Modelling Relational Statistics With Bayes Nets

Oliver Schulte, Hassan Khosravi, Arthur Kirkpatrick, Tong Man, Tianxiang Gao, and Yuke Zhu

School of Computing Science, Simon Fraser University,
Burnaby-Vancouver, 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 random selection semantics for PBNs, based on Halpern's classic semantics for probabilistic 1st-order logic [1], that supports class-level queries. Learning the parameters for this semantics can be done using the recent relational BN pseudo-likelihood measure [2] as the objective function. The parameter settings that maximize this objective function are 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 fast Möbius transform. Evaluation on four benchmark datasets shows that maximum pseudo-likelihood provides accurate estimates at different sample sizes.

1 Introduction

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 frequency estimation*: building a model that can answer generic statistical queries about classes of individuals in the database [3]. 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:

Statistical 1st-order Patterns. AI research into combining 1st-order logic and probability investigated in depth the representation of statistical patterns in relational structures [1, 4]. 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. Maier *et al.* [5] describe several applications of causal-relational knowledge for decision making. These causal relations reflect generic correlations in the database.
- **Query optimization.** A statistical model predicts a probability for given table join conditions that can be used to infer the size of the join result [3]. Estimating join sizes (selectivity estimation) is used to minimize the size of intermediate join tables [6].

Semantics. We focus on building a Bayes net model for relational statistics, using the Parametrized Bayes nets (PBNs) of Poole [7]. 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*, such as "if user *sam* has 3 friends, female *rozita*, males *ali* and *victor*, what is the probability that *sam* is a woman"? However, as pointed out by Getoor [8], 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 Halpern's classic random selection semantics for probabilistic 1st-order logic [1, 4]. Halpern's semantics views statements with 1st-order variables as expressing statistical information about classes (or domains) of individuals. For instance, the claim "the percentage of friendship pairs where both are women is 60%" could be expressed by the 1st-order formula

$$P(Gender(X) = female, Gender(Y) = female | Friend(X, Y)) = 60\%.$$

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, but this method is difficult to apply for Bayes nets that represent relational data because the cyclic data dependencies in relations violate the requirements of a traditional likelihood measure. We circumvent the limitations of classical likelihood measures by using a relational pseudo-likelihood measure for Bayes nets [2] 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 fast Möbius transform [9] makes MPLE tractable, even in the case of 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. This transform is a general procedure for computing relational statistics that involve negated links. It has application in Probabilistic Relational Models [10, Sec.5.8.4.2], multi-relational data mining, and inductive logic programming models with clauses containing negated relationships.
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) We show how the same 1st-order model can be used for both types of inference. (2) We show that 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 fast 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 [1, 4]. *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 [11] and instance probabilities [12]. Most statistical-relational learning has been concerned with instance probabilities: For instance, Probabilistic Relational Models (PRMs) [10] and Markov Logic Networks (MLNs) [13] define probabilities for ground instances using a grounding semantics.

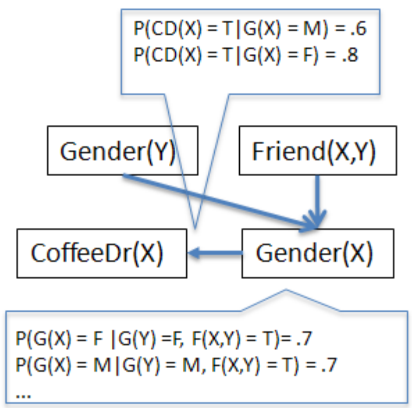
Statistical Relational Models. To our knowledge, Statistical Relational Models (SRMs) due to Getoor, Taskar and Koller [8], are the only prior statistical models with a class-level probability semantics. SRMs differ from PBNs and other statistical-relational models in several respects. (1) The SRM syntax is not that of first-order logic, but is derived from a tuple semantics [8, Def.6.3], which is different from the random selection semantics we propose for PBNs. (2) SRMs are less expressive: They cannot express general combinations of positive and negative relationships [8, Def.6.11]. (3) The published learning algorithm for SRMs uses information gain as a model selection score, not a pseudo-likelihood [3].

Unified Learning for Type 1 and Type 2 Probabilities. Previous SRL work used different models for the two basic types of probability query (SRMs for class-level, template models for instance-level). In this paper we use PBNs and the pseudo likelihood to learn models that are accurate for class-level probabilities. Previous work used the same model class and objective function for learning models that are accurate for instance-level probabilities [14, 2]. We believe that a unified approach to learning for both relational probability types is an exciting research direction for statistical-relational learning.

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. Parametrized Bayes nets are a basic graphical model for relational data [7]. The syntax of PBNs is as follows. We assume familiarity with Bayes nets and concepts such as CP-table and I-map [15]. 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*). To conform to statistical terminology, Poole refers to 1st-order variables as population variables. 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. The syntax of PBNs is similar to that of other directed relational graphical models (cf. [7]). 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.

The functor formalism is rich enough to represent an entity-relationship schema via the following translation: Entity sets correspond to populations, descriptive attributes to functors, relationship tables to Boolean functors, and foreign key constraints to type constraints on the arguments of relationship predicates. Figure 1 shows a Parametrized Bayes net and a simple relational database instance.



People

Name	Gender	Coffee Drinker
Anna	F	T
Bob	M	F

Friend

Name1	Name2
Anna	Bob
Bob	Anna

Fig. 1. Left: An illustrative Parametrized Bayes Net. $\mathit{Friend}(X, Y)$ is a relationship node, the other three nodes are attribute nodes. Right: A simple relational database instance.

4 Random Selection Semantics for Bayes Nets

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* [1, 4]. 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 Bayes net of Figure 1:

$$P(\mathit{Friend}(X, Y) = T, \mathit{Gender}(X) = M, \mathit{Gender}(Y) = F) = 1/4. \quad (1)$$

which assigns probability 1/4 to a sentence with free 1st-order variables. [1] To evaluate whether the statement (1) is true in a given interpretation $\mathcal{D}$, 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 type 1 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 statement is true in an interpretation if it assigns the type 1 probability correct for the interpretation.

In learning, an observed database instance $\mathcal{D}$ provides data only for a subpopulation. We define the **observed database frequency**, 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 in the database, divided by the number of all possible instantiations. The database frequency is the special case of the type 1 probability with a uniform distribution over all observed population members in the database. For example, the probability statement (1) is true in the database of Figure 1 given a uniform distribution over users.

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 BN 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".

Random Selection Pseudo-Likelihood. Schulte [2, 16] 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 values 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 $P_{\mathcal{D}}$ [2, Prop.3.1]. 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.

[1] The full syntax distinguishes between free variables and variables with a probabilistic interpretation.

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 $\mathit{Friend}(X, Y)$. 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 classic Möbius parametrization for binary variables provides an affirmative answer [17, p.239]. Consider a set $b_1, \ldots, b_m$ of binary variables, where all marginal probabilities are available that involve only positive values. Thus we have available probabilities such as $P(b_1 = 1)$; $P(b_1 = 1, b_2 = 1)$; $P(b_1 = 1, b_3 = 1, b_k = 1)$; etc. These joint probabilities are the **Möbius parameters** of the joint distribution. The Möbius inversion theorem entails that *all* joint probabilities, involving any number of 0 values, can be computed as an alternating sum of the Möbius parameters. We can apply this result for MPLE as follows. Consider a PBN family containing m relationship nodes. We wish to compute frequencies of the joint family assignments, from which conditional probabilities are easily derived. The Möbius inversion theorem entails that each joint frequency *can be computed from joint frequencies that involve existing relationships only.*

The **fast Möbius transform** (FMT) is an optimal algorithm for converting the Möbius parameters to a complete set of joint probabilities [9]. The FMT 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. To represent a Möbius parameter, we allow relationship nodes to take on the value $*$ for "unspecified". For instance, suppose that the family nodes are $\mathit{Int}(S)$, $\mathit{Reg}(S, C)$, $\mathit{RA}(S, P)$. Then the Möbius parameter $P(\mathit{Int}(S) = \mathit{1})$ is stored in the row where $\mathit{Int}(S) = \mathit{1}, \mathit{Registered}(S, C) = *, \mathit{RA}(S, P) = *$. The FMT uses a local update operation corresponding to the simple probabilistic identity

$$P(\sigma, \mathbf{R}, R = F) := P(\sigma, \mathbf{R}) - P(\sigma, \mathbf{R}, R = T) \tag{2}$$

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 FMT initializes the JP-table with the Möbius parameters without negated relationships, that is, all

Algorithm 1 The fast Möbuis transform for parameter estimation in a Parametrized 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 $A_1 := a_1, \ldots, A_j := a_j$ **do**
  initialize the JP-table with the Möbius parameters: set all relationship nodes to either T or $*$; find joint frequencies with data queries.
  for $i = 1$ to m **do**
    Change all occurrences of $R_i = *$ to $R_i = F$.
    Update the joint frequencies using (2).
  end for
end for

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 FMT in a schematic example with two relationship nodes.

Complexity Analysis. Kennes and Smets [9] provide a thorough theoretical analysis of the FMT. We summarize the main points. (1) The primary property of the FMT is that *it accesses data only about existing links*, never about nonexisting links. A secondary but attractive property of FMT is that the number of additions performed is $m2^{m-1}$. A lower bound argument shows that this is optimal [9, Cor.1]. (2) Kennes and Smets describe an "obvious algorithm" that applies the local update to each row in the JP-table. The obvious algorithm also uses only existing links, but requires $O(3^m)$ additions. [2] (3) Without a bound on m, computing sufficient statistics in a relational structure is #P-complete [13, Prop.12.4]. In practice, the number m of relationship nodes is small, 4 or less.

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. The datasets and our code are available on the Web [18].

[2] The obvious algorithm, but not the FMT, was rediscovered by Khosravi *et al.* and presented as a conference poster at ILP 2009. This work was not included in the proceedings or any other archival publication. For the case of a single relationship, Getoor et al. [10] introduced a "1-minus trick"; the FMT generalizes this to the multi-relational case.

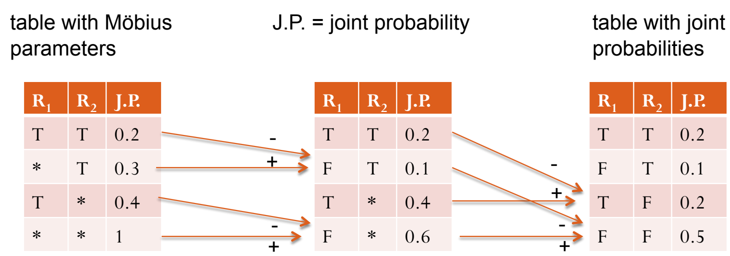


Fig. 2. The fast Möbius transform with $m = 2$ relationship nodes. For simplicity we omit attribute conditions.

6.1 Datasets

We used four benchmark real-world databases, with the modifications by [14], 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 [14] to each database. This is the state-of-the-art structure learning algorithm for PBNs; for an objective function, it uses the pseudo-likelihood described in Section 4. 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

Figure 1 shows the runtimes for computing parameter values. The Complement method uses SQL queries that explicitly construct tables for the complement of relationships (tables that contain tuples of unrelated entities), while the FMT method uses the fast Möbius transform to compute the conditional probabilities. The FMT is faster by orders of magnitude, ranging from a factor of 5–237.

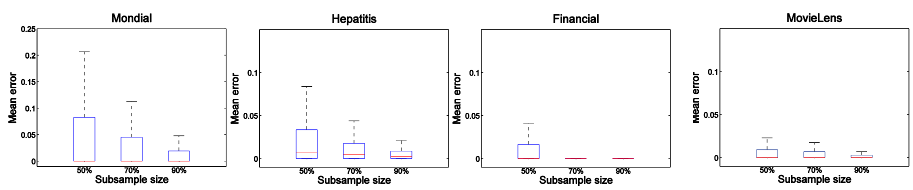
6.3 Conditional Probabilities

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 the other $(100 - N)\%$ of the data. Conceptually, we treated each benchmark database as specifying an entire population, and 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 [19, 14], which selects entities from each entity table uniformly at random

Table 1. Learning time results (sec) for the fast 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	FMT	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

Fig. 3. Error (absolute difference) in conditional probability estimates. Median (red center line) and spread of error in the estimates of conditional probability parameters, averaged over 10 random subdatabases and all parameters in a given BN.



and restricts the relationship tuples in each subdatabase to those that involve only the selected entities. Subgraph sampling matches the random selection semantics which is based on random draws from a population. It is applicable when the observations include positive and negative link information (e.g., not listing two countries as neighbors implies that they are not neighbors). The subgraph method satisfies an ergodic law of large numbers in the sense that as the subsample size increases, the subsample relational frequencies approach the population relational frequencies.

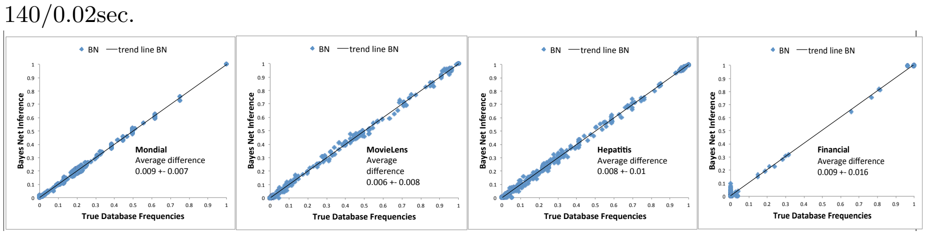
With increasing sample size, MPLE estimates approach the true value in all cases. Even for the smaller sample sizes, the median error is close to 0, confirming that most estimates are very close to correct. As the box plots show, the 3rd error quartile of estimates is bound within 10% on Mondial, the 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 [20]. 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 $P(V = a | V_1 = a_1, \ldots, V_k = a_k)$.

As in [3], 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 the query performance for each database. A point (x, y) on a curve indicates that there is a query such that the true probability value in the database is x and the probability value estimated by the model is y. The Bayes net inference is close to the ideal identity line, with an average error of less than 1%.

Fig. 4. Query Performance: Estimated vs. true probability. The average error and standard deviation are shown as well. Number of queries/average inference time per query: Mondial, 506/0.08sec; MovieLens, 546/0.05sec; Hepatitis, 489/0.1sec; Financial, 140/0.02sec.



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 fast Möbius transform, even for frequencies concerning negated links. In evaluation on four benchmark databases, the maximum pseudo-likelihood estimates approach the true conditional probabilities as observations increase. The fit is good even for medium data sizes.

A direction for future work is to adapt more techniques from i.i.d. Bayes net parameter learning, such as smoothing frequencies and incorporating uncertainty in parameter estimates [20]. A theoretical understanding of estimator variance would be desirable: we may adapt the asymptotic approximations of [20], or apply graph estimator theory [19]. Halpern [1] showed that any instance-level inference model can be used for class-level inference by using ground queries that contains new constants only (e.g., random-student, random-course, and random-prof). We plan to use this scheme to evaluate instance-level models, such as Markov Logic Networks, for class-level queries.

References

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

[2] Schulte, O.: A tractable pseudo-likelihood function for Bayes nets applied to relational data. In: SIAM SDM. (2011) 462–473

[3] Getoor, L., Taskar, B., Koller, D.: Selectivity estimation using probabilistic models. ACM SIGMOD Record **30**(2) (2001) 461–472

[4] Bacchus, F.: Representing and reasoning with probabilistic knowledge: a logical approach to probabilities. MIT Press, Cambridge, MA, USA (1990)

[5] Maier, M., Taylor, B., Oktay, H., Jensen, D.: Learning causal models of relational domains. In: AAAI. (2010)

[6] McMahan, B.J., Pan, G., Porter, P., Vardi, M.Y.: Projection pushing revisited. In: EDBT. (2004) 441–458

[7] Poole, D.: First-order probabilistic inference. In: IJCAI. (2003) 985–991

[8] Getoor, L.: Learning Statistical Models From Relational Data. PhD thesis, Department of Computer Science, Stanford University (2001)

[9] Kennes, R., Smets, P.: Computational aspects of the Möbius transformation. In: UAI. (1990) 401–416

[10] Getoor, L., Friedman, N., Koller, D., Pfeffer, A., Taskar, B.: Probabilistic relational models. [21] chapter 5 129–173

[11] Bacchus, F.: Using first-order probability logic for the construction of Bayesian networks. In: UAI. (1993) 219–226

[12] Ngo, L., Haddawy, P.: Answering queries from context-sensitive probabilistic knowledge bases. Theor. Comput. Sci. **171**(1-2) (1997) 147–177

[13] Domingos, P., Richardson, M.: Markov logic: A unifying framework for statistical relational learning. [21]

[14] Khosravi, H., Schulte, O., Man, T., Xu, X., Bina, B.: Structure learning for Markov logic networks with many descriptive attributes. In: AAAI. (2010) 487–493

[15] Pearl, J.: Probabilistic Reasoning in Intelligent Systems. Morgan Kaufmann (1988)

[16] Schulte, O., Khosravi, H., Man, T.: Learning directed relational models with recursive dependencies. In: ILP, 7207 (2011) 39–44

[17] Lauritzen, S.L.: Graphical Models (Oxford Statistical Science Series). Oxford University Press, USA (July 1996)

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

[19] Frank, O.: Estimation of graph totals. Scandinavian Journal of Statistics **4:2** (1977) 81–89

[20] Allen, T.V., Singh, A., Greiner, R., Hooper, P.: Quantifying the uncertainty of a belief net response: Bayesian error-bars for belief net inference. Artif. Intell. **172**(4-5) (2008) 483–513

[21] Getoor, L., Tasker, B.: Introduction to statistical relational learning. MIT Press (2007)