HUMAN PERCEPTRON, 2016

INTRODUCTION

The workshop “Human Perceptron” invites participants to manually compute all the steps of a Perceptron, the very first neural network, designed to identify classes of objects. Participants embody a primitive neural network through all the necessary steps towards the completion of a single operation. This provides opportunities to understand and deconstruct the underlying logic behind Artificial Intelligence, and to elaborate an informed critique; leading to speculations about what other kinds of logics could govern AI.

The “Human Perceptron” workshop takes advantage of the Perceptron, an algorithm invented by Frank Rosenblatt in 1958. The Perceptron proceeds to a suite of recursive calculations, to separate two classes of objects, identified by a set of data through supervised learning, until the objects have been classified. Based on this learning phase, the Perceptron is then able to make predictions and recognize any given object as belonging to its recorded classes.

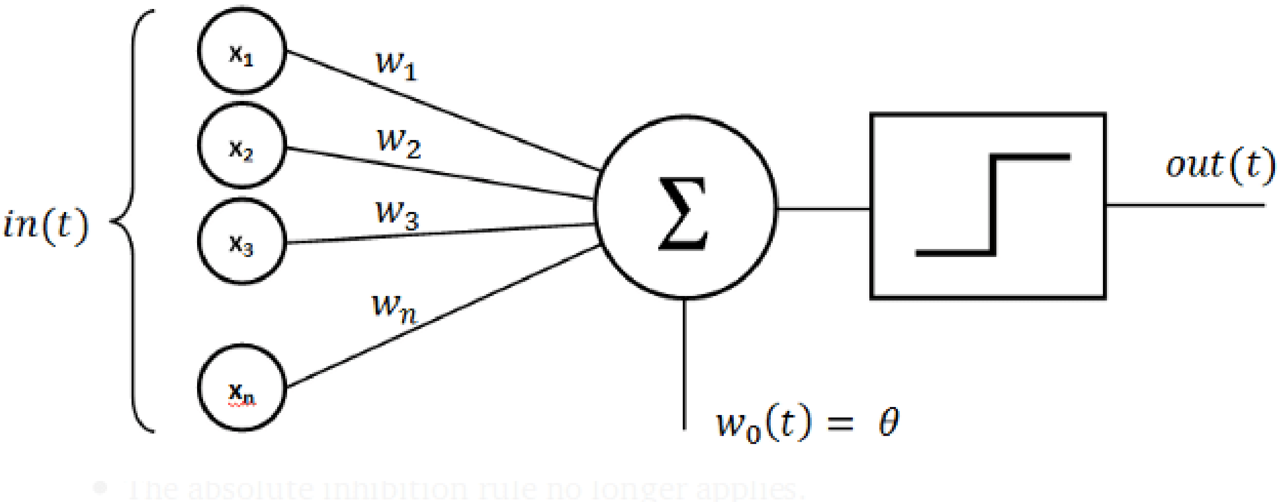

Fig 2: Diagram of a Perceptron

Known as the first neural network, the Perceptron exemplifies the logic at work in more recent Artificial Intelligence computing processes. Supervised machine learning operates on sets of data derived from the quantification of objects, making them computable and predictable. The complexity of neural networks has greatly improved and evolved since the Perceptron (multi-layer Perceptrons, deep convolutional neural networks, generative adversarial networks, etc). This has been enabled by the increase of available computing power, and by the development of facilities that provide prepared data, such as Amazon Mechanical Turk (MTurk). We presume, however, that the differences between the Perceptron and more recent neural networks is mostly quantitative and does not fundamentally affect the nature of the logic at work.

The Perceptron also has the advantage to perfectly embody the Cybernetics narrative, and analogies between computing and the nervous system. The model is simple enough to be executed manually until its completion, through a limited number of steps. It remains understandable in its successive steps, providing a base on which to build up an accurate understanding of Artificial Intelligence. This in turn can be mobilised with regards to ongoing issues such as digital labor, manual training, construction of data sets, algorithmic biases, etc.

PREREQUISITES

Technical Requirements :
• Paperboard
• Red thread and pins

- Several tables and chairs of various forms
- Rulers and pens
- Optional: a video projector and a computer

Attribution of roles / functions to the participants :
- 2 (or more) teams with, in each team
- 1 person (or more) to measure the objects
- 1 person (or more) to index the data
- 1 person (or more) to make the calculations

In addition, 1 person to be the “oracle”, verifying the separator line position, and repositioning it if necessary.

PROCEDURE

1. Data Collection

Data collection is executed manually by the participants, with the help of a ruler. For each piece of furniture, participants collect 2 values: 1) height 2) area, and the the type of object: table or chair.

Participants construct a data set, using pieces of paper, listing all of their collected measurements. For instance, a paper could have this information:
Table: height=0.8 (m), area=1 (m2)
Or, in another example:
Chair: height=0.5 (m), area=0.25 (m2)
All papers (data set) are put in a bag.

2. Graph and separator positioning

On the graph, the X axis is assigned to height, and the Y axis to the area of the measured furniture pieces.

A line (the separator), is materialised by a red thread and placed on the graph, separating it in two zones: the line is placed in an arbitrary position at the start of the process.

Each area on both sides of the line is arbitrarily assigned one of the two types of furniture: a zone for chairs and a zone for tables.

3. Training phase

The most important part in the elaboration of a neural network is the training phase. During this phase, the neural network “learns” to recognize patterns/entities.

Here is the transposition for a human Perceptron: during the training phase, one piece of furniture is randomly picked from the bag, the participants verify if its position on the graph (with coordinates x=height and y=area) is located in the right zone.

If the piece of furniture is in the right zone, then it is placed back in the bag and another paper is drawn.

If the point is in the wrong zone, the participants have to re-compute the position of the separator, with the help of the function “recalculate line”.
Recalculate line function:
if furniture is in the wrong zone:

if furniture = chair
D = -1
if furniture = table
D = 1
then:
line = line + (height, area, 1) * 0.1 * D

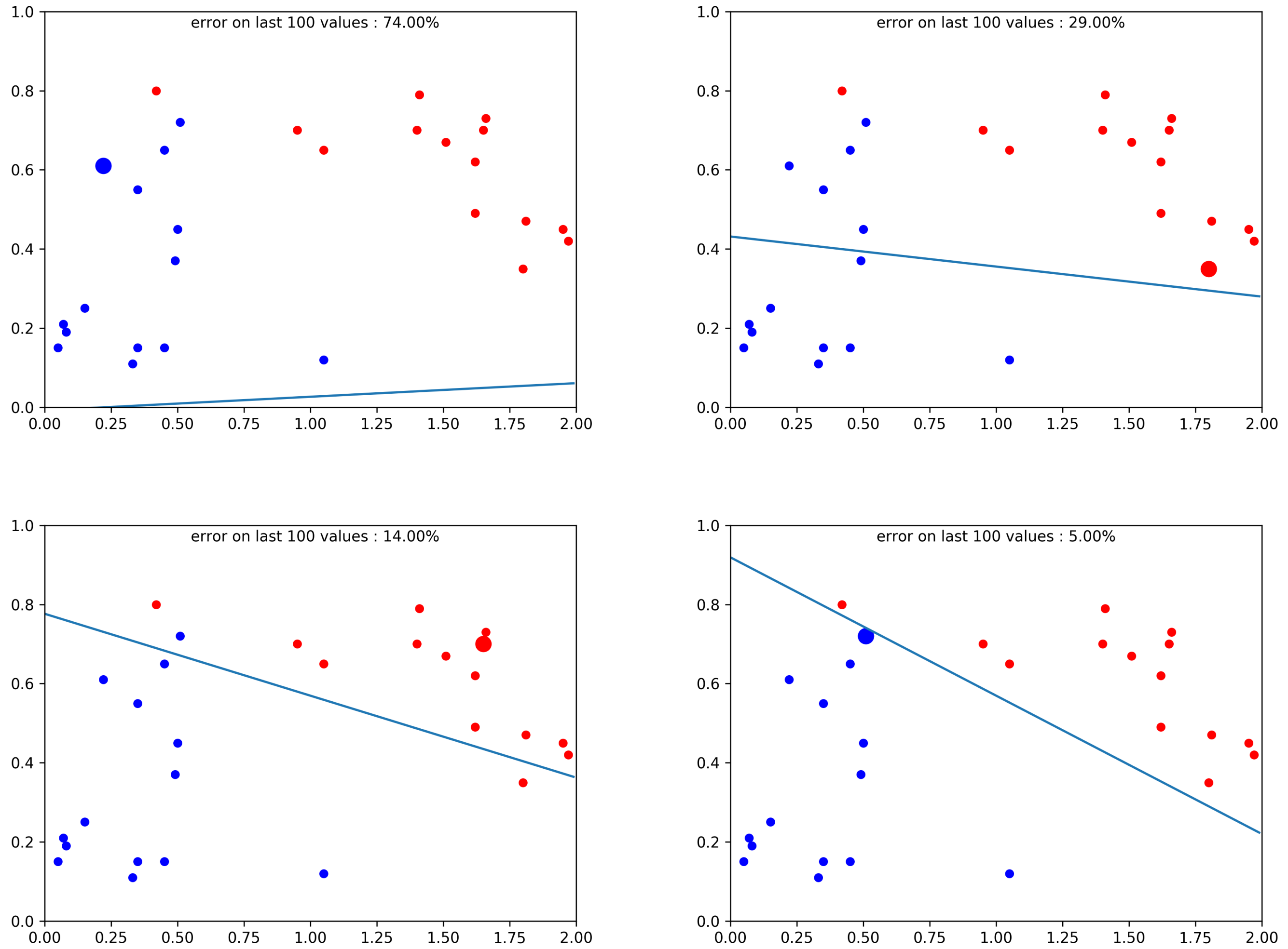

Fig 4: position of the separator, at the beginning, in intermediary steps and at the end of the process

Calculation after calculation, the position of the separator will be be corrected, a little bit closer to its final position each time: the training cycle is designed to recalculate the position of the separator until it separates the chairs from the tables with as much as precision as possible.

EXAMPLE

Example of calculation for a new position of the separator

We start with an arbitrary line x = y ; the line passes through the points (0, 0) and (1, 1).

So the line can be defined as: x - y = 0 ; equivalent to 1x - 1y + 0 = 0 equivalent to (1, -1, 0).

So, the line has the coordinates:
line = (1, -1, 0)
We pick an imaginary chair, its height is = 0.5 and its area = 0.25 (so on the graph: x=0.5 and y=0.25)

As the separator line position is (1, -1, 0), this chair is in the wrong zone.
D = -1
We recalculate the separator position:

New line position
= (1, -1, 0) + (0.5, 0.25, 1) * 0.1 * -1

= (1, -1, 0) + (-0.05, -0.025, -0.1)

= (0.95, -1.025, -0.1)
We now have the new equation for the separator line:
0.95x - 1.025y - 0.1 = 0
Calculate the coordinates of two points of the line to reposition it on the graph.
if x = 0, y = -0.1/1.025 = -0.098

if x = 1, y = 0.85/1.025 = 0.83
Reposition the line on the graph, according to these two points.

_
First iteration of Human Perceptron was done in the frame of Algorithmic design workshop, prototype of a human operated Perceptron, RYBN.ORG, PACT Zollverein, Essen, 2016.