Bioinformatika
Hidden Markov Models

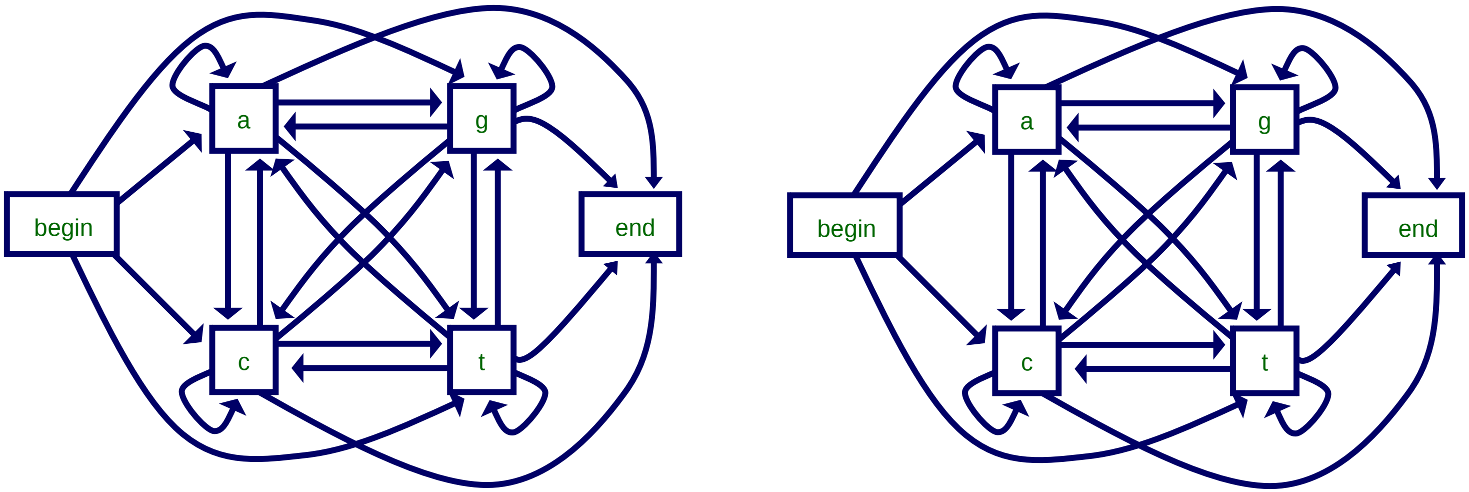
Michael Andel
(some slides are courtesy of Mark Craven, U. of Wisconsin)

Motivation

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

Motivation

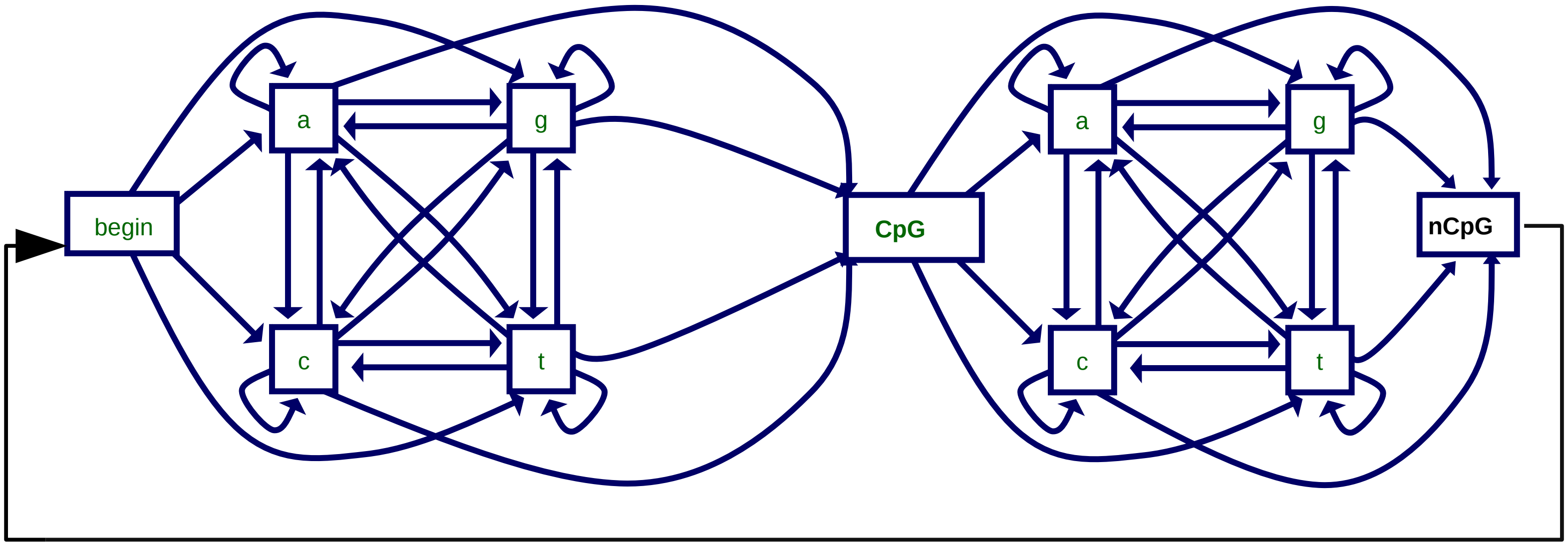
- Train two MMs: one to represent background sequence regions (*null*), another to represent CpG regions.



1. Given a test sequence, use two models to classify the sequence (*CpG* or *null*).
2. Given a test sequence, find CpG islands within. (**?!**)

Motivation

1. Train two MMs: one to represent background sequence regions (*null*), another to represent CpG regions.



2. Join the 2 models into one HMM,
3. Segment given test sequence into CpG and non-CpG regions. **How?**

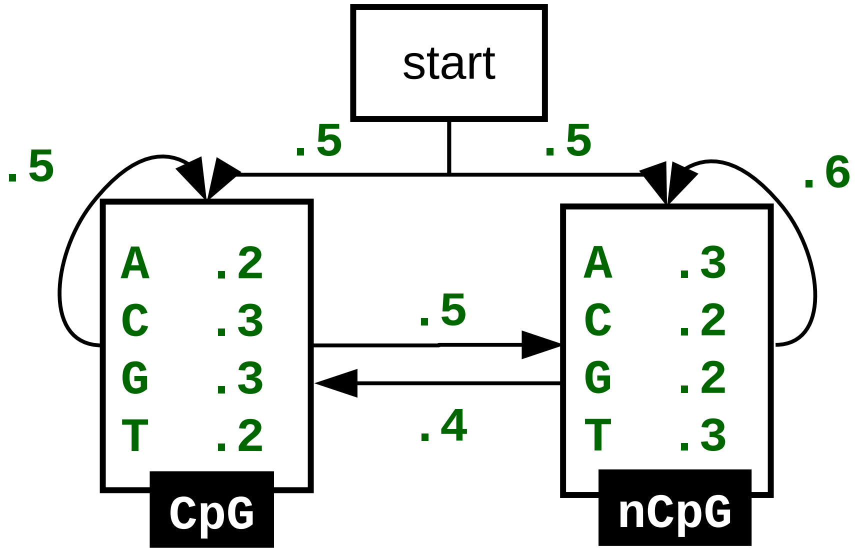
Viterbi algorithm

- Given an observed sequence ***x***.
- What is the most likely path ***s*** through the model, i.e. sequence anotation?
- Ex: Naive model of CpG detection

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

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

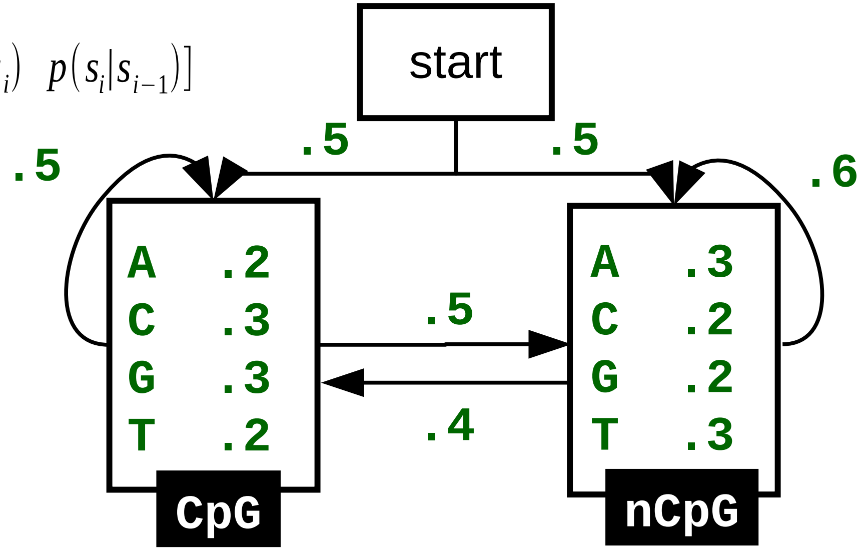
$$p(s_0) = 1$$



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									

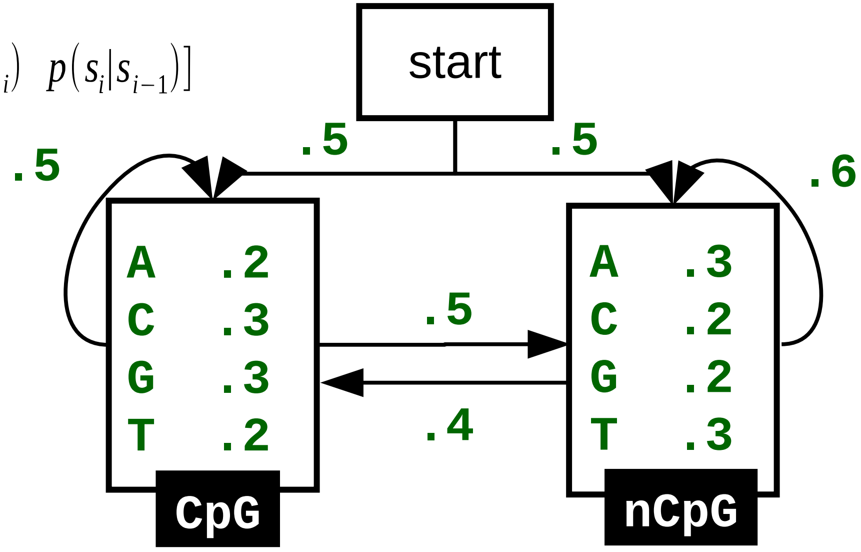
$$\max_{s_i \in S} p(x_0 \ldots x_i | s_i) = \max_{s_{i-1} \in S} [p(x_0 \ldots x_{i-1} | s_{i-1}) \max_{s_i \in S} p(x_i | s_i) \; p(s_i | s_{i-1})]$$



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								

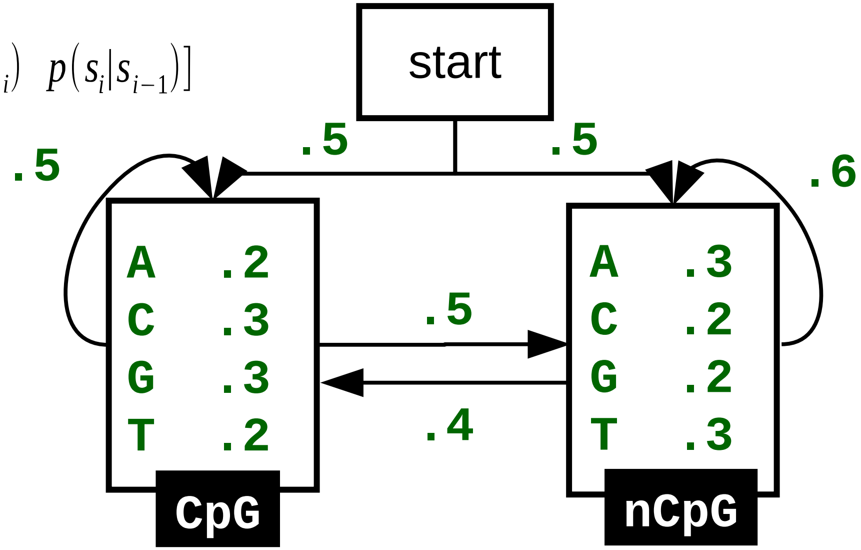
$$\max_{s_i \in S} p(x_0 \ldots x_i | s_i) = \max_{s_{i-1} \in S} [p(x_0 \ldots x_{i-1} | s_{i-1}) \ \max_{s_i \in S} p(x_i | s_i) \ \ p(s_i | s_{i-1})]$$



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 xx .6 .15	**0** x .3 x .5 .1 x .3 x .5 .15 x .3 x .6 .027							

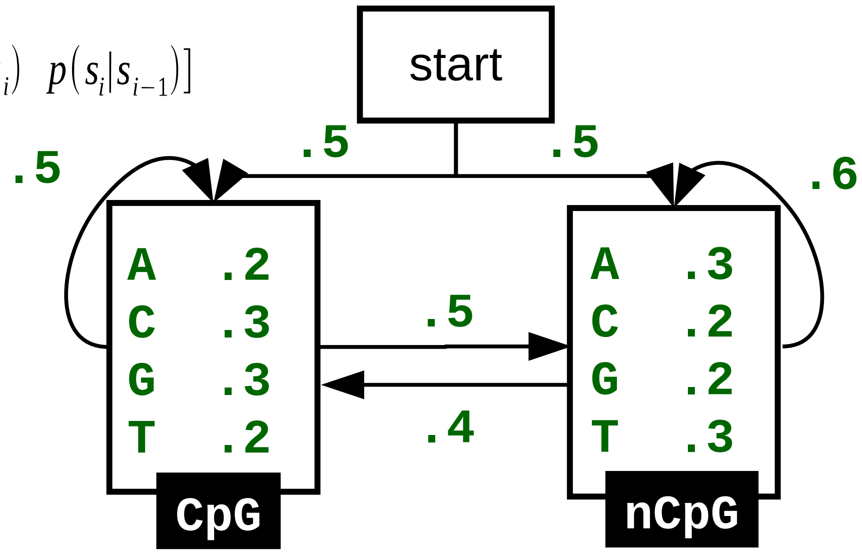
$$\max_{s_i \in S} p(x_0 \ldots x_i | s_i) = \max_{s_{i-1} \in S} [p(x_0 \ldots x_{i-1} | s_{i-1}) \max_{s_i \in S} p(x_i | s_i) \ \ p(s_i | s_{i-1})]$$



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	0 .012 x .3 x .5 .027 x .3 x .4 .0032	0 .0032 x .3 x .5 .0032 x .3 x .4 5e-4	0 .012 x.3 x .5 .027 x .3 x .4 5e-5				
nCpG	0	**1** x .3 x .5 0 x .3 x .5 0 .3 xx .6 .15	**0** x .3 x .5 .1 x .3 x .5 .15 x .3 x .6 .027	0 .012 x .2 x .5 .027 x .2 x .6 .0032	0 .0032 x .2 x .5 .0032 x .2 x .6 4e-4	0 .012 x .2 x .5 .027 x .2 x .6 4e-5				

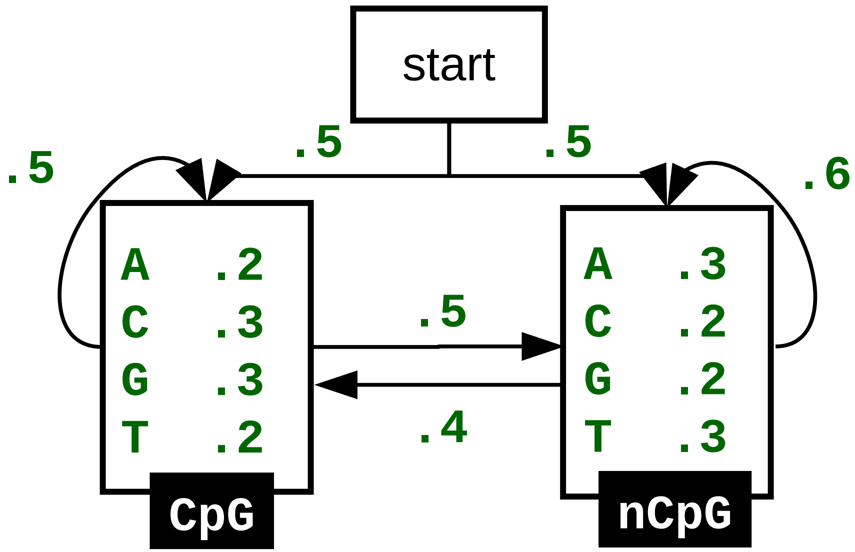
$$\max_{s_i \in S} p(x_0 \ldots x_i | s_i) = \max_{s_{i-1} \in S} \left[p(x_0 \ldots x_{i-1} | s_{i-1}) \ \max_{s_i \in S} p(x_i | s_i) \ \ p(s_i | s_{i-1}) \right]$$



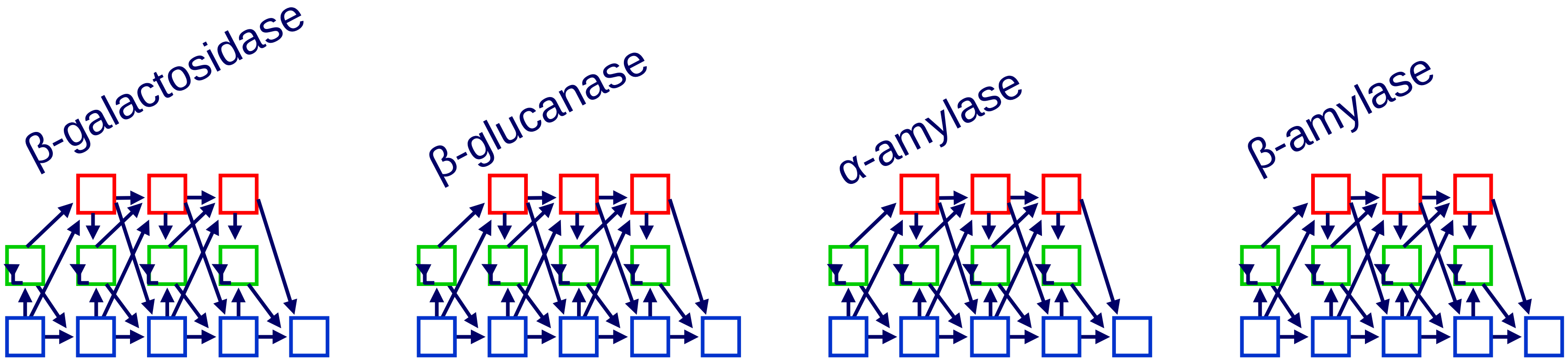
Viterbi algorithm (ex.)

	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							

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



Forward algorithm



- Given ***K*** models of ***K*** sequence families.
- Categorize a new sequence ***x***.

$$p(\alpha-amyl.)\,p(x_0\ldots x_N|\alpha-amyl.) < p(\beta-amyl.)\,p(x_0\ldots x_N|\beta-amyl.)$$

$$p(x_0\ldots x_N|\alpha-amyl.) = \sum_{s_0\ldots s_N \in S^N} p(x_0\ldots x_N;s_0\ldots s_N|\alpha-amyl.)$$

Forward algorithm

$$p(x_1 \ldots x_N) = \sum_{s_1 \ldots s_N \in S^N} p(x_1 \ldots x_N, s_1 \ldots s_N) = \ldots$$

$$\sum_{s_1 \ldots s_{N-1} s_N \in S^N} p(s_1) p(x_1|s_1) \ldots p(s_N|s_{N-1}) p(x_N|s_N) = \ldots$$

$$\sum_{s_N} [\sum_{s_1 \ldots s_{N-1}} p(s_1) p(x_1|s_1) \ldots p(s_N|s_{N-1})] p(x_N|s_N) = \ldots$$

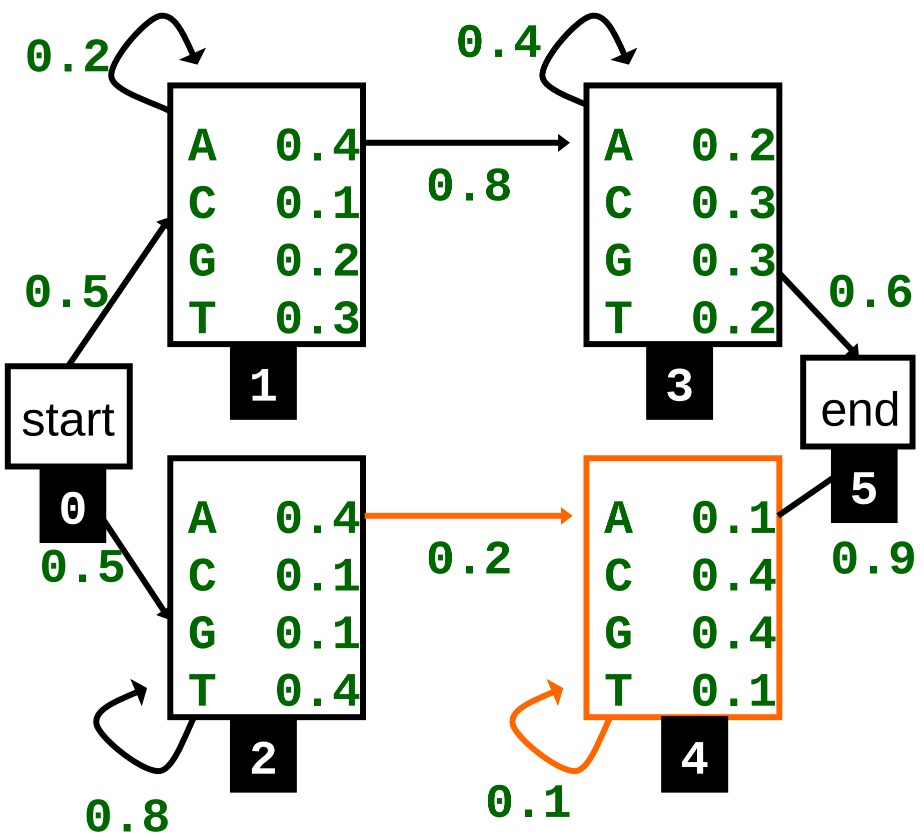
$$\sum_{s_N} [\sum_{s_1 \ldots s_{N-1}} p(x_1 \ldots x_{N-1}, s_1 \ldots s_N)] p(x_N|s_N) = \ldots$$

$$\sum_{s_N} p(x_1 \ldots x_{N-1}, s_N) p(x_N|s_N) = \ldots$$

$$\sum_{s_N} p(x_1 \ldots x_{N-1}|s_N) p(x_N, s_N)$$

$$\sum_{s_1 \ldots s_i} p(x_1 \ldots x_i, s_1 \ldots s_i) = \sum_{s_i} p(x_1 \ldots x_{i-1}, s_i) p(x_i|s_i),$$

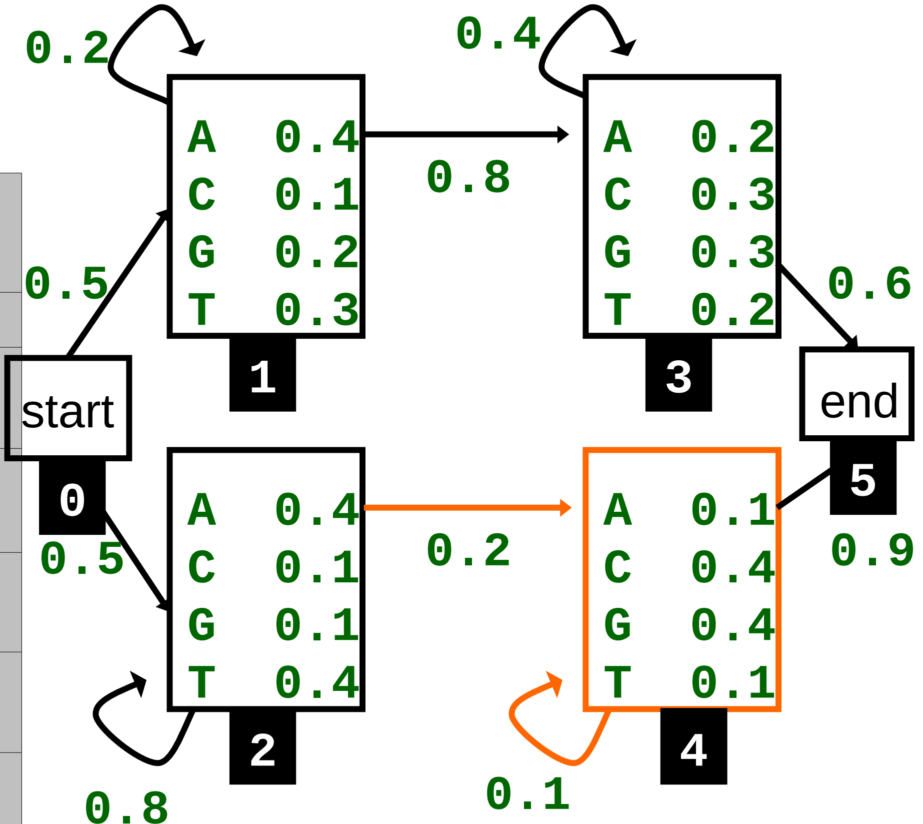
$$p(x_1 \ldots x_{i-1}, s_i) = \sum_{s_1 \ldots s_{i-1}} p(x_1 \ldots x_{i-1}, s_1 \ldots s_{i-1}) p(s_i|s_{i-1})$$



Forward algorithm (ex.)

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

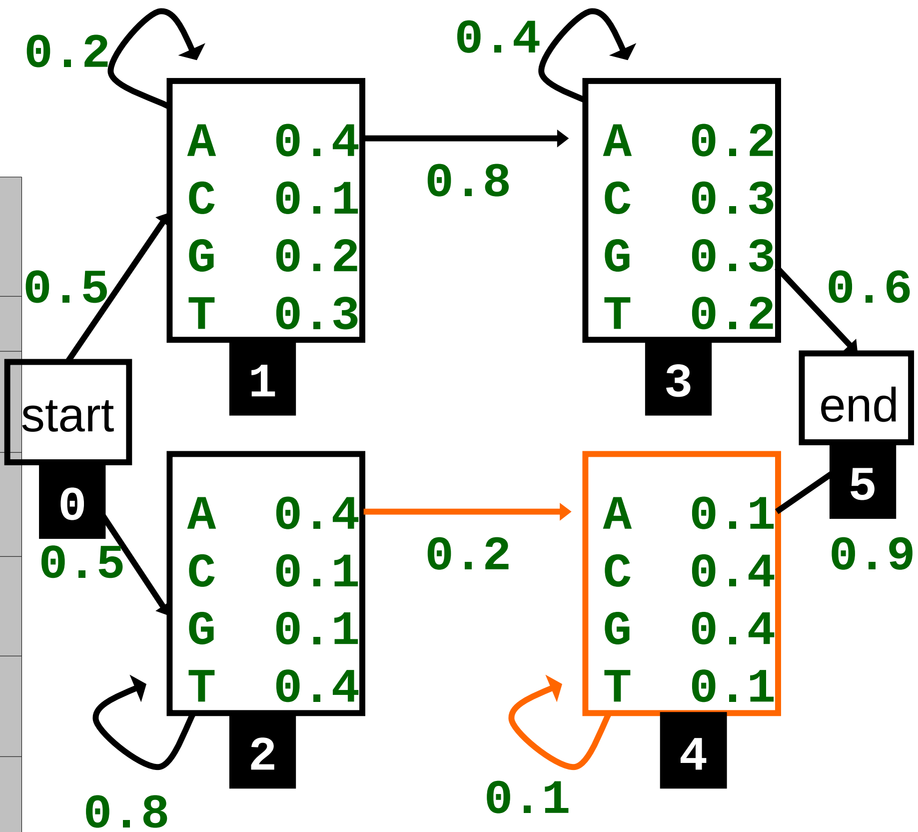
	ε	T	A	G	A	ε
0	**1**	0	0	0	0	
1	0					
2	0					
3						
4						
5		0	0	0	0	



Forward algorithm (ex.)

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

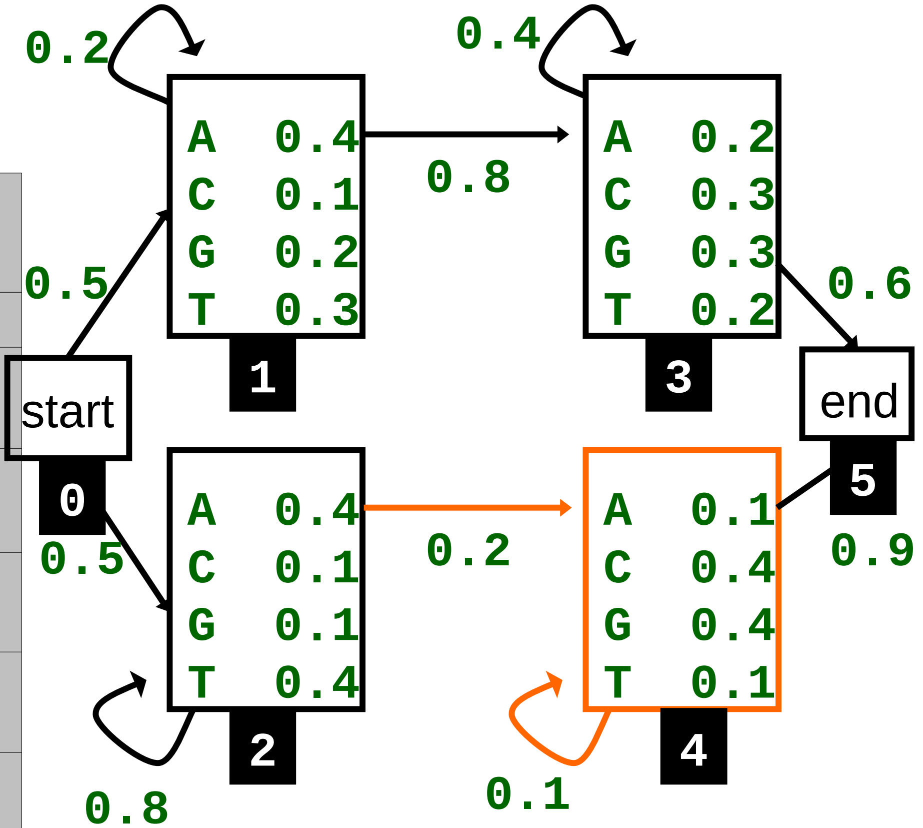
	ε	T	A	G	A	ε
0	**1**	0	0	0	0	
1	0	**1** x .3 x .5 0 x .3 x .2 .15				
2	0	**1** x .4 x .5 0 x .4 x .8 .2				
3		0				
4		0				
5		0	0	0	0	



Forward algorithm (ex.)

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

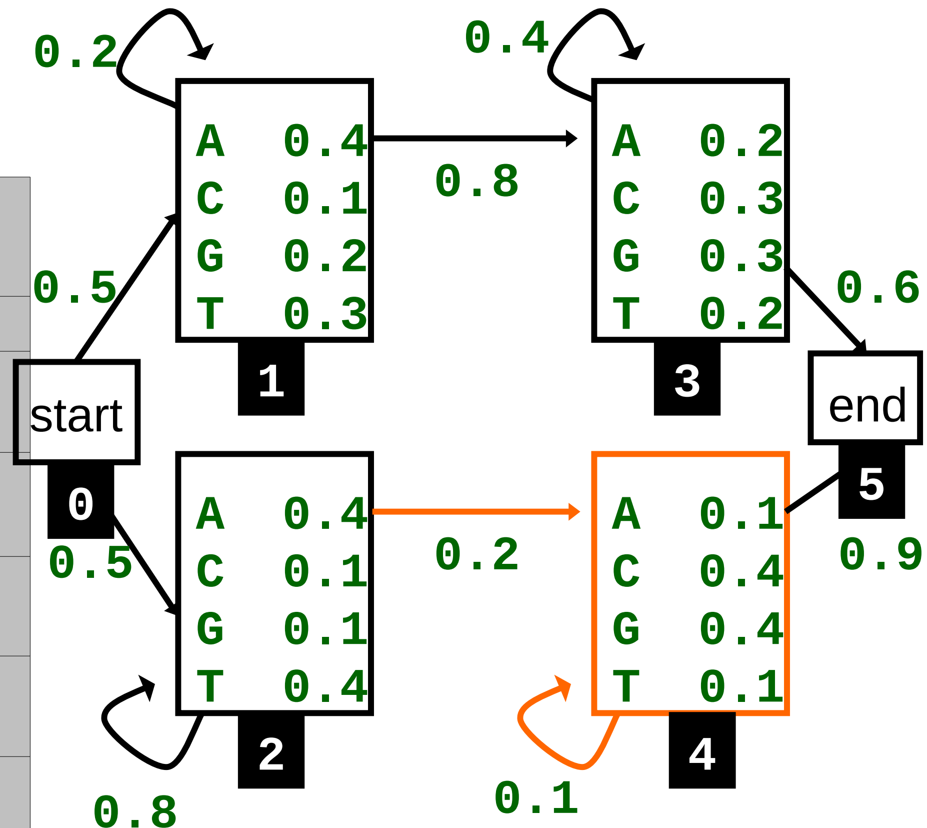
	ε	T	A	G	A	ε
0	**1**	0	0	0	0	
1	0	**1** x .3 x .5 0 x .3 x .2 .15	**0** x .4 x .5 .15 x .4 x .2 .012			
2	0	**1** x .4 x .5 0 x .4 x .8 .2	**0** x .4 x .5 .2 x .4 x .8 .064			
3		0	.15 x .2 x .8 0 x .2 x .4 .024			
4		0	.2 x .1 x .2 0 x .1 x .1 .004			
5		0	0	0	0	



Forward algorithm (ex.)

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

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



Forward algorithm (ex.)

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

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