Modeling Temporal Biomedical Data by SRL

Sriraam Natarajan*, Irene Ong+, David Haight#, David Page*+, Vitor Santos Costa$

* Department of CS, UW Madison, + Department of BioStatistics, UW Madison
# Department of Industrial and Systems Engineering, UW Madison $ Universidade Federal do Rio de Janeiro

Abstract. Many biomedical applications involve temporal data, for example time-series gene expression experiments or longitudinal clinical data. From the statistical modeling side of SRL, dynamic Bayesian networks are a natural fit for such data, but they normally cannot incorporate additional types of relational information, such as the interaction between genes. In this paper, we examine the construction of logical DBNs from rules, either learned by relational learning methods or human-provided; in some cases, this construction requires *combining rules*. In some cases using such DBNs requires improved inference algorithms; we propose efficient inference within such DBNs by a Rao-Blackwellized particle filter. This work is motivated by two very different biomedical applications. One is incorporating the rich available background knowledge about genes and proteins when modeling time-series gene expression data. The other is modeling longitudinal nursing home data to estimate health status and health trajectory of individual patients and of groups of patients at specific facilities. This paper focuses on general principles rather than specific representations; we believe the lessons here are applicable for modeling time-series data within SRL.

1 Introduction

An important broad class of applications for SRL is time-series analysis. In this work-in-progress paper, we examine two important biomedical applications within this class.

First, we examine the task of modeling time-series gene expression data in the presence of additional background knowledge. Much work has been invested into learning biological regulatory networks from gene expression data collected by microarray technology. A major challenge is determining the direction of causality in such networks. If Gene A is a good predictor of Gene B, this does not mean Gene A controls Gene B. Rather, Gene A may be a good predictor of Gene B because Gene B controls Gene A, or because both are controlled by a third gene, or even because of more complicated scenarios. Time-series data can give more insight into causality. If a change in Gene A is a good predictor of a change soon thereafter in Gene B, then we can be more confident (though still not certain) that the change in Gene A is helping to cause the change in Gene B, and not vice-versa. If we furthermore have additional background knowledge indicating that Gene A and Gene B are known to interact, we can be even more confident at least of a connection between Gene A and Gene B. Such additional background knowledge is especially important given the limited data we typically have from gene expression experiments, particularly time-series experiments. With this motivation, we examine SRL for modeling time-series gene expression data with respect to relevant background information. The resulting approach can be seen as an extension of previous work using dynamic Bayesian networks (DBNs) to model time-series gene expression data, where the extension incorporates background knowledge.

The second task we examine is modeling longitudinal clinical data. Specifically, we examine the task of modeling data about nursing home patients over time. Given observed variables about patient health, we wish to infer the hidden actual health state of a patient and accurately predict the patient's *health trajectory* - the values of this health state – over the coming weeks. If this task can be performed accurately, one major practical benefit to society is to intervene sooner when a patient's health is in danger of rapid deterioration. Another practical benefit is to identify patterns in patient health in various nursing homes, so that homes with poorer results can be improved.

Both of the applications in this study involve modeling of time-series data. Nevertheless, we use both rather than a single one because these two applications illustrate different issues and require different capabilities within SRL. The gene expression application requires *learning rules* to predict when one gene's expression will change based on changes in other genes; these rules can utilize background knowledge. Inference within the learned models is not particularly difficult. The clinical application, on the other hand, can begin with a reasonable set of human expert-constructed rules, and the major challenge is in the inference, because the key step is inferring the values of the hidden state variables over time. This application can also benefit from learning to further refine the rules, though initially parameter learning (which relies on inference) is sufficient for the task.

2 Gene Fold Prediction

We used time-series gene expression data of environmental stress response experiments, including DNA-damaging agents from Gasch *et al.* [5, 4]. Our study focused on the DNA damage checkpoint pathway because it has been widely studied. The time-series expression data was discretized by determining the relative change in expression from one time step to the next, i.e. comparing the expression levels between two consecutive time series measurements. The time-series data were discretized into one of three possible discrete values by comparing two consecutive time series measurements: if the change increased by 0.3, we consider the expression to be up-regulated, if the change decreased by 0.3, we consider the expression to be down-regulated, otherwise we say the expression stayed the same.

There are many other spatial and molecular interactions that are not captured by expression data, thus we incorporated other relational sources of data to facilitate learning. Known transcription factors for specific target genes can allow the learning algorithm to focus on specific proteins that are known to interact with the DNA of the target gene and potentially discover combinations of transcription factors (pairs, triples, etc.) required to trigger a change in expression of a particular set of genes. Because transcription factors can also interact with other proteins or metabolites on their way to activating gene expression, background knowledge of proteins that are known to interact with each other were included to allow for the discovery of novel proteins in the pathway. Furthermore, an estimated 30% of proteins need to be phosphorylated in order to trigger a change in the protein's function, activity, localization and stability [6]. Thus, background knowledge about a large number of protein phosphorylation in yeast was also included [2].

We aim to link known interactions with gene expression activity to possibly learn new mechanisms. We do this by associating the up- or down-regulation of specific genes from the previous time step with its transcription factor, a protein it might interact with, or a phosphorylation event. We assume that an event in the previous time step will contribute to the change in expression at the current time. This assumption does not necessarily hold for all biological activity but a similar assumption, that of using a gene's expression level to approximate the activity of other genes within the same pathway, have been used by others [14].

The ILP system, Aleph [13], was initially used to learn rules from the data. The following are three examples of rules learned:

Rule 1 up(GeneA,Time,Expt) :-
previous(Time,Time1), down(GeneA,Time1,Expt), interaction(tof1,GeneA), up(tof1,Time1,Expt),
function(GeneA,'CELL CYCLE AND DNA PROCESSING:cell cycle:mitotic cell
cycle and cell cycle control:cell cycle arrest').

Rule 2 up(GeneA,Time,Expt) :-
previous(Time,Time1), down(GeneA,Time1,Expt),
phosphorylates(GeneA,GeneE), up(GeneE,Time1,Expt),
transcriptionfactor(GeneF,GeneE), down(GeneF,Time1,Expt),
transcriptionfactor(GeneF,cdc20), down(cdc20,Time1,Expt).

Rule 3 up(GeneA,Time,Expt) :-
previous(Time,Time1), down(GeneA,Time1,Expt),
interaction(GeneE,GeneA), down(GeneE,Time1,Expt),
interaction(GeneE,mms4), down(mms4,Time1,Expt),
function(GeneA,'METABOLISM').

These rules all specify the activity of specific genes involved in the larger DNA damage pathway. For further details, see [12].

In order to develop a model that would allow us to understand the mechanism of the underlying regulatory network, we wanted to use the rules to construct a Dynamic Bayesian Network and learn the parameters. Though the approach seems to be efficient, the process of creating a ground DBN loses the relational information that was gained using ILP. We wanted to maintain the relational information learned as it could potentially provide a better understanding of the regulatory mechanisms involved. For example, there are many interactions between regulators and promoter regions that do not necessarily result in the expression of a particular gene, only some particular combination would result in expression. This interpretation would be lost without the relational information. Furthermore, the relational models can exploit parameter tying which helps in accelerating the learning process. In the next section, we outline our proposed relational learning method.

2.1 Utilizing a Relational Learner

The ILP rules are learned to predict a target predicate. We interpret the rules probabistically. In the spirit of SRL models, associated with every rule y : $x_1^i, .., x_n^i$ is a conditional probability tables $P(y|x_1^i, ...x_n^i)$. Note that this relational representation is similar to several logic based formalisms such as Bayesian Logic Programs [8],Markov Logic Networks [1] etc. It should be specified that our models are directed (unlike MLNs) and is in the spirit of other directed relational formalisms. As can be seen, there could be more than one rule learned using ILP for a single target predicate (denoted by the superscript i). In many cases, a single parameterized rule can result in multiple instantiated sets of parents that influence a single ground target variable. Also, there could be a number of rules for a single target predicate. Hence there is a necessity for combining the distributions at multiple levels. The multiple instances of a single rule are combined using the mean combining rule while the distributions that arise due to the different rules are combined using the weighted mean combining rule.

Let us assume that each ILP rule S_i ('rule i' for short) for the target predicate Y has k influents, X_1^i through X_k^i (which we jointly denote as $\mathbf{X}^i$), that influence the target variable. When this rule is instantiated or "unrolled" on a specific database, it generates multiple, say m_i, sets of influent instances, which we denote as $\mathbf{X}_1^i \ldots \mathbf{X}_{m_i}^i$. This is shown in Figure 1. In the figure, the instantiations of a particular statement are combined with the *mean* combining rule. The distributions resulting from the different ILP rules are combined via the Weighted-Mean combining rule. Recall that in our problem the variable Y refers to the predicate *up*.

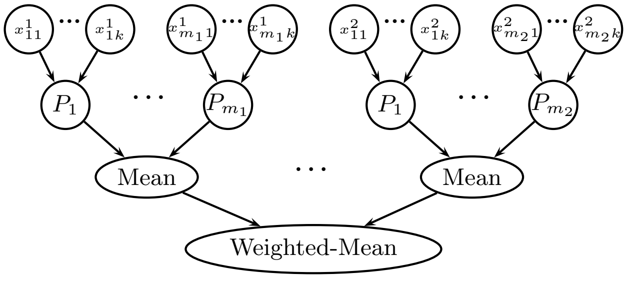

Fig. 1. Unrolling ILP rules

The role of the combining rule is to express the probability $P_i(Y|\mathbf{X}_1^i \ldots \mathbf{X}_{m_i}^i)$ as a function of the probabilities $P_i(Y|\mathbf{X}_j^i)$, one for each j, where P_i is the CPT associated with S_i. For example, with the mean combining rule, we obtain:

$$P(y|\mathbf{X}_1^i \ldots \mathbf{X}_{m_i}^i) = \frac{1}{m_i} \sum_{j=1}^{m_i} P_i(y|\mathbf{X}_j^i) \qquad (1)$$

If there are r such rules, we need to estimate the conditional probability $P(Y|X_{1,1}^1 ... X_{m_r,k}^r)$. If w_i represents the weight of the i^{th} combining rule, the conditional of Y is :

$$P(Y|X_{1,1}^1 ... X_{m_r,k}^r) = \frac{\sum_{i=1}^r w_i P(Y|\mathbf{X}_1^i \ldots \mathbf{X}_{m_i}^i)}{\sum_{i=1}^r w_i} \qquad (2)$$

Learning algorithms based on Gradient descent that employs either mean-squared error or log-likelihood and on EM have been presented in [11]. We are currently investigating the use of these methods in learning the CPTs associated with the ILP rules. It has been established in [11, 7] that learning with the presence of combining rules in relational models is very efficient compared to unrolling them to a propositional model. These results form the basis of our current research to directly learn the parameters of the ILP rules. Lastly, it should be mentioned that unlike the other domain, the inference in this domain is not a hard problem. This is due to the fact that all the influents of the different ILP rules are observed and hence the inference problem reduces to computing the probability distribution as presented in equation 2.

3 Nursing Home DataSet - Predicting Chronic diseases

Chronic diseases can comprise a significant portion of a life span and can have impact on a range of aspects of functional and biological health. Due to the long natural history of chronic disease, there is more potential for interaction with other diseases (e.g. diabetes and atherosclerosis), risk factor exposures (e.g. cigarette smoking and obesity) and age-related declines in physiology (e.g. changes in brain tissue and temporally accumulated arterial damage). These effects appear to leave the individual more at risk for developing other diseases, a condition known as comorbidity, which refers to the occurrence of one or more other diseases among people with an index or baseline disease. Comorbidity has been shown to be associated with mortality, length of hospital stay and disability thus when assessing a person's overall state of health, accounting for their total disease burden can improve diagnostic and prognostic efficiency. In elderly populations, comorbidity occurs frequently and as the aged population increases in our society, the assessment of comorbidity has become progressively more important, particularly for health policy makers and administrators in regard to health costs and health planning. Thus it is necessary to model disease behavior in the elderly as a multidimensional stochastic system evolving according to internally programmed age dynamics, the dynamics of the chronic diseases and comorbidity, and the interaction of the various dimensions of specific health status of individuals [3].

The data that is used in this analysis comes from a set of nursing home quality indicators that are gathered and maintained by the UW Center for Health Systems Research. All U.S. Nursing homes which have a resident population of over 2,000,00 have been required to assess the health status of each resident quarterly beginning in 1998. These assessments are standardized in United States and the assessment instrument (resident assessment instrument - RAI) consists of 250 separate items that are grouped into 12 categories (such as cognition, communication, vision, mood, health conditions, disease diagnosis etc). These are intended to measure dimensions of health that support the clinical processes in the nursing home. In this instrument, multiple items are grouped together to form a summary scale or index, representing the dimension of focus. It could be argued that all biomedical studies involve latent variables of some kind since it is almost impossible to directly measure the variables of primary interest. In the case of development of RAI there was no use of statistical methodology in the design phase. Hence, there is not a statistically sound description of the latent constructs that are actually measured by the instrument. Thus the modeling task involves determining the number of latent constructs and their relationship to the items measured by the instrument in addition to the determination of the temporal dynamics of the co-evolution of the latent constructs.

The general approach to modeling the relationship of measured assessment items to underlying latent constructs is to apply Bayesian networks structure learning methods. The unsupervised learning of an optimal Bayesian network from data is NP-hard and as such constrained versions of Bayesian networks such as nave Bayes have been applied with promising results. The assumption of nave Bayes is that all attributes are conditionally independent given the class label which in the context of the measurement of dementia in the RAI is clearly violated. One of the research ideas is to build a baseline learner based on Tree-Augumented Naive Bayes (TAN). These TAN networks retain the tractability for learning and inference while not making the independence assumption. The challenge in this approach would be to construct a dynamic version of the TANs to monitor the health variables over time.

3.1 Relational Modeling

Yet anther possible solution for this problem would be to construct a hand-crafted DBN and then perform inference on the latent variables. The main bottleneck is that this DBN would involve about 250 variables. It would be extremely tedious for a domain expert to construct this DBN. Also, the number of latent variables will be very high and hence performing inference is not possible in such cases. Hence we resort to using a relational model and performing inference. We are currently in the process of designing this model after collaborations with the domain expert and the work presented is preliminary in nature. In this section, we present a Rao-Blackwellized particle filter algorithm for performing inference.

The model that is being developed is a logical version of a DBN i.e., the model consists of two kinds of relationships. First is the intra-time slice relationship which is described by a set of probabilistic ILP rules as presented in the previous section. The second is the inter-time relationship which describes the probabilistic relationship between certain predicates at two consecutive time-steps (similar to a DBN except that the variables are logical predicates). Given certain observations, the task is to perform inference over the latent predicates. It is important to note that not all the other variables are observed.

Let us denote the set of latent variables as x_t and the observation at time step t is denoted by y_t. Corresponding to the relationships, there are 2 kinds of distributions in the model: the *selection* distribution which specifies the joint distribution $P(x_t, y_t)$ over the variables at the current time-step and the second is the *transition* distribution that specifies the joint distribution over the transitions between the variables at consecutive time-steps. Samples are drawn according to the optimal proposal distribution $P(x_t|x_{t-1}, y_t)$ where,

$$P(x_t|x_{t-1}, y_t) = \frac{P(y_t|x_t)P(x_t|x_{t-1})}{\sum_{x_t} P(y_t|x_t)P(x_t|x_{t-1})} \tag{3}$$

and the weight w_t is given by,

$$w_t \propto P(y_t|x^i_{t-1}) = \sum_{x_t} P(y_t|x_t)P(x_t|x_{t-1})$$

The states are then re-sampled based on the weights of the samples. The health variables can then be marginalized out of the resulting distribution. Since particle filter has been shown to have a poor performance in high-dimensional spaces, we plan to resort to a Rao-Blackwellized version. In this version, we plan to sample the predicates at the next time-step while exactly marginalizing out the variables in the predicate. The intuition is that since the data is highly relational, the variables would be related to one another. For example, a person's chance of having a disease might be influenced by his parents' attributes. Hence, if the person's identity is known, the parents' identities can be obtained. A similar particle filter was proposed recently for Logical Hierarchical HMMs [10]. The authors show that if a large number of relationships are observed, the particle filter can perform very well and is extremely fast compared to the exact inference procedure.

4 Conclusion

In this paper, we have presented two of our current focus time-series problems that pose slightly different challenges. In the gene fold prediction problem, the complexity lies in learning the relational models while in the chronic diseases domain, the problem is to perform efficient inference. We propose to use combining rules to combine the distributions due to different instantiations and different rules. We further propose to use learning algorithms based on EM and gradient descent for learning the parameters of the distributions and the rules. In future, we would also like to extend our work to search through a suite of combining rules and find the best combining rule that fits the data. In the chronic disease domain, we propose to use a logical version of DBN and develop an inference algorithm based on Rao-Blackwellized particle filter. We also currently looking to compare this relational model against a Tree Augumented Naive Bayes (TAN) model as a baseline. Also, it would be useful to compare other approaches such as MCMC that have been recently proposed for relational models[9]. Finally, it would be useful to develop a dynamic model based on SRL principles that can be used to perform both learning and efficient inference for temporal bio-medical applications.

References

1. Pedro Domingos and Mark Richardson. Markov logic: A unifying framework for statistical relational learning. In *Proceedings of the SRL Workshop in ICML*, 2004.
2. J. Ptacek et al. Global analysis of protein phosphorylation in yeast. *Nature*, 438:679–684, 2005.
3. Buntinx F., L. Niclaes, C. Suetens, B. Jans, R. Mertens, and M. Van den Akker. Evaluation of charlsons comorbidity index in elderly living in nursing homes. *Journal of Clinical Epidemiology*, 55:1144–1147, 2002.
4. A.P. Gasch, M. Huang, S. Metzner, D. Botstein, S.J. Elledge, and P.O. Brown. Genomic expression responses to DNA-damaging agents and the regulatory role of the yeast ATR homolog Mec1p. *Mol Biol Cell.*, 12:2987–3003, 2001.
5. A.P. Gasch, P.T. Spellman, C.M. Kao, O. Carmel-Harel, M.B. Eisen, G. Storz, D. Botstein, and P.O. Brown. Genomic expression programs in the response of yeast cells to environmental changes. *Mol Biol Cell.*, 11:4241–57, 2000.
6. Ptacek J. and Snyder M. Charging it up: global analysis of protein phosphorylation. *Trends in Genetics*, 22:545–554, 2006.
7. Manfred Jaeger. Parameter learning for relational bayesian networks. In *Proceedings of the International Conference in Machine Learning*, 2007.
8. K. Kersting and L. De Raedt. Bayesian logic programs. In *Proceedings of the Work-in-Progress Track at ILP*, 2000.
9. B. Milch and S. Russell. General-purpose mcmc inference over relational structures. In *Proceedings of UAI 06*, 2006.
10. S. Natarajan, H. Bui, P.Tadepalli, K. Kersting, and W. Wong. Logical hierarchical hidden markov models for modeling user activities. In *Proceedings of ILP 08*, 2008.
11. S. Natarajan, P. Tadepalli, E. Altendorf, T. G. Dietterich, A. Fern, and A. Restificar. Learning first-order probabilistic models with combining rules. In *Proceedings of the International Conference in Machine Learning*, 2005.
12. Irene M. Ong, Scott E. Topper, David Page, and Vítor Santos Costa. Inferring regulatory networks from time series expression data and relational data via inductive logic programming. In *Proceedings of the Sixteenth International Conference on Inductive Logic Programming*, volume 4455, pages 366–378. Springer Berlin/Heidelberg, 2007.
13. Ashwin Srinivasan. *The Aleph Manual.* University of Oxford, 2001.
14. A. Zien, R. Kuffner, R. Zimmer, and T. Lengauer. Analysis of gene expression data with pathway scores. *Proc Int Conf Intell Syst Mol Biol*, 8:407–17, 2000.