ACTIVE LEARNING FOR BAYESIAN NETWORK MODELS OF BIOLOGICAL NETWORKS USING STRUCTURE PRIORS

Antti Larjo*† and Harri Lähdesmäki*‡
*Department of Information and Computer Science, Aalto University, FI-00076 Aalto, Finland
{antti.larjo, harri.lahdesmaki}@aalto.fi
†Department of Signal Processing, Tampere University of Technology, FI-33101 Tampere, Finland
‡Turku Centre for Biotechnology, Turku University, Turku, Finland

***Abstract*—Active learning methods aim at identifying measurements that should be done in order to benefit a learning problem maximally. We use Bayesian networks as models of biological systems and show how active learning can be used to select new measurements to be incorporated via structure priors. Improved performance of the methods is demonstrated with both simulated and real datasets.**

I. Introduction

Bayesian networks (BN) are models often used for modeling, e.g., genetic regulatory and signaling networks. BNs can be learned from multiple types of data since it is possible to include additional datasets using structure priors, as long as the measurement data can be transformed to prior probabilities. This is usually possible because statistical significance of most of the measurements can be assessed using p-values, and they can be changed to probabilities of edges, as done e.g. in [1].

Here we are interested in selecting measurements to be incorporated through network structure priors so that they maximally benefit the learning of the BN model, which is called active learning. The proposed methods and simulations mimic biological experiments, where setting up the measurements and performing them can be difficult, slow and expensive, thus motivating selection of the measurements carefully. The proposed methods are demonstrated to perform better than randomly selecting experiments.

II. Methods

A. Bayesian networks

A Bayesian network is defined as a pair (g, θ), where g is a directed acyclic graph (DAG), which is a graphical representation of the conditional independencies between random variables $\{X_1, ..., X_N\}$, and θ is the set of parameters for the conditional probability distributions of these variables. DAG g allows factorization of the joint distribution over the random variables as

$$P\left(X_1, ..., X_n | g, \theta\right) = \prod_{i=1}^{n} P\left(X_i | Pa_g(X_i), \theta_i\right), \qquad (1)$$

where $Pa_g(X_i)$ denotes the set of parents of node X_i in g, and θ_i are the parameters for the distribution of X_i conditional on its parents.

Structure learning tries to identify the BN structure of the network that most probably generated the data. Thus, the aim is to find the DAG (or a set of DAGs) with the highest posterior probability given the data, which is given as $P(g|D) = P(D|g)P(g)/P(D)$, where $P(g)$ is the prior probability of g, $P(D) = \sum_{g'} P\left(D|g'\right) P(g')$ is the prior probability of data (summing over all possible DAG structures), and $P\left(D|g\right) = \int_\theta P\left(D|g, \theta\right) P\left(\theta|g\right) d\theta$.

Often the goal in structure learning is to find a single best network that maximizes the posterior probability and would represent the underlying true network as well as possible. However, for most applications (like prediction) it is generally more reasonable to instead utilize the whole posterior probability distribution and to perform a Bayesian analysis. Though exhaustive evaluation of the posterior distribution is possible for only the smallest structure spaces (say $N \leq 6$) due to number of DAGs growing super-exponentially with number of nodes, sampling from posterior distribution is possible with Markov chain Monte Carlo (MCMC), which is utilized also in our study.

B. Active learning

Active learning methods aim at choosing which queries or measurements should be done next, given the current data, so as to learn as efficiently as possible. Such methods are of particular interest in scenarios where measurements can be difficult and/or expensive to make.

We separate between two types of data: one we call *expression* data but instead of gene expression values it can as well be any other kind of data about the state of the nodes, e.g., concentrations of the molecules represented by the nodes, and the other type of data we call *binding* data, which measures (regulatory) relationships between nodes, i.e. gives more direct evidence for the edges of the network. The type of binding measurements we consider here is such that doing a measurement for node is actually measuring all outward links from this node. Such measurements could be, e.g., chromatin immunoprecipitation sequencing (ChIP-seq) for transcription factors or protein interaction measurements.

In learning BN structure from purely expression type of data, it is generally not possible to find the correct BN

structure with only observational data but instead perturbations to (some of) the node states are required in order to break so called equivalence classes, which are comprised of more than one structure producing the same combined probability distribution. On the other hand, structure priors are another way of breaking equivalence classes, which is what happens in the scenario of this study.

Methods of active learning have been presented for BN structure learning by utilizing expression type of data, such as in [2], [3], [4], [5]. Learning from binding type of measurements has been studied e.g. in the context of protein-protein interactions, such as [6], but not in BN context. Next we present two methods suited for this.

C. Active learning through structural priors

Binding information is included through the prior and expression data is included through the likelihood as usual. The posterior probability of a graph structure g is thus

$$P(g|D,A) = \frac{P(D|g)P(A|g)P(g)}{\sum_{g'} P(D|g')P(A|g')P(g')} \tag{2}$$

$$= \frac{P(D|g)P(g|A)}{\sum_{g'} P(D|g')P(g'|A)}, \tag{3}$$

where D is the expression data and A is the adjacency probability matrix (i.e., $A(i,j) = P(n_i \to n_j)$) that is used in constructing the prior $P(g|A)$ over DAG structures, which we define as follows

$$P(g|A) = \prod_{i,j:n_i \to n_j \in g} A(i,j) \prod_{i,j:n_i \to n_j \notin g} (1 - A(i,j)). \tag{4}$$

We assume no other prior information is available about the structure so initially all the elements of A are 0.5.

The utilization of additional binding data goes in principle as follows: 1) selection of node i for which measurement is done, 2) obtaining measurement vector c_i (containing p-values), 3) updating probabilities of edges based on the p-values, and 4) updating $P(g|A)$ by incorporating the new calculated probabilities to A. Active learning tries to address step 1) so as to make the learning maximally beneficial.

The binding measurements are assumed to be given as p-values. Carrying out a measurement for node i thus gives a vector $c_i = [p_{i,1}\ p_{i,2}\ \cdots\ p_{i,N}]$, where $p_{i,k}$ is the p-value of measurement of binding of gene i to gene k.

As in [1], we assume the measurement p-values to be exponentially distributed in case the edge is found in the underlying network structure and, by definition, uniformly distributed if the edge is not present:

$$P_\lambda\left(p_{i,j}|n_i \to n_j \in g\right) = \frac{\lambda e^{-\lambda p}}{1 - e^{-\lambda}} \tag{5}$$

$$P_\lambda\left(p_{i,j}|n_i \to n_j \notin g\right) \sim U(0,1) \tag{6}$$

The measurements $p_{i,k}$ can be assumed to be independent of D, A and each other so

$$P\left(c_i|g,\lambda,D,A\right) = P\left(p_{i,1},p_{i,2},\ldots,p_{i,N}|g,\lambda\right) \tag{7}$$

$$= \prod_{k=1}^{N} P\left(p_{i,k}|g,\lambda\right). \tag{8}$$

After an observation c_i, the values of A can be updated by calculating the probability of an edge from a p-value in the same way as in [1]:

$$A_{i,j} = P(n_i \to n_j|p_{i,j}) \tag{9}$$

$$= \frac{1}{\lambda_H - \lambda_L} \int_{\lambda_L}^{\lambda_H} \frac{\alpha\beta}{\alpha\beta + (1 - e^{-\lambda})(1-\beta)} \mathrm{d}\lambda, \tag{10}$$

where $\alpha = \lambda e^{-\lambda p_{i,j}}$ and $\beta = P(n_i \to n_j)$ is the prior of an edge (i.e., before observation) and marginalization over the scale parameter λ is done to avoid selecting only a single value. The integral must be solved numerically, for which we use recursive adaptive Simpson quadrature.

Probability of a structure after observing a new binding measurement is given by

$$P(g|c_i,D,A) = \frac{P(D|g)P(g|\widetilde{A})}{\sum_{g'} P(D|g')P(g'|\widetilde{A})}, \tag{11}$$

where $\widetilde{A}$ is A where $A(i,j), \forall j \in \{1 \ldots N\}$ have been updated with c_i using (10).

1) KL method: We utilize the approach in [2], which was presented for gene expression data, and modify it to cover binding measurements. Selection of the most "valuable" node to be measured is made based on the maximal expected Kullback-Leibler divergence between the distributions of graph structures before and after the measurement, i.e.,

$$V^* = \underset{i \in V_{av}}{\operatorname{argmax}}\, E_{c_i} KL\left(P(G|c_i,D,A)||P(G|D,A)\right) \tag{12}$$

$$= \underset{i \in V_{av}}{\operatorname{argmax}} \int_{c_i} P(c_i|D,A) \left[\sum_g P(g|c_i,D,A) \cdots \right.$$

$$\left. \times \log \frac{P(g|c_i,D,A)}{P(g|D,A)}\right] \mathrm{d}c_i \tag{13}$$

$$= \underset{i \in V_{av}}{\operatorname{argmax}} \sum_g P(g|D,A) \int_{c_i} P(c_i|g,D,A) \cdots$$

$$\times \log P(g|c_i,D,A) \mathrm{d}c_i, \tag{14}$$

where V_{av} denotes the set of nodes available for measurement. The calculation of this equation cannot be done analytically so we use a numerical solution, namely Monte Carlo integration.

2) Entropy minimization method: As the inference of network structure can be seen to aim at minimizing uncertainty about presence of edges, we decided to simply choose the node for which the sum of entropies for outward edges is highest, i.e.,

$$V^* = \underset{i \in V_{av}}{\operatorname{argmax}} \sum_{j=1}^{N} \Big(\hat{P}_{i,j} \log_2 \hat{P}_{i,j}$$

$$+ (1 - \hat{P}_{i,j}) \log_2 (1 - \hat{P}_{i,j})\Big) \tag{15}$$

where $\hat{P}(n_i \to n_j)$ are the posterior probability estimates for the edges estimated from the sampled DAGs.

This method is very fast to evaluate but on the downside the selection of the measures node is not guaranteed to be the one minimizing global entropy maximally. This problem is

due to dependencies introduced by acyclicity constraints. For example, if measuring node i and thus receiving probability for edge $i \to j$, we would need to set $P(j \to i)$ to $1 - P(i \to j)$ in matrix A. However, in biological systems cycles are very common so doing this is likely to cause worse performance than leaving the priors of unmeasured nodes untouched and we therefore use the approximate method.

III. Results

A. Simulated gene regulatory networks

To test the performance of the two active learning methods, we generated random networks with 8 nodes and random parameters. Both 30 simulated gene expression measurements as well as artificial ChIP-seq measurements for all nodes were generated. To get an estimate for the "true" edge posterior probability, we used MCMC with burn-in of 2,000,000 and sample size of 25,000 DAGs. The simulated learning procedure consisted of initial learning with MCMC using only the expression measurements (burn-in 200,000, sample size 5,000). This sample was then given to both active learners, which suggested nodes to be measured and after measurement the samples were updated with the new data. To measure the performance of the active learners, L_1 error is used, which is calculated as in [2], [3]

$$\begin{aligned} L_1(P_t) = \sum_{i=1}^{n} \sum_{j=i+1}^{n} & I_{g^*}(X_i \to X_j)(1 - P_t(X_i \to X_j)) \\ & + I_{g^*}(X_i \leftarrow X_j)(1 - P_t(X_i \leftarrow X_j)) \\ & + I_{g^*}(X_i \not\sim X_j)(1 - P_t(X_i \not\sim X_j)), \end{aligned} \tag{16}$$

where $P_t(\cdot) = P(\cdot|D_{1:t})$ is the posterior marginal probability of an edge given data points up to index t, and $I_{g^*}(x)$ is the indicator function which takes value 1 if x is present in the true structure g^* and 0 otherwise.

Results over 100 iterations are shown in Fig. 1 and Fig. 2, which demonstrate that active learning can achieve a consistent improvement in learning the BN network structure. Results with other node counts N were similar and in general the differences between the results of KL and entropy methods were small.

B. Signaling network

As another slightly different test case we used phosphoprotein measurements from [7], which are made from a signaling network of 11 proteins. From the dataset we randomly selected 100 observational datapoints as the initial data. As the "true" network we used the maximum a posteriori DAG obtained by learning using the whole dataset. High-throughput "binding" measurements for signaling networks correspond to measurements of protein-protein interactions, which can be obtained e.g. using mass-spectrometry based protein-protein interactions using tagged proteins as baits. In this case the data being generated is not directional but just tells that the proteins interact. Therefore we generated the binding data so that $A(i,j) = A(j,i)$ for every true edge $n_i \to n_j \in g$.

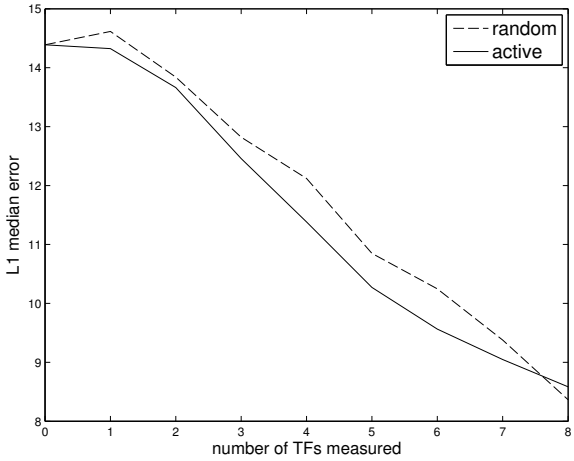

Fig. 1. L_1 median errors for the KL method from 100 runs, each with a random true DAG having 8 nodes and random parameters.

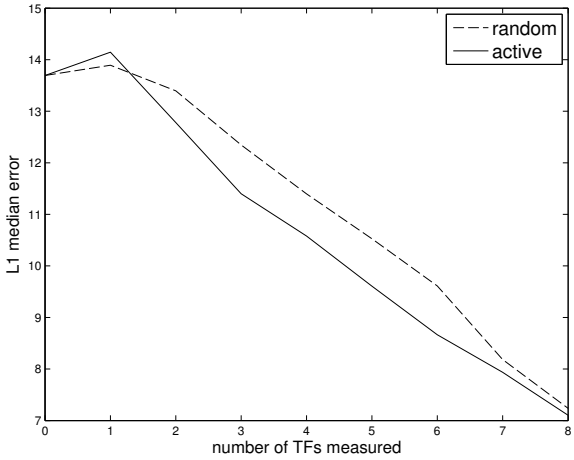

Fig. 2. L_1 median errors for the entropy method from 100 runs, each with a random true DAG having 8 nodes and random parameters.

As the network has 11 nodes and using KL method proved quite slow, Fig. 3 and Fig. 4 present results only for the entropy method. Performance of active learning was measured calculating Euclidean distances between the current edge posterior probabilities and the "true" ones estimated using the whole dataset. Parameters for the MCMC simulations were as follows:

- edge posterior probability estimation: burn-in 500,000, sample size 50,000
- initial learning: burn-in 500,000, sample size 10,000
- update rounds: burn-in: 50,000, sample size 10,000.

As with simulated data, active learning for signaling network data via structure priors provides substantial improvement in learning performance.

Although on average the utilization of active learning methods result in improved learning (Figs. 1-4), they are not guaranteed to do so all the time, given randomness in data. Fig. 5 shows difference of L_1 errors between a random and active learner. Even though the vast majority of trajectories

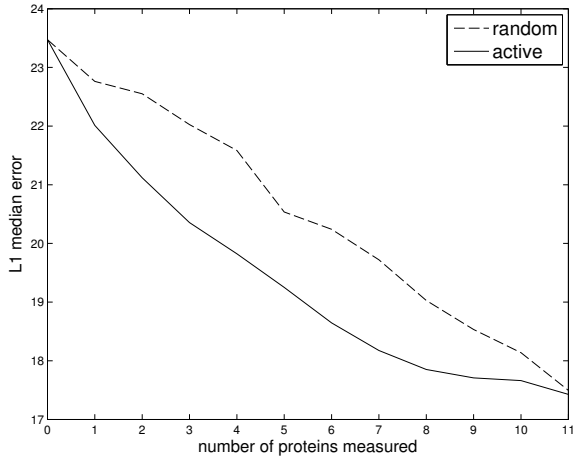

Fig. 3. L1 median errors for the entropy method from 100 runs, for the PPI network.

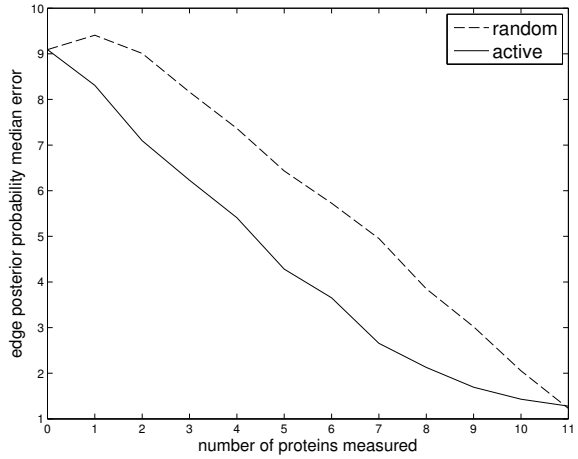

Fig. 4. Edge posterior probability median errors from the same 100 runs as in Figure 3.

show better performance for active learning, there are still some that do worse at least during some measurements. Nevertheless, active learning achieves better expected learning performance.

IV. Conclusion

Reducing the amount of resources needed for obtaining useful information about a (biological) system is the main motivation for active learning methods. We presented how active learning can be utilized with the ability of Bayesian networks to incorporate additional measurements through structure priors.

The problems associated with acyclicity are an obvious limitation for learning many types of biological networks and also methods presented here. These problems can be overcome by considering dynamic Bayesian networks (DBNs) but a problem with applying them to biological problems is the scarcity of time-series data. Even though learning DBNs from static data has been studied [8], we decided to utilize static BNs due to greater applicability.

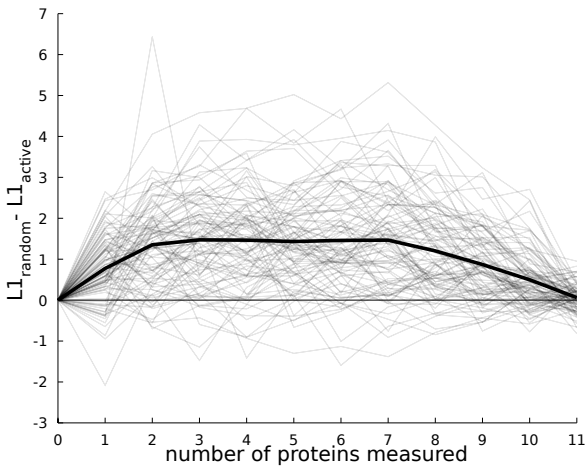

Fig. 5. L1 error differences for the iterations in Fig. 3. Black line is the mean.

The simulated binding measurements give data for all outward edges at the same time. Using data from measurements where only a selected edge is probed at a time (such as yeast two-hybrid assays) is easy to include with small modifications. Combining both expression-type measurements and binding measurements in active learning is also easily possible in our framework but requires a cost function for both types of measurements.

ACKNOWLEDGEMENT

This work was supported by the Academy of Finland (Centre of Excellence in Molecular Systems Immunology and Physiology Research (2012-2017)), EU FP7 grant (EC-FP7-SYBILLA-201106), and TISE graduate school.

References

[1] A. Bernard and A. J. Hartemink, "Informative structure priors: Joint learning of dynamic regulatory networks from multiple types of data." in *Pacific Symposium on Biocomputing (PSB05)*, R. B. Altman, A. K. Dunker, L. Hunter, T. A. Jung, and T. E. Klein, Eds., vol. 10, 2005, pp. 459–470.

[2] K. Murphy, "Active learning of causal Bayes net structure," 2001. [Online]. Available: citeseer.ist.psu.edu/murphy01active.html

[3] S. Tong and D. Koller, "Active learning for structure in Bayesian networks," in *Proceedings of the International Joint Conference on Artificial Intelligence*, 2001. [Online]. Available: citeseer.ist.psu.edu/article/tong01active.html

[4] I. Pournara and L. Wernisch, "Reconstruction of gene networks using Bayesian learning and manipulation experiments," *Bioinformatics*, vol. 20, no. 17, pp. 2934–2942, Nov 2004.

[5] A. Larjo, H. Lähdesmäki, M. Facciotti, N. Baliga, O. Yli-Harja, and I. Shmulevich, "Active learning of bayesian network structure in a realistic setting," in *Proceedings of the 5th International Workshop on Computational Systems Biology (WCSB 2008)*, pp. 85–88.

[6] T. P. Mohamed, J. G. Carbonell, and M. K. Ganapathiraju, "Active learning for human protein-protein interaction prediction," *BMC Bioinformatics*, vol. 11 Suppl 1, p. S57, 2010.

[7] K. Sachs, O. Perez, D. Pe'er, D. A. Lauffenburger, and G. P. Nolan, "Causal Protein-Signaling Networks Derived from Multiparameter Single-Cell Data," *Science*, vol. 308, no. 5721, pp. 523–529, 2005.

[8] H. Lähdesmäki and I. Shmulevich, "Learning the structure of dynamic bayesian networks from time series and steady state measurements," *Machine Learning*, vol. 71, no. 2-3, pp. 185–217, 2008.