DSC_540_T6_ASS/2021

Topic 6 Assignment: Fuzzy Logic Models

This assignment mixes theory and application, in the form of several challenges and problems. Perform the tasks described in each. Note that this (and other) assignments include a few challenging research-related tasks. They are aimed at gradually building your capacity to tackle complex topics, familiarize yourself with academic discourse, and provide context and practice for the skills you will eventually need when working on your capstone thesis or project.

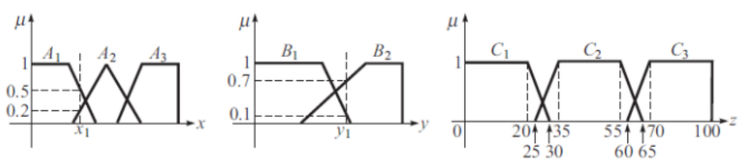
APA style is expected, as well as formal and rigorous scientific writing, using appropriate mathematical notation and references.

Part 1 - Theory

Task 1. Consider the following fuzzy model of a system with inputs x and y and output z:

Rule 1: If x is A_3 OR y is B_1 THEN z is C_1
Rule 2: If x is A_2 AND y is B_2 THEN z is C_2
Rule 3: If x is A_1 THEN z is C_3

The membership functions of the input and output variables are given in the graphs below:



Actual inputs are x_1 and y_1. Find the output z by applying standard fuzzy operation:

min for AND
max for OR.

Show and explain all your steps.

Rule 1:

If x is A3 OR y is B1 then z is C1
ie: If project funding is adequate OR project-staffing is small
then the risk is low

Rule 2:

If x is A2 AND y is B2 then z is C2

ie: project-funding is marginal AND project-staffing is large THEN risk is normal

Rule 3:
If x is A1 THEN z is C3

Ie: if project funding is inadequate THEN the risk is high

Step 1: Purification
Take the crisp input, x1, and y1 (project funding and project staffing
Determine the degree to which these inputs belong to each of the appropriate Fuzzy sets

$\mu(x = A_2) = 0.2$ $\mu(y = B_2) = 0.7$

project funding project staffing

Take the fuzzified inputs $\mu(x = A_1) = 0.5$
$\mu(x = A_2) = 0.2$, $\mu(y = B_1) = 0.1$
$\mu(y = B_2) = 0.7$
apply them to antecedents of the fuzzy rules.

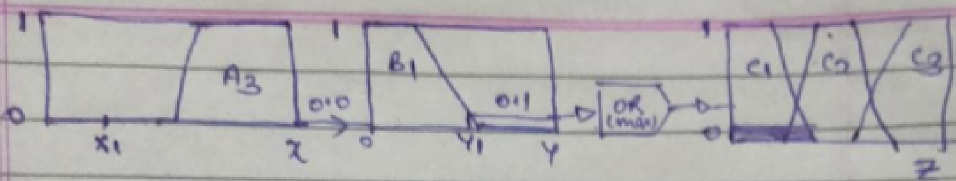
- If a given fuzzy rule has multiple antecedents, the fuzzy operator (AND or OR) is used to obtain a single number that reprsents the result of antecedent evaluation. This number (the truth value) is then applied to the consequent membership function.

To evaluate the disjunction of rule antecedents we use the OR fuzzy operation.

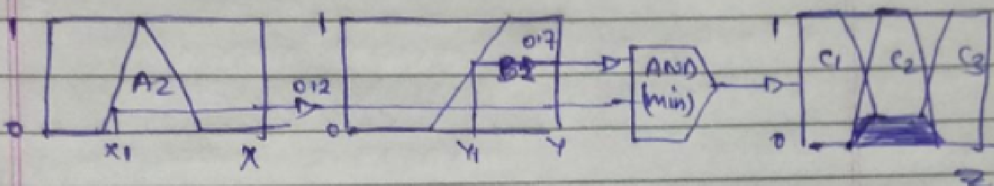
$$\mu_A \cup \mu_B (x) = \max[\mu_A(x), \mu_B(x)]$$

Similarly, in order to evaluate the conjunction of rule antecedents, we apply the AND fuzzy operation intersection.

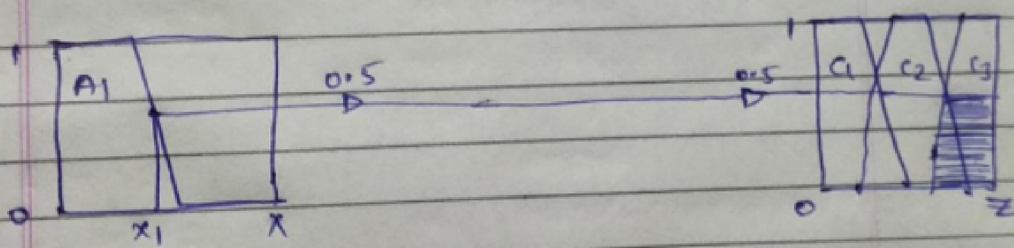
$$\mu_A \cap \mu_B(x) = \min[\mu_A(x), \mu_B(x)]$$



Rule (1) If x is A_3 (0.0) OR y is B_1 (0.1) THEN z is C_1 (0.1)



Rule (2) If x is A_2 (0.2) AND Y is B_2 (0.7) THEN z is C_2 (0.2)



Rule (3) If x is A_1 (0.5) THEN z is C_3 (0.5)

Task 2. Read "Solving the Ocean Color Inverse Problems by Using Evolutionary Multi-Objective Optimization of Neuro-Fuzzy Systems" from your topic resources.

a. Explain in a short paragraph (10-15 sentences) the application/use of the Takagi-Sugeno fuzzy model in the article.
b. Explain in a short paragraph (10-15 sentences) the application/use of ANFIS.

a. Explain the application use of the Takagi-Sugeno fuzzy model in the article:

The ocean color inverse problem consists of determining the concentrations of optically active constituents, such as chlorophyll, suspended particulate matter, and colored dissolved organic matter, from remotely sensed multispectral measurements of the reflected sunlight back-scattered by the water body. In this paper, we approach this regression problem by using an evolutionary multi-objective algorithm, namely the (2+2) Modified Pareto Archived Evolutionary Strategy ((2+2)M-PAES), to optimize Takagi-Sugeno type (TS-type) fuzzy rule-based systems (FRBSs). Accuracy and complexity are the two competing objectives to be simultaneously optimized. TS-type FRBSs are implemented as an artificial neural network; by training the neural network, the parameters of the fuzzy model are adjusted. In this way, the evolutionary optimization coarsely identifies the structure of the TS-type FRBSs, while the corresponding neural networks finely tune their parameters. As a result, a set of TS-type FRBSs with different trade-offs between accuracy and complexity is provided at the end of the optimization process. We show the effectiveness of our approach by comparing our results with those obtained on the ocean color inverse problem by other techniques recently proposed in the literature.

b. Explaining the application used of ANFIS model from the article:

ANFIS is an artificial neural network that embeds a TS-type FRBS. The neurons of the first layer of an ANFIS compute the membership degrees Af,t(xf), f = 1,...,F, t = 1,...,TF , of each component xf of an input vector to each fuzzy set Af,t defined over universe Uf of variable Xf .

The first layer has therefore F f=1 Tf neurons. The neurons of the second layer model the antecedents of the TS-type rules and compute the activation degrees wm(x). The neurons of the third layer normalize the activation degrees by computing the vm(x) values. The neurons of the fourth layer compute the weighted output vm(x) · ym(x) associated with the output of each rule ym(x). The single neuron of the fifth layer computes the output of ANFIS by summing up the weighted outputs provided by each rule. The classical approach to the generation of an ANFIS requires, first of all, the choice of the number Tf of fuzzy sets which partition each input linguistic variable Xf (note that the number of fuzzy sets can be different from an input variable to another) [27]. Second, a uniform partition Pf = {Af,1, . . ., Af,Tf }, f = 1,...,F, for each variable Xf is generated. Third, a rule in the form Eq. (1) is created for each possible combination of the fuzzy sets in the F partitions (grid partitioning). The number of these rules is Mgrid = F f=1 Tf, which is also the number of neurons on the second, third and fourth layers of ANFIS. Figure 1 shows an example of a classical ANFIS built using this appro

It has been approached the ocean color inverse problem by using a Pareto-based multi-objective evolutionary algorithm, namely, the (2+2)M-PAES, so as to generate a set of TS-type FRBSs with different trade-offs between accuracy and complexity. The TStype FRBSs are embedded in an ANFIS. We have employed the (2+2)M-PAES to evolve the antecedent part of the TS-type FRBSs, while we have taken advantage of the learning algorithm of ANFIS to estimate the consequent parameters and to tune the antecedent parameters. As a result, a set of TS-type FRBSs with different trade-offs between accuracy and complexity is provided at the end of the optimization process. We have shown the validity of the solutions generated by the (2+2)M-PAES by comparing the results obtained by these solutions with those achieved on the ocean color inverse problem by other techniques recently proposed in the literature. Finally, we have pointed out how our approach allows automatically performing a feature selection.

Part 2 – Fuzzy Models

Refer to the readings assigned for this topic and provide solutions to the following problems, using Jupyter notebooks. Include formal and detailed explanations to accompany the code.

Problem 1. Consider a two-dimensional *sinc* equation defined by:

$$y = \mathrm{sinc}(x_1, x_2) = \frac{\sin(x_1)\sin(x_2)}{x_1 x_2}$$

Training data are sampled uniformly from the input range [–10, 10] × [–10, 10]. With two
symmetric triangular membership functions assigned to each input variable, construct a
Takagi-Sugeno fuzzy model (linear static mappings) for the *sinc* function. Provide defining
equations for determination of the premise and consequent parameters of the model.

Problem 2. To identify the non-linear system

$$y = (1 + (x_1)^{0.5} + (x_2)^{-1} + (x_3)^{-1.5})^2$$

Assign two membership functions to each input variable. Training and testing data are
sampled uniformly from the input ranges:

Training data: [1, 6] × [1, 6] × [1, 6]
Testing data: [1.5, 5.5] × [1.5, 5.5] × [1.5, 5.5]

Extract *Takagi-Sugeno fuzzy rules* from the numerical input-output training data that could be employed in an *ANFIS* model.

List and explain all the rules.

Note: You may find "Fuzzy Interference System Implementation in Python" article from your topic resources useful for this assignment. The project described is not exactly like the one you are asked to address, but it

Problem1.

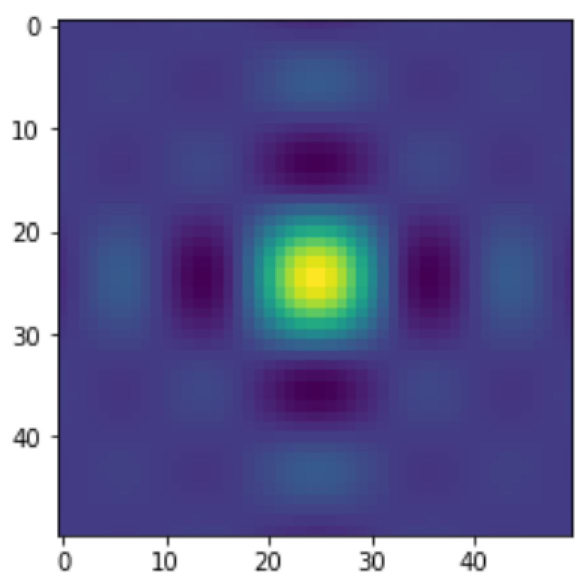
```python
# Solving a two-dimensional sinc function
```

```python
import numpy as np
import matplotlib.pyplot as plt

sinc2d = np.zeros((50, 50))
for x, x1 in enumerate(np.linspace(-10, 10, 50)):
    for y, x2 in enumerate(np.linspace(-10, 10, 50)):
        sinc2d[x,y] = np.sin(x1) * np.sin(x2) / (x1*x2)

# equivalently:
x1 = np.linspace(-10, 10, 50)
x2 = np.linspace(-10, 10, 50)
sinc2d = np.outer(np.sin(x1), np.sin(x2)) / np.outer(x1, x2)

plt.imshow(sinc2d)
plt.show()
```



Problem 2.

To identify the non-linear system y= (1 + (x1)0,5 + (x2)+(x3)-.5)2 Assign two membership functions to each input variable. Training and testing data are sampled uniformly from the input ranges: Training data: [1, 6] x [1, 6] x [1, 6] Testing data: [1.5, 5.5] x [1.5, 5.5] x [1.5, 5.5]

Extract Takagi-Sugeno fuzzy rulesfrom the numerical input-output training data that could be employed in anANFIS model.

From the input ranges [1.6]x[1.6]x[1.6], and [1.5,5.5]x[1.5,5.5]x[1.5,5.5] respectively.. Extract Suugeno Fuzzy rules from the numerical input-output training training data that could be employed in an ANFIS model.
Assume that a fuzzy inference system has two inputs x1 and x2 and one output y.
The rule base contains two Sugeno fuzzy rules as follows:

Rule 1: IF x_1 is $\underline{A}_{11}$ and x_2 is $\underline{A}_{21}$ THEN $y^{(1)} = a_0^{(1)} + a_1^{(1)} x_1 + a_2^{(2)} x_2$

Rule 2: IF x_1 is $\underline{A}_{12}$ and x_2 is $\underline{A}_{22}$ THEN $y^{(2)} = a_0^{(2)} + a_1^{(2)} x_1 + a_2^{(2)} x_2$

$\underline{A}_{ij}$ are GAUSSIAN functions.

For given input values x_1 and x_2 the inferred (output) output is calculated by

$$\bar{y} = \frac{\mu^{(1)} y^{(1)} + \mu^{(2)} y^{(2)}}{\mu^{(1)} + \mu^{(2)}}$$

where $\mu^{(i)}$, $r = 1.2$ are firing strengths of the two rules.

Product inference is used to calculate the firing strengths of the rules.

Developing ANFIS architecture for this modeling problem and deriving learning algorithm based on least squares estimation and the gradient descent methods.

REFERENCES:

https://towardsdatascience.com/fuzzy-inference-system-implementation-in-python