Decision Trees

Acer Blake, Alexander Mikheev, Hana Ali, Shreya Desai

Introduction to Machine Learning Coursework 1

Contents

List of Tables

1 Introduction

A common classification algorithm, decision trees are a way of mapping objects into a series of discrete classification functions by means of a tree-based representation. In decision trees, different attributes are examined and compared for all the data points in a dataset, and the aim is to find the best collection of attributes that can be used to accurately label test samples in a future prediction problem.

This report outlines the decision tree algorithm used to classify the WiFi signal data provided, consisting of 2000 samples, 4 classes and 7 attributes per class. This classification problem was approached as a binary problem.

2 Evaluation

We use a random generator to shuffle our dataset before training and testing the decision tree performance. All quoted results are from one experiment, and will change slightly between experiments, but these differences will be negligible.

2.1 Cross Validation Classification Metrics

This section summarises our model's performance before pruning, using different metrics. We perform k-fold cross-validation (using 10 folds) and compute confusion matrices, from which performance metrics are calculated and analysed.

2.1.1 Confusion Matrix

These confusion matrices are averaged for the 10 folds used for k-fold cross validation.

		Predicted Classes			
		1	**2**	**3**	**4**
True Classes	**1**	49.2	0.0	0.3	0.5
	2	0.0	47.6	2.4	0.0
	3	0.3	1.7	47.8	0.2
	4	0.4	0.0	0.1	49.5

(a) Confusion matrix for the clean data before pruning

		Predicted Classes			
		1	**2**	**3**	**4**
True Classes	**1**	38.4	2.7	3.5	4.4
	2	3.0	39.9	4.8	2.0
	3	2.9	2.8	42.9	2.9
	4	5.1	3.0	2.9	38.8

(b) Confusion matrix for the noisy data before pruning

Table 1: Confusion matrices before pruning.

2.1.2 Accuracy

Table 2 summarizes the accuracy of our model, obtained from the confusion matrix as the number of correct predictions over the total number of predictions:

Accuracy Results	
Clean Data	**Noisy Data**
0.9701	0.8000

Table 2: Accuracy on the clean (left) and noisy (right) data (rounded to 4 decimal places).

2.1.3 Recall and Precision Per Class

Table 3 summarizes the recall and precision of our model as obtained from the confusion matrix, where the recall tells us the ratio of true positives to true positives and false negatives per class, and the precision is the ratio of true positives to true and false positives per class.

Recall Results		
	Clean Data	**Noisy Data**
1	0.9840	0.7837
2	0.9520	0.8028
3	0.9560	0.8330
4	0.9900	0.7791

(a) Recall on the clean and noisy datasets before pruning

Precision Results		
	Clean Data	**Noisy Data**
1	0.9860	0.7773
2	0.9655	0.8243
3	0.9447	0.7930
4	0.9861	0.8067

(b) Precision on the clean and noisy datasets before pruning

Table 3: Recall and precision results before pruning.

2.1.4 F1-measures

Table 4 summarizes the general performance of our model as shown by the F1-measure.

F1 Results		
	Clean Data	**Noisy Data**
1	0.9850	0.7805
2	0.9587	0.8135
3	0.9503	0.8125
4	0.9880	0.7926

Table 4: F1-measures on the clean (left) and noisy (right) data.

2.2 Result Analysis

For the clean dataset, rooms 1 and 4 are those detected with highest accuracy (with small percentage difference), reflected by having the highest F1-measures. Rooms 2 and 3 are recognized with a lower accuracy. For the noisy data, we see the opposite; rooms 2 and 3 are recognised with a higher accuracy, while room 1 is classified with the lowest accuracy. For both datasets, there is approximately a 2-3% accuracy difference between the best two and worst two detected rooms.

2.3 Dataset Differences

There is an overall decrease in performance and appropriate class distinction on the noisy dataset in comparison with the clean dataset, which is reflected in the loss of accuracy and F1 measure. This decrease in performance is expected - with more noise, it becomes more difficult for the model to accurately distinguish which feature points are true and which are artificially bloated or diminished, and thus the classification process becomes less accurate overall.

3 Pruning and Evaluation

Just as with the evaluation before pruning, here we use a random generator to shuffle our dataset before training and testing the decision tree performance. All quoted results are from one experiment, and will change slightly between experiments, but these differences will be negligible.

3.1 Cross Validation Classification Metrics after Pruning

This section details the evaluation of our decision tree after pruning. Here, we perform nested k-fold cross validation. We use 10 folds and so run the nested cross validation 90 times.

3.1.1 Confusion Matrix

Table 5 summarizes the confusion matrices of our model on the clean and noisy datasets after pruning the Decision Tree.

		Predicted Classes			
		1	**2**	**3**	**4**
True Classes	**1**	49.6	0.0	0.3	0.0
	2	0.0	47.8	2.3	0.0
	3	0.8	1.9	47.0	0.3
	4	0.4	0.0	0.3	49.2

(a) Confusion matrix for the clean data after pruning

		Predicted Classes			
		1	**2**	**3**	**4**
True Classes	**1**	43.5	1.3	1.9	2.3
	2	1.9	43.5	2.9	1.3
	3	2.1	3.0	44.6	1.9
	4	2.4	1.5	1.8	44.1

(b) Confusion matrix for the noisy data after pruning

Table 5: Confusion matrices for the clean and noisy datasets after pruning.

3.1.2 Accuracy after Pruning

Table 6 summarizes the accuracy of our model after pruning.

Accuracy Results	
Clean Data	**Noisy Data**
0.9682	0.8784

Table 6: Post-pruning accuracy on the clean (left) and noisy (right) data (rounded to 4 decimal places).

3.1.3 Recall and Precision after Pruning

Table 7 summarizes the recall and precision of our model post-pruning.

Recall Results		
	Clean Data	**Noisy Data**
1	0.9924	0.8875
2	0.9540	0.8761
3	0.9409	0.8651
4	0.9853	0.8853

(a) Recall on the clean and noisy datasets after pruning.

Precision Results		
	Clean Data	**Noisy Data**
1	0.9770	0.8721
2	0.9613	0.8815
3	0.9407	0.8700
4	0.9937	0.8903

(b) Precision on the clean and noisy datasets after pruning.

Table 7: Recall and precision results after pruning.

3.1.4 F1-measures

Table 8 summarizes the general performance of our model as shown by the F1-measure.

F1 Results		
	Clean Data	**Noisy Data**
1	0.9847	0.8797
2	0.9576	0.8788
3	0.9408	0.8676
4	0.9895	0.8878

Table 8: F1-measures on the clean (left) and noisy (right) data.

3.2 Result Analysis after Pruning

On average, the F1-measure decreases by 0.23% for the clean dataset after pruning, and increases by an average 7.87% for the noisy dataset after pruning. This is consistent with what we expect; a noisy dataset would result in more unnecessary splits on noise. Therefore, by pruning, we ensure the model does not become too complex and overfit the data, and only maintains the nodes that have the majority vote. This results in better classification.

3.3 Depth Analysis

Table 9 shows the average depth of the decision tree generated for both datasets across the folds, before and after pruning.

Decision Tree Depths		
	Clean Data	**Noisy Data**
Before pruning	12.0	17.9
After pruning	8.5	13.9

Table 9: Depths of the decision tree on the clean (left) and noisy (right) datasets before and after pruning the decision tree (rounded to 1 decimal place).

The change in average depth is a consistent decrease by 4 after pruning. A deeper tree represents a more complex model. Therefore, a higher maximal depth may indicate that the model has overfitted to the training data. This would lead to a high training accuracy, but a lower prediction accuracy due to the model's inability to generalise on test data.