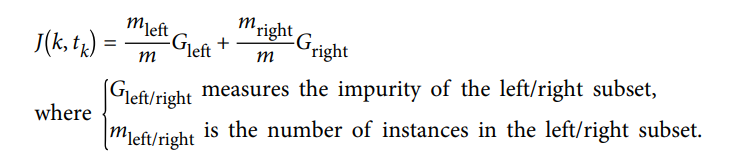
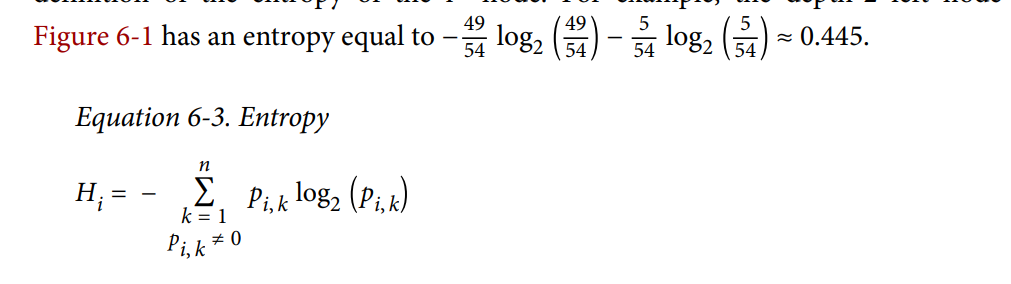
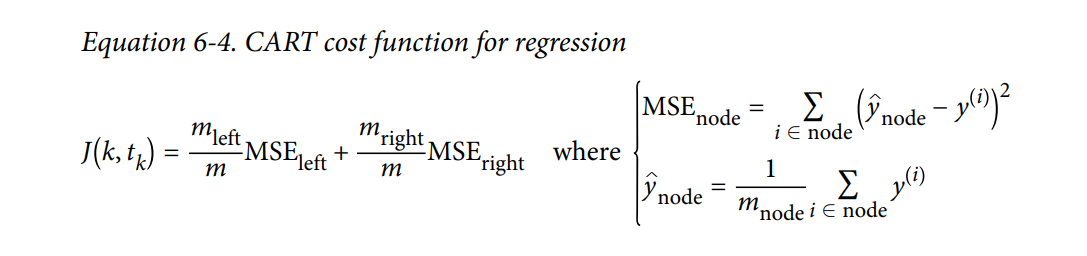
* The CART Training Algorithm
  + The algorithm first splits the training set in two subsets using a single feature k and a threshold tk (e.g., “petal length ≤ 2.45 cm”).
  + It searches for the pair ( k, tk ) that produces the purest subsets (weighted by their size).
  + The cost function that the algorithm tries to minimize is given by,



* It splits the subsets using the same logic, then the sub-subsets and so on, recursively. It stops recursing once it reaches the maximum depth ( defined by the max\_depth hyperparameter ), or if it cannot find a split that will reduce impurity.
* Entropy
  + A set’s entropy is zero when it contains instances of only one class. Equation of entropy of node is given by,



* Gini impurity is slightly faster to compute, so it is a good default.
* Gini impurity tends to isolate the most frequent class in its own branch of the tree, while entropy tends to produce slightly more balanced trees.
* Regularization Hyperparameters
  + **max\_depth** : The default value is None, which means unlimited. Reducing max\_depth will regularize the model and thus reduce the risk of overfitting.
  + **min\_samples\_split :** the minimum number of samples a node must have before it can be split.
  + **min\_samples\_leaf :** The minimum number of samples a leaf node must have.
  + **max\_leaf\_nodes :** maximum number of leaf node
  + **max\_features :** maximum number of features that are evaluated for splitting at each node
  + Increasing min\_\* hyperparameters or reducing max\_\* hyperparameters will regularize the model.
* Regression
  + The CART algorithm works mostly the same way as earlier, except that instead of trying to split the training set in a way that minimizes impurity, it now tries to split the training set in a way that minimizes the MSE.
  + CART Cost function for regression,

****