TÉCNICO LISBOA

Intelligent Systems

Master Degree in Mechanical Engineering

Assignment 2 [EN]

Authors:

Henrique Branquinho (ISTID 187203) henrique.branquinho@tecnico.ulisboa.pt
Diogo Dias (ISTID 196705) diogohbdias@tecnico.ulisboa.pt

Group 7

2024/2025 – 1st Semester, P1

Contents

1 Exercise 1 (NN)

The initial task of this assignment involved developing a neural network (NN) in Python utilizing the scikit-learn library.

1.1 Hair Dryer Dataset

1.1.1 Model

The first neural network developed was applied to the dataset illustrated in the following figure:

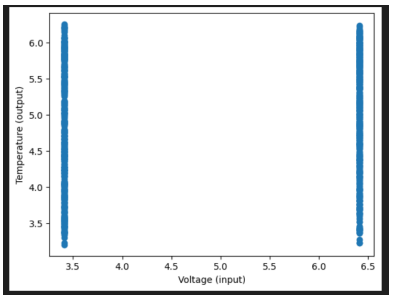

Figure 1: Hair Dryer Dataset

The developed neural network is a Single Input Single Output (SISO) model, characterized by this specifications:

Number of Hidden Layers	2
Neurons per Layer	20

Given that this is a straightforward regression task, we anticipated that the neural network would not require an excessive number of layers or neurons, so we opted for a simple architecture.

1.1.2 Results

The model's performance was evaluated using the Root Mean Squared Error (RMSE) as the evaluation metric. The results obtained are as follows:

Train RMSE	0.831
Test RMSE	0.779

It is evident that the NN struggled to accurately model the behavior of the hair dryer. This can be attributed to the randomness in the dataset, as shown in Figure 1, where each input corresponds to multiple outputs, leading to inherent inconsistencies in the data. As a result, the model's performance remained suboptimal. However, a positive outcome is that the model did not exhibit overfitting.

1.2 Breast Cancer Dataset

1.2.1 Model

In this part of the assignment, we are addressing a multiple input, single output (MISO) binary model, which presents a significantly more complex challenge compared to the previous dataset. Consequently, the new neural network has the following characteristics:

Number of Hidden Layers	3
Neurons per Layer	31

1.2.2 Results

Given that this problem involved binary classification on an unbalanced dataset, we evaluated the model's performance using Sensitivity (True Positive Rate) as a metric.

Train Sensitivity	1
Test Sensitivity	0.925

As indicated by the values in the table, this model demonstrated good performance, however, it may have slightly overfit.

2 Exercise 2 (ANFIS)

In this section of the assignment, we were required to train an Adaptive Neuro-Fuzzy Inference System (ANFIS) using MATLAB.

2.1 Hair Dryer Dataset

2.1.1 Model

Firstly we developed, once again, an SISO system to model the hair dryer behaviour. The characteristic of the model can be observed in Figure 2.

```
ANFIS info:
    Number of nodes: 24
    Number of linear parameters: 10
    Number of nonlinear parameters: 10
    Total number of parameters: 20
    Number of training data pairs: 770
    Number of checking data pairs: 0
    Number of fuzzy rules: 5
```

Figure 2: First ANFIS developed

2.1.2 Results

To evaluate the performance of the model, we selected Root Mean Squared Error as the evaluation metric. The results obtained are as follows:

Initial Test RMSE	0.778
Minimal Training RMSE	0.833
Final Test RMSE	0.778

It is clear that this model also struggles to adequately model the stochastic hair dryer dataset, as the RMSE remains relatively high and did not improve after 10 epochs of training. However, this model did not exhibit overfitting and achieved slightly better results than the neural network.

2.2 Breast Cancer Dataset

2.2.1 Model

A second ANFIS model was developed to address the binary classification problem using the Breast Cancer Dataset. The characteristics of this model are shown in Figure 3.

```
ANFIS info:
    Number of nodes: 112
    Number of linear parameters: 50
    Number of nonlinear parameters: 90
    Total number of parameters: 140
    Number of training data pairs: 547
    Number of checking data pairs: 0
    Number of fuzzy rules: 5
```

Figure 3: Second ANFIS developed

2.2.2 Results

For this problem, performance analysis was conducted using Sensitivity and Specificity (True Negative Rate) as key metrics.

The performance of this second model was significantly more favorable, as shown in the following table.

Initial Test Sensitivity	0.978
Initial Test Specificity	0.936
Final Test Sensitivity	0.966
Final Test Specificity	0.979

Once again, even after training for 10 epochs, the model did not show significant improvement. However, this time the initial performance was already quite satisfactory (even better than the neural network), making further enhancement challenging.

Although the final results are promising, there are some observations to note. Compared to the first iteration, the ANFIS model exhibited an increase in over-diagnosis of cancer, as indicated by the decrease in Sensitivity and the improvement in Specificity. In other words, while fewer actual cancer patients were misdiagnosed as healthy, there were more patients incorrectly diagnosed with cancer who did not have the disease.

3 GitHub Link

You can find the code on GitHub at:
https://github.com/diogohbd/SInt/tree/master/Assignments/Assignment%202