CS 188 Final Solutions

May 19, 2008
1 True/False Questions (14 points)

Circle True or False next to each statement below. Correct answers are worth 1 point each, incorrect answers are worth 0 points, and skipped parts are worth 0.5 points each.

(a) (False) Given any probability distribution over \( N \) variables, every Bayes’ net with \( N \) nodes can represent it.

(b) (False) All conditional independence properties of the probability distribution represented by a Bayes’ net are determined by the graph structure.

(c) (True) Given a Bayes’ net with \( N \) nodes that can represent any probability distribution over \( N \) variables, that same network can represent any marginal distribution over \( M \) variables where \( M \leq N \).

(d) (True) In the worst case, the complexity of inference could be exponential in the size (number of variables) of a Bayes’ net.

(e) (False) The smoothing algorithm for HMMs returns the single most likely sequence of hidden states given the observations.

(f) (True) In reinforcement learning, it is useful to sometimes choose an action believed to be suboptimal.

(g) (False) Q-learning algorithms require a model of the environment.

(h) (False) In a two-player zero-sum game, player A could do worse than the result implied by his minimax strategy if his opponent, player B, plays suboptimally.

(i) (True) For tree search problems, if an admissible heuristic is used, the A* search algorithm will never return a suboptimal goal node.

(j) (False) Given that \( h_1(x) \) and \( h_2(x) \) are both admissible heuristics for an A* search problem, \( h(x) = h_1(x) + h_2(x) \) is also an admissible heuristic.

(k) (False) The result of training both Neural Networks and Support Vector Machines may be a local optimum different from the global optimum.

(l) (True) Maximizing information gain is a good heuristic for constructing decision trees.

(m) (True) Brightness, color, texture, binocular disparity, and optical flow are all useful cues for detecting boundaries in natural images.

(n) (True) A sentence can have multiple syntactically correct parses.

For official use only:

<table>
<thead>
<tr>
<th>Q1</th>
<th>Q2</th>
<th>Q3</th>
<th>Q4</th>
<th>Q5</th>
<th>Q6</th>
<th>Q7</th>
<th>Q8</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>/14</td>
<td>/8</td>
<td>/7</td>
<td>/15</td>
<td>/20</td>
<td>/5</td>
<td>/11</td>
<td>/6</td>
<td>/86</td>
</tr>
</tbody>
</table>
2 A* Search (8 points)

Consider the following search graph starting from node S with a goal state G.

(a) (3 points) What is the path that is returned using A* search with the null heuristic, \( h(x) = 0 \)? Also list the remaining states in the priority queue when the goal state is found along with their enqueued values.

The optimal path is \( S \rightarrow B \rightarrow C \rightarrow D \rightarrow E \rightarrow G \), with a cost of 8. No states are left in the priority queue.

(b) (3 points) Create a table and record the best admissible heuristic for each node in the graph above.

The best admissible heuristic \( h \) is the exact cost to the goal. Thus, we have the following values:

<table>
<thead>
<tr>
<th>Node</th>
<th>Heuristic</th>
</tr>
</thead>
<tbody>
<tr>
<td>S</td>
<td>8</td>
</tr>
<tr>
<td>B</td>
<td>7</td>
</tr>
<tr>
<td>C</td>
<td>5</td>
</tr>
<tr>
<td>D</td>
<td>4</td>
</tr>
<tr>
<td>E</td>
<td>3</td>
</tr>
<tr>
<td>F</td>
<td>7</td>
</tr>
<tr>
<td>G</td>
<td>0</td>
</tr>
<tr>
<td>H</td>
<td>6</td>
</tr>
</tbody>
</table>

(c) (2 points) Why can’t the solution to part (b) be done practically for most problems?

It usually requires that we’ve already solved the search problem.
3 Games and Minimax Search (7 points)

Company A and company B are competitors for the same market. Currently, A makes a profit of $5 million and B makes a profit of $3 million. Neither company advertises heavily. If A were to advertise, they would steal part of the market from B, increases their own profit by $1 million and decreasing B’s profit by $1.5 million. If B were to advertise, they would increase their profit by $2 million and decrease A’s profit by $1 million. If both advertise, both profits decrease by $0.5 million.

(a) (2 points) What is the Nash equilibrium for this situation?

Regardless of the other player’s strategy, it is better to advertise. The Nash equilibrium is for both to advertise and lose $0.5 million.

Consider the following minimax tree where maximizing nodes are upward triangles that are shaded in and minimizing nodes are downward triangles.

![Minimax Tree](image)

(b) (2 points) What is the minimax value for the root? 15

(c) (3 points) Draw an X through any nodes which will not be visited by alpha-beta pruning, assuming we explore left subtrees before right subtrees.

Shown above.
Consider the following Markov Decision Process:

We have states $S_1, S_2, S_3, S_4, S_5,$ and $S_6$. We have actions $Up$, $Down$, $Left$, and $Right$, and each action deterministically leads to a successor state (with probability 1). Actions not shown are not allowed. For example, in $S_1$, the only available action is to go to $S_2$ ($Down$). We follow the Sutton-Barto convention and associate rewards with transitions. The reward for any action is specified above. For taking $Down$ from $S_4$, the reward is 10. Assume a discount factor $\gamma = 0.5$. Recall that $\sum_{i=1}^{\infty} 0.5^i = 1$

(a) (1 point) What is the optimal policy for this MDP?

\[
\{ S_1 : D \, , \, S_2 : R \, , \, S_3 : R \, , \, S_4 : D \, , \, S_5 : U \, , \, S_6 : D \} 
\]

Value Iteration: When performing value iteration, what is the value of state $S_3$ after

(b) (1 point) 1 iteration: 20

(c) (1 point) 2 iterations: 20.5

(d) (2 points) $\infty$ iterations: 21

Policy Iteration: Suppose you run policy iteration. During each iteration, you compute the exact values in the current policy, then update the policy using one-step lookahead given those values. Suppose, too, that when choosing actions when updating the policy, ties are broken by choosing $Up$, $Right$, $Down$, $Left$ in that order.

Suppose we start with the following policy:

\[
\{ S_1 : Down, \, S_2 : Right, \, S_3 : Up, \, S_4 : Down, \, S_5 : Up, \, S_6 : Down \} 
\]

What will be the policy after... (Note: Fill in with $U$, $D$, $R$, and $L$ for $Up$, $Down$, $Right$, and $Left$)

(e) (2 points) 1 iteration: $\{ S_1 : D, \, S_2 : R, \, S_3 : R, \, S_4 : D, \, S_5 : U, \, S_6 : D \}$

(f) (2 points) 2 iterations: $\{ S_1 : D, \, S_2 : R, \, S_3 : R, \, S_4 : D, \, S_5 : U, \, S_6 : D \}$

(g) (2 points) 3 iterations: $\{ S_1 : D, \, S_2 : R, \, S_3 : R, \, S_4 : D, \, S_5 : U, \, S_6 : D \}$
Q-learning: Consider executing Q-learning on this MDP. Assume the learning rate $\alpha = 0.5$, and that Q-learning uses a greedy policy, meaning that it always chooses the action with maximum Q-value. Suppose the algorithm breaks ties by choosing Up, Right, Down, Left in that order.

(h) (2 points) What are the first 10 \((state, action)\) pairs visited if our agent learns using Q-learning and starts in \(S_3\)?

\[(S_3, Up)\]
\[(S_4, Down)\]
\[(S_3, Up)\]
\[(S_4, Down)\]
\[(S_3, Up)\]
\[(S_4, Down)\]
\[(S_3, Up)\]
\[(S_4, Down)\]
\[(S_3, Up)\]
\[(S_4, Down)\]

Since we are not using an policy with exploration (such as $\epsilon$-greedy), Q-learning is stuck in a loop between \(S_3\) and \(S_4\).

(i) (2 points) What is the Q-value of \((S_5, Up)\) after these first 10 actions? 0
5 Hidden Markov Models (20 points)

Recall that on the midterm, we helped Billy model the state of his health using the following Bayes’ net. He could exhibit symptoms coughing (C), sneezing (S), and temperature (T), with underlying cause (X) as sick (X = sick), allergic (X = allergic), or well (X = well).

Billy has now been sick for several days and asks you to formalize the previous Bayes’ net as the Hidden Markov Model (HMM) shown in Figure 2.

Billy has not been keeping track of his temperature, but does remember the history of his coughing and sneezing symptoms. Hence, although we modeled his temperature in creating the diagnostic criteria, the only observations we have are whether Billy is coughing or sneezing on each day.

Recall that we parameterize the distribution $P(X_k, S_1, ..., S_n, C_1, ..., C_n)$ with $\alpha(X_k), \beta(X_k)$, so

$$P(X_k, S_1, ..., S_n, C_1, ..., C_n) = \alpha(X_k)\beta(X_k)$$

$$\alpha(X_k) = P(X_k, S_1, ..., S_k, C_1, ..., C_k)$$

$$\beta(X_k) = P(S_{k+1}, ..., S_n, C_{k+1}, ..., C_n|X_k)$$
Also recall the following equations from the midterm, which may be useful in answering the questions below:

\[
P(C, S) = \sum_x \sum_t P(X = x, C, T = t, S)
\]

\[
P(X|C, S) = \frac{\sum_t P(X, C, T = t, S)}{\sum_t \sum_s P(X = x, C, T = t, S)}
\]

\[
P(C, S|X) = \frac{\sum_t P(X, C, T = t, S)}{\sum_t \sum_s \sum_x P(X, S = c, T = t, S = s)}
\]

(a) (2 points) Derive the probability distribution for \(P(X_k, S_1, ..., S_k, C_1, ..., C_k)\) in terms of \(P(X_i|X_{i-1})\), and \(P(X, C, T, S)\) using marginalization and conditional independence.

\[
P(X_1, ..., X_k, S_1, ..., S_k, C_1, ..., C_k) = P(X_1) \left[ \prod_{i=2}^{k} P(X_i|X_{i-1}) \right] \left[ \prod_{i=1}^{k} P(C_i, S_i|X_i) \right]
\]

\[
P(X_k, S_1, ..., S_k, C_1, ..., C_k) = \sum_{x_1} \sum_{x_2} \cdots \sum_{x_{k-1}} P(X_1 = x_1, ..., X_{k-1} = x_{k-1}, X_k, S_1, ..., S_k, C_1, ..., C_k)
\]

\[
= \sum_{x_1} \sum_{x_2} \cdots \sum_{x_{k-1}} P(X_1) \left[ \prod_{i=2}^{k} P(X_i|X_{i-1}) \right] \left[ \prod_{i=1}^{k} P(C_i, S_i|X_i) \right]
\]

Substituting in \(P(X_1) = \sum_x \sum_t \sum_s P(X_1, C_1 = c, T_1 = t, S_1 = s)\) as well as our equation for \(P(C, S|X)\) above yields the desired formula.

(b) (2 points) Derive the probability distribution for \(P(S_{k+1}, ..., S_n, C_{k+1}, ..., C_n|X_k)\) in terms of \(P(X_i|X_{i-1})\), and \(P(X, C, T, S)\) using marginalization and conditional independence.

\[
P(X_{k+1}, ..., X_n, S_{k+1}, ..., S_n, C_{k+1}, ..., C_n|X_k) = \prod_{i=k+1}^{n} [P(C_i, S_i|X_i)P(X_i|X_{i-1})]
\]

\[
P(S_{k+1}, ..., S_n, C_{k+1}, ..., C_n|X_k) = \sum_{x_{k+1}} \sum_{x_{k+2}} \cdots \sum_{x_n} P(X_{k+1} = x_{k+1}, ..., X_n = x_n, S_{k+1}, ..., S_n, C_{k+1}, ..., C_n|X_k)
\]

\[
= \sum_{x_{k+1}} \sum_{x_{k+2}} \cdots \sum_{x_n} \prod_{i=k+1}^{n} [P(C_i, S_i|X_i)P(X_i|X_{i-1})]
\]

We need only substitute in the equation for \(P(C, S|X)\) above to produce the desired formula.

(c) (2 points) Derive the \(\alpha(X_k)\) update in terms of \(\alpha(X_{k-1})\), \(P(X_k|X_{k-1})\), and \(P(X, C, T, S)\). Remember that you only have observations for \(C\) and \(S\).

\[
\alpha(X_k) = \frac{P(X_k, S_1, ..., S_k, C_1, ..., C_k)}{\sum_x P(X_k, X_{k-1} = x, S_1, ..., S_k, C_1, ..., C_k)}
\]

\[
= \frac{\sum_x P(C_k, S_k|X_k)P(X_k|X_{k-1} = x)P(X_{k-1} = x, S_1, ..., S_{k-1}, C_1, ..., C_{k-1})}{\sum_x P(C_k, S_k|X_k)\sum_x P(X_k|X_{k-1} = x)P(X_{k-1} = x, S_1, ..., S_{k-1}, C_1, ..., C_{k-1})}
\]

\[
= \frac{P(C_k, S_k|X_k)\sum_x P(X_k|X_{k-1} = x)\alpha(X_{k-1})|X_{k-1}=x}{P(C_k, S_k|X_k)\sum_x P(X_k|X_{k-1} = x)\alpha(X_{k-1})|X_{k-1}=x}
\]

We again substitute in the formula for \(P(C, S|X)\) in terms of \(P(X, C, T, S)\).
(d) (2 points) Derive the $\beta(X_k)$ update in terms of $\beta(X_{k+1})$, $P(X_{k+1}|X_k)$, and $P(X,C,T,S)$. Remember that you only have observations for $C$ and $S$.

$$\beta(X_k) = \frac{P(S_{k+1}, \ldots, S_n, C_{k+1}, \ldots, C_n | X_k)}{\sum_x P(S_{k+1}, \ldots, S_n, C_{k+1}, \ldots, C_n, X_{k+1} = x | X_k)} = \frac{\sum_x P(S_{k+1}, C_{k+1} | X_{k+1} = x) P(S_{k+2}, \ldots, S_n, C_{k+2}, \ldots, C_n | X_{k+1} = x) P(X_{k+1} = x | X_k)}{\sum_x P(C_{k+1}, S_{k+1} | X_{k+1} = x) P(X_{k+1} = x | X_k) \beta(X_{k+1}) | X_{k+1} = x}$$

We again substitute in the formula for $P(C,S|X)$ in terms of $P(X,C,T,S)$.

(e) (2 points) Derive the equation for filtering using $\alpha(X_k)$, to find $P(X_k|S_1, \ldots, S_k, C_1, \ldots, C_k)$

$$P(X_k|S_1, \ldots, S_k, C_1, \ldots, C_k) = \frac{P(X_k, S_1, \ldots, S_k, C_1, \ldots, C_k)}{P(S_1, \ldots, S_k, C_1, \ldots, C_k)} = \frac{P(X_k, S_1, \ldots, S_k, C_1, \ldots, C_k)}{\sum_x P(X_k = x, S_1, \ldots, S_k, C_1, \ldots, C_k)} = \frac{\alpha(X_k)}{\sum_x \alpha(X_k) | X_k = x}$$

(f) (2 points) Derive the equation for prediction using $\alpha(X_k)$, to find $P(X_{k+1}|S_1, \ldots, S_k, C_1, \ldots, C_k)$. Hint: it may help to draw this Bayes’ net.

$$P(X_{k+1}|S_1, \ldots, S_k, C_1, \ldots, C_k) = \sum_x P(X_{k+1}, X_k = x | S_1, \ldots, S_k, C_1, \ldots, C_k) = \sum_x P(X_{k+1} | X_k = x) P(X_k = x | S_1, \ldots, S_k, C_1, \ldots, C_k)$$

Substituting in the result of part (e) for $P(X_k = x | S_1, \ldots, S_k, C_1, \ldots, C_k)$ in terms of $\alpha(X_k)$ yields the desired equation.

(g) (2 points) Derive the equation for smoothing using $\alpha(X_k)$ and $\beta(X_k)$, to find $P(X_k|S_1, \ldots, S_n, C_1, \ldots, C_n)$.

$$P(X_k|S_1, \ldots, S_n, C_1, \ldots, C_n) = \frac{P(X_k, S_1, \ldots, S_n, C_1, \ldots, C_n)}{P(S_1, \ldots, S_n, C_1, \ldots, C_n)} = \frac{P(X_k, S_1, \ldots, S_n, C_1, \ldots, C_n, C_{k+1}, \ldots, C_n | X_k)}{P(S_1, \ldots, S_n, C_1, \ldots, C_n, C_{k+1}, \ldots, C_n) \beta(X_k)} = \frac{\alpha(X_k) \beta(X_k)}{P(S_1, \ldots, S_n, C_1, \ldots, C_n)} \propto \frac{\alpha(X_k)}{\beta(X_k)}$$

as the denominator is the normalization constant.
(h) (2 points) Derive the Viterbi update ($\alpha^*(X_k)$) for this HMM.

The Viterbi update is the same as the update for $\alpha(X_k)$ with the sum replaced by max.

$$\alpha^*(X_k) = P(C_k, S_k | X_k) \max_x [P(X_k | X_{k-1} = x) \alpha(X_{k-1}) | X_{k-1} = x]$$

(i) (2 points) Briefly describe how you would extract the most-likely sequence from $\alpha^*(X_k)$. Would you need any data structures? If so, specify which and what you would keep track of.

You need a data structure that at each timestep, for each possible value of the hidden variable $X$, keeps a backpointer to the value of the hidden variable at the previous timestep that maximizes the equation in (h) (the argmax). The most likely sequence is extracted by following these pointers backwards from the final state with highest $\alpha^*(X_n)$.

Suppose that we knew the reward of taking Robitassium DM when Billy has a allergy ($X = allergy$) or when he is well ($X = well$) is $-1$. Additionally, if Billy takes Benadryll when he is sick or when he is well, he receives a reward of $-1$. If Billy uses the medicine correctly, taking Robitassium DM for a sickness or Benadryll for his allergies, or nothing when he is well, he will receive a reward of $+1$.

(j) (2 points) Assuming that Billy observes his symptoms ($S$ and $C$) when he wakes up each day before (possibly) taking medicine, explain how he should decide which (if any) medicine to use so as to maximize his utility. Assume that Billy has a record of his symptoms from previous days.

Billy should use filtering to estimate the probability distribution $P(X_k | S_1, ..., S_k, C_1, ..., C_k)$. Letting

$$x^* = \arg\max_x P(X_k = x | S_1, ..., S_k, C_1, ..., C_k)$$

he should then take the medicine (or do nothing) corresponding to diagnosis $x^*$. 
6 Neural Networks (5 points)

Consider the Neural Network shown below. We’ve drawn the output layer as well as the layer immediately before it. Nodes in the output layer are indexed by $i$ and nodes in the previous layer are indexed by $j$. The outputs are $a_i = g(in_i)$ where $g$ is the sigmoid function and $in_i = \sum_j W_{ji} a_j$ is the input to node $i$.

![Neural Network Diagram](image)

Neural Network. For clarity, only a subset of connections are shown.

(5 points) Given desired output vector $y_i$, we want to adjust weights so as to minimize the error $E = \frac{1}{2} \sum_i (y_i - a_i)^2$. Derive the expression for $\frac{\partial E}{\partial W_{ji}}$, the partial derivative of the error $E$ with respect to weight $W_{ji}$.

$$
\frac{\partial E}{\partial W_{ji}} = \frac{\partial}{\partial W_{ji}} \left[ \frac{1}{2} \sum_i (y_i - a_i)^2 \right] \\
= \frac{1}{2} \frac{\partial}{\partial W_{ji}} (y_i - a_i)^2 \\
= (y_i - a_i) \frac{\partial}{\partial W_{ji}} (y_i - a_i) \\
= -(y_i - a_i) \frac{\partial}{\partial W_{ji}} (a_i) \\
= -(y_i - a_i) g'(in_i) \frac{\partial}{\partial W_{ji}} (in_i) \\
= -(y_i - a_i) g'(in_i) \frac{\partial}{\partial W_{ji}} (\sum_j W_{ji} a_j) \\
= -(y_i - a_i) g'(in_i) a_j
$$
7 Classification (11 points)

We are designing an intrusion detection system using three sensors, a sound level meter \((S)\), an infrared camera \((C)\), and a floor pressure sensor \((F)\). Unfortunately, each sensor is relatively unreliable and hence we want to base the decision on whether or not an intruder \((I)\) is present on the output of all three.

The sound meter can indicate one of two states \((S = \text{quiet or loud})\).
The infrared camera can indicate one of three conditions \((C = \text{cold, warm, or hot})\).
The floor pressure sensor can indicate one of three conditions \((F = \text{light, medium, or heavy})\).

We want to decide whether or not an intruder is present \((I = \text{true or false})\).

(a) (2 points) Suppose we choose to use a naive Bayes’ classifier. Draw the corresponding Bayes’ net.

\[
\begin{array}{c}
I \\
\downarrow \\
S \\
\uparrow \\
C \\
\uparrow \\
F
\end{array}
\]

To collect training data for our classifier, we open the room in which our sensors are set up and observe the sensor values as people enter and leave the room. We obtain the following observations:

- \(S = \text{quiet}, C = \text{hot}, F = \text{heavy}, I = \text{true}\)
- \(S = \text{loud}, C = \text{warm}, F = \text{medium}, I = \text{true}\)
- \(S = \text{quiet}, C = \text{hot}, F = \text{light}, I = \text{false}\)
- \(S = \text{quiet}, C = \text{warm}, F = \text{light}, I = \text{false}\)
- \(S = \text{loud}, C = \text{warm}, F = \text{medium}, I = \text{true}\)
- \(S = \text{quiet}, C = \text{warm}, F = \text{medium}, I = \text{true}\)
- \(S = \text{loud}, C = \text{hot}, F = \text{medium}, I = \text{true}\)

(b) (3 points) Using this training data, compute estimates for the parameters of the naive Bayes’ classifier, without smoothing.

\[
\begin{align*}
P(I = \text{true}) &= 5/8 \\
P(S = \text{quiet}|I = \text{true}) &= 2/5 \\
P(S = \text{quiet}|I = \text{false}) &= 1 \\
P(C = \text{cold}|I = \text{true}) &= 0 \\
P(C = \text{cold}|I = \text{false}) &= 0 \\
P(C = \text{warm}|I = \text{true}) &= 3/5 \\
P(C = \text{warm}|I = \text{false}) &= 2/3 \\
P(C = \text{hot}|I = \text{true}) &= 2/5 \\
P(C = \text{hot}|I = \text{false}) &= 1/3 \\
P(F = \text{light}|I = \text{true}) &= 0 \\
P(F = \text{light}|I = \text{false}) &= 2/3 \\
P(F = \text{medium}|I = \text{true}) &= 4/5 \\
P(F = \text{medium}|I = \text{false}) &= 1/3 \\
P(F = \text{heavy}|I = \text{true}) &= 1/5 \\
P(F = \text{heavy}|I = \text{false}) &= 0
\end{align*}
\]
(c) (2 points) Are all entries in the conditional probability tables in part (b) reasonable? If not, describe how you’d use smoothing to correct the problem.

No, because of limited data we’ve estimated some probabilities as zero. To correct the problem, we can use smoothing to add a small constant $k$ to the count of each occurrence.

(d) (2 points) Assume that we do not make any corrections and use the parameters as they are in part (b). Our sensors read $S =$ quiet, $C =$ warm, and $F =$ medium. What is the probability an intruder is present?

$$P(I = T, S = Q, C = W, F = M) = P(I = T)P(S = Q|I = T)P(C = W|I = T)P(F = M|I = T)$$
$$= \frac{5}{8} \cdot \frac{2}{5} \cdot \frac{3}{5} \cdot \frac{4}{5}$$
$$= \frac{3}{25} = 0.12$$

$$P(I = F, S = Q, C = W, F = M) = P(I = F)P(S = Q|I = F)P(C = W|I = F)P(F = M|I = F)$$
$$= \frac{3}{8} \cdot 1 \cdot \frac{2}{3} \cdot \frac{1}{3}$$
$$= \frac{1}{12} = 0.083$$

$$P(I = T|S = Q, C = W, F = M) = \frac{P(I = T, S = Q, C = W, F = M)}{P(I = T, S = Q, C = W, F = M) + P(I = F, S = Q, C = W, F = M)}$$
$$= \frac{\frac{3}{25}}{\frac{3}{25} + \frac{1}{12}}$$
$$= \frac{3}{25 + \frac{1}{12}}$$
$$= 0.59$$

(e) (2 points) Without making any changes to the sensors, what additional information could we use and how would we model it?

We could use an HMM to take advantage of sensor data collected over time. Our hidden state, $I$ is expected to change slowly over time as once a person enters the room, they will likely remain for a while before leaving.
8 Utility (6 points)

A gambler has the opportunity to make bets on the outcome of a sequence of coin flips. If the coin comes up heads, she wins as many dollars as she has staked on that flip; if it is tails she loses her stake. The game ends when the gambler wins by reaching her goal of attaining $4, or loses by running out of money. On each flip, the gambler must decide what portion of her capital to stake, in integer number of dollars. The state is the gambler’s capital $s \in \{1, 2, 3\}$. The reward is zero on all transitions except ones in which the gambler reaches her goal, when it is +1.

Let the probability of the coin coming up heads be $p$. Solve for the utilities in two cases:

(a) (3 points) For $p = 0.7$, the optimal policy is to always bet $1. Calculate the utilities for each of the three non-terminal states.

\[
\begin{align*}
U_1 &= 0.7 \times U_2 + 0.3 \times 0 \\
U_2 &= 0.7 \times U_3 + 0.3 \times U_1 \\
U_3 &= 0.7 \times 1 + 0.3 \times U_2
\end{align*}
\]

Solving this system of equations yields

\[
\begin{align*}
U_1 &= 0.59 \\
U_2 &= 0.84 \\
U_3 &= 0.95
\end{align*}
\]

(b) (3 points) For $p = 0.4$, the optimal policy is non-obvious. Implement value iteration to calculate the utilities. Start with an initial guess of $U_1 = 0.25$, $U_2 = 0.5$, $U_3 = 0.75$

There are two actions to consider for state $s = 2$, betting $1$ or $2$. We need only consider betting $1$ when $s \in \{1, 3\}$. Running value iteration, we obtain the following sequence of utilities:

<table>
<thead>
<tr>
<th>$U_1$</th>
<th>$U_2$</th>
<th>$U_3$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.25</td>
<td>0.5</td>
<td>0.75</td>
</tr>
<tr>
<td>0.20</td>
<td>0.45</td>
<td>0.70</td>
</tr>
<tr>
<td>0.18</td>
<td>0.40</td>
<td>0.67</td>
</tr>
<tr>
<td>0.16</td>
<td>0.40</td>
<td>0.64</td>
</tr>
<tr>
<td>0.16</td>
<td>0.40</td>
<td>0.64</td>
</tr>
</tbody>
</table>