Classification fundamentals

Data Base and Data Mining Group of Politecnico di Torino

Elena Baralis
Politecnico di Torino

Classification

- Objectives
  - prediction of a class label
  - definition of an interpretable model of a given phenomenon

2
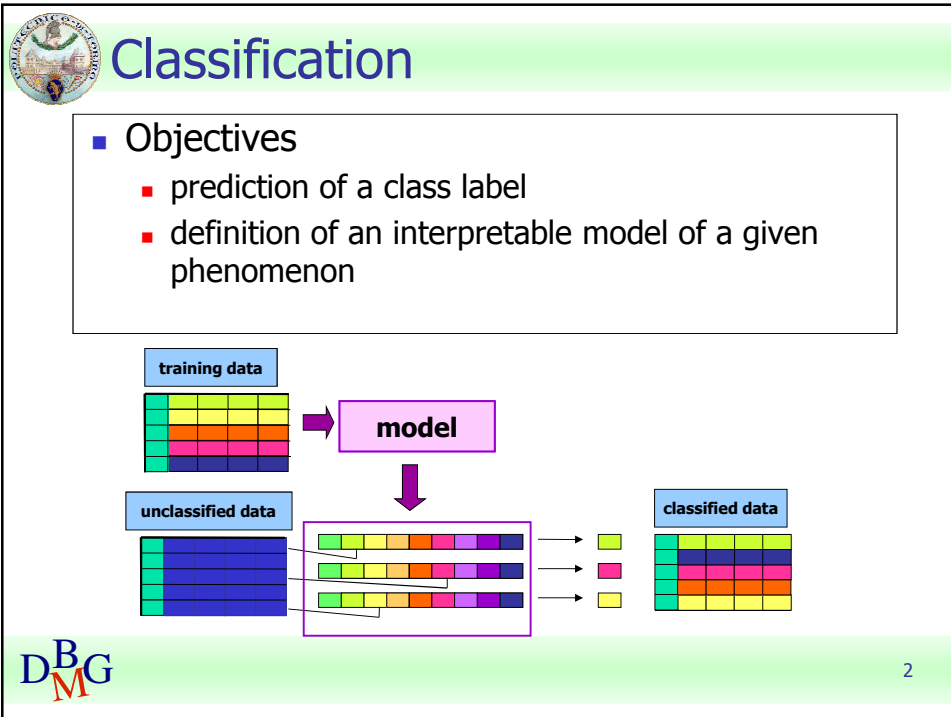
Classification

- Approaches
  - decision trees
  - bayesian classification
  - classification rules
  - neural networks
  - k-nearest neighbours
  - SVM

training data

model

unclassified data

classified data

3

Classification

- Requirements
  - accuracy
  - interpretability
  - scalability
  - noise and outlier management

training data

model

unclassified data

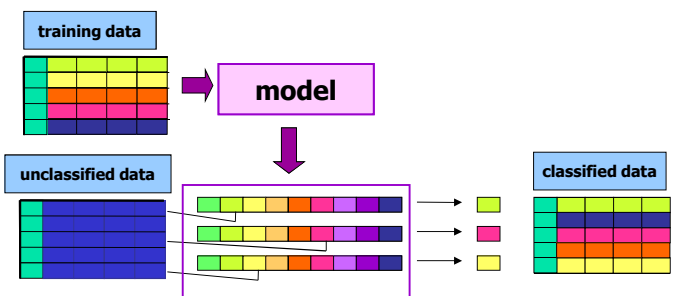classified data

4

Elena Baralis
Politecnico di Torino

Classification

- Applications
  - detection of customer propension to leave a company (churn or attrition)
  - fraud detection
  - classification of different pathology types
  - ...

training data

model

unclassified data

classified data

5

Classification: definition

- Given
  - a collection of class labels
  - a collection of data objects labelled with a class label
- Find a descriptive profile of each class, which will allow the assignment of unlabeled objects to the appropriate class

6

Elena Baralis
Politecnico di Torino

Definitions

- Training set
  - Collection of labeled data objects used to learn the classification model
- Test set
  - Collection of labeled data objects used to validate the classification model

DBMG
7

Classification techniques

- Decision trees
- Classification rules
- Association rules
- Neural Networks
- Naïve Bayes and Bayesian Networks
- k-Nearest Neighbours (k-NN)
- Support Vector Machines (SVM)
- ...

DBMG
8

Elena Baralis
Politecnico di Torino

Evaluation of classification techniques

- Accuracy
  - quality of the prediction
- Efficiency
  - model building time
  - classification time
- Scalability
  - training set size
  - attribute number
- Robustness
  - noise, missing data
- Interpretability
  - model interpretability
  - model compactness

DBMG   9

Decision trees

DBMG

Data Base and Data Mining Group of Politecnico di Torino

Elena Baralis
Politecnico di Torino

Elena Baralis
Politecnico di Torino

Example of decision tree

Tid	Refund	Marital Status	Taxable Income	Cheat
1	Yes	Single	125K	No
2	No	Married	100K	No
3	No	Single	70K	No
4	Yes	Married	120K	No
5	No	Divorced	95K	Yes
6	No	Married	60K	No
7	Yes	Divorced	220K	No
8	No	Single	85K	Yes
9	No	Married	75K	No
10	No	Single	90K	Yes

categorical categorical continuous class

Training Data

Splitting Attributes

Refund
- Yes → NO
- No → MarSt
  - Single, Divorced → TaxInc
    - < 80K → NO
    - > 80K → YES
  - Married → NO

Model: Decision Tree

From: Tan,Steinbach, Kumar, Introduction to Data Mining, McGraw Hill 2006

11

Another example of decision tree

Tid	Refund	Marital Status	Taxable Income	Cheat
1	Yes	Single	125K	No
2	No	Married	100K	No
3	No	Single	70K	No
4	Yes	Married	120K	No
5	No	Divorced	95K	Yes
6	No	Married	60K	No
7	Yes	Divorced	220K	No
8	No	Single	85K	Yes
9	No	Married	75K	No
10	No	Single	90K	Yes

categorical categorical continuous class

MarSt
- Married → NO
- Single, Divorced → Refund
  - Yes → NO
  - No → TaxInc
    - < 80K → NO
    - > 80K → YES

There could be more than one tree that fits the same data!
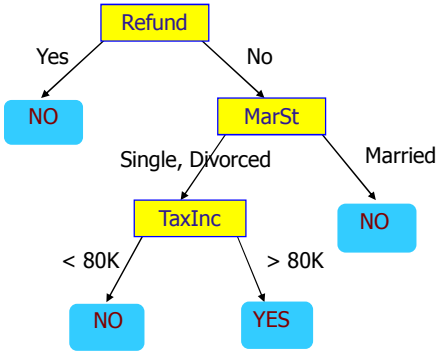
From: Tan,Steinbach, Kumar, Introduction to Data Mining, McGraw Hill 2006

12

Elena Baralis
Politecnico di Torino

6

Apply Model to Test Data

Start from the root of tree.

Test Data

Refund	Marital Status	Taxable Income	Cheat
No	Married	80K	?

Refund
Yes → NO
No → MarSt
Single, Divorced → TaxInc
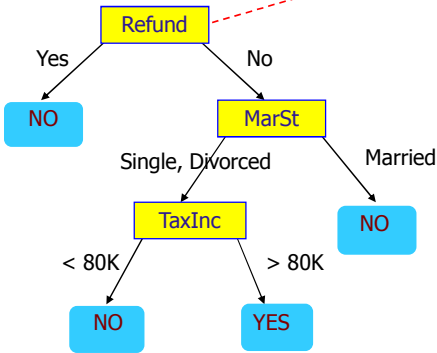Married → NO
< 80K → NO
> 80K → YES

From: Tan,Steinbach, Kumar, Introduction to Data Mining, McGraw Hill 2006

13

Apply Model to Test Data

Test Data

Refund	Marital Status	Taxable Income	Cheat
No	Married	80K	?

Refund
Yes → NO
No → MarSt
Single, Divorced → TaxInc
Married → NO
< 80K → NO
> 80K → YES

From: Tan,Steinbach, Kumar, Introduction to Data Mining, McGraw Hill 2006

14

Elena Baralis
Politecnico di Torino

Apply Model to Test Data

Test Data

Refund	Marital Status	Taxable Income	Cheat
No	Married	80K	?

Refund
- Yes → NO
- No → MarSt
  - Single, Divorced → TaxInc
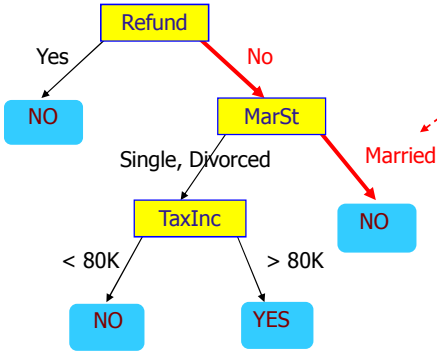   - < 80K → NO
    - > 80K → YES
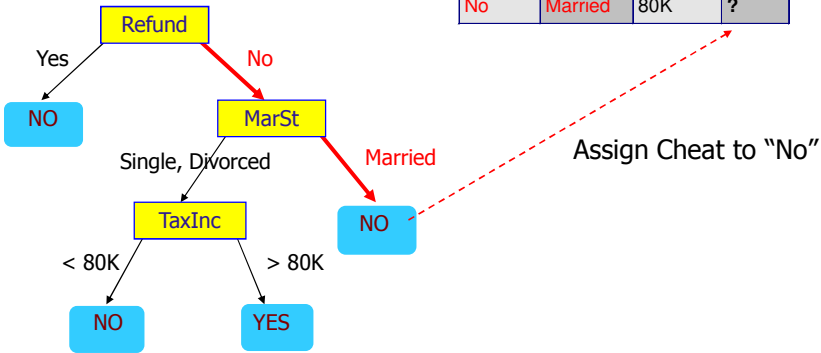 - Married → NO

From: Tan,Steinbach, Kumar, Introduction to Data Mining, McGraw Hill 2006

15

Apply Model to Test Data

Test Data

Refund	Marital Status	Taxable Income	Cheat
No	Married	80K	?

Refund
- Yes → NO
- No → MarSt
  - Single, Divorced → TaxInc
    - < 80K → NO
    - > 80K → YES
  - Married → NO

From: Tan,Steinbach, Kumar, Introduction to Data Mining, McGraw Hill 2006

16

Elena Baralis
Politecnico di Torino

Decision tree induction

- Many algorithms to build a decision tree
  - Hunt's Algorithm (one of the earliest)
  - CART
  - ID3, C4.5, C5.0
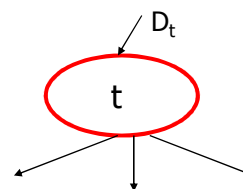 - SLIQ, SPRINT

19

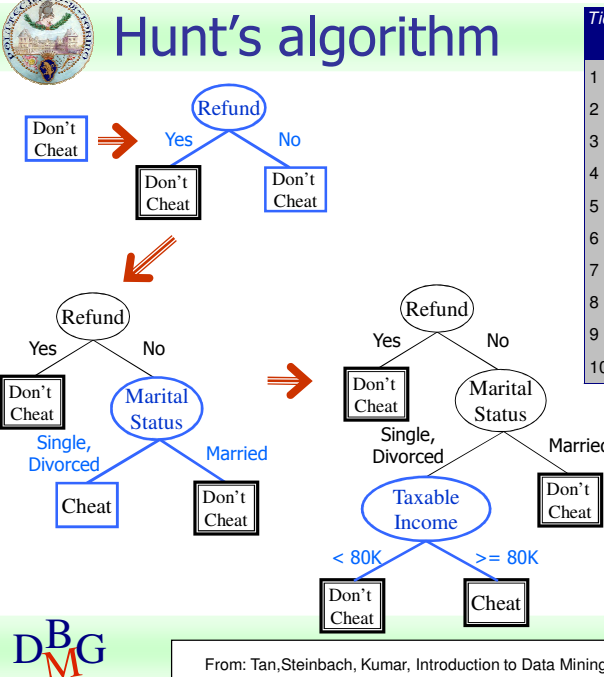# General structure of Hunt's algorithm

Basic steps

- If D_t contains records that belong to the same class y_t
  - then t is a leaf node labeled as y_t
- If D_t contains records that belong to more than one class
  - select the "best" attribute A on which to split D_t and label node t as A
  - split D_t into smaller subsets and recursively apply the procedure to each subset
- If D_t is an empty set
  - then t is a leaf node labeled as the default (majority) class, y_d

D_t, set of training records that reach a node t

20

Elena Baralis
Politecnico di Torino

Hunt's algorithm

Tid	Refund	Marital Status	Taxable Income	Cheat
1	Yes	Single	125K	No
2	No	Married	100K	No
3	No	Single	70K	No
4	Yes	Married	120K	No
5	No	Divorced	95K	Yes
6	No	Married	60K	No
7	Yes	Divorced	220K	No
8	No	Single	85K	Yes
9	No	Married	75K	No
10	No	Single	90K	Yes

From: Tan,Steinbach, Kumar, Introduction to Data Mining, McGraw Hill 2006

21

Decision tree induction

- Adopts a greedy strategy
  - "Best" attribute for the split is selected locally at each step
    - not a global optimum
- Issues
  - Structure of test condition
    - Binary split versus multiway split
  - Selection of the best attribute for the split
  - Stopping condition for the algorithm

22

Structure of test condition

- Depends on attribute type
  - nominal
  - ordinal
  - continuous
- Depends on number of outgoing edges
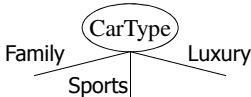 - 2-way split
  - multi-way split

23

Splitting on nominal attributes

- **Multi-way split**
  - use as many partitions as distinct values

Family — CarType — Luxury
Sports

- **Binary split**
  - Divides values into two subsets
  - Need to find optimal partitioning

{Sports, Luxury} — CarType — {Family}   OR   {Family, Luxury} — CarType — {Sports}
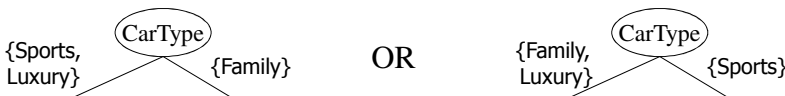
24

Elena Baralis
Politecnico di Torino

Splitting on ordinal attributes

- **Multi-way split**
  - use as many partitions as distinct values
- **Binary split**
  - Divides values into two subsets
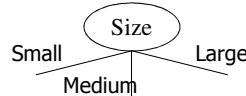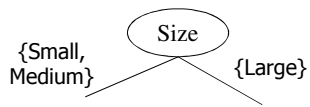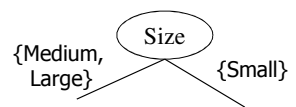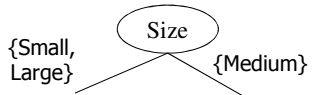 - Need to find optimal partitioning

Size
Small   Large
Medium

{Small, Medium}   Size   {Large}   OR   {Medium, Large}   Size   {Small}

What about this split?   {Small, Large}   Size   {Medium}

25

Splitting on continuous attributes

- Different techniques
  - **Discretization** to form an ordinal categorical attribute
    - Static – discretize once at the beginning
    - Dynamic – discretize during tree induction

    Ranges can be found by equal interval bucketing, equal frequency bucketing (percentiles), or clustering
  - **Binary decision** $(A < v)$ or $(A \geq v)$
    - consider all possible splits and find the best cut
    - more computationally intensive

26

Elena Baralis
Politecnico di Torino

Splitting on continuous attributes

Taxable Income > 80K?
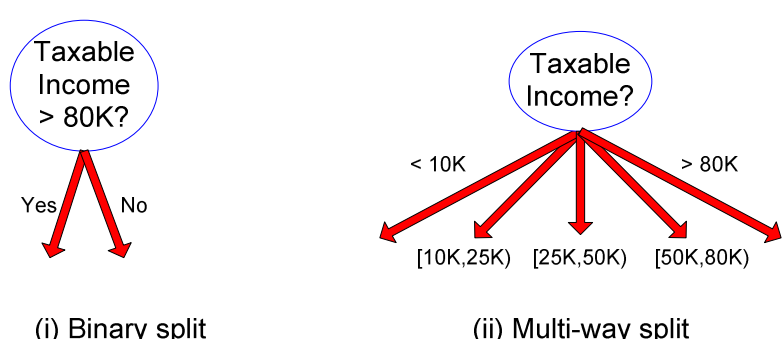
Yes   No

(i) Binary split

Taxable Income?

< 10K   > 80K

[10K,25K) [25K,50K) [50K,80K)
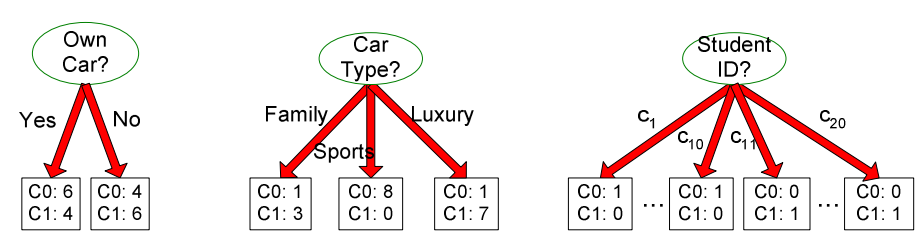
(ii) Multi-way split

27

Selection of the best attribute

Before splitting: 10 records of class 0,
10 records of class 1

Own Car?

Yes   No

C0: 6 C0: 4
C1: 4 C1: 6

Car Type?

Family   Luxury
Sports

C0: 1 C0: 8 C0: 1
C1: 3 C1: 0 C1: 7

Student ID?

c_1 c_{20}
c_{10} c_{11}

C0: 1 ... C0: 1 C0: 0 ... C0: 0
C1: 0   C1: 1 C1: 1   C1: 1

Which attribute (test condition) is the best?

28

Elena Baralis
Politecnico di Torino

14

Selection of the best attribute

- Attributes with *homogeneous* class distribution are preferred
- Need a measure of node impurity

C0: 5
C1: 5

Non-homogeneous,
high degree of impurity

C0: 9
C1: 1

Homogeneous, low
degree of impurity

29

Measures of node impurity

- Many different measures available
  - Gini index
  - Entropy
  - Misclassification error
- Different algorithms rely on different measures

30

How to find the best attribute

Before Splitting:

C0	N00
C1	N01

→ M0

A?

Yes — No

| Node N1 | | Node N2 |

B?

Yes — No

| Node N3 | | Node N4 |

C0	N10
C1	N11

C0	N20
C1	N21

C0	N30
C1	N31

C0	N40
C1	N41

M1   M2   M3   M4

M12   M34

Gain = M0 − M12 vs M0 − M34

From: Tan,Steinbach, Kumar, Introduction to Data Mining, McGraw Hill 2006

31

GINI impurity measure

- Gini Index for a given node t

$$GINI(t) = 1 - \sum_j [p(j\,|\,t)]^2$$

$p(j\,|\,t)$ is the relative frequency of class j at node t

- Maximum $(1 - 1/n_c)$ when records are equally distributed among all classes, implying higher impurity degree
- Minimum (0.0) when all records belong to one class, implying lower impurity degree

C1	0
C2	6
Gini=0.000	

C1	1
C2	5
Gini=0.278	

C1	2
C2	4
Gini=0.444	

C1	3
C2	3
Gini=0.500	

From: Tan,Steinbach, Kumar, Introduction to Data Mining, McGraw Hill 2006

32

Elena Baralis
Politecnico di Torino

Examples for computing GINI

$$GINI(t) = 1 - \sum_j [p(j \mid t)]^2$$

C1	0
C2	6

P(C1) = 0/6 = 0   P(C2) = 6/6 = 1

Gini = 1 − P(C1)^2 − P(C2)^2 = 1 − 0 − 1 = 0

C1	1
C2	5

P(C1) = 1/6   P(C2) = 5/6

Gini = 1 − (1/6)^2 − (5/6)^2 = 0.278

C1	2
C2	4

P(C1) = 2/6   P(C2) = 4/6

Gini = 1 − (2/6)^2 − (4/6)^2 = 0.444

From: Tan,Steinbach, Kumar, Introduction to Data Mining, McGraw Hill 2006

33

Splitting based on GINI

- Used in CART, SLIQ, SPRINT
- When a node p is split into k partitions (children), the quality of the split is computed as

$$GINI_{split} = \sum_{i=1}^{k} \frac{n_i}{n} GINI(i)$$

where

n_i = number of records at child i
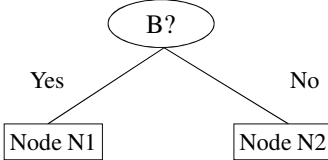
n = number of records at node p

From: Tan,Steinbach, Kumar, Introduction to Data Mining, McGraw Hill 2006

34

Elena Baralis
Politecnico di Torino

Computing GINI index: Boolean attribute

- Splits into two partitions
  - larger and purer partitions are sought for

	Parent
C1	**6**
C2	**6**
Gini = 0.500	

B?

Yes   No

Node N1   Node N2

Gini(N1)
$= 1 - (5/7)^2 - (2/7)^2$
$= 0.408$

Gini(N2)
$= 1 - (1/5)^2 - (4/5)^2$
$= 0.32$

	N1	**N2**
C1	**5**	**1**
C2	**2**	**4**
Gini=?		

Gini(split on B)
$= 7/12 * 0.408 +$
$\quad 5/12 * 0.32$
$= 0.371$

35

Computing GINI index: Categorical attribute

- For each distinct value, gather counts for each class in the dataset
- Use the count matrix to make decisions

Multi-way split

CarType		
Family	**Sports**	**Luxury**
1	2	1
4	1	1
0.393		

(with row labels C1, C2, Gini)

	CarType		
	Family	**Sports**	**Luxury**
C1	1	2	1
C2	4	1	1
Gini	**0.393**		

Two-way split
(find best partition of values)

	CarType	
	{Sports, Luxury}	**{Family}**
C1	3	1
C2	2	4
Gini	**0.400**	

	CarType	
	{Sports}	**{Family, Luxury}**
C1	2	2
C2	1	5
Gini	**0.419**	

36

Elena Baralis
Politecnico di Torino

Computing GINI index: Continuous attribute

- Binary decision on one splitting value
  - Number of possible splitting values
    = Number of distinct values
- Each splitting value v has a count matrix
  - class counts in the two partitions
    - A < v
    - A ≥ v

Tid	Refund	Marital Status	Taxable Income	Cheat
1	Yes	Single	125K	No
2	No	Married	100K	No
3	No	Single	70K	No
4	Yes	Married	120K	No
5	No	Divorced	95K	Yes
6	No	Married	60K	No
7	Yes	Divorced	220K	No
8	No	Single	85K	Yes
9	No	Married	75K	No
10	No	Single	90K	Yes

Taxable Income > 80K?

Yes   No

From: Tan,Steinbach, Kumar, Introduction to Data Mining, McGraw Hill 2006

37

Computing GINI index: Continuous attribute

- For each attribute
  - Sort the attribute on values
  - Linearly scan these values, each time updating the count matrix and computing gini index
  - Choose the split position that has the least gini index

Cheat		No		No		No		Yes		Yes		Yes		No		No		No		No	
							Taxable Income													
Sorted Values		60		70		75		85		90		95		100		120		125		220	
Split Positions	55		65		72		80		87		92		97		110		122		172		230	
	<=	>	<=	>	<=	>	<=	>	<=	>	<=	>	<=	>	<=	>	<=	>	<=	>
Yes	0	3	0	3	0	3	0	3	1	2	2	1	3	0	3	0	3	0	3	0
No	0	7	1	6	2	5	3	4	3	4	3	4	3	4	4	3	5	2	6	1	7	0
Gini		0.420		0.400		0.375		0.343		0.417		0.400		0.300		0.343		0.375		0.400		0.420	

From: Tan,Steinbach, Kumar, Introduction to Data Mining, McGraw Hill 2006

38

Elena Baralis
Politecnico di Torino

19

Entropy impurity measure (INFO)

- Entropy at a given node t

$$Entropy(t) = -\sum_j p(j\,|\,t)\log p(j\,|\,t)$$

  $p(j\,|\,t)$ is the relative frequency of class j at node t

  - Maximum ($\log n_c$) when records are equally distributed among all classes, implying higher impurity degree
  - Minimum (0.0) when all records belong to one class, implying lower impurity degree

- Entropy based computations are similar to GINI index computations

From: Tan,Steinbach, Kumar, Introduction to Data Mining, McGraw Hill 2006

39

Examples for computing entropy

$$Entropy(t) = -\sum_j p(j\,|\,t)\log_2 p(j\,|\,t)$$

C1	0
C2	6

P(C1) = 0/6 = 0   P(C2) = 6/6 = 1

Entropy = $-$ 0 log 0 $-$ 1 log 1 = $-$ 0 $-$ 0 = 0

C1	1
C2	5

P(C1) = 1/6   P(C2) = 5/6

Entropy = $-$ (1/6) $\log_2$ (1/6) $-$ (5/6) $\log_2$ (5/6) = 0.65

C1	2
C2	4

P(C1) = 2/6   P(C2) = 4/6

Entropy = $-$ (2/6) $\log_2$ (2/6) $-$ (4/6) $\log_2$ (4/6) = 0.92

From: Tan,Steinbach, Kumar, Introduction to Data Mining, McGraw Hill 2006

40

Elena Baralis
Politecnico di Torino

Splitting Based on INFO

- Information Gain

$$GAIN_{split} = Entropy(p) - \left(\sum_{i=1}^{k} \frac{n_i}{n} Entropy(i) \right)$$

Parent Node, p is split into k partitions;

n_i is number of records in partition i

  - Measures reduction in entropy achieved because of the split. Choose the split that achieves most reduction (maximizes GAIN)

- Used in ID3 and C4.5

- Disadvantage: Tends to prefer splits yielding a large number of partitions, each small but pure

From: Tan,Steinbach, Kumar, Introduction to Data Mining, McGraw Hill 2006

41

Splitting Based on INFO

- Gain Ratio

$$GainRATIO_{split} = \frac{GAIN_{Split}}{SplitINFO} \qquad SplitINFO = -\sum_{i=1}^{k} \frac{n_i}{n} \log \frac{n_i}{n}$$

Parent Node, p is split into k partitions
n_i is the number of records in partition i

  - Adjusts Information Gain by the entropy of the partitioning (SplitINFO). Higher entropy partitioning (large number of small partitions) is penalized

- Used in C4.5

- Designed to overcome the disadvantage of Information Gain

From: Tan,Steinbach, Kumar, Introduction to Data Mining, McGraw Hill 2006

42

Elena Baralis
Politecnico di Torino

Classification error impurity measure

- Classification error at a node t

$$Error(t) = 1 - \max_i P(i \mid t)$$

- Measures misclassification error made by a node
  - Maximum $(1 - 1/n_c)$ when records are equally distributed among all classes, implying least interesting information
  - Minimum (0.0) when all records belong to one class, implying most interesting information

43

Examples for computing error

$$Error(t) = 1 - \max_i P(i \mid t)$$

C1	0
C2	6

$P(C1) = 0/6 = 0 \quad P(C2) = 6/6 = 1$

Error $= 1 - \max(0, 1) = 1 - 1 = 0$

C1	1
C2	5

$P(C1) = 1/6 \quad P(C2) = 5/6$

Error $= 1 - \max(1/6, 5/6) = 1 - 5/6 = 1/6$

C1	2
C2	4

$P(C1) = 2/6 \quad P(C2) = 4/6$

Error $= 1 - \max(2/6, 4/6) = 1 - 4/6 = 1/3$

44

Elena Baralis
Politecnico di Torino

Comparison among splitting criteria

For a 2-class problem
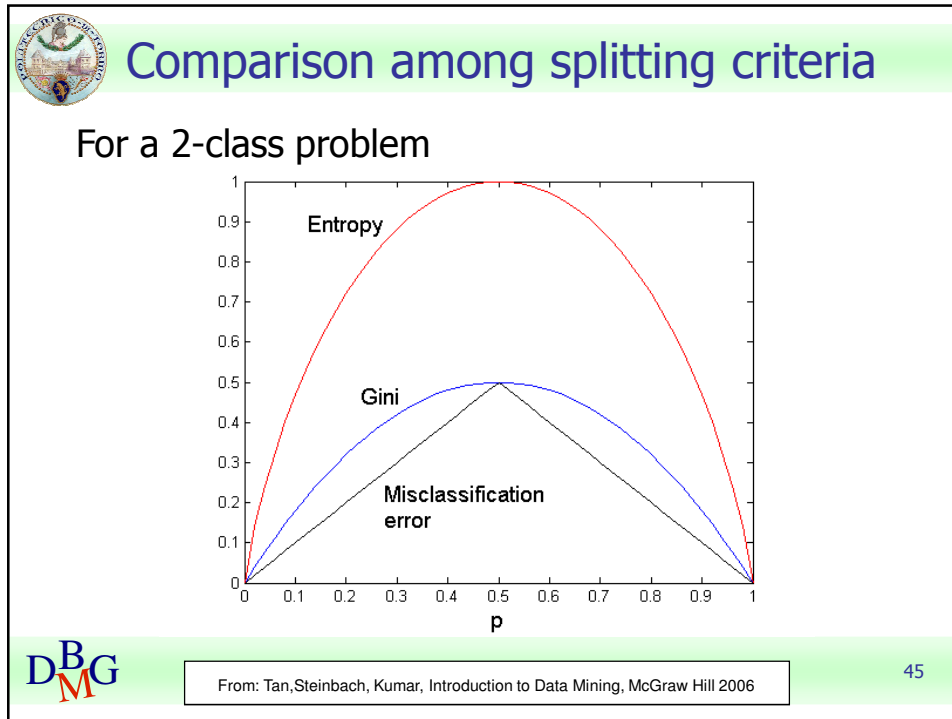
45

Stopping Criteria for Tree Induction

- Stop expanding a node when all the records belong to the same class

- Stop expanding a node when all the records have similar attribute values

- Early termination (to be discussed later)

46
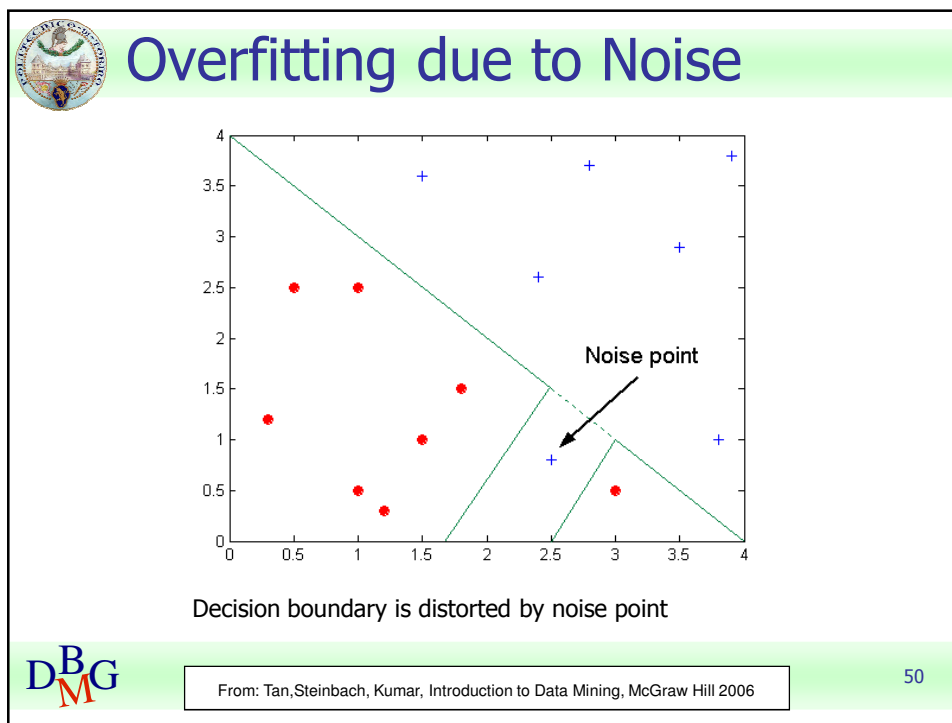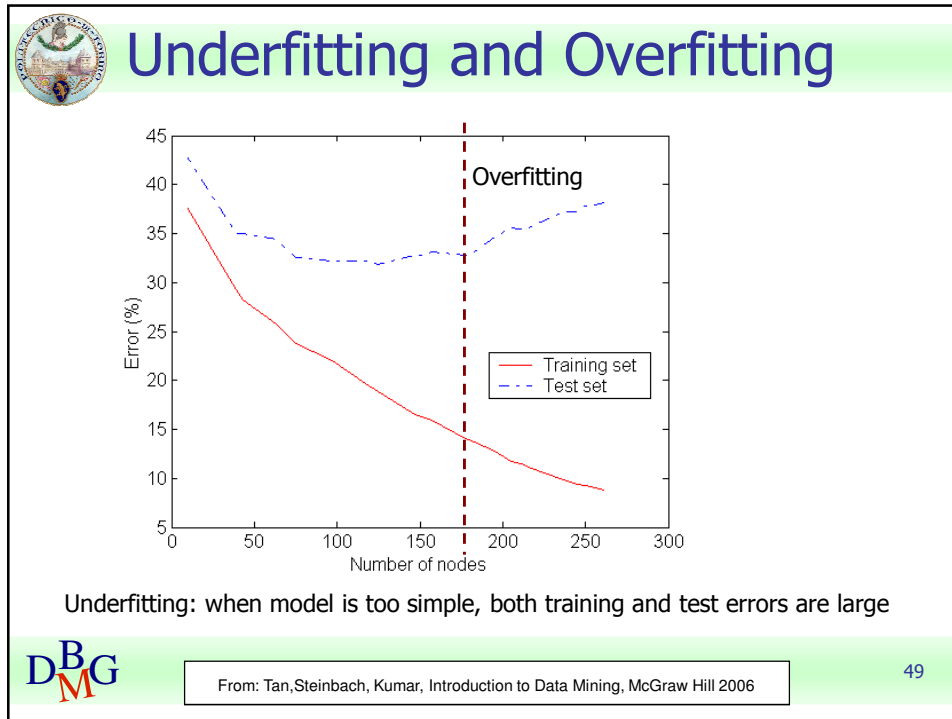
Elena Baralis
Politecnico di Torino

Decision Tree Based Classification

- Advantages
  - Inexpensive to construct
  - Extremely fast at classifying unknown records
  - Easy to interpret for small-sized trees
  - Accuracy is comparable to other classification techniques for many simple data sets
- Disadvantages
  - accuracy may be affected by missing data

47

Practical Issues of Classification

- Underfitting and overfitting
- Missing Values
- Costs of Classification

48

Elena Baralis
Politecnico di Torino

Underfitting and Overfitting

Underfitting: when model is too simple, both training and test errors are large

From: Tan,Steinbach, Kumar, Introduction to Data Mining, McGraw Hill 2006

49

Overfitting due to Noise

Decision boundary is distorted by noise point

From: Tan,Steinbach, Kumar, Introduction to Data Mining, McGraw Hill 2006

50

Elena Baralis
Politecnico di Torino

How to address overfitting

- Pre-Pruning (Early Stopping Rule)
  - Stop the algorithm before it becomes a fully-grown tree
  - Typical stopping conditions for a node:
    - Stop if all instances belong to the same class
    - Stop if all the attribute values are the same
  - More restrictive conditions:
    - Stop if number of instances is less than some user-specified threshold
    - Stop if class distribution of instances are independent of the available features (e.g., using χ^2 test)
    - Stop if expanding the current node does not improve impurity measures (e.g., Gini or information gain).

From: Tan,Steinbach, Kumar, Introduction to Data Mining, McGraw Hill 2006

51

How to address overfitting

- Post-pruning
  - Grow decision tree to its entirety
  - Trim the nodes of the decision tree in a bottom-up fashion
  - If generalization error improves after trimming, replace sub-tree by a leaf node.
  - Class label of leaf node is determined from majority class of instances in the sub-tree

From: Tan,Steinbach, Kumar, Introduction to Data Mining, McGraw Hill 2006

52

Elena Baralis
Politecnico di Torino

Handling missing attribute values

- Missing values affect decision tree construction in three different ways
  - Affects how impurity measures are computed
  - Affects how to distribute instance with missing value to child nodes
  - Affects how a test instance with missing value is classified

53

Other issues

- Data Fragmentation
- Search Strategy
- Expressiveness
- Tree Replication

54

Data fragmentation

- Number of instances gets smaller as you traverse down the tree

- Number of instances at the leaf nodes could be too small to make any statistically significant decision

55

Search strategy

- Finding an optimal decision tree is NP-hard

- The algorithm presented so far uses a greedy, top-down, recursive partitioning strategy to induce a reasonable solution

- Other strategies?
  - Bottom-up
  - Bi-directional

56

Elena Baralis
Politecnico di Torino

Expressiveness

- Decision tree provides expressive representation for learning discrete-valued function
  - But they do not generalize well to certain types of Boolean functions
    - Example: parity function:
      - Class = 1 if there is an even number of Boolean attributes with truth value = True
      - Class = 0 if there is an odd number of Boolean attributes with truth value = True
  - For accurate modeling, must have a complete tree

- Not expressive enough for modeling continuous variables
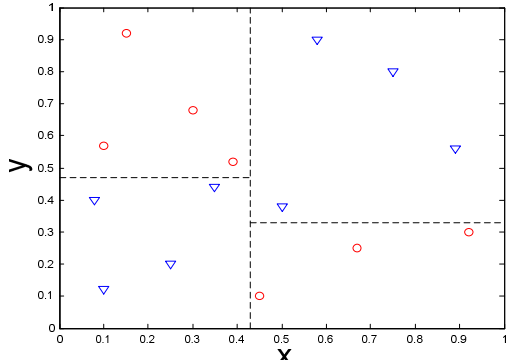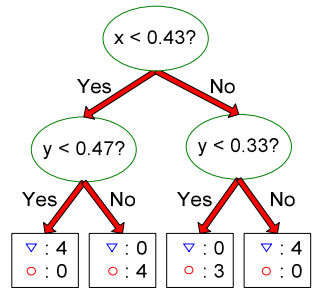 - Particularly when test condition involves only a single attribute at-a-time

From: Tan,Steinbach, Kumar, Introduction to Data Mining, McGraw Hill 2006
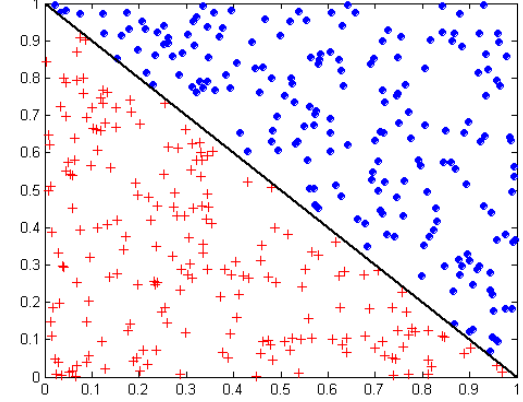
57

Decision boundary

• Border line between two neighboring regions of different classes is known as decision boundary

• Decision boundary is parallel to axes because test condition involves a single attribute at-a-time

From: Tan,Steinbach, Kumar, Introduction to Data Mining, McGraw Hill 2006

58

Elena Baralis
Politecnico di Torino

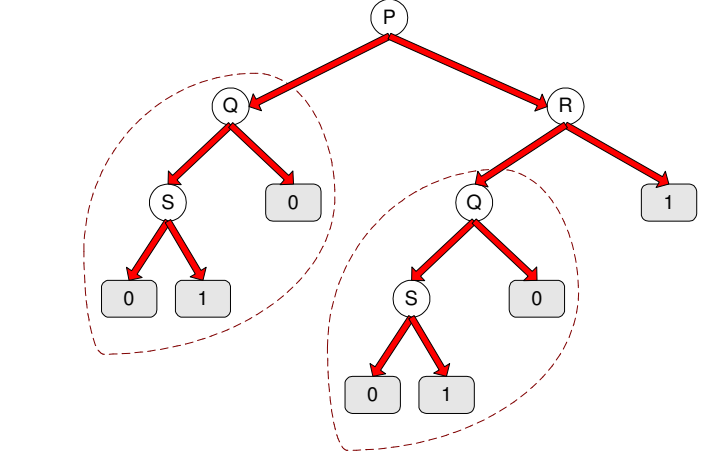## Oblique decision trees

$x + y < 1$

Class = +   Class = ●

- Test condition may involve multiple attributes
- More expressive representation
- Finding optimal test condition is computationally expensive

From: Tan,Steinbach, Kumar, Introduction to Data Mining, McGraw Hill 2006

59

Tree replication

- Same subtree appears in multiple branches

From: Tan,Steinbach, Kumar, Introduction to Data Mining, McGraw Hill 2006

60

Elena Baralis
Politecnico di Torino

30

Rule-based classification

Elena Baralis
Politecnico di Torino

Rule-based classifier

- Classify records by using a collection of "if…then…" rules
- Rule: (*Condition*) → *y*
  - where
    - *Condition* is a conjunction of attributes
    - *y* is the class label
  - *LHS*: rule antecedent or condition
  - *RHS*: rule consequent
- Examples of classification rules
  - (Blood Type=Warm) ∧ (Lay Eggs=Yes) → Birds
  - (Taxable Income < 50K) ∧ (Refund=Yes) → Evade=No

From: Tan,Steinbach, Kumar, Introduction to Data Mining, McGraw Hill 2006

62

Rule-based Classifier (Example)

Name	Blood Type	Give Birth	Can Fly	Live in Water	Class
human	warm	yes	no	no	mammals
python	cold	no	no	no	reptiles
salmon	cold	no	no	yes	fishes
whale	warm	yes	no	yes	mammals
frog	cold	no	no	sometimes	amphibians
komodo	cold	no	no	no	reptiles
bat	warm	yes	yes	no	mammals
pigeon	warm	no	yes	no	birds
cat	warm	yes	no	no	mammals
leopard shark	cold	yes	no	yes	fishes
turtle	cold	no	no	sometimes	reptiles
penguin	warm	no	no	sometimes	birds
porcupine	warm	yes	no	no	mammals
eel	cold	no	no	yes	fishes
salamander	cold	no	no	sometimes	amphibians
gila monster	cold	no	no	no	reptiles
platypus	warm	no	no	no	mammals
owl	warm	no	yes	no	birds
dolphin	warm	yes	no	yes	mammals
eagle	warm	no	yes	no	birds

R1: (Give Birth = no) $\wedge$ (Can Fly = yes) $\rightarrow$ Birds

R2: (Give Birth = no) $\wedge$ (Live in Water = yes) $\rightarrow$ Fishes

R3: (Give Birth = yes) $\wedge$ (Blood Type = warm) $\rightarrow$ Mammals

R4: (Give Birth = no) $\wedge$ (Can Fly = no) $\rightarrow$ Reptiles

R5: (Live in Water = sometimes) $\rightarrow$ Amphibians

From: Tan,Steinbach, Kumar, Introduction to Data Mining, McGraw Hill 2006

63

Rule-based classification

- A rule *r* covers an instance **x** if the attributes of the instance satisfy the condition of the rule

  R1: (Give Birth = no) $\wedge$ (Can Fly = yes) $\rightarrow$ Birds

  R2: (Give Birth = no) $\wedge$ (Live in Water = yes) $\rightarrow$ Fishes

  R3: (Give Birth = yes) $\wedge$ (Blood Type = warm) $\rightarrow$ Mammals

  R4: (Give Birth = no) $\wedge$ (Can Fly = no) $\rightarrow$ Reptiles

  R5: (Live in Water = sometimes) $\rightarrow$ Amphibians

Name	Blood Type	Give Birth	Can Fly	Live in Water	Class
hawk	warm	no	yes	no	?
grizzly bear	warm	yes	no	no	?

Rule R1 covers a hawk => Bird

Rule R3 covers the grizzly bear => Mammal

From: Tan,Steinbach, Kumar, Introduction to Data Mining, McGraw Hill 2006

64

Elena Baralis
Politecnico di Torino

Rule-based classification

R1: (Give Birth = no) ∧ (Can Fly = yes) → Birds
R2: (Give Birth = no) ∧ (Live in Water = yes) → Fishes
R3: (Give Birth = yes) ∧ (Blood Type = warm) → Mammals
R4: (Give Birth = no) ∧ (Can Fly = no) → Reptiles
R5: (Live in Water = sometimes) → Amphibians

Name	Blood Type	Give Birth	Can Fly	Live in Water	Class
lemur	warm	yes	no	no	?
turtle	cold	no	no	sometimes	?
dogfish shark	cold	yes	no	yes	?

A lemur triggers rule R3, so it is classified as a mammal
A turtle triggers both R4 and R5
A dogfish shark triggers none of the rules
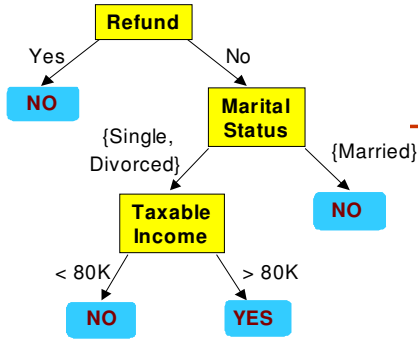
65

Characteristics of rules

- Mutually exclusive rules
  - Classifier contains mutually exclusive rules if the rules are independent of each other
  - Every record is covered by at most one rule

- Exhaustive rules
  - Classifier has exhaustive coverage if it accounts for every possible combination of attribute values
  - Each record is covered by at least one rule

66

Elena Baralis
Politecnico di Torino

From decision trees to rules

Classification Rules

(Refund=Yes) ==> No

(Refund=No, Marital Status={Single,Divorced}, Taxable Income<80K) ==> No

(Refund=No, Marital Status={Single,Divorced}, Taxable Income>80K) ==> Yes
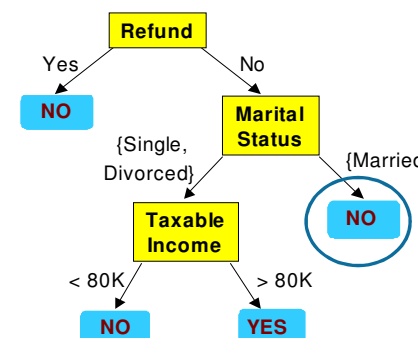
(Refund=No, Marital Status={Married}) ==> No

Rules are mutually exclusive and exhaustive

Rule set contains as much information as the tree

From: Tan,Steinbach, Kumar, Introduction to Data Mining, McGraw Hill 2006

67

Rules can be simplified

Tid	Refund	Marital Status	Taxable Income	Cheat
1	Yes	Single	125K	No
2	No	Married	100K	No
3	No	Single	70K	No
4	Yes	Married	120K	No
5	No	Divorced	95K	Yes
6	No	Married	60K	No
7	Yes	Divorced	220K	No
8	No	Single	85K	Yes
9	No	Married	75K	No
10	No	Single	90K	Yes

Initial Rule: (Refund=No) $\wedge$ (Status=Married) $\rightarrow$ No

Simplified Rule: (Status=Married) $\rightarrow$ No

From: Tan,Steinbach, Kumar, Introduction to Data Mining, McGraw Hill 2006

68

Elena Baralis
Politecnico di Torino

34

Effect of rule simplification

- **Rules are no longer mutually exclusive**
    - A record may trigger more than one rule
    - Solution?
        - Ordered rule set
        - Unordered rule set – use voting schemes
- **Rules are no longer exhaustive**
    - A record may not trigger any rules
    - Solution?
        - Use a default class

69

Ordered rule set

- Rules are rank ordered according to their priority
    - An ordered rule set is known as a decision list
- When a test record is presented to the classifier
    - It is assigned to the class label of the highest ranked rule it has triggered
    - If none of the rules fired, it is assigned to the default class

R1: (Give Birth = no) $\wedge$ (Can Fly = yes) $\rightarrow$ Birds
R2: (Give Birth = no) $\wedge$ (Live in Water = yes) $\rightarrow$ Fishes
R3: (Give Birth = yes) $\wedge$ (Blood Type = warm) $\rightarrow$ Mammals
R4: (Give Birth = no) $\wedge$ (Can Fly = no) $\rightarrow$ Reptiles
R5: (Live in Water = sometimes) $\rightarrow$ Amphibians

Name	Blood Type	Give Birth	Can Fly	Live in Water	Class
turtle	cold	no	no	sometimes	?

70

Associative classification

Data Base and Data Mining Group of Politecnico di Torino

Elena Baralis
Politecnico di Torino

Associative classification

- The classification model is defined by means of association rules

  $$(\textit{Condition}) \rightarrow y$$

  - rule body is an itemset

- Model generation

  - Rule selection & sorting

    - based on support, confidence and correlation thresholds

  - Rule pruning

    - Database coverage: the training set is covered by selecting topmost rules according to previous sort

74

Associative classification

- **Strong points**
  - interpretable model
  - higher accuracy than decision trees
    - correlation among attributes is considered
  - efficient classification
  - unaffected by missing data
  - good scalability in the training set size
- **Weak points**
  - rule generation may be slow
    - it depends on support threshold
  - reduced scalability in the number of attributes
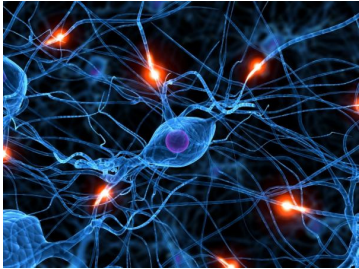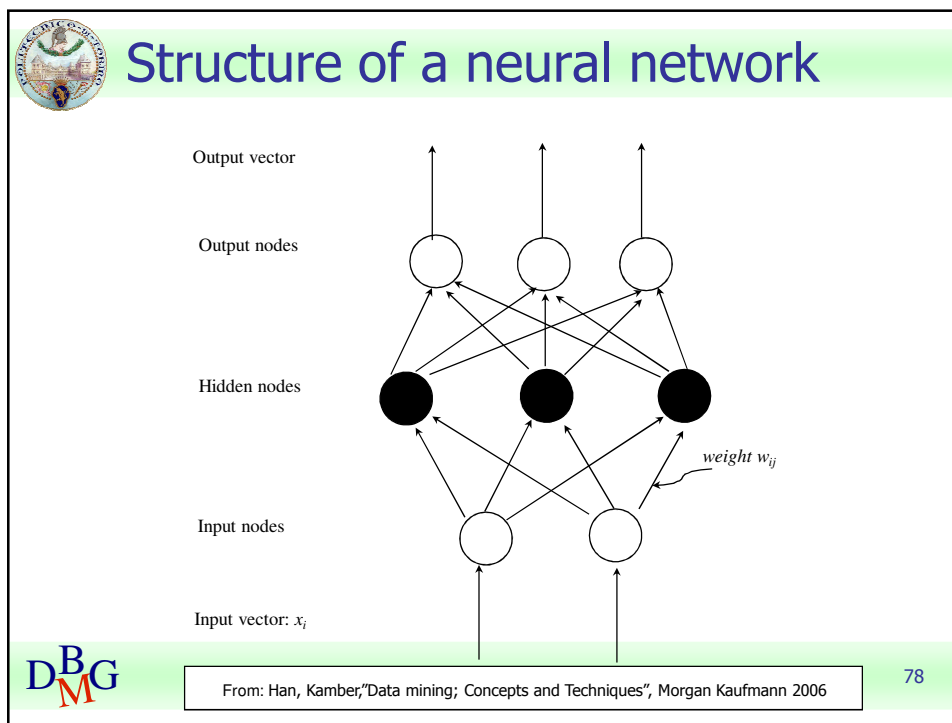   - rule generation may become unfeasible
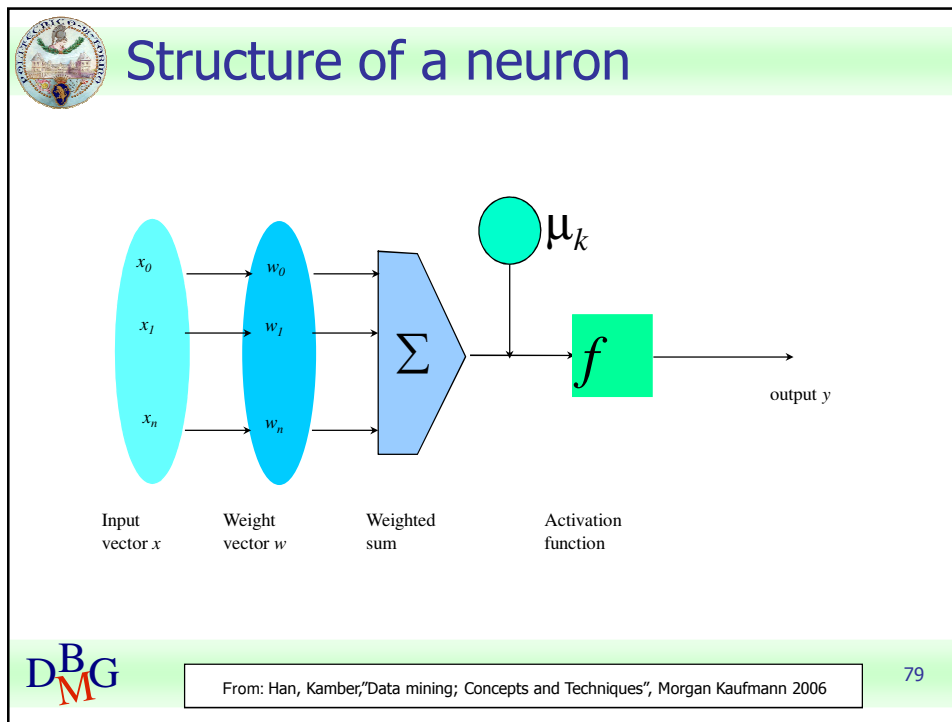
75

Neural networks

Elena Baralis
Politecnico di Torino

Neural networks

- **Inspired to the structure of the human brain**
  - Neurons as elaboration units
  - Synapses as connection network

Structure of a neural network

Output vector

Output nodes

Hidden nodes

Input nodes

Input vector: x_i

weight w_{ij}

From: Han, Kamber,"Data mining; Concepts and Techniques", Morgan Kaufmann 2006

Elena Baralis
Politecnico di Torino

Structure of a neuron

x_0 w_0

x_1 w_1

x_n w_n

μ_k

Σ f

output y

| Input vector x | Weight vector w | Weighted sum | Activation function |

From: Han, Kamber,"Data mining; Concepts and Techniques", Morgan Kaufmann 2006

79

Construction of the neural network

- For each node, definition of
  - set of weights
  - offset value

  providing the highest accuracy on the training data
- Iterative approach on training data instances

80

Construction of the neural network

- Base algorithm
  - Initially assign random values to weights and offsets
  - Process instances in the training set one at a time
    - For each neuron, compute the result when applying weights, offset and activation function for the instance
    - Forward propagation until the output is computed
    - Compare the computed output with the expected output, and evaluate error
    - Backpropagation of the error, by updating weights and offset for each neuron
  - The process ends when
    - % of accuracy above a given threshold
    - % of parameter variation (error) below a given threshold
    - The maximum number of epochs is reached

81

Neural networks

- Strong points
  - High accuracy
  - Robust to noise and outliers
  - Supports both discrete and continuous output
  - Efficient during classification
- Weak points
  - Long training time
    - weakly scalable in training data size
    - complex configuration
  - Not interpretable model
    - application domain knowledge cannot be exploited in the model

82

Bayesian Classification

Elena Baralis
Politecnico di Torino

Bayes theorem

- Let C and X be random variables

    P(C,X) = P(C|X) P(X)

    P(C,X) = P(X|C) P(C)

- Hence

    P(C|X) P(X) = P(X|C) P(C)

- and also

    P(C|X) = P(X|C) P(C) / P(X)

84

Bayesian classification

- Let the class attribute and all data attributes be random variables
    - C = any class label
    - X = $<x_1,…,x_k>$ record to be classified
- Bayesian classification
    - compute P(C|X) for all classes
        - probability that record X belongs to C
    - assign X to the class with *maximal* P(C|X)
- Applying Bayes theorem
    $$P(C|X) = P(X|C) \cdot P(C) / P(X)$$
    - P(X) constant for all C, disregarded for maximum computation
    - P(C) a priori probability of C
        $$P(C) = N_c/N$$

85

Bayesian classification

- How to estimate P(X|C), i.e. $P(x_1,…,x_k|C)$?
- Naïve hypothesis
    $$P(x_1,…,x_k|C) = P(x_1|C) P(x_2|C) … P(x_k|C)$$
    - *statistical independence* of attributes $x_1,…,x_k$
    - not always true
        - model quality may be affected
- Computing $P(x_k|C)$
    - for discrete attributes
        $$P(x_k|C) = |x_{kC}|/ N_c$$
        - where $|x_{kC}|$ is number of instances having value x_k for attribute k and belonging to class C
    - for continuous attributes, use probability distribution
- Bayesian networks
    - allow specifying a subset of dependencies among attributes

86

Bayesian classification: Example

Outlook	Temperature	Humidity	Windy	Class
sunny	hot	high	false	N
sunny	hot	high	true	N
overcast	hot	high	false	P
rain	mild	high	false	P
rain	cool	normal	false	P
rain	cool	normal	true	N
overcast	cool	normal	true	P
sunny	mild	high	false	N
sunny	cool	normal	false	P
rain	mild	normal	false	P
sunny	mild	normal	true	P
overcast	mild	high	true	P
overcast	hot	normal	false	P
rain	mild	high	true	N

From: Han, Kamber,"Data mining; Concepts and Techniques", Morgan Kaufmann 2006

87

Bayesian classification: Example

outlook	
P(sunny|p) = 2/9	P(sunny|n) = 3/5
P(overcast|p) = 4/9	P(overcast|n) = 0
P(rain|p) = 3/9	P(rain|n) = 2/5
temperature	
P(hot|p) = 2/9	P(hot|n) = 2/5
P(mild|p) = 4/9	P(mild|n) = 2/5
P(cool|p) = 3/9	P(cool|n) = 1/5
humidity	
P(high|p) = 3/9	P(high|n) = 4/5
P(normal|p) = 6/9	P(normal|n) = 2/5
windy	
P(true|p) = 3/9	P(true|n) = 3/5
P(false|p) = 6/9	P(false|n) = 2/5

$P(p) = 9/14$

$P(n) = 5/14$

From: Han, Kamber,"Data mining; Concepts and Techniques", Morgan Kaufmann 2006

88

Elena Baralis
Politecnico di Torino

44

Bayesian classification: Example

- Data to be labeled

    X = <rain, hot, high, false>

- For class p

    P(X|p)·P(p) =
    = P(rain|p)·P(hot|p)·P(high|p)·P(false|p)·P(p)
    = 3/9·2/9·3/9·6/9·9/14 = 0.010582

- For class n

    P(X|n)·P(n) =
    = P(rain|n)·P(hot|n)·P(high|n)·P(false|n)·P(n)
    = 2/5·2/5·4/5·2/5·5/14 = 0.018286

From: Han, Kamber,"Data mining; Concepts and Techniques", Morgan Kaufmann 2006

89
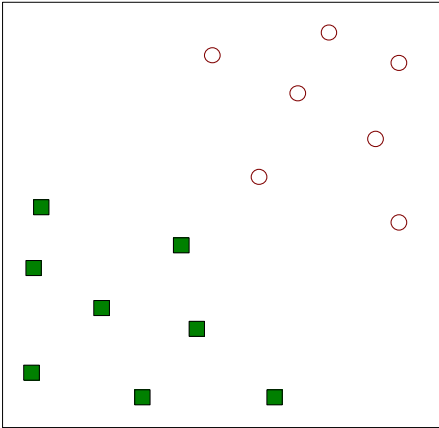
Support Vector Machines

Elena Baralis
Politecnico di Torino

Support Vector Machines

- Find a linear hyperplane (decision boundary) that will separate the data
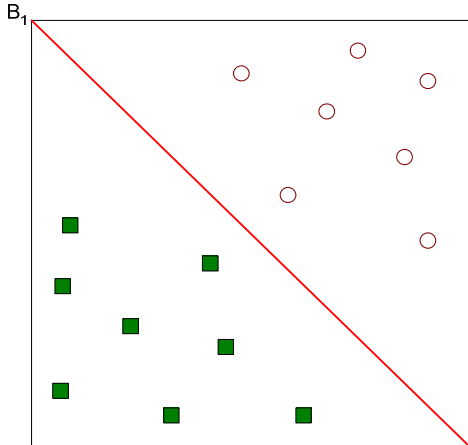
91

Support Vector Machines

B_1

- One Possible Solution
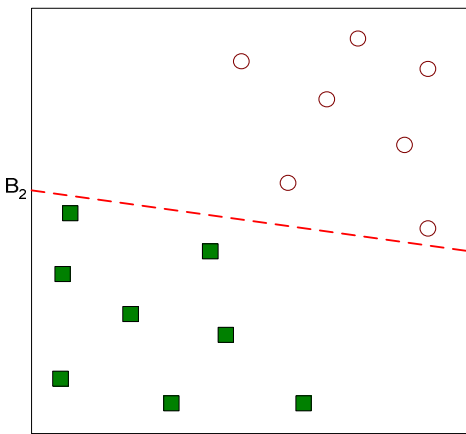
92

Elena Baralis
Politecnico di Torino

Support Vector Machines

- Another possible solution
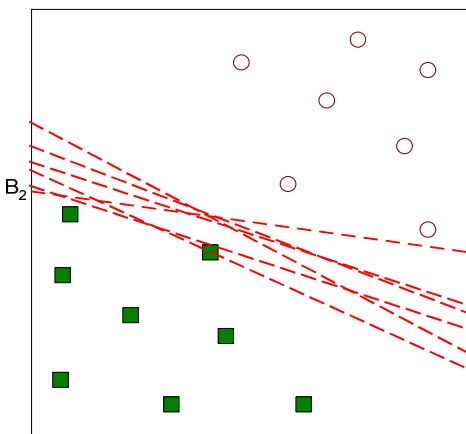
93

Support Vector Machines

- Other possible solutions

94

Elena Baralis
Politecnico di Torino
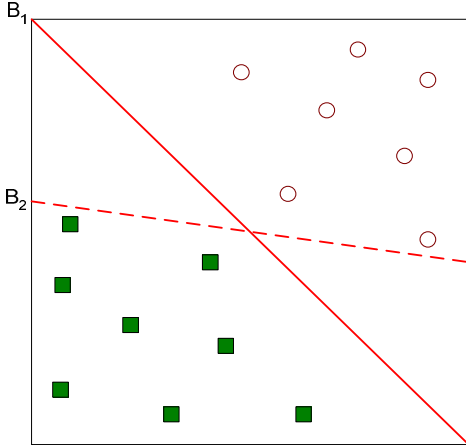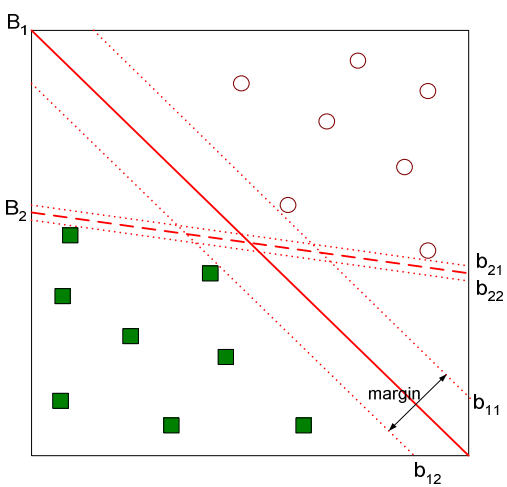
Support Vector Machines

- Which one is better? B1 or B2?
- How do you define better?

95

Support Vector Machines
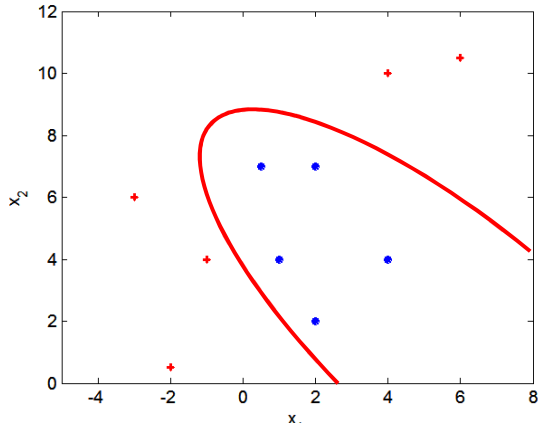
- Find hyperplane maximizes the margin => B1 is better than B2

96

Elena Baralis
Politecnico di Torino

Nonlinear Support Vector Machines

- What if decision boundary is not linear?
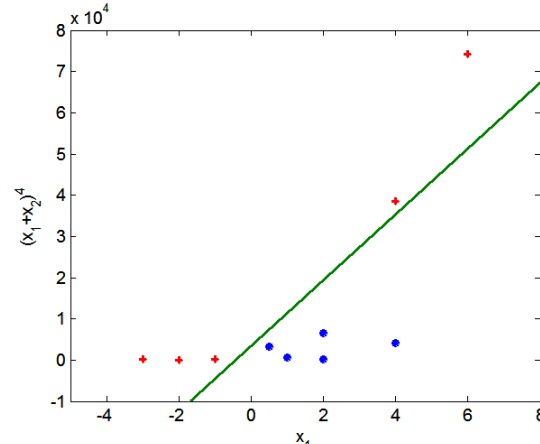
97

Nonlinear Support Vector Machines

- Transform data into higher dimensional space

98

Elena Baralis
Politecnico di Torino

K-Nearest Neighbor

Data Base and Data Mining Group of Politecnico di Torino

Elena Baralis
Politecnico di Torino

Instance-Based Classifiers

Set of Stored Cases

Atr1	……...	AtrN	Class
			A
			B
			B
			C
			A
			C
			B

• Store the training records

• Use training records to predict the class label of unseen cases

Unseen Case

Atr1	……...	AtrN

From: Tan,Steinbach, Kumar, Introduction to Data Mining, McGraw Hill 2006

100

Instance Based Classifiers

- Examples
  - Rote-learner
    - Memorizes entire training data and performs classification only if attributes of record match one of the training examples exactly
  - Nearest neighbor
    - Uses k "closest" points (nearest neighbors) for performing classification
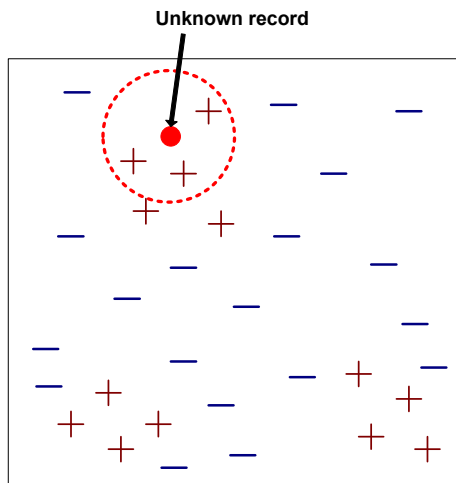
101

Nearest-Neighbor Classifiers
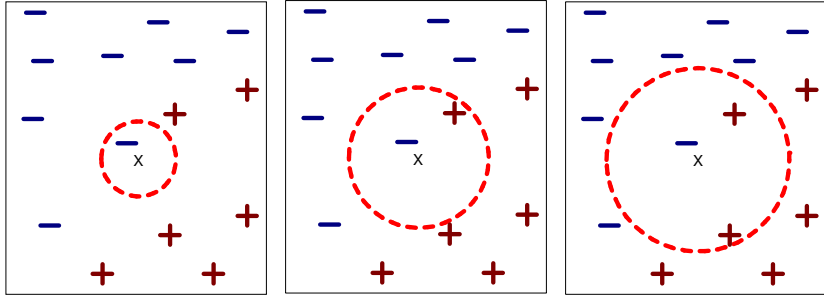
Unknown record

- Requires three things
  - The set of stored records
  - Distance Metric to compute distance between records
  - The value of k, the number of nearest neighbors to retrieve

- To classify an unknown record:
  - Compute distance to other training records
  - Identify k nearest neighbors
  - Use class labels of nearest neighbors to determine the class label of unknown record (e.g., by taking majority vote)

102

Definition of Nearest Neighbor

(a) 1-nearest neighbor   (b) 2-nearest neighbor   (c) 3-nearest neighbor

K-nearest neighbors of a record x are data points that have the k smallest distance to x
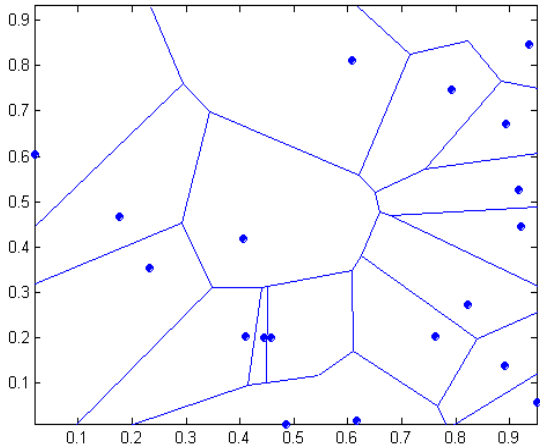
From: Tan,Steinbach, Kumar, Introduction to Data Mining, McGraw Hill 2006

103

1 nearest-neighbor

Voronoi Diagram

From: Tan,Steinbach, Kumar, Introduction to Data Mining, McGraw Hill 2006

104

Elena Baralis
Politecnico di Torino

Nearest Neighbor Classification

- Compute distance between two points:
  - Euclidean distance

$$d(p,q) = \sqrt{\sum_i (p_i - q_i)^2}$$

- Determine the class from nearest neighbor list
  - take the majority vote of class labels among the k-nearest neighbors
  - Weigh the vote according to distance
    - weight factor, $w = 1/d^2$

From: Tan,Steinbach, Kumar, Introduction to Data Mining, McGraw Hill 2006

105

Nearest Neighbor Classification

- Choosing the value of k:
  - If k is too small, sensitive to noise points
  - If k is too large, neighborhood may include points from other classes

From: Tan,Steinbach, Kumar, Introduction to Data Mining, McGraw Hill 2006

106
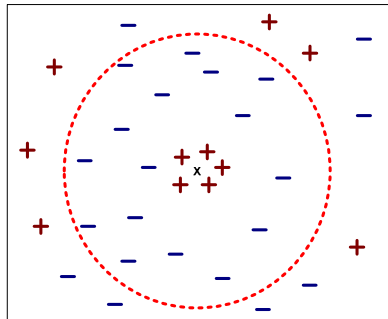
Nearest Neighbor Classification

- Scaling issues
  - Attribute domain should be normalized to prevent distance measures from being dominated by one of the attributes
  - Example: height [1.5m to 2.0m] vs. income [$10K to $1M]
- Problem with distance measures
  - High dimensional data
    - curse of dimensionality

107

Model evaluation

Elena Baralis
Politecnico di Torino

Model evaluation

- Methods for performance evaluation
    - Partitioning techniques for training and test sets
- Metrics for performance evaluation
    - Accuracy, other measures
- Techniques for model comparison
    - ROC curve

DBMG

109

Methods for performance evaluation

- Objective
    - reliable estimate of performance
- Performance of a model may depend on other factors besides the learning algorithm
    - Class distribution
    - Cost of misclassification
    - Size of training and test sets

DBMG

From: Tan,Steinbach, Kumar, Introduction to Data Mining, McGraw Hill 2006

110

Elena Baralis
Politecnico di Torino

Learning curve

- Learning curve shows how accuracy changes with varying sample size
- Requires a sampling schedule for creating learning curve:
    - Arithmetic sampling (Langley, et al)
    - Geometric sampling (Provost et al)

Effect of small sample size:
- Bias in the estimate
- Variance of estimate

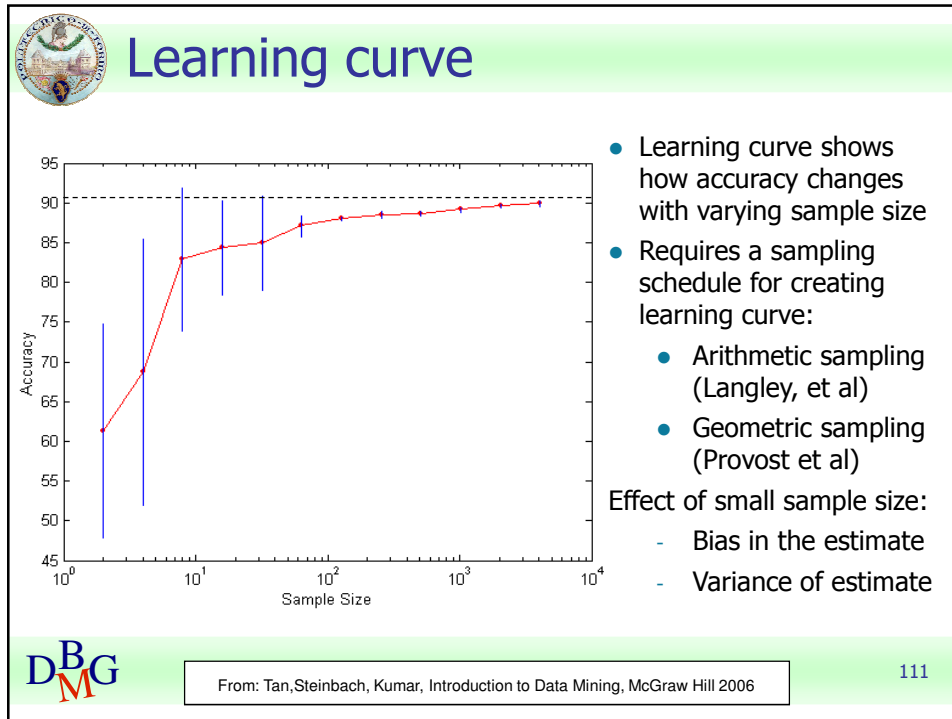From: Tan,Steinbach, Kumar, Introduction to Data Mining, McGraw Hill 2006

111

Methods of estimation

- Partitioning labeled data in
    - training set for model building
    - test set for model evaluation
- Several partitioning techniques
    - holdout
    - cross validation
- Stratified sampling to generate partitions
    - without replacement
- Bootstrap
    - Sampling with replacement

112

Elena Baralis
Politecnico di Torino

Holdout

- Fixed partitioning
  - reserve 2/3 for training and 1/3 for testing
- Appropriate for large datasets
  - may be repeated several times
    - repeated holdout

113

Cross validation

- Cross validation
  - partition data into k disjoint subsets (i.e., folds)
  - k-fold: train on k-1 partitions, test on the remaining one
    - repeat for all folds
  - reliable accuracy estimation, not appropriate for very large datasets
- Leave-one-out
  - cross validation for k=n
  - only appropriate for very small datasets

114

Elena Baralis
Politecnico di Torino

Metrics for model evaluation

- Evaluate the predictive accuracy of a model
- Confusion matrix
  - binary classifier

	PREDICTED CLASS		
ACTUAL CLASS		Class=Yes	Class=No
	Class=Yes	a	b
	Class=No	c	d

a: TP (true positive)
b: FN (false negative)
c: FP (false positive)
d: TN (true negative)

From: Tan,Steinbach, Kumar, Introduction to Data Mining, McGraw Hill 2006

115

Accuracy

- Most widely-used metric for model evaluation

$$Accuracy = \frac{Number\ of\ correctly\ classified\ objects}{Number\ of\ classified\ objects}$$

- Not always a reliable metric

116

Elena Baralis
Politecnico di Torino

Accuracy

- For a binary classifier

		PREDICTED CLASS	
		Class=Yes	Class=No
ACTUAL CLASS	Class=Yes	a (TP)	b (FN)
	Class=No	c (FP)	d (TN)

$$Accuracy = \frac{a+d}{a+b+c+d} = \frac{TP+TN}{TP+TN+FP+FN}$$

117

Limitations of accuracy

- Consider a binary problem
  - Cardinality of Class 0 = 9900
  - Cardinality of Class 1 = 100
- Model

    $() \rightarrow$ *class 0*

  - Model predicts everything to be class 0
    - accuracy is 9900/10000 = 99.0 %
- Accuracy is misleading because the model does not detect any class 1 object

118

Elena Baralis
Politecnico di Torino

59

Limitations of accuracy

- Classes may have different importance
  - Misclassification of objects of a given class is more important
  - e.g., ill patients erroneously assigned to the healthy patients class
- Accuracy is not appropriate for
  - unbalanced class label distribution
  - different class relevance

119

Class specific measures

- Evaluate separately for each class

$$Recall\ (r) = \frac{Number\ of\ objects\ correctly\ assigned\ to\ C}{Number\ of\ objects\ belonging\ to\ C}$$

$$Precision\ (p) = \frac{Number\ of\ objects\ correctly\ assigned\ to\ C}{Number\ of\ objects\ assigned\ to\ C}$$

- Maximize

$$F\text{-}measure\ (F) = \frac{2rp}{r+p}$$

120

Elena Baralis
Politecnico di Torino

Class specific measures

- For a binary classification problem
  - on the confusion matrix, for the positive class

$$\text{Precision (p)} = \frac{a}{a+c}$$

$$\text{Recall (r)} = \frac{a}{a+b}$$

$$\text{F - measure (F)} = \frac{2rp}{r+p} = \frac{2a}{2a+b+c}$$

From: Tan,Steinbach, Kumar, Introduction to Data Mining, McGraw Hill 2006

121

ROC (Receiver Operating Characteristic)

- Developed in 1950s for signal detection theory to analyze noisy signals
  - characterizes the trade-off between positive hits and false alarms
- ROC curve plots
  - TPR, True Positive Rate (on the y-axis)
        TPR = TP/(TP+FN)
  against
  - FPR, False Positive Rate (on the x-axis)
        FPR = FP/(FP + TN)

122

ROC curve

(FPR, TPR)

- (0,0): declare everything to be negative class
- (1,1): declare everything to be positive class
- (0,1): ideal

- Diagonal line
  - Random guessing
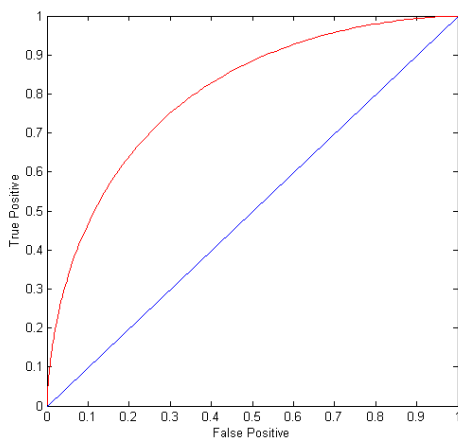 - Below diagonal line
    - prediction is opposite of the true class

From: Tan,Steinbach, Kumar, Introduction to Data Mining, McGraw Hill 2006

123

How to build a ROC curve

Instance	P(+|A)	True Class
1	0.95	+
2	0.93	+
3	0.87	-
4	0.85	-
5	0.85	-
6	0.85	+
7	0.76	-
8	0.53	+
9	0.43	-
10	0.25	+

- Use classifier that produces posterior probability for each test instance P(+|A)
- Sort the instances according to P(+|A) in decreasing order
- Apply threshold at each unique value of P(+|A)
- Count the number of TP, FP, TN, FN at each threshold
  - TP rate

    TPR = TP/(TP+FN)
  - FP rate

    FPR = FP/(FP + TN)

From: Tan,Steinbach, Kumar, Introduction to Data Mining, McGraw Hill 2006

124

How to build a ROC curve

Class	+	-	+	-	-	-	+	-	+	+	
P(+|A)	0.25	0.43	0.53	0.76	0.85	0.85	0.85	0.87	0.93	0.95	1.00
TP	5	4	4	3	3	3	3	2	2	1	0
FP	5	5	4	4	3	2	1	1	0	0	0
TN	0	0	1	1	2	3	4	4	5	5	5
FN	0	1	1	2	2	2	2	3	3	4	5
TPR	1	0.8	0.8	0.6	0.6	0.6	0.6	0.4	0.4	0.2	0
FPR	1	1	0.8	0.8	0.6	0.4	0.2	0.2	0	0	0

ROC Curve

From: Tan,Steinbach, Kumar, Introduction to Data Mining, McGraw Hill 2006

125

Using ROC for Model Comparison

- No model consistently outperforms the other
  - M_1 is better for small FPR
  - M_2 is better for large FPR
- Area under ROC curve
  - Ideal
    Area = 1.0
  - Random guess
    Area = 0.5

From: Tan,Steinbach, Kumar, Introduction to Data Mining, McGraw Hill 2006

126

Elena Baralis
Politecnico di Torino