Data Mining and Knowledge Discovery

Part of
Jožef Stefan IPS Programme – ICT3

2020 / 2021

Nada Lavrač

Jožef Stefan Institute

Ljubljana, Slovenia

Data Mining and Knowledge Discovery: Logistics and lecturers

Contacts: http://kt.ijs.si/petra_kralj/dmkd3.html

Nada Lavrač: nada.lavrac@ijs.si

- Introduction: ML and DM, decision tree learning, rule learning
- Relational learning: relational learning, semantic data mining
- Advanced topics: text mining, clustering, outlier detection

Petra Kralj Novak: petra.kralj.novak@ijs.si

- classification, evaluation, regression + practice with Orange in Scikit
- association rules, clustering + practice with Orange
- neural networks hands-on with Keras

Martin Žnidaršič: martin.znidarsic@ijs.si

- Advanced topics: SVM, neural networks, ensemble learning, active learning

ICT3 Course Schedule – 2020/21

ICT3 – for materials, see http://kt.ijs.si/petra_kralj/dmkd3.html
for lectures, use IPS ZOOM link

10.11.2020	15:00 - 17:00	prof. dr. Nada Lavrač
17.11.2020	15:00 - 17:00	doc. dr. Petra Kralj Novak
24.11.2020	15:00 - 17:00	prof. dr. Nada Lavrač
1.12.2020	15:00 - 17:00	doc. dr. Petra Kralj Novak
8.12.2020	15:00 - 17:00	doc. dr. Martin Žnidaršič
15.12.2020	15:00 - 17:00	doc. dr. Petra Kralj Novak, doc. dr. Martin Žnidaršič
22.12.2021	15:00 - 17:00	doc. dr. Petra Kralj Novak - **Oral exam** - **Using Petra's personal ZOOM link**
19.1.2021	15:00 - 18:00	prof. dr. Nada Lavrač - **Seminar presentations** - **Using IPS ZOOM link**

Data Mining and Knowledge Discovery: Credits and Coursework

Course requirements (10 ECTS credits):

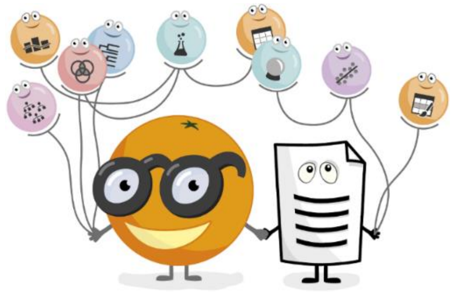
- Attending lectures and selected hands-on exercises
- Oral exam (40%)
- Seminar (60%):
  - Data analysis of your own data
  - …. own initiatives highly recommended …

Data Mining and Knowledge Discovery: Credits and Coursework

Exam: Oral exam - Theory

Seminar: topic selection + results presentation

- One hour available for seminar topic discussion – one page written proposal defining the task and the selected dataset
- Deliver written report + electronic copy (4 pages in Information Society paper format, instructions on the web)
  - Report on data analysis of own data needs to follow the CRISP-DM methodology
  - Presentation of your seminar results (15 minutes each: 10 minutes presentation + 5 minutes discussion)

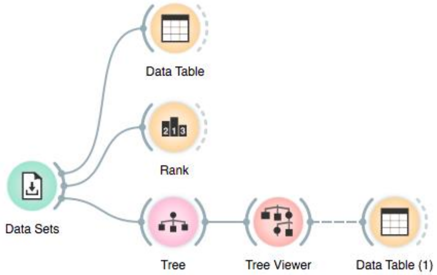


orange

- Open source machine learning and data visualization toolbox
  - https://orange.biolab.si/
  - http://file.biolab.si/datasets/
  - https://www.youtube.com/channel/UClKKWBe2SCAEyv7ZNGhIe4g
- Interactive data analysis workflows
- Visual programming
- Based on numpy, scipy and **scikit-learn**
- GUI: Qt framework

Hands-on exercises

Orange



- Open source machine learning and data visualization
- Interactive data analysis workflows with a large toolbox
- Visual programming
- *Demsar J, Curk T, Erjavec A, Gorup C, Hocevar T, Milutinovic M, Mozina M, Polajnar M, Toplak M, Staric A, Stajdohar M, Umek L, Zagar L, Zbontar J, Zitnik M, Zupan B (2013) Orange: Data Mining Toolbox in Python, JMLR 14(Aug): 2349−2353.*

scikit learn

- **scikit-learn i**s Gold standard of Python machine learning
- Simple and efficient tools for data mining and data analysis
- Well documented
- *Pedregosa et al. (2011) [Scikit-learn: Machine Learning in Python](), JMLR 12, pp. 2825-2830.*

K Keras

- Neural-network library written in Python.
- *Chollet, F. et al. (2015) "Keras"*

```
# -----------------------------------------------------------------------------------------
print("Train and test classification models")
classifiers = [
    # ("Naive Bayes", naive_bayes.MultinomialNB()),
    ("Logistic regression", linear_model.LogisticRegression(C=1e5, solver='lbfgs', multi_class='multinomial', max_iter=600)),
    ("MultinomialNB", MultinomialNB()),
    ("SVC", svm.LinearSVC()),
    ("SVC-RBF", svm.SVC(gamma='scale', decision_function_shape='ovo'))]


for name, classifier in classifiers:
    classifier.fit(train_data, y_train)
    predictions = classifier.predict(test_data)
    classifier.confusion_matrix = metrics.confusion_matrix(predictions, y_test, labels=["negative", "neutral", "positive"])
    classifier.accuracy = metrics.accuracy_score(predictions, y_test)
    print(name, classifier.accuracy, "\n Confusion matrix: \n", classifier.confusion_matrix)
    pickle_clf(classifier, path="./models/"+name+".pkl")
```

Data Mining and Knowledge Discovery: Supporting material

- Supporting material on videolectures.net:

  Seminar: AI for Industry and Society, Ljubljana 2020
  - http://videolectures.net/AIindustrySeminar2019/
  - Marko Robnik Šikonja: Artificial Intelligence: Techniques, Trends and Applications
  - Nada Lavrač: Data Science, Machine Learning and Big Data: Current trends
  - Blaž Zupan: Data Science with the OrangeToolbox

Machine Learning and Data Mining

- Machine Learning (ML) – computer algorithms/machines that learn predictive models from class-labeled data
- Data Mining (DM) – extraction of useful information from data: discovering relationships and patterns that have not previously been known, and use of ML techniques applied to solving real-life data analysis problems
- Knowledge discovery in databases (KDD) – the process of knowledge discovery

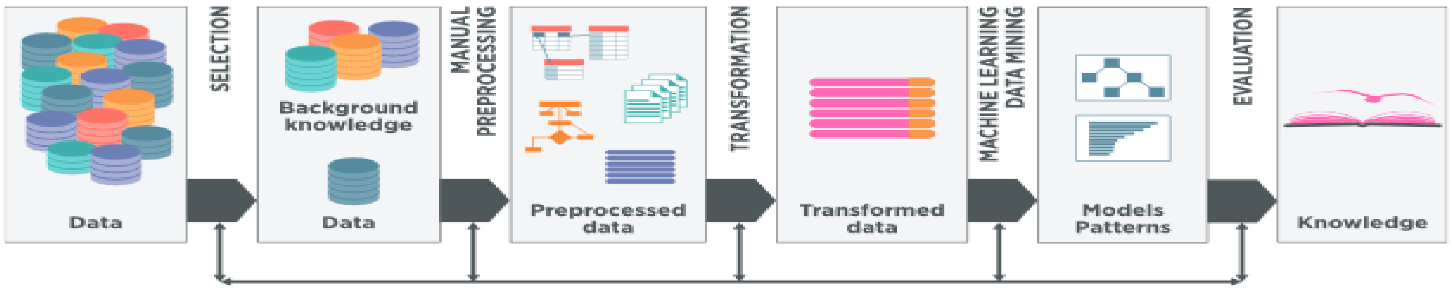
Data Mining and KDD

- Buzzword since 1996
- KDD is defined as “the process of identifying valid, novel, potentially useful and ultimately understandable models/patterns in data.” *
- Data Mining (DM) is the key step in the KDD process, performed by using data mining techniques for extracting models or interesting patterns from the data.

Usama M. Fayyad, Gregory Piatesky-Shapiro, Pedhraic Smyth: The KDD Process for Extracting Useful Knowledge form Volumes of Data. Comm ACM, Nov 96/Vol 39 No 11

KDD Process: CRISP-DM

KDD process of discovering useful knowledge from data

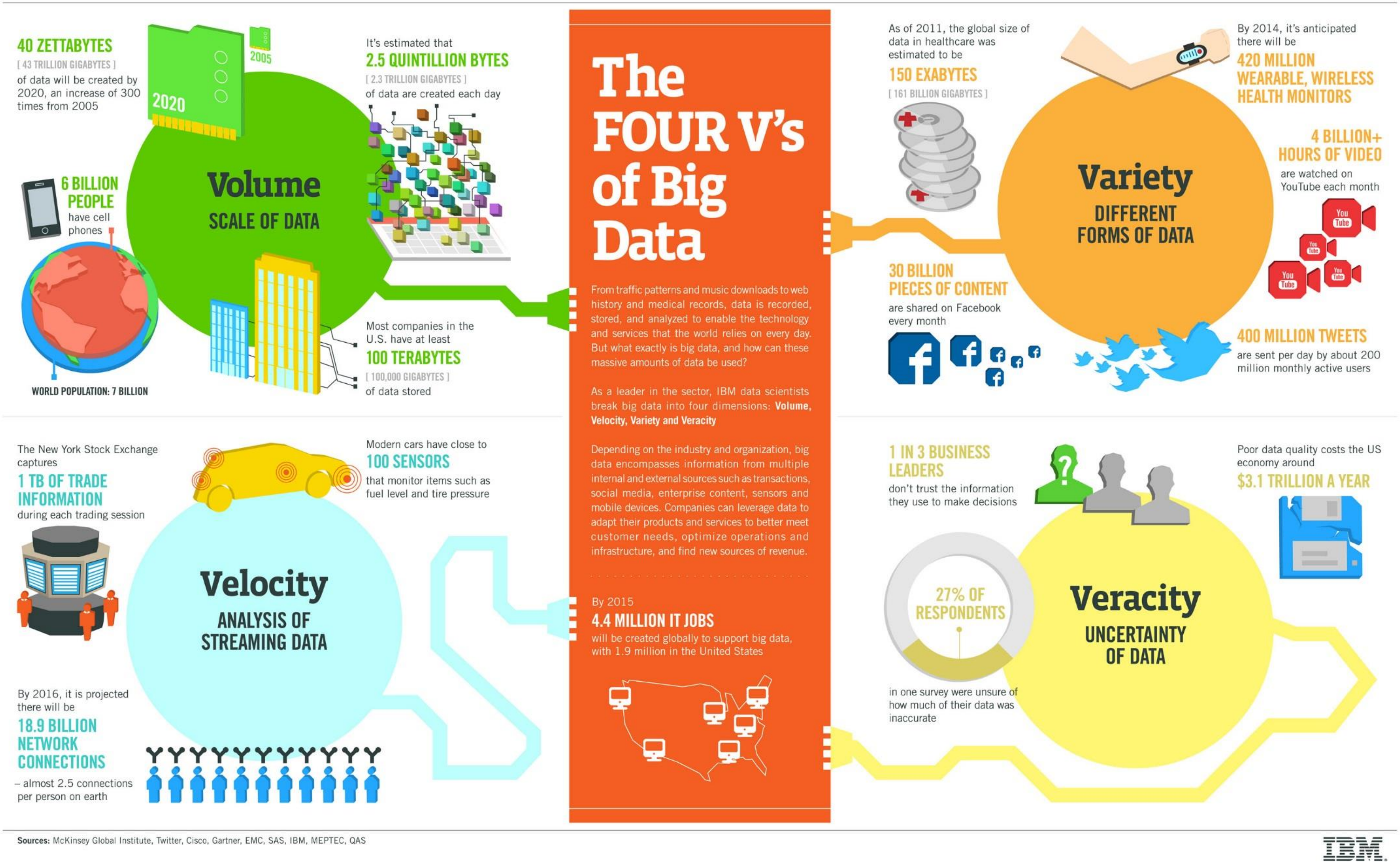


- KDD process involves several phases:
  - data preparation
  - data mining (machine learning, statistics)
  - evaluation and use of discovered patterns
- Data mining is the key step, but represents only 15%-25% of the entire KDD process

Big Data

- Big Data – Buzzword since 2008 (special issue of Nature on Big Data)
  - data and techniques for dealing with very large volumes of data, possibly dynamic data streams
  - requiring large data storage resources, special algorithms for parallel computing architectures.

The 4 Vs of Big Data



Data Science

- Data Science – buzzword since 2012 when Harvard Business Review called it "The Sexiest Job of the 21st Century"
  - an interdisciplinary field that uses scientific methods, processes, algorithms and systems to extract knowledge and insights from data in various forms, both structured and unstructured, similar to data mining.
  - used interchangeably with earlier concepts like business analytics, business intelligence, predictive modeling, and statistics.

Machine Learning and Data Mining

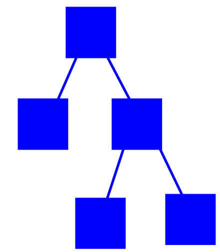
data

Person	Age	Spect. presc.	Astigm.	Tear prod.	Lenses
O1	17	myope	no	reduced	NONE
O2	23	myope	no	normal	SOFT
O3	22	myope	yes	reduced	NONE
O4	27	myope	yes	normal	HARD
O5	19	hypermetrope	no	reduced	NONE
O6-O13	...	...	...	...	...
O14	35	hypermetrope	no	normal	SOFT
O15	43	hypermetrope	yes	reduced	NONE
O16	39	hypermetrope	yes	normal	NONE
O17	54	myope	no	reduced	NONE
O18	62	myope	no	normal	NONE
O19-O23	...	...	...	...	...
O24	56	hypermetrope	yes	normal	NONE

data

knowledge discovery
from data

Machine Learning
Data Mining



model, patterns, …

Given: class labeled data

Find: a classification model, a set of interesting patterns

Machine Learning and Data Mining

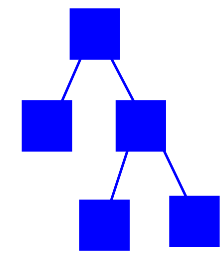
data

Person	Age	Spect. presc.	Astigm.	Tear prod.	Lenses
O1	17	myope	no	reduced	NONE
O2	23	myope	no	normal	SOFT
O3	22	myope	yes	reduced	NONE
O4	27	myope	yes	normal	HARD
O5	19	hypermetrope	no	reduced	NONE
O6-O13	...	...	...	...	...
O14	35	hypermetrope	no	normal	SOFT
O15	43	hypermetrope	yes	reduced	NONE
O16	39	hypermetrope	yes	normal	NONE
O17	54	myope	no	reduced	NONE
O18	62	myope	no	normal	NONE
O19-O23	...	...	...	...	...
O24	56	hypermetrope	yes	normal	NONE

data

knowledge discovery
from data

Machine Learning
Data Mining

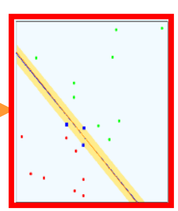


model, patterns, …

Given: class labeled data

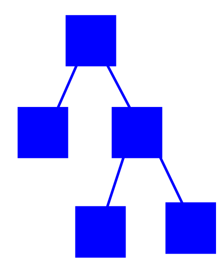
Find: a classification model, a set of interesting patterns

new unclassified instance



classified instance

black box classifier
no explanation



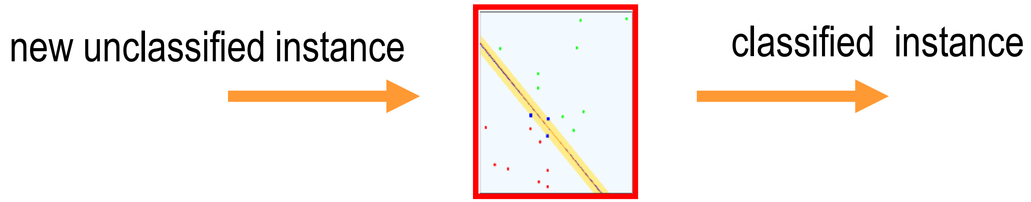
symbolic model
symbolic patterns

explanation

Why learn and use black-box models

Given: the learned classification model
(e.g, a linear classifier, a deep neural network, …)

Find: - the class label for a new unlabeled instance



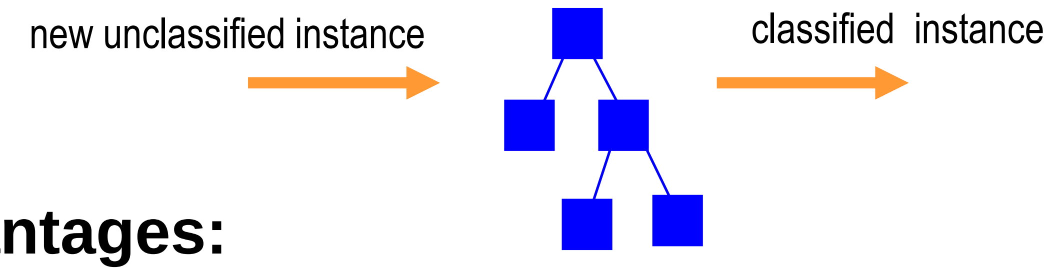
Advantages:
- best classification results in image recognition and other complex classification tasks

Drawbacks:
- poor interpretability of results
- can not be used for pattern analysis

Why learn and use symbolic models

Given: the learned classification model
(a decision tree or a set of rules)

Find: - the class label for a new unlabeled instance



Advantages:

- use the model for the explanation of classifications of new data instances
- use the discovered patterns for data exploration

Drawbacks:

- lower accuracy than deep NNs

Simplified example: Learning a classification model from contact lens data

Person	Age	Spect. presc.	Astigm.	Tear prod.	Lenses
O1	17	myope	no	reduced	NONE
O2	23	myope	no	normal	SOFT
O3	22	myope	yes	reduced	NONE
O4	27	myope	yes	normal	HARD
O5	19	hypermetrope	no	reduced	NONE
O6-O13	...	...	...	...	...
O14	35	hypermetrope	no	normal	SOFT
O15	43	hypermetrope	yes	reduced	NONE
O16	39	hypermetrope	yes	normal	NONE
O17	54	myope	no	reduced	NONE
O18	62	myope	no	normal	NONE
O19-O23	...	...	...	...	...
O24	56	hypermetrope	yes	normal	NONE

Pattern discovery in Contact lens data

Person	Age	Spect. presc.	Astigm.	Tear prod.	Lenses
O1	17	myope	no	reduced	NONE
O2	23	myope	no	normal	SOFT
O3	22	myope	yes	reduced	NONE
O4	27	myope	yes	normal	HARD
O5	19	hypermetrope	no	reduced	NONE
O6-O13	...	...	...	...	...
O14	35	hypermetrope	no	normal	SOFT
O15	43	hypermetrope	yes	reduced	NONE
O16	39	hypermetrope	yes	normal	NONE
O17	54	myope	no	reduced	NONE
O18	62	myope	no	normal	NONE
O19-O23	...	...	...	...	...
O24	56	hypermetrope	yes	normal	NONE

PATTERN

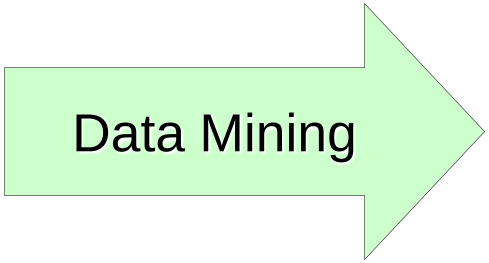
Rule:

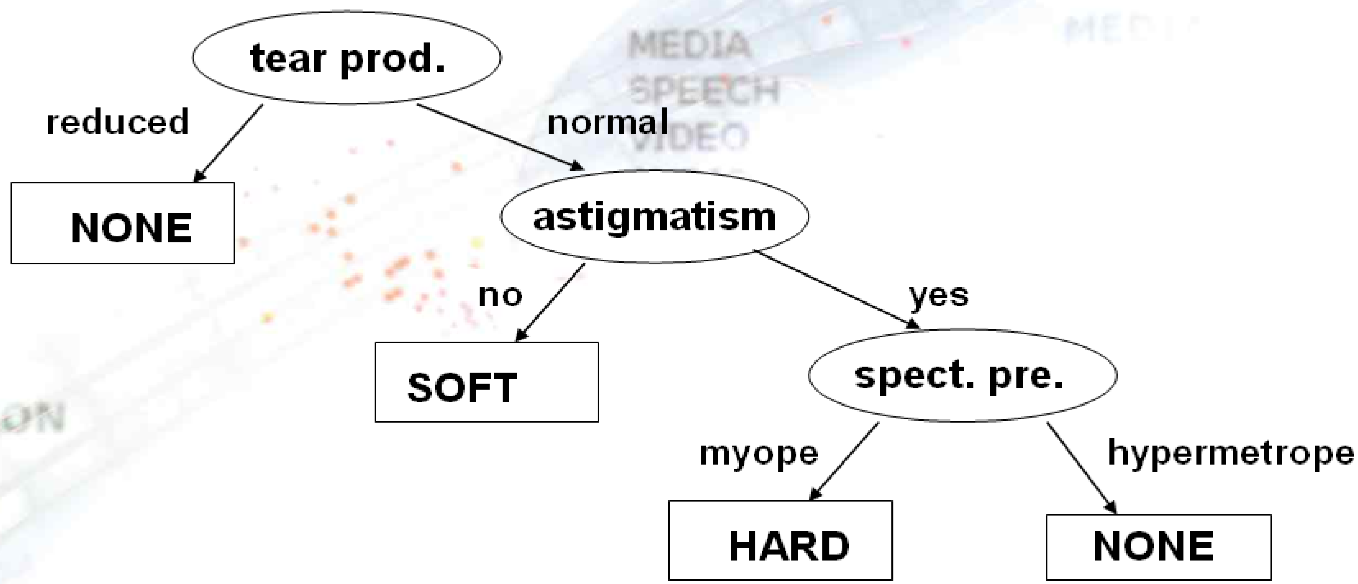
IF
Tear prod. =
reduced

THEN
Lenses =
NONE

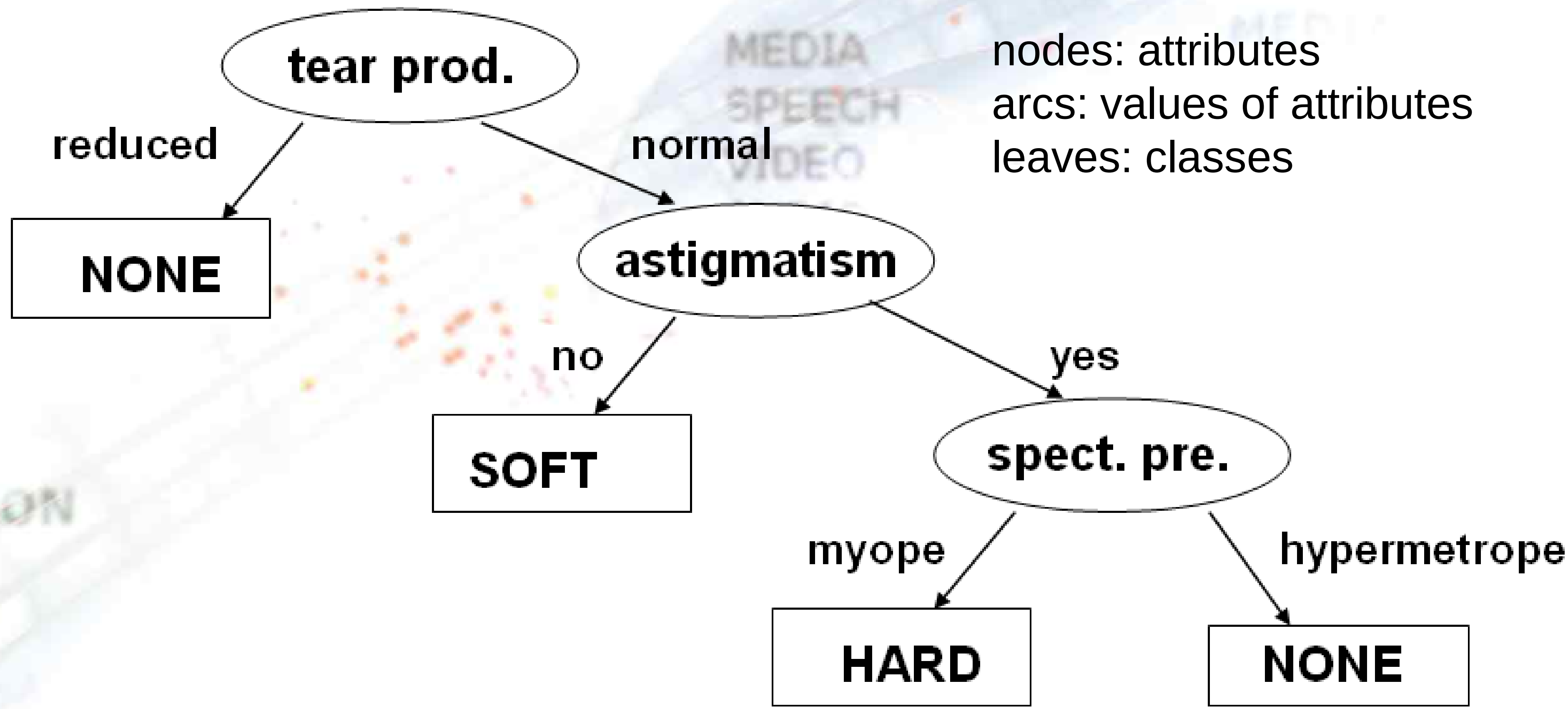
Learning a classification model from contact lens data

Person	Age	Spect. presc.	Astigm.	Tear prod.	Lenses
O1	young	myope	no	reduced	NONE
O2	young	myope	no	normal	SOFT
O3	young	myope	yes	reduced	NONE
O4	young	myope	yes	normal	HARD
O5	young	hypermetrope	no	reduced	NONE
O6-O13	...	...	...	...	...
O14	pre-presbyc	hypermetrope	no	normal	SOFT
O15	pre-presbyc	hypermetrope	yes	reduced	NONE
O16	pre-presbyc	hypermetrope	yes	normal	NONE
O17	presbyopic	myope	no	reduced	NONE
O18	presbyopic	myope	no	normal	NONE
O19-O23	...	...	...	...	...
O24	presbyopic	hypermetrope	yes	normal	NONE



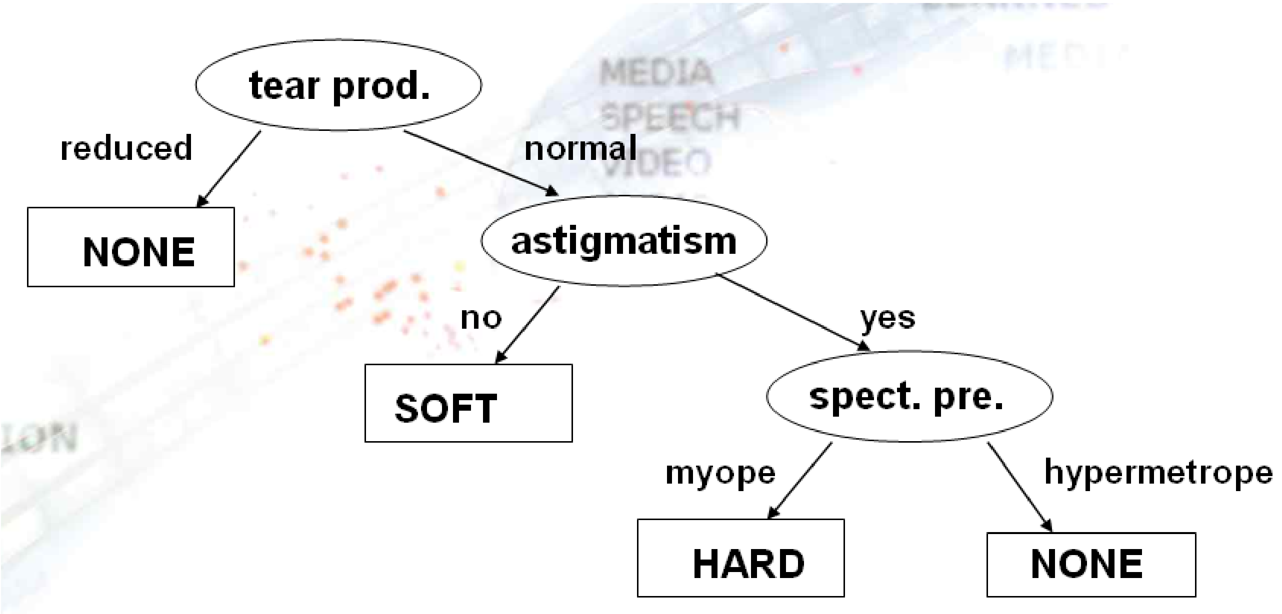


Decision tree classification model learned from contact lens data



nodes: attributes
arcs: values of attributes
leaves: classes

Learning a decision tree classification model



Search heuristics: Which attribute to test at each node in the tree ? The attribute that is most useful for classifying examples.

- First define a measure called **entropy**, to characterize the (im)purity of an arbitrary collection of examples
- **Information gain of an attribute** is measured as reduction of entropy of a training set S after splitting into subsets based on values of attribute A

Entropy

- **S** - training set, $\mathbf{C_1,...,C_N}$ - classes
- **Entropy E(S)** – measure of the impurity of training set S

$$E(S) = -\sum_{c=1}^{N} p_c . \log_2 p_c$$

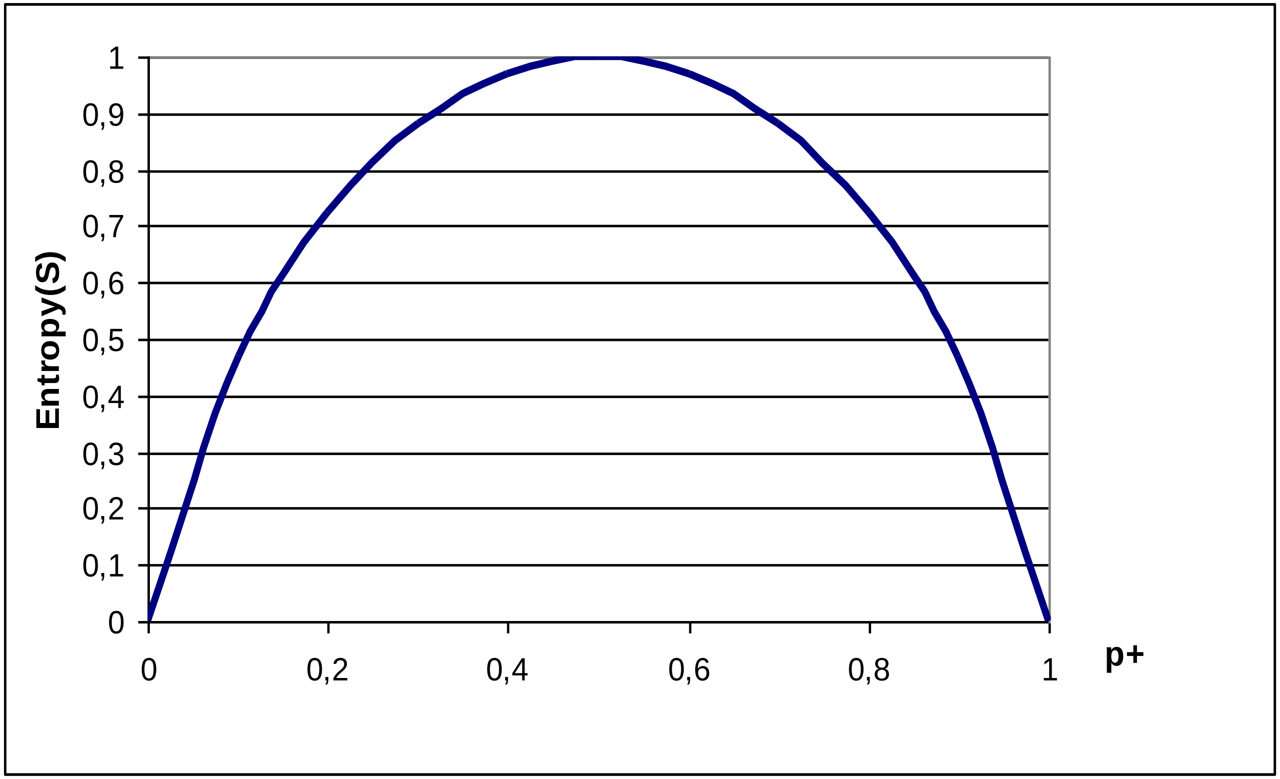
$\mathbf{p_c}$ - prior probability of class $\mathbf{C_c}$
(relative frequency of $\mathbf{C_c}$ in **S**)

- Entropy in binary classification problems

$$\mathbf{E(S) = - p_+ \log_2 p_+ - p_- \log_2 p_-}$$

Entropy

- $E(S) = - p_+ \log_2 p_+ - p_- \log_2 p_-$
- The entropy function relative to a Boolean classification, as the proportion $\mathbf{p_+}$ of positive examples varies between 0 and 1



Information gain search heuristic

- **Information gain measure** is aimed to minimize the number of tests needed for the classification of a new object

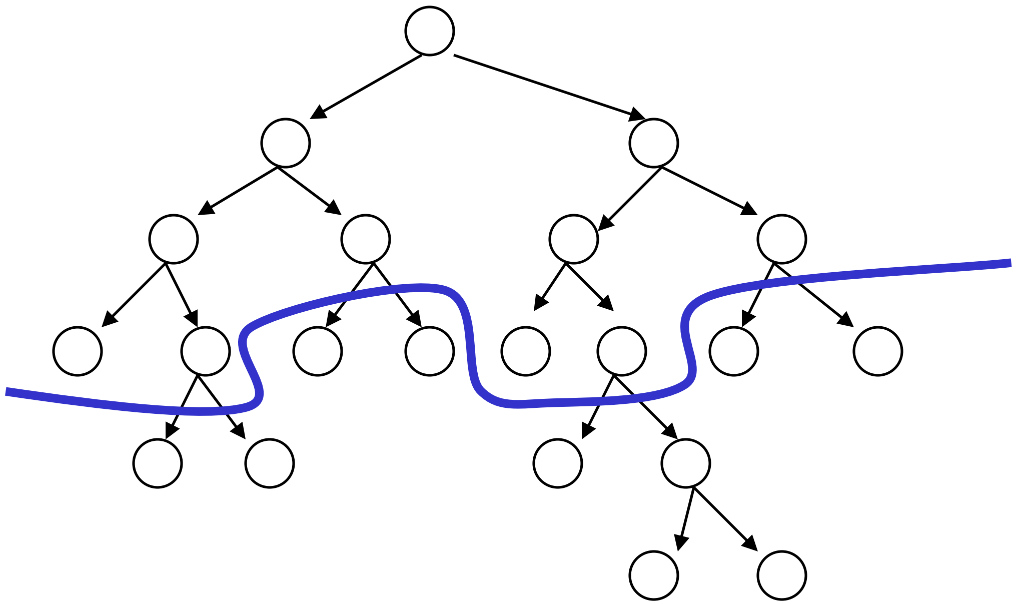
- **Gain(S,A)** – expected reduction in entropy of S due to sorting on A

$$Gain(S, A) = E(S) - \sum_{v \in Values(A)} \frac{|S_v|}{|S|} \cdot E(S_v)$$

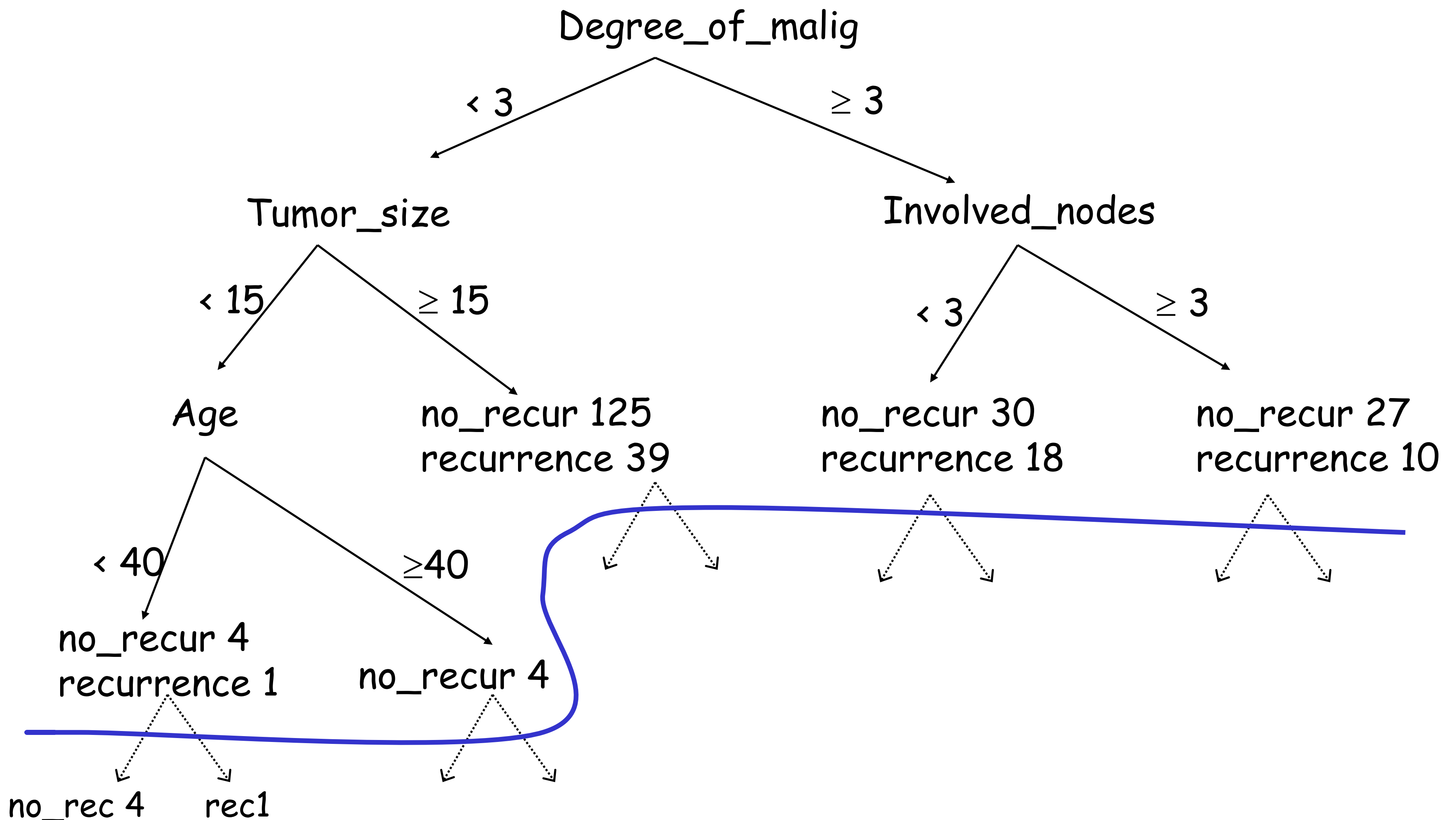
- **Most informative attribute** :
  - Select S
  - Select A to split S into $S_1, S_2, \ldots, S_v$
  - Select A, which maximizes info. Gain: **max Gain(S,A)**

Pruning of decision trees

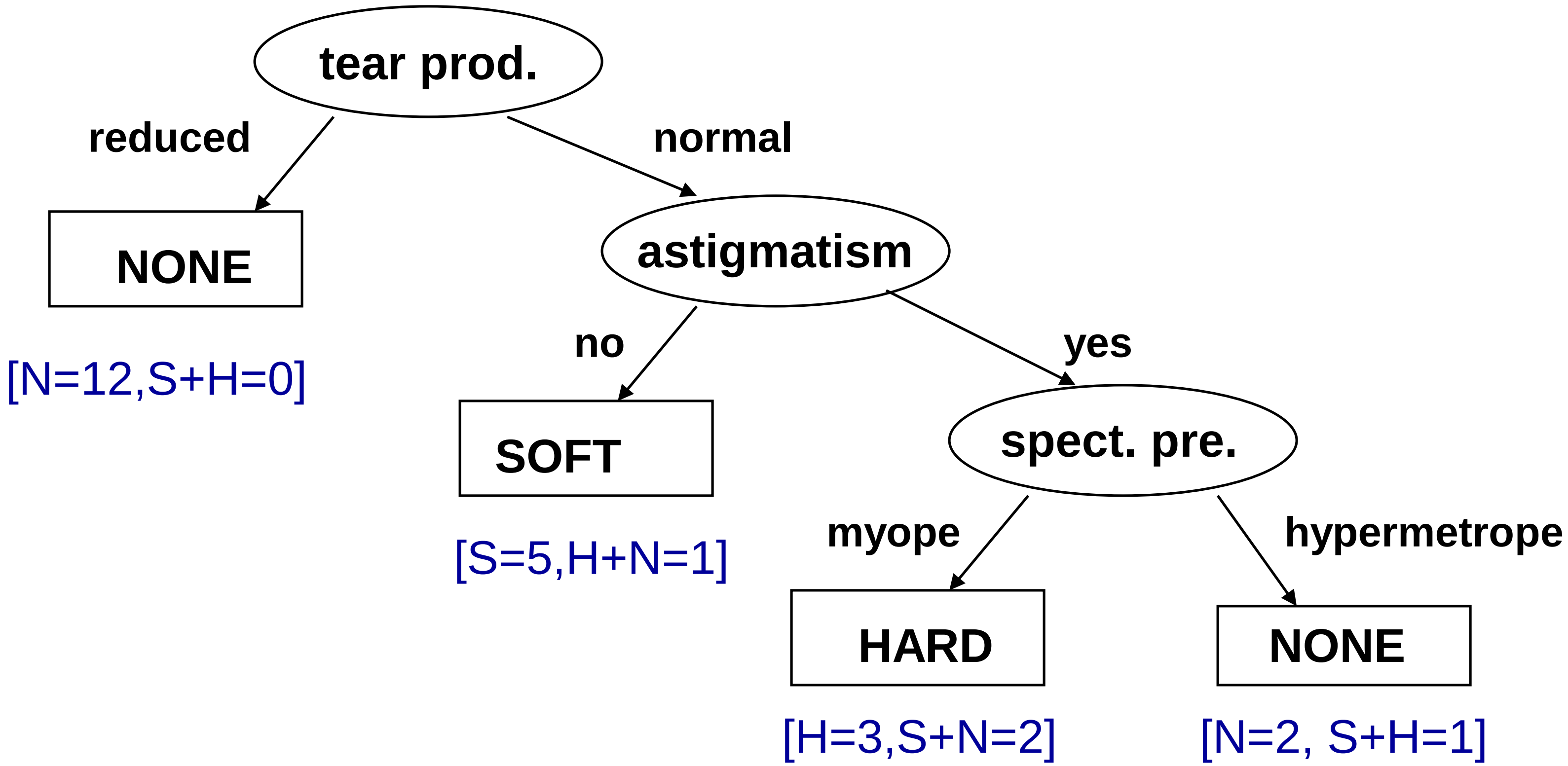
- Avoid overfitting the data by tree pruning
- Pruned trees are
  - less accurate on training data
  - more accurate when classifying unseen data



Prediction of breast cancer recurrence: Tree pruning

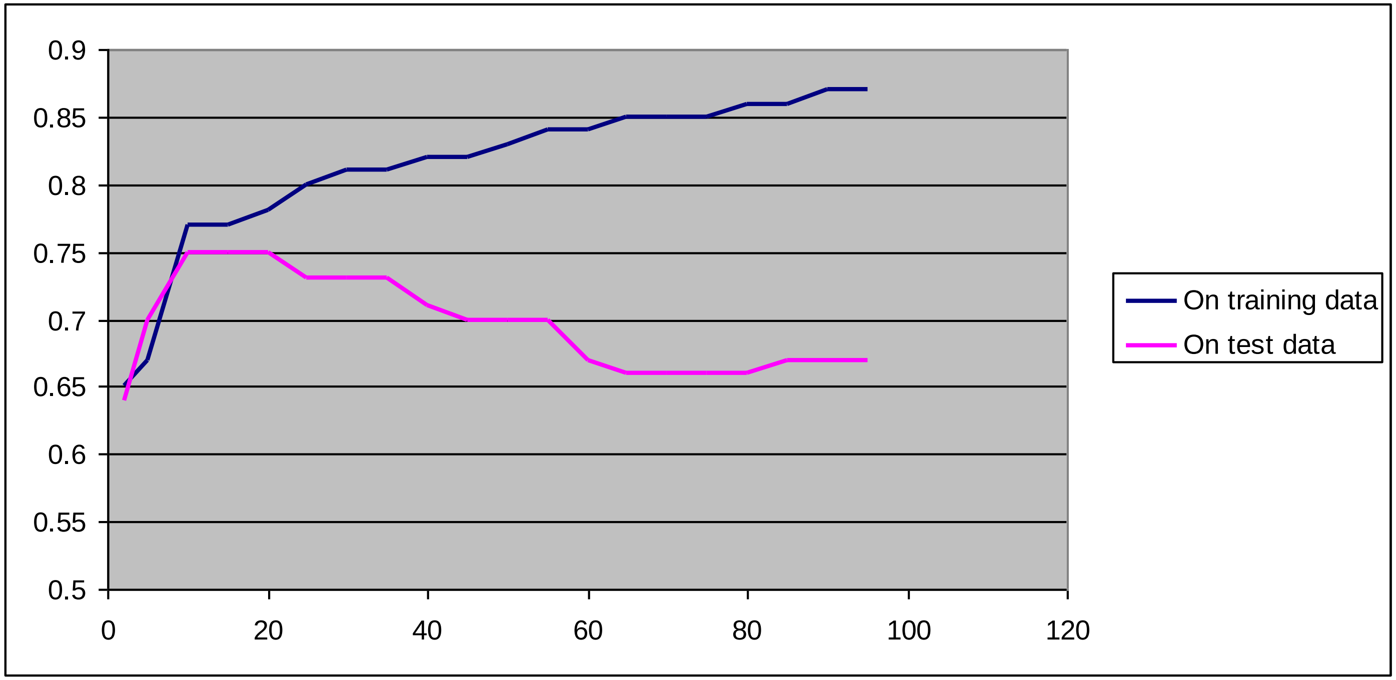


Pruned decision tree for contact lenses recommendation



Overfitting and accuracy

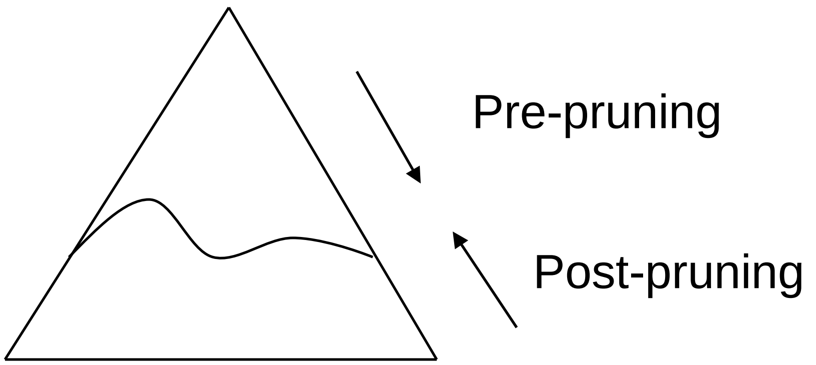
- Typical relation between tree size and accuracy



- Question: how to prune optimally?

Avoiding overfitting

- How can we avoid overfitting?
  - Pre-pruning (forward pruning): stop growing the tree e.g., when data split not statistically significant or too few examples are in a split
  - Post-pruning: grow full tree, then post-prune



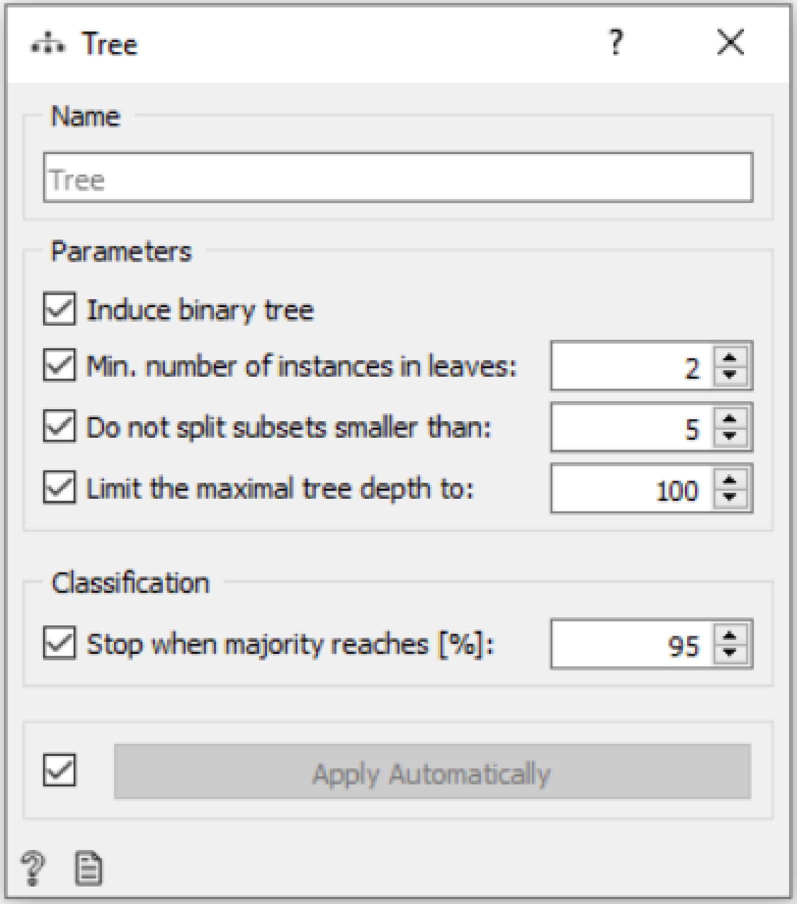
- forward pruning considered inferior (myopic)
- post pruning makes use of sub trees

Selected decision/regression tree learners

- Decision tree learners
  - ID3 (Quinlan 1979)
  - CART (Breiman et al. 1984)
  - Assistant (Cestnik et al. 1987)
  - C4.5 (Quinlan 1993), C5 (See5, Quinlan)
  - J48 (available in WEKA), Tree (in Orange)
- Regression tree learners, model tree learners
  - M5, M5P (implemented in WEKA), Tree (in Orange)

Selected decision tree learners

- Decision tree learners: Tree (in Orange)



Selected decision tree learners

- Homework
  - To prepare for the lecture of Petra Kralj Novak on 17 Nov. 2020:
  - see Blaž Zupan: Data Science with the OrangeToolbox

    http://videolectures.net/AIindustrySeminar2019_zupan_data_science/

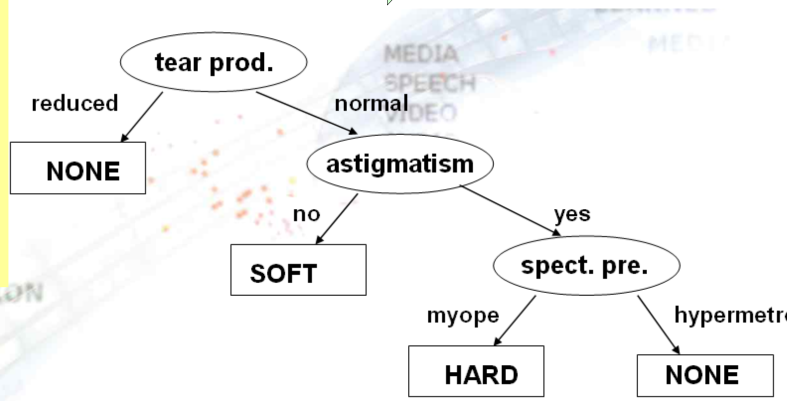
  - see also YouTube tutorials on Orange

    https://www.youtube.com/channel/UClKKWBe2SCAEyv7ZNGhIe4g

Learning a classification model from contact lens data

Person	Age	Spect. presc.	Astigm.	Tear prod.	Lenses
O1	17	myope	no	reduced	NONE
O2	23	myope	no	normal	SOFT
O3	22	myope	yes	reduced	NONE
O4	27	myope	yes	normal	HARD
O5	19	hypermetrope	no	reduced	NONE
O6-O13	...	...	...	...	...
O14	35	hypermetrope	no	normal	SOFT
O15	43	hypermetrope	yes	reduced	NONE
O16	39	hypermetrope	yes	normal	NONE
O17	54	myope	no	reduced	NONE
O18	62	myope	no	normal	NONE
O19-O23	...	...	...	...	...
O24	56	hypermetrope	yes	normal	NONE

Data Mining



lenses=NONE ← tear production=red

lenses=NONE ← tear production=normal AND astigmatism=yes AND spect. pre.=hypermetrope

lenses=SOFT ← tear production=normal AND astigmatism=no

lenses=HARD ← tear production=normal AND astigmatism=yes AND spect. pre.=myope

lenses=NONE ←

Classification rules model learned from contact lens data

lenses=NONE ← tear production=reduced

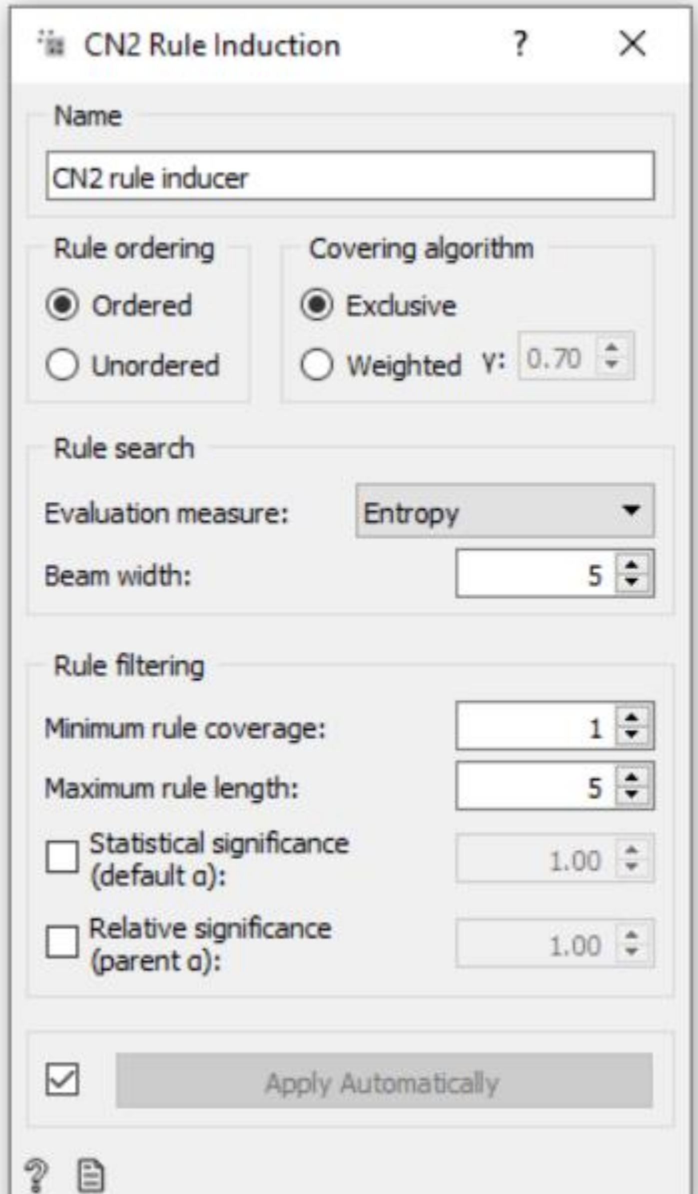
lenses=NONE ← tear production=normal AND
astigmatism=yes AND
spect. pre.=hypermetrope

lenses=SOFT ← tear production=normal AND
astigmatism=no

lenses=HARD ← tear production=normal AND
astigmatism=yes AND
spect. pre.=myope

lenses=NONE ←

CN2 rule learner in Orange



Learning from Unlabeled Data

Person	Age	Spect. presc.	Astigm.	Tear prod.	Lenses
O1	17	myope	no	reduced	NONE
O2	23	myope	no	normal	SOFT
O3	22	myope	yes	reduced	NONE
O4	27	myope	yes	normal	HARD
O5	19	hypermetrope	no	reduced	NONE
O6-O13	...	...	...	...	...
O14	35	hypermetrope	no	normal	SOFT
O15	43	hypermetrope	yes	reduced	NONE
O16	39	hypermetrope	yes	normal	NONE
O17	54	myope	no	reduced	NONE
O18	62	myope	no	normal	NONE
O19-O23	...	...	...	...	...
O24	56	hypermetrope	yes	normal	NONE

Unlabeled data - clustering: grouping of similar instances

- association rule learning

Multi-label Learning Task

Person	Age	Spect. presc.	Astigm.	Tear prod.	Lenses
O1	17	myope	no	reduced	NONE
O2	23	myope	no	normal	SOFT
O3	22	myope	yes	reduced	NONE
O4	27	myope	yes	normal	HARD
O5	19	hypermetrope	no	reduced	NONE
O6-O13	...	...	...	...	...
O14	35	hypermetrope	no	normal	SOFT
O15	43	hypermetrope	yes	reduced	NONE
O16	39	hypermetrope	yes	normal	NONE
O17	54	myope	no	reduced	NONE
O18	62	myope	no	normal	NONE
O19-O23	...	...	no	...	...
O24	56	hypermetrope	no	normal	NONE

Several class labels of training examples of a single Target class attribute

Binary Classification

Person	Age	Spect. presc.	Astigm.	Tear prod.	Lenses
O1	17	myope	no	reduced	NO
O2	23	myope	no	normal	YES
O3	22	myope	yes	reduced	NO
O4	27	myope	yes	normal	YES
O5	19	hypermetrope	no	reduced	NO
O6-O13	...	...	...	...	...
O14	35	hypermetrope	no	normal	YES
O15	43	hypermetrope	yes	reduced	NO
O16	39	hypermetrope	yes	normal	NO
O17	54	myope	no	reduced	NO
O18	62	myope	no	normal	NO
O19-O23	...	...	...	...	...
O24	56	hypermetrope	yes	normal	NO

Binary classes

- positive vs. negative examples of Target class
- Concept learning – binary classification and class description
  - for Subgroup discovery – exploring patterns characterizing groups of instances of target class

Multi-target Classification

Person	Age	Spect. presc.	Astigm.	Tear prod.	Lenses	Pilot
O1	17	myope	no	reduced	NO	NO
O2	23	myope	no	normal	YES	NO
O3	22	myope	yes	reduced	NO	NO
O4	27	myope	yes	normal	YES	NO
O5	19	hypermetrope	no	reduced	NO	NO
O6-O13	...	...	...	...	...	...
O14	35	hypermetrope	no	normal	YES	YES
O15	43	hypermetrope	yes	reduced	NO	NO
O16	39	hypermetrope	yes	normal	NO	NO
O17	54	myope	no	reduced	NO	NO
O18	62	myope	no	normal	NO	YES
O19-O23	...	...	...	...	...	...
O24	56	hypermetrope	yes	normal	NO	NO

Multi target classification

- each example belongs to several Target classes

Learning from Numeric Class Data

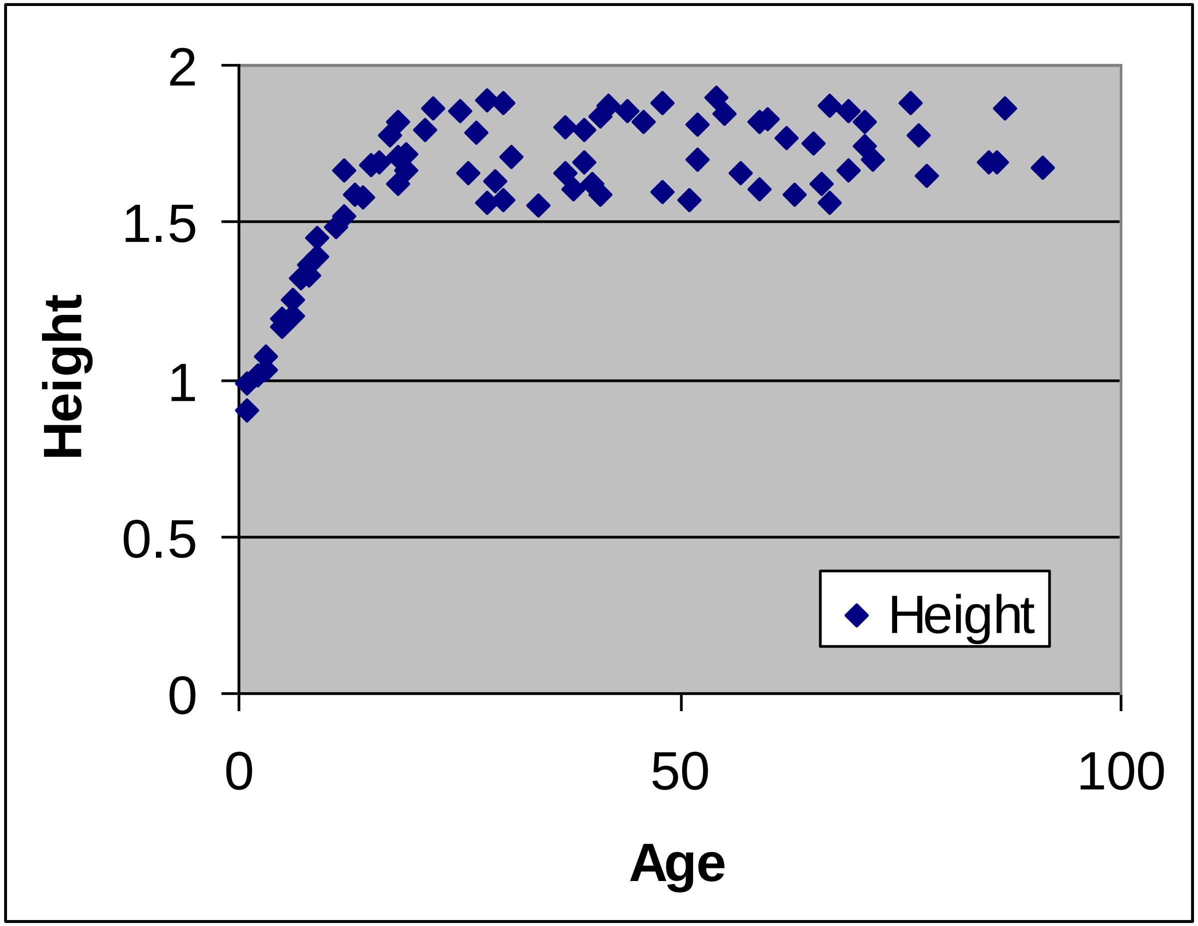
Person	Age	Spect. presc.	Astigm.	Tear prod.	LensPrice
O1	17	myope	no	reduced	0
O2	23	myope	no	normal	8
O3	22	myope	yes	reduced	0
O4	27	myope	yes	normal	5
O5	19	hypermetrope	no	reduced	0
O6-O13	...	...	...	...	...
O14	35	hypermetrope	no	normal	5
O15	43	hypermetrope	yes	reduced	0
O16	39	hypermetrope	yes	normal	0
O17	54	myope	no	reduced	0
O18	62	myope	no	normal	0
O19-O23	...	...	...	...	...
O24	56	hypermetrope	yes	normal	0

Numeric class values – regression analysis

Example regression problem

(see lectures of Petra Kralj Novak on 17 November 2020)

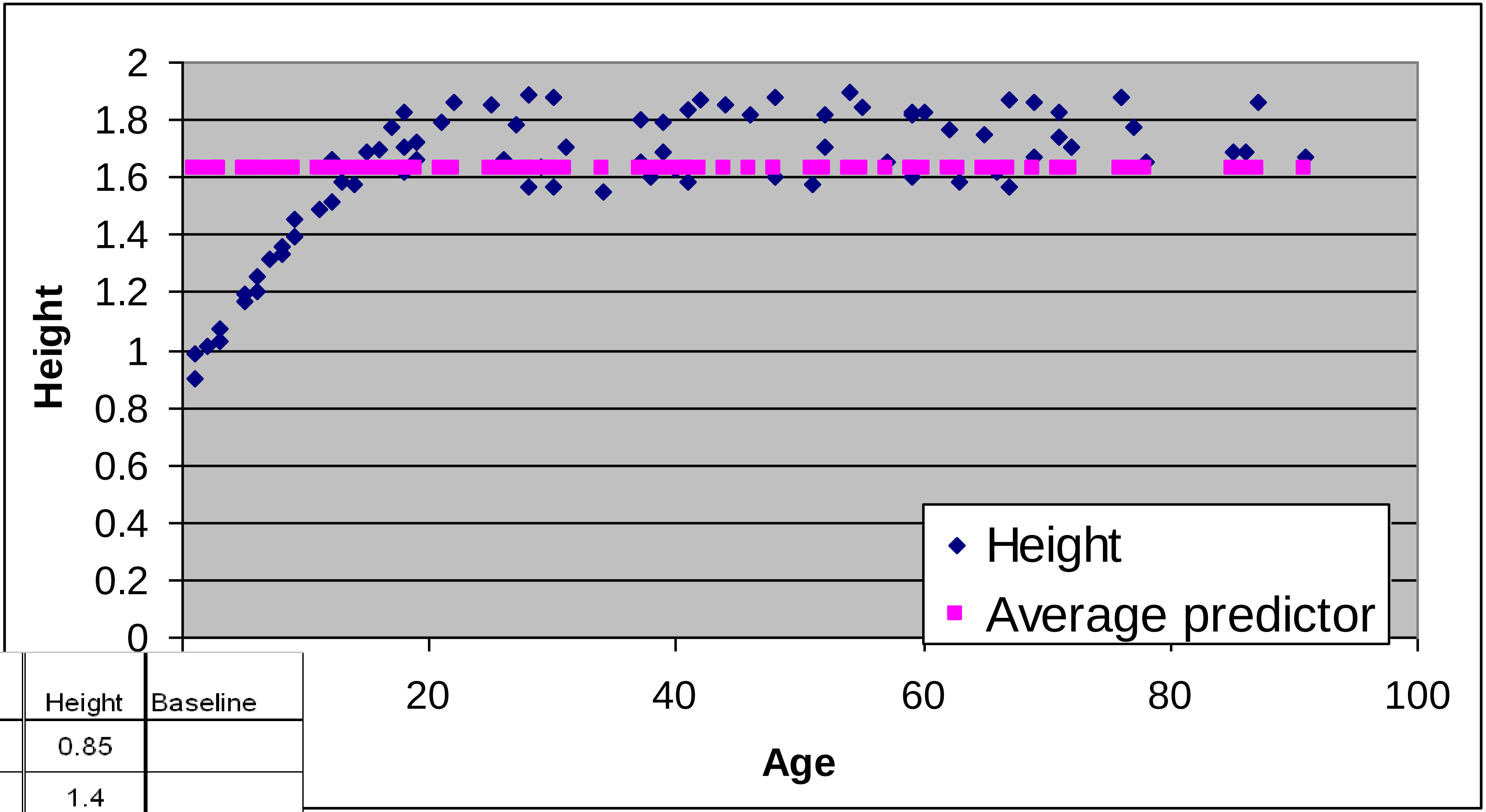
- data about 80 people: Age and Height



Age	Height
3	1.03
5	1.19
6	1.26
9	1.39
15	1.69
19	1.67
22	1.86
25	1.85
41	1.59
48	1.60
54	1.90
71	1.82
...	...

Baseline numeric model (predictor)

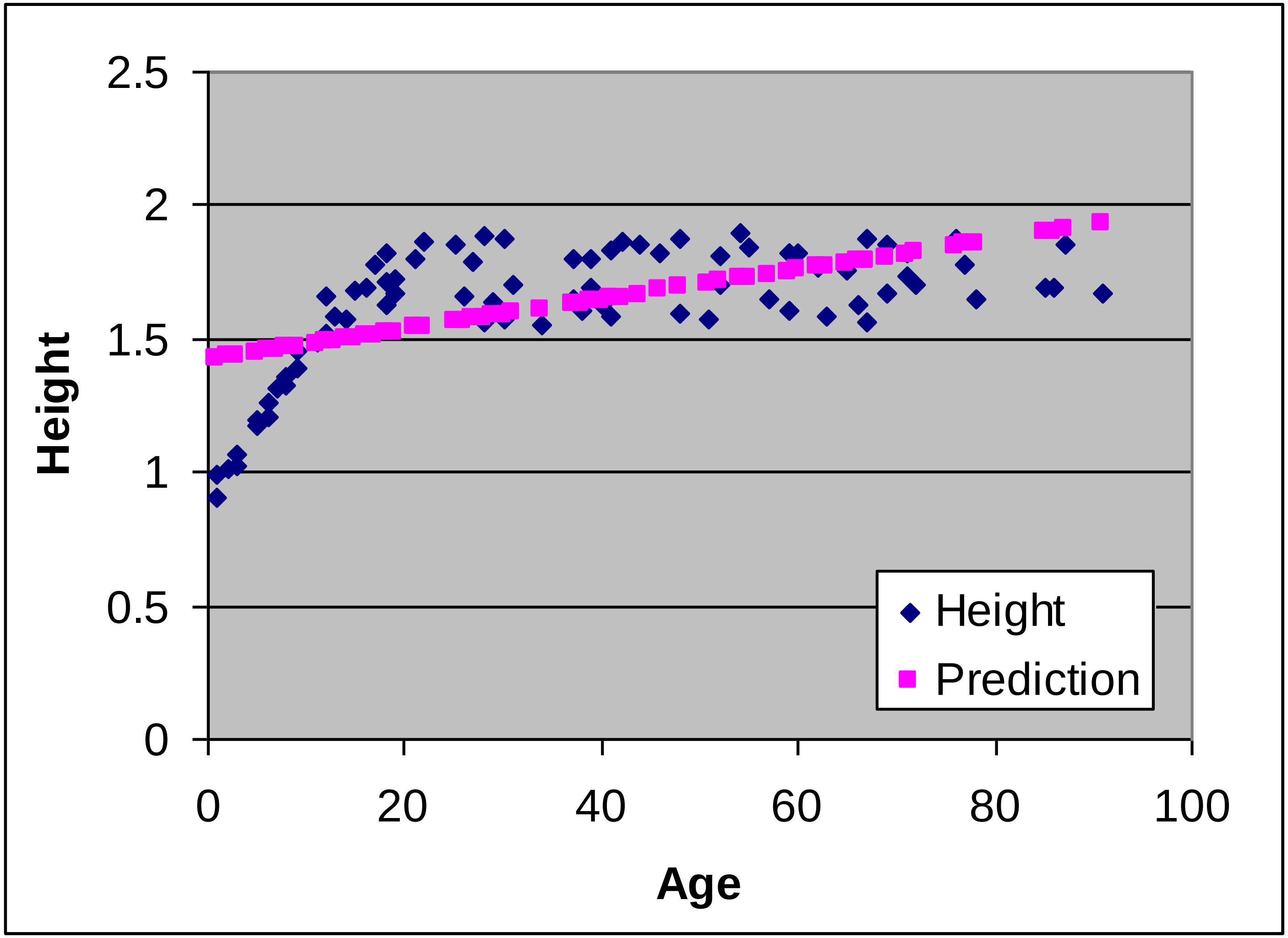
- Average of the target variable is 1.63



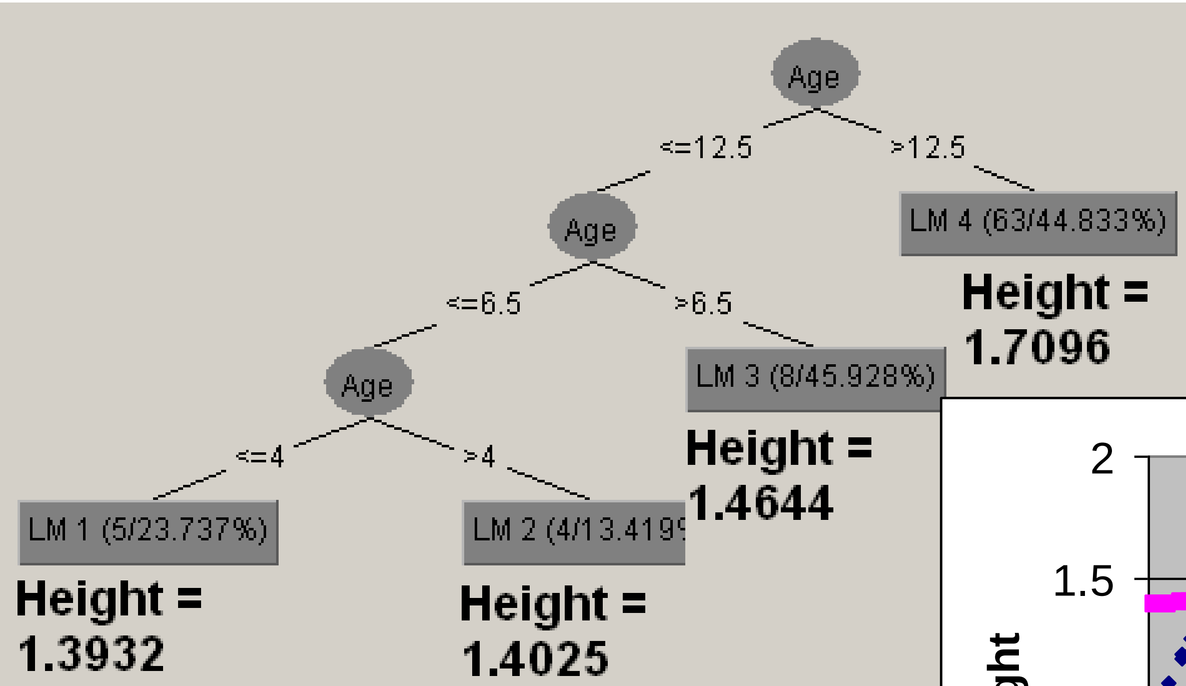
Age	Height	Baseline
2	0.85	
10	1.4	
35	1.7	
70	1.6	

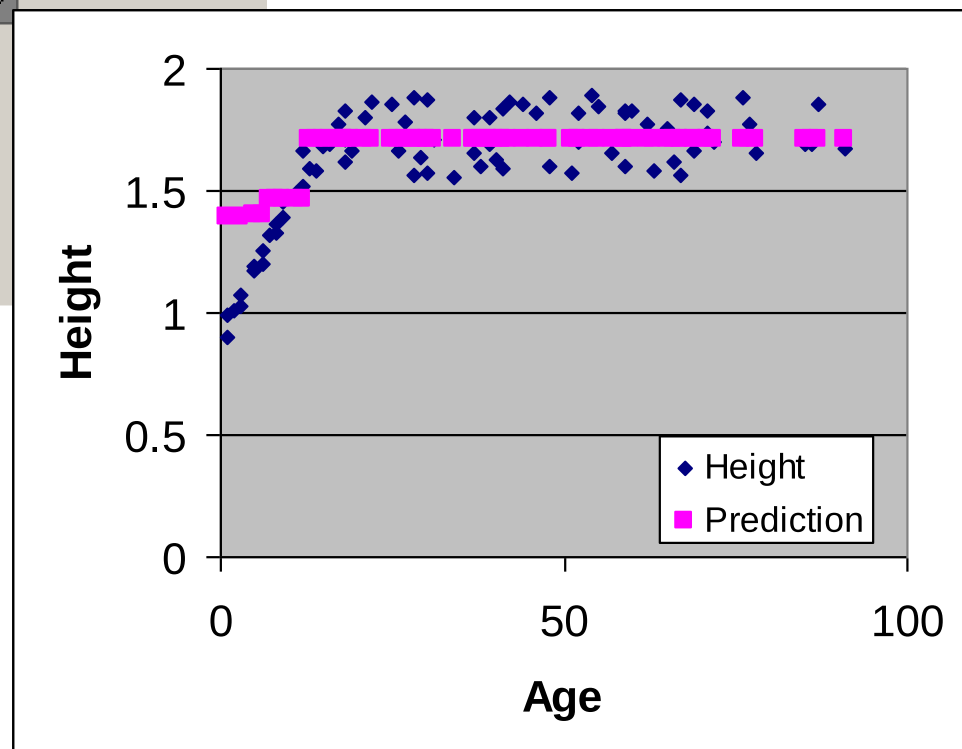
Linear Regression Model

Height = 0.0056 * Age + 1.4181

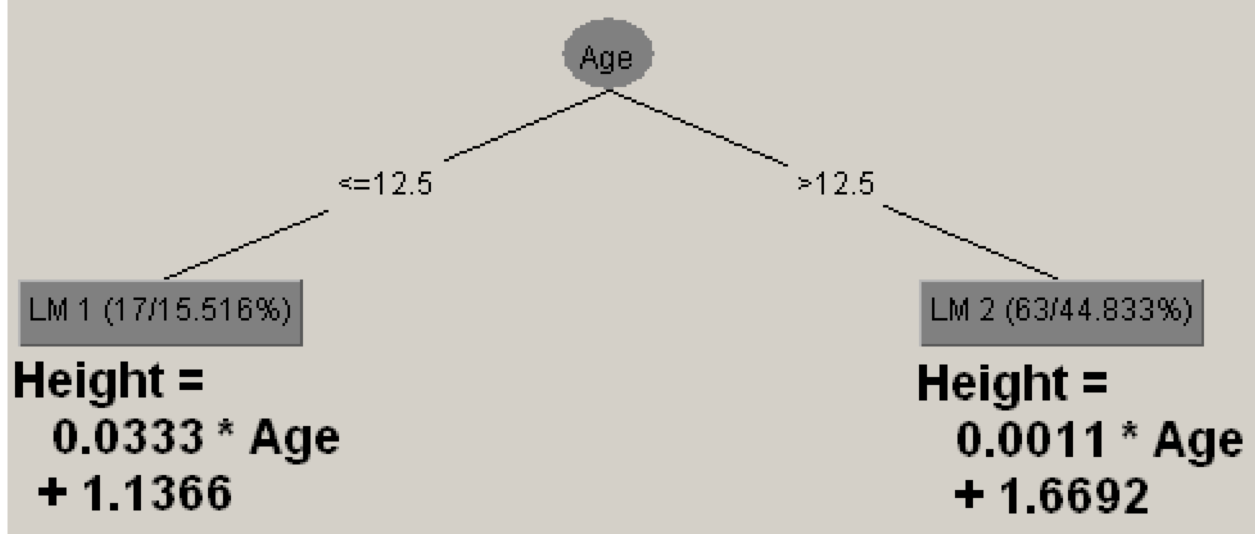


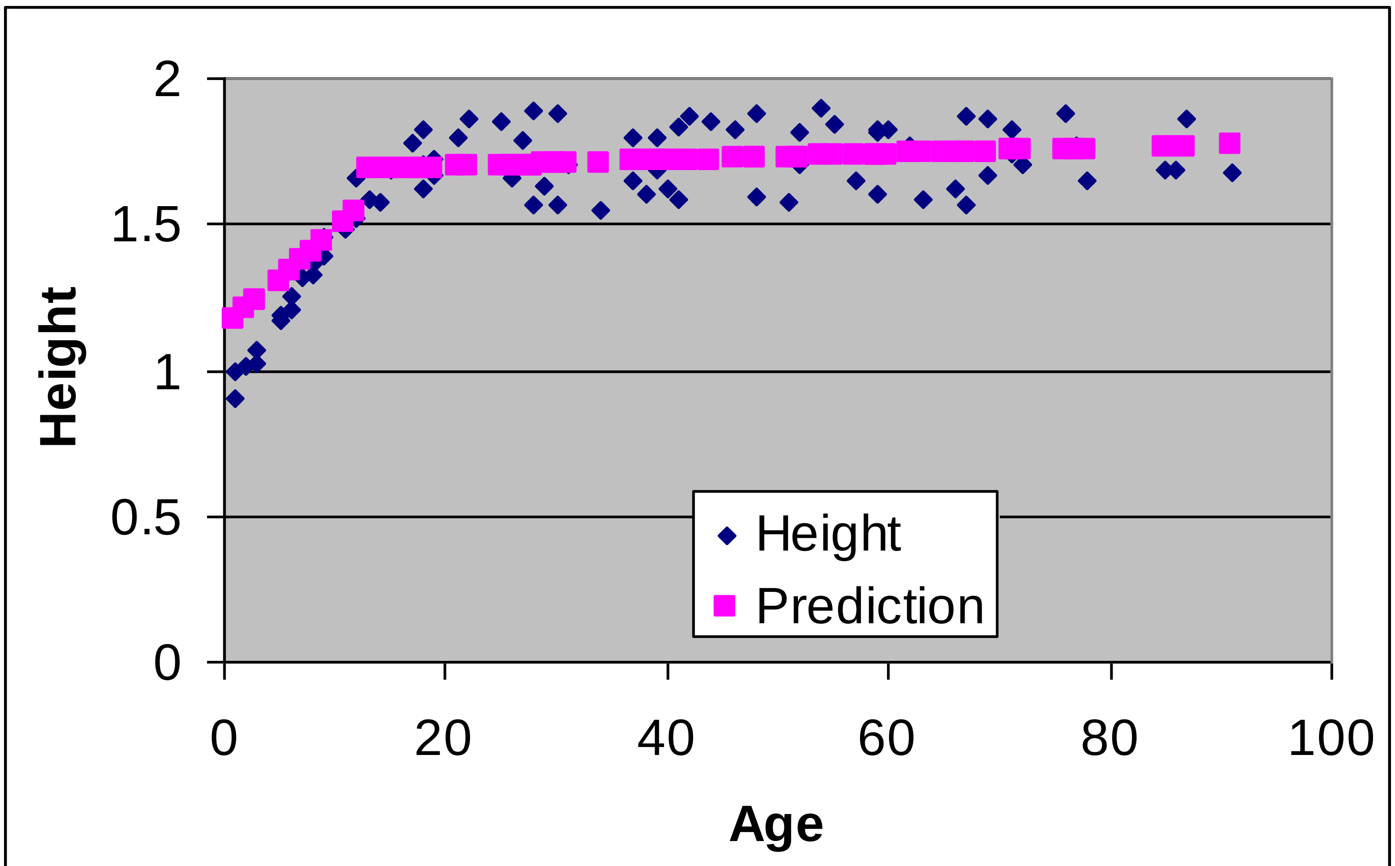
Regression tree





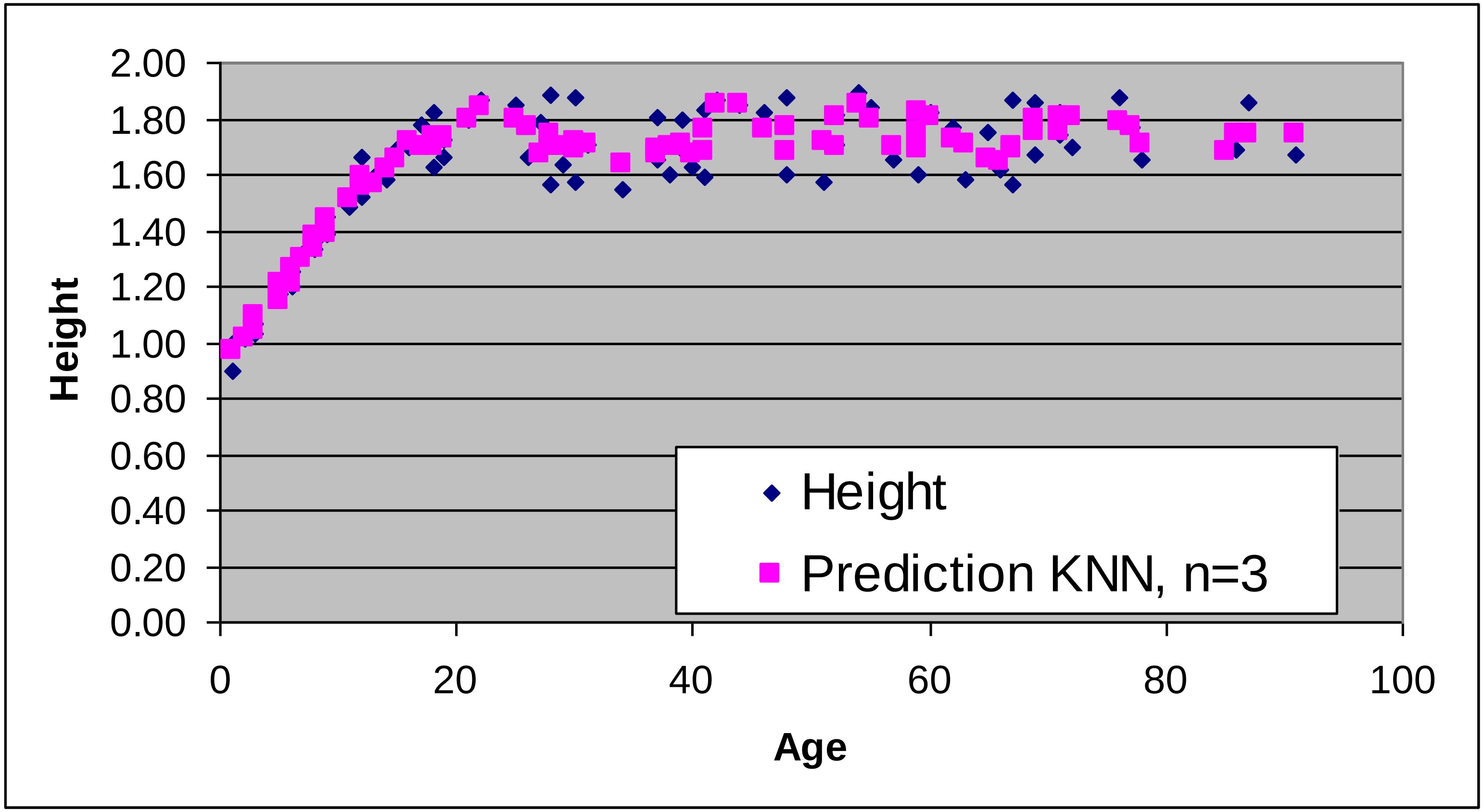
Model tree


Age
<=12.5
>12.5
LM 1 (17/15.516%)
LM 2 (63/44.833%)
Height =
0.0333 * Age
+ 1.1366
Height =
0.0011 * Age
+ 1.6692


Height
Age
2
1.5
1
0.5
0
0
20
40
60
80
100
Height
Prediction

kNN – K nearest neighbors

- Looks at K closest examples (by age) and predicts the average of their target variable
- K=3

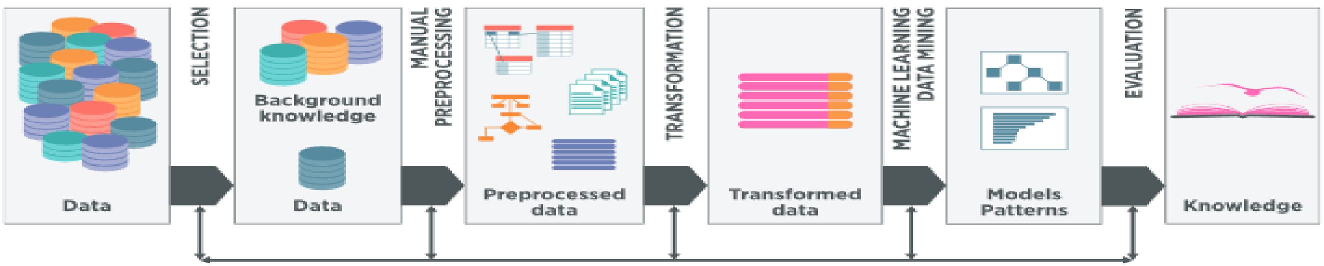


First Generation Machine Learning

- **First machine learning algorithms for**
  - Decision tree and rule learning in 1970s and early 1980s by Quinlan, Michalski et al., Breiman et al., …
- **Characterized by**
  - Learning from data stored in a single data table
  - Relatively small set of instances and attributes
- **Lots of ML research followed in 1980s**
  - Numerous conferences ICML, ECML, … and ML sessions at AI conferences IJCAI, ECAI, AAAI, …
  - Extended set of learning tasks and algorithms addressed

Second Generation Data Mining

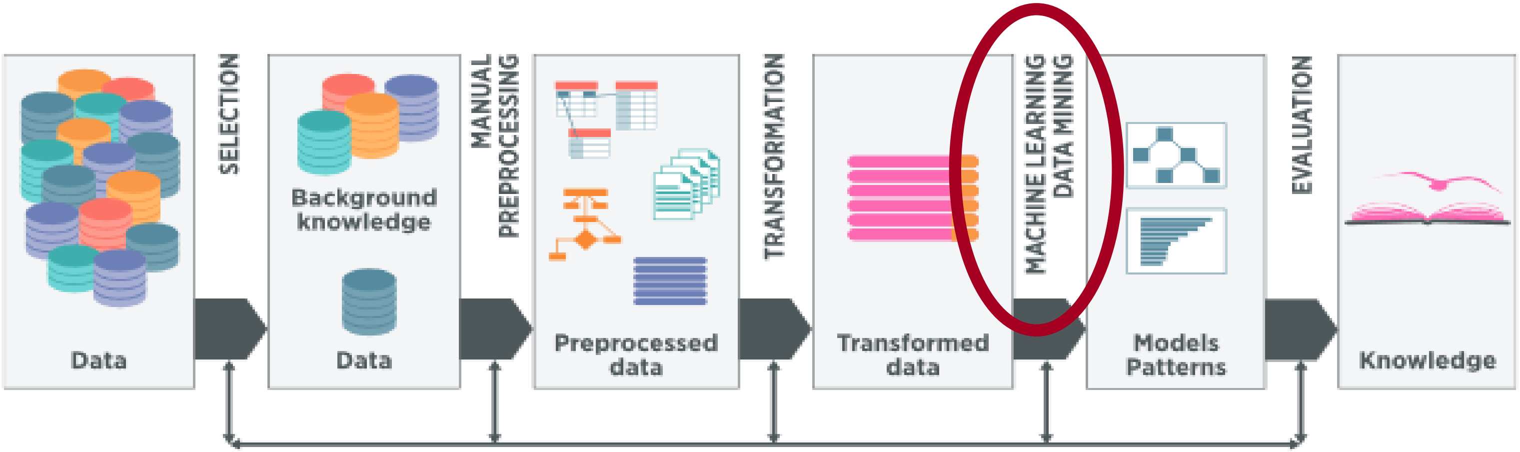
- **Developed since 1990s:**
  - Focused on data mining tasks characterized by large datasets described by large numbers of attributes
  - Industrial standard: CRISP-DM methodology (1997)



  - Since 1996 new buzzword: **Knowledge discovery in databases** (KDD)
  - KDD is defined as “the process of identifying valid, novel, potentially useful and ultimately understandable models or patterns in data.”

KDD Process

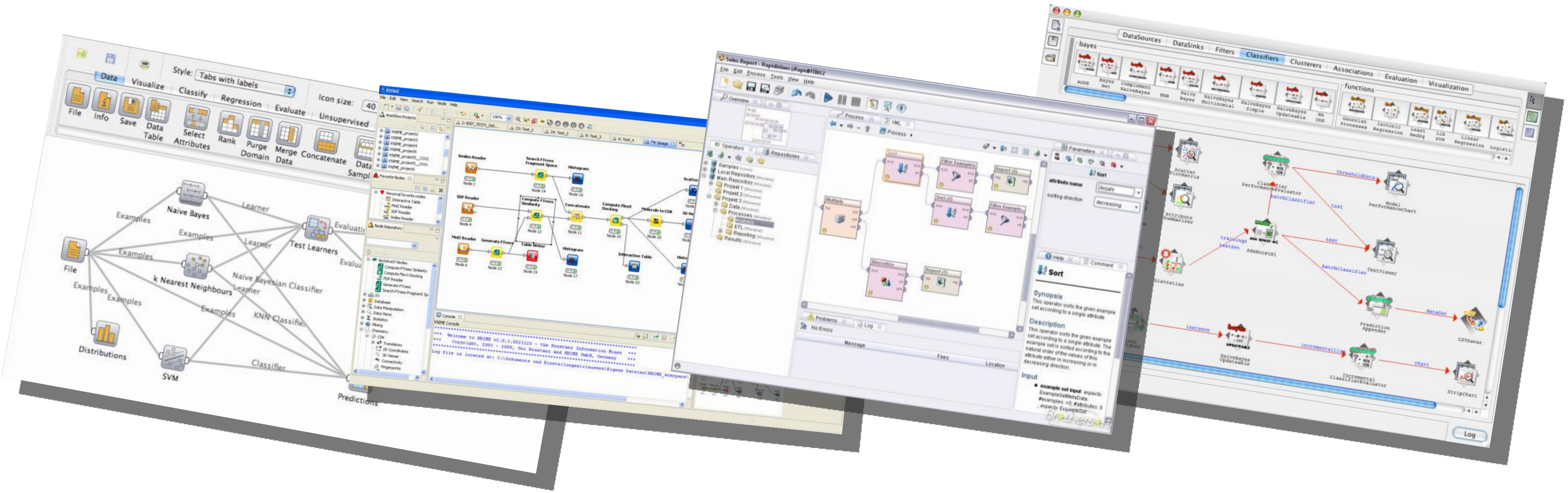
KDD process of discovering useful knowledge from data



- KDD process involves several phases:
  - data preparation
  - machine learning, data mining, statistics, …
  - evaluation and use of discovered patterns
- Machine Learning (ML) / Data Mining (DM) is the key step in the KDD process
  - performed using machine learning or pattern mining techniques for extracting classification models or interesting patterns in data
  - this key step represents only 15%-25% of entire KDD process

Second Generation Data Mining Platforms

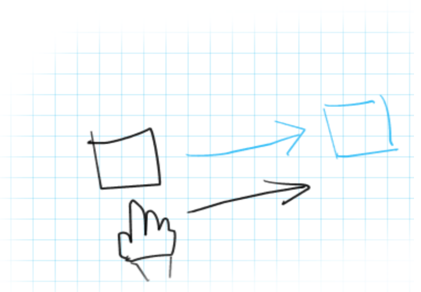
Orange, WEKA, KNIME, RapidMiner, …

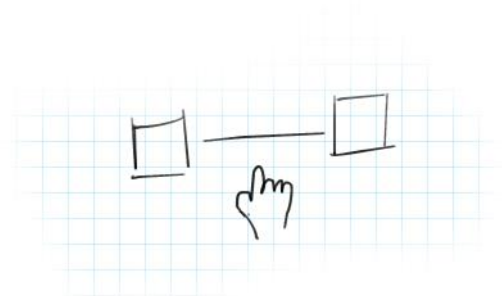


- include numerous data mining algorithms
- enable data and model visualization
- like Orange, Taverna, WEKA, KNIME, RapidMiner, also enable complex **workflow** construction

Data Mining Workflows for Open Data Science

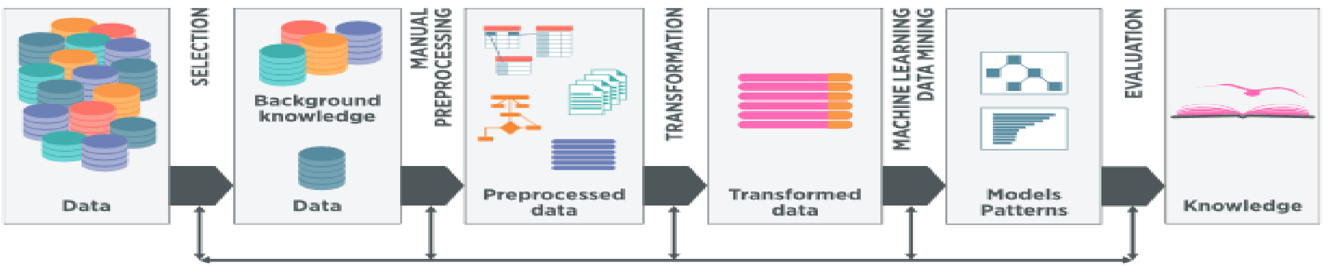
- Workflows are executable visual representations of procedures
  - divided into smaller chunks of code (components)
  - organized as sequences of connected components.
- Suitable for representing complex scientific pipelines
  - by explicitly modeling dependencies of components
- Building scientific workflows consists of simple operations on workflow elements (drag, drop, connect), suitable for non-experts





Second Generation Data Mining

- **Developed since 1990s:**
  - Focused on data mining tasks characterized by large datasets described by large numbers of attributes

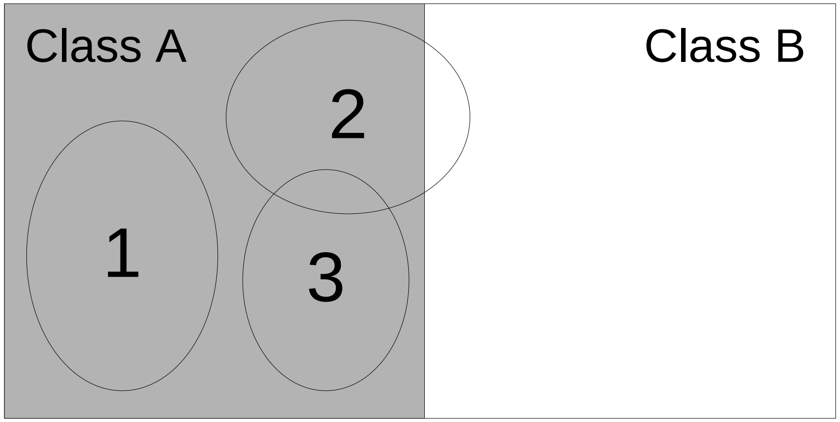


  - New conferences on practical aspects of data mining and knowledge discovery: KDD, PKDD, …
  - New learning tasks and efficient learning algorithms:
    - Learning descriptive patterns: association rule learning, subgroup discovery, …
    - Learning predictive models: Bayesian network learning,, relational data mining, statistical relational learning, SVMs, …

Subgroup Discovery

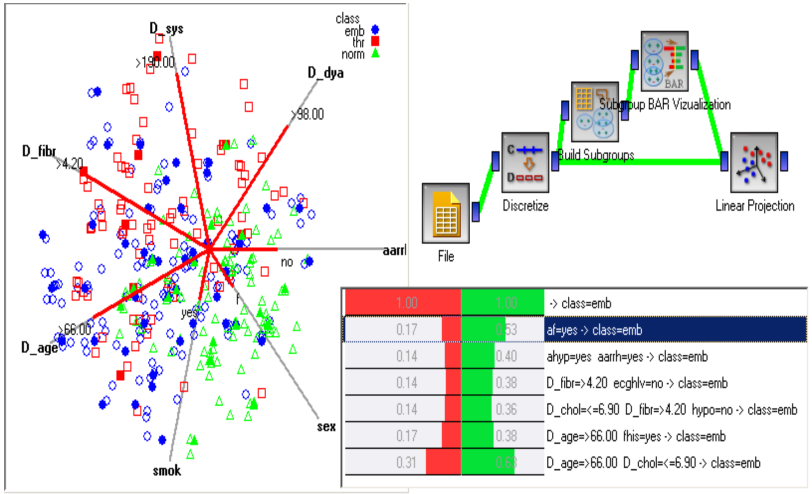
- Data transformation:
  - binary class values (positive vs. negative examples of Target class)
- Subgroup discovery:
  - a task in which individual interpretable patterns in the form of rules are induced from data, labeled by a predefined property of interest.
- SD algorithms learn several independent rules that describe groups of target class examples
  - subgroups must be large and significant

Person	Age	Spect. presc.	Astigm.	Tear prod.	Lenses
O1	17	myope	no	reduced	NO
O2	23	myope	no	normal	YES
O3	22	myope	yes	reduced	NO
O4	27	myope	yes	normal	YES
O5	19	hypermetrope	no	reduced	NO
O6-O13	...	...	...	...	...
O14	35	hypermetrope	no	normal	YES
O15	43	hypermetrope	yes	reduced	NO
O16	39	hypermetrope	yes	normal	NO
O17	54	myope	no	reduced	NO
O18	62	myope	no	normal	NO
O19-O23	...	...	...	...	...
O24	56	hypermetrope	yes	normal	NO



SD algorithms in Orange DM Platform

- **Orange** data mining toolkit
  - classification and subgroup discovery algorithms
  - data mining workflows
  - visualization



- **SD Algorithms in Orange**
  - SD (Gamberger & Lavrač, JAIR 2002)
  - Apriori-SD (Kavšek & Lavrač, AAI 2006)
  - CN2-SD (Lavrač et al., JMLR 2004)

Relational Data Mining

customer

ID	Zip	Sex	So St	In come	Age	Club	Resp
...	...	...	...	...	...	...	...
3478	34677	m	si	60-70	32	me	nr
3479	43666	f	ma	80-90	45	nm	re
...	...	...	...	...	...	...	...

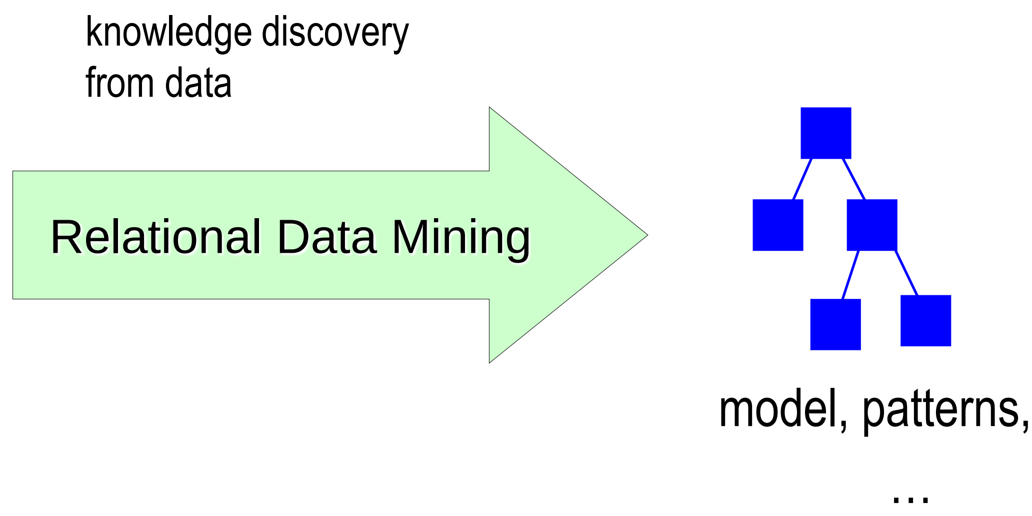
order

Customer ID	Order ID	Store ID	Delivery Mode	Paymt Mode
...	...	...	...	...
3478	2140267	12	regular	cash
3478	3446778	12	express	check
3478	4728386	17	regular	check
3479	3233444	17	express	credit
3479	3475886	12	regular	credit
...	...	...	...	...

store

Store ID	Size	Type	Location
...	...	...	...
12	small	franchise	city
17	large	indep	rural
...	...	...	...

Relational representation of customers, orders and stores.



Given: a relational database, a set of tables, sets of logical facts, a graph, …

Find: a classification model, a set of patterns

Relational Data Mining

- **ILP, relational learning, relational data mining**
  - Learning from complex relational databases

customer							
ID	Zip	Sex	SoSt	Income	Age	Club	Resp
...	...	...	...	...	...	...	...
3478	34677	m	si	60-70	32	me	nr
3479	43666	f	ma	80-90	45	nm	re
...	...	...	...	...	...	...	...

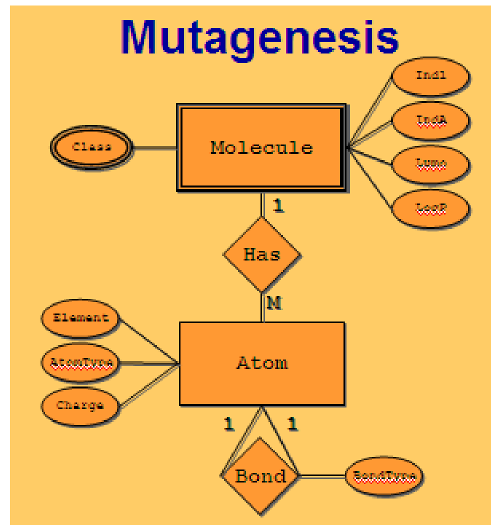
order				
Customer ID	Order ID	Store ID	Delivery Mode	Paymt Mode
...	...	...	...	...
3478	2140267	12	regular	cash
3478	3446778	12	express	check
3478	4728386	17	regular	check
3479	3233444	17	express	credit
3479	3475886	12	regular	credit
...	...	...	...	...

store			
Store ID	Size	Type	Location
...	...	...	...
12	small	franchise	city
17	large	indep	rural
...	...	...	...

Relational representation of customers, orders and stores.

Relational Data Mining

- **ILP, relational learning, relational data mining**
  - Learning from complex relational databases
  - Learning from complex structured data, e.g. molecules and their biochemical properties



customer							
ID	Zip	Sex	SoSt	Income	Age	Club	Resp
...	...	...	...	...	...	...	...
3478	34677	m	si	60-70	32	me	nr
3479	43666	f	ma	80-90	45	nm	re
...	...	...	...	...	...	...	...

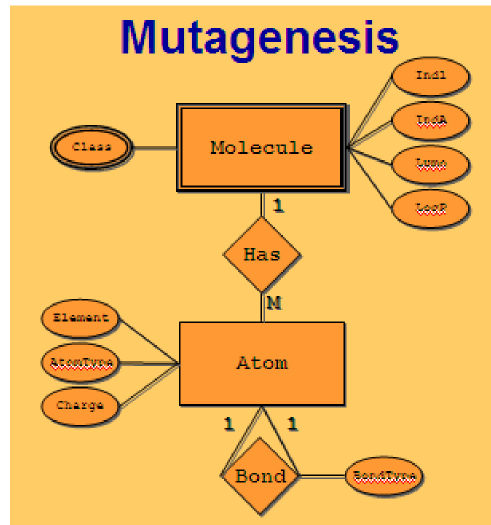
order				
Customer ID	Order ID	Store ID	Delivery Mode	Paymt Mode
...	...	...	...	...
3478	2140267	12	regular	cash
3478	3446778	12	express	check
3478	4728386	17	regular	check
3479	3233444	17	express	credit
3479	3475886	12	regular	credit
...	...	...	...	...

store			
Store ID	Size	Type	Location
...	...	...	...
12	small	franchise	city
17	large	indep	rural
...	...	...	...

Relational representation of customers, orders and stores.

Relational and Semantic Data Mining

- **ILP, relational learning, relational data mining**
  - Learning from complex relational databases
  - Learning from complex structured data, e.g. molecules and their biochemical properties
  - Learning by using domain knowledge in the form of ontologies = **semantic data mining**

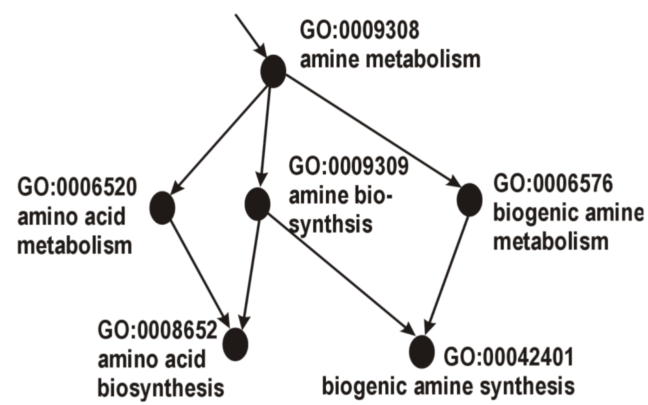


customer							
ID	Zip	S ex	So St	In come	A ge	Cl ub	Re sp
...	...	...	...	...	...	...	...
3478	34677	m	si	60-70	32	me	nr
3479	43666	f	ma	80-90	45	nm	re
...	...	...	...	...	...	...	...

order				
Customer ID	Order ID	Store ID	Delivery Mode	Paymt Mode
...	...	...	...	...
3478	2140267	12	regular	cash
3478	3446778	12	express	check
3478	4728386	17	regular	check
3479	3233444	17	express	credit
3479	3475886	12	regular	credit
...	...	...	...	...

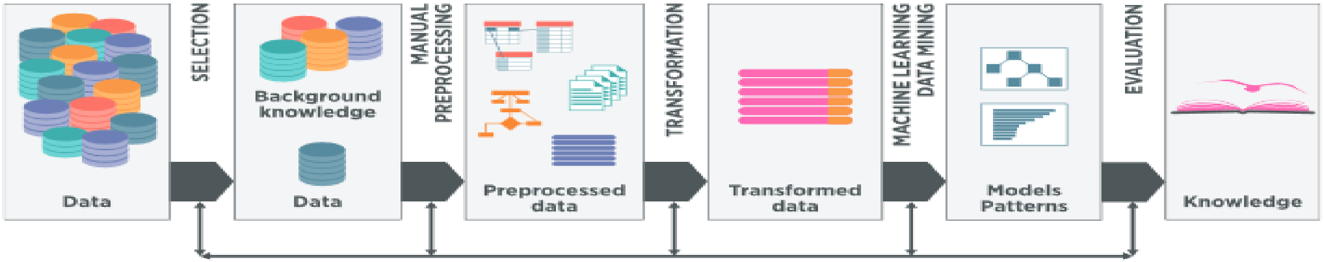
store			
Store ID	Size	Type	Location
...	...	...	...
12	small	franchise	city
17	large	indep	rural
...	...	...	...

Relational representation of customers, orders and stores.

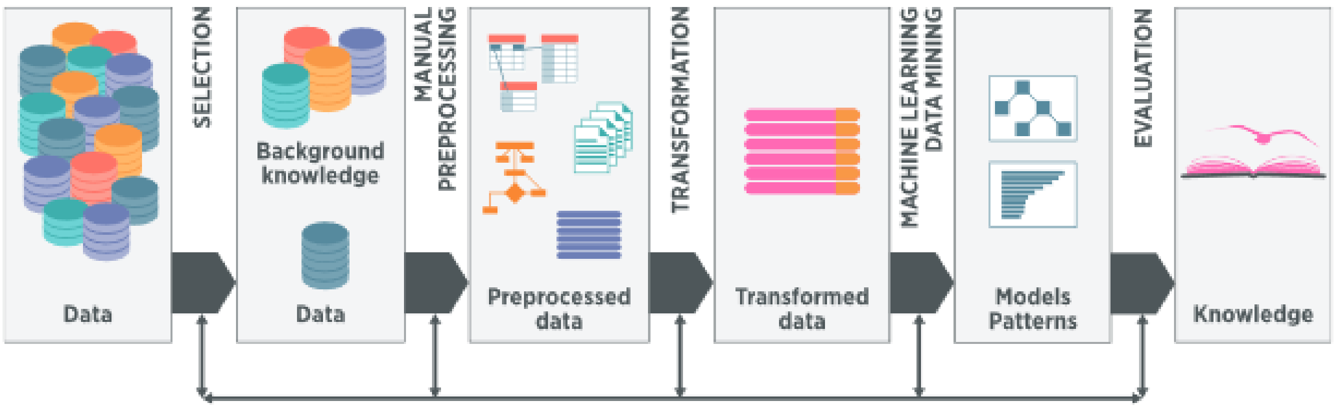


Third Generation Machine Learning

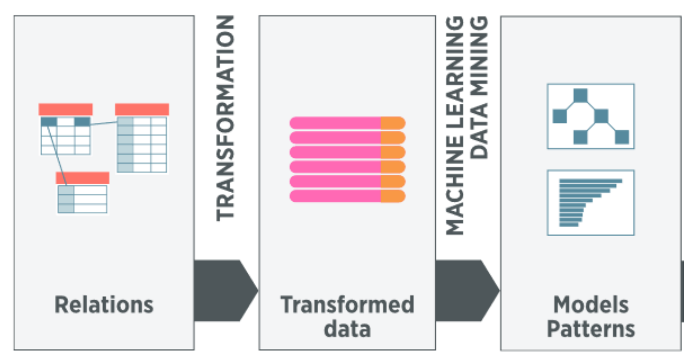
- **Developed since 2010s:**
  - Focused on big data analytics
  - Addressing complex data mining tasks and scenarios
  - New conferences on data science and big data analytics; e.g., IEEE Big Data, Complex networks, …
  - New learning tasks and efficient learning algorithms:
    - Analysis of dynamic data streams, Network analysis, Text mining, Semantic data analysis, …
  - Lots of emphasis on automated **data transformation**
    - Propositionalization of relational data, of heterogeneous information networks, …
    - Embedding of texts, networks, knowledge graphs, entities (features), … is highly popular in the last few years



Representation Learning



- Representation learning = Automated data transformation, performed on manually preprocessed data
- Transformation requires handling heterogeneous data
  - Data (feature vectors, documents, pictures, data streams, …)
  - Background knowledge (multi-relational data tables, networks, text corpora, …)
- Propositionalization:
  - Multi-relational data transformation



Propositionalization: Data transformation for Relational Learning

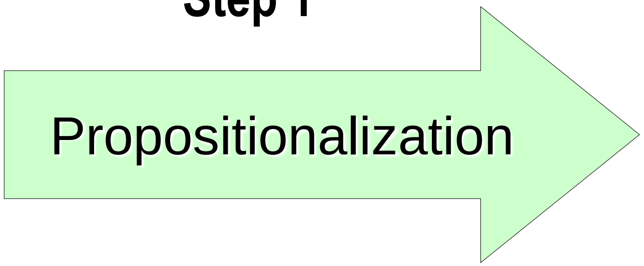
customer							
ID	Zip	Sex	SoSt	Income	Age	Club	Resp
...	...	...	...	...	...	...	...
3478	34677	m	si	60-70	32	me	nr
3479	43666	f	ma	80-90	45	nm	re
...	...	...	...	...	...	...	...

order				
Customer ID	Order ID	Store ID	Delivery Mode	Paymt Mode
...	...	...	...	...
3478	2140267	12	regular	cash
3478	3446778	12	express	check
3478	4728386	17	regular	check
3479	3233444	17	express	credit
3479	3475886	12	regular	credit
...	...	...	...	...

store			
Store ID	Size	Type	Location
...	...	...	...
12	small	franchise	city
17	large	indep	rural
...	...	...	...

Relational representation of customers, orders and stores.

Step 1



1. constructing relational features
2. constructing a propositional table

	f1	f2	f3	f4	f5	f6	...				...	fn
g1	1	0	0	1	1	1	0	0	1	0	1	1
g2	0	1	1	0	1	1	0	0	0	1	1	0
g3	0	1	1	1	0	0	1	1	0	0	0	1
g4	1	1	1	0	1	1	0	0	1	1	1	0
g5	1	1	1	0	0	1	0	1	1	0	1	0
g1	0	0	1	1	0	0	0	1	0	0	0	1
g2	1	1	0	0	1	1	0	1	0	1	1	1
g3	0	0	0	0	1	0	0	1	1	1	0	0
g4	1	0	1	1	1	0	1	0	0	1	0	1

Propositionalization: Data transformation for Relational Learning

customer							
ID	Zip	Sex	SoSt	Income	Age	Club	Resp
...	...	...	...	...	...	...	...
3478	34677	m	si	60-70	32	me	nr
3479	43666	f	ma	80-90	45	nm	re
...	...	...	...	...	...	...	...

order				
Customer ID	Order ID	Store ID	Delivery Mode	Paymt Mode
...	...	...	...	...
3478	2140267	12	regular	cash
3478	3446778	12	express	check
3478	4728386	17	regular	check
3479	3233444	17	express	credit
3479	3475886	12	regular	credit
...	...	...	...	...

store			
Store ID	Size	Type	Location
...	...	...	...
12	small	franchise	city
17	large	indep	rural
...	...	...	...

Relational representation of customers, orders and stores.

Step 1

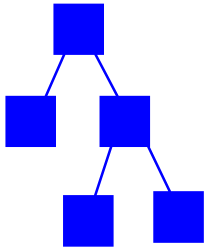
Propositionalization

	f1	f2	f3	f4	f5	f6	...				...	fn
g1	1	0	0	1	1	1	0	0	1	0	1	1
g2	0	1	1	0	1	1	0	0	0	1	1	0
g3	0	1	1	1	0	0	1	1	0	0	0	1
g4	1	1	1	0	1	1	0	0	1	1	1	0
g5	1	1	1	0	0	1	0	1	1	0	1	0
g1	0	0	1	1	0	0	0	1	0	0	0	1
g2	1	1	0	0	1	1	0	1	0	1	1	1
g3	0	0	0	0	1	0	0	1	1	1	0	0
g4	1	0	1	1	1	0	1	0	0	1	0	1

	f1	f2	f3	f4	f5	f6	...				...	fn
g1	1	0	0	1	1	1	0	0	1	0	1	1
g2	0	1	1	0	1	1	0	0	0	1	1	0
g3	0	1	1	1	0	0	1	1	0	0	0	1
g4	1	1	1	0	1	1	0	0	1	1	1	0
g5	1	1	1	0	0	1	0	1	1	0	1	0
g1	0	0	1	1	0	0	0	1	0	0	0	1
g2	1	1	0	0	1	1	0	1	0	1	1	1
g3	0	0	0	0	1	0	0	1	1	1	0	0
g4	1	0	1	1	1	0	1	0	0	1	0	1

Step 2

Machine Learning



model, patterns, …

Propositionalization:
Data transformation for Relational Learning

customer							
ID	Zip	Sex	So St	Income	Age	Club	Resp
...	...	...	...	...	...	...	...
3478	34677	m	si	60-70	32	me	nr
3479	43666	f	ma	80-90	45	nm	re
...	...	...	...	...	...	...	...

order				
Customer ID	Order ID	Store ID	Delivery Mode	Paymt Mode
...	...	...	...	...
3478	2140267	12	regular	cash
3478	3446778	12	express	check
3478	4728386	17	regular	check
3479	3233444	17	express	credit
3479	3475886	12	regular	credit
...	...	...	...	...

store			
Store ID	Size	Type	Location
...	...	...	...
12	small	franchise	city
17	large	indep	rural
...	...	...	...

Relational representation of customers, orders and stores.

Step 1

Propositionalization

1. construct relational features
2. construct a propositional table

	f1	f2	f3	f4	f5	f6	...				...	fn
g1	1	0	0	1	1	1	0	0	1	0	1	1
g2	0	1	1	0	1	1	0	0	0	1	1	0
g3	0	1	1	1	0	0	1	1	0	0	0	1
g4	1	1	1	0	1	1	0	0	1	1	1	0
g5	1	1	1	0	0	1	0	1	1	0	1	0
g1	0	0	1	1	0	0	0	1	0	0	0	1
g2	1	1	0	0	1	1	0	1	0	1	1	1
g3	0	0	0	0	1	0	0	1	1	1	0	0
g4	1	0	1	1	1	0	1	0	0	1	0	1

	f1	f2	f3	f4	f5	f6	...				...	fn
g1	1	0	0	1	1	1	0	0	1	0	1	1
g2	0	1	1	0	1	1	0	0	0	1	1	0
g3	0	1	1	1	0	0	1	1	0	0	0	1
g4	1	1	1	0	1	1	0	0	1	1	1	0
g5	1	1	1	0	0	1	0	1	1	0	1	0
g1	0	0	1	1	0	0	0	1	0	0	0	1
g2	1	1	0	0	1	1	0	1	0	1	1	1
g3	0	0	0	0	1	0	0	1	1	1	0	0
g4	1	0	1	1	1	0	1	0	0	1	0	1

Step 2

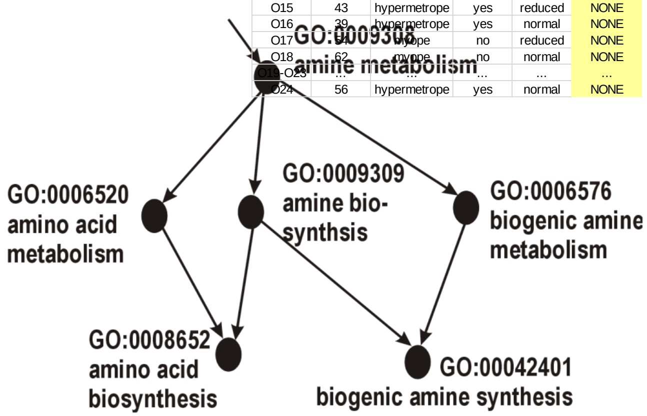
Subgroup discovery

```
target(A) :-
   'Doctor'(A), 'Italy'(A).

target(A) :-
   'Public'(A), 'Gold'(A).

target(A) :-
   'Poland'(A), 'Deposit'(A), 'Gold'(A).

target(A) :-
   'Germany'(A), 'Insurance'(A).

target(A) :-
   'Service'(A), 'Germany'(A).
```

patterns (set of rules)

Propositionalization: Data transformation for Semantic Data Mining

Person	Age	Spect. presc.	Astigm.	Tear prod.	Lenses
O1	17	myope	no	reduced	NONE
O2	23	myope	no	normal	SOFT
O3	22	myope	yes	reduced	NONE
O4	27	myope	yes	normal	HARD
O5	19	hypermetrope	no	reduced	NONE
O6-O13	...	...	...	...	...
O14	35	hypermetrope	no	normal	SOFT
O15	43	hypermetrope	yes	reduced	NONE
O16	39	hypermetrope	yes	normal	NONE
O17	54	myope	no	reduced	NONE
O18	62	myope	no	normal	NONE
O19-O23	...	...	...	...	...
O24	56	hypermetrope	yes	normal	NONE



Step 1

Propositionalization

	f1	f2	f3	f4	f5	f6	...				...	fn
g1	1	0	0	1	1	1	0	0	1	0	1	1
g2	0	1	1	0	1	1	0	0	0	1	1	0
g3	0	1	1	1	0	0	1	1	0	0	0	1
g4	1	1	1	0	1	1	0	0	1	1	1	0
g5	1	1	1	0	0	1	0	1	1	0	1	0
g1	0	0	1	1	0	0	0	1	0	0	0	1
g2	1	1	0	0	1	1	0	1	0	1	1	1
g3	0	0	0	0	1	0	0	1	1	1	0	0
g4	1	0	1	1	1	0	1	0	0	1	0	1

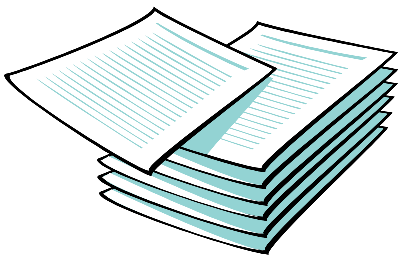
1. constructing relational features
2. constructing a propositional table

The approach: Using relational subgroup discovery in the SDM context

- General purpose system **RSD** for **Relational Subgroup Discovery**, using a propositionalization approach to relational data mining
- Applied to semantic data mining in a biomedical application by using the Gene Ontology as background knowledge in analyzing microarray data

Železny and Lavrac, MLJ 2006

Text mining: Viewed in propositionalization context: BoW data transformation



Step 1

BoW vector construction

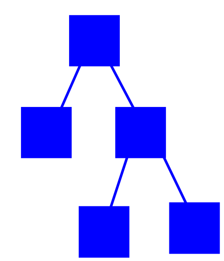
1. BoW features construction
2. Table of BoW vectors construction

Document	Word1	Word2	…	WordN	Class
d1	1	1	0	1	NO
d2	1	1	0	0	YES
d3	1	1	1	1	NO
d4	1	1	1	0	YES
d5	1	0	0	1	NO
d6-d13	…	…	…	…	…
d14	0	0	0	0	YES
d15	0	0	1	1	NO
d16	0	0	1	0	NO
d17	0	1	0	1	NO
d18	0	1	0	0	NO
d19-d23	…	…	…	…	…
d24	0	0	1	0	NO

Document	Word1	Word2	…	WordN	Class
d1	1	1	0	1	NO
d2	1	1	0	0	YES
d3	1	1	1	1	NO
d4	1	1	1	0	YES
d5	1	0	0	1	NO
d6-d13	…	…	…	…	…
d14	0	0	0	0	YES
d15	0	0	1	1	NO
d16	0	0	1	0	NO
d17	0	1	0	1	NO
d18	0	1	0	0	NO
d19-d23	…	…	…	…	…
d24	0	0	1	0	NO

Step 2

Data Mining



model, patterns, clusters, …

BoW construction: Feature weights and Cosine similarity between document vectors

- Each document D is represented as a vector of TF-IDF weights

$$tfidf(w) = tf \,.\, \log\left(\frac{N}{df(w)}\right)$$

- Similarity between two vectors is estimated by the similarity between their vector representations (cosine of the angle between the two vectors):

$$Similarity\ (D_1, D_2) = \frac{\sum_i x_{1i} x_{2i}}{\sqrt{\sum_j x_j^2}\sqrt{\sum_k x_k^2}}$$

Embeddings-based Data Transformation for Text mining

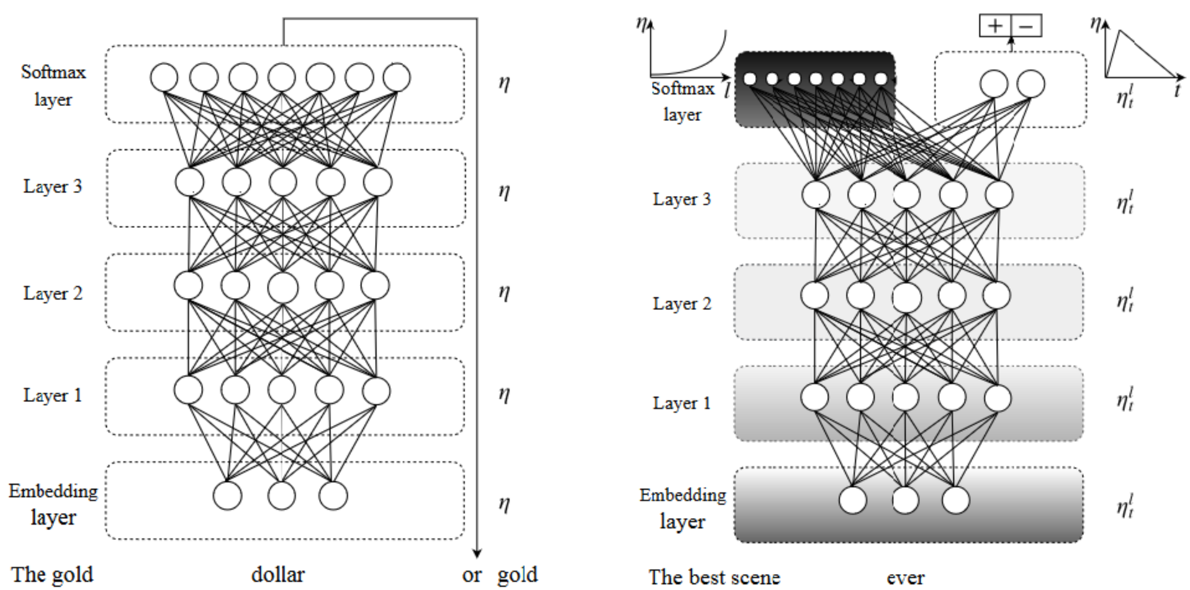
- Corpus embedding, Document embedding, Sentence embedding, **word embedding** (e.g., word2vec)
  - Transforming documents by projecting documents into vectors (rows of a data table)

Document	Word1	Word2	…	WordN	Class
d1	1	1	0	1	NO
d2	1	1	0	0	YES
d3	1	1	1	1	NO
d4	1	1	1	0	YES
d5	1	0	0	1	NO
d6-d13	…	…	…	…	…
d14	0	0	0	0	YES
d15	0	0	1	1	NO
d16	0	0	1	0	NO
d17	0	1	0	1	NO
d18	0	1	0	0	NO
d19-d23	…	…	…	…	…
d24	0	0	1	0	NO

Embeddings-based Data Transformation for Text mining

- Corpus embedding, Document embedding, Sentence embedding, **word embedding** (e.g., word2vec)
  - Transforming documents by projecting documents into vectors (rows of a data table)
  - Weights correspond to weights in the embedding layer of a neural network

Document	Word1	Word2	…	WordN	Class
d1	1	1	0	1	NO
d2	1	1	0	0	YES
d3	1	1	1	1	NO
d4	1	1	1	0	YES
d5	1	0	0	1	NO
d6-d13	…	…	…	…	…
d14	0	0	0	0	YES
d15	0	0	1	1	NO
d16	0	0	1	0	NO
d17	0	1	0	1	NO
d18	0	1	0	0	NO
d19-d23	…	…	…	…	…
d24	0	0	1	0	NO

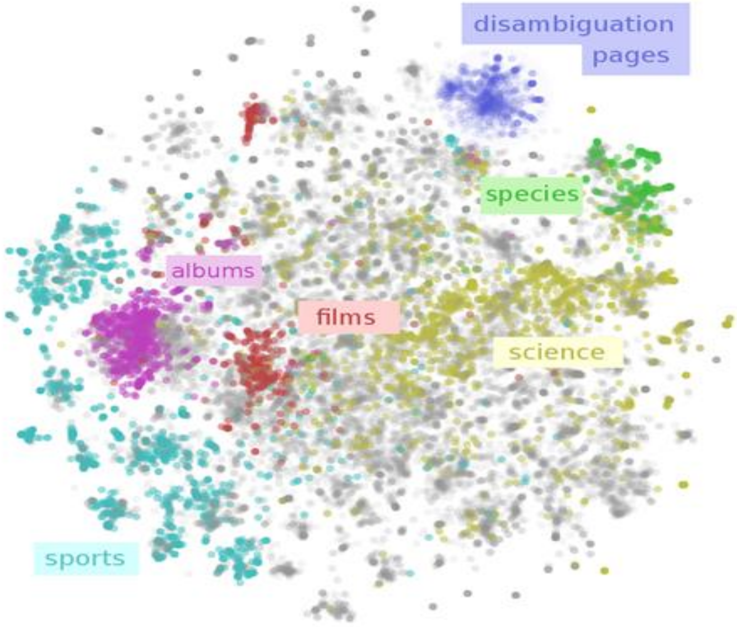


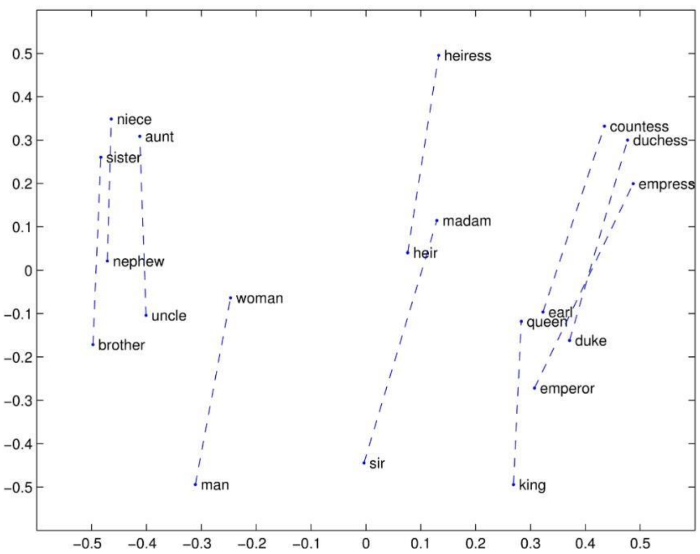
LM pre-training

Classifier fine-tuning

Embedding-based Data Transformation for Text mining

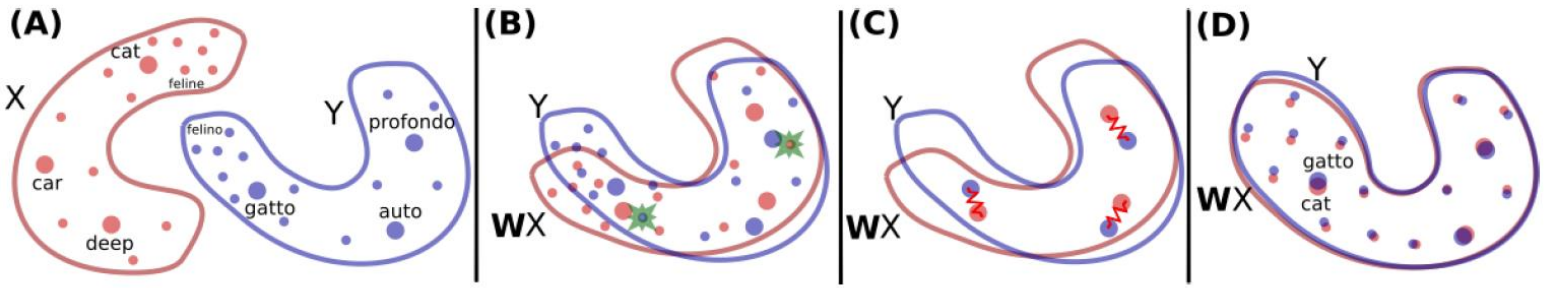
- Corpus embedding, Document embedding, Sentence embedding, **word embedding**, …
  - Representations of word meaning obtained from corpus statistics
  - Spatial relationships correspond to linguistic relationships





Cross-domain or cross-lingual Embeddings-based Data Transformation for Text mining

- Aligning embedding spaces across domains or languages



- **EMBEDDIA** H2020 project (2019-2021) coordinated by Jožef Stefan Institute: **Cross-lingual embeddings for less-represented languages in news media industry**
  - developing new language models for less represented languages
  - Using advanced embedding models like GloVe and contextual embedding models like **Bert** in news analysis applications and in UGC commentary filtering

Part I: Summary

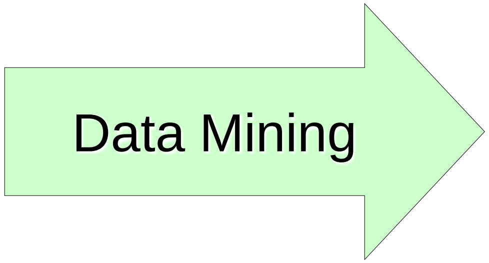
- KDD is the overall process of discovering useful knowledge in data
  - many steps including data preparation, cleaning, transformation, pre-processing
- Data Mining is the data analysis phase in KDD
  - DM takes only 15%-25% of the effort of the overall KDD process
  - employing techniques from machine learning and statistics
- Predictive and descriptive induction have different goals: classifier vs. pattern discovery
- Many application areas, many powerful tools available

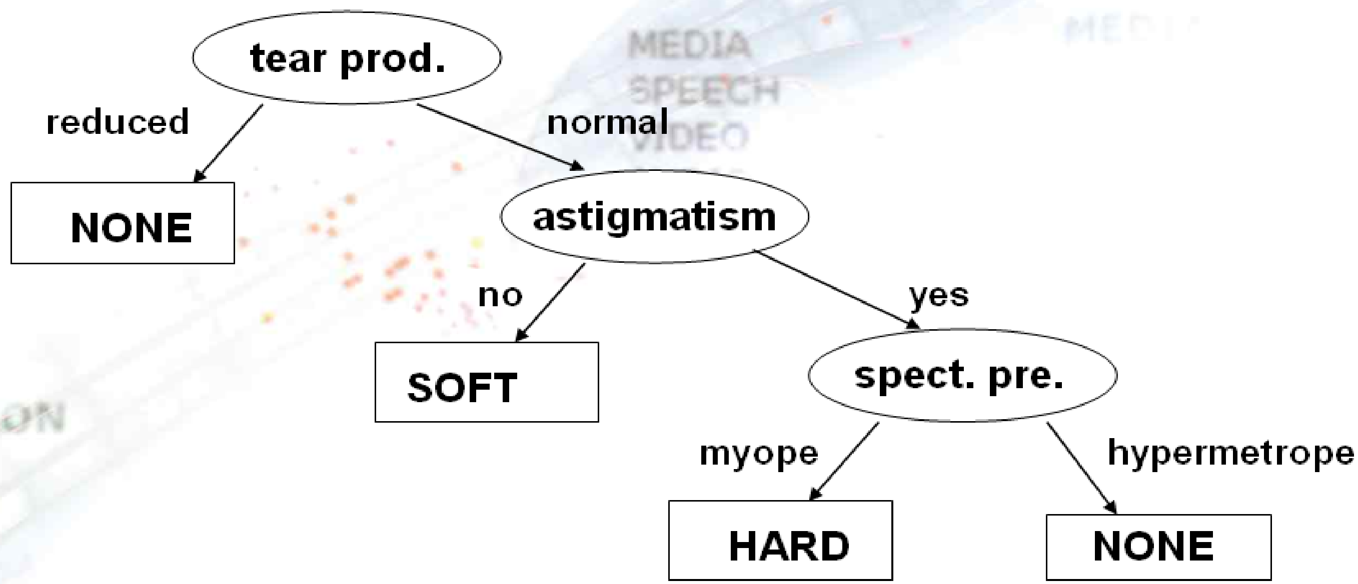
Outline

- Introduction to Machine Learning and Data Mining: Techniques overview
- Rule learning
- Relational learning: Propositionalization
- Semantic data mining
- Relational learning: Wordification

Learning a classification model from contact lens data

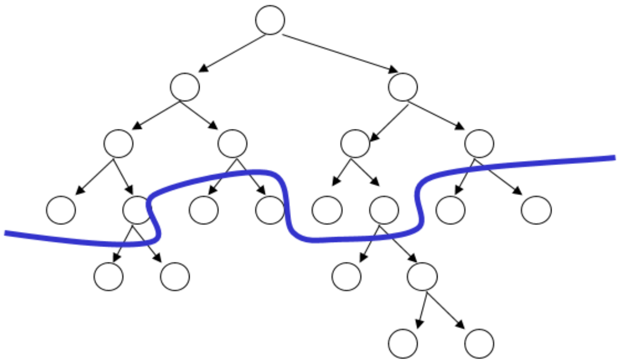
Person	Age	Spect. presc.	Astigm.	Tear prod.	Lenses
O1	young	myope	no	reduced	NONE
O2	young	myope	no	normal	SOFT
O3	young	myope	yes	reduced	NONE
O4	young	myope	yes	normal	HARD
O5	young	hypermetrope	no	reduced	NONE
O6-O13	...	...	...	...	...
O14	pre-presbyo	hypermetrope	no	normal	SOFT
O15	pre-presbyo	hypermetrope	yes	reduced	NONE
O16	pre-presbyo	hypermetrope	yes	normal	NONE
O17	presbyopic	myope	no	reduced	NONE
O18	presbyopic	myope	no	normal	NONE
O19-O23	...	...	...	...	...
O24	presbyopic	hypermetrope	yes	normal	NONE

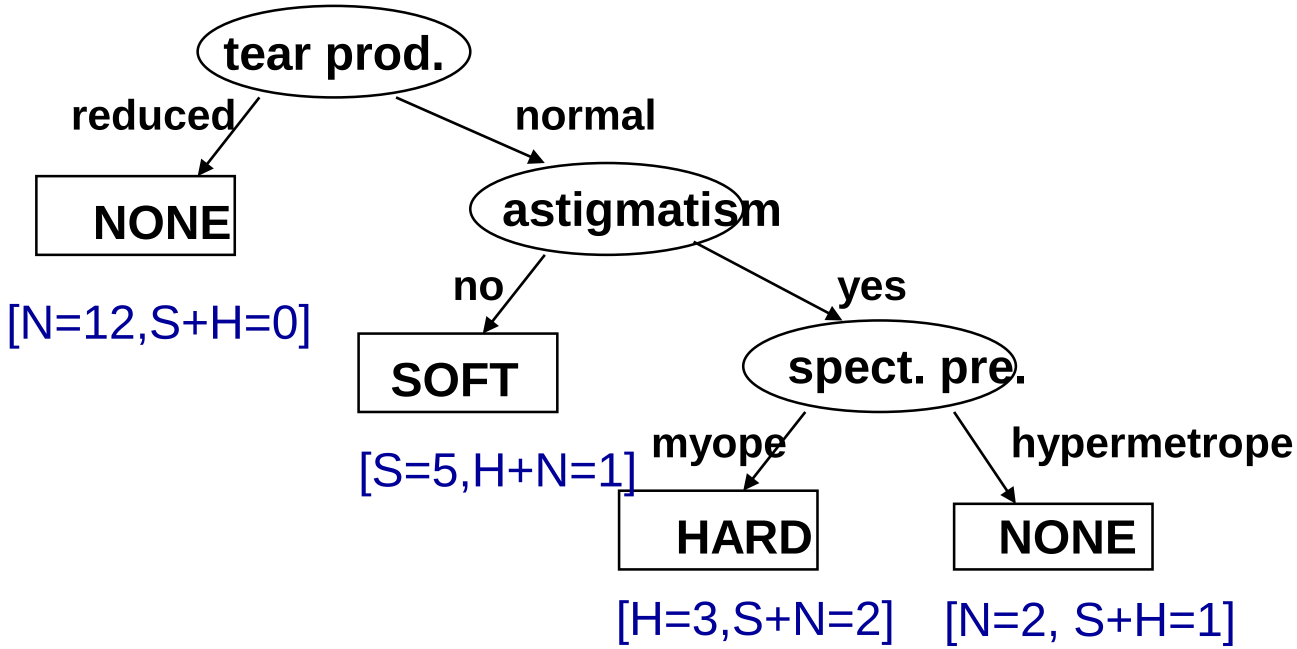




Decision tree learning and pruning

- Top-down construction of decision trees
- Tree pruning to avoid data overfitting
- Pruned trees are
    - less accurate on training data
    - more accurate o in classifying unseen data

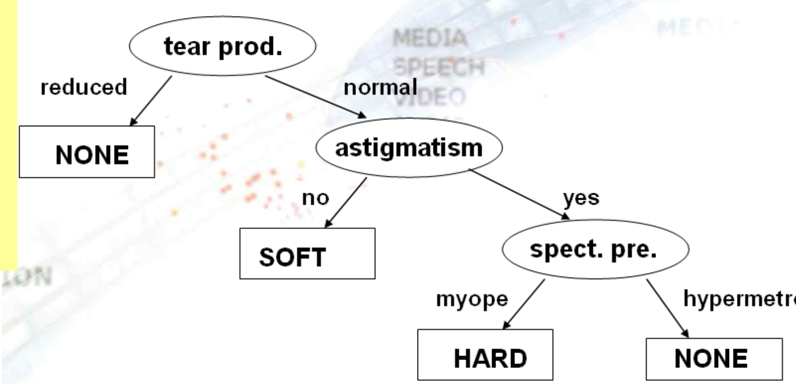




Learning a classification model from contact lens data

Person	Age	Spect. presc.	Astigm.	Tear prod.	Lenses
O1	17	myope	no	reduced	NONE
O2	23	myope	no	normal	SOFT
O3	22	myope	yes	reduced	NONE
O4	27	myope	yes	normal	HARD
O5	19	hypermetrope	no	reduced	NONE
O6-O13	...	...	...	...	...
O14	35	hypermetrope	no	normal	SOFT
O15	43	hypermetrope	yes	reduced	NONE
O16	39	hypermetrope	yes	normal	NONE
O17	54	myope	no	reduced	NONE
O18	62	myope	no	normal	NONE
O19-O23	...	...	...	...	...
O24	56	hypermetrope	yes	normal	NONE

Data Mining



lenses=NONE ← tear production=red

lenses=NONE ← tear production=normal AND astigmatism=yes AND spect. pre.=hypermetrope

lenses=SOFT ← tear production=normal AND astigmatism=no

lenses=HARD ← tear production=normal AND astigmatism=yes AND spect. pre.=myope

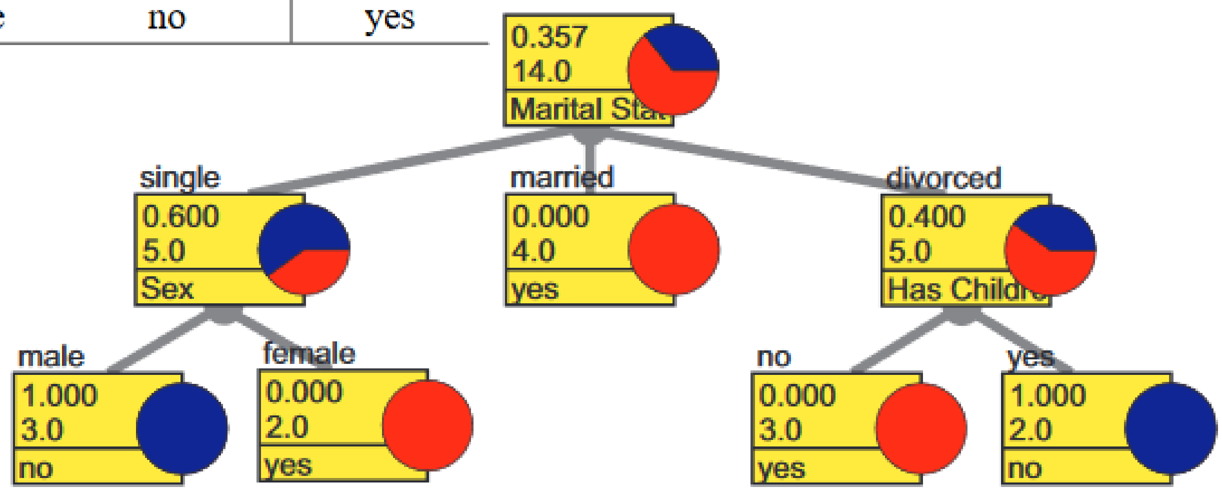
lenses=NONE ←

Converting decision tree to rules, and rule post-pruning (Quinlan 1993)

- Very frequently used method, e.g., in C4.5 and J48
- Procedure:
  - grow a full tree (allowing overfitting)
  - convert the tree to an equivalent set of rules
  - prune each rule independently of others
  - sort final rules into a desired sequence for use

Learning decision trees
Survey data

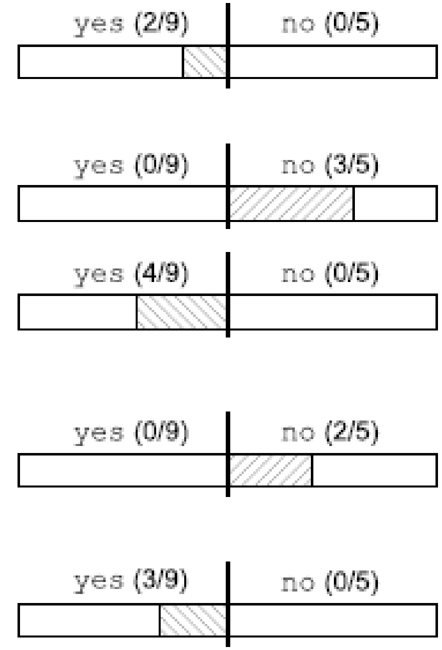
Education	Marital Status	Sex	Has Children	Approved
primary	single	male	no	no
primary	single	male	yes	no
primary	married	male	no	yes
university	divorced	female	no	yes
university	married	female	yes	yes
secondary	single	male	no	no
university	single	female	no	yes
secondary	divorced	female	no	yes
secondary	single	female	yes	yes
secondary	married	male	yes	yes
primary	married	female	no	yes
secondary	divorced	male	yes	no
university	divorced	female	yes	no
secondary	divorced	male	no	yes

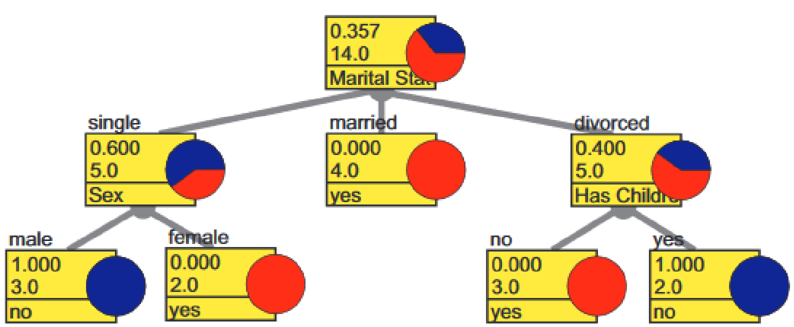


Transforming trees to rules: Survey data

Education	Marital Status	Sex	Has Children	Approved
primary	single	male	no	no
primary	single	male	yes	no
primary	married	male	no	yes
university	divorced	female	no	yes
university	married	female	yes	yes
secondary	single	male	no	no
university	single	female	no	yes
secondary	divorced	female	no	yes
secondary	single	female	yes	yes
secondary	married	male	yes	yes
primary	married	female	no	yes
secondary	divorced	male	yes	no
university	divorced	female	yes	no
secondary	divorced	male	no	yes

```
IF   MaritalStatus = single
 AND Sex = female
THEN Approved = yes

IF   MaritalStatus = single
 AND Sex = male
THEN Approved = no

IF   MaritalStatus = married
THEN Approved = yes

IF   MaritalStatus = divorced
 AND HasChildren = yes
THEN Approved = no

IF   MaritalStatus = divorced
 AND HasChildren = no
THEN Approved = yes
```

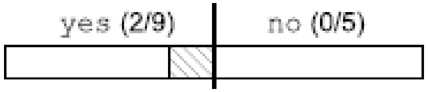




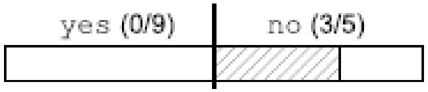
Pruning classification rules: Survey data

Education	Marital Status	Sex	Has Children	Approved
primary	single	male	no	no
primary	single	male	yes	no
primary	married	male	no	yes
university	divorced	female	no	yes
university	married	female	yes	yes
secondary	single	male	no	no
university	single	female	no	yes
secondary	divorced	female	no	yes
secondary	single	female	yes	yes
secondary	married	male	yes	yes
primary	married	female	no	yes
secondary	divorced	male	yes	no
university	divorced	female	yes	no
secondary	divorced	male	no	yes

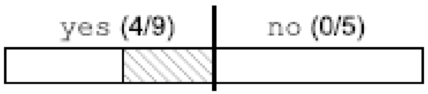
```
IF   MaritalStatus = single
 AND Sex = female
THEN Approved = yes
```



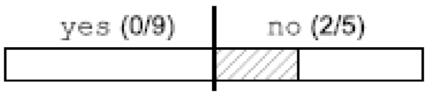

```
IF   MaritalStatus = single
 AND Sex = male
THEN Approved = no
```



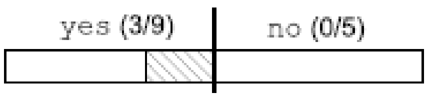

```
IF   MaritalStatus = married
THEN Approved = yes
```



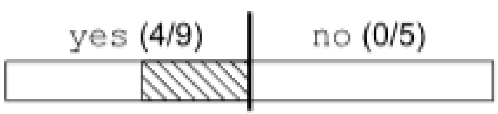

```
IF   MaritalStatus = divorced
 AND HasChildren = yes
THEN Approved = no
```



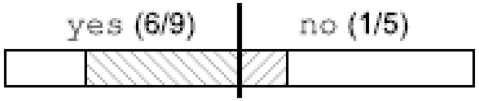

```
IF   MaritalStatus = divorced
 AND HasChildren = no
THEN Approved = yes
```



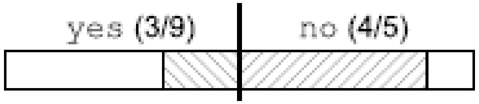

```
IF   MaritalStatus = married
THEN Approved = yes
```




```
IF   Sex = female
THEN Approved = yes
```

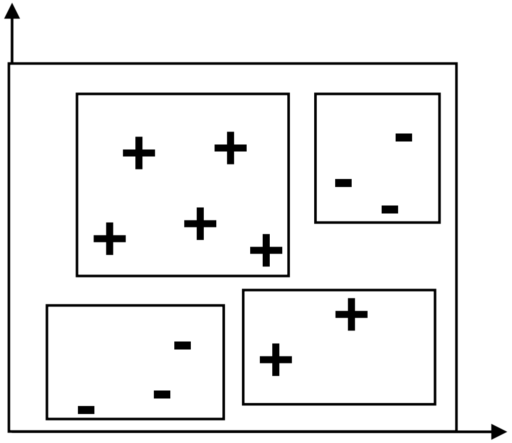



```
IF   Sex = male
THEN Approved = no
```




```
DEFAULT  Approved = yes
```

Covering algorithm for binary classification problems (AQ, Michalski 1969,86)



Given examples of 2 classes C_1, C_2

for each class Ci **do**

– Ei := Pi U Ni (Pi pos., Ni neg.)

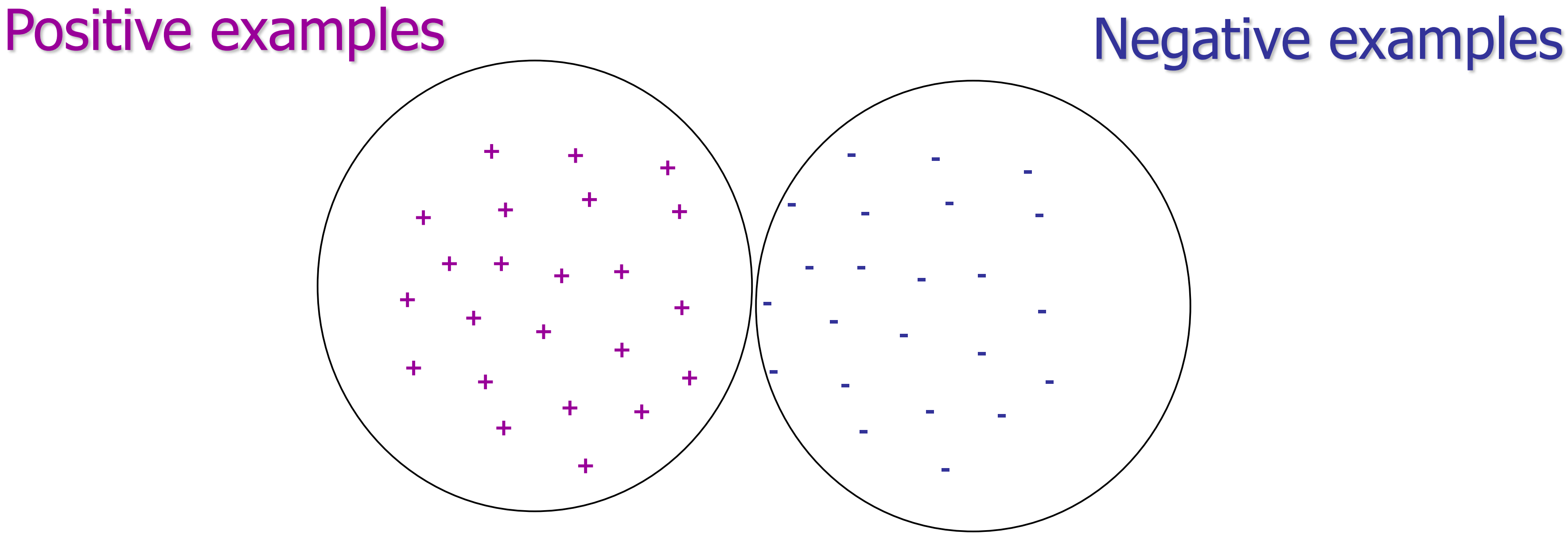
– RuleBase(Ci) := empty

– **repeat {learn-set-of-rules}**

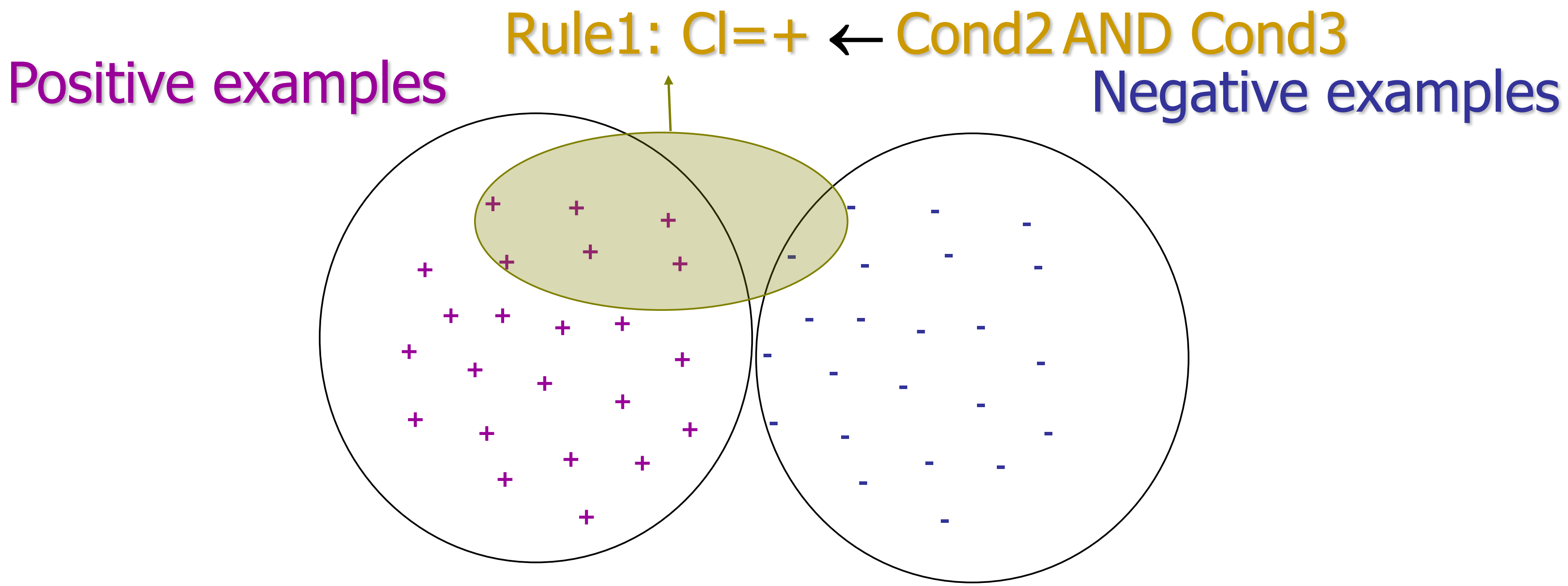
- **learn-one-rule** R covering some positive examples and no negatives
- add R to RuleBase(Ci)
- delete from Pi all pos. ex. covered by R

– **until** Pi = empty

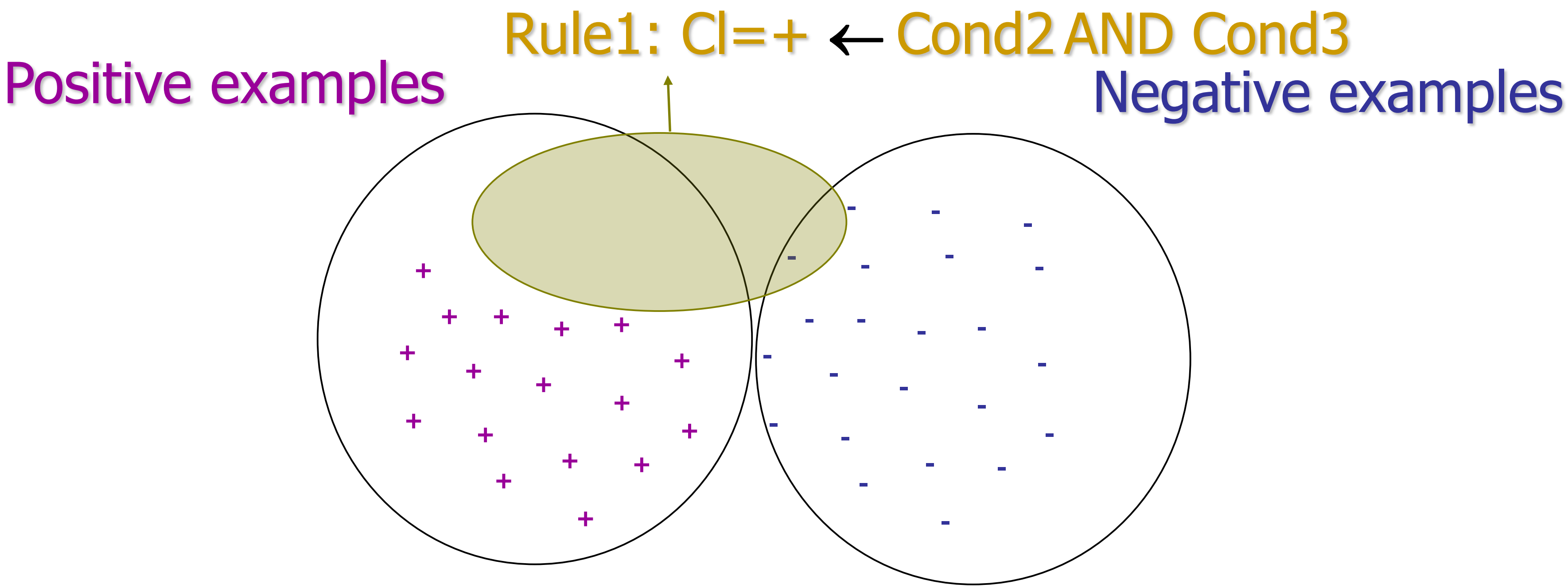
Covering algorithm



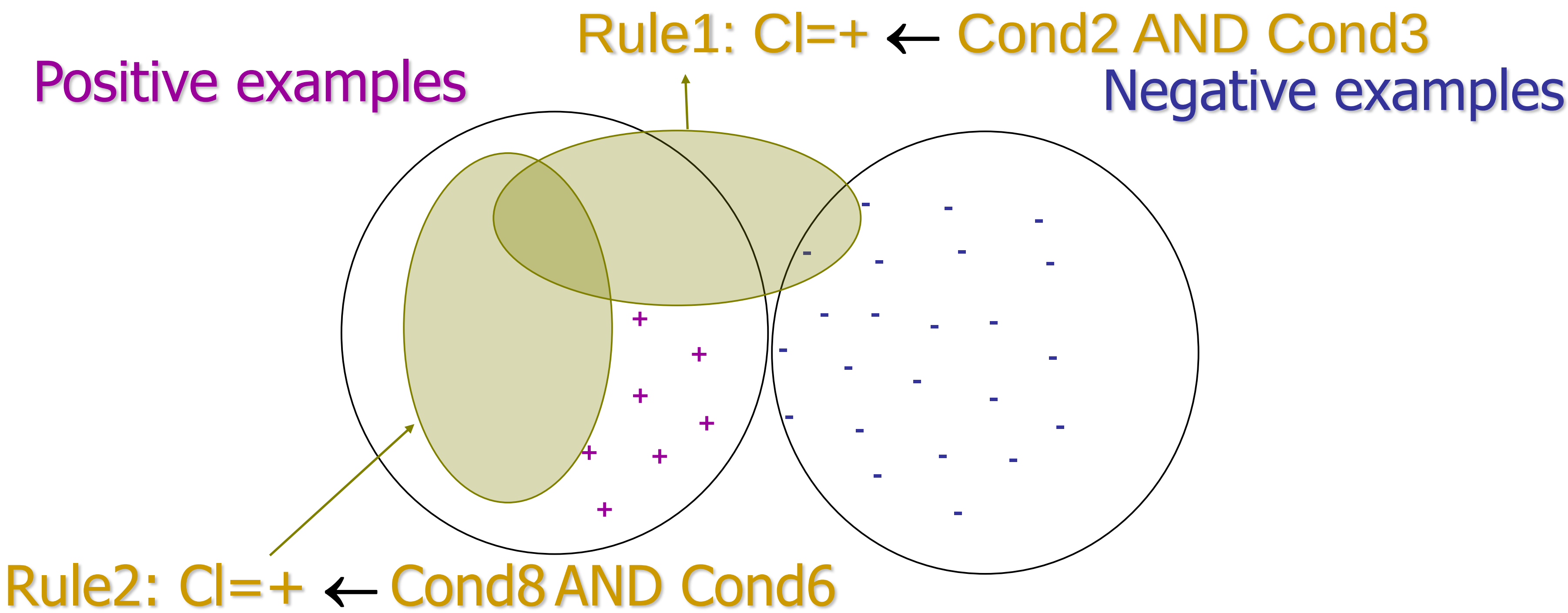
Covering algorithm



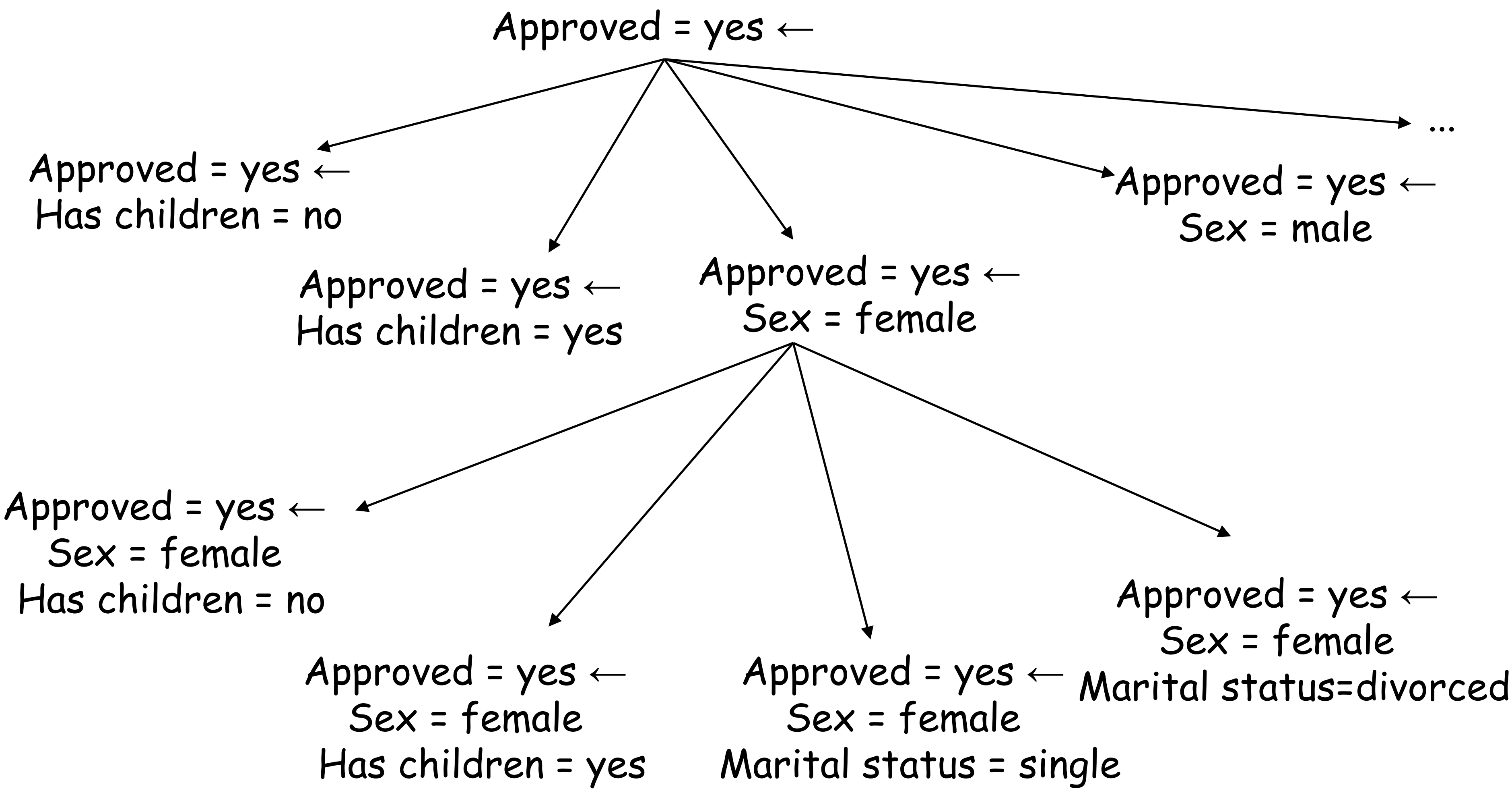
Covering algorithm



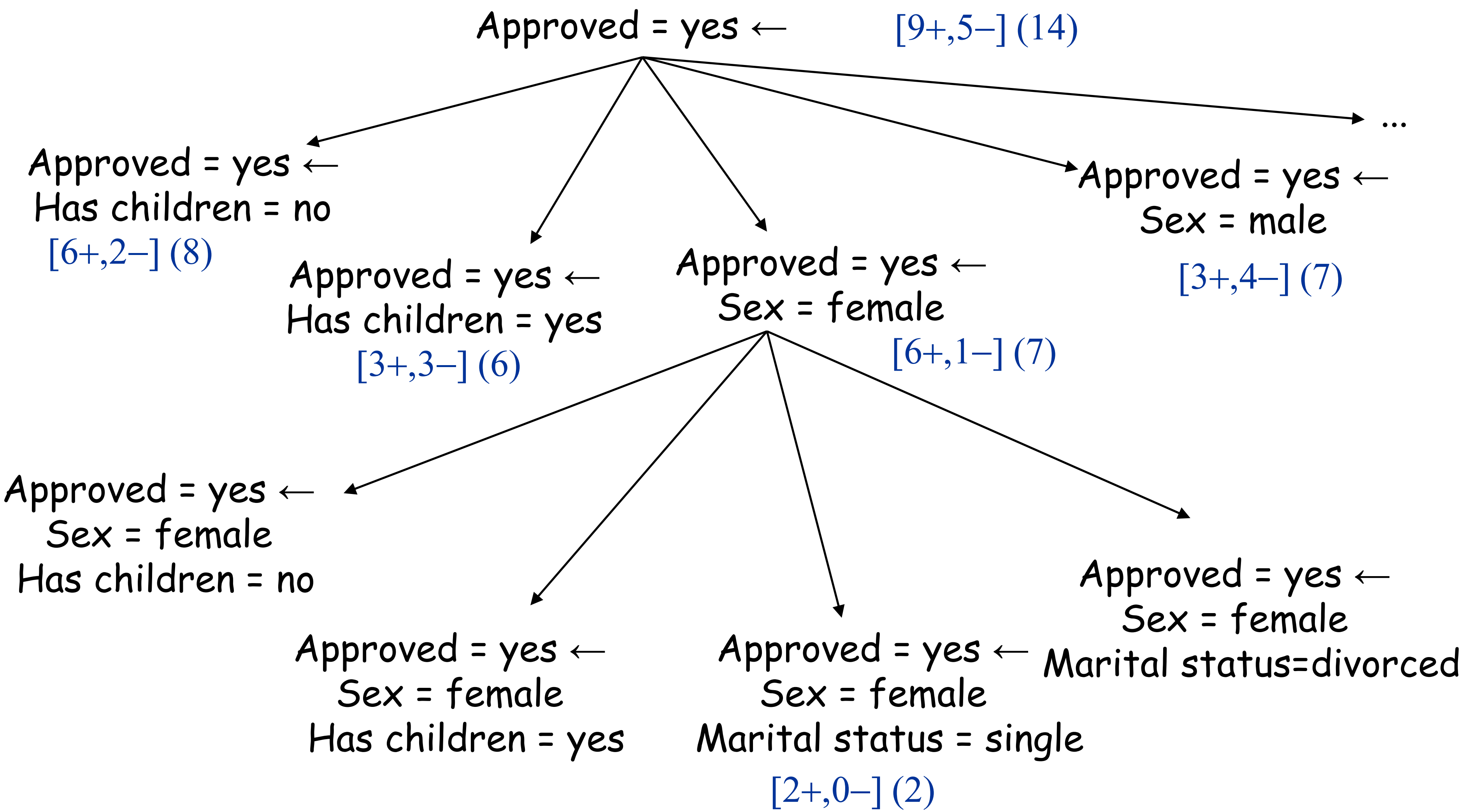
Covering algorithm



Learn-one-rule as heuristic search: Survey data



Learn-one-rule as heuristic search: Survey data



Rule evaluation measures

- Evaluation measures for rules Cl ← Cond
  - aimed at maximizing classification accuracy
  - minimizing Error = 1 – Accuracy
  - avoiding overfitting
- Expected accuracy/precision: A(R) = p(Cl|Cond)
- Traded off measures:

  – **Relative accuracy/precision***:* RAcc(Cl ← Cond) = p(Cl | Cond) – p(Cl)
  trade-off against the "default" accuracy of rule **Cl ←true**
  (e.g., 68% accuracy is OK if there are 20% examples of that class in the training set, but bad if there are 80%)

  – **Weighted relative accuracy:** WRAcc(R) = p(Cond).(p(Cl | Cond) - p(Cl))
  trades off coverage and relative accuracy

  – **Accuracy gain:** AG(R',R) = p(Cl | NewCond) - p(Cl | CurrentCond)
  increase in expected accuracy after rule specialization

Ordered set of rules: if-then-else rules

- rule Class IF Conditions is learned by first determining Conditions and then Class
- **Notice:** mixed sequence of classes C1, …, Cn in RuleBase
- **But: ordered** execution when classifying a new instance: rules are sequentially tried and the first rule that `fires' (covers the example) is used for classification
- **Decision list {R1, R2, R3, …, D}:** rules Ri are interpreted as **if-then-else** rules
- If no rule fires, then DefaultClass (majority class in E_{cur})

Sequential covering algorithm

- RuleBase := empty
- E_{cur}:= E
- **repeat**
  - learn-one-rule R
  - RuleBase := RuleBase U R
  - E_{cur} := E_{cur} - {examples covered and correctly classified by R} **(DELETE ONLY POS. EX.!)**
  - **until** performance(R, E_{cur}) < ThresholdR
- RuleBase := sort RuleBase by performance(R,E)
- return RuleBase

Learn ordered set of rules (CN2, Clark and Niblett 1989)

- RuleBase := empty
- E_{cur}:= E
- **repeat**
  - learn-one-rule R
  - RuleBase := RuleBase U R
  - E_{cur} := E_{cur} - {all examples covered by R}
    (NOT ONLY POS. EX.!)
- **until** performance(R, E_{cur}) < ThresholdR
- RuleBase := sort RuleBase by performance(R,E)
- RuleBase := RuleBase U DefaultRule(E_{cur})

Learn-one-rule: Beam search in CN2

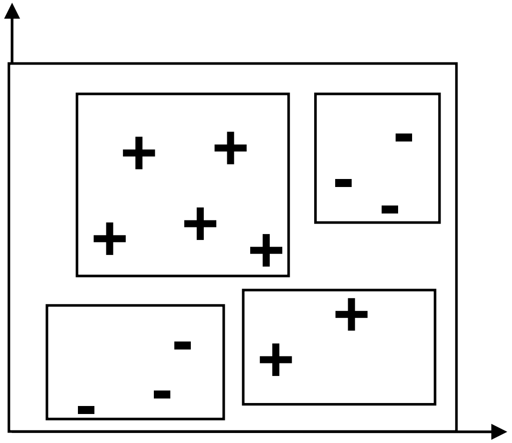
- Beam search in CN2 learn-one-rule algo.:
  - construct BeamSize of best rule bodies (conjunctive conditions) that are statistically significant
  - BestBody - min. entropy of examples covered by Body
  - construct best rule R := Head $\leftarrow$ BestBody by adding majority class of examples covered by BestBody in rule Head

Variations

- Sequential vs. simultaneous covering of data (as in TDIDT): choosing between attribute-values vs. choosing attributes
- Learning rules vs. learning decision trees and converting them to rules
- Pre-pruning vs. post-pruning of rules
- What statistical evaluation functions to use
- Probabilistic classification

- Best performing rule learning algorithm: Ripper
- JRip implementation of Ripper in WEKA, available in ClowdFlows

Covering algorithm for multiclass learning (AQ, Michalski 1969,86)



Given examples of N classes C_1, …, C_N

for each class Ci **do**

- Ei := Pi U Ni (Pi pos., Ni neg.)
- RuleBase(Ci) := empty
- **repeat {learn-set-of-rules}**
  - **learn-one-rule** R covering some positive examples and no negatives
  - add R to RuleBase(Ci)
  - delete from Pi all pos. ex. covered by R
- **until** Pi = empty

Multi-class learning: One-against-all learning strategy

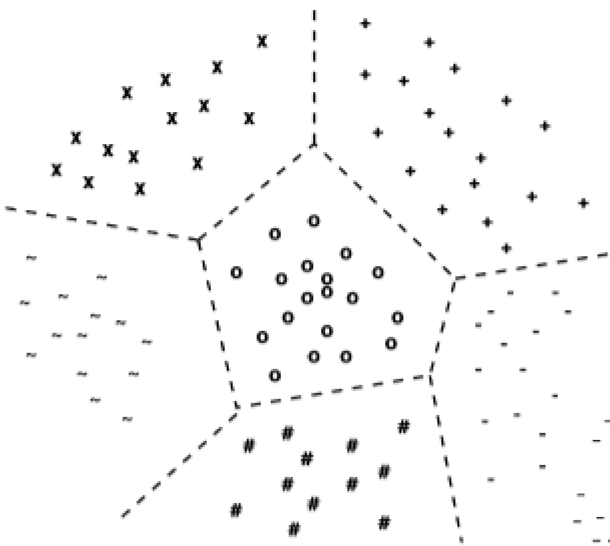


Fig. 10.2: A multiclass classification

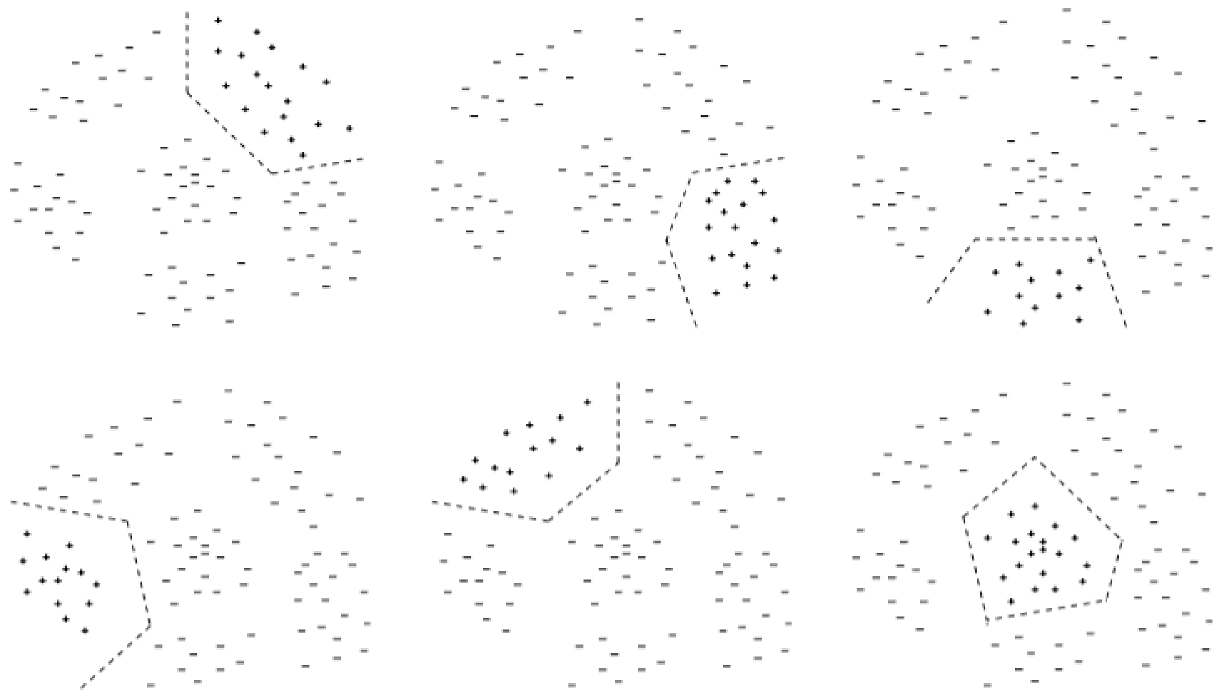
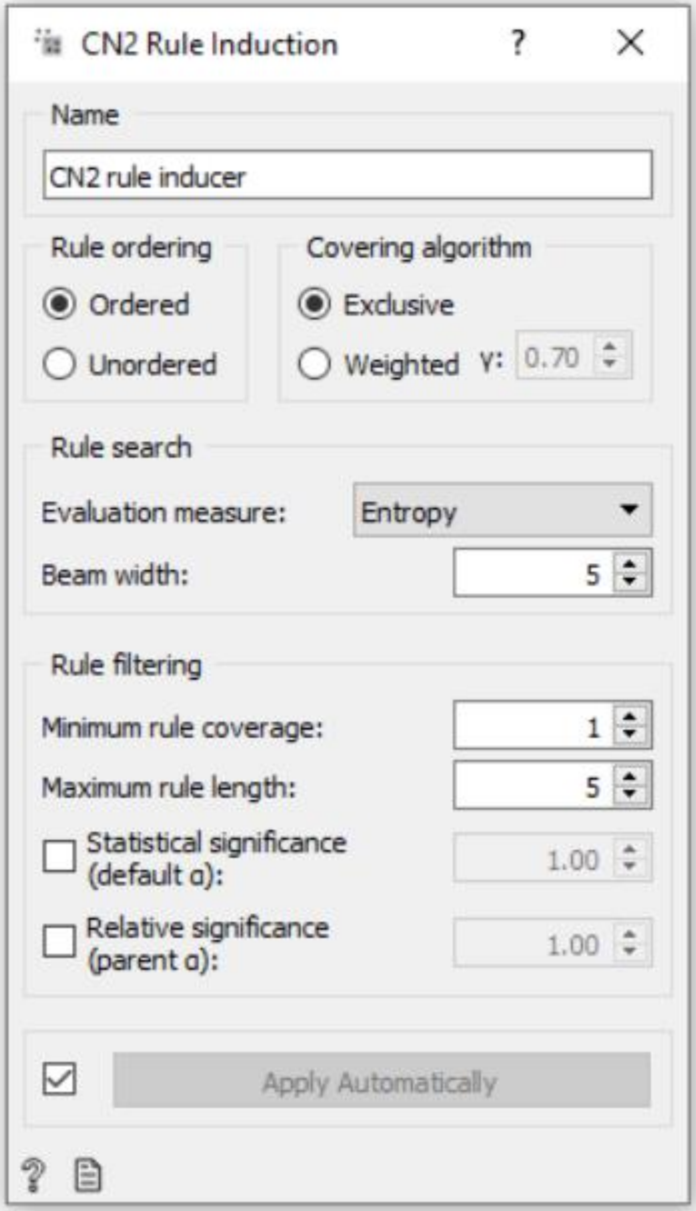


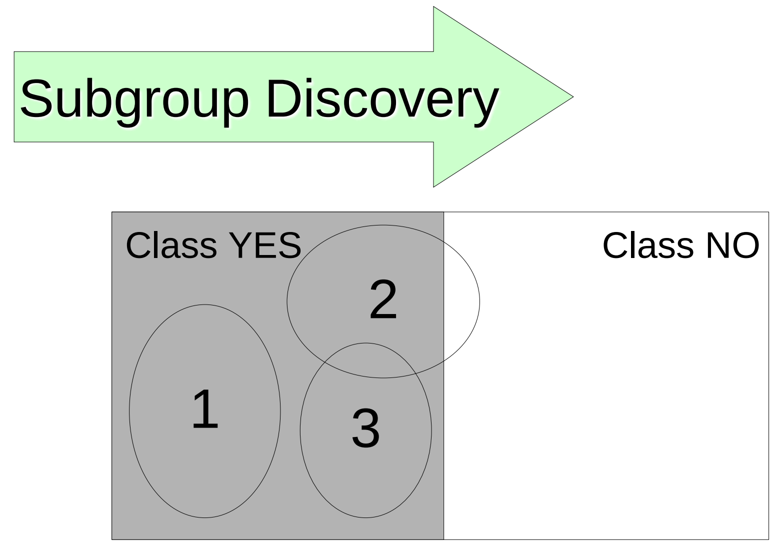
Fig. 10.4: The six binary learning problems that are the result of one-against-all class binarization of the multiclass dataset of Figure 10.2.

CN2 rule learner in Orange



Subgroup Discovery

Person	Age	Spect. presc.	Astigm.	Tear prod.	Lenses
O1	17	myope	no	reduced	NO
O2	23	myope	no	normal	YES
O3	22	myope	yes	reduced	NO
O4	27	myope	yes	normal	YES
O5	19	hypermetrope	no	reduced	NO
O6-O13	...	...	...	...	...
O14	35	hypermetrope	no	normal	YES
O15	43	hypermetrope	yes	reduced	NO
O16	39	hypermetrope	yes	normal	NO
O17	54	myope	no	reduced	NO
O18	62	myope	no	normal	NO
O19-O23	...	...	...	...	...
O24	56	hypermetrope	yes	normal	NO

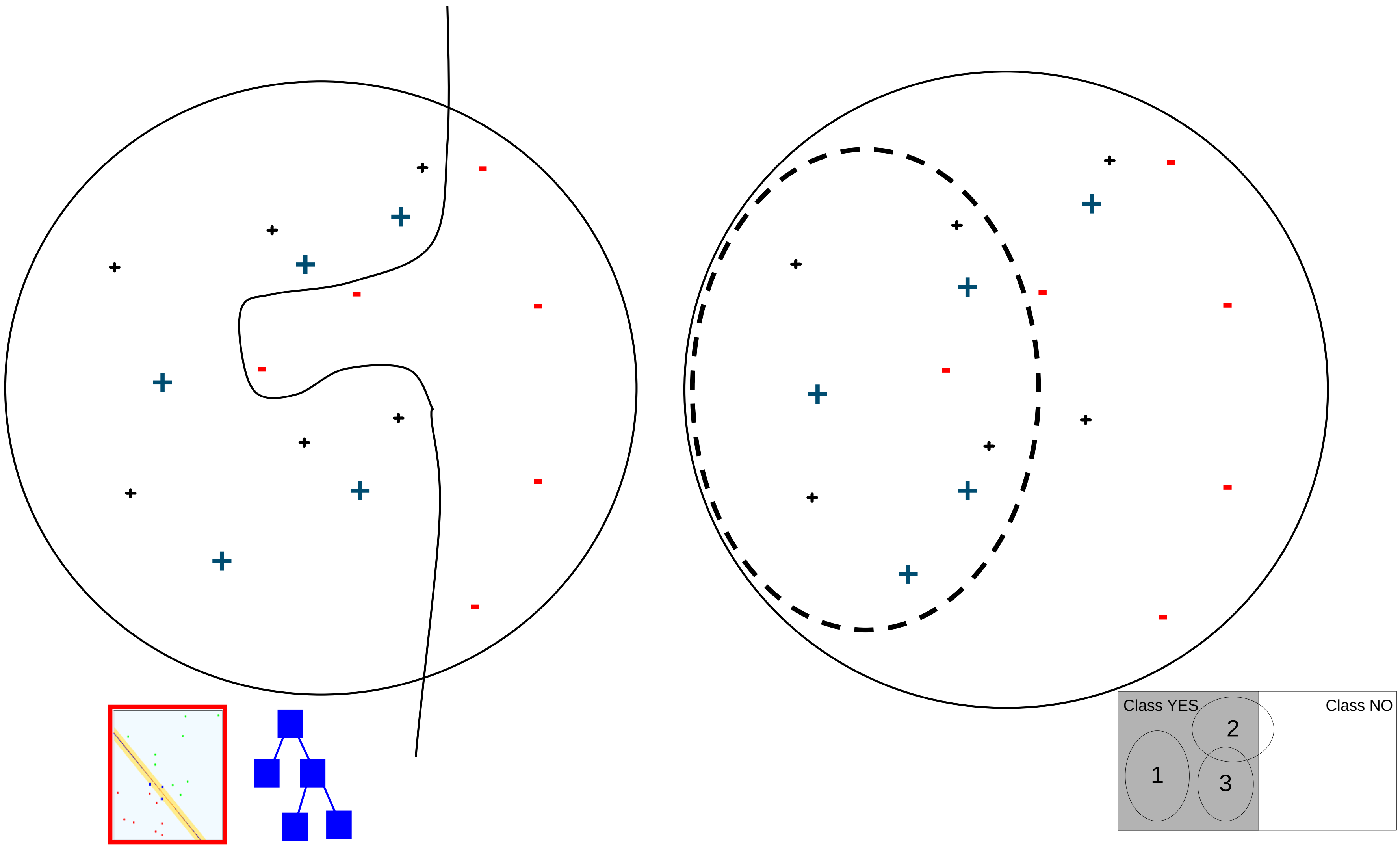


- A task in which individual interpretable patterns in the form of rules are induced from data, labeled by a predefined property of interest.
- SD algorithms learn several independent rules that describe groups of target class examples
  - subgroups must be large and significant

Classification versus Subgroup Discovery

- **Classification (predictive induction) - constructing sets of classification rules**
  - aimed at learning a model for classification or prediction
  - rules are dependent
- **Subgroup discovery (descriptive induction) – constructing individual subgroup describing rules**
  - aimed at finding interesting patterns in target class examples
    - large subgroups (high target class coverage)
    - with significantly different distribution of target class examples (high TP/FP ratio, high significance, high WRAcc
  - each rule (pattern) is an independent chunk of knowledge

Classification versus Subgroup discovery



Subgroup discovery in High CHD Risk Group Detection

Input: Patient records described by anamnestic, laboratory and ECG attributes

Task: Find and characterize population subgroups with high CHD risk (large enough, distributionaly unusual)

From **best induced descriptions**, five were selected by the expert as **most actionable** for CHD risk screening (by GPs):

high-CHD-risk ← male & pos. fam. history & age > 46

high-CHD-risk ← female & bodymassIndex > 25 & age > 63

high-CHD-risk ← ...

high-CHD-risk ← ...

high-CHD-risk ← ...

(Gamberger & Lavrač, JAIR 2002)

Subgroup discovery: Survey data

Education	Marital Status	Sex	Has Children	Approved
primary	single	male	no	no
primary	single	male	yes	no
primary	married	male	no	yes
university	divorced	female	no	yes
university	married	female	yes	yes
secondary	single	male	no	no
university	single	female	no	yes
secondary	divorced	female	no	yes
secondary	single	female	yes	yes
secondary	married	male	yes	yes
primary	married	female	no	yes
secondary	divorced	male	yes	no
university	divorced	female	yes	no
secondary	divorced	male	no	yes

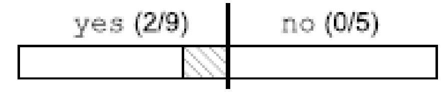
Approved = yes ← Sex = female
Approved = yes ← Marital status = married
Approved = yes ← Marital status = divorced & Has children = no
Approved = yes ← Education = university

Selected rules discovered by Apriori-SD subgroup discovery algorithm.

Subgroup discovery: Survey data

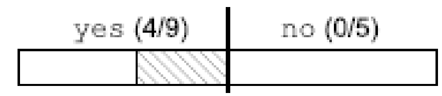
Education	Marital Status	Sex	Has Children	Approved
primary	single	male	no	no
primary	single	male	yes	no
primary	married	male	no	yes
university	divorced	female	no	yes
university	married	female	yes	yes
secondary	single	male	no	no
university	single	female	no	yes
secondary	divorced	female	no	yes
secondary	single	female	yes	yes
secondary	married	male	yes	yes
primary	married	female	no	yes
secondary	divorced	male	yes	no
university	divorced	female	yes	no
secondary	divorced	male	no	yes

```
IF   MaritalStatus = single
 AND Sex = female
THEN Approved = yes
```

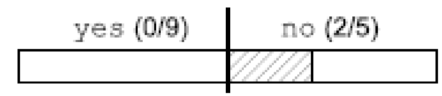



```
IF   MaritalStatus = single
 AND Sex = male
THEN Approved = no
```

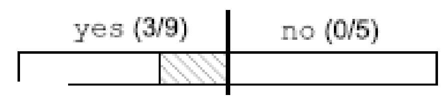

```
IF   MaritalStatus = married
THEN Approved = yes
```



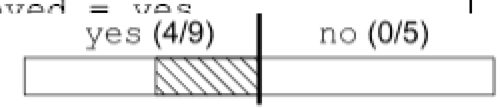

```
IF   MaritalStatus = divorced
 AND HasChildren = yes
THEN Approved = no
```



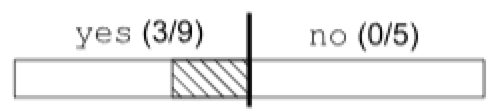

```
IF   MaritalStatus = divorced
 AND HasChildren = no
THEN Approved = yes
```



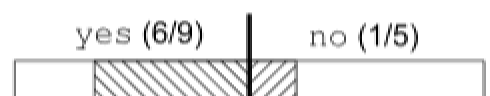

```
IF   MaritalStatus = married
THEN Approved = yes
```



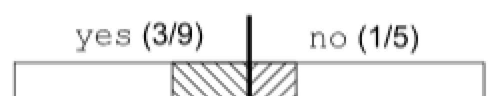

```
IF   MaritalStatus = divorced
 AND HasChildren = no
THEN Approved = yes
```




```
IF   Sex = female
THEN Approved = yes
```




```
IF   Education = university
THEN Approved = yes
```



Classification Rule Learning for Subgroup Discovery: Deficiencies

- Only first few rules induced by the covering algorithm have sufficient support (coverage)
- Subsequent rules are induced from smaller and strongly biased example subsets (pos. examples not covered by previously induced rules), which hinders their ability to detect population subgroups
- ‘Ordered’ rules are induced and interpreted sequentially as a **if-then-else** decision list

CN2-SD: Adapting CN2 Rule Learning to Subgroup Discovery

- Weighted covering algorithm
- Weighted relative accuracy (WRAcc) search heuristics, with added example weights
- Probabilistic classification
- Evaluation with different interestingness measures

CN2-SD: CN2 Adaptations

- General-to-specific search (beam search) for best rules
- Rule quality measure:
  - CN2: Laplace: Acc(Class $\leftarrow$ Cond) =

    = p(Class|Cond) = $(n_c+1)/(n_{rule}+k)$
  - CN2-SD: Weighted Relative Accuracy

    WRAcc(Class $\leftarrow$ Cond) =

    p(Cond) (p(Class|Cond) - p(Class))
- Weighted covering approach (example weights)
- Significance testing (likelihood ratio statistics)
- Output: Unordered rule sets (probabilistic classification)

CN2-SD: Weighted Covering

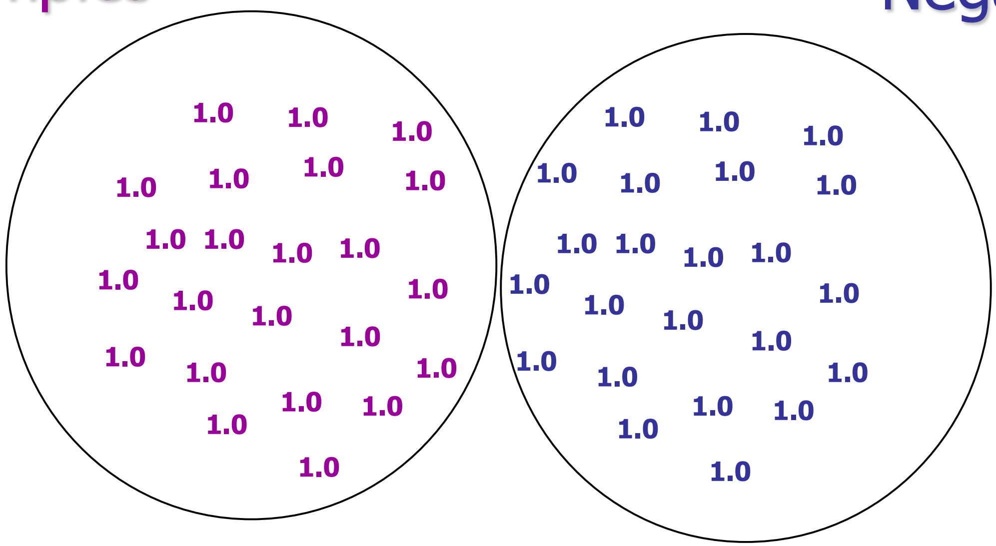
- Standard covering approach:
  covered examples are deleted from current training set

- Weighted covering approach:
  - weights assigned to examples
  - covered pos. examples are re-weighted: in all covering loop iterations, store count i how many times (with how many rules induced so far) a pos. example has been covered: w(e,i), w(e,0)=1
    - **Additive weights: `w(e,i) = 1/(i+1)`**
      `w(e,i)` – pos. example e being covered `i` times

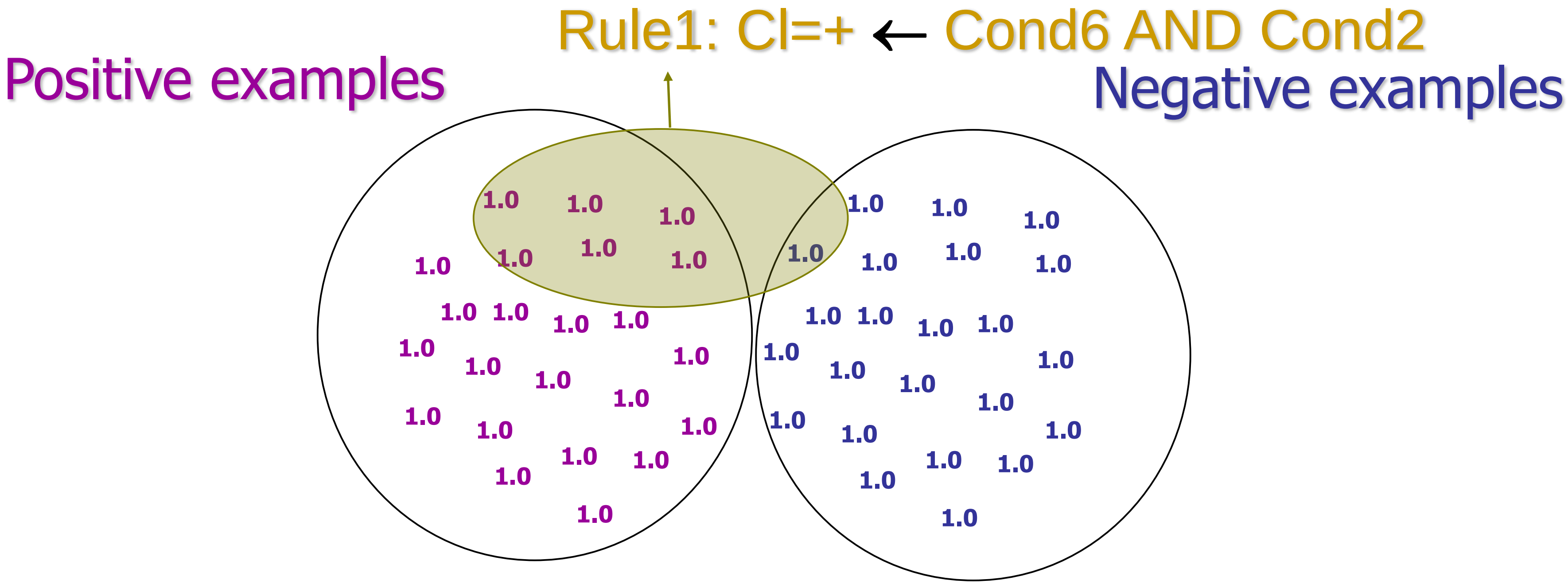
Subgroup Discovery

Positive examples

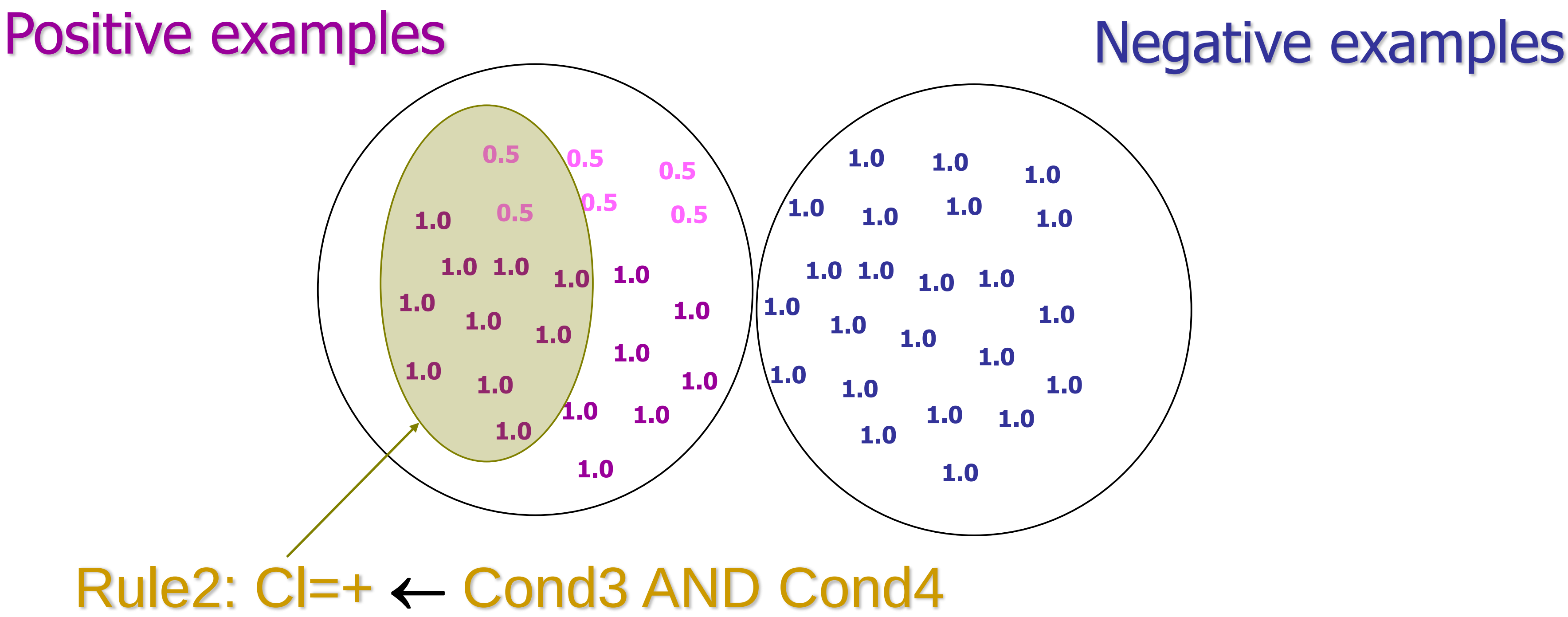
Negative examples



Subgroup Discovery



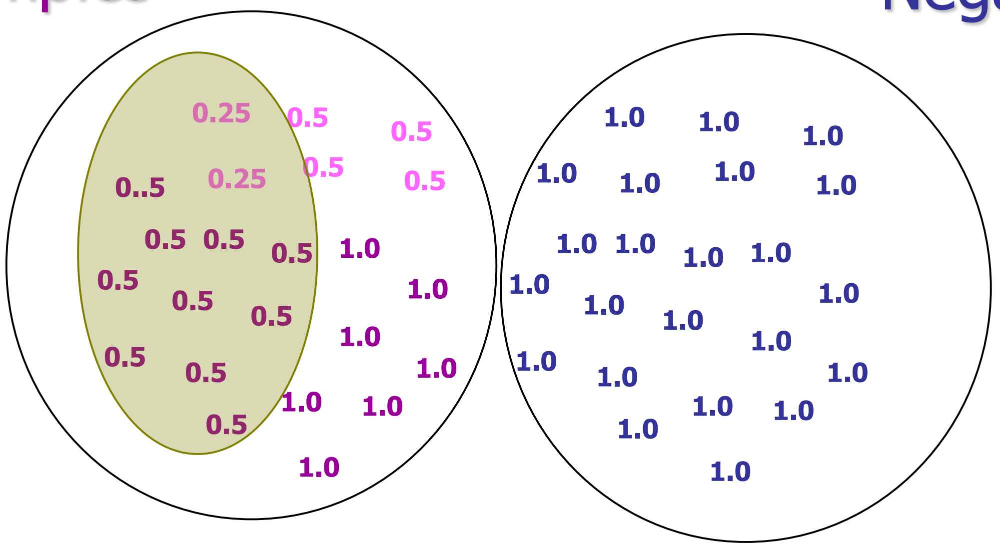
Subgroup Discovery



Subgroup Discovery

Positive examples

Negative examples



CN2-SD: Weighted WRAcc Search Heuristic

- **Weighted relative accuracy (WRAcc) search heuristics, with added example weights**

  WRAcc(Cl ← Cond) = p(Cond) (p(Cl|Cond) - p(Cl))

  increased coverage, decreased # of rules, approx. equal accuracy (PKDD-2000)

- In WRAcc computation, probabilities are estimated with relative frequencies, adapt:

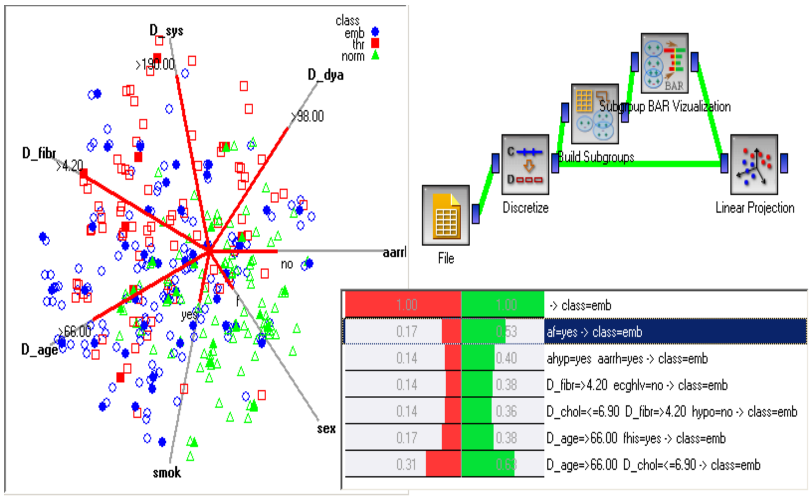
  WRAcc(Cl ← Cond) = p(Cond) (p(Cl|Cond) - p(Cl)) =

  n'(Cond)/N' (n'(Cl.Cond)/n'(Cond) - n'(Cl)/N')

  - N' : sum of weights of examples
  - n'(Cond) : sum of weights of all covered examples
  - n'(Cl.Cond) : sum of weights of all correctly covered examples

SD algorithms in the Orange DM Platform

- **Orange** data mining toolkit
  - classification and subgroup discovery algorithms
  - data mining workflows
  - visualization



- **SD Algorithms in Orange**
  - SD (Gamberger & Lavrač, JAIR 2002)
  - Apriori-SD (Kavšek & Lavrač, AAI 2006)
  - CN2-SD (Lavrač et al., JMLR 2004): Adapting CN2 classification rule learner to Subgroup Discovery

Outline

- Introduction to Machine Learning and Data Mining: Techniques overview
- Rule learning
- Relational learning: Propositionalization
- Semantic data mining
- Relational learning: Wordification

Relational Data Mining (Inductive Logic Programming) task

customer

ID	Zip	Sex	So St	In come	Age	Club	Resp
...	...	...	...	...	...	...	...
3478	34677	m	si	60-70	32	me	nr
3479	43666	f	ma	80-90	45	nm	re
...	...	...	...	...	...	...	...

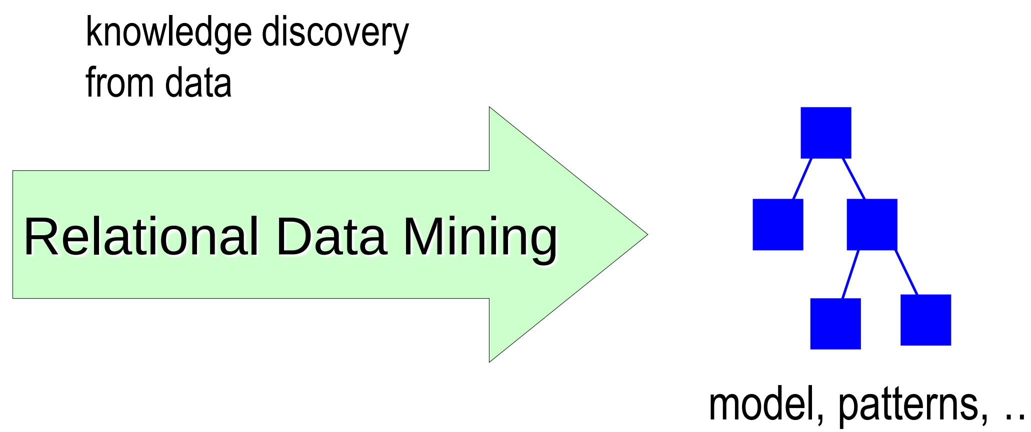
order

Customer ID	Order ID	Store ID	Delivery Mode	Paymt Mode
...	...	...	...	...
3478	2140267	12	regular	cash
3478	3446778	12	express	check
3478	4728386	17	regular	check
3479	3233444	17	express	credit
3479	3475886	12	regular	credit
...	...	...	...	...

store

Store ID	Size	Type	Location
...	...	...	...
12	small	franchise	city
17	large	indep	rural
...	...	...	...

Relational representation of customers, orders and stores.



Given: a relational database, a set of tables. sets of logical facts, a graph, …

Find: a classification model, a set of interesting patterns

Relational data mining

- **ILP, relational learning, relational data mining**
  - Learning from complex multi-relational data

customer							
ID	Zip	Sex	SoSt	Income	Age	Club	Resp
...	...	...	...	...	...	...	...
3478	34677	m	si	60-70	32	me	nr
3479	43666	f	ma	80-90	45	nm	re
...	...	...	...	...	...	...	...

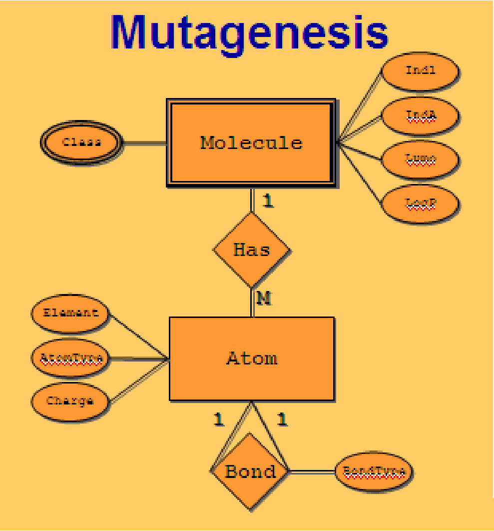
order				
Customer ID	Order ID	Store ID	Delivery Mode	Paymt Mode
...	...	...	...	...
3478	2140267	12	regular	cash
3478	3446778	12	express	check
3478	4728386	17	regular	check
3479	3233444	17	express	credit
3479	3475886	12	regular	credit
...	...	...	...	...

store			
Store ID	Size	Type	Location
...	...	...	...
12	small	franchise	city
17	large	indep	rural
...	...	...	...

Relational representation of customers, orders and stores.

Relational data mining

- **ILP, relational learning, relational data mining**
  - Learning from complex multi-relational data
  - Learning from complex structured data: e.g., molecules and their biochemical properties



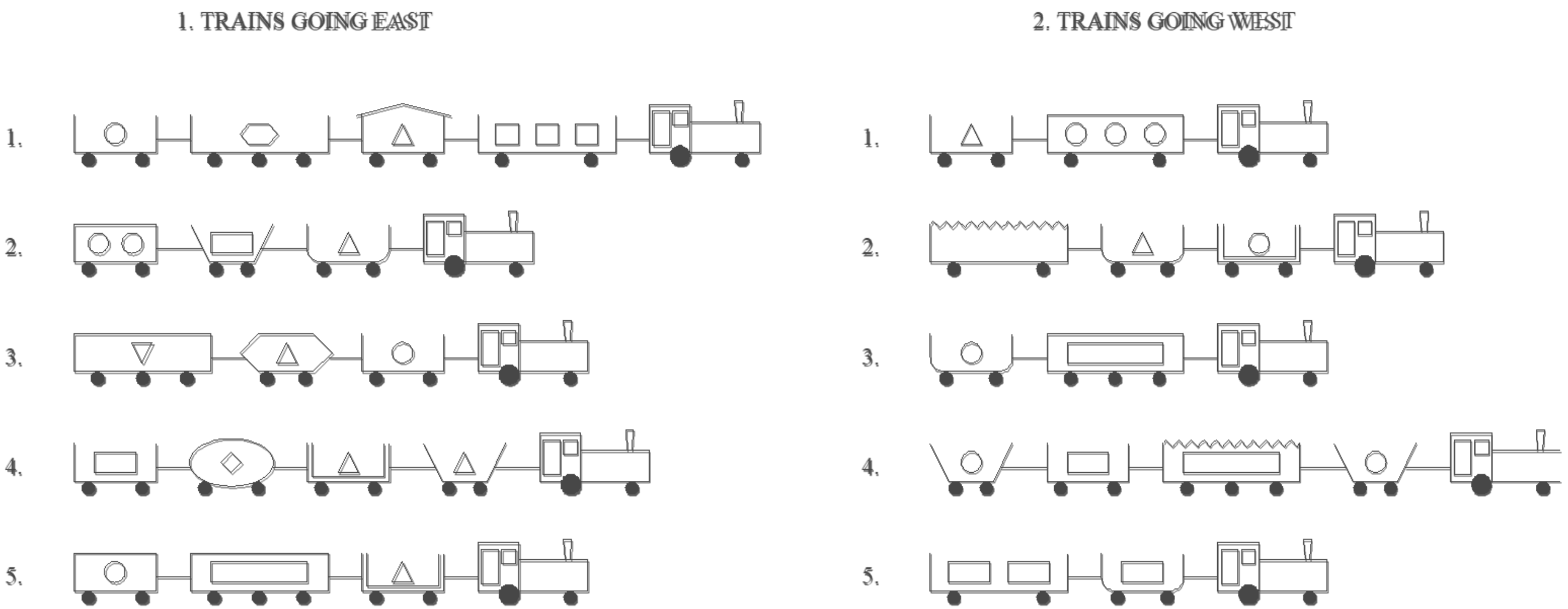
customer							
ID	Zip	Sex	SoSt	Income	Age	Club	Resp
...	...	...	...	...	...	...	...
3478	34677	m	si	60-70	32	me	nr
3479	43666	f	ma	80-90	45	nm	re
...	...	...	...	...	...	...	...

order				
Customer ID	Order ID	Store ID	Delivery Mode	Paymt Mode
...	...	...	...	...
3478	2140267	12	regular	cash
3478	3446778	12	express	check
3478	4728386	17	regular	check
3479	3233444	17	express	credit
3479	3475886	12	regular	credit
...	...	...	...	...

store			
Store ID	Size	Type	Location
...	...	...	...
12	small	franchise	city
17	large	indep	rural
...	...	...	...

Relational representation of customers, orders and stores.

Sample problem: East-West trains



RDM knowledge representation (database)

LOAD_TABLE

LOAD	CAR	OBJECT	NUMBER
l1	c1	circle	1
l2	c2	hexagon	1
l3	c3	triangle	1
l4	c4	rectangle	3
…	…	…	

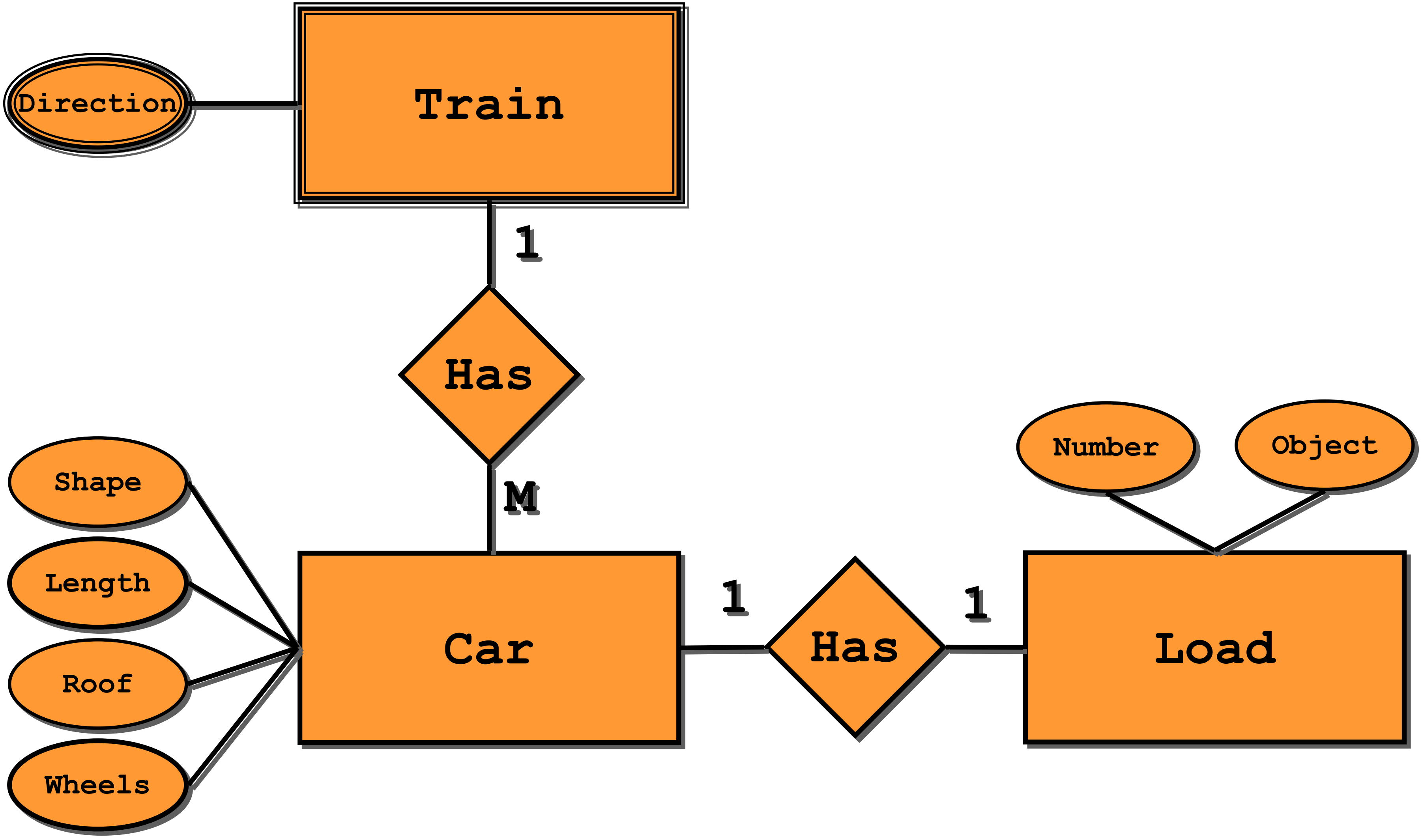
TRAIN_TABLE

TRAIN	EASTBOUND
t1	TRUE
t2	TRUE
…	…
t6	FALSE
…	…

CAR_TABLE

CAR	TRAIN	SHAPE	LENGTH	ROOF	WHEELS
c1	t1	rectangle	short	none	2
c2	t1	rectangle	long	none	3
c3	t1	rectangle	short	peaked	2
c4	t1	rectangle	long	none	2
…	…	…			…

ER diagram for East-West trains



Relational data mining

- Relational data mining is characterized by using background knowledge (domain knowledge) in the data mining process
- Selected approaches:
  - Inductive logic programming - ILP (Muggleton, 1991; Lavrač & Džeroski 1994), …
  - Relational learning (Quinlan,1993)
  - Learning in DL (Lisi 2004), …
  - Relational Data Mining (Džeroski & Lavrač, 2001),
  - Statistical relational learning (Domingos, De Raedt…)
  - Propositionalization approach to RDM (Lavrač et al.)

Our early work:
Semantic subgroup discovery

- Propositionalization approach: Using relational subgroup discovery in the SDM context
  - General purpose system **RSD** for **Relational Subgroup Discovery**, using a propositionalization approach to relational data mining
  - Applied to semantic data mining in a biomedical application by using the Gene Ontology as background knowledge in analyzing microarray data

(Železny and Lavrač, MLJ 2006)

Relational Data Mining through Propositionalization

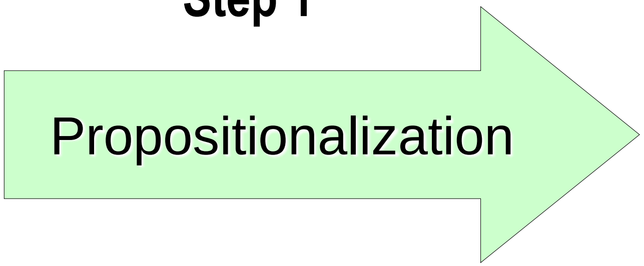
customer							
ID	Zip	Sex	SoSt	Income	Age	Club	Resp
...	...	...	...	...	...	...	...
3478	34677	m	si	60-70	32	me	nr
3479	43666	f	ma	80-90	45	nm	re
...	...	...	...	...	...	...	...

order				
Customer ID	Order ID	Store ID	Delivery Mode	Paymt Mode
...	...	...	...	...
3478	2140267	12	regular	cash
3478	3446778	12	express	check
3478	4728386	17	regular	check
3479	3233444	17	express	credit
3479	3475886	12	regular	credit
...	...	...	...	...

store			
Store ID	Size	Type	Location
...	...	...	...
12	small	franchise	city
17	large	indep	rural
...	...	...	...

Relational representation of customers, orders and stores.

Step 1



	f1	f2	f3	f4	f5	f6	...				...	fn
g1	1	0	0	1	1	1	0	0	1	0	1	1
g2	0	1	1	0	1	1	0	0	0	1	1	0
g3	0	1	1	1	0	0	1	1	0	0	0	1
g4	1	1	1	0	1	1	0	0	1	1	1	0
g5	1	1	1	0	0	1	0	1	1	0	1	0
g1	0	0	1	1	0	0	0	1	0	0	0	1
g2	1	1	0	0	1	1	0	1	0	1	1	1
g3	0	0	0	0	1	0	0	1	1	1	0	0
g4	1	0	1	1	1	0	1	0	0	1	0	1

Relational Data Mining through Propositionalization

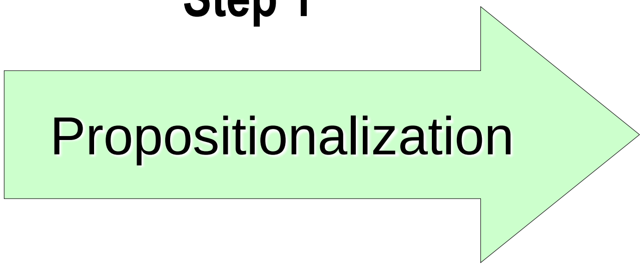
customer							
ID	Zip	Sex	So St	In come	Age	Club	Resp
...	...	...	...	...	...	...	...
3478	34677	m	si	60-70	32	me	nr
3479	43666	f	ma	80-90	45	nm	re
...	...	...	...	...	...	...	...

order				
Customer ID	Order ID	Store ID	Delivery Mode	Paymt Mode
...	...	...	...	...
3478	2140267	12	regular	cash
3478	3446778	12	express	check
3478	4728386	17	regular	check
3479	3233444	17	express	credit
3479	3475886	12	regular	credit
...	...	...	...	...

store			
Store ID	Size	Type	Location
...	...	...	...
12	small	franchise	city
17	large	indep	rural
...	...	...	...

Relational representation of customers, orders and stores.

Step 1



1. constructing relational features
2. constructing a propositional table

	f1	f2	f3	f4	f5	f6	...				...	fn
g1	1	0	0	1	1	1	0	0	1	0	1	1
g2	0	1	1	0	1	1	0	0	0	1	1	0
g3	0	1	1	1	0	0	1	1	0	0	0	1
g4	1	1	1	0	1	1	0	0	1	1	1	0
g5	1	1	1	0	0	1	0	1	1	0	1	0
g1	0	0	1	1	0	0	0	1	0	0	0	1
g2	1	1	0	0	1	1	0	1	0	1	1	1
g3	0	0	0	0	1	0	0	1	1	1	0	0
g4	1	0	1	1	1	0	1	0	0	1	0	1

Relational Data Mining through Propositionalization

customer

ID	Zip	Sex	So St	In come	Age	Cl ub	Re sp
...	...	...	...	...	...	...	...
3478	34677	m	si	60-70	32	me	nr
3479	43666	f	ma	80-90	45	nm	re
...	...	...	...	...	...	...	...

order

Customer ID	Order ID	Store ID	Delivery Mode	Paymt Mode
...	...	...	...	...
3478	2140267	12	regular	cash
3478	3446778	12	express	check
3478	4728386	17	regular	check
3479	3233444	17	express	credit
3479	3475886	12	regular	credit
...	...	...	...	...

store

Store ID	Size	Type	Location
...	...	...	...
12	small	franchise	city
17	large	indep	rural
...	...	...	...

Relational representation of customers, orders and stores.

Step 1

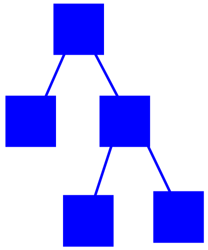
Propositionalization

	f1	f2	f3	f4	f5	f6	...				...	fn
g1	1	0	0	1	1	1	0	0	1	0	1	1
g2	0	1	1	0	1	1	0	0	0	1	1	0
g3	0	1	1	1	0	0	1	1	0	0	0	1
g4	1	1	1	0	1	1	0	0	1	1	1	0
g5	1	1	1	0	0	1	0	1	1	0	1	0
g1	0	0	1	1	0	0	0	1	0	0	0	1
g2	1	1	0	0	1	1	0	1	0	1	1	1
g3	0	0	0	0	1	0	0	1	1	1	0	0
g4	1	0	1	1	1	0	1	0	0	1	0	1

	f1	f2	f3	f4	f5	f6	...				...	fn
g1	1	0	0	1	1	1	0	0	1	0	1	1
g2	0	1	1	0	1	1	0	0	0	1	1	0
g3	0	1	1	1	0	0	1	1	0	0	0	1
g4	1	1	1	0	1	1	0	0	1	1	1	0
g5	1	1	1	0	0	1	0	1	1	0	1	0
g1	0	0	1	1	0	0	0	1	0	0	0	1
g2	1	1	0	0	1	1	0	1	0	1	1	1
g3	0	0	0	0	1	0	0	1	1	1	0	0
g4	1	0	1	1	1	0	1	0	0	1	0	1

Step 2

Data Mining



model, patterns, …

Relational Data Mining through Propositionalization

customer							
ID	Zip	Sex	SoSt	Income	Age	Club	Resp
...	...	...	...	...	...	...	...
3478	34677	m	si	60-70	32	me	nr
3479	43666	f	ma	80-90	45	nm	re
...	...	...	...	...	...	...	...

order				
Customer ID	Order ID	Store ID	Delivery Mode	Paymt Mode
...	...	...	...	...
3478	2140267	12	regular	cash
3478	3446778	12	express	check
3478	4728386	17	regular	check
3479	3233444	17	express	credit
3479	3475886	12	regular	credit
...	...	...	...	...

store			
Store ID	Size	Type	Location
...	...	...	...
12	small	franchise	city
17	large	indep	rural
...	...	...	...

Relational representation of customers, orders and stores.

Step 1

Propositionalization

	f1	f2	f3	f4	f5	f6	...				...	fn
g1	1	0	0	1	1	1	0	0	1	0	1	1
g2	0	1	1	0	1	1	0	0	0	1	1	0
g3	0	1	1	1	0	0	1	1	0	0	0	1
g4	1	1	1	0	1	1	0	0	1	1	1	0
g5	1	1	1	0	0	1	0	1	1	0	1	0
g1	0	0	1	1	0	0	0	1	0	0	0	1
g2	1	1	0	0	1	1	0	1	0	1	1	1
g3	0	0	0	0	1	0	0	1	1	1	0	0
g4	1	0	1	1	1	0	1	0	0	1	0	1

	f1	f2	f3	f4	f5	f6	...				...	fn
g1	1	0	0	1	1	1	0	0	1	0	1	1
g2	0	1	1	0	1	1	0	0	0	1	1	0
g3	0	1	1	1	0	0	1	1	0	0	0	1
g4	1	1	1	0	1	1	0	0	1	1	1	0
g5	1	1	1	0	0	1	0	1	1	0	1	0
g1	0	0	1	1	0	0	0	1	0	0	0	1
g2	1	1	0	0	1	1	0	1	0	1	1	1
g3	0	0	0	0	1	0	0	1	1	1	0	0
g4	1	0	1	1	1	0	1	0	0	1	0	1

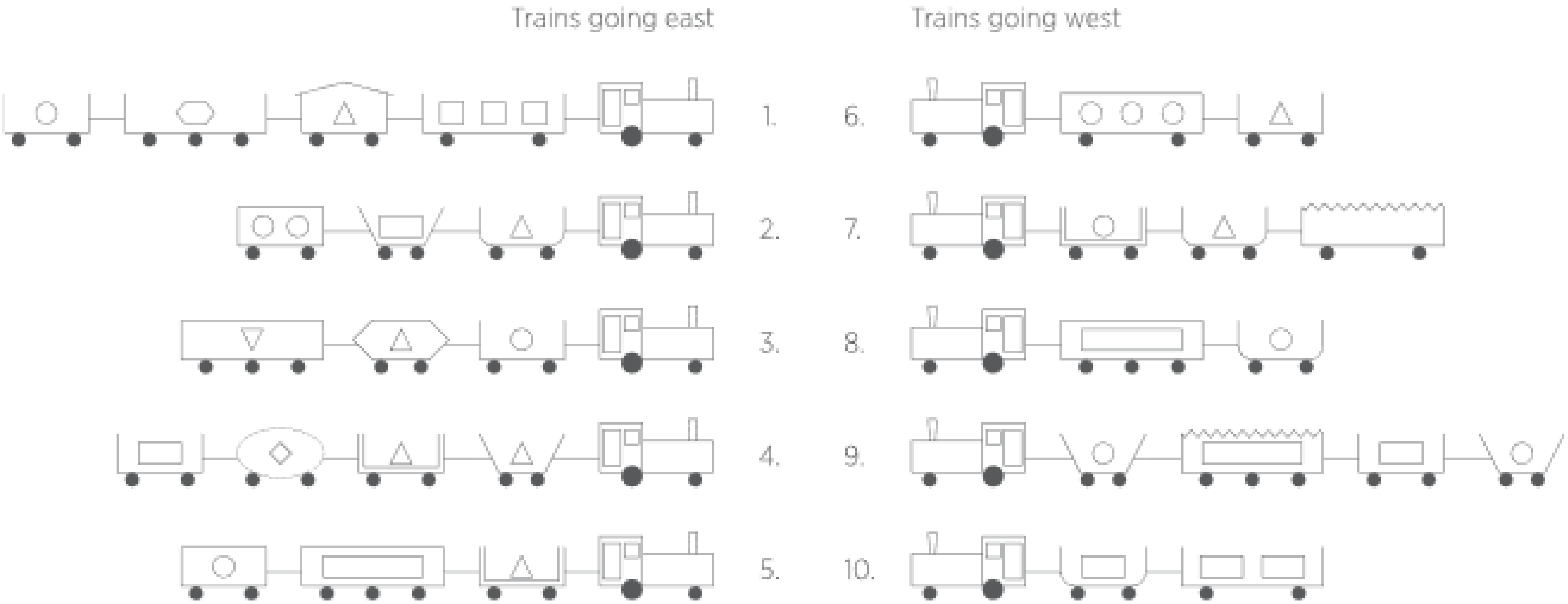
Step 2

Data Mining

```
target(A) :-
   'Doctor'(A), 'Italy'(A).

target(A) :-
   'Public'(A), 'Gold'(A).

target(A) :-
   'Poland'(A), 'Deposit'(A), 'Gold'(A).

target(A) :-
   'Germany'(A), 'Insurance'(A).

target(A) :-
   'Service'(A), 'Germany'(A).
```

patterns (set of rules)

Sample ILP problem: East-West trains



Relational data representation

LOAD	CAR	OBJECT	NUMBER
l1	c1	circle	1
l2	c2	hexagon	1
l3	c3	triangle	1
l4	c4	rectangle	3
...	...	...	

TRAIN_TABLE

TRAIN	EASTBOUND
t1	TRUE
t2	TRUE
...	...
t6	FALSE
...	...

CAR	TRAIN	SHAPE	LENGTH	ROOF	WHEELS
c1	t1	rectangle	short	none	2
c2	t1	rectangle	long	none	3
c3	t1	rectangle	short	peaked	2
c4	t1	rectangle	long	none	2
...	...	...			...

Propositionalization in a nutshell

Propositionalization task

Transform a multi-relational (**multiple-table**) representation to a propositional representation (**single table**)

Proposed in ILP systems LINUS (Lavrac et al. 1991, 1994), 1BC (Flach and Lachiche 1999), …

LOAD	CAR	OBJECT	NUMBER
l1	c1	circle	1
l2	c2	hexagon	1
l3	c3	triangle	1
l4	c4	rectangle	3
...	...	...	

TRAIN_TABLE

TRAIN	EASTBOUND
t1	TRUE
t2	TRUE
...	...
t6	FALSE
...	...

CAR	TRAIN	SHAPE	LENGTH	ROOF	WHEELS
c1	t1	rectangle	short	none	2
c2	t1	rectangle	long	none	3
c3	t1	rectangle	short	peaked	2
c4	t1	rectangle	long	none	2
...	...	...			...

Propositionalization in a nutshell

Main propositionalization step:

first-order feature construction

f1(T):-hasCar(T,C),clength(C,short).

f2(T):-hasCar(T,C), hasLoad(C,L),
loadShape(L,circle)

f3(T) :- ....

LOAD	CAR	OBJECT	NUMBER
l1	c1	circle	1
l2	c2	hexagon	1
l3	c3	triangle	1
l4	c4	rectangle	3
...	...	...	

TRAIN_TABLE

TRAIN	EASTBOUND
t1	TRUE
t2	TRUE
…	…
t6	FALSE
…	…

CAR	TRAIN	SHAPE	LENGTH	ROOF	WHEELS
c1	t1	rectangle	short	none	2
c2	t1	rectangle	long	none	3
c3	t1	rectangle	short	peaked	2
c4	t1	rectangle	long	none	2
…	…	…			…

Propositional learning:

t(T) ← f1(T), f4(T)

PROPOSITIONAL TRAIN_TABLE

train(T)	f1(T)	f2(T)	f3(T)	f4(T)	f5(T)
t1	t	t	f	t	t
t2	t	t	t	t	t
t3	f	f	t	f	f
t4	t	f	t	f	f
…	…	…			…

Relational interpretation:

eastbound(T) ←
hasShortCar(T),hasClosedCar(T).

RSD algorithm: Relational Data Mining in Orange4WS

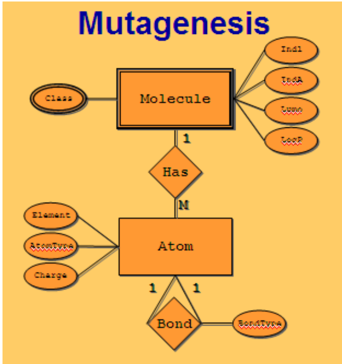
- service for propositionalization through efficient first-order feature construction (Železny and Lavrač, MLJ 2006)

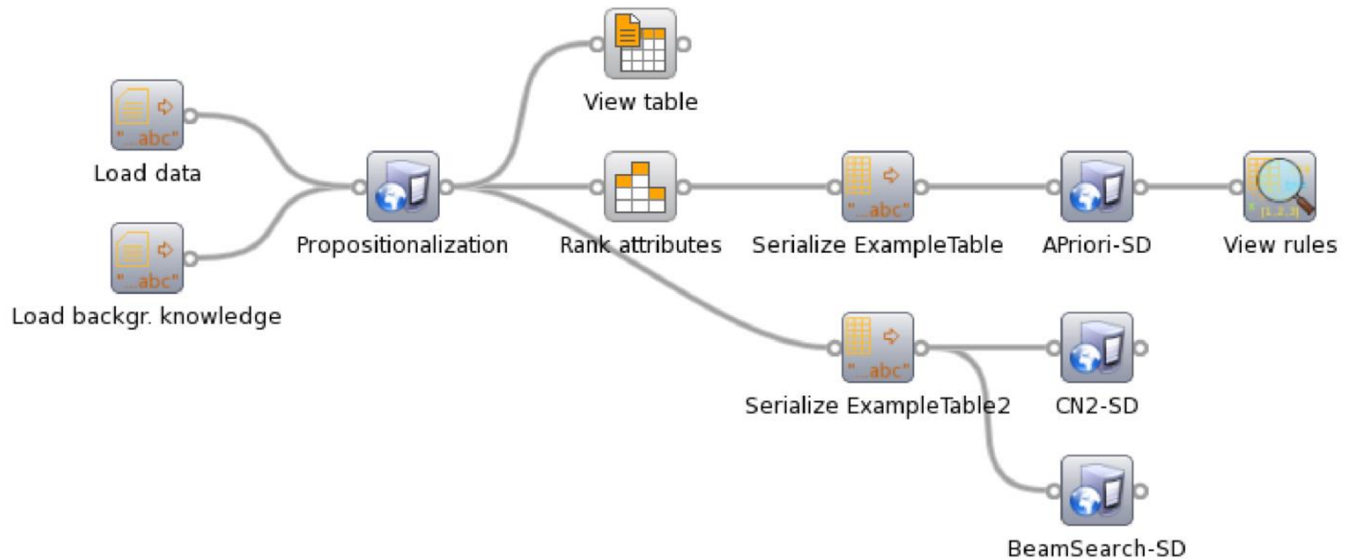
  f121(M):- hasAtom(M,A), atomType(A,21)

  f235(M):- lumo(M,Lu), lessThr(Lu,1.21)

- subgroup discovery using CN2-SD

  mutagenic(M) ← feature121(M), feature235(M)





RSD algorithm

Efficient propositionalization can be applied to individual-centered, multi-instance learning problems:

- one free global variable (denoting an individual, e.g. molecule M)
- one or more structural predicates: (e.g. has_atom(M,A)), each introducing a new existential local variable (e.g. atom A), using either the global variable (M) or a local variable introduced by other structural predicates (A)
- one or more utility predicates defining properties of individuals or their parts, assigning values to variables

  feature121(M):- hasAtom(M,A), atomType(A,21)

  feature235(M):- lumo(M,Lu), lessThr(Lu,-1.21)

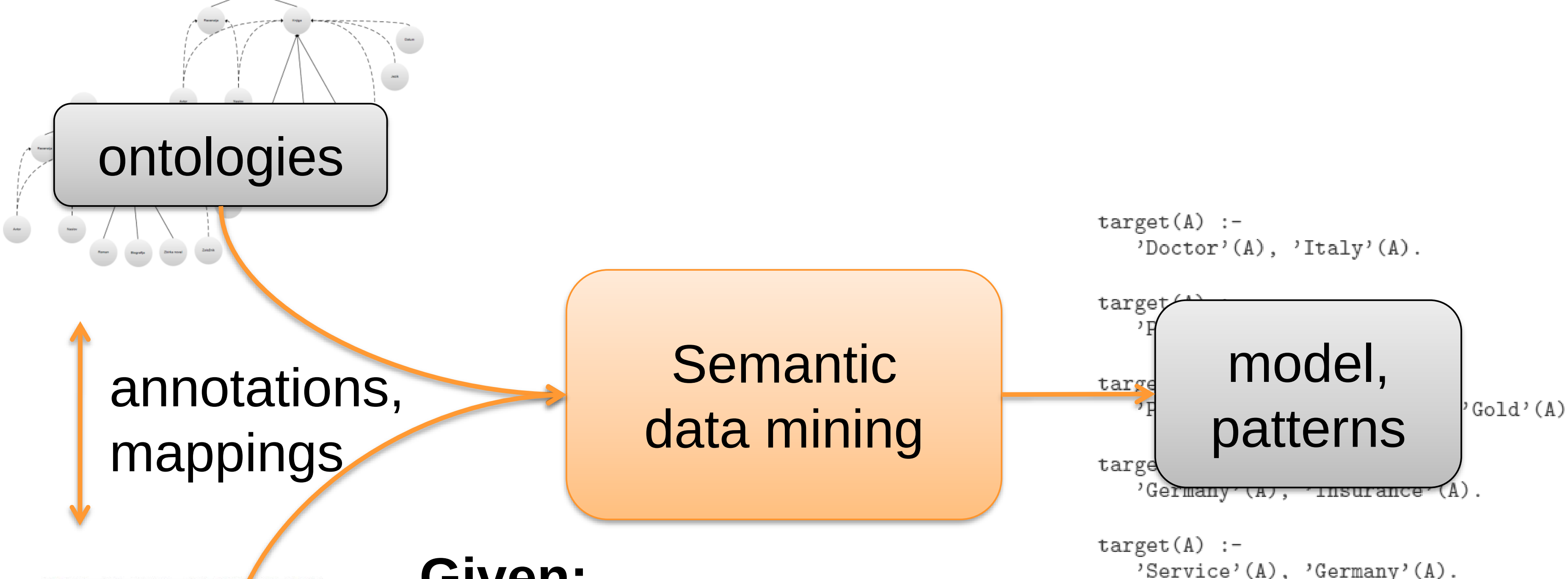
  mutagenic(M):- feature121(M), feature235(M)

Outline

- Introduction to Machine Learning and Data Mining: Techniques overview
- Rule learning
- Relational learning: Propositionalization
- Semantic data mining
- Relational learning: Wordification

What is Semantic Data Mining

SDM task definition



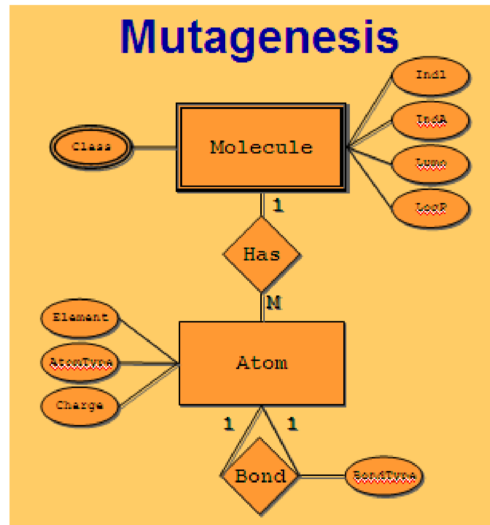
Given:

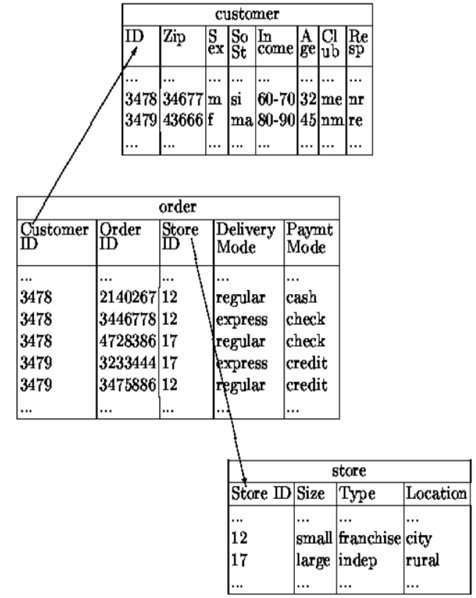
- transaction data table, relational database, text documents, Web pages, …
- one or more domain ontologies

Find: a classification model, a set of patterns

Semantic data mining

- **ILP, relational learning, relational data mining**
  - Learning from complex multi-relational data
  - Learning from complex structured data: e.g., molecules and their biochemical properties
  - Learning by using domain knowledge in the form of ontologies = **semantic data mining**





customer

ID	Zip	Sex	SoSt	Income	Age	Club	Resp
...	...	...	...	...	...	...	...
3478	34677	m	si	60-70	32	me	nr
3479	43666	f	ma	80-90	45	nm	re
...	...	...	...	...	...	...	...

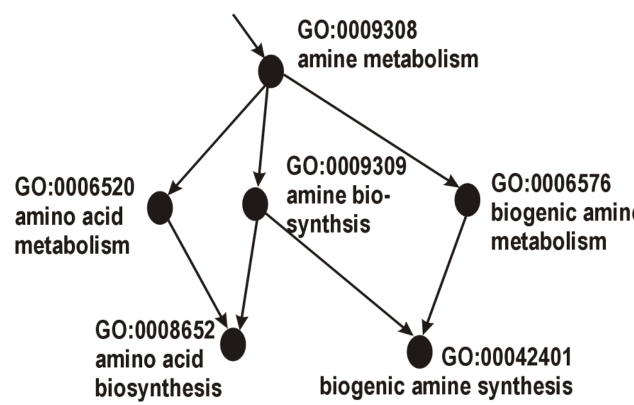
order

Customer ID	Order ID	Store ID	Delivery Mode	Paymt Mode
...	...	...	...	...
3478	2140267	12	regular	cash
3478	3446778	12	express	check
3478	4728386	17	regular	check
3479	3233444	17	express	credit
3479	3475886	12	regular	credit
...	...	...	...	...

store

Store ID	Size	Type	Location
...	...	...	...
12	small	franchise	city
17	large	indep	rural
...	...	...	...

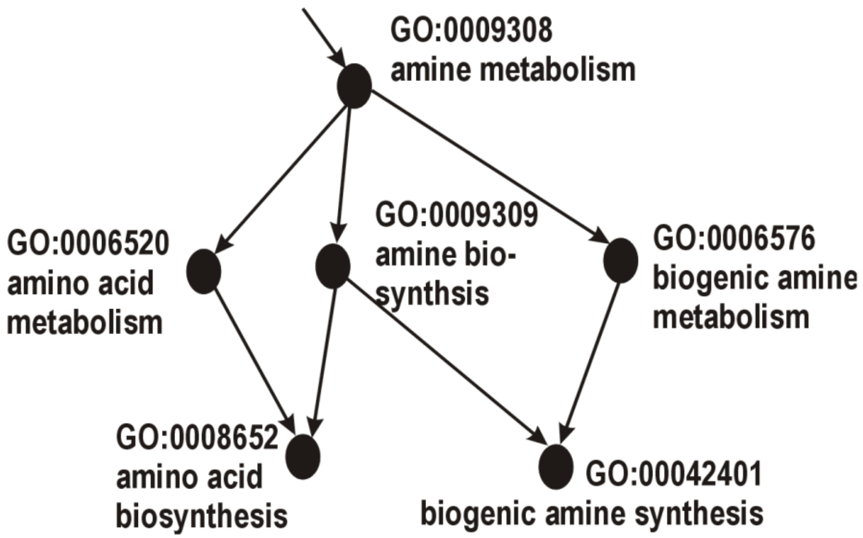
Relational representation of customers, orders and stores.



Using domain ontologies in Semantic Data Mining

Using domain ontologies as background knowledge, e.g., using the Gene Ontology (GO)

- GO is a database of terms, describing gene sets in terms of their
  - functions (12,093)
  - processes (1,812)
  - components (7,459)
- Genes are annotated to GO terms
- Terms are connected (is_a, part_of)
- Levels represent terms generality



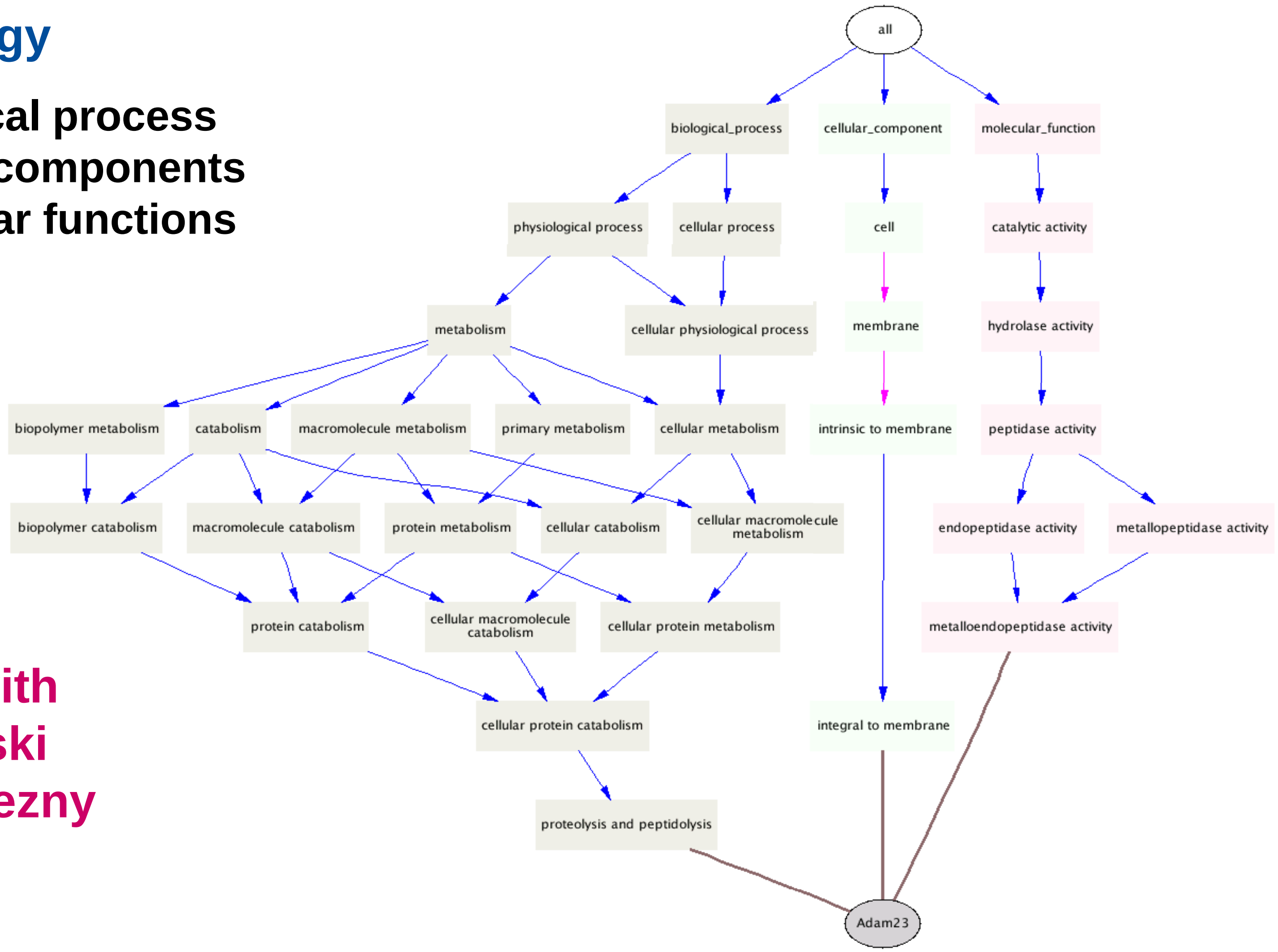
What is Semantic Data Mining

- Ontology-driven (semantic) data mining is an emerging research topic
- Semantic Data Mining (SDM) - a new term denoting:
  - the new challenge of mining semantically annotated resources, with ontologies used as background knowledge to data mining
  - approaches with which semantic data are mined

Using domain ontologies (e.g. Gene Ontology) as background knowledge for Data Mining

Gene Ontology

12093 biological process
1812 cellular components
7459 molecular functions

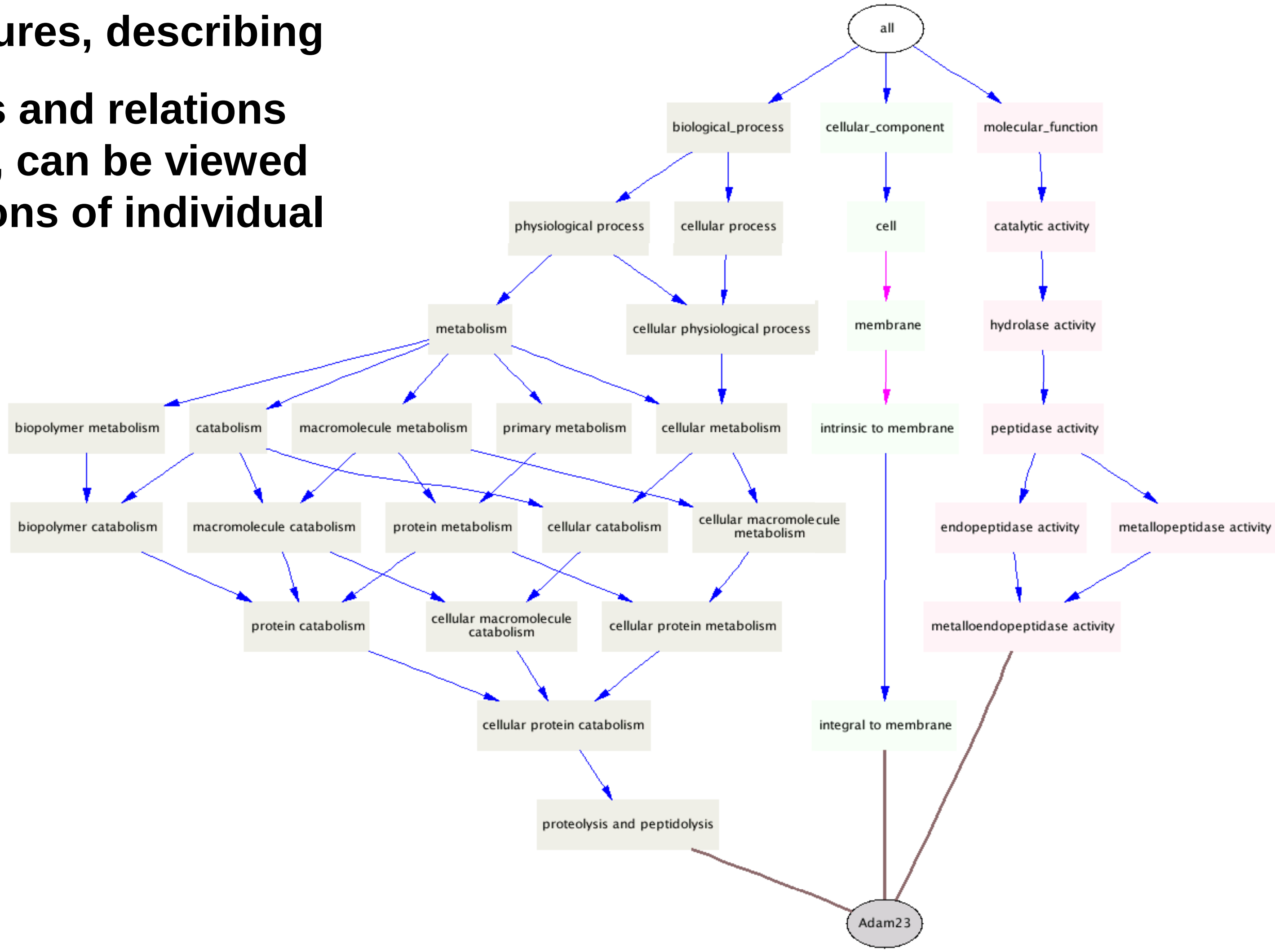


Joint work with Igor Trajkovski and Filip Zelezny

Using domain ontologies (e.g. Gene Ontology) as background knowledge for Data Mining

First-order features, describing

gene properties and relations between genes, can be viewed as generalisations of individual genes



Semantic subgroup discovery with RSD

1. Take ontology terms represented as logical facts in Prolog, e.g.

```
component(gene2532,'GO:0016020').
function(gene2534,'GO:0030554').
process(gene2534,'GO:0007243').
interaction(gene2534,gene4803).
```

2. Automatically generate generalized relational features:

```
f(2,A):-component(A,'GO:0016020').
f(7,A):-function(A,'GO:0030554').
f(11,A):-process(A,'GO:0007243').
f(224,A):- interaction(A,B), function(B,'GO:0016787'),
           component(B,'GO:0043231').
```

3. Propositionalization: Determine truth values of features

4. Learn rules by a subgroup discovery algorithm CN2-SD

Step 2: RSD feature construction

Construction of first order features, with support > *min_support*

f(7,A):-function(A,'GO:0046872').
f(8,A):-function(A,'GO:0004871').
f(11,A):-process(A,'GO:0007165').
f(14,A):-process(A,'GO:0044267').
f(15,A):-process(A,'GO:0050874').
f(20,A):-function(A,'GO:0004871'), process(A,'GO:0050874').
f(26,A):-component(A,'GO:0016021').
f(29,A):- function(A,'GO:0046872'), component(A,'GO:0016020').
f(122,A):-interaction(A,B),function(B,'GO:0004872').
f(223,A):-interaction(A,B),function(B,'GO:0004871'),
process(B,'GO:0009613').
f(224,A):-interaction(A,B),function(B,'GO:0016787'),
component(B,'GO:0043231').

existential

Step 3: RSD Propositionalization

diffexp g1 (gene64499)
diffexp g2 (gene2534)
diffexp g3 (gene5199)
diffexp g4 (gene1052)
diffexp g5 (gene6036)
....

random g1 (gene7443)
random g2 (gene9221)
random g3 (gene2339)
random g4 (gene9657)
random g5 (gene19679)
....

	f1	f2	f3	f4	f5	f6	…				…	fn
g1	1	0	0	1	1	1	0	0	1	0	1	1
g2	0	1	1	0	1	1	0	0	0	1	1	0
g3	0	1	1	1	0	0	1	1	0	0	0	1
g4	1	1	1	0	1	1	0	0	1	1	1	0
g5	1	1	1	0	0	1	0	1	1	0	1	0
g1	0	0	1	1	0	0	0	1	0	0	0	1
g2	1	1	0	0	1	1	0	1	0	1	1	1
g3	0	0	0	0	1	0	0	1	1	1	0	0
g4	1	0	1	1	1	0	1	0	0	1	0	1

Step 4: RSD rule construction with CN2-SD

	f1	f2	f3	f4	f5	f6	…				…	fn
g1	1	0	0	1	1	1	0	0	1	0	1	1
g2	0	1	1	0	1	1	0	0	0	1	1	0
g3	0	1	1	1	0	0	1	1	0	0	0	1
g4	1	1	1	0	1	1	0	0	1	1	1	0
g5	1	1	1	0	0	1	0	1	1	0	1	0
g1	0	0	1	1	0	0	0	1	0	0	0	1
g2	1	1	0	0	1	1	0	1	0	1	1	1
g3	0	0	0	0	1	0	0	1	1	1	0	0
g4	1	0	1	1	1	0	1	0	0	1	0	1

Over-expressed

IF

f2 and f3

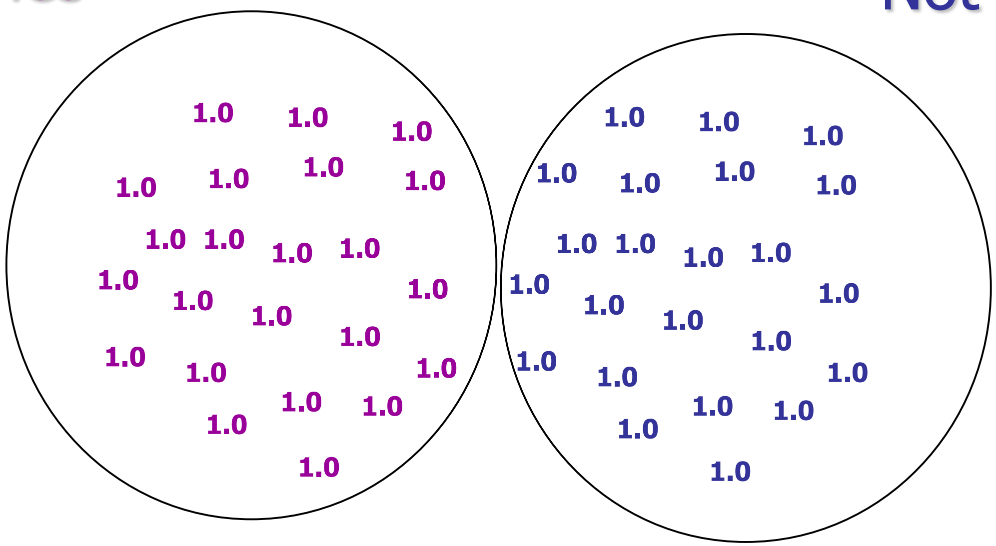
[4,0]

diffexp(A) :- interaction(A,B) & function(B,'GO:0004871')

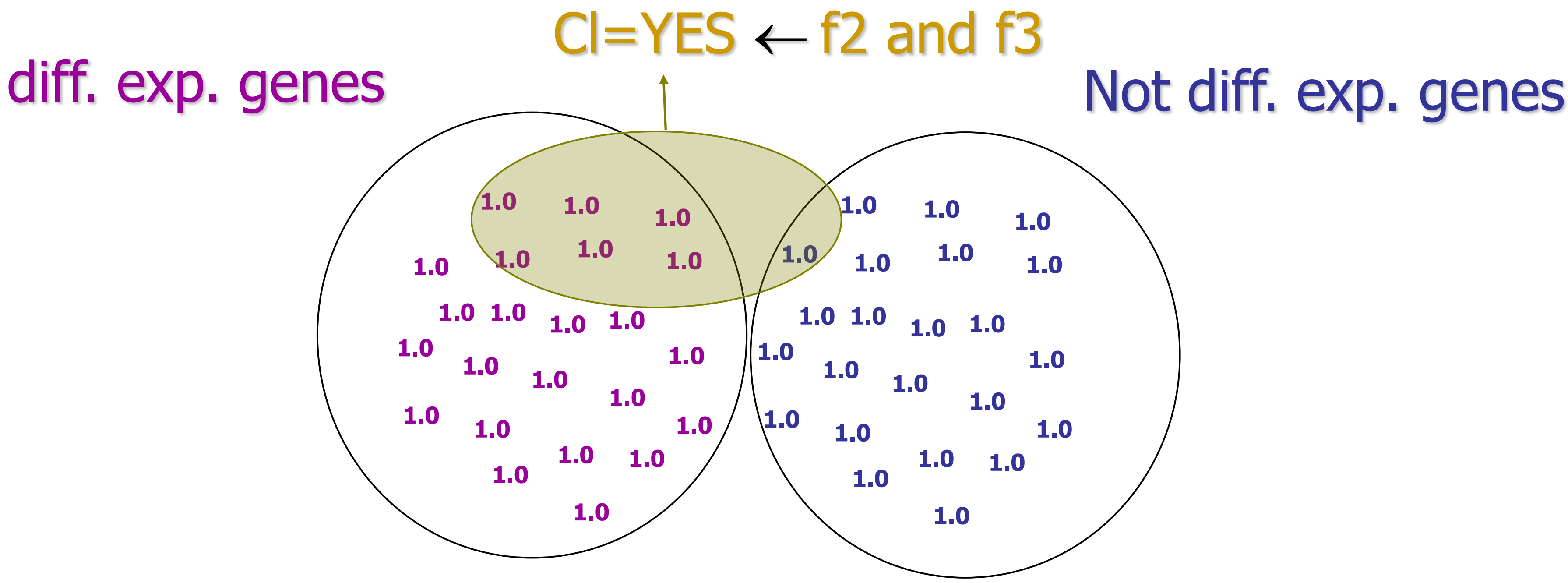
Subgroup Discovery

diff. exp. genes

Not diff. exp. genes



Subgroup Discovery



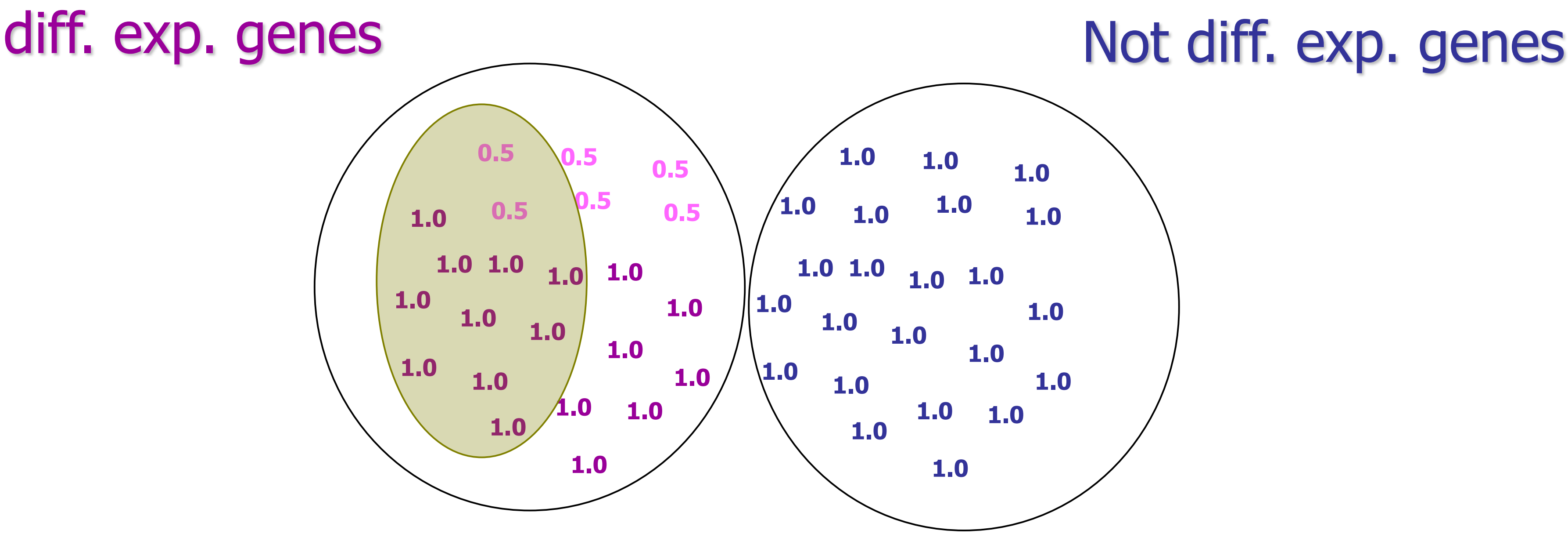
In RSD (using propositional learner CN2-SD):

Quality of the rules = Coverage x Precision

*Coverage = sum of the covered weights

*Precision = purity of the covered genes

Subgroup Discovery



RSD naturally uses gene weights in its procedure for repetitive subgroup generation, via its heuristic rule evaluation: weighted relative accuracy

Outline

- Introduction to Machine Learning and Data Mining: Techniques overview
- Rule learning
- Relational learning: Propositionalization
- Semantic data mining
- Relational learning: Wordification

Propositionaization through Wordification: Motivation

- Develop a RDM technique inspired by **text mining**
- Using a large number of simple, easy to understand features (**words**)
- **I**mproved **scalability**, handling large datasets
- Used as a preprocessing step to propositional learners

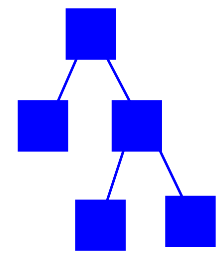
Background: Data mining

Person	Age	Spect. presc.	Astigm.	Tear prod.	Lenses
O1	17	myope	no	reduced	NONE
O2	23	myope	no	normal	SOFT
O3	22	myope	yes	reduced	NONE
O4	27	myope	yes	normal	HARD
O5	19	hypermetrope	no	reduced	NONE
O6-O13	...	...	...	...	...
O14	35	hypermetrope	no	normal	SOFT
O15	43	hypermetrope	yes	reduced	NONE
O16	39	hypermetrope	yes	normal	NONE
O17	54	myope	no	reduced	NONE
O18	62	myope	no	normal	NONE
O19-O23	...	...	...	...	...
O24	56	hypermetrope	yes	normal	NONE

data

knowledge discovery
from data

Data Mining



model, patterns, clusters,

…

Given: transaction data table, a set of text documents, …

Find: a classification model, a set of interesting patterns

Data mining: Task reformulation

Person	Young	Myope	Astigm.	Reuced tea	Lenses
O1	1	1	0	1	NO
O2	1	1	0	0	YES
O3	1	1	1	1	NO
O4	1	1	1	0	YES
O5	1	0	0	1	NO
O6-O13	...	...	...	...	...
O14	0	0	0	0	YES
O15	0	0	1	1	NO
O16	0	0	1	0	NO
O17	0	1	0	1	NO
O18	0	1	0	0	NO
O19-O23	...	...	...	...	...
O24	0	0	1	0	NO

Binary features and class values

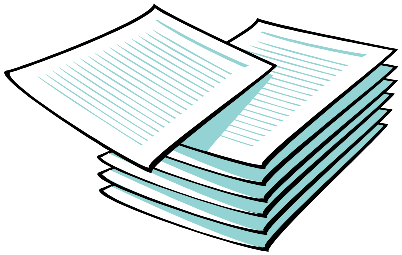
Text mining:
Words/terms as binary features

Document	Word1	Word2	…	WordN	Class
d1	1	1	0	1	NO
d2	1	1	0	0	YES
d3	1	1	1	1	NO
d4	1	1	1	0	YES
d5	1	0	0	1	NO
d6-d13	...	...	...	...	...
d14	0	0	0	0	YES
d15	0	0	1	1	NO
d16	0	0	1	0	NO
d17	0	1	0	1	NO
d18	0	1	0	0	NO
d19-d23	...	...	...	...	...
d24	0	0	1	0	NO

Instances = documents

Words and terms = Binary features

Text mining



Step 1

BoW vector construction

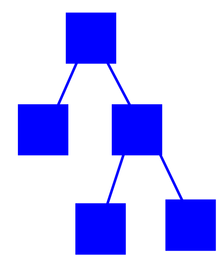
1. BoW features construction
2. Table of BoW vectors construction

Document	Word1	Word2	…	WordN	Class
d1	1	1	0	1	NO
d2	1	1	0	0	YES
d3	1	1	1	1	NO
d4	1	1	1	0	YES
d5	1	0	0	1	NO
d6-d13	…	…	…	…	…
d14	0	0	0	0	YES
d15	0	0	1	1	NO
d16	0	0	1	0	NO
d17	0	1	0	1	NO
d18	0	1	0	0	NO
d19-d23	…	…	…	…	…
d24	0	0	1	0	NO

Document	Word1	Word2	…	WordN	Class
d1	1	1	0	1	NO
d2	1	1	0	0	YES
d3	1	1	1	1	NO
d4	1	1	1	0	YES
d5	1	0	0	1	NO
d6-d13	…	…	…	…	…
d14	0	0	0	0	YES
d15	0	0	1	1	NO
d16	0	0	1	0	NO
d17	0	1	0	1	NO
d18	0	1	0	0	NO
d19-d23	…	…	…	…	…
d24	0	0	1	0	NO

Step 2

Data Mining



model, patterns, clusters, …

Text Mining

- Feature construction
  - StopWords elimination
  - Stemming or lemmatization
  - Term construction by frequent N-Grams construction
  - Terms obtained from thesaurus (e.g., WordNet)

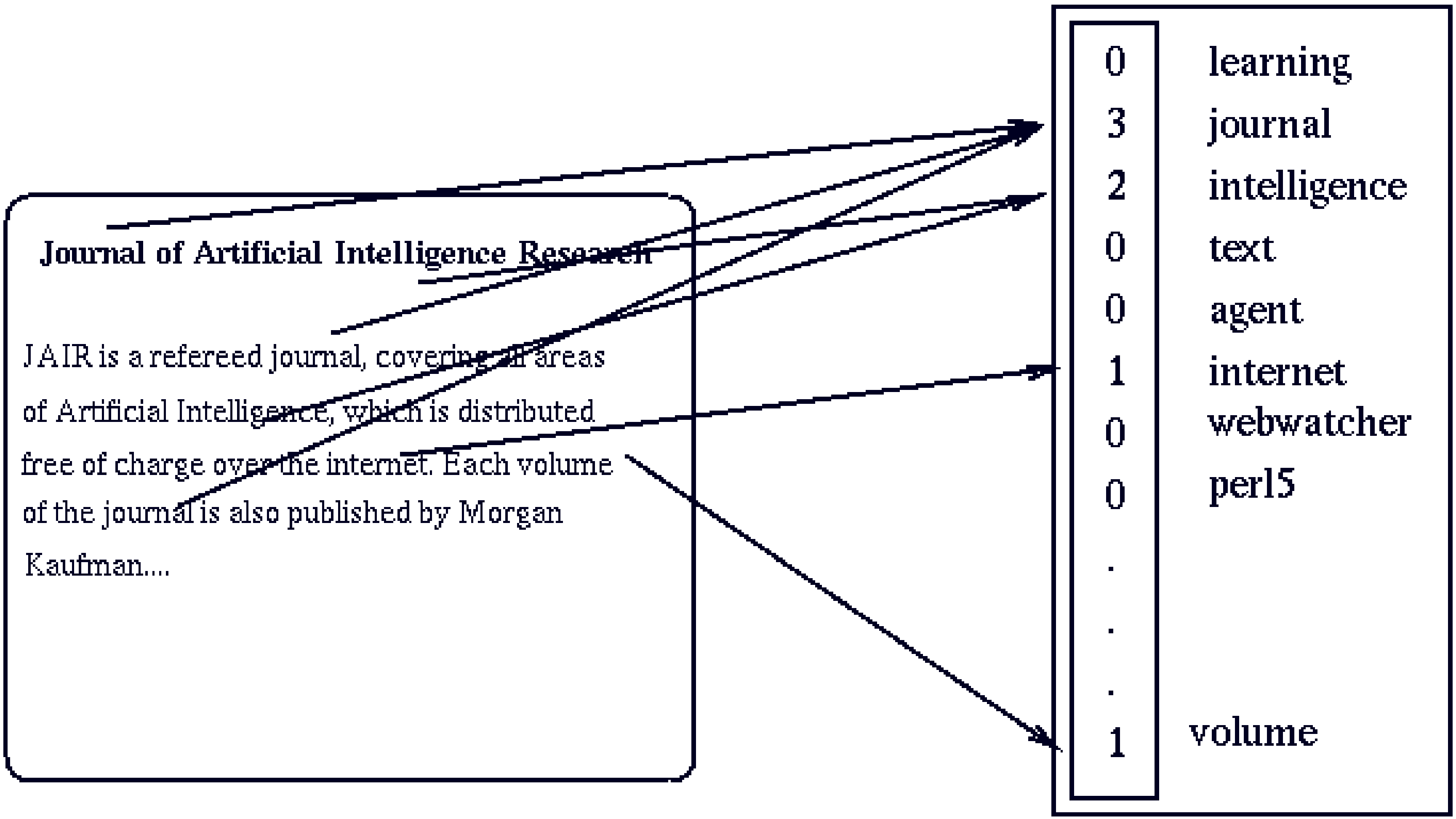
- BoW vector construction

- Mining of BoW vector table
  - Feature selection, Document similarity computation
  - Text mining: Categorization, Clustering, Summarization, …

Stemming and Lemmatization

- Different forms of the same word usually problematic for text data analysis
  - because they have different spelling and similar meaning (e.g. learns, learned, learning,…)
  - usually treated as completely unrelated words
- Stemming is a process of transforming a word into its stem
  - cutting off a suffix (eg., smejala -> smej)
- Lemmatization is a process of transforming a word into its normalized form
  - replacing the word, most often replacing a suffix (eg., smejala -> smejati)

Bag-of-Words document representation



Word weighting

- In bag-of-words representation each word is represented as a separate variable having numeric weight.
- The most popular weighting schema is normalized word frequency TFIDF:

$$tfidf(w) = tf \, . \log\left(\frac{N}{df(w)}\right)$$

  - Tf(w) – term frequency (number of word occurrences in a document)
  - Df(w) – document frequency (number of documents containing the word)
  - N – number of all documents
  - Tfidf(w) – relative importance of the word in the document

The word is more important if it appears several times in a target document

The word is more important if it appears in less documents

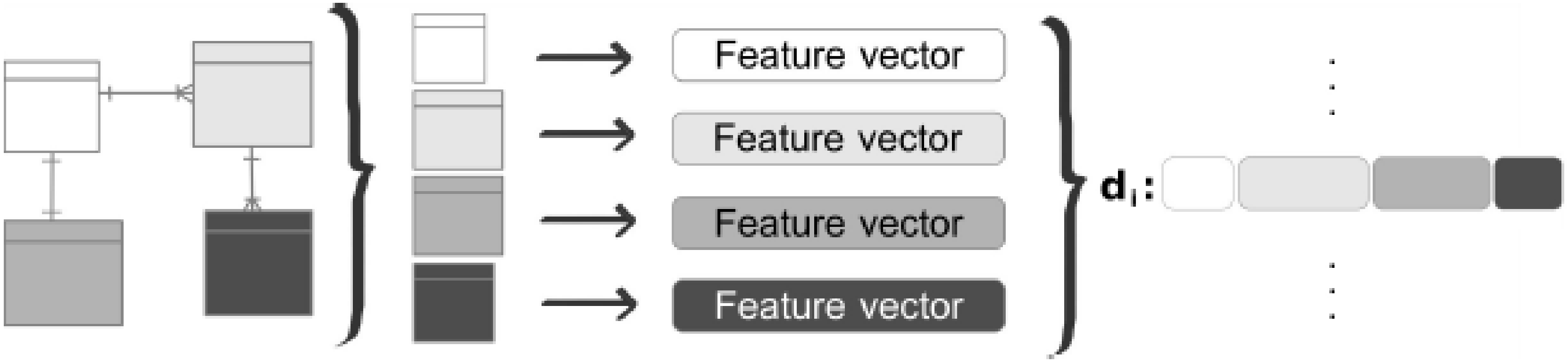
Cosine similarity between document vectors

- Each document D is represented as a vector of TF-IDF weights
- Similarity between two vectors is estimated by the similarity between their vector representations (cosine of the angle between the two vectors):

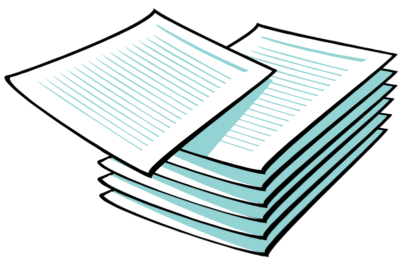
$$Similarity\ (D_1, D_2) = \frac{\sum_i x_{1i} x_{2i}}{\sqrt{\sum_j x_j^2}\sqrt{\sum_k x_k^2}}$$

Wordification Methodology

- Transform a relational database to a document corpus
  - For each individual (row) in the main table, concatenate words generated for the main table with words generated for the other tables, linked through external keys



Text mining



Step 1

BoW vector construction

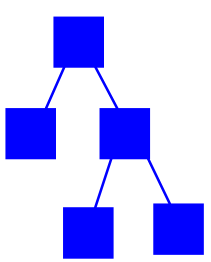
1. BoW features construction
2. Table of BoW vectors construction

Document	Word1	Word2	…	WordN	Class
d1	1	1	0	1	NO
d2	1	1	0	0	YES
d3	1	1	1	1	NO
d4	1	1	1	0	YES
d5	1	0	0	1	NO
d6-d13	…	…	…	…	…
d14	0	0	0	0	YES
d15	0	0	1	1	NO
d16	0	0	1	0	NO
d17	0	1	0	1	NO
d18	0	1	0	0	NO
d19-d23	…	…	…	…	…
d24	0	0	1	0	NO

Document	Word1	Word2	…	WordN	Class
d1	1	1	0	1	NO
d2	1	1	0	0	YES
d3	1	1	1	1	NO
d4	1	1	1	0	YES
d5	1	0	0	1	NO
d6-d13	…	…	…	…	…
d14	0	0	0	0	YES
d15	0	0	1	1	NO
d16	0	0	1	0	NO
d17	0	1	0	1	NO
d18	0	1	0	0	NO
d19-d23	…	…	…	…	…
d24	0	0	1	0	NO

Step 2

Data Mining



model, patterns, clusters,

…

Wordification Methodology

- One individual of the main data table in the relational database ~ one text document
- Features (attribute values) ~ the words of this document
- Individual words (called **word-items** or **witems**) are constructed as combinations of:

$$[table\ name]\_[attribute\ name]\_[value]$$

- **n-grams** are constructed to model feature dependencies:

$$[witem_1]\_[witem_2]\_\ \ldots\ \_[witem_n]$$

Wordification Methodology

- Transform a relational database to a document corpus
- Construct BoW vectors with TF-IDF weights on words

  (optional: Perform feature selection)
- Apply text mining or propositional learning on BoW table

Wordification

TRAIN

trainID	eastbound
t1	east
...	...
t5	west
...	...

CAR

carID	shape	roof	wheels	train
c11	rectangle	none	2	t1
c12	rectangle	peaked	3	t1
...	...	...	...	...
c51	rectangle	none	2	t5
c52	hexagon	flat	2	t5
...	...	...	...	...

t1: [car_roof_none, car_shape_rectangle, car_wheels_2, car_roof_none__car_shape_rectangle, car_roof_none__car_wheels_2, car_shape_rectangle__car_wheels_2, car_roof_peaked, car_shape_rectangle, car_wheels_3, car_roof_peaked__car_shape_rectangle, car_roof_peaked__car_wheels_3, car_shape_rectangle__car_wheels_3], **east**

Wordification

t1: [car_roof_none, car_shape_rectangle, car_wheels_2, car_roof_none__car_shape_rectangle, car_roof_none__car_wheels_2, car_shape_rectangle__car_wheels_2, car_roof_peaked, car_shape_rectangle, car_wheels_3, car_roof_peaked__car_shape_rectangle, car_roof_peaked__car_wheels_3, car_shape_rectangle__car_wheels_3], **east**

t5: [car_roof_none, car_shape_rectangle, car_wheels_2, car_roof_none__car_shape_rectangle, car_roof_none__car_wheels_2, car_shape_rectangle__car_wheels_2, car_roof_flat, car_shape_hexagon, car_wheels_2, car_roof_flat__car_shape_hexagon, car_roof_flat__car_wheels_2, car_shape_hexagon__car_wheels_2], **west**

TF-IDF calculation for BoW vector construction:

	car_shape_rectangle	car_roof_peaked	car_wheels_3	car_roof_peaked__car_shape_rectangle	car_shape_rectangle__car_wheels_3	...	class
t1	0.000	0.693	0.693	0.693	0.693	...	east
...	...	...	...	...	...	...	...
t5	0.000	0.000	0.000	0.000	0.000	...	west
...	...	...	...	...	...	...	...

TF-IDF weights

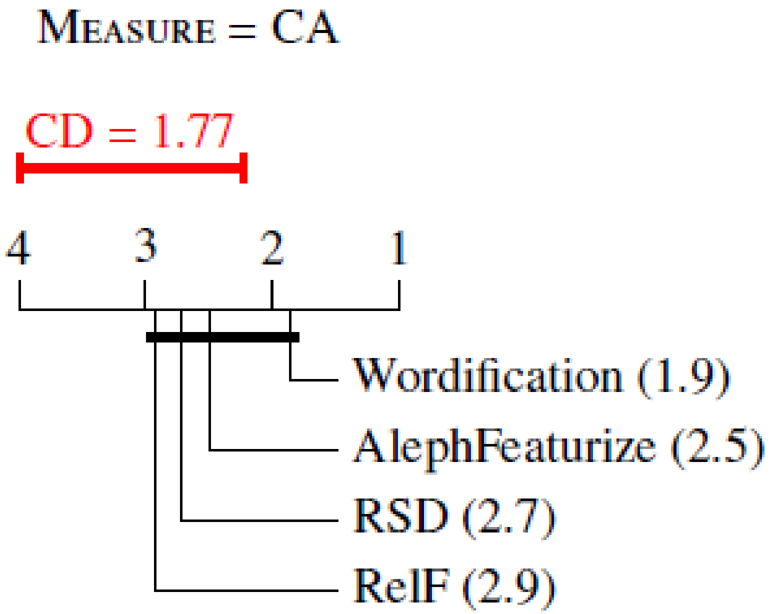
- No explicit use of existential variables in features, TF-IDF instead
- The weight of a word indicates how relevant is the feature for the given individual
- The TF-IDF weights can then be used either for filtering words with low importance or for using them directly by a propositional learner (e.g. J48)

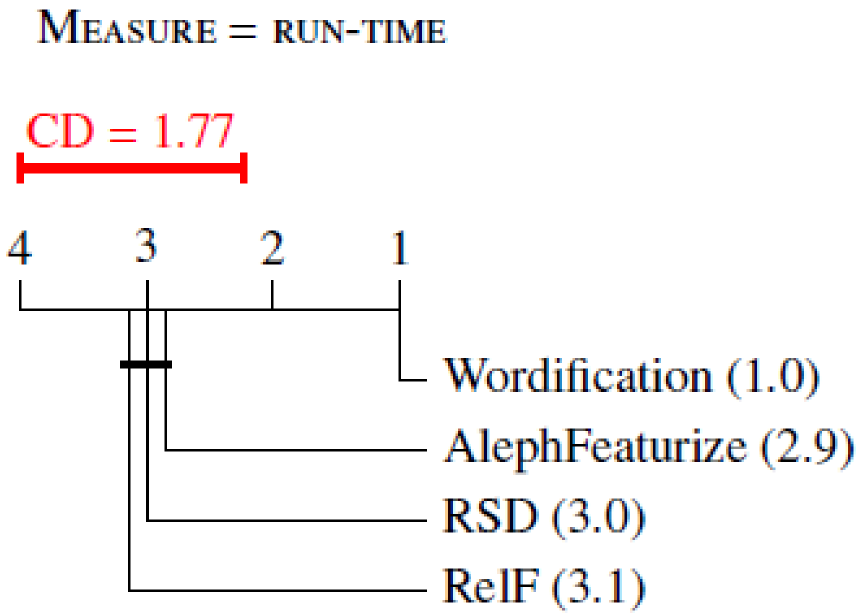
Experiments

- Cross-validation experiments on 8 relational datasets: Trains (in two variants), Carcinogenesis, Mutagenensis with 42 and 188 examples, IMDB, and Financial.
- Results (using J48 for propositional learning)
  - first applying Friedman test to rank the algorithms,
  - then post-hoc test Nemenyi test to compare multiple algorithms to each other

Experiments

- Cross-validation experiments on 8 relational datasets: Trains (in two variants), Carcinogenesis, Mutagenensis with 42 and 188 examples, IMDB, and Financial.
- Results (using J48 for propositional learning)





Experiments

Domain	Algorithm	J48-Accuracy[%]	J48-AUC	Run-time[s]
Trains without position	Wordification	55.00	0.51	**0.11**
	RelF	65.00	0.65	1.04
	RSD	65.00	0.68	0.53
	AlephFeaturize	**75.00**	**0.82**	0.40
Trains	Wordification	**95.00**	**0.91**	**0.12**
	RelF	65.00	0.62	1.06
	RSD	50.00	0.53	0.47
	AlephFeaturize	85.00	0.74	0.38
Mutagenesis42	Wordification	**97.62**	**0.93**	**0.39**
	RelF	80.95	0.59	2.11
	RSD	**97.62**	**0.93**	2.63
	AlephFeaturize	**97.62**	**0.93**	2.07
Mutagenesis188	Wordification	**95.74**	0.90	**1.65**
	RelF	75.53	0.79	7.76
	RSD	94.15	**0.91**	10.10
	AlephFeaturize	87.23	0.88	19.27
IMDB	Wordification	**84.34**	**0.79**	**1.23**
	RelF	79.52	0.73	32.49
	RSD	73.49	0.47	4.33
	AlephFeaturize	73.49	0.47	4.96
Carcinogenesis	Wordification	**61.09**	**0.62**	**1.79**
	RelF	54.71	0.53	16.44
	RSD	58.05	0.56	9.29
	AlephFeaturize	55.32	0.49	104.70
Financial	Wordification	86.75	0.48	**4.65**
	RelF	**97.00**	**0.91**	260.93
	RSD	86.75	0.48	533.68
	AlephFeaturize	86.75	0.48	525.86

Use Case: IMDB

- **IMDB subset:** Top 250 and bottom 100 movies
- Movies, actors, movie genres, directors, director genres
- Wordification methodology applied
- Association rules learned on BoW vector table

Use Case: IMDB

goodMovie ← director_genre_drama, movie_genre_thriller, director_name_AlfredHitchcock. (Support: 5.38% Confidence: 100.00%)

movie_genre_drama ← goodMovie, actor_name_RobertDeNiro.
(Support: 3.59% Confidence: 100.00%)

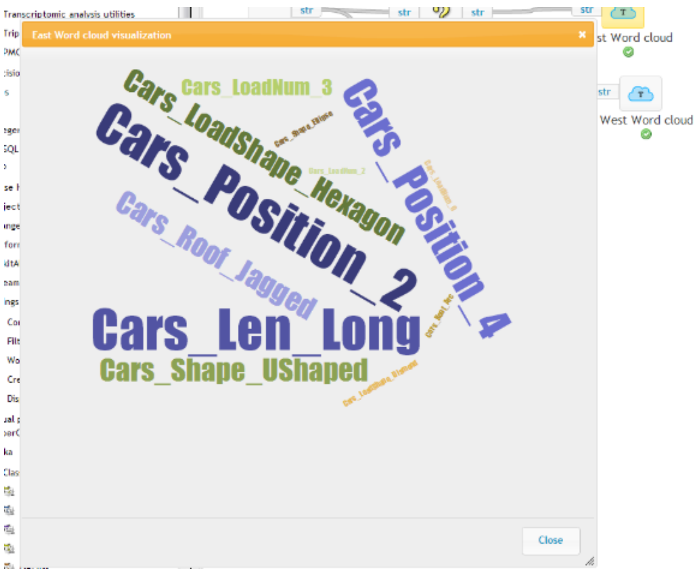
director_name_AlfredHitchcock ← actor_name_AlfredHitchcock.
(Support: 4.79% Confidence: 100.00%)

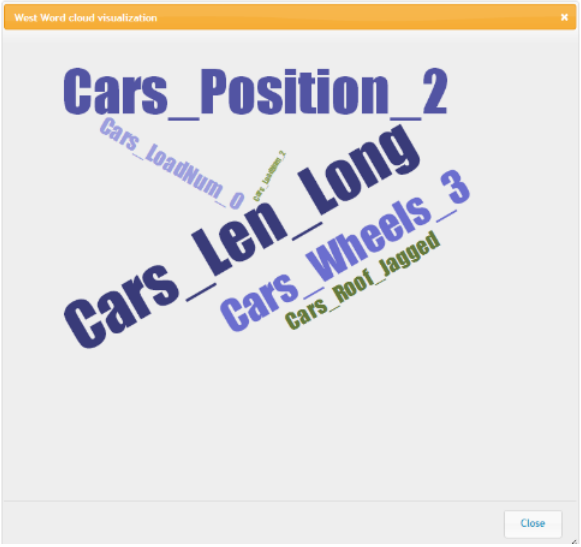
director_name_StevenSpielberg ← goodMovie, movie_genre_adventure, actor_name_TedGrossman.
(Support: 1.79% Confidence: 100.00%)

Summary

– Wordification methodology

– Allows for solving non-standard RDM tasks, including RDM clustering, **word cloud visualization**, **association rule learning**, topic ontology construction, outlier detection, …





Summary: From machine learning to Semantic Data Mining