Bayesian networks

Chapter 14.4–4

Chapter 15.1–2

Outline

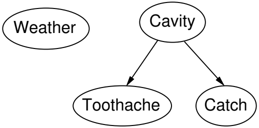
- Syntax
- Semantics
- Inference

Bayesian Networks

- Bayes nets allow for the *compact specification* of full *joint distributions*
- They do this by providing a simple, graphical notation for conditional independence assertions
- Syntax:
  - a set of nodes, one per variable
  - links: causal relations (or: link $\approx$ "directly influences")
  - a conditional distribution for each node, given its parents:
    $\mathbf{P}(X_i | Parents(X_i))$
- A Bayes net is a directed, acyclic graph of such vertices, links, and conditional distributions
- In the simplest case, the conditional distribution is represented as a *conditional probability table* (CPT) giving the distribution over X_i for each combination of parent values

Example

- The topology of the network encodes conditional independence assertions:



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

Example

- You are given the following information:
  - You have had a burglar alarm installed.
  - 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.
- Variables: ???

Example

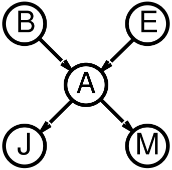
- You are given the following information:
  - You have had a burglar alarm installed.
  - A burglar can set the alarm off.
  - An earthquake can set the alarm off.
  - The alarm can cause Mary to call.
  - The alarm can cause John to call.
- Variables: $Burglar$, $Earthquake$, $Alarm$, $JohnCalls$, $MaryCalls$

Example

- You are given the following information:
  - You have had a burglar alarm installed.
  - 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.
- Variables: $Burglar$, $Earthquake$, $Alarm$, $JohnCalls$, $MaryCalls$
- Graph: ??

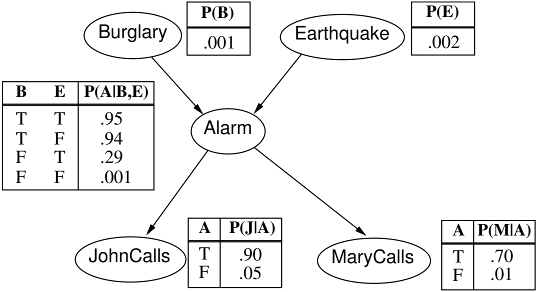
Example

- You are given the following information:
  - You have had a burglar alarm installed.
  - 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.
- Variables: *Burglar*, *Earthquake*, *Alarm*, *JohnCalls*, *MaryCalls*
- The network topology reflects “causal” knowledge:

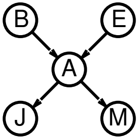


Example contd.

The complete network has associated probability tables:

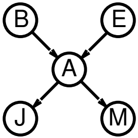


Conditional probability tables



- A CPT for Boolean X_i with k Boolean parents has 2^k rows for the combinations of parent values
- Each row requires one number p for $X_i = true$ (the number for $X_i = false$ 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 the full joint distribution
- For burglary net, $1+1+4+2+2 = 10$ numbers (vs. 2^5–$1 = 31$)

Global semantics

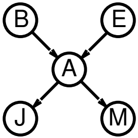


- *Global semantics* defines the full joint distribution as the product of the local conditional distributions:

$$P(x_1, \ldots, x_n) = \Pi_{i=1}^{n} P(x_i | parents(X_i))$$

- E.g., $P(j \wedge m \wedge a \wedge \neg b \wedge \neg e)$

Global semantics



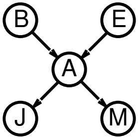
- *Global semantics* defines the full joint distribution as the product of the local conditional distributions:

$$P(x_1, \ldots, x_n) = \Pi_{i=1}^{n} P(x_i | parents(X_i))$$

- E.g., $P(j \wedge m \wedge a \wedge \neg b \wedge \neg e)$

$$= \quad P(j|a)P(m|a)P(a|\neg b, \neg e)P(\neg b)P(\neg e)$$

Global semantics



- *Global semantics* defines the full joint distribution as the product of the local conditional distributions:

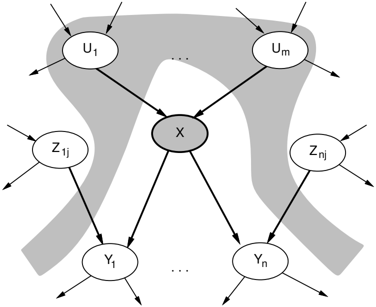
$$P(x_1, \ldots, x_n) = \Pi_{i=1}^{n} P(x_i | parents(X_i))$$

- E.g., $P(j \wedge m \wedge a \wedge \neg b \wedge \neg e)$

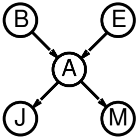
$$\begin{aligned} &= P(j|a)P(m|a)P(a|\neg b, \neg e)P(\neg b)P(\neg e) \\ &= 0.9 \times 0.7 \times 0.001 \times 0.999 \times 0.998 \\ &\approx 0.00063 \end{aligned}$$

Local semantics

- *Local semantics*: Each node is conditionally independent of its nondescendants given its parents



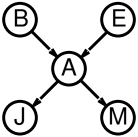
Local semantics



Examples:

- JohnCalls, MaryCalls are conditionally independent given Alarm.

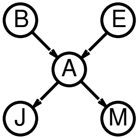
Local semantics



Examples:

- JohnCalls, MaryCalls are conditionally independent given Alarm.
- MaryCalls is conditionally independent of Burglary given Alarm.

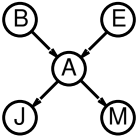
Local semantics



Examples:

- JohnCalls, MaryCalls are conditionally independent given Alarm.
- MaryCalls is conditionally independent of Burglary given Alarm.
- Burglary and Earthquake are independent.

Local semantics



Examples:

- JohnCalls, MaryCalls are conditionally independent given Alarm.
- MaryCalls is conditionally independent of Burglary given Alarm.
- Burglary and Earthquake are independent.

☞ The graph topology alone (without the probability tables) specifies the conditional independencies.

Inference tasks

- *Simple queries*: compute posterior marginal $\mathbf{P}(X_i|\mathbf{e})$
  - Use: $\mathbf{P}(X_i|\mathbf{e}) = \alpha\Sigma_{\mathbf{y}}\mathbf{P}(X_i, \mathbf{e}, \mathbf{y})$

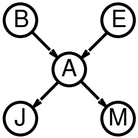
Inference tasks

- *Simple queries*: compute posterior marginal $\mathbf{P}(X_i|\mathbf{e})$
  - Use: $\mathbf{P}(X_i|\mathbf{e}) = \alpha \Sigma_{\mathbf{y}} \mathbf{P}(X_i, \mathbf{e}, \mathbf{y})$
- *Conjunctive queries*: $\mathbf{P}(X_i, X_j|\mathbf{e}) = \mathbf{P}(X_i|\mathbf{e})\mathbf{P}(X_j|X_i, \mathbf{e})$

Inference tasks

- *Simple queries*: compute posterior marginal $\mathbf{P}(X_i|\mathbf{e})$
  - Use: $\mathbf{P}(X_i|\mathbf{e}) = \alpha \Sigma_{\mathbf{y}} \mathbf{P}(X_i, \mathbf{e}, \mathbf{y})$
- *Conjunctive queries*: $\mathbf{P}(X_i, X_j|\mathbf{e}) = \mathbf{P}(X_i|\mathbf{e})\mathbf{P}(X_j|X_i, \mathbf{e})$
- Lots of others:
  - *Optimal decisions*: decision networks include utility information; probabilistic inference required for
    $P(outcome|action, evidence)$
  - *Value of information*: which evidence to seek next?
  - *Sensitivity analysis*: which probability values are most critical?
  - *Explanation*: e.g. why do I need a new starter motor?

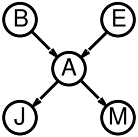
Inference by enumeration



- Slightly intelligent way to sum out variables from the joint without actually constructing its explicit representation
- Simple query on the burglary network: $\mathbf{P}(B|j,m)$
  $= \mathbf{P}(B,j,m)/P(j,m)$
  $= \alpha\mathbf{P}(B,j,m)$
  $= \alpha\ \Sigma_e\ \Sigma_a\ \mathbf{P}(B,e,a,j,m)$
  $= \alpha\langle 0.00059224, 0.0014919\rangle \approx \langle 0.284, 0.716\rangle$

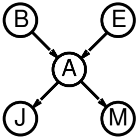
☞ Can always compute $\mathbf{P}(X|Y)$ by brute force.

Irrelevant variables



- Consider the query $\mathbf{P}(JohnCalls|Burglary = true)$
- Intuitively, whether Mary calls or not is irrelevant.

Irrelevant variables

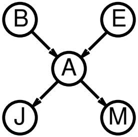


- Consider the query $\mathbf{P}(JohnCalls|Burglary = true)$

$$\mathbf{P}(J|b) = \alpha P(b) \sum_e P(e) \sum_a P(a|b,e) \mathbf{P}(J|a) \sum_m P(m|a)$$

- Sum over m is identically 1; M is *irrelevant* to the query

Irrelevant variables

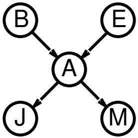


- Consider the query $\mathbf{P}(JohnCalls|Burglary = true)$

$$\mathbf{P}(J|b) = \alpha P(b) \sum_e P(e) \sum_a P(a|b,e) \mathbf{P}(J|a) \sum_m P(m|a)$$

- Sum over m is identically 1; M is *irrelevant* to the query
- Theorem: If $Y \notin Ancestors(\{X\} \cup \mathbf{E})$ then Y is irrelevant.

Irrelevant variables

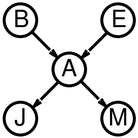


- Consider the query $\mathbf{P}(JohnCalls|Burglary = true)$

  $$\mathbf{P}(J|b) = \alpha P(b) \sum_e P(e) \sum_a P(a|b,e) \mathbf{P}(J|a) \sum_m P(m|a)$$

- Sum over m is identically 1; M is *irrelevant* to the query
- Theorem: If $Y \notin Ancestors(\{X\} \cup \mathbf{E})$ then Y is irrelevant.
  - Here, $X = JohnCalls$, $\mathbf{E} = \{Burglary\}$, and
    $Ancestors(\{X\} \cup \mathbf{E}) = \{Alarm, Earthquake\}$
  - So $MaryCalls$ is irrelevant

Irrelevant variables contd.



- Defn: *Moral graph* of Bayes net: connect all parents and make edges undirected
- Defn: **A** is *m-separated* from **B** by **C** iff **A**, **B** are separated by **C** in the moral graph
  - I.e. every path between **A** and **B** goes through a vertex in **C**.
- Theorem: Y is irrelevant if m-separated from X by **E**
- For $P(JohnCalls|Alarm = true)$, both
  Burglary and *Earthquake* are irrelevant

Application: Temporal Probability Models (Chapter 15)

- We'll briefly consider using BNs to reason about temporal processes
- We'll briefly discuss:
  - Time and uncertainty
  - Markov processes
  - Inference: filtering, prediction, smoothing

Time and Uncertainty

- The world is *stochastic*; we need to track and predict it
- Consider e.g.: Diabetes management
  - Adjust medication based on blood sugar and insulin levels.
  - Here the *dynamic* aspects of the problem are important
- Contrast this task with planning, where the agent changes the world.
  - (However one can also have probabilistic planning, where actions have effects with some probability.)

Time and Uncertainty

- *Problem*: Need to keep track of *time* and the unfolding of events.

Time and Uncertainty

- *Problem*: Need to keep track of *time* and the unfolding of events.
- *Basic idea*:
  - We have (unobserved) *state* variables and (observable) *evidence* variables that give information about the state.
  - Make a copy of state and evidence variables for each time step

Time and Uncertainty

- *Problem*: Need to keep track of *time* and the unfolding of events.
- *Basic idea*:
  - We have (unobserved) *state* variables and (observable) *evidence* variables that give information about the state.
  - Make a copy of state and evidence variables for each time step
- Thus:
  - $\mathbf{X}_t$ = set of unobservable state variables at time t
    e.g., $BloodSugar_t$, $StomachContents_t$, etc.
  - $\mathbf{E}_t$ = set of observable evidence variables at time t
    e.g., $MeasuredBloodSugar_t$, $PulseRate_t$, $FoodEaten_t$

Time and Uncertainty

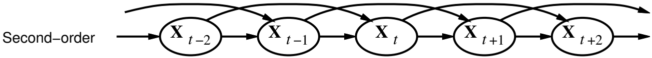
- *Problem*: Need to keep track of *time* and the unfolding of events.
- *Basic idea*:
  - We have (unobserved) *state* variables and (observable) *evidence* variables that give information about the state.
  - Make a copy of state and evidence variables for each time step
- Thus:
  - $\mathbf{X}_t$ = set of unobservable state variables at time t
    e.g., $BloodSugar_t$, $StomachContents_t$, etc.
  - $\mathbf{E}_t$ = set of observable evidence variables at time t
    e.g., $MeasuredBloodSugar_t$, $PulseRate_t$, $FoodEaten_t$
- Notation: $\mathbf{X}_{a:b} = \mathbf{X}_a, \mathbf{X}_{a+1}, \ldots, \mathbf{X}_{b-1}, \mathbf{X}_b$
- Aside: This assumes *discrete time*; where the step size depends on problem

Markov Processes (Markov Chains)

- Construct a Bayes net from these variables.
- *Markov assumption*: $\mathbf{X}_t$ depends on *bounded* subset of $\mathbf{X}_{0:t-1}$

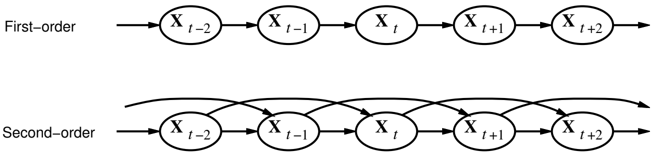
Markov Processes (Markov Chains)

- Construct a Bayes net from these variables.
- *Markov assumption*: $\mathbf{X}_t$ depends on *bounded* subset of $\mathbf{X}_{0:t-1}$
  - *First-order Markov process*: $\mathbf{P}(\mathbf{X}_t|\mathbf{X}_{0:t-1}) = \mathbf{P}(\mathbf{X}_t|\mathbf{X}_{t-1})$
  - *Second-order MP*: $\mathbf{P}(\mathbf{X}_t|\mathbf{X}_{0:t-1}) = \mathbf{P}(\mathbf{X}_t|\mathbf{X}_{t-2}, \mathbf{X}_{t-1})$



Markov Processes (Markov Chains)

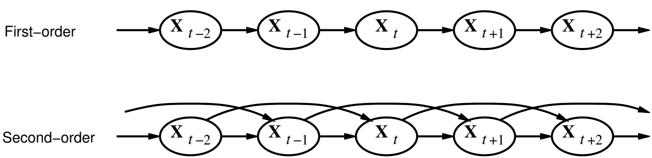
- Construct a Bayes net from these variables.
- *Markov assumption*: $\mathbf{X}_t$ depends on *bounded* subset of $\mathbf{X}_{0:t-1}$
  - *First-order Markov process*: $\mathbf{P}(\mathbf{X}_t|\mathbf{X}_{0:t-1}) = \mathbf{P}(\mathbf{X}_t|\mathbf{X}_{t-1})$
  - *Second-order MP*: $\mathbf{P}(\mathbf{X}_t|\mathbf{X}_{0:t-1}) = \mathbf{P}(\mathbf{X}_t|\mathbf{X}_{t-2}, \mathbf{X}_{t-1})$



- *Sensor Markov assumption*: $\mathbf{P}(\mathbf{E}_t|\mathbf{X}_{0:t}, \mathbf{E}_{0:t-1}) = \mathbf{P}(\mathbf{E}_t|\mathbf{X}_t)$

Markov Processes (Markov Chains)

- Construct a Bayes net from these variables.
- *Markov assumption*: $\mathbf{X}_t$ depends on *bounded* subset of $\mathbf{X}_{0:t-1}$
  - *First-order Markov process*: $\mathbf{P}(\mathbf{X}_t|\mathbf{X}_{0:t-1}) = \mathbf{P}(\mathbf{X}_t|\mathbf{X}_{t-1})$
  - *Second-order MP*: $\mathbf{P}(\mathbf{X}_t|\mathbf{X}_{0:t-1}) = \mathbf{P}(\mathbf{X}_t|\mathbf{X}_{t-2}, \mathbf{X}_{t-1})$



- *Sensor Markov assumption*: $\mathbf{P}(\mathbf{E}_t|\mathbf{X}_{0:t}, \mathbf{E}_{0:t-1}) = \mathbf{P}(\mathbf{E}_t|\mathbf{X}_t)$
- *Stationary process*: Transition model $\mathbf{P}(\mathbf{X}_t|\mathbf{X}_{t-1})$ and sensor model $\mathbf{P}(\mathbf{E}_t|\mathbf{X}_t)$ fixed for all t

Example

- You're stuck in a building with no windows and want to figure out whether it's raining outside.

Example

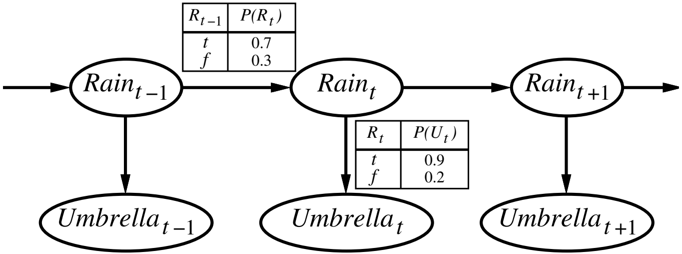
- You're stuck in a building with no windows and want to figure out whether it's raining outside.
- The only evidence you have is whether your office mate is carying an umbrella or not.

Example

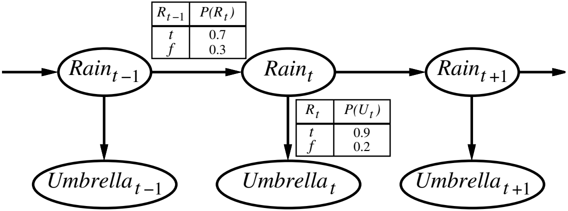
- You're stuck in a building with no windows and want to figure out whether it's raining outside.
- The only evidence you have is whether your office mate is carying an umbrella or not.
- As well, whether it's raining or not is related to whether it was raining previously or not.

Example

- You're stuck in a building with no windows and want to figure out whether it's raining outside.
- The only evidence you have is whether your office mate is carying an umbrella or not.
- As well, whether it's raining or not is related to whether it was raining previously or not.



Example



- First-order Markov assumption not exactly true in real world!
- Possible fixes:
  1. *Increase order* of Markov process
  2. *Augment state*, e.g., add $Temp_t$, $Pressure_t$
- Example: robot motion.
  - Augment position and velocity with $Battery_t$

Inference Tasks

- *Filtering*: $\mathbf{P}(\mathbf{X}_t|\mathbf{e}_{1:t})$
  - Compute a *belief state* given all evidence to date.

Inference Tasks

- *Filtering*: $\mathbf{P}(\mathbf{X}_t | \mathbf{e}_{1:t})$
  - Compute a *belief state* given all evidence to date.
- *Prediction*: $\mathbf{P}(\mathbf{X}_{t+k} | \mathbf{e}_{1:t})$ for $k > 0$
  - Compute the prob. of a *future state*, given all evidence to date.
  - Like filtering where some evidence is missing

Inference Tasks

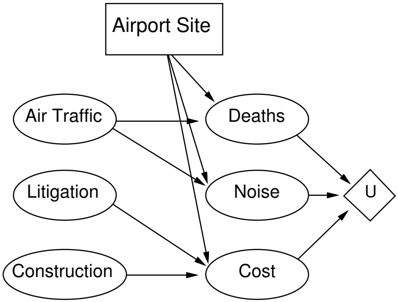
- *Filtering*: $\mathbf{P}(\mathbf{X}_t|\mathbf{e}_{1:t})$
  - Compute a *belief state* given all evidence to date.
- *Prediction*: $\mathbf{P}(\mathbf{X}_{t+k}|\mathbf{e}_{1:t})$ for $k > 0$
  - Compute the prob. of a *future state*, given all evidence to date.
  - Like filtering where some evidence is missing
- *Smoothing*: $\mathbf{P}(\mathbf{X}_k|\mathbf{e}_{1:t})$ for $0 \leq k < t$
  - Compute the prob. of a *past state*
  - Gives a better estimate of a state than was available at the time

Inference Tasks

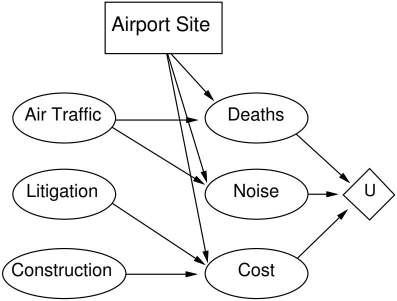
- *Filtering*: $\mathbf{P}(\mathbf{X}_t|\mathbf{e}_{1:t})$
  - Compute a *belief state* given all evidence to date.
- *Prediction*: $\mathbf{P}(\mathbf{X}_{t+k}|\mathbf{e}_{1:t})$ for $k > 0$
  - Compute the prob. of a *future state*, given all evidence to date.
  - Like filtering where some evidence is missing
- *Smoothing*: $\mathbf{P}(\mathbf{X}_k|\mathbf{e}_{1:t})$ for $0 \leq k < t$
  - Compute the prob. of a *past state*
  - Gives a better estimate of a state than was available at the time
- *Most likely explanation*: $\text{argmax}_{\mathbf{x}_{1:t}} P(\mathbf{x}_{1:t}|\mathbf{e}_{1:t})$
  - Given a sequence of observations, find the sequence of states most likely to have generated those observations
  - Speech recognition, decoding with a noisy channel

BN Extension: Decision Networks (Ch 16)

- Add *action nodes* and *utility nodes* to belief networks to enable rational decision making
- Action nodes are variables that are controlled by the user.
- The (single) utility node computes a utility, given its inputs.
- E.g.:

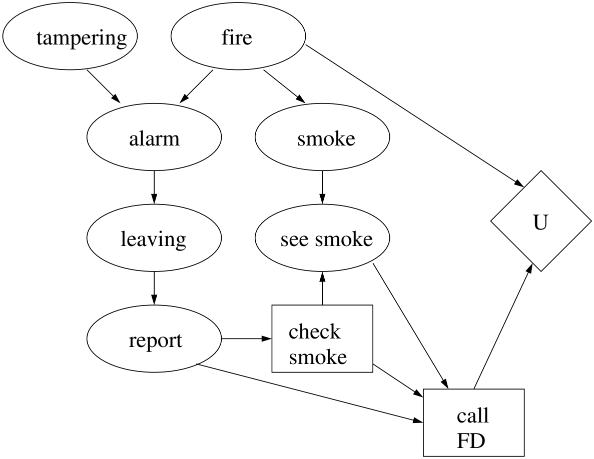


Decision Networks



- Algorithm:
  - For each value of action node:
    compute expected value of utility node given action, evidence
  - Return MEU action

Decision Networks: Another Example



Summary

- Bayes nets provide a natural representation for (causally induced) conditional independence
- Topology + CPTs = compact representation of joint distribution
- Generally easy for (non)experts to construct