CS 466: Final Review

December 9, 2020

Review Session Topics

1. Solutions to HW5 (Genome Assembly)

2. RNA Secondary Structure Prediction

3. Phylogeny Inference
   - Distance-based
   - Character-based

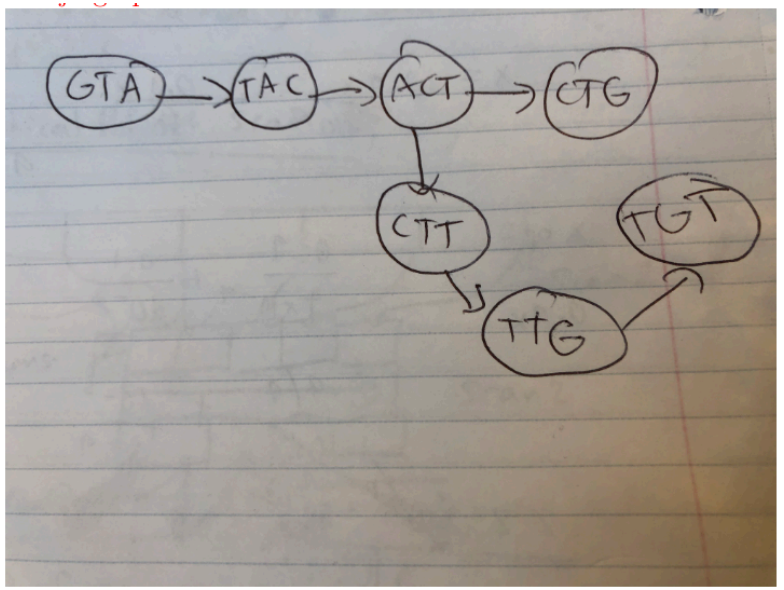
4. Hidden Markov Models

Genome Assembly

Homework 5 Solution Review

Clarification

- Theorem from class slides:
  - A note is semi-balanced if indegree differs from outdegree by 1.
  - A directed, connected graph is Eulerian if and only if it has at most 2 semi-balanced nodes and all other nodes are balanced.



- Here is the full theorem:

> A directed graph has an Eulerian trail if and only if at most one vertex has (out-degree) − (in-degree) = 1, at most one vertex has (in-degree) − (out-degree) = 1, every other vertex has equal in-degree and out-degree, and all of its vertices with nonzero degree belong to a single connected component of the underlying undirected graph.

RNA Secondary Structure Prediction

Things to Remember

- RNA can fold into structures due to nucleotide complementarity
  - A <--> U and G <--> C
- We can use dot-parenthesis format to represent secondary structures
- Find a pseudoknot-free secondary structure with the maximum number of complementary base pairings
  - Use dynamic programming with the Nussinov Algorithm
  - Recall that in homework we also looked at how to fill out a backtrace given an optimal solution.
    - Watch out for bifurcation when calling $\max_{i<k<j}\{s[i,k]+s[k+1,j]\}$
    - Watch out for more than one optimal solution

Nussinov Algorithm Example

	1 G	2 G	3 G	4 U	5 C	6 C	7 U	8 A	9 U	10 C		
G	0										G	1
G	0	0									G	2
G	0	0	0								G	3
U	0	0	0	0							U	4
C	0	0	0	0	0						C	5
C	0	0	0	0	0	0					C	6
U	0	0	0	0	0	0	0				U	7
A	0	0	0	0	0	0	0	0			A	8
U	0	0	0	0	0	0	0	0	0		U	9
C	0	0	0	0	0	0	0	0	0	0	C	10

Let $s[i,j]$ denote the maximum number of pseudoknot-free complementary base pairings in subsequence $v_i, \ldots, v_j$

$$\max \begin{cases} 0, & \text{if } i \geq j, \\ s[i+1, j-1] + 1, & \text{if } i < j \text{ and } (v_i, v_j) \in \Gamma, \quad (1) \\ s[i+1, j-1], & \text{if } i < j \text{ and } (v_i, v_j) \notin \Gamma, \quad (1^*) \\ s[i+1, j], & \text{if } i < j, \quad (2) \\ s[i, j-1], & \text{if } i < j, \quad (3) \\ \max_{i<k<j}\{s[i,k] + s[k+1,j]\}, & \text{if } i < j, \quad (4) \end{cases}$$

Nussinov Algorithm Example

	1 G	2 G	3 G	4 U	5 C	6 C	7 U	8 A	9 U	10 C	
G	0	0									G 1
G	0	0	0								G 2
G	0	0	0	0							G 3
U	0	0	0	0	0						U 4
C	0	0	0	0	0	0					C 5
C	0	0	0	0	0	0	0				C 6
U	0	0	0	0	0	0	0	1			U 7
A	0	0	0	0	0	0	0	0	1		A 8
U	0	0	0	0	0	0	0	0	0	0	U 9
C	0	0	0	0	0	0	0	0	0	0	C 10

(	)
7	8
U	A

Let $s[i,j]$ denote the maximum number of pseudoknot-free complementary base pairings in subsequence $v_i, \ldots, v_j$

$$\max \begin{cases} 0, & \text{if } i \geq j, \\ s[i+1, j-1]+1, & \text{if } i < j \text{ and } (v_i, v_j) \in \Gamma, \quad (1) \\ s[i+1, j-1], & \text{if } i < j \text{ and } (v_i, v_j) \notin \Gamma, \quad (1^*) \\ s[i+1, j], & \text{if } i < j, \quad (2) \\ s[i, j-1], & \text{if } i < j, \quad (3) \\ \max_{i<k<j}\{s[i,k]+s[k+1,j]\}, & \text{if } i < j, \quad (4) \end{cases}$$

Nussinov Algorithm Example

	1 G	2 G	3 G	4 U	5 C	6 C	7 U	8 A	9 U	10 C	
G	0	0	0								G 1
G	0	0	0	0							G 2
G	0	0	0	0	1						G 3
U	0	0	0	0	0	0					U 4
C	0	0	0	0	0	0	0				C 5
C	0	0	0	0	0	0	0	1			C 6
U	0	0	0	0	0	0	0	1	1		U 7
A	0	0	0	0	0	0	0	0	1	1	A 8
U	0	0	0	0	0	0	0	0	0	0	U 9
C	0	0	0	0	0	0	0	0	0	0	C 10

-	(	)
7	8	9
U	A	U

(	)	-
7	8	9
U	A	U

Let $s[i,j]$ denote the maximum number of pseudoknot-free complementary base pairings in subsequence $v_i, \ldots, v_j$

$$\max \begin{cases} 0, & \text{if } i \geq j, \\ s[i+1, j-1]+1, & \text{if } i < j \text{ and } (v_i, v_j) \in \Gamma, \quad (1) \\ s[i+1, j-1], & \text{if } i < j \text{ and } (v_i, v_j) \notin \Gamma, \quad (1^*) \\ s[i+1, j], & \text{if } i < j, \quad (2) \\ s[i, j-1], & \text{if } i < j, \quad (3) \\ \max_{i<k<j}\{s[i,k]+s[k+1,j]\}, & \text{if } i < j, \quad (4) \end{cases}$$

Nussinov Algorithm Example

(	-	)	-	(	)
3	4	5	6	7	8
G	**U**	**C**	**C**	**U**	**A**

	1	2	3	4	5	6	7	8	9	10	
	G	**G**	**G**	**U**	**C**	**C**	**U**	**A**	**U**	**C**	
G	0	0	0	0	1	2					**G** 1
G	0	0	0	0	1	2	2				**G** 2
G	0	0	0	0	1	1	1	2			**G** 3
U	0	0	0	0	0	0	0	1	1		**U** 4
C	0	0	0	0	0	0	0	1	1	1	**C** 5
C	0	0	0	0	0	0	0	1	1	1	**C** 6
U	0	0	0	0	0	0	0	1	1	1	**U** 7
A	0	0	0	0	0	0	0	0	1	1	**A** 8
U	0	0	0	0	0	0	0	0	0	0	**U** 9
C	0	0	0	0	0	0	0	0	0	0	**C** 10

Let $s[i,j]$ denote the maximum number of pseudoknot-free complementary base pairings in subsequence $v_i, \ldots, v_j$

$$\max \begin{cases} 0, & \text{if } i \geq j, \\ s[i+1, j-1]+1, & \text{if } i < j \text{ and } (v_i, v_j) \in \Gamma, \quad (1) \\ s[i+1, j-1], & \text{if } i < j \text{ and } (v_i, v_j) \notin \Gamma, \quad (1^*) \\ s[i+1, j], & \text{if } i < j, \quad (2) \\ s[i, j-1], & \text{if } i < j, \quad (3) \\ \max_{i<k<j}\{s[i,k]+s[k+1,j]\}, & \text{if } i < j, \quad (4) \end{cases}$$

Nussinov Algorithm Example

	(	(	(	-	)	)	-	(	)	)	
	1	2	3	4	5	6	7	8	9	10	
	G	G	G	U	C	C	U	A	U	C	
G	0	0	0	0	1	2	2	3	3	4	G 1
G	0	0	0	0	1	2	2	3	3	3	G 2
G	0	0	0	0	1	1	1	2	2	2	G 3
U	0	0	0	0	0	0	0	1	1	1	U 4
C	0	0	0	0	0	0	0	1	1	1	C 5
C	0	0	0	0	0	0	0	1	1	1	C 6
U	0	0	0	0	0	0	0	1	1	1	U 7
A	0	0	0	0	0	0	0	0	1	1	A 8
U	0	0	0	0	0	0	0	0	0	0	U 9
C	0	0	0	0	0	0	0	0	0	0	C 10

Let $s[i,j]$ denote the maximum number of pseudoknot-free complementary base pairings in subsequence $v_i, \ldots, v_j$

$$\max \begin{cases} 0, & \text{if } i \geq j, \\ s[i+1, j-1]+1, & \text{if } i < j \text{ and } (v_i, v_j) \in \Gamma, \quad (1) \\ s[i+1, j-1], & \text{if } i < j \text{ and } (v_i, v_j) \notin \Gamma, \quad (1^*) \\ s[i+1, j], & \text{if } i < j, \quad (2) \\ s[i, j-1], & \text{if } i < j, \quad (3) \\ \max_{i<k<j}\{s[i,k]+s[k+1,j]\}, & \text{if } i < j, \quad (4) \end{cases}$$

Phylogeny Inference

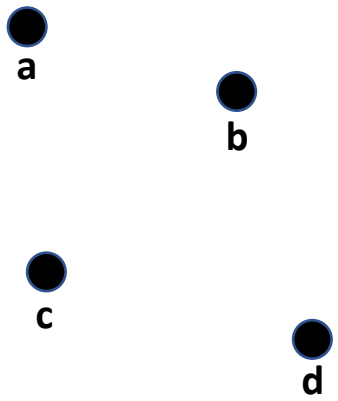
Distance-Based Methods

Things to Remember

- Tree topology represents similarity/distance between sequences
- Hierarchical clustering is one way to get tree from distance matrix
  - Greedy algorithm that merges closest clusters until just one cluster remains
  - The definition of distance between clusters, known as the linkage criterion, affects clustering result
    - Complete linkage – take the pairwise max
    - Single linkage – take the pairwise min
    - Mean linkage – take the pairwise average
  - Neighbor joining also hierarchical clustering. Intuitively, clusters that are more similar relative to their distance to other clusters are merged
    - Produces correct tree when matrix is additive

Hierarchical Clustering Single Linkage Example

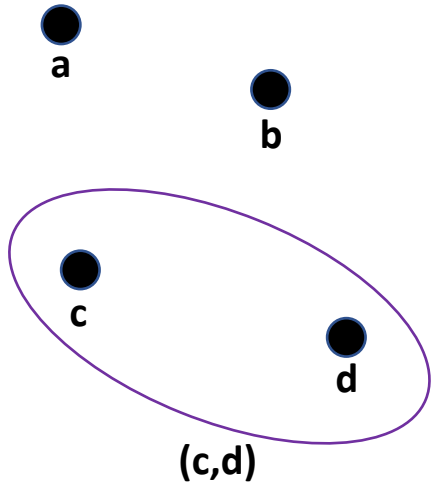
	a	b	c	d
a	0	6	7	8
b		0	11	12
c			0	5
d				0



Hierarchical Clustering Single Linkage Example

	a	b	c	d
a	0	6	7	8
b		0	11	12
c			0	5
d				0

	a	b	(c,d)
a	0	6	7
b		0	11
(c,d)			0

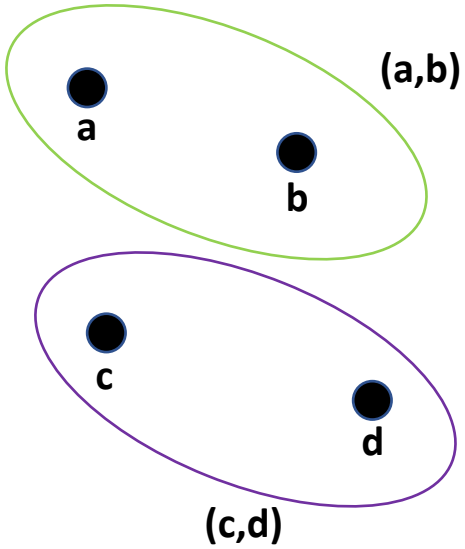


Hierarchical Clustering Single Linkage Example

	a	b	c	d
a	0	6	7	8
b		0	11	12
c			0	5
d				0

	a	b	(c,d)
a	0	6	7
b		0	11
(c,d)			0

	(a,b)	(c,d)
(a,b)	0	?
(c,d)		0

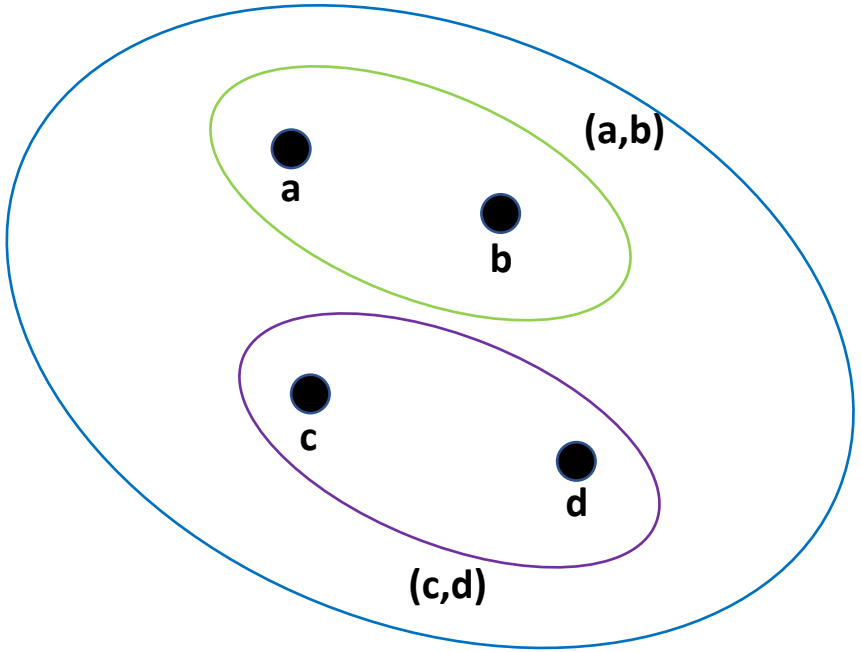


Hierarchical Clustering Single Linkage Example

	a	b	c	d
a	0	6	7	8
b		0	11	12
c			0	5
d				0

	a	b	(c,d)
a	0	6	7
b		0	11
(c,d)			0

	(a,b)	(c,d)
(a,b)	0	7
(c,d)		0



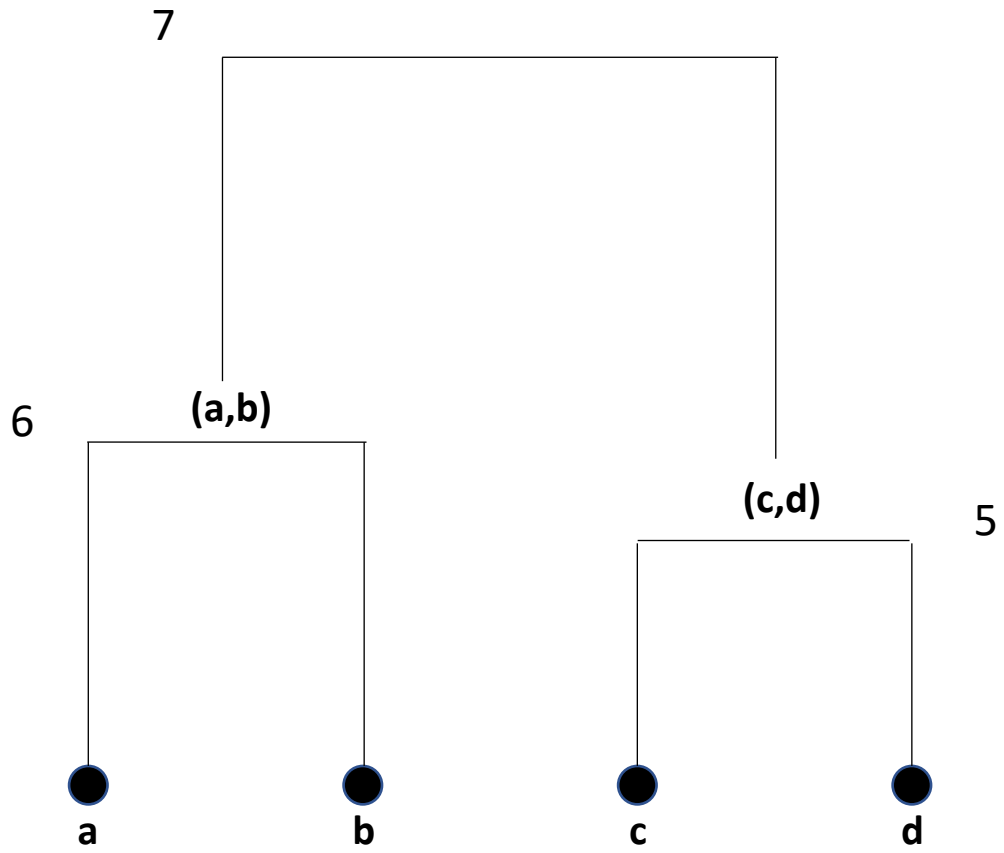
$$\min\{D(a,(c,d)), D(b,(c,d))\}$$

Hierarchical Clustering Single Linkage Example

	a	b	c	d
a	0	6	7	8
b		0	11	12
c			0	5
d				0

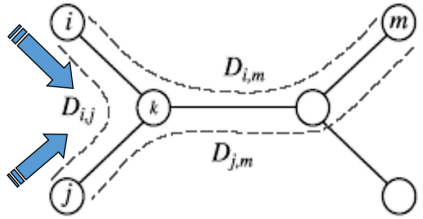
	a	b	(c,d)
a	0	6	7
b		0	11
(c,d)			0

	(a,b)	(c,d)
(a,b)	0	7
(c,d)		0



Things to Remember

- Tree topology represents similarity/distance between sequences
- We also saw how construct a tree from an additive matrix
  - Recall how to check if a matrix is additive
    - Check if distance matrix (non-negativity, zero diagonal, symmetry, triangle inequality)
    - Check Four Point Condition (for every set of four leaves, compute 3 sums, 2 should be equal, one should be less than or equal)
  - Small Additive Distance Problem: leaf-labeled tree + additive matrix



$$d_{k,m} = \frac{(d_{i,m}+d_{j,m}-d_{i,j})}{2}$$

  - Large Additive Distance Problem: additive matrix only
    - Find trimming parameter and remove degenerate triplet
    - Remember to decrease entries in matrix by two times the trimming parameter

Check if Matrix is Additive Example

	a	b	c	d
a	0	6	7	8
b	6	0	11	12
c	7	11	0	5
d	8	12	5	0

- Is it a distance matrix?
  - Non-negative ☑
  - Zero diagonal ☑
  - Symmetric ☑
  - Triangle inequality ?

Check if Matrix is Additive Example

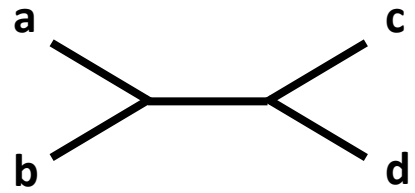
	a	b	c	d
a	0	6	7	8
b	6	0	11	12
c	7	11	0	5
d	8	12	5	0

- Is it a distance matrix?
  - Non-negative ☑
  - Zero diagonal ☑
  - Symmetric ☑
  - Triangle inequality ?
- Check Four Point Condition
  - $d_{ab} + d_{cd} = 6 + 5 = 11$
  - $d_{ac} + d_{bd} = 7 + 12 = 19$
  - $d_{ad} + d_{bc} = 8 + 11 = 19$

Check if Matrix is Additive Example

	a	b	c	d
a	0	6	7	8
b	6	0	11	12
c	7	11	0	5
d	8	12	5	0

- Is it a distance matrix?
  - Non-negative ☑
  - Zero diagonal ☑
  - Symmetric ☑
  - Triangle inequality ?
- Check Four Point Condition
  - $d_{ab} + d_{cd} = 6 + 5 = 11$
  - $d_{ac} + d_{bd} = 7 + 12 = 19$
  - $d_{ad} + d_{bc} = 8 + 11 = 19$
  - $11 = d_{ab} + d_{cd} \leq d_{ac} + d_{bd} = d_{ad} + d_{bc} = 19$ ☑



Phylogeny Inference

Character-Based Methods

Things to Remember

- Tree topology represents fewest state changes along edges
- Small Parsimony Problem: leaf-labeled tree + character matrix
  - Find labels of internal nodes in given tree maximizing parsimony
  - Recall that characters can be solved independently
  - Apply Sankoff dynamic programming algorithm to each character
    - Subproblem: minimum parsimony score of the subtree rooted at vertex v if v has character state t
- Large Parsimony Problem: character matrix only
  - In general, the problem is NP-hard (including multi-state perfect phylogeny)
  - Special case: two-state (i.e., binary) perfect (i.e., infinite sites) phylogeny
    - Check if conflict-free: no pair of columns contain the three pairs (0, 1), (1, 0) and (1, 1)
    - To reconstruct tree, sort columns and apply graph algorithm

Two-State Perfect Phylogeny Example

	c1	**c2**	**c3**	**c4**
a	0	0	1	0
b	1	0	0	0
c	0	1	1	0
d	0	1	1	1

Two-State Perfect Phylogeny Example

	c1	c2	c3	c4
a	0	0	1	0
b	1	0	0	0
c	0	1	1	0
d	0	1	1	1

Sort

|c1| = 1
|c2| = 2
|c3| = 3
|c4| = 1

	c3	c2	c1	c4
a	1	0	0	0
b	0	0	1	0
c	1	1	0	0
d	1	1	0	1

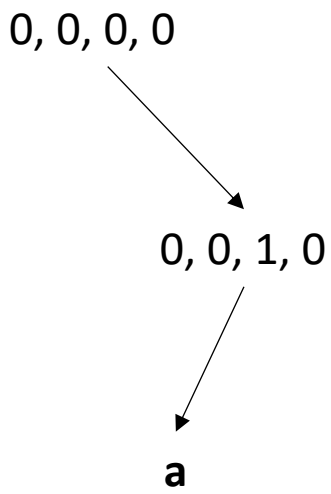
Two-State Perfect Phylogeny Example

	c1	c2	c3	c4
a	0	0	1	0
b	1	0	0	0
c	0	1	1	0
d	0	1	1	1

Sort

|c1| = 1
|c2| = 2
|c3| = 3
|c4| = 1

	c3	c2	c1	c4
a	1	0	0	0
b	0	0	1	0
c	1	1	0	0
d	1	1	0	1



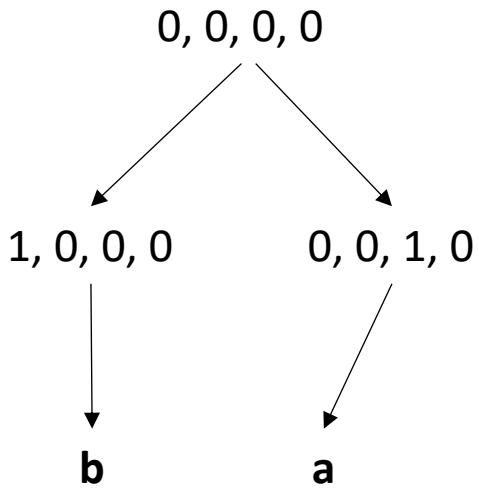
Two-State Perfect Phylogeny Example

	c1	c2	c3	c4
a	0	0	1	0
b	1	0	0	0
c	0	1	1	0
d	0	1	1	1

Sort

|c1| = 1
|c2| = 2
|c3| = 3
|c4| = 1

	c3	c2	c1	c4
a	1	0	0	0
b	0	0	1	0
c	1	1	0	0
d	1	1	0	1



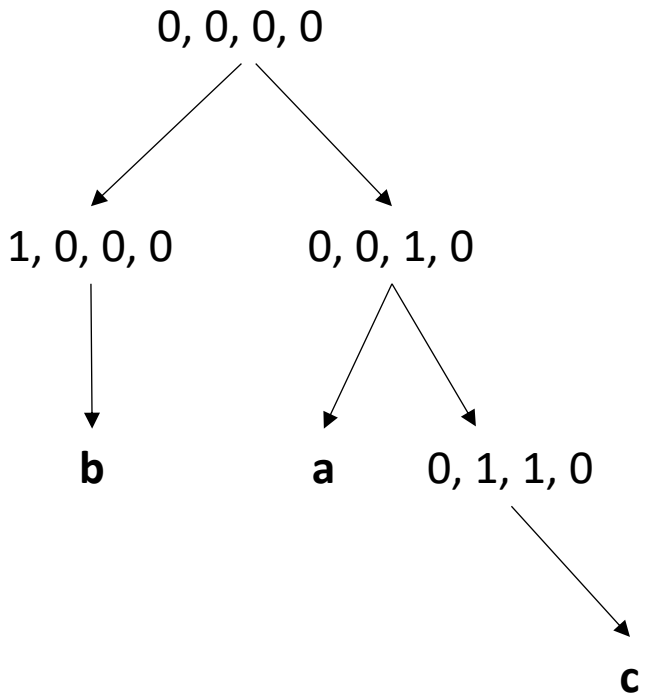
Two-State Perfect Phylogeny Example

	c1	c2	c3	c4
a	0	0	1	0
b	1	0	0	0
c	0	1	1	0
d	0	1	1	1

Sort

|c1| = 1
|c2| = 2
|c3| = 3
|c4| = 1

	c3	c2	c1	c4
a	1	0	0	0
b	0	0	1	0
c	1	1	0	0
d	1	1	0	1



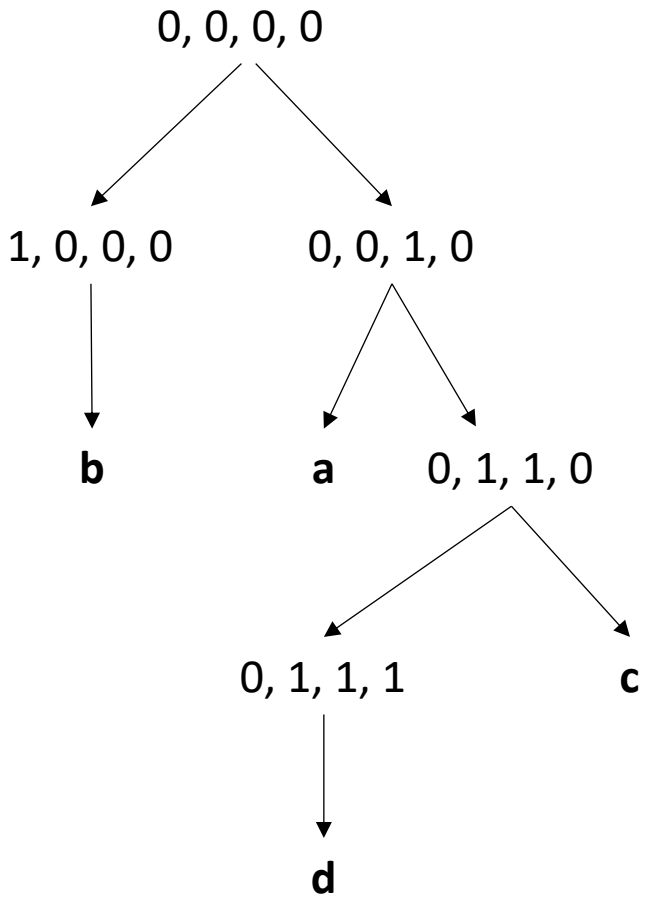
Two-State Perfect Phylogeny Example

	c1	c2	c3	c4
a	0	0	1	0
b	1	0	0	0
c	0	1	1	0
d	0	1	1	1

Sort

|c1| = 1
|c2| = 2
|c3| = 3
|c4| = 1

	c3	c2	c1	c4
a	1	0	0	0
b	0	0	1	0
c	1	1	0	0
d	1	1	0	1



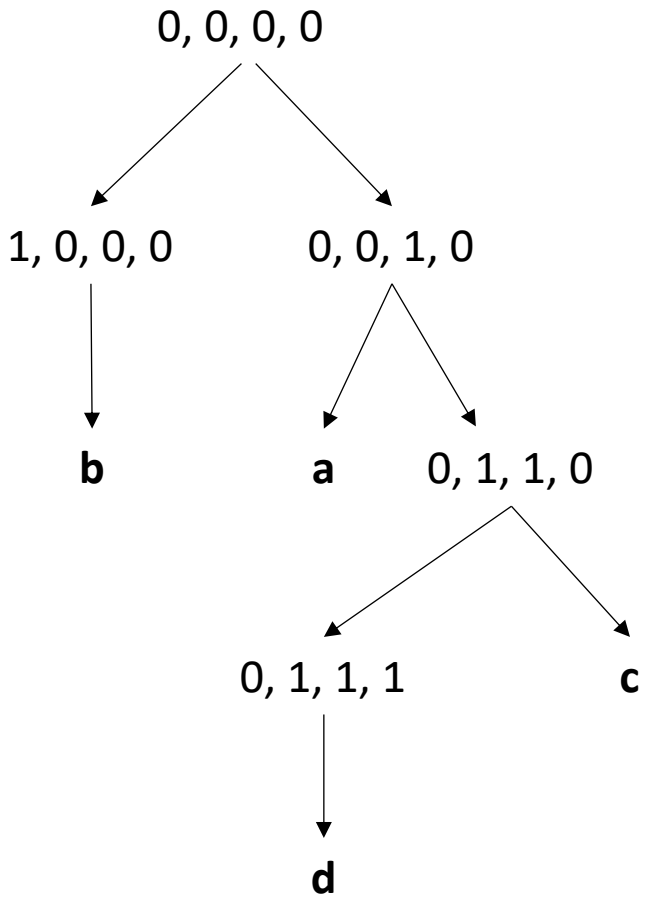
Two-State Perfect Phylogeny Example

	c1	c2	c3	c4
a	0	0	1	0
b	1	0	0	0
c	0	1	1	0
d	0	1	1	1

Sort

|c1| = 1
|c2| = 2
|c3| = 3
|c4| = 1

	c3	c2	c1	c4
a	1	0	0	0
b	0	0	1	0
c	1	1	0	0
d	1	1	0	1



Reminder – you can also check if the matrix is conflict free (i.e., no forbidden submatrix) to determine if it is a perfect phylogeny.

1	**1**
1	**0**
0	**1**

Hidden Markov Models

Things to Remember

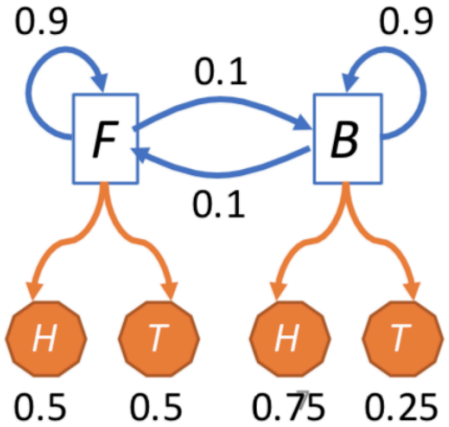
- The HMM setup requires the following
  - Set of hidden states Q
  - Transition probability matrix A
    - Future state only depends on current state
  - Set of emitted symbols Σ
  - Emission probability matrix E

$$Q = \{F, B\}$$

$$A = \begin{array}{c} \begin{matrix} F & B \end{matrix} \\ \begin{pmatrix} 0.9 & 0.1 \\ 0.1 & 0.9 \end{pmatrix} \begin{matrix} F \\ B \end{matrix} \end{array}$$

$$\Sigma = \{H, T\}$$

$$E = \begin{array}{c} \begin{matrix} H & T \end{matrix} \\ \begin{pmatrix} 0.5 & 0.5 \\ 0.75 & 0.25 \end{pmatrix} \begin{matrix} F \\ B \end{matrix} \end{array}$$



- We observe the emitted symbols but not the hidden states

Things to Remember

- We considered three questions about HMM systems
  - What is the most probable path π^* that generated observations $\boldsymbol{x}$?
    - Viterbi algorithm
  - What is probability of observations $\boldsymbol{x}$ generated by any path π?
    - Forward algorithm
  - What is the probability of observation x_i generated by state s?
    - Forward and backward algorithm

- Recall that we often do computation in logspace to avoid underflow

Forward Algorithm Example

What is the probability of observing **x** = [H, T] ?

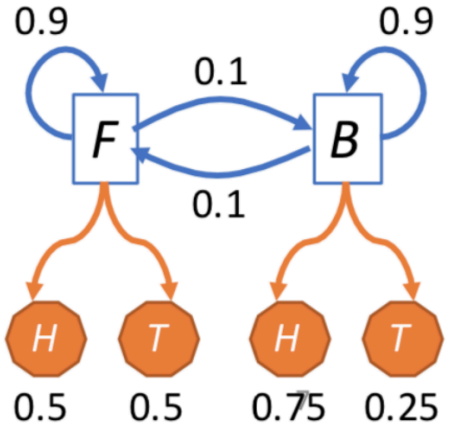
	H	T
F		
B		

$$Q = \{F, B\}$$

$$A = \begin{matrix} & \begin{matrix} F & B \end{matrix} \\ \begin{matrix} F \\ B \end{matrix} & \begin{pmatrix} 0.9 & 0.1 \\ 0.1 & 0.9 \end{pmatrix} \end{matrix}$$

$$\Sigma = \{H, T\}$$

$$E = \begin{matrix} & \begin{matrix} H & T \end{matrix} \\ \begin{matrix} F \\ B \end{matrix} & \begin{pmatrix} 0.5 & 0.5 \\ 0.75 & 0.25 \end{pmatrix} \end{matrix}$$



$$f[s,i] = \begin{cases} a_{0,s} e_{s,x_1}, & \text{if } i = 1, \\ e_{s,x_i} \sum_{t \in Q} \{ f[t, i-1] \cdot a_{t,s} \}, & \text{if } i > 1. \end{cases}$$

Forward Algorithm Example

What is the probability of observing **x** = [H, T] ?

	H	T
F	.25	
B	.375	

$$f[F,1] = a_{0,F} \cdot e_{F,H} = \frac{1}{2} \cdot \frac{1}{2} = .25$$

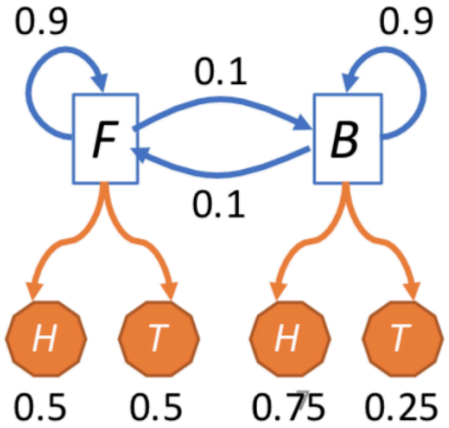
$$f[B,1] = a_{0,B} \cdot e_{B,H} = \frac{1}{2} \cdot \frac{3}{4} = .375$$

$$Q = \{F, B\}$$

$$A = \begin{matrix} & \begin{matrix} F & B \end{matrix} \\ \begin{matrix} F \\ B \end{matrix} & \begin{pmatrix} 0.9 & 0.1 \\ 0.1 & 0.9 \end{pmatrix} \end{matrix}$$

$$\Sigma = \{H, T\}$$

$$E = \begin{matrix} & \begin{matrix} H & T \end{matrix} \\ \begin{matrix} F \\ B \end{matrix} & \begin{pmatrix} 0.5 & 0.5 \\ 0.75 & 0.25 \end{pmatrix} \end{matrix}$$



$$f[s,i] = \begin{cases} a_{0,s} e_{s,x_1}, & \text{if } i = 1, \\ e_{s,x_i} \sum_{t \in Q} \{f[t, i-1] \cdot a_{t,s}\}, & \text{if } i > 1. \end{cases}$$

Forward Algorithm Example

What is the probability of observing **x** = [H, T] ?

	H	T
F	.25	.13
B	.375	.09

$$f[F,2] = e_{F,T} \sum_{t \in Q} f[t,1] \cdot a_{t,F}$$

$$= \frac{1}{2} \cdot \big((.25 \cdot .9) + (.375 \cdot .1)\big) \approx 0.13$$

$$f[B,2] = e_{B,T} \sum_{t \in Q} f[t,1] \cdot a_{t,B}$$

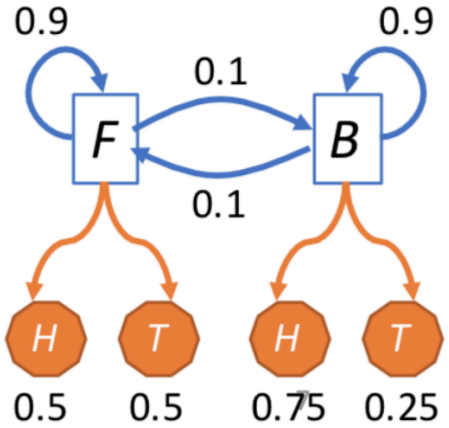
$$= \frac{1}{4} \cdot ((.25 \cdot .1) + (.375 \cdot .9)) \approx 0.09$$

$$Q = \{F, B\}$$

$$A = \begin{array}{c} \\ F \\ B \end{array} \begin{array}{c} \begin{array}{cc} F & B \end{array} \\ \begin{pmatrix} 0.9 & 0.1 \\ 0.1 & 0.9 \end{pmatrix} \end{array}$$

$$\Sigma = \{H, T\}$$

$$E = \begin{array}{c} \\ F \\ B \end{array} \begin{array}{c} \begin{array}{cc} H & T \end{array} \\ \begin{pmatrix} 0.5 & 0.5 \\ 0.75 & 0.25 \end{pmatrix} \end{array}$$



$$f[s,i] = \begin{cases} a_{0,s} e_{s,x_1}, & \text{if } i = 1, \\ e_{s,x_i} \sum_{t \in Q} \{f[t, i-1] \cdot a_{t,s}\}, & \text{if } i > 1. \end{cases}$$

Forward Algorithm Example

What is the probability of observing **x** = [H, T] ?

	H	T
F	.25	.13
B	.375	.09

We **sum the final column** to find the total probability

$.13 + .09 = .22$

$$f[F,2] = e_{F,T} \sum_{t \in Q} f[t,1] \cdot a_{t,F}$$

$$= \frac{1}{2} \cdot \big((.25 \cdot .9) + (.375 \cdot .1)\big) \approx 0.13$$

$$f[B,2] = e_{B,T} \sum_{t \in Q} f[t,1] \cdot a_{t,B}$$

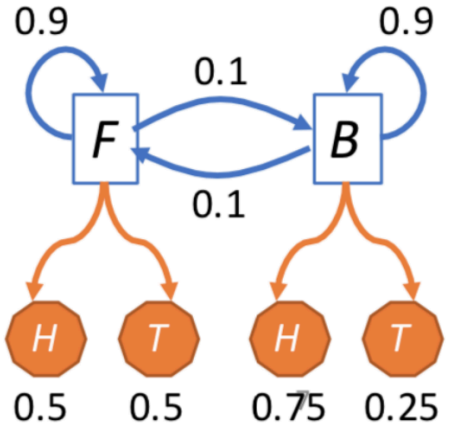
$$= \frac{1}{4} \cdot ((.25 \cdot .1) + (.375 \cdot .9)) \approx 0.09$$

$$Q = \{F, B\}$$

$$A = \begin{array}{c} \\ F \\ B \end{array} \begin{array}{c} \begin{array}{cc} F & B \end{array} \\ \begin{pmatrix} 0.9 & 0.1 \\ 0.1 & 0.9 \end{pmatrix} \end{array}$$

$$\Sigma = \{H, T\}$$

$$E = \begin{array}{c} \\ F \\ B \end{array} \begin{array}{c} \begin{array}{cc} H & T \end{array} \\ \begin{pmatrix} 0.5 & 0.5 \\ 0.75 & 0.25 \end{pmatrix} \end{array}$$



$$f[s,i] = \begin{cases} a_{0,s} e_{s,x_1}, & \text{if } i = 1, \\ e_{s,x_i} \sum_{t \in Q} \{f[t, i-1] \cdot a_{t,s}\}, & \text{if } i > 1. \end{cases}$$

Forward Algorithm Example

What is the probability of observing **x** = [H, T] ?

	H	T
F	.25	.13
B	.375	.09

We **sum the final column** to find the total probability

$.13 + .09 = .22$

$$f[F,2] = e_{F,T} \sum_{t \in Q} f[t,1] \cdot a_{t,F}$$

$$= \frac{1}{2} \cdot \big((.25 \cdot .9) + (.375 \cdot .1)\big) \approx 0.13$$

$$f[B,2] = e_{B,T} \sum_{t \in Q} f[t,1] \cdot a_{t,B}$$

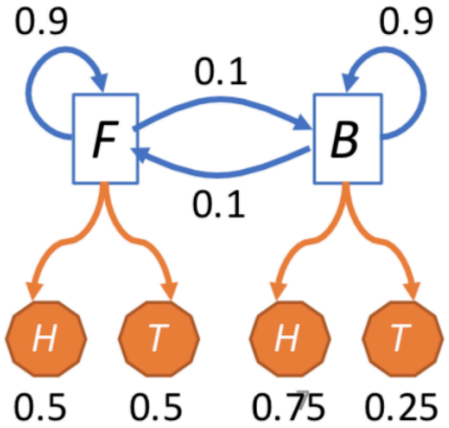
$$= \frac{1}{4} \cdot ((.25 \cdot .1) + (.375 \cdot .9)) \approx 0.09$$

$$Q = \{F, B\}$$

$$A = \begin{matrix} & \begin{matrix} F & B \end{matrix} \\ \begin{matrix} F \\ B \end{matrix} & \begin{pmatrix} 0.9 & 0.1 \\ 0.1 & 0.9 \end{pmatrix} \end{matrix}$$

$$\Sigma = \{H, T\}$$

$$E = \begin{matrix} & \begin{matrix} H & T \end{matrix} \\ \begin{matrix} F \\ B \end{matrix} & \begin{pmatrix} 0.5 & 0.5 \\ 0.75 & 0.25 \end{pmatrix} \end{matrix}$$



$$f[s,i] = \begin{cases} a_{0,s} e_{s,x_1}, & \text{if } i = 1, \\ e_{s,x_i} \sum_{t \in Q} \{f[t, i-1] \cdot a_{t,s}\}, & \text{if } i > 1. \end{cases}$$

$$\log_2 f[s,i] = \begin{cases} \log_2 a_{0,s} + \log_2 e_{s,x_1} \\ \log_2 e_{s,x_i} + \log_2 \sum_{t \in Q} 2^{\log_2 f[t,i-1]} \cdot 2^{\log_2 a_{t,s}} \end{cases}$$

Exam Released Tonight at 7pm

Good luck!!