

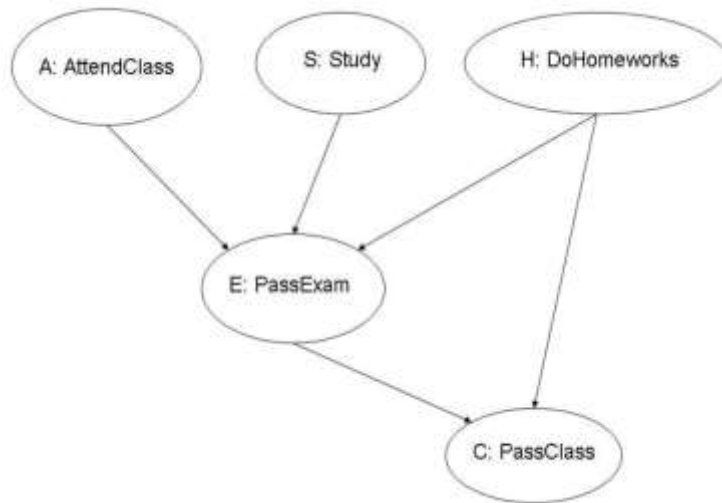
3. [15%] Bayes Network.

Suppose that a student PassExam(E), could be caused by AttendClass(A), Study(S), DoHomeworks(H). PassClass(C) could be caused by PassExam(E) and DoHomeworks(H).

P(A)	
+a	0.5
-a	0.5

P(S)	
+s	0.7
-s	0.3

P(H)	
+h	0.9
-h	0.1



P(C E,H)			
-e	-h	+c	0.1
-e	-h	-c	0.9
-e	+h	+c	0.4
-e	+h	-c	0.6
+e	-h	+c	0.3
+e	-h	-c	0.7
+e	+h	+c	0.9
+e	+h	-c	0.1

P(E A,S,H)				
-a	-s	-h	+e	0.2
-a	-s	-h	-e	0.8
-a	-s	+h	+e	0.5
-a	-s	+h	-e	0.5
-a	+s	-h	+e	0.4
-a	+s	-h	-e	0.6
-a	+s	+h	+e	0.8
-a	+s	+h	-e	0.2
+a	-s	-h	+e	0.3
+a	-s	-h	-e	0.7
+a	-s	+h	+e	0.7
+a	-s	+h	-e	0.3
+a	+s	-h	+e	0.6
+a	+s	-h	-e	0.4
+a	+s	+h	+e	0.9
+a	+s	+h	-e	0.1

3A).[3%]Compute the following entry from joint distribution: $P(+a,+s,+h,+e,+c)$?

$$P(+a) * P(+s) * P(+h) * (+e|+a,+s,+h) * P(+c|+e,+h)$$

As long as the formula is correct, give full credit

3B).[3%] What is the probability of passing the class, given that you attend class and study, but don't do the homeworks.

$$P(+c|+a,+s,-h) = \frac{P(+c,+a,+s,-h)}{P(+a,+s,-h)} = (1)$$

$$\frac{\sum_e P(+c,+a,+s,e,-h)}{\sum_{e,c} P(+c,+a,+s,e,-h)} = (1)$$

$$\frac{\sum_e P(+a) * P(+s) * P(-h) * P(e|+a,+s,-h) * P(+c|e,-h)}{\sum_{e,c} P(+a) * P(+s) * P(-h) * P(e|+a,+s,-h) * P(+c|e,-h)} = (2)$$

$$\frac{P(+a)*P(+s)*P(-h)*\sum_e P(e|+a,+s,-h)*P(+c|e,-h)}{P(+a)*P(+s)*P(-h)*\sum_e P(e|+a,+s,-h)*\sum_c P(+c|e,-h)} = (2)$$

$$\sum_e P(e|+a,+s,-h) * P(+c|e,-h) (3)$$

if student did not have (2) and (3), but have (1), give 1 pt. Either (2) and (3), give full

As long as the formula is correct, give full credit

3C).[3%] Compute $P(+a|+c,+h)$

$$P(+a|+c,+h) = \frac{P(+c,+a,+h)}{P(+c,+h)} = (1)$$

$$\frac{\sum_{e,s} P(+c,+a,s,e,+h)}{\sum_{a,e,c} P(+c,+a,s,e,+h)} = (1)$$

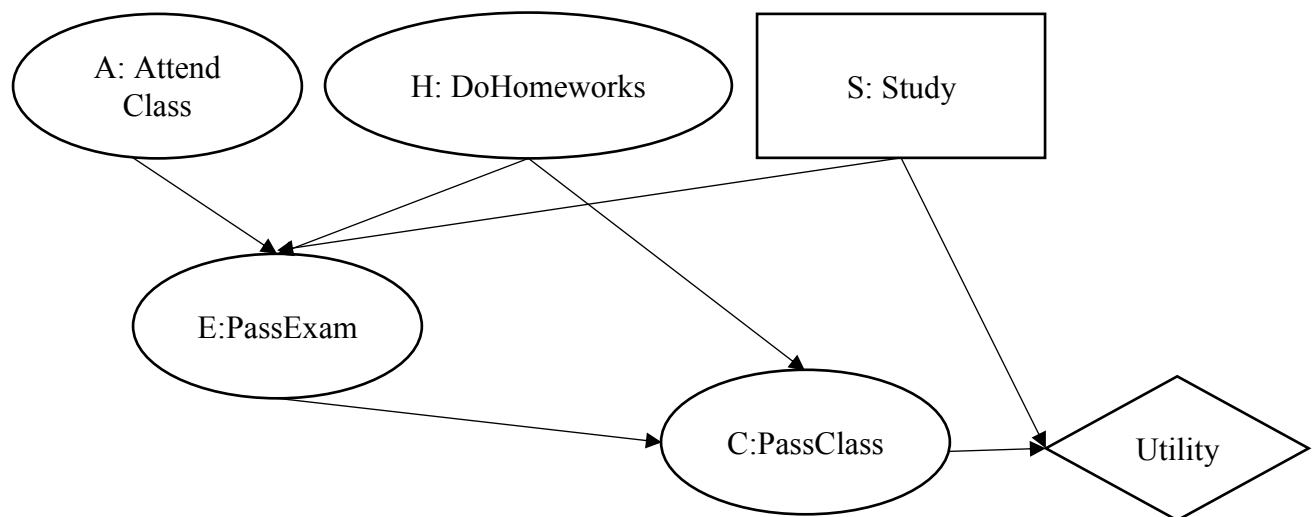
$$\frac{\sum_{e,s} P(+a) * P(+s) * P(+h) * P(e|+a,s,+h) * P(+c|e,+h)}{\sum_{a,e,s} P(+a) * P(+s) * P(+h) * P(e|+a,s,+h) * P(+c|e,+h)} = (2)$$

$$\frac{P(+a)*P(+h)*\sum_s P(+s)*\sum_e P(e|+a,s,+h)*P(+c|e,+h)}{P(+h)*\sum_a P(+a)*\sum_s P(+s)*\sum_e P(e|+a,s,+h)*P(+c|e,+h)} (3)$$

if student did not have (2) and (3), but have (1), give 1 pt. Either (2) and (3), give full

As long as the formula is correct, give full credit

3D. [6%] Now, consider a student who has the choice to Study(S) or not Study(S) for passing Exam. We'll model this as a decision problem with one Boolean decision node, S, indicating whether the agent chooses to study. The rest are chance nodes. The probabilities are the same as above. There is also a utility node U. We have the following utility function: The utility for PassClass is 100 and the utility for Not PassClass is -10. The utility for Study is -50 and the utility for Not Study is 0. Calculate the MEU for decision node S.



When decision is +s, the probability to passclass is:

$$P(+c|+s) = P(+c,+s) = \sum_{a,h,e} P(a,+s,h,e,+c) = \sum_{a,h,e} P(a) \times P(+s) \times P(h) \times P(e|a,+s,h) \times P(+c|e,h)$$

Where, $P(+s) = 1$

1pt

When decision is +s, the probability to not passclass is:

$$P(-c|+s) = P(-c,+s) = \sum_{a,h,e} P(a,+s,h,e,-c) = \sum_{a,h,e} P(a) \times P(+s) \times P(h) \times P(e|a,+s,h) \times P(-c|e,h)$$

where, $P(+s) = 1$

1pt

When decision is -s, the probability to passclass is:

$$P(+c|-s) = P(+c,-s) = \sum_{a,h,e} P(a,-s,h,e,+c) = \sum_{a,h,e} P(a) \times P(-s) \times P(h) \times P(e|a,-s,h) \times P(+c|e,h)$$

where, $P(-s) = 1$

1pt

When decision is -s , the probability to not passclass is:

$$P(-c|-s) = P(-c,-s) = \sum_{a,h,e} P(a, -s, h, e, -c) = \sum_{a,h,e} P(a) \times P(-s) \times P(h) \times P(e|a, -s, h) \times P(-c|e, h)$$

where, $P(-s) = 1$

1pt

Expected Utility when study: $P(+c|+s) \times (100-50) + P(-c|+s) \times (-10-50)$

1pt

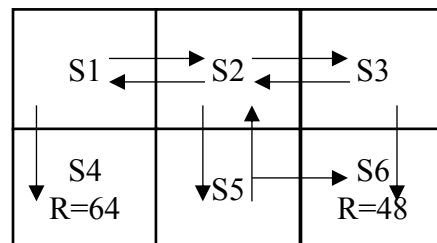
Expected Utility when not study: $P(+c|-s) \times (100+0) + P(-c|-s) \times (-10+0)$

1pt

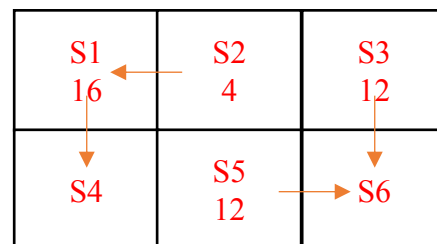
As long as the formula is correct, give full credit

4. [20%] Markov Decision Process

Consider the 6-state Markov Decision process below. The goals with rewards are in state S4 and S6. At each state, the possible transitions are **deterministic** and indicated by the arrows. You get a reward of $R_4=64$ if you get to the goal S4 and a reward of $R_6=48$ if you get to the goal S6.



4A[4%] Consider a discount factor of $\gamma = \frac{1}{4}$. On the figure below, show the optimal value V^* for each state and the arrows corresponding to the set of the optimal actions.



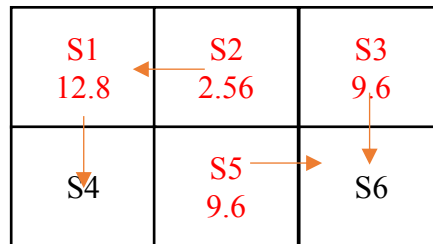
Each correct value and arrow 0.5pt.

4B.[5%] What values of γ would result in a different optimal action in S3? Indicate which policy action changes

$$64 * \gamma * \gamma * \gamma \geq 48 * \gamma$$

$$\gamma \geq \sqrt[2]{\frac{3}{4}}$$

4C.[8%] Suppose the action is no longer deterministic. The discount factor is still $\gamma = \frac{1}{4}$. Each action has a failure probability $f=0.25$, i.e. the action is now stochastic. Once an action is taken, with probability 0.25, it would stay in the original grid. On the figure below, show the optimal value V^* for each state and the arrows corresponding to the set of the optimal actions.



$$V(S1)^* = 0.25 * 0.25 * V(S1)^* + 0.75 * 0.25 * 64$$

$$V(S1)^* = 12 * 16 / 15 = 12.8$$

$$V(S3)^* = 0.25 * 0.25 * V(S3)^* + 0.75 * 0.25 * 48$$

$$V(S3)^* = 9 * 16 / 15 = 9.6$$

$$V(S5)^* = 0.25 * 0.25 * V(S5)^* + 0.75 * 0.25 * 48$$

$$V(S5)^* = 9 * 16 / 15 = 9.6$$

$$V(S2)^* = 0.25 * 0.25 * V(S2)^* + 0.75 * 0.25 * 12.8$$

$$V(S2)^* = 2.4 * 16 / 15 = 2.56$$

Each correct value and arrow 1pt.

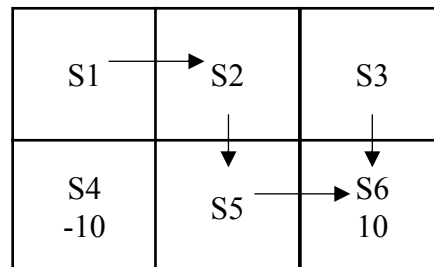
4D. [3%] In this scenario, the discount factor is $\gamma = 1$ and the learning rate $\alpha = 0.5$. S6 has the reward 10 and S4 has the reward -10. All other states have 0 rewards. Now we no longer know the details of the transition probabilities ahead of time. We must instead use reinforcement learning to compute the necessary values. We have the following sequence of actions. After 3 example runs below, what would be the Q-value of (S5, Right)? Remember that $Q(s, a)$ is initialized with 0 for all (s, a) .

$$Q(s,a) \leftarrow Q(s,a) + \alpha(R(s) + \gamma \max_{a'} Q(s',a') - Q(s,a))$$

Example 1			
s	a	s'	r
S1	Right	S2	0
S2	Down	S5	0
S5	Right	S6	0
S6	Exit	Terminal	10

Example 2			
s	a	s'	r
S1	Right	S2	0
S2	Down	S5	0
S5	Right	S6	0
S6	Exit	Terminal	10

Example 3			
s	a	s'	r
S1	Right	S2	0
S2	Down	S5	0
S5	Right	S4	0
S4	Exit	Terminal	-10



The first iteration, $Q(S5, \text{Right}) = 0$.

Second iteration, $Q(S5, \text{Right}) = Q(S5, \text{Right}) + 0.5(0 + 1 * 10 - 0) = 5$

Third iteration, $Q(S5, \text{Right}) = 5 + 0.5(0 + 1 * 10 - 5) = 7.5$

As long as students could give 7,5, give full credit. If not, give 1pt for each iteration.

5. [15%] Decision Trees

Assume we want to train a machine to decide whether someone gets an Uber or just walks based on weather information and the length of the path. Our training data is provided below. The weather outlook can be either sunny or cloudy or rainy. The humidity level can be either high or normal. The destination is either near or far.

Trip	Outlook	Humidity	Destination	TakeUber?
T1	s	h	n	y
T2	s	h	f	y
T3	c	h	n	n
T4	r	h	n	n
T5	r	n	n	n
T6	r	n	f	n
T7	c	n	f	n
T8	s	h	n	y
T9	s	n	n	n
T10	r	n	n	n
T11	s	n	f	n
T12	c	h	f	n
T13	c	n	n	n
T14	r	h	f	y
T15	r	n	n	n
T16	c	h	f	n

5A. [6%] Which attribute would be at the root of the tree? Give reason for your answer in terms of entropy and information gain (no need for exact calculation.)

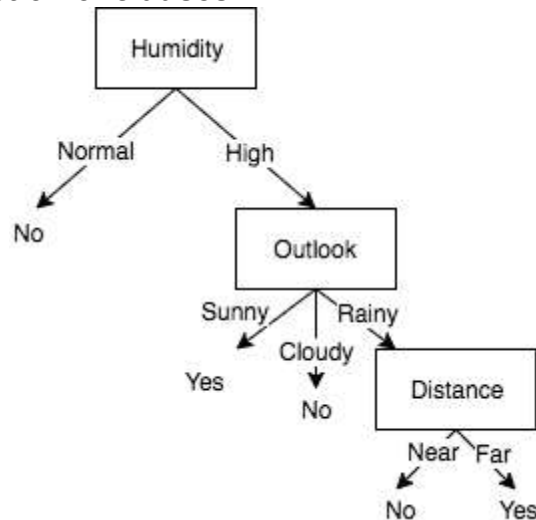
Remainder(Outlook) = $5/16 (-(\frac{3}{5} \log \frac{3}{5} + \frac{2}{5} \log \frac{2}{5})) + 5/16 (0) + 6/16 (-(\frac{1}{2} \log \frac{1}{2} + \frac{1}{2} \log \frac{1}{2}))$

Remainder(Humidity) = $\frac{1}{2} (1) + \frac{1}{2} (0) \Rightarrow$ least remainder, highest information gain \Rightarrow root attribute

Remainder(Destination) = $9/16 (-(\frac{2}{9} \log \frac{2}{9} + \frac{7}{9} \log \frac{7}{9})) + 7/16 (-(\frac{2}{7} \log \frac{2}{7} + \frac{5}{7} \log \frac{5}{7}))$

[1% for each correct expression of the remainder of an attribute, 3% for determining the root attribute]

5B. [9%] Outline the learned decision tree and write down the learned concept for TakeUber as a disjunction of clauses.



TakeUber = (HighHumidity \wedge Sunny) \vee (HighHumidity \wedge Rainy \wedge Far)

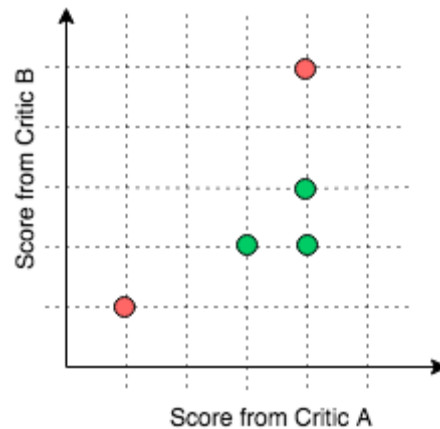
[2% for each correct level of the tree, 3% for the learned concept, no partial credit]

6. [10%] Neural Nets

Assume you want to predict how well a show will do with the general audience, based on the scores of two critics who score the show on the scale of 1 to 5. Here are five data points from some previous shows, including the critics' scores and the performance of the shows:

Show	Critic A Score	Critic B Score	Did the audience like the show?
1	1	1	No
2	3	2	Yes
3	4	5	No
4	4	3	Yes
5	4	2	Yes

6A. [2%] Determine if the data is linearly separable by plotting it on the 2D plane below.

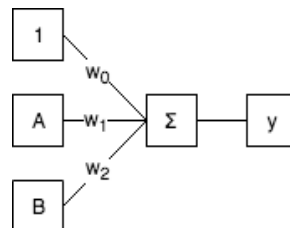


[Green dots should be x's, red dots should be o's.]

The data is linearly separable.

[1% for the complete plot, 1% for determining linear separability, no partial credit]

6B. [8%] After establishing linear separability, you decide to use a perceptron to classify the data, using the scores as features. So the perceptron would be like this (If $w_0 + w_1 \cdot A + w_2 \cdot B > 0$, the audience will like the show ($y = 1$); otherwise they won't ($y = -1$)).:



This is how perceptron update works:

Start with an initial vector of weights.

For each training instance (y is the output and y^* is the expected result):

If correct ($y = y^*$), no update is needed.

If wrong: update the weight vector by adding or subtracting the input vector. Subtract if y^* is -1.

So for example, if at a training iteration the input is (1, 2, 1), the weight vector is (1, 2, 3), the output is 1, and the expected result is -1, the updated weight vector will be: (0, 0, 2).

Assume we start with the weights $\{w_0: 0, w_1: 0, w_2: 0\}$ (weight vector (0, 0, 0)).

Determine the weights after two updates. Calculate the Accuracy of the perceptron on the data after these two updates.

After first update: $\{w_0: 1, w_1: 3, w_2: 2\}$

After second update: $\{w_0: 0, w_1: -1, w_2: -3\}$

Accuracy: 40%

[3% for each correct update, 2% for correct accuracy, no partial credit]