Building Bayesian Networks

COMPSCI 276, Fall 2014

Set 4: Rina Dechter

(Reading: Darwiche chapter 5)
Outline

• Bayesian networks and queries
• Building Bayesian Networks
• Special representations of CPTs
The construction of a Bayesian network involves three major steps:

- Identify relevant variables and their possible values.
- Build the network structure by connecting variables into DAG.
- Define the CPT for each network variable.

Two issues:

- The potentially large size of CPTs.
- The significance of the specific numbers used to populate them.

We present techniques for dealing with these issues.

**Queries:** Different queries may be relevant for different scenarios
The network Asia will be used as a running example. Screenshot from Samlam.

http://reasoning.cs.ucla.edu/samiam


For other tools see class page.
Other type of evidence: We may want to know the probability that the patient has either a positive X-ray or dyspnoea, $X = \text{yes}$ or $D = \text{yes}$.

Probability of some variable instantiation $e$, $Pr(e)$.

Probability that the patient has a positive X-ray, but no dyspnoea, $Pr(X = \text{yes}, D = \text{no})$, about 3.96%. Computed by Samlam.

The variables $E = \{X, D\}$ are called evidence variables. The query $Pr(e)$ is known as a probability-of-evidence.

Other type of evidence: We may want to know the probability that the patient has either a positive X-ray or dyspnoea, $X = \text{yes}$ or $D = \text{yes}$. 
**Auxiliary-node method**

Bayesian network tools do not usually provide direct support for computing the probability of arbitrary pieces of evidence, but such probabilities can be computed indirectly.

We can add an auxiliary node $E$, declare nodes $X$ and $D$ as the parents of $E$, and use the following CPT for $E$:

| $X$ | $D$ | $E$ | $\Pr(e|x, d)$ |
|-----|-----|-----|---------------|
| yes | yes | yes | 1             |
| yes | no  | yes | 1             |
| no  | yes | yes | 1             |
| no  | no  | yes | 0             |

Event $E = \text{yes}$ is then equivalent to $X = \text{yes} \vee D = \text{yes}$. 
Query: Prior and Posterior Marginals

**Prior Marginals**

Given a joint probability distribution $\Pr(x_1, \ldots, x_n)$, the *marginal distribution* $\Pr(x_1, \ldots, x_m)$, $m \leq n$, is defined as follows:

$$\Pr(x_1, \ldots, x_m) = \sum_{x_{m+1}, \ldots, x_n} \Pr(x_1, \ldots, x_n).$$

The marginal distribution can be viewed as a projection of the joint distribution on the smaller set of variables $X_1, \ldots, X_m$.

**Posterior marginal given evidence $e$**

$$\Pr(x_1, \ldots, x_m|e) = \sum_{x_{m+1}, \ldots, x_n} \Pr(x_1, \ldots, x_n|e).$$
Prior Marginals in the Asia Network

C = lung cancer

<table>
<thead>
<tr>
<th></th>
<th>Pr(C)</th>
</tr>
</thead>
<tbody>
<tr>
<td>yes</td>
<td>5.50%</td>
</tr>
<tr>
<td>no</td>
<td>94.50%</td>
</tr>
</tbody>
</table>
Query: Posterior Marginals in the Asia Network

| C       | Pr(C|e) |
|---------|--------|
| yes     | 25.23% |
| no      | 74.77% |

\( e : X = \text{yes}, D = \text{no} \)
Define a CPT for $V$ that satisfies this constraint

$$\frac{P(V=\text{yes}|E=\text{yes})}{P(V=\text{yes}|E=\text{no})} = 2$$

Soft evidence on $E$ as hard evidence on auxiliary variable $V$. 

Soft Evidence using Virtual Evidence (Noisy Sensor)
Query: Most Probable Explanation (MPE)

Let $X_1, \ldots, X_n$ be all network variables, and $e$ be evidence. Identify an instantiation $x_1, \ldots, x_n$ that maximizes the probability $\Pr(x_1, \ldots, x_n | e)$. Instantiation $x_1, \ldots, x_n$ is called a most probable explanation given evidence $e$.

MPE cannot be obtained directly from posterior marginals.

If $x_1, \ldots, x_n$ is an instantiation obtained by choosing each value $x_i$ so as to maximize the probability $\Pr(x_i | e)$, then $x_1, \ldots, x_n$ is not necessarily an MPE.

MPE is also called MAP
MPE given a positive X-ray and dyspnoea

A patient that made no visit to Asia; is a smoker; has lung cancer and bronchitis; but no tuberculosis.
MPE given a positive X-ray and no dyspnoea ($\approx 38.57\%$)

A patient that made no visit to Asia; is not a smoker; has no lung cancer, no bronchitis and no tuberculosis.

Choosing values with maximal probability, we get:

$\alpha$: $A = \text{no}$, $S = \text{yes}$, $T = \text{no}$, $C = \text{no}$, $B = \text{no}$, $P = \text{no}$, $X = \text{yes}$, $D = \text{no}$.

Probability $\approx 20.03\%$ given evidence $e$: $X = \text{yes}$, $D = \text{no}$. 
Query: Maximum a Posteriori Hypothesis (MAP)

MAP variables $M = \{A, S\}$ and evidence $e : X = yes, D = no$

MAP is $A = no, S = yes$.

MAP has probability of $\approx 50.74\%$ given the evidence.

MAP is also called Marginal Map
A common method for approximating MAP is to compute an MPE and then return the values it assigns to MAP variables. We say in this case that we are projecting the MPE on MAP variables.
Query: Maximum a Posteriori Hypothesis (MAP)

A common method for approximating MAP is to compute an MPE and then return the values it assigns to MAP variables. We say in this case that we are projecting the MPE on MAP variables.

Example

MPE given evidence $X = \text{yes}, \ D = \text{no}$:

$A = \text{no}, \ S = \text{no}, \ T = \text{no}, \ C = \text{no}, \ B = \text{no}, \ P = \text{no}, \ X = \text{yes}, \ D = \text{no}$

Projecting this MPE on MAP variables $\mathbf{M} = \{A, S\}$, we get:

$A = \text{no}, \ S = \text{no},$

with probability $\approx 48.09\%$ given the evidence.

MAP is $A = \text{no}, \ S = \text{yes}$ with a probability of about $50.74\%$. 
Bayesian networks will be constructed in three consecutive steps.

**Step 1**

Define the network variables and their values.

- A query variable is one which we need to ask questions about, such as compute its posterior marginal.
- An evidence variable is one which we may need to assert evidence about.
- An intermediary variable is neither query nor evidence and is meant to aid the modeling process by detailing the relationship between evidence and query variables.

The distinction between query, evidence and intermediary variables is not a property of the Bayesian network, but of the task at hand.
Bayesian networks will be constructed in three consecutive steps.

**Step 2**

Define the network structure (edges).

We will be guided by a causal interpretation of network structure.

The determination of network structure will be reduced to answering the following question about each network variable $X$: what set of variables we regard as the direct causes of $X$?

What about the boundary strata?
Modeling with Bayesian Networks

Step 3
Define the network CPTs.

- CPTs can sometimes be determined completely from the problem statement by objective considerations.
- CPTs can be a reflection of subjective beliefs.
- CPTs can be estimated from data.
Diagnosis I: Model from Expert

Example

The flu is an acute disease characterized by fever, body aches and pains, and can be associated with chilling and a sore throat. The cold is a bodily disorder popularly associated with chilling and can cause a sore throat. Tonsillitis is inflammation of the tonsils which leads to a sore throat and can be associated with fever.

Our goal here is to develop a Bayesian network to capture this knowledge and then use it to diagnose the condition of a patient suffering from some of the symptoms mentioned above.

Variables? Arcs? Try it.
A naive Bayes structure has the following edges $C \rightarrow A_1, \ldots, C \rightarrow A_m$, where $C$ is called the class variable and $A_1; \ldots; A_m$ are called the attributes.

Variables are binary: values are either true or false. More refined information may suggest different degrees of body ache.
The naive Bayes structure commits to the **single-fault** assumption.

Suppose the patient is known to have a cold.

**Naive Bayes structure**

Fever and sore throat become independent as they are d-separated by “Condition”.

**Original structure**

Fever may increase our belief in tonsillitis, which could then increase our belief in a sore throat.
If the only evidence we have is body ache, we expect the probability of flu to go up in both networks.

**Naive Bayes structure**
This leads to dropping the probability of cold or tonsillitis.

**Original structure**
These probabilities remain the same since both cold and tonsillitis are d-separated from body ache.
CPTs can be obtained from medical experts, who supply this information based on known medical statistics or subjective beliefs gained through practical experience.

CPTs can also be estimated from medical records of previous patients

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>true</td>
<td>false</td>
<td>?</td>
<td>true</td>
<td>false</td>
<td>false</td>
<td>false</td>
</tr>
<tr>
<td>2</td>
<td>false</td>
<td>true</td>
<td>false</td>
<td>true</td>
<td>true</td>
<td>false</td>
<td>true</td>
</tr>
<tr>
<td>3</td>
<td>?</td>
<td>?</td>
<td>true</td>
<td>false</td>
<td>?</td>
<td>true</td>
<td>false</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

? indicates the unavailability of corresponding data for that patient.
Tools for Bayesian network inference can generate a network parameterization $\Theta$, which tries to maximize the probability of seeing the given cases.

If each case is represented by event $d_i$, such tools will generate a parametrization $\Theta$ which leads to a probability distribution $\Pr$ that attempts to maximize:

$$
\prod_{i=1}^{N} \Pr(d_i).
$$

Term $\Pr(d_i)$ represents the probability of seeing the case $i$.

The product represents the probability of seeing all $N$ cases (assuming the cases are independent).
A few weeks after inseminating a cow, we have three possible tests to confirm pregnancy. The first is a scanning test which has a false positive of 1% and a false negative of 10%. The second is a blood test, which detects progesterone with a false positive of 10% and a false negative of 30%. The third test is a urine test, which also detects progesterone with a false positive of 10% and a false negative of 20%. The probability of a detectable progesterone level is 90% given pregnancy, and 1% given no pregnancy. The probability that insemination will impregnate a cow is 87%.

Our task here is to build a Bayesian network and use it to compute the probability of pregnancy given the results of some of these pregnancy tests.
Diagnosis II: Model from Expert

Pregnant? ($P$)

- Progesterone Level ($L$)
  - Urine Test ($U$)
  - Blood Test ($B$)
- Scanning Test ($S$)

<table>
<thead>
<tr>
<th>$P$</th>
<th>$\theta_P$</th>
</tr>
</thead>
<tbody>
<tr>
<td>yes</td>
<td>.87</td>
</tr>
</tbody>
</table>

| $P$ | $S$   | $\theta_{S|P}$ |
|-----|-------|----------------|
| yes | -ve   | .10            |
| no  | +ve   | .01            |

| $P$ | $L$                      | $\theta_{L|P}$ |
|-----|--------------------------|----------------|
| yes | undetectable             | .10            |
| no  | detectable               | .01            |

| $L$            | $B$ | $\theta_{B|L}$ |
|----------------|-----|----------------|
| detectable     | -ve | .30            |
| undetectable   | +ve | .10            |

| $L$ | $U$ | $\theta_{U|L}$ |
|-----|-----|----------------|
| detectable | -ve | .20            |
| undetectable | +ve | .10            |
Diagnosis II: Model from Expert

Example

We inseminate a cow, wait for a few weeks, and then perform the three tests which all come out negative:

\[ e: S = \neg ve, B = \neg ve, U = \neg ve. \]

Posterior marginal for pregnancy given this evidence:

\[
\begin{array}{|c|c|}
\hline
P & \Pr(P|e) \\
\hline
\text{yes} & 10.21\% \\
\text{no} & 89.79\% \\
\hline
\end{array}
\]

Probability of pregnancy is reduced from 87% to 10.21%, but still relatively high given that all three tests came out negative.
Example

A farmer is not too happy with this and would like three negative tests to drop the probability of pregnancy to no more than 5%. The farmer is willing to replace the test kits for this purpose, but needs to know the false positive and negative rates of the new tests, which would ensure the above constraint.

This is a problem of sensitivity analysis in which we try to understand the relationship between the parameters of a Bayesian network and the conclusions drawn based on the network.

Read in the book.
We will not cover this.
Try it: Variables? Values? Structure?

Problem statement
Given some values for the circuit primary inputs and output (test vector), decide if the circuit is behaving normally. If not, find the most likely health states of its components.
**Diagnosis III: Model from Design**

**Problem statement**
Given some values for the circuit primary inputs and output (test vector), decide if the circuit is behaving normally. If not, find the most likely health states of its components.

**Evidence variables**
Primary inputs and output of the circuit, A, B and E.
Problem statement

Given some values for the circuit primary inputs and output (test vector), decide if the circuit is behaving normally. If not, find the most likely health states of its components.

Evidence variables

Primary inputs and output of the circuit, A, B and E.

Query variables

Health of components X, Y and Z.
Problem statement
Given some values for the circuit primary inputs and output (test vector), decide if the circuit is behaving normally. If not, find the most likely health states of its components.

Evidence variables
Primary inputs and output of the circuit, $A$, $B$ and $E$.

Query variables
Health of components $X$, $Y$ and $Z$.

Intermediary variables
Internal wires, $C$ and $D$. 
Diagnosis III: Model from Design

Values of circuit wires: low or high

Health states: ok or faulty

faulty is too vague as a component may fail in a number of modes.

- **stuck-at-zero fault**: low output regardless of gate inputs.
- **stuck-at-one fault**: high output regardless of gate inputs.
- **input-output-short fault**: inverter shorts input to its output.

Fault modes demand more when specifying the CPTs.
### Three classes of CPTs

- primary inputs \((A, B)\)
- gate outputs \((C, D, E)\)
- component health \((X, Y, Z)\)

### CPTs for health variables depend on their values

<table>
<thead>
<tr>
<th>(X)</th>
<th>(\theta_X)</th>
</tr>
</thead>
<tbody>
<tr>
<td>ok</td>
<td>.99</td>
</tr>
<tr>
<td>faulty</td>
<td>.01</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>(X)</th>
<th>(\theta_X)</th>
</tr>
</thead>
<tbody>
<tr>
<td>ok</td>
<td>.99</td>
</tr>
<tr>
<td>stuckat0</td>
<td>.005</td>
</tr>
<tr>
<td>stuckat1</td>
<td>.005</td>
</tr>
</tbody>
</table>

Need to know the probabilities of various fault modes.
CPTs for component outputs determined from functionality.

Example

| A  | X    | C  | $\theta_{c|a,x}$ |
|----|------|----|------------------|
| high | ok   | high | 0                |
| low  | ok   | high | 1                |
| high | stuckat0 | high | 0                |
| low  | stuckat0 | high | 0                |
| high | stuckat1 | high | 1                |
| low  | stuckat1 | high | 1                |
CPTs for component outputs determined from functionality.

### Example

| $A$ | $X$      | $C$  | $\theta_{c|a,x}$ |
|-----|----------|------|------------------|
| high| ok       | high | 0                |
| low | ok       | high | 1                |
| high| stuck0   | high | 0                |
| low | stuck0   | high | 0                |
| high| stuck1   | high | 1                |
| low | stuck1   | high | 1                |

### If we do not represent health states:

| $A$ | $X$ | $C$ | $\theta_{c|a,x}$ |
|-----|-----|-----|------------------|
| high| ok  | high| 0                |
| low | ok  | high| 1                |
| high| faulty | high| ?                |
| low | faulty | high| ?                |

Common to use a probability of .50 in this case.
A Diagnosis Example

Example

Given test vector $e$: $A=$ high, $B=$ high, $E=$ low, compute MAP over health variables $X$, $Y$ and $Z$. 
A Diagnosis Example

**Example**

Given test vector $e$: $A=$ high, $B=$ high, $E=$ low, compute MAP over health variables $X$, $Y$ and $Z$.

**Network with fault modes gives two MAP instantiations:**

<table>
<thead>
<tr>
<th>MAP given $e$</th>
<th>$X$</th>
<th>$Y$</th>
<th>$Z$</th>
<th>each probability $\approx 49.4%$</th>
</tr>
</thead>
<tbody>
<tr>
<td>ok</td>
<td>ok</td>
<td>stuckat0</td>
<td>ok</td>
<td></td>
</tr>
<tr>
<td>ok</td>
<td>ok</td>
<td>ok</td>
<td>stuckat0</td>
<td></td>
</tr>
</tbody>
</table>
Example

Given test vector $e$: $A = \text{high}$, $B = \text{high}$, $E = \text{low}$, compute MAP over health variables $X$, $Y$ and $Z$.

Network with fault modes gives two MAP instantiations:

<table>
<thead>
<tr>
<th>MAP given $e$</th>
<th>$X$</th>
<th>$Y$</th>
<th>$Z$</th>
</tr>
</thead>
<tbody>
<tr>
<td>ok</td>
<td>ok</td>
<td>stuckat0</td>
<td>ok</td>
</tr>
<tr>
<td>ok</td>
<td>ok</td>
<td>ok</td>
<td>stuckat0</td>
</tr>
</tbody>
</table>

Network with no fault modes gives two MAP instantiations:

<table>
<thead>
<tr>
<th>MAP given $e$</th>
<th>$X$</th>
<th>$Y$</th>
<th>$Z$</th>
</tr>
</thead>
<tbody>
<tr>
<td>ok</td>
<td>ok</td>
<td>faulty</td>
<td>ok</td>
</tr>
<tr>
<td>ok</td>
<td>ok</td>
<td>ok</td>
<td>faulty</td>
</tr>
</tbody>
</table>
Suppose we have two test vectors instead of only one.
Integrating Time

Suppose we have two test vectors instead of only one.

Additional evidence variables

$A'$, $B'$ and $E'$
Suppose we have two test vectors instead of only one.

Additional evidence variables

\( A', B', \text{ and } E' \)

Additional intermediary variables

\( C' \text{ and } D' \)
Integrating Time

Suppose we have two test vectors instead of only one.

**Additional evidence variables**

$A'$, $B'$ and $E'$

**Additional intermediary variables**

$C'$ and $D'$

**Additional health variables on whether we allow intermittent faults**

If health of a component can change from one test to another, we need additional health variables $X'$, $Y'$, and $Z'$. Otherwise, the original health variables are sufficient.

Variables? Values? Structure?
Integrating Time: No Intermittent Faults

Two test vectors

\( e \): \( A = \text{high}, \ B = \text{high}, \ E = \text{low} \)

\( e' \): \( A = \text{low}, \ B = \text{low}, \ E = \text{low} \).
Integrating Time: No Intermittent Faults

Two test vectors

\[ e : A = \text{high}, \ B = \text{high}, \ E = \text{low} \]
\[ e' : A = \text{low}, \ B = \text{low}, \ E = \text{low}. \]

MAP using second structure

<table>
<thead>
<tr>
<th>MAP given ( e, e' )</th>
<th>( X )</th>
<th>( Y )</th>
<th>( Z )</th>
</tr>
</thead>
<tbody>
<tr>
<td>ok</td>
<td>ok</td>
<td>faulty</td>
<td></td>
</tr>
</tbody>
</table>

with probability \( \approx 97.53\% \)
Integrating Time: Intermittent Faults

Dynamic Bayesian network (DBN)

Two test vectors
\( e: A = \text{high}, \ B = \text{high}, \ E = \text{low} \)
\( e': A = \text{low}, \ B = \text{low}, \ E = \text{low}. \)

Persistence model for the health of component \( X \)

<table>
<thead>
<tr>
<th>( X )</th>
<th>( X' )</th>
<th>( \theta_{x' \mid x} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>ok</td>
<td>ok</td>
<td>.99</td>
</tr>
<tr>
<td>ok</td>
<td>faulty</td>
<td>.01</td>
</tr>
<tr>
<td>faulty</td>
<td>ok</td>
<td>.001</td>
</tr>
<tr>
<td>faulty</td>
<td>faulty</td>
<td>.999</td>
</tr>
</tbody>
</table>
Four bits $U_1, U_2, U_3$ and $U_4$ are sent from a source $S$ to a destination $D$ over a noisy channel, where there is a 1% chance that a bit will be inverted before it gets to the destination.
Channel Coding

Four bits $U_1, U_2, U_3$ and $U_4$ are sent from a source $S$ to a destination $D$ over a noisy channel, where there is a 1% chance that a bit will be inverted before it gets to the destination.

To improve the reliability of this process we will add three redundant bits $X_1, X_2$ and $X_3$ to the message, where $X_1$ is the XOR of $U_1$ and $U_3$, $X_2$ is the XOR of $U_2$ and $U_4$, and $X_3$ is the XOR of $U_1$ and $U_4$. 
Channel Coding

Four bits $U_1$, $U_2$, $U_3$ and $U_4$ are sent from a source $S$ to a destination $D$ over a noisy channel, where there is a 1% chance that a bit will be inverted before it gets to the destination.

To improve the reliability of this process we will add three redundant bits $X_1$, $X_2$ and $X_3$ to the message, where $X_1$ is the XOR of $U_1$ and $U_3$, $X_2$ is the XOR of $U_2$ and $U_4$, and $X_3$ is the XOR of $U_1$ and $U_4$.

Given that we received a message containing seven bits at destination $D$ our goal is to restore the message generated at the source $S$.

Try it: Variables, values, structure?
In channel coding terminology

$U_1, \ldots, U_4$ are known as information bits;
$X_1, \ldots, X_3$ are known as redundant bits;
$U_1, \ldots, U_4, X_1, \ldots, X_3$ is known as the code word or channel input;
$Y_1, \ldots, Y_7$ is known as the channel output.
In channel coding terminology

$U_1, \ldots, U_4$ are known as information bits;
$X_1, \ldots, X_3$ are known as redundant bits;
$U_1, \ldots, U_4, X_1, \ldots, X_3$ is known as the code word or channel input;
$Y_1, \ldots, Y_7$ is known as the channel output.

Goal to restore the channel input given some channel output.
Channel Coding

In channel coding terminology

$U_1, \ldots, U_4$ are known as information bits;
$X_1, \ldots, X_3$ are known as redundant bits;
$U_1, \ldots, U_4, X_1, \ldots, X_3$ is known as the code word or channel input;
$Y_1, \ldots, Y_7$ is known as the channel output.

Goal to restore the channel input given some channel output.

Evidence variables are

$Y_1, \ldots, Y_7$: bits received at destination $D$
Channel Coding

In channel coding terminology

$U_1, \ldots, U_4$ are known as information bits;
$X_1, \ldots, X_3$ are known as redundant bits;
$U_1, \ldots, U_4, X_1, \ldots, X_3$ is known as the code word or channel input;
$Y_1, \ldots, Y_7$ is known as the channel output.

Goal to restore the channel input given some channel output.

Evidence variables are

$Y_1, \ldots, Y_7$: bits received at destination $D$

Query variables are

$U_1, \ldots, U_4$: bits originating at source $S$
Channel Coding

In channel coding terminology

\( U_1, \ldots, U_4 \) are known as information bits;
\( X_1, \ldots, X_3 \) are known as redundant bits;
\( U_1, \ldots, U_4, X_1, \ldots, X_3 \) is known as the code word or channel input;
\( Y_1, \ldots, Y_7 \) is known as the channel output.

Goal to restore the channel input given some channel output.

Evidence variables are
\( Y_1, \ldots, Y_7 \): bits received at destination \( D \)

Query variables are
\( U_1, \ldots, U_4 \): bits originating at source \( S \)

Bits \( X_1, \ldots, X_3 \) either query variables or intermediary variables.
There are three CPT types in the problem.
There are three CPT types in the problem.

CPT for each redundant bit, say $X_1$:

$$P_r(x_1|u_1, u_3) = 1 \text{ iff } x_1 = u_1 \oplus u_3 \ (\oplus \text{ is the XOR function})$$
There are three CPT types in the problem.
Channel Coding

There are three CPT types in the problem.

CPT for a channel output bit, say $Y_1$:

| $U_1$ | $Y_1$ | $\theta_{Y_1|U_1}$ |
|-------|-------|---------------------|
| 1     | 0     | .01                 |
| 0     | 1     | .01                 |

CPT captures the simple noise model given in the problem statement.
There are three CPT types in the problem.
Channel Coding

There are three CPT types in the problem.

CPT for information bits, such as $U_1$:

<table>
<thead>
<tr>
<th>$U_1$</th>
<th>$\theta_{u_1}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>.5</td>
</tr>
<tr>
<td>0</td>
<td>.5</td>
</tr>
</tbody>
</table>

Captures the distribution of messages sent out from the source $S$.

What queries should we use here?
MAP or Posterior-Marginal (PM) Decoders?

To restore the channel input given channel output

1. Compute a **MAP** for the channel input $U_1, \ldots, U_4, X_1, \ldots, X_3$ given channel output $Y_1, \ldots, Y_7$.
2. Compute the **PM** for each bit $U_i/X_i$ in the channel input, given channel output $Y_1, \ldots, Y_7$, and then select the value of $U_i/X_i$ which is most probable.
MAP or Posterior-Marginal (PM) Decoders?

To restore the channel input given channel output

1. Compute a MAP for the channel input $U_1, \ldots, U_4, X_1, \ldots, X_3$ given channel output $Y_1, \ldots, Y_7$.
2. Compute the PM for each bit $U_i/X_i$ in the channel input, given channel output $Y_1, \ldots, Y_7$, and then select the value of $U_i/X_i$ which is most probable.

The choice between MAP and PM decoders is a matter of the performance measure one is interested in optimizing.

WER (word error rate), BER (bit error rate)

MAP (MPE) minimizes WER, PM minimize BER…
What do you think?
Noise Models and Soft Evidence

A more realistic and common noise model

Transmitting our code bits $x_i$ through a channel that adds Gaussian noise, with mean $x_i$ and standard deviation $\sigma$.

Channel output $Y_i$ is a continuous variable governed by

conditional density function $f(y_i|x_i) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(y_i-x_i)^2}{2\sigma^2}}$
Noise Models and Soft Evidence

A more realistic and common noise model

Transmitting our code bits $x_i$ through a channel that adds Gaussian noise, with mean $x_i$ and standard deviation $\sigma$.

Channel output $Y_i$ is a continuous variable governed by

conditional density function $f(y_i|x_i) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-(y_i-x_i)^2/2\sigma^2}$

Can be implemented by interpreting

channel output $y_i$ as soft evidence on the channel input $X_i=0$ with a Bayes factor $k = e^{(1-2y_i)/2\sigma^2}$
Convolutional and turbo codes correspond to different methods for generating redundant bits.
Convolutional and turbo codes correspond to different methods for generating redundant bits.

Convolutional and turbo codes provide examples of modeling systems with feedback loops using dynamic Bayesian networks.
An example convolutional encoder

Each node denoted with a “+” represents a binary addition, and each box $D_i$ represents a delay where the output of $D_i$ is the input of $D_i$ from the previous encoder state.
Convolutional Codes

Dynamic Bayesian network for a convolutional code.

A sequence of replicated slices

where slice $k$ is responsible for generating the codeword bits $x_{2k}$ and $x_{2k+1}$ for the information bit $u_k$. 
Convolutional Codes

A sequence of replicated slices
where slice $k$ is responsible for generating the codeword bits $x_{2k}$ and $x_{2k+1}$ for the information bit $u_k$.

Each slice has a variable $S_k$ representing the state of the encoder
This state is determined by the previous state variable $S_{k-1}$ and the information bit $U_k$. 

Dynamic Bayesian network for a convolutional code.
Given four information bits $u_0, \ldots, u_3$. 
Given four information bits $u_0, \ldots, u_3$.

In a convolutional code we generate 4 redundant bits leading to an 8-bit codeword.
Given four information bits $u_0, \ldots, u_3$.

**In a convolutional code**
we generate 4 redundant bits leading to an 8-bit codeword.

**In a turbo code we apply a convolutional code twice**

once on the original bit sequence $u_0, u_1, u_2, u_3$, and another on some permutation, say, $u_1, u_3, u_2, u_0$. This leads to 8 redundant bits and a 12-bit codeword.
Lower network represents a convolutional code
for the bit sequence $u_0, \ldots, u_3$.

Upper network represents a convolutional code
for the bit sequence $u_4, \ldots, u_7$. 
Commonsense Knowledge

Parameters based on a combination of sources

- **Statistical information** such as reliabilities of sensors and battery.
- **Subjective beliefs** relating to how often the wife goes out, guests are expected, the dog has bowel trouble, etc.
- **Objective beliefs** regarding the functionality of sensors.
A pedigree involving six individuals

Squares represent males, circles represent females. Horizontal edges connect spouses, while vertical edges connect couples to their children. For example, Jack and Sue are a couple with two daughters, Lydia and Nancy.
Genetic Linkage Analysis

The **ABO** gene is responsible for determining blood type. This gene has three alleles: **A**, **B** and **O**. Since each individual must have two alleles for this gene, we have six possible genotypes in this case.

<table>
<thead>
<tr>
<th>Genotype</th>
<th>Phenotype</th>
</tr>
</thead>
<tbody>
<tr>
<td>A/A</td>
<td>Blood type A</td>
</tr>
<tr>
<td>A/B</td>
<td>Blood type AB</td>
</tr>
<tr>
<td>A/O</td>
<td>Blood type A</td>
</tr>
<tr>
<td>B/B</td>
<td>Blood type B</td>
</tr>
<tr>
<td>B/O</td>
<td>Blood type B</td>
</tr>
<tr>
<td>O/O</td>
<td>Blood type O</td>
</tr>
</tbody>
</table>

There are only four different blood types

If someone has the blood type **A**, they could have the pair of alleles **A/A** or the pair **A/O** for their genotype.
The phenotype is not always determined precisely by the genotype.
The phenotype is not always determined precisely by the genotype.

### A disease gene with two alleles $H$ and $D$

<table>
<thead>
<tr>
<th>Genotype</th>
<th>Phenotype</th>
</tr>
</thead>
<tbody>
<tr>
<td>$H/H$</td>
<td>healthy</td>
</tr>
<tr>
<td>$H/D$</td>
<td>healthy</td>
</tr>
<tr>
<td>$D/D$</td>
<td>ill with probability .9</td>
</tr>
</tbody>
</table>
The phenotype is not always determined precisely by the genotype.

A disease gene with two alleles $H$ and $D$

<table>
<thead>
<tr>
<th>Genotype</th>
<th>Phenotype</th>
</tr>
</thead>
<tbody>
<tr>
<td>$H/H$</td>
<td>healthy</td>
</tr>
<tr>
<td>$H/D$</td>
<td>healthy</td>
</tr>
<tr>
<td>$D/D$</td>
<td>ill with probability .9</td>
</tr>
</tbody>
</table>

Penetrance

The conditional probability of observing a phenotype (e.g., healthy, ill) given the genotype (e.g., $H/H$, $H/D$, $D/D$).
The phenotype is not always determined precisely by the genotype.

A disease gene with two alleles $H$ and $D$

<table>
<thead>
<tr>
<th>Genotype</th>
<th>Phenotype</th>
</tr>
</thead>
<tbody>
<tr>
<td>$H/H$</td>
<td>healthy</td>
</tr>
<tr>
<td>$H/D$</td>
<td>healthy</td>
</tr>
<tr>
<td>$D/D$</td>
<td>ill with probability .9</td>
</tr>
</tbody>
</table>

Penetrance

The conditional probability of observing a phenotype (e.g., healthy, ill) given the genotype (e.g., $H/H$, $H/D$, $D/D$).

Example

Penetrance is always 0 or 1 for the $ABO$ gene.
Penetrance is .9 for the phenotype ill given the genotype $D/D$. 
Recombination Events

Haplotype

The alleles received by an individual from one parent. Each individual has two haplotypes, one paternal and another maternal.

Gene $G_1$ has alleles $A$ and $a$. Gene $G_2$ has alleles $B$ and $b$. 
Mary can pass only one haplotype to her child Jack: \( AB \).

John can pass only one haplotype to Jack: \( ab \).

Jack can pass one of four haplotypes to his children: \( AB, Ab, aB, ab \).
If two genes are inherited independently
the probability of a recombination is expected to be 1/2.

Genetic linkage
Two alleles which were passed in the haplotype from a grandparent to a parent tend to be passed again in the same haplotype from the parent to a child.

Goal of genetic linkage analysis
is to estimate the extent to which two genes are linked.
Genetic Linkage and Gene Maps

The extent to which genes $G_1$ and $G_2$ are linked is measured by a recombination fraction or frequency, $\theta$, which is the probability that a recombination between $G_1$ and $G_2$ will occur.

Genes that are inherited independently are characterized by a recombination frequency $\theta = 1/2$ and are said to be unlinked. Linked genes on the other hand are characterized by a recombination frequency $\theta < 1/2$. 
Genetic Linkage and Gene Maps

Linkage between genes is related to their locations on a chromosome within the cell nucleus. These locations are typically referred to as loci (singular: locus).

For genes that are closely located on a chromosome, linkage is inversely proportional to distance between their locations.

The recombination frequency can provide direct evidence on the distance between genes on a chromosome.
From Pedigrees to Bayesian Networks

Genotype and phenotype

— \( GP_{ij} \): paternal allele for individual \( i \) and gene \( j \)

— \( GM_{ij} \): maternal allele for individual \( i \) and gene \( j \)

— \( P_{ij} \): phenotype for individual \( i \) and gene \( j \)
From Pedigrees to Bayesian Networks

Selector variables

— $SP_{ij}$: determines how individual $i$ inherits alleles of gene $j$ from his father
— $SM_{ij}$: determines how individual $i$ inherits alleles of gene $j$ from his mother
From Pedigrees to Bayesian Networks

Selector variables

— $SP_{ij}$: determines how individual $i$ inherits alleles of gene $j$ from his father

— $SM_{ij}$: determines how individual $i$ inherits alleles of gene $j$ from his mother

If $SP_{ij} = p$ then individual $i$ will inherit the allele of gene $j$ that his father obtained from the grandfather.

If $SP_{ij} = m$ then individual $i$ will inherit the allele of gene $j$ that his father obtained from the grandmother.
\[ \theta_{gp_{ij}|gp_{kj},gm_{kj},sp_{ij}} = \begin{cases} 
1, & \text{if } sp_{ij} = p \text{ and } gp_{ij} = gp_{kj}; \\
1, & \text{if } sp_{ij} = m \text{ and } gp_{ij} = gm_{kj}; \\
0, & \text{otherwise.} 
\end{cases} \]

If \( SP_{ij} = p \) then the allele \( GP_{ij} \) for individual \( i \) and gene \( j \) will be inherited from the paternal haplotype of his father \( k \), \( GP_{kj} \).

If \( SP_{ij} = m \) then the allele \( GP_{ij} \) for individual \( i \) and gene \( j \) will be inherited from the maternal haplotype of his father \( k \), \( GM_{kj} \).
Selectors of second gene $SP_{32}$ and $SM_{32}$ have CPTs that are a function of recombination frequency $\theta_{12}$

Selectors of third gene $SP_{33}$ and $SM_{33}$ have CPTs that are a function of recombination frequency $\theta_{23}$
From Pedigrees to Bayesian Networks

CPT for selector variable $SP_{32}$ encodes the recombination frequency $\theta_{12}$
From Pedigrees to Bayesian Networks

CPT for selector variable $SP_{32}$ encodes the recombination frequency $\theta_{12}$

| $SP_{31}$ | $SP_{32}$ | $\theta_{sp_{32}|sp_{31}}$ |
|-----------|-----------|-----------------------------|
| $p$       | $p$       | 1 $-$ $\theta_{12}$         |
| $p$       | $m$       | $\theta_{12}$               | recombination between genes 1 and 2 |
| $m$       | $p$       | $\theta_{12}$               | recombination between genes 1 and 2 |
| $m$       | $m$       | 1 $-$ $\theta_{12}$         |
Two Loci Inheritance

A A 1 2  a a
B B  a b

A a 3 4  a a
B b

A a 5 6  A a
B b

Recombinant
\[ P(s_{23t} \mid s_{13t}, \theta) = \begin{bmatrix} 1 - \theta & \theta \\ \theta & 1 - \theta \end{bmatrix} \] where \( t \in \{m,f\} \)
Linkage analysis:
6 people, 3 markers