ARTIFICIAL AND COMPUTATIONAL INTELLIGENCE

Learning Strategies in Machine Learning

Adrian Horzyk

horzyk@agh.edu.pl

AGH University of
Science and Technology
Krakow, Poland

Learning Strategies

Main learning strategies are:

- **Supervised learning** – makes predictions about data or classify data.
- **Unsupervised learning** – groups data according to their similarity.
- **Reinforcement learning** – interacts with the environment and maximizes a cumulative reward that controls a training process where data are sequential in time. It is an area of machine learning concerned with how agents ought to take actions in the environment so as to maximize a cumulative reward.
- **Motivated learning** – defines fundamental needs and automatically develops secondary needs which affect the fundamental ones and control the interactions with the environment. During the learning process, fundamental needs should be satisfied what minimize pain (a penalty) and maximize pleasure (a reward) – they work as motivating factors.
- **Associative learning (cognitive learning)** – aggregates the representation of similar features and objects, links them due to their real relations and actions of various kinds, connects them with different strengths and allows to trigger created associations recalling back related objects for a given context in time.

Supervised Learning (SL)

Supervised learning is a kind of machine learning that maps an input (vector, matrix) to an output (vector, matrix) based on the set of samples consisting of input-output pairs.

Training sample set $\mathbb{S} = \{P^1, \dots, P^N\}$ is defined as a set of **pairs** $P^n = [X^n, Y^n]$, where the **input** $X^n = [x_1^n, \dots, x_K^n] \in X$ and **output** $Y^n = [y_1^n, \dots, y_L^n] \in Y$ are usually defined as vectors (sometimes as matrices).

The goal of supervised learning is to model the inferred function f which:

$$f: X \to Y$$

and for each n = 1, …, N we try to train the model to $f(X^n) \cong Y^n$, so we try to minimize the error $\delta = Y^n - f(X^n)$ and simultaneously achieved good generalization properties of the model. Such function can model classification or regression tasks.

The **generalization** means the ability of the model to achieved $f(X^n) \cong Y^n$ also for pairs that have not been used during the training process (learning of the model).

Supervised learning algorithms and models: delta rule, backpropagation, multilayer perceptrons, radial basis function networks, support vector machine, linear regression, logistic regression, naive Bayes, decision trees, k-nearest neighbors etc.

To use supervised learning, we need to have defined a training set of pairs that must be prepared by an expert or taken from some experiments. This might be a disadvantage!

Unsupervised Learning (UL)

Unsupervised learning is a kind of machine learning that uses input data (vectors or matrices) to self-organize, group or cluster them. During this learning process, we try to use similarities between input objects which allow to achieved this goal.

Training data $\mathbb{S} = \{X^1, \dots, X^N\}$ are not labelled, classified or categorized where $X^n = [x_1^n, \dots, x_K^n] \in X$ and we have no strictly defined goal of the learning process in terms of results that should be achieved.

It is also associated with hierarchical clustering, k-means, self-organizing maps, neural gas, Hebbian learning, autoencoders, deep belief nets, generative adversarial networks.

The goal of unsupervised learning is to better understand data, their groups, local densities, find clusters and relations between the data which differentiate them.

Reinforcement Learning (RL)

Reinforcement learning allows us to act in accordance with what is most probable and then evaluate whether we are correct or wrong that can be expressed in terms of a reward that is used to adapt the model.

Reinforcement learning refers to goal-oriented algorithms that learn how to attain a complex objectives (goals) or maximize along a particular dimension over many steps.

Reinforcement learning reminds the child's education, starting from scratch.
Like a child incentivized by spankings and candy, these algorithms are **penalized** when they make the wrong decisions and **rewarded** when they make the right ones – this is reinforcement.

Reinforcement Learning (RL)

Reinforcement learning uses:

Agent (I) – which is trained to choose or decide possibly in the best way choosing from a set of possible actions.

Action set (A) – is a set of possible actions (e.g. moves) that the agent can make. Each action in a given context should be rewarded or penalized to some extent to control the learning process.

Environment (E) – must interact with an agent, taking the agent's current state and action as inputs, and returns agent's reward or penalty as an output in its state, e.g. the environment could be the laws of physics and the rules of society that process the agent's actions and determine the consequences of them.

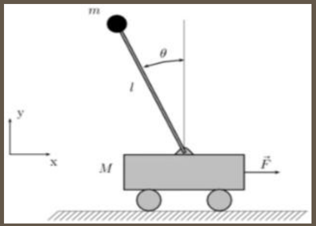
State (S) – is a specific and immediate situation in which the agent finds itself like places and moments, instantaneous configurations that put the agent in relations to other significant objects, obstacles, enemies, friends, prizes or losses.

Reward(R) – is feedback by which we measure the success or failure of agent's actions (choices, decisions). Rewards can be immediate or delayed. They evaluate the agent's actions or reactions.

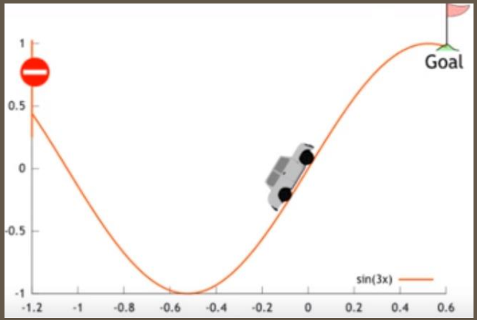
Policy (P) – is a strategy that the agent employs to determine the next action (move, choice, decision) based on the current state. It maps states to actions that promise the highest rewards!

Reinforcement Learning (RL)

Reinforcement learning allows to learn physics of objects to achieve a goal (defined as a reward). Some classic problems solved by RL are:



- **Cart-Pole problem** – where a goal is to balance a pole on a cart as long as possible.
- **Mountain Car problem** – where a car should get to the top of the mountain and we can use momentum because the car has not enough power to go straight up.
- **Playing video games** where we need to perform a sequence of actions (or reactions) to hit the target or achieved a goal.
- **Learning a physics** of movement in real environment.

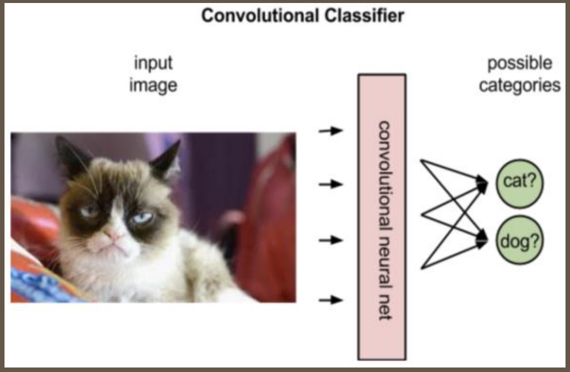


Other approaches try to learn the best move for a given state:

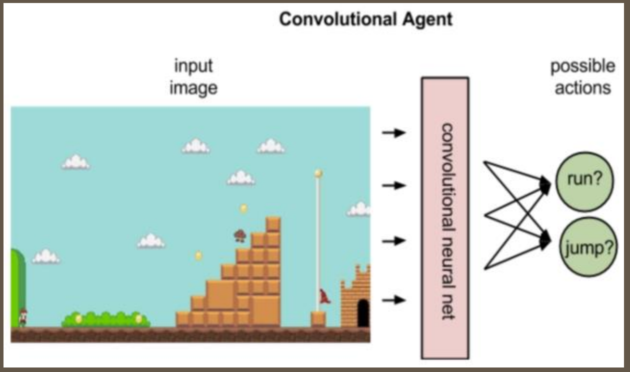
- **Markov Decision Processes**
- **Dynamic programming**
- **Monte Carlo**
- **Temporal Difference Learning**
- **Direct Policy Search**
- **Q-learning** (Q-tables grow exponentially with data!)
- **SARSA, A3C, DQN, DDPG, NAF, TRPO, PPO…**

Deep Reinforcement Learning

as a combinations of deep learning and reinforcement learning is used to power AlphaGo in a world championship, create self-driving cars, or for playing video games.



Convolutional networks (CNN) can be used to **rank the possible actions** to perform in that state, e.g. jumping or running, instead of finding a label to an image as in supervised learning approach.



Motivated Learning (ML)

Motivated learning develops a hierarchy of needs which help the agent to satisfy the defined needs (achieve goals). In this approach the agent:

- discovers and tests the environment,
- remembers results of undertaken actions and their results in a context,
- combines actions to try to satisfy needs,
- creates secondary needs which help to satisfy the defined needs,
- learns how to avoid penalties and get rewards satisfying secondary needs first when the defined goals are not directly attainable or the direct way is associated with penalties as well,
- tries to maximize rewards and minimize penalties.

Associative Learning (ASSL)

Associative learning is a kind of knowledge-based learning:

- aggregates representation of the same and similar objects,
- creates groups, clusters, and classes of objects automatically,
- links related objects as parts of the environment,
- orders objects and features which define these objects,
- self-organizes the objects and remembers them contextually,
- allows to trigger related objects and recall them context-sensitively,
- enables to define needs and use them in the adaptation process,
- allows defining actions in the context of needs and other objects.

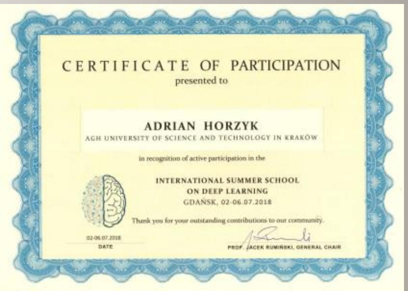
Bibliography and Literature

1. Nikola K. Kasabov, ***Time-Space, Spiking Neural Networks and Brain-Inspired Artificial Intelligence***, In Springer Series on Bio- and Neurosystems, Vol 7., Springer, 2019.
2. Ian Goodfellow, Yoshua Bengio, Aaron Courville, ***Deep Learning***, MIT Press, 2016, ISBN 978-1-59327-741-3 or PWN 2018.
3. Holk Cruse, ***Neural Networks as Cybernetic Systems***, 2nd and revised edition
4. R. Rojas, ***Neural Networks***, Springer-Verlag, Berlin, 1996.
5. ***Convolutional Neural Network*** (Stanford)
6. ***Visualizing and Understanding Convolutional Networks***, Zeiler, Fergus, ECCV 2014
7. IBM: **https://www.ibm.com/developerworks/library/ba-data-becomes-knowledge-1/index.html**
8. NVIDIA: **https://developer.nvidia.com/discover/convolutional-neural-network**
9. **A. Horzyk**, J. A. Starzyk, J. Graham, ***Integration of Semantic and Episodic Memories***, IEEE Transactions on Neural Networks and Learning Systems, Vol. 28, Issue 12, Dec. 2017, pp. 3084 - 3095, 2017, DOI: 10.1109/TNNLS.2017.2728203.
10. **A. Horzyk**, J.A. Starzyk, ***Multi-Class and Multi-Label Classification Using Associative Pulsing Neural Networks***, IEEE Xplore, In: 2018 IEEE World Congress on Computational Intelligence (WCCI IJCNN 2018), 2018, (in print).
11. **A. Horzyk**, J.A. Starzyk, ***Fast Neural Network Adaptation with Associative Pulsing Neurons***, IEEE Xplore, In: 2017 IEEE Symposium Series on Computational Intelligence, pp. 339 -346, 2017, DOI: 10.1109/SSCI.2017.8285369.
12. **A. Horzyk**, K. Gołdon, ***Associative Graph Data Structures Used for Acceleration of K Nearest Neighbor Classifiers***, LNCS, In: 27th International Conference on Artificial Neural Networks (ICANN 2018), 2018, (in print).
13. **A. Horzyk**, ***Deep Associative Semantic Neural Graphs for Knowledge Representation and Fast Data Exploration***, Proc. of KEOD 2017, SCITEPRESS Digital Library, pp. 67 - 79, 2017, DOI: 10.13140/RG.2.2.30881.92005.
14. **A. Horzyk**, ***Neurons Can Sort Data Efficiently***, Proc. of ICAISC 2017, Springer-Verlag, LNAI, 2017, pp. 64 - 74, ICAISC BEST PAPER AWARD 2017 sponsored by Springer.
15. https://skymind.ai/wiki/deep-reinforcement-learning
16. The Complete Guide to Mastering Artificial Intelligence using Deep Learning and Neural Networks, https://www.udemy.com/deep-reinforcement-learning-in-python/

Adrian Horzyk

horzyk@agh.edu.pl

Google: Horzyk



University of Science and Technology in Krakow, Poland