Cse352
Artificial Intelligence
Short Review for Midterm

Professor Anita Wasilewska
Computer Science Department
Stony Brook University
Midterm

- Midterm INCLUDES Lectures 1-11 and Homeworks 1,2, Resolution
- LOGIC
- PRODUCTION SYSTEMS
- PROPOSITIONAL RESOLUTION (material you needed for Q1)
- CLASSIFICATION
- CLASSIFICATION by Decision TREES
LOGIC: Restricted Domain Quantifiers

Translation for existential quantifier

\[ \exists_{A(x)} B(x) \equiv \exists x (A(x) \land B(x)) \]

↑ restricted  ↑ logic, not restricted

Example (mathematical formulas):
\[ \exists x \neq 1 (x > 0 \Rightarrow x + y > 5) \] - restricted
\[ \exists x ((x \neq 1) \land (x > 0 \Rightarrow x + y > 5)) \] - not restricted

↑ B(x, y)

English statement:
Some students are good.

Logic Translation (restricted domain):

\[ \exists_{S(x)} G(x) \]

Predicates are:
S(x) – x is a student
G(x) – x is good

TRANSLATION:
\[ \exists x (S(X) \land G(x)) \]
LOGIC: Restricted Quantifiers

Translation for universal quantifier

Restricted Logic (non-restricted)

$$\forall_{A(x)} B(x) \equiv \forall x (A(x) \Rightarrow B(x))$$

Example (mathematical statement)

$$\forall x \in \mathbb{N} \ (x = 1 \lor x < 0)$$ restricted domain

$$\equiv \forall x (x \in \mathbb{N} \Rightarrow (x = 1 \lor x < 0))$$ non-restricted
Translations to Logic

Rules:
1. **Identify** the domain: always a set $X \neq \phi$
2. **Identify** predicates (simple: atomic)
3. **Identify** functions (if needed)
4. **Identify** the connectives $\lor$, $\land$, $\Rightarrow$, $\neg$, $\Leftrightarrow$
5. **Identify** the quantifiers $\forall x$, $\exists x$ or Restricted Quantifiers $\forall P(x)$, $\exists Q(x)$
6. Write a formula using only symbols for 2, 3, 4, 5
   Use restricted domain quantifier translation rules, where needed to write
7. Write LOGIC formula – formula without Restricted Quantifiers
Translation to Logic

- **Translations from Natural Language**
- **Translate: “No house is red”**

1. Domain:  $X \neq \emptyset$
2. Predicates:  $A(x) – x$ is a House  $B(y) – y$ is red
3. Functions:  (none)
4. Connectives:  $\neg$ – “not”
5. Quantifiers:  $\exists_{A(x)} – “some houses”$ (restricted)
6. RESTRICTED FORMULA:  $\neg \exists_{A(x)} B(x)$
7. LOGIC FORMULA:  $\neg \exists x ( A(x) \land B(x) )$
Translations from Natural Language

BE CAREFUL!

YOU MUST ALWAYS DO DIRECT TRANSLATION

Never translate some logically EQUIVALENT statements such as

“All houses are not red”

instead of (via de Morgan Laws)

“No house is red”
Translations Example

Translate: Some patients like all doctors

Predicates:
P(x) – x is a patient in the domain \( X \neq \phi \), i.e. \( x \in X \)
D(y) – y is a doctor in the domain \( X \neq \phi \), i.e. \( y \in X \)
L( x,y ) – x likes y in the domain \( X \neq \phi \), i.e. \( x,y \in X \)

There is a patient( x ), such that for all doctors( y ), x likes y

\[ \exists_{P(x)} \forall_{D(y)} L( x,y ) \]

Non-restricted quantifiers

\[ \exists x(P(x) \land \forall y(D(y) \Rightarrow L( x,y ))) \]
Learning Process

• Short Questions:

• Describe all stages of the Learning Process

• Describe the role of Preprocessing stage and its main methods

• Discuss the Learning Proper stage
Learning Models

• Describe what is Descriptive/ non-Descriptive Learning

• List some Descriptive/ non-Descriptive Models

• How and what decides which type of Learning is the best to use (implement)

• Give examples of types of applications and the best Models (algorithms) for them
Classification

• Describe what is CLASSIFICATION; type of data, goals and applications

• Describe all stages of the classification process

• Describe and discuss basic classification Models and their differences
Classification

• Discuss the **Decision Tree Induction** and its strengths and weaknesses

• Define a **CLASSIFIER**

• Describe a process of **building a CLASSIFIER**
Given a classification dataset \( DB \) with a set
\[ A = \{a_1, a_2, \ldots, a_n\} \]
of attributes and a class attribute \( C \)
with values
\[ \{c_1, c_2, \ldots, c_k\} \]
- \( k \) classes

**Definition 1**
Any expression \( a_1 = v_1 \& \ldots \& a_k = v_k \) where \( a_i \in A \)
and \( v_i \) are corresponding values of attributes from \( A \)
is called a **DESCRIPTION**

Any expression \( C = c_i \) is for \( c_i \in \{c_1, c_2, \ldots, c_k\} \)
Is called a **CLASS DESCRIPTION**
Classification Data and Rules

Definition 2

A **CHARACTERISTIC FORMULA** is any expression

\[ C = c_k \implies a_1 = v_1 \& \ldots \& a_k = v_k \]

We write it as

\[ \text{CLASS} \implies \text{DESCRIPTION} \]

Definition 3

A **DETERMINANT FORMULA** is any expression

\[ a_1 = v_1 \& \ldots \& a_k = v_k \implies C = c_k \]

We write it as

\[ \text{DESCRIPTION} \quad \text{CLASS} \]
Classification Data and Rules

Definition 4

A characteristic formula

\[
\text{CLASS } \Rightarrow \text{DESCRIPTION}
\]

is called a \text{CHARACTERISTIC RULE} of the classification dataset \text{DB}

iff

it is \text{TRUE} in \text{DB}, i.e. when the following holds

\[
\{o: \text{DESCRIPTION}\} \setminus \{o: \text{CLASS}\} \text{ is a non-empty set}
\]

Where

\[
\{o: \text{DESCRIPTION}\}
\]

is the set of all records of \text{DB} corresponding to the \text{DESCRIPTION}

\[
\{o: \text{CLASS}\}
\]

is the set of all records of \text{DB} corresponding to the \text{CLASS}
Definition 5
A discriminant formula

\[ \text{DESCRIPTION} \Rightarrow \text{CLASS} \]

is called a **DISCRIMINANT RULE** of DB

iff

it is TRUE in DB, i.e. the following conditions hold

1. \{o: DESCRIPTION\} is a non-empty set

2. \{o: DESCRIPTION\} is included in \{o: CLASS\}
PROBLEM 1

Prove that for any classification data base DB and any of its DISCRIMINANT RULES of the form

\[ \text{DESCRIPTION} \Rightarrow \text{CLASS} \]

the formula

\[ \text{CLASS} \Rightarrow \text{DESCRIPTION} \]

is a CHARACTERISTIC RULE of the DB
By definition 5, for any database DB:

\[ \text{DESCRIPTION} \Rightarrow \text{CLASS} \]

is a **DISCRIMINANT RULE** iff

1. \( \{o: \text{DESCRIPTION}\} = \) not empty set

2. \( \{o: \text{DESCRIPTION}\} \) is included in \( \{o: \text{CLASS}\} \)

Therefore,

\[ \{o: \text{DESCRIPTION}\} \setminus \{o: \text{CLASS}\} = \) not empty set

and by Definition 4

\[ \text{CLASS} \Rightarrow \text{DESCRIPTION} \]

Is the **CHARACTERISTIC RULE**
PROBLEM 2

Given a dataset:

<table>
<thead>
<tr>
<th>Record</th>
<th>a1</th>
<th>a2</th>
<th>a3</th>
<th>a4</th>
<th>C</th>
</tr>
</thead>
<tbody>
<tr>
<td>O1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>O2</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>O3</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>O4</td>
<td>0</td>
<td>0</td>
<td>2</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>O5</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
</tbody>
</table>

Find the set \{o :DESCRIPTION\} for the following descriptions:

1) $a1 = 2 \& a2 = 1$
2) $a3 = 1 \& a4 = 0$
3) $a2 = 0 \& a3 = 2$
4) $c=1$
5) $c=0$
Problem 2 Solution

Find the set \( \{o : \text{DESCRIPTION}\} \) for the following descriptions

1) \( a_1 = 2 \) & \( a_2 = 1 \)  
   Answer: \( \{o_1\} \)

2) \( a_3 = 1 \) & \( a_4 = 0 \)  
   Answer: \( \{o_1, o_5\} \)

3) \( a_2 = 0 \) & \( a_3 = 2 \)  
   Answer: \( \{o_4\} \)

4) \( c = 1 \)  
   Answer: \( \{o_1, o_5\} \)

5) \( c = 0 \)  
   Answer: \( \{o_3, o_5\} \)
PROBLEM 3

For the following formulae use proper definitions to determine (it means prove) whether they are / are not DISCRIMINANT / CHARACTERISTIC RULES of our dataset.

6) \( a_1 = 1 \) & \( a_2 = 1 \) \( \Rightarrow \) \( C = 1 \)

7) \( C = 1 \) \( \Rightarrow \) \( a_1 = 0 \) & \( a_2 = 1 \) & \( a_3 = 1 \)

8) \( C = 2 \) \( \Rightarrow \) \( a_1 = 1 \)

9) \( C = 0 \) \( \Rightarrow \) \( a_1 = 1 \) & \( a_4 = 0 \)

10) \( a_1 = 2 \) & \( a_2 = 1 \) & \( a_3 = 1 \) \( \Rightarrow \) \( C = 0 \)

11) \( a_1 = 0 \) & \( a_3 = 2 \) \( \Rightarrow \) \( C = 1 \)
PROBLEM 3 SOLUTION

For the following formulae use proper definitions to determine (it means prove) whether they are / are not DISCRIMINANT / CHARACTERISTIC RULES of our dataset.

6) \( a_1 = 1 \& a_2 = 1 \Rightarrow C = 1 \)
   \{o1\} is a subset of \{o1, o5\} so this is a DISCRIMINANT rule

7) \( C = 1 \Rightarrow a_1 = 0 \& a_2 = 1 \& a_3 = 1 \)
   \{o: a_1 = 0 \& a_2 = 1 \& a_3 = 1\} is an empty set so this is not a CHARACTERISTIC rule

8) \( C = 2 \Rightarrow a_1 = 1 \)
   As the intersection is empty so this is not a CHARACTERISTIC rule

9) \( C = 0 \Rightarrow a_1 = 1 \& a_4 = 0 \)
   \{o3 , o4\} \(\setminus\) \{o5\} is empty set so this is not a CHARACTERISTIC rule

10) \( a_1 = 2 \& a_2 = 1 \& a_3 = 1 \Rightarrow C = 0 \)
    \{o5\} is not a subset of \{o3 , o4\}, so this is not a DISCRIMINANT rule

11) \( a_1 = 0 \& a_3 = 2 \Rightarrow C = 1 \)
    \{o4\} is not a subset of \{o1 , o5\}, so this is not a DISCRIMINANT rule
Classification

• Describe what is Classification; which is the goal, what data one needs etc....
• Describe all stages of the Classification Process
• Describe basic methods of training and testing
• Describe the process of building a CLASSIFIER
• What is a CLASSIFIER?
### Problem: Classification by DTREE

1. Use the data below to build a CLASSIFIER by basic DTREE algorithm.
2. Use 2 different testing methods of your choice and compare the results.

#### CLASSIFICATION DATA

<table>
<thead>
<tr>
<th>Record</th>
<th>A1</th>
<th>A2</th>
<th>C</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>4</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>5</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>6</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>7</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>8</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
</tbody>
</table>
Data Mining Process

1. **Data**
   - **Selection**
   - **Cleaning**
   - **Preparation**

2. **Processed Data**
   - **Data Mining (proper)**

3. **Transformed Data**
   - **Interpretation and Evaluation**

4. **Knowledge**
   - **Rules, Patterns, Models**
Preprocessing stage

• **Preprocessing:**
  • includes all the operations that have to be performed before a learning algorithm is applied

• **Data in the real world is dirty**: incomplete, noisy and inconsistent.

• **Quality decisions** must be based on quality Data.
Preprocessing stage

• **Data cleaning**
  – Fill in missing values, smooth noisy data (binning, clustering, regression), identify or remove outliers, and resolve inconsistencies

• **Data integration**
  – Integration of multiple databases, data cubes, or files
Preprocessing stage: Data transformation

- Data reduction and attribute selection
  - Obtains reduced presentation in volume but produces the same or similar analytical results (stratified sampling, PCA, cluster)

- Data discretization
  - Part of data reduction but reduces the number of values of the attributes by dividing the values of attributes into intervals (segmentation by natural partition, hierarchy generation) or disjoint subsets (binning)

- Normalization and aggregation
Learning Proper

• **Learning proper** is a step in the LEARNING process in which algorithms are applied to obtain patterns in data.

• It can be re-iterated- and usually is
Descriptive / non descriptive models

• Statistical - descriptive

• **Statistical** data mining uses historical data to predict some unknown or missing numerical values

• **Descriptive** data mining aims to find patterns in the data that provide some information about what the data contains

• often presents the knowledge as a set of rules of the form **IF.... THEN**...
Models

- **Descriptive**: Decision Trees, Rough Sets, Association, Classification by Association
- **Statistical**: Neural Networks, Bayesian Networks, Cluster, Outlier analysis, Trend and evolution analysis
- **Optimization method**: Genetic Algorithms – can be descriptive
Classification

• **Classification:**
  - Finding models (rules) that describe (characterize) or/ and distinguish (discriminate) classes or concepts for future prediction

• **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 attribute $C$ are called **class labels**
  - The class label attributes are discrete valued and unordered.
Classification

• **Goal:**

• **FIND** a minimal set of characteristic and/or discriminant rules, or **other descriptions** of the class C, or all, or some other classes

• We also want the found rules to involve as **few attributes** as it is possible
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 the **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**.
Classification Stages

- **Stage 1**: build the basic patterns structure - training
- **Stage 2**: optimize parameter settings; can use (N:N) re-substitution - parameter tuning
- Re-substitution error rate = training data error rate
- **Stage 3**: use test data to compute - predictive accuracy/error rate – testing
- **Stage 4**: finish building the classifier
Decision Tree

• **DECISION TREE**

• A flow-chart-like tree structure;

• **Internal node** denotes an **attribute**;

• **Branch** represents the **values** of the node attribute;

• **Leaf nodes** represent **class labels**
DT Basic Algorithm

- The **basic DT algorithm** for decision tree construction is a greedy algorithm that constructs decision trees in a top-down recursive divide-and-conquer manner.

- Tree **STARTS** as a single node representing all training dataset (data table with records called samples).

- **IF** the samples (records in the data table) are all in the same class, **THEN** the node becomes a leaf and is labeled with that class.

- The algorithm uses the same **process recursively** to form a **decision tree** at each partition.
DT Basic Algorithm

• The recursive partitioning **STOPS** only when any one of the following conditions is TRUE

• **1. All records** (samples) for the given node belong to the same class

• **2. There are no remaining attributes** on which the samples (records in the data table) may be further partitioned – a **LEAF** is created with **majority vote** for training sample

• **3. There is no records** (samples) left – a **LEAF** is created with **majority vote** for training sample

• **Majority voting** involves converting node N into a leaf and labeling it with the most common class in D which is a set of **training** tuples and their associated class labels
Attribute Selection Measures

• **Some Heuristics:**

• **DTree:** some Attribute Selection Measures

• **Information Gain, Gini Index**

• We use them for selecting the attribute that “best” discriminates the given tuples according to class
Rule Based Systems REVIEW
Exercise

• **Exercise 1**
• Here are three simple *expert rules*
• **R1:** If your savings are small, then don’t invest in stocks
• **R2:** If you have no children and large income, then invest in stocks
• **R3:** If you have children and small income, then invest in savings
Exercise 1

• Conceptualize rules R1, R2, R3 in Predicate Form using predicates
  attribute(x, value of attribute)
  attribute(object, value of attribute)

  WRITE a format of a database TABLE needed for your conceptualization

REMARK: In order to express the rules Predicate Form, we must first define appropriate attributes and their values
Exercise 1

We have the following ATTRIBUTES:

- **Savings**
  Values: small, large

- **Income**
  Values: small, large

- **InvestStocks**
  Values: yes, no

- **InvestSavings**
  Values: yes, no

- **Children**
  Values: yes, no
Exercise 1: Predicate Form Conceptualization

Data Table Example

<table>
<thead>
<tr>
<th>Records</th>
<th>Savings</th>
<th>Income</th>
<th>InvesrStocks</th>
<th>InvestSavings</th>
<th>Children</th>
</tr>
</thead>
<tbody>
<tr>
<td>O₁</td>
<td>small</td>
<td>small</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
</tr>
<tr>
<td>O₂</td>
<td>large</td>
<td>small</td>
<td>no</td>
<td>no</td>
<td>no</td>
</tr>
<tr>
<td>O₃</td>
<td>small</td>
<td>large</td>
<td>yes</td>
<td>yes</td>
<td>no</td>
</tr>
</tbody>
</table>
Exercise 1: Rules in Predicate Form

• RULES:

• **R1**: Savings(x, small) \(\rightarrow\) InvestStock (x, no)

• **R2**: Children(x, no) \(\land\) Income(x, large) \(\rightarrow\) InvestStocks(x, yes)

• **R3**: Children(x, yes) \(\land\) Income(x, small) \(\rightarrow\) InvestSavings(x, yes)
Exercise 2

• Exercise 2
• The initial database has the following FACTS
• F1: Savings(John, small)
• F2: Children(John, no)
• F3: Income(John, large)
• 1. Are these FACTS true in Exercise 1 Data Table for a record o = John?
• 2. Design a Data Table 2 in which the above FACTS are true
• 3. Can you deduce InvestStocks(John, yes) on the base of the Data Table 2
Part 3: Exercise 3

- Given rules from Exercise 1:
  - R1: If your savings are small, then don’t invest in stocks
  - R2: If you have no children and large income, then invest in stocks
  - R3: If you have children and small income, then invest in savings
Exercise 3

- Conceptualize rules **R1, R2, R3**

In *Propositional Logic* in two ways:

1. Rules admit **only** atomic formulas; i.e. rules are built from propositional variables only – call the set of rules **PR1**

2. Rules admit **atomic formulas** and **negation** of atomic formulas – call obtained set of rules **PR2**
Exercise 3

• Write initial databases B1 and B2 of facts corresponding to the facts F1, F2, F3 from Exercise 2 for
  • (1) propositional conceptualization 1.
  • (2) propositional conceptualization 2.
  • (3) use corresponding rules from sets PR1, PR2 to deduce all facts from B1 and B2, respectively

Use Conflict Resolution from Busse Handout