Informatics 2D: Reasoning and Agents

Alex Lascarides

School of Informatics
The University of Edinburgh

Lecture 27: Time and Uncertainty II

Where are we?

Last time ...

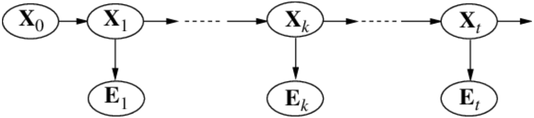
- Time in reasoning about uncertainty
- Markov assumption, stationarity
- Algorithms for reasoning about temporal processes
- Filtering and prediction

Today ...

- **Time and uncertainty II**

Smoothing

- Smoothing is computation of distribution of past states given current evidence, i.e. $\mathbf{P}(\mathbf{X}_k|\mathbf{e}_{1:t})$, $1 \leq k < t$



- Easiest to view as 2-step process (up to k, then $k+1$ to t)

$$\begin{aligned}
\mathbf{P}(\mathbf{X}_k|\mathbf{e}_{1:t}) &= \mathbf{P}(\mathbf{X}_k|\mathbf{e}_{1:k},\mathbf{e}_{k+1:t}) && \text{(split notation)}\\
&= \alpha\mathbf{P}(\mathbf{X}_k|\mathbf{e}_{1:k})\mathbf{P}(\mathbf{e}_{k+1:t}|\mathbf{X}_k,\mathbf{e}_{1:k}) && \text{(Bayes)}\\
&= \alpha\mathbf{P}(\mathbf{X}_k|\mathbf{e}_{1:k})\mathbf{P}(\mathbf{e}_{k+1:t}|\mathbf{X}_k) && \text{(conditional independence)}\\
&= \alpha\mathbf{f}_{1:k}\mathbf{b}_{k+1:t}
\end{aligned}$$

- Here "backward" message is $\mathbf{b}_{k+1:t} = \mathbf{P}(\mathbf{e}_{k+1:t}|\mathbf{X}_k)$ analogous to forward message

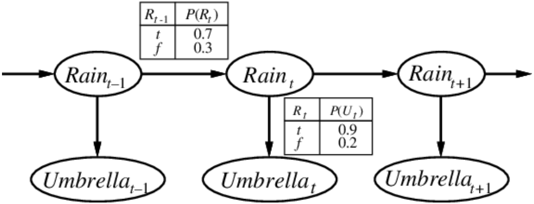
Smoothing

- Formula for backward message:

$$\mathrm{P}(\mathrm{e}_{k+1:t}|\mathrm{X}_k) = \sum_{\mathrm{x}_{k+1}} P(\mathrm{e}_{k+1}|\mathrm{x}_{k+1}) P(\mathrm{e}_{k+2:t}|\mathrm{x}_{k+1}) \mathrm{P}(\mathrm{x}_{k+1}|\mathrm{X}_k)$$

  (I'll show this is true shortly)
- First term is sensor model; third term is transition model; second is 'recursive call'
- Define $\mathrm{b}_{k+1:t} = \mathrm{Backward}(\mathrm{b}_{k+2:t}, \mathrm{e}_{k+1:t})$
- The backward phase has to be initialised with $\mathrm{b}_{t+1:t} = \mathrm{P}(\mathrm{e}_{t+1:t}|\mathrm{X}_t) = 1$ (a vector of 1s) because probability of observing empty sequence is 1
- As before, all this is quite abstract, back to our example

Umbrella World: Compute $\mathbf{P}(R_1|u_1,u_2)$



We have $\mathbf{P}(R_1|u_1,u_2) = \alpha\mathbf{P}(R_1|u_1)\mathbf{P}(u_2|R_1)$
So we'll need to remind ourselves of $\mathbf{P}(R_1|u_1)$ from last lecture:

- $\mathbf{P}(R_1) = \sum_{r_0} \mathbf{P}(R_1|r_0)P(r_0) = \langle 0.7, 0.3\rangle \times 0.5 + \langle 0.3, 0.7\rangle \times 0.5 = \langle 0.5, 0.5\rangle$
- Update with evidence $U_1 = true$ yields:

$$\mathbf{P}(R_1|u_1) = \alpha\mathbf{P}(u_1|R_1)\mathbf{P}(R_1) = \alpha\langle 0.9, 0.2\rangle\langle 0.5, 0.5\rangle \approx \langle 0.818, 0.182\rangle$$

Smoothing Example Continued

$\mathrm{P}(R_1|u_1, u_2) = \alpha \mathrm{P}(R_1|u_1)\mathrm{P}(u_2|R_1)$

- Forward filtering process yielded $\langle 0.818, 0.182\rangle$ for first term
- The second term can be obtained through backward recursion:

$$\mathrm{P}(u_2|R_1) = \sum_{r_2} P(u_2|r_2)P(|r_2)\mathrm{P}(r_2|R_1)$$
$$= (0.9 \times 1 \times \langle 0.7, 0.3\rangle) + (0.2 \times 1 \times \langle 0.3, 0.7\rangle) = \langle 0.69, 0.41\rangle$$

- Plugged into the above equation this yields

$$\mathrm{P}(R_1|u_1, u_2) = \alpha\langle 0.818, 0.182\rangle \times \langle 0.69, 0.41\rangle \approx \langle 0.883, 0.117\rangle$$

- So our confidence that it rained on Day 1 increases when we see the umbrella on the second day as well as the first.
- A simple improved version of this that stores results runs in linear time (**forward-backward** algorithm)

Deriving the backward message

$$\mathbf{P}(\mathbf{e}_{k+1:t}|\mathbf{X}_k) = \sum_{\mathbf{x}_{k+1}} P(\mathbf{e}_{k+1}|\mathbf{x}_{k+1})P(\mathbf{e}_{k+2:t}|\mathbf{x}_{k+1})\mathbf{P}(\mathbf{x}_{k+1}|\mathbf{X}_k)$$

$$\mathbf{P}(\mathbf{e}_{k+1:t}|\mathbf{X}_k) = \sum_{\mathbf{x}_{k+1}} \mathbf{P}(\mathbf{e}_{k+1:t}, x_{k+1}|\mathbf{X}_k) \quad \text{(marginalisation)}$$

$$= \sum_{\mathbf{x}_{k+1}} \mathbf{P}(\mathbf{e}_{k+1}, \mathbf{e}_{k+2:t}, x_{k+1}|\mathbf{X}_k) \quad \text{(split notation)}$$

$$= \sum_{\mathbf{x}_{k+1}} P(\mathbf{e}_{k+1}|\mathbf{e}_{k+2:t}, x_{k+1}, \mathbf{X}_k)\mathbf{P}(\mathbf{e}_{k+2:t}, x_{k+1}|\mathbf{X}_k) \quad \text{(Bayes)}$$

$$= \sum_{\mathbf{x}_{k+1}} P(\mathbf{e}_{k+1}|x_{k+1})\mathbf{P}(\mathbf{e}_{k+2:t}, x_{k+1}|\mathbf{X}_k) \quad \text{(independence)}$$

$$= \sum_{\mathbf{x}_{k+1}} P(\mathbf{e}_{k+1}|x_{k+1})\mathbf{P}(\mathbf{e}_{k+2:t}|x_{k+1}, \mathbf{X}_k)\mathbf{P}(\mathbf{x}_{k+1}|\mathbf{X}_k) \quad \text{(Bayes)}$$

$$= \sum_{\mathbf{x}_{k+1}} P(\mathbf{e}_{k+1}|x_{k+1})\mathbf{P}(\mathbf{e}_{k+2:t}|x_{k+1})\mathbf{P}(\mathbf{x}_{k+1}|\mathbf{X}_k) \quad \text{(Bayes)}$$

Deriving the backward message

$$\mathbf{P}(\mathbf{e}_{k+1:t}|\mathbf{X}_k) = \sum_{\mathbf{x}_{k+1}} P(\mathbf{e}_{k+1}|\mathbf{x}_{k+1})P(\mathbf{e}_{k+2:t}|\mathbf{x}_{k+1})\mathbf{P}(\mathbf{x}_{k+1}|\mathbf{X}_k)$$

$$\mathbf{P}(\mathbf{e}_{k+1:t}|\mathbf{X}_k) = \sum_{\mathbf{x}_{k+1}} \mathbf{P}(\mathbf{e}_{k+1:t}, x_{k+1}|\mathbf{X}_k) \qquad \text{(marginalisation)}$$

$$= \sum_{\mathbf{x}_{k+1}} \mathbf{P}(\mathbf{e}_{k+1}, \mathbf{e}_{k+2:t}, x_{k+1}|\mathbf{X}_k) \qquad \text{(split notation)}$$

$$= \sum_{\mathbf{x}_{k+1}} P(\mathbf{e}_{k+1}|\mathbf{e}_{k+2:t}, x_{k+1}, \mathbf{X}_k)\mathbf{P}(\mathbf{e}_{k+2:t}, x_{k+1}|\mathbf{X}_k) \qquad \text{(Bayes)}$$

$$= \sum_{\mathbf{x}_{k+1}} P(\mathbf{e}_{k+1}|x_{k+1})\mathbf{P}(\mathbf{e}_{k+2:t}, x_{k+1}|\mathbf{X}_k) \qquad \text{(independence)}$$

$$= \sum_{\mathbf{x}_{k+1}} P(\mathbf{e}_{k+1}|x_{k+1})\mathbf{P}(\mathbf{e}_{k+2:t}|x_{k+1}, \mathbf{X}_k)\mathbf{P}(\mathbf{x}_{k+1}|\mathbf{X}_k) \qquad \text{(Bayes)}$$

$$= \sum_{\mathbf{x}_{k+1}} P(\mathbf{e}_{k+1}|x_{k+1})\mathbf{P}(\mathbf{e}_{k+2:t}|x_{k+1})\mathbf{P}(\mathbf{x}_{k+1}|\mathbf{X}_k) \qquad \text{(Bayes)}$$

Deriving the backward message

$$P(e_{k+1:t}|X_k) = \sum_{x_{k+1}} P(e_{k+1}|x_{k+1})P(e_{k+2:t}|x_{k+1})P(x_{k+1}|X_k)$$

$$P(e_{k+1:t}|X_k) = \sum_{x_{k+1}} P(e_{k+1:t}, x_{k+1}|X_k) \quad \text{(marginalisation)}$$

$$= \sum_{x_{k+1}} P(e_{k+1}, e_{k+2:t}, x_{k+1}|X_k) \quad \text{(split notation)}$$

$$= \sum_{x_{k+1}} P(e_{k+1}|e_{k+2:t}, x_{k+1}, X_k)P(e_{k+2:t}, x_{k+1}|X_k) \quad \text{(Bayes)}$$

$$= \sum_{x_{k+1}} P(e_{k+1}|x_{k+1})P(e_{k+2:t}, x_{k+1}|X_k) \quad \text{(independence)}$$

$$= \sum_{x_{k+1}} P(e_{k+1}|x_{k+1})P(e_{k+2:t}|x_{k+1}, X_k)P(x_{k+1}|X_k) \quad \text{(Bayes)}$$

$$= \sum_{x_{k+1}} P(e_{k+1}|x_{k+1})P(e_{k+2:t}|x_{k+1})P(x_{k+1}|X_k) \quad \text{(Bayes)}$$

Deriving the backward message

$$\mathbf{P}(\mathbf{e}_{k+1:t}|\mathbf{X}_k) = \sum_{\mathbf{x}_{k+1}} P(\mathbf{e}_{k+1}|\mathbf{x}_{k+1})P(\mathbf{e}_{k+2:t}|\mathbf{x}_{k+1})\mathbf{P}(\mathbf{x}_{k+1}|\mathbf{X}_k)$$

$$\begin{aligned}
\mathbf{P}(\mathbf{e}_{k+1:t}|\mathbf{X}_k) &= \sum_{\mathbf{x}_{k+1}} \mathbf{P}(\mathbf{e}_{k+1:t}, x_{k+1}|\mathbf{X}_k) && \text{(marginalisation)} \\
&= \sum_{\mathbf{x}_{k+1}} \mathbf{P}(\mathbf{e}_{k+1}, \mathbf{e}_{k+2:t}, x_{k+1}|\mathbf{X}_k) && \text{(split notation)} \\
&= \sum_{\mathbf{x}_{k+1}} P(\mathbf{e}_{k+1}|\mathbf{e}_{k+2:t}, x_{k+1}, \mathbf{X}_k)\mathbf{P}(\mathbf{e}_{k+2:t}, x_{k+1}|\mathbf{X}_k) && \text{(Bayes)} \\
&= \sum_{\mathbf{x}_{k+1}} P(\mathbf{e}_{k+1}|x_{k+1})\mathbf{P}(\mathbf{e}_{k+2:t}, x_{k+1}|\mathbf{X}_k) && \text{(independence)} \\
&= \sum_{\mathbf{x}_{k+1}} P(\mathbf{e}_{k+1}|x_{k+1})\mathbf{P}(\mathbf{e}_{k+2:t}|x_{k+1}, \mathbf{X}_k)\mathbf{P}(\mathbf{x}_{k+1}|\mathbf{X}_k) && \text{(Bayes)} \\
&= \sum_{\mathbf{x}_{k+1}} P(\mathbf{e}_{k+1}|x_{k+1})\mathbf{P}(\mathbf{e}_{k+2:t}|x_{k+1})\mathbf{P}(\mathbf{x}_{k+1}|\mathbf{X}_k) && \text{(Bayes)}
\end{aligned}$$

Deriving the backward message

$$\mathbf{P}(\mathbf{e}_{k+1:t}|\mathbf{X}_k) = \sum_{\mathbf{x}_{k+1}} P(\mathbf{e}_{k+1}|\mathbf{x}_{k+1})P(\mathbf{e}_{k+2:t}|\mathbf{x}_{k+1})\mathbf{P}(\mathbf{x}_{k+1}|\mathbf{X}_k)$$

$$\begin{aligned}
\mathbf{P}(\mathbf{e}_{k+1:t}|\mathbf{X}_k) &= \sum_{\mathbf{x}_{k+1}} \mathbf{P}(\mathbf{e}_{k+1:t}, x_{k+1}|\mathbf{X}_k) && \text{(marginalisation)} \\
&= \sum_{\mathbf{x}_{k+1}} \mathbf{P}(\mathbf{e}_{k+1}, \mathbf{e}_{k+2:t}, x_{k+1}|\mathbf{X}_k) && \text{(split notation)} \\
&= \sum_{\mathbf{x}_{k+1}} P(\mathbf{e}_{k+1}|\mathbf{e}_{k+2:t}, x_{k+1}, \mathbf{X}_k)\mathbf{P}(\mathbf{e}_{k+2:t}, x_{k+1}|\mathbf{X}_k) && \text{(Bayes)} \\
&= \sum_{\mathbf{x}_{k+1}} P(\mathbf{e}_{k+1}|x_{k+1})\mathbf{P}(\mathbf{e}_{k+2:t}, x_{k+1}|\mathbf{X}_k) && \text{(independence)} \\
&= \sum_{\mathbf{x}_{k+1}} P(\mathbf{e}_{k+1}|x_{k+1})\mathbf{P}(\mathbf{e}_{k+2:t}|x_{k+1}, \mathbf{X}_k)\mathbf{P}(\mathbf{x}_{k+1}|\mathbf{X}_k) && \text{(Bayes)} \\
&= \sum_{\mathbf{x}_{k+1}} P(\mathbf{e}_{k+1}|x_{k+1})\mathbf{P}(\mathbf{e}_{k+2:t}|x_{k+1})\mathbf{P}(\mathbf{x}_{k+1}|\mathbf{X}_k) && \text{(Bayes)}
\end{aligned}$$

Deriving the backward message

$$\mathbf{P}(\mathbf{e}_{k+1:t}|\mathbf{X}_k) = \sum_{\mathbf{x}_{k+1}} P(\mathbf{e}_{k+1}|\mathbf{x}_{k+1})P(\mathbf{e}_{k+2:t}|\mathbf{x}_{k+1})\mathbf{P}(\mathbf{x}_{k+1}|\mathbf{X}_k)$$

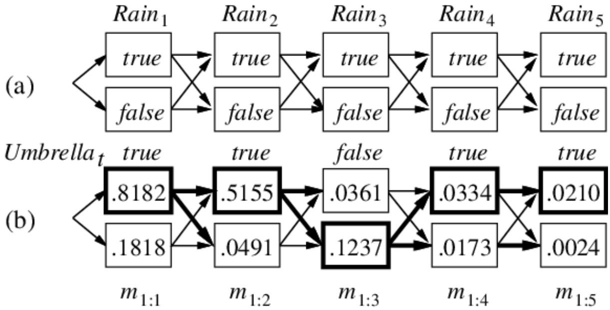
$$\begin{aligned}
\mathbf{P}(\mathbf{e}_{k+1:t}|\mathbf{X}_k) &= \sum_{\mathbf{x}_{k+1}} \mathbf{P}(\mathbf{e}_{k+1:t}, x_{k+1}|\mathbf{X}_k) && \text{(marginalisation)} \\
&= \sum_{\mathbf{x}_{k+1}} \mathbf{P}(\mathbf{e}_{k+1}, \mathbf{e}_{k+2:t}, x_{k+1}|\mathbf{X}_k) && \text{(split notation)} \\
&= \sum_{\mathbf{x}_{k+1}} P(\mathbf{e}_{k+1}|\mathbf{e}_{k+2:t}, x_{k+1}, \mathbf{X}_k)\mathbf{P}(\mathbf{e}_{k+2:t}, x_{k+1}|\mathbf{X}_k) && \text{(Bayes)} \\
&= \sum_{\mathbf{x}_{k+1}} P(\mathbf{e}_{k+1}|x_{k+1})\mathbf{P}(\mathbf{e}_{k+2:t}, x_{k+1}|\mathbf{X}_k) && \text{(independence)} \\
&= \sum_{\mathbf{x}_{k+1}} P(\mathbf{e}_{k+1}|x_{k+1})\mathbf{P}(\mathbf{e}_{k+2:t}|x_{k+1}, \mathbf{X}_k)\mathbf{P}(\mathbf{x}_{k+1}|\mathbf{X}_k) && \text{(Bayes)} \\
&= \sum_{\mathbf{x}_{k+1}} P(\mathbf{e}_{k+1}|x_{k+1})\mathbf{P}(\mathbf{e}_{k+2:t}|x_{k+1})\mathbf{P}(\mathbf{x}_{k+1}|\mathbf{X}_k) && \text{(Bayes)}
\end{aligned}$$

Finding the most likely sequence

- Suppose [*true*, *true*, *false*, *true*, *true*] is the umbrella sequence for first five days, what is the most likely weather sequence that caused it?
- Could we use smoothing procedure to find posterior distribution for weather at each step and then use most likely weather at each step to construct sequence?
- **NO!** Smoothing considers distributions over individual time steps, but we must consider **joint** probabilities over all time steps
- Actual algorithm is based on viewing each sequence as path through a graph (nodes=states at each time step)

Finding the most likely sequence

- In umbrella example:



- Look at states with $Rain_5 = true$ (part (a)), Markov property
  - most likely path to this state consists of most likely path to state at time 4 followed by transition to $Rain_5 = true$
  - state at time 4 that will become part of the path is whichever maximises likelihood of the path

Finding the most likely sequence

- There is a recursive relationship between most likely paths to x_{t+1} and most likely paths to each state x_t

$$\max_{x_1 \ldots x_t} P(x_1, \ldots, x_t, X_{t+1} | e_{1:t+1})$$
$$= \alpha P(e_{t+1} | X_{t+1}) \max_{x_t} (P(X_{t+1} | x_t) \max_{x_1 \ldots x_{t-1}} P(x_1, \ldots, x_{t-1}, x_t | e_{1:t}))$$

- This is like filtering only that the forward message is replaced by
$$m_{1:t} = \max_{x_1 \ldots x_{t-1}} P(x_1, \ldots, x_{t-1}, X_t | e_{1:t})$$
- And summing (marginalisation) is now replaced by maximisation

Finding the most likely sequence

- This algorithm (**Viterbi algorithm**) is similar to filtering
- Runs forward along sequence computing m message in each step
- Progress in example shown in part (b) of diagram above
- In the end it has probability for most likely sequence for reaching each final state
  Easy to determine overall most likely sequence
- Has to keep pointers from each state back to the best state that leads to it

Hidden Markov Models

- So far, we have seen a general model for temporal probabilistic reasoning (independent of transition/sensor models)
- In this and the following lecture we are going to look at more concrete models and applications
- **Hidden Markov Models (HMMs)**: temporal probabilistic model in which state of the process is described by a single variable
- Like our umbrella example (single variable $Rain_t$)
- More than one variable can be accommodated, but only by combining them into a single "mega-variable"
- Structure of HMMs allows for a very simple and elegant matrix implementation of basic algorithms

Summary

- The forward-backward algorithm
- Finding the most likely sequence (Viterbi algorithm)
- Talked about HMMs
- HMMs: single state variable, simplifies algorithms (see other courses for these)
- Huge significance, for example in speech recognition:

$$P(words|signal) = \alpha P(signal|words)P(words)$$

- Vast array of applications, but also limits.
- Next time: **Dynamic Bayesian Networks**