Lab 6-8 (Project 1.1) – Classification using NN (Weeks 3.2-4)

1 – Introduction

During the COVID-19 pandemic, a restriction on the maximum amount of people that could be simultaneously inside a room, was imposed by Técnico Lisboa. This capacity depended on several factors, including the room dimension, ventilation, etc. The need to automatically detect the number of persons inside a lab without affecting privacy, led to the implementation of an experimental lab based on low-cost, non-intrusive sensors.

The lab consists of a 13m^2 room where a Zigbee based wireless sensor network was installed. The lab has three workstations (a chair, and a desk with a dock station and a table lamp). There is a small window above workstation 3 and there is no heating/ventilation/AC system active in the room.

The wireless network is a Zigbee-based star network with six slave nodes feeding data to the master node. There is one CO2 sensor (MH-Z14A) in the center of the room, two digital infrared motion sensors (PIR) in opposed walls, and, in each workstation, a node containing a light sensor (BH1750) and a temperature sensor (LMT84LP) has been installed.

PIR sensor data indicates if movement was detected during the last 30s. For the remaining sensor nodes, the Arduino Uno microcontroller board sampled data from the sensors and transmitted it periodically via a Zigbee module every 30s.

Sensor measurements were taken over a period of several days. Each student manually annotated when entered and left the room during this period. Therefore, true occupancy was annotated during the measurement period.

The resulting dataset has now been made available (Lab6Dataset.csv).

2 – Objectives

During the worse times of the pandemic, Técnico imposed a limit of 2 persons inside the above lab. However, the students that use the lab had frequent deadlines, and often ignored the 2-person limit.

The objectives of this project are to develop a NN-based classifier that, using the dataset, is able to:

a) Detect when there are more than 2 persons inside the lab;
b) Detect how many people are inside the lab.

2.1 – Submission details and Deadline

This Lab will be evaluated as the first part of project 1, and accounts for 45% of the Lab final grade. The final code and a comprehensive report must be submitted via Fenix until **Monday, June 6th, at 23:59**.

The students must submit:

a) All the developed code used to train and create the models;

b) A piece of code that will allow me to test your multiclass model using unseen data. This code, called TestMe, accepts as a parameter the name of a .csv file that has the same structure of Lab6Dataset.csv. The code must test the model you created on this new data (the program cannot "fit" the model to this new data, only "predict" it). The output of the code must be the confusion matrix of the occupancy (0 person; 1 person; 2 persons, 3 persons), the Precision and Recall of each class, and the overall Macro-Precision, Macro-Recall and Macro-F1;

c) A report where you indicate the options you made regarding the data preparation, the experimental setup, the construction of the model(s), the evaluation and validation process, and the results you obtained. Remember to take note of all decisions you make while checking and preparing the data, deciding the hiperparameters, etc., since it will be useful for the report.

3 – Dataset

The "Lab6dataset.csv" file is composed of 10129 records, taken between and 11/01/2021 at 10:53, and 16/01/2021 at 9:04, approximately every 30s (some data points are missing). Each record contains the date/time, the data collected from the 9 sensors and the number of persons in the room.

Note that the dataset has not been preprocessed – it might contain noise, outliers, missing inputs, redundant features, etc.

The fields of the data set are:

- Date
- Time
- Si_Temp (^{o}C, float)
- S_i_Light (Lux, int))
- CO$_2$ (PPM, int)
- PIR$_i$ (Boolean)
- Number of persons in the room (0-3, int)

4 – Implementation, Evaluation and Validation

Use all the knowledge you have acquired so far regarding data preparation, experimental setup and Neural Networks, to see how well you can solve this problem and build a good model (please re-check the theory slides regarding data and experimental setup before starting the project).

It is up to you to analyze the problem, look at the data, clean it, see which features might or might not be helpful (you can obviously try to use them all), check if new features might be useful, decide if the order and/or the date/time is relevant, etc.

The dataset is imbalanced, so you might have to find ways to deal with this issue.

Note that objective a) is a binary classification problem, but objective b) is a multi-class classification problem. Don't forget how to properly validate and evaluate each goal.

Regarding the Neural Network, it's up to you to decide (or simply try) different architectures and the respective hyperparameters. I advise you to start with a "standard" MLP, using the most usual activation function (logistic), the most usual solver (stochastic gradient descent – sgd), not using regularization, etc. Note that these ARE NOT the default parameters for NN in Scikit, so you must indicate them explicitly.

Try to avoid OVERFITTING, since later I will check how your system performs under previously unseen data. The best way to know if there is overfitting or not is to use a Train/Validation/Test split, and only use the Test set once your model is completely defined (input features, number of layers, number of neurons per layer, activation functions, etc.). I.e, use a Holdout set! If after checking the results on the Test Set, you change any parameter or hyperparameter (even if it's only changing the number of neurons in a hidden layer), then you are accidentally fitting the model to the Test set (which might result in overfit).