Intelligent Systems (AI-2)

Computer Science cpsc422, Lecture 15

Oct, 14, 2016

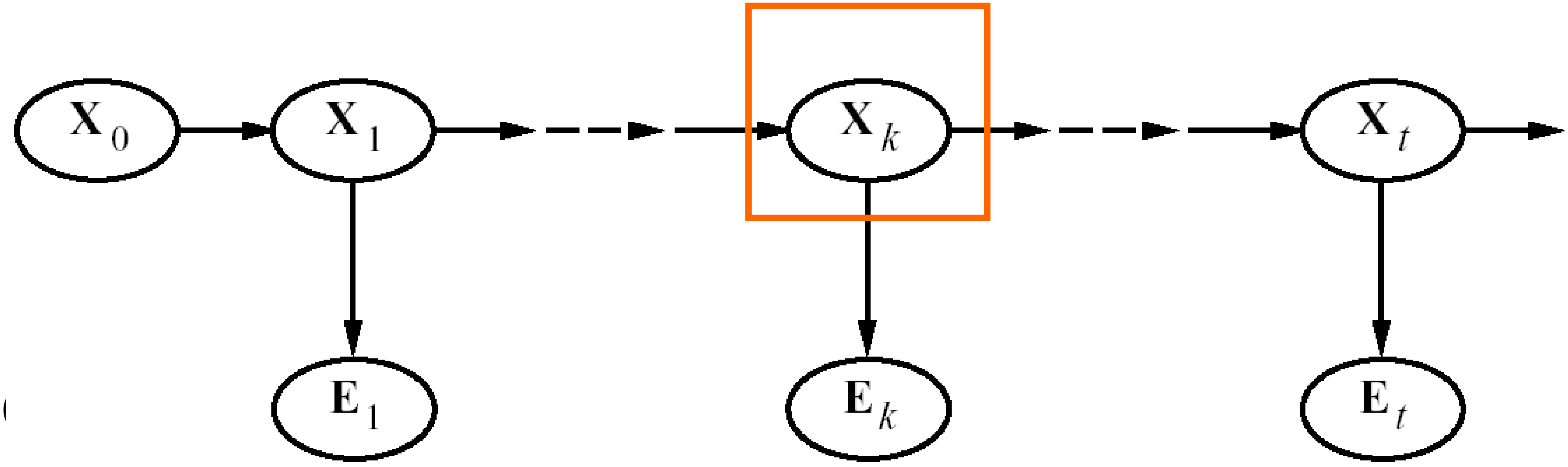
Lecture Overview

<u>Probabilistic temporal Inferences</u>

- Filtering
- Prediction
- Smoothing (forward-backward)
- Most Likely Sequence of States (Viterbi)

Smoothing

- ***Smoothing***: Compute the posterior distribution over a *past* state given all evidence to date
  - $P(X_k \,/\, e_{0:t})$ for $1 \leq k < t$



- ***To revise your estimates in the past based on more recent evidence***

Smoothing

- $\boldsymbol{P(X_k \mid e_{0:t})} = \boldsymbol{P(X_k \mid e_{0:k}, e_{k+1:t})}$ dividing up the evidence

$= \alpha\, \boldsymbol{P(X_k \mid e_{0:k})}\, \boldsymbol{P(e_{k+1:t} \mid X_k, e_{0:k})}$ using…

$= \alpha\, \boldsymbol{P(X_k \mid e_{0:k})}\, \boldsymbol{P(e_{k+1:t} \mid X_k)}$ using…

iclicker.

A. Bayes Rule

B. Cond. Independence

C. Product Rule

forward message from filtering up to state k, $\boldsymbol{f}_{0:k}$

backward message, $\boldsymbol{b}_{k+1:t}$ computed by a recursive process that runs backwards from *t*

Smoothing

- $\boldsymbol{P}(\boldsymbol{X}_k \mid \boldsymbol{e}_{0:t}) = \boldsymbol{P}(\boldsymbol{X}_k \mid \boldsymbol{e}_{0:k}, \boldsymbol{e}_{k+1:t})$ dividing up the evidence

  $= \alpha\, \boldsymbol{P}(\boldsymbol{X}_k \mid \boldsymbol{e}_{0:k})\, \boldsymbol{P}(\boldsymbol{e}_{k+1:t} \mid \boldsymbol{X}_k, \boldsymbol{e}_{0:k})$ using Bayes Rule

  $= \alpha\, \boldsymbol{P}(\boldsymbol{X}_k \mid \boldsymbol{e}_{0:k})\, \boldsymbol{P}(\boldsymbol{e}_{k+1:t} \mid \boldsymbol{X}_k)$ By Markov assumption on evidence

forward message from filtering up to state k, $\boldsymbol{f}_{0:k}$

backward message, $\boldsymbol{b}_{k+1:t}$ computed by a recursive process that runs backwards from *t*

Backward Message

Product Rule

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

$$= \sum_{x_{k+1}} \mathbf{P}(\mathbf{e}_{k+1:t} \mid \mathbf{x}_{k+1})\, \mathbf{P}(\mathbf{x}_{k+1} \mid \mathbf{X}_k) \text{ by Markov assumption on evidence}$$

Product Rule

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

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

because $\mathbf{e}_{k+1}$ and $\mathbf{e}_{k+2:t}$, are conditionally independent given $\mathbf{x}_{k+1}$

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

sensor model

recursive call

transition model

- In message notation

$$\mathbf{b}_{k+1:t} = \text{BACKWARD}(\mathbf{b}_{k+2:t}, \mathbf{e}_{k+1})$$

More Intuitive Interpretation (Example with three states)

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

$X = \{S_1, S_2, S_3\}$

e_K $\quad$ e_{K+1} $\quad$ e_{K+2} . . . . e_t

X_K $\quad$ X_{K+1}

S_1 $\quad$ . . . . $\quad$ $P(e_{K+2:t} \mid S_1)$

S_2 $\quad$ $P(e_{K+1:t} \mid S_2)$ $\quad$ $P(e_{K+2:t} \mid S_2)$

S_3 $\quad$ . . . . $\quad$ $P(e_{K+2:t} \mid S_3)$

Forward–Backward Procedure

- Thus,
  - $P(X_k \mid e_{0:t}) = \alpha\, f_{0:k}\, b_{k+1:t}$

and this value can be computed by recursion through time, running forward from 0 to k and backwards from t to $k+1$

$$f_{0:1} \quad f_{0:2} \quad \cdots \quad f_{0:k} * b_{k+1:t} \quad b_{k+2:t} \quad \cdots \quad b_{t+1:t}$$

direction of computation

How is it Backward initialized?

$$f_{0:1} \quad f_{0:2} \cdots f_{0:k} * b_{k+1:t} \quad b_{k+2:t} \cdots b_{t+1:t}$$

direction of computation

- The backwards phase is initialized with making an *unspecified* observation $\boldsymbol{e}_{t+1}$ at $t+1$……

$$\boldsymbol{b}_{t+1:t} = \mathrm{P}(\boldsymbol{e}_{t+1} / X_t) = \mathrm{P}(\textit{unspecified} \,/\, X_t) = ?$$

A. 0	B. 0.5	C. 1

How is it Backward initialized?

- The backwards phase is initialized with making an unspecified observation $\boldsymbol{e}_{t+1}$ at t+ 1……

  $\boldsymbol{b}_{t+1:t} = \mathrm{P}(\boldsymbol{e}_{t+1} / \boldsymbol{X}_t) = \mathrm{P}(\textit{unspecified} \mid \boldsymbol{X}_t) = 1$

- You will observe something for sure! It is only when you put some constraints on the observations that the probability becomes less than 1

Rain Example

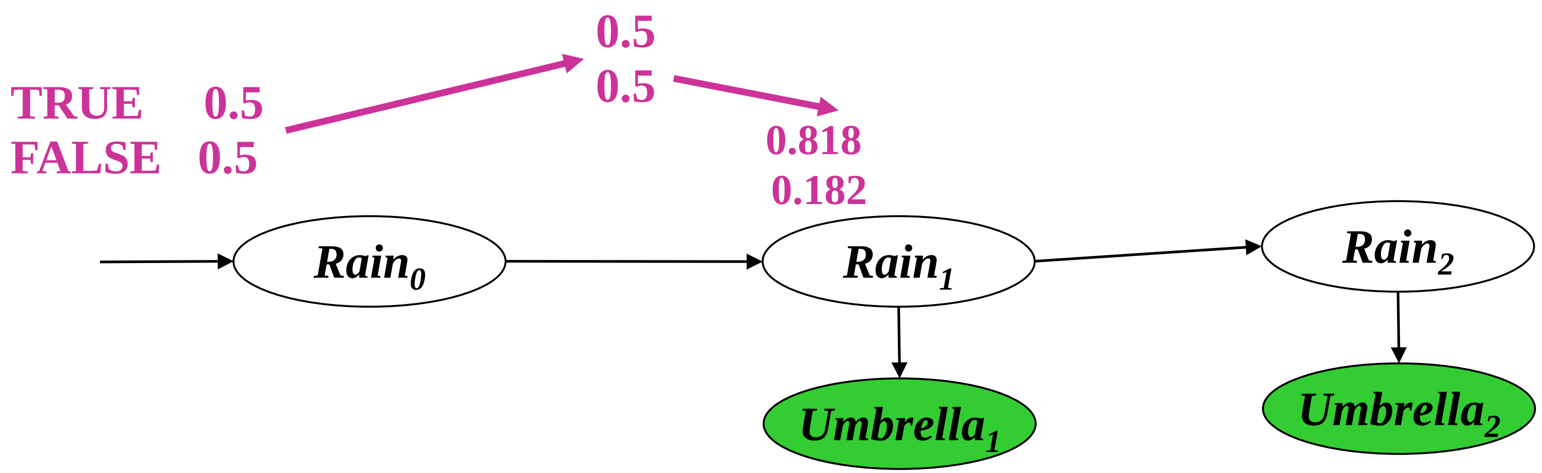
- Let's compute the probability of rain at t = 1, given umbrella observations at t=1 and t =2
- From $\boldsymbol{P}(\boldsymbol{X}_k \mid \boldsymbol{e}_{1:t}) = \alpha\, \boldsymbol{P}(\boldsymbol{X}_k \mid \boldsymbol{e}_{1:k})\, \boldsymbol{P}(\boldsymbol{e}_{k+1:t} \mid \boldsymbol{X}_k)$ we have

$$\boldsymbol{P}(R_1 \mid \boldsymbol{e}_{1:2}) = \boldsymbol{P}(R_1 \mid u_1, u_2) = \alpha\, \boldsymbol{P}(R_1 \mid u_1)\, \boldsymbol{P}(u_2 \mid R_1)$$

forward message from filtering up to state 1

backward message for propagating evidence backward from time 2

- $\boldsymbol{P}(R_1 \mid u_1) = \langle 0.818, 0.182 \rangle$ as it is the filtering to $t = 1$ that we did in lecture 14



Rain Example

- From $\mathbf{P}(\mathbf{e}_{k+1:t} \mid \mathbf{X}_k) = \sum_{x_{k+1}} P(\mathbf{e}_{k+1}|\mathbf{x}_{k+1})\, P(\mathbf{e}_{k+2:t}\, |\mathbf{x}_{k+1})\, \mathbf{P}(\mathbf{x}_{k+1} \mid \mathbf{X}_k)$
- $\mathbf{P}(u_2 \mid R_1) = \sum_{r \in r_2, \neg r_2} P(u_2|r)\, P(|r)\, P(r \mid R_1) =$

Term corresponding to the Fictitious unspecified observation sequence $e_{3:2}$

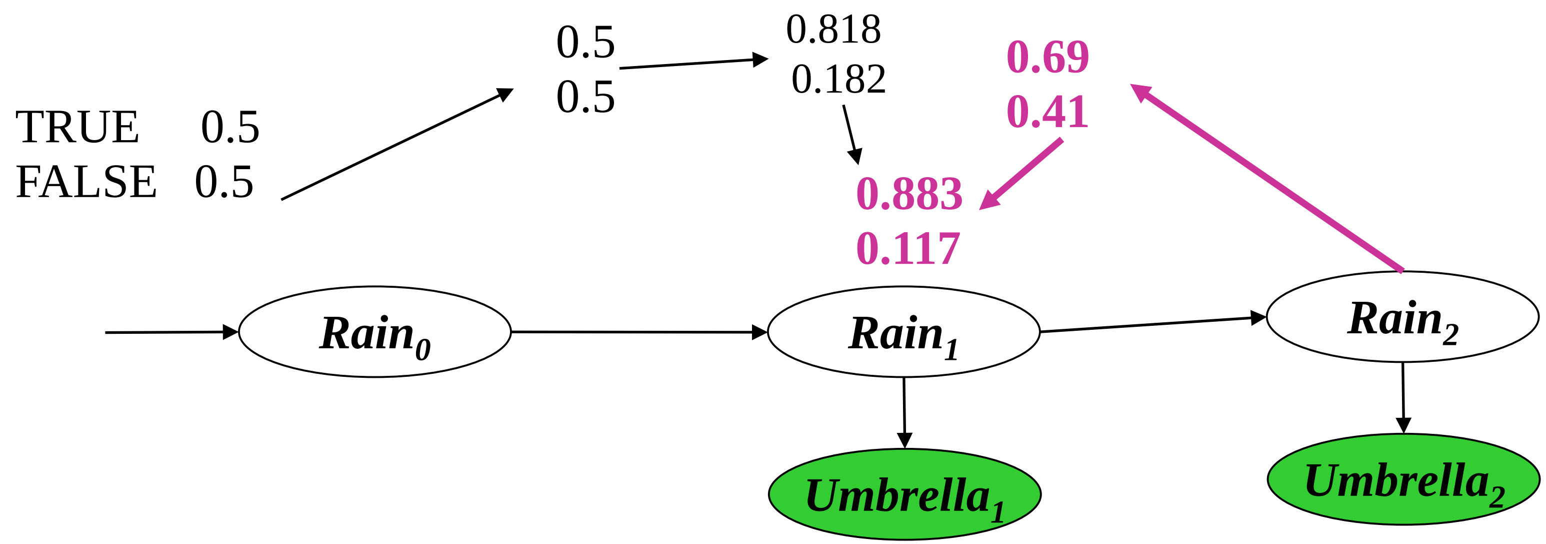
- $P(u_2|r_2)\, P(|r_2)\, \langle P(r_2 \mid r_1), P(r_2 \mid \neg r_1) \rangle +$

  $P(u_2|\neg r_2)\, P(|\neg r_2)\, \langle P(\neg r_2 \mid r_1), P(\neg r_2 \mid \neg r_1) \rangle$

$= (0.9 * 1 * \langle 0.7, 0.3 \rangle) + (0.2 * 1 * \langle 0.3, 0.7 \rangle) = \langle 0.69, 0.41 \rangle$

Thus

- $\alpha\, \mathbf{P}(R_1| u_1)\, \mathbf{P}(u_2 \mid R_1) = \alpha\langle 0.818, 0.182\rangle * \langle 0.69, 0.41\rangle \sim \langle 0.883, 0.117\rangle$



Lecture Overview

Probabilistic temporal Inferences

- Filtering
- Prediction
- Smoothing (forward-backward)
- **Most Likely Sequence of States (Viterbi)**

Most Likely Sequence

- Suppose that in the *rain* example we have the following *umbrella* observation sequence

  [true, true, false, true, true]

- Is the most likely state sequence?

  [rain, rain, no-rain, rain, rain]

- In this case you may have guessed right… but if you have more states and/or more observations, with complex transition and observation models…..

HMMs : most likely sequence (from 322)

Natural Language Processing: e.g., Speech Recognition

- *States:* phoneme ¥ word
- *Observations*: acoustic signal ¥ phoneme

HMM 1 HMM2

My name is

Bioinformatics: Gene Finding

- *States:* coding / non-coding region   x x v v v x x
- *Observations:* DNA Sequences → A T C G G A A

For these problems the critical inference is:

find the most likely sequence of states given a sequence of observations

Viterbi Algo

Part-of-Speech (PoS) Tagging

- Given a text in natural language, label (*tag*) each word with its syntactic category
  - E.g, Noun, verb, pronoun, preposition, adjective, adverb, article, conjunction
- ***Input***
  - Brainpower, not physical plant, is now a firm's chief asset.
- ***Output***
  - Brainpower_NN ,_, not_RB physical_JJ plant_NN ,_, is_VBZ now_RB a_DT firm_NN 's_POS chief_JJ asset_NN ._.

Tag meanings

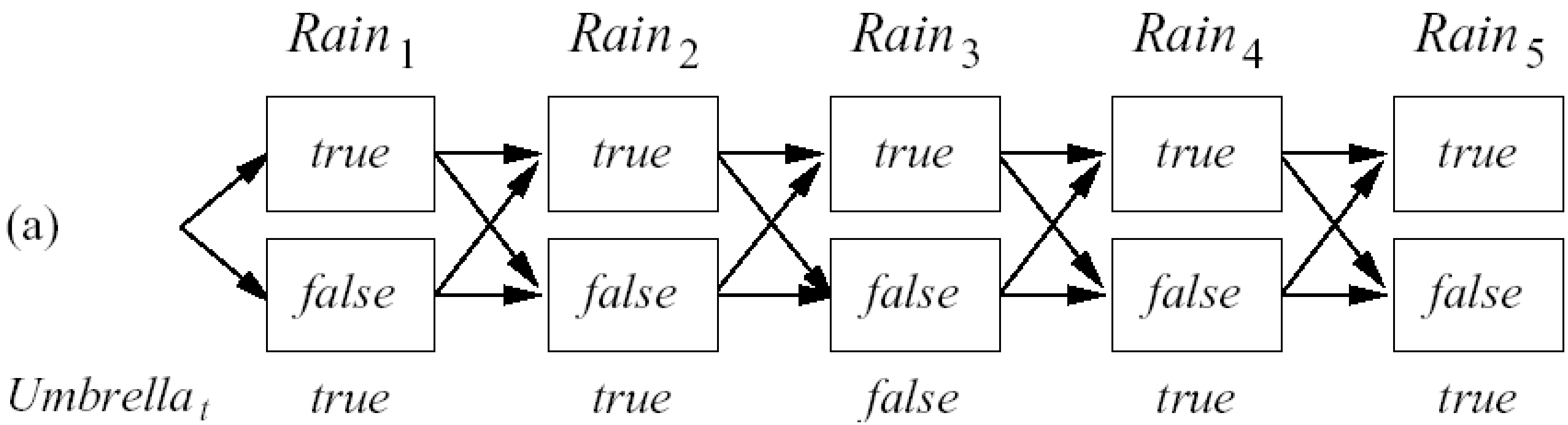
- NNP (Proper Noun singular), RB (Adverb), JJ (Adjective), NN (Noun sing. or mass), VBZ (Verb, 3 person singular present), DT (Determiner), POS (Possessive ending), . (sentence-final punctuation)

POS Tagging is very useful

- As a basis for **parsing** in NL understanding
- **Information Retrieval**
  - ✓ Quickly finding names or other phrases for information extraction
  - ✓ Select important words from documents (e.g., nouns)
- **Word-sense disambiguation**
  - ✓ I made her duck (*how many meanings does this sentence have*)?
- **Speech synthesis**: Knowing PoS produce more natural pronunciations
  - ✓ E.g,. Content (noun) vs. content (adjective); object (noun) vs. object (verb)

Most Likely Sequence (Explanation)

- ***Most Likely Sequence***: $\text{argmax}_{x_{1:T}} P(X_{1:T} \mid e_{1:T})$
- Idea
  - find the most likely path to each state in X_T
  - As for filtering etc. we will develop a recursive solution

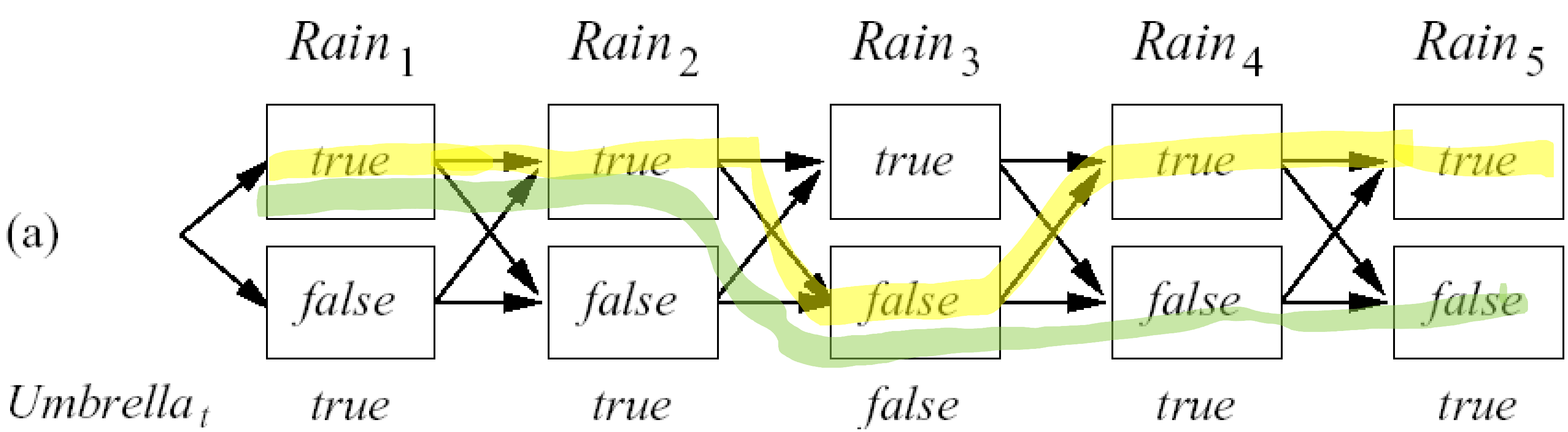


Most Likely Sequence (Explanation)

- ***Most Likely Sequence***: $\text{argmax}_{x_{1:T}} P(X_{1:T} \mid e_{1:T})$
- Idea
  - find the most likely path to each state in X_T
  - As for filtering etc. let's try to develop a recursive solution

Rain$_5$ = true

Rain$_5$ = false



Joint vs. Conditional Prob

- You have two binary random variables X and Y

$$\mathrm{argmax}_x\, P(X \mid Y=t) \;?\; \mathrm{argmax}_x\, P(X, Y=t)$$

A. Always equal

B. Always different

C. It depends

X	Y	$P(X, Y)$
t	t	.4
f	t	.2
t	f	.1
f	f	.3

Most Likely Sequence: Formal Derivation

- ***Most Likely Sequence***: $\text{argmax}_{x_{1:T}} P(X_{1:T} \mid e_{1:T})$
- => $\text{argmax}_{x_{1:T}} P(X_{1:T}, e_{1:T})$

- Let's focus on finding the prob. of the most likely path to state x_{t+1} with evidence $\mathbf{e}_{1:t+1}$.

$$\max_{\mathbf{x}_1,\ldots\mathbf{x}_t} \mathbf{P}(\mathbf{x}_1,\ldots.\ \mathbf{x}_t, \mathbf{x}_{t+1}, \mathbf{e}_{1:t+1}) = \max_{\mathbf{x}_1,\ldots\mathbf{x}_t} \mathbf{P}(\mathbf{x}_1,\ldots.\ \mathbf{x}_t, \mathbf{x}_{t+1}, \mathbf{e}_{1:t}, \mathbf{e}_{t+1}) = \boxed{\text{Cond. Prob}}$$

$$= \max_{\mathbf{x}_1,\ldots\mathbf{x}_t} P(\mathbf{e}_{t+1} | \mathbf{e}_{1:t}, \mathbf{x}_1,\ldots.\ \mathbf{x}_t, \mathbf{x}_{t+1})\ \mathbf{P}(\mathbf{x}_1,\ldots.\ \mathbf{x}_t, \mathbf{x}_{t+1}, \mathbf{e}_{1:t}) = \boxed{\text{Markov Assumption/Indep.}}$$

$$= \max_{\mathbf{x}_1,\ldots\mathbf{x}_t} P(\mathbf{e}_{t+1} | \mathbf{x}_{t+1})\ \mathbf{P}(\mathbf{x}_1,\ldots.\ \mathbf{x}_t, \mathbf{x}_{t+1}, \mathbf{e}_{1:t}) = \boxed{\text{Cond. Prob}}$$

$$= \max_{\mathbf{x}_1,\ldots\mathbf{x}_t} P(\mathbf{e}_{t+1} | \mathbf{x}_{t+1})\ \mathbf{P}(\mathbf{x}_{t+1} | \mathbf{x}_1,\ldots.\ \mathbf{x}_t, \mathbf{e}_{1:t}) \mathbf{P}(\mathbf{x}_1,\ldots.\ \mathbf{x}_t, \mathbf{e}_{1:t}) = \boxed{\text{Markov Assumption/Indep}}$$

$$= \max_{\mathbf{x}_1,\ldots\mathbf{x}_t} \mathbf{P}(\mathbf{e}_{t+1} | \mathbf{x}_{t+1})\ \mathbf{P}(\mathbf{x}_{t+1} | \mathbf{x}_t)\ \mathbf{P}(\mathbf{x}_1,\ldots.\ \mathbf{x}_{t-1}, \mathbf{x}_t, \mathbf{e}_{1:t}) =$$

Move outside the max

$$\mathbf{P}(\mathbf{e}_{t+1} | \mathbf{x}_{t+1}) \max_{\mathbf{x}_t} (\mathbf{P}(\mathbf{x}_{t+1} | \mathbf{x}_t) \max_{\mathbf{x}_1,\ldots\mathbf{x}_{t-1}} \mathbf{P}(\mathbf{x}_1,\ldots.\ \mathbf{x}_{t-1}, \mathbf{x}_t, \mathbf{e}_{1:t}))$$

Learning Goals for today's class

➢You can:

- Describe the **smoothing problem** and derive a solution by manipulating probabilities
- Describe the problem of finding the **most likely sequence of states** (given a sequence of observations)
- Derive recursive solution (if time)

TODO for Mon

- **Keep working on Assignment-2: new due date Fri Oct 21**
- **Midterm : October 26**