"Discover how to discover best"

How Computers Discover How Computers Discover

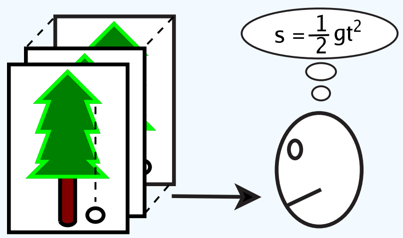
A Mini-Review of Algorithmic Meta-Discovery

Filip Železný

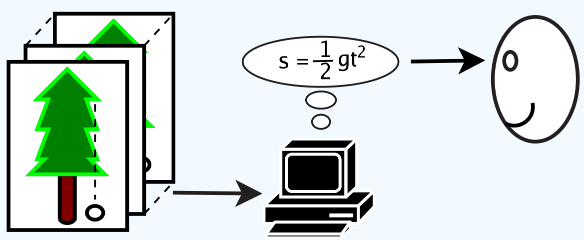
ČVUT Prague, School of Electrical Engineering, Dept. of Cybernetics
The Gerstner Laboratory for Intelligent Decision Making and Control

Introduction

:: Traditional **scientific discovery**: a human forming a hypothesis explaining observations of some natural phenomena.



:: Computer-based scientific discovery, usually employing **machine learning** algorithms.



Automated Discovery

:: Computer programs constructing hypotheses from data

– Machine Learning

– Data Mining

– Knowledge Discovery in Databases

:: Highlight: the **Robot-Scientist** project (UK)

– Robot develops predicate-logic hypotheses in functional genomics

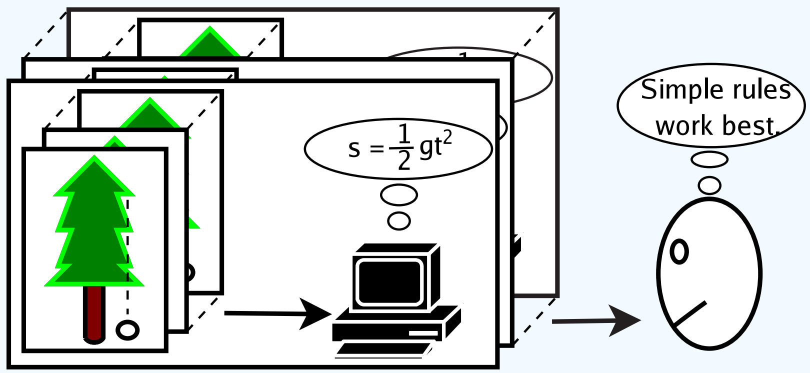
– Designs optimal experiments to validate hypotheses

– **Realizes the experiments physically**

– *King et al, Nature vol. 427, 2004*

Meta-Discovery

:: Viewing computer-based scientific discovery as an empirical phenomenon



:: Inferring hypotheses thereabout.

Phase Transitions

:: Originally: runtime statistics of **problem solving** algorithms on randomly generated problem instances. Example: propositional logic SATisfiability.

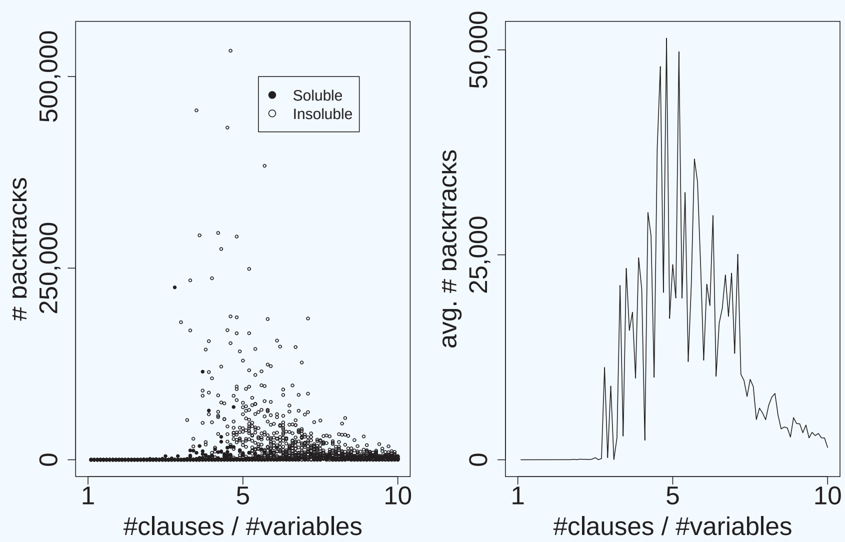


Figure 1: The NP-complete logic satisfiability problem. Algorithm: Davis-Putnam search

:: ← **Under**constrained (many solutions) vs. → **Over**constrained (small search) Hardest problems on the transition between.

Phase Transitions in Learning?

:: "**Inductive Logic Programming**" (ILP):
First-Order Logic representation of Data / Hypotheses.

:: Example: **Biochemistry**. Predicting mutagenic activity by compound structure.

:: Example Hypothesis

$$active(A) \leftarrow atm(A, B, c, 10, C) \wedge atm(A, D, c, 10, C) \wedge bond(A, B, D, 1)$$

:: **Verifying** the rule for given examples (chem. compounds) $\equiv$ SAT problem

:: Empirical studies *Serra et al, IJCAI 01; Botta et al, JMLR 4:2003*:
ILP systems tend to generate hypotheses in the Phase Transition region.

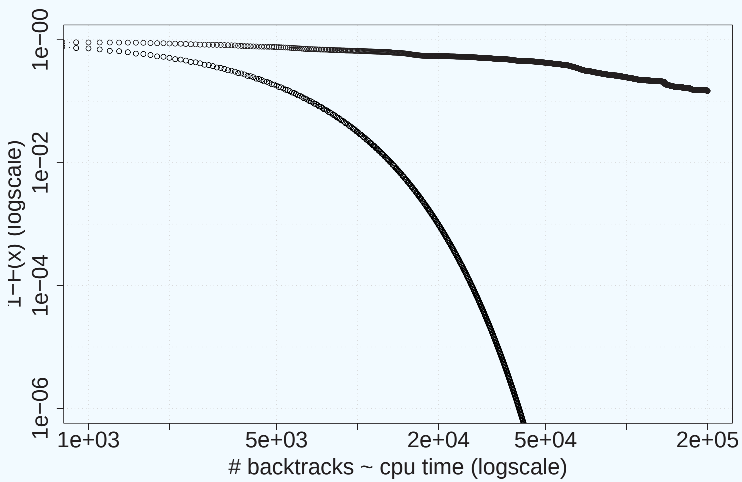
Heavy-Tailed Runtime Distributions

:: What goes on in the PT region? Model runtime distributions.

:: P(not achieving solution in time t)

– **normal**: decays exponentially with t

– **heavy-tailed** decays by power-law (may have infinite moments, eg. mean)



Heavy-Tailed Runtime Distributions

:: HT Distribs: "Statistical Curiosity", early 20th century:

– V. Pareto: Income Distributions,

– B. Mandelbrot: Fractal Phenomena in Nature

:: Empirical finding *Gomes et al, Jr Autom Reas 2001*

Important **combinatorial problems/algorithms** exhibit **heavy-tailed RTD**.

Surveyed randomized algorithms AND/OR problem instances

:: In hypothesis learning: *Zelezny et al, ILP 2002*

Heavy-tailed RTD's manifest themselves **in ILP**.

– Not only a consequence of involved hypothesis checking (=SAT)

– HT RTD also in terms of # of hypotheses searched

Restarted Randomized Search

:: HT RTD: Intriguing consequences.

– $\frac{f(t)\Delta t}{1-F(t)}$ prob finding a solution in the next Δt if not found until now $= t$.

– Decreases with t.

– The longer you search, the lower your chances...

:: Makes sense to **restart** search every now and then ?!

:: Indeed,

– Non-Restarted search RT cdf

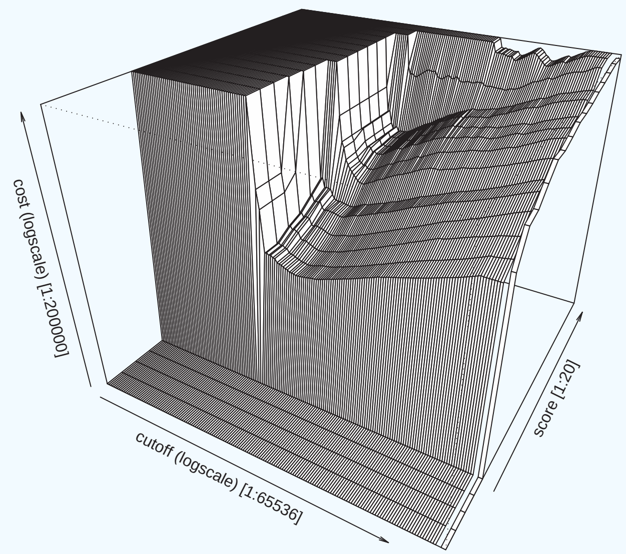
$$F(t)\text{: infinite mean, but } F(\gamma) > 0 \text{ for some } \gamma > 0$$

– Search restarted each time γ ("cut-off") time achieved. RT cdf:

$$F_\gamma(N) = 1 - (1 - F(\gamma))^N\text{: exponential} \Rightarrow \text{finite mean}$$

Restarted Randomized Search in ILP

:: Expected **runtime** of ILP algorithm with restart **cut-off** time γ, to find hypothesis of given **quality**. Log-scale, orders of magnitude performance gains.



:: Large empirical study (*Zelezny et al, ILP 2004*):

– 100-200 Condor Cluster PC's – UW Madison

– SGI Altix SuperComputer – CTU Prague

Occam's razor: Empirical Assessment

:: William of Ockham, 14^{th} century English logician.

"Entities should not be multiplied beyond necessity."

:: Traditional Machine Learning interpretation

"If several hypotheses explain data with roughly same accuracy, keep the simplest."

:: Reasons:

1. Evident: ease of human interpretation
2. Postulated: predictive ability (theory does not give a clue)

:: Thanks to automated discovery, Reason 2 can be empirically tested.

Occam's razor: Empirical Assessment

:: Some seminal empirical studies (*Holte, Mach Learn 1993*) apparently support the simplicity bias, but misinterpretation here.

:: Detrimental effect on predictive accuracy due to

– **too many** hypotheses tested

– rather than **too complex** hypotheses tested

Relation hyp space size / avg hyp complexity only incidental

:: *Domingos, Data Mining & Know Disc 1999* reviews empirical evidence againts Reason 2 for Occam's razor. Successes of

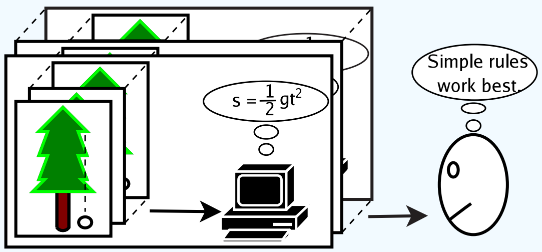
– Ensemble Learning (combining numerous complex hypotheses)

– Support Vector Machines (transforming data to high dimensional spaces)

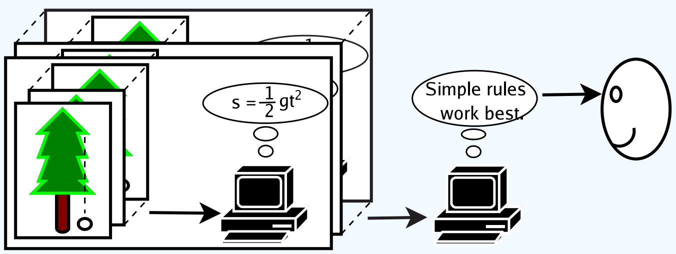
– Excessive search leading to simple inaccurate hyps *Quinlan et al, IJCAI 1995*

Computerized Meta-Learning

:: So far:



:: Now shifting to



Meta-Learning: "Learn how to learn best"

Meta-Learning Achievements

:: Traditional approaches: see *Mach Learn spec issue Meta Learning, 54:2004*. Examples:

– Meta-hypothesize on Which learning algorithm best for given data?

– Predict range of parameters (eg. kernel-width for SVM's) given meta-data.

:: Unorthodox approaches: *Maloberti, Sebag: Mach Learn 55(2):2004*

– Detect position of problem w.r.t the Phase Transition region

– Use to determine the best learning algorithm

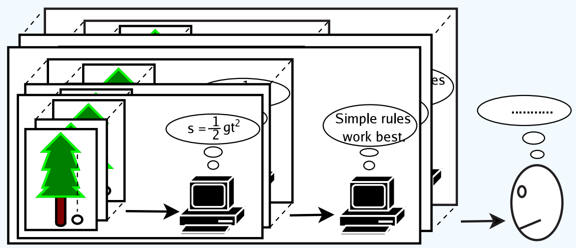
:: Other: *Bensusan, ECML 1998* meta-learns how much pruning should be used.

– Pruning $\approx$ simplifying hypotheses at some accuracy sacrifice

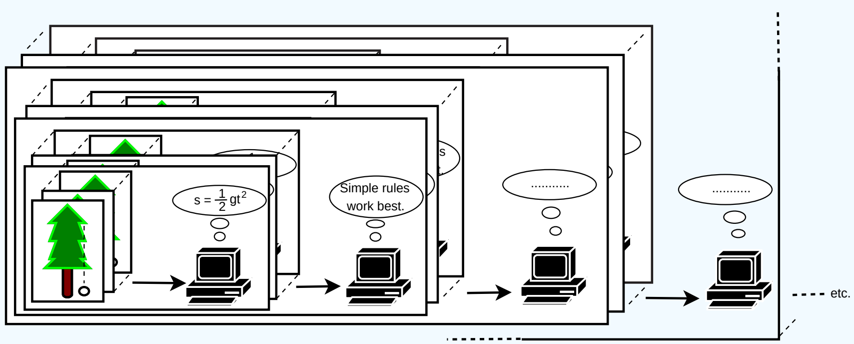
– Occam's razor motivated (title: "God does not always shave with Occam's razor")

Speculations

:: Given Meta-Learning is useful, would Meta-Meta-Learning be?



:: And $\underbrace{Meta - \ldots Meta-}_{n\times}$ learning?



What if n infinite? (much like Lisp/Prolog meta-interpretation *towers*)

Speculations

:: Recent research on links btw. **Machine Learning** and **Philosophy of Science** (eg. *ECML 2001 Dedicated Workshop*)

- mostly to improve Machine Learning
- note: also *Vapnik, Nat Stat Learn Th, Springer 2001* translates Popper's falsifiability thesis to learning theory.

:: But inversely: Do computerized meta-discovery lessons apply to scientific inquiry in general? Eg.

- Should scientist randomize and restart hypothesis forming?
- Should scientist combine diverse hypotheses to draw a conclusion?
- Should they devise overly complex hypotheses to generate variance needed for good ensembles?
- ...

*** THE END ***