Solutions for assignment 4

Question 1

a. \( P(\text{toothache}) = 0.108 + 0.016 + 0.012 + 0.064 = 0.2 \)

b. \( P(\text{catch}) = 0.108 + 0.016 + 0.072 + 0.144 = 0.34 \)

c. \( P(\text{catch} \mid \text{cavity}) = \frac{P(\text{cavity} \land \text{catch})}{P(\text{cavity})} = \frac{0.108 + 0.072}{0.108 + 0.012 + 0.072 + 0.008} = 0.9 \)

d. \( P(\text{cavity} \mid \text{toothache} \lor \text{catch}) = \frac{P(\text{cavity} \land (\text{toothache} \lor \text{catch}))}{P(\text{toothache} \lor \text{catch})} = \frac{0.108 + 0.012 + 0.072}{0.108 + 0.012 + 0.016 + 0.064 + 0.072 + 0.144} = 0.4615 \)

Question 2

The Bayesian network corresponding to the set up is given below:

![Bayesian Network Diagram]

The CPT for coin selection is

<table>
<thead>
<tr>
<th>C</th>
<th>P(C)</th>
</tr>
</thead>
<tbody>
<tr>
<td>a</td>
<td>1/3</td>
</tr>
<tr>
<td>b</td>
<td>1/3</td>
</tr>
<tr>
<td>c</td>
<td>1/3</td>
</tr>
</tbody>
</table>
The CPTs for $X_1$, $X_2$ and $X_3$ is same and is given as:

<table>
<thead>
<tr>
<th>C</th>
<th>$X_1$</th>
<th>$P(X_1 \mid C)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>a</td>
<td>H</td>
<td>0.4</td>
</tr>
<tr>
<td>b</td>
<td>H</td>
<td>0.5</td>
</tr>
<tr>
<td>c</td>
<td>H</td>
<td>0.7</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>C</th>
<th>$X_2$</th>
<th>$P(X_2 \mid C)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>a</td>
<td>H</td>
<td>0.4</td>
</tr>
<tr>
<td>b</td>
<td>H</td>
<td>0.5</td>
</tr>
<tr>
<td>c</td>
<td>H</td>
<td>0.7</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>C</th>
<th>$X_3$</th>
<th>$P(X_3 \mid C)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>a</td>
<td>H</td>
<td>0.4</td>
</tr>
<tr>
<td>b</td>
<td>H</td>
<td>0.5</td>
</tr>
<tr>
<td>c</td>
<td>H</td>
<td>0.7</td>
</tr>
</tbody>
</table>

For the observed sequence of $2H$ and $1T$, the coin most likely to have been drawn is given by value of $C$ with greatest posterior probability.

Using Bayes Rule,

$$P(C \mid 2H, 1T) = \frac{P(2H, 1T \mid C) \cdot P(C)}{P(2H, 1T)}$$

Since, $P(C)$ is equal for all values of $C$ and $P(2H, 1T)$ is not dependent on $C$, we can ignore them leading to

$$P(C \mid 2H, 1T) \propto P(2H, 1T \mid C)$$

Since $X_1$, $X_2$ and $X_3$ are conditionally independent given $C$, the RHS from above can be written as:

$$= P(H \mid C) \cdot P(H \mid C) \cdot P(T \mid C) \cdot 3$$

The order can be $H,H,T$; $H,T,H$; or $T,H,H$.

For $C = a$, $P(H \mid a) \cdot P(H \mid a) \cdot P(T \mid a) \cdot 3 = 0.4 \cdot 0.4 \cdot 0.6 \cdot 3 = 0.288$

For $C = b$, $P(H \mid b) \cdot P(H \mid b) \cdot P(T \mid b) \cdot 3 = 0.5 \cdot 0.5 \cdot 0.5 \cdot 3 = 0.375$

For $C = c$, $P(H \mid c) \cdot P(H \mid c) \cdot P(T \mid c) \cdot 3 = 0.7 \cdot 0.7 \cdot 0.3 \cdot 3 = 0.441$

Hence, Coin $c$ is most likely to have been drawn.

Question 3
For the first iteration with 9 features, the Information Gain calculated for different features is:
Here, **Pat** is selected since it has the highest information gain.

For the second iteration with 8 features, the information gain calculated for different features is:
- Alt: 0.109, Bar: 0, Fri: 0.109, **Hun: 0.252**, Price: 0.252, Rain: 0.044, Res: 0.252, Est: 0.252

Here, we select **Hun** as the decision feature among the possible alternatives.

For the third iteration with 7 features, the information gain calculated for different feature is:
- Alt: 0, Bar: 0, **Fri: 0.311**, Price: 0.311, Rain: 0.311, Res: 0.311, Est: 0

Here, we select **Fri** as the decision feature among other possible alternatives.

For the fourth iteration with 6 features, the information gain calculated for different feature is:
- Alt: 0, Bar: 0.252, **Price: 0.918**, Rain: 0.252, Res: 0.918, Est: 0.252

Here, we select **Price** as decision feature and then find that all data row has been decided and thus stop the decision tree growth process. The decision tree looks like:

[NEXT PAGE]
Question 4
The value of attribute used is:
- Alt: No=0.1, Yes=0.9
- Bar: No=0.1, Yes=0.9
- Fri: No=0.1, Yes=0.9
- Hun: No=0.1, Yes=0.9
- Pat: None=0.1, Some=0.5, Full=0.9 • Pri: $=0.1, $$=0.5, $$$=0.9
- Rain: No=0.1, Yes=0.9
- Res: No=0.1, Yes=0.9
- Est: 0-10=0.1, 10-30=0.3, 30-60=0.7, >60=0.9 • WillWait: No=0.1, Yes=0.9
The data looks something like:

<table>
<thead>
<tr>
<th>Alt</th>
<th>Bar</th>
<th>Fri</th>
<th>Hun</th>
<th>Pat</th>
<th>Price</th>
<th>Rain</th>
<th>Res</th>
<th>Est</th>
<th>WillWait</th>
</tr>
</thead>
<tbody>
<tr>
<td>.9</td>
<td>.1</td>
<td>.1</td>
<td>.9</td>
<td>.5</td>
<td>.9</td>
<td>.1</td>
<td>.9</td>
<td>.1</td>
<td>.9</td>
</tr>
<tr>
<td>.9</td>
<td>.1</td>
<td>.1</td>
<td>.9</td>
<td>.9</td>
<td>.1</td>
<td>.1</td>
<td>.1</td>
<td>.7</td>
<td>.1</td>
</tr>
<tr>
<td>.1</td>
<td>.9</td>
<td>.1</td>
<td>.1</td>
<td>.5</td>
<td>.1</td>
<td>.1</td>
<td>.1</td>
<td>.1</td>
<td>.9</td>
</tr>
<tr>
<td>.9</td>
<td>.1</td>
<td>.9</td>
<td>.9</td>
<td>.9</td>
<td>.1</td>
<td>.9</td>
<td>.1</td>
<td>.9</td>
<td>.9</td>
</tr>
<tr>
<td>.9</td>
<td>.1</td>
<td>.9</td>
<td>.9</td>
<td>.9</td>
<td>.9</td>
<td>.1</td>
<td>.9</td>
<td>.9</td>
<td>.1</td>
</tr>
<tr>
<td>.1</td>
<td>.9</td>
<td>.1</td>
<td>.1</td>
<td>.5</td>
<td>.5</td>
<td>.9</td>
<td>.9</td>
<td>.1</td>
<td>.1</td>
</tr>
<tr>
<td>.9</td>
<td>.1</td>
<td>.9</td>
<td>.9</td>
<td>.9</td>
<td>.1</td>
<td>.9</td>
<td>.9</td>
<td>.1</td>
<td>.1</td>
</tr>
<tr>
<td>.9</td>
<td>.1</td>
<td>.1</td>
<td>.1</td>
<td>.9</td>
<td>.9</td>
<td>.9</td>
<td>.9</td>
<td>.1</td>
<td>.9</td>
</tr>
<tr>
<td>.9</td>
<td>.9</td>
<td>.9</td>
<td>.9</td>
<td>.9</td>
<td>.1</td>
<td>.1</td>
<td>.1</td>
<td>.7</td>
<td>.1</td>
</tr>
</tbody>
</table>

A neural network with three hidden nodes is used. The weight learned are:

**Linear Node 0 (Output Node)**
- Inputs: Weights
  - Threshold: 1.577737377968832
  - Node 2: -3.111268607474342
  - Node 1: -1.1882211927170858
  - Node 3: -2.680360590844129

**Sigmoid Node 1**
- Inputs: Weights
  - Threshold: -1.931965536985428
  - Attrib Alt: -0.05397663183580977
  - Attrib Bar: -0.5567736290539168
  - Attrib Fri: 0.11819389514540268
  - Attrib Hun: -0.9861285848447071
  - Attrib Pat: -1.6617471621227962
  - Attrib Price: 0.5998563110993793
  - Attrib Rain: 1.357232070603538
  - Attrib Res: 0.0153514927767633

**Sigmoid Node 2**
- Inputs: Weights
  - Threshold: -3.4202774708167567
  - Attrib Alt: 0.7171455422060712
  - Attrib Bar: -1.147138200963823
  - Attrib Fri: -1.6617787841634795
  - Attrib Hun: 0.08772298375146859
  - Attrib Pat: -0.4558624837370548
  - Attrib Price: -0.6901769740456603
  - Attrib Rain: 9.801280479822365E-4
  - Attrib Res: -1.1478131734712538

**Sigmoid Node 3**
- Inputs: Weights
  - Threshold: -0.5280515253637101
Time taken to build model: 1179.81 seconds

This is the weights learned by the neural network during back-propagation.

Number of Epochs: 50  (Iterations: 50 * 12)
Learning rate: 0.3
Momentum : 0.1

Relative absolute error 0.0001 %
Root relative squared error 0.0002 %