Measuring Confidence Intervals in Link Discovery: A Bootstrap Approach

Jafar Adibi, Paul R. Cohen and Clayton T. Morrison
Center for Research on Unexpected Events (CRUE)
USC Information Sciences Institute
4676 Admiralty Way, Suite 1001
Marina del Rey, California 90292
{adibi,cohen,clayton}@isi.edu

Abstract

Recently there has been great interest in the development of information technology for Link Discovery (LD). LD is relevant to a wide range of research topics, including social network analysis, fraud detection, graph theory, pattern analysis and link analysis. The main goal of this research is the development of techniques for mining large amounts of data to find hidden patterns, extract valuable knowledge and discover hidden links among sparse pieces of evidence. In LD many such algorithms are deterministic and the constructed hypotheses are not qualified by probabilities. However, for nearly all applications the data available are sampled from a population. Hence, the discovered knowledge and implied hypotheses is probabilistic in nature and such uncertainty has to be measured. Due to the nature of the LD problems many of the current techniques and methods lack such measurement. We are interested in addressing this methodological problem and provide a general method for measuring the confidence intervals of hypotheses generated by LD algorithms. In this paper, we examine the bootstrap resampling method to measure the uncertainty in LD hypotheses. We study and analyze an example of of this method applied to the problem of discovering group membership, and discuss the effect such evaluation has on the generated hypotheses and their interpretation. Our preliminary results are encouraging and indicate that the bootstrap confidences are correlated with derived hypothesis.

1. Introduction

Recently there has been great interest in developing information technology for Link Discovery (LD). LD is relevant to a wide range of research topics, including social network analysis, fraud detection, graph theory, pattern analysis and link analysis. The common goal of this research is the development of techniques for mining large collections of data to extract valuable knowledge that may be present as hidden patterns or links among seemingly unrelated items. Successful LD applications will discover the hidden structure of organizations, relate groups, identify fraudulent behavior, model group activity and provide early detection of emerging threats.

LD requires a radically different approach to knowledge discovery, both in techniques and in approaches to evaluating LD algorithm results. The departure from standard approaches is made clear in the following five characteristics of LD problems and their representation: 1) Data is heterogenous, arriving from multiple sources. The data and patterns sought include representations of people, organizations, objects, actions and events. Each of these entities has its own set of attributes, and there are many types of relations that might exist between them. 2) Unlike conventional data mining, in which nodes are variables and links are statistical relations among variables, nodes represent entities and links are relations amongst entities. 3) LD assesses the likelihood that an instance of a specific graph-theoretic structure in the data matches a pattern of interest. The structure may include temporal, spatial, organizational, and/or transactional patterns. 4) All LD problems involve estimating a population based on a sample of data. Typically, a relatively low number of observations for each entity can be recorded, and the overall sample is typically small relative to the size of the population. 5) The data becomes available over time, so the timing of when to make a decision based on LD analysis is a central issue.

In LD many such algorithms are deterministic and the constructed hypotheses are not qualified by probabilities. However, for nearly all applications the data available are sampled from a population. Hence, the discovered knowledge and implied hypotheses is probabilistic in nature and such uncertainty has to be measured. Due to the nature of the LD problems many of the current techniques and methods lack such measurement. We are interested in addressing this methodological problem and provide a general method for measuring the confidence intervals of hypotheses generated by LD algorithms. Bootstrap sampling has been successfully applied in the areas of finance, modern econometrics, biomedical engineering, data mining, machine learning and applied statistics. In this paper, we examine the bootstrap resampling method to measure the uncertainty in LD hypotheses. We study and analyze an example of of this method applied to the problem of discovering group membership, and discuss the effect such evaluation has on the generated hypotheses and their interpretation.

The rest of this paper is organized as follows. We begin with a brief introduction to LD and identify the problem of measurement of confidence intervals in such space. Next, we briefly review bootstrap sampling and we illustrate application of such technique in LD. At the end, we report our finding of exploitation of bootstrap sampling on group membership followed by conclusion remarks.

2. Confidence Intervals in Link Discovery

The central goal of LD is to identify the relations amongst a wide variety of entities that may represent objects, events, persons, organizations and plans. Such identification may be based on known, complex and multi-relational patterns, or may be the product of the LD process itself. Many of the applications of LD are intended for real world databases that include information about people. Concerns about privacy and accuracy become paramount as LD technology is developed for applications to aid law enforcement and intelligence organizations in their efforts to detect and prevent illegal and fraudulent activities and threats to national security. For this reason, accuracy and justification for our confidence in a LD algorithm result are essential. For instance, the measures of the precision and recall of suspected individuals should be known: if considering a population of 2000 individuals, a 5% change in accuracy could translate into whether 100 of those individuals are correctly classified.

Figure 1 (left) illustrates the problem of LD in general. As it shows almost all entities are connected to each other directly or indirectly. Red (dark) nodes represent suspected individuals or Bad Guys. Assume a few members of this set are known in advance. The challenge then is to discover potential additional hidden members of such a group given evidence of low level sparse data such as communication events, business transactions, familial relationships, etc. The right part of Figure 1 illustrates the same environment after removing many weak connections among individuals. The new graph (on right)make it easier to separate dark and bright nodes form each other.

3. Confidence Intervals and Bootstrap Resampling

Our hope is that our LD algorithms accurately characterize the population based on properties identified in the sample. However, most current LD algorithms do not characterize the probabilistic properties of the hypotheses derived from the sample of data: if a community finding algorithm reports that an individual is part of a hypothesized community, *what is our confidence in that report?* Such confidence can be measured.

The conventional way to assess the certainty of a hypothesis generated from a sample is to first characterize the hypothesis as a function Θ of the sample (Θ is usually called a statistic) and then to imagine drawing an infinite number of samples from the sample space S, recalculating Θ for each sample. The resulting distribution of Θ will have some variance, called the *standard error* of Θ. The standard error is used to define an interval called a *confidence interval*. In general, we prefer hypotheses to have small standard error and associated small confidence intervals.

The standard error captures a simple intuition: the sample we observe is just one of a potentially unbounded number of samples. If the variability of the sample space is large and we could have drawn a very different sample, then we should not put a lot of faith that the hypothesis based on our current sample is representative of the population. There are three ways to estimate the variability of a sample space. One is to have some information about the population from which samples are drawn (e.g., the population variance). The other two approaches infer the variance of the population from the sample. Of these, the more ”classical” obtains the standard error of a statistic via the central limit theorem or some other asymptotic theorem. The other approach obtains the standard error via *bootstrap resampling*. It is this latter approach that we will investigate here.

Bootstrap resampling, similar to other nonparametric methods, is used in estimating and testing hypotheses while minimizing the number of arbitrary assumptions required. The value of assumptions, such as the assumption that the underlying sampling distribution is *normal*, is that they can lead to greater precision. However, these methods only work if the assumptions are valid. In many real-world problems, these assumptions are violated or their status is unknown, limiting the applicability of the methods. Bootstrap resampling trades precision for robustness. Bootstrap resampling makes no assumptions about the underlying sampling distribution, so is ideal for estimating statistical parameters, such as those used in LD, where the underlying sampling distribution is not well understood.

We employ the bootstrap method to estimate group membership standard errors and their associated confidence intervals. The bootstrap procedure uses resampling with replacement from an already-acquired sample, and thus does not incur further cost in acquiring more data. With an approximate of standard errors, we can associate p-values on estimates of the population parameters.

4. Using Bootstrap to estimate LD Confidence Intervals

In this section, we present the steps for deriving the confidence interval for a LD hypothesis generated by a group-finding algorithm. Although we focus on this one type of hypothesis, we believe this procedure is generally applicable to a large class of LD hypothesis generating methods.

A large class of LD algorithms is devoted to discovering hidden members of a group. Typically, a set of known group members is already available, acquired through some other process. The challenge is to discover the rest of the members that belong to the group.

Figure 2 represent the group detection space by projecting the whole data to a graph. Each node represents an entity (e.g., a person) and each link between two entities represents a set of events (e.g., sending e-mails, making phone calls, etc.). [1] provide a framework to measure the connections among individuals. In that work, characterizations of an individual's *activities* and *events* are exploited as indicators of whether two individuals are strongly connected to one another.

4.1. Group Finder

[2] introduces the *Integrated Knowledge-Based and Statistical Reasoning Group Finder* system. One of the modules in this system is the Group Finder algorithm. The system accesses a large database of the activities of individuals and events and projects them into random variables in a graph representation. (Figure 2) Once in this representation, the Group Finder employs a measure of mutual information (MI) to assess the relations between variables. If two variables are independent, then the MI between them is zero. If the two variables are strongly dependent, as is the case if one is a function of the other, then the MI between them is large. In other words, the MI between two random variables indicates how much we know about the first one given all information about the second, and vice versa. MI measures general (non-linear) dependence while a correlation function measures linear dependence, so MI is more desirable.

Each group is identified by its seed members – the individuals we assume we already know are members of the group. The challenge is to identify all other hidden members of all groups in the dataset. We assume there are $P = \{p_i, ..., p_N\}$ individuals and $G = \{g_i, ..., g_M\}$ in the database and each g_i represents by its seed members. For each g_i in G, Group Finder provides a list of individuals, ranked by the strength of their relation to seed members in g_j , according to the relation of all individuals in the database provides by MI module as $\{MI(p_i, g_j), ..., MI(p_N, g_j)\}$. The higher the value of $MI(p_i, g_j)$ the stronger the relationship between p_i and g_j.

Group Finder begins from the set of seed members and for each seed members it retrieves all activities it participated in and adds any new individuals previously not recorded but also participating in those activities. Next, it considers the expanded group as the new universe and computes MI for each connected pair in the graph. Finally, it looks for individuals that either have high MI score with one of the seed members or high MI with all of the seed members combined as one entity. Members whose score is below a user-defined threshold are dropped from the list. Group Finder then selects settings for the best precision and recall, using the F-measure defined as

$$F_{measure} = -\frac{(b^2 + 1) \times Precision \times Recall}{b^2 \times Precision + Recall}$$

In the equation, b indicates the desired weight for precision and recall. For our experiment, we used $b = 1.5$ which put more emphasis on recall than precision.

We may replace the Group Finder module with any other module capable of assigning group membership to individuals in the system's database. While the Group Finder output is outstanding, it does not provide the probability that an individual is a member of a group, let alone a confidence interval for the hypothesis. Instead it only ranks their relation strength based on the MI model.

We introduce a membership function $\Theta(p, g)$ as a binomial variable on relations of the form "*individual p belongs to group g*." If we assume there are G groups available in a given dataset, then for each g_j in $G = \{g_1, g_2, ..., g_m\}$ and for each p_j in $P = \{p_1, p_2, ..., p_N\}$, $\Theta(p_i, g_j)$ indicates the probability of such membership. As illustrated in Figure 2 all group membership hypotheses are extracted form events and activities among individuals

Since we have seen such events only once any group membership hypothesis is uncertain.To measure the confidence interval of such hypothesis we apply the bootstrap technique to the problem of group membership. Suppose that a sample X is used to estimate a membership function $\Theta(p, g)$ of the distribution and let $\hat{\Theta} = s(X)$ be a statistic that estimates Θ given a sample $X = \{X_1, X_2, ..., X_n\}$. For the purpose of statistical inference on Θ, we are interested in the sampling distribution of $\hat{\Theta}$ to evaluate the accuracy of our estimator and to set confidence intervals for our estimate of Θ. If the true distribution π were known, we could draw samples $X^{(b)}$, $b = 1, ..., B$ from π and use Monte Carlo methods to estimate the sampling distribution of our estimate $\hat{\Theta}$. Since π is unknown and we cannot sample from it, the bootstrap resampling method allows us to achieve our goal by resampling from the original sample instead. The distribution from which the bootstrap samples are drawn is called the *empirical distribution*.

4.2. The Bootstrap Principle

The basic idea of the bootstrap resampling method is that, in the absence of any other information about the distribution the observed sample contains all the available information about the underlying distribution. Thus, resampling the sample is the best estimate available for estimating the population distribution.

Let $X = (X_1, ..., X_n)$ be a sample of events and let $\hat{\Theta} = s(X)$ be a statistic that estimates Θ. For a sample $X_1, ..., X_n$ of independent real-valued random variables with distribution π, we define a probability distribution by:

$$\hat{\pi} = \frac{1}{n}\sum_{i=1}^{n} X_i$$

$\hat{\pi}$ is the empirical distribution of the sample X. $\hat{\pi}$ can be thought of as the distribution which puts mass $1/n$ on each event X_i. Clearly for events that occur more than once in the sample, the mass will be a multiple of $1/n$. It follows that $\hat{\pi}$ is a discrete probability distribution with the effective sample space $\{X_1, ..., X_n\}$. It can be shown that $\hat{\pi}$ is a nonparametric maximum likelihood estimator of π – this justifies our use of $\hat{\pi}$ to estimate π when no other information about π is available (such as, e.g., that $\hat{\pi}$ belongs to a parametric family). This addresses precisely the problem faced in nearly all LD problems, where the distribution of the population is generally unknown.

Suppose we want to draw an identically and independently distributed sample $X^* = (X_1^*, ..., X_n^*)$ from $\hat{\pi}$. As we have noted above, $\hat{\pi}$ puts mass $1/n$ on each observation X_i. Thus, when sampling from $\hat{\pi}$, the i^{th} observation X_i in the original sample is selected with probability $1/n$. This leads to the following two-step procedure:

- Draw $i_1, ..., i_n$ independently from the uniform distribution on $\{1, ..., n\}$
- Set $X_j^* = X_{ij}$ and $X^* = (X_1^*, ..., X_n^*)$.

In other words, we sample with replacement from the original sample $X_1, ..., X_n$. Sampling with replacement means that if any member X_i of the original set is chosen as the first value of the bootstrap sample, it could also be chosen as any of the successive values. In principle, therefore, a bootstrap sample could consist of the same value repeated n times. However, the probability of this actually happening is quite small as the number of different bootstrap samples available is n^n. Sampling with replacement preserves the *a priori* probabilities of the classes throughout the random selection process. The bootstrap principle assumes $X = (X_1, ..., X_n)$ is a random sample from a distribution P and $\hat{\Theta} = s(X)$ is an estimation for Θ.

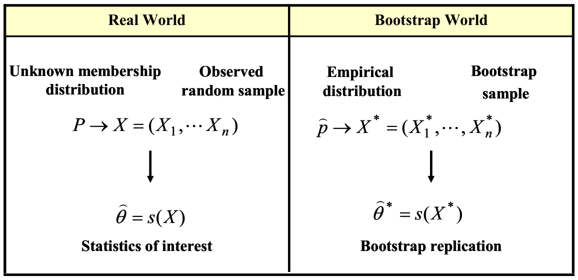


Figure 3. The real world versus the bootstrap world.

The bootstrap methods mimic the data generation process by sampling from $\hat{P}$ of the unknown distribution π. Hence, $X^* = (X_1^*, ..., X_n^*)$ is a bootstrap sample from $\hat{\pi}$ and $\hat{\Theta}^* = s(X^*)$ is the bootstrap replication of Θ. Figure 3 contrasts direct sampling from the population with the bootstrap resampling from the original sample.

The sampling distribution of $\hat{\Theta}$ is then estimated by its bootstrap equivalent. Having generated the B bootstrap resampling-based estimates $\hat{\Theta}^{(1)}, ..., \hat{\Theta}^{(B)}$, we have an estimate of the sampling distribution $\hat{\Theta}$. From this, we can construct confidence intervals for Θ. [[Is this supposed to be Θ or $\hat{\Theta}$]] If $\hat{\Theta}$ is approximately normally distributed with mean Θ and variance $se(\hat{\Theta})^2$, then an approximate $1 - \alpha$ confidence interval for Θ ranges from $\hat{\Theta}_L = \hat{\Theta} - z_{\alpha/2}\hat{se}_{boot}(\hat{\Theta})$ to $\hat{\Theta}_U = \hat{\Theta} + z_{\alpha/2}\hat{se}_{boot}(\hat{\Theta})$, where z_α is the α critical value of the standard normal distribution. However, since making such assumption is contrary to the nonparametric aspect of bootstrap, instead we can obtain the $[p,\ 1-p]$ confidence interval (e.g., [0.05, 0.95]) by finding the corresponding quantiles of bootstrapped sampling distribution (e.g., the 5^{th} and 95^{th} values in a sorted list of 100 bootstrap variates).

4.3. A Link Discovery Bootstrap Procedure

In this section, we briefly illustrate how to apply the bootstrap resampling method for the group membership problem. In order to perform the bootstrap procedure for a membership function $\Theta(p, g)$, for each g_i in G, we follow steps similar to the generic algorithm outlined by [6], as follows.

Assume we have a set of events (similar to the data in Figure 2)

1. Given a sample of events of size n, draw a random "bootstrap" sample of size n with replacement from X^*.
2. Calling Group Finder to represent the sampled events as a graph, expand the graph and find the strength of the relation between all individuals in the expanded graph and all $g_i \in G$.
3. For each g_j, construct a list of all p_i for which MI is above the user-defined threshold. In this list, store $L^b(g_j)$, $b = 1, ..., B$. Thus, each row of $L^b(g_j)$ indicates the MI of an individual with g_j in the $b^t h$ run.
4. Repeat steps 1 through 3 for some large number of times B.
5. Finally, estimate the "bootstrap standard error" of the parameter of interest using the B bootstrap statistics. An estimate of the bias of the statistic of interest is obtained by subtracting the mean bootstrap statistic from the original sample statistic.

We can compute the boundary of group membership following the method described in the previous section. For a $[p, 1-p]$ confidence interval (e.g., [0.05, 0.95]), we find the corresponding quantiles. Figure 4 illustrates this procedure.

4.4. Group Membership Probability

So far we have defined the confidence interval for a LD hypothesis such as that generated by the Group Finder algorithm. Group Finder provides a ranked list of individuals and the strength of the relation to a given group. For each g_j in $G = \{g_1, g_2, ..., g_3\}$ and for each p_i in $P = \{p_1, p_2, ..., p_N\}$, $MI(p_i, g_j)$ indicates the MI between group g_j and p_i.

We may define $\Theta(p, g)$ as follows:

$$pr(\Theta(p_i, g_j)|MI(p_i, g_j)) = pr(MI(p_i, g_j)|\Theta(p_i, g_j)) \times \frac{pr(\Theta(p_i,g_j))}{pr(MI(p_i,g_j))}$$

$pr(\Theta(p_i, g_j))$ is the prior probability that an individual belongs to a group. Such probability is proportional to the size of the group (the seed members). There is no additional information about group membership and each individual might be the member of more than one group. We are also able to calculate $pr(MI(p_i, g_j))$ because each group has a set of seed members which belongs to the group with a 100% certainty (by assumption). Since $pr(\Theta(p_i, g_j))$ and $pr(MI(p_i, g_j))$ can be calculated for each group g_j, we have:

$$\frac{pr(\Theta(p_i, g_j))}{pr(MI(p_i, g_j))} \approx C, Cisaconstant$$

5. Experimental results

In order to evaluate our method empirically, we applied the bootstrap procedure to measure the confidence in hypotheses generated by the Group Finder for a wide variety of synthetic data for which we know the true population parameters. In this section we analyze the results of applying the bootstrap resampling procedure over four datasets. First, we describe the synthetic world, then we discuss and evaluate our findings.

5.1. Synthetic Data

Real world data is expensive to collect and involves privacy issues. For this reason, we used synthetic data generated by a simulator designed by IET. The simulator generates data representing different numbers of agents, organizations, groups, activities and other entities and attributes. The representation can vary in accuracy, meaning that accurate representation of the underlying simulator events may be obscured by noise.

From the point of view of group detection, the artificial world consists of *individuals* that belong to *groups*. Groups can be *threat groups* (which perform *threat events*) or *non-threat-groups*. Targets can be exploited (in threat and non-threat way) using specific combinations of resources and capabilities. Each such combination is called a *mode*. Individuals may have any number of capabilities or resources, belong to any number of groups, and participate in any number of exploitations at the same time. Individuals are *threat* individuals or *non-threat* individuals. Every threat individual belongs to at least one threat group. Non-threat individuals belong only to non-threat groups. Threat groups have only threat individuals as members. Threat individuals can belong to non-threat groups as well. A group will have at least one member qualified for any capability required by any of its modes. Non-threat groups carry out only non-threat modes. Finally, individuals may be members of several groups.

Because we are using a simulator capable of generating data, we are able to create datasets with a variety of properties by adjusting the parameters of the simulation. Table 1 lists some of the parameters we selected for generating the datasets we used in our evaluation. Each dataset employs different values for each of these features. Of particular interest are observability (how much of the artificial world information is available as evidence), corruption (how much of the evidence is changed before being reported) and clutter (how much irrelevant information that is similar to the information being sought is added to the evidence).

Number of entities	Number of Events	Number of Distinct Threat Patterns	Lowest Signal to clutter ratio	Lowest Signal to Noise Ratio	Observability	Corruption of Evidence
10,000	100,000	20	0.3(-5 db)	.008(-21 db)	50%-100%	0-25%

Table 1. Synthetic Data Characteristics

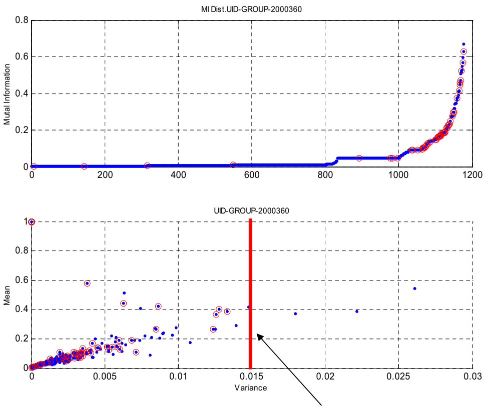
	Dataset Characteristics			Group Statistics		Group Finder Result		Result After Bootstrap	
Dataset	Clutter	Observability	Connectivity	Groups	Avg. Members in each group	Avg. Precision	Avg. Recall	Avg. Precision	Avg. Recall
3	**High**	**High**	**Medium**	**14**	**53**	**0.81**	**0.87**	**.83**	**.90**
8	**Medium**	**High**	**Medium**	**11**	**50**	**0.59**	**0.86**	**.63**	**.89**
9	**Medium**	**High**	**Medium**	**16**	**64**	**0.70**	**0.72**	**.73**	**.74**
10	**Medium**	**High**	**Medium**	**19**	**63**	**0.88**	**0.66**	**.88**	**.68**

Table 2. : The result of bootstrap resampling effect for group detection problem in 4 different datasets

5.2. Results

Table 2 shows a detailed specification of each of the four datasets we used,along with the result of the Group Finding engine. In particular, we varied the level of clutter (cases similar to a threat case), observability (how much of the real worlds is observable to us) and connectivity (membership to different group). In addition, Table 2 shows the number of groups in each dataset as well as the average number of members in each group.

At the end of each run, for all g_j in $G = \{g_1, g_2, ..., g_m\}$ and for each p_i in $P = \{p_1, p_2, ..., p_N\}$, we assigned a probability to $\Theta(p_i, g_j)$ based on the probably of membership of p_i in g_j. This probability represents our confidence that p_i is a member of g_j. There are some members which appear as members of a given group in each bootstrap run. These members are those which are most likely members of that group. However, there are some members which do not appear in the final members list in every bootstrap runs. Our confidence in assigning a membership between these individuals and groups is lower. Table 2 illustrates the average precision and recall for each group after applying a threshold. In the bootstrap sampling procedure, we applied the same threshold for each run of Group Finder and saved the result. In addition, those individuals with low confidence were dropped from the final list. Interestingly, bootstrap resampling not only introduces the notion of a “boundary” for Group Finder output, it also *improved* the accuracy in *both* precision and recall.



Example of upper boundaries

Figure 5. Example of a confidence interval for group detection. Top: plot of mutual information among all individuals and a given group. Bottom: mean versus. variance of mutual information for each member after the bootstrap procedure. The red (dark) line indicates the confidence interval for group detection module.

5.3. Confidence Intervals

Figure 5 shows an example of bootstrap resampling for one group. The plot in the top of the figure illustrates the MI value for all 1200 individuals after running the Group Finder.

Each dot in the graph represents an individual and circled dots represent the true members of a group. As the figure shows, most of the circled dots are located in the upper part of the list. Note that due to the nature of the synthetic data there many points which are very hard or impossible to accurately identify. The plot in bottom of the illustrates the distribution of mean and variance for each individual after bootstrap resampling.

For each group we ran the bootstrap test, determining the mean and variance of MI for each individual in the group. Figure 6 illustrates the distribution of mean and variance for 3 different groups of different datasets. The three top graphs show the list of MI output, ranked based on their MI with that group. Those points which are circled actually belong to the group. As the plots show, the MI model does a good job of finding group members.

Note that there are more than 1000 points in the lower left portion of the graph, a fact obscured by the resolution of the graph. In these graphs we are interested in those individuals which may receive a high score (according to the MI module) but also a low confidence. Points in the gray area satisfy this criteria. Thus, we drop these points from the final MI module list. Again, note that this procedure is independent of the accuracy of MI module – we may replace this module with any other group detection technique and apply the same bootstrap procedure to identify those points with high mean but low variance.

For each point in the example we calculated the boundaries of variance for the confidence level of $\alpha = 95\%$. All points with the variance outside this boundary have been dropped from the final list. The cut-off number for the example shown in Figure 5 was about 0.015. This cut in the MI output list boosts the overall precision and recall of the Group Finder module up to 5%. Increasing 5% in MI module translates into about 100 members in large groups (i.e. when group size is about 2000).

Figure 6 illustrates the same phenomenon for 5 other groups based on different datasets. As the figure shows, there are many groups for which bootstrap resampling increases the accuracy of the MI module output. However, for some of the groups, all of the discovered individuals already have a high confidence so bootstrap resampling does not help much.

5.4. Complexity

The complexity of bootstrap sampling is relatively low.The overall complexity depends on the complexity of the LD algorithm. B is relatively a small number – usually about 200 samples are needed for finding a standard error, whereas on the order of 2000 are needed to derive intervals. Some recent results suggest that you can do well with many fewer samples [**?**]. The main point is that you want to be confident that the results would not differ in a meaningful way if you were to repeat the bootstrap procedure. B is a constant and overall complexity depends on the order of algorithm itself.

For instance, in our experiment, computing MI between two individual is $O(M * N)$ in which N is thenumber of people connected to a given individual and M is the average number of relations a person is involved in. Since we call the MI module B times, the final complexity will is $O(M * N * B)$. In practice, bootstrap tests must use a finite number of bootstrap samples. It has been shown that the outcome of the bootstrap sampling depends on the sequence of random numbers used to generate the bootstrap samples, and it necessarily results in some loss of power. [4] presents a simple pretest procedure for choosing the number of bootstrap samples to minimize loss, depending on the chosen α.

6. Conclusion

LD techniques attempt to develop inference algorithms which estimate some hypotheses from an observed sample data. These hypotheses could be in the form of group membership, association rules, underlying data structures, etc. LD problems raise new issues, most important of which is the need for the ability to assess confidence intervals for implied hypotheses, something most previous methods have lack. In this paper we sketched a general methodology for LD techniques to measure such uncertainty. We introduced bootstrap resampling as an alternative to measure the standard error and to construct the confidence intervals of such hypotheses. Bootstrap sampling makes no assumption about the population (such as normality, equal variance or the central limit theorem) other than the assumption of random sampling. Since bootstrap resampling uses sampling with replacement, it preserves the *a priori* probabilities throughout the random selection process and is capable of obtaining accurate measures of both the bias and variance of the true error estimate. We believe this is a natural fit for LD problems in which the distribution of data is unknown and only relatively small samples of events are available.

The purpose of this paper was to identify the confidence interval problem in LD and to examine the applicability of the bootstrap for estimating confidence in LD hypotheses. We demonstrated this approach in finding hidden groups in large databases, focussing on the confidence of membership of each individual to each group. Our preliminary results are encouraging and indicate that bootstrap confidences are correlated with derived hypothesis in general and with Group Finding group membership in particular. We found that the majority of individuals that were labelled as a member of a group with high confidence were indeed members of that group. Group Finding exemplifies many LD approaches that use mutual information to estimate the relation strength among individuals.

6.1. Future Work

There are several directions for future work. We are interested in further theoretical analysis of the bootstrap in the context of LD problems. Also, our results are preliminary – much more experimentation is needed with different features of the LD problem and with different group finding methods. Another important direction is to find principled methods for incorporating the clues provided by the bootstrap confidence measures into LD techniques. For example, we may use the bootstrap result for tuning the mutual information engine and to modify the search and expansion strategy in the module.

Our final goal is twofold: first, to provide guarantees about the boundaries of discovered knowledge and second, to find better ways to guide the process of link analysis. Although current group finding methods are reliable, the empirical results in this paper indicate that there is definite room for improvement and adjustment in their performance. Bootstrap sampling as a nonparametric method is an excellent candidate for testing hypotheses and minimizing the arbitrary assumptions which are typically made during LD processing.

6.2. Related Work

Hypothesis testing has been an important issue in AI, statistics, data mining, machine learning and social network analysis. Different methods such as Monte Carlo simulation, cross validation, bootstrap resampling, and hold-out and Jack-Knife methods have been applied to measure the confidence intervals of hypotheses.[11]

Monte Carlo methods are most often used in simulation studies with computer-generated data to show how p-values for astatistical test or estimation method functions when no convenient real data exist. Monte Carlo methods are used to make inferences about the population from which a sample has been drawn. Bootstrapping is a special case of Monte Carlo estimation [5]. Bootstrapping is used most often to approximate standard errors and associated p-values on estimates of population parameters when the sampling distribution of the target population is either indeterminate or difficult to obtain empirically. Bootstrap sampling has been used successfully in stratified data, finite populations, missing data, classification, time series and spatial problems, and linear, nonlinear, and smooth regression models.

[3] presents a wide range of empirical methods for AI research. Cohen explains the blunt interrogation of statistical hypothesis testing through classical parametric methods and computer-intensive (Monte Carlo) resampling methods. He also presents flexible resampling techniques in an accurate, accessible manner. Many other researches in the field of data mining and machine learning also use appropriate tools for their estimation. For instance, [9] showed that cross validation might be a better choice in model selection and [10] examines several ways to measure the ROC confidence intervals and [7] Applied the bootstrap method for computing confidence measures on features of induced Bayesian networks.

We introduced bootstrap sampling as an alternative to estimate confidence intervals in LD hypothesis. Since the field of link discovery is relatively new there is still much research exploring different aspects of the evaluation of discovered knowledge. For instance, in the field of social networks, [8] used the bootstrap to estimate the confidence in friendship choices among Southern California middle school students.

6.3. Acknowledgments

This work was partially supported by the Defense Advance Research ProjectsAgency under Air Force Research Laboratory contract F30602-01-2-0583. Authors would like to thank Hans Chalupsky, Andre Valente, Eric Melz for their comments and support.

References

[1] J. Adibi. Real world characteristics of link discovery databases. In *Workshop on Data Mining for Counter Terrorism and Security with the Third SIAM International Conference on Data Mining (SDM 2003)*, 2003.

[2] J. Adibi, H. Chalupsky, V. A., and M. E. The kojak group finder: Connecting the dots via integrated knowledge-based and statistical reasoning. In *Submited to IAAI 2004*. AAAI-IAAI, 2004.

[3] P. R. Cohen. *Empirical Methods for Artificial Intelligence*. The MIT Press, Cambridge, 1995.

[4] A. C. Davison and D. V. Hinkley. *Bootstrap Methods and their Application*. Cambridge Univ. Press., Boston, 1997.

[5] B. Efron. *The Jackknife, the Bootstrap and other resampling plans*. Society for Industrial and Applied mathematics, Philadelphia, 1982.

[6] B. Efron and R. J. Tibshirani. *An introduction to the bootstrap*. Chapman & Hall, New York, 1993.

[7] N. Friedman, M. Goldszmidt, and A. Wyner. On the application of the bootstrap for computing confidence measures on features of induced bayesian networks. In *Seventh International Workshop on Artificial Intelligence and Statistics*, 1999.

[8] B. R. Hoffman and T. W. Valente. Ethnicity and friendship choices among southern california middle school students. In *Internationl Social Network Conference Sunbelt XXIII*, 2003.

[9] R. Kohavi. A study of cross-validation and bootstrap for accuracy estimation and model selection. In *14th International Joint Conference on Artificial Intelligence*, pages 1137–1143. Morgan Kaufmann Publishers, Inc., March 1995.

[10] S. A. Macskassy, F. J. Provost, and M. L. Littman. Confidence bands for roc curves. In *CeDER Working Paper*, 2003.

[11] E. W. Noreen. *Computer intensive methods for testing hypotheses: An introduction*. John Wiley & Sons, New York, 1989.

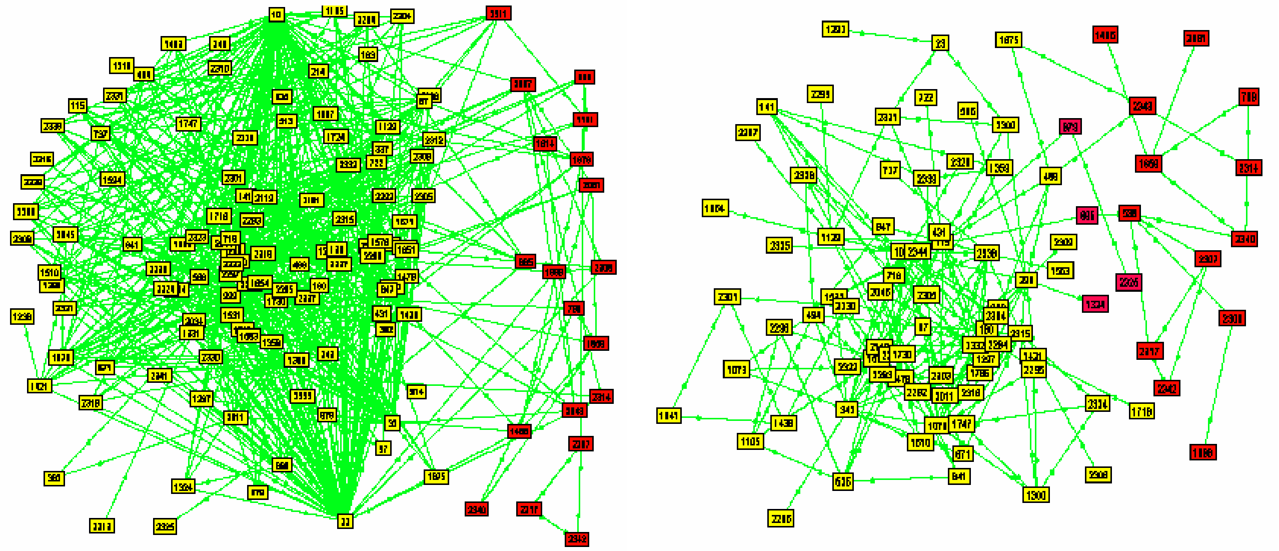
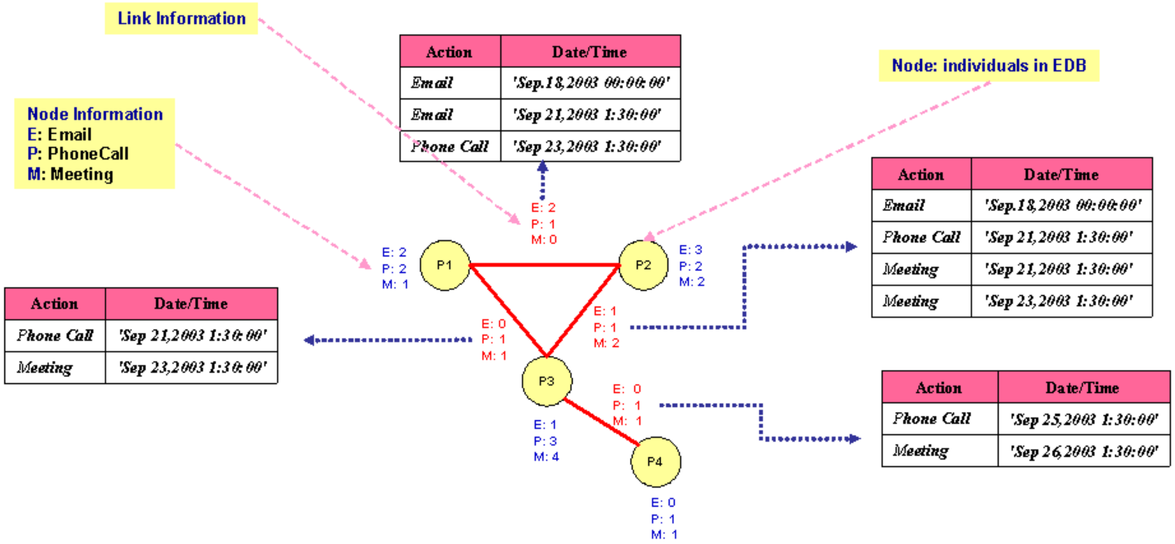


Figure 1. Visualization of the Link Discovery Problem. Red (dark) nodes represent suspected individuals or ÒBad GuysÓ and yellow (bright) nodes represent ÒGood GuysÓ. The challenge is to identify the link structure that will separate the two kinds of nodes with high accuracy. The left graph shows the original problem. The right graph shows the same graph after removing weak links using a group finding technique.



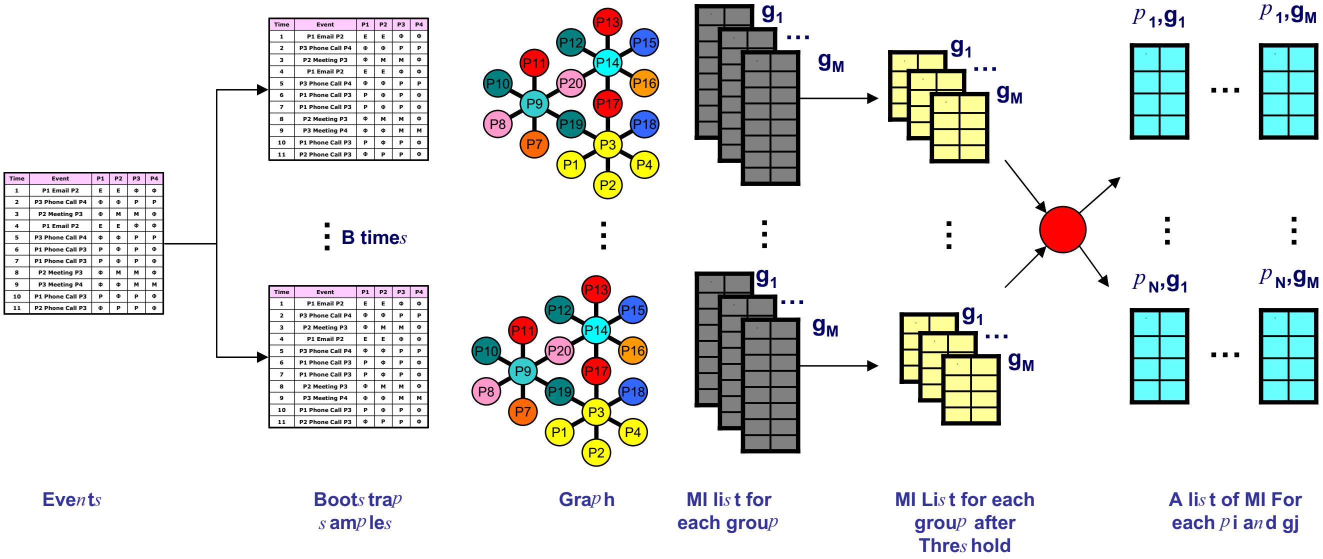
Action	Date/Time
Email	*'Sep.18,2003 00:00:00'*
Email	*'Sep 21,2003 1:30:00'*
Phone Call	*'Sep 23,2003 1:30:00'*

Action	Date/Time
Email	*'Sep.18,2003 00:00:00'*
Phone Call	*'Sep 21,2003 1:30:00'*
Meeting	*'Sep 21,2003 1:30:00'*
Meeting	*'Sep 23,2003 1:30:00'*

Action	Date/Time
Phone Call	*'Sep 21,2003 1:30:00'*
Meeting	*'Sep 23,2003 1:30:00'*

Action	Date/Time
Phone Call	*'Sep 25,2003 1:30:00'*
Meeting	*'Sep 26,2003 1:30:00'*

Figure 2. Representation of the link discovery space as a graph. Nodes represent people, links represent sets of events resulting from activities such as phone calls, meetings and emails. People are associated with each other through their activities.



Time	Event	P1	P2	P3	P4
1	P1 Email P2	E	E	Φ	Φ
2	P3 Phone Call P4	Φ	Φ	P	P
3	P2 Meeting P3	Φ	M	M	Φ
4	P1 Email P2	E	E	Φ	Φ
5	P3 Phone Call P4	Φ	Φ	P	P
6	P1 Phone Call P3	P	Φ	P	Φ
7	P1 Phone Call P3	P	Φ	P	Φ
8	P2 Meeting P3	Φ	M	M	Φ
9	P3 Meeting P4	Φ	Φ	M	M
10	P1 Phone Call P3	P	Φ	P	Φ
11	P2 Phone Call P3	Φ	P	P	Φ

Time	Event	P1	P2	P3	P4
1	P1 Email P2	E	E	Φ	Φ
2	P3 Phone Call P4	Φ	Φ	P	P
3	P2 Meeting P3	Φ	M	M	Φ
4	P1 Email P2	E	E	Φ	Φ
5	P3 Phone Call P4	Φ	Φ	P	P
6	P1 Phone Call P3	P	Φ	P	Φ
7	P1 Phone Call P3	P	Φ	P	Φ
8	P2 Meeting P3	Φ	M	M	Φ
9	P3 Meeting P4	Φ	Φ	M	M
10	P1 Phone Call P3	P	Φ	P	Φ
11	P2 Phone Call P3	Φ	P	P	Φ

Time	Event	P1	P2	P3	P4
1	P1 Email P2	E	E	Φ	Φ
2	P3 Phone Call P4	Φ	Φ	P	P
3	P2 Meeting P3	Φ	M	M	Φ
4	P1 Email P2	E	E	Φ	Φ
5	P3 Phone Call P4	Φ	Φ	P	P
6	P1 Phone Call P3	P	Φ	P	Φ
7	P1 Phone Call P3	P	Φ	P	Φ
8	P2 Meeting P3	Φ	M	M	Φ
9	P3 Meeting P4	Φ	Φ	M	M
10	P1 Phone Call P3	P	Φ	P	Φ
11	P2 Phone Call P3	Φ	P	P	Φ

Figure 4. The bootstrap procedure: take samples over events and for each run and for each group calculate the mutual information between all individuals in the expanded graph and such group. At the end of the procedure, for each member and each group there is a list of length B which stores the MI between a member and a group in each run

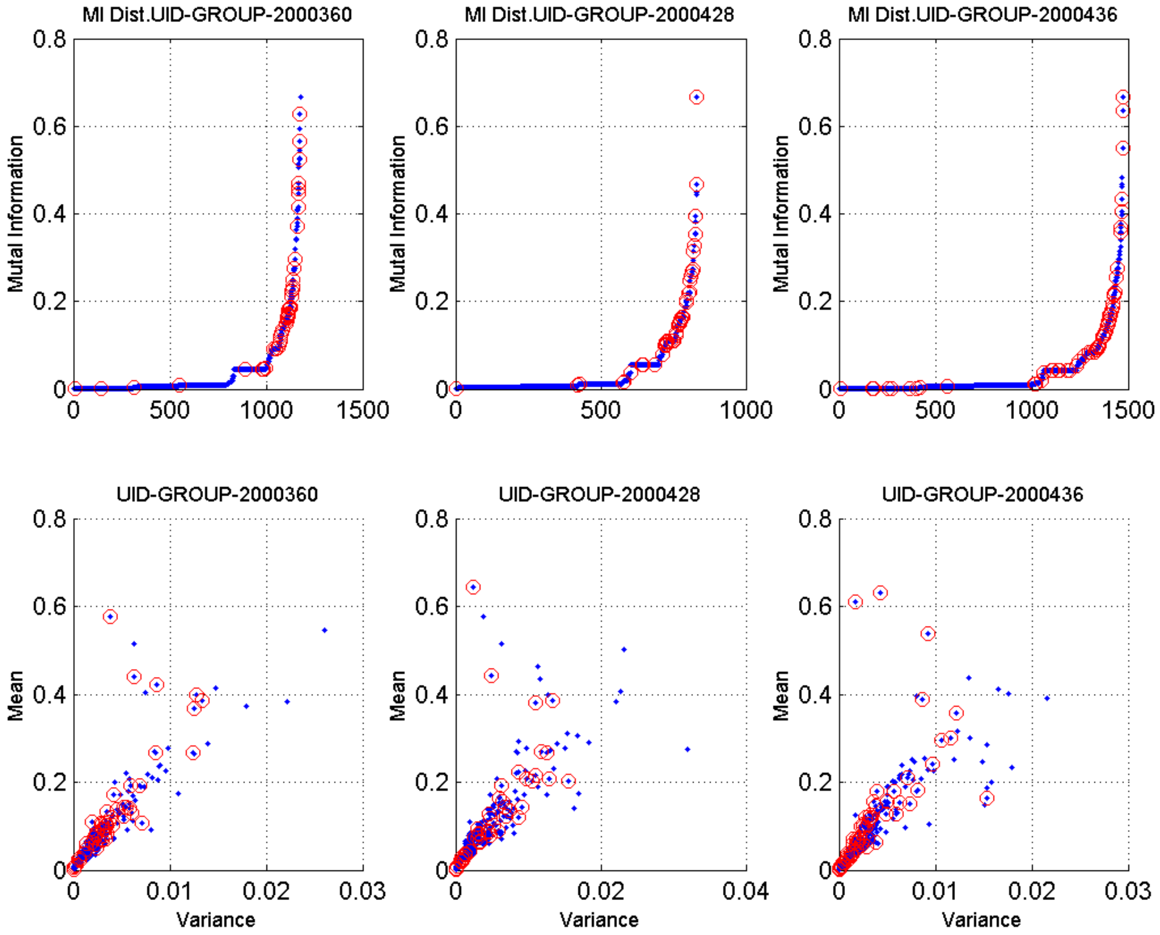


Figure 6. Top: Illustration of the Mutual Information module output for all individuals in 3 groups. Points with higher mutual information are strongly associated with the group and thus most likely members of that group. Bottom: Mean – Variance plot for several groups. Dots represent all individuals and circled dots represent those individuals who belong to the group.