

# What is the place of Reasoning in Machine Learning?

---

Antoine Cornuéjols

*AgroParisTech – INRA MIA 518*

EKINOCS research group

# Outline

---

1. What is reasoning (1)
2. Machine Learning nowadays: a path to fast thinking
3. The future of Machine Learning: what is reasoning (2)
4. Conclusion

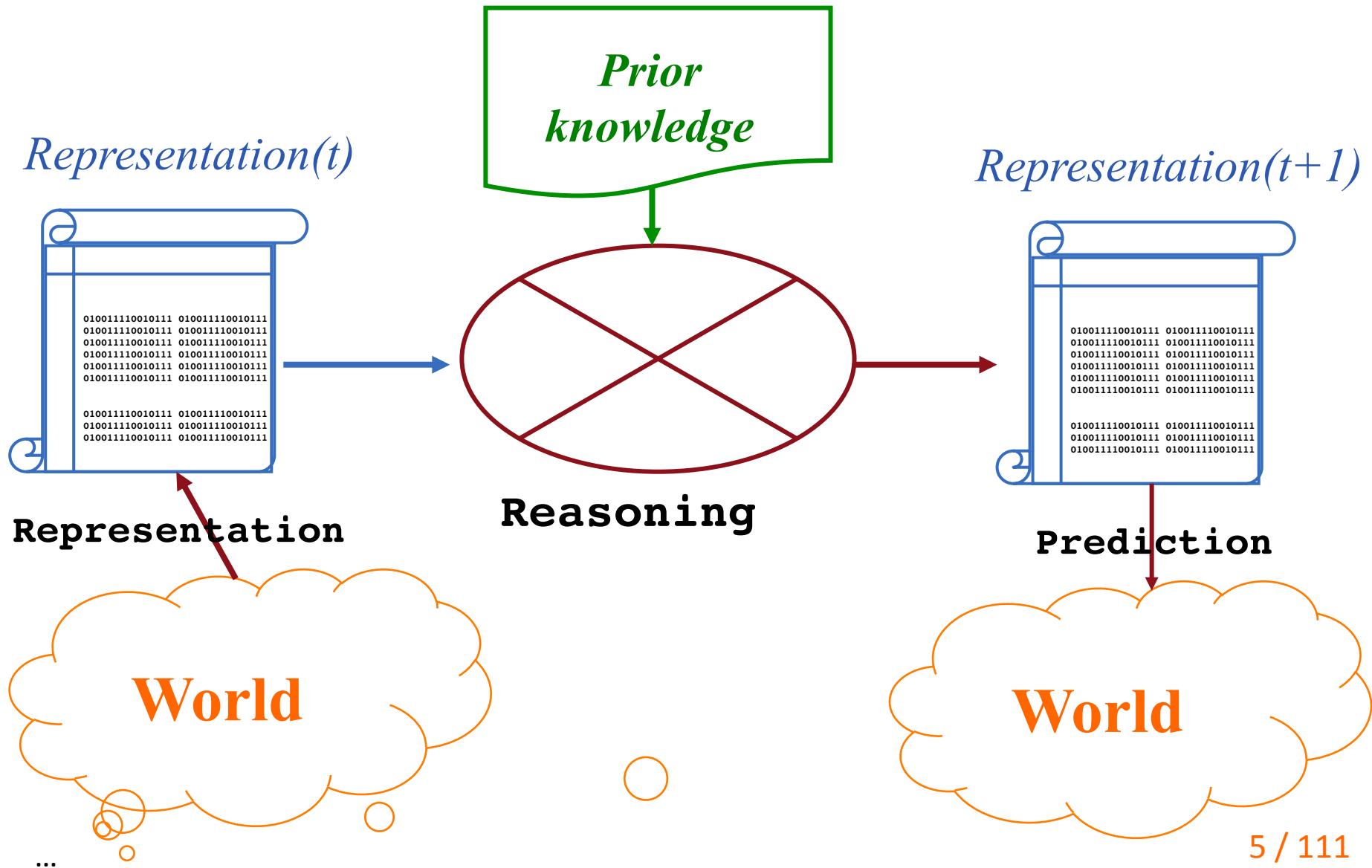
---

What is reasoning (1)

# Reasoning

---

- 1. Manipulate representations**
  
- 2. To solve a problem**

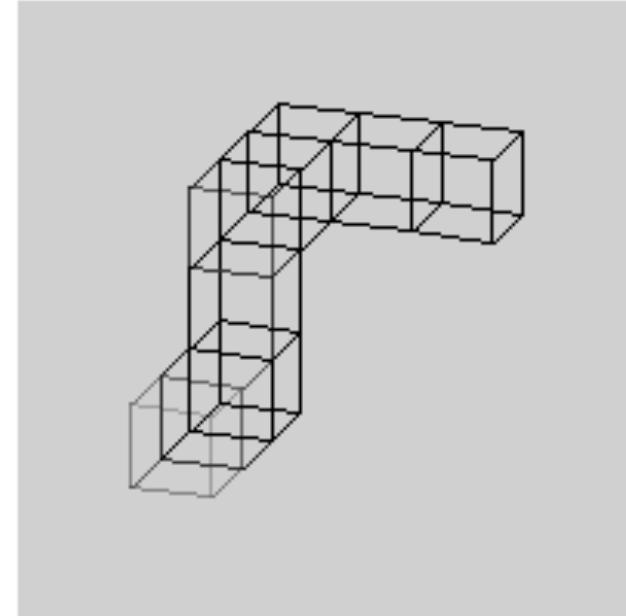
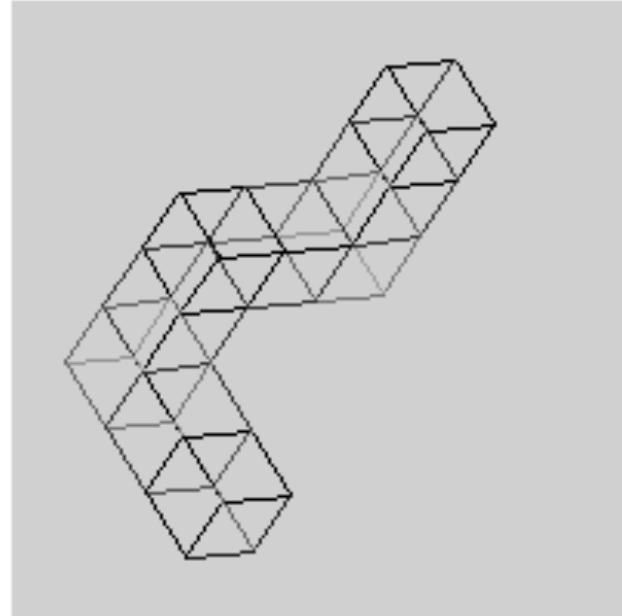


## Many types of representations and of operations

---

- Experiments of Shepard

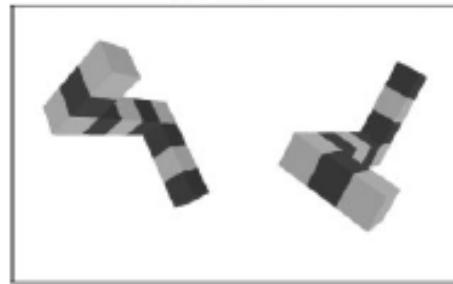
<http://www.ulb.ac.be/psycho/fr/docs/museum/Experiments/Shepard/Shepard.html>



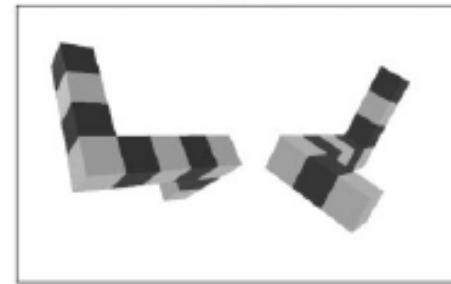
# Many types of representations and of operations

---

mirror

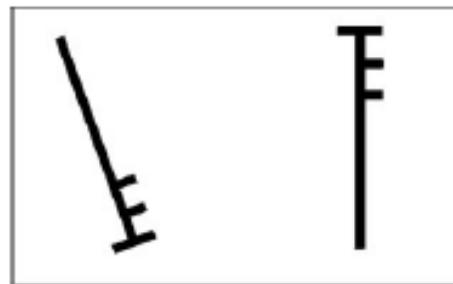
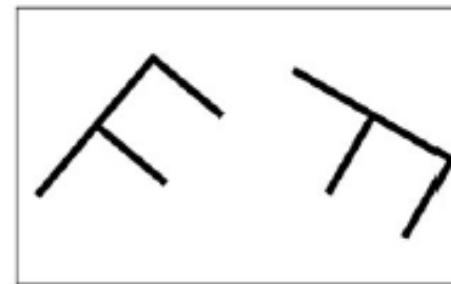


same

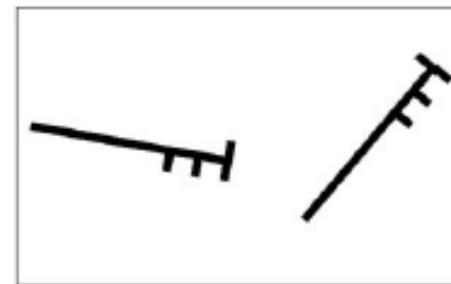


3D

letter



abstract

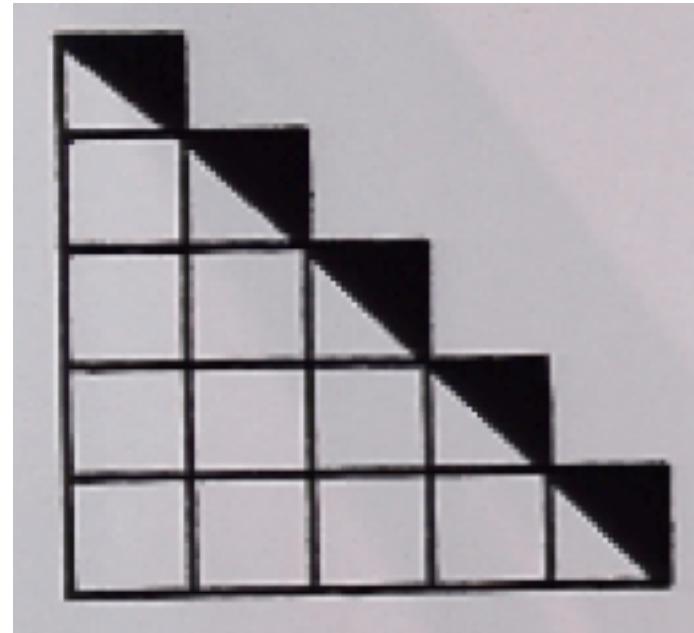


...

## Many types of representations and of operations

---

$$1 + 2 + 3 + \dots + n = \frac{n^2}{2} + \frac{n}{2}$$

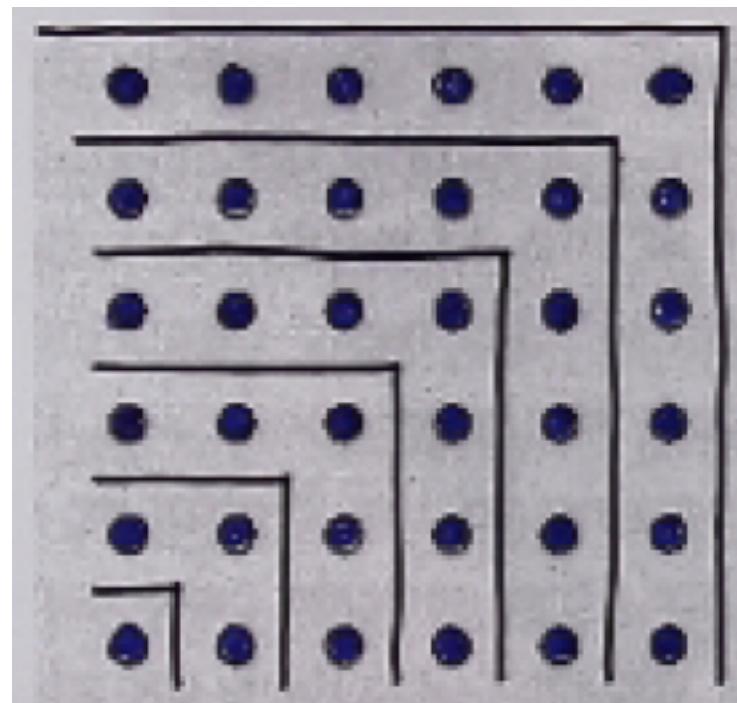


...

## Many types of representations and of operations

---

$$1 + 3 + 5 + \dots + (2n - 1) = n^2$$



...

- 
1. On a boat, suppose that two measurements of **distance** are made **three minutes apart**, and that **1500 yards** have been covered
  2. A **nautical mile** = 2000 yards

What is the speed of the boat in knots (nautical miles / hour)?

---

- Solution 1

$$d = v \times t \Rightarrow v = \frac{d}{t}$$

(prior K on algebra)

- $d = 1500 \text{ yards} = 1500/2000 \text{ mille}$
- $t = 3\text{mn} = 3/60 \text{ hour}$

►  $v = 0.75 / 0.05 = 15 \text{ miles/h}$

Rks :

- *Operations can be carried out in different orders*
- *They must be organized*

- 
- Solution 2
    - Same but without external memory in order to store the intermediate results
    - The difficulty lies in ordering the operations

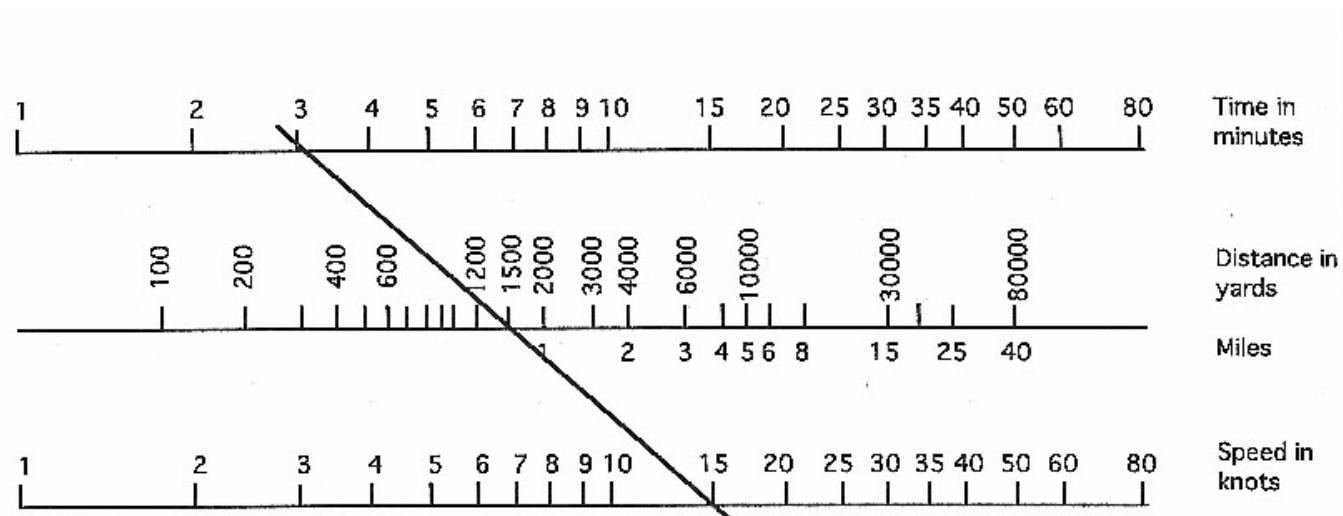


Figure 3.3 A three-scale nomogram.

- Solution 3

- Use of a nautical rule
- Draw a line between the *time* and the *distance* to get the speed
- The representation allows one to get the result easily
- But it is highly specialized

---

- Solution 4: the 3mn rule

- $3 \text{ mn} = 1/20 \text{ hour}$
  - $100 \text{ yards} = 1/20 \text{ mile}$
  - You just have to **remove the 2 last digits of the distance** (e.g. 1500 yards) to get the speed -> 15 knots
  - But the measurements have to be made three minutes apart

## Lesson

---

- A reasoning system
  - Must chose an **appropriate representation**
  - And select and organize the **right operations**

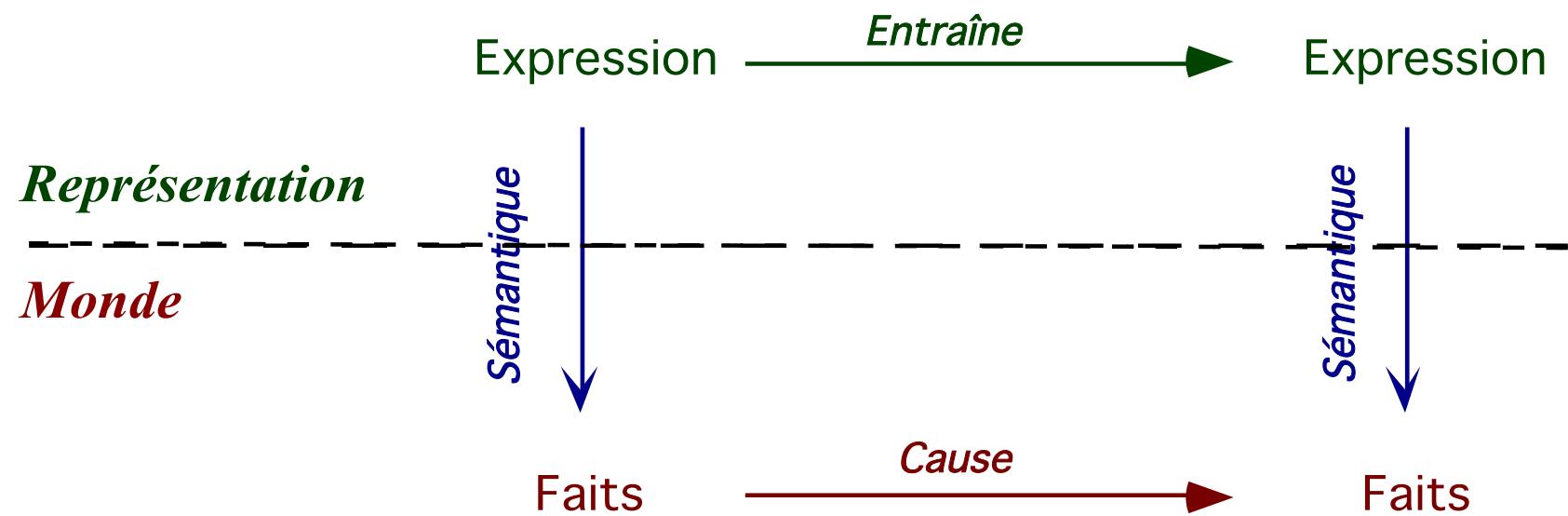
## Blending effect [Fauconnier & Turner]

---

### The Riddle of the Buddhist Monk:

- A Buddhist monk begins **at dawn** one day **walking up a mountain**, reaches the top **at sunset**, meditates at the top overnight until, **at dawn**, he begins to **walk back** to the foot of the mountain, which he reaches **at sunset**.
- Make no assumptions about his starting or stopping or about his pace during the trips.
- Riddle: *is there a place on the path that the monk occupies at the same hour of the day on the two trips?*

# Reasoning



...

# Reasoning

---

- Entails
  - Inferencing information that is **not explicitly given** in the initial representation
  - Jack is married either to Francesca or to Sophie  
Sophie is not married  
 $\Rightarrow \textit{Jack is married to Francesca}$
  - John looks at Isabel and Isabel looks at Kevin  
John is married, and Kevin is not married  
 $\Rightarrow \textit{A married person looks at a non married person}$

## Many forms of reasoning

---

- Deduction
  - Induction
  - Abduction
  - Analogy
  - Blending
  - Probabilistic reasoning
  - ...
- With rules that define  
legitimate operations
- Provide some guarantees  
on the results

# Propositional logics: inference rules

---

- Modus ponens

$$\frac{\alpha \Rightarrow \beta, \alpha}{\beta}$$

- AND-elimination

$$\frac{\alpha_1 \wedge \alpha_2 \wedge \dots \wedge \alpha_n}{\alpha_i}$$

- AND-introduction

$$\frac{\alpha_1, \alpha_2, \dots, \alpha_n}{\alpha_1 \wedge \alpha_2 \wedge \dots \wedge \alpha_n}$$

- OR-introduction

$$\frac{\alpha_i}{\alpha_1 \vee \alpha_2 \vee \dots \vee \alpha_n}$$

- Elimination double negation

$$\frac{\neg\neg\alpha}{\alpha}$$

- Unitary Resolution

$$\frac{\alpha \vee \beta, \neg\beta}{\alpha}$$

- Resolution

$$\frac{\alpha \vee \beta, \neg\beta \vee \gamma}{\alpha \vee \gamma}$$

## Reasoning

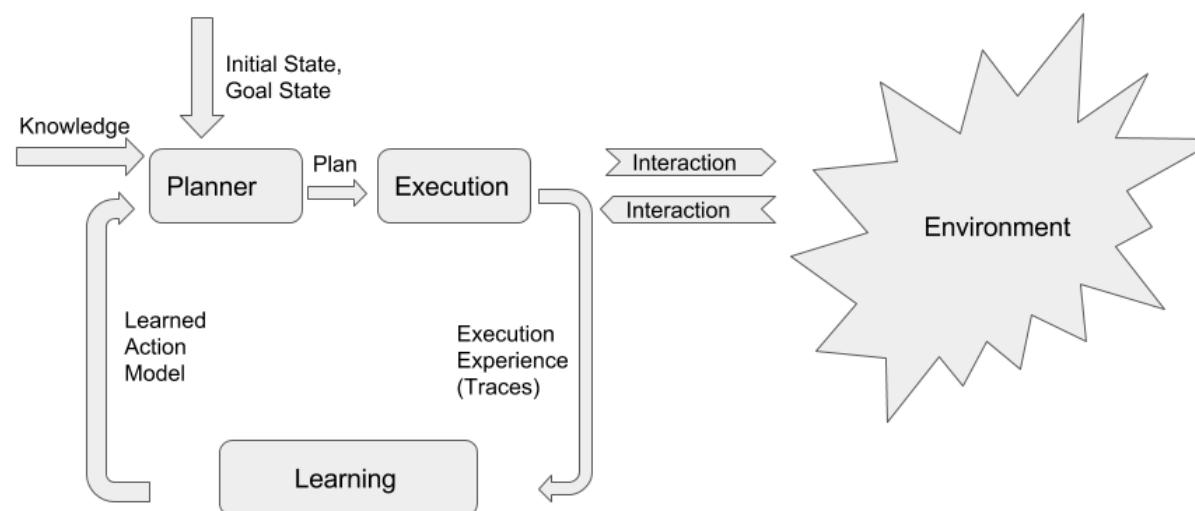
---

- 1. Manipulate representations**
  
- 2. To solve a problem**

# Prodigy

---

- Steve Minton



From [Ankuj Arora et al. (2019). "A review of learning planning action models"

## E.g. The PRODIGY system

---

ACM SIGART Bulletin, 1991, vol. 2, no 4, p. 51-55

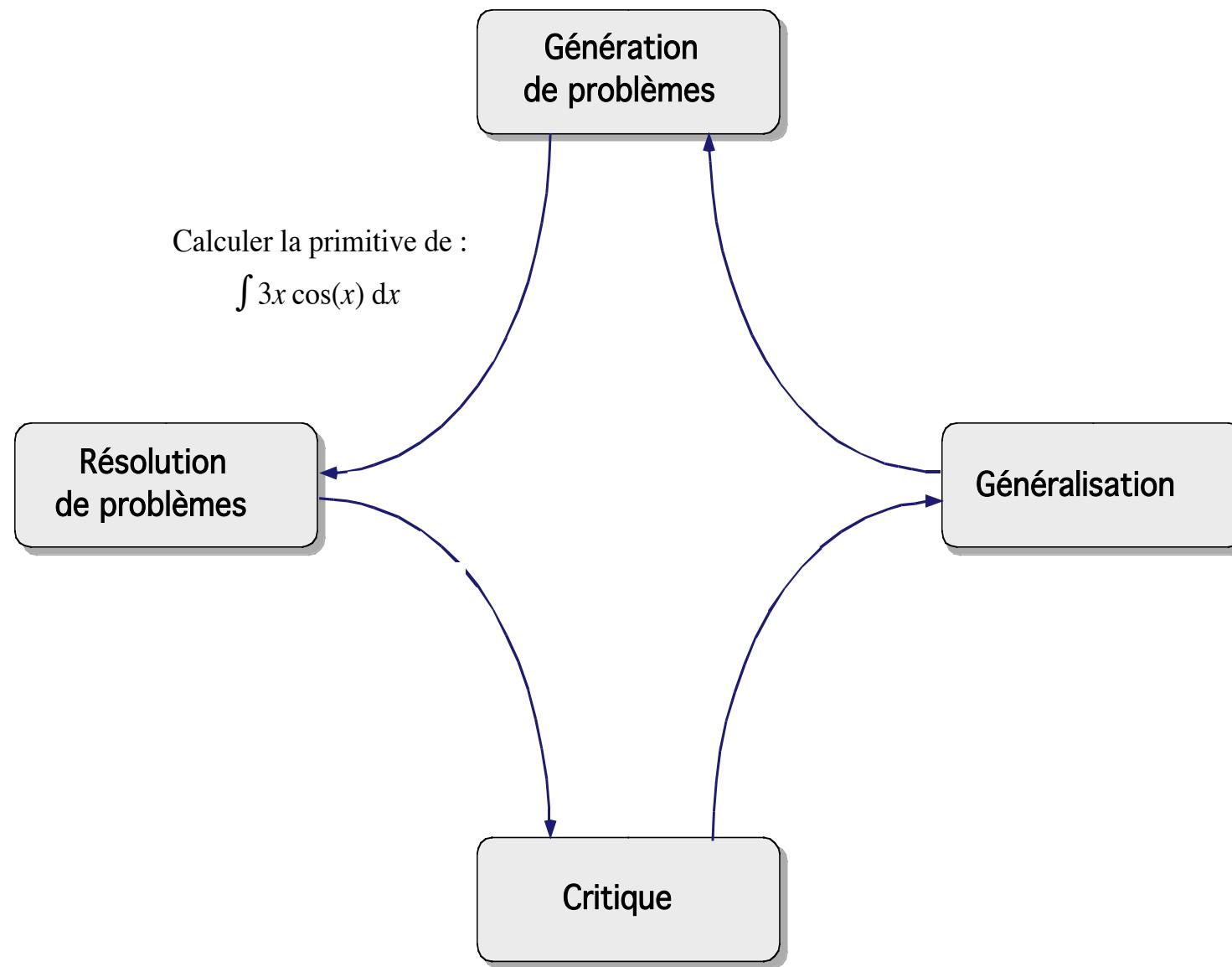
### PRODIGY: An Integrated Architecture for Planning and Learning

Jaime Carbonell, Oren Etzioni\*, Yolanda Gil, Robert Joseph  
Craig Knoblock, Steve Minton†, and Manuela Veloso

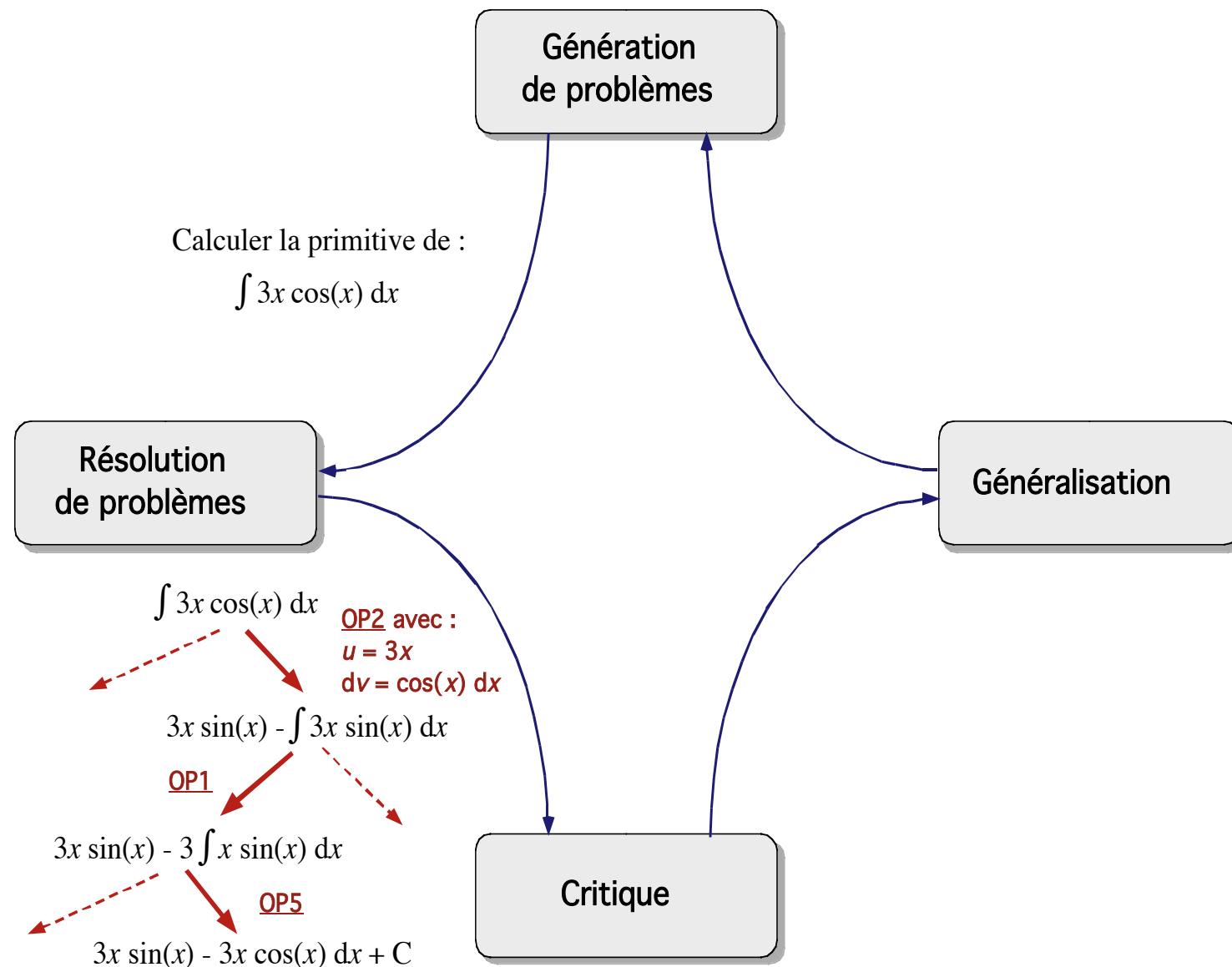
PRODIGY's basic reasoning engine is a general-purpose problem solver and planner [10] that searches for sequences of operators (i.e., plans) to accomplish a set of goals from a specified initial state description. Search in PRODIGY is guided by a set of control rules that apply at each decision point.

PRODIGY's reliance on explicit control rules, which can be learned for specific domains, distinguishes it from most domain independent problem solvers. Instead of using a least-commitment search strategy, for example, PRODIGY expects that any important decisions will be guided by the presence of appropriate control knowledge. If no control rules are relevant to a decision, then PRODIGY makes a quick, arbitrary choice. If in fact the wrong choice is made, and costly backtracking proves necessary, an attempt will be made to learn the control knowledge that must be missing.

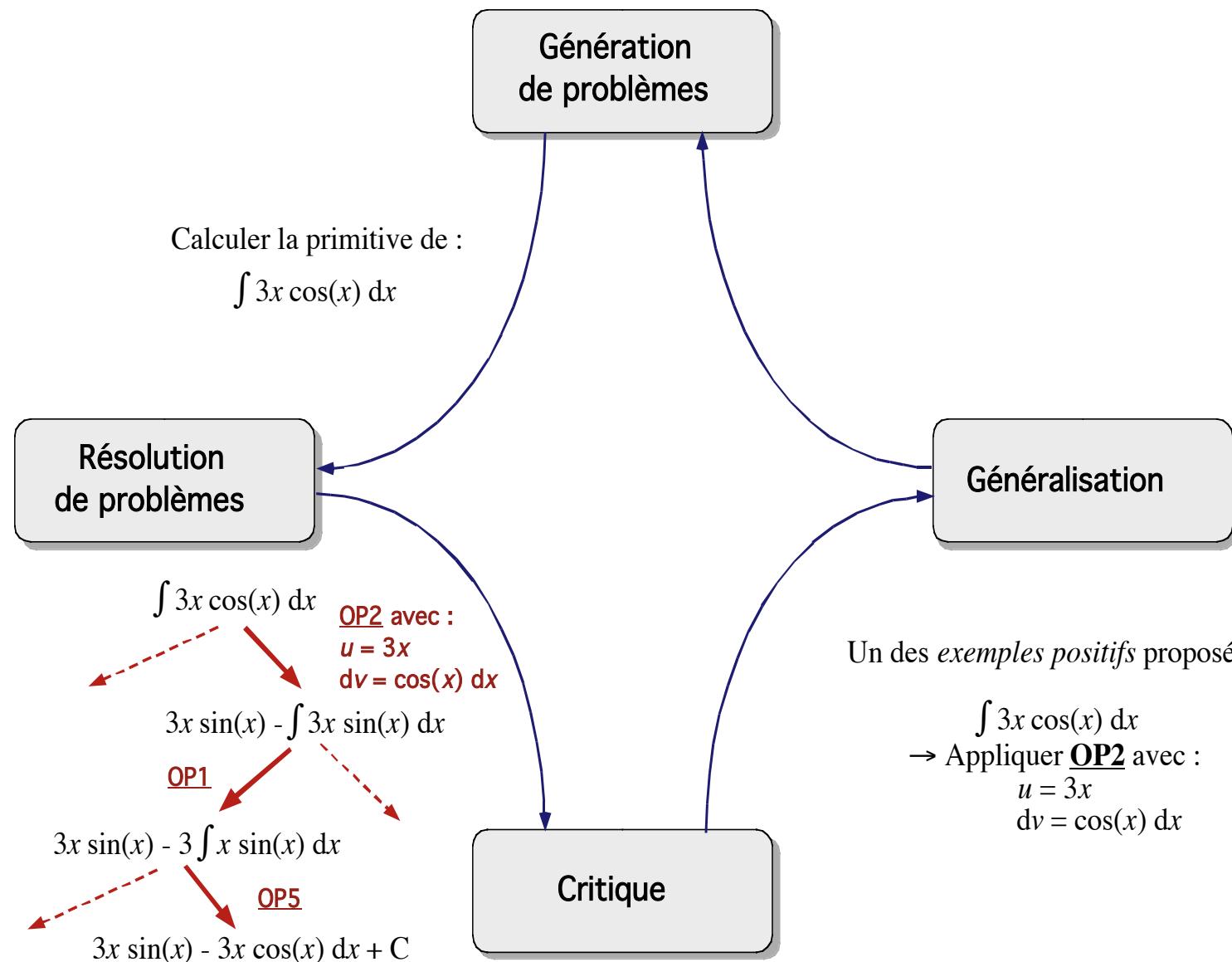
## Illustration: LEX (Tom Mitchell)



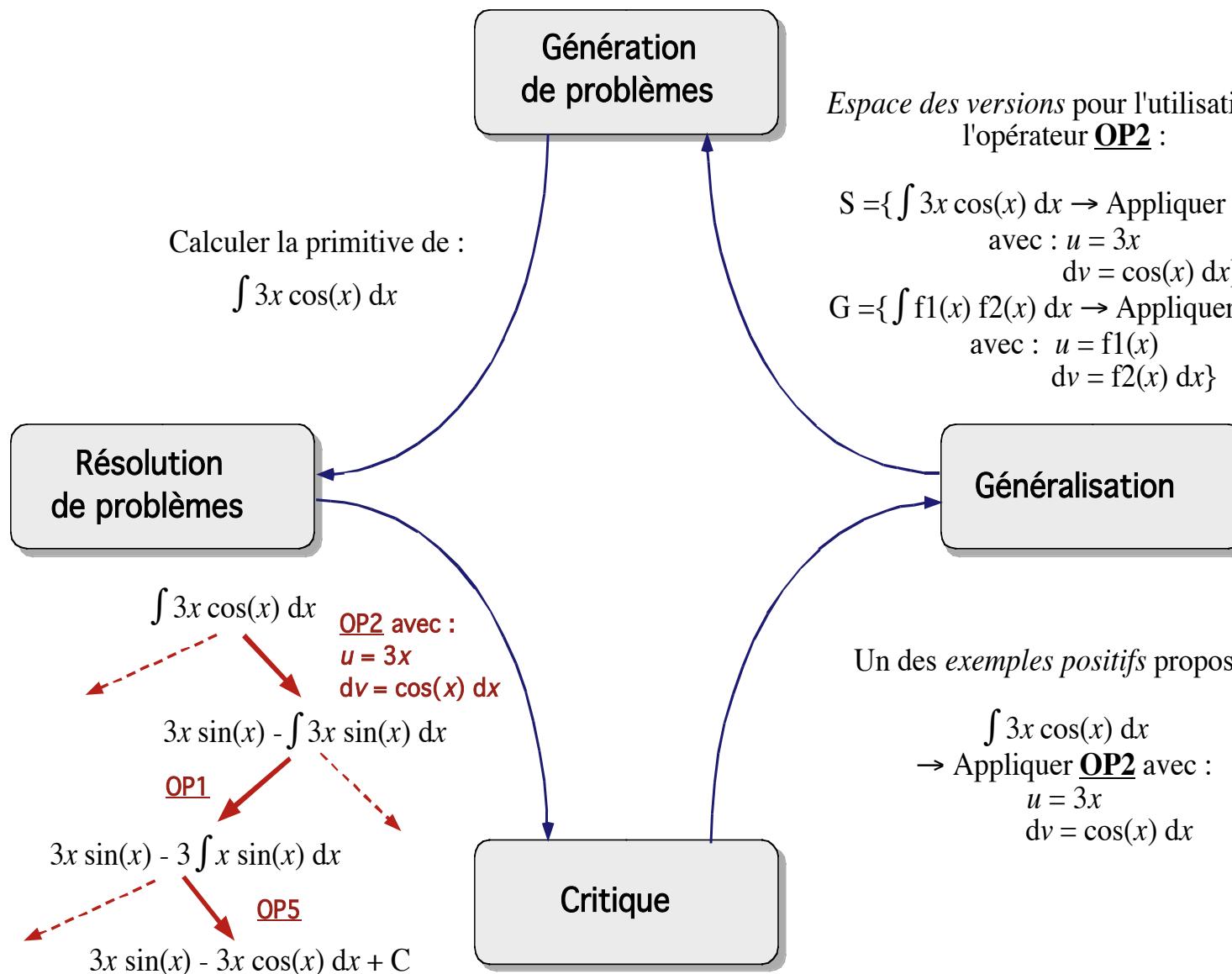
## Illustration: LEX (Tom Mitchell)



## Illustration: LEX (Tom Mitchell)



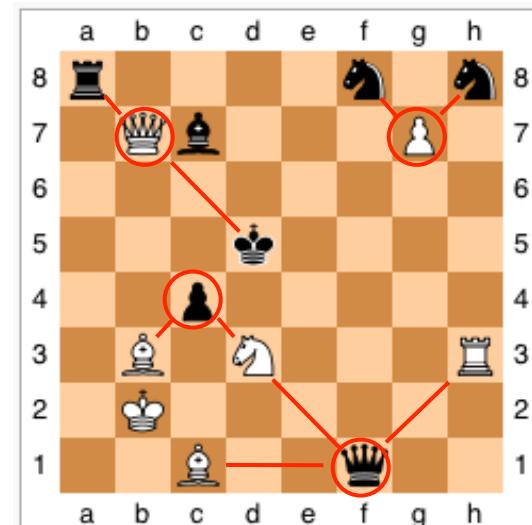
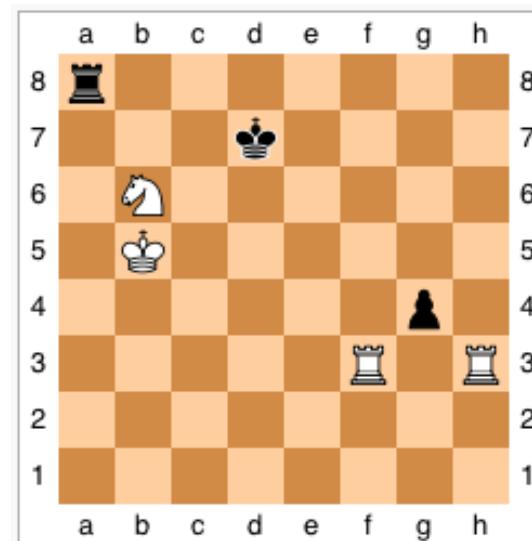
# Illustration: LEX (Tom Mitchell)



# Learning from a single example

## Explanation-Based Learning

1. A single example
2. Search for a proof of a « fork »
3. Generalization



# Explanation-Based Learning

---

Ex : **learn the concept** `stackable(Object1, Object2)`

- **Domain theory :**

```
(T1) : weight(X, W) :- volume(X, V), density(X, D), W is V*D.  
(T2) : weight(X, 50) :- is_a(X, table).  
(T3) : lighter_than(X, Y) :- weight(X, W1), weight(Y, W2), W1 < W2.
```

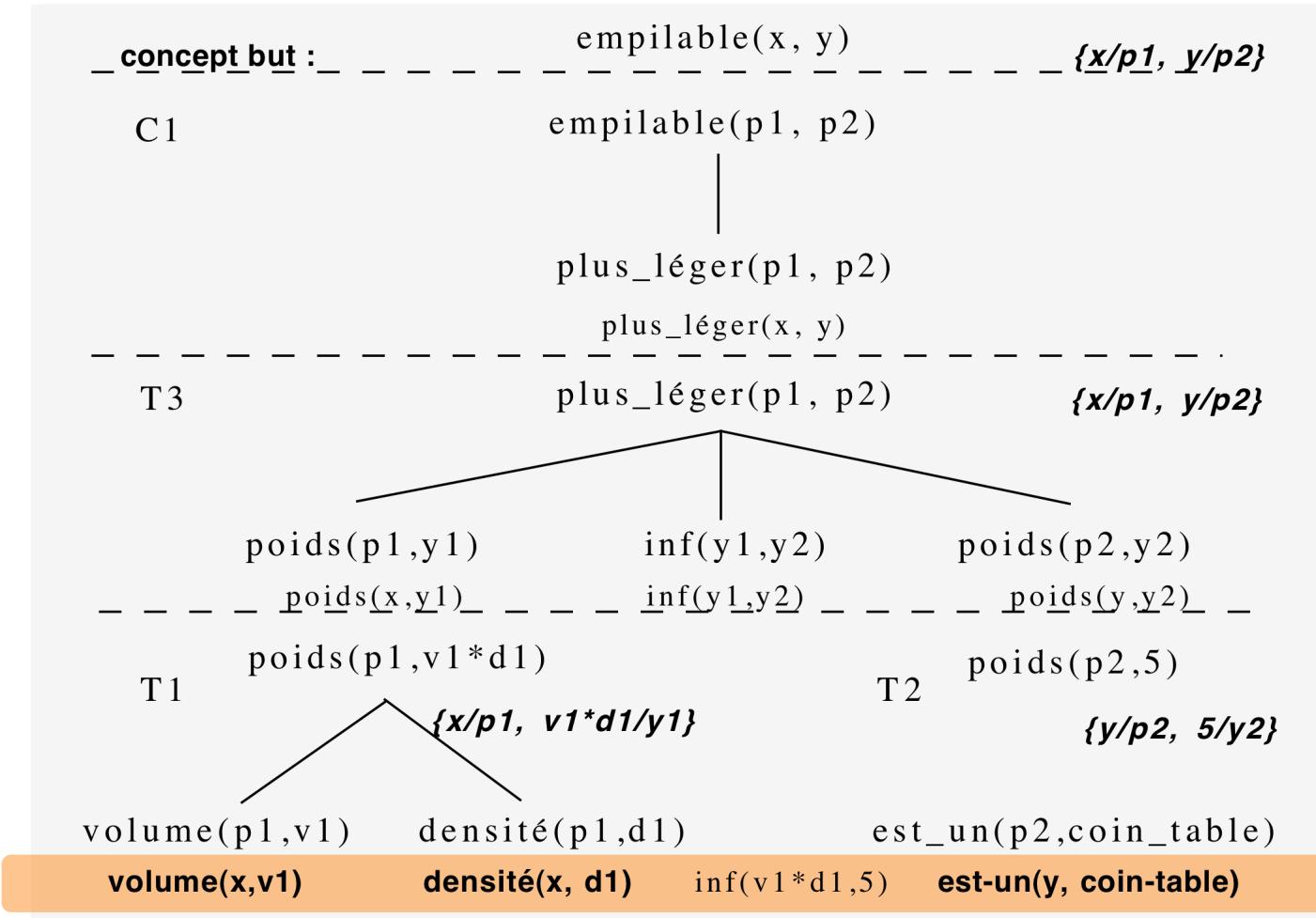
- **Operationality constraint:**

- Concept should be expressible using *volume, density, color, ...*

- **Positive example (solution) :**

|                                      |   |
|--------------------------------------|---|
| <code>on(obj1, obj2).</code>         | <code>volume(object1, 1).</code>          |
| <code>is_a(object1, box).</code>     | <code>volume(object2, 0.1).</code>        |
| <code>is_a(object2, table).</code>   | <code>owner(object1, frederic).</code>    |
| <code>color(object1, red).</code>    | <code>density(object1, 0.3).</code>       |
| <code>color(object2, blue).</code>   | <code>Made_of(object1, cardboard).</code> |
| <code>made_of(object2, wood).</code> | <code>owner(object2, marc).</code>        |

# Explanation-Based Learning



Generalized search tree resulting from regression of the target concept in the proof tree  
by computing at each step the most general literals allowing this step.

# Explanation-Based Learning

---

- Induction **from a single example**
  - ... and a **strong domain theory**
- Language of **logics**
- **Operators** for reasoning (deduction, ...)

*Now used in « solvers » of SAT problems  
And accelerate them immensely  
because the “data” is clean*

## Lesson (1-1)

---

- **Reasoning can be used** in order to learn
  - To recognize concept
    - By **abstracting** the conditions of membership
  - To accelerate problem-solving
    - By **compiling** search procedures

## Lesson (1-2)

---

- **Reasoning can be used in order to learn**

- Which **does not mean** that the results
  - Prediction
  - Learned hypothesis

Should be automatically readily interpretable

More on this later

# Outline

---

1. What is reasoning (1)
2. Machine Learning nowadays: a path to fast thinking
3. The future of Machine Learning: what is reasoning (2)
4. Conclusion

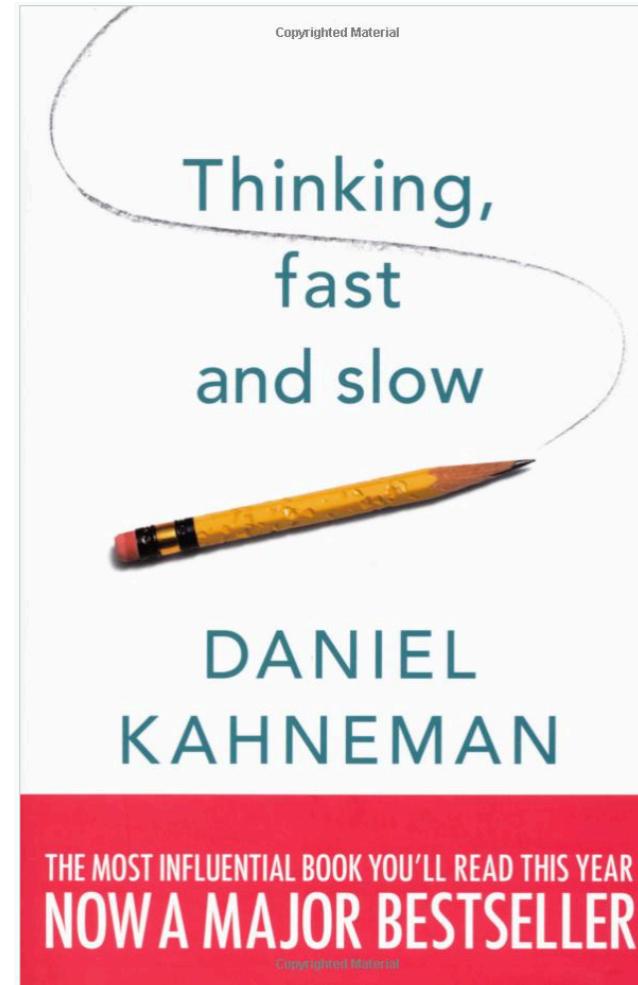
Machine Learning nowadays

A path to fast thinking

## Thinking: fast and slow

---

- Daniel Kahneman



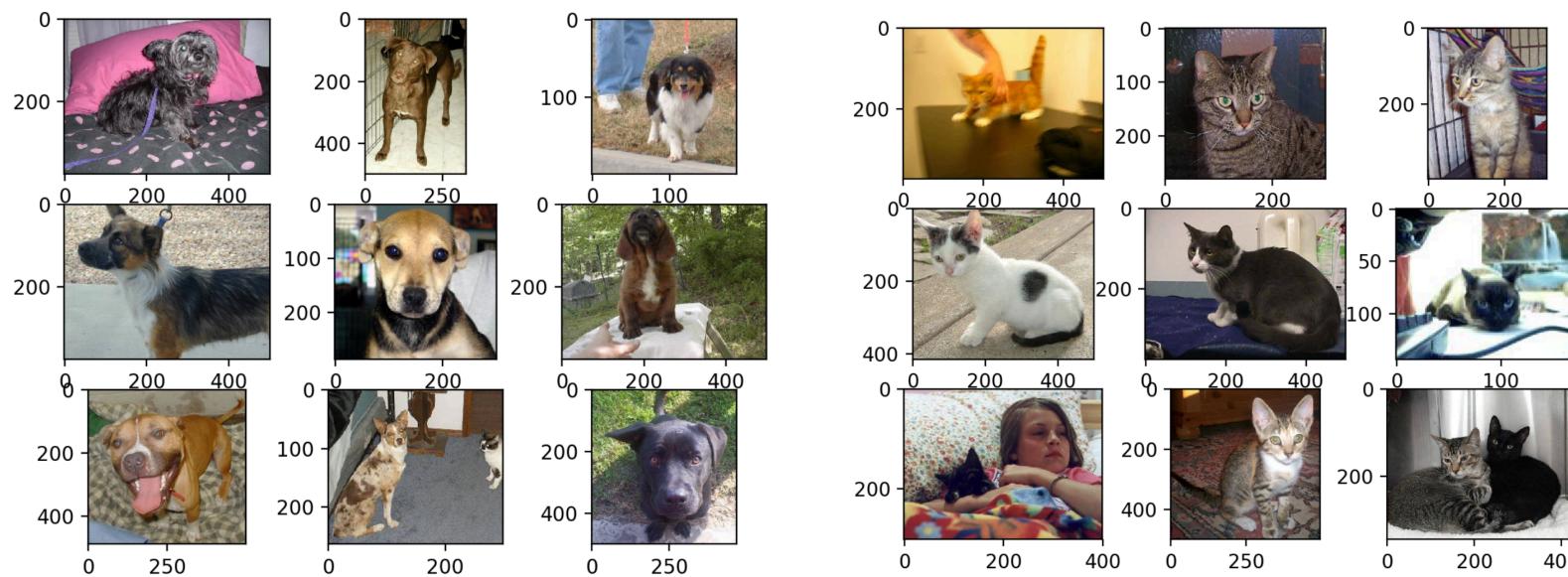
# Thinking: fast and slow

- Type 1 process: **Fast**
  - Fast
  - Effortless
  - Parallel
  - Unconscious
  - Automatic
  - Associative
  - Contextualized
  - Heuristic
  - Intuitive
  - Implicit
  - Nonverbal
  - Independent of general intelligence
  - Independent of working memory
  - Shared with non human animals
- Type 2 process: **Slow**
  - Slow
  - Effortful
  - Serial
  - Conscious
  - Controlled
  - Rule-Based
  - Decontextualized
  - Analytic
  - Reflective
  - Explicit
  - Linked to language
  - Linked to general intelligence
  - Involving working memory
  - Specifically human

# Machine Learning as ...

... Learning a **function** from an **input** space  $X$  to an **output** space  $Y$

Cats vs. dogs



## Supervised learning

---

Given a **training set**

$$\mathcal{S}_m = \{(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \dots, (\mathbf{x}_i, \underset{\textcolor{blue}{h}}{\textcolor{orange}{y_i}}), \dots, (\mathbf{x}_m, y_m)\}$$

- **Find** an hypothesis  $h \in \mathcal{H}$  such that  $h(\mathbf{x}_i) \approx y_i$
- Hoping that it **generalizes well** :

$$\forall \mathbf{x} \in \mathcal{X} : h(\mathbf{x}) \approx y$$

## One example that tells a lot ...

---

- Examples described using:  
*Number* (1 or 2); *size* (small or large); *shape* (circle or square); *color* (red or green)
- They belong either to class '+' or to class '-'

## One example that tells a lot ...

---

- Examples described using:  
**Number** (1 or 2); **size** (small or large); **shape** (circle or square); **color** (red or green)
- They belong either to class ‘+’ or to class ‘-’

| Description          | Your prediction | True class |
|----------------------|-----------------|------------|
| 1 large red square   |                 | -          |
| 1 large green square |                 | +          |
| 2 small red squares  |                 | +          |
| 2 large red circles  |                 | -          |
| 1 large green circle |                 | +          |
| 1 small red circle   |                 | +          |

## One example that tells a lot ...

- Examples described using:

**Number** (1 or 2); **size** (small or large); **shape** (circle or square); **color** (red or green)

| Description          | Your prediction | True class |
|----------------------|-----------------|------------|
| 1 large red square   |                 | -          |
| 1 large green square |                 | +          |
| 2 small red squares  |                 | +          |
| 2 large red circles  |                 | -          |
| 1 large green circle |                 | +          |
| 1 small red circle   |                 | +          |

How many possible functions altogether from  $X$  to  $Y$ ?

$$2^{2^4} = 2^{16} = 65,536$$

How many functions do remain after 6 training examples?

$$2^{10} = 1024$$

# One example that tells a lot ...

- Examples described using:

**Number** (1 or 2); **size** (small or large); **shape** (circle or square); **color** (red or green)

15

| Description           | Your prediction | True class |
|-----------------------|-----------------|------------|
| 1 large red square    |                 | -          |
| 1 large green square  |                 | +          |
| 2 small red squares   |                 | +          |
| 2 large red circles   |                 | -          |
| 1 large green circle  |                 | +          |
| 1 small red circle    |                 | +          |
| 1 small green square  |                 | -          |
| 1 small red square    |                 | +          |
| 2 large green squares |                 | +          |
| 2 small green squares |                 | +          |
| 2 small red circles   |                 | +          |
| 1 small green circle  |                 | -          |
| 2 large green circles |                 | -          |
| 2 small green circles |                 | +          |
| 1 large red circle    |                 | -          |
| 2 large red squares   | ?               |            |

How many  
remaining  
functions?



- 
- How to chose an hypothesis?

# A statistical theory of induction

---

What **performance** do we aim at?

- Cost of a prediction error

- The **loss function**

$$\ell(h(\mathbf{x}), y)$$

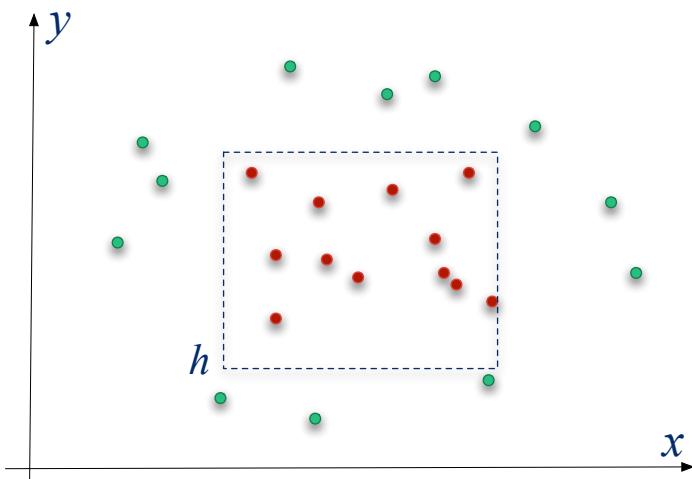
- What is the expected cost if I choose  $h$ ?

- Expected cost: the “**true risk**”

$$R(h) = \int_{\mathcal{X} \times \mathcal{Y}} \ell(h(\mathbf{x}), y) \mathbf{p}_{\mathcal{X}\mathcal{Y}}(\mathbf{x}, y) d\mathbf{x} dy$$

# A statistical theory of induction

- The empirical performance of  $h$ 
  - E.g. No prediction error on the training sample  $S$



*The “empirical risk”*

$$\hat{R}(h) = \frac{1}{m} \sum_{i=1}^m \ell(h(\mathbf{x}_i), y_i)$$

## Statistical study for $|\mathcal{H}|$ hypotheses

---

It leads to:

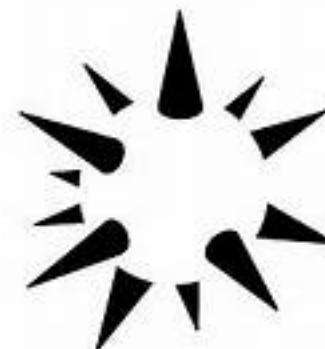
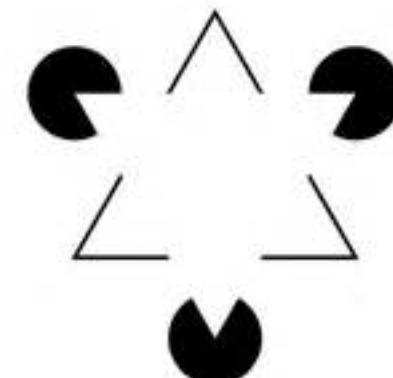
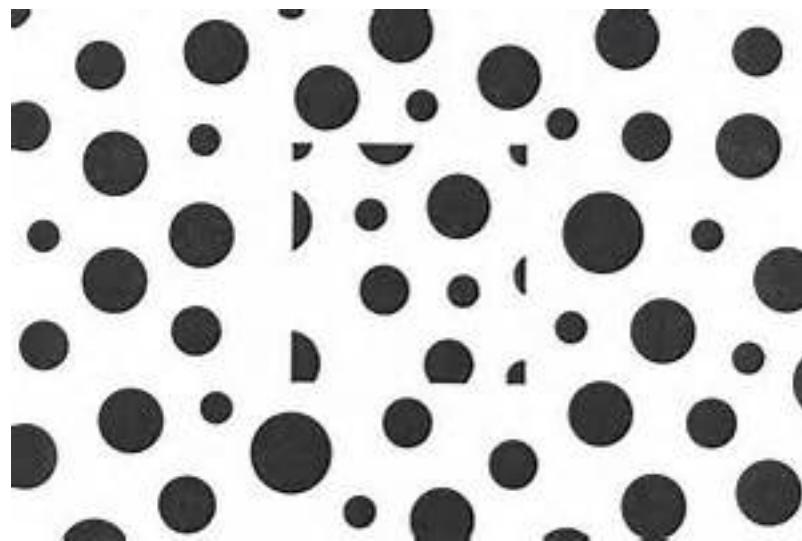
$$\forall h \in \mathcal{H}, \forall \delta \leq 1 : P^m \left[ R(h) \leq \widehat{R}(h) + \underbrace{\frac{\log |\mathcal{H}| + \log \frac{1}{\delta}}{m}}_{\varepsilon} \right] > 1 - \delta$$

The Empirical Risk Minimization principle

is sound only if there exists a limit (a bias) on the expressivity of  $\mathcal{H}$

## Learning – completing data with necessary a priors

---



## HOW TO ... devise learning algorithms

---

1. Define an appropriate **regularized** (inductive) **criterion**
  1. Translate the cost of errors of prediction in the domain into a **loss function**
  2. Define a **regularization term** that expresses  
*assumptions about the underlying regularities of the world*
  3. If possible, make the resulting **optimization** problem a **convex** one

$$h_{opt} = \underset{h \in \mathcal{H}}{\text{ArgMin}} \left[ \underbrace{\frac{1}{m} \sum_{i=1}^m l(h(\mathbf{x}_i), y_i)}_{\text{empirical risk}} + \lambda \underbrace{\text{reg}(\mathcal{H})}_{\text{bias on the world}} \right]$$

2. Use or develop an **efficient optimization solver**

## Learning sparse linear approximator

- The **hypothesis** is of the form  $h(\mathbf{x}) = \mathbf{w} \cdot \mathbf{x}$
- A priori assumption: few non zero coefficients

Ridge regression

$$\mathbf{w}_{\text{ridge}}^* = \underset{\mathbf{w}}{\operatorname{Argmin}} \left\{ \sum_{i=1}^m (y_i - \mathbf{w} \cdot \mathbf{x}_i)^2 + \lambda \|\mathbf{w}\|_2^2 \right\}$$

Lasso regression

$$\mathbf{w}_{\text{lasso}}^* = \underset{\mathbf{w}}{\operatorname{Argmin}} \left\{ \sum_{i=1}^m (y_i - \mathbf{w} \cdot \mathbf{x}_i)^2 + \lambda \|\mathbf{w}\|_1 \right\}$$

## Surrogate expression of the regularized empirical risk

### Optimization

## Regularized empirical risk

3.3 du chapitre 3. Ainsi, étant donnés un échantillon source étiqueté  $S = \{(x_i^s, y_i^s)\}_{i=1}^m$  constitué de  $m$  exemples *i.i.d.* selon  $P_S$  et un échantillon cible non étiqueté  $T = \{(x_i^t)\}_{i=1}^m$  composé de  $m$  exemples *i.i.d.* selon  $D_T$ , en posant  $S_u = \{x_i^s\}_{i=1}^m$  l'échantillon  $S$  privé de ses étiquettes, on veut minimiser :

$$\min_w cm R_S(G_{\rho_w}) + a m \text{dis}_{\rho_w}(S_u, T_u) + \text{KL}(\rho_w \| \pi_0), \quad (7.5)$$

où  $\text{dis}_{\rho_w}(S_u, T_u) = \left| \mathbb{E}_{(h,h') \sim \rho_w^2} R_{S_u}(h, h') - \mathbb{E}_{(h,h') \sim \rho_w^2} R_{T_u}(h, h') \right|$  est le désaccord empirique entre  $S_u$  et  $T_u$  spécialisé à une distribution  $\rho_w$  sur l'espace  $\mathcal{H}$  des classificateurs linéaires considéré. Les réels  $a > 0$  et  $c > 0$  sont des hyperparamètres de l'algorithme. Notons que les constantes  $A$  et  $C$  du théorème 7.7 peuvent être retrouvées à partir de n'importe quelle valeur de  $a$  et  $c$ . Étant donnée la fonction  $\ell_{\text{dis}}(x) = 2 \ell_{\text{Erf}}(x) \ell_{\text{Erf}}(-x)$  (illustrée sur la figure 7.1), pour toute distribution  $D$  sur  $X$ , on a :

$$\begin{aligned} \mathbb{E}_{(h,h') \sim \rho_w^2} R_D(h, h') &= \mathbb{E}_{x \sim D} \mathbb{E}_{(h,h') \sim \rho_w^2} \mathbf{I}[h(x) \neq h'(x)] \\ &= 2 \mathbb{E}_{x \sim D} \mathbb{E}_{(h,h') \sim \rho_w^2} \mathbf{I}[h(x) = 1] \mathbf{I}[h'(x) = -1] \\ &= 2 \mathbb{E}_{x \sim D} \mathbb{E}_{h \sim \rho_w} \mathbf{I}[h(x) = 1] \mathbb{E}_{h' \sim \rho_w} \mathbf{I}[h'(x) = -1] \\ &= 2 \mathbb{E}_{x \sim D} \ell_{\text{Erf}}\left(\frac{\langle w, x \rangle}{\|x\|}\right) \ell_{\text{Erf}}\left(-\frac{\langle w, x \rangle}{\|x\|}\right) \\ &= \mathbb{E}_{x \sim D} \ell_{\text{dis}}\left(\frac{\langle w, x \rangle}{\|x\|}\right). \end{aligned}$$

Ainsi, trouver la solution optimale de l'équation (7.5) revient à chercher le vecteur  $w$  qui minimise :

$$c \sum_{i=1}^m \ell_{\text{Erf}}\left(y_i^s \frac{\langle w, x_i^s \rangle}{\|x_i^s\|}\right) + a \left| \sum_{i=1}^m \left[ \ell_{\text{dis}}\left(\frac{\langle w, x_i^s \rangle}{\|x_i^s\|}\right) - \ell_{\text{dis}}\left(\frac{\langle w, x_i^t \rangle}{\|x_i^t\|}\right) \right] \right| + \frac{\|w\|^2}{2}. \quad (7.6)$$

L'équation précédente est fortement non convexe. Afin de rendre sa résolution plus facilement contrôlable, nous remplaçons la fonction  $\ell_{\text{Erf}}(\cdot)$  par sa relaxation convexe  $\ell_{\text{Erf}_{\text{cvx}}}(\cdot)$  (comme pour PBGD3 et illustrée sur la figure 7.1). L'optimisation se réalise ensuite par une descente de gradient. Le gradient de l'équation 7.6 étant :

## A lot of “Lamppost theorems”

---

Theorems that guarantee that:

- **If** the world obeys **my a priori assumptions**
- **Then** the learning algorithm will end up with a good hypothesis (closed to the “real” one)
- **Otherwise** learning can lead to very bad hypotheses  
(e.g. *If the world is not sparse*)



magritte  
The Collection

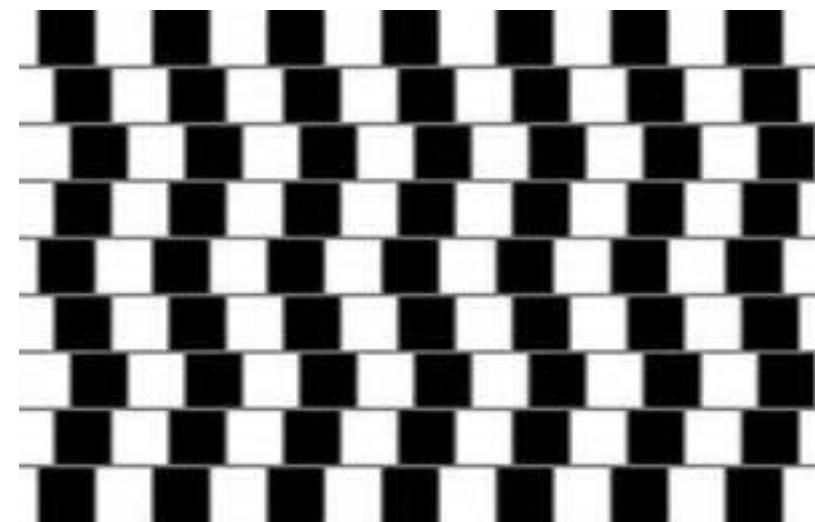
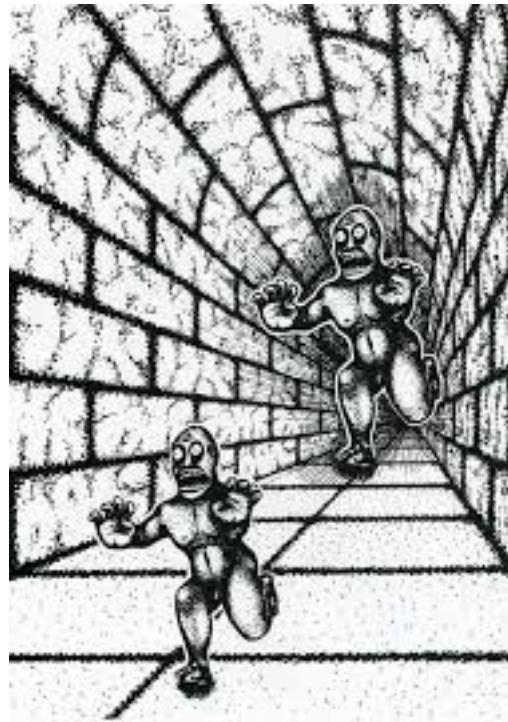
## Interpreting – completion of percepts

---



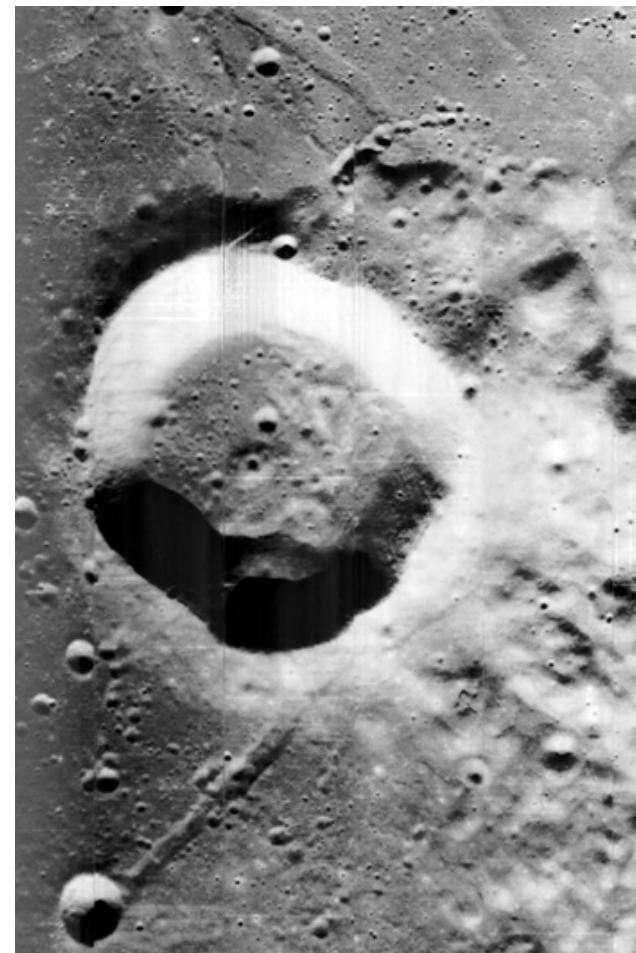
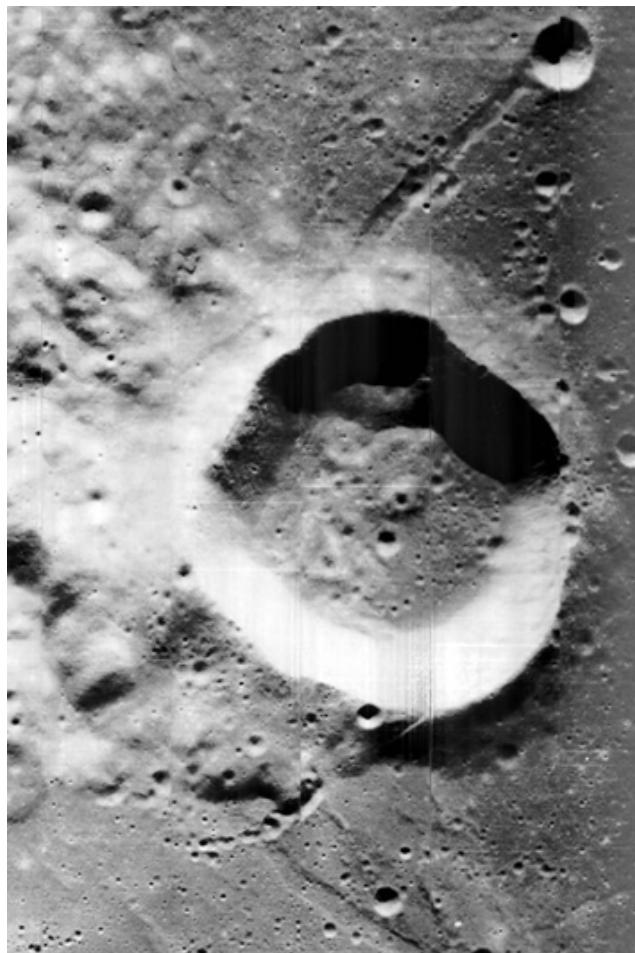
## Induction and its illusions

---



## Induction and illusions

---

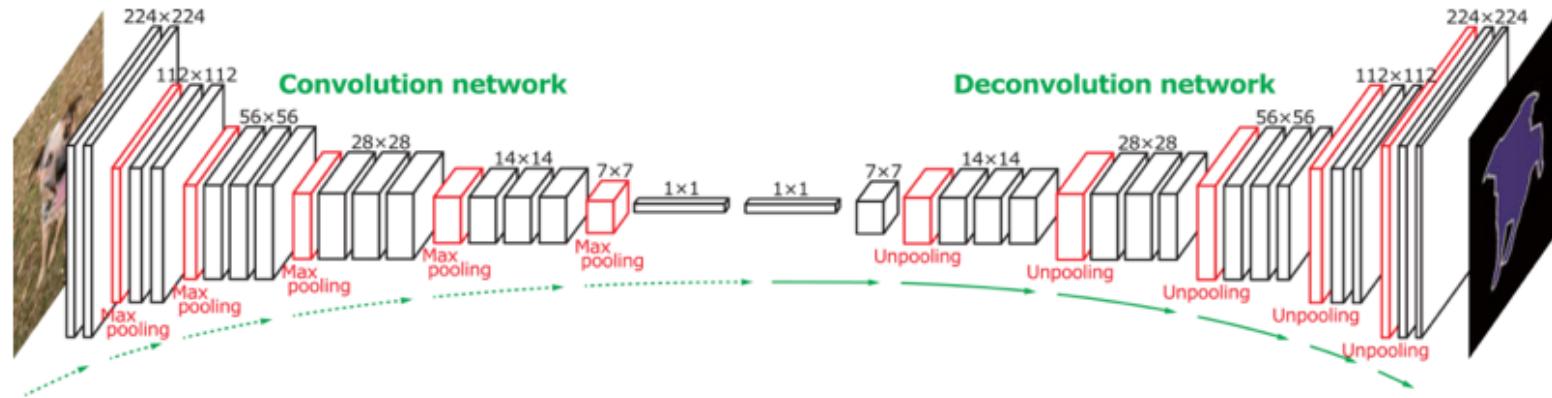


Crater or hill?

## Lesson (2-1)

---

- In the current ML methods, **prior knowledge** takes the form of imposed **bias on the type of regularities** that can be captured
- **No reasoning** takes place during learning
  - Just an **exploration** of the hypothesis space
  - **Subject to an optimization** criterion



- Of course there are biases in the current machine learning algorithms and techniques
- But it seems that in the new paradigm there is no place nor need for reasoning

- 
- Maybe we don't yet understand the biases that are entailed in deep Neural Networks

See C. Zhang, S. Bengio, M. Hardt, B. Recht, O. Vinyals (ICLR, May 2017).  
“Understanding deep learning requires rethinking generalization”

But what does it have to do with reasoning?

# A car in a swimming pool

---

... or no car ... ?



Is this less of a car  
because the context is wrong?

[Léon Bottou (ICML-2015, invited talk) « *Two big challenges in Machine Learning* »]

# Adversarial learning

**Central question:** which guarantees ?



Boxer: 0.40 Tiger Cat: 0.18

(a) Original image

Airliner: 0.9999

(b) Adversarial image

!!??

[Selvaraju et al. (2017) « Grad-CAM: Visual explanations from deep networks via gradient-based localization »]

# The case of AlphaGo

- Plays like an “alien”
- Stunning moves
- A revolution for go playing
- Effervescence in the go playing circles



AlphaGo And The Hand Of God

A Layperson's Guide to  
The Google Deepmind AlphaGo Challenge Match  
Brady Daniels, March 2016

1. Intro to Go and Computer Go
2. Amazing Moves and Adjustments
3. Significance of AlphaGo and Deep Learning
4. Impact on the Go World

Lee Sedol [9d] vs. AlphaGo  
Move 65 (B n15): White to play

CGoban: Lee Sedol [9]

## The case of AlphaGo: to understand its plays

---

Fan Hui, Gu Li, Zhou Ruyang (very strong players at go) are converting themselves in the analysis of the games played by AlphaGo

- Kind of exegesis. Explanations a posteriori
- Necessary for
  - Communication
  - Teaching



And even AlphaGo can make mistakes

## Lesson (2-2)

---

- For various reasons, sometimes, there are needs for:
  - Justification of the **result**
  - Transparency of the learned **hypothesis**

# Outline

---

1. What is reasoning (1)
2. Machine Learning nowadays: a path to fast thinking
3. The future of Machine Learning: what is reasoning (2)
4. Conclusion

The future of Machine Learning

What is reasoning (2)

## Reasoning ...

---

... comes **after** the fact

- Most of our inferences result from specialized modules with little access to their functioning
  - Perception
  - Memory access (and reconstruction)
  - Interpreting other's states of mind

We use reasons to justify ourselves and convince others

## Finding reasons

---

- Try to identify **bits of information** that **did play a role** in our beliefs and decisions

## The “bystander effect”

---

- People were told that they would participate in a market study on games. **Participants** were individually welcomed at the door of the lab by a **friendly assistant** who took them to a room connected to her office, gave them a questionnaire to fill out, and went back to her office, where she could be heard shuffling papers, opening drawers, and so on.  
**A while later**, the participants heard her climb on a chair and then heard a loud crash and a scream, *“Oh, my God, my foot ... I ... can’t move it. Oh ... my ankle! ... I can’t get this ... thing ... of me.”*

## The “bystander effect”

---

- In **one condition**, the participant was **alone in the room** when all this happened.
- In **another condition**, **there was a man in the room** who acted as if he were a participant too.  
This man **hardly reacted** to the crash and the scream.  
He just shrugged and went on filling out the questionnaire.

## The “bystander effect”

---

What do you expect ?

## The “bystander effect”

---

What do you expect ?

- In the first condition, 70% of the participants intervened to help.
- In the second condition, only 7% did

## The “bystander effect”

---

After all this the participants were **interviewed about their reactions**.

- First condition, typically: “I wasn’t quite sure what had happened; I thought I should at least find out.”
- Second condition, typically the participants thought that whatever had happened was not too serious

Not one ever mentioned the presence of the other participant!!

And even more, they **claimed this presence had no influence at all on their decision**

## Lesson

---

- The reasons invoked **come after the fact** of deciding
- Indeed they are blatantly faulty
- Reasons are **justifications**

- 
- If I base my “intuitions” on good reasons
    - Other actors will **increase their trust** in my claims
  - Reasons
    1. Premises or important influencing factors for the final decision
    2. The reasoning that uses them. Others should be able to duplicate it
    - Both together should lead to the same final decision deemed to be explained

## A new perspective

---

- Reasoning **can be used** to solve problems and then be the basis of learning in order to gain efficiency: compilation of knowledge
- But it **is not required** to learn
- Reasoning is **required when**
  - **sub-systems** are **interacting** in a repeated manner
  - Or there are **interactions with humans**

## An accident of an (autonomous) car

---

*On May 7th, 2016, a Tesla car in autopilot mode collided with a semi-trailer across the road*

- The analysis reveals that:
  - The **radar** did detect the semi-trailer but there were numerous road signs on the road with a radar "signature" similar to one of a car
  - The **camera** was unsure of its detections due to a dazzling milky sky (the semi-trailer was white).

## Combination of adaptive sub-systems and justifications

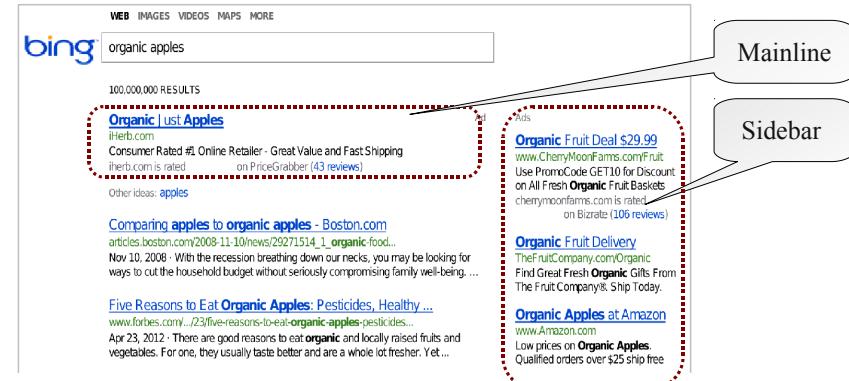
---

- The analysis reveals that:
  - The **radar** did detect a semi-trailer but ...
  - The **camera** was not sure of its detection of a semi-trailer because ...
- **What if** the radar had said: “Even though I am **not quite sure**, I do not believe it to be a semi-trailer **because** it looks to me like a road sign of which I have detected many on this road.”
- **What if** the camera had said: “Even though I am **not quite sure**, I do not believe it to be a semi-trailer **because** the color is close to the color of the sky which is milky today.”
- **What if** the radar and camera had a long history of such exchanges **and were influencing each other** ...

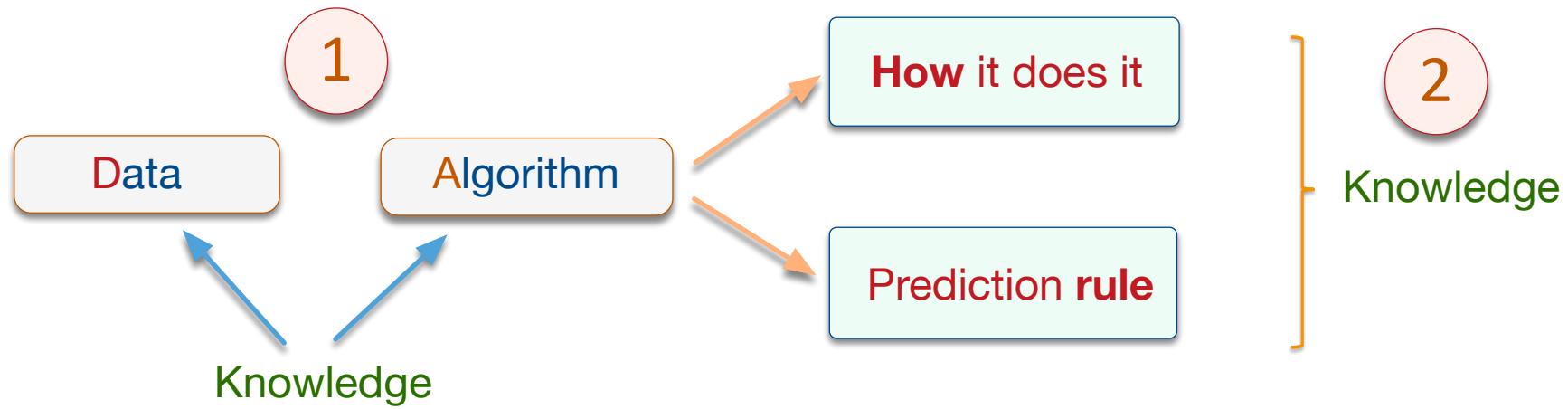
# Interactions between learning modules

## Adaptive advertising placement system

- Two sub-systems
  - One placing **advertising links**
  - The other one choosing the **adds**
- Mutually influencing each other
  - Each one is based on click data
  - Which also **depends on the intervention of the other system**
  - And other **uncontrolled factors** (price, user requests, ...)

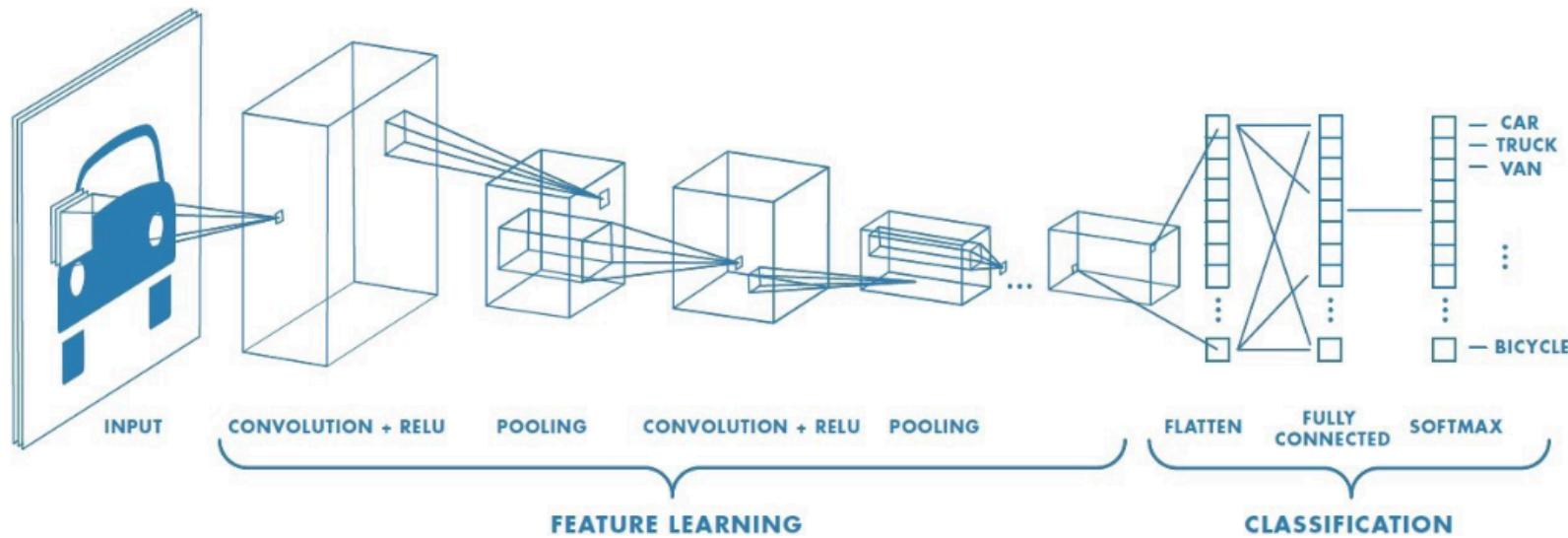


[L. Bottou et al. «*Counterfactual Reasoning and Learning Systems: The Example of Computational Advertising*», JMLR, 14, (2013), 3207-3260]



# Knowledge as input to ML

- Convolutional Neural Networks
  - Knowledge embedded in the architecture of the network

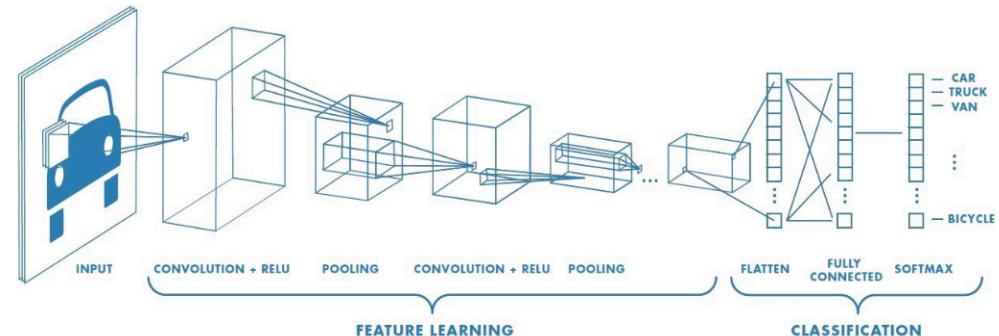


From <https://towardsdatascience.com/a-comprehensive-guide-to-convolutional-neural-networks-the-eli5-way-3bd2b1164a53>

# Knowledge as input to ML

---

- Convolutional Neural Networks
  - Knowledge embedded in the architecture of the network



- How to obtain an explanation
  - About the **procedure?** → Forget it! It is opaque
  - About the **premises?**

## What kind of “reasons” can we extract?

---

- The **hypothesis returned**: a decision function
  - **Recommending a movie**
  - **Recommending a life partner**
  - **Written character recognition**
  - **Recognize traffic signs**
  - Decide the **value of a position** in go
  - Predict the **risk of crime occurrence**
  - Decide if someone should **get a loan**
  - Decide to **hire or not** someone

# The case of deep NNs

---

- Various approaches to “explain” the decision-making of **pre-trained** models
  - 1. **Feature**-based explanations methods
    - Attribute the decision to **important features** in the input space
  - 2. **Sample**-based explanation methods
    - Attribute the decision to **previously observed samples**
  - 3. **Concept**-based explanation methods
    - estimates the **importance of a concept** (intermediate category / intermediate layers) to a given class

# The case of deep NNs

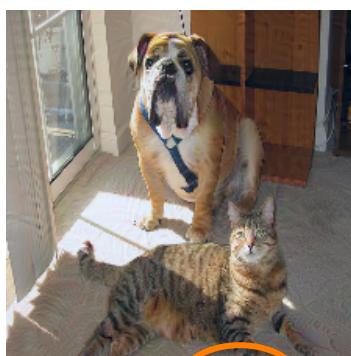
## 1. Feature-based explanations methods

- Attribute the decision to **important features in the input space**



Boxer: 0.40 Tiger Cat: 0.18

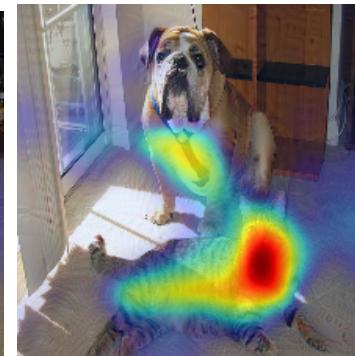
(a) Original image



Airliner: 0.9999  
(b) Adversarial image



Boxer: 1.1e-20  
(c) Grad-CAM “Dog”



Tiger Cat: 6.5e-17  
(d) Grad-CAM “Cat”

!!??

[Selvaraju et al. (2017) « *Grad-CAM: Visual explanations from deep networks via gradient-based localization* »]

# The case of deep NNs

## 2. Sample-based explanation methods

- Attribute the decision to previously observed samples

test id3092  
grizzly bear predicted as  
grizzly bear



train id13033  
grizzly bear predicted as  
grizzly bear



POSITIVE Example

train id12728  
grizzly bear predicted as  
grizzly bear



POSITIVE Example

train id12742  
grizzly bear predicted as  
grizzly bear



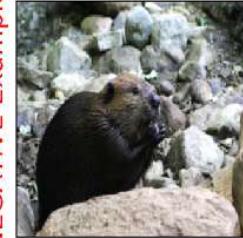
POSITIVE Example

train id21249  
polar bear predicted as  
polar bear



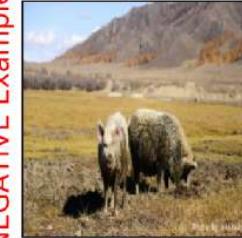
NEGATIVE Example

train id1228  
beaver predicted as  
beaver



NEGATIVE Example

train id20730  
pig predicted as  
pig

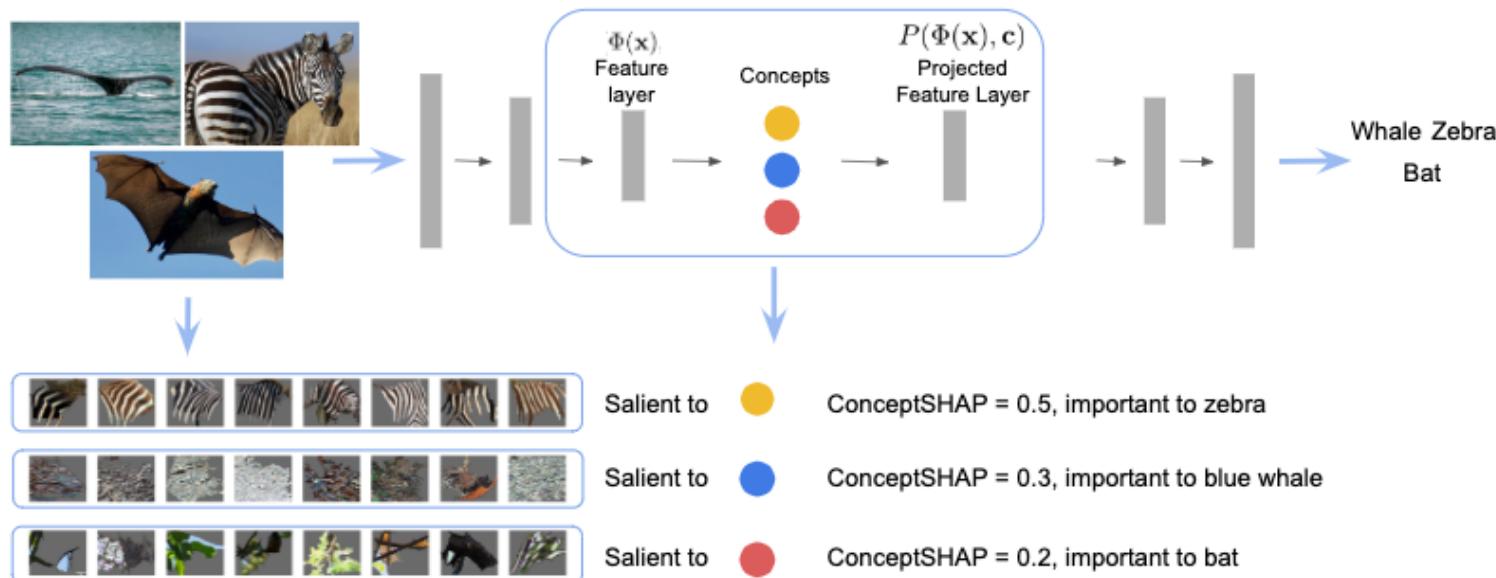


NEGATIVE Example

# The case of deep NNs

## 3. Concept-based explanation methods

- estimates the **importance of a concept** (intermediate category) to a given class

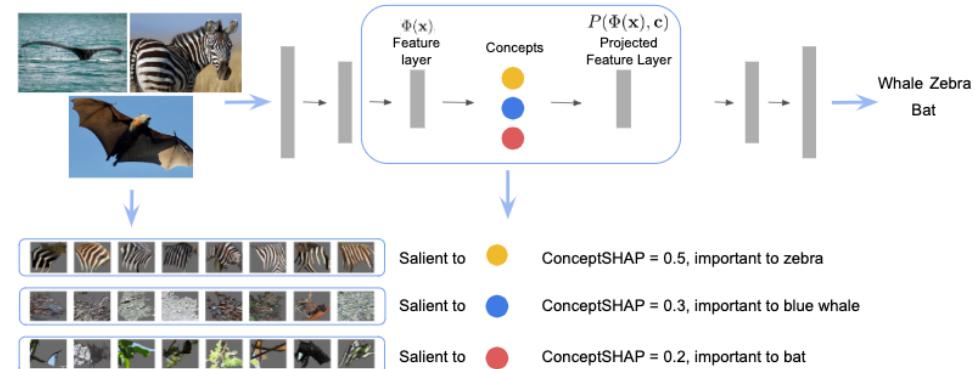


[Chih-Kuan Yeh et al. (2019) « On concept-based explanations in deep neural networks »]

# The case of deep NNs

## 3. Concept-based explanation methods

- estimates the **importance of a concept** (intermediate category) to a given class



- Concepts are taken (by clustering) at an **intermediate layer** of the NN
- Each concept should be **interpretable** and semantically-meaningful

[Chih-Kuan Yeh et al. (2019) « On concept-based explanations in deep neural networks »]

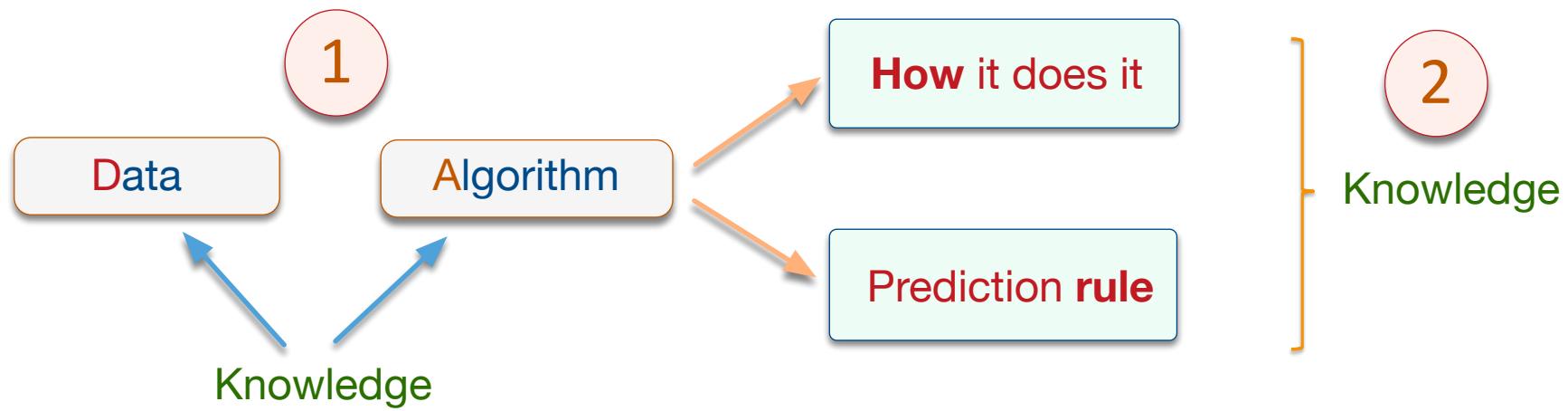
# The case of deep NNs

## 3. Concept-based explanation methods

- estimates the **importance of a concept** (intermediate category) to a given class
- Notion of **completeness** of the concepts
- Concepts are learned **after training**

| Concepts    | Nearest Neighbors |  |  | SHAP  | Related Classes   |
|-------------|-------------------|--|--|-------|---|
| stripe 1    |                   |  |  | 0.126 | horse, lion   |
| ripple 1    |                   |  |  | 0.120 | panda, hippopotamus, ox, wearus, wolf                   |
| stripe 2    |                   |  |  | 0.124 | bobcat, collie, rabbit, zebra                           |
| leaf/branch |                   |  |  | 0.121 | antelope, bat, deer, hamster, mouse, tiger,             |
| grass       |                   |  |  | 0.08  | grizzly-bear, raccoon                                   |
| thick snow  |                   |  |  | 0.109 | dalmatian, leopard, otter, squirrel                     |
| ripple 2    |                   |  |  | 0.105 | blue-whale, dolphin, mole, spider-monkey                |
| dots        |                   |  |  | 0.124 | persian-cat, polar-bear, rhinoceros, siamese-cat, skunk |

[Chih-Kuan Yeh et al. (2019) « On concept-based explanations in deep neural networks »]



## What kind of “reasons” can we extract?

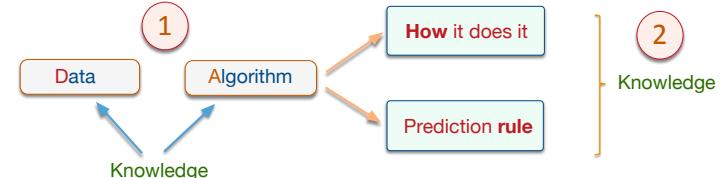
---

- **Interpretability** of the decision function
    - **Decision trees** seem readily interpretable
    - **Linear decision functions** are less so
    - **Random forests** are much less still
    - **SVM**
    - **Neural Networks**
- 
- Require a difficult analysis

# Knowledge as input to ML

- Knowledge in the learning algorithm

- Constraints on the hypothesis space: representation bias



$$h^* = \underset{h \in \mathcal{H}}{\text{ArgMin}} \left[ R_{\text{Emp}}(h) + \lambda \text{reg}(h) \right]$$

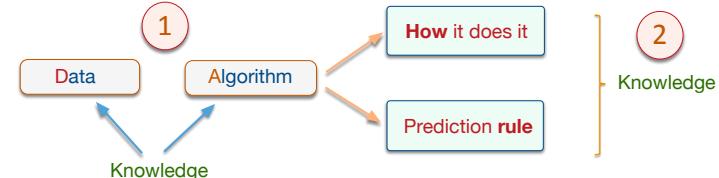
Looking for sparse linear hypotheses

$$h^* = \underset{h \in \mathcal{H}}{\text{ArgMin}} \left[ \frac{1}{m} \sum_{i=1}^m \ell(h(\mathbf{x}_i), y_i) + \lambda \|h\|_1 \right]$$

Favors hypotheses with few non null coefficients

# Knowledge as input to ML

- Knowledge in the learning algorithm



Looking for sparse linear hypotheses

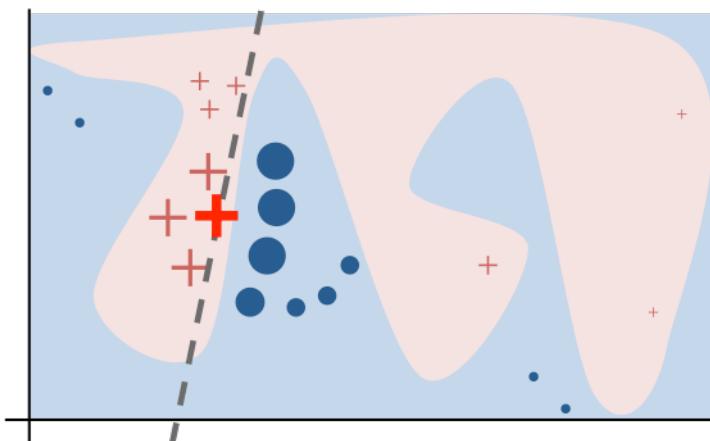
$$h^* = \underset{h \in \mathcal{H}}{\text{ArgMin}} \left[ \frac{1}{m} \sum_{i=1}^m \ell(h(\mathbf{x}_i), y_i) + \lambda \|h\|_1 \right]$$

Favors hypotheses with few non null coefficients

“Hey, I am looking for sparse hypotheses, and if this fits the data, I am satisfied”

# Explaining any classifier?

- [Ribeiro et al. (2016)]: LIME (Local Interpretable Model-agnostic Explanations)
  - An explanation is a **local linear approximation** of the decision function
  - Whereas the model may be very complex globally, it is easier to approximate it around the **vicinity of a particular instance**.
  - While treating the model as a black box, **perturb the instance** we want to explain and learn a sparse linear model around it, as an **explanation**.



“Explaining” the instance +

sample instances around +, and weight them according to their proximity to + (weight here is indicated by size).  
Then learn a linear model (dashed line) that approximates the model well in the vicinity of +

# Explaining any classifier?

- [Ribeiro et al. (2016)]: LIME (Local Interpretable Model-agnostic Explanations)

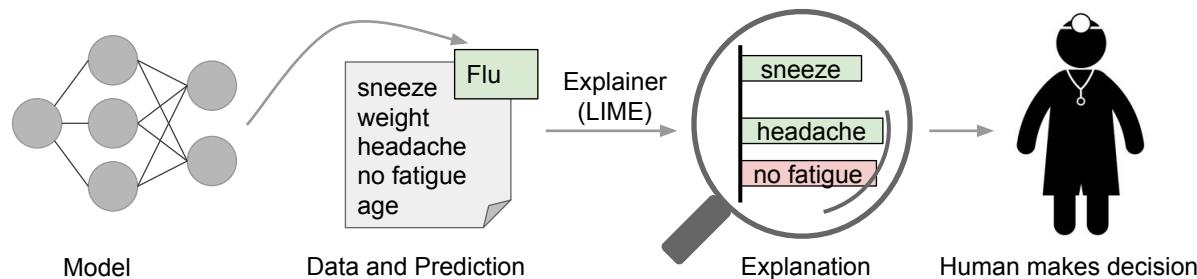
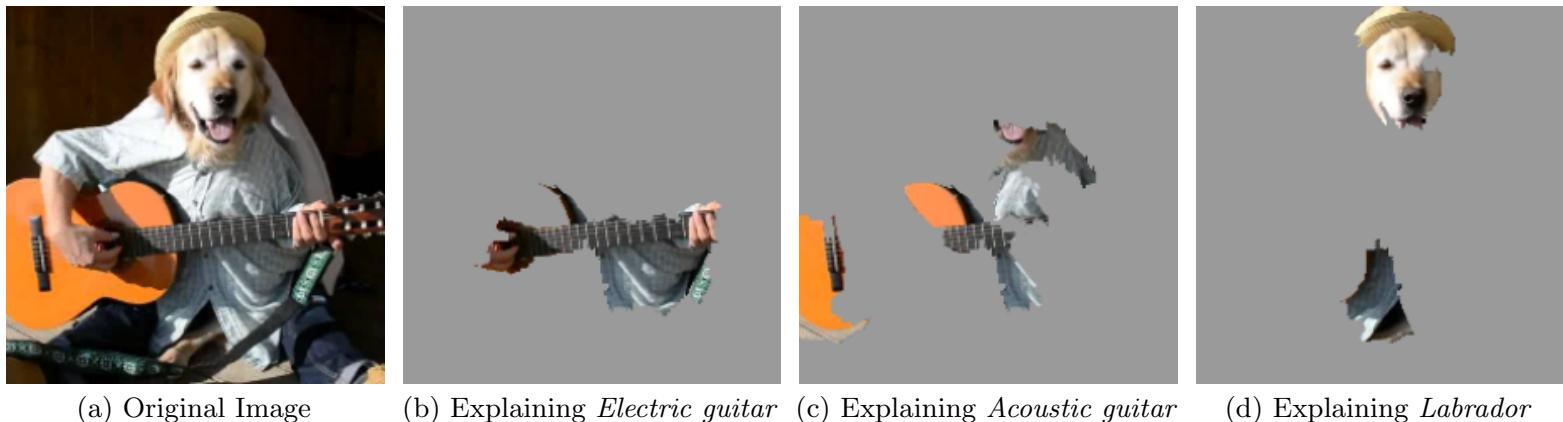


Figure 1: Explaining individual predictions. A model predicts that a patient has the flu, and LIME highlights the symptoms in the patient's history that led to the prediction. Sneeze and headache are portrayed as contributing to the "flu" prediction, while "no fatigue" is evidence against it. With these, a doctor can make an informed decision about whether to trust the model's prediction.

# Explaining any classifier?

---

- [Ribeiro et al. (2016)]: LIME (Local Interpretable Model-agnostic Explanations)



**Figure 4: Explaining an image classification prediction made by Google's Inception neural network. The top 3 classes predicted are “Electric Guitar” ( $p = 0.32$ ), “Acoustic guitar” ( $p = 0.24$ ) and “Labrador” ( $p = 0.21$ )**

## Explanations in Expert Systems

---

*Why should tetracycline not be prescribed to a child under 8 years of age?*

# Explanations in Expert Systems

---

*Why should tetracycline not be prescribed to a child under 8 years of age?*

## Supporting knowledge

Drug deposition on **developing bones**

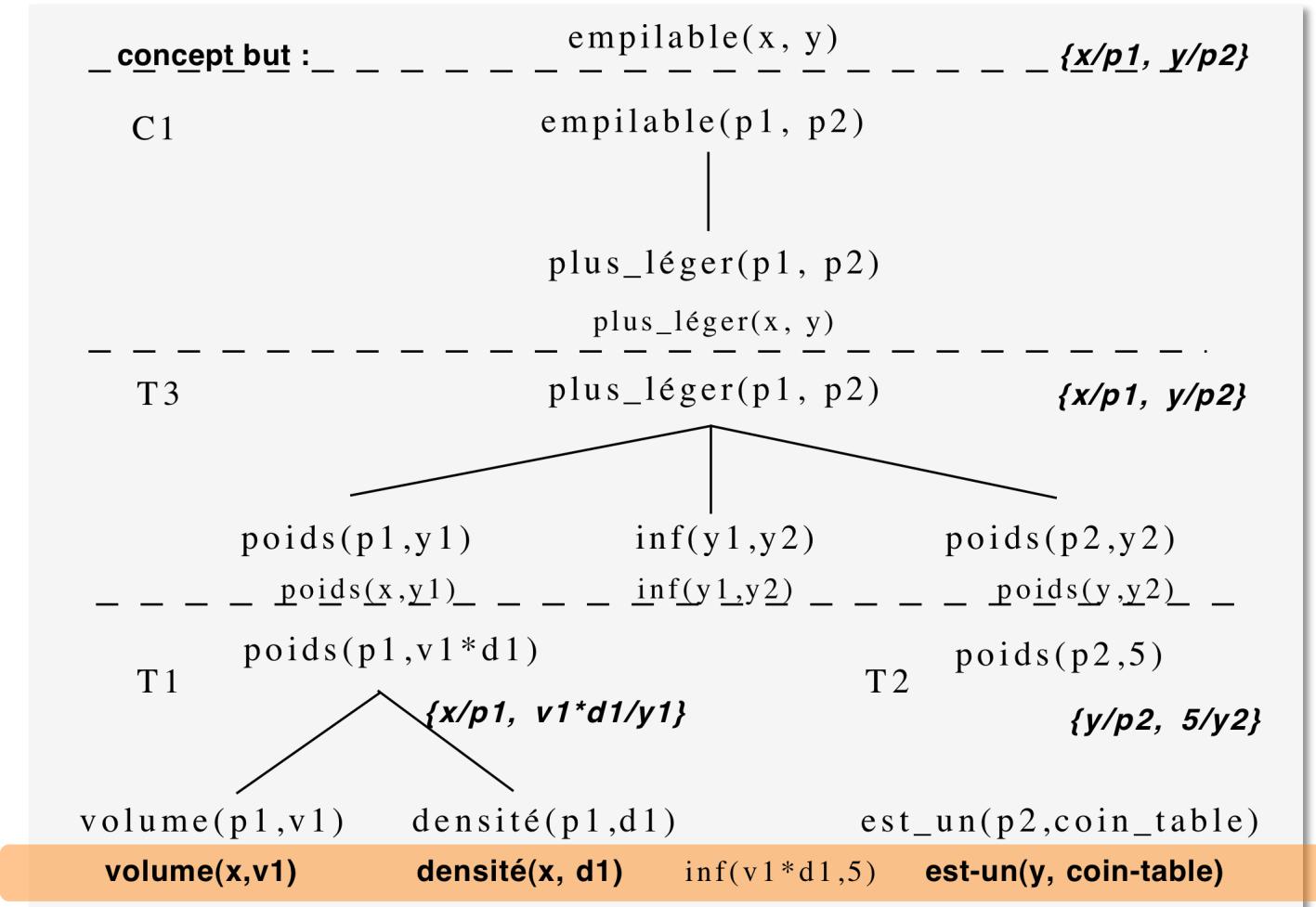
- Permanent blackening of the teeth
- Socially undesirable **staining**
- **Do not give tétracycline to children under 8 years of age**

Notion of undesirable **secondary effects**

**Causality** relationship

Explanation obtained by “decompiling” the initial reasoning

## Explanation-Based justification: decompiling



Generalized search tree resulting from regression of the target concept in the proof tree  
by computing at each step the most general literals allowing this step.

## Reasons not always in need

---

- When interpretability is **NOT** needed?

## Reasons not always in need

---

- When interpretability is **NOT** needed?
  - When **low risk** associated with the decision
    - E.g. *recommendation for a movie*
  - When **good guarantees** on performance exist
    - E.g. *character recognition*

## When reasons are in need

---

- When interpretability **IS** needed?

## When reasons are in need

---

- When interpretability **IS** needed?
  1. With **high risk decisions**
    - *E.g. chirurgical operation*
    - *E.g. shutting down a nuclear plant*
    - *E.g. autonomous vehicle*

## When reasons are in need

---

- When interpretability **IS** needed?
  1. With **high risk decisions**
    - *E.g. chirurgical operation*
    - *E.g. shutting down a nuclear plant*
    - *E.g. autonomous vehicle*

## When reasons are in need

---

- When interpretability **IS** needed?
  2. Satisfying **curiosity** (what science is about)
    - *E.g. explain surprising results*
    - *E.g. when no easy explanation exists*
    - *E.g. when the decision function must be included in a larger inference system (a domain theory)*

## When reasons are in need

---

- When interpretability **IS** needed?

### 3. Debugging / exchanges between sub-systems

- *E.g. why is that decision wrong (counterfactual)*
- *E.g. if a bicycle is recognized because it has two wheels,  
what if one is hidden behind side bags?*
- *E.g. why the system seems gender biased?*

## A paradox

---

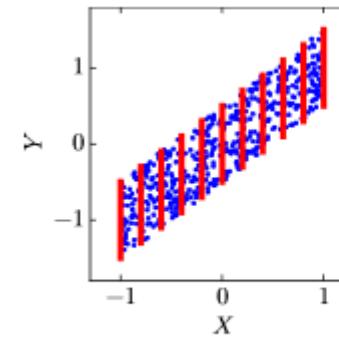
- An interpretable system **can be manipulated**
  - E.g. if someone knows that a loan is granted if you have more than 2 credit cards



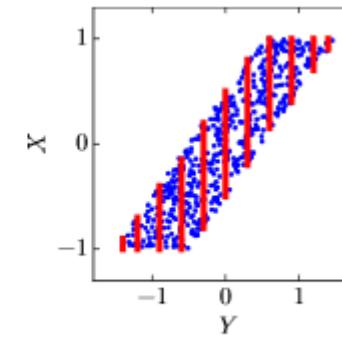
In order **not to be manipulated**,  
the predictive system **must use causal factors**

# Identification of causal relationships

- In images



(a) ANM  $X \rightarrow Y$ .



(b) ANM  $Y \rightarrow X$



[ David Lopez-Paz, Robert Nishihara, Soumith Chintala, Bernhard Schölkopf, and Léon Bottou.  
« Discovering causal signals in images ». arXiv preprint arXiv :1605.08179, 2016. ]

# Outline

---

1. What is reasoning (1)
2. Machine Learning nowadays: a path to fast thinking
3. The future of Machine Learning: what is reasoning (2)
4. Conclusion

## Claims about machine learning and reasoning

---

1. Reasoning does **not** need to **come first**
2. Intelligence lies in the **interplay of many specialized modules**
3. In order to interact in the long term and trust each other's productions, **these modules need to exchange “reasons” for their results**
4. **Reasons come after the fact.**  
They are most of the time post hoc reconstructions

## Central question

---

- How do we **reconstruct reasons** after the fact such that
  - They **provide justifications** for the conclusions reached
  - And are as **exact** and **informative** as possible

# Bibliography

---

- Léon Bottou, Jonas Peters, Joaquin Quiñonero-Candela, Denis X. Charles, D. Max Chickering, Elon Portugaly, Dipankar Ray, Patrice Simard and Ed Snelson 2013): “Counterfactual Reasoning and Learning Systems: The Example of Computational Advertising”, *Journal of Machine Learning Research*, 14(Nov):3207–3260, 2013.
- L. Bottou (2014). “From machine learning to machine reasoning: an essay”. *Machine Learning*, 94:133-149, January 2014.
- D. Kahneman (2012). Thinking, Fast and Slow. Penguin Books.
- H. Mercier & D. Sperber (2018). The enigma of reason. A new theory of human understanding. Penguin Books.
- M. T. Ribeiro, S. Singh & C. Guestrin (2016). ““Why Should I Trust You?” Explaining the Predictions of Any Classifier’. KDD-2016.

- 
- Learning modules are by essence subject to faults
    - Induction cannot be full proof
  - They therefore must justify their “hunches”
  - This justification, most of the times, comes after the fact, after the computation of the hunch
  - The justification provides plausible reasons for the production of the hunch
  - This may explain the conclusion because if it was true, or present in the input, then the conclusion would have been drawn

# Conclusions