Lecture Overview

Probabilistic temporal Inferences

- 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
  - \( P(X_k \mid e_{0:t}) \) for \( 1 \leq k < t \)

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

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

\[ = \alpha P(X_k \mid e_{0:k}) P(e_{k+1:t} \mid X_k, e_{0:k}) \] using...

\[ = \alpha P(X_k \mid e_{0:k}) P(e_{k+1:t} \mid X_k) \] using...

forward message from filtering up to state \( k \), \( f_{0:k} \)

backward message, \( b_{k+1:t} \)
computed by a recursive process that runs backwards from \( t \)

A. Bayes Rule
B. Cond. Independence
C. Product Rule
Smoothing

\[ P(X_k \mid e_{0:t}) = P(X_k \mid e_{0:k}, e_{k+1:t}) \quad \text{dividing up the evidence} \]

\[ = \alpha P(X_k \mid e_{0:k}) P(e_{k+1:t} \mid X_k, e_{0:k}) \quad \text{using Bayes Rule} \]

\[ = \alpha P(X_k \mid e_{0:k}) P(e_{k+1:t} \mid X_k) \quad \text{By Markov assumption on evidence} \]

forward message from filtering up to state \( k \), \( f_{0:k} \)

backward message, \( b_{k+1:t} \)
computed by a recursive process that runs backwards from \( t \)
Backward Message

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

\[ = \sum_{x_{k+1}} P(e_{k+1:t} \mid x_{k+1}) P(x_{k+1} \mid X_k) \text{ by Markov assumption on evidence} \]

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

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

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

\[ \text{because } e_{k+1} \text{ and } e_{k+2:t}, \text{ are conditionally independent given } x_{k+1} \]

Sensor

Model

Recursive call

Product Rule

Transition model

In message notation

\[ b_{k+1:t} = \text{BACKWARD} (b_{k+2:t}, e_{k+1}) \]
Proof of equivalent statements

1. \( P(X|Y,Z) = P(X|Z) \) =>

\[ \frac{P(X,Y,Z)}{P(Y,Z)} = \frac{P(X,Z)}{P(Z)} \] => \( P(Y|X,Z) = P(Y|Z) \)

2. \( P(X,Y,Z) = P(Y,Z) \)

\[ \frac{P(X,Y,Z)}{P(X,Z)} = \frac{P(Y,Z)}{P(Z)} \] => \( P(Y|X,Z) = P(Y|Z) \)

3. \( P(X,Y|Z) = \frac{P(X,Y,Z)}{P(Z)} \)

\[ \frac{P(X,Y,Z)}{P(Y,Z)} \cdot \frac{P(X,Z)}{P(Z)} = P(Y|Z) \cdot P(X|Z) \]
More Intuitive Interpretation (Example with three states)

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

\[ X = \{ S_1, S_2, S_3 \} \]
Forward–Backward Procedure

➢ To summarize, we showed

➢ \( P(X_k \mid e_{0:t}) = \alpha P(X_k \mid e_{0:k}) P(e_{k+1:t} \mid X_k) \)

➢ 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 \( k+1 \).
How is it Backward initialized?

The backwards phase is initialized with making an unspecified observation $e_{t+1}$ at $t+1$. …

$$b_{t+1:t} = P(e_{t+1} | X_t) = P(\text{unspecified} | X_t) = ?$$

A. 0  B. 0.5  C. 1
The backwards phase is initialized with making an unspecified observation $e_{t+1}$ at $t+1$:

$$b_{t+1:t} = P(e_{t+1}/X_t) = P(\text{unspecified}/X_t) = 1$$

You will observe something for sure! It is only when you put some constraints on the observations that the probability becomes less than 1.
Let’s compute the probability of rain at $t = 1$, given umbrella observations at $t=1$ and $t=2$.

From $P(X_k | e_{1:t}) = \alpha P(X_k | e_{1:k}) P(e_{k+1:t} | X_k)$, we have

$$P(R_1 | e_{1:2}) = P(R_1 | u_1, u_2) = \alpha P(R_1 | u_1) P(u_2 | R_1)$$

**forward message from filtering up to state 1**

**backward message for propagating evidence backward from time 2**

$P(R_1 | u_1) = <0.818, 0.182>$ as it is the filtering to $t = 1$ that we did in lecture 14.
Rain Example

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

- $P(u_2 | R_1) = \sum_{r \in r_2, \neg r_2} P(u_2 | r) P(r) P(r | R_1) =$

- $P(u_2 | r_2) P(r_2) <P(r_2 | r_1), P(r_2 | \neg r_1)> +$

- $P(u_2 | \neg r_2) P(\neg r_2) <P(\neg r_2 | r_1), P(\neg r_2 | \neg r_1)>$

$= (0.9 * 1 * <0.7, 0.3>) + (0.2 * 1 * <0.3, 0.7>) = <0.69, 0.41>$

Thus

- $\alpha P(R_1 | u_1) P(u_2 | R_1) = \alpha <0.818, 0.182> * <0.69, 0.41> \sim <0.883, 0.117>$

Term corresponding to the Fictitious unspecified observation sequence $e_{3:2}$
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
  
  \[ \text{true, true, false, true, true} \]

- Is the most likely state sequence?
  
  \[ \text{rain, rain, no-rain, rain, rain} \]

- 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 \( \xrightarrow{HMM_1} \) word
- Observations: acoustic signal \( \xrightarrow{HMM_2} \) phoneme

Bioinformatics: Gene Finding
- States: coding / non-coding region
- Observations: DNA Sequences

For these problems the critical inference is:
find the most likely sequence of states given a sequence of observations

\[ \text{Viterbi Algorithm} \]
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**: \(	ext{argmax}_{x_{1:T}} P(X_{1:T} | e_{1:T})\)

- **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. we will develop a recursive solution

---

Diagram:

```
(a) Rain_1  Rain_2  Rain_3  Rain_4  Rain_5
       |       |       |       |
       true  true  true  true  true
     /     /     /     /     /
true false false false false
     \
true false false false false
       \
true false false false false
```

- \( \text{Rain}_5 = \text{true} \)
- \( \text{Rain}_5 = \text{false} \)
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: due Fri Oct 20
- Midterm: October 25