# Decision trees

Sanjoy Dasgupta

University of California, San Diego

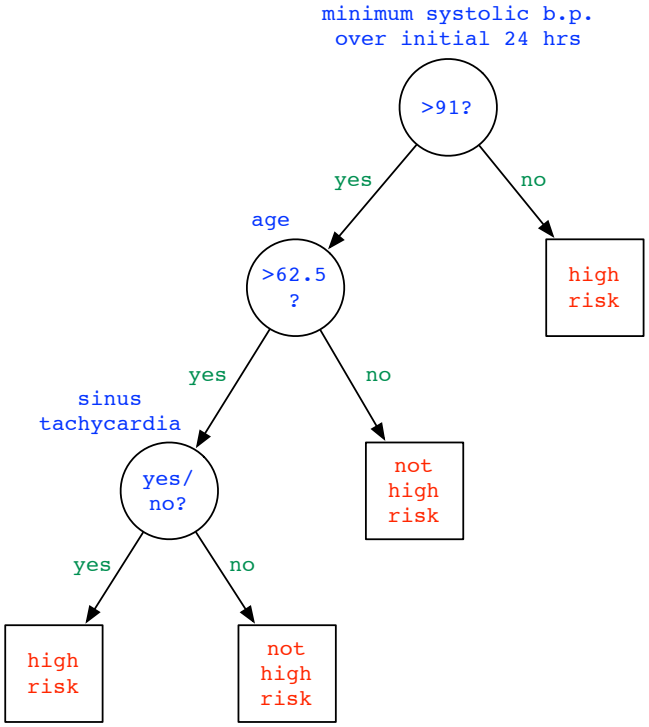
## Topics we'll cover

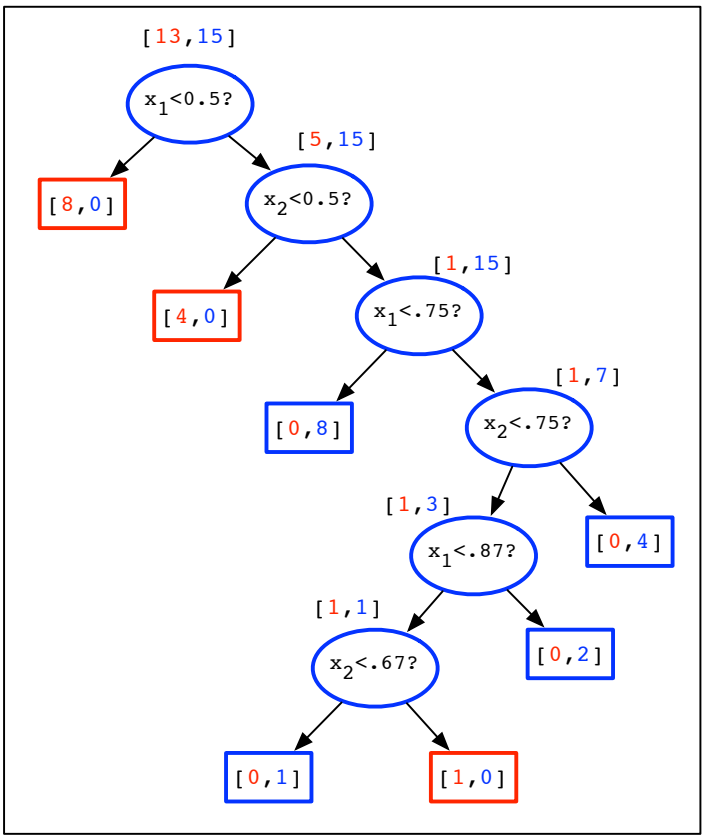
1. The form of a decision tree classifier
2. A top-down learning algorithm
3. Overfitting

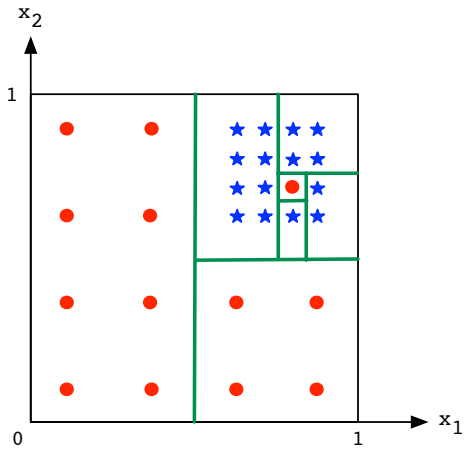
# Decision trees

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: summary

Greedy algorithm: build tree top-down.

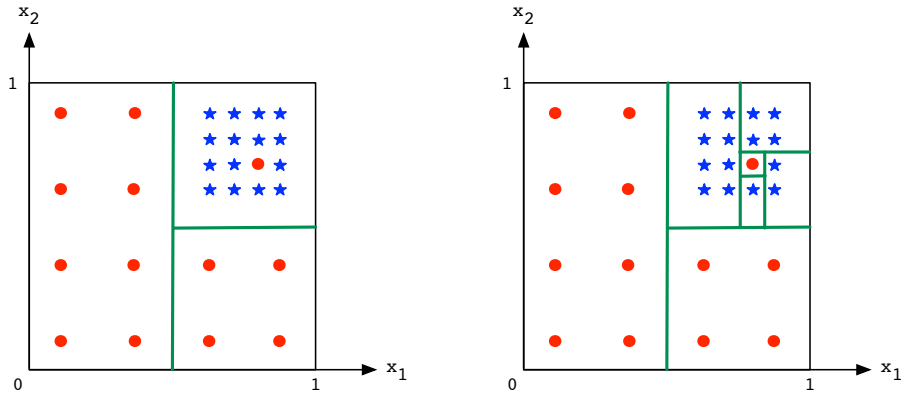
- Start with a single node containing all data points
- Repeat:
  - Look at all current leaves and all possible splits
  - Choose the split that most reduces **uncertainty in prediction**. Several ways to quantify this: Gini index, entropy, etc.

When to stop?

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

# Overfitting?

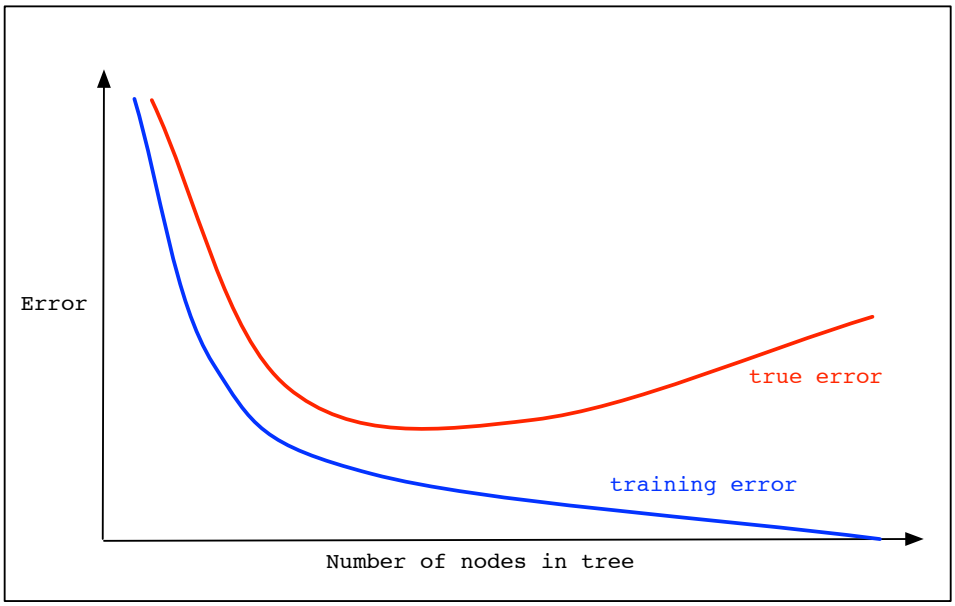
Go back a few steps...




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

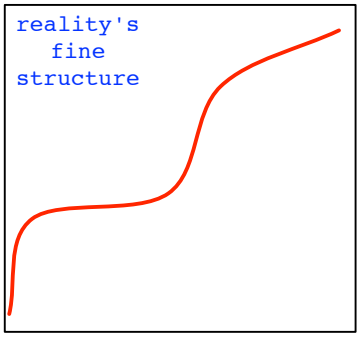
We have probably ended up **overfitting** the data.

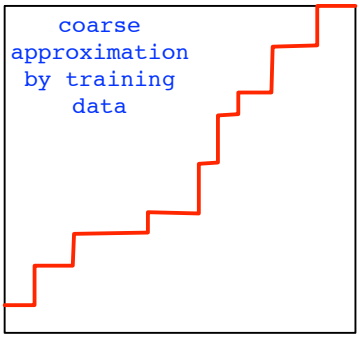
## Overfitting: picture




## 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 flexible and expressive family of classifiers:

- Can accommodate any type of data: numeric or categorical
- Can accommodate any number of classes
- Can fit any data set

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

Common strategies:

- Stop when leaves are pure enough
- Stop when tree reaches a certain size
- Grow tree, then **prune** with a validation set