Topics

- Hidden markov models
  - dynamic programming, examples
- Representation and graphical models
  - variables and states
  - graphical models
Dynamic programming: review

\[
\begin{align*}
  s_0 & \rightarrow s_1 \rightarrow s_2 \rightarrow \ldots \rightarrow s_{n-1} \rightarrow s_n \\
  \downarrow & \quad \downarrow \quad \downarrow \quad \downarrow \quad \downarrow \quad \downarrow \\
  x_0 & \quad x_1 \quad x_2 \quad x_{n-1} \quad x_n 
\end{align*}
\]

- Let \( \{s_0^{(t,i)}, \ldots, s_t^{(t,i)} = i\} \) be the most likely state sequence given \( x_0, \ldots, x_t \) that is forced to end up in \( s_t = i \) at time \( t \). Then

\[
\delta_t(i) = P(x_0, \ldots, x_t, s_0^{(t,i)}, \ldots, s_t^{(t,i)}).
\]

- We can evaluate these probabilities recursively by replacing each "sum" with a "max" in the forward propagation:

\[
\begin{align*}
  \delta_0(i) &= P_0(i)P_o(x_t|s_0 = i), \\
  \delta_t(i) &= \max_j \left\{ \delta_{t-1}(j)P_1(s_t = i|s_{t-1} = j) \right\} \times P_o(x_t|s_t = i)
\end{align*}
\]
Dynamic programming: review

- We can recover the most likely hidden state sequence from \( \{ \delta_t(\cdot) \} \) by retrospectively examining the “max” choices made in evaluating these probabilities.

We find the end state \( s_n^* \) of the most likely state sequence by maximizing over the probabilities associated with the most likely state sequences forced to land on different states at \( t = n \):

\[
s_n^* = \arg \max_j \delta_n(j)
\]

The recovery of the remaining states along the most likely path can be done recursively (backwards):

\[
s_t^* = \arg \max_j \left\{ \delta_t(j) P_1(s_{t+1} = s_{t+1}^* | s_t = j) \right\}
\]
The most likely path has the property that any partial path is also optimal:

If $s^*_t = i$ then $\{s^*_0, \ldots, s^*_t\}$ is also the most likely state sequence forced to end up in $s_t = i$ at time $t$ given only $x_0, \ldots, x_t$. 
Dynamic programming: example

- Same example as in the EM case (3 states, Gaussian outputs)

- The most likely hidden state sequence \( \{ s_0^*, \ldots, s_n^* \} \) need not agree with the most likely states derived from the posterior marginals \( \gamma_t(i) \)
Example cont’d

final

ML model, no observations
Examples: speech

- We can annotate or parse speech signals by evaluating the most likely hidden state sequence

A speech spectrogram example (refs)

Never touch a snake with your bare hands
Examples: alignment

- A “linear” HMM can be used to align sequences of observations.

![Diagram of HMM alignment]
Topics

- Representation and graphical models
  - variables and states
  - graphical models
What is a good representation?

- Properties of good representations
  1. Explicit
  2. Compact
  3. Modular
  4. Permits efficient computation
  5. etc.
Representing the model structure

- Two possible representations of Markov models:
  1. in terms of state diagrams (nodes in the graph correspond to the possible values of the states)
     - Example of a state diagram with states $s_0$, $s_1$, $s_2$
     - The transitions between states are indicated by arrows.

  2. in terms of variables (nodes in the graph are variables):
     - Example of variables $s_0$, $s_1$, $s_2$
     - The transitions between variables are indicated by arrows.

- The representations differ in terms of what aspects of the model are made explicit
Model structure cont’d

- Case 1: *sparse transition* structure

  1. State transition diagram is *explicit*

```
  s0  s1  s2
   1
   2
```

  2. Representation in terms of variables leaves this *implicit*

```
s0  s1  s2
  →  →  →
```

Tommi Jaakkola, MIT AI Lab
Model structure cont’d

- Case 2: successive states are *independent of each other*

  1. State transition diagram is fully connected

    ![State transition diagram]

    1. Representation in terms of variables is *explicit*

    \[
    s_0 \quad s_1 \quad s_2
    \]
Case 3: time series signals such as music may involve multiple relatively independent underlying processes operating at different *time scales*

1. State transition diagram (argh #$& ...)

2. In terms of variables (graph model)
Graphical models

- Graph representations of probability models in terms of variables are known as graphical models.

A mixture model as a graphical model:

- Different types of graph models differ in terms of how we represent dependencies and independencies among the variables:
  1. Bayesian networks (natural for “causal” relations)
  2. Markov random fields (natural for physical or symmetric relations)
  3. etc.
Bayesian networks: examples

A Markov chain:

A hidden Markov model:
Qualitative inference

- The graph provides a qualitative description of the domain:

  ![Diagram]

  \[ x_1 = \text{1st coin toss} \]
  \[ x_2 = \text{2nd coin toss} \]
  \[ x_3 = \text{same?} \]
Qualitative inference

- The graph provides a qualitative description of the domain

![Graph of coin tosses]

\[ x_1 = 1\text{st coin toss} \]
\[ x_2 = 2\text{nd coin toss} \]
\[ x_3 = \text{same?} \]

Marginal independence

Induced dependence

Note that the induced dependence pertains to our beliefs about the outcomes of the coin tosses.
Qualitative inference cont’d

• Just by looking at the graph, we can determine what we can and cannot ignore (why important?)
Marginal independence of “Earthquake” and “Burglary”

Earthquake  Burglary

Radio report  Alarm
Qualitative inference cont’d

- Induced dependence:

  Earthquake  Burglary
  
  Radio report  Alarm

- Explaining away:

  Earthquake  Burglary
  
  Radio report  Alarm
Two levels of description

- Graphical models need two levels of specification
  1. Qualitative properties captured by a graph
     \[x_1 = \text{first coin toss}\]
     \[x_2 = \text{second coin toss}\]
     \[x_3 = \text{same?}\]

   ![Diagram of coin tosses]

  2. Quantitative properties specified by the associated probability distribution
     \[P(x_1, x_2, x_3) = P(x_1)P(x_2)P(x_3|x_1, x_2)\]

     where, e.g.,
     \[P(x_1 = \text{heads}) = 0.5\]
     \[P(x_3 = \text{same}|x_1 = \text{heads}, x_2 = \text{tails}) = 0\]
More examples

- $i$ and $j$ correspond to the discrete choices in the mixture model.
- $\mathbf{x}$ is the (vector) variable whose density we wish to model.
- We cannot tell what the component distributions $P(\mathbf{x}|i)$ are based on the graph alone.

Mixture model  hierarchical mixture model