دانشگاه آزاد اسلامی واحد تبریز

نام درس: داده کاوی

بخش: طبقه بندی توسط درخت تصمیم

نام استاد: دکتر مسعود کارگر

Roadmap

- A Game for you!
- What is Decision Tree?
- Information Theory
- How to build Decision Tree
- Summary

Review of Classification

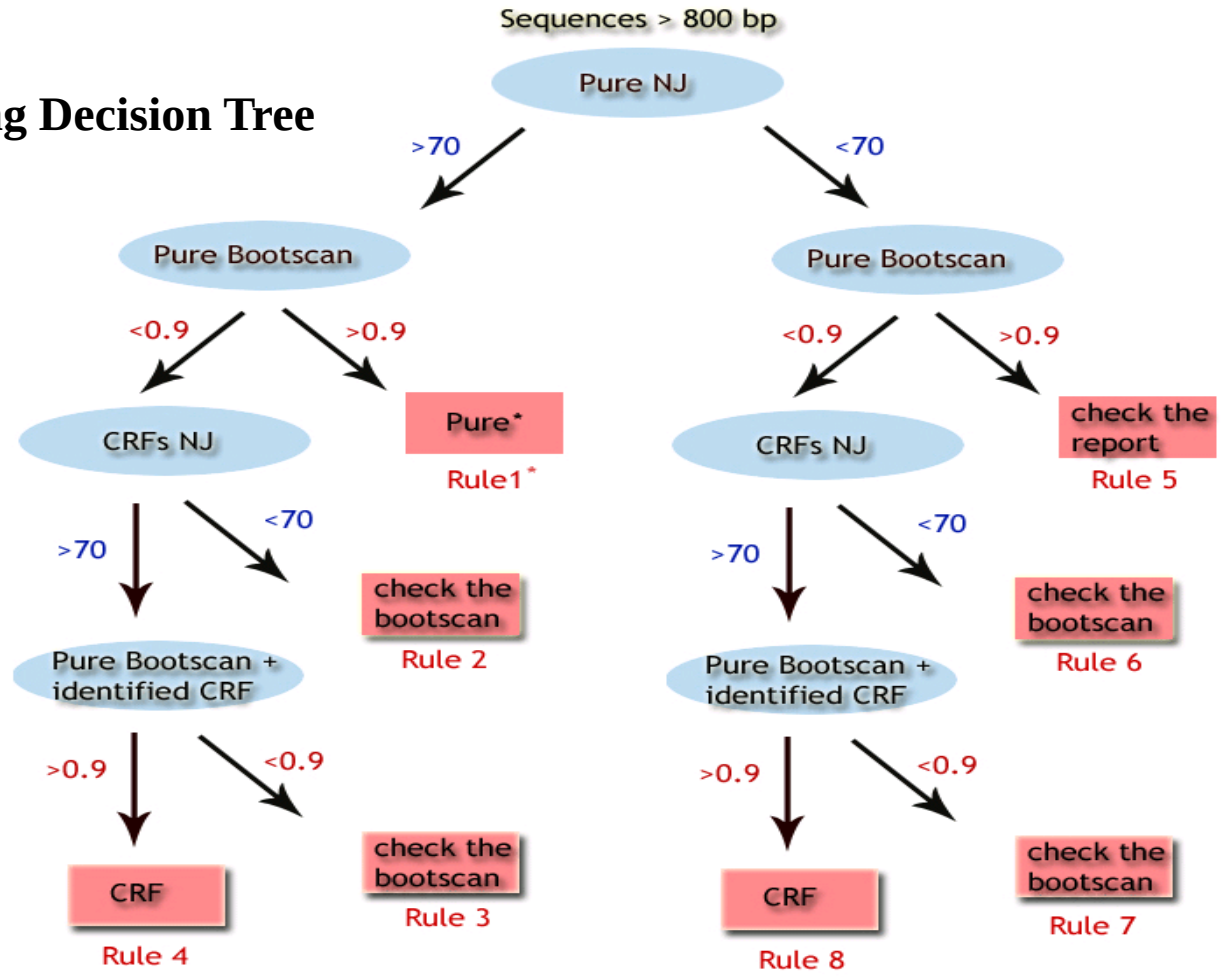
- Given a set of attributes (X1,X2, …, Xn) of an object, predict its label/class (C) based on training examples
- Three types of attributes:
  - *Numerical/continuous*: Domain is ordered and can be represented on the real line (e.g., age, income)
    - (0.3, 0.4, 0. 5, …)
  - *Ordinal*: Domain is ordered, but absolute differences between values is unknown (e.g., preference scale, severity of an injury)
    - Grade: (A, B, C, D)
  - *Nominal* or *categorical*: Domain is a finite set without any natural ordering (e.g., occupation, marital status, race)
    - Color: (red, blue, yellow)

Game of Guessing Animal

- I write dov ow)
- You are ex the animal to questions
- The winne he minimal N
- The tricky what ques

What is Decision Tree?

REGA HIV-1 Sub tying Decision Tree Classifier



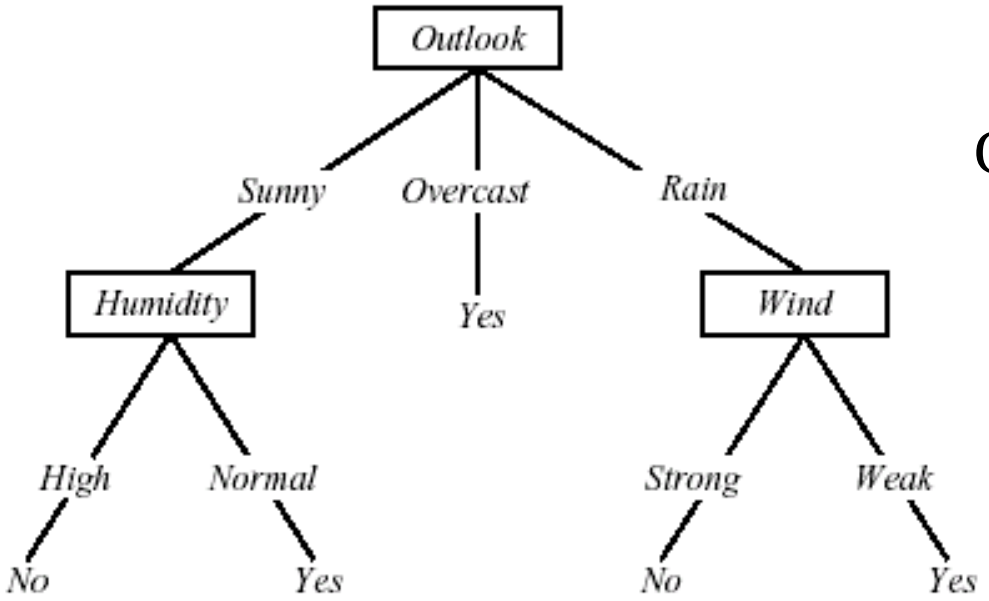
What is Decision Tree?

- They do **classification**: predict a categorical output from categorical and/or real inputs
- Decision trees are the single ***most popular data mining tool***
  - Easy to understand
  - Easy to implement
  - Easy to use
  - Computationally cheap
- **Mature, Easy-to-use software package freely available** (used for the assignment 2)
- **NO programming needed!**

What is Decision Tree?

- Extremely popular method
  - Credit risk assessment
  - **Medical diagnosis**
  - Market analysis
  - Bioinformatics
  - Chemistry
  - A literature search in pubmed.org retrieves **6906** papers related to decision trees!
- Good at dealing with symbolic feature

Decision Tree Representation



Classify instances by sorting them down the tree from the root to some leaf node

- Each **branch** corresponds to attribute value
- Each **internal node** has a splitting predicate
- Each **leaf node** assigns a classification

Internal Nodes

- Each internal node has an associated *splitting predicate*. Most common are binary predicates.
  Example predicates:
  - Age <= 20
  - Profession in {student, teacher}
  - 5000*Age + 3*Salary – 10000 > 0

Internal Nodes: Splitting Predicates

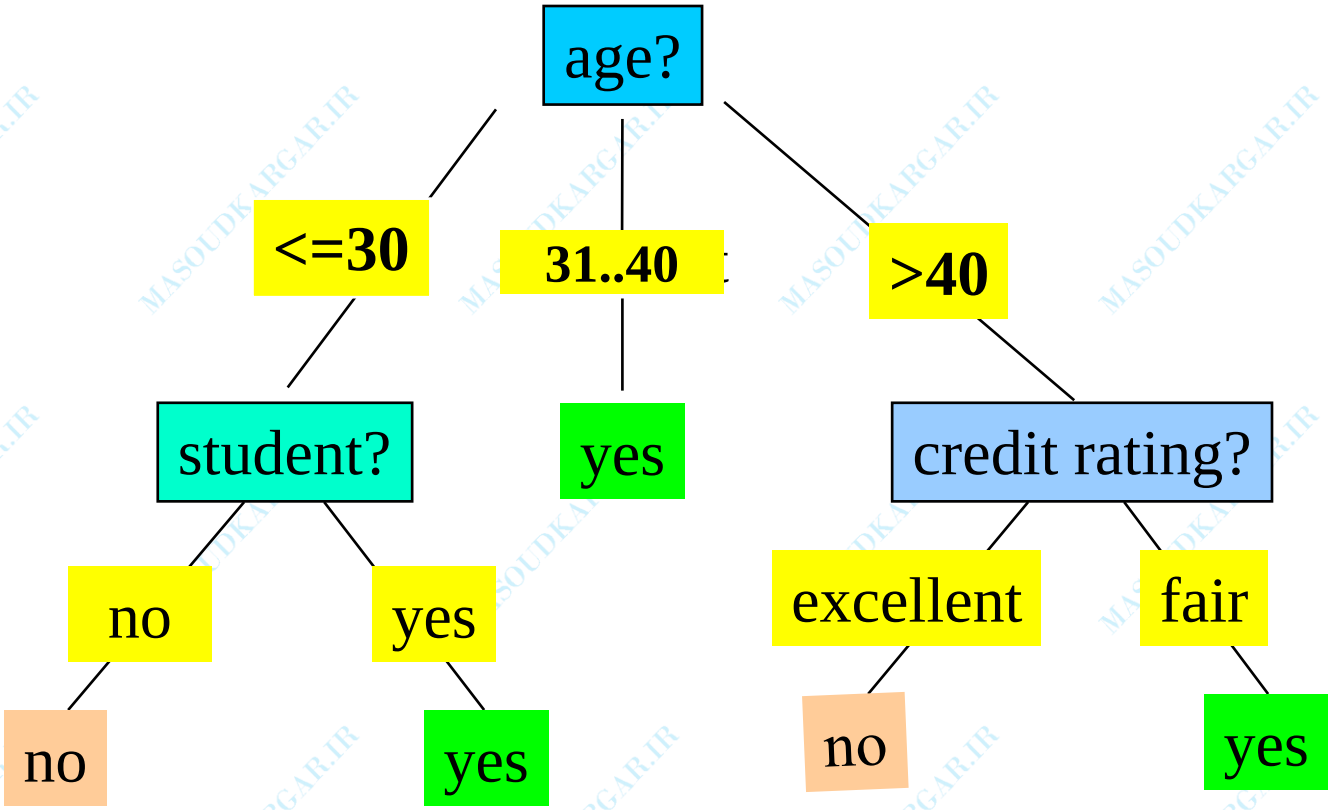
- Binary Univariate splits:
  - Numerical or ordered X: X <= c, c in dom(X)
  - Categorical X: X in A, A subset dom(X)
- Binary Multivariate splits:
  - Linear combination split on numerical variables:
    $\Sigma a_i X_i <= c$
- k-ary (k>2) splits analogous

Building Decision Tree Classifiers for DELL to Predict if a customer would buy a computer

- Training Data

age	income	student	credit_rating	buys_computer
<=30	high	no	fair	no
<=30	high	no	excellent	no
31…40	high	no	fair	yes
>40	medium	no	fair	yes
>40	low	yes	fair	yes
>40	low	yes	excellent	no
31…40	low	yes	excellent	yes
<=30	medium	no	fair	no
<=30	low	yes	fair	yes
>40	medium	yes	fair	yes
<=30	medium	yes	excellent	yes
31…40	medium	no	excellent	yes
31…40	high	yes	fair	yes
>40	medium	no	excellent	no

Output: A Decision Tree for “buys_computer”



Homogeneous: All labels of the instances are YES

Majority Voting 4/5

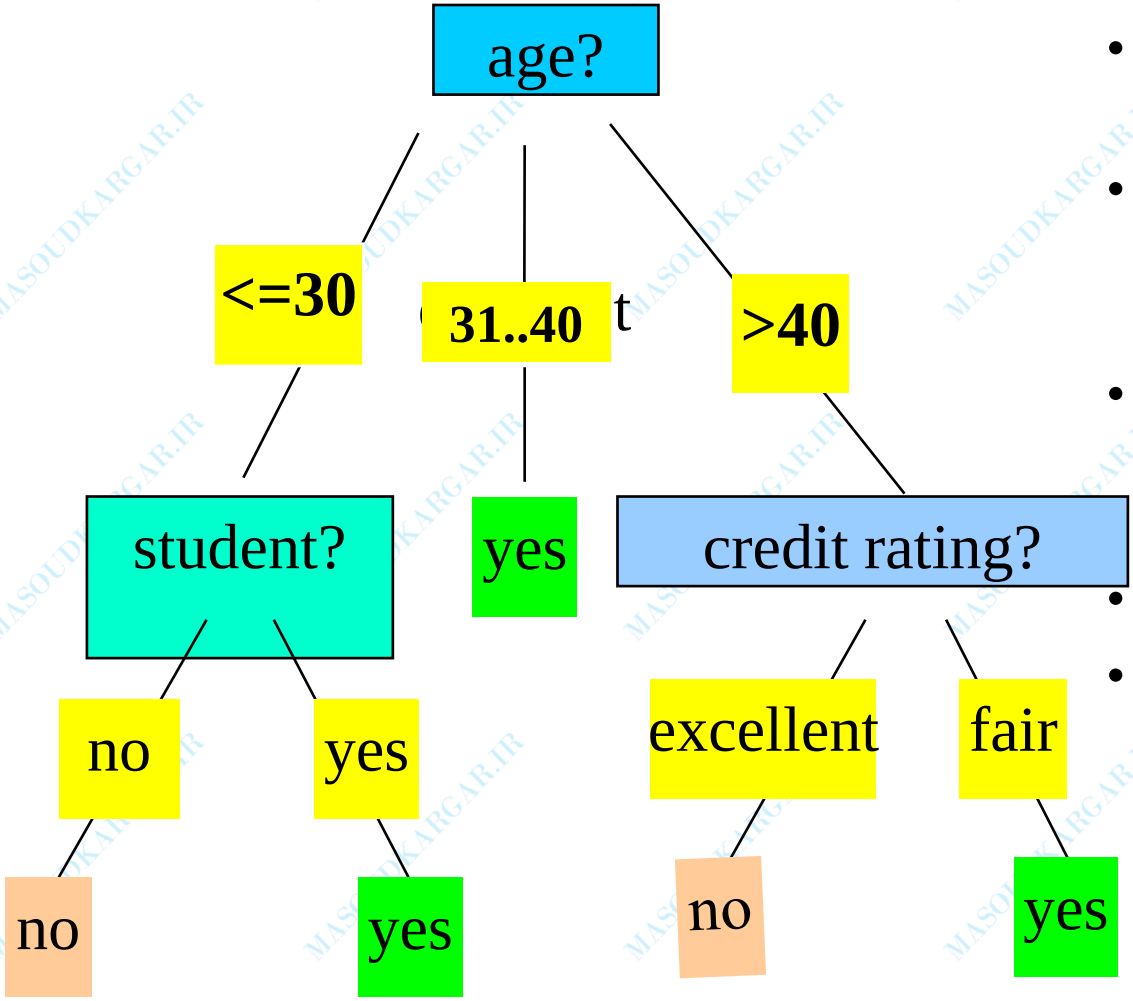
TDIDT Algorithm

- Also known as ID3 (Quinlan)
- To construct decision tree T from learning set S:
  - **If** all examples in S belong to some class C **Then** make leaf labeled C
  - **Otherwise**
    - select the “most informative” attribute A
    - partition S according to A’s values
    - recursively construct subtrees T1, T2, ..., for the subsets of S

Algorithm for Decision Tree Induction

- Basic algorithm (a greedy algorithm)
  - Tree is constructed in a top-down recursive divide-and-conquer manner
  - At start, all the training examples are at the root
  - Attributes are categorical (if continuous-valued, they are discretized in advance)
  - Examples are partitioned recursively based on selected attributes
  - Test attributes are selected on the basis of a heuristic or statistical measure (e.g., information gain)
- Conditions for **stopping partitioning**
  - All samples for a given node belong to the same class
  - There are no remaining attributes for further partitioning – majority voting is employed for classifying the leaf
  - There are no samples left

Design Issues of Decision Trees



- Which decision tree is the best?
- Which attributes to check first? How to split?
- How to decide the split values of real-value attributes (e.g. age)?
- When to stop splitting?
- How to evaluate decision trees?

Which Decision Tree is the Best?

- **Occam's razor**: (year 1320)
  - Prefer the simplest hypothesis that fits the data.
  - The principle states that the explanation of any phenomenon should make as few assumptions as possible, eliminating those that make no difference in the observable predictions of the explanatory hypothesis or theory
- **Albert Einstein**: Make everything as simple as possible, but not simpler.
- Why?
  - It's a philosophical problem.
  - Simple explanation/classifiers are more robust
  - Simple classifiers are more understandable

How To Build a Simple Decision Tree?: Attribute Selection

- **Intuitively** (as in the game):
  - We want to reduce the search space (ambiguity) ASAP by asking each question
- **Scientifically**:
  - Test attributes that **gain most information**.
- Remember: The splitting process of decision tree stops only when the labels of all instances in a node becomes pure (homogeneous) except other special cases.

How To Build a Simple Decision Tree

- Objective: Shorter trees are preferred over larger Trees
- Idea: want attributes that classifies examples well. The best attribute is selected.
  - Select attribute which partitions the learning set into subsets as “pure” as possible
- How well an attribute alone classifies the training data?

Information Theory

- Claude E. Shannon's classic paper "A Mathematical Theory of Communication" in the *Bell System Technical Journal* in July and October of 1948.
- Key Concepts:
  - **Entropy**, *H*, of a discrete random variable *X* is a measure of the amount of *uncertainty* associated with the value of *X*.
  - **Information (Gain)**: reduction of entropy (uncertainty)
  - **Mutual Information:** measures the amount of information that can be obtained about one random variable by observing another. (can be used for **feature selection**)

Measuring Entropy

- **Entropy**, *H*, of a discrete random variable *X* is a measure of the amount of *uncertainty* associated with the value of *X*.
  - In our case: the random variable is the class label of an instance (C)
  - For two training data with 10 instances at the root node
  - Data 1: 9 positive, 1 negative
  - Data 2: 5 positive, 5 negative
- For objects of Data 1 and 2, whose label is more uncertain?

$$H(X) = \mathbb{E}_X[I(x)] = -\sum_{x \in \mathbb{X}} p(x) \log p(x)$$

$$H_{\mathrm{b}}(p) = -p \log p - (1-p) \log(1-p).$$

Measuring Mutual Information

- measures the amount of information that can be obtained about one random variable by observing another.

$$I(X;Y) = \mathbb{E}_{X,Y}[SI(x,y)] = \sum_{x,y} p(x,y) \log \frac{p(x,y)}{p(x)\,p(y)}$$

  - Exercise: calculate the mutual information between the class label and an attribute
- Mutual information in feature selection:
  - Input Feature Selection by Mutual Information Based on Parzen Windows
  - Mutual information functions versus correlation functions. Journal of Statistical Physics, 2005

Information Gain (used in ID3)

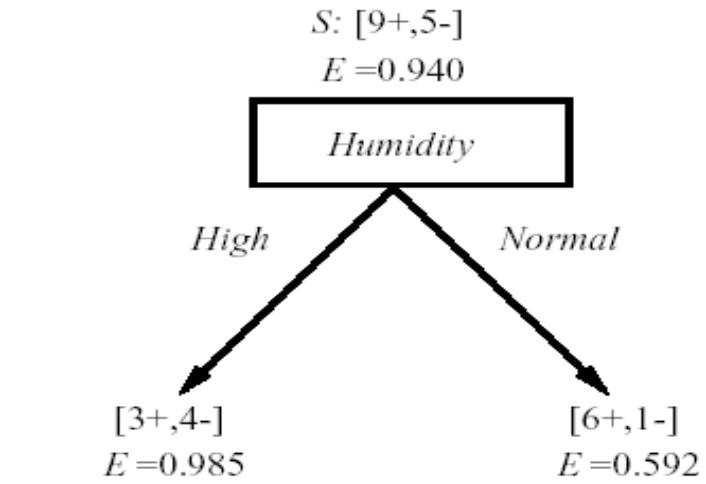
- What is the uncertainty removed by splitting on the value of A?
- The information gain of S relative to attribute A is the expected reduction in entropy caused by knowing the value of A
  - S_v: the set of examples in S where attribute A has value v

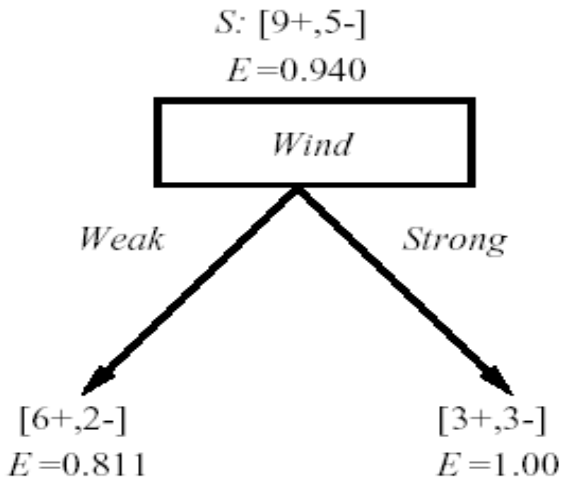
$$G(S,A) = E(S) - \sum_{v \in Values(A)} \frac{|S_v|}{|S|} E(S_v)$$

Play Tennis Example

Day	Outlook	Temperature	Humidity	Wind	PlayTennis
D1	Sunny	Hot	High	Weak	No
D2	Sunny	Hot	High	Strong	No
D3	Overcast	Hot	High	Weak	Yes
D4	Rain	Mild	High	Weak	Yes
D5	Rain	Cool	Normal	Weak	Yes
D6	Rain	Cool	Normal	Strong	No
D7	Overcast	Cool	Normal	Strong	Yes
D8	Sunny	Mild	High	Weak	No
D9	Sunny	Cool	Normal	Weak	Yes
D10	Rain	Mild	Normal	Weak	Yes
D11	Sunny	Mild	Normal	Strong	Yes
D12	Overcast	Mild	High	Strong	Yes
D13	Overcast	Hot	Normal	Weak	Yes
D14	Rain	Mild	High	Strong	No

Which attribute is the best classifier?





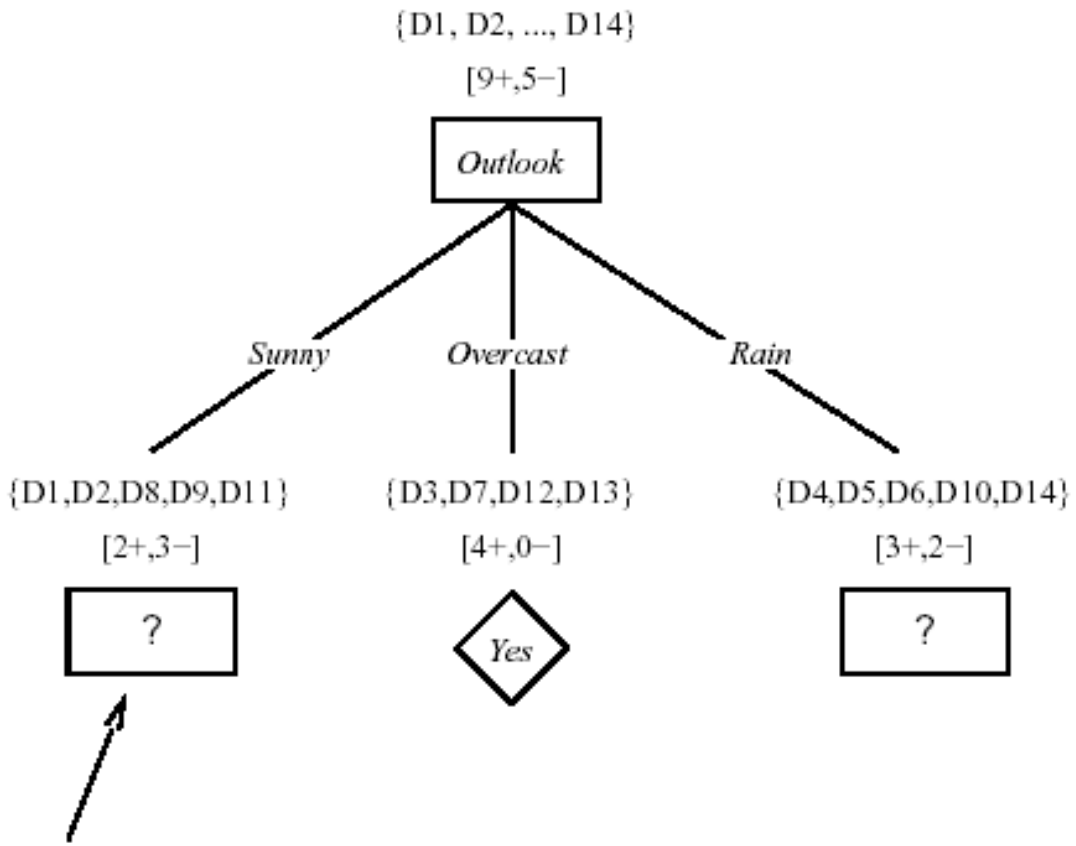
Gain (S, Humidity)
= .940 - (7/14).985 - (7/14).592
= .151

Gain (S, Wind)
= .940 - (8/14).811 - (6/14)1.0
= .048

$$Gain(S, Outlook) = 0.246 \quad Gain(S, Humidity) = 0.151$$

$$Gain(S, Wind) = 0.048 \quad Gain(S, Temperature) = 0.029$$

Which attribute is the best classifier?



Which attribute should be tested here?

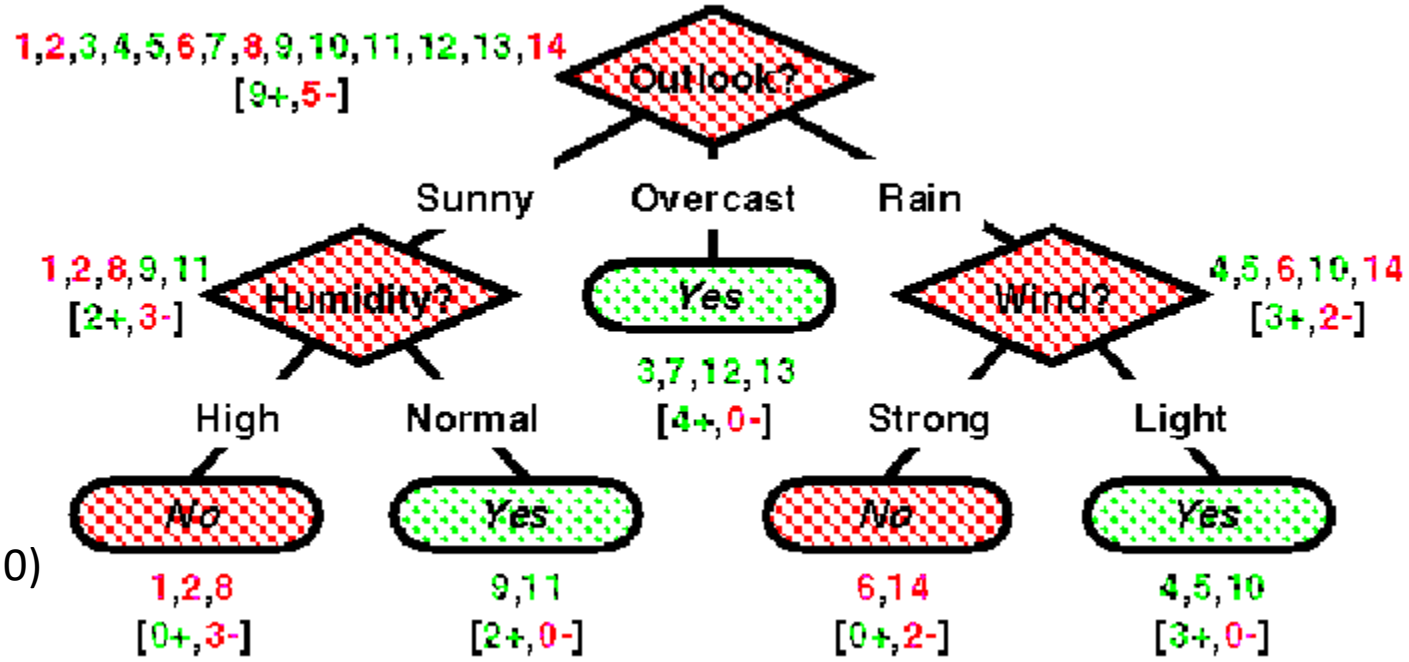
S_{sunny} = {D1,D2,D8,D9,D11}

Gain (S_{sunny}, *Humidity*) = .970 − (3/5) 0.0 − (2/5) 0.0 = .970

Gain (S_{sunny}, *Temperature*) = .970 − (2/5) 0.0 − (2/5) 1.0 − (1/5) 0.0 = .570

Gain (S_{sunny}, *Wind*) = .970 − (2/5) 1.0 − (3/5) .918 = .019

Which attribute is the best classifier?



A1 = overcast: + (4.0)
A1 = sunny:
| A3 = high: - (3.0)
| A3 = normal: + (2.0)
A1 = rain:
| A4 = weak: + (3.0)
| A4 = strong: - (2.0)

See/C 5.0

Gain Ratio (used in C4.5)

- Limitation of Information Gain
  - Biased towards attributes with many outcomes: not robust
  - Example: suppose an attribute A has 14 distinct values (product_id), splitting on A, will result maximum information gain. H(Di)=0.
- Gain Ratio: normalization applied to information gain
  - Normalized by the split information (penalize multiple-valued attributes/splits)
  - J.R. Quinlan (1986). Induction of Decision Trees, Machine Learning, (1), 81-106

$$GainRatio(A) = \frac{Gain(A)}{SplitInfo(A)}$$

$$SplitInfo(A) = -\sum_{j=1}^{v} \frac{|D_j|}{|D|} \times \log_2 \left(\frac{|D_j|}{|D|}\right)$$

Gini Index (CART, IBM Intelligent Miner)

- Another sensible measure of impurity (i and j are classes)
- a function the maximize when x=y? f=xy given x+y=1

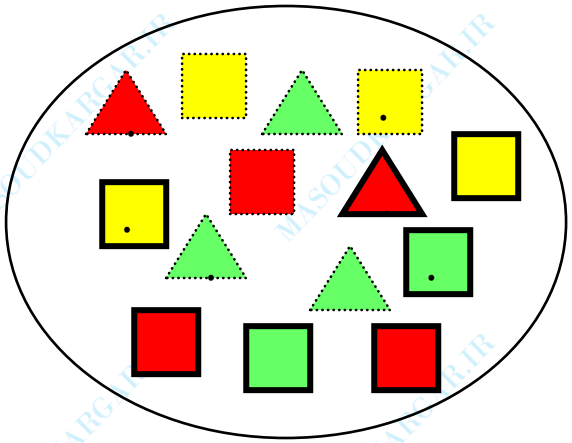
$$Gini = \sum_{i \neq j} p(i)p(j) \qquad or \; 1 - \sum_{i=1}^{v} p_i 2$$

- After applying attribute A, the resulting Gini index is

$$Gini(A) = \sum_{v} p(v) \sum_{i \neq j} p(i|v)p(j|v)$$

- Gini can be interpreted as expected error rate

Gini Index



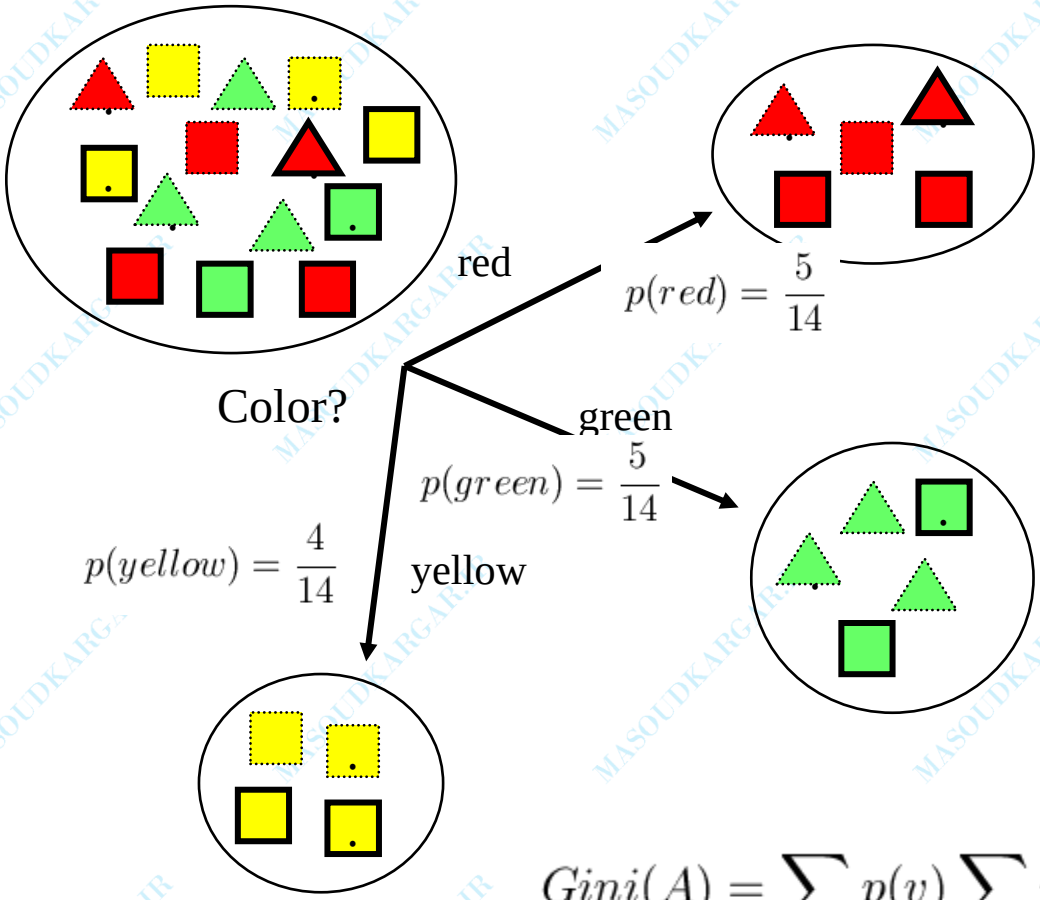
$$p(\square) = \frac{9}{14}$$

$$p(\triangle) = \frac{5}{14}$$

$$Gini = \sum_{i \neq j} p(i)p(j)$$

$$Gini = \frac{9}{14} \times \frac{5}{14} = 0.230$$

Gini Index for Color



$$Gini(A) = \sum_{v} p(v) \sum_{i \neq j} p(i|v)p(j|v)$$

$$Gini(\text{Color}) = \frac{5}{14} \times (\frac{3}{5} \times \frac{2}{5}) + \frac{5}{14} \times (\frac{2}{5} \times \frac{3}{5}) + \frac{4}{14} \times (\frac{4}{4} \times \frac{0}{4}) = 0.171$$

Gain of Gini Index

$$Gini = \frac{9}{14} \times \frac{5}{14} = 0.230$$

$$Gini(\text{Color}) = \frac{5}{14} \times (\frac{3}{5} \times \frac{2}{5}) + \frac{5}{14} \times (\frac{2}{5} \times \frac{3}{5}) + \frac{4}{14} \times (\frac{4}{4} \times \frac{0}{4}) = 0.171$$

$$GiniGain(\text{Color}) = 0.230 - 0.171 = 0.058$$

Comparing Attribute Selection Measures

- The three measures, in general, return good results but
  - Information gain:
    - biased towards multivalued attributes
  - Gain ratio:
    - tends to prefer unbalanced splits in which one partition is much smaller than the others
  - Gini index:
    - biased to multivalued attributes
    - has difficulty when # of classes is large
    - tends to favor tests that result in equal-sized partitions and purity in both partitions

Computing Information-Gain for Continuous-Value Attributes

- Let attribute A be a continuous-valued attribute
- Must determine the *best split point* for A
  - Sort the value A in increasing order
  - Typically, the midpoint between each pair of adjacent values is considered as a possible *split point*
    - $(a_i+a_{i+1})/2$ is the midpoint between the values of a_i and a_{i+1}
  - The point with the *minimum expected information requirement* for A is selected as the split-point for A
- Split:
  - D1 is the set of tuples in D satisfying A ≤ split-point, and D2 is the set of tuples in D satisfying A > split-point

Other Attribute Selection Measures

- CHAID: a popular decision tree algorithm, measure based on χ^2 test for independence
- C-SEP: performs better than info. gain and gini index in certain cases
- G-statistics: has a close approximation to χ^2 distribution
- MDL (Minimal Description Length) principle (i.e., the simplest solution is preferred):
  - The best tree as the one that requires the fewest # of bits to both (1) encode the tree, and (2) encode the exceptions to the tree
- Multivariate splits (partition based on multiple variable combinations)
  - CART: finds multivariate splits based on a linear comb. of attrs.
- Which attribute selection measure is the best?
  - Most give good results, none is significantly superior than others

Random Forest

- **random forest** is a classifier that consists of many decision trees and outputs the class that is the mode of the classes output by individual trees
  - For many data sets, it produces a highly accurate classifier.
  - It handles a very large number of input variables.
  - It estimates the importance of variables in determining classification.
  - It generates an internal unbiased estimate of the generalization error as the forest building progresses.
  - It includes a good method for estimating missing data and maintains accuracy when a large proportion of the data are missing.
  - It provides an experimental way to detect variable interactions.
  - It can balance error in class population unbalanced data sets.
  - It computes proximities between cases, useful for clustering, detecting outliers, and (by scaling) visualizing the data.
  - Using the above, it can be extended to unlabeled data, leading to unsupervised clustering, outlier detection and data views.
  - Learning is fast.

Issues in Decision Tree

- Overfiting
- Tree Pruning
- Cross-validation
- Model Evaluation
- Advanced Decision Tree
- C4.5 Software Package

Summary: Advantages of Decision Trees

- **Simple to understand and interpret.** People are able to understand decision tree models after a brief explanation.
- **Have value even with little hard data.** Important insights can be generated based on experts describing a situation (its alternatives, probabilities, and costs) and their preferences for outcomes.
- **Use a white box model.** If a given result is provided by a model, the explanation for the result is easily replicated by simple math.
- **Can be combined with other decision techniques.**

قدردانی

- Dr. Jianjun Hu
  http://mleg.cse.sc.edu/edu/csce822/
- University of South Carolina
- Department of Computer Science and Engineering