A Software Toolkit for Learning Dynamic Graphical Models of Gene Regulatory Structure from Microarray Data

William H. Hsu[1], Youping Deng[2], Judith L. Roe
[1]bhsu@cis.ksu.edu, Kansas State University; [2]ypdeng@ksu.edu, Kansas State University

General-purpose software tools for probabilistic inference are abundant, but similar tools for learning graphical models from data have been explored only recently. Specifically, much of the work in Bayesian approaches to data mining from microarray data has focused on clustering of genes to identify related families of interest, but less attention has been given so far to the problem of automatically discovering regulatory dependencies and whole pathways and networks. Techniques for learning the structure of both atemporal and dynamic Bayesian networks (DBNs) from microarray data have produced models of some organisms (e.g., yeast) of which some regulatory relationships have been validated by bootstrap sampling, then corroborated using independent biological experiments. Pe&#65533;er et al. recently showed how this approach, though not yet sufficiently robust for discovery of whole-organism network models, is sufficient for discovering global active regulators corresponding to the root or source (nondominated) genes in a network model. To facilitate evaluation of Bayesian network (BN) and DBN models produced using machine learning and biological (wet lab) experiments, we have developed BNJ, a software toolkit with a common application programmer interface (API) for structure learning, interactive network construction and validation, and inference using evidence. One important method for evaluating DBN models is to measure their predictive loss given evidence [Mu02]. This entails learning the parameters (conditional pdfs) of the network model and using it to perform predictive inference using real or simulated data. These may include the expression levels of known global active regulators, consisting of data acquired independently of the original microarray experiments (e.g., using northern blot analysis). BNJ therefore implements a full suite of traditional algorithms for exact inference (clustering, conditioning, and variable elimination) and approximate inference (by importance sampling and using other Markov chain Monte Carlo sampling approaches). This supports further experimentation with models learned from data, in the context of data held out for validation by inference. Evaluation and Application to Genomic Modeling. Friedman et al. (2000) and Pe&#65533;er et al. (2002) consider two kinds of features or relations represented in Bayesian network models of gene regulation: order (ancestry) and Markov (isolation). The Sparse Candidate algorithm, a score-based algorithm based on pre-filtering of candidate parent variables in the network by their mutual information with child variables, is used to learn these relations between variables representing the expression level of 800 out of 6177 S. cerevisiae open reading frames (ORFs), which correspond to sequenced genes, meeting criteria for being cell cycle-dependent. Friedman et al. then evaluate performance through robustness analysis using bootstrap sampling, to estimate the statistical confidence in discovered relationships. We experiment with the yeast cell cycle data of Spellman et al. (1998) and stress data of Gasch et al. (2000), using our reimplementation of the Sparse Candidate algorithm and a new stochastic structure search lalgorithm. If existing microarray data is randomly sampled many times and used to build multiple BN models, and a link, pathway or parents-child relationship among two or more genes is frequently found using these many samples. If this frequency is high enough, the finding may be used to suggest confirmatory microarray experiments, providing a decision support tool, or recommender system, for biological researchers. This is a significant benefit because of the high cost of data acquisition.