

Assignment 3 – Support System For Project Team Formation

The problem of optimal team formation is domestic to many areas of work organization including education, sport, and business. It is beyond manual implementation to build near optimal teams as soon as the pool of available personnel grows into several tens. The selection process itself is usually well defined for each team we construct the criteria relating to the required properties of the team members. Because these properties can be arbitrarily combined in the personnel, the objective function becomes self-conflicting. This aggravates the team formation and calls for specialized software support.

We use quantities values to describe the employees' capabilities like technical expertise and also to describe the budget limits needed to format a team

The problem is to estimate the level of risk involved in a software engineering project. For the sake of simplicity, we will arrive at our conclusion based on two inputs: project funding and technical experience for the team members.

Suppose our inputs are `project_funding` and `team_experience_level`. We can get the fuzzy values for these crisp values by using the membership functions of the appropriate sets.

The sets defined for **project_funding** are very low (0,0,10,30), low (10,30,40,60), medium (40,60,70,90), high (70,90,100,100).

The sets defined for **team_experience_level** are beginner (0,15,30), intermediate (15,30,45), expert (30,60,60).

The set defined for the **risk** is low (0,25,50), normal (25,50,75), high (50,100,100).

The Rules

Now that we have the fuzzy values and we can use the fuzzy rules to arrive at the final fuzzy value. The rules are as follows:

1. If *project_funding* is **high** or *team_experience_level* is **expert** then risk is **low**.
2. If *project_funding* is **medium** and *team_experience_level* is **intermediate** or *team_experience_level* is **beginner** then risk is **normal**.
3. If *project_funding* is **very low** then risk is **high**.
4. If *project_funding* is **low** and *team_experience_level* is **beginner** then risk is **high**.

Input:

- 1- First line represents number of input variables = 2
- 2- Second line gives a Variable Name and its crisp input to fuzzify it later (e.g. project funding 50, experience level 40).

Output:

- 1- Fuzzifying the inputs
- 2- Inference of rules
- 3- Defuzzification output

Test Case

Input:

Variables: 2

Project Fund: 50

Experience Level: 40

Output:

Predicted Value (Risk) = 37.5

Risk will be Normal

Assignment submission notes:

- The maximum number of students in a team is 3 and the minimum is 2.
- No late submission is allowed.
- Please submit one compressed folder. The folder name should follow this structure: ID1_ID2_ID3 _GROUPNAME
- Cheating students will take ZERO and no excuses will be accepted. If you have any problems during the submission, contact your TA but don't, under any circumstances, give your code to or take the code from your friends.

Grading Criteria: (6 marks)

Lines Equations	1
Membership Values	1
Inference	1
Centroids	1
Predicted Risk Value & Range in which predicted risk value exists	2
Total	6