The Application of Statistical Relational Learning to a Database of Criminal and Terrorist Activity*

B. Delaney†, A. Fast‡, W. Campbell†, C. Weinstein†, D. Jensen‡

Abstract

We apply statistical relational learning to a database of criminal and terrorist activity to predict attributes and event outcomes. The database stems from a collection of news articles and court records which are carefully annotated with a variety of variables, including categorical and continuous fields. Manual analysis of this data can help inform decision makers seeking to curb violent activity within a region. We use this data to build relational models from historical data to predict attributes of groups, individuals, or events. Our first example involves predicting social network roles within a group under a variety of different data conditions. Collective classification can be used to boost the accuracy under data poor conditions. Additionally, we were able to predict the outcome of hostage negotiations using models trained on previous kidnapping events. The overall framework and techniques described here are flexible enough to be used to predict a variety of variables. Such predictions could be used as input to a more complex system to recognize intent of terrorist groups or as input to inform human decision makers.

1 Background and Motivation

During the last decade, there has been an increasing effort toward data collection on criminal and terror networks using open source materials (e.g. news articles, police reports, and court documents.) A straightforward use of such data includes manual analysis of groups and individuals involved in nefarious activity to inform key decision makers tasked with preventing future bombings or other violent attacks. However, if the collection is detailed with specific annotations including continuous variables and categorical fields, the application of statistical machine learning becomes possible. An example of such an analysis is shown in [1], where the author used statistical methods to indentify extremist groups responsible for surprise terror attacks. By modeling past behavior, statistical techniques can help find large scale patterns in the data and possibly be used to prevent or inform future activities. This paper investigates the use of statistical machine learning to predict individual attributes and event outcomes from a graphical representation of a relational database of terrorist activity.

We apply statistical relational learning algorithms to predict leadership roles of individuals in a group based on patterns of activity, communication, and individual attributes. Using labeled training data, we apply supervised learning to build a model which describes the structures and patterns of leadership roles. The relational model returns a probability that a particular person is in a leadership role given a graphical representation of the individuals activities and attributes. A held out test set is used for evaluation and receiver operator curves (ROC) for correct prediction of leadership is presented. A more complex model is applied to give improved performance in a more realistic "data poor" test condition. Such features can be important components of an overall automatic threat detection system such as the one presented in [2]. In such a system, automatic identification of individual roles and activities from basic features can help infer intent of groups and individuals through higher-level pattern recognition and social network analysis.

In addition to predicting attributes of individuals, we use the relational model to predict the outcome of an event, in this case, the fate of a hostage in a kidnapping event. Given a particular hostage taking event, the system will be able to predict the probability that the hostage will be released or killed based on known properties of the event. Features in the this model might include ransom demands and payment, regions and countries of the event, hostage nationality, and groups or individuals involved along with their past activities. Each of these features indicates the likelihood that a successful hostage release can be negotiated. The aggregration of relational features such as the percentage of hostages released by similar groups in the past can be used to improve performance. Aggregation

*This work was sponsored by the Department of Defense under Air Force Contract FA8721-05-C-0002. Opinions, interpretations, conclusions, and recommendations are those of the authors and are not necessarily endorsed by the United States Government.

†MIT Lincoln Laboratory, Information Systems Technology Group

‡University of Massachusetts Amherst, Knowledge Discovery Laboratory. A. Fast now at Elder Research Inc.

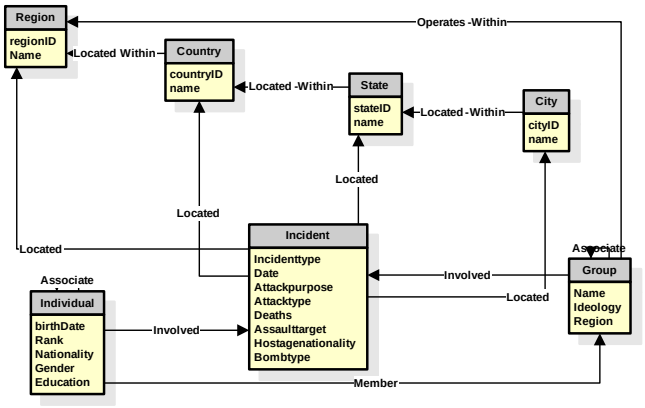

Figure 1: Sample ISVG database schema.

of traditional social network analysis features such as betweenness and degree are also predictive of sucessful hostage relase. Such a model of hostage fate can help decision makers better understand how extremist groups operate and provide key indicator variables that are highly correlated to a positive release outcome. We compare our results on the hostage release model with previously published results on the subject and show improvements over the standard statistical classification approach.

1.1 ISVG Database The Institute for the Study of Violent Groups (ISVG) is a research group that maintains a database of terrorist and criminal activity from open-source documents, including news articles, court documents, and police reports [3]. The database scope is worldwide and covers all known terrorist and extremist groups as well as individuals and related funding entities. The original source documents are contained in the database along with over 1500 carefully hand annotated variable types and categories. These variables range from free text entries to categorical fields and continuous variables. Associations between groups, individuals, and events are also included in the annotation. There are over 100,000 incidents, nearly 30,000 individuals, and 3,000 groups or organizations covered in the database. The data is continually updated, but the version we used covered incidents up until April, 2008.

1.2 Graphical Schema For our purposes, we selected a subset of the overall database schema that contained most of the categorical fields and continuous variables available in the database. A continuous variable may consist of a date, age, number of casualties or other such variables represented by a single number. Categorical fields generally represent the type of a particular object in the database. This may include incident types (bombing, armed assault, kidnapping, etc.) or specific information about weapons or bombs used in an attack. We have represented a sample of this schema graphically in Figure 1. Each node represents an object type, and the text below the object type represents the available attribute fields for each object. For example, *individuals* have a "birthdate," "nationality," "gender," etc. Objects are linked via a specific link type indicated by the text on the line. At the center of the schema is the incident and groups and individuals connect to particular incidents via their involvement. The entire data set consists 9 different node types, 90 different attribute types, and 11 link types. The actual instantiation of the graphical database contains over 180,000 nodes with over 2 million attributes spread across the 90 attribute types. Nodes are connected by over 1 million links.

1.3 Graphical Query Once the database is represented as a graph with nodes, edges, and attributes, we use the QGraph software to pull selected subgraphs from the larger database for analysis. QGraph is a graphical query language designed for querying large relational data sets, such as social networks [4]. Queries are specified visually by drawing the structure of the desired matches and adding annotations to that structure to further refine the query. Matches are returned as subgraphs which are small subsections of the overall data containing only the desired structure.

2 Methods and Technical Solutions

Classification uses statistical methods to predict the status of an (unknown) characteristic, or feature, of a particular entity given a set of observed characteristics also on the entity. Most classification algorithms assume data are independent and identically distributed (i.i.d.). However, due to the connections inherent in social, technological, and communication networks, data arising from these sources does not meet either of these conditions. For example, in criminal networks, known associates of convicted criminals are likely to be criminals as well (non-independent) and some criminals have many more associates than others (heterogeneous). Furthermore, network data often exhibits autocorrelation among class labels of related instances [5]. The concept of autocorrelation, sometimes called homophily, is best summarized by the phrase "birds of a feather flock together" indicating that individuals with similar characteristics tend to be related. Failure to account for non-independence and heterogeneity in network data, can lead to biases in learned models when using traditional

approaches for classification [6, 7]. While traditional classification algorithms can incorporate relational features, an exhaustive aggregation of relational features becomes less efficient as the data set becomes large. Even with the incorporation of relational features, the standard classification approach still makes predictions for each instance that are independent, making collective classification more difficult.

2.1 Overview of Statistical Relational Learning

Statistical relational learning (SRL) is a sub-discipline of the machine learning and data mining communities [8]. As its name implies, the focus of SRL is extending traditional machine learning and data mining algorithms for use with data stored in multiple relational tables, as typically occurs in a relational database such as MySQL, or Oracle. This storage model permits analysis of data that are non-independent and heterogeneous such as social network data. The primary focus of this work is classification in social network data. For example, in criminal networks we are often interested in predicting a binary variable indicating whether a particular individual will commit a crime in the near future. The true value of this variable is generally unknown at the time of analysis, however, there are a number of observable features that are predictors such as whether the individual or a closely related individual has committed a crime in the past, recently filed bankruptcy, or lost their job etc. Tools developed in the SRL community extend the traditional classification paradigm to include features on both the individual in question and features on individuals related through social or organizational ties.

In addition to using features on related individuals, social network data also provides the opportunity for collective classification. Collective classification is possible when many individual class labels (e.g., future crime status) are unknown, but are connected via social or organizational ties. These relations among individuals permit the predictions about one individual to propagate to predictions about related individuals. Collective approaches, which infer the value all unknown labels simultaneously, have been shown to yield higher accuracies than non-collective models, particularly when the labels of related instances exhibit autocorrelation [9]. Thus, collective classification is widely studied within the field of SRL [10]. In the following sections, we describe two specific SRL techniques for classification in relational data: the relational probability tree and relational dependency network.

2.2 Relational Probability Trees

The Relational Probability Tree (RPT) is a probability estimation tree for classification in relational domains [11]. A probability estimation tree is a conditional model similar to a classification tree, however, the leaves contain a probability distribution rather than a class label assignment [12]. To account for non-independence in network data, the RPT is designed to use both intrinsic features on the target individual and relational features on related individuals. However, due to heterogeneity in the data, the number of relational features can vary from individual to individual. To account for possible heterogeneity, the RPT is designed to automatically construct features by searching over possible aggregations of the training data. The RPT applies standard aggregations COUNT, AVERAGE, MODE, etc. to dynamically flatten the data before selecting features to be included in the model. To find the best feature, the RPT searches over values and thresholds for each aggregator. For example, if we are aggregating over criminal activity of an individual then an appropriate feature might be [COUNT(CriminalActivity.type=Larceny) > 1] where the type and number are determined by the algorithm. The RPT has been used successfully to predict high-risk behavior in the securities industry in the United States using the social network among individuals in the industry [13, 14].

2.3 Relational Dependency Networks

The Relational Dependency Network (RDN) is joint relational model for performing collective classification [15]. An RDN is a pseudolikelihood model consisting of a collection of conditional probability distributions (CPDs) that have been learned independently from data. The CPDs used in a dependency network are often represented by probability estimation trees, although any conditional model suffices [16]. For our purposes, we use an RDN consisting of a set of individually-learned RPTs for each attribute that are combined into a single, joint model of relational data. Inference (prediction) in the RDN is accomplished using Gibbs sampling, a technique that relies on repeated sampling from conditional distributions [17]. The RDN can represent autocorrelation relationships, and was the first joint model that permitted learning autocorrelation relationships from data. Collective classification is performed via inference using multiple iterations of Gibbs sampling, whenever relational features are included in the learned trees.

3 Empirical Evaluation

We performed a number of experiments using the ISVG relational database. Each of these experiments requires a labeled set of subgraphs to be used for training of the relational models and another, non-overlapping set, for evaluation. Using the QGraph software,

we can construct appropriate queries to return these subgraphs and randomly divide them into training and test sets using 4-fold cross validation. We present ROC performance curves after applying the learned model to the evaluation set on one fold of the randomly selected data.

Our first experiment involves the prediction of leadership attributes of individuals within a group. The ISVG data contains a rich set of individual roles. Here, we have binned them into categories pertaining to leadership (e.g. field commander, cell leader, spiritual leader, etc.) and non-leadership (e.g. group member, aide, activist, etc.), resulting in a binary classification. We apply different learning techniques, highlighting the differences in data-rich vs. data poor operational conditions.

Next, we consider predicting the outcome of a simple event, in this case, the fate of a hostage in a kidnapping incident. Once again, we have a binary classification, where the hostage is ultimately released or is eventually killed. We do not consider incidents where the hostage is still being held or whose fate is otherwise unknown. Our results on the hostage fate prediction task are compared to previously published methods.

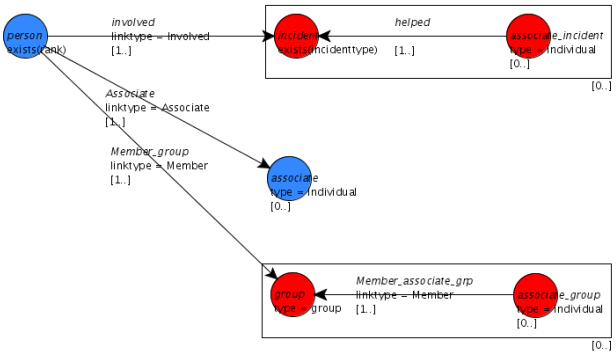

Figure 2: Graphical query for leadership prediction.

3.1 Leadership Role Prediction The ISVG database contains 3854 individuals with labeled roles. As discussed earlier, we binned these roles into leadership and non-leadership categories. We use QGraph to extract relevant subgraphs for training and testing. There are 2890 randomly selected subgraphs for training and 964 for evaluation. The query is shown in Figure 2. The query looks for persons with a leadership attribute and all of their associates, including those related through the same group or incident as well as the groups and incidents themselves. The query in Figure 2 contains two sub-queries (rectangular boxes) that look for zero or more incidents and groups along with individuals involved in the incident or members of the group. In this way, we expand the total number of related individuals beyond what the ISVG annotator labeled in the "Associate" link. Additionally, the incident attributes and group type may indicate an organizational structure to help predict leadership.

An example result of this graphical query is shown in Figure 3. The first experiment assumes a data rich condition where full information about neighboring associates is known (e.g. age, education level, nationality, etc.) All of this information is used in the RPT model to predict whether or not the node under consideration is in a leadership role. Under a more realistic assumption, this information may not be known (as shown in Figure 4.) In this case, we may observe a pattern of ac-

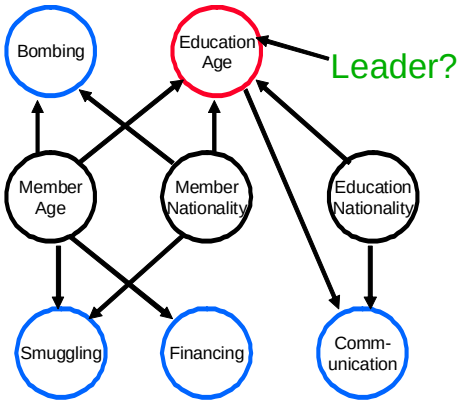

Figure 3: Ideal case of relational classification where many neighboring attributes are known.

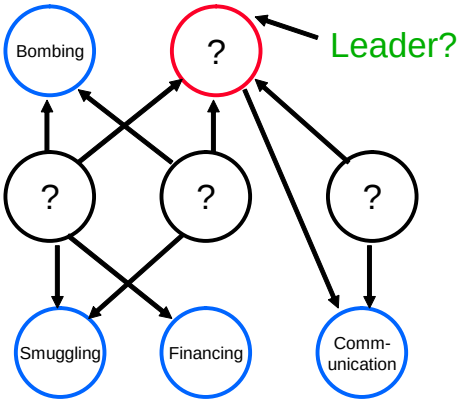

Figure 4: Realistic data condition where attributes of associates are unknown.

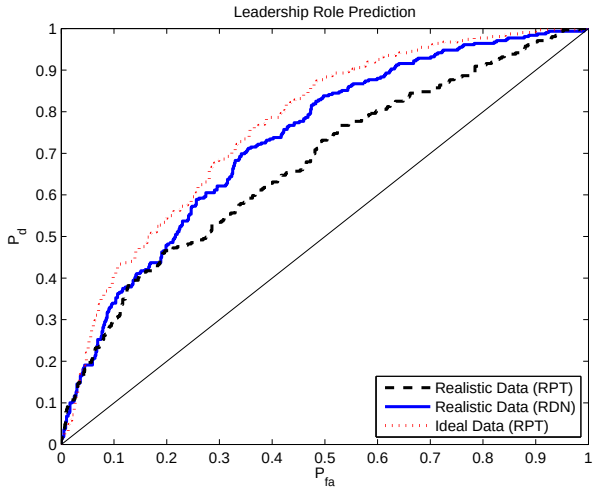

Figure 5: Performance of social network leadership prediction models.

Condition	AUC
RPT Realistic Data	0.6725
RDN Realistic Data	0.7314
RPT Ideal Data (upper bound)	0.7672

Table 1: AUC performance of social network leadership prediction models.

tivity and communication but know very little about the actors involved and wish to determine who the leader is.

The results for both of these RPT models on the held out evaluation set are shown in Figure 5 and Table 1. Figure 5 is a receiver operator characteristic (ROC) curve, where the probability of detection is plotted on the Y-axis and the probability of false alarm is plotted on the X-axis. In this instance, a correct detection is when the system correctly predicts a leadership role for an individual. A false alarm occurs when the system predicts a positive leadership label for an individual who is not in a leadership role. Table 1 shows the area under each ROC curve. The dotted line labeled "Ideal Data (RPT)" represents the query result from Figure 3, where all information about associates are known. This represents an upper bound on performance should we know all information about the individuals, including the leadership characteristics of the associates. The dashed line labeled "Realistic Data (RPT)" represents the more realistic condition where specific attributes of associates are hidden from the RPT model. The former represents the data rich case and gives the best performance. In more realistic data conditions the performance is significantly worse. The use of relational dependency networks can be used to improve performance in the latter, data poor, condition.

3.1.1 Collective Classification As discussed in Section 2.3, RDNs can perform collective classification where several attributes are estimated simultaneously based on a joint probability model across selected variables. In this experiment, individual conditional models (RPTs) are built for each distinguishing attribute of the associate. These include the following variables *leader*, *status*, *education*, *nationality*, *race*. Multiple Gibbs sampling iterations are used to approximate the joint distribution. The results are shown in the solid line labeled "Realistic Data (RDN)" in Figure 5. The RDN results are almost as good as the upper bound "data rich" condition of the original RPT model. This result is promising as it becomes possible to predict leadership roles with some degree of accuracy in situations where very little specific information is known about the individual actors.

Additional insight into the data can be learned from the relational dependency network diagram in Figure 6. The interpretation of the RDN is that of a relational extension of a "dependency network", a type of model in which arcs between variables indicate strict dependence – rather than the more complex encoding of independence that the arcs in a Bayes net indicate. The large colored boxes (plates) represent entities. White circles indicate variables on those entities and arrows indicate dependence. The structure has been learned automatically from data, with RPTs underlying the individual attributes in the RDN. The model shows

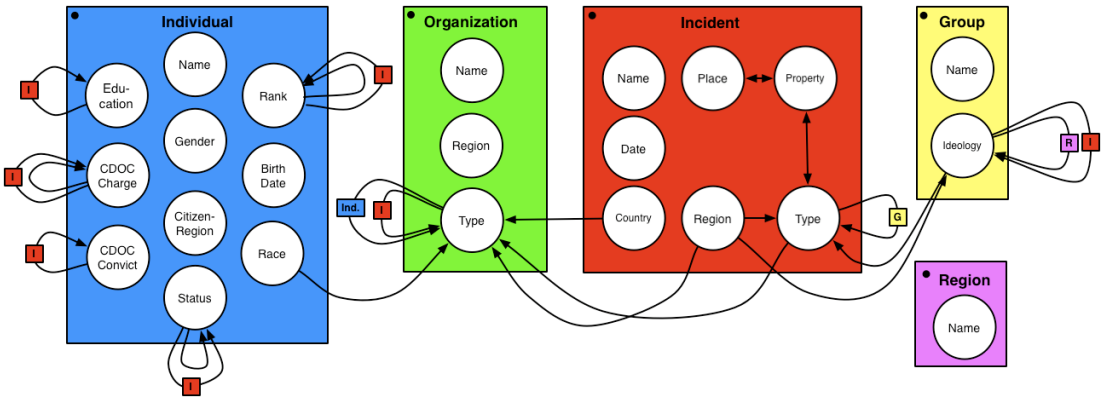

Figure 6: Relational dependency network on selected ISVG variables.

three kinds of dependencies:

1. Dependence between variables on the same entity.
   - The region of an incident and the incident type are dependent on each other.
2. Dependence between variables on different entities.
   - The type of an organization depends on the country of incidents to which the organization has been tied.
3. Autocorrelation (special case of 2) - dependence on the same attributes across different entities.
   - Incident type is autocorrelated through groups (i.e., groups tend to be involved with the same types of incidents.)
   - CDOC (court document) charge and CDOC convictions of individuals are both autocorrelated through incidents. Individuals involved in the same incidents tend to have the same charges and conviction status.

Autocorrelation through an intermediate entity is indicated by a colored box on the self loop. Individuals are autocorrelated through incidents, indicating the individuals with similar attributes tend to be involved in the same incidents. Self loops with no box indicate direct linkage via known associates. Individuals with the same status (incarcerated, at large, etc.) tend to be directly associated.

3.2 Prediction of Successful Hostage Release

Predicting the fate of a hostage in a kidnapping event requires a labeled set of kidnapping incidents where the event outcome (released or killed) is known. Each event contains a set of additional possible attributes including the presence or absence of ransom negotiations, payment of ransom, country of the event, date, nationality of the hostages, length of captivity, and other related attributes. Each kidnapping event in the database is also connected to many other incidents, groups, and individuals and is accessed via a graphical query. The query looks for related incidents that are linked to the kidnapping event, individuals and groups that are part of the kidnapping as well as other kidnapping events perpetrated by the same group in the past. All of these related nodes contain additional information which will help predict the outcome of the kidnapping event.

The query results in 1519 kidnapping events with known outcomes. These are split into non-overlapping sets of 1139 incidents for training and 380 incidents for

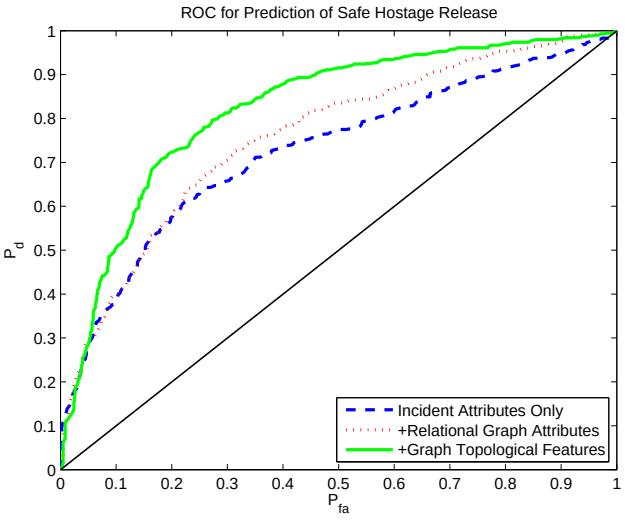

Figure 7: Performance of hostage fate prediction.

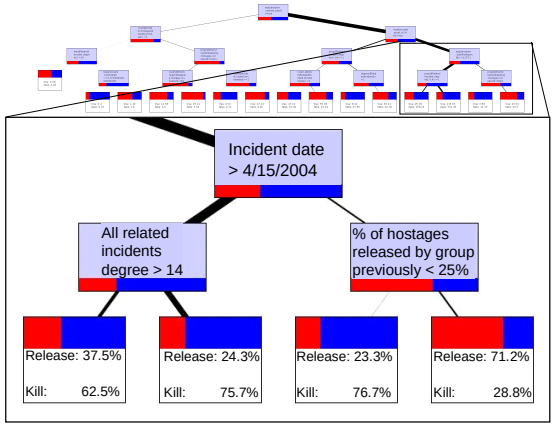

Figure 8: A sample of the hostage fate prediction RPT.

Condition	AUC
Incident Attributes Only	0.7266
+Relational Graph Attributes	0.7592
+Graph Topological Features	0.8228

Table 2: AUC performance of hostage fate prediction.

evaluation. We trained a series of relational probability trees on data with an increasing number of nodes and attributes. First we used only attributes from the core incident and no relational features. We added relational features, including information about related incidents, groups, and individuals. The final system used some additional features derived exclusively from the graph topology based on some well known social network analysis features. Specifically, we calculated *betweenness*, *degree*, *coreness*, and *constraint* for each node in the graph and allowed the RPT training algorithm to choose these as features. The results of each of these experiments is shown in Figure 7, and the area under the curve is shown in Table 2. Once again, an ROC curve is plotted with probability of false alarm on the X-axis and probability of detection on the Y-axis. A correct detection is when the hostage is released when the model predicts that outcome.

The dashed "incident attributes only" line depicts the performance when only the core kidnapping incident features are included. Most of the predictive power of the model is contained within the core incident features. The dashed line shows the performance when additional relational features are added to the model. Finally, the solid line demonstrates the performance when the social network analysis metrics are added to the features. This results in a fairly large improvement in performance, although it is not quite clear why this is the case. Overall, the final system is able to correctly detect hostage release just above 70% of the time at a 20% false alarm rate. Clearly this is not a template for hostage negotiation as the stakes are quite high, but the model itself can shed light on the predictive power of certain variables. Figure 8 shows an example of an RPT model zoomed in on a selection of leaf nodes. Among the predictive variables chosen by the RPT are the percentage of hostages released by this particular group in previous incidents. Violent groups that use kidnapping as a means of fund-raising are generally highly motivated to release hostages in exchange for money and will likely continue to do so.

This idea of hostage fate prediction was originally discussed in [18], where logistic regression was applied to a hand selected set of variables from the ISVG database. Here we have recreated those results and applied the same logistic regression analysis on the training set to provide a series of weights which are then used on the held out test set. A series of binary values are used in the regression, including presence or absence of ransom demands, nationality of hostage (foreign/domestic), date of incident (before or after Iraq war), and length of captivity (less/more than three days). Missing data are treated as positive outcomes in the logistic regression evaluation. The results of the logistic regression vs. the RPT model are plotted in Figure 9 using 4-fold cross validation. The solid line represents the RPT performance. While the logistic regression performs well in each data fold, the RPT model outperforms it in each instance. This is largely

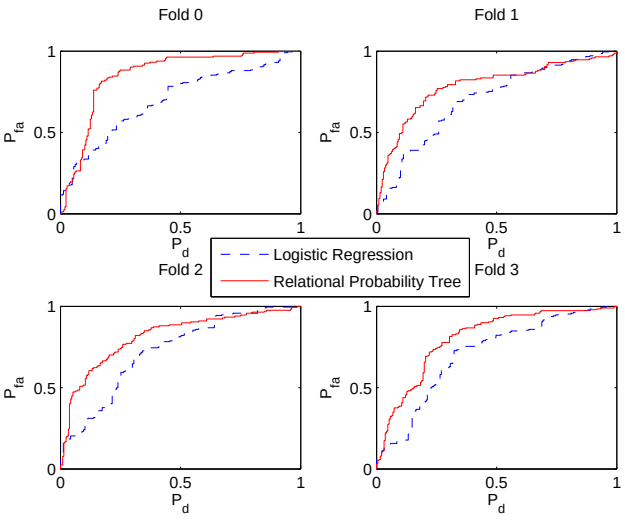

Figure 9: Comparison of RPT model to logistic regression with 4-fold cross validation.

due to the additional features available to the RPT and its flexibility in dealing with missing data.

4 Significance and Impact

In this paper, we have highlighted a series of experiments using the ISVG database of terrorist activity. The predictive models that we built are not inclusive, but rather suggest a framework that could be used to predict a variety of variables of interest. Many other variables exist in the ISVG data, and models such as RPTs or RDNs could be used to fill in gaps where data collection was not as complete. Prediction or inference could also be used to quickly label new data and aid decision makers in taking appropriate and immediate action against violent groups. Additionally, the models themselves could be useful as they indicate variables or phenomena in the data which are not necessarily intuitive or obvious. For example, the RDN in Figure 6 indicates that individuals with similar attributes tend to participate in the same incidents. This may not be immediately obvious from a cursory look at a particular set of incidents.

The probabilistic nature of the predictions lends itself for use in a late stage pattern recognition system which can infer the intent of terrorist groups. Such a system is described in [2], where "transactions" automatically derived from multimedia content are used to build social networks and infer intent. In this context, the probability that a particular individual is a leader or that a group subscribes to a particular ideology could help in the overall "intent recognition" process. Our foray into prediction of hostage fate is once such example of intent recognition but with an admittedly simplistic set of conditions.

In the future, we hope to apply these and more advanced techniques to a more regional set of questions such as the prediction of hot-spot areas for terrorist activity. Such prediction may require the integration of additional data sources and new technical methodology. With appropriate amounts and quality of data, we may begin to infer cause and effect relationships [19] which can provide decision makers with information beyond simple correlation.

5 Acknowledgments

Portions of this analysis were conducted using Proximity, an open-source software environment developed by the Knowledge Discovery Laboratory at the University of Massachusetts Amherst (`http://kdl.cs.umass.edu/proximity/`). The authors would like to thank Dr. Richard Ward, Dean of the Henry C. Lee College of Criminal Justice and Forensic Sciences at the University of New Haven, and Dr. Daniel Mabrey, Director of IT/Analysis at the Institute for the Study of Violent Groups at the University of New Haven for providing support and guidance with the ISVG database. More information about the Institute for the Study of Violent Groups can be found at `http://www.isvg.org`.

References

[1] D.J. Mabrey, *Tactical terrorism analysis: A comparative study of statistical learning techniques to predict culpability for terrorist bombings in two regional low-intensity conflicts (Iraq, Israel/Palestine)*, Ph.D. thesis, SAM HOUSTON STATE UNIVERSITY, 2006.

[2] C. Weinstein, W. Campbell, B. Delaney, and J. O'Leary, "Modeling and detection techniques for counter-terror social network analysis and intent recognition," in *Proceedings of the IEEE Aerospace Conference*, 2009.

[3] Sam Houston State University, Huntsville, TX, *Institute for the Study of Violent Groups Codebook*, 2007.

[4] H. Blau, N. Immerman, and D. Jensen, "A visual language for querying and updating graphs," Tech. Rep. 2002-37, University of Massachusetts, 2002.

[5] M. McPherson, L. Smith-Lovin, and J.M. Cook, "Birds of a feather: Homophily in social networks," *Annual review of sociology*, vol. 27, no. 1, pp. 415–444, 2001.

[6] D. Jensen and J. Neville, "Linkage and autocorrelation cause feature selection bias in relational learning," in *Proceedings of the 19th International Conference on Machine Learning*. 2002, pp. 259–266, Morgan Kauffman.

[7] David Jensen, Jennifer Neville, and Michael Hay, "Avoiding bias when aggregating relational data with degree disparity," in *Proceedings of the 20th International Conference on Machine Learning*, 2003.

[8] L. Getoor and B. Taskar, *Introduction to statistical relational learning*, The MIT Press, 2007.

[9] D. Jensen, J. Neville, and B. Gallagher, "Why collective inference improves relational classification," in *Proceedings of the 10th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2004.

[10] P. Sen, G. M. Namata, M. Bilgic, L. Getoor, B. Gallagher, and T. Eliassi-Rad, "Collective classification in network data," Tech. Rep. CS-TR-4905, University of Maryland, College Park, 2008.

[11] J. Neville, D. Jensen, L. Friedland, and M. Hay, "Learning relational probability trees," in *Proceedings of the 9th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining.*, 2003.

[12] Foster Provost and Pedro Domingos, "Tree induction of probability-based ranking," *Machine Learning Journal*, vol. 52, no. 3, 2002.

[13] Andrew Fast, Lisa Friedland, Marc Maier, Brian Taylor, David Jensen, Henry G. Goldberg, and John Komoroske, "Relational data pre-processing techniques for improved securities fraud detection," in *The Proceedings of the 13th International Conference on Knowledge Discovery and Data Mining*, 2007.

[14] Jennifer Neville, Ozgur Simsek, David Jensen, John Komoroske, Kelly Palmer, and Henry Goldberg, "Using relational knowledge discovery to prevent securities fraud," in *Proceedings of the 11th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2005.

[15] J. Neville and D. Jensen, "Relational dependency networks," *Journal of Machine Learning Research*, vol. 8, 2007.

[16] D. Heckerman, D.M. Chickering, C. Meek, R. Rounthwaite, and C. Kadie, "Dependency networks for inference, collaborative filtering, and data visualization," *The Journal of Machine Learning Research*, vol. 1, pp. 49–75, 2001.

[17] George Casella and Edward I. George, "Explaining the gibbs sampler," *The American Statistician*, vol. 46, no. 3, pp. 167–174, August 1992.

[18] Minwoo Yun and Mitchel Roth, "Terrorist hostage-taking and kidnapping: Using script theory to predict the fate of a hostage," .

[19] David Jensen, Andrew Fast, Brian Taylor, and Marc Maier, "Automatic identification of quasi-experimental designs for discovering causal knowledge," in *Proceedings of the Fourteenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2008.