Wang Yinan 001530926

1. The monkey-and-bananas problem is faced by a monkey in a laboratory with some bananas hanging out of reach from the ceiling. A box is available that will enable the monkey to reach the bananas if he climbs on it. Initially, the monkey is at A, the bananas at B, and the box at C. The monkey and box have height Low, but if the monkey climbs onto the box he will have height High, the same as the bananas. The actions available to the monkey include Go from one place to another, Push an object from one place to another, ClimbUp onto or ClimbDown from an object, and Grasp or Ungrasp an object. The result of a Grasp is that the monkey holds the object if the monkey and object are in the same place at the same height.
   a) In this problem's context, explain what is meant by Classical Planning, and its 5 key elements (Action Schema, Precondition, Effect etc.) [5 points]
   b) Refer to Figure 10.1 (A PDDL description of an air cargo transportation planning problem) on pg. 369 of Norvig AI textbook. Develop a similar table for the above problem. [10 points]

a) Classical planning is an important field of artificial intelligence research. Its main task is to design the corresponding planning system under the given initial state, executable actions and target conditions, so that the current initial state can reach the state meeting the target conditions by executing the appropriate action sequence.

   For this problem's context, the problem is considering as a classical planning problem with 5 key elements(initial condition, goals, action schema, precondition, effect)

   Initial condition: monkey at A, bananas at B, box at C, both monkey and box are low, bananas are High and A B C is three different point.

   Goal: Monkey has bananas and both them is high, box is low. Monkey, bananas, box are at B and monkey is on the box

   Action: such like climbup to the box and grab the bananas.

   Preconditions: Monkey is on the box and, neither monkey and box are at A or C

   Effect: Monkey grabs bananas

b)

   INIT:
   At(monkey,A) ∧ At(bananas,B) ∧ At(box,C) ∧ Low(monkey) ∧ Low(box) ∧ High(bananas)∧(A,B,C are different)

   GOAL:
   Grab(monkey,bananas) ∧ High(monkey,bananas) ∧ Low(box) ∧ At(monkey,B), At(box,B)∧On(monkey,box)

   ACTION: To(monkey,A,C)
   PRECOND: At(monkey,A)∧(A,B,C are different)
   EFFECT: At(monkey,C)∧not At(monkey,A)

ACTION: Move(monkey,box,B,C)
PRECOND: At(monkey,C)∧At(box,C)∧Low(monkey,box) ∧not At(monkey,C) ∧not At(box,C)
EFFECT: At(monkey,B) ∧ At(box,B) ∧ Low(monkey,box) ∧ not At(monkey,C) ∧not At(box,C)

ACTION: Climb(monkey, box)
PRECOND: At(monkey,B) ∧ At(box,B) ∧ At(bananas,B) ∧ Low(monkey,box,bananas)
EFFECT: High(monkey,bananas)∧not Low(monkey)∧On(monkey,box)

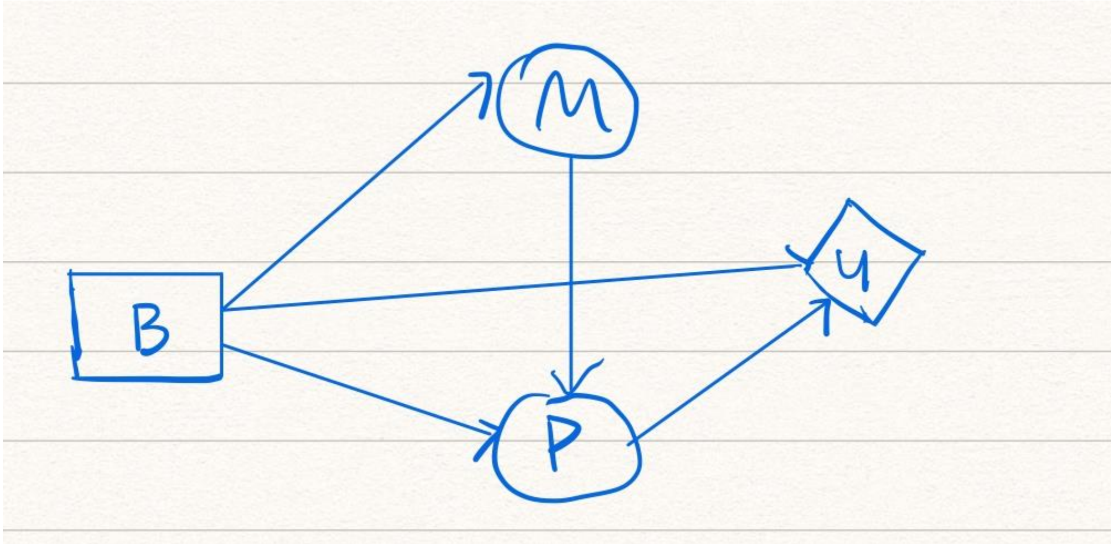
ACTION: Grasp(monkey,bananas)
PRECOND: At(monkey,B)∧At(box,B)∧At(bananas,B)∧high(monkey,bananas) ∧On(monkey,box)
EFFECT: Grab(monkey,bananas) ∧ High(monkey,bananas) ∧ Low(box) ∧ At(monkey,B), At(box,B)∧On(monkey,box)

2. Ch. 16 **Making Simple Decisions**
Consider a student who has the choice to buy or not buy a textbook for a course. We'll model this as a decision problem with one Boolean decision node, B, indicating whether the agent chooses to buy the book, and two Boolean chance nodes, M, indicating whether the student has mastered the material in the book, and P, indicating whether the student passes the course. Of course, there is also a utility node, U. A certain student, Sam, has an additive utility function: 0 for not buying the book and -\$100 for buying it; and \$2000 for passing the course and 0 for not passing. Sam's conditional probability estimates are as follows:

$P(p \mid b, m) = 0.9$ $P(p \mid b, \neg m) = 0.5$	$P(m \mid b) = 0.9$ $P(m \mid \neg b) = 0.7$

$P(p \mid \neg b, m) = 0.8$
$P(p \mid \neg b, \neg m) = 0.3$
You might think that P would be independent of B given M, But this course has an openbook final—so having the book helps.
a. Draw the decision network for this problem. [5 points]
b. Compute the expected utility of buying the book and of not buying it. [10 points]

a)



b) U(buybook)=1620, U(notbuybook)=1300

$$U(b) = \sum P(P|b) \cdot U(P, b)$$
$$= P(P|b) \cdot U(P, b) + P(\neg P|b) \cdot U(\neg P, b)$$
$\because \sum P(P|m) P(m) = P(p)$

$P(P|b) = \sum P(p|m, b) p(m|b) = 0.86.$

$P(\neg P|b) = 1 - 0.86 = 0.14$

$U(P, b) = 2000 - 100 = 1900,$

$U(P|\neg b) = 0 - 100 = -100$

$\therefore U(b) = 0.86 \times 1900 + 0.14 \times (-100) = 1620$

$$U(\neg b) = \sum P(P|\neg b) \cdot U(P, \neg b)$$
$$= P(P|\neg b) \cdot U(P, \neg b) + P(\neg P|\neg b) \cdot U(\neg P, \neg b)$$
$\because \sum P(P|m) P(m) = P(p)$

$P(P|\neg b) = \sum P(p|m, \neg b) p(m|\neg b) = 0.65.$

$P(\neg P|\neg b) = 1 - 0.65 = 0.35$

$U(P, b) = 2000 - 0 = 2000$

$U(P|\neg b) = 0 - 0 = 0$

$\therefore U(b) = 0.65 \times 2000 + 0.35 \times 0 = 1300$

3. Given in Table 1 below is Data on customers who purchased a PC at a computer store, showing age group, income, education and whether they purchased a PC earlier.

(A) What is **Naïve Bayes** algorithm? What is it used for in Machine Learning? [3 Points]

(B) For the same PC purchase dataset given in Table 1 below, develop a Naïve Bayes Classification method, and use the same to classify the new example given below. [7 Points]

Senior	High	High School	Yes		?

Clearly show all of your calculations, step by step.

Table 1. Data on customers who purchased a PC

Age	Income	Education	Prior Purchase	Buys PC?
Young adult	High	College	Yes	Yes
Middle Age	Low	College	No	No
Middle Age	High	High School	Yes	No
Senior	Low	College	No	No
Young adult	Medium	High School	Yes	Yes
Senior	High	College	yes	Yes
Middle Age	Medium	College	No	No
Young adult	Low	High School	No	No
Middle Age	Low	College	Yes	Yes
Young adult	High	College	No	Yes
Senior	Medium	High School	No	No

(A) Bayesian method is based on Bayesian principle and uses the knowledge of probability and statistics to classify the sample data set. Naive Bayesian method is simplified based on Bayesian algorithm, that is, assuming that the attributes are conditionally independent of each other when the target value is given.

The formula is:

$$P(B|A) = \frac{P(A|B)\,P(B)}{P(A)}$$

Naive Bayesian algorithm plays an important role in classification. An unknown text or image can be classified according to its existing classification rules, and finally achieve the purpose of classification.

(B) Assume pp represent the prior purchase, we can get:

P(pp-yes)=5/11

P(pp-no)=6/11

And pc represent the have pc, we can get:

	Yes(pc)	No(pc)
Yes(pp)	4	1
no(pp)	1	5

So the probility:

P(pp-y | pc-y)=4/5 P(pp-y | pc-n)=1/6
P(pp-n | pc-y)=1/5 P(pp-n | pc-n)=5/6

And then, for Education:
P(H)=4/11
P(C)=7/11

	Yes(pc)	No(pc)
Hich school	1	3
College	4	3

P(H | pc-y)=1/5 P(H | pc-n)=3/6
P(C | pc-y)=4/5 P(C | pc-n=3/6

For Income:
P(HI)=4/11
P(M)=3/11
P(L)=4/11

	Yes(pc)	No(pc)
High Income	3	1
Medium	1	2
Low	1	3

P(HI | pc-y)=3/5 P(HI | pc-n)=1/6
P(M | pc-y)=1/5 P(M | pc-n)=2/6
P(L | pc-y)=1/5 P(L | pc-n)=3/6

For Age:
P(Y)=4/11
P(MA)=4/11
P(S)=3/11

	Yes(pc)	No(pc)
Young	3	1
Middle Age	1	3
Senior	1	2

P(Y | pc-y)=3/5 P(Y | pc-n)=1/6
P(MA | pc-y)=1/5 P(MA | pc-n)=3/6
P(S | pc-y)=1/5 P(S | pc-n)=2/6

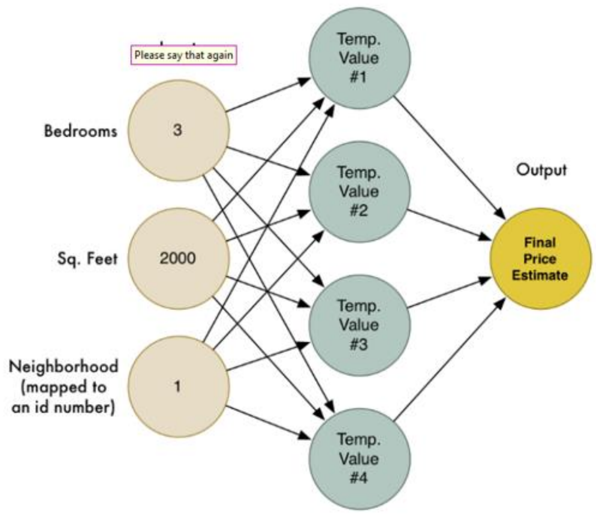
Then we have:
P(Senior,High Income,High School,Prior | pc-y)=1/5×1/5×3/5×4/5=0.0192
P(Senior,High Income,High School,Prior | pc-n)=2/6×3/6×1/6×1/6=0.00463
P(Senior,High Income,High School,Prior)=0.0192×5/11+0.00463×6/11=0.01125

P(pc-y | Senior,High Income,High School,Prior)= （0.0192×5/11） 0.01125=0.775

P(pc-n | Senior,High Income,High School,Prior)= （0.00463×6/11） 0.01125=0.225

Because P(pc-y | Senior,High Income,High School,Prior) is larger, so Naïve Bayes Classifier will output yes.

4. Given the below neural network for home price prediction: [15 Points]



Weights w on ALL of the forward arrows from every neuron is equal to 0.1.

Describe the forward propagation algorithm, and show an example calculation of a final price estimate.

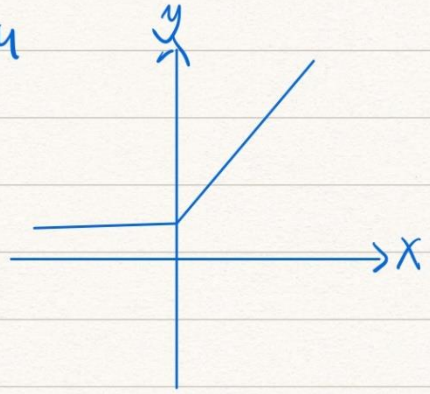
Forward propagation algorithms:

Suppose that some nodes such as nodes I, J, K,... Of the previous layer are connected with node w of this layer, how to calculate the value of node w? The weighted sum operation is carried out through the I, J, K and other nodes of the upper layer and the corresponding connection weights. The final result is added with an offset term (omitted in the figure for simplicity). Finally, through a nonlinear function (i.e. activation function), such as relu, sigmoid and other functions, the final result is the output of node w of this layer. Finally, through this method, the output layer results are obtained.

Example:

Use Relu as activation function.

ReLu



$h = \max(\sigma, z)$

$\because$ all the weights $= 0.1$.

$\therefore Z_x = W_x^T X$

$Z_1 = 0.1 \times (200 + 3 + 1) = 200.4$

$Z_2 = 0.1 \times (3 + 2000 + 1) = 200.4$ 1st layer.

$Z_3 = 200.4$ $h_{1,2,3,4} = 200.4$

$Z_4 = 200.4$

$Z_5 = 0.1 \times (200.4 \times 4) = 80.16$ output layer

For output layer, Linear activation function is used.

$\therefore \sigma = Z_5 = 80.16$

$\therefore$ The final price is 80.16

5. Computer Vision [15 Points]

Consider the image below: **A couple with a cat**

Develop a series of algorithms (i. e., a design) for face recognition for this object – your algorithm needs to recognize faces of people and animals, provide a count of them and identify the animal.

This algorithm starts with the recognition of facial features, usually giving priority to the recognition of vally region, that is, eyes, and then recognizing the facial features of humans and animals. And extract features, mark the extracted features, detect the number of faces, and distinguish faces into human and animal.

6. From Chapter 25, use Execice 25.7 as the basis for this question. Our goal is for the robotic arm (Fig 25.32 on page 1017) to start in Position (a) and reach position (c) for Picking an object – say a red Cube. Identify 5 algorihms/methods to accomplish the needed operations to do this: Perception, Planning to Move and Moving. Describe each of them in suficient detail (1-2 small paragraphs, diagrams to explain). [15 Points]

Perception: the robot's sensors could gain data from environment. And this data could be used to model the behavior of robots. One of the important things to do is localization. These algorithms could be used:

1. Monte Carlo Localization

This is a general term of a class of random algorithms. Its idea is to replace the "probability" of events with the "frequency" of events. Further, we may not know the probability of an event accurately. In this case, we can use multiple sampling and use the frequency of the event to replace its probability.

$$V_\pi(S) \approx average(G_t), \quad s.t. \quad S_t = S$$

Features:

① Approximate results can be obtained by random sampling

② The more sampling times, the closer the estimation result to the real value.

Based on the interaction between agent and environment, we can get the trajectory sequence, and then, based on the sampled sequence, we can calculate the value corresponding to the corresponding state

2. Kalman Filtering:

As long as it is a dynamic system with uncertain information, Kalman filter can make informed speculation about what the system will do next. Even if there is noise information interference, Kalman filter can usually find out what happened and find the imperceptible correlation between images.

Features:

①The information needs to be modeled as Gaussian distribution and is only suitable for linear systems

②Small memory consumption (only the previous state needs to be retained) and high speed

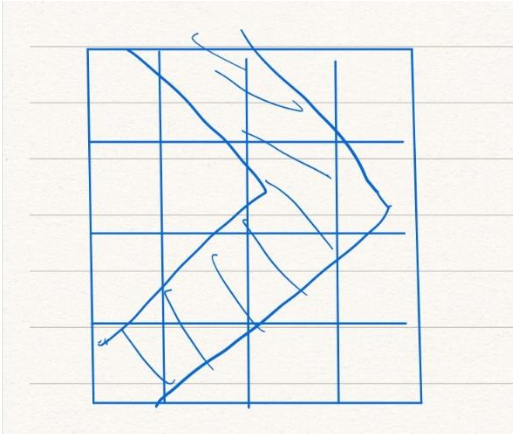
Planning to move: planning a route to move from one place or position to another place or position

3. Cell Decomposition

The decomposition of free space into a finite number of adjacent regions, called units. The path planning problem in a single region can be solved in a simple way, turning the path planning problem into a discrete graph search problem

Features:

The path can be found using the deterministic form of the value-Iteration algorithm, or the A* algorithm

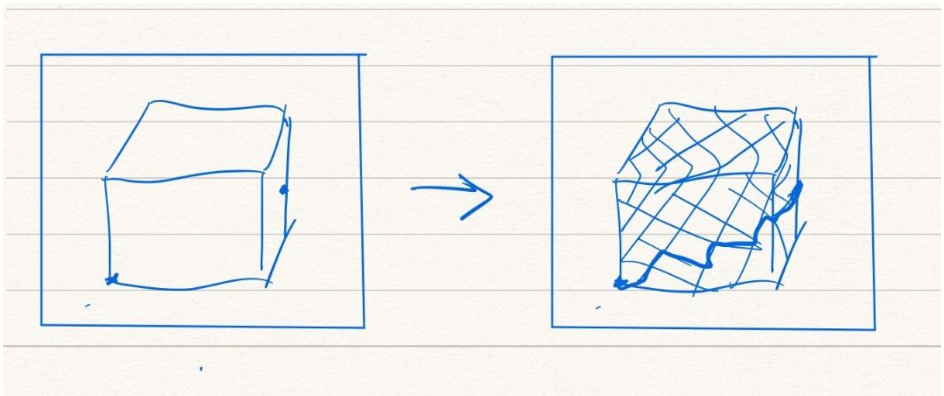


4. Skeletonization

Changing the free space of the robot into a one-dimensional representation makes the problem easier to solve.

Features:

It can be difficult to calculate, especially in space where the shape of the obstacle is very complex.



Moving

After planning, the robot is going to move

5. Reactive control

It is often difficult to obtain a sufficiently accurate environmental model, and errors will invalidate algorithms that require accuracy. Therefore, reactive control algorithms should be used. When the movement is blocked, use some very simple control rules to control the robot's movement, such as when it encounters an obstacle, pull the leg back in and raise it to try again.

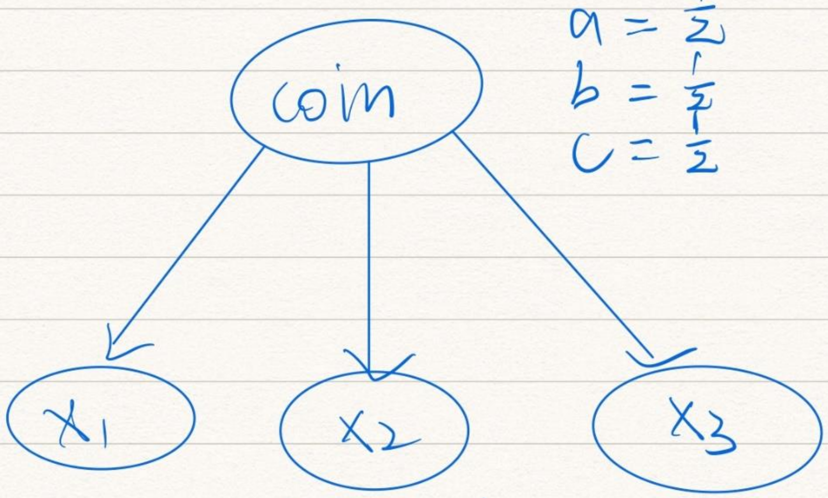
Features

As a finite state machine, the controller constitutes a reflective agent with states

7. We have a bag of three biased coins a, b, and c with probabilities of coming up heads of 30%, 50%, and 70%, respectively. One coin is drawn randomly from the bag (with equal likelihood of drawing each of the three coins), and then the coin is flipped three times to generate the outcomes X1, X2, and X3. [15 Points]
a. Draw the Bayesian network corresponding to this setup and define the necessary CPTs.
b. Calculate which coin was most likely to have been drawn from the bag if the observed flips come out heads twice and tails once.

a.

$a = \frac{1}{2}$
$b = \frac{1}{2}$
$c = \frac{1}{2}$



	a	b	c
up	$\frac{1}{5}$	$\frac{1}{3}$	$\frac{7}{15}$
down	$\frac{7}{15}$	$\frac{1}{3}$	$\frac{1}{5}$

$$P(up) = P(up|a) \cdot P(a) + P(up|b) \cdot P(b) + P(up|c) \cdot P(c)$$
$$= \frac{1}{2}$$

$$P(a|up) = P(a) \cdot \frac{P(up|a)}{P(up)} = \frac{1}{5}$$

$$P(b|up) = \frac{1}{3}$$
$$P(c|up) = \frac{7}{15}$$

c

b.

$\because P(up) = P(up|a) \cdot P(a) + P(up|b) \cdot P(b)$
$+ P(up|c) \cdot P(c)$
$= \frac{1}{2}$

2 up, 1 down $\rightarrow P(up|X) = \frac{2}{3}$

$\therefore P(up|X) = P(up) \cdot \frac{P(X|up)}{P(X)}$. $P(X) = \frac{1}{3}$.

$\therefore$

$$\frac{2}{3} = \frac{1}{2} \cdot \frac{P(X|up)}{\frac{1}{3}}$$

$\therefore P(X|up) = \frac{4}{9}$

$\therefore$ coin C is most likely to be drawn.