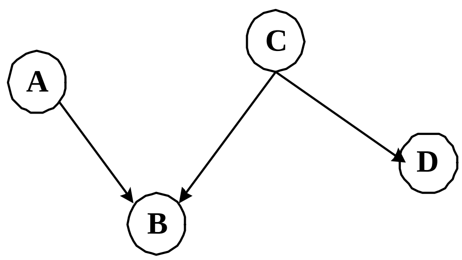
1. Consider the following events:
A = You receive a million dollars;
B = You receive a utility of 0.2;
C = You receive a utility of 0.7.

If you are indifferent between A and a lottery between B and C where your chances of winning B are 0.15 and your chances of winning C are 0.85, what is the utility of a million dollars to you?

Solution: The utility of the lottery L between B and C is, $L = 0.15 * 0.2 + 0.85 * 0.7 = 0.625$ Since A and L are indifferent to you, the utility of a million dollars should also be 0.625.

2. Consider the following Bayesian network structure, where A, B, Cand D are boolean variables:



a) Is A independent of D? Y.
b) Is A independent of D given B? N.
c) Is A independent of D given C? Y.
d) Suppose you are given the following set of training examples:

A	B	C	D
0	1	0	1
0	?	1	1
1	0	0	0
1	0	1	?
0	0	?	1

Show the sequence of filled-in values and parameters produced by the EM algorithm, assuming the parameters are initialized by ignoring missing values.
Solution.
Initialization: $P(A) = \frac{2}{5}$, $P(C) = \frac{1}{2}$

$P(B|AC)$:

	A	$\neg A$
C	0	$0^\star$
$\neg C$	0	1

$P(D|C)$:

C	$\neg C$
1	$\frac{1}{2}$

$\star$: note here we can not get any information from the data what happens when $A=0$ and $C=1$. So we just initialize it a random value, say, $P(B|\neg A, C)=0$ here.
E-step1: from CPT, we can get directly $P(B|\neg A, C)=0$ and $P(D|C)=1$, so $B=0$ when $A=0$ and $C=1$, which means $(0,?,1,1)\rightarrow(0,0,1,1)$; and $D=1$ when $C=1$, i.e. $(1,0,1,?)\rightarrow(1,0,1,1)$

$$\begin{aligned} P(C|\neg A,\neg B,D) &= \frac{P(\neg A,\neg B,C,D)}{P(\neg A,\neg B,C,D)+P(\neg A,\neg B,\neg C,D)} \\ &= \frac{P(\neg A)P(\neg B|\neg A,C)P(C)P(D|C)}{P(\neg A)P(\neg B|\neg A,C)P(C)P(D|C)+P(\neg A)P(\neg B|\neg A,\neg C)P(\neg C)P(D|\neg C)} \\ &= 1 \end{aligned}$$

which means that when $A=0$, $B=0$, and $D=1$, $C=1$ with probability 1. So $(0,0,?,1)\rightarrow(0,0,1,1)$.
M-step1: re-calculate the CPT for each node. $P(A)=\frac{2}{5}$, $P(C)=\frac{3}{5}$

$P(B|AC)$:

	A	$\neg A$
C	0	0
$\neg C$	0	1

$P(D|C)$:

C	$\neg C$
1	$\frac{1}{2}$

E-step2: according to the updated CPTs, we can prove that the unknown values are just the same as in E-step1. So the process converges.

3. Representing the following boolean functions using:
(1) decision trees;
(2) neural networks: show the structure of the network and the weights on the edges.

(a) $A \wedge \neg B$

(b) $A \vee [B \wedge C]$

(c) $A\ XOR\ B$

Solution is shown in figure 1.

4. Suppose we want to classify a given ball into one of these three classes: {H,M, L}, based on three attributes: the color of the ball({Y, R, P}), the size of the ball({L,S}), and the price of the ball({C1, C2, C3}). Build a decision tree to learn the classification, choosing the best attribute at each step according to information gain.

Price	Color	Size	Class
C1	Y	L	M
C2	Y	S	H
C2	R	L	L
C3	R	S	M
C3	P	L	H
C1	P	S	H

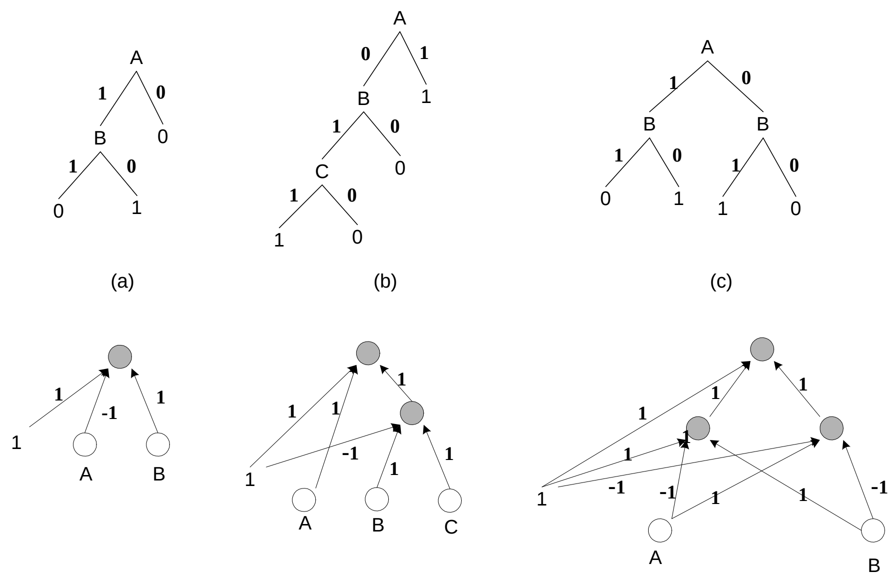


Figure 1: Solution for prob. 3

Solution:
1) $Entropy = -\sum_i p_i log P_i = \frac{2}{3} + \frac{log3}{2}$.
2) Choose the "best" feature for step 1:

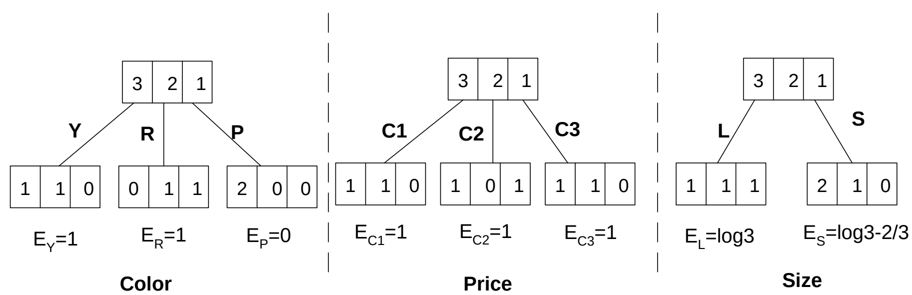


Figure 2: Prob4-step1: choose the best feature

$InfoGain_{color} = Entropy - (E_Y P(Y) + E_R P(R) + E_P P(P)) = \frac{log3}{2}$
$InfoGain_{price} = Entropy - (E_{C_1} P(C_1) + E_{C_2} P(C_2) + E_{C_3} P(C_3)) = \frac{log3}{2} - \frac{1}{3}$
$InforGain_{size} = Entropy - (E_L P(L) + E_S P(S)) = 1 - \frac{log3}{2}$
So the best feature is color.
3) Choose the best feature for step 2. The process is similar as in step1, and the result is, price is as good as size, so just randomly pick one.

5. Consider the learning approaches we've learned in class, which might be the best in the following cases:

1. there are 13 examples in the training set, each is a vector of six continuous value, the attributes are tight-connected;

2. 1000-dimension instance space, the attribute values are independent given the classifications, and are normal distributed;

3. training set of size 10000, the attributes are loosely connected.

Solution:

1. instance-based algorithm should be the best in this situation. Note for tree algorithm, it always needs more samples than a dozen. And since the attributes are tight-connected, we can not simply assume they are independent, as we do in Bayesian algorithm.

2. naive Bayesian.

3. Bayesian network.

6. What is the "curse of dimensionality"? Explain two approaches to select "best" features. What is the asymptotic time complexity of them for nearest-neighbor as a function of the number of training anvalidation examples and the number of attributes?

Solution: suppose N_t denotes the size of the training set, N_v denotes the size of the validation set, and N_f the number of features. Then

$$\bigcirc(t) = \bigcirc(N_t N_v N_f^2)$$