

Data Clustering with a Relational Push-Pull Model *

Adam Anthony and Marie desJardins

Department of Computer Science and Electrical Engineering
University of Maryland Baltimore County
1000 Hilltop Circle, Baltimore, MD 21250
(410) 747-2593
{aanthon2 , mariedj}@cs.umbc.edu

Abstract

Relational data clustering is the task of grouping data objects together when both attributes and relations between objects are present. We present a new generative model for relational data in which relations between objects can have either a binding or separating effect.

Introduction

Relational data clustering is a form of relational learning that clusters data using the relational structure of data sets to guide the clustering. Many approaches for relational clustering have been proposed recently with varying results. The common assumption in most of this research is that *relations have a binding tendency*. That is, edges are assumed to appear more frequently within clusters than between clusters.

This binding quality may be too strong an assumption. Bhattacharya & Getoor (2005) acknowledge that it is possible for a relation to provide “negative evidence,” where the presence of a relation between two objects implies that the objects belong in different clusters. If most of the edges in a relational set provide negative evidence, then this set should be considered to have a separating, rather than a binding, tendency. As a motivating example, consider a social network of university students containing both men and women. If the network is partitioned by gender, edges for a dating relation will appear most frequently between the clusters. This can be visualized as a lattice structure between the two clusters that “pushes” them apart. Edges for a roommate relation will appear most frequently within the clusters. This binding relation can be visualized as a net that “pulls” the objects in a cluster closer together.

Given that a relation can have either of these two tendencies, accurate clustering in relational data requires determining whether each relation tends to bind or to separate objects. Bhattacharya & Getoor (2005) and others have used domain-specific knowledge to identify and process negative evidence. The contribution of our research is a model that handles the case where the tendency of an edge is unknown, allowing its tendency to be inferred.

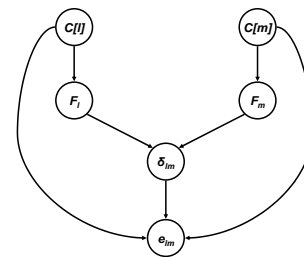


Figure 1: Graphical model of the dependencies in the Relational Push-Pull Model

Related Work

There have been a number of proposed generative models for relational data in recent research. The most similar model to the RPPM is the Latent Group Model (LGM) proposed by Neville & Jensen (2006). LGMs have the same dependency for object features, but edge existence depends on what they call *latent groups* that influence the feature values of objects that belong to a specific group. Another generative model proposed recently is by Kemp *et al.* (2006) who propose the Infinite Relational Model, an extension of stochastic block models to handle the case where the number of clusters is unknown. Their work has very general applications for learning concepts in data and extends well beyond clustering. Finally, Kubica *et al.* (2002) present a model where links exist as the result of some event (e.g., a phone call or a meeting). They use this model to detect groups of associated people from a social network. Their model is effective in representing such a situation, but does not generalize well to other problems.

The Relational Push-Pull Model

We propose the Relational Push-Pull Model (RPPM), a Bayesian model that can be used to find clusterings in relational data. Figure 1 is a graphical representation of the model that shows the dependencies for a pair of objects, l and m . Object features depend on the cluster the object belongs to, modeled as a mixture of Gaussians. To formulate an edge distribution, we use the concept of *existence uncertainty* (Getoor *et al.* 2002) for relations. For each pair of ob-

*This work supported by NSF grant #0545726
Copyright © 2007, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

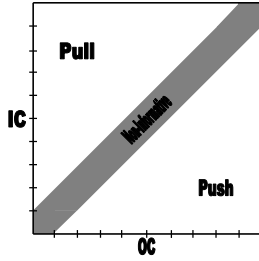


Figure 2: The relationship between IC and OC.

jects l and m in a data set, we calculate the probability that a link exists between the objects. The value of this probability depends on both the distance between l and m as well as the cluster membership of l and m .

When considering the cluster membership separately, we specify two constant probabilities, IC and OC . IC is the probability of an edge existing between l and m if the objects are in the same cluster, and OC is the probability that the edge exists if they are in a different cluster. Figure 2 shows the relationship between the two values. Pull-type relations exist when values of IC are much greater than OC . Likewise, push-type relation graphs exist when values of IC are much smaller than OC . The gray region illustrates that when IC is very similar to OC , the relation graph's tendency is ambiguous.

We can get the benefits of both approaches if we make edge existence dependent on both cluster membership and the features of the objects the edge connects. We model this dependence with the following formula:

$$(1) \quad \begin{aligned} Pr(e_{lm} \mid l \in C_i, m \in C_j, \delta_{lm}, \lambda_{I_t}, \lambda_{O_t}) \\ = \begin{cases} IC(\delta_{lm}) = f(f_l, f_m) - (1 - \lambda_{I_t}) & \text{if } C_i = C_j; \\ OC(\delta_{lm}) = f(f_l, f_m) - (1 - \lambda_{O_t}) & \text{if } C_i \neq C_j. \end{cases} \end{aligned}$$

Now, instead of being constant, IC and OC are functions that are selected based upon the cluster membership of l and m . λ_{I_t} and λ_{O_t} are parameters that determine which of the two functions is higher. Using Equation 1, we can compute the probability distribution of an entire relational graph as a product of each edge's probability of existence.

Our goal is to use the RPPM to find an accurate clustering. We assume that the number of clusters is known, though this assumption could be relaxed by introducing additional parameters in the model. The assignment of each object to a cluster, the Gaussian mixture parameters (weight, mean and covariance for each cluster) are all unknown. Because of the large number of unknown values, we use an implementation of simulated annealing to search for optimal values because there could be several local maxima. From the model, we derive the following objective function:

$$Pr(C \mid O, R) = \alpha Pr(R \mid C, O) Pr(C \mid O).$$

Which is the probability that a clustering C is accurate, where R is a relational graph, O is the set of objects.

Table 1 shows the results from an experiment using artificial data where λ_{I_t} and λ_{O_t} are fixed, and the Gaussian

Assumption:	Correct	All Pull	All Push
Graph and Features	0.9576	0.6640	0.5696
Graph Only	0.9564	0.6676	0.5768
Features Only	0.6432	0.6432	0.6432

Table 1: The impact of a poor hypothesis.

Mixture parameters vary. The data has three relation types, one push-type, one pull-type, and one that is neither push nor pull. The results show that making an incorrect assumption about relation types can harm performance dramatically and motivates the need for a model where edges may either appear more frequently within or between clusters, depending on the situation. The results are averaged over 10 trials. The key observation is that assuming relations are all of one type results in a clustering that is not statistically better than ignoring the relations altogether. But correct assumptions about the relation types do result in a significant improvement.

Conclusion

The Relational Push-Pull Model is a unique framework that models the specific tendencies that a relation can have with regards to a specific clustering. This model expands the space of possible clusterings beyond previous works that considered all relations to be pull-type relations. By searching in this expanded space, we hypothesize that better clusterings may be discovered for different data sets.

Probabilistic models are useful tools for solving complex problems, but are limited by their speed and somewhat complex and difficult to interpret results. Future work will involve the investigation of discrete relational data clustering methods and how they perform when the underlying data model changes.

References

- Bhattacharya, I., and Getoor, L. 2005. Entity resolution in graph data. Technical Report CS-TR-4758, University of Maryland.
- Getoor, L.; Friedman, N.; Koller, D.; and Taskar, B. 2002. Learning probabilistic models of link structure. *Journal of Machine Learning Research* 3:679–707.
- Kemp, C.; Tenenbaum, J. B.; Griffiths, T. L.; Yamada, T.; and Ueda, N. 2006. Learning systems of concepts with an infinite relational model. In *Proceedings of the Twenty-First National Conference on Artificial Intelligence*, 381 – 388. Menlo Park, California: American Association for Artificial Intelligence.
- Kubica, J.; Moore, A.; Schneider, J.; and Yang, Y. 2002. Stochastic link and group detection. In *Proceedings of the Eighteenth National Conference on Artificial Intelligence*, 798–804. AAAI Press/MIT Press.
- Neville, J., and Jensen, D. 2006. Leveraging relational autocorrelation with latent group models. In *Proceedings of the Fifth IEEE International Conference on Data Mining*.