

[QuantFS] Is Here to Stay

Thanks for coming despite les exams

Agenda

- ▣ Hi, I'm Varun! This is the Board!
- ▣ Website Stuff
 - ▣ Forum
 - ▣ Features – no funny line here ☹️
- ▣ Membership Benefits and Badafits
- ▣ ICC...Bow Chicka Bow Wow!
- ▣ Speakers, Workshops, & Food / Naptime
- ▣ Going Forward <(*.*<) <(*.*)> (>*.*)>
- ▣ DJIA WTFWTFWTFWTF!!!!

DJI WTF HAPPENED TODAY?

Compy that calculates DJI fails →
1-hour lapse w/o good calculation →
Backup goes back online →
Backlog kicks in, traders get scared →
Dow Falls Fast →
Program Trading Kicks in →
Dow Falls Faster →
*Morals? No LEGACY; CompSci Need

Decision Trees

A Natural Approach to Finance

Steven Ambadjes &
Igor Schmertzler

Decision Trees

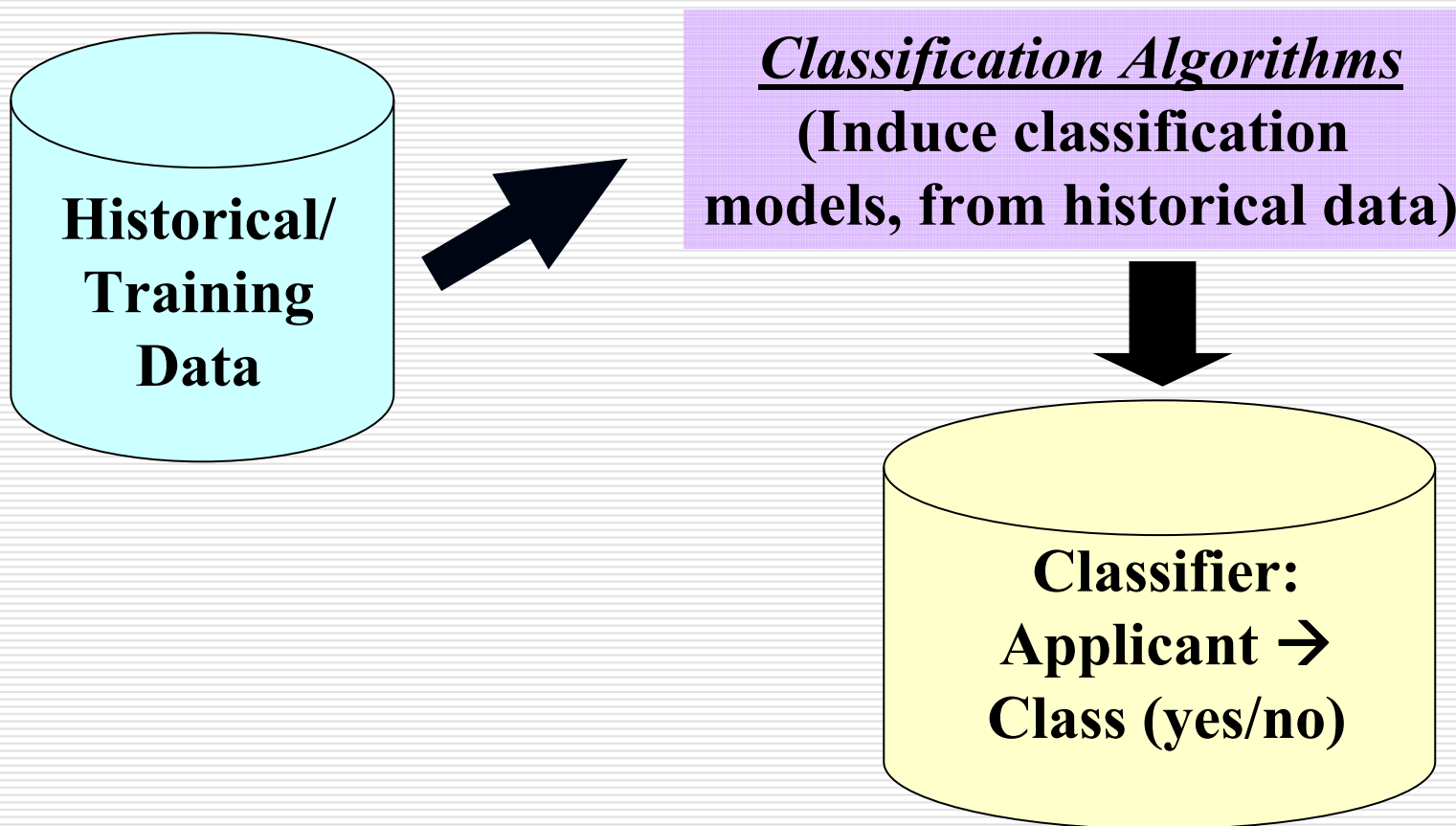
Agenda

- What is a decision tree?
- How is a decision tree created?
- How do we build the best tree?
- Entropy vs. information gain
- How to overcome sensitivity
- Standard applications

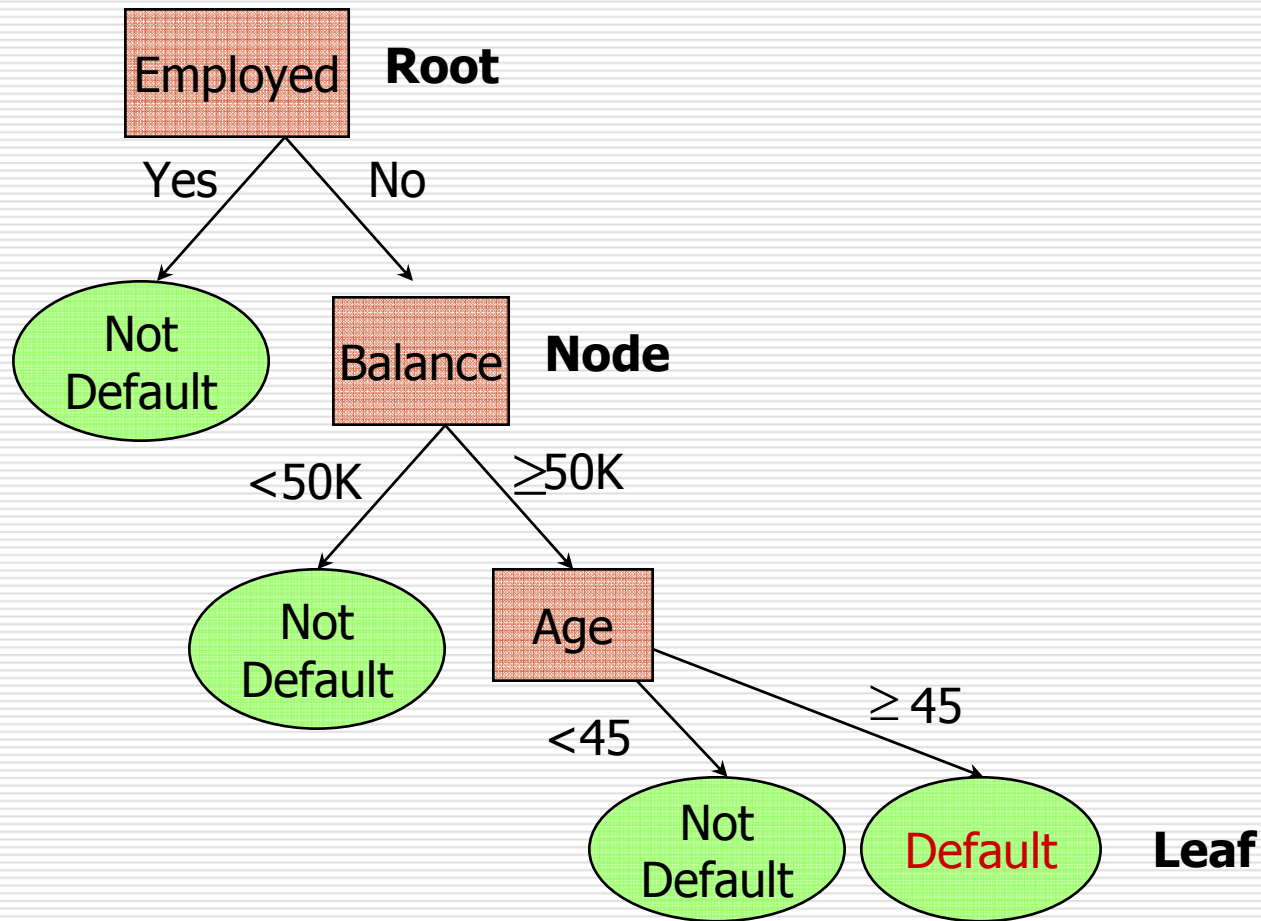
What is a decision tree?

- ❑ Predictive task
- ❑ Takes data on many examples to create filter for future examples
- ❑ Great for attribute based pairs on discrete splits
- ❑ Can be extended to real-valued attributes

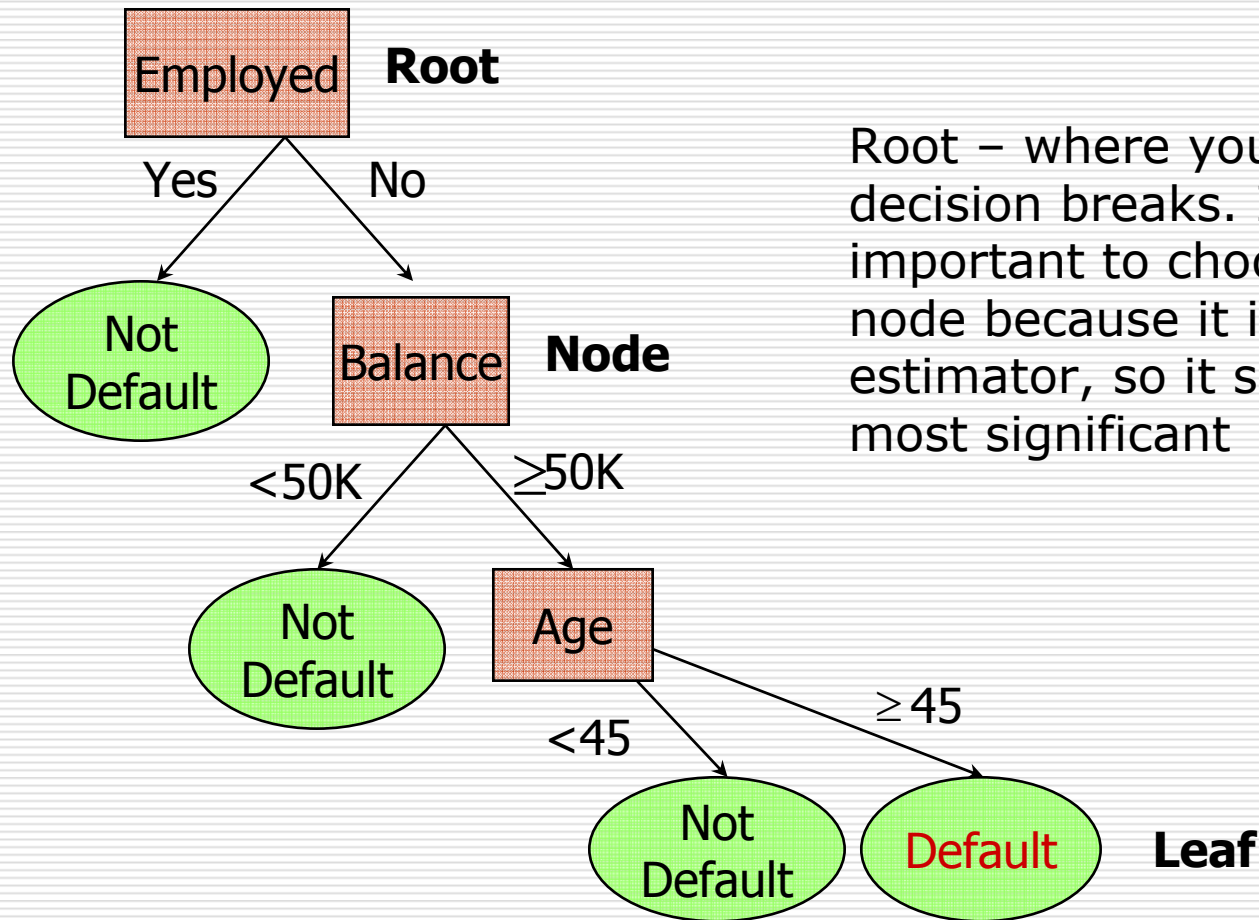
What is a decision tree?



What is a decision tree?

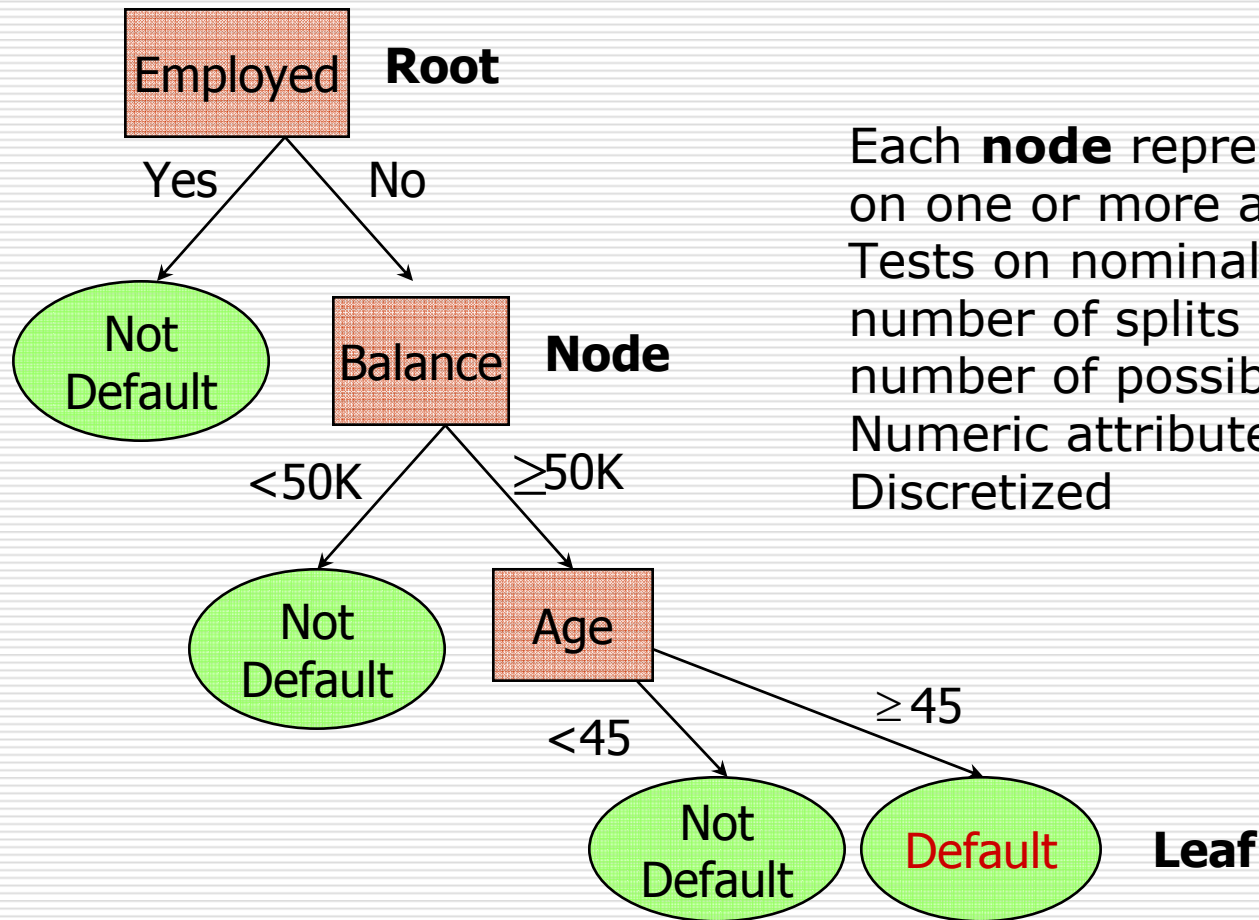


What is a decision tree?



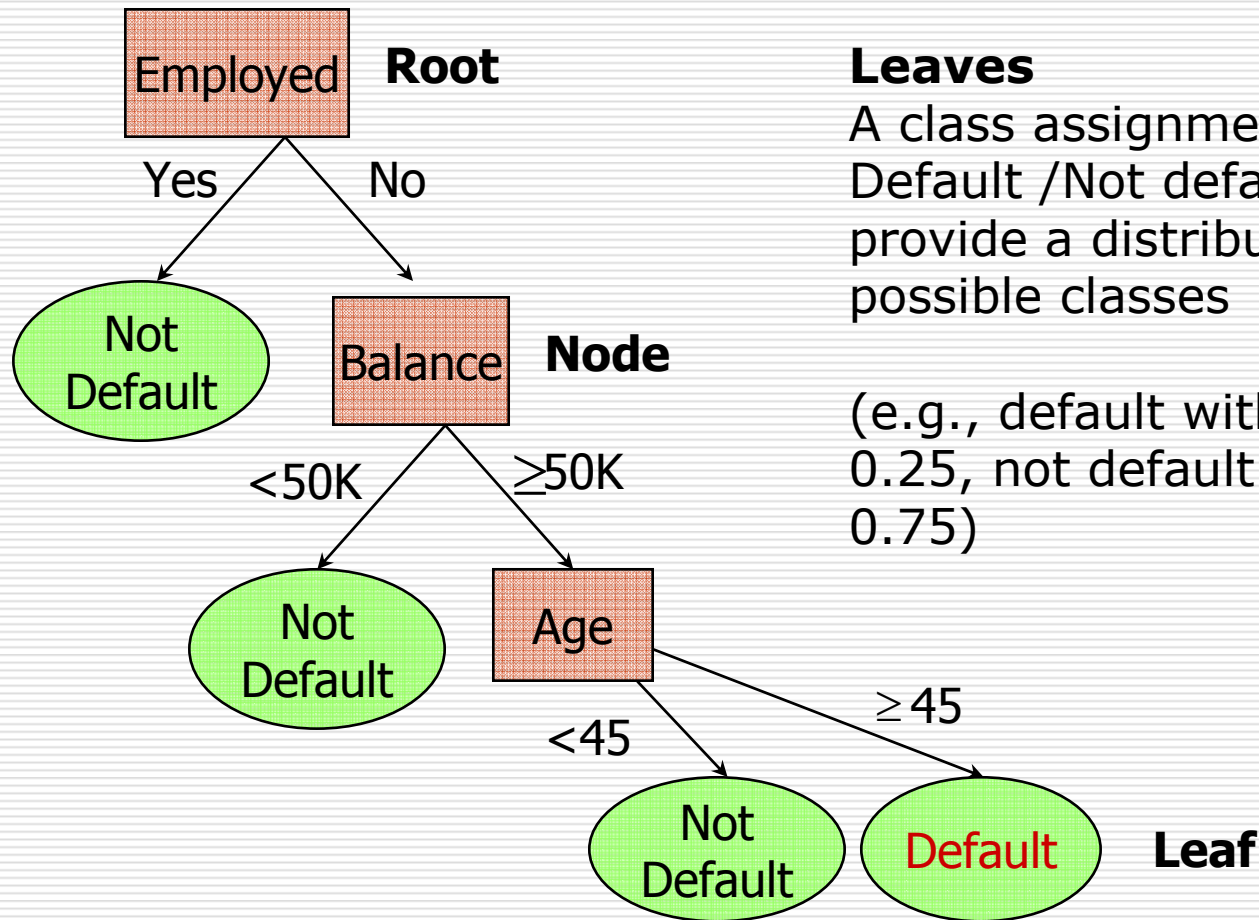
Root – where you start the decision breaks. It can be very important to choose the root node because it is the first estimator, so it should be the most significant

What is a decision tree?



Each **node** represents a test on one or more attributes
Tests on nominal attribute:
number of splits (branches) is
number of possible values
Numeric attributes are
Discretized

What is a decision tree?

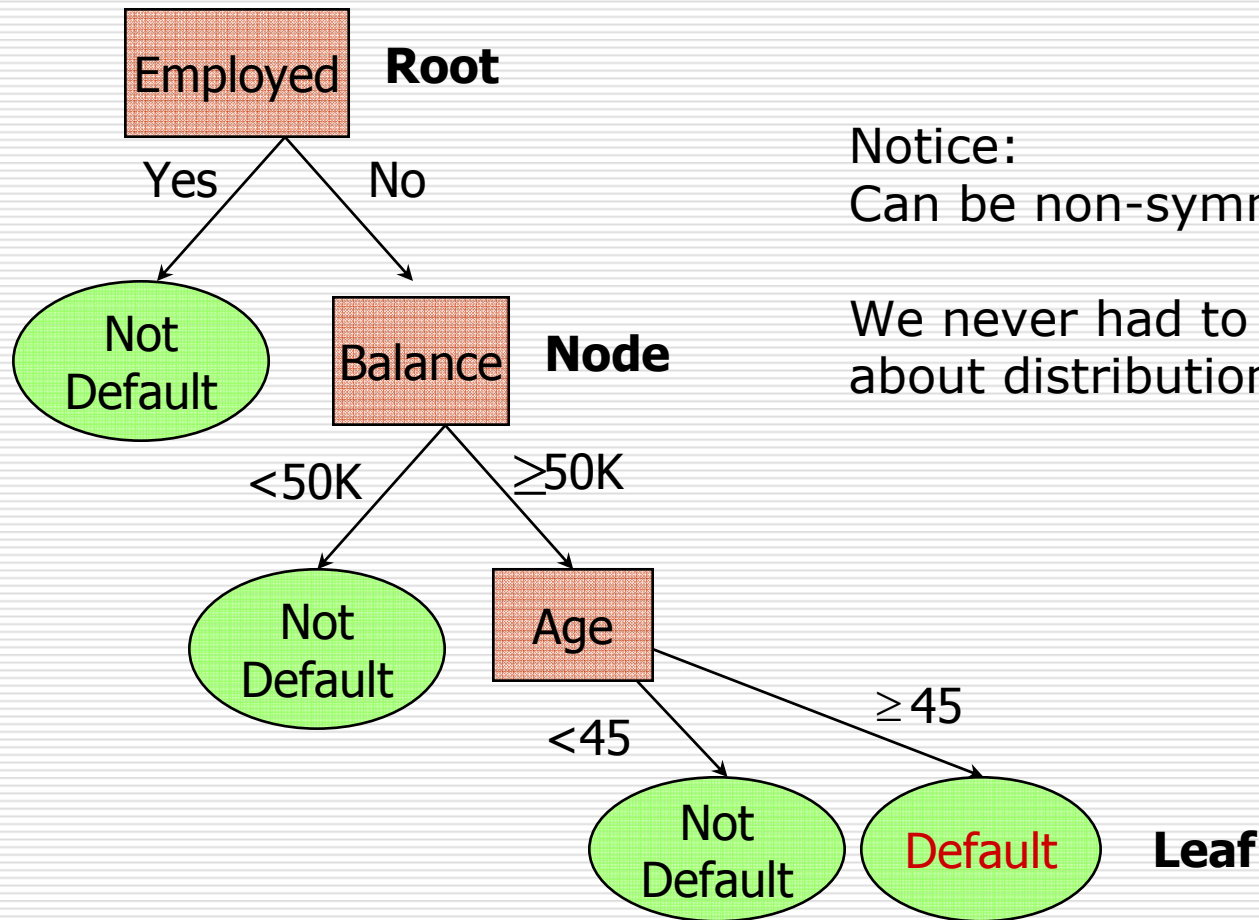


Leaves

A class assignment (E.g, Default /Not default)Also provide a distribution over all possible classes

(e.g., default with probability 0.25, not default with prob. 0.75)

What is a decision tree?



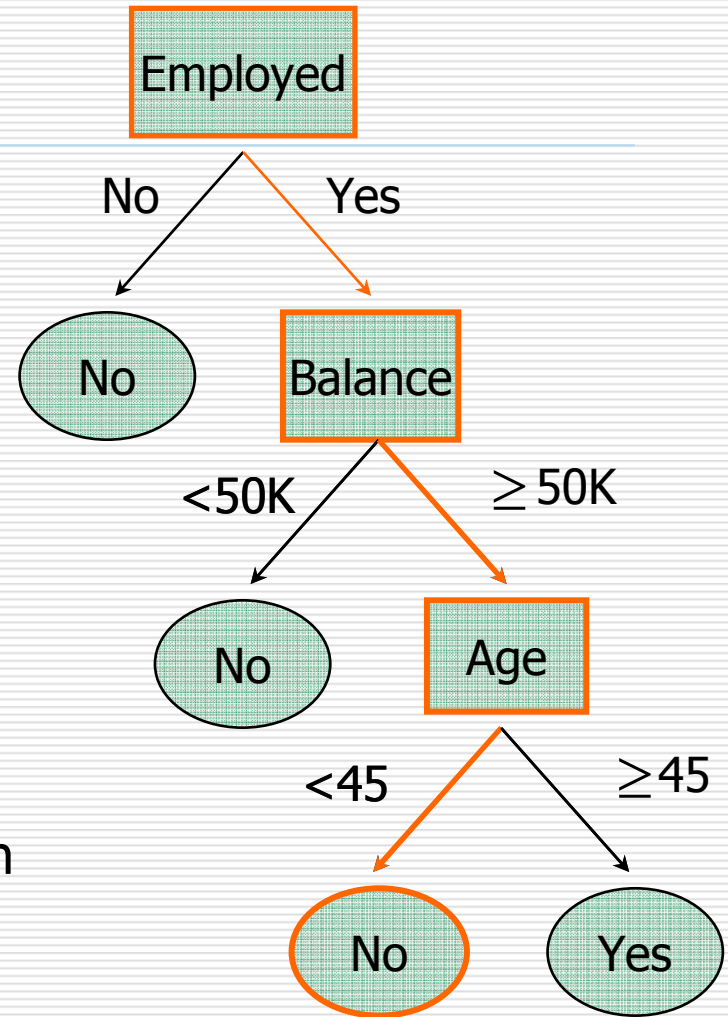
Notice:
Can be non-symmetrical

We never had to say anything
about distribution

Example

Mark, age 40, works for CitiBank, balance 88K.

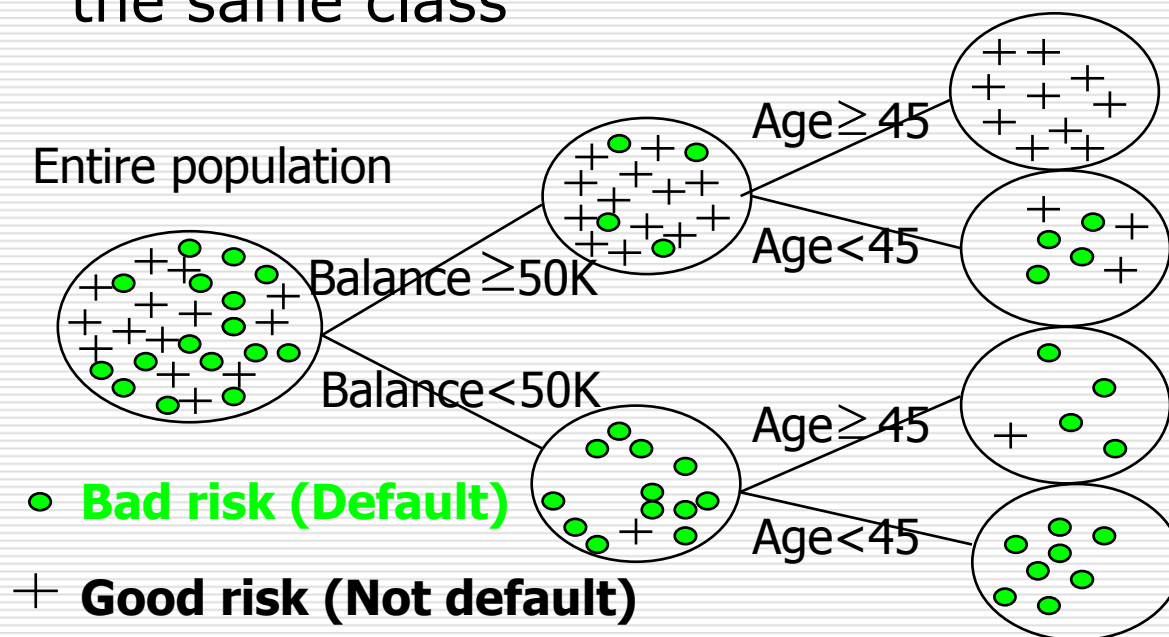
- The example is routed down the tree according to values of attributes tested successively.
- At each node a test is applied to one or more attributes.
- When a leaf is reached the example is assigned to a class, or alternatively to a distribution over the possible classes (e.g., default with probability 0.25, not default with prob. 0.75).



Decision Tree Induction

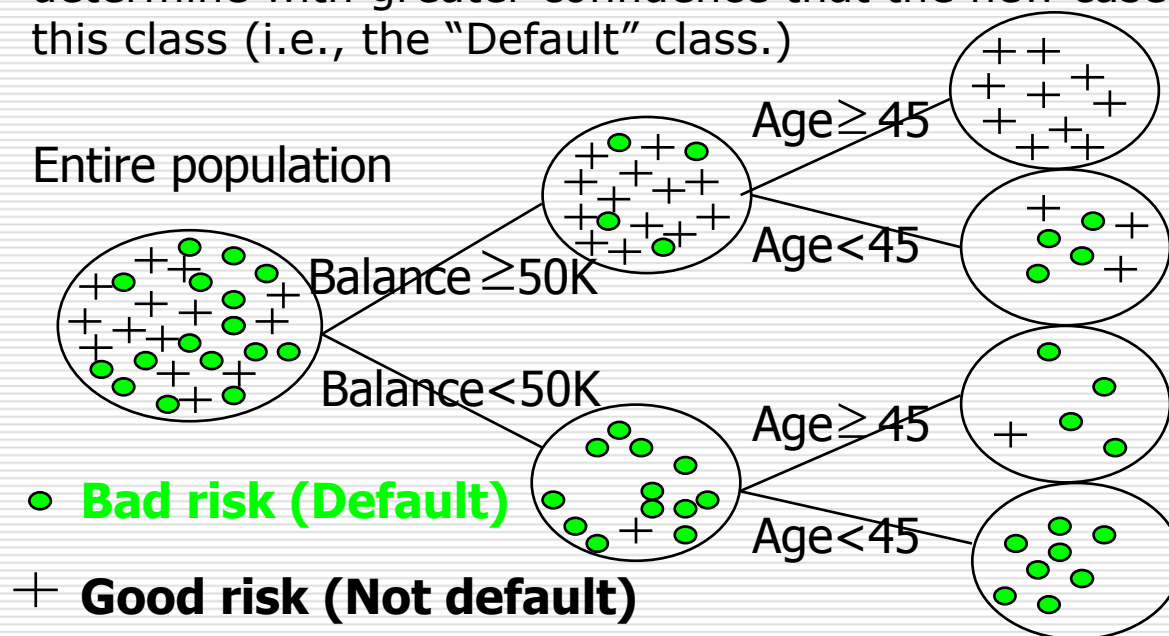
Partition the training examples into purer subgroups

Purity: groups of examples that (most) belong to the same class



Decision Tree Construction

- Why do we want to identify pure sub groups?
- To classify a new case, we can determine based on its attributes, leaf the case belongs to.
- If the respective leaf is very pure (say, all have defaulted) we can determine with greater confidence that the new case belongs to this class (i.e., the "Default" class.)

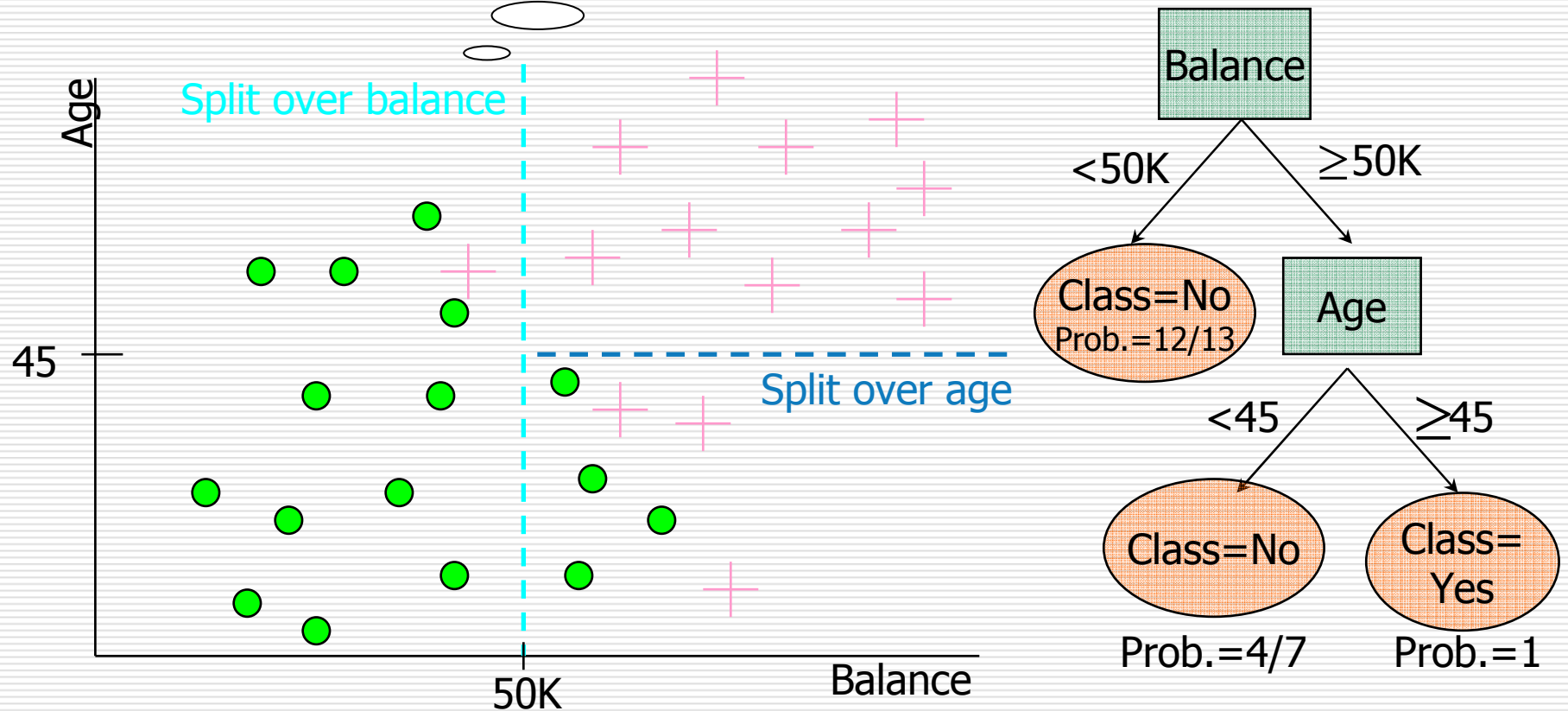


Decision Tree Construction

- A tree is constructed by recursively partitioning the examples.
- With each partition the examples are split into increasingly purer sub groups.

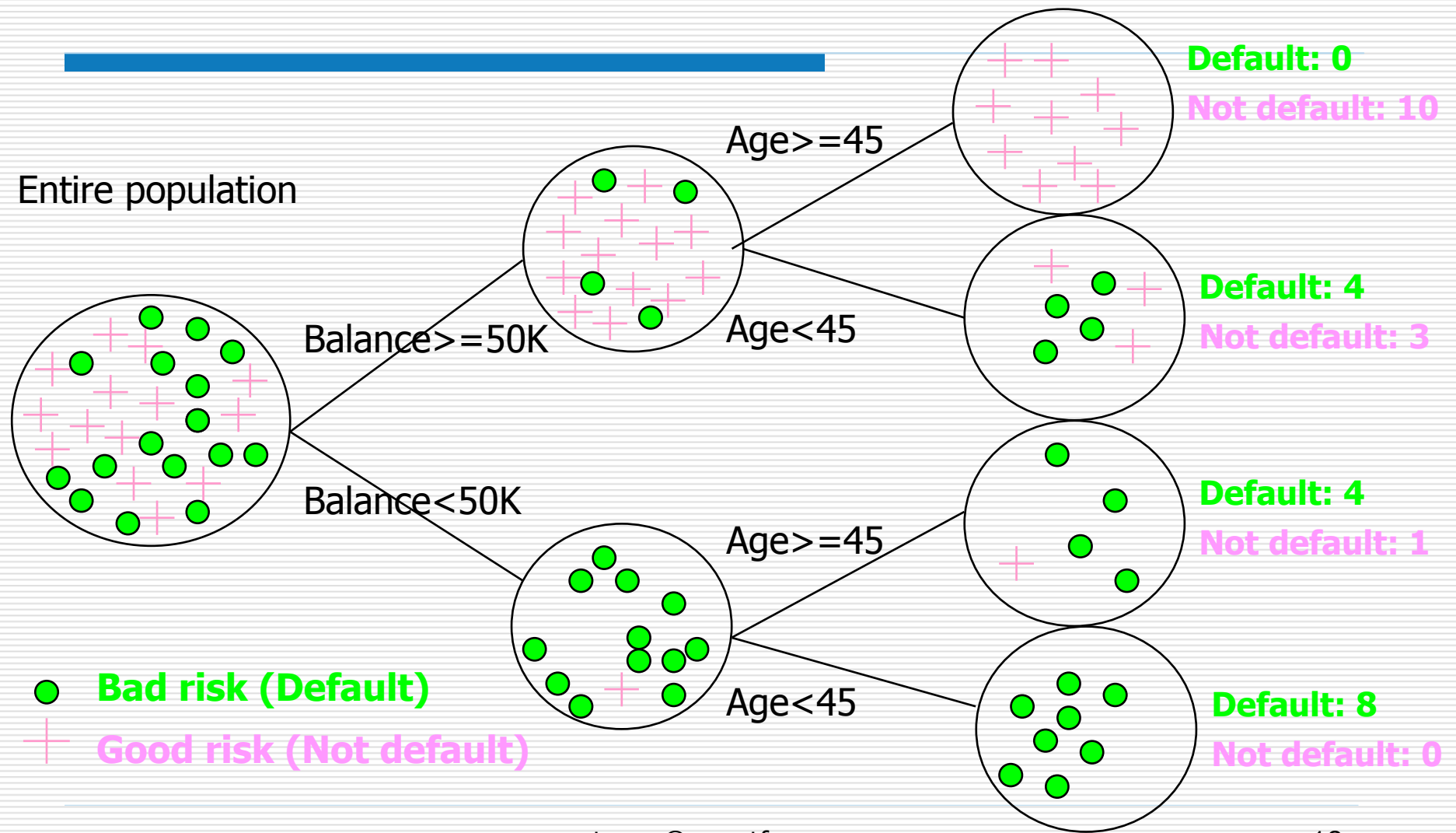
Split can be thought of as lines cutting the data

A Two-Class Problem (Default/Not Default)



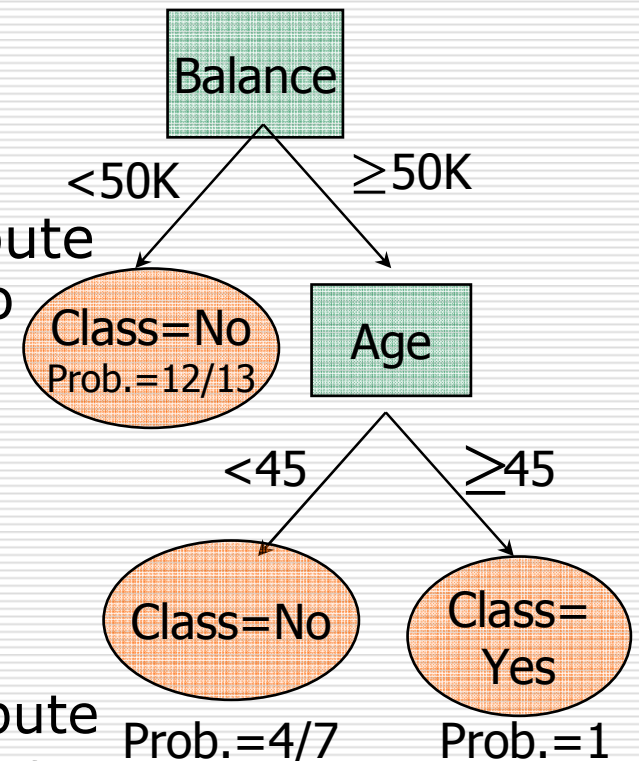
● **Bad risk (Default) – 16 cases** + **Good risk (Not default) – 14 cases**

Tree Representation of the Splits



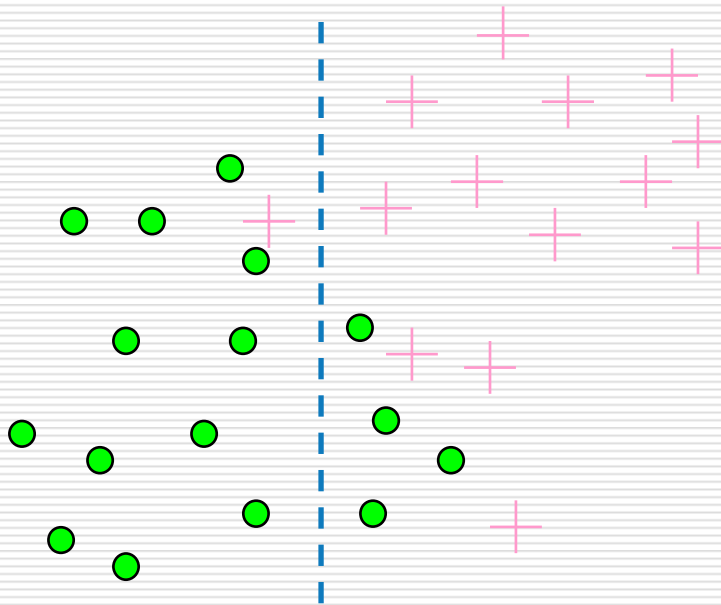
How to choose which attribute to split over?

- Objective: minimum number of splits (nodes) in the tree.
 - More accurate, more compact
- At each decision node choose the attribute that **best** partitions the population into **puer** groups.
- **Purity measures:** Many available
Most common one (from information theory) is: *Information Gain*
Informally: How informative is the attribute in distinguishing among instances (e.g., credit applicants) from different classes (Yes/No default)



Consider the two following splits (tests)
Which one is more informative?

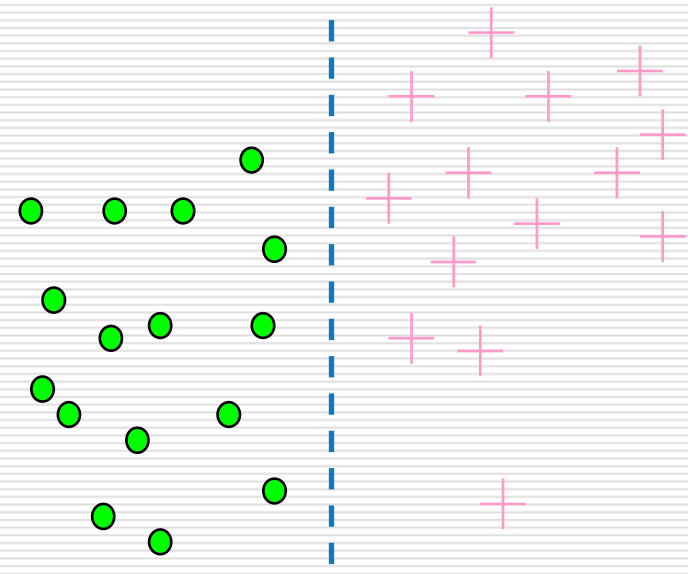
**Split over whether
Balance exceeds 50K**



Less or equal 50K

Over 50K

**Split over whether
applicant is employed**



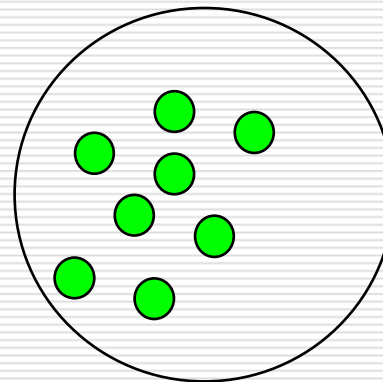
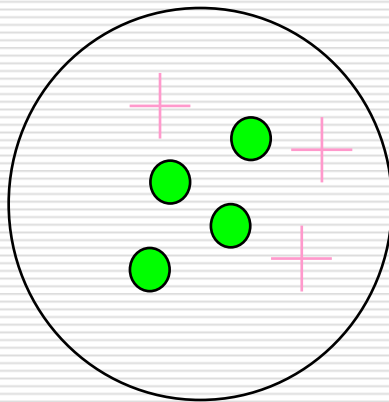
Unemployed

Employed

Information Gain

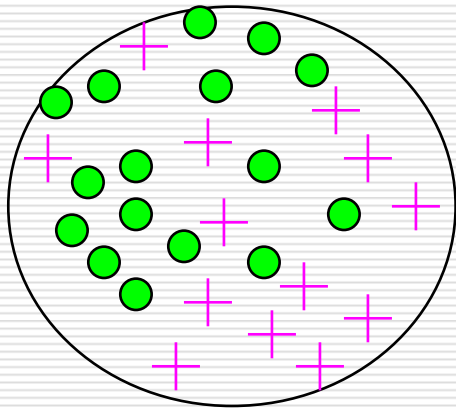
Impurity/Entropy (informal)

- Measures the level of **impurity** in a group of examples

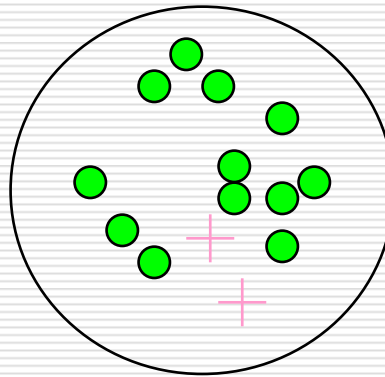


Impurity

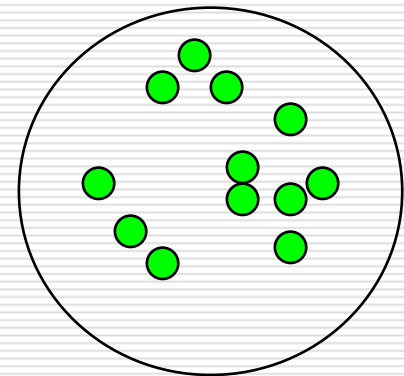
Very impure group



Less impure



Minimum impurity

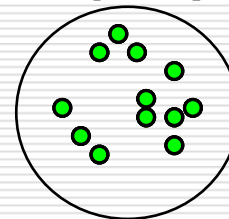


**When examples can belong to one of two classes:
What is the worst case of impurity?**

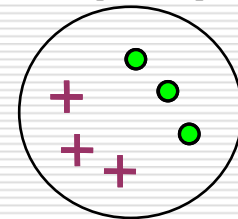
Calculating Impurity

- Assume there are n different output classes:
 - Each class i has a probability of p_i of appearing
 - Gini Index = $1 - \sum p_i^2$
 - The larger the value of GI, the larger the impurity
- If there are two classes
 - $GI = 1 - p_1^2 - p_2^2$
- Examples:
 - If $p_1 = 1, p_2 = 0$, $GI = 0$ (purest)
 - If $p_1 = p_2 = 0.5$, $GI = 0.5$ (least pure)
 - If $p_1 = 0.3, p_2 = 0.7$, $GI = 0.42$

**Minimum
impurity**



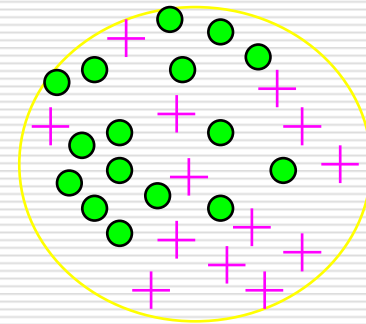
**Maximum
impurity**



Calculating Impurity

$$\text{Impurity} = \sum_i -p_i \log_2 p_i$$

p_i is proportion of class i

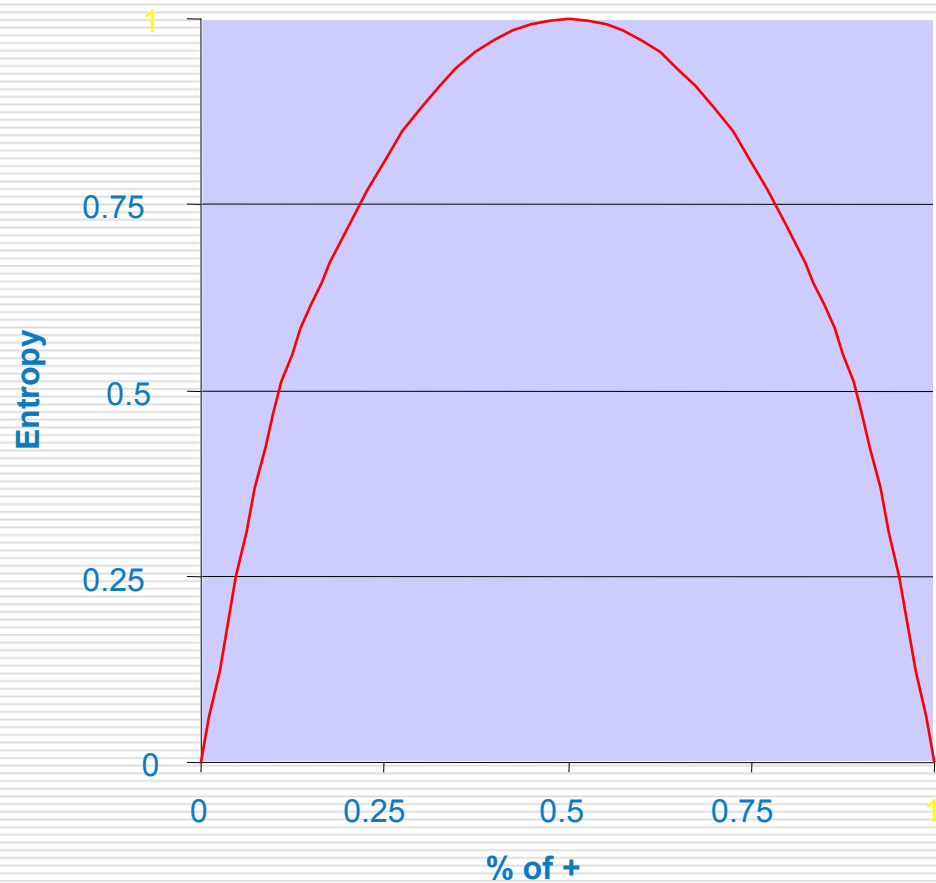


- For example: our initial population is composed of 14 cases of class "Default" and 16 cases of class "Not default"

Impurity (entire population of examples)=

$$-\left(\frac{14}{30} \cdot \log_2 \frac{14}{30}\right) - \left(\frac{16}{30} \cdot \log_2 \frac{16}{30}\right) = 0.996$$

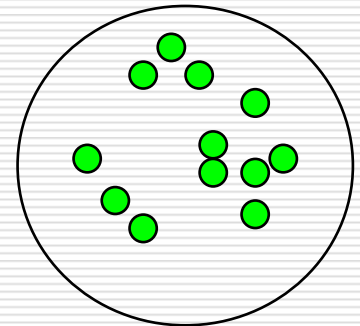
Entropy Function in 2 Class Case



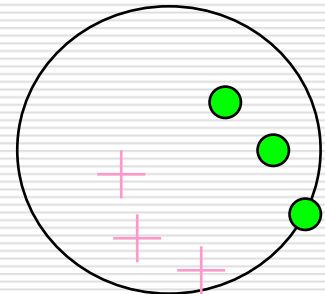
2-Class Cases:

- What is the impurity of a group in which all examples belong to the same class?
 - Impurity = $-1 \log_2 1 = 0$
- What is the impurity of a group with 50% in either class?
 - Impurity = $-0.5 \log_2 0.5 - 0.5 \log_2 0.5 = 1$

Minimum impurity



Maximum impurity



Information Gain: Reducing Entropy

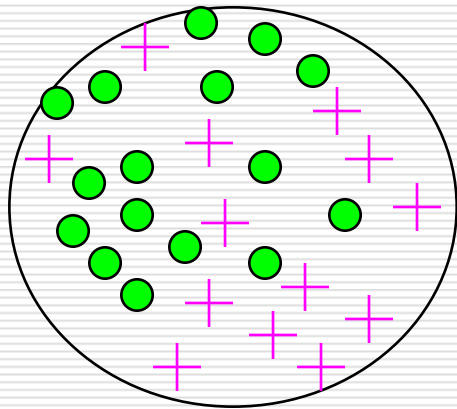
- Splitting a set into purer subsets reduces total entropy
- Calculation of the reduction in Entropy
 - Original E: straightforward
 - E of each subset: straightforward
 - How to combine the Es of the subsets?
Use a weighted sum:
 - $w_1 = \# \text{ of cases in subset 1} / \text{total \# of cases}$
 - $w_2 = \# \text{ of cases in subset 2} / \text{total \# of cases}$
 - $E_{\text{New}} = w_1 * E_1 + w_2 * E_2$

Calculating Information Gain

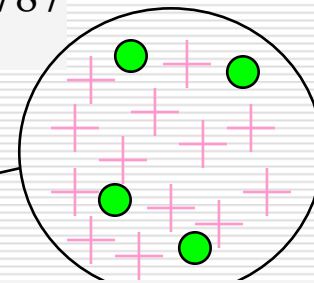
Information Gain = Impurity (parent) – [Impurity (children)]

$$\text{impurity} = -\left(\frac{13}{17} \cdot \log_2 \frac{13}{17}\right) - \left(\frac{4}{17} \cdot \log_2 \frac{4}{17}\right) = 0.787$$

Entire population (30 instances)



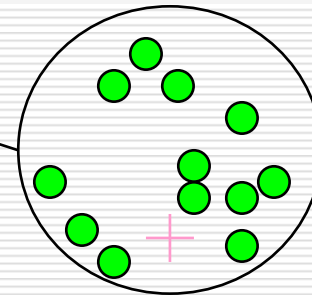
Balance $\geq 50K$



17 instances

$$\text{impurity} = -\left(\frac{1}{13} \cdot \log_2 \frac{1}{13}\right) - \left(\frac{12}{13} \cdot \log_2 \frac{12}{13}\right) = 0.391$$

Balance $< 50K$



13 instances

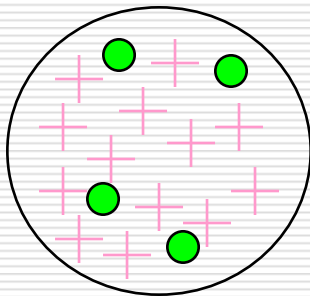
$$\text{impurity} = -\left(\frac{14}{30} \cdot \log_2 \frac{14}{30}\right) - \left(\frac{16}{30} \cdot \log_2 \frac{16}{30}\right) = 0.996$$

Calculating Information Gain

Information Gain = Impurity (parent) – [Impurity (children)]

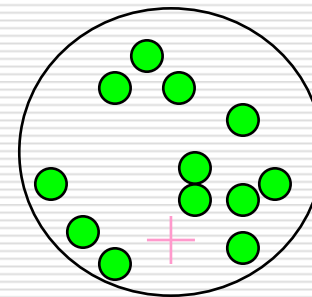
$$\text{impurity} = -\left(\frac{13}{17} \cdot \log_2 \frac{13}{17}\right) - \left(\frac{4}{17} \cdot \log_2 \frac{4}{17}\right) = 0.787$$

Balance \geq 50K



17 instances

Balance < 50K



13 instances

$$\text{impurity} = -\left(\frac{1}{13} \cdot \log_2 \frac{1}{13}\right) - \left(\frac{12}{13} \cdot \log_2 \frac{12}{13}\right) = 0.391$$

(Weighted) Average Impurity of **Children** =

$$\begin{aligned} \text{Information Gain} &= \text{Entropy (parent)} - \text{Entropy (Children)} \\ &= 0.996 - 0.615 = 0.38 \end{aligned}$$

Information Gain

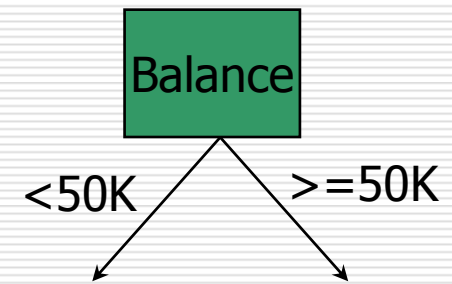
The Point!

At each node chose first the attribute that obtains maximum *information gain*:
providing maximum information

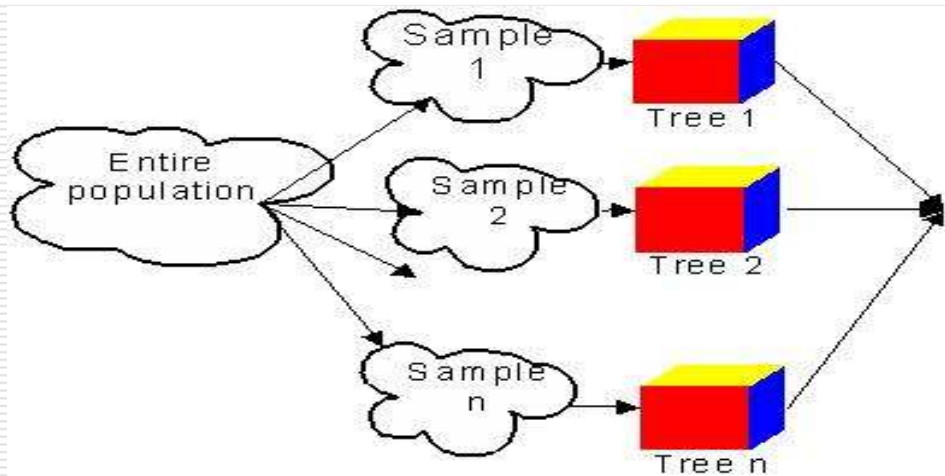
Which attribute to split over?

Brute-force search:

- At each node examine splits over each of the attributes
- Select the attribute for which the maximum information gain is obtained



Boosted Decision Trees



(tea) Bagged Decision Trees

$$\Pr .Default = \sum_i \Pr(Model_i)$$

- Draw N samples from the data set. Each sample contains a subset of the cases
- Build a tree from each sample
- For prediction: Use all trees to generate N predictions
 - The output is the average of all predictions

Standard Applications

- ❑ **Fraud:** Classifying cell calls into fraudulent and legitimate calls
- ❑ **Stock direction prediction:** determining whether a stock price will go up
- ❑ **Portfolio Selection:** Which stocks have the best attributes for the coming months

???

Member Benefits

- Members will enjoy:
 - seminars to discuss research with NYU professors like Brown, Figlewski, and Avellaneda
 - corporate mingles and corporate speaking events
 - members will receive job offers directly from corporate connections
 - members will work on research for major journals
 - youname@quantfs.com
 - CDs with video / audio lectures from Finance, Computer Science, and Mathematics

Target Membership

- You should be a member if:
 - you want a quant job
 - you want to go to graduate school
 - you respect knowledgeable people
 - you want to stay competitive in computerized world (eg. Goldman Trading)
 - you want to be better prepared for high level course work
 - you are from CAS in physics, chemistry, CS, economics, mathematics and are interested in finance
 - You want the CD

ApolloClap™

Get on up here

Midterms and Spring Break

- Computer Science workshop
- Linear Algebra Workshop
 - How to prepare
- Members are encourage to submit presentation topics
- Visit the website for goodies
- Get wild