

## Using the Perceptron class

The perceptron behaves like a single neuron. You specify the perceptron's threshold value, input stimuli and the weights associated with each input, and then run the neuron to produce an output value. If the sum of the weighted inputs is greater than the threshold, the perceptron fires and has an output value of 1. Otherwise, the perceptron will have an output value of 0.

The perceptron class can be used to compute any kind of output (such as the decision to turn, go forward, change state, etc), but the simplest example of an agent using a perceptron is to decide when to fire a shot towards a WUB. There are a number of factors which determine whether an agent should shoot at a WUB: its energy level, the distance to the WUB, the angle to the WUB, etc. These factors can each be used as an input to a perceptron.

The steps and methods required to setup a Perceptron to control an agent's firing are outlined below.

- 1) Create a Perceptron instance
- 2) Make the Perceptron control the Agent's firing
- 3) Add inputs to the Perceptron and adjust their weights
- 4) Update the Perceptron's inputs at every iteration step
- 5) The gun fires automatically when the Perceptron fires!
- 6) Adjust the weights to find the optimal shooting behavior

### **The following methods are part of the Perceptron class.**

- `set-threshold to newThreshold`: sets the threshold of the perceptron

Example:

```
perceptron set-threshold to 1.0.
```

- `add-weighted-input with-weight newWeight`: adds a new input to the perceptron with the specified weight. This method also returns a number indicating the index of the newly created input. You will refer to this index when using the methods `set-weight` and `set-input`.

Example:

```
perceptron add-weighted-input with-weight 1.0.
```

- `set-weight number N to value`: sets the *N*th weight to the specified value

Example:

```
perceptron set-weight number 3 to .7.
```

- `set-input number N to value`: sets the *N*th input to the specified value

Example:

```
perceptron set-input number 3 to 1.0.
```

- `run`: uses the weights and inputs to compute the perceptron's output

Example:

```
perceptron run.
```

- `get-output`: after the perceptron has been run, this returns the perceptron's output value

Example:

```
if (perceptron get-output) == 1: self fire.
```

**The following method is used by an Agent to control its firing with a Perceptron.**

- control-gun with *perceptron*: make's *perceptron* control how the agent fires shots

Example:

```
self control-gun with thePerceptron.
```