COHEN'S KAPPA AS A METHOD FOR LINK PREDICTION

A Thesis

presented to

the Faculty of the Graduate School

at the University of Missouri-Columbia

In Partial Fulfillment

of the Requirements for the Degree

Master of Arts

---

by

MICHAELA HOFFMAN

Dr. Doug Steinley, Thesis Supervisor

MAY 2013

The undersigned, appointed by the dean of the Graduate School, have examined the thesis entitled

COHEN'S KAPPA AS A METHOD FOR LINK PREDICTION

presented by Michaela Hoffman,

a candidate for the degree of master of arts,

and hereby certify that, in their opinion, it is worthy of acceptance.

Dr. Doug Steinley

Dr. Ken Sher

Dr. Chris Wikle

ACKNOWLEDGEMENTS

I would like to thank my advisor and the chair of my master's committee Dr. Douglas Steinley, who has guided me through my first two years of Graduate school and my master's thesis. I look forward to continuing to work with him though the rest of my career. I would also like to thank Dr. Christopher Wikle and Dr. Kenneth Sher, who were able to take the time out of their schedules to be on my master's committee. Their suggestions have greatly improved this project.

TABLE OF CONTENTS

LIST OF TABLES

Figure Page

INTRODUCTION

Background

A network (e.g., graph) is a data set made up of nodes and edges. The nodes can represent any object of interest and the edges are the links between these nodes signifying some form of a connection. This type of data is used in many fields to represent different structures from the neural networks and food chains of the biological sciences (Zhu, Gerstein & Snyder, 2007) to analysis of terrorist networks (Krebs, 2002; Ressler, 2006). Networks can be either directed, where a link from one node to another is not necessarily reciprocated or undirected, where two nodes are either connected or not. Links can also be weighted, providing a measure of strength for each connection. For this study, we focus only on undirected/unweighted graphs.

One increasingly popular subset of networks used in fields such as sociology and psychology is social networks (see Wasserman & Faust, 1994). These are comprised specifically of people or groups and the connections between them. Social networks tend to be large and complex and contain a large amount of information in a unique way.

Given the complexity of social networks and their tendency to grow over time, detecting links not present in the data is an important question. This is divided into two problems that would have similar solutions, the missing link problem and link prediction problem. Looking for missing links asks what links should be in the data set that were not observed, whether from data measurement errors or unknown information. The link prediction problem focuses on links that might occur in the future based on the observed network. Examples often seen in link prediction literature use a network of authors

collaborating on papers, where the goal would be to predict collaborations for future papers (e.g., see Newman, 2001; Shibata, Kajikawa & Sakata 2012).

In looking at the graph structure of social networks, it is common to see clustering. In Newman's analysis of citation networks, the clustering coefficient indicated they were highly clustered (2001). Other research has gone into being able to detect clusters in social networks (e.g., see Mishra et al., 2009; Duan, et. al., 2012). Here, we will not look into the structure of network data, but the presence of clustering is taken into account in our simulation.

Current Solutions

There are many methods that have been created to predict links. They involve a wide range of techniques from the most basic graph theory ideas to more complicated machine learning algorithms and those that require more substantive information about the network. In this study, we focus on proximity measures that indicate a "distance" between each pair of nodes using just the structure of the network. This distance (or function thereof) represents how likely it is that a link should be there. These scores are then ranked, and a certain number or percentage of the best scores are taken to be the "predicted" edges.

Starting with the most basic of these techniques we have graph distance, which creates scores by calculating the shortest pathway between each pair of nodes. Here, the nodes with the lowest graph distance score would be the most likely to form a link. Another basic solution, common neighbors, makes its predictions based on the number of neighbors (other nodes they are linked to) that two nodes have in common. Similar to

common neighbors is Jaccard's Coefficient which calculates scores by dividing the number of common neighbors by the total number of neighbors that the two nodes possess. Preferential attachment, a solution used to model growth, takes the product of the number of neighbors two nodes has to be their score. The following are the equations, note that $\Gamma(X)$ is the set of nodes that node X is linked to.

$$\text{Graph Distance} = \text{Length of Shortest Path} \quad (1)$$

$$\text{Common Neighbors} = |\Gamma(x) \cap \Gamma(y)| \quad (2)$$

$$\text{Jaccard's Coefficient} = |\Gamma(x) \cap \Gamma(y)| / |\Gamma(x) \cup \Gamma(y)| \quad (3)$$

$$\text{Preferential Attachment} = |\Gamma(x) \cdot \Gamma(y)| \quad (4)$$

Of the methods described, work has been done to evaluate how well they are performing. In 2003, Liben-Nowell and Kleinberg compared these solutions along with a few others, and they found that although many are able to outperform a random predictor, the best of the algorithms was only correct on 16% of its predictions. This suggests that there is useful information for making predictions in the graphs, but more precise methods are needed.

Cohen's κ

Here we present a new method for the prediction of links in a network, which applies Cohen's κ (κ) to the link prediction or missing link problem. κ is a measure originally created to evaluate agreement between two judges (Cohen, 1960), and recently has been applied to other problems, in particular cluster analysis. Warrens (2008) demonstrated that κ is equivalent to the Hubert-Arabie Adjusted Rand Index (Hubert &

Arabie, 1985) when the cluster labels are known ahead of time. The Adjusted Rand Index is the preferred method for validating clusters (Steinley, 2004).

Cohen's $\kappa=2(ad-bc)/p1q1+p2q2$ (5)

Table 1: The contingency table from which κ is calculated for each pair of nodes X and Y, using formula (5).

	Link	No Link	
Link	a	b	p1
No Link	c	d	q1
	p2	q2	

In this application, κ gives a similarity measure to each pair of nodes. κ is calculated from a contingency table as seen in table 1. Here the value for a is the number of links to other nodes that the two nodes being evaluated have in common, and d is the number of links they both are missing. Together, a and d represent a measure of agreement. Disagreement is represented by values b and c, which indicate where one node has a link and the other does not. κ is calculated by finding the difference between agreement and disagreement and dividing by the expected difference. In this application, κ gives each potential link a score between -1 and 1, where a score greater than zero indicates that the probability of a link being there is greater than random chance. This is a benefit of this method because it gives a defined cutoff value to detect links.

METHOD

Simulation 1: Link Prediction

For this study we generated data sets with a block model structure to best represent the clusters found in real social networks. The density within blocks represents the number of links within each cluster of nodes. Three levels of within density were

used: 60%, 75% and 90%. The density between blocks is the amount of links between clusters, also set at three levels: 10%, 25% and 40%. An additional 50%-50% combination was created as well to represent a random graph. We looked at three sizes of data sets, containing 2, 4, or 6 blocks (each containing 20 nodes).

Each data set was split by randomly removing a percentage of the links to form a test data set, leaving the training data. This was done on three different levels, removing 10%, 30% and 50% of the links. With this setup, the goal of the algorithms is to find the "missing links" that are the test data set. Ten data sets were created at every combination of levels of cluster number, density within and between and number of links removed. The four existing measures of node similarity were compared to Cohen's ν.

Cohen's κ gives a definite cutoff for the number of links it predicts, since any score greater than zero is a link and less than zero is no link. This is not the case with the other algorithms; meaningful cutoff values are not provided. To compare the links "predicted" by these algorithms, the top scores were chosen as the links, with the number corresponding to the number of links originally removed.

After running the link prediction algorithms, the effectiveness of each was assessed using the area under the ROC curve statistic. The area under the ROC curve (AUC) is a statistic that is often used to evaluate the performance of link prediction methods (Fawcett, 2006; Hanley & McNeil, 1982). It is calculated by first creating the ROC curve which compares the rate of true positives to false positives at different cutoffs levels for the scores. The area under this “curve” is taken to find the AUC statistic. The AUC can be interpreted as the probability that the algorithm will detect a true link over a false link. This can evaluate how much better than chance this method is predicting,

because an AUC of .5 is expected from random guessing. Because it evaluates each method this way, it allows all methods to be compared without the issue of arbitrarily selecting a cutoff value.

Simulation 2: Detecting Incorrect Links

We ran a second set of simulations to look into another potential use of the Cohen's κ algorithm: in finding incorrect links in the data. In the link prediction context, the κ values at or below 0 indicate that a link should not be added. If links present in the test data are assessed with κ, it may point out incorrect links. To do this, data sets were simulated in the same manner as before with the same levels of between and within density and number of clusters. Then random links were generated and added to the data set. κ values were then calculated for the random links added and the links already present in the test set.

RESULTS

Over the different simulated data sets, the Cohen's κ statistic's performance varied as expected. The AUC scores for the results can be found in Table 2. Modifying the number of clusters in the data sets had some effect on the performance of κ ($F=79.656$, $p<.01$), with the greater number of clusters in the data set corresponding to poorer performance measured by the AUC statistic.

The effect of the number of links removed was significant ($F=20.674$, $p<.01$). However, this difference was not as large as the others. Follow up tests using Tukey's

HSD showed no difference between cases with 30% and 10% of the links removed (p=.098).

Table 2: AUC statistics for the five algorithms averaged over 10 iterations. An asterisk indicates the highest AUC for each row.

AUC Scores					
	Graph Distance	**Common Neighbors**	**Preferential Attachment**	**Jaccard's Coefficient**	**Cohen's κ**
Number of Clusters					
2	0.63	0.67	0.42	0.72	0.72*
4	0.54	0.64	0.45	0.67	0.68*
6	0.52	0.62	0.46	0.64	0.64*
Density Within					
60%	0.55	0.61	0.45	0.64	0.65*
75%	0.56	0.64	0.44	0.67	0.68*
90%	0.58	0.67	0.43	0.71	0.71*
Density Between					
10%	0.56	0.73	0.44	0.76	0.76*
25%	0.58	0.62	0.45	0.66	0.66*
40%	0.55	0.57	0.45	0.61	0.62*
Links Removed					
10%	0.52	0.66	0.44	0.70*	0.7
30%	0.57	0.64	0.44	0.68	0.68*
50%	0.6	0.62	0.45	0.64	0.66*
Average	**0.56**	**0.64**	**0.44**	**0.67**	**0.68***

As the density within clusters increased, κ's performance improved as expected ($F=49.15$, $p<.01$). Similarly, as the density between clusters was increased the algorithm had more difficulty finding correct links ($F=544.99$, $p<.01$). The density between clusters seemed to be particularly influential, as the average AUC values dropped from about .7605 to .6151 when the density between was increased from 10% to 40%.

The random graph, 50% within and 50% between density, was analyzed separately from the other conditions. The overall average AUC for κ was around .5853. Here, κ is not predicting much better than chance, but this is expected since there is not a cluster structure.

An ANOVA comparing all five algorithms revealed there is a significant difference between their AUC scores ($F=1164.36$, $p<.01$). Looking further into the AUC all of the algorithms, given in table 2, Cohen's κ seems to have the strongest performance under most of the variations in the data sets, with Jaccard's Coefficient performing second best. Follow up tests using Tukey's HSD found no difference between the Jaccard's Coefficient and Cohen's κ AUCs ($p=.808$). Overall, the poorest performance is seen from the preferential attachment algorithm, where the AUC scores suggest it is actually worse than random guessing. Graph distance falls pretty close to the "random chance" cutoff of the AUC statistic.

To further investigate the relationship between these algorithms, a rank correlation was performed on the scores they produced. Table 3 shows the correlation, averaged over all iterations (excluding the random graph conditions). Cohen's κ is most closely related to Jaccard's Coefficient and common neighbors. This is an expected result given that Jaccard's Coefficient and Cohen's κ use the common neighbors information in their scores.

Table 3: Rank correlations between the scores given by each of the five algorithms.

	Graph Distance	Common Neighbors	Preferential Attachment	Jaccard's Coefficient	Cohen's κ
Graph Distance	1	-0.13	-0.19	-0.08	-0.07
Common Neighbors		1	0.46	0.95	0.81
Preferential Attachment			1	0.2	-0.08
Jaccard's Coefficient				1	0.94
Cohen's κ					1

Table 4 shows the average percentage of correctly classified links. The main focus for this simulation is on the correctly classified random links, which are those that were given a negative κ. As in link prediction, κ seems to do best when there are fewer clusters ($F=292.66$, $p<.01$). Having less of an effect was the number of links added. As more random links were included performance did decrease, though this was not as influential ($F=6.495$, $p<.01$).

Table 4: Percentages of links correctly classified by Cohen's κ, when a true link was given a positive κ or when the random links added were given κ less than or equal to zero.

Percentages for Incorrect Link Simulation

		True Links	Random Links
		$\kappa > 0$	$\kappa \leq 0$
Number of Clusters	2	64.35%	74.74%
	4	65.25%	62.01%
	6	63.75%	54.97%
Density Within	60%	64.10%	54.63%
	75%	64.22%	64.34%
	90%	65.04%	72.75%
Density Between	10%	80.16%	65.73%
	25%	63.28%	62.81%
	40%	49.91%	63.17%
Random Links Added	10%	68.12%	66.05%
	30%	65.29%	63.42%
	50%	59.94%	62.25%
Average		64.45%	63.90%

Looking at the cluster structure of the data, the density within clusters also seems to be an important factor ($F=211.574$, $p<.01$). It is easiest to detect incorrect links when the original clusters are better defined within. Density between, however, does not seem to have this same relationship. The ANOVA indicated the effect was significant ($F=4.331$, $p=.01$), but not strong. The relationship is not as clear as with the link

prediction simulation. The percentage of random links correctly classified at 10%, 25%, and 40% density within changes from 66% to 62% and then 63%. Looking at the original links that were correctly classified (with a positive κ) the relationships with density within and between is almost opposite, where the density within does not have a significant effect at all ($F=.354$, $p=.7$), but the density between is very influential ($F=1262.03$, $p<.01$).

DISCUSSION

Looking at Cohen's κ's performance over the variations of network data sets, we can start to define the situations where κ might be best applied to detect missing links. AUC scores over the variations in cluster strength (density within and between) show that κ's ability to find links is highest when the density within is high and density between is low, creating a stronger cluster structure. The density between clusters was particularly influential. It is also interesting to note that the AUC values for Cohen's κ on the random graphs were consistently greater than .5, when we would expect its scores to be .5 on average.

The greater number of clusters included the data set was associated with a lower AUC for κ. Since the number of nodes per cluster was held constant at 20, more clusters is associated with a larger data set. It is possible that the decreasing performance was because of more mistakes being made, with the greater number of links to predict.

As the number of links removed from the data set increased, κ's AUC decreased, though not significantly until 50% were removed. This is an ideal characteristic of a link

prediction algorithm, since it continues to recover the structure of the data set even with increasing amounts of missing data.

The comparison between Cohen's κ and the other four algorithms placed κ first in all but one of the rows of table 2, (Jaccard's Coefficient had the top AUC average for the cases where 10% of the links were removed). The follow up to the Anova test was not able to show that Cohen's K is significantly different from Jaccard's Coefficient, and the two are very similar in their predictions as demonstrated with the rank correlation. While the scores may be similar, Cohen's κ does have its benefit of a natural cutoff value ($\kappa>1$) that the other algorithms including Jaccard's Coefficient lack.

From the second simulation, detecting incorrect links, we can see the potential for κ's use in cleaning up data sets. On average, κ correctly classified 63.9% of the random links and 64.45% of the original links. Similar to the results of the link prediction simulation, fewer clusters was associated with better classification of the random links added to the data set, indicating more difficulty in larger data sets. Classification as more links were added decreased, but not with a very strong effect.

Looking at the classification given the varying cluster structure, we see some interesting effects. It seems that the correct classification of random links added to the data set is strongly influenced by the density within clusters, but not by the density between. This is the opposite of the corresponding tests for the correct classification of the original links, where density between was very important and density within not even significant. Generally, it seems a stronger cluster structure is associated with better performance when the goal is to clean up a data set, but the density within and between influence different factors.

BIBLIOGRAPHY

Cohen, J. (1960). A coefficient of agreement for nominal scales. Educational and Psychological Measurement, 20(1), 37–46.

Duan, D., Yuhua, L., Ruixuan, L., & Zhengding, L.(2012). Incremental K-clique clustering in dynamic social networks. Artifical Intelligence Review, 38(2, 129–147.

Fawcett, T. (2006). An introduction to ROC analysis.Pattern Recognition Letters, 27, 861–847.

Hanley, J. A., & McNeil, B. J. (1982). The Meaning and Use of the Area under a Receiver Operating Characteristic (ROC) Curve. Radiology, 143, 29–36.

Hubert, L., & Arabie, P. (1985). Comparing partitions. Journal of Classification, 2, 193–218.

Krebs, V. E. (2002). Uncloaking terrorist networks. First Monday, 7(4).

Liben-Nowell, A. & Kleinberg, J. (2003). The link-prediction problem for social networks. Journal of the American Society for Information Science and Technology, 58(7), 1019–1031.

Mishra, N., Schreiber, R., Stanton, I. & Tarjan, R. E. (2009). Finding strongly-knit clusters in social networks. Internet Mathematics, 5(1), 155–174.

Newman, M. E. J. (2001). The structure of scientific collaboration networks. Proceedings of the National Academy of Sciences USA, 98, 404–409.

Ressler, S. (2006). Social network analysis as an approach to combat terrorism: Past, present, and future. Homeland Security Affairs, 2(2).

Shibata, N., Kajikawa, Y., & Sakata, I. (2012). Link prediction in citation networks. Journal of the American Society for Information Science and Technology, 63(1), 78–85.

Steinley, D. (2004). Properties of the Hubert-Arabie adjusted rand index. Psychological Methods, 9(3), 386–396.

Warrens, M. J. (2008). On the equivalence of Cohen's kappa and the Hubert-Arabie adjusted rand index. Journal of Classification, 25, 177–183.

Wasserman, S. & Faust, K. (1994). Social network analysis: Methods and applications. Cambridge University Press, Cambridge.

Zhu, X., Gerstein, M., & Snyder, M. (2007). Getting connected: analysis and principles of biological networks. Genes & Development 21, 1010–1024.