Spring 2017

A review of random matrix theory with an application to biological data

Jesse Aaron Marks

Department:

A REVIEW OF RANDOM MATRIX THEORY WITH AN APPLICATION TO BIOLOGICAL DATA

by

JESSE AARON MARKS

A THESIS

Presented to the Graduate Faculty of the

MISSOURI UNIVERSITY OF SCIENCE AND TECHNOLOGY

In Partial Fulfillment of the Requirements for the Degree

MASTER OF SCIENCE

in

APPLIED MATHEMATICS

2017

Approved by

Gayla R. Olbricht, Advisor
Ilene H. Morgan
Yanzhi Zhang

ABSTRACT

Random matrix theory (RMT) is an area of study that has applications in a wide variety of scientific disciplines. The foundation of RMT is based on the analysis of the eigenvalue behavior of matrices. The eigenvalues of a random matrix (a matrix with stochastic entries) will behave differently than the eigenvalues from a matrix with non-random properties. Studying this bifurcation of the eigenvalue behavior provides the means to which system-specific signals can be distinguished from randomness. In particular, RMT provides an algorithmic approach to objectively remove noise from matrices with embedded signals.

Major advances in data acquisition capabilities have changed the way research is conducted in many fields. Biological sciences have been revolutionized with the advent of high-throughput techniques that enable genome-wide measurements and a systems-level approach to biology. These new techniques are very promising, yet they produce a massive influx of data, which presents unique data processing challenges. A major task researchers are confronted with is how to properly filter out inherent noise from the data, while not losing valuable information. Studies have shown that RMT is an effective method to objectively process biological data. In this thesis, the underpinnings of RMT are explained and the function of the RMT algorithm used for data filtering is described. A survey of network analysis tools is also included as a way to provide insight on how to begin a rigorous, mathematical analysis of networks. Furthermore, the results of applying the RMT algorithm to a set of miRNA data extracted from the *Bos taurus* (domestic cow) are provided. The results of applying the RMT algorithm to the data are provided along with an implementation of the resulting network into a network analysis tool. These preliminary results demonstrate the facility of RMT coupled with network analysis tools as a basis for biological discovery.

ACKNOWLEDGMENTS

I would like to take this time to acknowledge those who have helped make this thesis possible. I want to first thank my adviser, Dr. Gayla Olbricht. I am truly grateful for your guidance and flexibility during this endeavor. I am also immensely appreciative of my other two committee members, Drs. Ilene Morgan and Yanzhi Zhang. You two have been a constant source of support and I am much obliged for the guidance you have given me.

I want to express my gratitude for other mentors and advisers I have had throughout my higher education. Firstly, I must recognize the Oak Ridge National Laboratory (ORNL) and the Higher Educational Research Experiences Program for providing me this invaluable research opportunity. Thanks to Dr. Dan Jacobson and my group members at ORNL for your indispensable advice and insight during my time at the lab. Thanks to previous mentors Drs. Annabelle Pratt and Robert Leaf for challenging me with unique research experiences. I am also greatly appreciative of my previous academic advisers, Prof. Barb Thurmon and Dr. Jerry Priddy, for the instrumental role you both had on my education throughout my undergraduate studies at Central Methodist University.

Finally, I want to thank my family and friends for all the love and support you have given me along the way. I hope you know that I am wholeheartedly honored and blessed to be among such wonderful people. Thank you.

TABLE OF CONTENTS

Page

LIST OF ILLUSTRATIONS

1. INTRODUCTION

1.1. BACKGROUND

The digital revolution in recent years has ushered in an unprecedented boom in new technologies. Information is more easily accessible now than ever before via the internet, which is making for a widely-connected, information-rich world. Advancements in science have led to technological feats that would have been unthinkable only decades ago. Biology is one field of science that has seen great advancements in recent times. High-throughput techniques—biological methods that produce simultaneous measurements on the order of thousands across the genome—are revolutionizing our understanding of cells and how their constituents interact [40]. High-throughput techniques use automated equipment that produce data which, when analyzed properly, help scientists tackle systems-level biology questions that would have been unattainable with the use of traditional reductionist approaches. These new experimental methods in biology are producing massive amounts of "omics" data—any of the biological studies that end in the suffix "omics", such as proteomics, genomics, or metabolomics—that can be very difficult to interpret due to the size and complexity of the data. Though the prospects of what can be learned from these influx of data are exciting, there are large obstacles in managing, processing, and extracting meaningful information from these unrefined data. These obstacles arise due to the data being high-dimensional and noisy, which can make uncovering any underlying signals very difficult. With so many variables in the data, there are bound to be spurious correlations just by the nature of big data [25]. In order to process these data and extract meaningful information, appropriate methods that include various statistical tests, bioinformatics methods, and data mining tools can be used. These methods are means to filter out random noise

from the data. Once the data have been filtered, the process of discovering new information and learning new science can begin. A network analysis approach can be used to facilitate this discovery stage.

Using networks to analyze complex data is an emerging approach in biological sciences. To use networks effectively, a critical step to make analysis feasible is the data filtering process (or network pruning). The aim of this process is to reduce the amount of static or irrelevant components in the data. There are various methods for network pruning; however, new methods may be required in order to filter out random noise and non-information. As high-throughput techniques are improved upon, computational complexity increases so that previous data filtering methods become ineffective. One method for network pruning that is explored in this thesis is called random matrix theory (RMT). RMT has been already used in a wide variety of fields because of its objectivity and mathematically sound approach to data filtering. It has been applied in areas such as wireless communication [82], quantum physics [33], number theory [42], and the financial field [48]. The discovery of new and innovative applications of RMT is becoming a common occurrence [9]. The diverse assortment of its applications provides evidence for the efficacy of using RMT as a data filtering technique. It is an unbiased, system-information independent method that can be applied to biological systems. More specifically, this evidence suggests that RMT would be useful for filtering biological data, namely to prune co-expression networks. Also, there is a need for RMT because pruning co-expression networks without ambiguity is difficult and current methods are not robust and consistent with different data sets [52]. Luo et al. [52] showed how RMT could be used to prune co-expression networks and subsequently to facilitate learning about the functions of unknown genes.

1.2. AIMS

This thesis will discuss networks and how they can be used as a tool for a systems-level approach to biological research. A discussion of how networks are visually represented with network diagrams, or graphs will be provided. Various network analysis methods will be discussed, including 1) network reduction methods, which includes a detailed description of RMT, and 2) metrics for measuring the topological properties and features of networks in order to compare different networks to each other. The main focus of this thesis will be to discuss the fundamental principles of RMT and its application as a network pruning tool, a pertinent step in data processing required to extract meaningful information from raw data. RMT was applied to an open-source *Bos taurus* (domestic cow) miRNA data set and the network reduction results are discussed. Lastly, after pruning the networks with the RMT algorithm, the networks were then uploaded into the network analysis tool called Cytoscape [73]. Provided are the results from one of the co-expression networks, so as to showcase how the next stage of scientific discovery could be conducted.

1.3. STRUCTURE

The structure of this thesis is as follows: Section 2 provides a literature review that contains background information on 1) what networks are and why they are an effective tool for conducting systems-biology research, 2) statistical metrics used to create networks from raw data, 3) an introduction to various network pruning methods, 4) an enumeration of various network analysis methods, and 5) a detailed description of the network pruning method RMT. Note that a description of RMT is not included in Section 2.3, where the other pruning methods are described. The discussion of RMT is delayed until the end of Section 2, due to the depth and breadth of the content in which RMT is examined. Section 3 provides the methods and materials used to apply the RMT algorithm to prune the *Bos taurus* miRNA co-expression networks. In Section 4, the results of applying RMT to the co-expression

networks are presented. More specifically, four similarity metrics—Pearson, Spearman, Czekanowski, and SPS—were each applied to the same set of raw miRNA expression data to create four networks. It was at this point that the RMT algorithm was applied to each co-expression network (correlation matrix) in order to remove noise and reduce the size of each network. One of the pruned networks was then uploaded—the co-expression network created with the Pearson similarity metric—into the network analysis tool called Cytoscape [73]. The network was briefly inspected and a cursory demonstration and discussion was provided of how the network could be analyzed further with Cytoscape for the next step in scientific discovery. Finally, a summary of the work and a discussion of possible future work is provided in Section 5.

2. LITERATURE REVIEW

The age of big data has enabled researchers to collect comprehensive sets of data on biological systems, which bestows golden opportunities for systems-level approaches to understanding biology [64]. Trying to understand biology as a whole system, rather than component-by-component analysis, has been a focal point since the emergence of the this big data age. A systems-level approach endeavors to understand something as a unitary whole from a top-down approach. This opposes a bottom-up approach that strives to understand the whole system by piecing together the comprehension of the system's constituent components. Processing information about performance and component interactions as a whole fosters insight into complex systems that were inaccessible before the post-genomic era. One way to study these complex systems is with networks.

Networks have been used to represent a wide variety of systems and have demonstrated their efficacy in areas such as social network analysis [12], the World Wide Web [1], and many domains of biology such as gene co-expression networks [39, 80, 91], protein-protein interaction networks [28, 45, 69], and cell biology [2, 36]. Biological systems are an excellent candidate for the use of networks as an analytical tool because they are composed of many inherent complexities and interacting components. A reductionist approach of studying isolated components is not well-suited for biological systems because of the universal modularity exhibited in these systems [5]. Trying to understand biological systems by studying the individual components can be likened to the parable of the blind men and the elephant. When the men try to describe the elephant as a whole from their individual isolated view points, they miss the mark. Though each description may be accurate, the men are limited by the failure to account for other truths outside of their own account. This describes the limitations of a bottom-up, or reductionist, approach to biology. A systems-level approach, rather than a purely reductionist view, should be used to understand how

the constituent parts interact and affect each other in order to get the big picture of what is happening in the system. Networks are an effective analytical tool for this systems-level approach. They have been widely used in various branches of biology because of the convenience with which the relationships and interactions between the biological entities can be represented [56], especially since the advent of biological big data [15].

2.1. NETWORK OVERVIEW

A network is a way to represent the relationship between objects and how they interact. Any system that is composed of a collection of objects and their interactions can be represented with a network. The data required for a network can be represented as a table in a simple interaction file (SIF). A network that is in SIF format, in its simplest form, will contain three columns. The first column represents object 1, the second column represents object 2, and the last column is the relationship between those two objects. This last column could be a quantification of some association between the objects from columns one and two. Networks in SIF format are practical because they can easily be converted into a visual representation of the network.

A visual representation of a network is called a network diagram or simply a graph. Graphs are essentially composed of a collection of points and lines where the points are the objects and the lines are the associations between the objects. In the jargon of network theory, the points are called nodes, or vertices, and if two nodes are associated with each other a line called an edge connects them [56]. The entire complexity of the network can be captured visually with a graph composed of this collection of nodes and edges [5]. In a biological system, the nodes represent the constituent parts that constitute the system (e.g. genes, miRNA, or proteins) and the edges represent the interactions the biological components have with each other. The collection of these edges and nodes make up

the biological system as a whole. Representing biological networks this way provides a mathematical representation of the system and has applications in ecological, evolutionary, and physiological studies [63].

The purpose of converting a network into a graph is to enable the visualization of the system at hand, which facilitates understanding and can help scientists glean insight that might otherwise be lost. Visualizations are very important and powerful in modern scientific research because they allow humans to take advantage of our biology (i.e. our eyes and the connection with the mind) to see, explore, and process large amounts of information at once [27]. Graphs essentially allow for a better understanding of how objects are associated with each other and the implications of the correlations. Networks are flexible and generic in the sense that many different systems can be represented by them. Biological systems are, therefore, a great candidate for a network analysis approach. A gene co-expression network is a common example of a biological system represented with a network [13, 80, 95]. The nodes represent the different genes in the system and two nodes are connected by an edge if the two genes have a significant correlation. Many other biological systems can be represented with networks, besides just gene co-expression data, because the network architecture of biological systems exhibit characteristics that are generic to systems-level cellular organization [67]. Modeling biological systems with networks has hence become a prevalent method for systems biology research.

Note that the terms network and graph are often used interchangeably. In this thesis, the term network will sometimes be used to refer to the visual representation of the system as well as to reference the table format of the system, such as the SIF formatted file described previously. It will be clear what the term network is referring to from the context. The term graph, however, will strictly mean the visual representation of the system. This section has been a high-level introduction to graph theoretical concepts and definitions that are applicable for representing biological systems. For a more thorough introduction, see [41].

2.2. CREATING NETWORKS: SIMILARITY METRICS

Networks are often constructed to represent a biological system. To construct a network of a biological system, a similarity metric or a distance metric can be applied to a set of experimental data. The result is a matrix of values that quantifies the association between the constituents of the biological system. To reiterate, the constituents of the biological system could be various entities, such as different genes from which a gene co-expression network could be constructed. In this thesis, the components of the biological data are different miRNA sequences.

Typically, similarity metrics—also known as correlation metrics—are applied to two or more objects (e.g. two different miRNA sequences). The similarity metric will quantify an association the objects have with each other. This quantification could be a variety of measurements, such as how often the objects are involved in a similar process, how likely the objects are to appear in the same location, etc. The value representing the quantified correlation is often referred to as the similarity coefficient, or the correlation coefficient. This correlation coefficient is a real-valued number that describes to what extent the objects are related.

Similarity metrics are ubiquitous in statistics and related fields and many different similarity metrics have been developed as well [14]. The plethora of similarity metrics is not a superfluity because each metric is important in its own right. Different metrics were developed to measure different aspects of correlation. Each metric measures different aspects of similarity and quantifies those associations in different manners. Using the appropriate measure for a specific need is of fundamental importance in pattern classification, clustering, and retrieval problems [23]. The similarity metric chosen will highlight specific features of the data. Different similarity metrics applied to the data will ultimately result in different biological interpretations of the data. Therefore, a similarity metric must be chosen based on the biological significance that is trying to be captured.

This thesis will cover four similarity metrics used to create different networks: Pearson product moment correlation, Spearman's rank-order correlation, Czekanowski index, and the stringent proportional similarity index (SPS).

2.2.1. Pearson Product-Moment Correlation. The first rigorous mathematical derivation of the Pearson product-moment correlation coefficient, or Pearson's r, was developed by Karl Pearson in 1895 [57]. Francis Galton is credited for first introducing the notion of regression in 1886 upon which the Pearson's r was built [30]. Pearson's r was the primary measurement of correlation in regression analysis and is still very widely used today. It is a dimensionless index that is calculated with the formula:

$$r = \frac{\sum_{i=1}^{n} \left(X_i - \bar{X}\right)\left(Y_i - \bar{Y}\right)}{\sqrt{\sum_{i=1}^{n} \left(X_i - \bar{X}\right)^2 \sum_{i=1}^{n} \left(Y_i - \bar{Y}\right)^2}} \tag{2.1}$$

where $\bar{X}$ and $\bar{Y}$ are the expected values of the vectors $\mathbf{X} = \{x_1, x_2, x_3, \ldots, x_n\}$ and $\mathbf{Y} = \{y_1, y_2, y_3, \ldots, y_n\}$, respectively. The Pearson's r can be interpreted as the strength of the linear relationship between two variables [50]. Using the Cauchy-Schwarz inequality, it can be shown that the absolute value of the numerator is less than or equal to the denominator and therefore $-1 \leq r \leq 1$. The closer to 1 or -1 the r value is, the stronger the positive or negative linear relationship is between the two vectors, respectively. A Pearson's r value near 0 indicates a lack of linear relatedness; this does not, however, indicate that the two vectors are independent.

2.2.2. Spearman's Rank-Order Correlation. Spearman's rank-order correlation was developed by Charles Spearman in 1904 [78]. The correlation coefficient is often denoted by ρ, which gives way to its reference of Spearman's ρ. Spearman's ρ is a nonparametric version of the Pearson correlation, meaning it does not necessarily matter what the values of the individual entries are, it matters only what the values are relative to the other values. Because only a variable's rank matters, each rank is given equal weight. This

makes Spearman's correlation less sensitive to strong outliers that would more heavily affect Pearson's correlation. For a definition of Spearman's ρ, let $\mathbf{X} = \{x_1, x_2, x_3, \ldots, x_n\}$ and $\mathbf{Y} = \{y_1, y_2, y_3, \ldots, y_n\}$ represent two vectors of length n. Also, let $\mathbf{R} = \{R_1, R_2, R_3, \ldots, R_n\}$ and $\mathbf{Q} = \{Q_1, Q_2, Q_3, \ldots, Q_n\}$ represent the sequence of ranks for the vectors $\mathbf{X}$ and $\mathbf{Y}$, respectively. Spearman's ρ is calculated with the formula:

$$\rho = \frac{\sum_{i=1}^{n} \left(R_i - \bar{R}\right)\left(Q_i - \bar{Q}\right)}{\sqrt{\sum_{i=1}^{n} \left(R_i - \bar{R}\right)^2 \sum_{i=1}^{n} \left(Q_i - \bar{Q}\right)^2}} \tag{2.2}$$

where $\bar{R}$ and $\bar{Q}$ are the mean ranks of the sequence of values for the variables $\mathbf{R}$ and $\mathbf{Q}$, respectively [61]. From this definition, it is easy to see the similarity between the Spearman correlation and the Pearson correlation; Spearman's correlation is the same formula as given by Equation (2.1), where $X_i = R_i$ and $Y_i = Q_i$, indicating the formula is applied to the ranks rather than the raw data as it is in Pearson's correlation. Spearman's correlation is also similar to the Pearson correlation in that it is on the scale $-1 \leq \rho \leq 1$. A Spearman's ρ value close to -1 represents a strong negative correlation in the monotonicity, a Spearman's ρ value near 0 indicates a very weak correlation in monotonicity, and a Spearman's ρ value near 1 indicates a strong positive correlation in monotonicity of the two vectors. The two correlation tests differ in that the Pearson correlation is quantifying the strength of the linear relationship between two variables (vectors) whereas the Spearman correlation is simply measuring the strength of the monotonic relationship between the variables. Consequently, the variables do not necessarily need to have a linear relationship to exhibit a correlation [61].

2.2.3. Czekanowski Index. The Czekanowski index was described by Jan Czekanowski as early as 1909 [18] and more in-depth in 1913 [19] to quantify the likeness between two biological samples. The Czekanowksi index is a quantitative version of a presence-absence similarity index called the Sørensen index [76]. The Sørensen index, which is also

known as Czekanowski's binary [29], is widely used to calculate ecological community measurements [94]. The Czekanowski index is also known as the proportional similarity index [7]. This metric is used to quantify the amount of set intersection two or more vectors may have with each other. The Czekanowski index, therefore, takes on values in the range $[0, 1]$. An index value between two samples that is near 0 indicates that the two samples had very little overlap. An index value near 1 indicates that the two samples had substantial overlap. For two vectors $\mathbf{X}$ and $\mathbf{Y}$, the Czekanowski index is defined as:

$$C_z = \frac{\sum_{i=1}^{n} 2\min(X_i, Y_i)}{\sum_{i=1}^{n} (X_i + Y_i)} \tag{2.3}$$

2.2.4. Stringent Proportional Similarity Index. The stringent proportional similarity index (SPS) was developed by D. Weighill and D. Jacobson [85]. SPS is a modified version of the Czekanowski index created with the intention of being an index that quantifies vector overlap similar to how the Czekanowski index does. SPS is more strict than the Czekanowski index though, in the sense that vectors have to be more similar to each other in order to achieve the same score as that of a Czekanowski index. For two vectors $\mathbf{X}$ and $\mathbf{Y}$, Weighill defines the SPS index as:

$$1 - \frac{1}{n} \sum_{i=1}^{n} \frac{\left|X_i^2 - Y_i^2\right|}{X_i^2 + Y_i^2}. \tag{2.4}$$

It is apparent that if in the ith position, both vectors $\mathbf{X}$ and $\mathbf{Y}$ have a value of 0, then the expression is undefined due to a division by zero error. To resolve this issue, a slightly modified version of the SPS index was created. The modified SPS is defined as:

$$1 - \frac{1}{n}\left[\sum_{\{i:X_i^2+Y_i^2\neq 0\}}^{n} \frac{\left|X_i^2 - Y_i^2\right|}{X_i^2 + Y_i^2} \quad + \sum_{\{i:X_i^2+Y_i^2=0\}}^{n} 1\right]. \tag{2.5}$$

If there is a value of 0 in the ith position for both vectors $\mathbf{X}$ and $\mathbf{Y}$, then the expression is calculated slightly different than that of Equation 2.4 in order to remedy the divistion by zero error. The new term included in Equation 2.5 can be considered as a penalty term for the calculation of the SPS index. If two vectors are both very sparse (i.e. the majority of the values are 0), then these two vectors should not be considered to be similar to each other.

The SPS index and also the Czekanowski index are metrics for quantifying how similar two vectors are, or the amount of biological overlap exhibited. With this in mind, that is why the penalty term was added so that two sparse vectors will not attain a high SPS index value. The scale for the SPS index is the same as for the Czekanowski index, namely values in the range $[0, 1]$ because it also measures the amount of overlap between two vectors. Therefore, the values can be interpreted in a similar fashion, namely an SPS index value near 1 indicates that the two vectors of data were very similar and on the other hand, if two vectors were nearly disjoint, then the SPS index will have a value near 0. Note that when the SPS index is referred to throughout the rest of this thesis, it will be referring to the modified version in Equation 2.5

2.3. NETWORK PRUNING

Network pruning is the process of removing unwanted or irrelevant parts of a network before one begins to analyze it. Pruning is a necessary step in data processing in order for the emergence of significant associations in the data to become observable.

Constructing a network by using a similarity metric, such as the ones defined in the previous section, quantifies the association between objects. The resulting network from applying a particular similarity metric is a complete network of the data which includes an all-versus-all comparison between the different elements of the data. The resulting network could contain significant amounts of noise, or irrelevant data, that mask the true associations and correlations that some elements may have with each other. One of the major challenges in network construction is to produce a network of the biological data which captures the complex interactions without too much computational complexity overhead [38]. Pruning endeavors to filter out this noise so that the system specific signals can emerge. A selection of some widely used pruning methods will be covered in this section. However, the random matrix theory (RMT) pruning method has a dedicated portion by itself in the last section of this literature review, since it is the focus of this thesis.

2.3.1. Thresholding Methods. A threshold is a base value against which one can compare all of the values in the network that were obtained by applying a particular similarity metric to the experimental data. This threshold is the value that demarcates the irrelevant noisy data from the system specific signals. The entries that have a correlation value whose magnitude is less than the threshold value will be regarded as less significant.

Thresholding has been suggested by several researchers as a way to create robust networks [10, 13, 20, 95]. Zhang et al. [95] proposes two methods of thresholding, namely hard thresholding and soft thresholding. Hard thresholding completely eliminates the entries whose (absolute) value is less than the specified threshold. This hard thresholding method is essentially a sign function, or signum function, which assigns a "sign", a 1 or a 0, to the entries in the similarity (correlation) matrix based on their value compared to a predefined threshold value τ. This is defined mathematically by:

$$a_{ij} = \operatorname{sgn}(s_{ij}, \tau) := \begin{cases} 1, & \text{if } |s_{ij}| \geq \tau \\ 0, & \text{otherwise} \end{cases} \tag{2.6}$$

where s_{ij} is the correlation value between the ith and jth nodes in the similarity matrix, τ is the predefined hard threshold value, and a_{ij} is the resulting connection or disconnection between the ith and jth nodes. All of the edges in which a 0 was assigned are then deleted from the network. This removes the edges that signify a weak correlation and retains a network with only the most prominent correlations. It reduces the network and therefore some of the computational expense. Though there are strategies for picking hard thresholds based on statistical significance [95], information can still be lost with a hard thresholding method [13, 95]. An example of this is if $\tau = 0.90$ then any correlation coefficient c whose value is below τ will be eliminated from the network: if $c = 0.899$, the edge will still be removed. Note that this hard thresholding method is for creating unweighted graphs [81].

Another thresholding method proposed by Zhang et al. [95] is called soft thresholding. The soft thresholding method differs from hard thresholding in that if a soft thresholding method is used, none of the edges will be removed from the network. This is an approach when weighted networks [81] are used for analysis. Soft thresholding essentially applies a function to the correlation values in the similarity matrix that increases the relative weight of strong correlation values. Similarly, the function decreases the relative weight of weaker correlation values. An example of such a soft thresholding function is [95]:

$$a_{ij} = f(s_{ij}, \beta) := |s_{ij}|^{\beta} \tag{2.7}$$

where s_{ij} is the correlation value between the ith and jth nodes, β is a parameter which affects the relative weight assignment, and a_{ij} is the resulting edge weight assigned by applying the function f. This has the benefit of not losing any information from the data; however, it does not decrease the computational expense, which becomes significant when there is an increasing amount of data being produced. Because the soft thresholding method neglects to remove any potentially irrelevant edges, the network can be plagued with spurious node connections, which can mask the true signal in the network.

2.3.2. Maximum Spanning Tree. The maximum spanning tree algorithm is a way to construct a spanning tree with maximum weight from a weighted graph. Essentially, it is a method to prune a network so that only the most significant portions are maintained. Maximum spanning trees are the result of applying a sister algorithm, minimum spanning trees [60], to a set of data whose entries are inverted [72]. More specifically, each entry in a network a_{ij} is replaced with the value $a'_{ij} = 1 - |a_{ij}|$ and then subsequently, a minimum spanning tree algorithm is applied to the network.

2.4. METHODS OF NETWORK ANALYSIS

Once a network has been created and properly filtered to remove any spurious correlations, one can begin investigating the patterns that emerge. The patterns in the network interactions are often overlooked, but almost invariably crucial to the behavior of the system [56]. Kitano and Hiroaki [47] describe the importance of network analysis well with an analogy to examining traffic patterns; creating a diagram or a graph is an important first step, which is like a static road map, but what is really important is to know the traffic patterns, why the patterns emerge, and how to control them. Understanding the patterns and structures of a system contribute to an understanding of how the system works as a whole.

Networks are now a very useful tool for analyzing these complex systems as a whole because of modern computational power which is enabling data acquisitions that allows for a better analysis of network topologies. The topology of an object is concerned with its geometric properties in space, as well as its logical arrangement. Various network topology metrics have been developed to analyze and quantify different properties of the networks [39]. These metrics analyze the layout of the graph, the geometric properties of the nodes, and the node-based interactions and connectivity. A network's topology can be assessed on the microscale (local) and on the macroscale (global). Local properties of a network involve

properties at the individual node-based level, whereas the global network properties assess the more general properties of the network as a whole. Quantifying aspects of a network allows for the ability to compare the properties of one network with another.

Many different networks can be created from the same set of data. The various combinations of procedures to produce the network will generate the possibility for the exhibition of many different topological characteristics. The choice of a similarity metric to calculate the correlations, in conjunction with the chosen method of data filtering, can have a significant effect on the network topology. Comparing and contrasting topological features of two networks could show prevailing characteristics that would be of interest to explore. Also, there may be discrepancies that warrant further investigation and could lead to new understandings of the system. This is why having metrics to use as a tool for network comparison is important. The following subsections will briefly list and discuss some of the metrics and methods used to analyse network topologies. These metrics quantify network characteristics at the local-level and the global-level, which then allows for the ability to compare and contrast biological networks.

2.4.1. Local Network Topology Measures. These are called node-based topology measures. The cornerstone of these local measures is in the properties of the individual nodes and or node-pair associations. Weighill [85] provides a thorough discussion of eight local network topology measures and therefore a simple enumeration of those measures that were detailed will be provided. The eight different measures included in [85] are: adjacency [95], connectivity [39], maximum adjacency ratio [39], topological overlap [67, 95], TOM-based connectivity [95], clustering coefficient [84], betweenness [68], and efficiency [49]. To reiterate, these local measures listed will provide a way to quantify the properties of a network on the node-based level. These quantifications provide the grounds on which one could begin comparing two different networks.

2.4.2. Global Network Topology Measures. These metrics quantify network characteristics that may be more visually apparent when the network is examined with a network visualization tool. Weighill [85] provides a diligent report of four main global network topology measures that are enumerated here. These measures include: density [75], centralization [22], heterogeneity [22], path length [84], and degree correlation [68]. These global measures take a high-level view of the network as opposed to the local node/node-pair level view. Global characterizations of networks can capture emerging patterns that a granular local metric may not capture.

2.5. RANDOM MATRIX THEORY

2.5.1. Section Overview. This section will include a detailed description of random matrix theory (RMT). RMT is a network pruning method whose discussion was delayed until now because a much more in-depth description than the other pruning methods is provided; as it is the focus of this thesis. Included is a description of the origins of study, the mathematical theory and foundation relevant to this thesis, and also definitions and descriptions of relevant statistical distributions necessary for understanding the application of RMT. The section concludes with a discussion of various candidate software tools that were investigated for applying the RMT algorithm.

2.5.2. Introduction to Random Matrix Theory. One focus area in mathematical physics involves studying the spacings of ordered entities in a system [55]. Suppose there is a system with the observables $\{\epsilon_1, \epsilon_2, \epsilon_3, \ldots, \epsilon_n\}$, where $\epsilon_1 \leq \epsilon_2 \leq \epsilon_3 \leq \ldots \leq \epsilon_n$, which describe some property of that system. The spacings between the distinct ϵ_i could be a central question to explore. Studying spacings has long been an active area of research, which includes examining the spacings between different energy levels of complex nuclei [86], prime gaps [77], and the ordered arrangement of the eigenvalues of symmetric matrices [24]. Having a full, comprehensive body of knowledge about the system itself would lend

to an exact knowledge of the spacing between each ϵ_i and ϵ_{i+1}. When studying complex systems whose interrelationships quickly become too thorny to grasp, the converse approach is often taken; namely trying to gain more of an understanding about the underlying system from the knowledge of how the ϵ_i are spaced.

RMT is an approach for studying massive, complex systems that are too difficult to directly analyze because of their labyrinth of connected components. The main functionality of RMT is to contribute to an understanding of complex systems primarily by analyzing the statistical properties of eigenvalue spacing distributions of the systems. By examining the distribution of the eigenvalue spacings of a network and comparing that distribution to known global properties associated with some universal system, one can detect system specific behavior and determine what is unique about the network that is being analyzed. The behavior just described is called universality, a key concept to the mechanics of RMT. Universality is the observation that there are behaviors exhibited in a large collection of systems that share some similar characteristics, which are independent of the individual systems. The universality property that RMT utilizes is the known eigenvalue spacing characteristics exhibited by established ensembles of random matrices. Before delving deeper into the mechanics of RMT and discussing more about these random matrix ensembles, the history of RMT should first be discussed along with the emergence of its multifaceted applications in various domains.

The study of RMT has is roots in the nascent research of random matrices by John Wishart in 1928 [89]. Wishart was analyzing the statistical properties of the eigenvalues of random matrices. This ground work helped build the foundation for modern research and applications of RMT. It has since matured and developed within the realm of mathematics and physics into a tool with many useful applications. RMT was brought to the forefront of physics and mathematical research by Eugene Wigner and Freeman Dyson in the 1960s [88] for studying high-dimensional complex systems, namely atomic nuclei.

Wigner had labored to study the energy levels of the uranium nucleus in the 1950s [86] and was finding that it was far too difficult to understand how all the constituent components interacted; it is known that matrices can be used to represent the energy levels of an atom. According to Dyson, "Every quantum system is governed by a matrix representing the total energy of the system, and the eigenvalues of the matrix are the energy levels of the quantum system" [90]. Simple atoms like hydrogen and helium can easily be calculated to astounding precision; however, when dealing with heavy nuclei such as uranium and plutonium, the calculations become infeasible. There are simply too many interconnected constituent parts of a uranium nucleus to handle.

Wigner's goal was to describe the general properties of the energy levels of atomic nuclei, like uranium, with a Hamiltonian $\mathbf{H}$ [44]. A Hamiltonian is a Hermitian operator that corresponds to the total energy in a system, also known as the total energy operator [17]. The eigenvalues of the Hamiltonian would correspond to the energy levels of the physical system. But again, the nuclei of heavy atoms are too difficult to directly be represented with a matrix $\mathbf{H}$. Wigner then postulated that $\mathbf{H}$ could be regarded as a random matrix that is a member of a large group or ensemble of Hamiltonians which would all have some of the same universal properties [87]. This leads into the discussion of random matrix ensembles.

2.5.3. Wigner Matrices. The most basic model for a random matrix ensemble is the Wigner matrix ensemble. For the sake of clarity, an ensemble of random matrices is a family, group, or collection of random matrices where any member of the infinite group of matrices can represent a state of the entire group. Wigner matrices are historically important because they were the first model of a random matrix ensemble.

Now to define a Wigner matrix (ensemble): specifically a real Wigner matrix which is a Wigner matrix with real-number entries. Without loss of generality, when referring to a real Wigner matrix it will be referred to simply as a Wigner matrix because neither complex nor quaternionic Wigner matrices will be considered in this thesis. Consider a symmetric

$N \times N$ matrix $\mathbf{A}$:

$$\mathbf{A} = \begin{pmatrix} a_{1,1} & a_{1,2} & \cdots & a_{1,N} \\ a_{2,1} & a_{2,2} & \cdots & a_{2,N} \\ \vdots & \vdots & \ddots & \vdots \\ a_{N,1} & a_{N,2} & \cdots & a_{N,N} \end{pmatrix} = \begin{pmatrix} a_{1,1} & a_{2,1} & \cdots & a_{N,1} \\ a_{1,2} & a_{2,2} & \cdots & a_{N,2} \\ \vdots & \vdots & \ddots & \vdots \\ a_{1,N} & a_{2,N} & \cdots & a_{N,N} \end{pmatrix} = \mathbf{A}^T \tag{2.8}$$

where $\mathbf{A}^T$ denotes the transpose of $\mathbf{A}$. Let the matrix elements $\{a_{ij} : i < j\}$ (off-diagonal) and $\{a_{ii} : i\}$ (diagonal) be two independent and identically distributed (i.i.d.) random variables. Suppose the matrix elements a_{ij} have mean zero and unit variance: $\mathbb{E}(a_{ij}) = 0$ and $\mathbb{E}(a_{ij}^2) = 1$. Further suppose the matrix elements a_{ii} also have mean zero and unit variance: $\mathbb{E}(a_{ii}) = 0$ and $\mathbb{E}(a_{ii}^2) = 1$. This matrix $\mathbf{A}$ is then a representative of the Wigner matrices. Note that the random variables a_{ii} and a_{ij} are not necessarily from the same distribution.

2.5.4. Wigner Semi-Circle Law. Now, with a Wigner matrix defined, the staple of RMT can be introduced: the Wigner semi-circle law. This can be seen as the inception to the study of RMT. The Wigner semi-circle law was first derived by Eugene Wigner in 1955 when he observed that the empirical distribution of the eigenvalues of a Wigner matrix closely followed the distribution of a semi-circle [87]. The standard semi-circle distribution of a random variable X with a continuous distribution on $[-1, 1]$ is given by the probability density function f defined as:

$$f(x) = \frac{2}{\pi}\sqrt{1 - x^2}, \quad x \in [-1, 1]. \tag{2.9}$$

The semi-circle law is concerned with the asymptotic behavior exhibited by the normalized eigenvalues

$$\lambda_1\left(\frac{\mathbf{A}}{\sqrt{N}}\right) \leq \lambda_2\left(\frac{\mathbf{A}}{\sqrt{N}}\right) \leq \lambda_3\left(\frac{\mathbf{A}}{\sqrt{N}}\right) \leq \cdots \leq \lambda_n\left(\frac{\mathbf{A}}{\sqrt{N}}\right) \tag{2.10}$$

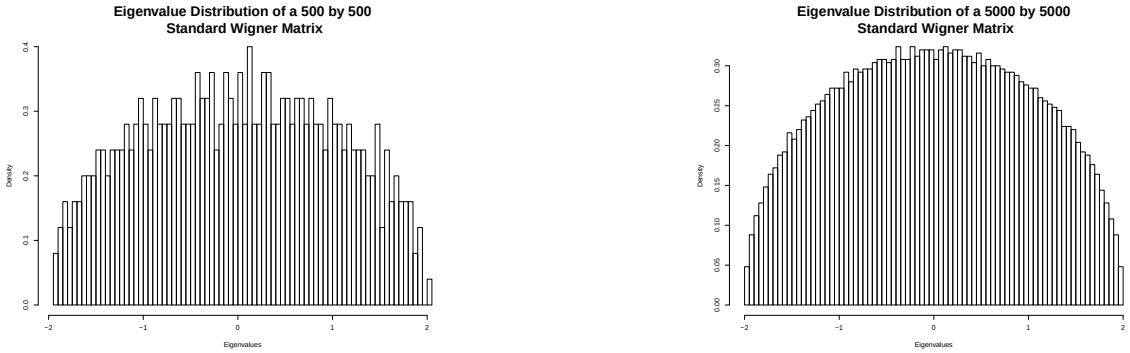
of an $N \times N$ Wigner matrix $\mathbf{A}$ in the limit as $N \to \infty$. It should also be noted that the eigenvalues of a real symmetric matrix are real [65], and there are exactly N eigenvalues (not necessarily distinct) due to the fundamental theorem of algebra.

Wigner observed that the histogram of the eigenvalue density approached the deterministic curve of the semi-circle or half-circle distribution. This in fact turned out to be universally true, which the Wigner semi-circle law describes. This law implies that there is a universal probability density distribution, σ_v, for which the density of eigenvalues of any Wigner matrix, with a second moment v, will converge to σ_v [46]. This limiting distribution is given by:

$$\sigma_v(x) = \frac{1}{2v\pi}\sqrt{(4v - x^2)_+} \tag{2.11}$$

where $(x)_+ = x$ if $x > 0$ and $(x)_+ = 0$ otherwise. This law turns out to be critically important in RMT. It, in effect, advances that the eigenvalues of certain ensembles of random matrices will behave in a predictable fashion. This is very useful because if one knows how the eigenvalues of a matrix should behave, then when the distribution of the eigenvalues behave differently, it indicates that there has been some kind of deviation from the norm. The point at which the deviation occurs is when there has been an underlying change in the data. This idea is the basis of the application of RMT, which will be discussed further in this section.

To illustrate the Wigner semi-circle law and how it is valid for $N \times N$ Wigner matrices as $N \to \infty$, two plots were produced with the statistical programming language R [66]. The R code that was written to produce these plots is provided in Appendix A. Figure 2.1a is a 500×500 normalized Wigner matrix and Figure 2.1b is a 5000×5000 normalized Wigner matrix. As the dimensions of the symmetric Wigner matrix increase, the eigenvalue distribution gets closer to the semi-circle distribution. These matrices were constructed with entries a_{ii} and a_{ij} from a normal distribution: $a_{ii} \sim \mathcal{N}(0, 1)$ and $a_{ij} \sim \mathcal{N}(0, 1)$. Wigner matrices with entries from the normal distribution are equivalently called Gaussian Wigner matrices. This leads into the discussion of the Gaussian orthogonal ensemble (GOE).



(a) (b)

Figure 2.1. Plots (a) and (b) illustrate the convergence of the eigenvalue density distribution, as $N \to \infty$, to the limiting semi-circle distribution as described by the Wigner semi-circle law. Plot (a) is the distribution for a 500×500 normalized Wigner matrix and similarly, Plot (b) is for a 5000×5000 normalized Wigner matrix.

2.5.5. Gaussian Orthogonal Ensemble. Gaussian ensembles, namely Gaussian orthogonal ensemble (GOE), Gaussian unitary ensemble (GUE), and Gaussian symplectic ensemble (GSE), are some of the most typical and frequently studied ensembles of random matrices. This is likely because these ensembles are the least computationally expensive ensembles of random matrices [8]. Dyson introduced these three matrix ensembles with time reversal as a motivation [44]. In this thesis, the application of RMT on biological data required the use of the statistical properties of the GOE.

Now for a definition of this special Wigner matrix ensemble: an $N \times N$ Wigner matrix $\mathbf{A}$, whose entries are $a_{ij} \sim \mathcal{N}(0,1)$ and $a_{ii} \sim \sqrt{2}\mathcal{N}(0,1)$, is considered a member of the GOE. One of the main features of the GOE is the concept of level repulsion that Wigner described for energy level of quantum systems. The eigenvalues from a GOE matrix have the tendency to repel each other, or in other words, they will likely have distinct and isolated values. Wigner posited this idea of level repulsion while studying the nuclei of complex heavy atoms. He realized that the energy levels (eigenvalues) should repel each other and so he began investigating the distribution of the spacings between the different levels. This lead to the idea of the nearest neighbor spatial distribution (NNSD) of the

eigenvalues. Before the NNSD is described, a preliminary discussion of Poisson statistics and the negative exponential distribution needs to be established along with a description of the χ^2 goodness-of-fit test.

2.5.6. Statistical Distributions. In this subsection, a description of the statistical procedures involved in the RMT algorithm is provided. Specifically, a description of the negative exponential distribution, the χ^2 goodness-of-fit test, and the nearest neighbor spatial distribution (NNSD) is provided.

2.5.6.1. Poisson statistics and the negative exponential distribution. The Poisson distribution is a well known and widely used discrete probability distribution. Despite its name, it is not the *fish* distribution. The Poisson distribution was developed in 1837 by the French mathematician Siméon Denis Poisson [62]. It is a discrete probability distribution, meaning that it gives the probability of an event X occurring at some specific value x, where x can take on a finite or a countably infinite number of values. The sum of the probabilities of each event x occurring over all possible values of X adds to 1. Therefore, X is a discrete random variable defined as:

$$\sum_{x=0}^{\infty} \Pr(X = x) = 1 \tag{2.12}$$

where x are all of the possible values of the variable X. A Poisson random variable X is defined as the number of events that occur in some interval of space or time. If a random variable X is described by a Poisson distribution, it is said to follow a Poisson process, namely 1) the events must occur independently, i.e. the occurrence of one event does not diminish nor increase the probability of occurrence of another event, 2) the rate at which the events occur is constant, 3) the number of times an event can occur is non-negative, and also 4) the mean is equal to the variance: $\mathrm{E}(X) = \mathrm{Var}(X)$ [34]. The probability mass function of the Poisson distribution is given by:

$$f_X(x) = \Pr(X = x) = \frac{\lambda^x e^{-\lambda}}{x!} \tag{2.13}$$

where x, the number of occurrences per interval, takes values $0, 1, 2, \ldots$ and the mean number of occurrences per interval is given by λ.

The Poisson distribution is closely related to the negative exponential distribution. Whereas the Poisson distribution might be used to model the probability of occurrences of random events, the negative exponential distribution would be used to model the how much time elapses between those random events. Both of these distributions relate to the same Poisson process, they just govern different aspects of it. The negative exponential distribution is defined by:

$$\mathrm{P}(s) = e^{-s}. \tag{2.14}$$

In the application of the negative exponential distribution to eigenvalue occurrences, s would denote the spacing between successive eigenvalues, where s can take on values $[0, \infty)$.

2.5.6.2. The χ^2 goodness-of-fit test. Here a description of the χ^2 goodness-of-fit test, or just χ^2 test, is provided. It is a critical component in the RMT algorithm and the statistical procedure that the RMTGeneNet software implements: a software package used in this thesis. The χ^2 test is the oldest and perhaps best-known goodness-of-fit test that was first investigated by Pearson in 1900 [58]. It provides a quantitative statistical basis to see if some hypothesized distribution "fits" the distribution of some observed data. A χ^2 test can be used to analyze how "close" the observed values from an experiment are from those which one would expect if the values were from a random variable that followed a particular distribution.

The null hypothesis, H_0, is that the observed data follow a particular distribution. The outcome of the χ^2 test provides evidence to either support the assertion of the H_0 or to not support it. If the results fail to reject the H_0, then there is strong evidence to suggest that the observed data does in fact come from the same random variable as the one that follows the particular hypothesized distribution. On the other hand, if the results do not support the H_0, then the H_0 is rejected for an alternative hypothesis, H_1. This H_1 states that the observed data is not from a random variable that follows the particular

hypothesized distribution. These hypotheses should be stated in a way such that they are mutually exclusive: if one hypothesis is true, then the other hypothesis is effectively not true, and vice versa.

Once the H_0 and H_1 have been established, the next step is to partition the observed data into n intervals, or bins, of finite length. The χ^2 test statistic is then defined as:

$$\chi^2 = \sum_{i}^{n} \frac{(O_i - \mu_i)^2}{\mu_i} \tag{2.15}$$

where O_i is the observed frequency for bin i and μ_i is the expected frequency for bin i; see Conover [16] for an explanation of how to calculate the expected frequency μ_i. The exact χ^2 test statistic is approximated by the χ^2 distribution with $n-1$ degrees of freedom. Therefore, one can see that the χ^2 distribution varies depending on the number of degrees of freedom, or the number of bins used to partition the observed data.

An analysis plan is formulated by choosing a statistical relationship parameter α. Recall that a statistically significant relationship means that the results from an experiment are not likely to occur by pure chance alone, instead there is likely a specific cause for what is being seen. A significance level α is established, which is the maximum probability of rejecting a true null hypothesis one is willing to accept [16]. The significance level α is used to determine if the observed sample frequencies significantly differ from what was expected in the H_0. If the χ^2 test statistic value is greater than the $1-\alpha$ quantile from the χ^2 distribution with $n-1$ degrees of freedom, then the H_0 would be rejected. The p-value associated with the χ^2 test statistic is the probability that a χ^2 random variable with $n-1$ degrees of freedom would be greater than the χ^2 test statistic value if the H_0 were true. This can be found with a χ^2 table. These tables can be used to complete a statistical analysis in order to discern the χ^2 values. A χ^2 table is included in Appendix C.

For the RMTGeneNet software used in this thesis, the H_0 is that the nearest neighbor spatial distribution (NNSD)—discussed in the next subsection— follows the negative exponential distribution. The H_1 is that the NNSD does not follow the negative exponential distribution. When the default p-value of ~ 0.001 is obtained (χ^2=100 with 59 degrees of freedom), the H_0 is rejected; i.e. the distribution does not follow the negative exponential distribution [32].

2.5.6.3. Nearest neighbor spatial distribution. The discussion of the statistics involved in RMT can continue with a description of the nearest neighbor spatial distribution (NNSD). The NNSD looks at the difference between successive eigenvalues from an ordered sequence of eigenvalues and quantifies the distribution of the spacings. This is a convenient way to compare eigenvalue distributions. Consider the ordered sequence of eigenvalues $\{\lambda_1, \lambda_2, \lambda_3, \ldots, \lambda_n\}$, where $\lambda_1 \leq \lambda_2 \leq \lambda_3 \leq \cdots \leq \lambda_n$. Then, let the sequence $\{x_1, x_2, x_3, \ldots, x_n\}$ represent the normalized spacings between successive eigenvalues:

$$x_i = \frac{x_{i+1} - x_i}{S}, \tag{2.16}$$

where S is the average spacing of the sequence. The NNSD is then the probability density function $P(x)$, which gives the probability of finding the next eigenvalue λ_{i+1} a distance of x away from a known eigenvalue of λ_i.

There are two universal ensembles of matrices that RMT distinguishes for, 1) the GOE, which are a class of random matrices and 2) the Poisson ensemble, which are matrices whose eigenvalue behaviors follow statistics described by the Poisson process. Observing the NNSD of these two distributions allows for a characterization of the matrix as a member of the GOE or as a member of the Poisson ensemble. This thesis will denote $P_{GOE}(x)$ and $P_{Poisson}(x)$ as the NNSD of the GOE and Poisson ensemble, respectively, where x is the random variable that denotes the eigenvalue spacing. As it was stated earlier, the eigenvalues of matrices that are a member of the GOE exhibit level repulsion; the eigenvalues tend to not

be close in value. Contrary to this characteristic are the eigenvalues from matrices that are a member of the Poisson ensemble. The eigenvalues of these matrices do not exhibit repulsion and are in no way correlated. Therefore, it would not be unlikely to have eigenvalues with similar values. These two opposing characteristics allow for a straightforward way to distinguish when a matrix is a member of one ensemble and not the other.

From RMT, it is known that when a matrix is a member of the GOE, it will closely follow GOE statistics described by the Wigner-Dyson distribution, also known as the Wigner surmise:

$$P_{GOE}(x) \approx \frac{1}{2}\pi x e^{-\pi x^2/4}. \tag{2.17}$$

As for Poisson statistics, if the eigenvalue spacings show no correlation then the distribution will be given by the negative exponential distribution:

$$P_{Poisson}(x) = e^{-x}. \tag{2.18}$$

The two distributions contrast each other the most when their behavior at small values of x are compared:

$$\lim_{x\to 0} P_{GOE}(x) = 0 \tag{2.19}$$

whereas

$$\lim_{x\to 0} P_{Poisson}(x) = 1. \tag{2.20}$$

This marked difference between the two ensembles as $x \to 0$ creates a significant demarcation that allows for a method to easily distinguish between the two distributions. This method gives rise to a protocol or algorithm which enables RMT to be applied as a data filtering technique.

2.5.7. Random Matrix Theory Algorithm for Applications. RMT is a thresholding technique that is knowledge-independent. This means that the algorithm is not affected by what the actual data is measuring. The threshold is a cutoff number between two units

RMT Algorithm

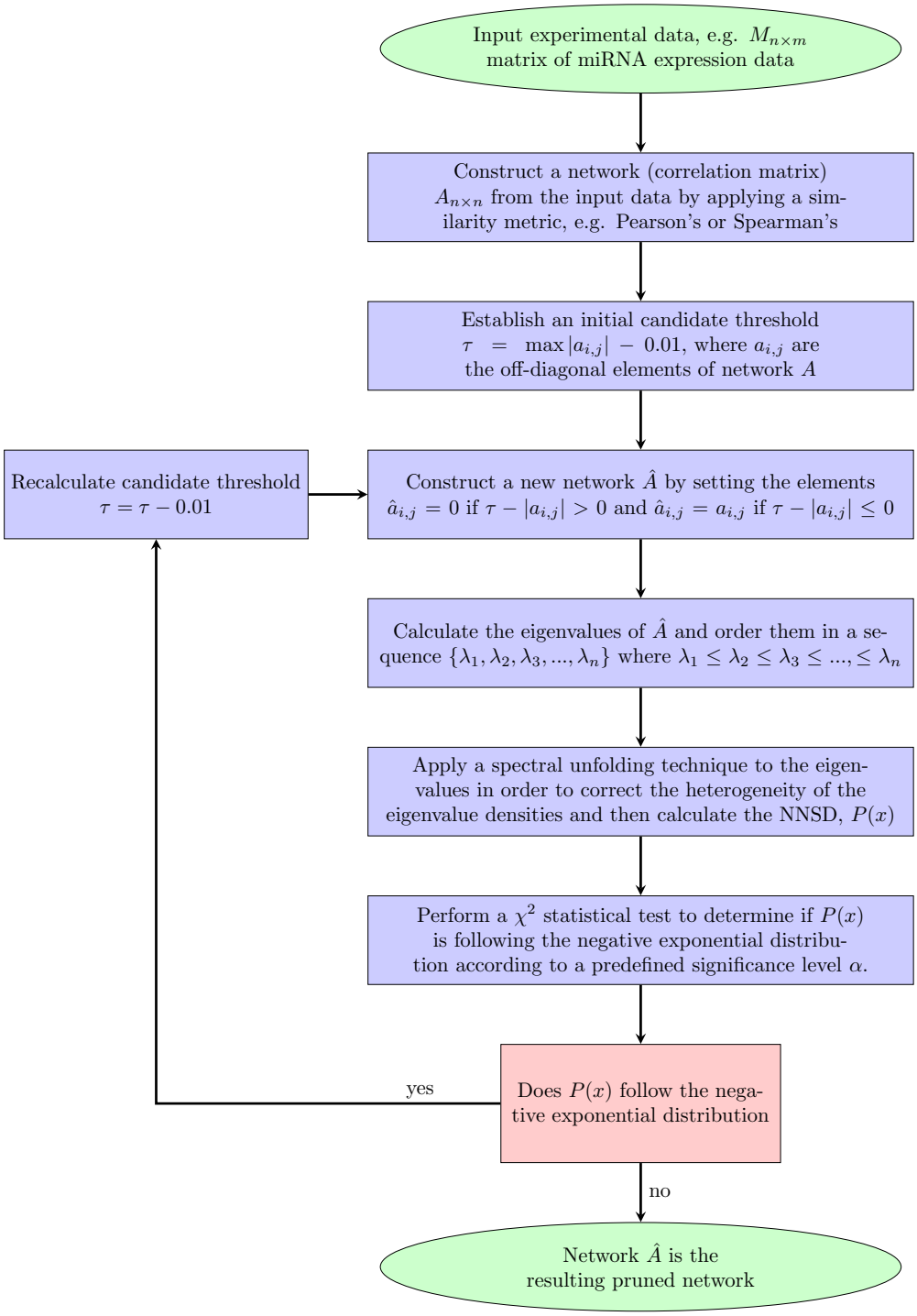


Figure 2.2. A flowchart illustrating the steps and logic of the RMT algorithm.

or elements that determines if they are correlated or not. The threshold will be determined from the transition of the NNSD of the eigenvalues from statistics described by the Poisson process to statistics described by the GOE.

The eigenvalue fluctuations follow two universal predictions depending on the correlation or non-correlation that the eigenvalues exhibit. The first prediction is that if the data of interest is mostly saturated with random noise, static, or non-information—which masks the expression of actual correlations—then these data will follow the statistics described by GOE. Specifically, the eigenvalues will exhibit a strong correlation, as described by the semi-circle law, and the NNSD of the eigenvalues will closely follow the Wigner-Dyson distribution. A matrix with completely random entries is the archetype which follows GOE statistics. This is because the off-diagonal entries, which represent mutual relationships between the diagonal entries, are predominantly non-zero and therefore induce a strong correlation between the eigenvalues. The NNSD will consequently follow the Wigner-Dyson distribution. The second universal prediction is that if there is mostly just system specific information left in the data, then the data has been sufficiently denoised and so the eigenvalues will not exhibit a strong correlation. This is because the data is mainly left with only non-zero values for the diagonal (or block-diagonal) entries. The NNSD from this kind of data will follow Poisson statistics because of the lack of eigenvalue correlation.

Now, a more specific description of the RMT algorithm is provided. Figure 2.2 is provided to help illustrate the RMT algorithm logic. Consider a real-valued, symmetric, $N \times N$ matrix A, such as a correlation matrix. This correlation matrix A is made up of elements which signify highly correlated elements, A_{hc}, and also elements that are weakly correlated A_{wc}. The elements A_{wc} represent the spurious correlations that the matrix contains. The highly correlated elements A_{hc}, which represent actual meaningful data, along with the weakly correlated elements A_{wc}, which represent irrelevant data, together make up the entire matrix:

$$A = A_{hc} + A_{wc}. \tag{2.21}$$

The goal is to find a threshold of correlation, at which point all or most of the elements from A_{wc} have been removed and therefore the remaining elements are mostly true correlations from A_{hc}: $A \approx A_{hc}$. In order to remove the A_{wc} part from the data, the RMT algorithm is needed. Suppose a set of experimental data is presented as a matrix $M_{n\times m}$, where n is the number of variables and m is the number of experimental samples in the data. The first step is to choose a statistical method, such as one of the four similarity metrics described in Section 2.2, to calculate the correlations between the unique variables (rows). The next step is to create a similarity matrix (correlation matrix) $A_{n\times n}$ by applying one of these similarity metrics. The last step is to determine a threshold of correlation by performing the RMT algorithm. Let us describe the step for determining a threshold of correlation in more detail.

First, one establishes an initial threshold whose value is significantly larger than most of the other values in the matrix. This step of establishing the first threshold value is somewhat arbitrary. However, the value needs to be high enough so that one can be fairly certain that anything with a value above this threshold (in absolute value) will actually be correlated and not just have coincidental correlation. So for example, if two elements have a correlation value c, where $|c| \geq 0.99$ on a scale from -1 to 1, one can be fairly certain that those two elements are not randomly correlated.

After choosing a threshold that has a significantly high value, set every element in the correlation matrix that is below this threshold to zero. Then calculate the eigenvalues and order them in a sequence so that the level spacings can be determined. After appropriately normalizing the spacings, the distribution $P(x)$ is calculated. Then determine which distribution, the Wigner-Dyson or negative exponential, the eigenvalue spacings most closely follow. A χ^2 test is used to gain an understanding of how closely or how far away the NNSD of the eigenvalues is from the negative exponential distribution. If the χ^2 test statistic value is less than 100 (the default value in the software RMTGeneNet that was used in this thesis), then the stated process is repeated. The χ^2 test was described in detail in

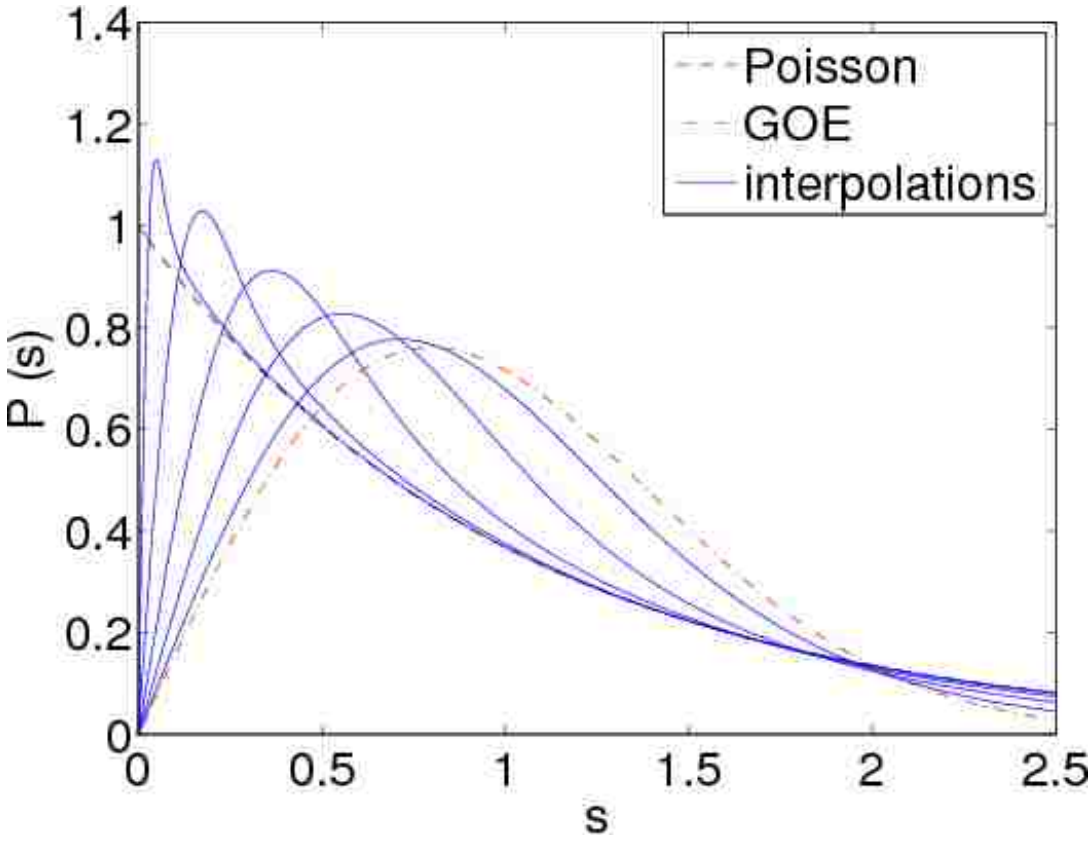


Figure 2.3. This plot [71] illustrates the transition of the NNSD of the eigenvalues of a correlation matrix from statistics described by the GOE, the Wigner-Dyson distribution, to statistics described by the Poisson process, the negative exponential distribution. The transition indicates when a matrix has been sufficiently denoised.

Subsection 2.5.6. The threshold is then iteratively refined by decreasing the value by a small increment and repeating the steps to determine which distribution the NNSD most closely follows. For example, if the starting threshold value is 0.920 (the default starting threshold for the RMTGeneNet software used in this thesis) then this initial estimated threshold can be decreased by 0.001 to the value of 0.919 and then continually decreased until the χ^2 test statistic value is greater than or equal to 100. Note that a χ^2 value of 100 is a reference point and a more or less stringent value may been chosen, depending on the level of certainty desired for one's experiment.

A less arbitrary way to set the cutoff threshold could be to base it on the maximum correlation value in a correlation matrix. So for example, a default initial candidate threshold value c of an $N \times N$ correlation matrix A could be set according to the following:

$$c = \max\{|a_{ij}|\} - 0.01 \tag{2.22}$$

where $1 \leq i < j \leq n$ because the diagonal elements should not be counted; they all have a value of 1 in a correlation matrix. This method for determining an initial threshold would provide a starting point that is less arbitrary and better suited for the specific data set being analyzed. The point at which the NNSD transitions from the negative exponential distribution to the Wigner-Dyson distribution marks the threshold of correlation. This means that everything above that threshold value represents actual correlations in the correlation matrix and everything below this value is simply static and can be ignored. There should be a sharp transition the from negative exponential distribution to the Wigner-Dyson distribution [51]. Figure 2.3 illustrates the transition that NNSD makes from Poisson to GOE statistics. Also, refer to Flowchart 2.2 for the RMT algorithm logic.

To investigate the statistics of the eigenvalue distributions of various correlation matrices, an unbiased way of comparing their eigenvalues is needed. The eigenvalues need to be standardized because the density of eigenvalues for different correlation matrices is not constant, generally speaking. Because the densities are generally not comparable, it is difficult to separate the global variation of the eigenvalue densities from the underlying universal properties one matrix should exhibit since it is a member of an ensemble of matrices [96, 97]. In order to compare eigenvalues of different systems, the mean density needs to be constant, for example by normalizing to 1 [35]. This is done via spectral unfolding methods, which map the eigenvalues to a new sequence but maintain some of the system-specific properties, or universality. Essentially, unfolding maps each eigenvalue λ_i to another value $\tilde{\lambda}_i$ in such a way that the resulting unfolded eigenvalue density is constant. Therefore, without loss of generality, when the NNSD of the eigenvalues is mentioned it

will be implied that a spectral unfolding technique was used to obtain a constant density of eigenvalues. There are various methods for unfolding the eigenvalues as well, such as the Gaussian kernel density or a cubic spline interpolation on the cumulative distribution function.

2.5.8. RMTGeneNet. The software used in this thesis is called RMTGeneNet v1.0a. RMTGeneNet is an open-source software package available on a Github repository at the web address https://github.com/spficklin/RMTGeneNet. This software was developed by Gibson et al. [32] to create gene co-expression networks using the RMT algorithm. It is an improved version of the RMT algorithm, made to be highly scalable, based on a software program written by Luo et al [52] that implemented the RMT algorithm to show how RMT can be applied to biological systems.

The RMTGeneNet software has three main functions called: 'similarity', 'threshold', and 'extract'. The first function, 'similarity', is used to construct a correlation matrix, or network. The user must supply a matrix of raw data, for instance a gene expression matrix constructed from a method such as microarray analysis or RNA-sequencing. The format of the raw data being imported should be a tab-delimited file, where the columns represent the experiment samples and the rows represent the variables that are being measured. An archetype data set is the sample data that is provide with the RMTGeneNet software. The column headers indicate the unique experimental sample ID's, the rows are the gene ID's, and the entries are the gene expression levels. Once the raw data have been supplied, the user must then decide which similarity metric to choose that will generate a correlation matrix. The 'similarity' function uses a pair-wise similarity calculation to output a correlation matrix from the input expression data. RMTGeneNet supports three similarity metrics: Pearson, Spearman, and Mutual information. This prerequisite function is the first step in order to apply the thresholding algorithm, refer to Figure 2.2.

After the correlation matrix is created via the 'similarity' function, the next function called 'threshold' can be applied. This function finds the cutoff threshold that indicates when the correlation matrix has been sufficiently filtered to the point where only significant correlations remain. The 'threshold' function works by first establishing an initial default threshold value of 0.92, which is a parameter that can be altered. It gradually iterates through successively smaller possible correlation thresholds—default decreases of 0.001 at each iteration—and computes the NNSD of eigenvalues at each iteration. This algorithm continues to iteratively decrease the threshold values until the NNSD stops following the negative exponential distribution. A χ^2 test is performed to calculate how closely the NNSD follows the negative exponential distribution. The NNSD is considered to have sufficiently diverged from the negative exponential distribution when a default value of $p \sim 0.001$ ($\chi^2 = 100$, with 59 degrees of freedom) has been reached. The software allows the user to alter this default significance value to allow for more or less stringency.

The threshold value identified can then be applied to the correlation matrix which essentially will weed out the random data and leave only the system specific signal. That is the role of the 'extract' function. It actually applies the correlation value that was identified by the 'threshold' function to the original correlation matrix produced from the 'similarity' function. The output of the 'extract' function is a table, called a network, with all of the correlations that were at or above the identified threshold. The user can then investigate the correlations in this network with a visualization tool such as Cytoscape [73]. Cytoscape is a tool for gaining new insights by visually exploring relationships in data.

The reason the RMTGeneNet software was utilized in this thesis is twofold. One reason was to establish a proof of concept of applying RMT using the RMTGeneNet. Luo et al. [51, 52] used RMT in their biological studies and showed it was a robust and sensitive algorithm for network pruning. Gibson et al. [32] were then able to build upon this work to use RMT for the construction of massive gene co-expression networks by creating the RMTGeneNet software.

The second reason was because RMTGeneNet already had some built in functionality that was needed for our analysis; there was no need reinvent the wheel. This software had the Spearman and Pearson similarity metrics already built into its 'similarity' function. RMTGeneNet also had efficient code written for applying the RMT algorithm and finding the threshold of correlation without any user interaction after the initial call of the 'threshold' function. This was a very convenient aspect that influenced the decision to implement RMTGeneNet for applying the RMT algorithm to the data over other possible software packages.

2.5.9. Candidate Software Packages. Other candidate software packages were tested and vetted for applying the RMT algorithm. Although, these software packages were not used in their full capacity to implement the RMT algorithm, specific aspects of them were used to aid in various aspects of the data analysis.

One software package that was useful, but was not ultimately used for its application of the RMT algorithm, is called RMThreshold [54]. RMThreshold is a CRAN - R package developed in 2016. This software package contains some useful functions and graphing capabilities. For example, RMThreshold has a function called 'rm.matrix.validation' which can be applied to an input matrix (network) to determine if it is well-conditioned for the RMT algorithm. By well-conditioned, it means that the matrix must be real-valued, symmetric, and sufficiently large. The matrix must be sufficiently large because RMT is valid as the dimensions of the matrix approach infinity. This function therefore checks all of these requisite conditions and creates validation plots that help the user determine if the RMT algorithm is the viable option for the analysis of the given data.

RMThreshold applies the RMT algorithm to find a threshold of correlation with a built in function called 'rm.get.threshold'. One helpful and useful aspect of this function is that it creates plots of the NNSD, in conjunction with the negative exponential distribution

and the Wigner-Dyson distribution, at each tested candidate threshold of correlation. This makes it possible to see the change in the NNSD as it transitions from one distribution to the other.

The aspect of this functionality that was not particularly preferred for the analysis in this thesis was that it required user interaction to estimate the threshold of correlation. The user would have to determine a possible threshold of correlation by examining the output plots at each iteration. There was no χ^2 test to statistically determine when the NNSD had sufficiently transitioned from the negative exponential distribution to the Wigner-Dyson distribution. This made it less objective than the procedure that RMTGeneNet implements for determining the threshold of correlation in the data. The interactive requirement from the user to determine the threshold of correlation by analyzing diagnostic plots also made it impractical for someone to use if connected to another computing device. For example, many high-performance-computing (HPC) platforms require remote access by secure shell logins and require users to submit jobs, which makes it impossible for shell interactions. This was another reason why RMThreshold was not ultimately used for implementing the RMT algorithm.

The interactive RMThreshold software was helpful in some situations. For example, there are times when RMTGeneNet, the non-interactive software, might not perform as desired. If the RMT algorithm is applied to a matrix that is not well-conditioned, as described previously, then a threshold may not ever be found. There are no troubleshooting diagnostics to perform with RMTGeneNet, so the user would not be notified about what went wrong. A more specific example of this instance is if RMTGeneNet is used to apply the RMT algorithm to a matrix that is overly sparse. There may not be enough noise in the data so that there is never a transition from the Wigner-Dyson distribution to the negative exponential distribution. Recall, this transition would indicate the point at which the matrix transitions from a member of the GOE—a class of random matrices—to a matrix that follows statistics described by the Poisson process—a matrix without random correlations

in the data. On the other hand, with the interactive software of RMThreshold, the user would be able to tell right away that a threshold may not be found after applying and analyzing the appropriate built-in diagnostic function (i.e. the 'rm.matrix.validation' function). So, it would behoove someone that is wanting to apply the RMT algorithm to his or her data to first apply the diagnostic functions to the data that the RMThreshold software contains. After verifying that the data is well-conditioned, a threshold of correlation could then be found with the RMTGeneNet software. This is precisely the analysis process performed in this thesis.

Another notable R software package called KINC.R was also investigated for possible utility in this thesis. KINC.R is a lightweight package, released in 2016, that contains a function for applying the RMT algorithm. Similar to the method the RMTGeneNet software implements, KINC.R uses a χ^2 test to evaluate when the matrix has been sufficiently thresholded. This software prints to screen the analysis that is being performed in real-time. The user must then parse through this output to find the transition point in the data, or the threshold value. KINC.R was less automated than the procedure the RMTGeneNet software implemented; however, the real-time plots were beneficial for verifying the NNSD behavior as the RMT algorithm was performed.

Several other software packages exist that implement the RMT algorithm, besides the ones previously described. It was not possible, however, to fully investigated every software package. It was encouraging to discover that the RMT algorithm had been implemented in several software packages already. This further supported the assertion that RMT is becoming a popular tool because of its effectiveness in filtering data in an objective manner.

Of the software packages investigated, RMTGeneNet was the best suited software for our analysis. Albeit, there were a few things that were needed for the analysis that it did not have. The functionality of RMTGeneNet was extended by including the correlation metrics

to its repertoire: 1) the Czekanowski index and 2) the stringent proportional similarity index (SPS). This extended functionality added to the RMTGeneNet software will be described in Section 3.2.

3. METHODS AND MATERIALS

This section will discuss the data to which the RMT algorithm was to applied to in order to investigate its capability as a network pruning method on biological data. A description of how the data were processed and formatted is provided. The tools used to verify the appropriateness of applying the RMT algorithm to the data are presented, along with a description of the other software tools that were used in various stages of the analysis. These tools include implementations of the RMT algorithm, as well as the software too Cytoscape [73], which used to analyze the co-expression networks after the RMT algorithm had been applied to prune them.

3.1. DATA DESCRIPTION

The data used in this thesis are microRNA (miRNA) data from the *Bos taurus* (domestic cow). MiRNAs are small RNA sequences, approximately 22 nucleotides in length, that are involved in gene expression regulation [11]. Mounting experimental evidence from miRNAs studies suggest that miRNAs play a more significant role in cellular functions than previously thought [3, 37]. The *Bos taurus* miRNA data used in this thesis were adopted from a study that investigated the potential uses of circulating miRNA as a signature for early pregnancy in dairy cattle [43]. The data set is publicly available in the ArrayExpress database [21].

3.2. DATA FORMATTING AND PRE-PROCESSING

The miRNA data downloaded from the ArrayExpress database were formatted as a table that contained 870 rows and 46 columns, where the rows indicated the miRNA sequences and the columns were the miRNA sequencing samples (i.e. samples taken from

19 different heifers at various stages of their estrous cycle or pregnancy). The data entries are the expression levels from profiling the miRNA sequences in the biological samples found by applying Illumina small-RNA to their collected plasma samples.

Some of the miRNA sequences were found to not be expressed at all, or very little, across all the samples. To reduce some of the sparsity in the data, the data were preprocessed to reduce some of the potential effects these outliers might have had on the construction of the co-expression networks and ultimately on the analysis of the RMT algorithm. Therefore, a Python script was written to remove any row in the table whose sum did not add up to a value of 5 or more, indicating the lack of expression of that miRNA sequence across all samples. A value of 5 was chosen, by inspection, as the cutoff to filter out these miRNAs that did not have a significant presence in the data. This preprocessing procedure reduced the data by nearly 29%, leaving 618 rows and 46 columns in the data set.

After preprocessing the raw data, four networks were then created by applying the four similarity metrics: Pearson, Spearman, Czekanowski, and SPS, which are detailed in Section 2.2. Applying these metrics to the miRNA expression data was made possible by writing a Python script that implemented the packages: scikit-learn [59], NumPy [83], and Pandas [53].

3.3. APPLYING RMT

After the data were preprocessed to filter out any severe sparseness, the next step was to create the co-expression networks. Four co-expression networks were created by applying each of the four different similarity metrics discussed in Section 2.2 to the miRNA data set. After the networks were created, the RMT algorithm could be applied to prune the co-expression networks by finding the threshold of correlation. Before the RMT algorithm was implemented, each network was inspected with the R package called RMThreshold [54]. This was done in order to validate the appropriateness of applying the RMT algorithm to each network. Specifically, the RMThreshold package contains a built-in function called

'rm.matrix.validation', which was applied to each of the four networks. This function was applied to the networks to determine if they were well-conditioned for the RMT algorithm: well-conditioned meaning the input matrix is symmetric, sufficiently large, and not overly sparse. Refer to Section 2.5.9 for further discussion of RMThreshold and the 'rm.matrix.validation' function. After this validation process, the networks were pruned by applying the RMT algorithm that the RMTGeneNet software contains [32], which was described in Section 2.5.8.

This software contains three functions that, when applied in succession to a data set, will yield a product that is a SIF formatted file. This SIF file will contain the network that was created from the input data after a particular similarity metric had been applied and then was pruned with the RMT algorithm. Recall, the SIF formatted file is a configuration of the data that is a simple and convenient way to create graphs from a list of interactions; see Section 2.1 for more information regarding the SIF format. The resulting pruned network will have been denoised so that only the remaining constituents will have true correlations. The interactions between different components which were pruned from the network would have been randomly correlated according to RMT algorithm. The SIF formatted networks can then be uploaded into a network visualization tool, such as the Cytoscape software [73], for visual inspection and heuristic hypothesis generation.

The RMTGeneNet software was used for analysis in this thesis because it already had the proven capability of applying the RMT algorithm [32], and thus allowed us to take advantage of existing code that has been thoroughly evaluated. RMTGeneNet supports three similarity metrics: Pearson, Spearman, and Mutual Information. To apply the RMT algorithm implemented in RMTGeneNet, the user must first input the raw data, such as the *Bos taurus* miRNA data downloaded directly from ArrayExpress [21]. Next, the user must apply one of the three supported similarity metrics with the built-in function called 'similarity' in order to construct a network. Next, the RMT algorithm can be applied to the resulting network with the function called 'threshold'. Applying the 'similarity' function

and the 'threshold' function is a streamlined process that works very well if the user is only wanting to create his or her network from one of those three supported similarity metrics. A caveat is if the user wants to apply the RMT algorithm to a network created by some other similarity metric, e.g. Czekanowski or SPS metrics, RMTGeneNet does not currently have the functionality to facilitate this kind of flexibility.

Additional functionality was augmented to the software to be able to apply the RMT algorithm to the co-expression networks that were created with the Czekanowski and SPS metrics. An interfacing script was written so that the RMT algorithm in RMTGeneNet was still utilized, but any similarity metric of interest could be applied, not just the Pearson, Spearman, or mutual information metrics that the software supports. This interfacing script supports the use of the four similarity metrics that were used in this thesis, namely the Pearson, Spearman, Czekanowski, and SPS metrics. The user inputs the raw data as an argument in the script, along with the desired similarity metric to create the network. The script creates the desired network and then automatically applies the RMT algorithm that RMTGeneNet implements to find the threshold of correlation and prune the network.

In addition to finding the the threshold of correlation with the RMTGeneNet software, the R package KINC.R [26] was used to verify the threshold of correlation found with the RMTGeneNet software. The KINC.R software provides real-time plots of the NNSD of the eigenvalues. The user can use these plots to verify the NNSD transitioning from the negative exponential distribution to the Wigner-Dyson distribution once the data has been sufficiently denoised.

The miRNA co-expression networks pruned by applying the RMT algorithm were then uploaded into the visualization software called Cytoscape [73]. Cytoscape allows the user to interactively explore networks in real-time. This is a helpful tool for developing new hypotheses about the structural and functional relationships in data. This tool uses edges (lines) to connect different nodes (objects) which are associated to each other in some way. It provides a convenient platform on which data can be visually and analytically scrutinized.

Cytoscape has been shown to be a powerful tool for complex network analysis and for making hypothesis about uncharacterized nodes by using a guilt-by-association approach [91]. In the case for this study, the nodes represented the different *Bos taurus* miRNA sequences and the edges represented an association between different miRNAs whose co-expression correlation values were at least as high as the threshold of correlation identified with the RMT algorithm.

4. RESULTS AND DISCUSSION

In this section, the results from applying the RMT algorithm to a set of *Bos taurus* miRNA expression data, an open source data set at the time of this publication [21], are presented. After the miRNA data were download and parsed to remove any unwanted extreme sparsity, the four similarity metrics Pearson, Spearman, Czekanowski, and SPS were applied to the data. This resulted in the construction of four co-expression networks that were of the dimensions 618×618. The structure of this section is as follows: 1) a discussion of the network validation process, 2) details of the identified threshold for each co-expression network and the corroboration of these values using an alternative RMT software called KINC.R, and 3) an analysis of the pruned co-expression networks after applying the identified thresholds, which includes an introductory demonstration of network analysis with the network visualization tool Cytoscape [73].

4.1. NETWORK VALIDATION PROCESS

The RMThreshold package [54] was used to validate the appropriateness of using RMT as a pruning tool for each of the networks. A built-in function from the RMThreshold package called 'rm.matrix.validation' analyzes an imported network (matrix) to determine if it is well-conditioned for the RMT algorithm. This function checks to make sure the network is not too sparse, is symmetric, sufficiently large, and has real-valued entries. If the network is too sparse, then a threshold of correlation may not be found because it is as if there is no randomness in the data, i.e. the matrix is already thresholded. The matrix must be sufficiently large because RMT is valid for matrices with the dimensions $N \times N$ as $N \rightarrow \infty$, as described in Subsection 2.5.4. The matrix being symmetric and having real-valued entries are also requisite properties in order to properly apply the RMT algorithm for thresholding.

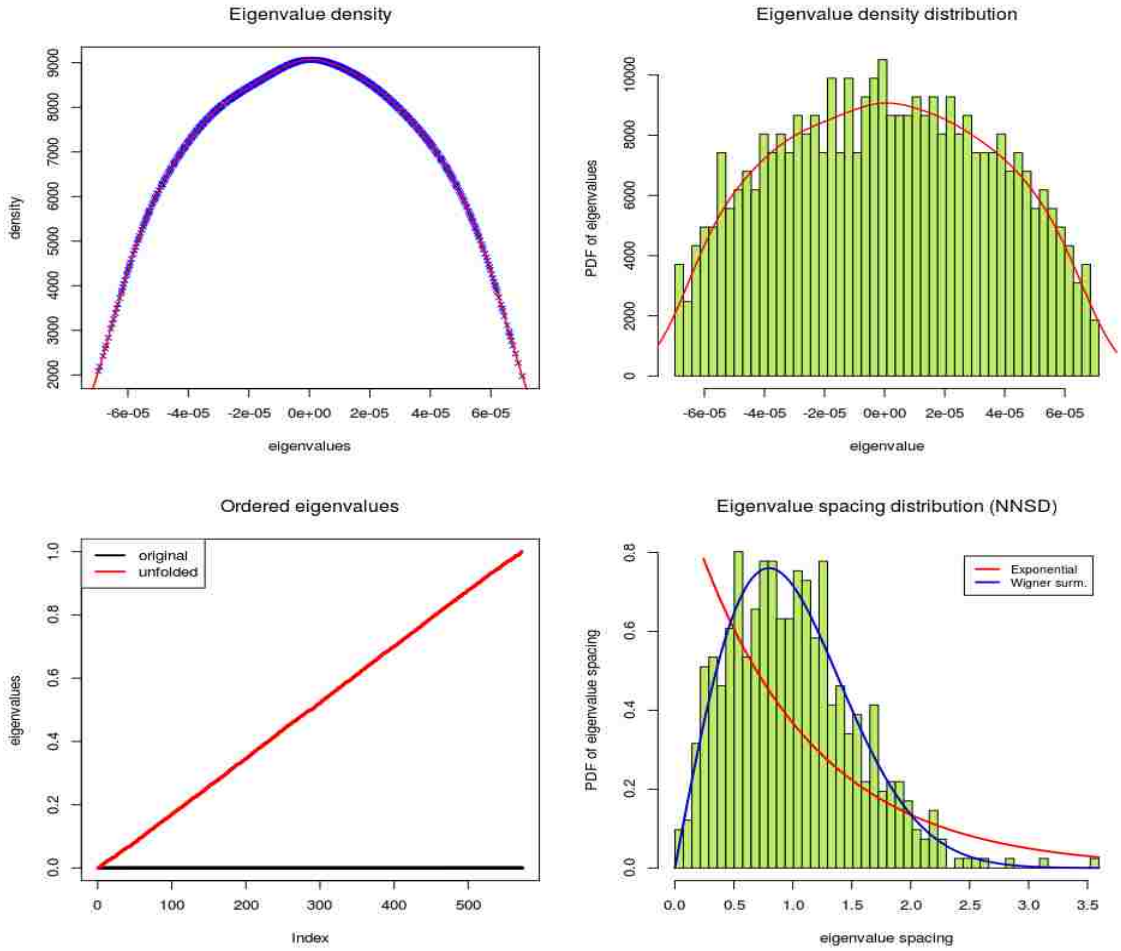


Figure 4.1. RMT Validation Plot. A function called 'rm.matrix.validation' from the R package RMThreshold was performed on each network to validate the appropriateness of pruning the networks with the RMT algorithm. Above are the results from applying this validation function to the miRNA network that was created with the Pearson correlation metric.

To illustrate the functionality of 'rm.matrix.validation', Figure 4.1 demonstrates the results when applied to the network that was produced by applying the Pearson similarity metric to the *Bos taurus* miRNA data. This validation plot produced by the 'rm.matrix.validation' function is a diagnostic tool for verifying that the input network is well-suited for the RMT algorithm. Recall that Figures 2.1a and 2.1b illustrate how the eigenvalue density distribution of a Gaussian random matrix follows the Wigner semi-circle law as the dimensions of the matrix increase. The plots in the upper left and right-hand

corners of Figure 4.1 illustrate the eigenvalue density distributions following the semi-circle distribution, an exhibition characteristic of random matrices. This distribution demonstrably following the semi-circle distribution supports the reasoning that the input network needed to be pruned. The plot in the lower left-hand corner of Figure 4.1 illustrates the requisite eigenvalue unfolding that is described in Subsection 2.5.7. The image in the lower right-hand corner of Figure 4.1 demonstrates how the nearest neighbor spatial distribution (NNSD) of the eigenvalues followed the Wigner-Dyson distribution, which was another indication that the network contained noise and needed to be pruned. This validation procedure was performed on all four networks in order to substantiate the use of the RMT algorithm as a viable pruning method for the particular miRNA data set.

4.2. ANALYZING EACH IDENTIFIED THRESHOLD

Once the networks were created by applying the four similarity metrics, and they were all deemed viable candidates for the use of the RMT as a pruning tool, the RMT algorithm was then applied. The RMT algorithm was applied with the implementation of the RMTGeneNet software [32]. Recall, that the RMT algorithm that this software implements is illustrated with Flowchart 2.2. The threshold of correlation is identified by determining the point at which the NNSD of the eigenvalues has deviated from the negative exponential distribution and begins to follow the Wigner-Dyson distribution. Specifically, the algorithm will iteratively test smaller and smaller candidate threshold values until the χ^2 value reaches 100, which marks the point at which the NNSD has sufficiently diverged from the negative exponential distribution and thus random correlations are beginning to mask the actual correlated data. RMTGeneNet will continue to iteratively test smaller candidate threshold values until a χ^2 value of 200 is reached. These additional computations are described to be a preventative measure so that the software does not indicate a threshold

that may have been part of a local maximum [32]. Ultimately, the last threshold value whose associated χ^2 value was less than 100, before a χ^2 of 200 is reached, will be identified as the threshold of correlation for the network.

The threshold of correlation found by applying RMTGeneNet to the networks created by the Pearson, Spearman, Czekanowski, and SPS similarity metrics were: $0.81, 0.77, 0.78$, and 0.63, respectively. These thresholds indicate the point at which significant correlations begin to emerge in the data. All correlations that are not as strong as these thresholds are therefore putative random correlations in the data. Note that for the networks that were created with the Pearson and Spearman metrics, the correlation values are in the range $[-1, 1]$. Therefore, any correlation value in these two co-expression networks whose *absolute* value is less than the indicated threshold is deemed as noise.

The R package KINC.R [26] was additionally applied to each of the networks in order to corroborate the different thresholds that were found. KINC.R provides a way to visualize the behavior of the NNSD from each of the networks, in real-time, as the RMT algorithm is being applied. At each iteration, a successively smaller candidate threshold value is evaluated and a plot of the NNSD is produced with superimposed plots of the Wigner-Dyson distribution along with the negative exponential distribution. This allows the user to observe the transition of the NNSD from one distribution to the next as the threshold of correlation is iteratively identified.

4.2.1. The Pearson Network. In Figure 4.2 are four snapshots that were created as the RMT algorithm was being applied to the co-expression network created with the Pearson similarity metric. Note that the four histogram plots in Figure 4.2 depict precisely what is illustrated in Flowchart 2.2. Plot 4.2a is the real-time plot of the NNSD when the initial candidate threshold value 0.88 was tested. Recall that the initial tested threshold should be stringent enough so that there will be very few, if any, random correlations contained in the network. The initial tested threshold of 0.88, after being applied to the original 618×618 network, resulted in a 196×196 network. This is a $\sim$ 90% reduction in node-to-node

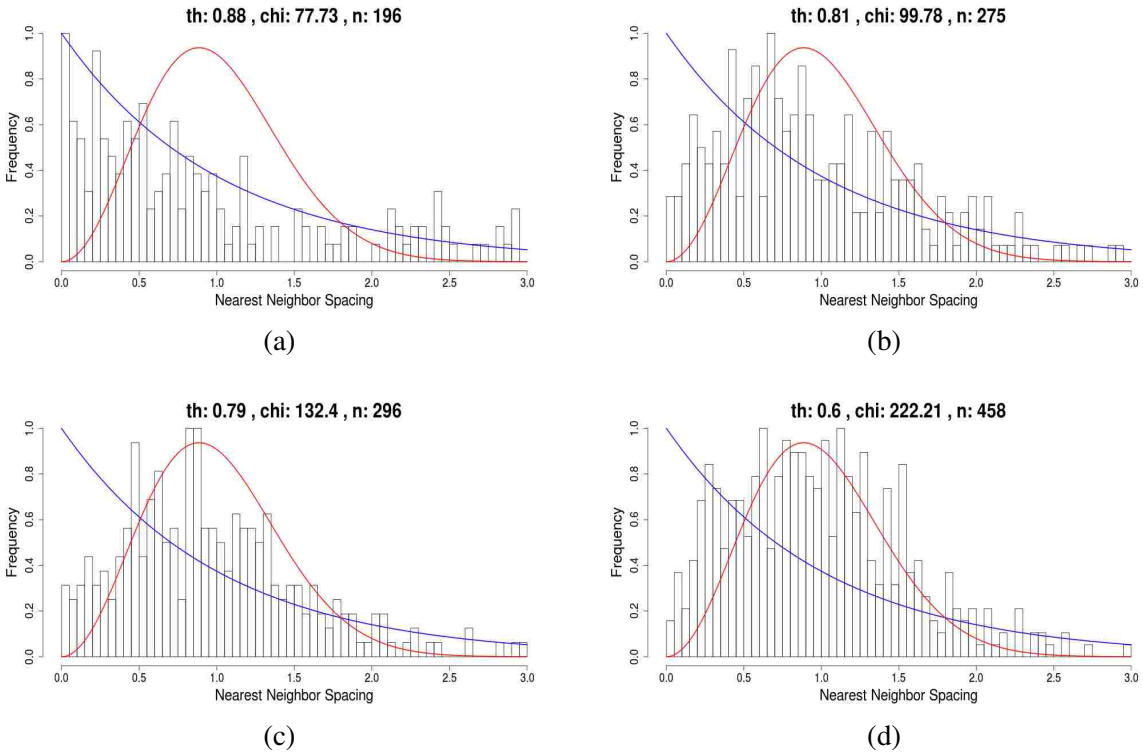


Figure 4.2. The Pearson Network: An illustration of the NNSD transitioning from the negative exponential distribution (blue) to the Wigner-Dyson distribution (red) while the RMT algorithm is begin applied. A threshold value of 0.81 was identified with the RMTGeneNet software for this network, which is corroborated with Plot 4.2b in this figure. in this figure.

correlations in the original network; a significant reduction that has likely filtered out some actual correlations in the network. As the candidate threshold is iteratively decreased, the network will begin to include more and more elements. Figure 4.2b is a plot of the NNSD when the candidate threshold tested was equal to the actual threshold value of 0.81 that was found using the RMTGeneNet software. A χ^2 value of 99.78 was attained at this tested candidate threshold, which indicates that the NNSD has all but diverged completely away from the negative exponential distribution. It marks the last tested candidate threshold whose χ^2 value was less than 100; all other successively tested thresholds produced a χ^2 value that was greater than 100. This provides strong evidence that the null hypothesis, H_0, which states that the NNSD of the eigenvalues is from the same random variable as a

random variable that follows the negative exponential distribution, can be rejected. In other words, the threshold value of 0.81 is the demarcation point between random correlations and true correlations.

The transition is marked when visually comparing the four plots in Figure 4.2. Plot 4.2a followed the negative exponential, superimposed in blue, when the initial threshold value of 0.88 was tested. Then, in Plot 4.2b, the NNSD has nearly diverged completely from the negative exponential and is clearly beginning to follow the Wigner-Dyson distribution, superimposed in red, when the tested threshold value was 0.81. As the candidate threshold bar is lowered, elements with weaker, and presumably random, correlations are being added to the network. In Plot 4.2d the tested threshold was 0.6 and the associated χ^2 value was 222.21; therefore, the NNSD has clearly diverged from the negative exponential distribution. The NNSD is undoubtedly following the Wigner-Dyson distribution at this point. The NNSD following the Wigner-Dyson so closely is a good indication that the matrix contains a substantial amount of randomness.

4.2.2. The Spearman Network. Similar diagnostic plots were generated for the other networks created with the Spearman metric (Figure 4.3), the Czekanowski metric (Figure 4.4), and the SPS metric (Figure 4.5). A threshold value of 0.77 was identified with the RMTGeneNet software for the network created with the Spearman metric. Plots 4.3a–4.3d illustrate the NNSD making an unmistakable transition from following the negative exponential distribution to the Wigner-Dyson distribution. Plot 4.3b is the results from testing the candidate threshold value of 0.78, a value slightly greater than the value that the RMTGeneNet software identified as the threshold of correlation. A χ^2 value of 93.29 is associated with the candidate threshold in Plot 4.3b, and therefore the RMT algorithm continues to perform successive iterations by decreasing the candidate threshold value by 0.001 at each iteration. This indicates that the threshold value tested in Plot 4.3b is putatively still a valid correlation in the network, not just a correlation caused by happenstance. As the smaller candidate thresholds were successively tested, a threshold with the value of 0.77

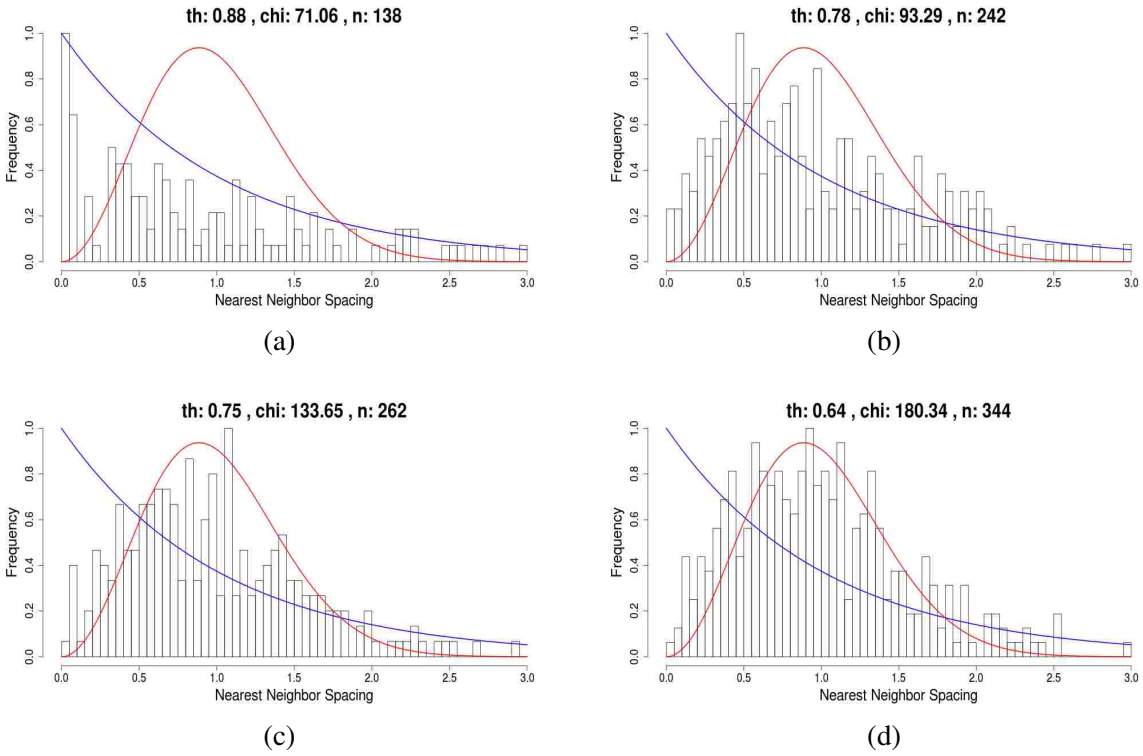


Figure 4.3. The Spearman Network: An illustration of the NNSD transitioning from the negative exponential distribution (blue) to the Wigner-Dyson distribution (red) while the RMT algorithm is begin applied. A threshold value of 0.77 was identified with the RMTGeneNet software for this network.

proved to be the last candidate threshold whose χ^2 value was below 100. As the network grew larger by decreasing the candidate threshold values, random correlations begin to infiltrate the network, as Plots 4.3c and 4.3d demonstrate.

4.2.3. The Czekanowski Network. The NNSD transition for the Czekanowski network is in Figure 4.4. Plot 4.4a is the NNSD when the initial candidate threshold value was 0.87. The χ^2 value of 56.79 associated with this threshold implies a failure to reject the H_0; the NNSD is following the negative exponential distribution. In Plot 4.4b, the χ^2 value associated with the candidate threshold of 0.77 is 104.32. This implies a rejection of the H_0 and an acceptance of the alternative hypothesis, H_1, which states that the NNSD does not follow the negative exponential distribution. The threshold of correlation identified with the RMTGeneNet software for the network created with the Czekanowski metric was 0.78.

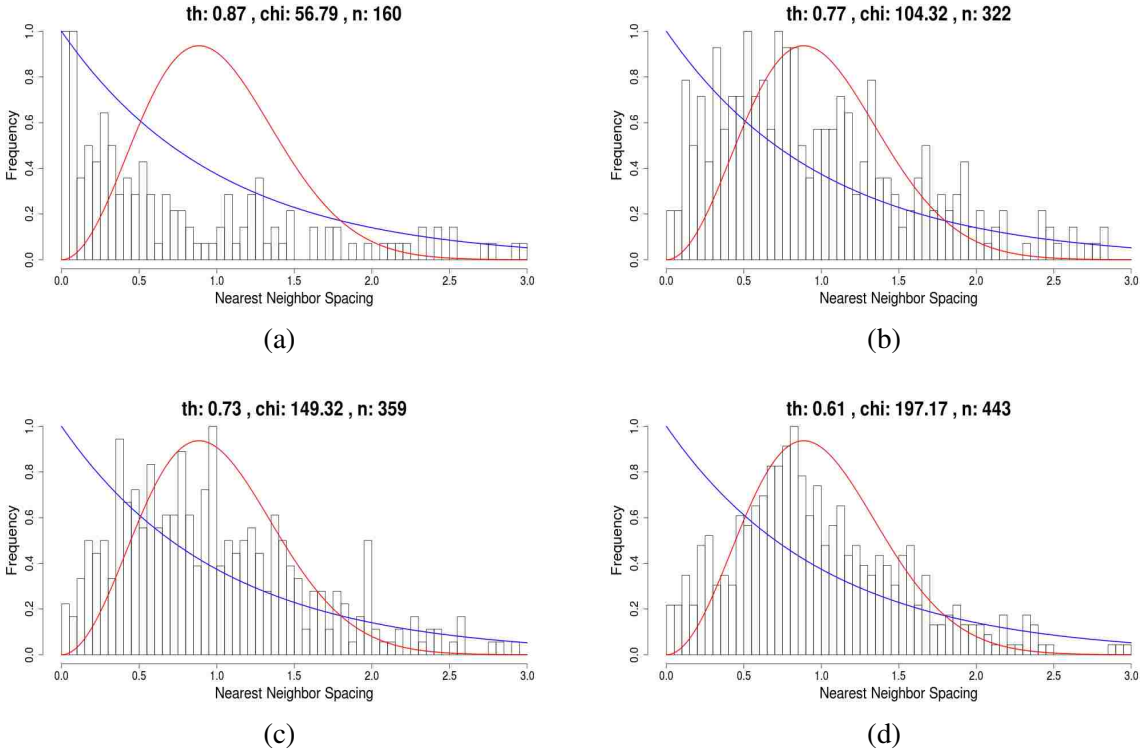


Figure 4.4. The Czekanowski Network: An illustration of the NNSD transitioning from the negative exponential distribution (blue) to the Wigner-Dyson distribution (red) while the RMT algorithm is begin applied. A threshold value of 0.78 was identified with the RMTGeneNet software for this network.

Plot 4.4b illustrates that, as candidate threshold values were successively tested after the threshold of correlation 0.78 was identified, the networks only became more infiltrated by noise, as the χ^2 value indicates. Plots 4.4c and 4.4d further solidify this claim.

4.2.4. The SPS Network. The threshold of correlation identified in the network created with the SPS metric was also corroborated by the diagnostic plots in Figure 4.5. The behavior of the NNSD during successive iterations of the RMT algorithm was similar to that of the other co-expression networks. Specifically, a sufficiently high candidate threshold was initially tested, as shown in Plot 4.5a. Note that the initial candidate threshold for the SPS network was substantially smaller than the initial candidate thresholds tested in the other networks. For the SPS network, the initial tested threshold was 0.79, which is even lower than the threshold identified to be the threshold of correlation in the Pearson

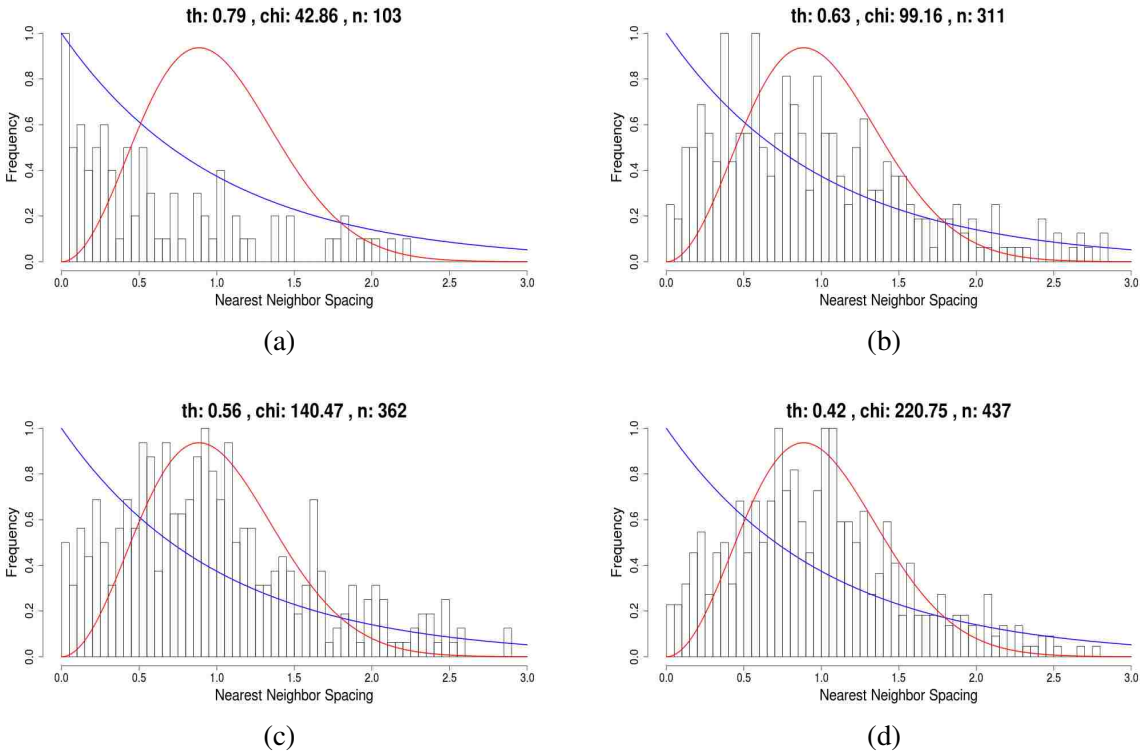


Figure 4.5. The SPS Network: An illustration of the NNSD transitioning from the negative exponential distribution (blue) to the Wigner-Dyson distribution (red) while the RMT algorithm is begin applied. A threshold value of 0.63 was identified with the RMTGeneNet software for this network.

network. This should serve as a reminder that establishing an initial candidate threshold that is sufficiently large is of utmost importance. If this same initial threshold had been used for the RMT algorithm in the Pearson network, then a threshold of correlation would not have been identified. Establishing an appropriate starting threshold was discussed in more detail in Subsection 2.5.7 and specifically with Equation 2.22.

The SPS co-expression network is reduced down from 618×618 to a co-expression network of size 103×103, which equates to a reduction of $\sim 97\%$. The NNSD markedly follows the negative exponential distribution in Plot 4.5a, which is substantiated by the χ^2 value of 42.86. The RMT algorithm iteratively decreases to the value of 0.63, as shown in Plot 4.5b, which is the threshold of correlation that was identified with the RMTGeneNet software. Successively lower tested candidate thresholds resulted in higher and higher χ^2

values, 140.47 and 220.75 for Plots 4.5c and 4.5d, respectively. This supports the claim that the identified threshold of correlation, 0.63 as shown in Plot 4.5b, is the actual threshold of correlation in the SPS network; a smaller correlation value between two elements (miRNA sequences) in the co-expression network would indicate just an anomalous correlation.

4.3. ANALYSIS OF PRUNED NETWORKS

In this section, a description of the analysis of the co-expression networks is provided. This step in the analysis occurred after the identified threshold for each network had been applied to filter out, or prune, the unwanted random correlations from the co-expression network. First, a histogram for each network was created to illustrate the distributions of the co-expression values for each of the four co-expression networks. Then, the co-expression network created with the Pearson similarity metric was uploaded into the network visualization tool Cytoscape [73]. Preliminary analysis with this tool is provided to advocate its potential potency as a powerful heuristic device when it is coupled with the network pruning method RMT.

4.3.1. Co-Expression Frequency Distributions. A histogram was created for each of the four co-expression networks so that the frequency distributions of the *Bos taurus* miRNA co-expression correlation values could be analyzed. These histograms are illustrated in Figure 4.6. The Pearson co-expression network is in Plot 4.6a, Spearman in Plot 4.6b, Czekanowski in Plot 4.6c, and SPS in Plot 4.6d. Recall that the identified thresholds of correlation for the Pearson, Spearman, Czekanowski, and SPS co-expression networks were 0.81, 0.77, 0.78, and 0.63, respectively. For the Pearson co-expression network, this resulted in a $\sim$ 80% size reduction of the original co-expression network. The Spearman co-expression network was reduced by $\sim$ 84%, the Czekanowski co-expression network by $\sim$ 72%, and the SPS co-expression network by $\sim$ 75%.

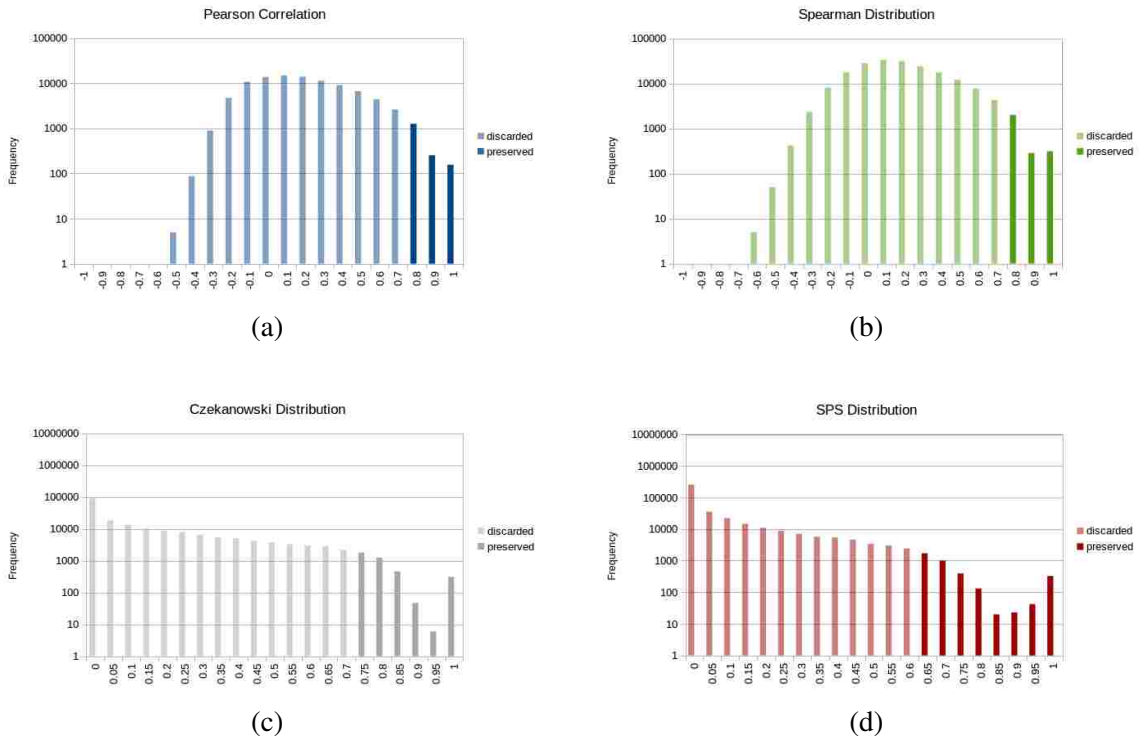


Figure 4.6. Frequency distributions of the miRNA co-expression correlation values for each of the four analyzed networks. The more translucent columns in the histograms indicate the co-expression values that were removed from the network. The more opaque columns are the co-expression values that were maintained after the RMT algorithm was applied.

The co-expression values that were pruned are depicted differently in the histograms of Figure 4.6 from the co-expression values that were retained. Namely, the values that were pruned are depicted with the more translucent columns whereas the more opaque columns depict the co-expression values that were retained. Note that the histograms created for the Pearson and Spearman correlation networks had similar frequency distributions for their co-expression values because they are calculated similarly, as can be seen from their formulation in the Subsections 2.2.1 and 2.2.2. Likewise, the frequency distributions for the co-expression values of the Czekanowski and SPS networks are similar because of their homologous formulations, as seen in Subsections 2.2.3 and 2.2.4.

4.3.2. Network Analysis with Cytoscape. After analyzing the frequency distributions of the co-expression networks, as described in the previous subsection, the co-expression networks were uploaded into the network visualization tool Cytoscape [73]. Cytoscape provides a platform on which heuristic analysis can take place. The RMTGeneNet software contains a function called 'extract', which prunes the given network with the threshold of correlation indicated by the RMT algorithm. The output from the 'extract' function is a SIF formatted table of the co-expression network which, conveniently, can be directly imported into Cytoscape. Here, the results from analyzing the Pearson network in Cytoscape are provided. There are various layouts and style options that the user can adjust, based on his or her preference. The specific layout chosen for this Pearson co-expression network was the 'organic' layout with the style option 'default black' and the 'show graphic details' option turned on so that the individual nodes can be identified by their specific miRNA ID names.

To gain perspective on the outcome of the pruned network, included in Figure 4.7 is an image of the original unpruned *Bos taurus* miRNA network. This all-vs-all network illustrated with Cytoscape demonstrates just how incomprehensible a network can be if it is not properly filtered to remove irrelevant data. The convoluted globe in Figure 4.7 consists of 618 nodes with just over $380,000$ edges. It is obvious from this figure that in order to begin making any sense of the miRNA network, much of the network needs to be eliminated. RMT takes the guesswork out of this process. The results in Cytoscape after pruning the network in Figure 4.7, with the RMT algorithm, will be expanded upon next.

Figure 4.8 is the *Bos taurus* miRNA co-expression network constructed with the Pearson metric, which was then pruned with the RMT algorithm and upload into Cytoscape. The threshold of correlation identified by the RMTGeneNet for this network was 0.81, which was corroborated with the KINC.R software as described in Subsection 4.2. Plot 4.2b indicates that the threshold identified by the KINC.R software was 0.81 and the pruned co-expression network would be 275×275. A Slight discrepancy was discovered however.

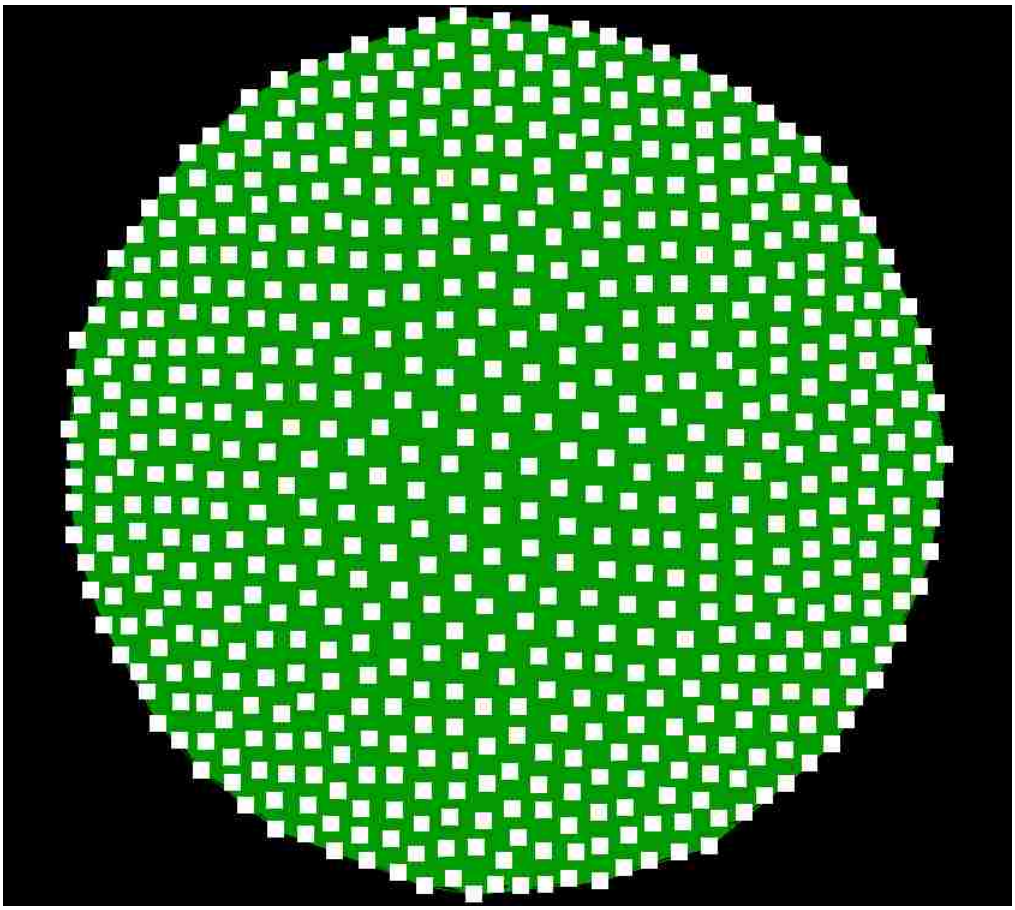


Figure 4.7. The original unpruned *Bos taurus* miRNA network created with the Pearson correlation metric.

The size of the resulting pruned networks that RMTGeneNet outputted, as opposed to the size of the pruned network that KINC.R outputted, were distinct. This was likely due to a inconsistency in the number of significant digits recognized by the two different software packages. The resulting co-expression network from the RMTGeneNet software uploaded into Cytoscape contained 277 nodes with 2669 edges.

In Figure 4.9, a small portion of the nodes was selected at random, along with the first neighboring nodes of these selected nodes. The nodes that were randomly selected are highlighted in yellow and the edges that linked these nodes to the first neighboring nodes are highlighted in red. From the selected nodes and edges in Figure 4.9, Cytoscape has the

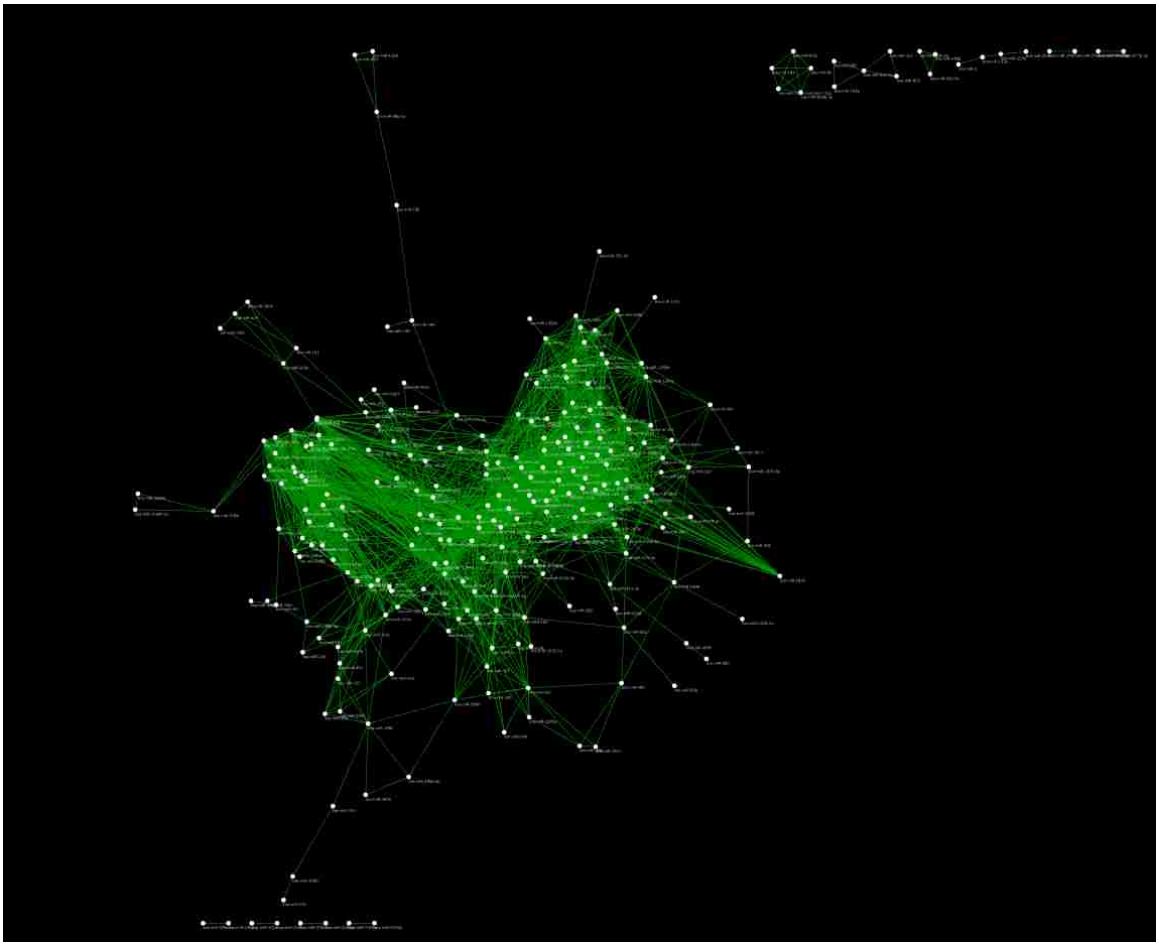


Figure 4.8. The *Bos taurus* miRNA network that was created using the Pearson correlation metric and thresholded with the RMT algorithm. The visualization of this network has been created using the Cytoscape software.

functionality to isolate this portion of the network and created a new network containing just this portion, as illustrated in Figure 4.10a. The user can then begin to construct hypotheses about the data by exploring the partitioned network in real-time.

The user can select a specific node, as in Figure 4.10b, and examine all of its nearest neighbor correlations. The specific node selected in this figure was associated with the miRNA sequence labeled as 'hsa-miR-1260b'. Cytoscape provides the functionality to search external links for any public information associated with any selected nodes. These external links include public databases, including the National Center for Biotechnology

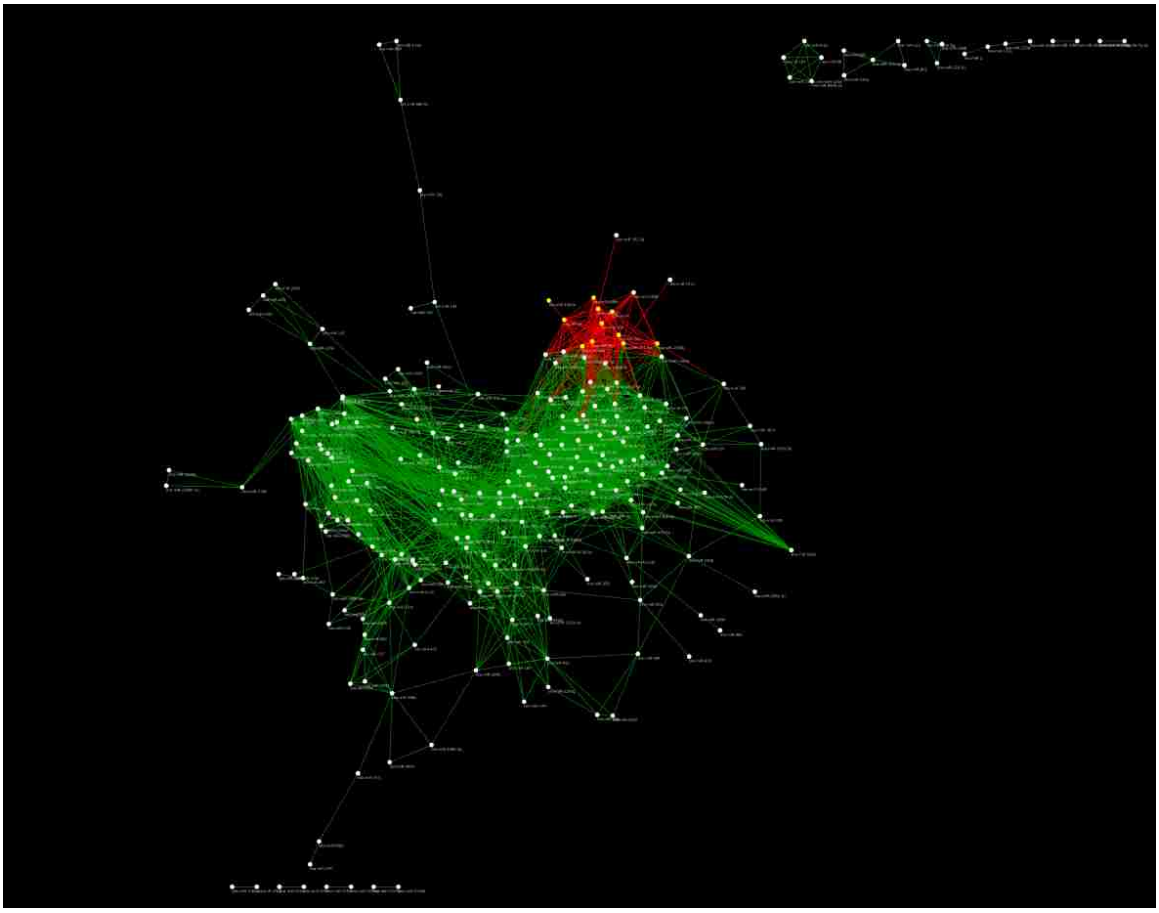
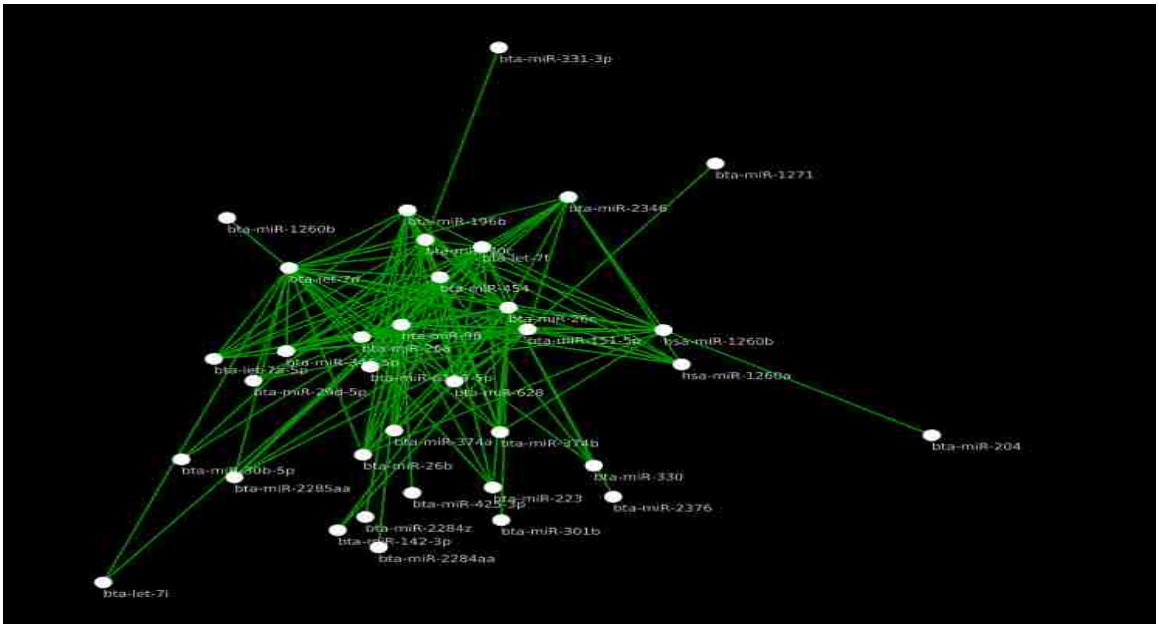
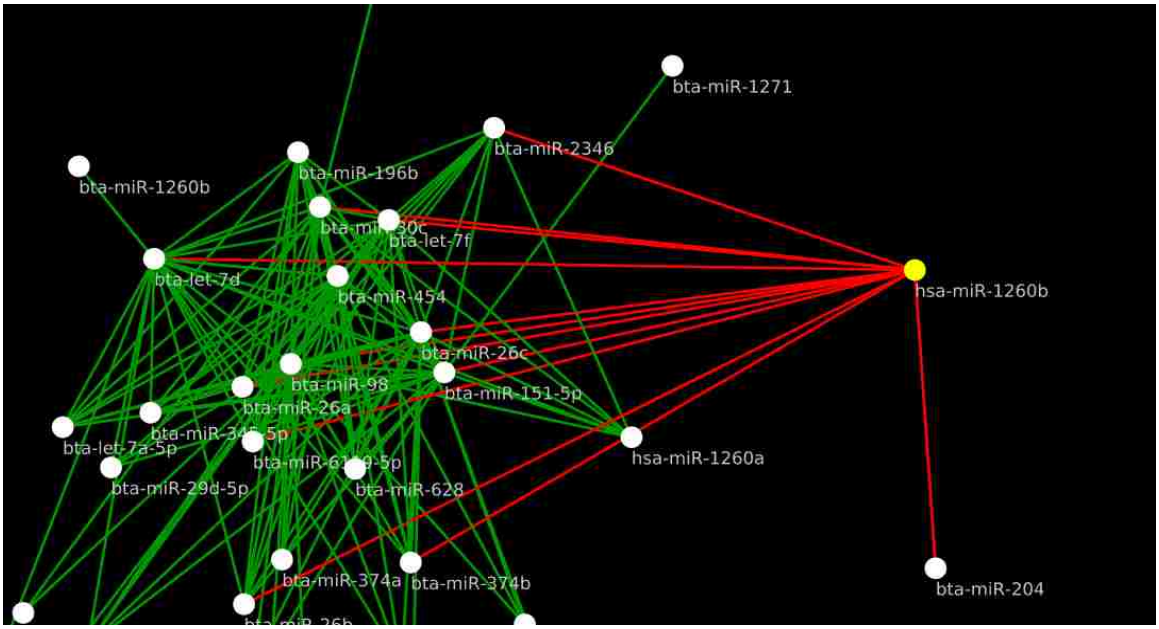


Figure 4.9. Selecting a random portion of the Pearson network from Figure 4.8 for heuristic exploration.

Information (NCBI), which is a platform on which advances in science can be made by providing access to biomedical and genomic information [70]. From NCBI, the target miRNA was searched for in the Gene Expression Omnibus (GEO), an international archive for high-throughput functional genomic data sets whose contributors are from the research community [6]. GEO provides tools that assist users to query and analyze their data. The user can utilize this archive to facilitate hypothesis generation about the functionality of a particular gene or miRNA, for example. From searching the GEO for this target miRNA 'hsa-miR-1260b', various studies were discovered involving this particular miRNA, for example, a study was published in 2012 on viral-miRNA in *Homo sapiens* in which the miRNA 'hsa-miR-1260b' played a significant role [79].



(a) Selected nodes and edges partitioned from Figure 4.9.



(b) Nearest neighbor correlations of selected node 'hsa-miR-1260b' highlighted in yellow.

Figure 4.10. Figures (a) and (b) illustrating the functionality of Cytoscape as a tool for network analysis.

The previous example is provided only to showcase the functionality Cytoscape provides. It is not necessarily the most appropriate approach one should take to gain an accurate biological interpretation of these specific data; an accurate biological interpretation

of these *Bos taurus* miRNA data is not in the scope of this thesis. The intent was to convey that Cytoscape supports the user during the discovery stage of his or her research by providing an easy access to all of the tools available that might be of assistance. It allows the user to harness the power of on-line resources, where scientists collaborate and share information about their findings. This example was provided as a motivation for the scientific capabilities of coupling the network pruning method RMT with the network visualization tool Cytoscape [73].

5. CONCLUSIONS

Random matrix theory (RMT) has been shown to be a reliable method for constructing meaningful networks in a variety of biological systems such as in maize, rice, and human beings [32]. It has also been shown to be an effective tool for predicting functions of unknown genes [52]. In this thesis, the efficacy of RMT being applied to other biological data was evaluated, namely miRNA expression data. Four different similarity metrics were applied to construct co-expression networks from an experiment involving *Bos taurus* miRNA data [43], which were publicly available at the time of this publication [21]. Then, the effect of using RMT as a pruning method was investigated. More specifically, the threshold of correlation was found for each network using the software RMTGeneNet [32] that, at the time of this publication, is currently available on a GitHub repository with an open source GNU GPLv2.0 license at https://github.com/spficklin/RMTGeneNet.

These miRNA data were chosen because miRNA expression data is closely associated with gene expression data, for which the RMTGeneNet software provides as a sample data set. Literature supports the use of co-expression networks as a tool for analyzing miRNA data [4, 31, 74, 93]. Also, this *Bos taurus* miRNA data set was of manageable size (< 1MB) for the personal computer on which the analysis were performed. For future work, a logical next step would be to apply these procedures to some larger data sets. This would further test the capabilities of RMT. The computational efficiency and applicability to other types of biological data, such as DNA methylation data, could be evaluated.

RMT was explored in this thesis as a network pruning tool because it is a way to objectively find a threshold of correlation in a network. Network analysis fills a crucial need in biology; however, to analyze the networks, they must be filtered to some extent.

This data filtering, or network pruning, process is necessary in order to remove spurious correlations from the data. The choice of a threshold value in which to eliminate the noise in a network greatly influences the resulting network.

The choice of a threshold value impacts the sensitivity and specificity of the node connections. Therefore, an omnipresent question encountered by scientists is what is the most appropriate threshold value to use. If the threshold value chosen is too high, then the computational complexity is lowered at the expense of losing node connections. The resulting network may therefore be too sparse and information could be lost. On the other hand, if the threshold value is too low, then information is not lost but the computational requirements are increased. Too many spurious correlations may appear which could mask the true signal from emerging. RMT provides a way to objectively choose the most appropriate threshold value.

The use of RMT was investigated by applying the RMT algorithm, incorporated in the RMTGeneNet software, to an open-source miRNA data set. Preliminary results and figures were provided to demonstrate the performance of RMT. The preliminary results are encouraging and leave room for much future work to be done. In the future, a more detailed analysis of the co-expression network topologies could be completed by applying some of the network analysis metrics discussed in Section 2.4. A further exploration of the capabilities that the Cytoscape software possesses could be incorporated into future work. Furthermore, applying the RMT algorithm to larger data sets, such as DNA methylation data where hundreds of thousands to millions of nodes are possible, would also be of interest to determine the efficacy of RMT on big data. It would also be advantageous to recruit persons with a strong background knowledge of the particular data set being examined. This would assist in providing a sound biological interpretation of the results.

APPENDIX A

R CODE

Wigner Semi-Circle Law Demonstration

```
# Set size of matrix
n <- 500;

# generate an (n by n) matrix with
# i.i.d entries distributed as N(0,1)
A <- array(rnorm(n^2), c(n,n));

# Standard symmetric Wigner matrix
wigMat <- (A + t(A)) / sqrt(2 * n);

# Calculate eigenvalues
lambda <- eigen(wigMat, symmetric = TRUE, only.values = T);
lambda <- lambda$values;

# Generate histogram
hist(lambda, breaks = 100,
     main = paste("Eigenvalue Distribution of a ",
                  n, " by ", n, '\n',
                  " Standard Wigner Matrix", sep = ""),
     cex.main = 2, xlab = "Eigenvalues", freq = FALSE)
```

APPENDIX B

PYTHON CODE

Similarity Metrics

```
#! /usr/bin/env python

import argparse
import os
from sklearn.neighbors import DistanceMetric
import numpy as np
import pandas as pd

#######################################################################
#######################################################################
def my_pearson(X, Y):
    num = sum((X - X.mean()) * (Y - Y.mean()))
    denom = np.sqrt(sum((X-X.mean())**2) * sum((Y - Y.mean())**2))
    r = num / denom
    return r

p_dist = DistanceMetric.get_metric(my_pearson).pairwise

# (start) Pearson Correlation Function
#######################################################################
# If the user does not specify the output location by using the
# flag -o then the file will be saved in the current directory
# with the default name of:
# the name of the input file + Pearson_cor.tsv
# Example, samp_df.tsv will be saved as samp_df_Pearson_cor.tsv
```

```
def Pearson(df, name = None, toFile = None):
    df = df.T
    P = df.corr(method = 'pearson')
    P = P.round(3)
    try:
        P.index = list(df.index)
        P.index.name = df.index.names
        P.columns = list(df.index)
    except:
        pass
    if toFile == None:
        P.to_csv(name, sep='\t')
        printStatement = """ The Pearson correlation metric
        was performed on the input file and the resulting
        correlation matrix will be saved in the current
        directory as: """ + name
        print(printStatement)


    else:
        P.to_csv(toFile, sep='\t')
        outFile = """The Pearson correlation metric was
        performed on the input file and the resulting
        correlation matrix will be saved as: """
        outFile += toFile
        print(outFile)
#################################################################
```

```
# End Pearson Correlation


# (start) Czekanowski Index
#####################################################################

def my_Czek(X,Y):
    X = np.array(X)
    Y = np.array(Y)


    Cz = 2 * sum(np.minimum(X, Y)) / sum(X + Y)
    return Cz

czek_dist = DistanceMetric.get_metric(my_Czek).pairwise

def Czekanowski(df, name = None, toFile = None):
    C = pd.DataFrame(czek_dist(df))
    C = C.round(3)
    try:
        C.index = list(df.index)
        C.index.name = df.index.names
        C.columns = list(df.index)
    except:
        pass
    if toFile == None:
        C.to_csv(name, sep='\t')
        printStatement = """ The Czekanowski correlation
        metric was performed on the input file and the
```

```
            resulting correlation matrix will be saved in
            the current diretory as : """ + name
            print(printStatement)
        else:
            C.to_csv(toFile, sep='\t')
            outFile = """ The Czekanowski correlation metric
            was performed on the input file and the resulting
            correlation matrix will be saved as : """ + toFile
            print(outFile)
    ##################################################################
    # End Czekanowski


    # (start) Spearman Correlation
    ##################################################################
    def Spearman(df, name = None, toFile = None):
        df = df.T
        S = df.corr(method='spearman')
        S = S.round(3)
        try:
            S.index = list(df.index)
            S.index.name = df.index.names
            S.columns = list(df.index)
        except:
            pass
        if toFile == None:
```

```
            S.to_csv(name, sep='\t')
            printStatement = """ Spearman correlation metric
            was performed on the input file and the resulting
            correlation matrix will be saved in the current
            directory as: """ + name
            print(printStatement)
        else:
            S.to_csv(toFile, sep='\t')
            printStatement =  """ The Spearman correlation
            metric was performed on the input file and the
            resulting correlation matrix will be
            saved as: """ + toFile

            print(printStatement)
#################################################################
# End Spearman


# Stringent Proportional Similarity (SPS)
#################################################################
def my_sps(X, Y):
    X = np.array(X)
    Y = np.array(Y)
    id0 = np.logical_and((X != 0) , (Y != 0))
    sps = 1. - (1. / len(X)) * ( sum(np.absolute(X[id0]**2 - \
        Y[id0]**2)/(X[id0]**2 + Y[id0]**2)) + sum(~id0) )
```

```
    return sps
sps_dist = DistanceMetric.get_metric(my_sps).pairwise


def SPS(df, name = None, toFile = None):
    Sp = pd.DataFrame(sps_dist(df))
    Sp = Sp.round(3)
    try:
        Sp.index = list(df.index)
        Sp.index.name = df.index.names
        Sp.columns = list(df.index)
    except:
        pass
    if toFile == None:
        Sp.to_csv(name,  sep='\t')
        printStatement = """ The SPS correlation metric
        was performed on the input file and the resulting
        correlation matrix will be saved in the current
        directory as: """ + name
        print(printStatement)
    else:
        Sp.to_csv(toFile, sep='\t')
        outFile = """ The SPS correlation metric was
        performed on the input file and the resulting
        correlation matrix will be saved as: """ + toFile
        print(outFile)
#################################################################
# End SPS
```

```
def Main():
    parser = argparse.ArgumentParser()

    parser.add_argument('inputFile', help= """The inputFile
    argument specifies which dataframe you are reading in
    to perform a correlation test on.""")

    parser.add_argument('CorFunc' , help= """The CorFunc
    argument specifies which correlation metric to
    perform on the input file. The options are:
    <Pearson>, <Spearman>, <Czekanowski>, or <SPS>.""" )

    parser.add_argument('-o' , '--output', help="""This
    is an optional argument that specifies where your
    output file (a correlation matrix) should be
    written to. Default is the current working
    directory.""", type=str)

    args = parser.parse_args()
    try:
        df = pd.read_csv(args.inputFile, sep='\t', \
                index_col=0, skipinitialspace=True)
    except OSError:
        errorStatement = "This input file: " +
        args.inputFile + """
```

```
            does not exist."""
            print(errorStatement)
    ##############################################################
        if args.CorFunc == 'Pearson':
            if args.output:
                toFile = args.output
                return Pearson(df = df, toFile = toFile)
            else:
                name = os.path.splitext(args.inputFile)[0] + \
                "_Pearson_cor.tsv"
                return Pearson(df = df, name = name)
    ##############################################################
        elif args.CorFunc == 'Czekanowski':
            if args.output:
                toFile = args.output
                return Czekanowski(df = df, toFile = toFile)
            else:
                name = os.path.splitext(args.inputFile)[0] + \
                "_Czekanowski_cor.tsv"
                return Czekanowski(df = df, name = name)
    ##############################################################


    ##############################################################
        elif args.CorFunc == 'Spearman':
            if args.output:
                toFile = args.output
                return Spearman(df = df, toFile = toFile)
```

```
        else:
            name = os.path.splitext(args.inputFile)[0] + \
                    "_Spearman_cor.tsv"
            return Spearman(df = df, name = name)
##################################################################


##################################################################
    elif args.CorFunc == 'SPS':
        if args.output:
            toFile = args.output
            return SPS(df = df, toFile = toFile)
        else:
            name = os.path.splitext(args.inputFile)[0] + \
                    "_SPS_cor.tsv"
            return SPS(df = df, name = name)
##################################################################


    else:
       print("Invalid entry for the argument specifying " + \
       "the correlation function. Please see help for " + \
       "this script by typing: <python ex1.py -h>.")

if __name__=='__main__':
    Main()
```

APPENDIX C

χ^2 TABLE

Chi Square Distribution Table							
d.f.	$\chi^2_{.25}$	$\chi^2_{.10}$	$\chi^2_{.05}$	$\chi^2_{.025}$	$\chi^2_{.010}$	$\chi^2_{.005}$	$\chi^2_{.001}$
1	1.32	2.71	3.84	5.02	6.63	7.88	10.8
2	2.77	4.61	5.99	7.38	9.21	10.6	13.8
3	4.11	6.25	7.81	9.35	11.3	12.8	16.3
4	5.39	7.78	9.49	11.1	13.3	14.9	18.5
5	6.63	9.24	11.1	12.8	15.1	16.7	20.5
6	7.84	10.6	12.6	14.4	16.8	18.5	22.5
7	9.04	12.0	14.1	16.0	18.5	20.3	24.3
8	10.2	13.4	15.5	17.5	20.1	22.0	26.1
9	11.4	14.7	16.9	19.0	21.7	23.6	27.9
10	12.5	16.0	18.3	20.5	23.2	25.2	29.6
11	13.7	17.3	19.7	21.9	24.7	26.8	31.3
12	14.8	18.5	21.0	23.3	26.2	28.3	32.9
13	16.0	19.8	22.4	24.7	27.7	29.8	34.5
14	17.1	21.1	23.7	26.1	29.1	31.3	36.1
15	18.2	22.3	25.0	27.5	30.6	32.8	37.7
16	19.4	23.5	26.3	28.8	32.0	34.3	39.3
17	20.5	24.8	27.6	30.2	33.4	35.7	40.8
18	21.6	26.0	28.9	31.5	34.8	37.2	42.3
19	22.7	27.2	30.1	32.9	36.2	38.6	32.8
20	23.8	28.4	31.4	34.2	37.6	40.0	45.3
21	24.9	29.6	32.7	35.5	38.9	41.4	46.8
22	26.0	30.8	33.9	36.8	40.3	42.8	48.3
23	27.1	32.0	35.2	38.1	41.6	44.2	49.7
24	28.2	33.2	36.4	39.4	32.0	45.6	51.2
25	29.3	34.4	37.7	40.6	44.3	46.9	52.6
26	30.4	35.6	38.9	41.9	45.6	48.3	54.1
27	31.5	36.7	40.1	43.2	47.0	49.6	55.5
28	32.6	37.9	41.3	44.5	48.3	51.0	56.9
29	33.7	39.1	42.6	45.7	49.6	52.3	58.3
30	34.8	40.3	43.8	47.0	50.9	53.7	59.7
40	45.6	51.8	55.8	59.3	63.7	66.8	73.4
50	56.3	63.2	67.5	71.4	76.2	79.5	86.7
60	67.0	74.4	79.1	83.3	88.4	92.0	99.6
70	77.6	85.5	90.5	95.0	100	104	112
80	88.1	96.6	102	107	112	116	125
90	98.6	108	113	118	124	128	137
100	109	118	124	130	136	140	149

Table C.1. A χ^2 table from the book *Statistics: Discovering Its Power* [92].

REFERENCES

[1] Lada A Adamic. *Network dynamics: The world wide web*. PhD thesis, Stanford University, 2001.

[2] Reka Albert. Scale-free networks in cell biology. *Journal of cell science*, 118(21):4947–4957, 2005.

[3] Victor Ambros. The functions of animal microRNAs. *Nature*, 431(7006):350–355, 2004.

[4] Sanghamitra Bandyopadhyay and Malay Bhattacharyya. Analyzing miRNA co-expression networks to explore TF-miRNA regulation. *BMC bioinformatics*, 10(1):163, 2009.

[5] Albert-Laszlo Barabasi and Zoltan N Oltvai. Network biology: understanding the cell's functional organization. *Nature reviews genetics*, 5(2):101–113, 2004.

[6] Tanya Barrett, Stephen E Wilhite, Pierre Ledoux, Carlos Evangelista, Irene F Kim, Maxim Tomashevsky, Kimberly A Marshall, Katherine H Phillippy, Patti M Sherman, Michelle Holko, et al. NCBI GEO: archive for functional genomics data sets–update. *Nucleic acids research*, 41(D1):D991–D995, 2013.

[7] Stephen A Bloom. Similarity indices in community studies: potential pitfalls. *Mar. Ecol. Prog. Ser*, 5(2):125–128, 1981.

[8] Gaëtan Borot. Introduction to random matrix theory. Universität Bonn, http://guests.mpim-bonn.mpg.de/gborot/files/RMT-15fev2015.pdf, 2015.

[9] Mark Buchanan. Enter the matrix. *New Scientist*, 206(2755):28–31, 2010.

[10] Atul J Butte and Isaac S Kohane. Mutual information relevance networks: functional genomic clustering using pairwise entropy measurements. In *Pac Symp Biocomput*, volume 5, pages 418–429, 2000.

[11] Yimei Cai, Xiaomin Yu, Songnian Hu, and Jun Yu. A brief review on the mechanisms of miRNA regulation. *Genomics, proteomics & bioinformatics*, 7(4):147–154, 2009.

[12] Peter J Carrington, John Scott, and Stanley Wasserman. *Models and methods in social network analysis*, volume 28. Cambridge University Press, 2005.

[13] Scott L Carter, Christian M Brechbühler, Michael Griffin, and Andrew T Bond. Gene co-expression network topology provides a framework for molecular characterization of cellular state. *Bioinformatics*, 20(14):2242–2250, 2004.

[14] Sung-Hyuk Cha. Comprehensive survey on distance/similarity measures between probability density functions. *City*, 1(2):1, 2007.

[15] Qingfeng Chen and Ming Chen. Editorial (Thematic Issue: Protein Systems Biology: Method, Regulation, and Network). *Current Protein and Peptide Science*, 15(6):519–521, 2014.

[16] W. J. Conover. *Practical nonparametric statistics*. Academic Internet Publishers, 3rd edition, 2007.

[17] Damian Conway and Robert Resnick. *Quantum Physics of Atoms, Molecules, Solids, Nuclei, and Particles*. John Wiley & Sons, New York, USA, 1974.

[18] Jan Czekanowski. Die anthropologisch-ethnographischen arbeiten der expedition sh des herzogs adolf friedrich zu mecklenburg für den zeitraum vom 1. juni 1907 bis 1. August 1908. *Zeitschrift für Ethnologie*, 41(H. 5):591–615, 1909.

[19] Jan Czekanowski. Zarys metod statystycznych: w zastosowaniu do antropologii [an outline of statistical methods applied in anthropology]. *Prace Towarzystwa Naukowego Warszawskiego. 3, Wydział Nauk Matematycznych i Przyrodniczych*, 1913.

[20] Eric H Davidson, David R McClay, and Leroy Hood. Regulatory gene networks and the properties of the developmental process. *Proceedings of the National Academy of Sciences*, 100(4):1475–1480, 2003.

[21] F X Donadeu and Jason Ioannidis. E-geod-85090-early pregnancy-biomarkers. http://www.ebi.ac.uk/arrayexpress/experiments/E-GEOD-85090/, 2016.

[22] Jun Dong and Steve Horvath. Understanding network concepts in modules. *BMC systems biology*, 1(1):1, 2007.

[23] Richard O Duda, Peter E Hart, and David G Stork. *Pattern classification*. John Wiley & Sons, 2012.

[24] Alan Edelman and Per-Olof Persson. Numerical methods for eigenvalue distributions of random matrices. *arXiv preprint math-ph/0501068*, 2005.

[25] Jianqing Fan, Fang Han, and Han Liu. Challenges of big data analysis. *National science review*, 1(2):293–314, 2014.

[26] Stephen Ficklin. KINC.R. https://github.com/SystemsGenetics/KINC.R, 2016.

[27] Jessica Folliett. The importance of big data and data visualization. Dataversity, http://www.dataversity.net/the-importance-of-big-data-and-data-visualization/, January 2016.

[28] Andrea Franceschini, Damian Szklarczyk, Sune Frankild, Michael Kuhn, Milan Simonovic, Alexander Roth, Jianyi Lin, Pablo Minguez, Peer Bork, Christian Von Mering, et al. String v9. 1: protein-protein interaction networks, with increased coverage and integration. *Nucleic acids research*, 41(D1):D808–D815, 2013.

[29] E Gallagher. COMPAH documentation. *User's Guide and application published at: http://www.es.umb.edu/edgwebp.htm*, 1999.

[30] Francis Galton. Regression towards mediocrity in hereditary stature. *The Journal of the Anthropological Institute of Great Britain and Ireland*, 15:246–263, 1886.

[31] Vincenzo Alessandro Gennarino, Giovanni D'Angelo, Gopuraja Dharmalingam, Serena Fernandez, Giorgio Russolillo, Remo Sanges, Margherita Mutarelli, Vincenzo Belcastro, Andrea Ballabio, Pasquale Verde, et al. Identification of microRNA-regulated gene networks by expression analysis of target genes. *Genome research*, 22(6):1163–1172, 2012.

[32] Scott M Gibson, Stephen P Ficklin, Sven Isaacson, Feng Luo, Frank A Feltus, and Melissa C Smith. Massive-scale gene co-expression network construction and robustness testing using random matrix theory. *PloS one*, 8(2):e55871, 2013.

[33] Thomas Guhr, Axel Müller-Groeling, and Hans A Weidenmüller. Random-matrix theories in quantum physics: common concepts. *Physics Reports*, 299(4):189–425, 1998.

[34] Frank Avery Haight. *Handbook of the Poisson distribution*. Wiley, 1967.

[35] Manfred Hanke. Spectral statistics: The form factor in semiclassical theory and numerical analysis, March 2006.

[36] Leland H Hartwell, John J Hopfield, Stanislas Leibler, and Andrew W Murray. From molecular to modular cell biology. *Nature*, 402:C47–C52, 1999.

[37] Lin He and Gregory J Hannon. MicroRNAs: small RNAs with a big role in gene regulation. *Nature Reviews Genetics*, 5(7):522–531, 2004.

[38] Bentolhoda Helmi and Adel Torkaman Rahmani. Estimation of distribution algorithm using factor graph and markov blanket canonical factorization. In *Proceedings of the 2014 Annual Conference on Genetic and Evolutionary Computation*, GECCO '14, pages 349–356, New York, NY, USA, 2014. ACM.

[39] Steve Horvath and Jun Dong. Geometric interpretation of gene coexpression network analysis. *PLoS comput biol*, 4(8):e1000117, 2008.

[40] Albert Hsiao and Michael D Kuo. High-throughput biology in the postgenomic era. *Journal of vascular and interventional radiology*, 17(7):1077–1085, 2006.

[41] Wolfgang Huber, Vincent J Carey, Li Long, Seth Falcon, and Robert Gentleman. Graphs in molecular biology. *BMC bioinformatics*, 8(6):1, 2007.

[42] Chris P Hughes, Jon P Keating, et al. Random matrix theory and the derivative of the Riemann zeta function. In *Proceedings of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, volume 456, pages 2611–2627. The Royal Society, 2000.

[43] Jason Ioannidis and F Xavier Donadeu. Circulating miRNA signatures of early pregnancy in cattle. *BMC genomics*, 17(1):184, 2016.

[44] Alan J Izenman. Introduction to random-matrix theory, 2008.

[45] Hawoong Jeong, Sean P Mason, A-L Barabási, and Zoltan N Oltvai. Lethality and centrality in protein networks. *Nature*, 411(6833):41–42, 2001.

[46] Todd Kemp. Math 247a: Introduction to random matrix theory. University of California, San Diego, 2013.

[47] Hiroaki Kitano. Systems biology: a brief overview. *Science*, 295(5560):1662–1664, 2002.

[48] Laurent Laloux, Pierre Cizeau, Marc Potters, and Jean-Philippe Bouchaud. Random matrix theory and financial correlations. *International Journal of Theoretical and Applied Finance*, 3(03):391–397, 2000.

[49] Vito Latora and Massimo Marchiori. Efficient behavior of small-world networks. *Physical review letters*, 87(19):198701, 2001.

[50] Joseph Lee Rodgers and W Alan Nicewander. Thirteen ways to look at the correlation coefficient. *The American Statistician*, 42(1):59–66, 1988.

[51] Feng Luo, Pradip K Srimani, and Jizhong Zhou. Application of random matrix theory to analyze biological data. In *Handbook of Data Intensive Computing*, pages 711–732. Springer, 2011.

[52] Feng Luo, Yunfeng Yang, Jianxin Zhong, Haichun Gao, Latifur Khan, Dorothea K Thompson, and Jizhong Zhou. Constructing gene co-expression networks and predicting functions of unknown genes by random matrix theory. *BMC bioinformatics*, 8(1):1, 2007.

[53] Wes McKinney. Data structures for statistical computing in python. In Stéfan van der Walt and Jarrod Millman, editors, *Proceedings of the 9th Python in Science Conference*, pages 51 – 56, 2010.

[54] Uwe Menzel and Maintainer Uwe Menzel. *RMThreshold*, 2016.

[55] Steven J. Miller and Ramin Takloo-bighash. Introduction to random matrix theory from an invitation to modern number theory. https://web.williams.edu/Mathematics/sjmiller/public_html/416/currentnotes/IntroRMT_Math54.pdf, 2007.

[56] Mark Newman. *Networks: an introduction*. Oxford university press, 2010.

[57] Karl Pearson. Note on regression and inheritance in the case of two parents. *Proceedings of the Royal Society of London*, 58:240–242, 1895.

[58] Karl Pearson. X. on the criterion that a given system of deviations from the probable in the case of a correlated system of variables is such that it can be reasonably supposed to have arisen from random sampling. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, 50(302):157–175, 1900.

[59] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[60] Seth Pettie and Vijaya Ramachandran. An optimal minimum spanning tree algorithm. *Journal of the ACM (JACM)*, 49(1):16–34, 2002.

[61] Joaquim Pinto da Costa and Carlos Soares. A weighted rank measure of correlation. *Australian & New Zealand Journal of Statistics*, 47(4):515–529, 2005.

[62] Siméon Denis Poisson and Christian Heinrich Schnuse. *Recherches sur la probabilité des jugements en matière criminelle et en matière civile*. Meyer, 1841.

[63] Stephen R Proulx, Daniel EL Promislow, and Patrick C Phillips. Network thinking in ecology and evolution. *Trends in Ecology & Evolution*, 20(6):345–353, 2005.

[64] Nataša Pržulj and Noël Malod-Dognin. Network analytics in the age of big data. *Science*, 353(6295):123–124, 2016.

[65] Richard E Quandt. Some basic matrix theorems. Princeton University. http://www.quandt.com/papers/basicmatrixtheorems.pdf.

[66] R Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2016.

[67] Erzsébet Ravasz, Anna Lisa Somera, Dale A Mongru, Zoltán N Oltvai, and A-L Barabási. Hierarchical organization of modularity in metabolic networks. *science*, 297(5586):1551–1555, 2002.

[68] Jaap C Reijneveld, Sophie C Ponten, Henk W Berendse, and Cornelis J Stam. The application of graph theoretical analysis to complex networks in the brain. *Clinical Neurophysiology*, 118(11):2317–2331, 2007.

[69] Jean-François Rual, Kavitha Venkatesan, Tong Hao, Tomoko Hirozane-Kishikawa, Amélie Dricot, Ning Li, Gabriel F Berriz, Francis D Gibbons, Matija Dreze, Nono Ayivi-Guedehoussou, et al. Towards a proteome-scale map of the human protein–protein interaction network. *Nature*, 437(7062):1173–1178, 2005.

[70] Eric W Sayers, Tanya Barrett, Dennis A Benson, Evan Bolton, Stephen H Bryant, Kathi Canese, Vyacheslav Chetvernin, Deanna M Church, Michael DiCuccio, Scott Federhen, et al. Database resources of the national center for biotechnology information. *Nucleic acids research*, 39(suppl 1):D38–D51, 2011.

[71] Sebastian Schierenberg, Falk Bruckmann, and Tilo Wettig. Wigner surmise for mixed symmetry classes in random matrix theory. *Physical Review E*, 85(6):061130, 2012.

[72] Mathabatha Evodia Setati, Daniel Jacobson, Ursula-Claire Andong, and Florian Bauer. The vineyard yeast microbiome, a mixed model microbial map. *PLoS One*, 7(12):e52609, 2012.

[73] Paul Shannon, Andrew Markiel, Owen Ozier, Nitin S Baliga, Jonathan T Wang, Daniel Ramage, Nada Amin, Benno Schwikowski, and Trey Ideker. Cytoscape: a software environment for integrated models of biomolecular interaction networks. *Genome research*, 13(11):2498–2504, 2003.

[74] Neil R Smalheiser, Giovanni Lugli, Hooriyah S Rizavi, Vetle I Torvik, Gustavo Turecki, and Yogesh Dwivedi. MicroRNA expression is down-regulated and reorganized in prefrontal cortex of depressed suicide subjects. *PloS one*, 7(3):e33201, 2012.

[75] Tom AB Snijders. The degree variance: an index of graph heterogeneity. *Social networks*, 3(3):163–174, 1981.

[76] Thorvald Sørensen. A method of establishing groups of equal amplitude in plant sociology based on similarity of species and its application to analyses of the vegetation on danish commons. *Biol. Skr.*, 5:1–34, 1948.

[77] K Soundararajan. Small gaps between prime numbers: The work of goldston-pintz-yildirim. *Bulletin of the American Mathematical Society*, 44(1):1–18, 2007.

[78] Charles Spearman. The proof and measurement of association between two things. *The American journal of psychology*, 15(1):72–101, 1904.

[79] C Stoecklin-Wasmer, P Guarnieri, R Celenti, RT Demmer, M Kebschull, and PN Papapanou. Micrornas and their target genes in gingival tissues. *Journal of dental research*, 91(10):934–940, 2012.

[80] Joshua M Stuart, Eran Segal, Daphne Koller, and Stuart K Kim. A gene-coexpression network for global discovery of conserved genetic modules. *science*, 302(5643):249–255, 2003.

[81] Richard J Trudeau. *Introduction to graph theory*. Courier Corporation, 2013.

[82] Antonia M Tulino and Sergio Verdú. *Random matrix theory and wireless communications*, volume 1. Now Publishers Inc, 2004.

[83] Stéfan van der Walt, S Chris Colbert, and Gael Varoquaux. The numpy array: a structure for efficient numerical computation. *Computing in Science & Engineering*, 13(2):22–30, 2011.

[84] Duncan J Watts and Steven H Strogatz. Collective dynamics of 'small-world' networks. *nature*, 393(6684):440–442, 1998.

[85] D Weighill and D Jacobson. Network metamodeling: Effect of correlation metric choice on phylogenomic and transcriptomic network topology. In *Advances in Biochemical Engineering/Biotechnology*, pages 1–41. Springer Berlin Heidelberg, Berlin, Heidelberg, 2016.

[86] Eugene P Wigner. On the statistical distribution of the widths and spacings of nuclear resonance levels. In *Mathematical Proceedings of the Cambridge Philosophical Society*, volume 47, pages 790–798. Cambridge Univ Press, 1951.

[87] Eugene P Wigner. Characteristic vectors of bordered matrices with infinite dimensions. *Annals of Mathematics*, 62(3):548–564, 1955.

[88] Eugene P Wigner. Random matrices in physics. *SIAM review*, 9(1):1–23, 1967.

[89] J Wishart. The generalized product moment distribution in samples from a normal multivariate population, biometrika, a20, 32-52. *Anwar H Joarder Department of Mathematical Sciences King Fahd University of Petroleum and Minerals Dhahran*, 31261, 1928.

[90] Natalie Wolchover. In Mysterious Pattern, Math and Nature Converge. *Quanta Magazine*, 2013.

[91] Cecily J Wolfe, Isaac S Kohane, and Atul J Butte. Systematic survey reveals general applicability of "guilt-by-association" within gene coexpression networks. *BMC bioinformatics*, 6(1):1, 2005.

[92] Ronald J Wonnacott and Thomas H Wonnacott. *Statistics: Discovering Its Power*, volume 1 of *Probability & Mathematical Statistics*. John Wiley & Sons, 1982.

[93] Juan Xu, Chuan-Xing Li, Yong-Sheng Li, Jun-Ying Lv, Ye Ma, Ting-Ting Shao, Liang-De Xu, Ying-Ying Wang, Lei Du, Yun-Peng Zhang, et al. MiRNA–miRNA synergistic network: construction via co-regulating functional modules and disease miRNA topological features. *Nucleic acids research*, 39(3):825–836, 2011.

[94] Paul M Yoshioka. Misidentification of the bray-curtis similarity index. *Marine Ecology Progress Series*, 368:309–310, 2008.

[95] Bin Zhang, Steve Horvath, et al. A general framework for weighted gene co-expression network analysis. *Statistical applications in genetics and molecular biology*, 4(1):1128, 2005.

[96] JX Zhong and T Geisel. Level fluctuations in quantum systems with multifractal eigenstates. *Physical Review E*, 59(4):4071, 1999.

[97] JX Zhong, U Grimm, Rudolf A Roemer, and M Schreiber. Level-spacing distributions of planar quasiperiodic tight-binding models. *Physical review letters*, 80(18):3996, 1998.

VITA

Jesse Aaron Marks was born in Springfield, Missouri, in the year 1991. In December 2014, he earned a Bachelor of Science in Mathematics (*summa cum laude*) from Central Methodist University (CMU) in Fayette, Missouri. At CMU, Jesse was employed as a mathematics and logic tutor and also completed an honors research project that investigated a special class of spirals called Euler spirals. In the summer of 2014, he participated in a research experience for undergraduates (REU) opportunity at the Gulf Coast Research Laboratory in Ocean Springs, Mississippi. During the REU, he designed a simulation model for an alternative biological sampling procedure called adaptive cluster sampling, and presented his work at multiple venues. After graduating from CMU, he completed a Science Undergraduate Laboratory Internship (SULI) at the National Renewable Energy Laboratory in Golden, Colorado. As a SULI intern, Jesse worked in the Power Systems Engineering Center on modeling temperature dynamics and the power consumption of heating, ventilation, and air-conditioning systems in simulated residential homes. In August 2015, he began pursuing a graduate degree from Missouri University of Science and Technology (S&T) in Rolla, Missouri. After completing two semesters of graduate study, Jesse was accepted into the Higher Education Research Experiences (HERE) Program at the Oak Ridge National Laboratory in Oak Ridge, Tennessee. At ORNL, he worked in the Biosciences Division with the Computational Biology and Bioinformatics group. The HERE program allowed Jesse to complement his academic program at S&T by utilizing the unique resources at ORNL and enhance his mathematics background, while conducting his master's thesis research. Jesse returned for his final semester of graduate studies at S&T, where he earned a Master of Science in Applied Mathematics in May 2017.