Data and Text Mining
Introduction to Data Mining

2023 / 2024

Nada Lavrač

Department of Knowledge Technologies

Jožef Stefan Institute

Ljubljana, Slovenia

Introduction to Data Mining

8-11-2023

Nada Lavrač: Lesson 1 - Introduction

- Basics of Machine Learning
- Standard learning tasks
- Three generations of machine learning
- Advanced learning tasks

Nada Lavrač: Lesson 2 - Decision Tree Learning

- Basic decision tree learning algorithm
- Entropy and information gain heuristics
- Decision tree pruning
- Selected decision tree learning algorithms
- Regression tree learning

Introduction to Data Mining

8-11 and 15-11-2023

Nada Lavrač, Blaž Škrlj: Lesson 3 – Rule Learning

- Transforming decision trees to rules
- Classification rule learning
- Covering algorithm
- Association rule learning

Nada Lavrač, Blaž Škrlj: Lesson 4 – Text Mining

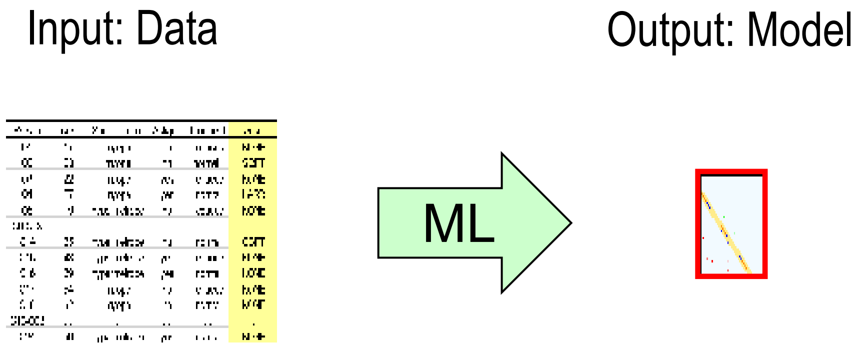
- Introduction to text mining
- Text mining process
- Text mining tasks and applications
- From BoW to dense text embeddings

Lesson 1:
Introduction to Data Mining

- Basics of Machine Learning
- Standard learning tasks
- Three generations of machine learning
- Advanced learning tasks

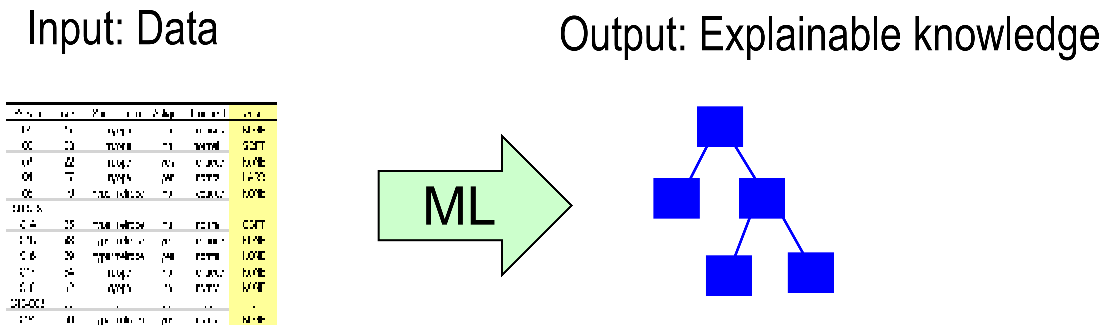
Basics of Machine Learning

- What is Machine Learning (ML)
  - Area of computer science, concerned with the development of computer algorithms that learn from data



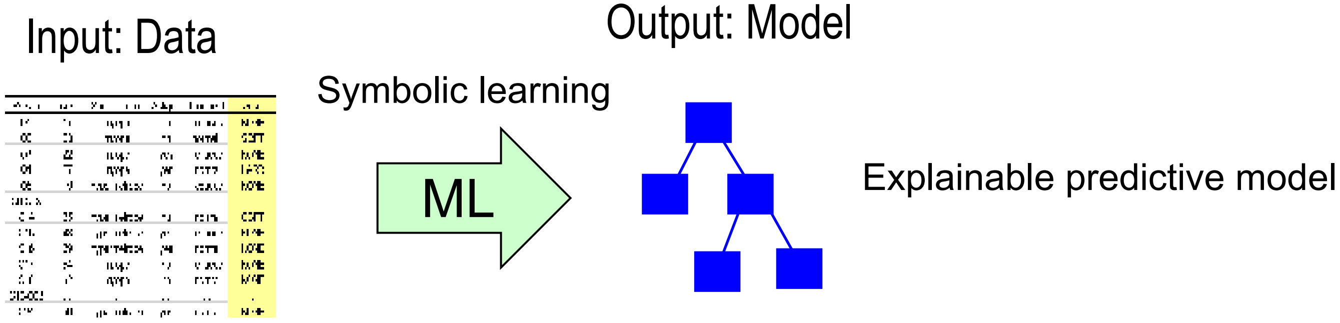
Basics of Machine Learning

- Origins of terms
  - Term **Machine learning** comes from early AI research in 1960s and 1970s: Perception of learning algorithms as “machines”, able to learn (generalize) from data automatically, without human intervention
  - Term **Inductive learning** refers to the capability of learners to generalize – to automatically induce models from data
  - Term **Symbolic learning** refers to the capability of learners to induce explainable knowledge from data - XAI

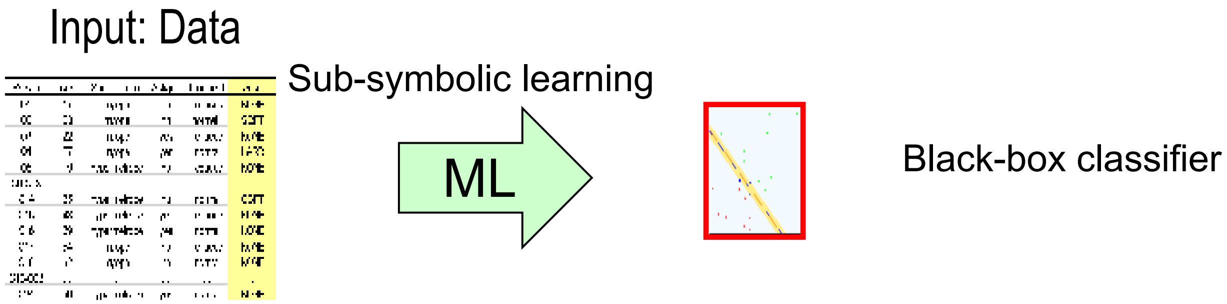


Basics of Machine Learning

- Two basic learning settings
  - **Symbolic learning** – inducing explainable predictive models, such as decision trees or classification rules



  - **Sub-symbolic (neural) learning** – inducing black-box classifiers, such as neural networks



Basics of Machine Learning

- Early history of symbolic learning algorithms:
  - Early rule learning algorithms: AQ (Michalski 1969), …
  - Early decision tree learning algorithms since 1970s: ID3 (Quinlan 1979), …
  - Early regression tree learners CART (Breiman et al. 1984), …
  - Advantage: explainable models, but less accurate classifiers

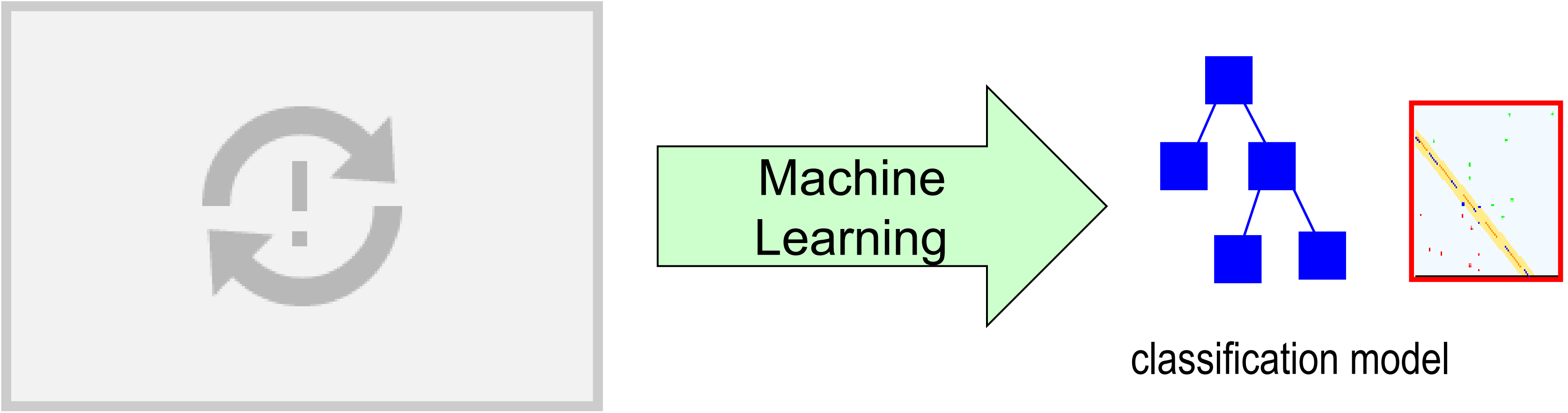
- Sub-symbolic (neural) learning algorithms
  - Early perceptron (Rosenblatt 1962), backpropagation neural networks (Rumelhart et al. 1986), …
  - Modern deep neural networks (Hinton & Salakhutdinov 2006, Goodfellow et al. 2016), …
  - Advantage: more accurate classifiers, but black-box models

Basics of Machine Learning

- Learning tasks depend on the type of input data and the goal of learning
  - tabular data – prediction and classification, clustering, …
  - relational databases – relational learning, inductive logic programming, …
  - graphs – network analysis, social network analysis, link prediction, node classification, network completion, …
  - texts – text mining, sentiment analysis, hate speech detection, …
  - Web pages – Web page recommendation, …
  - heterogeneous data and heterogeneous information networks – classification of data instances, node classification, link prediction, …

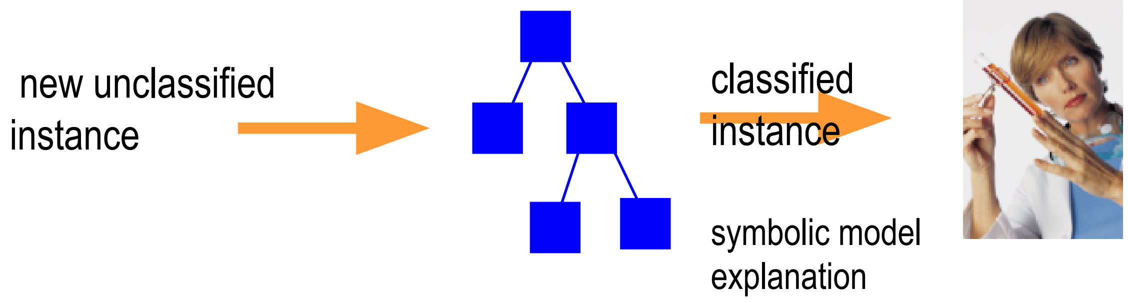
Basics of Machine Learning

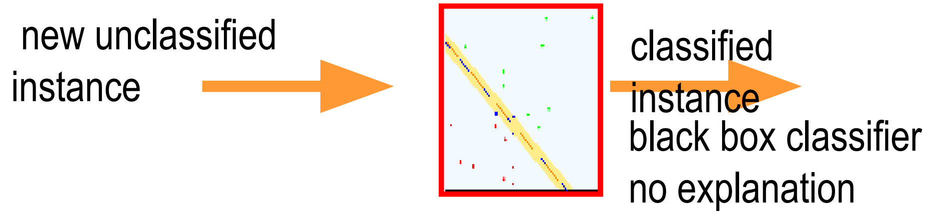
- Definition of a standard machine learning task
  - **Given**: class-labeled data set (e.g., transaction data table, relational database, text documents, Web pages, …)
  - **Find**: a classification model, able to predict new instances



Basics of Machine Learning

- Standard machine learning scenario
  1. Use a ML algorithm to learn a predictive model from class-labeled data
  2. Use the induced model to predict the class of new (unlabeled) data instances





Basics of Machine Learning

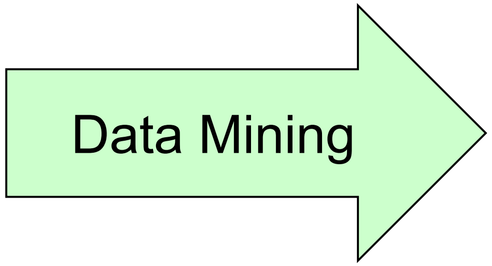
Illustrative example: Contact lens data set

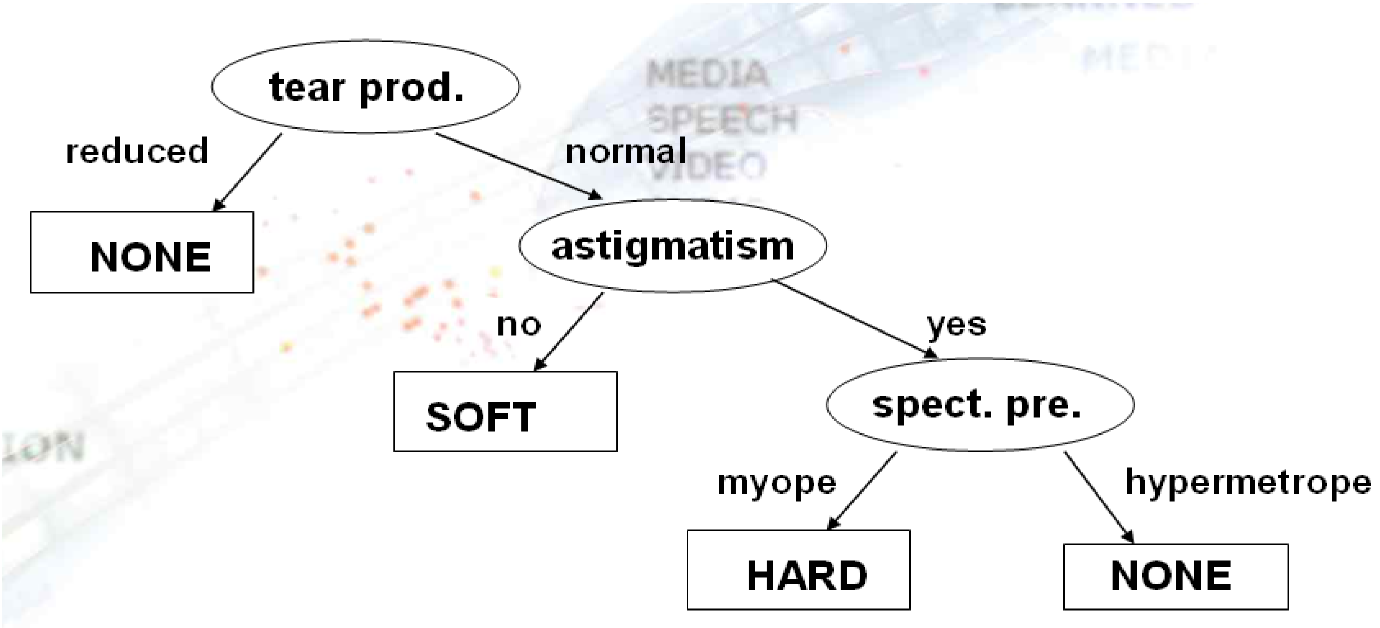
Person	Age	Spect. presc.	Astigm.	Tear prod.	Lenses
O1	17	myope	no	reduced	NONE
O2	23	myope	no	normal	SOFT
O3	22	myope	yes	reduced	NONE
O4	27	myope	yes	normal	HARD
O5	19	hypermetrope	no	reduced	NONE
O6-O13	...	...	...	...	...
O14	35	hypermetrope	no	normal	SOFT
O15	43	hypermetrope	yes	reduced	NONE
O16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

Basics of Machine Learning

Decision tree learning from Contact lens data

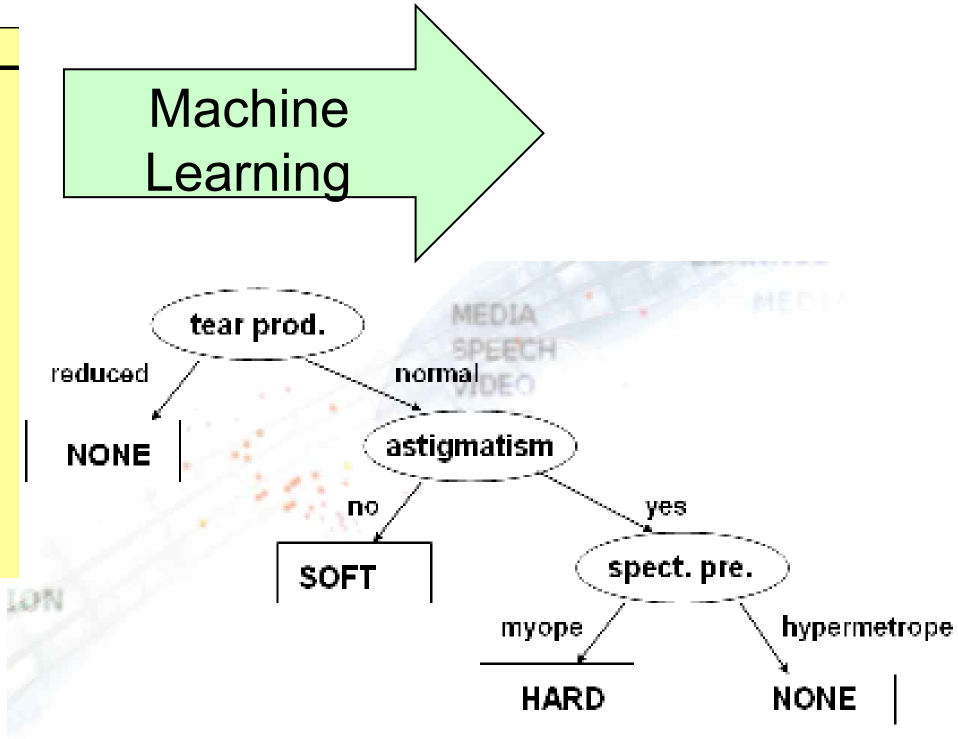
Person	Age	Spect. presc.	Astigm.	Tear prod.	Lenses
O1	young	myope	no	reduced	NONE
O2	young	myope	no	normal	SOFT
O3	young	myope	yes	reduced	NONE
O4	young	myope	yes	normal	HARD
O5	young	hypermetrope	no	reduced	NONE
O6-O13	...	...	...	...	...
O14	ore-presbyc	hypermetrope	no	normal	SOFT
O15	ore-presbyc	hypermetrope	yes	reduced	NONE
O16	ore-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





Basics of Machine Learning
Rule learning 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



lenses=NONE ← tear production=reduced
lenses=NONE ← tear production=normal AND astigmatism=yes AND
spect. presc.=hypermetrope
lenses=SOFT ← tear production=normal AND astigmatism=no
lenses=HARD ← tear production=normal AND astigmatism=yes AND
spect. presc.=myope
lenses=NONE ←

Basics of Machine Learning
Data Mining

dat

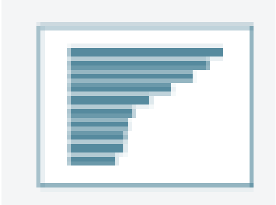
a

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

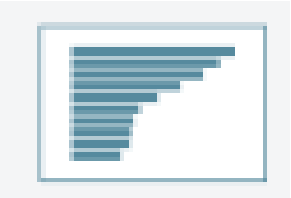
knowledge discovery
from data

Data Mining



patterns

Given: class labeled or non-labeled data
Find: a set of interesting patterns, explaining the data

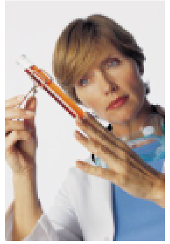


IF
Tear prod. = reduced

THEN
Lenses = NONE

symbolic patterns

explanation



Basics of Machine Learning

Pattern discovery 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

Rule:

IF
Tear prod. =
reduced

THEN
Lenses =
NONE

Basics of Machine Learning
Summary

- Basic definition of Machine Learning
  - Computer algorithms/machines that learn predictive models from class-labeled data
- Extended definition of Machine Learning - Used interchangeably with the term Data Mining
  - computer algorithms/machines that learn patterns or models from class-labeled or non-labeled data
  - sometimes used to denote the practical use of ML techniques applied to solving real-life data analysis problems
- Deep Learning - Used in popular literature interchangeably with the term AI ??

Introduction to Data Mining

- Basics of Machine Learning
- Standard learning tasks
- Three generations of machine learning
- Advanced learning tasks

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-class 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 attribute

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

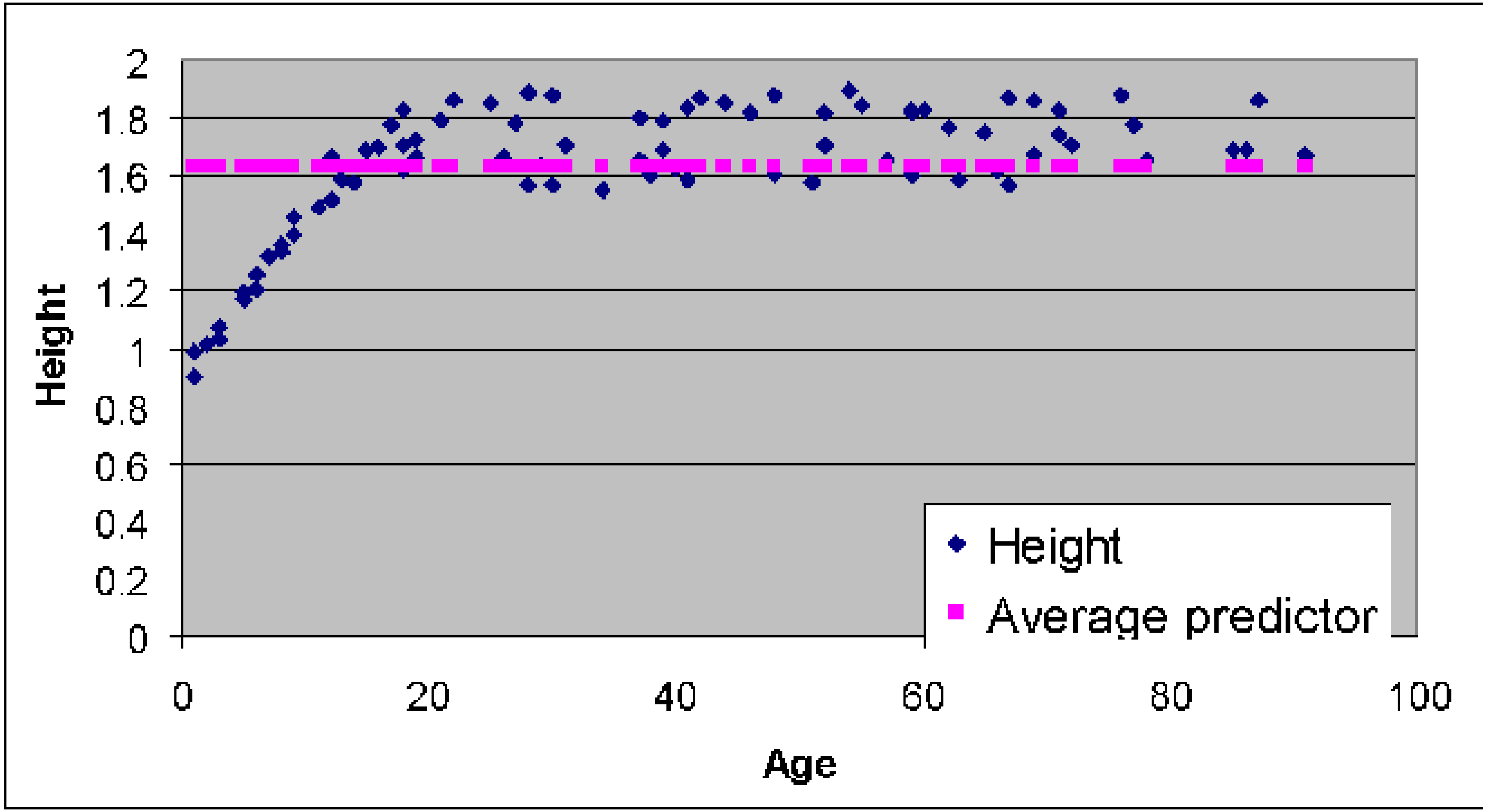
Example regression problem

- 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
...	...

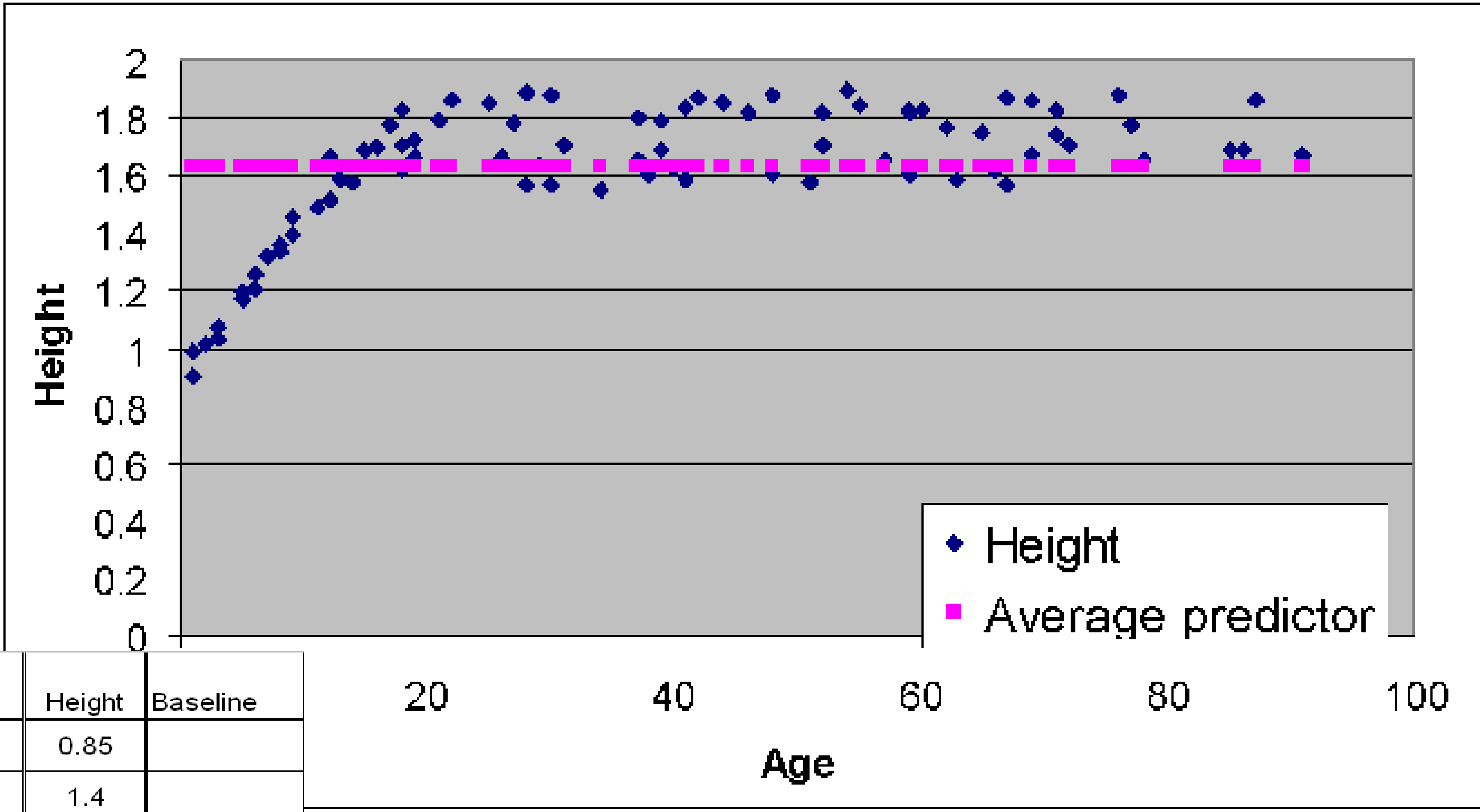
Baseline numeric model

- Average of the target variable



Baseline numeric predictor

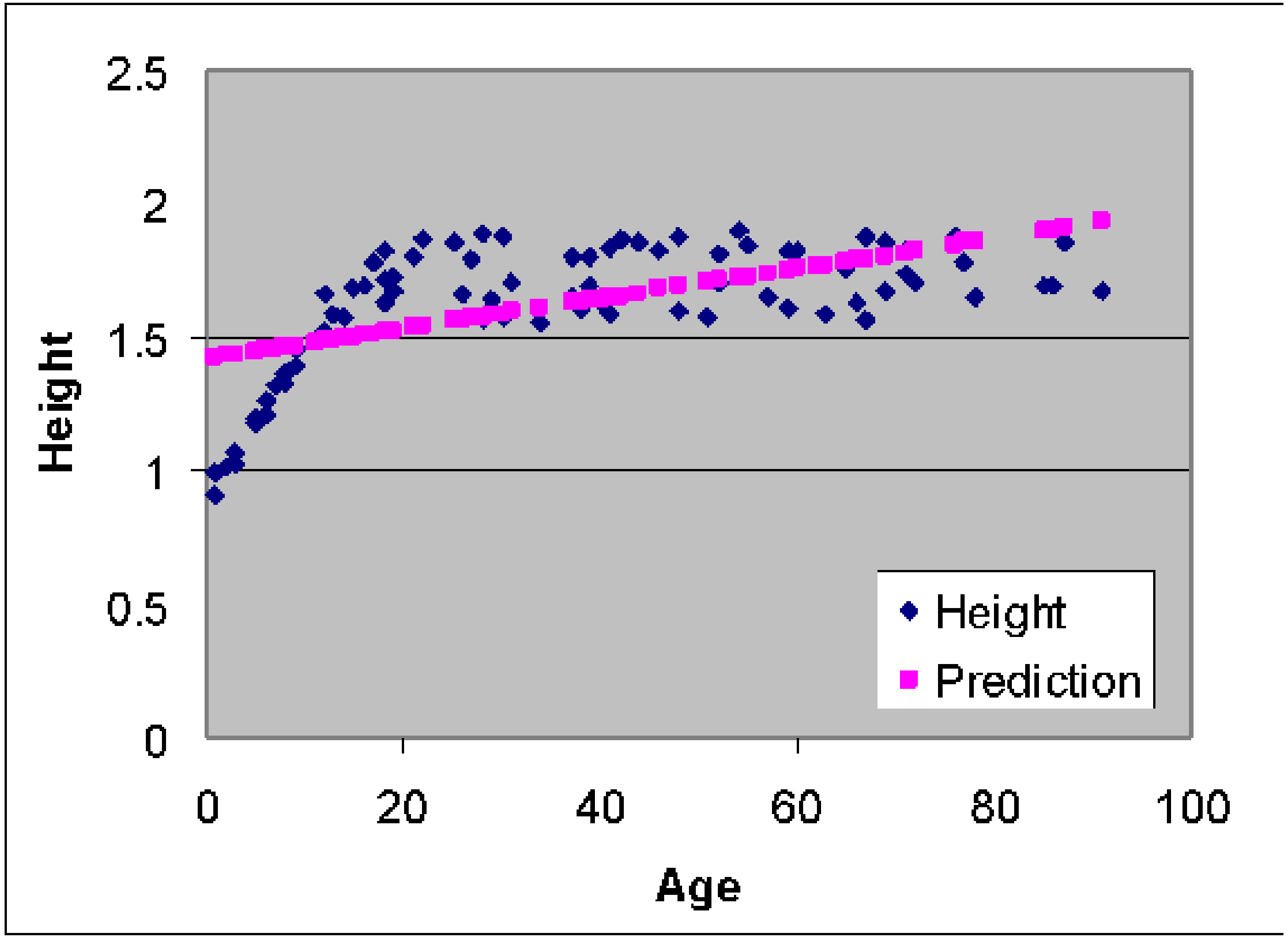
- Average of the target variable is 1.63



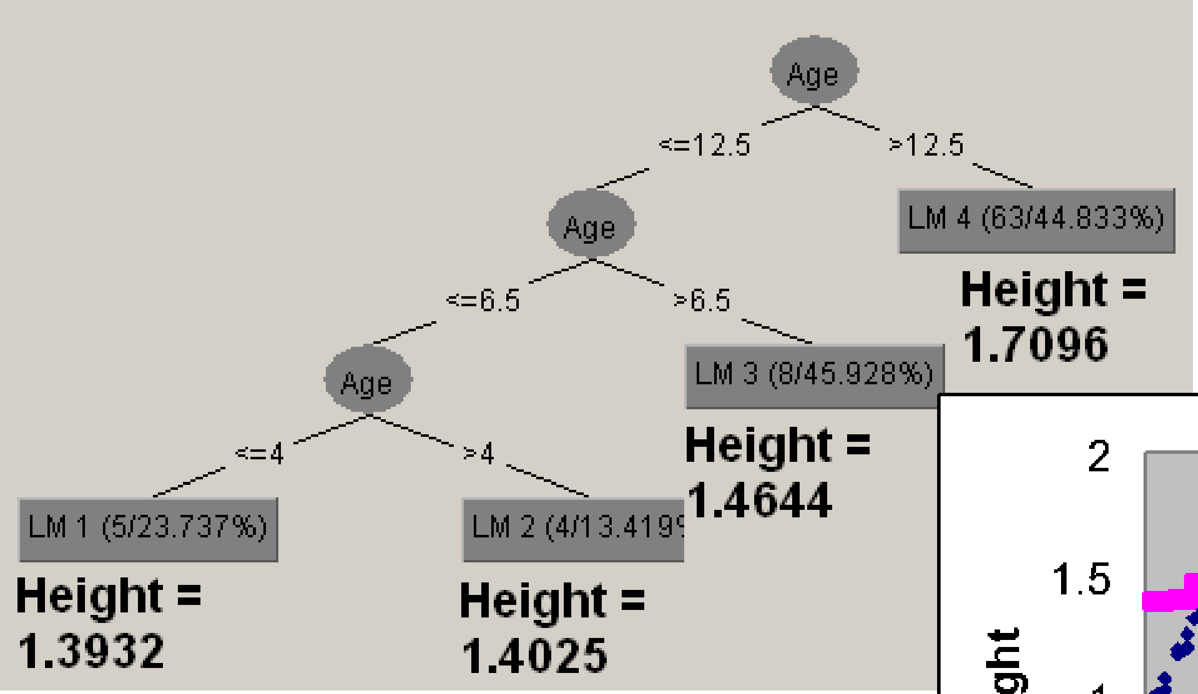
Age	Height	Baseline
2	0.85	
10	1.4	
35	1.7	
70	1.6	

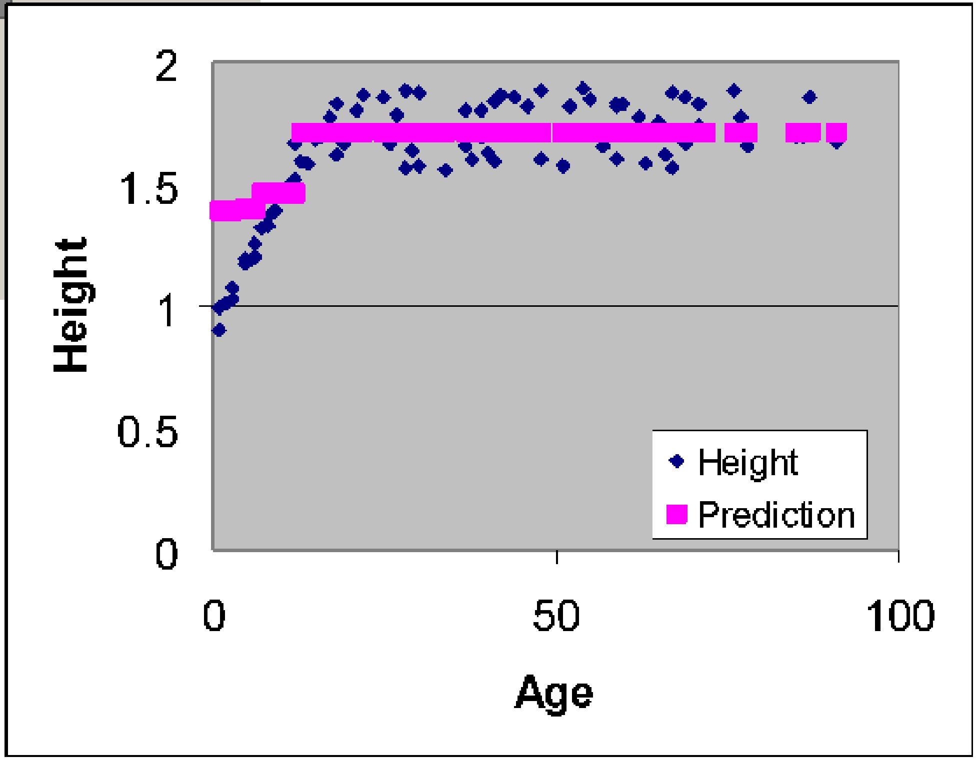
Linear Regression Model

Height = 0.0056 * Age + 1.4181



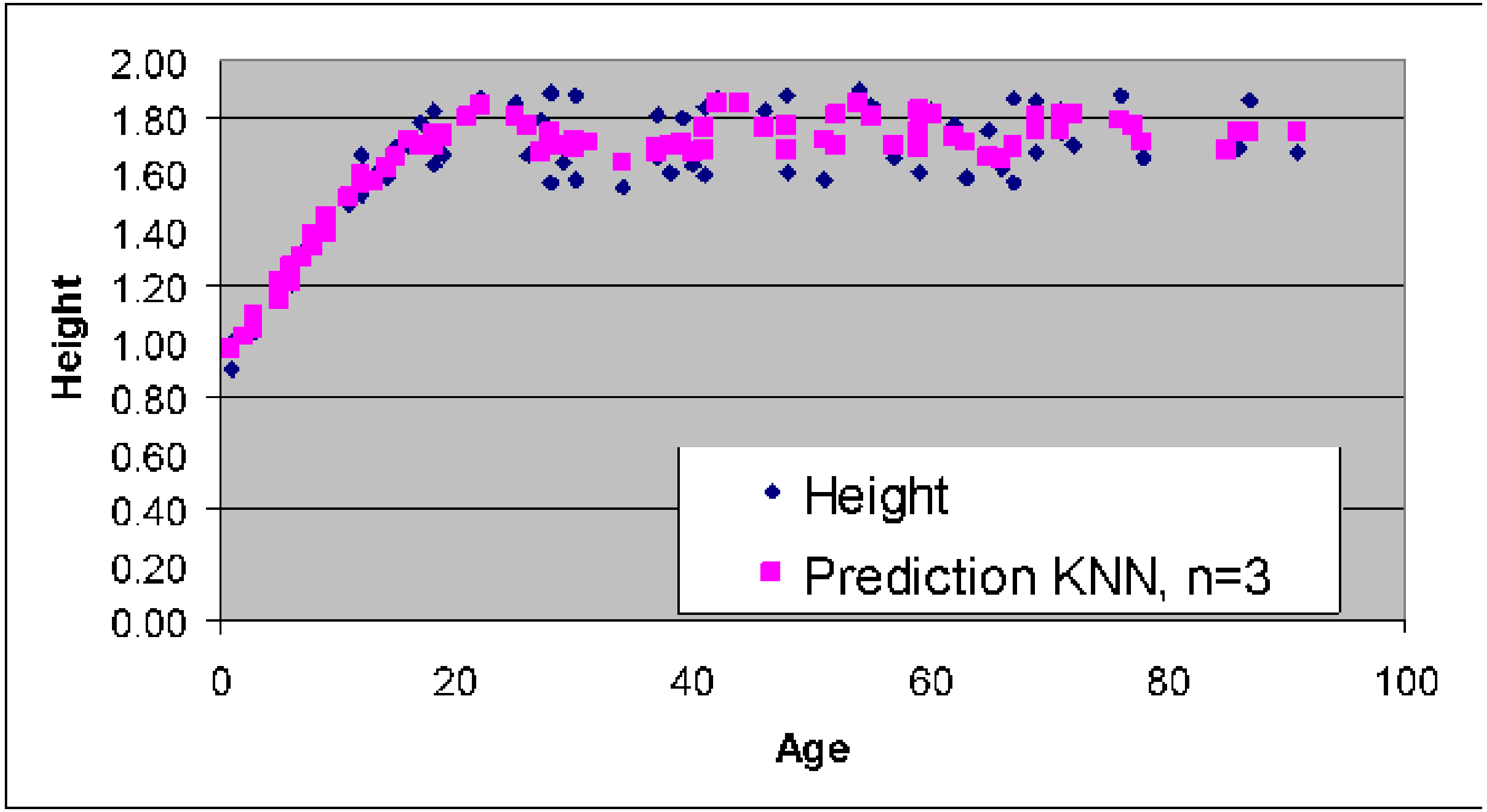
Regression tree





Simple sub-symbolic classifier:
K nearest neighbors (kNN)

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



Lesson 1:
Introduction to Data Mining

- Basics of Machine Learning
- Standard learning tasks
- Three generations of machine learning
- Advanced learning tasks

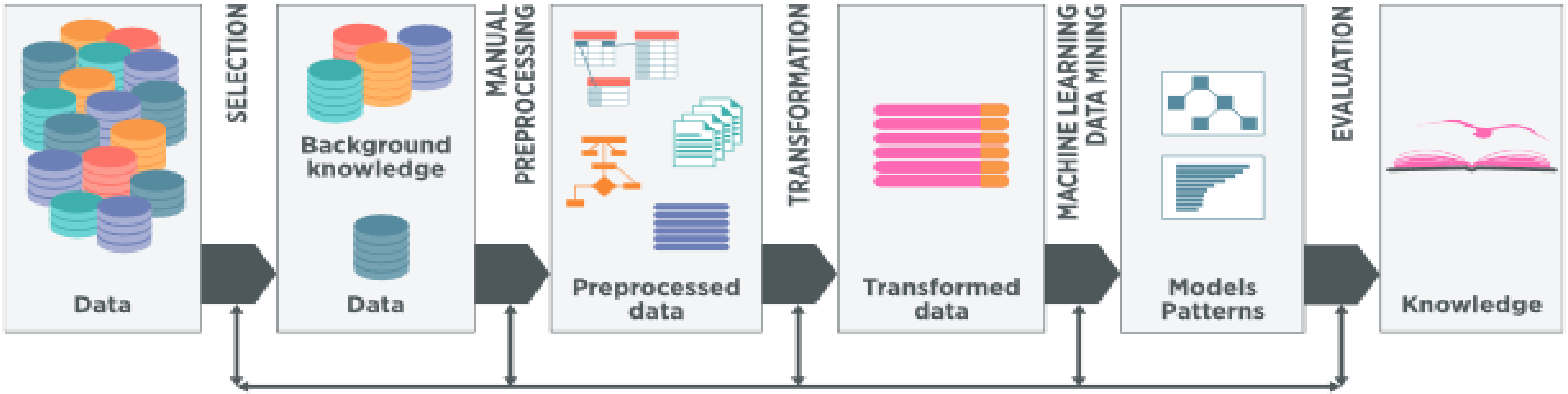
First Generation Machine Learning

- **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

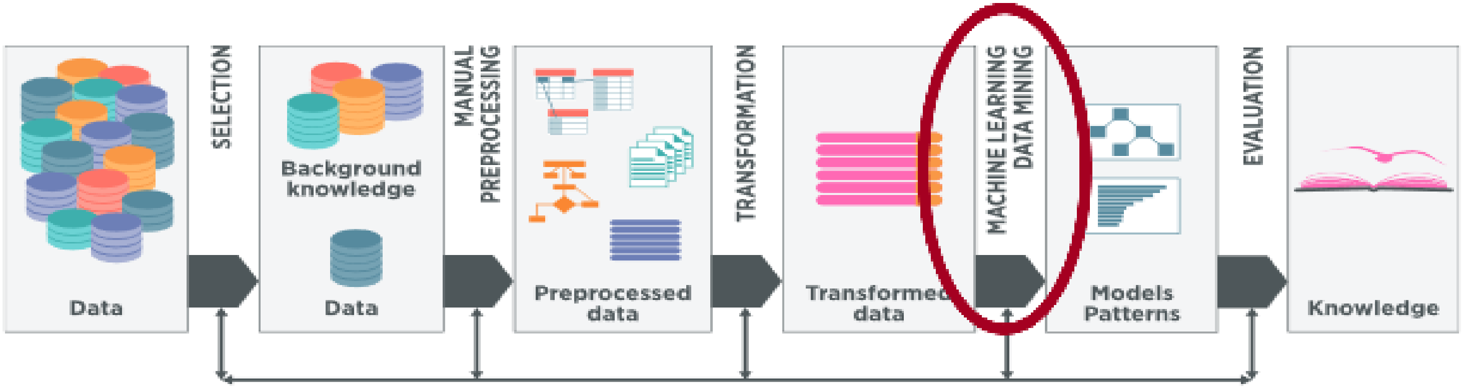
Second Generation Machine Learning

- Developed since 1990s:
  - Focused on data mining tasks characterized by large datasets described by large numbers of attributes
  - Addressing the entire process of Knowledge Discovery in Databases (KDD): process understandable models or patterns in 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
  - CRISP-DM methodology
  - KDD buzzword since 1996



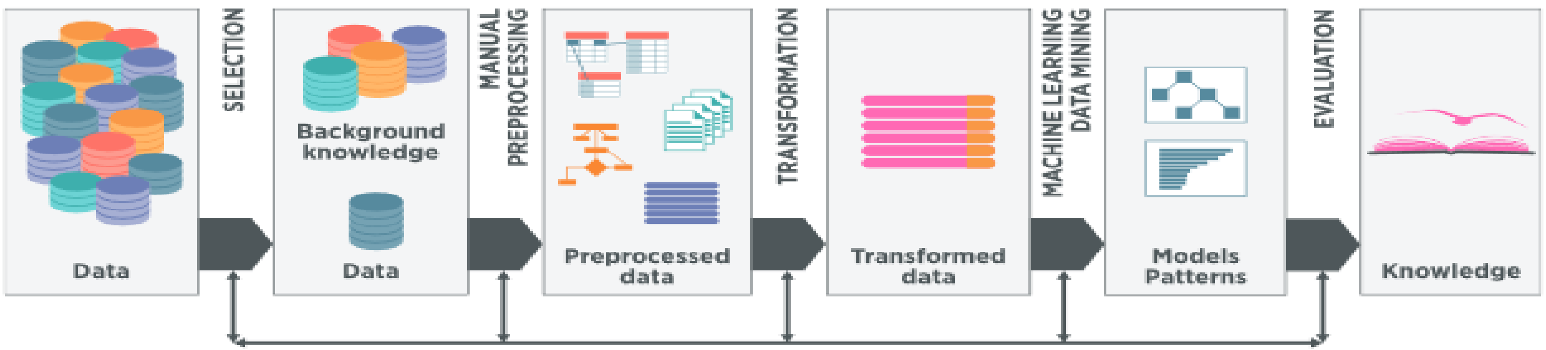
Second Generation Machine Learning
KDD Process



- KDD process (CRISP-DM methodology) involves several phases:
  - data preparation
  - machine learning, data mining, statistics, …
  - evaluation and use of discovered patterns
- Machine Learning / Data Mining is the key step in the 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 Machine Learning

– Industrial KDD standard: CRISP-DM methodology (1997)



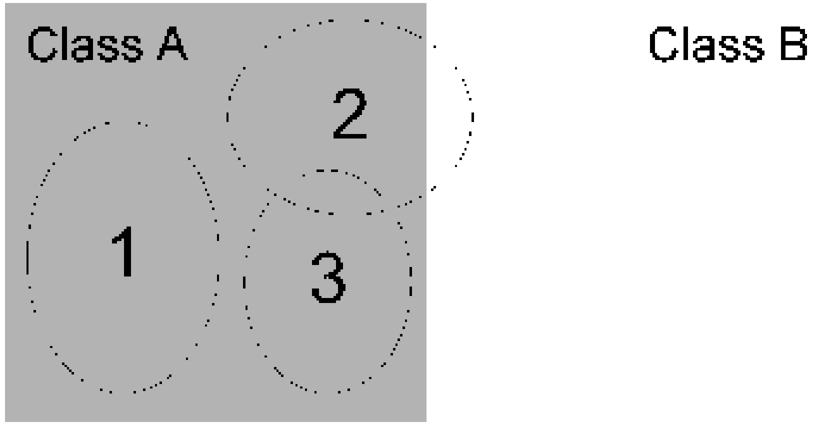
– New conferences on practical aspects of data mining and knowledge discovery: KDD, PKDD, …
– New learning tasks and efficient learning algorithms:
  - Learning descriptive patterns: association rule learning, subgroup discovery, …
  - Learning predictive models: Bayesian network learning, Support Vector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

Second Generation Machine Learning
Subgroup Discovery learning task

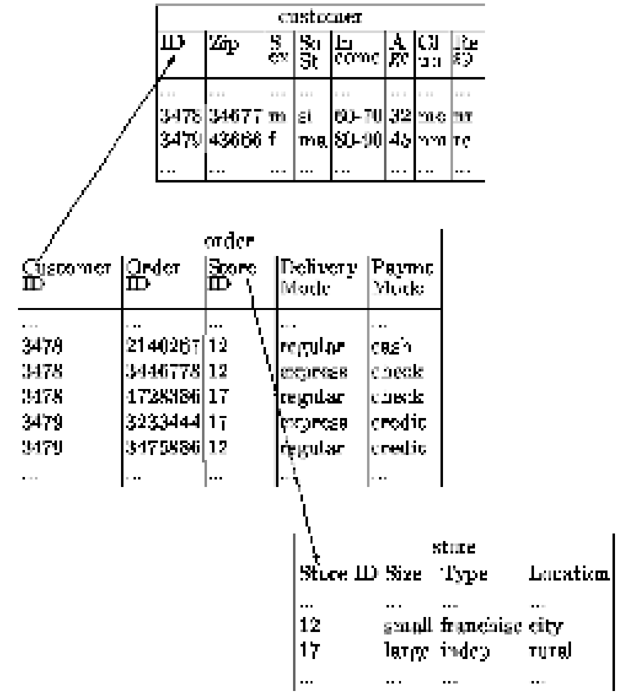
- Data transformation:
  - binary class values (positive vs. negative examples of Target class)

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

- 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.



Second Generation Machine Learning
Relational Data Mining task



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

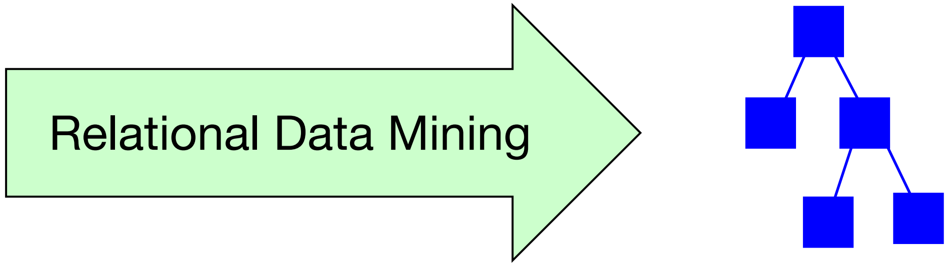
Customer ID	Order ID	Store ID	Delivery Mode	Paymt Mode
...	...	...	...	...
3478	2140267	12	regular	cash
3478	3446778	12	express	check
3478	1728386	17	regular	check
3479	3233444	17	express	credit
3479	3475886	12	regular	credit
...	...	...	...	...

store

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

Relational representation of customers, orders and stores.

knowledge discovery
from data



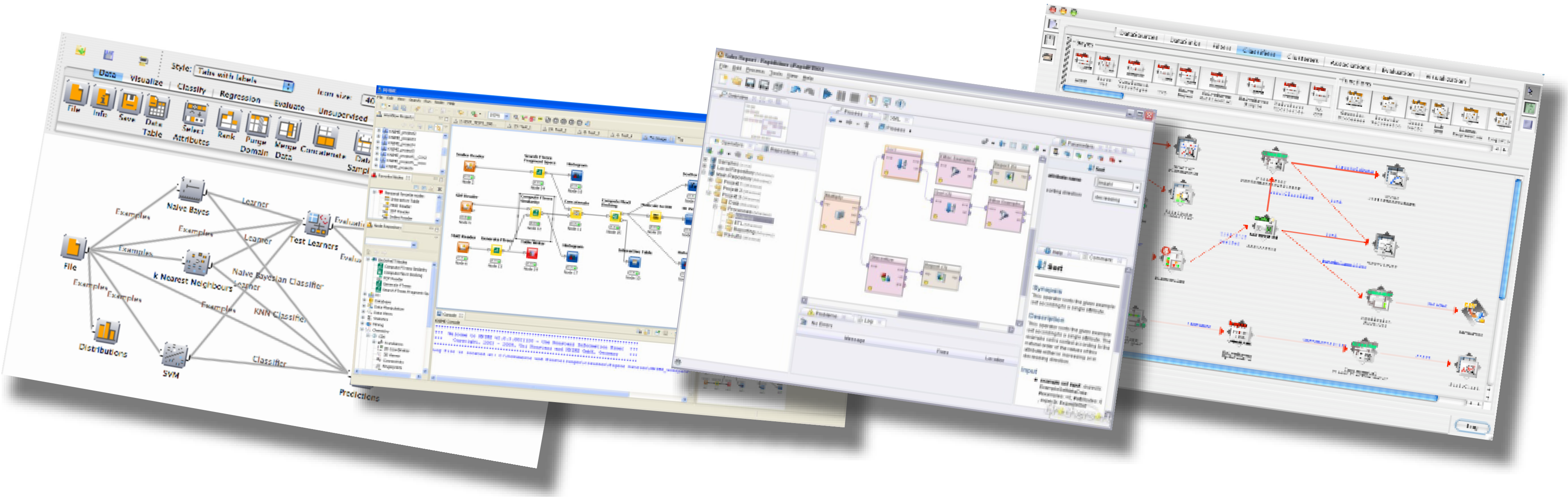
model, patterns,
…

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

Find: a classification model, a set of patterns

Second Generation Data Mining Platforms

Orange, WEKA, KNIME, RapidMiner, …



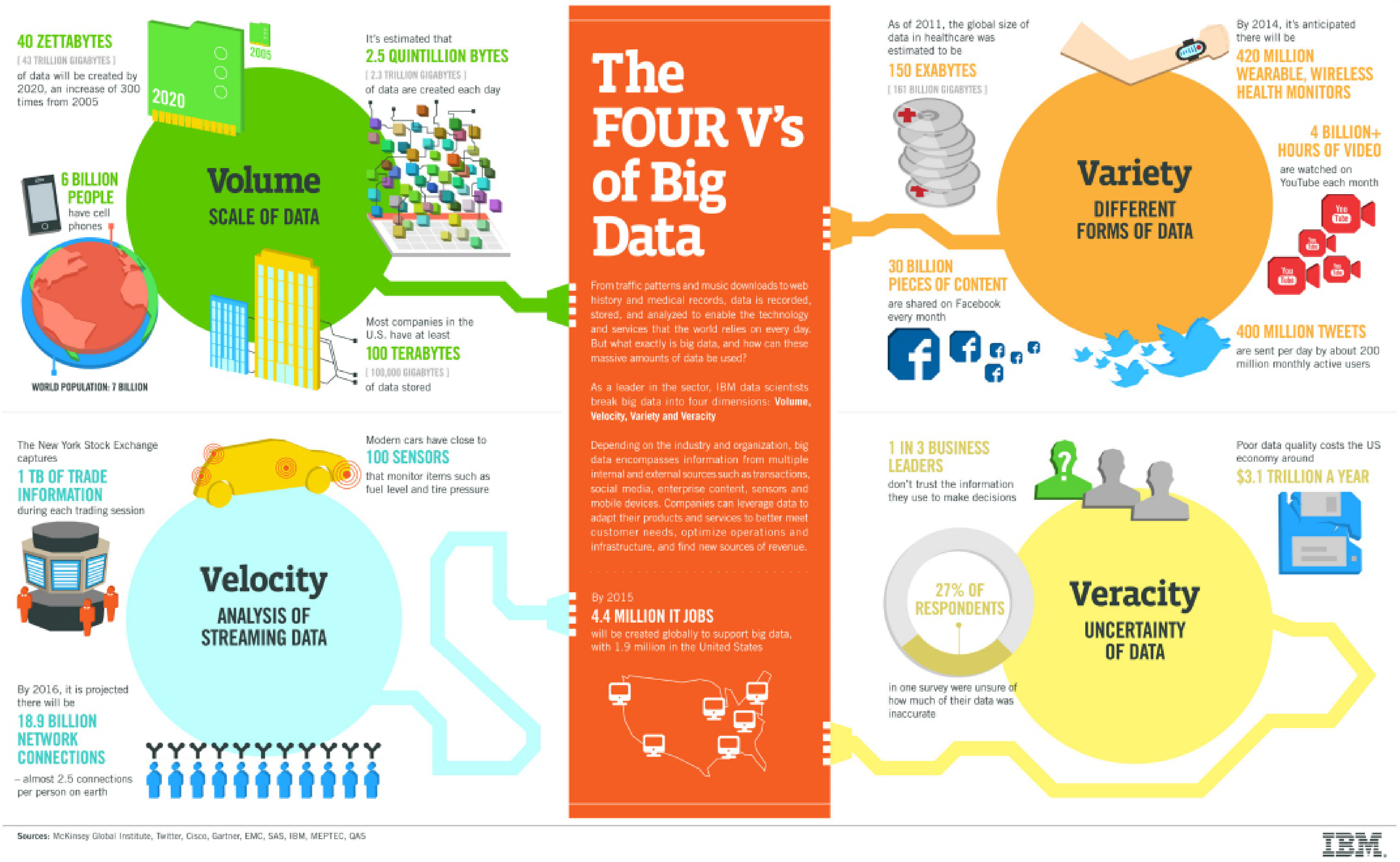
- include numerous data mining algorithms
- enable data and model visualization
- like Orange, Taverna, WEKA, KNIME, RapidMiner, also enable complex **workflow** construction

Second Generation Machine Learning
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.

Second Generation Machine Learning

The 4 Vs of Big Data

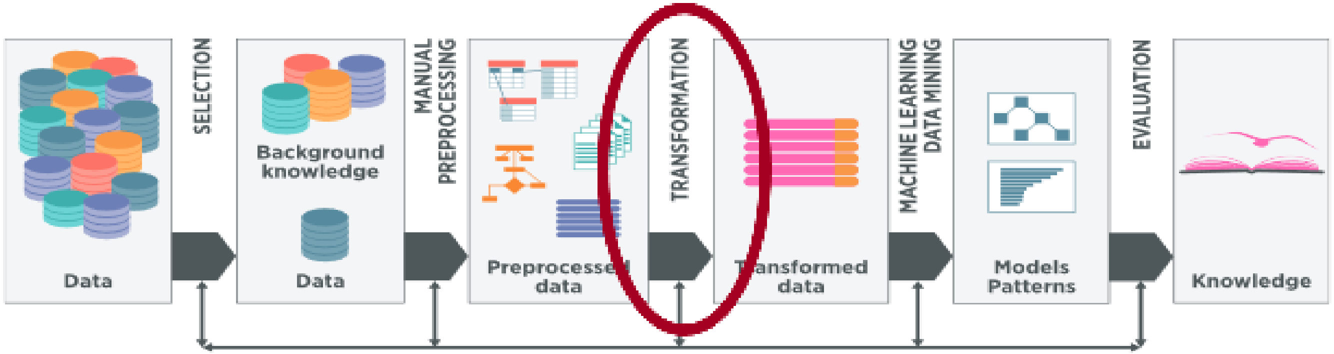


Second Generation Machine Learning
Data Science

- 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.

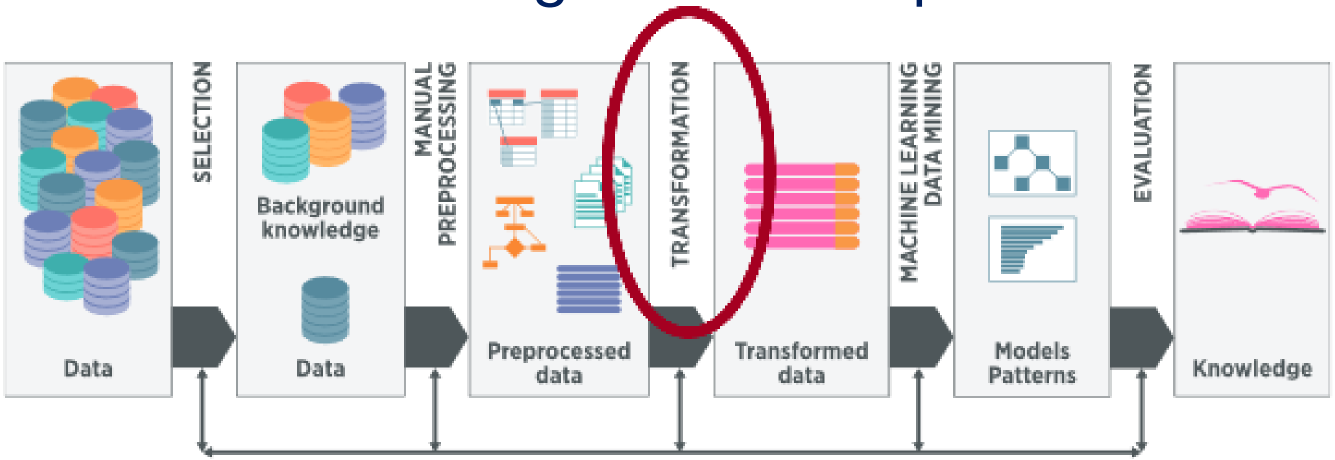
Third Generation Machine Learning

- **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, Semantic data mining, Text mining, …
  - Lots of emphasis on automated **data transformation, i. e. representation learning**



Third Generation Machine Learning

- Representation learning in the KDD process



- Representation learning = Automated data transformation, performed on manually preprocessed data

- Data transformation requires handling heterogeneous data
  - Data (feature vectors, documents, pictures, data streams, …)
  - Background knowledge (multi-relational data tables, networks, text corpora, …)

Current Generation Machine Learning

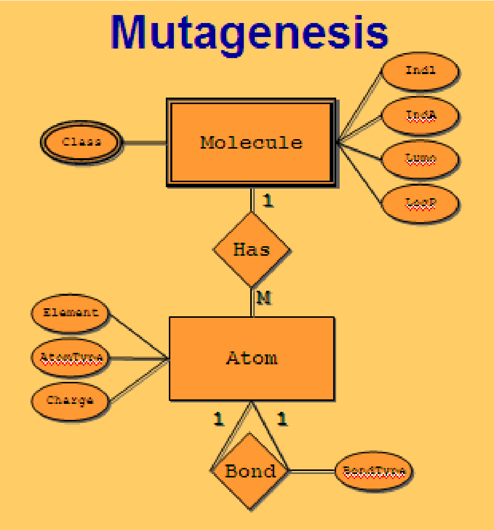
- Automated representation learning without manual data preprocessing
- Using pre-trained deep neural networks for handling heterogeneous data
  - Data (feature vectors, documents, picture
  - Using pre-trained deep neural networks for handling heterogeneous data
- Transformer architectures allowing to adapt deep learning models to new tasks
- Using open source Large Language Models for handling text data
- Machine Learning = AI ?

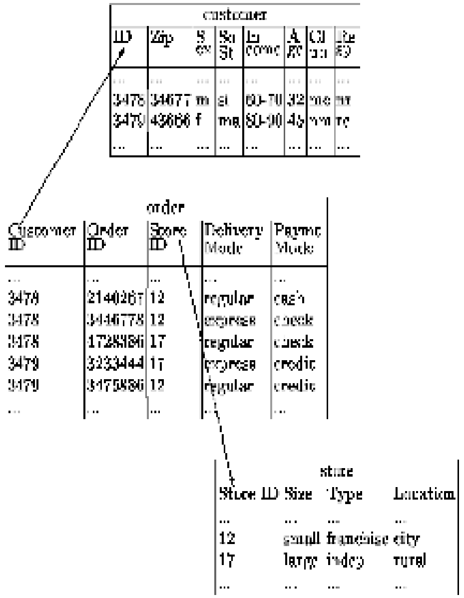
Lesson 1:
Introduction to Data Mining

- Basics of Machine Learning
- Standard learning tasks
- Three generations of machine learning
- Advanced learning tasks

Representation Learning
Relational Data Mining

- **Relational data mining:** Learning from complex relational databases
- **Inductive logic programming:** Learning from complex structured data, e.g. molecules and their biochemical properties

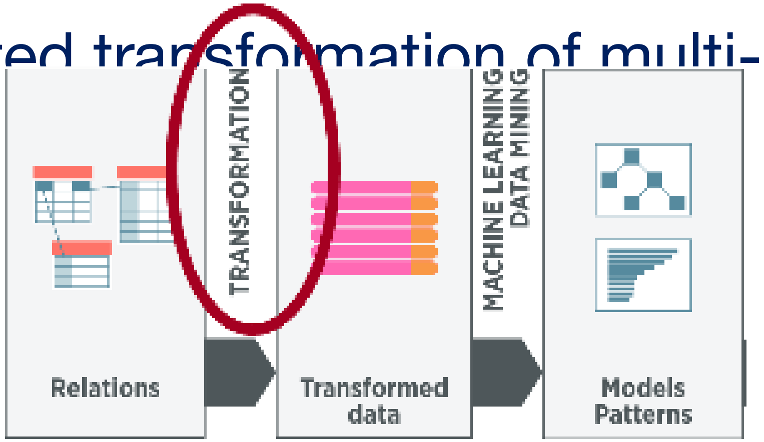




Relational representation of customers, orders and stores.

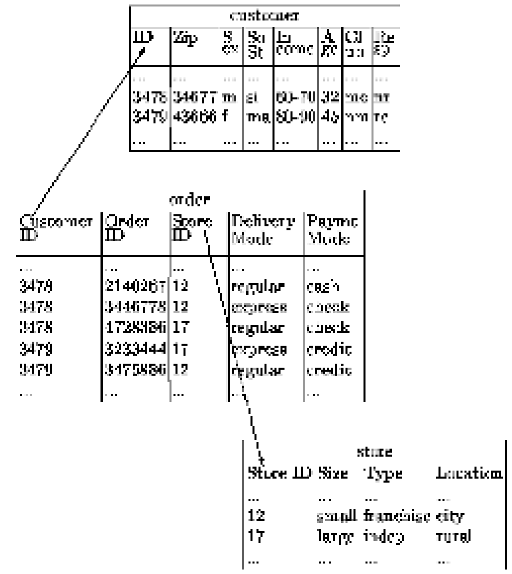
Representation Learning
Relational Data Mining

- Representation learning in a relational learning setting:
  - automated transformation of multi-relational data



- Two main approaches:
  - Traditional approach: Propositionalization of relational databases, heterogeneous information networks, …
  - Recent approach: Embedding of knowledge graphs, network node embeddings, entity embeddings, …

Representation Learning
Relational Data Mining



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

Representation Learning
Relational Data Mining

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	1728386	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

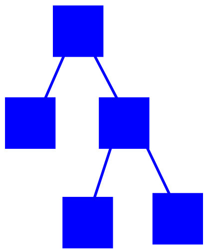
1. construct relational features
2. construct a propositional table

	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

Machine Learning



classification model

Representation Learning
Relational Data Mining

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	1728386	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

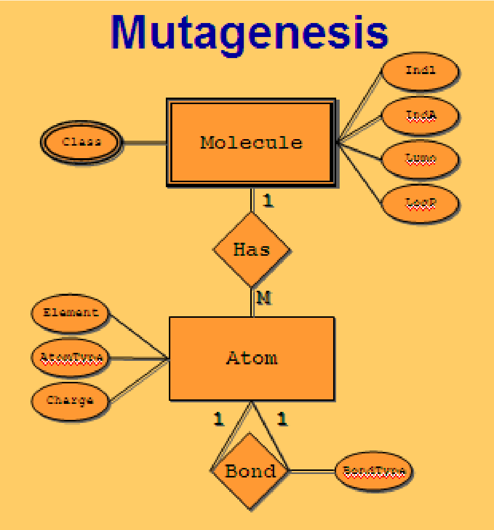
Subgroup discovery

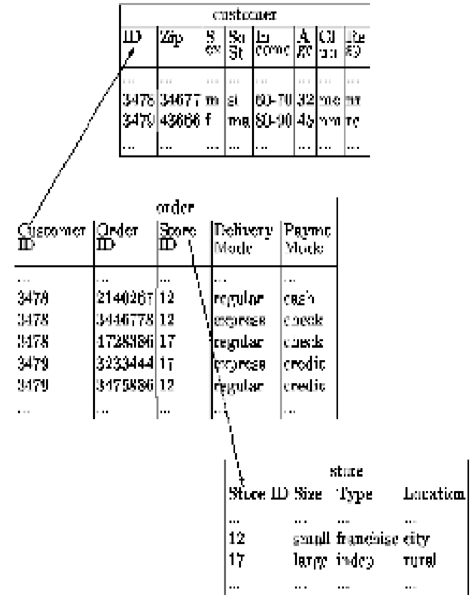
```
target(A) :-
   'Doctor'(A), 'Italy'(A).

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)

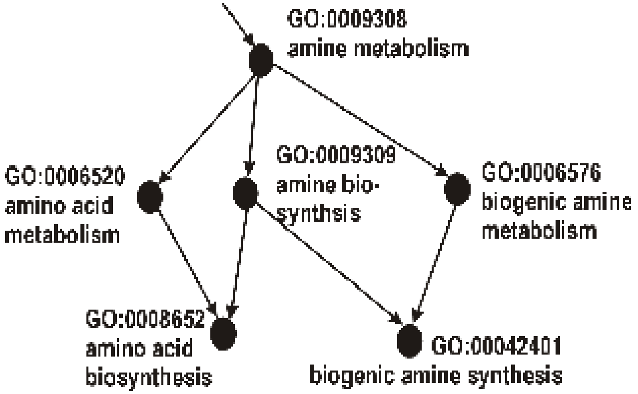
Relational and Semantic Data Mining

- **Relational data mining:** Learning from complex relational databases
- **Inductive logic programming:** Learning from complex structured data, e.g. molecules and their biochemical properties
- **Semantic data mining:** Learning by using domain knowledge in the form of ontologies

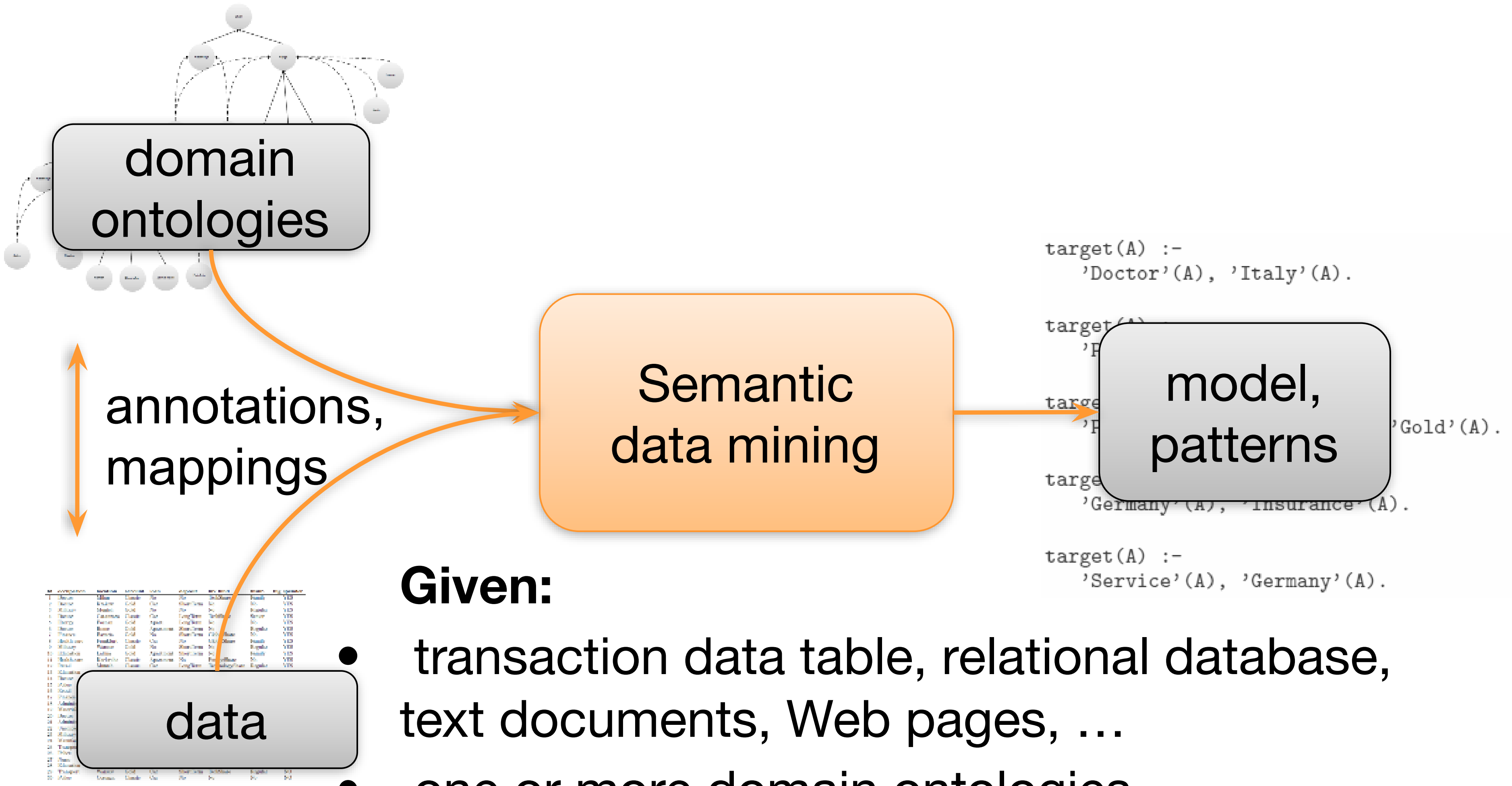




Relational representation of customers, orders and stores.



Semantic Data Mining: Using ontologies as background knowledge in RDM



Given:

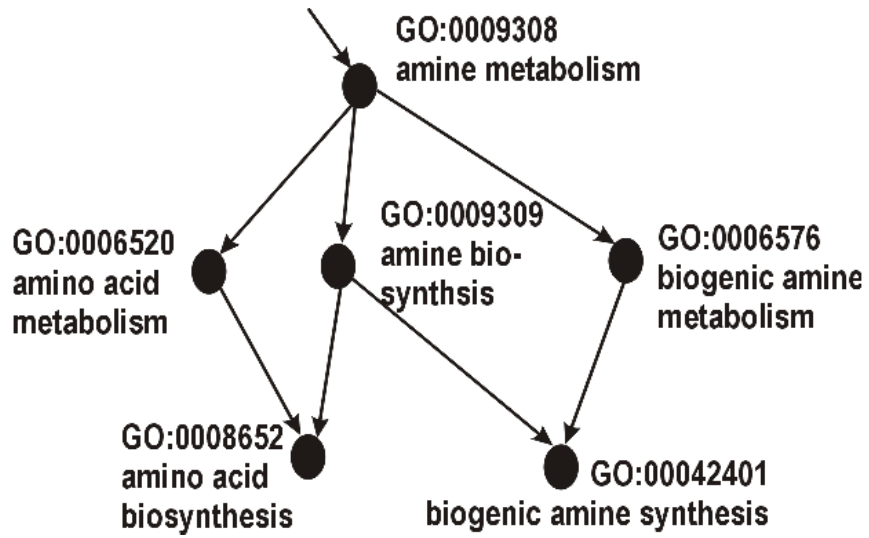
- transaction data table, relational database, text documents, Web pages, …
- one or more domain ontologies

Find: a classification model, a set of patterns

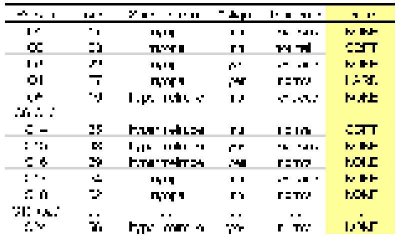
Using domain ontologies

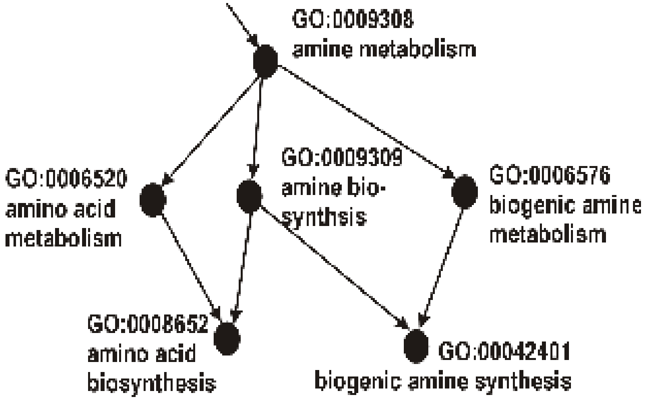
Using domain ontologies as background knowledge, e.g., using the Gene Ontology (GO)

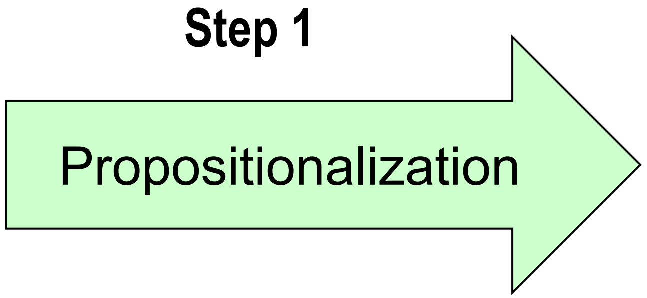
- GO is a database of terms, describing gene sets in terms of their
  - functions
  - processes
  - components
- Genes are annotated to GO terms
- Terms are connected (is_a, part_of)
- Levels represent terms generality



Representation Learning
Semantic Data Mining







	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

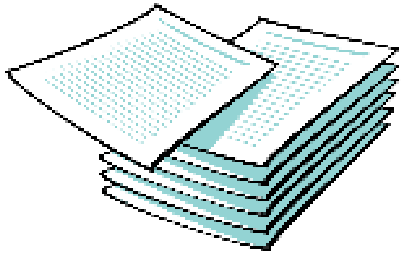
Approach:

- Using relational learning in the SDM context, using a propositionalization approach

Sample application:

- Semantic data mining in a biomedical application by using the Gene Ontology as background knowledge in analyzing microarray data

Text mining: Viewed in propositionalization context: BoW data transformation



Step 1

BoW vector construction

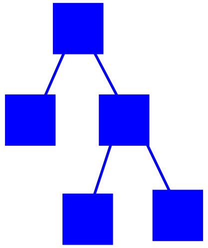
Document	Word1	Word2	...	WordN	Class
d1	1	1	0	1	NO
d2	1	1	0	0	YES
d3	1	1	1	1	NO
d4	1	1	1	0	YES
d5	1	0	0	1	NO
d6-d13	...	...	...	...	...
d14	0	0	0	0	YES
d15	0	0	1	1	NO
d16	0	0	1	0	NO
d17	0	1	0	1	NO
d18	0	1	0	0	NO
d19-d23	...	...	...	...	...
d24	0	0	1	0	NO

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

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

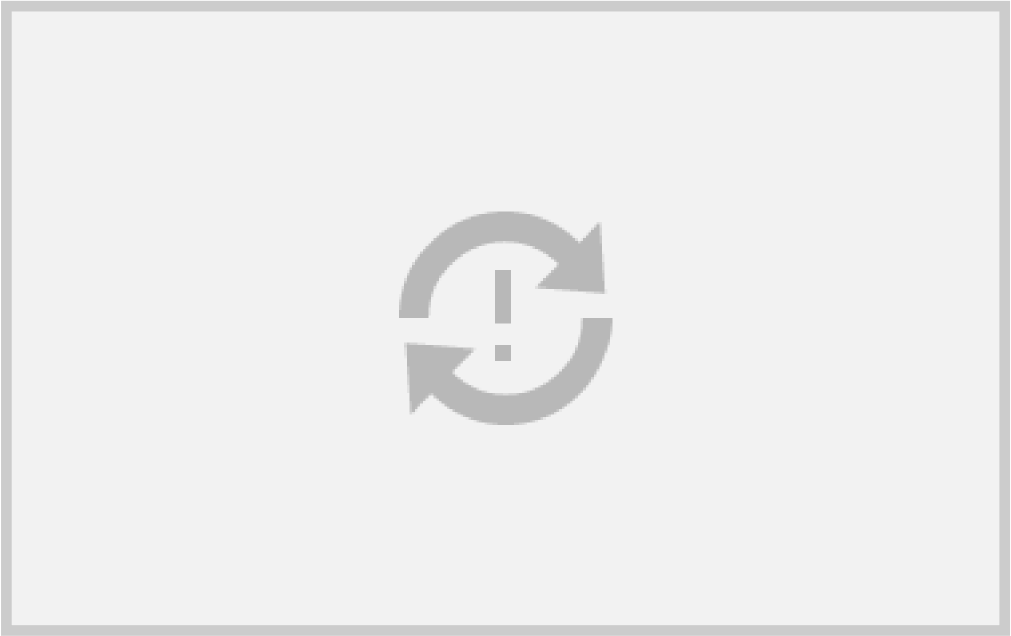
- Similarity between two vectors is estimated by the similarity between their vector representations (cosine of the angle between the two vectors):

$$Similarity(D_1, D_2) = \frac{\sum_i X_{1i} X_{2i}}{\sqrt{\sum_j X_j^2}\sqrt{\sum_k X_k^2}}$$

- Similarity between BoW vectors can be used for document clustering, i. e. for finding natural groups of documents in an unsupervised way (no class labels pre-assigned to documents)

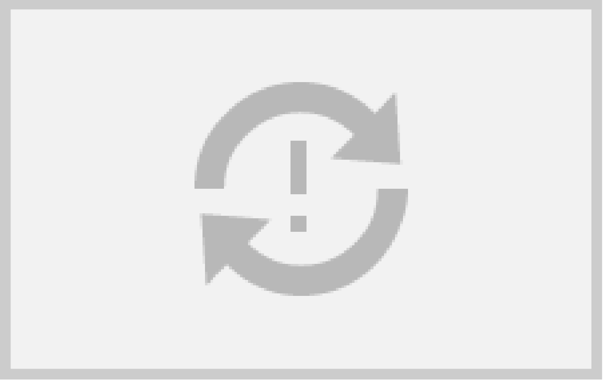
Embeddings-based Data Transformation for Text mining

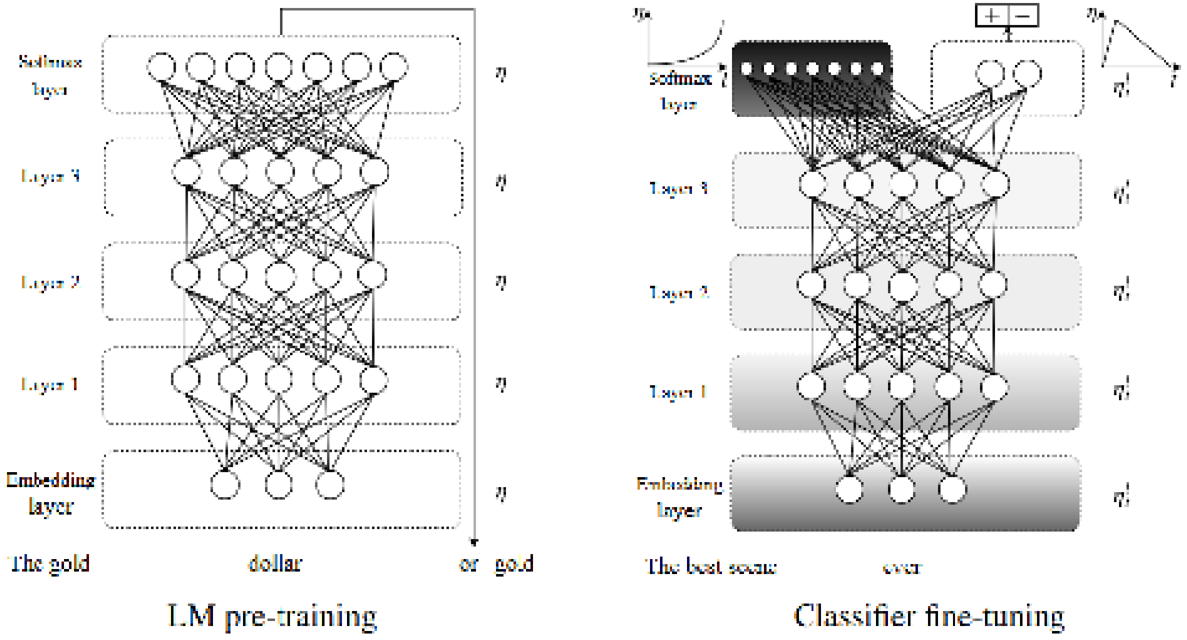
- Corpus embedding, **Document embedding,** Sentence embedding, word embedding (e.g., word2vec)
  - Transforming documents by projecting documents into vectors (rows of a data table)



Embeddings-based Data Transformation for Text mining

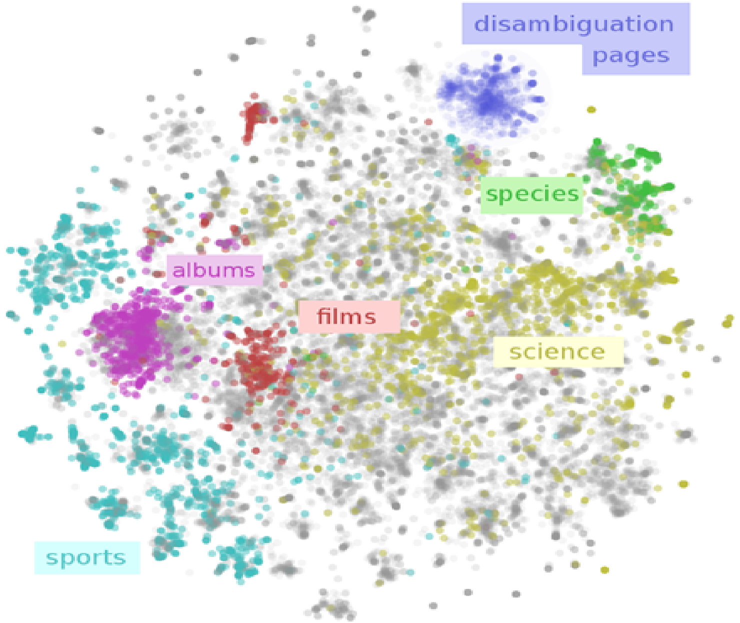
- Corpus embedding, Document embedding, Sentence embedding, **word embedding** (e. g., word2vec)
  - Transforming documents by projecting documents into vectors (rows of a data table)
  - Table values correspond to weights in the embedding layer of a neural network

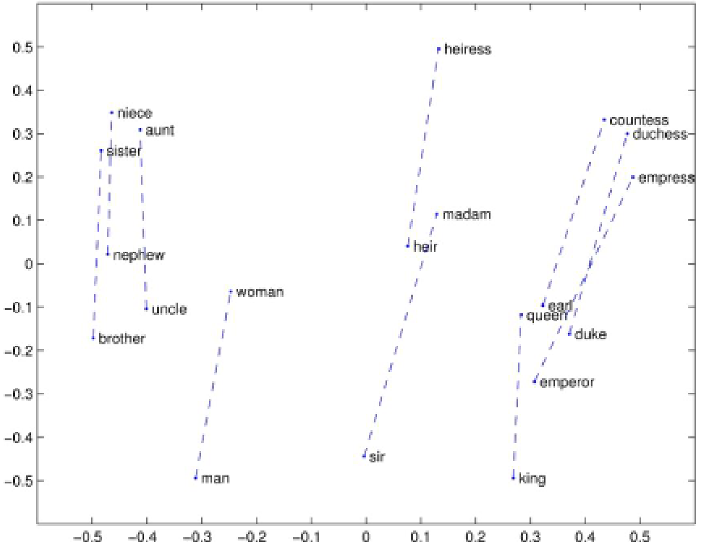




Embedding-based Data Transformation for Text mining

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





Data Mining Lesson 1:
Summary and Take away messages

- Motivation for studying Machine Learning
  - ML is highly relevant, as motivated by two epidemiology spreading case studies
  - Course outline should motivate for studying this modern ML approach to become a skilled data scientist
- Introduction to Machine Learning
  - ML basics and illustrative examples were presented for elementary classification and regression learning tasks
  - Three generations of machine learning and data mining methods were outlined
- Representation Learning
  - Representation learning is a highly relevant contemporary ML problem
  - ML basics and illustrative examples were presented for advanced relational, semantic and text mining tasks

Selected literature

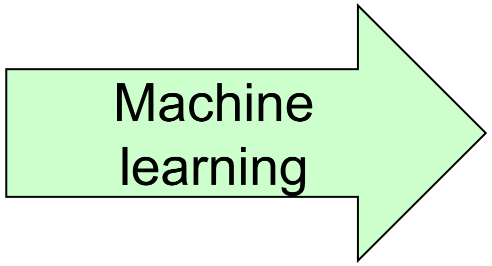
- James G, Witten D, Hastie T and Tibshirani R (1st Edition 2013, 2nd Edition 2021) An Introduction to Statistical Learning - with Applications in R. Springer, New York. Available at https://statlearning.com/. Chapters 1 and 2.
- Bramer M (2007) Principles of Data Mining. Springer, Berlin. DOI:10.1007/978-1-84628-766-4. An introductory textbook for refreshing your knowledge on basics of data mining. The first edition of the textbook is also available at ResearchGate, https://www.researchgate.net/publication/220688376_Principles_of_Data_Mining
- Lavrač N, Podpečan V and Robnik-Šikonja M (2021) Representation Learning: Propositionalization and Embeddings. Springer, Berlin. Chapters 1 and 2.

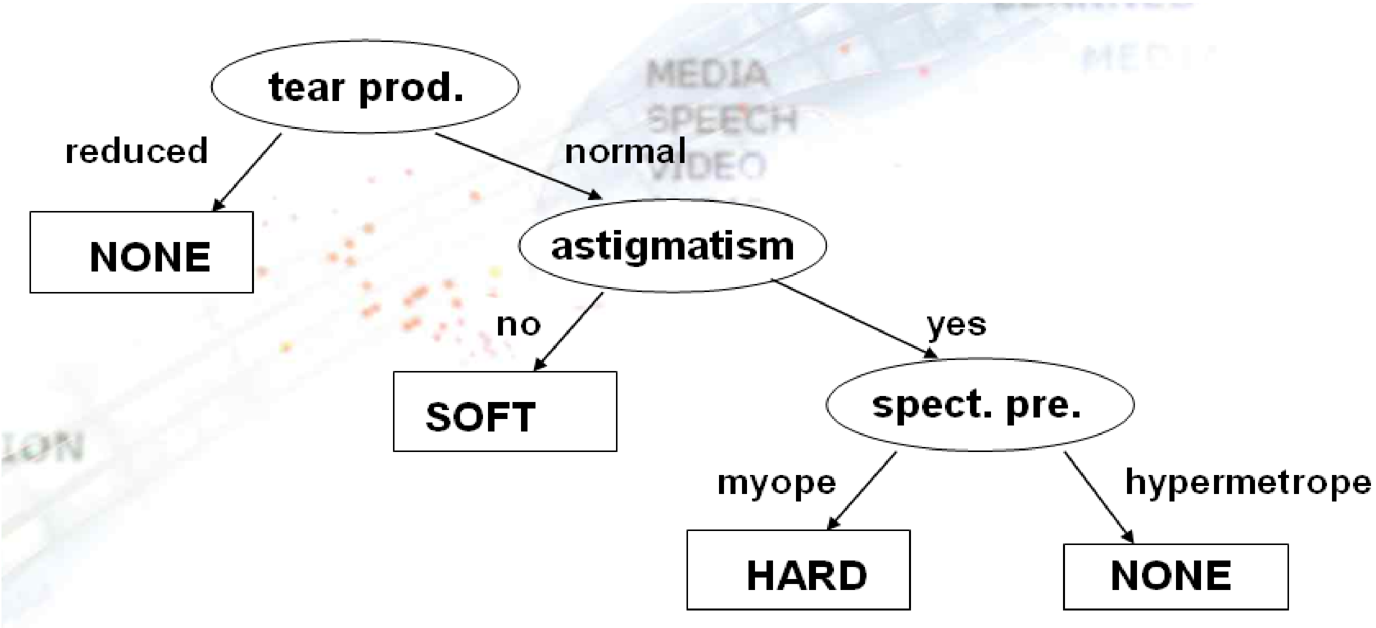
Lesson 2
Decision tree learning

- Basic decision tree learning algorithm
- Classifier evaluation and decision tree pruning
- Selected decision tree learning algorithms
- Regression tree learning

Decision tree learning: an illustrative example

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





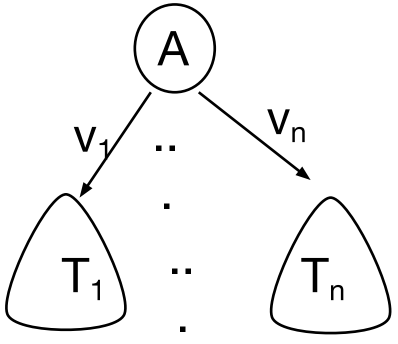
Predictive DM task: Basic notions

- Data are objects, characterized with attributes A_i and class-labels C_j
- Objects (data instances, training examples) are described with attribute values
- Attributes can be discrete, nominal or numeric
- Classes can be discrete (binary classification) or nominal (multi-class learning) or numeric (regression)
- Classification learning task is to induce a model capable to predict the class-label for a new (unclassified) instance

TDIDT - Decision tree learning algorithm

Elementary decision tree learning algorithm ID3 (Quinlan 1979)

- create the root node of the tree
- if all examples from S belong to the same class C_j
  - then label the root with C_j
- else
  - select the 'most informative' attribute A with values v_1, v_2, … v_n
  - divide training set S into S_1,… , S_n according to values v_1, v_2, … v_n
  - recursively build sub-trees T_1,… , T_n for S_1,… , S_n



Decision tree search heuristics

- Central choice in decision tree algorithms: Which attribute to test at each node in the tree ? The attribute that is most useful for classifying examples.
- Define a statistical property, called **information gain**, measuring how well a given attribute separates the training examples w.r.t their target classification.
- First define a measure commonly used in information theory, called **entropy**, to characterize the (im)purity of an arbitrary collection of examples.

Entropy

- **Entropy E(S)** – measure of impurity of training set S

- In concept learning (**binary classification**) problems, with training set S labeled by two classes C_+ and C_-

$$E(S) = -p_+ \log_2 p_+ - p_- \log_2 p_-$$

p_+ - prior probability of class **C_+** (relative frequency of **C_+** in **S**)
p_- - prior probability of class **C_-**

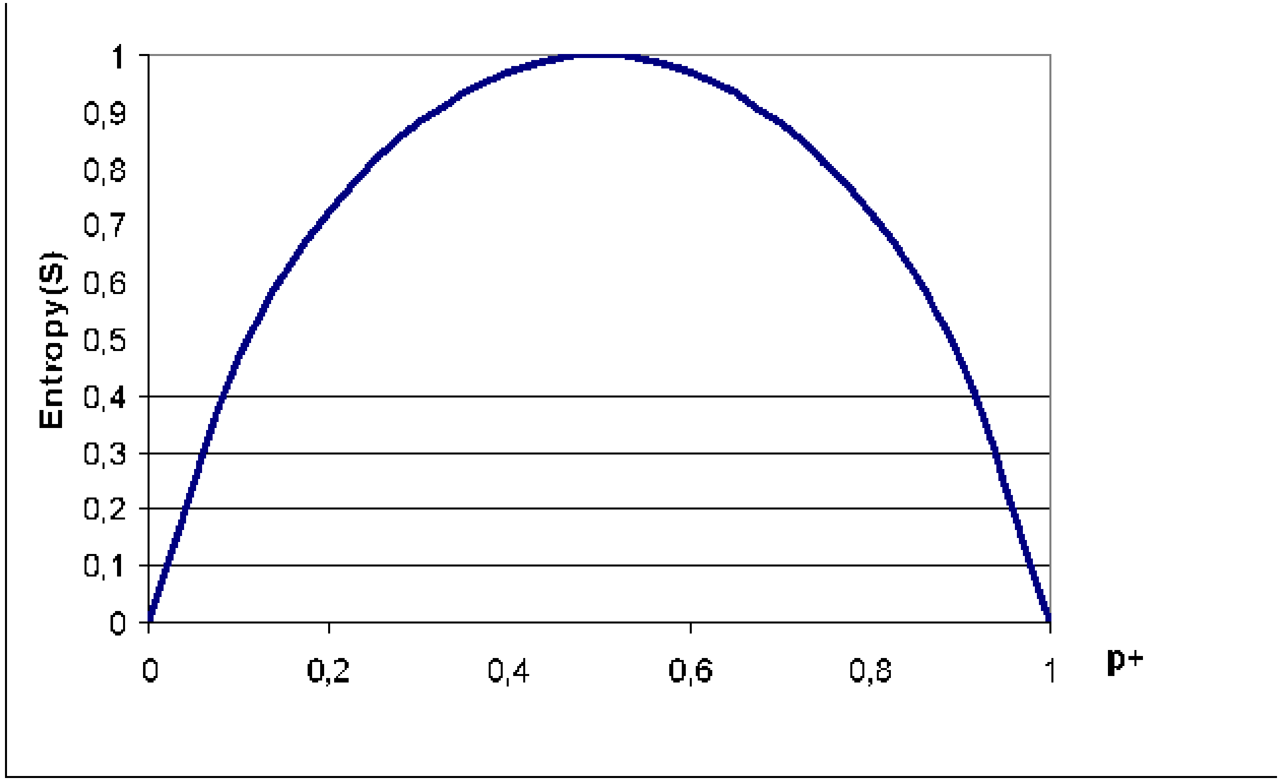
- In **multi-class** learning problems, with training set S labeled by N classes $C_1, C_2, \dots, C_N$

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

p_c - prior probability of class **C_c** (relative frequency of **C_c** in **S**)

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



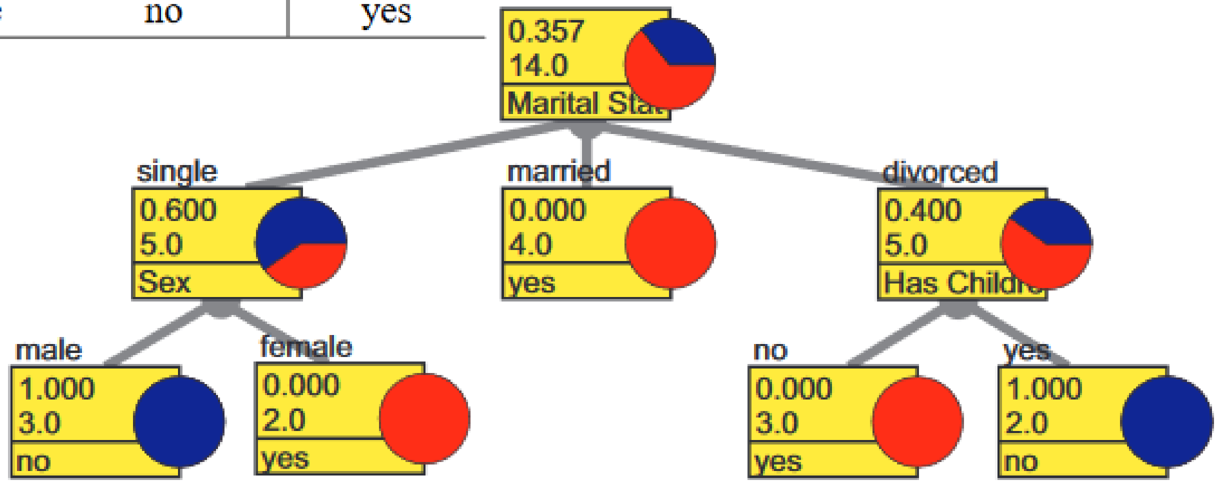
Entropy – why ?

- **Entropy E(S) =** expected amount of information (in bits) needed to assign a class to a randomly drawn object in S (under the optimal, shortest-length code)
- Why ?
- Information theory: optimal length code assigns $-\log_2 p$ bits to a message having probability p
- So, in binary classification problems, the expected number of bits to encode + or – of a random member of S is:

$$p_+ (-\log_2 p_+) + p_- (-\log_2 p_-) = -p_+ \log_2 p_+ - p_- \log_2 p_-$$

Binary classification problem: 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



Entropy – example calculation

- Training set S: 14 examples (9 pos., 5 neg.)
- Notation: S = [9+, 5-]
- $E(S) = - p_+ \log_2 p_+ - p_- \log_2 p_-$
- Computing entropy, if probability is estimated by relative frequency

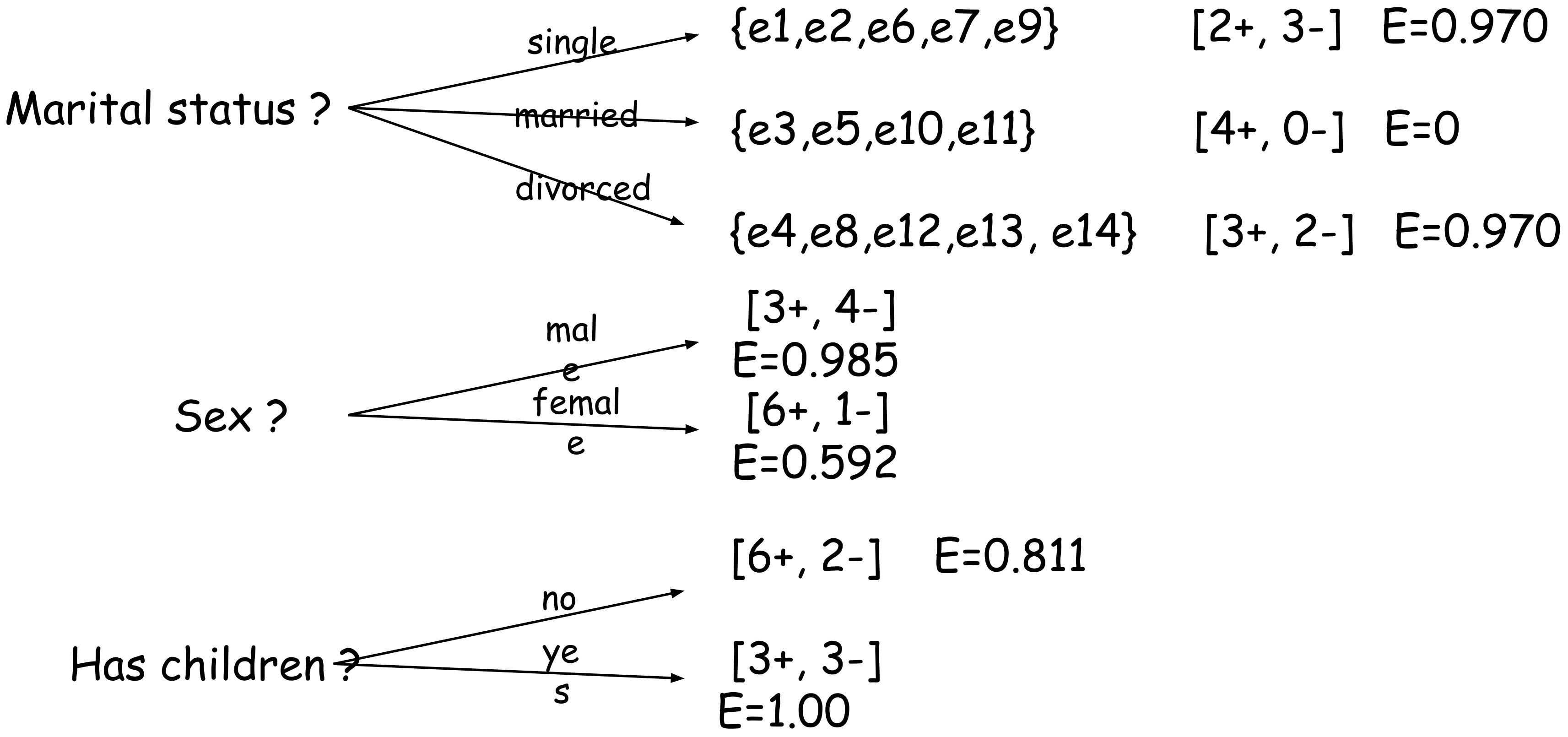
$$E(S) = -\left(\frac{|S_+|}{|S|} \cdot \log \frac{|S_+|}{|S|}\right) - \left(\frac{|S_-|}{|S|} \cdot \log \frac{|S_-|}{|S|}\right)$$

- $E([9+,5-]) = - (9/14) \log_2(9/14) - (5/14) \log_2(5/14)$

  $= 0.940$

Survey data: Entropy

- $E(S) = - p_+ \log_2 p_+ - p_- \log_2 p_-$
- $E([9+,5-]) = - (9/14) \log_2(9/14) - (5/14) \log_2(5/14) = 0.940$



Information gain search heuristic

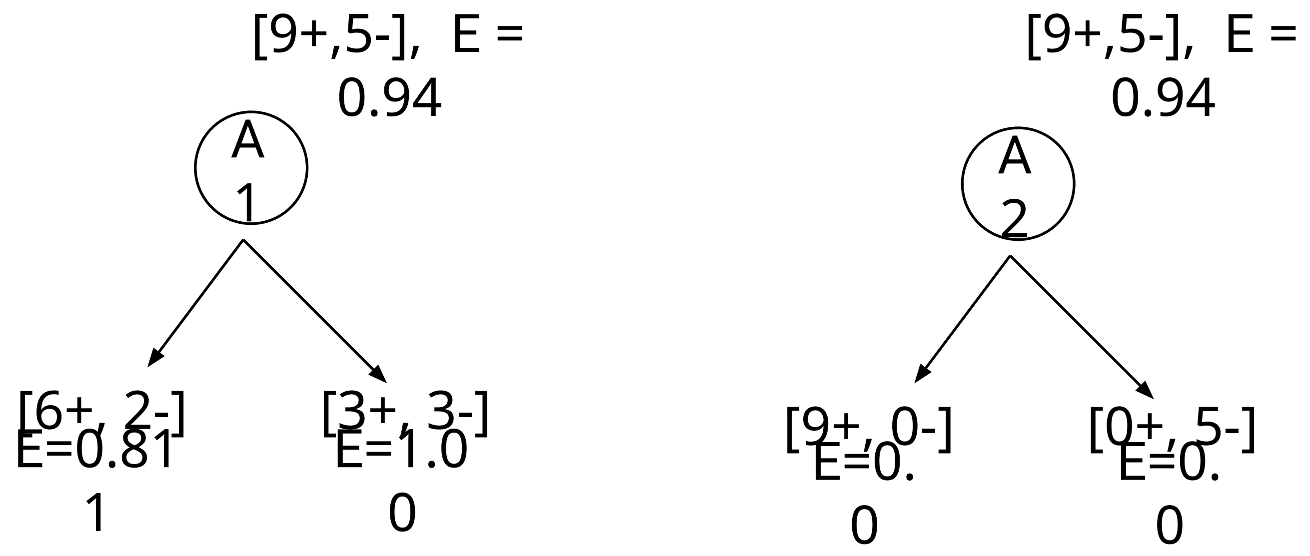
- **Information gain** measure is aimed to minimize the number of tests needed for the classification of a new object
- **Gain(S,A)** – expected reduction in entropy of S due to sorting on A

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

- **Most informative** attribute: **max Gain(S,A)**

Information gain search heuristic

- Which attribute is more informative, A1 or A2 ?

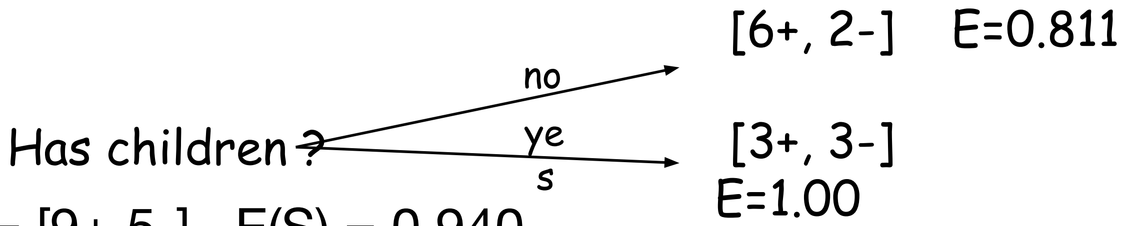


- Gain(S,A1) = 0.94 – (8/14 x 0.811 + 6/14 x 1.00) = 0.048
- Gain(S,A2) = 0.94 – 0 = 0.94 A2 has max Gain

Survey data: Information gain

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

- Values(Has children) = {no, yes}



  - S = [9+,5-], E(S) = 0.940
  - S_{no} = [6+,2-], E(S_{no}) = 0.811
  - S_{yes} = [3+,3-], E(S_{yes}) = 1.0
  - **Gain(S, Has children) =** E(S) - (8/14)E(S_{no}) - (6/14)E(S_{yes}) = 0.940 - (8/14)x0.811 - (6/14)x1.0=**0.048**

Survey data: Information gain

- **Which attribute is the best?**
  - Gain(S, Marital status)=0.246 *MAX !*
  - Gain(S, Sex)=0.151
  - Gain(S, Has children)=0.048
  - Gain(S, Education)=0.029

Survey data: Information gain

Marital status ?

singl e → {D4,D5,D6,D10,D14} [3+, 2-] E=0.970 **???**

married → {D3,D7,D12,D13} [4+, 0-] E = 0 ***OK - assign class Yes***

divorced → {D1,D2,D8,D9,D11} [2+, 3-] E=0.970 **???**

- Which attribute should be tested here?
  - Gain($S_{divorced}$, Sex) = 0.97-(3/5)0-(2/5)0 = 0.970 ***MAX !***
  - Gain($S_{divorced}$,Has children) = 0.97-(2/5)0-(2/5)1-(1/5)0 = 0.570
  - Gain($S_{divorced}$,Education) = 0.97-(2/5)1-(3/5)0.918 = 0.019

Alternative probability estimates

- **Relative frequency** :
  - Computed as |S+ | / |S|
  - problems with small samples

[6+,1-] (7) = 6/7

[2+,0-] (2) = 2/2 = 1

- **Laplace estimate** :
  - assumes uniform prior distribution of k classes
  - For k=2, Computed as (|S+ |+1) / (|S|+2)

[6+,1-] (7) = (6+1) / (7+2) = 7/9

[2+,0-] (2) = (2+1) / (2+2) = 3/4

Heuristic search in ID3

- **Search bias:** Search the space of decision trees from simplest to increasingly complex (top-down greedy search, no backtracking, prefer small trees)
- **Search heuristics:** At a node, select the attribute that is most useful for classifying examples, split the node accordingly
- **Stopping criteria:** A node becomes a leaf
  - if all examples belong to same class C_j, label the leaf with C_j
  - if all attributes were used, label the leaf with the most common value C_k of examples in the node
- **Extension to ID3:** handling noise - tree pruning

Decision tree learning

- Basic decision tree learning algorithm
- Classifier evaluation and decision tree pruning
- Selected decision tree learning algorithms
- Regression tree learning

Classifier evaluation

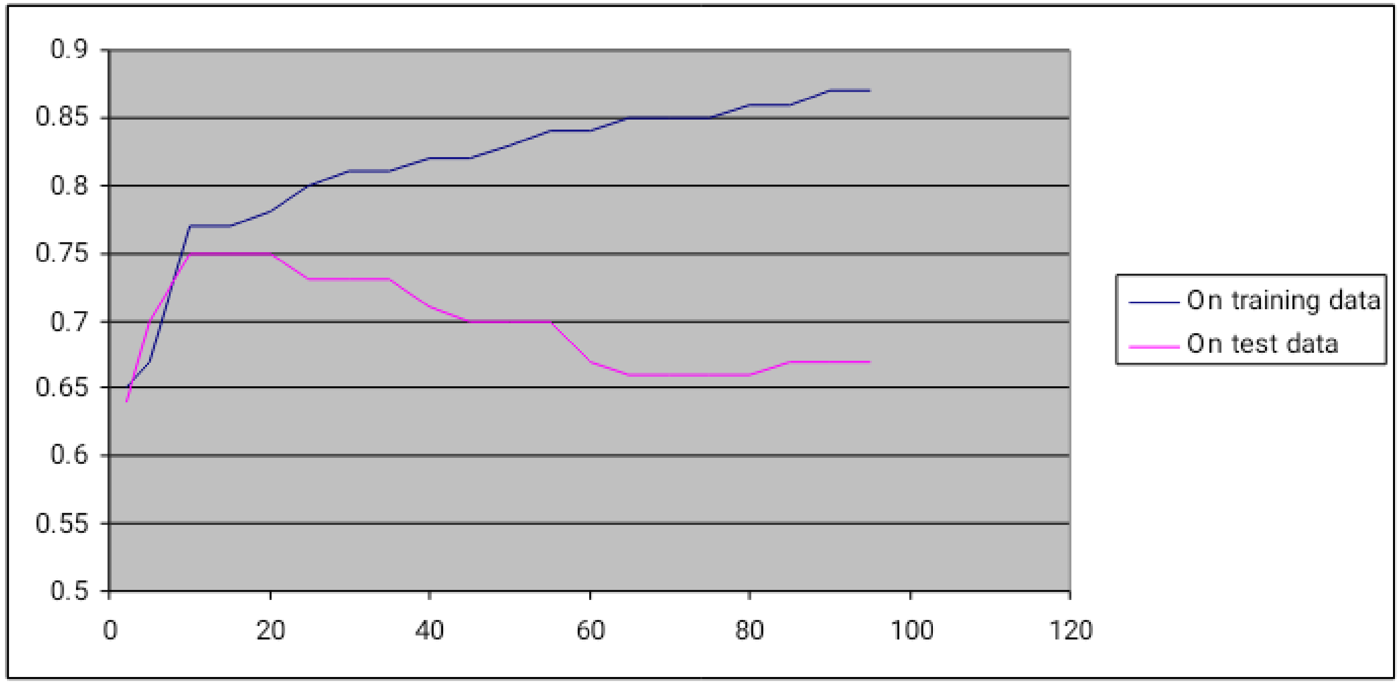
- **Evaluation of learned models**
  - discovery of new patterns, new knowledge
  - explainability and compactness - XAI
  - information contents (information score) - significance
  - classification of new objects – accuracy
- **Evaluating the accuracy of learned models**
  - Accuracy, Error = 1 – Accuracy
  - high accuracy on testing examples = high percentage of correctly classified unseen instances – high predictive power
  - high accuracy on training examples – possible data overfitting

Classifier evaluation

- **Evaluation methodology**
  - split the example set into training set (e.g. 70%) to induce a concept, and test set (e.g. 30%) to test its accuracy
  - more elaborate strategies: 10-fold cross validation, leave-one-out, ...
- **N-fold cross-validation method for accuracy estimation of classifiers**
  - Partition set D into n disjoint, almost equally-sized folds T_i where $U_i T_i = D$
  - **for** i = 1, ..., n **do**
    - form a training set out of n-1 folds: Di = D\T_i
    - induce classifier H_i from examples in Di
    - use fold T_i for testing the accuracy of H_i
  - Estimate the accuracy of the classifier by averaging accuracies over 10 folds T_i

Overfitting and accuracy

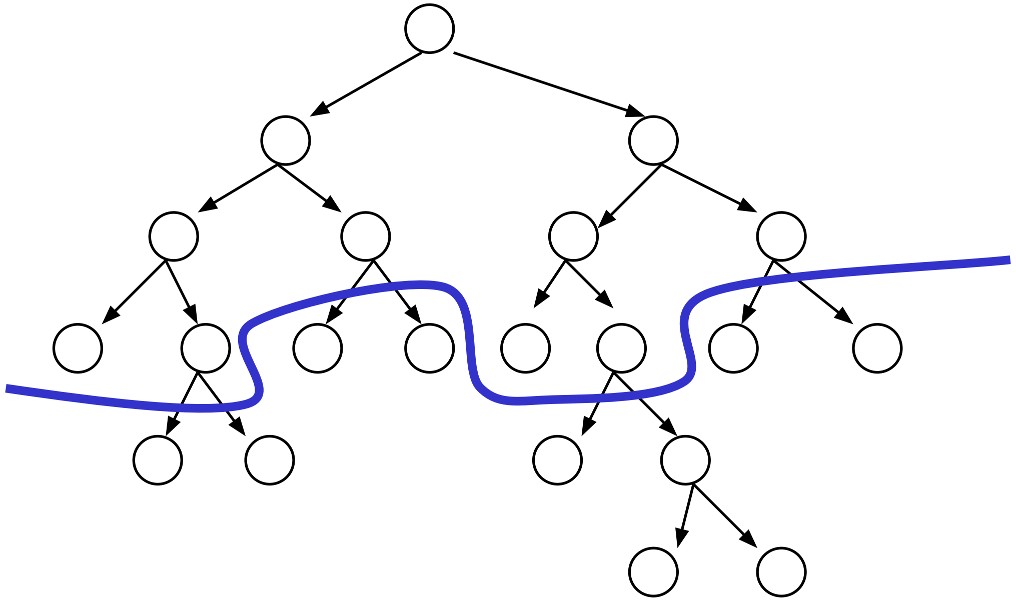
- Typical relation between tree size and accuracy



- Question: how to prune optimally?

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



Handling noise – Tree pruning

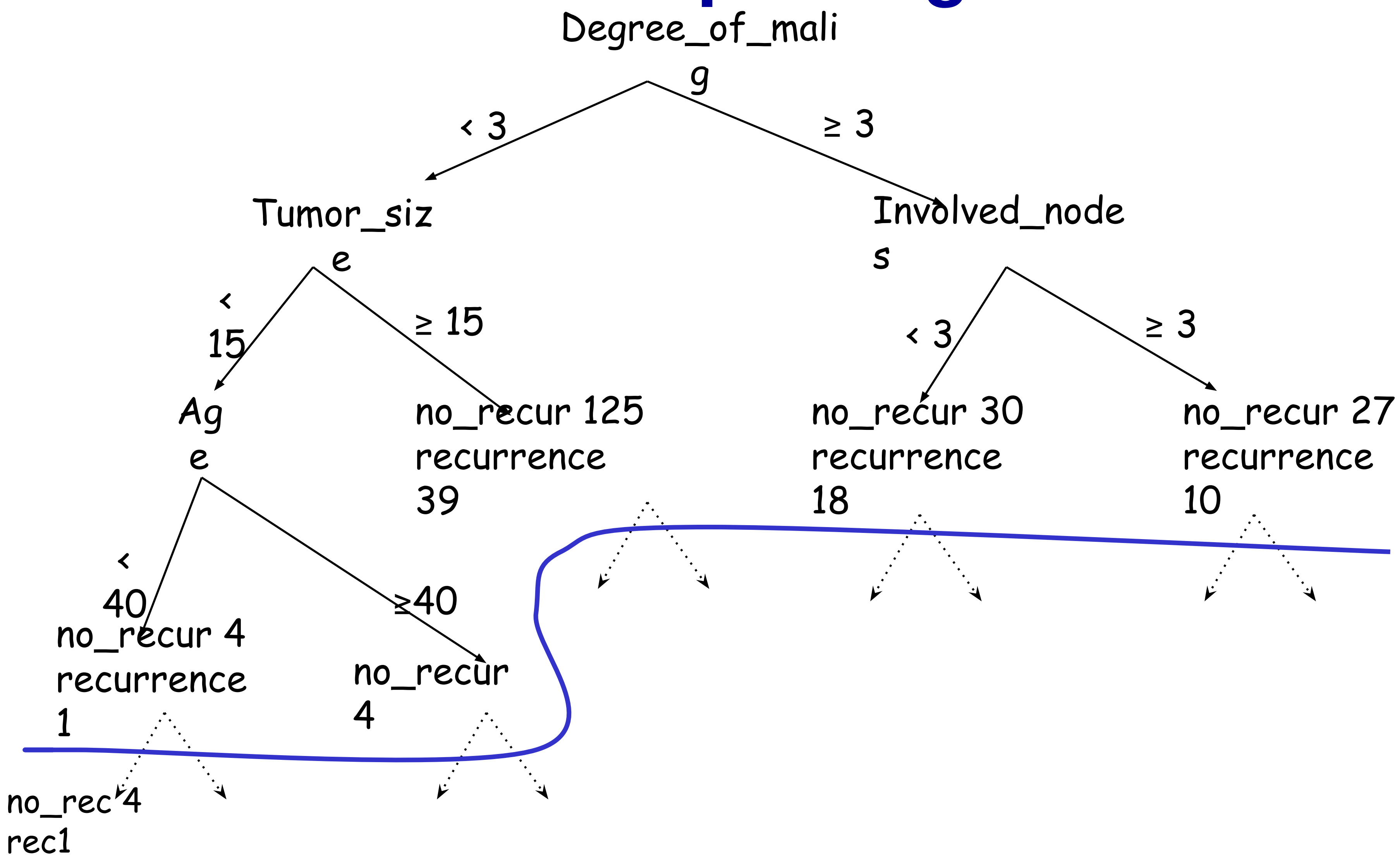
Sources of imperfection

1. Random errors (noise) in training examples
   - erroneous attribute values
   - erroneous classification
2. Too sparse training examples (incompleteness)
3. Inappropriate/insufficient set of attributes (inexactness)
4. Missing attribute values in training examples

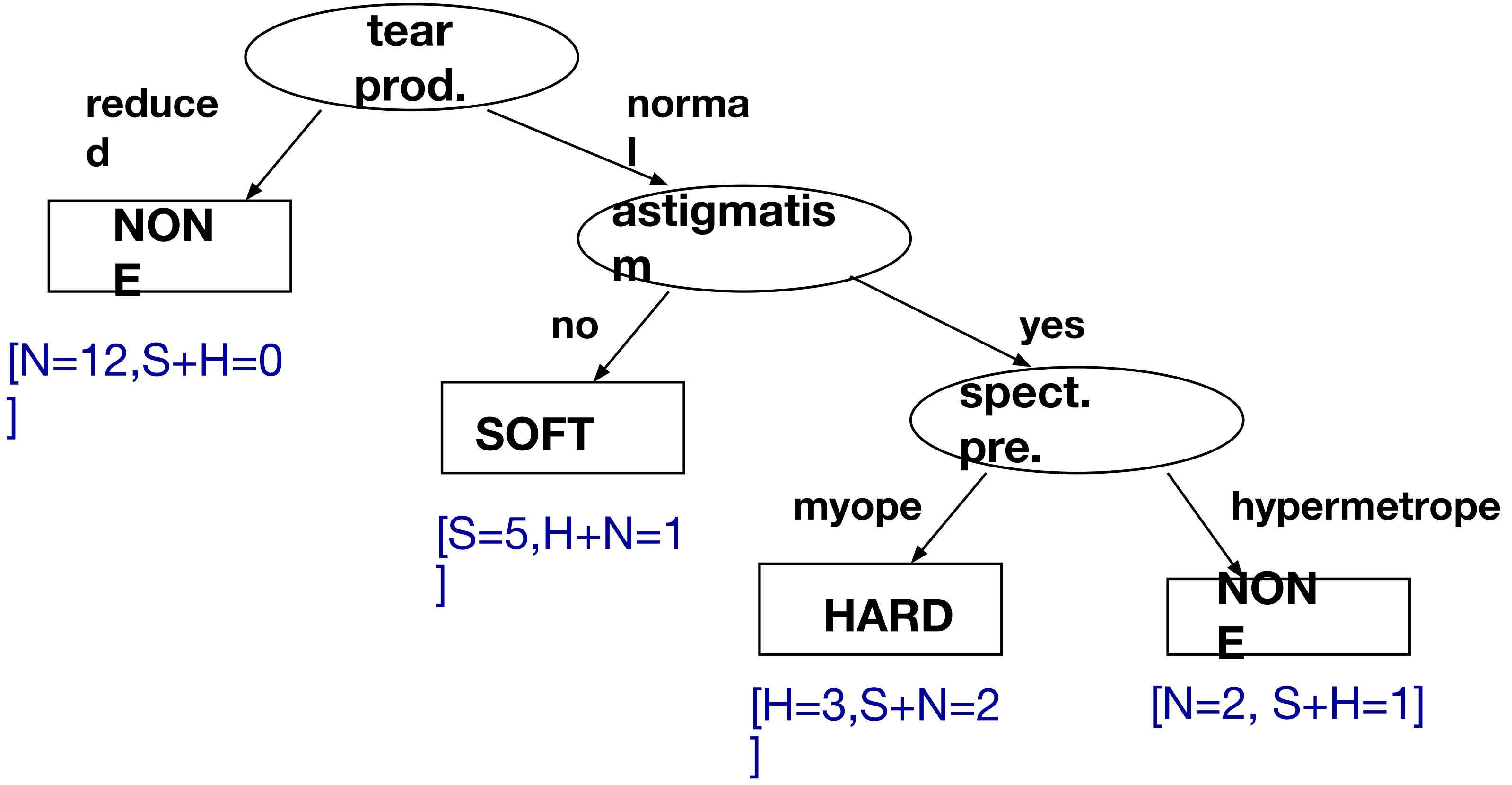
Handling noise – Tree pruning

- Handling imperfect data
  - handling imperfections of type 1-3
    - pre-pruning (stopping criteria)
    - post-pruning / rule truncation
  - handling missing values
- Pruning avoids perfectly fitting noisy data: relaxing the completeness (fitting all +) and consistency (not fitting all -) criteria in ID3

Prediction of breast cancer recurrence: Tree pruning



Pruned decision tree for contact lenses recommendation



Decision tree learning

- Basic decision tree learning algorithm
- Classifier evaluation and decision tree pruning
- Selected decision tree learning algorithms
- Regression tree learning

Selected decision/regression tree learners

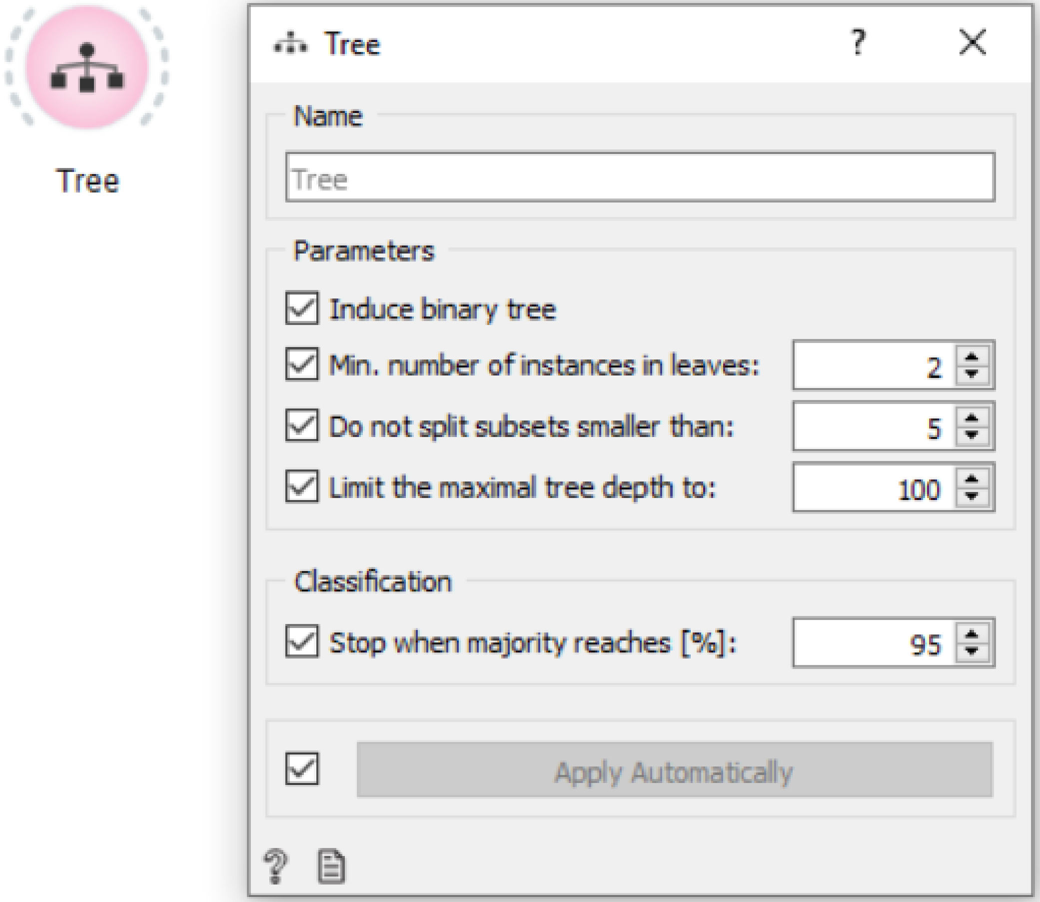
- Decision tree learners
  - ID3 (Quinlan 1979)
  - CART (Breiman et al. 1984)
  - Assistant (Cestnik et al. 1987)
  - C4.5 (Quinlan 1993), C5 (See5, Quinlan)
  - J48 (available in WEKA), Tree (in Orange)
- Regression tree learners, model tree learners
  - M5, M5P (implemented in WEKA), Tree (in Orange)

Appropriate problems for decision tree learning

- Classification problems: classify an instance into one of a discrete set of possible categories (medical diagnosis, classifying loan applicants, …)
- Characteristics:
  - instances described by attribute-value pairs
    (discrete or real-valued attributes)
  - target function has discrete output values
    (boolean or multi-valued, if real-valued then regression trees)
  - disjunctive hypothesis may be required
  - training data may be noisy
    (classification errors and/or errors in attribute values)
  - training data may contain missing attribute values

Selected decision tree learners

- Decision tree learners: Tree (in Orange)



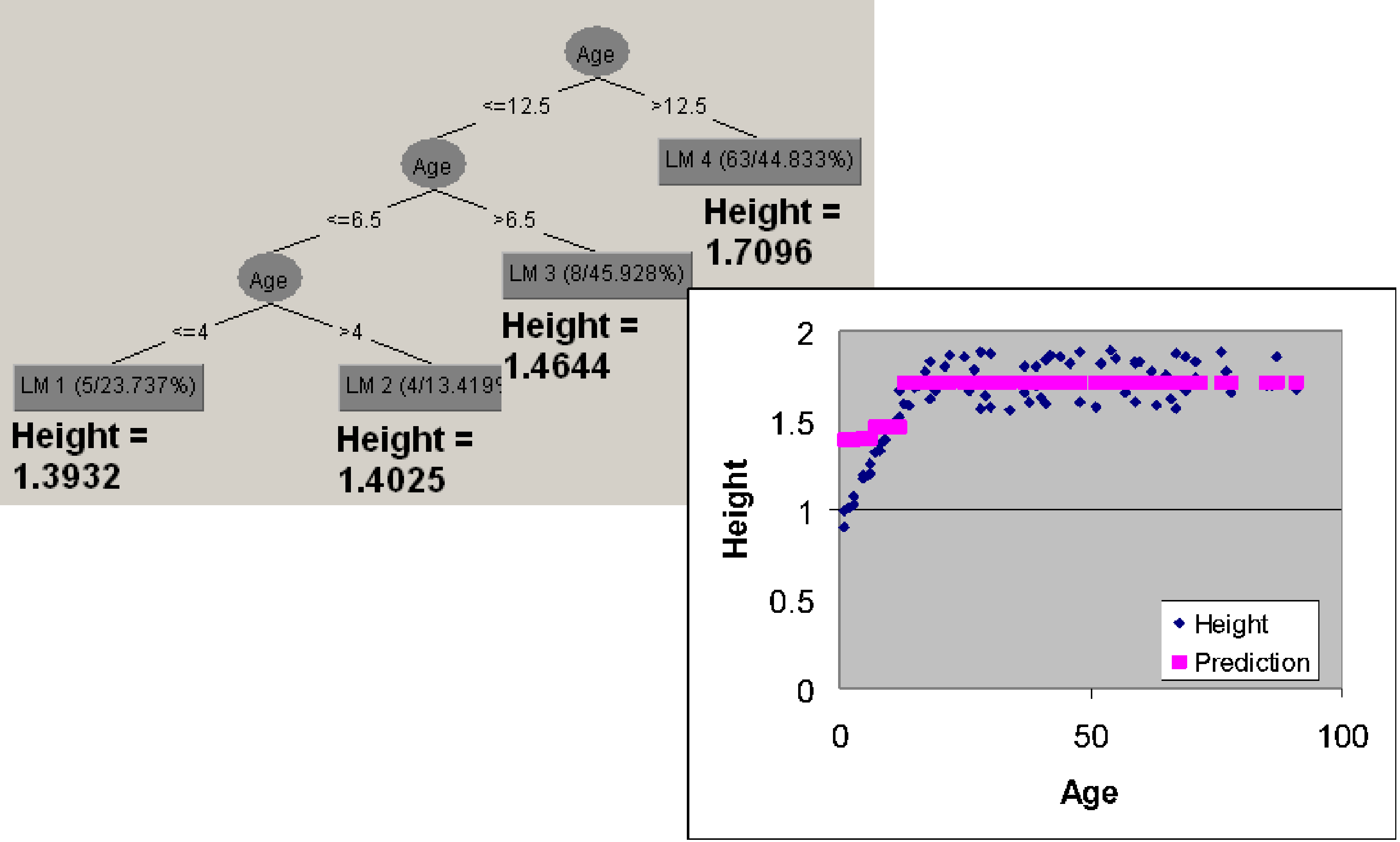
Decision tree learning

- Basic decision tree learning algorithm
- Classifier evaluation and decision tree pruning
- Selected decision tree learning algorithms
- Regression tree learning

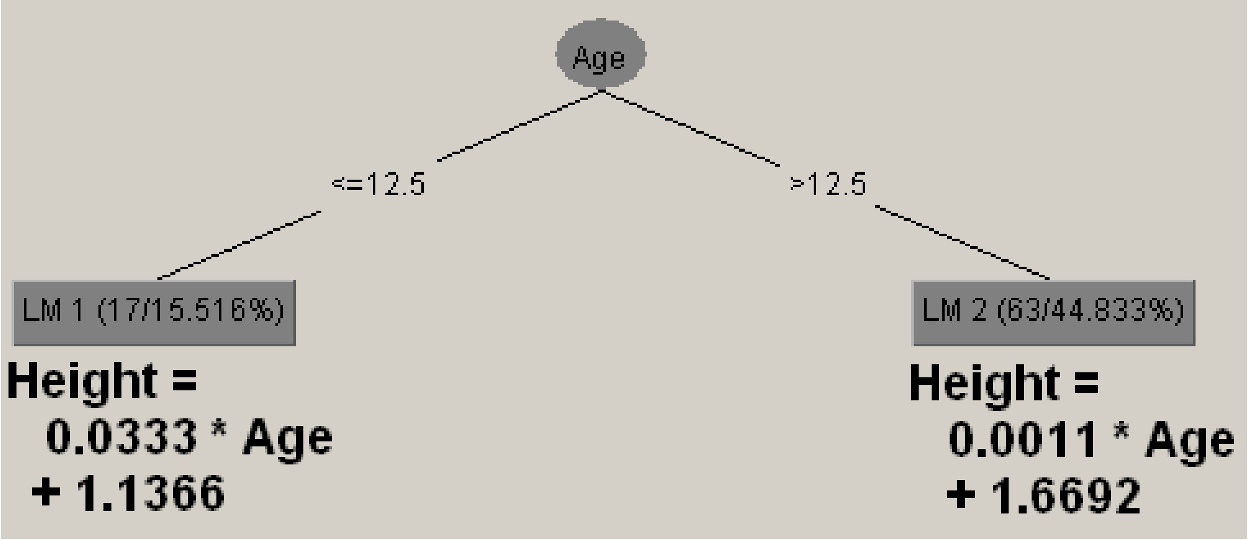
Regression tree learning

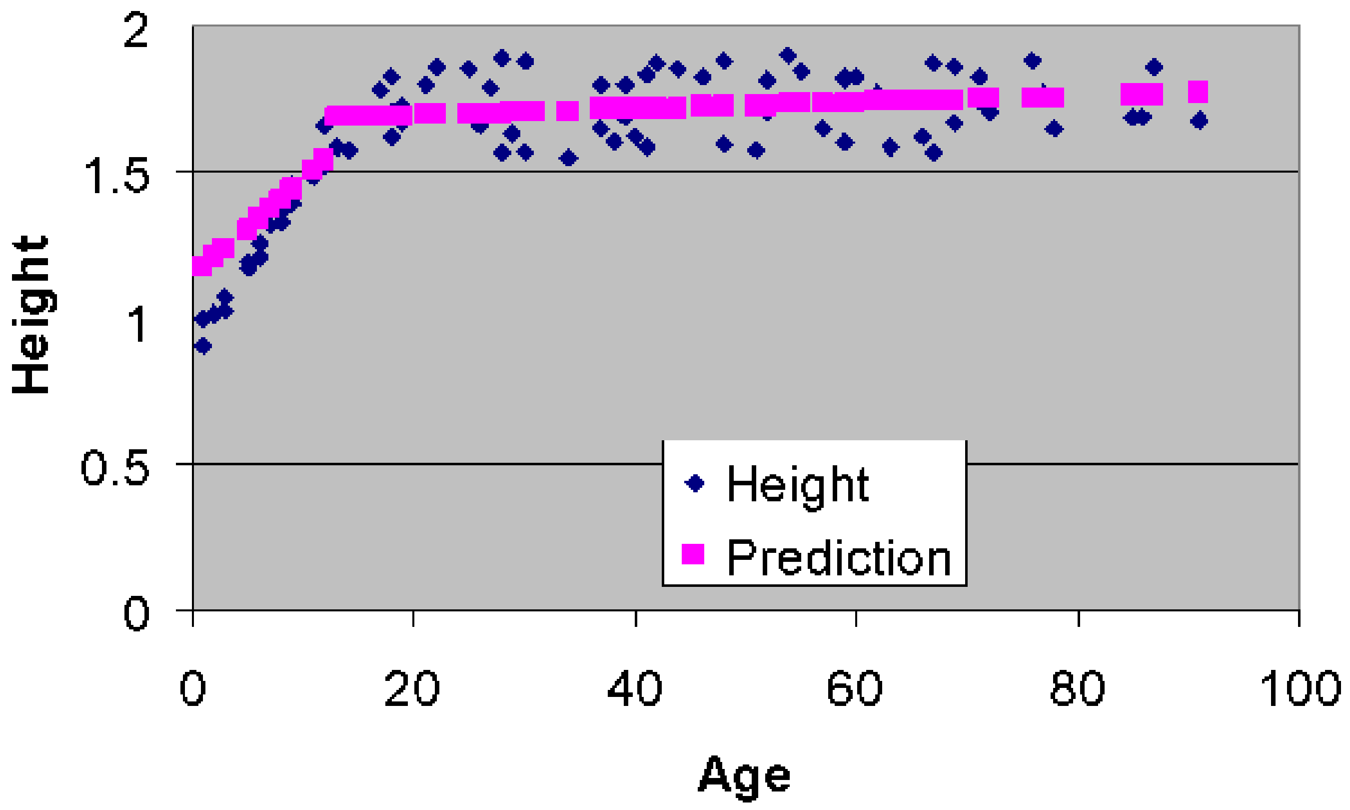
- Estimation or regression task: given objects described with attribute values, induce a model to predict the numeric class value
- Data are objects, characterized with attributes (discrete or continuous), classes of objects are continuous (numeric)
- Regression tree learners, model tree learners:
  - M5
  - M5P (implemented in WEKA)
  - Tree (in Orange)

Regression tree

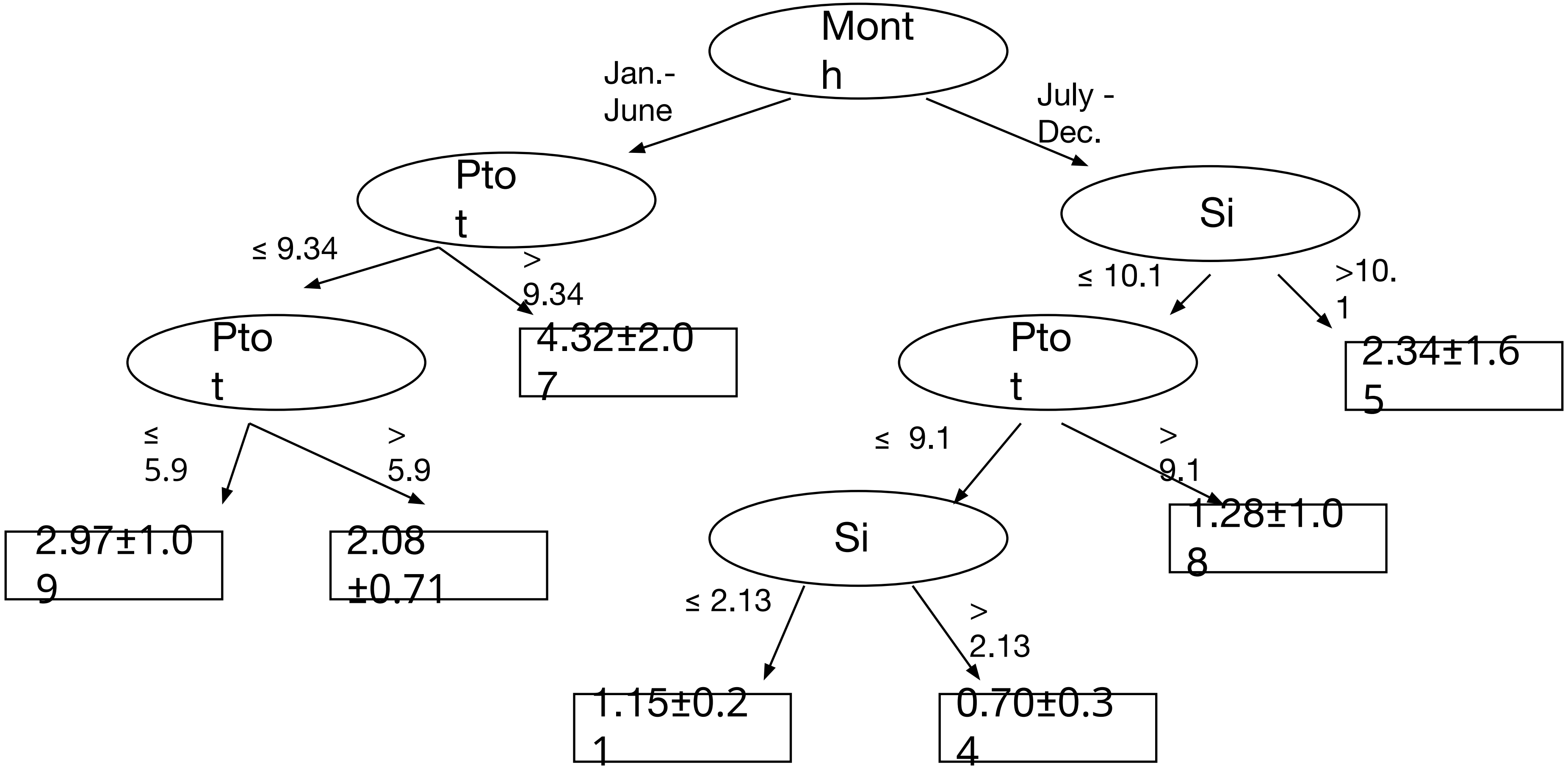


Model tree


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


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

Predicting algal biomass: regression tree



Regression learners:
Which predictor is the best?

Age	Height	Baseline	Linear regression	Regression tree	Model tree	kNN
2	0.85	1.63	1.43	1.39	1.20	1.01
10	1.4	1.63	1.47	1.46	1.47	1.51
35	1.7	1.63	1.61	1.71	1.71	1.67
70	1.6	1.63	1.81	1.71	1.75	1.81

Regression	Classification
Data: attribute-value description	
Target variable: Continuous	**Target variable**: Categorical (nominal)
Evaluation: cross validation, separate test set, …	
Error: MSE, MAE, RMSE, …	**Error**: 1-accuracy
Algorithms: Linear regression, regression trees, …	**Algorithms**: Decision trees, Naïve Bayes, …
Baseline predictor: Mean of the target variable	**Baseline predictor**: Majority class

Lesson 2

Summary and Take away messages

- Decision tree learning
  - Addresses classification problems
  - Algorithms use search heuristics to search the space of possible trees in a top-down manner
  - Training data may be noisy - tree pruning help dealing with noisy data to improve predictive accuracy on new, unlabeled data
- Regression tree learning
  - Addresses predictive modeling from numeric data
  - Advanced regression tree and model tree learners exist
- Notice different evaluation criteria for classification and regression

Lesson 3: Rule Learning

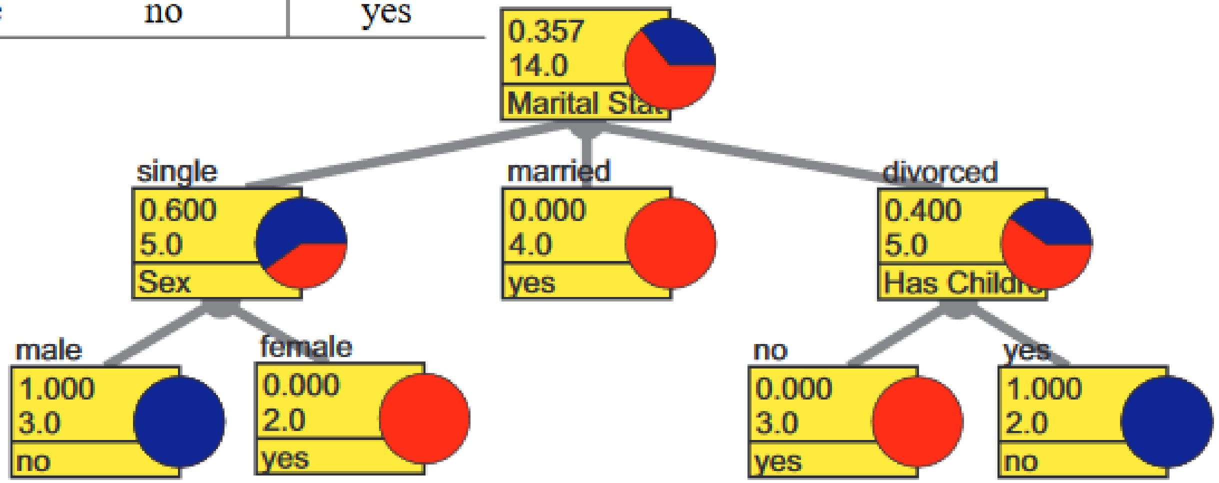
- Transforming decision trees to rules
- Classification rule learning algorithm
  - Covering algorithm
  - Learning individual rules
- Association rule learning

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

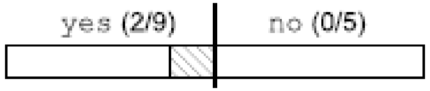
Education	Marital Status	Sex	Has Children	Approved
primary	single	male	no	no
primary	single	male	yes	no
primary	married	male	no	yes
university	divorced	female	no	yes
university	married	female	yes	yes
secondary	single	male	no	no
university	single	female	no	yes
secondary	divorced	female	no	yes
secondary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



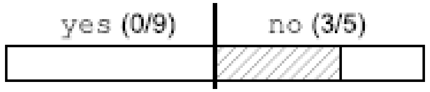
Transforming trees to rules: Survey data

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

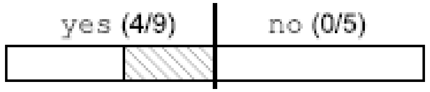
```
IF   MaritalStatus = single
 AND Sex = female
THEN Approved = yes
```



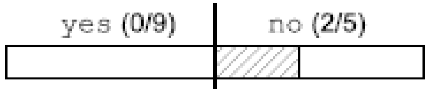

```
IF   MaritalStatus = single
 AND Sex = male
THEN Approved = no
```



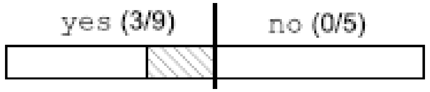

```
IF   MaritalStatus = married
THEN Approved = yes
```

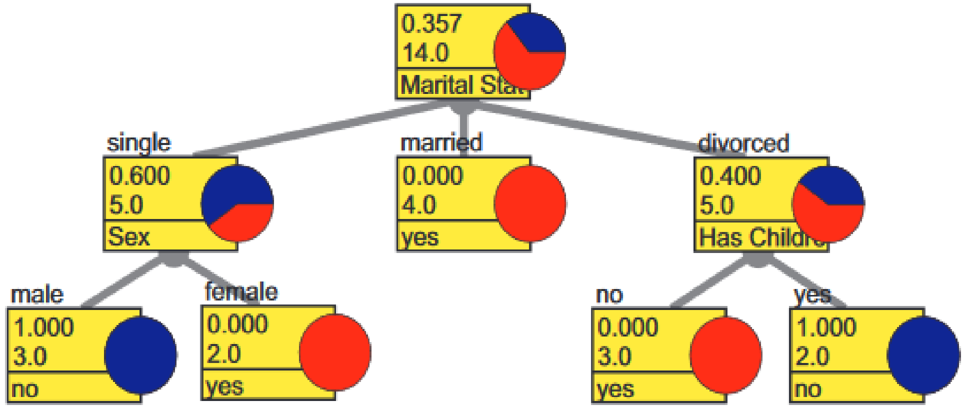



```
IF   MaritalStatus = divorced
 AND HasChildren = yes
THEN Approved = no
```




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

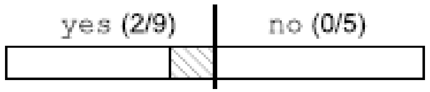




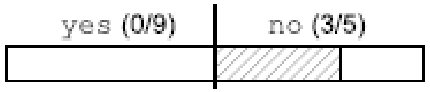
Pruning classification rules: Survey data

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

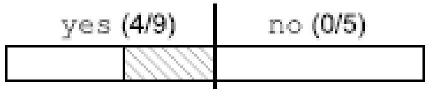
```
IF   MaritalStatus = single
 AND Sex = female
THEN Approved = yes
```



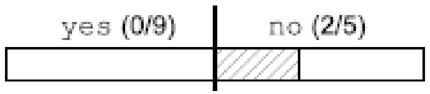

```
IF   MaritalStatus = single
 AND Sex = male
THEN Approved = no
```



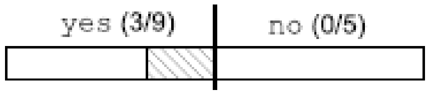

```
IF   MaritalStatus = married
THEN Approved = yes
```



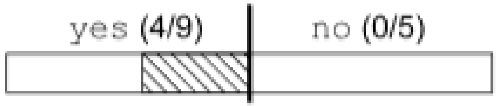

```
IF   MaritalStatus = divorced
 AND HasChildren = yes
THEN Approved = no
```



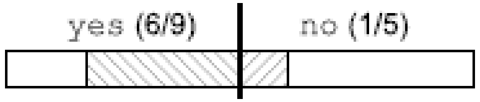

```
IF   MaritalStatus = divorced
 AND HasChildren = no
THEN Approved = yes
```



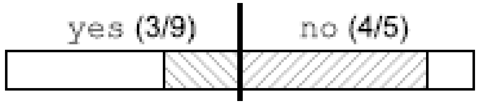

```
IF   MaritalStatus = married
THEN Approved = yes
```




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




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

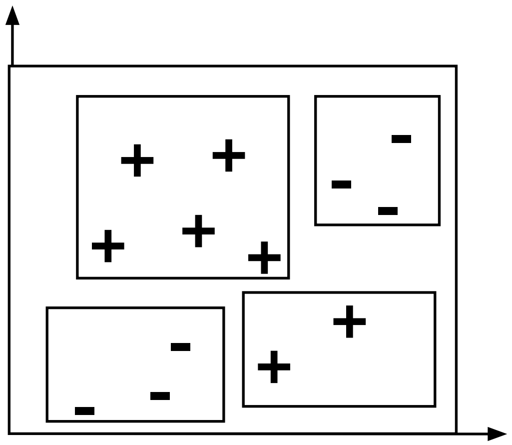



```
DEFAULT  Approved = yes
```

Lesson 3:
Rule Learning

- Transforming decision trees to rules
- Classification rule learning algorithm
  - Covering algorithm
  - Learning individual rules
- Association rule learning

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

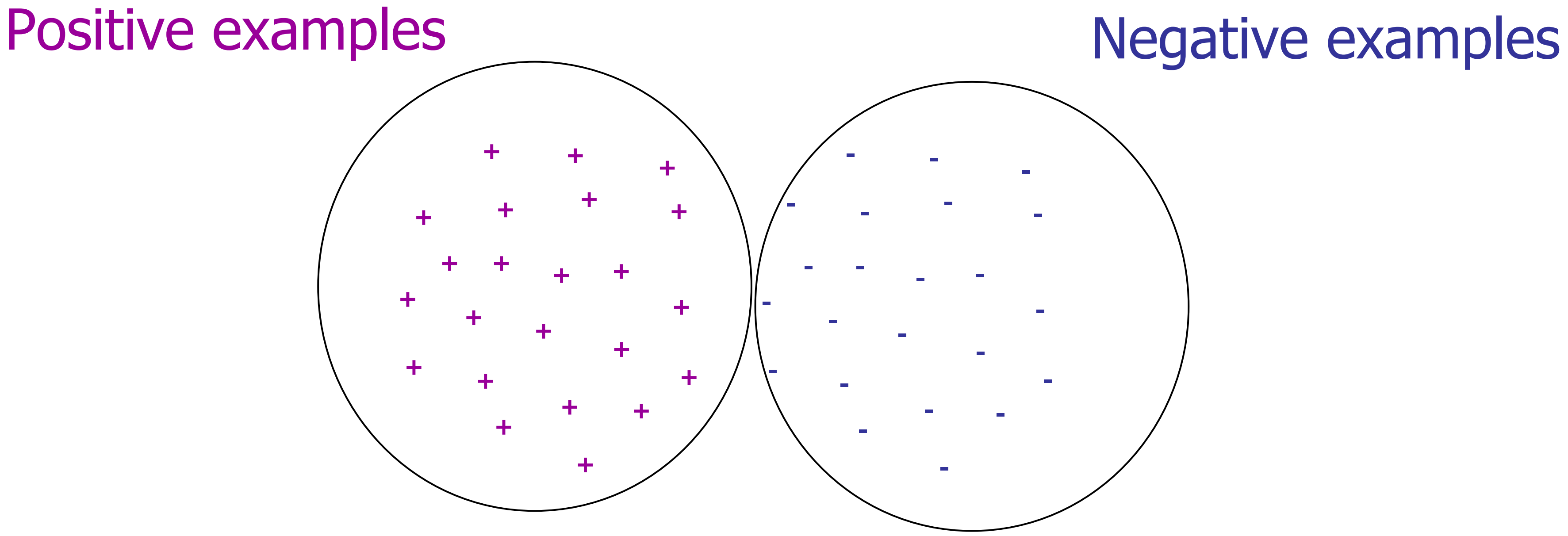


Given examples of 2 classes C_1, C_2

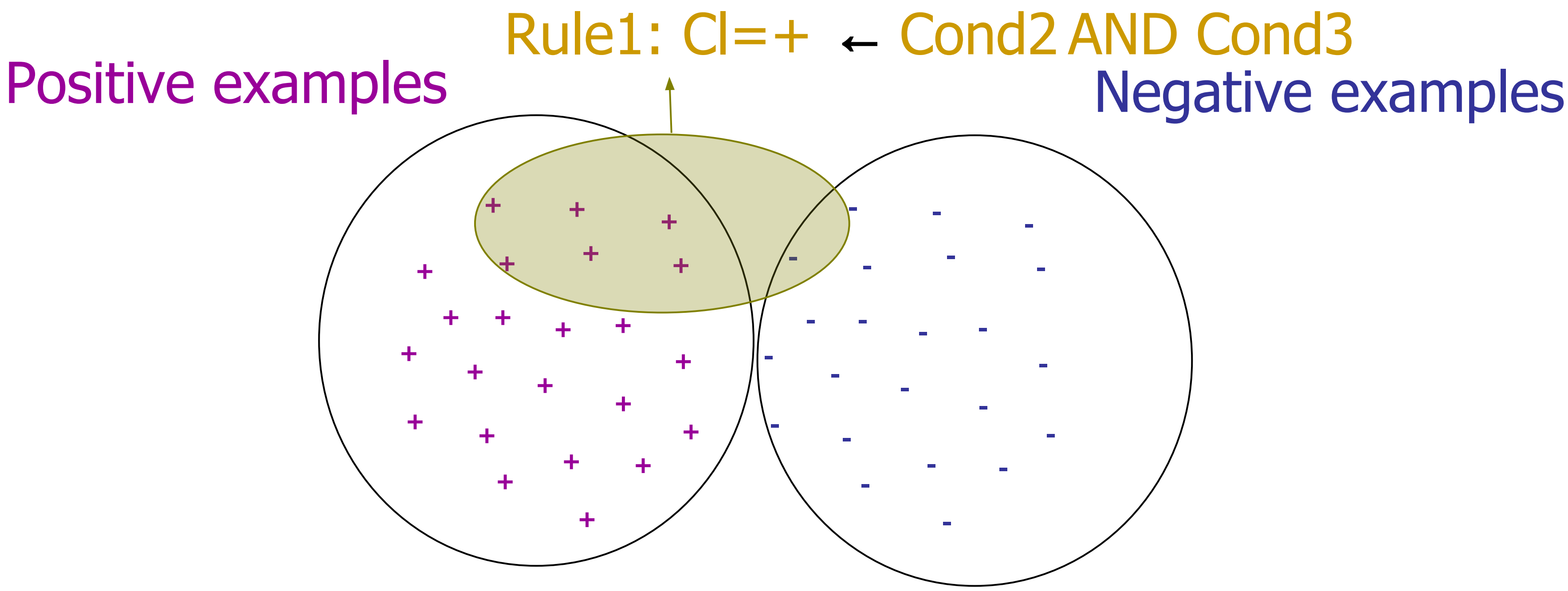
for each class Ci **do**

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

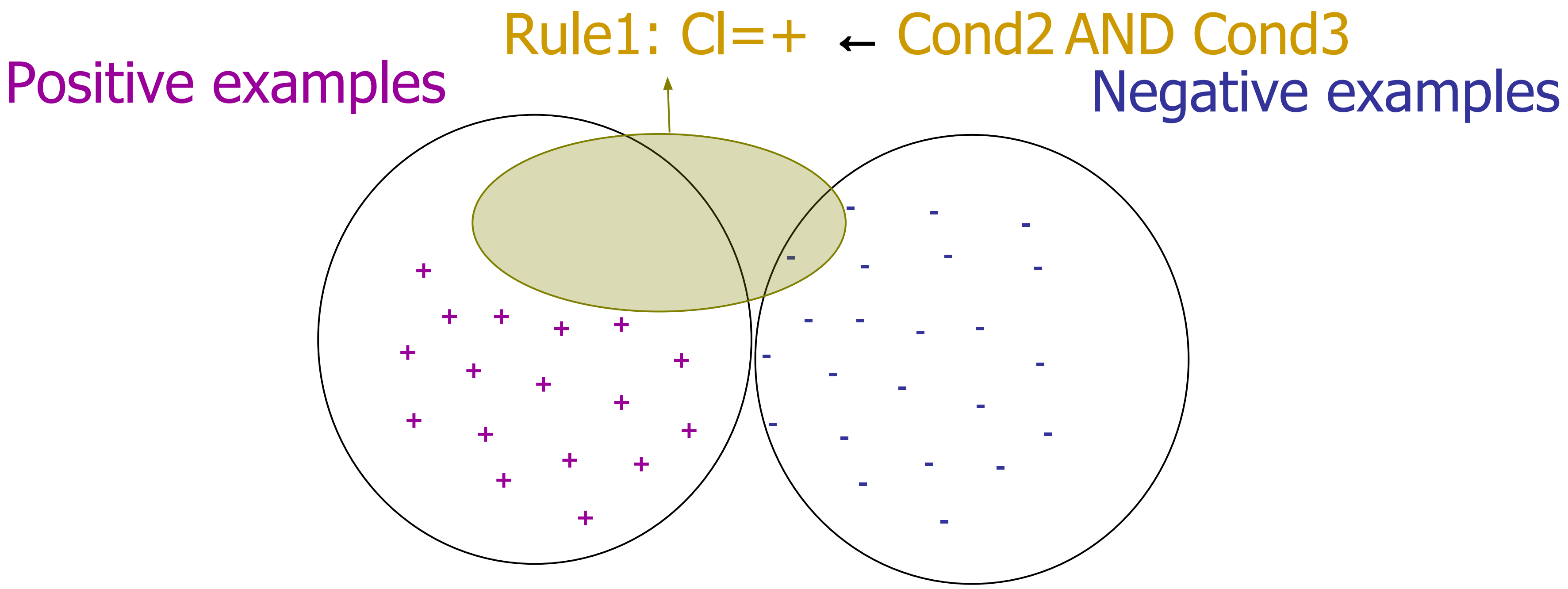
Covering algorithm



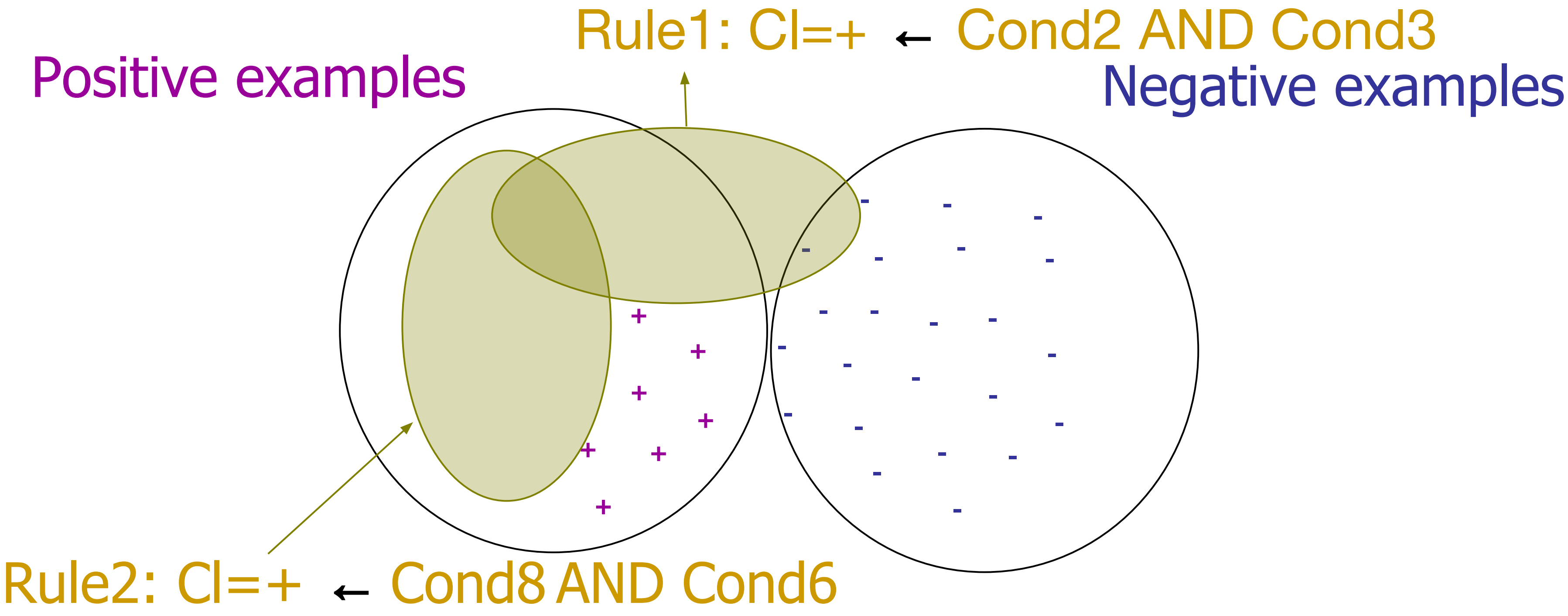
Covering algorithm



Covering algorithm



Covering algorithm



Principles of learning classification rules

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

Important notions:

- Rules are learned separately for each class
  (e.g., separately for two classes: Yes and No)
- Aiming at large "coverage" of the target class
  - Large coverage of class Yes when learning rules for class Yes
  - Large coverage of class No when learning rules for class No
- Default (majority class) rule is added when coverage becomes low (below some user-defined rule pruning parameter)

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

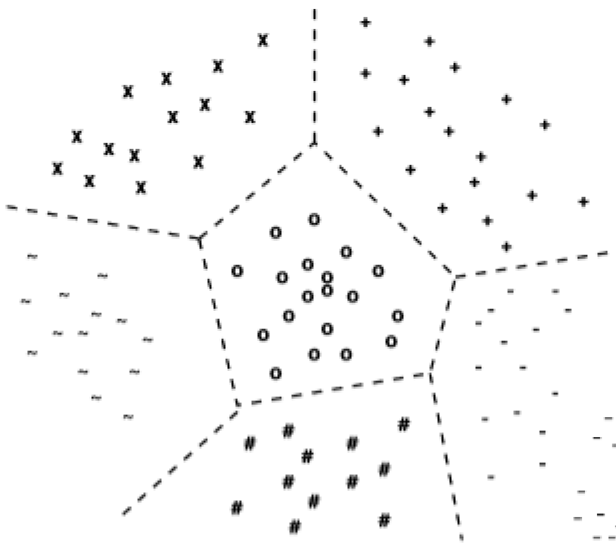


Fig. 10.2: A multiclass classification

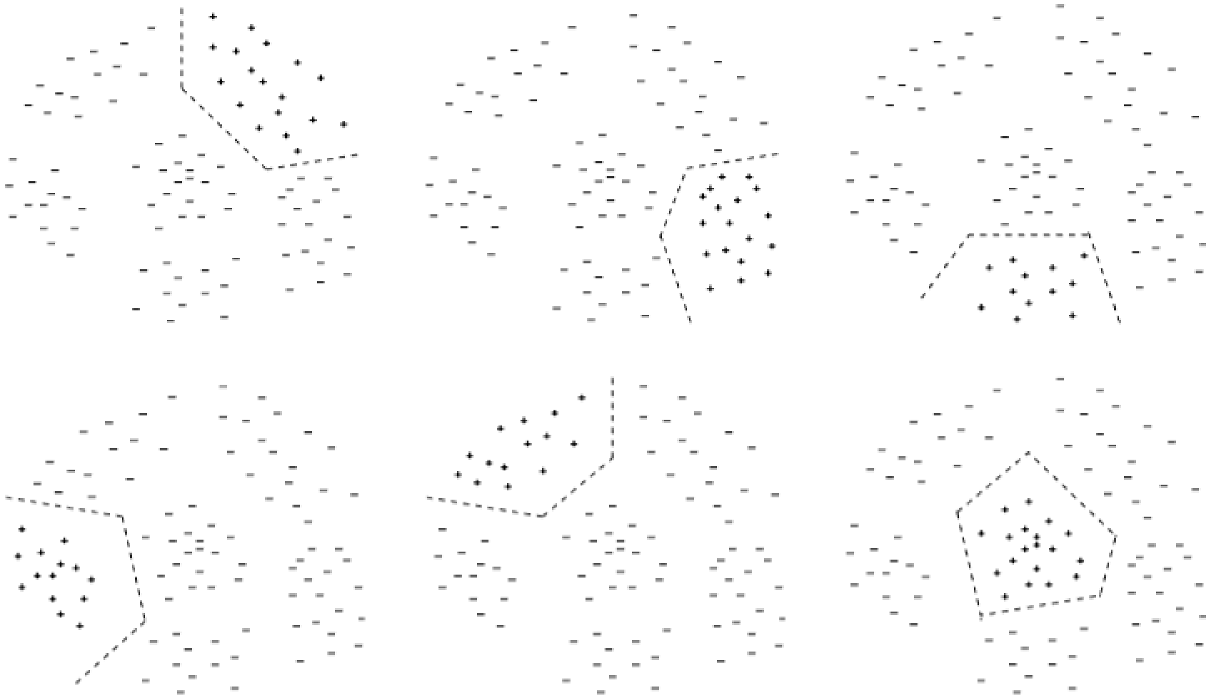


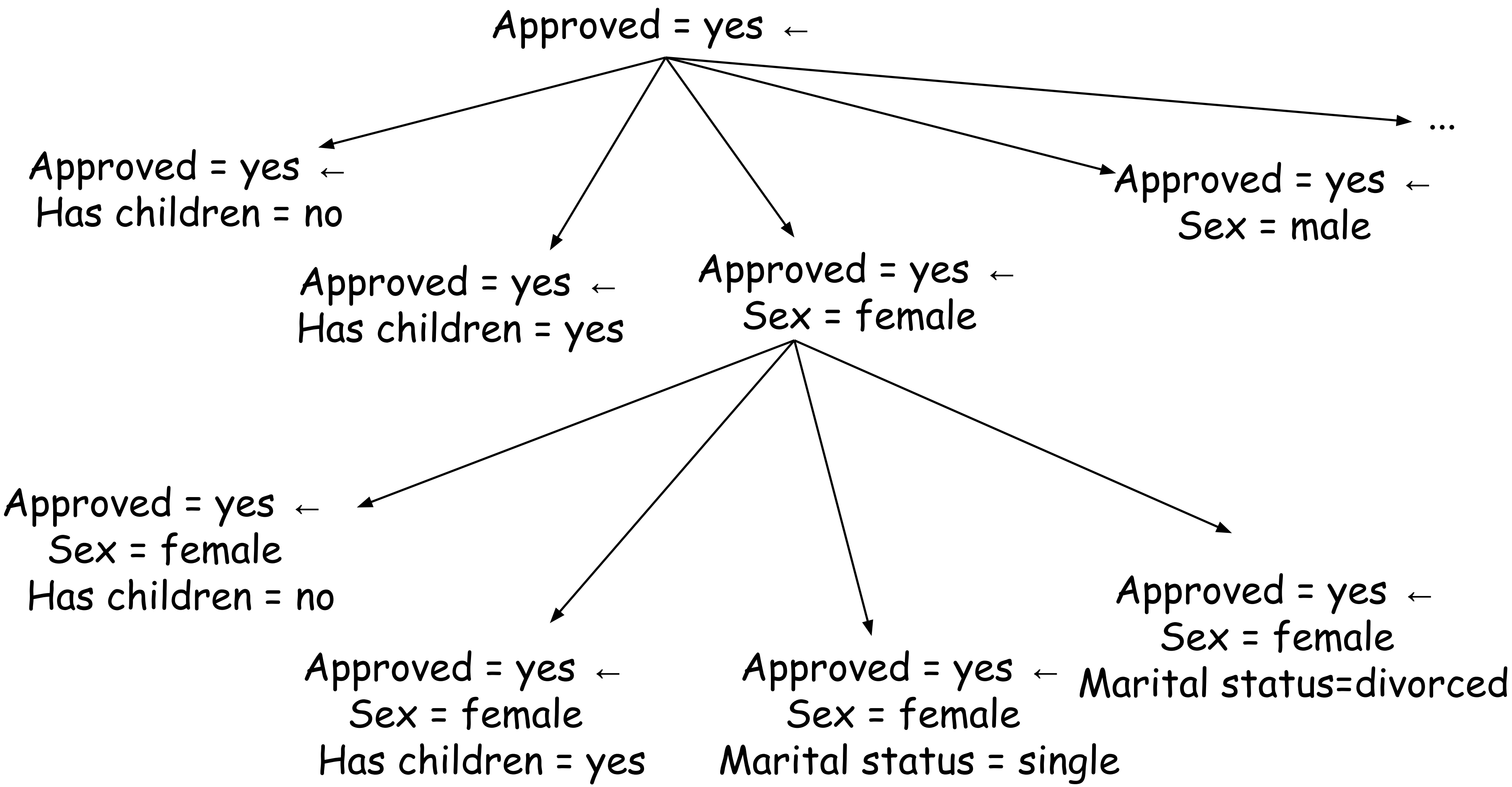
Fig. 10.4: The six binary learning problems that are the result of one-against-all class binarization of the multiclass dataset of Figure 10.2.

Learn-one-rule: Search mechanism and heuristics

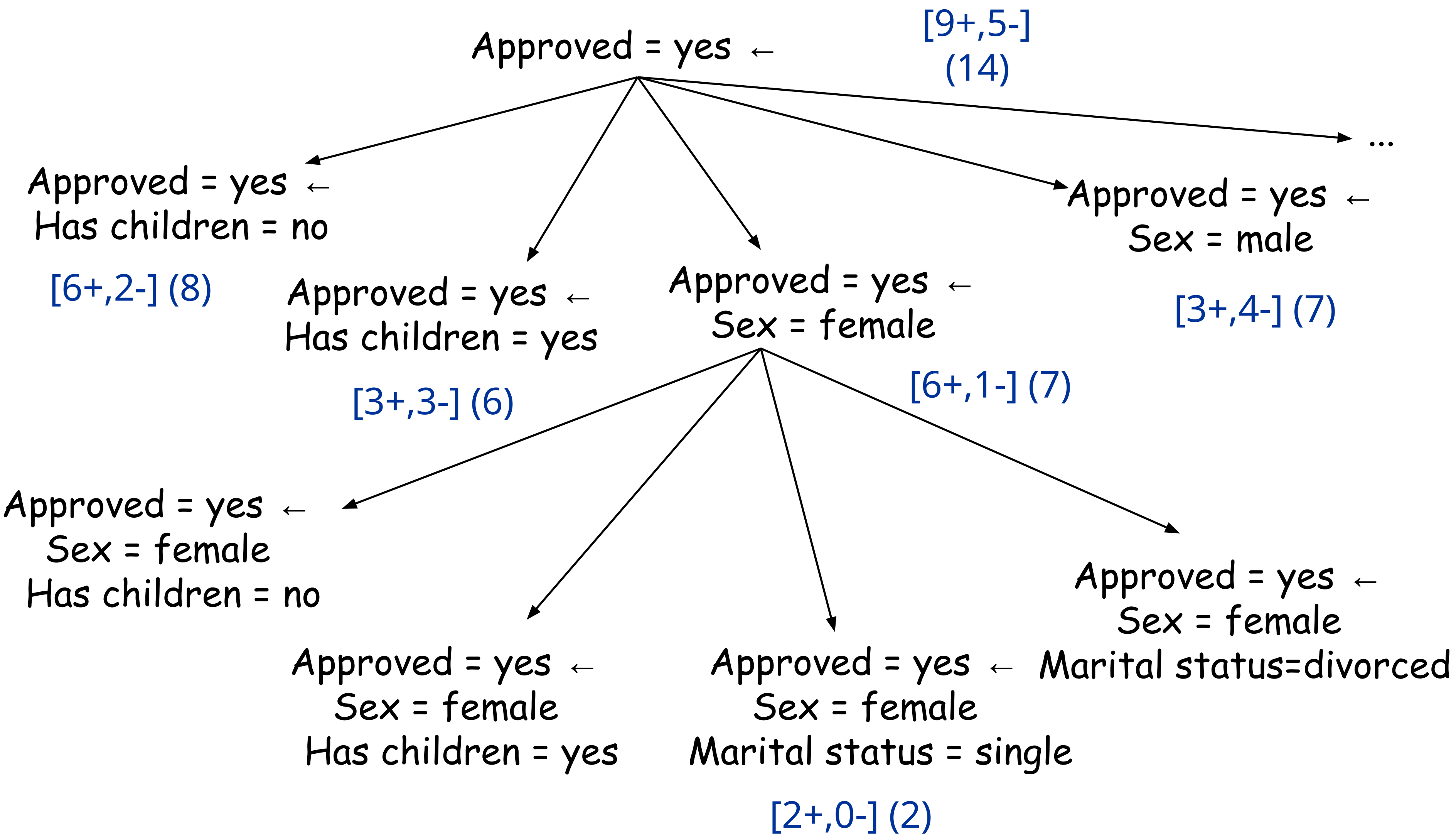
- Assume a two-class problem
- Two classes (+,-), learn rules for + class (Cl)
- Search for specializations R' of a rule R = Cl ← Cond from the RuleBase
- Specialization R' of rule R = Cl ← Cond

  has the form R' = Cl ← Cond & Cond'
- Heuristic search for rules: find the best Cond' to be added to the current rule R, such that rule accuracy is improved, e.g., such that Acc(R') > Acc(R)
  - where the expected **accuracy (precision)** of a rule can be estimated as A(R) = p(Cl|Cond)

Learn-one-rule as heuristic search: Survey data



Learn-one-rule as heuristic search: Survey data



Probability estimates for calculating rule accuracy

- **Relative frequency** :
  - problems with small samples

$$p(Class \mid Cond) =$$

$$= \frac{n(Class.Cond)}{n(Cond)}$$

[6+,1-] (7) = 6/7
[2+,0-] (2) = 2/2 = 1

- **Laplace estimate** :
  - assumes uniform prior distribution of k classes

$$= \frac{n(Class.Cond)+1}{n(Cond)+k} \qquad k = 2$$

[6+,1-] (7) = (6+1) / (7+2) = 7/9
[2+,0-] (2) = (2+1) / (2+2) = 3/4

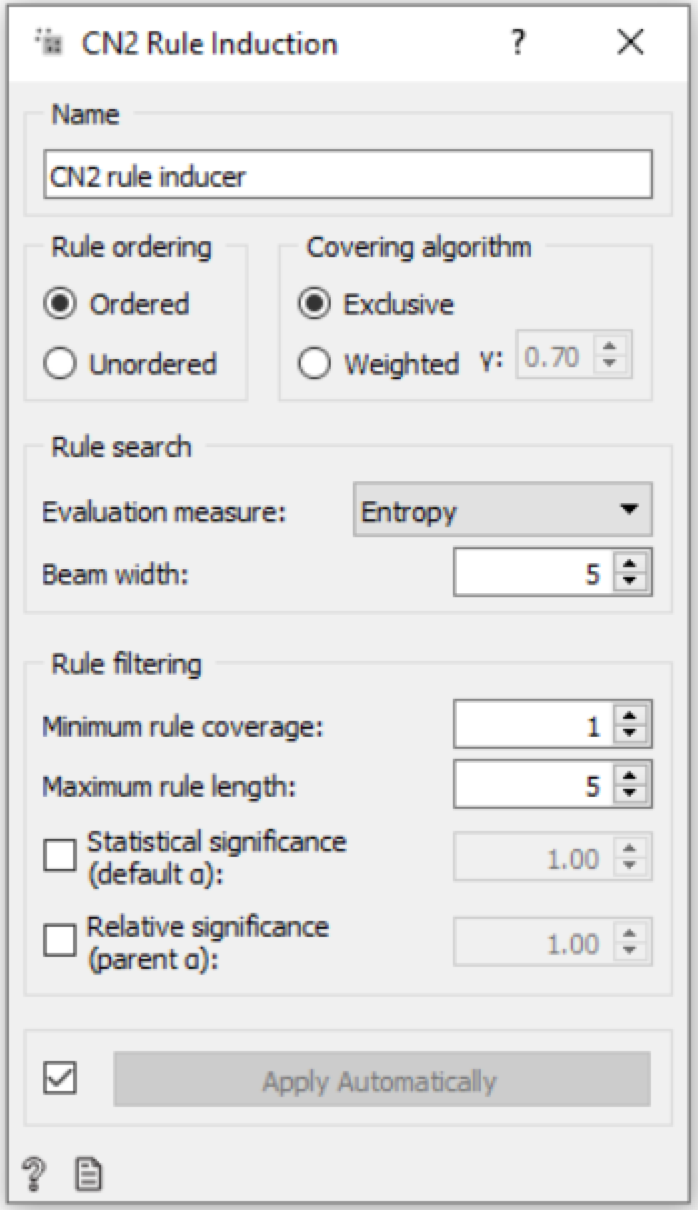
Learn-one-rule: Beam search in CN2 (Clark and Niblett 1989)

- Beam search in CN2 learn-one-rule algorithm:
  - 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 ← BestBody by adding majority class of examples covered by BestBody in rule Head

- A variant of CN-2 is implemented in Orange toolbox

- Best performing rule learning algorithm is Ripper - JRip implementation of Ripper is implemented in WEKA toolbox

CN2 rule learner in Orange



Lesson 3: Rule Learning

- Transforming decision trees to rules
- Classification rule learning algorithm
  - Covering algorithm
  - Learning individual rules
- Association rule learning

Association Rule Learning

Rules: A ⍰ B, if A then B

A and B are itemsets (records, conjunction of items), where items/features are binary-valued attributes)

Given: Transactions

itemsets (records)

	i1	i2	....	i50
t1	1	1		0
t2	0	1		0
	...	...	....	...

Find: A set of association rules in the form A ⍰ B

Example: Market basket analysis

beer & coke => peanuts & chips (0.05, 0.65)

- Support: Sup(A,B) = #AB/#D = p(AB)
- Confidence: Conf(A,B) = #AB/#A = Sup(A,B)/Sup(A) = = p(AB)/p(A) = p(B|A)

Association Rule Learning: Motivation

What people buy in a given shopping experience.

- 25 Osco Drug stores
- 1.2 million market baskets

(A market basket is the stuff you put in
the physical cart and check out at the register.)

- An unexpected pattern

Between 5p.m. and 7p.m. **diapers → beer**

http://www.dssresources.com/newsletters/66.php

Association Rule Learning: Motivation

- Determine associations between groups of items bought by customers.
- No predefined target variable(s).
- Find interesting, useful patterns and relationships.
- Data mining, business intelligence.

* Terminology from market basket analysis (transactions, items, itemsets, …)

Support and Confidence

- The dataset consists of **n** transactions
- We have an association rule A⍰ B

The **support** of an itemset A is defined as the fraction of the transactions in the database T = {T1 . . . Tn} that contain A as a subset.

$$supp(A) = \frac{|A|}{n}$$

The **confidence** of the rule A⍰ B i

$$supp(A \rightarrow B) = \frac{|A \wedge B|}{n}$$

robability of A and B occurring in a transaction, given that the transaction contains A.

$$conf(A \rightarrow B) = \frac{|A \wedge B|}{|A|} = P(B|A)$$

Association Rule Learning: Examples

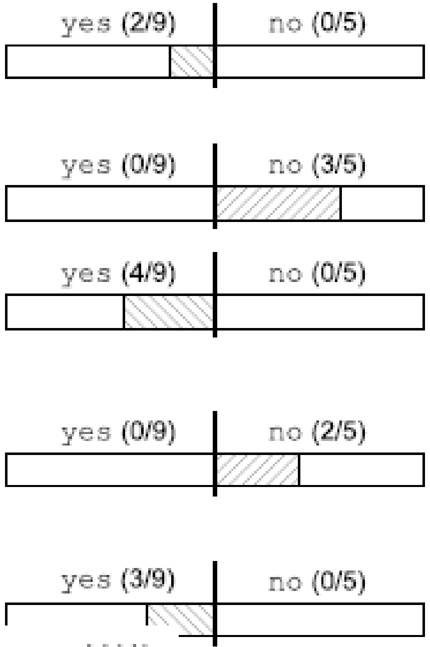
- Market basket analysis
  - beer & coke $\Rightarrow$ peanuts & chips (5%, 65%)

    (IF beer AND coke THEN peanuts AND chips)
  - Support 5%: 5% of all customers buy all four items
  - Confidence 65%: 65% of customers that buy beer and coke also buy peanuts and chips
- Insurance
  - mortgage & loans & savings $\Rightarrow$ insurance (2%, 62%)
  - Support 2%: 2% of all customers have all four
  - Confidence 62%: 62% of all customers that have mortgage, loan and savings also have insurance

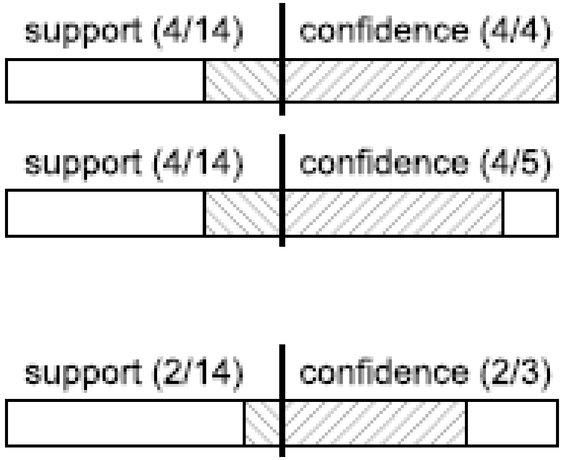
Survey data
association rule learning

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
```




```
IF   Education = university
THEN Sex = female

IF   Approved = no
THEN Sex = male

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



Association Rule Learning

Given: a set of transactions D

Find: all association rules that hold on the set of transactions that have

- user defined minimum support, i.e., support > MinSup, and
- user defined minimum confidence, i.e., confidence > MinConf

It is a form of exploratory data analysis, rather than hypothesis verification

Searching for associations

- Find all large itemsets
- Use the large itemsets to generate association rules
- If XY is a large itemset, compute

  $r = \text{support}(XY) / \text{support}(X)$
- If $r >$ MinConf, then $X \Rightarrow Y$ holds

  (support > MinSup, as XY is large)

Large itemsets

- Large itemsets are itemsets that appear in at least MinSup transaction
- All subsets of a large itemset are large itemsets (e.g., if A,B appears in at least MinSup transactions, so do A and B)
- This observation is the basis for very efficient algorithms for association rules discovery (linear in the number of transactions)

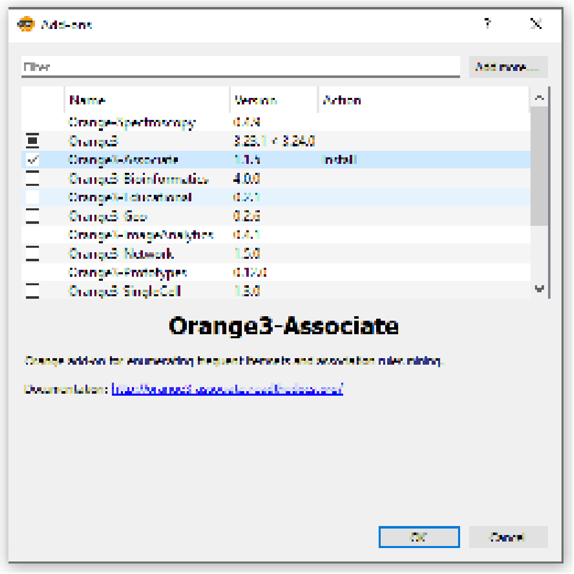
Apriori algorithm

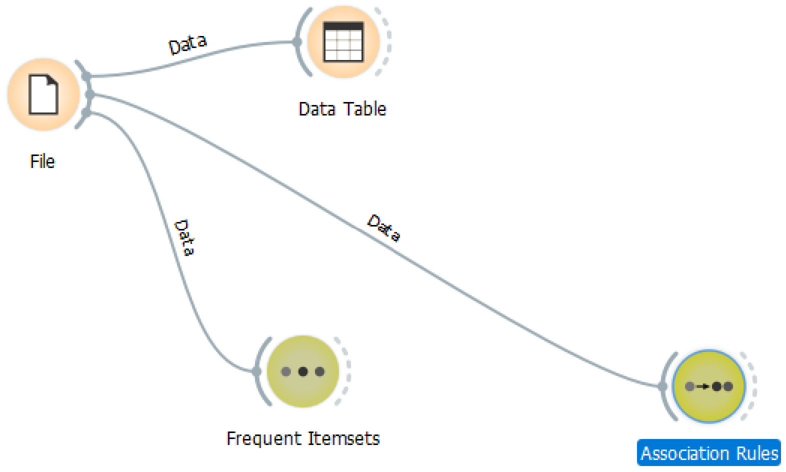
Frequent itemsets → Generate rules → Association rules

- Find all itemsets within the *minSupport* constraint
- For all frequent itemsets, find rules which satisfy the *minConfidence* constraint

*Frequent itemsets = large itemsets, sometimes also frequent patterns

Association rules:
Orange workflow





* Start with a small minSupport and we increase it gradually (to avoid running out of memory)

Association rules vs. Classification rules

Association rules

- Exploration of dependencies
- Different combinations of dependent and independent attributes
- Complete search (all rules found)

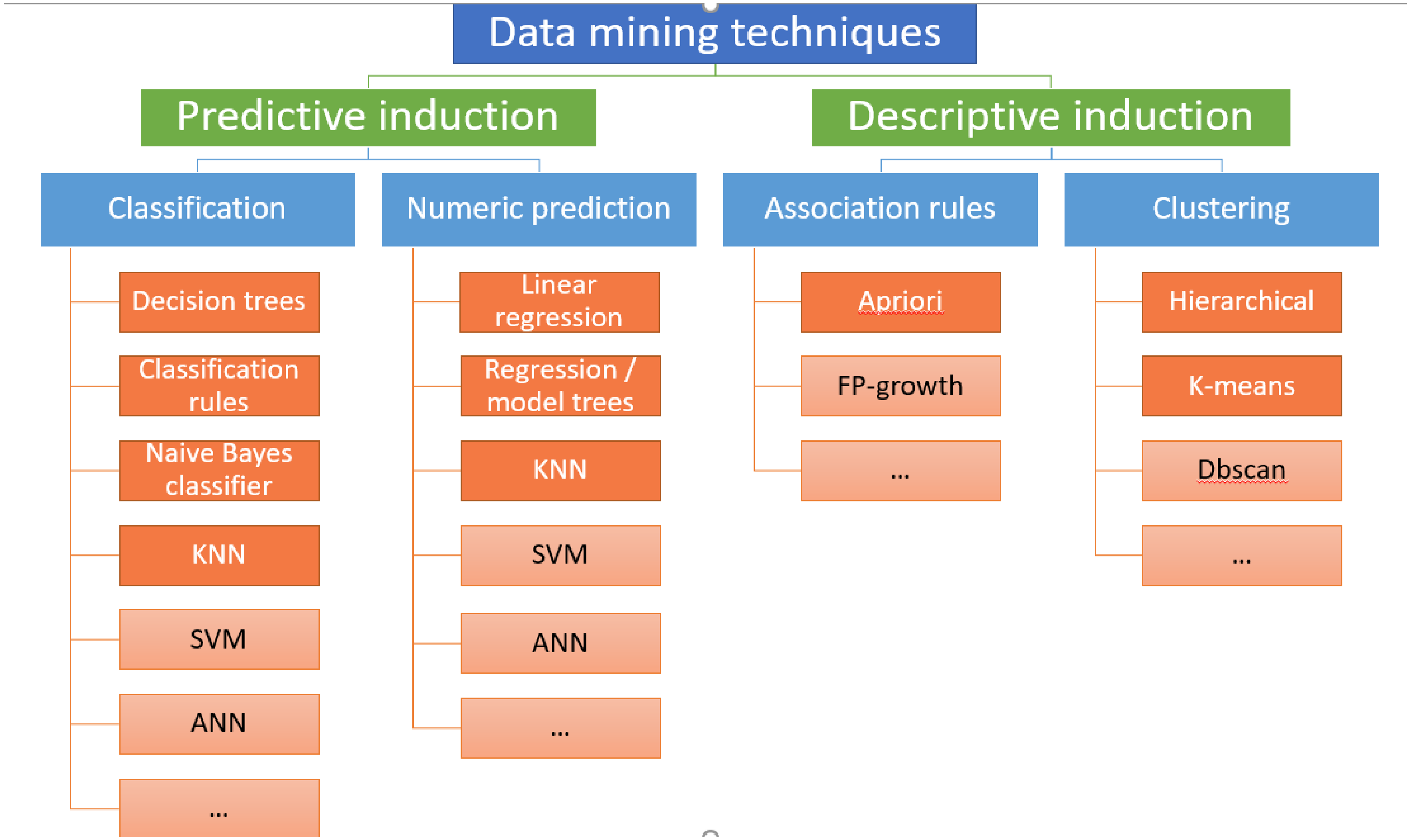
Classification rules

- Focused prediction
- Predict one attribute (class) from the others
- Heuristic search (subset of rules found)

Lesson 3
Summary and Take away messages

- Classification rule learning addresses classification problems
- Algorithms use search heuristics to search the space of possible rules in a general-to-specific manner
- Training data may be noisy - rule truncation help dealing with noisy data to improve predictive accuracy on new, unlabeled data
- Association rule learning is an example of descriptive induction algorithms, aimed at finding interesting patterns in data

Lesson 1 - 3
Summary and Take away messages



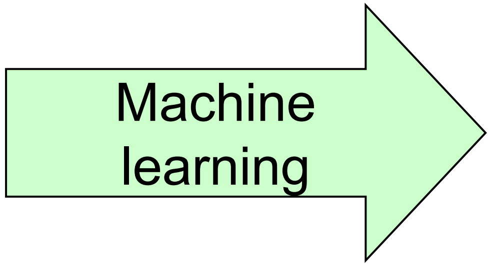
Lesson 4: Text Mining

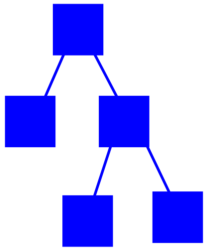
- Introduction to text mining
- Text mining process
- Text mining tasks and applications
- From BoW to dense text embeddings

Background: Machine learning

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





model, patterns, clusters,
…

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

Find: a classification model, a set of interesting patterns

Machine learning: 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

Machine learning vs. text mining

Machine learning:

- instances are objects, belonging to different classes
- instances are feature vectors, described by attribute values
- classification model is learned using machine learning algorithms

Text mining:

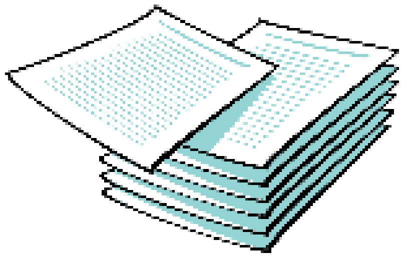
- instances are text documents
- text documents need to be transformed into feature vector representation in data preprocessing
- data mining algorithms can then be used for learning the model

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

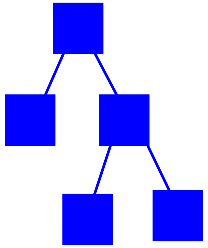
1. BoW features construction
2. Table of BoW vectors construction

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

Machine learning



model, patterns, clusters, …

Text Mining from unlabeled data

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~~

Unlabeled data - clustering: grouping of similar instances

- association rule learning

Lesson 4:
Text Mining (non-obligatory material)

- Introduction to text mining
- Text mining process
- Text mining tasks and applications
- From BoW to dense text embeddings

Text Mining process

- Document preprocessing
- BoW vector construction
- Mining of BoW vector table
  - for text Categorization, Clustering, Summarization, …

Document preprocessing

- Tokenization
  - Convert text to a list of tokens (e.g., words, bigrams ...)
- Stop-word removal
  - Remove words that carry little or no semantic or lexical information (e.g., prepositions “a” or “the” or very frequent words such as “and” which are part of every document, …)
- Part-of-Speech (POS) tagging
  - Annotate words to their POS category (e.g., noun, verb ...)
- …
- Lowercase transformation
- Lemmatization or stemming

Stemming and Lemmatization

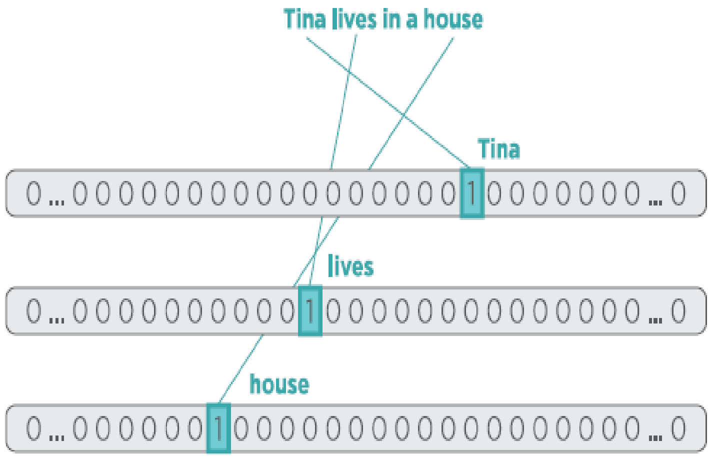
- Different forms of the same word are usually problematic for text data analysis
  - because they have different spelling and similar meaning (e.g., learns, learned, learning,…) should not be treated as unrelated words
- Lemmatization is a process of transforming a word into its normalized form
  - replacing the word by dictionary form of a word (e.g., am, is, are → be), various lemmatizers for English available in R
  - most often by replacing a word's suffix (e.g., in Slovene language, replacing smejem → smejati)
- Stemming is a process of transforming a word into its stem
  - cutting off the suffix of a word (e.g., cats → cat, works, working → work), Porter stemming algorithm for English available in R

Document preprocessing

- The order of preprocessing steps is important
  - Always start with tokenization
- Removal of stop-words is optional
  - Can lead to loss of information
- Lemmatization/stemming is sometimes not necessary
  - Can lead to loss of information
  - But is very useful in highly inflected languages, such as Russian or Slovene
- Other possible preprocessing operations:
  - remove punctuation, spell checking ...
  - Use terms obtained from thesaurus (e.g., WordNet)
  - Construct terms by frequent N-Grams construction

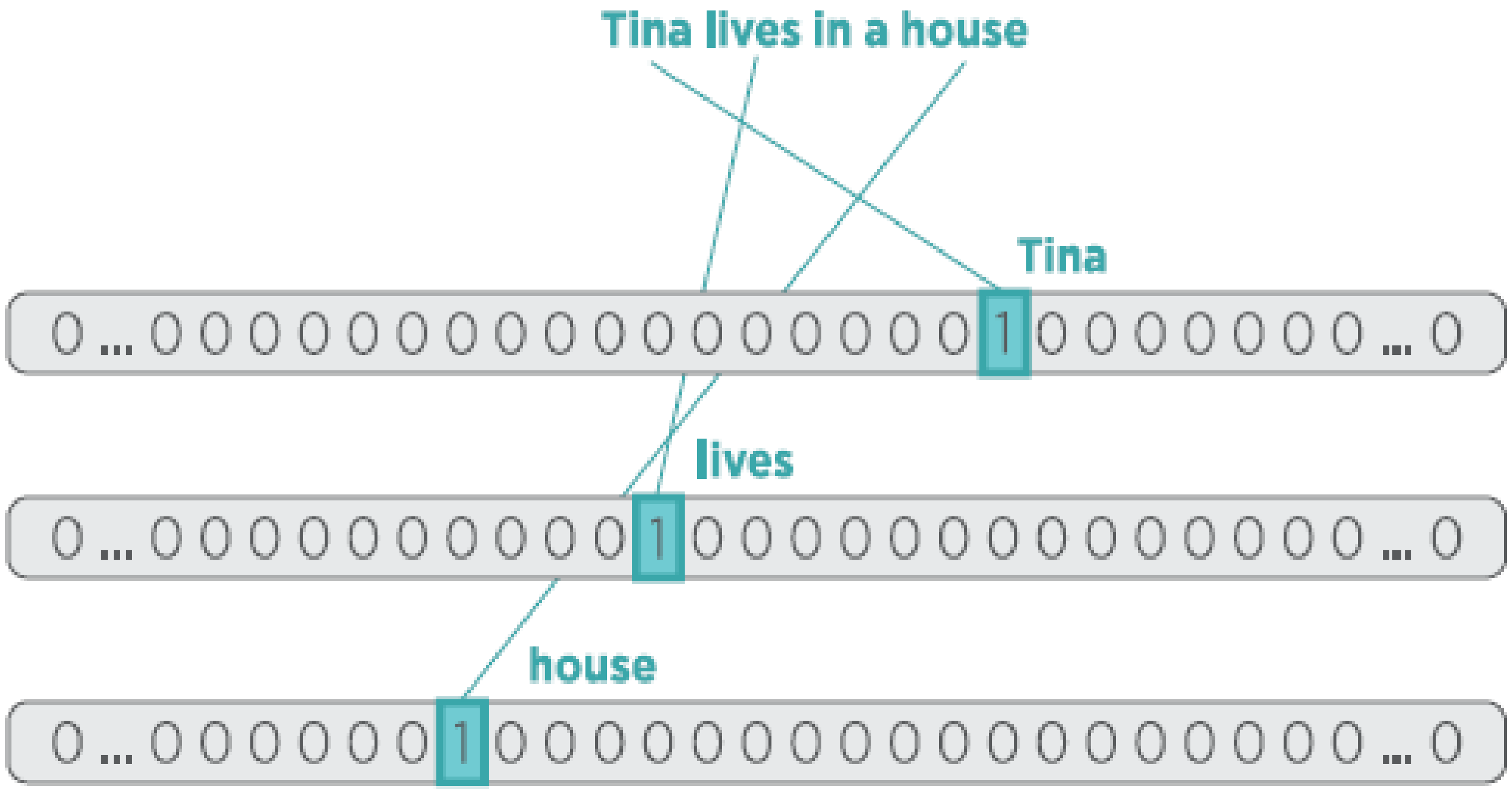
Words/terms representation -
One-hot encoding of dictionary terms

- In machine learning, binary vector representation used for representing nominal variables in tabular data is referred to also as one-hot encoding
- In text mining, binary representation of words/terms is referred also as one-hot encoding of terms, formalized as follows:
  - Dictionary V is an ordered set of vocabulary terms
  - Terms can include single words or more complex terms
  - Vector x
    - is the encoding of term t from V
    - X has length |V|
    - $x_i = 1$ for in V;
    - $x_i = 0$ otherwise



Words/terms representation -
One-hot encoding of dictionary terms

- One-hot encoding of individual words or terms



Document representation as Bag-of-words vectors

- E.g., take a corpus of 1,000 documents, using a dictionary of 50,000 words, where vectors x_i are BoW encodings of documents d_i

document / word	w_1	…	*house*	…	large	…	lives	…	Tina	…	w_{50000}
d_1	0	…	1	…	0	…	0	…	0	…	0
⋮	⋮	⋱	⋮	⋱	⋮	⋱	⋮	⋱	⋮	⋱	⋮
Tina lives in a house.	0	…	1	…	0	…	1	…	1	…	0
⋮	⋮	⋱	⋮	⋱	⋮	⋱	⋮	⋱	⋮	⋱	⋮
The house is large.	0	…	1	…	1	…	0	…	0	…	0
⋮	⋮	⋱	⋮	⋱	⋮	⋱	⋮	⋱	⋮	⋱	⋮
d_{1000}	0	…	0	…	0	…	0	…	0	…	1

Bag-of-words document representation

Document 1:
The quick brown dog jumps over the lazy dog.

Document 2:
This is another lazy person.

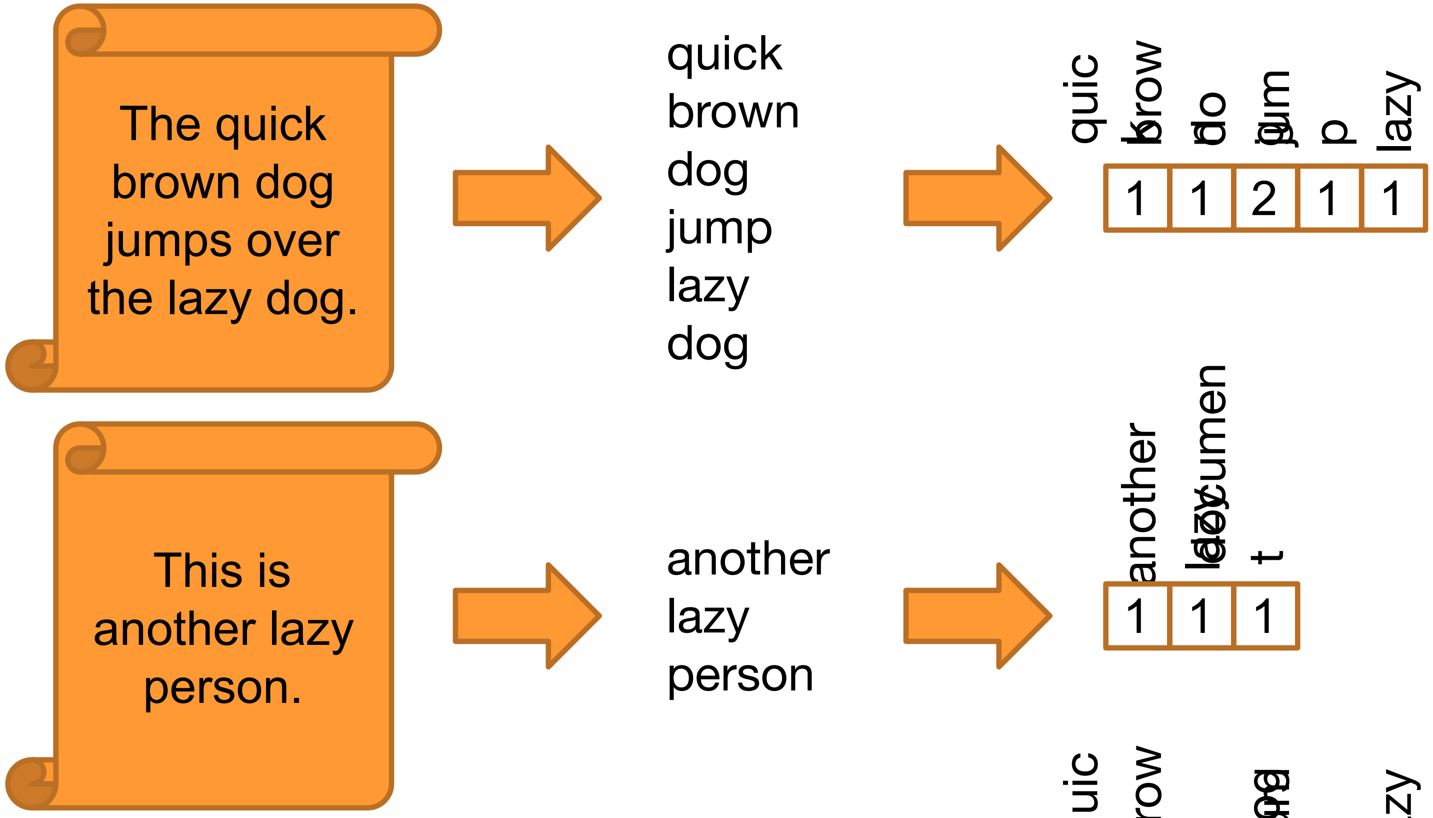
Unigram BoW:

[quick, brown, dog, jump, lazy, another, person]
Doc1 [1 , 0 , 1, 0, 0, 1, 1]
Doc2 [0 , 1 , 0, 1, 1, 1, 1]

Bigram BoW:

[quick brown, brown dog, dog jump, jump lazy, lazy dog, another lazy, lazy person]

Bag-of-words document representation with term frequency weighting



quick	brown	dog	jump	lazy	another	person
1	1	2	1	1		
				1	1	1

TF(t,d) = f(t,d)

- logarithmic TF variant: log(1+TF(t,d))
- normalized TF variant: TF(t,d)/|d|, where |d| is the number of terms in document d

TF-IDF term weighting heuristic (Salton, 1989)

- In bag-of-words representation each word/term is represented as a separate variable having numeric weight.
- The most popular weighting schema is TF-IDF:

$$\text{TF-IDF}(w_i, d_j) = \text{TF}(w_i, d_j) \cdot \log \frac{|D|}{\text{DF}(w_i)}$$

  - $TF(w_i,d_j)$ – term frequency (number of occurrences of w_i in document d_j)
  - $DF(w_i)$ – document frequency (number of documents containing word w_i)
  - $|D|$ – number of all documents
  - $TF\text{-}IDF(w_i,d_j)$ – 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

TF-IDF term weighting heuristic
A more realistic example

- A hotel is an establishment that provides paid lodging on a short-term basis.
- A motel or motor lodge is a hotel designed for motorists.
- Yulia rented room 215 in the Night Lodge motel.
- Aleksei is staying in a hotel.

TF-IDF term weighting heuristic
A more realistic example

- A hotel is an establishment that provides paid lodging on a short-term basis.
- A motel or motor lodge is a hotel designed for motorists.
- Yulia rented room 215 in the Night Lodge motel.
- Aleksei is staying in a hotel.

ID	hotel	motel	lodg	short
1	1	0	1	1
2	1	1	1	0
3	0	1	1	0
4	1	0	0	0
DF	3	2	3	1
$IDF = \log(4/DF)$	0.415	1	0.415	2

ID	hotel	motel	lodg	short
1	0.415	0	0.415	2
2	0.415	1	0.415	0
3	0	1	0.415	0
4	0.415	0	0	0

Document similarity measures

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

$$\text{similarity}(x,y) = \cos(\alpha) = \frac{x \cdot y}{||x|| \cdot ||y||} = \frac{\sum_1^n x_i \cdot y_i}{\sqrt{\sum_1^n x_i^2} \cdot \sqrt{\sum_1^n y_i^2}}$$

- If each document d is represented as a vector of TF-IDF weights

$$\text{similarity}(x,y) = \frac{\sum_{i=1}^{|V|} (\text{TF-IDF}(w_i,x) \cdot \text{TF-IDF}(w_i,y))}{\sqrt{\sum_{i=1}^{|V|} \text{TF-IDF}(w_i,x)^2} \cdot \sqrt{\sum_{i=1}^{|V|} \text{TF-IDF}(w_i,y)^2}}$$

Lesson 4:
Text Mining (non-obligatory material)

- Introduction to text mining
- Text mining process
- Text mining tasks and applications
  - Document classification and document clustering
  - Document clustering for topic ontology construction
  - Document clustering for outlier document detection
  - Literature-based discovery
- From BoW to dense text embeddings

Text mining tasks and applications

- Document clustering and topic identification
- Document classification and categorization
- Anomaly and outlier detection
- Analyzis of sentiment in tweets
- Authorship attribution
- Support in searching the web
- Web user profiling
- Detection of hidden links between domains
- …

Document classification and categorization

- Classification of documents by categories
- Training set consists of pre-categorized documents (class-labeled data)
- The task is to learn a classifier able to classify new documents into a predefined set of categories
- Metaphor: documents are folded into folders, labeled by a topic category

Clustering

- Clustering is a process of finding natural groups in data in a unsupervised way (no class labels pre-assigned to documents)

- Clustering principles:
  - Use similarity/distance measures to determine document similarity
  - Data within cluster should be as similar as possible
  - Data from different clusters should be as different as possible

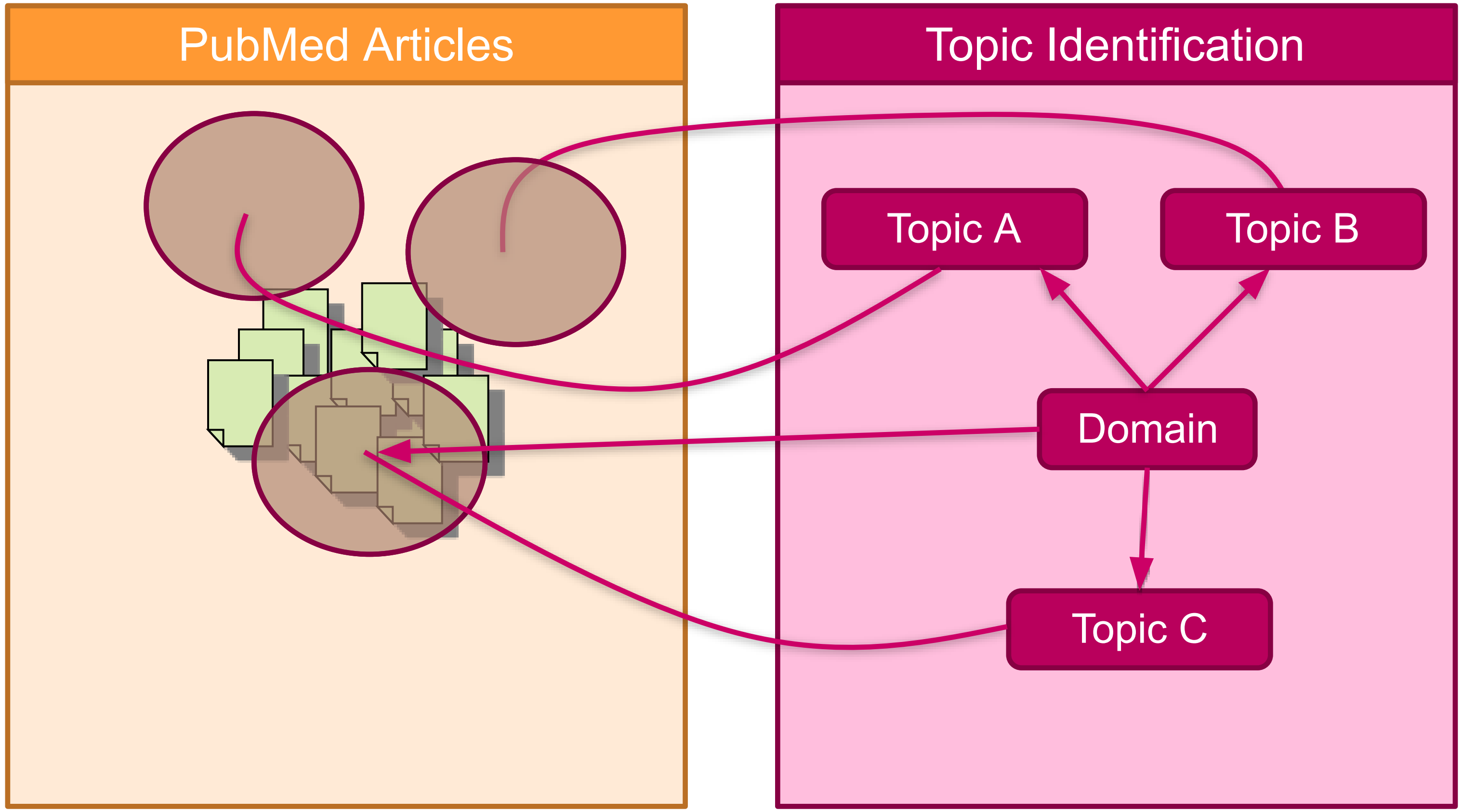
- Most popular clustering methods:
  - K-Means, Agglomerative hierarchical clustering, EM (Gaussian Mixture), …

K-Means clustering

k-Means clustering can be used for semi-automated topic ontology construction

- Given:
  - set of documents (e.g., word-vectors with TFIDF),
  - distance measure (e.g., cosine similarity)
  - K - number of groups
- For each group initialize its centroid with a random document
- While not converging
  - each document is assigned to the nearest group (represented by its centroid)
  - for each group calculate new centroid (group mass point, average document in the group)

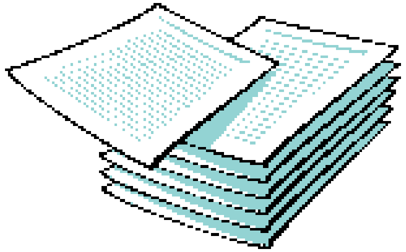
Document clustering for topic identification



Lesson 4:
Text Mining (non-obligatory material)

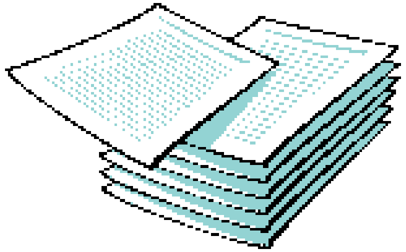
- Introduction to text mining
- Text mining process
- Text mining tasks and applications
- From BoW to dense text embeddings

From sparse to dense text representations: Text embedding



BoW vector 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



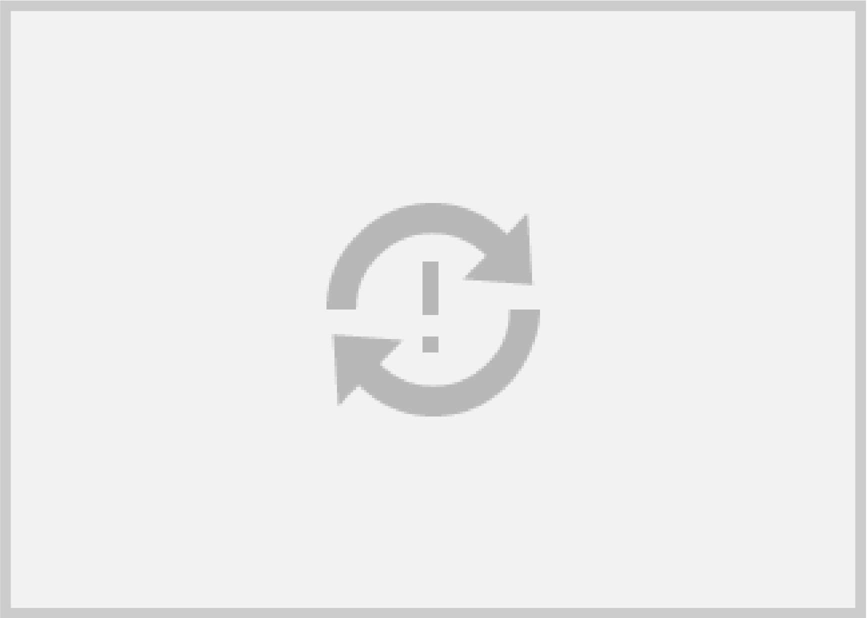
Embedding

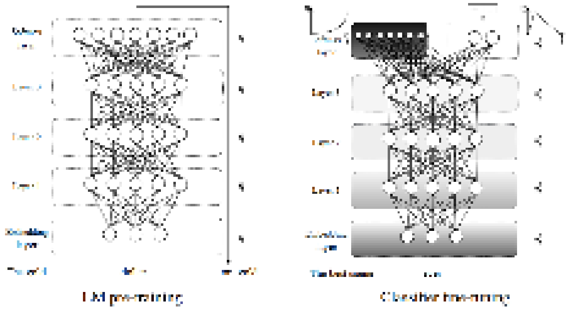
Embeddings-based Data Transformation for Text mining

- Transforming text into compact vector representation (projection into a small number of dimensions k << N)

- Document embedding transforms documents to low-dimensional numeric vectors (rows in a data table). Corpus embedding corresponds to the data table.

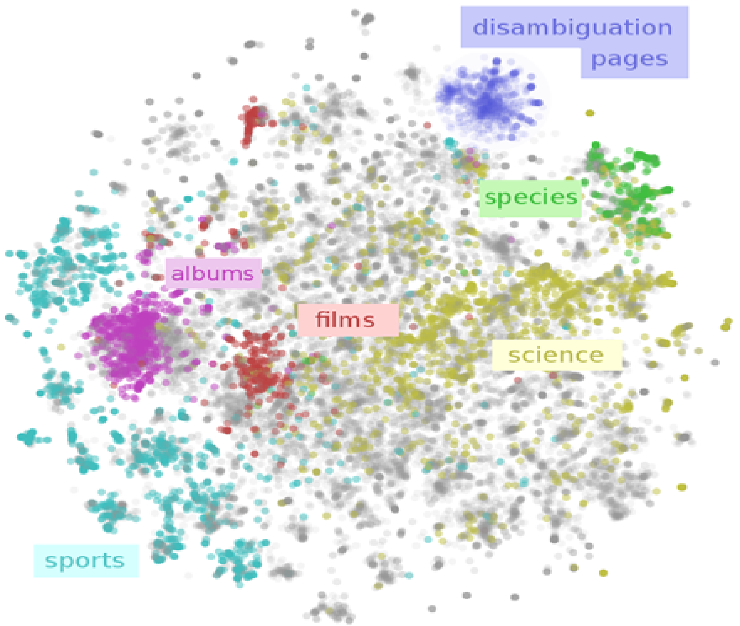
- Table values correspond to weights in the embedding layer of a neural network

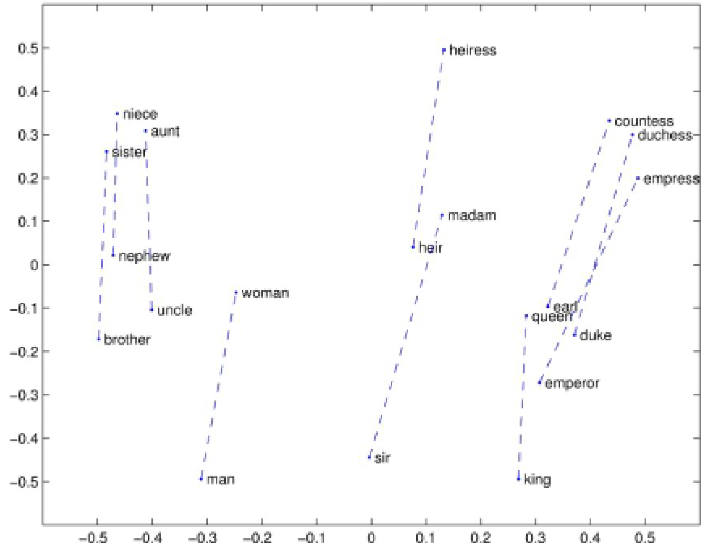




Embedding-based Data Transformation for Text mining

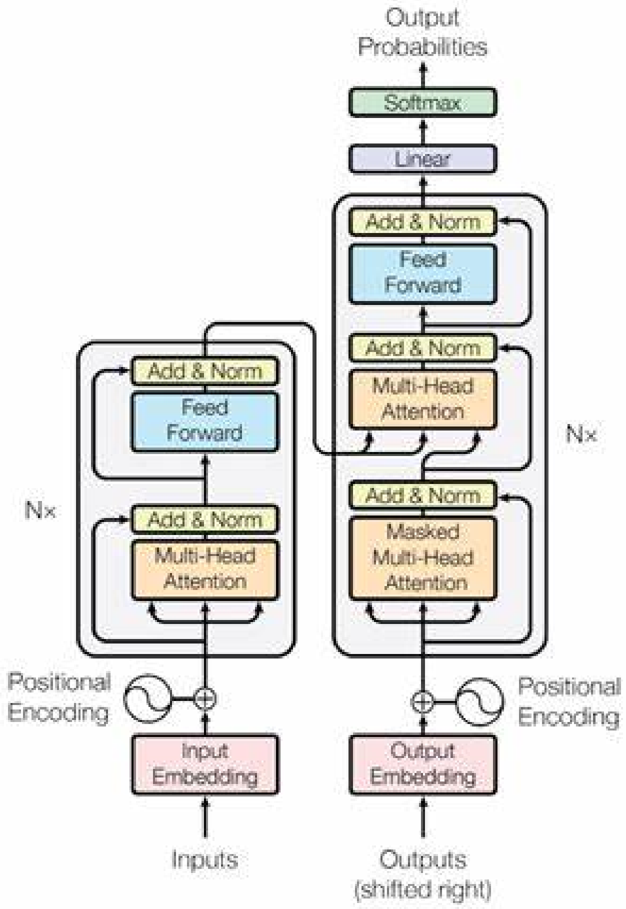
- Corpus embedding, Document embedding, Sentence embedding, …
- **Word embedding** (e.g., word2vec), …
    - Representations of word meaning obtained from corpus statistics
    - Spatial relationships correspond to linguistic relationships





Contemporary Large Language Models

- Transformer neural network architectures
  - Learning in 2 phases
  - 1st phase: pretrained models
    - Predicting masked words, next words, etc.
    - Large document corpora
    - Long training times
  - 2nd phase: model training for a specific task
    - Much faster than 1st phase
    - Transfer of general knowledge from 1st phase
- Example transformer architecture: BERT
- ChatGPT, OpenAI,
  neural network with 175 billion parametrers
- demo on https://chat.openai.com/



Devlin, J., Chang, M.W., Lee, K. and Toutanova, K., 2019 BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *NAACL-HLT*.

Lesson 4 - Text mining
Summary and take away messages

- Text mining definitions, process, typical tasks and selected applications were presented
- Standard BoW document representation and weighting heuristics were presented in detail
- Document classification and document clustering as main text mining approaches were outlined
- Dense text embeddings were briefly introduced
- Contemporary LLMs (Large Language Models) were just briefly mentioned