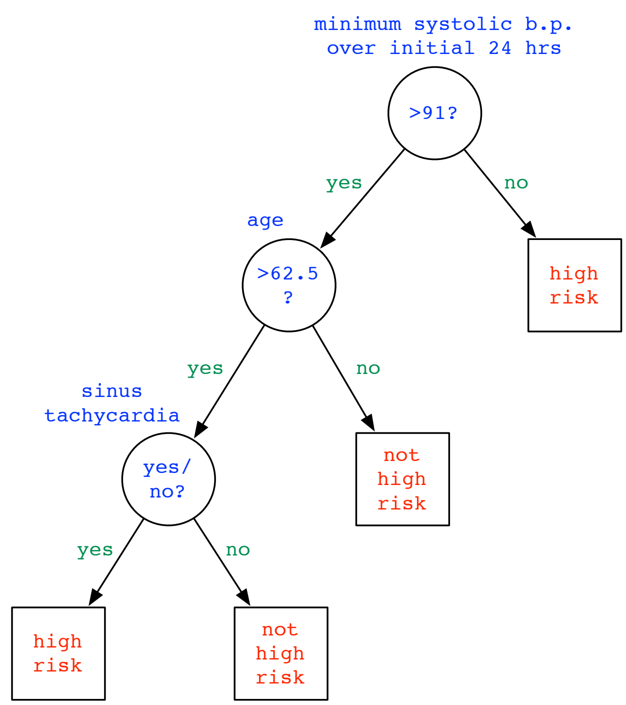
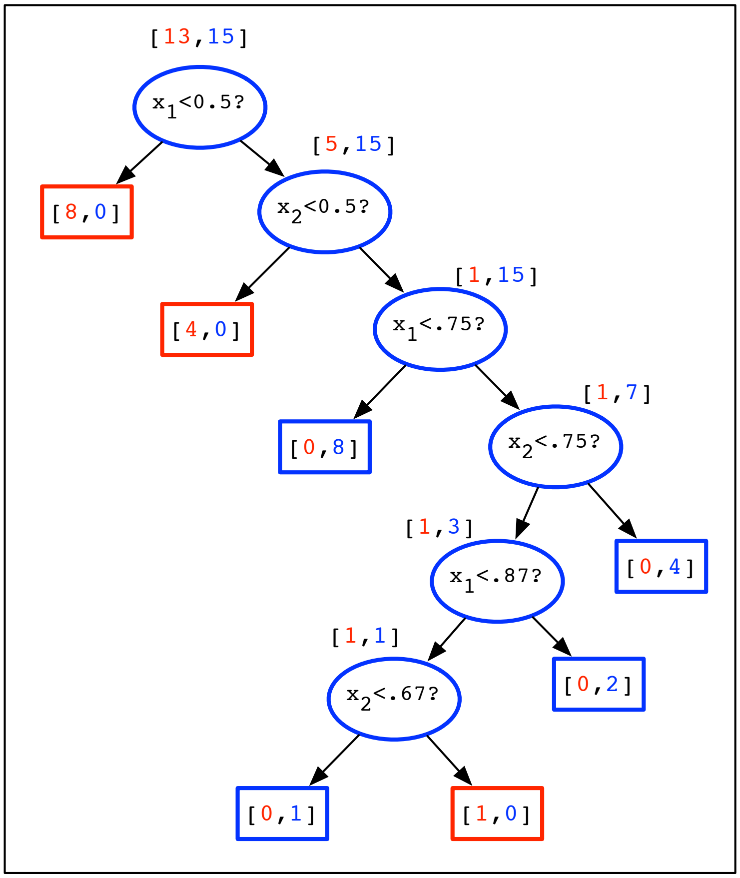
Decision trees

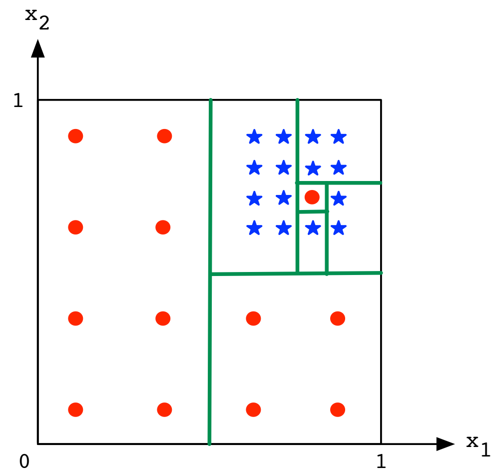
Decision trees

UCSD Medical Center (1970s):
identify patients at risk of dying
within 30 days after heart attack.

Data set:
215 patients.
37 (=20%) died.
19 features.







Building a decision tree

Greedy algorithm: build tree top-down.

- Start with a single node containing all data points
- Repeat:
  - Look at all current leaves and all possible splits
  - Choose the split that most decreases the uncertainty in prediction

We need a measure of **uncertainty in prediction**.

Uncertainty in prediction

Say there are two labels:

$+$ label $\quad p$ fraction of the points
$-$ label $\quad (1-p)$ fraction of the points

What uncertainty score should we give to this?

1. Misclassification rate

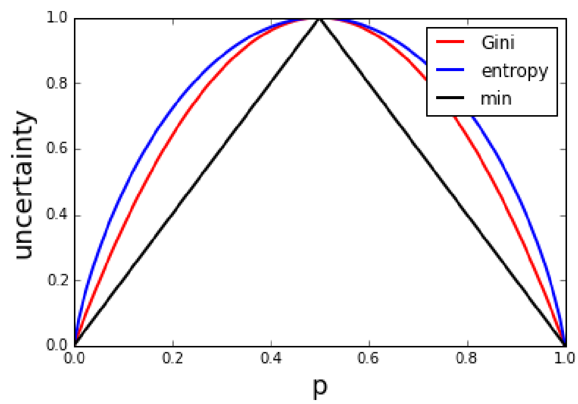
$$\min\{p, 1-p\}$$

2. Gini index

$$2p(1-p)$$

3. Entropy

$$p \log \frac{1}{p} + (1-p) \log \frac{1}{1-p}$$



Uncertainty: k classes

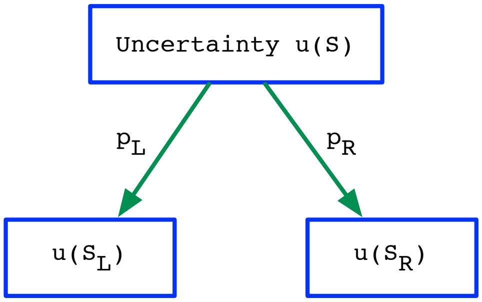
Suppose there are k classes, with probabilities $p_1, p_2, \ldots, p_k$.

	$k = 2$	General k
Misclassification rate	$\min\{p, 1-p\}$	$1 - \max_i p_i = 1 - \lVert p \rVert_\infty$
Gini index	$2p(1-p)$	$\sum_{i \neq j} p_i p_j = 1 - \lVert p \rVert^2$
Entropy	$p \log \frac{1}{p} + (1-p) \log \frac{1}{1-p}$	$\sum_i p_i \log \frac{1}{p_i}$

Benefit of a split

Let $u(S)$ be the uncertainty score for a set of labeled points S.

Consider a particular split:



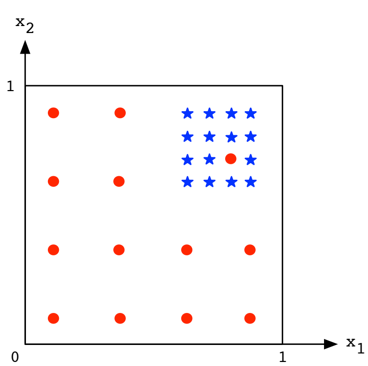
Of the points in S:

- p_L fraction go to S_L
- p_R fraction go to S_R

Benefit of split = reduction in uncertainty:

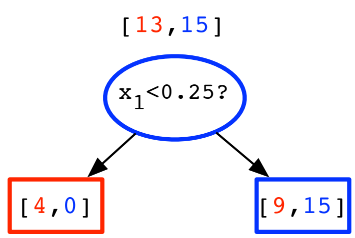
$$\left(u(S) - \underbrace{(p_L\, u(S_L) + p_R\, u(S_R))}_{\text{expected uncertainty after split}} \right) \times |S|$$

Benefit of a split: example

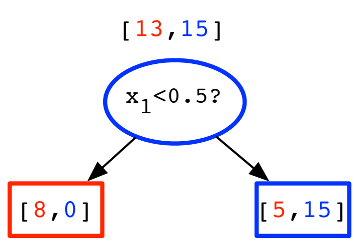


Initial Gini uncertainty:

$$2 \times \frac{13}{28} \times \frac{15}{28}$$



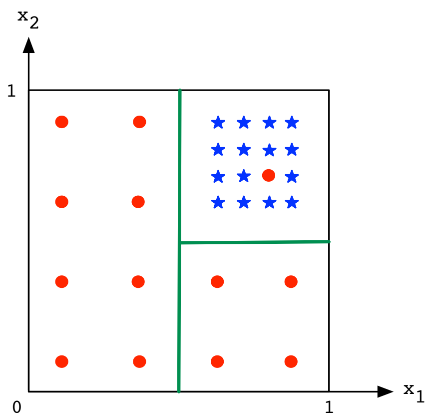
$$p_L u_L + p_R u_R = \frac{4}{28} \cdot 0 + \frac{24}{28} \cdot 2 \cdot \frac{9}{24} \cdot \frac{15}{24} = \frac{45}{112}$$

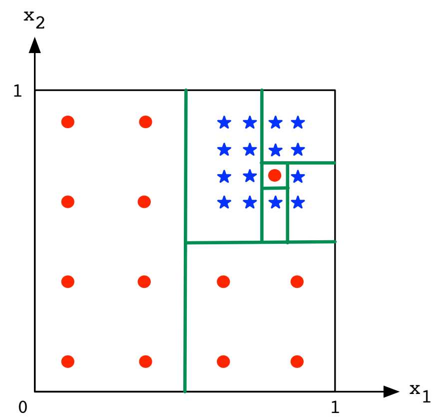


$$p_L u_L + p_R u_R = \frac{8}{28} \cdot 0 + \frac{20}{28} \cdot 2 \cdot \frac{5}{20} \cdot \frac{15}{20} = \frac{30}{112}$$

Overfitting?

Go back a few steps...

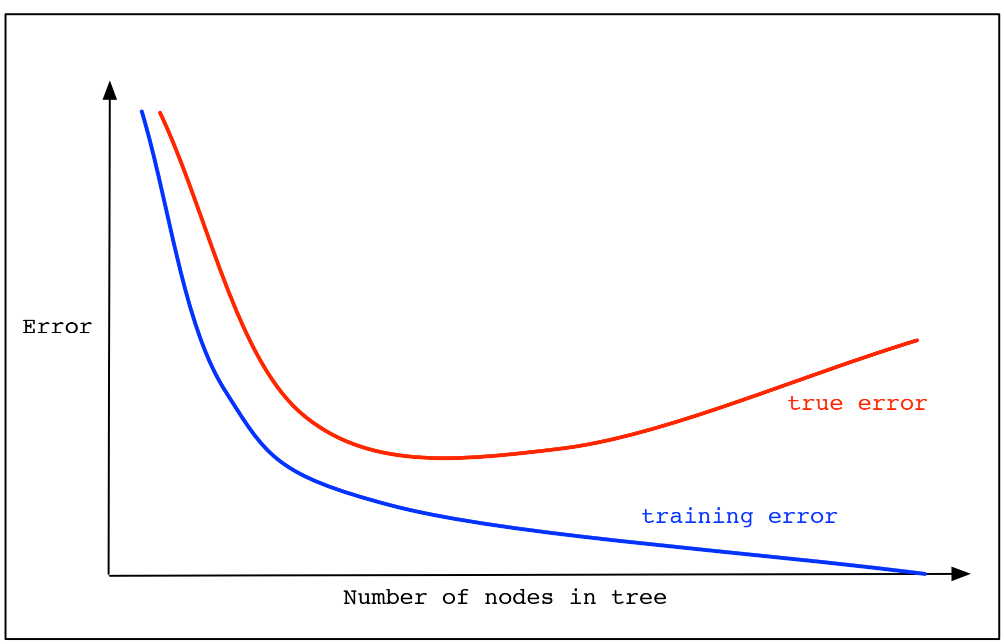




Final partition does better on training data, but is more complex.
That one point might have been an outlier anyway.

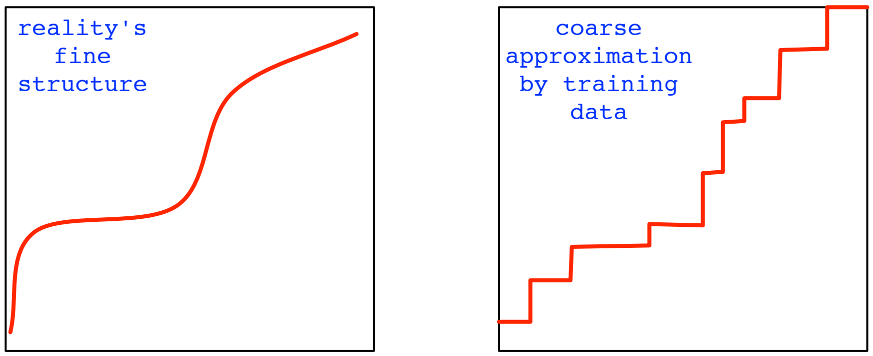
We have probably ended up **overfitting** the data.

Overfitting: picture



Overfitting: perspective

- The training data reflects an underlying reality, so it helps us.
- But it also has chance structure of its own – we must avoid modeling this.



Decision tree properties

A very expressive family of classifiers:

- Can accommodate any type of data: real, Boolean, categorical, ...
- Can accommodate any number of classes
- Can fit any data set

But this also means that there is serious danger of overfitting.

Building a decision tree

- Start with a single node containing all data points
- Repeat:
  - Look at all current leaves and all possible splits
  - Choose the split with the greatest benefit

When to stop?

- When each leaf is pure?
- When the tree is already pretty big?
- When each leaf has uncertainty below some threshold?

Common strategy: keep going until leaves are pure.
Then, shorten the tree by **pruning**, to correct for overfitting.