Exercise 7 - Week 8
Solution Classification

CLASSIFICATION

Assume a sample of data which categorizes a DRINK as YES or NO, based on whether people with certain AGE and GENDER would drink it or not. The attributes can take the following values.

```
DRINK  possible values: Chocolate, Egg Milk, Strawberry Juice, Tea

AGE    possible values: <=25, >25

GENDER possible values: Male, Female
```

(Since each feature value starts with a different letter, for shorthand we'll just use that initial letter, e.g., 'C' for Chocolate)

The category labels are marked as YES or NO.

Here is our **TRAINING** set:

```
DRINK = C AGE <= 25 GENDER = M CATEGORY = NO
DRINK = E AGE > 25 GENDER = M CATEGORY = NO
DRINK = C AGE <= 25 GENDER = F CATEGORY = NO
DRINK = C AGE > 25 GENDER = F CATEGORY = NO
DRINK = S AGE > 25 GENDER = F CATEGORY = YES
DRINK = E AGE <= 25 GENDER = M CATEGORY = YES
DRINK = E AGE <= 25 GENDER = F CATEGORY = YES
```

Our **TUNING** set:

```
DRINK = C AGE > 25 GENDER = M CATEGORY = NO
DRINK = S AGE > 25 GENDER = F CATEGORY = YES
DRINK = E AGE > 25 GENDER = F CATEGORY = YES
DRINK = E AGE <= 25 GENDER = F CATEGORY = NO
DRINK = C AGE <= 25 GENDER = M CATEGORY = NO
```

Our **TESTING** set:

```
DRINK = C AGE > 25 GENDER = F CATEGORY = NO
DRINK = C AGE <= 25 GENDER = M CATEGORY = NO
DRINK = T AGE <= 25 GENDER = F CATEGORY = YES
DRINK = E AGE <= 25 GENDER = M CATEGORY = YES
DRINK = E AGE > 25 GENDER = F CATEGORY = YES
```

Question 1. Inducing the initial decision tree

First, compute Information gains for each attribute and select the best partition.

From now on, our training set is referred as **D**.

No. of categories = 2 (YES and NO)

$$I(D) = -\sum_{i=1}^{2} p_i log_2(p_i)$$

where p_i is the probability that a sample belongs to the i^{th} category

- YES, $p_i = 3/7$
- NO, $p_i = 4/7$

$$I(D) = -\left[\frac{3}{7}log_2\frac{3}{7} + \frac{4}{7}log_2\frac{4}{7}\right] = 0.985$$

Compute Entropy for each attribute A

For attribute DRINK

- DRINK splits D into 4 partitions D_1, D_2, D_3 and D_4
- D_1 contains all samples where DRINK = C, D_2 contains all samples where DRINK = E, D_3 contains all samples where DRINK = S, and D_4 contains all samples where DRINK = T.

DRINK	YES	NO
C	0	3
E	2	1
S	1	0
T	0	0

$$H_{DRINK} = \sum_{i-1}^{4} \frac{|D_i|}{|D|} I(D_i)$$

$$H_{DRINK} = \frac{3}{7}\left[-0 - \frac{3}{3}log_2 1\right] + \frac{3}{7}\left[-\frac{2}{3}log_2\frac{2}{3} - \frac{1}{3}log_2\frac{1}{3}\right] + \frac{1}{7}\left[-0 - log_2 1\right] = 0.39$$

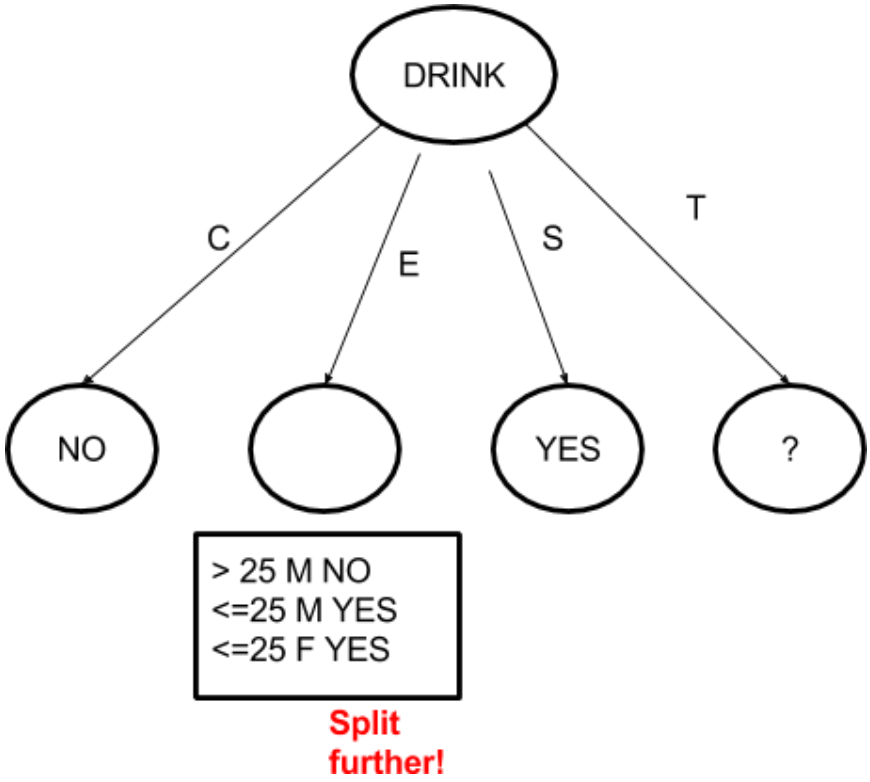
For attributes AGE, GENDER

- $H_{AGE} = 0.964$
- $H_{GENDER} = 0.964$

Compute the gains

- Gain(DRINK) = I(D) - H_{DRINK} = 0.985 - 0.39 = 0.595
- Gain(AGE) = 0.02
- Gain(GENDER) = 0.02

DRINK is the attribute with the highest gain, so it is chosen as the first split attribute.



Split for the new sample set

```
DRINK = E AGE > 25 GENDER = M CATEGORY = NO
DRINK = E AGE <= 25 GENDER = M CATEGORY = YES
DRINK = E AGE <= 25 GENDER = F CATEGORY = YES
```

$$I(D) = -\left[\frac{2}{3}log_2\frac{2}{3} + \frac{1}{3}log_2\frac{1}{3}\right] = 0.918$$

Compute gains for attributes

AGE splits D in two partitions, D_1 and D_2

AGE	YES	NO
>25	0	1
<=25	2	0

$$H_{AGE} = \frac{1}{3}\left[-0-1log_2 1\right] + \frac{2}{3}\left[-1log_2 1 - 0\right] = 0$$

GENDER splits D in two partitions D_1 and D_2

GENDER	YES	NO
M	1	1
F	1	0

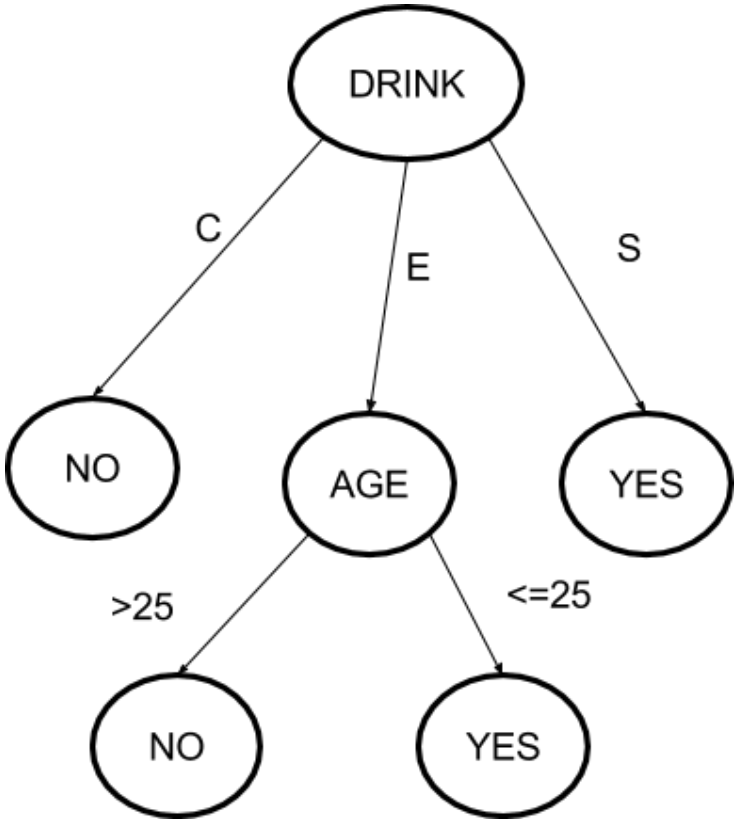
$$H_{GENDER} = \frac{2}{3}\left[-\frac{1}{2}log_2\frac{1}{2} - \frac{1}{2}log_2\frac{1}{2}\right] + \frac{1}{3}\left[-1log_2 1 - 0\right] = \frac{2}{3}$$

Gains:

- Gain(AGE) = 0.918
- Gain(GENDER) = 0.31

AGE has the highest gain

DECISION TREE



Question 2. Pruning the tree to reduce overfitting

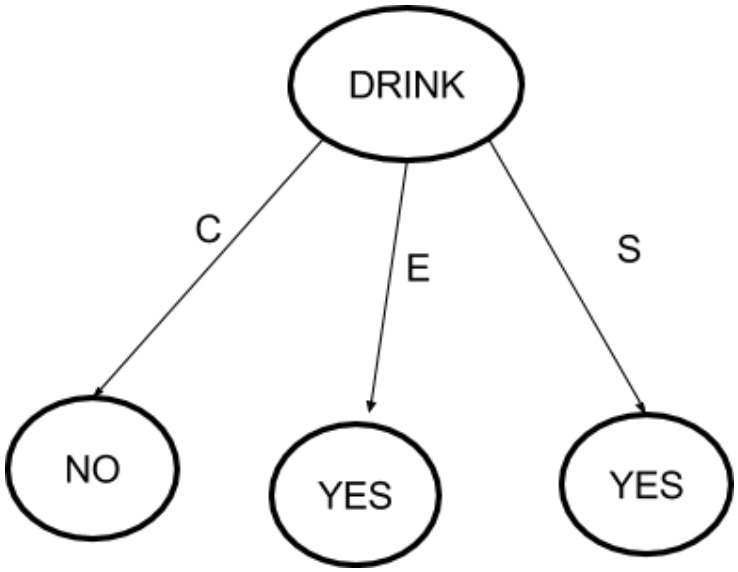
Overfitting occurs when a decision tree conforms too closely to the training data and does not accurately model the underlying concept. One way to address this problem in decision-tree induction is to use a tuning set in conjunction with a pruning algorithm. Here we will use a 'greedy' algorithm sketched in Figure 1 below.

Apply the pruning algorithm to the decision tree produced in (a).

First iteration:

Accuracy over Tunning set = 60%, bestTree is set to this tree.

Second iteration:



Accuracy over Tunning set = 80%, bestTree is set to this tree.

No improvements in future iterations

Question 3. Estimating future accuracy

Apply the decision tree produced in (b) to the TESTING data samples and report its accuracy. What is the accuracy of the unpruned tree (produced in (a)) for these data samples. Briefly discuss your results.

Accuracy over **TESTING** set:

- **Decision tree** = 60%
- **Pruned tree** = 80% (Everything matches, except “T” due to missing samples)

Discussion:
The decision tree algorithm tries to overfit and ends up with lesser accuracy. It's partially due to the limited number of samples in the TRAINING set.
A training set with missing samples for some attributes can lead to less accuracy. The training set must be chosen carefully.

Pruning Algorithm: BEGIN

Let *bestTree* be the tree produced by decision tree induction on the TRAINING set.
Let *bestAccuracy* be the accuracy of *bestTree* on the TUNING set. Let *progressMade* = true

```
While (progressMade) // Continue as long as improvement on TUNING SET

{

  progressMade = false; currentTree = bestTree

  For each non-leaf node N  in currentTree   {

      /* consider various pruned versions of the current tree and see if any, is
     better than the best tree found so far */

    (STEP-1) prunedTree = currentTree

     (STEP-2)Replace node N in prunedTree by a leaf node and label it with the
     category label of majority class among TRAINING set examples that reached
     node N (break ties in favor of 'NO')

    (STEP-3) newAccuracy = accuracy of prunedTree  on the TUNING set

      /* is this pruned tree an improvement, based on the TUNE set? In case of a
     tie, go with the smaller tree */

    (STEP-4)If (newAccuracy >= bestAccuracy) {

         bestAccuracy = newAccuracy; bestTree = prunedTree

         progressMade = true

       }

   }

 } return bestTree;
```

Pruning Algorithm: END

Figure 1 Pruning Algorithm for question (b)