Intelligent Systems

Assignment 1

Bernardo Morais Chagas (103639)

September 25, 2025

Repository link: **GitHub Repository**

1 Introduction

Fuzzy systems are computational models that allow variables to take values between completely true and completely false state. Instead of working with rigid categories, degrees of membership are used to represent uncertainty and imprecision, with resource to a set of rules. This technique provides a balance between interpretability and adaptability, making fuzzy systems widely applicable across different fields.

In practice, with this assignment, both classification and regression tasks were solved using fuzzy systems methodology supported by the TSK_pytorch.inbpy code provided. Between the several algorithms possible to use, the Takagi-Sugeno-kang (TSK) was the one choosen. This methodology allows the mapping of the clusters into The general approach can be defined in 8 steps:

1. Determine relevant input and output features and collect data
2. Preprocess data (clean, impute missing values, normalize/scale)
3. Select the fuzzy system structure (Mamdani, Takagi–Sugeno, etc.)
4. Choose the number of clusters and clustering algorithm
5. Cluster the data and inspect membership strengths
6. Define antecedent membership functions (MFs) from clusters
7. Obtain consequents (fuzzy sets or parameters) for each rule
8. Simplify and validate the model using clustering and task metrics

Dataset 1: Diabetes Dataset (Regression)

For the task 1, a given dataset "datasets.load_diabetes(as_frame=True)" with 442 samples was used to predict the progression of the diabetes (target) based on the 10 features analysis.

After importing the dataset and create some code cells to allow visualization of the data, was possible to start tuning the parameters. For this approach, the number of clusters and the exponentiation parameter, m, were changed.

It was verified that a number of cluster bigger than 5, would increase the model complexity, without improving the model performance.

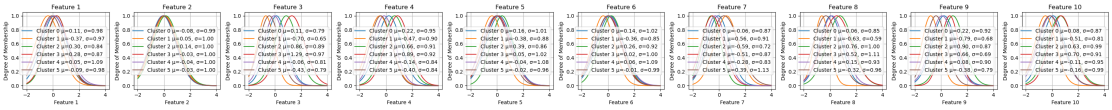

Figure 1: Degree of Membership for number of clusters = 6

By analysing the plot, it was possible to verify that a large number of clusters ended up overlapping each other, not being beneficial.

After setting a number of clusters up to 5, the value of m was discretized into:

- $m = 1.1$: almost hard clustering, giving sharp cluster boundaries.
- $m = 1.5$: moderate fuzziness, balancing smoothness and specificity.
- $m = 2$: standard fuzzy c-means, with smooth membership degrees.

Several simulations were performed using the mean squared error (MSE) as the performance indicator. The following table was constructed to identify the parameters that yield the minimal error:

Table 1: MSE values for different numbers of clusters and m values

m	2 clusters	3 clusters	4 clusters	5 clusters
1.1	2522	2972	2476	2692
1.5	2534	2929	2484	2726
2.0	2545	2933	2528	2705

With this analysis it was possible to verify that the the smaller overall MSE was improved when it was considered a number of 4 clusters. For a given number

of clusters, a lower value of m presented better results as well (with exception of m = 1.1 for 3 clusters). It was concluded that, by analysing the MSE, a 4 clusters segmentation with a value of $m = 1.1$ would be the best model.

It was also possible to verify that the model struggles to predict the diabetes evolution value, given that all the MSE values were consedered high, given the range values of the target. Th following chart was built to better visualize the flotation of values:

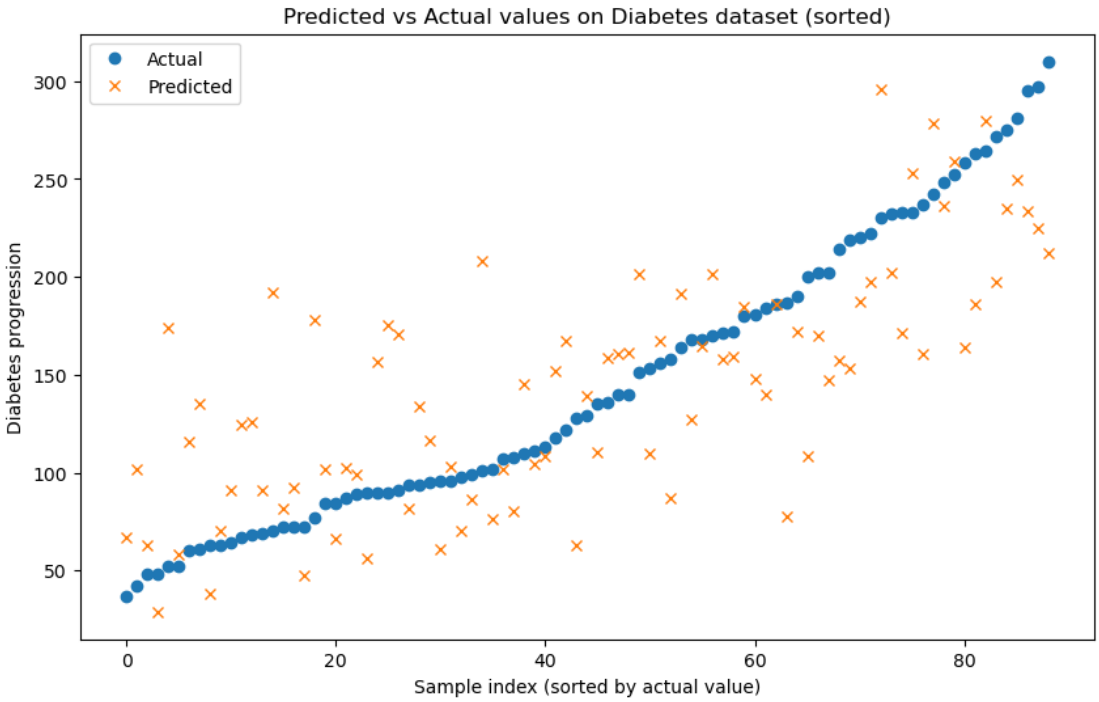

Figure 2: Best Model Predicted vs Actual values on Diabetes dataset (sorted)

The tested target was sorted allowing a better visualization of the error distribution. As it is possible to see, the predicted values seems to follow the tendency (higher the actual value, higher the predicted one), with the addition of a large noise band. The band with tend to decrease for higher values of the target.

In resume, the built model can't be considered good given that the predicted result is presented with a large error.

Task 2

For the task 2, a given dataset "fetch_openml("diabetes", version = 1, as_frame=True)" with 768 samples was used to predict the progression of the diabetes (target) based on the 8 features analysis.

After importing the dataset and create some code cells to allow visualization of the data, was possible to start tuning the parameters. For this approach, the number of clusters and the exponentiation parameter, m, were changed, as well as the threshold value used do classify the results as positive or negative.

It was verified that a number of cluster bigger than 4, would increase the model complexity, without improving the model performance.

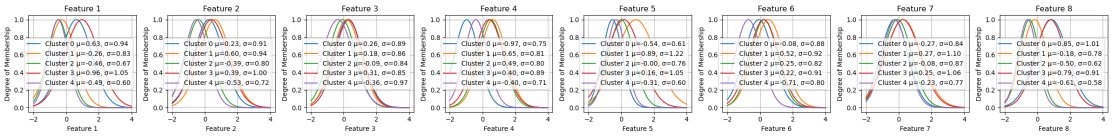

Figure 3: Degree of Membership for number of clusters = 4

Considering a maximum number of 4 clusters as well as a discretized value of $m = 1.1, 1.5, 2$, the accuracy value and the confusion matrix were analysed. Given the difficulty of setting the threshold, a new chart was built to allow the visualize the distribution of the real target vs predicted one, identifying the separation of the data:

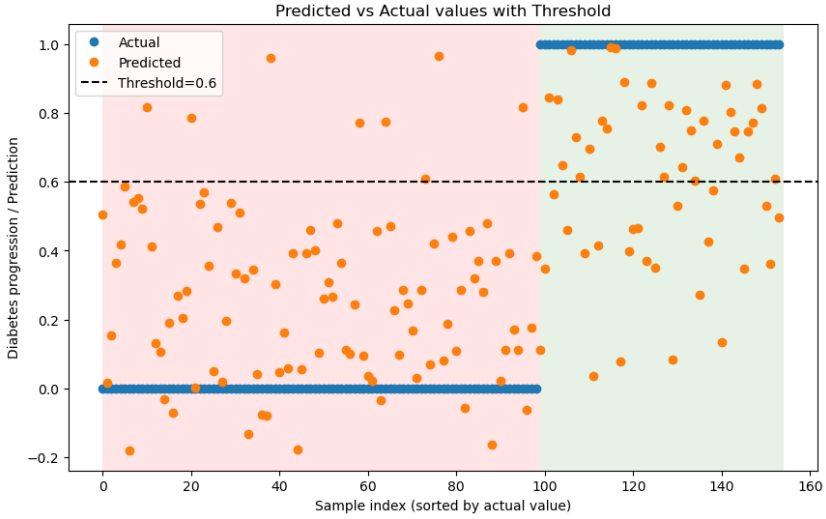

Figure 4: Best Model Predicted vs Actual values with Threshold

Consecutive iterations of tuning the number of clusters and the exponentiation, m, and then the threshold in order to reduce the number of FP and FN allowed to find the best model (Figure 4). The best model was achieved for 4 clusters, a value of $m = 1.1$, and a threshold, thr, $= 0.6$, presenting an accuracy of 80%.

From the chart, as well as from the correlation matrix, is was possible to see that there are a lot of FN (people that have diabetes that would not be correctly classified in this model). Given the problem context, it is more important to reduce false negatives (FN) than false positives (FP). In other words, it is preferable to incorrectly classify someone as having the disease when they do not, rather than miss someone who actually has it. Therefore, lowering the decision threshold may be acceptable: this can increase sensitivity (true positive rate) at the cost of reducing precision (increasing FP).