Information Gain & Decision Trees

Slides adopted from
Data Mining for Business Analytics

Lecture 3: Supervised Segmentation

Stern School of Business
New York University
Spring 2014

Supervised Segmentation

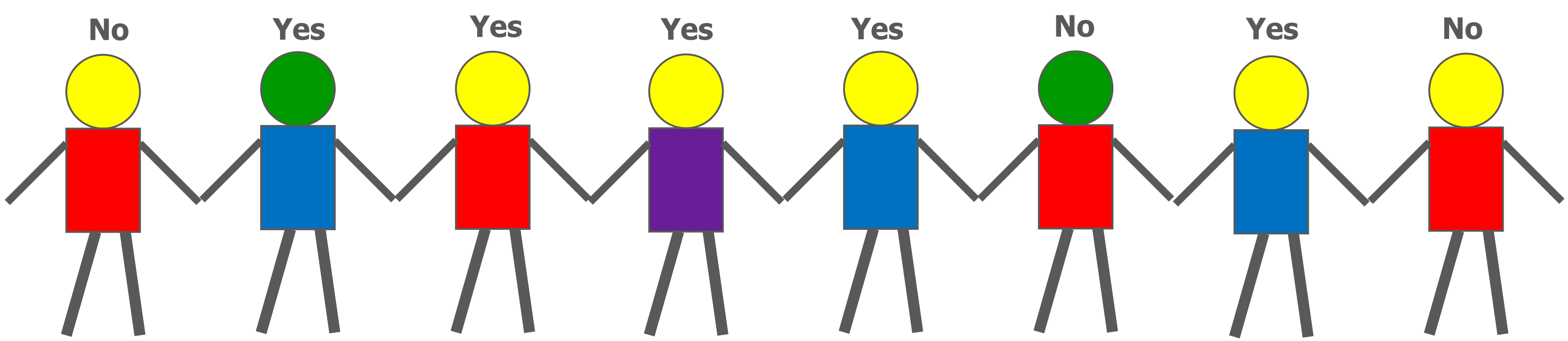
- How can we segment the population into groups that differ from each other with respect to some quantity of interest?
- Informative attributes
  - Find **knowable** attributes that correlate with the target of interest
    - Increase accuracy
    - Alleviate computational problems
    - E.g., *tree induction*

Supervised Segmentation

- How can we judge whether a variable contains important information about the target variable?
  - How much?

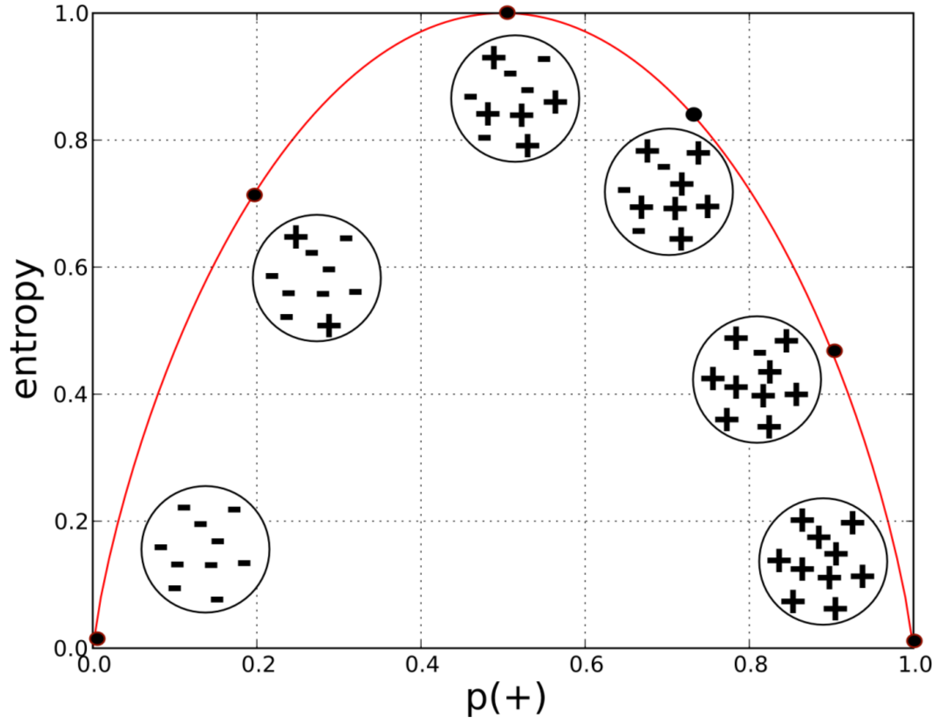
Selecting Informative Attributes

Objective: Based on customer attributes, partition the customers into subgroups that are less impure – with respect to the class (i.e., such that in each group as many instances as possible belong to the same class)



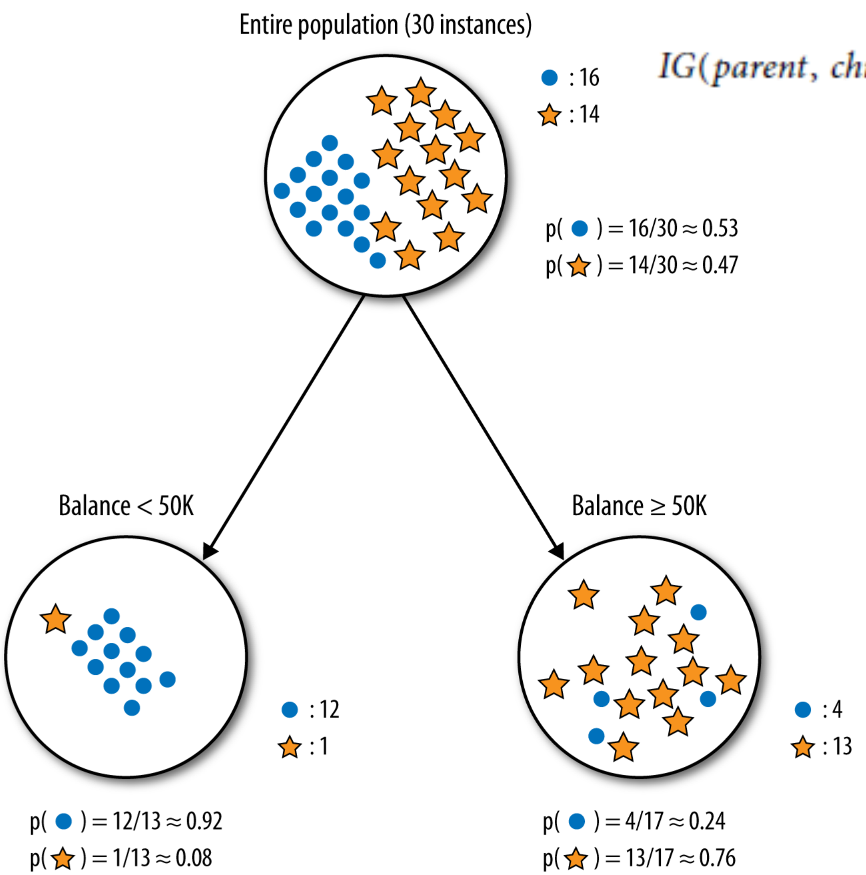
Selecting Informative Attributes

- The most common splitting criterion is called **information gain** (IG)
  - It is based on a **purity measure** called **entropy**
    - $entropy\ = -\mathrm{p}_1 \log_2(p_1) - p_2 \log_2(p_2) - ..$
    - Measures the general disorder of a set



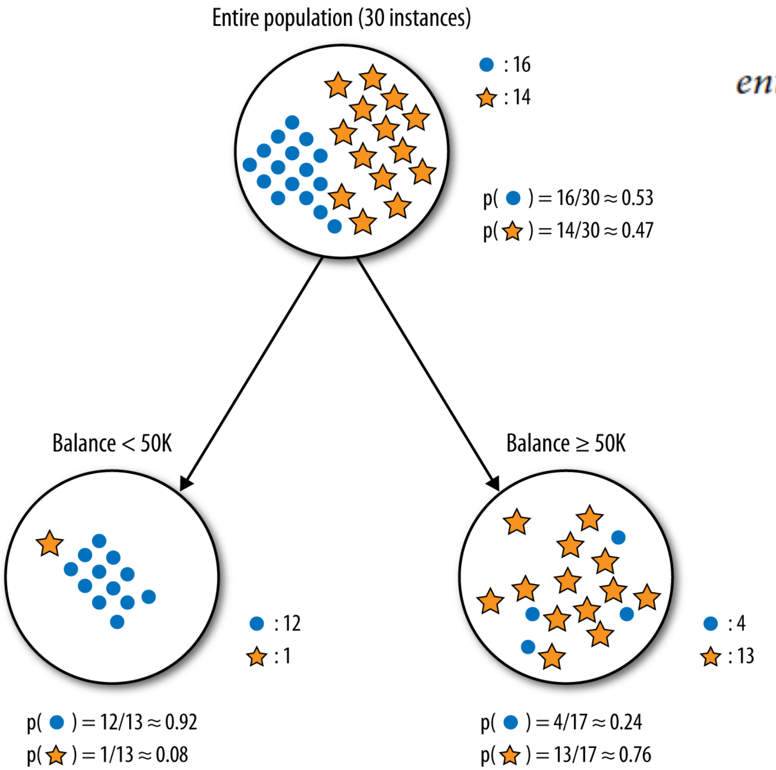
Information Gain

- Information gain measures the *change* in entropy due to any amount of new information being added



$$IG(parent, children) = entropy(parent) - [p(c_1) \times entropy(c_1) + p(c_2) \times entropy(c_2) + \cdots]$$

Information Gain



$$
\begin{aligned}
entropy(parent) &= -[p(\bullet) \times \log_2 p(\bullet) + p(\star) \times \log_2 p(\star)] \\
&\approx -[0.53 \times -0.9 + 0.47 \times -1.1] \\
&\approx 0.99 \quad \text{(very impure)}
\end{aligned}
$$

The entropy of the *left* child is:

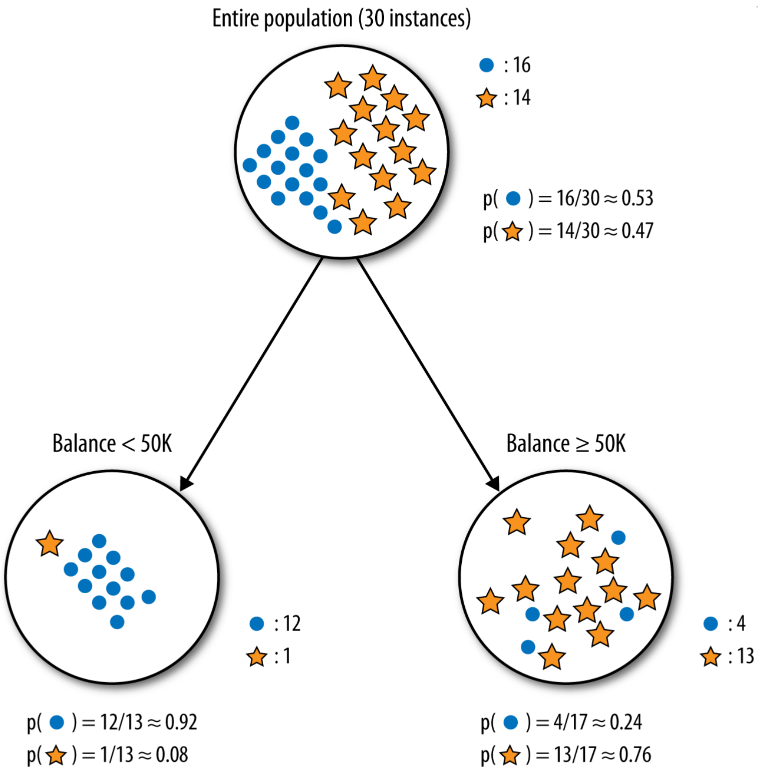
$$
\begin{aligned}
entropy(Balance < 50K) &= -[p(\bullet) \times \log_2 p(\bullet) + p(\star) \times \log_2 p(\star)] \\
&\approx -[0.92 \times (-0.12) + 0.08 \times (-3.7)] \\
&\approx 0.39
\end{aligned}
$$

The entropy of the *right* child is:

$$
\begin{aligned}
entropy(Balance \geq 50K) &= -[p(\bullet) \times \log_2 p(\bullet) + p(\star) \times \log_2 p(\star)] \\
&\approx -[0.24 \times (-2.1) + 0.76 \times (-0.39)]
\end{aligned}
$$

=0.79

Information Gain



$$
\begin{aligned}
IG &= \mathit{entropy}(\mathit{parent}) - [p(\text{Balance} < 50\text{K}) \times \mathit{entropy}(\text{Balance} < 50\text{K}) \\
&\qquad + p(\text{Balance} \geq 50\text{K}) \times \mathit{entropy}(\text{Balance} \geq 50\text{K})] \\
&\approx 0.99 - [0.43 \times 0.39 + 0.57 \times 0.79] \\
&\approx 0.37
\end{aligned}
$$

Relative IG = IG/entropy(parent)=0.37/0.99=0.37

Attribute Selection

Reasons for selecting only a subset of attributes:

- Better insights and business understanding
- Better explanations and more tractable models
- Reduced cost
- Faster predictions
- Better predictions!
  - Over-fitting (*to be continued*..)

and also determining the most informative attributes.

Example: Attribution Selection with Information Gain

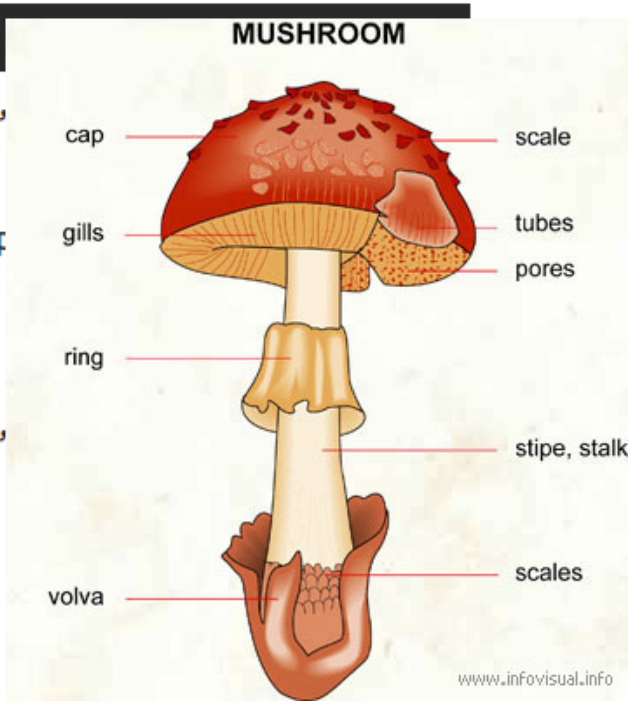
- This dataset includes descriptions of hypothetical samples corresponding to 23 species of gilled mushrooms in the Agaricus and Lepiota Family

- Each species is identified as definitely edible, definitely poisonous, or of unknown edibility and not recommended
  - This latter class was combined with the poisonous one

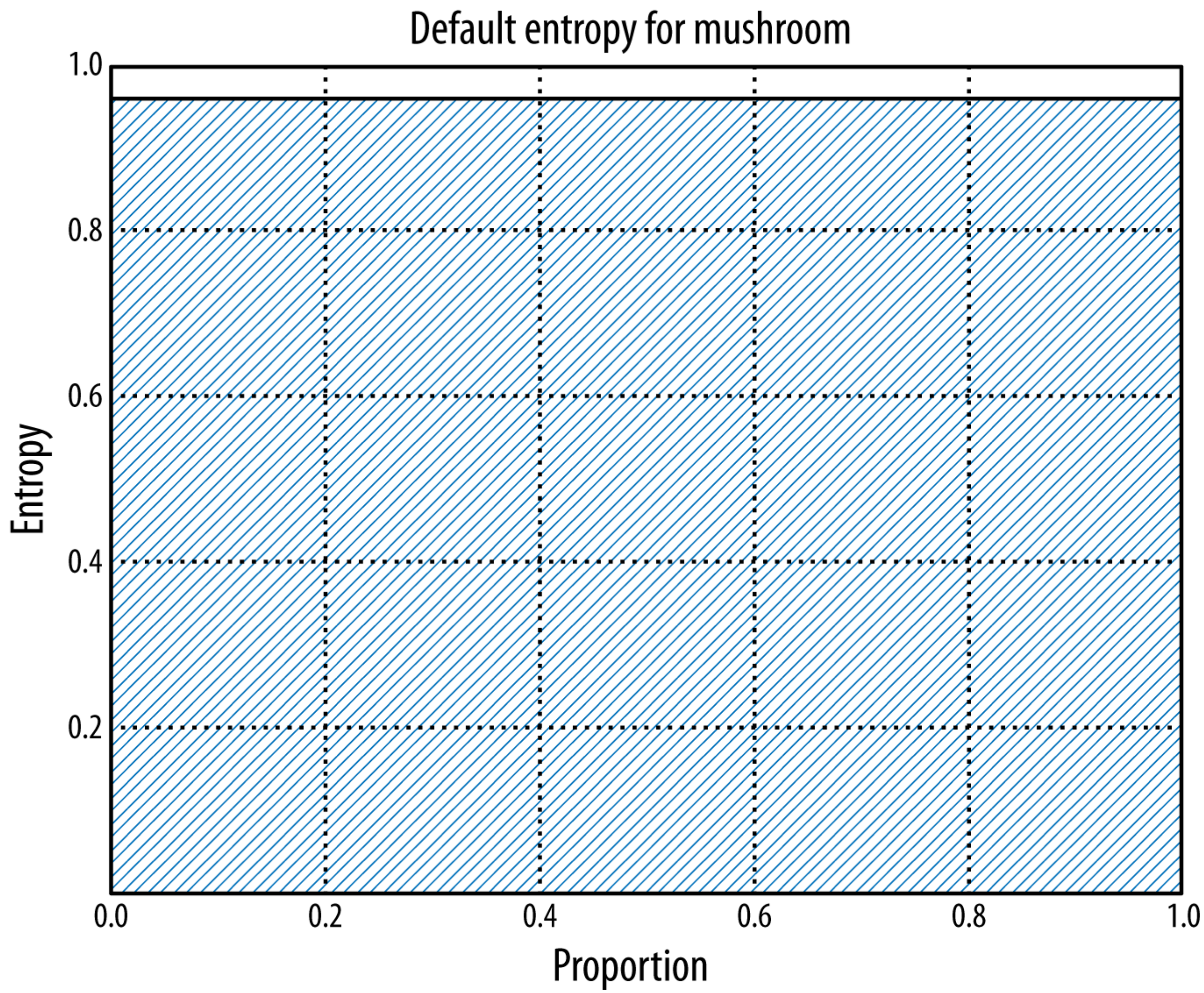
- The Guide clearly states that there is no simple rule for determining the edibility of a mushroom; no rule like “leaflets three, let it be” for Poisonous Oak and Ivy

Example: Attribution Selection with Information Gain

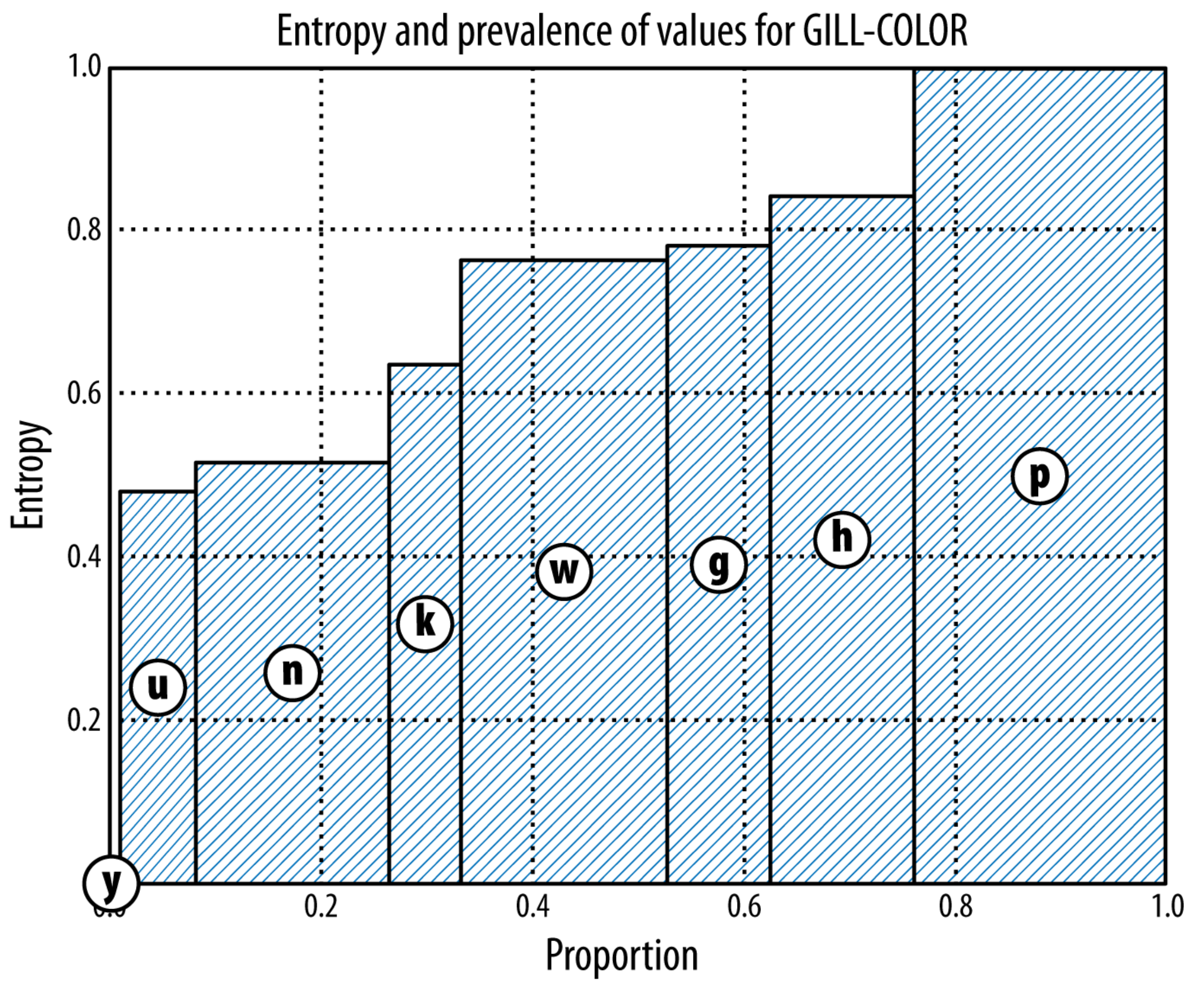
Attribute name	Possible values
CAP-SHAPE	bell, conical, convex, flat, knobbed,
CAP-SURFACE	fibrous, grooves, scaly, smooth
CAP-COLOR	brown, buff, cinnamon, gray, green, p white, yellow
BRUISES?	yes, no
ODOR	almond, anise, creosote, fishy, foul, pungent, spicy
GILL-ATTACHMENT	attached, descending, free, notched
GILL-SPACING	close, crowded, distant
GILL-SIZE	broad, narrow
GILL-COLOR	black, brown, buff, chocolate, gray, green, orange, pink, purple, red, white, yellow
STALK-SHAPE	enlarging, tapering
STALK-ROOT	bulbous, club, cup, equal, rhizomorphs, rooted, missing
STALK-SURFACE-ABOVE-RING	fibrous, scaly, silky, smooth
STALK-SURFACE-BELOW-RING	fibrous, scaly, silky, smooth



Example: Attribution Selection with Information Gain

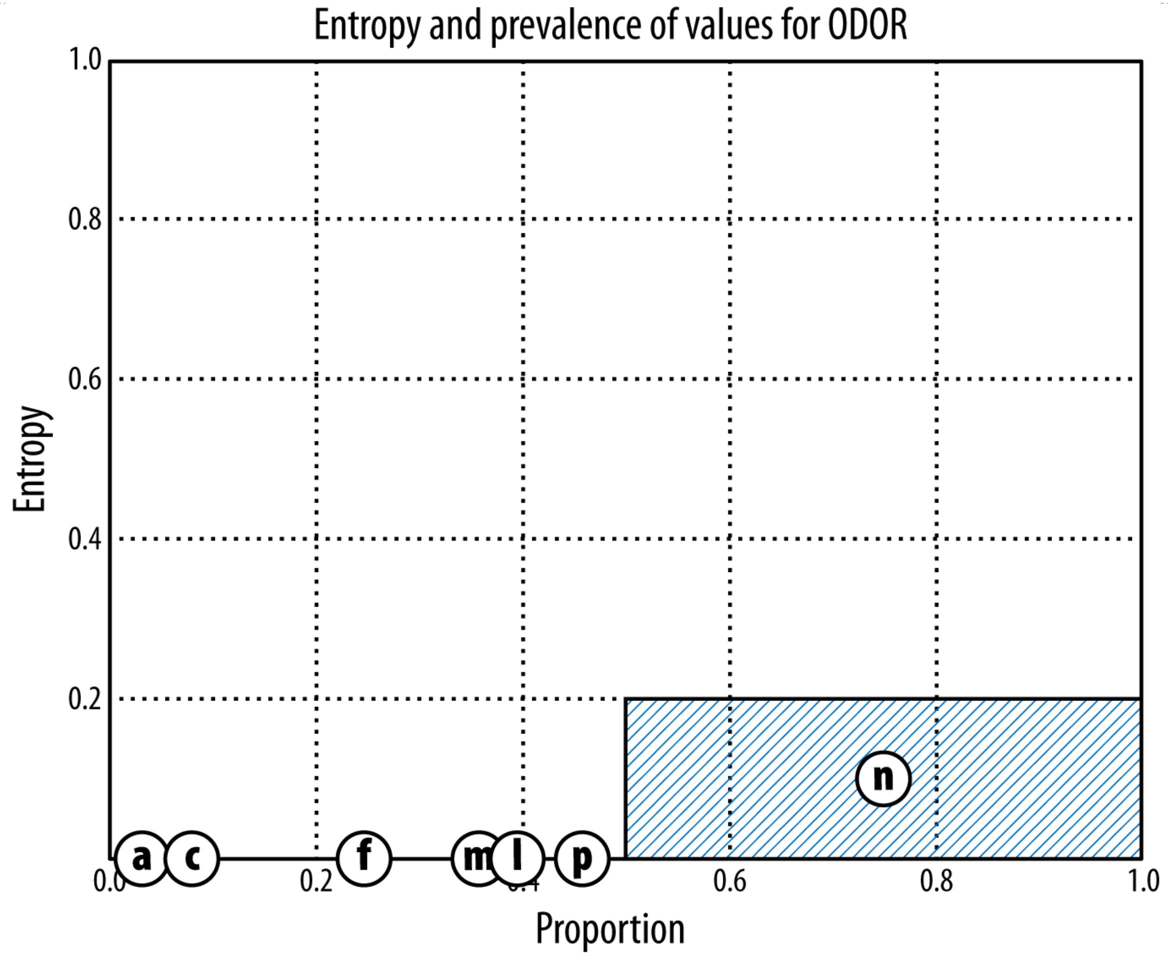


Example: Attribution Selection with Information Gain



GILL-COLOR `black, brown, buff, chocolate, gray, green, orange, pink, purple, red, white, yellow`

Example: Attribution Selection with Information Gain

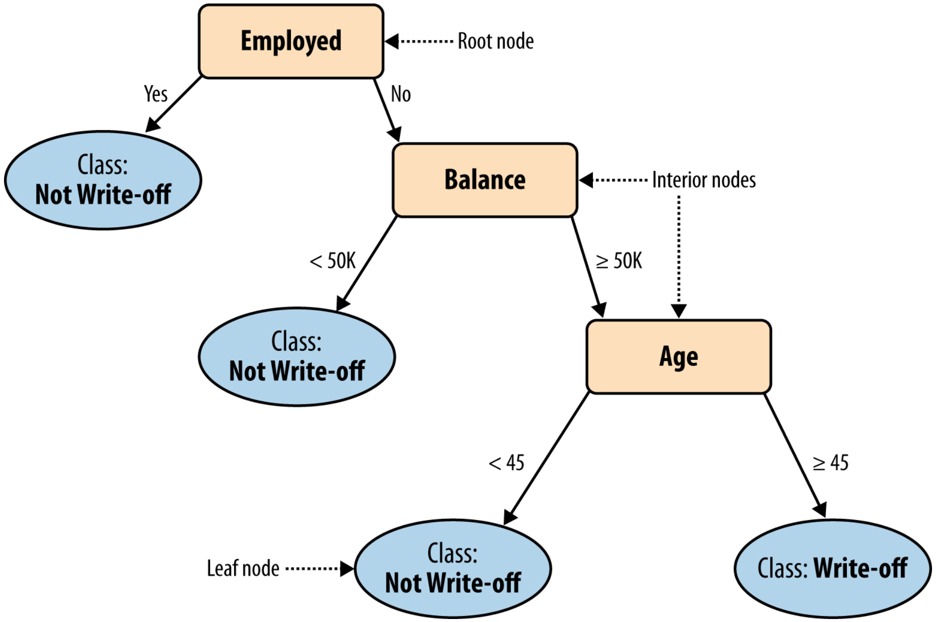


ODOR almond, anise, creosote, fishy, foul, musty, none, pungent, spicy

Multivariate Supervised Segmentation

- If we select the *single* variable that gives the most information gain, we create a very *simple* segmentation
- If we select multiple attributes each giving some information gain, how do we put them together?

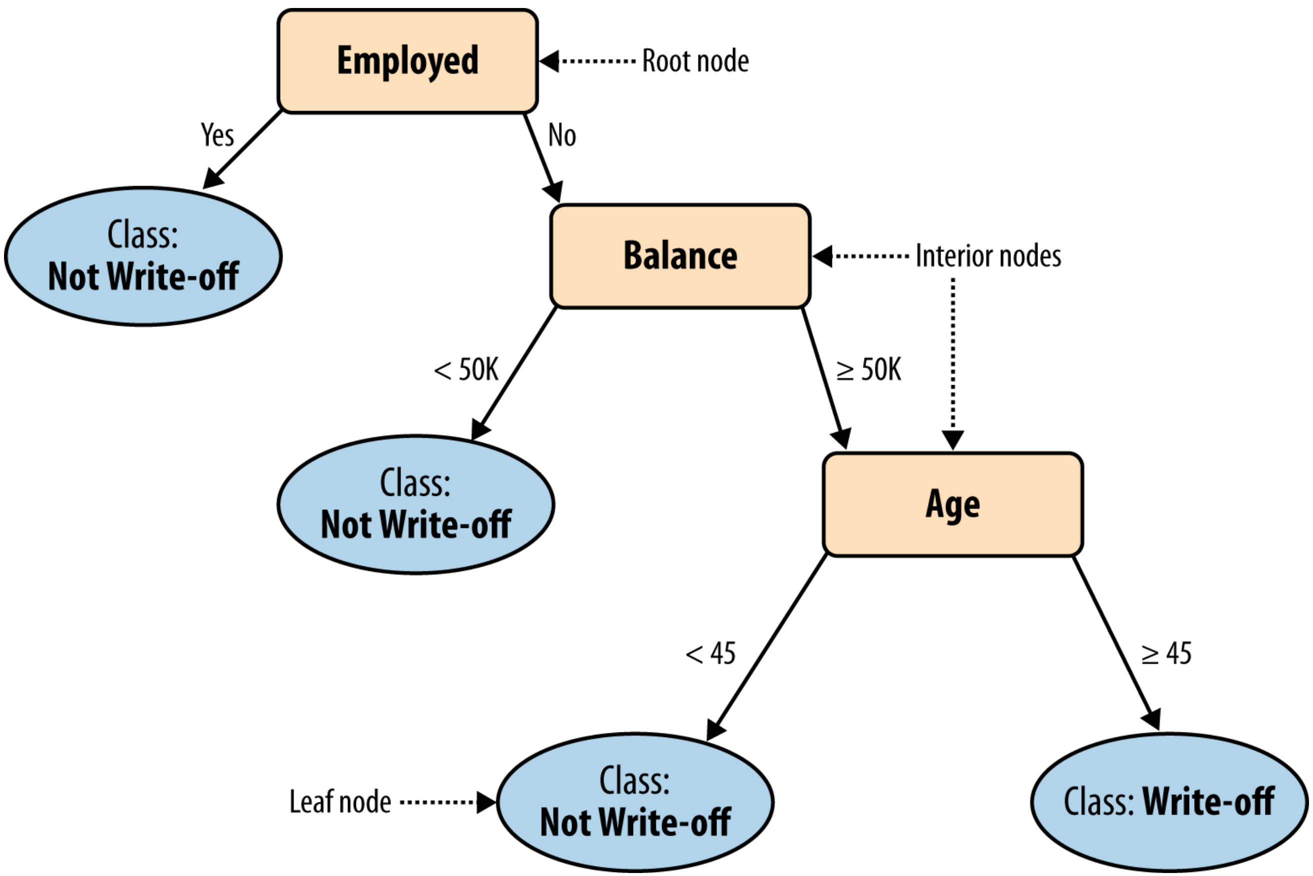
Tree-Structured Models



Write-off: not to pay off their account balances. i.e., defaulting on one’s phone bill or credit card balance

Tree-Structured Models

- Classify ‘John Doe’
  - Balance=115K, Employed=No, and Age=40



Tree-Structured Models: “Rules”

- No two parents share descendants
- There are no cycles
- The branches always “point downwards”
- Every example always ends up at a leaf node with some specific class determination
  - Probability estimation trees, regression trees (*to be continued*..)

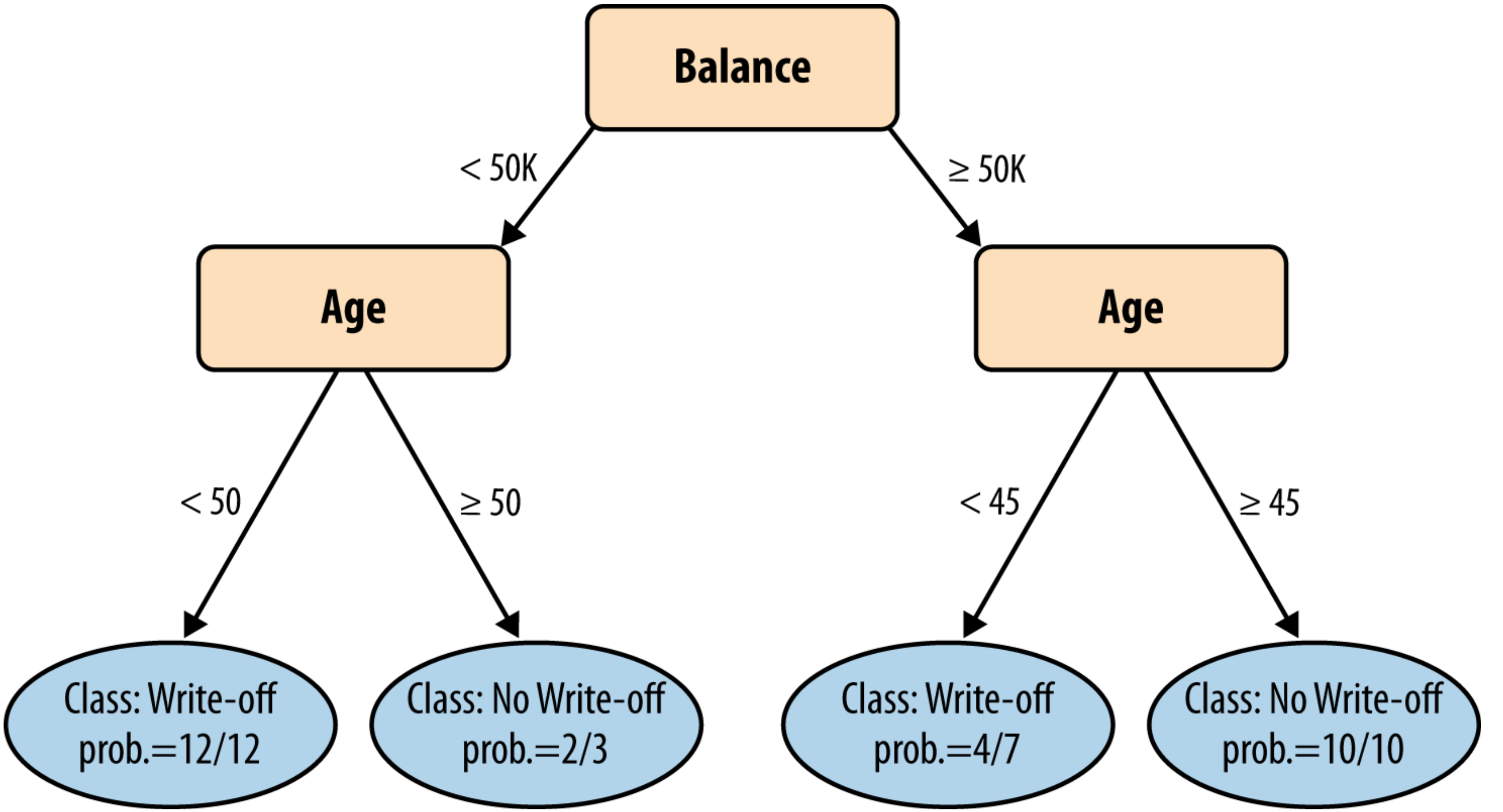
Tree Induction

- How do we create a classification tree from data?
  - **divide-and-conquer** approach
  - take each data subset and ***recursively*** apply attribute selection to find the best attribute to partition it
- When do we stop?
  - The nodes are pure,
  - there are no more variables, or
  - even earlier (over-fitting – *to be continued*..)

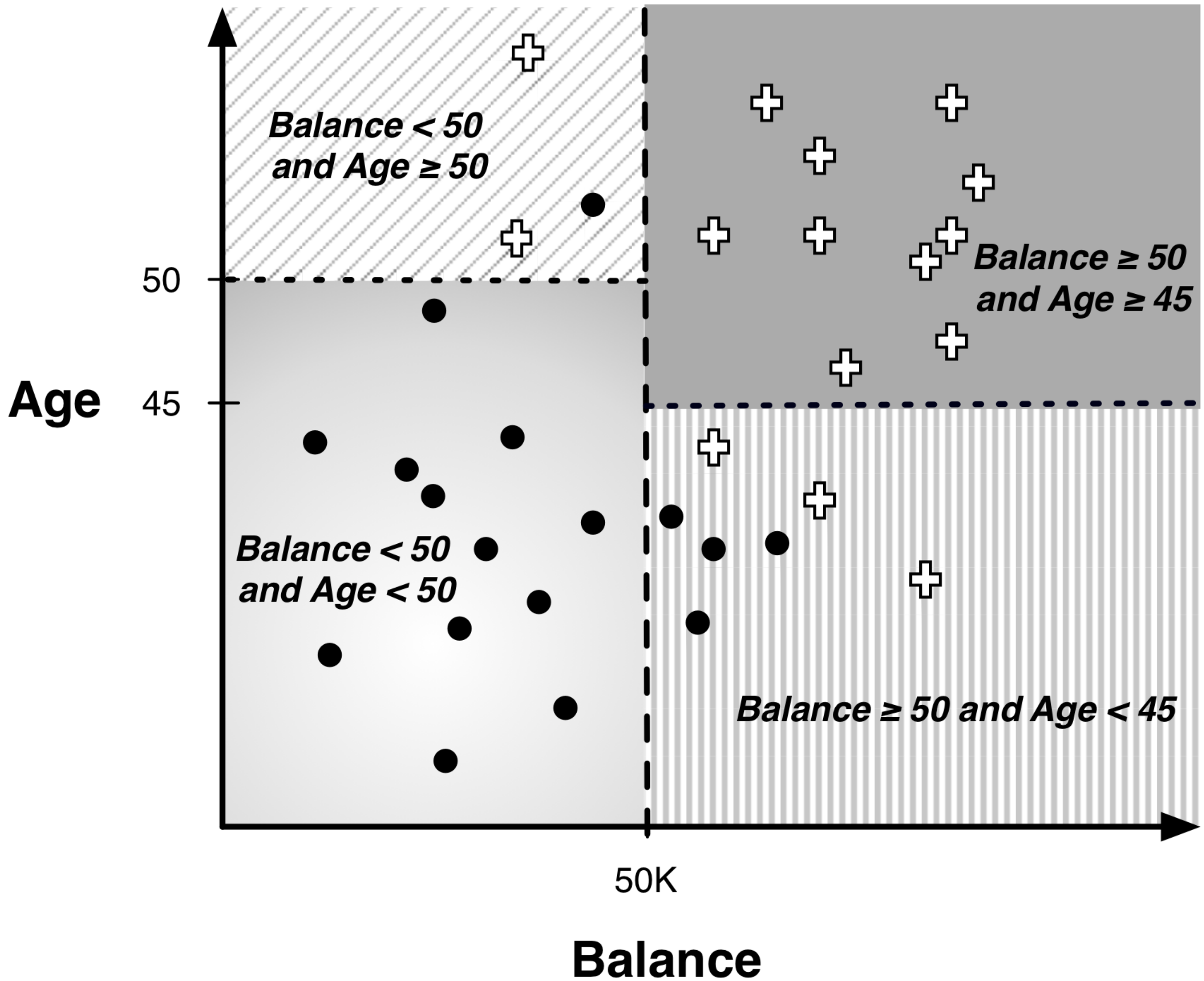
Why trees?

- Decision trees (DTs), or classification trees, are one of the most popular data mining tools
  - (along with linear and logistic regression)
- They are:
  - Easy to understand
  - Easy to implement
  - Easy to use
  - Computationally cheap
- Almost all data mining packages include DTs
- They have advantages for model comprehensibility, which is important for:
  - model evaluation
  - communication to non-DM-savvy stakeholders

Visualizing Segmentations



Visualizing Segmentations



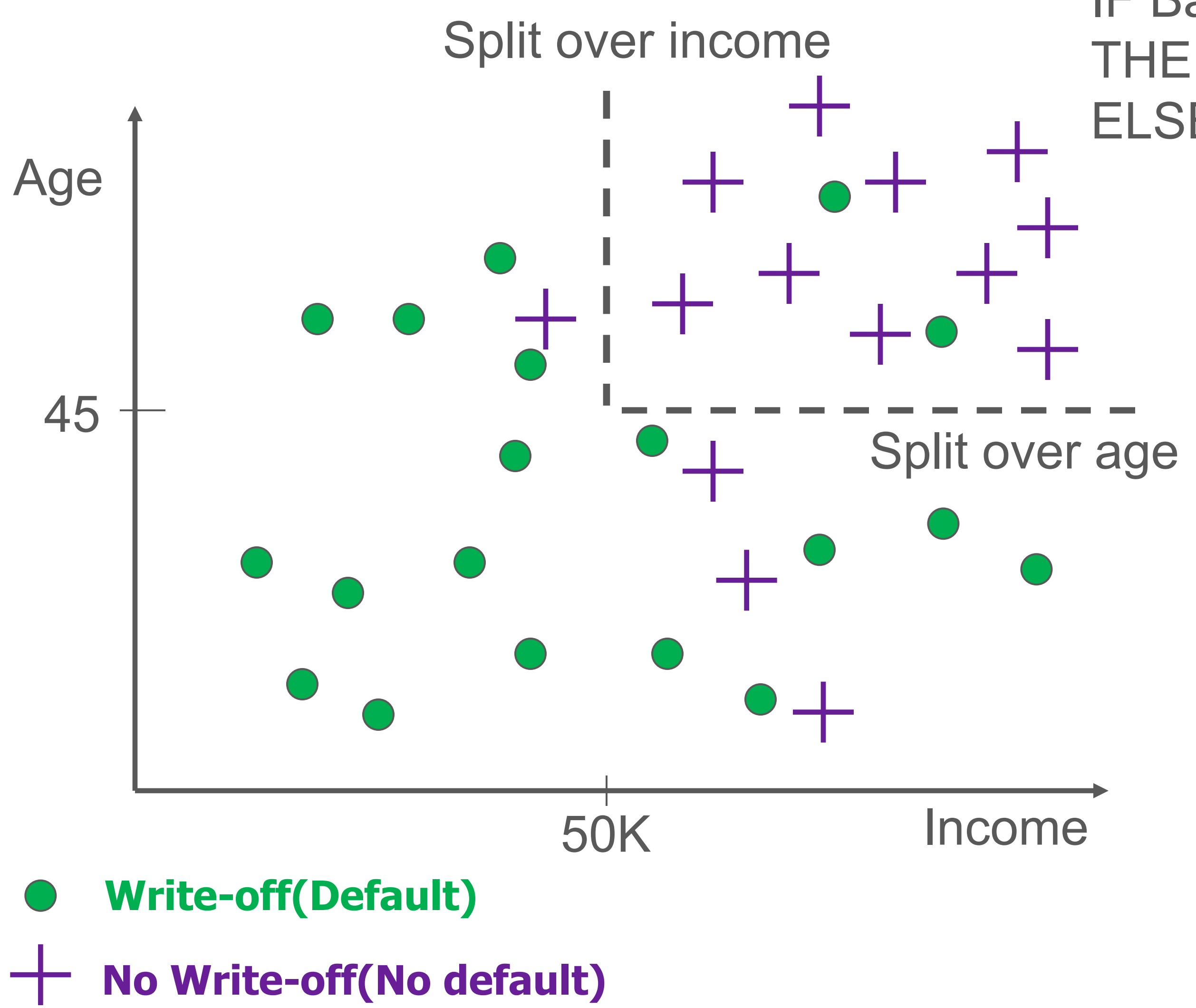
Geometric interpretation of a model

Pattern:

IF Balance >= 50K & Age > 45

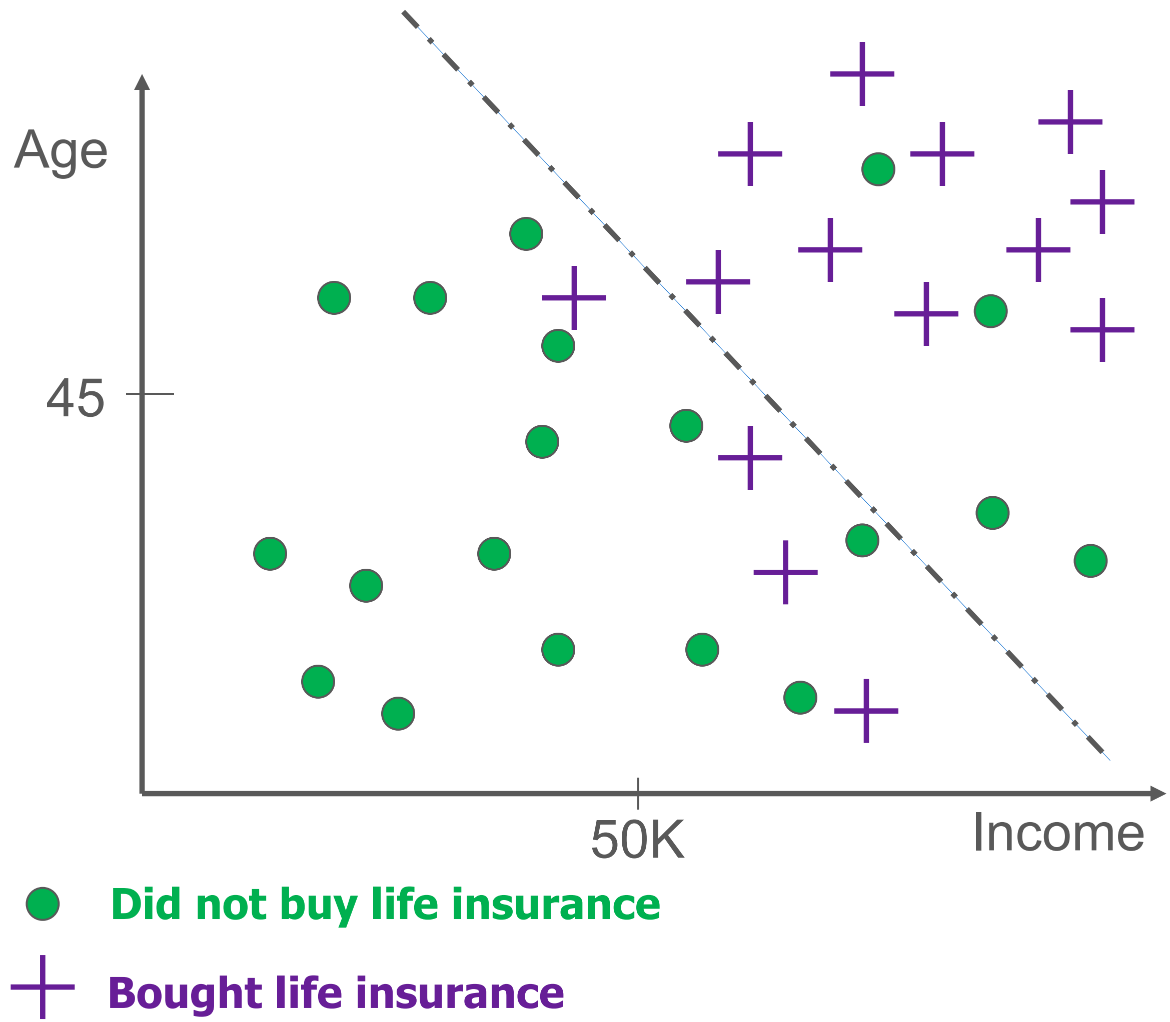
THEN Write-off = 'no'

ELSE Write-off = 'yes'



Geometric interpretation of a model

What alternatives are there to partitioning this way?

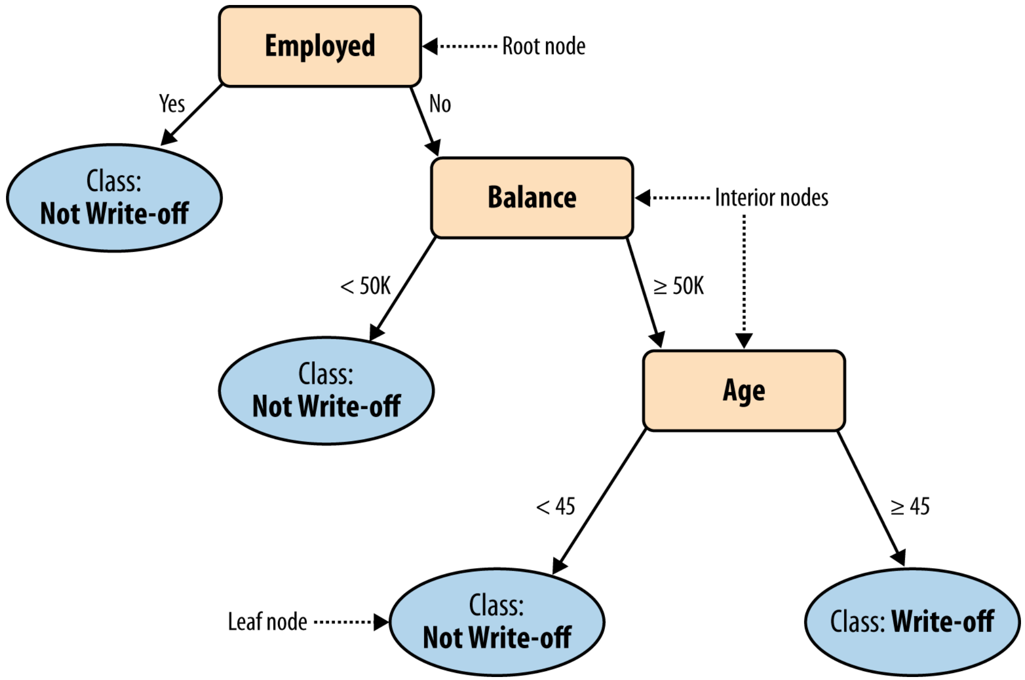


"True" boundary may not be closely approximated by a linear boundary!

Trees as Sets of Rules

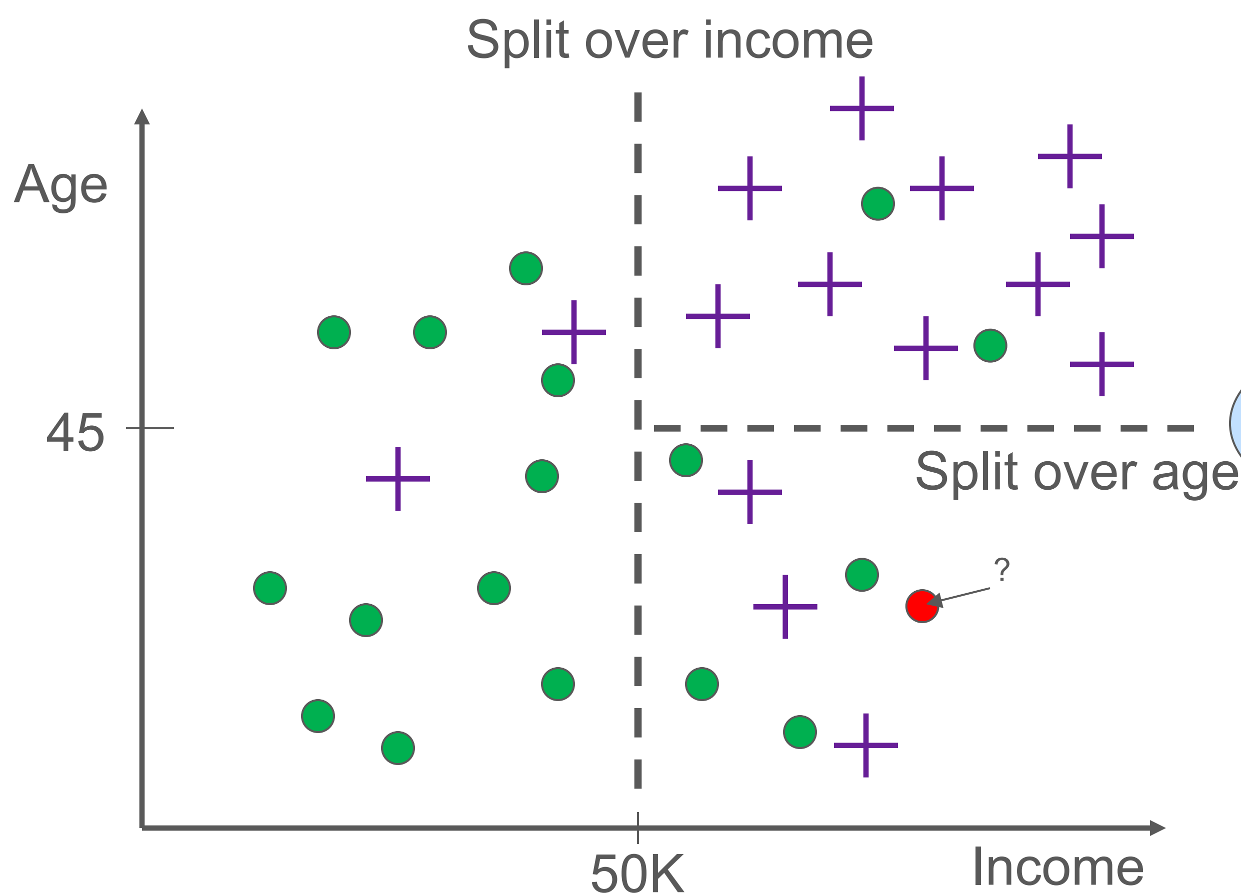
- The classification tree is equivalent to this rule set
- Each rule consists of the attribute tests along the path connected with **AND**

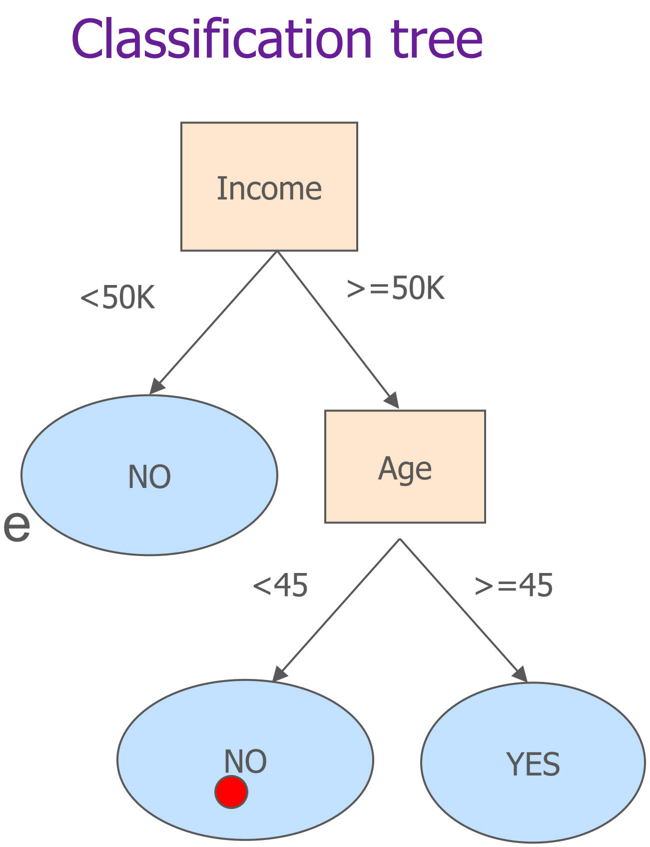
Trees as Sets of Rules



- IF (Employed = Yes) THEN Class=No Write-off
- IF (Employed = No) AND (Balance < 50k) THEN Class=No Write-off
- IF (Employed = No) AND (Balance ≥ 50k) AND (Age < 45) THEN Class=No Write-off
- IF (Employed = No) AND (Balance ≥ 50k) AND (Age ≥ 45) THEN Class=Write-off

What are we predicting?



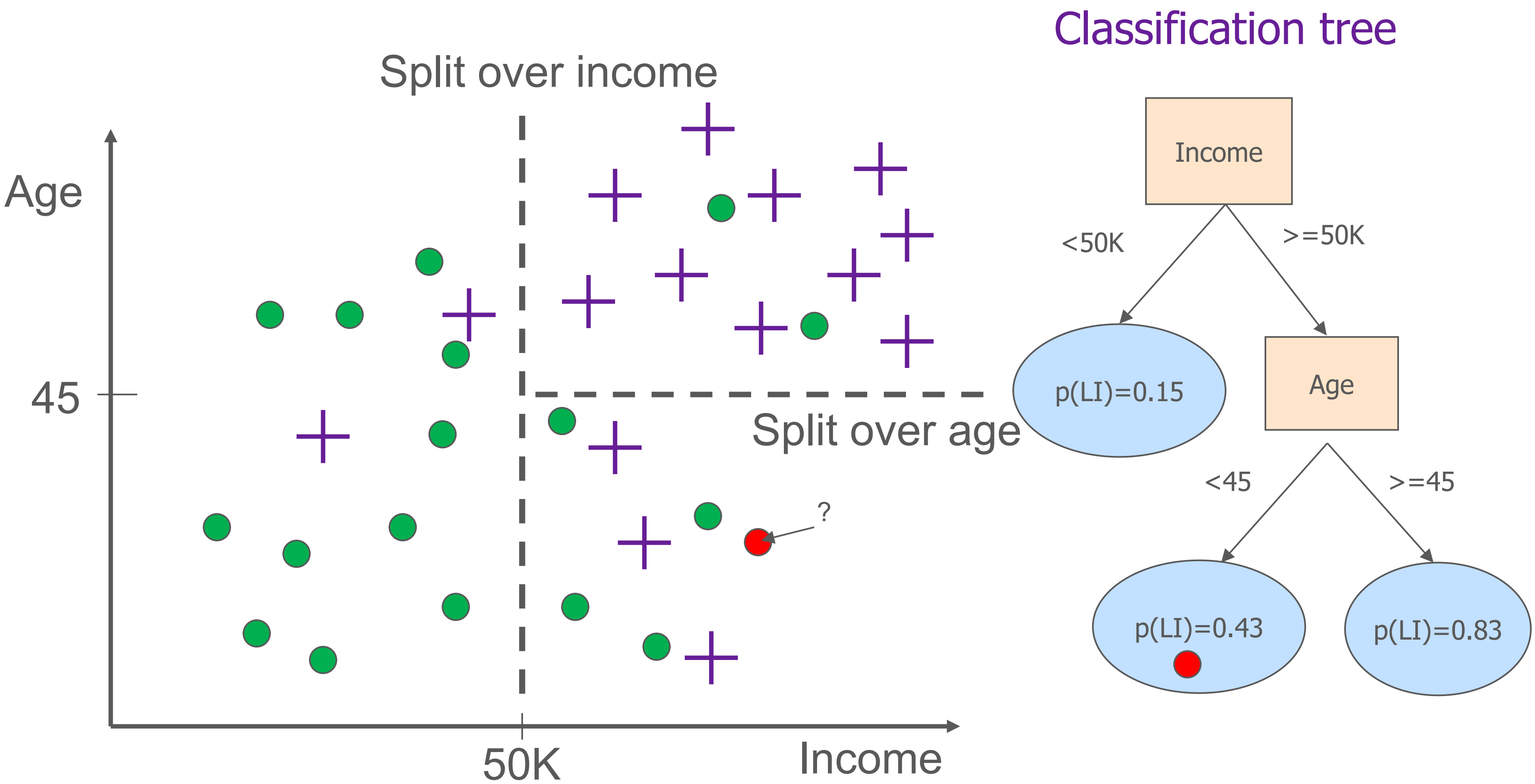


Did not buy life insurance

Bought life insurance

Interested in LI? = NO

What are we predicting?



Did not buy life insurance

Bought life insurance

Interested in LI? = 3/7

Questions?