A Shrinkage Approach for Modeling Non-Stationary Relational Autocorrelation

Pelin Angin
Purdue University
Department of Computer Science
pangin@cs.purdue.edu

Jennifer Neville
Purdue University
Departments of Computer Science and Statistics
neville@cs.purdue.edu

Abstract

Recent research has shown that collective classification in relational data often exhibit significant performance gains over conventional approaches that classify instances individually. This is primarily due to the presence of autocorrelation in relational datasets, meaning that the class labels of related entities are correlated and inferences about one instance can be used to improve inferences about linked instances. Statistical relational learning techniques exploit relational autocorrelation by modeling global autocorrelation dependencies under the assumption that the level of autocorrelation is stationary throughout the dataset. To date, there has been no work examining the appropriateness of this stationarity assumption. In this paper, we examine two real-world datasets and show that there is significant variance in the autocorrelation dependencies throughout the relational data graphs. We develop a shrinkage technique for modeling this non-stationary autocorrelation and show that it achieves significant accuracy gains over competing techniques that model either local or global autocorrelation dependencies in isolation.

1. Introduction

The presence of *autocorrelation* provides a strong motivation for using relational techniques for learning and inference. Autocorrelation is a statistical dependency between the values of the same variable on related entities, which is a nearly ubiquitous characteristic of relational datasets. For example, pairs of brokers working at the same branch are more likely to share the same fraud status than randomly selected pairs of brokers. The presence of autocorrelation offers a unique opportunity to improve model performance because inferences about one object can be used to improve inferences about related objects. Recent work in relational domains has shown that *collective inference* over an entire dataset can result in more accurate predictions than conditional inference for each instance independently (see e.g., [2, 11, 20]) and that the gains over conditional models increase as autocorrelation increases [8].

There have been a number of approaches to modeling autocorrelation and the success of each approach depends on the characteristics of the target domain. When there is overlap between the data used for learning and the dataset where the model will be applied (i.e., some instances appear in both graphs), then models can reason about *local* autocorrelation dependencies by incorporating the *identity* of instances in the data [15]. For example, in relational data with a temporal ordering, we can learn a model on all data up to time t, then apply the model to the same dataset at time $t + x$, inferring class values for the instances that appear after t. On the other hand, when the training set and test set are either disjoint or largely separable, then models must generalize about the dependencies in the training set. This is achieved by learning *global* autocorrelation dependencies and inferring the labels in the test set *collectively* [19, 12, 16]. For example, a single website can be used to learn a model of page topics, then the model can be applied to predict page topics in other (separate) websites.

One limitation of models that represent and reason with global autocorrelation is that the methods assume the autocorrelation dependencies are *stationary* throughout the relational data graph. To date, there has been no work examining the appropriateness of this assumption. We conjecture that many real-world relational datasets will exhibit significant variability in the autocorrelation dependencies throughout the dataset and thus violate the assumption of stationarity. The variability could be due to a number of factors, including an underlying latent community structure with varying group properties or an association between the graph topology and autocorrelation dependence. When the autocorrelation varies significantly throughout a dataset, it may be more accurate to model the dependencies *locally* rather than *globally*. In this case, identity-based approaches may more accurately capture local variations in autocorrelation. However, identity-based approaches, by definition, do not generalize about dependencies but focus on the char-

P. Angin and J. Neville. A Shrinkage Approach for Modeling Non-stationary Relational Autocorrelation. In Proceedings of the 8th IEEE International Conference on Data Mining (ICDM'08), pp. 707-712, 2008.

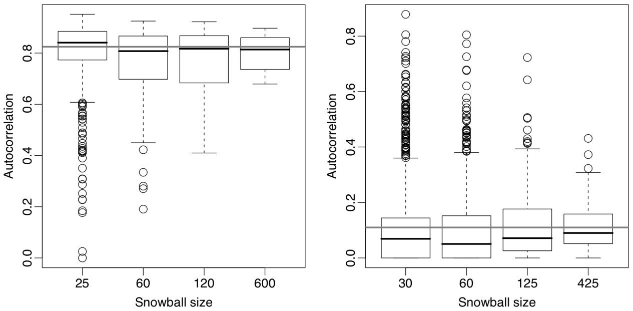
acteristics of a single instance in the data. This limits their applicability to situations when there is significant overlap between the training and test sets, because they can only reason about the identity of instances that appeared in the training set. When there is insufficient information about an instance in the training set, a *shrinkage* approach, which backs off to the global estimate, may be better able to exploit the full range of autocorrelation in the data.

In this work, we develop an approach to modeling nonstationary autocorrelation in relational data. Our approach combines local and global dependencies in three shrinkage models. We evaluate our models on two real-world relational datasets and one synthetic dataset, comparing to models that reason with only local or global dependencies in isolation, as well as two state-of-the-art relational classification algorithms, and show that the shrinkage models achieve significantly higher accuracy over all three datasets.

2. Relational autocorrelation

Relational autocorrelation refers to a statistical dependency between values of the same variable on related objects. More formally, we define relational autocorrelation with respect to an attributed graph $G = (V, E)$, where each node $v \in V$ represents an object and each edge $e \in E$ represents a binary relation. Autocorrelation is measured for a set of instance pairs P_R related through paths of length l in a set of edges E_R: $P_R = \{(v_i, v_j) : e_{ik_1}, e_{k_1k_2}, ..., e_{k_lj} \in E_R\}$, where $E_R = \{e_{ij}\} \subseteq E$. It is the correlation between the values of a variable X on the instance pairs $(v_i.x, v_j.x)$ such that $(v_i, v_j) \in P_R$. Autocorrelation is a nearly ubiquitous characteristic of relational datasets. For example, recent analysis of relational datasets has reported autocorrelation in the topics of hyperlinked web pages [2], the topics of coreferent scientific papers [20], and the industry categorization of corporations that share board members [11].

When there are dependencies among the class labels of related instances, relational models can exploit those dependencies in two ways. The first technique takes a local approach to modeling autocorrelation. More specifically, the probability distribution for the target class label (Y) of an instance i can be conditioned not only on the attributes (X) of i in isolation, but also on the *identity* and observed attributes of instances[1] $(R = \{1, ..., r\})$ related to i: $p(y^i|\mathbf{x}^i, \{\mathbf{x}^1, ..., \mathbf{x}^r\}, \{id^1, ..., id^r\}) = p(y^i|x_1^i, ..., x_m^i, x_1^1, ..., x_m^1, \quad ... \quad x_1^r, ..., x_m^r, id^1, ..., id^r)$. This approach assumes that there is sufficient information to estimate $P(Y|ID)$ for all instances in the data. However, it is unlikely that we can accurately estimate this probability for all instances—for instances that are only in the test set there is no information to estimate this probability distribution and for instances in the training data that only link to a few labeled instances our estimates will have high variance.



(a) Cora

(b) IMDB

Figure 1. Empirical autocorrelation variation.

The second technique exploits autocorrelation by including related class labels as dependent variables in the model. More specifically, the probability distribution for the target class label (Y) of an instance i can be conditioned not only on the attributes (X) of i in isolation, but also on the attributes and class labels of instances $(R = \{1, ..., r\})$ related to i: $p(y^i|\mathbf{x}^i, \{\mathbf{x}^1, ..., \mathbf{x}^r\}, \{y^1, ..., y^r\}) = p(y^i|x_1^i, ..., x_m^i, x_1^1, ..., x_m^1, \quad ... \quad x_1^r, ..., x_m^r, y^1, ..., y^r)$. Depending on the overlap between training and test sets, some of the values of $\mathbf{y}^R$ may be unknown when inferring the value for y^i. In this case, *collective inference* techniques can be used to jointly infer the unknown y values. Collective inference approaches typically model the autocorrelation at a global level (e.g., $P(Y|Y^r)$).

One key assumption of global autocorrelation models is that the autocorrelation dependencies do not vary significantly throughout the data. We have empirically investigated the validity of this assumption on two real-world datasets. The first dataset is Cora, a database of computer science research papers extracted automatically from the web using machine learning techniques [9]. We investigated the variation of autocorrelation dependencies in the set of 4,330 machine-learning papers in the following way. We first calculated the autocorrelation among the topics of pairs of cited papers using Pearson's corrected contingency coefficient [17]. The global autocorrelation is 0.824. Then we used snowball sampling [6] to partition the citation graph into sets of roughly equal-sized subgraphs and calculated the autocorrelation within each subgraph.

Figure 1(a) graphs the distribution of autocorrelation across the subgraphs, for snowball samples of varying size. The solid gray line is the global level of autocorrelation. At the smallest snowball size (25), the variation of local autocorrelation is clear. More than 25% of the subgraphs have autocorrelation significantly higher than the global level. In addition, there are a non-trivial number of subgraphs that have significantly lower levels of autocorrelation.

The second dataset is the Internet Movie Database

[1] Here we use superscripts to refer to instances and subscripts to refer to attributes.

(IMDB; www.imdb.com). Our sample consists of movies released between 1980-2006. Figure 1(b) graphs the distribution of autocorrelation between the *isblockbuster* attribute of related movies, which indicates whether a movie has total earnings of more than $\$30mil$ in box office receipts. Here we consider two movies to be related if they have at least one actor in common. The same procedure described above was used to generate subgraphs of different sizes. We observe that non-stationarity in autocorrelation is also apparent in this dataset. The global autocorrelation (0.112) is low, but more than 30% of the subgraphs have significantly higher local values at a snowball size of 30.

3. Related Work

Classification in the presence of sparse training data is an important problem in many domains. Previous research on shrinkage techniques has exploited background domain knowledge to create hierarchies of parameters to develop more accurate estimation methods that combine information from multiple levels of the hierarchy (see e.g., [1, 10]). This work has primarily focused on modeling independent and identically distributed (i.i.d.) data. To the best of our knowledge, our algorithms are the first shrinkage methods developed for relational domains.

In spatial statistics, there have been recent efforts to model non- stationary autocorrelation process in geospatial datasets [7, 5]. These approaches are similar in spirit to the shrinkage models that we propose in this paper, in that they combine local estimates of autocorrelation to represent non-stationary distributions. However, the spatial models use repeated measures at a single site (i.e., node) to estimate local parameters and then they combine estimates based on geospatial location instead of using global parameter estimates.

Relational models that estimate global levels of autocorrelation were initially developed for across-network tasks, where the training data is a full-labeled disjoint graph and there is no opportunity to exploit local measures of autocorrelation in the test set (since it is unlabeled and disjoint from the training set). Examples of models that estimate global levels of autocorrelation include: probabilistic relational models [4], relational Markov networks [19], and relational dependency networks [12].

Relational models that exploit local autocorrelation dependencies generally focus on modeling the identity and characteristics of *hub* (i.e., high degree) nodes in the training set [18, 15]. If the hub nodes tend to link to objects with the same class label (i.e., they exhibit autocorrelation), and there is significant overlap between the training and test set, then application of these models has been shown to improve classification performance.

In this work we aim to combine the strengths of both local and global relational techniques to model non-stationary relational autocorrelation. Here we use a two-level hierarchy for shrinkage (i.e., local and global). However, future work will consider a more continuous hierarchy that uses varying-sized snowball autocorrelation estimates between the local and global levels.

4. Classification models

The basic model we present in this work is a local classification model employing a simple Naive Bayes classification scheme, where the probability of class is conditioned on the characteristics of the nodes related to that instance (which we call the *neighbor* nodes) and each neighbor is conditionally independent given the class. The probability that a node i has class label y is defined as: $p(y^i|N(i)) \propto p(y) \cdot \prod_{j \in N(i)} p(y|j)$, where $N(i)$ represents the set of labeled *neighbor* nodes of i, $p(y)$ is the prior probability of class label y, and $p(y|j)$ is the conditional probability of y given that a node is linked to instance j.

All of the following classification models that we explore in this paper employ this basic model and differ in the method used to estimate $p(y|j)$.

4.1. Local model

The *local* model uses maximum likelihood estimation considering only local dependencies between class labels of related instances for classification. Let j be a *neighbor* of the node to be classified and $I_y(k)$ be an indicator function, which returns 1 if node k has class label y and 0 otherwise. Then the probability that a node i has class label y, conditioned on its neighbor node j is: $p_L(y|j) = \sum_{k \in N(j)} I_y(k) \ / \ |N(j)|$.

4.2. Global model

The *global* model uses maximum likelihood estimation considering only global dependencies between class labels of related instances to classify nodes. Let Y be the set of all possible class labels and $G_{y^l y^m}$ be the set of linked node pairs in the whole network, where the class label of the first node in the pair is y^l and that of the other is y^m. Then the probability that the node i has class label y, conditioned on its neighbor node j with class label y^j is: $p_G(y|j) = |G_{yy^j}| \ / \ \sum_{y' \in Y} |G_{y'y^j}|$.

4.3. Shrinkage models

Here we present three classification models accounting for non-stationary autocorrelation in relational data. The models are designed for domains with overlapping training and test sets, as they use identity-based *local* dependencies as a significant component. The shrinkage models we propose below use a combination of the global p_G and local p_L dependencies to classify nodes in a network.

Regularization with Dirichlet prior

This model uses the *global* component (p_G) as a prior, which is then updated with the local observed data. The Dirichlet prior is defined, with a scale parameter β, as follows: $\alpha_y = \beta \cdot p_G(y|j)$. Using the prior, the probability that a node i has class label y, conditioned on its neighbor node j is: $p_D(y|j) = [\ \sum_{k \in N(j)} I_y(k) + \alpha_y - 1\ ]\ /\ [\ |N(j)| + \sum_{y' \in Y}(\alpha_{y'} - |Y|)\ ]$.

Here the scale parameter β controls the contribution of the *global* prior relative to that of the *local* component, with a small value resulting in reliance primarily on the local dependencies. This factor also offsets for the lack of sufficient labeled instances in the local neighborhood, providing more effective use of the global level of autocorrelation in such cases. For the experiments below, we set $\beta = 20$.

Two-branch rule

In this model, we use a weighted combination of the local (p_L) and global (p_G), where the weight varies based on the number of neighbors available for the local estimates. The two-branch shrinkage model defines the probability that node i has class label y, conditioned on its neighbor j as follows: $p(y|j) = \alpha_j \cdot p_L(y|j) + (1 - \alpha_j) \cdot p_G(y|j)$. As α_j is the weight we assign to the local component, we would like its value to be high when there is sufficient number of labeled instances at the local surrounding of the neighbor being considered. We define α_j as follows: $\alpha_j = \gamma$ if $|N(j)| \geq 5$ and $1 - \gamma$ otherwise.

A single parameter γ is chosen via cross-validation and used to determine the values of α used in the model. We use the following cross-validation procedure:

```
$S_\gamma$ = $\emptyset$
Repeat five times:
  • Randomly select 50% of the objects in the training set,
    mark them as unlabeled and use as the test set; use the
    remaining data as the training set.
  • For every value of $\gamma$ in the set {0.99, 0.9, 0.8, 0.7, 0.6,
    0.5}:
      – Estimate the two-branch model on the training
        set using $\gamma$.
      – Apply the model to the test set.
      – Calculate AUC.
  • Add the $\gamma$ with highest AUC to $S_\gamma$
$\gamma$ = average of values in $S_\gamma$.
```

The choice of a threshold of five on the number of labeled neighbors of node j is based on our background knowledge regarding the number of data points needed to estimate parameters accurately.

Chernoff bound on sample confidence

This model, which is similar to the two-branch model in terms of its general structure, provides a different way of deciding about the weight to place on the local estimate. Our approach is motivated by the classic sample size problem, which aims to find the smallest sample size n such that: $pr\{|\hat{p}_n - p| < \epsilon\} > 1 - \delta$.

It is a well-known result that Chernoff bounds can be used to compute a lower bound on n which guarantees that the above will hold for any p. Thus, for a binomial distribution with parameter p, we can compute the confidence level $c = 1 - \delta$ for a given sample size n and error threshold ϵ [3]: $c = 1 - 2e^{-2n\epsilon^2}$.

In this third shrinkage model, we compute the probability that a node i has class label y, conditioned on its neighbor node j as: $p_C(y|j) = c \cdot p_L(y|j) + (1 - c) \cdot p_G(y|j)$. Here, n is the number of labeled instances linked to node j and ϵ is a threshold on the acceptable level of parameter error, which we set to 0.4 in the experiments below.

5. Experimental Evaluation

Experimental evaluation of the proposed models was done on two real-world datasets and a synthetic dataset. The performance metric used for all evaluations is the area under the ROC curve (AUC). For each classification task, a subset of the nodes in the network is treated as *unlabeled* and only the *labeled* instances are taken into account while learning the models.

In addition to the local and global baselines, we include two competing relational classification approaches for comparison. We compare to relational dependency networks (RDNs) [14] as an example of a state-of-the-art statistical relational learner that uses global estimates of autocorrelation. To make a fair comparison with the shrinkage models (and to limit variation due to attribute correlation), we only consider the class labels of related instances in the model, we do not use the other available attributes in the dataset.

The second competing approach is a Gaussian random field (GRF) [21]. GRFs are a graph-based semi-supervised learning approach for i.i.d. data that constructs a graph of conceptual links among independent instances to encode similarities among their attributes. However, the method can also be applied in our setting to propagate label information in an existing relational graph, which represent physical connections or associations among instances. The method assumes autocorrelation exists in the graph and propagates label information accordingly. We include this as a comparison to assess whether it is useful to *learn* the autocorrelation dependencies in the data.

In the figures, we use *Shrinkage2B* to refer to the shrinkage method with the two-branch rule, *ShrinkageD* to refer to the shrinkage method using regularization with Dirichlet

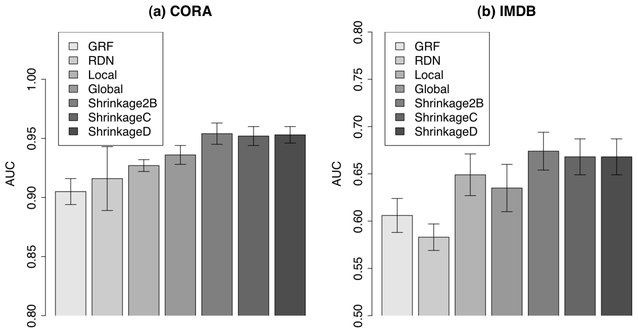


Figure 2. Classification results

prior and *ShrinkageC* to refer to the shrinkage model using Chernoff bounds on the sample size confidence parameter.

5.1. Cora experiments

The Cora dataset is a collection of computer science research papers as described in Section 2. The task for the Cora dataset was to classify research papers published between 1994-1998, in the *Machine Learning* field, into seven sub-topics based on their citation relations. We used temporal sampling, where the training and test sets were based on the *publication year* attribute of the papers in the dataset. For each year t, the task was to predict the topics of the papers published in t from models learned on the set of papers published before t.

Figure 2a provides a comparison of the average performance of all classification models on the Cora classification task. We performed one-tailed, paired t-tests to measure the significance of the difference between the AUC values we observed for the different classification methods. All shrinkage models proposed were found to be significantly better than all other models ($p < 0.05$) and they achieve more than a 25% reduction in error.

5.2. IMDB experiments

The IMDB dataset consists of movies released between 1980-2006 and related entities such as directors, actors, and studios. Each movie in the dataset has the binary class label *isblockbuster*, which indicates whether the total earnings of the movie was greater than \$30mil (inflation adjusted). We considered the movies released in the U.S. in the years 1982-1988 and again sampled temporally by year. For the experiments, we considered two movies to be linked if they have at least one actor in common.

Figure 2b shows the average performance obtained for the IMDB classification task using the different classification methods discussed. The 2B model is significantly better than all other models ($p < 0.05$) and the Chernoff and Dirichlet models are significantly better than all models except the local model ($p = 0.06$). The shrinkage approaches achieve a 7-11% reduction in error over the local and global models and 17-22% over the other competing approaches.

Note that the RDN and GRF models perform significantly worse than the local and global baselines in both the Cora and IMDB data experiments. For this reason, we do not consider these models further in the synthetic data experiments in the next section.

5.3. Synthetic data experiments

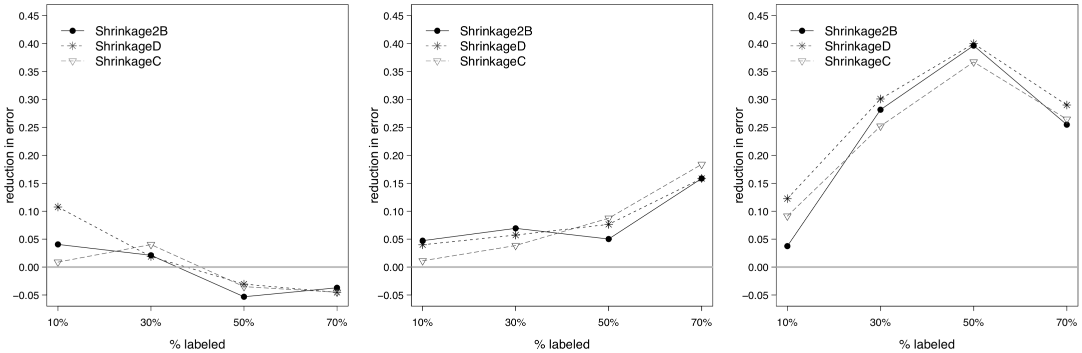
Data for this set of experiments were generated with a latent group model framework [13], where group membership is used to determine the class label of the object. We generated three different datasets for each level of autocorrelation variance and report average performance over five trials. Each dataset consists of 500 objects linked through undirected edges; each object has a binary class label and is a member of a single group. For the experiments, 20% of all nodes were selected randomly to form the test sets and different proportions of the remaining 80% were chosen randomly to form the training sets.

Variance in autocorrelation was explicitly introduced during the data generation process by controlling the class label probabilities for different types of groups. In the *no-variance* datasets, there are two types of groups (equally likely). Objects in groups of the first type have a positive class label with $p(+) = 0.85$; objects in groups of the second type are positive with $p(+) = 0.15$. In the *medium-variance* datasets, there are six types of groups with class distributions: $\mathbf{P}(+) = \{0.95, 0.85, 0.75, 0.25, 0.15, 0.05\}$. Similarly, in the *high-variance* datasets, there are ten group types with class distributions: $\mathbf{P}(+) = \{0.95, 0.80, 0.85, 0.70, 0.75, 0.25, 0.20, 0.15, 0.10, 0.05\}$.

In these experiments the global model always outperformed the local model, so we present the reduction in error for the three shrinkage models compared to the global model. Figure 3 graphs the reduction in error for different percentages of labeled data and different levels of variance in the autocorrelation distribution. As seen in the figure, the reduction of error over the *global* model increases with increasing variance in autocorrelation. While there is almost no difference between the performances of the *global* and *shrinkage* models for the uniform autocorrelation case, a 40% reduction in error is observed in the high variance case. We note that the decrease after 50% labeled data is not due to a decrease in the accuracy achieved with the *shrinkage* models, rather it is due to an increase in the overall accuracy of the *global* model.

6. Conclusions

In this paper, we proposed three classification models accounting for non-stationary autocorrelation in relational



(a) No autocorrelation variance (b) Medium autocorrelation variance (c) High autocorrelation variance

Figure 3. Synthetic data classification results

data. While previous research on collective inference models for relational data has assumed that autocorrelation is stationary throughout the data graph, we have demonstrated that this assumption does not hold in at least two real-world datasets. The models that we propose combine *local* and *global* dependencies between attributes and/or class labels of related instances to make accurate predictions.

We evaluated the performance of the proposed shrinkage models on one synthetic and two real-world datasets and compared with two baseline models considering either *global* or *local* dependencies alone as well as two other competing classification models. The results indicate that our *shrinkage* models, which *back off* from local to global autocorrelation as necessary, allow significant improvements in prediction accuracy over all the other classification models discussed.

We provided empirical evalution on two real-world relational datasets, but the models we propose can be used for classification tasks in any relational domain due to their simplicity and generality. In addition, the shrinkage approach could easily be incorporated into other statistical relational models that use global autocorrelation and collective inference.

Acknowledgments

This research is supported by DARPA, IARPA, and AFRL under contract numbers FA8750-07-2-0158 and NBCH1080005. The U.S. Government is authorized to reproduce and distribute reprints for governmental purposes notwithstanding any copyright notation hereon. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements either expressed or implied, of DARPA, IARPA, AFRL, or the U.S. Government.

References

[1] R. Blattberg and E. George. Shrinkage estimation of price and promotional elasticities: Seemingly unrelated equations. *JASA*, 86(414):304–315, 1991.

[2] S. Chakrabarti, B. Dom, and P. Indyk. Enhanced hypertext categorization using hyperlinks. In *SIGMOD'98*, 1998.

[3] X. Chen. Exact computation of minimum sample size for estimation of binomial parameters. In *arxiv math.ST*, 2007.

[4] N. Friedman, L. Getoor, D. Koller, and A. Pfeffer. Learning probabilistic relational models. In *IJCAI'99*, 1999.

[5] M. Fuentes. Spectral methods for nonstationary spatial processes. *Biometrika*, 89(1):197–210, 2002.

[6] L. Goodman. Snowball sampling. *Annals of Mathematical Statistics*, 32:148–170, 1961.

[7] D. Higdon, J. Swall, and J. Kern. Non-stationary spatial modeling. *Bayesian Statistics*, 6:761–768, 1998.

[8] D. Jensen, J. Neville, and B. Gallagher. Why collective inference improves relational classification. In *Proc. of KDD'04*, 2004.

[9] A. McCallum, K. Nigam, J. Rennie, and K. Seymore. A machine learning approach to building domain-specific search engines. In *IJCAI'99*, 1999.

[10] A. McCallum, R. Rosenfeld, T. Mitchell, and A. Ng. Improving text classification by shrinkage in a hierarchy of classes. In *ICML'98*, 1998.

[11] J. Neville and D. Jensen. Iterative classification in relational data. In *AAAI Workshop on Statistical Relational Learning*, 2000.

[12] J. Neville and D. Jensen. Dependency networks for relational data. In *ICDM'04*, 2004.

[13] J. Neville and D. Jensen. Leveraging relational autocorrelation with latent group models. In *ICDM'05*, 2005.

[14] J. Neville and D. Jensen. Relational dependency networks. *Journal of Machine Learning Research*, 8:653–692, 2007.

[15] C. Perlich and F. Provost. Acora: Distribution-based aggregation for relational learning from identifier attributes. *Machine Learning*, 62(1/2):65–105, 2006.

[16] M. Richardson and P. Domingos. Markov logic networks. *Machine Learning*, 62:107–136, 2006.

[17] L. Sachs. *Applied Statistics*. Springer-Verlag, 1992.

[18] S. Slattery and T. Mitchell. Discovering test set regularities in relational domains. In *ICML'00*, 2000.

[19] B. Taskar, P. Abbeel, and D. Koller. Discriminative probabilistic models for relational data. In *UAI'02*, 2002.

[20] B. Taskar, E. Segal, and D. Koller. Probabilistic classification and clustering in relational data. In *IJCAI'01*, 2001.

[21] X. Zhu, Z. Ghahramani, and J. Lafferty. Semi-supervised learning using gaussian fields and harmonic functions. In *ICML'03*, 2003.