CS 495 & 540: Problem Set 3

Section: MW 10-11:50 am

Total: 150pts Due: 04/06/2016

Instructions:
1. I leave plenty of space on each page for your computation. If you need more sheet, please attach your work right behind the corresponding problem. If your answer is incorrect but you show the computation process, then partial credits will be given. Please staple your solution and use the space wisely.

2. This assignment contains two parts, both are due on April 6th. Non-coding part counts for 110/150 of the assignment. You are allowed to work on the homework in a group of size up to two. No late assignment is accepted. Identical solutions (same wording, paragraph, code), turned in by different groups (persons), will be considered cheating.

3. Full credit will be given only to the correct solution which is described clearly. Convoluted and obtuse descriptions might receive low marks, even when they are correct. Also, aim for concise solutions, as it will save you time spent on write-ups, and also help you conceptualize the key idea of the problem.

First Name:

Last Name:

Group ID:

Score: /

Problem 1 Inference with Baye's Rule: 20pts

Given the following information: (if not computable, please say so)

$$P(+m) = 0.0001, \quad P(+s|+m) = 0.8 \quad P(+s|-m) = 0.1.$$

Please compute the following:
(a) $P(-s)$

(b) $P(-m|+s)$

(c) $P(-m|-s)$

(d) $P(+m|-s)$

Problem 2 Random Variable Independence: 20pts

Given random variables $X_1, X_2, \cdots, X_5$ we know $X_3 \perp\!\!\!\perp X_1 | X_2$, $X4 \perp\!\!\!\perp X_1, X_2 | X_3$ and $X_5 \perp\!\!\!\perp X_1, X_2, X_3 | X_4$. Please show $P(X_1, X_2 \cdots X_5) = P(X_1) \prod_{t=2}^{5} P(X_t | X_{t-1})$

(b) Suppose now we have a Markov chain for binary random variables X_1, X_2, X_3. Please generate a joint probability distribution for those three binary random variables and **please VERIFY** that your joint probability distribution is indeed Markovian.

Problem 3 Stochastic Matrix: Stationary Distribution 20pts

(a) Please describe the conditions (properties) of a stochastic matrix that will converge to a stationary distribution.

(b) Please describe the parameter that captures the convergence rate (how fast the initial distribution will converge to stationary distribution).

Problem 4 Hidden Markov Model: Computation 28pts (8+8+12)

Hidden Markov Model (HMM) basically is a belief propatation process. It starts with an initially uniform distribution for the unseen random variable X by checking the highly correlated observable evidence variable e. Please briefly describe (along with mathematical expression)

(I) passage of time phase $B'(X_{t+1}) = P(X_{t+1}|e_{1\cdots t})$

(II) observation phase $B(X_{t+1}) = P(X_{t+1}|e_{1\cdots t+1})$

(III) Follow the weather HMM in the slide with the following modification: $P(+r \to +r) = 0.6, P(+r \to -r) = 0.4, P(-r \to +r) = 0.4, P(-r \to -r) = 0.6, P(+r \to +u) = 0.8, P(+r \to -u) = 0.2, P(-r \to +u) = 0.3, P(-r \to -u) = 0.7$. And let assume you live in Seattle, therefore your initial belief is $B_0(+r) = 0.65, B_0(-r) = 0.35$. Please compute $B'_i(+r), B'_i(-r), B_i(+r), B_i(-r)$ where $i = 1, 2, 3$.

1.$B'_1(+r) =$

2.$B'_1(-r) =$

3.$B_1(+r) =$

4.$B_1(-r) =$

5.$B'_2(+r) =$

6.$B'_2(-r) =$

$7. B_2(+r) =$

$8. B_2(-r) =$

$9. B'_3(+r) =$

$10. B'_3(-r) =$

$11. B_3(+r) =$

$12. B_3(-r) =$

Problem 5 HMM: Implied Indepedence 12pts

Given a HMM as shown in the lecture slides, we know that

$$X_2 \perp\!\!\!\perp E_1 | X_1, \quad E2 \perp\!\!\!\perp X_1, E_1 | X_2, \quad X_3 \perp\!\!\!\perp X_1, E_1, E_2 | X_2, \quad E_3 \perp\!\!\!\perp X_1, E_1, X_2, E_2 | X_3$$

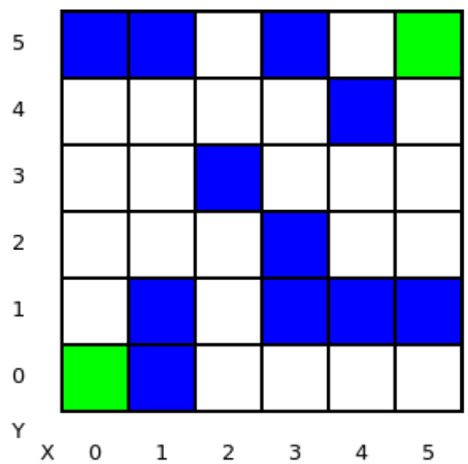
Please show the implied independence $E_1 \perp\!\!\!\perp X_2, E_2, X_3, E_3 | X_1$

Problem 6 Research: Applications 10pts

So far we have learned about CSP (cutset, tree-like structure, simulated annealing, k-consistency, hill climbing), various search (A*, backtracking, Iterative Deepening, BFS, DFS, adversarial search), reinforcement learning (Q-value iteration), MDP (value iteration, policy iteration, policy evaluation), Markov Model, Hidden Markov Model. Please briefly describe a paper (or an algorithm) that adopts any of the techniques mentioned above to solve practical problems (or how the techniques are modfied to improve the performance).

Problem 7 Programming Assignment II: 40pts (5+35)

This part of programming is getting you familiar with Python and A* search. Here is the configuration:
You are given a 6 cells by 6 cells maze. The barriers are colored in blue. The starting cell is located at coordinate ($x = 0$ and $y = 0$) colored in green. The ending cell is located at the top right ($x = 5$ and $y = 5$) colored in green. Allowed moves are up, down, left and right, 1 cell at a time. Lets assume the cost of each move (1 cell) is equal to 10. A



As you know in the A* search algorithm we must have two cost functions. One is the G and the other is H. G is the cost to move from the starting cell to a given cell and H is a cost estimate from a given cell to the ending cell. You will have to define your H function but make sure it follows the constraint that it must be less than the real cost. You must also have the open list and the closed list in your implementation (closed list: cells that have been visited, open list: cells that are queued up waiting to be explored).

(a) Briefly describe your H function and explain why it is appropriate to use it in your code.

(b) To make your life earlier, you can simply code for the situation given in the picture (1. the barrier $= ((0,5),(1,0),(1,1),(1,5),(2,3),(3,1),(3,2),(3,5),(4,1),(4,4),(5,1))$ 2. the maze is 6 by 6 3. the start is $(0,0)$ and the end is $(5,5)$. You should have a function called *initGrid* that would take the the barrier coordinate to represent the maze(configuration) such that your agent can explore based on this configuration. Whenever your agent makes a move, please print out the elemnts in the open list and the closed list.

(c) [**Bonus For Midterm: 5pts**] Similar to (b), but now *initGrid* that would take the dimenion of the maze (arbitrary size $m \times n$) , the barrier, the start and the end cell coordinate (you can assume it is $(0,0)$ and $(m-1, n-1)$[1]) to represent the maze(configuration) such that your agent can explore based on this configuration. Whenever your agent makes a move, please print out the elemnts in the open list and the closed list.

[1]Please note that whether fix the start/end points or not, you still need to have these two parameters in your function. Within your function, you must also check to make sure the inputs are valid. For instance, if $(m, n) = (5, 4)$, you cannot have a start point $(-1, 5)$