

1 Entropy

- Entropy is calculated via an equation (fill it in).
- Entropy is a measure of how disparate the data pool in question is wrt. to classification. A zero entropy describes the a data pool that is fully aligned that is, every data point in this pool is classified the same. A entropy of one describes a data pool that is perfectly polarised wrt. to the set of classifications available, there is a equal distributed of data points that is classified for each classification available.
- The properties overfitting and underfitting within the context of decision trees are ways of describing how well the model "generalises" wrt. to the testing data. More concretely, we can look at the "training error" and the "testing error". A model that has very low "training error", but a relatively large "testing error" is said to be "overfit" to the training data. While a model that has a relatively large "training error" alongside the large "testing error" is said to be "underfitting" the data.
- In the grand scheme of things, decision tree's are incredibly quick to build. But wrt. to just the four given, we expect the baseline rule tree would be fastest to build and quickest to evaluate an observation. Restricting your node splitting to only one variable means the decision tree builder doesn't need to iterate through all possible values to find inequalities with best information gain. And hence this would be quicker to build than an unrestricted unpruned tree build. For pruned trees, there's two cases to consider. Post and pre-pruned. Pre-pruned trees with pre-defined restrictions (such as max depth) etc... may terminate in build earlier than an unpruned tree hence be quicker to build. However post-pruned trees are strictly the same to an unpruned tree in initial build, but includes a follow up step of arbitrarily (on the post-pruning technique employed) iterating through each leaf node and potentially "trimming" them. Hence out of all options here, the post-pruned decision tree would take longest to build. **Given an observation and evaluating it though** is a different story. Simply put, the tree with the overall least amount of nodes would be the quickest. (and hence regardless of build time) there are less conditions to traverse then that of a bigger tree.
- Considering all four approaches operate on the same data, the baseline rule tree would certainly be most prone to underfitting, followed by the single variable decision tree. Ofcourse it depends on the data given, and the variable chosen on that data, but given the capability of these trees are just based on one variable of potential thousands, it would struggle the most wrt. to "generalizing" well. Unpruned tree's in the most basic sense are by definition "encouraged" to overfit the given "training" data. Whereas any sort of pruned tree, post or pre, takes an overfitted model and "trims" it in a way to generalise better. Ofcourse we can be overly obtuse

in pruning and prune this tree back down to a stump and hence we must say, the pruned trees sit somewhere between "underfit" and "overfit".

2 Manually creating a decision tree

Need to review the iterative process here but the tutorial kinda outlines the process...