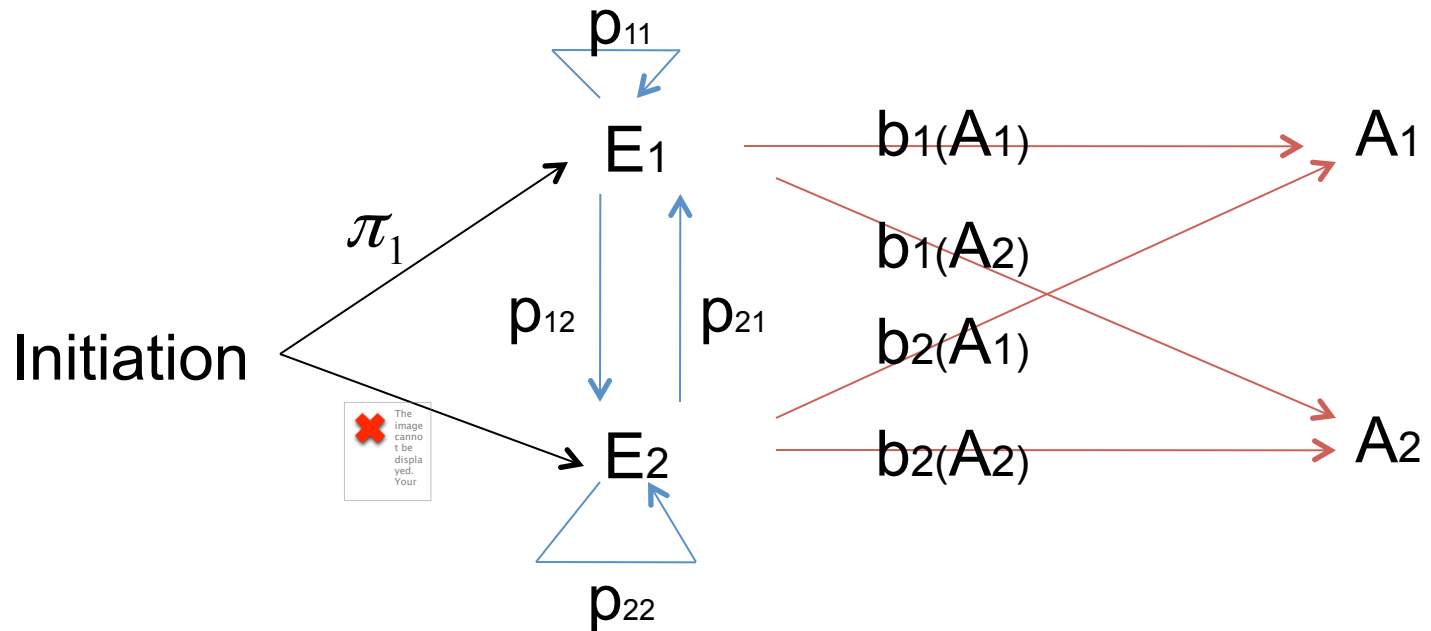


Hidden Markov Model

Part 2: Parameter Estimation

- Parameter estimation
 - Viterbi training
 - Baum-Welch algorithm
- Some examples of HMM applications

HMM



$$\lambda = \{\pi, P, B\}$$

initial distribution
transition probabilities
emission probabilities

Hidden Markov Model

Common questions:

How to efficiently calculate emissions: $P(O | \lambda)$

How to find the most likely hidden state: $\arg \max_Q P(Q | O)$

How to find the most likely parameters: $\arg \max_{\lambda} P(O | \lambda)$

Notations:

Let $Q = \{q^{(1)} q^{(2)} \dots q^{(T)}\}$ denote the hidden state sequence;

Let $O = \{O_1, O_2, \dots, O_T\}$ denote the observed emission sequence;

Let $O^{(t)} = (O_1, O_2, \dots, O_t)$ denote the emission up to time t.

Reminders

Calculating emissions probabilities:

$$\alpha(t, i) = P(O^{(t)}, q^{(t)} = E_i)$$

Emissions up to step t

chain at state i at step t

$$\alpha(1, i) = \pi_i b_i(O_1)$$

$$\alpha(t + 1, i) = \sum_{k=1}^S \alpha(t, k) p_{ki} b_i(O_{t+1})$$


$$P(O) = \sum_{i=1}^S \alpha(T, i)$$

Reminders

Calculating posterior probability of hidden state:

Emissions after step t

chain at state i at step t


$$\beta(t, i) = P(O_{t+1}, O_{t+2}, \dots, O_T \mid q_t = E_i)$$

$$\beta(T, j) = 1, \forall j$$

$$\beta(t-1, i) = \sum_{k=1}^S p_{ik} b_k(O_t) \beta(t, k)$$

$$P(q^{(t)} = Ei \mid O) = \frac{P(O, q^{(t)} = Ei)}{P(O)} = \frac{\alpha(t, i) \beta(t, i)}{\sum_{i=1}^S \alpha(T, i)}$$

Reminders

Finding the most probable path: $\arg \max_Q P(Q | O)$

$$\delta_t(i) = \max_{q_1, q_2, \dots, q_{t-1}} P(q_1, q_2, \dots, q_{t-1}, q_t = Ei, \text{ and } O^{(t)})$$

$$\textit{Initiation} : \delta_1(i) = \pi_i b_i(O_1)$$

$$\textit{Induction} : \delta_t(i) = \max_k \delta_{t-1}(k) p_{ki} b_i(O_t)$$

$$\max_Q P(Q, O) = \max_i \delta_T(i)$$

The Estimation of parameters

When the topology of an HMM is known, how do we find the initial distribution, the transition probabilities and the emission probabilities, from a set of emitted sequences?

$$\arg \max_{\lambda} P(O \mid \lambda), \quad O = \{O^{(T)}\}$$

Difficulties:

Usually the number of parameters is not small. Even the simple chain in our example has 5 parameters.

Normally the landscape of the likelihood function is bumpy. Finding the global optimum is hard.

Basic idea

There are three pieces: λ , O , and Q . O is observed.

We can iterate between λ and Q .

When λ is fixed, we can do one of two things:

(1) find the best Q given O

or (2) integrate out Q

When Q is fixed, we can estimate λ using frequencies.

The Viterbi training

- (1) Start with some initial parameters
- (2) Find the most likely paths using the Viterbi algorithm.

$$\delta_t(i) = \max_{q_1, q_2, \dots, q_{t-1}} P(q_1, q_2, \dots, q_{t-1}, q_t = Ei, \text{ and } O^{(t)})$$
$$\max_Q P(Q, O) = \max_i \delta_T(i)$$

- (3) Re-estimate the parameters.
On next page.

- (4) Iterate until the most likely paths don't change.

The Viterbi training

Estimating the parameters.

Transition probabilities:

$$\hat{p}_{ij} = \frac{\# \text{ transitions from state } i \text{ to state } j}{\# \text{ transitions from state } i}$$

Emission probabilities:

$$\hat{b}_i(o_k) = \frac{\# \text{ chain at state } i \text{ and emits } o_k}{\# \text{ chain at state } i}$$

Initial probabilities: need multiple training sequences.

$$\hat{\pi}_i = \frac{\# \text{ chains starting at state } i}{\# \text{ chains}}$$

Baum-Welch algorithm

- The Baum-Welch is a special case of the EM algorithm.
- The Viterbi training is equivalent to a simplified version of the EM algorithm, replacing the expectation with the most likely classification.
- General flow:
 - (1) Start with some initial parameters
 - (2) Given these parameters, $P(Q|O)$ is known for every Q . This is done effectively by calculating the forward and the backward variables.
 - (3) Take expectations to obtain new parameter estimates.
 - (4) Iterate (2) and (3) until convergence.

Baum-Welch algorithm

Reminder:

$$\begin{aligned} P(O, q^{(t)} = Ei) \\ &= P(O_1, O_2, \dots, O_t, q^{(t)} = Ei) P(O_{t+1}, \dots, O_T \mid O_1, O_2, \dots, O_t, q^{(t)} = Ei) \\ &= P(O_1, O_2, \dots, O_t, q^{(t)} = Ei) P(O_{t+1}, \dots, O_T \mid q^{(t)} = Ei) \\ &= \alpha(t, i) \beta(t, i) \end{aligned}$$

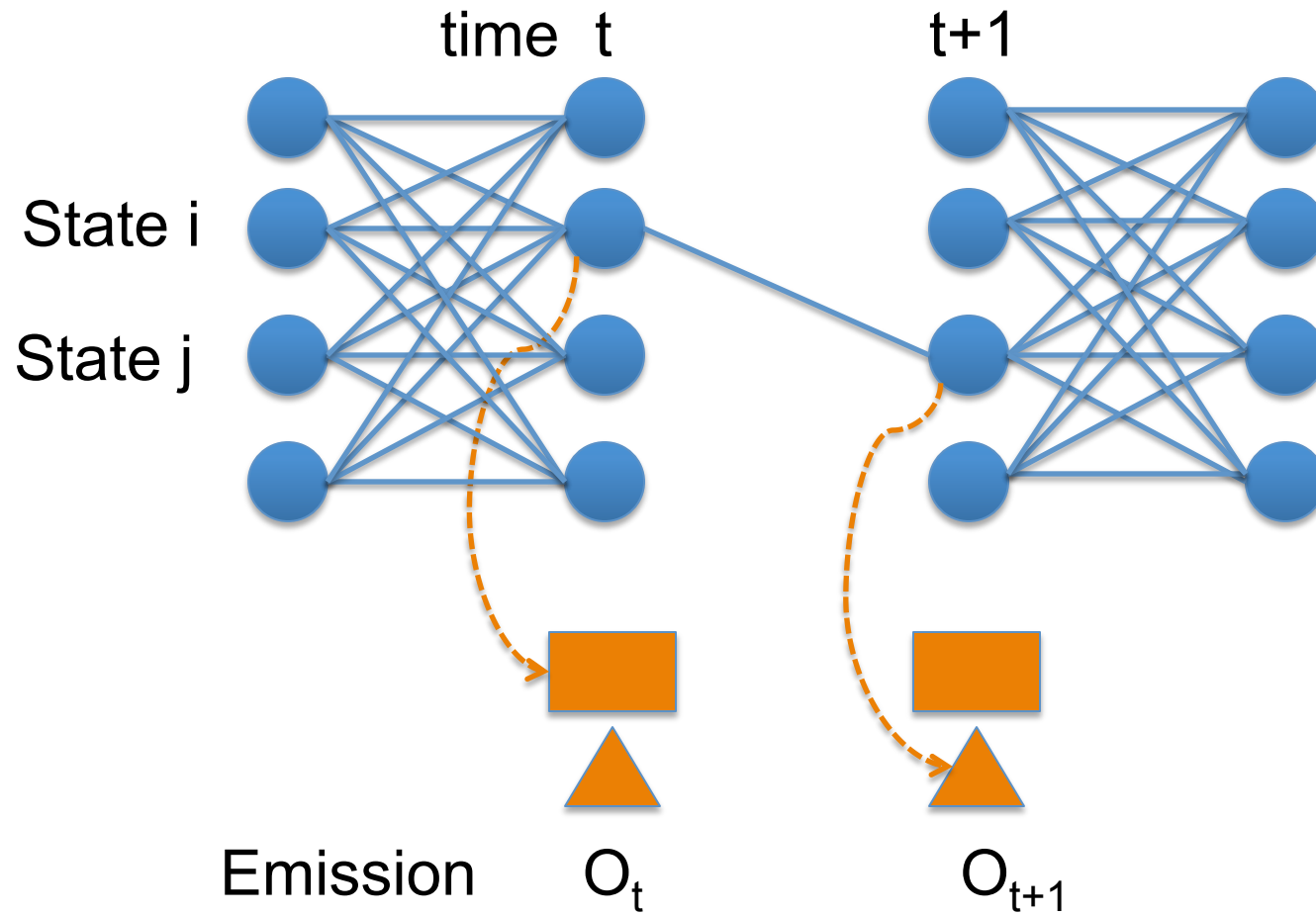
$$P(q^{(t)} = Ei \mid O) = \frac{P(O, q^{(t)} = Ei)}{P(O)} = \frac{\alpha(t, i) \beta(t, i)}{\sum_{i=1}^S \alpha(T, i)} = \eta(t, i)$$

Estimation of emission probabilities:

$$\hat{b}_i(v_k) = \frac{E(\text{chain at state } i \text{ and emit } k^{\text{th}} \text{ emission} \mid O)}{E(\text{chain at state } i \mid O)} = \frac{\sum_{t=1}^T I(O_t = v_k) \eta(t, i)}{\sum_{t=1}^T \eta(t, i)}$$

(i: state, v_k : k^{th} emission, $I()$: 1 if true, 0 otherwise)

Baum-Welch algorithm



Baum-Welch algorithm

Definition

$$\alpha(t, i) = P(O_1, O_2, \dots, O_t, q^{(t)} = S_i)$$

$$\beta(t+1, j) = P(O_{t+2}, O_{t+2}, \dots, O_T \mid q^{(t+1)} = S_j)$$

Now, let

$$\begin{aligned} \xi_t(i, j) &= \frac{\alpha(t, i) p_{i,j} b_j(O_{t+1}) \beta(t+1, j)}{\sum_{k=1}^N \alpha(T, k)} \\ &= \frac{P(O_1, O_2, \dots, O_t, q^{(t)} = S_i) p_{i,j} b_j(O_{t+1}) P(O_{t+2}, O_{t+2}, \dots, O_T \mid q^{(t+1)} = S_j)}{P(O)} \\ &= \frac{P(O_1, O_2, \dots, O_t, O_{t+1}, q^{(t)} = S_i, q^{(t+1)} = S_j) P(O_{t+2}, O_{t+2}, \dots, O_T \mid q^{(t+1)} = S_j)}{P(O)} \\ &= \frac{P(O_1, O_2, \dots, O_t, O_{t+1}, q^{(t)} = S_i, q^{(t+1)} = S_j) P(O_{t+2}, O_{t+2}, \dots, O_T \mid q^{(t+1)} = S_j, \dots)}{P(O)} \\ &= \frac{P(O, q^{(t)} = S_i, q^{(t+1)} = S_j)}{P(O)} = P(q^{(t)} = S_i, q^{(t+1)} = S_j \mid O) \end{aligned}$$

Baum-Welch algorithm

Estimation of transition probabilities:

$$\xi_t(i, j) = P(q^{(t)} = S_i, q^{(t+1)} = S_j \mid O)$$

Summing over all steps:

$$\hat{p}_{i,j} = \frac{E(N(i \rightarrow j) \mid O)}{E(N(i \rightarrow \bullet) \mid O)} = \frac{\sum_{t=1}^{T-1} \xi_t(i, j)}{\sum_{t=1}^{T-1} \sum_{k=1}^N \xi_t(i, k)}$$

Estimation of initial distribution:

$$\hat{\pi}_i = P(q^{(1)} = Ei \mid O) = \frac{\alpha(1, i) \beta(1, i)}{\sum_{i=1}^S \alpha(T, i)}$$

Baum-Welch algorithm

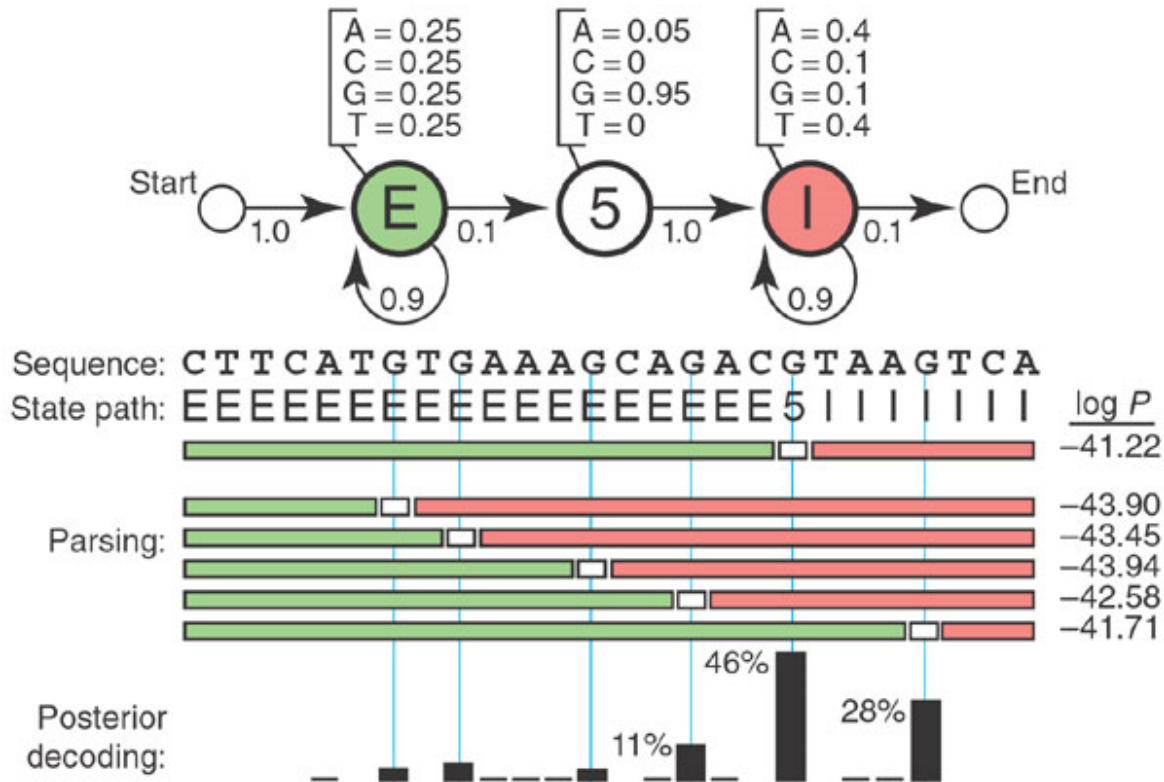
Comments:

We don't actually calculate $P(Q|O)$. Rather, the necessary information pertaining to all the thousands of $P(Q|O)$ is summarized in the forward- and backward- variables.

When the chain runs long, logarithm need to be used for numerical stability.

In applications, the emission is often continuous. A distribution need to be assumed, and the density used for emission probabilities.

Examples of HMM: Splice site recognition



E: exon
I: intron
5: 5's splice site

CpG Island

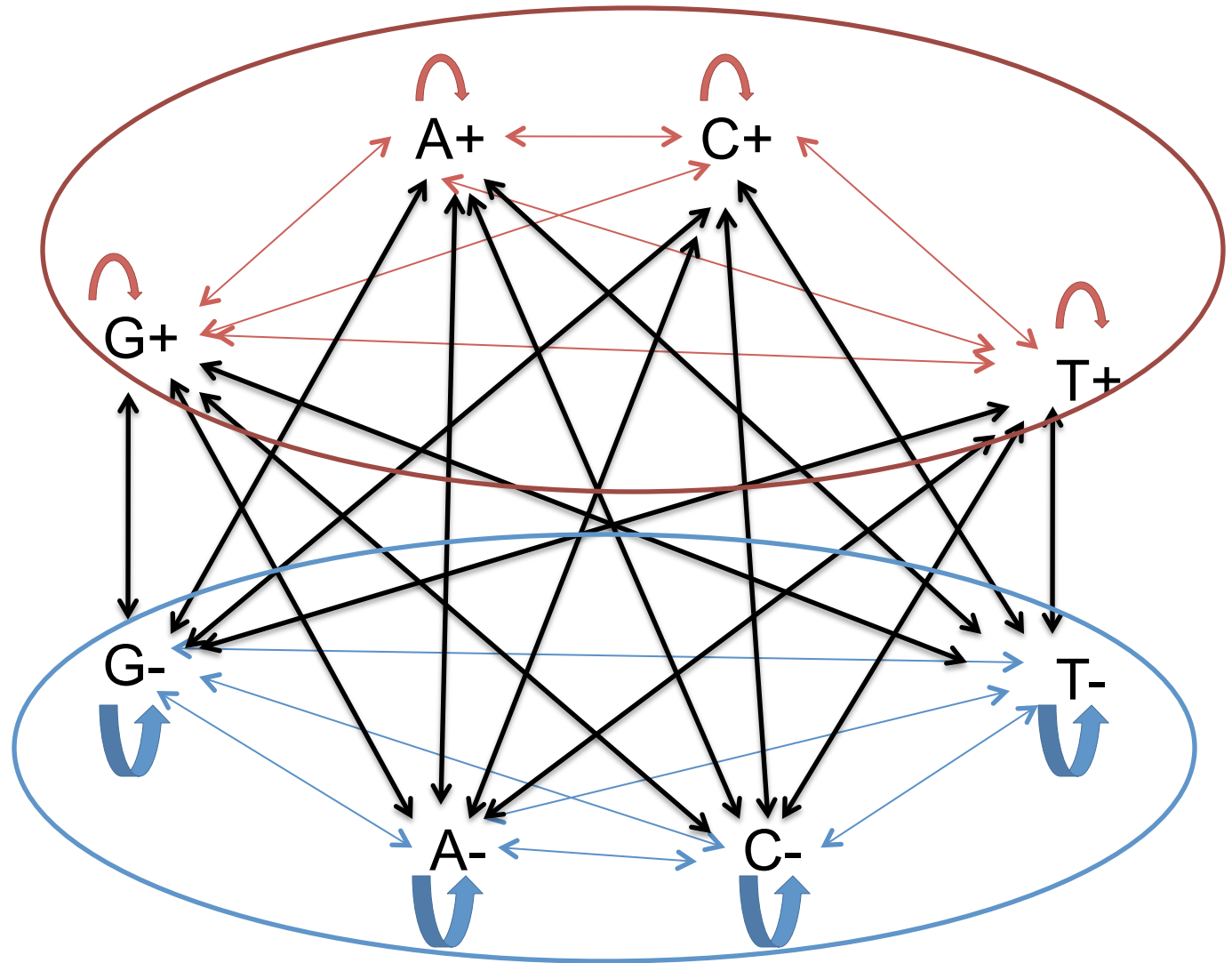
In the four character DNA book, the appearance of “CG” is rare, due to some issues related to the probability of mutations.

However, there are regions where “CG” is more frequent. And such regions are mostly associated with genes. (Remember, only 5% of the DNA sequence are genes.)

.....GTATTT**CG**AAGAAT.....

+	A	C	G	T	—	A	C	G	T
A	0.180	0.274	0.426	0.120	A	0.300	0.205	0.285	0.210
C	0.171	0.368	0.274	0.188	C	0.322	0.298	0.078	0.302
G	0.161	0.339	0.375	0.125	G	0.248	0.246	0.298	0.208
T	0.079	0.355	0.384	0.182	T	0.177	0.239	0.292	0.292

CpG Island



Within the “+” and “-” groups, the transition probabilities are close to shown in the table. A small probability is left for between “+” and “-” transitions.

CpG Island

Transition probabilities are summarized from known CpG islands.

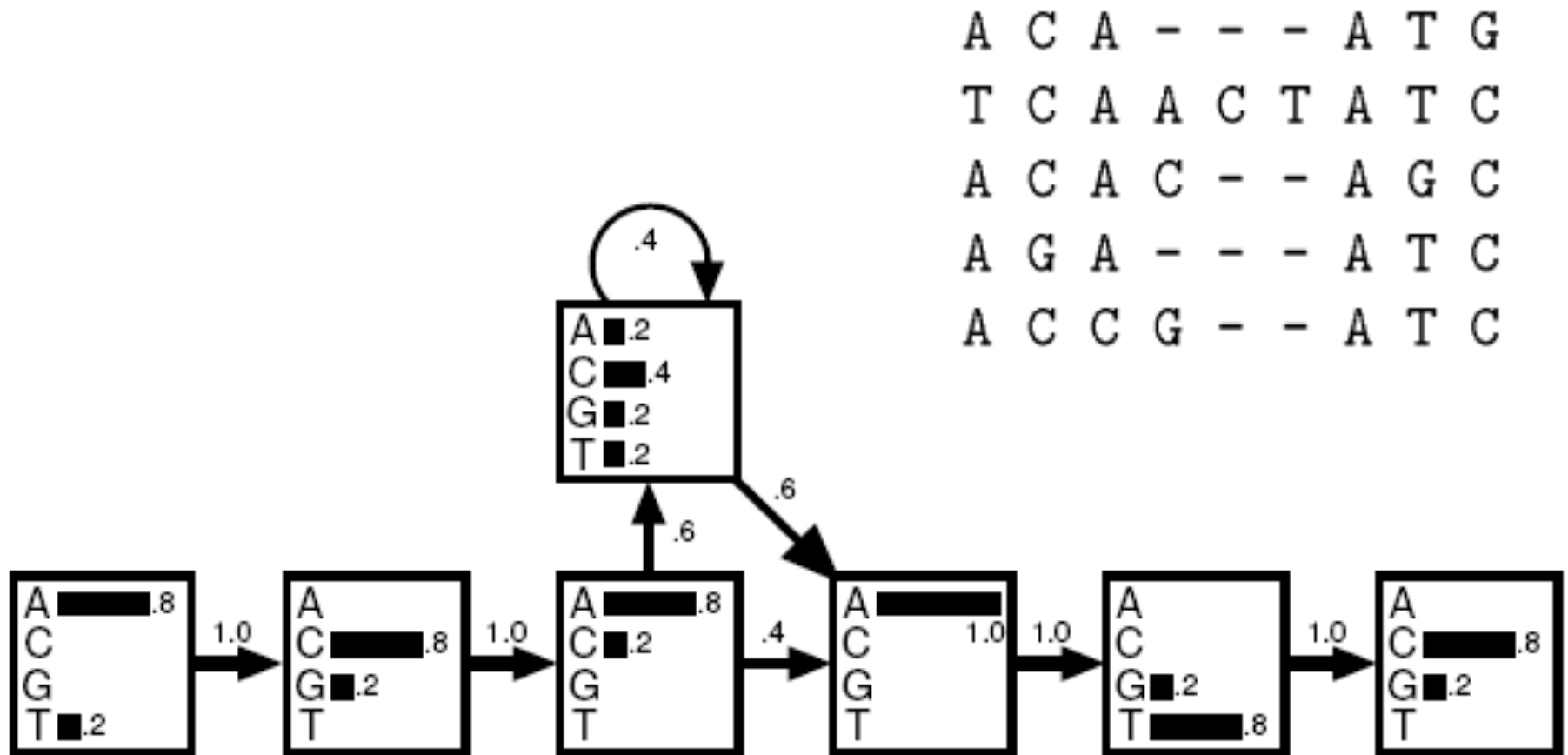
To identify CpG island from new sequences:

Using the most likely path: run the Viterbi algorithm. When the chain passes through any of the four “+” states, call it an island

Using the posterior decoding: run the forward- and backward- algorithm. Obtain the posterior probabilities of each letter being in a CpG island.

Profile HMM --- a simple case

Motif: functional short sequence that tends to be conserved.

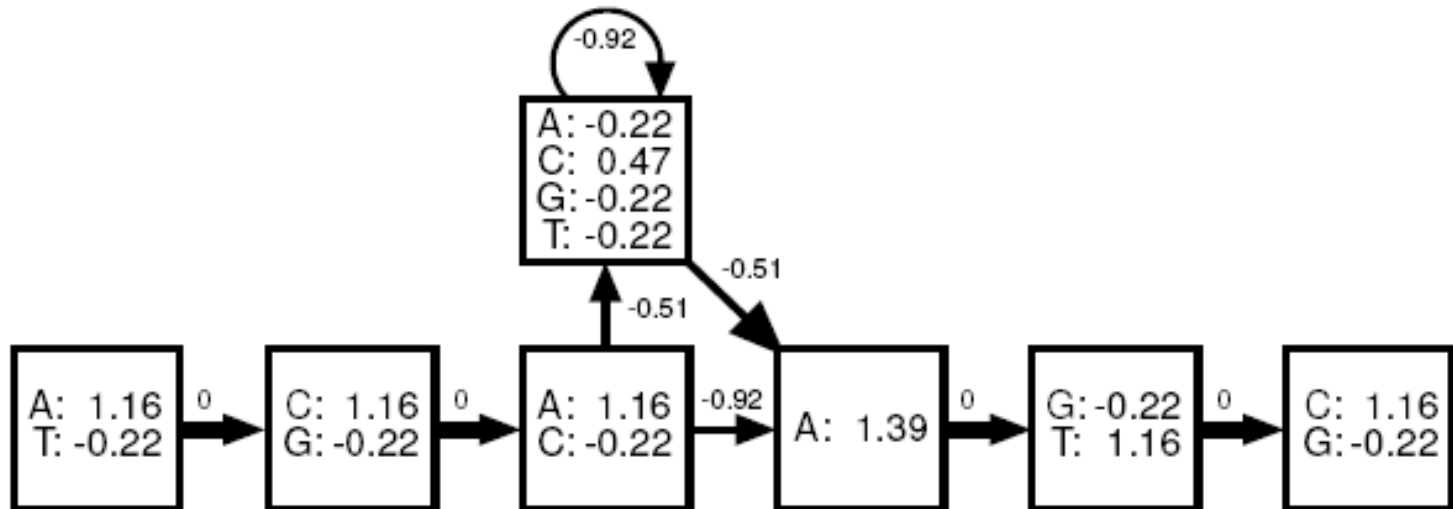


Profile HMM --- a simple case

$$\begin{aligned}P(\text{ACACATC}) &= 0.8 \times 1 \times 0.8 \times 1 \times 0.8 \times 0.6 \times \\ &\quad 0.4 \times 0.6 \times 1 \times 1 \times 0.8 \times 1 \times 0.8 \\ &\simeq 4.7 \times 10^{-2}.\end{aligned}$$

$$\text{log-odds for sequence } S = \log \frac{P(S)}{0.25^L} = \log P(S) - L \log 0.25.$$

For (1) accounting for sequence length, (2) computational stability, Log odds ratios are preferred.

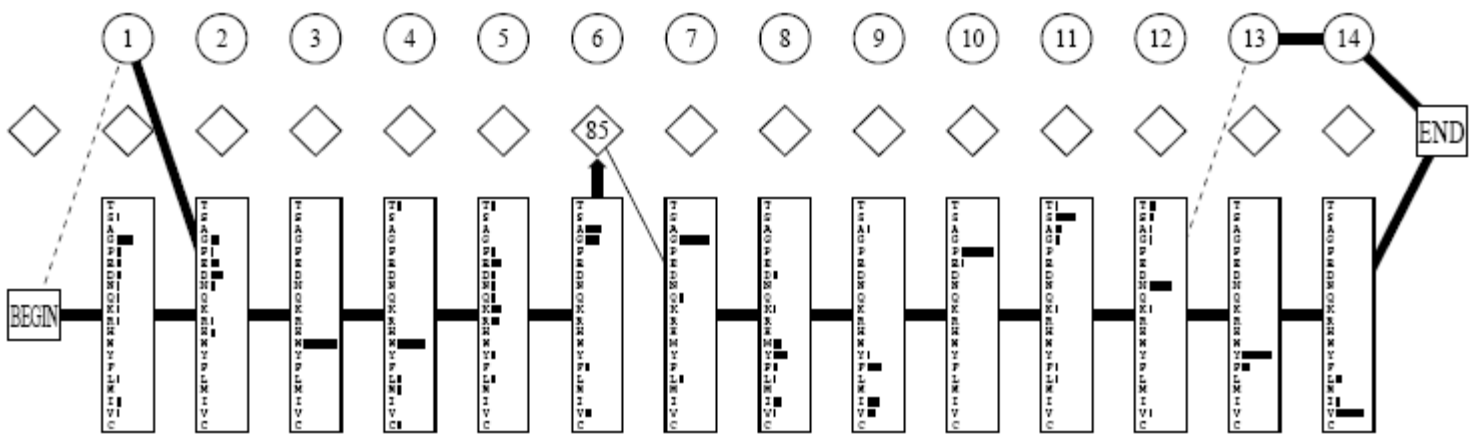


Profile HMM --- a simple case

	Sequence	Probability $\times 100$	Log odds
Consensus	A C A C - - A T C	4.7	6.7
Original	A C A - - - A T G	3.3	4.9
sequences	T C A A C T A T C	0.0075	3.0
	A C A C - - A G C	1.2	5.3
	A G A - - - A T C	3.3	4.9
	A C C G - - A T C	0.59	4.6
Exceptional	T G C T - - A G G	0.0023	-0.97

Profile HMM --- the general model

G	G	W	W	R	G	d	y	.	g	g	k	k	q	L	W	F	P	S	N	Y	V	
I	G	W	W	N	G	q	n	e	.	n	g	r	r	G	D	F	P	G	T	Y	V	
P	N	W	W	E	G	r	l	.	.	d	n	e	i	G	I	F	P	S	N	-	V	
D	E	W	W	Q	A	r	r	.	.	s	g	q	e	G	I	F	P	S	N	F	V	
G	D	W	W	L	A	r	s	.	.	k	g	q	t	G	K	V	P	S	N	Y	V	
-	D	W	W	E	A	r	s	.	.	s	g	h	r	G	Y	V	P	S	N	Y	V	
G	E	W	W	K	A	r	s	.	.	l	t	r	k	G	Y	I	P	S	N	Y	V	
G	D	W	W	L	A	r	s	.	.	v	t	g	r	G	Y	V	P	S	N	F	V	
G	E	W	W	K	A	k	s	.	.	k	n	g	q	G	F	I	P	S	N	Y	V	
G	E	W	W	C	E	A	q	.	.	t	n	g	r	G	W	V	P	S	N	Y	I	
S	D	W	W	R	V	r	n	.	.	k	t	n	g	G	L	I	P	L	N	F	V	
L	P	W	W	R	F	r	k	.	.	a	r	n	g	G	Y	I	P	S	N	Y	I	
R	E	H	W	W	K	V	k	.	.	d	v	n	g	G	Y	I	P	S	N	Y	V	
I	K	D	W	W	K	V	q	.	.	e	.	n	r	G	F	V	P	A	A	Y	V	
V	G	W	W	M	P	G	l	.	.	n	.	r	q	G	D	F	P	G	T	Y	V	
P	D	W	W	E	G	e	l	.	.	n	g	q	r	G	V	F	P	A	T	Y	V	
E	E	W	W	L	E	G	e	.	.	k	g	k	v	G	I	F	P	K	V	F	V	
G	G	W	W	K	G	e	d	.	.	y	.	.	q	G	I	F	P	P	S	N	Y	V
D	G	W	W	R	G	s	e	.	.	n	g	q	v	G	W	F	P	A	N	Y	V	
Q	G	W	W	R	G	r	r	.	.	a	n	g	e	G	I	I	P	P	T	N	Y	V
G	G	W	W	T	O	G	e	.	.	k	s	t	.	G	W	I	A	P	A	N	Y	V
G	D	W	W	E	A	r	t	.	.	n	.	.	.	G	Y	I	P	A	N	Y	V	
N	D	W	W	T	G	r	t	.	.	n	.	.	.	G	I	F	P	A	N	Y	V	

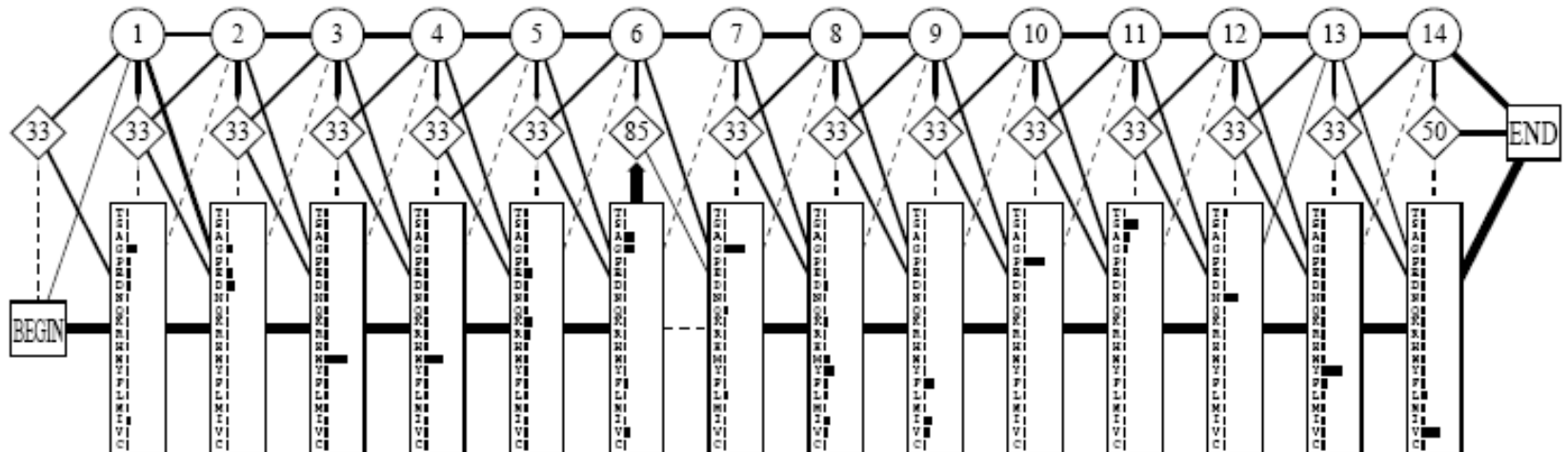


Profile HMM --- the general model

Avoid over-fitting with limited number of observed sequences.

Add pseudocounts proportional to the observed frequencies of the amino acids.

And add pseudocounts to transitions.



Profile HMM --- parameterization of the model

Make the backbone of the HMM: Given an alignment, which columns should be counted as matched states, and which should be insertions?

Find the transition and emission probabilities: (1) add pseudocounts heuristically, (2) find the frequencies as estimates of probabilities.

Profile HMM --- searching with profile HMM

Given a profile HMM, how to find if a short sequence is such a motif?

Use the Viterbi algorithm to find the best fit and the log odds ratio:

$$V_j^M(i) = \log \frac{e_{M_j}(x_i)}{q_{x_i}} + \max \begin{cases} V_{j-1}^M(i-1) + \log a_{M_{j-1}M_j}, \\ V_{j-1}^I(i-1) + \log a_{I_{j-1}M_j}, \\ V_{j-1}^D(i-1) + \log a_{D_{j-1}M_j}; \end{cases}$$

$$V_j^I(i) = \log \frac{e_{I_j}(x_i)}{q_{x_i}} + \max \begin{cases} V_j^M(i-1) + \log a_{M_jI_j}, \\ V_j^I(i-1) + \log a_{I_jI_j}, \\ V_j^D(i-1) + \log a_{D_jI_j}; \end{cases}$$

$$V_j^D(i) = \max \begin{cases} V_{j-1}^M(i) + \log a_{M_{j-1}D_j}, \\ V_{j-1}^I(i) + \log a_{I_{j-1}D_j}, \\ V_{j-1}^D(i) + \log a_{D_{j-1}D_j}. \end{cases}$$