MACHINE LEARNING (CSCI 8950): HOMEWORK-2

FEBRUARY 18, 2022

CHINTAN B. MANIYAR
chintanmaniyar@uga.edu

Solution to 1(a):

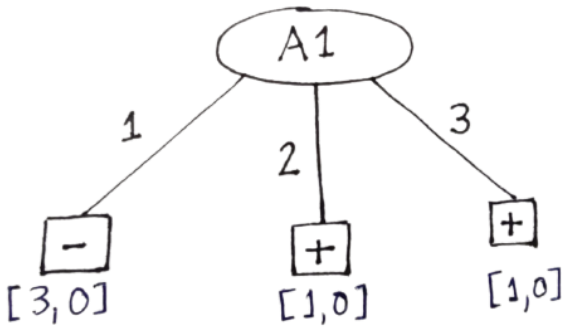


Figure 1: A minimal size decision tree

Figure 1 shows a minimal size decision tree that classifies all the training examples as given in the question, given the label assignment (a=-, b=+, c=-, d=+). Attribute *A1* was chosen as the root node as it showed the highest information gain measure, and all of its child nodes are leaf nodes with 100% purity. More intuitively, the values for attribute *A1* formed a many-to-one correspondence with the labels.

Solution to 1(b): The tree given in Part (a) (Figure 1) would classify the two examples as follows:

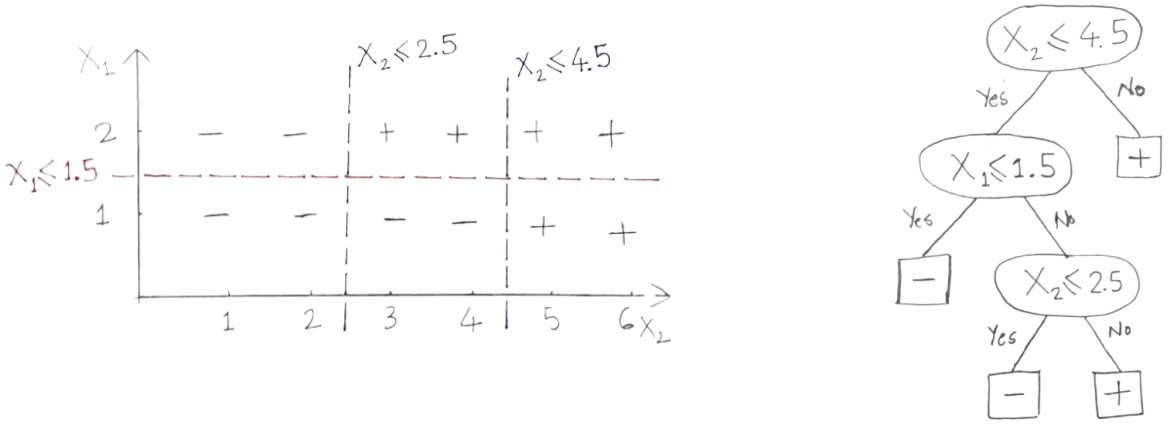
1. (1,2,2,3) → -
2. (3,2,1,1) → +

This classification can be justified with the fact that there is only one decision node, attribute *A1*.

Solution to 1(c): A label assignment that makes attribute *A4* better than attribute *A3* according to the ID3 information gain measure is as follows: (a=+, b=+, c=-, d=+). The motivation for this label assignment is ensuring a many-to-one correspondence between attribute *A4* and labels, and at the same time ensuring a one-to-many correspondence between attribute *A3* and labels, hence increasing the number of decisions to be made if *A3* were the root node. With the proposed label assignment, the ID3 information gain measure for *A4* is 1 and that for *A3* is 0.208.

Solution to 2(a): For the label assignment (a=-, b=+, c=+, d=-), all samples cannot be correctly classified using 3-nearest neighbour rule in terms of Euclidean distance. This is because after any two samples are classified, there are always two sets of 3 nearest neighbours for the third sample, one from each class. For eg, if we classify samples 'a' and 'b' according to the 3NN rule, they will both be labelled as '-' (which is also an incorrect assignment for 'b'). Now, if we try to classify 'c' next, it will have 3 '+' nearest neighbours and 3 '-' nearest neighbours, which is not a majority for one class. Same is the case if we try to classify 'd' next, instead of 'c'. So not only will there be an incorrect label assignment, but we will also not be able to classify all samples correctly using the 3-nearest neighbour rule.

Solution to 2(b): For the label assignment (a=+, b=-, c=+, d=-), all the given samples cannot be classified by a binary decision tree with at most two levels or by a decision tree having at most two decisions along the path to each leaf node. This is because the spread of the samples in the 2D space, X_1X_2, is such that any combination would require at least 3 decisions to separate the '+' and '-' samples, and hence there is no way only two intercepts (or decisions) can separate the classes (Figure 2(a)). Figure 2(b) shows one of the possible decision trees with at least 3 decision nodes in the longest path.



(a) *(b)*

Figure 2: (a) The spread of labels in the 2D space requiring at least 3 intercepts to have a single class in each resulting grid. (b) One of the possible binary decision trees according to the conditions described in the question

Solution to 3(a): In KNN, a higher value of k gives a smoother classification boundary. Moreover, k=1 would likely overfit because it will only consider the one class point, which is closest to the query point; while k=3 considers 3 class points, instead of considering only the one closest to the query point, which means that it will not overfit and rather try to generalize.

Solution to 3(b): A case in which 1NN (k=1) would be better than 3NN (k=3), is an ideal case where: i) there would be no noisy data points at all and, ii) there are only two classes (binary classification problem) with a large margin of separation.

Solution to 3(c): Information Gain Ratio (IGR) is the ratio of Information Gain (IG) and Intrinsic Information (II), which inherently tends to avoid overfitting by considering lesser categories at the root node. For instance, if two attributes with different number of distinct values (branches) have the same entropy, the IG for both will be same, which means we will have to randomly choose one of those attributes as the root node. However, in the same case, IGR will be higher for the attribute which will have lesser distinct values (branches) and hence we will choose that attribute as the root node.

Solution to 3(d): Two-fold cross-validation would potentially be faster than ten-fold cross validation, as in two-fold there will only be two iterations (50% training and 50% testing in each iteration), while in case of ten-fold there will be ten iterations (90% training and 10% testing in each iteration),

Solution to 3(e): An advantage of ten-fold cross-validation over two-fold cross-validation is that ten-fold will inherently carry less bias than two-fold, because the difference in the size of test data and train data would be higher in case of ten-fold (since there will be 10 splits of the data). Hence, as argued by Luntz et. al (1969), two-fold (50% train and 50% test iteratively) may carry more bias because while training the model is unaware of as much as half of the whole dataset. On the other hand, ten-fold (90% train and 10% test iteratively) will carry lesser bias as the model will be aware of almost the whole dataset (90%) while training in every iteration.