Decision Trees

Background

Entropy, also called *Shannon Entropy*, is denoted by H(S) for a finite set S, and it corresponds to the amount of uncertainty (randomness) in a set.

$$H(S)=\sum_{x\in X} p(x)\log_b \frac{1}{p(x)}=-\sum_{x\in X} p(x)\log_b(x)$$

Intuitively, the entropy tells us about the predictability of a certain event. A fully random exent (e.g., tossing a coin) will give us the maximum entropy (b = 2, then H=1). On the other hand, an event without randomness will give us the lowest possible entropy (0).

Information gain, also referred as the Kullback-Leibler divergence, denoted by IG(S, A) for a set S is the effective change in entropy after deciding on a particular attribute A. It measures the relative change in entropy with respect to the independent variables.

$$IG(S,A)=H(S)-H(S,A)=H(S)-\sum P(x)H(x)$$

H(S) is the Entropy of the entire set, and the second term calculates the Entropy after applying the feature A, where P(x) is the probability of event x.

ID3 algorithm will perform following tasks recursively

1. Calculate the entropy of every attribute A of the data set S.
2. Partition the set S into subsets using the attribute for which the resulting entropy after splitting is minimized, i.e., maximize IG(S,A).
3. Add a decision tree node containing that attribute.
4. Recurse on subsets using the remaining attributes.

Example “Attending today’s lecture”

Let us consider the following data samples:

Day	**Outlook**	**Topic**	**Day before**	**Attending**
D1	Sunny	Interesting	Party	Yes
D2	Sunny	Boring	Party	No
D3	Overcast	Boring	Sleep	Yes
D4	Sunny	Boring	Sleep	No
D5	Rain	Interesting	Party	Yes
D6	Overcast	Interesting	Party	Yes

First, we need to calculate the total entropy of the set with respect to attending or not, i.e., the classification target.

$$H(S)=\frac{-4}{6}\log_2(\frac{4}{6})-\frac{2}{6}\log_2(\frac{2}{6})=0.92$$

Then, let us calculate the information gain of each variable:

$$IG(S,Outlook)=H(S)+\frac{3}{6}(\frac{1}{3}\log_2(\frac{1}{3})+\frac{2}{3}\log_2(\frac{2}{3}))+\frac{2}{6}(\frac{2}{2}\log_2(\frac{2}{2}))+\frac{1}{6}(\frac{1}{1}\log_2(\frac{1}{1}))$$

$$IG(S,Outlook)=0.92-0.46=0.46$$

$$IG(S,Topic)=H(S)+\frac{3}{6}(\frac{3}{3}\log_2(\frac{3}{3}))+\frac{3}{6}(\frac{1}{3}\log_2(\frac{1}{3})+\frac{2}{3}\log_2(\frac{2}{3}))$$

$$IG(S,Topic)=0.92-0.46=0.46$$

$$IG(S,Day\,before)=H(S)+\frac{4}{6}(\frac{1}{4}\log_2(\frac{1}{4})+\frac{3}{4}\log_2(\frac{4}{4}))+\frac{2}{6}(\frac{1}{2}\log_2(\frac{1}{2})+\frac{1}{2}\log_2(\frac{1}{2}))$$

$$IG(S,Day\,before)=0.92-0.54-0.33=0.05$$

These numbers tell us that both “Outlook” and “Topic” would give us the most valuable information to classify the samples. Then, the next step consist of recursively calculate the IG for all partitions. Let us pick “Topic”, then the first partition will already give us a decision:

Day	**Outlook**	**Topic**	**Day before**	**Attending**
D1	Sunny	Interesting	Party	Yes
D5	Rain	Interesting	Party	Yes
D6	Overcast	Interesting	Party	Yes

The other partition looks as follow:

Day	**Outlook**	**Topic**	**Day before**	**Attending**
D2	Sunny	Boring	Party	No
D3	Overcast	Boring	Sleep	Yes
D4	Sunny	Boring	Sleep	No

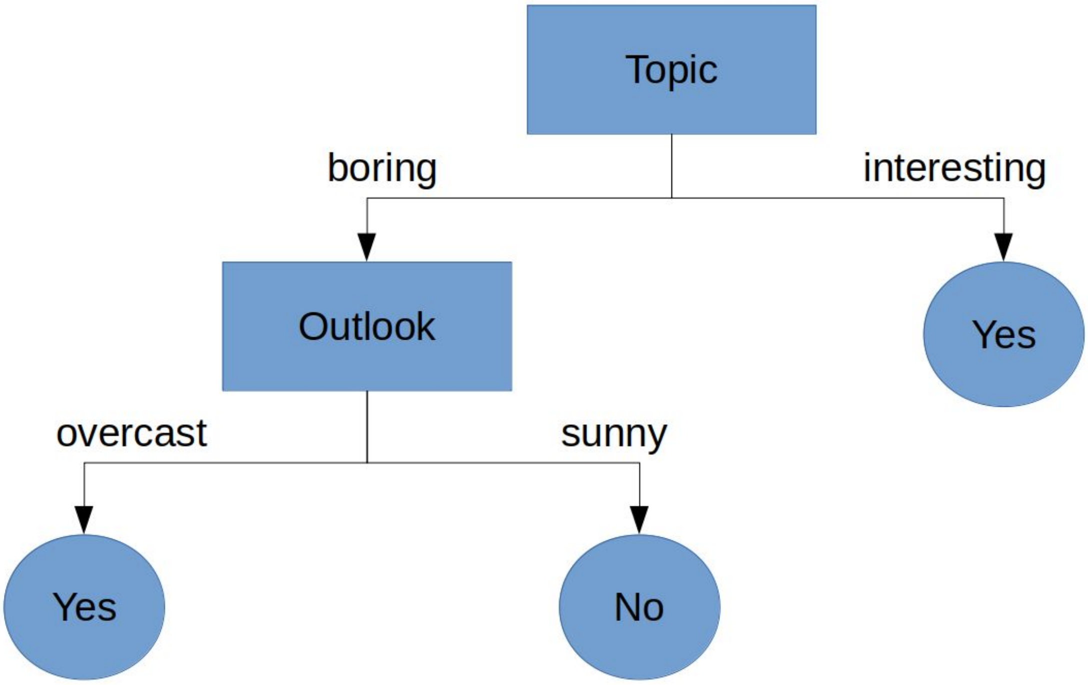
In this case, we should recursively compute the IG.

$$H(S_{boring})=\frac{-1}{3}\log_2(\frac{1}{3})-\frac{2}{3}\log_2(\frac{2}{3})=0.92$$

$$IG(S_{boring},Outlook)=H(S_{boring})+\frac{2}{3}(\frac{2}{2}\log_2(\frac{2}{2}))+\frac{1}{3}(\frac{1}{1}\log_2(\frac{1}{1}))=0.92$$

$$IG(S_{boring},Topic)=H(S_{boring})+\frac{3}{3}(\frac{1}{3}\log_2(\frac{1}{3})+\frac{2}{3}\log_2(\frac{2}{3}))=0.92-0.92=0$$

Then, the decision tree will look like:


Topic
boring
interesting
Outlook
Yes
overcast
sunny
Yes
No