Discussion

by looking at the leaf nodes we see this pattern repeated frequently

```
<node classes="unacc:1,acc:3" entropy="0.406" lug_boot="med">
	<node classes="unacc:1" doors="2" entropy="0.0">unacc</node>
	<node classes="acc:1" doors="3" entropy="0.0">acc</node>
	<node classes="acc:1" doors="4" entropy="0.0">acc</node>
	<node classes="acc:1" doors="5more" entropy="0.0">acc</node>
 </node>
```

we see here that we end up with 4 instances with the attribute “lug_boot” valued “med” 3 out of them are classified “acc” and only one of them classified “unacc”. For that anomaly we had to partition the data again over another attribute “doors”. This one anomaly can be just noise in the data and when we take decisions based on it, we are more vulnerable to prediction errors when the algorithm is applied on unseen data. We need to divide the data into training data, validation data and test data. Use the training data to build the tree. Use the validation data to prune the tree which means eliminate the unnecessary nodes that may be based only on noise in the data by comparing the tree performance with the real classification on unseen data “the validation set”. We use the test data to test the final accuracy of the pruned tree.