MCTA 3371

COMPUTATIONAL INTELLIGENCE

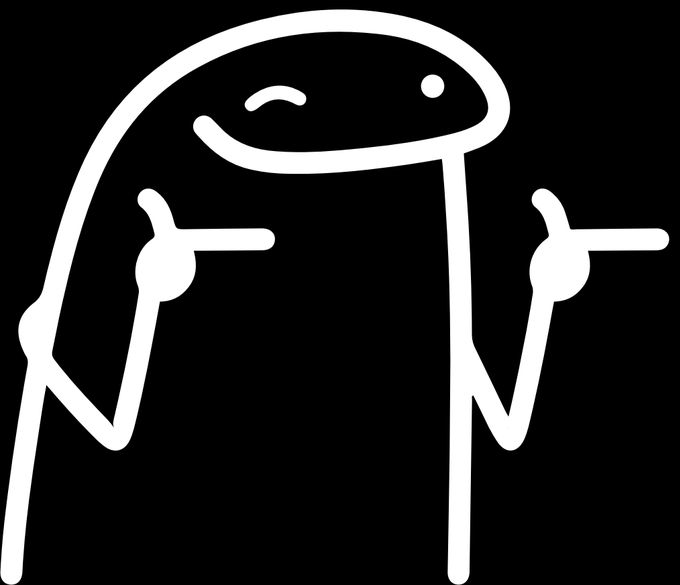
GROUP 2

Eiman 2110565

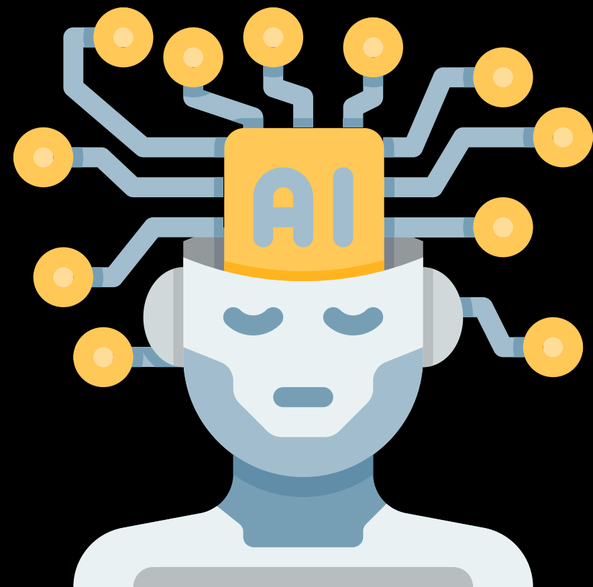
Luqman 2114635

Muhairis 2114599

Zulhilmi 2117679

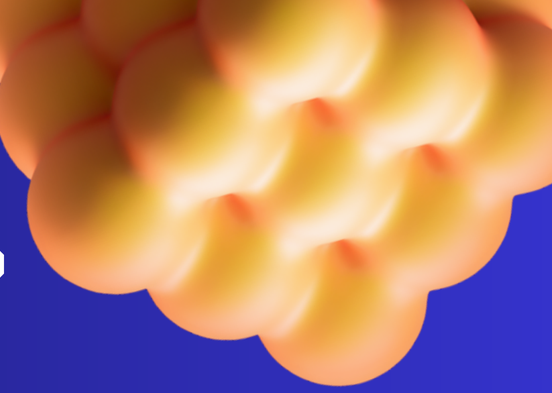


HEART RISK PREDICTION AI



Develop a predictive model using computational intelligence technique to access the likelihood of an individual experiencing heart problems

Fuzzy - Genetic Algorithms



Fuzzy Logic

MATHEMATICAL PRINCIPLES IN DEALING WITH UNCERTAINTY AND IMPERCISION

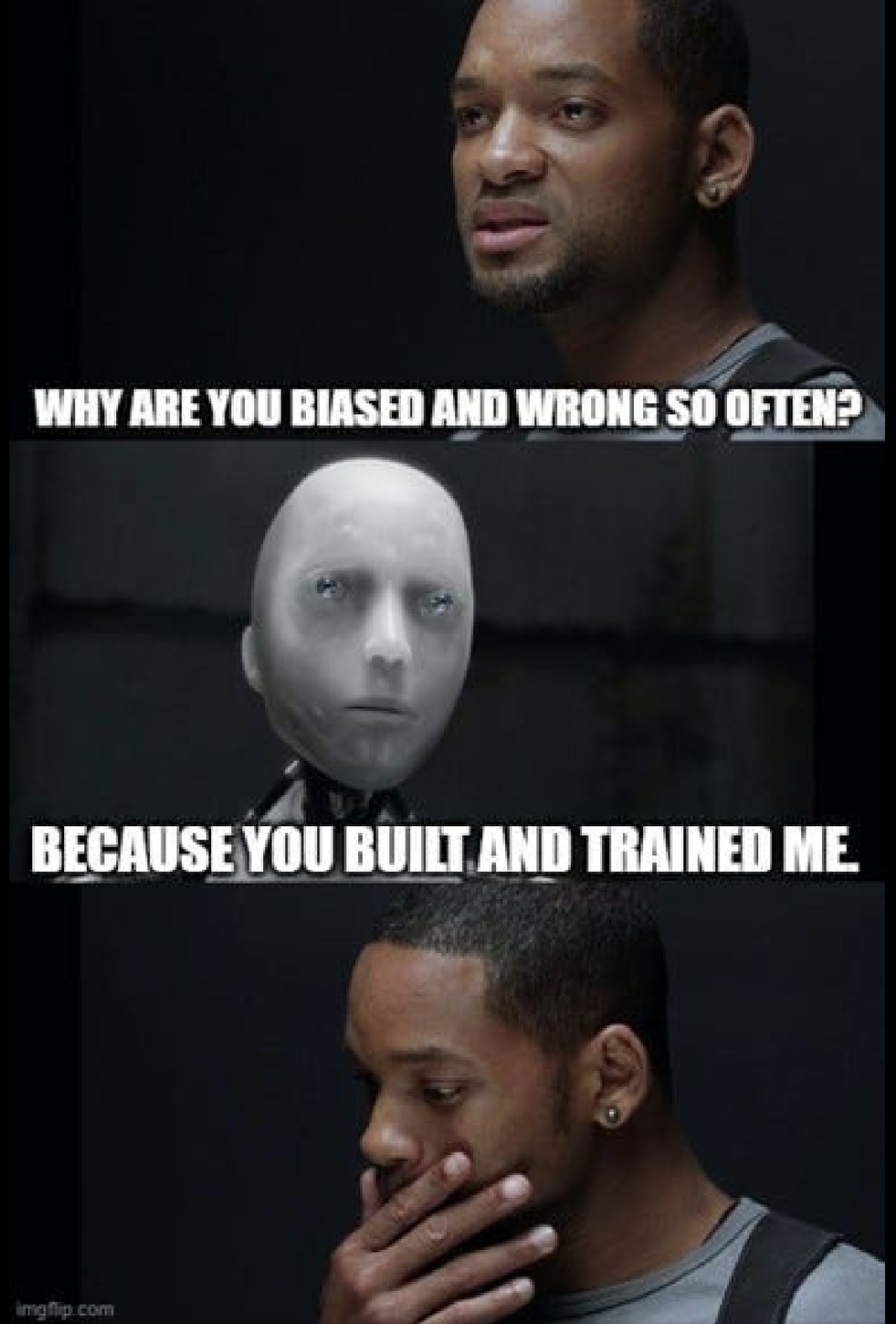
I do my exercise two days a week, do i count as an active or a non-active person?

My BP is not high nor low, a I healthy?

Genetic Algorithms

OPTIMIZATION ALGORITHMS INSPIRED BY PRINCIPLES OF NATURAL SELECTION AND GENETICS

Used in fine tuning the system generating generations of algorithms in order to find the best error and updating the sysrtem accordingly


WHY ARE YOU BIASED AND WRONG SO OFTEN?
BECAUSE YOU BUILT AND TRAINED ME.
imgflip.com

WHY??

Fuzzy - Genetic Algorithms

REASONS

01

Interpretability

Fuzzy logic provides transparency, as doctors need to know why did the AI get that outcome

02

Global Optimization

More data means the better the results will be, different combinations from different inputs will provide for better solutions

03

Adaptability

The genetic algorithm can optimize the system and find the most suitable tunings for the system so it can adapt to the upcoming future

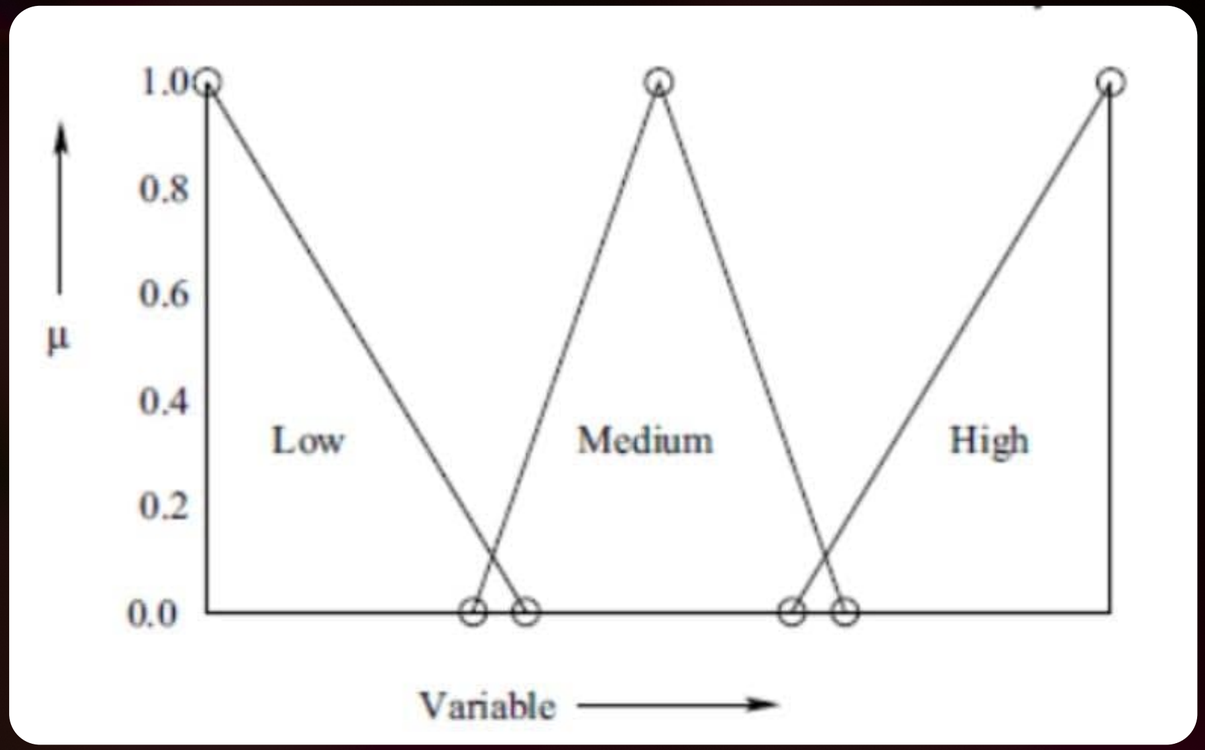
FUZZY IS SO HARD THO!!

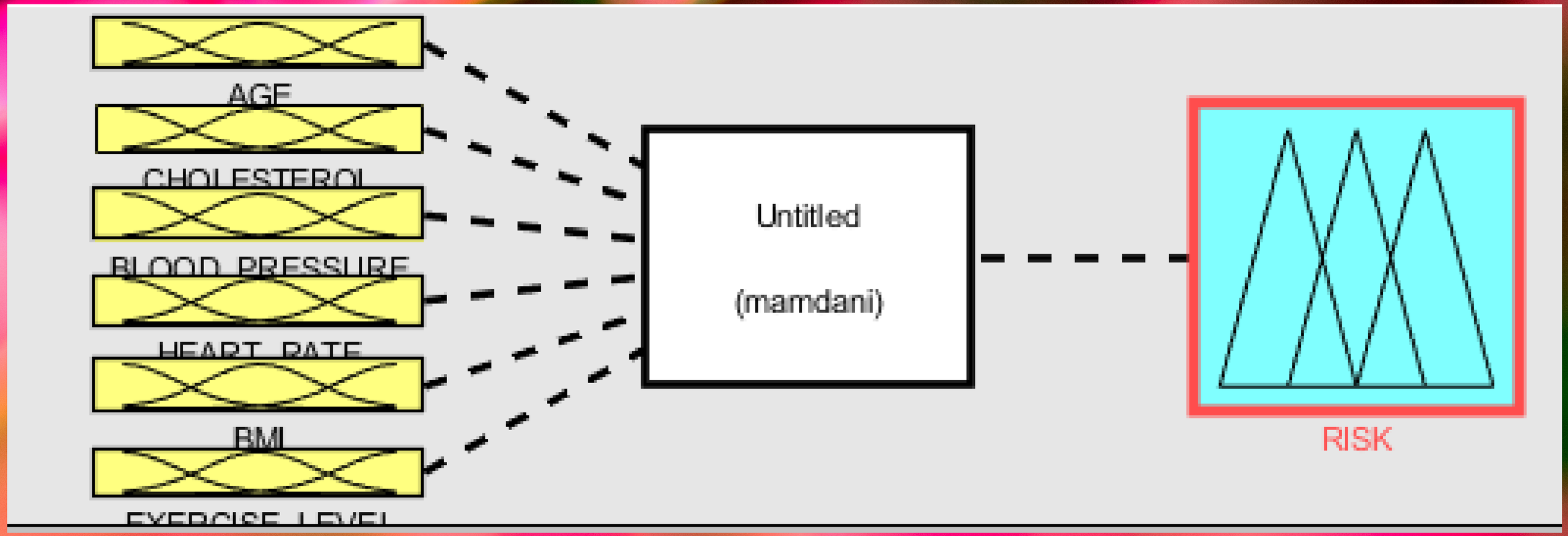
WE USE MAMDANI METHOD

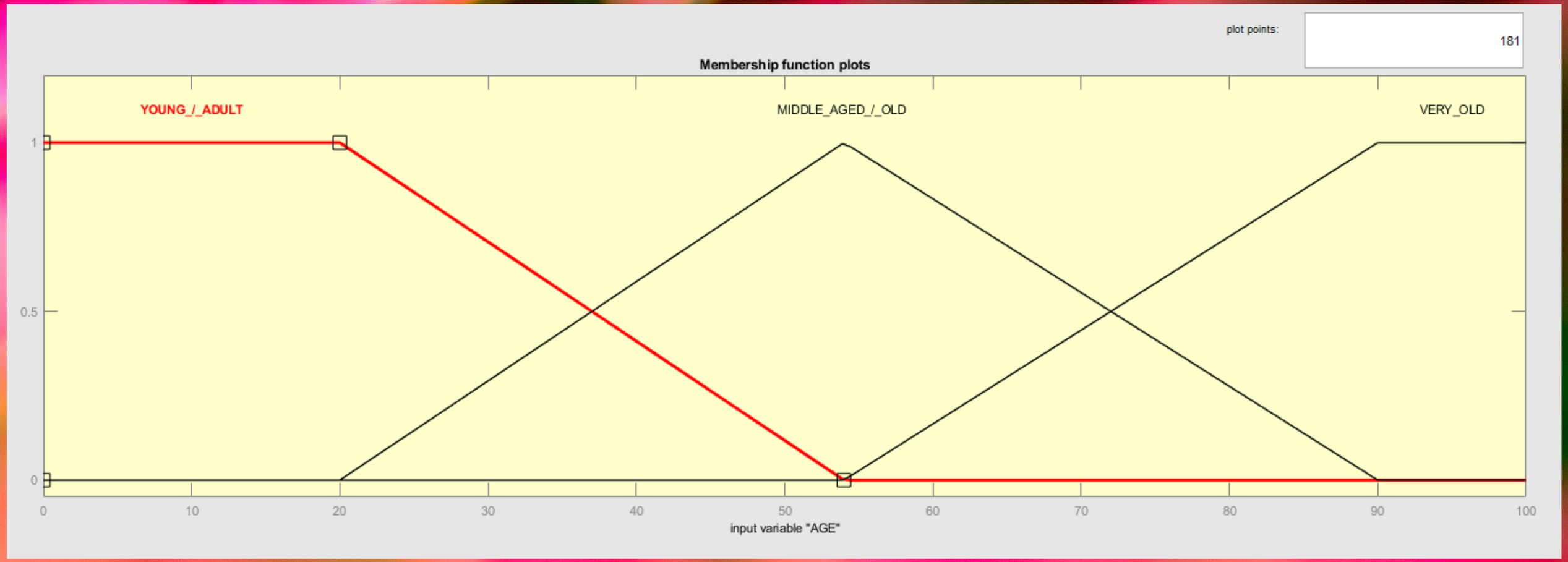
EASILY UNDERSTANDABLE

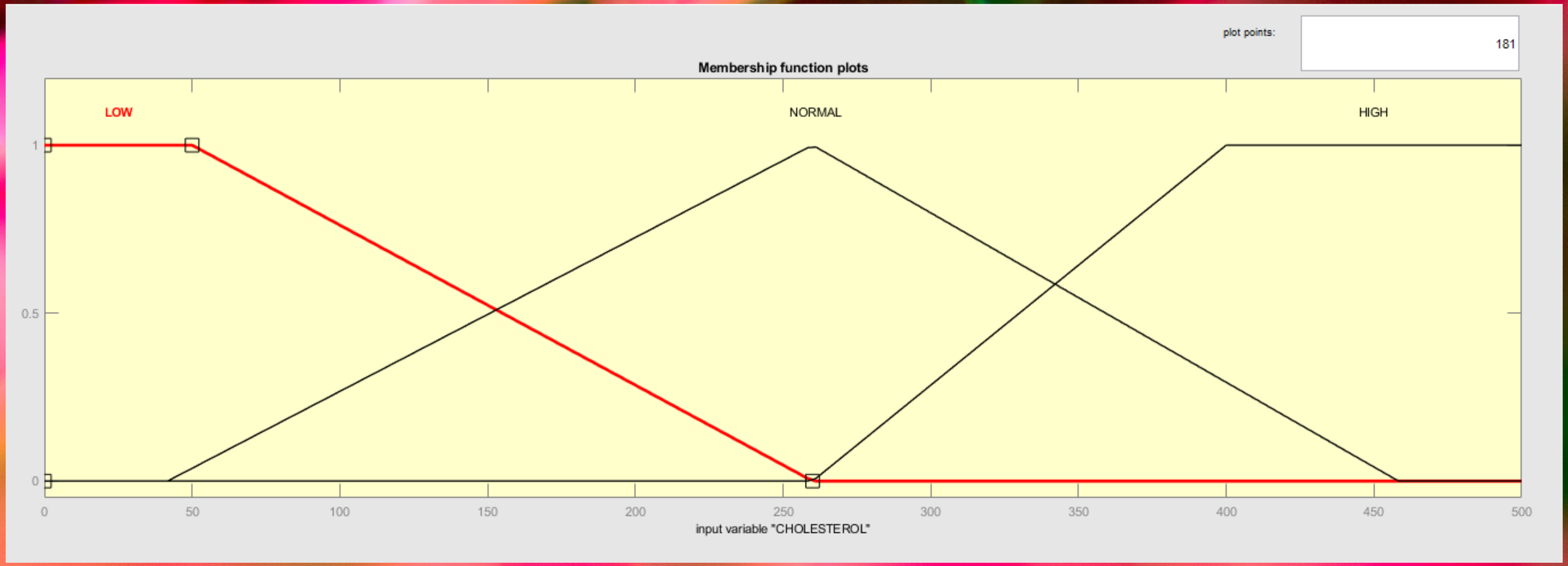
TAKAGI SUGENO METHOD MIGHT BE MORE PRECISE BUT MAMDANI IS DEFINITELY BETTER IN TERMS OF UNDERSTANDABILITY SINCE THE USER MIGHT NOT UNDERSTAND HOW THE AI PRODUCED THE ANSWER

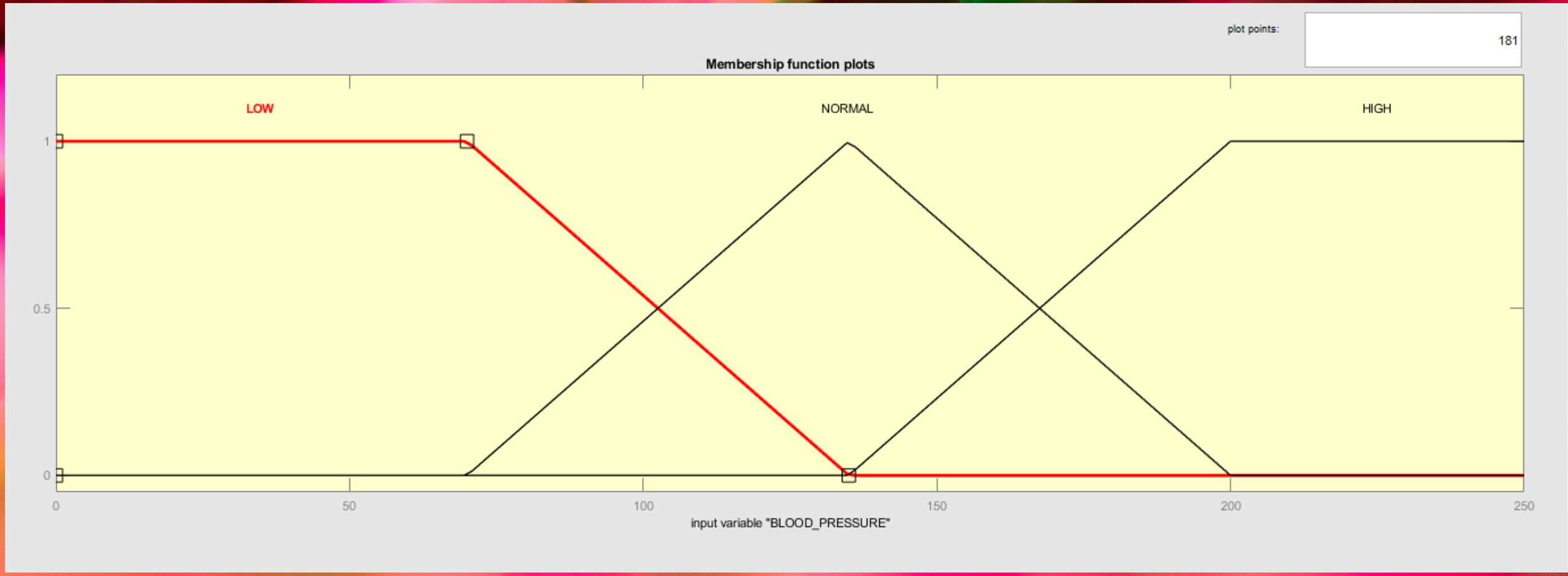
WE CHOSE THIS AS OUR FITNESS FUNCTION

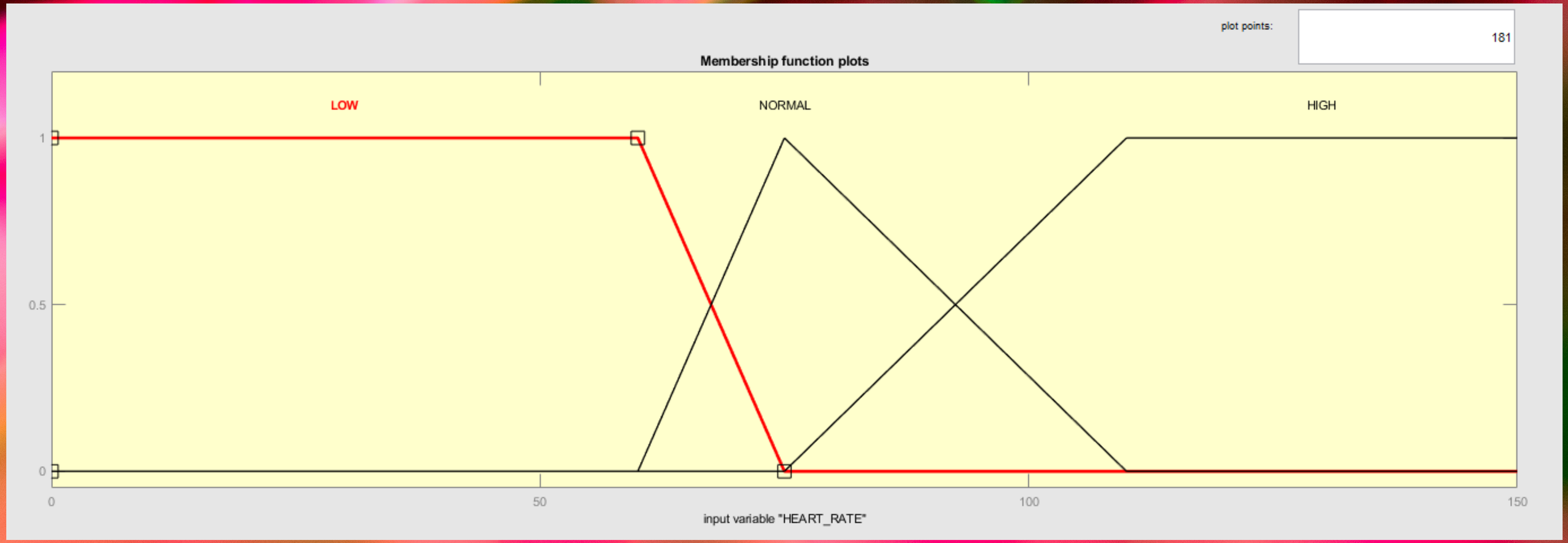


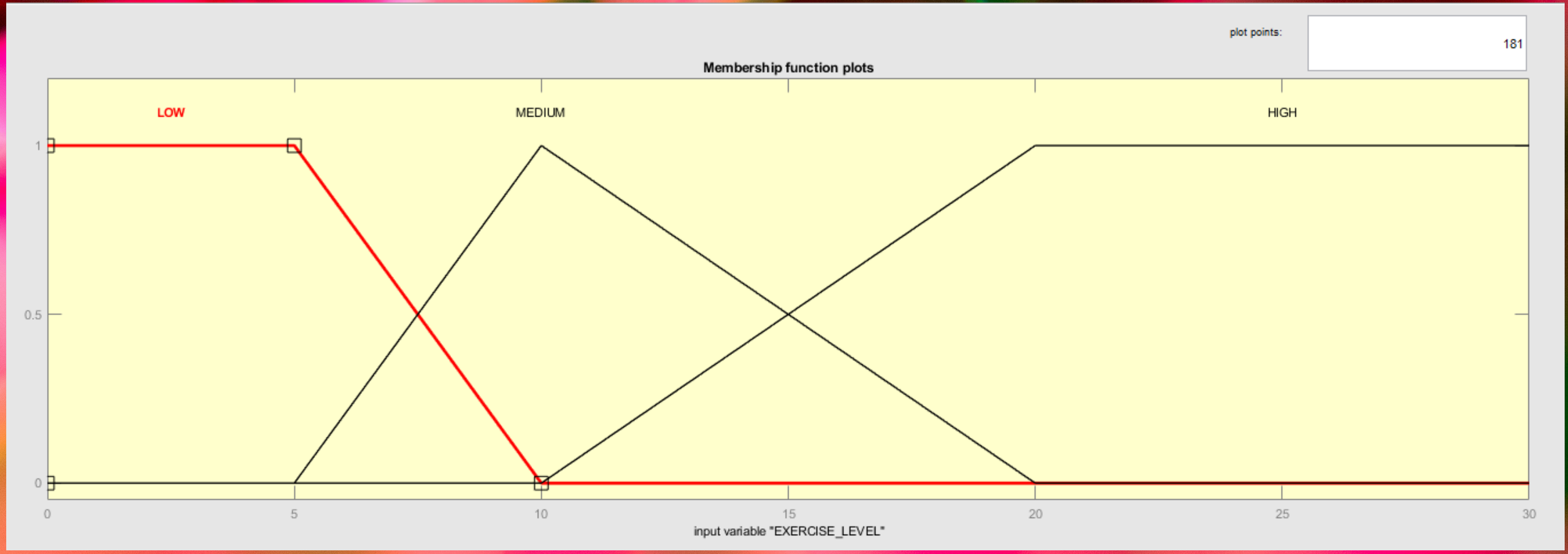

AGE
CHOLESTEROL
BLOOD PRESSURE
HEART RATE
BMI
EXERCISE LEVEL
Untitled
(mamdani)
RISK

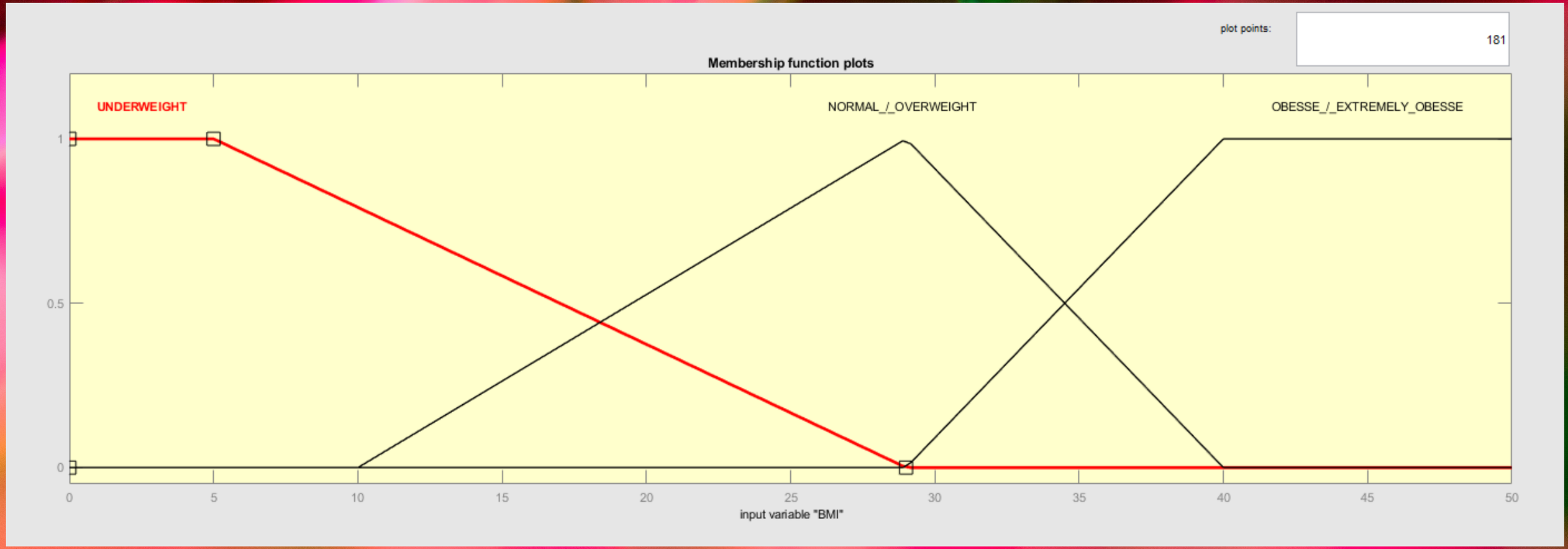

plot points:
181
Membership function plots
YOUNG_/_ADULT
MIDDLE_AGED_/_OLD
VERY_OLD
1
0.5
0
0
10
20
30
40
50
60
70
80
90
100
input variable "AGE"


plot points:
181
Membership function plots
LOW
NORMAL
HIGH
1
0.5
0
0
50
100
150
200
250
300
350
400
450
500
input variable "CHOLESTEROL"


plot points:
181
Membership function plots
LOW
NORMAL
HIGH
1
0.5
0
0
50
100
150
200
250
input variable "BLOOD_PRESSURE"

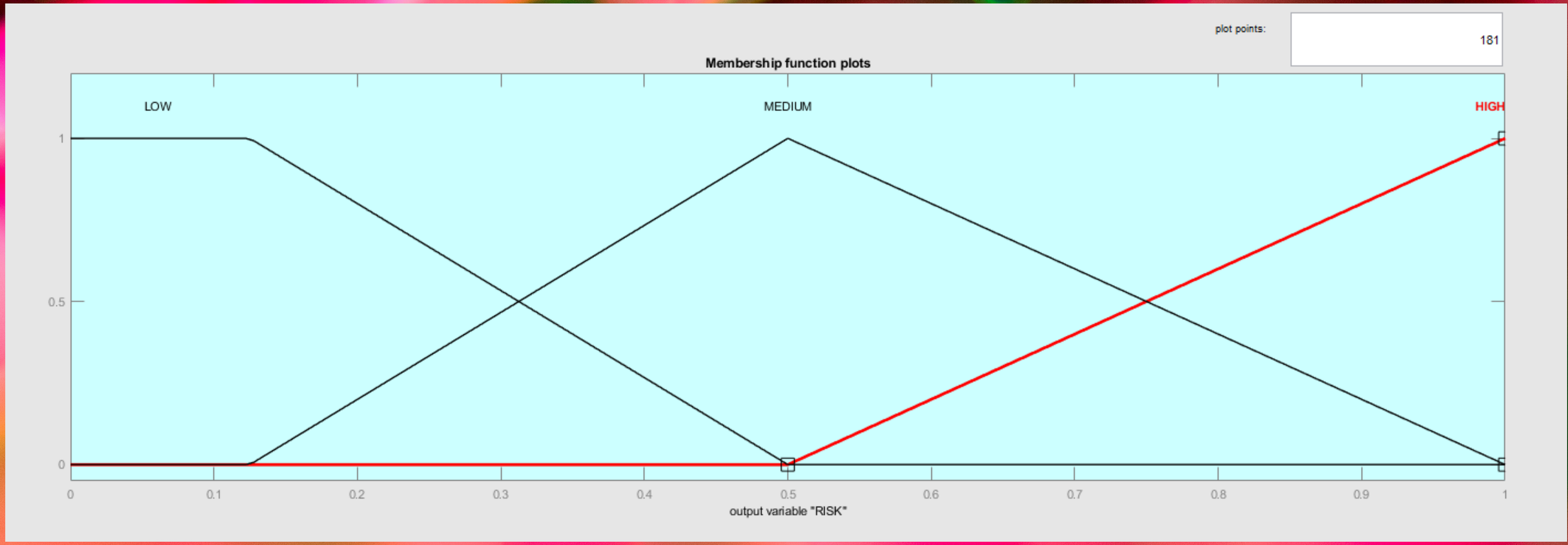

plot points:
181
Membership function plots
LOW
NORMAL
HIGH
1
0.5
0
0
50
100
150
input variable "HEART_RATE"


plot points:
181
Membership function plots
LOW
MEDIUM
HIGH
1
0.5
0
0
5
10
15
20
25
30
input variable "EXERCISE_LEVEL"


plot points:
181
Membership function plots
UNDERWEIGHT
NORMAL_/_OVERWEIGHT
OBESSE_/_EXTREMELY_OBESSE
1
0.5
0
0
5
10
15
20
25
30
35
40
45
50
input variable "BMI"

Rules	Age			Cholestrol level			Blood Pressure			Heart Rate per Minutes		
	young	adult	old	low	medium	high	low	medium	high	low	medium	high
Rule 1						■			■			■
Rule 2												■
Rule 3												
Rule 4			■		■			■			■	
Rule 5					■			■			■	
Rule 6					■			■			■	
Rule 7				■				■				
Rule 8					■		■					
Rule 9												
Rule 10					■			■			■	
Rule 11				■			■			■		
Rule 12												
Rule 13					■			■			■	
Rule 14					■			■			■	
Rule 15					■			■			■	
Rule 16	■											
Rule 17		■										
Rule 18			■									

Exercise Hours Per Week			BMI			Heart attack risk		
low	medium	high	low	medium	high	low	medium	high


plot points:
181
Membership function plots
LOW
MEDIUM
HIGH
1
0.5
0
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
output variable "RISK"

DONE FUZZY PART!

EASY NO?

Process

01

- Begin with six input variables and 5000 data points.
- Implement a Fuzzy Genetic Algorithm (GA).

02

-Group subjects into batches of 10 and assign random sets of membership functions.
- Generate an initial population consisting of binary strings of length randomly

03

-Test 10 subjects using six parameters following recommendations from a website.
- Fuzzify the inputs by finding membership functions.
- Defuzzify the output.

04

-For each of the 10 subjects, calculate the average of 10 outputs.
- Find the average target and calculate the error by subtracting the output modulus.

Process

05

- Find the average target and calculate the error by subtracting the output modulus.
- Use the error to calculate the fitness function, aiming to minimize it.

06

-Repeat the process for 500 iterations.
- Select the best 100 fitness values and place them in a mating pool.

07

-Apply a 10% mutation rate to individuals in the mating pool.
- New parameters change the membership functions, updating them accordingly..

08

-With 6 membership functions, continue this process for 1000 generations.
- Aim to achieve the lowest error possible to determine accuracy.

Result of GA

```
=== Gen 15 best solutions ===
(5.982454593088228, (35, 175, 33.418535490844675, 6.368430995446502, 1.5848085944652792, 14))
=== Gen 16 best solutions ===
(5.982454593088228, (35, 175, 29.175170548619707, 23.26305589099745, 1.5895030079282568, 14))
=== Gen 17 best solutions ===
(5.982454593088228, (35, 175, 32.766348687400715, 20.685417123667246, 1.5777715411601108, 14))
=== Gen 18 best solutions ===
(5.982454593088228, (34.70741356565124, 175, 29.588065935223046, 7.166004910089814, 1.5782382809974802, 14))
=== Gen 19 best solutions ===
(5.982454593088228, (35, 175, 28.569316374823856, 22.231746126566144, 1.5964251807353091, 14))
=== Gen 20 best solutions ===
(5.982454593088228, (35, 175, 28.033227917509603, 25, 1.565982535574482, 14))
```

BY GEN 20, THE FITNESS FUNCTION SATURATED TO 5.982454593088228

FINDINGS

By testing 1000 subjects, we got 54.9% accuracy
For 3000 subjects, we got 53.8% accuracy

OUR MODEL HAS THE ACCURACY AROUND 53 - 55% accuracy


THANK YOU
FOR ATTENDING AND CONFIDENTIALITY