, an edge deletion results in that particular edge being removed or hidden, and represents a relationship that we do not know about. A vertex deletion results in the vertex and all its links being removed or hidden, and represents a person, and by extension, his relationships, that we do not know about.

In simulation of removals, we step through the deletion process n times in order to achieve the user specified percentage of removals. For example, referencing Figure 6, the user specified percentage of removals was 15%. With an initial edge list containing 20 edges, over $n = 2$ steps, a total of three edges were deleted. After each step, the graph is stored temporarily and the graph statistics (mean distance, edge density, assortativity, degree distribution, transitivity) are archived in a list (and eventually plotted).

Figure 6. Sample Sequence of Edge Deletions. Adapted from (Kell 2006).



This figure illustrates the process with which edges are removed. In the first step, one edge is removed. In the second step, two edges are removed. This process continues until the specified proportion of removals is achieved.

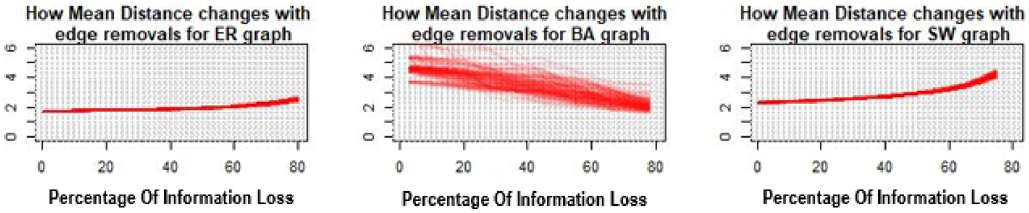
3. Graph Statistics Analysis

In this section, we will discuss how (1) we plot all the network statistics from the three networks on one panel to determine if they converge and how (2) we compute distance measures between the observed and theoretical degree distributions.

a. Network Statistics

Figure 7 illustrates the effect on the mean distance statistic of removing up to 80% of the information from the three types of graph for a given situation. As can be seen in this figure, the values for mean distance between the three graph types overlap for high values of information loss, indicating that mean distance cannot be used to distinguish between these three types of graphs in this scenario. Note also that this figure plots 100 replicates of the removal of information from a defined starting point for each graph. Tables 1 and 2 elaborate on the replication procedure.

Figure 7. Plotting Mean Distance Under Information Loss of 80%



Each plotted line in this figure records a change in the mean distance statistic from 0–80% removals of edges, over 10 independent replicates of the information removal procedure applied to 10 graphs generated for the defined starting point. This provides a total of 100 plotted lines that depict both the variance of the statistic over the simulated scenario and the general effect of information loss on the statistics. As can be seen in this figure, the values for mean distance between the three graph types overlap for some values of information loss, indicating that mean distance cannot always be used to distinguish between these three types of graphs.

Table 1. Framework of Analysis of Graphs for Descriptive Modeling

Graph Type	**Deletion Type**	**No. of Graphs**	**Iterations/ Graph**
ER, SW, BA	Edge	10	10
	Vertex	10	10
	Edge	10	10
	Vertex	10	10
	Edge	10	10
	Vertex	10	10

As described in this table, for each of the three graph types, edges or vertices were deleted, with 10 random starting graphs generated, over 10 iterations per graph, over a step size of n.

Table 1 provides an overview of the replication procedure employed for plotting the performance of graph statistics. For each considered scenario (see Table 2 below) we generate 10 unique graphs of each type and simulate the random removal of information 10 times on each graph, providing 100 replicates (i.e. 100 plotted lines). This provides the opportunity to observe the variance in the statistic in a given scenario. In Figure 7 it can be observed that the mean distance statistic has little variance and behaves similarly for the ER and SW graphs, but the BA graph demonstrates much higher variance as information is removed.

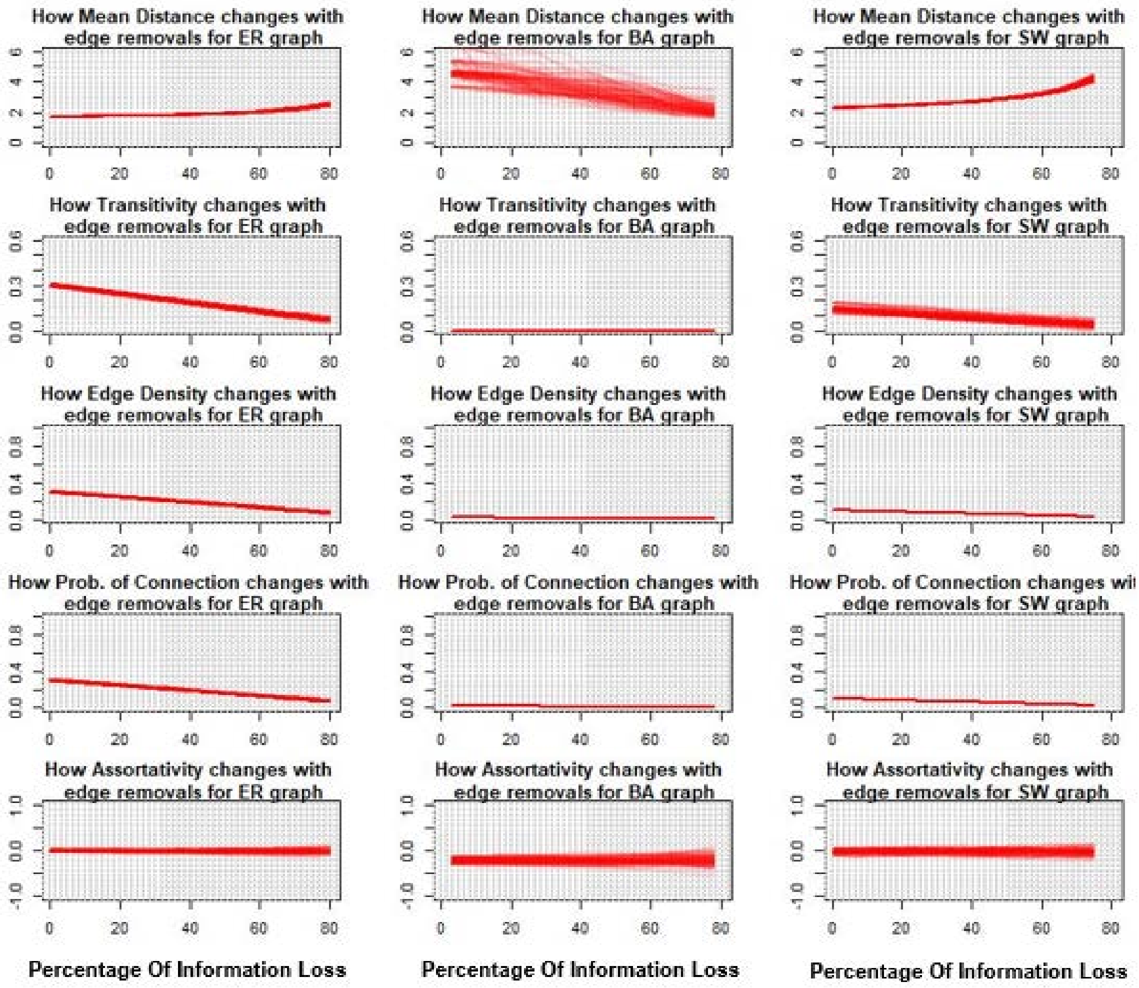
Table 2 summarizes the domain of consideration for scenarios during the descriptive analysis. As can be seen in the table, we varied graph type, graph starting size (i.e., the number of nodes before we begin removing information via simulation), and edge density. We then applied the simulation procedure discussed above to remove both edges and vertices and plotted the results (full results available in Appendix A).

Table 2. Domain of Consideration for Starting Graphs Generated for Descriptive Analysis of the Effects of Information Loss on Graph Statistics

Graph Type	**Size of starting graph**	**Edge Density**
ER, SW, BA	100 nodes	p = 0.1
		p = 0.2
		p = 0.3
		p = 0.4
		p = 0.5
	500 nodes	p = 0.1
		p = 0.2
		p = 0.3
		p = 0.4
		p = 0.5
	1000 nodes	p = 0.1
		p = 0.2
		p = 0.3
		p = 0.4
		p = 0.5

Figure 8 displays all of the statistics among the three different graphs (ER, SW, BA) for a given scenario in one panel with the same axes in order to make a comparison as to whether the statistics will eventually overlap (take on the same value), thus making classification difficult, or will remain distinct, with increasing proportion of removals. For example, in Figure 8, mean distance, transitivity, edge density and assortativity are plotted on the same axes between ER, SW and BA graphs. Plots for all of the scenarios considered in Table 2 are available in Appendix A.

Figure 8. Plotting Different Statistics with Varying Proportions of Incomplete Information for ER, SW and BA Graphs



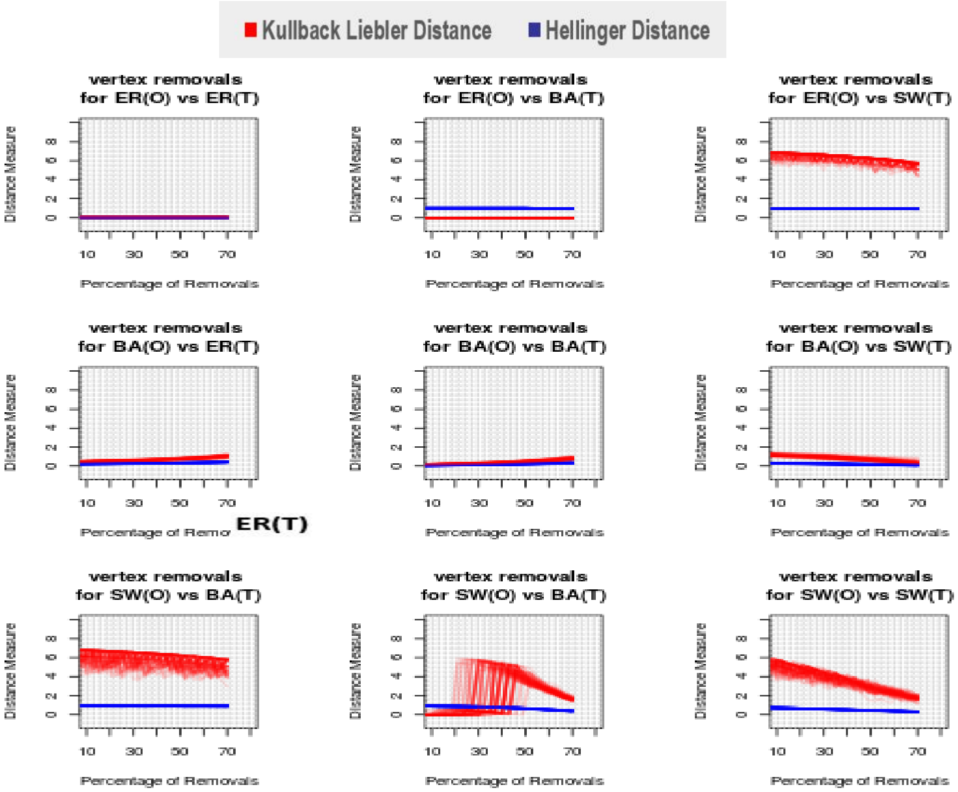
This figure shows that the mean distance statistics for this scenario (edge removals, starting graph: 100 nodes, 10 iterations, 10 graphs, edge density = 0.2) overlaps at various levels of information loss between the three different types of graphs as edges are removed. This indicates that this statistic cannot always be used to differentiate between the three types of network as information (edges in this case) is removed. As can be seen above, and by thoroughly reviewing the results provided in Appendix A, there is no single statistic that provides the ability to always differentiate between the three types of graph as information about the network is lost.

b. Distance Measures

In terms of distance measures, in order to determine which type of network the observed degree distribution resembles, we use distance measures to compare the observed degree distribution with the theoretical degree distribution of the networks at increasing proportions of removal. In order to do this, we store the observed degree distribution at different proportions of removal. For the theoretical degree distributions, we put observed statistics (such as current size of vertex list and the observed graph edge density) into the theoretical degree distribution of the ER, SW and BA networks as outlined in equations (5), (6) and (7) respectively, in order to compute a “goodness-of-fit test” with those distributions.

Then, we compute the Kullback-Leibler Divergence (Equation 8) and the Hellinger Statistic (Equation 9) on the observed and theoretical degree distributions at increasing proportions of removal of edges and vertices to determine if the distributions looked similar or distinct. From the results, we will be able to determine if degree distributions of the different graph types can be clearly distinguished. If the distance measure is small between two different graph types, we may conclude that these graphs might be mistaken for each other and may not appear distinct. In Figure 9, it can be seen that both the Kullback-Leibler and Hellinger distance statistics for BA observed degree distribution, i.e. BA(O) and SW theoretical degree distribution, i.e., SW(T) get closer to 0 with a larger proportions of vertex removals. This means that as information is lost, it becomes increasingly difficult to distinguish between the degree distributions of both graph types. Identifying the thresholds at which this confusion in classification of graph type is one of the key goals for this thesis.

Figure 9. Plotting Kullback-Leibler (red) and Hellinger (blue) Distance Measures for Observed (O) and Theoretical (T) Degree Distributions of all Three Network Types as Vertices Are Removed



In the center panel of this figure, looking at the plots from left to right for the observed BA graph, both the Kullback-Leibler and Hellinger distance statistic are close to 0 for the various levels of information loss. This means that the degree distributions of the observed BA graph look similar to that of the theoretical ER and SW degree distributions, making it difficult to distinguish between the three graph types under information loss.

4. Development of Datasets for Predictive Analysis Using a Design of Experiments (DOE)

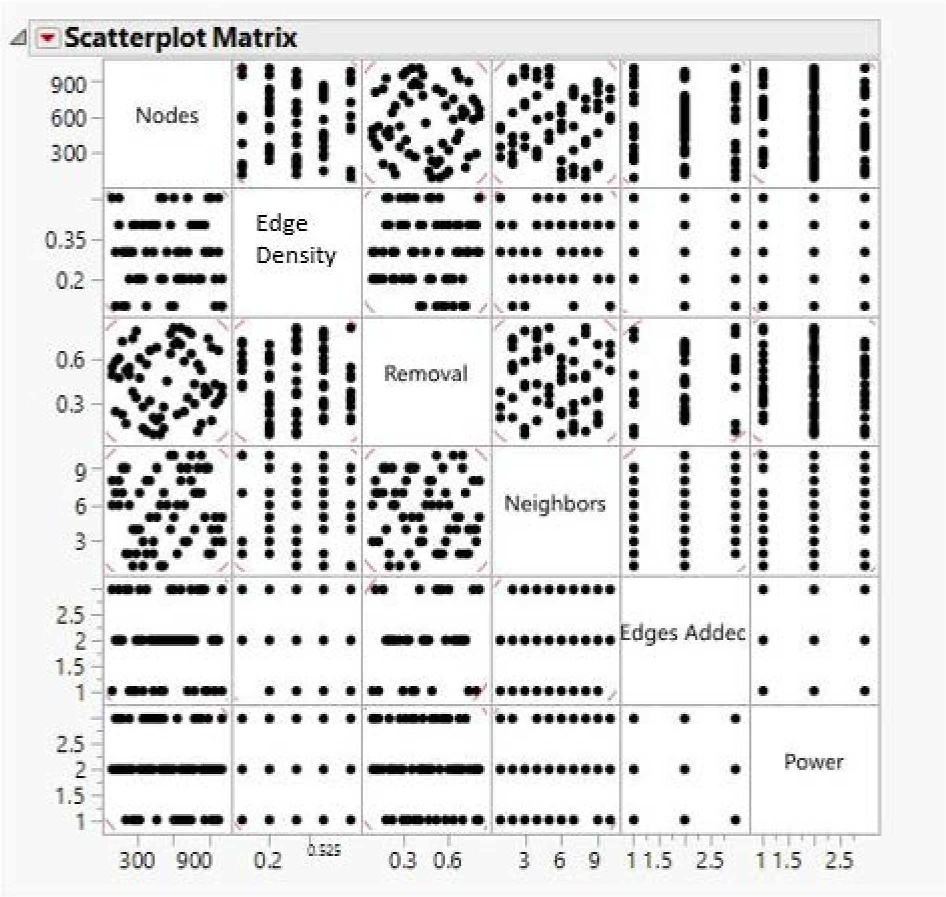
After establishing the behavior of the various statistics under the conditions of information loss using `igraph` defaults, we generate a *representative sample* of graphs under different proportions of information loss and calculate their observed statistics. Thus, we broaden the scope to include a greater variety of SW and BA graphs. We included cases for SW graphs with varying number of neighbors within which vertices are connected (from one to ten, where six is the standard for social networks, hence the term “six degrees of separation). For BA graphs, we vary settings in its preferential attachment model, namely the power of preferential attachment from one to three, and the number of additional edges added to highly connected nodes in each time step, also ranging from one to three. We apply a space-filling Design of Experiments (DOE) to generate the training and test datasets. This DOE provides a space-filling combination of factors that serves as a representative sample of graphs in the domain of consideration. One replication of the DOE table forms the training set; the second replication forms the test set.

The Design of Experiments (DOE) covers the following domain space:

- edge density (Continuous Factor), with range p = 0.1 to p = 0.5
- size (number of nodes) of the original graph (Continuous Factor), with a range of 100 to 1000
- proportion of information loss (Continuous Factor), with range 0.1 to 0.8
- number of neighbors within which the vertices will be connected in the SW graph (Continuous Factor), with range 1 to 10
- number of additional edges added in each time step (Continuous Factor) to the BA graph in its preferential attachment model, with range 1 to 3
- power of preferential attachment for BA graph (Continuous Factor), with range 1 to 3
- type of information loss (Categorical Factor), with factor levels edge or vertex
- graph type (Categorical Factor), with factor levels ER, SW or BA

The Nearly Orthogonal Balanced (NOB) design is often recommended for handling models with both discrete/categorical and continuous variables (Vieira et al. 2013). However, because we only have two categorical factors (graph type and deletion type), we used a cross design of the Nearly Orthogonal Latin Hypercube (NOLH) (Cioppa and Lucas 2007) for continuous factors and all the enumerations of the categorical factors. This approach provides a more space-filling design than the NOB because NOB does not guarantee that every combination of categorical factors is taken into account. For all the continuous factors, we choose the 33 level NOLH (Cioppa and Lucas 2007) at two stacks. This provides a good combination of being space-filling (covering new points) as well as replications (covering the same point). The space-filling design of the continuous factors are indicated in Figure 10. The design points are provided in Appendix B.

Figure 10. Design A: Space filling design for Continuous Factors at 33 level NOLH at Two Stacks and Two Replications



In this figure, the dots on the diagram represent the points of sampling of the various factors. The dots are space-filling and within the range and number of decimal places specified, the factors are sampled at many levels. For example, in the bottom row, power of preferential attachment (Power) is sampled at levels 1 to 3, for network size (Nodes) at levels 100 to 1000 (bottom row first column), edge density at levels 0.1 to 0.5 (bottom row second column), Proportion of Removals (Removal) at levels 0.1 to 0.8, number of neighbors within which SW graph is connection (Neighbors) at levels 1 to 10 and number of edges added in each time step for BA graph (No. of Edges) at levels 1 to 3. This is applied for all continuous factors to ensure they are sampled at space-filling levels.

Given that we only have two categorical factors, we enumerated all possibilities. Table 3 shows the various levels of the two categorical factors and Table 4 shows the crossed design (i.e., all enumerations of categorical factors).

Table 3. Categorical Factors

Deletion Type	Graph Type
Edge	ER
Vertex	SW
	BA

This table provides the two categorical factors in the Design of Experiments and their factor levels. For Deletion Type, either edge or vertex can be deleted. For Graph Type, either ER, SW or BA graphs can be generated.

Table 4. Design B: Crossed Design (All Enumerations of) Categorical Factors

Crossed Design
Edge, ER
Vertex, ER
Edge, SW
Vertex, SW
Edge, BA
Vertex, BA

This table provides the crossed design of all enumerations of categorical factors. For ER graphs, they can either be paired with an edge or a vertex deletion: hence the possibilities are (Edge, ER), (Vertex, ER). The same can be said for SW and BA graphs. This gives all enumerations of the categorical factors, which when crossed with the NOLH, provides a more space-filling design than that of the NOB.

We cross Design A (Continuous Factors) with Design B (Categorical Factors) to get Design C. Table 5 shows a sample of Design C. For the full Design of Experiments, see Appendix B.

Table 5. Design C: Crossed Design of Continuous Factors and Categorical Factors

Size of Starting Graph (Nodes) (100-1000)	Edge Density (0.1 to 0.8)	Proportion of Removals (0.1-0.8)	No. of Neighbors within which vertices are connected (1 to 10) for SW Graph)	No. of edges added in each time step (1 to 3) for BA graph	Power of preferential attachment (1 to 3) for BA graph	Removal Type (edge/ vertex)	Graph Type (ER/SW/ BA)
103	**0.1**	**0.23**	**1**	**10**	**1**	**Edge**	**ER**
407	**0.5**	**0.14**	**6**	**9**	**2**	**Vertex**	**SW**
609	**0.8**	**0.67**	**7**	**8**	**3**	**Edge**	**BA**

This table shows a sample of the various network parameters with which the initial graphs will be generated, and obscured (either edge/vertex) to the specified proportion. The complete DOE is in Appendix B. For each line (every design point, graph statistics are computed and recorded. This data will later be used in graph classification.

Based on this DOE, for each design point (every line in the Table in Appendix B), we computed the following statistics.

- mean distance
- edge density
- transitivity
- assortativity
- KL statistic with theoretical ER degree distribution
- KL statistic with theoretical SW degree distribution
- KL statistic with theoretical BA degree distribution
- H statistic with theoretical ER degree distribution
- H statistic with theoretical SW degree distribution
- H statistic with theoretical BA degree distribution

One replication of the crossed NOLH design formed my training set and the second replication formed my test set.

D. PREDICTIVE ANALYSIS USING MACHINE LEARNING METHODS

We leverage the Classification and Regression Trees (CART) (Breiman et al. 1984) and Random Forest (RF) (Breiman 2001) machine learning algorithms to build models for classification of observed networks. The CART model provides simple and easily interpreted rules for classifications and can be used to generate a simple ranking of variable importance (i.e., which statistics are the most important in classification). The RF model, which often provides a lower misclassification rate and higher predictive power, is used to understand how well we can predict (i.e., classify) networks in practice when less interpretable but more powerful machine learning methods are employed. We also leverage logistic regression modeling for the analysis of results which will be discussed further in the next section. A short description of each of these methodological approaches is provided here.

1. CART

Huddleston and Brown (2018) note that the Classification and Regression Tree (CART) algorithm provides highly interpretable models that illustrate the interaction between predictor variables in an easily understood format. However, this algorithm's predictive performance is typically not as good as less interpretable machine learning models such as those developed using the Random Forest (RF) or Adaboost algorithms (Huddleston and Brown 2018). CART models were used for two applications in this thesis. We used a CART model to develop a classifier that asserts one of three types of graphs (ER, SW, or BA) when presented with a set of statistics describing a network. This model was developed using the training set and the performance of the model evaluated on the test dataset.

2. Random Forest (RF)

Typically, the RF method, which grows many trees, provides better predictive power than CART. It classifies new objects by running the input data through many classification trees and consolidating the number of "votes" for a particular classification (Huddleston and Brown 2018). An RF model is thus a large ensemble of many (perhaps hundreds) different models and thus is much less interpretable. In this thesis, RF model will be used to understand how well we can predict (i.e., classify) networks in practice, as well as determine the level of information loss with which good (>90% accuracy) classification of networks can still take place.

E. RELATING DOE FACTORS TO CLASSIFICATION PERFORMANCE

The last step in this analysis uses both logistic regression and CART models to relate classification performance (i.e., the ability to accurately classify an obscured network) to the various factors considered in the design of experiments table. Due to the nearly orthogonal design used for the DOE, we can develop models that use classification performance (i.e., correct or incorrect classification of a DOE design point in the test dataset) as the response (dependent) variable of a regression analysis with the parameters specified in the DOE table as the predictor (independent) variables. We employ both logistic regression and CART models for this purpose.

Logistic regression models are used to predict a response variable that is categorical from continuous and categorical predictors (Agresti 2012). Logistic regression provides a means for both visually and statistically capturing the effect of parameters varied in the simulation such as percentage of information lost/hidden, the different graph types, the edge density (*p*), etc. We also employ CART models to perform the same task because CART models classify using thresholds of the predictor variables rather than mapping continuous relationships as logistic regression models do. Both modeling approaches provide the opportunity to study the effect of the various DOE factors on classifier performance and identify the scenarios in which it is reasonable to assert a classification for a network in real-world practice. The results of this analysis are discussed in depth in the next section.

IV. ANALYSIS OF RESULTS

This chapter summarizes the key results. An exhaustive set of diagrams of results with the parameters listed in Table 2 are written in Appendix A. In this section, we summarize the key observations derived from both the descriptive and predictive analysis. The three most significant results of this analysis are:

- As information on edges is lost (>80%), it becomes more difficult to distinguish between ER and SW graphs.
- As information on vertices is lost (>80%), it becomes more difficult to distinguish between BA and SW graphs.
- If at least 20% of the information about the network is available, RF can classify a network with >90% accuracy.

A. DESCRIPTIVE ANALYSIS

1. Stability of Graph Statistics for Classification of Graphs

In Chapter II, we outline some characteristics of three network models. For instance, ER and SW networks have small mean distances, but between ER and SW networks with comparable size, SW networks would have a higher transitivity. BA networks are also characterized by a scale-free degree distribution, while ER networks have a Poisson degree distribution and SW networks have a binomial degree distribution (Costa et al. 2005).

With descriptive modeling under the environment of incomplete information, we find that the above-mentioned characteristics for classification between ER and SW holds true for low edge density (p = 0.1) of starting graphs. However, as p increases to 0.5, the transitivity of ER graphs increased and we find that ER graphs have a higher transitivity than SW graphs. A high proportion of **edge** removals (> 80%) show both the transitivity and edge density for ER and SW graphs converging. This observation is also supported by simulation results that with increasing edge removals and as p increases to 0.5, the KL statistic converges to 0 for the observed and theoretical degree distributions of the ER and SW graphs. ***In other words, as information on edges is lost (>80%) and as the starting edge density in the graph increases, it becomes more difficult to distinguish between ER and SW graphs.*** Increasing the starting graph's edge density would make the ER graphs denser, hence they appear to be more like SW graphs, which are characterized by high clustering coefficients. For BA and SW graphs under vertex removals, their observed statistics (transitivity, edge density and assortativity) remain constant and equal despite information loss. In addition, there is little difference between the KL statistics for the observed and theoretical degree distributions for BA and SW graphs as the proportion of vertex removals increase. ***In other words, as information on vertices is lost (> 80%), it becomes more difficult to distinguish between BA and SW graphs.***

2. Comparison of the Effects of Edge vs. Vertex Removals

The key difference between edge and vertex removals is their effect on the edge density. These change behaviors of statistics that are dependent on edge density. In particular, referencing equation (3), with edge removals, the number of edges, which is the numerator of the edge density (*p*), decreases, while the number of vertices *N*, and hence the denominator of *p*, which is the maximum possible number of edges $\binom{N}{2}$**,** remains the same. As a result, with edge removals the observed *p* decreases. This affects statistics such as the mean distance calculation. With a decreased edge density, mean distance for ER graphs increases. In Appendix C, (Yoshida 2018) explicitly computes the mean distance for ER graphs and shows that the mean distance for an ER graph converges almost surely; hence the simulation results match and motivate the mathematical proof developed.

Vertex removal produces different results. Both the numerator and denominator decrease proportionately with vertex removal because when vertices all of their corresponding edges are removed as well. Hence, with vertex removals, the edge density remains constant. One can prove that the vertex removals would not change the mean distance if we consider the ER model. In fact, we can prove that the mean distance for an ER graph remains the same almost surely via vertex removal and we can explicitly compute the mean distance for the ER model.

3. General Behavior of Statistics under Information Loss

a. Mean Distance

Mean distance measures respond slightly differently for edge and vertex removals, and also according to the graph type. For edge removals, the edge density (*p*) decreases, hence the mean distance increases for both the ER and SW graph. For vertex removals, *p* remains constant, hence the mean distance remains constant for both ER and SW. For BA graphs, as the edges are removed, and vertices become singletons, they are removed from the mean distance computation. In this case, the general relationship is the same for both edge and vertex removal because the BA network is not affected by the value of *p*. Instead it is affected by the power of preferential attachment. There were also possible confusions in magnitude of the statistics and this would mean that graphs might possibly be confused with each other. For instance, an ER network with many edge removals would have the same magnitude of mean distance as a BA network.

b. Transitivity

With edge removals, transitivity decreases for all networks except BA, and for vertex removals, the number remains constant. With a higher starting edge density *p*, this statistic starts out higher with the magnitude *p*. An ER and SW network with a large proportion (~80%) of edge removals would have the same magnitude of transitivity.

c. Edge Density

With edge removals, the edge density decreases for all networks except BA (which does not depend on the edge density), and for vertex removals, the edge density remains constant. *This is because vertex removals remove the vertices and corresponding edges in a set, hence the edge density remains constant.* With a higher starting edge density *p*, this statistic starts out higher with the magnitude *p*. An ER graph and SW graph with a large proportion (~80%) of edge removals often have the same magnitude of density.

d. Assortativity

This statistic remains at the same magnitude despite incomplete information and changes in the starting graph's edge density.

e. Kullback-Leibler (KL) Divergence vs. Hellinger Distance Statistic

For both KL divergence and the Hellinger Distance statistic, if the observed data is from the same graph type as the theoretical distribution, the distance values are close to 0 (i.e. ER(O) and ER(T), SW(O) and SW(T), BA(O) and BA(T).

However, the KL divergence had higher magnitudes as a distance statistic compared to the Hellinger Distance statistic because it is computed using the logarithm of one element of the degree distribution over the other, which could inflate the distance measures for values of elements of degree distributions near 0.

B. PREDICTIVE MODELING

Next, we use the training set to develop two models using Classification and Regression Tree (CART) Breiman et al. (1984) and Random Forest (RF) Breiman (2001). The main findings are as follows:

- Up to approximately 80% information loss, the three graphs could be classified with >90% accuracy (79.74% for CART and 91.25% for RF). This means that intelligence analysts only require ~20% of the network to assert an accurate network classification when machine learning methods are employed.

- Mean distance, the Hellinger distance statistic, transitivity and edge density are the top four network statistics in terms of variable importance in classifying the network type.

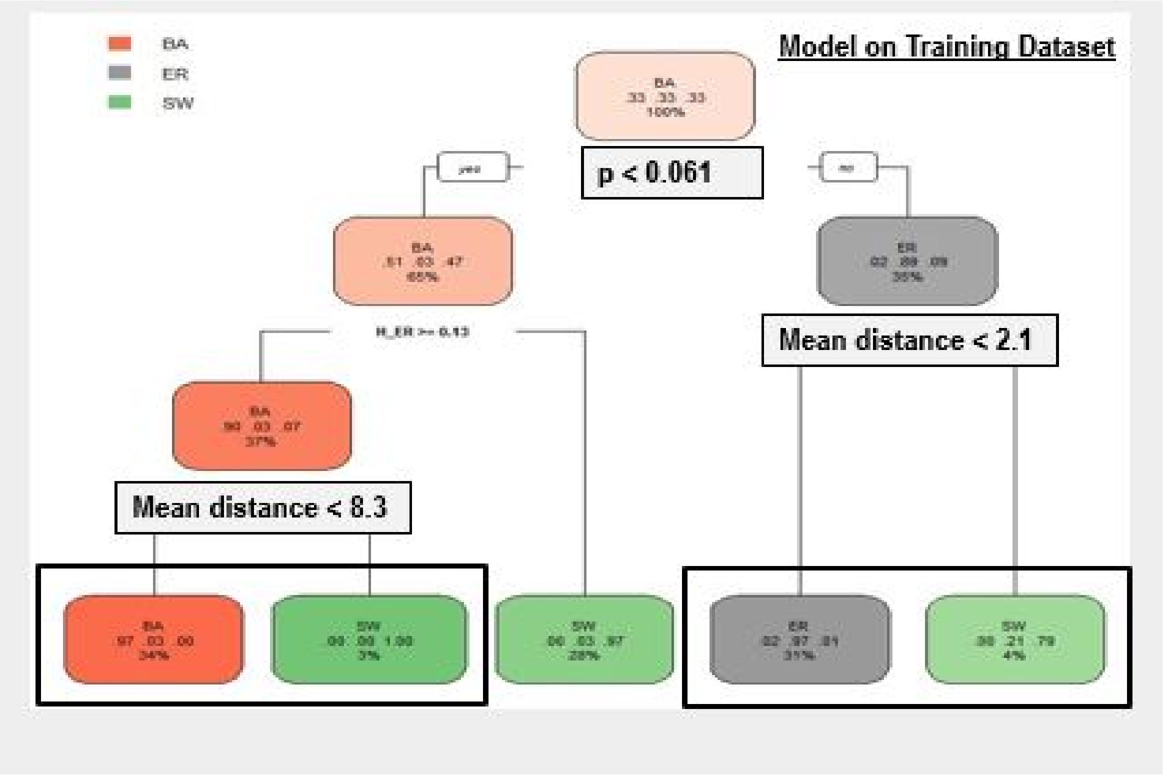
- Proportion of removal (i.e., percentage of information loss), deletion type (nodes or edges) and graph type (ER, SW, and BA) are the top three situational factors that affected the ability to accurately classify networks.

1. Rules/Thresholds of Observed Graph Statistics that Guide the Classification of the Three Networks

From the CART model using network statistics for classification, as indicated in both Figure 11 and Table 6, we find the following to be true about classification of networks. First, the SW networks are the most difficult to accurately classify. If the network has an observed edge density > 0.061 and an observed mean distance < 2.1, they can be classified as SW graphs. If the network has an observed edge density < 0.061 and an observed mean distance

> 8.3, they can also be classified as SW networks. However, if they have an observed edge density < 0.061 an observed Hellinger distance statistic with the theoretical ER model < 0.13, they can also be classified as SW networks. For ER networks, if the observed edge density > 0.061 and an observed mean distance >2.1, they can be classified as ER networks. For BA networks, if they have an observed edge density < 0.061 and observed mean distance < 8.3, they can be classified as BA networks.

Figure 11. CART Model Using Graph Statistics for Network Classification



For classification of networks, this figure outlines that edge density (*p*) of the network is the first differentiating factor. Similar to the results from the descriptive analysis, a high edge density (*p*) can lead to misclassification between the ER and SW graphs. They are later differentiated by mean distance threshold of 2.1. Similar to results from descriptive analysis, BA and SW graphs are in danger of being misclassified. They are later differentiated by mean distance threshold of 8.3 with 97% accuracy for BA graphs and 100% accuracy for SW graphs. They can also be differentiated with 97% accuracy with the Hellinger distance statistic fitted with an ER graph.

Table 6. Findings from CART (Network Statistics) on Classification of Graphs

Graph Type	Edge Density/ Prob. of conn. between vertices	Mean Distance	Hellinger statistic with theoretical ER model
ER	> 0.061	> 2.1	N.A
SW	> 0.061	< 2.1	N.A
	< 0.061	> 8.3	N.A
	< 0.061	N.A	< 0.13
BA	< 0.061	< 8.3	N.A

The table shows that if the network has an observed edge density > 0.061 and an observed mean distance < 2.1, they can be classified as SW graphs. If the network has an observed edge density < 0.061 and an observed mean distance > 8.3, they can also be classified as SW networks. However, if they have an observed edge density < 0.061 an observed Hellinger distance statistic with the theoretical ER model < 0.13, they can also be classified as SW networks. For ER networks, if the observed edge density > 0.061 and an observed mean distance >2.1, they can be classified as ER networks. For BA networks, if they have an observed edge density < 0.061 and observed mean distance < 8.3, they can be classified as BA networks.

Table 7 provides a summary of the variable importance for the different statistics employed by the CART model. The mean distance, Hellinger distance statistic, transitivity and edge density are the top four network statistics in terms of variable importance in classifying the network type as detailed in Table 7. Note that this machine learning approach provides a “competition” for a given network statistic and thus a means for evaluating the effectiveness of newly developed network statistics for classification of networks under real-world conditions (i.e., with missing information). Any newly developed statistic can be inserted into this analysis and its effectiveness evaluated against that of its “peers” based upon its contributions to the ability to classify under this context.

Table 7. Variable Importance from CART (Network Statistics)

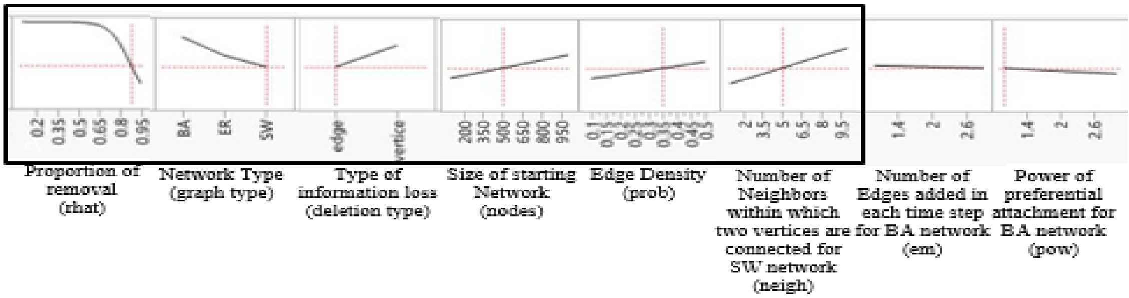
S/N	Variable	Score
1.	Mean Distance	18
2.	Transitivity	12
3.	Hellinger Statistic with theoretical ER degree distribution	10
4.	Edge Density	10
5.	Assortativity	9
6.	Kullback-Leibler statistic with theoretical ER degree distribution	8
7.	Hellinger Statistic with theoretical BA degree distribution	8
8.	Kullback-Leibler statistic with theoretical BA degree distribution	7
9.	Kullback-Leibler statistic with theoretical SW degree distribution	5
10.	Hellinger Statistic with theoretical SW degree distribution	1

From the CART (Network Statistics), the mean distance, transitivity, Hellinger statistic with theoretical ER degree distribution, as well as the edge density are the top four in terms of importance in classifying the network.

2. Effects of Changing Network Parameters on Classification Accuracy

Figure 12 illustrates the effects that the studied network characteristics have on the ability to accurately classify networks derived through logistic regression modeling. This figure depicts the general shape of the effect, with a positive slope from left to right in Figure 12 indicating improvement in classification performance as network parameter values increase. For example, as the starting network size increases, our ability to classify networks improves. The same can be said edge density (prob) and number of neighbors (neigh) within which two vertices are connected in a SW network: as they increase, our ability to classify networks correctly improves. Factors within the bold box in Figure 12 were statistically significant effects (as identified through results of a logistic regression in Figure 13), which provides the summary of the logistic regression model used to analyze these effects. As seen in Figure 13, the p-values for the edge density (prob), size of starting network (nodes), and number of neighbors within which two vertices are connected in a SW network (neigh) are low as indicated in Figure 13, thus we reject the null hypothesis that they are not statistically significant in favor of the hypothesis that they are significant. The edge density (prob), size of starting network (nodes), and number of neighbors within which two vertices are connected in a SW network (neigh) have a significant effect on our ability to classify networks correctly.

Figure 12. Prediction Profiler for Network Parameters in Classifying Networks

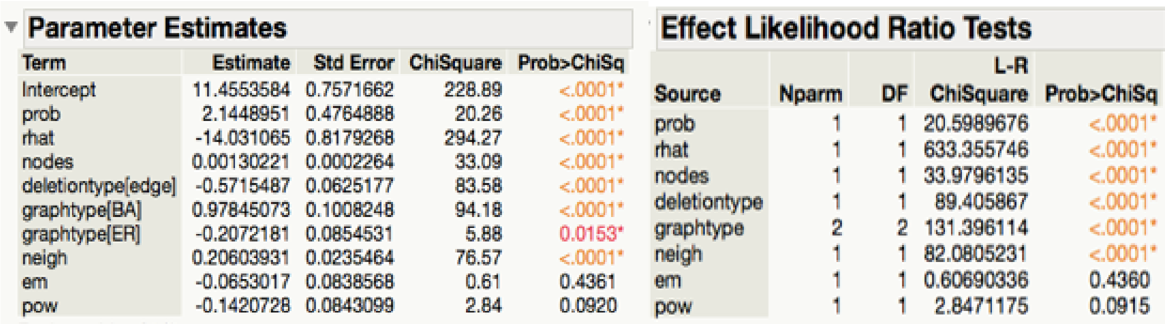


The figure outlines the relationship between factor levels and classification performance as found in logistic regression analysis. Factors which are statistically significant are highlighted in the bold box. For those with a positive slope (network size, edge density, number of neighbors within which two vertices are connected), increasing factor levels increases the probability of correct classification. A high proportion of removals, having a SW graph or hidden edges increases the probability that the graph is misclassified.

Conversely, parameters with a negative slope such as proportion of removals (>80%) indicate that as proportion of removals increase, (or as network type is SW), our ability to classify networks accurately worsens. The p-values of the proportion of removals (rhat) and the network type (graph type) are low as indicated in Figure 13. Thus, we reject the null hypothesis that they are not statistically significant in favor of the hypothesis that they are significant factors. Therefore, we can conclude that the proportion of removals and network type have a significant effect on our ability to classify networks correctly.

Both the number of edges added in each time step for the BA network (em) as well as the power of preferential attachment for the BA network (pow) have p-values greater than the significance level which we set to .05. Though both have a slightly negative slope, these factors are not statistically significant and hence we can assert they have a negligible effect on our ability to classify networks correctly.

Figure 13. Parameter Estimates and Effect Likelihood Ratio Test Results for Network Parameters in Classifying Networks



Parameter Estimates

Term	Estimate	Std Error	ChiSquare	Prob>ChiSq
Intercept	11.4553584	0.7571662	228.89	<.0001*
prob	2.1448951	0.4764888	20.26	<.0001*
rhat	-14.031065	0.8179268	294.27	<.0001*
nodes	0.00130221	0.0002264	33.09	<.0001*
deletiontype[edge]	-0.5715487	0.0625177	83.58	<.0001*
graphtype[BA]	0.97845073	0.1008248	94.18	<.0001*
graphtype[ER]	-0.2072181	0.0854531	5.88	0.0153*
neigh	0.20603931	0.0235464	76.57	<.0001*
em	-0.0653017	0.0838568	0.61	0.4361
pow	-0.1420728	0.0843099	2.84	0.0920

Effect Likelihood Ratio Tests

			L-R	
Source	Nparm	DF	ChiSquare	Prob>ChiSq
prob	1	1	20.5989676	<.0001*
rhat	1	1	633.355746	<.0001*
nodes	1	1	33.9796135	<.0001*
deletiontype	1	1	89.405867	<.0001*
graphtype	2	2	131.396114	<.0001*
neigh	1	1	82.0805231	<.0001*
em	1	1	0.60690336	0.4360
pow	1	1	2.8471175	0.0915

In the figure, the edge density (prob), proportion of removals (rhat), starting network size (nodes), type of removal (deletiontype), graph type (graphtype) and number of neighbors (neigh) within which two vertices are connected (SW graph) have p values less the significance level (which we set to 0.05), hence we reject the null hypothesis that they are not statistically significant and consider them statistically significant network parameter.

Both the CART and logistic regression on network parameters indicated that the proportion of removals, type of missing information (edge vs. vertex) and starting network type are the top three parameters in classifying networks correctly. The detailed results are in Table 8 and the full CART analysis can be found in Appendix C.

Table 8. Variable Importance from CART (Network Parameters)

S/N	Variable	Score
1.	Proportion of Removals	34
2.	Deletion Type (Nodes vs. Edges)	25
3.	Starting Network Type (ER, SW, BA)	16
4.	Edge Density	12
5.	No. of Neighbors (SW graph)	6
6.	No. of nodes of starting graph	4
7.	Power of preferential attachment (BA graph)	2
8.	No. of Edges added in each time step (BA graph)	1

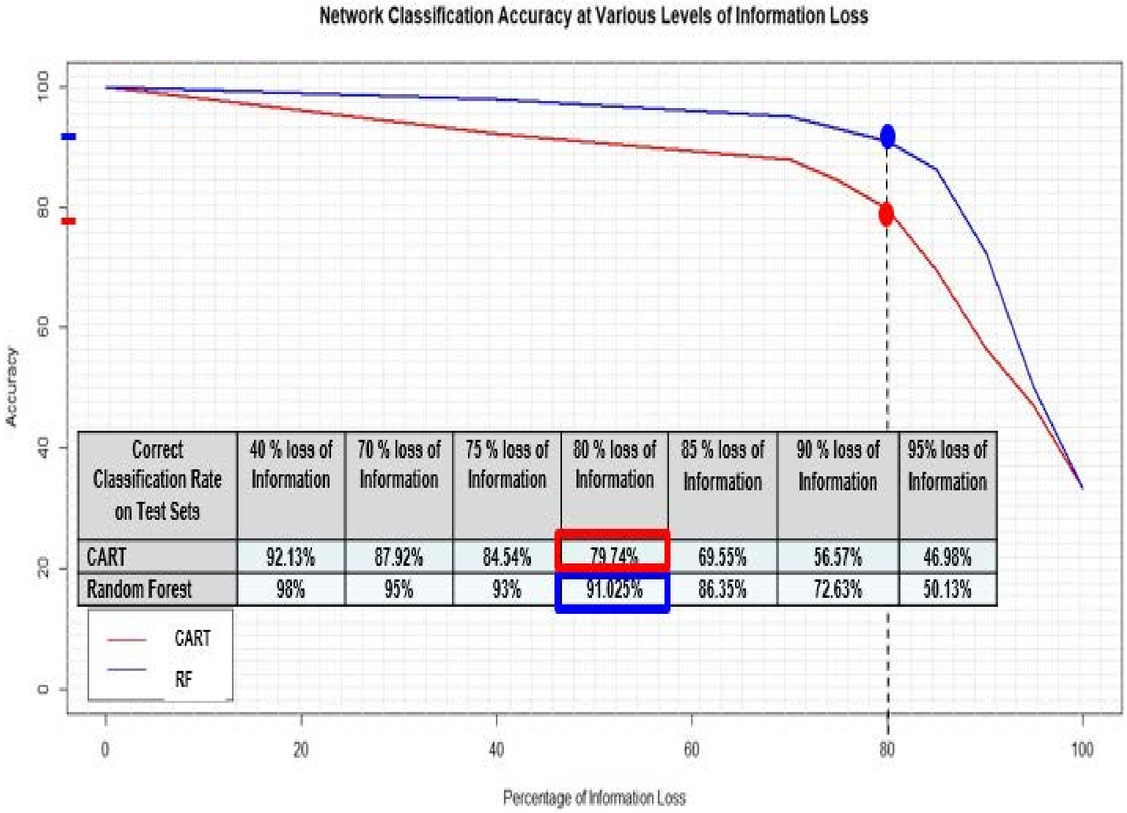
As outlined in this table, from the CART (Network Parameters), the proportion of removals, deletion type and graph type are the top three in terms of importance in classifying the network.

3. Classification Accuracy vs. Information Loss

Once the CART and RF models had been generated, we extend the DOE design to generate additional scenarios in order to better estimate the effect of information loss on our ability to classify. We conduct full designs at additional points of information loss in order to further quantify the effect of information loss (this had the effect of “smoothing the curve” of the estimated effect). Additional design points are created at information losses of 40%, 70%, 75%, 80%, 85%, 90% and 95% respectively. Figure 14 illustrates the relationship between network classification accuracy via-a-vis information loss.

We find that the classification accuracy remained relatively stable up to about 80% information loss before it dropped sharply, as per the results in Figure 12 for parameter proportion of removals. This finding has a significant impact for intelligence analysts. In particular, it tells us that ***we do not need to spend resources attempting to map the entire network in order for us to be able to assert an accurate network classification***. In fact, ***only about 20% of the network is required to give us a classification of >90% accuracy***.

Figure 14. Network Classification vis-à-vis Information Loss



Correct Classification Rate on Test Sets	40 % loss of Information	70 % loss of Information	75 % loss of Information	80 % loss of Information	85 % loss of Information	90 % loss of Information	95% loss of Information
CART	92.13%	87.92%	84.54%	79.74%	69.55%	56.57%	46.98%
Random Forest	98%	95%	93%	91.025%	86.35%	72.63%	50.13%

In the figure, both the CART (red line) and RF (blue line) have classification accuracy that dips after information loss is beyond 80%. The RF model performs better than the CART model for all levels of information loss, but a key finding is that at 80% information loss, the CART has 79.74% of accuracy while the RF model has 91.025% accuracy.

V. CONCLUSION

A. ANSWERS TO RESEARCH QUESTIONS

The fundamental research questions that this thesis had sought to answer were as follows. **In the context of missing information (i.e., an incomplete network mapping):**

1. Which network statistic gives the highest predictive power in classifying the network type?

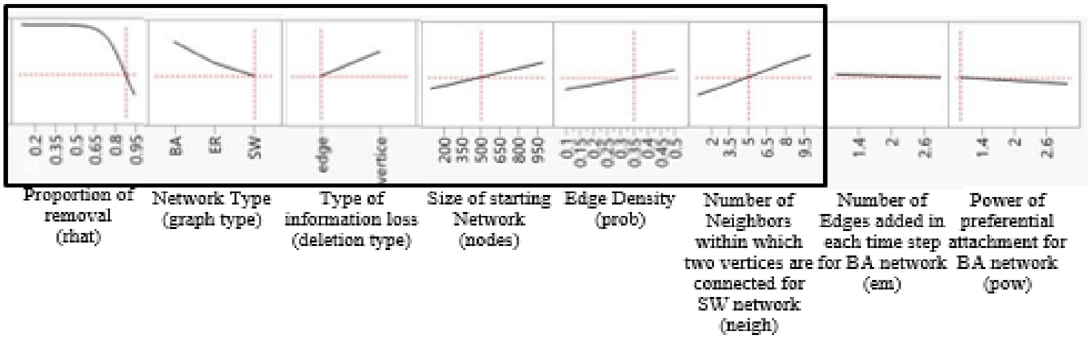
    - From the findings of the thesis, the statistic with the highest predictive power in classifying the network type is mean distance. However**, *no one statistic is sufficient to distinguish between these three network types when information loss is considered***. Even the simplest CART model requires more than one statistic to help to classify the network, hence there is ***significant benefit in using machine learning methods such as random forest to form ensembles for classification***.

2. What is the effect of changing the following parameters on the ability to classify a graph type correctly?

    - edge density of the original network (*p*)
    - size (number of vertices) of the original network (nodes)
    - proportion of information loss (rhat)
    - type of information loss (deletion type)
    - network type (graph type)

    - **The most significant finding is that the ability to accurately classify networks declines precipitously once more than 80% of the information about the network is missing.** Figure 15 summarizes the general shape of the effects, with movement towards the top of the figures indicating improvement in classification performance. Factors within the bold box were statistically significant effects. Both the CART and logistic regression models indicate that the proportion of removals (i.e., amount of missing information), type of missing information (edge vs. vertex) and true network type are the most significant factors affecting network classification.

Figure 15. Prediction Profiler for Graph Characteristics in Classifying Networks



The figure outlines the relationship between factor levels and classification performance as found in logistic regression analysis. Factors which are statistically significant are highlighted in the bold box. For those with a positive slope (network size, edge density, number of neighbors within which two vertices are connected), increasing factor levels increases the probability of correct classification. A high proportion of removal, SW graph or hidden edge, increases the probability that the graph is misclassified.

3. Can we establish a framework through which we can learn (1) and (2) for any network statistic?

    - This thesis provides a framework for the evaluation of the contribution of various network statistics on our ability to classify graphs. To date, research in this area has focused on analyzing the performance of individual network statistics for classifying networks. While this thesis has found that an ensemble of statistics can classify a network with high accuracy using RF, it has also established a framework by which any new statistic can be evaluated for its utility to classify networks under conditions of incomplete information. In particular, we develop a framework to generate a space-filling set of graph statistics under real-world conditions of information loss that can be used to generate both training and test sets to develop predictive models and simple rules for classification. These models provide a means to assert the relative importance of network statistics under various levels of information loss based on their contribution of predictive power within these ensemble models.

B. APPLICATIONS FOR INTELLIGENCE ANALYSIS

a. Importance of Edges

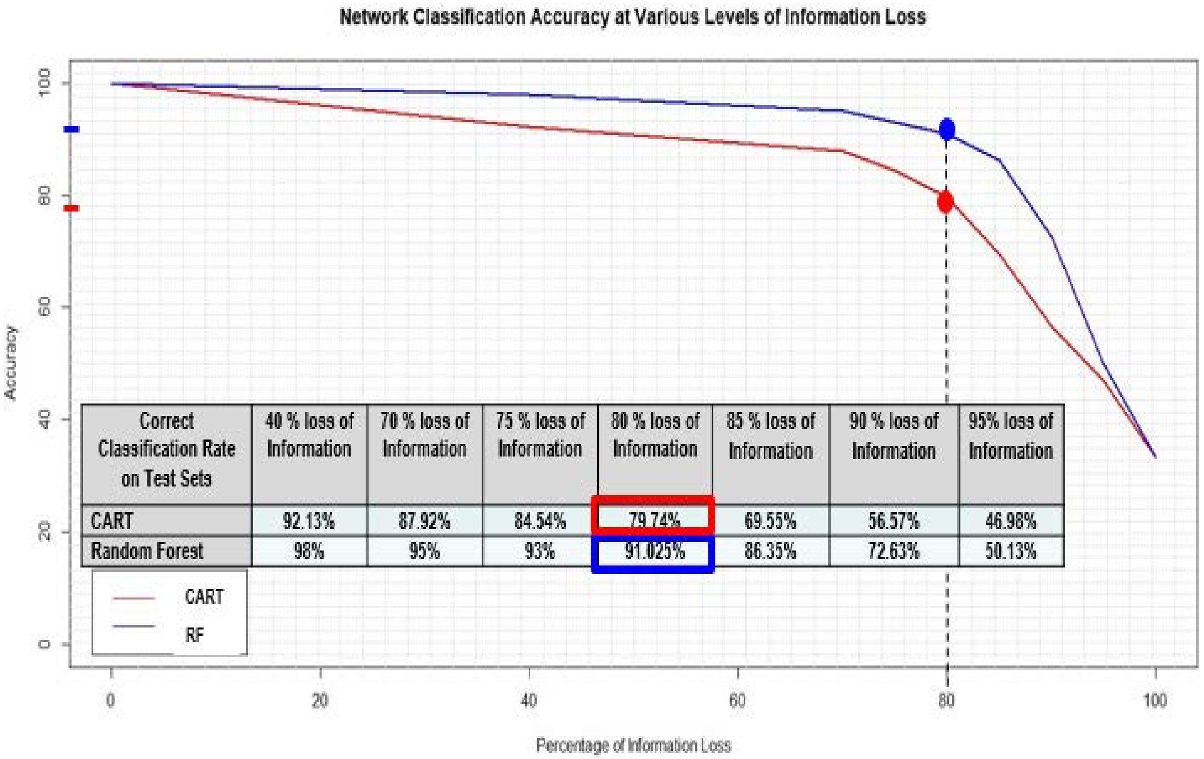
Traditionally, the intelligence community regards information on vertices more highly than that of edges because vertices traditionally represent people or organizations that serve as the starting seed with which it builds information on networks. However, this thesis has brought out the importance of edges for classification. Besides the fact that we can only discover new vertices through existing edges, and knowledge of the network grows one edge at a time, this thesis illustrates how hidden edges can lead to a mis-estimation of the edge density, *p*, making the graph look vastly different, and possibly leading to a misclassification of the network type. For the purpose of classifying networks in order to develop strategies for the destruction of networks, understanding the way the network is connected is more important than mapping out all the entities in the network.

b. Classification of Graphs under Information Loss

Figure 16 provides a detailed view of the effect of information loss on our ability to classify network graphs with both CART and RF models. The amount of information missing is the most significant effect of our ability to distinguish between these three network types commonly encountered in intelligence analysis. This graph provides several key insights for applications of network analysis in the intelligence domain. First, only about 20% of the information about a network is needed in order to achieve better than 90% accuracy in network classification. This means that we do not need to spend resources to completely map a network in order to accurately classify it.

Second, the significant performance improvement of the RF ensemble machine learning model in this study over the use of simple thresholds based on individual statistics suggests that this approach should be directly fielded for counter-network applications in the DoD. The classification models developed as part of this thesis, trained on a wide variety of synthetically generated networks, should provide significantly improved classification performance in practice over the current methods used, which use single statistics and assume a complete mapping of the network.

Figure 16. Network Classification vis-à-vis Information Loss



Correct Classification Rate on Test Sets	40 % loss of Information	70 % loss of Information	75 % loss of Information	80 % loss of Information	85 % loss of Information	90 % loss of Information	95% loss of Information
CART	92.13%	87.92%	84.54%	79.74%	69.55%	56.57%	46.98%
Random Forest	98%	95%	93%	91.025%	86.35%	72.63%	50.13%

In the figure, both the CART (red line) and RF (blue line) have classification accuracy that dips after information loss is beyond 80%. The RF model performs better than the CART model for all levels of information loss, but a key finding is that at 80% information loss, the CART has 79.74% of accuracy while the RF model has 91.025% accuracy.

This research also suggests that, unless we can be reasonably sure that we have sufficient information, we should be very cautious about proposing specific strategies for the dismantling of threat networks based on network classifications conducted on small samples of larger (and mostly unobserved) networks. The widespread practice of asserting a network classification based on a single statistic such as degree distribution calculated on a small observed sample of a much larger (but mostly unobserved) network is unlikely to result in accurate network classification and therefore effective strategies. Rather, a reasonable standard would require that we (1) believe we have observed at least 20% of the network, (2) have reason to believe the network is one of the three types studied in this thesis (or we have replicated this framework for additional network types), and (3) we have developed a classification model whose performance for the desired application is known and validated.

C. FUTURE WORK

This thesis provides a foundation for future work on network analysis in the environment of incomplete information. In particular, it examines how a random removal of nodes and edges affected network statistics. In future work, analysis on the effect on vertex (node) centrality measures would be helpful in helping the intelligence community understand how they can interpret such measures in light of incomplete information. In addition, while the thesis simulates incomplete information through the random removal of vertices and edges starting from a complete network, work can be done to examine the effect on statistics of random addition of nodes and edges, mimicking the network mapping process that intelligence analysts do. Combinatory losses of vertices and edges should also be examined, as well as finding methods to quantify and model a non-random (systematic) loss of information due to systematic bias. The effect of extra nodes and edges (false positives in intelligence collection) as well as investigating network statistics taking into account direction should also be explored. Finally, an exploration can also be done on how well we can classify known networks mapped with a simulated Query-Expand-Collapse process.

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX A. RESULTS FROM DESCRIPTIVE MODELING

Figure 17. Edge Removals, Starting Graph: 100 Nodes, 10 Iterations, 10 Graphs, *p* = 0.1

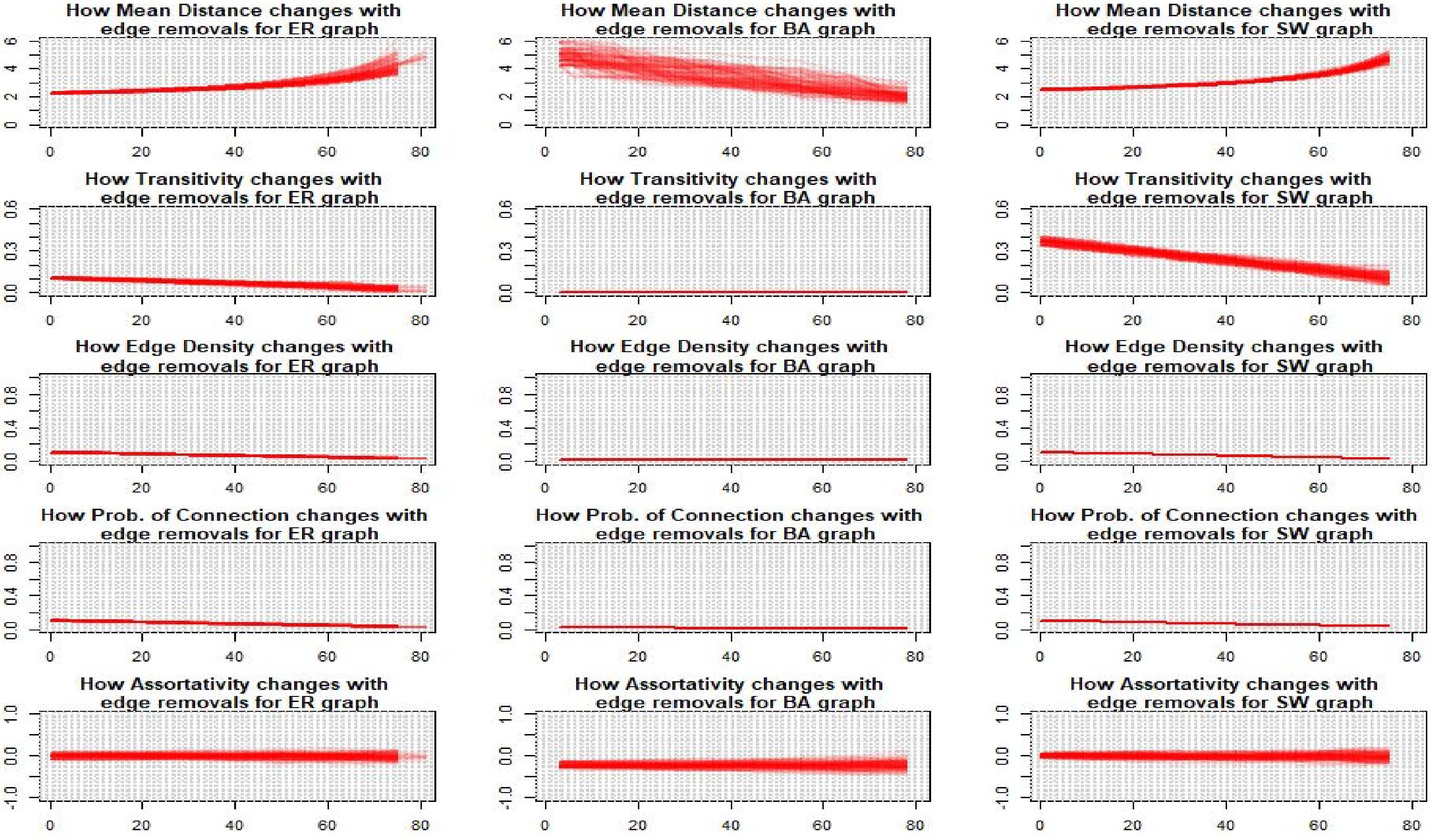


Figure 18. Edge Removals, Starting Graph: 100 Nodes, 10 Iterations, 10 Graphs, *p* = 0.2

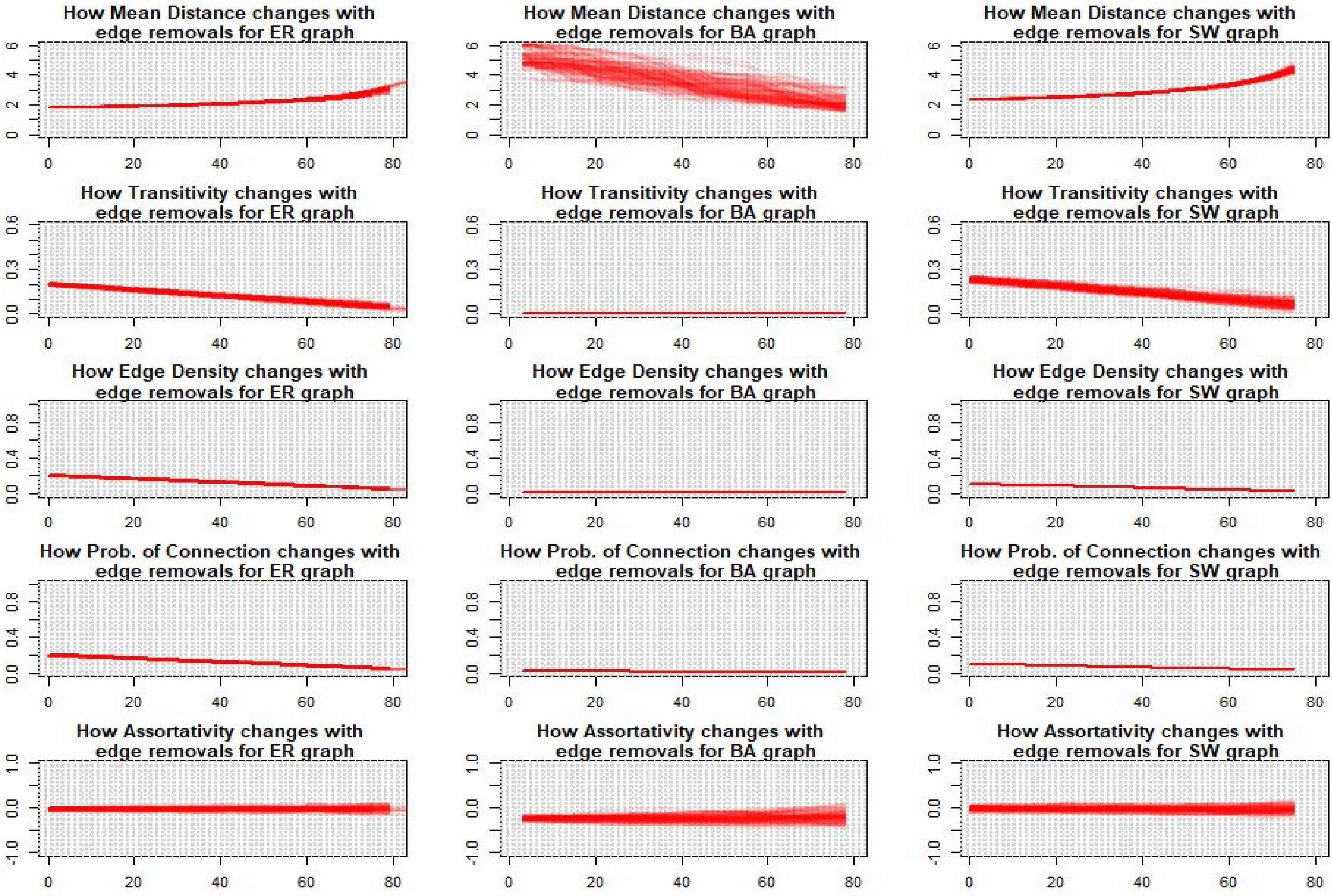
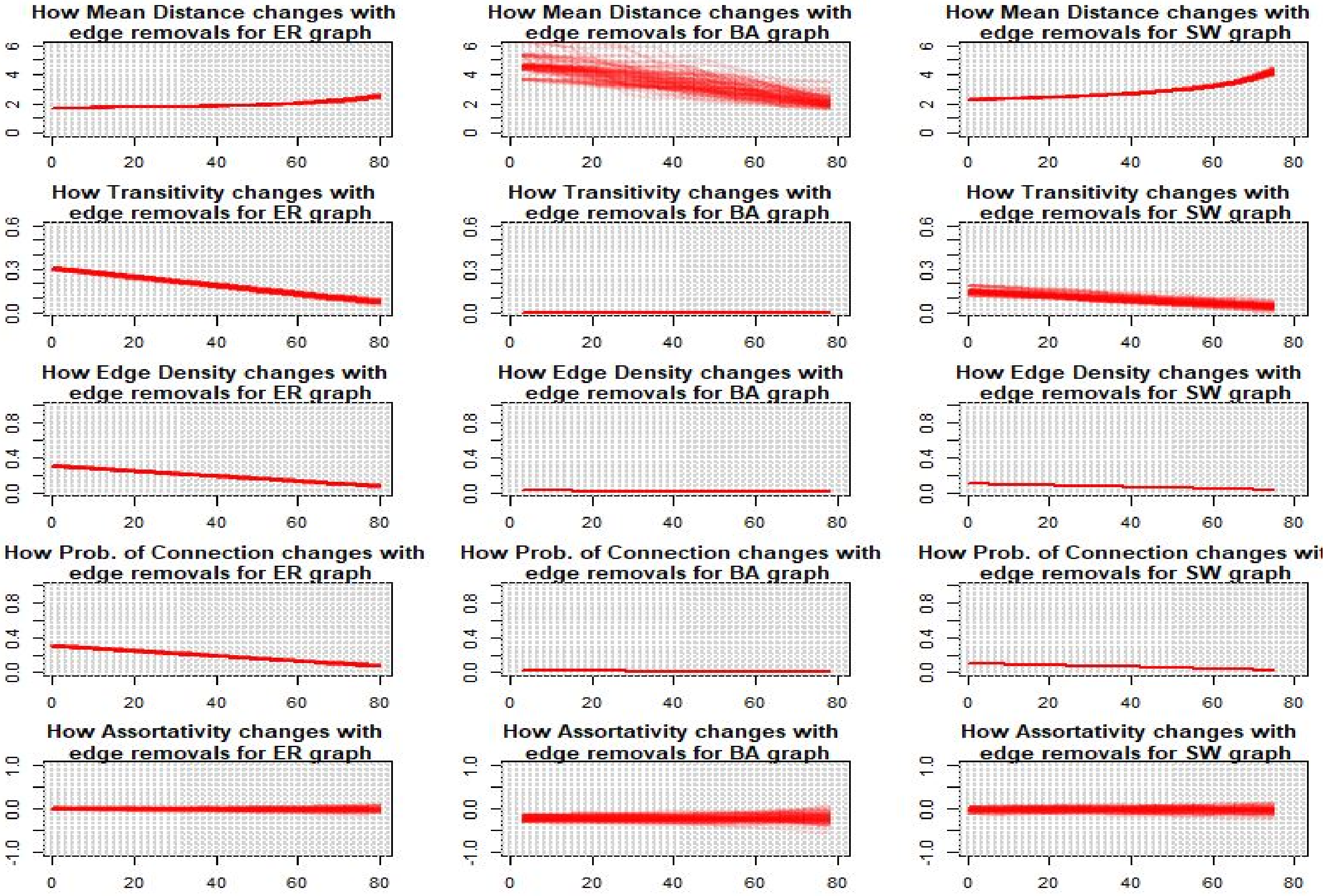


Figure 19. Edge Removals, Starting Graph: 100 Nodes, 10 Iterations, 10 Graphs, p = 0.3



How Mean Distance changes with edge removals for ER graph

How Mean Distance changes with edge removals for BA graph

How Mean Distance changes with edge removals for SW graph

How Transitivity changes with edge removals for ER graph

How Transitivity changes with edge removals for BA graph

How Transitivity changes with edge removals for SW graph

How Edge Density changes with edge removals for ER graph

How Edge Density changes with edge removals for BA graph

How Edge Density changes with edge removals for SW graph

How Prob. of Connection changes with edge removals for ER graph

How Prob. of Connection changes with edge removals for BA graph

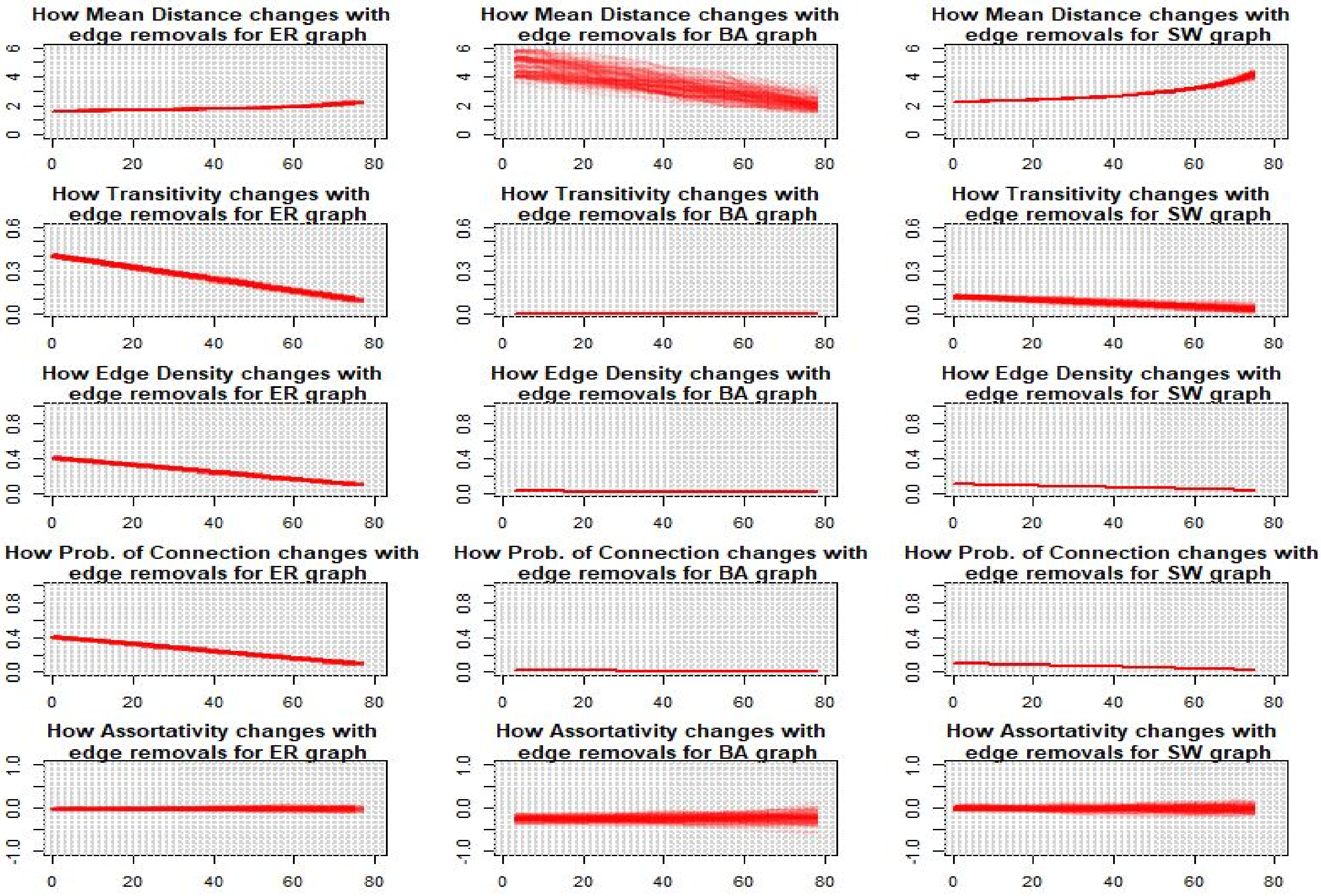
How Prob. of Connection changes wi edge removals for SW graph

How Assortativity changes with edge removals for ER graph

How Assortativity changes with edge removals for BA graph

How Assortativity changes with edge removals for SW graph

Figure 20. Edge Removals, Starting Graph: 100 Nodes, 10 Iterations, 10 Graphs, p = 0.4



How Mean Distance changes with edge removals for ER graph

How Mean Distance changes with edge removals for BA graph

How Mean Distance changes with edge removals for SW graph

How Transitivity changes with edge removals for ER graph

How Transitivity changes with edge removals for BA graph

How Transitivity changes with edge removals for SW graph

How Edge Density changes with edge removals for ER graph

How Edge Density changes with edge removals for BA graph

How Edge Density changes with edge removals for SW graph

How Prob. of Connection changes with edge removals for ER graph

How Prob. of Connection changes with edge removals for BA graph

How Prob. of Connection changes with edge removals for SW graph

How Assortativity changes with edge removals for ER graph

How Assortativity changes with edge removals for BA graph

How Assortativity changes with edge removals for SW graph

Figure 21. Edge Removals, Starting Graph: 100 Nodes, 10 Iterations, 10 Graphs, *p* = 0.5

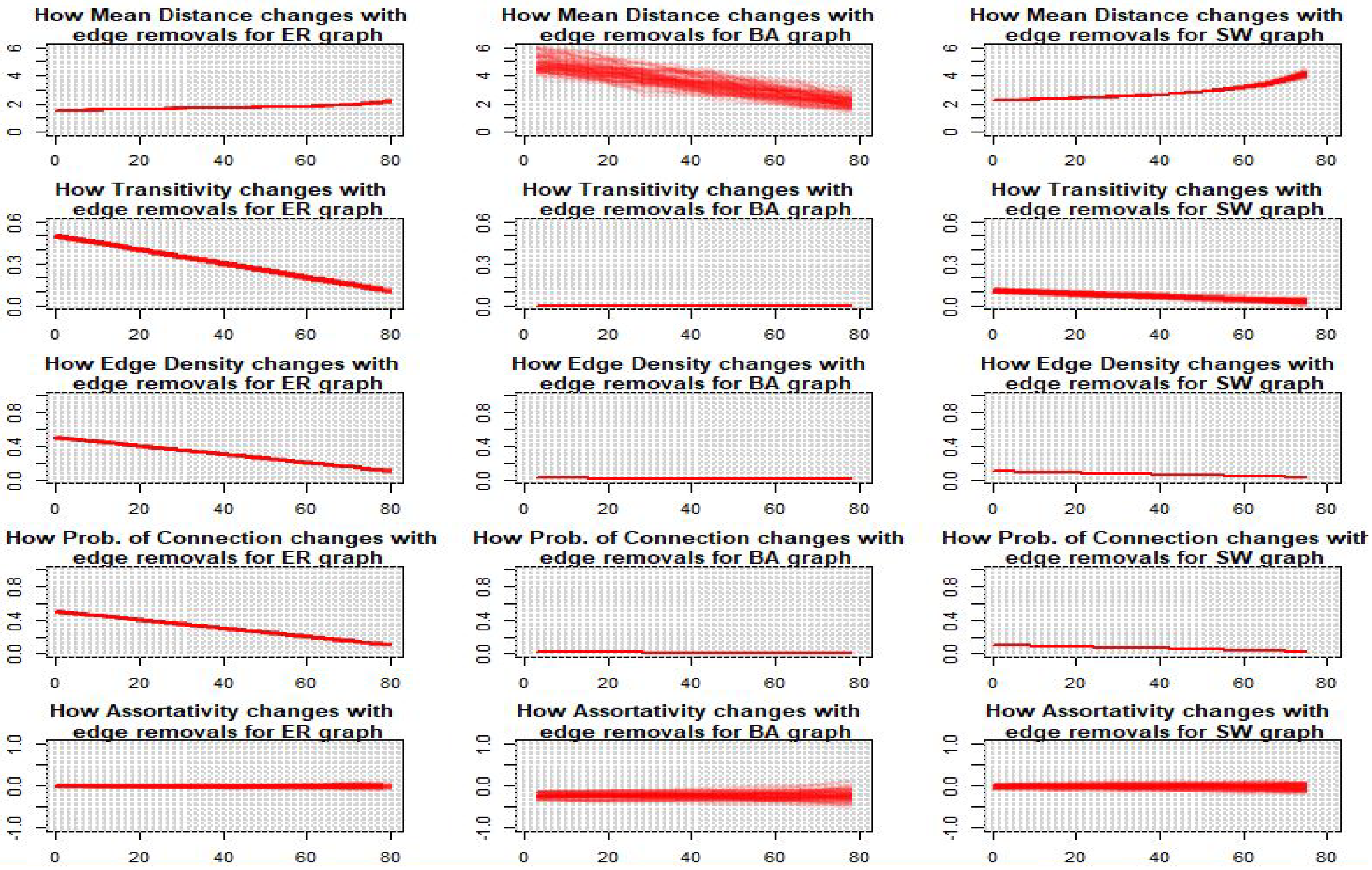


Figure 22. Vertex Removals, Starting Graph: 100 Nodes, 10 Iterations, 10 Graphs, p = 0.1

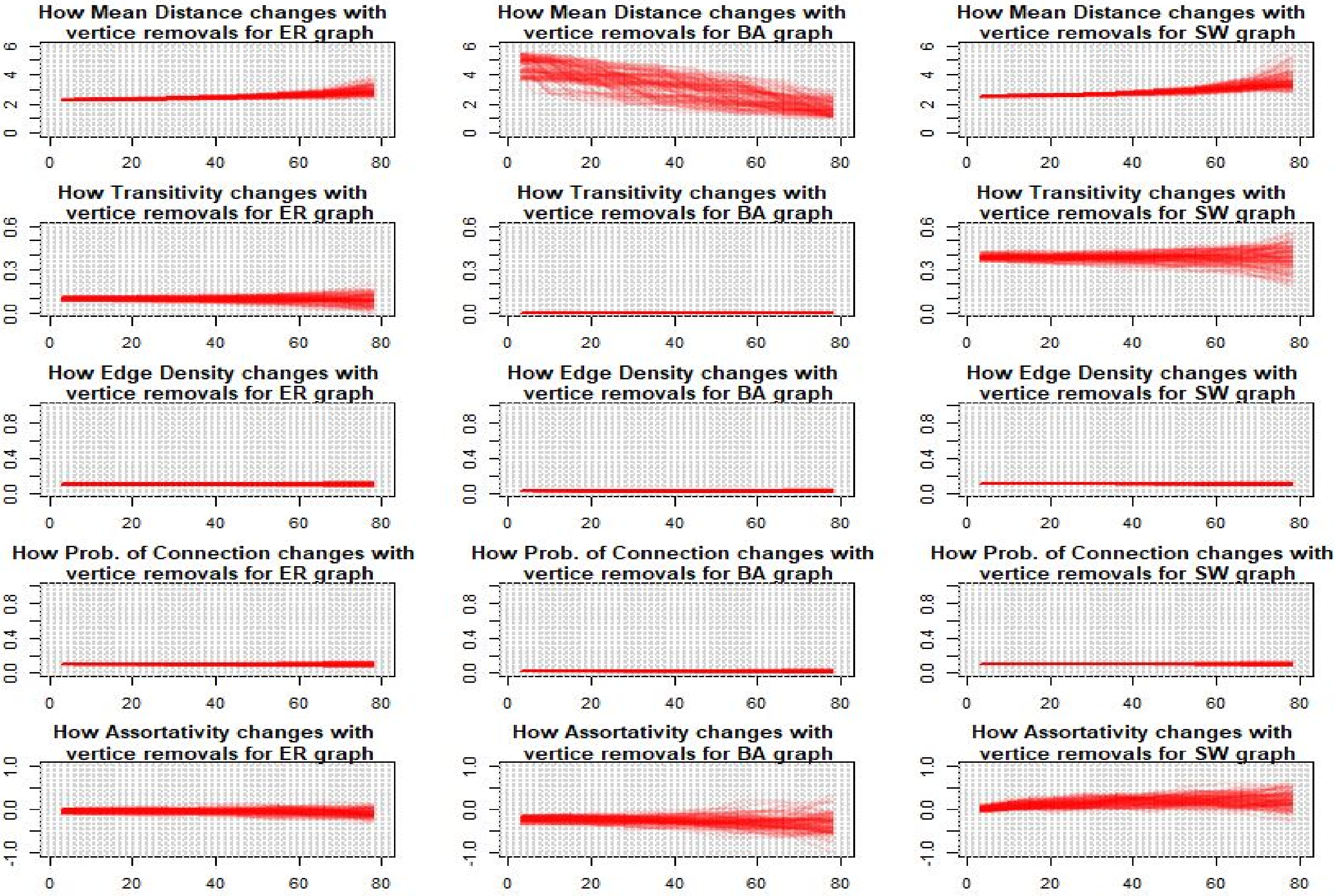


Figure 23. Vertex Removals, Starting Graph: 100 Nodes, 10 Iterations, 10 Graphs, p = 0.2

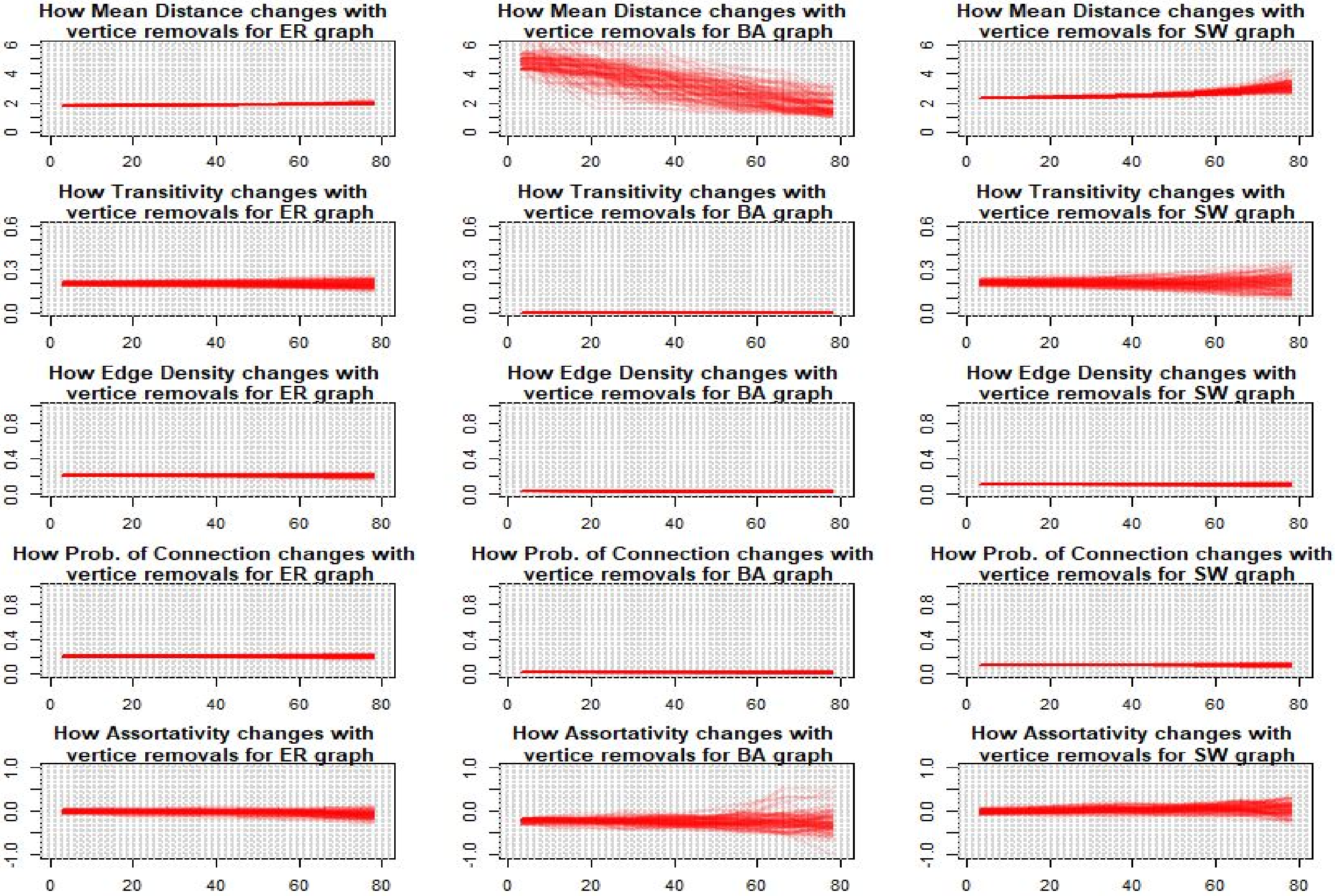


Figure 24. Vertex Removals, Starting Graph: 100 Nodes, 10 Iterations, 10 Graphs, p = 0.3

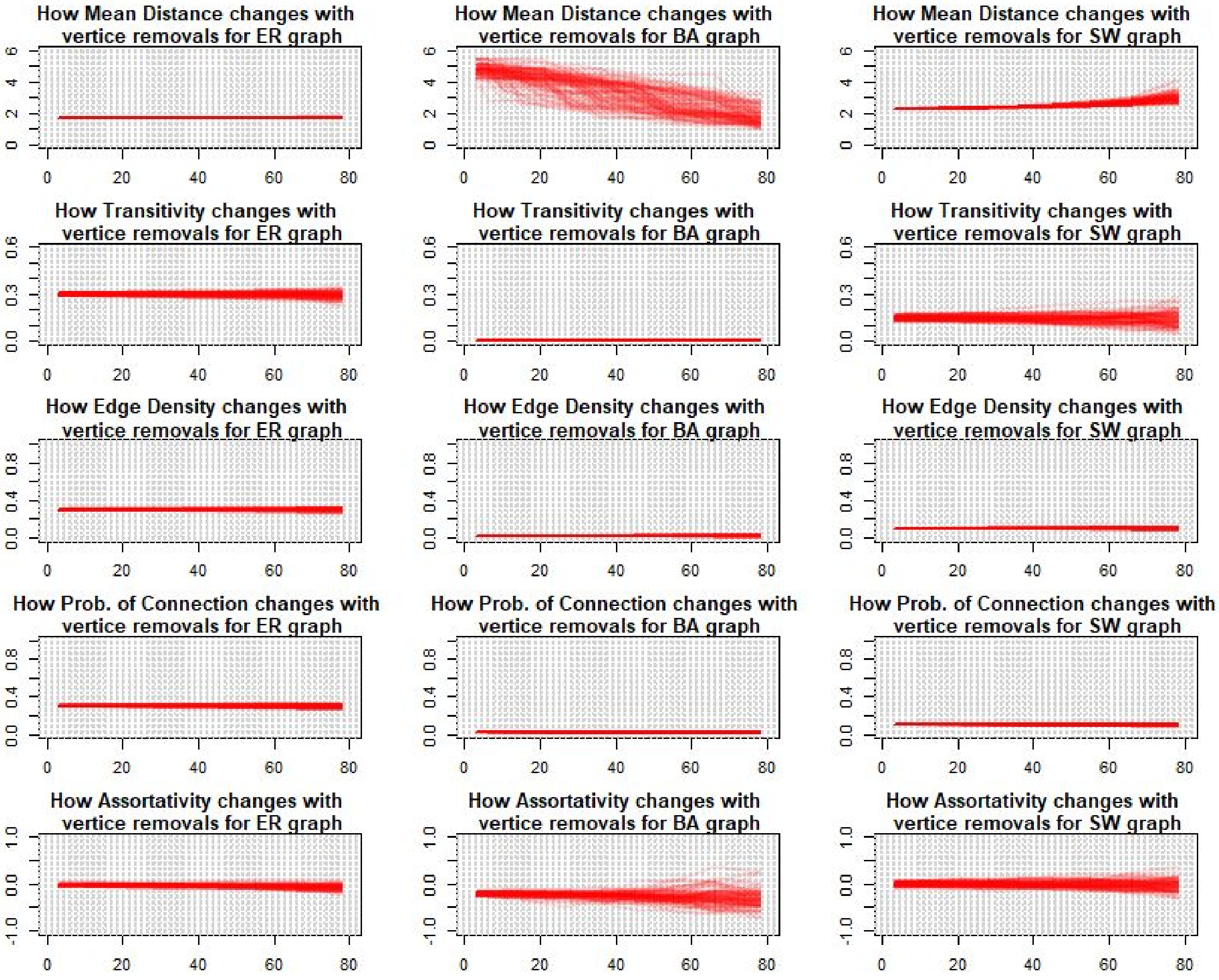


Figure 25. Vertex Removals, Starting Graph: 100 Nodes, 10 Iterations, 10 Graphs, p = 0.4

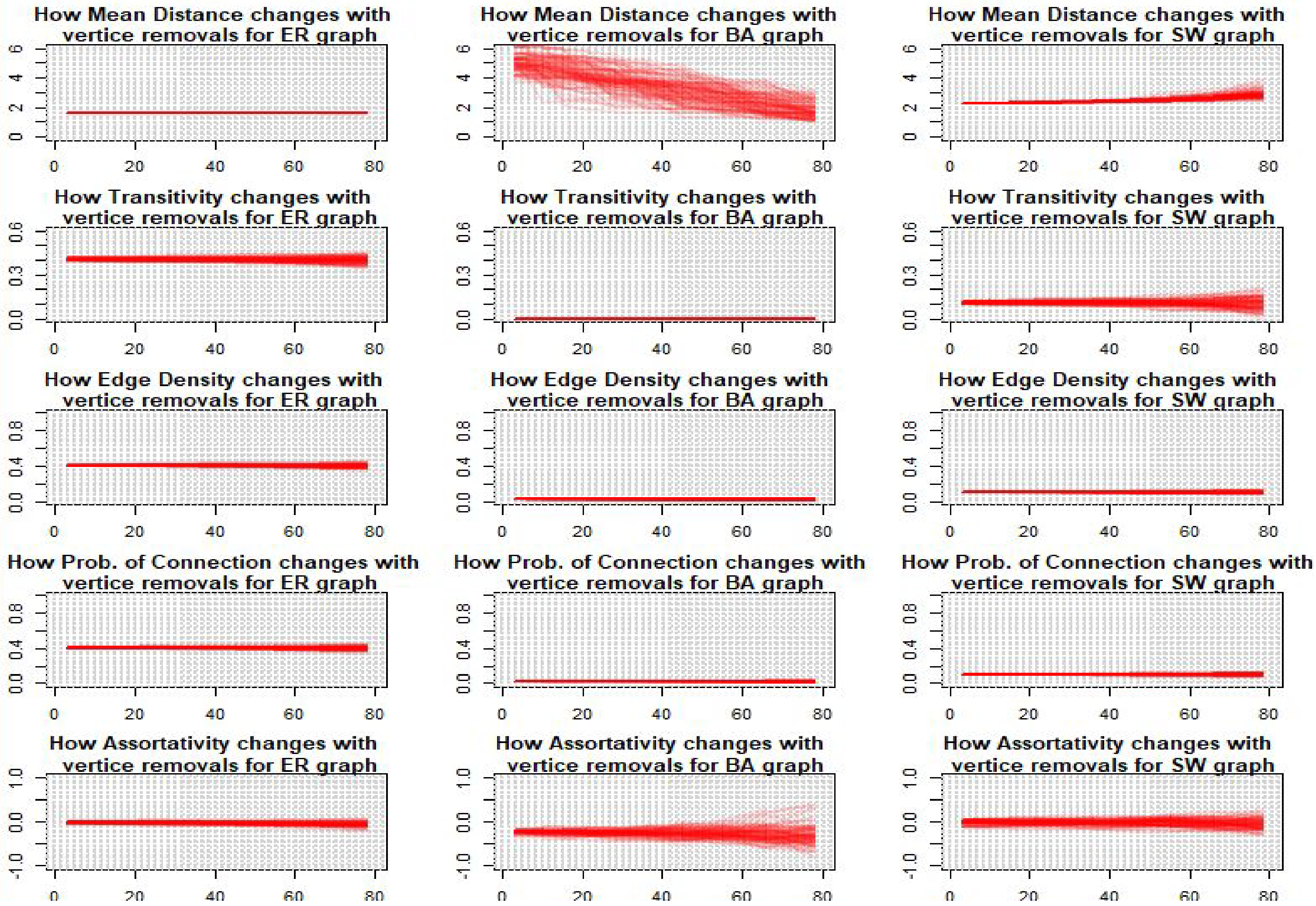


Figure 26. Vertex Removals, Starting Graph: 100 Nodes, 10 Iterations, 10 Graphs, p = 0.5

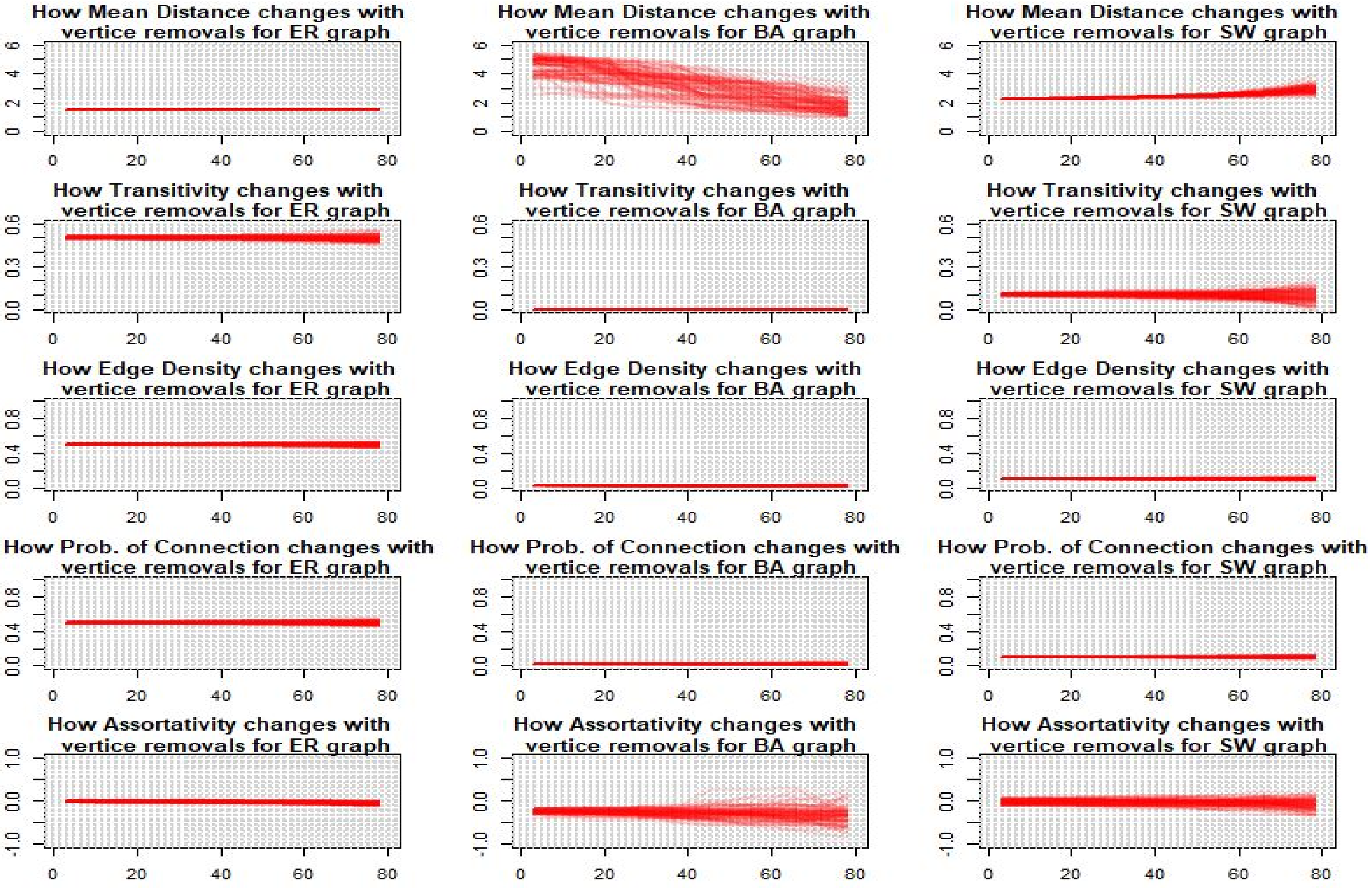


Figure 27. Edge Removals, Starting Graph: 500 Nodes, 10 Iterations, 10 Graphs, p = 0.1

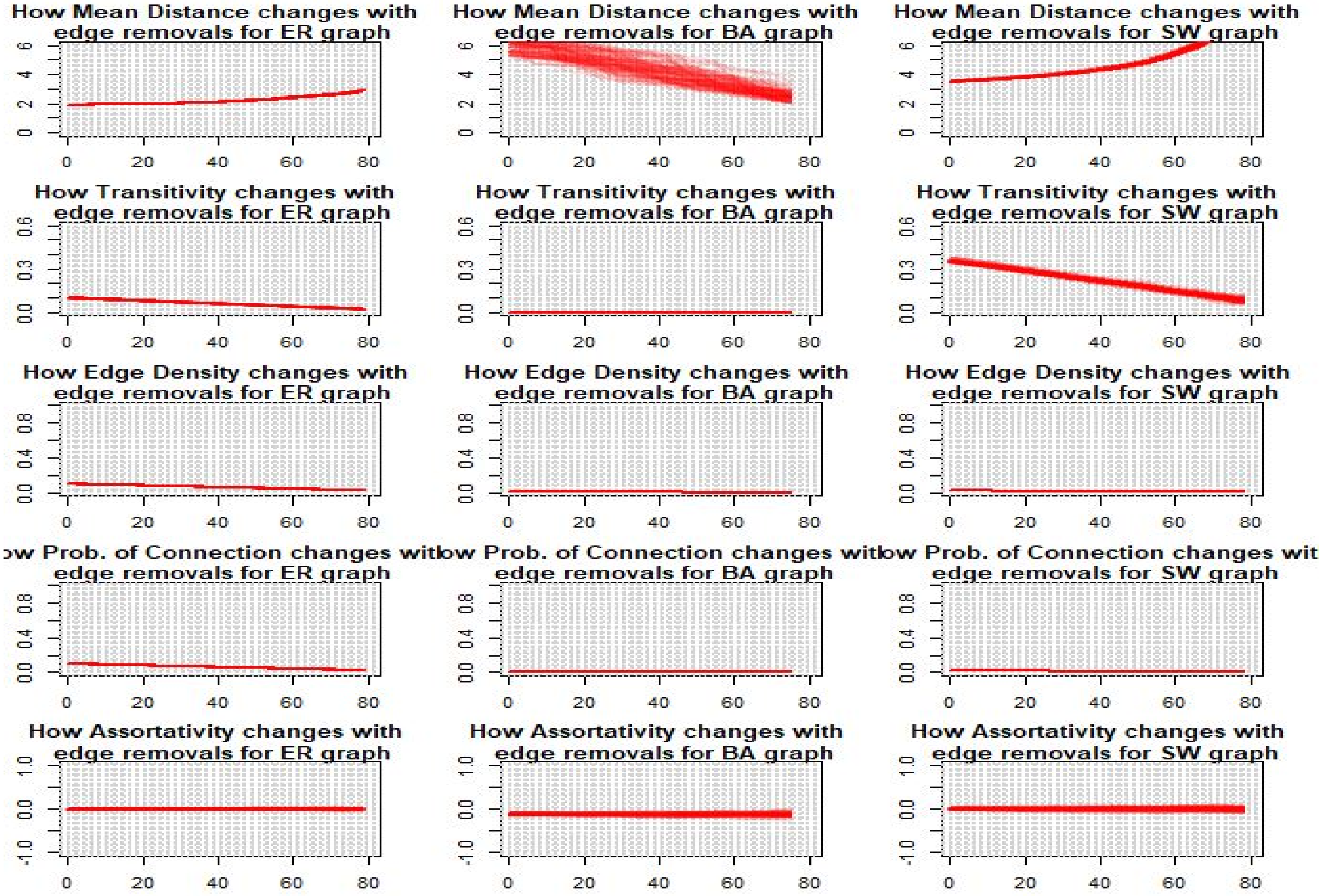
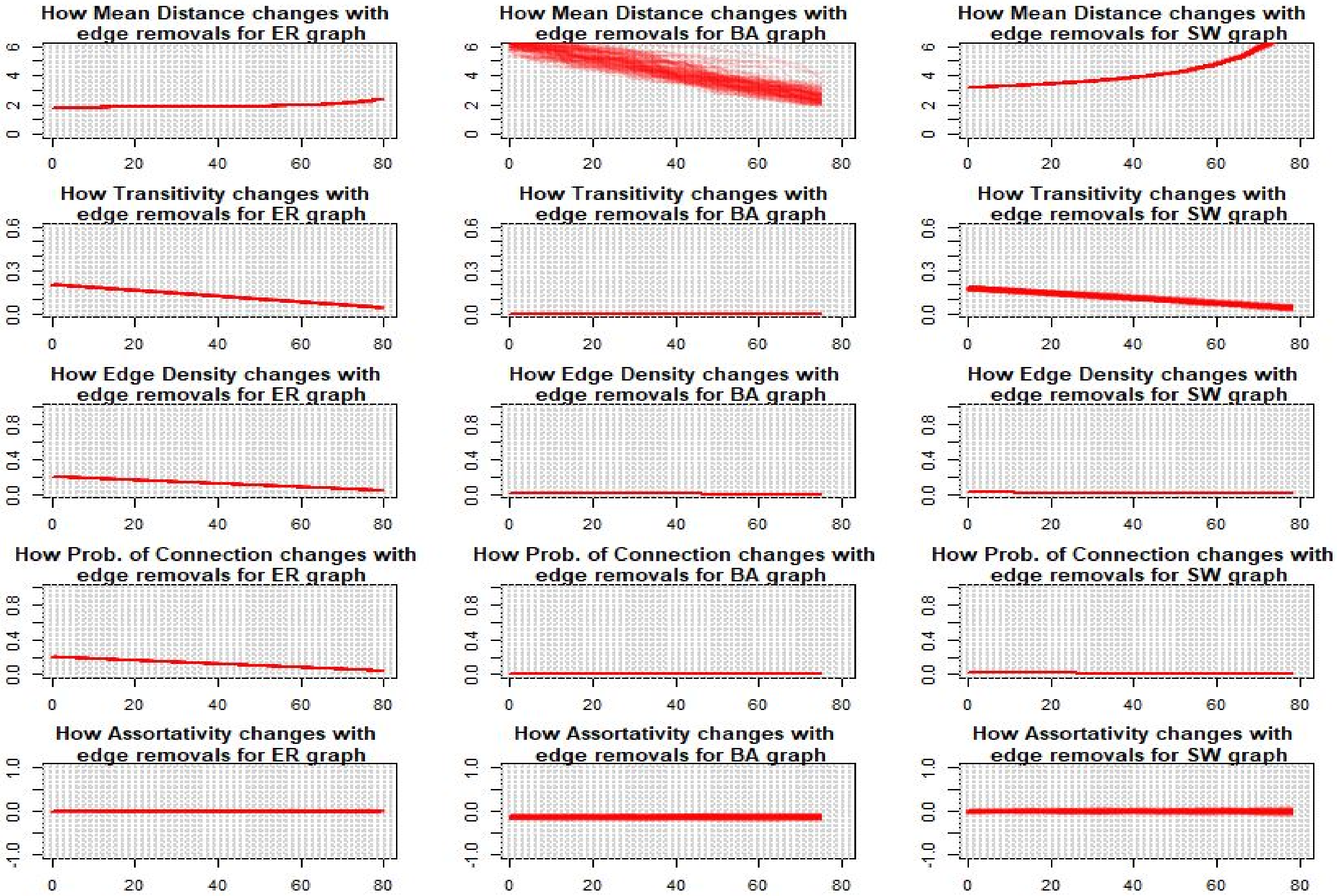


Figure 28. Edge Removals, Starting Graph: 500 Nodes, 10 Iterations, 10 Graphs, *p* = 0.2



How Mean Distance changes with edge removals for ER graph

How Mean Distance changes with edge removals for BA graph

How Mean Distance changes with edge removals for SW graph

How Transitivity changes with edge removals for ER graph

How Transitivity changes with edge removals for BA graph

How Transitivity changes with edge removals for SW graph

How Edge Density changes with edge removals for ER graph

How Edge Density changes with edge removals for BA graph

How Edge Density changes with edge removals for SW graph

How Prob. of Connection changes with edge removals for ER graph

How Prob. of Connection changes with edge removals for BA graph

How Prob. of Connection changes with edge removals for SW graph

How Assortativity changes with edge removals for ER graph

How Assortativity changes with edge removals for BA graph

How Assortativity changes with edge removals for SW graph

Figure 29. Edge Removals, Starting Graph: 500 Nodes, 10 Iterations, 10 Graphs, $p = 0.3$

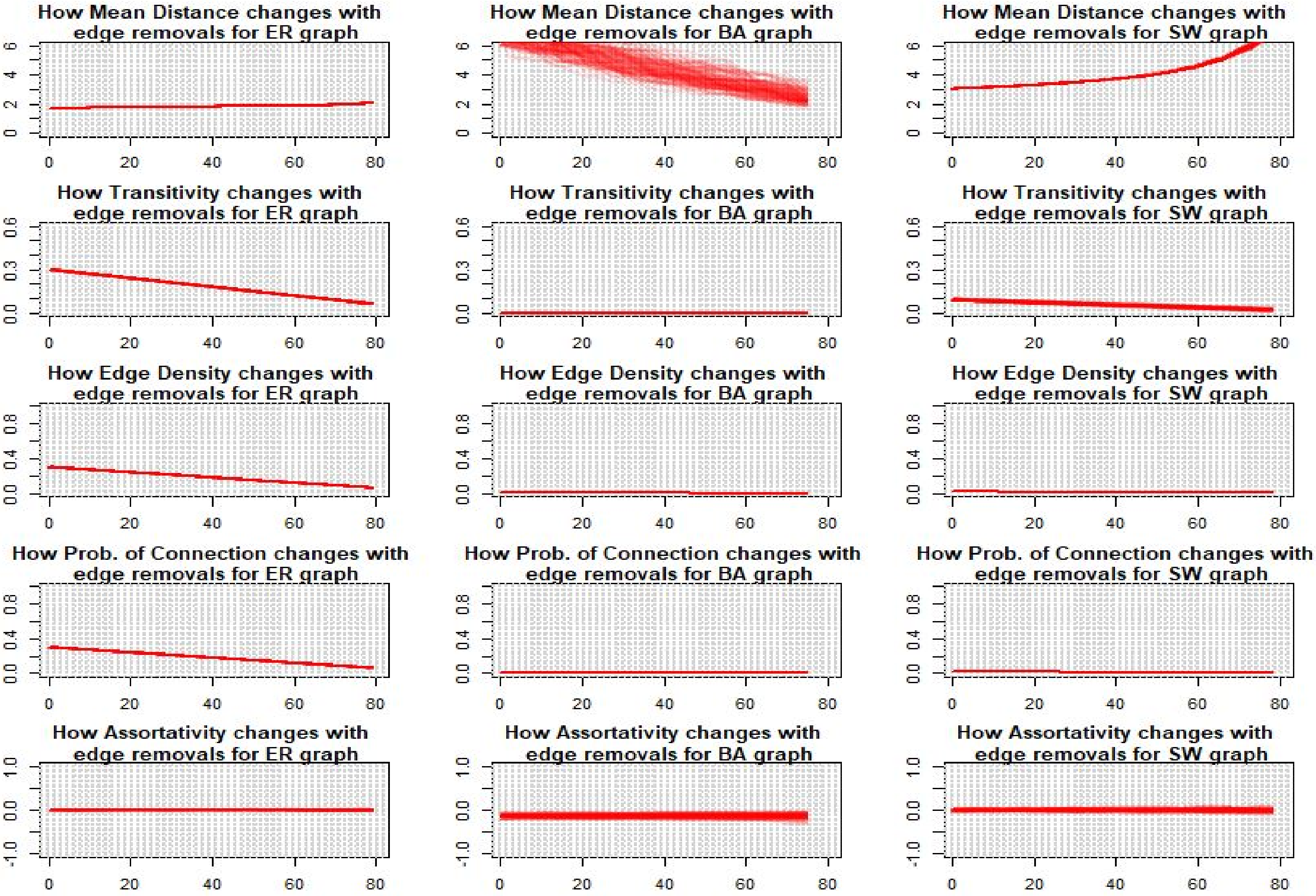


Figure 30. Edge Removals, Starting Graph: 500 Nodes, 10 Iterations, 10 Graphs, p = 0.4

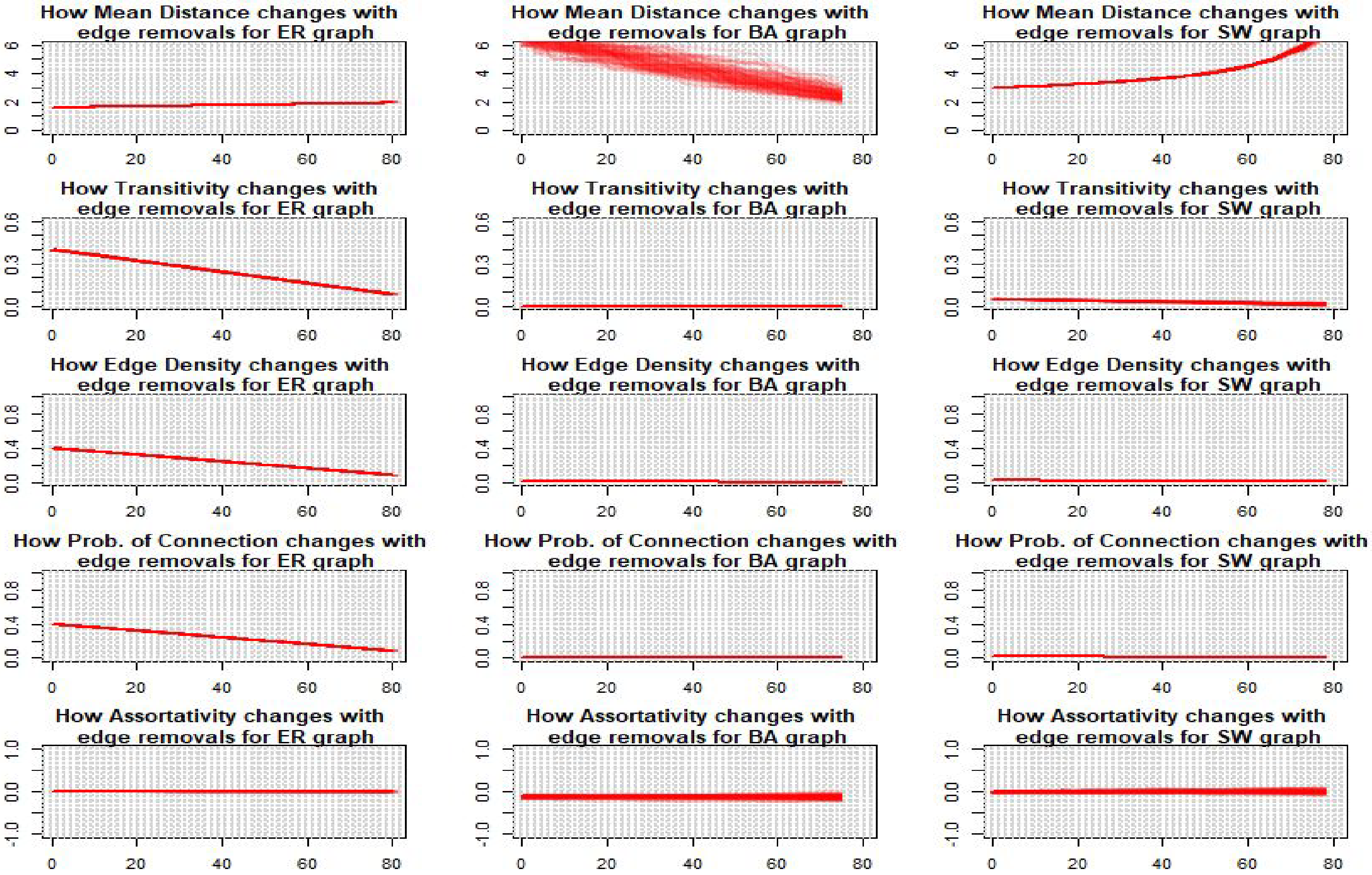
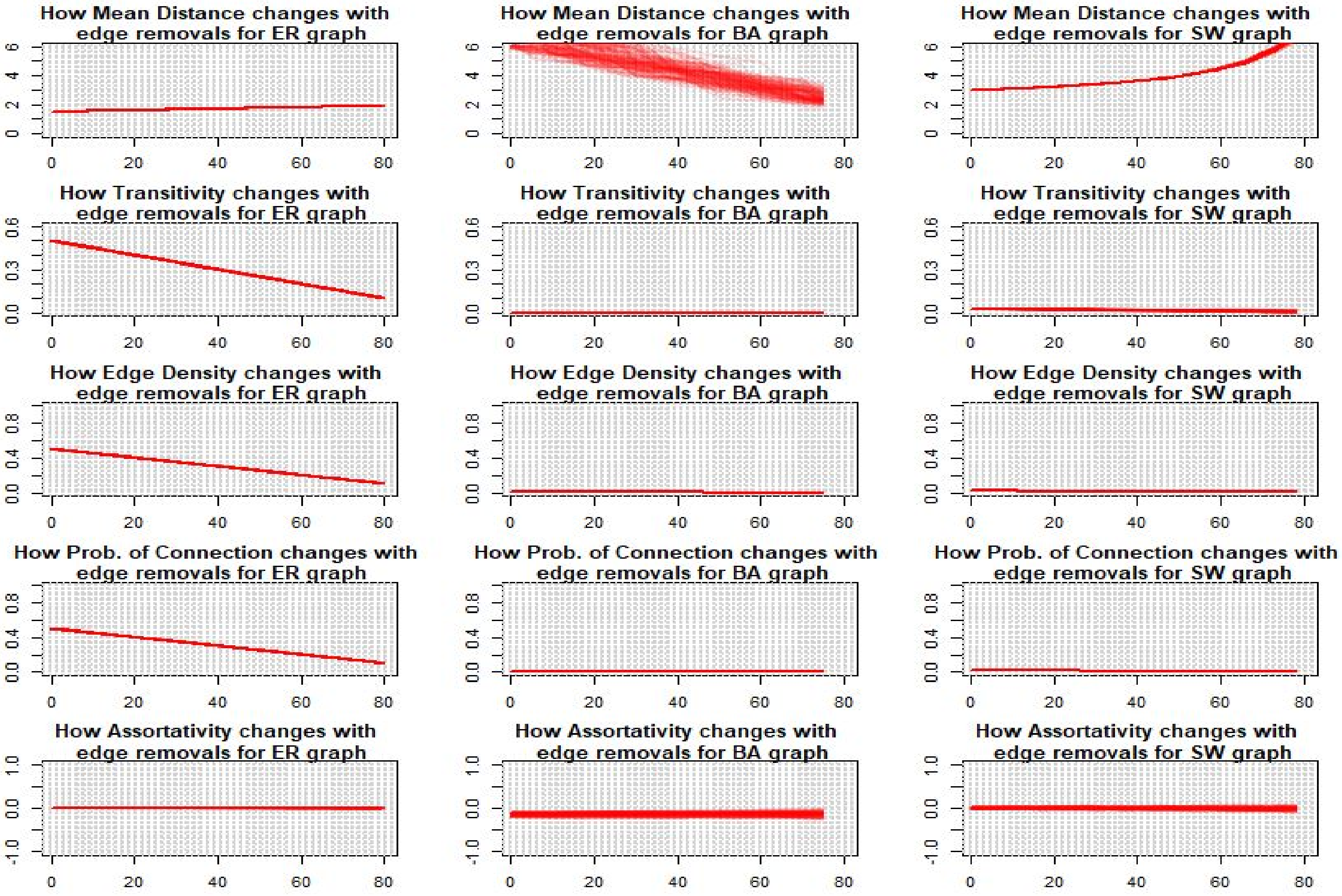


Figure 31. Edge Removals, Starting Graph: 500 Nodes, 10 Iterations, 10 Graphs, p = 0.5



How Mean Distance changes with edge removals for ER graph

How Mean Distance changes with edge removals for BA graph

How Mean Distance changes with edge removals for SW graph

How Transitivity changes with edge removals for ER graph

How Transitivity changes with edge removals for BA graph

How Transitivity changes with edge removals for SW graph

How Edge Density changes with edge removals for ER graph

How Edge Density changes with edge removals for BA graph

How Edge Density changes with edge removals for SW graph

How Prob. of Connection changes with edge removals for ER graph

How Prob. of Connection changes with edge removals for BA graph

How Prob. of Connection changes with edge removals for SW graph

How Assortativity changes with edge removals for ER graph

How Assortativity changes with edge removals for BA graph

How Assortativity changes with edge removals for SW graph

Figure 32. Vertex Removals, Starting Graph: 500 Nodes, 10 Iterations, 10 Graphs, p = 0.1

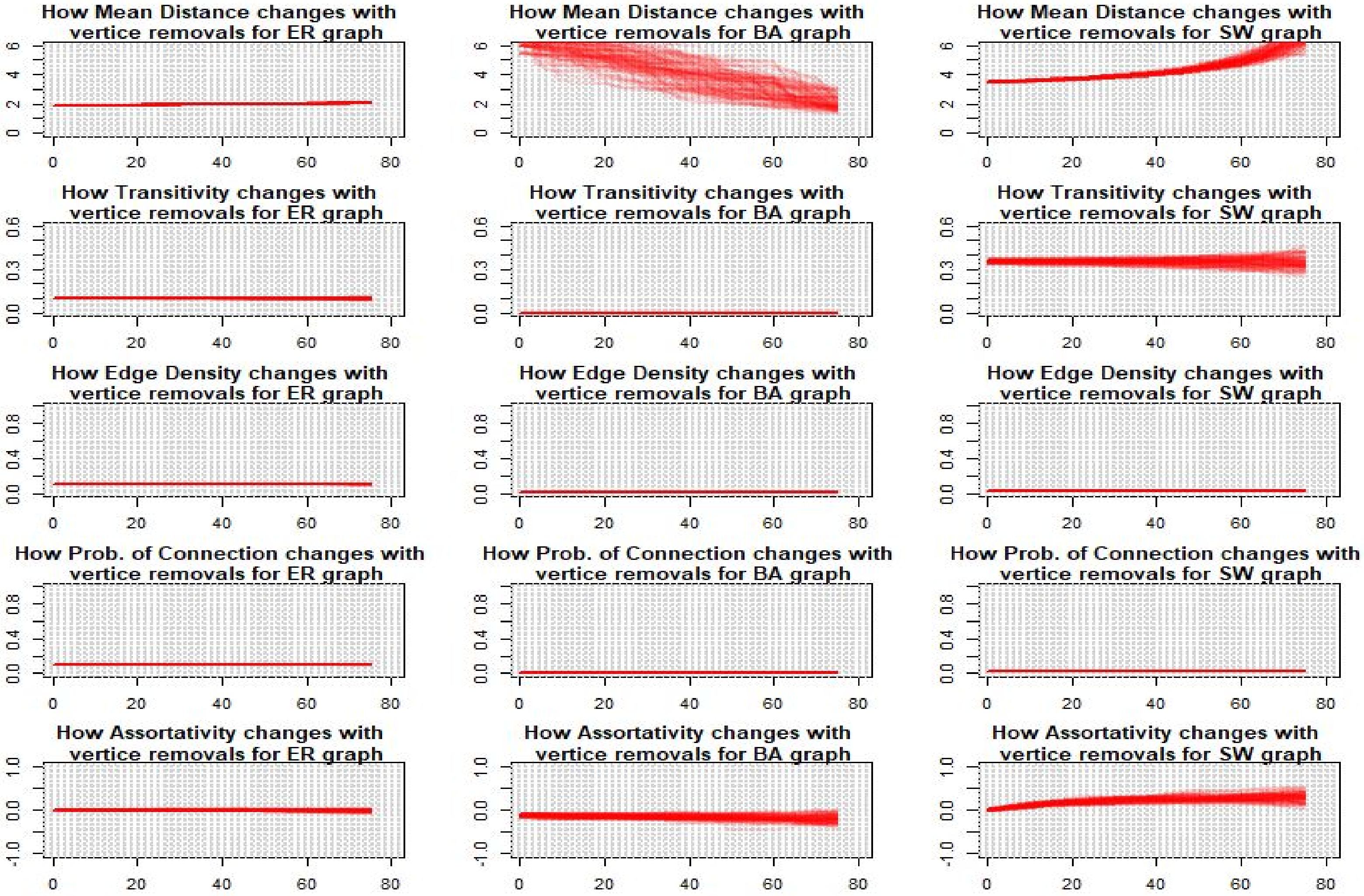


Figure 33. Vertex Removals, Starting Graph: 500 Nodes, 10 Iterations, 10 Graphs, p = 0.2

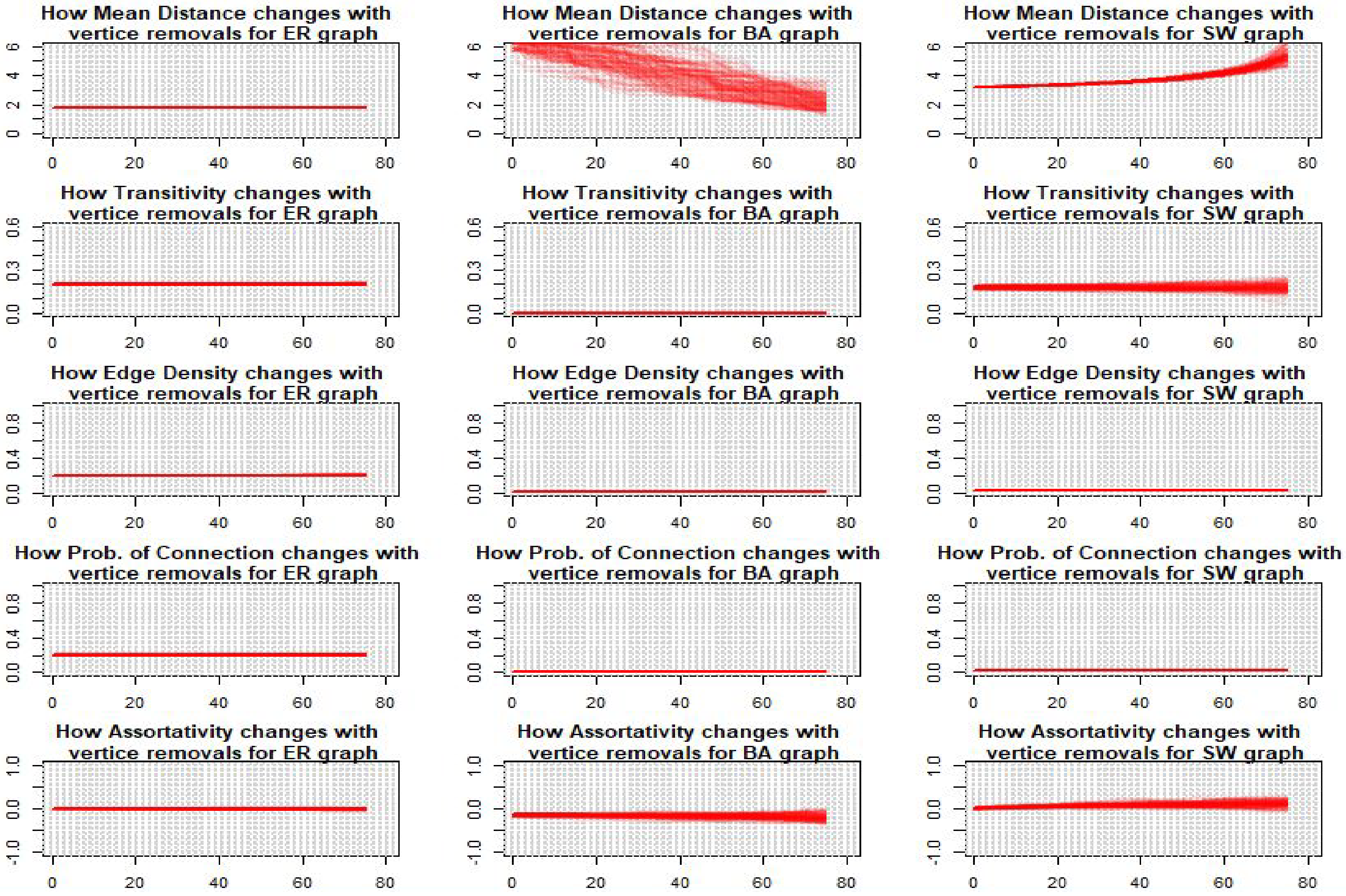
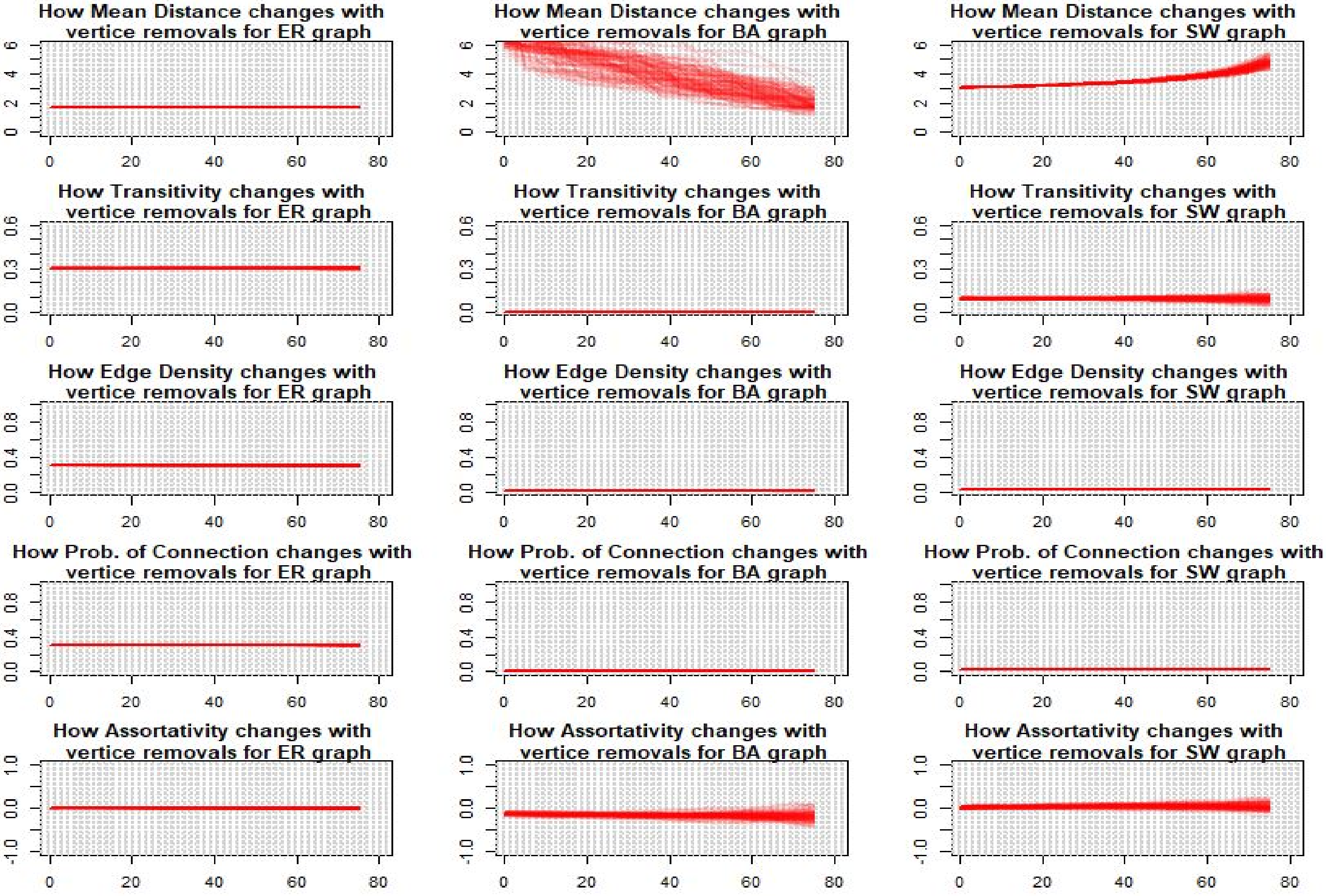


Figure 34. Vertex Removals, Starting Graph: 500 Nodes, 10 Iterations, 10 Graphs, p = 0.3



How Mean Distance changes with vertice removals for ER graph

How Mean Distance changes with vertice removals for BA graph

How Mean Distance changes with vertice removals for SW graph

How Transitivity changes with vertice removals for ER graph

How Transitivity changes with vertice removals for BA graph

How Transitivity changes with vertice removals for SW graph

How Edge Density changes with vertice removals for ER graph

How Edge Density changes with vertice removals for BA graph

How Edge Density changes with vertice removals for SW graph

How Prob. of Connection changes with vertice removals for ER graph

How Prob. of Connection changes with vertice removals for BA graph

How Prob. of Connection changes with vertice removals for SW graph

How Assortativity changes with vertice removals for ER graph

How Assortativity changes with vertice removals for BA graph

How Assortativity changes with vertice removals for SW graph

Figure 35. Vertex Removals, Starting Graph: 500 Nodes, 10 Iterations, 10 Graphs, p = 0.4

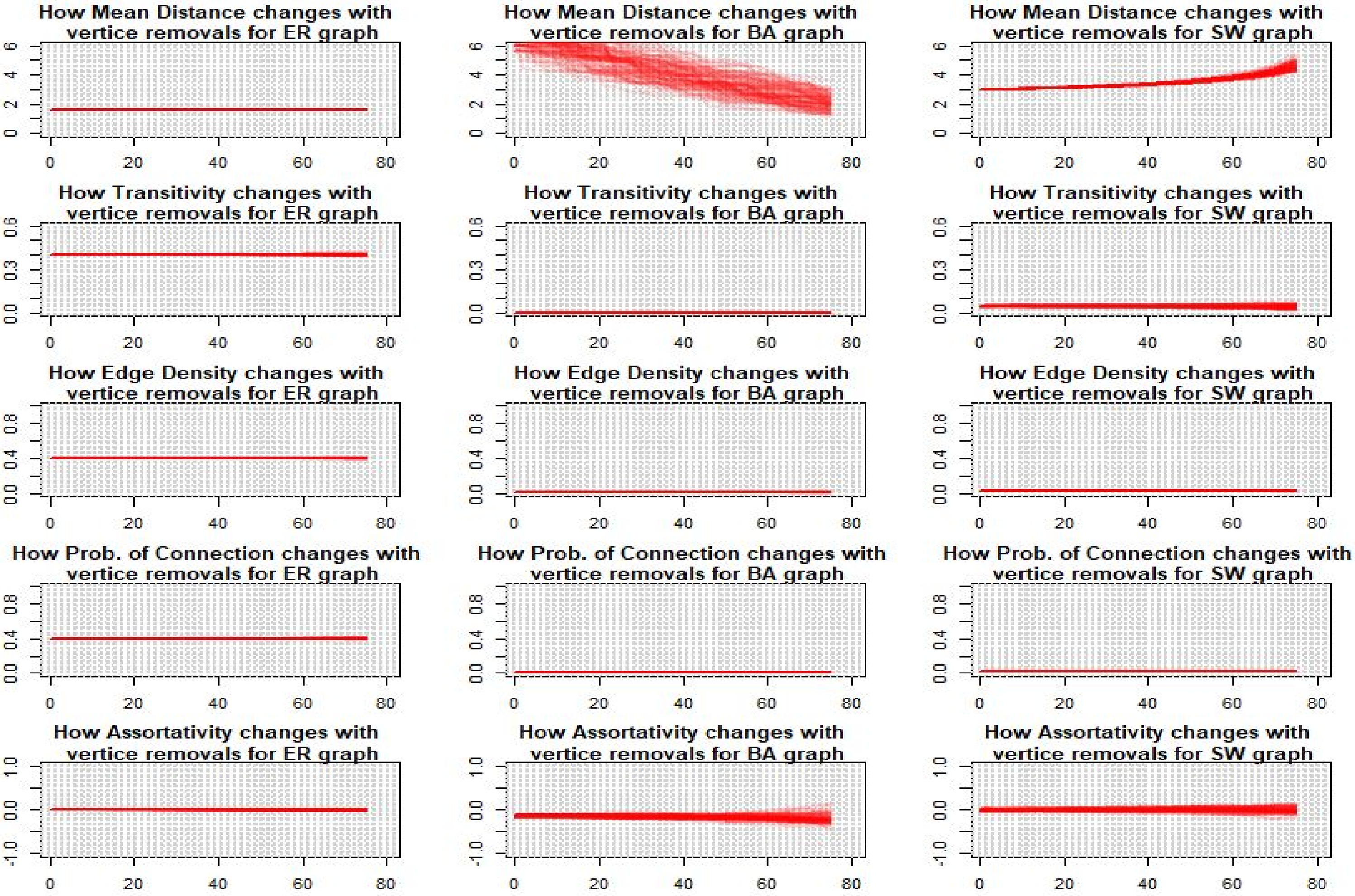


Figure 36. Vertex Removals, Starting Graph: 500 Nodes, 10 Iterations, 10 Graphs, p = 0.5

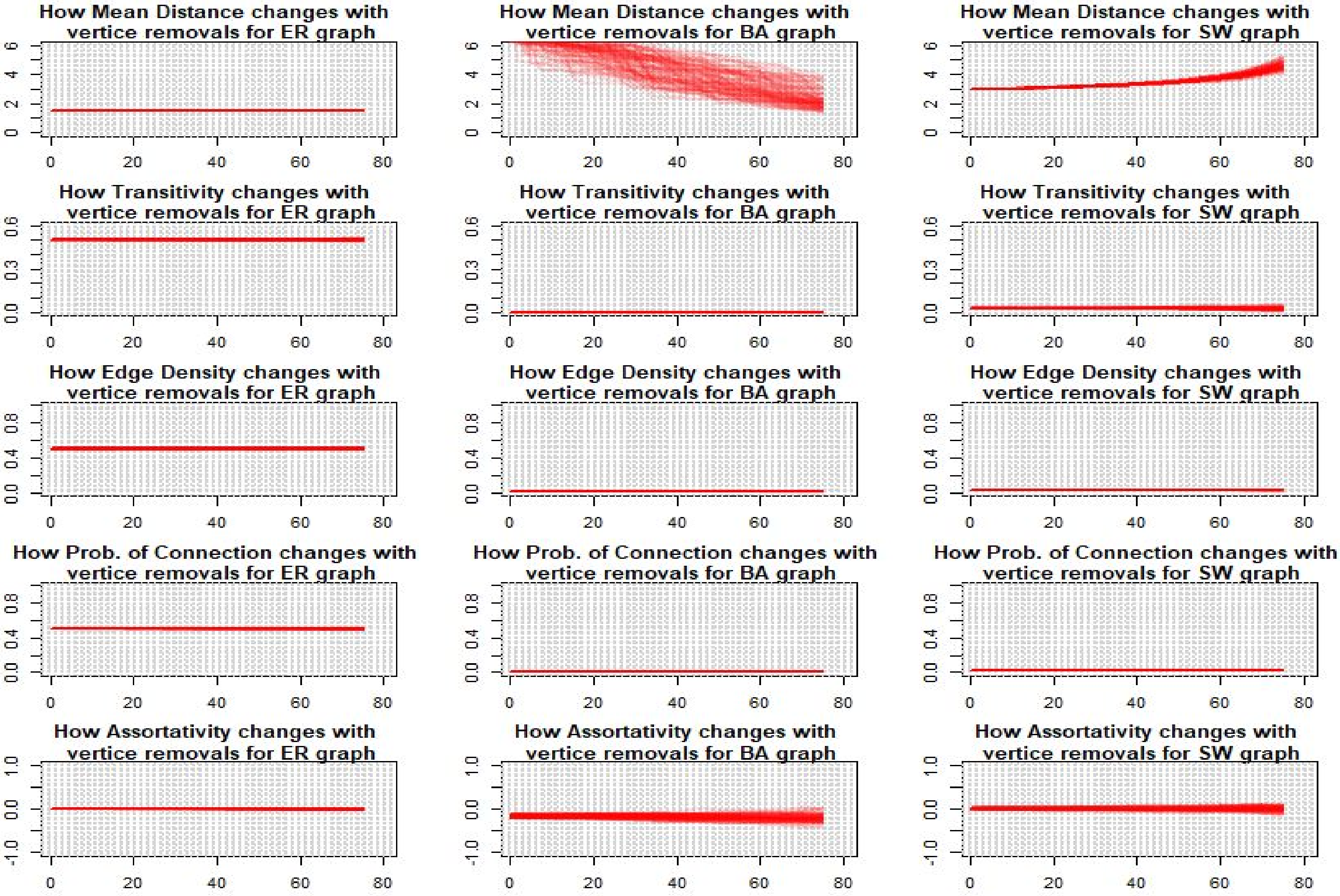


Figure 37. Edge Removals, Starting Graph: 1000 Nodes, 10 Iterations, 10 Graphs, *p* = 0.1

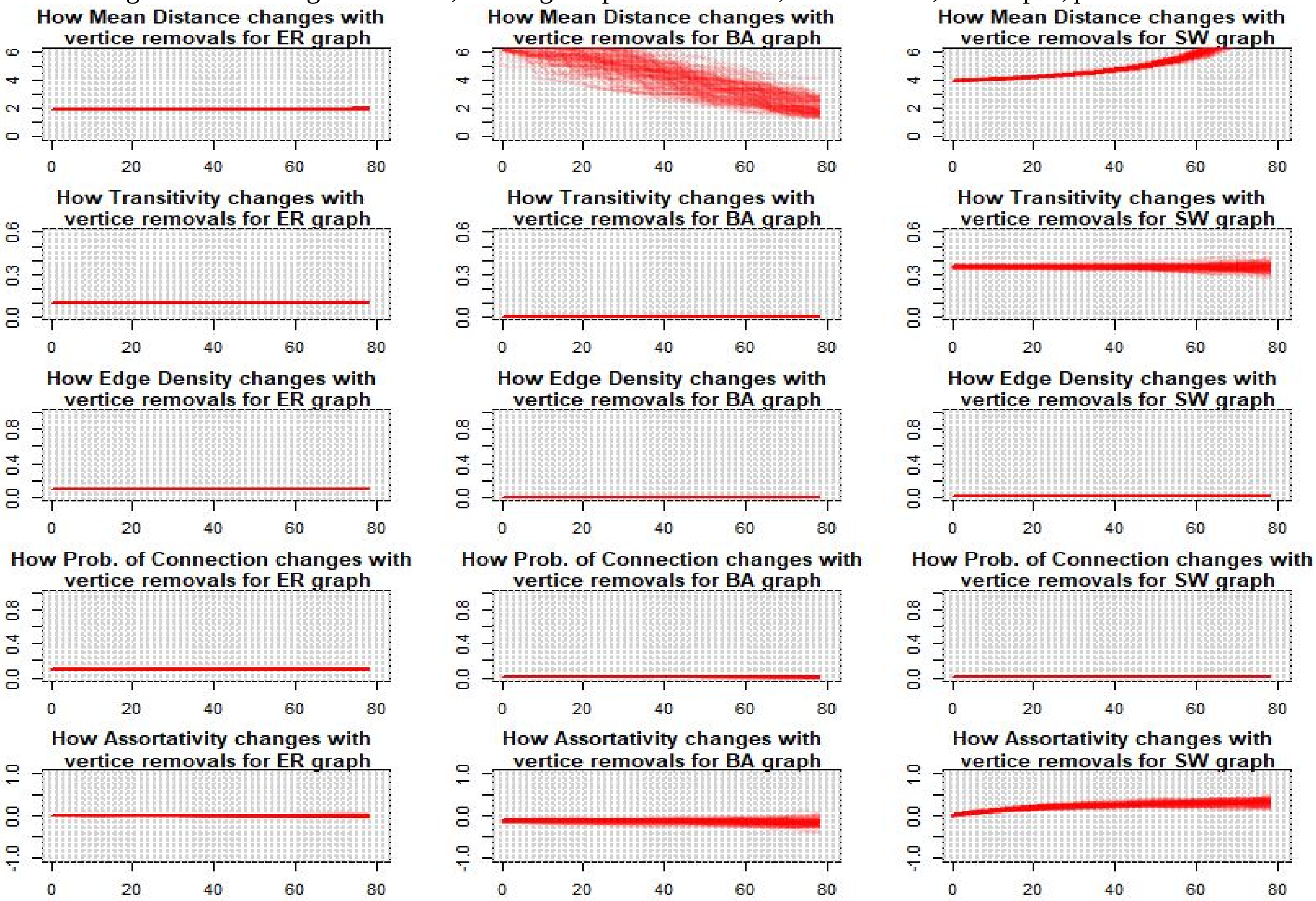


Figure 38. Edge Removals, Starting Graph: 1000 Nodes, 10 Iterations, 10 Graphs, p = 0.2

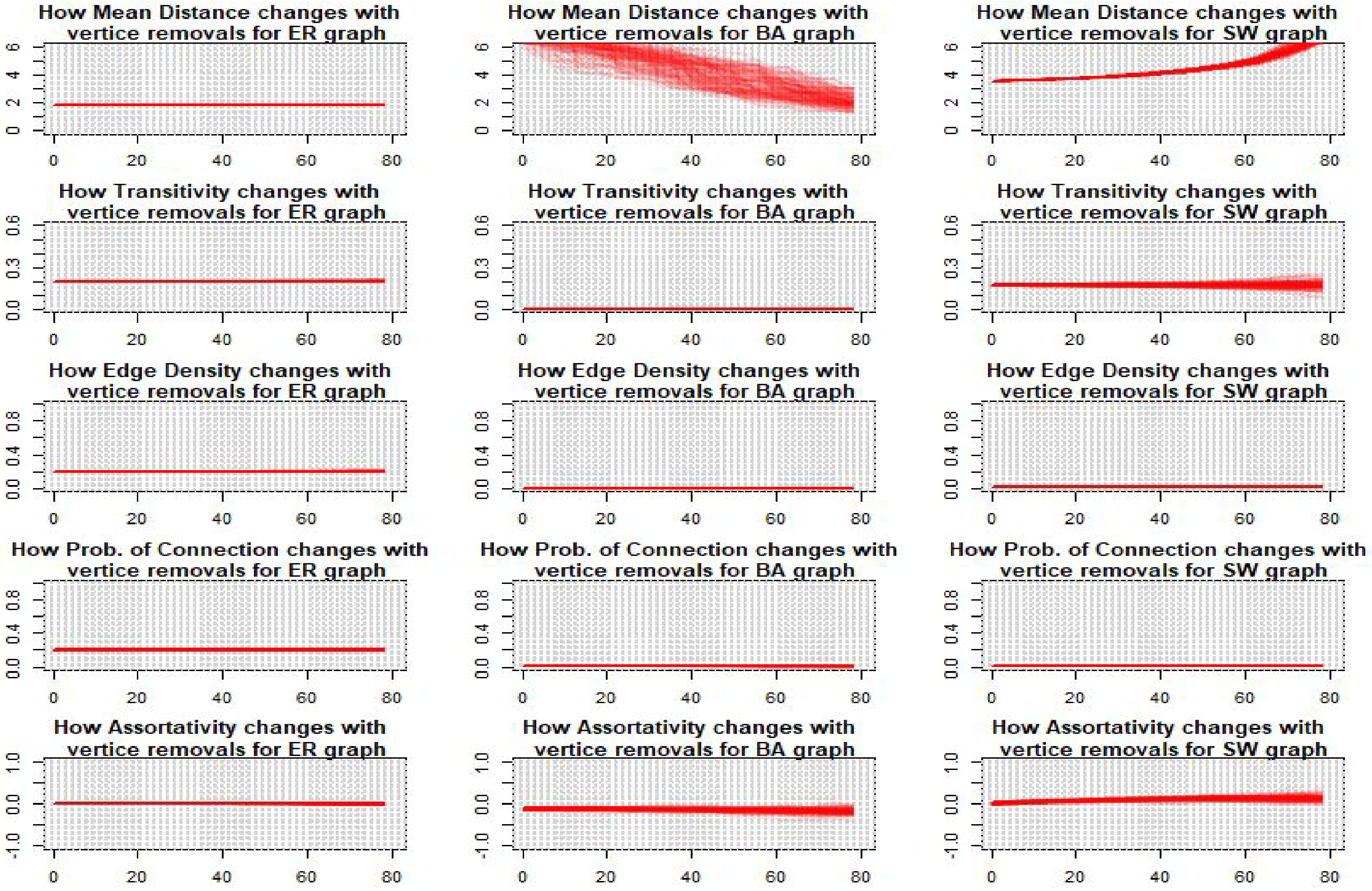


Figure 39. Edge Removals, Starting Graph: 1000 Nodes, 10 Iterations, 10 Graphs, p = 0.3

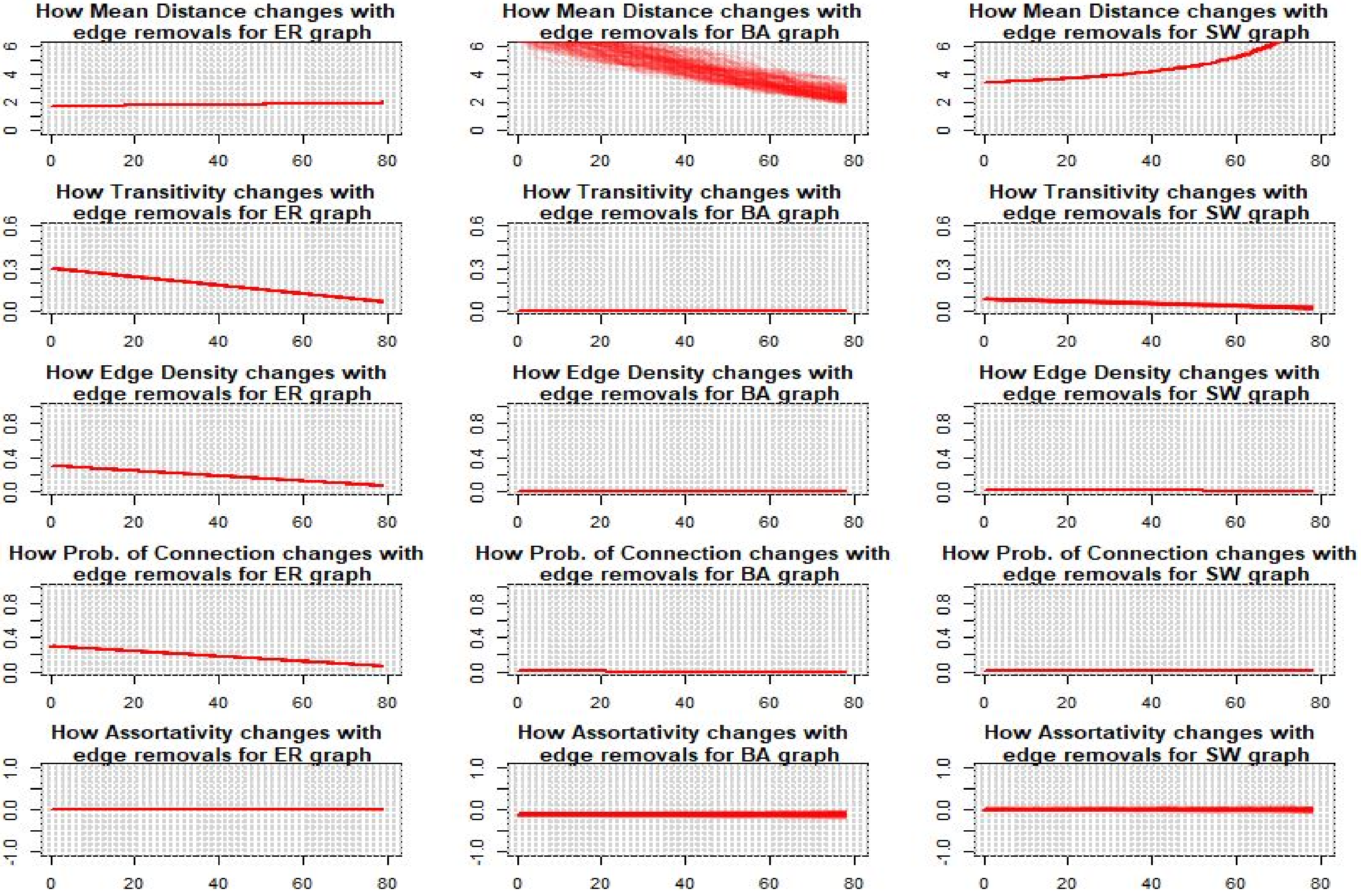


Figure 40. Edge Removals, Starting Graph: 1000 Nodes, 10 Iterations, 10 Graphs, *p* = 0.4

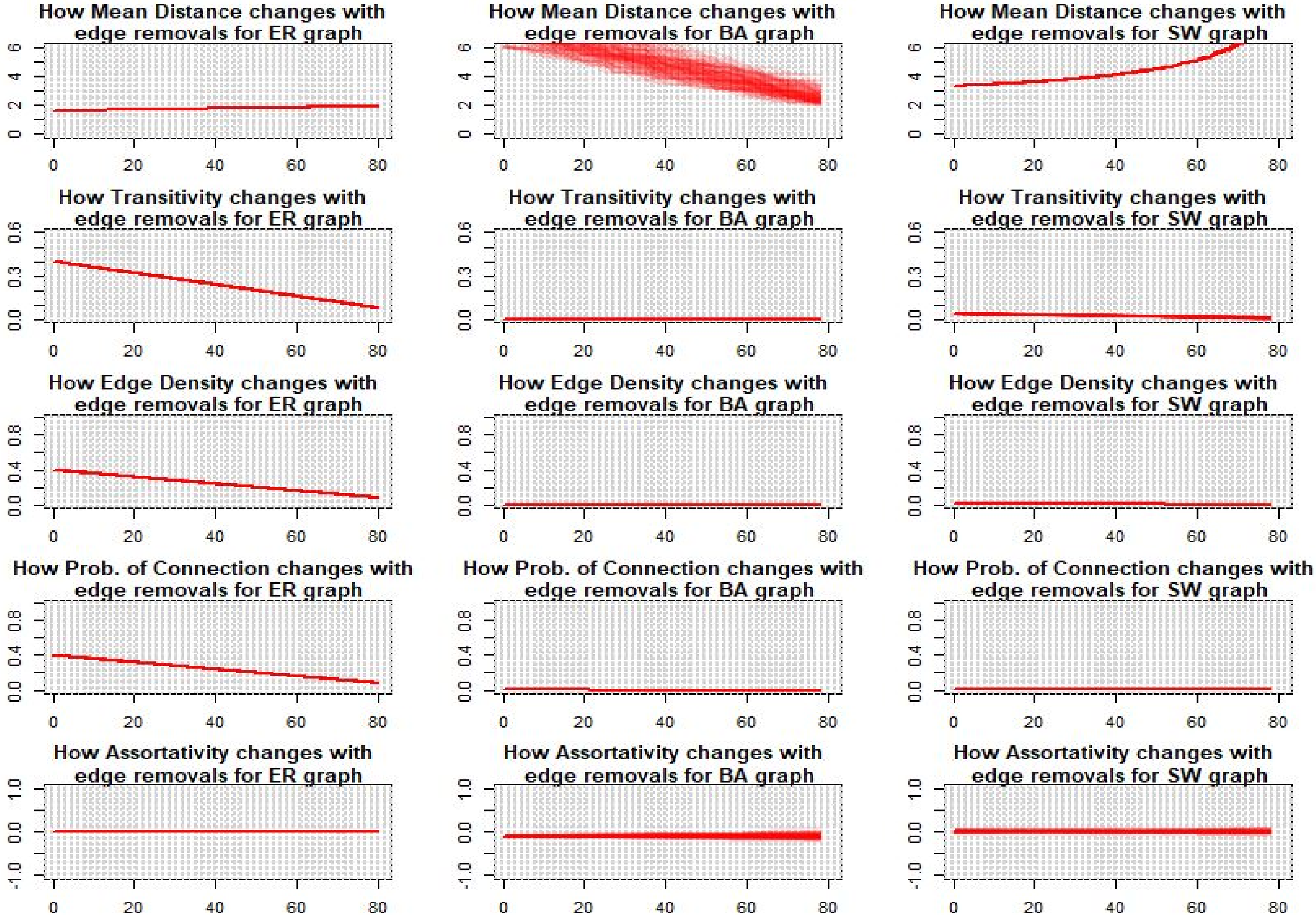


Figure 41. Edge Removals, Starting Graph: 1000 Nodes, 10 Iterations, 10 Graphs, *p* = 0.5

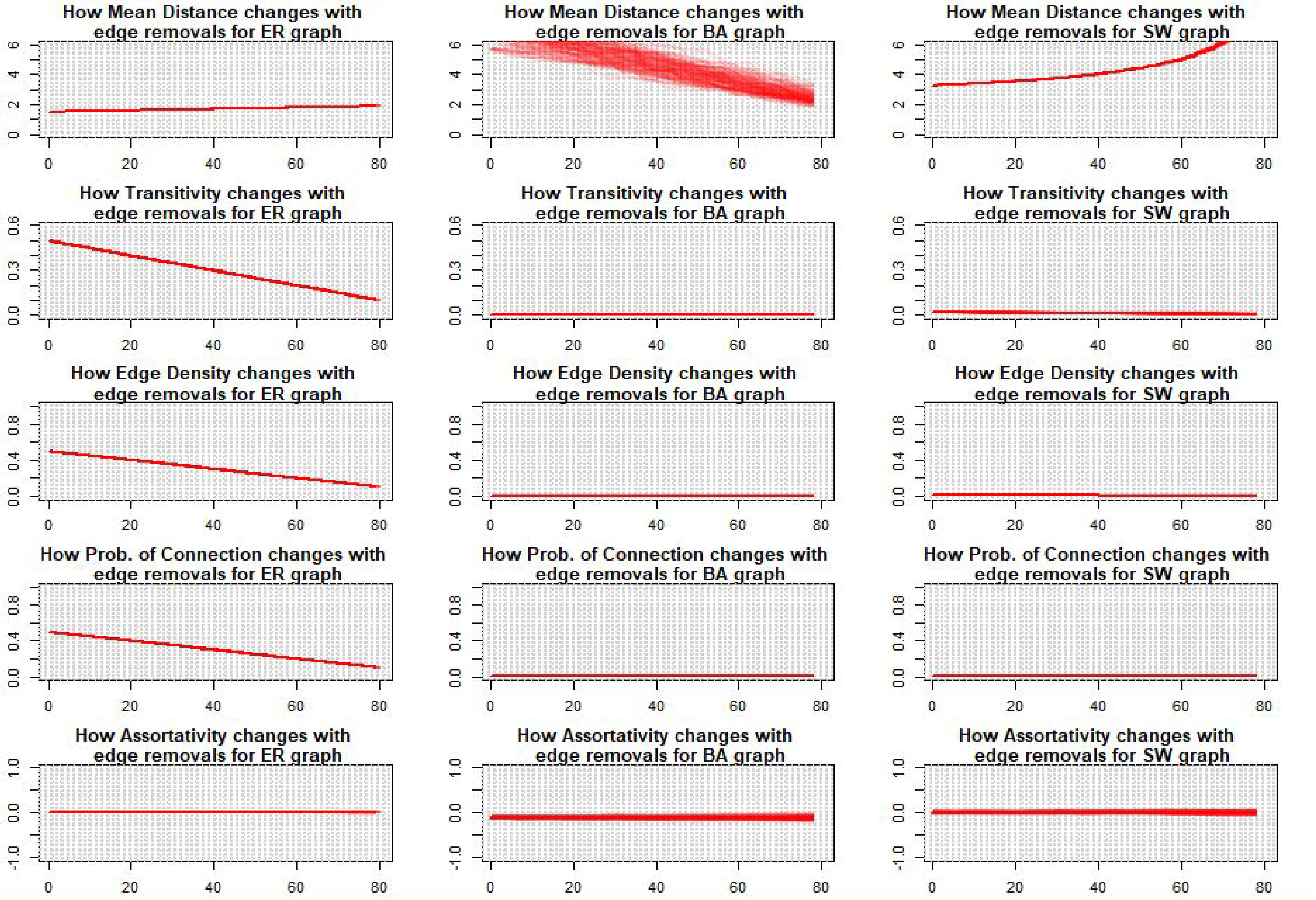
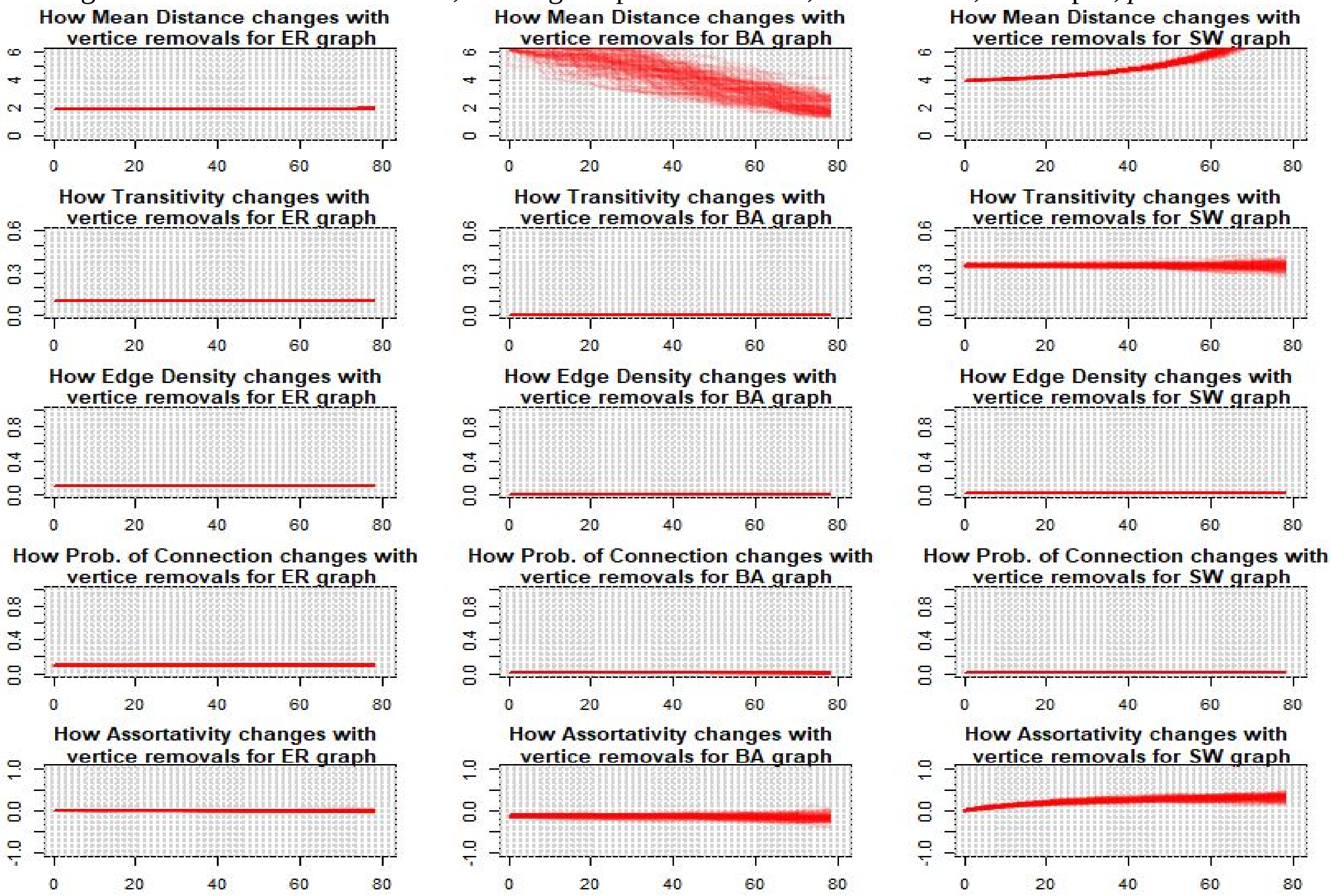


Figure 42. Vertex Removals, Starting Graph: 1000 Nodes, 10 Iterations, 10 Graphs, $p = 0.1$



How Mean Distance changes with vertice removals for ER graph

How Mean Distance changes with vertice removals for BA graph

How Mean Distance changes with vertice removals for SW graph

How Transitivity changes with vertice removals for ER graph

How Transitivity changes with vertice removals for BA graph

How Transitivity changes with vertice removals for SW graph

How Edge Density changes with vertice removals for ER graph

How Edge Density changes with vertice removals for BA graph

How Edge Density changes with vertice removals for SW graph

How Prob. of Connection changes with vertice removals for ER graph

How Prob. of Connection changes with vertice removals for BA graph

How Prob. of Connection changes with vertice removals for SW graph

How Assortativity changes with vertice removals for ER graph

How Assortativity changes with vertice removals for BA graph

How Assortativity changes with vertice removals for SW graph

Figure 43. Vertex Removals, Starting Graph: 1000 Nodes, 10 Iterations, 10 Graphs, p = 0.2

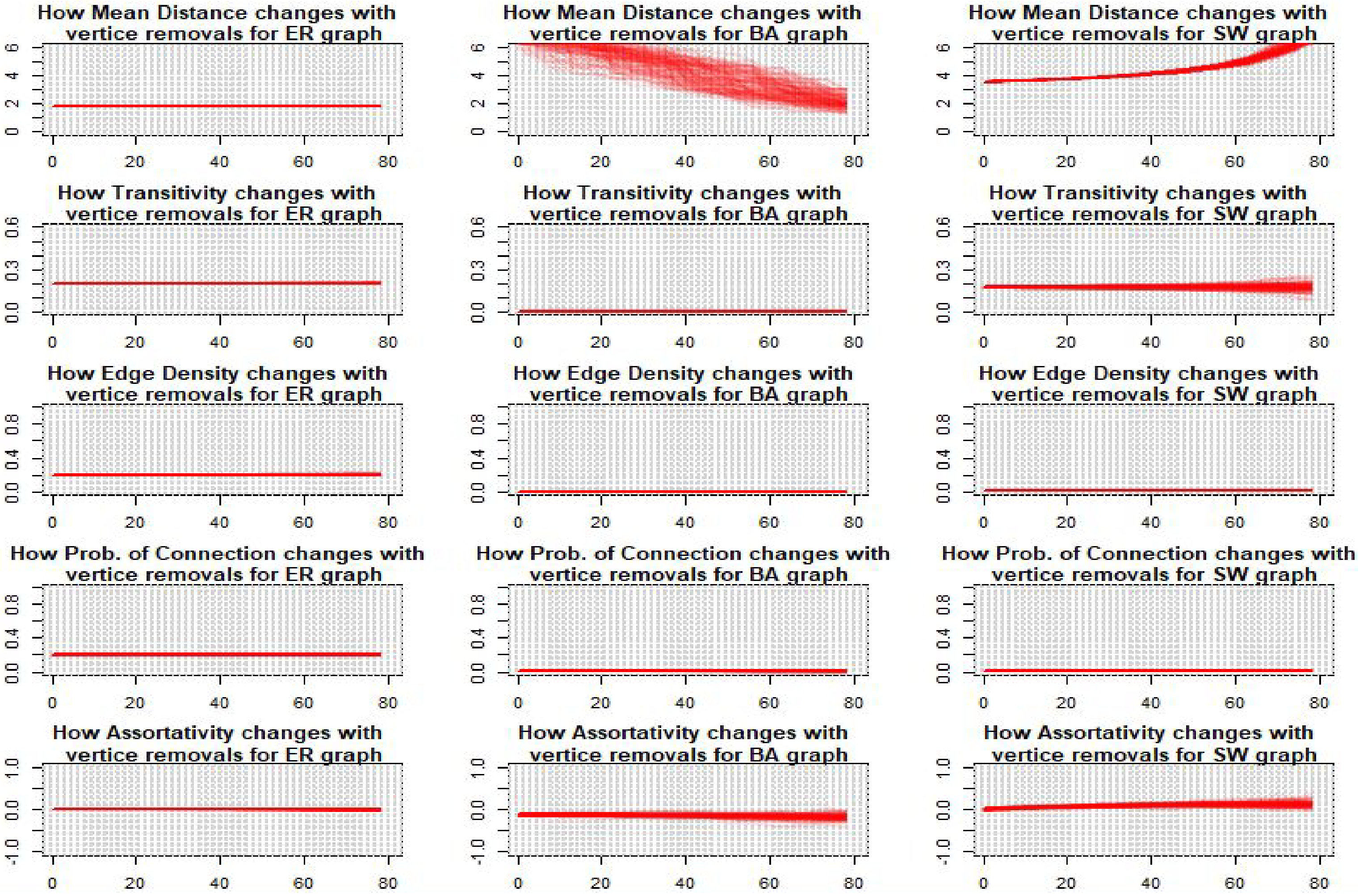


Figure 44. Vertex Removals, Starting Graph: 1000 Nodes, 10 Iterations, 10 Graphs, p = 0.3

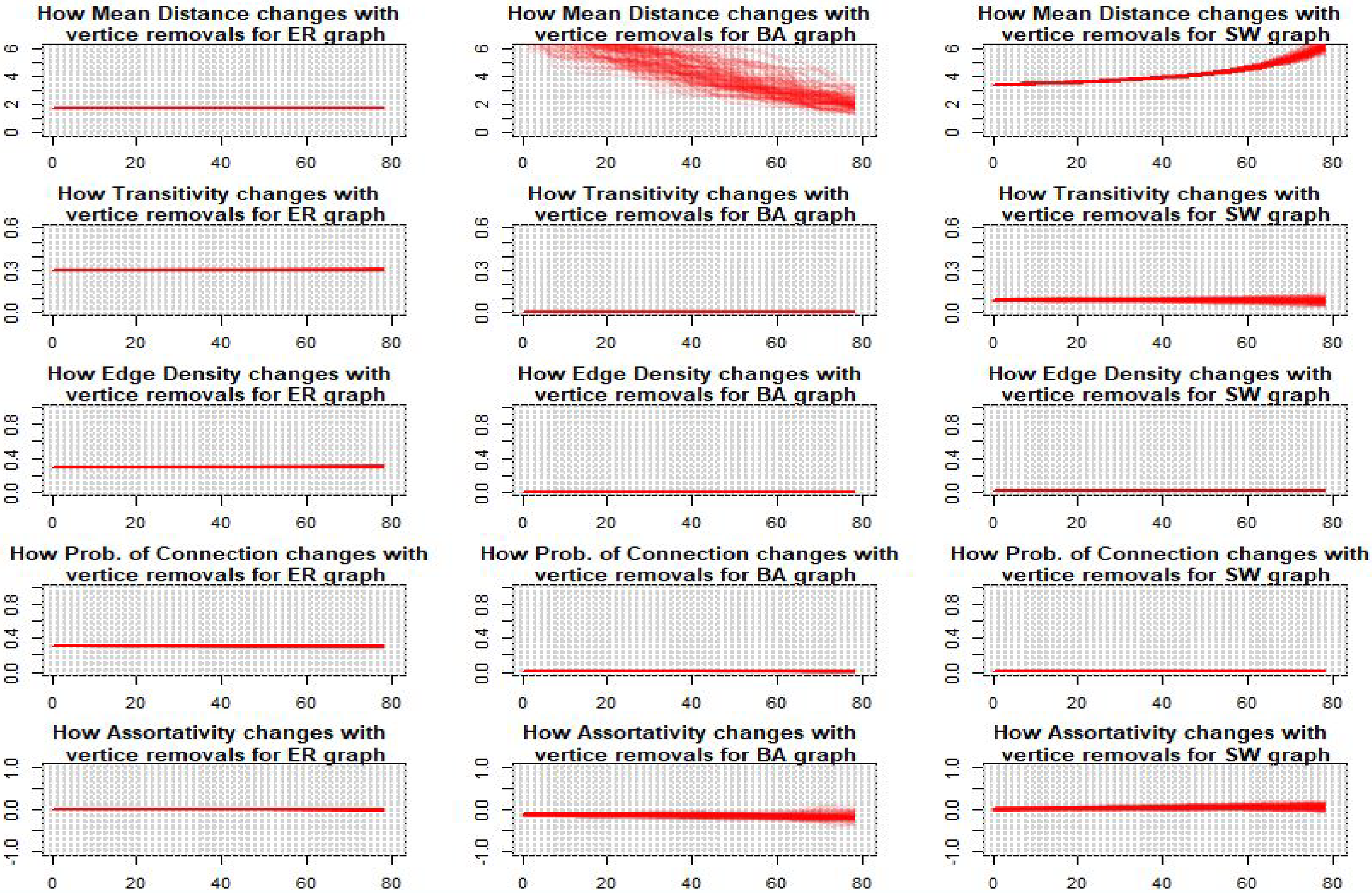
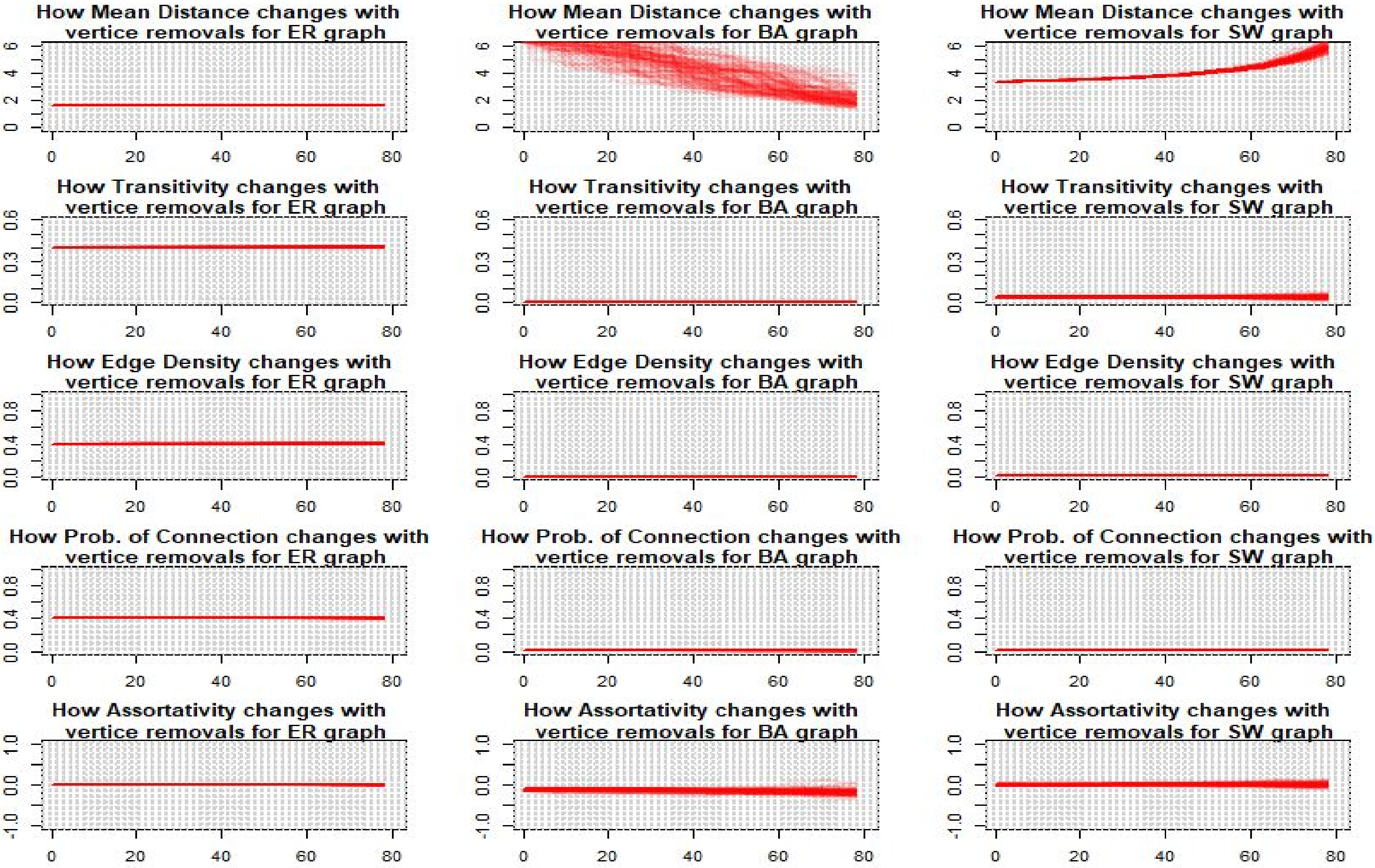


Figure 45. Vertex Removals, Starting Graph: 1000 Nodes, 10 Iterations, 10 Graphs, p = 0.4



How Mean Distance changes with vertice removals for ER graph

How Mean Distance changes with vertice removals for BA graph

How Mean Distance changes with vertice removals for SW graph

How Transitivity changes with vertice removals for ER graph

How Transitivity changes with vertice removals for BA graph

How Transitivity changes with vertice removals for SW graph

How Edge Density changes with vertice removals for ER graph

How Edge Density changes with vertice removals for BA graph

How Edge Density changes with vertice removals for SW graph

How Prob. of Connection changes with vertice removals for ER graph

How Prob. of Connection changes with vertice removals for BA graph

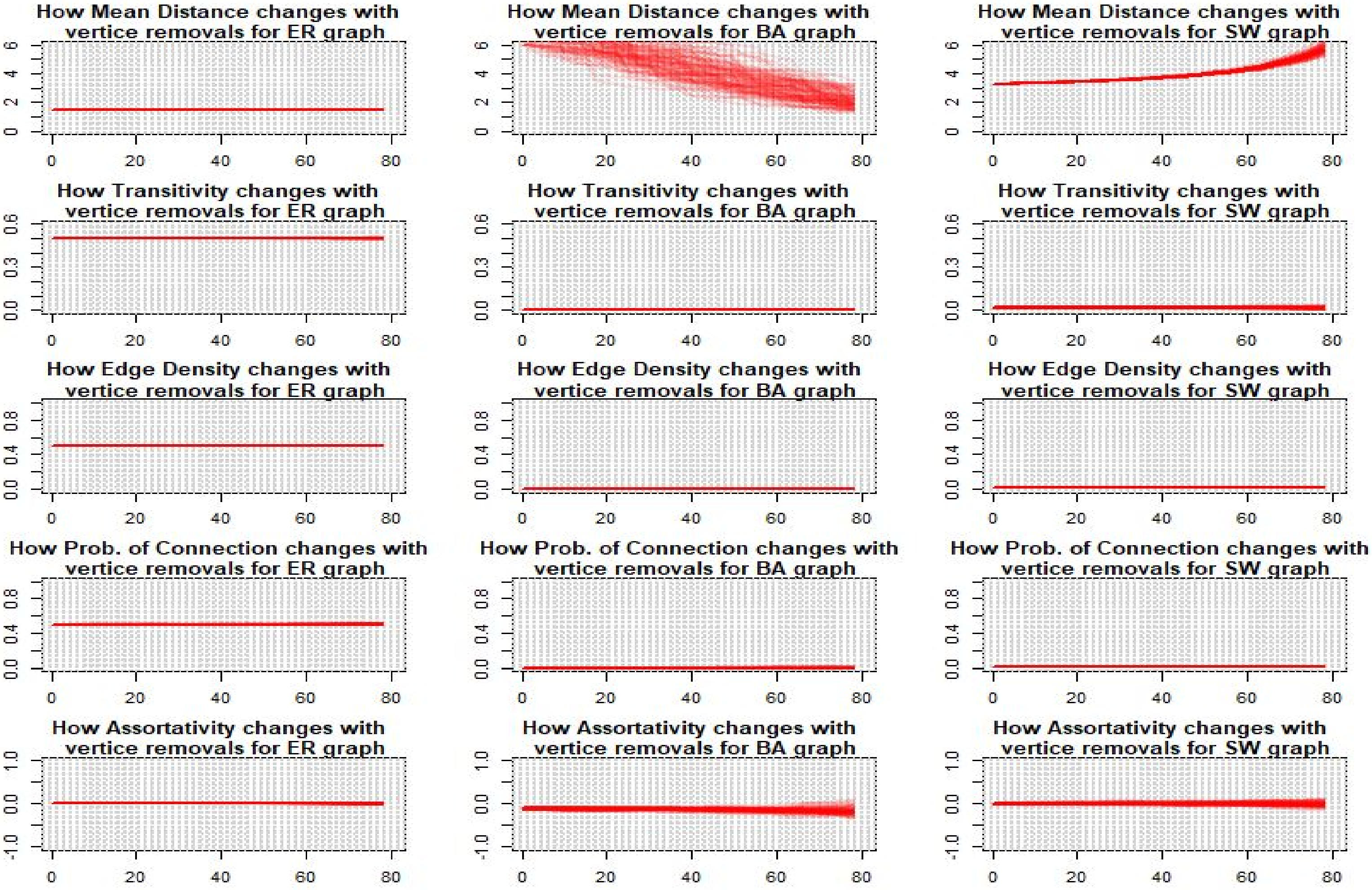
How Prob. of Connection changes with vertice removals for SW graph

How Assortativity changes with vertice removals for ER graph

How Assortativity changes with vertice removals for BA graph

How Assortativity changes with vertice removals for SW graph

Figure 46. Vertex Removals, Starting Graph: 1000 Nodes, 10 Iterations, 10 Graphs, p = 0.5



How Mean Distance changes with vertice removals for ER graph

How Mean Distance changes with vertice removals for BA graph

How Mean Distance changes with vertice removals for SW graph

How Transitivity changes with vertice removals for ER graph

How Transitivity changes with vertice removals for BA graph

How Transitivity changes with vertice removals for SW graph

How Edge Density changes with vertice removals for ER graph

How Edge Density changes with vertice removals for BA graph

How Edge Density changes with vertice removals for SW graph

How Prob. of Connection changes with vertice removals for ER graph

How Prob. of Connection changes with vertice removals for BA graph

How Prob. of Connection changes with vertice removals for SW graph

How Assortativity changes with vertice removals for ER graph

How Assortativity changes with vertice removals for BA graph

How Assortativity changes with vertice removals for SW graph

Figure 47. KL and H Statistic: Edge Removals, Starting Graph: 100 Nodes, 10 Iterations, 10 Graphs, *p* = 0.1

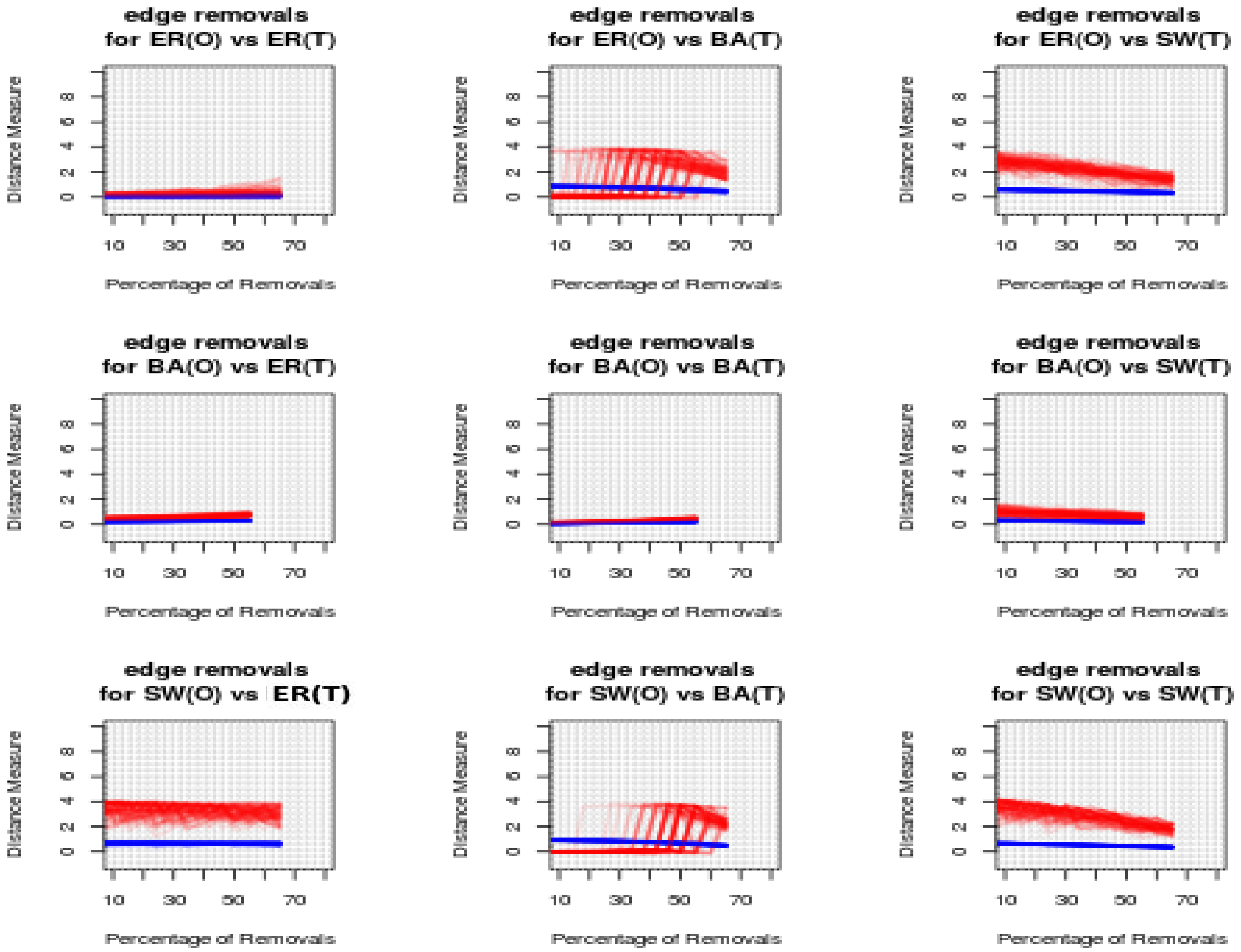


Figure 48. KL and H Statistic: Edge removals, Starting Graph: 100 Nodes, 10 Iterations, 10 Graphs, *p* = 0.2

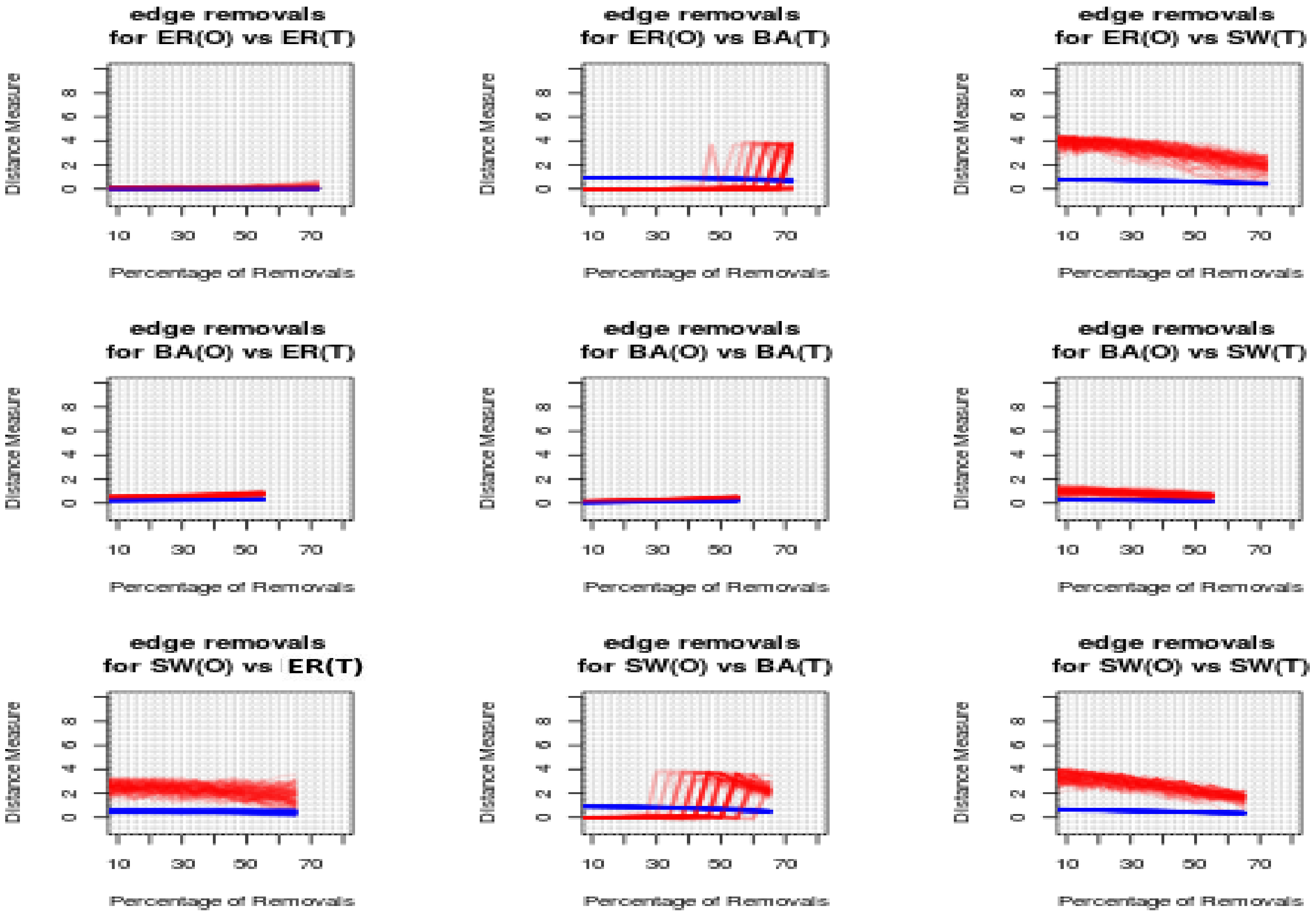


Figure 49. KL and H Statistic: Edge Removals, Starting Graph: 100 Nodes, 10 Iterations, 10 Graphs, p = 0.3

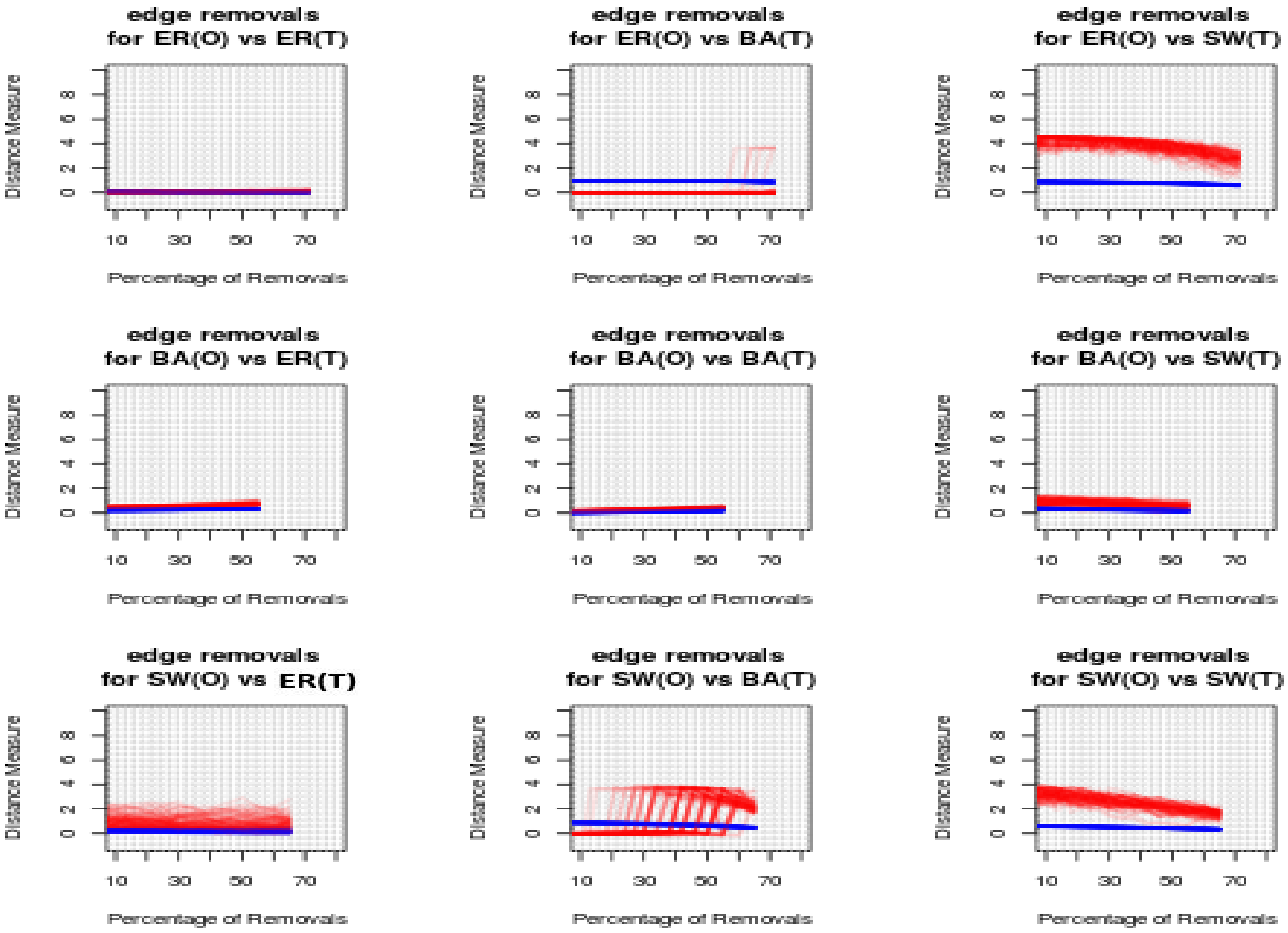


Figure 50. KL and H Statistic: Edge Removals, Starting Graph: 100 Nodes, 10 Iterations, 10 Graphs, p = 0.4

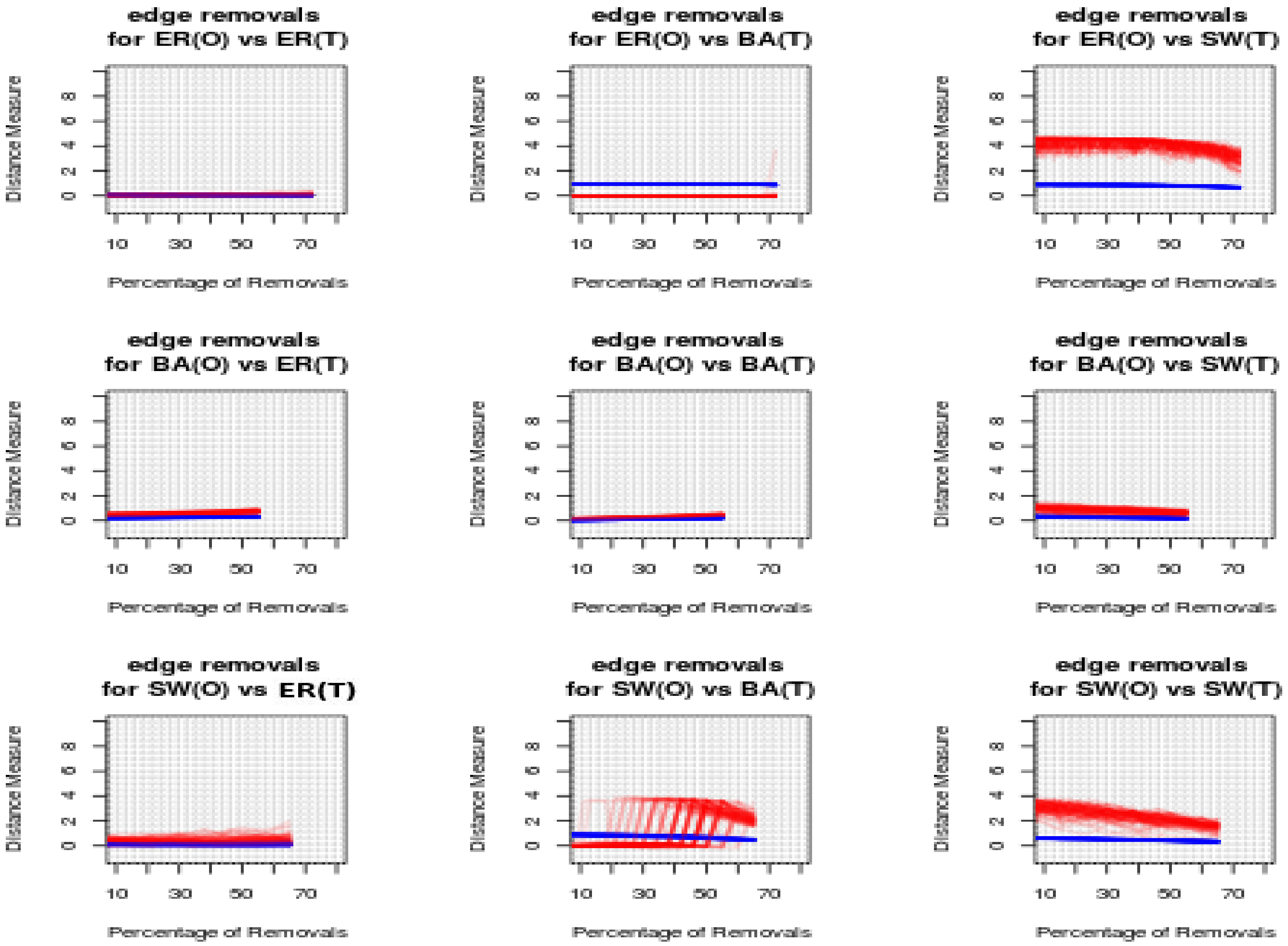


Figure 51. KL and H Statistic: Edge Removals, Starting Graph: 100 Nodes, 10 Iterations, 10 Graphs, *p* = 0.5

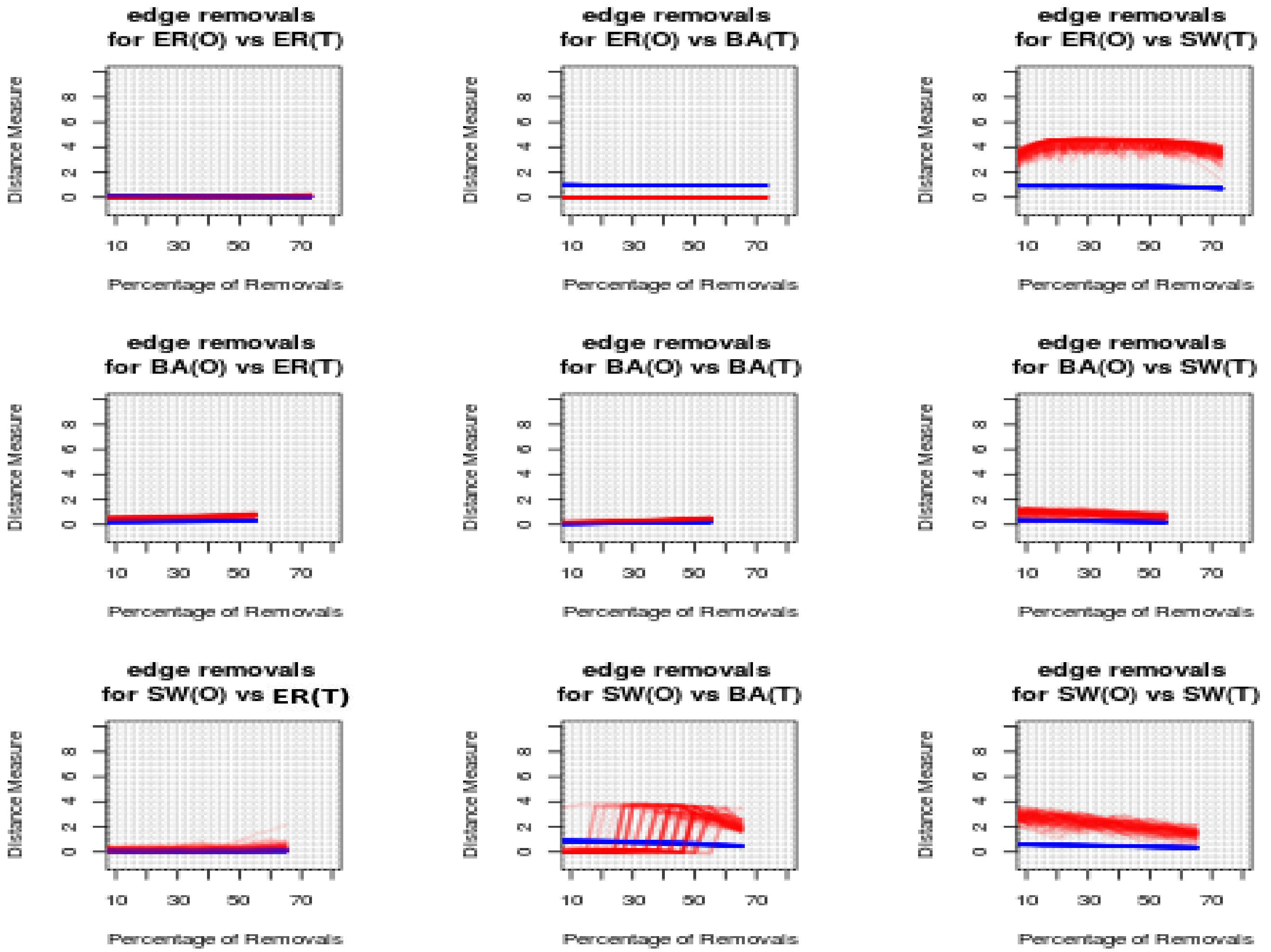


Figure 52. KL and H Statistic: Vertex Removals, Starting Graph: 100 Nodes, 10 Iterations, 10 Graphs, *p* = 0.1

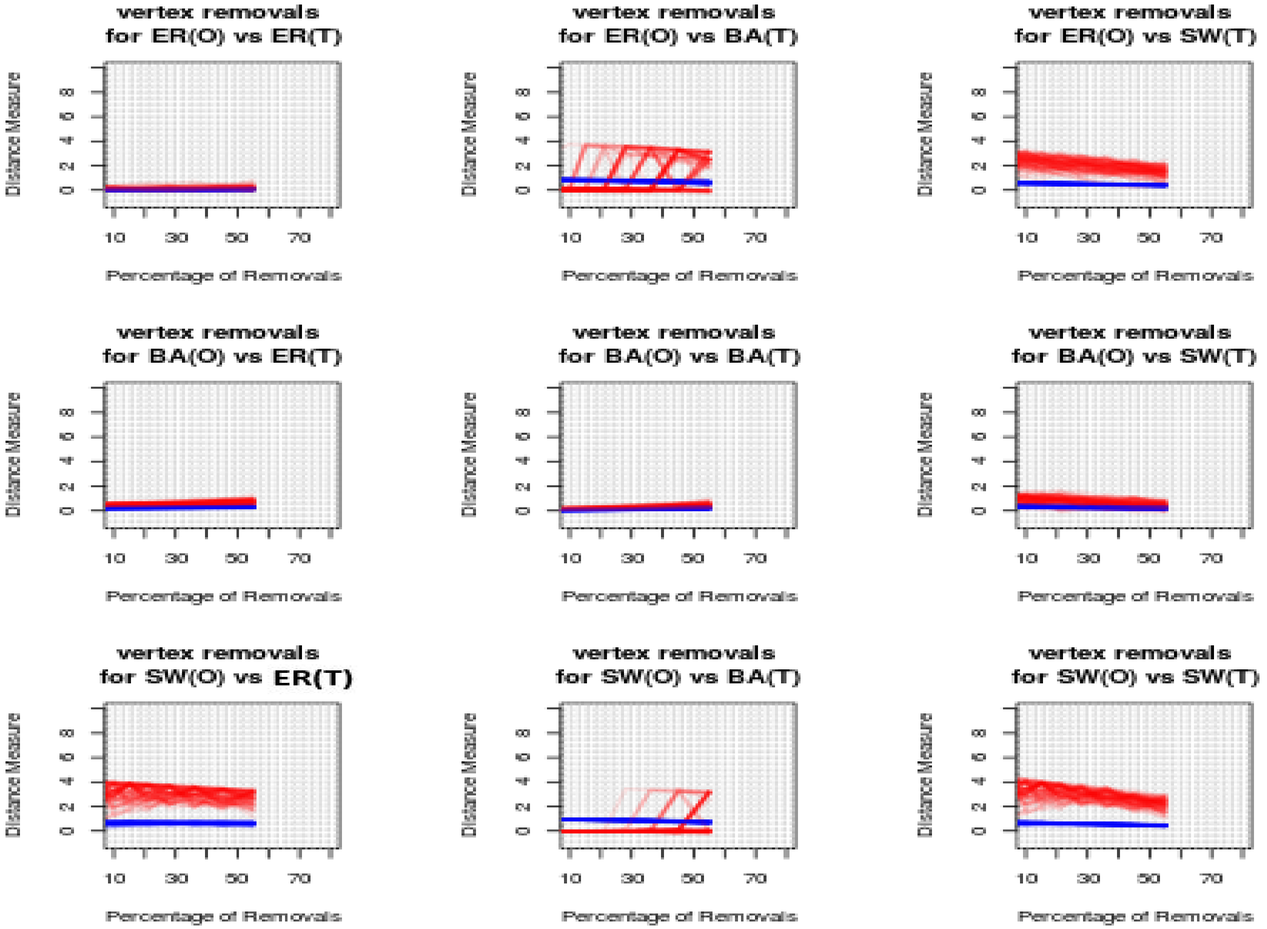


Figure 53. KL and H Statistic: Vertex Removals, Starting Graph: 100 Nodes, 10 Iterations, 10 Graphs, p = 0.2

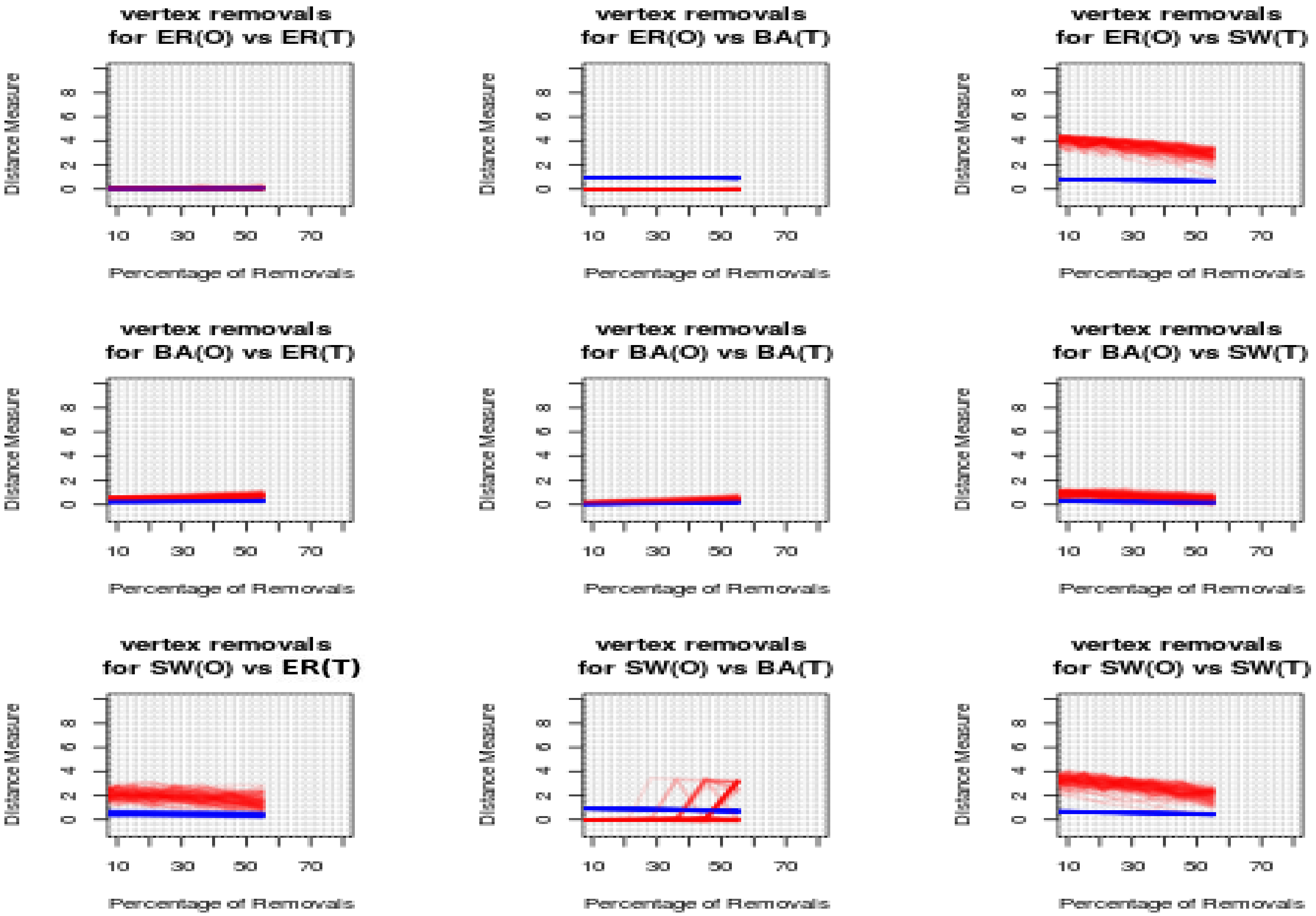


Figure 54. KL and H Statistic: Vertex Removals, Starting Graph: 100 Nodes, 10 Iterations, 10 Graphs, p = 0.3

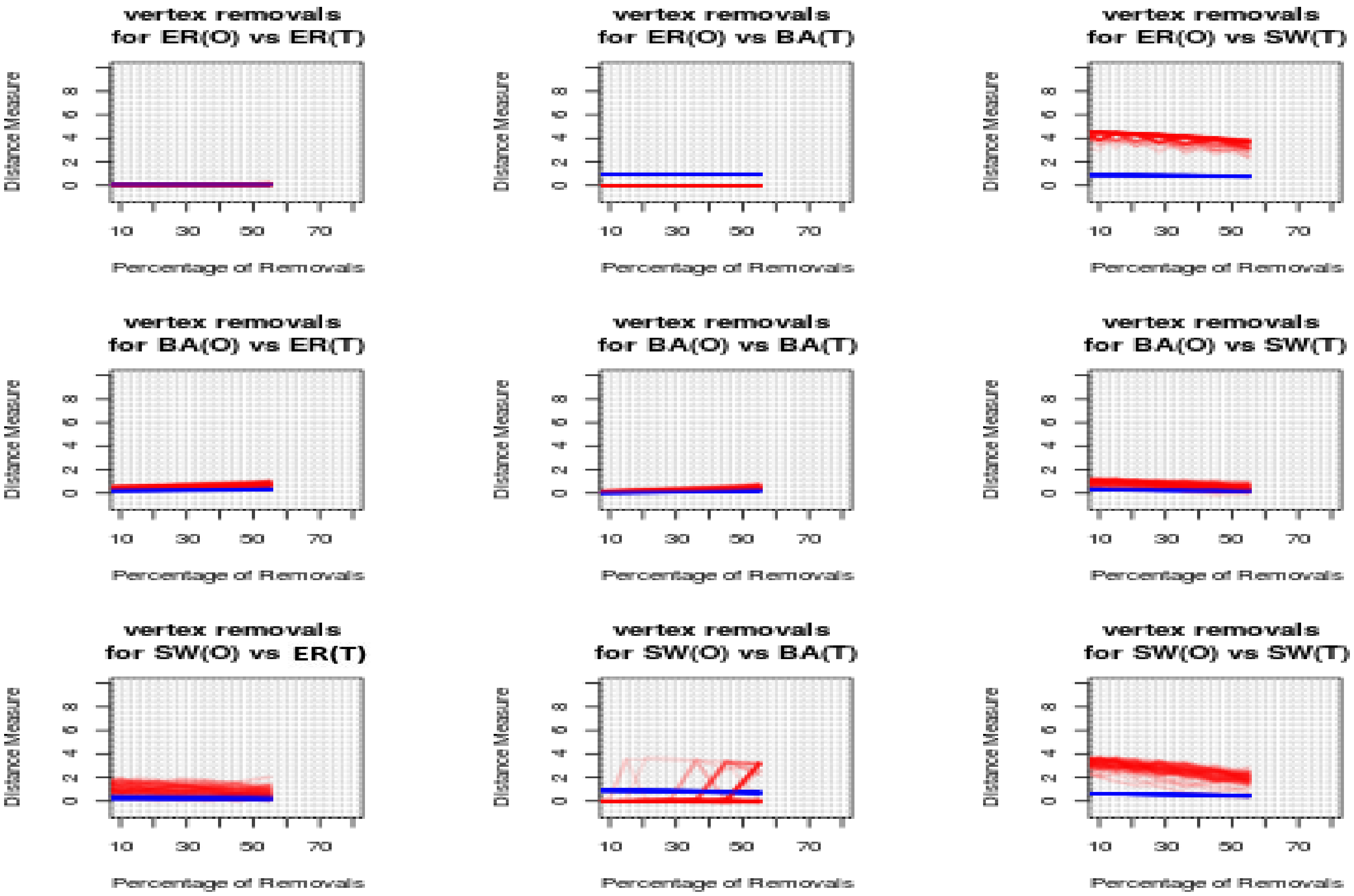


Figure 55. KL and H Statistic: Vertex Removals, Starting Graph: 100 Nodes, 10 Iterations, 10 Graphs, p = 0.4

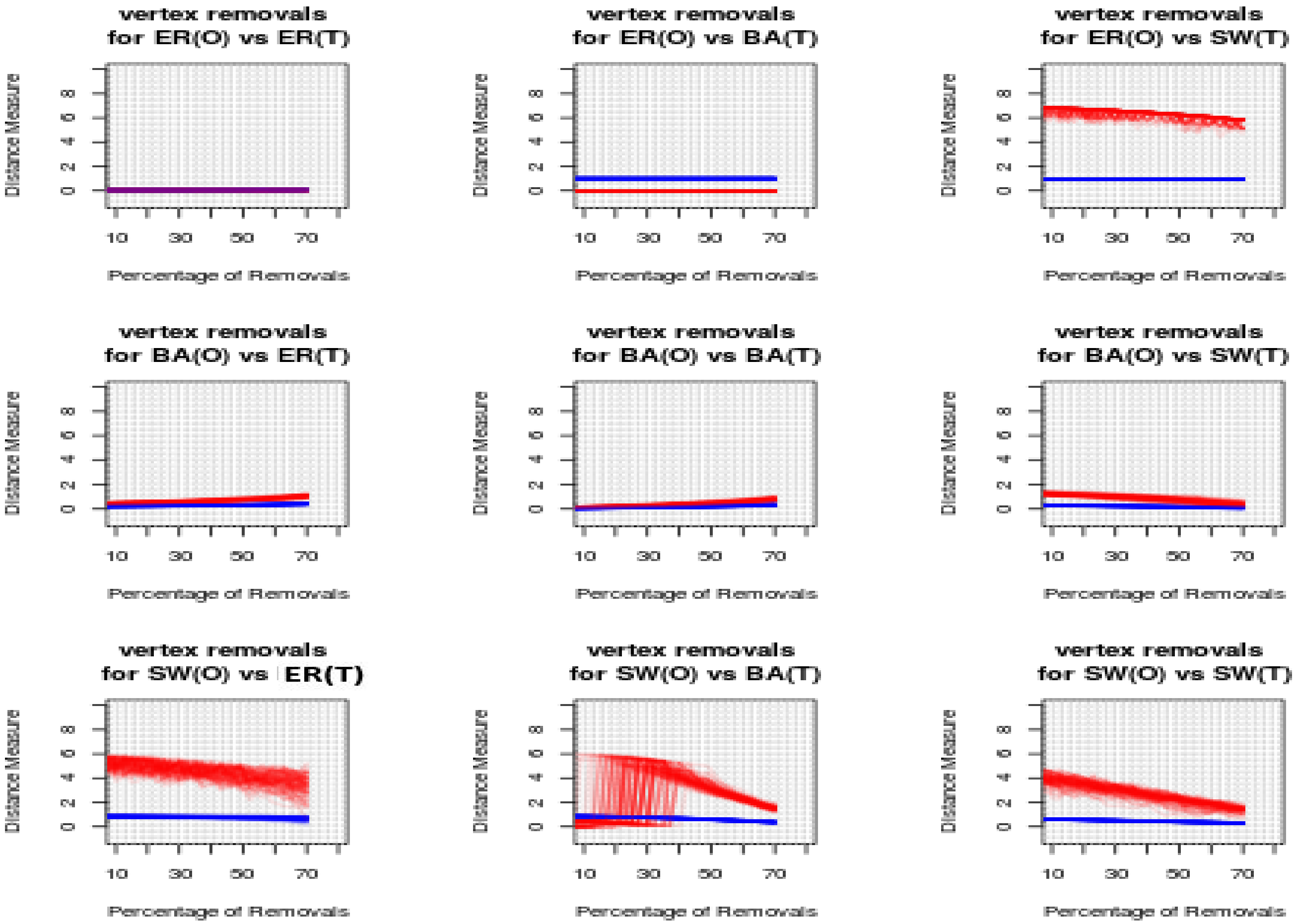


Figure 56. KL and H Statistic: Vertex Removals, Starting Graph: 100 Nodes, 10 Iterations, 10 Graphs, *p* = 0.5

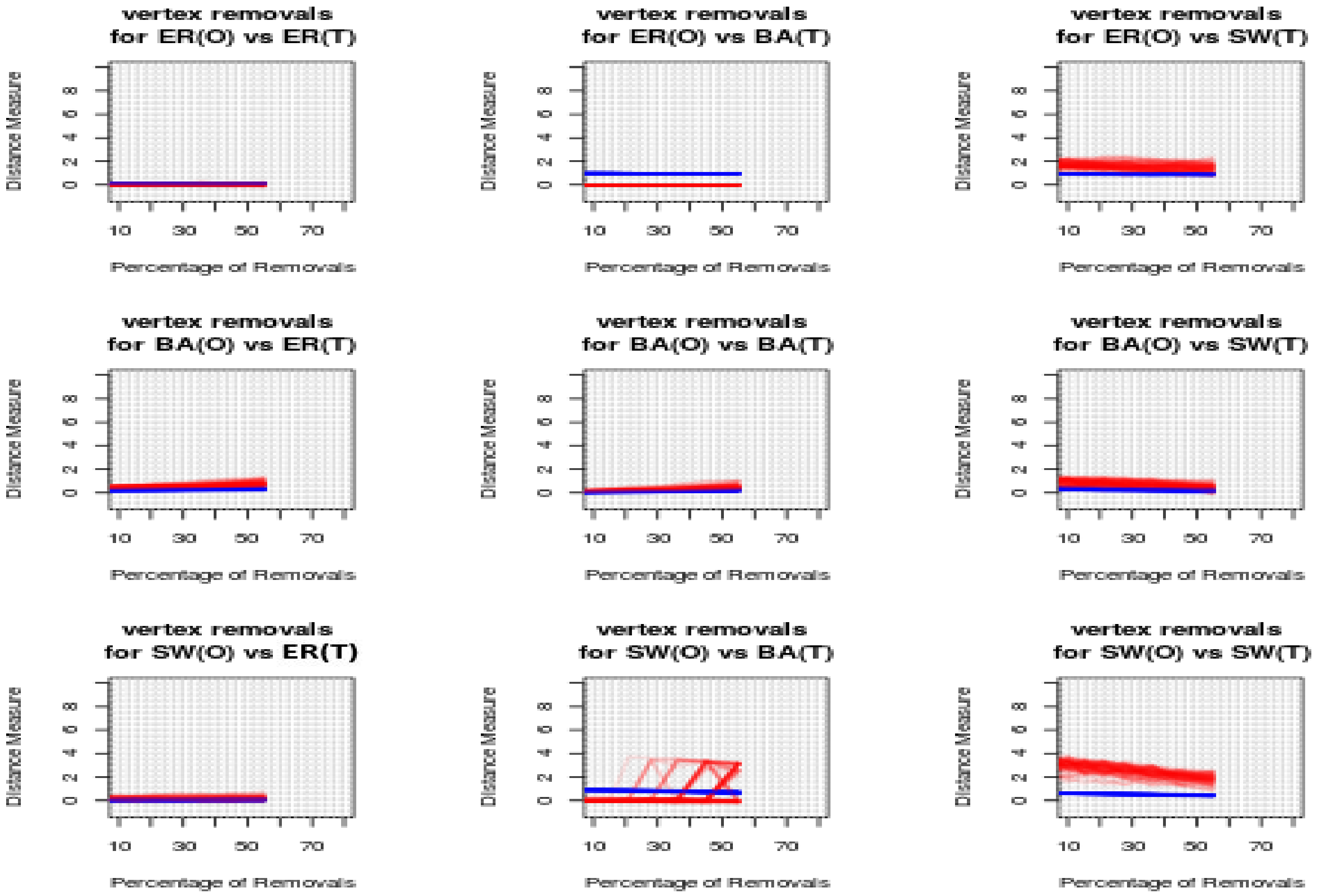


Figure 57. KL and H Statistic: Edge Removals, Starting Graph: 500 Nodes, 10 Iterations, 10 Graphs, *p* = 0.1

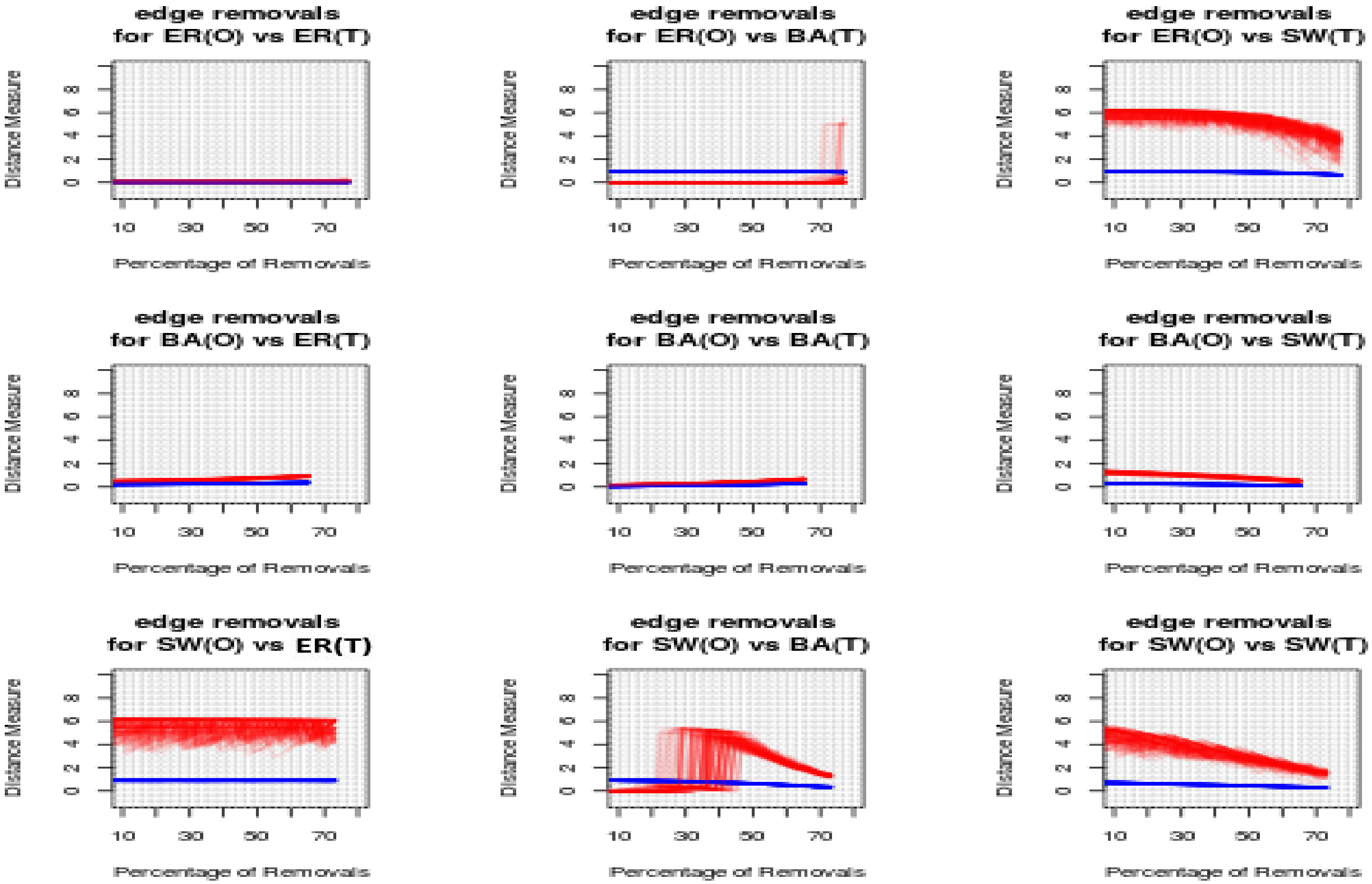


Figure 58. KL and H Statistic: Edge Removals, Starting Graph: 500 Nodes, 10 Iterations, 10 Graphs, $p = 0.2$

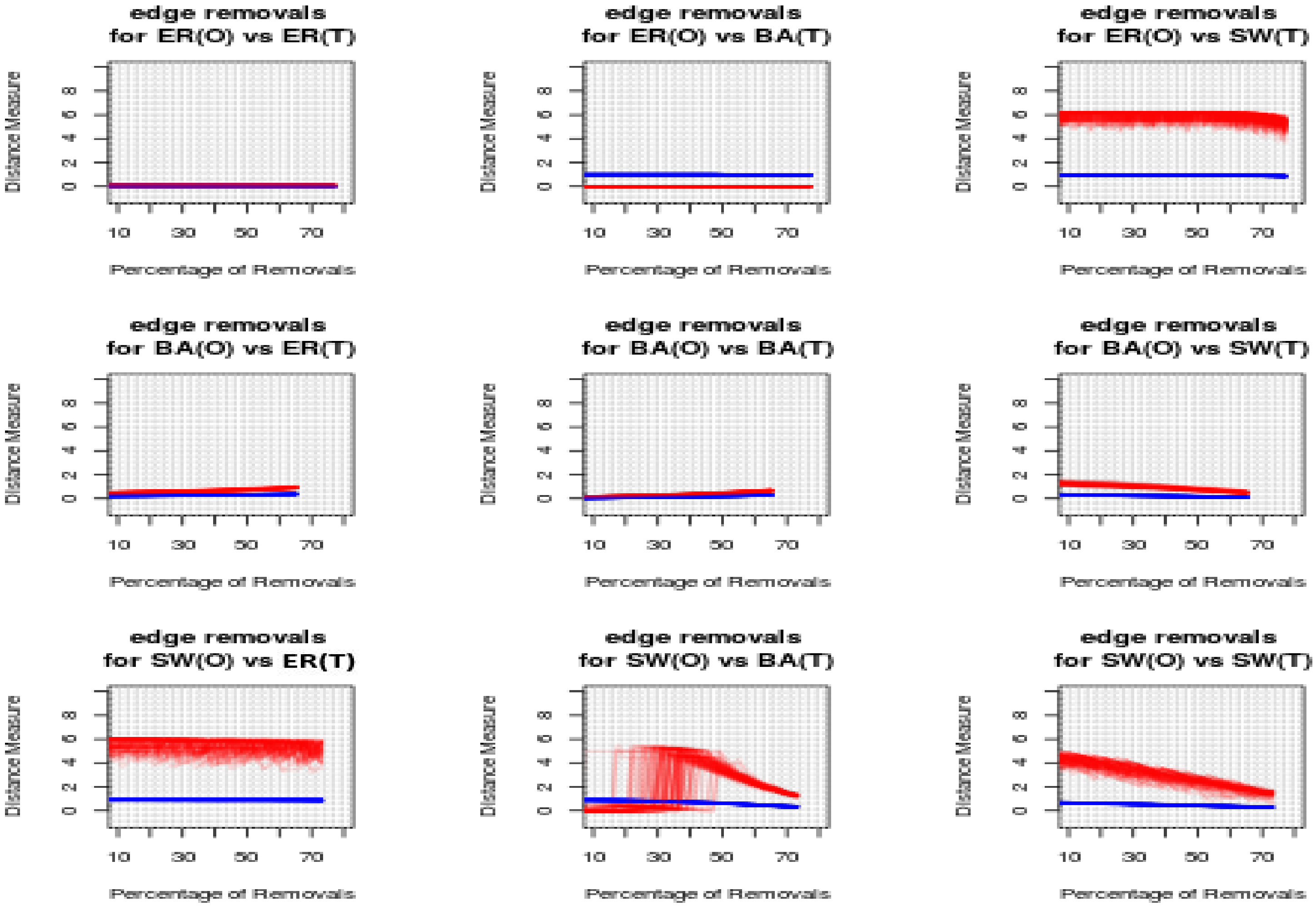
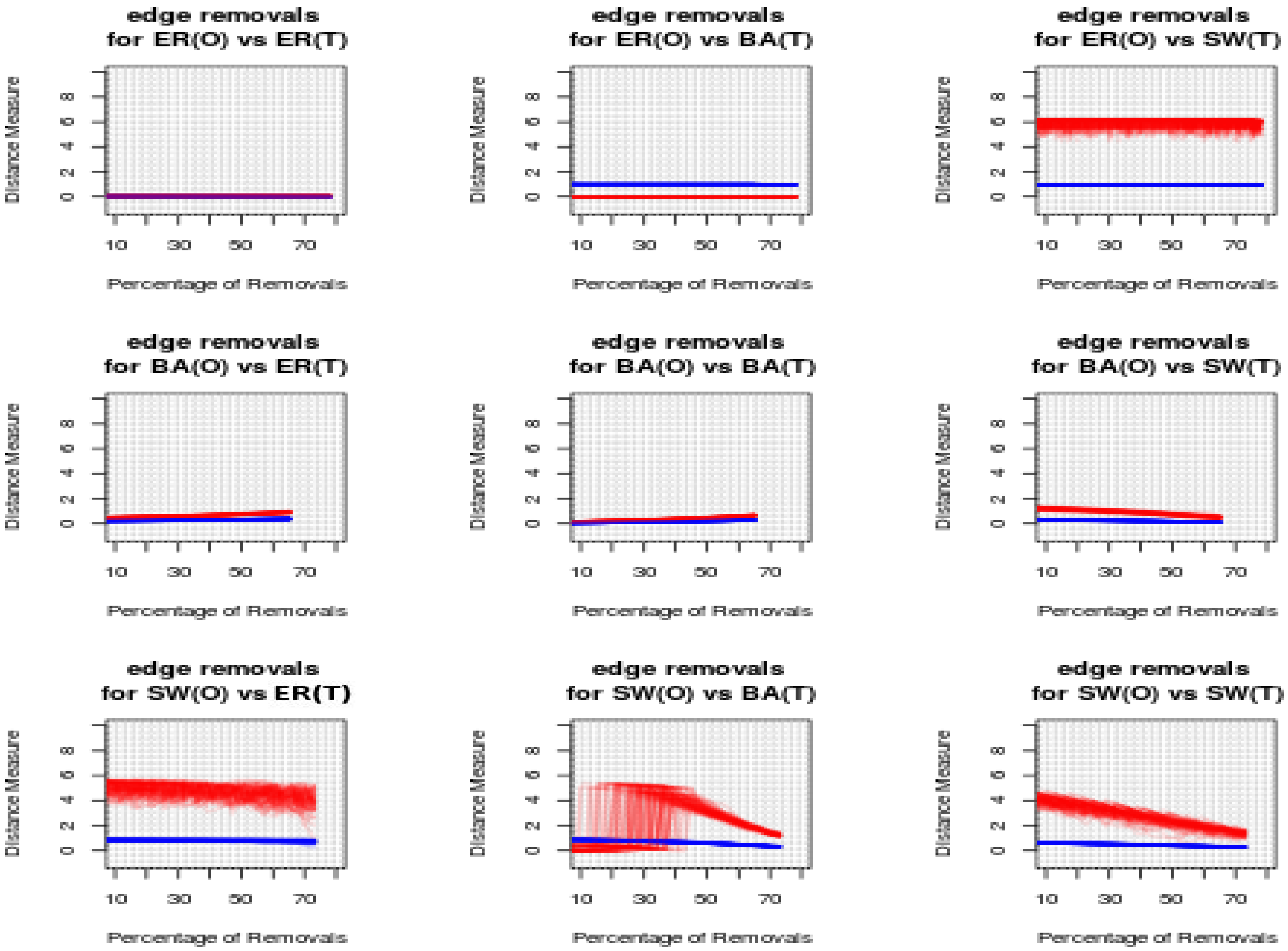


Figure 59. KL and H Statistic: Edge Removals, Starting Graph: 500 Nodes, 10 Iterations, 10 Graphs, p = 0.3



KL and H Statistic: Edge Removals, Starting Graph: 500 Nodes, 10 Iterations, 10 Graphs, p = 0.4

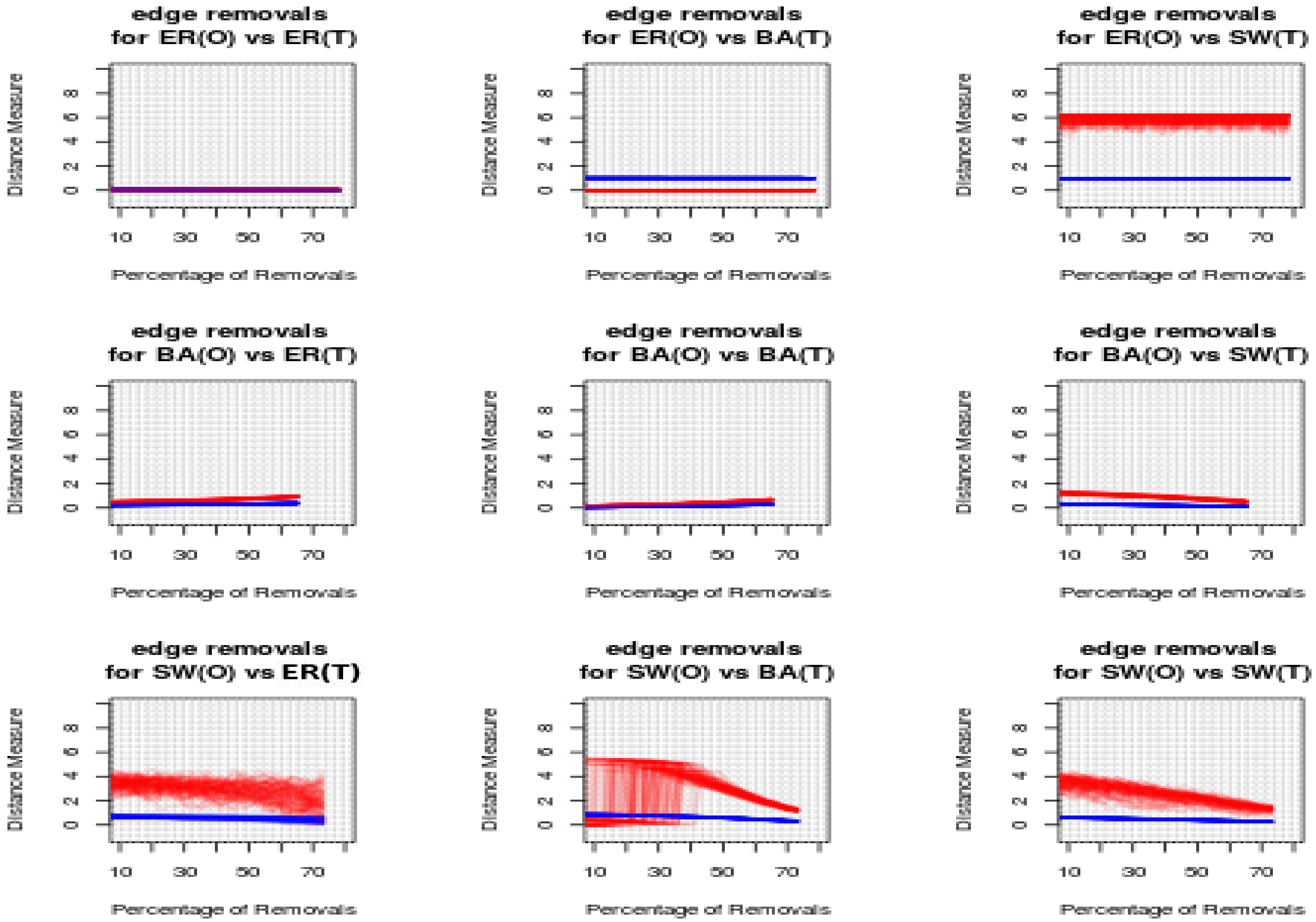
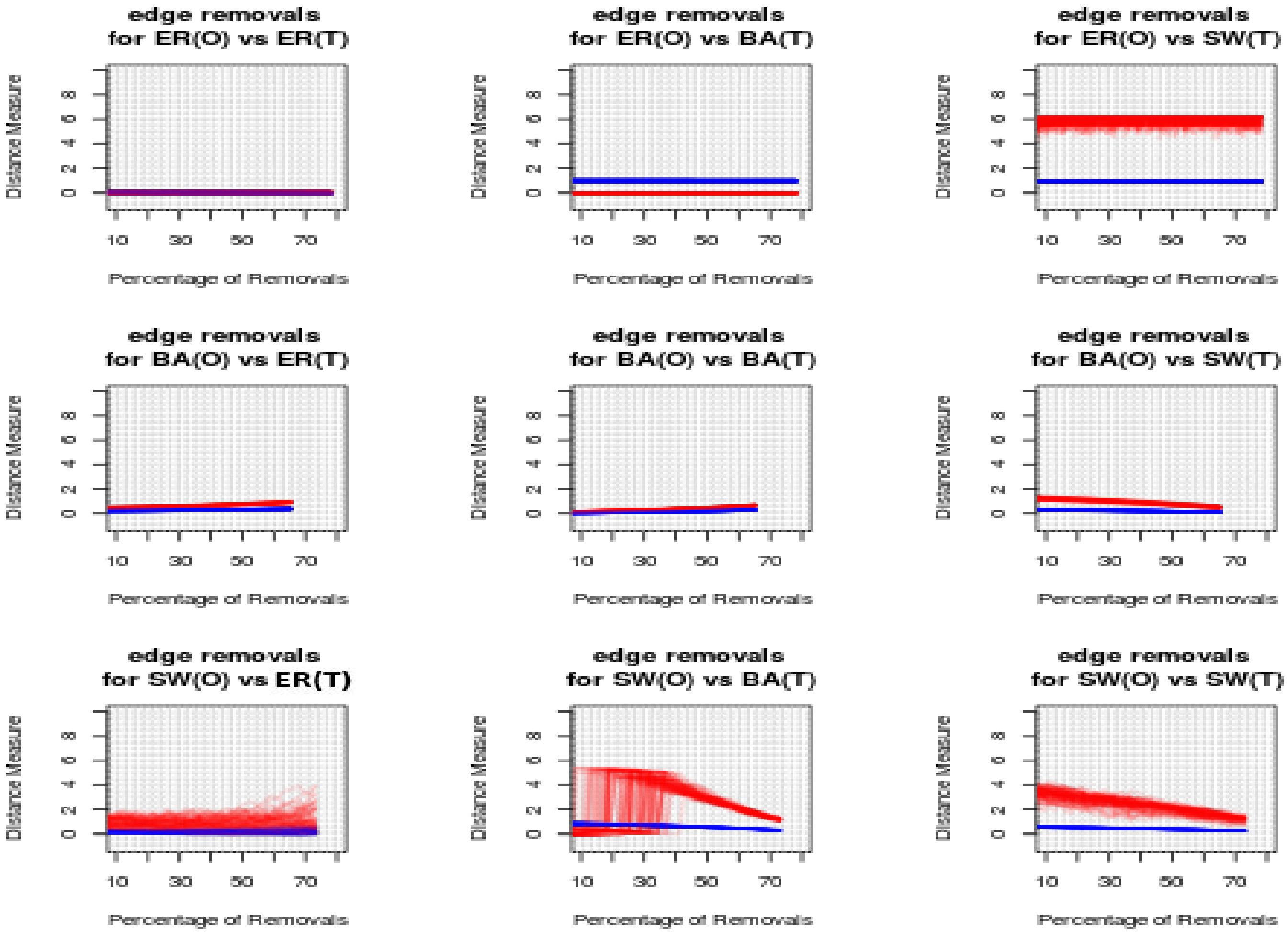


Figure 60. KL and H Statistic: Edge Removals, Starting Graph: 500 Nodes, 10 Iterations, 10 Graphs, p = 0.5



edge removals for ER(O) vs ER(T)

edge removals for ER(O) vs BA(T)

edge removals for ER(O) vs SW(T)

edge removals for BA(O) vs ER(T)

edge removals for BA(O) vs BA(T)

edge removals for BA(O) vs SW(T)

edge removals for SW(O) vs ER(T)

edge removals for SW(O) vs BA(T)

edge removals for SW(O) vs SW(T)

Figure 61. KL and H Statistic: Vertex Removals, Starting Graph: 500 Nodes, 10 Iterations, 10 Graphs, *p* = 0.1

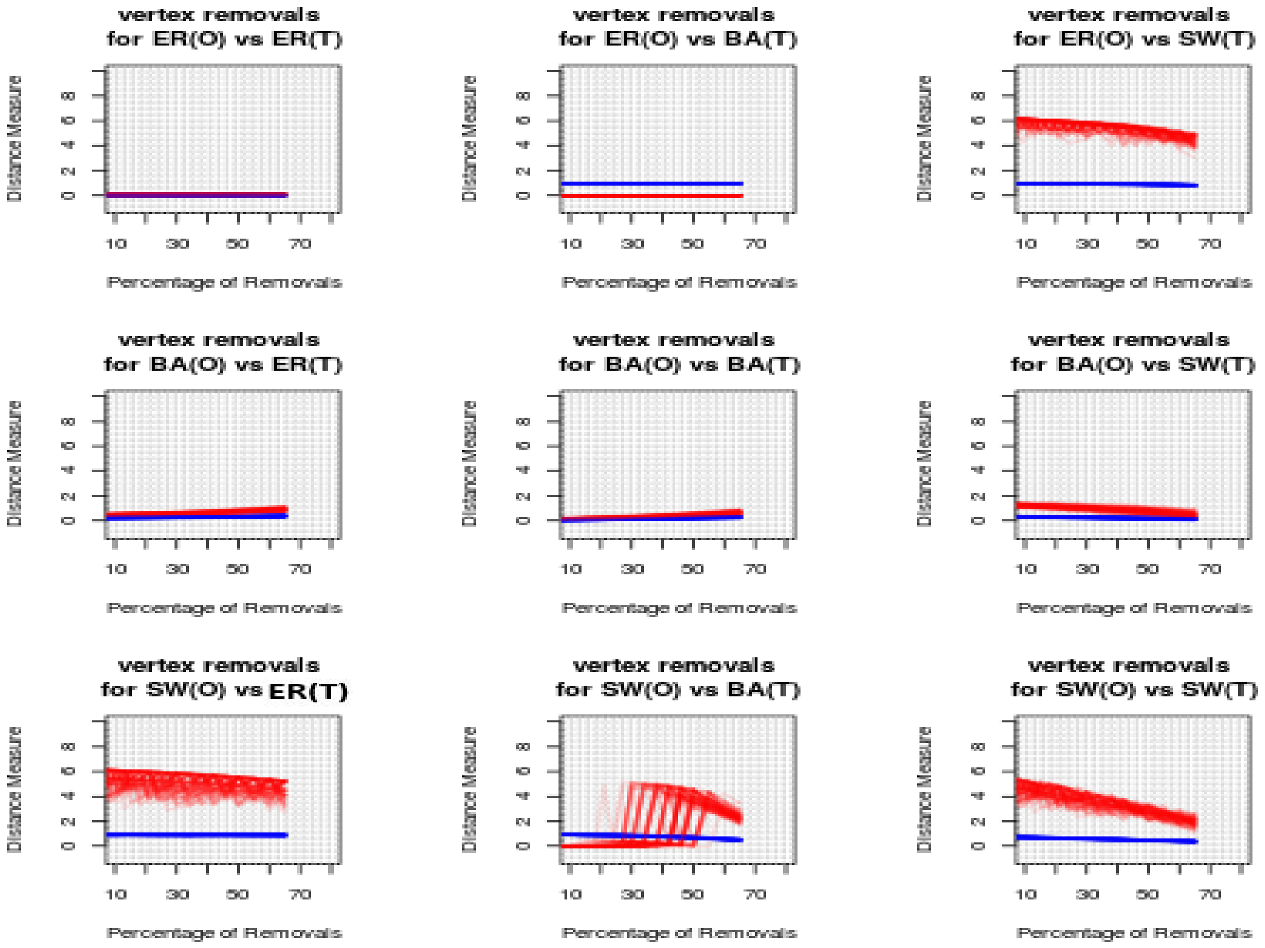


Figure 62. KL and H Statistic: Vertex Removals, Starting Graph: 500 Nodes, 10 Iterations, 10 Graphs, p = 0.2

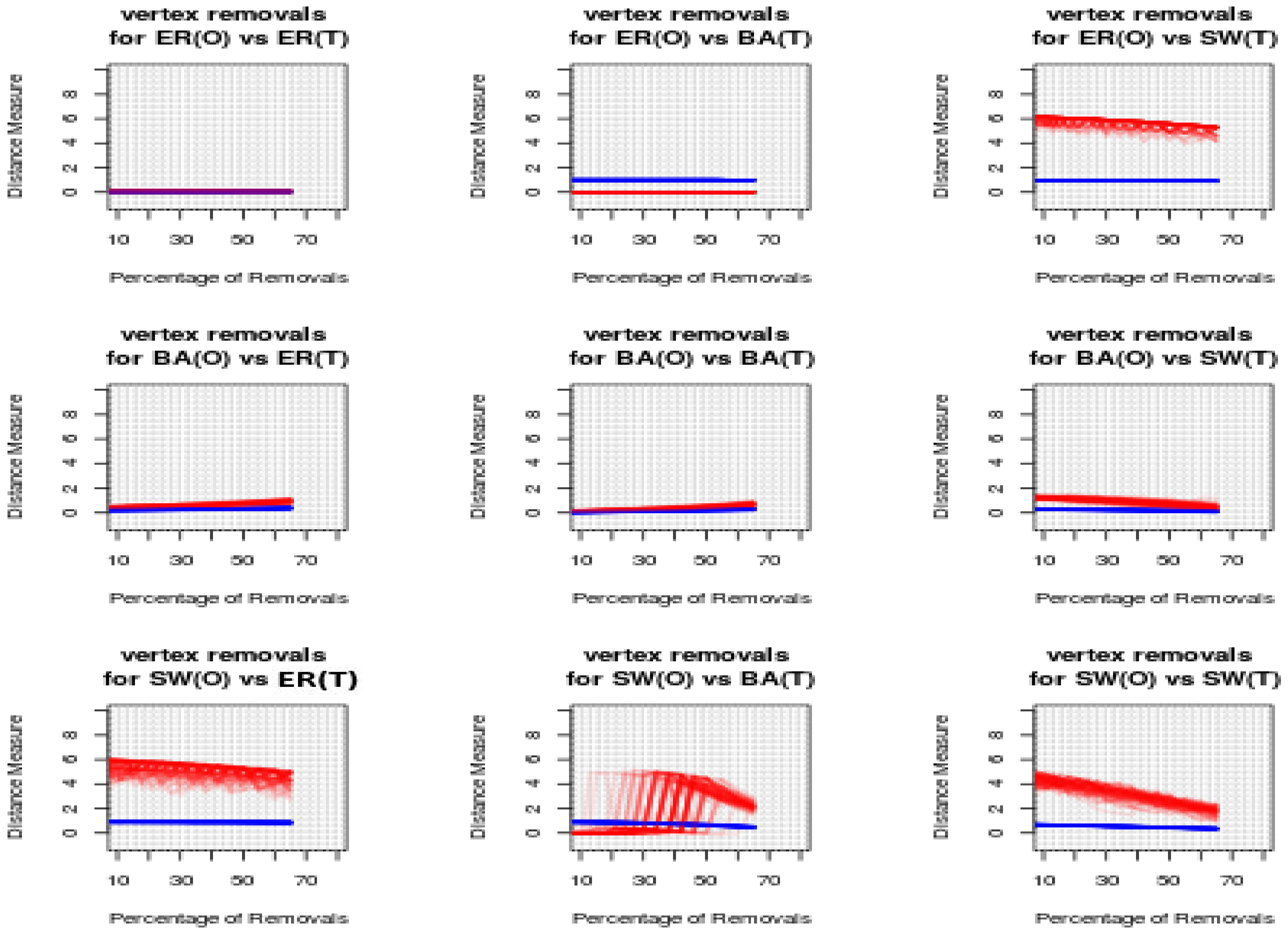


Figure 63. KL and H Statistic: Vertex Removals, Starting Graph: 500 Nodes, 10 Iterations, 10 Graphs, p = 0.3

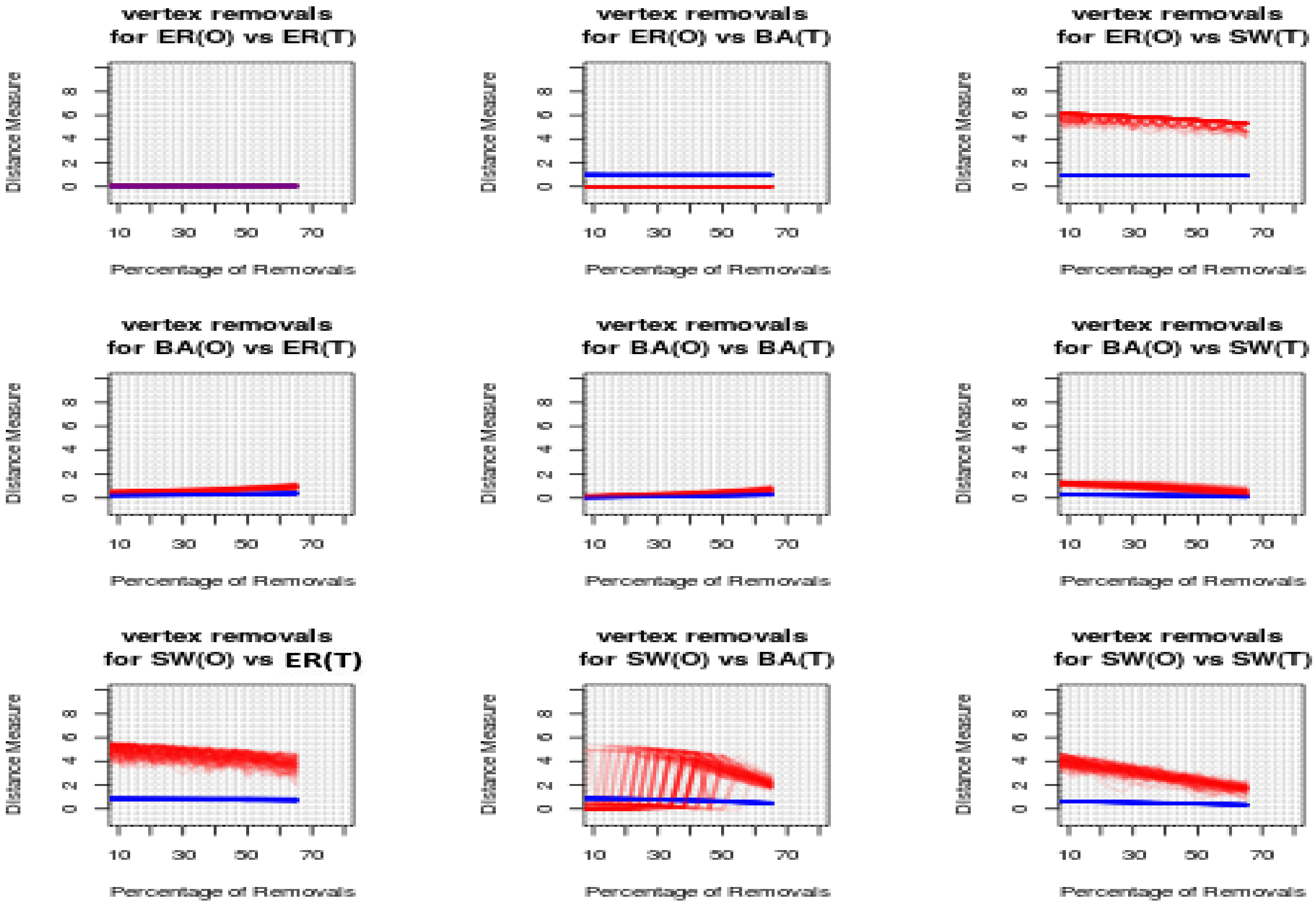
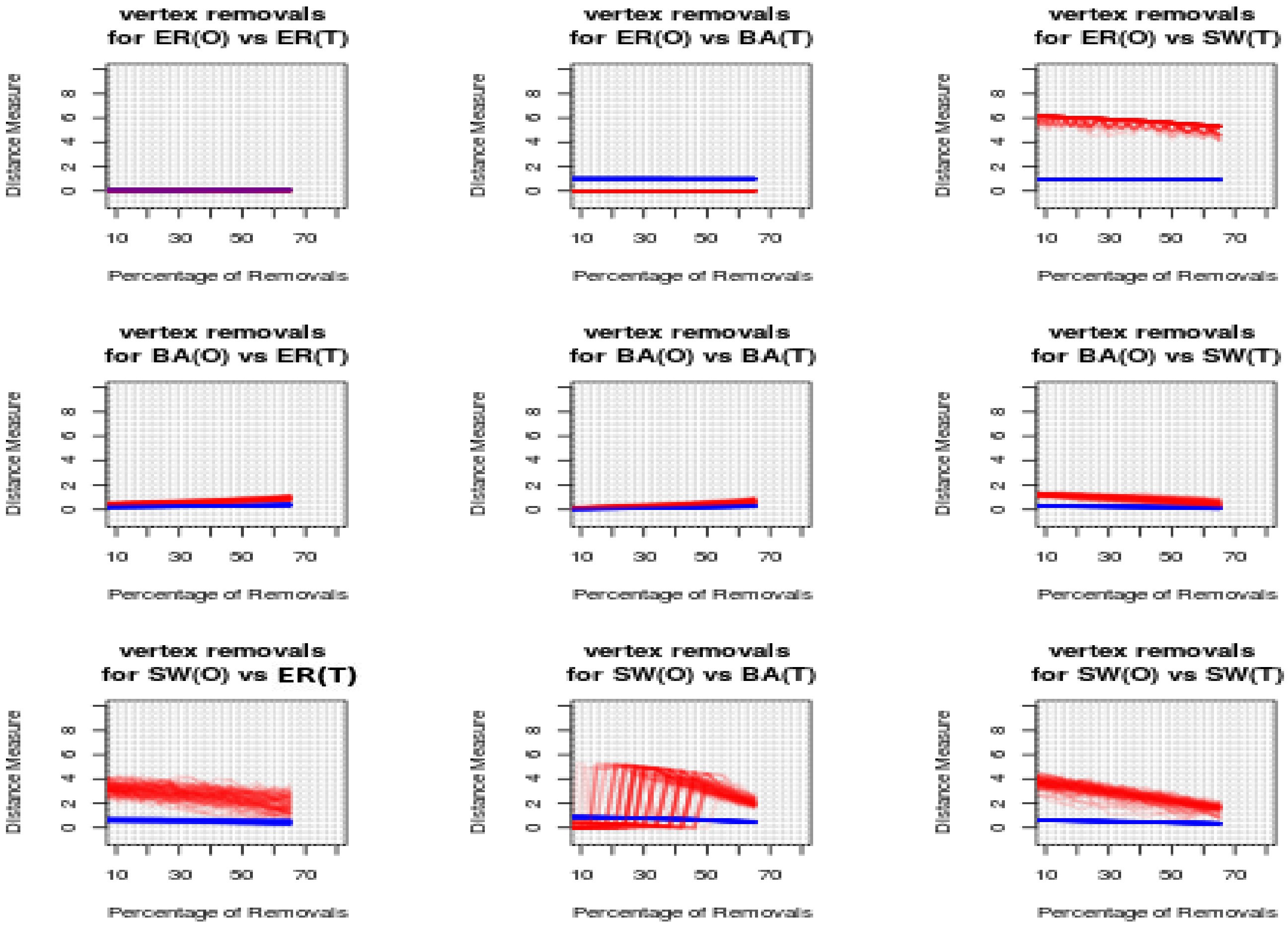


Figure 64. KL and H Statistic: Vertex Removals, Starting Graph: 500 Nodes, 10 Iterations, 10 Graphs, p = 0.4



vertex removals for ER(O) vs ER(T)

vertex removals for ER(O) vs BA(T)

vertex removals for ER(O) vs SW(T)

vertex removals for BA(O) vs ER(T)

vertex removals for BA(O) vs BA(T)

vertex removals for BA(O) vs SW(T)

vertex removals for SW(O) vs ER(T)

vertex removals for SW(O) vs BA(T)

vertex removals for SW(O) vs SW(T)

Figure 65. KL and H Statistic: Vertex Removals, Starting Graph: 500 Nodes, 10 Iterations, 10 Graphs, p = 0.5

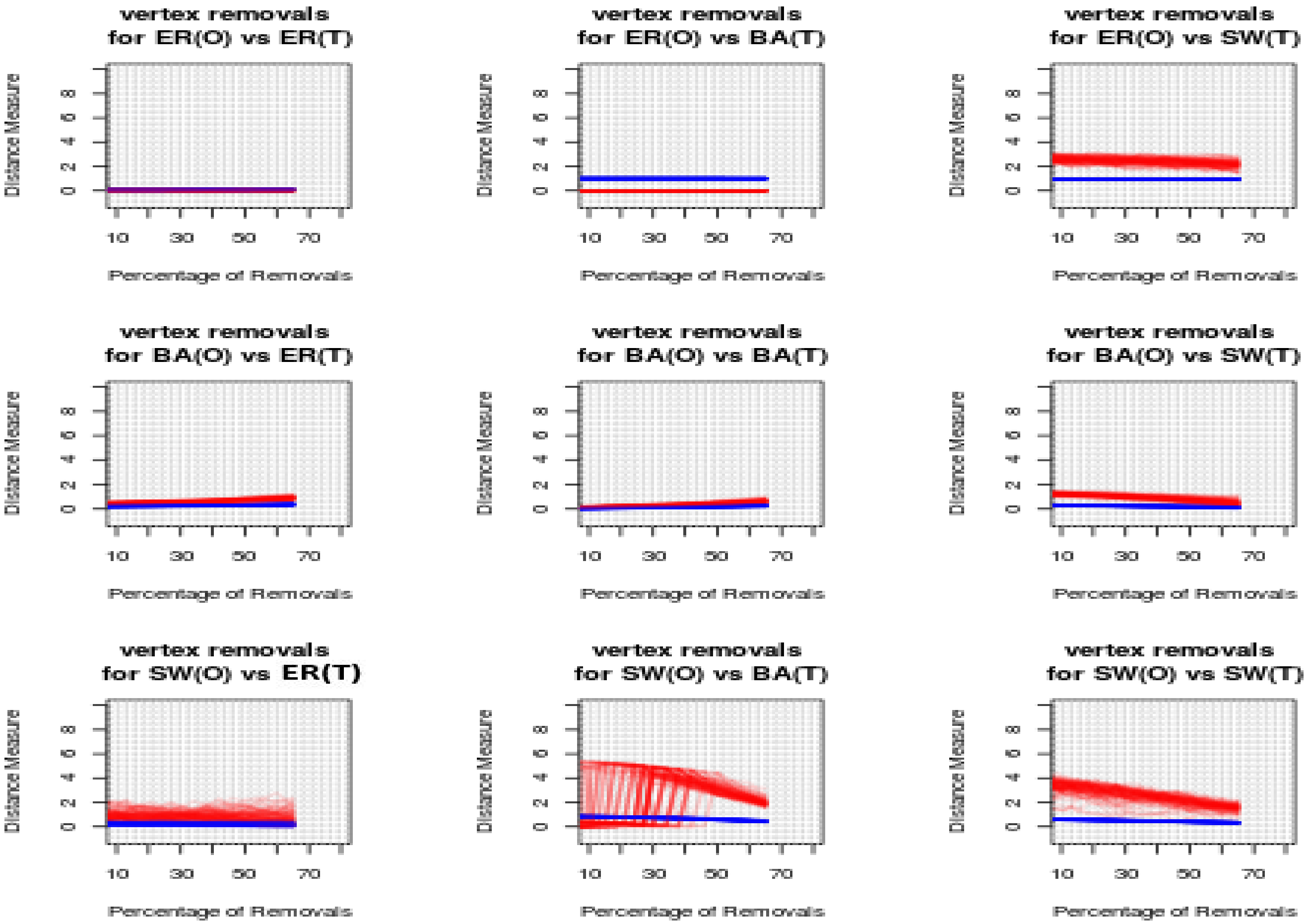


Figure 66. KL and H Statistic: Edge Removals, Starting Graph: 1000 Nodes, 10 Iterations, 10 Graphs, p = 0.1

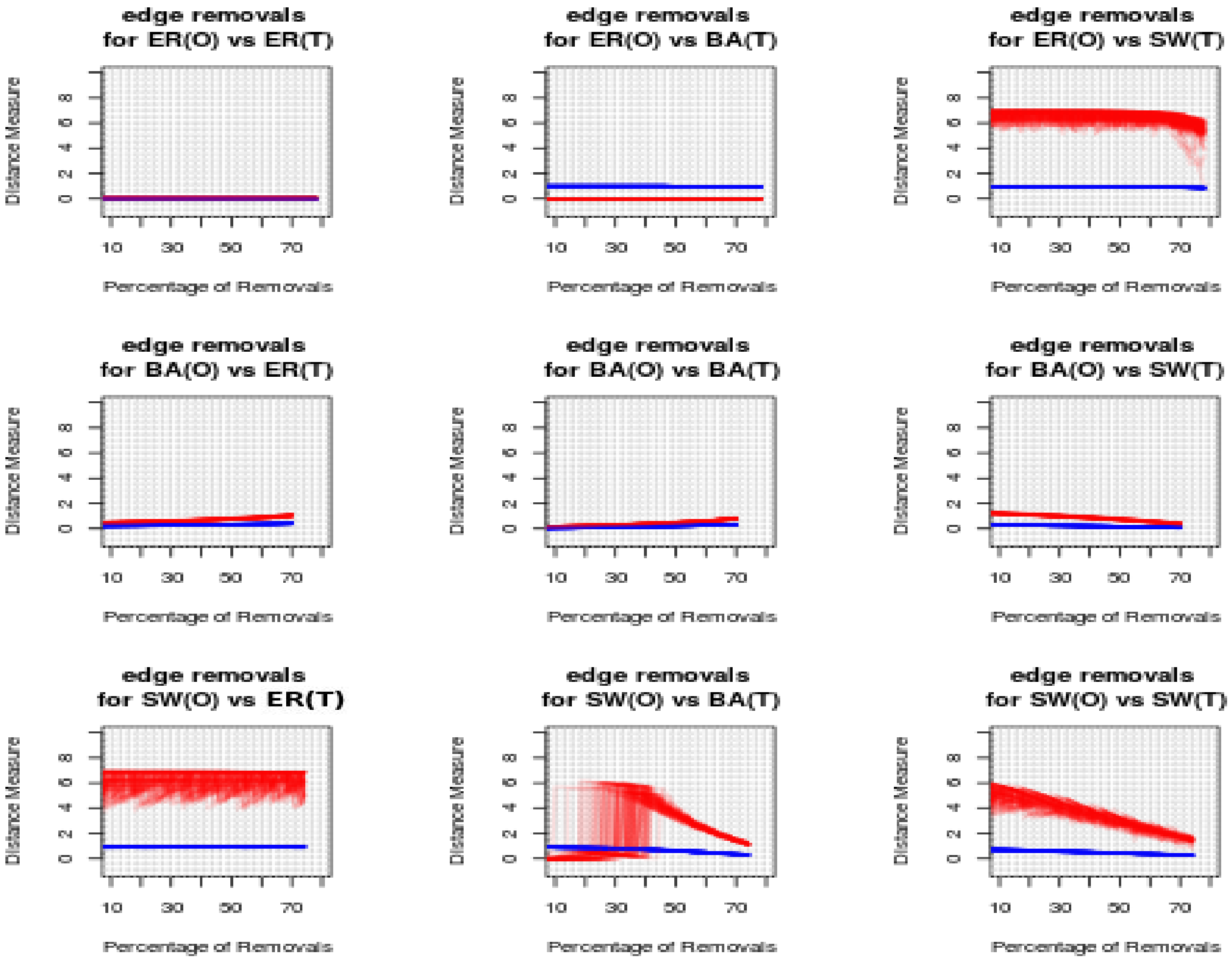


Figure 67. KL and H Statistic: Edge Removals, Starting Graph: 1000 Nodes, 10 Iterations, 10 Graphs, p = 0.2

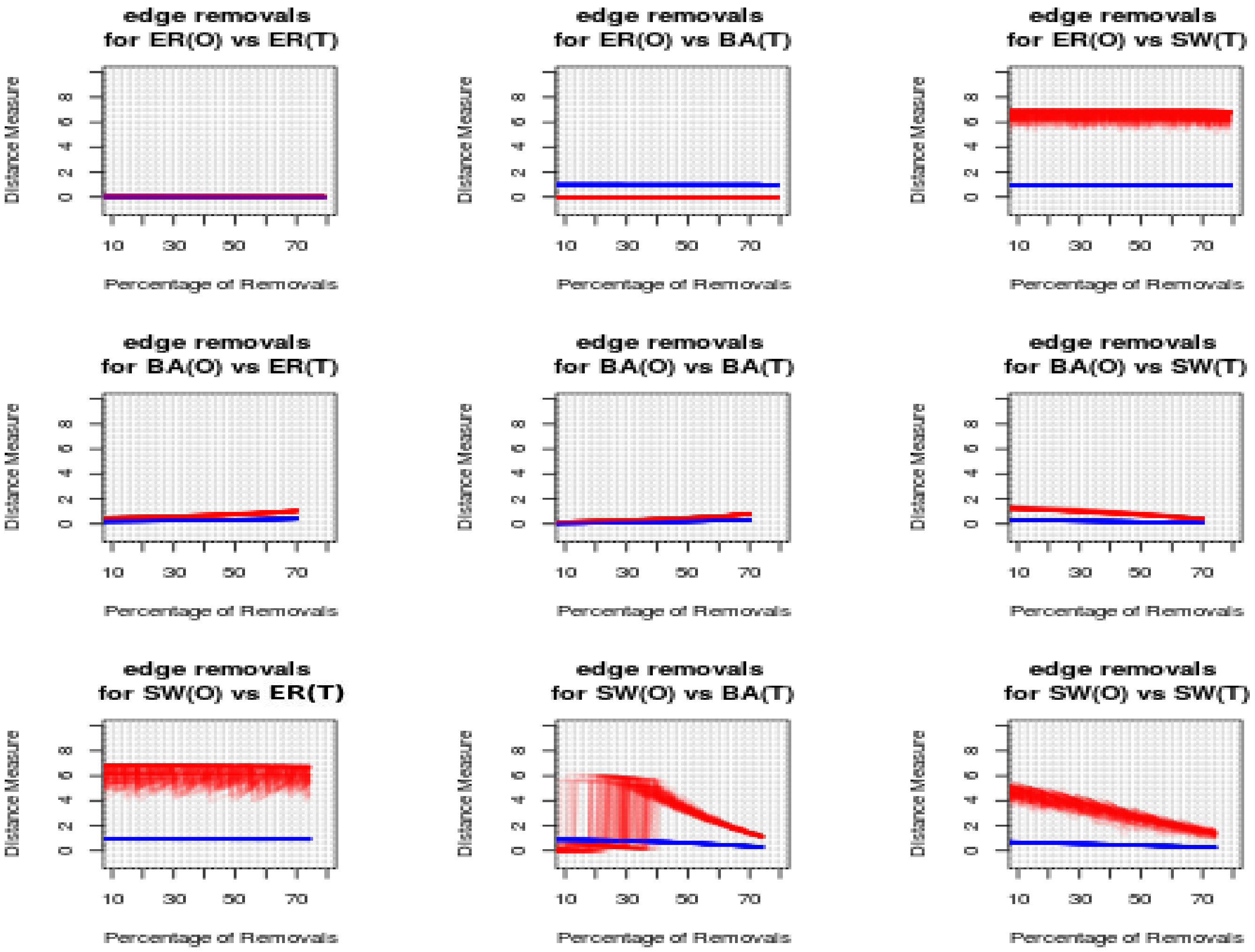


Figure 68. KL and H Statistic: Edge Removals, Starting Graph: 1000 Nodes, 10 Iterations, 10 Graphs, p = 0.3

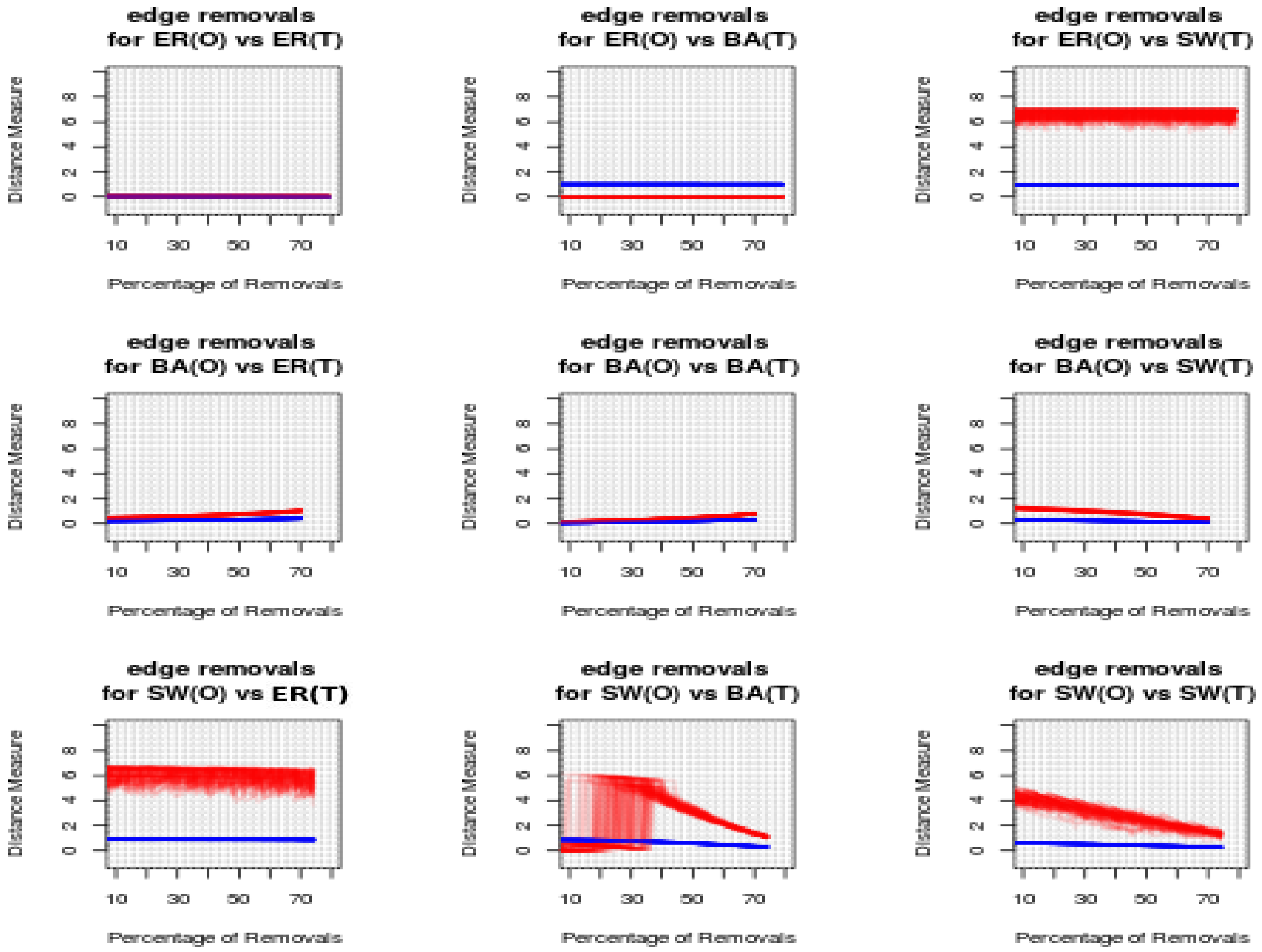
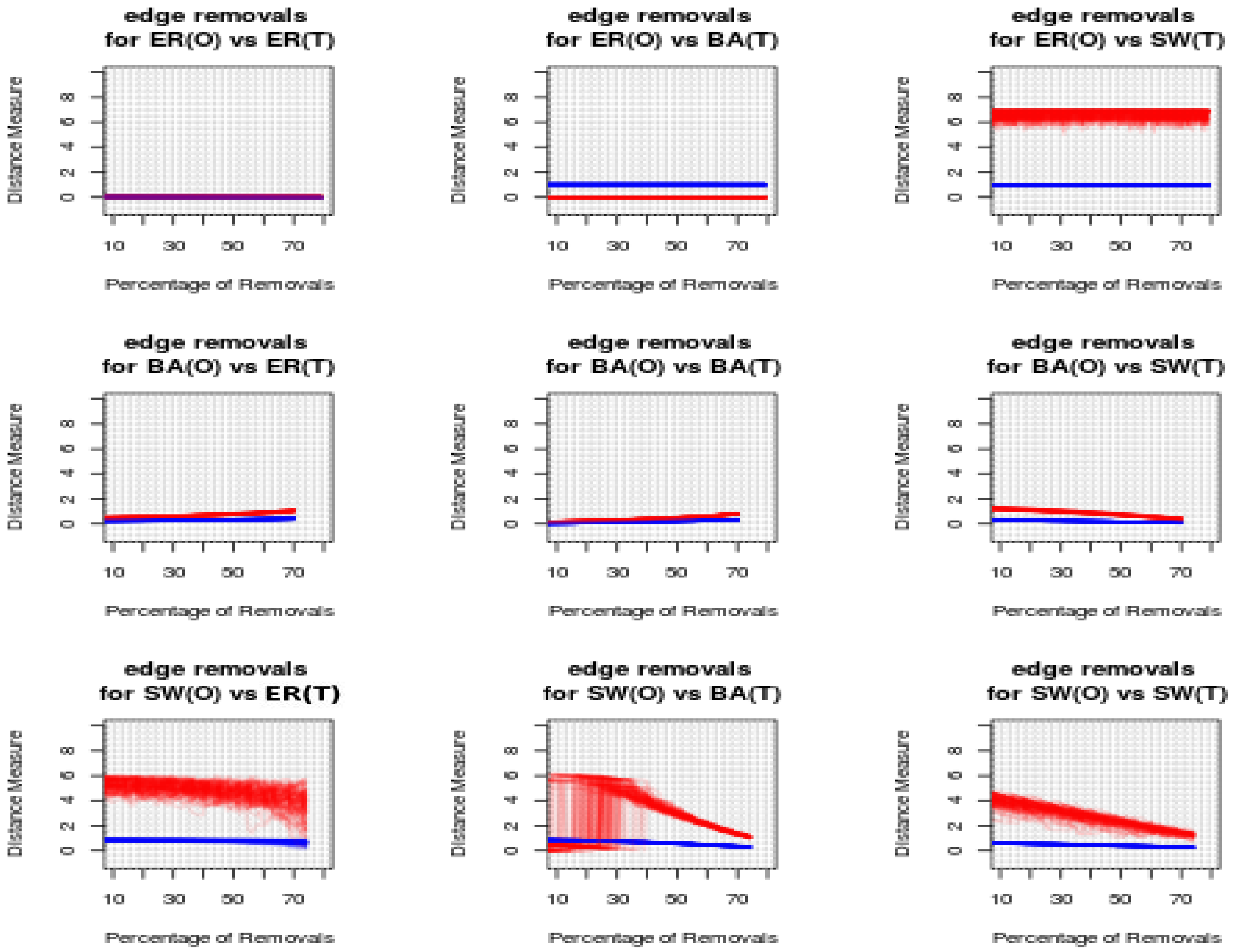


Figure 69. KL and H Statistic: Edge Removals, Starting Graph: 1000 Nodes, 10 Iterations, 10 Graphs, *p* = 0.4



edge removals for ER(O) vs ER(T)

edge removals for ER(O) vs BA(T)

edge removals for ER(O) vs SW(T)

edge removals for BA(O) vs ER(T)

edge removals for BA(O) vs BA(T)

edge removals for BA(O) vs SW(T)

edge removals for SW(O) vs ER(T)

edge removals for SW(O) vs BA(T)

edge removals for SW(O) vs SW(T)

Figure 70. KL and H Statistic: Edge Removals, Starting Graph: 1000 Nodes, 10 Iterations, 10 Graphs, p = 0.5

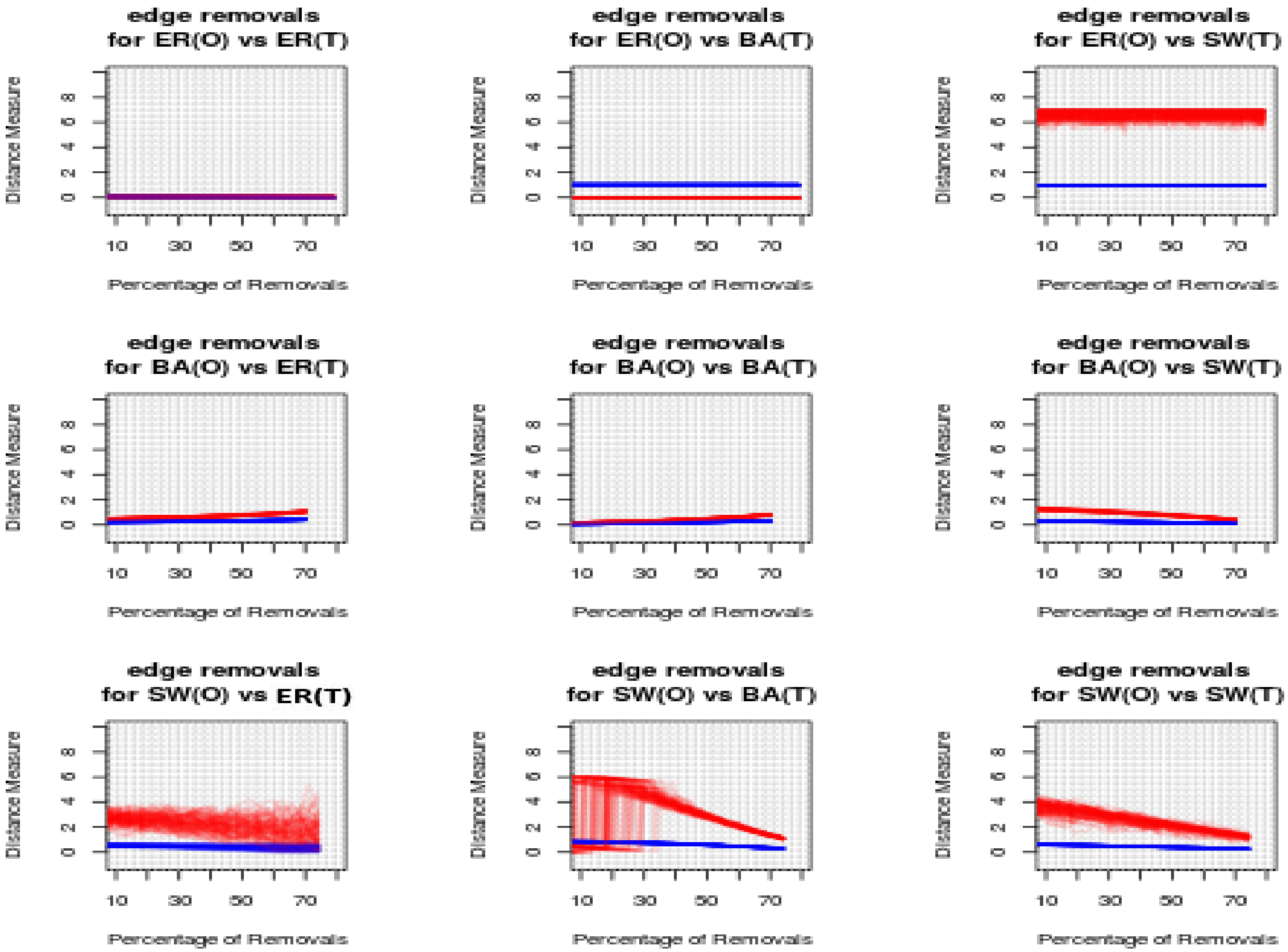


Figure 71. KL and H Statistic: Vertex Removals, Starting Graph: 1000 Nodes, 10 Iterations, 10 Graphs, *p* = 0.1

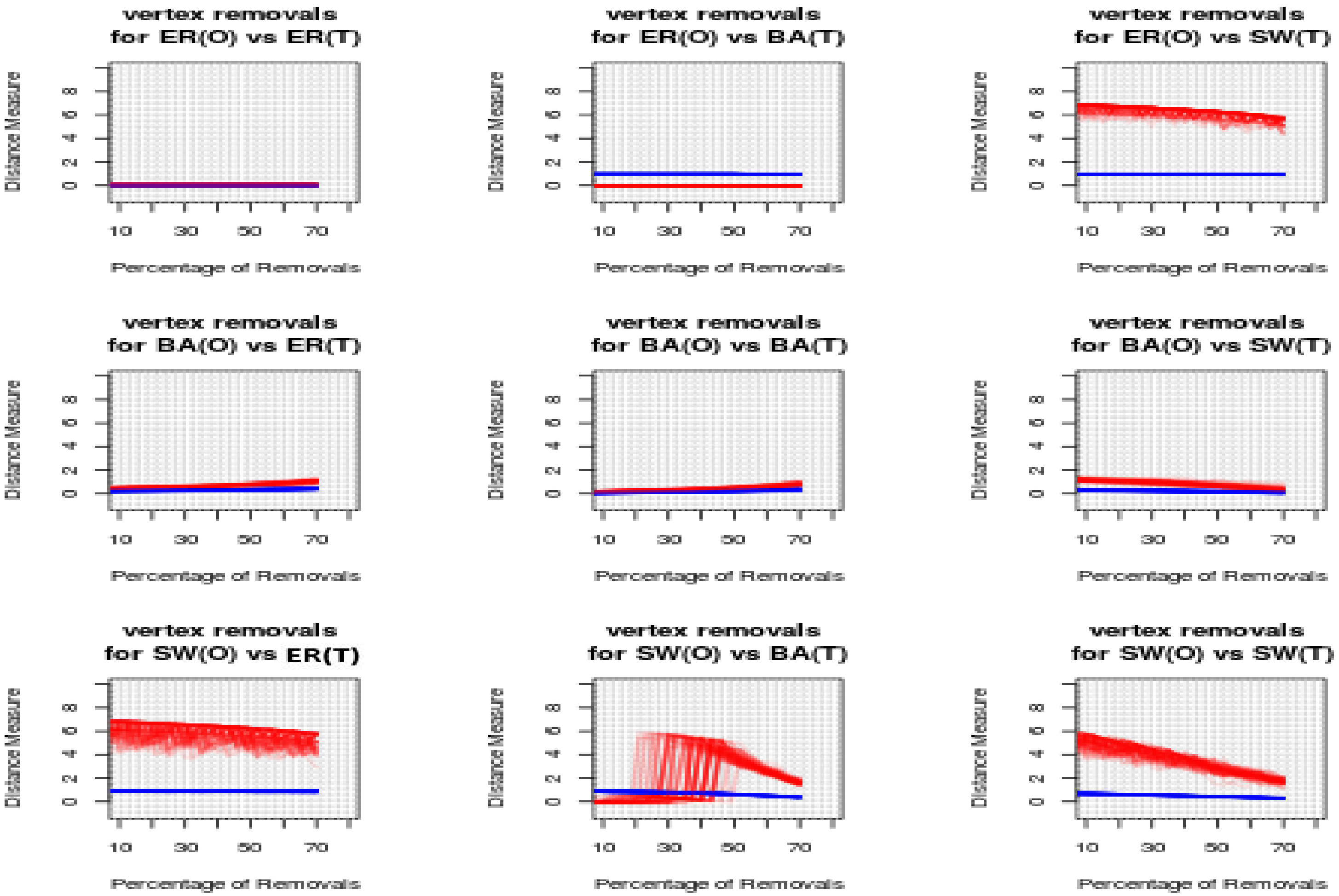


Figure 72. KL and H Statistic: Vertex Removals, Starting Graph: 1000 Nodes, 10 Iterations, 10 Graphs, *p* = 0.2

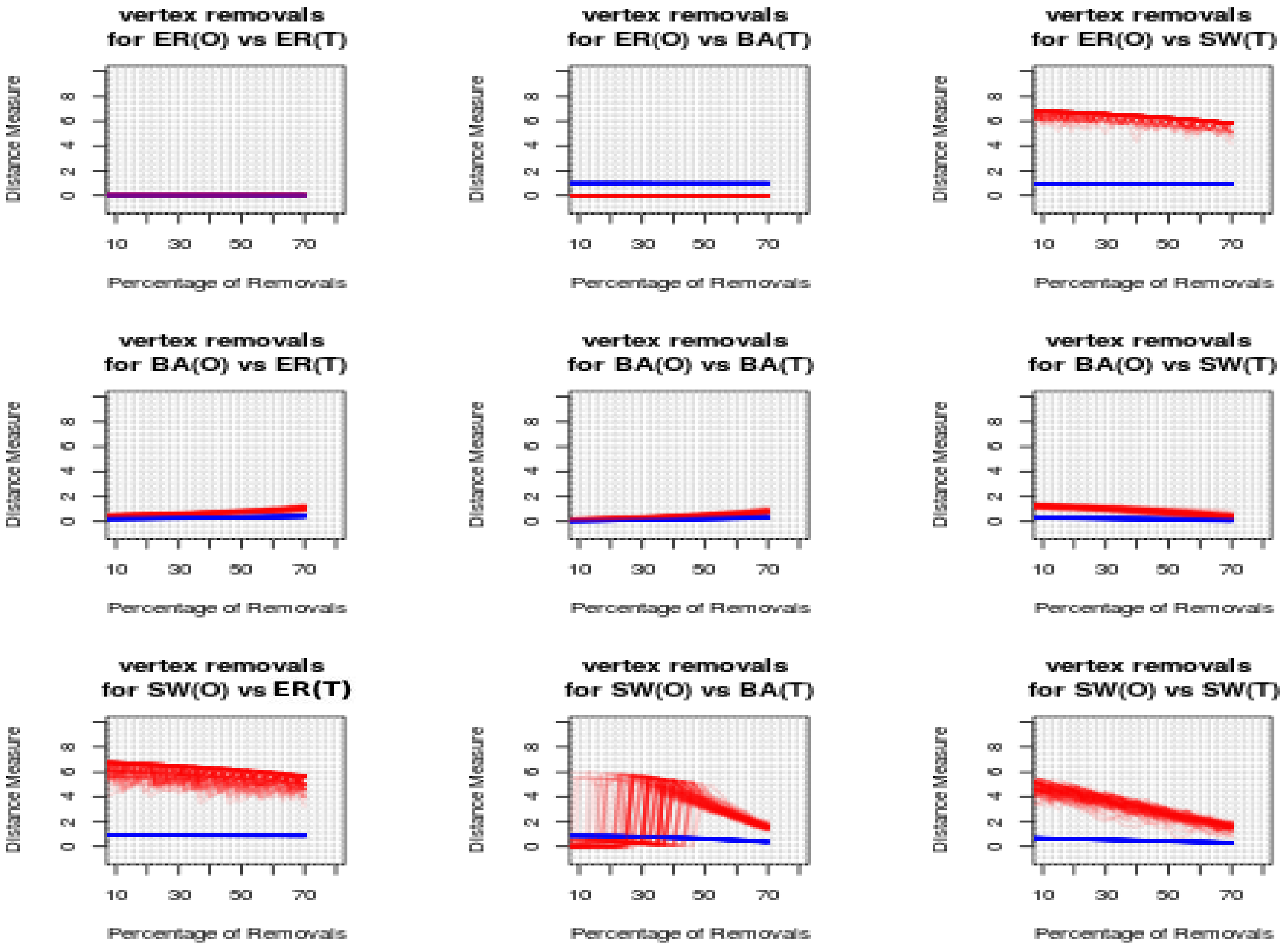


Figure 73. KL and H Statistic: Vertex Removals, Starting Graph: 1000 Nodes, 10 Iterations, 10 Graphs, $p = 0.3$

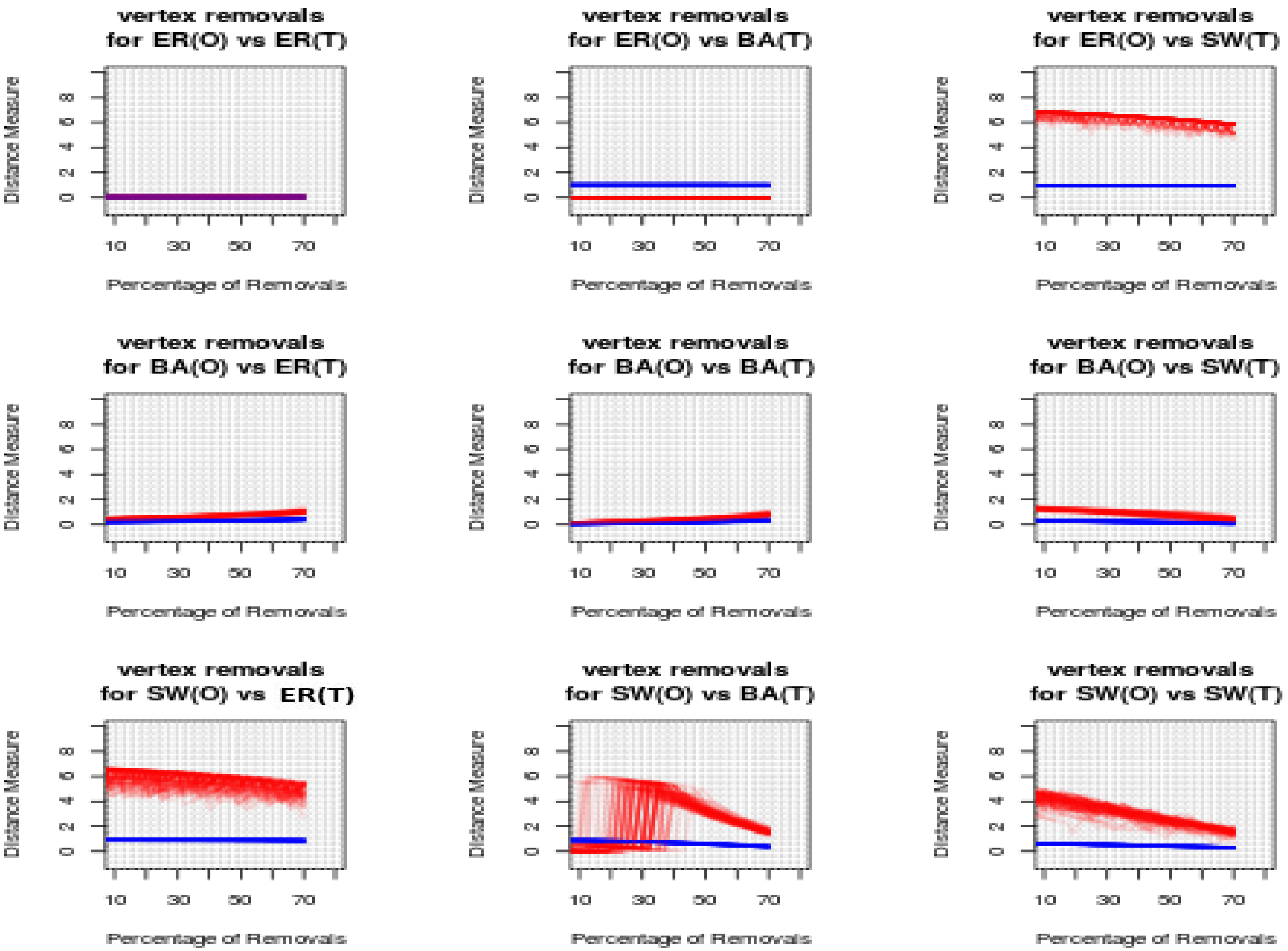


Figure 74. KL and H Statistic: Vertex Removals, Starting Graph: 1000 Nodes, 10 Iterations, 10 Graphs, p = 0.4

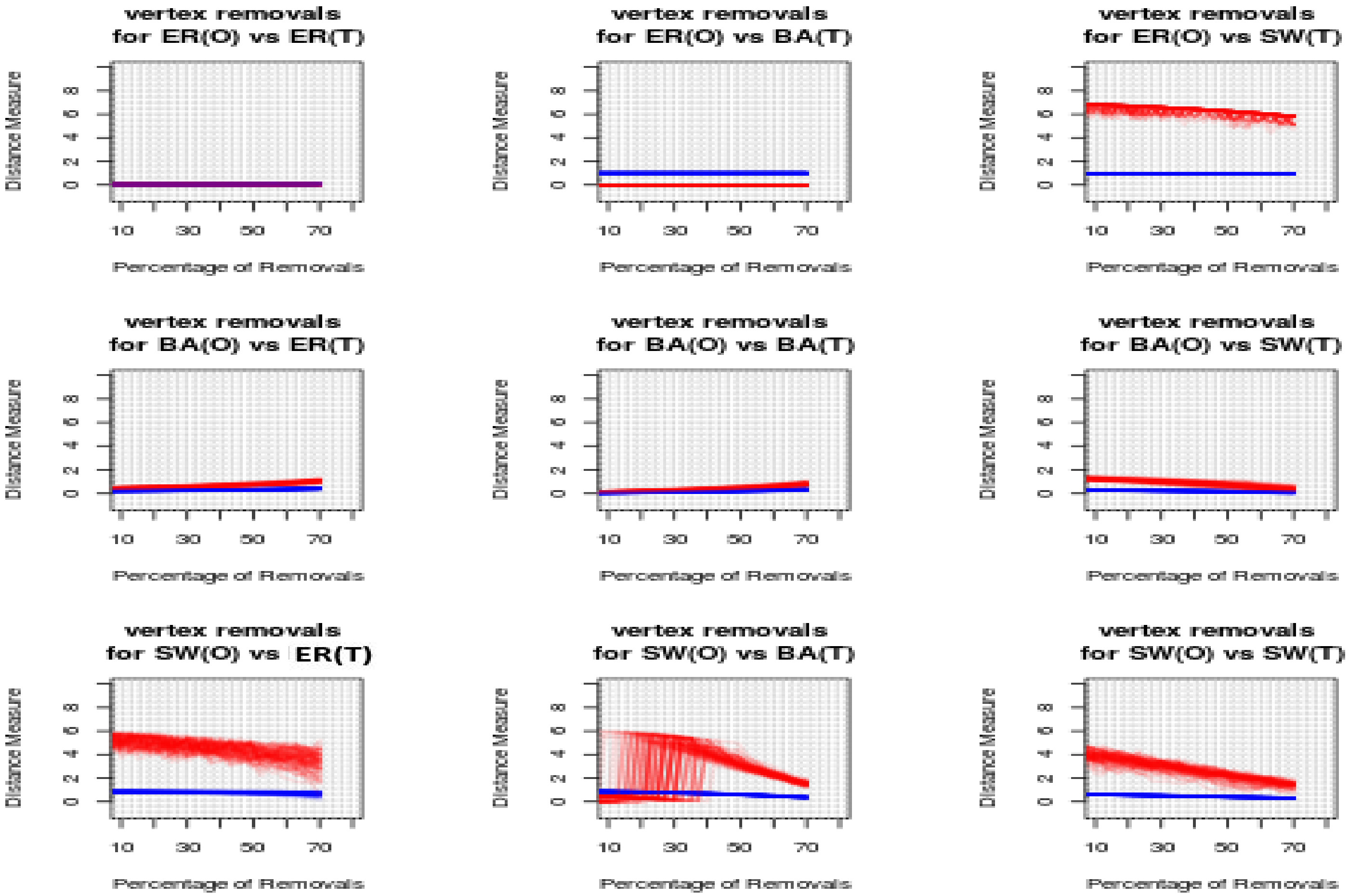
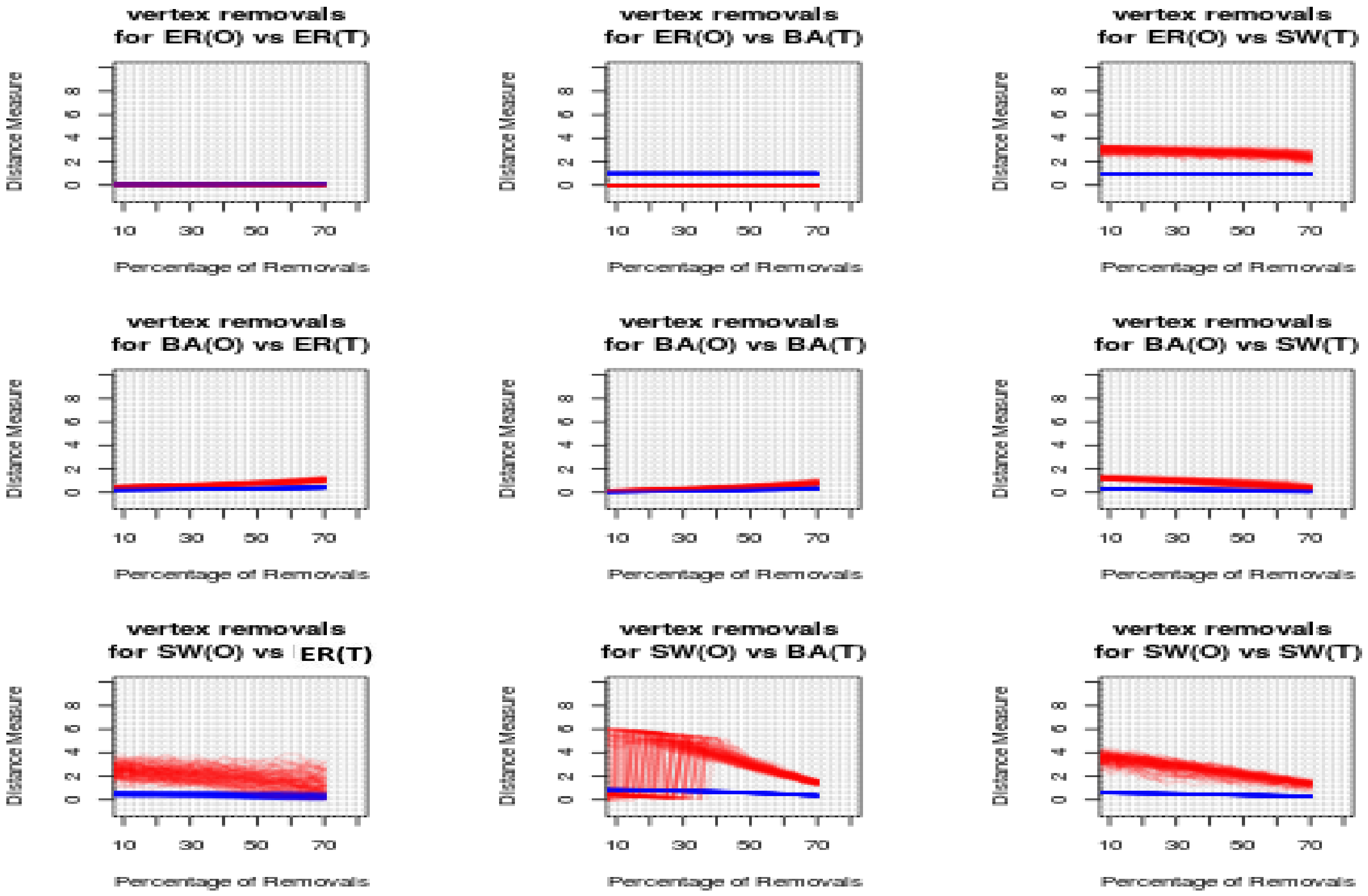


Figure 75. KL and H Statistic: Vertex Removals, Starting Graph: 1000 Nodes, 10 Iterations, 10 Graphs, p = 0.5



APPENDIX B. RESULTS FROM PREDICTIVE MODELING

Figure 76. Design of Experiments

S/N	Number of nodes	Edge Density	Proportion of Removal	No. of neighbors within which vertices are connected (SW Graph)	No. of Edges added in each time step	Power of Pref. Attachment (BA Graph)	Deletion Type	Graph Type
1	1000	0.1	0.41	3	3	2	edge	ER
2	1000	0.1	0.41	3	3	2	vertex	ER
3	1000	0.1	0.41	3	3	2	edge	SW
4	1000	0.1	0.41	3	3	2	vertex	SW
5	1000	0.1	0.41	3	3	2	edge	BA
6	1000	0.1	0.41	3	3	2	vertex	BA
7	916	0.5	0.19	4	2	1	edge	ER
8	916	0.5	0.19	4	2	1	vertex	ER
9	916	0.5	0.19	4	2	1	edge	SW
10	916	0.5	0.19	4	2	1	vertex	SW
11	916	0.5	0.19	4	2	1	edge	BA
12	916	0.5	0.19	4	2	1	vertex	BA
13	888	0.3	0.73	2	1	2	edge	ER
14	888	0.3	0.73	2	1	2	vertex	ER
15	888	0.3	0.73	2	1	2	edge	SW
16	888	0.3	0.73	2	1	2	vertex	SW
17	888	0.3	0.73	2	1	2	edge	BA
18	888	0.3	0.73	2	1	2	vertex	BA

S/N	Number of nodes	Edge Density	Proportion of Removal	No. of neighbors within which vertices are connected (SW Graph)	No. of Edges added in each time step	Power of Pref. Attachment (BA Graph)	Deletion Type	Graph Type
19	606	0.5	0.8	5	3	1	edge	ER
20	606	0.5	0.8	5	3	1	vertex	ER
21	606	0.5	0.8	5	3	1	edge	SW
22	606	0.5	0.8	5	3	1	vertex	SW
23	606	0.5	0.8	5	3	1	edge	BA
24	606	0.5	0.8	5	3	1	vertex	BA
25	944	0.1	0.43	3	2	2	edge	ER
26	944	0.1	0.43	3	2	2	vertex	ER
27	944	0.1	0.43	3	2	2	edge	SW
28	944	0.1	0.43	3	2	2	vertex	SW
29	944	0.1	0.43	3	2	2	edge	BA
30	944	0.1	0.43	3	2	2	vertex	BA
31	972	0.5	0.32	4	2	1	edge	ER
32	972	0.5	0.32	4	2	1	vertex	ER
33	972	0.5	0.32	4	2	1	edge	SW
34	972	0.5	0.32	4	2	1	vertex	SW
35	972	0.5	0.32	4	2	1	edge	BA
36	972	0.5	0.32	4	2	1	vertex	BA
37	719	0.3	0.78	3	1	2	edge	ER
38	719	0.3	0.78	3	1	2	vertex	ER
39	719	0.3	0.78	3	1	2	edge	SW
40	719	0.3	0.78	3	1	2	vertex	SW

S/N	Number of nodes	Edge Density	Proportion of Removal	No. of neighbors within which vertices are connected (SW Graph)	No. of Edges added in each time step	Power of Pref. Attachment (BA Graph)	Deletion Type	Graph Type
41	719	0.3	0.78	3	1	2	edge	BA
42	719	0.3	0.78	3	1	2	vertex	BA
43	578	0.4	0.76	4	3	2	edge	ER
44	578	0.4	0.76	4	3	2	vertex	ER
45	578	0.4	0.76	4	3	2	edge	SW
46	578	0.4	0.76	4	3	2	vertex	SW
47	578	0.4	0.76	4	3	2	edge	BA
48	578	0.4	0.76	4	3	2	vertex	BA
49	691	0.2	0.25	6	2	2	edge	ER
50	691	0.2	0.25	6	2	2	vertex	ER
51	691	0.2	0.25	6	2	2	edge	SW
52	691	0.2	0.25	6	2	2	vertex	SW
53	691	0.2	0.25	6	2	2	edge	BA
54	691	0.2	0.25	6	2	2	vertex	BA
55	775	0.4	0.3	7	1	2	edge	ER
56	775	0.4	0.3	7	1	2	vertex	ER
57	775	0.4	0.3	7	1	2	edge	SW
58	775	0.4	0.3	7	1	2	vertex	SW
59	775	0.4	0.3	7	1	2	edge	BA
60	775	0.4	0.3	7	1	2	vertex	BA
61	747	0.2	0.63	10	2	1	edge	ER
62	747	0.2	0.63	10	2	1	vertex	ER

S/N	Number of nodes	Edge Density	Proportion of Removal	No. of neighbors within which vertices are connected (SW Graph)	No. of Edges added in each time step	Power of Pref. Attachment (BA Graph)	Deletion Type	Graph Type
63	747	0.2	0.63	10	2	1	edge	SW
64	747	0.2	0.63	10	2	1	vertex	SW
65	747	0.2	0.63	10	2	1	edge	BA
66	747	0.2	0.63	10	2	1	vertex	BA
67	803	0.4	0.56	9	3	3	edge	ER
68	803	0.4	0.56	9	3	3	vertex	ER
69	803	0.4	0.56	9	3	3	edge	SW
70	803	0.4	0.56	9	3	3	vertex	SW
71	803	0.4	0.56	9	3	3	edge	BA
72	803	0.4	0.56	9	3	3	vertex	BA
73	634	0.2	0.23	6	2	1	edge	ER
74	634	0.2	0.23	6	2	1	vertex	ER
75	634	0.2	0.23	6	2	1	edge	SW
76	634	0.2	0.23	6	2	1	vertex	SW
77	634	0.2	0.23	6	2	1	edge	BA
78	634	0.2	0.23	6	2	1	vertex	BA
79	859	0.3	0.36	9	1	2	edge	ER
80	859	0.3	0.36	9	1	2	vertex	ER
81	859	0.3	0.36	9	1	2	edge	SW
82	859	0.3	0.36	9	1	2	vertex	SW
83	859	0.3	0.36	9	1	2	edge	BA
84	859	0.3	0.36	9	1	2	vertex	BA

S/N	Number of nodes	Edge Density	Proportion of Removal	No. of neighbors within which vertices are connected (SW Graph)	No. of Edges added in each time step	Power of Pref. Attachment (BA Graph)	Deletion Type	Graph Type
85	663	0.2	0.69	9	2	1	edge	ER
86	663	0.2	0.69	9	2	1	vertex	ER
87	663	0.2	0.69	9	2	1	edge	SW
88	663	0.2	0.69	9	2	1	vertex	SW
89	663	0.2	0.69	9	2	1	edge	BA
90	663	0.2	0.69	9	2	1	vertex	BA
91	831	0.4	0.52	10	3	3	edge	ER
92	831	0.4	0.52	10	3	3	vertex	ER
93	831	0.4	0.52	10	3	3	edge	SW
94	831	0.4	0.52	10	3	3	vertex	SW
95	831	0.4	0.52	10	3	3	edge	BA
96	831	0.4	0.52	10	3	3	vertex	BA
97	550	0.3	0.45	6	2	2	edge	ER
98	550	0.3	0.45	6	2	2	vertex	ER
99	550	0.3	0.45	6	2	2	edge	SW
100	550	0.3	0.45	6	2	2	vertex	SW
101	550	0.3	0.45	6	2	2	edge	BA
102	550	0.3	0.45	6	2	2	vertex	BA
103	100	0.5	0.49	8	1	2	edge	ER
104	100	0.5	0.49	8	1	2	vertex	ER
105	100	0.5	0.49	8	1	2	edge	SW
106	100	0.5	0.49	8	1	2	vertex	SW

S/N	Number of nodes	Edge Density	Proportion of Removal	No. of neighbors within which vertices are connected (SW Graph)	No. of Edges added in each time step	Power of Pref. Attachment (BA Graph)	Deletion Type	Graph Type
107	100	0.5	0.49	8	1	2	edge	BA
108	100	0.5	0.49	8	1	2	vertex	BA
109	184	0.1	0.71	7	2	3	edge	ER
110	184	0.1	0.71	7	2	3	vertex	ER
111	184	0.1	0.71	7	2	3	edge	SW
112	184	0.1	0.71	7	2	3	vertex	SW
113	184	0.1	0.71	7	2	3	edge	BA
114	184	0.1	0.71	7	2	3	vertex	BA
115	213	0.3	0.17	9	3	2	edge	ER
116	213	0.3	0.17	9	3	2	vertex	ER
117	213	0.3	0.17	9	3	2	edge	SW
118	213	0.3	0.17	9	3	2	vertex	SW
119	213	0.3	0.17	9	3	2	edge	BA
120	213	0.3	0.17	9	3	2	vertex	BA
121	494	0.2	0.1	6	1	3	edge	ER
122	494	0.2	0.1	6	1	3	vertex	ER
123	494	0.2	0.1	6	1	3	edge	SW
124	494	0.2	0.1	6	1	3	vertex	SW
125	494	0.2	0.1	6	1	3	edge	BA
126	494	0.2	0.1	6	1	3	vertex	BA
127	156	0.5	0.47	8	2	2	edge	ER
128	156	0.5	0.47	8	2	2	vertex	ER

S/N	Number of nodes	Edge Density	Proportion of Removal	No. of neighbors within which vertices are connected (SW Graph)	No. of Edges added in each time step	Power of Pref. Attachment (BA Graph)	Deletion Type	Graph Type
129	156	0.5	0.47	8	2	2	edge	SW
130	156	0.5	0.47	8	2	2	vertex	SW
131	156	0.5	0.47	8	2	2	edge	BA
132	156	0.5	0.47	8	2	2	vertex	BA
133	128	0.1	0.58	7	2	3	edge	ER
134	128	0.1	0.58	7	2	3	vertex	ER
135	128	0.1	0.58	7	2	3	edge	SW
136	128	0.1	0.58	7	2	3	vertex	SW
137	128	0.1	0.58	7	2	3	edge	BA
138	128	0.1	0.58	7	2	3	vertex	BA
139	381	0.3	0.12	8	3	2	edge	ER
140	381	0.3	0.12	8	3	2	vertex	ER
141	381	0.3	0.12	8	3	2	edge	SW
142	381	0.3	0.12	8	3	2	vertex	SW
143	381	0.3	0.12	8	3	2	edge	BA
144	381	0.3	0.12	8	3	2	vertex	BA
145	522	0.2	0.14	7	1	3	edge	ER
146	522	0.2	0.14	7	1	3	vertex	ER
147	522	0.2	0.14	7	1	3	edge	SW
148	522	0.2	0.14	7	1	3	vertex	SW
149	522	0.2	0.14	7	1	3	edge	BA
150	522	0.2	0.14	7	1	3	vertex	BA

S/N	Number of nodes	Edge Density	Proportion of Removal	No. of neighbors within which vertices are connected (SW Graph)	No. of Edges added in each time step	Power of Pref. Attachment (BA Graph)	Deletion Type	Graph Type
151	409	0.4	0.65	5	2	2	edge	ER
152	409	0.4	0.65	5	2	2	vertex	ER
153	409	0.4	0.65	5	2	2	edge	SW
154	409	0.4	0.65	5	2	2	vertex	SW
155	409	0.4	0.65	5	2	2	edge	BA
156	409	0.4	0.65	5	2	2	vertex	BA
157	325	0.2	0.6	4	3	2	edge	ER
158	325	0.2	0.6	4	3	2	vertex	ER
159	325	0.2	0.6	4	3	2	edge	SW
160	325	0.2	0.6	4	3	2	vertex	SW
161	325	0.2	0.6	4	3	2	edge	BA
162	325	0.2	0.6	4	3	2	vertex	BA
163	353	0.4	0.28	1	2	3	edge	ER
164	353	0.4	0.28	1	2	3	vertex	ER
165	353	0.4	0.28	1	2	3	edge	SW
166	353	0.4	0.28	1	2	3	vertex	SW
167	353	0.4	0.28	1	2	3	edge	BA
168	353	0.4	0.28	1	2	3	vertex	BA
169	297	0.2	0.34	2	2	1	edge	ER
170	297	0.2	0.34	2	2	1	vertex	ER
171	297	0.2	0.34	2	2	1	edge	SW
172	297	0.2	0.34	2	2	1	vertex	SW

S/N	Number of nodes	Edge Density	Proportion of Removal	No. of neighbors within which vertices are connected (SW Graph)	No. of Edges added in each time step	Power of Pref. Attachment (BA Graph)	Deletion Type	Graph Type
173	297	0.2	0.34	2	2	1	edge	BA
174	297	0.2	0.34	2	2	1	vertex	BA
175	466	0.4	0.67	5	2	3	edge	ER
176	466	0.4	0.67	5	2	3	vertex	ER
177	466	0.4	0.67	5	2	3	edge	SW
178	466	0.4	0.67	5	2	3	vertex	SW
179	466	0.4	0.67	5	2	3	edge	BA
180	466	0.4	0.67	5	2	3	vertex	BA
181	241	0.3	0.54	2	3	2	edge	ER
182	241	0.3	0.54	2	3	2	vertex	ER
183	241	0.3	0.54	2	3	2	edge	SW
184	241	0.3	0.54	2	3	2	vertex	SW
185	241	0.3	0.54	2	3	2	edge	BA
186	241	0.3	0.54	2	3	2	vertex	BA
187	438	0.4	0.21	2	2	3	edge	ER
188	438	0.4	0.21	2	2	3	vertex	ER
189	438	0.4	0.21	2	2	3	edge	SW
190	438	0.4	0.21	2	2	3	vertex	SW
191	438	0.4	0.21	2	2	3	edge	BA
192	438	0.4	0.21	2	2	3	vertex	BA
193	269	0.3	0.38	1	1	1	edge	ER
194	269	0.3	0.38	1	1	1	vertex	ER

S/N	Number of nodes	Edge Density	Proportion of Removal	No. of neighbors within which vertices are connected (SW Graph)	No. of Edges added in each time step	Power of Pref. Attachment (BA Graph)	Deletion Type	Graph Type
195	269	0.3	0.38	1	1	1	edge	SW
196	269	0.3	0.38	1	1	1	vertex	SW
197	269	0.3	0.38	1	1	1	edge	BA
198	269	0.3	0.38	1	1	1	vertex	BA
199	184	0.3	0.23	9	2	2	edge	ER
200	184	0.3	0.23	9	2	2	vertex	ER
201	184	0.3	0.23	9	2	2	edge	SW
202	184	0.3	0.23	9	2	2	vertex	SW
203	184	0.3	0.23	9	2	2	edge	BA
204	184	0.3	0.23	9	2	2	vertex	BA
205	1000	0.2	0.36	5	1	3	edge	ER
206	1000	0.2	0.36	5	1	3	vertex	ER
207	1000	0.2	0.36	5	1	3	edge	SW
208	1000	0.2	0.36	5	1	3	vertex	SW
209	1000	0.2	0.36	5	1	3	edge	BA
210	1000	0.2	0.36	5	1	3	vertex	BA
211	494	0.5	0.21	1	2	2	edge	ER
212	494	0.5	0.21	1	2	2	vertex	ER
213	494	0.5	0.21	1	2	2	edge	SW
214	494	0.5	0.21	1	2	2	vertex	SW
215	494	0.5	0.21	1	2	2	edge	BA
216	494	0.5	0.21	1	2	2	vertex	BA

S/N	Number of nodes	Edge Density	Proportion of Removal	No. of neighbors within which vertices are connected (SW Graph)	No. of Edges added in each time step	Power of Pref. Attachment (BA Graph)	Deletion Type	Graph Type
217	888	0.5	0.38	9	1	3	edge	ER
218	888	0.5	0.38	9	1	3	vertex	ER
219	888	0.5	0.38	9	1	3	edge	SW
220	888	0.5	0.38	9	1	3	vertex	SW
221	888	0.5	0.38	9	1	3	edge	BA
222	888	0.5	0.38	9	1	3	vertex	BA
223	128	0.3	0.25	7	2	2	edge	ER
224	128	0.3	0.25	7	2	2	vertex	ER
225	128	0.3	0.25	7	2	2	edge	SW
226	128	0.3	0.25	7	2	2	vertex	SW
227	128	0.3	0.25	7	2	2	edge	BA
228	128	0.3	0.25	7	2	2	vertex	BA
229	944	0.2	0.3	5	1	1	edge	ER
230	944	0.2	0.3	5	1	1	vertex	ER
231	944	0.2	0.3	5	1	1	edge	SW
232	944	0.2	0.3	5	1	1	vertex	SW
233	944	0.2	0.3	5	1	1	edge	BA
234	944	0.2	0.3	5	1	1	vertex	BA
235	522	0.5	0.28	1	2	2	edge	ER
236	522	0.5	0.28	1	2	2	vertex	ER
237	522	0.5	0.28	1	2	2	edge	SW
238	522	0.5	0.28	1	2	2	vertex	SW

S/N	Number of nodes	Edge Density	Proportion of Removal	No. of neighbors within which vertices are connected (SW Graph)	No. of Edges added in each time step	Power of Pref. Attachment (BA Graph)	Deletion Type	Graph Type
239	522	0.5	0.28	1	2	2	edge	BA
240	522	0.5	0.28	1	2	2	vertex	BA
241	719	0.5	0.34	9	2	1	edge	ER
242	719	0.5	0.34	9	2	1	vertex	ER
243	719	0.5	0.34	9	2	1	edge	SW
244	719	0.5	0.34	9	2	1	vertex	SW
245	719	0.5	0.34	9	2	1	edge	BA
246	719	0.5	0.34	9	2	1	vertex	BA
247	325	0.2	0.47	7	2	1	edge	ER
248	325	0.2	0.47	7	2	1	vertex	ER
249	325	0.2	0.47	7	2	1	edge	SW
250	325	0.2	0.47	7	2	1	vertex	SW
251	325	0.2	0.47	7	2	1	edge	BA
252	325	0.2	0.47	7	2	1	vertex	BA
253	691	0.2	0.58	3	2	1	edge	ER
254	691	0.2	0.58	3	2	1	vertex	ER
255	691	0.2	0.58	3	2	1	edge	SW
256	691	0.2	0.58	3	2	1	vertex	SW
257	691	0.2	0.58	3	2	1	edge	BA
258	691	0.2	0.58	3	2	1	vertex	BA
259	297	0.4	0.78	4	1	1	edge	ER
260	297	0.4	0.78	4	1	1	vertex	ER

S/N	Number of nodes	Edge Density	Proportion of Removal	No. of neighbors within which vertices are connected (SW Graph)	No. of Edges added in each time step	Power of Pref. Attachment (BA Graph)	Deletion Type	Graph Type
261	297	0.4	0.78	4	1	1	edge	SW
262	297	0.4	0.78	4	1	1	vertex	SW
263	297	0.4	0.78	4	1	1	edge	BA
264	297	0.4	0.78	4	1	1	vertex	BA
265	747	0.4	0.76	8	3	2	edge	ER
266	747	0.4	0.76	8	3	2	vertex	ER
267	747	0.4	0.76	8	3	2	edge	SW
268	747	0.4	0.76	8	3	2	vertex	SW
269	747	0.4	0.76	8	3	2	edge	BA
270	747	0.4	0.76	8	3	2	vertex	BA
271	241	0.2	0.49	6	1	3	edge	ER
272	241	0.2	0.49	6	1	3	vertex	ER
273	241	0.2	0.49	6	1	3	edge	SW
274	241	0.2	0.49	6	1	3	vertex	SW
275	241	0.2	0.49	6	1	3	edge	BA
276	241	0.2	0.49	6	1	3	vertex	BA
277	634	0.3	0.71	2	2	3	edge	ER
278	634	0.3	0.71	2	2	3	vertex	ER
279	634	0.3	0.71	2	2	3	edge	SW
280	634	0.3	0.71	2	2	3	vertex	SW
281	634	0.3	0.71	2	2	3	edge	BA
282	634	0.3	0.71	2	2	3	vertex	BA

S/N	Number of nodes	Edge Density	Proportion of Removal	No. of neighbors within which vertices are connected (SW Graph)	No. of Edges added in each time step	Power of Pref. Attachment (BA Graph)	Deletion Type	Graph Type
283	269	0.4	0.73	4	1	2	edge	ER
284	269	0.4	0.73	4	1	2	vertex	ER
285	269	0.4	0.73	4	1	2	edge	SW
286	269	0.4	0.73	4	1	2	vertex	SW
287	269	0.4	0.73	4	1	2	edge	BA
288	269	0.4	0.73	4	1	2	vertex	BA
289	663	0.3	0.8	8	3	2	edge	ER
290	663	0.3	0.8	8	3	2	vertex	ER
291	663	0.3	0.8	8	3	2	edge	SW
292	663	0.3	0.8	8	3	2	vertex	SW
293	663	0.3	0.8	8	3	2	edge	BA
294	663	0.3	0.8	8	3	2	vertex	BA
295	916	0.3	0.67	2	2	2	edge	ER
296	916	0.3	0.67	2	2	2	vertex	ER
297	916	0.3	0.67	2	2	2	edge	SW
298	916	0.3	0.67	2	2	2	vertex	SW
299	916	0.3	0.67	2	2	2	edge	BA
300	916	0.3	0.67	2	2	2	vertex	BA
301	100	0.5	0.54	6	3	2	edge	ER
302	100	0.5	0.54	6	3	2	vertex	ER
303	100	0.5	0.54	6	3	2	edge	SW
304	100	0.5	0.54	6	3	2	vertex	SW

S/N	Number of nodes	Edge Density	Proportion of Removal	No. of neighbors within which vertices are connected (SW Graph)	No. of Edges added in each time step	Power of Pref. Attachment (BA Graph)	Deletion Type	Graph Type
305	100	0.5	0.54	6	3	2	edge	BA
306	100	0.5	0.54	6	3	2	vertex	BA
307	606	0.1	0.69	10	2	2	edge	ER
308	606	0.1	0.69	10	2	2	vertex	ER
309	606	0.1	0.69	10	2	2	edge	SW
310	606	0.1	0.69	10	2	2	vertex	SW
311	606	0.1	0.69	10	2	2	edge	BA
312	606	0.1	0.69	10	2	2	vertex	BA
313	213	0.1	0.52	2	3	1	edge	ER
314	213	0.1	0.52	2	3	1	vertex	ER
315	213	0.1	0.52	2	3	1	edge	SW
316	213	0.1	0.52	2	3	1	vertex	SW
317	213	0.1	0.52	2	3	1	edge	BA
318	213	0.1	0.52	2	3	1	vertex	BA
319	972	0.3	0.65	4	2	2	edge	ER
320	972	0.3	0.65	4	2	2	vertex	ER
321	972	0.3	0.65	4	2	2	edge	SW
322	972	0.3	0.65	4	2	2	vertex	SW
323	972	0.3	0.65	4	2	2	edge	BA
324	972	0.3	0.65	4	2	2	vertex	BA
325	156	0.4	0.6	6	3	3	edge	ER
326	156	0.4	0.6	6	3	3	vertex	ER

S/N	Number of nodes	Edge Density	Proportion of Removal	No. of neighbors within which vertices are connected (SW Graph)	No. of Edges added in each time step	Power of Pref. Attachment (BA Graph)	Deletion Type	Graph Type
327	156	0.4	0.6	6	3	3	edge	SW
328	156	0.4	0.6	6	3	3	vertex	SW
329	156	0.4	0.6	6	3	3	edge	BA
330	156	0.4	0.6	6	3	3	vertex	BA
331	578	0.1	0.63	10	2	2	edge	ER
332	578	0.1	0.63	10	2	2	vertex	ER
333	578	0.1	0.63	10	2	2	edge	SW
334	578	0.1	0.63	10	2	2	vertex	SW
335	578	0.1	0.63	10	2	2	edge	BA
336	578	0.1	0.63	10	2	2	vertex	BA
337	381	0.1	0.56	2	3	3	edge	ER
338	381	0.1	0.56	2	3	3	vertex	ER
339	381	0.1	0.56	2	3	3	edge	SW
340	381	0.1	0.56	2	3	3	vertex	SW
341	381	0.1	0.56	2	3	3	edge	BA
342	381	0.1	0.56	2	3	3	vertex	BA
343	775	0.4	0.43	4	2	3	edge	ER
344	775	0.4	0.43	4	2	3	vertex	ER
345	775	0.4	0.43	4	2	3	edge	SW
346	775	0.4	0.43	4	2	3	vertex	SW
347	775	0.4	0.43	4	2	3	edge	BA
348	775	0.4	0.43	4	2	3	vertex	BA

S/N	Number of nodes	Edge Density	Proportion of Removal	No. of neighbors within which vertices are connected (SW Graph)	No. of Edges added in each time step	Power of Pref. Attachment (BA Graph)	Deletion Type	Graph Type
349	409	0.4	0.32	8	2	3	edge	ER
350	409	0.4	0.32	8	2	3	vertex	ER
351	409	0.4	0.32	8	2	3	edge	SW
352	409	0.4	0.32	8	2	3	vertex	SW
353	409	0.4	0.32	8	2	3	edge	BA
354	409	0.4	0.32	8	2	3	vertex	BA
355	803	0.2	0.12	7	3	3	edge	ER
356	803	0.2	0.12	7	3	3	vertex	ER
357	803	0.2	0.12	7	3	3	edge	SW
358	803	0.2	0.12	7	3	3	vertex	SW
359	803	0.2	0.12	7	3	3	edge	BA
360	803	0.2	0.12	7	3	3	vertex	BA
361	353	0.2	0.14	3	1	2	edge	ER
362	353	0.2	0.14	3	1	2	vertex	ER
363	353	0.2	0.14	3	1	2	edge	SW
364	353	0.2	0.14	3	1	2	vertex	SW
365	353	0.2	0.14	3	1	2	edge	BA
366	353	0.2	0.14	3	1	2	vertex	BA
367	859	0.4	0.41	5	3	1	edge	ER
368	859	0.4	0.41	5	3	1	vertex	ER
369	859	0.4	0.41	5	3	1	edge	SW
370	859	0.4	0.41	5	3	1	vertex	SW

S/N	Number of nodes	Edge Density	Proportion of Removal	No. of neighbors within which vertices are connected (SW Graph)	No. of Edges added in each time step	Power of Pref. Attachment (BA Graph)	Deletion Type	Graph Type
371	859	0.4	0.41	5	3	1	edge	BA
372	859	0.4	0.41	5	3	1	vertex	BA
373	466	0.4	0.19	9	2	1	edge	ER
374	466	0.4	0.19	9	2	1	vertex	ER
375	466	0.4	0.19	9	2	1	edge	SW
376	466	0.4	0.19	9	2	1	vertex	SW
377	466	0.4	0.19	9	2	1	edge	BA
378	466	0.4	0.19	9	2	1	vertex	BA
379	831	0.2	0.17	7	3	2	edge	ER
380	831	0.2	0.17	7	3	2	vertex	ER
381	831	0.2	0.17	7	3	2	edge	SW
382	831	0.2	0.17	7	3	2	vertex	SW
383	831	0.2	0.17	7	3	2	edge	BA
384	831	0.2	0.17	7	3	2	vertex	BA
385	438	0.3	0.1	3	1	2	edge	ER
386	438	0.3	0.1	3	1	2	vertex	ER
387	438	0.3	0.1	3	1	2	edge	SW
388	438	0.3	0.1	3	1	2	vertex	SW
389	438	0.3	0.1	3	1	2	edge	BA
390	438	0.3	0.1	3	1	2	vertex	BA

APPENDIX C. ANALYSIS OF RESULTS FROM DESCRIPTIVE MODELING

Note: This appendix is a partial reproduction of the work by Ruriko Yoshida on Average Distance between Nodes in a Random Graph (2018), an unpublished working paper at the time this thesis is published. Its results support the findings in the results from the descriptive analysis in Chapter 4 on mean distance in ER graphs under conditions of information loss.

(1) Average Distance between Nodes in a Random Graph by Ruriko Yoshida (2018)

Suppose we have a random graph $G_0 = (N_0, E_0)$ where N_0 is the set of nodes (vertices) $N_0 = \{1, ..., n\}$ and a set of edges E_0. Let G_i^E be a graph with N_0 and the edge set $E_i \subset E_0$ such that *i* many edges are randomly (uniformly) deleted. Let G_i^N be a graph with the node set $N_i \subset N_0$ and edge set $E_i \subset E_0$ such that *i* many nodes are randomly (uniformly) deleted and also edges adjacent to the deleted nodes. Without loss of generality, let $N_i = \{1, ..., (N - i)\}$. We assume here $G_0 = (N_0, E_0)$ is generated by the ER model.

Properties:

The main ingredient of the proof for our theorem is from (Chung and Lu 2002).

Suppose we have a degree distribution.

$$w = (w_0, w_1, ..., w_n)$$

will be the expected degree of the node *i*.

Let $d = \frac{\sum_i w_i^2}{\sum_i w_i}$, that is, the second order average degree of nodes.

Definition:

The volume of a subset of nodes $S \subset N$ in a graph $G = (N, E)$ is defined as

$$Vol(S) = \sum_{v \in S} \deg(v)$$

where $\deg(v)$ is the degree of a node v.

Let $Vol_k(S) = \sum_{i \in S} w_i^k$ and $Vol_k(G) = \sum_{i \in N} w_i^k$

Definition:

The expected degree sequence w for a graph G is called *strongly sparse* if G satisfies the following:

The second order average degree *d* satisfies the condition

$$0 < \log(d) << \log(n)$$

The average expected degree is strictly greater than $1+\varepsilon$ for some positive value ε which is independent of the number of nodes *n* in G.

Note that if G is generated under the ER model with p < 1 then it is admissible.

Theorem 1: (Theorem 1 in (Chung and Lu [2002]**)**

For a random graph G with admissible expected degree sequence $(w_1, ..., w_n)$, the average distance is almost surely $(1+o(1))(\log(n)/\log(d))$.

Proposition 1. *The expected degree of each node for a graph* $G = (N, E)$ *with* n *nodes generated under Erdos-Renyi model with p,* $p \in [0, 1]$ *is* $p \cdot (n-1)$.

Proof. If $p = 1$, then G is the complete graph with n nodes. This means that the degree of each node is $(n-1)$. If $p < 1$, then the probability to be an edge between a node $i \in N$ to another node $j \neq i$ is p. Therefore, since there are *(n-1)* possible $j \neq i$, the average degree of the node *i* is *p(n - 1)*. QED.

Theorem 2. *If a graph $G = (N, E)$ with n nodes generated under Erdos-Renyi model with $p \in (0, 1)$, then the average distance is almost surely* $(1 + o(1))(\log(n)/\log(p \cdot (n - 1)))$.

Proof. Using Proposition 4, the expected degree sequence has $w_i = p \cdot (n - 1)$ for $i \in N$. Then we have $d = p \cdot (n - 1)$. Using Theorem 3, since w is admissible, we are done.

Corollary 1. Suppose a graph $G_0 = (N_0, E_0)$ with n nodes generated under Erdos-Renyi model with $p \in (0, 1)$. If $i << n$, then the average distance for a graph G_i^N is almost surely

$$(1 + o(1))(\log(n - i) / \log(p.(n - 1 - i)))$$

Proof. It is immediately proven by Theorem 2. QED.

Theorem 3. Suppose a graph $G_0 = (N_0, E_0)$ with n nodes generated under Erdos-Renyi model with $p \in (0, 1)$. The average distance for a graph G_i^E is almost surely

$$(1 + o(1))(\log(n) / \log(\hat{p}.(n - 1)))$$

where $\hat{p} = |E_i| / \binom{n}{2}$.

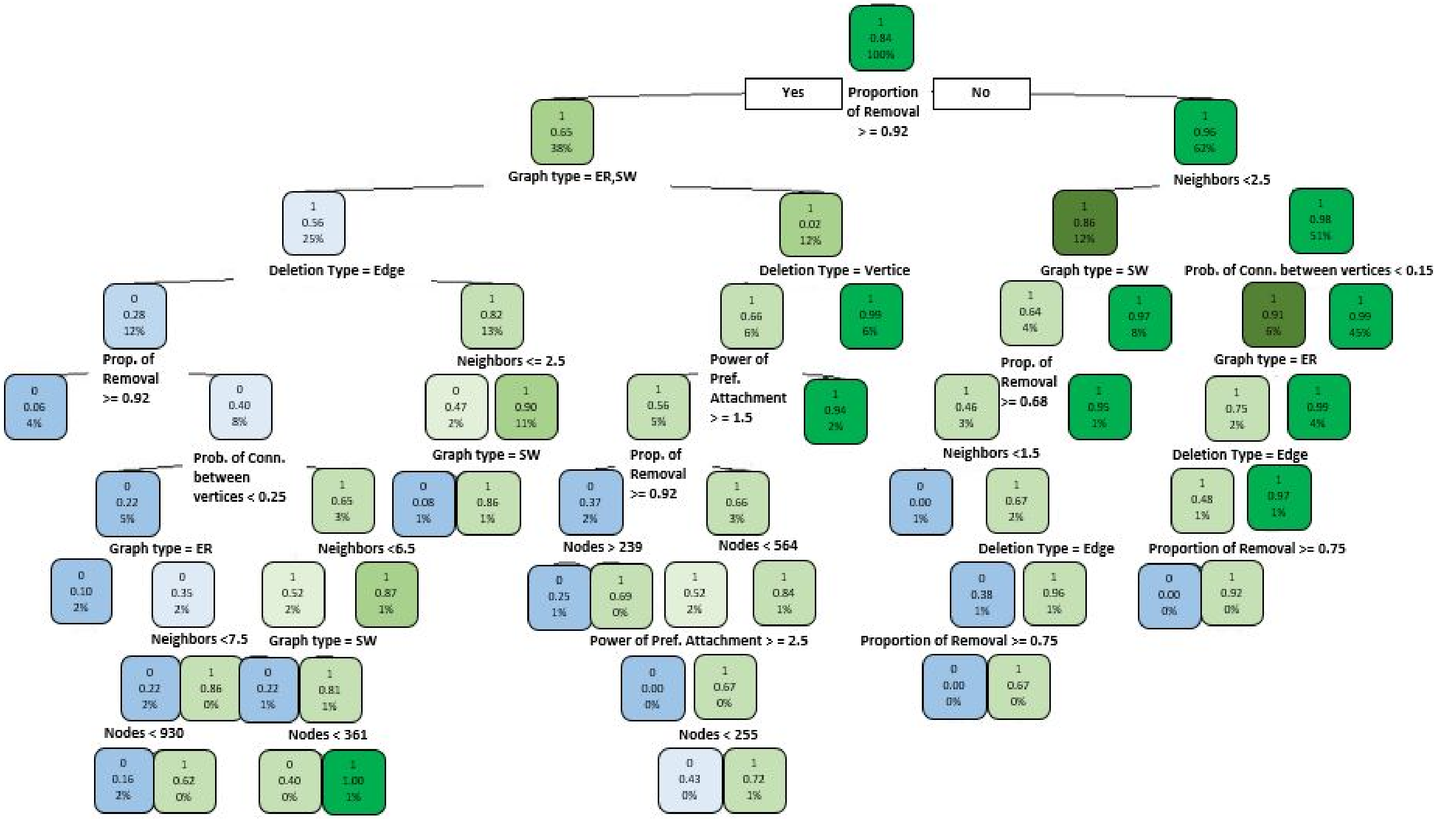
Proof. The number of all possible edges for G to have is $\binom{n}{2}$. We can estimate the probability of being an edge between a node $i \in N_0$ and a node $j \neq i$ is $\hat{p} = |E_i| / \binom{n}{2}$

This converges to the true parameter almost surely by the strong law of large numbers since each edges are independent and identically distributed (*iid).* Therefore, applying Theorem 1 and Proposition 1, we have the result. QED.

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX D. ANALYSIS OF RESULTS FROM PREDICTIVE MODELING

Figure 77. Results from CART model on Network Parameters on Classification of Networks



THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF REFERENCES

Agresti A (2012) *Categorical Data Analysis* 3rd ed. (John Wiley, Hoboken, NJ).

Ahuja RK, Magnanti TL, Orlin JB (1993) *Network Flows: Theory, Algorithms, and Applications* (Prentice Hall, Englewood Cliffs, NJ).

Alderson DL (2008) OR FORUM—Catching the "network science" bug: Insight and opportunity for the operations researcher. *Operations Research* 56(5): 1047–1065, https://doi.org/10.1287/opre.1080.0606.

Arquilla J., Ronfeldt D (2001) Networks and netwars: The future of terror, crime, and militancy. MR-1382, RAND Corporation, Santa Monica, CA, http://www.rand.org/pubs/monograph_reports/MR1382/index.html.

Barabasi A (2015) *Network Science.* http://barabasi.com/networksciencebook/.

Breiman L (2001) Random forests. University of California, Berkeley, https://www.stat.berkeley.edu/~breiman/randomforest2001.pdf.

Breiman L, Friedman J, Stone, Charles J, Olshen R. (1984) *Classification and Regression Trees* (Taylor & Francis, New York).

Carley KM, Kim EJ (2008) Random graph standard network metrics distributions in ORA. Center for the Computational Analysis of Social and Organizational System, CMU-ISR-08-103, Pittsburgh, PA, http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.220.2176&rep=rep1&type=pdf.

Carley KM, Reminga J, Kamneva N (2003) Destabilizing terrorist networks. *NAACSOS Conference Proceedings* (Carnegie Mellon University Pittsburgh, PA), http://www.casos.cs.cmu.edu/publications/papers/Carley-NAACSOS-03.pdf.

Chomsky CL (2005) Viewing September 11 through the lens of history. *Minnesota Law Review* 89, 1437–1463, https://scholarship.law.umn.edu/faculty_articles/11.

Chung F, Lu L (2002) The average distances in random graphs given expected degrees. *Proceedings of the National Academy of Sciences* 99(25): 15879–15882, https://www.ncbi.nlm.nih.gov/pmc/articles/PMC138532/.

Cinar MS, Genc B, Sever H, Raghavan V V. (2017) Analyzing structure of terrorist networks by using graph metrics. *2017 IEEE International Conference on Big Knowledge* (Hefei, China), 9–16.

Cioppa TM, Lucas TW (2007) Efficient nearly orthogonal and space-filling Latin hypercubes. *Technometrics* 49(1): 45–55.

Costa L da F, Rodrigues FA, Travieso G, Boas PRV (2005) Characterization of complex networks: A survey of measurements. *Advances in Physic*s, 56(1):167–242.

Csardi G, Nepusz T (2006) The igraph software package for complex network research. *InterJournal Complex Sy*: 1965.

Ebel H, Davidsen J, Bornholdt S (2003) Dynamics of social networks. *Proceedings of Concepts for Complex Adaptive System*s *2002*. 1–4. https://arxiv.org/pdf/cond-mat/0301260.pdf.

Faloutsos C (2008) Graph mining: Laws, generators and tools. *Lecture Notes in Computer Science* (Springer, Berlin, Heidelberg) vol. 5012: 54.

Friemel TN (2011) Dynamics of social networks. *Procedia—Social and Behavioral Sciences* 22:2–3.

Hellinger E (1909) Neue Begründung der Theorie quadratischer Formen von unendlichvielen Veränderlichen. *J. für die reine und Angew. Math*, 136: 210–271.

Hopkins A (2010) Graph theory, social networks and counter terrorism. Master's thesis, Department of Mathematics, University of Massachusetts Dartmouth, Dartmouth, MA. https://compmath.files.wordpress.com/2010/05/ahopkins_freports10.pdf.

Huddleston SH, Brown GG (2018) Machine learning. *Analytics Body of Knowledge* (INFORMS, Catonsville, MD), 1–41.

Huddleston SH, Kloo I, Greenway R, Haskell M, Usher A (2016) working paper on data science for threat finance intelligence: Center for Army Analysis, U. S. Central Command (CENTCOM)/Defense Intelligence Agency (DIA), http://faculty.nps.edu/shhuddle/.

Kell B (2006), File:Grötzsch graph.svg. Wikimedia Commons. Accessed February 1, 2018, https://commons.wikimedia.org/wiki/File:Gr%C3%B6tzsch_graph.svg.

Kullback S, Leibler RA (1951) On information and sufficiency, *Annals of Mathematical Statistics* 22(1):79–86, https://www.projecteuclid.org/download/pdf_1/euclid.aoms/1177729694.

Noldus R, Mieghem P Van (2014) Assortativity in complex networks. *Journal of Complex Networks* 3(4):507–542, https://www.nas.ewi.tudelft.nl/people/Piet/papers/JCN2015AssortativitySurveyRogier.pdf.

Ressler S (2006) Social network analysis as an approach to combat terrorism: past, present, and future research. *Homeland Security Affairs VII*(2):1–10, https://www.hsaj.org/articles/171.

Rodrigue, J-P, Ducruet C (2017) Graph theory: Definition and properties. *The Geography of Transport Systems*, 4th ed. https://transportgeography.org/?page_id=5976.

Satell G (2013) How the NSA uses social network analysis to map terrorist networks. Digital Tonto. Accessed February 14, 2018, http://www.digitaltonto.com/2013/how-the-nsa-uses-social-network-analysis-to-map-terrorist-networks/.

Sparrow MK (1991) The application of network analysis to criminal intelligence. *Social Networks* 13:251–274, https://sites.hks.harvard.edu/fs/msparrow/documents--in%20use/Application%20of%20Network%20Analysis%20to%20Criminal%20Intelligence--Social%20Networks--1991.pdf.

Vieira H, Sanchez SM, Kienitz KH, Belderrain MCN (2013) Efficient, nearly orthogonal-and-balanced, mixed designs: An effective way to conduct trade-off analyses via simulation. *Journal of Simulation* 7(4):264–275, https://link.springer.com/content/pdf/10.1057%2Fjos.2013.14.pdf.

Watts D, Strogatz S (1998) Collective dynamics of small-world networks, *Nature,* 393:440–442.

Xu J, Chen H (2008) The topology of dark networks. *Communications of the ACM* 51(10):58 – 65, https://www.researchgate.net/publication/220424273_The_Topology_of_Dark_Networks.

Yip M (2008) Social Network Analysis as a tool to study organised cybercrime [Poster]. School of Engineering and Computer Science, University Of Southampton, http://users.ecs.soton.ac.uk/lac/dtc/posters/yip.pdf.

Yoshida R (2018) Average distance between nodes in a random graph provided to the author via personal communication, February 1.

Zhu L, Ng WK, Han S (2011) Database systems for advanced applications. *16th International Conference, DASFAA 2011 International Workshops: GDB, SIM3, FlashDB, SNSMW, DaMEN, DQIS* (Hong Kong, China).

THIS PAGE INTENTIONALLY LEFT BLANK

INITIAL DISTRIBUTION LIST

1. Defense Technical Information Center
   Ft. Belvoir, Virginia

2. Dudley Knox Library
   Naval Postgraduate School
   Monterey, California