Model-Driven Data Analysis

C. R. Ramakrishnan  I. V. Ramakrishnan  David S. Warren

I/UCRC Center for Dynamic Data Analytics
Stony Brook University
**Project Definition**

- Analysts construct mental models from data
  - Models support understanding, prediction
  - Inferences can be *explained* in terms of models
- **Probabilistic Tabled Logic Programming (PTLP):** a framework to combine logical and statistical models
- This project: *algorithms and system for approximate inference in PTLP*
Related Work

- Language support for statistical models (e.g. GMTK)
- Combination of Probabilistic and Relational Knowledge (e.g. PRMs, Plate Models)
- Combination of First-Order Logic and Markov Random Fields (e.g. Markov Logic Networks)
- XSB: a deductive system based on logic programming (Warren et al, since ’93).
Models of Relationships

Smoking behavior and friendships

- *A smoker’s friend is likely a smoker*

\[ \forall X, Y. \text{smoker}(X), \text{friend}(X, Y) \Rightarrow \text{smoker}(Y) \]

- *People having common friends become friends*

\[ \forall X, Y, Z. \text{friend}(X, Z), \text{friend}(Y, Z) \Rightarrow \text{friend}(X, Y) \]

- Note these formulas are “soft constraints”: they may not always hold
- Some formulas may express “hard constraints” which always hold:

\[ \forall X, Y. \text{friend}(X, Y) \Rightarrow \text{friend}(Y, X) \]
Probabilistic Tabled Logic Programming

- Based on logic programming, that can perform deduction over general logic programs (LPs with negation in rules).
- Hard constraints are written as traditional rules (clauses) in a logic program.
- Soft constraints are expressed by combining rules with random variables:

\[
\text{smoker}(X), \text{friend}(X, Y), \text{rv}(\text{influence}(X, Y), \text{true}) \Rightarrow \text{smoker}(Y)
\]

Distribution of Boolean random variable “influence” governs the likelihood of this rule being true.

- Queries generate or check conclusions that are *entailed* by the program.
- *Deduction in PTLP builds a proof, which explains how a conclusion was reached.*
Inference and Learning in PTLP

- Inference is done based on **tabled resolution**: 
  
  *traditional resolution + memoization*

- When a random variable is evaluated, the proof may branch
  
  - and the probability of branches are based on the random variable’s distribution.

- Each proof thus has a probability; the probability of a conclusion is the probability that **at least one of its proofs hold**.

- Parameter learning is done using an **EM algorithm**.
Models in PTLP

- **Statistical Models:**
  - Bayesian Networks
  - Hidden Markov Models

- **Logical Models:**
  - Object-oriented logics
  - Well-founded models (3-valued logic)
  - Stable models (Answer Set Programming)

- **Hybrid Models**
Recent Advances in PTLP

- Algorithm for *exact inference* over models with discrete as well as continuous random variables.
  - Enables the use of directed graphical models with continuous variables (e.g. Kalman Filters)
- Generalized *exact inference* algorithm to handle cases where
  - Different proofs may not be mutually exclusive
  - Random variables used within a proof may not be independent
  - There are infinitely many proofs
  - Enables the use of undirected graphical models (Markov Random Fields) and stochastic (temporal) models
- Use of the generalized algorithm in the formal verification of probabilistic systems (i.e. analyzing their temporal behavior).
The Need: Exact inference does not scale well to realistic models with complex dependencies between thousands of random variables.

The Approach: Adapt MCMC or other sampling based algorithms for approximate inference.

Challenge: The logical (hard) constraints in PTLPs mean that the set of consistent samples (i.e. those that satisfy the hard constraints) may be a very small subset of the entire sample space.

Proposed Solution:
- Constrained sampling: restricted to the space of consistent samples.
- Techniques for quick determination of consistency of a sample.
  [Adapted from incremental evaluation techniques developed earlier for non-probabilistic logic programs]
Impact of the Proposed Project

**Outcome:**
- A system providing cognitive support for data analysis
- Efficient support for model construction and model refinement

**Possible Applications:**
- Add-on to CRM tools to support advanced data analysis
- Analysis of vulnerabilities and risks in large networks
Deliverables

- **First 6 months:**
  - PTLP prototype with approximate inference
    - Support for discrete random variables
    - Preliminary scalability studies

- **Second 6 months:**
  - Full-featured PTLP system
    - Support for continuous random variables
    - System evaluation

- **Knowledge transfer:**
  - System prototype
  - Technical reports

- **Longer term plans:**
  - Support for complex statistical models
  - Logical abduction (analysis of what-if scenarios)