

# Bayesian Network Webserver: a comprehensive tool for biological network modeling

Jesse D. Ziebarth<sup>1,2,†</sup>, Anindya Bhattacharya<sup>1,2,†</sup> and Yan Cui<sup>1,2,\*</sup>

<sup>1</sup>Department of Microbiology, Immunology and Biochemistry and <sup>2</sup>Center for Integrative and Translational Genomics, University of Tennessee Health Science Center, Memphis, TN 38163, USA

Associate Editor: Igor Jurisica

## ABSTRACT

**Summary:** The Bayesian Network Webserver (BNW) is a platform for comprehensive network modeling of systems genetics and other biological datasets. It allows users to quickly and seamlessly upload a dataset, learn the structure of the network model that best explains the data and use the model to understand relationships between network variables. Many datasets, including those used to create genetic network models, contain both discrete (e.g. genotype) and continuous (e.g. gene expression traits) variables, and BNW allows for modeling hybrid datasets. Users of BNW can incorporate prior knowledge during structure learning through an easy-to-use structural constraint interface. After structure learning, users are immediately presented with an interactive network model, which can be used to make testable hypotheses about network relationships.

**Availability and implementation:** BNW, including a downloadable structure learning package, is available at <http://compbio.uthsc.edu/> BNW. (The BNW interface for adding structural constraints uses HTML5 features that are not supported by current version of Internet Explorer. We suggest using other browsers (e.g. Google Chrome or Mozilla Firefox) when accessing BNW).

**Contact:** ycui2@uthsc.edu

**Supplementary information:** Supplementary data are available at *Bioinformatics* online.

Received on May 6, 2013; revised on August 6, 2013; accepted on August 7, 2013

## 1 INTRODUCTION

Bayesian network modeling has been successfully applied to many biological problems, ranging from assessment of disease risk (Sebastiani *et al.*, 2005) to identifying cellular interactions that link genotypes with complex phenotypes (Miyairi *et al.*, 2012). However, widespread use of Bayesian networks by biologists has been hindered by a lack of easy-to-use tools that can perform all of the steps needed to create a Bayesian network model and use it to make predictions. In particular, while several tools for Bayesian network structure learning from data exist (e.g. Wilczyński and Dojer, 2009; Scutari, 2010), they often require additional tools to make predictions with the models, increasing both the learning curve for inexperienced users and the time required to perform modeling. In response, we have

created the Bayesian Network Webserver (BNW) to provide a web-based tool for comprehensive Bayesian network modeling. BNW allows users to load a text file containing a dataset, identify the structure of the network that best explains the data, perform parameter learning of the network and use the network to make predictions. These predictions are a crucial step in Bayesian network modeling, as they provide testable hypotheses that can be used to assess model structures and fully understand the relationships between variables in the network.

## 2 TOOL DESCRIPTION

### 2.1 Structure learning

The first step in Bayesian network modeling of a dataset is structure learning, or identifying which directed edges between variables (nodes) should be included in the network to represent the conditional dependencies observed in the data. As performing exhaustive searches is an NP-hard problem, structure learning is often the most computationally intensive part of Bayesian network modeling. To increase the speed of structure learning, BNW integrates several recent improvements in structure learning algorithms, allowing for an exhaustive search for hybrid networks containing as many as 19 variables within 2 min (Supplementary Data). To begin structure learning in BNW, we perform an exhaustive search, given structural constraints specified by the user, of local structures, and score the local structures using a metric that allows for modeling hybrid datasets (i.e. datasets containing both discrete and continuous variables) through the use of conditional Gaussian distributions (Boettcher and Dethlesen, 2003). After calculating local scores, BNW searches for the *k*-best global optimal structures, using a user-specified value of *k* (Silander and Myllymaki, 2006; Tian *et al.*, 2010). Model averaging over the *k*-best structures is then performed to select directed edges to include in the final network structure by calculating the posterior probability of each edge using a weighted average over the *k*-best networks. Model averaging can be used to reduce the risk of over-fitting data to a single model and may be particularly advantageous when learning network models using small datasets.

BNW includes a structural constraint interface that allows users to specify options that can aid in identifying robust network structures and limit structure searches to biologically meaningful networks by incorporating prior knowledge. First, users can set several options that impact the global properties of the network structure, including the maximum number of parents

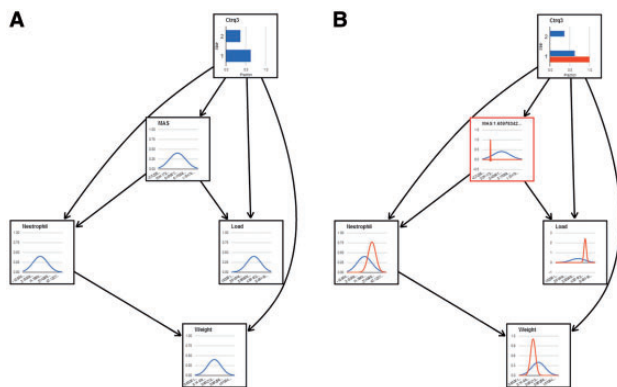
\*To whom correspondence should be addressed.

†The authors wish it to be known that, in their opinion, the first two authors should be regarded as joint First Authors.

for any node in the network and the value of  $k$  when identifying the  $k$ -best network structures. Additionally, a drag-and-drop interface is available for incorporating prior knowledge about interactions between variables in the network search. In addition to banning or requiring specific interactions between pairs of variables, users can assign variables to different tiers and then denote interactions that are allowed within and between tiers. For example, the variables in a systems genetics dataset containing genotypes, intermediate traits (e.g. gene expression traits) and phenotypes can be assigned to tiers to require that networks begin with genotypes and terminate with phenotypes. Also shown on the structural constraint interface page is an estimate of the time required for structure learning in BNW that is updated in real-time based on user settings and the number of nodes and samples in the network. Finally, users can download a structure learning package to more easily learn structures of larger networks or larger values of  $k$ . The output of the downloadable structure learning package can be uploaded to the BNW Web site to visualize the network and make predictions.

## 2.2 Parameter learning and predictions

After structure learning is completed, BNW automatically performs parameter learning of the network using the Bayes Net Toolbox (Murphy, 2001) and displays an interactive network model that can be used to make predictions. Initially, the network model is shown with discrete variables represented as bar charts showing the fraction of samples in each state and continuous variables represented as a line chart of a Gaussian curve fit to the overall distribution of the variable (Fig. 1a). To make predictions, users simply click on a network node and enter a value for that node. Within seconds, new distributions for each of the network variables are shown, allowing for visualization of the impact of setting the variable to the given value (Fig. 1b). BNW



**Fig. 1.** Screenshots of using BNW with a systems biology dataset (Miyairi *et al.*, 2012). The model connects a genotype (*Citr3*), three cellular level phenotypes [neutrophils, pathogen load and macrophage activation status (MAS)] and a disease severity phenotype (weight). (a) The network structure and distributions of the variables learned from the data. (b) Predictions after MAS are observed to have a low value. The MAS node is outlined to indicate that evidence has been entered for this node, and the additional lines show predicted distributions considering the observed value of MAS

provides two prediction modes: an evidence mode, in which predictions are made after observing the value of one or more network variables, and an intervention mode, in which predictions consider an experimental intervention that sets a variable to a particular value. BNW can be used with training/test datasets by using the evidence mode to predict values of test set samples after learning the network structure and parameters with the training set.

## 3 CONCLUSIONS

The BNW is designed to be a comprehensive tool to put the power of sophisticated Bayesian network models into the hands of biologists with little modeling experience. BNW provides network structure learning, parameter learning and predictions in a single package, allowing users to go from a simple tab-delimited input data file to predictions about relationships among variables from a Bayesian network model within minutes. BNW includes features that make it particularly suitable for systems biology and systems genetics. First, it allows for modeling hybrid biological datasets that contain both discrete (e.g. genotypes) and continuous variables, without discretization of continuous variables and its associated information loss. Second, BNW allows users to incorporate prior knowledge when performing structure learning using an interactive interface, focusing structure learning on only those structures that include known regulatory relationships and exclude biologically meaningless interactions. Although BNW was designed to be flexible enough to be applied to a wide variety of datasets, it does have some limitations. It currently supports only static Bayesian networks and cannot use time-series data for dynamic modeling. It also is intended for small network models containing tens of variables and limits structure learning to networks of  $\leq 19$  variables. The performance of BNW, including plots of execution time while varying the number of variables and samples, is further discussed in the Supplementary Data. Tutorials and examples using both real and simulated datasets in BNW are available at <http://compbio.uthsc.edu/BNW>.

## ACKNOWLEDGEMENT

We thank Dr. Jin Tian for providing source codes for the programs used in Tian *et al.* (2010).

**Funding:** This work was supported by The University of Tennessee Center for Integrative and Translational Genomics and Department of Defense [W81XHW-05-01-0227].

**Conflict of Interest:** none declared.

## REFERENCES

- Boettcher, S.G. and Dethlefsen, C. (2003) deal: a package for learning Bayesian networks. *J. Stat. Soft.*, **8**, 1–40.
- Miyairi, I. *et al.* (2012) Host genetics and Chlamydia disease: prediction and validation of disease severity mechanisms. *PLoS One*, **7**, e33781.
- Murphy, K. (2001) The Bayes Net Toolbox for Matlab. *Comp. Sci. Stat.*, **33**, 331–350.
- Scutari, M. (2010) Learning Bayesian networks with the bnlearn R package. *J. Stat. Soft.*, **35**, 1–22.

- Sebastiani,P. *et al.* (2005) Genetic dissection and prognostic modeling of overt stroke in sickle cell anemia. *Nat. Genet.*, **37**, 435–440.
- Silander,T. and Myllymaki,P. (2006) A simple approach for finding the globally optimal Bayesian network structure. In: *Proceedings of the Conference in Artificial Intelligence (UAI2006)*. Cambridge, Massachusetts, USA, pp. 445–452.
- Tian,J. *et al.* (2010) Bayesian model averaging using the k-best Bayesian network structures. In: *Proceedings of the Conference on Uncertainty in Artificial Intelligence (UAI2010)*. pp. 589–597.
- Wilczyński,B. and Dojer,N. (2009) BNFinder: exact and efficient method for learning Bayesian networks. *Bioinformatics*, **25**, 286–287.