



Bioinformatika

Hidden Markov

Models

Michael Anelli

(some slides are courtesy of Mark Craven, U. of Wisconsin)

Motivation

- ← What is a gene?

Motivation

- ⤵ What is a gene?
- ⤵ Can you formalize a gene as a regular expression?

Motivation

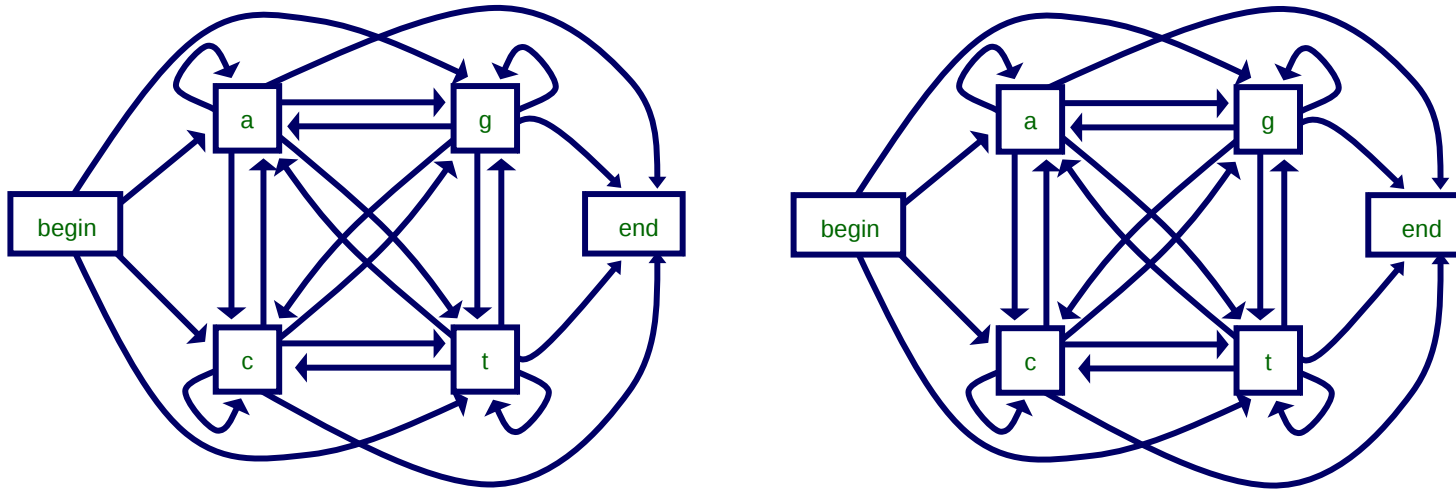
- ← What is a gene?
- ← Can you formalize a gene as a regular expression?
- ← What is HMM?

Motivation

- ← What is a gene?
- ← Can you formalize a gene as a regular expression?
- ← What is HMM?
- ← What are the general tasks with HMMs?

Motivation

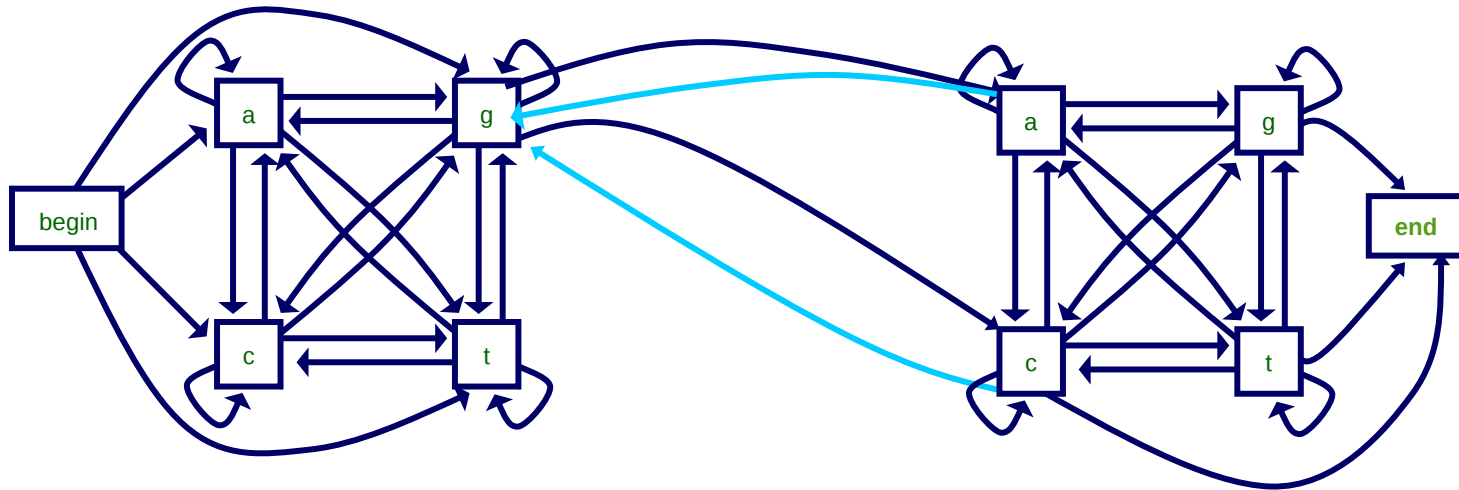
1. Train two MMs: one to represent represent CpG regions, the other to the background (nCpG)



- Given a new sequence, use two models to *classify* the sequence (CpG or nCpG).
- Given a new sequence, find the CpG islands within (**?!?**)

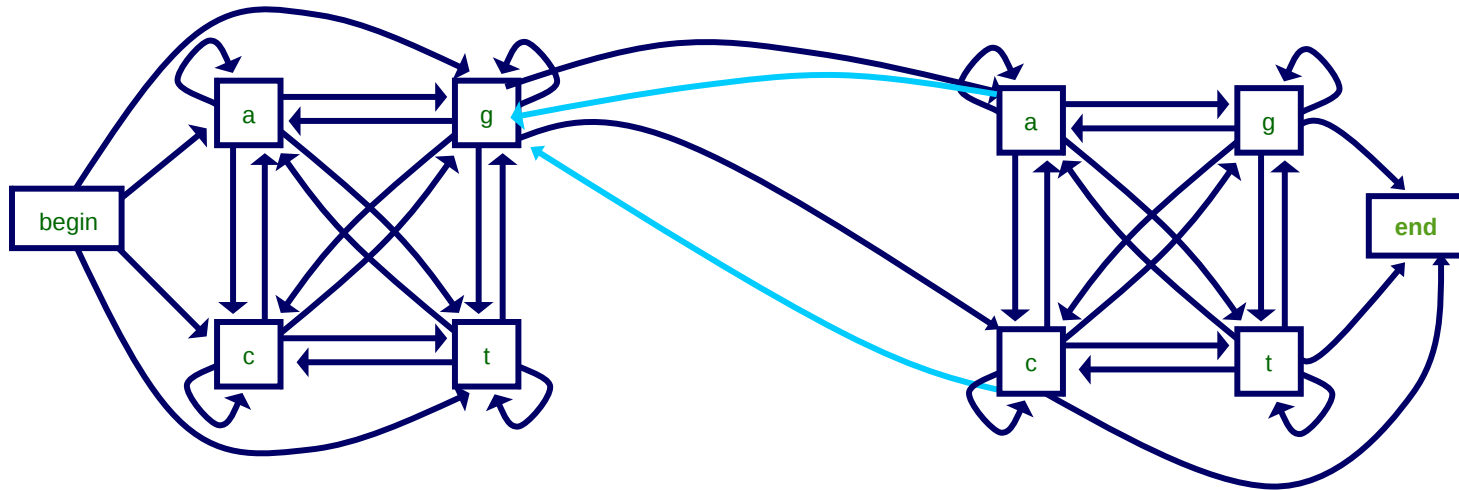
Motivation

1. Train two MMs: one to represent represent CpG regions, the other to the background (nCpG)



Motivation

1. Train two MMs: one to represent represent CpG regions, the other to the background (nCpG)



2. Join the 2 models into one HMM:

$$\rightarrow \{a, c, t, g\} \rightarrow \{a_{\text{CpG}}, a_{\text{nCpG}}, c_{\text{CpG}}, c_{\text{nCpG}}, t_{\text{CpG}}, t_{\text{nCpG}}, g_{\text{CpG}}, g_{\text{nCpG}}\}$$

3. Segment a sequence as a maximum likely walk through the state space.

Hidden Markov Model

$$M = (A, S, P_t, P_e)$$

$$\leftarrow A = \{a, c, t, g\}$$

$$\leftarrow S = \{s_1, \dots, s_K\}$$

$$\leftarrow P_t : S \times S \rightarrow [0, 1]$$

$$\leftarrow P_e : S \times A \rightarrow [0, 1]$$

$$\begin{aligned} P(x_1, \dots, x_L; s_1, \dots, s_L) &= \\ &= P(s_1) \cdot P(x_1 | s_1) \cdot P(x_2 | s_2) \cdot P(s_2 | s_1) \cdot \\ &\cdot \dots \cdot P(x_L | s_L) \cdot P(s_L | s_{L-1}) \\ &\text{with } x_i \in A, s_i \in S \end{aligned}$$

Sequence Annotation

Given:

- ↪ observed sequence $\mathbf{x} \in \{a, c, t, g\}^L$
- ↪ model $M = (A, S, P_t, P_e)$

Find:

- ↪ max. likely labeling $\mathbf{s} \in S^L \rightarrow$ Viterbi alg.

Sequence Annotation

Given:

- ⊖ observed sequence $\mathbf{x} \in \{a, c, t, g\}^L$
- ⊖ model $M = (A, S, P_t, P_e)$

Find:

- ⊖ max. likely labeling $s \in S^L \rightarrow$ Viterbi alg.

But how have the P_t, P_e been learnt??

Sequence Annotation

Given:

- ↻ observed sequence $\mathbf{x} \in \{a, c, t, g\}^L$
- ↻ model $M = (A, S, P_t, P_e)$

Find:

- ↻ max. likely labeling $s \in S^L \rightarrow$ Viterbi alg.

But how have the P_t, P_e been learnt??

- ↻ Supervised: $T = \{(x_i, s_i)\}_{i=1 \dots N}$ where $x_i \in A^*, s_i \in S^*$

Sequence Annotation

Given:

- ↻ observed sequence $\mathbf{x} \in \{a, c, t, g\}^L$
- ↻ model $M = (A, S, P_t, P_e)$

Find:

- ↻ max. likely labeling $s \in S^L \rightarrow$ Viterbi alg.

But how have the P_t, P_e been learnt??

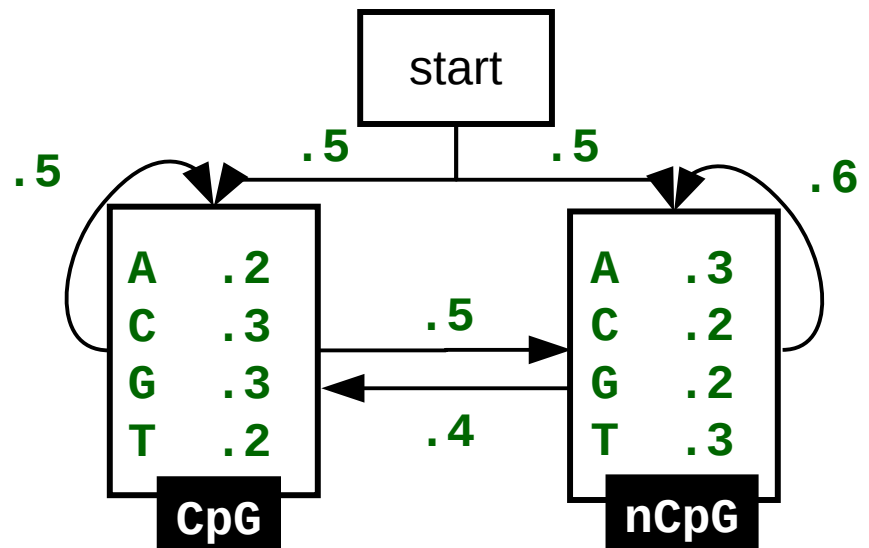
- ↻ Supervised: $T = \{(\mathbf{x}_i, \mathbf{s}_i)\}_{i=1\dots N}$ where $\mathbf{x}_i \in A^*, \mathbf{s}_i \in S^*$
- ↻ Unsupervised: $T = \{\mathbf{x}_i\}_{i=1\dots N}$ where $\mathbf{x}_i \in A^*$
 - Expectation-Maximization \rightarrow Baum-Welsh alg. (later)

Viterbi algorithm

Ex: Naive model of CpG detection

$$s^* = \arg \max_{s_0 \dots s_N \in S^N} p(x_0 \dots x_N; s_0 \dots s_N)$$

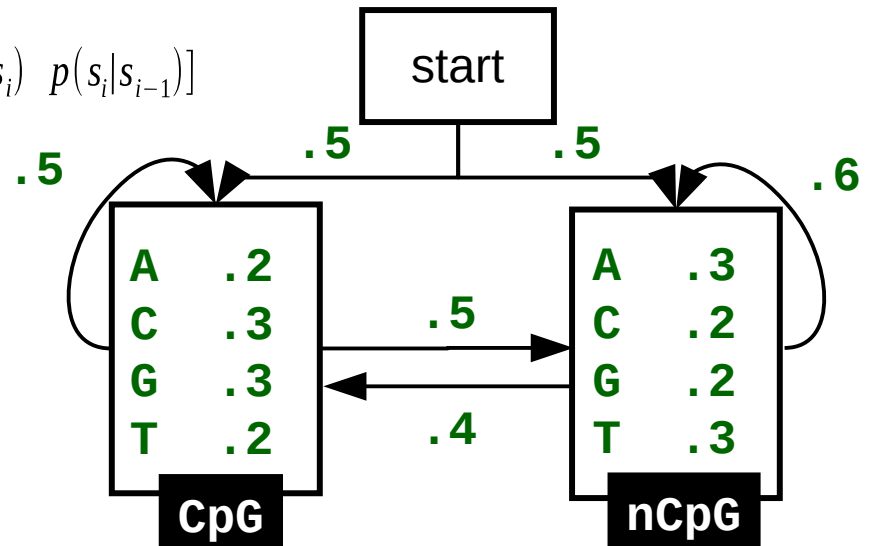
$$p(x_1 \dots x_N; s_1 \dots s_N) = \prod_{i=1}^N p(x_i | s_i) p(s_i | s_{i-1}),$$
$$p(s_0) = 1$$



Viterbi algorithm (ex.)

	ϵ	A	T	G	G	C	A	C	T	A
START	1	0	0	0	0	0	0	0	0	0
CpG	0									
nCpG	0									

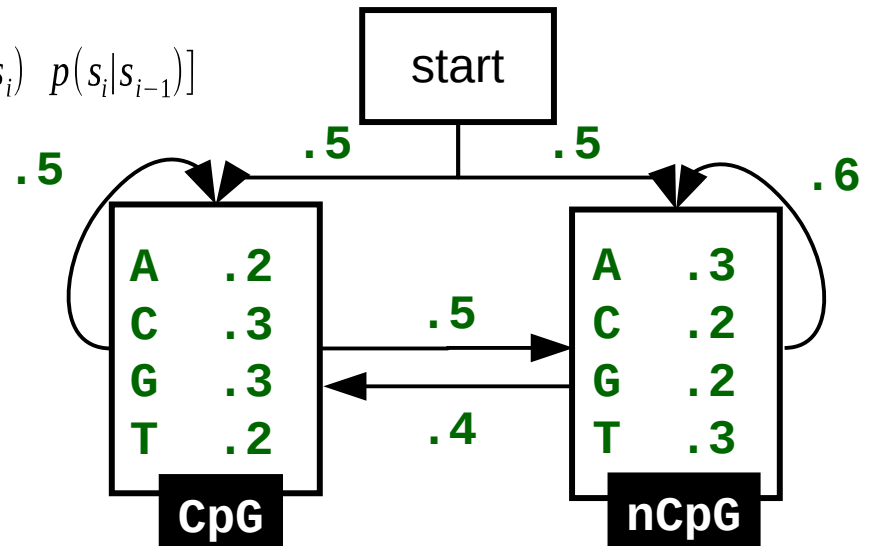
$$\max_{s_i \in S} p(x_0 \dots x_i | s_i) = \max_{s_{i-1} \in S} [p(x_0 \dots x_{i-1} | s_{i-1}) \max_{s_i \in S} p(x_i | s_i) p(s_i | s_{i-1})]$$



Viterbi algorithm (ex.)

	ϵ	A	T	G	G	C	A	C	T	A
START	1	0	0	0	0	0	0	0	0	0
CpG	0	1 x .2 x .5 0 x .2 x .5 0 x .2 x .4 .1								
nCpG	0	1 x .3 x .5 0 x .3 x .5 0 .3 xx .6 .15								

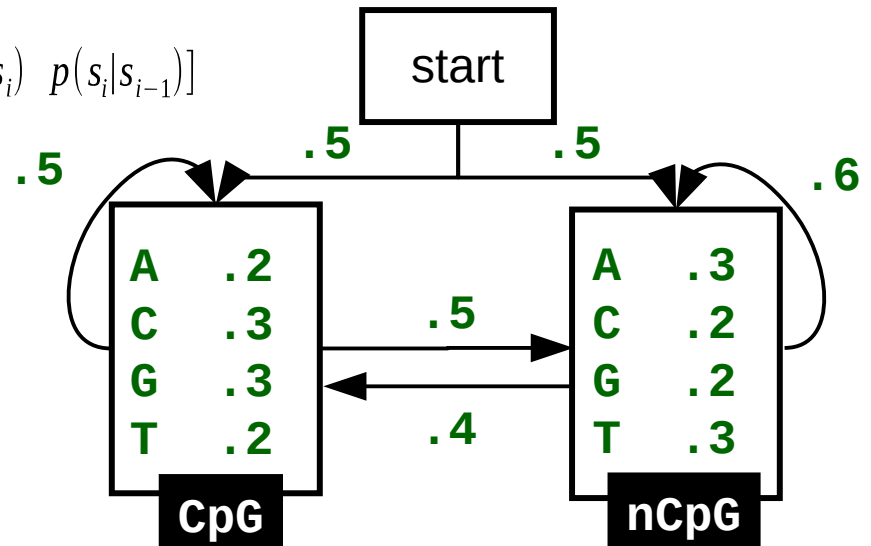
$$\max_{s_i \in S} p(x_0 \dots x_i | s_i) = \max_{s_{i-1} \in S} [p(x_0 \dots x_{i-1} | s_{i-1}) \max_{s_i \in S} p(x_i | s_i) p(s_i | s_{i-1})]$$



Viterbi algorithm (ex.)

	ϵ	A	T	G	G	C	A	C	T	A
START	1	0	0	0	0	0	0	0	0	0
CpG	0	1 x .2 x .5 0 x .2 x .5 0 x .2 x .4 .1	0 x .2 x .5 .1 x .2 x .5 .15 x .2 x .4 .012							
nCpG	0	1 x .3 x .5 0 x .3 x .5 0 .3 x .6 .15	0 x .3 x .5 .1 x .3 x .5 .15 x .3 x .6 .027							

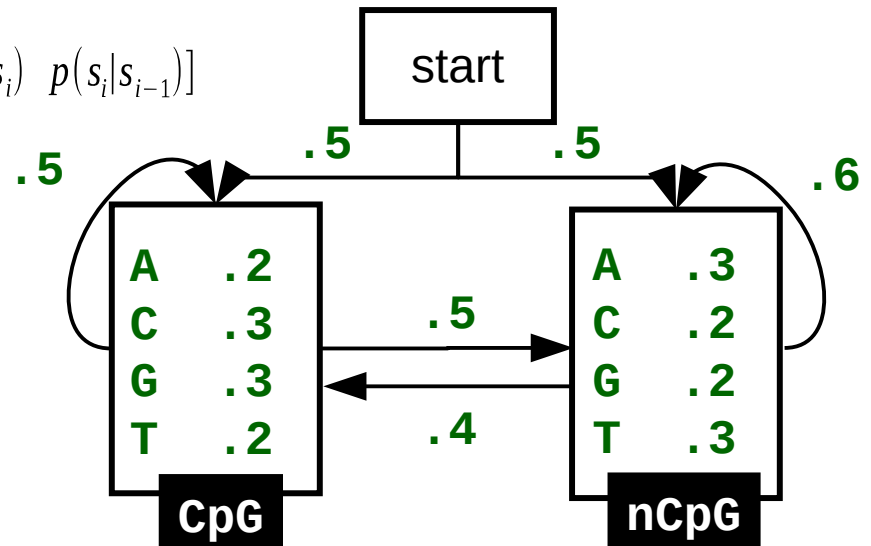
$$\max_{s_i \in S} p(x_0 \dots x_i | s_i) = \max_{s_{i-1} \in S} [p(x_0 \dots x_{i-1} | s_{i-1}) \max_{s_i \in S} p(x_i | s_i) p(s_i | s_{i-1})]$$



Viterbi algorithm (ex.)

	ϵ	A	T	G	G	C	A	C	T	A
START	1	0	0	0	0	0	0	0	0	0
CpG	0	$1 \times .2 \times .5$ $0 \times .2 \times .5$ $0 \times .2 \times .4$ $.1$	$0 \times .2 \times .5$ $.1 \times .2 \times .5$ $.15 \times .2 \times .4$ $.012$	0	$.012 \times .3 \times .5$ $.027 \times .3 \times .4$ $.0032$	0 $.0032 \times .3 \times .5$ $.0032 \times .3 \times .4$ $5e-4$	0 $.012 \times .3 \times .5$ $.027 \times .3 \times .4$ $5e-5$			
nCpG	0	$1 \times .3 \times .5$ $0 \times .3 \times .5$ $0.3 \times .6$ $.15$	$0 \times .3 \times .5$ $.1 \times .3 \times .5$ $.15 \times .3 \times .6$ $.027$	0	$.012 \times .2 \times .5$ $.027 \times .2 \times .6$ $.0032$	0 $.0032 \times .2 \times .5$ $.0032 \times .2 \times .6$ $4e-4$	0 $.012 \times .2 \times .5$ $.027 \times .2 \times .6$ $4e-5$			

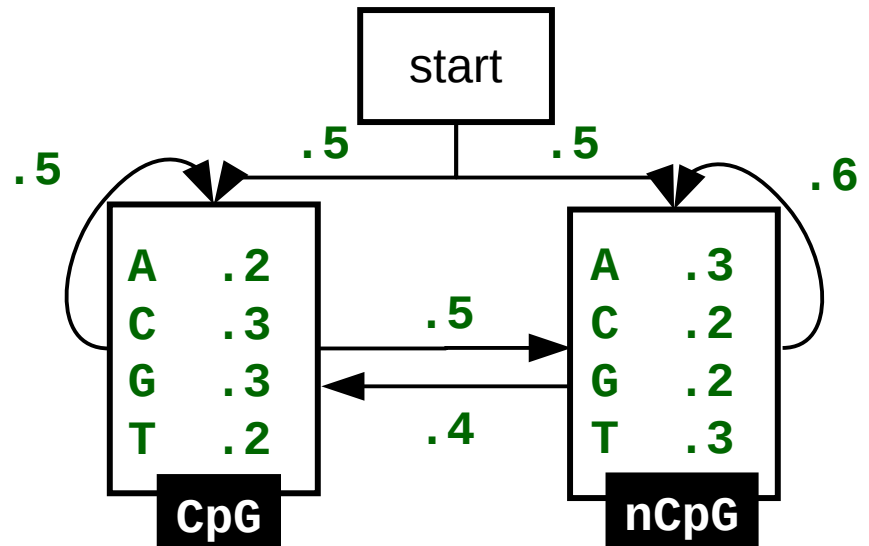
$$\max_{s_i \in S} p(x_0 \dots x_i | s_i) = \max_{s_{i-1} \in S} [p(x_0 \dots x_{i-1} | s_{i-1}) \max_{s_i \in S} p(x_i | s_i) p(s_i | s_{i-1})]$$



Viterbi algorithm (ex.)

	ϵ	A	T	G	G	C	A	C	T	A
START	0	-inf	-inf	-inf	-inf	-inf	-inf	-inf	-inf	-inf
CpG	-inf	ln.2+0+ln.5 ln.2+ -inf +ln.5 ln.2+ -inf +ln.4 -2.30	ln.2+ -inf +ln.5 ln.2+0+ln.5 ln.2+ln.15+ln.4 -2.3							
nCpG	-inf	ln.3+0+ln.5 ln.3+ -inf +ln.5 ln.3+ -inf +ln.6 -1.9	-inf ln.3+ln.1+ln.5 ln.3+ln.15+ln.6 -1.9							

$$\arg \max_{s_i \in S} p(x_0 \dots x_i | s_i) = \arg \max_{s_i \in S} \log p(x_0 \dots x_i | s_i)$$



Assignment – Gene Finding

- ⬅ <http://www.biostat.wisc.edu/~craven/776/hw3.html>
- ⬅ **You can use an existing implementation of Viterbi alg.**
- ⬅ 15 pt.

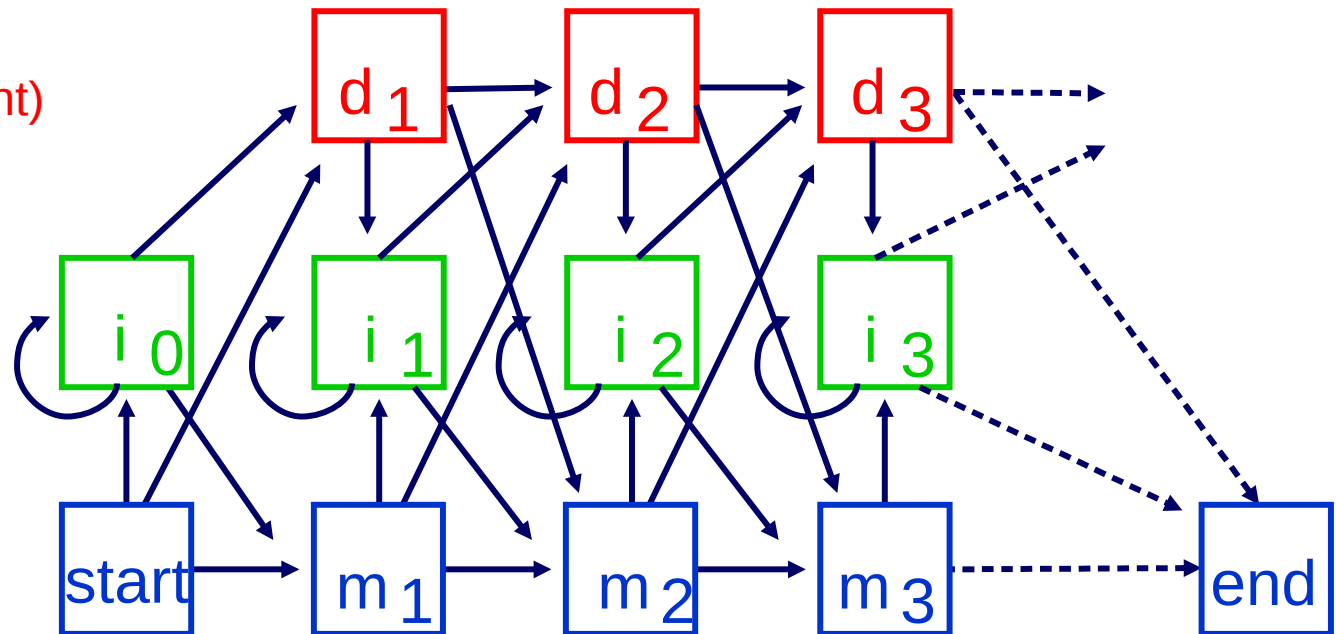
Profile HMM

ATTGCC- A TT--
ATGGCC- A TT--
ATC-CA- A TTTT
ATCTTC- - TT--
ATTGCCG A TT--

Delete states (silent)

Insert states

Match states



Profile HMM – Exercise

AG - - - C

A - AG - C

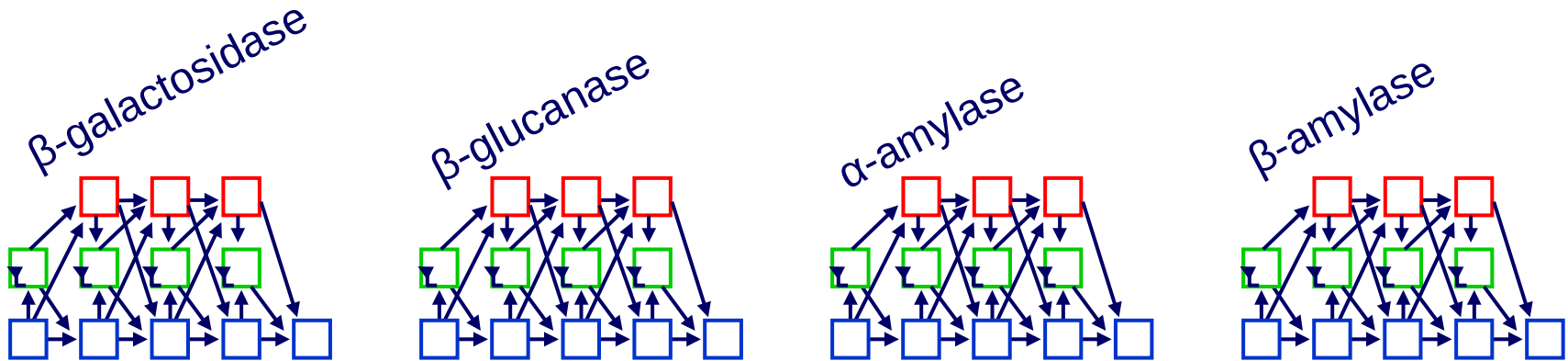
AG - AA -

- - AAAC

AG - - - C

12 - 3 - 4

Sequence Categorization



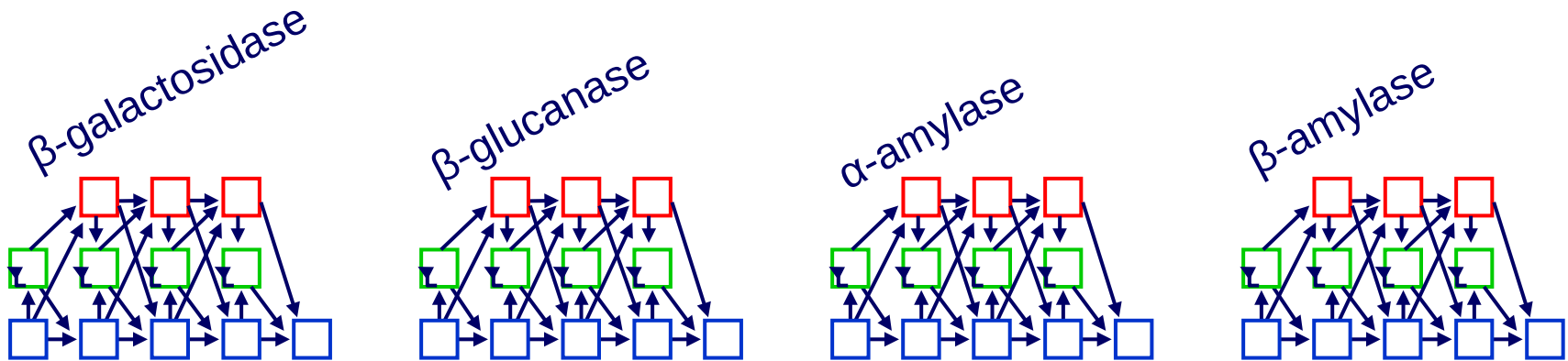
Given:

- ⌚ observed sequence $\mathbf{x} \in \{a, c, t, g\}^L$
- ⌚ Set of K models $\{M_k = (A, S, P_t, P_e)\}$ of K families

Do:

- ⌚ categorize \mathbf{x} into one of the families

Sequence Categorization



Given:

- ↻ observed sequence $\mathbf{x} \in \{a, c, t, g\}^L$
- ↻ Set of K models $\{M_k = (A, S, P_t, P_e)\}$ of K families

Do:

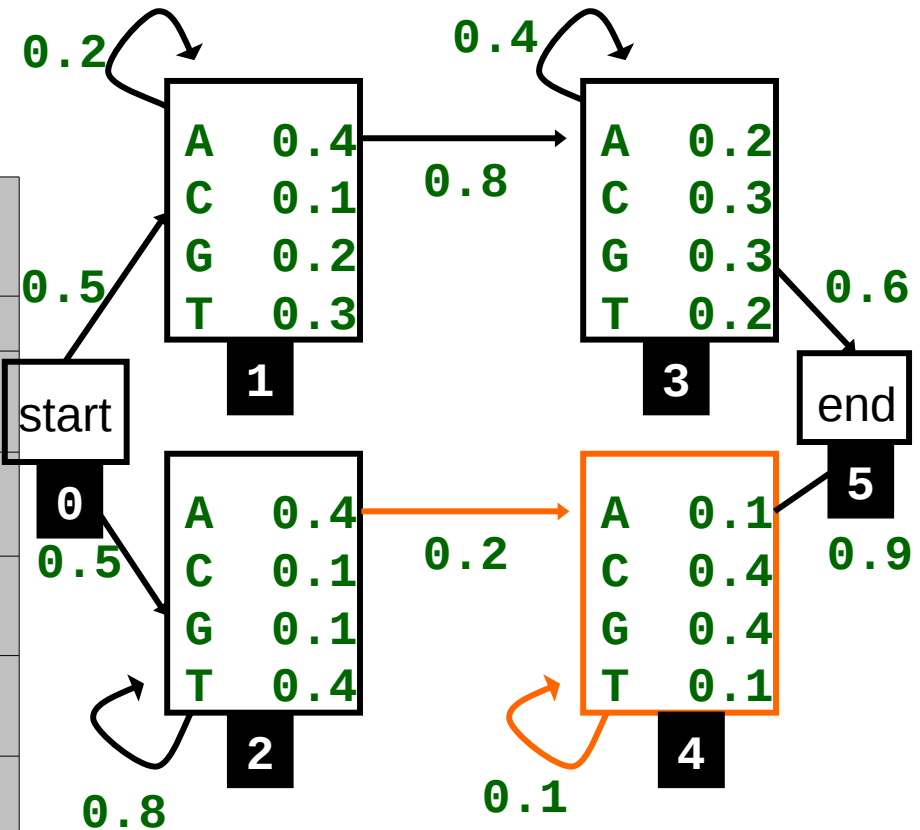
- ↻ categorize \mathbf{x} into one of the families

$$p(\alpha\text{-amyl.} | x_0 \dots x_N) < p(\beta\text{-gluc.} | x_0 \dots x_N)$$
$$p(\alpha\text{-amyl.}) p(x_0 \dots x_N | \alpha\text{-amyl.}) < p(\beta\text{-gluc.}) p(x_0 \dots x_N | \beta\text{-gluc.})$$
$$p(x_0 \dots x_N | \text{family}_k) = \sum_{s_0 \dots s_N \in S^N} p(x_0 \dots x_N; s_0 \dots s_N | \text{family}_k)$$

Forward algorithm (ex.)

$$\sum_{s_1 \dots s_i} p(x_0 \dots x_i, s_1 \dots s_i) = \sum_{s_i \in S} \sum_{s_1 \dots s_{i-1}} p(x_1 \dots x_{i-1}, s_1 \dots s_{i-1}) p(x_i | s_i) p(s_i | s_{i-1})$$

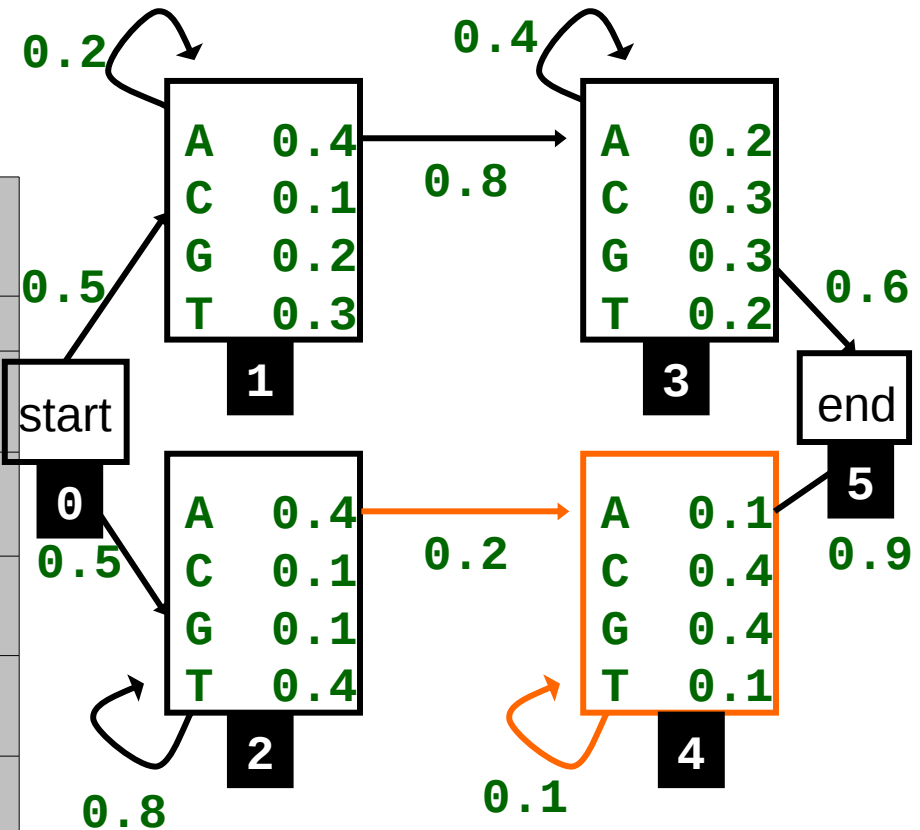
	ϵ	T	A	G	A	ϵ
0	1	0	0	0	0	
1	0					
2	0					
3						
4						
5		0	0	0	0	



Forward algorithm (ex.)

$$\sum_{s_1 \dots s_i} p(x_0 \dots x_i, s_1 \dots s_i) = \sum_{s_i \in S} \sum_{s_1 \dots s_{i-1}} p(x_1 \dots x_{i-1}, s_1 \dots s_{i-1}) p(x_i | s_i) p(s_i | s_{i-1})$$

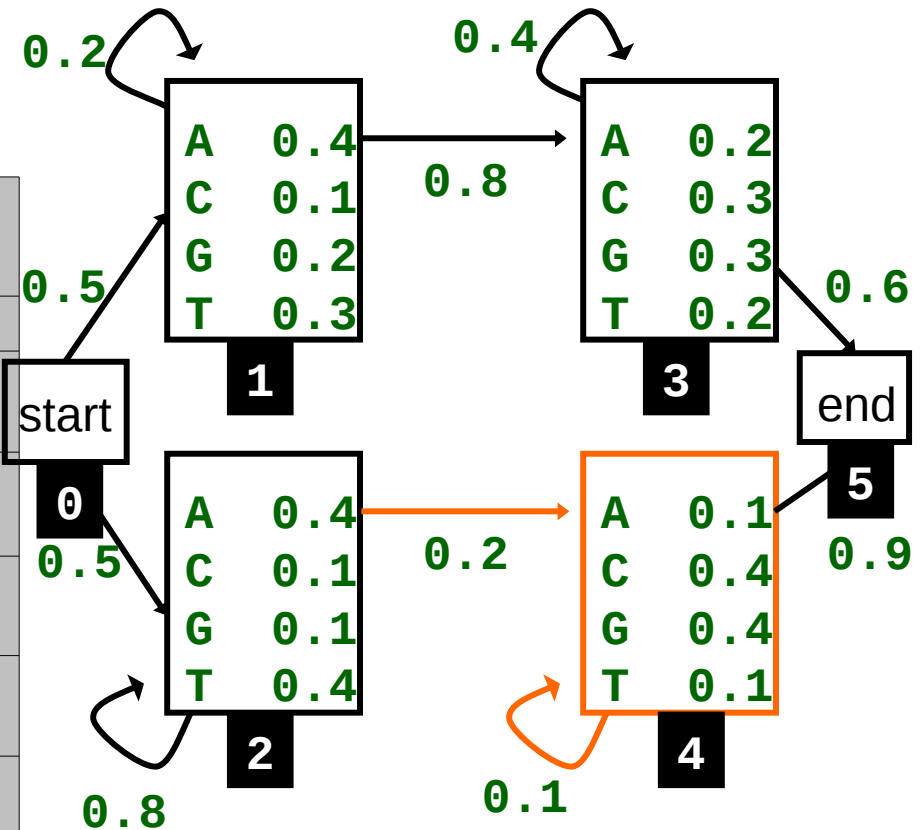
	ϵ	T	A	G	A	ϵ
0	1	0	0	0	0	
1	0	$\begin{matrix} 1 \times .3 \times .5 \\ 0 \times .3 \times .2 \\ .15 \end{matrix}$				
2	0	$\begin{matrix} 1 \times .4 \times .5 \\ 0 \times .4 \times .8 \\ .2 \end{matrix}$				
3		0				
4		0				
5		0	0	0	0	



Forward algorithm (ex.)

$$\sum_{s_1 \dots s_i} p(x_0 \dots x_i, s_1 \dots s_i) = \sum_{s_i \in S} \sum_{s_1 \dots s_{i-1}} p(x_1 \dots x_{i-1}, s_1 \dots s_{i-1}) p(x_i | s_i) p(s_i | s_{i-1})$$

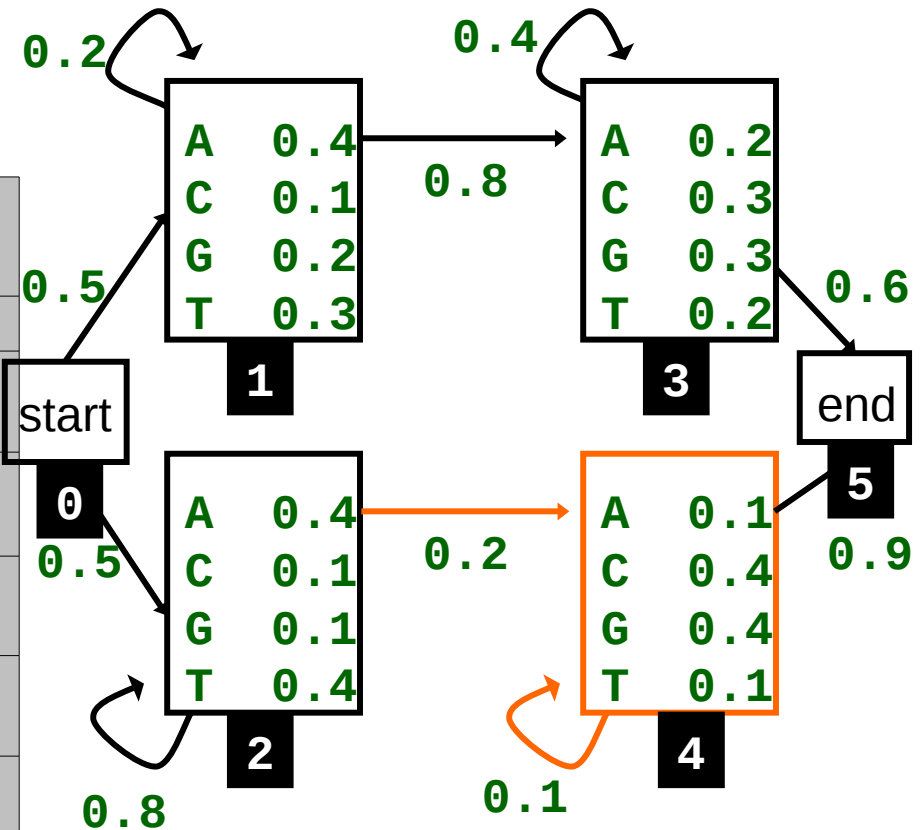
	ϵ	T	A	G	A	ϵ
0	1	0	0	0	0	
1	0	$1 \times .3 \times .5$ $0 \times .3 \times .2$.15	$0 \times .4 \times .5$ $.15 \times .4 \times .2$.012			
2	0	$1 \times .4 \times .5$ $0 \times .4 \times .8$.2	$0 \times .4 \times .5$ $.2 \times .4 \times .8$.064			
3		0	$.15 \times .2 \times .8$ $0 \times .2 \times .4$.024			
4		0	$.2 \times .1 \times .2$ $0 \times .1 \times .1$.004			
5		0	0	0	0	



Forward algorithm (ex.)

$$\sum_{s_1 \dots s_i} p(x_0 \dots x_i, s_1 \dots s_i) = \sum_{s_i \in S} \sum_{s_1 \dots s_{i-1}} p(x_1 \dots x_{i-1}, s_1 \dots s_{i-1}) p(x_i | s_i) p(s_i | s_{i-1})$$

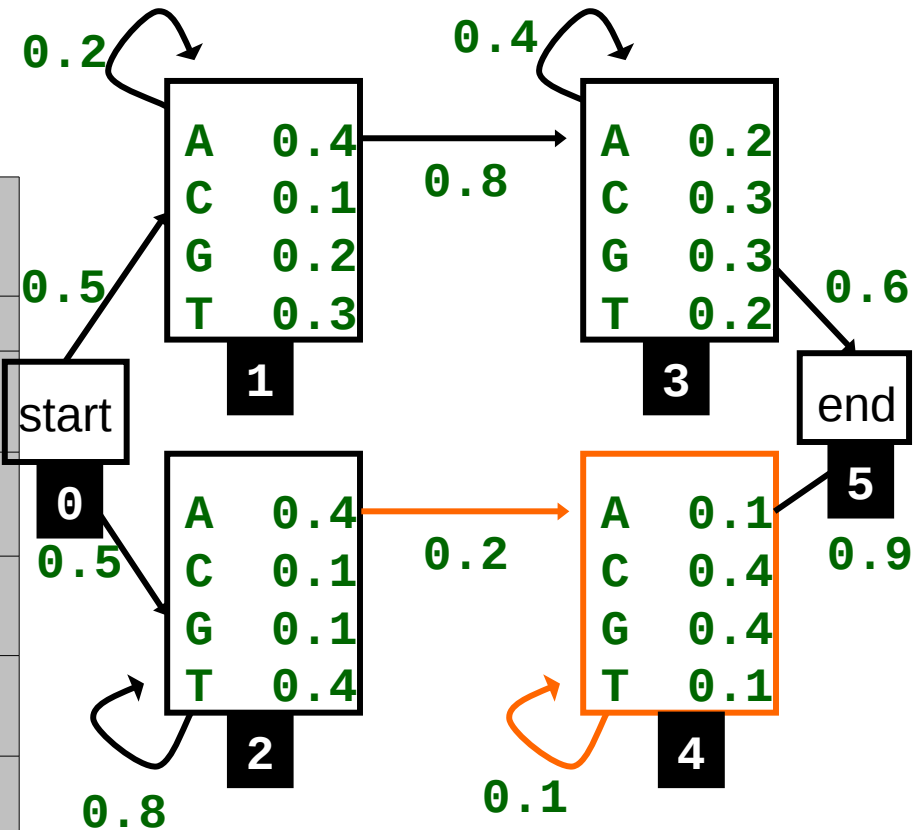
	ϵ	T	A	G	A	ϵ
0	1	0	0	0	0	
1	0	$1 \times .3 \times .5$ $0 \times .3 \times .2$.15	$0 \times .4 \times .5$ $.15 \times .4 \times .2$.012	$0 \times .2 \times .5$ $.012 \times .2 \times .2$ 5e-4		
2	0	$1 \times .4 \times .5$ $0 \times .4 \times .8$.2	$0 \times .4 \times .5$ $.2 \times .4 \times .8$.064	$0 \times .1 \times .5$ $.064 \times .1 \times .8$.00512		
3		0	$.15 \times .2 \times .8$ $0 \times .2 \times .4$.024	$.012 \times .3 \times .8$ $.024 \times .3 \times .4$.00576		
4		0	$.2 \times .1 \times .2$ $0 \times .1 \times .1$.004	$.064 \times .4 \times .2$ $.004 \times .4 \times .1$.00528	.	
5		0	0	0	0	



Forward algorithm (ex.)

$$p(x_0 \dots x_i) = \sum_{s_i \in S} p(x_0 \dots x_{i-1}) p(x_i | s_i) p(s_i | s_{i-1})$$

	ϵ	T	A	G	A	ϵ
0	1	0	0	0	0	
1	0	$1 \times .3 \times .5$ $0 \times .3 \times .2$.15	$0 \times .4 \times .5$ $.15 \times .4 \times .2$.012	$0 \times .2 \times .5$ $.012 \times .2 \times .2$ $5e-4$	$0 \times .4 \times .5$ $5e-4 \times .4 \times .2$ $4e-5$	0
2	0	$1 \times .4 \times .5$ $0 \times .4 \times .8$.2	$0 \times .4 \times .5$ $.2 \times .4 \times .8$.064	$0 \times .1 \times .5$ $.064 \times .1 \times .8$.00512	$0 \times .4 \times .5$ $5e-3 \times .4 \times .8$.0016	0
3		0	$.15 \times .2 \times .8$ $0 \times .2 \times .4$.024	$.012 \times .3 \times .8$ $.024 \times .3 \times .4$.00576	$5e-4 \times .2 \times .8$ $6e-3 \times .2 \times .4$ $6e-4$	0
4		0	$.2 \times .1 \times .2$ $0 \times .1 \times .1$.004	$.064 \times .4 \times .2$ $.004 \times .4 \times .1$.00528	$.005 \times .1 \times .2$ $.005 \times .1 \times .1$ $1.5e-4$	0
5		0	0	0	0	$6e-4 \times .6$ $1.5e-4 \times .9$ $4.6e-4$



Sum-up

- ↩ Sequence categorization into family of sequences (Forward alg.)
- ↩ Sequence anotation: CpG detection, gene finding (Viterbi alg.)
- ↩ Learning ***hidden*** parameters (Baum-Welsh alg.)