Comp 555 Problem Set 4

Elliott Hauser

December 6, 2012

Question 1

1. Consider the following four sequences Chimp GTTG Gorilla GTCA Human ACCA Orangutan ATTA Assume the following scoring matrix A T G C A 0 2 3 8 T 2 0 1 3 G 3 1 0 3 C 8 3 3 0

a. Six binary tree topologies with two different tree structures are shown in Figure 1.

b. b. Apply Sankoffs algorithm to the topologies created in (a) to determine the weighted parsimony score. What is the minimum parsimony score? What is the most plausible topology of the six you created? Why?

c. c. Apply Fitchs algorithm to the same topologies. Find the sequences for the roots for all. Which is the one with the least parsimony score.

d. d. Do b) and c) give same results. Why or Why not?

Question 2

2. Consider the following distance matrix A B C D E F A 0 18 15 21 6 16 B 0 23 19 20 24 C 0 26 17 19 D 0 23 27 E 0 18 F 0

1,1,2,1,2,1

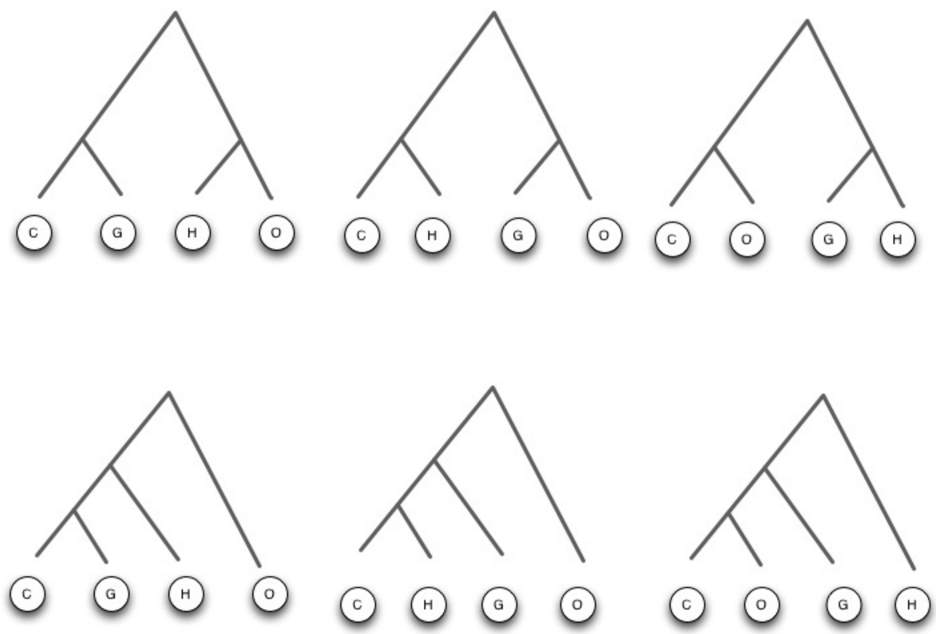

Figure 1: Six binary tree topologies, utilizing two different tree structures.

a. A matrix is additive if there exists a tree T with $d_{i,j}(T) = D_{i,j}$. But an efficient way to determine this is the Four point theorem, which states that, for every combination of leaves i, j, k, and l, the sums $D_{i,j}, D_{j,k}$, and $D_{k,l}$ yeild two identical sums, with the third sum smaller than these two.

To test this, I wrote a simple algorithm in Python. It is shown in Figure 2. It confirms that every combinaion of i, j, k, and l satisfy the four point condition; the matrix is thus additive.

b. An efficient method for finding the δ for an interation is given by the pseudocode in Figure 3

c. Figure 4 shows the δ and i, j, k for the first two iterations on this matrix.

```
def four_point():
# Determines whether a matrix is additive by testing for the four point
    condidtion.  Expects matrix as a list of lists

    n = len(matrix)

    for i in xrange(0,n):
        for j in xrange(0,n):
            if i == j:
                continue
            for k in xrange(0,n):
                if k == j:
                    continue
                if k == i:
                    continue
                for l in xrange(0,n):
                    if l == k:
                        continue
                    if l == j:
                        continue
                    if l == i:
                        continue
                    one = matrix[i][j] + matrix[k][l]
                    two = matrix[i][k] + matrix[j][l]
                    three = matrix[i][l] + matrix[j][k]
                    lump = sorted([one,two,three])
                    if lump[1] != lump[2]:
                        print "Not additive"
```

Figure 2: An algorithm to calculate whether the four point condition holds for an $n \times n$ matrix.

```
FindDelta(matrix)
    select any cell comtaining the minimum weight
    combine the leaves indicated (column, row) with another leaf
    connect all three leaves to a central node by edges x,y,z
    solve the following system:
        x + y = D(the nodes they connect)
        y + z = D(the nodes they connect)
        z + x = D(the nodes they connect)
    the minium of x,y,z is the minium value of δ
```

Figure 3: An algorithm to to find the trimming parameter for additive tree reconstruction.

	A	B	C	D	E	F
A	0	18	15	21	6	16
B	18	0	23	19	20	24
C	15	23	0	26	17	19
D	21	19	26	0	23	27
E	6	20	17	23	0	18
F	16	24	19	27	18	0

||

V ∂=2

	A	B	C	D	**E**	F	
A	0	16	13	19	**4**	14	
B	16	0	21	17	**18**	22	
C	13	21	0	24	**15**	17	I <- A
D	19	17	24	0	**21**	25	J <- E
E	**4**	**18**	**15**	**21**	**0**	**16**	K <- F
F	14	22	17	25	**16**	0	

||

V

	A	B	C	D	F
A	0	16	13	19	14
B	16	0	21	17	22
C	13	21	0	24	17
D	19	17	24	0	25
F	14	22	17	25	0

||

V ∂=5

	A	B	C	D	F	
A	**0**	**6**	**3**	**9**	**4**	
B	**6**	0	11	7	12	I <- C
C	**3**	11	0	14	7	J <- A
D	**9**	7	14	0	15	K <- F
F	**4**	12	7	15	0	

\\

V

	B	C	D	F
B	0	11	7	12
C	11	0	14	7
D	7	14	0	15
F	12	7	15	0

Figure 4: THe first two iterations of additive phylogenetic reconstruction.

Question 3

For this problem let us define i as the number of islands connected in a graph of arbitrary vertices. In this problem, $i = 8$. Figure 5 shows an initial formulation of the boundaries of the problem of visiting each island. On the left, the minumum possible trips are shown, which are $i - 1 = 7$. On the right is an arrangement where the maximum number of trips must be taken. Because of the inclusion of a central hub, many routes must be taken more than once. In fact, were we not able to choose our starting island in this problem, this example would disprove the desired limit of 12 since the trip length is $((i - 2) * 2) - 1 = 13$.

But the problem places no such restriction on us, and so the minumum number of trips in the pathological case is show in Figure 6, and is $((i - 1) * 2) - 2 = 12$. How can we prove that this is the maximum number of trips? The graph shown on the right in Figure 5 and in Figure 6 is pathological because there are the maximum amount of nodes connected by only one edge, 7, which is $i - 1$, i.e. all nodes except the central node. Each of these potentially requires two trips to visit and then keep visiting other islands, yielding $(i - 1) * 2 = 14$ But we know that all of the islands are visited before each vertex is traveled twice. In fact, if one both starts and ends on one of the outlying islands as int he irght of Figure 6, two of these return trips can be eliminated, one each for the Start and End islands, giving us the answer of $((i - 1) * 2) - 2 = 12$. Starting on the central island, as in the right of Figure 5 eliminates only one of these return trips, yielding $((i - 1) * 2) - 1 = 13$.

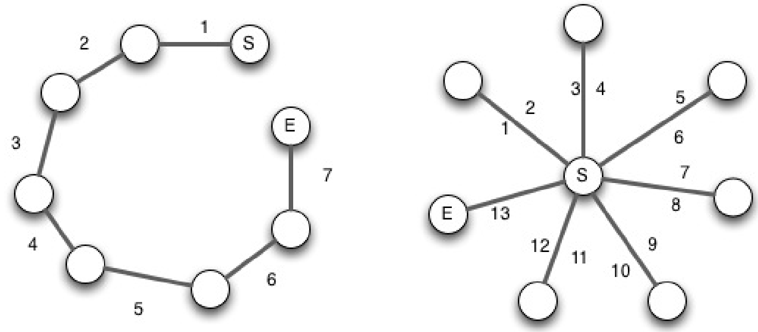

Figure 5: Initial formulation of the problem. The tree on the left shows the minimum trips possible to visit all islands (i, while the tree on the right shows the pathological case where the maximum amount of trips must be taken if the starting islands cannot be chosen. Start and end positions are marked by S and E, respectively, and trips are numbered.

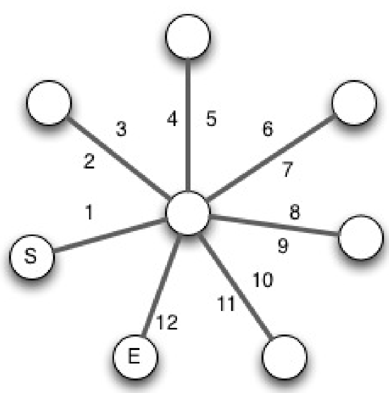

Figure 6: The shortest path visiting all islands in the pathological case, where the starting islands are able to be chosen. Its length is $(i-1)*2-2=12$

	1	1	2	1	2	2	End (p=.25)
D1	**0.250000**	**0.062500**	**0.007813**	**<--0.001953125**	0.000244	0.000031	0.000008
D2	0.125000	0.015625	**0.007813**	**<--0.000976562**	**0.000244**	**0.000061**	**0.000015**
Vitterbi							

Table 1: The Viterbi table for problem 4. The arrows are implicit for most of the table, pointing towards cells in bold; in the two ambiguous cases arrows are indicated, showing two possible paths.

Question 4

a The hidden markov model for this problem is in Figure 7

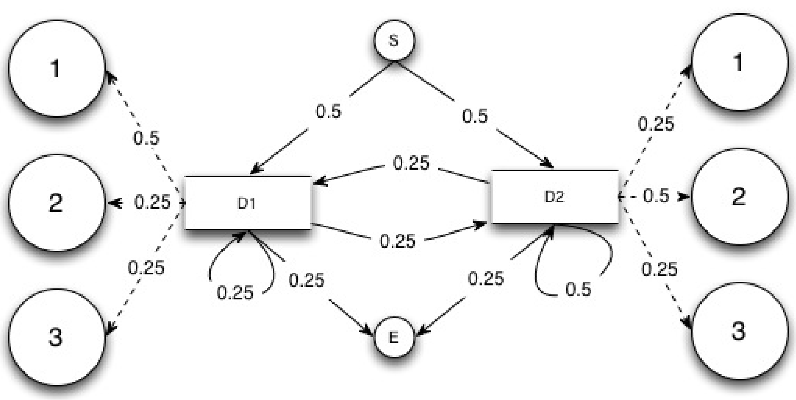

Figure 7: A hidden markov model of $Q = start, D1, D2, end$ and $\Sigma = 1, 2, 3$

b Table 1 shows a Viterbi table for this sequence. A sequnce that would satisfy this table is

$$D_1, D_1, D_1, D_1, D_2, D_2, End$$

The probability of this particular sequence

c The second sequence of states implied by Table 1 is

$$D_1, D_1, D_2, D_2, D_2, D_2, End$$

The probability of this sequence occuring, independent of the emisions above, is given by

$$\begin{array}{ccccccccccccccc} \text{start} & & D_1 & & D_1 & & D_2 & & D_2 & & D_2 & & D_2 & & \text{end} \\ & .5 & \cdot & .5 & \cdot & .5 & \cdot & .25 & \cdot & .5 & \cdot & .5 & \cdot & .5 & \cdot & .25 & = 0.0009765 \end{array}$$

Question 5

The probability that state k caused emission x at moment i is given by

$$P(\pi_i = k|x) = \frac{f_k(i) \cdot b_k(i)}{P(x|\pi)}$$

Also,

$$f_k(i) = e_k(x_i) \cdot \sum_{l \in Q} f_{l,i-1} \cdot a_{l,k}$$

and

$$b_k(i) = e_k(x_i) \cdot \prod_{l \in Q} f_{l,i-1} \cdot a_{l,k}$$

Finally,

$$P(x) = \sum_{\pi} P(x|\pi)$$

I calculate these three terms, respectively, as

$$\begin{aligned} f_{D_1}(x_4) =& 0.005127 \\ b_{D_1}(x_4) =& 0.017578 \\ P(x) =& 0.000159 \end{aligned}$$

This yields

$$P(\pi_i = k|x) = \frac{0.005127 \cdot 0.017578}{0.000159} = 0.566807$$

This means that there's greater than 56% chance that the dealer rolled the third 1 from D_1. This makes intuitive sense: there are only two choices at that point, and the calculations I have slightly favor D_1 at that roll. The tables I used for these calculations are shown in Table 2

	1	1	2	1	2	2	End
D1	0.250000	0.078125	0.011719	0.005127	0.000824	0.000189	0.000047
D2	0.125000	0.031250	0.017578	0.002930	0.001373	0.000446	0.000112
Forward (D1, x=4)		0.005127					
	1	1	2	1	2	2	End
D1	**0.250000**	**0.078125**	**0.011719**	**1.000000**	**0.125000**	**0.023438**	0.005859
D2	0.125000	0.031250	0.017578	0.000000	0.125000	0.046875	**0.011719**
Backwards (5->End):		0.017578					
Sum of all Paths:		0.000159					

Table 2: Supporting calculations for $f_{D_1}(x_4), b_{D_1}(x_4)$, and $P(x)$