Applied Computational Intelligence 2023/2024

Lab 5: Fuzzy Sets and Systems (Week 3.1 – 24/9/2024)

1 – Objectives

In this work you will learn how to implement Fuzzy Systems using Simpful, a friendly Python library for Fuzzy Logic reasoning.

You will start by following two examples, one to control a gas burner using a TS Fuzzy System, and another to implement the "Tipping Problem" using a Fuzzy Mamdani expert system. Then you will implement the "CPU Fan Speed Controller" used as a case study during the lectures.

You can also try SciKit Fuzzy as an alternative package to implement Fuzzy Systems.

2 – Simpful

Simpful is a friendly Python library for Fuzzy Logic reasoning. You can find information about this library at:

- https://github.com/aresio/simpful
- https://simpful.readthedocs.io/en/latest/

Install the library and then skim through the documentation.

2.1 – Controlling a gas burner with a Takagi-Sugeno fuzzy system

Check the first example in https://github.com/aresio/simpful. It implements a simple gas burner controller using a Takagi-Sugeno fuzzy system. Study and implement it to learn how to specify information about linguistic variables, fuzzy sets, fuzzy rules, and input values in Simpful.

The last line of code prints the result of the fuzzy reasoning.

2.2 – The Tipping Problem

Now check and implement the second example to learn how to implement a Mamdani fuzzy system (and a few additional tricks).

2.3 – CPU Fan Speed Controller

Based on the previous examples, implement the "CPU Fan Speed Controller" used as a case study during the lectures (see the slides 4-ACI_Fuzzy_Sets_Systems, section "Fuzzy Systems Implementation using a case study"). Try to tune the proposed system by adding an extra membership function to the Fan Speed variable. Check how it affects the output by generating 3D plots.

3 – SciKit-Fuzzy

A more complex alternative to Simpful is the SciKit-Fuzzy, a Fuzzy Logic Toolbox for SciKit. You can find all information about this toolbox at https://pythonhosted.org/scikit-fuzzy/.

You may follow the "User Guide" to get familiar with the toolbox ("Getting started", "Finding your way around").

The main advantage of SciKit-Fuzzy over Simpful is that it facilitates viewing and plotting membership functions and control surfaces. However, you already know how to implement code to do such tasks using, for example, Matplotlib, so it's up to you to see which library you prefer.

3.1 – The Tipping Problem in SciKit-Fuzzy

The SciKit-Fuzzy user guide also includes an example of how to implement a fuzzy expert system. The description of the example is at https://pythonhosted.org/scikit-fuzzy/userguide/fuzzy_control_primer.html.

There are two tutorials with different ways to implement the example. Check both and implement the code in one of the examples. Experience how the Fuzzy system works:

- https://pythonhosted.org/scikit-fuzzy/auto_examples/plot_tipping_problem.html#example-plot-tipping-problem-py
- https://pythonhosted.org/scikit-fuzzy/auto_examples/plot_tipping_problem_newapi.html#example-plot-tipping-problem-newapi-py

The next step should be following the example in https://pythonhosted.org/scikit-fuzzy/auto_examples/plot_control_system_advanced.html#example-plot-control-system-advanced-py, which shows how you can use matplotlib to generate 3D plots of the output of a fuzzy system.