10.8 Inductive Bias

All learning methods have an *inductive bias*. Inductive bias refers to the restrictions that are imposed by the assumptions made in the learning method. For example, in the above discussions we have been assuming that the solution to the problem of road safety can be expressed as a conjunction of a set of eight concepts. This does not allow for more complex expressions that cannot be expressed as a conjunction. This inductive bias means that there are some potential solutions that we cannot explore, and which are, therefore, not contained within the version space we examine.

This may seem like an unfortunate limitation, but in fact inductive bias is essential for learning. In order to have an unbiased learner, the version space would have to contain every possible hypothesis that could possibly be expressed. This would impose a severe limitation: the solution that the learner produced could never be any more general than the complete set of training data. In other words, it would be able to classify data that it had previously seen (as the rote learner could) but would be unable to generalize in order to classify new, unseen data.

The inductive bias of the candidate elimination algorithm is that it is only able to classify a new piece of data if all the hypotheses contained within its version space give the data the same classification. Hence, the inductive bias does impose a limitation on the learning method.

In the 14th century, William of Occam proposed his famous "**Occam's razor**," which simply states that it is best to choose the simplest hypothesis to explain any phenomenon. We can consider this to be a form of inductive bias, which states that the best hypothesis to fit a set of training data is the simplest hypothesis. We will see later how this inductive bias can be useful in learning decision trees.

10.9 Decision-Tree Induction

In Chapter 3, we see a tree that was used to determine which species a particular bird belonged to, based on various observed features of the bird. A variation of this kind of tree, where the leaf nodes are all Boolean values is

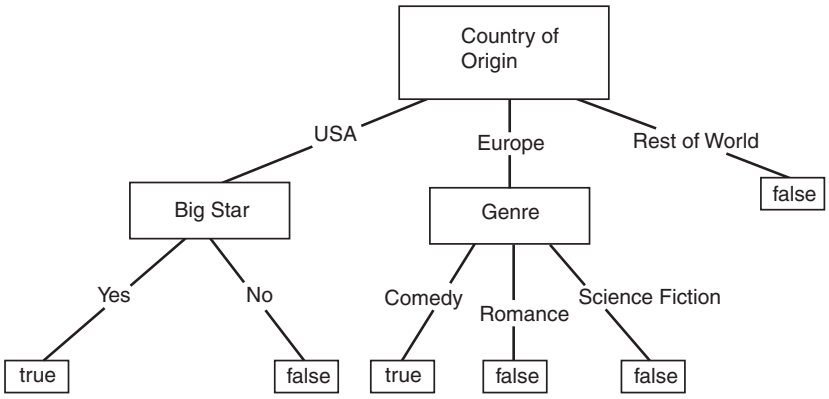

Figure 10.1
A simple decision tree for determining whether or not a film will be a box-office success

called a **decision tree**. A decision tree takes in a set of attribute values and outputs a Boolean decision.

An example of a decision tree is shown in Figure 10.1. This decision tree can be used to determine whether or not a given film will be a success at the box office.

To use the decision tree, we start at the top and apply the question to the film. If the film is made in the United States, we move down the first branch of the tree; if it is made in Europe the second; and if elsewhere then we explore the third branch. The final boxes represent the Boolean value, true or false, which expresses whether a film is a success or not.

According to this extremely simplistic (and possibly somewhat contentious) decision tree, a film can only be a box-office success if it is made in the United States and has a big star, or if it is a European comedy.

Whereas version spaces are able to represent expressions that consist solely of conjunctions, decision trees can represent more complex expressions, involving disjunctions and conjunctions. For example, the decision tree in Figure 10.1 represents the following expression:

```
((Country = USA) ∧ (Big Star = yes)) ∨ ((Country = Europe) ∧
(Genre = comedy))
```

Decision-tree induction (or decision-tree learning) involves using a set of training data to generate a decision tree that correctly classifies the training data. If the learning has worked, this decision tree will then correctly classify new input data as well.

The best-known decision tree induction algorithm is ID3, which was developed by Quinlan in the 1980s.

The ID3 algorithm builds a decision tree from the top down. The nodes are selected by choosing features of the training data set that provide the most information about the data and turning those features into questions. For example, in the above example, the first feature to be noted might be that the country of origin is a significant determinant of whether a film will be a success or not. Hence, the first question to be placed into the decision tree is "what is the film's country of origin?".

The most important feature of ID3 is how the features are chosen. It would be possible to produce a decision tree by selecting the features in an arbitrary order, but this would not necessarily produce the most efficient decision tree. The ID3 algorithm finds the shortest possible decision tree that correctly classifies the training data.

10.9.1 Information Gain

The method used by ID3 to determine which features to use at each stage of the decision tree is to select, at each stage, the feature that provides the greatest **information gain**. Information gain is defined as the reduction in entropy. The entropy of a set of training data, S, is defined as

$$H(S) = -p_1 \log_2 p_1 - p_0 \log_2 p_0$$

where p_1 is defined as the proportion of the training data that includes positive examples, and p_0 is defined as the proportion that includes negative examples. The entropy of S is zero when all the examples are positive, or when all the examples are negative. The entropy reaches its maximum value of 1 when exactly half of the examples are positive and half are negative.

The information gain of a particular feature tells us how closely that feature represents the entire target function, and so at each stage, the feature that gives the highest information gain is chosen to turn into a question.

10.9.2 Example

We will start with the training data given below:

Film	Country of origin	Big star	Genre	Success
Film 1	United States	yes	Science Fiction	true
Film 2	United States	no	Comedy	false
Film 3	United States	yes	Comedy	true
Film 4	Europe	no	Comedy	true
Film 5	Europe	yes	Science fiction	false
Film 6	Europe	yes	Romance	false
Film 7	Rest of World	yes	Comedy	false
Film 8	Rest of World	no	Science fiction	false
Film 9	Europe	yes	Comedy	true
Film 10	United States	yes	Comedy	true

We will now calculate the information gain for the three different attributes of the films, to select which one to use at the top of the tree.

First, let us calculate the information gain of the attribute "country of origin." Our collection of training data consists of five positive examples and five negative examples, so currently it has an entropy value of 1.

Four of the training data are from the United States, four from Europe, and the remaining two from the rest of the world.

The information gain of this attribute is the reduction in entropy that it brings to the data. This can be calculated as follows:

First, we calculate the entropy of each subset of the training data as broken up by this attribute. In other words, we calculate the entropy of the items that are from the United States, the entropy of the items from Europe, and the entropy of the items from the rest of the world.

Of the films from the United States, three were successes and one was not. Hence, the entropy of this attribute is

$$H(\text{USA}) = -(3/4)\log_2(3/4) - (1/4)\log_2(1/4)$$

$$= 0.311 + 0.5$$

$$= 0.811$$

Similarly, we calculate the entropies of the other two subsets as divided by this attribute:

$$H(\text{Europe}) = 1$$

(since half of the European films were successes, and half were not).

$$H(\text{Rest of world}) = 0$$

(since none of these films were successes).

The total information gain is now defined as the original entropy of the set minus the weighted sum of these entropies, where the weight applied to each entropy value is the proportion of the training data that fell into that category. For example, four-tenths of the training data were from the United States, so the weight applied to H(USA) is 4/10 = 0.4.

The information gain is defined as:

$$\text{Gain} = 1 - (0.4 \times 0.811) - (0.4 \times 1) - (0.2 \times 0)$$

$$= 1 - 0.3244 - 0.4 - 0$$

$$= 0.2756$$

Hence, at this stage, the information gain for the "country of origin" attribute is 0.2756.

For the "Big star" attribute

$$H(\text{yes}) = 0.9852$$

$$H(\text{no}) = 1$$

so, the information gain for this attribute is

$$\text{Gain} = 1 - (0.7 \times 0.9852) - (0.3 \times 1)$$

$$= 1 - 0.68964 - 0.3$$

$$= 0.01$$

For the "Genre" attribute

$$H(\text{science fiction}) = 0.918296$$

$$H(\text{comedy}) = 0.918296$$

$$H(\text{romance}) = 0$$

(note that we treat $0 \times \log_2 0$ as 0)

hence, the information gain for this attribute is

$$\text{Gain} = 1 - (0.3 \times 0.918296) - (0.6 \times 0.918296) - (0.1 \times 0)$$

$$= 1 - 0.2754888 - 0.5509776 - 0$$

$$= 0.17$$

Hence, at this stage, the category "Country of origin" provides the greatest entropy gain and so is placed at the top of the decision tree. This method is then applied recursively to the sub-branches of the tree, examining the entropy gain achieved by subdividing the training data further.

10.9.3 Inductive Bias of ID3

ID3's inductive bias is that it tends to produce the shortest decision tree that will correctly classify all of the training data. This fits very well with Occam's razor, which was briefly introduced in Section 10.8. It is not the case that Occam's razor can be applied in all situations to provide the optimal solution: it is, however, the case that ID3 tends to produce adequate results. Additionally, a smaller decision tree is clearly easier for humans to understand, which in some circumstances can be very useful, for example if the need arises to debug the learner and find out why it makes a mistake on a particular piece of unseen data.