What are Bayesian networks?

- Simple, graphical notation for conditional independence assertions
- Allows compact specification of full joint distributions
- Syntax:
  - a set of nodes, one per random variable
  - a directed, acyclic graph (link = "directly influences")
  - a conditional distribution for each node given its parents:
    \[ P(X_i \mid \text{Parents}(X_i)) \]

- For discrete variables, conditional distribution = conditional probability table (CPT) = distribution over \( X_i \) for each combination of parent values
Back at the Dentist’s

- Topology of network encodes conditional independence assertions:

  ![Graph](image)

  - Weather is independent of the other variables
  - Toothache and Catch are conditionally independent of each other given Cavity

Example 2: Burglars and Earthquakes

- You are at a "Done with 473" party at a friend’s.
- Neighbor John calls to say your home alarm is ringing (but neighbor Mary doesn’t).
- Sometimes your alarm is set off by minor earthquakes.

- Question: Is your home being burglarized?

- Variables: Burglary, Earthquake, Alarm, JohnCalls, MaryCalls

- Network topology reflects "causal" knowledge:
  - A burglar can set the alarm off
  - An earthquake can set the alarm off
  - The alarm can cause Mary to call
  - The alarm can cause John to call
Compact Representation of Probabilities in Bayesian Networks

- A CPT for Boolean $X_i$ with $k$ Boolean parents has $2^k$ rows for the combinations of parent values
- Each row requires 1 number $p$ for $X_i = \text{true}$
  (the number for $X_i = \text{false}$ is just $1-p$)
- If each variable has no more than $k$ parents, the complete network requires $O(n \cdot 2^k)$ numbers
  I.e., grows linearly with $n$, vs. $O(2^n)$ for full joint distribution
- For our network, $1+1+4+2+2 = 10$ numbers (vs. $2^5-1 = 31$)
Semantics

Full joint distribution is defined as product of local conditional distributions:

\[ P(X_1, ... , X_n) = \prod_{i=1}^{n} P(X_i | \text{Parents}(X_i)) \]

e.g., \( P(j \land m \land a \land \neg b \land \neg e) \)

\[ = P(j | a) P(m | a) P(a | \neg b, \neg e) P(\neg b) P(\neg e) \]

Constructing Bayesian networks

• 1. Choose an ordering of variables \( X_1, ... , X_n \)
• 2. For \( i = 1 \) to \( n \)
  add \( X_i \) to the network
  select parents from \( X_1, ... , X_{i-1} \) such that
  \[ P(X_i | \text{Parents}(X_i)) = P(X_i | X_1, ... , X_{i-1}) \]

This choice of parents guarantees:

\[ P(X_1, ... , X_n) = P(X_1, ... , X_{i-1}, X_i) P(X_{i+1}, ..., X_n) \]

\[ = P(X_1, ... , X_{i-1}) P(X_{i+1} | X_1, ... , X_{i-1}) P(X_{i+2} | X_2, ... , X_{i-2}) \]

\[ = \prod_{i=1}^{n} P(X_i | X_1, ... , X_{i-1}) \text{ (chain rule)} \]

\[ = \prod_{i=1}^{n} P(X_i | \text{Parents}(X_i)) \text{ (by construction)} \]
Example

• Suppose we choose the ordering $M, J, A, B, E$

$$P(J \mid M) = P(J)$$?

Example

• Suppose we choose the ordering $M, J, A, B, E$

$$P(J \mid M) = P(J)? \quad \text{No}$$

$$P(A \mid J, M) = P(A \mid J)? \quad P(A \mid J, M) = P(A)?$$
Example

- Suppose we choose the ordering $M, J, A, B, E$

$$P(J | M) = P(J) \ ? \ Yes$$
$$P(A | J, M) = P(A | J) \ ? \ Yes$$
$$P(A | J, M) = P(A) \ ? \ No$$
$$P(B | A, J, M) = P(B) \ ? \ Yes$$
$$P(B | A, J, M) = P(B | A) \ ? \ Yes$$

Example

- Suppose we choose the ordering $M, J, A, B, E$

$$P(J | M) = P(J) \ ? \ Yes$$
$$P(A | J, M) = P(A | J) \ ? \ Yes$$
$$P(A | J, M) = P(A) \ ? \ Yes$$
$$P(B | A, J, M) = P(B) \ ? \ Yes$$
$$P(B | A, J, M) = P(B | A) \ ? \ Yes$$
$$P(E | B, A, J, M) = P(E | A) \ ? \ Yes$$
$$P(E | B, A, J, M) = P(E | A, B) \ ? \ Yes$$
Example

- Suppose we choose the ordering $M, J, A, B, E$

\[
\begin{align*}
P(J | M) &= P(J)? \text{No} \\
P(A | J, M) &= P(A | J)? \text{No} \quad P(A | J, M) = P(A)? \text{No} \\
P(B | A, J, M) &= P(B)? \text{No} \\
P(B | A, J, M) &= P(B | A)? \text{Yes} \\
P(E | B, A, J, M) &= P(E | A)? \text{No} \\
P(E | B, A, J, M) &= P(E | A, B)? \text{Yes}
\end{align*}
\]

Example contd.

- Deciding conditional independence is hard in non-causal directions
- Causal models and conditional independence seem hardwired for humans! (recent Cog Sci research)
- Non-causal network is less compact: $1 + 2 + 4 + 2 + 4 = 13$ numbers (vs. $1+1+4+2+2 = 10$ numbers)
Lesson: Add nodes representing “root causes” first, then the variables they influence, and so on.

Keep it causal, baby!

Probabilistic Inference in BNs

• The graphical independence representation yields efficient inference schemes
• We generally want to compute
  \[ P(X|E) \] where \( E \) is evidence from sensory measurements etc. (known values for variables)
  Sometimes, may want to compute just \( P(X) \)
• One simple algorithm:
  variable elimination (VE)
\[ P(B \mid J = \text{true}, M = \text{true}) \]

\[
P(b \mid j, m) = \alpha \sum_{e,a} P(b,j,m,e,a)
\]

\[ P(B \mid J = \text{true}, M = \text{true}) \]

\[
P(b \mid j, m) = \alpha P(b) \sum_e P(e) \sum_a P(a \mid b, e)P(j \mid a)P(m \mid a)
\]
### Structure of Computation

Repeated computations $\Rightarrow$ use dynamic programming?

### Variable Elimination

- A *factor* is a function from some set of variables to a specific value: e.g., $f(E, A, \text{Mary})$

  CPTs are factors, e.g., $P(A|E, B)$ function of $A, E, B$

- VE works by *eliminating* all variables in turn until there is a factor with only query variable

- To eliminate a variable:
  1. *join* all factors containing that variable (like DBs/SQL), multiplying probabilities
  2. *sum out* the influence of the variable on new factor
Example of VE: $P(J)$

$$P(J) = \sum_{M,A,B,E} P(J,M,A,B,E) = \sum_{M,A,B,E} P(J|A)P(M|A) P(B|E)P(A|B,E)P(E)$$

$$= \sum_A P(J|A) \sum_M P(M|A) \sum_B P(B) \sum_E P(A|B,E)P(E)$$

$$= \sum_A P(J|A) \sum_M P(M|A) \sum_B P(B) f_1(A,B)$$

$$= \sum_A P(N1|A) \sum_M P(M|A) f_2(A)$$

$$= \sum_A P(J|A) f_3(A)$$

$$= f_4(J)$$

Other Inference Algorithms

- **Direct Sampling:**
  Repeat $N$ times:
  - Use random number generator to generate sample values for each node
  - Start with nodes with no parents
  - Condition on sampled parent values for other nodes
  Count frequencies of samples to get an approximation to joint distribution

- **Other variants:** Rejection sampling, likelihood weighting, Gibbs sampling and other MCMC methods (see text)

- **Belief Propagation:** A "message passing" algorithm for approximating $P(X|evidence)$ for each node variable $X$

- **Variational Methods:** Approximate inference using distributions that are more tractable than original ones
Summary

- Bayesian networks provide a natural way to represent conditional independence
- Network topology + CPTs = compact representation of joint distribution
- Generally easy for domain experts to construct
- BNs allow inference algorithms such as VE that are efficient in many cases

Next Time

- Machine Learning!