ID3 Decision Tree Learning: Step-by-Step Numerical Example

This document shows a complete worked example of building a decision tree using the ID3 algorithm. We use a small dataset, compute entropy at each step, calculate the information gain for candidate attributes, and select the attribute with the highest gain at each node.

1. Dataset

We consider a simple dataset with three input attributes and one target class:
- Outlook (Sunny, Overcast, Rain)
- Humidity (High, Normal)
- Wind   (Weak, Strong)
- Play   (Yes, No) – target class

Example	Outlook	Humidity	Wind	Play
1	Sunny	High	Weak	No
2	Sunny	High	Strong	No
3	Sunny	Normal	Weak	Yes
4	Overcast	High	Weak	Yes
5	Overcast	Normal	Strong	Yes
6	Rain	High	Weak	Yes
7	Rain	High	Strong	No
8	Rain	Normal	Weak	Yes
9	Sunny	Normal	Strong	Yes
10	Overcast	Normal	Weak	Yes

Out of the 10 examples, 7 have class Play = Yes and 3 have class Play = No.

2. Entropy of the Full Dataset

Let S denote the full training set. There are 7 positive examples (Play = Yes) and 3 negative examples (Play = No). The entropy of S is:

$$\text{Entropy}(S) = -\,p(\text{Yes}) \cdot \log_2 p(\text{Yes}) - p(\text{No}) \cdot \log_2 p(\text{No})$$

where $p(\text{Yes}) = 7/10 = 0.7$ and $p(\text{No}) = 3/10 = 0.3$.

Substituting, we obtain approximately:

$$\text{Entropy}(S) \approx -0.7 \cdot \log_2(0.7) - 0.3 \cdot \log_2(0.3)$$
$$\approx 0.881 \text{ (bits)}$$

3. Choosing the Root Attribute (Information Gain)

We now compute the information gain for each candidate attribute at the root: Outlook, Humidity, and Wind. The attribute with the highest gain will become the root of the decision tree.

3.1 Information Gain for Outlook

We partition S by Outlook into three subsets: Sunny, Overcast, and Rain.

Outlook	$|S_v|$	#Yes	#No	Entropy(S_v)	Weight $|S_v|/|S|$
Sunny	4	2	2	1.000	0.4
Overcast	3	3	0	0.000	0.3
Rain	3	2	1	0.918	0.3

The expected entropy after splitting on Outlook is the weighted sum:

$$\text{Entropy\_outlook} = 0.4 \cdot 1.000 + 0.3 \cdot 0.000 + 0.3 \cdot 0.918$$
$$\approx 0.675$$

Therefore, the information gain for Outlook is:

$$\text{Gain}(S, \text{Outlook}) = \text{Entropy}(S) - \text{Entropy\_outlook}$$
$$\approx 0.881 - 0.675 = 0.206$$

3.2 Information Gain for Humidity

We partition S by Humidity into High and Normal.

Humidity	$|S_v|$	#Yes	#No	Entropy(S_v)	Weight $|S_v|/|S|$
High	5	2	3	0.971	0.5
Normal	5	5	0	0.000	0.5

The expected entropy after splitting on Humidity is:

$$\text{Entropy\_humidity} = 0.5 \cdot 0.971 + 0.5 \cdot 0.000$$
$$\approx 0.485$$

Thus, the information gain for Humidity is:

Gain(S, Humidity) = Entropy(S) – Entropy_humidity

≈ 0.881 – 0.485 = 0.396

3.3 Information Gain for Wind

We partition S by Wind into Weak and Strong.

Wind	$|S\_v|$	#Yes	#No	Entropy(S_v)	Weight $|S\_v|/|S|$
Weak	6	5	1	0.650	0.6
Strong	4	2	2	1.000	0.4

The expected entropy after splitting on Wind is:

Entropy_wind = 0.6 · 0.650 + 0.4 · 1.000

≈ 0.790

Thus, the information gain for Wind is:

Gain(S, Wind) = Entropy(S) – Entropy_wind

≈ 0.881 – 0.790 = 0.091

Comparing the gains:

Gain(S, Outlook) ≈ 0.206

Gain(S, Humidity) ≈ 0.396

Gain(S, Wind)   ≈ 0.091

Humidity has the highest information gain, so it is chosen as the root attribute.

4. Expanding the Tree Below the Root

The root node tests Humidity. This gives two branches:
• Humidity = Normal
• Humidity = High

4.1 Branch: Humidity = Normal

For Humidity = Normal, we have 5 examples, all with Play = Yes. Therefore, this branch becomes a leaf node with class Yes.

4.2 Branch: Humidity = High

For Humidity = High, there are 5 examples: 2 with Play = Yes and 3 with Play = No. We must continue splitting this subset using the remaining attributes (Outlook and Wind).

Example	Outlook	Humidity	Wind	Play

1	Sunny	High	Weak	No
2	Sunny	High	Strong	No
3	Overcast	High	Weak	Yes
4	Rain	High	Weak	Yes
5	Rain	High	Strong	No

Entropy of this subset (Humidity = High) is approximately 0.971 bits.

4.3 Information Gain within Humidity = High

We now compute the information gain for Outlook and Wind using only the five examples where Humidity = High.

4.3.1 Gain for Outlook (given Humidity = High)

Outlook	$|S_v|$	#Yes	#No	Entropy(S_v)	Weight $|S_v|/|S\_high|$
Sunny	2	0	2	0.000	0.4
Overcast	1	1	0	0.000	0.2
Rain	2	1	1	1.000	0.4

Expected entropy after this split:

$$\text{Entropy\_high\_outlook} = 0.4 \cdot 0.000 + 0.2 \cdot 0.000 + 0.4 \cdot 1.000$$
$$= 0.400$$

Information gain for Outlook (within Humidity = High):

$$\text{Gain(High, Outlook)} = 0.971 - 0.400 \approx 0.571$$

4.3.2 Gain for Wind (given Humidity = High)

Wind	$|S_v|$	#Yes	#No	Entropy(S_v)	Weight $|S_v|/|S\_high|$
Weak	3	2	1	0.918	0.6
Strong	2	0	2	0.000	0.4

Expected entropy after splitting on Wind:

$$\text{Entropy\_high\_wind} = 0.6 \cdot 0.918 + 0.4 \cdot 0.000$$

≈ 0.551

Information gain for Wind (within Humidity = High):
  Gain(High, Wind) = 0.971 − 0.551 ≈ 0.420

Outlook gives the higher gain within this subset, so the next test below Humidity = High is Outlook.

4.4 Final Split for Outlook = Rain and Humidity = High

Within the branch Humidity = High and Outlook = Rain, there are 2 examples:
• (Rain, High, Weak, Yes)
• (Rain, High, Strong, No)
This subset has entropy 1.0 (one Yes and one No). Splitting on Wind perfectly separates the classes, so we obtain two leaf nodes:

• Wind = Weak → Play = Yes
• Wind = Strong → Play = No

5. Resulting Decision Tree (Text Description)

The final decision tree learned by ID3 can be described in words as follows:

• If Humidity = Normal, then Play = Yes.
• If Humidity = High and Outlook = Sunny, then Play = No.
• If Humidity = High and Outlook = Overcast, then Play = Yes.
• If Humidity = High and Outlook = Rain and Wind = Weak, then Play = Yes.
• If Humidity = High and Outlook = Rain and Wind = Strong, then Play = No.

This completes a full worked example of the ID3 algorithm with explicit entropy and information gain calculations at each step.