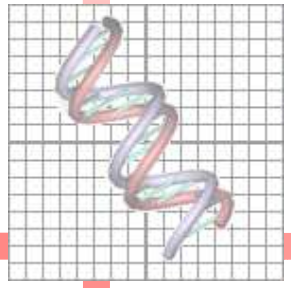


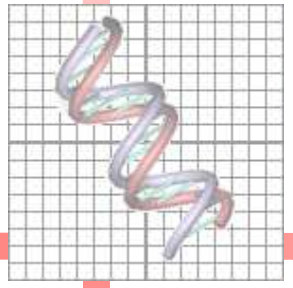
Learning Bayesian Network Models of Gene Regulation

CIBM Retreat October 3, 2003

Keith Noto

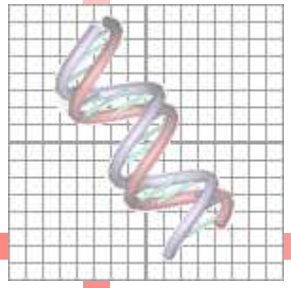
Mark Craven's Group

University of Wisconsin-Madison



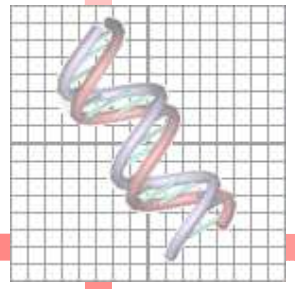
Abstract

Our knowledge of gene-regulatory networks, even for well studied organisms like E. coli, is nowhere near complete. We are interested in learning Bayesian-network models of gene regulation from high-throughput data sources such as microarrays. We are exploring an approach that uses "templates" of various regulatory mechanisms to constrain the search through the space of possible networks. Our initial approach (i) is able to learn the cellular conditions which largely influence regulation, (ii) includes a representation of each regulator under these influences, and (iii) can represent the set of genes which are regulated by each regulator. We believe this approach will help to confirm predicted and suggest new regulatory pathways, as well as offer a model of gene regulation that can be integrated into models of other cellular processes.



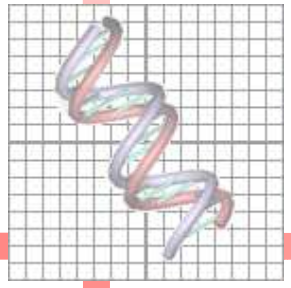
Task

- Given: Microarray data from several experiments under varying conditions in E.Coli. Sequence coordinates of transcription factor binding sites and transcription start sites
- Do: Learn model characterizing transcription regulation relationships among genes



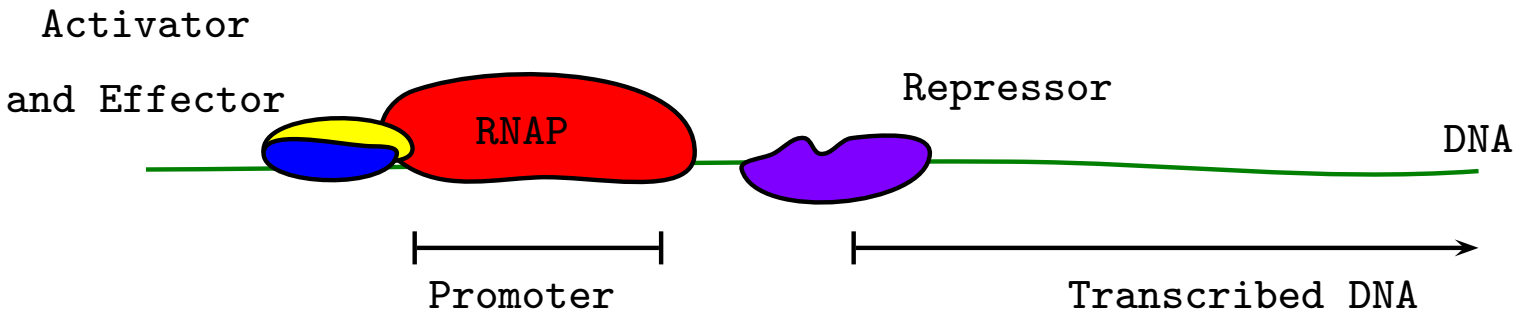
Motivation

- Predict expression
- Confirm predicted regulatory pathways
- Suggest new regulatory pathways
- Identify relevant cellular conditions that influence certain regulatory pathways and describe these effects
- Create a model for gene regulation that can be combined with other models (*e.g.* metabolic pathway models)

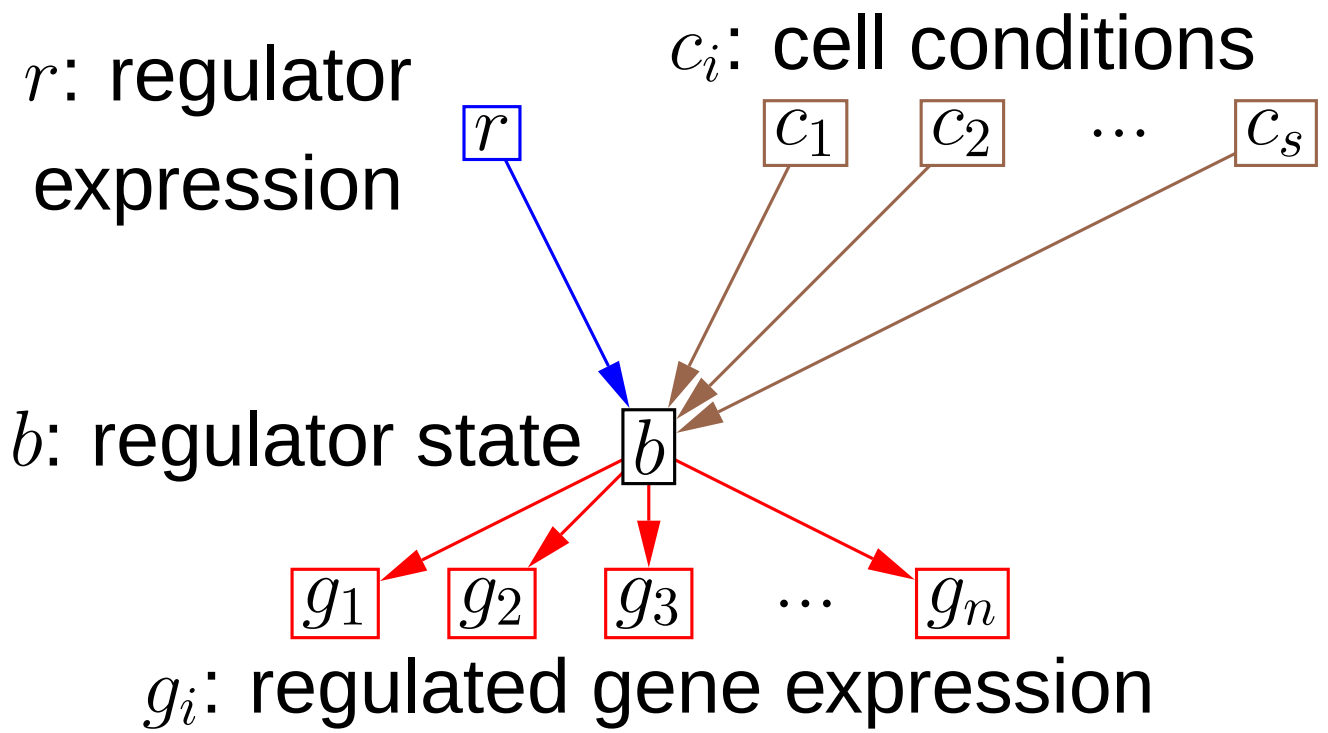


Transcription Regulation

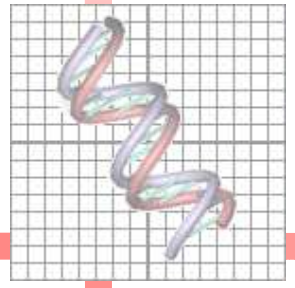
Before transcribing DNA, RNA Polymerase (RNAP) must bind to the promoter. An *activator* is often necessary to make the promoter effective. Also, *repressors* may prevent transcription by binding to the DNA in a way that hinders the binding or movement of RNAP. The activity of both types of regulators can be determined by small molecules called *effectors* present in the cell.



Model Template



In a Bayesian network, each variable is represented by a node, and the probability distribution over its possible values is determined by the nodes connected to it by incoming arcs (its parents). This is typically represented by a *conditional probability table* (CPT) with one probability distribution for each possible combination of parent values. There are algorithms for learning these probability distributions and using them to answer queries.

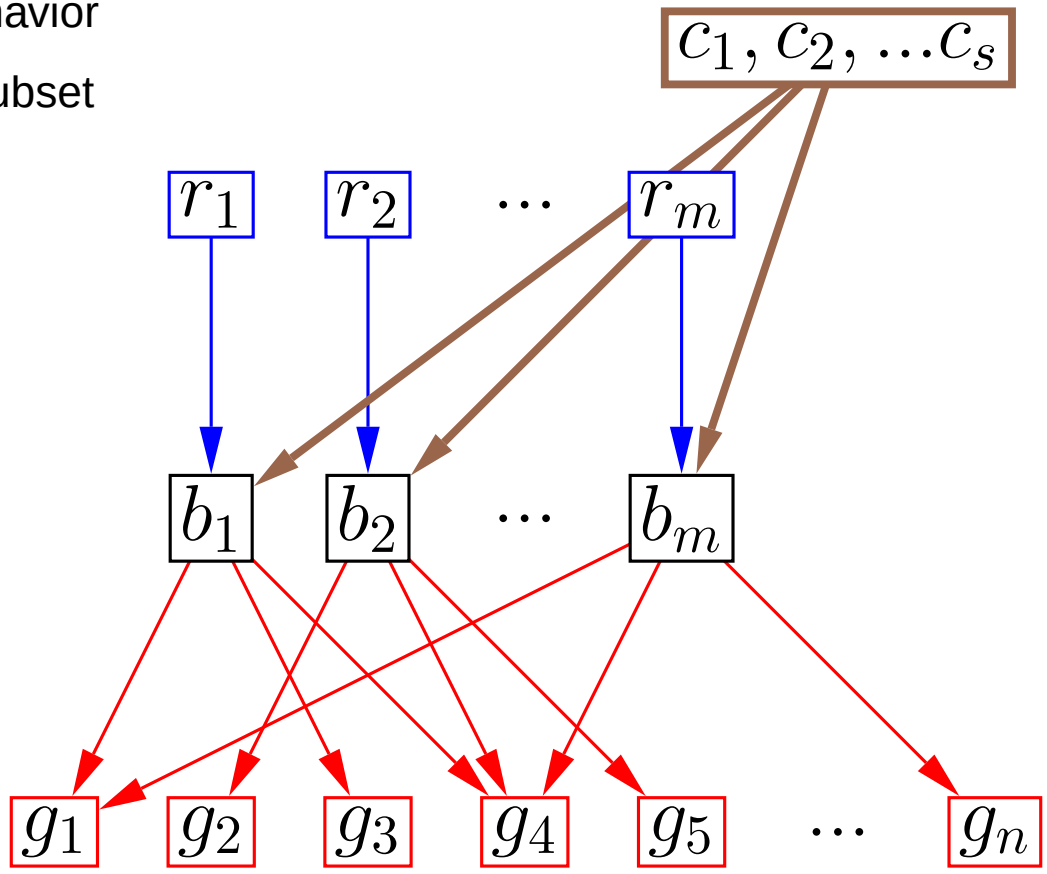


Model Template

- A hidden boolean variable, b, represents the behavior of the regulator (*i.e.* can this repressor stop transcription, can this activator enable transcription?)
- The value of b is determined by the presence of the regulator and the cellular conditions (which represent the presence or absence of relevant effectors)
- The regulated genes' expression is determined by the behavior of the regulator

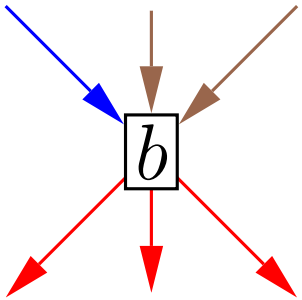
Full Model

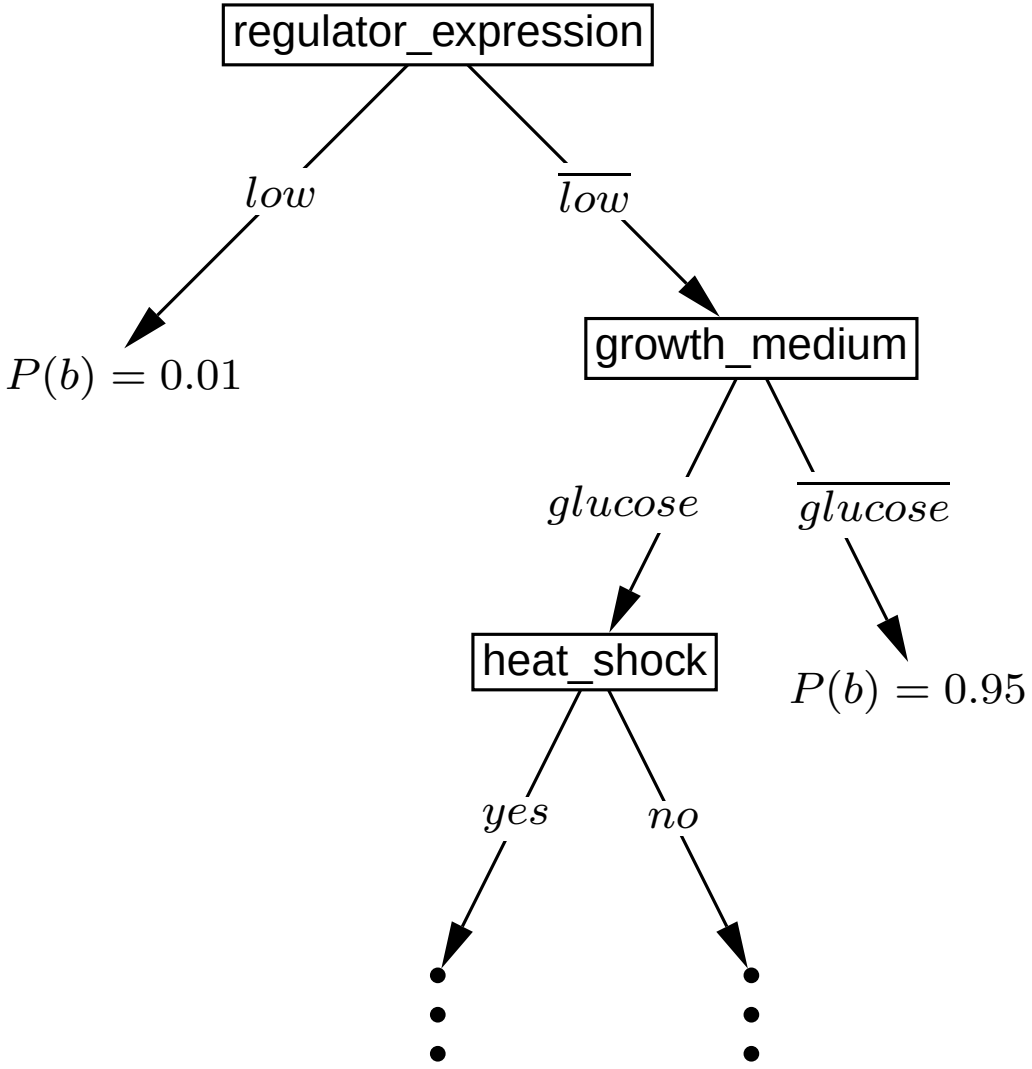
- Cellular conditions affect each regulator's behavior
- Each regulator affects a subset of all genes (known as a *regulon*)
- The network can be wired so that each gene is given a specific set of regulators, or it can learn this from the data

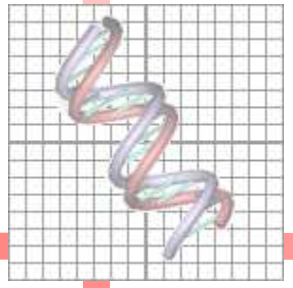


Probability Models for the Hidden Variables

- The probability of b is determined only by certain combinations of values among parent variables
- These trees are learned from the data





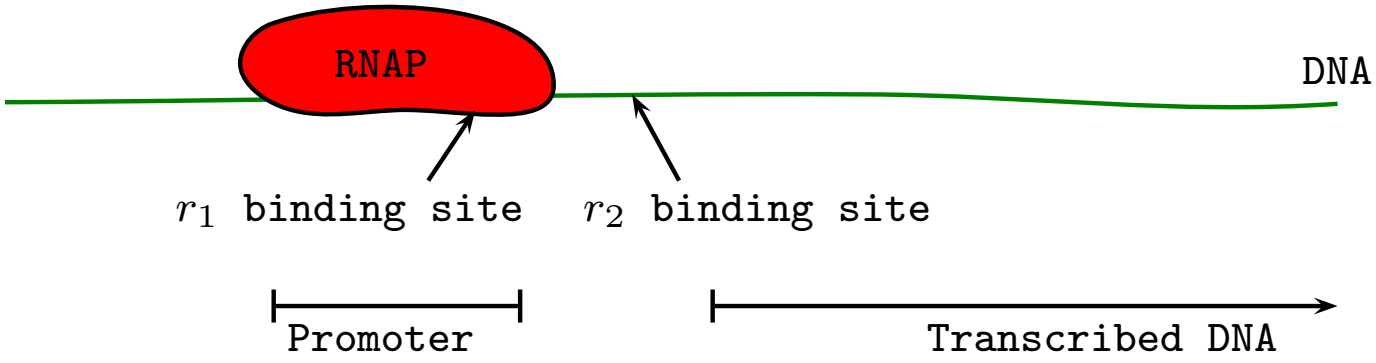


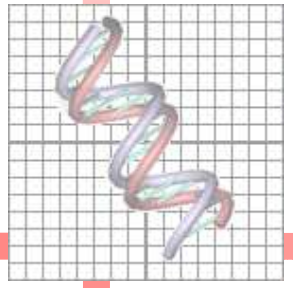
Using Predicted Binding Sites

If a regulator binds close to transcription start, we guess the regulator is a repressor and the probability of high expression is low if this regulator is acting on the gene; otherwise we guess the regulator is an activator and the probability is higher

b_1	b_2	$Pr(x = low)$	$Pr(x = average)$	$Pr(x = high)$
0	0	0.2	0.3	0.5
0	1	0.7	0.2	0.1
1	0	0.7	0.2	0.1
1	1	0.9	0.09	0.01

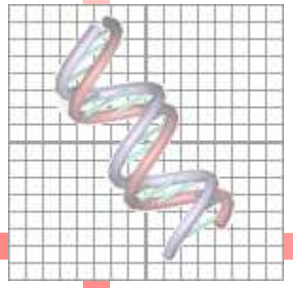
This may be a typical initial CPT for a gene with two regulators both guessed to be repressors (Note that this is just the initial value; after training on the data, these values change)





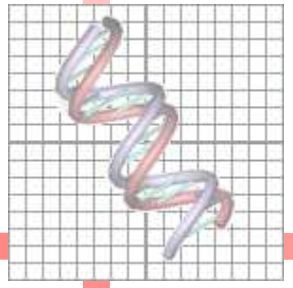
Learning the Trees

- The actual value of b is not present in the data
- Use an Expectation-Maximization algorithm to get a distribution that matches the data
  1. Set up the gene variables' CPTs with the prior probability distributions based on predicted binding sites
  2. E-Step: Estimate the values of b from the data and the current value of the genes' CPTs
  3. M-Step: Use these estimated values to update the genes' CPTs and build the tree
  4. Go back to step 2 and repeat until convergence



Features of This Model

- Effectors relevant to regulation are not all identified, and their presence cannot be measured. We represent them by the cellular conditions which cause them.
- Instead of simply representing the presence of gene products known or suspected to be regulators, our model represents whether or not they are acting as regulators.
- Instead of modeling each pairwise relationship between regulator and regulated gene separately, our model's regulator behavior must be consistent with all gene expression that it influences.
- We use known and predicted regulator binding sites to constrain search.

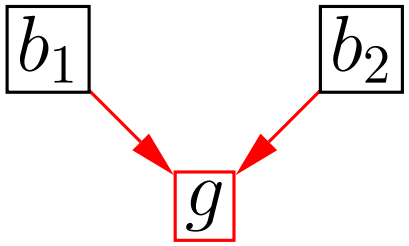


Experiments

- We test features of this model by running experiments with variations on the model.
- We use data from E. Coli gene expression from 18 microarrays
- All expression data is discretized into { low, average, high }
- We use six regulators and 76 total genes
- We use leave-one-out testing because there were so few microarrays
- We score the model on test data in two ways: (1) The accuracy measured as how often a gene's expression value is the most probable, and (2) The probability of all the test data given the model parameters
- As a baseline, we compare these with the results from simply using the frequency of the genes' expression values.

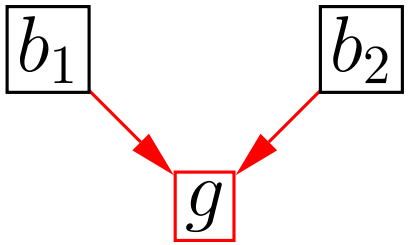
Experiment 1

Compare this model with one that does not use the regulators' binding sites to initialize the model parameters but instead just uses random values



b_1	b_2	$Pr(x = low)$	$Pr(x = average)$	$Pr(x = high)$
0	0	0.2	0.3	0.5
0	1	0.7	0.2	0.1
1	0	0.7	0.2	0.1
1	1	0.9	0.09	0.01

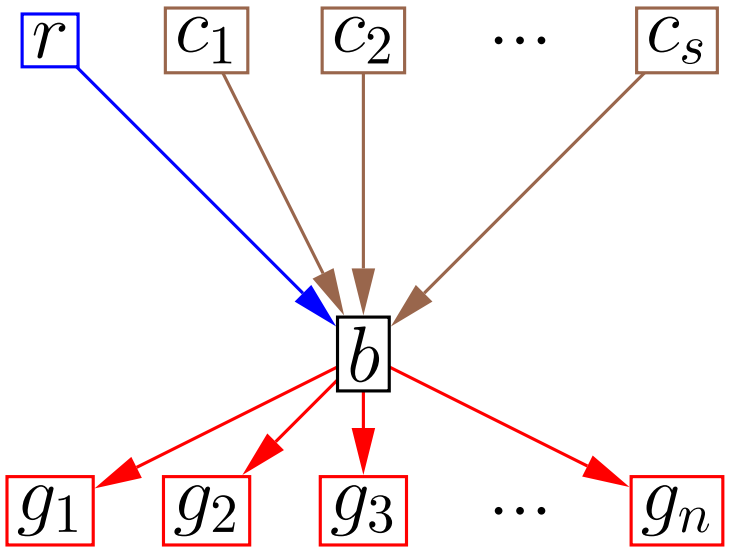
vs.



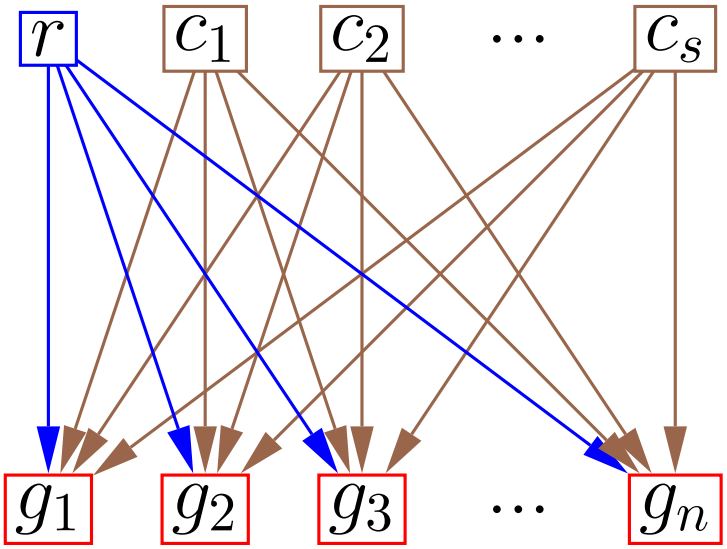
b_1	b_2	$Pr(x = low)$	$Pr(x = average)$	$Pr(x = high)$
0	0	0.420	0.501	0.079
0	1	0.118	0.452	0.430
1	0	0.088	0.821	0.091
1	1	0.195	0.179	0.626

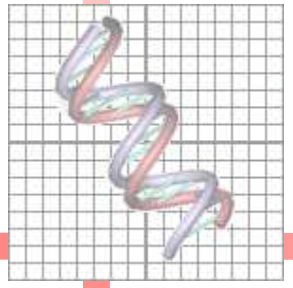
Experiment 2

Compare this model with one that does not use the hidden variable at all, and rather sets the genes' expression probability directly from the regulator expression and the cellular conditions



vs.

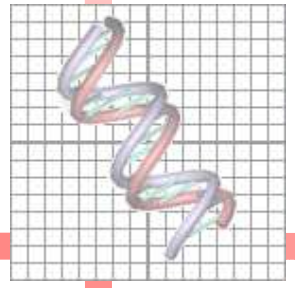




Results

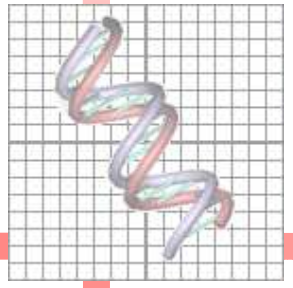
Description	Data Points	Correct (%)	$log(Pr(Data\|Model))$
Full Model	1368	998 (73%)	-900
Without Regulator Binding Sites	1368	977 (71%)	-908
Without Hidden Variable	1368	962 (70%)	-1910
Baseline	1368	546 (39%)	-946

- The overall probability of the model without the hidden variable is hurt by a lack of a good prior probability estimate (so the probability of a never-before-seen data point is very low)
- Without using the regulator binding sites to guide the E-M algorithm, the score is more variable than with this prior. Sometimes it settles on a good tree, sometimes not
- The full model seems to perform better on most of the test experiments, but much worse than the simple gene frequency distribution when testing on experiments that were significantly different from the training set.



Future Work

- Use a larger data set for testing this model which is more typical of a current microarray assay (we are currently organizing data from about 55 experiments and 110 microarrays)
- Learn the regulon of each regulator directly from the data
- Make more extensive use of the sequenced genome by predicting regulator binding sites or using possible binding sites to confirm new learned regulators



Acknowledgements

- Thanks to Mark Craven, Joseph Bockhorst, Fred Blattner and Yu Qiu for help with design on this model and with the data sets