

Practical Session 2 – Neural Networks for regression and classification

The objective of this tutorial is to implement a neural network model in Python. Two versions will be developed: one for a regression case and the other for a classification case. Experiments will then be conducted to evaluate the performance of the models on test datasets.

1. Presentation of the provided elements

A Python file (*nn_regression.py*) is provided, containing functions that need to be completed to make the program operational for the case of regression.

To test the developed functions, four data files are provided:

- **food_truck.txt**: contains a dataset characterized by a single predictive variable (the population size of cities, in the first column). The target variable (second column) corresponds to the profit of a food truck in that city. The problem associated with this dataset is to predict the profit a food truck could make based on the size of the city.
- **houses.txt**: contains a dataset characterized by two predictive variables (the area of a house and the number of rooms, in the first two columns). The target variable (third column) corresponds to the price of the house. The problem associated with this dataset is to predict the price of a house based on its area and the number of rooms.
- **scores.txt** contains a dataset characterized by two predictor variables (the students' scores on two exams) (the first two columns). The target variable (third column) indicates whether the student is admitted (1) or not (0) to the University. The problem associated with this dataset is to predict whether a student will be admitted to the University based on the scores obtained in the two exams.
- **iris.txt** contains a famous dataset consisting of 150 samples corresponding to plants of the "Iris" type. These samples are characterized by 4 predictive variables (sepal length, sepal width, petal length, petal width). Each sample belongs to one of three Iris families: Setosa (code 0), Versicolour (code 1), Virginica (code 2).

2. Neural Network for a regression problem

Complete the functions in the *nn_regression.py* file to make the program operational. It is recommended to follow the order of the functions indicated in the previous section and to test them at each step.

food_truck.txt, houses.txt et scores.txt files can be used to test the program.

3. Neural Network for a classification problem

Inspired by the program in the file *nn_regression.py*, write a new program in a file named *nn_classification.py* to implement a neural network for classification. As seen in the course, a typical solution for classification is to use the softmax activation function for the network's final layer and a cross-entropy loss function.

To test the program, scores.txt and iris.txt files can be used.

4. Further experiments

The two neural networks models for regression and classification must be tested on various datasets of your choice (for example, from: <https://archive.ics.uci.edu/>) and with different configurations (number of hidden layers, number of neurons per hidden layer, types of activation functions, etc.) to thoroughly study the behavior of the networks.