COMP4033 Assignment (2022/2023)

Designing and Tuning a Fuzzy Inference System

Preliminaries

This assignment follows lab sheets 1 and 2. It is worth 20% of the overall module mark.

Deadline for assignment submission via Moodle (report and code) is **21st April 2023, 2pm UK time**, with further detail available on Moodle. The university's standard late submission policy applies to all coursework.

Assessment of the assignment will comprise assessment of the **report** as well as a **viva** conducted in-person during the lab sessions in **weeks 33 and 34**. During the viva you will be asked to download and demonstrate the code you submitted via Moodle, discuss your design and implementation, following by a short Q&A.

Feedback will be provided orally as well as optionally via written comments.

Moodle Submission

You should submit **one** .zip file via Moodle by the deadline, including:

- the full source code, suitably commented and with basic instructions on how to run the code
- the report as below

Remember to **include your name and student number in the code and the report**.

Assignment Overview

The coursework is to design imaginary fuzzy inference systems for advising a doctor whether a patient should be referred to a hospital for emergency investigations based on three inputs: the patient's temperature, severity of their headache, and their age. You will create two fuzzy inference systems, and design/tune the models' parameters by using literature, existing or self-generated (synthetic) data sets and analysing the performance of the various systems. You will write a report discussing your inference systems, comparing and contrasting their different designs, the resulting outputs and associated impact—answering the question of which of the systems may be more or less suitable as a basis for real-world deployment and why.

Marks will be awarded for the processes employed and the analysis carried out, rather than the absolute performance of your final system (i.e. there is no 'correct' final model that you must end up with).

Scenario

The following scenario is a drastically simplified version of a medical scenario, but nevertheless you will need to use systematic principles and processes as involved in creating a solution for a realistic real-world scenario. You have been asked to create a fuzzy inference system (FIS) to provide advice to family doctors that they can use to assess patients when they come in with an illness, to establish whether the patient needs to be sent to hospital for emergency attention. In this very simple scenario, the FIS will take three inputs, *Temperature*, *Headache and Patient-age*, which will be represented as three linguistic variables, and produce a single output, *Urgency*, which represents the degree of urgency (or severity) of the patient's condition. When provided with inputs in terms of temperature (rated on a scale from 0 to 60 degrees Celsius), severity of headache (rated on a subjective scale from 0 to 10), and age (on a scale from 0 to 130), the system will use a set of rules to produce the output i.e. how urgent medical treatment is for the patient.

The output should be quantified in the range of 0 to 100, with 0 meaning the patient's condition is not at all urgent (no need to go to hospital) and 100 meaning extremely urgent (rush to hospital as quickly as possible). Hint: Not all variables are the same. For example, for *Temperature* there is a normal range, and both low and high values can indicate a health problem; while for example for *Headache*, the higher the rating, the worse is the problem. Feel free to research normal and abnormal ranges for example of temperature on sites such as *Wikipedia* and *NHS Direct*, and the various medical terms that may be used to describe abnormal temperatures and different severities of headache.

Fuzzy Inference System(s) – Minimum Requirements

You should develop your fuzzy systems in either R, Python or Java. You can build on your own fuzzy system implementation as developed during lab sheets 1 and 2. You can also make use of the fuzzy system libraries as discussed—see information on Moodle. Note that not all libraries may provide all the features you are looking for. Where this is the case, you should extend the functionality provided using your own code.

You should construct two fuzzy systems designed specifically to work for the following cases:

Case 1: All inputs are numeric, i.e. Temperature, Headache and Age are all provided as **numbers**.

Case 2: All inputs are interval-values, i.e. Temperature, Headache and Age are provided as **intervals**.

When constructing the FIS models, you should carefully consider and justify:

- the number of terms in each input variable and in the output variable, giving each term an appropriate linguistic label
- the types of membership functions (triangular, trapezoidal, Gaussian, sigmoidal)
- the parameters of each membership function
- the fuzzification approach
- the rules that connect the input terms to the output terms
- the operators used in the FIS (consider both min/max and product/probabilistic-or families)
- various defuzzification methods including numeric and linguistic

Note:

- your systems will not be evaluated for medical accuracy in any way. However, their function should be sensible and you should justify how/why it makes sense
- you should **develop the most appropriate type of fuzzy system for Case 1 and Case 2**, using your report to discuss how inputs are processed differently and what the impact is on the outputs generated
- you may develop more than two systems in order to compare and contrast for example different parameter selections or approaches, however the same page limit applies.

Report Deliverable

In addition to your final FISs, you must submit a written report describing the fuzzy modelling process conducted. The length of your report is **1000 - 2000 words (maximum) and eight sides of A4**, *excluding the cover page, but* ***including all*** *tables and figures*, minimum font size 11pt (a full page of text in a similar style to this document would contain about 800 words, so the only around 2½ pages **maximum** of the total report will be text). The report should clearly explain what you did, how and why you did it, and should be well structured and illustrated. It is alright to use less than 2000 words if you are able to describe the details required. Your report should be clear, concise and well structured. It should include three main sections in the main body:

- The first should describe and justify the models you have created, taking into account the model design choices described above.
- The second section should detail the final models, including configuration and full rule set. You must provide illustrative example input-outputs **and these should be comparable across Case 1 and Case 2** to help your discussion. (Hint: For example, you could use example intervals for Case 2, as well as the same intervals' midpoints for both Case 1 and Case 2.)
- The third section should briefly discuss, compare and contrast both of your systems, highlight their strengths/weaknesses and provide a short personal reflection on your work.

You should not include lengthy code, or raw output in the main body of your report, but you may include these in appendices. Note that references and appendices will not contribute to the word count. Appendices and are not explicitly marked: they are for additional reference only. While marks will not be specifically deducted for going over length, ONLY the first 2000 words will be marked (so, for example, if you use 2000 words in Sections 1 & 2, as per below, you will receive no marks for Section 3).

Viva

The viva will be conducted individually during the lab sessions as discussed in the module. You will be asked to demonstrate your implementation (downloaded from your Moodle submission) followed by short Q&A. No preparation is required for the viva.

Assessment Criteria and Marking Guidelines

The assignment will be assessed out of 100 marks as follows:

[70-100] an excellent, well-written report and viva demonstrating extensive understanding, good insight and strong justification for design approach taken, complemented by a clear implementation and convincing presentation.

[60-69] a comprehensive, well-written report and viva demonstrating thorough understanding, some insight and justification for design approach taken, complemented by a clear implementation and presentation

[50-59] a competent report/viva, implementation and presentation demonstrating good understanding of the implementation

[40-49] an adequate report/viva, implementation and demonstration covering all specified topics at a basic level of understanding

[0-39] an inadequate report/viva failing to cover the specified topics

Marks will be awarded for the overall structure, style and presentation of your report. A first class report would have a clear structure with appropriate sub-headings, well-written, clear informative text, and clear and attractive pictures, graphs and tables as well as suitable references. Note that while your source code will not be marked individually, it should be suitably commented and be submitted complete with running-instructions as outlined above.

Plagiarism vs. Group Discussions

As you know, plagiarism is completely unacceptable and will be dealt with according to the University's standard policies. Having said this, we do encourage students to have general discussions regarding the coursework with each other in order to promote the generation of new ideas and to enhance the learning experience. This being said, you must be careful not to cross the boundary into plagiarism. The important part is that when you sit down to actually do the system design and write about it, you do it individually. If you do this, and you truly understand what you have written, you will not be guilty of plagiarism. Do NOT, under any circumstances, share code, figures, or graphs, etc.