cse634
Data Mining

Chapter 6: Classification
Introduction

Professor Anita Wasilewska
Computer Science Department
Stony Brook University

)

Classification

- **PART 1:**
- Classification = Supervised Learning
- Building a Classifier

- **PART 2:** Classification Algorithms
  (Models, Basic Classifiers)

- **PART 3:** Classification by Association
- **PART 4:** Other Classification Methods

Part 1: Classification
Introduction

- Supervised learning = Classification
- Data format: **training** and **test** data
- **Class** definitions and **class** descriptions
- **Rules learned**: characteristic and discriminant
- Classification process = **building a classifier**
-

Part 1: Classification

- Supervised learning = Classification
- **Building a Classifier:**

  Training and Testing
- Evaluating predictive accuracy
- the most common methods
- Unsupervised learning= Clustering

Classification Algorithms
(Models, Basic Classifiers)

Part 2:

- **Decision Trees** (ID3, C4.5) –descriptive
- **Neural Networks-** statistical
- **Bayesian Networks** - statistical
- **Rough Sets** - descriptive
- **Genetic Algorithms** – descriptive or statistical- but mainly an optimization method

Part 3: **Classification by Association** - descriptive

Part 3: Other Classification Methods

- k-nearest neighbor classifier
- Case-based reasoning
- Support Vector Machines
- Fuzzy sets approaches

Classification Data Format

- **Classification Data Format:**
- a **data table** with **key attribute removed**
- A special attribute, called a **class attribute** must be **distinguished**
- **The values** of the **class attribute** are called **class labels**
- The **class labels** are **discrete-valued** and **unordered.**
- **Class attributes** are **categorical** in that each value serves as a **category**, or a **class**

Classification Data Format

- **The records** in the **classification data**
- are called **data tuples** with their **associated class labels**

- It means that we **distinguish** in a **record** its **attribute part** and **class part**

- **The attribute part** is called data tuple, or attribute vector, data vector, sample, example, instance, data point (with associate label)

Classification Data Example

- **Example: Data Table with class attribute C**
-

Rec	a1	a2	a3	a4	C
o1	1	1	m	g	c1
o2	0	1	v	g	c2
o3	1	0	m	b	c1

- **This data** consists of tuples (examples, instances):
- o1= (1, 1, m, g) with the **class label** c1
- o2= (0, 1, v, g) with the **class label** c2
- o3 =(1, 0, m, b) with the **class label** c1

Classification Data 1

- **Classification Data Format:** a data table with **key attribute removed**.
- **Special attribute**, called **a class attribute** 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

Classification Data 2

(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

Class definitions

- **Syntactically** a **class C** is **defined** by the

  class attribute **c** and its value **v**

- **Semantically a class** is **defined** as a **subset of records**

- A **description** of a **class C** **defined** by the class attribute **c** and its value **v** is written as : **c=v**

Classes Definition

- Given a class attribute **C** with attribute class values c1**,** c2**,** ⋯ ck
- **Semantically**, classes **C1, C2, ⋯ Ck** defined by the class values c1**,** c2**,** ⋯ ck **are sets of all records** for which the class attribute **C** has a value ci, respectively, i.e.

  C1 ={ r: **C=** c1}, **C2 =**{ r: **C** = c2}, **…..**

Class and Class Description

- **Example:**

Set of records **C** = { r1, r2, r6, r8, r14} of the classification **Data 2** on the previous slide is a

class defined by the **class attribute** buys_computer and its value **no**

The **class** **C** = { r1, r2, r6, r8, r14} **description** is: **buys_computer= no** because

C = {r: buys_computer= no }

C = { r1, r2, r6, r8, r14} is a **class defined** by the **class description** **buys_computer= no**

Class characteristics

Characteristics of a class **C ={ r: c=v}**

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

Characteristics of the class **C** are written as

a1=v1 & a2=v2&…..ak=vk

Class characteristics

REMARK

A class C can have many characteristics, i.e many characteristic descriptions

Different classes can have (and often have) the same characteristics

Characteristic Descriptions

Definition:

A formula **a1=v1 & a2=v2&…..ak=vk** is called **a characteristic description** for a class **C={ r: c= v }**

If and only if

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

i.e.

{r: a1=v1 & a2=v2&…..ak=vk } /\ {r: c= v } = not empty set

Characteristic Descriptions

Example: given classification Data 1, 2

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

of the class **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

Characteristic Descriptions

- **A formula**
- **Income=low** is a **characteristic description**

of the class C1 with description:

buys_computer= yes

and of the class *C2* with description:

buys_computer= no

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

of the class ***C2* = {r: buys_computer=no }**

because:

{ **r:** Age<=30 & Income=low } /\ **{r: buys_computer=no }= emptyset**

Characteristic Formula

Any formula of a form

IF class description THEN characteristics

is called **a characteristic formula**

Example: : given classification Data 1, 2

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

Characteristic Rule

- A characteristic formula:

IF class description 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: class description} /\{r: characteristics} = not emptyset

Classification Data 1

- **Classification Data Format:** a data table with **key attribute removed**.
- **Special attribute**, called **a class attribute** 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

Characteristic Rule

EXAMPLE: **:** given classification Data 1, 2

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, r16 }

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

and

{r1,r2, r6, r8, r16 } /\ {r6,r7} = not empty set

Characteristic Rule

EXAMPLE: : given classification Data 1, 2

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, r16 }
{r: income = low & credit=fair} **=** {r5, r9 }
and
{r1,r2, r6, r8, r16 } /\ {r5,r9} = **empty set**

Discrimination

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

If characteristics then class

- Example : given classification Data 1, 2
- **If** Age=<= 30 & income=high & student=no & credit_rating=fair **then** buys_computer= no

Discriminant Formula

Discriminant Formula Definition

A discriminant formula is any formula

If characteristics then class

- **Example: :** given classification Data 1, 2
- **IF** Age=>40 & inc=low **THEN** buys_comp= no

Discriminant Rule

- **Discriminant Rule Definition**
- A discriminant **formula**

If characteristics then class

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

If and only if

1. **{r: characteristic} is a non empty set**
2. **{r: characteristic}** $\sqsubset$ **{r: class}**

Discriminant Rule

- **Example: :** given classification Data 1, 2
- A discriminant **formula**

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

is NOT a discriminant rule *in our data base*

because

***{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 the **chracteristic 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 classes** with description buys_computer= no
- and buys_computer= yes
- (see records r7 and r8 in our Data 2)

Supervised Learning Goal (1)

- Given a data set and a **class C** defined in a given **classification dataset**
- **Supervised Learning Goal** is to
- **FIND** a minimal set (or as small as possible set) of **characteristic** and/or **discriminant** rules,
- **or other descriptions** of the **class C**, or of (all) other classes
- When we find **RULES** we talk about
- The **Descriptive Supervised Learning**

Supervised Learning Goal (2)

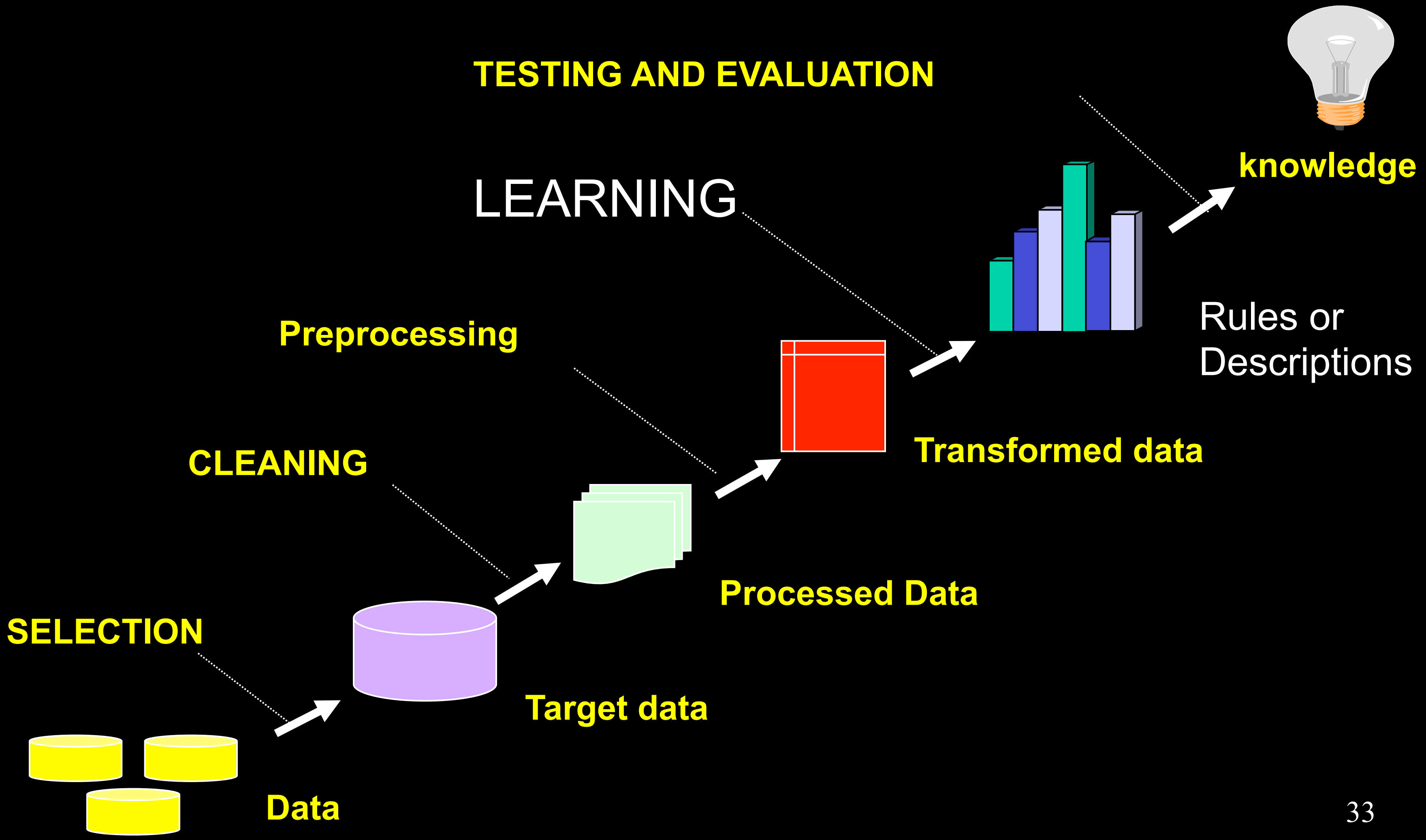
- We also want the **found rules** to **involve as few** attributes as it is possible

It means that we want **the rules** to have **as short** as possible length of the descriptions

Supervised Learning

- The process of **CREATING** (learning) discriminant and/or characteristic rules, **or other** descriptions and **TESTING** them is called a **supervised learning process**
- When the **process** (look at the Learning process slide) is **finished** we say that the **classification** has been learned and tested from examples (records in the classification dataset)
- It is called **supervised learning** because **we know the class labels** of all data examples

The Learning Process (LP)



Classification Data 1

- **Classification Data Format:** a data table with **key attribute removed**.
- **Special attribute**, called **a class attribute** 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

A small, full set DISCRIMINANT RULES for classes: 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"
  - **Exercise:** verify that they all are true in the Data1,2

Testing and Classifying

- In order to **use discovered rules** for **testing**, and later, when testing is **finished** and **predictive accuracy** is acceptable to use them for **future classification** we write rules in a following **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 a **new record x** are **matched** with the **IF** part of the rule and the **record** is **classified** accordingly to the **THEN** part of the rule

Testing and Training

- The **Test Dataset** has the same format as the **Training Dataset**, i.e.
- In both datasets the **values** of class attribute **are known**
- **Test Dataset** and **Training Dataset** are disjoint sets
- We use the **Test Dataset** to evaluate the **predictive accuracy** of our **discovered set of rules**

Predictive accuracy

- **PREDICTIVE ACCURACY** of the **set of rules**, or any other **result** of a classification algorithm is a **percentage** of well classified data in the **Test Dataset**

- If the **predictive accuracy** is not high enough we **chose** a different training and testing datasets and **start learning process again**

- There are **many methods** of training and testing and they will be discussed later

Classification Data

Classification Data Format: **a data table with key attribute removed .**

- **Special attribute, called a class attribute must be distinguished.**
- The values: **$c_1, c_2, \ldots c_n$** of the class atrribute C are called **class labels**
- **Exercise:** for the database below write **2 discriminant rules** and **3 characteristic rules** – and PROVE them to be what you claim

Obj	a1	a2	a3	a4	C
o1	1	1	m	g	c1
o2	0	1	v	g	c2
o3	1	0	m	b	c1

Classification and Classifiers

- **An algorithm** (model, method) is called **a classification algorithm**

- if it uses the classification data **to build a set of patterns:**
- **discriminant** and /or **characteristic rules**
- or other pattern descriptions

- These **patterns** are **structured** in such a way that **we can use** them to **classify** unknown sets of objects: unknown tuples, records

Classification and Classifiers

- For the reason that
- **we can use** discovered **patterns** to classify unknown sets of objects a **classification algorithm** is often called shortly **a classifier**

- **Remember** that the name **classifier** implies **more** than just a **classification algorithm**

- **A classifier** is a **final product** of a **process** that uses **data set** and a **classification algorithm**

Building a Classifier

- **Building** a classifier consists of two phases:

training and **testing**

In both phases we use

- **training data set** and **disjoint** with it
- **test data set** for both of which the **class labels** are **known for all** of the records

Building a Classifier

- 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 **test data**
- The **measure** for a **trained classifier** is called **predictive accuracy**

- **The classifier is build** i.e. we **terminate** the process if it has been **trained** and **tested** and the **predictive accuracy** is on an **acceptable level**

Classifiers Predictive Accuracy

- **PREDICTIVE ACCURACY** of a classifier is a percentage of well classified data in the **test data** set

- **PREDICTIVE ACCURACY depends heavily on a choice of the test and training data sets**

- There are **many methods of choosing test** and and **training sets** and hence evaluating the **predictive accuracy**
- **Basic methods** are presented in **Testing Classifiers** lecture

Correctly and Not Correctly Classified Records

- A **record** **is** **correctly** **classified** if and only if the following conditions hold:

(1) we **can classify** the record, i.e. **there is a rule** such that its LEFT side **matches** the record,

(2) **classification** **determined by the rule** is **correct**, i.e. the RIGHT side of the rule **matches** the value of the record's class attribute

OTHERWISE

- **the record is not correctly classified**
- Words used:
- not correctly = incorrectly = misclassified

Exercise 1

- Assume that we have a following set of rules:
- R1: a1=1 /\ a2= 0 => class= yes
- R2: a1=0 /\ a2=3 => class=no
- R3: a2=1 => class=yes
- The **TEST data** has the following 6 records, where the attributes are a1, a2, class
- r1 = (1, 0)- record, (yes) associated class label,
- r2 = (0, 3) (yes), r3 = (1, 1) (no),
  r4 = (2, 1) (yes), r5= (3, 1) (yes), r6 = (1, 2) (no)

WRITE the rules in **predicate form** and

CALCULATE the **Predictive Accuracy** of this set of rules with respect to the above **TEST data** of **6 records** above

Exercise 2

- Evaluate the **Predictice Accuracy** of the set of rules:
  - R1: **IF** *age* = “<=30” AND *student* = “*no*” **THEN**
  - *buys_computer* = “*no*”
  - R2: **IF** *age* = “<=30” AND *student* = “*yes*” **THEN**
  - *buys_computer* = “*yes*”
  - R3: **IF** *age* = “31…40” **THEN** *buys_computer* = “*yes*”
  - R4: **IF** *age* = “>40” AND *credit_rating* = “*excellent*” **THEN** *buys_computer* = “no”
  - R5: **IF** *age* = “<=30” AND *credit_rating* = “*fair*” **THEN** *buys_computer* = “yes”
  - with respect to the **TEST data** on the next slide .
  - REMARK: you must FIRST **re-write the rules in predicate form**

TEST DATA for Example 2

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

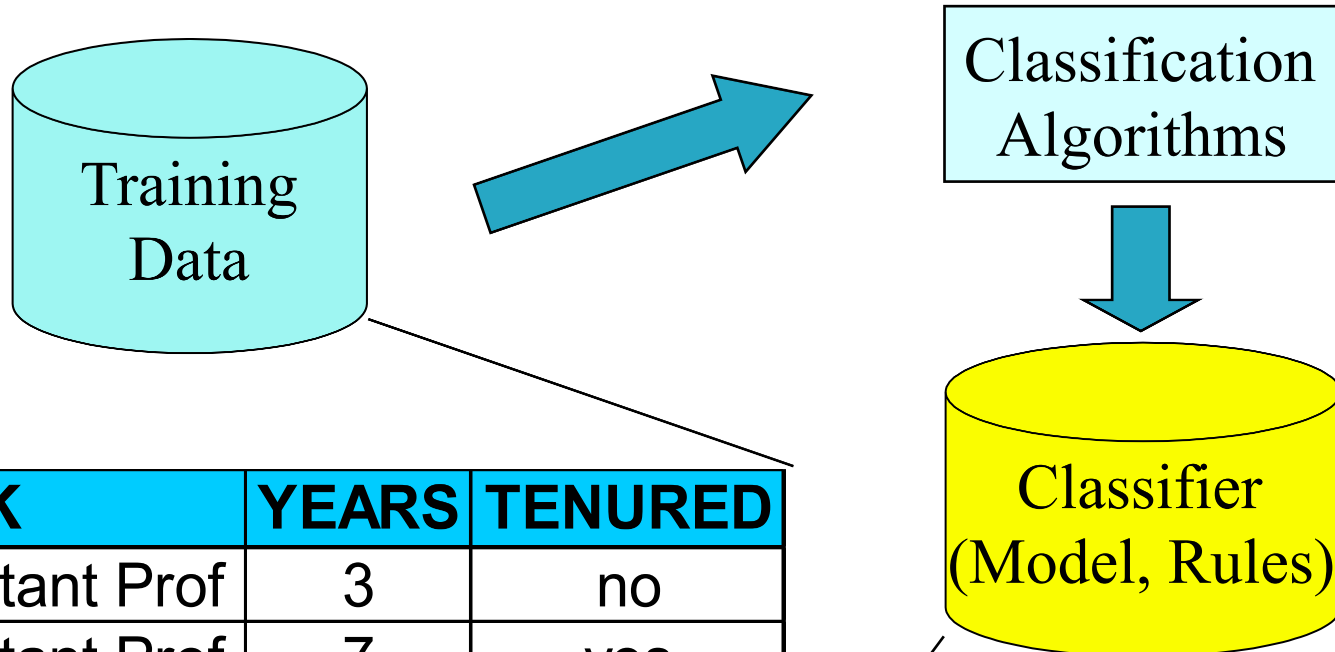
Predictive Accuracy

- For our 10 **TEST records** and 5 rules R1, R2 … R5
- Record r1 is well classified by rule R5
- Record r2 is misclassified
- Record r3 is well classified by rule R5
- Record r4 is well classified by rule R5
- Record r5 is misclassified
- Record r6 is misclassified
- Record r7 is well classified by rule R3
- Record r8 is well classified by rule R1
- Record r9 is well classified by rule R3
- Record r10 is misclassified
- We **have 6 correctly classified** records out of **10**
- **Predictive accuracy is 60%**

- **Exercise:** prove that rules R1, R2 … R5 are TRUE in the Classification Data 1, 2

Classification Process : a Classifier

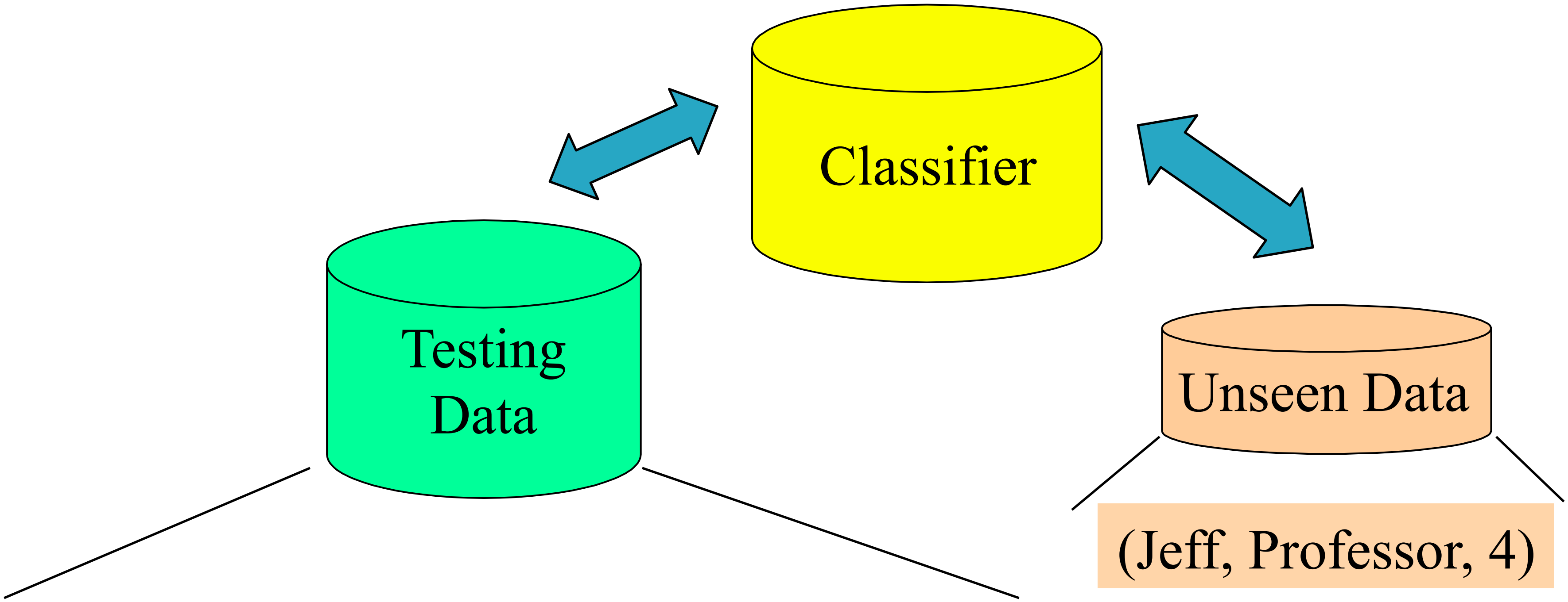
Book slide



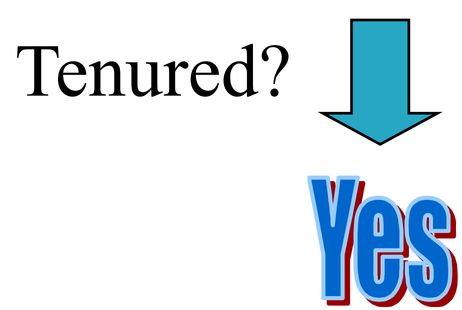
NAME	RANK	YEARS	TENURED
Mike	Assistant Prof	3	no
Mary	Assistant Prof	7	yes
Bill	Professor	2	yes
Jim	Associate Prof	7	yes
Dave	Assistant Prof	6	no
Anne	Associate Prof	3	no

IF rank = 'professor'
THEN tenured = 'yes'
IF years > 6,
THEN tenured = 'yes'

Testing and Prediction
Book Slide



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



Supervised vs. Unsupervised Learning

- **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 are 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