

Report for Banco Pichin: Development of a Mamdani Fuzzy Inference System (MFIS) for Loan Risk Assessment

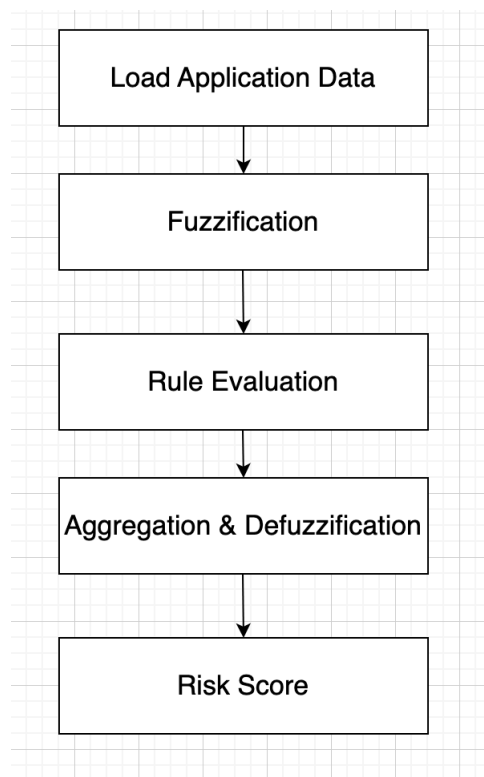
-By Aryan Daga(100531454), Sarthak Patel(100531789)

Executive Summary

This project involves developing a decision-support system for Banco Pichin to assist in the assessment of personal loan applications. The system is built using a Mamdani Fuzzy Inference System (MFIS) that evaluates the risk associated with loan applications based on various factors such as the applicant's age, income level, assets, loan amount, job stability, and credit history. By leveraging fuzzy logic, the system can handle uncertainties and imprecise inputs, providing a nuanced risk assessment. The output is a risk score, which aids Banco Pichin in making informed decisions regarding loan approvals.

The system begins by taking loan application data as input, including variables like age, income level, assets, amount requested, job stability, and credit history. These inputs are then fuzzified into degrees of membership using predefined fuzzy sets. The fuzzified inputs are evaluated against a set of expert-defined rules to determine their contributions to the risk assessment. The results from the rule evaluations are aggregated, and the final risk score is obtained through defuzzification. This risk score indicates the likelihood of loan repayment and helps Banco Pichin decide whether to approve or reject a loan application.

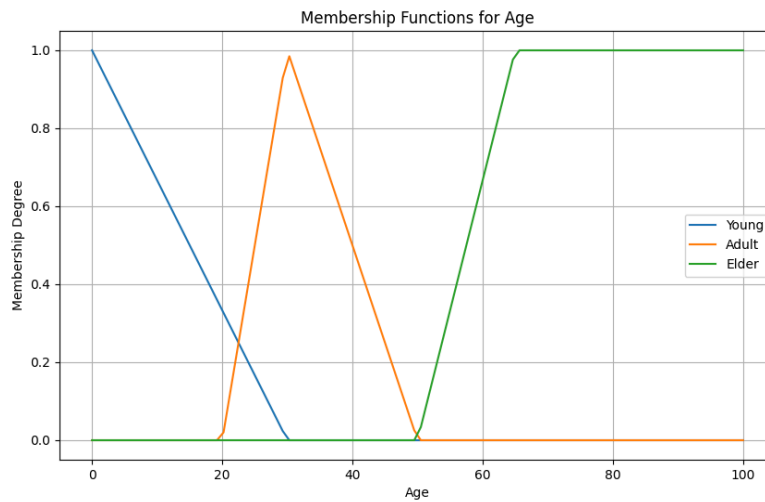
System Diagram:

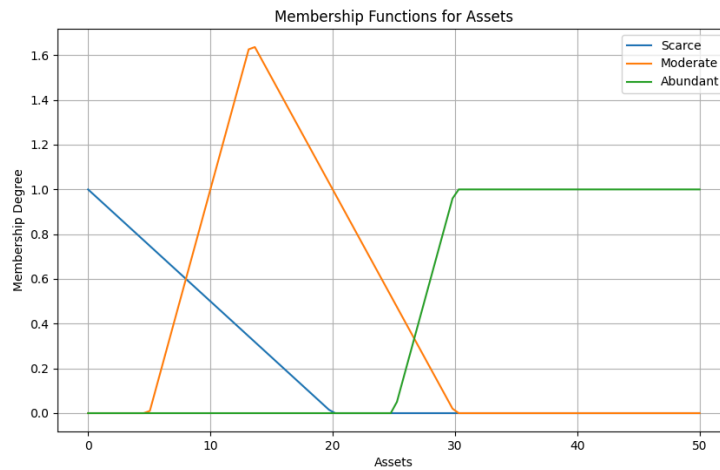
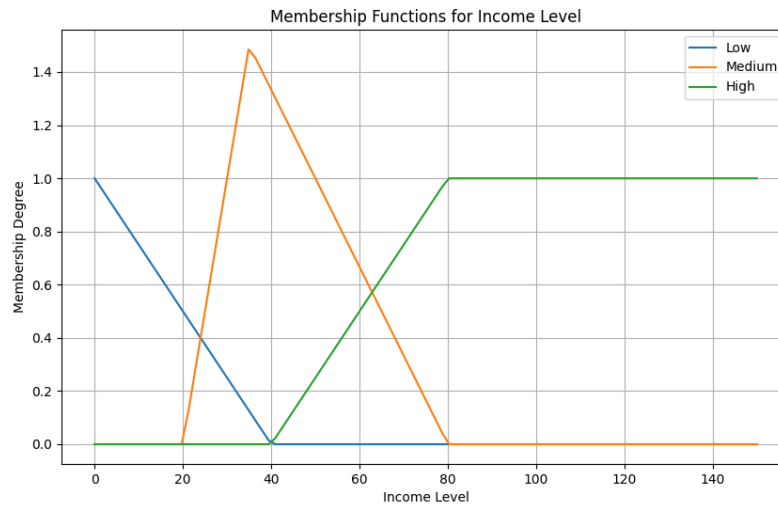


Description of the Inference System

The constructed inference system for Banco Pichin is a Mamdani Fuzzy Inference System (MFIS). This system evaluates the risk level of each loan application based on multiple input factors and provides a crisp risk score as the output. The system operates through several key components:

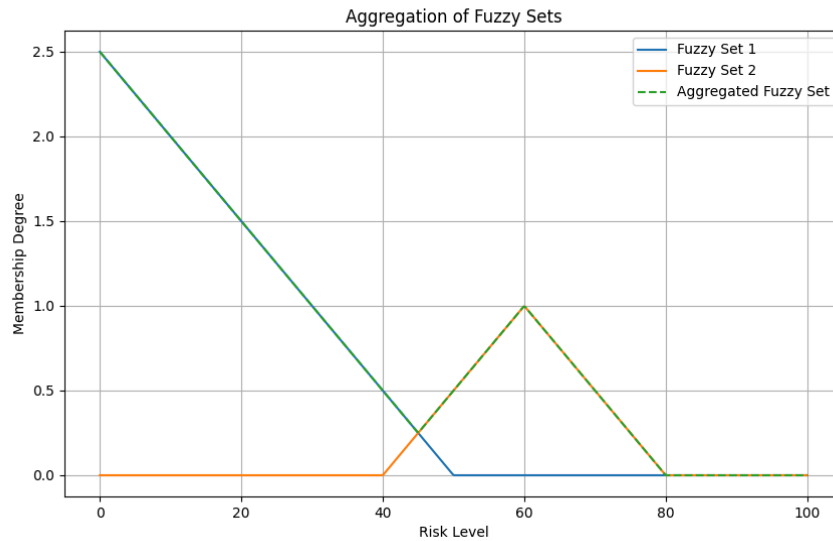
1. **Fuzzification:** This initial step converts crisp input values into degrees of membership for fuzzy sets. The input variables include Age, Income Level, Assets, Amount, Job, and History. Each variable is associated with several fuzzy sets, such as 'Young', 'Adult', and 'Elder' for Age, which are defined using trapezoidal or triangular membership functions.



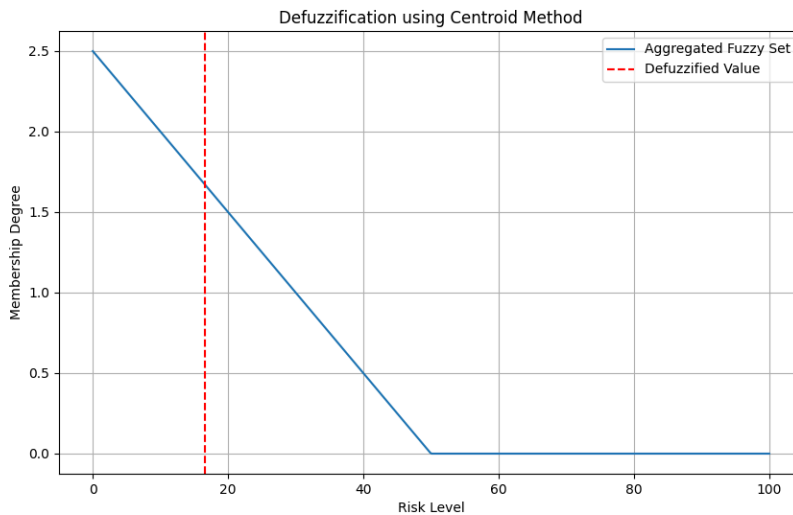


2. **Rule Evaluation:** The system uses a set of expert-defined rules provided by Banco Pichin. Each rule consists of antecedents (conditions on the input variables) and a consequent (the resulting risk level). For example, a rule might state that if the Age is 'Young' and the Income Level is 'Low', then the Risk is 'High'. The system evaluates each rule by calculating the degree to which the antecedents are satisfied and then determining the strength of the consequent.
3. **Aggregation:** After evaluating all the rules, the system aggregates the outputs. This process involves combining the fuzzy sets resulting from the rule evaluations into a single fuzzy set for the risk level. The aggregation method ensures that all relevant

information from the rules is considered in the final risk assessment.



4. **Defuzzification:** The final step converts the aggregated fuzzy set into a single crisp risk score. The centroid method is used for defuzzification, which finds the center of gravity of the aggregated fuzzy set. This risk score is a numeric value that represents the overall risk associated with the loan application.



By employing this MFIS, Banco Pichin can assess the risk level of each loan application more accurately, considering the uncertainties and complexities inherent in the input data. This system provides a robust and flexible approach to risk assessment, enabling more informed and reliable decision-making regarding loan approvals.

Methodology

Phase 1: Definition of Variables

- Task 1: Identify input variables crucial for loan risk assessment. The key variables identified were Age, Income Level, Assets, Amount, Job, and History. These variables were selected based on their significant impact on an applicant's ability to repay a loan.
- Task 2: Create fuzzy sets for each variable with corresponding membership functions. For instance, Age was categorized into 'Young', 'Adult', and 'Elder', each represented by a fuzzy set with specific membership functions. These sets were defined in the InputVarSets.txt file.

Phase 2: Definition of Rules

- Task 1: Collect expert knowledge to define inference rules. We consulted with domain experts from Banco Pichin to gather insights on how different combinations of input variables affect loan risk.
- Task 2: Compile rules into the Rules.txt file. Each rule specifies conditions on input variables and the resulting risk level. For example, one rule might state that if an applicant is 'Young' and has a 'Low' income level, then the risk level is 'High'.

Phase 3: System Implementation

- Task 1: Develop the FuzzySet and Rule classes in Python to represent fuzzy sets and rules. The FuzzySet class handles the membership functions, while the Rule class manages the antecedents and consequents of the rules.
- Task 2: Implement functions for reading fuzzy sets, rules, and application data from text files. These functions ensure that the system can dynamically load and process the necessary input data.
- Task 3: Construct the fuzzification, rule evaluation, aggregation, and defuzzification processes. These processes are the core components of the MFIS, responsible for transforming input data into a final risk score.
- Task 4: Integrate all components into a complete system. This involved combining the classes and functions into a cohesive program that can evaluate loan applications and produce risk scores.

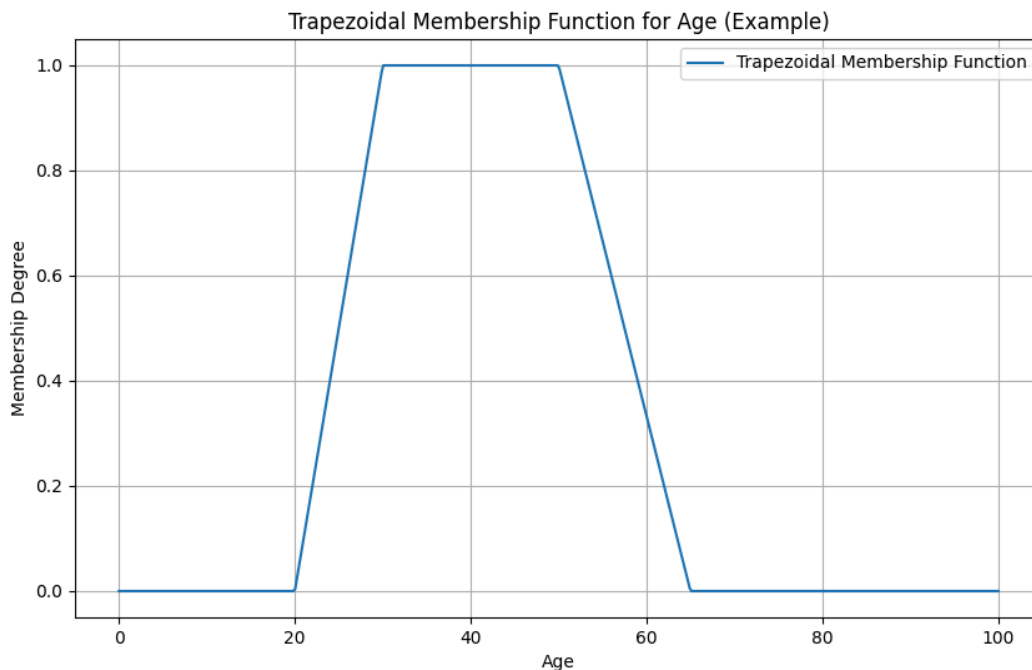
Phase 4: Testing and Validation

- Task 1: Test the system with various loan applications to ensure correct risk assessment. We used sample data to test the system, verifying that the outputs matched expected results.
- Task 2: Validate the system's performance by comparing the output risk scores with expected results. We compared the system's risk assessments with historical data and expert evaluations to ensure accuracy and reliability.

Phase 5: Documentation and Delivery

- Task 1: Prepare the project report detailing the system, methodology, and results. This report provides a comprehensive overview of the project's development and outcomes.
- Task 2: Create a video presentation explaining the project and demonstrating the system. The video showcases the system's functionality and benefits to Banco Pichin.
- Task 3: Deliver the final package, including source code, results, and analysis. The final package ensures that Banco Pichin has all the necessary materials to understand and utilize the system effectively.

Example of a Fuzzy Set



Budget

The total cost for the development and implementation of the MFIS project is estimated at \$50,000. This cost includes:

- Development Costs: \$30,000
 - System design and implementation: This covers the expenses for developing the MFIS, including coding, testing, and integrating the system.
 - Integration and testing: This involves ensuring the system works seamlessly with Banco Pichin's existing processes and thoroughly testing it for reliability.
- Expert Consultation: \$10,000
 - Involvement of domain experts for rule definition and validation: This cost covers the consultation fees for experts who provided the necessary knowledge to define and validate the inference rules.
- Documentation and Training: \$5,000
 - Preparation of detailed project report and training materials: This includes the cost of creating comprehensive documentation and training resources to help Banco Pichin understand and use the system.
- Project Management: \$5,000
 - Coordination and oversight of project activities: This covers the expenses related to managing the project, ensuring it stays on track and meets all objectives.

The comprehensive solution provided by C3L ensures that Banco Pichin receives a robust and effective decision-support system, enhancing their loan approval process. The budget reflects the expertise and effort required to deliver a high-quality system tailored to Banco Pichin's needs.