

Hidden Markov Models (HMMs) (10/22/07)

Lecturer: Daniel Ting adapted from notes by Barbara Engelhardt

Scribes:

Explain that we are going to go SLOWLY over HMM material.

- Draw up HMM (q_t, y_t , where q_t is a multinomial rv.)
- Note that this is another latent variable model.
- Draw up conditional probabilities
- Draw up matrices A (transition matrix), π and η (emission matrix).
- Discuss conditional independencies, e.g., $p(y|q_t) = p(y_0, \dots, y_t|q_t)p(y_{t+1}, \dots, y_T|q_t)$.
- Goal 1: Calculate marginals on the model given θ_{ML}
- Goal 2: Estimate $\theta = A, \pi, \eta$ from a set of data (using EM)
- Goal 3: Find the maximum likelihood sequence of states for a given output sequence, given θ_{ML} (Viterbi)

1 Inference: Calculating marginals

- **Filtering problem:** $p(q_t|y_0, \dots, y_t)$
- **Prediction problem:** $p(q_t|y_0, \dots, y_s)$, where $s < t$.
- **Smoothing problem:** $p(q_t|y_0, \dots, y_u)$, where $u > t$.

Assume we have M states in the Markov chain. Furthermore, assume the Markov chain is *homogenous*.

1.1 Notation

- q_t is represented as a vector of length M with a single 1 (and the rest zero).
- $q_t^i = 1$ if the 1 appears in the i^{th} position
- The initial distribution π is defined by $\pi_0^i = p(q_0^i = 1)$. This is a vector of length M .
- Define $a_{i,j} = p(q_{t+1}^j = 1|q_t^i = 1)$, ie. the probability of transitioning from state i to j . This is an $M \times M$ matrix A which we have assumed is the same across all t .

For this part, we assume π_0 , A , and $p(y_t|q_t)$ are known.

We can write out the complete log likelihood using the joint probability:

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

introducing π, A parameters into this equation, we get:

$$= \prod_{i=1}^M \pi_i^{q_0^i} \prod_{t=0}^{T-1} \prod_{i,j=1}^M [a_{i,j}]^{q_t^i, q_{t+1}^j} \prod_{t=0}^T p(y_t|q_t)$$

We can do use brute force to sum over the q_t 's to calculate the marginal conditional probabilities, or we can be more clever.

1.2 α, β recursion, Forward-Backward algorithm

Let's look at the filtering/smoothin problems. Let y without a subscript denote all the data y_0, \dots, y_T .

Break the problem into pieces (bayer rule, conditional independencies):

$$\begin{aligned} p(q_t|y) &= \frac{p(y|q_t)p(q_t)}{p(y)} \\ &= \frac{p(y_0, \dots, y_t|q_t)p(y_{t+1}, \dots, y_T|q_t)p(q_t)}{p(y)} \\ &= \frac{p(y_0, \dots, y_t, q_t)p(y_{t+1}, \dots, y_T|q_t)}{p(y)} \end{aligned}$$

and make the definition:

$$= \frac{\alpha(q_t)\beta(q_t)}{p(y)}$$

1.3 α step

Now: how do we compute $\alpha(q_t)$? Refer to conditional independencies structure, we'd like to obtain a recursion between $\alpha(q_t)$ and $\alpha(q_{t+1})$, or $\alpha(q_{t+1}) = f(\alpha(q_t))$. Idea is to condition on a state then use the conditional independencies to decompose the probability. We want to break the chain into a chunk with all the variables up to time t (this will yield something involving $\alpha(q_t)$) and a chunk with all the variables after time t . We can do this by conditioning on q_t (Show this in diagram.)

$$\begin{aligned} \alpha(q_{t+1}) &= p(y_0, \dots, y_{t+1}, q_{t+1}) \\ &= p(y_0, \dots, y_{t+1}|q_{t+1})p(q_{t+1}) \\ &= p(y_0, \dots, y_t|q_{t+1})p(y_{t+1}|q_{t+1})p(q_{t+1}) \end{aligned}$$

$$\begin{aligned}
&= p(y_0, \dots, y_t, q_{t+1})p(y_{t+1}|q_{t+1}) \\
&= \sum_{q_t} p(y_0, \dots, y_t, q_{t+1}, q_t)p(y_{t+1}|q_{t+1}) \\
&= \sum_{q_t} p(y_0, \dots, y_t, q_{t+1}|q_t)p(q_t)p(y_{t+1}|q_{t+1}) \\
&= \sum_{q_t} p(y_0, \dots, y_t|q_t)p(q_{t+1}|q_t)p(q_t)p(y_{t+1}|q_{t+1}) \\
&= \sum_{q_t} p(y_0, \dots, y_t, q_t)p(q_{t+1}|q_t)p(y_{t+1}|q_{t+1}) \\
&= \sum_{q_t} \alpha(q_t)p(q_{t+1}|q_t)p(y_{t+1}|q_{t+1})
\end{aligned}$$

The complexity of each step of the alpha recursion is $O(M^2)$: each of the M values for q_{t+1} , it takes M multiplications to compute the inner product of $\alpha(q_t)$ and the appropriate column of the A matrix. Thus the total time for the length of the chain of length T is $O(M^2T)$. Implementation-wise this is just $(A^T \alpha_t) .* p(y_{t+1}|q_{t+1})$ where $.*$ represents component-wise multiplication of 2 vectors.

To initialize the recursion:

$$\begin{aligned}
\alpha(q_0) &= p(y_0, q_0) \\
&= p(y_0|q_0)p(q_0) \\
&= p(y_0|q_0)\pi_{q_0}
\end{aligned}$$

1.4 β step

To determine the recursion rules for the β recursion, i.e., $\beta(q_t) = f(\beta(q_{t+1}))$:

$$\begin{aligned}
\beta(q_t) &= p(y_{t+1}, \dots, y_T|q_t) \\
&= \sum_{q_{t+1}} p(y_{t+1}, \dots, y_T, q_{t+1}|q_t) \\
&= \sum_{q_{t+1}} p(y_{t+1}, \dots, y_T|q_{t+1}, q_t)p(q_{t+1}|q_t) \\
&= \sum_{q_{t+1}} p(y_{t+2}, \dots, y_T|q_{t+1})p(y_{t+1}|q_{t+1})p(q_{t+1}|q_t) \\
&= \sum_{q_{t+1}} \beta(q_{t+1})p(y_{t+1}|q_{t+1})p(q_{t+1}|q_t)
\end{aligned}$$

As for initialization, $p(y_{T+1}|q_T)$ is a meaningless quantity since there is no y_{T+1} . However, if we set $\beta(q_T)$ to a vector of ones, then we see $\beta(q_{T-1})$ is the correct quantity.

Now think about time T (omit if time is limited):

$$p(y) = \sum_i \alpha(q_T^i)\beta(q_T^i)$$

$$\begin{aligned}
&= \sum_i \alpha(q_T^i) \\
&= \sum_i p(y_0, \dots, y_T, q_T^i) = p(y)
\end{aligned}$$

This gives us all the components we needed to calculate

$$p(q_t|y) = \frac{p(y|q_t)p(q_t)}{p(y)}$$

1.5 Sum-product and the α, β recursion

Note that if we write things in terms of potentials, the α, β recursion is exactly the same as sum-product.

1.6 Alternative inference algorithm

Instead of computing $\alpha(q_t)$ and then multiplying to solve the filtering problem, lets directly compute:

$$\gamma(q_t) = \frac{\alpha(q_t)\beta(q_t)}{p(y)} = p(q_t|y_0, \dots, y_T)$$

in order to do this, we can rely on the following conditional independence assumption:

$$p(q_t|q_{t+1}, y_0, \dots, y_T) = p(q_t|q_{t+1}, y_0, \dots, y_t)$$

so:

$$\begin{aligned}
\gamma(q_t) &= p(q_t|y_0, \dots, y_T) \\
&= \sum_{q_{t+1}} p(q_t, q_{t+1}|y_0, \dots, y_T) \\
&= \sum_{q_{t+1}} p(q_t|q_{t+1}, y_0, \dots, y_t) p(q_{t+1}|y_0, \dots, y_T) \\
&= \sum_{q_{t+1}} \frac{p(q_t, q_{t+1}, y_0, \dots, y_t)}{\sum_{q_t} p(q_t, q_{t+1}, y_0, \dots, y_t)} p(q_{t+1}|y_0, \dots, y_T) \\
&= \sum_{q_{t+1}} \frac{p(q_t, y_0, \dots, y_t) p(q_{t+1}|q_t)}{\sum_{q_t} p(q_t, y_0, \dots, y_t) p(q_{t+1}|q_t)} p(q_{t+1}|y_0, \dots, y_T) \\
&= \sum_{q_{t+1}} \frac{\alpha(q_t) p(q_{t+1}|q_t)}{\sum_{q_t} \alpha(q_t) p(q_{t+1}|q_t)} p(q_{t+1}|y_0, \dots, y_T) \\
&= \sum_{q_{t+1}} \frac{\alpha(q_t) p(q_{t+1}|q_t)}{\sum_{q_t} \alpha(q_t) p(q_{t+1}|q_t)} \gamma(q_{t+1})
\end{aligned}$$

So we can now use the forward (α) recursion to compute the gamma variables (so we have an alpha-gamma algorithm). Gamma recursion is initialized with $\gamma(q_T) = \alpha(q_T)$.

1.7 ξ Variables

Building up to perform EM, we will need to compute one more thing:

$$p(q_t, q_{t+1}|y) = \xi(q_t, q_{t+1})$$

Exercise for the reader to derive a recursion for this from first principles.

Instead, here we will use our existing recursions to write out $\xi(q_t, q_{t+1})$.

$$\begin{aligned} \xi(q_t, q_{t+1}) &= p(q_t, q_{t+1}|y) \\ &= \frac{p(y|q_t, q_{t+1})p(q_{t+1}|q_t)p(q_t)}{p(y)} \\ &= \frac{p(y_0, \dots, y_t|q_t)p(y_{t+1}|q_{t+1})p(y_{t+2}, \dots, y_T|q_{t+1})p(q_{t+1}|q_t)p(q_t)}{p(y)} \\ &= \frac{\alpha(q_t)p(y_{t+1}|q_{t+1})\beta(q_{t+1})p(q_{t+1}|q_t)}{p(y)} \end{aligned}$$

Recall the homework exercise for calculating edge marginals in a tree. We are doing exactly the same thing.

2 Numerical issues

You will often run into numerical problems when daling with HMMs, especially when you are doing parameter estimation. There are $M(M - 1)$ parameters in the transition matrix, M parameters for the starting distribution, and MK parameters for the emission probabilities assuming the observations Y_t are multinomial with K possible values. The time scale you consider must be at least some reasonable multiple of the number of parameters you wish to estimate, so your time scale will be long.

Note: “Normalize alphas”:

$$\alpha(q_t) = p(y_0, \dots, y_t, q_t)$$

get to be really really small as $t \rightarrow \infty$.

Instead, normalize by $p(y_0, \dots, y_t)$ to get $\alpha'(q_t) = p(q_t|y_0, \dots, y_t)$.

Why this is good:

- Can be used in recursion for γ variables without changing a thing.
- Also used in the recursion for β (although without a real statistical interpretation): rescale beta variables by the normalization factor used in the alpha recursions at each timestep t (ie. $\beta'(q_t) = \beta(q_t)p(y_0, \dots, y_t)$), and you will get the right answer since $\alpha'(q_t)\beta'(q_t) = \alpha(q_t)\beta(q_t)$. Actually, you probably want to calculate $\beta'(q_t)/p(y) = \beta(q_t) * p(y_0, \dots, y_t)/p(y)$

Let's calculate what the renormalization factor at each step is

$$\begin{aligned}\alpha'(q_{t+1}) &\propto \sum_{q_t} p(q_t|y_0, \dots, y_t)p(q_{t+1}|q_t)p(y_{t+1}|q_{t+1}) \\ &= \sum_{q_t} p(q_t, q_{t+1}, y_{t+1}|y_0, \dots, y_t) \\ &= p(q_{t+1}, y_{t+1}|y_0, \dots, y_t)\end{aligned}$$

So in order to get $\alpha'(q_{t+1}) = p(q_{t+1}|y_0, \dots, y_{t+1})$, the appropriate renormalization const is $z_{t+1} = p(y_{t+1}|y_0, \dots, y_t)$.

Thus, if you keep track of the renormalization constants used, you get

$$\log p(y) = \sum_{t=0}^T \log z_t$$

3 Parameter Estimation

In class, we discussed parameter estimation, focusing on the M-step of the EM algorithm.

Expected Complete Log-Likelihood:

$$\log p(q, y) = \sum_{i=1}^M \langle q_0^i \rangle \log \pi_i + \sum_{t=0}^{T-1} \sum_{i,j=1}^M \langle q_t^i q_{t+1}^j \rangle \log a_{i,j} + \sum_{t=0}^T \sum_{i,j=1}^M \log p(y_i|q_i)$$

Let $m_{ij} = \sum_{t=0}^{T-1} q_t^i q_{t+1}^j$, then:

$$\begin{aligned}\hat{a}_{ij} &= \frac{m_{ij}}{\sum_{k=1}^M m_{ik}} \\ \hat{\pi}_i &= q_0^i\end{aligned}$$

But how do we estimate $\langle q_t^i q_{t+1}^j \rangle$?

$$\begin{aligned}E(m_{ij}|y, \theta^{(p)}) &= \sum_{t=0}^{T-1} E(q_t^i q_{t+1}^j | y, \theta^{(p)}) \\ &= \sum_{t=0}^{T-1} p(q_t^i q_{t+1}^j | y, \theta^{(p)}) \\ &= \sum_{t=0}^{T-1} \xi_{t,t+1}^{ij}\end{aligned}$$

But we showed how to compute these earlier based on our α, β variables.

So writing out \hat{a}_{ij} :

$$\begin{aligned}\hat{a}_{ij}^{(p+1)} &= \frac{m_{ij}}{\sum_{k=1}^M m_{ik}} \\ &= \frac{\sum_{t=0}^{T-1} \xi_{t,t+1}^{ij}}{\sum_{k=1}^M \sum_{t=0}^{T-1} \xi_{t,t+1}^{ik}} \\ &= \frac{\sum_{t=0}^{T-1} \xi_{t,t+1}^{ij}}{\sum_{t=0}^{T-1} \gamma_t^i}\end{aligned}$$

and similarly $\hat{\pi}_i^{(p+1)} = \gamma_0^i$.

So they are easy.

4 Viterbi

Mentioned in class that we can compute the maximum likelihood set of states given estimates of the parameters. this is called max-product, and written (alpha's converted):

$$\delta(q_{t+1}) = \max_{q_t} \delta(q_t) p(q_{t+1}|q_t) p(y_{t+1}|q_{t+1})$$

vector for each timestep, matrix for all timesteps (M x T).

At this point, you have $p(q_T|q_0, \dots, q_T)$.

Starting at time T , trace back along the graph (selecting the last state q_T to be $\operatorname{argmax}_{q_T} \delta(q_T)$):

$$\nu(q_t) = \operatorname{argmax}_{q_{t-1}} \delta(q_{t-1}) p(q_t|q_{t-1})$$

Because of markov properties, this is a Dynamic Programming problem: given the state we are in now, which we assume is the maximum likelihood state given the forward and backward set of observations, we can optimally choose the best previous state given δ and A . Quick draw this (if there's time), showing how its a simple trace back.