Probabilistic Modeling: Bayesian Networks

Bioinformatics: Sequence Analysis

COMP 571 - Spring 2015
Luay Nakhleh, Rice University

Bayesian Networks

- Bayesian networks are probabilistic descriptions of the regulatory network.

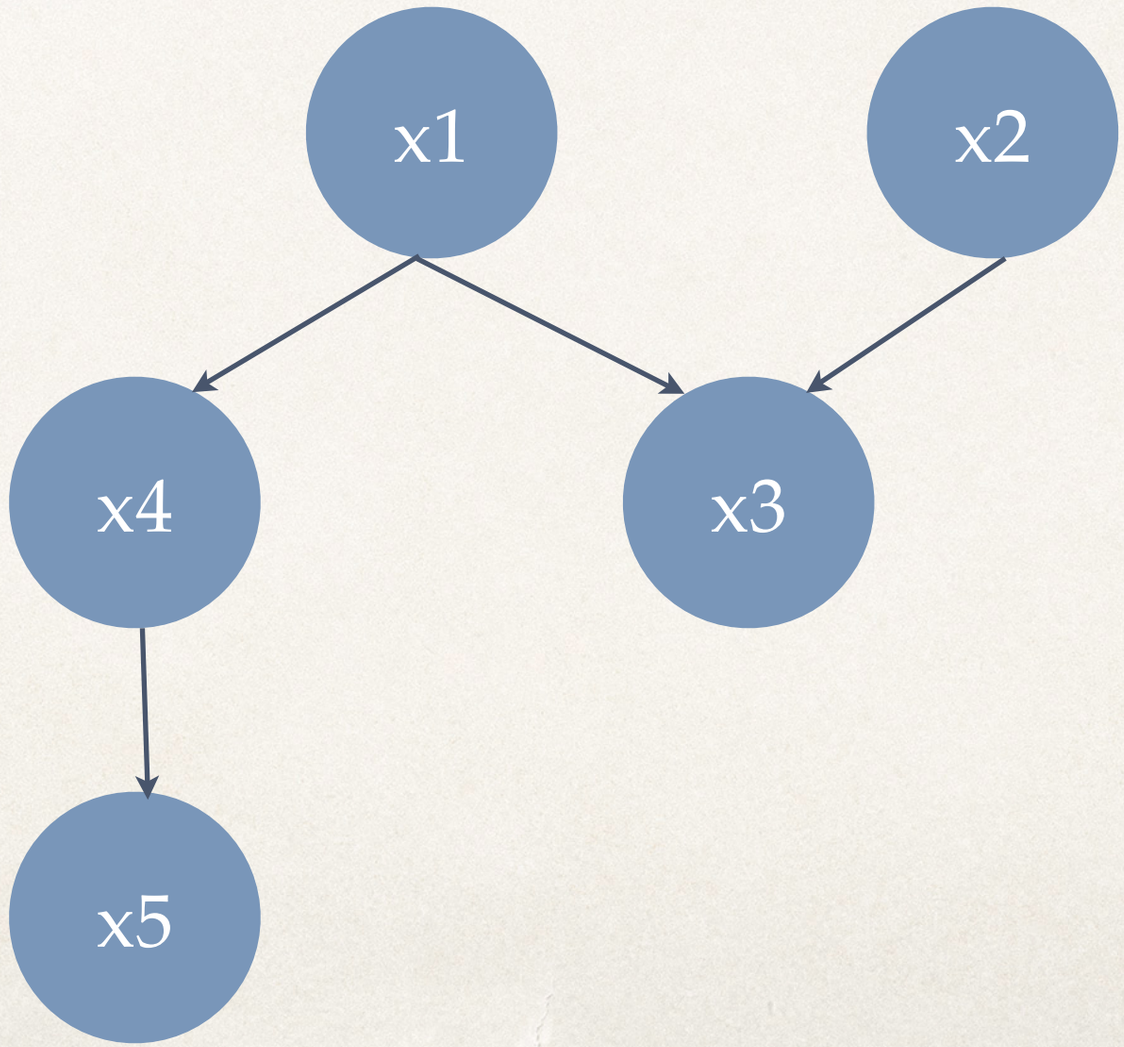
- A Bayesian network consists of (1) a directed, acyclic graph, G=(V,E), and (2) a set of probability distributions.

- The n vertices (n genes) correspond to random variables x_i, $1 \leq i \leq n$.

- For example, the random variables describe the gene expression level of the respective gene.

Bayesian Networks

- For each x_i, a conditional probability $p(x_i \mid L(x_i))$ is defined, where $L(x_i)$ denotes the parents of gene i, i.e., the set of genes that have a direct regulatory influence on gene i.



Bayesian Networks

- The set of random variables is completely determined by the joint probability distribution.

- Under the Markov assumption, i.e., the assumption that each x_i is conditionally independent of its non-descendants given its parents, this joint probability distribution can be determined by the factorization via

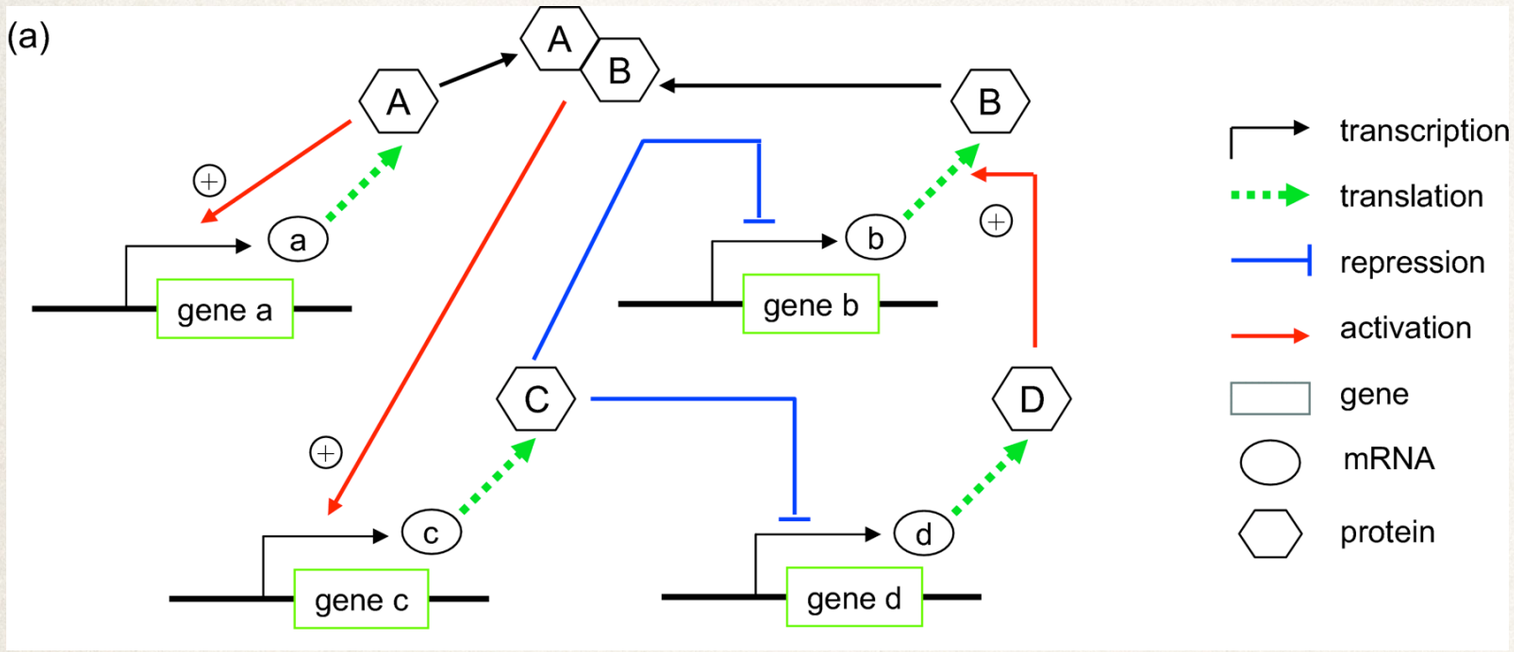
$$p(x) = \prod_{i=1}^{n} p(x_i | L(x_i))$$

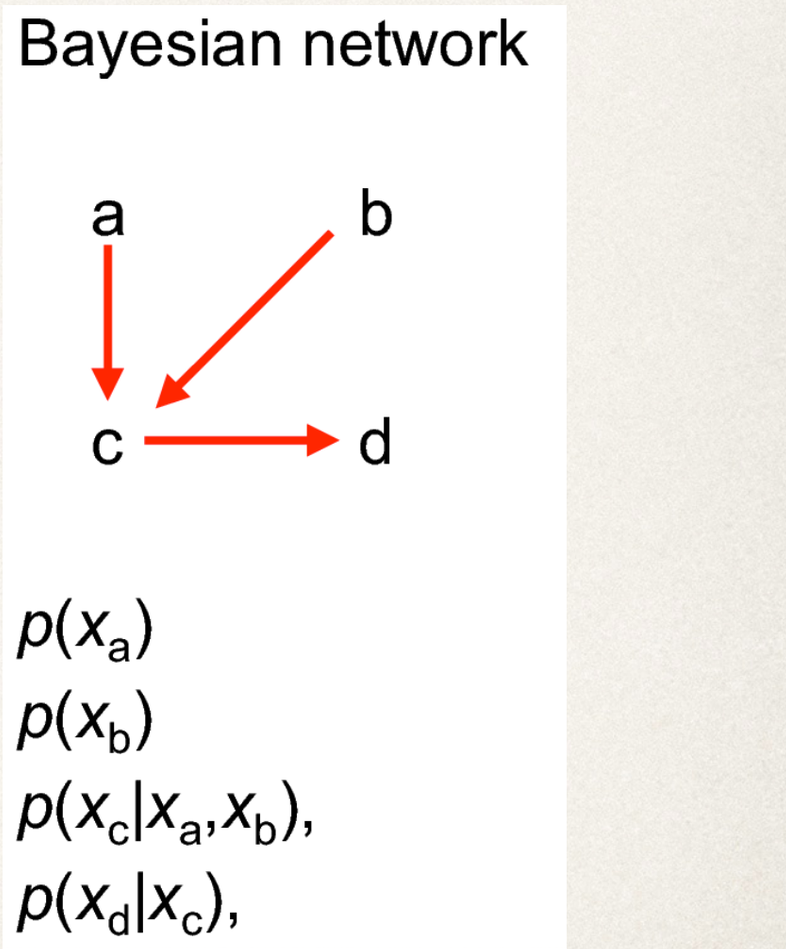
Bayesian Networks

- Conditional independence of two random variables x_i and x_j given a random variable x_k means that $p(x_i,x_j \mid x_k)=p(x_i \mid x_k)p(x_j,x_k)$, or, equivalently, $p(x_i \mid x_j,x_k)=p(x_i \mid x_k)$.

- The conditional distributions $p(x_i \mid L(x_i))$ are typically assumed to be linearly normally distributed, i.e., $p(x_i|L(x_i)) \sim N\left(\sum_k a_k x_k, \sigma^2\right)$, where x_k is in the parent set of x_i.

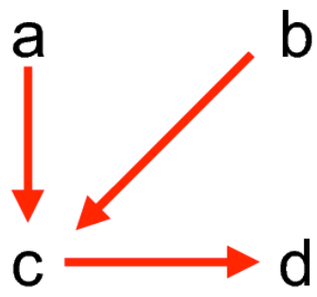
Bayesian Networks





Bayesian Networks

Bayesian network



$p(x_a)$
$p(x_b)$
$p(x_c|x_a,x_b)$,
$p(x_d|x_c)$,

a	b	P(c=1)
0	0	0.02
0	1	0.08
1	0	0.06
1	1	0.88

c	P(d=1)
0	0.03
1	0.92

Inputs: a,b
Outputs: d
Hidden: c

Bayesian Networks

- Given a network structure and a conditional probability table (CPT) for each node, we can calculate the output of the system by simply looking up the relevant input condition (row) in the CPT of the inputs, generating a “1” with the output probability specified for that condition, then using these newly generated node values to evaluate the outputs of nodes that receive inputs from these, and so on.

- We can also go backwards, asking what input activity patterns could be responsible for a particular observed output activity pattern.

Bayesian Networks

- To construct a Bayesian network, we need to estimate two sets of parameters:
  - the values of the CPT entries, and
  - the connectivity pattern, or structure (dependencies between variables)
  - The usual approach to learning both sets of parameters simultaneously is to first search for network structures, and evaluate the performance of each candidate network structure after estimating its optimum conditional probability values.

Learning the CPT entries

Bayesian Networks

- *Learning conditional probabilities from full data: Counting*

- If we have full data, i.e., for every combination of inputs to every node we have several measurements of node output value, then we can estimate the node output probabilities by simply counting the proportion of outputs at each level (e.g., on, off). These can be translated to CPTs, which together with the network structure fully define the Bayesian network.

Bayesian Networks

- *Learning conditional probabilities from full data: Counting*

- If we have full data, i.e., for every combination of inputs to every node we have several measurements of node output value, then we can estimate the node output probabilities by simply counting the proportion of outputs at each level (e.g., on, off). These can be translated to CPTs, which together with the network structure fully define the Bayesian network.

In practice, we don't have enough data for this to work.
Also, if we don't discretize the values, this is a problematic approach.

Bayesian Networks

- *Learning conditional probabilities from full data: Maximum Likelihood (ML)*

- Find the parameters that maximize the likelihood function given a set of observed training data D={$\mathbf{x}_1,\mathbf{x}_2,...,\mathbf{x}_N$}:$\theta\theta\theta$

$$\theta^* \leftarrow \text{argmax}_\theta L(\theta)$$

where

$$L(\theta) = p(D|\theta) = \prod_{i=1}^{N} p(\mathbf{x}_i|\theta)$$

Bayesian Networks

- *Learning conditional probabilities from full data: Maximum Likelihood (ML)*

- Find the parameters that maximize the likelihood function given a set of observed training data D={$\mathbf{x}_1,\mathbf{x}_2,...,\mathbf{x}_N$}:$\theta\theta\theta$

$$\theta^* \leftarrow \text{argmax}_\theta L(\theta)$$

where

$$L(\theta) = p(D|\theta) = \prod_{i=1}^{N} p(\mathbf{x}_i|\theta)$$

Does not assume any prior.

Bayesian Networks

- *Learning conditional probabilities from full data: Maximum a posteriori (MAP)*

- Compute$\theta\theta\theta$

$$\theta^* \leftarrow \text{argmax}_\theta \ln p(\theta|D)$$

- Through Bayes' theorem:

$$p(\theta|D) = \frac{p(D|\theta)p(\theta)}{p(D)}$$

Bayesian Networks

- Often, ML and MAP estimates are good enough for the application in hand, and produce good predictive models.

- Both ML and MAP produce a point estimate for θ.

- Point estimates are a single snapshot of parameters.

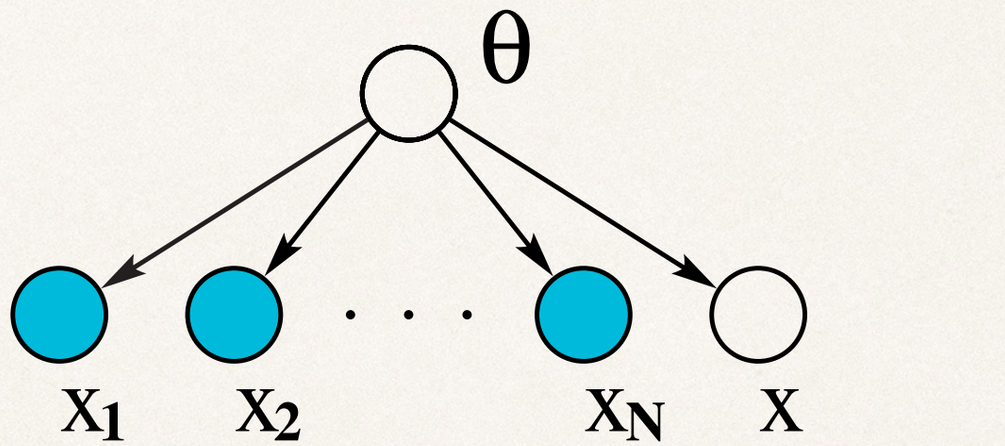
- A full Bayesian model captures the uncertainty in the values of the parameters by modeling this uncertainty as a probability distribution over the parameters.

Bayesian Networks

- *Learning conditional probabilities from full data: a full Bayesian model*
  - The parameters are considered to be latent variables, and the key idea is to marginalize over these unknown parameters, rather than to make point estimates (this is known as marginal likelihood).

Bayesian Networks

- *Learning conditional probabilities from full data: a full Bayesian model*



$$p(D, \theta, x) = p(x|\theta)p(D|\theta)p(\theta)$$

Bayesian Networks

- *Learning conditional probabilities from full data: a full Bayesian model*

$$p(D, \theta, x) = p(x|\theta)p(D|\theta)p(\theta)$$

Bayesian Networks

- *Learning conditional probabilities from full data: a full Bayesian model*

$$p(D, \theta, x) = p(x|\theta)p(D|\theta)p(\theta)$$

$$p(x, D) = \int p(D, \theta, x) d\theta$$

Bayesian Networks

- *Learning conditional probabilities from full data: a full Bayesian model*

$$p(D, \theta, x) = p(x|\theta)p(D|\theta)p(\theta)$$

$$\downarrow$$

$$p(x, D) = \int p(D, \theta, x)d\theta$$

$$\downarrow$$

$$p(x|D)p(D) = \int p(D, \theta, x)d\theta$$

Bayesian Networks

- *Learning conditional probabilities from full data: a full Bayesian model*

$$p(D, \theta, x) = p(x|\theta)p(D|\theta)p(\theta)$$

$$\downarrow$$

$$p(x, D) = \int p(D, \theta, x)d\theta$$

$$\downarrow$$

$$p(x|D)p(D) = \int p(D, \theta, x)d\theta$$

$$\downarrow$$

$$p(x|D) = \frac{1}{p(D)} \int p(x|\theta)p(D|\theta)p(\theta)d\theta = \int p(x|\theta)p(\theta|D)d\theta$$

Bayesian Networks

- *Learning conditional probabilities from full data: a full Bayesian model*

- A prior distribution, $p(\theta)$, for the model parameters needs to be specified.

- There are many types of priors that may be used, and there is much debate about the choice of prior.

- Often, the calculation of the full posterior is intractable, and approximate methods must be used.

Bayesian Networks

- *Learning conditional probabilities from incomplete data*
- If we do not have data for all possible combinations of inputs to every node, or when some individual data values are missing, we start by giving all missing CPT values equal probabilities. Next, we use an optimization algorithm (Expectation Maximization, Markov Chain Monte Carlo search, etc.) to curve-fit the missing numbers to the available data. When we find parameters that improve the network's overall performance, we can replace the previous "guess" with the new values and repeat the process.

Bayesian Networks

- *Learning conditional probabilities from incomplete data*

- Another case of incomplete data pertains to hidden nodes: no data is available for certain nodes in the network.

- A solution is to iterate over plausible network structures, and to use a “goodness” score to identify the Bayesian network.

Structure Learning

- In biology, the inference of network structures is the most interesting aspect.

- This involves identifying real dependencies between measured variables, and distinguishing them from simple correlations.

- The learning of model structures, and particularly causal models, is difficult, and often requires careful experimental design, but can lead to the learning of unknown relationships and excellent predictive models.

DREAM is a Dialogue for Reverse Engineering Assessments and Methods. The main objective is to catalyze the interaction between experiment and theory in the area of cellular network inference and quantitative model building in systems biology.

DREAM8, 2013

1. HPN-DREAM Breast Cancer Network Inference Challenge - Participants in this Challenge will be provided with an extensive proteomics time-course dataset on four breast cancer cell lines and tasked with analyzing these data to solve the following 3 sub-challenges: 1) build network models that represent the active cell signaling pathways in breast cancer, 2) predict the dynamic response of various phospho-proteins to drug perturbations, and 3) propose novel strategies to visualize these high dimensional data.
2. NIEHS-NCATS-UNC DREAM Toxicogenetics Challenge - We will provide genetics and transcriptomics information of the 1000 Genomes Project (www.1000genomes.org), as well as cytotoxicity measures derived from compound exposure to over a hundred toxic agents using the 1000 genomes lymphoblastoid cell lines. Participants are tasked with solving two related subchallenges: (1) develop predictive models of cytotoxicity using genetic and genomic data to predict individual responses to compound exposure and (2) use chemical attributes to predict population-based cytotoxicity characteristics (median, variance) for a set of compounds.
3. The Whole-Cell Parameter Estimation DREAM Challenge - Participants will be provided with a whole cell model of Mycoplasma genitalium and tasked with estimating the model parameters for specific biological processes from simulated data. The simulated data to be provided represents possible measurements in actual experiments: participants will will have a credit budget and will be able to purchase simulated data on demand with the aim to refine the parameters under estimation.

- The marginal likelihood over structure hypotheses S as well as model parameters:

$$p(x|D) = \sum_S p(S|D) \int p(x|\theta_S, S) p(\theta_S|D, S) d\theta_S$$

Intractable, but for very small networks!

- Markov chain Monte Carlo (MCMC) methods, for example, can be used to obtain a set of "good" sample networks from the posterior distribution $p(S,\theta \mid D)$.

- This is particularly useful in biology, where D may be sparse and the posterior distribution diffuse, and therefore much better represented as averaged over a set of model structures than through choosing a single model structure.

Structure Learning Algorithms

- The two key components of a structure learning algorithm are:
  - searching for a good structure, and
  - scoring these structures

Structure Learning Algorithms

- While the scoring can be done using, for example, the marginal likelihood, the space of possible structures can be searched using, for example, greedy search, simulated annealing, etc.

- In biology, one can use biological knowledge, such as protein-protein interaction data, binding site information, existing literature, etc., to effectively limit the number of structures considered to be the most biologically relevant.

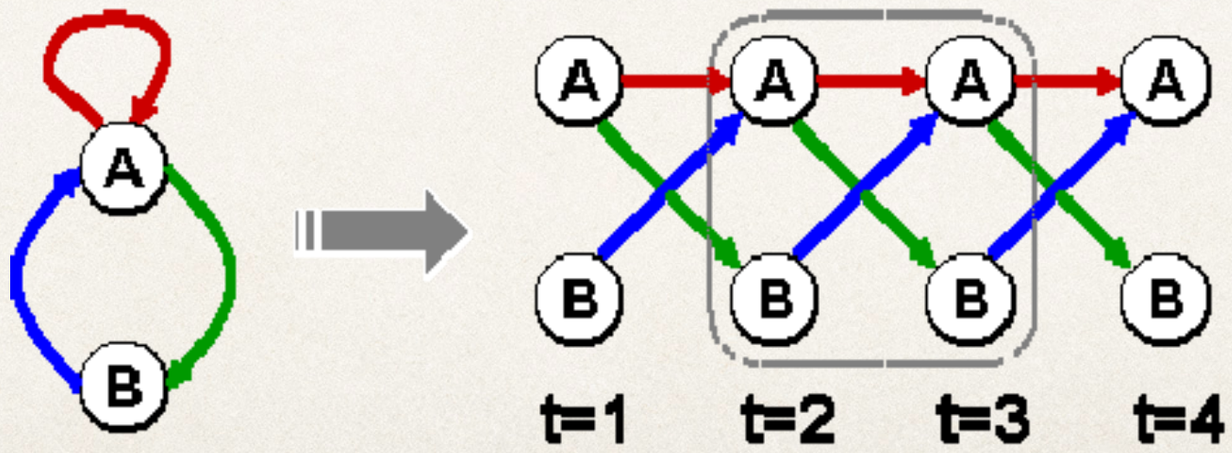
Dynamic Bayesian Networks

- An essential feature of many biological systems is feedback.

- Feedback loops in a graphical representation of dependencies between different quantities correspond to networks that are <u>not</u> Bayesian networks, according to the definition above.

Dynamic Bayesian Networks

- To specify the behavior of networks with feedback, we need to introduce an explicit concept of time and "unfold" the edges in a new time direction.



Probabilistic Graphical Models

- Bayesian networks belong to a large class of models known as probabilistic graphical models.

- This includes: probabilistic Boolean networks, Gaussian network models, factor graphs,...

- Inferring, analyzing, and extending these models is a very active area of research today.

Probabilistic Graphical Models