University of Osnabrück, Institute of Cognitive Science

Machine Learning

1 – Introduction

SS 2018

Gunther Heidemann

Survey

1. Organization of the course
2. Survey of machine learning:
   - Why ML ?
   - Examples
   - Relevant disciplines
   - What is the learning problem?
   - Major issues in ML

Organization of the course

4h lecture + 2h practice

Time and location:

Tuesday	14h – 16h	(c.t.)	66/E33	Practice
Wednesday	10h – 12h	(c.t.)	93/E31	Lecture
Thursday	10h – 12h	(c.t.)	93/E31	Lecture

Lectures and practice sessions may be switched.

See Studip for the schedule!

(See *Ablaufplan*, practice is called *Sitzung*)

Written exam Thursday 5th July.

Contents

- Unsupervised learning
- Supervised learning
- Reinforcement learning
- Aspects of data mining and pattern recognition
- Clustering
- Dimension reduction
- Artificial neural networks
- Classification

Textbooks: Machine Learning, Pattern Recognition

1. Tom M. Mitchell: *Machine Learning*, McGraw-Hill
2. Ethem Alpaydin: *Introduction to Machine Learning*, MIT Press; Ethem Alpaydin: *Maschinelles Lernen*, Oldenbourg
3. Christopher M. Bishop: *Pattern Recognition and Machine Learning*, Springer
4. Trevor Hastie, Robert Tibshirani, Jerome Friedman: *The Elements of Statistical Learning*, Springer
5. Vladimir Cherkassky, Filip Mulier: *Learning from Data*, IEEE Press
6. B. D. Ripley: *Pattern Recognition and Neural Networks*, Cambridge University Press
7. Ian H. Witten, Eibe Frank, Mark A. Hall: *Data Mining*, Morgan Kaufmann
8. Stuart Russell, Peter Norvig: *Artificial Intelligence*, Pearson

Textbooks: Neural Networks

1. Simon Haykin: *Neural Networks*, Prentice Hall
2. Robert Callan: *The Essence of Neural Networks*, Prentice Hall
3. John Hertz, Anders Krogh, Richard G. Palmer: *Introduction to the Theory of Neural Computation*, Addison-Wesley
4. Helge Ritter, Thomas Martinetz, Klaus Schulten: *Neuronale Netze*, Addison-Wesley
5. Teuvo Kohonen: *Self-Organizing Maps*, Springer

Course material

- This course is based primarily on the textbook by Tom M. Mitchell *Machine Learning*.
- Many slides are based on the slides and graphics by Tom M. Mitchell accompanying the textbook, available at www.cs.cmu.edu/~tom/mlbook.html
- If you find an error ➔ send an email!
- Slides and practice materials will be available at Studip.
- Slides may be corrected and updated at times, so get the latest version at the end of the semester.

Outline

- Why ML ?
- Data mining as a part of ML
- What is a well defined learning problem?
- Example
- Issues in ML

Why machine learning?

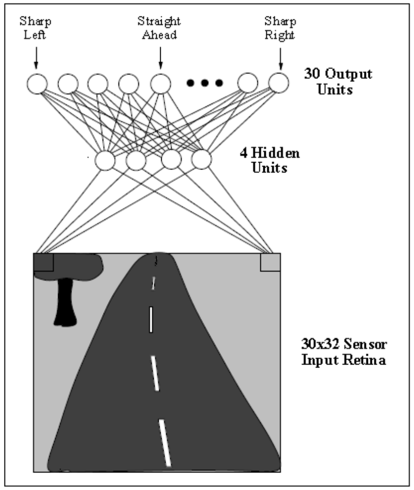
- Growing flood of data
- Growing computational power

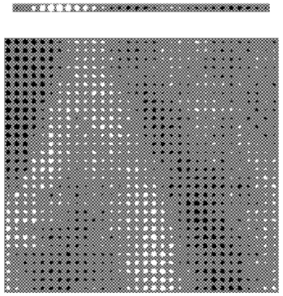
- Knowledge representation in “classical” AI:
  - Explicit representation of knowledge,
  - based on symbols,
  - that usually represent high-level concepts
  - made by humans.
- Problems of explicit models:
  - Huge effort;
  - many knowledge domains are not accessible, e.g., how to walk;
  - low-level, i.e., close-to-signal knowledge can not be acquired and represented.

Problems too difficult to program explicitly

Thus problems remain we can't program entirely manually, e.g.,

- pattern recognition,
- vision and speech recognition,
- low level control and its adaptation.





[M]

Alvinn [Pomerleau 1989]
drives 70 mph on highways

Why machine learning?

ML solves many of the above problems:

- No need for modeling
- Knowledge can be acquired from examples
- Low level knowledge accessible
- Learning more human-like
- Some algorithms have a cognitive motivation

Why machine learning?

Thus ML makes many problems tractable:

- Pattern recognition
  - Vision
  - Speech and audio signals
- Control
  - robot control
  - vehicle control
  - biological / chemical process control
- Prediction, e.g., for time series
- Fusion of different data sources and modalities
- Context can be handled
- Erroneous data can be handled
- Data analysis and **data mining**

What is the learning problem?

Learning = improving with experience at some task

- Improve over task *T*
- with respect to performance measure *P*
- based on experience *E*.

Example: Learn to play checkers

- *T*: Play checkers
- *P*: % of games won in tournament
- *E*: playing against self

Problems of learning

- What exactly is experience?
- What should be learned?
- How to represent experience?
- Learning algorithms?
- Types of training experience:
  - Direct or indirect?
  - Teacher or not?
- Is training representative of the performance goal?
- How can a target function be defined?

Target functions

Choose the target function *V*:

- ChooseMove: *Board* ➔ *Move* ??
- *V*: *Board* ➔ $\mathfrak{R}$??

Possible definition of a target function:

- If *b* is a final board state that is won, then $V(b) = 100$.
- If *b* is a final board state that is lost, then $V(b) = -100$.
- If *b* is a final board state that is drawn, then $V(b) = 0$.
- If *b* is not a final state, then $V(b) = V(b')$ where *b´* is the best final board state that can be achieved starting from *b* and playing optimally until the end of the (deterministic!) game.

This gives correct values, but is not operational.

Representation of the target function

- Collection of rules?
- Neural network?
- Analytical function of board features?

Example:

$V'(b) = w_0 + w_1\, wp(b) + w_2\, rp(b) + w_3\, wk(b) + w_4\, rk(b) + w_5 wt(b) + w_6 rt(b)$

- $wp(b)$ # white pieces
- $rp(b)$ # red pieces
- $wk(b)$ # white kings
- $rk(b)$ # red kings
- $wt(b)$ “white threatens“, i.e., # red pieces which can be taken on whites next turn
- $rt(b)$ “red threatens“

Obtaining training examples

- $V(b)$: the true target function
- $V'(b)$: the learned function
- $V_{\text{train}}(b)$: the training value

One rule for estimating training values:

$V_{\text{train}}(b) \leftarrow V'(Successor(b))$

Adapting the target function

Choose weight training rule, e.g., LMS weight update rule:

Repeat

1. Select a training example b at random.
2. Calculate $V'(b)$ based on current weights
3. Compute $error(b)$:

$$error(b) = V_{train}(b) - V'(b)$$

4. For each board feature f_i, update weight w_i :

$$w_i \leftarrow w_i + \varepsilon \cdot f_i \cdot error(b)$$

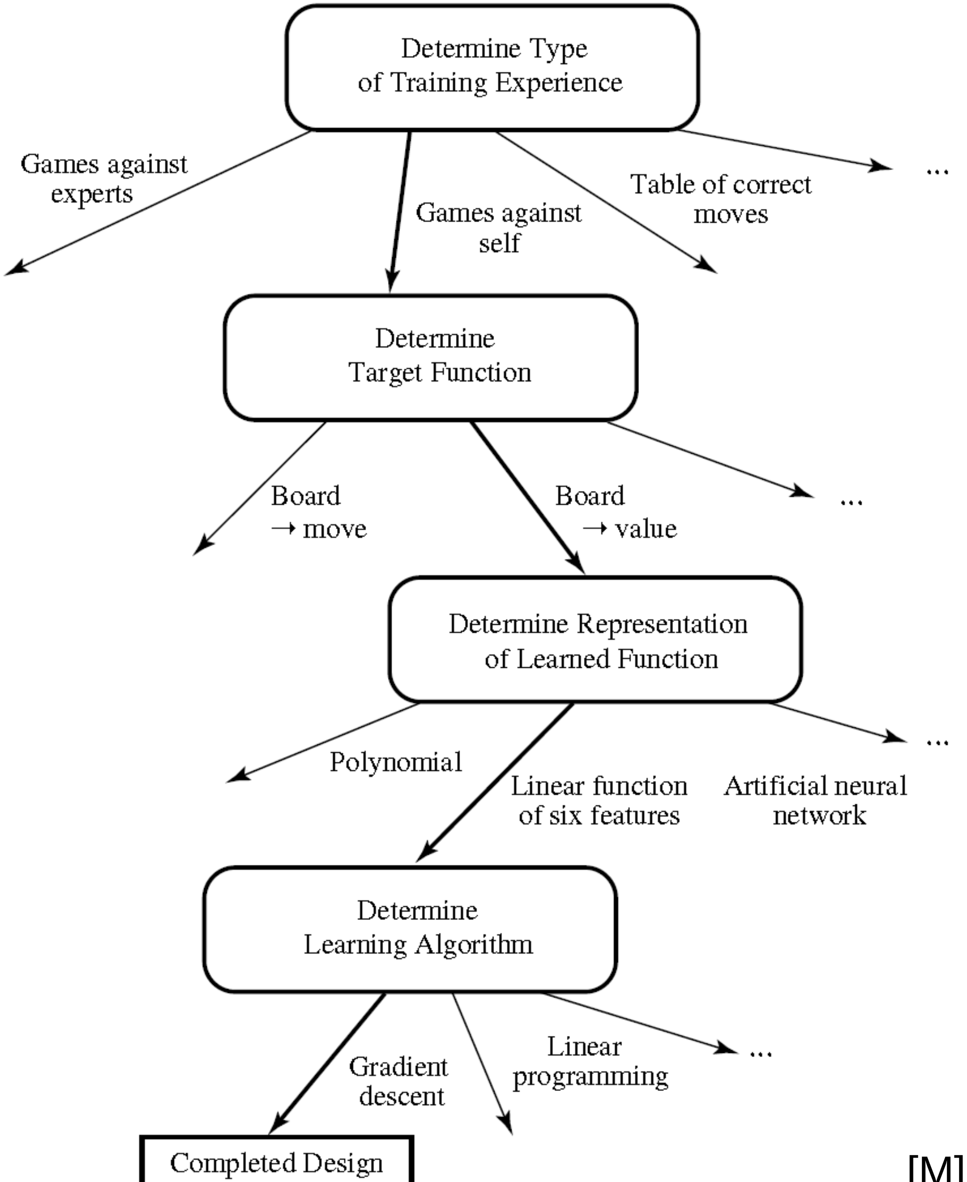
ε is some small constant to moderate the rate of learning (e.g., 0.1).

The LMS rule optimizes the mean square error function E locally:

$$E = \sum_{\{\text{All training samples } (b,\, V_{train}(b))\}} (V_{train}(b) - V'(b))^2$$

Design choices

For the learning system, we need to make the following design choices:



[M]

Relevant disciplines in ML

- Artificial intelligence
- Statistics
- Bayesian methods
- Computational complexity theory
- Control theory
- Information theory
- Neurobiology
- Psychology

Issues in ML

- What algorithms can approximate functions well (and when) ?
- How does the number of training examples influence accuracy?
- How can we get training examples?
- How does complexity of hypothesis representation impact accuracy?
- How does noisy data influence accuracy?
- What are the theoretical limits of learning?
- How can prior knowledge help?
- What clues can we get from biological learning systems?
- Interaction of unsupervised and supervised techniques
- Hybrid systems: Integrating ML and explicit models
  - on the representational level
  - for learning

ML and data mining

What is data mining (DM) and what is the relation to ML?

So far:

- ML sounds good, machines can learn everything from examples
- We don't need to model anymore

But:

- We have to supply the examples
- Example acquisition may be heavy, e.g.
  - Visual training of objects may require manual segmentation or lab setup
  - Training of audio samples even more difficult: either lab setup or "segmentation" requires solution of a complex separation problem ("party problem")

ML and data mining

- Examples need to be “typical”
- Examples should exhibit adequate distribution
- Examples must not show “clutter”
- Examples require labeling
- Labeling not straight forward, e.g.
  - Object recognition: Identity, pose (numerical or qualitative description?)
  - Which scene category is adequate: *Car, red car, Porsche, sports car, fun vehicle, fun, nonsense, pollution, youth, midlife crisis*?
  - Particularly difficult: Scene labeling
    - Identification of components vs. clutter
    - Meaning of a scene
    - Human categorization differs strongly

ML and data mining

- Exhaustive representation by examples (e.g. object on turntable in steps of 5 deg.) is
  - impractical
  - not cognitively adequate
- How does nature solve the problem?

➔ Child learns many concepts without help (e.g., *car*) and gets the label ("*car*") later

- So we should supply concepts wherever possible
- Two ways out of the dilemma:
  1. Hybrid systems: Fuse explicit modeling with ML
  2. Learn as much as possible by **unsupervised** algorithms, use expensively labeled examples sparingly

ML and data mining

- Unsupervised learning solves the labeling part of the above
- The rest remains (e.g., supplying adequate statistics) but can be handled more easily
- Up to here we have the part of the DM motivation connected to *learning*
- There is another motivation for DM: Exploration !
- Introductory DM examples: using historical data to improve decisions, e.g.,
  - medical records ➔ medical knowledge
  - credit risk analysis
  - customer purchase behavior
  - customer retention
  - process optimization

Typical data mining tasks

Data:

Patient103 time=1 →	*Patient103* time=2 ... →	*Patient103* time=n
Age: 23	Age: 23	Age: 23
FirstPregnancy: no	FirstPregnancy: no	FirstPregnancy: no
Anemia: no	Anemia: no	Anemia: no
Diabetes: no	Diabetes: YES	Diabetes: no
PreviousPrematureBirth: no	PreviousPrematureBirth: no	PreviousPrematureBirth: no
Ultrasound: ?	Ultrasound: abnormal	Ultrasound: ?
Elective C−Section: ?	Elective C−Section: no	Elective C−Section: no
Emergency C−Section: ?	Emergency C−Section: ?	**Emergency C−Section: Yes**
...	...	...

[M]

Data [M]:

- 9714 patient records, each describing a pregnancy and birth
- Each patient record contains 215 features

Task:

Find classes of patients at high risk for emergency C-section !

Typical data mining tasks

Result:

One of 18 learned rules:

```
If    No previous vaginal delivery, and
      Abnormal 2nd Trimester Ultrasound, and
      Malpresentation at admission
Then Probability of Emergency C-Section is 0.6

 Over training data: 26/41 = .63,
 Over test data: 12/20 = .60
```

[M]

Typical data mining tasks

Credit risk analysis:

Customer103: (time=t0)

Years of credit: 9
Loan balance: $2,400
Income: $52k
Own House: Yes
Other delinquent accts: 2
Max billing cycles late: 3
Profitable customer?: ?
...

Customer103: (time=t1)

Years of credit: 9
Loan balance: $3,250
Income: ?
Own House: Yes
Other delinquent accts: 2
Max billing cycles late: 4
Profitable customer?: ?
...

...

Customer103: (time=tn)

Years of credit: 9
Loan balance: $4,500
Income: ?
Own House: Yes
Other delinquent accts: 3
Max billing cycles late: 6
Profitable customer?: No
...

[M]

Rules learned from data:

```
If   Other-Delinquent-Accounts > 2, and
     Number-Delinquent-Billing-Cycles > 1
Then Profitable-Customer? = No
     [Deny Credit Card application]

If   Other-Delinquent-Accounts = 0, and
     (Income > $30k)  OR  (Years-of-Credit > 3)
Then Profitable-Customer? = Yes
     [Accept Credit Card application]
```

[M]

Typical data mining tasks

Customer purchase behavior:

Customer103: (time=t0)	*Customer103:* (time=t1)	...	*Customer103:* (time=tn)
Sex: M	Sex: M		Sex: M
Age: 53	Age: 53		Age: 53
Income: $50k	Income: $50k		Income: $50k
Own House: Yes	Own House: Yes		Own House: Yes
MS Products: Word	MS Products: Word		MS Products: Word
Computer: 386 PC	Computer: Pentium		Computer: Pentium
Purchase Excel?: ?	Purchase Excel?: ?		**Purchase Excel?: Yes**
...	...		...

[M]

Typical data mining tasks

Customer retention:

Customer103: (time=t0)	*Customer103:* (time=t1)	...	*Customer103:* (time=tn)
Sex: M	Sex: M		Sex: M
Age: 53	Age: 53		Age: 53
Income: $50k	Income: $50k		Income: $50k
Own House: Yes	Own House: Yes		Own House: Yes
Checking: $5k	Checking: $20k		Checking: $0
Savings: $15k	Savings: $0		Savings: $0
Current−customer?: yes	Current−customer?: yes		**Current−customer?: No**

[M]

Typical data mining tasks

Process optimization:

Product72: (time=t0)	*Product72:* (time=t1)	...	*Product72:* (time=tn)
Stage: mix	Stage: cook		Stage: cool
Mixing-speed: 60rpm	Temperature: 325		Fan-speed: medium
Viscosity: 1.3	Viscosity: 3.2		Viscosity: 1.3
Fat content: 15%	Fat content: 12%		Fat content: 12%
Density: 2.8	Density: 1.1		Density: 1.2
Spectral peak: 2800	Spectral peak: 3200		Spectral peak: 3100
Product underweight?: ??	Product underweight?: ??		**Product underweight?: Yes**
...	...		...

[M]

In summary:

We are looking for *generic* DM methods that can be equally applied to problems like risk analysis in medicine, credit risk analysis, prediction of purchase behaviour or customer retention, and process analysis.

Data mining

- "*Knowledge Discovery in Databases*" – KDD
- A bit of history:
  - 1989 first KDD workshop by AAAI (Association for the Advancement of AI)
  - 1995 first international KDD conference at IJCAI
  - 1997 first journal
  - Today „Big Data"
- General idea: *Convert data to knowledge !*
  - Finding patterns, regularities, anomalies
  - Automatic detection of correlations
  - Trend detection
  - Prediction

Data mining

  - Rule extraction
  - Automated modeling
- Different line: Making data accessible to humans
  - Data visualization
  - Data sonification
  - HCI for “navigation” in data

Image sources

[M] Online material available at www.cs.cmu.edu/~tom/mlbook.html
for the textbook: Tom M. Mitchell: *Machine Learning*, McGraw-Hill