Introduction to Reinforcement Learning

A. LAZARIC (*SequeL Team @INRIA-Lille*)
Ecole Centrale - Option DAD

SequeL – INRIA Lille

Outline

The law of effect [Thorndike, 1911]

"Of several responses made to the same situation, those which are accompanied or closely followed by ***satisfaction*** *to the animal will, other things being equal, be more firmly connected with the situation, so that, when it recurs, they will be more likely to recur; those which are accompanied or closely followed by* ***discomfort*** *to the animal will, other things being equal, have their connections with that situation weakened, so that, when it recurs, they will be less likely to occur.*

The greater the satisfaction or discomfort, the greater the strengthening or weakening of the bond."

Experimental psychology

- *Classical (human and) animal conditioning*: "the magnitude and timing of the conditioned response changes as a result of the contingency between the conditioned stimulus and the unconditioned stimulus" [Pavlov, 1927].

Experimental psychology

- *Classical (human and) animal conditioning*: "the magnitude and timing of the conditioned response changes as a result of the contingency between the conditioned stimulus and the unconditioned stimulus" [Pavlov, 1927].
- *Operant conditioning (or instrumental conditioning)*: process by which humans and animals *learn* to behave in such a way as to obtain *rewards* and avoid *punishments* [Skinner, 1938].

Experimental psychology

- *Classical (human and) animal conditioning*: "the magnitude and timing of the conditioned response changes as a result of the contingency between the conditioned stimulus and the unconditioned stimulus" [Pavlov, 1927].
- *Operant conditioning (or instrumental conditioning)*: process by which humans and animals *learn* to behave in such a way as to obtain *rewards* and avoid *punishments* [Skinner, 1938].

Remark: ***reinforcement*** denotes any form of conditioning, either positive (*rewards*) or negative (*punishments*).

Computational neuroscience

- *Hebbian learning*: development of formal models of how the synaptic weights between neurons are reinforced by simultaneous activation. *"Cells that fire together, wire together."* [Hebb, 1961].

Computational neuroscience

- *Hebbian learning*: development of formal models of how the synaptic weights between neurons are reinforced by simultaneous activation. *"Cells that fire together, wire together."* [Hebb, 1961].
- *Emotions theory*: model on how the emotional process can bias the decision process [Damasio, 1994].

Computational neuroscience

- *Hebbian learning*: development of formal models of how the synaptic weights between neurons are reinforced by simultaneous activation. *"Cells that fire together, wire together."* [Hebb, 1961].
- *Emotions theory*: model on how the emotional process can bias the decision process [Damasio, 1994].
- *Dopamine and basal ganglia model*: direct link with motor control and decision-making (e.g., [Doya, 1999]).

Computational neuroscience

- *Hebbian learning*: development of formal models of how the synaptic weights between neurons are reinforced by simultaneous activation. *"Cells that fire together, wire together."* [Hebb, 1961].
- *Emotions theory*: model on how the emotional process can bias the decision process [Damasio, 1994].
- *Dopamine and basal ganglia model*: direct link with motor control and decision-making (e.g., [Doya, 1999]).

Remark: ***reinforcement*** denotes the effect of dopamine (and surprise).

Optimal control theory and dynamic programming

- *Optimal control*: *formal framework* to define optimization methods to derive control policies in continuous time control problems [Pontryagin and Neustadt, 1962].

Optimal control theory and dynamic programming

- *Optimal control*: *formal framework* to define optimization methods to derive control policies in continuous time control problems [Pontryagin and Neustadt, 1962].
- *Dynamic programming*: set of methods used to *solve control problems* by decomposing them into subproblems so that the optimal solution to the global problem is the conjunction of the solutions to the subproblems [Bellman, 2003].

Optimal control theory and dynamic programming

- *Optimal control*: *formal framework* to define optimization methods to derive control policies in continuous time control problems [Pontryagin and Neustadt, 1962].
- *Dynamic programming*: set of methods used to *solve control problems* by decomposing them into subproblems so that the optimal solution to the global problem is the conjunction of the solutions to the subproblems [Bellman, 2003].

Remark: ***reinforcement*** denotes an objective function to maximize (or minimize).

Reinforcement learning

Learn of a behavior strategy (a *policy*) which maximizes the long term sum of rewards (*delayed reward*) by a direct interaction (*trial-and-error*) with an unknown and uncertain environment.

Reinforcement learning

Learn of a behavior strategy (a *policy*) which maximizes the long term sum of rewards (*delayed reward*) by a direct interaction (*trial-and-error*) with an unknown and uncertain environment.

Reinforcement learning

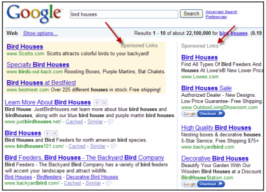
Learn of a behavior strategy (a *policy*) which maximizes the long term sum of rewards (*delayed reward*) by a direct interaction (*trial-and-error*) with an unknown and uncertain environment.

Reinforcement learning

Learn of a behavior strategy (a *policy*) which maximizes the long term sum of rewards (*delayed reward*) by a direct interaction (*trial-and-error*) with an unknown and uncertain environment.

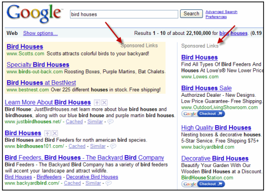
Reinforcement learning

Learn of a behavior strategy (a *policy*) which maximizes the long term sum of rewards (*delayed reward*) by a direct interaction (*trial-and-error*) with an unknown and uncertain environment.

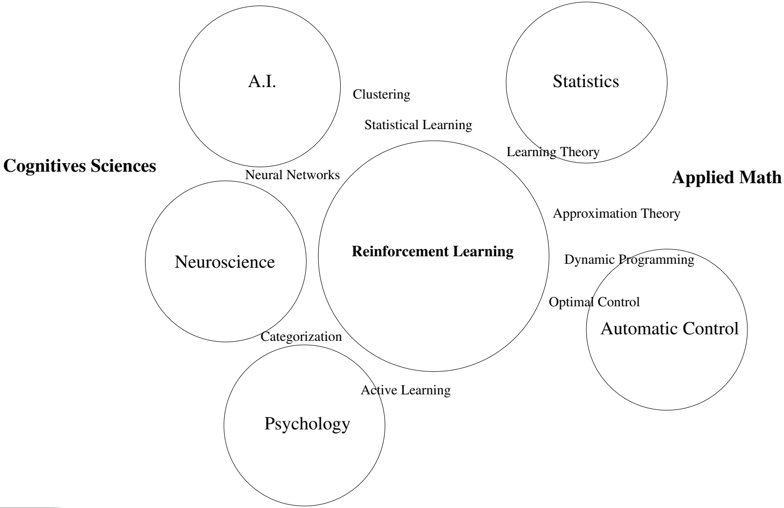


Reinforcement learning

Learn of a behavior strategy (a *policy*) which maximizes the long term sum of rewards (*delayed reward*) by a direct interaction (*trial-and-error*) with an unknown and uncertain environment.



A multi-disciplinary field



A machine learning paradigm

- *Supervised learning:* an expert (*supervisor*) provides examples of the right strategy (e.g., classification of clinical images). *Supervision is expensive.*

A machine learning paradigm

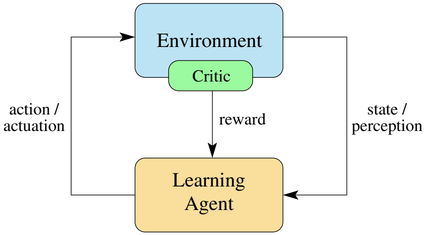
- *Supervised learning:* an expert (*supervisor*) provides examples of the right strategy (e.g., classification of clinical images). *Supervision is expensive.*
- *Unsupervised learning:* different objects are clustered together by similarity (e.g., clustering of images on the basis of their content). *No actual performance is optimized.*

A machine learning paradigm

- *Supervised learning:* an expert (*supervisor*) provides examples of the right strategy (e.g., classification of clinical images). *Supervision is expensive.*
- *Unsupervised learning:* different objects are clustered together by similarity (e.g., clustering of images on the basis of their content). *No actual performance is optimized.*
- *Reinforcement learning:* learning by direct interaction (e.g., autonomous robotics). *Minimum level of supervision (reward) and maximization of long term performance.*

Outline

The Agent-Environment Interaction Protocol



for $t = 1, \ldots, n$ **do**
 The agent perceives state s_t
 The agent performs action a_t
 The environment evolves to s_{t+1}
 The agent receives reward r_t
end for

The Agent-Environment Interaction Protocol

The environment

- *Controllability*: fully (e.g., chess) or partially (e.g., portfolio optimization)
- *Uncertainty*: deterministic (e.g., chess) or stochastic (e.g., backgammon)
- *Reactive*: adversarial (e.g., chess) or fixed (e.g., tetris)
- *Observability*: full (e.g., chess) or partial (e.g., robotics)
- *Availability*: known (e.g., chess) or unknown (e.g., robotics)

The Agent-Environment Interaction Protocol

The environment

- *Controllability*: fully (e.g., chess) or partially (e.g., portfolio optimization)
- *Uncertainty*: deterministic (e.g., chess) or stochastic (e.g., backgammon)
- *Reactive*: adversarial (e.g., chess) or fixed (e.g., tetris)
- *Observability*: full (e.g., chess) or partial (e.g., robotics)
- *Availability*: known (e.g., chess) or unknown (e.g., robotics)

The critic

- Sparse (e.g., win or loose) vs informative (e.g., closer or further)
- Preference reward
- Frequent or sporadic
- Known or unknown

The Agent-Environment Interaction Protocol

The environment

- *Controllability*: fully (e.g., chess) or partially (e.g., portfolio optimization)
- *Uncertainty*: deterministic (e.g., chess) or stochastic (e.g., backgammon)
- *Reactive*: adversarial (e.g., chess) or fixed (e.g., tetris)
- *Observability*: full (e.g., chess) or partial (e.g., robotics)
- *Availability*: known (e.g., chess) or unknown (e.g., robotics)

The critic

- Sparse (e.g., win or loose) vs informative (e.g., closer or further)
- Preference reward
- Frequent or sporadic
- Known or unknown

The agent

- Open loop control
- Close loop control (i.e., *adaptive*)
- Non-stationary close loop control (i.e., *learning*)

The Problems

- *How do we formalize the agent-environment interaction?*

The Problems

- *How do we formalize the agent-environment interaction?*
- *How do we solve an RL problem?*

The Problems

- *How do we formalize the agent-environment interaction?*
- *How do we solve an RL problem?*
- *How do we solve an RL problem "online"?*

The Problems

- *How do we formalize the agent-environment interaction?*
- *How do we solve an RL problem?*
- *How do we solve an RL problem "online"?*
- *How do we collect useful information to solve an RL problem?*

The Problems

- *How do we formalize the agent-environment interaction?*
- *How do we solve an RL problem?*
- *How do we solve an RL problem "online"?*
- *How do we collect useful information to solve an RL problem?*
- *How do we solve a "huge" RL problem?*

The Problems

- *How do we formalize the agent-environment interaction?*
- *How do we solve an RL problem?*
- *How do we solve an RL problem "online"?*
- *How do we collect useful information to solve an RL problem?*
- *How do we solve a "huge" RL problem?*
- *How "sample-efficient" RL algorithms are?*

Bibliography I

Bellman, R. (2003).
Dynamic Programming.
Dover Books on Computer Science Series. Dover Publications, Incorporated.

Damasio, A. R. (1994).
Descartes' Error: Emotion, Reason and the Human Brain.
Grosset/Putnam.

Doya, K. (1999).
What are the computations of the cerebellum, the basal ganglia, and the cerebral cortex.
Neural Networks, 12:961–974.

Hebb, D. O. (1961).
Distinctive features of learning in the higher animal.
In Delafresnaye, J. F., editor, *Brain Mechanisms and Learning*. Oxford University Press.

Pavlov, I. (1927).
Conditioned reflexes.
Oxford University Press.

Bibliography II

Pontryagin, L. and Neustadt, L. (1962).
The Mathematical Theory of Optimal Processes.
Number v. 4 in Classics of Soviet Mathematics. Gordon and Breach Science Publishers.

Skinner, B. F. (1938).
The behavior of organisms.
Appleton-Century-Crofts.

Thorndike, E. (1911).
Animal Intelligence: Experimental Studies.
The animal behaviour series. Macmillan.

Reinforcement Learning

Alessandro Lazaric
alessandro.lazaric@inria.fr
sequel.lille.inria.fr