

(b) . [7.5 pts] Suppose that X_1, \dots, X_m are categorical input attributes and Y is categorical output attribute. Suppose we plan to learn a decision tree without pruning, using the standard algorithm.

b.1 (**True or False -1.5 pts**) : If X_i and Y are independent in the distribution that generated this dataset, then X_i will not appear in the decision tree.

Answer: False (because the attribute may become relevant further down the tree when the records are restricted to some value of another attribute) (e.g. XOR)

b.2 (**True or False -1.5 pts**) : If $IG(Y|X_i) = 0$ according to the values of entropy and conditional entropy computed from the data, then X_i will not appear in the decision tree.

Answer: False for same reason

b.3 (**True or False -1.5 pts**) : The maximum depth of the decision tree must be less than $m+1$.

Answer: True because the attributes are categorical and can each be split only once

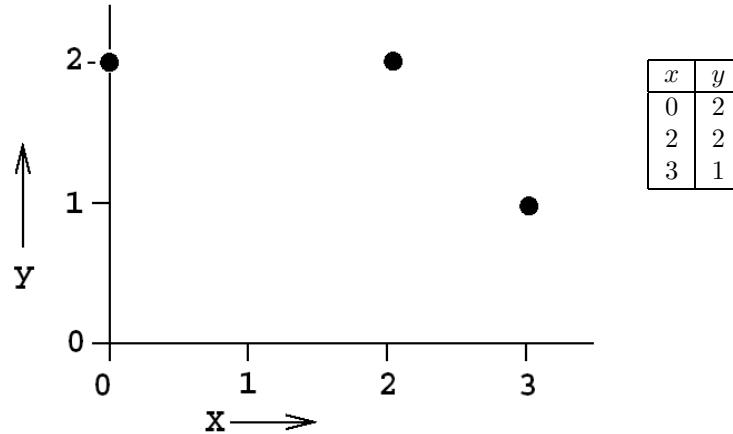
b.4 (**True or False -1.5 pts**) : Suppose data has R records, the maximum depth of the decision tree must be less than $1 + \log_2 R$

Answer: False because the tree may be unbalanced

b.5 (**True or False -1.5 pts**) : Suppose one of the attributes has R distinct values, and it has a unique value in each record. Then the decision tree will certainly have depth 0 or 1 (i.e. will be a single node, or else a root node directly connected to a set of leaves)

Answer: True because that attribute will have perfect information gain. If an attribute has perfect information gain it must split the records into "pure" buckets which can be split no more.

(c) [5 pts] Suppose you have this data set with one real-valued input and one real-valued output:



(c.1) What is the mean squared leave one out cross validation error of using linear regression ? (i.e. the model is $y = \beta_0 + \beta_1 x + \text{noise}$)

Answer: $\frac{2^2 + (2/3)^2 + 1^2}{3} = 49/27$

(c.2) Suppose we use a trivial algorithm of predicting a constant $y = c$. What is the mean squared leave one out error in this case? (Assume c is learned from the non-left-out data points.)

Answer: $\frac{0.5^2 + 0.5^2 + 1^2}{3} = 1/2$

Problem 2. Bayes Rule and Bayes Classifiers (12 points)

Suppose you are given the following set of data with three Boolean input variables a, b , and c , and a single Boolean output variable K .

a	b	c	K
1	0	1	1
1	1	1	1
0	1	1	0
1	1	0	0
1	0	1	0
0	0	0	1
0	0	0	1
0	0	1	0

For parts (a) and (b), assume we are using a naive Bayes classifier to predict the value of K from the values of the other variables.

(a) [1.5 pts] According to the naive Bayes classifier, what is $P(K = 1 | a = 1 \wedge b = 1 \wedge c = 0)$?

Answer: $1/2$.

$$\begin{aligned}
 P(K = 1 | a = 1 \wedge b = 1 \wedge c = 0) &= P(K = 1 \wedge a = 1 \wedge b = 1 \wedge c = 0) / P(a = 1 \wedge b = 1 \wedge c = 0) \\
 &= P(K = 1) \cdot P(a = 1 | K = 1) \cdot P(b = 1 | K = 1) \cdot P(c = 0 | K = 1) / \\
 &\quad P(a = 1 \wedge b = 1 \wedge c = 0 \wedge K = 1) + P(a = 1 \wedge b = 1 \wedge c = 0 \wedge K = 0).
 \end{aligned}$$

(b) [1.5 pts] According to the naive Bayes classifier, what is $P(K = 0 | a = 1 \wedge b = 1)$?

Answer: $2/3$.

$$\begin{aligned}
 P(K = 0 | a = 1 \wedge b = 1) &= P(K = 0 \wedge a = 1 \wedge b = 1) / P(a = 1 \wedge b = 1) \\
 &= P(K = 0) \cdot P(a = 1 | K = 0) \cdot P(b = 1 | K = 0) / \\
 &\quad P(a = 1 \wedge b = 1 \wedge K = 1) + P(a = 1 \wedge b = 1 \wedge K = 0).
 \end{aligned}$$

Now, suppose we are using a joint Bayes classifier to predict the value of K from the values of the other variables.

- (c) [1.5 pts] According to the joint Bayes classifier, what is $P(K = 1|a = 1 \wedge b = 1 \wedge c = 0)$?

Answer: 0.

Let $\text{num}(X)$ be the number of records in our data matching X . Then we have $P(K = 1|a = 1 \wedge b = 1 \wedge c = 0) = \text{num}(K$

- (d) [1.5 pts] According to the joint Bayes classifier, what is $P(K = 0|a = 1 \wedge b = 1)$?

Answer: $1/2$.

$P(K = 0|a = 1 \wedge b = 1) = \text{num}(K = 0 \wedge a = 1 \wedge b = 1) / \text{num}(a = 1 \wedge b = 1) = 1/2$.

In an unrelated example, imagine we have three variables X, Y , and Z .

- (e) [2 pts] Imagine I tell you the following:

$$\begin{aligned}P(Z|X) &= 0.7 \\P(Z|Y) &= 0.4\end{aligned}$$

Do you have enough information to compute $P(Z|X \wedge Y)$? If not, write “not enough info”. If so, compute the value of $P(Z|X \wedge Y)$ from the above information.

Answer: Not enough info.

- (f) [2 pts] Instead, imagine I tell you the following:

$$\begin{aligned}P(Z|X) &= 0.7 \\P(Z|Y) &= 0.4 \\P(X) &= 0.3 \\P(Y) &= 0.5\end{aligned}$$

Do you now have enough information to compute $P(Z|X \wedge Y)$? If not, write “not enough info”. If so, compute the value of $P(Z|X \wedge Y)$ from the above information.

Answer: Not enough info.

- (g) [2 pts] Instead, imagine I tell you the following (falsifying my earlier statements):

$$\begin{aligned}P(Z \wedge X) &= 0.2 \\P(X) &= 0.3 \\P(Y) &= 1\end{aligned}$$

Do you now have enough information to compute $P(Z|X \wedge Y)$? If not, write “not enough info”. If so, compute the value of $P(Z|X \wedge Y)$ from the above information.

Answer: $2/3$.

$P(Z|X \wedge Y) = P(Z|X)$ since $P(Y) = 1$. In this case, $P(Z|X \wedge Y) = P(Z \wedge X) / P(X) = 0.2 / 0.3 = 2/3$.

Problem 3. SVM (9 points)

(a) (**True/False** - 1 pt) Support vector machines, like logistic regression models, give a probability distribution over the possible labels given an input example.

Answer: False

(b) (**True/False** - 1 pt) We would expect the support vectors to remain the same in general as we move from a linear kernel to higher order polynomial kernels.

Answer: False (There are no guarantees that the support vectors remain the same. The feature vectors corresponding to polynomial kernels are non-linear functions of the original input vectors and thus the support points for maximum margin separation in the feature space can be quite different.)

(c) (**True/False** - 1 pt) The maximum margin decision boundaries that support vector machines construct have the lowest generalization error among all linear classifiers.

Answer: False (The maximum margin hyperplane is often a reasonable choice but it is by no means optimal in all cases.)

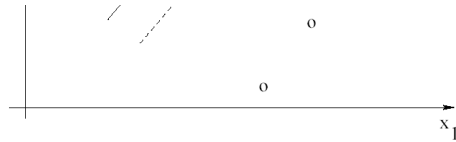
(d) (**True/False** - 1 pt) Any decision boundary that we get from a generative model with class-conditional Gaussian distributions could in principle be reproduced with an SVM and a polynomial kernel of degree less than or equal to three.

Answer: True (A polynomial kernel of degree two suffices to represent any quadratic decision boundary such as the one from the generative model in question.)

(e) (**True/False** - 1 pts) The values of the margins obtained by two different kernels $K_1(x, x_0)$ and $K_2(x, x_0)$ on the same training set do not tell us which classifier will perform better on the test set.

Answer: True (We need to normalize the margin for it to be meaningful. For example, a simple scaling of the feature vectors would lead to a larger margin. Such a scaling does not change the decision boundary, however, and so the larger margin cannot directly inform us about generalization.)

(f) (2 pts) What is the leave-one-out cross-validation error estimate for maximum margin separation in the following figure ? (we are asking for a number)



Answer: 0

Based on the figure we can see that removing any single point would not change the resulting maximum margin separator. Since all the points are initially classified correctly, the leave-one-out error is zero.

(g) (2 pts) Now let us discuss a SVM classifier using a second order polynomial kernel. The first polynomial kernel maps each input data x to $\Phi_1(x) = [x, x^2]^T$. The second polynomial kernel maps each input data x to $\Phi_2(x) = [2x, 2x^2]^T$.

In general, is the margin we would attain using $\Phi_2(x)$

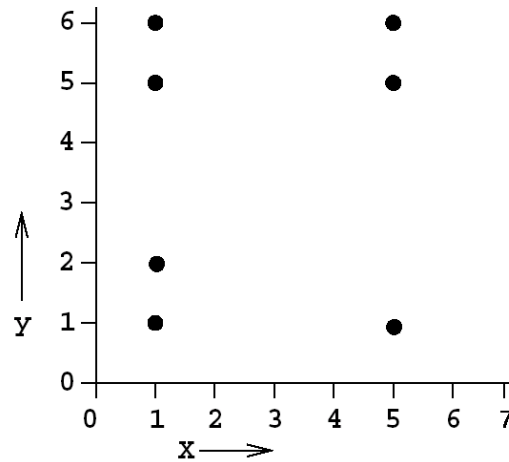
- A. () greater
- B. () equal
- C. () smaller
- D. () any of the above

in comparison to the margin resulting from using $\Phi_1(x)$?

Answer: A.

Problem 4. Instance based learning (8 points)

The following picture shows a dataset with one real-valued input x and one real-valued output y . There are seven training points.



Suppose you are training using kernel regression using some unspecified kernel function. The only thing you know about the kernel function is that it is a monotonically decreasing function of distance that decays to zero at a distance of 3 units (and is strictly greater than zero at a distance of less than 3 units).

(a) (2 pts) What is the predicted value of y when $x = 1$?

Answer: $\frac{1+2+5+6}{4} = 3.5$

(b) (2 pts) What is the predicted value of y when $x = 3$?

Answer: $\frac{1+2+5+6+1+5+6}{7} = 26/7$

(c) (2 pts) What is the predicted value of y when $x = 4$?

Answer: $\frac{1+5+6}{3} = 4$

(d) (2 pts) What is the predicted value of y when $x = 7$?

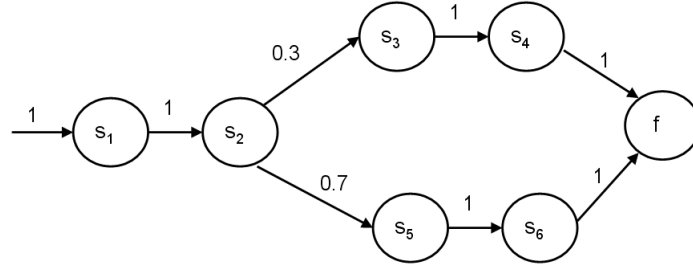
Answer: $\frac{1+5+6}{3} = 4$

Problem 5. HMM (12 points)

Consider the HMM defined by the transition and emission probabilities in the table below. This HMM has six states (plus a start and end states) and an alphabet with four symbols (A,C, G and T). Thus, the probability of transitioning from state S_1 to state S_2 is 1, and the probability of emitting A while in state S_1 is 0.3.

	0	S_1	S_2	S_3	S_4	S_5	S_6	f	A	C	G	T
0	0	1	0	0	0	0	0	0				
S_1	0	0	1	0	0	0	0	0	0.5	0.3	0	0.2
S_2	0	0	0	0.3	0	0.7	0	0	0.1	0.1	0.2	0.6
S_3	0	0	0	0	1	0	0	0	0.2	0	0.1	0.7
S_4	0	0	0	0	0	0	0	1	0.1	0.3	0.4	0.2
S_5	0	0	0	0	0	0	1	0	0.1	0.3	0.3	0.3
S_6	0	0	0	0	0	0	0	1	0.2	0.3	0	0.5

Here is the state diagram:



For each of the pairs belows, place $<$, $>$ or $=$ between the right and left components of each pair. (2 pts each):

(a) $P(O_1 = A, O_2 = C, O_3 = T, O_4 = A, q_1 = S_1, q_2 = S_2)$
 $P(O_1 = A, O_2 = C, O_3 = T, O_4 = A | q_1 = S_1, q_2 = S_2)$

Below we will use a shortened notation. Specifically we will use $P(A, C, T, A, S_1, S_2)$ instead of $P(O_1 = A, O_2 = C, O_3 = T, O_4 = A, q_1 = S_1, q_2 = S_2)$, $P(A, C, T, A)$ instead of $P(O_1 = A, O_2 = C, O_3 = T, O_4 = A)$ and so forth.

Answer: =

$$P(A, C, T, A, S_1, S_2) = P(A, C, T, A | S_1, S_2) P(S_1, S_2) = P(A, C, T, A | S_1, S_2), \text{ since } P(S_1, S_2) = 1$$

(b) $P(O_1 = A, O_2 = C, O_3 = T, O_4 = A, q_3 = S_3, q_4 = S_4)$
 $P(O_1 = A, O_2 = C, O_3 = T, O_4 = A | q_3 = S_3, q_4 = S_4)$

Answer: $<$

As in (b), $P(A, C, T, A, S_3, S_4) = P(A, C, T, A | S_3, S_4) P(S_3, S_4)$ however, since $P(S_3, S_4) = 0.3$, then the right hand side is bigger.

$$(c) P(O_1 = A, O_2 = C, O_3 = T, O_4 = A, q_3 = S_3, q_4 = S_4) \\ P(O_1 = A, O_2 = C, O_3 = T, O_4 = A, q_3 = S_5, q_4 = S_6)$$

Answer: <

The first two emissions (A and C) do not matter since they are the same. Thus, the right hand side translates to $P(O_3 = T, O_4 = A, q_3 = S_3, q_4 = S_4) = P(O_3 = T, O_4 = A | q_3 = S_3, q_4 = S_4) P(S_3, S_4) = 0.7 * 0.1 * 0.3 = 0.021$ while the right hand side is $0.3 * 0.2 * 0.7 = 0.042$.

$$(d) P(O_1 = A, O_2 = C, O_3 = T, O_4 = A) \quad P(O_1 = A, O_2 = C, O_3 = T, O_4 = A, q_3 = S_3, q_4 = S_4)$$

Answer: >

Here the left hand side is: $P(A, C, T, A, S_3, S_4) + P(A, C, T, A, S_5, S_6)$. The right side of the summation is the right hand side above. Since the left side of the summation is greater than 0, the left hand side is greater.

$$(e) P(O_1 = A, O_2 = C, O_3 = T, O_4 = A) \quad P(O_1 = A, O_2 = C, O_3 = T, O_4 = A | q_3 = S_3, q_4 = S_4)$$

Answer: <

As mentioned for (e) the left hand side is: $P(A, C, T, A, S_3, S_4) + P(A, C, T, A, S_5, S_6) = P(A, C, T, A | S_3, S_4) P(S_3, S_4) + P(A, C, T, A | S_5, S_6) P(S_5, S_6)$. Since $P(A, C, T, A | S_3, S_4) > P(A, C, T, A | S_5, S_6)$ the left hand side is lower from the right hand side.

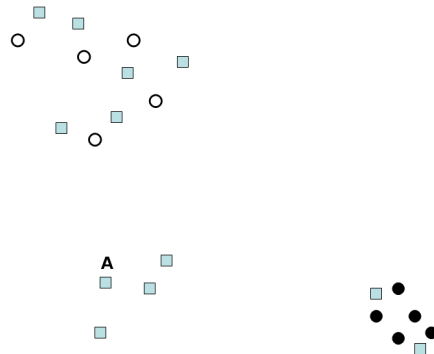
$$(f) P(O_1 = A, O_2 = C, O_3 = T, O_4 = A) \quad P(O_1 = A, O_2 = T, O_3 = T, O_4 = G)$$

Answer: <

Since the first and third letters are the same, we only need to worry about the second and fourth. The left hand side is: $0.1 * (0.3 * 0.1 + 0.7 * 0.2) = 0.017$ while the right hand side is: $0.6 * (0.7 * 0 + 0.3 * 0.4) = 0.072$.

Problem 6. Learning from labeled and unlabeled data (10 points)

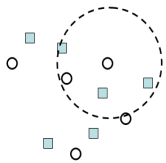
Consider the following figure which contains labeled (class 1 black circles class 2 hollow circles) and unlabeled (squares) data. We would like to use two methods discussed in class (re-weighting and co-training) in order to utilize the unlabeled data when training a Gaussian classifier.



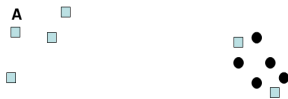
(a) (2 pts) How can we use co-training in this case (what are the two classifiers) ?

Answer:

Co-training partitions the feature space into two separate sets and uses these sets to construct independent classifiers. Here, the most natural way is to use one classifier (a Gaussian) for the x axis and the second (another Gaussian) using the y axis.



(b) We would like to use re-weighting of unlabeled data to improve the classification performance. Re-weighting will be done by placing a dashed circle on each of the labeled data points and counting the number of unlabeled data points in that circle. Next, a Gaussian classifier is run with the new weights computed.



(b.1). (2 pts) To what class (hollow circles or full circles) would we assign the unlabeled point A if we were training a Gaussian classifier using **only** the labeled data points (with no re-weighting)?

Answer:

Hollow class. Note that the hollow points are much more spread out and so the Gaussian learned for them will have a higher variance.

(b.2). (2 pts) To what class (hollow circles or full circles) would we assign the unlabeled point A if we were training a classifier using the re-weighting procedure described above?

Answer:

Again, the hollow class. Re-weighting will not change the result since it will be done independently for each of the two classes, and will produce very similar class centers to the ones in 1 above.

Degree	Disagreement on unlabeled data					Disagreement on training data
	1	2	3	4	5	
1	0	0.3	0.5	0.6	0.7	0.4
2		0	0.2	0.4	0.5	0.2
3			0	0.2	0.5	0.1
4				0	0.3	0
5					0	0

(c) (4 pts) When we handle a polynomial regression problem, we would like to decide what degree of polynomial to use in order to fit a test set. The table below describes the disagreement between the different polynomials on unlabeled data and also the disagreement with the labeled data. Based on the method presented in class, which polynomial should we choose for this data? **Which of the two tables do you prefer?**

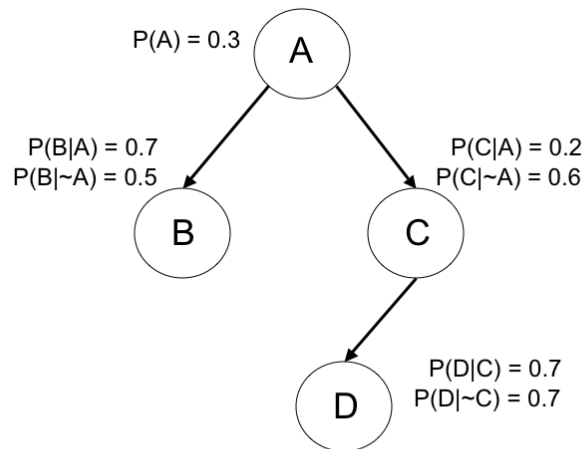
Answer:

The degree we would select is 3. Based on the classification accuracy, it is beneficial to use higher degree polynomials. However, as we said in class these might overfit. One way to test if they do or don't is to check consistency on unlabeled data by requiring that the triangle inequality will hold for the selected degree. For a third degree this is indeed the case since $u(2, 3) = 0.2 \leq l(2) + l(3) = 0.2 + 0.1$ (where $u(2, 3)$ is the disagreement between the second and third degree polynomials on the unlabeled data and $l(2)$ is the disagreement between degree 2 and the labeled data). Similarly, $u(1, 3) = 0.5 \leq l(1) + l(3) = 0.4 + 0.1$. In contrast, this does not hold for a fourth degree polynomial since $u(3, 4) = 0.2 > l(3) + l(4) = 0.1$.

Problem 7. Bayes Net Inference (10 points)

For (a) through (c), compute the following probabilities from the Bayes net below.

Hint: These examples have been designed so that none of the calculations should take you longer than a few minutes. If you find yourself doing dozens of calculations on a question sit back and look for shortcuts. This can be done easily if you notice a certain special property of the numbers on this diagram.



(a) (2 pts) $P(A|B) =$

Answer: $3/8$.

$$P(A|B) = P(A \wedge B) / P(B) = P(B|A) \cdot P(A) / (P(B|A) \cdot P(A) + P(B|\neg A) \cdot P(\neg A)) = 0.21 / (0.21 + 0.35) = 3/8.$$

(b) (2 pts) $P(B|D) =$

Answer: 0.56 .

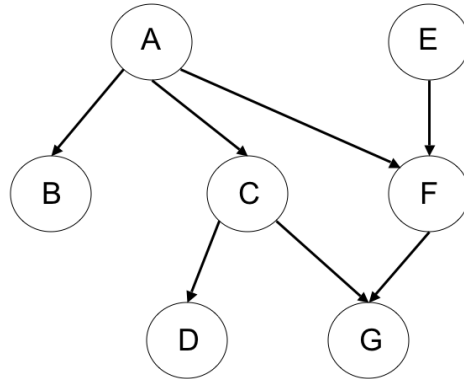
$P(D|C) = P(D|\neg C)$ so D is independent of C , and is not influencing the Bayes net. So $P(B|D) = P(B)$, which we calculated in (a) to be 0.56 .

(c) (2 pts) $P(C|B) =$

Answer: $5/11$.

$$P(C|B) = (P(A \wedge B \wedge C) + P(\neg A \wedge B \wedge C)) / P(B) = (P(A) \cdot P(B|A) \cdot P(C|A) + P(\neg A) \cdot P(B|\neg A) \cdot P(C|\neg A)) / P(B) = (0.042 + 0.21) / 0.56 = 5/11.$$

For (d) through (g), indicate whether the given statement is **TRUE** or **FALSE** in the Bayes net given below.



(d) [**T/F** - (1 pt)] $I \langle A, \{\}, E \rangle$

Answer: TRUE.

(e) [**T/F** - (1 pt)] $I \langle A, G, E \rangle$

Answer: FALSE.

(f) [**T/F** - (1 pt)] $I \langle C, \{A, G\}, F \rangle$

Answer: FALSE.

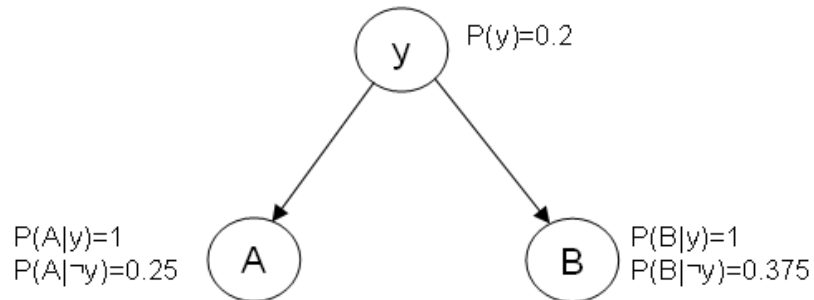
(g) [**T/F** - (1 pt)] $I \langle B, \{C, E\}, F \rangle$

Answer: FALSE.

Problem 8. Bayes Nets II (12 points)

- (a) (4 points) Suppose we use a naive Bayes classifier to learn a classifier for $y = A \wedge B$, where A, B are independent of each other boolean random variables with $P(A) = 0.4$, $P(B) = 0.5$. Draw the Bayes net that represents the independence assumptions of our classifier and fill in the probability tables for the net.

Answer:



In computing the probabilities for the Bayes net we use the following Boolean table with corresponding probabilities for each row:

A	B	y	P
0	0	0	$0.6 \cdot 0.5 = 0.3$
0	1	0	$0.6 \cdot 0.5 = 0.3$
1	0	0	$0.4 \cdot 0.5 = 0.2$
1	1	1	$0.4 \cdot 0.5 = 0.2$

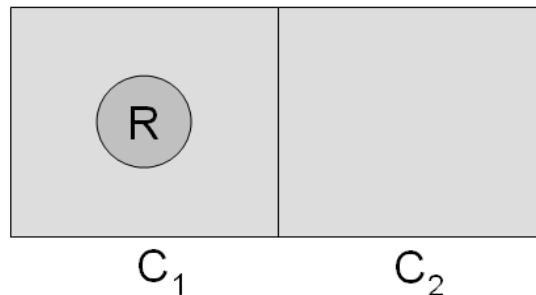
Using the table we can compute the probabilities for the Bayes net: $P(y) = 0.2$

$$P(B|y) = \frac{P(B,y)}{P(y)} = 1$$

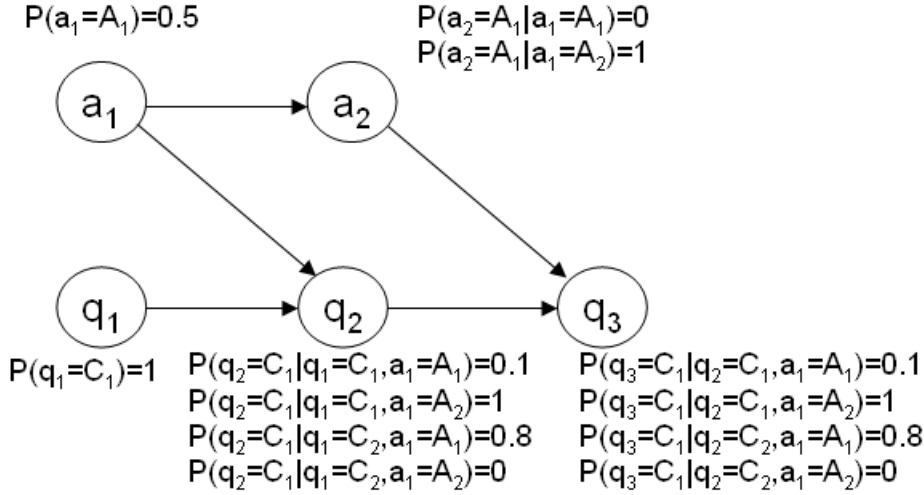
$$P(B|\neg y) = \frac{P(B,\neg y)}{P(\neg y)} = \frac{0.3}{0.8} = 0.375$$

$$P(A|y) = \frac{P(A,y)}{P(y)} = 1$$

$$P(A|\neg y) = \frac{P(A,\neg y)}{P(\neg y)} = \frac{0.2}{0.8} = 0.25$$



- (b) (8 points) Consider a robot operating in the two-cell gridworld shown below. Suppose the robot is initially in the cell C_1 . At any point of time the robot can execute any of the two actions: A_1 and A_2 . A_1 is "to move to a neighboring cell". If the robot is in C_1 the action A_1 succeeds (moves the robot into C_2) with the probability 0.9 and fails (leaves the robot in C_1) with the probability 0.1. If the robot is in C_2 the action A_1 succeeds (moves the robot into C_1) with the probability 0.8 and fails (leaves the robot in C_2) with the probability 0.2. The action A_2 is "to stay in the same cell", and when executed it keeps the robot in the same cell with probability 1. The first action the robot executes is chosen at random (with an equal probability between A_1 and A_2). Afterwards, the robot alternates the actions it executes. (for example, if the robot executed action A_1 first, then the sequence of actions is $A_1, A_2, A_1, A_2, \dots$). Answer the following questions.



- (b.1) (4 points) Draw the Bayes net that represents the cell the robot is in during the first two actions the robot executes (e.g, initial cell, the cell after the first action and the cell after the second action) and fill in the probability tables. (Hint: The Bayes net should have five variables: q_1 - the initial cell, q_2, q_3 - the cell after the first and the second action, respectively, a_1, a_2 - the first and the second action, respectively).

Answer:

- (b.2) (4 points) Suppose you were told that the first action the robot executes is A_1 . What is the probability that the robot will appear in cell C_1 after it executes close to infinitely many actions?

Answer: Since actions alternate and the first action is A_1 the transition matrix for any odd action is:

$$P(a_{odd}) = \begin{pmatrix} 0.1 & 0.9 \\ 0.8 & 0.2 \end{pmatrix},$$

where p_{ij} element is a probability of transitioning into cell j as a result of an execution of an odd action given that the robot is in cell i before executing this action.

Similarly, the transition matrix for any even action is: $P(a_{even}) = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$.

If we consider the pair of actions as one "meta-action", then we have a Markov chain with the transition probability matrix:

$$P = P(a_{odd}) * P(a_{even}) = \begin{pmatrix} 0.1 & 0.9 \\ 0.8 & 0.2 \end{pmatrix}.$$

At $t = \infty$, the state distribution satisfies $P(q_t) = P^T * P(q_t)$. So,

$$P(q_t = C_1) = 0.1 * P(q_{t-1} = C_1) + 0.8 * P(q_{t-1} = C_2).$$

Since there are only two cells possible we have:

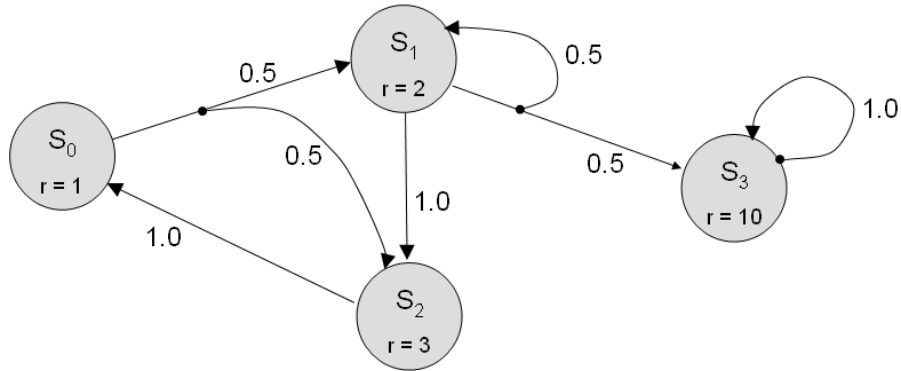
$$P(q_t = C_1) = 0.1 * P(q_{t-1} = C_1) + 0.8 * (1 - P(q_{t-1} = C_1)).$$

Solving for $P(q_t = C_1)$ we get:

$$P(q_t = C_1) = 0.8/1.7 = 0.4706.$$

Problem 9. Markov Decision Processes (11pts)

- (a) (8 points) Consider the MDP given in the figure below. Assume the discount factor $\gamma = 0.9$. The r -values are rewards, while the numbers next to arrows are probabilities of outcomes. Note that only state S_1 has two actions. The other states have only one action for each state.



- (a.1) (4 points) Write down the numerical value of $J(S_1)$ after the first and the second iterations of Value Iteration.

Initial value function: $J^0(S_0) = 0$; $J^0(S_1) = 0$; $J^0(S_2) = 0$; $J^0(S_3) = 0$;

$$J^1(S_1) = \text{Answer:}$$

$$J^1(S_1) = 2$$

$$J^2(S_1) =$$

$$\begin{aligned} J^2(S_1) &= \max(2 + 0.9(0.5 * J^1(S_1) + 0.5 * J^1(S_3)), 2 + 0.9 * J^1(S_2)) \\ &= \max(2 + 0.9(0.5 * 2 + 0.5 * 10), 2 + 0.9 * 3) \\ &= 7.4 \end{aligned}$$

- (a.2) (4 points) Write down the optimal value of state S_1 . There are few ways to solve it, and for one of them you may find useful the following equality: $\sum_{i=0}^{\infty} \alpha^i = \frac{1}{1-\alpha}$

$$J^*(S_1) =$$

$$\frac{1}{1-\alpha} \text{ for any } 0 \leq \alpha < 1.$$

Answer:

It is pretty clear from the given MDP that the optimal policy from S_1 will involve trying to move from S_1 to S_3 as this is the only state that has a large reward. First, we compute optimal value for S_3 :

$$J^*(S_3) = 10 + 0.9 * J^*(S_3)$$

$$J^*(S_3) = \frac{10}{0.1} = 100$$

We can now compute optimal value for S_1 :

$$J^*(S_1) = 2 + 0.9(0.5 * J^*(S_1) + 0.5 * J^*(S_3)) = 2 + 0.9(0.5 * J^*(S_1) + 50);$$

Solving for $J^*(S_1)$ we get:

$$J^*(S_1) = \frac{47}{0.55} = 87.\overline{45}$$

- (b) (3 points) A general MDP with N states is guaranteed to converge in the limit for Value Iteration as long as $\gamma < 1$. In practice one cannot perform infinitely many value iterations to guarantee convergence. Circle **all** the statements below that are **true**.

- (1) Any MDP with N states converges after N value iterations for $\gamma = 0.5$;

Answer: False

- (2) Any MDP converges after the 1st value iteration for $\gamma = 1$;

Answer: False

- (3) Any MDP converges after the 1st value iteration for a discount factor $\gamma = 0$;

Answer: True, since all the converged values will be just immediate rewards.

- (4) An acyclic MDP with N states converges after N value iterations for any $0 \leq \gamma \leq 1$.

Answer: True, since there are no cycles and therefore after each iteration at least one state whose value was not optimal before is guaranteed to have its value set to an optimal value (even when $\gamma = 1$), unless all state values are already converged.

- (5) An MDP with N states and no stochastic actions (that is, each action has only one outcome) converges after N value iterations for any $0 \leq \gamma < 1$.

Answer: False. Consider a situation where there are no absorbing goal states.

- (6) One usually stops value iterations after iteration $k+1$ if: $\max_{0 \leq i \leq N-1} |J^{k+1}(S_i) - J^k(S_i)| < \xi$, for some small constant $\xi > 0$.

Answer: True.