LAB Artificial Intelligence

Programming Assignment 7 – Artificial Neural Networks

You are required to submit your solutions of this assignment on Moodle, during the next lab time. Submit your own work. Cheating will not be tolerated and will be penalized.

Artificial Neural Networks

Neural networks are one of the most popular and effective forms of learning systems.

A neural network is a collection of nodes or units connected by directed links. A link from unit `i` to unit `j` serves to propagate the activation a_i from `i` to `j`. Each link activation also has a numeric weight $w_{i,j}$ associated with it, which determines the strength and sign of the weight of the connection.
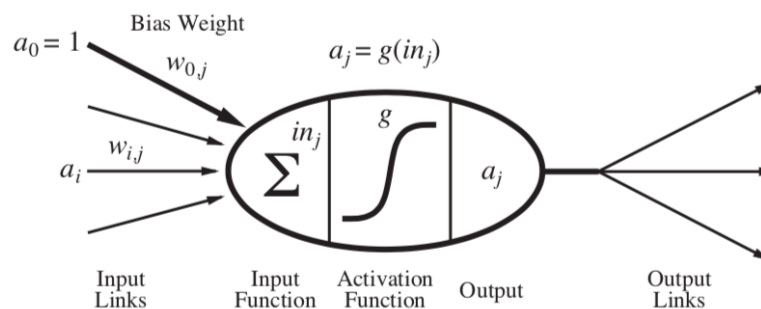

Figure 1-A simple mathematical model for a neuron.

Several types of artificial neural networks exist. The most common are:

- A **feed-forward network** has connections only in one direction, there are no loops. A feed-forward network represents a function of its current input; thus, it has no internal state other than the weights themselves.

  Feed-forward networks are usually arranged in **layers**, such that each unit receives input only from units in the immediately preceding layer. In a multilayer network, one or more layers of **hidden units** that are not connected to the outputs of the network might exist.

- A **recurrent network**, on the other hand, feeds its outputs back into its own inputs. Signals travel in both directions by introducing loops in the network. Computations derived from earlier input are fed back into the network, which gives them a kind of memory.

- A **convolutional network** consists of an input and an output layer, as well as multiple hidden layers. The hidden layers of a convolutional network typically consist of a series of convolutional layers that convolve with a multiplication or other dot product

In this lab, we are going to model a simple feed-forward Artificial Neural Network (ANN), also known as Multi-Layer Perceptron (MLP).

Keras: The Python Deep Learning library

Keras is a high-level neural networks API, written in Python and capable of running on top of TensorFlow, CNTK, or Theano. It was developed with a focus on enabling fast experimentation. Being able to go from idea to result with the least possible delay is key to doing good research.

Keras advantages:

- Allows for easy and fast prototyping (through user friendliness, modularity, and extensibility).
- Supports both convolutional networks and recurrent networks, as well as combinations of the two.
- Runs seamlessly on CPU and GPU.

The core data structure of Keras is a model, a way to organize layers. The simplest type of model is the **Sequential** model, a linear stack of layers. Stacking layers into your NN is as easy as `.add()`

In order to install Keras, and TensorFlow, refer to the pdf document in your folder entitled "Keras installation instruction.pdf"

The Bank customers problem

In this lab, we are going to predict using ANN if a bank's customer is expected to leave the bank or not.

Open the file ANN.ipynb. and complete the missing parts of the code. All the steps needed to solve this problem are given to you in the file.

The dataset you are going to work with is the Bank_customers.csv. Inside this file you have several columns. You should check what every column represents and see if it is a relevant information to be used as a feature or not.

The lab is split into several parts. You are required to finalize Part 1 in this first lab session, and continue the remaining parts in the next lab.

At home, install keras and tensorflow on your machine and import them into your file to make sure they are properly installed.