ML HW1

Avishek Choudhury - u1428226

September 2023

Decision Tree

1. (a) Algorithm:

   i. First check if all labels are same or not. If same then the Entropy is 0. Hence, we can conclude that the root node is the leaf node and return the label value.

   ii. If there are multiple labels we have to find the Entropy of the entire dataset:

$$Entropy(S) = H(S) = -\sum_{i=1}^{k} p_i \log_2^{p_i}$$

$$= -(5/7 * \log_2^{5/7} + 2/7 * \log_2 2/7) = 0.86312$$

   iii. After getting the Entropy we need to calculate the Information Gain(IG) of all the features and consider the feature with highest IG as the root node. IG for X_1:

$$IG(S, A) = H(S) - \sum_{v \in Values(A)} \frac{|S_v|}{S} Entropy(S_v)$$

$$IGforX_1 = 0.863 - (5/7 * 0.721 + 2/7 * 1) = 0.06312$$

$$IGforX_2 = 0.469$$

$$IGforX_3 = 0.002$$

$$IGforX_4 = 0.39$$

   Hence, the root node will be X_2 as it has the highest information gain.

   iv. Now X_2 will have two branches 0 and 1 as there are two value types for the X_2 column. To calculate the node connected to both the branches we have to repeat the ID3 algorithm.

   v. To find the root node for the 0 branch coming out of X_2 we have to consider the subset of the original dataset, only considering the rows where the value of X_2 column was 0 and also excluding the X_2 column.

vi. Now applying the ID3 algorithm to the smaller subset we get by excluding X_2 and filtering the rows based on $X_2 = 0$ we get:

$$TotalEntropy = H(S) = 0.982$$

$$IGforX_1 = 0.315$$

$$IGforX_3 = 0.315$$

$$IGforX_4 = 0.98$$

vii. Now we can select X_4 as the root node as it has highest information gain.

viii. Now, to find the root node for the 1 branch coming out of X_2 we have to consider the subset of the original dataset, only considering the rows where the value of X_2 column was 1 and also excluding the X_2 column.

ix. Now applying ID3 algorithm to this subset. We can see that all the value of y in the subset are same. Hence the entropy of the dataset is 0.

x. We can directly return the a leaf node with the value 0.

xi. Now going further down on the left side from the X_4 node. Again repeating the ID3 algorithm this time in a much smaller dataset by exclusion for values of 0 and 1 for the X_4 column.

xii. We can see from the two subset that both have 0 entropy as all the y values are 0 in one table and 1 in another.

xiii. Hence, we can directly return leaf node from both the branches. Branch 0 will return value 0 and branch 1 will return value 1.

Here is the final decision tree obtained: Please refer to Figure 1. in the other PDF.

(b) Function:

$$f(X_1, X_2, X_3, X_4) = X_2||(X_2||X_4)$$

2. (a) i. Starting with the first step of the ID3 algorithm. First check if all labels are same or not. If same then Majority Error(ME) is 0.

ii. If there are multiple labels we have to find the ME of the entire dataset:

$$ME(S) = 5/14 = 0.357$$

iii. After getting the overall ME(S) we need to calculate the Information Gain(IG) of all the features and consider the feature with highest IG as the root node. IG for X_1:

$$IG(S, A) = ME(S) - \sum_{v \in Values(A)} \frac{|S_v|}{S} ME(S_v)$$

$$IGforOutlook = 0.357 - (5/14 * (2/5) + 4/14 * (0) + 5/14 * (2/5)) = 0.07$$

$$IGforTemperature = 0.357 - 0.357 = 0$$

$$IGforHumidity = 0.07$$

$$IGforWind = 0$$

iv. From the above values we can claim Outlook or Humidity can be placed in the root node.

v. Now, applying ID3 algorithm to the subset excluding the Outlook data and considering the remaining features when outlook in ”Sunny”, we get,

$$ME(S) = 2/5 = 0.4$$

$$IGforTemperature = 0.2$$

$$IGforHumidity = 0.4$$

$$IGforWind = 0$$

vi. Humidity will now be the root node for the ”Sunny” branch coming out of Outlook node.

vii. Applying ID3 algorithm to the subset excluding the Outlook data and considering the remaining features when outlook in ”Overcast”, we see that all labels are +. We can return a leaf node with the value + in this case.

viii. Applying ID3 algorithm to the subset excluding the Outlook data and considering the remaining features when outlook in ”Rainy”, we get,

$$ME(S) = 2/5 = 0.4$$

$$IGforTemperature = 0$$

$$IGforHumidity = 0$$

$$IGforWind = 0.4$$

ix. Wind will now be the root node for the ”Rainy” branch coming out of Outlook node.

x. Now filtering the dataset for ”High” humidity, excluding Outlook and Humidity column and applying ID3 algorithm we get all ’-’ labels. Hence, we can return a leaf node with the value ’-’

xi. Now filtering the dataset for ”Normal” humidity, excluding Outlook and Humidity column and applying ID3 algorithm we get all ’+’ labels. Hence, we can return a leaf node with the value ’+’

xii. Now filtering the dataset for ”Low” humidity, excluding Outlook and Humidity column and applying ID3 algorithm we get all ’-’ labels. Hence, we can return a leaf node with the value ’-’

xiii. For the right subtree when the wind is weak all the labels are '+' and when the wind is strong all the labels are '-'. Hence, we will be return the leaf node for the Strong and Weak branches of the Wind node.

xiv. Here, is the final decision tree: Please refer to Figure 2 in the other pdf.

(b) Decision tree using Gini Index:

i. Starting with the first step of the ID3 algorithm. First check if all labels are same or not. If same then return the leaf node with the label value.

ii. If not same then calculate the GI of the entire dataset:

$$GI = 1 - \sum_{k=1}^{k} p_k^2$$

$$= 1 - (9/14)^2 - (5/14)^2$$

$$= 0.459$$

iii. Now calculate Information gain for all the features, we get,

$$IG(S, A) = GI(S) - \sum_{v \in Values(A)} \frac{|S_v|}{S} GI(S_v)$$

$$IG(S, Outlook) = 0.11614$$

$$IG(S, Temp) = 0.0192$$

$$IG(S, Humid) = 0.093$$

$$IG(S, Wind) = 0.03$$

iv. Outlook will now be the root node with "Sunny", "Overcast" and "Rainy" branches coming out.

v. Now filtering the dataset for "Sunny" outlook, excluding Outlook and applying ID3 algorithm to the subset we get,

$$GI = 0.48$$

$$IG(S, Temp) = 0.28$$

$$IG(S, Humid) = 0.48$$

$$IG(S, Wind) = 0.014$$

Now, the root of the subtree branching out from "Sunny" will be "Humid".

vi. Now the subset that is only "overcast" which is the second branch of the "Outlook" node has all labels +. Hence it can be a leaf node.

vii. Now applying ID3 to the subset which are "Rainy", we get,

$$GI = 0.48$$

$$IG(S, Temp) = 0.014$$

$$IG(S, Humid) = 0.02$$

$$IG(S, Wind) = 0.48$$

So, the root of the subtree branching out from "Rainy" will be "Wind".

viii. Now, applying ID3 to the subset of "Wind" each for Weak and Strong branches we get '+' and '-' leaf node from the branches respectively.

ix. Here is the structure of the decision tree: Please refer to Figure 3 in the other PDF.

(c) The tree obtained in the previous two approaches is quite similar to the tree discussed in the class. As we can see that due to the overcast attribute the Information gain get quite high in all three approaches. Beyond the dataset gets quite small because of which the label becomes same for an attribute.

3. (a) Based on the given dataset, we can assume the missing data to be "Sunny" in the Outlook feature.
Calculating the Entropy of the entire dataset we get,

$$H(S) = 0.9182$$

Information Gain for all the features,

$$IG(S, Outlook) = 0.198$$

$$IG(S, Temperature) = 0.039$$

$$IG(S, Humid) = 0.17$$

$$IG(S, Wind) = 0.06$$

The most favorable feature for root node will be "Outlook" here as it has the highest IG

(b) Assuming the missing value as the majority attribute for the same label that the missing row has, we can replace the missing field with "Overcast" attribute.
Calculating the Entropy of the entire dataset we get,

$$H(S) = 0.9182$$

Information Gain for all the features,

$$IG(S, Outlook) = 0.2722$$

$$IG(S, Temperature) = 0.039$$

$$IG(S, Humid) = 0.17$$

$$IG(S, Wind) = 0.06$$

Here also the most favorable feature for root node will be ”Outlook” as it has the highest IG

(c) Using the fractional counts to assume the missing feature, we get,
Row 16: All the values in the proportion of 5/14S
Row 17: All the values in the proportion of 4/14O
Row 18: All the values in the proportion of 5/14R
Based on the assumed data we get the below results:

$$H(S) = 0.9182$$

Information Gain for all the features,

$$IG(S, Outlook) = 0.3217$$

$$IG(S, Temperature) = 0.039$$

$$IG(S, Humid) = 0.17$$

$$IG(S, Wind) = 0.06$$

(d) From the last problem we can confirm that the root node will ”Outlook” as it has highest IG.

i. Now applying IG3 algorithm to the fractional data for all the branches of the ”Outlook” node we get,

$$H(S) = 0.99$$

$$IG(S, Temperature) = 0.525$$

$$IG(S, Humid) = 0.99$$

$$IG(S, Wind) = 0$$

Based on the above results we can consider ”Humid” to be the root node of the ”Sunny” branch.

ii. The ”Overcast” branch of the ”Outlook” node has all labels positive, hence it can return a leaf node with the value ’+’

iii. Applying IG3 algorithm to the ”Rainy” branch, we get,

$$H(S) = 0.99$$

$$IG(S, Temperature) = 0.065$$

$$IG(S, Humid) = 0.0689$$

$$IG(S, Wind) = 0.99$$

Based on the above results we can consider ”Wind” to be the root node of the ”Rainy” branch.

iv. Applying ID3 for the remaining data for the "Humid" node, we can return '-', '+' and '+' leaf node for the "High", "Normal" and "Low" branches respectively as the entropy becomes 0 because there is just one label for each branch.

v. Here is the structure of the decision tree using fractional assumption: Please refer to figure 4 in the other pdf.

4. We know that the Entropy is,

$$Entropy(S) = H(S) = -\sum_{i=1}^{k} p_i \log_2^{p_i}$$

Let's assume that the list of features is represented by S.
The values we are trying to split feature on is represented using 'A'.
So, A_1, A_2, A_3, A_4 ..... are different values of a feature.
We know that the Information Gain is,

$$IG(S, A) = H(S) - \sum_{v \in Values(A)} \frac{|S_v|}{S} Entropy(S_v)$$

$$IG(S, A) = H(S) - \sum_{v \in Values(A)} \frac{|S_v|}{S} Entropy(S_v)$$

$$IG(S, A) = H(S) - \sum_{A=i}^{k} [P_i(-P(S_1|A_i) * \log_2^{P(S_1|A_i)} - P(S_2|A_i) * \log_2^{P(S_2|A_i)} - P(S_3|A_i) * \log_2^{P(S_3|A_i)} ......)]$$

$$IG(S, A) = \sum_{S} -P(S) \log_2^{P(S)} - \sum_{A} P(A) \sum_{S} (-P(S|A) \log_2^{P(S|A)})$$

$$-IG(S, A) = \sum_{S} P(S) \log_2^{P(S)} - \sum_{A} P(A) \sum_{S} (P(S|A) \log_2^{P(S|A)})$$

$$-IG(S, A) = \sum_{S} (\sum_{A} P(A)P(S|A) \log_2^{P(S)}) - \sum_{A} \sum_{S} P(A)(P(S|A) \log_2^{P(S|A)})$$

$$-IG(S, A) = \sum_{S} \sum_{A} [P(A)P(S|A) \log_2^{P(S)} - P(A)P(S|A) \log_2^{P(S|A)}]$$

$$-IG(S, A) = \sum_{S} \sum_{A} P(A)P(S|A) \log_2^{\frac{P(S)}{\log_2^{P(S|A)}}}$$

According to Jensen's inequality if we take the log out we can establish the below inequality,

$$-IG(S, A) <= \log_2^{\sum_S \sum_A \frac{P(A)P(S|A)P(S)}{P(S|A)}}$$

$$-IG(S, A) <= \log_2^{\sum_S \sum_A P(A)P(S)}$$

$$-IG(S,A) <= \log_2^1$$
$$-IG(S,A) <= 0$$
$$IG(S,A) >= 0$$

Hence, proved.

Decision Tree Practice

1. GITHUB Link: https://github.com/avishek04/CS6350

2. (a) Please check github

   (b) Please refer to the table in figure 5. in another PDF.

   (c) From the results we can conclude that the errors in the training data is less than the test data as tree is build using the training data itself.
   It can also be said that as the depth increases, Information Gain based on Entropy performs quite well.
   In the test errors we can also notice the when the depth is 6 there is a slight increase in the error. This could be a result of overfitting.

3. (a) Please refer to the table in Figure 6. in another PDF.

   (b) Please refer to the table in Figure 7. in another PDF.

   (c) We can see that after a certain depth the error remained constant as the tree already reached maximum depth.
   It can also be concluded that the error is higher for the test data compared to the training data.
   On comparing the data with ”unknown” and the one with blank spaces replaced with majority, we can observe the error has increased in the one where we replaced blank spaces with majority as the we assumed those field which reduced the probability of accuracy.