University of Colorado Boulder

CSCI 4502/5502

Data Mining

Fall 2019
Lecture 11 (Oct 1)

Reminders

- Due at 9:30am, Thursday, Oct 3
  - course project announcement — post at project discussion forum
  - proposal summary slides — submit at Moodle
- Due at 9:30am, Thursday, Oct 10
  - course project proposal — submit at Moodle

Course Project Presentations

- Moodle survey on team availability in Week 7
  - Tu Th 9:30am to 10:45am, HUMN 1B80
  - Other in-person meetings, ECCR 1B10/11
  - Zoom meetings
  - Choose ALL time slots that work for you
  - Each team should fill out the survey by 9:30am, Thursday, Oct 3

Course Project Proposal

- Due at 9:30am, Thursday, Oct 10
- Course project proposal (~3 pages)
  - ACM SIG Proceedings Templates
  - title, team (name, course section)
  - motivation, literature survey
  - proposed work (data, subtasks)
  - how to evaluate, milestones
  - brief summary of project discussion

More on Course Projects

- Get your data NOW and make a BACKUP
- List of subtasks: no right or wrong questions, just interesting questions, good reasoning
- Possible components: data collection, preprocessing, management, initial analysis, actual design, evaluation, visualization
- Finish one pipeline, then augment/expand
- Coordination among team members

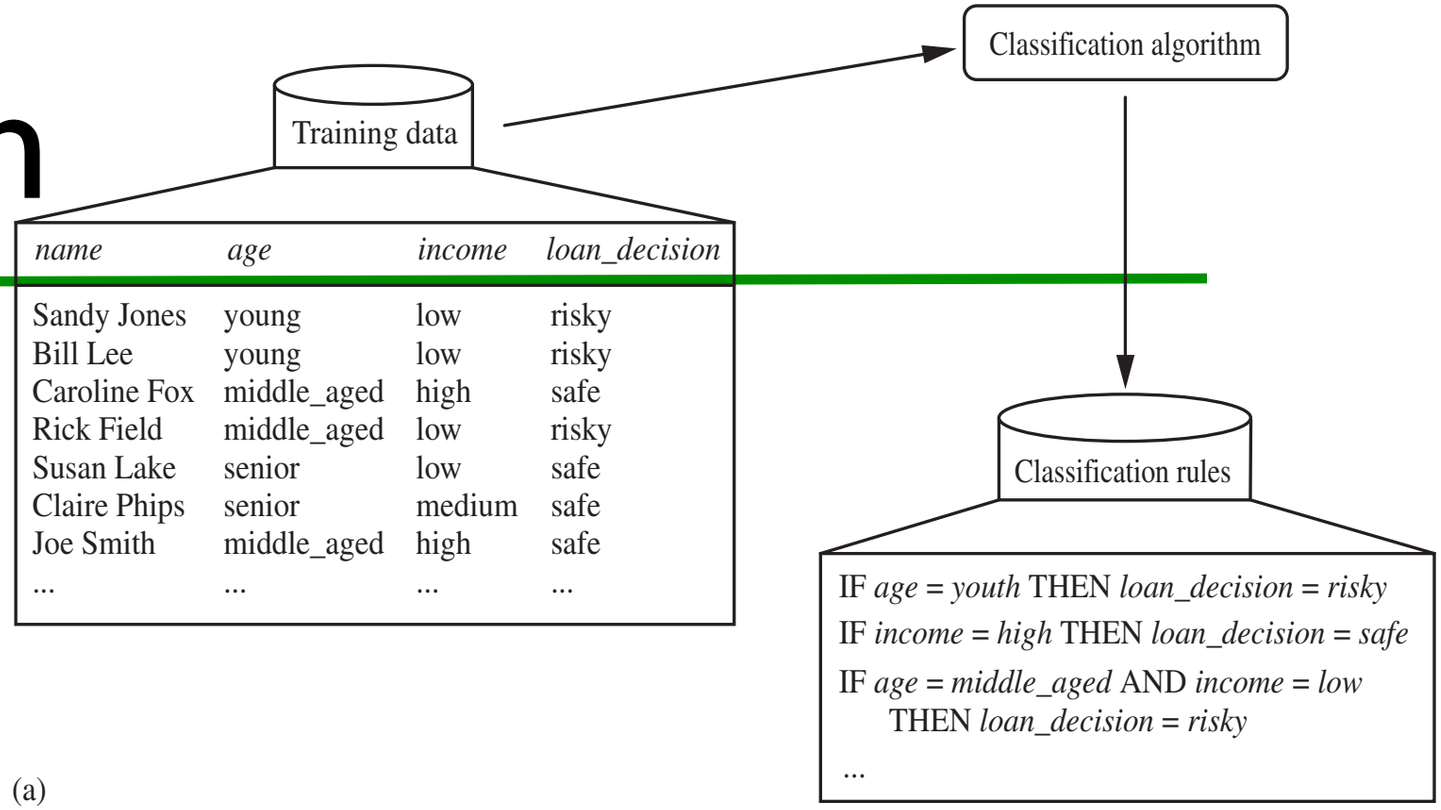
Review

- Chapter 6 & 7: Frequent Pattern Mining
  - itemset, subsequence, substructure
  - frequent itemsets: Apriori, FP-growth
  - association rules: multi-level, multi-dimension, quantitative
  - correlation analysis
  - constraint-based mining

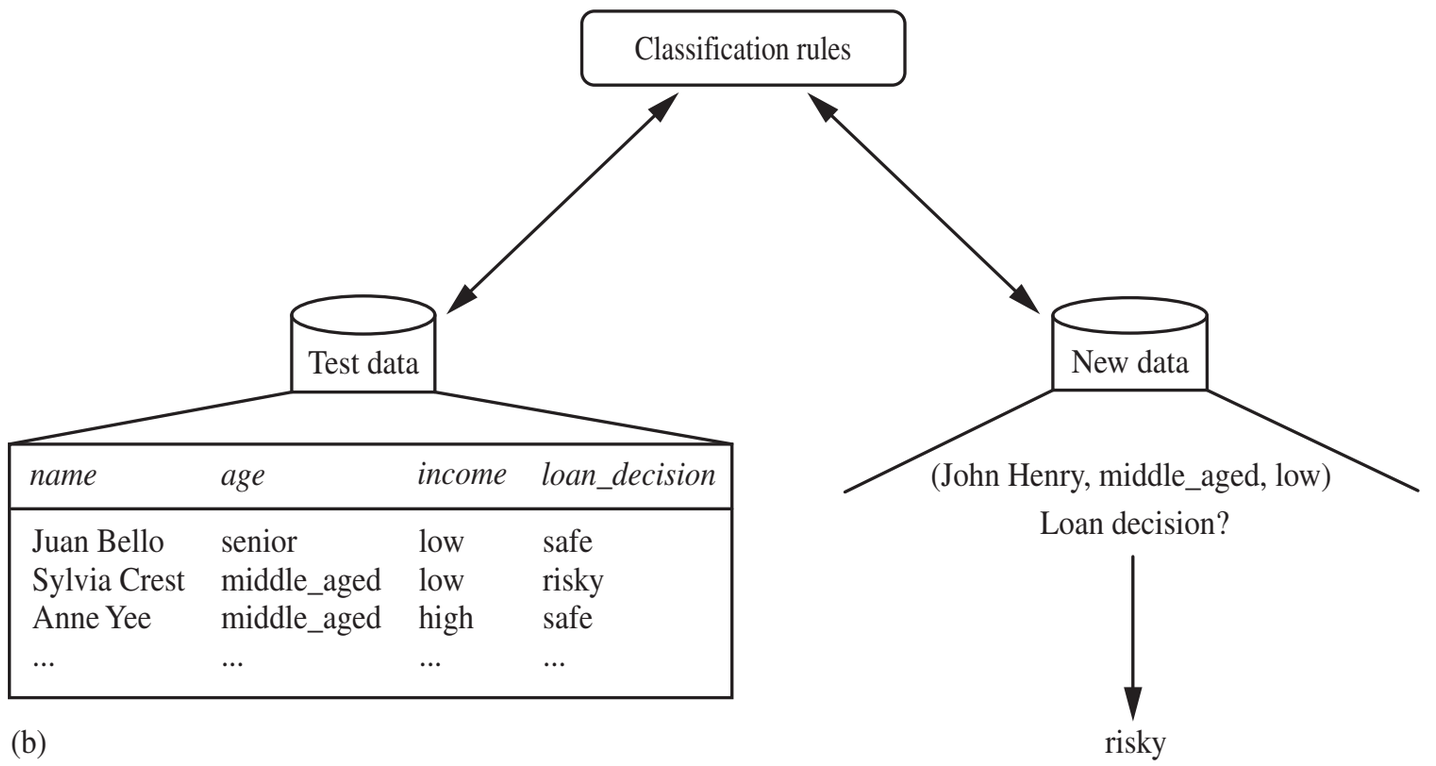
Chapter 8
Classification: Basic Concepts

Classification

- Step 1: Learning
  - model construction
  - training set
  - class labels
- Step 2: Classification
  - test set
  - accuracy



(a)

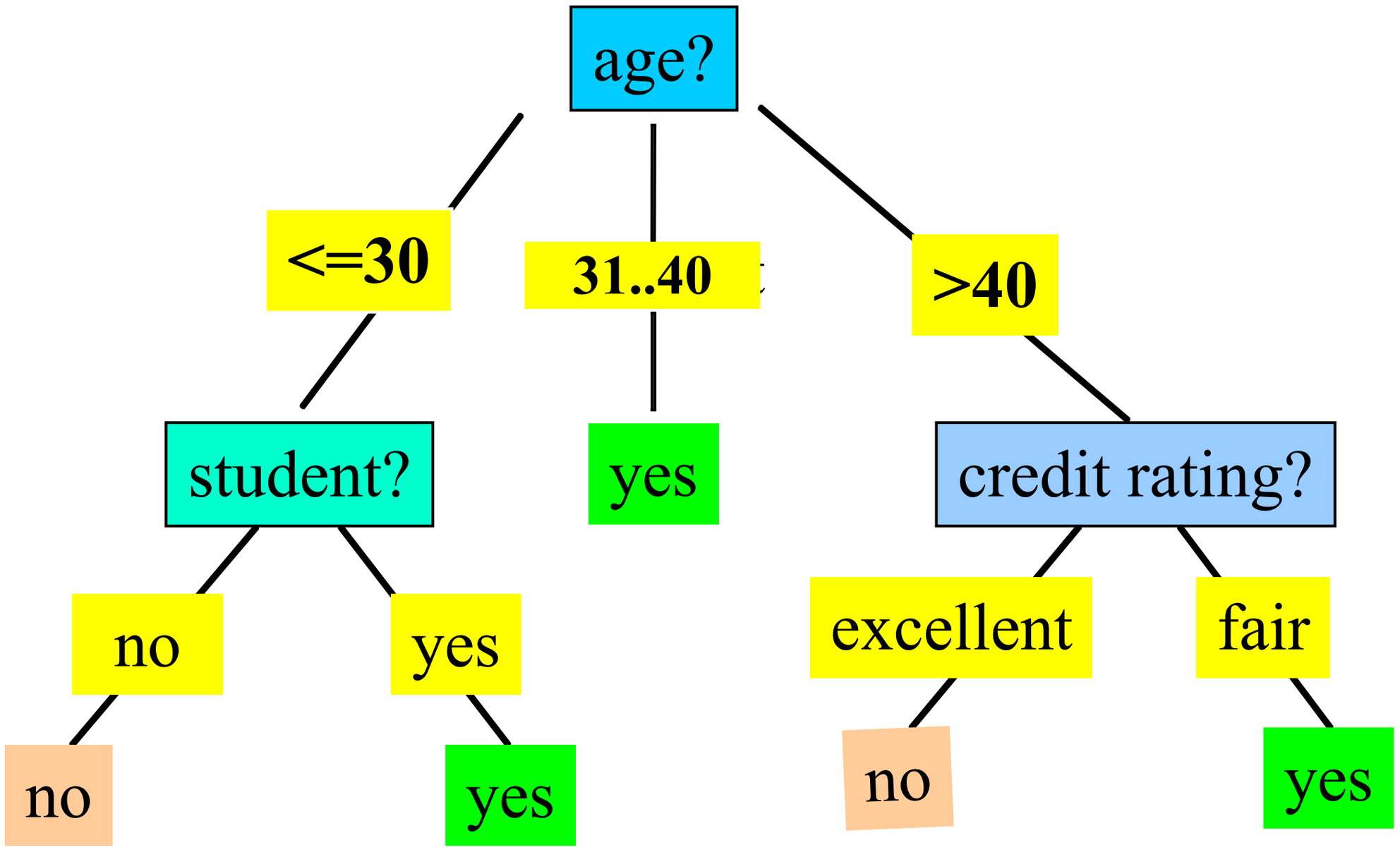


(b)

Chap 8: Classification

- Basic concepts
- Decision tree induction
- Bayesian classification
- Rule-based classification
- Model evaluation and selection
- Improve classification accuracy
- Summary

Example: Decision Tree



Attribute Selection Measures

- Information gain (ID3/C4.5)
  - D, m classes C_i $p_i = |C_{i,D}|/|D|$
  - expected information (entropy) needed to classify D $Info(D) = -\sum_{i=1}^{m} p_i log_2(p_i)$
  - information needed to classify D using A
    - attribute A: a_1, a_2, ..., a_v

$$Info_A(D) = \sum_{j=1}^{v} \frac{|D_j|}{|D|} \times Info(D_j)$$

  - information gain $Gain(A) = Info(D) - Info_A(D)$

Information Gain Example

- Two classes
  - buy: 9
  - not buy: 5
  - total: 14

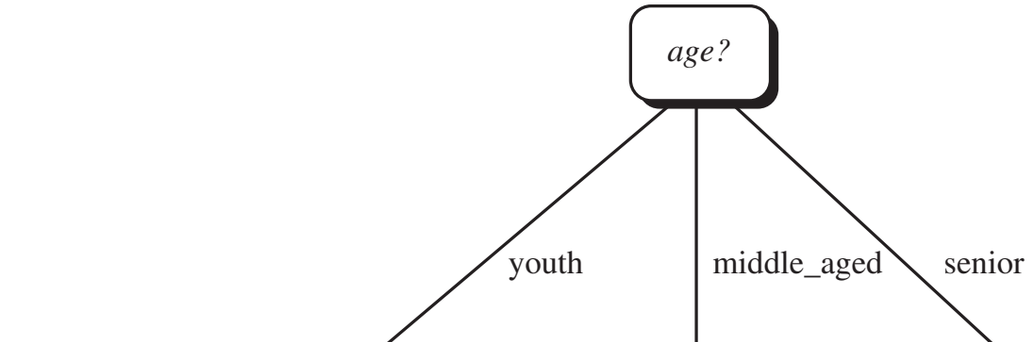
CID	age	income	student	credit_rating	buys_computer
1	<= 30	high	no	fair	no
2	<= 30	high	no	excellent	no
3	31-40	high	no	fair	yes
4	>40	medium	no	fair	yes
5	>40	low	yes	fair	yes
6	>40	low	yes	excellent	no
7	31-40	low	yes	excellent	yes
8	<= 30	medium	no	fair	no
9	<= 30	low	yes	fair	yes
10	>40	medium	yes	fair	yes
11	<= 30	medium	yes	excellent	yes
12	31-40	medium	no	excellent	yes
13	31-40	high	yes	fair	yes
14	>40	medium	no	excellent	no

$$Info(D) = -\sum_{i=1}^{m} p_i log_2(p_i)$$

$$Info(D) = I(9,5) = -\frac{9}{14} log_2(\frac{9}{14}) - \frac{5}{14} log_2(\frac{5}{14}) = 0.940$$

Information Gain Example

CID	age	income	student	credit_rating	buys_computer
1	<= 30	high	no	fair	no
2	<= 30	high	no	excellent	no
3	31-40	high	no	fair	yes
4	>40	medium	no	fair	yes
5	>40	low	yes	fair	yes
6	>40	low	yes	excellent	no
7	31-40	low	yes	excellent	yes
8	<= 30	medium	no	fair	no
9	<= 30	low	yes	fair	yes
10	>40	medium	yes	fair	yes
11	<= 30	medium	yes	excellent	yes
12	31-40	medium	no	excellent	yes
13	31-40	high	yes	fair	yes
14	>40	medium	no	excellent	no



income	*student*	*credit_rating*	*class*
high	no	fair	no
high	no	excellent	no
medium	no	fair	no
low	yes	fair	yes
medium	yes	excellent	yes

income	*student*	*credit_rating*	*class*
medium	no	fair	yes
low	yes	fair	yes
low	yes	excellent	no
medium	yes	fair	yes
medium	no	excellent	no

income	*student*	*credit_rating*	*class*
high	no	fair	yes
low	yes	excellent	yes
medium	no	excellent	yes
high	yes	fair	yes

Information Gain Example

CID	age	income	student	credit_rating	buys_computer
1	<= 30	high	no	fair	no
2	<= 30	high	no	excellent	no
3	31-40	high	no	fair	yes
4	>40	medium	no	fair	yes
5	>40	low	yes	fair	yes
6	>40	low	yes	excellent	no
7	31-40	low	yes	excellent	yes
8	<= 30	medium	no	fair	no
9	<= 30	low	yes	fair	yes
10	>40	medium	yes	fair	yes
11	<= 30	medium	yes	excellent	yes
12	31-40	medium	no	excellent	yes
13	31-40	high	yes	fair	yes
14	>40	medium	no	excellent	no

age	p	n	I(p, n)
<=30	2	3	0.971
31-40	4	0	0
>40	3	2	0.971

Gain(age) = 0.246
Gain(income) = 0.029
Gain(student) = 0.151
Gain(credit_rating) = 0.048

$$Info(D) = I(9,5) = -\frac{9}{14}log_2(\frac{9}{14}) - \frac{5}{14}log_2(\frac{5}{14}) = 0.940$$

$$Info_{age}(D) = \frac{5}{14}I(2,3) + \frac{4}{14}I(4,0) + \frac{5}{14}I(3,2) = 0.694$$

Information Gain

- Continuous-valued attribute A
- Determine the best split point for A
  - sort A values in increasing order
  - consider the midpoint of adjacent values
    - $(a_i + a_{i+1}) / 2$
  - pick the midpoint w/ minimum $Info_A(D)$
- Split
  - D1: A <= split point; D2: A > split point

Gain Ratio (C4.5)

- Information gain measure biased towards attributes with a large number of values
  - e.g., customerID, productID
- C4.5 (a successor of ID3)
- select attribute with maximum gain ratio

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

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

Gini Index (CART)

- Gini index

$$Gini(D) = 1 - \sum_{i=1}^{m} p_i^2$$

- Binary split using attribute A

$$Gini_A(D) = \frac{|D_1|}{|D|} Gini(D_1) + \frac{|D_2|}{|D|} Gini(D_2)$$

- Reduction in impurity

$$\Delta Gini(A) = Gini(D) - Gini_A(D)$$

- Select attribute with largest impurity reduction

Attribute Selection Measures

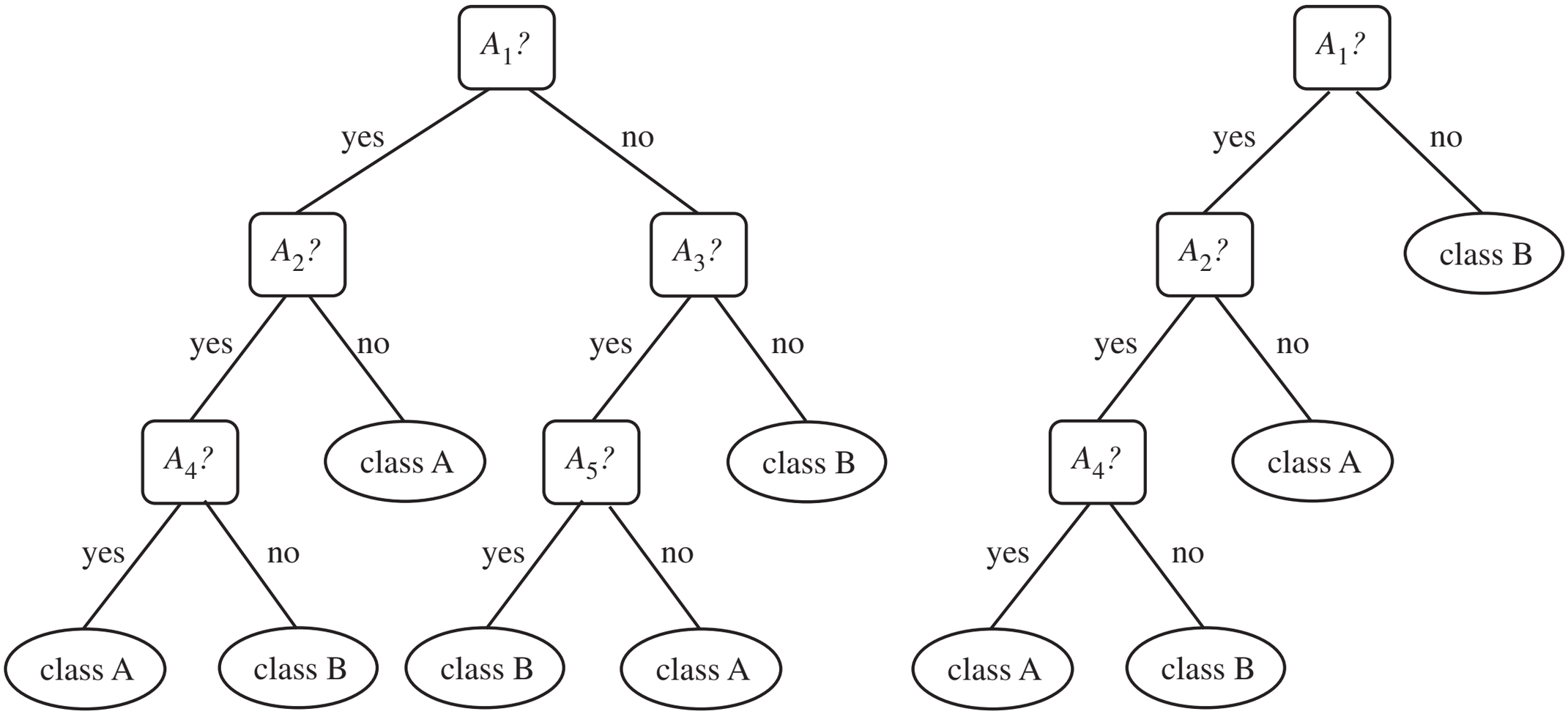
- Comparison of the three measures
  - good results in general but some biases
  - information gain: multi-valued attributes
  - gain ratio: unbalanced splits
  - gini index: multi-valued, equal-sized & pure partitions, not good when number of classes is large

Overfitting & Tree Pruning

- Overfitting of the training data
  - too many branches, reflect anomalies due to noise or outliers
  - poor accuracy for unseen data
- Tree pruning to avoid overfitting
  - preprunning: halt tree construction early
  - postpruning: remove branches from a "fully-grown" tree

Tree Pruning Example

- Replace a subtree w/ most freq. class label



Chapter 8: Classification

- Basic concepts
- Decision tree induction
- Bayesian classification
- Rule-based classification
- Model evaluation and selection
- Improve classification accuracy
- Summary

Bayesian Classification

- A statistical classifier
  - predicts class membership probabilities
- Foundation
  - based on Bayes' Theorem
- Performance (naïve Bayesian classifier)
  - comparable to decision tree & some neural network classifiers
- Incremental

Bayes' Theorem

- X: a data sample (evidence), class unknown
  - e.g., X: age = 35, income = $40,000
- H: a hypothesis that X belongs to class C
  - e.g., H: buys a computer
- Classification: determine P(H|X)
- P(H), P(X): prior probability
- P(X|H), P(H|X): posterior probability
- Bayes' Theorem $$P(H|X) = \frac{P(X|H)P(H)}{P(X)}$$

Naïve Bayesian Classifier (1)

- X = (x_1, x_2, ..., x_n) (i.e., n attributes)
- m classes: C_1, C_2, ..., C_m
- Classification: maximal $P(C_i|X)$
- Based on Bayes' Theorm

$$P(C_i|X) = \frac{P(X|C_i)P(C_i)}{P(X)}$$

- Since P(X) is constant for all classes, only need to maximize $P(X|C_i)P(C_i)$

Naïve Bayesian Classifier (2)

- Naïve assumption: class conditional independence (no dependence between attributes)

$$P(X|C_i) = \prod_{k=1}^{n} P(x_k|C_i) = P(x_1|C_i) \times P(x_2|C_i) \times \cdots \times P(x_n|C_i)$$

- If A_k is categorical, $P(x_k|C_i)$
- If A_k is continuous-valued, assume Gaussian distribution, $P(x_k|C_i) = g(x_k, \mu_{C_i}, \sigma_{C_i})$

$$g(x, \mu, \sigma) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x-\mu)^2)}{2\sigma^2}}$$

Naïve Bayesian Classifier Example

- 2 classes: buys_computer
  - C_1: yes
  - C_2: no
- X = (age<=30, income=medium, student=yes, credit_rating = fair)

CID	age	income	student	credit_rating	buys_computer
1	<= 30	high	no	fair	no
2	<= 30	high	no	excellent	no
3	31-40	high	no	fair	yes
4	>40	medium	no	fair	yes
5	>40	low	yes	fair	yes
6	>40	low	yes	excellent	no
7	31-40	low	yes	excellent	yes
8	<= 30	medium	no	fair	no
9	<= 30	low	yes	fair	yes
10	>40	medium	yes	fair	yes
11	<= 30	medium	yes	excellent	yes
12	31-40	medium	no	excellent	yes
13	31-40	high	yes	fair	yes
14	>40	medium	no	excellent	no

Naïve Bayesian Classifier Example

- buys_computer: C_1 = yes, C_2 = no
- X = (age <= 30, income = medium, student = yes, credit_rating = fair)
- $P(C_i|X) = P(X|C_i)\ P(C_i) / P(X)$

$P(X|C_i) = P(age<=30 \mid C_i)$

$* P(income=medium \mid C_i)$

$* P(student=yes \mid C_i)$

$* P(credit\_rating=fair \mid C_i)$

Naïve Bayesian Classifier Example

- P(buys_computer = “yes”) = 9/14 = 0.643
  P(buys_computer = “no”) = 5/14= 0.357

- P(age = “<=30” | buys_computer = “yes”) = 2/9 = 0.222
  P(age = “<= 30” | buys_computer = “no”) = 3/5 = 0.6
  P(income = “medium” | buys_computer = “yes”) = 4/9 = 0.444
  P(income = “medium” | buys_computer = “no”) = 2/5 = 0.4
  P(student = “yes” | buys_computer = “yes) = 6/9 = 0.667
  P(student = “yes” | buys_computer = “no”) = 1/5 = 0.2
  P(credit_rating = “fair” | buys_computer = “yes”) = 6/9 = 0.667
  P(credit_rating = “fair” | buys_computer = “no”) = 2/5 = 0.4

- P(X|buys_computer = “yes”) = 0.222 x 0.444 x 0.667 x 0.667 = 0.044
  P(X|buys_computer = “no”) = 0.6 x 0.4 x 0.2 x 0.4 = 0.019

- P(X|buys_computer = “yes”) * P(buys_computer = “yes”) = 0.028
  P(X|buys_computer = “no”) * P(buys_computer = “no”) = 0.007

Avoid 0-Probability

- The 0-probability problem

$$P(X|C_i) = \prod_{k=1}^{n} P(x_k|C_i) = P(x_1|C_i) \times P(x_2|C_i) \times \cdots \times P(x_n|C_i)$$

  - e.g., income: low(0), medium(990), high(10)
- Laplacian correction (or Laplace estimator)
  - add 1 to each case
  - income: low(1), medium(991), high(11)
  - non-zero, close to original probabilities

Naïve Bayesian Classifier

- Advantage
  - easy to compute
  - good results in most cases
- Disadvantage
  - assumption: class conditional independence
  - dependencies exist in practice
    - e.g., hospital patients: age, family history, fever, cough, lunch cancer, diabetes, etc.