



Cybersecurity Anomaly Detection Using Graph Vertex Degree Assortativity

Daniel Juda

Two fundamental questions in cybersecurity are what is malicious activity and how do we detect it. These questions are generally answered using one of two paradigms. The first is signature-based threat detection, where known vulnerabilities are scanned using some form of detection software. The second is broadly termed anomaly detection, where analysts simply look for anything on the network that is considered atypical. We can distill this process down to asking the following: given network traffic during a fixed time period, can we determine if the character of the traffic is anomalous relative to some historical norm? Due to the constant change inherent in modern networks, establishing a historical norm can be difficult. Moreover, this constant change leads to many techniques highlighting behavior that is anomalous, but not actually malicious. There is a good deal of literature on these topics, too much to adequately summarize here, so we reference only the papers we borrowed techniques from. Our goal is to analyze internal netflow data from an enterprise network and identify subtle changes which may indicate malicious activity.

[Photo credit: iStock.com/carloscastilla]

Intuitively, network flow traffic lends itself to being encoded as a graph. This is formalized by Iliofotou, et al. in [1], where the authors describe a method for analyzing network traffic by considering directed graphs which they call traffic dispersion graphs (TDGs). These are graphs which consist of taking all nodes and edges in a network and then removing edges using some filtration criteria. We leverage this technique to study protocols independently. In particular, we look at well-understood protocols and analyze the TDG associated to each protocol using a well-known graph theoretic connectivity measure.

There are numerous techniques in graph theory to measure connectivity and communities. We utilize degree assortativity which was first introduced by Newman in [2, 3] and has since been used in a variety of fields. We use vertex assortativity on TDGs to identify days where subnetworks of an enterprise network are behaving atypically. We then use classical techniques from time series analysis to further investigate this behavior. The resulting combination of techniques provides a new method for analyzing network data and tipping of anomalies to cybersecurity threat analysts.

In this article, we first provide the necessary graph theory background for our technique, including an in-depth discussion of vertex assortativity. Next we discuss the necessary network theory background for our testing. Subsequent to that, we describe our testing and results, and we outline some ideas for using time series to drill down on detected anomalies. In the final section we offer conclusions. Suggestions for future directions are interspersed throughout the paper.

Graph theory background

A graph $G = (V, E)$ is a pair of sets, whose elements are called vertices and edges respectively, such that

$e \in E$ is a two element subset of V . If G is a directed graph, then edges are ordered pairs rather than unordered subsets of V . For each vertex $v \in V$ we call the values

$$d^-(v) = |\{e \in E \mid e = (v, x), x \in V\}| \text{ and} \\ d^+(v) = |\{e \in E \mid e = (x, v) \in E, x \in V\}|$$

the out-degree and in-degree of v respectively. The out-degree counts the number of edges originating at v and the in-degree counts the number of edges terminating at v . For each vertex $v \in V$, the total degree of v is $d(v) = d^-(v) + d^+(v)$, the count of

edges either originating or terminating at v . When G is an undirected graph we use $d(v)$ to denote the degree of v . For general background on graphs see, for example, [4] for undirected graphs or [5] for directed graphs.

For a graph $G = (V, E)$, assortativity or the assortativity coefficient is a measure of the preference for the vertices of G to connect to other vertices that are similar under a chosen measure. The assortativity coefficient for G is a value $r \in [-1, 1]$. If G has $r = -1$, then we say G is completely disassortative and if $r = 1$, then we say G is assortative. It was first defined by Newman in [2, 3] and has since been applied in a wide variety of fields such as biology [6] and social networking [7]. An extensive survey of assortativity results can be found in [8]. Throughout the paper, we will use the various degrees introduced above for our measure of similarity.

Let G be an undirected graph and p_k be the probability that a randomly sampled vertex from G has degree k , that is, $p_k = \frac{1}{|V|} |\{v \in V \mid d(v) = k\}|$. Consider the distribution $q_k \propto p_{k+1}$ given by $q_k = \frac{(k+1)p_{k+1}}{\sum_j j p_j}$. This weighted distribution associates to k the number of times a vertex of degree $k+1$ appears as an endpoint of an edge and is referred to as the remaining degree distribution. Intuitively, for an endpoint, v , of an edge, e , the remaining degree counts the number of edges incident to v other than e and is given by $d(v) - 1$. Let \bar{d} and s be the sample mean and sample standard deviation of the remaining degrees of the vertices. Let $f: E \rightarrow \mathbb{R}$ be defined as

$$f(e) = \frac{((d(v_i)-1)-\bar{d})((d(v_j)-1)-\bar{d})}{s^2}$$

where $e = (v_i, v_j)$. The assortativity coefficient of G is defined as

$$r = r(G) = \frac{1}{|E|} \sum_{e \in E} f(e)$$

The reader who is familiar with statistics will recognize this as a special case of the Pearson correlation coefficient of the degrees of the endpoints of a randomly sampled edge. If $s = 0$, then we define $r = 1$. This matches our intuition based on the definition, since if the standard deviation of the degrees is zero, then all edges connect to vertices that have the same degree.

The local assortativity at v , defined in [6, 9, 10] and denoted $\rho(v)$, is the amount v contributes

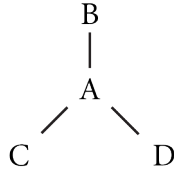


FIGURE 1. Example of a disassortative undirected graph. We have $d(A) = 3$ and $d(B) = d(C) = d(D) = 1$. This gives $q_0 = q_2 = \frac{1}{2}$, $\bar{d} = \frac{0+2}{2} = 1$, $s^2 = \frac{(0-1)^2 + (2-1)^2}{1} = 1$, and therefore $r = 1$. We also have $\rho(B) = \rho(C) = \rho(D) = -\frac{1}{6}$.

to the global assortativity. Given this definition, $r = \sum_{v \in V} \rho(v)$. For $v \in V$, we denote by $N(v)$, the set containing v and all vertices adjacent to v called the neighborhood of v . To compute the local assortativity at v , we consider $H = (V_H = N(v), E_H) = G|_{N(v)}$, the induced subgraph on G restricted to the neighborhood of v . Using the notation above, we have

$$\rho(v) = r|_{N(v)} = \frac{\frac{1}{|E|} \sum_{e \in E_H} f(e)}{2}$$

If, that is, the variance of the scaled vertex degree distribution is, then we define $\rho(v) = \frac{d(v)}{2|E|}$.

Let G be a directed graph. In [7], the authors extend the notion of assortativity to G by defining four different assortativity coefficients associated to G . Recall, for a vertex v in G , we have three notions of degree: in-degree, out-degree, and total degree. Also recall, that the purpose of assortativity is to measure the preference of vertices to connect to vertices that are similar. For undirected graphs, we used degree (or total degree) to measure similarity. In the case of a directed graph, we can choose which type of degree we want to use to measure similarity.

Let $V_i \subseteq V$ be the set of initial vertices and $V_t \subseteq V$ be the set of terminal vertices. Let $p_{k,+}^i$ be the probability that a randomly sampled vertex from V^i has in-degree k , that is, $p_{k,+}^i = \frac{1}{|V^i|} |\{v \in V^i \mid d_+(v) = k\}|$. Notice, if v is the initial vertex of an edge e , then e contributes to the out-degree of v , not the in-degree. Thus we let \bar{d}_i and s_i be the sample mean and sample standard deviation of the in-degree. Thus we let \bar{d}_+ and s_+ be the sample mean and sample standard deviation of the in-degree of the initial vertices respectively. On the other hand, if v is the terminal vertex of an edge e , then e does contribute to the in-degree of v . Thus, for the terminal vertices we again consider the distribution $q_{k,+}^t \propto p_{k+1,+}^t$ given

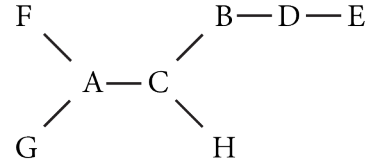


FIGURE 2. Example of an undirected graph. We have $d(A) = d(C) = 3$, $d(B) = d(D) = 2$, and $d(E) = d(F) = d(G) = 1$. This gives $q_0 = q_1 = q_2 = \frac{1}{3}$, $\bar{d} = 1$, $s_2 = 1$, and therefore $r = -\frac{2}{7}$.

by $q_{k,+}^t = \frac{(k+1)p_{k+1,+}^t}{\sum_j j p_{j,+}^t}$. Let \bar{d}_+ and s_+ be the sample mean and sample standard deviation of the remaining in-degree of the terminal vertices respectively. Let $f_{+,+}: E \rightarrow \mathbb{R}$ be defined as

$$f_{+,+}(e) = \frac{(d^+(v_i) - \bar{d}_+)((d^+(v_j) - 1) - \bar{d}_+)}{s_+^i s_+^t}$$

where $e = (v_i, v_j)$. Note, for the terminal vertices we account for the edge contributing to the in-degree and for the initial vertices, we do not. The in-in vertex degree assortativity coefficient of G is defined as

$$r_{+,+} = r_{+,+}(G) = \frac{1}{|E|} \sum_{e \in E} f_{+,+}(e)$$

The reader who is familiar with statistics will again recognize this as a special case of the Pearson correlation coefficient of the in-degrees of the endpoints of a randomly sampled edge. Similar definitions yield the in-out ($r_{+,-}$), out-in ($r_{-,+}$), and out-out ($r_{-,-}$) vertex degree assortativities. Suppose that $s_*^i = s_*^t$. If $\bar{d}_*^i = \bar{d}_*^t$, then we set $r_{*,*} = 1$. If the sample means don't agree, then we set $r_{*,*} = -1$. This again matches with our intuition based on the original definition. Intuitively, if there is no sample standard deviation and the sample means match, we have perfect correlation between initial and terminal vertices. On the other hand, if there is no sample standard deviation and the sample means don't match, we have perfect anticorrelation between initial and terminal vertices. Finally, if only one of s_*^i and s_*^t is 0, then we define $r_{*,*} = 0$.

Local assortativity was extended to directed graphs and discussed by Piraveenan, et al. in [6, 10]. It is once again defined, in each of the four different cases of directed assortativity, to be the contribution a vertex v makes to the global value. Thus, for example, the local in-in assortativity of a vertex v is given by $\rho_{+,+}(v) = r_{+,+}|_{N(v)} = \frac{\frac{1}{|E|} \sum_{e \in E_H} f_{+,+}(e)}{2}$, where $H = (V_H = N(v), E_H) = G|_{N(v)}$ as before. The other notions

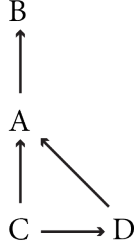


FIGURE 3. Example of a directed graph with in-in

vertex degree assortativity $r_{+,+} = -\frac{\sqrt{2}}{8}$. We have $d^{+,+}(C)=0, d^+(B)=d^+(D)=1, d^+(A)=2$. For the initial vertices, we have $V^i=\{A,C,D\}, p_{0,+}^i = p_{1,+}^i = p_{2,+}^i = \frac{1}{3}, \bar{d}_i^+ = \frac{0+1+2}{3} = 1$ and $S_i^+ = \frac{(0-1)^2+(1-1)^2+(2-1)^2}{2} = \frac{1}{2}$. For the terminal vertices, we have $V^t=\{A,B,D\}, q_{0,+}^t = q_{1,+}^t = \frac{1}{2}, \bar{d}_t^+ = \frac{1}{2}$ and $S_t^+ = \frac{1}{\sqrt{2}}$.

of directed local assortativity are defined similarly. Once again, if $s^i = s^t = 0$, then we define $\rho_{+,+}(v) = \frac{d^+(v)}{|E|}$ when $\bar{d}_*^i = \bar{d}_*^t$ and $\rho_{*,+}(v) = \frac{d^+(v)}{|E|}$ when $\bar{d}_*^i \neq \bar{d}_*^t$.

We close this section with some remarks on assortativity. It is generally observed that for large disassortative networks whose degree distribution follows a power law, such as an enterprise network or the Internet, the assortativity value decreases, that is, tends to 0. This is true in both the undirected and directed case. An alternative based on Spearman's Rho was proposed in [11, 12], but it is significantly more computationally intensive. We chose to limit the size of our graphs so that asymptotic behavior was not a concern and to avoid the additional computational complexity that is caused by using Spearman's Rho.

Cybersecurity background

Our goal is to apply graph theory to network flow (netflow) data to enable anomaly detection. Netflow data consists of metadata associated to transmission control protocol/Internet protocol (TCP/IP) connections. At the most basic level, a TCP/IP connection is a series of packets exchanged by the participants. In netflow data analysis, packets associated to a single communication are often combined into sessions. In particular, for our application a TCP/IP session between two hosts is a series of communications, such as a handshake followed by the transfer of desired information.

In order to use the graph theoretic techniques developed in the previous section, we need a

well-defined method for transferring our netflow data to a graph. A traffic dispersion graph (TDG) as described in [1], is a graph, built from netflow data, with edges limited to those meeting some desired criteria. Formally, let G represent a set of network traffic where each vertex corresponds to an address^a and each edge is a connection from a source address to a destination address. In particular, the edges are directed and G is a directed graph. We define a boolean function $f:E \rightarrow \{0,1\}$ for determining if an edge satisfies the chosen criteria. Our TDG is the subgraph $H=(V,E(H)) \subseteq G$ such that $E(H)=\{e \in E \mid f(e)=1\}$. Thus, f functions as a filter for which edges are admitted to the subgraph H . In our application, we will filter the edges based on protocol using the standard TCP/IP port addressing scheme.

As an example, let's examine a protocol of interest: lightweight directory access protocol (LDAP). This protocol is a fundamental directory look-up protocol within a network used for things such as verifying a login name and password before allowing network access. In this example, f is defined piecewise as 1 if an edge represents a connection under the LDAP protocol and 0 otherwise. Thus $E(H)$ is the set of all edges that are a connection from a source IP address to a destination IP address under the LDAP protocol.

Within this protocol, a host (IP address) acts as either a server or a client. Typically, there is no peer-to-peer communication between clients and therefore all traffic on clients should be either to or from a server. Servers communicate to each other only when there is a need to utilize secondary servers due to traffic volume; that is, when a primary server receives too many access requests to handle, it will begin pushing excess access requests over to a secondary server. This will appear as "peer-to-peer" traffic between servers and may be an indicator of either a network malfunction or an attack such as a denial-of-service attack. Moreover, this will affect the assortativity value which we expect to be close to -1 in general for this TDG.

Application to the LDAP protocol

For our application, we consider netflow traffic from an enterprise network. Throughout this section, all graphs are directed as described in the graph theory background discussion. We made several choices to limit the amount of data being used to avoid the issues previously discussed. We limited our graphs

a. Throughout we will be using network layer addresses, that is, IP addresses. It is also reasonable to use physical layer Media Access Control (MAC) addresses.

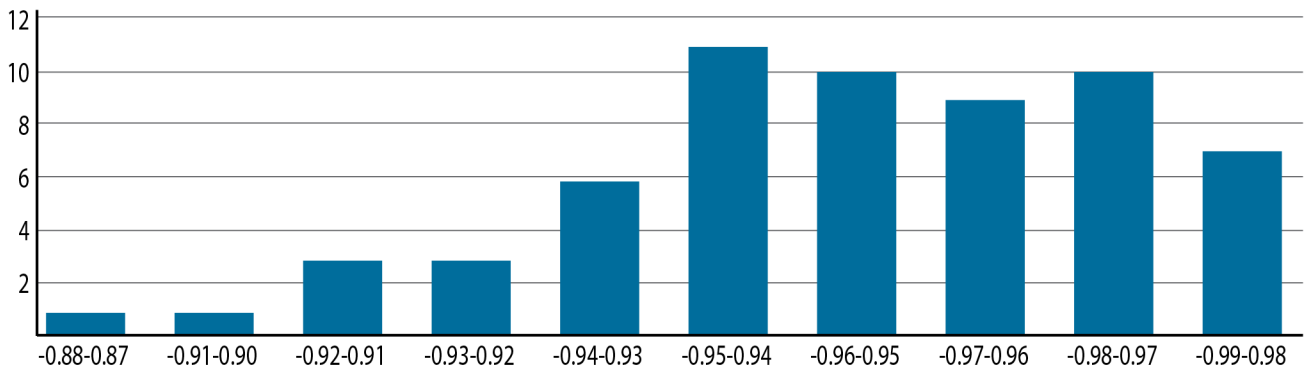


FIGURE 4. Example histogram of assortativity values built from 60 days of netflow using the LDAP protocol. Note the potential outliers in the $[-0.87, -0.88]$ bin.

to subnets with the first three IP octets in common (commonly referred to as /24 subnets). We generate an edge between two hosts when there is a session between them, that is, we will not add edges for each packet in the session. We also only consider unique address pairings; that is, if a host initiates communication multiple times to the same destination, we only allow one edge between them in each direction. As an example, consider two IP addresses, A and B . We add two vertices A and B to our graph. If A initiates a session with B , then we add an edge (A, B) to our graph. If A later initiates another session with B , we do not add an edge. On the other hand, if B initiates a session with A , we add an edge (B, A) .

Thus our workflow is the following. Fix a time-window from which to examine traffic and build TDGs, filtering based on LDAP, to represent each of several days. For each TDG, compute the directed assortativity coefficients. Given the four sets of coefficients, build a histogram to examine the distribution of each coefficient. We then visually isolate dates where one or more of the coefficients is anomalous. [Figure 3](#) shows an example of a histogram of the in-out assortativity coefficients for a 60-day window on an enterprise network. Note, the values are clustered near $r = -1$ as expected. Two things are immediately clear from this histogram. Although our sample is small, if we considered the in-out assortativity value on a given day to be a random variable, this histogram suggests that the probability density function is far from any standard distribution. This makes hypothesis testing to determine if a value is anomalous more challenging since we do not have distributional information available. Despite this, the histogram suggests that there are some reasonable choices for

outliers, such as the assortativity coefficients which fall into the $[-0.87, -0.88]$ bin in the histogram. These outliers correspond to particular days.

Suppose we have identified a potentially anomalous day. We now want to establish a cause for this behavior. To do so we consider the local assortativity coefficients for each IP address seen in our graph. In [figure 3](#) our anomalous day(s) appears to have an unusually high in-out assortativity value. Recall, the local assortativity coefficient for a particular vertex is that vertex's contribution to the global assortativity coefficient. Thus we are interested in individual vertices, or correspondingly IP addresses, that exhibit an unusually high local assortativity coefficient. This generates a set of IP addresses of interest for further study. These addresses can then be passed to a cybersecurity expert for examination.

A time series approach to analyzing and resolving anomalies

A time series is a collection of real-valued observations, $X = \{x_0, \dots, x_N\}$ made sequentially in time. We will briefly introduce some ideas in the area of time series analysis. For greater detail see, for example, [13]. Our goal is to use time series analysis to explain our anomalous IP addresses identified in the previous section on application to the LDAP protocol. To this end, we examine the time series of local assortativity coefficients for a particular IP address. We develop a time series model that best fits the series. We then use this model to forecast or predict the expected local assortativity values for the IP address and measure how far the predictions are from the corresponding true values.

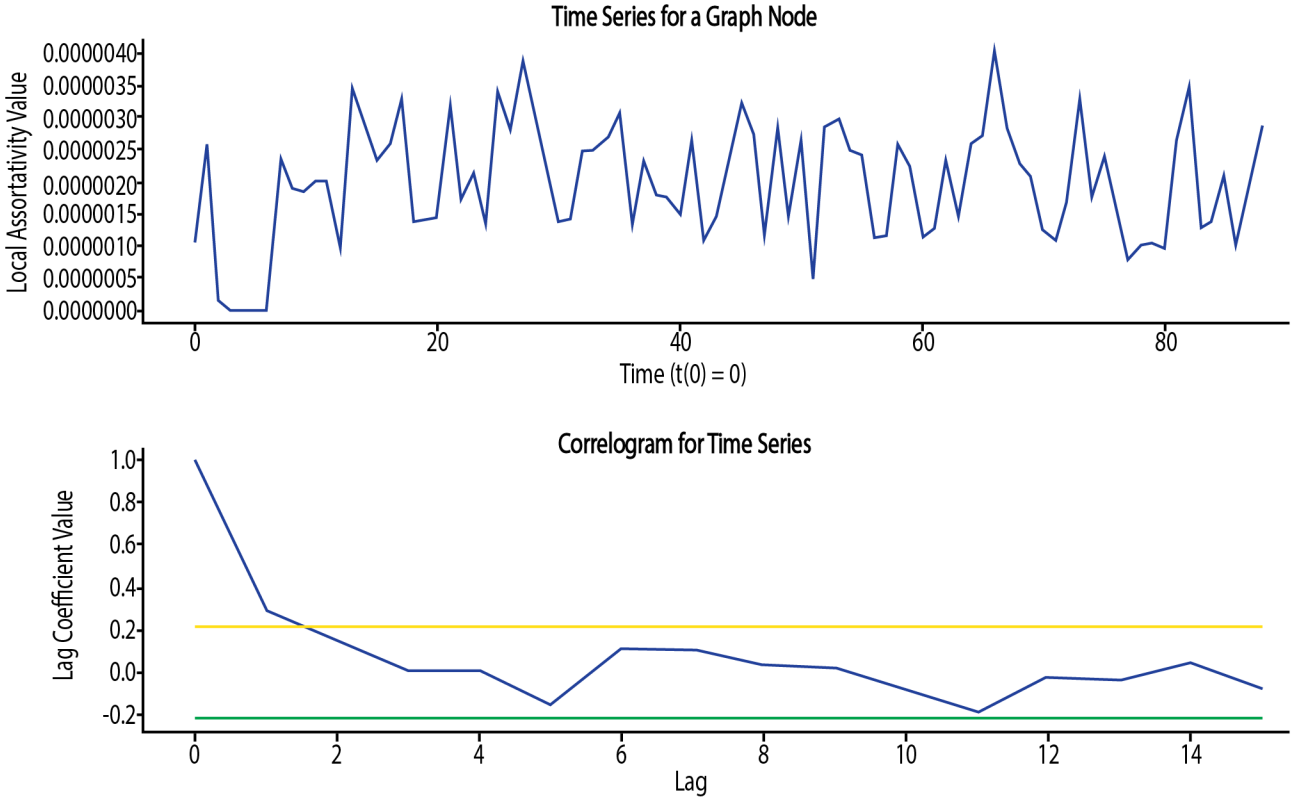


FIGURE 5. Example time series and correlogram of local assortativity values.

Given a time series $X = \{x_0, \dots, x_N\}$, we want to develop a model which can be used to forecast or predict series values. For X , the lag correlation coefficient of lag k measures how well correlated values that are k units of time apart in the time series are. It is given by

$$r_k = \frac{\sum_{i=0}^{N-k} (x_i - \bar{x})(x_{i+k} - \bar{x})}{\sum_{i=0}^N (x_i - \bar{x})^2}$$

Where $\bar{x} = N^{-1} \sum x_i$ is the average value of the time series. These values can be plotted in a correlogram and are useful for determining a model type that is likely to fit the data. Figure 5 shows a plot of a time series of the local assortativity coefficients of an individual IP address for 100 days and the corresponding correlogram. Recall, our goal is to use the time series of local assortativity coefficients to explain anomalous IP addresses. To do so, we identify a best-fit time series model of our data and then consider the residual error as a test statistic for quantifying how anomalous the behavior of the IP address is.

Under suitable assumptions, for a random time series, the lag correlation coefficients are approximately normally distributed with mean 0 and variance $\frac{1}{N}$. Thus, in the random case we expect the majority of the lag correlation coefficients to fall in the interval $[-\frac{2}{\sqrt{N}}, \frac{2}{\sqrt{N}}]$. The horizontal lines in the correlogram of figure 5 are placed to mark this interval. Notice that the autocorrelation coefficients initially decrease monotonically to $r_3=0$ and $r_i \approx 0$ for $i > 3$. This suggests that an autoregressive (AR) model would be a reasonable choice of model. We choose an AR model of order three given $r_3=0$, that is, the series value at time t , depends linearly on the values at times $t-1$, $t-2$, $t-3$. Thus, a framework for our model is given by

$$x_{t+3} = \alpha_0 x_t + \alpha_1 x_{t+1} + \alpha_2 x_{t+2} + \epsilon,$$


where $\epsilon \in \mathbb{R}$ represents some small error value that is normally distributed with mean 0. The α_i are parameters that are estimated by the method of least squares to find the model which gives the best approximation of the original time series.

Once we have found the coefficients for our model, we can generate the new series of predicted values $\{\hat{x}_3, \dots, \hat{x}_N\}$ with the value \hat{x}_i , corresponding to in the original series. Moreover, we can generate a series of residual error values $\mathcal{E} = \{e_i = \hat{x}_i - x_i \mid i=3, \dots, N\}$, which we can think of as a series of random variables, and consider the empirical probability density function for \mathcal{E}

Under idealized circumstances, that is, we know the true values for the and the choice of model is perfect, we would have $e_i = \epsilon$ and therefore $\mathcal{E} \approx N(0, \sigma_\epsilon)$. Unfortunately, in practice, the coefficients α_i are approximations. Thus we have to be careful in how we choose to use the e_i for hypothesis testing in general. Fortunately, if the model is chosen well, then in most applications \mathcal{E} tends to be approximately

normal. This fact can be leveraged to apply standard hypothesis testing techniques. For example, we can use Z-score testing on the empirical distribution of the residual errors to find outliers which correspond to days when the local assortativity coefficient of the IP address is anomalous.

Conclusion

We developed an application of graph theoretic and time series analysis techniques to answer questions about anomalies in the cybersecurity domain. This work can be used to develop a semi-automated workflow for monitoring an enterprise network. We introduced some interesting directions for future work including using alternative connectivity measures. 

References

- [1] Iliofotou M, Pappu P, Faloutsos M, Mitzenmacher M, Singh S, Varghese G. "Network monitoring using traffic dispersion graphs (TDGs)." In: *Proceedings of the 7th ACM SIGCOMM Conference on Internet Measurement (IMC '07)*; 2007; New York (NY): Association for Computing Machinery: pp. 315–320. Available at: <https://doi.org/10.1145/1298306.1298349>.
- [2] Newman MEJ. "Assortative mixing in networks." *Physical Review Letters*. 2002;89(20):208701. Available at: <https://doi.org/10.1103/PhysRevLett.89.208701>.
- [3] Newman MEJ. "Mixing patterns in networks." *Physical Review E*. 2003;67(2):26126. Available at: <https://doi.org/10.1103/PhysRevE.67.026126>.
- [4] Diestel R. *Graph Theory*. Springer-Verlag, 2005. ISBN: 9783540261834.
- [5] Bang-Jensen J, Gutin G. *Digraphs: Theory, Algorithms, and Applications*. Springer-Verlag, 2001. ISBN: 9781852332686.
- [6] Piraveenan M, Prokopenko M, Zomaya A. "Assortative mixing in directed biological networks." *IEEE/ACM Transactions on Computational Biology and Bioinformatics*. 2012;9(1):66–78. Available at: <https://doi.org/10.1109/TCBB.2010.80>.
- [7] Foster JG, Foster DV, Grassberger P, Paczuski M. "Edge direction and the structure of networks." In: *Proceedings of the National Academy of Sciences*, 2010 Jun 15;107(24):10815–20. doi: 10.1073/pnas.0912671107.
- [8] Noldus R, Van Mieghem P. "Assortativity in complex networks." *Journal of Complex Networks*. 2015;3(4):507–542. Available at: <https://doi.org/10.1093/comnet/cnv005>.
- [9] Piraveenan M, Prokopenko M, Zomaya AY. "Classifying complex networks using unbiased local assortativity." In: *Artificial Life XII: Proceedings of the 12th International Conference on the Synthesis and Simulation of Living Systems, ALIFE*; 2010. pp. 329–336.
- [10] M. Piraveenan, Prokopenko M, Zomaya AY. "Local assortativeness in scale-free networks." *Euro-Physics Letters*. 2008;84(2). doi: 10.1209/0295-5075/84/28002.
- [11] Hoorn P, Litvak N. "Degree-degree dependencies in directed networks with heavy-tailed distributions." *Internet Mathematics*. 2015;11(2):155–179. Available at: <https://doi.org/10.1080/15427951.2014.927038>.
- [12] Litvak N, Hofstad R. "Uncovering disassortativity in large scale-free networks." *Physical Review E*. 2013;87(2):22801–22808. Available at: <https://doi.org/10.1103/PhysRevE.87.022801>.
- [13] Chatfield C. *The Analysis of Time Series An Introduction*. Chapman and Hall/CRC: 2004.