Automated Discovery of Novel Anomalous Patterns

Edward McFowland III
Machine Learning Department
School of Computer Science
Carnegie Mellon University
mcfowland@cmu.edu

DAP Committee:
Daniel B. Neill
Jeff Schneider
Roy Maxion

Abstract

We propose Discovering Novel Anomalous Patterns (DAP), a new method for continual and automated discovery of anomalous patterns in general datasets. Currently, general methods for anomalous pattern detection attempt to identify data patterns that are unexpected as compared to "normal" system behavior. We propose a novel approach for discovering data patterns that are unexpected given a profile of previously known, both normal and abnormal, system behavior. This enables the DAP algorithm to identify previously unknown data patterns, add these newly discovered patterns to the profile of "known" system behavior, and continue to discover novel (unknown) patterns. We evaluate the performance of DAP in two domains of computer system intrusion detection (network intrusion detection and masquerade detection), demonstrating that DAP can successfully discover and characterize relevant patterns for these two tasks. As compared to the current state of the art, DAP provides a substantially improved ability to discover novel patterns in massive multivariate datasets.

1 Introduction

The ability to automatically and continually discover novel anomalous patterns in massive multivariate data is an important tool for knowledge discovery. For concreteness consider an analyst who is tasked with detecting attacks or intrusions on a system. It is imperative that the analyst identify novel (previously unknown) attacks, allowing for the proper rectification of possible system exploits. There is still utility in identifying previously known attacks, as it may highlight the persistence of specific vulnerabilities, however this is a challenge of classification not necessarily pattern discovery. After identifying a new attack and following the established security protocols, the analyst will resume his task of discovering previously unknown threats to the system.

Here we focus on the task of continual anomalous pattern discovery in general data, i.e., datasets where data records are described by an arbitrary set of attributes. We describe the anomalous pattern discovery problem as continually detecting groups of anomalous records and characterizing their anomalous features, with the intention of understanding the anomalous process that generated these groups. The anomalous pattern discovery task begins with the assumption that there are a set of processes generating records in a dataset and these processes can be partitioned into two sets: those currently known and unknown to the system. The set of known processes contains the "background" process, which generates records that are typical and expected; these records are assumed to constitute the majority of the dataset. Records that do not correspond to the background or any of the other known data patterns, and therefore represent unfamiliar system behavior, are assumed to have been generated by an anomalous process and follow an alternative data pattern.

Anomalous pattern discovery diverges from most previous anomaly detection methods, as the latter traditionally focuses on the detection of single anomalous data records, e.g., detecting a malicious network connection or session. If these anomalies are generated by a process which is very different from every element in the set of known processes, it may be sufficient to evaluate each individual record in isolation because many of the records' attributes will be atypical, or individual attributes may take on extremely surprising values, when considered under the known data distributions. However, a subtle anomalous process—an intelligent intruder who attempts to disguise their activity so that it closely resembles legitimate connections—will generate records that may each be only slightly anomalous and therefore extremely challenging to detect. In such a case, each individual record (malicious session) may only be slightly anomalous. The key insight is to acknowledge and leverage the group structure of these records, since we expect records generated by the same process to have a high degree of similarity. Therefore, we propose to incrementally detect self-similar groups of records, for which some subset of attributes are unexpected given the known data distributions.

The rest of the paper is organized as follows: in Section 2, we review related work; then in Section 3, we present our general anomalous pattern discovery framework, while our specific algorithmic solution to the framework is presented in Section 4; the experimental results are discussed in 5; finally, we conclude this work in Section 6.

2 Related Work

In this section, we review the related work, which can be categorized into three major groups: intrusion detection, rare category analysis and anomalous pattern detection for general data. Additionally, we juxtapose the general tasks of anomalous pattern discovery and anomalous pattern detection, while proposing a simple extension to the current state of the art method in anomalous pattern detection, allowing it accomplish the task of discovery. We will utilize this simple extension as a baseline for comparison to our method presented in Section 4.

Network Intrusion Detection continues to be a popular topic of research in computer systems security and machine learning. [13] provides a historical overview of intrusion and intrusion detection from the early 1970s to the early 2000s, while [15] and [11] focus more on methodological advances over the latter part of this time span. Similarly, [20] provide an overview of popular and more recent machine learning algorithms designed for classifying or detecting intrusions. We argue that all the methods surveyed by these authors suffer from at least one important limitation, separating it from the work we introduce in this article as well as the other related work discussed below. Many methods require previous examples of relevant intrusions and are specifically designed for the challenge of network intrusions; therefore, these methods can attribute a significant amount of their success to prior knowledge of the behavior of interest. A second limitation is that most methods evaluate each record in isolation, while the methods evaluated below attempt to leverage the group structure of records the potentially relevant records. A third limitation is the inability of these methods to provide a framework for the continual uncovering of the various abnormal pattern types. Instead of focusing on the specific task of identifying network intrusions, our work is a general method for *continual* pattern discovery, most useful for incrementally identifying interesting and non-obvious patterns occurring in the data, when there is little knowledge of what patterns to look for. Additionally, the anomalous pattern detection literature [3, 17, 1, 9] described below demonstrates that evaluating potential records of interest in the context of all other records likely to be generated by the same anomalous process, can significantly improve our power to detect anomalies (including those in the network intrusion domain) as compared to evaluating records in isolation.

Rare Category Detection is defined as given a set of data points $\{R_1, \ldots, R_n\}$, with $R_i \in \mathbb{R}^d$, each from one of K distinct classes, finding at least one example from each class with the help of a labeling oracle, while minimizing the number of label queries. An initial solution to this task fits a mixture model to the data, and selects points to label based on the Interleaving approach [18]. However, this approach and many others require that the majority classes and minority (rare) classes are separable, or experience significantly compromised detection power when this assumption is violated. The authors of [5, 6] allow for inseparable classes, but still require knowing the number of rare classes and the proportion of of the data that belong to this classes. Additionally, there is the assumption that the probability densities of the majority classes are sufficiently smooth, and the support regions of the each minority class is small. [7] relaxes the requirement for prior information, but assumes the probability densities follows a semi-parametric form.

Each these methods are essentially mechanisms to rank data points based on the likelihood they are members of a rare class. The goal of our work is to *continually* discover the existence of an anomalous process by finding a *group* of records for which a *subset* of its attributes are anomalous. The authors of [8] do propose a method for simultaneously detecting and characterizing the features of a minority class; however, it is only optimized for existence of one minority class and again requires the proportion of the data belonging to the minority class be provided. More generally, rare category detection paradigm assumes each data point is a vector of real-values, while the anomalous pattern discovery paradigm permits categorical and real-valued data features. This assumption of only real-values is exploited by the theory and optimization mechanisms found in much of this literature; if possible, extending these methods to mixed datasets would be non-trivial.

Anomalous Pattern Detection for general datasets can be reduced to a sequence of tasks: learning a model M_0 to representing the *normal* or expected data distribution, defining the search space (i.e., which subsets of the data will be considered), choosing a function to score the interestingness or anomalousness of a subset, and optimizing this function over the search space in order to find the highest scoring subsets. Each anomalous pattern detection method discussed here, learns the structure and parameters of a Bayesian network from training data to represent M_0, and then searches for subsets of records in test data that are collectively anomalous given M_0. [2] presents a simple solution to the problem of individual record anomaly detection by computing each record's likelihood given M_0, and assuming that the lowest-likelihood records are most anomalous. Although efficient, this method will lose power to detect anomalous groups produced by a subtle anomalous process where each record, when considered individually, is only slightly anomalous. The authors of [3] allow for subsets larger than one record by finding rules (conjunction of attribute values) with higher than expected number of low-likelihood records. However, this method also loses power to detect subtle anomalies because of its dependency on individual record anomalousness, permission of records within a group to each be anomalous for different reasons, and the need to reduce its search space of all rules to only those composed of 2-components. [17, 1] increases detection ability by maximizing a likelihood ratio statistic over *subsets* of records, however this method must reduce its search space and use a greedy heuristic because the space of all possible subsets of records is too vast to search over exhaustively. The authors of [9] improve on these previous methods by defining the pattern detection problem as a search over subsets of data records *and* subsets of attributes, and searching for self-similar groups of records for which some subset of attributes is anomalous given M_0.

3 Anomalous Pattern Discovery Framework

We propose a general framework that allows for the *continual* discovery of novel anomalous patterns by maintaining a collection of models representing known pattern types $M_{\text{known}} = \{M_0, \ldots, M_K\}$. Similar to [9], we frame our discovery problem as a search over subsets of data record and subsets of attributes. More precisely, we define a set of data records $R = \{R_1 \ldots R_N\}$ and attributes $A = \{A_1 \ldots A_M\}$. For each record R_i, we assume that we have a value v_{ij} for each attribute A_j. We then define the subsets S under consideration to be $S = R \times A$, where $S_R \subseteq R$ and $S_A \subseteq A$. The null hypothesis H_0 is that every record R_i is drawn from a known pattern type M_k, while the alternative hypothesis $H_1(S = S_R \times S_A)$ is that for the subset of records S_R, the subset of attributes S_A are not drawn from any of the known models. Therefore, we wish to identify the most anomalous subset

$$S^* = S^*_R \times S^*_A = \arg\max_S F(S), \tag{1}$$

where the score function $F(S)$ defines the anomalousness of a subset of records and attributes when considered across $K + 1$ known pattern types.

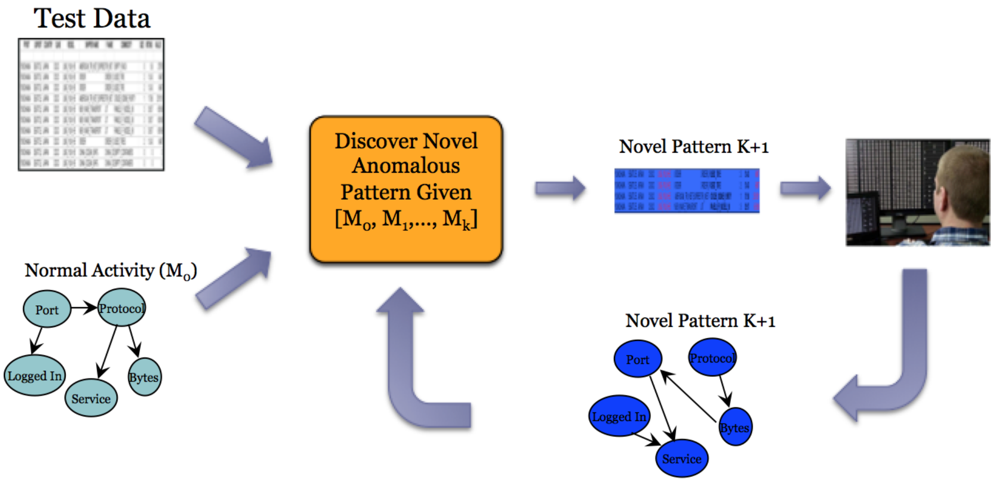

Figure 1: Anomalous Pattern Discovery Framework

Our framework, shown in Figure 3, captures the integral components of an anomalous pattern discovery method:

(1) A (probabilistic) model capable of representing a data generating process, i.e., the "normal" and the other known pattern types.
(2) Given test data, and possibly multiple known pattern types, a method capable of identifying the subset of data most likely generated by an unknown process.
(3) A mechanism to update component (2) with curated patterns representing an novel data generating process.

The pattern detection task addressed in [9] and our task of pattern discovery may appear to be framed similarly–e.g., a search over subsets of records and subsets of attributes–however, every pattern detection method described in §2 fails to demonstrate the last two integral features described above. The pattern detection methods cannot distinguish between subsets generated by multiple known and unknown processes, while continually incorporating information from discovered patterns, because they are only capable of modeling the background pattern type M_0. It is therefore very unlikely that our analyst from §1 would be able to utilize any of these pattern detection methods to identify pernicious system activity indicative of a novel attack because patterns that have been previously identified, but are highly abnormal as compared to baseline behavior, would continue to be reported. Furthermore, if the analyst was capable of identifying a novel threat, these methods provide no updating mechanism that allows for this knowledge to be incorporated into future searches. We note that one simple way to allow pattern detection methods to incorporate this knowledge into future searches is to simply update the background model M_0, having it represent all known pattern types. Intuitively, this is less than ideal because with each additional pattern type M_0 is required to represent, the model dilutes and loses the ability to represent any individual pattern type well. In §5 we will demonstrate this phenomena empirically by comparing our method for anomalous pattern discovery to this simple extension, which we will refer to as FGSS-MM (a mixture model extension to the Fast Generalized Subset Scan algorithm proposed in [9]).

Although the current state of pattern detection does not address these necessary conditions of pattern discovery because it assumes the existence of only one known pattern type M_0, we can think of the pattern detection task as a special case of pattern discovery, occurring when the collection of known data models only includes M_0. More specifically, our Discovering Novel Anomalous Patterns algorithm presented in section 4 is a generalization of the Fast Generalized Subset Scan algorithm for anomalous pattern detection proposed by the authors in [9], allowing for the *continual* discovery of self-similar groups of records, for which some subset of attributes are unexpected given a *collection* of known data distributions.

4 Discovering Novel Anomalous Patterns

Discovering Novel **A**nomalous **P**atterns (DAP) is a method for continual anomalous pattern discovery in general data given a set of known models, where each model represents the data distribution when a particular known pattern type is present. The DAP algorithm assumes that all attributes are categorical, binning continuous valued attributes, but future work will extend the approach to explicitly model continuous attributes as well. The DAP algorithm is designed to discover the existence of an anomalous data generating process, by detecting groups of self-similar records that are unlikely to be generated by a known data process. A record is considered unlikely if the conditional probabilities of a subset of its attributes are significantly low when evaluated under each known data distribution separately. An overview of the DAP algorithm is given in Section 4.7, and we now explain each step in detail.

4.1 Modeling Pattern Types

The DAP algorithm first learns a Bayesian network corresponding to each of the known pattern types M_{known} using the Optimal Reinsertion method proposed by [14] to learn the structures, and using smoothed maximum likelihoods to estimate the parameters of the conditional probability tables. The background model M_0 is typically learned from a separate "clean" dataset of training data assumed to contain no anomalous patterns, but can also be learned from the test data if the proportion of anomalies is assumed to be very small. The Bayesian network representing a particular anomalous pattern type can similarly be learned from labeled training data, or from examples discovered in the test data after a particular iteration of the DAP algorithm. For the purposes of this current work, we assume that we have a sufficiently large collection of training data from which to learn each model, while a focus of future work will be the challenge of learning representative models from possibly small samples of novel anomalous patterns.

4.2 Computing The Anomalousness of Data Values

Given a model M_k representing a known pattern type, we compute

$$l_{ijk} = P_{M_k}(A_j = v_{ij} \mid A_{p(j)} = v_{i,p(j)}) \tag{2}$$

representing the conditional probability of the observed value v_{ij} under M_k, given its observed parent attribute values for record R_i. We compute these individual attribute-value likelihoods for all records in the test dataset, under each model M_k, producing a $N \times M$ matrix of likelihoods for each of the $K+1$ models. We note that

$$l_{ijk} = P_{M_k}(A_j = v_{ij} \mid A_{b(j)} = v_{i,b(j)}), \tag{3}$$

where $b(j)$ returns the Markov blanket of attribute A_j, is an alternative formulation to (2). More specifically, (3) represents the conditional probability of the observed value v_{ij} under M_k, given *all* the other attributes' values for record R_i. The parent attributes of A_j, the conditioning set in (2), are a subset of A_j's Markov blanket, the conditioning set in (3). The major difference is that (2) assumes that an anomalous process will make an ex ante intervention on the underlying generating process (i.e., a node is forced to take a given value while the data record is being generated, thus also affecting the probability distributions of the node's descendants in the Bayesian network model), while (3) assumes an anomalous process will make an ex post intervention (a node is forced to take a given value after the data is generated, leaving all other attributes' values unaffected). Our DAP algorithm can accommodate either conditional probability formulation, but for evaluation purposes we select (2) as it better corresponds to the data generating process of our experimental data described in §5.

The next step of our algorithm converts each element in the matrix of test likelihoods l_{ijk} to a corresponding empirical p-value range p_{ijk}. To properly understand this conversion, consider that for an attribute A_j

there is some true distribution of likelihoods l_{ijk} under M_k. If we allow L_{jk} to represent a random variable drawn from this this distribution, we can define the quantities

$$p_{\min}(p_{ijk}) = P_{M_k}\left(L_{jk} < l_{ijk}\right), \tag{4}$$
$$p_{\max}(p_{ijk}) = P_{M_k}\left(L_{jk} \leq l_{ijk}\right). \tag{5}$$

Intuitively, (5) represents the probability of getting an equally likely or more unlikely value of attribute A_j for record R_i, if it were truly generated by model M_k. More specifically, it is equivalent to evaluating the cumulative distribution function (represented by M_k) of A_j's likelihoods at l_{ijk}. Therefore, if R_i is truly drawn from M_k, then (5) represents an empirical p-value and is asymptotically distributed Uniform$[0, 1]$. We then define the empirical p-value range corresponding to likelihood l_{ijk} as

$$p_{ijk} = \left[p_{\min}(p_{ijk}), p_{\max}(p_{ijk})\right].$$

In addition to the many advantages of empirical p-value ranges described by [9], including their ability to appropriately handle ties in likelihoods, we will present advantages in the specific context of anomalous pattern discovery. We begin by defining the quantity $n_\alpha(p_{ij})$, representing the significance of a p-value range, as follows:

$$n_\alpha(p_{ijk}) = \begin{cases} 1 & \text{if } p_{\max}(p_{ijk}) < \alpha \\ 0 & \text{if } p_{\min}(p_{ijk}) > \alpha \\ \frac{\alpha - p_{\min}(p_{ijk})}{p_{\max}(p_{ijk}) - p_{\min}(p_{ijk})} & \text{otherwise.} \end{cases}$$

This quantity is a generalization of the traditional binary measurement of significance for p-values, necessary for p-value ranges, and can be considered the proportion of a range that is significant at level α, or equivalently, the probability that a p-value drawn uniformly from $[p_{\min}(p_{ijk}), p_{\max}(p_{ijk})]$ is less than α. Next we define the following functions:

$$m(R_i, S_A) = \arg\min_k \sum_{A_j \in S_A} n_\alpha(p_{ijk}), \tag{6}$$
$$n_\alpha(R_i, S_A) = \sum_{A_j \in S_A} n_\alpha\left(p_{ijk}\right) \quad s.t. \quad k = m\left(R_i, S_A\right) \tag{7}$$

The function in (6) identifies which known model minimizes a record R_i's number of significant p-value ranges (or more intuitively which model "best fits" the record) for a given subset of attributes S_A, while (7) calculates how many p-value ranges are significant for record R_i and attributes S_A when considered under the minimizing (empirically "best fitting") model. Now for a subset $S = S_R \times S_A$, we can then define the following quantities:

$$N_\alpha\left(S\right) = \sum_{R_i \in S_R} n_\alpha(R_i, S_A), \tag{8}$$
$$N(S) = \sum_{R_i \in S_R} \sum_{A_j \in S_A} 1. \tag{9}$$

The quantity in (8) can informally be described as the number of p-value ranges in S which are significant at level α–when each record is mapped to its minimizing model–but is more precisely the total probability mass less than α in these p-value ranges, since it is possible for a range p_{ijk} to have $p_{\min}(p_{ijk}) \leq \alpha \leq p_{\max}(p_{ijk})$. The quantity in (9) represents the total number of empirical p-value ranges contained in subset S, after each record has been mapped to a particular "best fitting" model. We also define

$$m^{true}(R_i) = M_k \in M_{\text{known}} \quad s.t. \quad R_i \sim M_k, \tag{10}$$
$$n_\alpha^{true}(R_i, S_A) = \sum_{A_j \in S_A} n_\alpha\left(p_{ijk}\right) \quad s.t. \quad k = m^{true}\left(R_i\right), \tag{11}$$

where (10) returns R_i's true data generating process, and note that

$$N_\alpha(S) \leq \sum_{R_i \in S_R} n_\alpha^{true}(R_i, S_A),$$

because the minimum of a set is trivially bounded above by any element of the set, and recall we assume under H_0 that the true generating process in contained in M_{known}.

Therefore, for a subset $S = S_R \times S_A$ consisting of $N(S)$ empirical p-value ranges, we can compute an upper-bound on the expected number of significant p-value ranges under the null hypothesis H_0:

$$\begin{aligned} E\left[N_\alpha(S)\right] &= E\left[\sum_{R_i \in S_R} n_\alpha(R_i, S_A)\right] \\ &= \sum_{R_i \in S_R} E\left[n_\alpha(R_i, S_A)\right] \\ &\leq \sum_{R_i \in S_R} E\left[n_\alpha^{true}(R_i, S_A)\right] \\ &= \sum_{R_i \in S_R} \sum_{A_j \in S_A} E\left[n_\alpha\left(p_{ijk}\right)\right] \quad s.t. \quad k = m^{true}\left(R_i\right) \\ &= \sum_{R_i \in S_R} \sum_{A_j \in S_A} \alpha \\ &= \alpha N(S). \end{aligned} \tag{12}$$

We note that

$$E\left[N_\alpha(S)\right] = \alpha N(S) \iff n_\alpha(R_i, S_A) = n_\alpha^{true}(R_i, S_A) \quad \forall \quad R_i \in S_R,$$

otherwise,

$$E\left[N_\alpha(S)\right] < \alpha N(S).$$

Additionally, (12) follows from the property that the empirical p-values are identically distributed as Uniform[0,1] under the null hypothesis, and holds regardless of whether the p-values are independent. Under the alternative hypothesis, we expect the all the likelihoods l_{ijk} (and therefore the corresponding p-value ranges p_{ijk}) to be lower for the affected subset of records and attributes, resulting in a higher value of $N_\alpha(S)$ for some α. Therefore a subset S where $N_\alpha(S) > \alpha N(S)$ (i.e., a subset with a higher than expected number of low, significant p-value ranges) is potentially affected by an anomalous process.

4.3 Evaluating Subsets

To determine which subsets of the data are most anomalous, FGSS utilizes a nonparametric scan statistic [9] to compare the observed and expected number of significantly low p-values contained in subset S. We define the general form of the nonparametric scan statistic as

$$F(S) = \max_\alpha F_\alpha(S) = \max_\alpha \phi(\alpha, N_\alpha(S), N(S)) \tag{13}$$

where $N_\alpha(S)$ and $N(S)$ are defined as in (8) and (9) respectively. Then our algorithm utilizes the Higher Criticism (HC) nonparametric scan statistic [4, 9] to compare the observed and expected number of significantly low p-value ranges contained in subset S. The HC statistic is defined as follows:

$$\phi_{\text{HC}}(\alpha, N_\alpha, N) = \frac{N_\alpha - N\alpha}{\sqrt{N\alpha(1-\alpha)}}. \tag{14}$$

Under the null hypothesis of uniformly distributed p-value ranges, and the additional simplifying assumption of independence between p-value ranges, the number of empirical p-value ranges less than α is binomially distributed with parameters N and α. Therefore the expected number of p-value ranges less than α under H_0 is $N\alpha$, with a standard deviation of $\sqrt{N\alpha(1-\alpha)}$. This implies that the HC statistic can be interpreted as the test statistic of a Wald test for the number of significant p-value ranges. We note that the assumption of independent p-value ranges is not necessarily true in practice, since our method of generating these p-value ranges may introduce dependence between the p-values for a given record; nevertheless, this assumption results in a simple and efficiently computable score function.

Although, we can use our nonparametric scan statistic $F(S)$ to evaluate the anomalousness of subsets in the test data, naively maximizing $F(S)$ over all possible subsets of records and attributes would be infeasible for even moderately sized datasets, with a computational complexity of $O(2^N \times 2^M)$. However, [9, Corollary 2] demonstrates that a general class of nonparametric scan statistics, including (14), have the the linear-time subset scanning (LTSS) property [16], which allows for efficient and exact maximization of (13) over all subsets of the data. More specifically, for a pair of functions $F(S)$ and $G(R_i)$, which represent the "score" of a given subset S and the "priority" of data record R_i respectively, the LTSS property guarantees that the only subsets with the potential to be optimal are those consisting of the top-t highest priority records $\{R_{(1)} \dots R_{(t)}\}$, for some t between 1 and N. This property enables us to search only N of the 2^N subsets of records, while still guaranteeing that the highest-scoring subset will be found.

The objective of maximizing (1) can be re-written, using (13), as

$$\max_{\alpha} \max_{S} \phi(\alpha, N_\alpha(S), N(S)). \tag{15}$$

Furthermore, [9, Corollary 2] demonstrates that for a given α and subset of attributes S_A we can compute

$$\max_{S} \phi(\alpha, N_\alpha(S), N(S)) \tag{16}$$

over all 2^N subsets $S = S_R \in 2^R \times S_A$ with the following steps:

(R1) Compute the priority for each record using

$$G_\alpha(R_i) = n_\alpha^-(R_i, S_A)). \tag{17}$$

(R2) Sort the records from highest to lowest priority.
(R3) For each of the subsets $S = \{R_{(1)} \dots R_{(t)}\} \times S_A$, with $t = 1 \dots N$, compute $\phi\left(\alpha, N_\alpha(S), N(S)\right)$.

Similarly, for a given α, subset of records S_R, and a subset of attributes $S_A^{'}$ used to map each record to a model, we can compute (16) over all 2^M subsets $S = S_R \times S_A \in 2^A$ with the following steps:

(A1) Compute the priority for each attribute using

$$G_\alpha(A_i, S_A^{'}) = \sum_{R_i \in S_R} n_\alpha\left(p_{ijk}\right) \quad s.t. \quad k = m\left(R_i, S_A^{'}\right),$$

(A2) Sort the attributes from highest to lowest priority.
(A3) For each of the subsets $S = S_R \times \{A_{(1)} \dots A_{(t)}\}$, with $t = 1 \dots M$, compute $\phi\left(\alpha, N_\alpha(S), N(S)\right)$.

Thus the LTSS property enables efficient computation of (16), over subsets of records or subsets of attributes, for a given value of α, but we still want to consider "significance levels" α between 0 and some constant $\alpha_{\max} < 1$. Maximizing $F(S)$ over a range of α values, rather than for a single arbitrarily-chosen value of α, enables the nonparametric scan statistic to detect a small number of highly anomalous p-values, a larger number of subtly anomalous p-values, or anything in between. [9, Theorem 7] demonstrates that when computing $\max_\alpha F_\alpha(S)$, only a small quantity of values can possibly be the maximizing value of α.

More specifically, the values of $\alpha \in U(S, \alpha_{\max})$ must be considered, where $U(S, \alpha_{\max})$ is the set of distinct values $\{p_{\max}(p_{ijk}) : v_{ij} \in S,\, p_{\max}(p_{ijk}) \leq \alpha_{\max}\} \cup \{0, \alpha_{\max}\}$. Therefore,

$$\begin{aligned} \max_S F(S) &= \max_\alpha \max_S F_\alpha(S) \\ &= \max_{\alpha \in U(S,\alpha_{\max})} \max_S F_\alpha(S) \end{aligned} \tag{18}$$

can be efficiently and exactly computed over all subsets $S = S_R \times S_A$, where $S_R \subseteq \{R_1 \ldots R_N\}$, for a given subset of attributes S_A. To do so, we consider the set of distinct α values $U = U(\{R_1 \ldots R_N\} \times S_A, \alpha_{\max})$. For each $\alpha \in U$, we follows steps R1-R3 described above. Similarly, (18) can be efficiently and exactly computed over all subsets $S = S_R \times S_A$, where $S_A \subseteq \{A_1 \ldots A_M\}$, for a given subset of records R. In this case, we consider the set of distinct α values $U = U(S_R \times \{A_1 \ldots A_M\}, \alpha_{\max})$. For each $\alpha \in U$, we follow the steps A1-A3.

4.4 Search Procedure

We propose the DAP search procedure that scales well with both N and M, utilizing the optimizations described above to efficiently maximize over subsets of records and subsets of attributes. To do so, we first choose a subset of attributes $A \subseteq \{A_1 ... A_M\}$ uniformly at random. We then iterate between the efficient optimization steps described above: optimizing over all subsets of attributes for a given subset of records, mapping each record to its best fit model, and optimizing over all subsets of records for a given subset of attributes. We first compute the minimizing mapping $MAP = \{map_1, \ldots, map_N\}$ for each record to its "best fit" model, for the current subset of attributes A, as follows: $map_i = \arg\min_{\mathrm{MAP}: R_i \dashrightarrow M_0, M_1, \ldots, M_K} N_\alpha(S = A \times R_i | map_i) \quad \forall i$. We then maximize $F(S)$ over all subsets of records for the current subset of attributes A and mapping MAP, and set the current set of records as follows: $R = \arg\max_{R \subseteq \{R_1 \ldots R_N\}} F(R \times A | MAP)$. We then maximize $F(S)$ over all subsets of attributes for the current subset of records R and mapping MAP, and set the current set of attributes as follows: $A = \arg\max_{A \subseteq \{A_1 \ldots A_M\}} F(R \times A | MAP)$ We continue iterating between these steps until the procedure stops increasing. This ordinal ascent approach is not guaranteed to converge to the joint optimum but multiple random restarts can be used to approach the global optimum. Moreover, if N and M are both large, this iterative search is much faster than an exhaustive search approach, making it computationally feasible to detect anomalous subsets of records and attributes in datasets that are both large and high-dimensional. Each iteration (mapping of records, followed by optimization over records and optimization over attributes) has a complexity of $O(|U|(KNM + N \log N + M \log M))$, where $|U|$ is the average number of α thresholds considered. In this expression, the $O(KNM)$ term results from aggregating over records and attributes for each model, while the $O(N \log N)$ and $O(M \log M)$ terms result from sorting the records and attributes by priority respectively. Thus the DAP search procedure has a total complexity of $O(YZ|U|(KNM + N \log N + M \log M))$, where Y is the number of random restarts and Z is the average number of iterations before the procedure stops increasing. Since each iteration step optimizes over all subsets of records (given the current subset of attributes) and all subsets of attributes (given the current subset of records), convergence is extremely fast, with average values of Z less than 3.0 for all of our experiments described below.

4.5 Self-Similarity

We believe records generated by the same anomalous process are expected to be similar to each other. The self-similarity of the detected subsets can be ensured by enforcing a similarity constraint. We augment the above the DAP search procedure by defining the "local neighborhood" of each record in the test dataset, and then performing an unconstrained DAP search for each neighborhood, where $F(S)$ is maximized over all subsets of attributes and over all subsets of records contained within that neighborhood. Given a metric $d(R_i, R_j)$ which defines the distance between any two data records, we define the local neighborhood of R_i as $\{R_j : d(R_i, R_j) \leq r\}$, where r is some predefined distance threshold. We then find the maximum score over all similarity-constrained subsets. The DAP constrained search procedure has a complexity of

$O(YZ|U|N(hKM + h\log h + M\log M))$, where h is the average neighborhood size (number of records) corresponding to distance threshold r.

4.6 Randomization Testing

The step of the algorithm is optional as it includes randomization testing to compute the statistical significance of the detected subset S. To perform randomization testing, we create a large number T of "replica" datasets under the null hypothesis, perform the same scan (maximization of $F(S)$ over self-similar subsets of records and attributes) for each replica dataset, and compare the maximum subset score for the original data to the distribution of maximum subset scores for the replica datasets. More precisely, we create each replica dataset, containing the same number of records as the original test dataset, by sampling uniformly at random from the training data or by generating random records according to our Bayesian Networks representing H_k where $k = 0 \ldots K$. We then use the previously described steps of the DAP algorithm to find the score of the most anomalous subset $F^* = \max_S F(S)$ of each replica. We can then determine the statistical significance of each subset S detected in the original test dataset by comparing $F(S)$ to the distribution of F^*. The p-value of subset S can be computed as $\frac{T_{\text{beat}}+1}{T+1}$, where T_{beat} is the number of replicas with F^* greater than $F(S)$ and T is the total number of replica datasets. If this p-value is less than our significance level fpr, we conclude that the subset is significant. An important benefit of this randomization testing approach is that the overall false positive rate is guaranteed to be less than or equal to the chosen significance level fpr. However, a disadvantage of randomization testing is its computational expense, which increases run time proportionally to the number of replications performed. Our results discussed in §5 directly compare the scores of "clean" and anomalous datasets, and thus do not require the use of randomization testing.

4.7 DAP Algorithm

Inputs: test dataset, training dataset(s), $\alpha_{\max}$, r, Y.

1. Learn Bayesian Networks (structure and parameters) from the training dataset(s).

2. For each data record R_i and each attribute A_j, in both training and test datasets, compute the likelihood l_{ijk} given the Bayesian Networks $M_0, \ldots, M_k$.

3. Compute the p-value range $p_{ijk} = [p_{\min}(p_{ijk}), p_{\max}(p_{ijk})]$ corresponding to each likelihood l_{ijk} in the test dataset.

4. For each (non-duplicate) data record R_i in the test dataset, define the local neighborhood S_i to consist of R_i and all other data records R_j where $d(R_i, R_j) \leq r$.

5. For each local neighborhood S_i, iterate the following steps Y times. Record the maximum value F^* of $F(S)$, and the corresponding subsets of records R^* and attributes A^* over all such iterations:

   (a) Initialize $A \leftarrow$ random subset of attributes.

   (b) Repeat until $F(S)$ stops increasing:

      i. Minimize $MAP_i = \arg\min_{\text{map}:R_i \dashrightarrow M_0, M_1, \ldots, M_K} N_\alpha(S = A \times R_i | map_i) \quad \forall i$.

      ii. Maximize $F(S) = \max_{\alpha \leq \alpha_{\max}} F_\alpha(R \times A|MAP)$ over subsets of records $R \subseteq \{R_1 \ldots R_N\}$, for the current subset of attributes A, and set $R \leftarrow \arg\max_{R \subseteq \{R_1 \ldots R_N\}} F(R \times A|MAP)$.

      iii. Maximize $F(S) = \max_{\alpha \leq \alpha_{\max}} F_\alpha(R \times A|MAP)$ over all subsets of attributes A, for the current subset of records R, and set $A \leftarrow \arg\max_{A \subseteq \{A_1 \ldots A_M\}} F(R \times A|MAP)$.

6. Output $S^* = R^* \times A^*$.

5 Evaluation

In this section, we compare the detection performance of the DAP algorithm to FGSS-MM algorithm introduced in §3. We consider datasets from the domain of computer system intrusion detection (network intrusion detection and masquerade detection) in order to evaluate both method's ability identify anomalous patterns given a set of known models. These datasets are described in §5.1 and §5.2 respectively, along with the evaluation results for each domain. In §5.3, we consider the we simulate a realistic pattern discovery scenario and compare the methods' ability to continually discovery novel anomalous patterns.

We define two metrics for our evaluation of detection power: area under the precision/recall (PR) curve, which measures how well each method can distinguish between anomalous and normal records, and area under the receiver operating characteristic (ROC) curve, which measures how well each method can distinguish between datasets which contain anomalous patterns and those in which no anomalous patterns are present. In each case, a higher curve corresponds to better detection performance.

To precisely define these two metrics, we first note that three different types of datasets are used in our evaluation. The *training dataset* only contains records representing typical system behavior (i.e., no anomalous patterns are present) and is used to learn the null model. Each *test dataset* is composed of records that represent typical system behavior as well as anomalous groups, while each *normal dataset* has the same number of records as the test datasets but does not contain any anomalous groups.

For the PR curves, each method assigns a score to each record in each test dataset, where a higher score indicates that the record is believed to be more anomalous, and we measure how well the method ranks true anomalies above non-anomalous records. The list of record scores returned by a method are sorted and iterated through: at each step, we use the score of the current record as a threshold for classifying anomalies, and calculate the method's precision (number of correctly identified anomalies divided by the total number of predicted anomalies) and recall (number of correctly identified anomalies divided by the total number of true anomalies). For each method, the the PR curve is computed for each of the 50 test datasets, and its average PR curve are reported.

For the ROC curves, each method assigns a score to each test and normal dataset, where a higher score indicates that the dataset is believed to be more anomalous, and we measure how well the method ranks the test datasets (which contain anomalous groups) above the normal datasets (which do not contain anomalous groups). For each method, the algorithm is run on an equal number of datasets containing and not containing anomalies. The list of dataset scores returned by a method are sorted and iterated through: at each step, we compute the true positive rate (fraction of the 50 test datasets correctly identified as anomalous) and false positive rate (fraction of the 50 normal datasets incorrectly identified as anomalous). The ROC curve is reported for each method .

To compute the PR and ROC curves, each method must return a score for every record in each dataset, representing the anomalousness of that record. Both DAP and FGSS-MM find the top-k highest scoring disjoint subsets S, by iteratively finding the optimal S in the current test dataset and then removing all of the records that belong to this group; we repeat this process until we have grouped all the test data records. In this framework, a record R_i can only belong to one group, and thus the score of each record R_i is the score of the group of which it is a member. For all of the results described in this paper, we utilize the similarity-constrained search procedure (a maximum radius of $r = 1$ and an $\alpha_{\max}$ of 0.1) for both the DAP and FGSS-MM methods.

5.1 KDD Cup Network Intrusion Data

The KDD Cup Challenge of 1999 [10] was designed as a supervised learning competition for network intrusion detection. Contestants were provided a dataset where each record represents a single connection to a simulated military network environment. Each record was labeled as belonging to normal network activity or one of a variety of known network attacks. The 41 features of a record, most of which are continuous, represent various pieces of information extracted from the raw data of the connection. As a result of the provided labels, we can generate new, randomly sampled datasets either containing only normal network activity or normal activity injected with examples of a particular intrusion type. The anomalies from a given

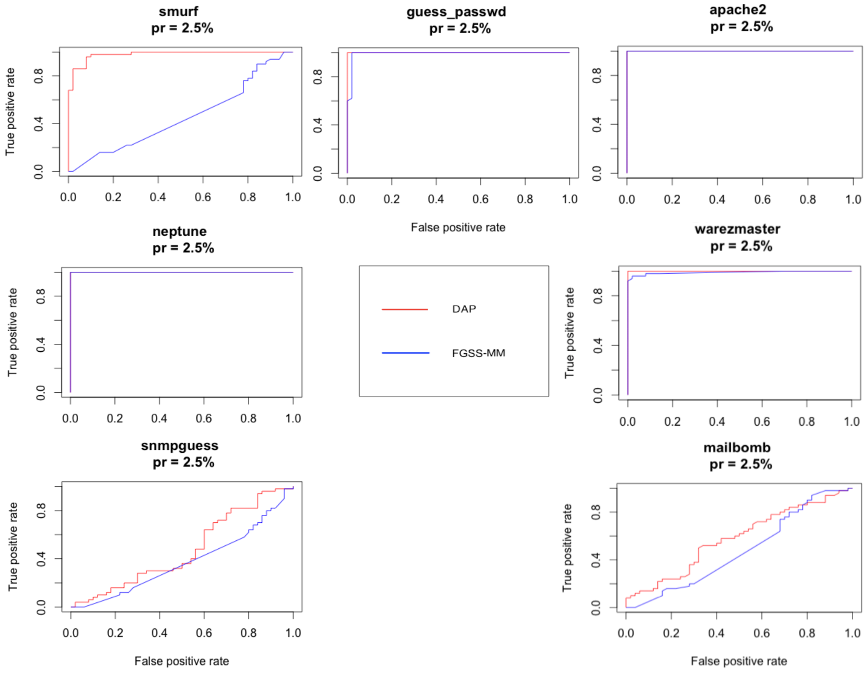

Figure 2: KDD Network Intrusion Data: ROC curves (measuring performance for distinguishing affected vs. unaffected datasets). Curves that are higher and more left correspond to better performance.

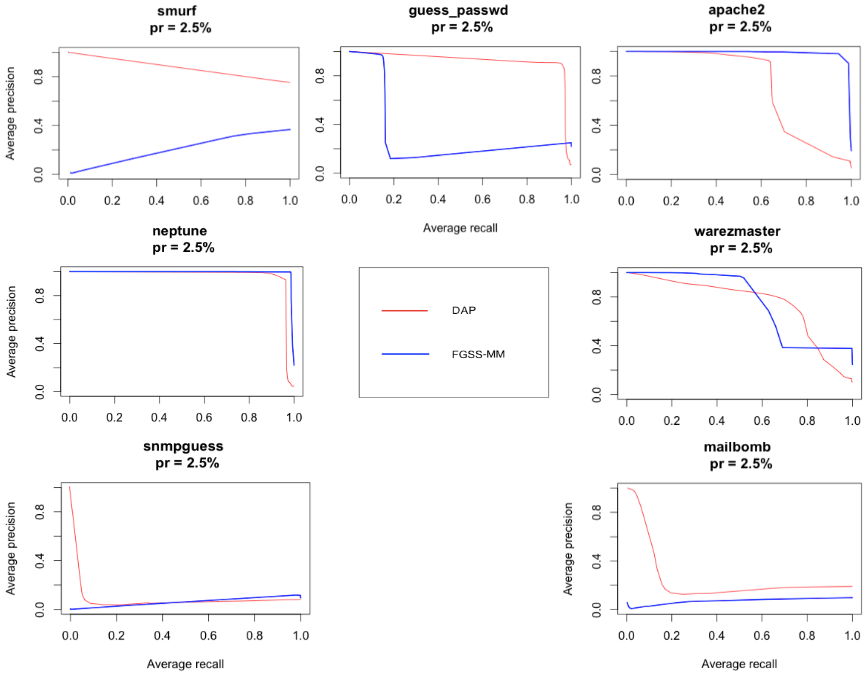

Figure 3: KDD Network Intrusion Data: PR curve (measuring performance for distinguishing affected vs. unaffected data records). Higher curves correspond to better performance.

intrusion type are likely to be both self-similar and different from normal activity, as they are generated by the same underlying anomalous process. These facts should make it possible to detect intrusions by identifying anomalous patterns of network activity, without requiring labeled training examples of each intrusion type.

We should note that there have been critiques of this KDD Cup network intrusion dataset; including the comprehensive critique offered by [12]. We acknowledge and agree with the concerns raised by [12], but explain why the results we show below are valid despite them. The general concern addressed in [12] is the transparency surrounding the consistency and veracity (realistic nature) of the synthetic data generating process. This general indictment is supported by specific examples: there is no mechanism to examine the realism of the background data and its inherent false alarm rate; real data is not well behaved, therefore the data generating process may not handle complications like fragmented packets; the reported data rates over the network appear to be low, based on the data generating scenario; the network sessions that included intrusion activity may not have also included a realistically amount of normal network activity, which could make detection of intrusions much easier; the proportion of each attack type may not resemble the proportion in a realistic network environment. To assess the relevance of these issues to our purposes, we must consider the data corpus's original intention. The data sponsor wanted the ability to develop a single empirical measurement from data, allowing for to evaluating and comparison of approaches on the task of network intrusion detection. Therefore, if the data corpus is found to fall prey to the above concerns, the performance (i.e., ability to identify intrusions and disregard background data) of a detection algorithm on this data corpus may be an inaccurate measure of its performance in the wild. However, the methods evaluated are created to be a general algorithms for identifying anomalous patterns, and are not necessarily designed to serve as an intrusion detection systems. Therefore, we are not necessarily making claims on how well the methods would serve if incorporated into intrusion detection systems in the wild, just on the specific challenge presented in the KDD Cup. Essentially, we require a scenario rich with patterns of anomalies and background data, to compare these methods' relative ability to identify the anomalous patterns. In order to construct the anomalous patterns we do not utilize the data as presented by the KDD Cup challenge, but instead sample from it to produce our own data sets conforming to the challenge of pattern discovery.

To create instances of the anomalous pattern discovery challenge we generate *test* datasets of size $N = 10000$, of which $pr = 2.5\%$ of its records are examples of each of the seven intrusion types; the remaining records $(N - (7 \times pr))$ are examples of normal network activity. For each of the five intrusion types, we generate *normal* datasets of size $N = 10000$, of which $pr = 2.5\%$ of its records are examples of each of the other six intrusion types. We also generate separate *training* datasets of up to $100,000$ records for the other six intrusion types and normal system activity. Finally, we must generate a training dataset for FGSS-MM that is a collection of all the records from these separate training datasets. This leave-one-out detection scheme will allow for generalizing ability of both methods to discovery novel anomalous patterns.

Additionally, [1, 9] notes that using all 41 features makes the anomalies very individually anomalous, such that any individual record anomaly detection method could easily distinguish these records from normal network activity. In this case, methods that search for groups of anomalies also achieve high performance, but the differences between methods are not substantial. Thus, following [1, 9], we use a subset of 22 features that provide only basic information for the connection, making the anomalies less obvious and the task of detecting them more difficult. We also use the same five common attack types as described by [9], and discretize all continuous attributes to five equal-width bins.

In Figure 2 and 3 respectively, we compare the ROC and PR curves for each of the seven leave-one-out detection scenarios. We observe that in each of the ROC scenarios and PR scenarios, our DAP algorithm experiences from comparable detection power to drastic improvements in detection power. These results are consistent with our understanding of the data and both methods, because by modeling a mixture of various intrusions and normal system activity FGSS-MM will frequently still consider subsets belonging to known data pattern as anomalous. However, the DAP produces a conservative estimate of the anomalousness of each subset it considers. This conservative estimates comes as a result of modeling each component of the mixture separately, and then mapping of each record to the model that minimizes its contribution to a subset's score. Thus, DAP only indicates a subset S as anomalous when even its conservative estimate of the score $F(S)$ is

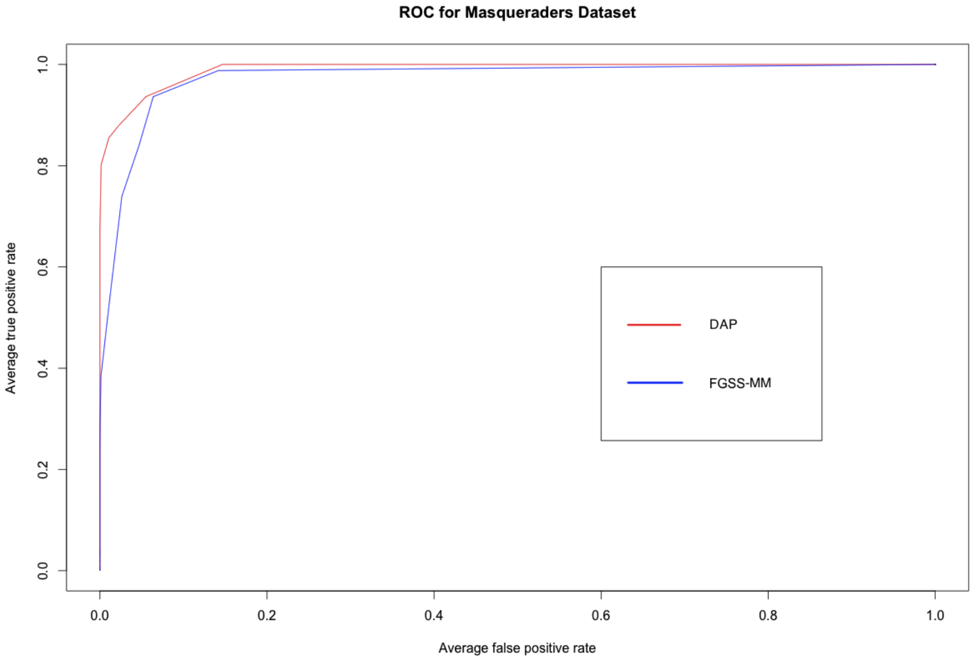

Figure 4: RUU Masquerade Detection Data: ROC curves (measuring performance for distinguishing affected vs. unaffected datasets). Curves that are higher and more left correspond to better performance.

high, which increases our confidence the true anomalous nature of S. Although, this conservative estimate increases the our confidence in the true nature of high scoring subsets, it resulted in discrepancy in detection ability between FGSS-MM and DAP for the apache and neptune PR scenario. Records corresponding to the these intrusions are either "extremely" or just "slightly" anomalous given normal system activity; these extremely anomalous records are more individually anomalous than any intrusion type considered. Both methods are able to easily detect the extremely anomalous apache2 and neptune records, but the DAP algorithm's conservative estimate of the subset containing the slightly anomalous apache2 and neptune records in the ranking. In this particular case, he FGSS-MM was rewarded for its possibly inflated estimate of the anomalousness of the subsets it considered. However, as overwhelming majority of these results demonstrate the DAP procedure maintains significantly higher power discover a novel anomalous pattern.

5.2 R-U-U (Are you you?) Masquerade Data

This R-U-U Masquerade dataset [19] was the result of supervised learning experiment for masquerade detection. The data was generated by creating a host sensor for Windows based personal computers which monitored and logged information concerning all registry-based activity, process creation and destruction, window GUI and file accesses, as well as DLL libraries' activity. For each of these activities, the sensor collects detailed process characteristics: process name and ID, the process path, the parent of the process, the type of process action (e.g. type of registry access, process creation, process destruction, window title change, etc.), the process command arguments, action flags (success or failure), registry activity results, and time stamp. All the information collected by each low-level action is summarized in nine attributes. To collect normal activity, this sensor was placed on the person computers of eighteen computer science students at Colombia University for on average four days. To collect masquerade data, the activity of forty computer science students were recorded as they participated in a 'capture the flag' exercise and individually followed

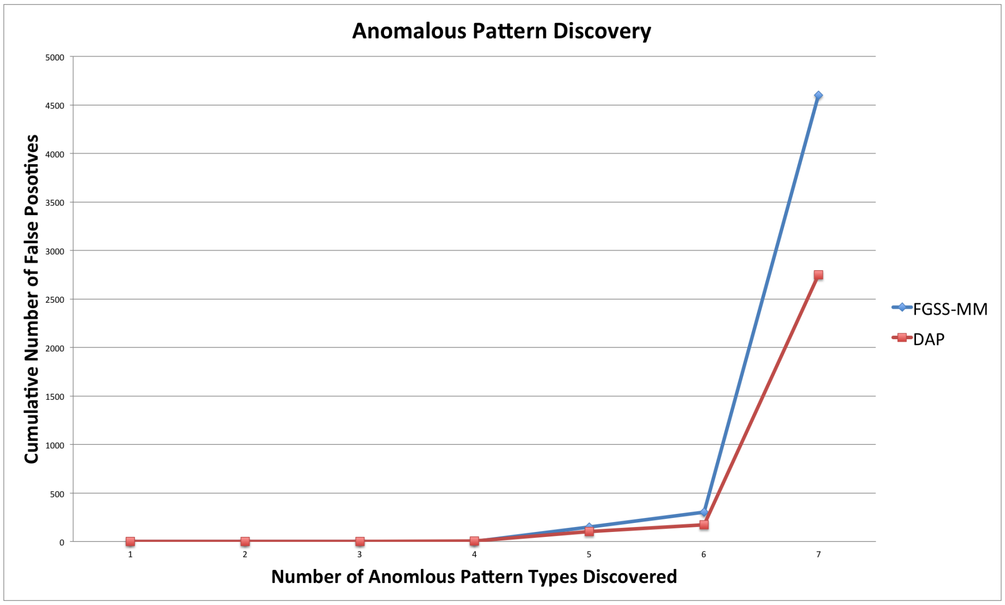

Figure 5: Anomalous Pattern Discovery: Measures how many false positives a method reports as it attempts to incrementally discover each novel anomalous pattern.

a particular malicious attacker scenario for fifteen minutes.

The challenge of masquerade detection in this particular scenario is captured by a systems ability to determine that someone other then a known user is utilizing the system, based on the low-level activity patterns generated. Therefore for each of the eighteen normal users we set-aside the first half of their activity data to learn models of activity patterns. We select two legitimate users and from the second half of the first user's activity data, we select twenty-five disjoint fifteen-minute chunks; all users and chunks of activity are chosen at random; these twenty-five chunks are used as our *normal* datasets. Next we select twenty-five masqueraders at random, and their fifteen-minutes of activity are used as our *test* datasets. We repeat this process 50 times, and compute the average ROC curves for the DAP and FGSS-MM methods found in Figure 4. For each iteration of this process, we must learn a mixture model of the two legitimate users by combining their training data.

Figure 4 demonstrates that both methods a very capable of distinguishing between the activity of a masquerader and one of two known users, although DAP has slightly better performance on average. We think this high performance was a result of two factors in the experiment: fifteen-minute provides more than enough information to make the masquerader's activity appear extremely anomalous and by only modeling two users FGSS-MM does not significantly compromised. Therefore, future work could measure each method's power a function of how many legitimate user are modeled and how much time the test datasets cover. We hypothesis, that as the number of legitimate users increase and the time represented in the test dataset decrease, FGSS-MM will experience a faster decay in power as compared to DAP.

5.3 Incremental Anomalous Pattern Discovery

As we described in Section 3, the real benefit of anomalous pattern discovery is the continual (incremental) discovery of novel anomalous patterns. Therefore using the Kdd Cup data from Section 5.1, we compare the ability of DAP and FGSS-MM to iteratively find a novel patterns as a function of how many false positives are detected at each iteration. Figure 5 shows the average results over 50 random test data sets created in the same manner as described in 5.1. What we see is that the DAP curve is a lower-bound on the FGSS-MM

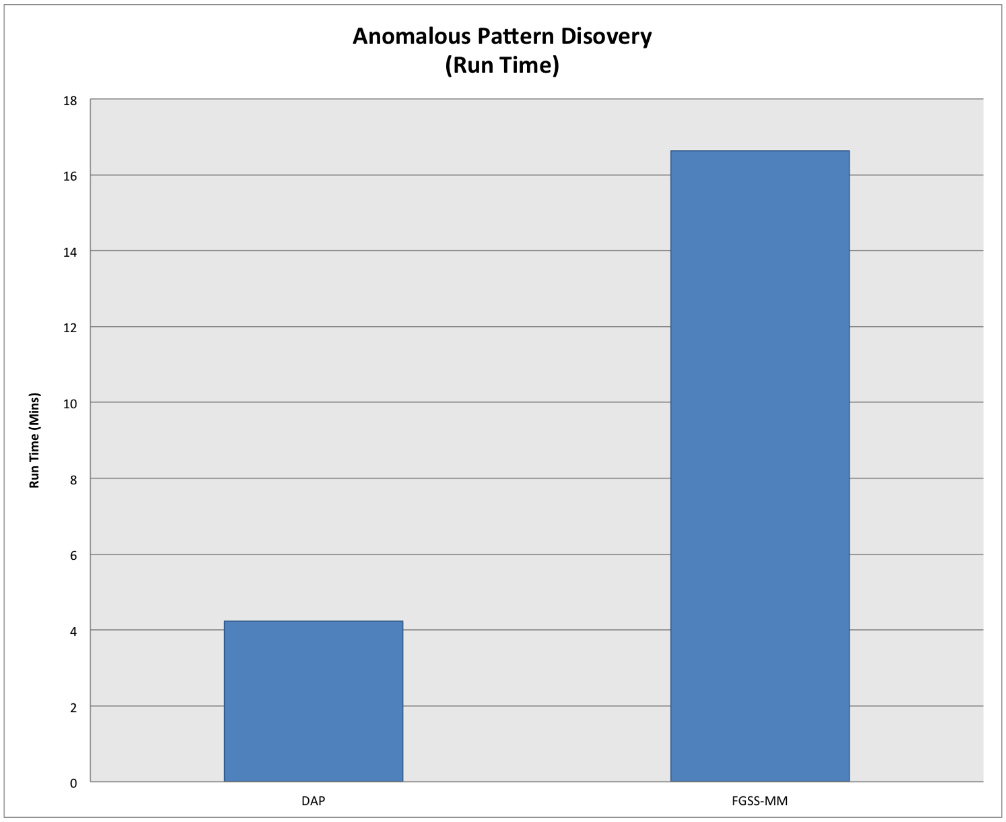

Figure 6: Anomalous Pattern Discovery Run Time: Measures how may minutes each algorithm required to accomplish the discovery task.

curve indicating that it demonstrates superior discovery ability. Additionally, we compare the run-times of these methods and observe in Figure 6 that DAP is significantly faster (approximately 4x), at the discovery task.

In addition to evaluating the performance of DAP to FGSS-MM, we can also measure how often the DAP algorithm finds the globally optimal subset. Recall that DAP iteratively optimizes over subsets of attributes, model mappings, and records until the subset scores no longer increases; additionally, for a given subset of attributes, the model mapping and records are optimal. Therefore, by exhaustively considering all subsets of attributes, and computing their subsequent optimal model mapping and records, we can discover the globally optimal subset. We define the *approximation ratio* as the largest value p such that the DAP algorithm achieves a score within $(100 - p)\%$ of the global maximum score (computed by exhaustively searching over the attributes) at least $p\%$ of the time. For example, an approximation ratio of 95% would signify that DAP achieves a score within 5% of the global maximum with 95% probability. Results were computed for values of $N \in \{10, 100, 10000\}$ and $M \in \{1, 2, 4, 8, 10, 12, 16\}$, considering 100 datasets for each combination of N and M, each randomly sampled from the KDD Cup normal network activity data. For values of $M > 16$, it was computationally infeasible to run the exhaustive search to completion. For each scenario, the DAP search achieved a approximation ratio of 100% (finding the exact global optimum for each of the 100 datasets we evaluated). These results empirically demonstrate that very little, if any, detection ability is lost when using the DAP algorithm to iteratively maximize over subsets of records and attributes.

6 Conclusion

This paper has presented significant contributions to the literature by presenting the challenge of anomalous pattern discovery. We formalize the pattern discovery problem as a search over subsets of data records and attributes given a collection of known pattern types, and present the Discovering Novel Anomalous Patterns (DAP) algorithm, which efficiently discovers anomalous patterns in general categorical datasets. From this formulation, we can demonstrate that the current state-of-the-art for anomalous pattern detection, i.e. the Fast Generalized Subset Scan (FGSS) [9], is a special case of the DAP method. The DAP algorithm utilizes a systematic procedure to map dataset values to an unbiased measure of anomalousness under each of the known pattern types, empirical p-value ranges. The algorithm then utilizes the distribution of these empirical p-value ranges under the known types in order to find subsets of data records and attributes that *as a group* significantly deviate from their expectation as measured by a nonparametric scan statistic. We demonstrate that by using a nonparametric scan statistics that satisfies the linear-time subset scanning property, we can search efficiently and exactly over all subsets of data records or attributes while evaluating only a linear number of subsets. These efficient optimization steps are then incorporated into an iterative procedure which jointly maximizes over subsets of records and attributes. Additionally, similarity constraints can be easily incorporated into our DAP framework, allowing for the detection of self-similar subsets of records which have anomalous values for some subset of attributes.

We propose a simple extension of FGSS, FGSS Mixture Model (FGSS-MM), and provide an extensive comparison between DAP and FGSS-MM on real-world two computer system intrusion datasets: network intrusion detection and masquerade detection. Both settings were motivated by the necessity to maintain models of multiple known pattern types. In network intrusion data, DAP learned a model to describe known and benign activity as well as a known and malicious activity, with the ultimate goal of discovery novel malicious activity. In the masquerader data DAP learned separate models of activity from two system users, both legitimate, with the ultimate goal of identifying when a masquerader was utilizing the system based solely on the activity generated. In the network intrusion detection setting DAP consistently outperforms the FGSS-MM, while in the masquerade detection setting DAP provides a slight increase in power to detect. DAP also demonstrates improved ability to continually discover novel anomalous patterns, the essential goal of this work.

In future work, we plan to extend DAP to better model real-valued attributes. DAP can only handle categorical attributes, which forces it to discretize real-valued attributes when evaluating mixed datasets. This constraint only exists because our current method for obtaining record-attribute likelihoods, modeling the conditional probability distribution between attributes with a Bayesian Network and using Optimal Reinsertion [14] to learn the network structure, can only handle categorical attributes. By discretizing real-valued attributes, we may lose vital information that would make the task of discovery anomalous patterns easier. Therefore we are currently investigating extensions of DAP which better exploit the information contained in real-valued attributes. We believe that augmenting a Bayesian Network, learned only from the categorical attributes, with a regression tree for each continuous attribute will increase the power of DAP to detect patterns.

Acknowledgements

This work was partially supported by the National Science Foundation, grants IIS-0916345, IIS-0911032, and IIS-0953330. Additionally the author was directly supported by an NSF Graduate Research Fellowship (NSF GRFP-0946825) and an AT&T Labs Fellowship.

References

[1] K. Das. Detecting patterns of anomalies. Technical Report CMU-ML-09-101, Ph.D. thesis, Carnegie Mellon University, Department of Machine Learning, 2009.

[2] K. Das and J. Schneider. Detecting anomalous records in categorical datasets. In *Proc. of the 13th ACM SIGKDD Conf. on Knowledge Discovery and Data Mining*, pages 220–229, 2007.

[3] K. Das, J. Schneider, and D. B. Neill. Anomaly pattern detection in categorical datasets. In *Proc. of the 14th ACM SIGKDD Conf. on Knowledge Discovery and Data Mining*, 2008.

[4] D. Donoho and J. Jin. Higher criticism for detecting sparse heterogeneous mixtures. *Annals of Statistics*, 32(3):962–994, 2004.

[5] J. He and J. G. Carbonell. Nearest-neighbor-based active learning for rare category detection. In *Proceedings of the Twenty-First Annual Conference on Neural Information Processing Systems*, 2007.

[6] J. He and J. G. Carbonell. Rare class discovery based on active learning. In *International Symposium on Artificial Intelligence and Mathematics*, 2008.

[7] J. He and J. G. Carbonell. Prior-free rare category detection. In *Proceedings of the SIAM International Conference on Data Mining*, pages 155–163, 2009.

[8] J. He and J. G. Carbonell. Co-selection of features and instances for unsupervised rare category analysis. In *Proceedings of the SIAM International Conference on Data Mining*, pages 525–536, 2010.

[9] E. M. III, S. Speakman, and D. B. Neill. Fast generalized subset scan for anomalous pattern detection. *Journal of Machine Learning Research*, 14(June):579–593, 2013.

[10] KDDCup. The 3rd international knowledge discovery and data mining tools competition. In *5th ACM SIGKDD Conf. on Knowledge Discovery and Data Mining*, pages 220–229, 1999.

[11] A. Lazarevic, L. Ertoz, V. Kumar, A. Ozgur, and J. Srivastava. A comparative study of anomaly detection schemes in network intrusion detection. *In Proceedings of the SIAM International Conference on Data Mining.*, 2003.

[12] J. McHugh. Testing intrusion detection systems: a critique of the 1998 and 1999 darpa intrusion detection system evaluations as performed by lincoln laboratory. *ACM transactions on Information and system Security*, 3(4):262–294, 2000.

[13] J. Mchugh. Intrusion and intrusion detection. *International Journal of Information Security*, 1:14–35, 2001.

[14] A. W. Moore and W.-K. Wong. Optimal reinsertion: A new search operator for accelerated and more accurate bayesian network structure learning. In *Proceedings of the 20th International Conference on Machine Learning*, pages 552–559. AAAI Press, 2003.

[15] B. Mukherjee, T. L. Heberlein, and K. N. Levitt. Network intrusion detection. *Network, IEEE*, 8(3):26–41, 1994.

[16] D. B. Neill. Fast subset scan for spatial pattern detection. *Journal of the Royal Statistical Society (Series B: Statistical Methodology)*, 74(2):337–360, 2012.

[17] D. B. Neill, G. F. Cooper, K. Das, X. Jiang, and J. Schneider. Bayesian network scan statistics for multivariate pattern detection. In J. Glaz, V. Pozdnyakov, and S. Wallenstein, editors, *Scan Statistics: Methods and Applications*, 2008.

[18] D. Pelleg and A. Moore. Active learning for anomaly and rare-category detection. In *Advances in Neural Information Processing Systems 18*, pages 1073–1080. MIT Press, 2004.

[19] M. B. Salem. Towards effective masquerade attack detection. Technical report, Ph.D. thesis, Colombia University, Graduate School of Arts and Sciences, 20011.

[20] M. Zamani. Machine Learning Techniques for Intrusion Detection. *ArXiv e-prints*, Dec. 2013.