Final: 12.14

Machine Learning

4771

Instructor: Tony Jebara

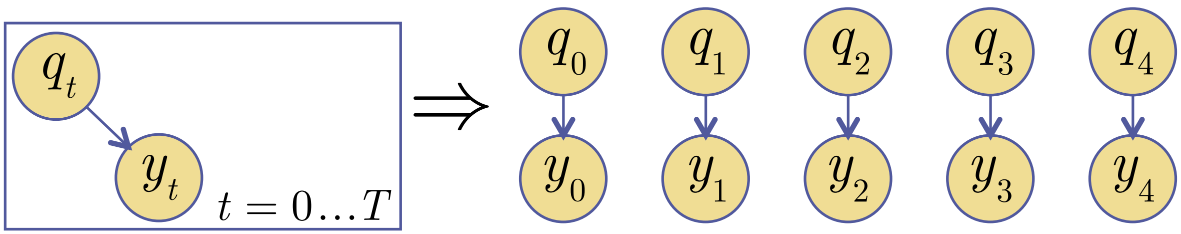
Topic 19

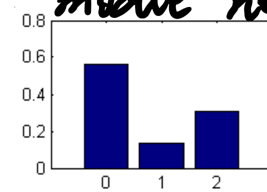
- Hidden Markov Models
- HMMs as State Machines & Applications
- HMMs Basic Operations
- HMMs via the Junction Tree Algorithm

Hidden Markov Models

- A great application of Junction Tree Algorithm and EM
- Recall mixture of Gaussians model on IID data

Mixture of Gaussian modle needs IID condition.



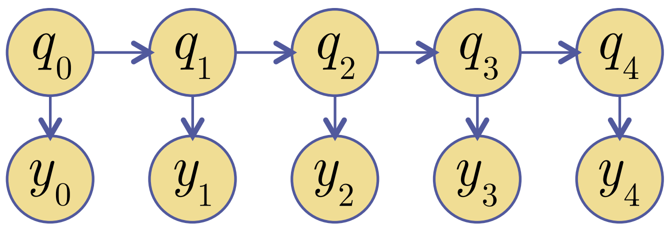


$$p\left(q_t \mid \theta\right) = \begin{bmatrix} .5 & .2 & .3 \end{bmatrix}$$

- Example: location data of a single parent as a mixture of Gaussians
- Parent has 3 internal states:
  q={home,daycare,work}
- Based on q, sample from appropriate Gaussian mean and covariance to get
  y=(latitude,longitude)

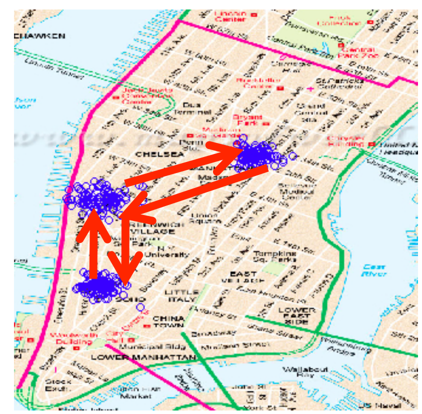
Hidden Markov Models

- Parent drops child at daycare before & after work. Not IID!



q={1=home,2=daycare,3=work}

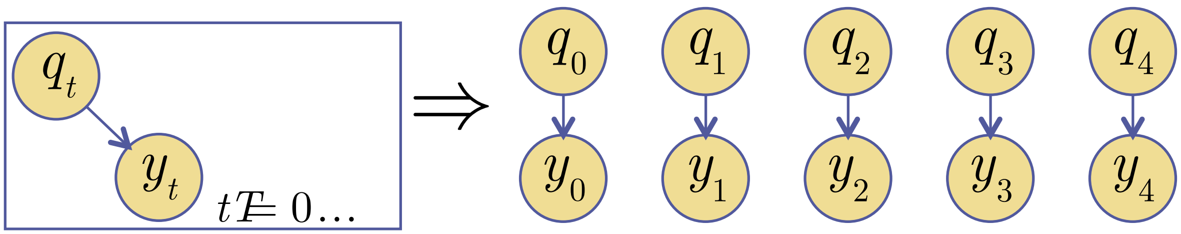
$$p\left(q_t \mid q_{t-1}\right) = \begin{array}{cc} \begin{bmatrix} 0.8 & 0.2 & 0 \\ 0.1 & 0.8 & 0.1 \\ 0 & 0.2 & 0.8 \end{bmatrix} & \begin{array}{l} q_{t-1}=1 \\ q_{t-1}=2 \\ q_{t-1}=3 \end{array} \\ \begin{array}{ccc} q_t=1 & q_t=2 & q_t=3 \end{array} & \end{array}$$

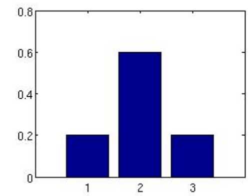


- Have dependence on previous state
- Can't go straight from home to work!
- Now, order of $y_0,\ldots,y_T$ matters (in IID order doesn't matter)

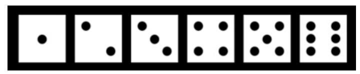
Hidden Markov Models

•Consider mixture of multinomials (dice) y={1,2,3,4,5,6}

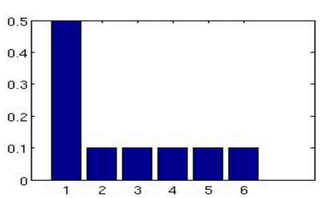




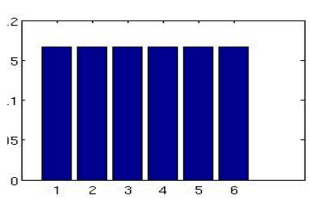
$$p\left(q_t \mid \theta\right) = \begin{bmatrix} .2 & .6 & .2 \end{bmatrix}$$



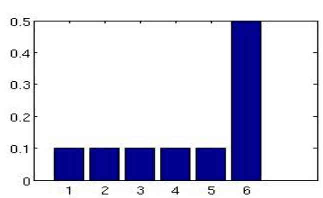
•Example: a crooked casino croupier using mixture of dice.
•You win if he rolls 1,2,3. You lose he rolls 4,5,6.
•Croupier has 3 internal states q={helpful,fair,adversarial}
•Based on q, sample different ‘dice’ multinomial



1=helpful



2=fair

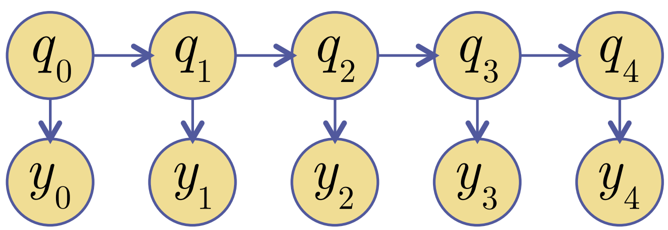


3=adversarial

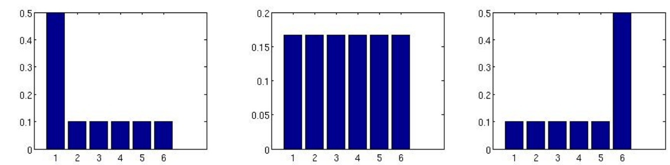
Hidden Markov Models

- But if the dealer has a memory or mood? Not IID!

  5646166166 4321534161414341634 1113114121



p(y|q)



q={1=helpful,2=fair,3=adversarial}
y={1,2,3,4,5,6}

$$p\left(q_t \mid q_{t-1}\right) = \begin{array}{cc} \begin{bmatrix} 0.8 & 0.1 & 0.1 \\ 0.1 & 0.8 & 0.1 \\ 0.1 & 0.1 & 0.8 \end{bmatrix} & \begin{array}{l} q_{t-1} = 1 \\ q_{t-1} = 2 \\ q_{t-1} = 3 \end{array} \\ \begin{array}{ccc} q_t = 1 & q_t = 2 & q_t = 3 \end{array} & \end{array}$$

$p(q_t=1 \mid q_{t-1}=1) = 0.8$

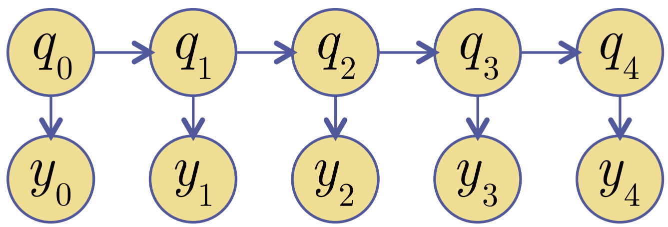
$p(q_t=2 \mid q_{t-1}=1) = 0.1$

$p(q_t=3 \mid q_{t-1}=1) = 0.1$

- If you tip, dealer starts to like you and rolls the helpful die
- Dealer has a memory of his mood and last type of die q_{t-1}
- Will often use same die for q^t as was rolled before…
- Now, order of $y_0,\ldots,y_T$ matters (if IID order doesn't matter)

Hidden Markov Models

- Since next choice of the dice depends on previous one…



Order of $y_0,\ldots,y_T$ matters
Temporal or sequence model!

- Add left-right arrows. This is a hidden Markov model
- Markov: $\quad future \perp\!\!\!\perp past \mid present$ This is what markov means.

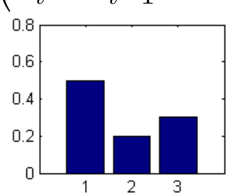
$$p\left(q_t \mid q_{t-1}, q_{t-2}, \ldots, q_1, q_0\right) = p\left(q_t \mid q_{t-1}\right)$$

- From graph, have the following general pdf:

$$p(q,y) = p\left(X_U\right) = p\left(q_0\right)\prod_{t=1}^{T} p\left(q_t \mid q_{t-1}\right)\prod_{t=0}^{T} p\left(y_t \mid q_t\right)$$

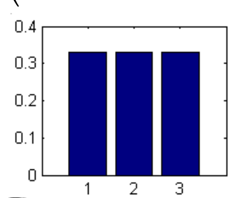
- So $p(q_t)$ depends on previous state q_{t-1} …

$p\left(q_t \mid q_{t-1} = 1\right)$



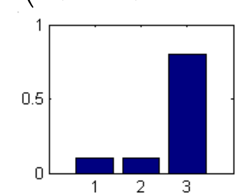
$\sum_{q_t} p(q_t \mid q_{t-1}=1)=1$

$p\left(q_t \mid q_{t-1} = 2\right)$



$\sum_{q_t} p(q_t \mid q_{t-1}=2)=1$

$p\left(q_t \mid q_{t-1} = 3\right)$



$\sum_{q_t} p(q_t \mid q_{t-1}=3)=1$

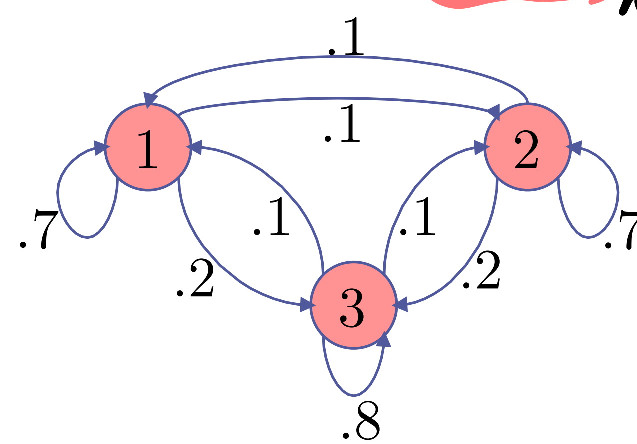
This is observed.

HMMs as State Machines

- HMMs have two variables: state q and emission y
- Typically, we don't know q (hidden variable 1,2,3,?)

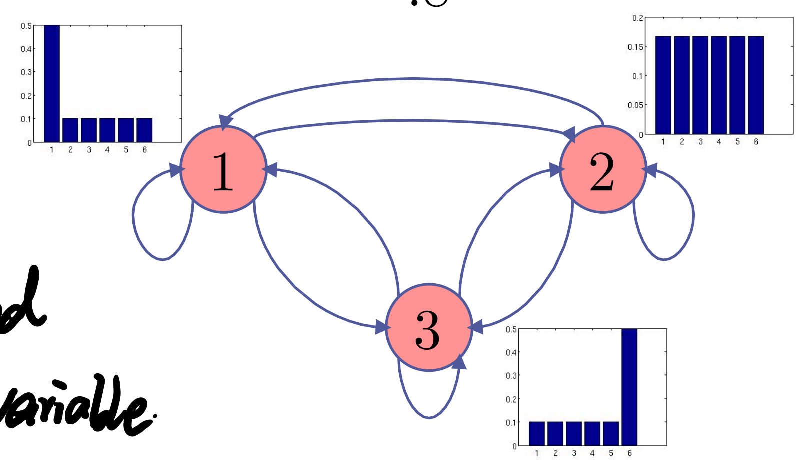
This is hidden
hidden state.

- HMMs are like stochastic automata or finite state machines…
  next state depends on previous one…
  (helpful, fair, adversarial)



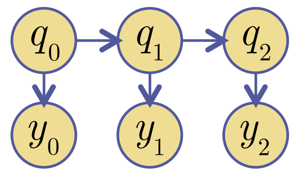
- Can't observe state q directly, just a random related emission y outcome (dice roll) so… doubly-stochastic automaton

y is observed
state: q is hidden variable.

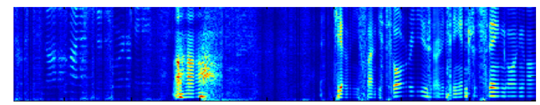


HMM Applications

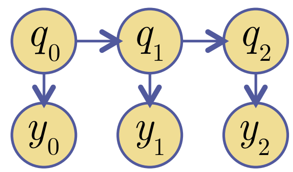
- Speech Recognition
  phonemes from
  audio cepstral vectors



Ba-ra-kk-oo-oo-dd-ah



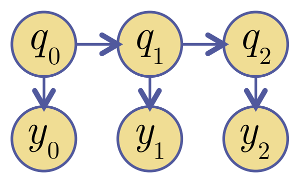
- Language Parsing
  parts of speech
  from words



Noun + Verb + Noun

John Ate Pizza

- Genomics
  splice site from
  gene sequence



-Intron-|-Exon-|-Promoter-

GATTACATTATACCACCATACG

Tony Jebara, Columbia University

① Variable size of state q: $R^{1\times M}$

Hidden parameter: State q

HMMs: Parameters

② Variable size of observed emission y: $R^{1\times N}$

- We focus on HMMs with: discrete state q (of size M)
  discrete emission y (of size N)
- Input will be arbitrary length string: $y_0,\ldots,y_T$ T: length of Markov model.
- The pdf or (complete) likelihood is:

$$P(X_{TT}) = p(q,y) = p(q_0)\prod_{t=1}^{T} p(q_t \mid q_{t-1})\prod_{t=0}^{T} p(y_t \mid q_t)$$

- We don't know hidden states, the incomplete likelihood is:

$$p(y) = \sum_{q_0}\cdots\sum_{q_T} p(q,y)$$

α_{ij}: Probability of transition from status j to i

- Assume HMM is stationary, tables are repeated: $\theta = \{\pi,\eta,\alpha\}$

α: $p(q_t \mid q_{t-1})$

η: $p(y_t \mid q_t)$

π: $p(q_0)$

$\sum_{y_t} p(y_t \mid q_t) = 1$

$$p(q_t \mid q_{t-1}) = \prod_{i=1}^{M}\prod_{j=1}^{M}\left[\alpha_{ij}\right]^{q_{t-1}^i q_t^j} \qquad \sum_{j=1}^{M}\alpha_{ij} = 1 \qquad M\times M$$

$$p(y_t \mid q_t) = \prod_{i=1}^{M}\prod_{j=1}^{N}\left[\eta_{ij}\right]^{q_t^i y_t^j} \qquad \sum_{j=1}^{N}\eta_{ij} = 1 \qquad M\times N$$

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

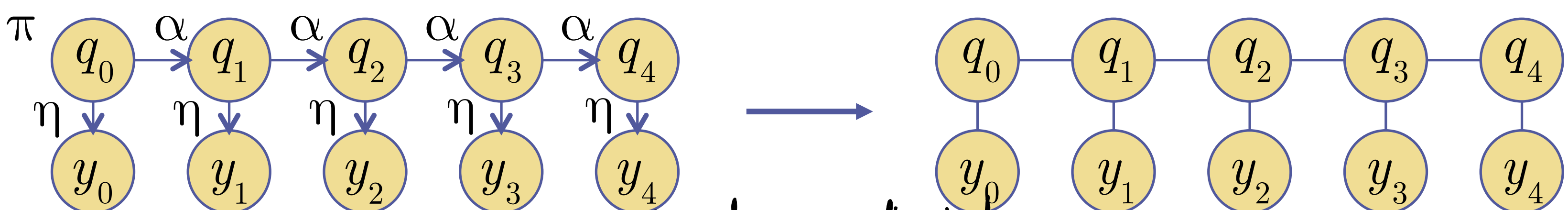
① Compute $p(X_{0:T}) = p(y,q) = p(q_0)\cdot\prod_{t=0}^{T} p(q_t|q_{t-1})\cdot\prod_{t=0}^{T} p(y_t|q_t)$

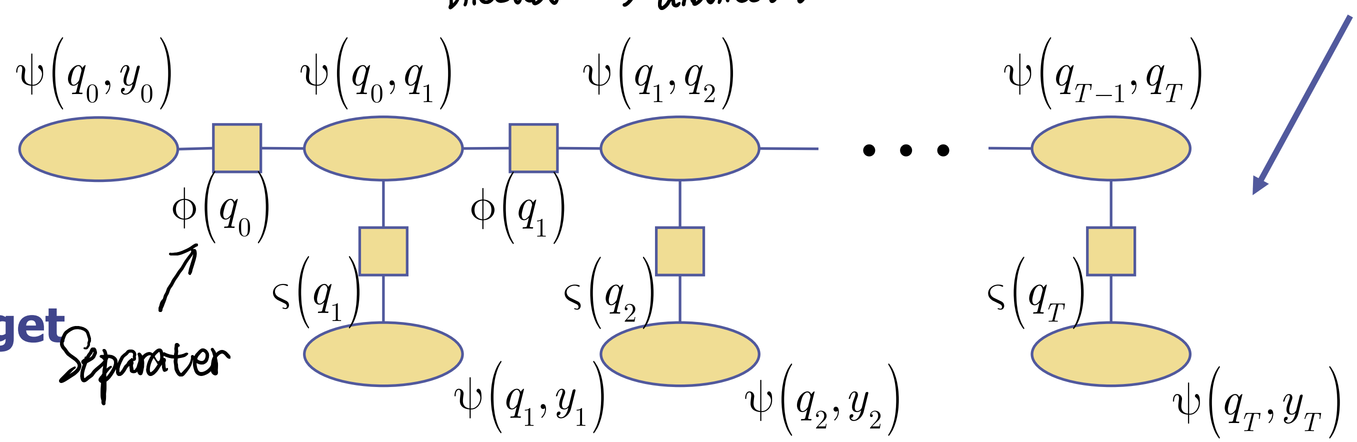
② Compute $p(y) = \sum_q p(X_{0:T}) = \sum_{q_0}\sum_{q_1}\cdots\sum_{q_T} p(X_{0:T})$

Tony Jebara, Columbia University

HMMs: Basic Operations

- Would like to do 3 basic things with our HMMs:
  1) **Evaluate**: given $y_0,\ldots,y_T$ & θ compute $p(y_1,\ldots,y_T)$ → π.
  2) **Decode**: given $y_0,\ldots,y_T$ & θ find $[q_0,\ldots,q_T$ or $p(q_0),\ldots,p(q_T)]$
  3) **Max Likelihood**: given $y_0,\ldots,y_T$ learn parameters θ
- Typically use Baum-Welch (α-β algo)… JTA is more general:





HMMs easily get Junction Tree

① $\varsigma^*(q_1)=\sum_{y_1}\psi(q_1,y_1)=\sum_{y_1}p(y_1|q_1)=1$
② $\psi^*(q_0,q_1)=\frac{\varsigma^*(q_1)}{\varsigma(q_1)}\cdot\psi(q_0,q_1)=\psi(q_0,q_1)$
No change

③ $\phi^*(q_0)=\sum_{y_0}\psi(y_0,q_0)=\sum_{y_0}p(y_0|q_0)=p(q_0)$
④ $\psi^*(q_0,q_1)=\frac{\phi^*(q_0)}{\phi(q_0)}\cdot\psi(q_0,q_1)=p(q_0)p(q_1|q_0)=p(q_0,q_1)$
collect

In this step: $\varsigma(q_T)=1$ → Distribute
Y① $\psi^*(q_{T-1},q_T)=p(q_{T-1},q_T)$, $\varsigma^*(q_T)=\sum_{q_{T-1}}\psi^*(q_{T-1},q_T)=p(q_T)$
Y② $\psi^{**}(q_T,y_T)=\frac{\varsigma^{**}(q_T)}{\varsigma^*(q_T)}\cdot\psi^*=\frac{p(q_T)}{1}\cdot p(y_T|q_T)=p(q_T,y_T)$

In this step:
$\Phi(q_0)=\varsigma(q_0)=1$

HMMs: JTA Init & Verify

α is given to us. Given to us.

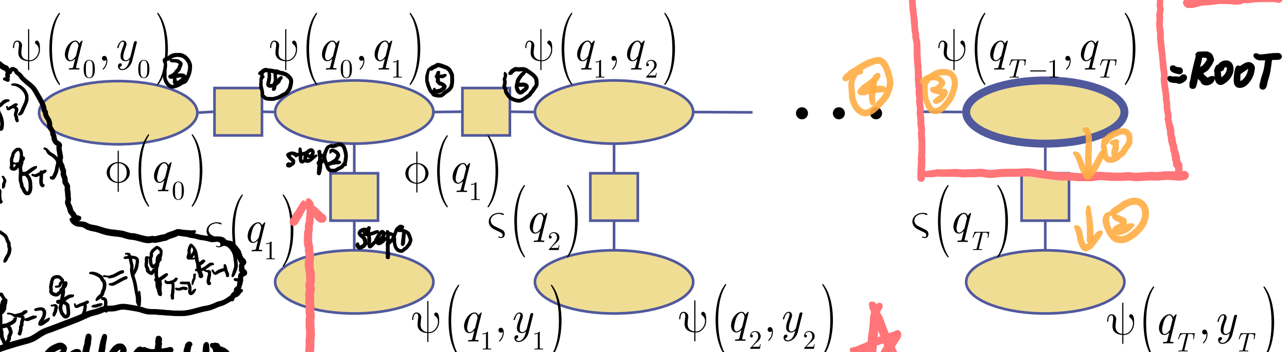
$\psi(q_1,y_1)$ collect makes no change

•Init: $\psi(q_0,y_0)=p(q_0)p(y_0\mid q_0)$ $\quad\psi(q_t,q_{t+1})=p(q_{t+1}\mid q_t)=\alpha_{q_t,q_{t+1}}$ $\quad\psi(q_t,y_t)=p(y_t\mid q_t)$

$Z=1$

$\phi(q_t)=1$

$\varsigma(q_t)=1$



Y3 $\phi^{**}(q_{T-1})=\sum_{q_T}\psi^*(q_{T-1},q_T)=\sum_{q_T}p(q_{T-1},q_T)=p(q_{T-1})$

Y4 $\psi^*(q_{T-2},q_{T-1})=\frac{\phi^{**}}{\phi^*}p(q_{T-2},q_{T-1})=p(q_{T-2},q_{T-1})$

No change

Collect up.

•Collect *up* (this time it actually doesn't change the zetas)

① $\varsigma^*(q_t)=\sum_{y_t}\psi(q_t,y_t)=\sum_{y_t}p(y_t\mid q_t)=1$ ② $\psi^*(q_{t-1},q_t)=\frac{\varsigma^{*}_{=1}}{\varsigma_{=1}}\psi(q_{t-1},q_t)=\psi(q_{t-1},q_t)$

•Collect *left-right* via phi's: change backbone to marginals

Send message: $\psi(q_0,y_0)\to\phi(q_0)$

③ $\phi^*(q_0)=\sum_{y_0}\psi(q_0,y_0)=p(q_0)$ ④ $\psi^*(q_0,q_1)=\frac{\phi^*}{\phi_{=1}}\psi(q_0,q_1)=p(q_0,q_1)$

Send message: $\phi(q_0)\to\psi(q_0,q_1)$

⑤ $\phi^*(q_t)=\sum_{q_{t-1}}\psi^*(q_{t-1},q_t)=p(q_t)$ ⑥ $\psi^*(q_{t-1},q_t)=\frac{p(q_{t-1})}{1}p(q_t\mid q_{t-1})=p(q_{t-1},q_t)$

•Distribute:
① $\varsigma^{**}(q_t)=\sum_{q_{t-1}}\psi^*(q_{t-1},q_t)=\sum_{q_{t-1}}p(q_{t-1},q_t)=p(q_t)$
② $\psi^{**}(q_t,y_t)=\frac{\varsigma^{**}}{\varsigma^*}\psi(q_t,y_t)=\frac{p(q_t)}{1}p(y_t\mid q_t)=p(y_t,q_t)$

...done!