

(1)

LECTURE 10MATH 285Notes for class on 05/02/05Hidden Markov Models (HMM):

Last time, the forward algorithm was discussed for solving the problem of finding $P^\lambda(\mathcal{O})$

Now, the solution to Problem 2 (DECODING) will be discussed. For this λ will be fixed. Outputs of HMM are observed over a fixed time period. The path that the Markov Chain (MC) follows with the highest probability will be required.

With λ fixed, fix $\mathcal{O} = (o_0^*, o_1^*, \dots, o_{T-1}^*)$

It is required to determine $V(\mathcal{O})$ (the Viterbi score of the output sequence)

$$\text{where } V(\mathcal{O}) = \max_{\text{all } q} P(\mathcal{O}, Q=q) \quad q = (q_0, q_1, \dots, q_i, \dots, q_{T-1})$$

where there are N possible values of q_i

$$\text{Let } q_j^* = \operatorname{argmax}_q P(\mathcal{O}, Q=q)$$

$$\text{Now, } \max_{\text{all } q} P(\mathcal{O}, Q=q)$$

$$= \max_{\text{all } q} \left(\pi_{q_0} b_{q_0, o_0^*} a_{q_0, q_1} b_{q_1, o_1^*} a_{q_1, q_2} b_{q_2, o_2^*} \dots a_{q_{T-2}, q_{T-1}} b_{q_{T-1}, o_{T-1}^*} \right)$$

This is a function of q ($f(q)$) on T dimensional space $\mathbb{R}^T$. Direct maximization is difficult, so we use Viterbi algorithm.

Viterbi algorithm:

Fix $i \in \{1, 2, \dots, N\}$, $t \in \{0, 1, 2, \dots, (T-1)\}$

(2)

Define $\delta_t(i) = \max_{q|_{t-1}} P(Q_t = i, Q|_{t-1} = q|_{t-1}, O|_t = o^*_t)$

where $Q|_{t-1}$ is equivalent to $(Q_0, Q_1, \dots, Q_{t-1})$

(i) At $t=0$, let $\delta_0(i) = P(Q_0 = i, O_0 = o^*_0) = \pi_i b_{i o^*_0}$

Proceeding inductively,

$$\begin{aligned} \text{(ii)} \quad \delta_{t+1}(i) &= \max_{q|_t} P(Q_{t+1} = i, Q|_t = q|_t, O|_{t+1} = o^*_{t+1}) \\ &= \max_{j=1}^N \max_{q|_{t-1}} P(Q_{t+1} = i, O_{t+1} = o^*_{t+1}, Q_t = j, Q|_{t-1} = q|_{t-1}, O|_t = o^*_t) \\ &= \max_{j=1}^N \max_{q|_{t-1}} P(Q_{t+1} = i, O_{t+1} = o^*_{t+1} \mid Q_t = j, Q|_{t-1} = q|_{t-1}, O|_t = o^*_t) \cdot \\ &\quad P(Q_t = j, Q|_{t-1} = q|_{t-1}, O|_t = o^*_t) \quad \text{--- (1)} \end{aligned}$$

But $P(Q_{t+1} = i, O_{t+1} = o^*_{t+1} \mid Q_t = j, Q|_{t-1} = q|_{t-1}, O|_t = o^*_t)$

$= P(Q_{t+1} = i, O_{t+1} = o^*_{t+1} \mid Q_t = j)$ because of the Markov property

Therefore the expression (1) reduces to

$$\delta_{t+1}(i) = \max_{j=1}^N a_{ji} b_{i o^*_{t+1}} \delta_t(j) \quad \text{--- (2)}$$

$$\text{Also } \psi_{t+1}(i) = \operatorname{argmax}_{j=1, 2, \dots, N} a_{ji} b_{i o^*_{t+1}} \delta_t(j) \quad \text{--- (3)}$$

It is required to keep track of both (2) and (3) as the iterations progress.

The Viterbi score is $V(O) = \max_{i=1}^N \delta_{T-1}(i)$

(3)

To find the most probable path for the HMM use backtracking.

$$q_{T-1}^* = \arg \max_{i=1}^N f_{T-1}(i)$$

$$q_{T-2}^* = \Psi_{T-1}(q_{T-1}^*) = \arg \max_{j=1}^N P(Q_{T-1} = q_{T-1}^*, Q_{T-2} = q_{T-2}^* | Q_{T-2} = i) \cdot \delta_{T-2}(j)$$

Proceeding iteratively, we get most probable sequence at time 0.

PROFILE HMMs:

For a common ancestor, descendants have differences in DNA.

By looking at them, we can ^{try to} get DNA sequence for common ancestor.

Sequences generated by such a HMM are regarded as being for descendants of a common ancestor generated by mutations, insertions and deletions.

DNA $\longrightarrow$ mRNA $\longrightarrow$ sequence of amino acids which are codes for proteins

(A, C, G, T)

(A, C, G, U)

sequence of amino acids which are codes for proteins

three element

The mapping has redundancy since more than one ^{three element} sequence of mRNA can map to one amino acid.

Example:

Consider 2 individuals with the following sequences of amino acids:

1. ADKTN

2. AEEKSN

Note that the sequences are not of the same length i.e. outputs for given segment of time have different lengths.

(4)

A potential common ancestor can be ADKTSN

This can ~~happen~~^{evolve} in the following ways:

1st sequence: A • D K T - N

2nd sequence: A c E K - S N

Insertion is shown with lower-case letters, • put for alignment, - signals a deletion.

These sequences correspond to the path that the HMM followed.

Profile HMMs have the following kinds of states:

- → match states, correspond to real states in original ancestor
- ◇ → insert states, states which generate extra outputs not in ancestor
- → delete states (silent states). When in these states do not output anything. This allows us to omit states from ancestor.

Notes on given drawing of profile HMM:

- i) Time in MC corresponds to position in ~~chain~~ sequence
- ii) Given an output sequence, we want to construct path through MC which is most likely to generate the output
- iii) The BEGIN and END states are stop codons
- iv) Delete states have no outputs

Can Viterbi algorithm work on variable length outputs?
It will be shown that this can be done.