

TP NN (Neural Networks)

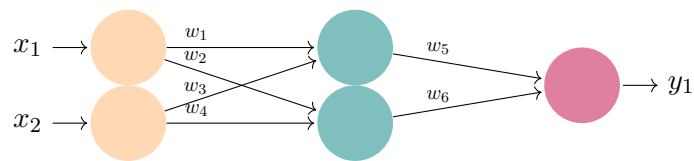
Techniques of AI [INFO-H-410]

v1.0.0

Source files, code templates and corrections related to practical sessions can be found on the UV or on github (<https://github.com/iridia-ulb/INFOH410>).

Feed forward neural network

Question 1. Consider the following fully connected feed forward NN:



- Give the equation for the output value of the network, given the network is linear.
- Show that the output equation of the perceptron is the same as the one of the neural net shown above.
- Now add an activation function f to each neuron and rewrite the first equation.
- Using the following inputs and weights, compute the output of the network using the sigmoid activation function first, and then the ReLU activation function.

inputs		weights					
x_1	x_2	w_1	w_2	w_3	w_4	w_5	w_6
1	1.5	1	2	3	2	1	3

Question 2. We will now implement, create, and train a 4 layer fully connected feed forward neural network to solve a classification problem. The dataset for this problem will be generated automatically by the `create_dataset` function in `utils.py`

- Using the provided templates, implement a simple neural network. Start by filling the `nn_template.py` to implement the forward pass, the backpropagation, and the training function.
- Train your network using the automatically generated dataset, use the template `ex2_template.py`
- Observe the effect when changing some of the hyperparameters

Question 3. Use tensorflow with keras to instantiate, train and test the same network as in the previous question (use `ex3_template.py`).