AI, ML, RL, Rewards, Energy, Values

Michèle Sebag

TAO

RL with Generalized Feedback, Praha sept. 2013

Where are we coming from ?

ML family roots

- Statistical modelling
- Computational learning
- Artificial Intelligence

What are they called for ?

- Stats: first principles (what not to do)
- AI: where to go

AI research agenda

J. McCarthy 56

We propose a study of artificial intelligence [..]. The study is to proceed on the basis of the conjecture that every aspect of learning or any other feature of intelligence can in principle be so precisely described that a machine can be made to simulate it.

Before AI...

Alan Turing

Machine Learning, 1950

by (...) mimicking education, we should hope to modify the machine until it could be relied on to produce definite reactions to certain commands.

Before AI...

Alan Turing

Machine Learning, 1950

by (...) mimicking education, we should hope to modify the machine until it could be relied on to produce definite reactions to certain commands.

How ?

One could *carry through the organization of an intelligent machine with only two interfering inputs, one for pleasure or reward, and the other for pain or punishment.*

The imitation game

The criterion:

Whether the machine could answer *questions in such a way that it will be extremely difficult to guess whether the answers are given by a man, or by the machine*

Critical issue

The extent we regard something as behaving in an intelligent manner is determined as much by our own state of mind and training, as by the properties of the object under consideration.

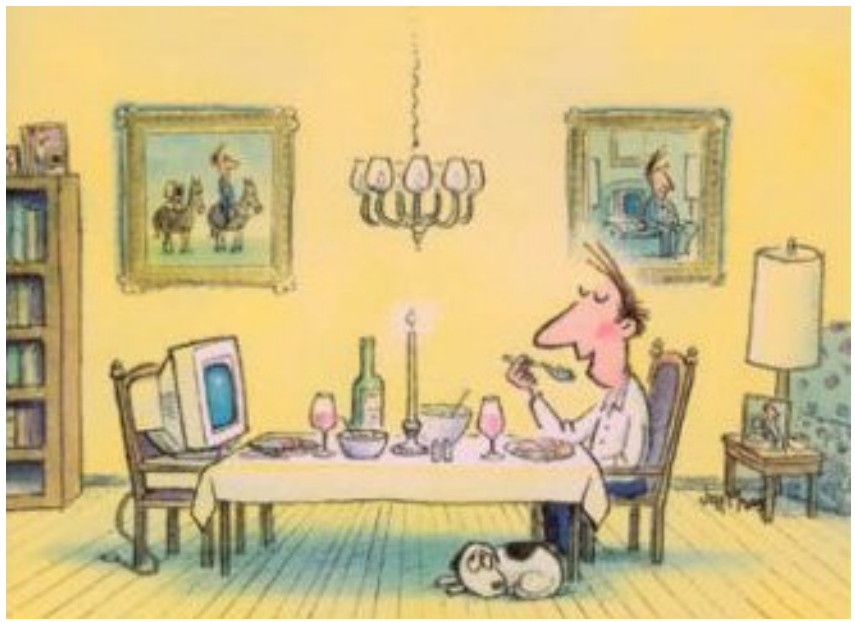
The imitation game, 2

A regret-like criterion

- Comparison to reference performance (oracle)
- More difficult task $\not\Rightarrow$ higher regret

Oracle = human being

- Social intelligence matters
- Weaknesses are OK.



But AI took another turn

- General Problem Solver
- Solving (any) puzzle

Lessons

Lenat 2001

the promise that *the more you know the more you can learn (..) sounds fine until you think about the inverse, namely, you do not start with very much in the system already. And there is not really that much that you can hope that it will learn completely cut off from the world*.

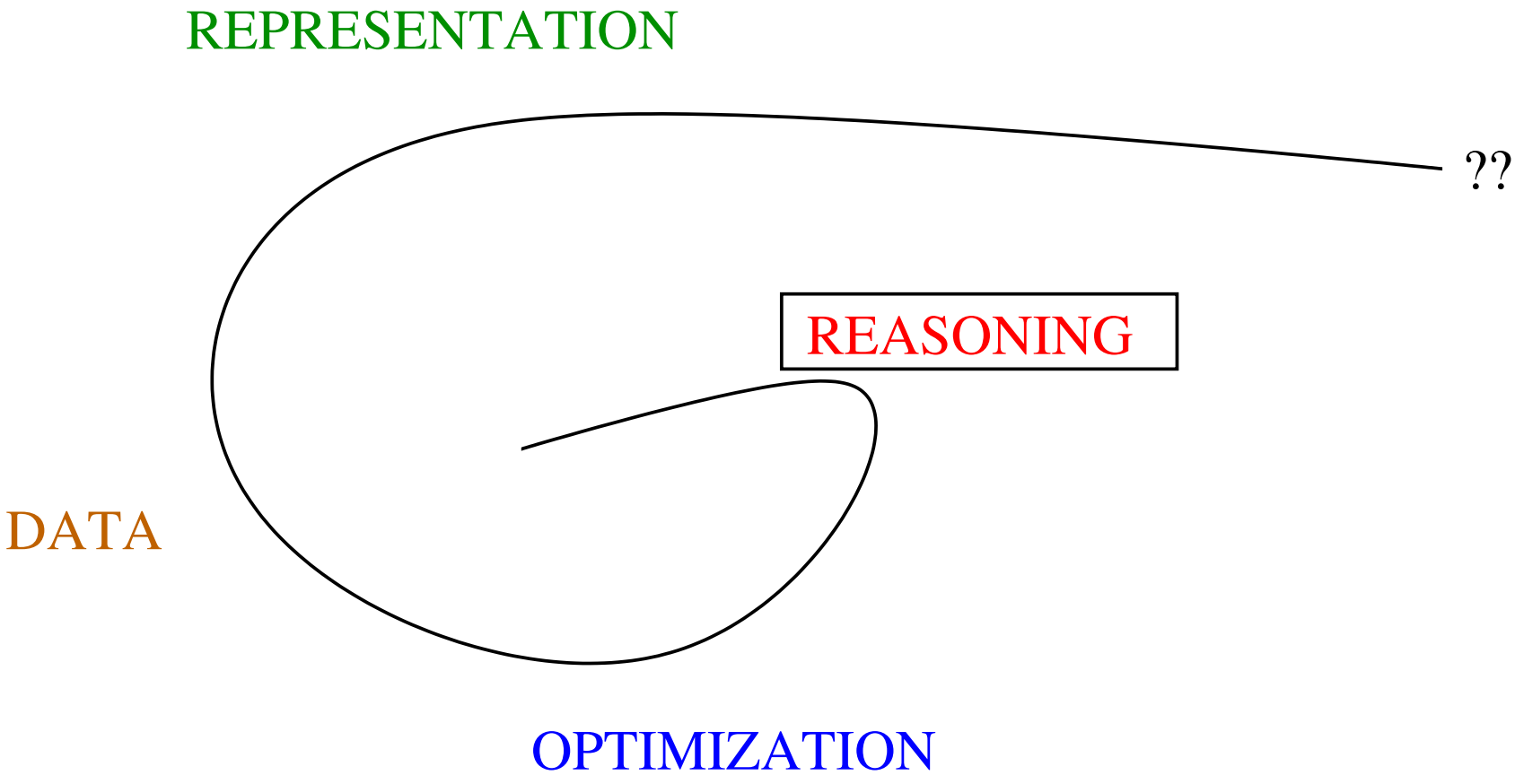
Interacting with the world is a must-have

Overview

Where are we coming from

What have we been doing

Some new directions


REPRESENTATION
??
REASONING
DATA
OPTIMIZATION

The Robot Scientist

King et al, 04, 11

The robot scientist: completes the cycle

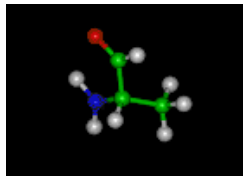
from hypothesis to experiment to reformulated hypothesis

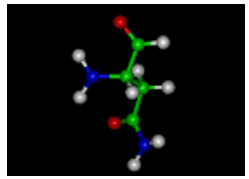
without human intervention.

The Robot Scientist, 2

Why does it work ?

- A proper representation

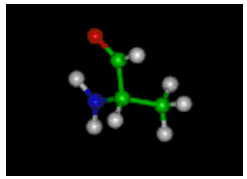


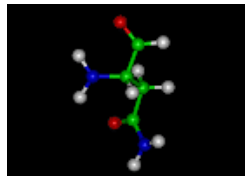


The Robot Scientist, 2

Why does it work ?

- A proper representation



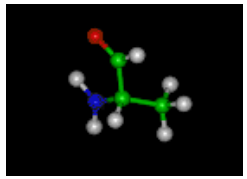


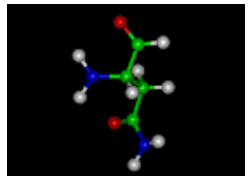
- Active Learning – Design of Experiment

The Robot Scientist, 2

Why does it work ?

- A proper representation



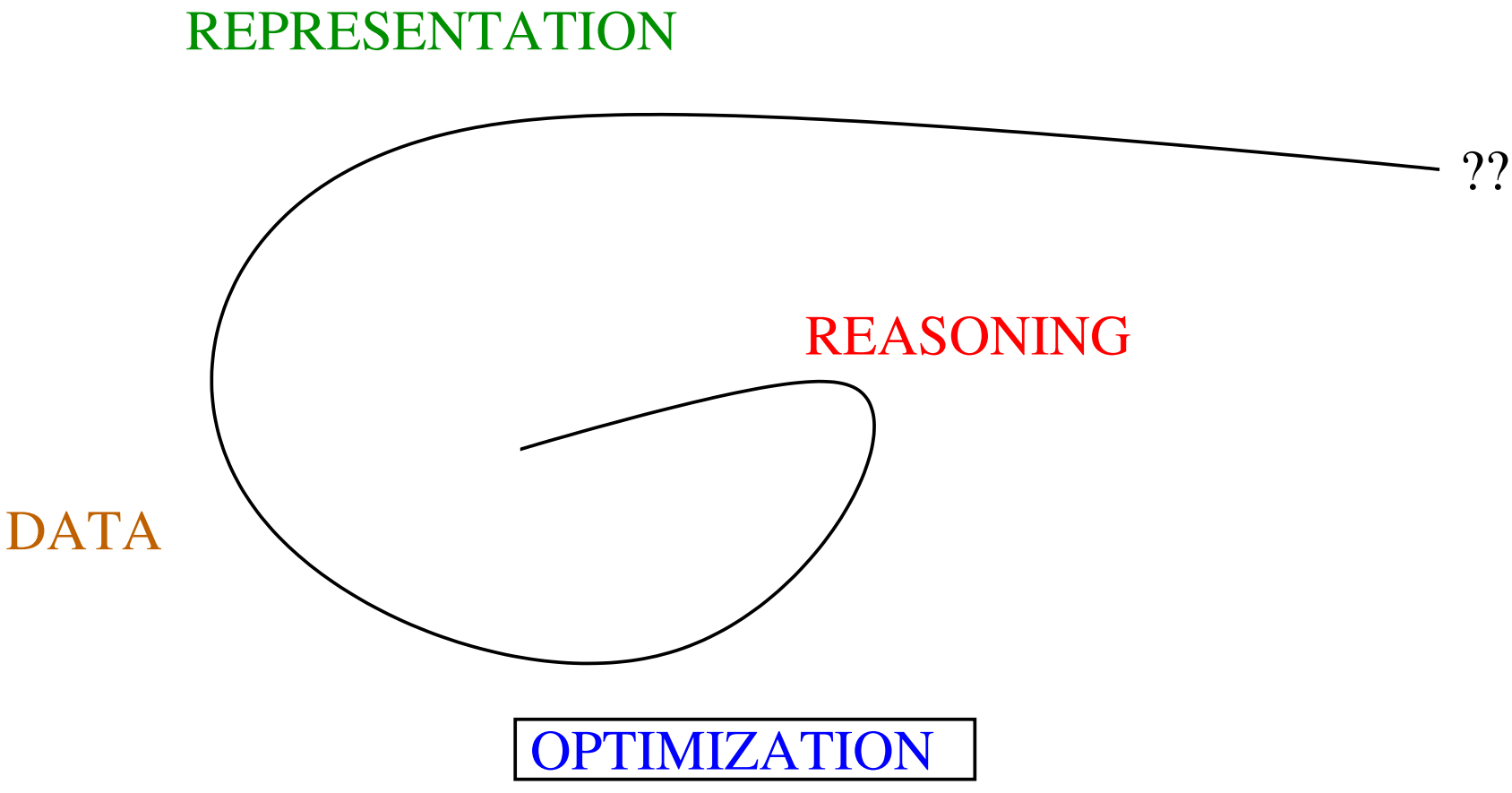


- Active Learning – Design of Experiment
- Control of noise

Does the Robot Scientist have a plan ?

1. From facts to conjectures
2. From conjectures to new experiments
3. From new experiments to facts

Any new sound conjecture is worth


REPRESENTATION
??
REASONING
DATA
OPTIMIZATION

ML second era: Optimization is everything

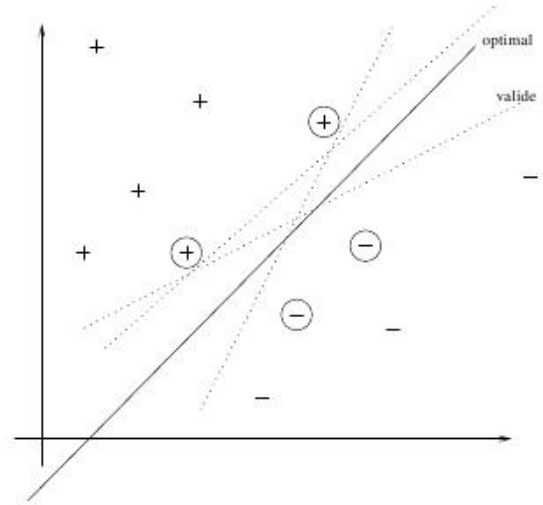
- Neural Nets: first success
- Support Vector Machines

Reading cheques

LeCun et al. 1990

Support Vector Machines

Not all separating hyperplanes are equal



Divine surprise: a quadratic optimization problem

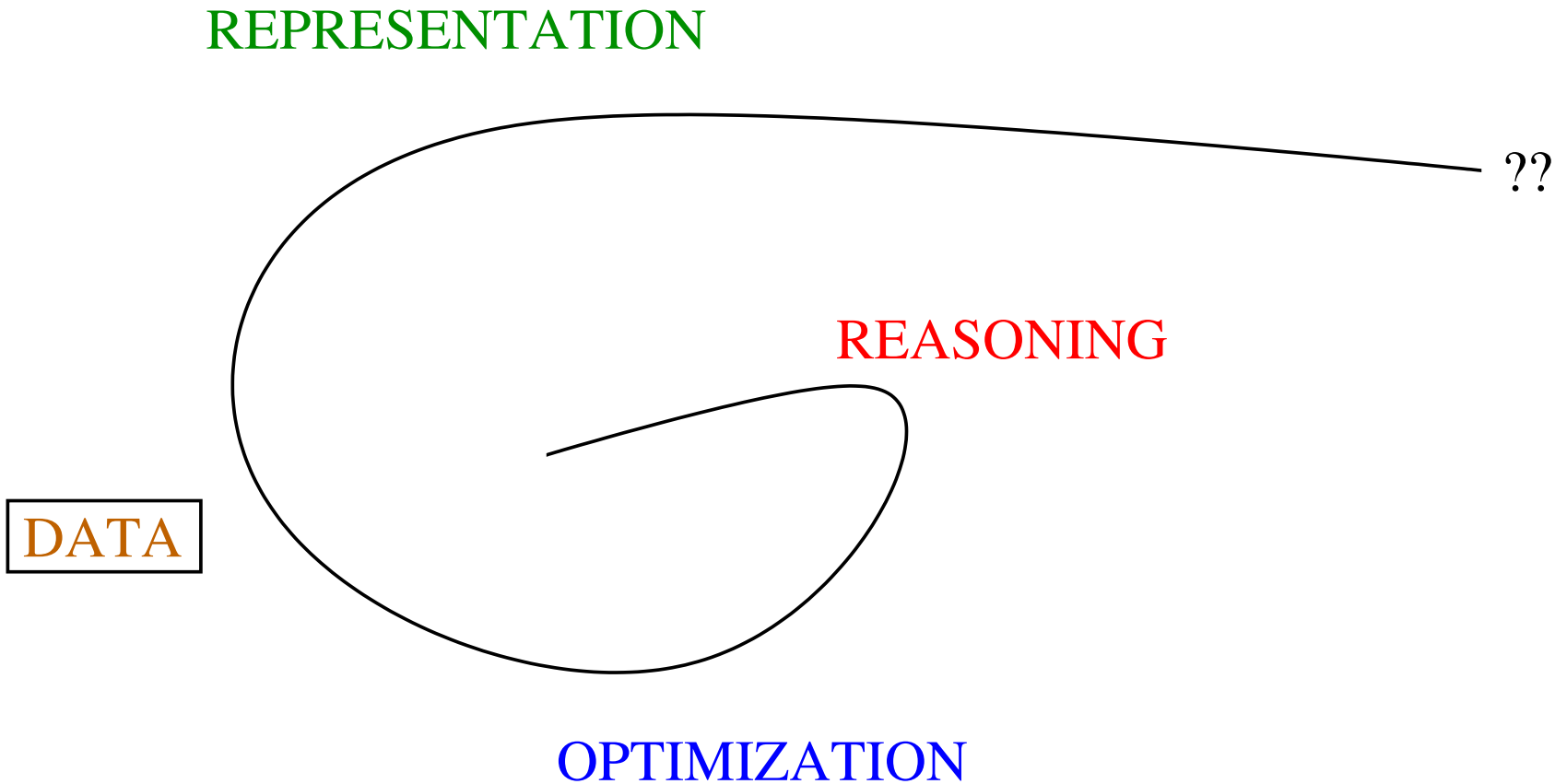
Boser et al. 92; Cortes et al. 95

$$\begin{cases} \text{Minimize} & \frac{1}{2}\ ||w||^2 \\ \text{subject to} & \forall\ i,\ y_i(\langle w, x_i\rangle + b) \geq 1 \end{cases}$$

But...

- Hyper-parameter calibration
- Problem reduction

Langford et al. 06


REPRESENTATION
??
REASONING
DATA
OPTIMIZATION

ML third era: all you need is more !

- More data
- More hypotheses
- (Does one still need reasoning ?)

All you need is more data

If algorithms are consistent — Daelemans 03

- When the data amount goes to infinity,
- ... all algorithms get same results

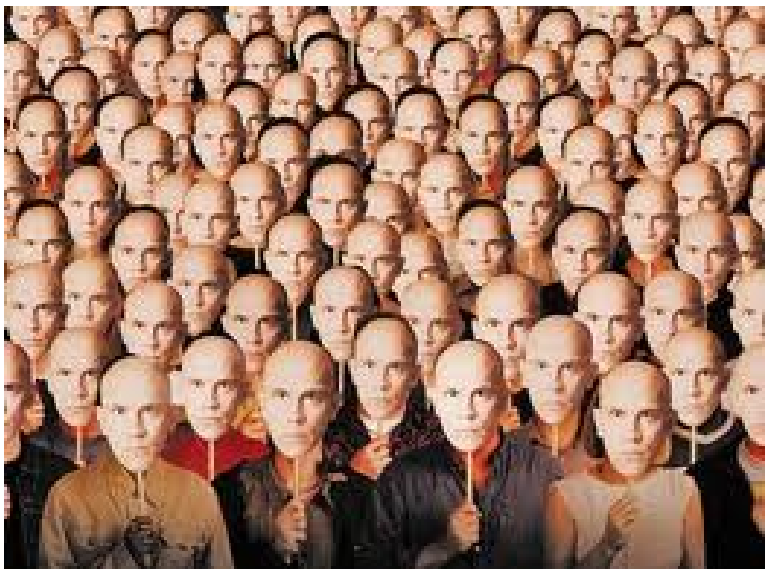
When data size matters

- Statistical machine translation
- The **textual entailment challenge** — Dagan et al. 05
  - Text: *Lyon is actually the gastronomic capital of France*
  - Hyp: *Lyon is the capital of France*
  - Does T entail H ?

All you need is more diversified hypotheses

Ensemble learning

- The strength of weak learnability — Schapire 90
- The wisdom of crowds



NO

YES

Is more data all we need ?

A thought experiment

Grefenstette, pers.

- The web: a world of information
- Question: what is the color of cherries ?

Is more data all we need ?

A thought experiment — Grefenstette, pers.

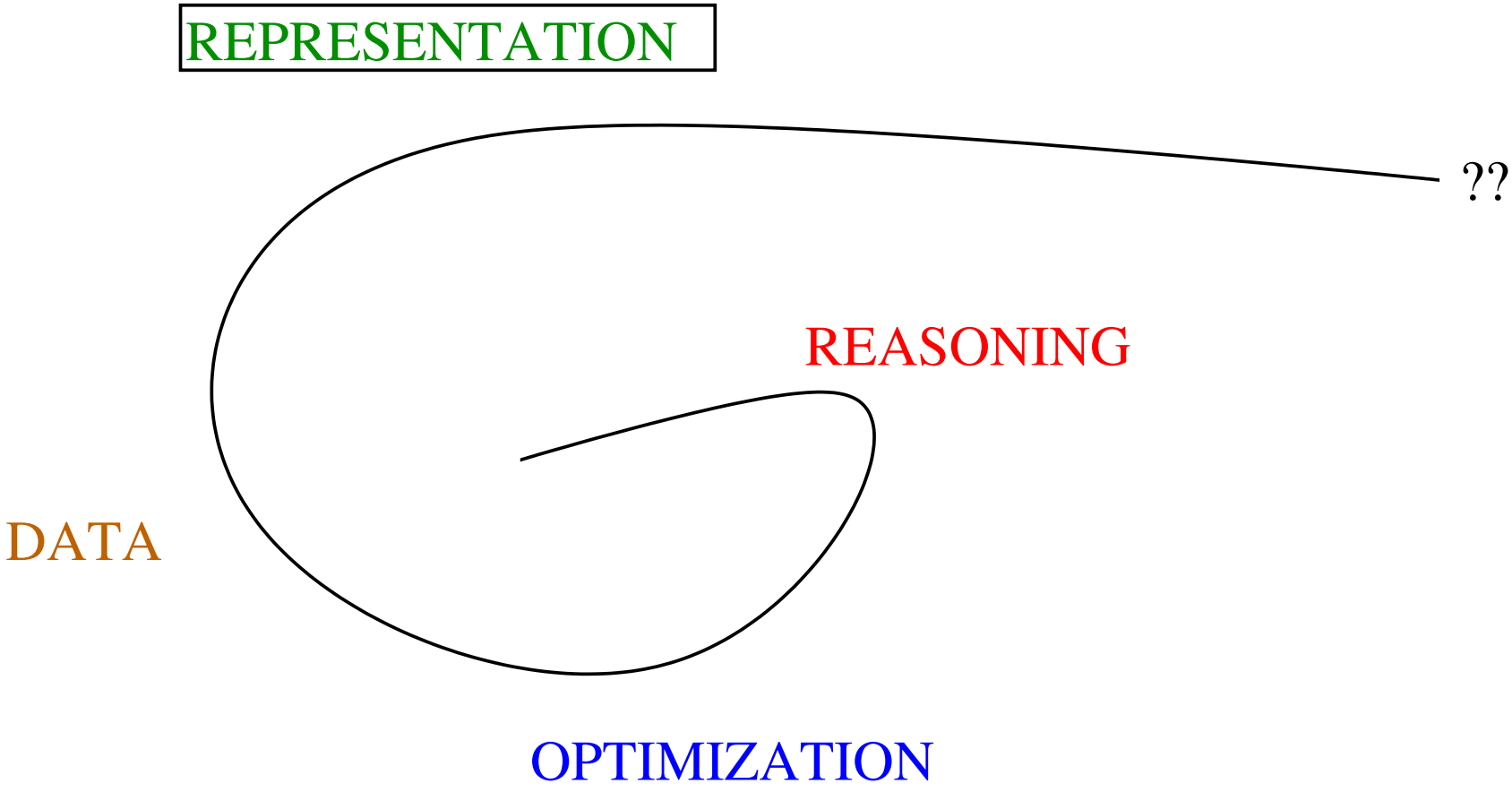
- The web: a world of information
- Question: what is the color of cherries ?
- After Google hits, 20% of cherries are black...

Is more data all we need ?

A thought experiment — Grefenstette, pers.

- The web: a world of information
- Question: what is the color of cherries ?
- After Google hits, 20% of cherries are black...

- Something else is needed...


REPRESENTATION
??
REASONING
DATA
OPTIMIZATION

Representation is everything

- Bayesian nets — Pearl 00
- Deep Networks — Hinton et al. 06, Bengio et al. 06
- Dictionary learning — Donoho et al. 05; Mairal et al. 10

Causality: Models, Reasoning and Inference

Pearl 2000

- associational inference
  what if I see X ?
  evidential or statistical reasoning
- interventional inference
  what if I do X ?
  experimental or causal reasoning
- retrospectional inference
  what if I had not done X ?
  counterfactual reasoning

Issue

- Learning the structure

Cussens 08

Deep Networks

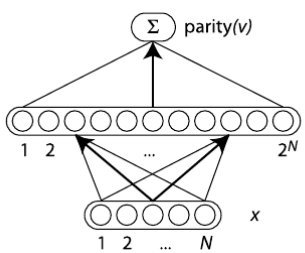
Hinton et al. 06, Bengio et al. 06

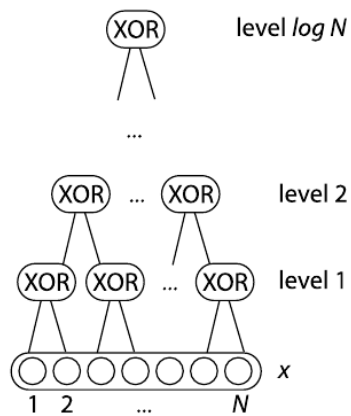
Grand goal

- Using ML to reach AI: (...) understanding of high-level abstractions
- Trade-off: computational, statistical, student-labor efficiency

Bottleneck

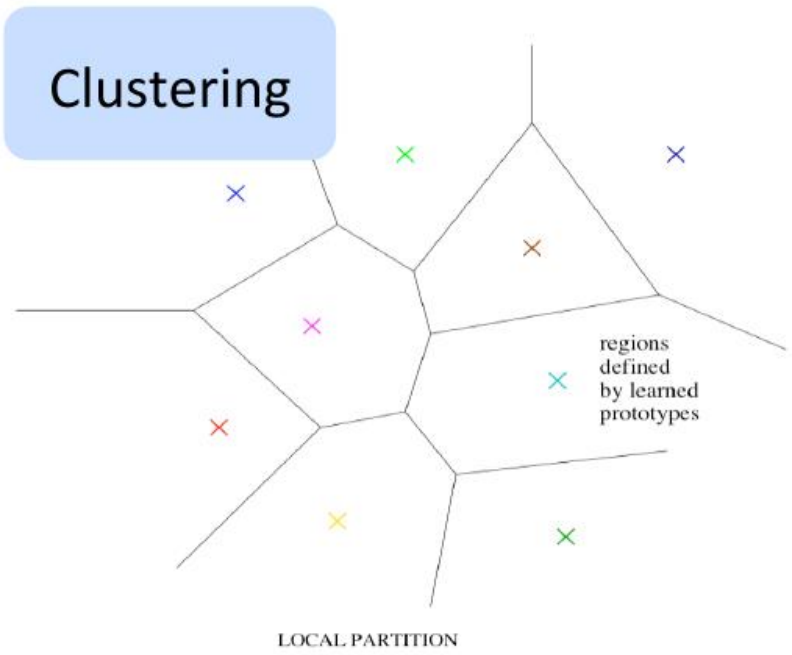
- Pattern matchers: partition the space
- Inefficient at representing highly varying functions

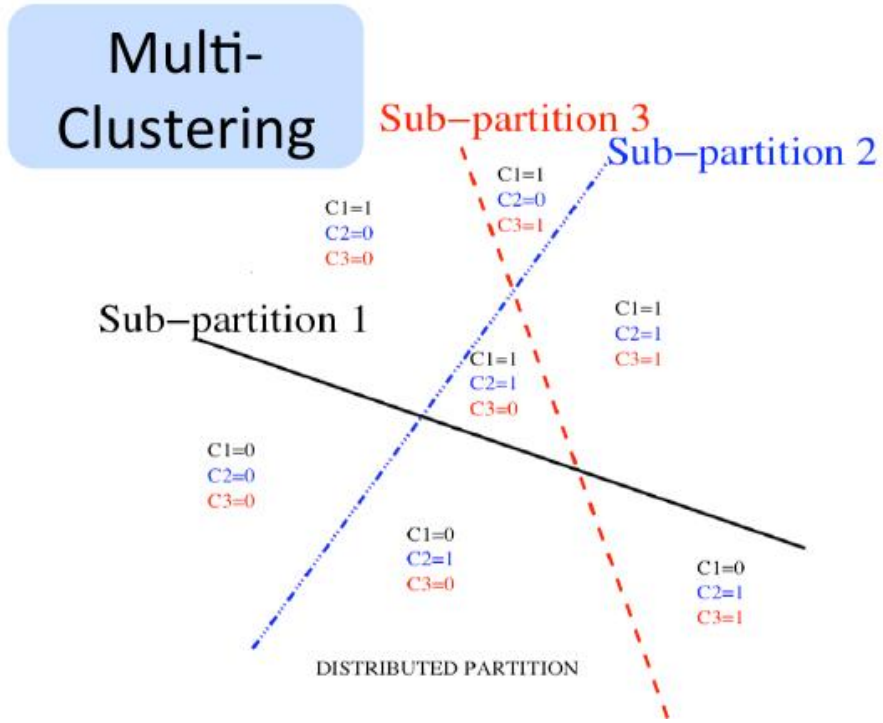




Deep Networks, 2/3

Bengio 12





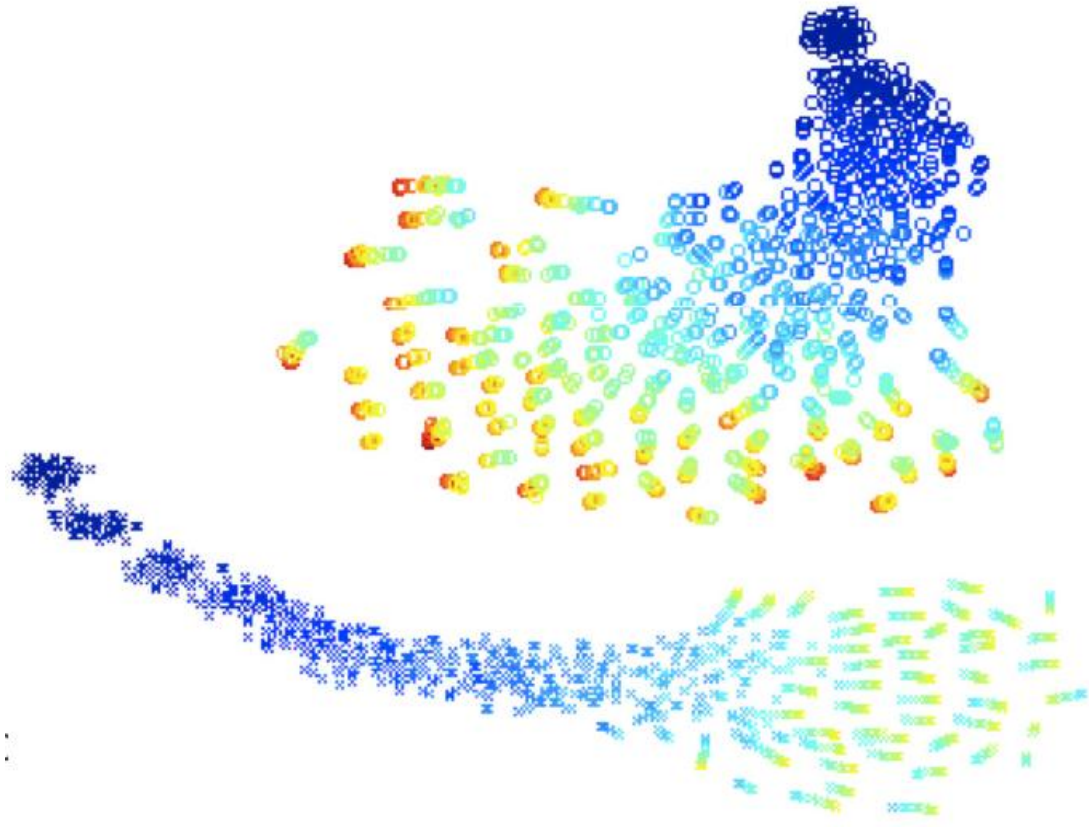
From prototypes to features

N regions

2^N regions

Deep Networks, 3/3

Top: 2d projection of the NN weights, no pre-training
Bottom: 2d projection of the NN weights, with pre-training



Reducing variance

Erhan et al. 09

Dictionary Learning

Principle

- A large dictionary, where you can express your thoughts in few words
- Robustness against noise
- Given data Y, find dictionary D and coefficient α s.t.

$$\min ||Y - D\alpha|| + ||\alpha||_1$$

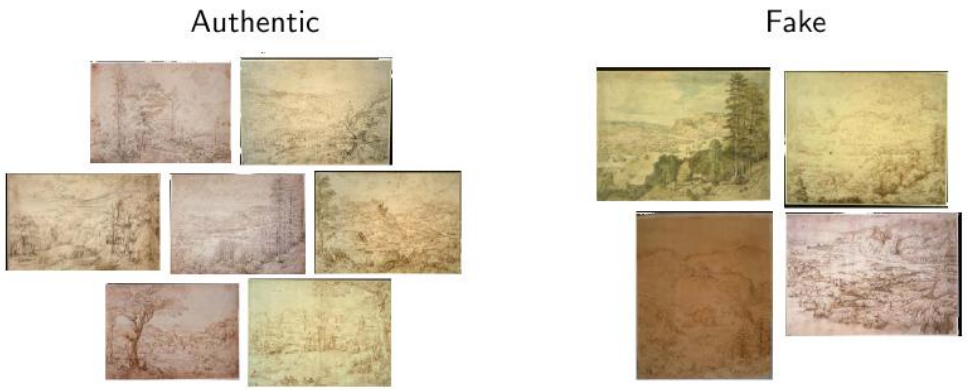
Dictionary Learning

Principle

- A large dictionary, where you can express your thoughts in few words
- Robustness against noise
- Given data Y, find dictionary D and coefficient α s.t.

$$\min ||Y - D\alpha|| + ||\alpha||_1$$

Hugues et al. 09; Mairal et al. 10



Dictionary Learning

Principle

- A large dictionary, where you can express your thoughts in few words
- Robustness against noise
- Given data Y, find dictionary D and coefficient α s.t.

$$\min ||Y - D\alpha|| + ||\alpha||_1$$

Hugues et al. 09; Mairal et al. 10

Dictionary Learning

Principle

- A large dictionary, where you can express your thoughts in few words
- Robustness against noise
- Given data Y, find dictionary D and coefficient α s.t.

$$\min ||Y - D\alpha|| + ||\alpha||_1$$

Hugues et al. 09; Mairal et al. 10

Finding the optimization objective *vs* optimizing it

- **The NIPS community has suffered of an acute convexivitis epidemic**
  - ML applications seem to have trouble moving beyond logistic regression, SVMs, and exponential-family graphical models.
  - For a new ML model, convexity is viewed as a virtue
  - Convexity is sometimes a virtue
  - But it is often a limitation
  - ML theory has essentially never moved beyond convex models
    - the same way control theory has not really moved beyond linear systems
  - Often, the price we pay for insisting on convexity is an unbearable increase in the size of the model, or the scaling properties of the optimization algorithm [O(n^2), O(n^3)...]

`http://videolectures.net/eml07_lecun_wia/`

Overview

Where are we coming from

What have we been doing

Some new directions

Decomposition of tasks and recomposition of solutions

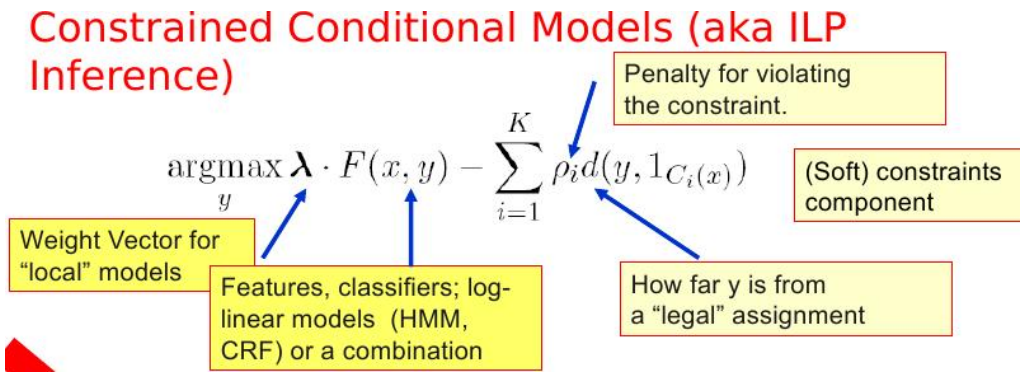
Kundu et al. 12-13, Roth et al. 13

Natural Language Decisions are Structured

Global decisions in which several local decisions play a role but there are mutual dependencies on their outcome.

It is essential to make coherent decisions in a way that takes the interdependencies into account.

How to support real, high level, natural language decisions How to learn models that are used, eventually, to make global decisions:

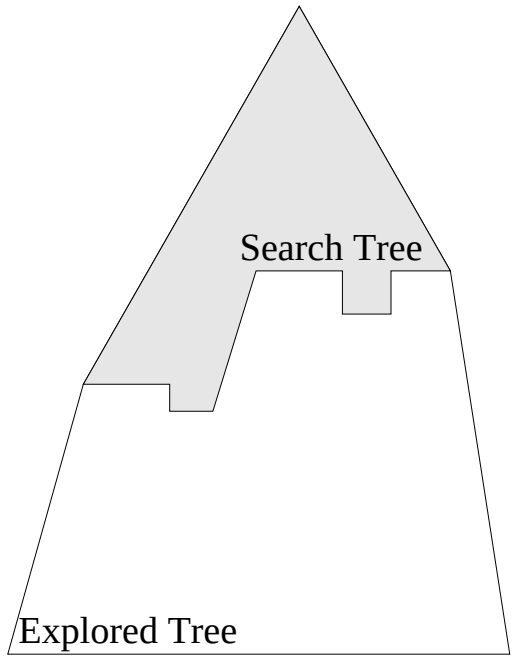


Monte-Carlo Tree Search

Kocsis Szepesvári, 06; Gelly Silver 07

Gradually grow the search tree:

- Iterate Tree-Walk
  - Building Blocks
    - Select next action
      Bandit phase
    - Add a node
      Grow a leaf of the search tree
    - Select next action bis
      Random phase, roll-out
    - Compute instant reward
      Evaluate
    - Update information in visited nodes
      Propagate
- Returned solution:
  - Path visited most often

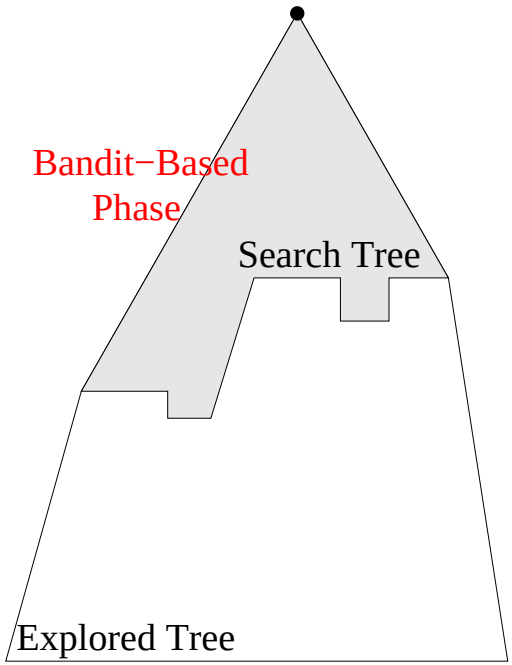


Monte-Carlo Tree Search

Kocsis Szepesvári, 06; Gelly Silver 07

Gradually grow the search tree:

- Iterate Tree-Walk
  - Building Blocks
    - Select next action
      Bandit phase
    - Add a node
      Grow a leaf of the search tree
    - Select next action bis
      Random phase, roll-out
    - Compute instant reward
      Evaluate
    - Update information in visited nodes
      Propagate
- Returned solution:
  - Path visited most often

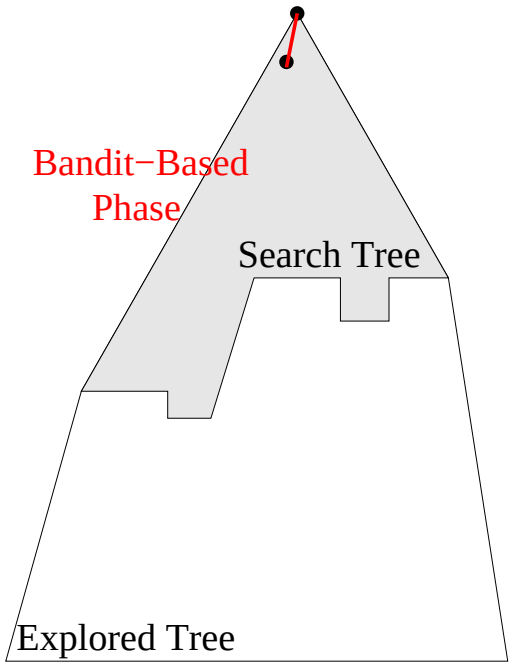


Monte-Carlo Tree Search

Kocsis Szepesvári, 06; Gelly Silver 07

Gradually grow the search tree:

- Iterate Tree-Walk
  - Building Blocks
    - Select next action
      Bandit phase
    - Add a node
      Grow a leaf of the search tree
    - Select next action bis
      Random phase, roll-out
    - Compute instant reward
      Evaluate
    - Update information in visited nodes
      Propagate
- Returned solution:
  - Path visited most often

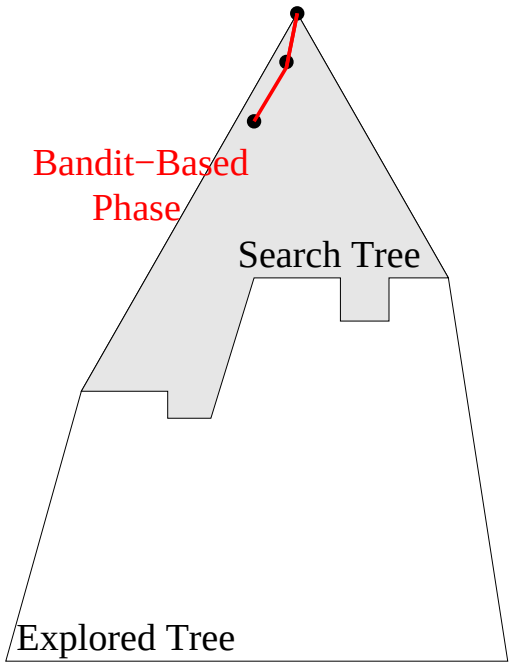


Monte-Carlo Tree Search

Kocsis Szepesvári, 06; Gelly Silver 07

Gradually grow the search tree:

- Iterate Tree-Walk
  - Building Blocks
    - Select next action
      Bandit phase
    - Add a node
      Grow a leaf of the search tree
    - Select next action bis
      Random phase, roll-out
    - Compute instant reward
      Evaluate
    - Update information in visited nodes
      Propagate
- Returned solution:
  - Path visited most often

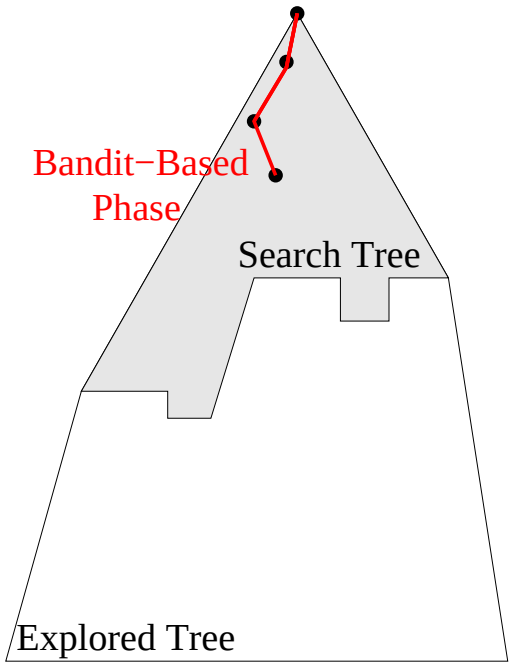


Monte-Carlo Tree Search

Kocsis Szepesvári, 06; Gelly Silver 07

Gradually grow the search tree:

- Iterate Tree-Walk
  - Building Blocks
    - Select next action
      Bandit phase
    - Add a node
      Grow a leaf of the search tree
    - Select next action bis
      Random phase, roll-out
    - Compute instant reward
      Evaluate
    - Update information in visited nodes
      Propagate
- Returned solution:
  - Path visited most often

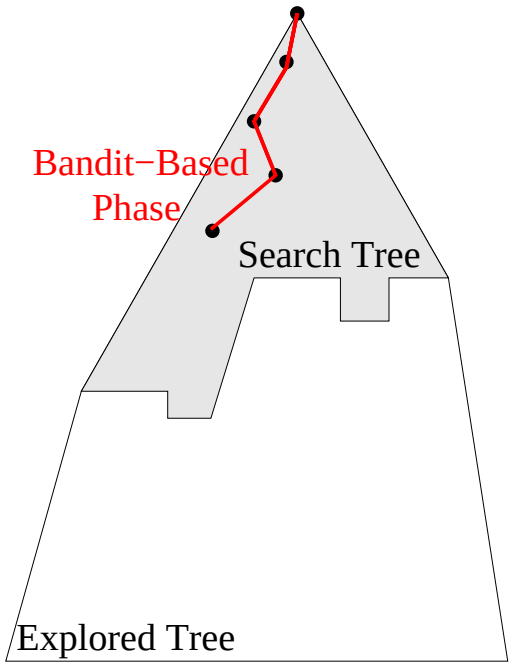


Monte-Carlo Tree Search

Kocsis Szepesvári, 06; Gelly Silver 07

Gradually grow the search tree:

- Iterate Tree-Walk
  - Building Blocks
    - Select next action
      Bandit phase
    - Add a node
      Grow a leaf of the search tree
    - Select next action bis
      Random phase, roll-out
    - Compute instant reward
      Evaluate
    - Update information in visited nodes
      Propagate
- Returned solution:
  - Path visited most often

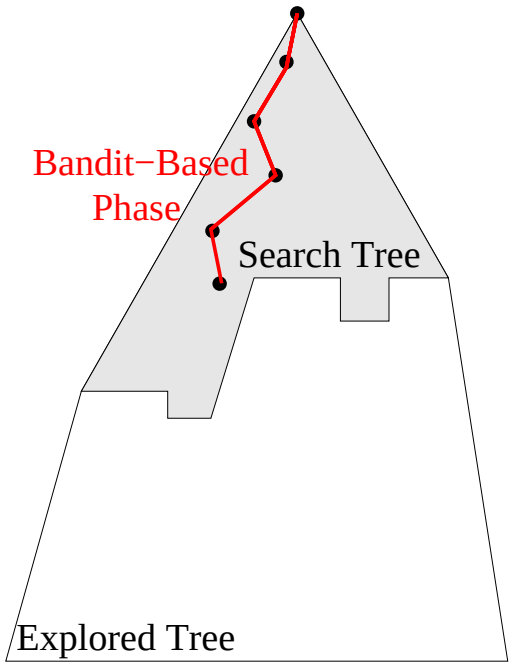


Monte-Carlo Tree Search

Kocsis Szepesvári, 06; Gelly Silver 07

Gradually grow the search tree:

- Iterate Tree-Walk
  - Building Blocks
    - Select next action
      Bandit phase
    - Add a node
      Grow a leaf of the search tree
    - Select next action bis
      Random phase, roll-out
    - Compute instant reward
      Evaluate
    - Update information in visited nodes
      Propagate
- Returned solution:
  - Path visited most often

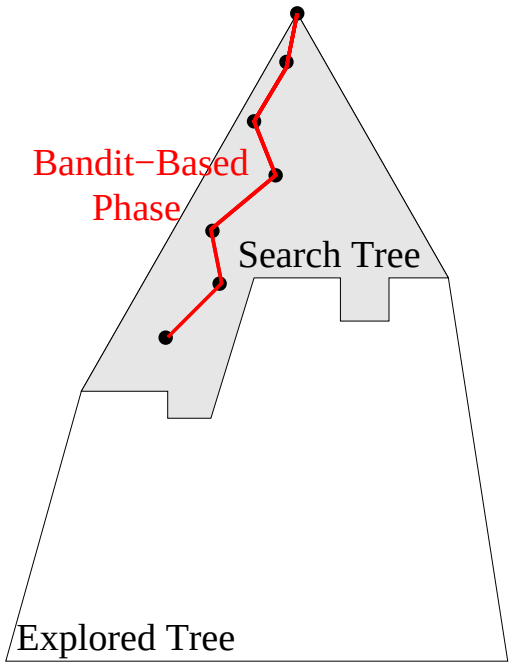


Monte-Carlo Tree Search

Kocsis Szepesvári, 06; Gelly Silver 07

Gradually grow the search tree:

- Iterate Tree-Walk
  - Building Blocks
    - Select next action
      Bandit phase
    - Add a node
      Grow a leaf of the search tree
    - Select next action bis
      Random phase, roll-out
    - Compute instant reward
      Evaluate
    - Update information in visited nodes
      Propagate
- Returned solution:
  - Path visited most often

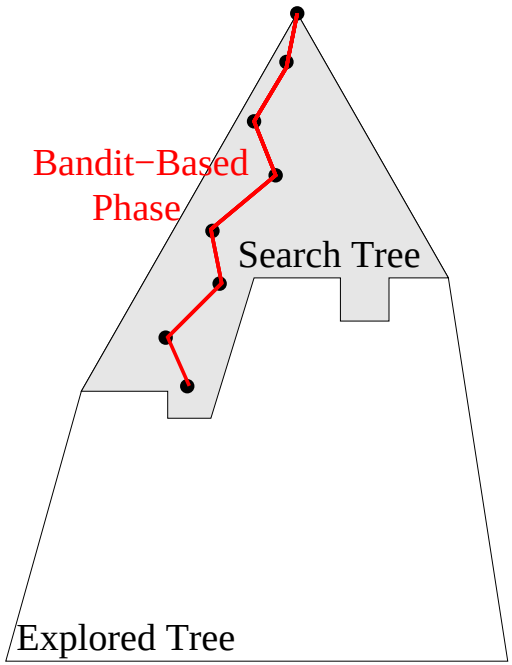


Monte-Carlo Tree Search

Kocsis Szepesvári, 06; Gelly Silver 07

Gradually grow the search tree:

- Iterate Tree-Walk
  - Building Blocks
    - Select next action
      Bandit phase
    - Add a node
      Grow a leaf of the search tree
    - Select next action bis
      Random phase, roll-out
    - Compute instant reward
      Evaluate
    - Update information in visited nodes
      Propagate
- Returned solution:
  - Path visited most often

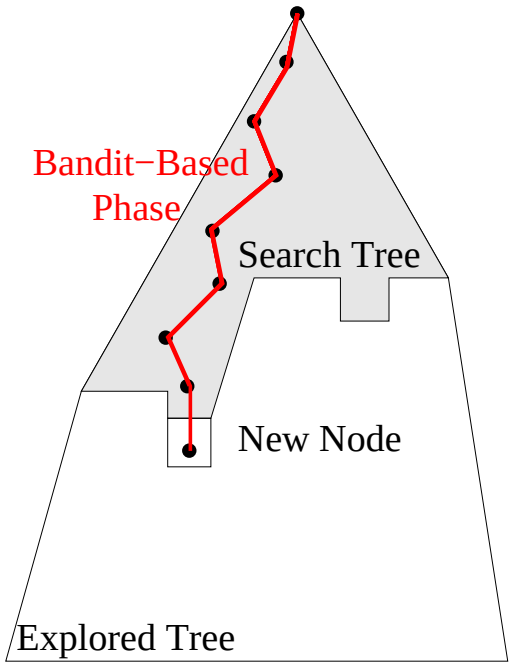


Monte-Carlo Tree Search

Kocsis Szepesvári, 06; Gelly Silver 07

Gradually grow the search tree:

- Iterate Tree-Walk
  - Building Blocks
    - Select next action
      Bandit phase
    - Add a node
      Grow a leaf of the search tree
    - Select next action bis
      Random phase, roll-out
    - Compute instant reward
      Evaluate
    - Update information in visited nodes
      Propagate
- Returned solution:
  - Path visited most often

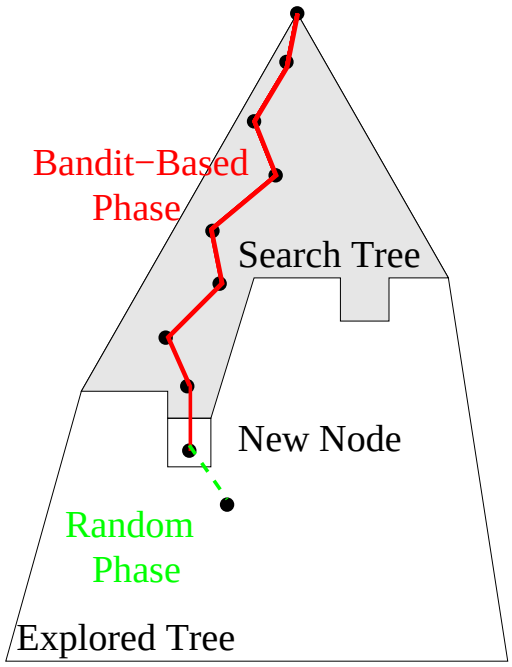


Monte-Carlo Tree Search

Kocsis Szepesvári, 06; Gelly Silver 07

Gradually grow the search tree:

- Iterate Tree-Walk
  - Building Blocks
    - Select next action
      Bandit phase
    - Add a node
      Grow a leaf of the search tree
    - Select next action bis
      Random phase, roll-out
    - Compute instant reward
      Evaluate
    - Update information in visited nodes
      Propagate
- Returned solution:
  - Path visited most often

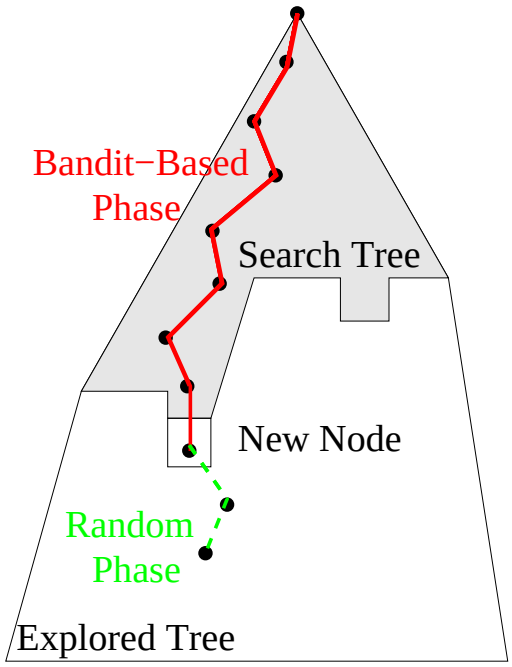


Monte-Carlo Tree Search

Kocsis Szepesvári, 06; Gelly Silver 07

Gradually grow the search tree:

- Iterate Tree-Walk
  - Building Blocks
    - Select next action
      Bandit phase
    - Add a node
      Grow a leaf of the search tree
    - Select next action bis
      Random phase, roll-out
    - Compute instant reward
      Evaluate
    - Update information in visited nodes
      Propagate
- Returned solution:
  - Path visited most often

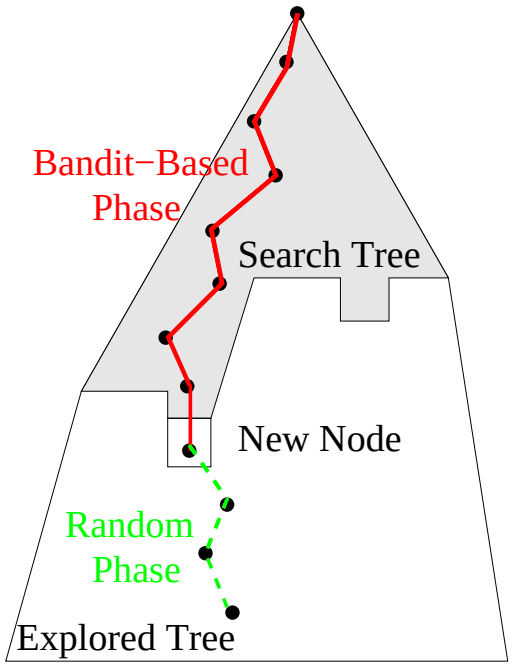


Monte-Carlo Tree Search

Kocsis Szepesvári, 06; Gelly Silver 07

Gradually grow the search tree:

- Iterate Tree-Walk
  - Building Blocks
    - Select next action
      Bandit phase
    - Add a node
      Grow a leaf of the search tree
    - Select next action bis
      Random phase, roll-out
    - Compute instant reward
      Evaluate
    - Update information in visited nodes
      Propagate
- Returned solution:
  - Path visited most often

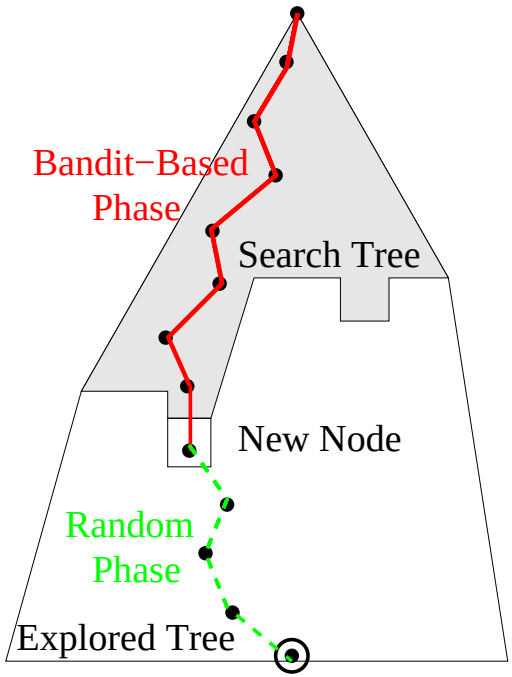


Monte-Carlo Tree Search

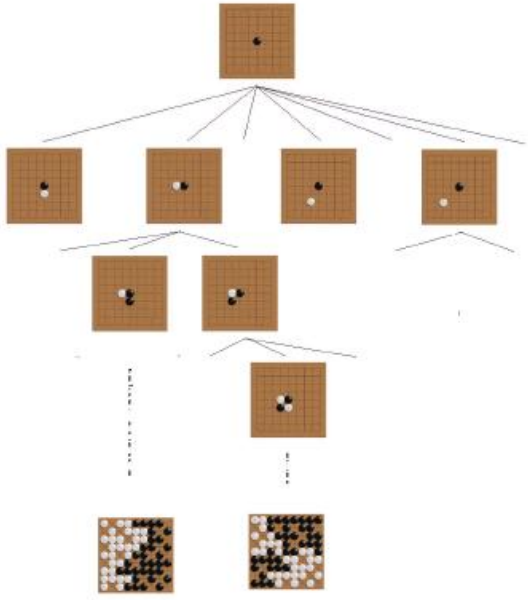
Kocsis Szepesvári, 06; Gelly Silver 07

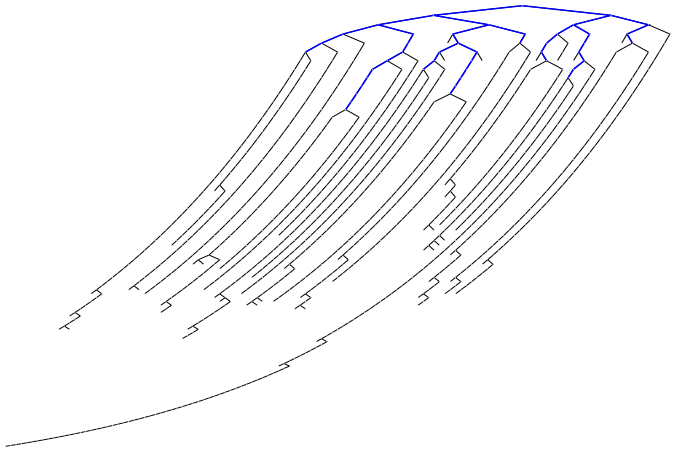
Gradually grow the search tree:

- Iterate Tree-Walk
  - Building Blocks
    - Select next action
      Bandit phase
    - Add a node
      Grow a leaf of the search tree
    - Select next action bis
      Random phase, roll-out
    - Compute instant reward
      Evaluate
    - Update information in visited nodes
      Propagate
- Returned solution:
  - Path visited most often



MCTS, some applications





computer-Go
MoGo

constraint programming
in Gecode

Reward

- 1/0 win/loss
- Ad hoc: e.g. depth failure in BASCOP Loth et al. 2013

Strategy vs tactic

- More tree simulations vs a better rollout policy Couetoux et al. 13

Energy-based learning

Le Cun et al., 06

"Energy-Based Models capture dependencies between variables by associating a scalar energy to each configuration of the variables. Inference consists in clamping the value of observed variables and finding configurations of the remaining variables that minimize the energy. Learning consists in finding an energy function in which observed configurations of the variables are given lower energies than unobserved ones. The EBM approach provides a common theoretical framework for many learning models (...)".

See also: Heess et al, 2012

Programming by optimizing

Hoos et al. 12

PbO "allows human experts to focus on the creative task of imagining possible mechanisms for solving given problems or subproblems, while the tedious job of determining what works best in a given use context is performed automatically, substituting human labor with computation. "

How

Learn $\hat{\mathcal{F}}$(pb. param; alg. hyper-param) $\approx$ performance

http://www.prog-by-opt.net/

Human competitive (Humies) award

GECCO 2012

- Yavalath: an automatically designed game
- more popular than Backgammon and Chinese Checkers

What was the optimization objective ? C. Browne

uncertainty; killer moves; permanence; completion; duration (negative)

Human competitive (Humies) award

GECCO 2012

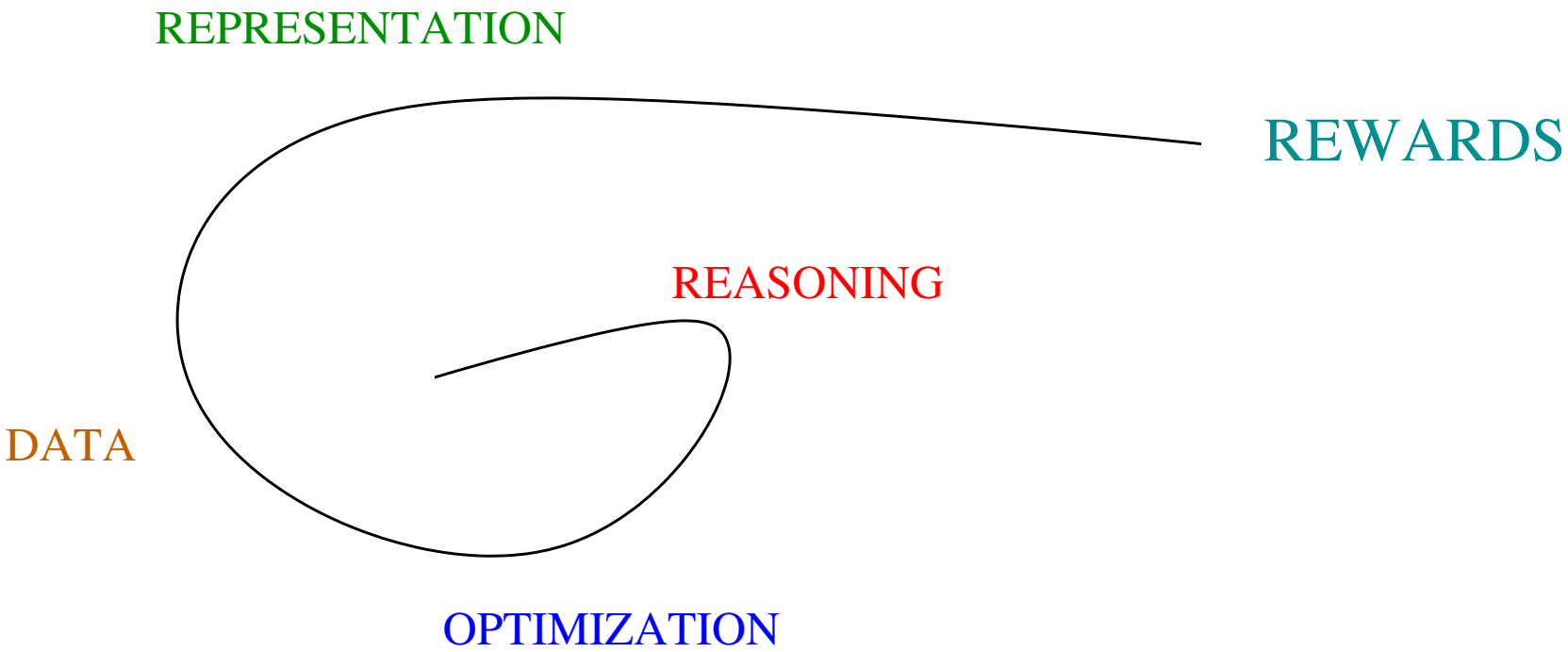
- Yavalath: an automatically designed game
- more popular than Backgammon and Chinese Checkers

What was the optimization objective ? C. Browne

uncertainty; killer moves; permanence; completion; duration (negative)

Then what should an AI system learn ?

Learn the objective


REPRESENTATION
REWARDS
REASONING
DATA
OPTIMIZATION

Call to arms

Some key ML steps handled by trial and error, including

- Priors
- Hyper-parameters, algorithm selection
- Management of the processing loop

if only the interestingness function were known...

Call to arms

Some key ML steps handled by trial and error, including

- Priors
- Hyper-parameters, algorithm selection
- Management of the processing loop

if only the interestingness function were known...

We can learn it !

Call to arms

Some key ML steps handled by trial and error, including

- Priors
- Hyper-parameters, algorithm selection
- Management of the processing loop

if only the interestingness function were known...

We can learn it !

- Non stationary preferences; noise; sampling complexity
- Rewards or Values ?