University of Colorado
Boulder

# CSCI 4502/5502
# Data Mining

Fall 2020
Lecture 12 (Oct 8)

# Reminder/Announcement

- Course project announcement, slides, report
  - Project proposal report due at 9:30am, Thursday, Oct 8
- Homework 4
  - posted in Canvas, due at 9:30am, Thursday, Oct 15
- Midterm Exam: Thursday, Oct 29, more details later

# Course Project Proposal

- Interesting project ideas, wide variety
- Subtasks, data mining pipeline
  - data collection, understand your data, preprocessing, management, mining, evaluation, visualization
- Analytical thinking: what? why? how?
  - potential applications, subset selection, spatial-temporal, global/local trends and anomalies, methods, results, efficiency, challenges, lessons

# Course Project Schedule

- Course project proposal: Week 6 & 7
- Course project checkpoint: Week 12 (11/10, 11/12)
- Course project final report: Week 16 (12/8, 12/10)
- Team work: Individual contributions, communications
- Continue working on your project. Have fun!

# Review: Chapter 6 & 7

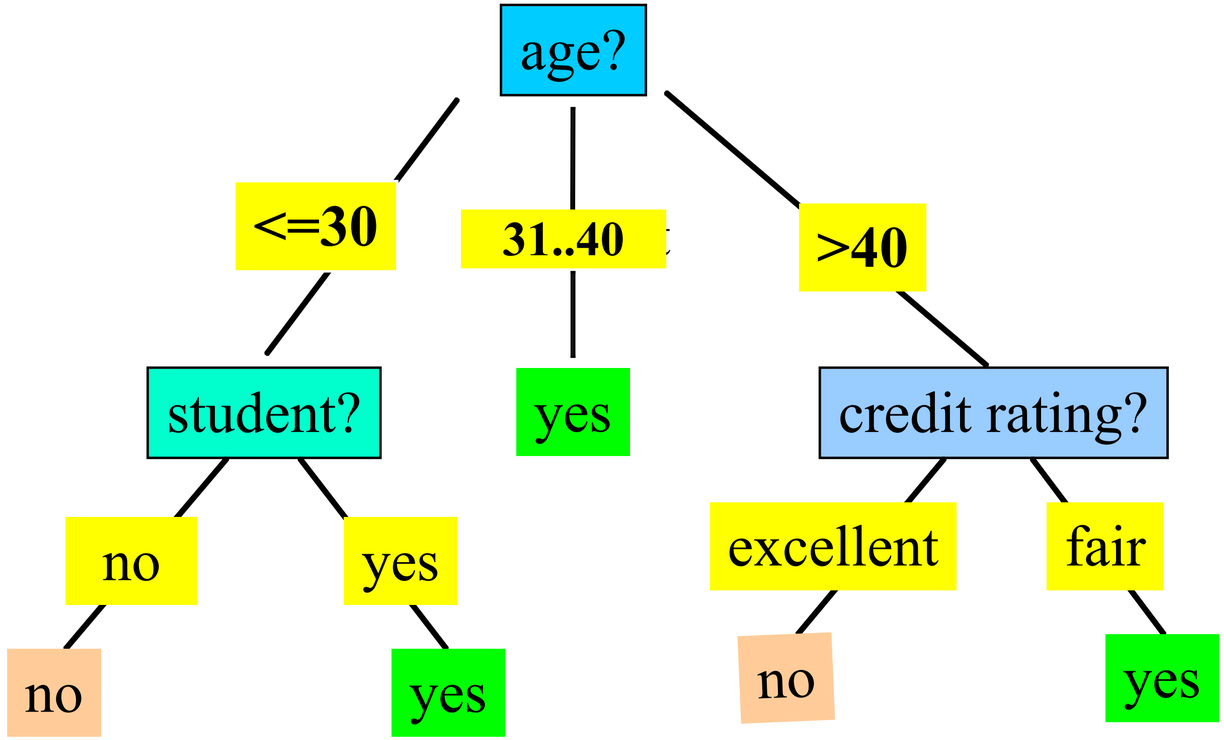
- Frequent patterns: itemset, subsequence, substruture
- Mining frequent itemsets: Apriori, FP-growth
- Mining association rules, correlation analysis
  - support, confidence, correlation, multi-level/dimension
- Constraint-based mining
  - Metarule, anti-monotonic, monotonic, succinct, convertible

# Review: Chapter 8: Classification

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

# Example: Training Set & Decision Tree

| CID | age | income | student | credit_ratin | buys_compute |
|---|---|---|---|---|---|
| 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 |



# Decision Tree Induction

- Basic algorithm (a greedy algorithm)
  - top-down, recursive, divide-and-conquer
  - attribute selection
  - attribute split
- Stopping conditions
  - all samples belong to the same class
  - no remaining attributes: majority voting
  - no samples left

# Attribute Selection Measures

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

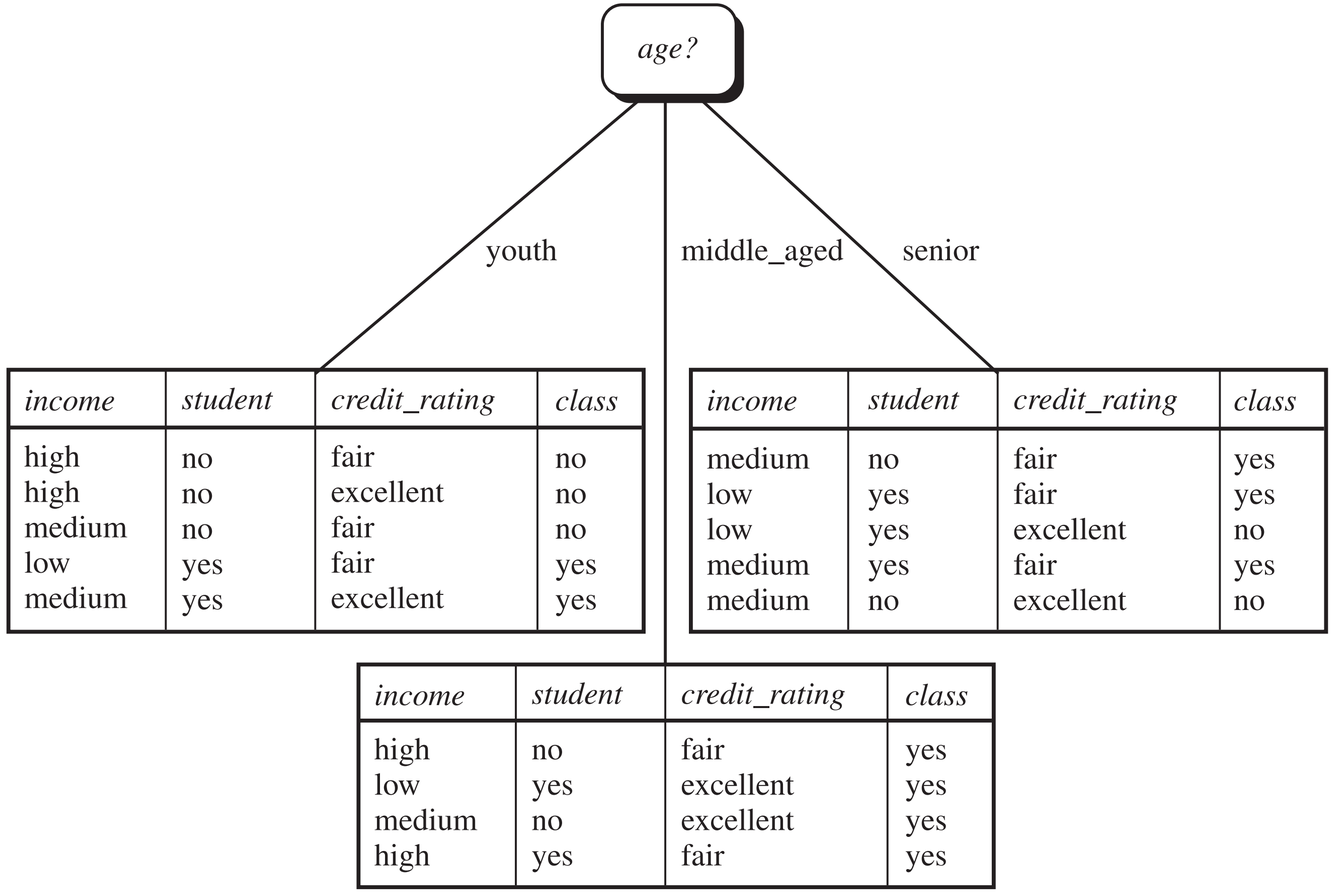
| CID | age | income | student | credit_rating | buys_computer |
|---|---|---|---|---|---|
| 1 | <= 30 | high | no | fair | no |
| 2 | <= 30 | high | no | excellent | no |
| 3 | 31-40 | high | no | fair | yes |
| 4 | >40 | medium | no | fair | yes |
| 5 | >40 | low | yes | fair | yes |
| 6 | >40 | low | 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 |



# 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) $SplitInfo_A(D) = -\sum_{j=1}^{v} \frac{|D_j|}{|D|} \times log_2\left(\frac{|D_j|}{|D|}\right)$
  - select attribute with maximum gain ratio

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

# 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

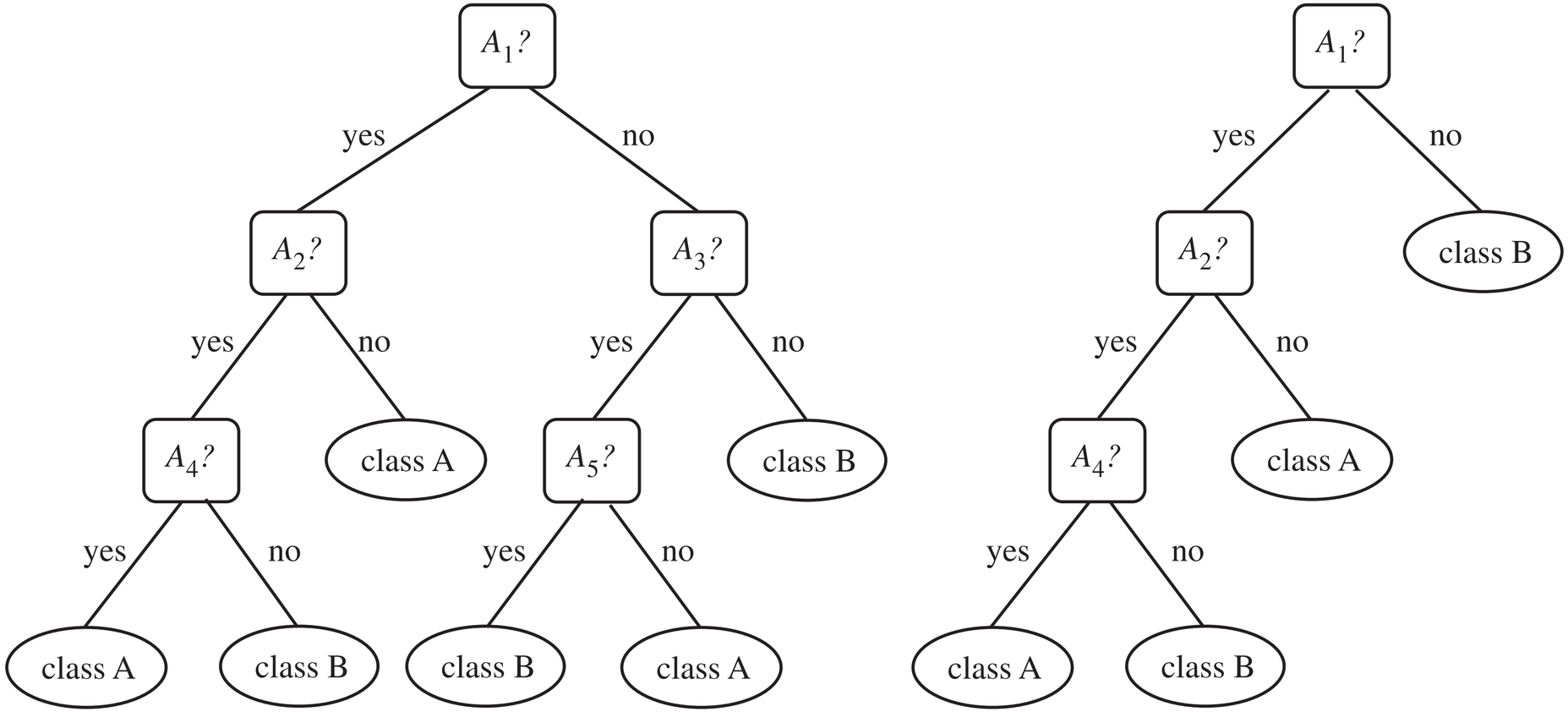
- Comparison of the three measures
  - good results in general but some biases
  - information gain: multi-valued attributes
  - gain ratio: unbalanced splits
  - gini index: multi-valued, equal-sized & pure partitions, not good when number of classes is large

# 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

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

- 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

# 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

$$P(x_k|C_i) = g(x_k, \mu_{C_i}, \sigma_{C_i}) \qquad g(x, \mu, \sigma) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x-\mu)^2)}{2\sigma^2}}$$

# 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 |