Reduced Error Pruning for Decision Trees

- Split training data into a training and pruning set Often 2/3 train and 1/3 pruning
- Train Decision Tree.
- Bottom up (from leaves) see if leaving out a test makes accuracy better or leaves it the same. If yes, prune test (node).
- Accuracy measured how? Count of correct vs. incorrect using the pruning data.

Reduced Error Pruning

- Continue examining nodes (tests) in a bottom up fashion until nothing more can be pruned.
- Use pruned tree on test data (which is not in the training or pruning data).
- This is similar to pruning rules with Ripper.

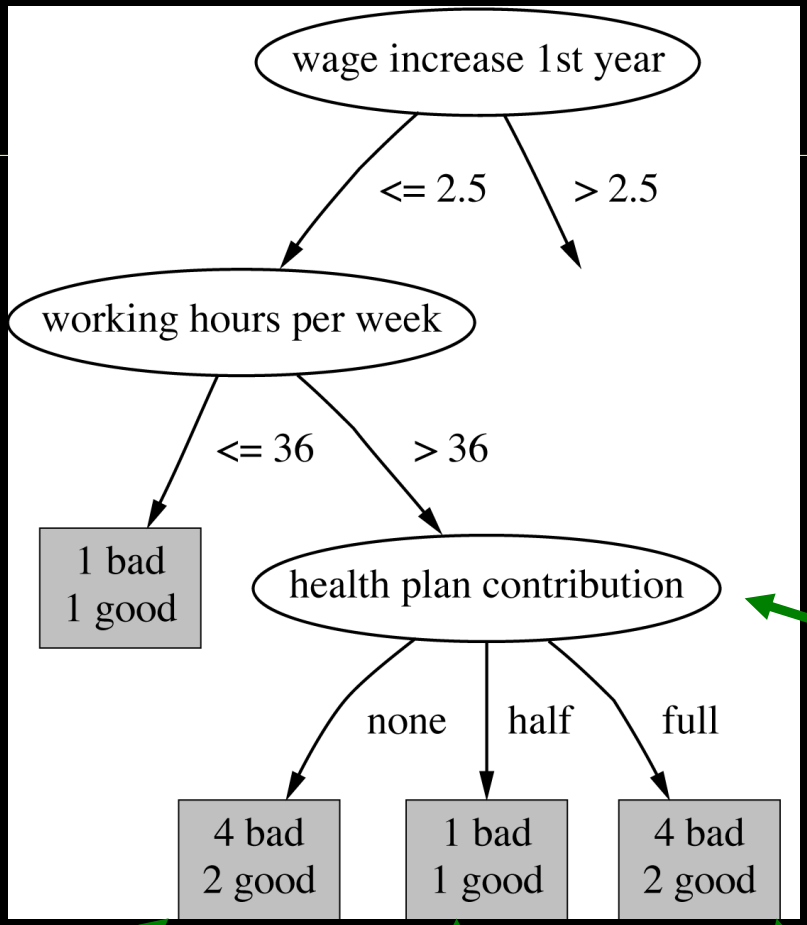
C4.5's method

- Error estimate for subtree is weighted sum of error estimates for all its leaves
- Error estimate for a node:

$$e = (f + \frac{z^2}{2N} + z\sqrt{\frac{f}{N} - \frac{f^2}{N} + \frac{z^2}{4N^2}})/(1 + \frac{z^2}{N})$$

- If c = 25% then z = 0.69 (from normal distribution)
- f is the error on the training data
- N is the number of instances covered by the leaf

Example



f = 5/14
e = 0.46
e < 0.51
so prune!

f=0.33
e=0.47

f=0.5
e=0.72

f=0.33
e=0.47

Combined using ratios 6:2:6 gives 0.51