

Hidden Markov Models



Lecture 6, Thursday April 17, 2003

Review of Last Lecture



Lecture 6, Thursday April 17, 2003

1. When the true underlying states are known

Given $x = x_1 \dots x_N$
for which the true $\pi = \pi_1 \dots \pi_N$ is known,

Define:

A_{kl} = # times $k \rightarrow l$ transition occurs in π
 $E_k(b)$ = # times state k in π emits b in x

We can show that the maximum likelihood parameters θ are:

$$a_{kl} = \frac{A_{kl}}{\sum_l A_{kl}} \quad e_k(b) = \frac{E_k(b)}{\sum_c E_k(c)}$$

Lecture 7, Tuesday April 22, 2003

2. When not – The Baum-Welch Algorithm

Initialization:

Pick the best-guess for model parameters
(or arbitrary)

Iteration:

Forward

Backward

Calculate A_{kl} , $E_k(b)$

Calculate new model parameters a_{kl} , $e_k(b)$

Calculate new log-likelihood $P(x | \theta)$

GUARANTEED TO BE HIGHER BY EXPECTATION-MAXIMIZATION

Until $P(x | \theta)$ does not change much

Lecture 7, Tuesday April 22, 2003

Alternative: Viterbi Training

Initialization: Same

Iteration:

Perform Viterbi, to find π^*
Calculate A_{kl} , $E_k(b)$ according to π^* + pseudocounts
Calculate the new parameters a_{kl} , $e_k(b)$
Until convergence

Notes:

- Convergence is guaranteed – Why?
- Does not maximize $P(x | \theta)$
- In general, worse performance than Baum-Welch
- Convenient – when interested in Viterbi parsing, no need to implement additional procedures (Forward, Backward)!!

Lecture 7, Tuesday April 22, 2003

Variants of HMMs



Lecture 6, Thursday April 17, 2003

Higher-order HMMs

The Genetic Code

3 nucleotides make 1 amino acid

Statistical dependencies in triplets

Question:

Recognize protein-coding segments with a HMM

	U	C	A	G
U	UUU phe UUC UUA leu UUG	UCU UCC UCA ser UCG	UAU tyr UAC UAA Stop UAG Stop	UGU cys UGC UGA Stop UGG Stop
C	CUU CUC CUA leu CUG	CCU CCC CCA pro CCG	CAU his CAC CAA gln CAG	CGU CGC CGA arg CGG
A	AUU AUC AUA AUG met	ACU ACC ACA thr ACG	AAU asn AAC AAA lys AAG	AGU ser AGC AGA arg AGG
G	GUU GUC GUA val GUG	GCU GCC GCA ala GCG	GAU asp GAC GAA glut GAG	GGU GGC GGA gly GGG

Lecture 7, Tuesday April 22, 2003

One way to model protein-coding regions

$$P(x_{i+1} x_{i+2} | x_{i-1} x_i x_{i+1})$$

Every state of the HMM emits 3 nucleotides

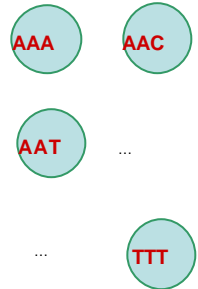
Transition probabilities:

Probability of one triplet, given previous triplet $P(\pi_i | \pi_{i-1})$

Emission probabilities:

$$P(x_{i+1} x_{i+2} | \pi_i) = 1/9$$

$$P(x_{i-1} x_i x_{i+1} | \pi_{i-1}) = 1/9$$



Lecture 7, Tuesday April 22, 2003

A more elegant way

Every state of the HMM emits 1 nucleotide

Transition probabilities:

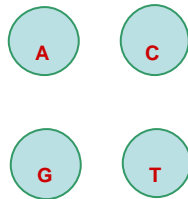
Probability of one triplet, given previous 3 triplets

$$P(\pi_i | \pi_{i-1}, \pi_{i-2}, \pi_{i-3})$$

Emission probabilities:

$$P(x_i | \pi_i)$$

Algorithms extend with small modifications



Lecture 7, Tuesday April 22, 2003

Modeling the Duration of States

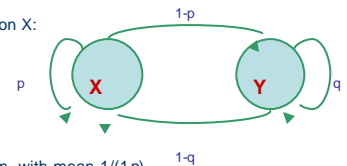
Length distribution of region X:

$$E[l_X] = 1/(1-p)$$

- Exponential distribution, with mean $1/(1-p)$

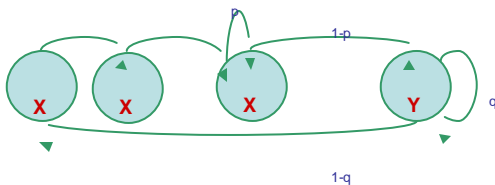
This is a significant disadvantage of HMMs

Several solutions exist for modeling different length distributions



Lecture 7, Tuesday April 22, 2003

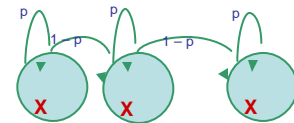
Solution 1: Chain several states



Disadvantage: Still very inflexible
 $l_X = C + \text{exponential with mean } 1/(1-p)$

Lecture 7, Tuesday April 22, 2003

Solution 2: Negative binomial distribution



$$P(l_X = n) = \binom{n-1}{n-1} p^{n-1} (1-p)^n$$

Lecture 7, Tuesday April 22, 2003

Solution 3: Duration modeling

Upon entering a state:

1. Choose duration d , according to probability distribution
2. Generate d letters according to emission probs
3. Take a transition to next state according to transition probs



Disadvantage: Increase in complexity:

Time: $O(D^2)$
 Space: $O(D)$
 Where D = maximum duration of state

Lecture 7, Tuesday April 22, 2003

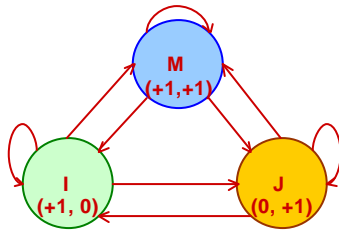
Connection Between Alignment and HMMs



Lecture 6, Thursday April 17, 2003

A state model for alignment

Alignments correspond 1-to-1 with sequences of states M, I, J

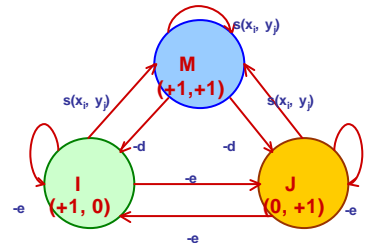


-AGGCTATCACCTGACCTCCAGGCCGA--TGCCC--
 TAG-CTATCAC--GACCGC-GGTCGATTGCCCCGACC
 IMMJMMMMMMJJMMMMMMJJMMMMMMIIMMMMMIIII

Lecture 7, Tuesday April 22, 2003

Let's score the transitions

Alignments correspond 1-to-1 with sequences of states M, I, J



-AGGCTATCACCTGACCTCCAGGCCGA--TGCCC--
 TAG-CTATCAC--GACCGC-GGTCGATTGCCCCGACC
 IMMJMMMMMMJJMMMMMMJJMMMMMMIIMMMMMIIII

Lecture 7, Tuesday April 22, 2003

How do we find optimal alignment according to this model?

Dynamic Programming:

$M(i, j)$: Optimal alignment of $x_1 \dots x_i$ to $y_1 \dots y_j$ ending in M

$I(i, j)$: Optimal alignment of $x_1 \dots x_i$ to $y_1 \dots y_j$ ending in I

$J(i, j)$: Optimal alignment of $x_1 \dots x_i$ to $y_1 \dots y_j$ ending in J

The score is additive, therefore we can apply DP recurrence formulas

Lecture 7, Tuesday April 22, 2003

Needleman Wunsch with affine gaps - state version

Initialization:

$M(0,0) = 0$; $M(i,0) = M(0,j) = -\infty$, for $i, j > 0$
 $I(i,0) = d + i \times e$; $J(0,j) = d + j \times e$

Iteration:

$M(i, j) = s(x_i, y_j) + \max \begin{cases} M(i-1, j-1) \\ I(i-1, j-1) \\ J(i-1, j-1) \end{cases}$
 $I(i, j) = \max \begin{cases} e + I(i-1, j) \\ e + J(i, j-1) \\ d + M(i-1, j-1) \end{cases}$
 $J(i, j) = \max \begin{cases} e + I(i-1, j) \\ e + J(i, j-1) \\ d + M(i-1, j-1) \end{cases}$

Termination:

Optimal alignment given by $\max \{ M(m, n), I(m, n), J(m, n) \}$

Lecture 7, Tuesday April 22, 2003

Probabilistic interpretation of an alignment

An alignment is a hypothesis that the two sequences are related by evolution

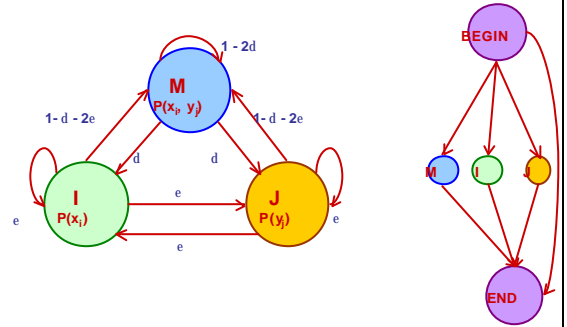
Goal:

Produce the **most likely** alignment

Assert the likelihood that the sequences are indeed related

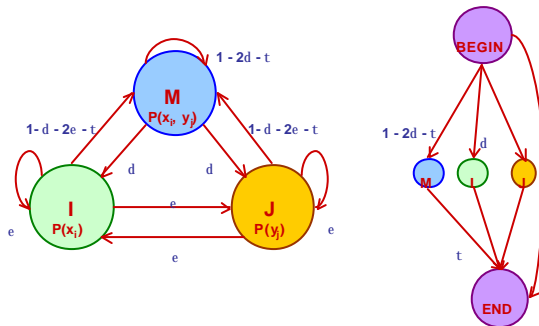
Lecture 7, Tuesday April 22, 2003

A Pair HMM for alignments



Lecture 7, Tuesday April 22, 2003

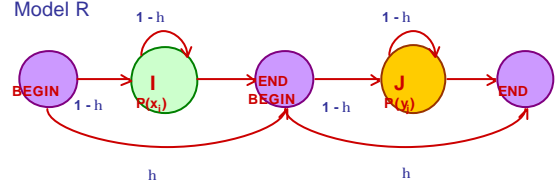
A Pair HMM for alignments



Lecture 7, Tuesday April 22, 2003

A Pair HMM for not aligned sequences

Model R



$$P(x, y | R) = \eta(1 - \eta)^m P(x) \dots P(x^m) \eta(1 - \eta)^n P(y) \dots P(y^n) \\ = \eta^2(1 - \eta)^{m+n} \prod_i P(x_i) \prod_j P(y_j)$$

Lecture 7, Tuesday April 22, 2003

To compare ALIGNMENT vs. RANDOM hypothesis

Every pair of letters contributes:

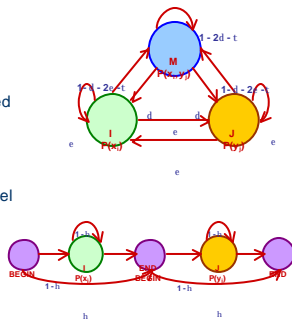
$(1 - 2\delta - \tau) P(x, y)$ when matched

$\epsilon P(x_i) P(y_j)$ when gapped

$(1 - \eta)^2 P(x_i) P(y_j)$ in random model

Focus on comparison of

$P(x, y)$ vs. $P(x_i) P(y_j)$



Lecture 7, Tuesday April 22, 2003

To compare ALIGNMENT vs. RANDOM hypothesis

Idea:

We will divide alignment score by the random score, and take logarithms

Let

$$s(x_i, y_j) = \log \frac{P(x_i, y_j)}{P(x_i) P(y_j)} + \log \frac{(1 - 2\delta - \tau)}{(1 - \eta)^2}$$

$$d = -\log \frac{\delta(1 - \epsilon - \tau) P(x_i)}{(1 - \eta)(1 - 2\delta - \tau) P(x_i)}$$

$$e = -\log \frac{\epsilon P(x_i)}{(1 - \eta) P(x_i)}$$

Every letter b in random model contributes $(1 - \eta) P(b)$

Lecture 7, Tuesday April 22, 2003

The meaning of alignment scores

Because δ , ϵ , are small, and η , τ are very small,

$$s(x_i, y_j) = \log \frac{P(x_i, y_j)}{P(x_i) P(y_j)} + \log \frac{(1 - 2\delta - \tau)}{(1 - \eta)^2} \cong \log \frac{P(x_i, y_j)}{P(x_i) P(y_j)}$$

$$d = -\log \frac{\delta(1 - \epsilon - \tau)}{(1 - \eta)(1 - 2\delta - \tau)} \cong -\log \delta$$

$$e = -\log \frac{\epsilon}{(1 - \eta)} \cong -\log \epsilon$$

Lecture 7, Tuesday April 22, 2003

The meaning of alignment scores

The Viterbi algorithm for Pair HMMs corresponds exactly to the Needleman-Wunsch algorithm with affine gaps

However, now we need to score alignment with parameters that add up to probability distributions

δ : 1/mean arrival time of next gap
 ϵ : 1/mean length of next gap

affine gaps decouple arrival time with length

τ : 1/mean length of conserved segments (set to ~ 0)
 η : 1/mean length of sequences of interest (set to ~ 0)

Lecture 7, Tuesday April 22, 2003

The meaning of alignment scores

Match/mismatch scores:

$$s(a, b) \cong \log \frac{P(x_i, y_j)}{P(x_i) P(y_j)}$$

Example:

Say DNA regions between human and mouse have average conservation of 50%

Then $P(A,A) = P(C,C) = P(G,G) = P(T,T) = 1/8$ (so they sum to $1/2$)
 $P(A,C) = P(A,G) = \dots = P(T,G) = 1/24$ (24 mismatches, sum to $1/2$)

Say $P(A) = P(C) = P(G) = P(T) = 1/4$

Then, $s(a, b) = \log \left[\frac{(1/8) / (1/4 * 1/4)}{(1/24) / (1/4 * 1/4)} \right] = \log 16/24 = -0.585$

Note: $0.585 / 1.585 = 37.5$

According to this model, a 37.5%-conserved sequence with no gaps would score on average $0.375 * 1 - 0.725 * 0.585 = 0$

Why?

37.5% is between the 50% conservation model, and the random 25% conservation model !

Lecture 7, Tuesday April 22, 2003

Substitution matrices

A more meaningful way to assign match/mismatch scores

For protein sequences, different substitutions have dramatically different frequencies!

BLOSUM matrices:

1. Start from BLOCKS database (curated, gap-free alignments)
2. Cluster sequences according to % identity
3. For a given L% identity, calculate A_{ab} : # of aligned pos a-b
4. Estimate

$$P(a) = (\sum_b A_{ab}) / (\sum_{cd} A_{cd}); \quad P(a, b) = A_{ab} / (\sum_{cd} A_{cd})$$

Lecture 7, Tuesday April 22, 2003

BLOSUM matrices

BLOSUM 50

BLOSUM 62

	A	C	D	E	F	G	H	I	L	K	M	N	P	Q	R	S	T	V	W	X	Y	Z
A	4	-2	0	-2	-1	-2	0	-2	-1	-1	-1	-2	-1	-1	-1	1	0	0	-5	-1	-2	-1
C	-2	6	-5	4	-2	-5	-1	-5	-3	-4	-4	-3	1	-1	0	-2	0	-1	-5	-4	-1	-3
D	0	-5	9	-3	-4	-2	-3	-3	-5	-5	-1	-1	-3	-3	-5	-1	-1	-5	-2	-1	-2	-4
E	-2	4	-3	6	-2	-5	-1	-5	-3	-4	-4	-3	1	-1	0	-2	0	-1	-5	-4	-1	-3
F	-1	-2	4	2	5	-3	-2	0	-3	-3	0	-2	0	1	2	0	0	-1	-5	-3	-1	-2
G	-2	-5	-3	-2	-5	6	-5	-1	0	-5	0	-5	-4	-3	-2	-2	-1	1	-1	-3	-5	
H	0	-1	-3	-1	-2	-5	6	-2	-4	-2	-4	-3	-2	-2	0	-2	-2	-1	-3	-2	-2	
I	-1	-5	-3	-3	0	-4	-2	6	-5	-1	-2	1	0	0	-1	-2	-2	-2	-1	-2	-3	
L	-1	-3	-1	-1	-3	0	-4	-5	4	-5	2	1	-3	-3	-2	-1	3	-5	-1	-3	-2	
K	-1	-4	-4	-4	-3	-5	-2	-1	-5	5	-2	0	1	2	0	-1	-2	-5	-1	-2	-3	
M	-1	-4	-4	-4	-3	-5	-2	-1	-5	-2	5	-2	0	1	1	0	-1	-5	-1	-2	-3	
N	-2	-3	-1	0	-2	-5	0	-3	-3	-2	-2	5	-2	-1	-1	1	0	-4	-1	-2	-5	
P	-1	-5	-1	-1	-4	-2	-2	-1	-3	-2	-2	-2	5	-1	-2	-1	-2	-4	-1	-3	-5	
Q	-1	-1	0	0	-2	-3	-1	-3	-3	-2	-1	-1	-1	5	1	1	0	-1	-2	-5	-2	
R	-1	-4	-4	-4	-3	-5	-2	-1	-5	-2	-2	-2	-1	-1	5	1	-1	-2	-5	-1	-2	
S	1	0	0	0	-2	-3	-1	-3	-3	-2	-1	-1	0	-1	-1	0	5	-3	-2	-5	-2	
T	0	-1	-1	-1	-2	-2	-1	-1	-1	0	-1	0	-1	1	1	0	-2	5	-3	-1	-2	
V	0	-5	-1	-2	-2	-1	-3	-3	-2	1	-3	-2	-2	-2	0	0	0	-3	4	-1	-1	
W	-4	-1	-2	-1	1	-2	-2	-2	-2	-1	-4	-2	-2	-2	-2	-1	-1	-1	-3	3	-2	
X	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	2	7	
Y	-2	-3	-2	-2	-2	-3	-2	-2	-2	-1	-1	-2	-1	-1	-2	-1	-2	-2	-1	2	1	
Z	-1	-2	4	2	5	-2	-2	0	-1	-3	-2	0	-1	2	0	0	-1	-2	-5	-1	-2	

(The two are scaled differently)

Lecture 7, Tuesday April 22, 2003

Lecture 7, Tuesday April 22, 2003