TD 8 – Learning II Supervised Learning

Practical Information

TD Assistant
eponiatowski@clipper.ens.psl.eu

TD Material
https://github.com/esther-poniatowski/2223_UlmM2_ThNeuro

Goals of the TD

Three types of learning algorithms are studied in this series of TDs : unsupervised, supervised and reinforcement learning.

This TD presents the standard paradigm of the **supervised perceptron**. The perceptron model originates with Rosenblatt in 1958 and marks the beginning of Artificial Neural Networks, which have now attained high performance in various tasks, placing learning as a central question of neurosciences.

Part 1 General model of the Perceptron.

Part 2 Proof of the convergence of the Perceptron algorithm to a solution for a binary classification (when such a solution exists).

1 Perceptron model

Perceptron

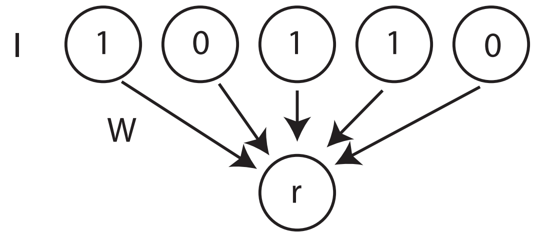
The Perceptron can be conceived as a single 'output' neuron receiving signals from N 'input' neurons.

At any time, the inputs received by the output neuron can be represented by an **input vector** $\vec{I}$, in which each entry I_i equals the activity of neuron i. In the simplest model, neurons are binary units, which can be either active or silent, therefore the vector $\vec{I}$ contains only 0s and 1s.

Input neurons communicate with the output neuron through synapses. Synaptic strengths are contained in a **weights vector** $\vec{W}$, in which the entry W_n is the strength of the synapse connecting neuron n to the output neuron.

The "computation" performed by the output neuron is a weighted sum of the inputs, followed by thresholding. Specifically, the output neuron is active (value 1) if the total weighted input it receives is larger than a threshold θ, ortherwise it stays silent (value 0) .

The Perceptron can be used to perform a **classification** task, which consists in associating a set of P input patterns $\vec{I_p}$ with a target output $r_p \in \{0, 1\}$. To achieve this function, **learning** consists in adjusting the parameters $\vec{W}$ and θ until the output of the computation gives the correct value for all the patterns to classify.



① Expression the total input received by the output neuron when a given pattern $\vec{I_p}$ is presented.

② Write a condition on the input patterns $\vec{I_p}$, the target outputs r_p and the parameters $\vec{W}$ and θ such that the task is solved.

In order to simplify the condition, it is convenient to introduce an additional input $I_0 = 1$ which is always turned on, and an additional weight $W_0 = -\theta$.

③ Rewrite the condition as a function of $(I_0, \vec{I_p})$ and $(W_0, \vec{W})$. Give a vectorial relation with modified vectors.

④ Transform the input patterns $\vec{I_p}$ into a set of other input patterns $\vec{J_p}$ such that the condition is equivalent to :

$$\forall p, \quad \vec{J_p} \cdot \vec{W} > 0$$

2 Perceptron algorithm

Perceptron algorithm

To correctly associate inputs to their respective output, the perceptron is trained according to an iterative learning algorithm :

- Randomly pick an input pattern $\vec{J}_p$
- If $\vec{J}_p \cdot \vec{W} > 0$, pick a new input pattern.
- If $\vec{J}_p \cdot \vec{W} < 0$, perform a learning step : $\vec{W}(t+1) = \vec{W}(t) + \epsilon \vec{J}_p$

The fundamental property of the perceptron algorithm is that, *if there exists a solution* $\vec{W}^*$ to the classification problem, then it necessarily converges towards a solution.

Notation

$\cos[\alpha(t)] = \dfrac{\vec{W}(t) \cdot \vec{W}^*}{\|\vec{W}(t)\| \|\vec{W}^*\|}$ with $\alpha(t)$ the angle between the vectors $\vec{W}(t)$ and $\vec{W}^*$.

5 Explain why the perceptron algorithm is an implementation of *supervised learning*. Explain the meaning of *generalization* after learning.

6 Sketch the situation in a 2D graph, representing :
- A few input patterns $\vec{J}_p$.
- A line dividing them in two classes.
- A vector $\vec{W}^*$ which is a solution for this classification (at least one).
- A current weight vector $W(\vec{t})$ which is not a solution yet.
- A learning step $\epsilon \vec{J}_p$.

7 Introduce $l = \min_p \vec{J}_p \cdot \vec{W}^* > 0$ and find a lower bound on $\vec{W}(t+1) \cdot \vec{W}^*$ given $\vec{W}(t) \cdot \vec{W}^*$.
Considering $\vec{W}(0) = \vec{0}$, deduce a lower bound on $\vec{W}(t) \cdot \vec{W}^*$.

8 Introduce $L = \max_p \|\vec{J}_p\|^2$ and find an upper bound on $\|\vec{W}(t+1)\|^2$ given $\|\vec{W}(t)\|^2$.
Deduce an upper bound on $\|\vec{W}(t)\|^2$.

9 Given those results, justify that $\cos[\alpha(t)]$ admits the following lower bound (assuming that $\vec{W}^*$ is a unit vector) :

$$\cos[\alpha(t)] \geq \frac{t\epsilon l}{\sqrt{t\epsilon^2 L}} = \sqrt{t}\frac{l}{L}$$

10 Conclude about the convergece of the algorithm towards a solution.