ECE 6143 - Introduction to Machine Learning
Homework 5

Tolga Dimlioglu
td2249@nyu.edu

12.07.2021

1 Question 1

Random Variables:

- F: Door number of player's first selection
- H: Door number that is revealed by the host
- C: Door number that has the car behind it

$$P(H = 1 \mid C = 1, F = 3) = 0$$
$$P(H = 1 \mid C = 2, F = 3) = 1$$
$$P(H = 1 \mid C = 3, F = 3) = \frac{1}{2}$$

If we use the Bayes rule, we get the following:

$$P(C = 3 \mid H = 1, F = 3) = \frac{P(H = 1, C = 3, F = 3)}{\sum_{i=1}^{3} P(H = 1, C = i, F = 3)}$$
$$P(C = 3 \mid H = 1, F = 3) = \frac{P(H = 1 \mid C = 3, F = 3)P(C = 3 \mid F = 3)P(F = 3)}{\sum_{i=1}^{3} P(H = 1 \mid C = i, F = 3)P(C = i \mid F = 3)P(F = 3)}$$

We also know that:

$$P(C = i \mid F = j) = P(C = i) \text{ for all } i, j \in \{1, 2, 3\}$$
$$P(F = i) = \frac{1}{3} \text{ for all } i \in \{1, 2, 3\}$$

So we can write:

$$P(C = 3 \mid H = 1, F = 3) = \frac{\frac{1}{2}}{0 + \frac{1}{2} + 1} = \frac{1}{3}$$
$$P(C = 2 \mid H = 1, F = 3) = 1 - P(C = 3 \mid H = 1, F = 3) = 1 - \frac{1}{3} = \frac{2}{3}$$

Since the problem is symmetric for all values of $C, H, F \in \{1, 2, 3\}$, this is the general result. The resultant probability values in the last two equation shows that the player has higher probability of winning the car if he switches the door number.

2 Question 2

1.
   - $x_2 - x_5 - x_4$ does not go thru (2 causes)
   - $x_2 - x_1 - x_4$ goes thru (2 effects)

   So the statement is FALSE.

2.
   - $x_2 - (x_5) - x_4$ goes thru (2 causes)

   So the statement is FALSE.

3.
   - $x_2 - x_5 - x_4$ does not go thru (2 causes)
   - $x_2 - (x_1) - x_4$ does not go thru (2 effects)

   So the statement is TRUE.

4.
   - $x_5 - (x_4) - x_3$ does not go thru (Markov chain)
   - $x_5 - x_2 - x_1$ goes thru (Markov chain)
     $x_1 - (x_4) - x_3$ goes thru (2 causes)

   So the statement is FALSE.

5.
   - $x_5 - (x_4) - x_3$ does not go thru (Markov chain)
   - $x_5 - (x_2) - x_1$ does not go thru (Markov chain)

   So the statement is TRUE.

6.
   - $x_1 - x_4 - x_3$ does not go thru (2 causes)
   - $x_1 - x_2 - x_5$ goes thru (Markov chain)
     $x_2 - (x_5) - x_4$ goes thru (2 causes)
     $x_5 - x_4 - x_3$ goes thru (Markov chain)

   So the statement is FALSE.

7.
   - $x_1 - x_4 - x_3$ does not go thru (2 causes)
     $x_1 - (x_2) - x_5$ does not go thru (Markov chain)

   So the statement is TRUE.

8.
   - $x_3 - x_4 - x_1$ does not go thru (2 effects)
   - $x_3 - x_4 - x_5$ goes thru
     $x_4 - x_5 - x_2$ does not go thru (2 effects)

   So the statement is TRUE.

9.
   - $x_3 - x_4 - x_1$ does not go thru (2 effects)
   - $x_3 - x_4 - (x_5)$ goes thru
     $x_4 - (x_5) - x_2$ does not go thru (2 effects)

   So the statement is FALSE.

10.
    - $x_3 - (x_4) - x_1$ goes thru (2 causes)
      $x_4 - x_1 - x_2$ goes thru (2 effects)

    So the statement is FALSE.

3 Question 3

Using the graph in the question, from moralization, we marry the parents that have the same child. Particularly, B1 and B4 have the same child so they are married with the red link. Then, for triangulation, we draw the diagonal line between B2 and B4 to get rid of the 4-cycle in the middle. After these steps, cliques are captured with black drawings.

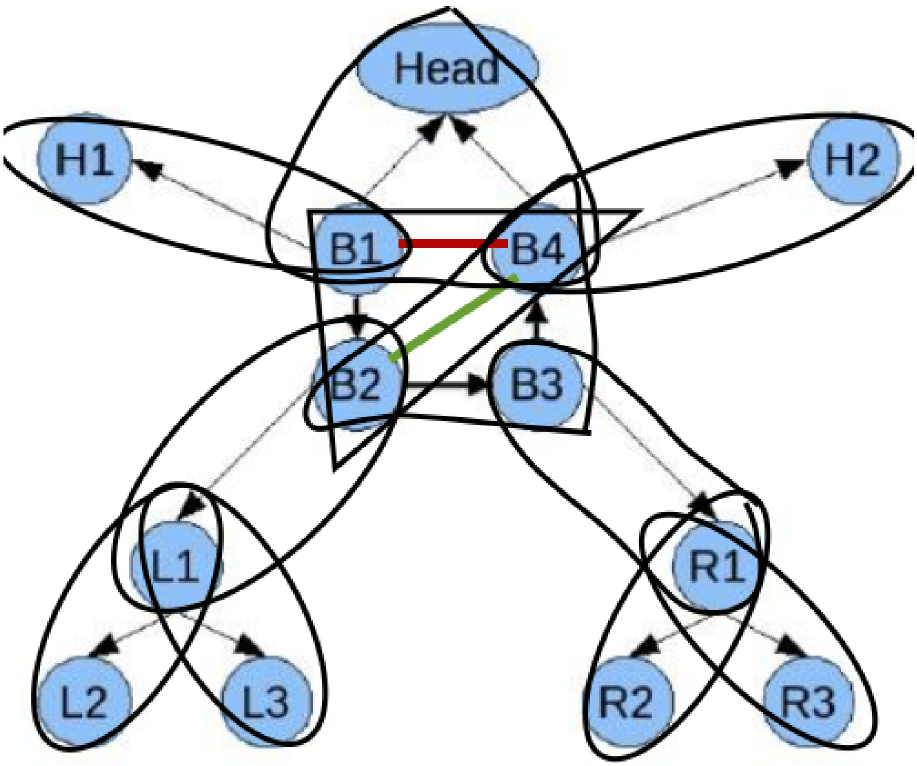

Figure 1: The graph shown with cliques after moralization and triangulation steps

In the next page, junction tree is provided.

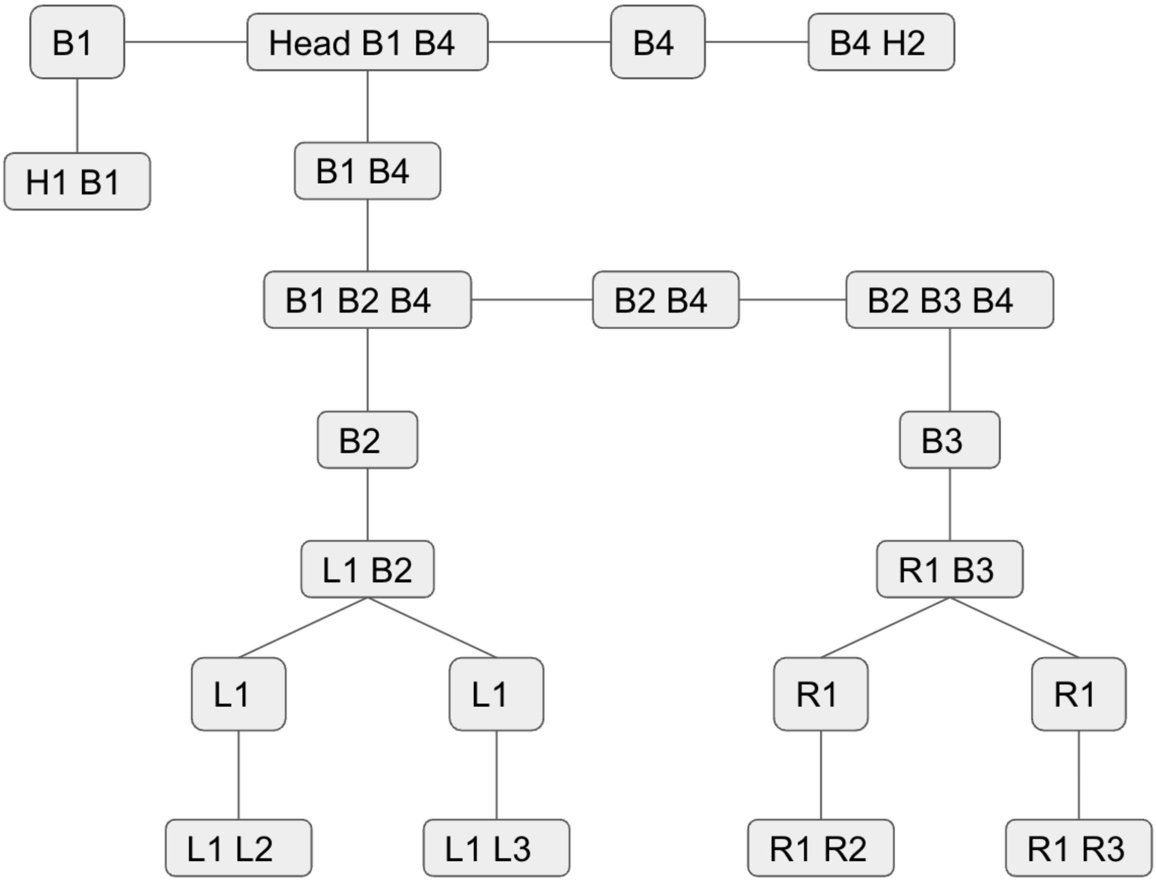

Figure 2: Junction tree that corresponds to the given graph

4 Question 4

In this problem, first the graphical model is converted to a junction tree and its diagram is provided below.

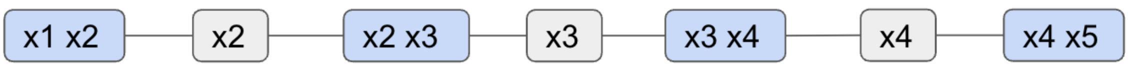

Figure 3: The constructed junction tree from the given graphical model

For the Junction Tree Algorithm (JTA), right-most clique is selected as the root to collect and distribute. After the implementation of the algorithm, it is first tested with random clique potentials as it is stated in the problem. In the figure below, you can see the updated clique potentials after the convergence.

```
Random clique potentials:

upd_psis{1} =

    0.2020    0.0533
    0.6765    0.0683



upd_psis{2} =

    0.0338    0.8446
    0.0314    0.0901



upd_psis{3} =

    0.0320    0.0333
    0.5786    0.3562



upd_psis{4} =

    0.5729    0.0377
    0.2681    0.1214
```

Figure 4: Converged clique potentials after random initialization

And for all clique potentials, sums are add up to 1.

$$\psi(x_1, x_2) = \left\{ \begin{array}{c|cc} & x_2 = 0 & x_2 = 1 \\ \hline x_1 = 0 & 0.1 & 0.7 \\ x_1 = 1 & 0.8 & 0.3 \end{array} \right\}$$

$$\psi(x_2, x_3) = \left\{ \begin{array}{c|cc} & x_3 = 0 & x_3 = 1 \\ \hline x_2 = 0 & 0.5 & 0.1 \\ x_2 = 1 & 0.1 & 0.5 \end{array} \right\}$$

$$\psi(x_3, x_4) = \left\{ \begin{array}{c|cc} & x_4 = 0 & x_4 = 1 \\ \hline x_3 = 0 & 0.1 & 0.5 \\ x_3 = 1 & 0.5 & 0.1 \end{array} \right\}$$

$$\psi(x_4, x_5) = \left\{ \begin{array}{c|cc} & x_5 = 0 & x_5 = 1 \\ \hline x_4 = 0 & 0.9 & 0.3 \\ x_4 = 1 & 0.1 & 0.3 \end{array} \right\}$$

Figure 5: Given clique potentials

Next, we run the algorithm with the given clique potentials that are specified above.

```
Clique potentials:

upd_psis{1} =

    0.0405    0.4451
    0.3237    0.1908


upd_psis{2} =

    0.2601    0.1040
    0.0578    0.5780


upd_psis{3} =

    0.1192    0.1987
    0.6395    0.0426


upd_psis{4} =

    0.5690    0.1897
    0.0603    0.1810
```

(a) Converged potentials

```
p(x1):
    "x1=0"    "x1=1"

    0.4855    0.5145

p(x2):
    "x2=0"    "x2=1"

    0.3642    0.6358

p(x3):
    "x3=0"    "x3=1"

    0.3179    0.6821

p(x4):
    "x4=0"    "x4=1"

    0.7587    0.2413

p(x5):
    "x5=0"    "x5=1"

    0.6293    0.3707
```

(b) Marginal probabilities

Figure 6: JTA algorithm results for the given potentials

As it can be seen from the figures above, converged clique potentials and marginal probabilities are calculated. Notice that, marginal probabilities sum up to 1 which indicates the validity of the implementation.

5 Question 5

In this question, we will be using the Junction Tree Algorithm(JTA) on Hidden Markov Models. Specifically, we will be using ArgMax JTA so that we can find the largest states in the separators. In the MATLAB code,

For emotional states:

- Happy is represented with 1
- Angry is represented with 2

For observed reaction:

- Smile is represented with 1
- Frown is represented with 2
- Laugh is represented with 3
- Yell is represented with 4

Then, the algorithm finds the following result:

```
States found by the algorithm:
     1     2     2     2     2

Happy
Angry
Angry
Angry
Angry
>>
```

Figure 7: ArgMax JTA result

To put it in a table, we have the following most likely emotional states:

Day 1	Day 2	Day 3	Day 4	Day 5
Happy	Angry	Angry	Angry	Angry

6 Appendix

```
% Code for Question 4
clc
clear all

% random psi initialization
n = 5;
rng(31)
psis = cell(n-1, 1);
for i = 1:(n-1)
    psis{i} = rand(2,2);
end

upd_psis = JTA(psis);
disp("Random clique potentials:")
celldisp(upd_psis)

% given psi values
psi_1 = [0.1, 0.7; 0.8, 0.3];
psi_2 = [0.5, 0.1; 0.1, 0.5];
psi_3 = [0.1, 0.5; 0.5, 0.1];
psi_4 = [0.9, 0.3; 0.1, 0.3];
psis = {psi_1, psi_2, psi_3, psi_4};

```

```
upd_psis = JTA(psis);
disp("Clique potentials:")
celldisp(upd_psis)

%calculating marginals

marg_i = [1, 2, 3, 4];
for i = 1:length(upd_psis)
    cur_psi = upd_psis{i};
    disp("p(x" + string(marg_i(i)) + "):" )
    disp("x" + string(marg_i(i)) + "=" + string([0, 1]))
    disp(sum(cur_psi, 2)')
end

disp("p(x5):")
disp("x5=" + string([0, 1]))
disp(sum(upd_psis{4}, 1))


% Function to carry out JTA
function [psis ] = JTA( psis )
    n = length(psis);
    seperators = cell(n-1,1);
    for i = 1:n-1
        seperators{i} = ones(2,1);
    end

    sep_olds = seperators;

    %left to right
    for i = 1:n-1
        sep = sum(psis{i})'; % 2x1
        seperators{i} = sep;
        psis{i+1} = sep./sep_olds{i}.*psis{i+1};
    end


    % right to left
    for i = 1:n-1
        s_old = seperators{n-i};
        sep = sum(psis{n-i+1},2); % 2x1
        seperators{n-i} = sep;
        psis{n-i} = psis{n-i} .* (sep./s_old)';
    end

    % normalization
    for i = 1:n
        psis{i} = psis{i} / sum(sum(psis{i}));
    end

end
```

```
% Code for Question 5
clc
clear all

transition = [[0.8, 0.2];[0.2,0.8]];
emission = [[0.4, 0.1, 0.3, 0.2];[0.1, 0.4, 0.2, 0.3]];
obs = [1, 4, 2, 2, 3];
```

```
init = [1, 0]; % first state is known to be Happy

[a, H] = HMM_JTA( transition, emission, obs, init );


disp("States found by the algorithm:")
disp(H)
for i = 1:length(H)
    if H(i) == 1
        disp("Happy")
    end
    if H(i) == 2
        disp("Angry")
    end

end


function [a, H ] = HMM_JTA( trns, ems, obs, pi )

    % function to run argmax JTA on HMM for Question 5

    m = size(trns, 1); % number of states
    n = length(obs); % number of observations

    psi = zeros(m, m, n);
    phi = ones(m, n);
    phi(:, 1) = pi;

    % get the marginals given observations
    margs =  zeros(m, n);
    for i = 1:n
        j = obs(i);
        margs(:,i) = ems(:,j);
    end

    % left to right
    for i = 2 : n
        psi(:, :, i) =   (phi(:, i - 1) * [1, 1] ) .* trns .* ( margs(:,i) ...
            * [1, 1] );
        phi(:, i) = max(psi(:, :, i)); % we use max rather than summing ...
            over i
    end

    % right to left
    for i = n - 1 : -1 : 1
        upd_phi = max(psi(:, :, i + 1), [], m); % we use max rather than ...
            summing over i
        psi(:, :, i) = psi(:, :, i) * ((upd_phi ./ phi(:, i)) * [1, 1] );
        phi(:, i) = upd_phi;
    end

    [a, H] = max(phi);
end
```