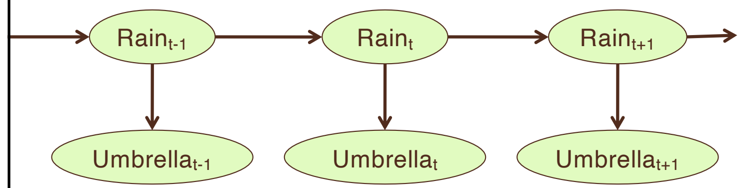
Tufts

Lecture 15:
Inference in DBNs

Artificial Intelligence (CS 131)

1

Review: DBN Transition Model



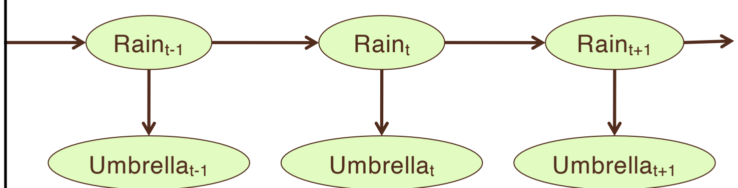
- Probability that system state will change over time
  - Stationary assumption: probabilities are the same for all times t
  - Markov assumption: state X_t is conditionally independent of all other states, given evidence about those that influence it (by the Markov assumption)
- For a 1st-order model, this means:

$$P(X_t \,|\, X_{0:t-1}) = P(X_t \,|\, X_{t-1})$$

$X_{0:t-1}$ abbreviates: $X_0, X_1, X_2, \ldots, X_{t-1}$

2

Review: DBN Observation Model



- Probability of observation, E_t, given current state, X_t
  - E_t is conditionally independent of everything else, given evidence about the local state:

$$P(E_t \,|\, X_{0:t}, E_{1:t-1}) = P(E_t \,|\, X_t)$$

3

Doing Forward Inference

- Both filtering and prediction can be treated as examples of the same basic recursive updating procedure
- Say we are at time t and get new observation, e_{t+1}
- We can determine the probability of the *next state*:

$$\begin{aligned} P(X_{t+1} \,|\, e_{1:t+1}) &= P(X_{t+1} \,|\, e_{1:t},\, e_{t+1}) \\ &= \alpha\, P(e_{t+1} \,|\, X_{t+1},\, e_{1:t})\, P(X_{t+1} \,|\, e_{1:t}) \\ &= \alpha\, P(e_{t+1} \,|\, X_{t+1})\, P(X_{t+1} \,|\, e_{1:t}) \end{aligned}$$

(We get 2nd line by Bayes' Rule, and 3rd by Markov Property)

Normalization constant: $\alpha = 1/P(e_{t+1} \,|\, e_{1:t})$

4

Components of Prediction/Filtering

- First part of equation is simply the *observation model*

$$P(X_{t+1} \,|\, e_{1:t+1}) = \alpha \boxed{P(e_{t+1} \,|\, X_{t+1})} P(X_{t+1} \,|\, e_{1:t})$$

- The second part is a *one-step prediction* of the state at time $t + 1$, given everything observed up to time t

$$P(X_{t+1} \,|\, e_{1:t+1}) = \alpha \, P(e_{t+1} \,|\, X_{t+1}) \boxed{P(X_{t+1} \,|\, e_{1:t})}$$

Computing the Necessary Prediction

- We condition on all possible values of current state (X_t) to get the prediction of the next state (X_{t+1}) :

$$\begin{aligned} P(X_{t+1} \,|\, e_{1:t+1}) &= \alpha \, P(e_{t+1} \,|\, X_{t+1}) \, P(X_{t+1} \,|\, e_{1:t}) \\ &= \alpha \, P(e_{t+1} \,|\, X_{t+1}) \sum_{x_t} P(X_{t+1} \,|\, x_t, e_{1:t}) \, P(x_t \,|\, e_{1:t}) \\ &= \alpha \, P(e_{t+1} \,|\, X_{t+1}) \sum_{x_t} P(X_{t+1} \,|\, x_t) \, P(x_t \,|\, e_{1:t}) \end{aligned}$$

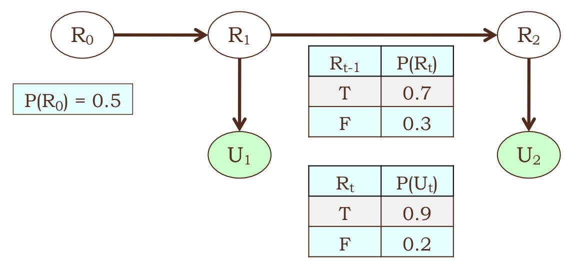
- (Where the simplification in the last line is a result of using the Markov Property again)

Belief Updates

$$\boxed{P(X_{t+1} \,|\, e_{1:t+1})} = \alpha \, P(e_{t+1} \,|\, X_{t+1}) \sum_{x_t} P(X_{t+1} \,|\, x_t) \boxed{P(x_t \,|\, e_{1:t})}$$

- Now, we simply have to plug in our *current belief* (our distribution over states at the last time-step, on the right), to give us our prediction (on the left)
- If we haven't computed and stored this belief already, then we would have a *recursive* situation
  - The formula at time t, to get probability of state at $(t + 1)$, is written using the *same* filtering equation, but using values for *prior state*
  - To get *that*, we can do the same thing…

An Example

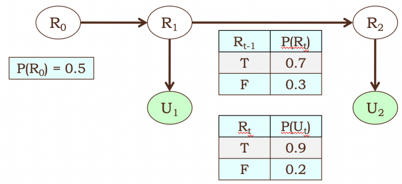


R_{t-1}	$P(R_t)$
T	0.7
F	0.3

R_t	$P(U_t)$
T	0.9
F	0.2

- We have Rain/Umbrella Network with an observation history where an umbrella is observed ($U_t = T$) for two days
- Want to know probability of rain on the last day ($R_2 = T$) given this observation-history

Umbrella Example, continued



R_{t-1}	$P(R_t)$
T	0.7
F	0.3

R_t	$P(U_t)$
T	0.9
F	0.2

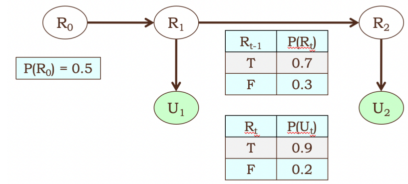
- Want probability of rain, given umbrellas on both days
- Can use the conditionalized version of Bayes' Rule (found in text as equation 12.13, page 399):

$$P(r_2 \mid u_1, u_2) = \frac{P(u_2 \mid r_2, u_1)\, P(r_2 \mid u_1)}{P(u_2 \mid u_1)}$$

$$= \alpha\, P(u_2 \mid r_2, u_1)\, P(r_2 \mid u_1)$$

9

Umbrella Example, continued



R_{t-1}	$P(R_t)$
T	0.7
F	0.3

R_t	$P(U_t)$
T	0.9
F	0.2

- Now, since the observation U_2 is independent of *everything else* given its parent state, R_2:

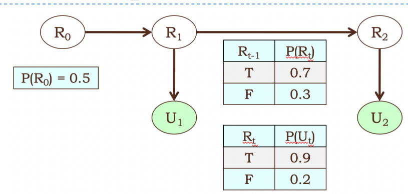
$$P(r_2 \mid u_1, u_2) = \alpha \boxed{P(u_2 \mid r_2, u_1)} P(r_2 \mid u_1)$$

$$= \alpha \boxed{P(u_2 \mid r_2)} P(r_2 \mid u_1)$$

$$= \alpha \boxed{0.9} P(r_2 \mid u_1)$$

10

Umbrella Example, continued



R_{t-1}	$P(R_t)$
T	0.7
F	0.3

R_t	$P(U_t)$
T	0.9
F	0.2

- We can then get the probability of rain at day 2, given an umbrella at day 1, by conditionalizing on prior rain values:

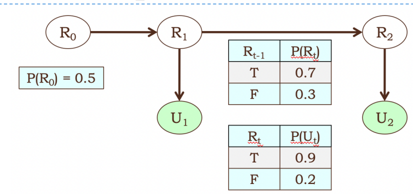
$$P(r_2 \mid u_1, u_2) = \alpha\, 0.9\, P(r_2 \mid u_1)$$

$$= \alpha\, 0.9 \sum_{r_1} P(r_2 \mid r_1, u_1)\, P(r_1 \mid u_1)$$

$$= \alpha\, 0.9\, (P(r_2 \mid r_1, u_1)\, P(r_1 \mid u_1) + P(r_2 \mid \neg r_1, u_1) P(\neg r_1 \mid u_1))$$

11

Umbrella Example, continued



R_{t-1}	$P(R_t)$
T	0.7
F	0.3

R_t	$P(U_t)$
T	0.9
F	0.2

- Again, the network structure tells us that R_2 is independent of *all prior information*, given the previous state, R_1:

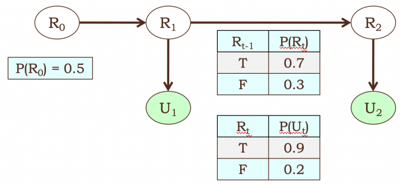
$$P(r_2 \mid u_1, u_2) = \alpha\, 0.9\, (\boxed{P(r_2 \mid r_1, u_1)} P(r_1 \mid u_1) + \boxed{P(r_2 \mid \neg r_1, u_1)} P(\neg r_1 \mid u_1))$$

$$= \alpha\, 0.9\, (\boxed{P(r_2 \mid r_1)} P(r_1 \mid u_1) + \boxed{P(r_2 \mid \neg r_1)} P(\neg r_1 \mid u_1))$$

$$= \alpha\, 0.9\, (\boxed{0.7} P(r_1 \mid u_1) + \boxed{0.3} P(\neg r_1 \mid u_1))$$

12

Umbrella Example, continued



R_{t-1}	$P(R_t)$
T	0.7
F	0.3

R_t	$P(U_t)$
T	0.9
F	0.2

- Now, we can use Bayes' Rule again to get $P(r_1 \mid u_1) = x$, and the other value, $P(\neg r_1 \mid u_1)$, is simply $(1 - x)$:

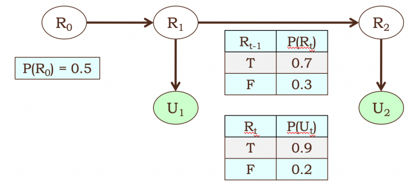
$$P(r_2 \mid u_1, u_2) = \alpha\, 0.9\, (0.7\, \boxed{P(r_1 \mid u_1)} + 0.3\, P(\neg r_1 \mid u_1))$$

$$P(r_1 \mid u_1) = \frac{P(u_1 \mid r_1)\, P(r_1)}{P(u_1)}$$

$$P(r_1 \mid u_1) = \frac{P(u_1 \mid r_1)\, P(r_1)}{P(u_1 \mid r_1)\, P(r_1) + P(u_1 \mid \neg r_1)\, P(\neg r_1)}$$

13

Umbrella Example, continued



R_{t-1}	$P(R_t)$
T	0.7
F	0.3

R_t	$P(U_t)$
T	0.9
F	0.2

- Last, we can get $P(r_1)$ by normalizing again, using all possible values of the previous state, R_0:

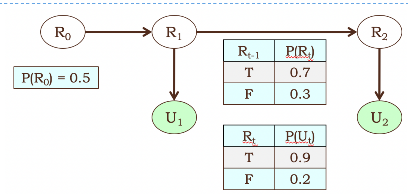
$$P(r_1 \mid u_1) = \frac{P(u_1 \mid r_1)\, P(r_1)}{P(u_1 \mid r_1)\, P(r_1) + P(u_1 \mid \neg r_1)\, P(\neg r_1)}$$

$$P(r_1 \mid u_1) = \frac{0.9\, P(r_1)}{0.9\, P(r_1) + 0.2\, P(\neg r_1)} = \frac{0.9\, \boxed{P(r_1)}}{0.9\, P(r_1) + 0.2\, (1 - P(r_1))}$$

$$\boxed{P(r_1)} = \sum_{r_0} P(r_1 \mid r_0)\, P(r_0) = P(r_1 \mid r_0)\, P(r_0) + P(r_1 \mid \neg r_0)\, P(\neg r_0)$$

14

Umbrella Example, continued



R_{t-1}	$P(R_t)$
T	0.7
F	0.3

R_t	$P(U_t)$
T	0.9
F	0.2

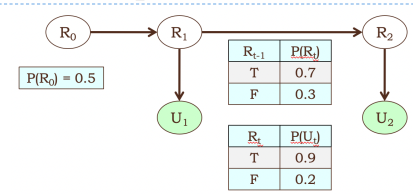
- Now we solve for $P(r_1)$, again by conditionalizing:

$$P(r_1) = P(r_1 \mid r_0)\, P(r_0) + P(r_1 \mid \neg r_0)\, P(\neg r_0)$$

$$P(r_1) = (0.7 \times 0.5) + (0.3 \times 0.5) = 0.5$$

15

Umbrella Example, continued



R_{t-1}	$P(R_t)$
T	0.7
F	0.3

R_t	$P(U_t)$
T	0.9
F	0.2

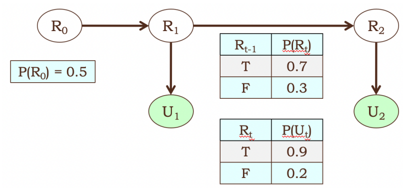
- And plug in that value to solve for $P(r_1 \mid u_1)$:

$$P(r_1 \mid u_1) = \frac{0.9\, P(r_1)}{0.9\, P(r_1) + 0.2\, (1 - P(r_1))}$$

$$= \frac{0.9 \times 0.5}{0.9 \times 0.5 + 0.2 \times 0.5} = 0.818$$

16

Umbrella Example, continued



- Then plug in *that* number to solve for $P(r_2 \mid u_1,\ u_2)$:

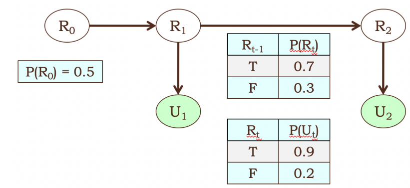
$$P(r_2 \mid u_1, u_2) = \alpha\, 0.9\, (0.7\, P(r_1 \mid u_1) + 0.3\, P(\neg r_1 \mid u_1))$$

$$= \alpha\, 0.9\, (0.7\, P(r_1 \mid u_1) + 0.3\, (1 - P(r_1 \mid u_1)))$$

$$= \alpha\, 0.9\, ((0.7 \times 0.818) + (0.3 \times 0.182)) = \alpha\, 0.565$$

17

Umbrella Example, continued



- Finally, solve for normalization constant α (in basically the same way, details skipped here) to get the final answer:

$$\alpha = \frac{1}{P(u_2 \mid u_1)} = \frac{1}{0.64}$$

$$P(r_2 \mid u_1, u_2) = \alpha\, 0.565 = \frac{0.565}{0.64} \approx 0.883$$

18

Recursion in Filtering/Prediction

- Here, for sake of example, we worked out the full calculation from the most recent observation, e_{t+1}, all the way back to the initial state distribution, $P(X_0)$
- In doing so, clearly, the amount of computation required is directly proportional to the size of the BN structure at any stage of the problem, ***multiplied by*** the length of the time horizon $(t + 1)$
- To avoid requiring possibly unbounded memory, however, we use the basic definition of the update:

$$\boxed{P(X_{t+1} \mid e_{1:t+1})} = \alpha\, P(e_{t+1} \mid X_{t+1}) \sum_{x_t} P(X_{t+1} \mid x_t) \boxed{P(x_t \mid e_{1:t})}$$

19

Continual Updates for Filtering/Prediction

$$\boxed{P(X_{t+1} \mid e_{1:t+1})} = \alpha\, P(e_{t+1} \mid X_{t+1}) \sum_{x_t} P(X_{t+1} \mid x_t) \boxed{P(x_t \mid e_{1:t})}$$

- The update for time $(t + 1)$ simply requires the results of the same calculation for the prior step
- Thus, we *retain* this belief-state as we move through time, updating it at each step, and doing all of our filtering and one-step prediction in a constant amount of memory space and processing time

20

Complexity of Temporal Inference

- In general, for basic updating via filtering or one-step prediction, we get a linear amount ($O(n)$) of work in the size of the overall one-stage network
- For backwards inference, this will also be linear in the time-horizon, $O(t)$, for a total of $O(n * t)$
  - Even very long-range updates can be done relatively efficiently
- For more complex inference problems, like the *most likely event-sequence*, the algorithms are more complex, but involve the same basic manipulations, and can also be made linear in the length of the sequence