CS 4824 / ECE 4424 Homework 1 Solution (Written Portion) [Total Points: 48]

Question 1 [18 points]

a. [0.5 x 3 = 1.5 points]
Gini of overall data = 0.5
Entropy of overall data = 1
Misclassification error of overall data = 0.5

b. [0.5 x 12 = 6 points]

	Customer ID	Housing Type	Gender	Marital Status
Gini	0	0.2	0.5	0.5
Entropy	0	0.451	1	1
Misclassification Error	0	0.125	0.5	0.5

c. [0.5 x 6 = 3 points]
IG (Customer ID) = 1
IG (Housing Type) = 0.549
IG (Gender) = 0
IG (Marital Status) = 0
IG (Customer ID) = 1 is the highest
IG (Gender) = IG (Marital Status) = 0 is the lowest

d. [0.5 x 5 = 2.5 points]
Gain Ratio (Customer ID) = 1/4 = 0.25
Gain Ratio (Housing Type) = 0.549/1.579 = 0.348
Gain Ratio (Gender) = 0
Gain Ratio (Marital Status) = 0
Gain Ratio (Housing Type) is highest

e. [1 + 1 = 2 points]
We would choose Housing Type for splitting at the root node, since it provides the maximum Gain Ratio. Gain Ratio penalizes excessive number of partitions of node into multiple children, thus accounting for the entropy of constructing multiple children nodes. Hence, even though Customer ID provides the maximum Information Gain = 1, it constructs 16 children (one for every unique value of Customer ID), each of which contains a single data instance. Having a large number of small-size partitions can lead to overfitting since we would be learning the majority class at every leaf using only a few training instances, which can be highly biased by noise.

f. [0.5 x 3 + 1 + 0.5 = 3 points]

[0.5 x 3 points]
Difference in Entropy for Tree1 = 1
Difference in Entropy for Tree2 = 1
Difference in Entropy for Tree3 = 0.655

[1 point] Looking at the difference between the entropy of the overall data and the weighted entropy at the leaves, we can observe that Tree1 and Tree2 provide the maximum difference (drop) in entropy. However, we have already seen in (e) that splitting on the Customer ID provides lower

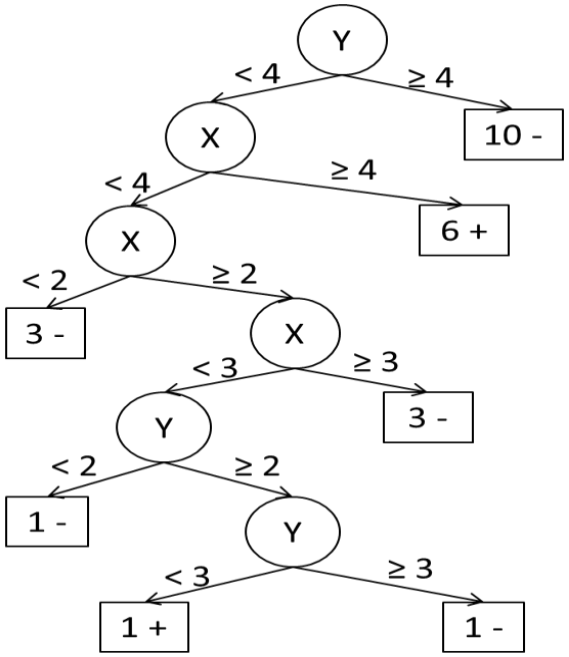
Gain Ratio than splitting on Housing Type, suggesting it to be a too complex model prone to overfitting. Hence Tree2 is preferred over Tree1 for performing classification. The attribute chosen at the root node is thus Marital Status which is different that the attribute chosen in (e), which is Housing Type.

[0.5 points] This shows that even though using Marital Status at the root node does not show any improvement in performance initially, but by further splitting the children according to Gender, we are able to obtain an optimal tree which does a perfect job at classification (0 misclassifications). This shows that the greedy approach of choosing the best attribute at each node may not always produce the optimal solution, especially in cases where there is interaction amongst variables.

Question 2 [14 points]

a. [6 points]

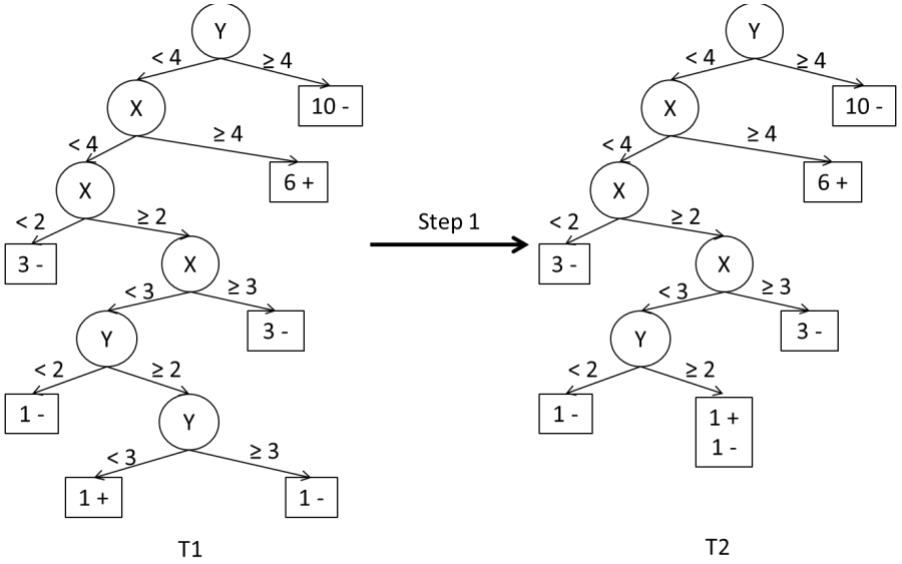
An example of a decision tree with 6 number of splits is given below.



There can be other variants of this tree with 6 internal nodes which can correctly classify each training instance. Any such tree would be accepted as a valid answer. If they use multiple attributes to split at an internal node or use equality conditions at an internal node, the number of internal nodes will be less than 6, and they will lose 1 point.

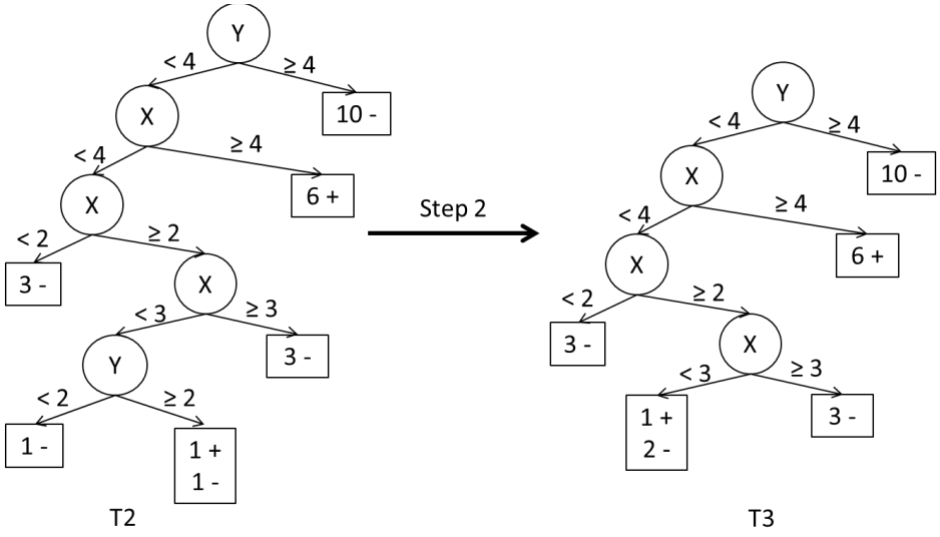
b. [5 points]

Using sub-tree replacement method, we choose a pair of leaf nodes in tree T and replace the subtree rooted at its parent with a leaf node, creating a pruned tree, Tpruned. If the pessimistic estimate of error reduces for Tpruned when compared with the original tree T, we replace T with Tpruned and recursively keep pruning T till the pessimistic estimate on Tpruned starts exceeding the pessimistic estimate on T. We then output T as our pruned tree. The steps of this procedure on the tree shown in (c) can be shown as follows (1 point for each step):



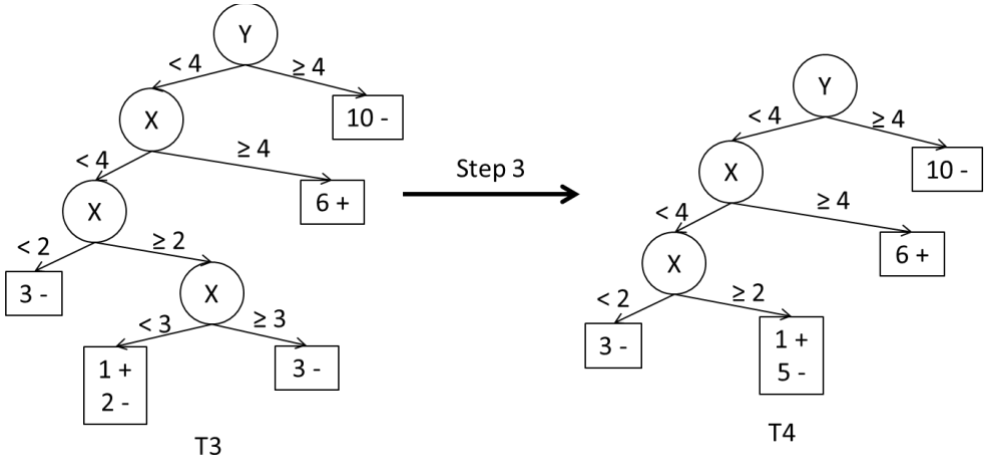
T1 T2

Pessimistic Estimate (T1) = (0 + 7x2)/25 = 14/25
Pessimistic Estimate (T2) = (1 + 6x2)/25 = 13/25
Pessimistic Estimate(T1) < Pessimistic Estimate (T1), hence proceed further\

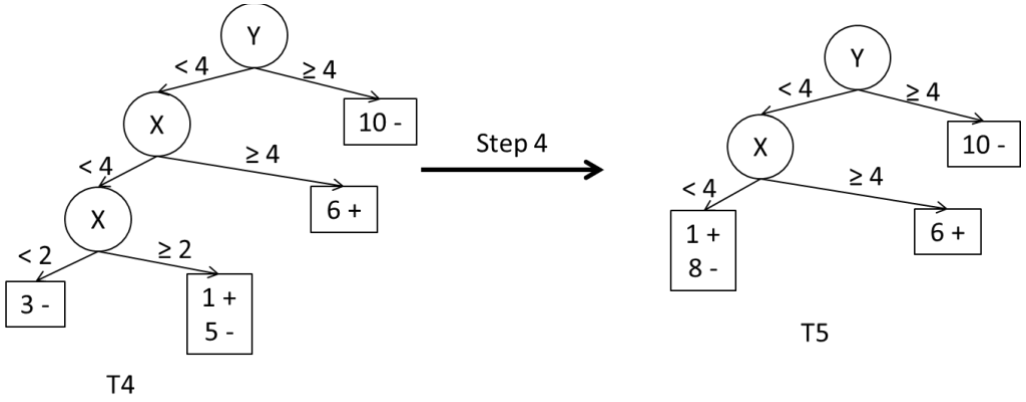


T2 T3

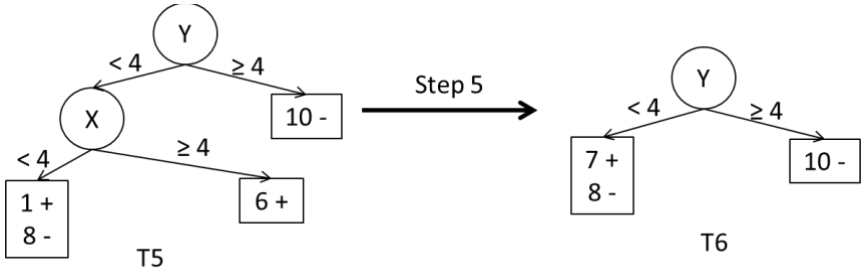
Pessimistic Estimate (T3) = (1 + 5x2)/25 = 11/25
Pessimistic Estimate(T3) < Pessimistic Estimate (T2), hence proceed further



Pessimistic Estimate (T4) = (1 + 4x2)/25 = 9/25
Pessimistic Estimate(T4) < Pessimistic Estimate (T3), hence proceed further



Pessimistic Estimate (T5) = (1 + 3x2)/25 = 7/25
Pessimistic Estimate(T5) < Pessimistic Estimate (T4), hence proceed further



Pessimistic Estimate (T6) = (7 + 2x2)/25 = 11/25
Pessimistic Estimate(T6) > Pessimistic Estimate (T4), hence stop. T5 is our pruned tree.

c. [2 + 1 = 3 points]

[2 points] The original tree seems to be too complex given the pattern in the training data because of the presence of a noisy (+)-instance at (X = 2, Y = 2). By performing the subtree replacement pruning procedure, we obtain T5 as our pruned tree which appears to be simpler than the original

tree T1. This is captured in the fact that the optimistic error estimate of T1 is better than T5, but the pessimistic error estimate of T5 is better than T1. Hence, we would choose the pruned tree T5 for performing future classification.

[1 point] The phenomena being explored in this question was "overfitting."

Question 3 [16 points]

a. **[0.5 x 6 = 3 points]**

P(A = 1|+) = 0.6, P(B = 1|+) = 0.4, P(C = 1|+) = 0.8, P(A = 1|−) = 0.4, P(B = 1|−) = 0.4 and P(C = 1|−) = 0.2

b. **[1 + 1 = 2 points]**

Let R : (A = 1,B = 1,C = 1) be the test record. To determine its class, we need to compute P(+|R) and P(−|R). Using Bayes theorem, P(+|R) = P(R|+)P(+)/P(R) and P(−|R) = P(R|−)P(−)/P(R). Since P(+) = P(−) = 0.5 and P(R) is constant, R can be classified by comparing P(R|+) and P(R|−).

For this question,

[1 point] P(R|+) = P(A = 1|+) * P(B = 1|+) * P(C = 1|+) = 0.192
[1 point] P(R|−) = P(A = 1|−) * P(B = 1|−) * P(C = 1|−) = 0.032
Since P(R|+) is larger, the record is assigned to (+) class.

c. [1.5 + 0.5 = 2 points]

[0.5 x 3 = 1.5 points] P(A = 1|+) = 0.6, P(B = 1|+) = 0.4 and P(A = 1,B = 1|+) = 0.2.
[0.5 points] Therefore, A and B are not conditionally independent.

d. [0.5 x 4 + 1 = 3 points]

Given the set of attributes for a data instance, R = {r1, r2, … rk}, we can compute the posterior probabilities of it belonging to the + class, or the – class, using the Bayes formula:

$P(+|R) = P(R|+).P(+)/P(R)$, and $P(-|R) = P(R|-).P(-)/P(R)$

P(+) and P(-) are the prior probabilities that can be computed using the data. To compute the conditional probabilities, we can make the Naïve Bayes assumption to say:

$P(R|+) = P(r1|+).P(r2|+)…P(rk|+)$, and $P(R|-) = P(r1|-).P(r2|-)…P(rk|-)$

It is generally difficult to directly compute P(R) from the data, and we treat it as a constant if we are required to compare the posterior probabilities of + and – in order to predict the class of a given data instance (since P(R) is the constant denominator term in the posterior probabilities of both + and -). However, if we are required to compute the posterior probabilities, we would need to compute the value of P(R), and a common approach for that is to consider the fact the sum of the posterior probabilities of + and – should be equal to 1. This implies:

$P(R|+).P(+)/P(R) + P(R|-).P(-)/P(R) = 1$, or
$P(R) = P(R|+).P(+) + P(R|-).P(-)$

[0.5 x 4 = 2 points] Using the above approach, we can compute the posterior probabilities for each of the subparts as follows: (i) P(+) = 0.5, (ii) P(+|A=1) = 0.6, (iii) P(+|A=1,B=1) = 0.6, and (iv) P(+|A=1,B=1,C=1) = 0.857.
[1 point] As more and more information/attributes is available, the classifier is getting more certain about labelling the given instance with its true class: +, as indicated in row 10 of Table 3.

e. **[0.5 x 6 = 3 points]**

P(A = 1|+) = 0, P(B = 1|+) = 0.4, P(C = 1|+) = 0.8, P(A = 1|−) = 0.4, P(B = 1|−) = 0 and P(C = 1|−) = 0.2

f. **[0.5 x 2 = 1 point]**

P(+|x) = 0/0 and P(-|x) = 0/0. Thus, we cannot assign a class to this instance

e. **[1 + 1 = 2 points]**

[1 point] If a conditional probability is 0, then the complete posterior probability becomes zero. If this happens for both classes then it is not possible to assign a class to the instance.
[1 point] To fix this problem, we can use alternate ways of calculating posteriors such as the m-estimate and the Laplace estimate.

Practice Questions

Question 4

a. 15 (for choosing X1) x 14 (for choosing X2) x 14 (for choosing X3) = 2940
b. $0.5^{10} = 9.7 \times 10^{-4}$
c. $1 - (1 - 0.5^{10})^{2940} = 0.943$

d. Even though the probability of a single classifier correctly classifying a single data instance is 0.5 (random guess), the probability of finding at least one classifier which accurately classifies all 10 data instances is quite high (> 0.9). This is happening because the total number of possible classifiers that can be constructed (= 2940) is quite high since we have 15 attributes, and it is highly likely that by random chance we would find a classifier that correctly classifies all 10 data instances even though each classifier is randomly guessing. The phenomena being observed over here is 'multiple comparison procedure', which is creating a false impression of the quality of the classifiers.

Question 5

b only. T2 will fit the training set no better than T1, since T1 is the tree with the minimum classifier error on the training set, and some information is lost in the pruning of T1 to create T2. However, there is no guarantee on the test error of T1 being smaller or greater than T2 (either case can happen).

Question 6

i.
a. Same. The last attribute will never be used for splitting.
b. Same. Since predictions are same, there is no change in accuracy.

ii.
a. Different. Additional probability term corresponding to X4.
b. Worse/same/better. All three cases are possible.