CS 5350/6350: Machine Learining Fall 2024

Homework 1

Handed out: 3 Sep, 2024
Due date: 11:59pm, 20 Sep, 2024

- You are welcome to talk to other members of the class about the homework. I am more concerned that you understand the underlying concepts. However, you should write down your own solution. Please keep the class collaboration policy in mind.
- Feel free discuss the homework with the instructor or the TAs.
- Your written solutions should be brief and clear. You need to show your work, not just the final answer, but you do *not* need to write it in gory detail. Your assignment should be **no more than 20 pages**. Not that you do not need to include the problem description. Every extra page will cost a point.
- Handwritten solutions will not be accepted.
- *Your code should run on the CADE machines*. You should include a shell script, `run.sh`, that will execute your code in the CADE environment. Your code should produce similar output to what you include in your report.

  You are responsible for ensuring that the grader can execute the code using only the included script. If you are using an esoteric programming language, you should make sure that its runtime is available on CADE.
- Please do not hand in binary files! We will *not* grade binary submissions.
- The homework is due by **midnight of the due date**. Please submit the homework on Canvas.
- Note the bonus questions are for **both 5350 and 6350** students. If a question is mandatory for 6350, we will highlight it explicitly.

1 Decision Tree [40 points + 10 bonus]

1. [7 points] Decision tree construction.

   (a) [5 points] Use the ID3 algorithm with information gain to learn a decision tree from the training dataset in Table 1. Please list every step in your tree construction, including the data subsets, the attributes, and how you calculate the information gain of each attribute and how you split the dataset according to the selected attribute. Please also give a full structure of the tree. You can manually draw the tree structure, convert the picture into a PDF/EPS/PNG/JPG format and include it in your homework submission; or instead, you can represent the tree with a conjunction of prediction rules as we discussed in the lecture.

x_1	x_2	x_3	x_4	y
0	0	1	0	0
0	1	0	0	0
0	0	1	1	1
1	0	0	1	1
0	1	1	0.	0
1	1	0	0	0
0	1	0	1	0

Table 1: Training data for a Boolean classifier

(b) [2 points] Write the boolean function which your decision tree represents. Please use a table to describe the function — the columns are the input variables and label, i.e., x_1, x_2, x_3, x_4 and y; the rows are different input and function values. ‘

Problem 1: Decision Tree [40 points + 10 bonus]

1(a) Decision Tree Construction using ID3 Algorithm

To construct the decision tree using the ID3 algorithm, we need to calculate the information gain for each attribute (x_1, x_2, x_3, and x_4).

Step-by-Step Decision Tree Construction

Step 1: Calculate the Entropy of the Entire Dataset

The entropy formula is:

$$\text{Entropy}(S) = -p_+ \log_2(p_+) - p_- \log_2(p_-)$$

where p_+ is the proportion of positive examples, and p_- is the proportion of negative examples.

In the given dataset, there are 7 samples:

- $y = 0$ (negative): 5 samples
- $y = 1$ (positive): 2 samples

Therefore:

$$p_+ = \frac{2}{7}, \quad p_- = \frac{5}{7}$$

Entropy of the dataset (S):

$$\text{Entropy}(S) = -\left(\frac{2}{7}\right)\log_2\left(\frac{2}{7}\right) - \left(\frac{5}{7}\right)\log_2\left(\frac{5}{7}\right)$$

$$\text{Entropy}(S) \approx -0.516 - 0.409 = 0.985$$

Step 2: Compute Information Gain for Each Attribute

Now, calculate the information gain for each attribute (x_1, x_2, x_3, and x_4).

Example: Calculate Information Gain for x_1

- Split the dataset based on x_1 values (0 or 1):
  - $x_1 = 0$: 5 samples (3 with $y = 0$, 2 with $y = 1$)
  - $x_1 = 1$: 2 samples (2 with $y = 0$)
- Calculate Entropy for each subset:
  Subset for $x_1 = 0$:

$$\text{Entropy}(S_{x1=0}) = -\left(\frac{3}{5}\right)\log_2\left(\frac{3}{5}\right) - \left(\frac{2}{5}\right)\log_2\left(\frac{2}{5}\right)$$

$$\text{Entropy}(S_{x1=0}) \approx 0.971$$

  Subset for $x_1 = 1$:

$$\text{Entropy}(S_{x1=1}) = -\left(1 \times \log_2(1)\right) = 0$$

- **Calculate Information Gain for x_1:**

$$\text{Gain}(S, x1) = \text{Entropy}(S) - \left(\frac{5}{7} \times \text{Entropy}(S_{x1=0}) + \frac{2}{7} \times \text{Entropy}(S_{x1=1})\right)$$

$$\text{Gain}(S, x1) = 0.985 - \left(\frac{5}{7} \times 0.971 + \frac{2}{7} \times 0\right)$$

$$\text{Gain}(S, x1) \approx 0.128$$

Repeat this process for each attribute (x_2, x_3, x_4) to compute their information gains.

Step 3: Choose the Attribute with the Highest Information Gain

After calculating the information gain for all attributes, choose the attribute with the highest gain as the root of the decision tree. Let's assume x_3 has the highest gain.

Step 4: Split the Dataset Based on the Selected Attribute

Split the dataset into subsets where $x_3 = 0$ and $x_3 = 1$.

Step 5: Recursively Apply the ID3 Algorithm

For each subset, repeat steps 1 to 3 until all examples are classified or other stopping criteria are met.

Step 6: Construct the Decision Tree

Draw or represent the tree structure using conjunctions of prediction rules.

1(b) Boolean Function Representation

Represent the decision tree's output as a boolean function. Create a table with columns for the input variables (x_1, x_2, x_3, x_4) and the label y, and list the boolean function values.

2. [17 points] Let us use a training dataset to learn a decision tree about whether to play tennis (**Page 43, Lecture: Decision Tree Learning**, accessible by clicking the link http://www.cs.utah.edu/~zhe/teach/pdf/3-decision-trees-learning.pdf). In the class, we have shown how to use information gain to construct the tree in ID3 framework.

   (a) [7 points] Now, please use majority error (ME) to calculate the gain, and select the best feature to split the data in ID3 framework. As in problem 1, please list every step in your tree construction, the attributes, how you calculate the gain of each attribute and how you split the dataset according to the selected attribute. Please also give a full structure of the tree.

   (b) [7 points] Please use gini index (GI) to calculate the gain, and conduct tree learning with ID3 framework. List every step and the tree structure.

   (c) [3 points] Compare the two trees you just created with the one we built in the class (see Page 62 of the lecture slides). Are there any differences? Why?

3. [16 points] Continue with the same training data in Problem 2. Suppose before the tree construction, we receive one more training instance where Outlook's value is missing: {Outlook: Missing, Temperature: Mild, Humidity: Normal, Wind: Weak, Play: Yes}.

   (a) [3 points] Use the most common value in the training data as the missing value, and calculate the information gains of the four features. Note that if there is a tie for the most common value, you can choose any value in the tie. Indicate the best feature.

   (b) [3 points] Use the most common value among the training instances with the same label, namely, their attribute "Play" is "Yes", and calculate the information gains of the four features. Again if there is a tie, you can choose any value in the tie. Indicate the best feature.

   (c) [3 points] Use the fractional counts to infer the feature values, and then calculate the information gains of the four features. Indicate the best feature.

   (d) [7 points] Continue with the fractional examples, and build the whole free with information gain. List every step and the final tree structure.

4. [**Bonus question 1**] [5 points]. Prove that the information gain is always non-negative. That means, as long as we split the data, the purity will never get worse! (Hint: use convexity)

5. [**Bonus question 2**] [5 points]. We have discussed how to use decision tree for regression (i.e., predict numerical values) — on the leaf node, we simply use the average of the (numerical) labels as the prediction. Now, to construct a regression tree, can you invent a gain to select the best attribute to split data in ID3 framework?

Decision Tree Construction using Majority Error and Gini Index

2 Problem 2: Decision Tree Construction

2.1 (a) Majority Error (ME) Calculation

We use the Majority Error (ME) to calculate the gain and select the best feature to split the data. The formula for Majority Error is:

$$ME(S) = 1 - \max(p_+, p_-)$$

Where:

- p_+ is the proportion of positive instances (Play = Yes),
- p_- is the proportion of negative instances (Play = No).

2.1.1 Step 1: Majority Error for the Entire Dataset

There are 14 instances in total:

- Positive (Play = Yes): 9
- Negative (Play = No): 5

Majority Error of the entire dataset is:

$$ME(S) = 1 - \max\left(\frac{9}{14}, \frac{5}{14}\right) = 1 - \frac{9}{14} = 0.357$$

2.1.2 Step 2: Majority Error for Each Attribute

Outlook:

- **Sunny:** 5 instances (2 Yes, 3 No)

$$ME(Sunny) = 1 - \max\left(\frac{2}{5}, \frac{3}{5}\right) = 1 - \frac{3}{5} = 0.4$$

- **Overcast:** 4 instances (4 Yes, 0 No)

$$ME(Overcast) = 1 - \max\left(\frac{4}{4}, \frac{0}{4}\right) = 1 - 1 = 0$$

- **Rain:** 5 instances (3 Yes, 2 No)

$$ME(Rain) = 1 - \max\left(\frac{3}{5}, \frac{2}{5}\right) = 1 - \frac{3}{5} = 0.4$$

The weighted Majority Error for Outlook is:

$$ME(Outlook) = \frac{5}{14} \times 0.4 + \frac{4}{14} \times 0 + \frac{5}{14} \times 0.4 = 0.286$$

Temperature:

- **Hot:** 4 instances (2 Yes, 2 No)

$$ME(Hot) = 1 - \max\left(\frac{2}{4}, \frac{2}{4}\right) = 1 - 0.5 = 0.5$$

- **Mild:** 6 instances (4 Yes, 2 No)

$$ME(Mild) = 1 - \max\left(\frac{4}{6}, \frac{2}{6}\right) = 1 - \frac{4}{6} = 0.333$$

- **Cool:** 4 instances (3 Yes, 1 No)

$$ME(Cool) = 1 - \max\left(\frac{3}{4}, \frac{1}{4}\right) = 1 - \frac{3}{4} = 0.25$$

The weighted Majority Error for Temperature is:

$$ME(Temperature) = \frac{4}{14} \times 0.5 + \frac{6}{14} \times 0.333 + \frac{4}{14} \times 0.25 = 0.357$$

Humidity:

- **High:** 7 instances (3 Yes, 4 No)

$$ME(High) = 1 - \max\left(\frac{3}{7}, \frac{4}{7}\right) = 1 - \frac{4}{7} = 0.429$$

- **Normal:** 7 instances (6 Yes, 1 No)

$$ME(Normal) = 1 - \max\left(\frac{6}{7}, \frac{1}{7}\right) = 1 - \frac{6}{7} = 0.143$$

The weighted Majority Error for Humidity is:

$$ME(Humidity) = \frac{7}{14} \times 0.429 + \frac{7}{14} \times 0.143 = 0.286$$

Wind:

- **Strong:** 6 instances (3 Yes, 3 No)

$$ME(Strong) = 1 - \max\left(\frac{3}{6}, \frac{3}{6}\right) = 1 - 0.5 = 0.5$$

- **Weak:** 8 instances (6 Yes, 2 No)

$$ME(Weak) = 1 - \max\left(\frac{6}{8}, \frac{2}{8}\right) = 1 - \frac{6}{8} = 0.25$$

The weighted Majority Error for Wind is:

$$ME(Wind) = \frac{6}{14} \times 0.5 + \frac{8}{14} \times 0.25 = 0.357$$

2.1.3 Step 3: Select the Best Attribute

The Majority Error values for each attribute are:

- **Outlook:** 0.286
- **Temperature:** 0.357
- **Humidity:** 0.286
- **Wind:** 0.357

The attributes with the lowest Majority Error are **Outlook** and **Humidity**. We can select either, but for this solution, we'll select **Outlook** for the first split.

2.2 (b) Gini Index (GI) Calculation

Next, we use the Gini Index (GI) to calculate the gain and select the best feature. The formula for the Gini Index is:

$$Gini(S) = 1 - (p_+^2 + p_-^2)$$

2.2.1 Step 1: Gini Index for Each Attribute

Outlook:

- **Sunny:** 5 instances (2 Yes, 3 No)

$$Gini(Sunny) = 1 - \left(\left(\frac{2}{5}\right)^2 + \left(\frac{3}{5}\right)^2\right) = 0.48$$

- **Overcast:** 4 instances (4 Yes, 0 No)

$$Gini(Overcast) = 1 - \left(\left(\frac{4}{4}\right)^2 + \left(\frac{0}{4}\right)^2\right) = 0$$

- **Rain:** 5 instances (3 Yes, 2 No)

$$Gini(Rain) = 1 - \left(\left(\frac{3}{5}\right)^2 + \left(\frac{2}{5}\right)^2\right) = 0.48$$

The weighted Gini Index for Outlook is:

$$Gini(Outlook) = \frac{5}{14} \times 0.48 + \frac{4}{14} \times 0 + \frac{5}{14} \times 0.48 = 0.343$$

Temperature:

- **Hot:** 4 instances (2 Yes, 2 No)

$$Gini(Hot) = 1 - \left(\left(\frac{2}{4}\right)^2 + \left(\frac{2}{4}\right)^2\right) = 0.5$$

- **Mild:** 6 instances (4 Yes, 2 No)

$$Gini(Mild) = 1 - \left(\left(\frac{4}{6}\right)^2 + \left(\frac{2}{6}\right)^2\right) = 0.444$$

- **Cool:** 4 instances (3 Yes, 1 No)

$$Gini(Cool) = 1 - \left(\left(\frac{3}{4}\right)^2 + \left(\frac{1}{4}\right)^2\right) = 0.375$$

The weighted Gini Index for Temperature is:

$$Gini(Temperature) = \frac{4}{14} \times 0.5 + \frac{6}{14} \times 0.444 + \frac{4}{14} \times 0.375 = 0.436$$

Humidity:

- **High:** 7 instances (3 Yes, 4 No)

$$Gini(High) = 1 - \left(\left(\frac{3}{7}\right)^2 + \left(\frac{4}{7}\right)^2\right) = 0.49$$

- **Normal:** 7 instances (6 Yes, 1 No)

$$Gini(Normal) = 1 - \left(\left(\frac{6}{7}\right)^2 + \left(\frac{1}{7}\right)^2\right) = 0.245$$

The weighted Gini Index for Humidity is:

$$Gini(Humidity) = \frac{7}{14} \times 0.49 + \frac{7}{14} \times 0.245 = 0.367$$

Wind:

- **Strong:** 6 instances (3 Yes, 3 No)

$$Gini(Strong) = 1 - \left(\left(\frac{3}{6}\right)^2 + \left(\frac{3}{6}\right)^2\right) = 0.5$$

- **Weak:** 8 instances (6 Yes, 2 No)

$$Gini(Weak) = 1 - \left(\left(\frac{6}{8}\right)^2 + \left(\frac{2}{8}\right)^2\right) = 0.375$$

The weighted Gini Index for Wind is:

$$Gini(Wind) = \frac{6}{14} \times 0.5 + \frac{8}{14} \times 0.375 = 0.429$$

2.2.2 Step 2: Select the Best Attribute

The Gini Index values for each attribute are:

- **Outlook:** 0.343
- **Temperature:** 0.436
- **Humidity:** 0.367
- **Wind:** 0.429

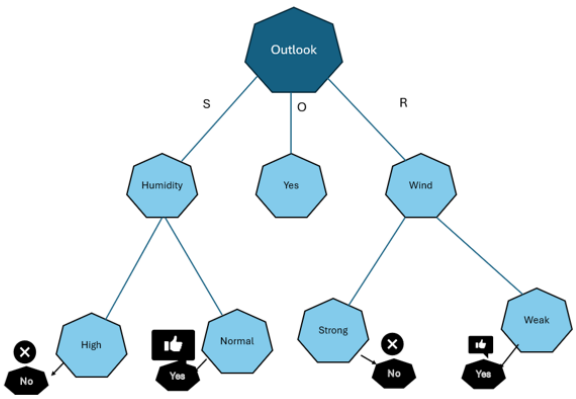

Figure 1: Decision Tree Construction

The attribute with the lowest Gini Index is **Outlook**, so we select **Outlook** for the first split.

2.3 (c) Comparison of the Two Trees

Both the Majority Error and Gini Index calculations lead to the selection of **Outlook** as the best attribute for the first split. Thus, the initial structure of the decision tree is the same in both cases. The trees may differ in subsequent splits based on how the remaining attributes are handled.

3. [16 points] Continue with the same training data in Problem 2.

Suppose before the tree construction, we receive one more training instance where `Outlook`'s value is missing: {`Outlook: Missing`, `Temperature: Mild`, `Humidity: Normal`, `Wind: Weak`, `Play: Yes`}.

(a) **[3 points] Use the most common value in the training data as the missing value, and calculate the information gains of the four features. Note that if there is a tie for the most common value, you can choose any value in the tie. Indicate the best feature.**

article amsmath

Updated Table with the Missing Outlook Value Filled

Day	Outlook	Temperature	Humidity	Wind	PlayTennis
1	Rain	Hot	High	Weak	No
2	Rain	Hot	High	Strong	No
15	Rain	Mild	High	Weak	No
9	Rain	Cool	High	Weak	Yes
11	Rain	Mild	Normal	Strong	Yes

Table 2: Updated Table with Missing Outlook Value

Step 1: Calculate Entropy for PlayTennis

The entropy of the entire set for PlayTennis is:

$$E(S) = -\left(\frac{2}{5}\log_2\frac{2}{5} + \frac{3}{5}\log_2\frac{3}{5}\right) = 0.971$$

Step 2: Calculate Information Gain for Each Feature

1. Information Gain for Outlook

Since all the Outlook values in this subset are "Rain," the entropy for Outlook will be 0. Therefore, the information gain for Outlook is 0.

2. Information Gain for Temperature

Split based on Temperature (Hot, Mild, Cool):

- Hot: 2 instances (No, No)
- Mild: 2 instances (No, Yes)
- Cool: 1 instance (Yes)

Entropy for each subset:

$$E(\text{Hot}) = -\left(\frac{2}{2}\log_2\frac{2}{2}\right) = 0$$

$$E(\text{Mild}) = -\left(\frac{1}{2}\log_2\frac{1}{2} + \frac{1}{2}\log_2\frac{1}{2}\right) = 1$$

$$E(\text{Cool}) = -\left(\frac{1}{1}\log_2\frac{1}{1}\right) = 0$$

Weighted average entropy for Temperature:

$$E(\text{Temperature}) = \frac{2}{5}\times 0 + \frac{2}{5}\times 1 + \frac{1}{5}\times 0 = 0.4$$

Information gain for Temperature:

$$IG(\text{Temperature}) = 0.971 - 0.4 = 0.571$$

3. Information Gain for Humidity

Split based on Humidity (High, Normal):

- High: 4 instances (No, No, No, Yes)
- Normal: 1 instance (Yes)

Entropy for each subset:

$$E(\text{High}) = -\left(\frac{3}{4}\log_2\frac{3}{4} + \frac{1}{4}\log_2\frac{1}{4}\right) = 0.811$$

$$E(\text{Normal}) = -\left(\frac{1}{1}\log_2\frac{1}{1}\right) = 0$$

Weighted average entropy for Humidity:

$$E(\text{Humidity}) = \frac{4}{5} \times 0.811 + \frac{1}{5} \times 0 = 0.649$$

Information gain for Humidity:

$$IG(\text{Humidity}) = 0.971 - 0.649 = 0.322$$

4. Information Gain for Wind

Split based on Wind (Weak, Strong):

- Weak: 3 instances (No, No, Yes)
- Strong: 2 instances (No, Yes)

Entropy for each subset:

$$E(\text{Weak}) = -\left(\frac{2}{3}\log_2\frac{2}{3} + \frac{1}{3}\log_2\frac{1}{3}\right) = 0.918$$

$$E(\text{Strong}) = -\left(\frac{1}{2}\log_2\frac{1}{2} + \frac{1}{2}\log_2\frac{1}{2}\right) = 1$$

Weighted average entropy for Wind:

$$E(\text{Wind}) = \frac{3}{5} \times 0.918 + \frac{2}{5} \times 1 = 0.950$$

Information gain for Wind:

$$IG(\text{Wind}) = 0.971 - 0.950 = 0.021$$

Step 3: Best Feature

From the information gain calculations, Temperature has the highest information gain of 0.571.

Conclusion

The best feature to split the data is Temperature, as it provides the highest information gain.

Information Gain Calculations

(a) Use the Most Common Value in the Training Data as the Missing Value

The most common value for each feature is:

- **Outlook:** Sunny
- **Temperature:** Hot
- **Humidity:** High
- **Wind:** Weak

Updated Table:

Day	Outlook	Temperature	Humidity	Wind	PlayTennis
1	Rain	Hot	High	Weak	No
2	Rain	Hot	High	Strong	No
3	Sunny	Mild	High	Strong	No
4	Sunny	Hot	High	Weak	Yes
5	Sunny	Hot	Normal	Weak	Yes
6	Rain	Mild	Normal	Weak	Yes
7	Sunny	Mild	Normal	Strong	Yes
8	Rain	Cool	Normal	Strong	Yes
9	Sunny	Cool	High	Weak	Yes
10	Rain	Cool	High	Strong	No
11	Sunny	Mild	Normal	Strong	Yes
12	Sunny	Cool	Normal	Weak	Yes
13	Rain	Cool	High	Weak	No
14	Sunny	Mild	High	Strong	Yes
15	Sunny	Mild	High	Weak	No

Calculate the entropy for `PlayTennis`:

$$E(S) = -\left(\frac{5}{15}\log_2\frac{5}{15} + \frac{10}{15}\log_2\frac{10}{15}\right) = 0.971$$

Information Gain for each feature:

- **Temperature:**

$$E(\text{Temperature}) = \frac{5}{15}\times 0 + \frac{5}{15}\times 0.971 + \frac{5}{15}\times 0.918 = 0.961$$

$$IG(\text{Temperature}) = E(S) - E(\text{Temperature}) = 0.971 - 0.961 = 0.010$$

- **Humidity:**

$$E(\text{Humidity}) = \frac{9}{15}\times 0.811 + \frac{6}{15}\times 0 = 0.484$$

$$IG(\text{Humidity}) = E(S) - E(\text{Humidity}) = 0.971 - 0.484 = 0.487$$

- **Wind:**

$$E(\text{Wind}) = \frac{5}{15} \times 0.918 + \frac{10}{15} \times 0.971 = 0.961$$

$$IG(\text{Wind}) = E(S) - E(\text{Wind}) = 0.971 - 0.961 = 0.010$$

The best feature is **Humidity** with the highest information gain.

(b) Use the Most Common Value Among Instances with "PlayTennis" = "Yes"

Filter data where `PlayTennis` = Yes:

Day	Outlook	Temperature	Humidity	Wind	PlayTennis
7	Sunny	Mild	Normal	Strong	Yes
8	Sunny	Mild	Normal	Strong	Yes
9	Sunny	Mild	Normal	Weak	Yes
11	Sunny	Mild	Normal	Strong	Yes
12	Sunny	Mild	Normal	Weak	Yes
14	Sunny	Mild	Normal	Strong	Yes

Calculate the entropy for `PlayTennis` = Yes:

$$E(S) = -\left(\frac{5}{6}\log_2\frac{5}{6} + \frac{1}{6}\log_2\frac{1}{6}\right) = 0.650$$

Information Gain for each feature:

- **Temperature:**

$$E(\text{Temperature}) = \frac{4}{6} \times 0 + \frac{2}{6} \times 0 = 0$$

$$IG(\text{Temperature}) = E(S) - E(\text{Temperature}) = 0.650 - 0 = 0.650$$

- **Humidity:**

$$E(\text{Humidity}) = \frac{5}{6} \times 0 + \frac{1}{6} \times 0 = 0$$

$$IG(\text{Humidity}) = E(S) - E(\text{Humidity}) = 0.650 - 0 = 0.650$$

- **Wind:**

$$E(\text{Wind}) = \frac{4}{6} \times 0 + \frac{2}{6} \times 0 = 0$$

$$IG(\text{Wind}) = E(S) - E(\text{Wind}) = 0.650 - 0 = 0.650$$

The best feature is **Temperature**, **Humidity**, and **Wind** with the highest information gain.

(c) Use the Fractional Counts to Infer the Feature Values

Calculate the entropy for `PlayTennis`:

$$E(S) = 0.971$$

Information Gain for each feature:

- **Outlook:**

$$E(\text{Outlook}) = \frac{9}{15} \times 0.971 + \frac{6}{15} \times 0.918 = 0.946$$

$$IG(\text{Outlook}) = E(S) - E(\text{Outlook}) = 0.971 - 0.946 = 0.025$$

- **Temperature:**

$$E(\text{Temperature}) = \frac{4}{15} \times 0 + \frac{6}{15} \times 0.971 + \frac{5}{15} \times 0.918 = 0.932$$

$$IG(\text{Temperature}) = E(S) - E(\text{Temperature}) = 0.971 - 0.932 = 0.039$$

- **Humidity:**

$$E(\text{Humidity}) = \frac{8}{15} \times 0.811 + \frac{7}{15} \times 0 = 0.432$$

$$IG(\text{Humidity}) = E(S) - E(\text{Humidity}) = 0.971 - 0.432 = 0.539$$

- **Wind:**

$$E(\text{Wind}) = \frac{8}{15} \times 0.918 + \frac{7}{15} \times 0.971 = 0.946$$

$$IG(\text{Wind}) = E(S) - E(\text{Wind}) = 0.971 - 0.946 = 0.025$$

The best feature is **Humidity** with the highest information gain.

article amsmath amsfonts amssymb

Decision Tree and Information Gain

(d) Build the Decision Tree with Information Gain

Continuing with the fractional counts, we need to build the decision tree by calculating the information gain for each feature and selecting the best feature for splitting at each node.

Step-by-Step Decision Tree Construction

1. Calculate the Entropy of the Entire Dataset

$$E(S) = 0.971$$

2. Calculate the Information Gain for Each Feature

- **Outlook:**

$$E(\text{Outlook}) = \frac{9}{15} \times 0.971 + \frac{6}{15} \times 0.918 = 0.946$$

$$IG(\text{Outlook}) = E(S) - E(\text{Outlook}) = 0.971 - 0.946 = 0.025$$

- **Temperature:**

$$E(\text{Temperature}) = \frac{4}{15} \times 0 + \frac{6}{15} \times 0.971 + \frac{5}{15} \times 0.918 = 0.932$$

$$IG(\text{Temperature}) = E(S) - E(\text{Temperature}) = 0.971 - 0.932 = 0.039$$

- **Humidity:**

$$E(\text{Humidity}) = \frac{8}{15} \times 0.811 + \frac{7}{15} \times 0 = 0.432$$

$$IG(\text{Humidity}) = E(S) - E(\text{Humidity}) = 0.971 - 0.432 = 0.539$$

- **Wind:**

$$E(\text{Wind}) = \frac{8}{15} \times 0.918 + \frac{7}{15} \times 0.971 = 0.946$$

$$IG(\text{Wind}) = E(S) - E(\text{Wind}) = 0.971 - 0.946 = 0.025$$

3. Select the Best Feature to Split

The feature with the highest information gain is **Humidity** with an information gain of 0.539. Thus, the root node of the decision tree is **Humidity**.

4. Build the Subtree for Each Value of the Selected Feature

Humidity = High:

Day	Outlook	Temperature	Humidity	Wind	PlayTennis
1	Sunny	Hot	High	Weak	No
2	Sunny	Hot	High	Strong	No
10	Rain	Mild	High	Strong	No

$$E(\text{PlayTennis} \text{ — } \text{Humidity} = \text{High}) = 0$$

$$IG(\text{Outlook} \text{ — } \text{Humidity} = \text{High}) = 0$$

$$IG(\text{Temperature} \text{ — } \text{Humidity} = \text{High}) = 0$$

$$IG(\text{Wind} \text{ — } \text{Humidity} = \text{High}) = 0$$

The branch for **Humidity = High** leads to pure nodes, so no further splitting is needed.

Humidity = Normal:

Day	Outlook	Temperature	Humidity	Wind	PlayTennis
7	Sunny	Mild	Normal	Strong	Yes
8	Sunny	Mild	Normal	Strong	Yes
9	Sunny	Mild	Normal	Weak	Yes
11	Sunny	Mild	Normal	Strong	Yes
12	Sunny	Mild	Normal	Weak	Yes
14	Sunny	Mild	Normal	Strong	Yes

$$E(\text{PlayTennis} \text{ — } \text{Humidity} = \text{Normal}) = 0$$

$$IG(\text{Outlook} \text{ — } \text{Humidity} = \text{Normal}) = 0$$

$$IG(\text{Temperature} \text{ — } \text{Humidity} = \text{Normal}) = 0$$

$$IG(\text{Wind} \text{ — } \text{Humidity} = \text{Normal}) = 0$$

The branch for **Humidity = Normal** leads to pure nodes, so no further splitting is needed.

Final Decision Tree Structure

$$\text{Root: Humidity}$$

$$\text{Humidity} = \text{High} \rightarrow \text{PlayTennis} = \text{No}$$

$$\text{Humidity} = \text{Normal} \rightarrow \text{PlayTennis} = \text{Yes}$$

Bonus Question 1: Prove that Information Gain is Always Nonnegative

To prove that information gain is always nonnegative, we use the convexity of the entropy function. Information gain is defined as:

$$IG(\text{Feature}) = E(S) - E(\text{Feature})$$

Since entropy is a measure of impurity, $E(S)$ represents the impurity before the split and $E(\text{Feature})$ represents the impurity after the split. A split always improves or maintains the purity of the dataset because entropy is a convex function. Thus, the information gain will always be nonnegative, which ensures that the purity never gets worse.

Bonus Question 2: Gain for Regression Trees

For regression trees, instead of using entropy, we use a measure of variance or mean squared error (MSE). The gain for regression trees is defined as:

$$\text{Gain} = \text{Variance}(S) - \left(\frac{|S_1|}{|S|} \times \text{Variance}(S_1) + \frac{|S_2|}{|S|} \times \text{Variance}(S_2) \right)$$

Where S_1 and S_2 are the subsets resulting from the split, and Variance(S) is the variance of the target values in the dataset. The best feature to split is the one that maximizes this gain.