**Northeastern University**

CS 5100  Foundations of Artificial Intelligence

                                                Homework and PA 6

                                    NAME: **PARSHVA TIMBADIA**

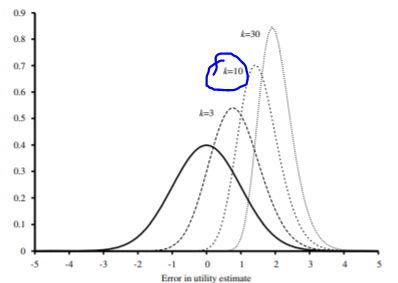
                                    EMAIL: [timbadia.p@northeastern.edu](mailto:timbadia.p@northeastern.edu" \t "https://mail.google.com/mail/u/0/" \l "advanced-search/from=parshwat%40gmail.com&query=parshwat%40gmail.com&isrefinement=true&fromdisplay=parshwat/_blank)

                                    NUID: **001091783**

**Q1**

A>

In this scenario, as we know the number of cars explored by Chris is four and ten by Pat. It is interesting to note that we need the one that has the **maximum expected utility for Pat would be higher since he is exploring more cars than Chris**. However, who gets more **disappointed depends** on the way the expected utility is selected. Like, let's consider that Chris wants to get a car that is black in color and sedan, whereas Pat might want to get a car that is more fuel-efficient and hence expected utility judge the disappointment.



As per the diagram provided in AIMA Fig 16.3.3 we can conclude that the SD of disappointment should range between **1.4 and 1.6.**

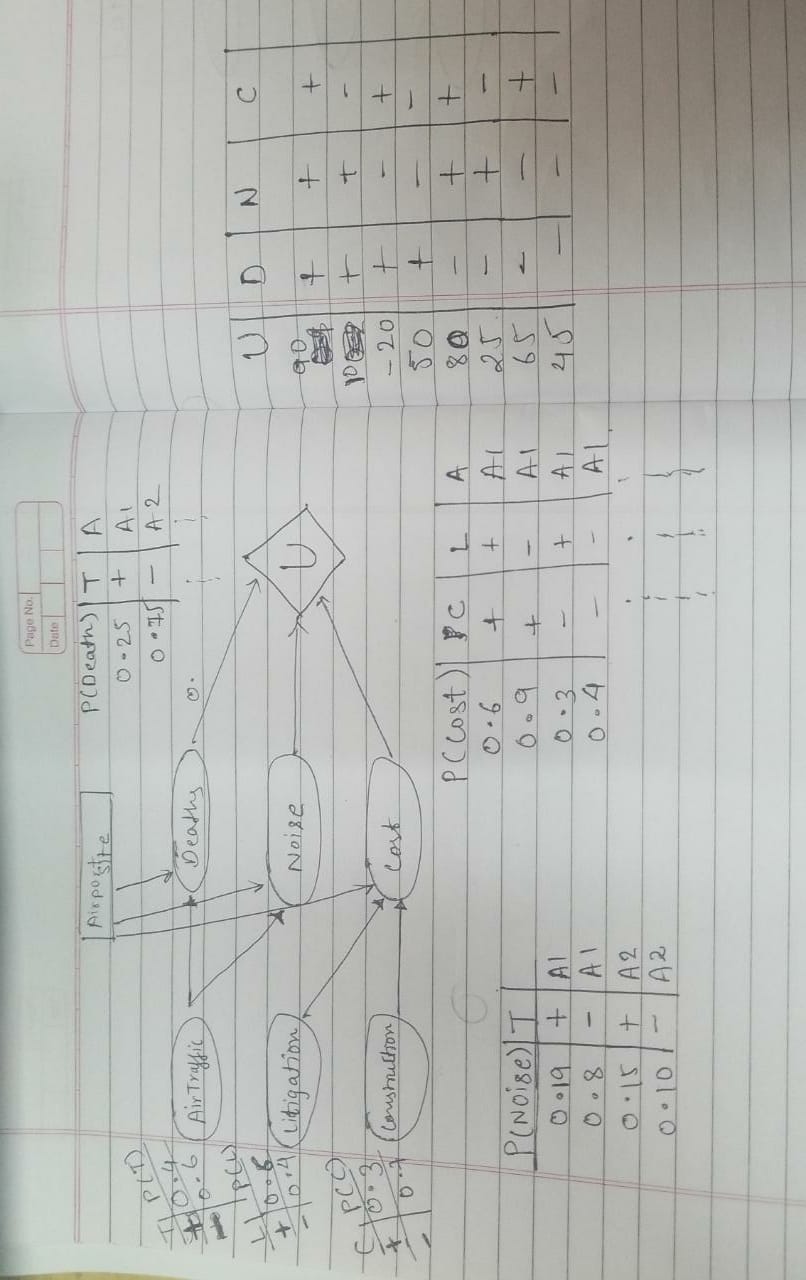
B>

I think **figure(ii)** can be represented only because the graph is totally connected. For instance, it is possible to represent the graph as P(Flavor,Wrapper,Shape), in fact it is possible to represent using any joint probability distribution.

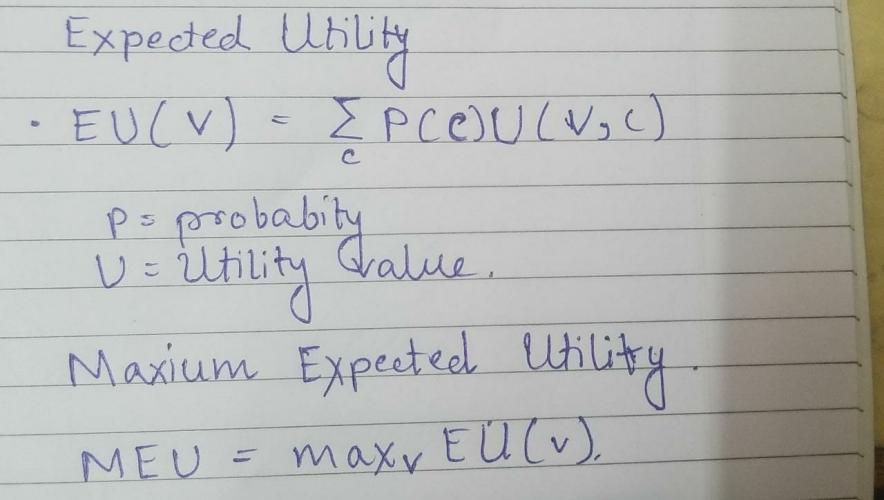
**Figure (iii)** is also possible since is represent the normal Bayes Net. Hence, this can also be represented by P(Flavor,Wrapper,Shape).

**Figure (i) cannot** be correct since it claims Wrapper and Shape to be independent. However, that can not be the case only because of this given statement “80% of the strawberry candies are round and 80% have a red wrapper, while 90% of the anchovy candies are square and 90% have a brown wrapper”. This tells that most of the time wrappers are selected based on the shapes.

**C>**



1. The variable domain can be numeric number in valid range. For the other requirement you may look at the demo diagram shown above.
2. We can calculate expected utilities using the following formula.



**Q2>**

A>

 In the **active learning** the agent’s policy is not fixed and hence the agent needs to make a decision of what should be done further. So in short active learning is to act and keep learning what is the optimal policy. So in our case of moving cart problems, the problem can be solved using the active learning method. This is because let's say in the **first few episodes** the robot will fail terribly since it is still in the learning state and **after few episodes**, it will grasp the idea that the fast movement in one direction will lead to quick failure but the fast continuous movements in both the direction can help the pole to stay up for the longer duration.

The Q-learning representation can be shown as below:

**Input:** Pole Angle

**Output:** Resultant action either left or right

**States:** Based on the possible angles with the help of which it can stay up. It depends on Theta ie. Angle

**Actions:** Left or Right.

Q learning considers the further states and hence helps to get the larger reward. So the agent will learn based on the Q values and improve hence active RL.

**Q(s,a) <- r(s,a) + Ymax(Q(s’,a’))**

s is nothing but the current state of the pole on the cart, actions for our problem are left and right and for correct actions, it gets positive rewards.

B>

Unlike in active learning, agents' policy is fixed in case of **passive learning.** Hence the agent possesses the knowledge of what is to be done exactly. So passive learning actions are sequential since it has a fixed policy for execution.

**Now,** our problem can be designed as per **passive learning** by **fixing the policy** such that the pole remains up and does not fall, in short balancing the pole at each and every action performed by the cart.