Relational Data Mining: A Quick Introduction

Sašo Džeroski
Institut Jožef Stefan, Ljubljana
Saso.Dzeroski@ijs.si

Overview

- Data mining in a nutshell
  - Data mining tasks
  - Patterns and models
- The single table assumption
- First-order/Relational
  - rule induction
  - tree induction
  - association rules
  - IBL and distance-based clustering

Knowledge Discovery in Databases

- The process of extracting useful knowledge from data
- Involves identifying valid, novel, and useful patterns in data
- The patterns should be ultimately understandable

Data Mining

- Data mining is one step in the KDD process
- It is concerned with finding patterns in data,
  i.e., applying specific algorithms
  for extracting patterns from data
- Significant pre-processing and post-processing is required for data mining results to be useful

Data: A single database table

People

Person	Age	Sex	Income	Customer
Ann Smith	32	F	10000	yes
Joan Dew	53	F	1000000	yes
Mary Stew	27	F	20000	no
Jane Brown	55	F	20000	yes
Bob Smith	30	M	100000	yes
Jack Brown	50	M	200000	yes

Data Mining Tasks

- Predictive modeling: predict a field (class) from some other fields (attributes)
  - classification: predicted field is discrete
  - regression: predicted field is numeric
- Subgroup discovery (given target variable, find groups where its value has unusual distribution)
- Clustering: separate a set of records into subsets, so that records in a subset are similar to each other
- Finding frequent patterns and association analysis

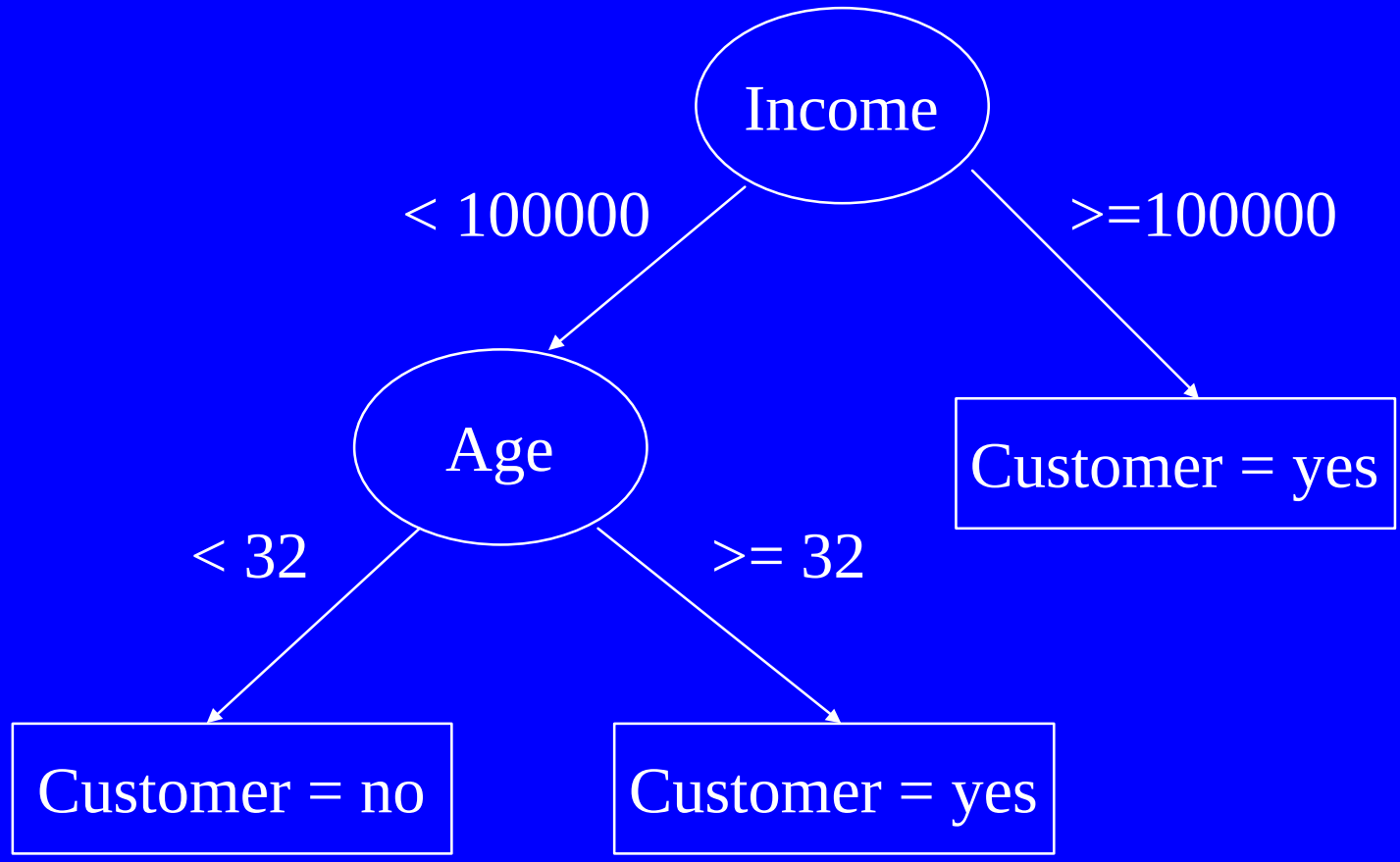
Patterns and models

- Equations and discriminants (linear and non-linear)
- Trees: classification and regression
- Rules: classification and regression
- Frequent itemsets and association rules
- Probabilistic models (networks)
- Partitions of the instance space / clusters

Equations/discriminants

- IF Income + Age - 20027 > 0
  THEN Customer = yes
  ELSE Customer = no

Classification trees



Classification rules

- IF Income >= 100000
  THEN Customer = yes
  (Support = 3; Confidence = 100%)
- IF Sex = F
  AND Age >= 32
  THEN Customer = yes
  (Support = 2; Confidence = 100%)

Frequent itemsets

- Customer=yes; Income >= 20000 (5)
- Sex=F (4)
- Income >= 100000;
  Income >=100000 AND Customer=yes (3)
- Income =< 20000 AND Age > 27;
  Income =< 20000 AND Age > 27 AND
  Sex = F AND Customer = yes (2)

Association rules

- IF Income =< 20000

  AND Age > 27
  THEN Sex = F
  AND Customer = yes
  (Support: 2; Confidence: 100%)
- The typical example:
  IF beer AND coke
  THEN potato chips AND peanuts

Probabilistic models

- Naïve Bayes: Customer -> Sex;
  Customer->Income; Customer-> Age
- Customer (yes: 5/6; no:1/6)
- Customer | Sex
  - Sex=M (yes: 1; no: 0)
  - Sex=F (yes: ¾; no: ¼)
- Bayesian networks

Distance-based methods

- Clustering
  - Hierarchical agglomerative/divisive
  - K-means; k-medoids
- Prediction
  - Nearest neighbor
  - K-Nearest neighbor methods

The single table assumption

- Most data mining methods work on a single table: we have to put all the data in one
- For one-to-one and many-to-one relations, we can join in the extra fields to the original relation without problems
- For one to many relations, problems occur
  - either loss of meaning or
  - loss of information through aggregation

A tale of two tables

Client number	Date of purchase	Magazine purchased
2303	04-15-94	car
2303	06-21-93	music
2309	05-30-92	comic
2313	11-11-11	sports
2319	11-11-11	house

Client number	Name	Address	Age
2303	Jones	1 Elm st.	35
2309	Smith	2 Oak dr.	27
2313	King	3 Low rd.	52

Data Integration

Join data from several tables

- One-to one relationships (extra information on age, ...)

Client number	Date of purchase	Magazine purchased	Age
2303	04-15-94	car	35
2303	06-21-93	music	35
2309	05-30-92	comic	27
2313	NULL	sports	52
2303	NULL	house	35

Data Integration(ctd)

Aggregate data from several tables

- One-to-many relationships (e.g. number of purchases)

Client number	Last purchase	Number of mags.	Age
2303	04-15-94	3	35
2309	05-30-92	1	27
2313	NULL	1	52

(Multi-)Relational Data Mining

- DM: Finding patterns in data
- (M)RDM: Finding patterns in (multi-)relational data; Note that relational databases are really multi-relational databases
- Patterns involving multiple relations are typically expressed in relational/first-order logic
  - this is more powerful than formalisms used by single table data mining methods
    - variables
    - recursion

Database and FOL/LP terms

- Relation name
- Attribute of relation
- Tuple (a1, …, an) of relation p
- Relation p as set of tuples
- Relation p defined as a view

- Predicate symbol
- Argument of predicate
- Ground fact p(a1, …,an) of predicate p
- Predicate p defined extensionally
- Predicate p defined intensionally

A database with two relations

People

Person	Age	Sex	Income	Customer
Ann Smith	32	F	10000	yes
Joan Gray	53	F	1000000	yes
Mary Stew	27	F	20000	no
Jane Brown	55	F	20000	yes
Bob Smith	30	M	100000	yes
Jack Brown	50	M	200000	yes

Marriages

Husband	Wife
Bob Smith	Ann Smith
Jack Brown	Jane Brown

Relational Data Mining Example

- IF People(Person,Income,Age,Sex,Customer)
  AND Income >= 100000
  THEN Customer = yes
- IF People(Person,Income,_,_,_)
  Income >= 100000,
  AND Marriages(Person,Wife)
  AND People(Wife,_,_,_,Customer)
  THEN Customer = yes

Rule induction

- Most common form of ILP
- Typically two classes + and -
- Rules are clauses in first-order logic
  - grandfather(X,Y) :- father(X,Z), parent(Z,Y)
- Many rule induction systems exist
  - FOIL (Quinlan 1990)
  - PROGOL (Muggleton 1995)

An example database

mother(blaguna,saso).
mother(blaguna,sonja).

father(veljo,saso).
father(veljo,sonja).

parent(blaguna,saso).
parent(blaguna,sonja).
parent(veljo,saso).
parent(veljo,sonja).

male(veljo).
male(saso).

female(blaguna).
female(sonja).

human(veljo).
human(saso).
human(blaguna).
human(sonja).

ILP - Rule induction

- Rules define relations in terms of other relations

  parent(X,Y) :- mother(X,Y)

  parent(X,Y) :- father(X,Y)

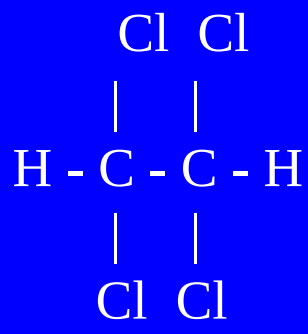
- Rules can also express constraints about relations

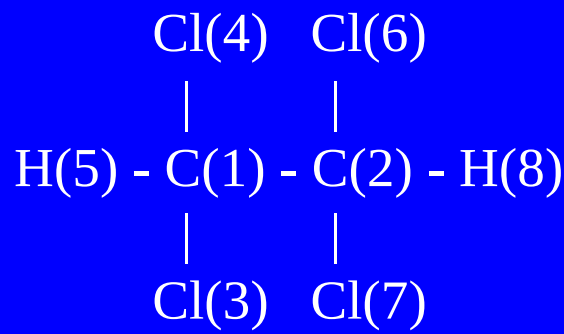
  female(X) :- mother(X,Y)

  male(X) :- father(X,Y)

  IF X is the mother of Y, X must be female

A biodegradability example





Representing compounds

in the atom-bond formalism

atom(AtomID,Element, AtomType,Charge)

bond(Atom1,Atom2, BondType).

slow.
atom(1,c,10,0.388).
atom(2,c,10,0.388).
atom(3,cl,93,-0.212).
atom(4,cl,93,-0.212).
atom(5,h,3,0.037).
atom(6,cl,93,-0.213).
atom(7,cl,93,-0.213).
atom(8,h,3,0.037).

bond(1,2,1).
bond(1,3,1).
bond(1,4,1).
bond(1,5,1).
bond(2,6,1).
bond(2,7,1).
bond(2,8,1).

Biodegradability rules

slow OR moderate :-

atom(A1,Elem1,Type1,Charge1), Elem1 = n, Charge1 < 0.8

- The biodegradability rate of the compound is slow or moderate if the compound contains a nitrogen atom of charge less than 0.8

slow OR moderate :-

atom(A1,Elem1,Type1,Charge1), Type1 = 1,bond(A5,A6,7)

- The biodegradability rate of the compound is slow or moderate if the compound contains a hydrogen atom (Type = 1) and an aromatic bond (7)

Background knowledge for chemical compounds

- An example: carbon rings
  - Benzene ring: benzene(RingList)
  - Carbon six ring (non aromatic): carbon_6_ring(RingList)
  - Carbon five aromatic ring: carbon_5_aromatic_ring(RingList)
  - Carbon five ring (non aromatic): carbon_5_ring(RingList)

```
benzene([C1,C2,C3,C4,C5,C6]) :-
    bond(C1,C2,7), bond(C2,C3,7), bond(C3,C4,7),
    bond(C4,C5,7), bond(C5,C6,7), bond(C6,C1,7).
```

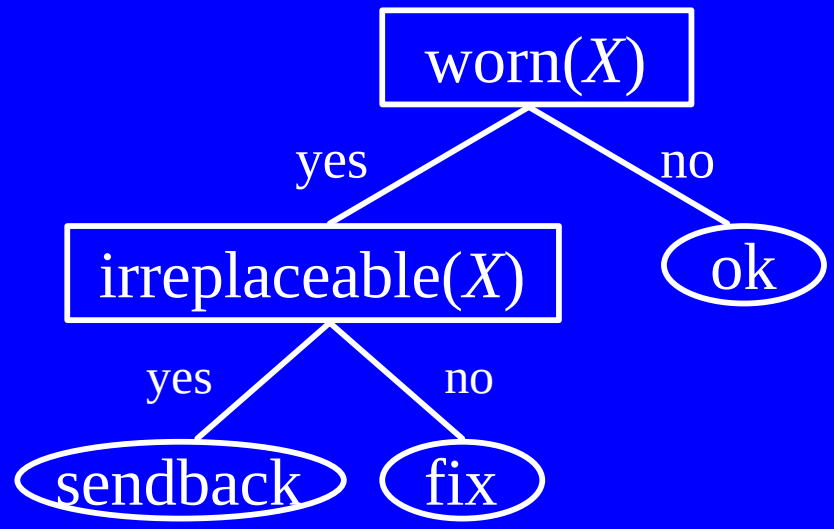
RDM – Key approaches

- Transforming RDM problems to propositional form
- Inverting deductive reasoning in FOL (inverting resolution, entailment)
- Adapting/upgrading propositional approaches to FOL

Decision trees in ILP

- First order logical decision trees
  - Binary trees
  - Each node contains a logical conjunction
  - Nodes can share variables
  - Test in node = conjunction in node + conjunctions on path from root to node
- Complex semantics
- High expressiveness

Example FOLDT



($\forall x$: $\neg$ worn(x)) => ok

($\exists x$: worn(x) $\wedge$ irreplaceable(x)) => sendback

($\exists x \forall y$: worn(x) $\wedge$ $\neg$(worn(y) $\wedge$ irreplaceable(y))) => fix

Expressiveness

FOL formula equivalent with tree:

$(\forall x: \neg\ \text{worn}(x))$ => ok
$(\exists x: \text{worn}(x) \wedge \text{irreplaceable}(x))$ => sendback
$(\exists x \forall y: \text{worn}(x) \wedge \neg(\text{worn}(y) \wedge \text{irreplaceable}(y)))$ => fix

Logic program equivalent with tree:

$a \leftarrow \text{worn}(X)$
$b \leftarrow \text{worn}(X), \text{irreplaceable}(X)$
$ok \leftarrow \neg\ a$
$sendback \leftarrow b$
$fix \leftarrow a \wedge \neg\ b$

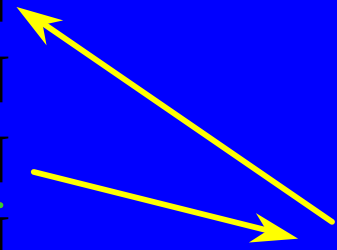
Decision trees in ILP - example

FINES

Name	Job	Speed	Fine?
Ann	teacher	150	Y
Bob	politician	160	N
Chris	engineer	120	N
Dave	writer	155	N
Earnest	politician	120	N

KNOWS

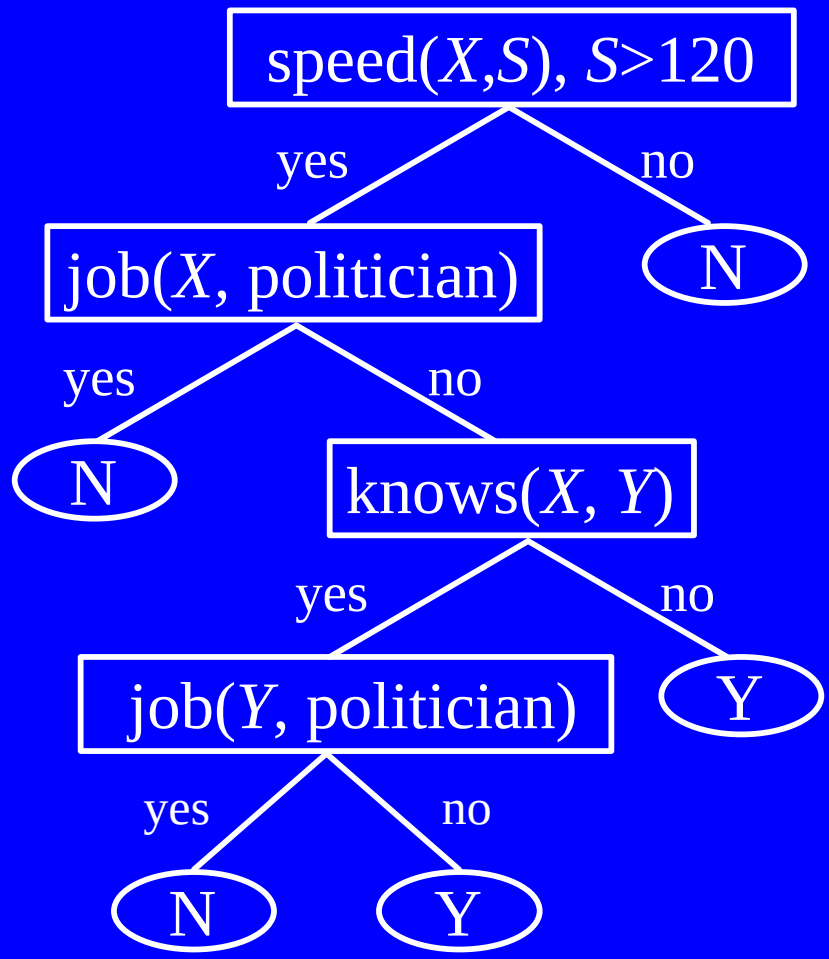
Name1	Name2
Ann	Chris
Ann	Dave
Bob	Earnest
Chris	Dave
Dave	Bob



(AVL) Speed > 120 and Job ≠ politician => Fine? = Y

(ILP) speed(*x*,*s*), *s* > 120, not job(*x*, politician), not (knows(*x*,*y*), job(*y*,politician)) => fine(*x*,Y)

speed(*x*,*s*), *s* > 120, not job(*x*, politician), not (knows(*x*,*y*), job(*y*,politician)) => fine(*x*,Y)



Predictive Clustering Trees

- Can perform classification, regression and clustering
- Conceptual clustering, an explicit description of the cluster hierarchy produced
- Also simultaneous prediction of several target variables
- Implemented in the ILP system TILDE (Blockeel 1998)

Relational IBL and Clustering

- Relational distance measure: when calculating similarity between two objects, takes into account similarity of related objects, e.g., between the children of the compared persons
- RIBL (Emde and Wettschereck 1996)
- Relational Distance-Based Clustering (RDBC) (Kirsten and Wrobel 1998)

First-order Association Rules

- IF win(W),start(W,A),in_window(W,120,B), alarm_class(B,bsc_message) THEN in_window(W,120,C),alarm_class(C,trans), same_urgency(A,C),same_urgency(C,B)
- Example from handling alarms in telco nets:
  'If a window starts with an alarm A and contains alarm B of class bsc_message then it will also contain alarm C of class trans of same urgency as the alarms A and B.'
- WARMR (Dehaspe 1998)
- Generalizes association rules/sequential patterns

Multi-Relational Discovery of Subgroups

- patient(PID,Name,Age,Sex,Outcome)
- patient_diagnosis(PID,DID,Date,HID)
- hospital(HID,Name,Location,Size,Owner,Class)
- Interesting subgroup: ‘Patients older than 65 who were diagnosed at a small hospital have an unusually high mortality rate.’
  patient(PID,_,A,_,_), A > 65,
  patient_diagnosis(PID,_,_,H),
  hospital(H,_,_,small,_,_)

Successful Applications

- In bioinformatics
  - Drug design: predicting activity of compounds
  - Predicting mutagenicity, carcinogenicity
  - Predicting protein structure and function
- Environmental applications
  - Predicting biodegradility; predicting physical/chemical parameters of river water quality from bioindicator data
- Mechanical and traffic engineering
- Growing interest in text and web mining apps

RDM Summer School Program

- An introduction to RDM: Saso Dzeroski
- An introduction to ILP and propositionalization: Peter Flach & Nada Lavrac
- Upgrading propositional learners to a relational setting: Luc De Raedt
- Logical decion trees: Hendrik Blockeel
- Relational subgroup discovery: Stefan Wrobel
- Relational distance-based methods: Stefan Wrobel
- Kernel-Based Learning from Structured Data: Thomas Gaertner
- Learning Statistical Models from Relational Data: Lise Getoor
- Bayesian logic programs: Luc De Raedt and Kristian Kersting
- Applications of RDM to bioinformatics: Ross King
- Overview of RDM applications: Saso Dzeroski
- Inductive databases: Luc De Raedt
- Future research / open issues in ILP/RDM: Panel discussion