Modelling Relational Statistics With Bayes Nets (Poster Presentation SRL Workshop)

Oliver Schulte, Hassan Khosravi, Arthur Kirkpatrick, Tianxiang Gao, Yuke Zhu
School of Computing Science, Simon Fraser University,Vancouver-Burnaby, Canada

Abstract

Class-level dependencies model general relational statistics over attributes of linked objects and links. Class-level relationships are important in themselves, and they support applications like policy making, strategic planning, and query optimization. An example of a class-level query is "what is the percentage of friendship pairs where both friends are women?". To represent class-level statistics, we utilize Parametrized Bayes nets (PBNs), a 1st-order logic extension of Bayes nets. The standard grounding semantics for PBNs is appropriate for answering queries about specific ground facts but not appropriate for answering queries about classes of individuals. We propose a novel random selection semantics for PBNs, based on Halpern's classic semantics for probabilistic 1st-order logic (Halpern, 1990), that supports class-level queries. For parameter learning we use the empirical frequencies in the relational data. A naive computation of the empirical frequencies of the relations is intractable due to the complexity imposed by negated relations. We render the computation tractable by using the Möbius transform. Evaluation on four benchmark datasets indicates that maximum pseudo-likelihood provides accurate estimates at different sample sizes.

1. Introduction: Class-Level Queries

Many applications store data in relational format, with different tables for entities and their links. Relational data introduces the machine learning problem of *class-level probability estimation*: building a model that can answer generic statistical queries about classes of individuals in the database (Getoor et al., 2001). For example, a class-level query for a social network database may be "what is the percentage of friendship pairs where both are women"? A movie database example would be "what is the percentage of male users who have rated highly an action movie?" A model of database statistics can be used for several applications, such as the following.

Appearing in *Proceedings of the 29th International Conference on Machine Learning*, Edinburgh, Scotland, UK, 2012.

Statistical 1st-order Patterns. AI research into combining 1st-order logic and probability investigated in depth the representation of statistical patterns in relational structures (Halpern, 1990; Bacchus, 1990). Often such patterns can be expressed as *generic statements*, like "intelligent students tend to take difficult courses".

Policy making and strategic planning. A university administrator may wish to know which program characteristics attract high-ranking students in general, rather than predict the rank of a specific student in a specific program. Such predictions are based on generic class-level correlations.

Query optimization. Getoor, Taskar and Koller showed that knowledge of class-level dependencies can be exploited in database query optimization (2001). A statistical model predicts a probability for given table join conditions that can be used to infer the size of the join result. Estimating join sizes (selectivity estimation) is used to minimize the size of intermediate join tables. For a simple example, to find Canadians that live in Vancouver, it is much faster to check the residents of Vancouver to see whether they are Canadian than to check all the Canadians to see whether they live in Vancouver.

Semantics. We focus on building a Bayes net model for relational statistics, using Parametrized Bayes nets (PBNs) (Poole, 2003). The nodes in a PBN are constructed with functors and 1st-order variables (e.g., $Gender(X)$ may be a node). The original PBN semantics is a grounding semantics where the 1st-order Bayes net is instantiated with all possible groundings to obtain a directed graph whose nodes are functors with constants (e.g., $Gender(sam)$). The ground graph can be used to answer queries *about individuals*. However, as pointed out by Getoor (2001), the ground graph is not appropriate for answering class-level queries because these are about generic rates and percentages, not about any particular individuals.

We propose a new semantics for Parametrized Bayes nets that supports class-level queries. The semantics is based on the classic random selection semantics for probabilistic 1st-order logic (Halpern, 1990; Bacchus, 1990). While we focus on PBNs, the random selection semantics can be applied to any statistical-relational model whose syntax is based on 1st-order logic.

Learning. A standard Bayes net parameter learning method is maximum likelihood estimation. A traditional likelihood measure is difficult to define for relational data, because of cyclic data dependencies. We utilize a recent relational pseudo-likelihood measure for Bayes nets (Schulte, 2011) that is well defined even in the presence of cyclic dependencies. In addition to this robustness, the relational pseudo-likelihood matches the random selection semantics because it is also based on the concept of random instantiations. An estimator that chooses the parameters that maximize this pseudo-likelihood function (MPLE), has a closed-form solution: the MPLE parameters are the empirical frequencies, as with classical i.i.d. maximum likelihood estimation. Since MPLE depends only on the generic event frequencies in the data, it can be viewed as an instance of *lifted learning.* Computing the empirical frequencies for negated relationships is difficult, however, because enumerating the complement of a relationship table is computationally infeasible. We show that the Möbius transform (Kennes & Smets, 1990) makes MPLE tractable, even in the case of negated relationships. This transform is a general procedure for computing relational statistics that involve negated links. It has application in Probabilistic Relational Models (Getoor et al., 2007, Sec.5.8.4.2), multi-relational data mining, and inductive logic programming models with clauses containing negated relationships.

Results. We evaluate MPLE on four benchmark real-world datasets. On complete-population samples MPLE achieves near perfect accuracy in parameter estimates, and excellent performance on Bayes net queries. The accuracy of MPLE parameter values is high even on medium-size samples.

Contributions. Our main contributions for frequency modelling in relational data are the following:

1 A new class-level semantics for graphical 1st-order models, derived from the random selection semantics for probabilistic 1st-order logic.

2 Making the computation of frequency estimates tractable by computing database statistics using the fast Möbius transform.

3 Evaluating the empirical accuracy of the Bayes net class-level models at medium to large sample sizes.

4 We contribute to unification of instance-level and class-level relational probabilities (defined in the next section) in two ways. (1) The same 1st-order model can be used for both types of inference. (2) The same objective function is suitable for learning models for both types of queries.

Paper Organization. We review background and notation in the next section. Section 4 presents the random selection semantics for Bayes nets. Section 5 presents the Möbius transform for relational data. Simulation results are presented in Section 6, showing the runtime cost of estimating parameters, and evaluations of their quality by (a) comparison with the true population parameter values, and (b) inference on random queries.

2. Related Work

Class-level and Instance-level Relational Probabilities. Classic AI research established a fundamental distinction between two types of probabilities associated with a relational structure (Halpern, 1990; Bacchus, 1990). *Class-level probabilities*, also called type 1 probabilities are assigned to the rates, statistics, or frequencies of events in a database. These concern classes of entities (e.g., students, courses, users) rather than specific entities. *Instance-level probabilities*, also called type 2 probabilities are assigned to specific, non-repeatable events or the properties of specific entities. Syntactically, class-level probabilities are assigned to formulas that contain 1st-order variables (e.g., $P(\mathit{Flies}(X)|\mathit{Bird}(X)) = \mathit{90}\%$, or "birds fly" with probability 90%), whereas instance-level probabilities are assigned to formulas that contain constants only (e.g., $P(\mathit{Flies}(\mathit{tweety})) = \mathit{90}\%$). There has been much AI research on using Bayes nets for representing and reasoning both with class probabilities (Bacchus, 1990) and instance probabilities (Ngo & Haddawy, 1997). Most statistical-relational learning has been concerned with instance probabilities: For instance, Probabilistic Relational Models (PRMs) (Getoor et al., 2003) and Markov Logic Networks (MLNs) (Domingos & Lowd, 2009) define probabilities for ground instances using a grounding semantics.

Statistical Relational Models. To our knowledge, Statistical Relational Models (SRMs) due to Getoor, Taskar and Koller (2001), are the only prior statistical model with a class-level probability semantics. SRMs differ from PBNs and other statistical-relational models in several respects. (1) The SRM syntax is not that of first-order logic, but is derived from a tuple semantics (Getoor, 2001), which is different from the random selection semantics we propose for PBNs. (2) SRMs are less expressive. One restriction is that they cannot express general combinations of positive and negative relationships (Getoor, 2001). Other restrictions are described by Getoor (2001, Ch.6). A direct empirical comparison between PBNs and SRMs is difficult as SRM code has not been released (Getoor, personal communication).

Computing Sufficient Statistics With Negated Relations For the case of a single relationship, Getoor et al. (2007) introduced a "1-minus trick" that computes the number of tuples that are not related, from the number that are related. The Möbius transform generalizes this to an arbitrary number of relationships.

In their description of Markov Logic Network (MLN) learning, Domingos and Richardson indicate that for smaller domains, they use an efficient recursive algorithm to find the exact number of satisfying groundings of a formula (Domingos & Richardson, 2007, Sec.12.8). We have not found a description of the algorithm. We performed experiments where we compared answering class-level queries with Parametrized Bayes nets to answering them with MLNs using the Alchemy system (details not shown due to lack of space). Alchemy parameter learning did not terminate on any of the MLN structures involving negated relationships, but did terminate after removing formulas with negated relationships (as was done in (Khosravi et al., 2010)). This illustrates the difficulties that negated relationships cause for current SRL systems.

3. Background: Parametrized Bayes Nets

Our work combines concepts from relational databases and graphical models. As much as possible, we use standard notation in these different areas. We assume familiarity with Bayes nets and concepts such as CP-table and I-map. A **family** in a Bayes net comprises a child node and its parents. Parametrized Bayes nets are a basic graphical model for relational data (Poole, 2003). The syntax of PBNs is as follows. A **functor** is a function symbol or a predicate symbol. Each functor has a set of values (constants) called the **range** of the functor. There are two types of functor nodes: Boolean **relationship functors** that indicate whether a relationship holds (e.g., *Friend*), and **attribute functors** that correspond to the value of an attribute (e.g., *Gender*). A **population variable** X is associated with a population, a set of individuals, corresponding to a type, domain, or class in logic. A **functor random variable** or **functor node** is of the form $f(X_1, \ldots, X_k)$. In this paper we assume that functor nodes contain 1st-order variables only (no constants). A **Parametrized Bayes Net** is a Bayes net whose nodes are functor nodes. In the following we often omit the prefix "Parametrized" and speak simply of Bayes nets. Figure 1 shows a PBN. An **instantiation** or **grounding** for a set of variables $X_1, \ldots, X_k$ assigns a constant c_i from the population of X_i to each variable X_i. Figure 1 shows a Parametrized Bayes net and a simple relational database instance.

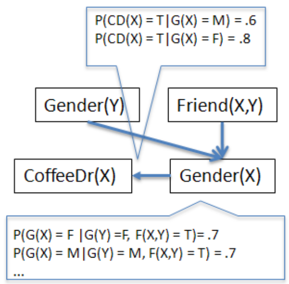
4. Random Selection Semantics for Bayes Nets

Random Selection Semantics for First-Order Logic. For a single population, a distribution over population members induces a joint distribution over their attributes (e.g., age, height, gender). Classic AI research generalized the concept of single population frequencies to 1st-order logic using the idea of a *random selection* (Halpern, 1990; Bacchus, 1990). We provide a brief review in the context of a functor language. For example, consider a probabilistic 1st-order statement using the obvious abbreviations for the functors in Figure 1:

$$P(Fr(X, Y) = T, G(X) = M, G(Y) = F) = 1/4, \tag{1}$$

which assigns probability 1/4 to a sentence with free 1st-order variables. The random selection semantics assumes a distribution over the population/domain associated with each free 1st-order variable. Assuming the independence of these distributions, we obtain a joint distribution over the values of population variables $X_1, X_2, \ldots, X_k$; that is, a joint distribution over tuples of individuals. The class-level probability of a 1st-order statement is then the sum over all tuples that satisfy the statement, weighted by the probability of each tuple.

The database distribution. In learning, an observed database instance $\mathcal{D}$ provides data only for a subpopulation. We define the **database distribution**, denoted by $P_{\mathcal{D}}$, of a functor node assignment to be the number of instantiations of the population variables in the functor nodes that satisfy the assignment



People

Name	Gender	Coffee Drinker
Anna	F	T
Bob	M	F

Friend

Name1	Name2
Anna	Bob
Bob	Anna

Figure 1. Left: An illustrative Parametrized Bayes Net. *Friend*(X, Y) is a relationship node, the other three nodes are attribute nodes. Right: A simple relational database instance.

in the database, divided by the number of all possible instantiations. The database distribution is the special case of the class-level probability where the population distribution of a variable X is the uniform distribution over all observed members of the X-population in the database. For example, the probability statement above is true in the database of Figure 1 given a uniform distribution over users.

Random Selection Semantics for Bayes Nets. The random selection concept provides a class-level semantics for Parametrized Bayes nets: if we view 1st-order variables $X_1, X_2, \ldots, X_k$ as independent random variables that each sample an individual, then a functor of the form $f(X_1, X_2, \ldots, X_k)$ represents a function of a random k-tuple. Since a function of a random variable is itself a random variable, this shows how we can view functor nodes containing 1st-order variables as random variables in their own right, without grounding the variables first.

For example, using the obvious abbreviations for the PBN of Figure 1, the semantics of a joint assignment like

$$P(F(X,Y) = T, G(X) = M, G(Y) = M, CD(X) = T) = 10\%$$

is "if we randomly select two users X and Y, there is a 10% chance that they are friends, both are men, and one is a coffee drinker".

With regard to selectivity estimation, the class-level semantics that we propose for Parametrized Bayes Nets covers all queries that involve projections, selections and joins over the functors in the PBN. This is the main class of queries, called $\pi - \sigma - \times$ queries, considered in standard query optimization systems (Ramakrishnan & Gehrke, 2003, Ch.15.2.1).

Random Selection Pseudo-Likelihood. Schulte (2011) proposed a way to measure the fit of a Bayes net model to relational data that matches the random selection semantics. The pseudo log-likelihood for a database $\mathcal{D}$ given a PBN B is the expected log-likelihood of a random instantiation of the 1st-order variables in the PBN with individuals and values from the database $\mathcal{D}$. For a fixed database $\mathcal{D}$ and Bayes net structure, the parameter values that maximize the pseudo-likelihood are the **MPLE** values. These are the conditional empirical frequencies defined by the database distribution $P_{\mathcal{D}}$ (Schulte, 2011). This result is exactly analogous to maximum likelihood estimation for i.i.d. data. In the remainder of the paper we evaluate MPLE parameter estimates. We begin with a procedure for computing them.

5. Computing Relational Frequencies

Initial work in SRL modelled the distribution of descriptive attributes given knowledge of existing links. Database statistics conditional on the *presence* of one or more relationships can be computed by table joins with SQL. More recent models represent *uncertainty about relationships* with link indicator variables. For instance, a Parametrized Bayes net includes relationship indicator variables such as *Friend*(X, Y). Learning with link uncertainty requires computing sufficient statistics that involve the *absence* of relationships. A naive approach would explicitly construct new data tables that enumerate tuples of objects that are *not* related. However, the number of unrelated tupes is too large to make this scalable (think about the number of user pairs who are *not* friends on Facebook). Can we instead reduce the computation of sufficient statistics that involve negated relationships to the computation of sufficient statistics that involve existing (positive) relationships only? The **inverse Möbius transform** (IMT) provides an affirmative answer (Kennes & Smets, 1990).

The IMT was originally described using category theory with lattice structures. Our version is adapted for **joint probability tables** (JP-tables). A JP-table is just like a CP-table whose rows correspond to joint probabilities rather than conditional probabilities. A sufficient statistic that involves positive relationships only is called a *Möbius parameter*. To represent a Möbius parameter, we allow relationship nodes to take on the value $*$ for "unspecified". For instance, suppose that the family nodes are *Int*(S), *Reg*(S, C), *RA*(S, P). Then the Möbius parameter $P(\mathit{Int}(S) = 1)$ is stored in the row where $\mathit{Int}(S) = 1, \mathit{Registered}(S, C) = *, \mathit{RA}(S, P) = *$. The IMT uses a local update operation corresponding to the simple probabilistic identity

$$P(\sigma, \mathbf{R}, R = F) := P(\sigma, \mathbf{R}) - P(\sigma, \mathbf{R}, R = T)$$

where σ is an attribute condition that does not involve relationships and $\mathbf{R}$ specifies values for a list of relationship nodes. This shows how a probability that involves $k+1$ false relationships can be computed from two probabilities that each involve only k false relationships, for $k \geq 0$. The IMT initializes the JP-table with the observed frequencies that do *not* involve negated relationships, that is, all relationship nodes have the value T or $*$. It then goes through the relationship nodes $R_1, \ldots, R_m$ in order, replaces at stage i all occurrences of $R_i = *$ with $R_i = F$, and applies the local update equation for the probability value for the modified row. At termination, all $*$ values have been replaced by F and the JP-table specifies all joint frequencies. Algorithm 1 gives pseudocode and Figure 2 illustrates the IMT in a schematic example with two relationship nodes.

Complexity Analysis. (1) The primary property of the IMT is that *it accesses data only about existing links*, never about nonexisting links. (2) A secondary but attractive property of IMT is that the number of additions performed is $m2^{m-1}$. A lower bound argument shows that this is optimal (Kennes & Smets, 1990).

Algorithm 1 The inverse Möbius transform for parameter estimation in a Parametrized Bayes Net. The algorithm is applied to each family in the Bayes net.

Input: database $\mathcal{D}$; a set of functor nodes divided into attribute nodes $A_1, \ldots, A_j$ and relationship nodes $R_1, \ldots, R_m$.
Output: Joint Probability specifying the data frequencies for each joint assignment to the input functor nodes.

```
for all attribute value assignments A_1 := a_1, ..., A_j := a_j do
    initialize the JP-table with the Möbius parameters: set all relationship nodes to either T or *; find joint probabilities with data queries.
    for i = 1 to m do
        Change all occurrences of R_i = * to R_i = F.
        Update the joint probabilities using (5).
    end for
end for
```

6. Evaluation

All experiments were done on a QUAD CPU Q6700 with a 2.66GHz CPU and 8GB of RAM. We evaluated the algorithm on real-world datasets that have been used in many studies of multi-relational learning (e.g., (Chen et al., 2009)). The datasets and our code are available on the Web (Khosravi et al.).

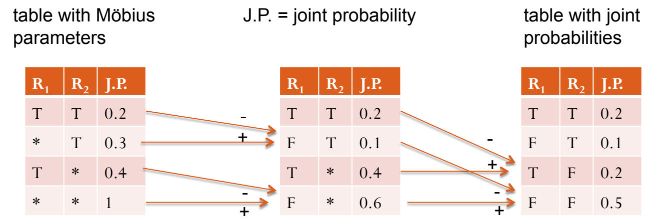


Figure 2. The fast Möbius transform with $m = 2$ relationship nodes. For simplicity we omit attribute conditions.

6.1. Datasets

We used four benchmark real-world databases, with the modifications by (Khosravi et al., 2010), which contains details and references.

Mondial Database. A geography database. Mondial features a self-relationship, *Borders*, that indicates which countries border each other.

Hepatitis Database. A modified version of the PKDD'02 Discovery Challenge database.

Financial A dataset from the PKDD 1999 cup.

MovieLens. A dataset from the UC Irvine machine learning repository.

To obtain a Bayes net structure for each dataset, we applied the learn-and-join algorithm (Khosravi et al., 2010) to each database. We also conducted experiments with synthetic graphs and datasets. The results are similar to those on real-life datasets. We omit details for lack of space.

6.2. Learning Times

Table 1 shows the runtimes for computing parameter values. The Complement method uses MySQL queries that explicitly construct tables for the complement of relationships, while the IMT method uses the inverse Möbius transform to compute the conditional probabilities. Table 1 shows that the IMT is faster by orders of magnitude, ranging from a factor of 15–237. More complex SQL queries are possible, such as using the *Count*($*$) aggregate function or indices; we leave such extensions for future work.

6.3. Conditional Probabilities

To study parameter estimation at different sample sizes, we performed a set of experiments to train the model on N% of the data and test on 100% of the data. Conceptually, we treated each benchmark database as specifying an entire population, and

Figure 3. The estimates of conditional probability parameters, averaged over 10 random subdatabases and all BN parameters. Error (absolute difference) in conditional probability estimates. The median observation is the red center line and the box comprises 75% of the observed values. The whisker indicates the maximum acceptable value (1.5 IQR upper).

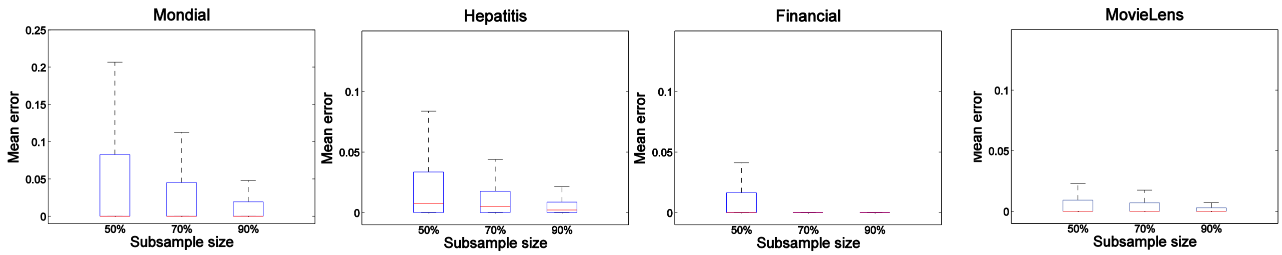


Table 1. Learning time results (sec) for the Möbius transform vs. constructing complement tables. For each database, we show the number of tuples, and of parameters in the fixed Bayes net structure.

Database	Parameters	#tuples	Complement	IMT	Ratio
Mondial	1618	814	157	7	22
Hepatitis	1987	12,447	18,246	77	237
Financial	10926	17,912	228,114	14,821	15
MovieLens	326	82,623	2,070	50	41

then estimated the complete-population frequencies from partial-population data. A fractional sample size parameter is uniform across tables and databases. We employed standard subgraph subsampling (Frank, 1977; Khosravi et al., 2010), which selects entities uniformly at random and restricts the relationship tuples in each subdatabase to those that involve only the selected entities.

Figure 3 illustrates that with increasing sample size, MPLE estimates approach the true value in all cases. Even for the smaller sample sizes, the median error is close to 0, confirming that most estimates are very close to correct. As the box plots show, the 3rd error quartile of estimates is bound within 10% on Mondial, the worst case, and within less than 5% on the other datasets.

6.4. Inference

The basic inference task for Bayes nets is answering probabilistic queries. If the given Bayes net structure is an I-map of the true distribution, then correct parameter values lead to correct predictions. Thus the performance on queries has been used to evaluate parameter learning. We randomly generate queries for each dataset according to the following procedure. First, choose a target node V 100 times, and go through each possible value a of V such that $P(V = a)$ is the probability to be predicted. For each value a, choose the number k of conditioning variables, ranging from 1 to 3. Select k variables $V_1, \ldots, V_k$ and corresponding values $a_1, \ldots, a_k$. The query to be answered is then $P(V = a|V_1 = a_1, \ldots, V_k = a_k)$. An example query could be

$$P(\mathit{Int}(S) = \mathit{high}|\mathit{Registered}(S, C) = T, \mathit{diff}(C) = \mathit{high}).$$

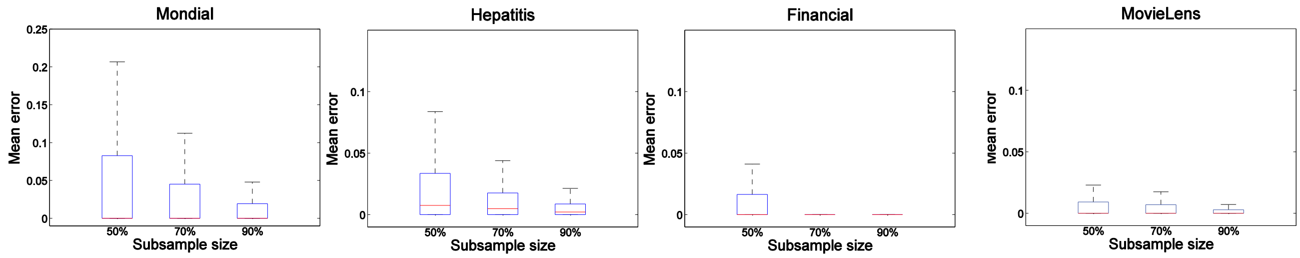
This asks for the number of student-course pairs with a highly intelligent student, out of the class of student-course pairs where the student is registered in the course and the course is difficult.

As in (Getoor et al., 2001), we evaluate queries after learning parameter values on the entire database. Thus the BN is viewed as a statistical summary of the data rather than generalizing from a sample. BN inference is carried out using the Approximate Updater in CMU's Tetrad program. Figure 4 shows that the accuracy of Bayes net query estimates is high. We also compared the runtime cost of performing model inference vs. estimating query sizes using SQL, but cannot show the details due to lack of space. Basically, model inferences are substantially faster, because for larger databases, the cost of data access is much greater than the CPU cost of inference computations; see also (Getoor et al., 2001). For queries that involve negated relations, model inference is faster than density estimation from data by orders of magnitude.

7. Conclusion

We introduced a new semantics for Parametrized Bayes nets as models of class-level statistics in a relational structure. For parameter learning we utilized the empirical database frequencies, which can be feasibly computed using the Möbius transform, even for frequencies concerning negated links. In evaluation on four benchmark databases, the maximum

Figure 4. Query Performance: Absolute difference between estimated vs. true probability. The median observation is the red center line and the box comprises 75% of the observed values. The whisker indicates the maximum acceptable value (1.5 IQR upper). Number of queries/average inference time per query: Mondial, 506/0.08sec; MovieLens, 546/0.05sec; Hepatitis, 489/0.1sec; Financial, 140/0.02sec.



pseudo-likelihood estimates approach the true conditional probabilities as observations increase. The fit is good even for medium data sizes.

An important topic for future work is to extend class-level learning to relational data with missing values and/or Bayes net models with latent variables. In the propositional case, maximum likelihood methods such as EM have been successfully used for such problems; adapting EM for use with the random selection pseudo-likelihood is therefore a promising approach.

Acknowledgements

This research was supported by a Discovery Grant from the National Sciences and Engineering Research Council of Canada. Anonymous reviewers for the SRL Workshop provided helpful comments, especially on the literature regarding the inverse Möbius transform.

References

Bacchus, Fahiem. *Representing and reasoning with probabilistic knowledge: a logical approach to probabilities*. MIT Press, Cambridge, MA, USA, 1990.

Chen, Hailiang, Liu, Hongyan, Han, Jiawei, and Yin, Xiaoxin. Exploring optimization of semantic relationship graph for multi-relational Bayesian classification. *Decision Support Systems*, 48:1:112–121, July 2009.

Domingos, Pedro and Lowd, Daniel. *Markov Logic: An Interface Layer for Artificial Intelligence*. Morgan and Claypool Publishers, 2009.

Domingos, Pedro and Richardson, Matthew. Markov logic: A unifying framework for statistical relational learning. In *Introduction to Statistical Relational Learning* Getoor & Tasker (2007).

Frank, O. Estimation of graph totals. *Scandinavian Journal of Statistics*, 4:2:81–89, 1977.

Getoor, Lisa, Friedman, Nir, Koller, Daphne, and Taskar, Benjamin. Learning probabilistic models of link structure. *J. Mach. Learn. Res.*, 3:679–707, 2003.

Getoor, Lise. *Learning Statistical Models From Relational Data*. PhD thesis, Department of Computer Science, Stanford University, 2001.

Getoor, Lise and Tasker, Ben. *Introduction to statistical relational learning*. MIT Press, 2007.

Getoor, Lise, Taskar, Benjamin, and Koller, Daphne. Selectivity estimation using probabilistic models. *ACM SIGMOD Record*, 30(2):461–472, 2001.

Getoor, Lise, Friedman, Nir, Koller, Daphne, Pfeffer, Avi, and Taskar, Benjamin. Probabilistic relational models. In *Introduction to Statistical Relational Learning* Getoor & Tasker (2007), chapter 5, pp. 129–173.

Halpern, Joseph Y. An analysis of first-order logics of probability. *Artificial Intelligence*, 46(3):311–350, 1990.

Kennes, Robert and Smets, Philippe. Computational aspects of the Möbius transformation. In *UAI*, pp. 401–416, 1990.

Khosravi, H., Man, T., Hu, J., Gao, E., and Schulte, O. Learn and join algorithm code. URL = `http://www.cs.sfu.ca/~oschulte/jbn/`.

Khosravi, Hassan, Schulte, Oliver, Man, Tong, Xu, Xiaoyuan, and Bina, Bahareh. Structure learning for Markov logic networks with many descriptive attributes. In *AAAI*, pp. 487–493, 2010.

Ngo, Liem and Haddawy, Peter. Answering queries from context-sensitive probabilistic knowledge bases. *Theor. Comput. Sci.*, 171(1-2):147–177, 1997.

Poole, David. First-order probabilistic inference. In *IJCAI*, pp. 985–991, 2003.

Ramakrishnan, Raghu and Gehrke, Johannes. *Database Management Systems*. McGraw-Hill, 3rd edition, 2003.

Schulte, Oliver. A tractable pseudo-likelihood function for Bayes nets applied to relational data. In *SIAM SDM*, pp. 462–473, 2011.