Lecture 12: Hidden Markov Models
Machine Learning

Andrew Rosenberg

March 12, 2010

Last Time

- Clustering

Today

- Hidden Markov Models

Dice Example

Imagine a game of dice.

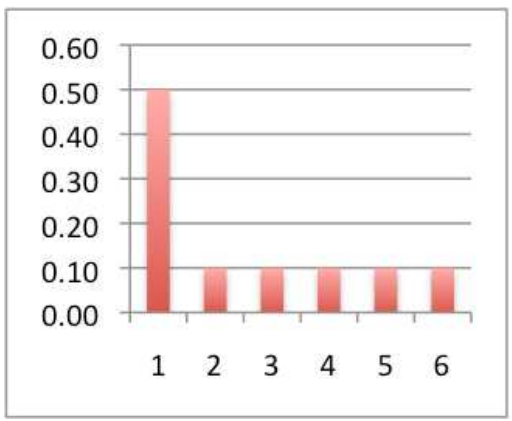
- When the croupier rolls 4,5,6 you win.
- When the croupier rolls 1,2,3 you lose.

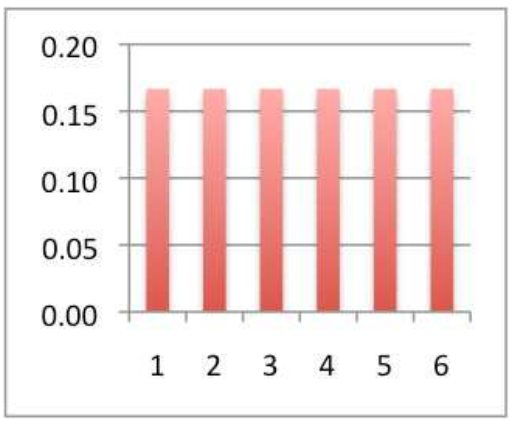
Model the likelihood of winning.

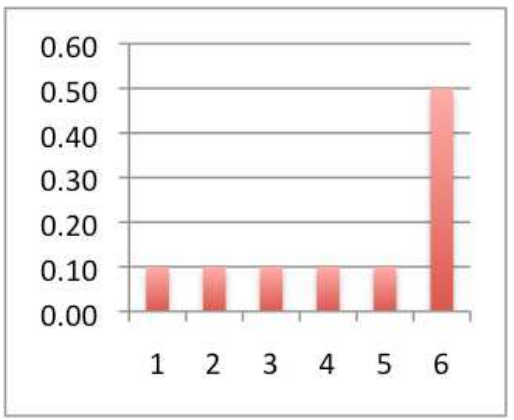
- IID multinomials

The Moody Croupier

- Now imagine that the croupier cheats.
- There are three dice.
  - One fair (**fair**)
  - One good for the house (**bad**)
  - One good for you (**good**)



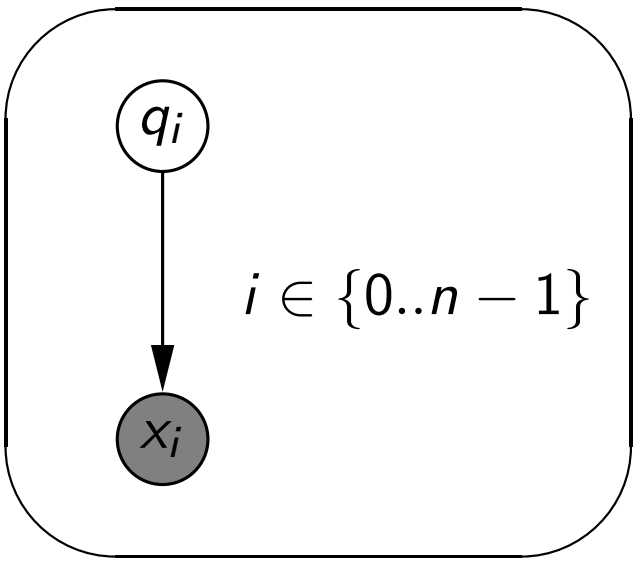




The Moody Croupier

Model the Likelihood of winning.

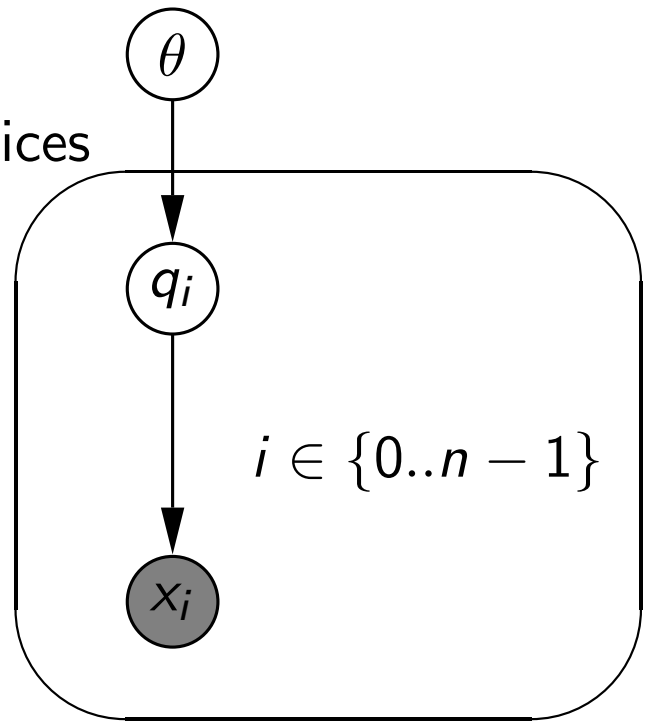
- IID multinomials
- Latent variable



The Moody Croupier

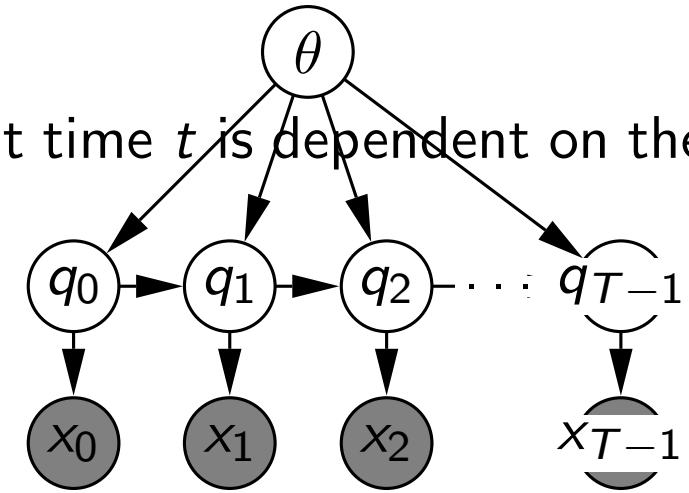
Model the Likelihood of winning.

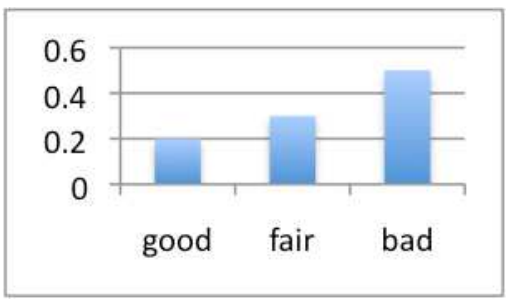
- IID multinomials
- Latent variable
- Allow a prior over die choices

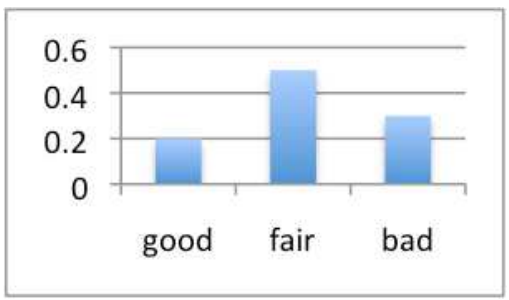


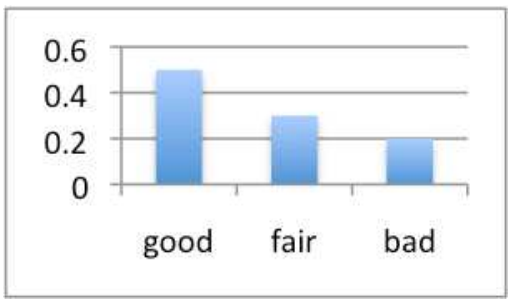
The Moody Croupier

- Now what if the dealer is moody?
- The dealer doesn't like to change the die that often
- The dealer doesn't like to switch from the **good** die to the **bad** die.
- No longer iid!
- The die he uses at time t is dependent on the die used at $t-1$

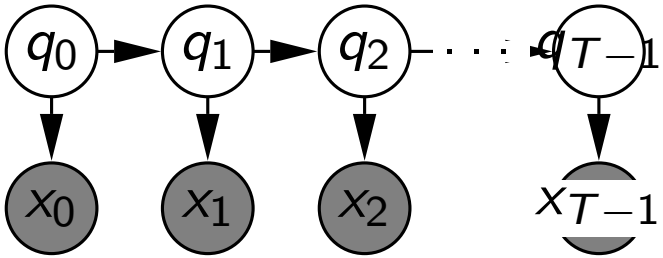








Sequential Modeling



- Temporal or sequence model.

Markov Assumption

- $future \perp\!\!\!\perp past | present$
- $p(q_t|q_{t-1}, q_{t-2}, q_{t-3}, ..., q_0) = p(q_t|q_{t-1})$

Get the overall likelihood from the graphical model.

$$p(\mathbf{x}) = p(q_0) \prod_{t=1}^{T-1} p(q_t|q_{t-1}) \prod_{t=0}^{T-1} p(x_t|q_t)$$

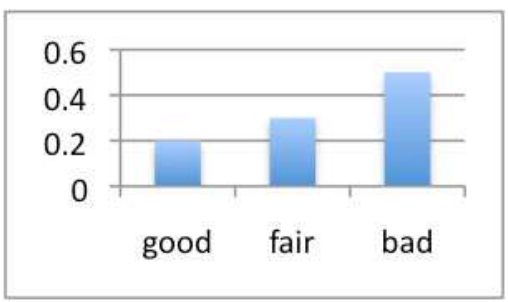
Sequential Modeling

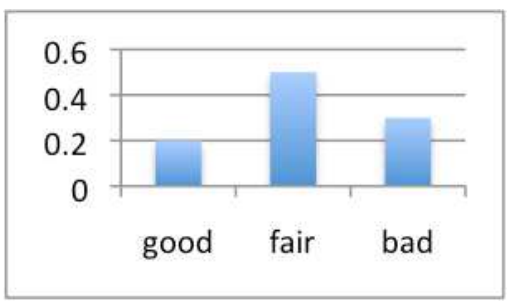
- $future \perp\!\!\!\perp past | present$
- $p(q_t|q_{t-1}, q_{t-2}, q_{t-3}, ..., q_0) = p(q_t|q_{t-1})$

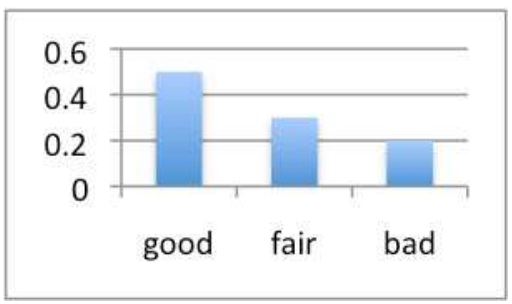
Get the overall likelihood from the graphical model.

$$p(\mathbf{x}) = p(q_0) \prod_{t=1}^{T-1} p(q_t|q_{t-1}) \prod_{t=0}^{T-1} p(x_t|q_t)$$

- $p(q_t|q_{t-1})$?



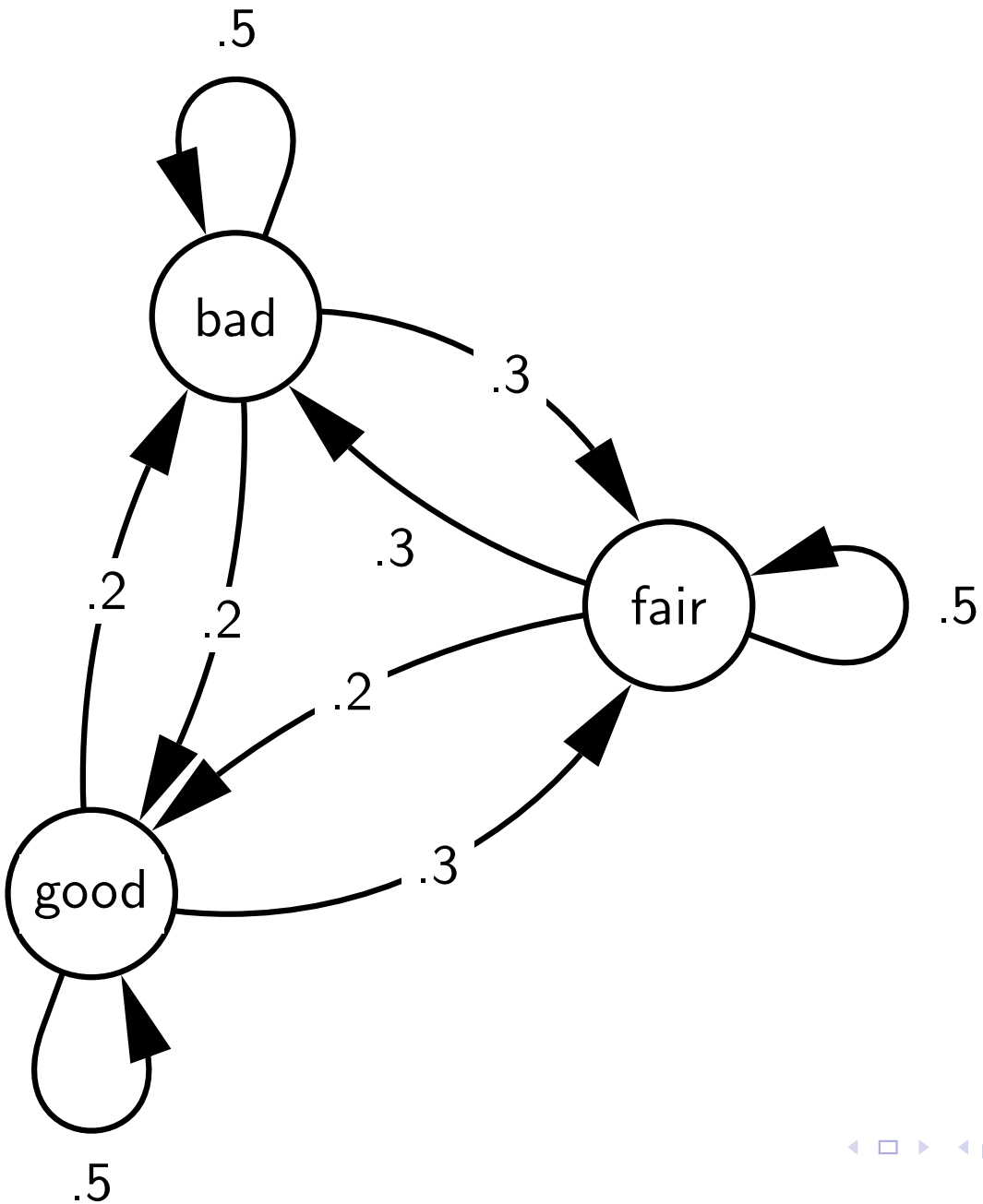




HMMs as state machines

- HMMs have two variables: **state** q and **emission** y
- In general the **state** is an unobserved latent variable.
- Can consider HMMs as stochastic automata – weighted finite state machines.

HMM state machine



HMMs as state machines

- HMMs have two variables: **state** q and **emission** x
- In general the **state** is an unobserved latent variable.
- No observation of q directly. Only a related emission distribution. "doubly-stochastic automaton".

HMM Applications

- **Speech Recognition** (Rabiner): phonemes from audio cepstral vectors
- **Language** (Jelinek): part of speech tag from words
- **Biology** (Baldi): splice site from gene sequence
- **Gesture** (Starner): word from hand coordinates
- **Emotion** (Picard): emotion from EEG

Types of Variables

- Continuous States
  - E.g. Kalman filters
  - $p(q_t|q_t - 1) = N(q_t|Aq_{t-1}, Q)$
- **Discrete** States
  - E.g., Finite state machine
  - $p(q_t|q_t - 1) = \prod_{i=0}^{M-1} \prod_{j=0}^{M-1} [\alpha_{ij}]^{q_{t-1}^i q_t^j}$
- Continuous Observations
  - E.g. time series data
  - $p(x_t|q_t) = N(x_t|\mu_{q_t}, \Sigma_{q_t})$
- **Discrete** Observations
  - E.g. strings
  - $p(x_t|q_t) = \prod_{i=0}^{M-1} \prod_{j=0}^{N-1} [\eta_{ij}]^{q_t^i x_t^j}$

HMM Parameters

M states and N-class observations
Complete likelihood from Graphical Model

$$p(\mathbf{x}) = p(q_0) \prod_{t=1}^{T-1} p(q_t|q_{t-1}) \prod_{t=0}^{T-1} p(x_t|q_t)$$

Marginalize over unobserved hidden states

$$p(x) = \sum_{q_0} \cdots \sum_{q_{T-1}} p(q, x)$$

CPTs are reused: $\theta = \{\pi, \eta, \alpha\}$

$$p(q_t|q_{t-1}) = \prod_{i=0}^{M-1} \prod_{j=0}^{M-1} [\alpha_{ij}]^{q_{t-1}^i x_t^j} \qquad \sum_{j=0}^{M-1} \alpha_{ij} = 1$$

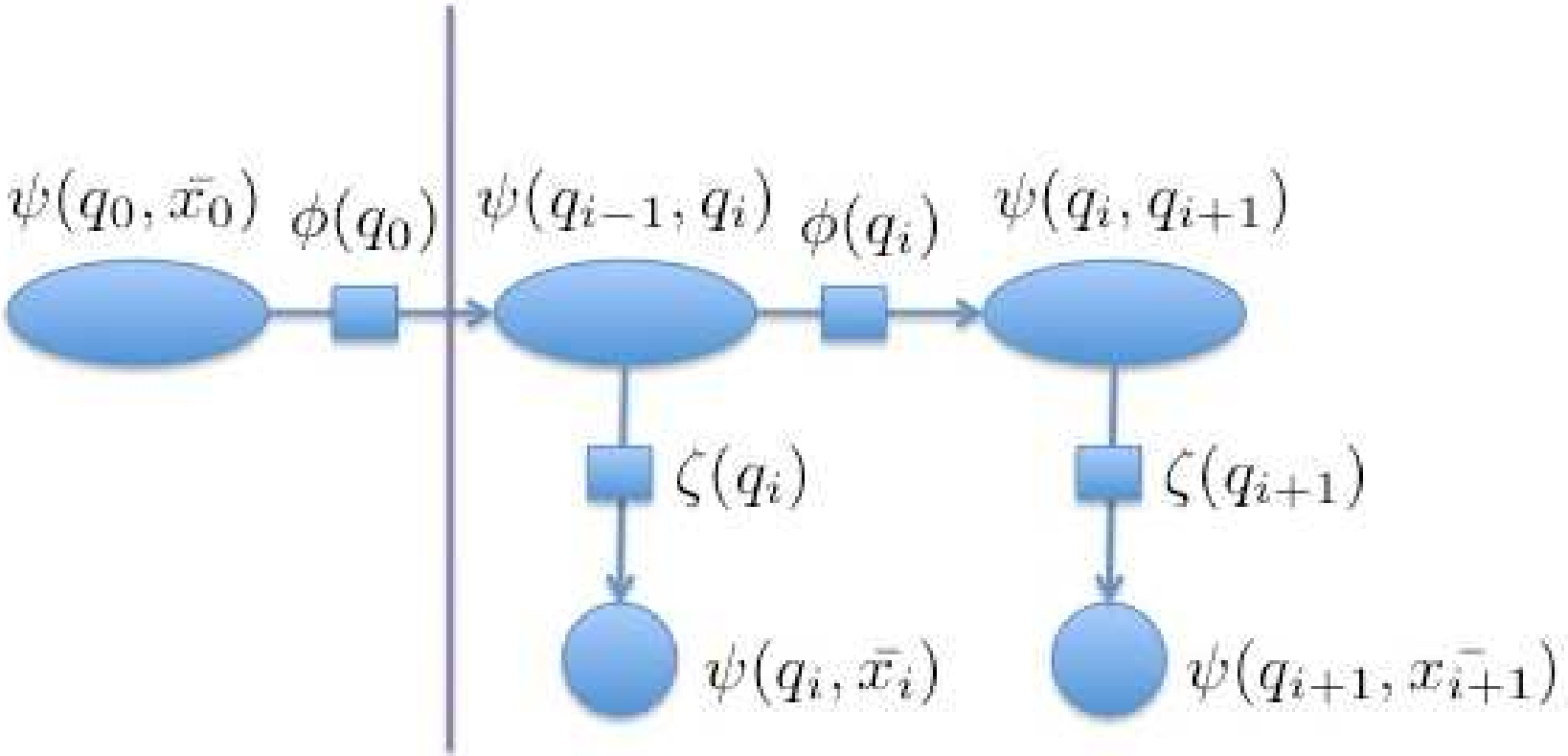
$$p(x_t|q_t) = \prod_{i=0}^{M-1} \prod_{j=0}^{N-1} [\eta_{ij}]^{q_t^i x_t^j} \qquad \sum_{j=0}^{N-1} \eta_{ij} = 1$$

$$p(q_0) = \prod_{i=0}^{M-1} [\pi_i]^{q_0^i} \qquad \sum_{j=0}^{M-1} \pi_i = 1$$

HMM Operations

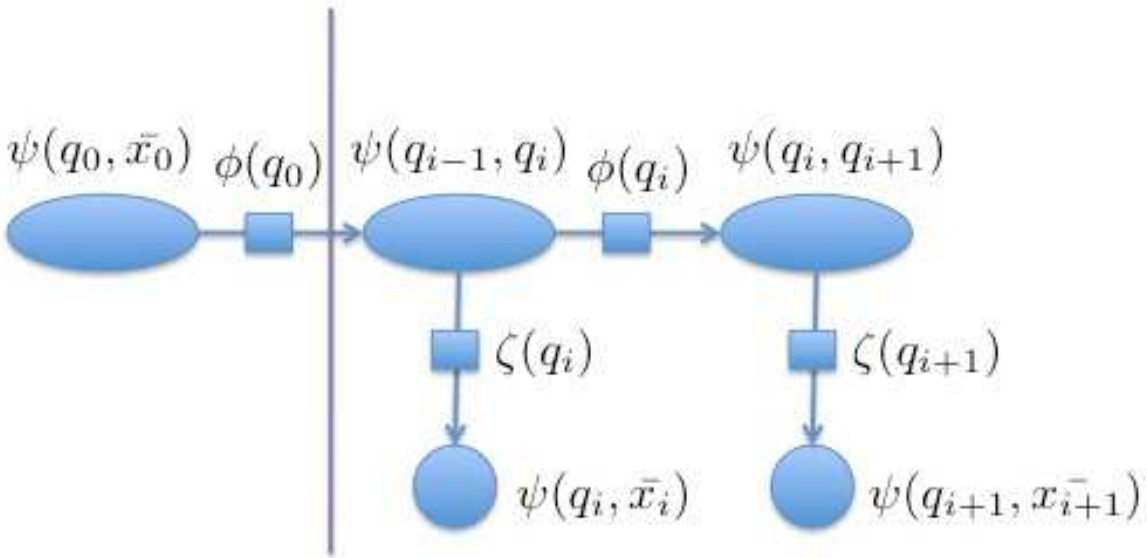
- Evaluate
  - Evaluate the likelihood of a model given data.
- Decode
  - Identify the most likely sequence of states
- Max Likelihood
  - Estimate the parameters.

Junction Tree



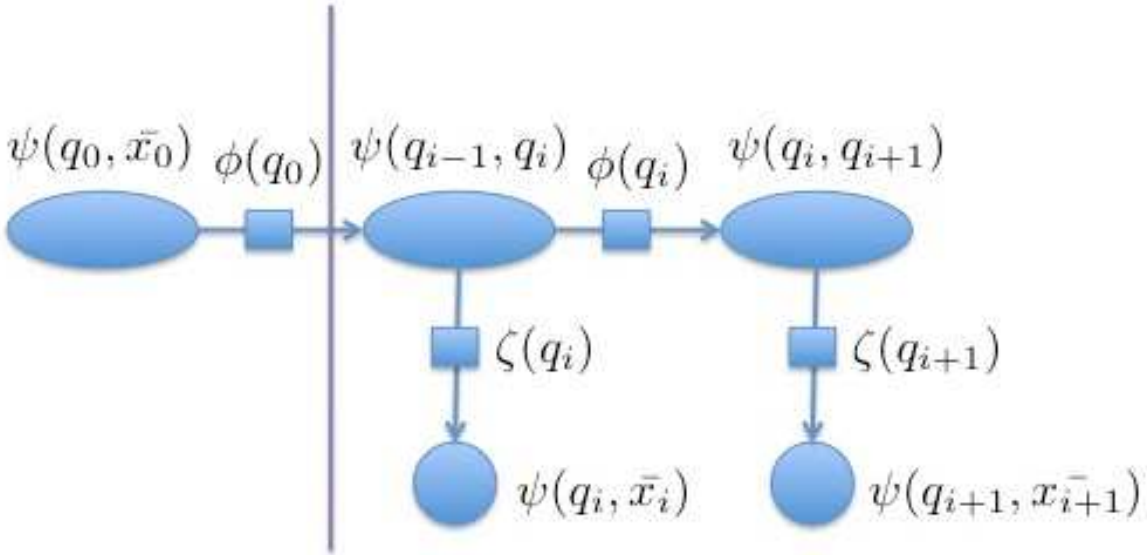
JTA on HMMs

Initialization



$$
\begin{aligned}
\psi(q_0, x_0) &= p(q_0)p(x_0, q_0) \\
\psi(q_t, q_{t+1}) &= p(q_{t+1}|q_t) = A_{q_t, q_{t+1}} \\
\psi(q_t, x_t) &= p(x_t|q_t) \\
Z &= 1 \\
\phi(q_t) &= 1 \\
\zeta(q_t) &= 1
\end{aligned}
$$

Update

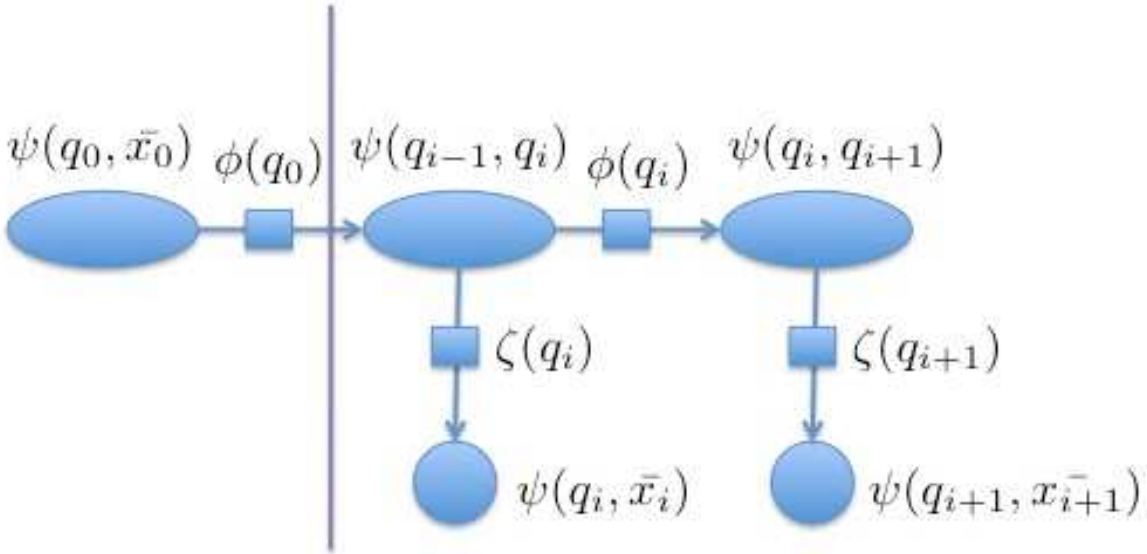


Collect up from leaves – don't change zeta separators.

$$\zeta^*(q_t) = \sum_{x_t} \psi(q_t, x_t) = \sum_{x_t} p(x_t|q_t) = 1$$

$$\psi^*(q_{t-1}, q_t) = \frac{\zeta^*(q_t)}{\zeta(q_t)} \psi(q_{t-1}, q_t) = \psi(q_{t-1}, q_t)$$

Update



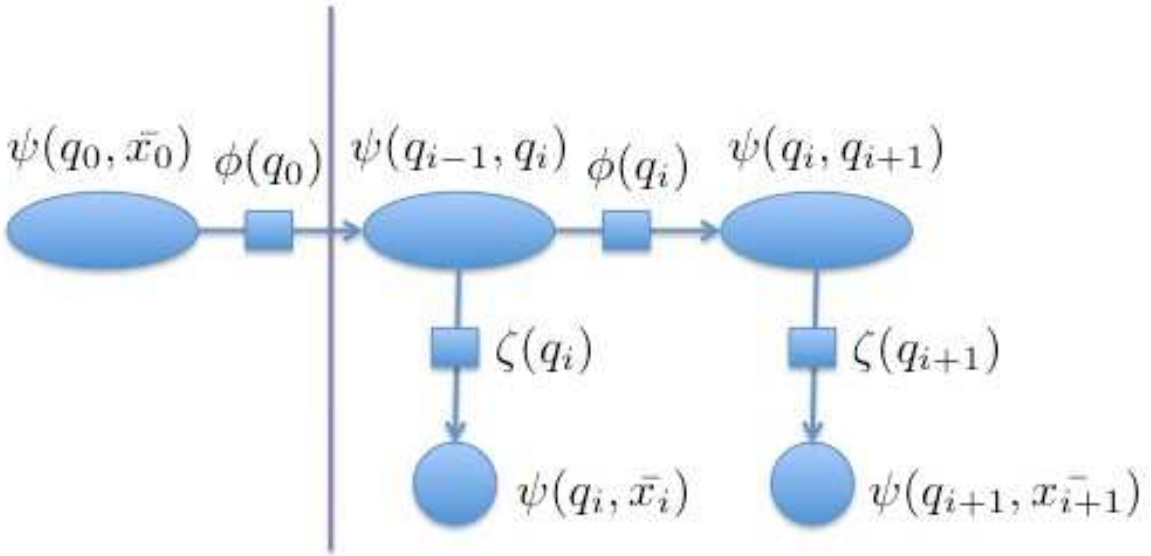
Collect left-right over phi – state sequence becomes marginals.

$$\phi^*(q_0) = \sum_{x_0} \psi(q_0, x_0) = p(q_0)$$

$$\phi^*(q_t) = \sum_{q_{t-1}} \psi(q_t, q_{t-1}) = p(q_t)$$

$$\psi^*(q_0, q_1) = \frac{\phi^*(q_0)}{\phi(q_0)} \psi(q_0, q_1) = p(q_0, q_1)$$

Distribute



Distribute to separators

$$\zeta^{**}(q_t) = \sum_{q_{t-1}} \psi^*(q_{t-1}, q_t) = \sum_{q_{t-1}} p(q_{t-1}, q_t) = p(q_t)$$

$$\psi^{**}(q_t, x_t) = \frac{\zeta^{**}(q_t)}{\zeta^*(q_t)} \psi(q_t, x_t) = \frac{p(q_t)}{1} p(x_t|q_t) = p(x_t, q_t)$$

Introduction of Evidence

$$p(q|\bar{x}) = p(q_0) \prod_{t=0}^{T-1} p(q_t|q_{t-1}) \prod_{t=0}^{T-1} p(\bar{x}_t|q_t)$$

- Observe a sequence of data.
- Potentials become slices

$$\begin{aligned} \psi(q_t, \bar{x}_t) &= p(\bar{x}_t|q_t) \\ \zeta^*(q_t) &= \psi(q_t, \bar{x}_t) = p(\bar{x}_t|q_t) \\ \zeta^*(q_t) &\neq \sum_{x_t} \psi(q_t, \bar{x}_t) \end{aligned}$$

- Collect zeta separators bottom up
  - $\zeta^*(q_t) = \psi(q_t, \bar{x}_t) = p(\bar{x}_t|q_t)$
- Collect phi separators to the right
  - $\phi^*(q_0) = \sum_{x_0} \psi(q_0, \bar{x}_0)\delta(x_0 - \bar{x}_0) = p(q_0, \bar{x}_0)$

JTA collect

- Collecting up and to the left, updating potentials by left and bottom separators

$$\psi^*(q_t, q_{t+1}) = \frac{\phi^*(q_t)}{1}\frac{\zeta^*(q_{t+1})}{1}\psi(q_t, q_{t+1}) = \phi^*(q_t)p(\bar{x}_{t+1}|q_{t+1})\alpha_{q_t q_{t+1}}$$

$$\phi^*(q_{t+1}) = \sum_{q_t}\psi^*(q_t, q_{t+1}) = \sum_{q_t}\phi^*(q_t)p(\bar{x}_{t+1}|q_{t+1})\alpha_{q_t q_{t+1}}$$

Note:

$$\begin{aligned}
\phi^*(q_0) &= p(\bar{x}_0, q_0) \\
\phi^*(q_1) &= \sum_{q_0} p(\bar{x}_0, q_0)p(\bar{x}_1|q_1)p(q_1|q_0) = p(\bar{x}_0, \bar{x}_1, q_1) \\
\phi^*(q_2) &= \sum_{q_1} p(\bar{x}_0, \bar{x}_1, q_0)p(\bar{x}_2|q_2)p(q_2|q_1) = p(\bar{x}_0, \bar{x}_1, \bar{x}_2, q_2) \\
\phi^*(q_{t+1}) &= \sum_{q_t} p(\bar{x}_0, \ldots, \bar{x}_{t+1}, q_{t+1})p(\bar{x}_{t+1}|q_{t+1})p(q_{t+1}|q_t) = p(\bar{x}_0, \ldots, \bar{x}_{t+1}, q_{t+1})
\end{aligned}$$

Evaluation

- Compute the likelihood of the sequence.
- Collection is sufficient.

From previous slide

$$\phi^*(q_{t+1}) = \sum_{q_t} p(\bar{x_0}, \ldots, \bar{x_t}, q_t) p(\bar{x_{t+1}}, q_{t+1}) p(q_{t+1}|q_t) = p(\bar{x_0}, \ldots, \bar{x_{t+1}}, q_{t+1})$$

So the rightmost node gives:

$$\phi^*(q_{T-1}) = p(\bar{x_0}, \ldots, \bar{x_{T-1}}, q_{T-1})$$

The likelihood just requires marginalization over q_{T-1}.

$$p(\bar{x_0}, \ldots, \bar{x_{T-1}}) = \sum_{q_{T-1}} p(\bar{x_0}, \ldots, \bar{x_{T-1}}, q_{T-1}) = \sum_{q_{T-1}} \phi^*(q_{T-1})$$

Distribute

But the potentials cannot be read as marginals without the Distribute step of the JTA.

- Last state of collection

$$\psi^*(q_{T-2}, q_{T-1}) = \frac{\phi^*(q_{T-2})}{1} \frac{\zeta^*(q_{T-1})}{1} \psi(q_{T-2}, q_{T-1}) = \phi^*(q_{T-2}) p(\bar{x}_{T-1}|q_{T-1}) \alpha_{q_{T-2} q_{T}}$$

- Distribute ** along the state nodes to the left.
- Distribute ** down from state nodes to observation nodes.

Update parameters.

$$\begin{aligned}
\psi^{**}(q_{T-2}, q_{T-1}) &= \psi^*(q_{T-2}, q_{T-1}) \\
\phi^{**}(q_t) &= \sum_{q_{t+1}} \psi^{**}(q_t, q_{t+1}) \\
\zeta^{**}(q_{t+1}) &= \sum_{q_t} \psi^{**}(q_t, q_{t+1}) \\
\psi^{**}(q_t, q_{t+1}) &= \frac{\phi^{**}(q_{t+1})}{\phi^*(q_{t+1})} \psi^*(q_t, q_{t+1})
\end{aligned}$$

Decoding

- **Decode**: Given $x_0, \ldots x_{T-1}$ identify the most likely $q_0, \ldots q_{T-1}$.
- Now that JTA is finished we have marginals in the potentials and separators

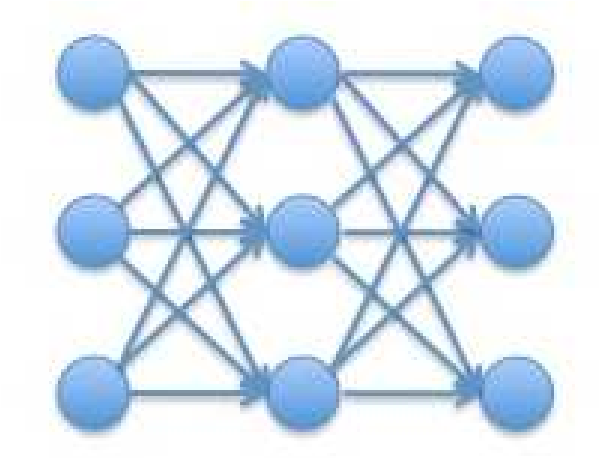
$$\begin{aligned} \phi^{**}(q_t) &\propto p(q_t|\bar{x}_0, \ldots, \bar{x}_{T-1}) \\ \zeta^{**}(q_{t+1}) &\propto p(q_{t+1}|\bar{x}_0, \ldots, \bar{x}_{T-1}) \\ \psi^{**}(q_t, q_{t+1}) &\propto p(q_t, q_{t+1}|\bar{x}_0, \ldots, \bar{x}_{T-1}) \end{aligned}$$

- Need to find the most likely path from q_0 to q_{T-1}
- Argmax JTA.
  - Run JTA but rather than sums in the update rule, use the max operator.
  - Then find the largest entry in the separators

$$\hat{q}_t = \operatorname*{argmax}_{q_t} \phi^{**}(q_t)$$

Viterbi Decoding

- Finding an optimal state sequence can be intractable.
- There are M^T possible paths, for M states and T time steps.
  - T can easily be on the order of 1000 in speech recognition.



- Construct a Lattice of state transitions

Viterbi Decoding

- Only continue to explore paths with likelihood greater than some threshold, or only continue to explore the top N-paths
- Also known as **beam search**
- Polynomial time algorithm to *approximately* decode a lattice.

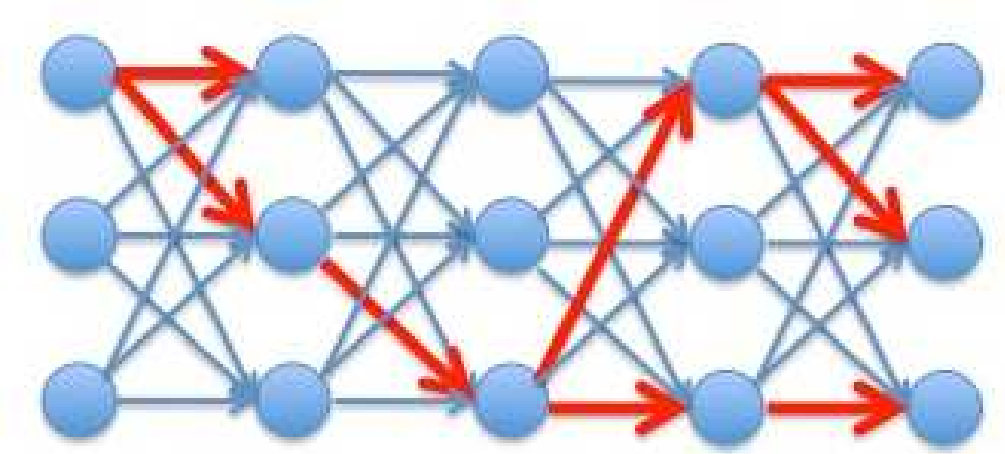
Algorithm:

- Initialize paths at every state.
- For each transition follow only the most likely edge.

or

- Initialize paths at every state.
- For each transition follow only those paths that have a likelihood over some threshold.

Viterbi decoding



Maximum Likelihood

Training parameters with observed states.

- Maximum likelihood (as ever).

$$
\begin{aligned}
l(\theta) &= \log(p(q, \bar{x})) \\
&= \log\left(p(q_0)\prod_{t=1}^{T-1} p(q_t|q_{t-1})\prod_{t=0}^{T-1} p(\bar{x}_i|q_i)\right) \\
&= \log p(q_0) + \sum_{t=1}^{T-1}\log p(q_t|q_{t-1}) + \sum_{t=0}^{T-1}\log p(\bar{x}_i|q_i) \\
&= \log\prod_{i=0}^{M-1}[\pi_i]^{q_0^i} + \sum_{t=1}^{T-1}\log\prod_{i=0}^{M-1}\prod_{j=0}^{M-1}[\alpha_{ij}]^{q_{t-1}^i q_t^j} + \sum_{t=0}^{T-1}\log\prod_{i=0}^{M-1}\prod_{j=0}^{N-1}[\eta_{ij}]^{q_t^i \bar{x}_t^j} \\
&= \sum_{i=0}^{M-1} q_0^i \log\pi_i + \sum_{t=1}^{T-1}\sum_{i=0}^{M-1}\sum_{j=0}^{M-1} q_{t-1}^i q_t^j \log\alpha_{ij} + \sum_{t=0}^{T-1}\sum_{i=0}^{M-1}\sum_{j=0}^{N-1} q_t^i \bar{x}_t^j \log\eta_{ij}
\end{aligned}
$$

- Introduce Lagrange multipliers, take partials, set to zero.

Maximum Likelihood

Training parameters with observed states.

- Maximum likelihood – as ever.

$$l(\theta) = \sum_{i=0}^{M-1} q_0^i \log \pi_i + \sum_{t=1}^{T-1} \sum_{i=0}^{M-1} \sum_{j=0}^{M-1} q_{t-1}^i q_t^j \log \alpha_{ij} + \sum_{t=0}^{T-1} \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} q_t^i \bar{x}_t^j \log \eta_{ij}$$

- Introduce Lagrange multipliers, take partials, set to zero.

$$\sum_{i=0}^{M-1} \pi_i = 1$$

$$\sum_{j=0}^{M-1} \alpha_{ij} = 1$$

$$\sum_{j=0}^{N-1} \eta_{ij} = 1$$

$$\hat{\pi}_i = q_0^i$$

$$\hat{\alpha}_{ij} = \frac{\sum_{t=0}^{T-2} q_t^i q_{t+1}^j}{\sum_{k=1}^{M-1} \sum_{t=0}^{T-2} q_t^i q_{t+1}^k}$$

$$\hat{\eta}_{ij} = \frac{\sum_{t=0}^{T-1} q_t^i \bar{x}_t^j}{\sum_{k=1}^{M-1} \sum_{t=0}^{T-1} q_t^i \bar{x}_t^k}$$

Expectation Maximization

- However, we may not have observed state sequences.
  - The Moody Croupier
- Need to do unsupervised learning (clustering) on the states.
  - Maximize the **Expected** likelihood given a guess for $p(q)$
  - **Expectation Maximization** – Covered when we move to unsupervised techniques

Bye

- Next
  - Perceptron and Neural Networks