Assignment 6: Fuzzy Logic Models

Sophia J Quinton

Grand Canyon University

DSC-540: Machine Learning

Dr. Aiman Darwiche

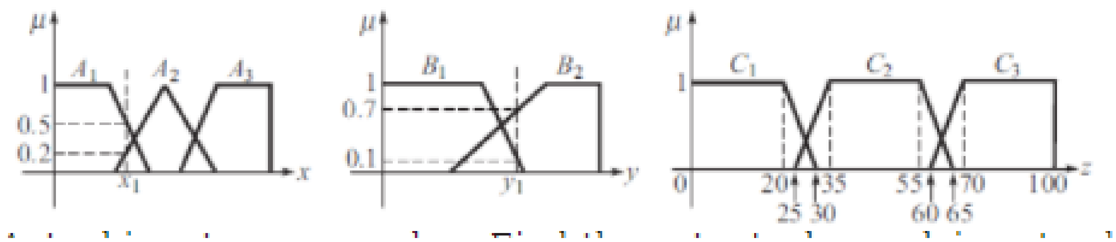
8 December 2021

Assignment 6: Fuzzy Logic Models

Part 1 – Theory

Task 1

$$\text{Rule - based: } \begin{cases} \text{If x is } A_3 \text{ OR y is } B_1 \text{ THEN z is } C_1 \\ \text{If x is } A_2 \text{ AND y is } B_2 \text{ THEN z is } C_2 \\ \text{If x is } A_1 \text{ THEN z is } C_3 \end{cases}$$

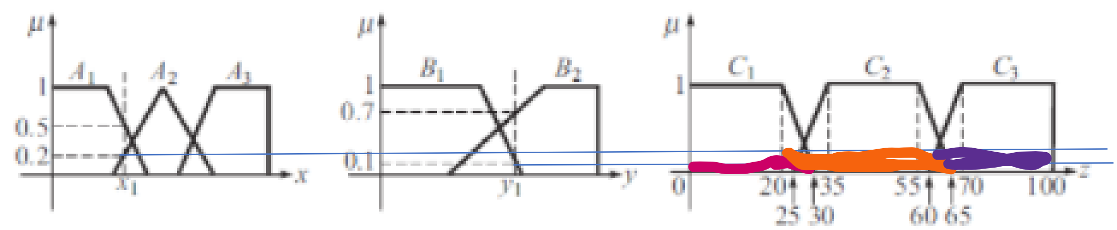


Actual inputs: x1 and y1

i) $\alpha^3 = \max[\mu_{A3}(X1), \mu_{B1}(y1)] = \max(0,0.1) = 0.1$

ii) $\alpha^2 = min[\mu_{A2}(X1), \mu_{B2}(y1)] = min(0.2,0.7) = 0.2$

iii) $\alpha^1 = \mu_{A1}(x1) = 0.2$



$$\sum yk\mu B'(yk) = 20 * 0.1 + 15 * 0.1 + 35 * 0.2 + 55 * 0.2 + 70 * 0.2 + 100 * 0.2$$
$$= 55.5$$

$$\sum uB'(yk) = 0.1 + 0.1 + 0.2 + 0.2 + 0.2 + 0.2 = 1$$

$$y' = 55.5/1 = 55.5$$

$$z = 55.1$$

I used the graph to help find the results. First, I followed the results in Gopal (2019) to obtain the alpha values using the min max inference method. Then I used the COG estimation technique. “In theory, the COG is calculated over a continuum of points in the aggregate output membership function uB’(y), but in practice, a reasonable estimate can be obtained by calculating it over a sample of points” (Gopal, 2019). The result is z = 55.1.

Task 2

The Takagi-Sugeno fuzzy model is used as the model to model the “relation between reflectance and optically active constituents” (Cococcioni et. al., 2008). In the case of the T-S model, the consequents are modeled from the least square estimation. The authors use the T-S type within their artificial neural network. The general idea behind this model is that the T-S type of fuzzy inference system is used to establish the membership function for each layer of the neural network. Layer one is the T-S model based on the fuzzy inference system, layer two measured by the activation degrees, layer three, normalizeds the degrees, layer four computes the weighted values, and then layer five is the results (Cococcioni et. al., 2008). These layers help to establish the weighted outputs that are normalized after the T-S system is established. Their fuzzy inference system uses an and statement. Each of the algorithms that the authors run is based on the T-S fuzzy inference system. It should be noted that the T-S fuzzy inference system is different from the Mamdani fuzzy system. The article uses the T-S type because it is shown that “the fuzzy model allows achieving good results” because “the method adopted to build the fuzzy model requires to determine a-priori the number of rules based on the distribution of data rather than on accuracy” (Cococcioni et. al., 2008). Here, the T-S model is best because it focusses on the accuracy whereas the Mamdani fuzzy system focusses on the distribution.

The ANFIS is the artificial neural network that is focused on the T-S model. It is briefly described in the previous paragraph. The ANFIS has several sets of layers that utilize the T-S model, normalizes it, weighs it, and then outputs to result. It is based on the number of clusters, the uniform partition, and fuzzy rules are generated which helps determine the number of neurons in the ANFIS (Cococcioni et. al., 2008). A general thought to the ANFIS is that it is just the T-S fuzzy inference system when several modification steps to help normalize and weigh the data. It is a consistent way to combine the fuzzy inference framework and the neural networks. “The relatively high number of inputs in the ocean color inverse problem makes the use of classical ANFIS awkward, because of the exponential explosion of the number of rules when using the grid partitioning of inputs” (Cococcioni et. al., 2008). The learning phase of the ANFIS uses the gradient descent version of the antecedent parameters, and the consequent parameters are based on the lease square method to build the model (Cococcioni et. al., 2008). This is based off of the parameters set by the T-S model. The T-S model provides the algorithm and membership functions that are modified and controlled by the ANFIS. The result is a more robust model that can handle the different colors of the ocean.

Part 2

Problem 1

For the sinc equation. Two different triangular membership functions were created for the range [-10, 10] using the scikit-fuzzy library from Warner et. al. (2019). These are shown. Next the Takagi-Sugeno fuzzy model was generated. The rule for this model is: {If x1 is in A1 and x2 is in A2 then $y = \mathrm{sinc}(x1, x2)$

This model is example was designed following the examples in Gopal (2019) and Princeton University (nd). From here, an example point was given as x1 = 2 and x2=4. Following the methods and defuzzy methods given by Topperly (2021), the result was -0.086. The process to getting this result is by using the membership functions to calculate the w and the output function, the sinc function, with the input values. The defuzzy method is the w value times the y value over the w value.

```
In [121]: #If input 1 = x and input 2 = y, then output is sinc function
          def sinc_2d(input1, input2):
              result = (np.sin(input1) * np.sin(input2))/(input1*input2)
              return result

In [122]: ##let input 1 be 2 and let input 3 be 4 (Topperly, 2021)
          in1 = membership1[1]
          in2 = membership2[3]
          w_value = min(in1, in2)
          y1 = sinc_2d(2, 4)
          defuzz = (w_value * y1) / w_value
          defuzz

Out[122]: -0.08601982019984428
```

Problem 2

This problem was a bit more complex because a model needed to be trained. I followed the research from Fuchs et. al. (2020) as they created a python library for the Takagi-Sugeno fuzzy model called pyFume. The gitlab for the library has an example to train with a fuzzy model. First, I created the x and y train and test data using the linear system given in the instructions. This y data will help to create consequent parameter in the analysis stage. The antecedent parameters are given as a gaussian array. The antecedent parameter is another name for the rules for the model. The model successfully was trained because after running the test data, the MSE was 1.546. The original MAE was 2.89.

```
In [150]: ae = AntecedentEstimator(x_train=x_train, partition_matrix=partition_matrix)
          antecedent_parameters = ae.determineMF()
          antecedent_parameters

Out[150]: [('gauss', array([5.64535237, 1.69097102])),
           ('gauss', array([1.8217247 , 1.88210544]))]
```

Figure 1: rules for this model

The rule: $\begin{cases} \text{if x1 is in A1 then y} = (1 + (x1)^{0.5} + (x2)^{-1} + (x3)^{-1.5})^2 \\ \quad \text{x2 is A2 then y} = (1 + (x1)^{0.5} + (x2)^{-1} + (x3)^{-1.5})^2 \\ \quad \text{x3 is A3 then y} = (1 + (x1)^{0.5} + (x2)^{-1} + (x3)^{-1.5})^2 \end{cases}$

This rule means that for each of the values in the membership function the consequent parameter is the non-linear system.

The ANFIS model would use the results of the consequent parameter to compute a network (Gopal, 2019). The input datta and training set are computed in each layer and result in a consequent parameter by least square estimator. In terms of the problem above, the consequent parameter would be used to define network output. If the error is high the parameter can be updated (Gopal, 2019).

References

Cococcioni, M., Corsini, G., Lazzerini, B., & Marcelloni, F. (2008). Solving the ocean color inverse problem by using evolutionary multi-objective optimization of neuro-fuzzy systems. *International Journal of Knowledge Based Intelligent Engineering Systems*, *12*(5/6), 339–355.

Fuchs, C., Spolaor, S., Nobile, M. S., & Kaymak, U. (2020). pyFUME: a Python Package for Fuzzy Model Estimation. *2020 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*, 1–8. doi:10.1109/FUZZ48607.2020.9177565

Gopal, M. (2019). *Applied machine learning*. McGraw-Hill Education.

Princeton University. (n.d.). *Fuzzy sets and pattern recognition*. Princeton University. Retrieved December 7, 2021, from https://www.cs.princeton.edu/courses/archive/fall07/cos436/HIDDEN/Knapp/fuzzy004.htm.

Topperly. (2021, Jan. 20). *Sugeno Systems | Graphical Techniques of Inference – Part II | Fuzzy Logic* [Video]. YouTube. https://www.youtube.com/watch?v=GnuseFrYctI

Warner, J., Sexauer, J., scikit-fuzzy, twmeggs, alexsavio, Unnikrishnan, A., … 99991. (2019, November 14). JDWarner/scikit-fuzzy: Scikit-Fuzzy version 0.4.2 (Version v0.4.2). Version v0.4.2. Zenodo. https://doi.org/10.5281/zenodo.3541386

Github: https://github.com/squinton-gcu/Data-Science/tree/main/DSC-540/Assignment6