

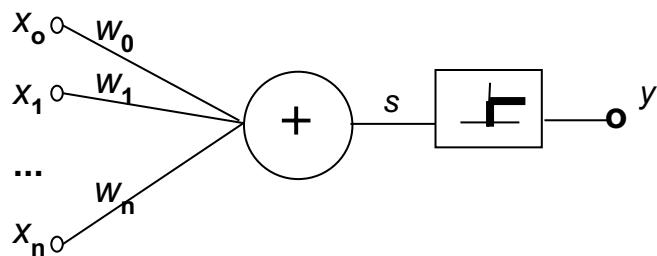
Lab 1: Perceptrons**Working group assignment**

Submit your code and report (within a zip file) to
<http://deei-mooshak.ualg.pt/~jvo/DL/submissions/>
as TP1

up to: February 18, 2026

Task

Consider a two layers network where each unit/neuron is a perceptron as the following:



Where $x_0=1$, $x_1, \dots, x_j, \dots, x_n$ are the inputs, and $w_0, w_1, \dots, w_j, \dots, w_n$ are their respective weights.

1. Considering a three perceptron 2:1 network, *compute*, without training, the weights of each neuron for obtaining a zero error in the NOT XOR binary function. Briefly explain.
2. Implement and test the above network in Jupyter notebook.

New to Python or Jupyter notebook? Follow these steps

1. Installing Python and Jupyter Notebook

We will need to install Python, some packages, and Jupyter Notebook:

1) Install the latest version of Python, suitable for your operating system, from:

<https://www.python.org/downloads/>

2) Install a virtual environment. This is optional for some operating systems. However, for some Linux distributions, it is required:

<https://pypi.org/project/virtualenv/>

3) Go to the command line and use pip to install the following Python packages to support numeric and ML development:

```
pip3 install --user numpy
pip3 install --user matplotlib
pip3 install --user plotly
pip3 install --user pandas
pip3 install --user keras
```

Note the --user switch to install in the user home directory. If willing to install system wide, just omit this switch.

4) Go to the command line and use pip to install the Jupyter Notebook and Jupyter Lab:

```
pip3 install --user notebook
pip3 install --user jupyterlab
```

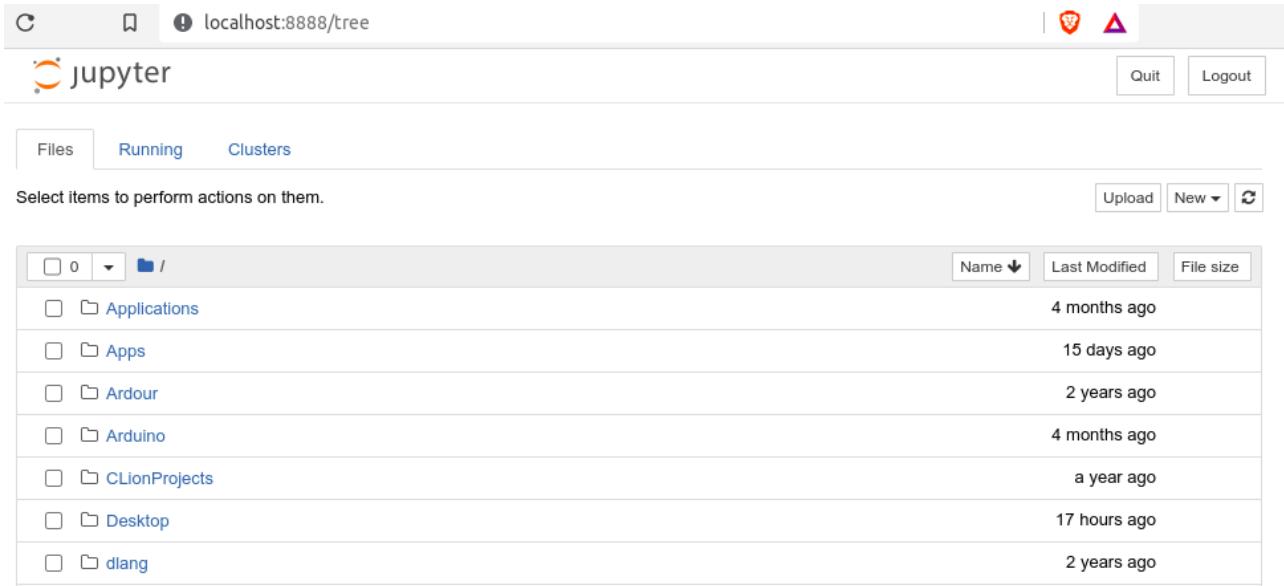
2. Using Jupyter Notebook

In the last step we install Jupyter Notebook and Jupyter lab. The latter is a newer version of the notebook. We will learn how to work with the Notebook first since the basics are the same.

Go to the command line and type:

```
jupyter notebook
```

The Jupyter Notebook opens in your browser, showing a view of the current folder. From there you can open a file or create a new one by pressing the new button.



The screenshot shows a Jupyter Notebook interface. At the top, there are tabs for 'Files', 'Running', and 'Clusters'. Below the tabs, a message says 'Select items to perform actions on them.' There are buttons for 'Upload' and 'New'. A file list table is shown with columns for 'Name', 'Last Modified', and 'File size'. The table contains the following data:

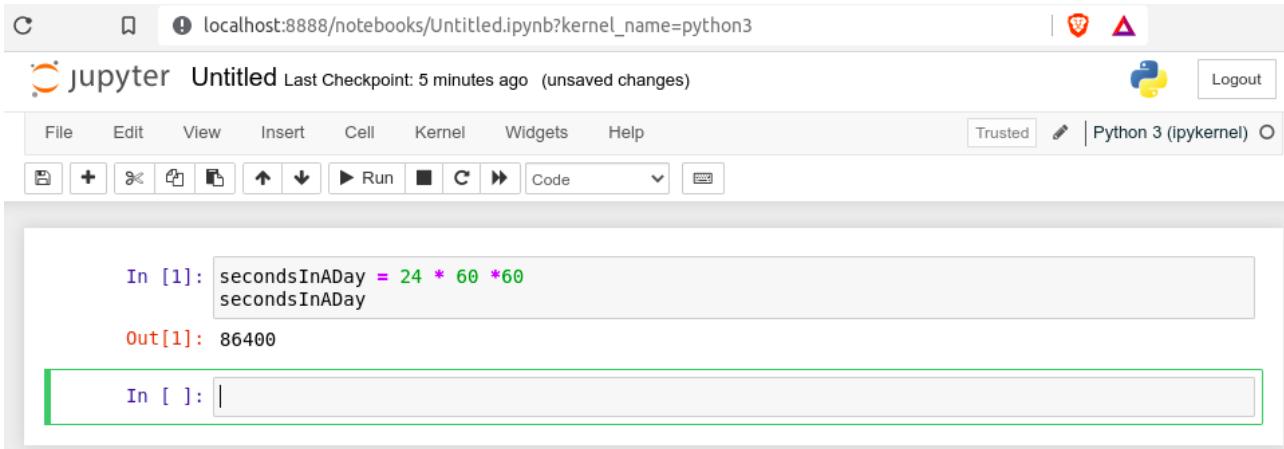
	Name	Last Modified	File size
<input type="checkbox"/>	0	4 months ago	
<input type="checkbox"/>	Applications	4 months ago	
<input type="checkbox"/>	Apps	15 days ago	
<input type="checkbox"/>	Ardour	2 years ago	
<input type="checkbox"/>	Arduino	4 months ago	
<input type="checkbox"/>	CLionProjects	a year ago	
<input type="checkbox"/>	Desktop	17 hours ago	
<input type="checkbox"/>	dlang	2 years ago	

Press the new button to create a new file, and choose Python 3 (ipykernel).

The basic execution unit is a cell. We can type Python code in a cell and then execute this cell only. We can also execute the code in all the notebook cells at once. Move the mouse to the cell, and type the following code:

```
secondsInADay = 24 * 60 *60
secondsInADay
```

Now go to the **run** button and press it. The cell code is executed. The output of the cell is printed below the cell. It prints the value of the variable in the last line in the cell:



The screenshot shows a Jupyter Notebook interface with a code cell. The cell contains the following code:

```
In [1]: secondsInADay = 24 * 60 *60
secondsInADay
```

The output of the cell is:

```
Out[1]: 86400
```

The cell is currently in 'Code' mode, indicated by the dropdown menu.

The file name is in the upper left corner, with the default name **Untitled**. By clicking on this name we can change it. We can save, load or create a new file from the file menu.

Now follow the Real Python tutorial below to learn the basic notebook operations:

<https://realpython.com/jupyter-notebook-introduction>

After completing the tutorial above, follow the one below. This second tutorial from Google shows more basic operations and also some oriented to scientific computation and chart plotting:

https://colab.research.google.com/github/bebi103a/bebi103a.github.io/blob/master/lessons/00/intro_to_jupyterlab.ipynb

Note that this tutorial is implemented as a Jupyter Notebook itself. When the cells are run, they are NOT executed locally but in the Google Colab cloud.

We will not need to use the Google Colab for now, but we can use it to run this tutorial. Alternatively, we can go to File → Download and save the notebook to our local drive to run it locally. We can choose to download in 2 formats:

intro_to_jupyterlab.py Plain Python runnable with Python interpreter

intro_to_jupyterlab.ipynb Jupyter Notebook runnable version