Second Order Pseudolikelihood Learning in Relational Domain

Krishna Kumar Tiwari
Under the guidance of Dr.V. Vijaya Saradhi

Department of Computer Science
Indian Institute Of Technology Guwahati

k.tiwari@iitg.ernet.in

Dec 14, 2019

Traditional Vs. Relational Learning

Traditional Domain

- Instances follows *i.i.d* assumption.
- Homogeneous objects.

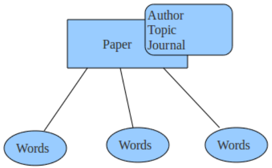
Statistical Relational Learning

- Violates *i.i.d* assumption.
- Heterogeneous objects and links.

Real World Data

- Structured, Semi structured, Unstructured.
- Heterogeneous objects and links.

Citation Database



Examples

citation database (HEP), Movie database (IMDB), Hypertext classification database (ProxWebKB)

Relational Learning Opportunities

- **Object Classification** – Example, in citation database, predicting the topic of a paper.
- **Object Type Prediction**– Predict whether a set of pages belong to conference paper, master's thesis or technical report.
- **Link Type Prediction**– Predicting a paper is published in journal, conference or workshop.
- **Predicting Link Existence**– Given two papers are related or not.
- **Link Cardinality Estimation**– Predicting the number of citations of a paper.
- **Group Detection**– In movie database, identifying groups of same type of movies.

Challenges

How to model relational data ?

- Represent as graphs.
- Represent in multiple tables.
- Represent in terms of fist order logic statements.

Individual vs. Collective classification

- Individual classifiers assumes class label of related entities are independent.
- Collection classification uses related entities class label along with other attribute.

Logical vs. Statistical dependencies

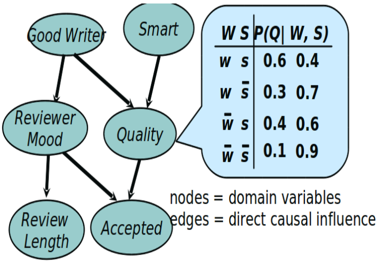
- Logical : $A \rightarrow B$
- Statistical : $Correlation(A, B) = 0.6$

Graphical Models

Directed Models

- Does not allow cyclic dependencies among the attributes.
- Simple parameter estimation and structure learning technique.
- Relational Bayesian Networks.

Bayesian Networks

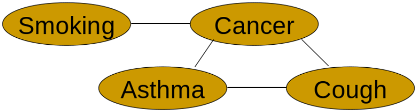


Graphical Models

Undirected Models

- Represent cyclic dependencies.
- Requires known network structure.
- Parameter estimation requires repeated inference over large values.
- Relational Markov Networks.

Markov Networks



Relational Dependency Networks

- Approximate model.
- First model to learn autocorrelation.
- Simple structure learning and parameter estimation.

Approximation Methods

Assume data contains n samples of m dimensional vectors following *i.i.d*, sampled from a distribution p_{θ_0} with $\theta_0 \in \Theta \subset R^r$,

$$D = (X^1, ...X^n), \quad X^i \in R^m,$$

Maximum likelihood estimator

MLE is defined as:

$$l_n(\theta; D) = \sum_{i=1}^{n} p_\theta(X^i) \qquad (1)$$

$$\widehat{\theta}_n^{ml} = \arg\max_{\theta_0 \in \theta} l_n(\theta; D)$$

Pseudolikelihood

PL is defined as:

$$pl_n(\theta; D) = \sum_{i=1}^{n} \sum_{j=1}^{m} log p_\theta(X_j^i | X_{-j}^i). \qquad (2)$$

Where subscript represents dimension and $X_{-j}^i = \{X_k^i \; : k \neq j\}$.

Approximation Methods

Composite Likelihood

- Composite likelihood is a generalization of pseudolikelihood function.
- Composite likelihood function is defined as :

$$cl_n = \sum_{i=1}^{n} \sum_{j=1}^{k} logp_\theta(X^i_{A_j} | X^i_{B_j}). \tag{3}$$

$A \neq \emptyset = A \cap B$, where A and B represents the dimension set of the instance X.

- If $|A_i| = 1$ then composite likelihood represents pseudolikelihood.
- Dillon extended composite likelihood by introducing component weight and selection probabilities to make it stochastic composite likelihood.
- Variance of the model is minimum in case of full likelihood and maximum in pseudolikelihood.

Relational Dependency Networks

Data Graph $G_D = (V_D, E_D)$. V_D represents objects in the data graph and E_D represents relationship between these objects.

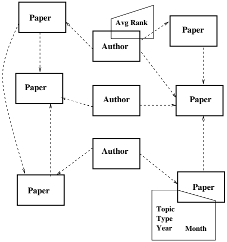


Figure: Data Graph

RDN represents a joint distribution over the values of the attributes in the data graph,

$$x = \left\{ \{ X_{v_i}^{t_{v_i}} : v_i \in V \ s.t. \ T(v_i) = t_{v_i} \} \cup \{ X_{e_j}^{t_{e_j}} : e_j \in E \ s.t. \ T(e_j) = t_{e_j} \} \right\}$$

Approximation of $p(x)$ is done by pseudolikelihood to learn the parameters.

$$PL(G_D;\theta) = \prod_{t\in T} \prod_{X_i^t\in X^t} \prod_{v:T(v)=t} p(x_{v_i}^t | pa_{x_{v_i}^t};\theta) \prod_{e:T(e)=t} p(x_{e_i}^t | pa_{x_{e_i}^t};\theta) \quad (4)$$

Model Graph $G_M = (V_M, E_M)$ represents probabilistic relationship between the attributes.

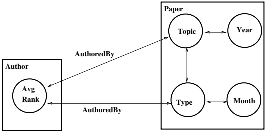


Figure: Model Graph

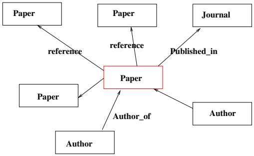
RDN uses relational learners to learn CPDs.

- Relational Bayesian Classifier (RBC)
- Relational Probability Tree (RPT)

Relational Bayesian Classifier

- Treats heterogeneous subgraphs as a homogeneous set of attributes multisets.
- Deals with multisets of different sizes. Ex. In case of citation database, considering publication dates of cited papers form multisets of varying size (e.g. {2002,2002,2006,2009}, {2004,2007,2009}).

Relational data representation



Homogeneous attribute representation

published	cited paper year	Author name	cited papers journal name	Number of pages
Yes	1988,2002,2004	Jennifer, David Jenson	JMLR, KDD	10
No	2001,2001,2000	D Koller, B Taskor	JMLR, JSTOR	6
Yes	1986,2000	Taskar, Abbele, Dephne	SIGCOM, ICDM, ILP, JMLR	11
No	1995,2002,2004, 2006,2009	Scuse, David	JAM, JSTOR	9
Yes	2009,2010	Stephne, koller	JMLR, JSTOR	12

- Independent assumption among the values of set performs best.

Second Order PL in RDN

Motivation

- Choice of different order of likelihood object gives us different ways to approximate joint distribution $p(x)$.
- In highly correlated environment it is better to deal with appropriate combination of attribute.

We define composite likelihood in RDN as :

$$cl(G_D;\theta) = \prod_{t \in T} \prod_{X^t_{A_i} \in X^t}^{k} \prod_{v:T(v)=t} p(x^t_{v_{A_i}} | pa_{x^t_{v_{B_i}}}) \prod_{e:T(e)=t} p(x^t_{e_{A_i}} | pa_{x^t_{e_{B_i}}}) \quad (5)$$

Subject to the constraint:

$$A_i \neq \emptyset = A_i \cap B_i$$

Where A_i and B_i represents set of dimensions.

Second Order PL in RDN

We are denoting second order pseudolikelihood as $pl_2(G_D;\theta)$.

$$pl_2(G_D;\theta) = \sum_{t\in T} \sum_{X^t_{\{p,q\}}\in X^t}^{k} \sum_{v:T(v)=t} p(x^t_{v_{A_i}}|pa_{x^t_{v_{B_i}}}) \sum_{e:T(e)=t} p(x^t_{e_{A_i}}|pa_{x^t_{e_{B_i}}}) \quad (6)$$

Comparison with PL

- Generalization of PL in the context of RDN.
- Second order PL deals with pair of attributes.

Second Order Relational Learners?

Second Order RBC

- Initially we have set of attributes denoted as $A = \{X_1, X_2, \cdots, X_m\}$.
- Second order RBC makes all possible pair of attributes and denoted as P.

$$P = \{\{X_1, X_2\}, \{X_1, X_3\}, \cdots, \{X_{m-1}, X_m\}\}$$

- Second order RBC select elements from P which lead to the full likelihood denoted as set S.

Definition of set S

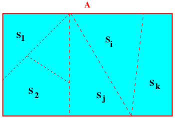
$$S = \{s_i \ : s_i \in P\} \tag{7}$$

Subject to the constraints:

$$\forall \ s_i, s_j \ \in \ S \quad s_i \cap s_j \ = \emptyset \tag{7.a}$$

$$\cup_{i=1}^{|S|} \ s_i = A \tag{7.b}$$

Construction of set S



Exhaustive Search

Choose the subset from P which maximize the likelihood of the class.

$$S \equiv \underset{p \subsetneq P}{\arg\max} \; P(C|p) \tag{8}$$

Greedy Approach

- Assign score to all elements of set P.

$$score(p_i) = logP(C|p_i \in P) \equiv logP(C|\{X_i, X_j\}) \tag{9}$$

- Add maximum score elements of P to S by maintaining the constraints of equation (7).

Second Order PL in RBC

Second Order PL in RBC

According to modified second order PL,

$$P(C|\{a_1, a_2, ...a_m\}) \propto P(A|C) * P(C)$$
$$\equiv P(S|C) * P(C) = \prod_{i=1}^{|S|} P(s_i|C) * P(C) \tag{10}$$

Complexity Analysis

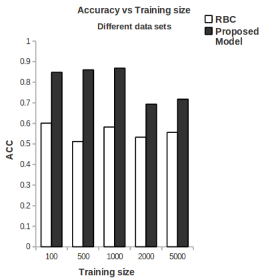
Second order RBC learning has three major components.

- Assignment of score to all elements of set P, it takes $O(|P| \times N)$, where N is number of subgraphs.
- Sorting of the scores, it takes $O(|P| \times log(|P|)$.
- Construction of set S takes $O(|P|^2)$

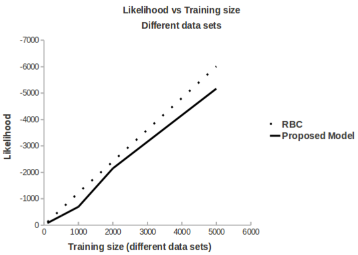
Overall *asymptotic* complexity of second order RBC is $O(|P| \times N)$.

Synthetic Data Results

Accuracy Comparison

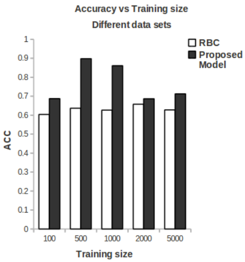


Likelihood Comparison

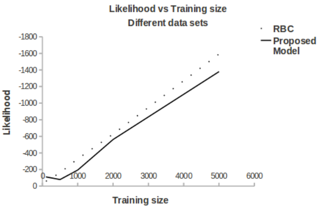


Synthetic Data Results

Accuracy Comparison



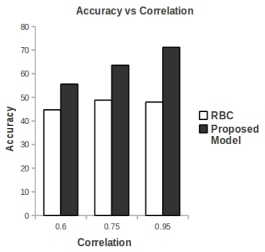
Likelihood Comparison



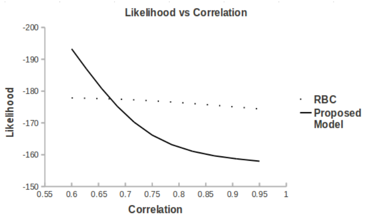
Synthetic Data Results

- **Effect of Correlation**

Accuracy vs. Correlation



Likelihood vs. Correlation

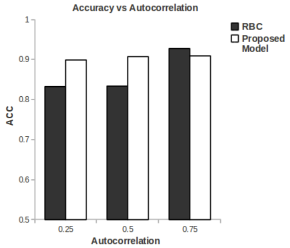


Our model performs better than existing RBC in highly correlated environment in both likelihood estimation as well as accuracy.

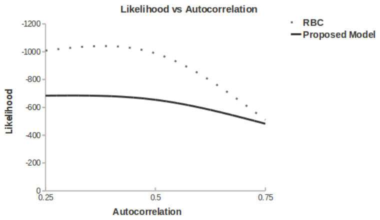
Synthetic Data Results

- **Effect of Autocorrelation**

Accuracy vs. Autocorrelation



Likelihood vs. Autocorrelation

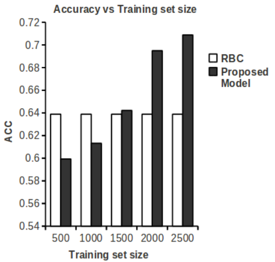


- Performs better in low to moderate autocorrelation and comparable in high autocorrelation environment.
- In high autocorrelation scenarios prediction is biased.

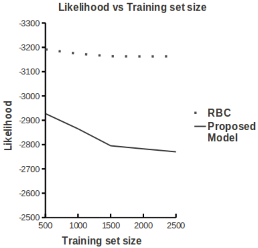
Synthetic Data Results

- **Effect of Training Size**

Accuracy vs. Training Size



Likelihood vs. Training Size

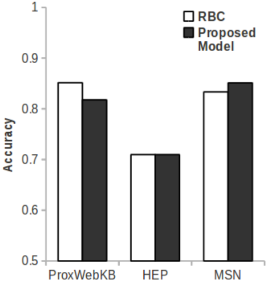


- Second order RBC takes more training data to learn the high correlation present in the data.
- Performs better than existing RBC in highly correlated environment in both likelihood estimation as well as accuracy.

Real World Data Results

Experiments on three real world data sets

- **HEP** Predict the topic of paper given paper attributes, author names and publisher.
- **MSN** Predict time stamp of mote to mote connections.
- **ProxWebKB** Predict the category of a web page given it's linked page categories.

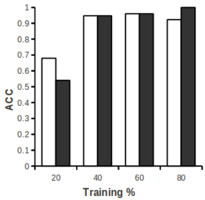


HEP Data Set Result

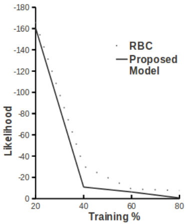
Task

We want to predict the *"acceptability"* of a paper which is formed using paper citation degree and journal name.

Accuracy vs. Training Size



Likelihood vs. Training Size



Conclusion and Future Work

Conclusion

- We demonstrated the use of second order pseudolikelihood in the RDN learning.
- Extended RBC to second order setting.
- Shown improvements in highly correlated data sets both in parameter estimation and classification accuracy.

Future Work

- Making score function which will observe characteristic of test subgraphs.
- Selective models for Second order PL.
- Bias-variance analysis of the model.

DMAP 2019!!

Thank You!