UNIVERSITÄT PADERBORN
Die Universität der Informationsgesellschaft

MASCHINELLES LERNEN: ÜBER DATEN, WISSEN UND ECORITHMEN

Eyke Hüllermeier
Department of Computer Science
Paderborn University

eyke@upb.de

"Machine learning is the science and art of algorithms that make sense of data."

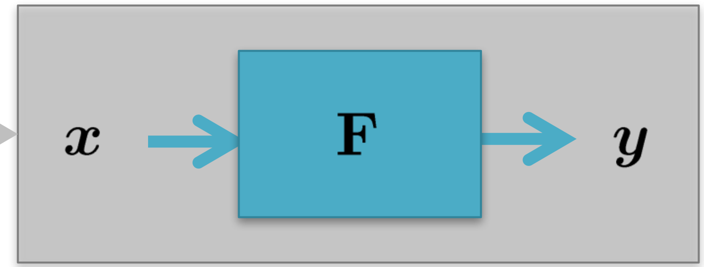
Peter Flach, 2012

"Machine learning is the science of getting computers to act without being explicitly programmed."

Andrew Ng, 2013

THE ALGORITHMIC APPROACH

algorithm

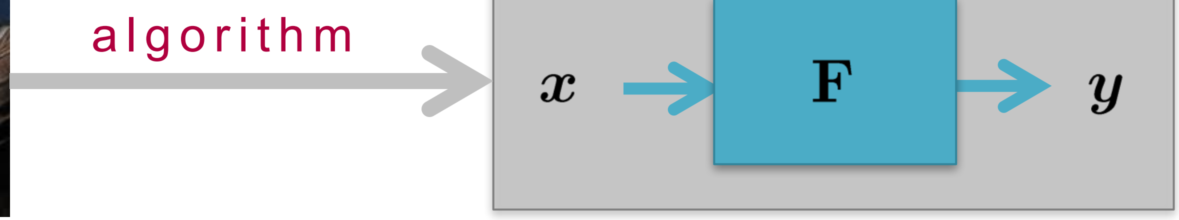


domain expert = programmer

```
ALGORITHM shortest-path(V,T)
 W := {v1}
 ShortDist[v1] :=0
 FOR each u in V - {v1}
     ShortDist[u] := T[v1,u]
 WHILE W /= V
       MinDist := INFINITE
       FOR each v in V - W
           IF ShortDist[v] < MinDist
              MinDist = ShortDist[v]
              w := v
           END {if}
       END {for}
       W := W U {w}
       FOR each u in V - W
           ShortDist[u] := Min(ShorDis[u],ShortDist[w] + T[w,u])
 END {while}
```

THE ALGORITHMIC APPROACH

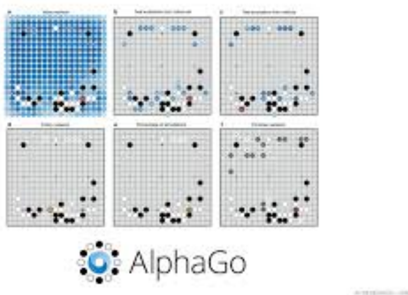
domain expert = programmer



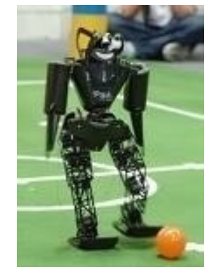
*Requires a **comprehensive understanding** and adequate formalization, not only of the problem, but also **of the solution process**.*

COMPLEX PROBLEMS

GAME PLAYING



ROBOT SOCCER



state vector describing the environment

$$x \rightarrow \mathbf{F} \rightarrow y$$

action vector

→ MALE

technology and science news — **19 September 2013**

The End of Driving?

A chorus of carmakers has declared that they expect autonomous cars to reach commercial viability by 2020. Computer systems and sensors that handle parking, braking, and to a limited degree, steering are already giving us a glimpse of a future in which machines not only drive unassisted but do so better than any human can. Now Tesla Motors, maker of the eponymous electric luxury sports car that debuted to rave reviews, has upped the ante. Tesla's CEO, Elon Musk, says that within the next three years, his company aims to produce systems capable of safely taking the helm for 90 percent of miles driven.

IMAGE RECOGNITION

AUTONOMOUS CARS

THE ALGORITHMIC APPROACH

INTELLIGENT SYSTEMS

> "Our problem then is to find out how to programme these machines to [behave intelligently]. At my present rate of working I produce about a thousand digits of programme a day, so that about sixty workers, working steadily through the fifty years might accomplish the job, if nothing went into the waste-paper basket. **Some more expeditious** method seems desirable."

Alan Turing, Computing Machinery and Intelligence, 1950

Goal of **automated programming** ever since (e.g. Turing Award Lecture by Jim Gray, 1999)

IMPLICIT SKILLS

Human skills are not always easy to explain!

$$\rightarrow \boxed{F} \rightarrow$$

MALE
OR
FEMALE

$$\boldsymbol{x} \in \mathbb{R}^N \qquad \boldsymbol{y} \in \{0, 1\}$$

Human skills are not always easy to explain!

Optimal Sample Complexity of M-wise Data for Top-K Ranking

Algorithm 1 Rank Centrality (Negahban et al., 2012)

Input the collection of statistics $\boldsymbol{s} = \{s_{\mathcal{I}} : \mathcal{I} \in \mathcal{E}^{(M)}\}$.
Convert the M-wise sample for each hyper-edge $\mathcal{I}$ into M pairwise samples:

1. Choose a circular permutation of the items in $\mathcal{I}$ uniformly at random,
2. Break it into the M pairs of adjacent items, and denote the set of pairs by $\phi(\mathcal{I})$,
3. Use the (pairwise) data of the pairs in $\phi(\mathcal{I})$.

Compute the transition matrix $\hat{\boldsymbol{P}} = [\hat{P}_{ij}]_{1 \le i,j \le n}$:

$$\hat{P}_{ij} = \begin{cases} \frac{1}{2d_{\max}} y_{ij} & \text{if } i \neq j; \\ 1 - \sum_{k:k \neq j} \hat{P}_{kj} & \text{if } i = j; \\ 0 & \text{otherwise,} \end{cases}$$

where $d_{\max}$ is the maximum out-degree of vertices in $\mathcal{E}$.
Output the stationary distribution of matrix $\hat{\boldsymbol{P}}$.

$$y_{ij} := \sum_{\mathcal{I}:\{i,j\} \in \phi(\mathcal{I})} \frac{1}{L} \sum_{\ell=1}^{L} y_{ij,\mathcal{I}}^{(\ell)}. \tag{16}$$

In an ideal scenario where we obtain an infinite number of samples per M-wise comparison, i.e., $L \to \infty$, sufficient statistics $\frac{1}{L}\sum_{l=1}^{L} y_{ij,\mathcal{I}}^{(\ell)}$ converge to $\frac{w_i}{w_i + w_j}$, as the PL model is a natural generalized version of the BTL model. Then, the constructed matrix $\hat{\boldsymbol{P}}$ defined in Algorithm 1 becomes a matrix $\boldsymbol{P}$ whose entries $[P_{ij}]_{1 \le i,j \le n}$ are defined as

$$P_{ij} = \begin{cases} \frac{1}{2d_{\max}} \sum_{\mathcal{I}:\{i,j\} \in \phi(\mathcal{I})} \frac{w_i}{w_i + w_j} & \text{for } \mathcal{I} \in \mathcal{E}^{(M)}; \\ 1 - \sum_{k:k \neq j} P_{kj} & \text{if } i = j; \\ 0 & \text{otherwise.} \end{cases} \tag{17}$$

The entries for observed item pairs represent the relative likelihood of item i being preferred over item j. Intuitively, random walks of $\boldsymbol{P}$ in the long run visit some states more often, if they have been preferred over other frequently-visited states and/or preferred over many other states.

The random walks are reversible as $w_i P_{ji} = w_j P_{ij}$ holds, and irreducible under the connectivity assumption. Once we obtain the unique stationary distribution, it is equal to $\boldsymbol{w} = \{w_1, \ldots, w_n\}$ up to some constant scaling.

It is clear that random walks of $\hat{\boldsymbol{P}}$, a noisy version of $\boldsymbol{P}$, will give us an approximation of $\boldsymbol{w}$. The algorithm adopts a power method, known to be computationally efficient in obtaining the leading eigenvalue of a sparse matrix (Meirovitch, 1997), to obtain the stationary distribution.

et al., 2013) directly follows the ordering evaluated in each sample; if it is $1 \prec 2 \prec \cdots \prec M-1 \prec M$, it is broken into pairs of adjacent items: $1 \prec 2$ up to $M-1 \prec M$. Our method turns out to be consistent, i.e., $\frac{\Pr[y_{ij}=1]}{\Pr[y_{ji}=0]} = \frac{w_i}{w_j}$ (see (17)), whereas the adjacent breaking method is not (Azari Soufiani et al., 2013).

3.2. Proof outline

To outline the proof of Theorem 2, let us introduce Theorem 3. We show that Theorem 3 leads to Theorem 2.

Theorem 3. *When* Rank Centrality *is employed, with high probability, the ℓ_∞ norm estimation error is upper-bounded by*

$$\frac{\|\hat{\boldsymbol{w}} - \boldsymbol{w}\|_\infty}{\|\boldsymbol{w}\|_\infty} \lesssim \sqrt{\frac{n \log n}{\binom{n}{M} pL}} \sqrt{\frac{1}{M}}, \tag{18}$$

where $p \ge c_1 (M-1) \sqrt{\frac{\log n}{\binom{n-1}{M-1}}}$, and c_1 is some numerical constant.

Let $\|\boldsymbol{w}\|_\infty = w_{\max} = 1$ for ease of demonstration. Suppose $\Delta_K = w_K - w_{K+1} \gtrsim \sqrt{\frac{\log n}{\binom{n}{M} pL}} \sqrt{\frac{1}{M}}$. Then,

$$\begin{aligned} \hat{w}_i - \hat{w}_j &\ge w_i - w_j - |\hat{w}_i - w_i| - |\hat{w}_j - w_j| \\ &\ge w_K - w_{K+1} - 2\|\hat{\boldsymbol{w}} - \boldsymbol{w}\|_\infty > 0, \end{aligned} \tag{19}$$

for all $1 \le i \le K$ and $j \ge K+1$. That is, the top-K items are identified as desired. Hence, as long as $\Delta_K \gtrsim \sqrt{\frac{\log n}{\binom{n}{M} pL}} \sqrt{\frac{1}{M}}$, i.e., $\binom{n}{M} pL \gtrsim \frac{n \log n}{\Delta_K^2} \frac{1}{M}$, reliable top-$K$ ranking is achieved with the sample size of $\frac{n \log n}{\Delta_K^2} \frac{1}{M}$.

Now, let us prove Theorem 3. To find an ℓ_∞ error bound, we first derive an upper bound on the point-wise error between the score estimate of item i and its true score, which consists of three terms:

$$\begin{aligned} |\hat{w}_i - w_i| \le |\hat{w}_i - w_i| \hat{P}_{ii} &+ \sum_{j:j \neq i} |\hat{w}_j - w_j| \hat{P}_{ij} \\ &+ \left| \sum_{j:j \neq i} (w_i + w_j) \left(\hat{P}_{ji} - P_{ji} \right) \right|. \end{aligned} \tag{20}$$

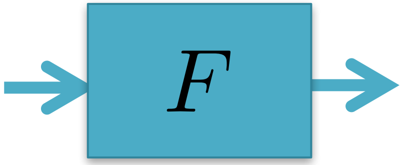
This can be obtained applying $\hat{\boldsymbol{w}} = \hat{\boldsymbol{P}} \hat{\boldsymbol{w}}$ and $\boldsymbol{w} = \boldsymbol{P}\boldsymbol{w}$. We obtain upper bounds on these three terms as follows.

$$\hat{P}_{ii} < 1, \tag{21}$$

$$\left| \sum_{j:j \neq i} (w_i + w_j) \left(\hat{P}_{ji} - P_{ji} \right) \right| \lesssim \sqrt{\frac{n \log n}{\binom{n}{M} pL}} \sqrt{\frac{1}{M}}, \tag{22}$$

$$\sum_{j:j \neq i} |\hat{w}_j - w_j| \hat{P}_{ij} \lesssim \sqrt{\frac{n \log n}{\binom{n}{M} pL}} \sqrt{\frac{1}{M}}, \tag{23}$$

with high probability (see Lemmas 1, 2 and 3 in the supplementary for details). One can see that the inequalities (21)



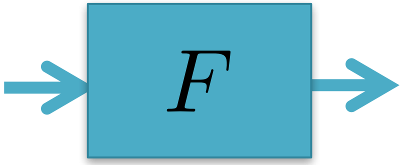
Abstract

Given a sample of instances with binary labels, the top ranking problem is to produce a ranked list of instances where the *head* of the list is dominated by positives. Popular existing approaches to this problem are based on surrogates to a performance measure known as the fraction of positives of the top (PTop). In this paper, we show that the measure and its surrogates have an undesirable property: for certain noisy distributions, it is optimal to trivially predict *the same score for all instances*. We propose a simple rectification of the measure which avoids such trivial solutions, while still focussing on the head of the ranked list and being as easy to optimise.

Human skills are not always easy to explain!

Abstract

Given a sample of instances with binary labels, the top ranking problem is to produce a ranked list of instances where the *head* of the list is dominated by positives. Popular existing approaches to this problem are based on surrogates to a performance measure known as the fraction of positives of the top (PTop). In this paper, we show that the measure and its surrogates have an undesirable property: for certain noisy distributions, it is optimal to trivially predict *the same score for all instances*. We propose a simple rectification of the measure which avoids such trivial solutions, while still focussing on the head of the ranked list and being as easy to optimise.



Optimal Sample Complexity of M-wise Data for Top-K Ranking

Algorithm 1 Rank Centrality (Negahban et al., 2012)

Input the collection of statistics $\boldsymbol{s} = \{s_{\mathcal{I}} : \mathcal{I} \in \mathcal{E}^{(M)}\}$.
Convert the M-wise sample for each hyper-edge $\mathcal{I}$ into M pairwise samples:

1. Choose a circular permutation of the items in $\mathcal{I}$ uniformly at random,
2. Break it into the M pairs of adjacent items, and denote the set of pairs by $\phi(\mathcal{I})$,
3. Use the (pairwise) data of the pairs in $\phi(\mathcal{I})$.

Compute the transition matrix $\hat{\boldsymbol{P}} = [\hat{P}_{ij}]_{1\le i,j\le n}$:

$$\hat{P}_{ij} = \begin{cases} \frac{1}{2d_{\max}} y_{ij} & \text{if } i \neq j; \\ 1 - \sum_{k:k\neq j} \hat{P}_{kj} & \text{if } i = j; \\ 0 & \text{otherwise,} \end{cases}$$

where $d_{\max}$ is the maximum out-degree of vertices in $\mathcal{E}$.
Output the stationary distribution of matrix $\hat{\boldsymbol{P}}$.

$$y_{ij} := \sum_{\mathcal{I}:\{i,j\}\in\phi(\mathcal{I})} \frac{1}{L} \sum_{\ell=1}^{L} y_{ij,\mathcal{I}}^{(\ell)}. \tag{16}$$

In an ideal scenario where we obtain an infinite number of samples per M-wise comparison, i.e., $L \to \infty$, sufficient statistics $\frac{1}{L}\sum_{l=1}^{L} y_{ij,\mathcal{I}}^{(\ell)}$ converge to $\frac{w_i}{w_i+w_j}$, as the PL model is a natural generalized version of the BTL model. Then, the constructed matrix $\hat{\boldsymbol{P}}$ defined in Algorithm 1 becomes a matrix $\boldsymbol{P}$ whose entries $[P_{ij}]_{1\le i,j\le n}$ are defined as

$$P_{ij} = \begin{cases} \frac{1}{2d_{\max}} \sum_{\mathcal{I}:\{i,j\}\in\phi(\mathcal{I})} \frac{w_i}{w_i+w_j} & \text{for } \mathcal{I} \in \mathcal{E}^{(M)}; \\ 1 - \sum_{k:k\neq j} P_{kj} & \text{if } i = j; \\ 0 & \text{otherwise.} \end{cases} \tag{17}$$

The entries for observed item pairs represent the relative likelihood of item i being preferred over item j. Intuitively, random walks of $\boldsymbol{P}$ in the long run visit some states more often, if they have been preferred over other frequently-visited states and/or preferred over many other states.

The random walks are reversible as $w_i P_{ji} = w_j P_{ij}$ holds, and irreducible under the connectivity assumption. Once we obtain the unique stationary distribution, it is equal to $\boldsymbol{w} = \{w_1, \dots, w_n\}$ up to some constant scaling.

It is clear that random walks of $\hat{\boldsymbol{P}}$, a noisy version of $\boldsymbol{P}$, will give us an approximation of $\boldsymbol{w}$. The algorithm adopts a power method, known to be computationally efficient in obtaining the leading eigenvalue of a sparse matrix (Meirovitch, 1997), to obtain the stationary distribution.

et al., 2013) directly follows the ordering evaluated in each sample; if it is $1 \prec 2 \prec \cdots \prec M-1 \prec M$, it is broken into pairs of adjacent items: $1 \prec 2$ up to $M-1 \prec M$. Our method turns out to be consistent, i.e., $\frac{\Pr[y_{ij}=1]}{\Pr[y_{ji}=0]} = \frac{w_i}{w_j}$ (see (17)), whereas the adjacent breaking method is not (Azari Soufiani et al., 2013).

3.2. Proof outline

To outline the proof of Theorem 2, let us introduce Theorem 3. We show that Theorem 3 leads to Theorem 2.

Theorem 3. *When* Rank Centrality *is employed, with high probability, the ℓ_∞ norm estimation error is upper-bounded by*

$$\frac{\|\hat{\boldsymbol{w}} - \boldsymbol{w}\|_\infty}{\|\boldsymbol{w}\|_\infty} \lesssim \sqrt{\frac{n\log n}{\binom{n}{M} pL}} \sqrt{\frac{1}{M}}, \tag{18}$$

where $p \ge c_1(M-1)\sqrt{\frac{\log n}{\binom{n-1}{M-1}}}$, and c_1 is some numerical constant.

Let $\|\boldsymbol{w}\|_\infty = w_{\max} = 1$ for ease of demonstration. Suppose $\Delta_K = w_K - w_{K+1} \gtrsim \sqrt{\frac{\log n}{\binom{n}{M}pL}}\sqrt{\frac{1}{M}}$. Then,

$$\begin{aligned} \hat{w}_i - \hat{w}_j &\ge w_i - w_j - |\hat{w}_i - w_i| - |\hat{w}_j - w_j| \\ &\ge w_K - w_{K+1} - 2\|\hat{\boldsymbol{w}} - \boldsymbol{w}\|_\infty > 0, \end{aligned} \tag{19}$$

for all $1 \le i \le K$ and $j \ge K+1$. That is, the top-K items are identified as desired. Hence, as long as $\Delta_K \gtrsim \sqrt{\frac{\log n}{\binom{n}{M}pL}}\sqrt{\frac{1}{M}}$, i.e., $\binom{n}{M}pL \gtrsim \frac{n\log n}{\Delta_K^2}\frac{1}{M}$, reliable top-$K$ ranking is achieved with the sample size of $\frac{n\log n}{\Delta_K^2}\frac{1}{M}$.

Now, let us prove Theorem 3. To find an ℓ_∞ error bound, we first derive an upper bound on the point-wise error between the score estimate of item i and its true score, which consists of three terms:

$$\begin{aligned} |\hat{w}_i - w_i| \le |\hat{w}_i - w_i|\,\hat{P}_{ii} &+ \sum_{j:j\neq i} |\hat{w}_j - w_j|\,\hat{P}_{ij} \\ &+ \left|\sum_{j:j\neq i} (w_i + w_j)\left(\hat{P}_{ji} - P_{ji}\right)\right|. \end{aligned} \tag{20}$$

This can be obtained applying $\hat{\boldsymbol{w}} = \hat{\boldsymbol{P}}\hat{\boldsymbol{w}}$ and $\boldsymbol{w} = \boldsymbol{P}\boldsymbol{w}$. We obtain upper bounds on these three terms as follows.

$$\hat{P}_{ii} < 1, \tag{21}$$

$$\left|\sum_{j:j\neq i} (w_i + w_j)\left(\hat{P}_{ji} - P_{ji}\right)\right| \lesssim \sqrt{\frac{n\log n}{\binom{n}{M}pL}}\sqrt{\frac{1}{M}}, \tag{22}$$

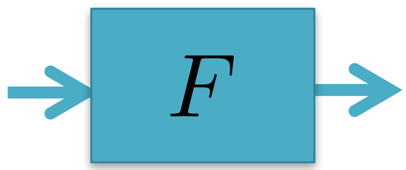
$$\sum_{j:j\neq i} |\hat{w}_j - w_j|\,\hat{P}_{ij} \lesssim \sqrt{\frac{n\log n}{\binom{n}{M}pL}}\sqrt{\frac{1}{M}}, \tag{23}$$

with high probability (see Lemmas 1, 2 and 3 in the supplementary for details). One can see that the inequalities (21)

IMPLICIT SKILLS

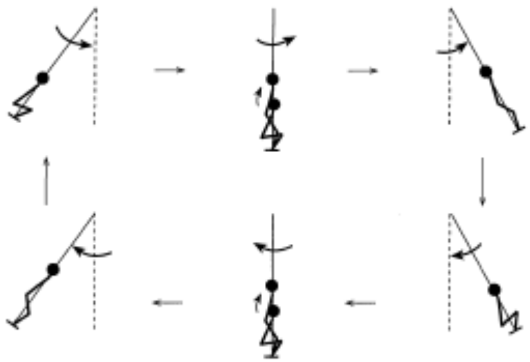
Human skills are not always easy to explain!

For example, a reduction of the search space does not immediately imply better solutions.

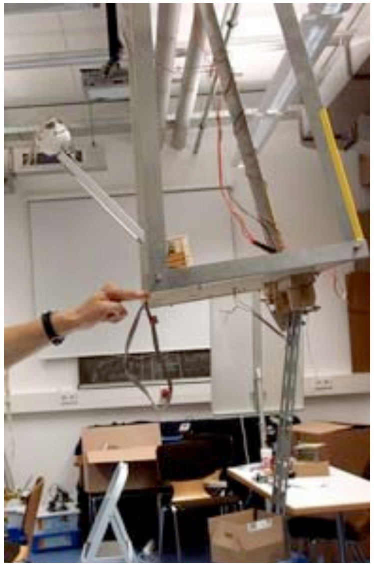


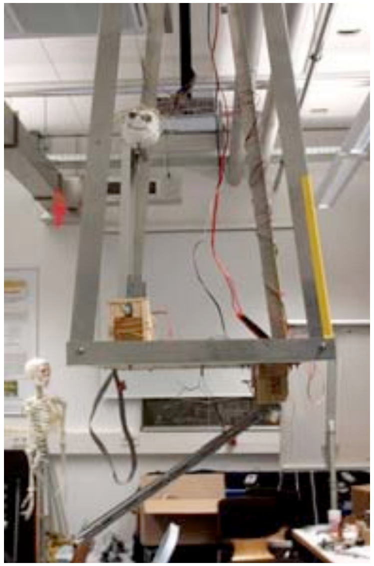
Eine Beschränkung des Suchraums führt beispielsweise nicht unmittelbar zu besseren Lösungen.

IMPLICIT SKILLS



How to design a swinging robot?





LEARNING FROM DATA

Instead of providing a complete and consistent description of domain knowledge, or designing a model by hand, it is easier to ...

– give **examples** and let the system **generalize**

→ supervised learning

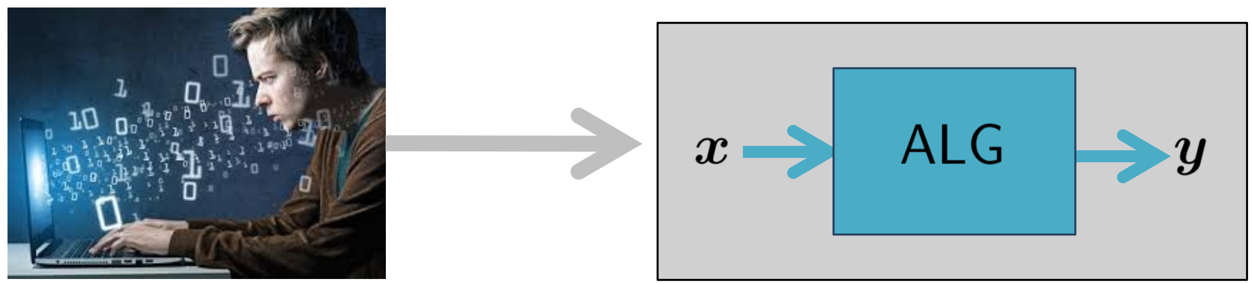
– let the system **explore** and provide **feedback**

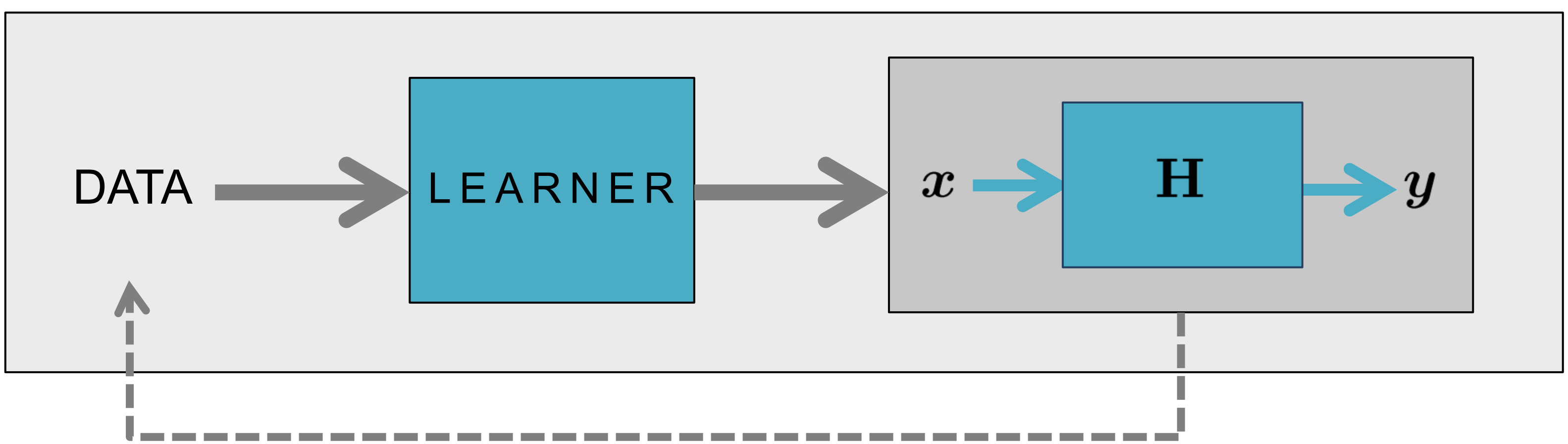
→ reinforcement learning

– **demonstrate** and let the system **imitate**

→ imitation learning

LEARNING FROM DATA



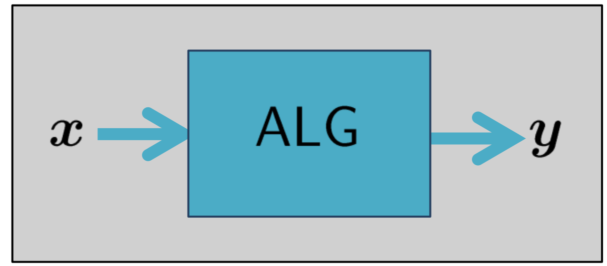


LEARNING FROM DATA

INTELLIGENT SYSTEMS

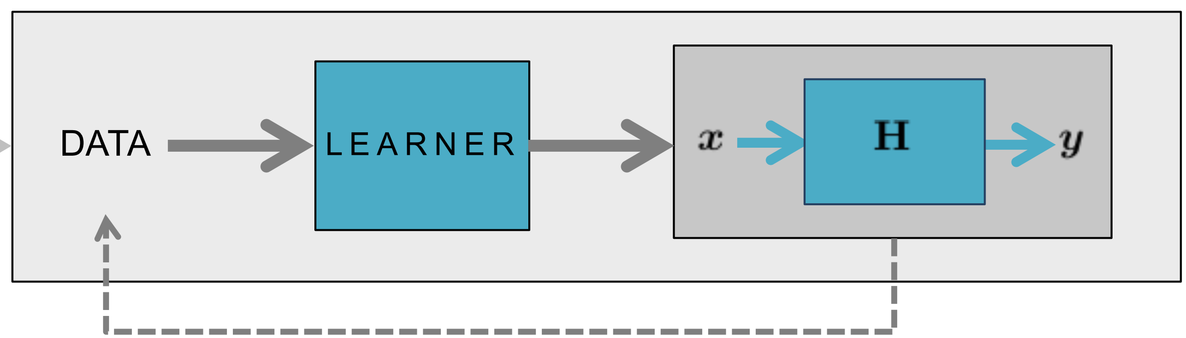
computer scientist

- correctness
- complexity (time, space)



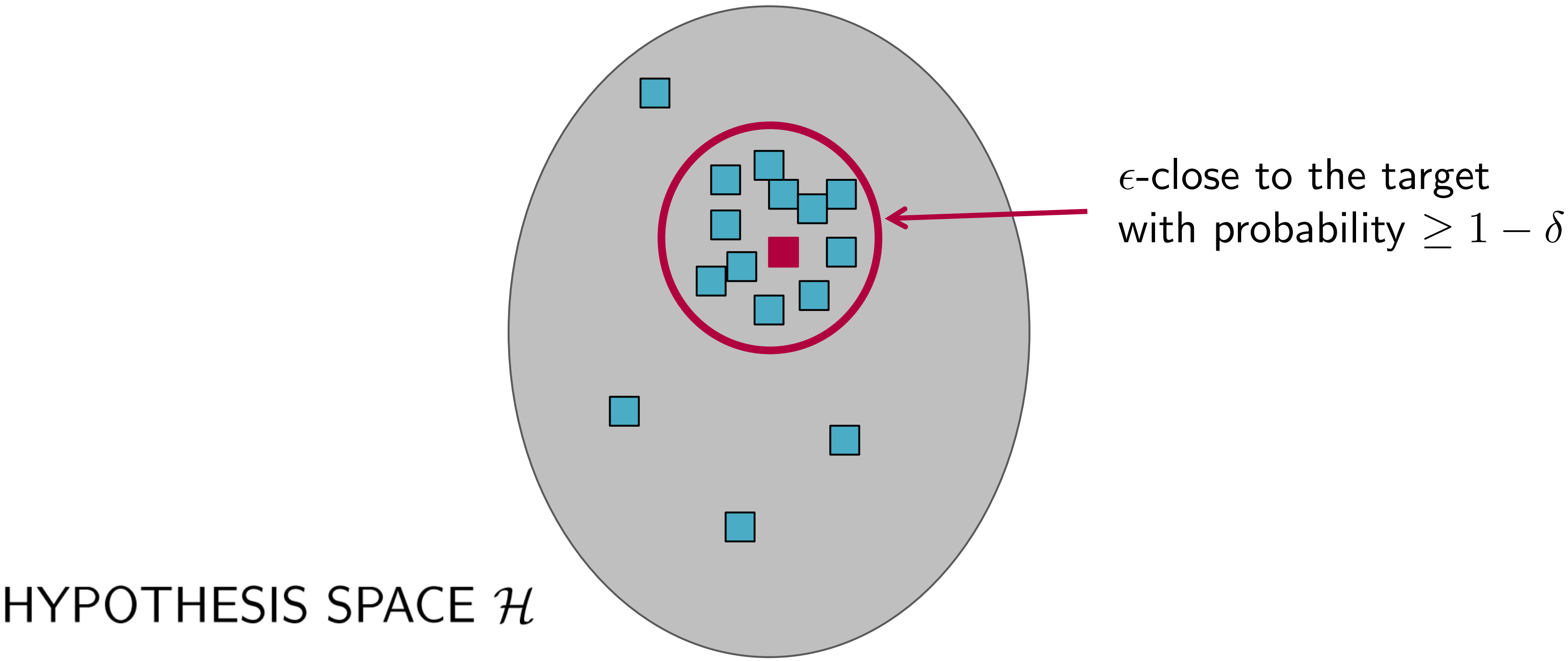
data scientist

- *correctness* (?)
- complexity (time, space)
- *sample complexity*



Probably Approximately Correct (PAC) learning:

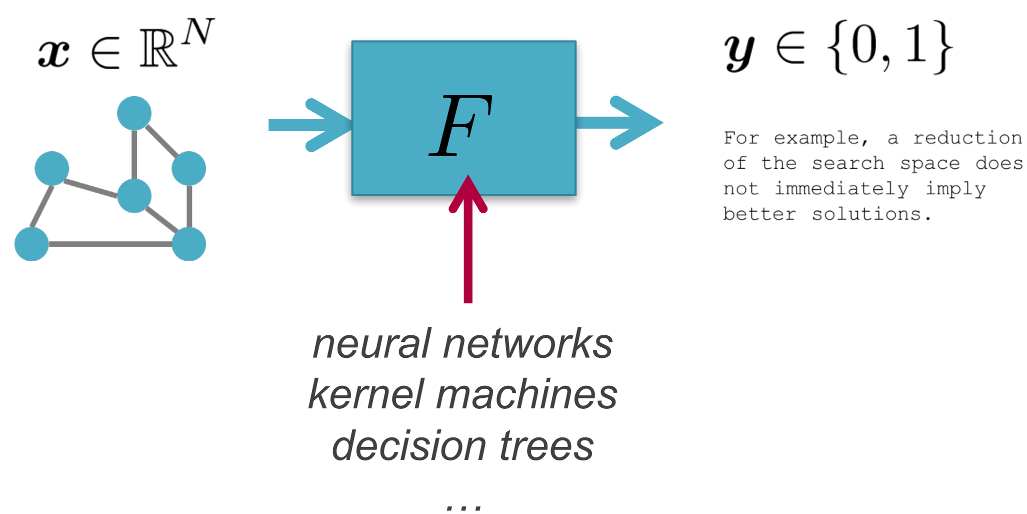
Efficiently finding a hypothesis that is „good“ with high probability!



ACM Turing Award 2010 for Leslie G. Valiant

LEARNING FROM DATA

Machine learning is an option whenever explicitly designing an algorithm by hand appears intricate, while **data** is available that provides, in one way or the other, **useful hints** at what the sought **functionality** may look like.



LEARNING FROM DATA

INTELLIGENT SYSTEMS

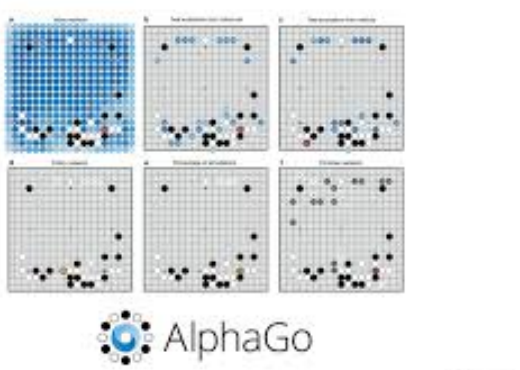
data is readily available or can easily be produced

make use of existing data

unsupervised learning

crowdsourcing

simulation, training through trial and error



reinforcement learning

amazon mechanical turk™
Artificial Artificial Intelligence

o male
o female

supervised learning

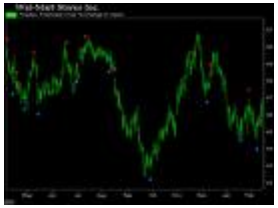
APPLICATIONS OF MACHINE LEARNING

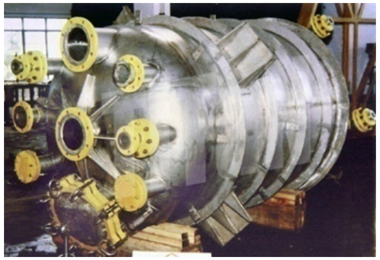
INTELLIGENT SYSTEMS

business (CRM, response prediction, …)

smart environments

banking and finance (stock prediction, fraud detection, …)

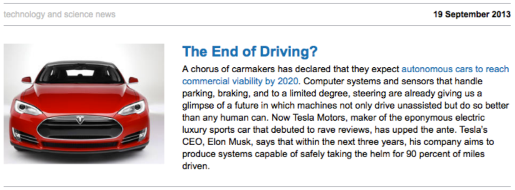




technical systems (diagnosis, control, monitoring, …)

Internet (information retrieval, email classification, personalization, …)

biometrics (person identification, …)



technology and science news — 19 September 2013

The End of Driving?

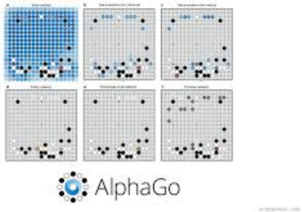
A chorus of carmakers has declared that they expect autonomous cars to reach commercial viability by 2020. Computer systems and sensors that handle parking, braking, and to a limited degree, steering are already giving us a glimpse of a future in which machines not only drive unassisted but do so better than any human can. Now Tesla Motors, maker of the eponymous electric luxury sports car that debuted to rave reviews, has upped the ante. Tesla's CEO, Elon Musk, says that within the next three years, his company aims to produce systems capable of safely taking the helm for 90 percent of miles driven.

autonmous driving

medicine (diagnosis, prosthetics, ...)

media (speech/image recognition, video mining, …)

bioinformatics, genomic data analysis



games (e.g. soccer, go, …)

ANALYTIC VERSUS SYNTHETIC ML

ANALYTIC VIEW


Polizei-Software zur Vorhersage von Verbrechen

Gesucht: Einbrecher der Zukunft

Ein Fall für den Algorithmus: In München überwachen Polizisten jetzt auch die Zukunft. Hier wird eine Software getestet, die berechnet, wann und wo der nächste Einbruch verübt wird. *Von Jannis Brühl und Florian Fuchs* mehr... 360°

…mazon files patent for "anticipatory" shipping

10 Comments / Shares / Tweets / Stumble / Email / More +

Amazon.com has filed for a patent for a shipping system that would anticipate what customers buy to decrease shipping time.

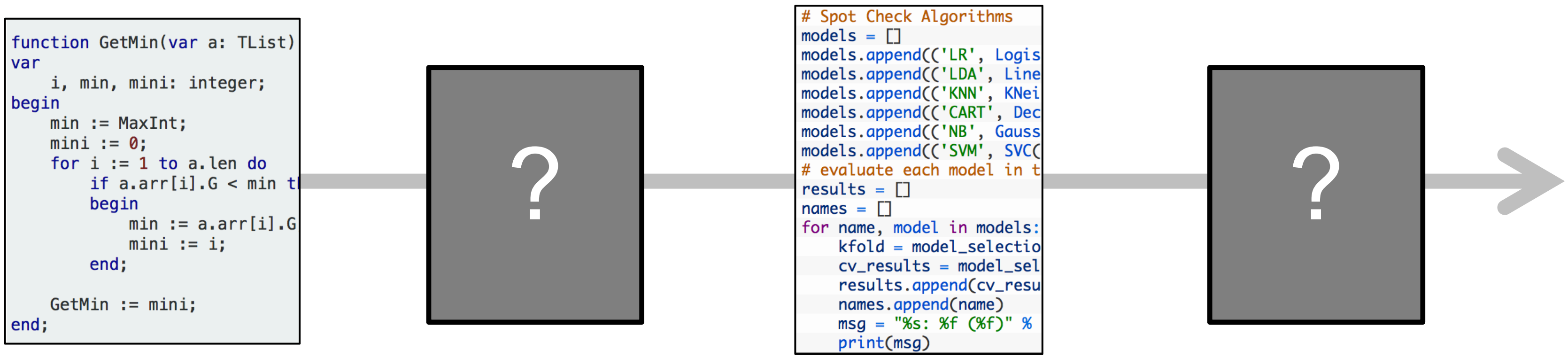
Amazon says the shipping system works by analyzing customer data like, purchasing history, product searches, wish lists and shopping cart contents, the Wall Street Journal reports. According to the patent filing, items would be moved from Amazon's fulfillment center to a shipping hub close to the customer in anticipation of an eventual purchase.

→ *analyze and help understand a phenomenon that exists in the real world*

SYNTHETIC VIEW

→ *support the design/engineering of a system with certain desirable properties*

ALGORITHMS AND ECORITHMS

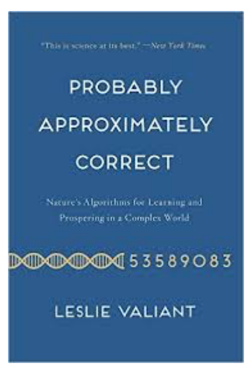


classical programming

"implicit" programming

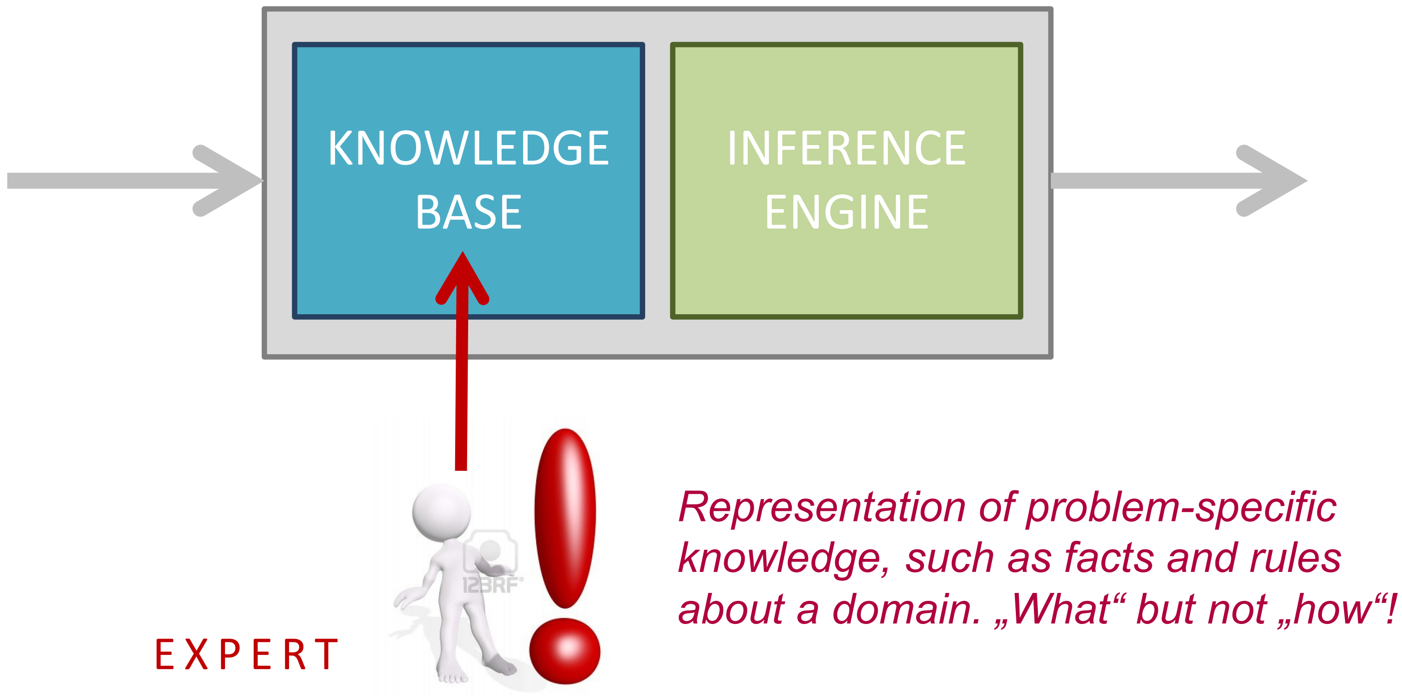
algorithm

ecorithm



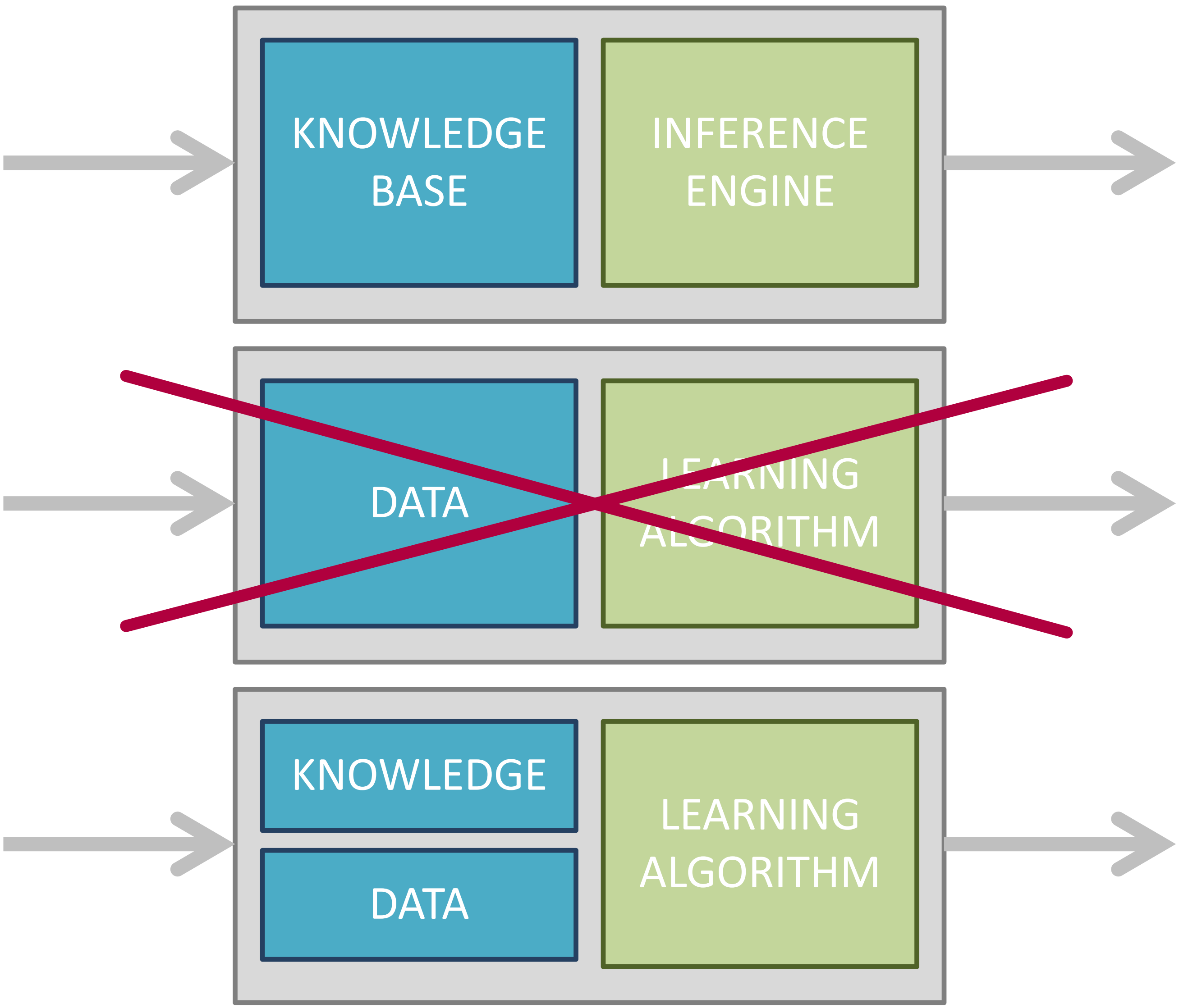
Leslie Valiant's broad term for an algorithm occurring in nature. An ecorithm is an algorithm "living" in and interacting with an external environment. Its goal is to perform well in that environment. Parallel to evolution of ecosystems.

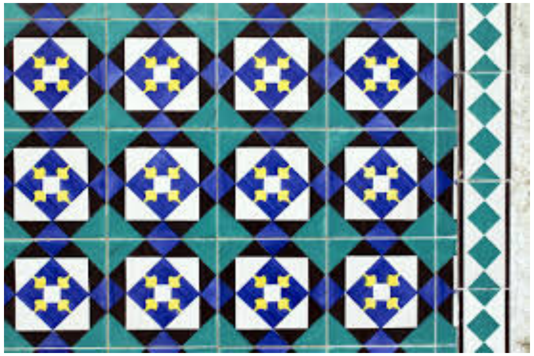
KNOWLEDGE-BASED PROGRAMMING

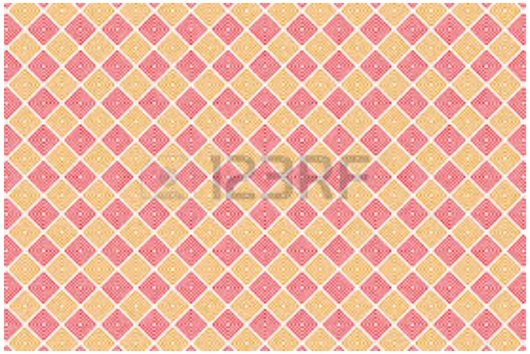


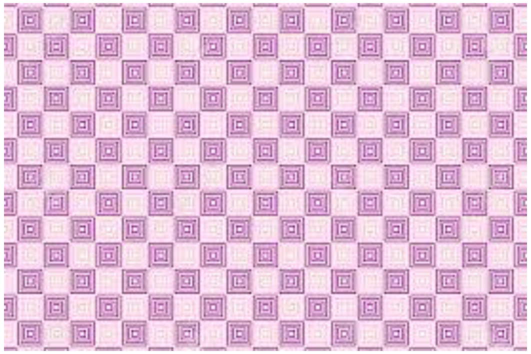
- Generic control structure implemented by the inference engine.
- programs = theories of a formal logic, computations = deductions
- Closely connected to declarative programming languages such as PROLOG.
- Appealing if it‘s difficult to explain HOW the problem is solved.

KNOWLEDGE-BASED PROGRAMMING





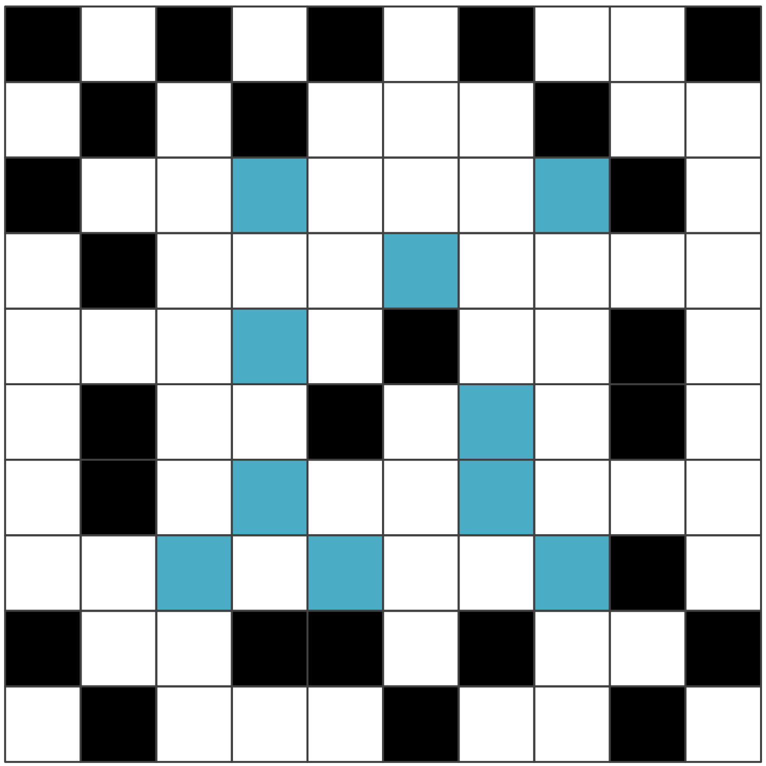




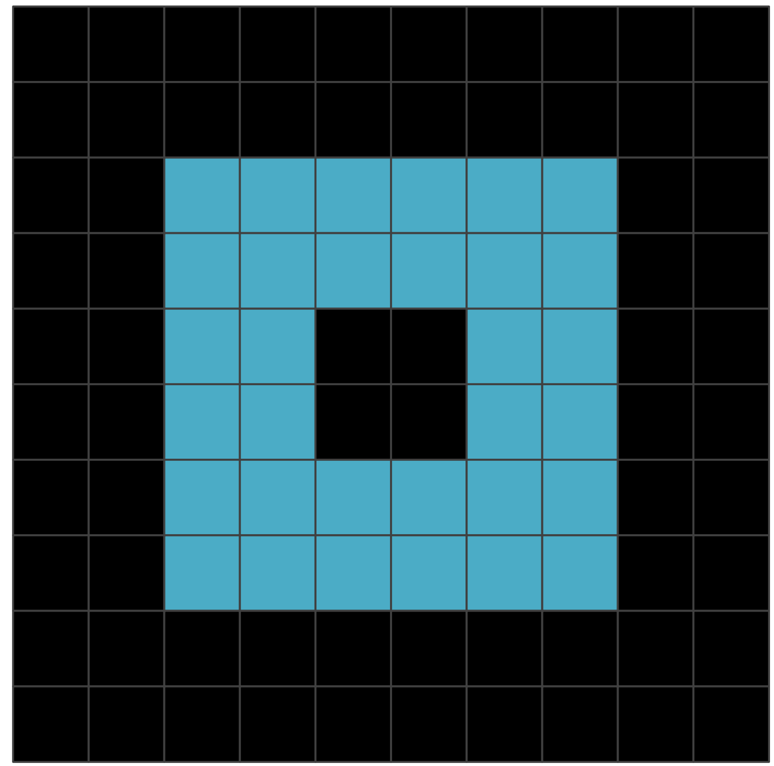
The world is regular …

KNOWLEDGE AND DATA

data



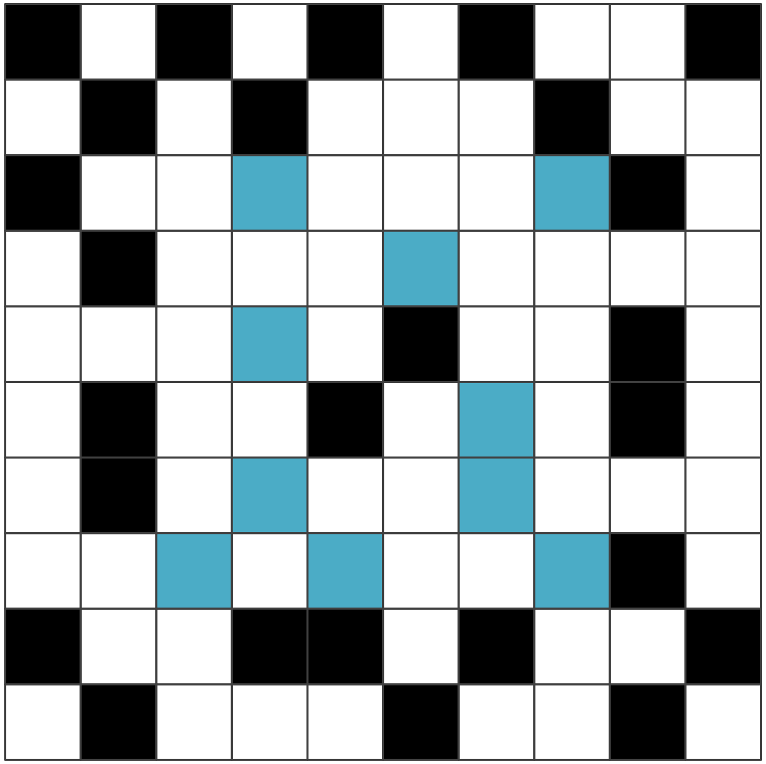
generalization



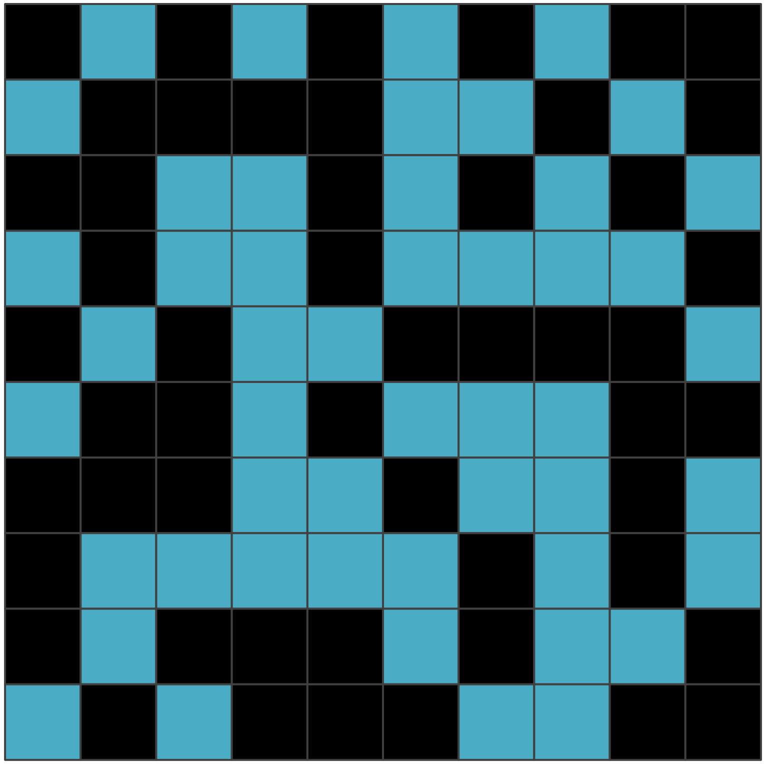
The world is regular …

INTELLIGENT
SYSTEMS

data

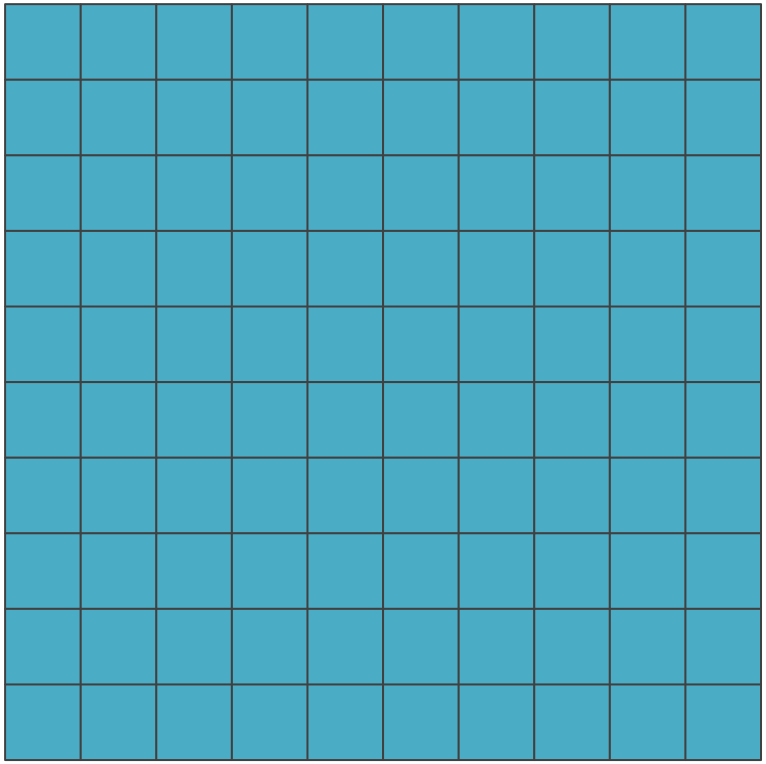


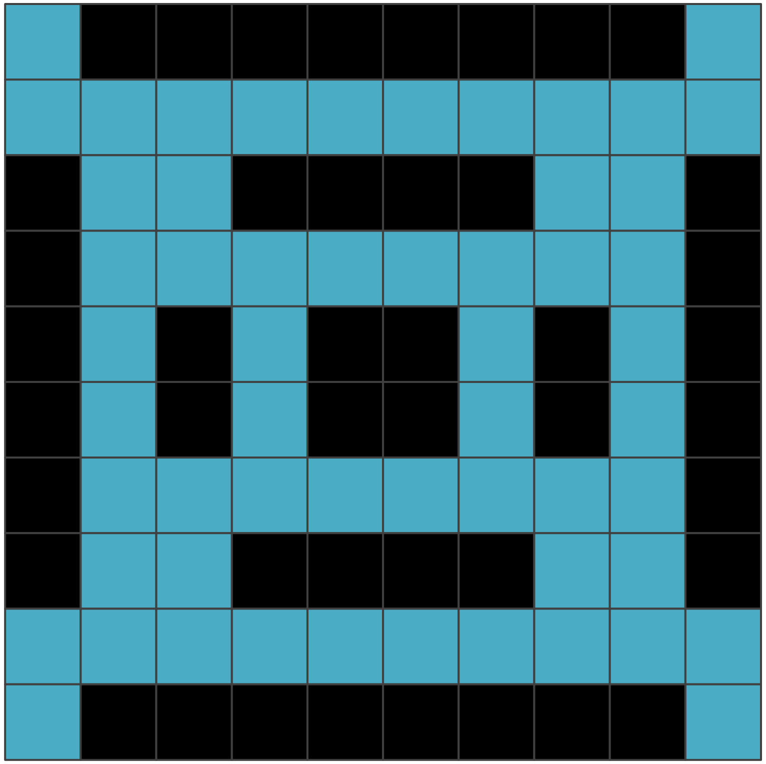
generalization



The world is regular …

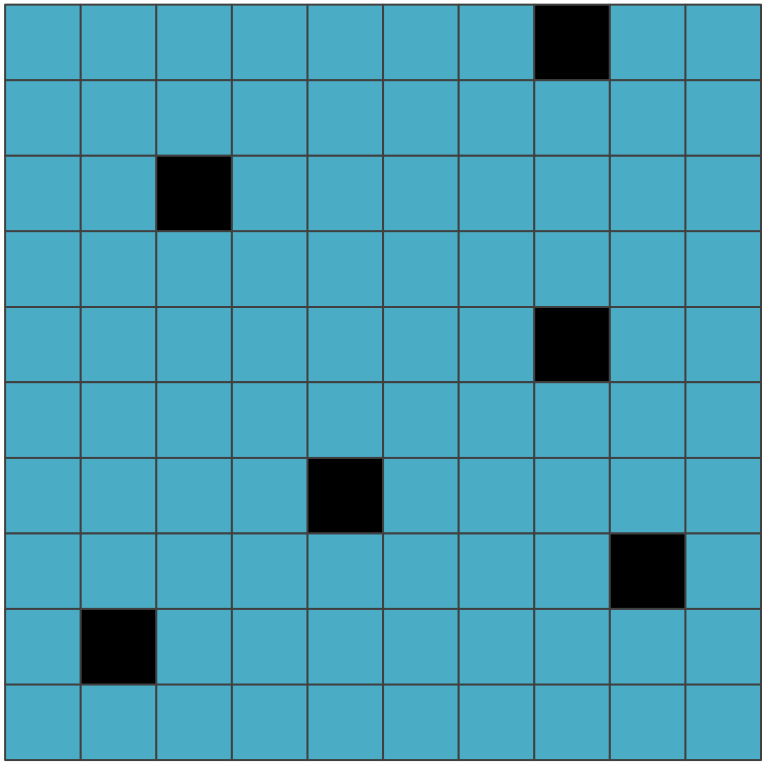
KNOWLEDGE AND DATA

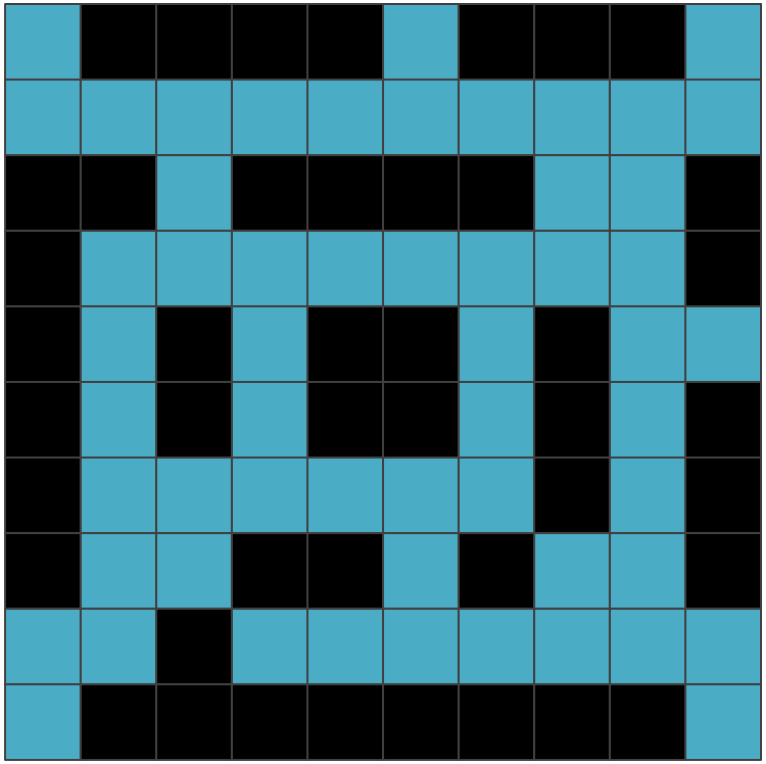




The world is more or less regular …

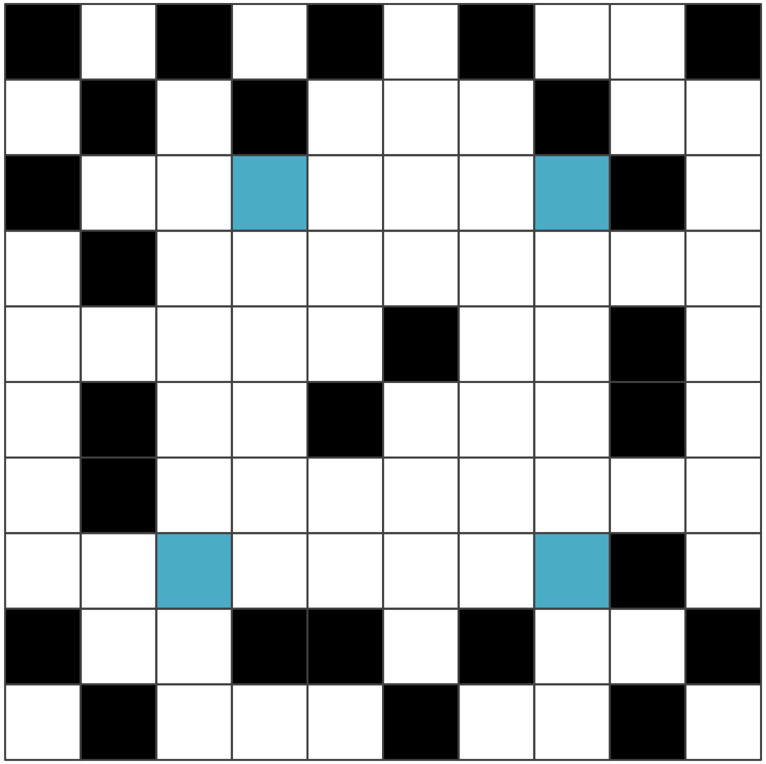
NOISY DATA



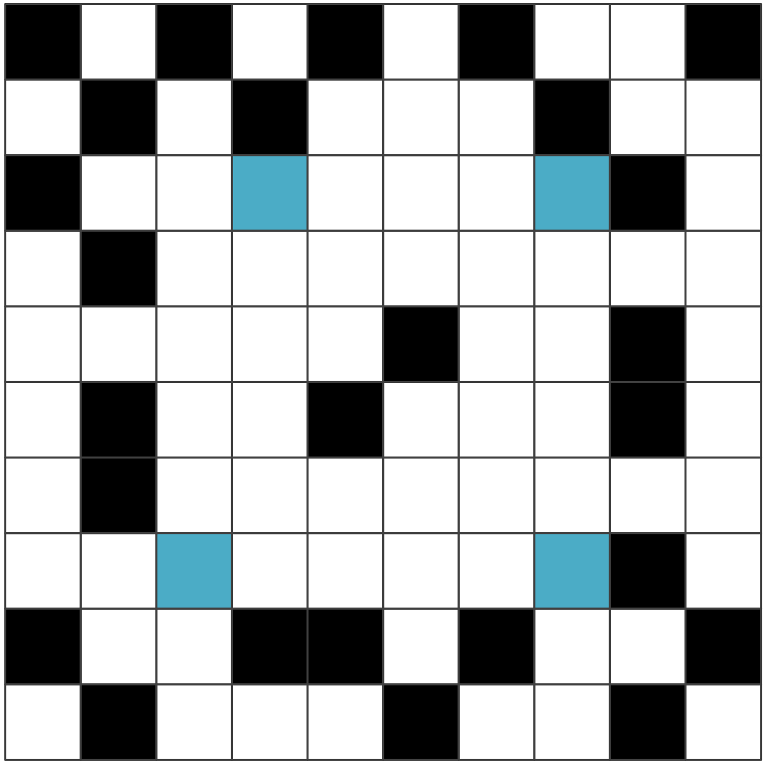


The world is regular but noisy …

NOISY DATA

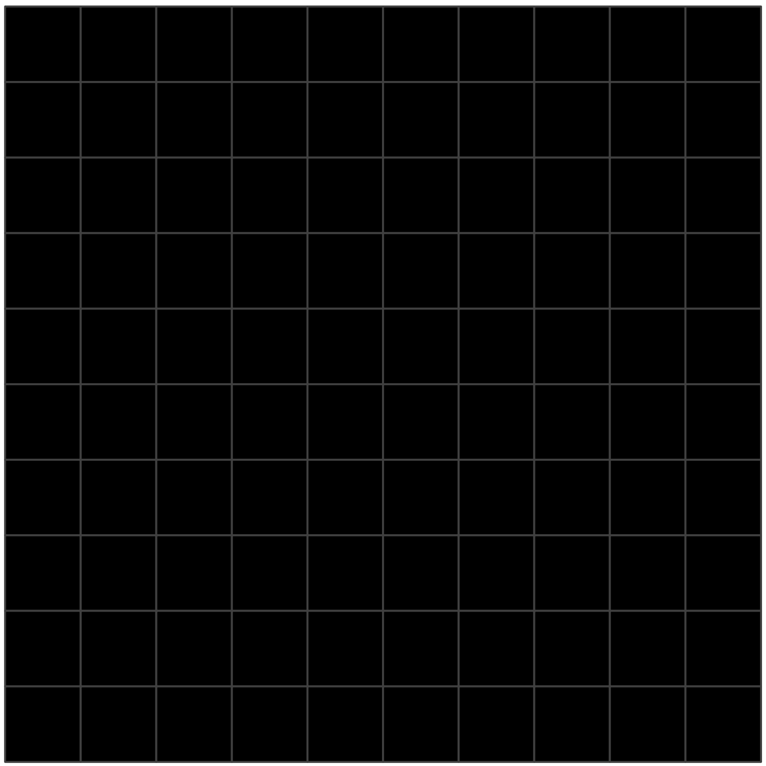


Noise?

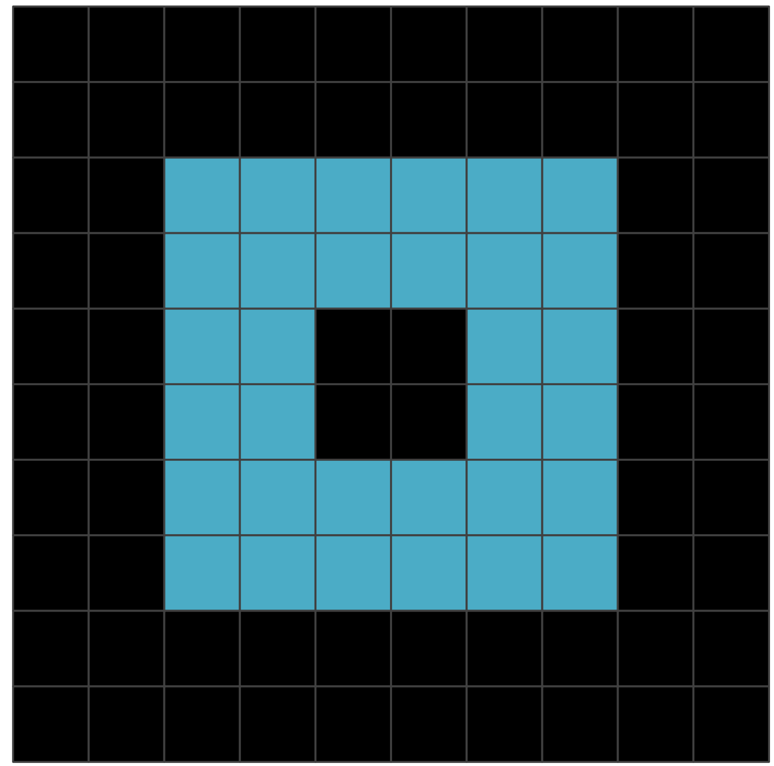


Part of a pattern?

NOISY DATA



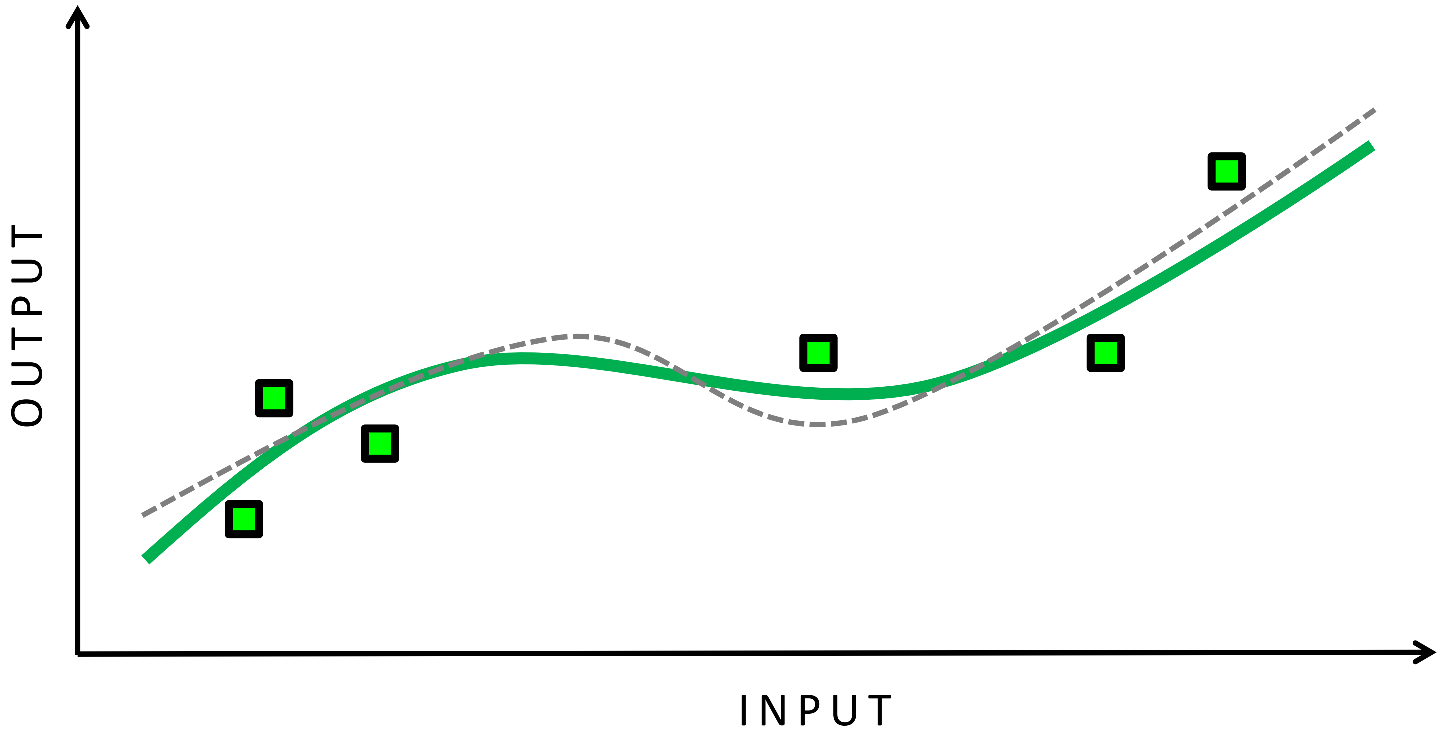
simple world, noisy data



complex world, noise-free data

KNOWLEDGE AND DATA

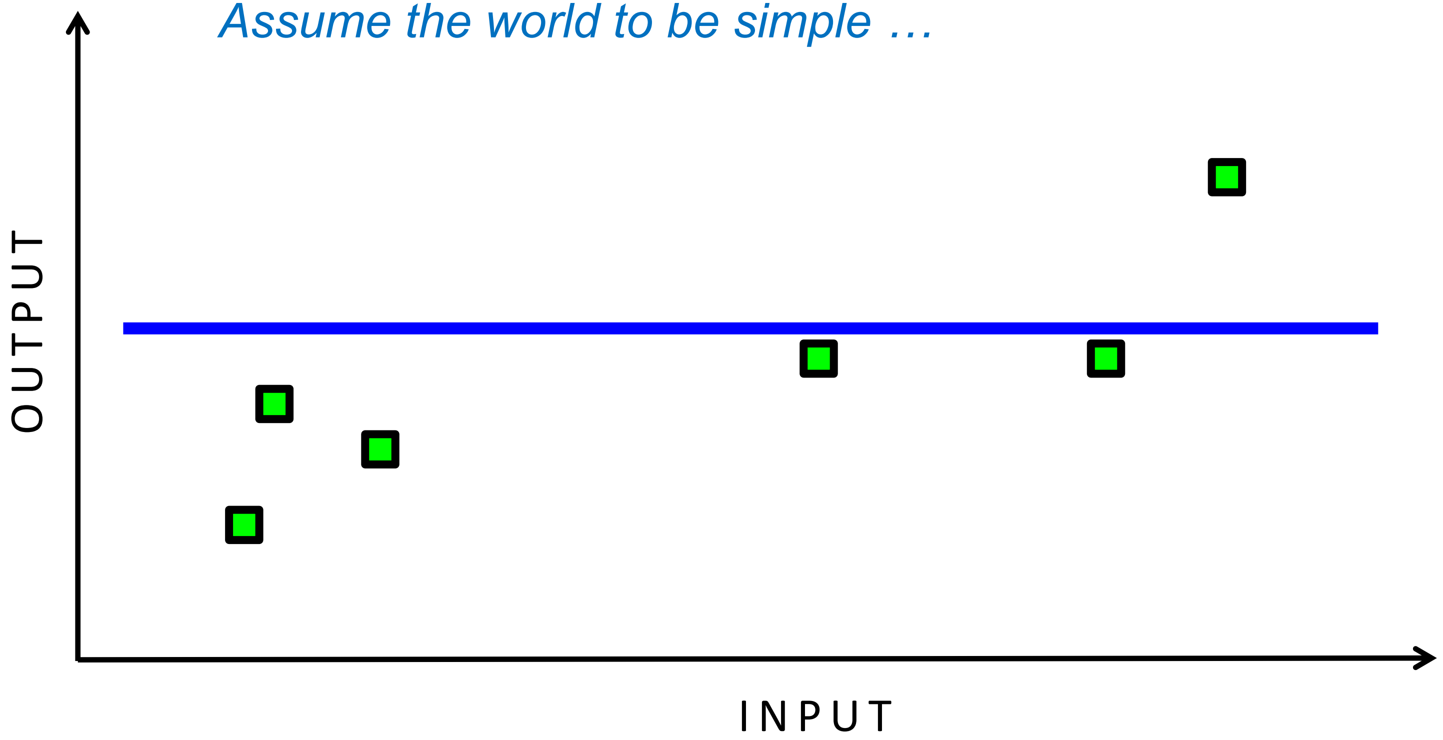
$$Y = f(X) + \epsilon$$



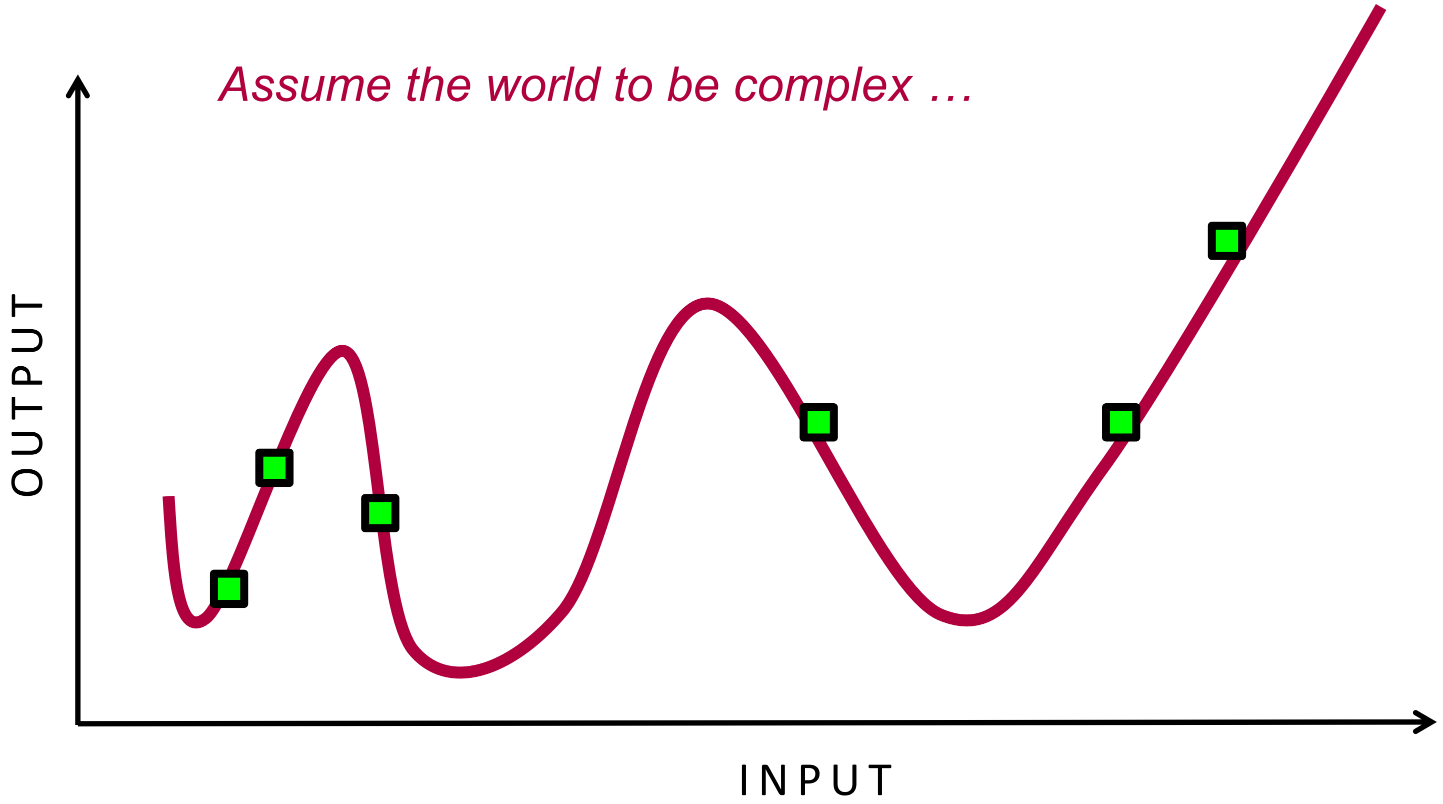
KNOWLEDGE AND DATA

INTELLIGENT
SYSTEMS

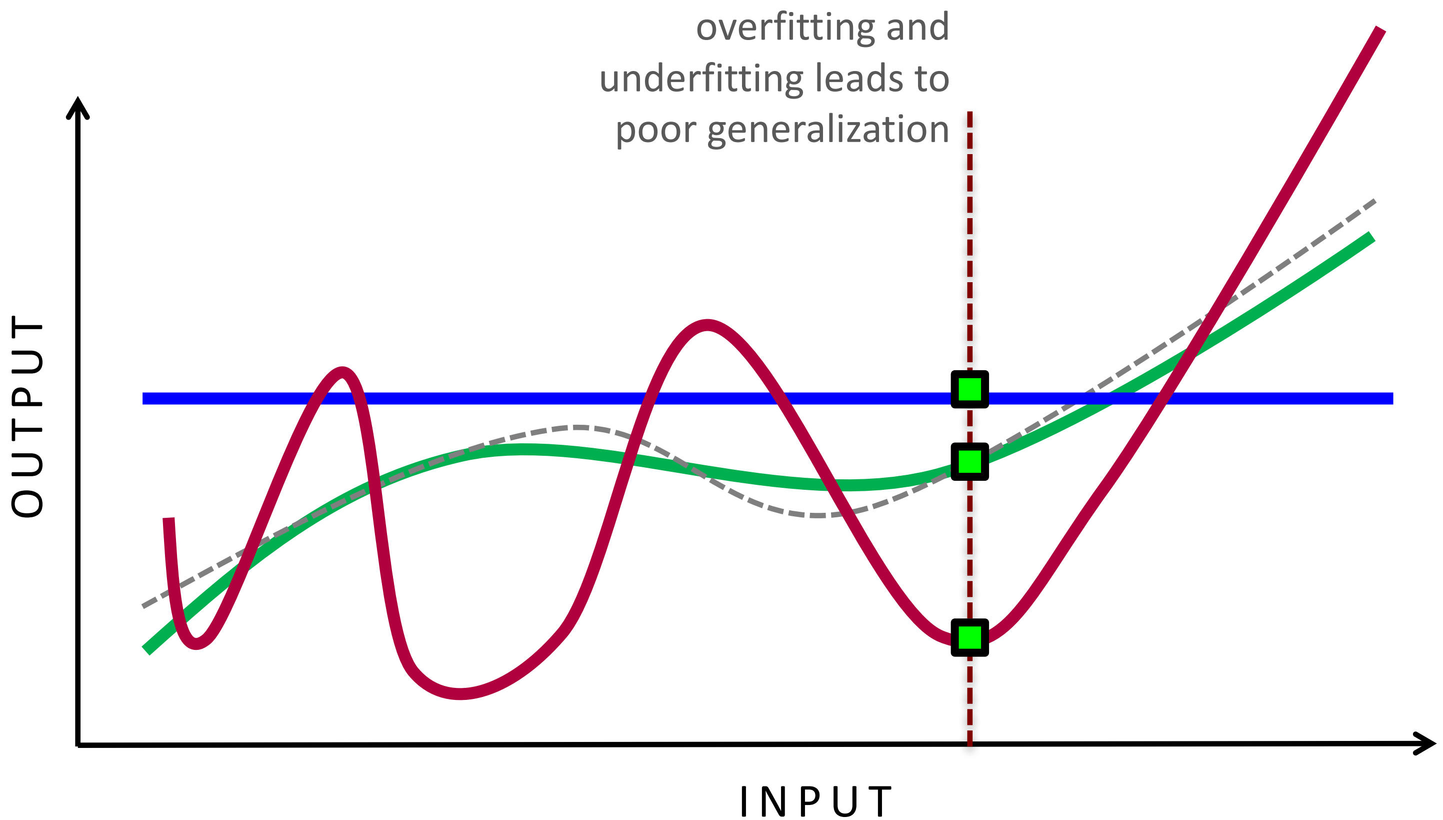
Assume the world to be simple …



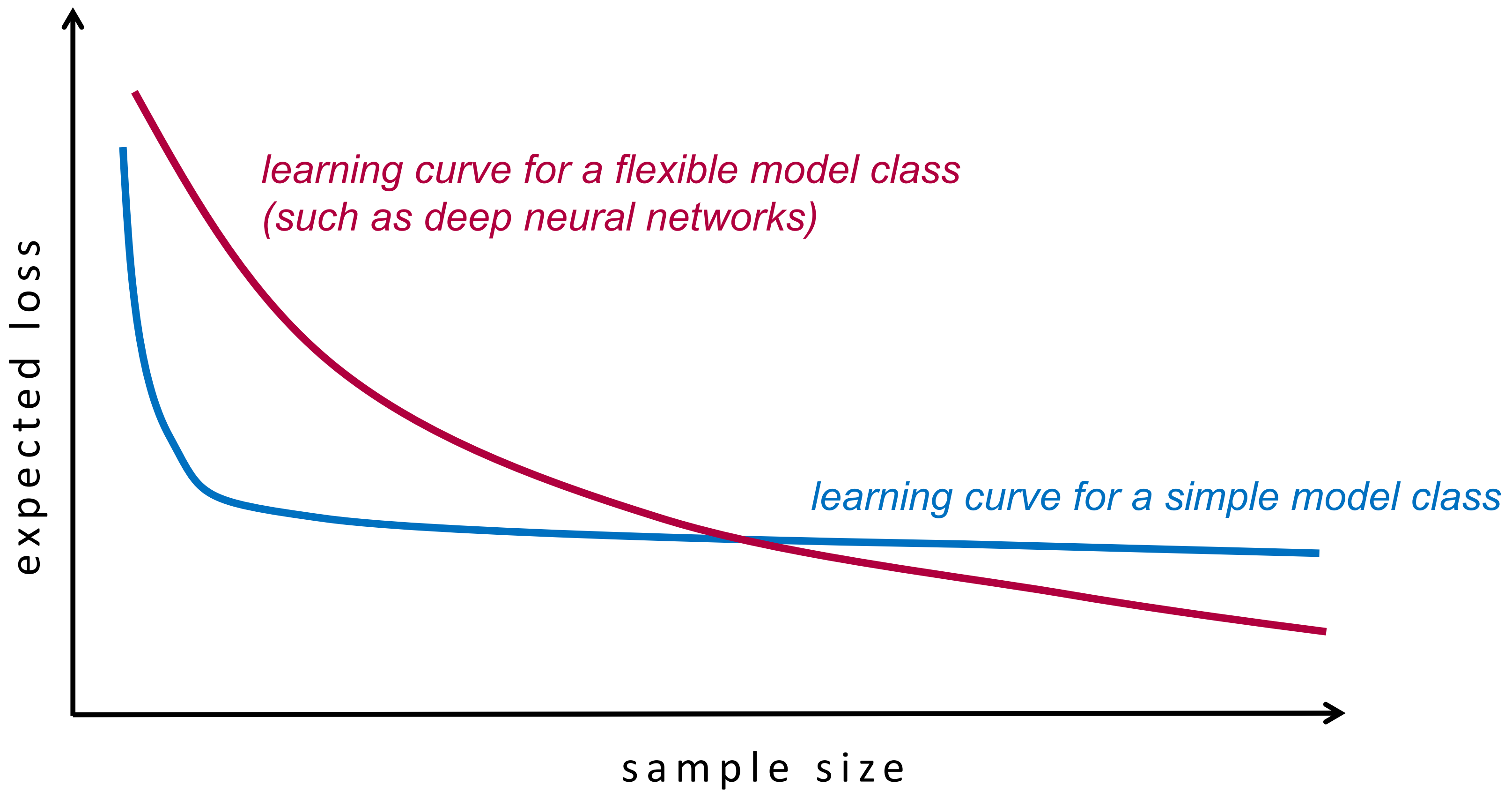
KNOWLEDGE AND DATA



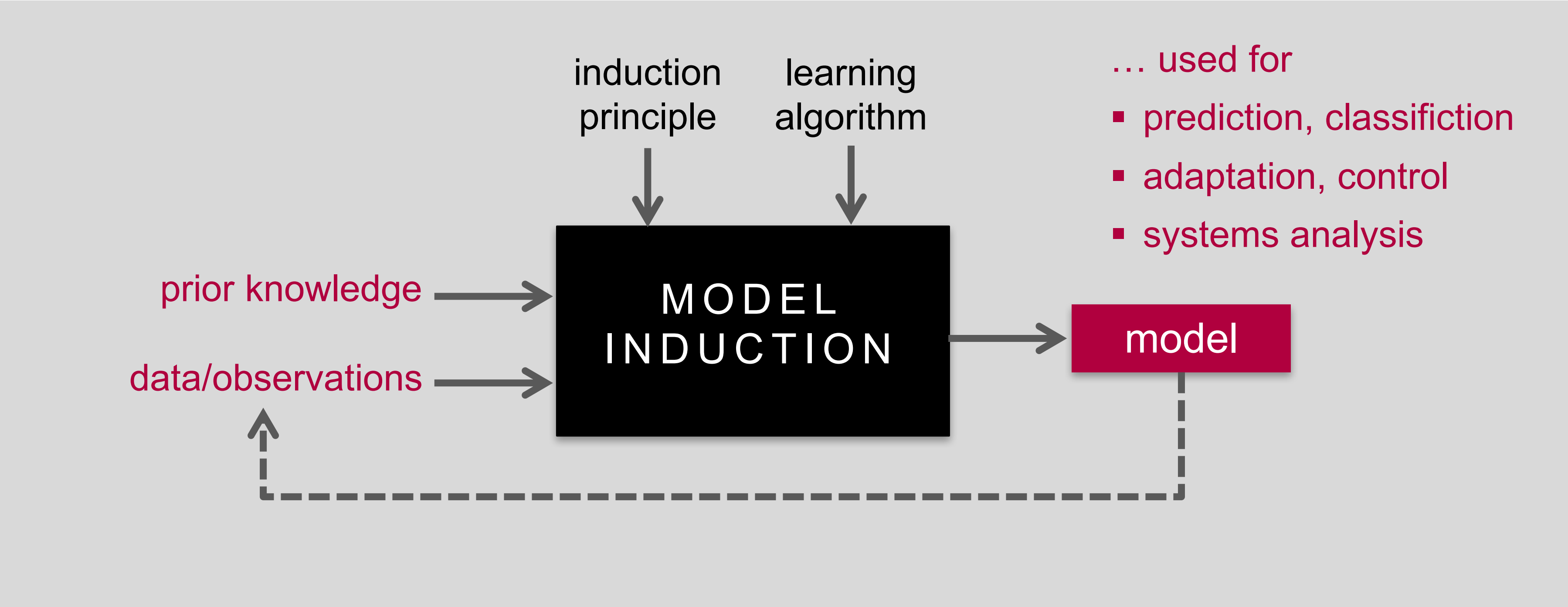
KNOWLEDGE AND DATA



KNOWLEDGE AND DATA



MODEL INDUCTION



- *Learning essentially means revising **prior knowledge** in the light of **observed data**!*
- *Most explicit in frameworks such as Bayesian inference, ILP, ...*
- *Without prior knowledge, data is meaningless ...*
- *Data can compensate for a lack of knowledge, and vice versa.*

MILESTONES OF AI

*Essentially based on **machine learning** technology, makes use of deep neural networks and combines different types of learning (supervised, reinforcement, MCTS)*

AlphaGo beats Lee Sedol (2016)

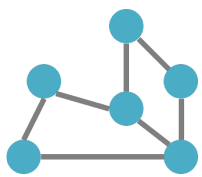
*Massive information **retrieval** (four terabytes of structured and unstructured content), yet little **reasoning** and **learning**.*

Watson wins Jeopardy! (2011)

*Brute force **computing power** (massively parallel system, evaluation of 200 million positions per second), **systematic search**, structured domain.*

Deep Blue beats Garry Kasparov (1997)

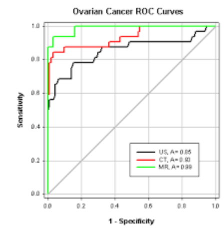
ISG RESEARCH THEMES



For example, a reduction of the search space does not immediately imply better solutions.

structured data and predictions

$x_1 \succ x_2 \succ x_3 \succ x_4$



$$F = \frac{2\sum_{i=1}^{N} y_i \hat{y}_i}{\sum_{i=1}^{N} y_i + \sum_{i=1}^{N} \hat{y}_i}$$

non-standard performance metrics

weak supervision

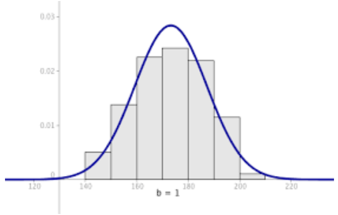
$(x_1, \{A, B\})$

$(x_2, [1, 3])$

$x \succ y$

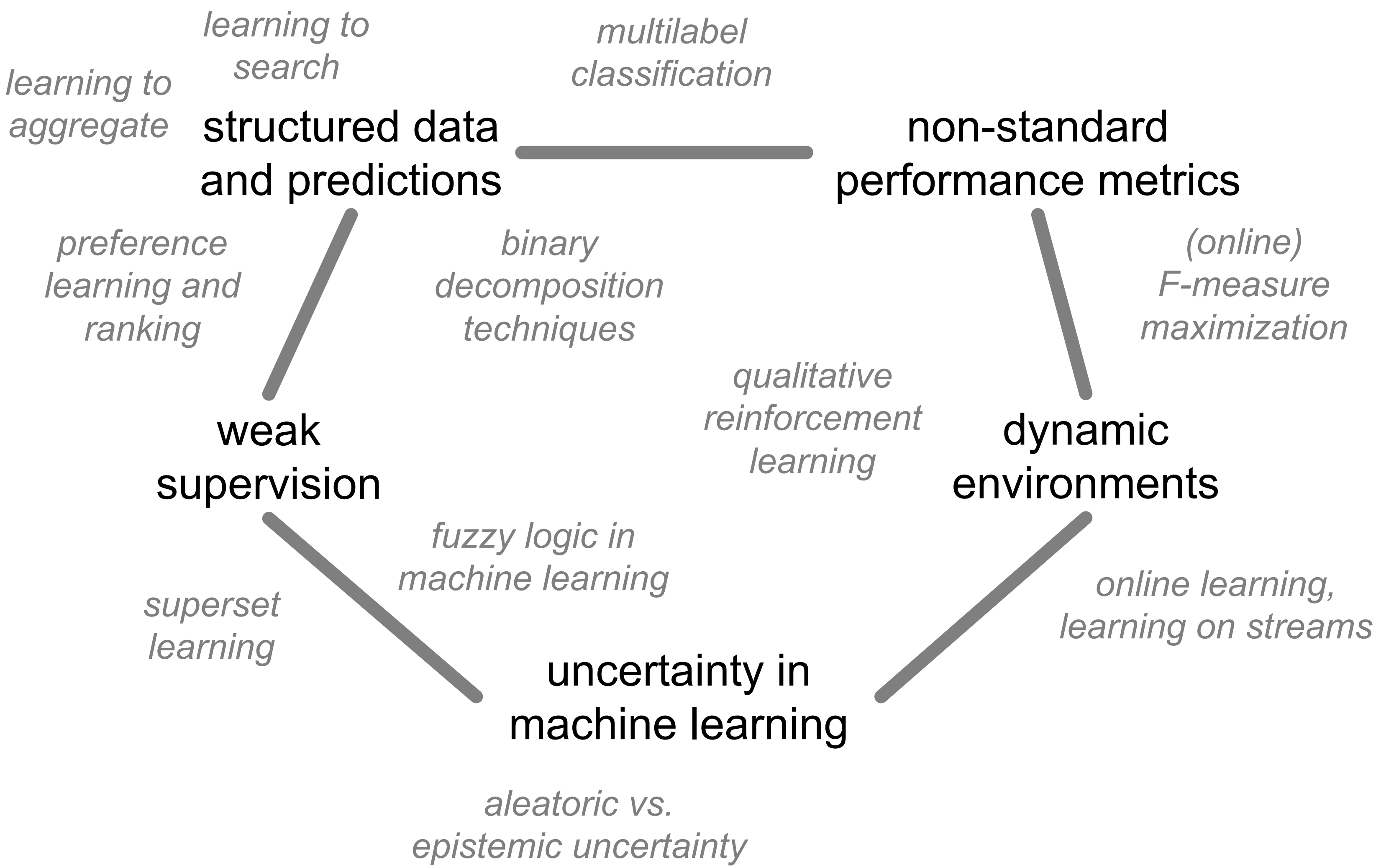
dynamic environments

uncertainty in machine learning

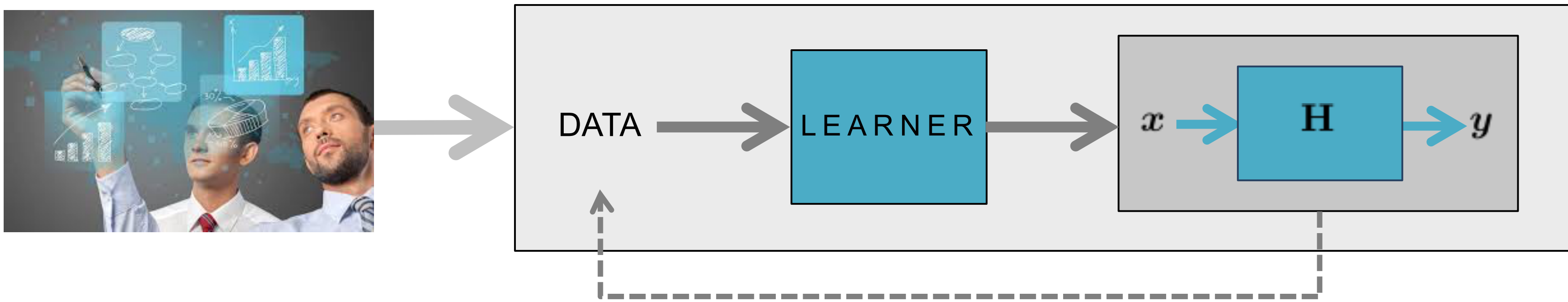


RESEARCH TOPICS

INTELLIGENT SYSTEMS



AUTOMATED MACHINE LEARNING



*The data scientist is not supposed to solve the actual problem (provide an algorithm) but the problem to **learn how to solve that problem** (provide an ecorithm).*

That's not necessarily an easy task either …

INTELLIGENT SYSTEMS

Objective of the learning problem

- specify the prediction task
- success criteria (accuracy/loss function, model com[illegible], ...)
- ...

Specifying the model induction problem

- feature description
- kernel functions
- ...

Solving the model induction problem

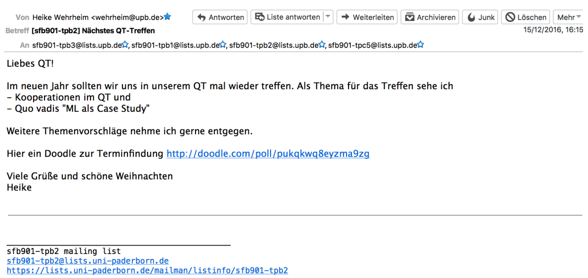
- choice of the learning algorithm
- model evaluation and selection
- ...

MODELING IS IMPORTANT !

All learning algorithms have **(hyper-)parameters**, which have a critical influence on the generalization performance. Tuning these parameters is often tedious and difficult.

INTELLIGENT SYSTEMS

EMAIL


Von Heike Wehrheim <wehrheim@upb.de> Antworten Liste antworten Weiterleiten Archivieren Junk Löschen Mehr
Betreff [sfb901-tpb2] Nächstes QT-Treffen 15/12/2016, 16:15
An sfb901-tpb3@lists.upb.de, sfb901-tpb1@lists.upb.de, sfb901-tpb2@lists.upb.de, sfb901-tpc5@lists.upb.de

Liebes QT!

Im neuen Jahr sollten wir uns in unserem QT mal wieder treffen. Als Thema für das Treffen sehe ich
– Kooperationen im QT und
– Quo vadis "ML als Case Study"

Weitere Themenvorschläge nehme ich gerne entgegen.

Hier ein Doodle zur Terminfindung http://doodle.com/poll/pukqkwq8eyzma9zg

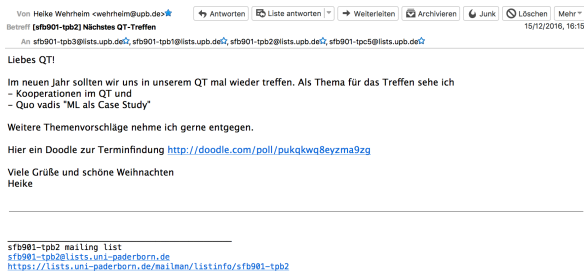
Viele Grüße und schöne Weihnachten
Heike

sfb901-tpb2 mailing list
sfb901-tpb2@lists.uni-paderborn.de
https://lists.uni-paderborn.de/mailman/listinfo/sfb901-tpb2

SPAM or Not SPAM

Many ML algorithms operate in Euclidean spaces …

EMAIL



Von Heike Wehrheim <wehrheim@upb.de> Antworten Liste antworten Weiterleiten Archivieren Junk Löschen Mehr
Betreff **[sfb901-tpb2] Nächstes QT-Treffen** 15/12/2016, 16:15
An sfb901-tpb3@lists.upb.de, sfb901-tpb1@lists.upb.de, sfb901-tpb2@lists.upb.de, sfb901-tpc5@lists.upb.de

Liebes QT!

Im neuen Jahr sollten wir uns in unserem QT mal wieder treffen. Als Thema für das Treffen sehe ich
– Kooperationen im QT und
– Quo vadis "ML als Case Study"

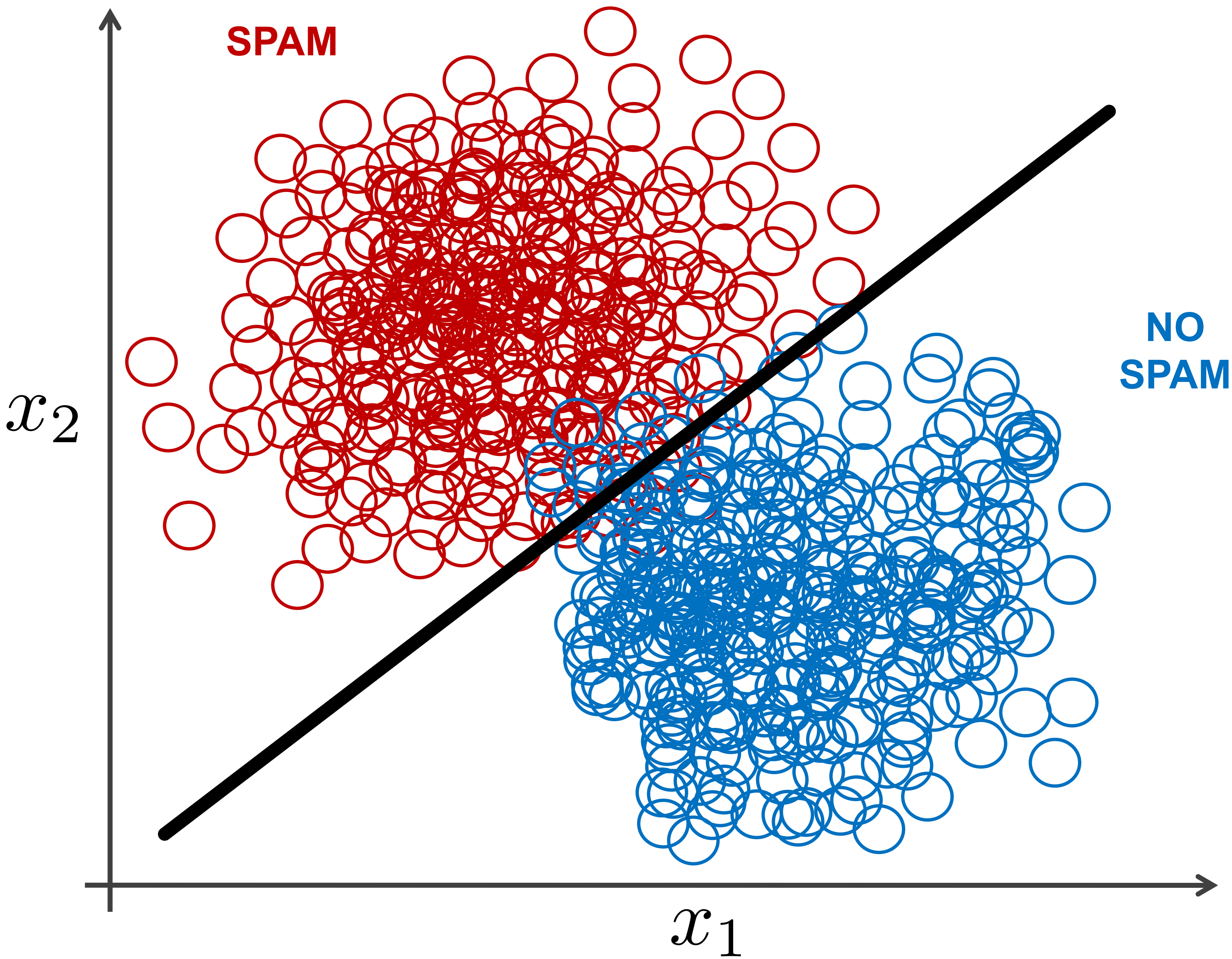
Weitere Themenvorschläge nehme ich gerne entgegen.

Hier ein Doodle zur Terminfindung http://doodle.com/poll/pukqkwq8eyzma9zg

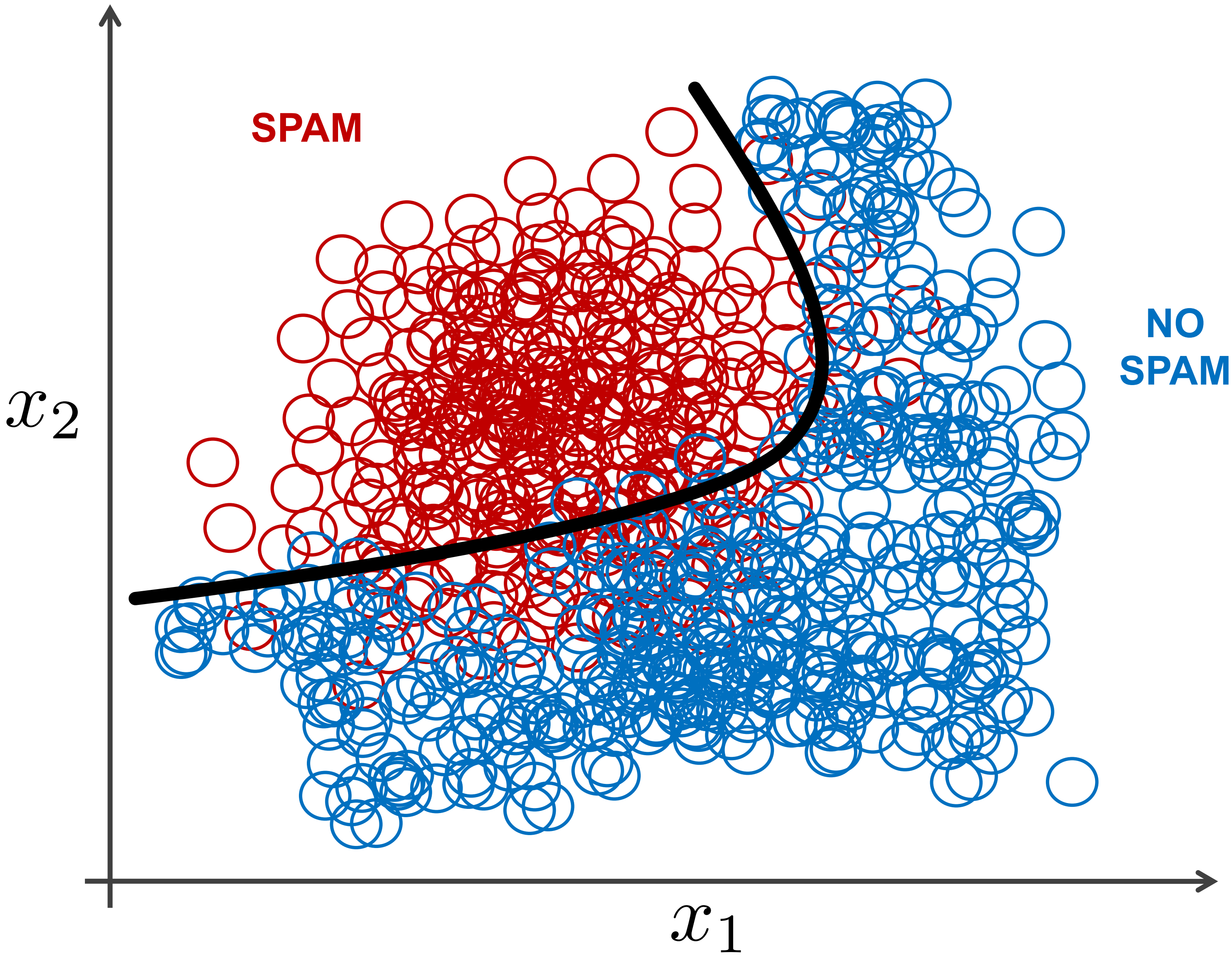
Viele Grüße und schöne Weihnachten
Heike

sfb901-tpb2 mailing list
sfb901-tpb2@lists.uni-paderborn.de
https://lists.uni-paderborn.de/mailman/listinfo/sfb901-tpb2

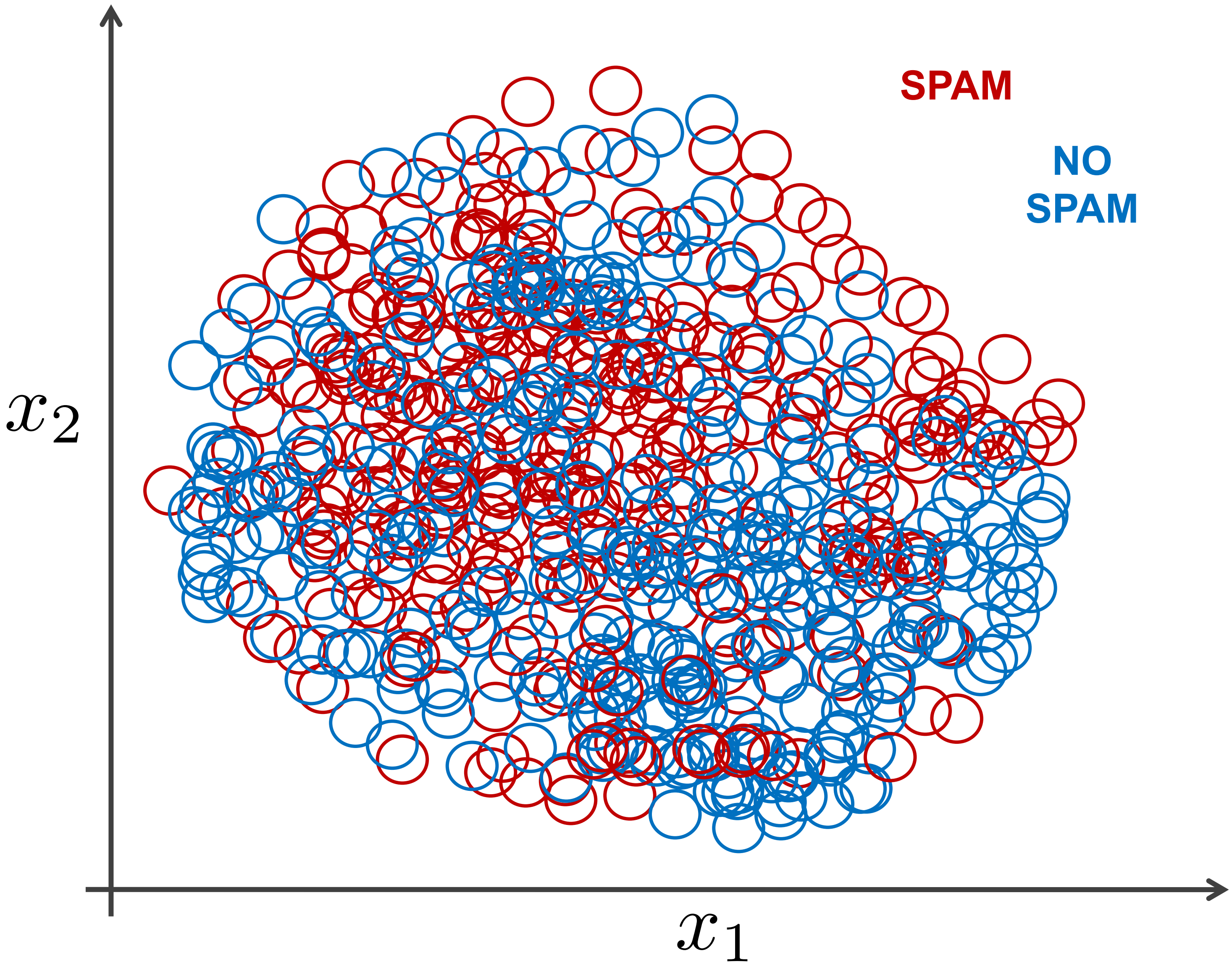
$$\boldsymbol{x} = (x_1, x_2, \ldots, x_d) \in \mathbb{R}^d$$


SPAM
NO SPAM
x_2
x_1

FEATURE ENGINEERING

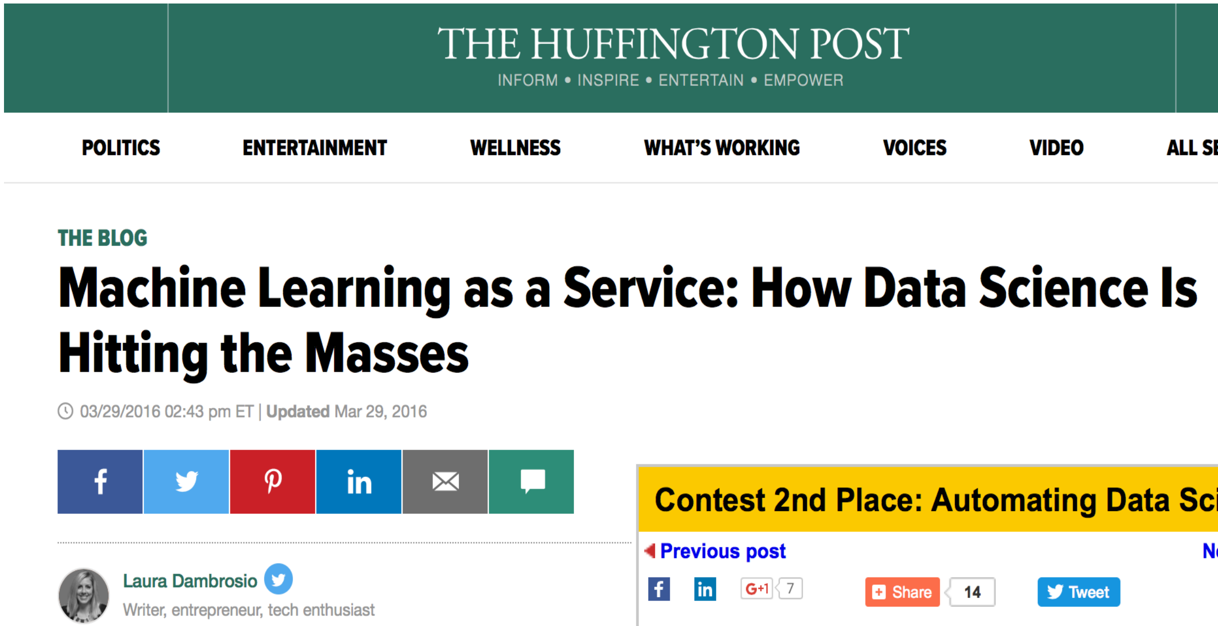


FEATURE ENGINEERING



AUTOMATED MACHINE LEARNING

INTELLIGENT SYSTEMS



THE HUFFINGTON POST
INFORM • INSPIRE • ENTERTAIN • EMPOWER

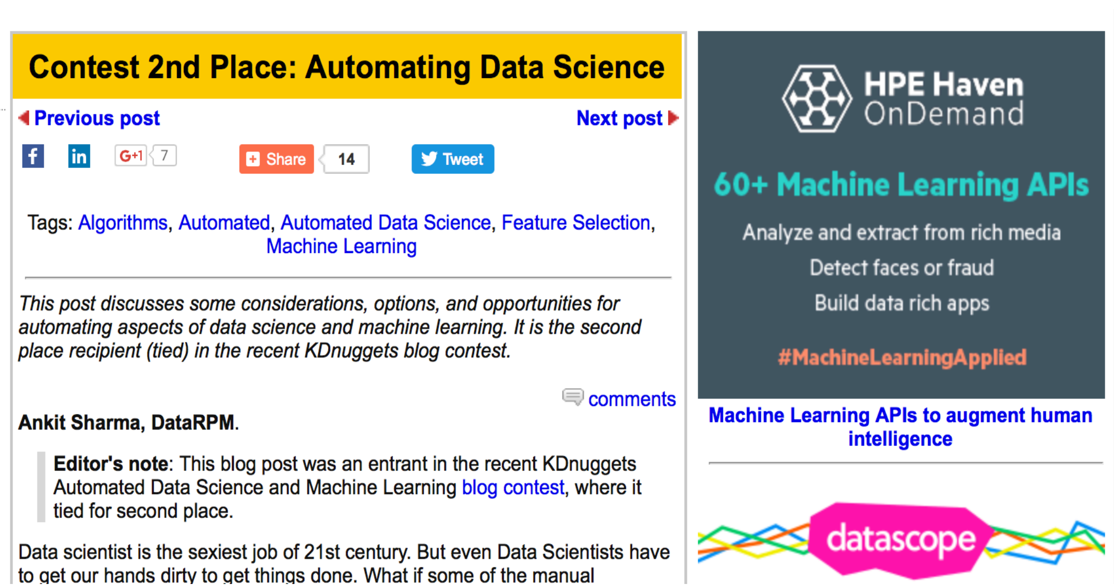
POLITICS ENTERTAINMENT WELLNESS WHAT'S WORKING VOICES VIDEO ALL SE

THE BLOG

Machine Learning as a Service: How Data Science Is Hitting the Masses

03/29/2016 02:43 pm ET | **Updated** Mar 29, 2016

Laura Dambrosio
Writer, entrepreneur, tech enthusiast



Contest 2nd Place: Automating Data Science

Previous post Next post

Share 14 Tweet

Tags: Algorithms, Automated, Automated Data Science, Feature Selection, Machine Learning

This post discusses some considerations, options, and opportunities for automating aspects of data science and machine learning. It is the second place recipient (tied) in the recent KDnuggets blog contest.

comments

Ankit Sharma, DataRPM.

> **Editor's note**: This blog post was an entrant in the recent KDnuggets Automated Data Science and Machine Learning blog contest, where it tied for second place.

Data scientist is the sexiest job of 21st century. But even Data Scientists have to get our hands dirty to get things done. What if some of the manual

HPE Haven OnDemand
60+ Machine Learning APIs
Analyze and extract from rich media
Detect faces or fraud
Build data rich apps
#MachineLearningApplied

Machine Learning APIs to augment human intelligence

datascope

AUTOMATED MACHINE LEARNING


```
function GetMin(var a: TList)
var
    i, min, mini: integer;
begin
    min := MaxInt;
    mini := 0;
    for i := 1 to a.len do
        if a.arr[i].G < min t
        begin
            min := a.arr[i].G
            mini := i;
        end;

    GetMin := mini;
end;
```

```
mann(adam).
mann(tobias).
mann(frank).
frau(eva).
frau(daniela).
frau(ulrike).
vater(adam,tobias).
vater(tobias,frank).
vater(tobias,ulrike).
mutter(eva,tobias).
mutter(daniela,frank).
mutter(daniela,ulrike).
```

```
# Spot Check Algorithms
models = []
models.append(('LR', Logis
models.append(('LDA', Line
models.append(('KNN', KNei
models.append(('CART', Dec
models.append(('NB', Gauss
models.append(('SVM', SVC(
# evaluate each model in t
results = []
names = []
for name, model in models:
    kfold = model_selectio
    cv_results = model_sel
    results.append(cv_resu
    names.append(name)
    msg = "%s: %f (%f)" %
    print(msg)
```

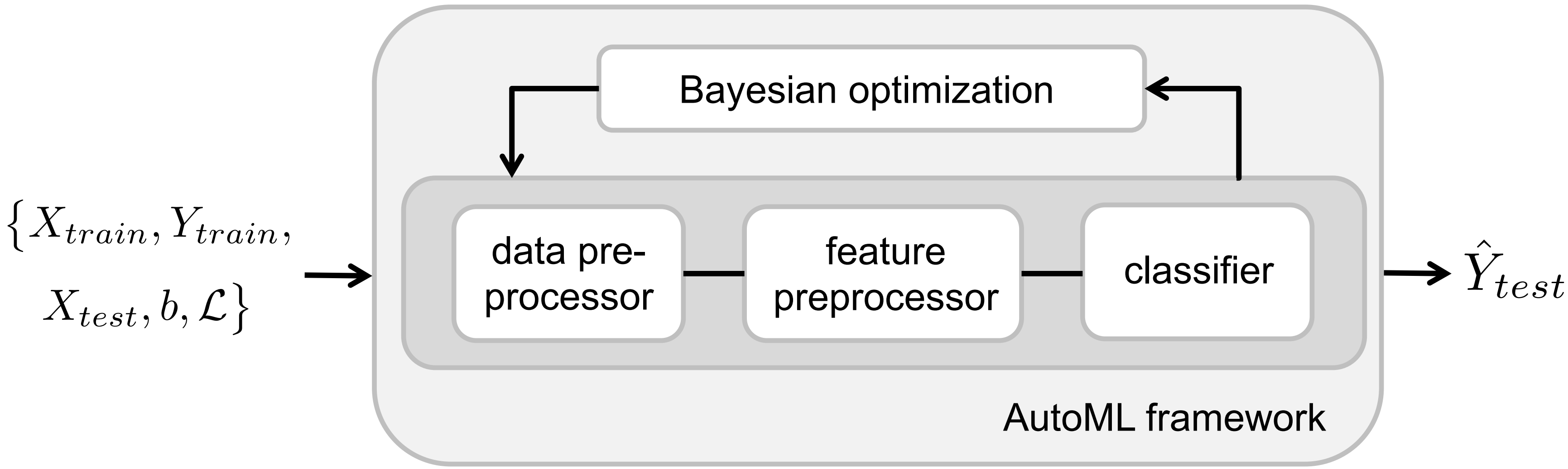
classical programming

knowledge-based programming

“implicit” programming

automated machine learning

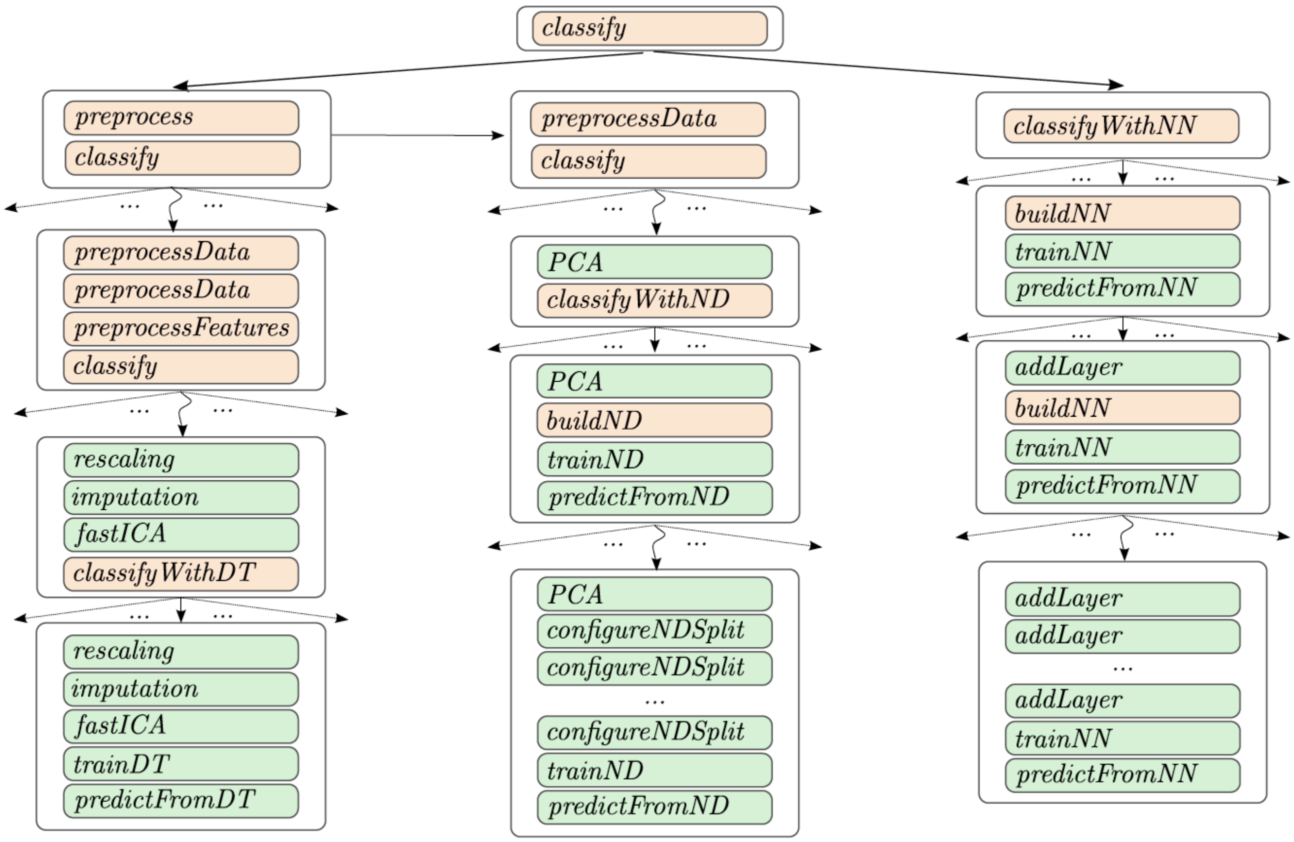
AUTOMATED MACHINE LEARNING



Existing approaches optimize parameters of a ***fixed ML pipeline.***

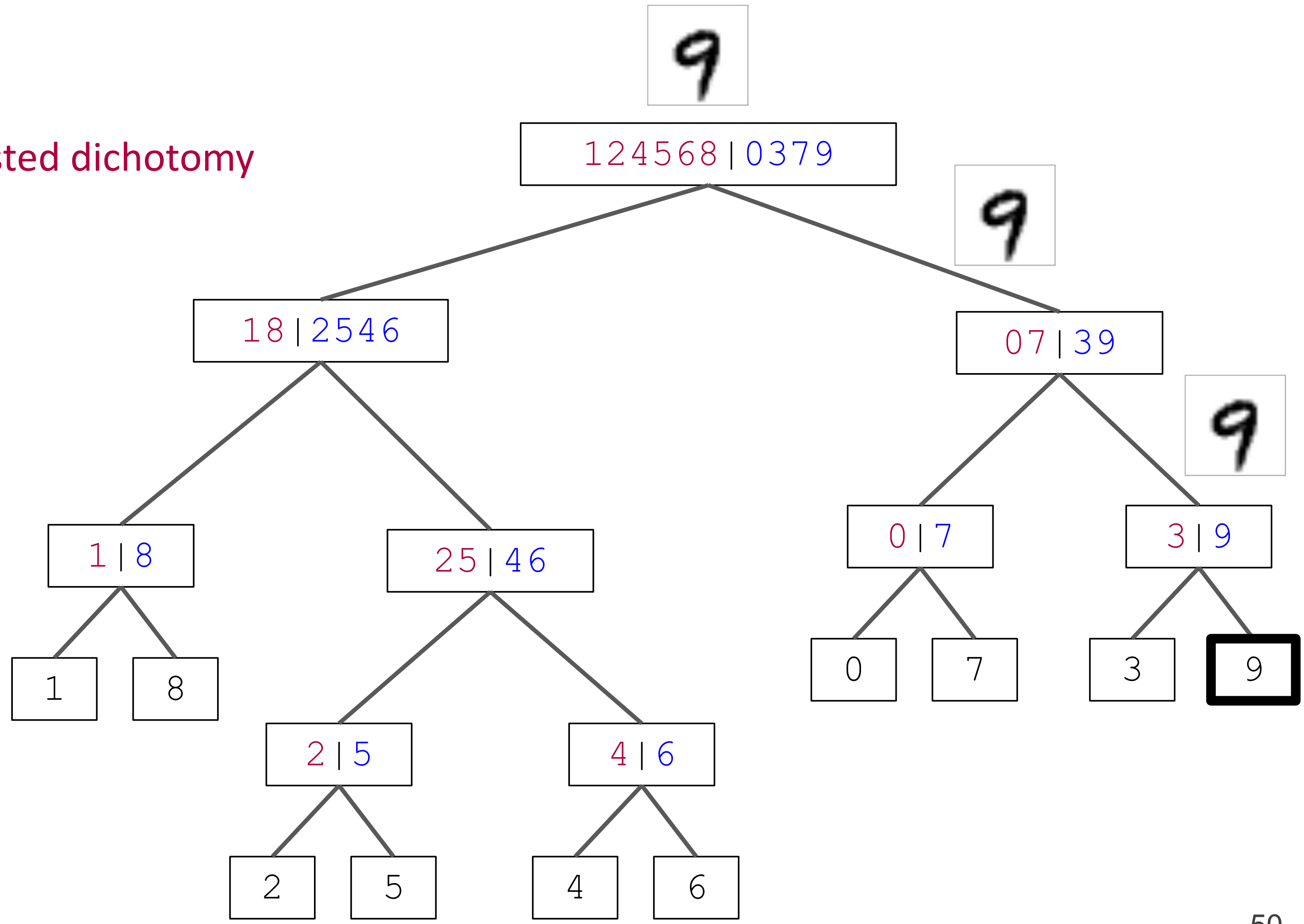
AUTO-ML VIA HIERARCHICAL PLANNING

Combining ML and planning (hierarchical task networks):



AUTO-ML VIA HIERARCHICAL PLANNING

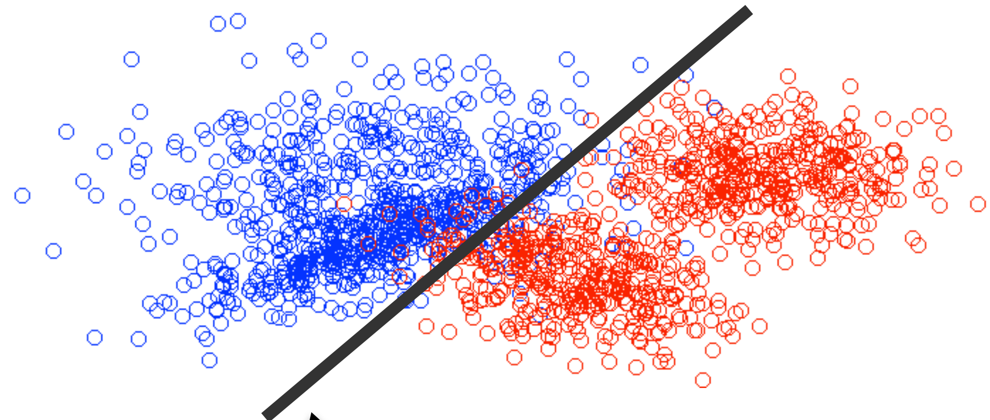
nested dichotomy

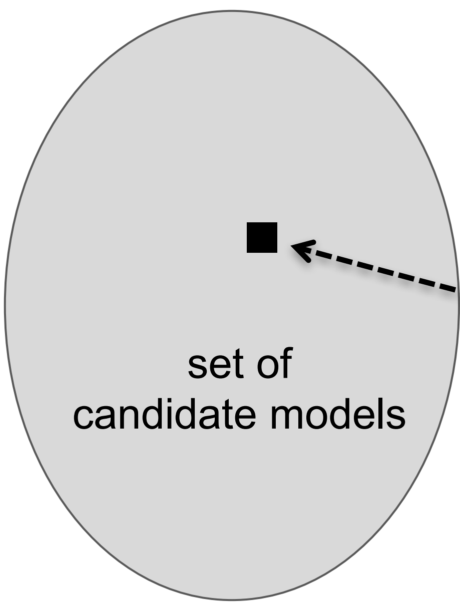


Dataset	RTN	RPND	ND	RTN	RPND	ND
audiology	76.92± 3.65	73.39± 5.28 •	68.76± 6.15 •	74.46± 3.91	74.81± 4.10	70.97± 5.08 •
kropt	33.25± 0.96	32.55± 0.87 •	27.96± 1.36 •	48.98± 1.50	49.19± 1.33	44.91± 1.85 •
letter	71.50± 1.69	66.82± 2.55 •	51.51± 3.38 •	80.12± 0.74	79.83± 0.74 •	78.95± 1.04 •
mfeat-factors	94.59± 1.36	94.12± 1.54	92.14± 1.52 •	87.61± 1.50	87.24± 1.61	86.31± 1.71 •
mfeat-fourier	75.79± 2.29	74.71± 1.91 •	71.77± 2.30 •	72.87± 1.80	72.83± 1.94	71.43± 1.99 •
mfeat-karhunen	89.09± 1.84	88.66± 1.70	84.91± 2.50 •	80.72± 1.98	80.25± 2.05	78.87± 2.09 •
optdigits	93.40± 0.77	92.03± 1.64 •	89.93± 2.38 •	90.49± 0.90	89.67± 1.22 •	88.75± 1.18 •
page-blocks	96.49± 0.38	96.30± 0.43 •	95.71± 0.66 •	96.96± 0.36	96.96± 0.39	96.93± 0.36
pendigits	93.78± 0.82	90.35± 2.26 •	87.19± 3.53 •	95.37± 0.49	94.99± 0.52 •	94.77± 0.52 •
segment	95.17± 0.78	93.91± 1.96 •	90.20± 4.04 •	95.71± 0.90	95.60± 0.79	94.94± 0.97 •
shuttle	98.96± 5.83	98.99± 5.74	98.98± 5.77	100.00± 0.00	100.00± 0.00	100.00± 0.00
vowel	82.91± 2.21	79.96± 3.64 •	52.12± 8.83 •	72.97± 3.45	72.49± 3.52	71.09± 3.48 •
yeast	58.48± 1.92	58.27± 1.97	56.41± 1.89 •	57.14± 2.22	57.25± 1.82	56.29± 2.35 •
zoo	93.88± 4.27	93.62± 4.91	90.98± 5.69 •	93.66± 4.88	92.93± 4.90	91.16± 4.63 •

Table 2. Experimental results (mean accuracy ± standard deviation) using logistic regression (left) and C4.5 (right)

ML as search, Auto-ML as generalized search …

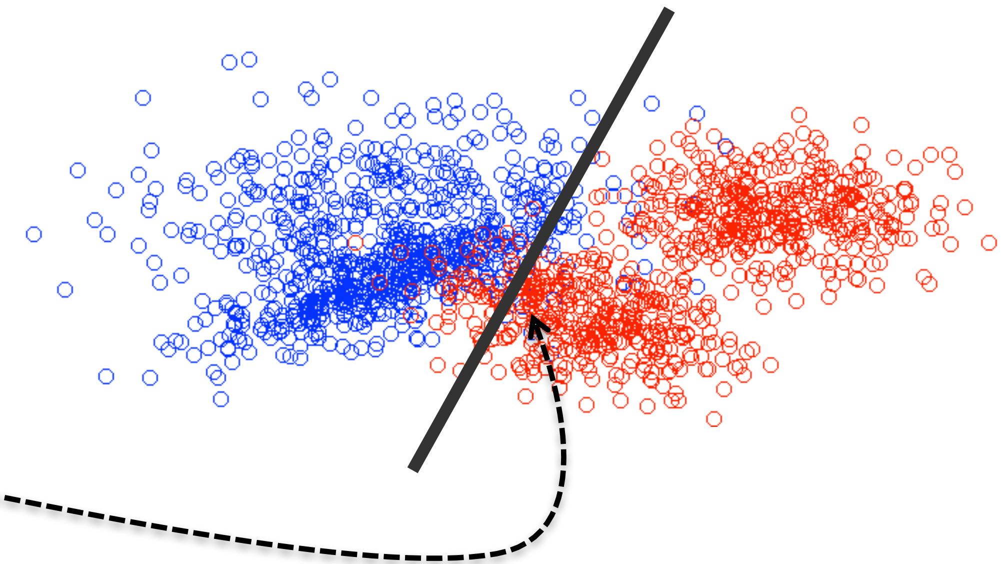


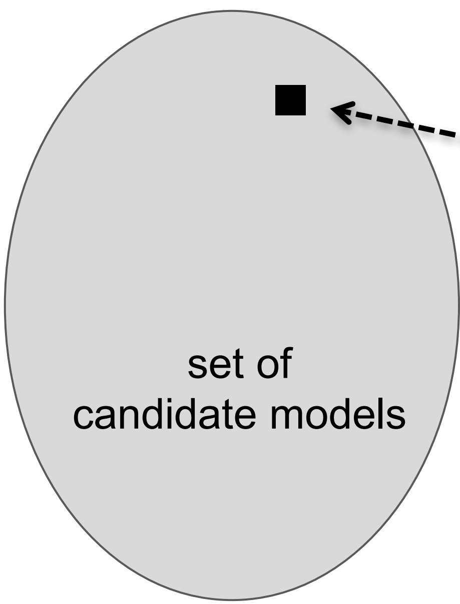


$$h^* \in \underset{h \in \mathcal{H}}{\operatorname{argmin}} \frac{1}{N} \sum_{i=1}^{N} \ell\big(h(\boldsymbol{x}_i), y_i\big) + \lambda\, \Phi(h)$$

HYPOTHESIS SPACE $\mathcal{H}$

ML as search, Auto-ML as generalized search …





$$h^* \in \operatorname*{argmin}_{h \in \mathcal{H}} \frac{1}{N} \sum_{i=1}^{N} \ell\big(h(\boldsymbol{x}_i), y_i\big) + \lambda\, \Phi(h)$$

HYPOTHESIS SPACE $\mathcal{H}$

RESEARCH TOPICS

INTELLIGENT SYSTEMS

learning to aggregate

learning to search

structured data and predictions

multilabel classification

non-standard performance metrics

preference learning and ranking

binary decomposition techniques

(online) F-measure maximization

qualitative reinforcement learning

weak supervision

dynamic environments

fuzzy logic in machine learning

superset learning

online learning, learning on streams

uncertainty in machine learning

aleatoric vs. epistemic uncertainty

… applications

INTELLIGENT SYSTEMS

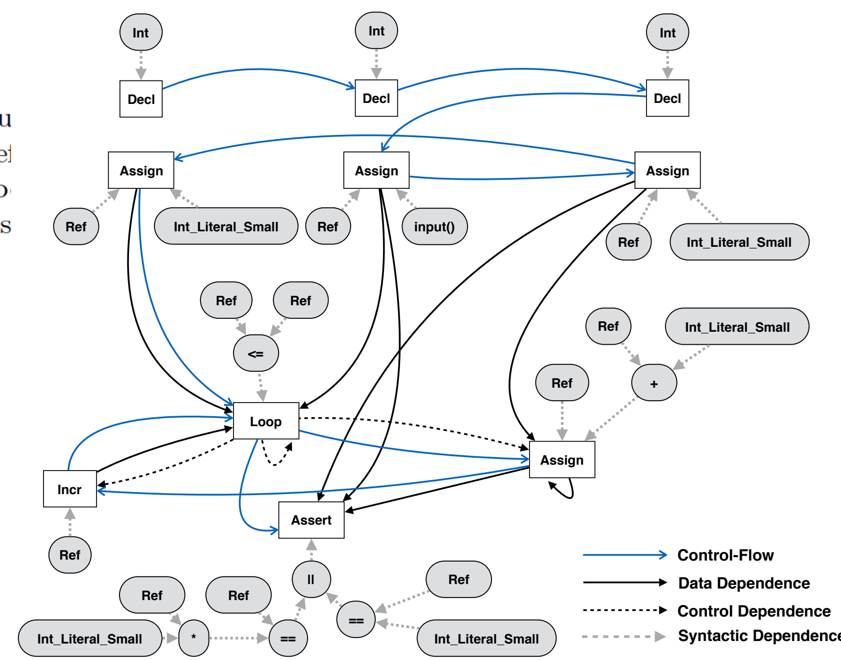
Predicting Rankings of Software Verification Competitions*

Mike Czech, Eyke Hüllermeier, and Heike Wehrheim

Department of Computer Science
Paderborn University
Germany

Abstract. Software verification competitions, such as the annu
COMP, evaluate software verification tools with respect to their ef
ity and efficiency. Typically, the outcome of a competition is a (p
category-specific) *ranking* of the tools. For many applications, s

***Weisfeiler-Lehman subtree kernels** on a graph representation for software source code that mixes elements of control flow and program dependence graphs with abstract syntax trees.*



Imprecise Matching of Requirements Specifications for Software Services using Fuzzy Logic

Marie C. Platenius, Wilhelm Schäfer
Software Engineering
Heinz Nixdorf Institute
Paderborn University, Germany
{m.platenius,wilhelm}@upb.de

Ammar Shaker, Eyke Hüllermeier
Intelligent Systems
Department of Computer Science
Paderborn University, Germany
{ammar.shaker,eyke}@upb.de

Matthias Becker
Software Engineering
Fraunhofer IEM
Paderborn, Germany
matthias.becker@iem.fraunhofer.de

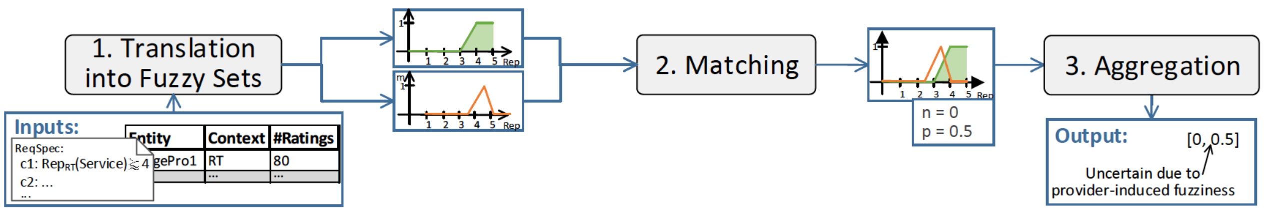
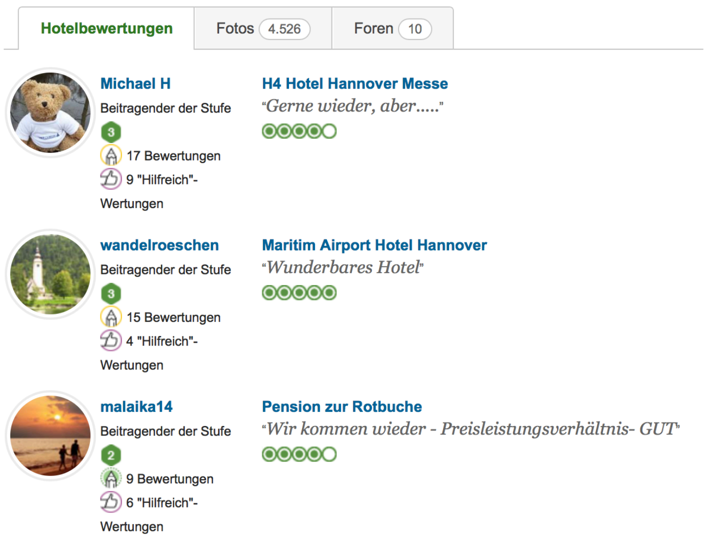


Fig. 2. Fuzzy Reputation Matching Procedure

COOPERATION @ UPB (Frick)



Economic aspects of rating and reputation, reverse engineering of rating systems such as TripAdvisor.

Pairwise versus Pointwise Ranking: A Case Study

Vitalik Melnikov[1], Pritha Gupta[1], Bernd Frick[2],
Daniel Kaimann[2], Eyke Hüllermeier[1]
[1]Department of Computer Science
[2]Faculty of Business Administration and Economics
Paderborn University
Warburger Str. 100, 33098 Paderborn
e-mail: {*melnikov,prithag,eyke*}*@mail.upb.de*, {*bernd.frick,daniel.kaimann*}*@upb.de*

Abstract. Object ranking is one of the most relevant problems in the realm of preference learning and ranking. It is mostly tackled by means of two different techniques, often referred to as pairwise and pointwise ranking. In this paper, we present a case study in which we systematically compare two representatives of these techniques, a method based on the reduction of ranking to binary classification and so-called expected rank regression (ERR). Our experiments are meant to complement existing studies in this field, especially previous evaluations of ERR. And indeed, our results are not fully in agreement with previous findings and partly support different conclusions.

Keywords: Preference learning, object ranking, linear regression, logistic regression, hotel rating, TripAdvisor

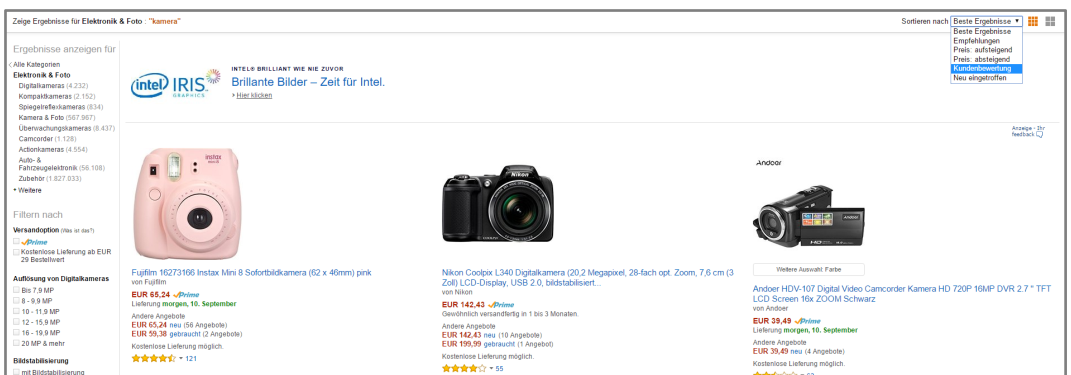
1. Introduction

Preference learning is an emerging subfield of machine learning that has received increasing attention in recent years [3]. Roughly speaking, the goal in preference learning is to induce preference models from observed data that reveals information about the preferences of an individual or a group of individuals in a direct or indirect way; these models are then used to predict the preferences in a new situation.

In general, a preference learning system is provided with a set of items (e.g., products) for which preferences are known, and the task is to learn a function that

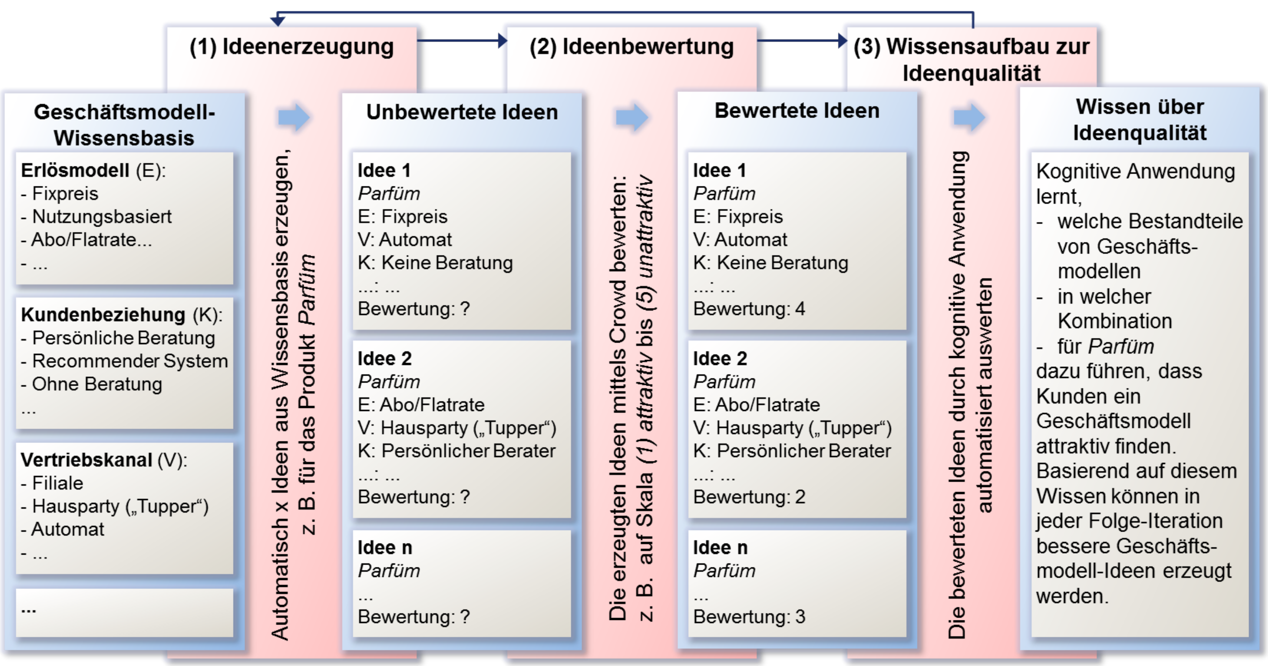
COOPERATION @ UPB (Fahr)

Behavioral economics:
How people aggregate customer reviews
(→ "learning to aggregate")



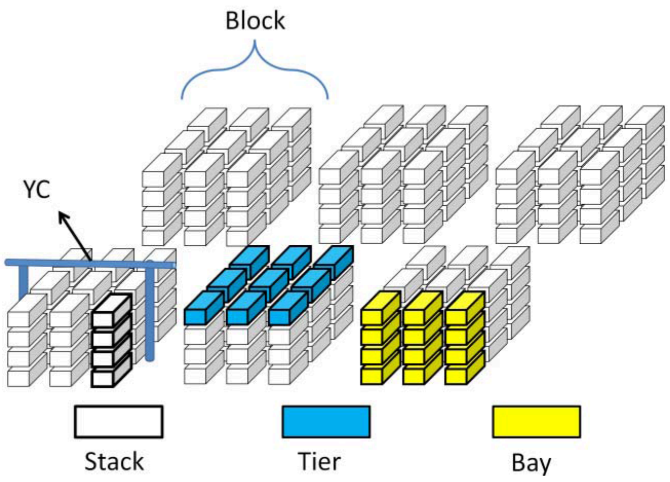
COOPERATION @ UPB (Kundisch/John)

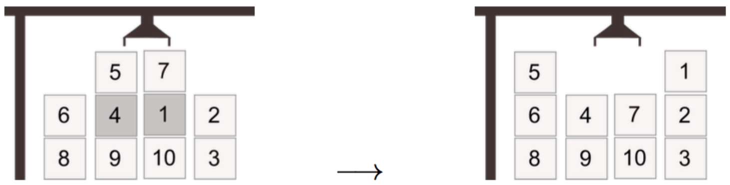
Machine learning for the support of technology-based consulting for the innovation of business models.



COOPERATION @ UPB (Tierney)

Machine Learning for improving optimization methods for the Container Pre-Marshalling problem.





COOPERATION @ UPB (Tophinke, Geierhos)

Analyse des Sprachausbaus und der Entwicklung von Grammatik im Mittelniederdeutschen.

08.02.2017 |

Fakultätsübergreifendes DFG-Projekt im Bereich Digital Humanities an der Universität Paderborn gestartet

Neues Forschungsprojekt im Bereich Digital Humanities an der Universität Paderborn:

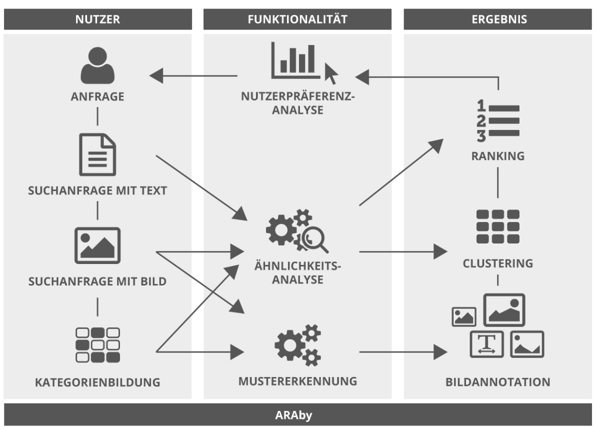
„InterGramm“ untersucht den Sprachausbau im Mittelniederdeutschen. Die Deutsche Forschungsgemeinschaft (DFG) fördert das Vorhaben mit rund einer halben Million Euro. **Prof. Dr. Doris Tophinke** (Fakultät für Kulturwissenschaften), **Jun.-Prof. Dr. Michaela Geierhos** (Fakultät für Wirtschaftswissenschaften) und **Prof. Dr. Eyke Hüllermeier** (Fakultät für Elektrotechnik, Informatik und Mathematik) arbeiten an einer interaktiven Grammatikanalyse historischer Texte.

Foto (Universität Paderborn, Johannes Pauly): Das interdisziplinäre Forschen kennzeichnet das innovative DFG-Forschungsprojekt „InterGramm“

COOPERATION @ UPB (Seng)

INTELLIGENT SYSTEMS

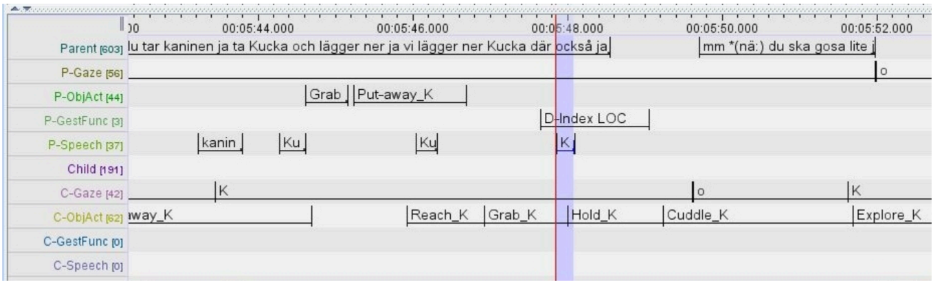
ARAby: An adaptive retrieval and analysis tool for supporting image-based research processes



„Aby gets digital“: Digitalization of the systematic comparison and analysis of images as practiced by Aby Warburg.

COOPERATION @ UPB (Rohlfing)

Temporal data mining for analyzing multimodal parent-child interaction.

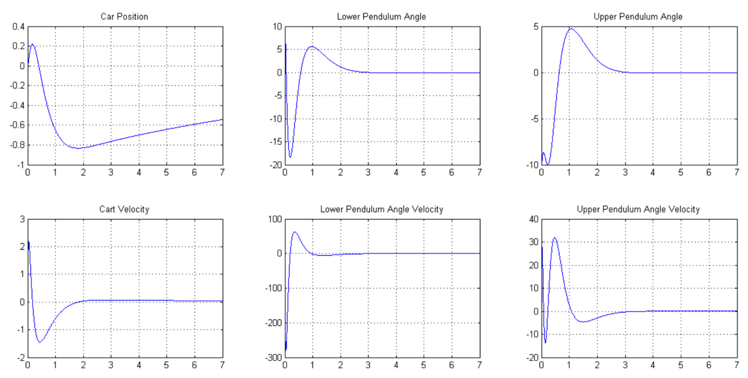


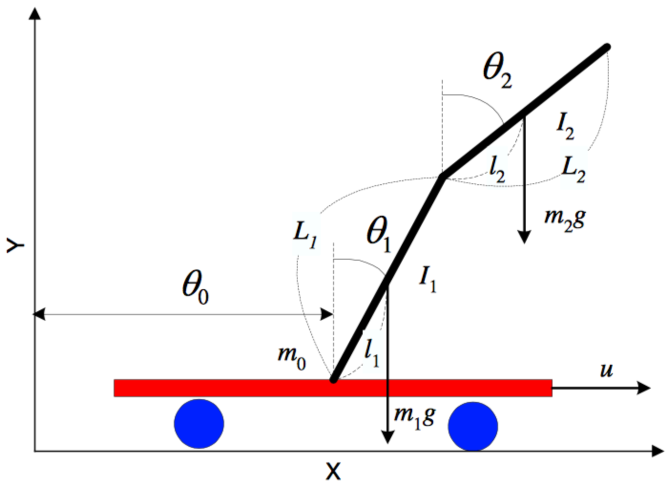
INTELLIGENT SYSTEMS

Die Macht der Algorithmen:
Zur epistemologischen und gesellschaftlichen Dimension aktueller Algorithmik.

COOPERATION @ UPB (Traechtler)

Machine learning for the control of technical systems.





SUMMARY

- Machine learning is developing rapidly, emerging topics include Auto-ML, large-scale learning, deep learning, …
- Many applications and opportunities for interdisciplinary projects.

SUMMARY

INTELLIGENT
SYSTEMS

- Machine learning is developing rapidly, emerging topics include Auto-ML, large-scale learning, deep learning, …
- Many applications and opportunities for interdisciplinary projects.

- We looked at ML from the point of view of automated programming.
- Standard ML can be seen as combining knowledge and data (revising the former in light of the latter).
- Quest for “real” automation motivates work on Auto-ML.
- ML as an art, science, and technology, with mathematical, computational, technical, philosophical, social, psychological, and biological dimensions, amongst others …