MACHINE LEARNING COMS 4771, HOMEWORK 5

UNI: yb2356 Name: Yang Bai

1 Probability

I should change.

Define the following random variables:

- Si : Si=1 if I select door i; Si= 0 otherwise.
- Oi : Oi=1 if the host opens door i; Oi= 0 otherwise.
- Ci : Ci=1 if the car is behind door i; Ci=0 otherwise.

Since the order of the door doesn't matter, we can assume that she initially picks door 1 and the host opens door 3 (any orders will work in the same way). We need to compute and compare the posteriors:

$p(C_1 = 1|S_1 = 1, O_3 = 1)$ and $p(C_2 = 1|S_1 = 1, O_3 = 1)$.

Using Bayes' rule:

$$p(C_1 = 1|S_1 = 1, O_3 = 1) = \frac{p(S_1 = 1, O_3 = 1|C_1 = 1)p(C_1 = 1)}{\sum_{j\in\{0,1\}} p(S_1 = 1, O_3 = 1|C_1 = j)p(C_1 = j)}$$

$$= \frac{p(O_3 = 1|S_1 = 1, C_1 = 1)p(S_1 = 1|C_1 = 1)p(C_1 = 1)}{\sum_{j\in\{0,1\}} p(O_3 = 1|S_1 = 1, C_1 = j)p(S_1 = 1|C_1 = j)p(C_1 = j)}$$

$$= \frac{\frac{1}{2}\times\frac{1}{3}\times\frac{1}{3}}{\frac{1}{2}\times\frac{1}{3}\times\frac{1}{3} + 1\times\frac{1}{3}\times\frac{1}{3}} = \frac{1}{3}$$

$$p(C_2 = 1|S_1 = 1, O_3 = 1)$$

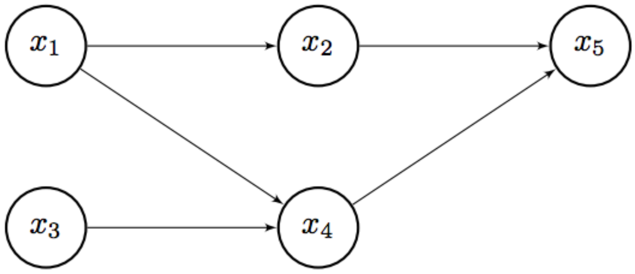
$$= \frac{p(O_3 = 1|S_1 = 1, C_2 = 1)p(S_1 = 1|C_2 = 1)p(C_2 = 1)}{\sum_{j\in\{0,1\}} p(O_3 = 1|S_1 = 1, C_2 = j)p(S_1 = 1|C_2 = j)p(C_2 = j)}$$

$$= \frac{\frac{1}{3}\times\frac{1}{3}}{\frac{1}{3}\times\frac{1}{3} + \frac{1}{2}\times\frac{1}{3}\times\frac{1}{3}} = \frac{2}{3}$$

$$p(C_2 = 1|S_1 = 1, O_3 = 1) > p(C_1 = 1|S_1 = 1, O_3 = 1)$$

So I should change my idea to get a higher probability to win the car.

2 Bayesian Network Conditional Independence



$$p(x_1, \dots, x_5) = p(x_1)p(x_2|x_1)p(x_3)p(x_4|x_3, x_1)p(x_5|x_2, x_4)$$

1. false
   there is a path: x2 -- x1-- x4
2. false
   there is a path: x2 -- x5 --x4
3. true

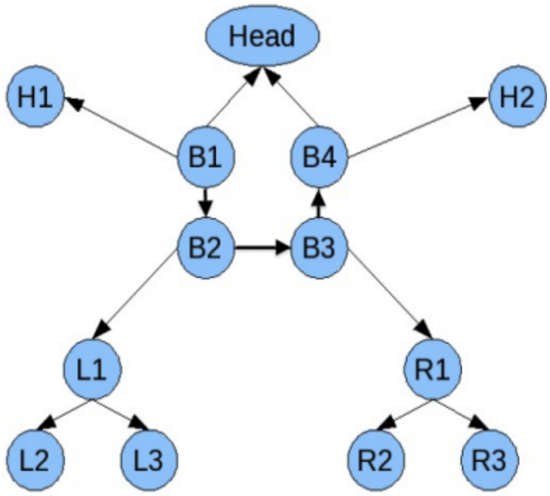
4. false
   there is a path: x3 – x4-- x1-- x2 – x5
5. true

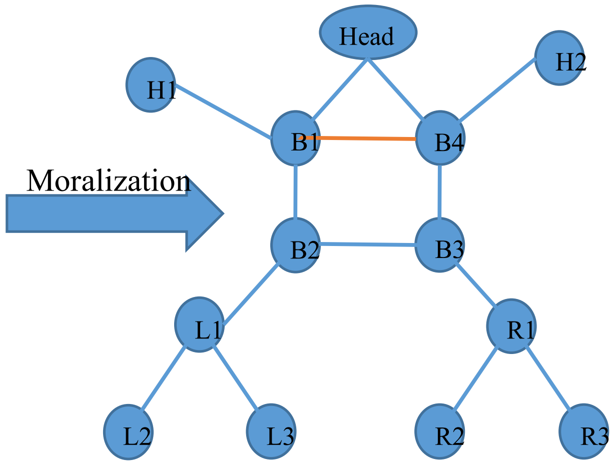
6. false
   there is a path: x1 – x2 – x5 – x4—x3
7. true

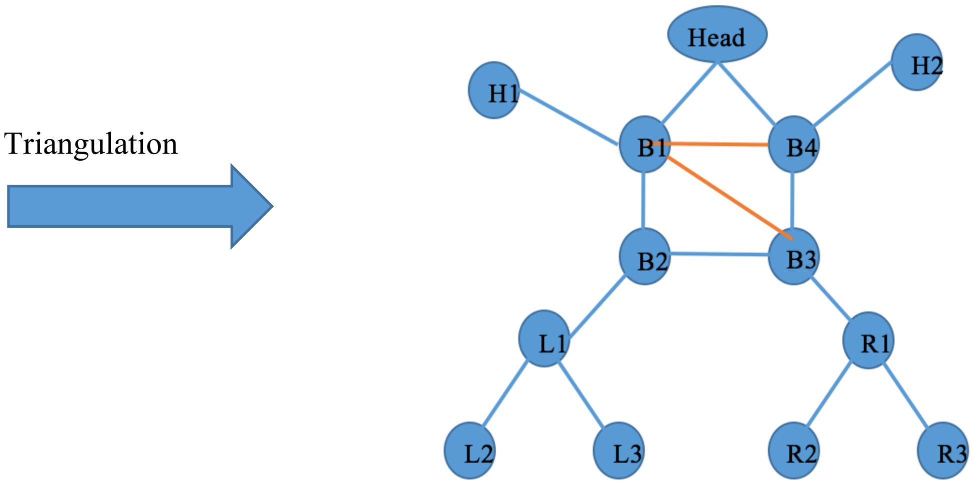
8. true

9. false
   there is a path: x3 – x4 – x5 –x2
10. false
    there is a path: x3 – x4 – x1 – x2

3. Junction Tree Construction



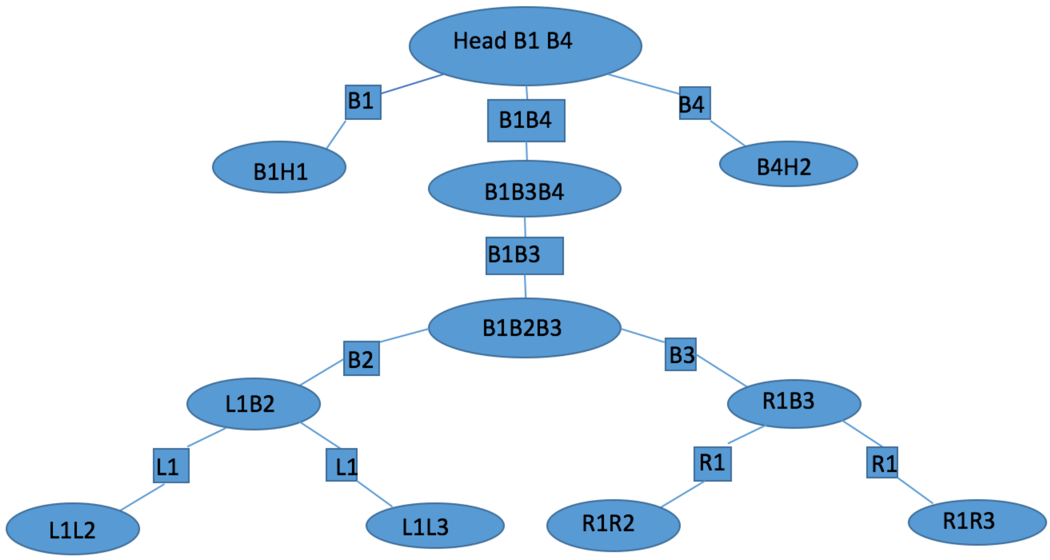




cliques:

HeadB1B4, B1B2B3, B1B3B4, H1B1, H2B4, L1B2, R1B3, L1L2, L1L3, R1R2, R2R3

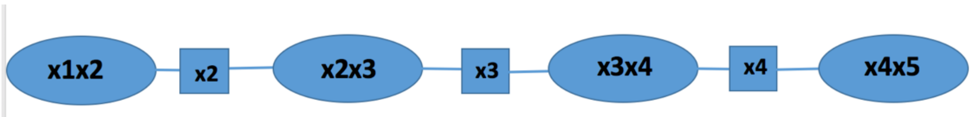
Junction Tree:



4 Junction Tree Algorithm

Show code, results, and discussion.

Building junction tree:



We need to perform left to right message passing and then right to left message passing by using a for loop or standard iteration.

Code:

```
n=5;
%initialize
psis = cell(n-1,1);
separator = cell(n-2,1);

for i=1:n-2
separator{i} = [1,1];
end

for i = 1:(n-1)
psis{i} = rand(2,2);
end
```

```
% left to right message passing
for t = 1:n-2
    temp_s = separator{t} ;
    separator{t} = sum( psis{t},1 );
    psis{t+1} =repmat( (separator{t} ./temp_s)',1,2) .* psis{t+1};
end

 % then right to left message passing
for k = n-2:-1:1
    temp_s = separator{k} ;
    separator{k} = sum( psis{k+1}' );
    psis{k} = repmat(separator{k} ./temp_s,2,1) .* psis{k};
end

% normalize
for i=1:n-2
    separator{i} = separator{i}./sum(separator{i});
end

for i=1:n-1
    psis{i} = psis{i} ./ sum(sum(psis{i})) ;
end
```

results:

Using the given potential function, we come to the result below:

$$\psi(x_1, x_2) = \left\{ \begin{array}{c|cc} & x_2 = 0 & x_2 = 1 \\ \hline x_1 = 0 & 0.0405, & 0.4451 \\ x_1 = 1 & 0.3237, & 0.1907 \end{array} \right\}$$

$$\psi(x_2, x_3) = \left\{ \begin{array}{c|cc} & x_3 = 0 & x_3 = 1 \\ \hline x_2 = 0 & 0.2601, & 0.1041 \\ x_2 = 1 & 0.0578, & 0.5780 \end{array} \right\}$$

$$\psi(x_3, x_4) = \left\{ \begin{array}{c|cc} & x_4 = 0 & x_4 = 1 \\ \hline x_3 = 0 & 0.1192, & 0.1987 \\ x_3 = 1 & 0.6395, & 0.0426 \end{array} \right\}$$

$$\psi(x_4, x_5) = \left\{ \begin{array}{c|cc} & x_5 = 0 & x_5 = 1 \\ \hline x_4 = 0 & 0.5690, & 0.1897 \\ x_4 = 1 & 0.0603, & 0.1810 \end{array} \right\}$$

$$\phi(x_2) = \begin{Bmatrix} x_2 = 0 & x_2 = 1 \\ 0.3642 & 0.6358 \end{Bmatrix} \qquad \phi(x_2) = \sum_{x_1} \psi(x_1, x_2) = \sum_{x_3} \psi(x_2, x_3)$$

$$\phi(x_3) = \begin{Bmatrix} x_3 = 0 & x_3 = 1 \\ 0.3179 & 0.6821 \end{Bmatrix} \qquad \phi(x_3) = \sum_{x_2} \psi(x_2, x_3) = \sum_{x_4} \psi(x_3, x_4)$$

$$\phi(x_4) = \begin{Bmatrix} x_2 = 0 & x_2 = 1 \\ 0.7587 & 0.2413 \end{Bmatrix} \qquad \phi(x_4) = \sum_{x_3} \psi(x_3, x_4) = \sum_{x_5} \psi(x_4, x_5)$$

5 Hidden Markov Model

Let $q_t = 1$ when the state is "happy"; Let $q_t = 2$ when the state is "angry";
The state transition matrix:

$$p(q_t|q_{t-1}) = \begin{array}{cc} \begin{bmatrix} 0.8, & 0.2 \\ 0.2, & 0.8 \end{bmatrix} & \begin{array}{l} q_{t-1} = 1 \\ q_{t-1} = 2 \end{array} \\ q_t = 1, q_t = 2 & \end{array}$$

the emission probability matrix:

	smile	frown	laugh	yell
Happy	0.4	0.1	0.3	0.2
Angry	0.1	0.4	0.2	0.3

Let $O_t = 1$ when observation is "smile"; Let $O_t = 2$ when observation is "frown";
Let $O_t = 3$ when observation is "laugh"; Let $O_t = 4$ when observation is "yell";

$$p(O_t|q_t) = \begin{array}{cc} \begin{bmatrix} 0.4, & 0.1, & 0.3, & 0.2 \\ 0.1, & 0.4, & 0.2, & 0.3 \end{bmatrix} & \begin{array}{l} q_t = 1 \\ q_t = 2 \end{array} \\ O_t{:}\ 1, \quad 2, \quad 3, \quad 4 & \end{array}$$

Here is the Viterbi algorithm for calculating most possible path in HMM.
Let vi(t) be the probability of the most probable path ending in state i at time t.

$$v_j(t) = \max P(q_1 \ldots q_{t-1} q_t = j, O_1 \ldots O_t | H)$$

H is the condition. $H = (p(q_t = j | q_{t-1} = i), p(O_t = a | q_t = j),\ p(q_1 = i))$
Then vj (t) can be calculated recursively using

$$v_j(t) = \max[v_i(t-1) p(q_t = j | q_{t-1} = i)] * p(O_t = a | q_t = j)$$

Initial condition:

$$v_i(1) = p(q_1 = i) p(O_1 = a | q_1 = i)$$

In this problem,

$$v_1(1) = p(q_1 = happy) p(O_1 = smile | q_1 = happy) = 1 \times 0.4 = 0.4$$

$$v_2(1) = p(q_1 = angry) p(O_1 = smile | q_1 = angry) = 0$$

Other $v_j(t)$ can be easily calculated and are shown in the table below.

	Day 1	Day 2	Day 3	Day 4	Day 5
$v_1(t)$ happy	0.4	0.064	0.00512	4.090e-4	1.47456e-04
$v_2(t)$ angry	0	0.024	0.00768	2.457e-3	3.93216e-04

The arrows record the path.

3.93216e-04 > 1.47456e-04
The probability of second path is bigger than the first path.
So the most likely sequence for the fisrt five days is **Happy, Angry, Angry, Angry, Angry**.