Announcements, Assignments, and Reminders

- Visit the URL links that were covered in previous lectures.
- Homework 1 is out, due Thursday, May 2nd, at 11:45pm via our dropbox (https://catalyst.uw.edu/collectit/dropbox/bilmes/26924).
First Guest Lecture, next Thursday, May 16th

Dr. Alex Acero, from microsoft research, will be giving a guest lecture next Thursday. The abstract follows:

**Speech Coding and Enhancement**  
_Dr. Alex Acero_  
_Microsoft Research_  
_Thursday, May 16th, 2013, 1:30-3:20pm, Thomson Hall, 235_  
_In this lecture we’ll cover the fundamentals of speech coding and speech enhancement. Speech coding is used to compress the digital signal for efficient transmission and is used in cellphones and all internet transmissions. Speech enhancement deals with acquiring and restoring a signal in the presence of noise using techniques such as microphone arrays, echo cancellation and single microphone noise suppression._

Cumulative Outstanding Reading

- Read chapters 1 and 2 in our book (Huang, Acero, Hon, “Spoken Language Processing”).
- Read chapters 3 and 4 in our book (Huang, Acero, Hon, “Spoken Language Processing”).
- Read Chapter 6 in our book (Huang, Acero, Hon, “Spoken Language Processing”).
General definition of HMMs: 
\[ p(x_{1:T}, q_{1:T}) = \prod_t p(x_t | q_t) p(q_t | q_{t-1}) \]

- The HMM Trellis
- The HMM forward/backward (\( \alpha/\beta \)) recursion
- The posterior \( p(q_t | x_{1:T}) \) and \( p(q_t, q_{t+1} | x_{1:T}) \).
- The Viterbi algorithm.

Outline of today

- Hidden Markov Models, more description and properties.
Good books (for today)

- our book (Huang, Acero, Hon, “Spoken Language Processing”)
- Deller et. al. “Discrete-time Processing of speech signals”
- J. Bilmes, “What HMMs can do”, 2010

What HMMs can do

- HMMs are more powerful than you might think.
- It is worthwhile to understand how flexible and powerful HMMs are to see how applicable they are to speech recognition.
- We next visit a set of properties about HMMs that should be remembered.
Observations are not i.i.d.

- Joint probability under an HMM:
  \[ p(X_{t:t+h} = x_{t:t+h}) = \sum_{q_{t:t+h}} \prod_{j=t}^{t+h} p(X_j = x_j \mid Q_j = q_j) a_{q_j-1, q_j} \cdot \]

- Unless only one state is possible, observations do not factorize.
- In an HMM, there are no statements of the form \( A \perp \perp B \).
- HMMs have conditional independence properties (like all DGMs have).

\[ p(X_{t:t+h} = x_{t:t+h} \mid Q_{t:t+h} = q_{t:t+h}) = \prod_{\tau=t}^{t+h} p(X_\tau = x_\tau \mid Q_\tau = q_\tau) \cdot \]

where \( c \) is a positive normalizing constant over \( x_{1:T} \).

This is just a different semi-ring. The resulting distribution over observations does not in general factorize, so no i.i.d. here either.
The Viterbi path (most-probable explanation) of an HMM is defined as follows:

$$q^{*}_{1:T} \in \arg\max q_{1:T} p(X_{1:T} = x_{1:T}, q_{1:T})$$  \hspace{1cm} (11.2)

This is a standard method for finding a mapping from observations $x_{1:T}$ to candidate answers $q^{*}_{1:T}$, also called a “decoding.”

Other times, we might have $k$ HMMs, corresponding to $k$ classes

$$\{p_k(x_{1:T}, q_{1:T}) : k = 1 \ldots K\}$$  \hspace{1cm} (11.3)

and we wish to perform classification amongst the $K$ discrete set of classes as follows

$$(k^{*}, q^{*}_{1:T}) = \arg\max_{q_{1:T},k} p_k(x_{1:T}, q_{1:T})$$  \hspace{1cm} (11.4)

and $k^{*}$ becomes part of the hypothesized answer.

We can view this as $K$ distributions over just the observations, i.e.,

$$p_{\text{vit}}^k(x_{1:T}) \propto p_k(x_{1:T}, q^{*}_{1:T})$$  \hspace{1cm} (11.5)

$$\propto \max_{q_{1:T}} p_k(x_{1:T}, q_{1:T})$$  \hspace{1cm} (11.6)

$$\propto \max_{q_{1:T}} \prod_{t=1}^{T} p_k(x_t|q_t)p_k(q_t|q_{t-1})$$  \hspace{1cm} (11.7)

Note that this max-marginal can be completely different than the standard summation marginal

$$\sum_{q_{1:T}} \prod_{t=1}^{T} p_k(x_t|q_t)p_k(q_t|q_{t-1})$$  \hspace{1cm} (11.8)

This is just a different semi-ring. The resulting distribution over observations does not in general factorize, so no i.i.d. here either. 

The distribution $p_{\text{vit}}^k(x_{1:T})$ also does not in general have any (marginal) independence assumptions.
HMMs and stationarity

- HMM is a stochastic process over \( \{X_t\} \)
- An HMM is stationary whenever

\[
p(X_{t_1+h:n+h} = x_{1:n}) = p(X_{t_1:n} = x_{1:n})
\]  \hspace{1cm} (11.9)

When might this be the case?

- We have

\[
p(X_{t_1:n+h} = x_{1:n})
\]
\[
= \sum_{q_1:n} p(X_{t_1:n+h} = x_{1:n}, Q_{t_1:n+h} = q_{1:n}) \\
= \sum_{q_1:n} p(Q_{t_1+h} = q_1)p(X_{t_1+h} = x_1|Q_{t_1+h} = q_1) \\
\prod_{i=2}^{n} p(X_{t_i+h} = x_i|Q_{t_i+h} = q_i)p(Q_{t_i+h} = q_i|Q_{t_{i-1}+h} = q_{i-1})
\]

where \( f(x_{2:n}, q_1) \) is a function that is independent of the variable \( h \).

- For HMM stationarity to hold, it is required that

\[
p(Q_{t_1+h} = q_1) = p(Q_{t_1} = q_1) \text{ for all } h.
\]
HMMs’ stationarity depends on MC

- Therefore, the HMM’s stationarity condition is entirely determined by the stationarity condition of the underlying hidden Markov chain.
- Consider the way in which HMMs are often used:
  - Long chains
  - Chains with cycle transition matrices
  - Chains with upper-triangular matrices
  - Chains with strictly left-to-right transitions. Ex: speech recognition

<table>
<thead>
<tr>
<th>Word</th>
<th>Pronunciation</th>
</tr>
</thead>
<tbody>
<tr>
<td>cat</td>
<td>[kæt]</td>
</tr>
<tr>
<td>cats</td>
<td>[kæts]</td>
</tr>
<tr>
<td>pig</td>
<td>[pɪɡ]</td>
</tr>
<tr>
<td>pigs</td>
<td>[pɪɡz]</td>
</tr>
<tr>
<td>fox</td>
<td>[fɒks]</td>
</tr>
<tr>
<td>foxes</td>
<td>[fɒksəz]</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Word</th>
<th>Pronunciation</th>
</tr>
</thead>
<tbody>
<tr>
<td>goose</td>
<td>[ɡuːz]</td>
</tr>
<tr>
<td>geese</td>
<td>[ɡiːz]</td>
</tr>
<tr>
<td>hedgehog</td>
<td>[ˈhɛdʒ.hɔɡ]</td>
</tr>
<tr>
<td>hedgehogs</td>
<td>[ˈhɛdʒ.hɔɡz]</td>
</tr>
</tbody>
</table>

- Hence, in only rare cases, when HMMs are used, are they stationary stochastic processes.

Gaussian Mixture HMM

- One of the most widely used HMMs in practice is one where the observation distributions are Gaussian mixtures, where

\[
p(x|q) = \sum_c p(x|q,c)p(c|q)
\]

\[
= \sum_m \mathcal{N}(x|\mu_{qm}, \Sigma_{qm})c_{mq}
\]

and where

\[
\mathcal{N}(x|\mu, \Sigma) = \frac{1}{\sqrt{2\pi|\Sigma|^{d/2}}} \exp\left(-\frac{1}{2}(x-\mu)^{\top}\Sigma^{-1}(x-\mu)\right)
\]

- The HMM BN becomes

\[
\begin{array}{c}
Q_{t-1} \downarrow Q_t \downarrow Q_{t+1} \downarrow Q_{t+2} \\
\mathcal{C}_{t-1} \downarrow \mathcal{C}_t \downarrow \mathcal{C}_{t+1} \downarrow \mathcal{C}_{t+2} \\
X_{t-1} \downarrow X_t \downarrow X_{t+1} \downarrow X_{t+2}
\end{array}
\]
Correlated & Covariance

- Correlation between two real random vectors $X$ and $Y$
  \[ \text{cor}(X, Y) = E[XY^\top] \quad (11.16) \]

- Covariance between two real random vectors $X$ and $Y$
  \[ \text{cov}(X, Y) = E[(X - EX)(Y - EY)^\top] = E[XY^\top] - E[X]E[Y^\top] \quad (11.17) \]

- $X$ and $Y$ are **uncorrelated** if $\text{cov}(X, Y) = 0$.
- $\text{cov}(X, Y) = \text{cor}(X, Y)$ if either the means are zero.
- If $X \perp Y$ then $\text{cov}(X, Y) = 0$ but vice verse only if $X, Y$ are jointly Gaussian.
- $\text{cov}(X, Y) = 0$ is indication of lack of linear dependence, and hint there might not be strong dependence at all.

Correlation over time of simple HMM

- Consider single-component Gaussian HMM (i.e., for each state, the observation distribution is a single multivariate Gaussian).
- Assume that the Markov chain is currently stationary, with stationary distribution $\pi$.
- What about $\text{cov}(X_t, X_{t+h})$? Is it zero? How does it decay?
- Computing $E[X_t]$
  \[ EX_t = \int xp(X_t = x)dx = \int x \sum_i p(X_t = x|Q_t = i)\pi_i dx \]
  \[ = \sum_i E[X_t|Q_t = i]\pi_i = \sum_i \mu_i \pi_i \quad (11.19) \]

  so this is just the weighted sums of means, where the weights are the stationary distribution probabilities.
Correlation over time of simple HMM

- Similarly, let’s compute the correlation:

\[
E[X_t X_{t+h}^T] = \int x y^T p(X_t = x, X_{t+h} = y) dxdy
\]  
(11.20)

\[
= \int x y^T \sum_{ij} p(X_t = x, X_{t+h} = y|Q_t = i, Q_{t+h} = j) p(Q_{t+h} = j|Q_t = i) \pi_i dxdy
\]  
(11.21)

\[
= \sum_{ij} E[X_t X_{t+h}^T|Q_t = i, Q_{t+h} = j] (A^h)_{ij} \pi_i
\]  
(11.22)

\[
E[X_t | Q_t = i] E[X_{t+h}^T | Q_{t+h} = j] = \mu_i \mu_j^T
\]  
(11.23)

\[
\text{cov}(X_t, X_{t+h}) = \sum_{ij} \mu_i \mu_j^T (A^h)_{ij} \pi_i - \left( \sum_i \mu_i \pi_i \right) \left( \sum_i \mu_i \pi_i \right)^T
\]  
(11.24)

The above equations follow from 
\[p(Q_{t+h} = j|Q_t = i) = (A^h)_{ij}\] (i.e., 
the Chapman-Kolmogorov equations) where \((A^h)_{ij}\) is the \(i,j\)th 
element of the matrix \(A\) raised to the \(h\) power.
Correlation over time of simple HMM

- Thus, cov\(_{(X_t, X_{t+h})}\) is in general not equal to zero.
- Fact: \(A^h \rightarrow 1\pi^\top\), and this is a matrix with all rows equal to the stationary distribution.
- Therefore, we have

\[
\text{cov}(X_t, X_{t+h}) = \sum_{ij} \mu_i \mu_j^\top (A^h)_{ij} \pi_i - \left( \sum_i \mu_i \pi_i \right) \left( \sum_i \mu_i \pi_i \right)^\top \tag{11.28}
\]

\[
\rightarrow \sum_{ij} \mu_i \mu_j^\top \pi_j \pi_i - \left( \sum_i \mu_i \pi_i \right) \left( \sum_i \mu_i \pi_i \right)^\top = 0 \tag{11.30}
\]

- Thus, while the covariance between observations is not necessarily zero in an HMM, once we are at a stationary distribution, this covariance goes to zero exponentially fast.

Example of the decay in the mutual-information correlation from a real-world HMM. I.e., we see \(f(\tau) = I(X_t; X_{t+\tau})\), where \(I()\) is the mutual information function.

- Mutual information is stronger than correlation.
- This is compared against i.i.d. samples (high peak at \(\tau = 0\) is expected).
### State Duration Modeling

- Markov chain, state duration distribution is geometric
- Let $D$ be such a random variable, then
  \[ P(D = d) = p^{d-1}(1 - p) \]  \hspace{1cm} (11.31)
  where $d \geq 1$ is an integer and $p = a_{ii}$, if $D$ is the duration random variable for state $i$ of the chain, giving:

- Many sequential tasks have sub-segments that do not follow this distribution

### State Duration Modeling

- Many “tricks” to using an HMM, can alleviate such problems.
- “state-tying”, where multiple states have the same observation distribution (parameters are shared).
- That is, state $q$ and $q'$ are tied if it is the case that
  \[ p(x|Q_t = q) = p(x|Q_t = q') \quad \forall x \in D_X \]  \hspace{1cm} (11.32)
- If $n$ states in a series are strung together, all of which share the same observation distribution, that observation distribution will be active for as long as we are in that state. For example:
State Duration Modeling - Tied States

- This is a form of parameter sharing: different states share the same observation distribution.
- For distinct states $q_1$ and $q_2$, we have $p(x_t|q_1) = p(x_t|q_2)$ for all $x_t$.
- We can visualize this informally as:

![Diagram of HMM states and transitions]

- We use this to model the time (and duration) during which a given $p(x|q)$ is “active” in a sample. Here, it can not be less than 4.!
- This moreover gives us a new distribution as we show next.

This corresponds to the sum of random variables. Let $\{D_i\}$ be a collection of independent geometrically distributed random variables with parameter $p$, and let $W_r = \sum_{i=1}^r D_r$, then

$$p(W_r = k) = \binom{k-1}{r-1} p^r (1-p)^{k-r}, \quad k = r, r+1, \ldots \quad (11.33)$$

- This is a “negative binomial distribution”:

![Graph of negative binomial distribution]
State Duration Modeling

- If we have multiple parallel states in series, all of which share the same observation distribution, we can construct much more interesting (multimodal) distributions.
- For example, the following left chain will have distribution as shown on the right (a mixture of negative binomial distributions).

Other examples: very long chains, ladders, fixed-length distributions (histograms), and so on.

HMMs can have flexible distributions, cost of extra states.

State Duration and Viterbi

Consider the following two super-state duration distributions:

- The Viterbi path (most-probable explanation) $q_{1:T}^*$ of an HMM is defined:
  \[
  q_{1:T}^* \in \arg\max_{q_{1:T}} p(x_{1:T}, q_{1:T}) = \arg\max_{q_{1:T}} \prod_t p(x_t | q_t) p(q_t | q_{t-1})
  \]
  \[\text{(11.34)}\]

- Under which duration model would any sequence have a higher probability?
- Viterbi decoding destroys the meaning of “duration” in the duration modeling.
State Duration and Viterbi

- The inherent problem is Viterbi — in taking the max, we are not considering all possible paths, rather only the most likely one.
- Similar to the “typical set” idea in information theory — the most probable sequence is not a typical one (one with typical properties).
- E.g., biased coin flips: $P(H) = 1 - P(T) = 0.9$. Let $x_{1:T}$ be a sequence of flips, most probable sequence is all heads $x_t = H$.
- The most probable probability goes to zero.

$$
\Pr(HHHHH \ldots H) \xrightarrow{n \to \infty} 0 
$$

(11.35)

- However, the typical set probability goes to one:

$$
\Pr(\{x_{1:T} : \text{number of heads in } x_{1:T} \text{ is } \approx 0.9T\}) \xrightarrow{n \to \infty} 1 
$$

(11.36)

For a Markov chain, there is one most probable sequence (see above right).

However, there are many sequences (each of less probability) but they, in their collective contribution, increase the probability of a sequence of a longer length.
State Duration and Viterbi

- We want a combination of max-sum. I.e., given a model of the form:

\[ p(x_{1:T}, y_{1:T}, q_{1:T}) = \prod_t p(x_t|q_t, y_t)p(y_t|y_{t-1}, q_t, q_{t-1})p(q_t|q_{t-1}) \]  

(11.37)

do the computation:

\[ q_{1:T}^* \in \arg\max_{q_{1:T}} \sum_{y_{1:T}} p(x_{1:T}, y_{1:T}, q_{1:T}) \]  

(11.38)

\[ = \arg\max_{q_{1:T}} \sum_{y_{1:T}} \prod_t p(x_t|q_t, y_t)p(y_t|y_{t-1}, q_t, q_{t-1})p(q_t|q_{t-1}) \]  

(11.39)

- Unfortunately, this computation is intractable to do exactly.

State Duration, Viterbi, and Discriminability

- One approach: make all objects represented by a model very probable under that model — generative accuracy.
- Discriminability: rather than make objects represented by a model more probable, make objects not represented by model less probable.
- I.e., let \( x_A \) be an object of type \( A \) and \( x_B \) be an object of type \( B \). Two models \( p(x|A) \) and \( p(x|B) \).
- Rather than making \( p(x_A|A) \) and \( p(x_B|B) \) bigger, we make \( p(x_A|B) \) and \( p(x_B|A) \) smaller — this will help discriminability.
State Duration, Viterbi, and Discriminability

- Applied to duration modeling under Viterbi computation, again consider:

- Still there are some benefits to do this. Why?
- Key is minimum duration modeling, achieved using multiple states sharing same output distribution. Each “super-state” has a minimum duration that it must be in.
- This can significantly improve discriminability since objects of lengths different than current model will get a reduced (sometimes significantly) score.

What HMMs can do - summary so far

- Observations are not i.i.d., but conditioned on state variables, they are independent.
- Observations are not “Viterbi i.i.d.”
- HMMs are a stationary process over $p(x_{1:n})$ whenever the underlying hidden Markov chain is a stationary process.
- Single Gaussian per state HMM: Covariance decays as:

$$
\text{cov}(X_t, X_{t+h}) = \sum_{i,j} \mu_i \mu_j^\top (A^h)_{ij} \pi_i - \left( \sum_i \mu_i \pi_i \right) \left( \sum_i \mu_i \pi_i \right)^\top \\
\overset{h \to \infty}{\rightarrow} \sum_{i,j} \mu_i \mu_j^\top \pi_j \pi_i - \left( \sum_i \mu_i \pi_i \right) \left( \sum_i \mu_i \pi_i \right)^\top = 0
$$
What HMMs can do - summary so far

- but mutual information (in practice) can apparently extend in time reasonably far (but also decays).
- Parameter sharing means enormous flexibility in state duration models (e.g., negative binomial, mixtures thereof, fixed histograms).

HMMs Generative Accuracy

- We can view an HMM as an approximate generative distribution of the observation variables, as in $p_h(x_{1:T}) \approx p(x_{1:T})$
- Given that $p_h$ is an approximation, one that is a mixture

$$p_h(x_{1:T}) = \sum_{q_{1:T}} p_h(x_{1:T}, q_{1:T}) \quad (11.40)$$

what can we say about $p_h$ and its accuracy?
- Accuracy can be measured by KL-divergence

$$D(p(x_{1:T}) \parallel p_h(x_{1:T})) = \sum_{x_{1:T}} p(x_{1:T}) \log \frac{p(x_{1:T})}{p_h(x_{1:T})} \quad (11.41)$$

and if $D(p(x_{1:T}) \parallel p_h(x_{1:T})) = 0$, then the HMM is perfectly generatively accurate.
For an HMM to be generatively accurate, we can derive necessary conditions on the HMM, e.g., number of required states.

Recall, $n^{th}$-order Markov chain convertable to 1st-order one.

If $D(p(x_{1:T})||p_h(x_{1:T})) = 0$, then the following mutual information quantities must be equal

$$I(X_{S_1}; X_{S_2}) = I_h(X_{S_1}; X_{S_2})$$

and where $S_1, S_2 \subseteq 1:T$, and where $I(\cdot; \cdot)$ is true mutual information, and $I_h(\cdot; \cdot)$ is the mutual information under the HMM

$$I_h(X_{S_1}; X_{S_2}) = \sum_{x_{S_1 \cup S_2}} p_h(x_{S_1}, x_{S_2}) \log \frac{p_h(x_{S_1}, x_{S_2})}{p_h(x_{S_1})p_h(x_{S_2})}$$

Define $X_{-t} \Delta = \{X_1, X_2, \ldots, X_{t-1}, X_{t+1}, \ldots, X_T\}$ (i.e., $-t$ is set of all indices sans $t$).

Theorem 11.4.1 (Necessary conditions for generative HMM accuracy.)

An HMM with joint observation distribution $p_h(X_{1:T})$ will accurately model the true distribution $p(X_{1:T})$ only if the following three conditions hold for all $t$:

- $I_h(X_{-t}; Q_t) \geq I(X_t; X_{-t})$
- $I_h(Q_t; X_t) \geq I(X_t; X_{-t})$
- $|D_Q| \geq 2I(X_t; X_{-t})$

where $I_h(X_{-t}; Q_t)$ (resp. $I_h(Q_t; X_t)$) is the information transmission rate between $X_{-t}$ and $Q_t$ (resp. $Q_t$ and $X_t$) under an HMM, and $I(X_t; X_{-t})$ is the true information transmission rate between $I(X_t; X_{-t})$.
HMMs Generative Accuracy

Proof. Accurate HMM (i.e., zero KL-divergence from true distribution) implies

\[ I(X_{-t}; X_t) = I_h(X_{-t}; X_t). \]

We expand \( I_h(X_{-t}; Q_t, X_t) \) in two ways using the chain rule of mutual information:

\[
I_h(X_{-t}; Q_t, X_t) = I_h(X_{-t}; Q_t) + I_h(X_{-t}; X_t|Q_t) \tag{11.44}
\]

\[
I_h(X_{-t}; Q_t, X_t) = I(X_{-t}; X_t) + I_h(X_{-t}; Q_t|X_t) \tag{11.45}
\]

The HMM conditional independence properties say that

\[ I_h(X_{-t}; X_t|Q_t) = 0, \]

implying

\[ I_h(X_{-t}; Q_t) = I(X_{-t}; X_t) + I_h(X_{-t}; Q_t|X_t) \tag{11.48} \]

Reasoning as above, this leads to

\[ I_h(X_{-t}; Q_t) \geq I(X_{-t}; X_t), \tag{11.49} \]

since \( I_h(X_{-t}; Q_t|X_t) \geq 0 \). This is the first condition. Similarly, the quantity \( I_h(X_t; Q_t, X_{-t}) \) may be expanded as follows:

\[
I_h(X_t; Q_t, X_{-t}) = I_h(X_t; Q_t) + I_h(X_t; X_{-t}|Q_t) \tag{11.50}
\]

\[
I_h(X_t; Q_t, X_{-t}) = I(X_t; X_{-t}) + I_h(X_t; Q_t|X_{-t}) \tag{11.51}
\]

Reasoning as above, this leads to

\[ I_h(X_t; Q_t) \geq I(X_t; X_{-t}), \tag{11.53} \]

the second condition.
Proof cont.: HMMs Generative Accuracy

... cont.

A sequence of inequalities establishes the third condition:

\[
\log |D_Q| \geq H(Q_t) \geq H(Q_t) - H(Q_t | X_t) \\
= I_h(Q_t; X_t) \geq I(X_t; X_{-t})
\]

so \( |D_Q| \geq 2^{I(X_t; X_{-t})} \).

- This is a lower bound - the number of states must have enough capacity so that it is not a bottleneck, at the very least!
- This could be quite large, and grow with \( T \).
- r.h.s. \( I(X_t; X_{-t}) \) is upper bounded by \( H(X_{-t}) \) which could be as bad as \( \log |D_{X_{-t}}| \)

Nec. conditions for HMMs Generative Accuracy

- Insufficient states can lead to model inaccuracies (e.g., state duration distribution using a geometric rather than something more realistic, add states to improve duration distribution while sharing observation parameters)
- Observation density family must be rich enough (2nd inequality)
- Two bottlenecks: observation density (e.g., number of Components of a Gaussian mixture), and time-dependency (number of states).
Suff conditions for HMMs Generative Accuracy

**Theorem 11.4.2**

**Sufficient conditions for HMM accuracy.** An HMM $p_h(X_{1:T})$ will accurately represent a true discrete distribution $p(X_{1:T})$ if the following conditions hold for all $t$:

- $H(Q_t|X_{<t}) = 0$
- $p_h(X_t = x_t|q_{x_{<t}}) = p(X_t = x_t|X_{<t} = x_{<t})$.

where $q_{x_{<t}} = f(x_{<t})$ is the unique state sub-sequence associated with $x_{<t}$.

- Quite strong and unrealistic requirements, but they guarantee accuracy nonetheless.
- Note $\{< t\} \triangleq \{1, 2, \ldots, t - 1\}$

**Proof.**

We have for all $t$:

\[
D(p(X_t|X_{<t})||p_h(X_t|X_{<t}))
\]

\[
= \sum_{x_{1:t}} p(x_{1:t}) \log \frac{p(x_t|x_{<t})}{p_h(x_t|x_{<t})}
\]

\[
= \sum_{x_{1:t}} p(x_{1:t}) \log \frac{p(x_t|x_{<t})}{\sum_{q_t} p_h(x_t|q_t)p_h(q_t|x_{<t})}
\]

\[
= \sum_{x_{1:t}} p(x_{1:t}) \log \frac{p(x_t|x_{<t})}{p_h(x_t|q_{x_{<t}})}
\]

\[
= \sum_{x_{1:t}} p(x_{1:t}) \log \frac{p(x_t|x_{<t})}{p(x_t|x_{<t})}
\]

\[
= 0
\]
Cont. Proof: Suff conds for HMMs Generative Accuracy

It then follows, using the above equation, that:

\[
0 = \sum_t D(p(X_t|X_{<t})||p_h(X_t|X_{<t})) = \sum_t \sum_{x:1:t} p(x:1:t) \log \frac{p(x_t|x_{<t})}{p_h(x_t|x_{<t})} = \sum_t \sum_{x:1:T} p(x:1:T) \log \frac{p(x_t|x_{<t})}{p_h(x_t|x_{<t})} \tag{11.61}
\]

\[
= \sum_{x:1:T} p(x:1:T) \log \prod_t p(x_t|x_{<t}) \prod_t p_h(x_t|x_{<t}) = \sum_{x:1:T} p(x:1:T) \log \frac{p(x:1:T)}{p_h(x:1:T)} \tag{11.62}
\]

\[
= D(p(X:1:T)||p_h(X:1:T)) \tag{11.63}
\]

- Strong conditions \(H(Q_t|X_{<t}) = 0\), not likely to happen in practice.
- Is this really what we need for an HMM, generative accuracy? We’ll address this again soon.

Mealy vs. Moore Machines

- Mealy vs. Moore finite state automata:

  Moore machine has only one possible output for each state — the output is a function only of the current state.

  Mealy machine has only one possible output for each state-input pair — given current state, the input determines both the next state and the current output.

  Mealy machine: given a current state, and a next state (determined by input), there is an output.

  Hence, we can think of a Mealy machine as one where the output is a function of the transition between two states.
Mealy vs. Moore HMMs

- Mealy (or Jelinek) HMMs are ones where the observation distribution is a function of the transition between states.
- Markov transition diagram often shown with symbols on the edges between state nodes (note, this is not a graphical model, rather a labeled state-transition diagram).

Moore (or Rabiner) HMMs are ones where the observation is a function of the state itself.
- Markov transition diagram often shown with symbols on the nodes of the diagram.

Two ways to show the pronunciation model for the word “and”

Left: Moore machine, Right: Mealy machine.
- The output label on each Mealy arcs is shown as input:output and $\epsilon$ means “null”
Mealy vs. Moore HMMs

- Graphical model for Moore-HMM: $(Q_t)$ is a parent of $X_t$ for each $t$.

- Graphical model for Mealy-HMM: $X_t$ has two parents, either $(Q_{t-1}, Q_t)$ or $(Q_t, Q_{t+1})$.

Mealy and Moore HMMs are equivalent in that they correspond to the same class of probability distributions.

Mealy → Moore conversion: cluster adjacent variables in the graphical model.

More → Mealy conversion: Use a separate arc for each possible output with observation randomness applied to the arc.

It is sometimes more efficient, in terms of number of states, to specify a language in terms of a Mealy HMM (see parenthesis matching example in the readings).
Transition matrix structure

- A matrix Markov chain topology (where zeros are) is crucial for good performance.
- Most common: strictly left-to-right (used for sub-word sequences):

```
c/ a/ a/ t/ c/ a/ a/ t/
```

- Generally: upper triangular:

\[
a_{ij} = 0 \text{ for } j < i
\]
\[
a_{NN} = 1, \quad a_{Ni} = 0
\]
\[
a_{11} = 0, \quad a_{12} = 1, \quad a_{1j} = 0, j > 2
\]

L2R property of matrix also helps discriminability — ability to distinguish one sound from another. More on this when we cover discriminative training.

HMM Training: Collecting Data

- One of the benefits of HMMs is that there are formal procedures for training them.
- We will discuss two general approaches, the EM algorithm (generative training under the maximum likelihood criterion), and various discriminative training approaches.
- In either case, it is necessary to have training data.
HMM Training: Forms of Data

- Training data $\mathcal{D} = \{(x_{1:T_i}^{(i)}, w^{(i)})\}_{i},$ where $x_{1:T_i}^{(i)}$ is a matrix of speech features and $w^{(i)}$ is the set of labels (word transcription) of speech.

- Unsupervised training data: when one has only $\mathcal{D}_u = \{x_{1:T_i}^{(i)}\}_{i}$

- Semi-supervised training data: when one has only $\mathcal{D}_u = \{x_{1:T_i}^{(i)}\}_{i}$ and $\mathcal{D}_s = \{(x_{1:T_i}^{(i)}, w^{(i)})\}_{j}$

- Extra language data $\mathcal{D}_\ell = \{w^{(i)}\}_{j}$ useful since HMMs are generative models. Acoustic and language data can have little overlap.

- Language data can be either: 1) sequence of words, 2) sequence of words where center time-mark is known, and 3) sequence of words where word boundaries are known.

Collecting speech data

- Unfortunately, collecting data of a particular in is not always easy.

DTW (and DP) is one early method people used to recognize speech, and is based on templates.