Classification Lecture Notes (cse352)

- PART ONE: Supervised learning and Classification
- Data format: training and test data
- Concept, or class definitions and description
- Rules learned: characteristic and discriminant
- Supervised learning = classification process = building a classifier.
- Classification algorithms
- Evaluating predictive accuracy of a classifier: the most common methods
- Unsupervised learning= clustering
- Clustering methods

Part 2: Classification Algorithms (Models, Basic Classifiers)

- Decision Trees (ID3, C4.5)
- Neural Networks
- Genetic Algorithm
- Bayesian Classifiers (Networks)
- Rough Sets

Part 3: Other Classification Methods

- k-nearest neighbor classifier
- Case-based reasoning
- Fuzzy set approaches

PART 1: Learning Functionalities (1)
Classification Data

- **Data format:** a data table with key attribute removed. Special attribute- class attribute must be distinguished,
- Here it is Buys_computer

age	income	student	credit_rating	buys_computer
<=30	high	no	fair	no
<=30	high	no	excellent	no
30…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

Part 1: Learning Functionalities
Classification Data with objects

rec	Age	Income	Student	Credit_rating	Buys_computer
r1	<=30	High	No	Fair	No
r2	<=30	High	No	Excellent	No
r3	31…40	High	No	Fair	Yes
r4	>40	Medium	No	Fair	Yes
r5	>40	Low	Yes	Fair	Yes
r6	>40	Low	Yes	Excellent	No
r7	31…40	Low	Yes	Excellent	Yes
r8	<=30	Medium	No	Fair	No
r9	<=30	Low	Yes	Fair	Yes
r10	>40	Medium	Yes	Fair	Yes
r11	<-=30	Medium	Yes	Excellent	Yes
r12	31…40	Medium	No	Excellent	Yes
r13	31…40	High	Yes	Fair	Yes
r14	>40	Medium	No	Excellent	No

Learning Functionalities (2)
Concept or Class Definitions

- **Syntactically a Concept or a Class** is defined by the concept (class) attribute **c** and its value **v**
- **Semantically Concept or Class** – is any subset of records.
- **Concept or Class (syntactic) description** is written as : **c=v**
- **Semantically**, a concept, or a class defined by the attibute **c** is the set of all records for which the attribute **c** has a value **v**.

Learning Functionalities (3)
Concept or Class definitions

- **Example:**

Set of records { r1, r2, r6, r8, r14} of the table on the previous slide is a CONCEPT.

It is defined syntactically by the class attribute buys_computer and its value no;

Concept (class) { r1, r2, r6, r8, r14} description is: buys_computer= no because

{ r1, r2, r6, r8, r14} = {r: buys_computer= no }

Learning Functionalities (4)
Concept, Class characteristics

Characteristics of a class (concept) C

is a set of attributes a1, a2, … ak, and their respective values v1, v2, …. vk such that the intersection of set ***R1*** *of all records for which a1=v1 & a2=v2&…..ak=vk with set* ***C*** *is not empty*

Characteristics description *of* ***C*** *of is then syntactically written as a1=v1 & a2=v2&…..ak=vk*

REMARK: A concept ***C*** *can have many characteristic descriptions.*

Learning Functionalities (5)
Characteristics
Characteristc Descriptions

Definition:

A formula a1=v1 & a2=v2&…..ak=vk (of a proper language) is called a characteristic description for a class ***C***

If and only if

{r: *a1=v1 & a2=v2&…..ak=vk* ***} Λ C = not empty set***

Learning Functionalities (6)
Characteristic Descriptions

Example:

- *Some of the **characteristic descriptions***

*of the concept **C** with description:* **buys_computer= no**

are

- Age=<= 30 & income=high & student=no & credit_rating=fair
- Age=>40& income=medium & student=no & credit_rating=excellent
- Age=>40& income=medium
- Age=<= 30
- student=no & credit_rating=excellent

Learning Functionalities (7)
Concept, Class characteristics

- A formula
- Income=low is *the* ***characteristic description***
- *of the concept* ***C1*** *with description:*

buys_computer= yes

and *of the concept* ***C2*** *with description:*

buys_computer= no

- A formula
- Age<=30 & Income=low is NOT *the* ***characteristic description***

of the concept ***C1*** *with description:* **buys_computer= yes**

because:

{ r: Age<=30 & Income=low } /\ {r: **buys_computer= yes }=**

emptyset

Characteristic Formula

Any formula (of a proper language) of a form

IF concept desciption THEN characteristics

is called a characteristic formula

Example:

- IF buys_computer= no **THEN** income = low & student=yes & credit=excellent
- IF buys_computer= no **THEN** income = low & credit=fair

Characteristic Rule (1)

- A characteristic formula

IF concept desciption THEN characteristics

is called **a characteristic rule** (for a given database)

if and only if it is TRUE in the given database, i.e.

{r: concept description} &{r: characteristics} = not empty set

Characteristic Rule (2)

EXAMPLE:

The formula

- IF buys_computer= no **THEN** income = low & student=yes & credit=excellent

is a characteristic rule for our database because

{r: buys_computer= no **} = {**r1,r2, r6, r8, r14 **},**

{r: income = low & student=yes & credit=excellent **} = {**r6,r7**}**

and

{r1,r2, r6, r8, r14 **} & {**r6,r7**} =** not emptyset

Characteristic Rule (3)

EXAMPLE:

The formula

- IF buys_computer= no **THEN** income = low & credit=fair

Is NOT a characteristic rule for our database because

{r: buys_computer= no **} = {**r1,r2, r6, r8, r14 **},**

{r: income = low & credit=fair**} = {**r5, r9 **}**

and

{r1,r2, r6, r8, r16 **} & {**r5,r9**} =** emptyset

Discrimination

- *Discrimination is the process which aim is to find rules that allow us to **discriminate** the objects (records) belonging to a given concept (one class) from the rest of records (classes)*

If characteristics then concept

- *Example*
- ***If*** Age=<= 30 & income=high & student=no & credit_rating=fair ***then*** buys_computer= no

Discriminant Formula

A discriminant formula is any formula

If characteristics then concept

- Example:
- IF **Age=>40 & inc=low** THEN **buys_comp= no**

Discriminant Rule

- A discriminant formula

  If characteristics then concept

  *is a **DISCRIMINANT RULE** (in a given database)*

iff

{r: Characteristic} ⊏ {r: concept}

Discriminant Rule

- ***Example:***

A discrimant formula

IF **Age=>40 & inc=low** THEN **buys_comp= no**

IS NOT a discriminant rule *in our data base*

As {r: Age=>40 & inc=low} = *{r5, r6} is not a subset of the set*

{r: buys_comp= no}= {r1, r2, r6, r8, r14}

Characteristic and discriminant rules

- The inverse implication to the characteristic rule is usually NOT a discriminant rule
- Example : the inverse implication to our characteristic rule: ***If*** buys_computer= no **then** income = low & student=yes & credit=excellent is
- ***If*** income = low & student=yes & credit=excellent **then** buys_computer= no
- The above rule is NOT a discriminant rule as it can't discriminate between concept with description buys_computer= no and buys_computer= yes
- (see records r6 and r8 in our training dataset)

Supervised Learning Goal (1)

- Given a data set and a concept **c** defined in this dataset FIND a minimal set (or as small as possible set) characteristic, and/or discriminant rules, or other descriptions for the concept **c**, or class, or classes.

Supervised Learning Goal (2)

- We also want these rules to involve as few attributes as it is possible, i.e. we want the rules to have as short as possible length of descriptions.

Supervised Learning

- The process of creating discriminant and/or characteristic rules and TESTING them
- is called a learning process, and when it is finished we say that the concept has been learned (and tested) from examples (records in the dataset).

- It is called a supervised learning because we know the concept description and examples.

A small, full set DISCRIMINANT RULES for concepts: buys_comp=yes, buys_comp=no

- The rules are:

IF *age* = "<=30" AND *student* = "*no*" THEN *buys_computer* = "*no*"

IF *age* = "<=30" AND *student* = "*yes*" THEN *buys_computer* = "*yes*"

IF *age* = "31…40" THEN *buys_computer* = "*yes*"

IF *age* = ">40" AND *credit_rating* = "*excellent*" THEN *buys_computer* = "no"

IF *age* = "<=30" AND *credit_rating* = "*fair*" THEN *buys_computer* = "yes"

Rules testing

- In order to use rules for testing, and later when testing is done and predictive accuracy is acceptable we write rules in a **predicate form**:

IF *age*(x, <=30) AND *student(x, no)* THEN

buys_computer (x, *no*)

IF *age(x,* <=30) AND *student* (x, *yes*) THEN

buys_computer (x, *yes)*

- Attributes and their values of the new record x are matched with the IF part of the rule and the record is is classified accordingly to the THEN part of the rule.

Test dataset

- The Test Dataset has the same format as the training dataset, i.e. the values of concept attribute are known
- We use it to evaluate the predictive accuracy of our rule
- PREDICTIVE ACCURACY of the set of rules, or any classification algorithm is a percentage of well classified data in the testing dataset.
- If the predictive accuracy is not high enough we chose a different learning and testing datasets and start process again
- There are many methods of testing the rules and they will be discussed later

Genaralization: Classification and Classifiers

- Given a data base table DB with a special atribute C, called a class attribute (or decision attribute). The values: C1, C2, ...Cn of the class atrribute are called class labels.
- Example:

a1	a2	a3	a4	C
1	1	m	g	c1
0	1	v	g	c2
1	0	m	b	c1

Classification and Classifiers

- The class attribute C partitions records in the DB i.e. divides records into disjoint subsets defined by the attributes C values, called classes or shortly CLASSIFIES the records. It means we use the attribute C and its values to divide the set R of records od DB into n disjoint classes:

  $C1=\{ r \in DB: \quad C=c1\} \quad ..... \quad Cn=\{r \in BD: C=cn\}$

- Example (from our table)

  $C1 = \{ r: c=c1\} = \{r1,r3\}$

  $C2 = \{r: c=c2\}=\{r2\}$

Classification and Classifiers

- An algorithm (model, method) is called a classification algorithm if it uses the data and its classification to build a set of patterns: discriminant and /or characteristic rules or other pattern descriptions. Those patters are structured in such a way that we can use them to classify unknown sets of objects- unknown records.
- For that reason, and because of the goal a classification algorithm is often called shortly a classifier.
- The name classifier implies more then just classification algorithm.
- A classifier is a final product of the data set and a classification algorithm.

Classification and Classifiers

- Building a classifier consists of two phases:

  training and testing.

- In both phases we use data (training data set and disjoint with it test data set) for which the class labels are known for ALL of the records.
- We use the training data set to create patterns (rules, trees, or to train a Neural or Bayesian network).
- We evaluate created patterns with the use of of test data, which classification is known.
- The measure for a trained classifier accuracy is called predictive accuracy.
- **The classifier is build** i.e. we terminate the process if it has been trained and tested and predictive accuracy was on an acceptable level.

Classification = Supervised Learning

- **Classification = Supervised learning goal:**

  Finding models (rules) that describe (characterize) or/ and distinguish (discriminate) classes or concepts for future prediction

  Example: classify countries based on climate, or classify cars based on gas mileage and use it to predict classification of a new car on a base of other attributes

  Presentation: decision-tree, classification rules, neural network, Bayes network

Classification vs. Prediction

- Classification:

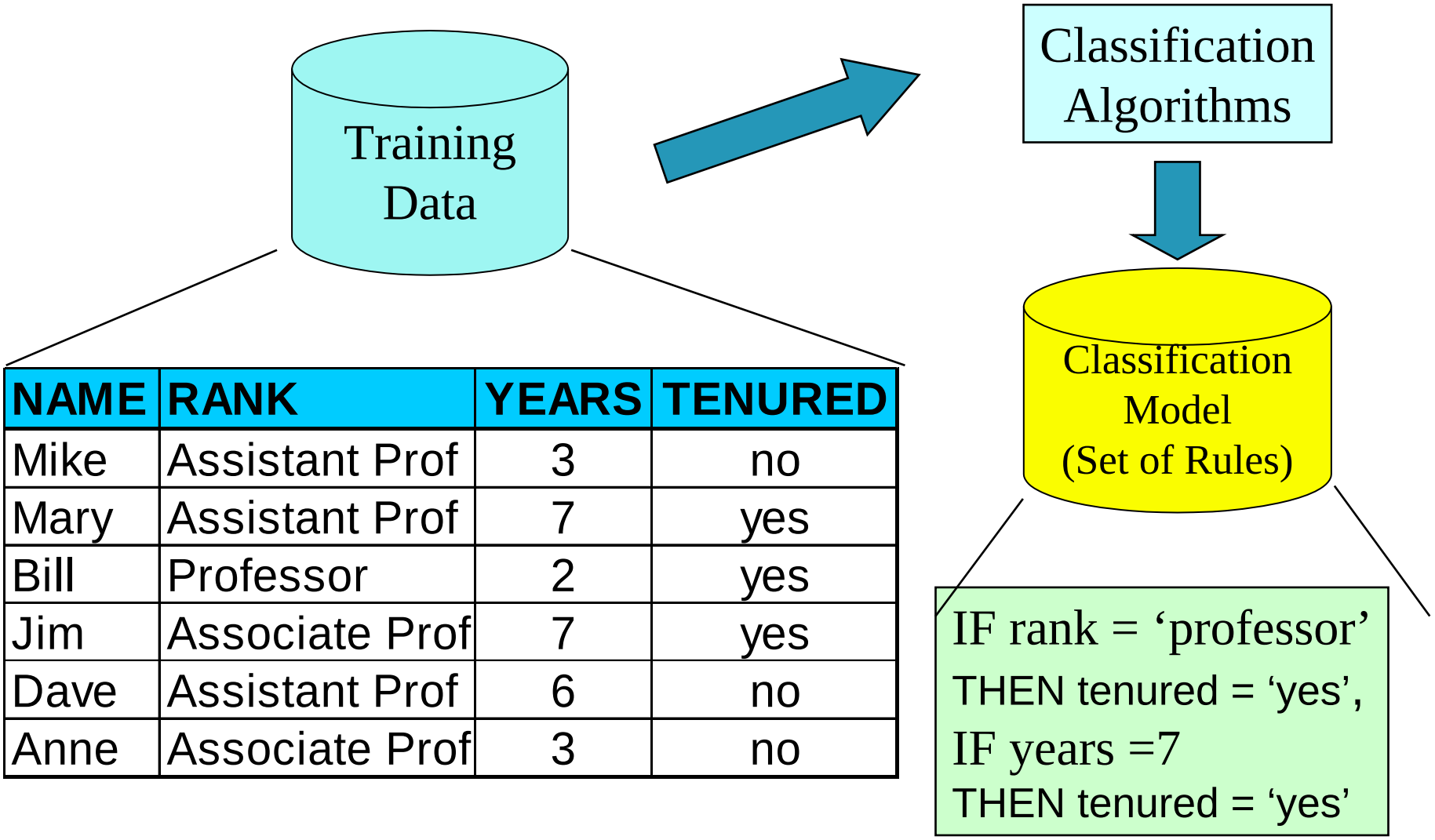
  When a classifier is build it predicts categorical class labels of new data – classifies unknown data. We also say that it predicts class labels of the new data

  Construction of the classifier (a model) is based on a training set in which the values of a decision attribute (class labels) are given and is tested on a test set

- Prediction

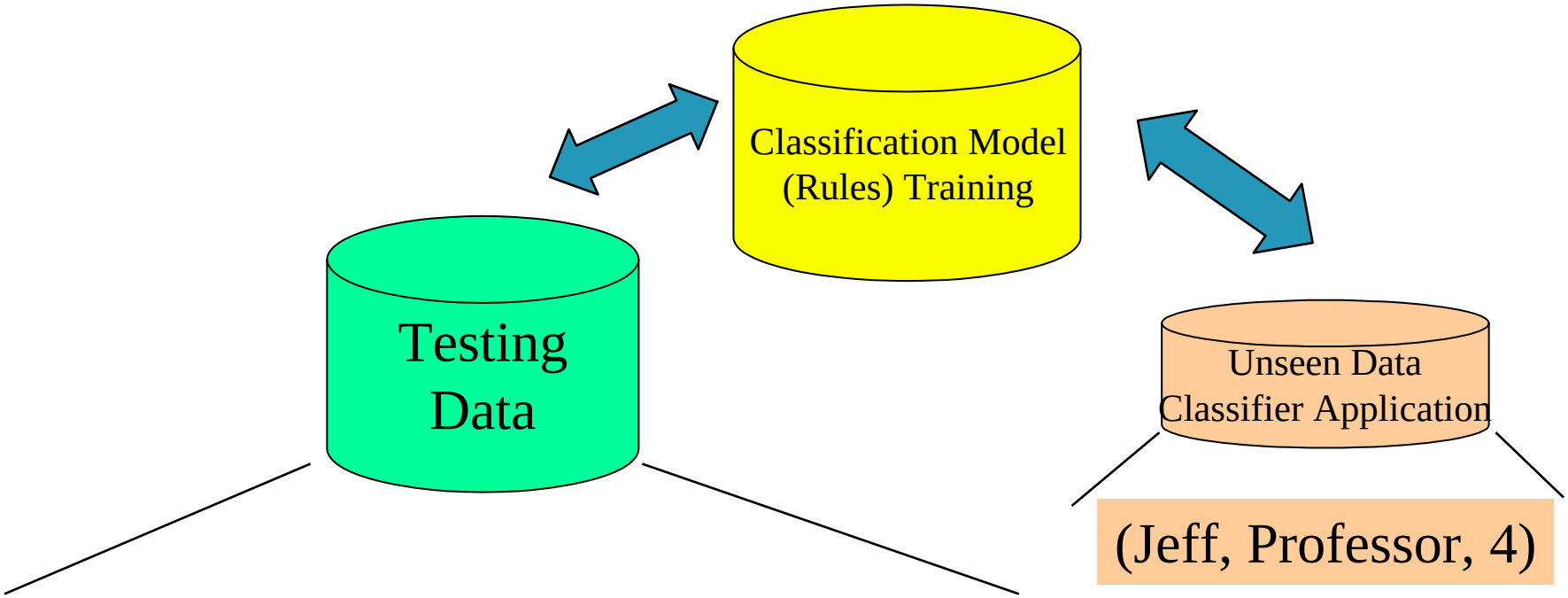
  Statistical method that models continuous-valued functions, i.e., predicts unknown or missing values

Classification Process : a Classifier Construction



NAME	RANK	YEARS	TENURED
Mike	Assistant Prof	3	no
Mary	Assistant Prof	7	yes
Bill	Professor	2	yes
Jim	Associate Prof	7	yes
Dave	Assistant Prof	6	no
Anne	Associate Prof	3	no

Testing and Prediction (by a classification algorthm)



NAME	RANK	YEARS	TENURED
Tom	Assistant Prof	2	no
Merlisa	Associate Prof	7	no
George	Professor	5	yes
Joseph	Assistant Prof	7	yes

Classifiers Predictive Accuracy

- PREDICTIVE ACCURACY of a classifier is a percentage of well classified data in the testing data set.

- **Predictive accuracy depends heavily on a choice of the test and training data.**

- There are many methods of choosing test and and training sets and hence evaluating the predictive accuracy. This is a separate field of research.

Predictive Accuracy Evaluation

The main methods of predictive accuracy evaluations are (see Testing Classifiers slides):

- Re-substitution (N ; N)
- Holdout (2N/3 ; N/3)
- x-fold cross-validation (N-N/x ; N/x)
- Leave-one-out (N-1 ; 1),

where N is the number of instances in the dataset (see separate presentaion)

- **The process of building and evaluating a classifier is also called a supervised learning, or lately when dealing with large data bases a classification method in Data Mining.**

Supervised vs. Unsupervised Learning (Data Mining book slide)

- **Supervised learning (classification)**

  Supervision: The training data (observations, measurements, etc.) are accompanied by labels indicating the class of the observations.

  New data is classified based on a tested classifier

Supervised vs. Unsupervised Learning

- **Unsupervised learning (clustering)**

  The class labels of training data is unknown

  We are given a set of records (measurements, observations, etc.)

  with the aim of establishing the existence of classes or clusters in the data