DATA MINING
Extracting Knowledge From Data

Petr Olmer
CERN

petr.olmer@cern.ch

Motivation

Computers
are useless,
they can only
give you answers.

- What if we do not know what to ask?
- How to discover a knowledge in databases without a specific query?

CERN School of Computing

Many terms, one meaning

- Data mining
- Knowledge discovery in databases
- Data exploration

- A non trivial extraction of novel, implicit, and actionable knowledge from large databases.
  – without a specific hypothesis in mind!
- Techniques for discovering structural patterns in data.

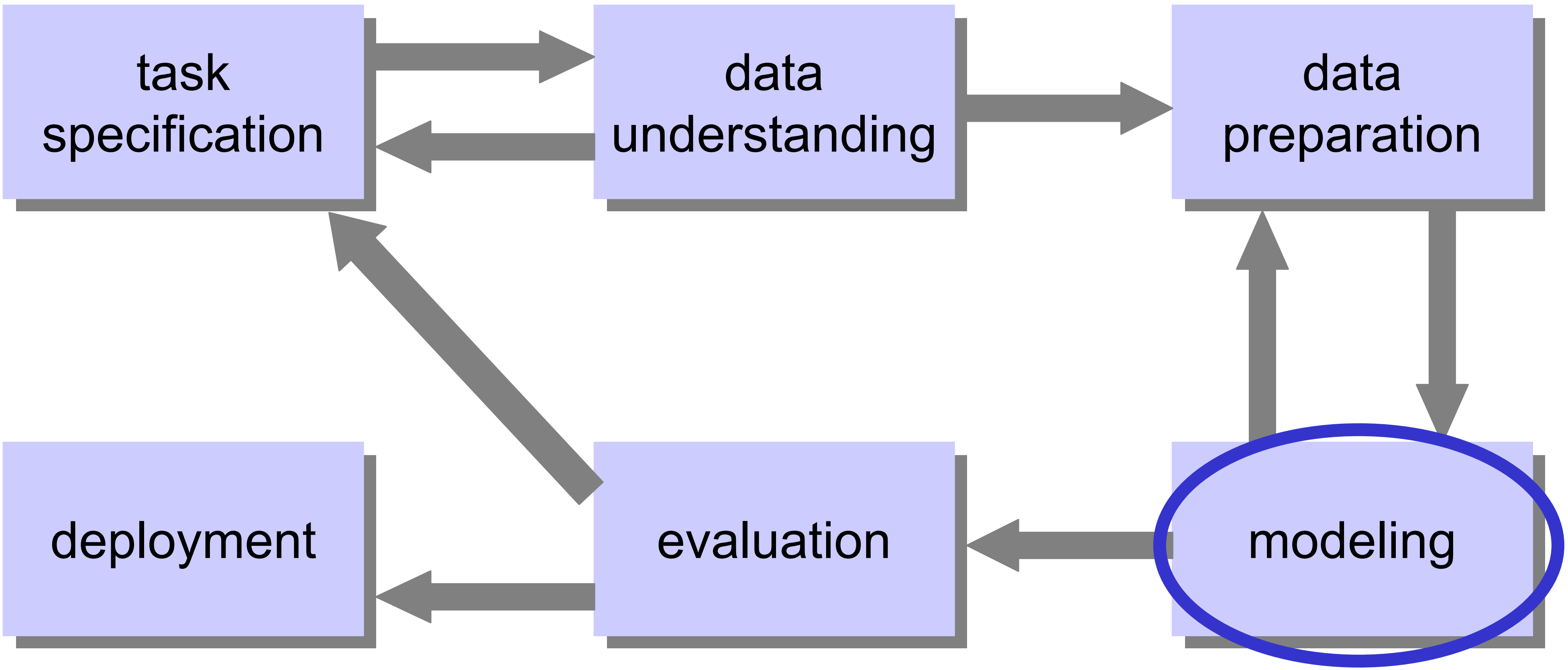
What is inside?

- Databases
  - data warehousing
- Statistics
  - methods
  - but different data source!
- Machine learning
  - output representations
  - algorithms

CERN School of Computing

CRISP-DM

CRoss Industry Standard Process for Data Mining



http://www.crisp-dm.org

CERN School of Computing

Input data: Instances, attributes

A	B	C	D
Mon	21	a	yes
Wed	19	b	yes
Mon	23	a	no
Sun	23	d	yes
Fri	24	c	no
Fri	18	d	yes
Sat	20	c	no
Tue	21	b	no
Mon	25	b	no

Output data: Concepts

- Concept description = what is to be learned

- Classification learning
- Association learning
- Clustering
- Numeric prediction

CERN School of Computing

Task classes

- Predictive tasks
  - Predict an unknown value of the output attribute for a new instance.
- Descriptive tasks
  - Describe structures or relations of attributes.
  - Instances are not related!

CERN School of Computing

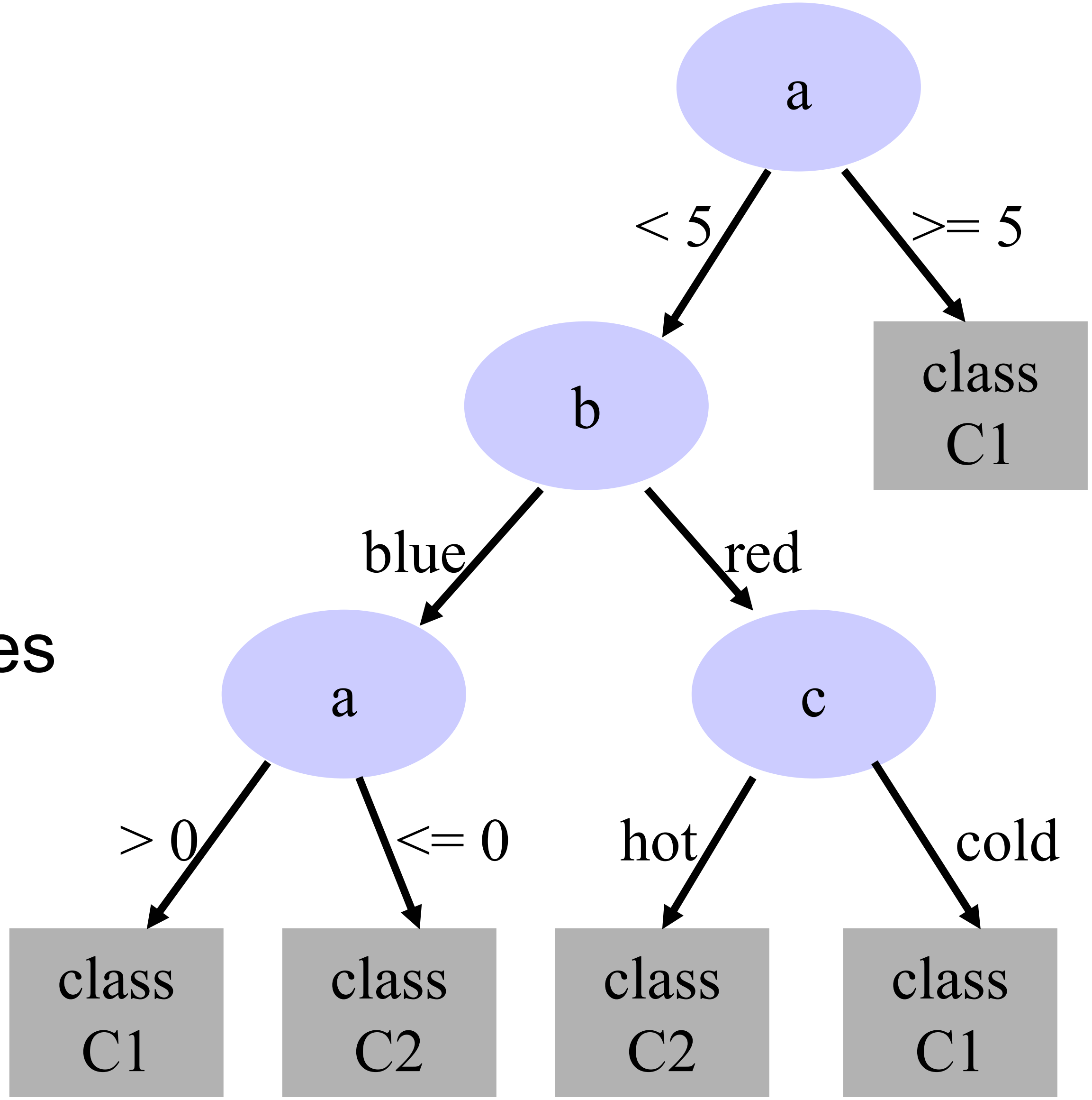
Models and algorithms

- Decision trees
- Classification rules
- Association rules
- k-nearest neighbors
- Cluster analysis

CERN School of Computing

Decision trees

- Inner nodes
  - test a particular attribute against a constant
- Leaf nodes
  - classify all instances that reach the leaf



Classification rules

- If *precondition* then *conclusion*
- An alternative to decision trees
- Rules can be read off a decision tree
  - one rule for each leaf
  - unambiguous, not ordered
  - more complex than necessary

```
If (a>=5) then class
  C1

If (a<5) and
  (b="blue") and
  (a>0) then class
  C1

If (a<5) and
  (b="red") and
  (c="hot") then
  class C2
```

Classification rules
Ordered or not ordered execution?

- Ordered
  - rules out of context can be incorrect
  - widely used
- Not ordered
  - different rules can lead to different conclusions
  - mostly used in boolean closed worlds
    - only *yes* rules are given
    - one rule in DNF

Decision trees / Classification rules 1R algorithm

for each attribute:

  for each value of that attribute:

    count how often each class appears

    find the most frequent class

    rule = assign the class to this attribute-value

  calculate the error rate of the rules

choose the rules with the smallest error rate

CERN School of Computing

Decision trees / Classification rules Naïve Bayes algorithm

$$P(H\,|\,E)=\frac{P(E\,|\,H)\cdot P(H)}{P(E)}$$

- Attributes are
  - equally important
  - independent
- For a new instance, we count the probability for each class.
- Assign the most probable class.
- We use *Laplace estimator* in case of zero probability.
- Attribute dependencies reduce the power of NB.

CERN School of Computing

Decision trees
ID3: A recursive algorithm

- Select the attribute with the biggest *information gain* to place at the root node.
- Make one branch for each possible value.
- Build the subtrees.
- Information required to specify the class
  - when a branch is empty: zero
  - when the branches are equal: a maximum
  - f(a, b, c) = f(a, b + c) + g(b, c)
- Entropy:

$$\sum p_i = 1$$

$$e(p_1, p_2, \ldots, p_n) = -p_1 \log p_1 - p_2 \log p_2 - \ldots - p_n \log p_n$$

CERN School of Computing

Classification rules
PRISM: A covering algorithm

only correct unordered rules

- For each class seek a way of covering all instances in it.
- Start with: If ? then class C1.
- Choose an attribute-value pair to maximize the probability of the desired classification.
  - include as many positive instances as possible
  - exclude as many negative instances as possible
- Improve the precondition.
- There can be more rules for a class!
  - Delete the covered instances and try again.

CERN School of Computing

Association rules

- Structurally the same as C-rules: If - then
- Can predict any attribute or their combination
- Not intended to be used together
- Characteristics:
  - Support = a
  - Accuracy = a / (a + b)

	C	non C
P	a	b
non P	c	d

CERN School of Computing

Association rules
Multiple consequences

- **If A and B then C and D**

- If A and B then C
- If A and B then D

- If A and B and C then D
- If A and B and D then C

CERN School of Computing

Association rules Algorithm

- Algorithms for C-rules can be used
  - very inefficient
- Instead, we seek rules with a given minimum support, and test their accuracy.
- Item sets: combinations of attribute-value pairs
- Generate items sets with the given support.
- From them, generate rules with the given accuracy.

CERN School of Computing

k-nearest neighbor

- Instance-based representation
  - no explicit structure
  - lazy learning
- A new instance is compared with existing ones
  - distance metric
    - a = b, d(a, b) = 0
    - a <> b, d(a, b) = 1
  - closest *k* instances are used for classification
    - majority
    - average

Cluster analysis

- Diagram: how the instances fall into clusters.
- One instance can belong to more clusters.
- Belonging can be probabilistic or fuzzy.
- Clusters can be hierarchical.

Data mining Conclusion

- Different algorithms discover different knowledge in different formats.
- Simple ideas often work very well.
- There’s no magic!

Text mining

- Data mining discovers knowledge in structured data.
- Text mining works with unstructured text.
  - Groups similar documents
  - Classifies documents into taxonomy
  - Finds out the probable author of a document
  - …
- Is it a different task?

CERN School of Computing

How do mathematicians work

- Settings 1:
  - empty kettle
  - fire
  - source of cold water
  - tea bag
- How to prepare tea:
  - put water into the kettle
  - put the kettle on fire
  - when water boils, put the tea bag in the kettle

- Settings 2:
  - kettle with boiling water
  - fire
  - source of cold water
  - tea bag
- How to prepare tea:
  - empty the kettle
  - follow the previous case

CERN School of Computing

Text mining
Is it different?

- Maybe it is, but we do not care.
- We convert free text to structured data…
- … and “follow the previous case”.

Google News
How does it work?

- http://news.google.com
- Search web for the news.
  - Parse content of given web sites.
- Convert news (documents) to structured data.
  - Documents become vectors.
- Cluster analysis.
  - Similar documents are grouped together.
- Importance analysis.
  - Important documents are on the top

CERN School of Computing

From documents to vectors

- We match documents with terms
  - Can be given (ontology)
  - Can be derived from documents
- Documents are described as vectors of weights
  - $d = (1, 0, 0, 1, 1)$
  - $t1$, $t4$, $t5$ are in d
  - $t2$, $t3$ are not in d

TFIDF
Term Frequency / Inverse Document Frequency

- TF(*t*, *d*) = how many times *t* occurs in *d*
- DF(*t*) = in how many documents *t* occurs at least once
- $\mathrm{IDF}(t) = \log \frac{|D|}{\mathrm{DF}(t)}$

- Term is important if its
  - TF is high
  - IDF is high
- Weight(*d*, *t*) = TF(*t*, *d*) · IDF(*t*)

Cluster analysis

- Vectors
  - Cosine similarity

$$\mathrm{sim}(d_i, d_j) = \frac{d_i \times d_j}{|d_i| \cdot |d_j|}$$

- On-line analysis
  - A new document arrives.
  - Try k-nearest neighbors.
  - If neighbors are too far, leave it alone.

Text mining
Conclusion

- Text mining is very young.
  - Research is on-going heavily
- We convert text to data.
  - Documents to vectors
  - Term weights: TFIDF
- We can use data mining methods.
  - Classification
  - Cluster analysis
  - …

CERN School of Computing

References

- Ian H. Witten, Eibe Frank:
  Data Mining: Practical Machine Learning Tools and Techniques with Java Implementations
- Michael W. Berry:
  Survey of Text Mining: Clustering, Classification, and Retrieval
- http://kdnuggets.com/
- http://www.cern.ch/Petr.Olmer/dm.html

Questions?

Computers
are useless,
they can only
give you answers.

Petr Olmer
petr.olmer@cern.ch