CS 514
Applied Artificial Intelligence
Project 2

Fuzzy Fitness Star

A Personal Fitness Assistant Fuzzy Expert System

INDEX

Abstract:

Fitness Star is a fuzzy based expert system built on FuzzyJ that is designed to take certain health related parameters from the user and infer certain vital parameters and makes recommendations to the user of the system. The system is a prototype which classifies the user as underweight, normal or overweight. The system also diagnoses diabetes, chances of coronary heart diseases, recommends workout and also the stress levels.

Features:

1. The system can calculate parameters like Body Mass Index on the basis of height and weight.
2. The system can classify the individual into different obesity fitness levels.
3. The system is also capable to identifying stress based on hypertension symptoms and sleep patterns.
4. Based on obesity indices and stress levels the system also estimates risk of Coronary heart diseases.
5. The system also infers chances of diabetes.
6. In case the system finds users with extreme symptoms, the system recommends the user to consult a physician immediately.

Rules and descriptions:

#	Rule Name	Description
1	`init-FuzzyVariables`	Initialize all the fuzzy variables for the expert system to reason
2	`init`	Starting point of execution. This rule will create all the basic facts in the working memory.
3	`fuzzify_bmi`	Based on the height and the weight of the user, calculate the BMI and fuzzify it. The inputs are crisp set and the output is a fuzzy set as well.
4	`mod_workout_required`	Based on the BMI recommend workout type. For people with “normal” BMI will have moderate workout routine. The inputs are fuzzy and the output is a fuzzy set as well.
5	`extrm_workout_required`	Based on the BMI recommend workout type. For people categorized as “overweight” will have an intense workout routine. The inputs are fuzzy and the output is a fuzzy set as well.
6	`stress_lvls_high`	Based on user input about the sleep patterns, diastolic and systolic blood pressures, infer stress of the individual. The inputs are fuzzy and the output is a fuzzy set as well.
7	`chd_risks_high`	A person with high stress and who is over weight is prone to coronary heart diseases. The inputs are fuzzy and the output is a fuzzy set as well.
8	`chd_risks_mod`	A person with “more or less high” stress levels and is not over weight will have “moderate” chances of coronary heart diseases. The inputs are fuzzy and the output is a fuzzy set as well.
9	`diabetes_chck`	Calculate chances of diabetes based on the sugar level that is taken as an input from the user. The inputs are crisp sets and the output is a fuzzy set as well.
10	`disImmediateDiagnosisHeartCondn`	For users who have both high chances of coronary heart conditions as well as high chances of diabetes, the system recommends consulting a specialist as soon as possible. The inputs are fuzzy and the output is a fuzzy set as well.

Usage Manual:

Instructions:

Create a new Java project in eclipse. Make sure you include the JAR file “fuzzyJ-2.0.jar” under New Project > Libraries.

Copy the contents of the files “fuzzy_health.clp” into a new text file under the newly created Java project. Save the file with extension “.clp”.

Run the project after setting up the correct run configurations.

In case you run into errors, make sure that the run as configuration is pointing to the right .main class.

By default, the main class will be jess.main. For the fuzzy parts to execute it should point to nrc.fuzzy.jess.FuzzyMain.

In case the grader wants to change inputs and test new inputs please only make changes to the rule init.

Format of the input for the init rule:

(assert (person (name Batman)(height 1.6256)(weight 90))) ;

*(assert (person_sleep (new nrc.fuzzy.FuzzyValue ?*sleepVar* "poor")))*

*(assert (person_bp_sys (new nrc.fuzzy.FuzzyValue ?*bloodPressSVar* "hypertension")))*

*(assert (person_bp_dias (new nrc.fuzzy.FuzzyValue ?*bloodPressDVar* "hypertension")))*

(assert (sugar-level (level 100)))

Allowed input values in the init rule

person:

name: <Any valid string>

height: <height of the user in centimeters>

weight: <Weight of the user in kilograms>

person_sleep: <poor,normal>

person_bp_sys: <low,hypertension,ideal>

person_bp_dias: <low,hypertension,ideal>

sugar-level : <any valid positive float in the range 30 to 200>

Sample runs:

Run #1

```
; Rule 2 - Initialize the knowledge base
(defrule init
    (declare (salience 50))
=>
    ;; write assert statement here, try forming an interactive prompt
    (assert (person (name Batman)(height 1.6256)(weight 90))) ;
    (assert (person_sleep (new nrc.fuzzy.FuzzyValue ?*sleepVar* "poor")))
    (assert (person_bp_sys (new nrc.fuzzy.FuzzyValue ?*bloodPressSVar* "hypertension")))
    (assert (person_bp_dias (new nrc.fuzzy.FuzzyValue ?*bloodPressDVar* "hypertension")))
    (assert (sugar-level (level 100)))
)
```

Problems @ Javadoc Declaration Console

<terminated> FuzzyHealth [Jess Application] C:\Program Files\Java\jdk1.7.0_79\bin\javaw.exe (17-Feb-2016 9:11:15 AM)

```
Jess, the Rule Engine for the Java Platform
Copyright (C) 2008 Sandia Corporation
Jess Version 7.1p2 11/5/2008

This copy of Jess will expire in 29 day(s).
You have high chances of diabetes.
With your poor sleeping pattens and hypertension you are diagnosed with very high stress.
Your stress is to the degree of 1.0
Your BMI is calculated as 34.05768530287061
With very high stress and obesity there are high chances of coronory heart diseases.
With such high chances of both diabetes and coronary heart conditions
I recommend you to consult your physician in order for proper medication if required.

Based on your BMI you need an extreme workout.
Rules fired is: 8


Testing with:
   Rule Executor = mamdanimin
   Antecedent Combine Operator = minimum
   Global Contribution Operator = union
```

Run #2

FuzzyHealth.clp

```
; Rule 2 - Initialize the knowledge base
(defrule init
    (declare (salience 50))
=>
    ;; write assert statement here, try forming an interactive prompt
    (assert (person (name Batman)(height 1.8256)(weight 60))) ;
    (assert (person_sleep (new nrc.fuzzy.FuzzyValue ?*sleepVar* "normal")))
    (assert (person_bp_sys (new nrc.fuzzy.FuzzyValue ?*bloodPressSVar* "ideal")))
    (assert (person_bp_dias (new nrc.fuzzy.FuzzyValue ?*bloodPressDVar* "ideal")))
    (assert (sugar-level (level 50)))
)
```

Problems @ Javadoc Declaration Console

<terminated> FuzzyHealth [Jess Application] C:\Program Files\Java\jdk1.7.0_79\bin\javaw.exe (17-Feb-2016 9:13:44 AM)

```
Jess, the Rule Engine for the Java Platform
Copyright (C) 2008 Sandia Corporation
Jess Version 7.1p2 11/5/2008

This copy of Jess will expire in 29 day(s).
You have low chances of diabetes.
Your BMI is calculated as 18.002797490707678
Based on your BMI you need a normal workout.
Rules fired is: 5


Testing with:
   Rule Executor = mamdanimin
   Antecedent Combine Operator = minimum
   Global Contribution Operator = union
```

Run #3

FuzzyHealth.clp

```
; Rule 2 - Initialize the knowledge base
(defrule init
    (declare (salience 50))
=>
    ;; write assert statement here, try forming an interactive prompt
    (assert (person (name Batman)(height 1.8256)(weight 60))) ;
    (assert (person_sleep (new nrc.fuzzy.FuzzyValue ?*sleepVar* "poor")))
    (assert (person_bp_sys (new nrc.fuzzy.FuzzyValue ?*bloodPressSVar* "ideal")))
    (assert (person_bp_dias (new nrc.fuzzy.FuzzyValue ?*bloodPressDVar* "ideal")))
    (assert (sugar-level (level 50)))
)
```

Problems @ Javadoc Declaration Console

<terminated> FuzzyHealth [Jess Application] C:\Program Files\Java\jdk1.7.0_79\bin\javaw.exe (17-Feb-2016 9:16:05 AM)

```
Jess, the Rule Engine for the Java Platform
Copyright (C) 2008 Sandia Corporation
Jess Version 7.1p2 11/5/2008

This copy of Jess will expire in 29 day(s).
You have low chances of diabetes.
With your poor sleeping pattens and hypertension you are diagnosed with very high stress.
Your stress is to the degree of 0.25
Your BMI is calculated as 18.002797490707678
With more or less high stress there are moderate chances of coronory heart diseases.
Based on your BMI you need a normal workout.
Rules fired is: 7


Testing with:
   Rule Executor = mamdanimin
   Antecedent Combine Operator = minimum
   Global Contribution Operator = union
```

Test cases:

Please use only one test case at a time in the init rule:

Test case #1:

```
(assert (person (name Batman)(height 1.6256)(weight 90))) ;

(assert (person_sleep (new nrc.fuzzy.FuzzyValue ?*sleepVar* "poor")))

(assert (person_bp_sys (new nrc.fuzzy.FuzzyValue ?*bloodPressSVar*
"hypertension")))

(assert (person_bp_dias (new nrc.fuzzy.FuzzyValue ?*bloodPressDVar*
"hypertension")))

(assert (sugar-level (level 100)))
```

Test case #2:

```
(assert (person (name Batman)(height 1.8256)(weight 60))) ;

(assert (person_sleep (new nrc.fuzzy.FuzzyValue ?*sleepVar* "normal")))

(assert (person_bp_sys (new nrc.fuzzy.FuzzyValue ?*bloodPressSVar* "ideal")))

(assert (person_bp_dias (new nrc.fuzzy.FuzzyValue ?*bloodPressDVar*
"ideal")))

(assert (sugar-level (level 50)))
```

Test case #3:

```
(assert (person (name Batman)(height 1.8256)(weight 60))) ;

(assert (person_sleep (new nrc.fuzzy.FuzzyValue ?*sleepVar* "poor")))

(assert (person_bp_sys (new nrc.fuzzy.FuzzyValue ?*bloodPressSVar* "ideal")))

(assert (person_bp_dias (new nrc.fuzzy.FuzzyValue ?*bloodPressDVar*
"ideal")))

(assert (sugar-level (level 50)))
```