Final Project Report: Heart Disease Detection Using Fuzzy Logic

ABSTRACT:

This project introduces a Fuzzy Logic-Based System for the early detection of heart disease, utilizing the flexibility of fuzzy logic to handle uncertainties in medical datasets. By integrating various clinical parameters, including medical history and diagnostic tests, the system aims to model complex relationships among risk factors. Through rigorous training and validation, the system's performance will be assessed, with the goal of providing a reliable and interpretable tool for clinicians. This research highlights the potential of fuzzy logic in enhancing diagnostic accuracy and contributing to more effective healthcare decision-making.

INTRODUCTION:

The term “heart disease” refers to several types of heart conditions. There are many different heart conditions and problems which are collectively called heart disease. The four main types are: coronary heart disease, stroke, peripheral arterial disease, and aortic disease. Coronary artery disease or CAD is the most common type of heart disease in the United States and affects the blood flow to the heart. Decreased blood flow can cause a heart attack.

Sometimes heart disease may be “silent” and not diagnosed until a person experiences signs or symptoms of a heart attack, heart failure, or an arrhythmia. When these events happen, symptoms may include:

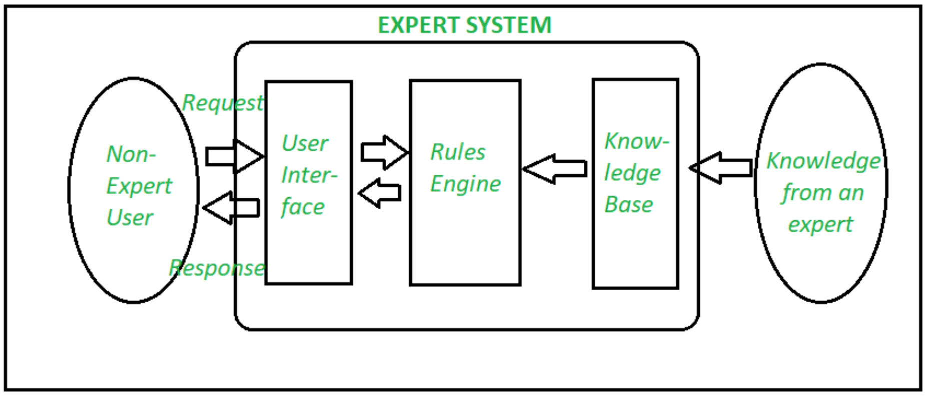
1. ***Heart attack:*** *Chest pain or discomfort, upper back or neck pain, indigestion, heartburn, nausea or vomiting, extreme fatigue, upper body discomfort, dizziness, and shortness of breath.*
2. ***Arrhythmia:*** *Fluttering feelings in the chest (palpitations).*
3. ***Heart failure:*** *Shortness of breath, fatigue, or swelling of the feet, ankles, legs, abdomen, or neck veins.*

High blood pressure, high blood cholesterol, and smoking are key risk factors for heart disease. Several other medical conditions and lifestyle choices can also put people at a higher risk for heart disease, including:

- *Diabetes*
- *Obesity*
- *Unhealthy diet*
- *Physical inactivity*
- *Excessive alcohol use*

Importance of Early Detection: The impact of late diagnosis of medical conditions is profound, leading to increased treatment complexity, reduced treatment success rates, and higher healthcare costs. Late-stage interventions often require more extensive and resource-intensive approaches. In contrast, early detection offers a multitude of benefits, including improved treatment efficacy, enhanced quality of life for patients, and ultimately lower healthcare costs. Early detection plays a pivotal role in preventive healthcare by empowering patients with timely information, contributing to positive population health outcomes. Regular check-ups are crucial in identifying potential health issues at their nascent stages, allowing for more effective and less burdensome interventions. This proactive approach not only benefits individual patients but also promotes a healthier society at large.

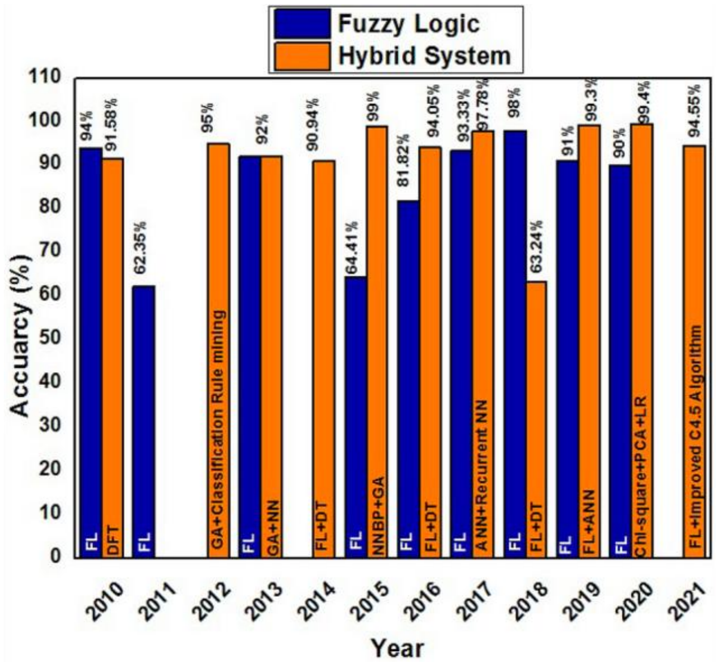
Expert systems are computer-based systems mimicking problem-solving ability of a human expert. They can be used to analyze medical data and emulate expert decision-making. The diagram shows the structure of a simple expert system:



Machine learning models can also be used to analyze healthcare data for predictions, diagnosis, and treatment planning. Decision Trees are versatile supervised machine-learning algorithms applicable to both classification and regression problems. Their distinctive flowchart-like tree structure involves internal nodes representing features, branches denoting rules, and leaf nodes indicating the algorithm's result. The C4.5 algorithm enhances traditional decision trees by accommodating both numeric and categorical attributes, incorporating pruning techniques, and utilizing a robust attribute selection measure known as Gain Ratio.

Random Forests, on the other hand, are ensemble algorithms comprising numerous decision trees. Renowned for their high accuracy and scalability in handling large datasets, Random Forests leverage the collective wisdom of multiple trees to yield robust and reliable predictions in diverse applications.

Combining expert systems and machine learning gives better and more accurate results. The graph shown below shows statistics of how a fuzzy logic system performs against a system that combines both approaches to form a hybrid system. This was taken from the survey paper I referred to.



Heart disease remains a leading cause of mortality worldwide. Early detection is crucial for effective intervention and management. This project aimed to develop a Heart Disease Detection System using Fuzzy Logic, a computational paradigm that mimics human decision-making processes. The system analyzed a dataset of relevant clinical parameters to provide accurate and interpretable results.

2. **OBJECTIVES:**

- Develop a Fuzzy Logic model for heart disease detection.
- Utilize a comprehensive dataset containing clinical parameters.
- Evaluate the model's performance against existing benchmarks.

3. **METHODOLOGY:**

3.1 **Data Collection and Preprocessing:**

The Cleveland Heart Disease dataset is taken from the UCI ML repository, and it comprises various clinical parameters such as age, blood pressure, cholesterol levels, and electrocardiogram results. Data preprocessing involved handling missing values, normalization, and feature selection to enhance model efficiency.

It consists of 303 records with 76 attributes. Research papers tend to use only a subset of 14, with 13 feature variables and 1 target variable. This is how the features look like –

	age	sex	cp	trestbps	chol	fbs	restecg	thalach	exang	oldpeak	slope	ca	thal
0	63	1	1	145	233	1	2	150	0	2.3	3	0.0	6.0
1	67	1	4	160	286	0	2	108	1	1.5	2	3.0	3.0
2	67	1	4	120	229	0	2	129	1	2.6	2	2.0	7.0
3	37	1	3	130	250	0	0	187	0	3.5	3	0.0	3.0
4	41	0	2	130	204	0	2	172	0	1.4	1	0.0	3.0
...	...	...	...	...	...	...	...	...	...	...	...	...	...
298	45	1	1	110	264	0	0	132	0	1.2	2	0.0	7.0
299	68	1	4	144	193	1	0	141	0	3.4	2	2.0	7.0
300	57	1	4	130	131	0	0	115	1	1.2	2	1.0	7.0
301	57	0	2	130	236	0	2	174	0	0.0	2	1.0	3.0
302	38	1	3	138	175	0	0	173	0	0.0	1	0.0	3.0

303 rows × 13 columns

Sr. No.	Column Name	Abbreviation	Column Type	Description	Data Type
1	Age	-	Feature	Age of the patient	Integer
2	Sex	-	Feature	Gender of the patient	Categorical (typically binary: 1 for female, 0 for male)
3	Chest Pain	cp	Feature	Type of chest pain experienced by the patient	Categorical (values may represent different types of chest pain)
4	Resting Blood Pressure	trestbps	Feature	Resting blood pressure of the patient	Integer
5	Cholesterol	chol	Feature	Serum cholesterol levels of the patient	Integer
6	Fasting Blood Sugar	fbs	Feature	Fasting blood sugar levels of the patient	Categorical (binary: 0 for false, 1 for true)
7	Resting Electrocardiographic Results	restecg	Feature	Resting electrocardiographic results, which may indicate the presence of heart-related abnormalities	Categorical
8	Maximum Heart Rate Achieved	thalach	Feature	Maximum heart rate achieved during a stress test	Integer
9	Exercise-Induced Angina	exang	Feature	Whether the patient experienced angina	Categorical (binary)

				(chest pain) induced by exercise.	
10	Old Peak	-	Feature	ST depression induced by exercise relative to rest	Numeric/ Integer
11	Slope	-	Feature	Slope of the peak exercise ST segment	Categorical
12	Number of Major Vessels	numvessels	Feature	Number of major vessels colored by fluoroscopy	Integer
13	Thallium Scan	thal	Feature	Results of thallium heart scan	Categorical
14	Target (Diagnosis of Heart Disease)	-	Target	Presence or absence of heart disease	Categorical (binary: 0 for no heart disease, 1 for heart disease)

3.2 **Fuzzy Logic Model Design:**
Introduced by Lotfi Zadeh in the 1960s, Fuzzy Logic is a paradigm that addresses reasoning and decision-making in scenarios characterized by uncertainty, imprecision, and vagueness. Diverging from classical or "crisp" logic, which relies on binary true or false values, Fuzzy Logic enables the representation of degrees of truth. This unique feature allows for a more nuanced and flexible approach to modeling complex systems, where precise categorizations may not capture the inherent uncertainties of real-world situations. Fuzzy Logic has found applications across various fields, providing a powerful tool for handling the intricacies of systems influenced by imprecise information and uncertain conditions.

Fuzzy Logic incorporates linguistic variables and human-like reasoning to handle uncertainty. The membership functions and rules were defined based on domain knowledge and input from healthcare professionals. The Mamdani-type fuzzy inference system was employed.

3.4 **How is a Fuzzy Logic Based Expert System Designed?**
Designing a Fuzzy Logic Based Expert System involves several key stages, ensuring the effective incorporation of domain knowledge and the creation of a robust decision-making framework.

1. **Knowledge Acquisition:** involves gathering relevant knowledge from domain experts and available literature. This step includes identifying the key variables and their relationships, as well as defining the linguistic terms that describe these variables. Experts' insights are crucial for establishing membership functions and rules that capture the nuances of human decision-making in the specific domain.

2. **Rule-base Construction:** The acquired knowledge is translated into a rule-based system. Rules in fuzzy logic consist of "if-then" statements, expressing the relationships between the input variables and the desired output. Membership functions for each variable are defined, representing the degree of membership to different linguistic terms (e.g., low, medium, high).

The rule-base is constructed by mapping the combinations of linguistic terms from input variables to output linguistic terms, reflecting the expert's knowledge in a formalized manner.

3. **Implementation and Integration:** Once the rule-base is established, the fuzzy logic system is implemented using programming languages and tools suitable for fuzzy logic, such as MATLAB or Python with libraries like scikit-fuzzy. The implementation involves defining the fuzzy sets, membership functions, and rules within the chosen programming environment. The fuzzy inference engine processes input data through the rule-base, aggregating the rules to generate a fuzzy output. This output is then defuzzified to obtain a crisp result that can be easily interpreted.

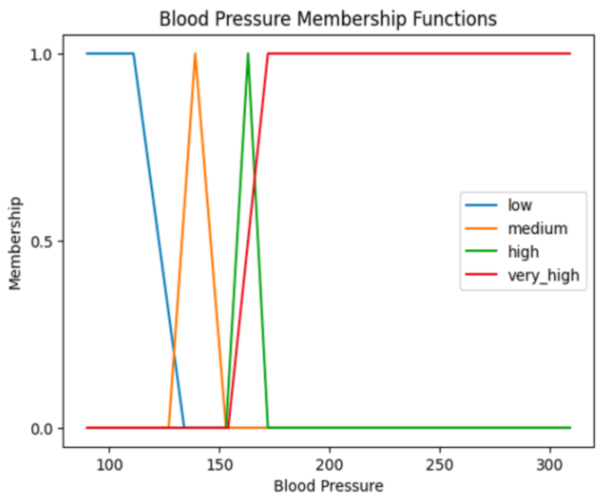
4. **Testing and Maintenance:** Thorough testing is essential to ensure the reliability and accuracy of the fuzzy logic expert system. Test cases, including scenarios covering a wide range of input combinations, are used to evaluate the system's performance. Adjustments to membership functions or rules may be made based on the testing results, and feedback from domain experts can further refine the system. Continuous maintenance involves updating the system as new knowledge becomes available or when changes in the domain occur. This may include refining rules, modifying membership functions, or expanding the system's capabilities to adapt to evolving requirements.

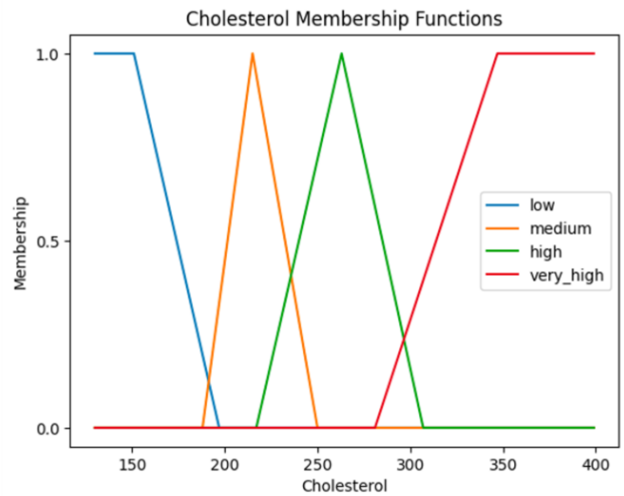
3.3 **Implementation:**

The model was implemented using Python. The machine learning models used were a Decision Tree model and a Random Forest model. I used the *"control"* sub package in the *skfuzzy* toolkit in Python for fuzzy system design. The system allowed users to input relevant clinical data, and the results were presented visually for easy interpretation.

3.3.1 **Membership Functions:** Here are the membership ranges of the crisp sets and the functions of fuzzy sets:

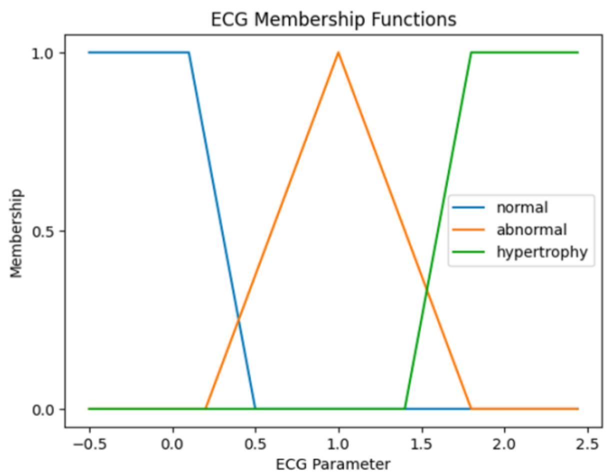
chest_pain_values = {'typical_anginal': 1, 'atypical_anginal': 2, 'non_aginal_pain': 3, 'asymptomatic': 4}
exercise_values = {'true': 1, 'false': 0}
thallium_scan_values = {'normal': 3, 'fixed_defect': 6, 'reversible_defect': 7}
sex_values = {'female': 1, 'male': 0}

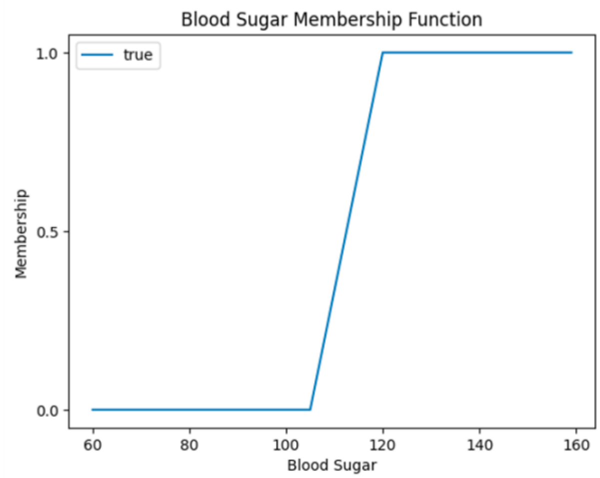




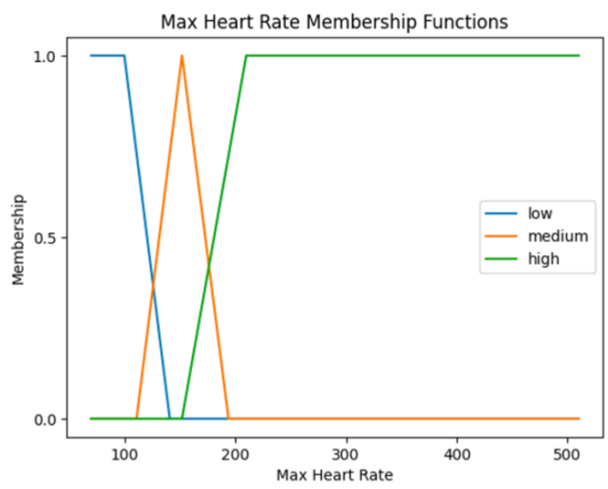
INPUT FIELD	RANGE	FUZZY SETS
Systolic Blood Pressure	<134	Low
	127-153	Medium
	142-172	High
	154>	Very high

INPUT FIELD	RANGE	FUZZY SETS
Cholesterol	<197	Low
	188-250	Medium
	217-307	High
	281>	Very high

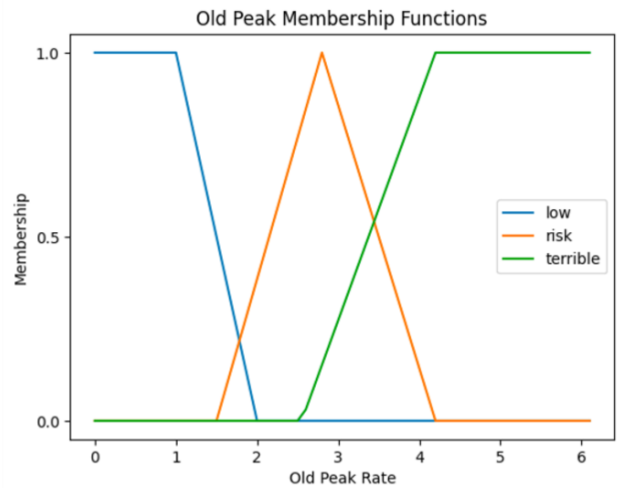




INPUT FIELD	RANGE	FUZZY SETS
Resting Electrocardiography (ECG)	(0) [-0.5, 0.4]	Normal
	(1) [2.45, 1.8]	ST-T abnormal
	(2) [1.4, 2.5]	Hypertrophy

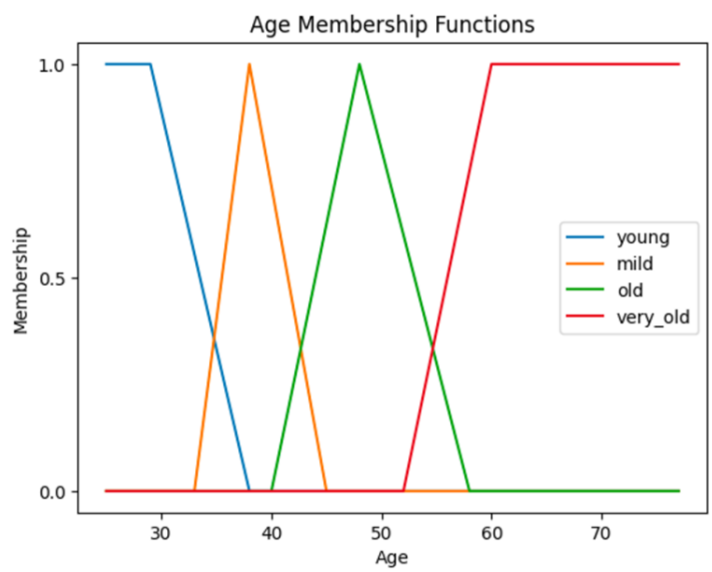


INPUT FIELD	RANGE	FUZZY SETS
Maximum Heart Rate	<141	Low
	111-194	Medium
	152>	High



INPUT FIELD	RANGE	FUZZY SETS
Old Peak	<2	Low
	1.5-4.2	Risk
	2.55>	Terrible

INPUT FIELD	RANGE	FUZZY SETS
Age	<38	Young
	33-45	Mild
	40-58	Old
	52>	Very old



3.3.2 **Fuzzy Rule Base:** In the original paper, there were 44 rules established, and I further expanded the rule set by incorporating additional rules that implemented complex logical conditions using AND and OR operators. These augmentations were derived from insights gained from other relevant papers and thorough analyses conducted with machine learning models. Each rule in the system comprises an antecedent, representing conditions based on input variables, and a consequent, specifying the resulting output. This comprehensive set of rules forms the foundation of the fuzzy logic-based expert system, allowing it to make nuanced and informed decisions in the domain of heart disease detection.

Here is how it looks:

```python
rules.append(ctrl.Rule(antecedent=((age['very_old'] & chest_pain['atypical_anginal'])), consequent=health['sick_4']))
rules.append(ctrl.Rule(antecedent=((maximum_heart_rate['high'] & age['old'])), consequent=health['sick_4']))
rules.append(ctrl.Rule(antecedent=((sex['male'] & maximum_heart_rate['medium'])), consequent=health['sick_3']))
rules.append(ctrl.Rule(antecedent=((sex['female'] & maximum_heart_rate['medium'])), consequent=health['sick_2']))
rules.append(ctrl.Rule(antecedent=((chest_pain['non_anginal_pain'] & blood_pressure['high'])), consequent=health['sick_3']))
rules.append(ctrl.Rule(antecedent=((chest_pain['typical_anginal'] & maximum_heart_rate['medium'])), consequent=health['sick_2']))
rules.append(ctrl.Rule(antecedent=((blood_sugar['true'] & age['mild'])), consequent=health['sick_3']))
rules.append(ctrl.Rule(antecedent=((blood_sugar['false'] & blood_pressure['very_high'])), consequent=health['sick_2']))
rules.append(ctrl.Rule(antecedent=((chest_pain['asymptomatic'] | age['very_old'])), consequent=health['sick_1']))
rules.append(ctrl.Rule(antecedent=((blood_pressure['high'] | maximum_heart_rate['low'])), consequent=health['sick_1']))

rules.append(ctrl.Rule(antecedent=(chest_pain['typical_anginal']), consequent=health['healthy']))
rules.append(ctrl.Rule(antecedent=(chest_pain['atypical_anginal']), consequent=health['sick_1']))
rules.append(ctrl.Rule(antecedent=(chest_pain['non_anginal_pain']), consequent=health['sick_2']))
rules.append(ctrl.Rule(antecedent=(chest_pain['asymptomatic']), consequent=health['sick_3']))

rules.append(ctrl.Rule(antecedent=(chest_pain['asymptomatic']), consequent=health['sick_4']))

rules.append(ctrl.Rule(antecedent=(sex['female']), consequent=health['sick_1']))
rules.append(ctrl.Rule(antecedent=(sex['male']), consequent=health['sick_2']))

rules.append(ctrl.Rule(antecedent=(blood_pressure['low']), consequent=health['healthy']))
rules.append(ctrl.Rule(antecedent=(blood_pressure['medium']), consequent=health['sick_1']))
rules.append(ctrl.Rule(antecedent=(blood_pressure['high']), consequent=health['sick_2']))
rules.append(ctrl.Rule(antecedent=(blood_pressure['high']), consequent=health['sick_3']))
rules.append(ctrl.Rule(antecedent=(blood_pressure['very_high']), consequent=health['sick_4']))

rules.append(ctrl.Rule(antecedent=(cholesterol['low']), consequent=health['healthy']))
rules.append(ctrl.Rule(antecedent=(cholesterol['medium']), consequent=health['sick_1']))
rules.append(ctrl.Rule(antecedent=(cholesterol['high']), consequent=health['sick_2']))
rules.append(ctrl.Rule(antecedent=(cholesterol['high']), consequent=health['sick_3']))
rules.append(ctrl.Rule(antecedent=(cholesterol['very_high']), consequent=health['sick_4']))

rules.append(ctrl.Rule(antecedent=(blood_sugar['true']), consequent=health['sick_2']))
```

```python
rules.append(ctrl.Rule(antecedent=(ECG['normal']), consequent=health['healthy']))
rules.append(ctrl.Rule(antecedent=(ECG['normal']), consequent=health['sick_1']))
rules.append(ctrl.Rule(antecedent=(ECG['abnormal']), consequent=health['sick_2']))
rules.append(ctrl.Rule(antecedent=(ECG['hypertrophy']), consequent=health['sick_3']))
rules.append(ctrl.Rule(antecedent=(ECG['hypertrophy']), consequent=health['sick_4']))

rules.append(ctrl.Rule(antecedent=(exercise['true']), consequent=health['sick_2']))

rules.append(ctrl.Rule(antecedent=(maximum_heart_rate['low']), consequent=health['healthy']))
rules.append(ctrl.Rule(antecedent=(maximum_heart_rate['medium']), consequent=health['sick_1']))
rules.append(ctrl.Rule(antecedent=(maximum_heart_rate['medium']), consequent=health['sick_2']))
rules.append(ctrl.Rule(antecedent=(maximum_heart_rate['high']), consequent=health['sick_3']))
rules.append(ctrl.Rule(antecedent=(maximum_heart_rate['high']), consequent=health['sick_4']))

rules.append(ctrl.Rule(antecedent=(old_peak['low']), consequent=health['healthy']))
rules.append(ctrl.Rule(antecedent=(old_peak['low']), consequent=health['sick_1']))
rules.append(ctrl.Rule(antecedent=(old_peak['terrible']), consequent=health['sick_2']))
rules.append(ctrl.Rule(antecedent=(old_peak['terrible']), consequent=health['sick_3']))
rules.append(ctrl.Rule(antecedent=(old_peak['risk']), consequent=health['sick_4']))

rules.append(ctrl.Rule(antecedent=(thallium['normal']), consequent=health['healthy']))
rules.append(ctrl.Rule(antecedent=(thallium['normal']), consequent=health['sick_1']))
rules.append(ctrl.Rule(antecedent=(thallium['fixed_defect']), consequent=health['sick_2']))
rules.append(ctrl.Rule(antecedent=(thallium['reversible_direct']), consequent=health['sick_3']))
rules.append(ctrl.Rule(antecedent=(thallium['reversible_direct']), consequent=health['sick_4']))

rules.append(ctrl.Rule(antecedent=(age['young']), consequent=health['healthy']))
rules.append(ctrl.Rule(antecedent=(age['mild']), consequent=health['sick_1']))
rules.append(ctrl.Rule(antecedent=(age['old']), consequent=health['sick_2']))
rules.append(ctrl.Rule(antecedent=(age['old']), consequent=health['sick_3']))
rules.append(ctrl.Rule(antecedent=(age['very_old']), consequent=health['sick_4']))
```

4. **RESULTS:**

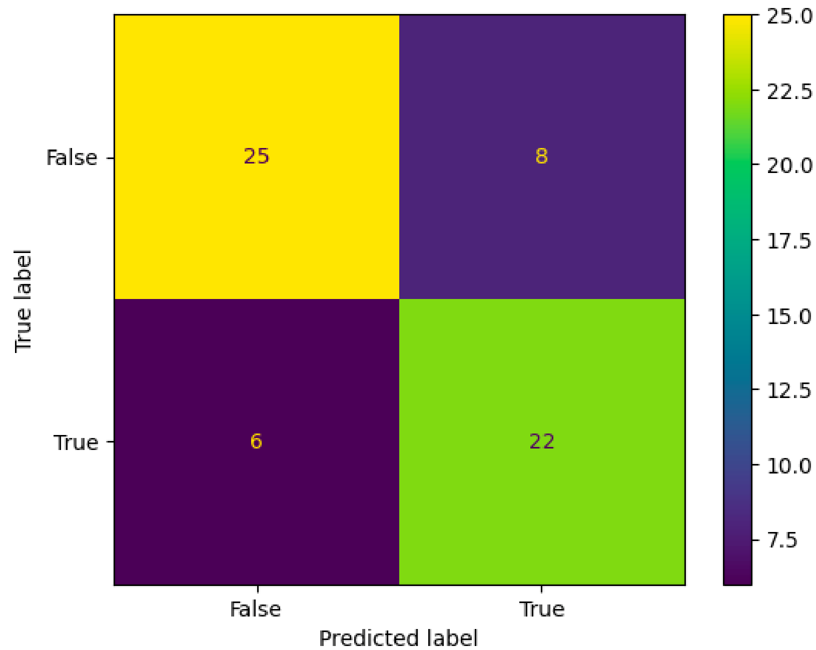
4.1 **Model Evaluation:**

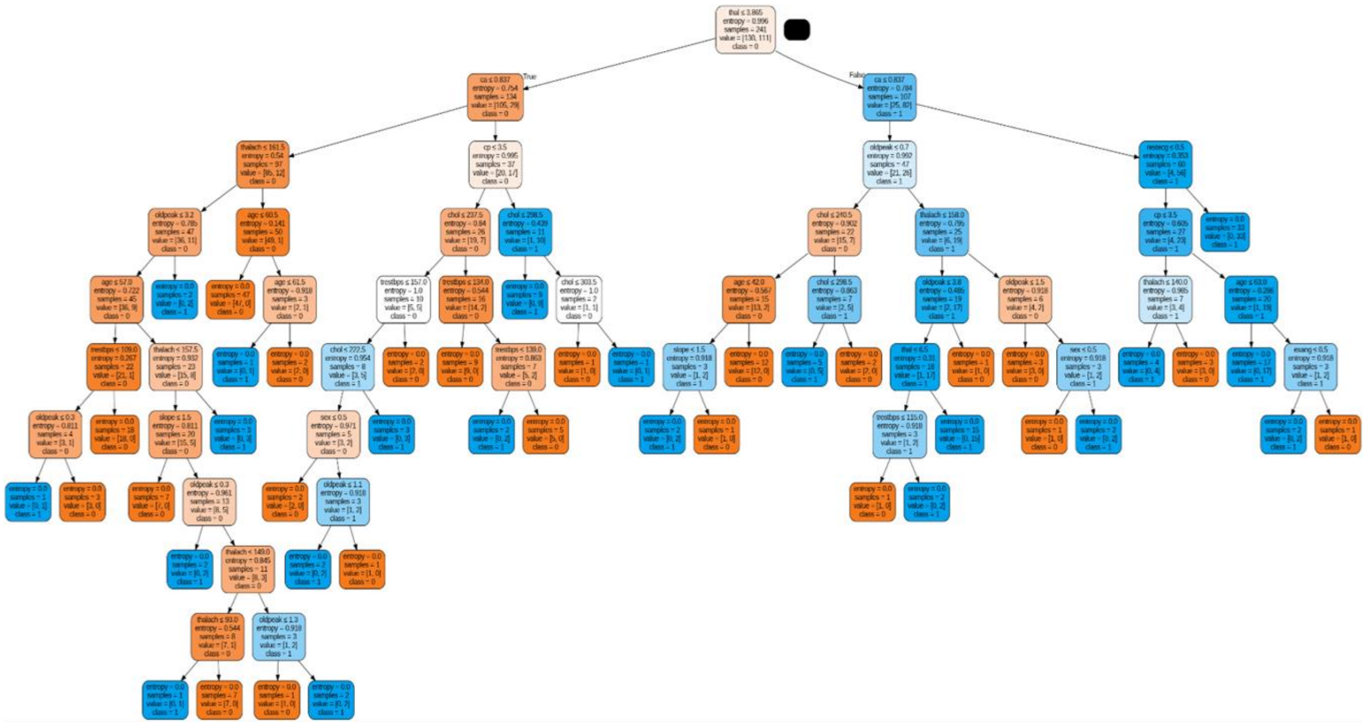
The performance of the Fuzzy Logic model was assessed using metrics such as accuracy, confusion matrix and root node selection. Comparative analysis with between both machine learning models was conducted to validate the effectiveness of the proposed approach.

4.1.1. **Decision Tree Analysis:**

The decision tree model was implemented using Scikit-learn with the criterion set to "entropy," essentially emulating the C4.5 Algorithm. An accuracy of 77% was achieved with the decision tree model. I noticed that the decision tree performed better when the criterion was set to entropy, compared to without.

Here is how the confusion matrix and decision tree look like:

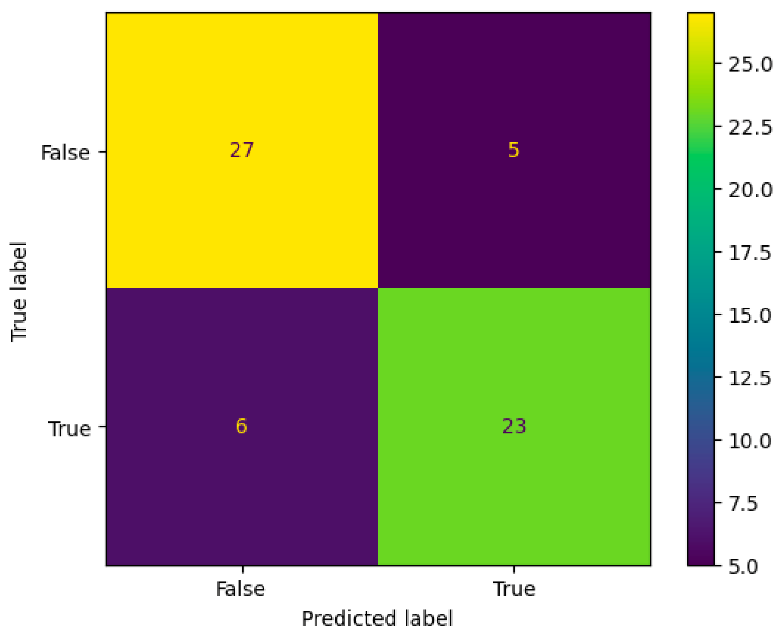


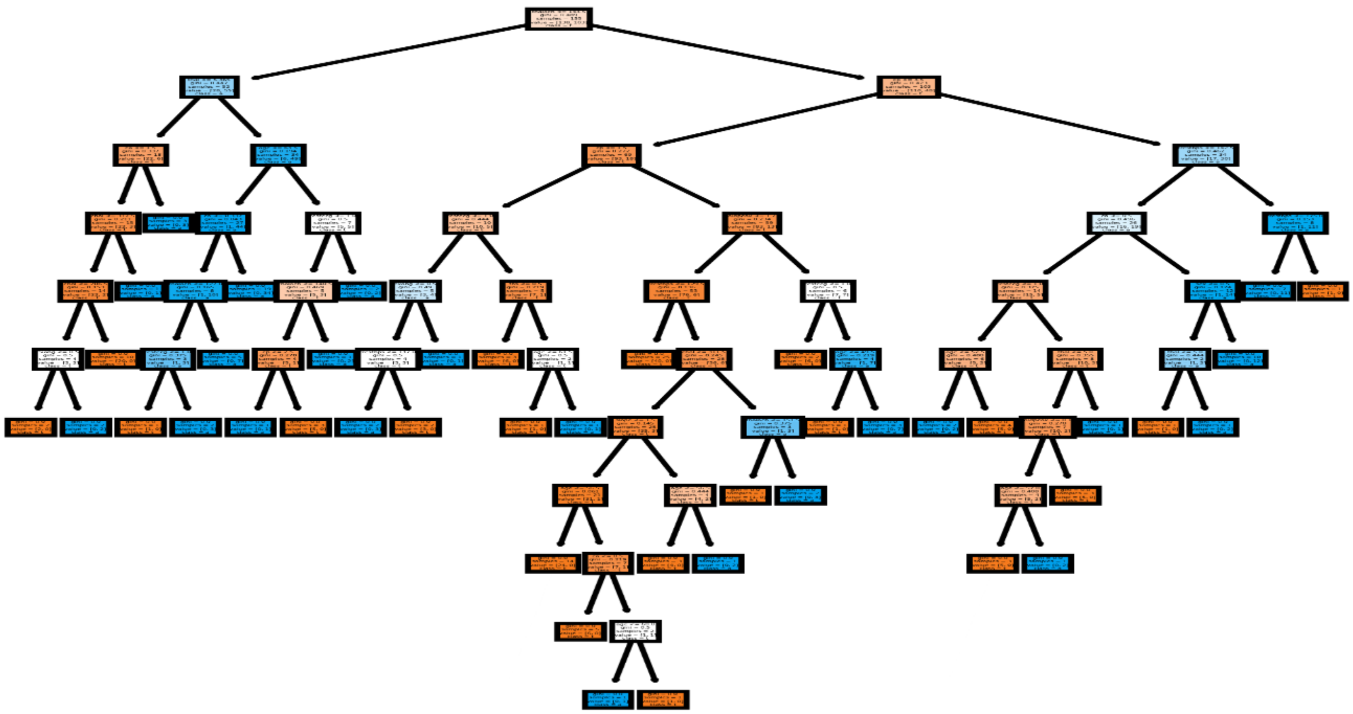


4.1.1. **Random Forest Analysis:**

The random forest model was implemented using Scikit-learn. As expected, a higher accuracy of 79% was achieved with this model.

Here is how the confusion matrix and the first decision tree look like:





4.1.3 **Analysis of Both Models:** When compared to the 77% accuracy obtained with the decision tree model, the random forest model performed better with 79% accuracy. It predicted fewer false positives and false negatives. Root node selection, a critical aspect in decision tree models, revealed distinct preferences between the two algorithms. The decision tree model identified 'thallium scan' as the root node, while the random forest model favored 'max heart rate' as the pivotal root node. This comparative assessment underscores the nuanced variations in model performance and emphasizes the need for tailored considerations when selecting the most appropriate algorithm for heart disease detection.

4.2 **Fuzzy System:** To assess the system's performance and evaluate its predictive capabilities, several test cases were provided as input, and the corresponding health status predictions were observed. This testing methodology allows for a comprehensive examination of the fuzzy logic-based expert system, providing insights into its effectiveness in interpreting various combinations of health-related input parameters. Through the analysis of these test cases, the system's ability to make accurate and meaningful predictions regarding health status can be gauged, contributing to the validation and refinement of the implemented fuzzy logic model for heart disease detection.

In this specific test case, the fuzzy logic-based expert system for heart disease prediction is provided with a set of input parameters representing various health-related features. *The individual's age is set to 75, chest pain level is 8, maximum heart rate is 110, sex is represented as 1 (indicating male), blood pressure is 120, blood sugar level is 1, cholesterol is 180, ECG parameter is 2, old peak rate is 3, thallium scan is 3, and exercise level is 1.*

Upon passing these input values through the fuzzy system, the predicted health status is determined to be 'healthy.' This outcome suggests that, based on the given combination of

health parameters, the individual is assessed as having a favorable health status according to the fuzzy logic rules and membership functions defined in the system. It's important to note that the accuracy and reliability of such predictions are contingent on the soundness of the fuzzy logic rules, the membership functions, and the overall design of the expert system. In this case, the predicted 'healthy' status indicates a positive assessment of the individual's health based on the provided input variables.
The following image shows the output obtained:

Predict results

```
health_prediction.compute()
predicted_health = health_prediction.output['health']
```

```
predicted_health
```

```
2.0
```

Find health status percentage

```
for level, (lower, upper) in health_levels.items():
  if lower <= predicted_health <= upper:
    percentage = ((predicted_health - lower) / (upper - lower)) * 100
    print(f"Predicted Health: {percentage:.2f}% ({level})")
```

```
Predicted Health: 8.00% (healthy)
```

5. **CHALLENGES FACED:**

The development of the fuzzy logic-based expert system for heart disease detection posed several challenges that required careful consideration and resolution. One significant challenge involved defining the appropriate ranges for the membership functions associated with each input variable. This step is crucial as it determines the fuzzy sets' shapes and their influence on the overall system. Selecting accurate and meaningful ranges is essential to ensure the model captures the nuances of the input data effectively.

Fuzzifying and defuzzifying the input presented another obstacle. Achieving a balanced and accurate transformation of crisp input values into fuzzy sets and, subsequently, converting fuzzy output back into a crisp decision demanded precision. The accuracy of these processes significantly impacts the system's ability to make reliable predictions and interpretations.

Building the rule base, an integral aspect of the fuzzy logic system, presented its own set of challenges. While the initial set of rules from the paper provided a foundation, incorporating additional rules to enhance the system's decision-making capabilities required careful consideration. Challenges included determining the appropriate logical conditions, incorporating AND and OR operators judiciously, and ensuring that the expanded rule base contributed positively to the overall system performance.

Overcoming these challenges involved a combination of domain knowledge, iterative testing, and fine-tuning of parameters. Addressing these intricacies was essential to enhance the accuracy, interpretability, and robustness of the fuzzy logic-based expert system for heart disease detection.

6. **CONCLUSION:**

The Heart Disease Detection System using Fuzzy Logic demonstrated promising results in accurately identifying potential cases. The model's interpretability and ease of use make it a valuable tool for healthcare professionals. It presents a promising step towards leveraging computational intelligence for early diagnosis. Continuous improvement and collaboration with the healthcare community are crucial for ensuring its practical utility in the field of cardiology. Continuous refinement and validation with diverse datasets are also necessary to enhance its robustness and generalizability.

7. **FUTURE WORK:**

Fuzzy systems can be employed in remote patient monitoring applications to analyze streaming health data, providing real-time assessments of patients' conditions. Expert systems can interpret these assessments and trigger alerts or interventions when necessary, enhancing proactive healthcare management.

Fuzzy logic and expert systems can streamline the drug discovery process by analyzing complex biological data. They can aid in predicting drug interactions, assessing potential side effects, and optimizing drug formulations, accelerating the development of new therapeutic solutions. Fuzzy logic and expert systems can play a vital role in chronic disease management by providing continuous monitoring, personalized treatment plans, and early detection of exacerbations. This can lead to improved patient outcomes and a reduction in hospitalization rates.

This fuzzy expert system holds promising avenues for future enhancements, paving the way for an even more sophisticated and comprehensive application in the realm of heart disease detection. Several potential areas for improvement and expansion include:

6.1. **Incorporating Additional Features:**

Expanding the dataset to include more diverse parameters and employing advanced feature engineering techniques could improve the model's predictive capabilities.

6.2. **Real-world Validation:**

Collaboration with healthcare institutions for real-world validation is essential to ensure the model's applicability in clinical settings.

6.3. **Integration with Electronic Health Records (EHR):**

Integrating the system with EHR systems can enhance its usability and enable seamless incorporation into existing healthcare infrastructure.

6.4. **Personalized Health Recommendations:**

Moving beyond health status prediction, the system could evolve to provide personalized health recommendations based on individual risk factors. Tailoring advice for lifestyle modifications, dietary changes, or specific exercise regimens could empower individuals to take proactive measures for heart disease prevention.

6.5. **Enhanced Explainability and Interpretability:**
Further efforts can be directed towards improving the interpretability of the fuzzy rule base, making it more understandable for healthcare professionals and patients. Clearer explanations of how input variables contribute to the final health status prediction could foster trust and acceptance of the system in clinical settings.

6.6. **Collaboration with Healthcare Providers:**
Collaborating with healthcare providers for data sharing and validation could strengthen the system's foundation. Inclusion of diverse and extensive datasets, possibly from multiple healthcare institutions, would contribute to a more robust and generalizable model.

8. ACKNOWLEDGEMENTS:

The successful completion of this project was made possible through the support of my professor, Prof. Mohammadreza Hajiarbabi, and my peers.

9. REFERENCES:

i. *Kaur J, Khehra BS. Fuzzy Logic and Hybrid based Approaches for the Risk of Heart Disease Detection: State-of-the-Art Review. J. Inst. Eng. India Ser. B. 2022;103(2):681–97. doi: 10.1007/s40031-021-00644-z. Epub 2021 Aug 2. PMCID: PMC8328141.*
ii. *A. Adeli, M. Neshat, A fuzzy expert system for heart disease diagnosis, in: International Multi Conference of Engineers and Computer Scientists, 1, pp. 134–139 (2010)*
iii. *Muhammad LJ, Algehyne EA. Fuzzy based expert system for diagnosis of coronary artery disease in Nigeria. Health Technol. 2021;11(2):1–11. doi: 10.1007/s12553-021-00531-z*
iv. *W.M. Baihaqi, N.A. Setiawan, I. Ardiyanto, Rule extraction for fuzzy expert system to diagnose coronary artery disease, in: Proceedings of International Conference on Information Systems and Electrical Engineering (ICITISEE), IEEE, pp. 136–141 (2016)*
v. *CDC.gov - Heart Disease Info*
vi. *Heart Disease Dataset - UCI ML Database*