Causal Bayesian Networks

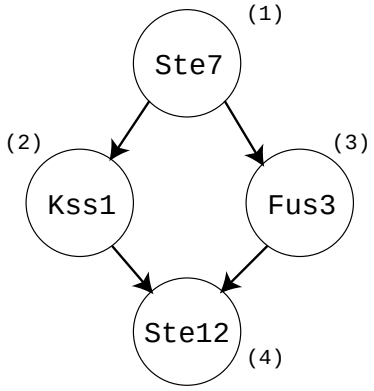


Figure 1: Simple Example

While Bayesian networks should typically be viewed as acausal, it is possible to impose a causal interpretation on these models with additional care. What we get as a result is a probabilistic extension of the qualitative causal models. The extension is useful since the qualitative models discussed last time are a bit limited in their ability to quantify interactions, e.g., that Ste7 may only have a certain probability of activating Kss1.

The modification necessary for maintaining a causal interpretation is exactly analogous to the qualitative models. Suppose we intervene and set the value of one variable, say we knock-out Kss1. Then the mechanism through which Kss1 is activated by Ste7 can no longer be used for inference (it wasn't responsible for the value set in the intervention). Graphically, this just means deleting all the arrows to the variable(s) set in the intervention.

More formally, the probability model associated with the graph in the above figure factors according to

$$P(x_1)P(x_2|x_1)P(x_3|x_1)P(x_4|x_2,x_3) \tag{1}$$

If we now set $\texttt{set}(x_2 = -1)$, i.e., knock out Kss1, then the probability model over the remaining variables is simply

$$P(x_1)P(x_3|x_1)P(x_4|x_2 = -1, x_3) \tag{2}$$

Note that the parameters in the conditional tables are exactly as before; the only difference is that one of the conditionals is missing.

The estimation of Bayesian network proceeds otherwise as before. For example, suppose we have two observed expression patterns, one arising from a causal intervention (knock-out), the other one not.

	x_1	x_2	x_3	x_4
D_1	1	1	1	1
$D_2(\ set(x_2 = -1)\ )$	0	-1	0	0

To estimate the parameters we just write down the log-likelihood of the observed data while taking into account that the model may have to be modified slightly when incorporating causal measurements

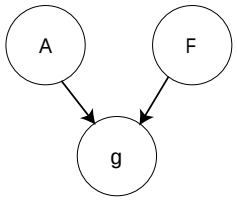
$$\begin{aligned}
&\log P(D1) + \log P(D2) = \\
&\quad \log P(x_1 = 1) + \log P(x_2 = 1|x_1 = 1) \\
&\quad + \log P(x_3 = 1|x_1 = 1) + \log P(x_4 = 1|x_2 = 1, x_3 = 1) \\
&\quad + \log P(x_1 = 0) + \log P(x_3 = 0|x_1 = 0) + \log P(x_4 = 0|x_2 = -1, x_3 = 0)
\end{aligned} \tag{3}$$

We try to maximize this log-likelihood relative to the parameters in the conditional probabilities (tables). The maximum likelihood setting of the parameters reduces to observed frequencies such as

x_1	$\hat{P}(x_1)$
-1	0
0	1/2
1	1/2

Decision tree models

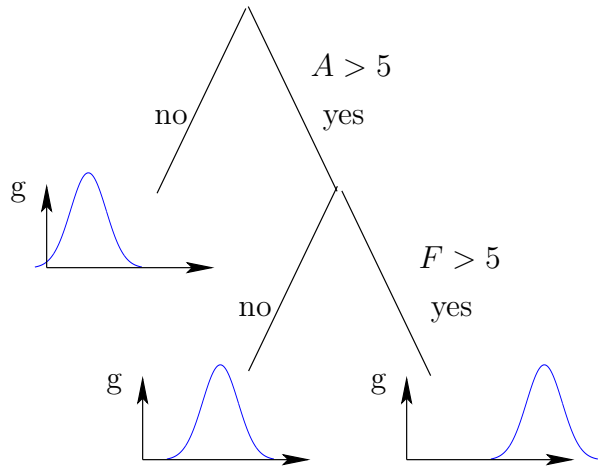
Another issue with the Bayesian networks is that the regulatory program represented by the network is not explicit in terms of the interactions involved but rather buried in the parameters. For example:



where A represents the acetylation level in the neighborhood of gene g and F is a known regulator. We cannot infer from the graph how acetylation of the histone molecules might interact with the transcription factor to regulate the downstream genes. A more explicit description might look like:

```
if A > 5 (enrichment) then
   if F>5 (enrichment) then
      high transcription (Gaussian with a large mean)
   otherwise
      moderate transcription
otherwise
   no transcription
```

This is a decision tree



that aims to capture the distribution over transcript levels conditionally on A and F.

Why is this more interpretable? Because we understand rules more easily than dependencies. A decision tree representation of the interaction between the parents (causes) and the gene is not always compact, however. For example, we need a large decision tree to represet a simple linear interaction provided that the decisions (branching) is based only on individual variables.

How can we estimate decision tree models? First, we find the variable with the largest effect on transcription (e.g., acetylation) and specify it as the root decision (as above).

Then we expand each branch similarly according to some estimation criterion such as gain in the likelihood of the predictions. Expanding the tree endlessly is not helpful; more leaves you have, the less data you have to justify that the branch should be expanded. In other words, you can simply run out of data by expanding the tree.

Automated Experiment Design (active learning)

Estimating complex models from data typically requires more data than we have. One way to minimize the amount of data needed either to estimate a model or distinguish between competing models is by optimizing the selection of experiments to carry out. We consider here automating the selection process.

Automated experiment design or active learning can be used in many biological contexts, including probe design, which factor to profile, which gene to knockout, and so on. We will focus here for simplicity on gene deletions to discriminate among qualitative causal models discussed earlier. A similar approach can be used with (causal) Bayesian networks as well.

The models we consider are

$$\begin{array}{llll} M_0 & F & & G \\ M_1 & F & \overset{+}{\rightarrow} & G \\ M_2 & F & \overset{-}{\rightarrow} & G \\ M_3 & F & \overset{+}{\leftarrow} & G \\ M_4 & F & \overset{-}{\leftarrow} & G \end{array}$$

There are four key problems we have to solve

1. represent model uncertainty, i.e., $P(M_i)$
2. predict possible outcomes for each pair of model and experiment, e.g., $P(g|M, \texttt{set}(F = -1))$
3. evaluate posterior probability over the models for each possible experiment and (hypothetical) outcome, e.g., compute $P(M|g, \texttt{set}(F = -1))$
4. define the (information) "gain" from carrying out each possible experiment (deleting g or F in this case)

We will solve these problems in turn when the two possible experiments are gene deletions: $\texttt{set}(F = -1)$ or $\texttt{set}(g = -1)$.

Problem 1

As before, it is simply to define a distribution over the possible few models. This is obviously somewhat harder when we have a large number of possible models (need to decompose the probability into a probability of each model feature)

$$\begin{array}{rcccc} P(M_0) & = & \frac{1}{8} & F & \quad G \\ P(M_1) & = & \frac{1}{4} & F & \stackrel{+}{\rightarrow} G \\ P(M_2) & = & \frac{1}{8} & F & \stackrel{-}{\rightarrow} G \\ P(M_3) & = & \frac{1}{4} & F & \stackrel{+}{\leftarrow} G \\ P(M_4) & = & \frac{1}{4} & F & \stackrel{-}{\leftarrow} G \end{array}$$

Can you guess which deletion we should carry out to maximally reduce the space of possible models?

Problem 2

We have already discussed how to evaluate these probabilities. For example:

$$\begin{aligned} & P(g = 0|\texttt{set}(F = -1)) \\ & \quad = P(M_0)P(g = 0|M_0, \texttt{set}(F = -1)) + \ldots + P(M_4)P(g = 0|M_4, \texttt{set}(F = -1)) \end{aligned}$$

We get two tables corresponding to the two possible experiments:

$$\begin{array}{rcccc} g & : & -1 & 0 & 1 \\ P(g|\texttt{set}(F = -1)) & : & \frac{1}{4} & \frac{5}{8} & \frac{1}{8} \end{array}$$

$$\begin{array}{rcccc} g & : & -1 & 0 & 1 \\ P(F|\texttt{set}(g = -1)) & : & \frac{1}{4} & \frac{1}{2} & \frac{1}{4} \end{array}$$

Problem 3

We have to evaluate

$$P(M_i|\texttt{set}(F=-1),g) = \frac{P(M_i)P(g|\texttt{set}(F=-1),M_i)}{P(g|\texttt{set}(F=-1))} \tag{4}$$

for all experiments (here $\texttt{set}(F=-1)$) and all outcomes (here $g=-1,0,1$)

Problem 4

We define the gain as the reduction of model uncertainty (information gain): "uncertainty before" - "uncertainty after" as in

$$\text{Gain}(\texttt{set}(F=-1)) = \text{entropy}(M) - \sum_{g\in\{-1,0,1\}} P(g|\texttt{set}(F=-1))\cdot\text{entropy}(M|\texttt{set}(F=-1),g) \tag{5}$$

where

$$\text{entropy}(M) = -\sum_{i=0}^{4} P(M_i)\log_2 P(M_i) \tag{6}$$

(this is the expected number of yes/no questions we would need to ask to guess the identity of a randomly selected model from $P(M_i)$)

and

$$\text{entropy}(M|\texttt{set}(F=-1),g) = -\sum_{i=0}^{4} P(M_i|\texttt{set}(F=-1),g)\log_2 P(M_i|\texttt{set}(F=-1),g) \tag{7}$$

In our case we get

$$\begin{aligned} \text{Gain}(\texttt{set}(F=-1)) &= 1.3 \text{ bits} \\ \text{Gain}(\texttt{set}(g=-1)) &= 1.5 \text{ bits} \end{aligned}$$

Note:

There are many other ways of defining the gain from carrying out a specific experiment. The definition should depend on what you want to get out of carrying out the experiment. For example, suppose we are only interested in finding the most likely model and do not care whether we can rank the remaining models. In this case we could define the gain as

$$Gain(\texttt{set}(F=-1)) = \sum_{g\in\{-1,0,1\}} P(g|\texttt{set}(F=-1))[\max_j P(M_j|\texttt{set}(F=-1),g) - \text{ next best}] \tag{8}$$

and we would get (by plugging in the numbers)

$$\begin{aligned} \text{Gain}(\texttt{set}(F=-1)) &= 0.375 \\ \text{Gain}(\texttt{set}(g=-1)) &= 0.625 \end{aligned}$$

This criterion would also select $\texttt{set}(g=-1)$. In general these don't have to end up selecting the same experiment.