Introduction to Predictive Modeling

Dr. Yi Long (Neal)

Most contents (text or images) of course slides are from the following textbook

Provost, Foster, and Tom Fawcett. Data Science for Business: What you need to know about data mining and data-analytic thinking. " O'Reilly Media, Inc.", 2013

Outline

- Overview of Predictive Modeling
- Variable Selection
- Introduction of Decision Trees
- Quiz

Predictive Modeling

- **A model is a simplified representation of reality created to serve a purpose**
  - ✓ Based on some assumptions about what is and is not important
  - ✓ Map, Black-Scholes model of option pricing, PB/PE for firm value
- **A predictive model is a formula for estimating the <u>unknown value of interest</u>: the target**
  - ✓ A formula can be a set of rules, a mathematical function, neural networks
  - ✓ Tasks can be classification, regression, link prediction, recommendation …
- **Prediction = estimate an unknown value**
  - ✓ Credit scoring, spam filtering, fraud detection, sentiment analysis

Model Learning/Training/Induction/Fitting

- **Supervised learning**
  - ✓ Describe a relationship between a set of selected variables (attributes/features) and a predefined variable (target), e.g., target as a function of input features

    f(1,1)=2, f(1,2)=3, f (2,2)=4, f(2,3) =5, f(3,8)=11 ➡ f(x1,x2) =x1+x2

- **Model induction**
  - ✓ Create model from data, but usually refers to generalizing from specific cases to general rules

    Higher P/E ➡ More likely to be overpriced 朝霞不出门，晚霞走千里

  - ✓ The input used for inducing/training/learning/fitting the model, are called the <u>training data</u>, and are always <u>labeled data</u>

Labelled/Training Data

Attributes

Target attribute

Name	Balance	Age	Employed	Write-off
Mike	$200,000	42	no	yes
Mary	$35,000	33	yes	no
Claudio	$115,000	40	no	no
Robert	$29,000	23	yes	yes
Dora	$72,000	31	no	no

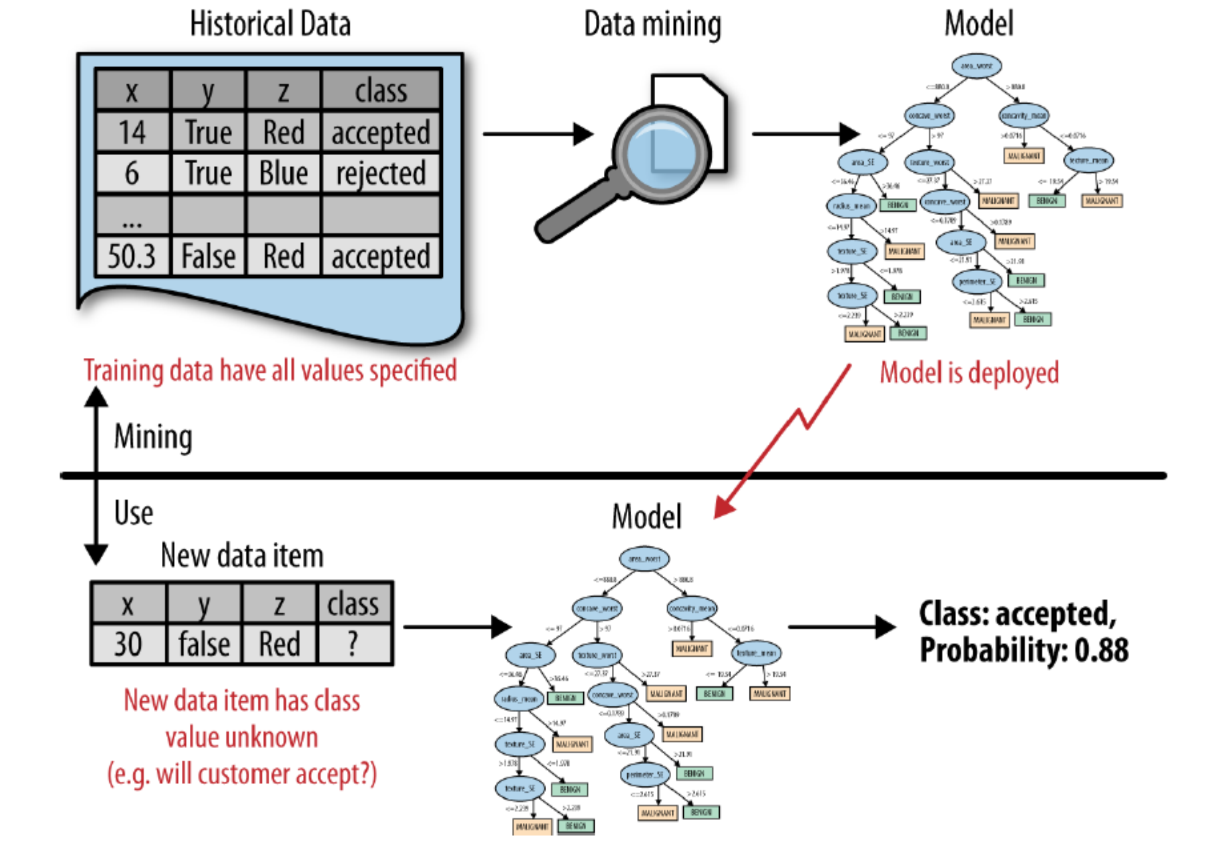
This is one row (example).
Feature vector is: **<Claudio,115000,40,no>**
Class label (value of Target attribute) is **no**

Model Induction and Prediction

f(1,1)=2, f(1,2)=3, f (2,2)=4, f(2,3) =5, f(3,8)=11 ➔ f(x1,x2) =x1+x2 ➔ f(4,3)=7

Mining = Model induction (training)

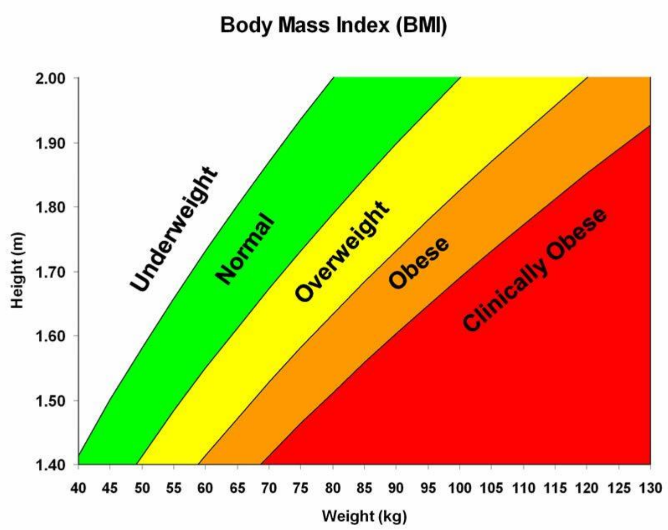
Prediction= Use



Supervised Segmentation

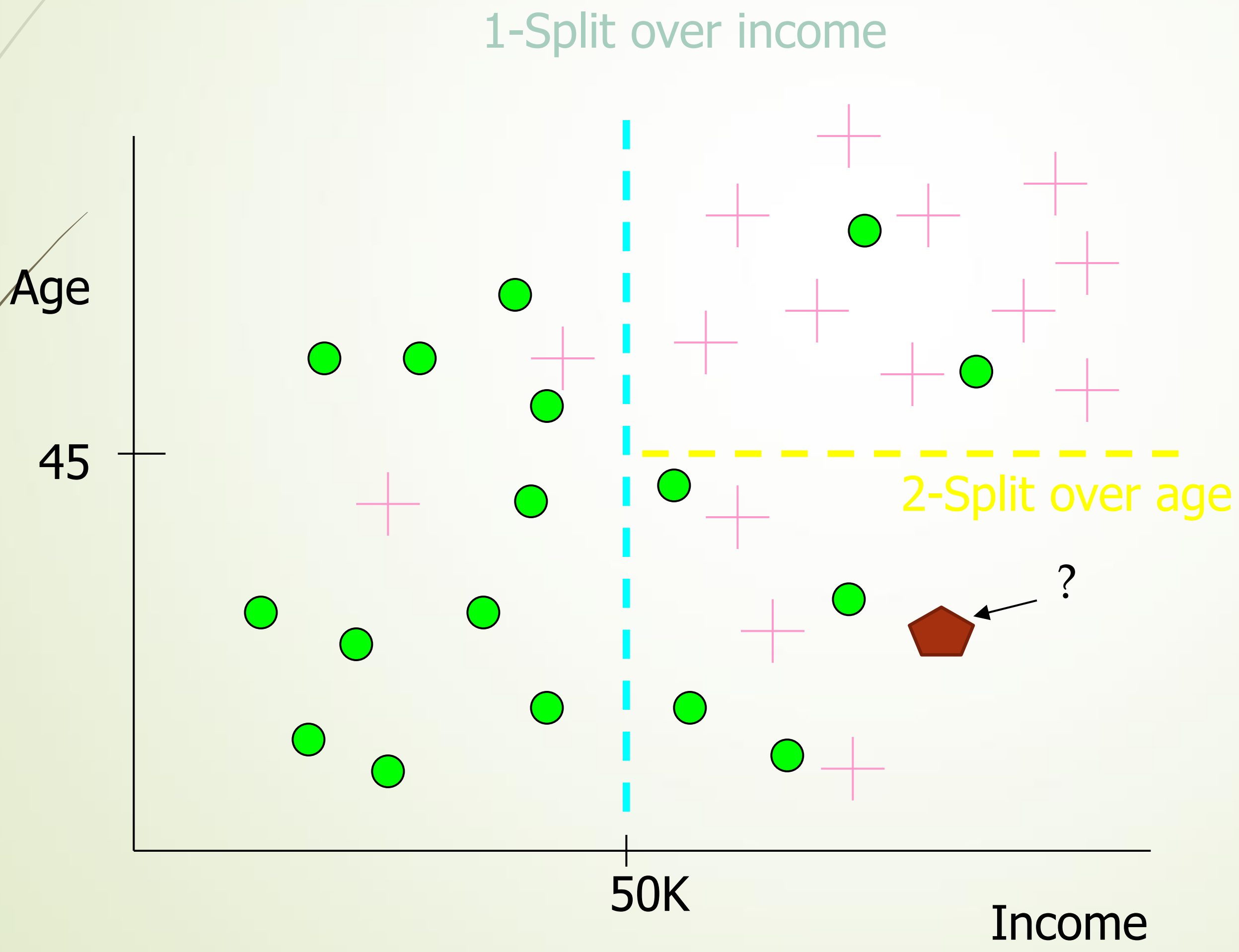
- **Supervised Segmentation is an Intuitive approach for prediction**
  - ✓ Segment the population into subgroups that have different values for the target variable (and within the subgroup the instances have similar values for the target)
  - ✓ Supervised Segmentation on multiple attributes one by one is much easier to explain and understand(e.g., Middle-aged professionals who reside in New York City on average have a churn rate of 5%")
  - ✓ Supervised Segmentation on multiple attributes together is also powerful

$$BMI = \frac{\text{weight (kg)}}{\text{height}^2\ (m^2)}$$



Segmentation on Attribute Iteratively

Segmentation for targeting our Life Insurance product (decision lines)



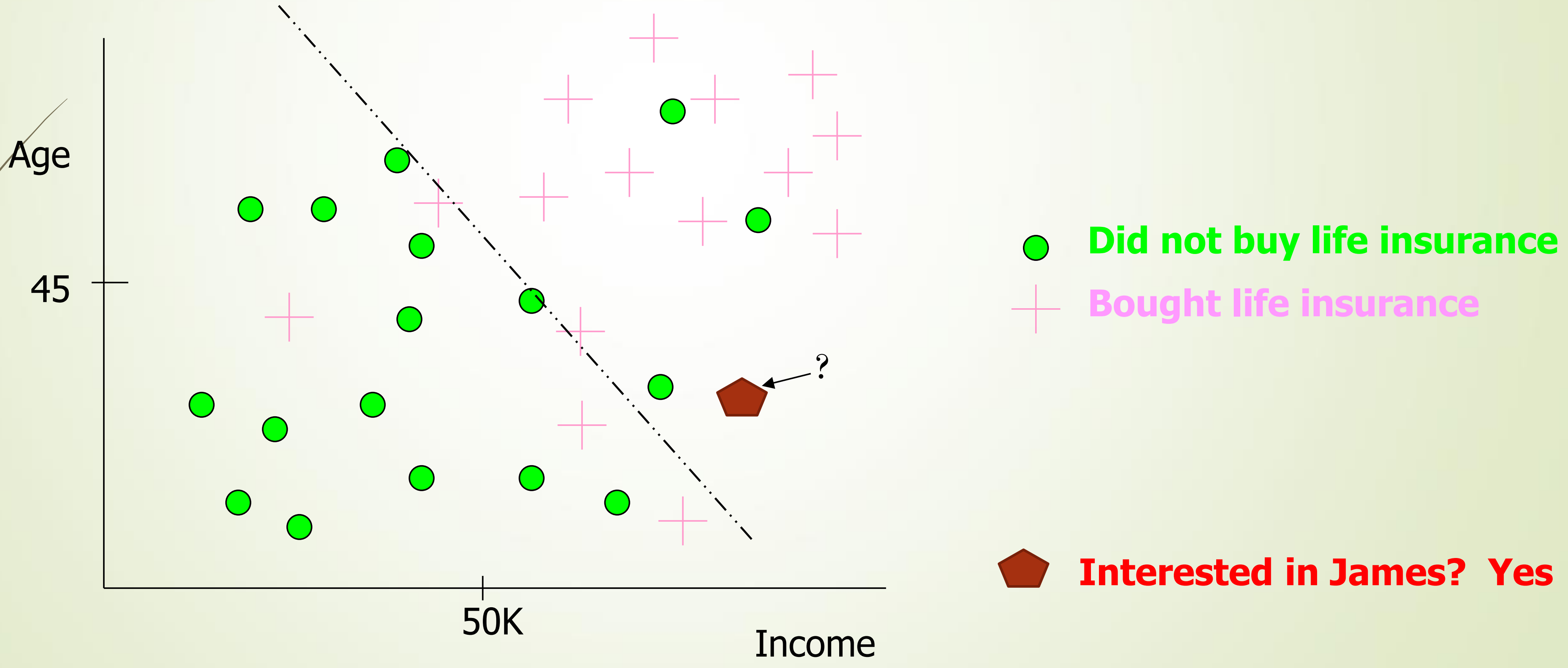
Did not buy life insurance

Bought life insurance

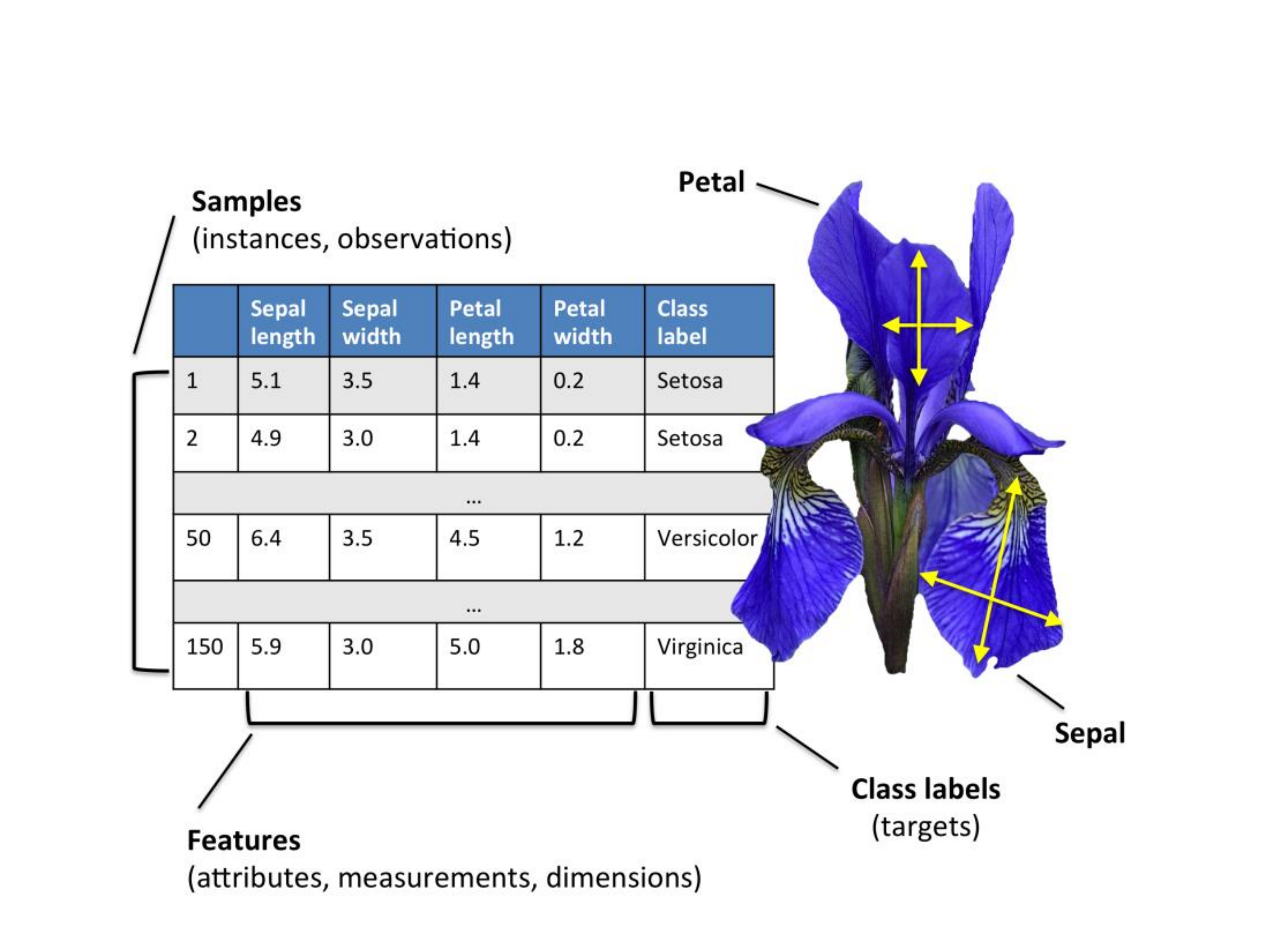
Interested in James? NO

Segmentation on Multiple Attribute

Segmentation for targeting our Life Insurance product (hyper-planes)

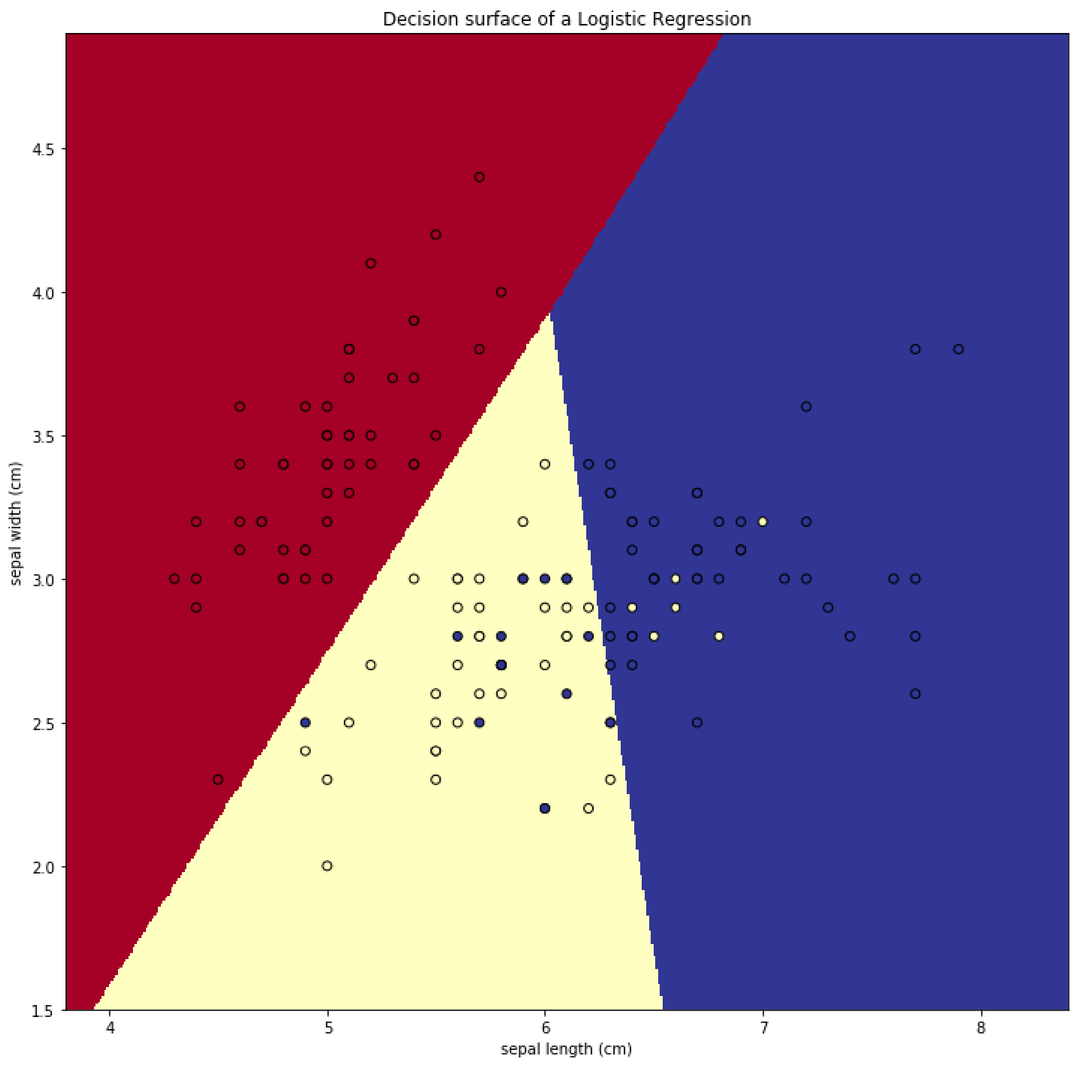


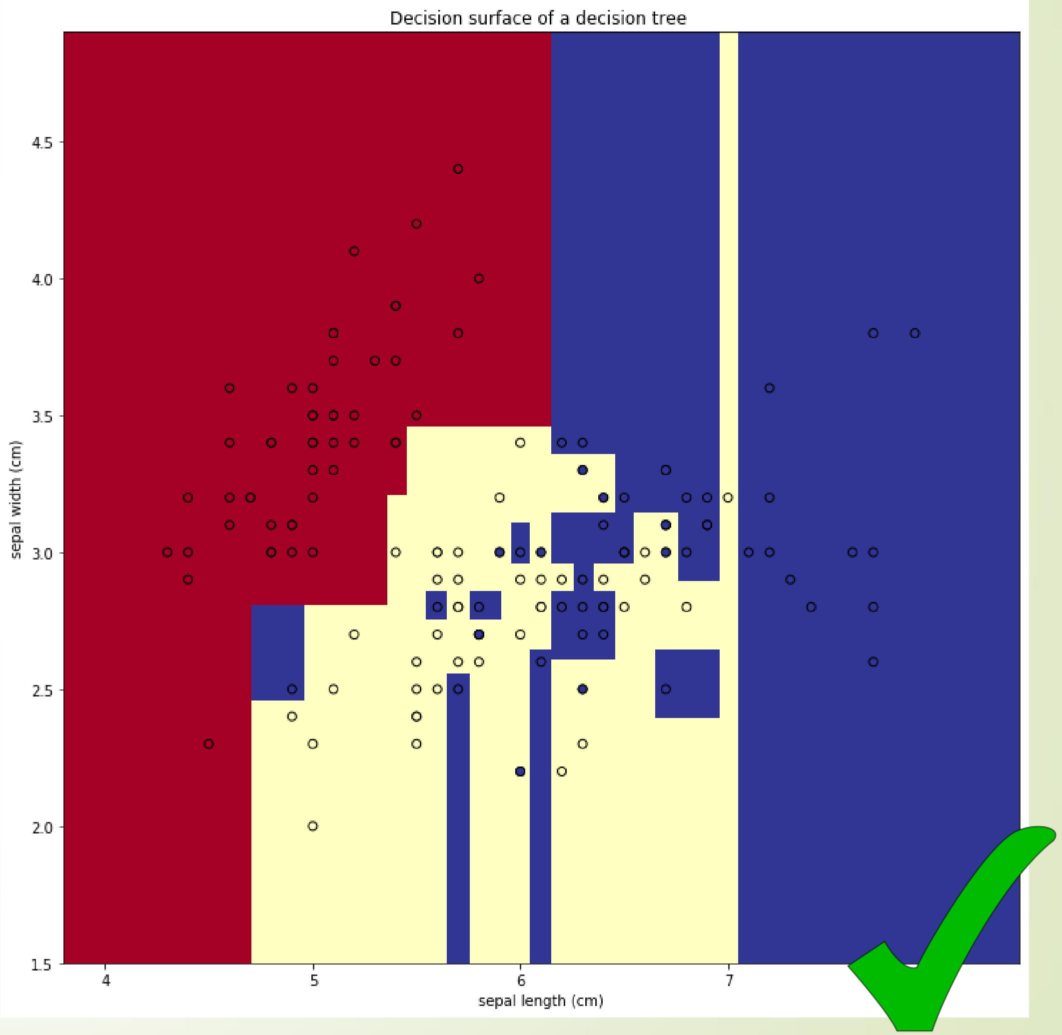
Iris Dataset



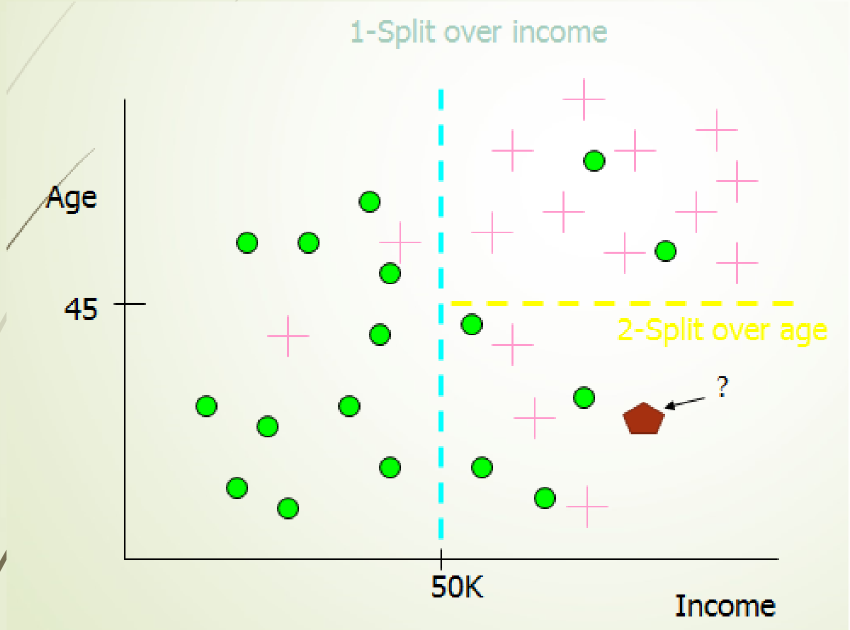
	Sepal length	Sepal width	Petal length	Petal width	Class label
1	5.1	3.5	1.4	0.2	Setosa
2	4.9	3.0	1.4	0.2	Setosa
...					
50	6.4	3.5	4.5	1.2	Versicolor
...					
150	5.9	3.0	5.0	1.8	Virginica

Segmentation Boundary





Attribute Selection



Customers has income higher than 50K and older than 45 years old are likely to buy life insurance

How can we (automatically) identify/rank attributes with important information about the target variable, such as income/age here?

Python Class and Object

- Python is an object oriented programming language.
  - Almost everything in Python is an object, with its **methods** and **properties**
  - Methods in objects are functions that belong to the object.
- A **Class** is like an object constructor, or a "blueprint" for creating objects.
  - The __init__() Function: **all** classes have a function called __init__(), which is always executed **once when the class is being initiated**.
  - To create a class, use the keyword class
  - The self parameter is a reference to the **current instance** of the class, and is used to access variables that belongs to the class.
- **Properties** and **methods** can be accessed using dot (.) operator. Eg.: myObject.myattribute
- Class level properties or Functions (*)

Any Difference?

model_1 = Sentiment_Polarity**()**
model_1.analyze(article)

?
=

model_1 = Sentiment_Polarity
model_1.analyze(article)

|| ?

|| ?

Sentiment_Polarity**()**.analyze(article)

?
=

Sentiment_Polarity.analyze(article)

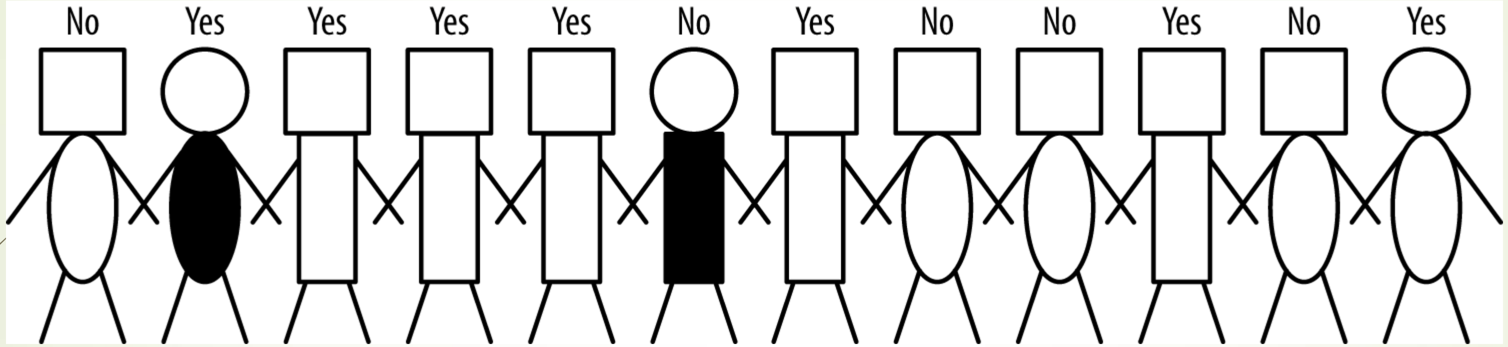
Common Error

model_1 = Sentiment_Polarity**()** model_1.analyze(article)	≠	model_1 = Sentiment_Polarity model_1.analyze(article)
‖		‖
Sentiment_Polarity**()**.analyze(article)	≠	Sentiment_Polarity.analyze(article)

Outline

- Overview of Predictive Modeling
- Variable Selection
- Introduction of Decision Trees
- Quiz

Simplified Churn Prediction



A set of people to be classified. The label over each head represents the value of the target variable (write-off or not). Colors and shapes represent different values of attributes.

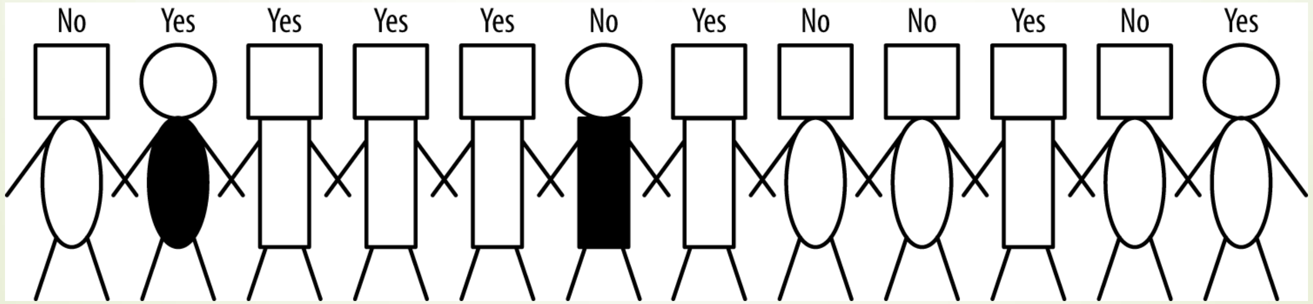
- Attributes:
  — head-shape: square, circular
  — body-shape: rectangular, oval
  — body-color: gray, white
- Target variable:
  — write-off: Yes, No

Selecting Informative Attributes

- **Segment the population into subgroups that have different values for the target variable (and within the subgroup the instances have similar values for the target)**

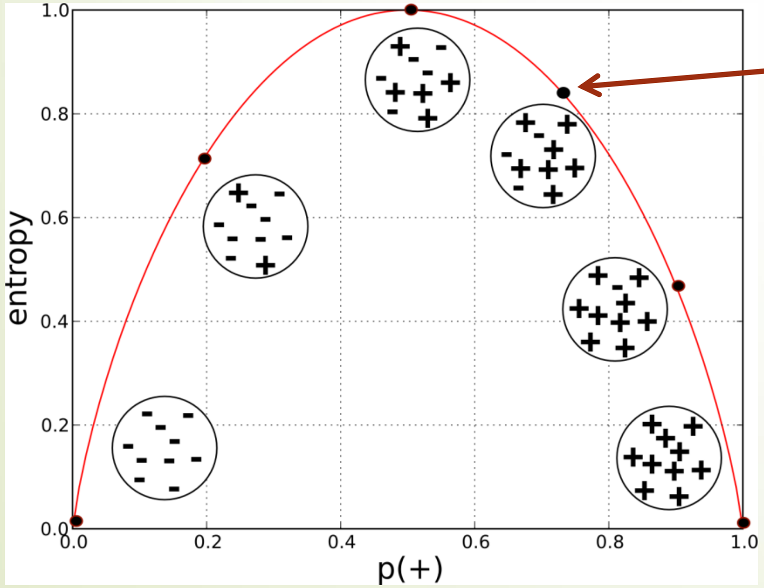
  Pure subgroups(pure means homogeneous with respect to the target variable)

- **Attributes rarely split a group perfectly**
  - ✓ Selecting attributes that can reduce the impurity substantially
  - ✓ How to measure the "**Impurtiy**" of a group?



Entropy as Impurity Measure

- **Entropy proposed by Shannon can measure how mixed/impure a population is**
  - ✓ Definition: $entropy = -p_1 \log(p_1) - p_2 \log(p_2) - \cdots$
  - ✓ p_i is the relative percentage of instances with target label being class-i
  - ✓ Example: population S has 10 instances of two classes, + and –.



When there are 7 + instances and 3 – instances, then

p(+) = 0.7

p(-) = 0.3

$$
\begin{aligned}
entropy(S) &= -[0.7 \times \log_2(0.7) + 0.3 \times \log_2(0.3)] \\
&\approx -[0.7 \times -0.51 + 0.3 \times -1.74] \\
&\approx 0.88
\end{aligned}
$$

Selecting Attributes by Information Gain

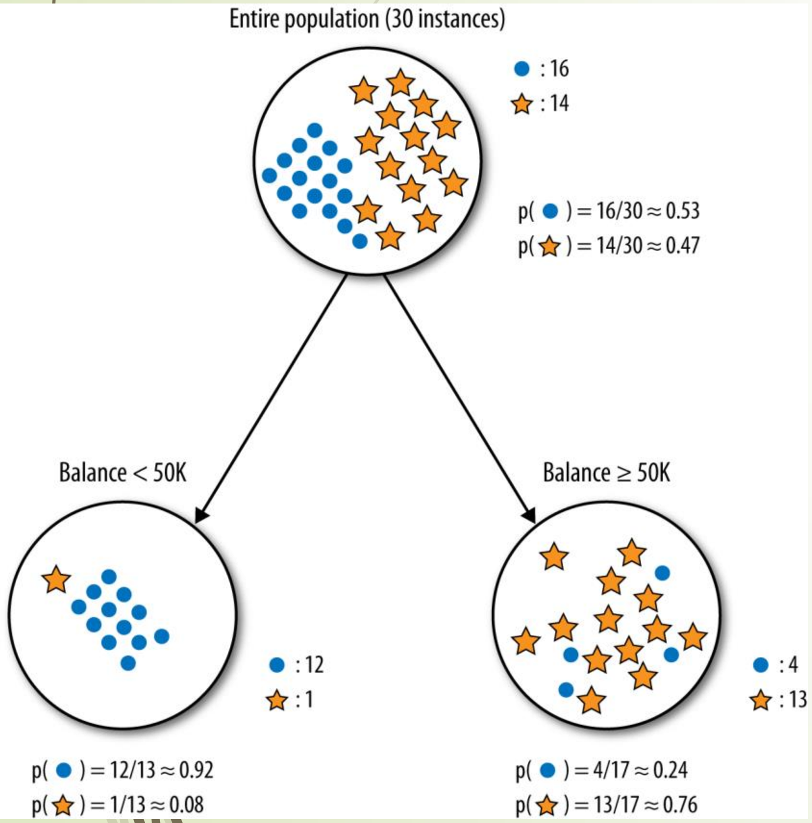
- **Information gain (IG)**
  - ✓ Segment <u>one parent</u> group to <u>multiple children</u> groups by a given attribute, then

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

  - ✓ The entropy for each child is <u>weighted</u> by the proportion of instances belonging to that child
  - ✓ Measure how much an attribute <u>decreases</u> entropy(impurity) over the whole segmentation it creates
  - ✓ Strictly speaking, IG measures the change in entropy due to any amount of <u>*new information added*</u>

Information Gain (Example1)

- **Two class prediction problem (• and ☆) and segment by *balance***



Parent

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

Left child

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

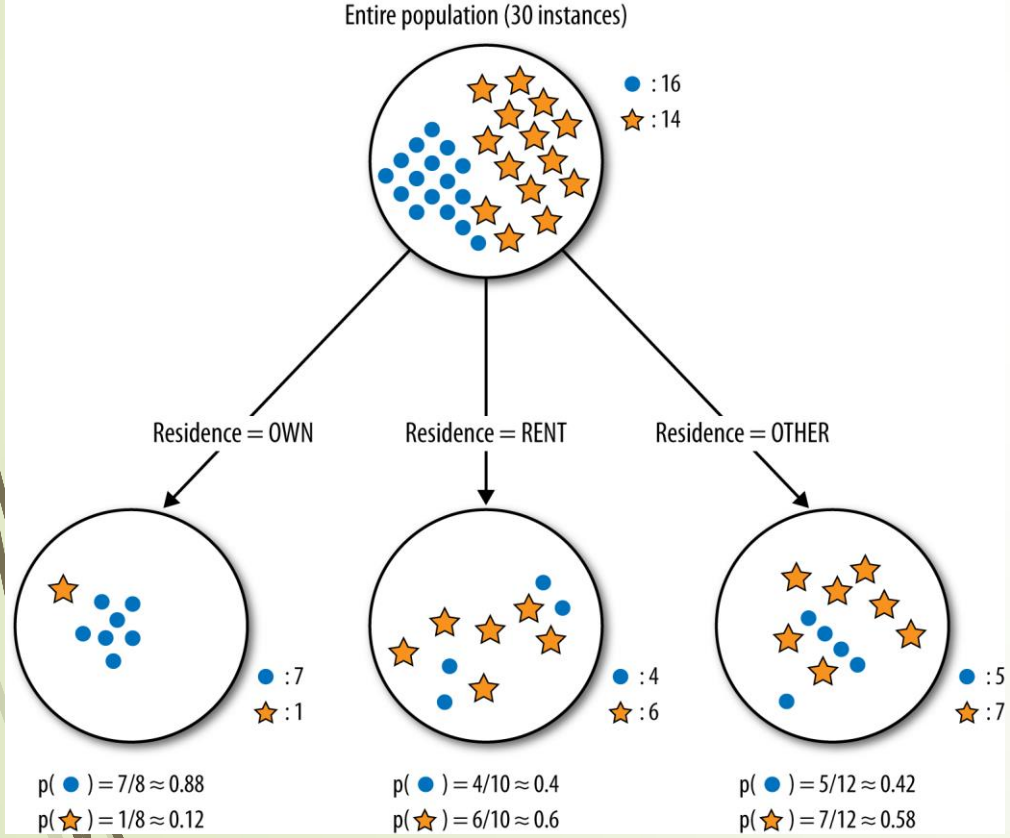
Right child

$$\begin{aligned} entropy(Balance \geq 50K) &= -[p(\bullet) \times \log_2 p(\bullet) + p(\star) \times \log_2 p(\star)] \\ &\approx -[0.24 \times (-2.1) + 0.76 \times (-0.39)] \\ &\approx 0.79 \end{aligned}$$

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

Information Gain (Example2)

- **Two class prediction problem (• and ☆) and segment by *<u>residence</u>***



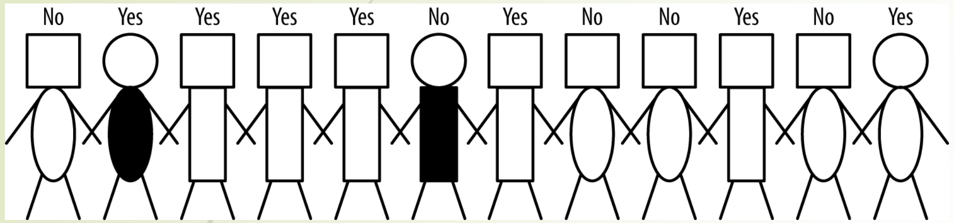
$$entropy(parent) \approx 0.99$$
$$entropy(\text{Residence=OWN}) \approx 0.54$$
$$entropy(\text{Residence=RENT}) \approx 0.97$$
$$entropy(\text{Residence=OTHER}) \approx 0.98$$
$$IG \approx 0.13$$

Segment by *<u>balance (IG=0.37)</u>* will bring more information gain than by *<u>residence (IG=0.13)</u>*

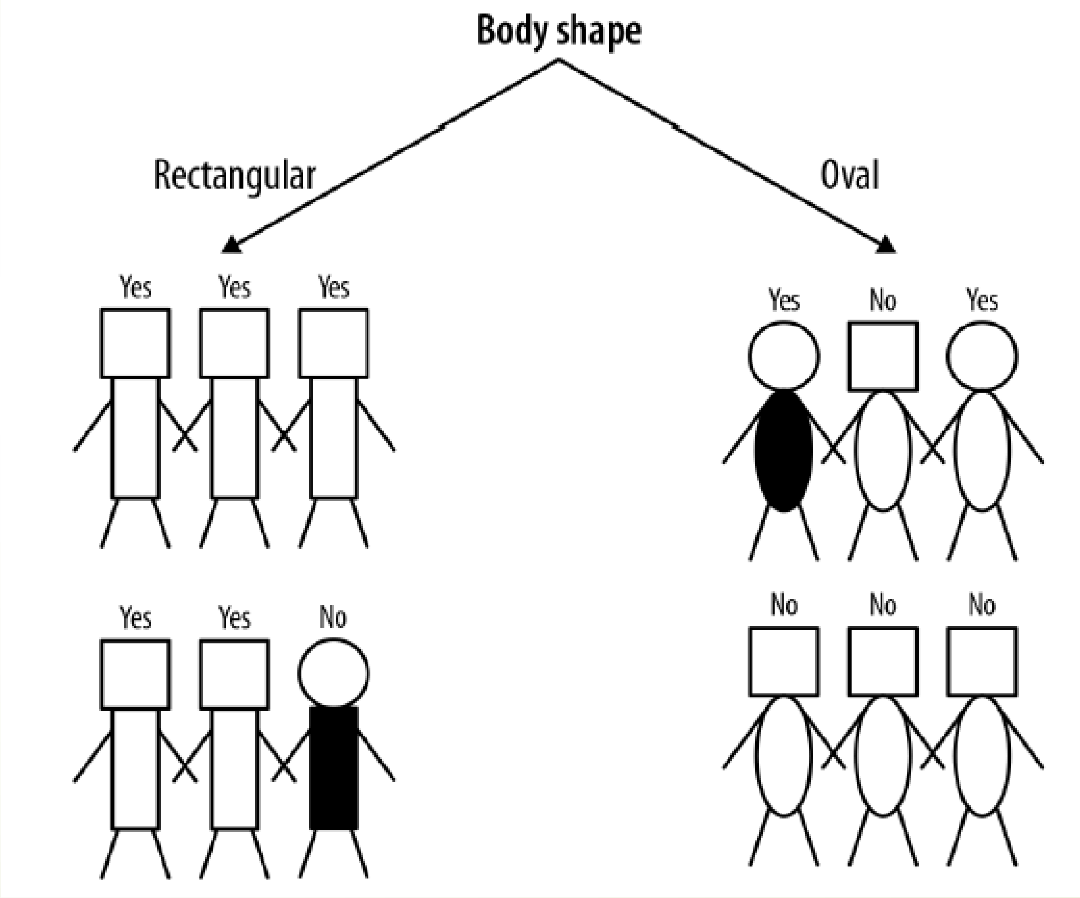
Handle Numerical Values

- **When attributes have numerical values, how to segment?**
  - ✓ "*Discretize*" numeric attributes by split points, such as "Income>50K ？"
  - ✓ How to decide the split points？ Among breakpoints that making changes
- **When the target label are numerical (regression problem)**
  - ✓ Information gain is not appropriate
  - ✓ Using variance to measure the impurity of groups
  - ✓ Choose best weighted average variance reduction (weighted by group size)

Best Segment for Churn Prediction



- Attributes:
  - — head-shape: square, circular
  - — body-shape: rectangular, oval
  - — body-color: gray, white
- Target variable:
  - — write-off: Yes, No

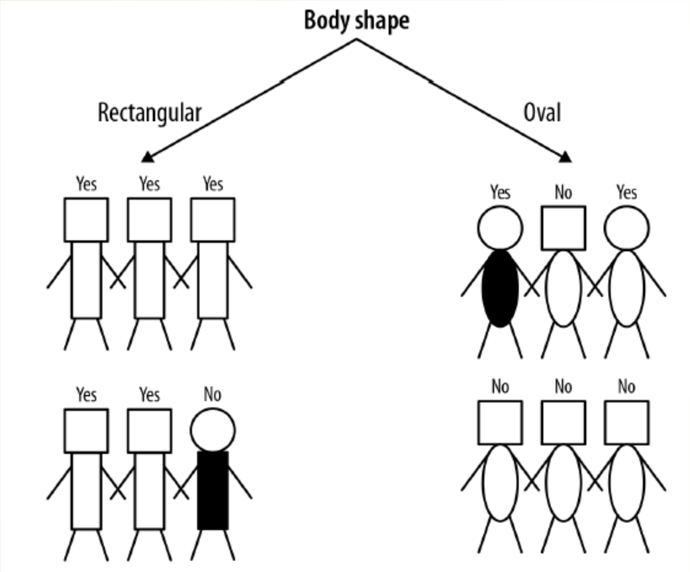


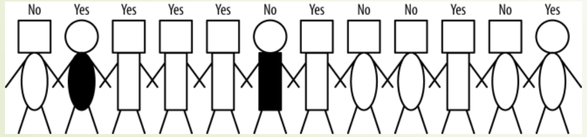
Outline

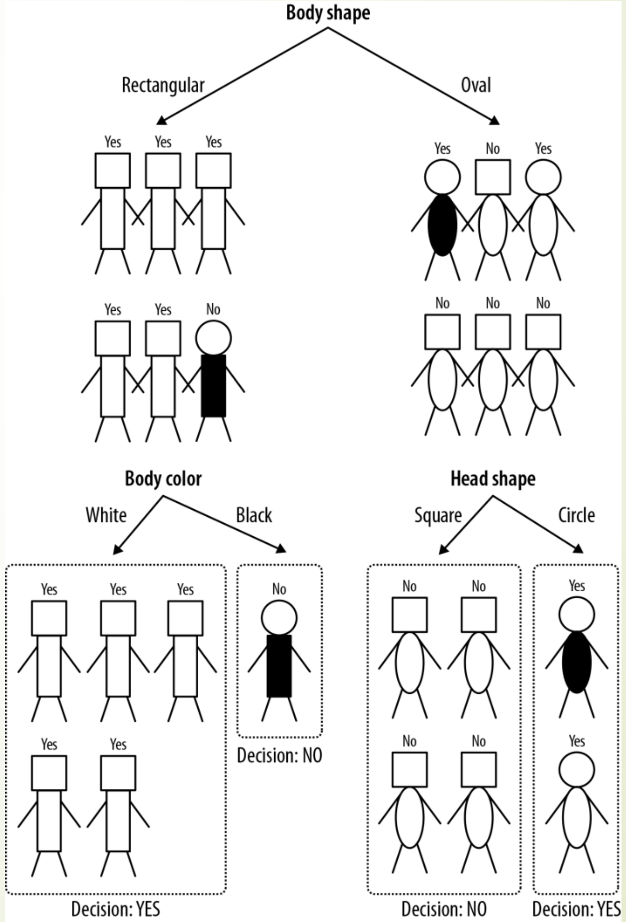
- Overview of Predictive Modeling
- Variable Selection
- Introduction of Decision Trees
- Quiz

Building Decision Trees

- **Recursively apply attribute selection to find the best attribute to partition the current groups into subgroups that are as pure as possible**

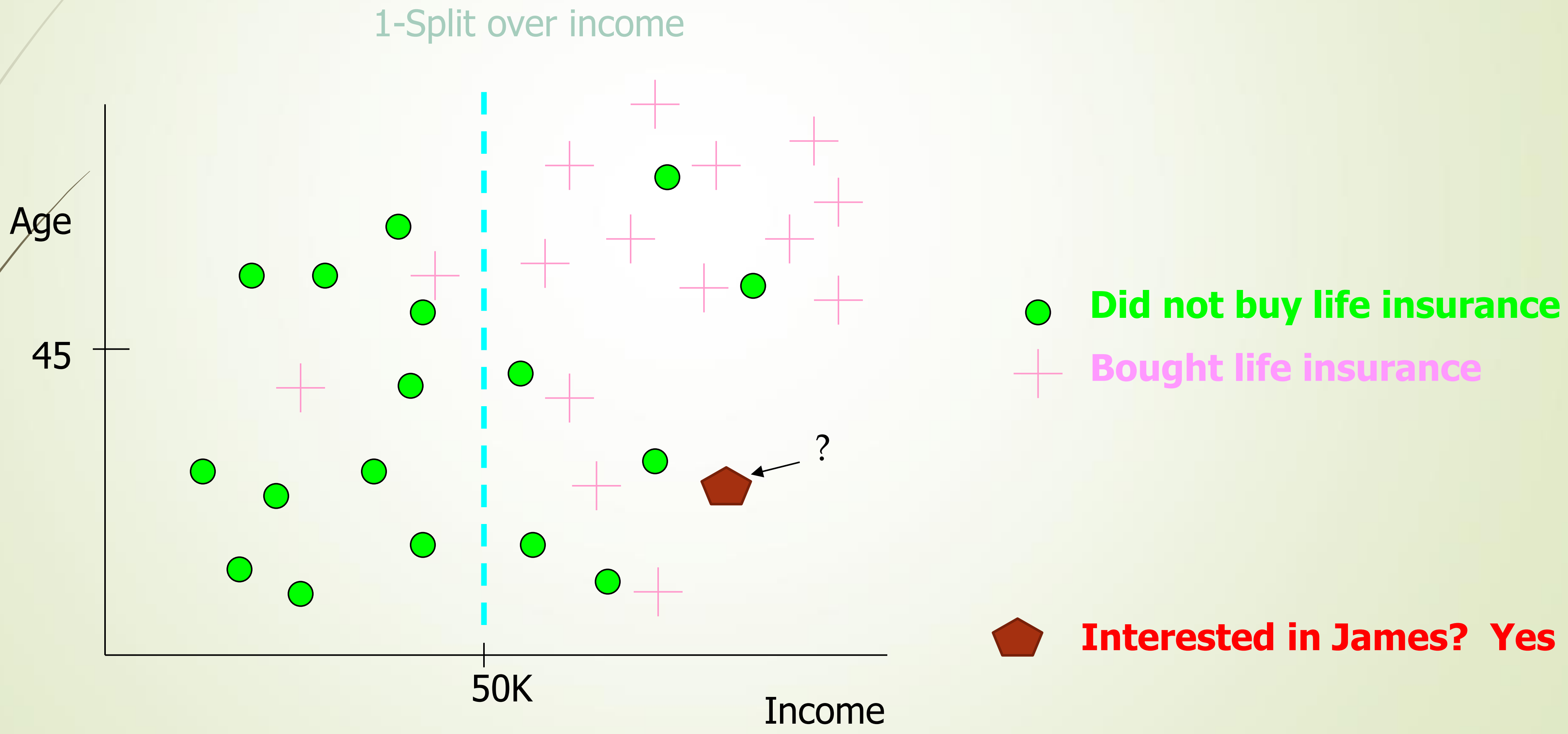






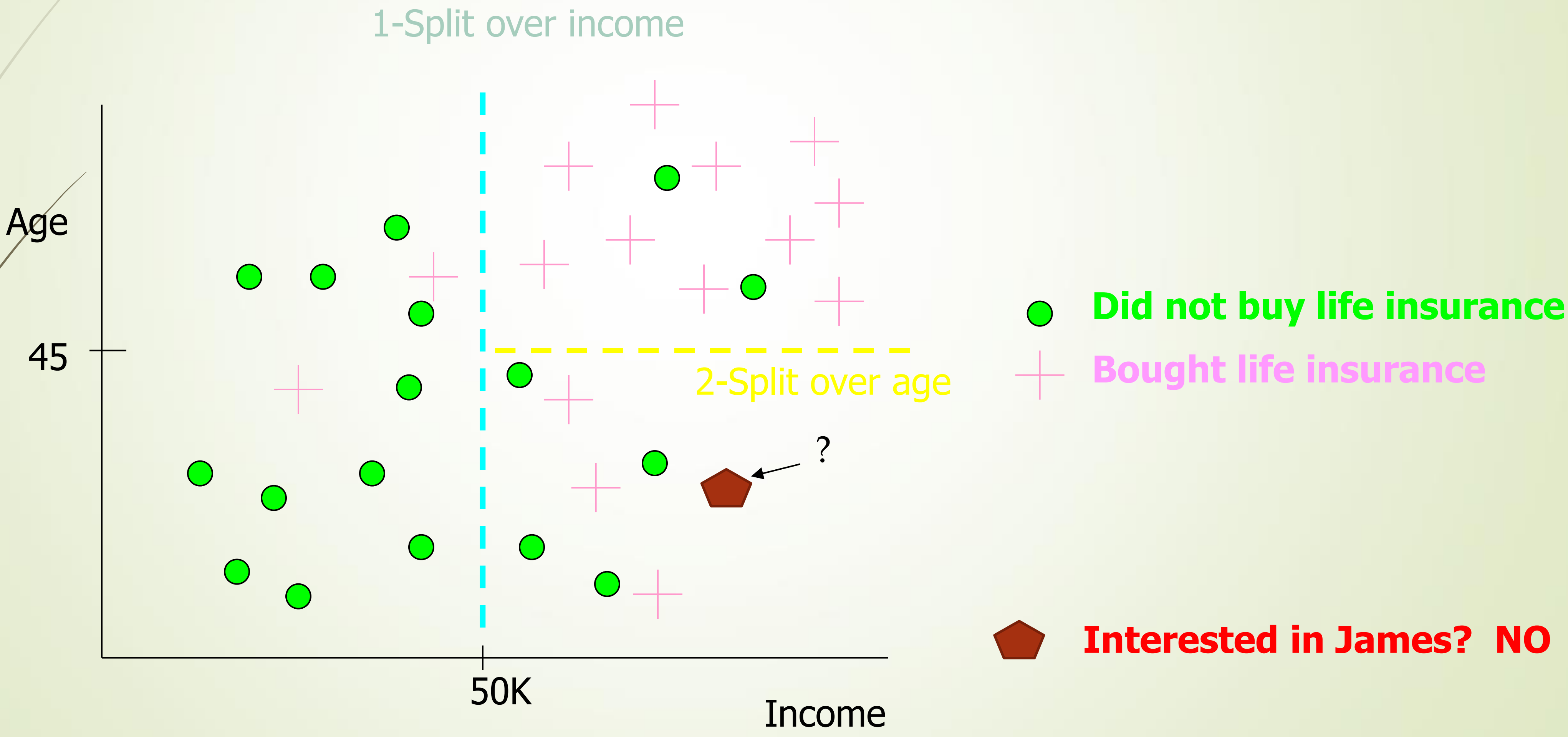
Segment by Attributes Iteratively(1)

- If we select multiple attributes step by step, each giving some information gain

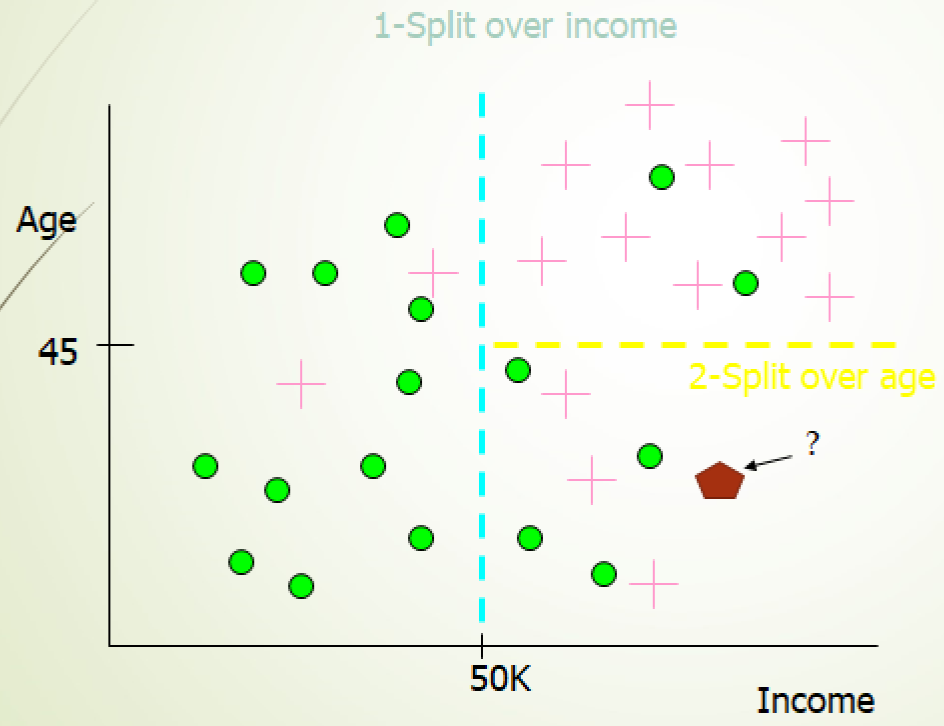


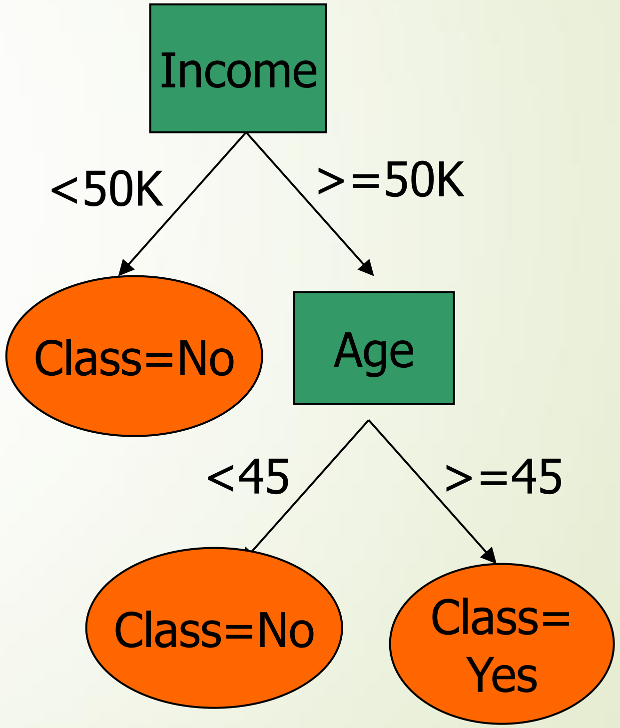
Segment by Attributes Iteratively(2)

- **If we select multiple attributes step by step, each giving some information gain**



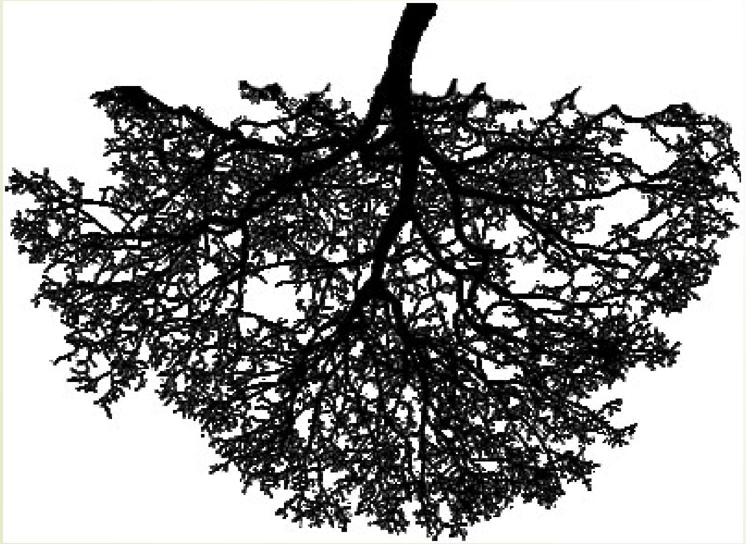
Put Segmentations Together

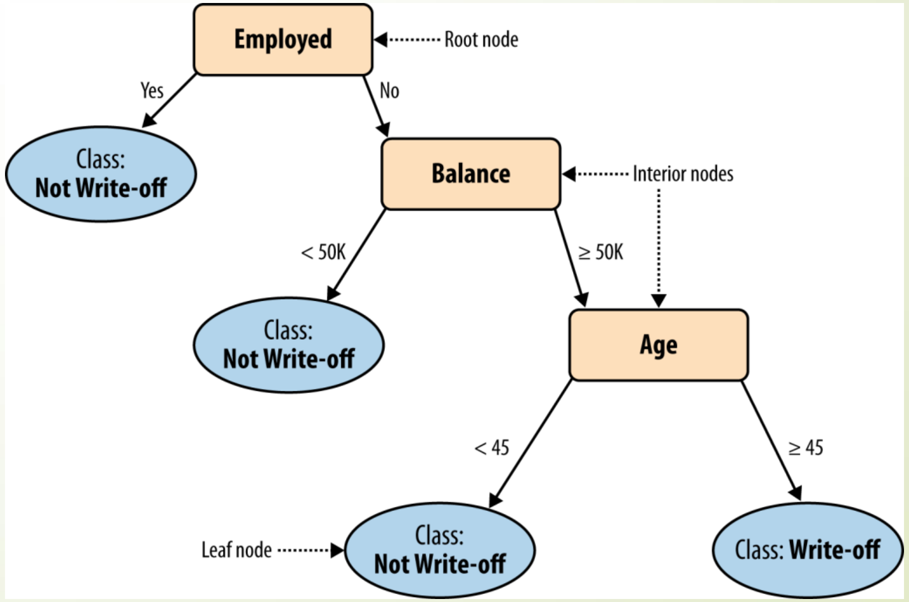




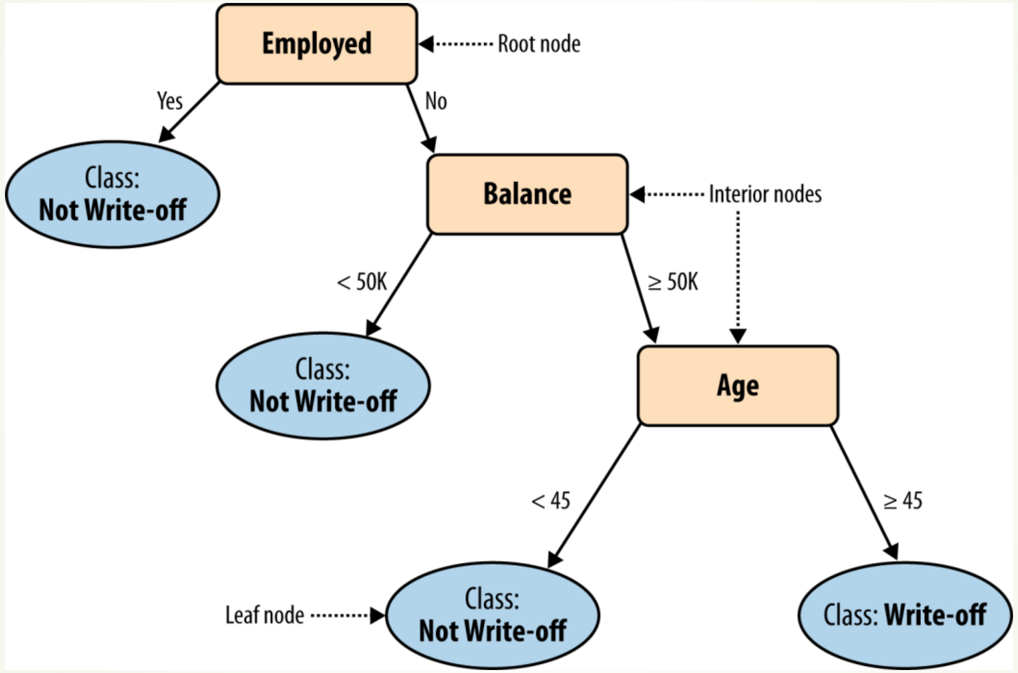
Decision Trees

- **Decision tree creates a segmentation of the data by multiple attributes**
  - ✓ Each ***internal node*** in the tree contains a test of an attribute
  - ✓ Each ***leaf node*** represents a class label (the attributes/values define the group characteristics)
  - ✓ Each ***path*** from root to leaf represent classification rules



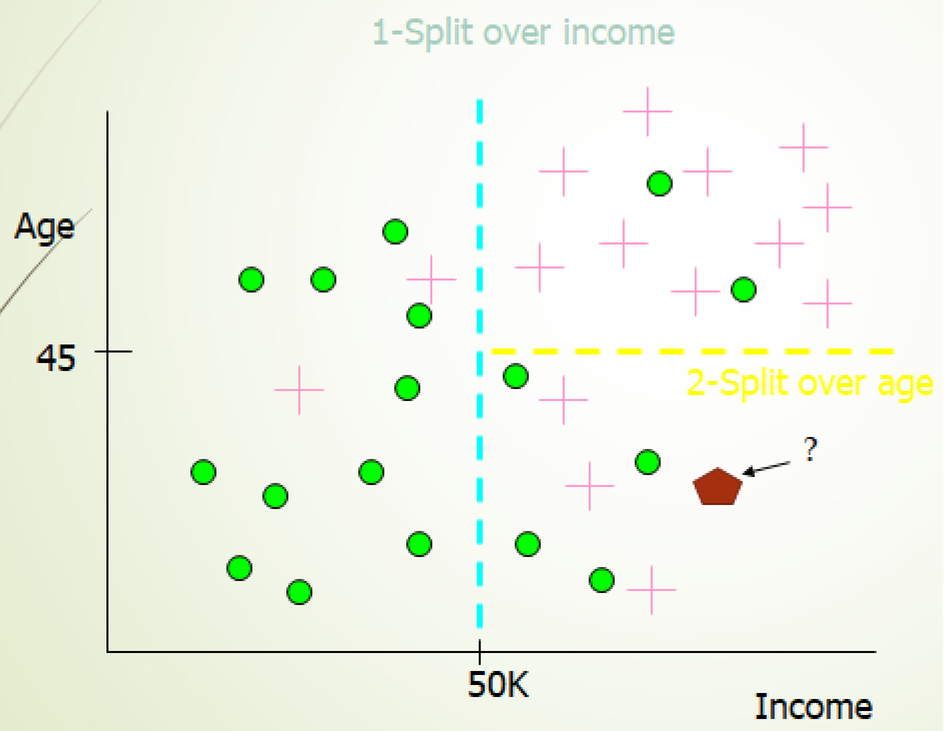


Decision trees Are A Set of Rules

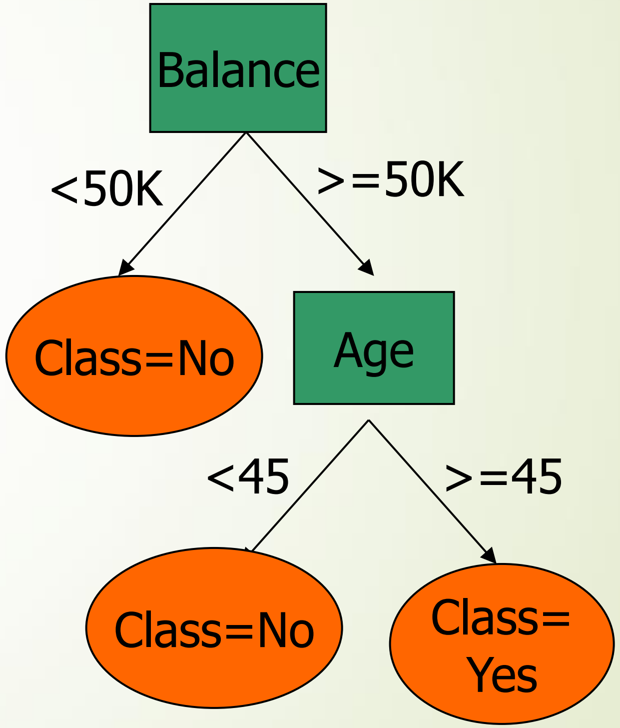


- ✓ IF (Employed = Yes) THEN Class=No Write-off
- ✓ IF (Employed = No) AND (Balance < 50k) THEN Class=No Write-off
- ✓ IF (Employed = No) AND (Balance ≥ 50k) AND (Age < 45) THEN Class=No Write-off
- ✓ IF (Employed = No) AND (Balance ≥ 50k) AND (Age ≥ 45) THEN Class=Write-off

Decision Tree Visualization



Scatterplot of instances



Tree model

Advantages of Decision Tree

- **Decision trees should be tried at first in general for prediction tasks**
  - ✓ Are simple to understand and interpret (most important)
  - ✓ Easy to be combined with other decision techniques, especially expertise
  - ✓ Nonlinear classification with relatively efficient performance
  - ✓ Require relatively little effort from users for data preparation: do not need to rescale and can handle categorical attributes naturally

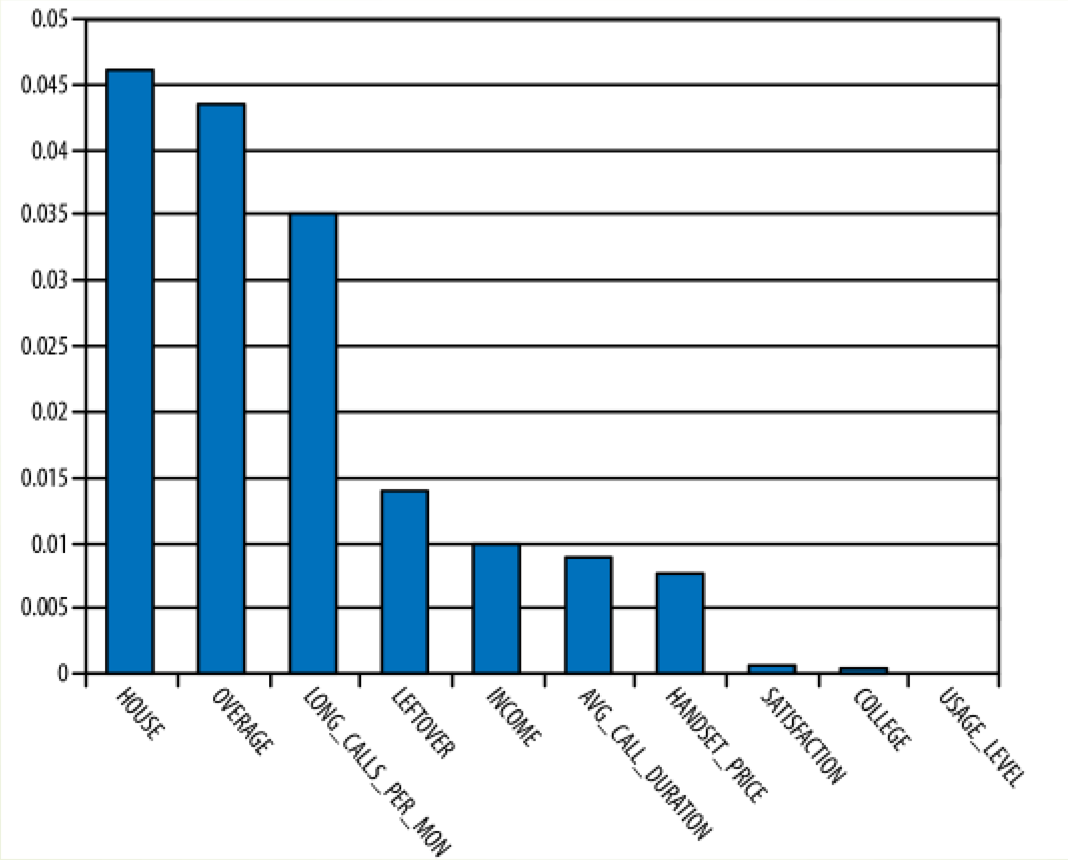
Example of Churn Prediction (1)

- **Given a historical data set of 20,000 customers**
  - ✓ Each customer either had stayed with the company or had left (churned)
  - ✓ Each customer is described by following attributes
  - ✓ How could we predict the churn probability of a new customer

Variable	Explanation
COLLEGE	Is the customer college educated?
INCOME	Annual income
OVERAGE	Average overcharges per month
LEFTOVER	Average number of leftover minutes per month
HOUSE	Estimated value of dwelling (from census tract)
HANDSET_PRICE	Cost of phone
LONG_CALLS_PER_MONTH	Average number of long calls (15 mins or over) per month
AVERAGE_CALL_DURATION	Average duration of a call
REPORTED_SATISFACTION	Reported level of satisfaction
REPORTED_USAGE_LEVEL	Self-reported usage level
LEAVE *(Target variable)*	Did the customer stay or leave (churn)?

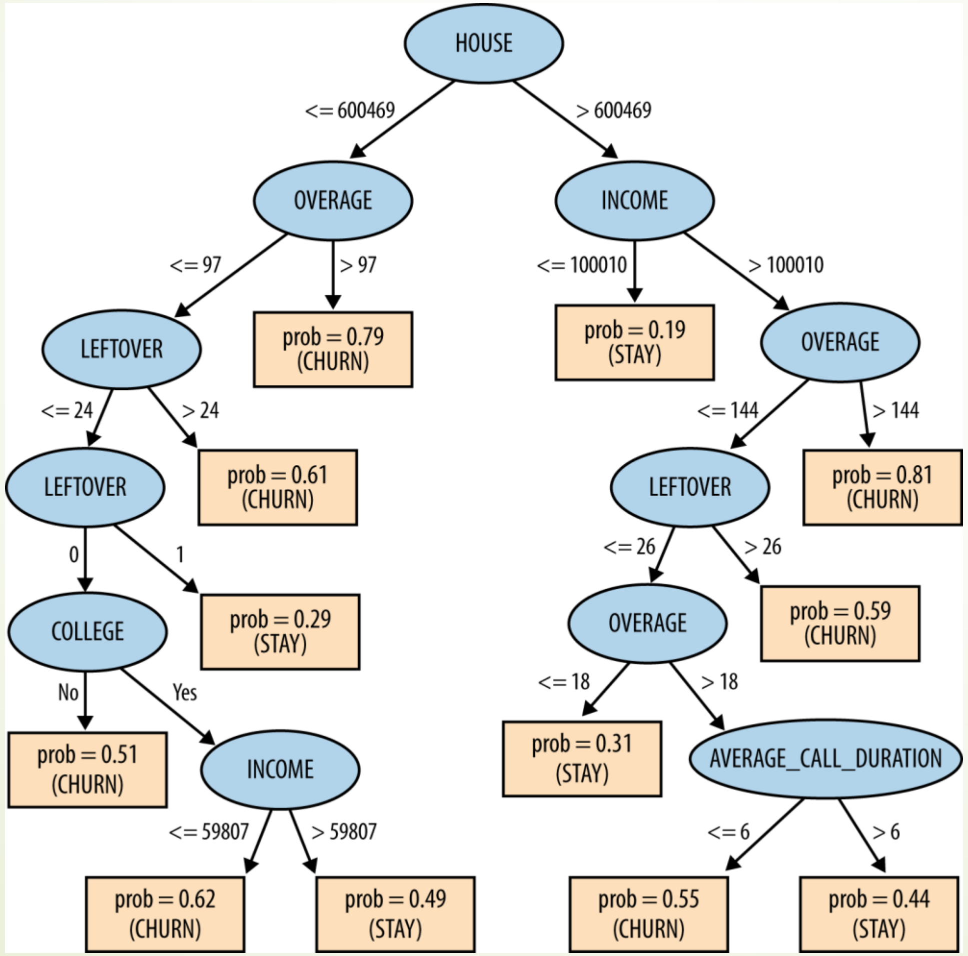
Example of Churn Prediction (2)

- **Ranking 10 informative attributes by information gain**



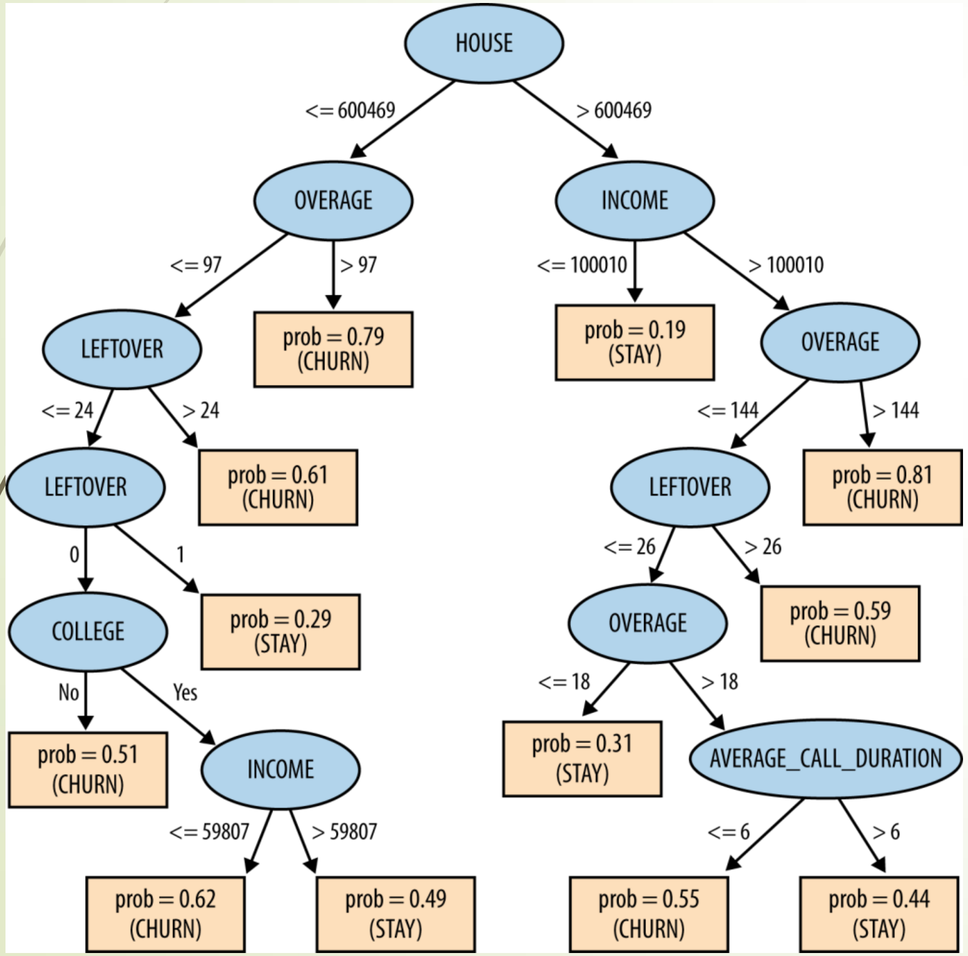
Example of Churn Prediction (3)

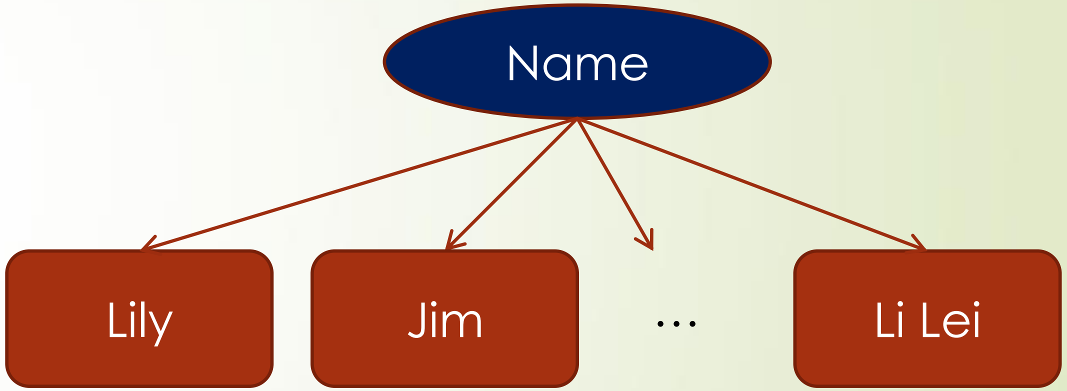
- **Recursively apply attribute selection and segmentation**



When to Stop Growing

- Grow as long as we have positive information gain?





Outline

- Overview of Predictive Modeling
- Variable Selection
- Introduction of Decision Trees
- Quiz

Lab Quiz-3

- **Deadline**: 17:59 p.m., Mar. 6, 2020
- Two questions accounting for **5%** of overall score
- **Upload** the **answer worksheet** and the accomplished **Python files** to the **Blackboard**
- You may submit **unlimited times** but only the **LAST** submission will be considered
- Note： **MUST attach ALL** the required files in every submission/resubmission, otherwise other files will be missing.