



*Research article*

## **A Clinical Decision Support System based on fuzzy rules and classification algorithms for monitoring the physiological parameters of type-2 diabetic patients**

**Colella Ylenia<sup>1</sup>, De Lauri Chiara<sup>1</sup>, Improta Giovanni<sup>2,3</sup>, Rossano Lucia<sup>1</sup>, Vecchione Donatella<sup>1</sup>, Spinosa Tiziana<sup>4</sup>, Giordano Vincenzo<sup>4</sup>, Verdoliva Ciro<sup>4</sup> and Santini Stefania<sup>1,\*</sup>**

<sup>1</sup> Department of Electronic Engineering and Information Technology, Faculty of Engineering, University of Naples Federico II, Naples, Italy

<sup>2</sup> Department of Public Health of the University Hospital, University of Naples Federico II, Naples, Italy

<sup>3</sup> Interdepartmental Center for Research in Health Management and Innovation in Health (CIRMIS), University of Naples Federico II, Naples, Italy

<sup>4</sup> ASL Napoli 1 Centro, Naples, Italy

\* **Correspondence:** Email: [stefania.santini@unina.it](mailto:stefania.santini@unina.it); Tel: +390817683914.

**Abstract:** The use of different types of Clinical Decision Support Systems (CDSS) makes possible the improvement of the quality of the therapeutic and diagnostic efficiency in health field. Those systems, properly implemented, are able to simulate human expert clinician reasoning in order to suggest decisions on treatment of patients. In this paper, we exploit fuzzy inference machines to improve the quality of the day-by-day clinical care of type-2 diabetic patients of Anti-Diabetes Centre (CAD) of the Local Health Authority ASL Naples 1 (Naples, Italy). All the designed functionalities were developed thanks to the experience on the field, through different phases (data collection and adjustment, Fuzzy Inference System development and its validation on real cases) executed by an interdisciplinary research team comprising doctors, clinicians and IT engineers. The proposed approach also allows the remote monitoring of patients' clinical conditions and, hence, can help to reduce hospitalizations.

**Keywords:** Fuzzy Inference System; diabetes; health status; expert system; classification algorithms

---

## 1. Introduction

The healthcare sector is currently facing significant economic and organizational problems due to the continuous increase of patients with chronic diseases: indeed, treatment targets in patients with chronicity require continuity of care aimed at improving their clinical status as they cannot be aimed at healing. This affects the health expenditure so strongly that in Italy chronic disease accounts for approximately 80% of the health care spending. In this context, diabetic disease represents a great concern. Diabetes is one of the main pathologies causing death and at the same time implies a great incidence in national health expenditure, especially because of the stringent need for patient monitoring. Moreover, these diseases more easily result in comorbidity and for this reason more attention needs to be paid to their monitoring. In Italy, diabetes affects more than 3.5 million people, about 5.5% of the general population. Current epidemiological data show that 1 person out of 3 affected by diabetes is older than 65 [1]. Therefore, the aim of health systems is to ensure continuity of medical services reducing their economic impact. This problem has led to the search for alternative models to the traditional patients care approach. Telemedicine represents a new way of providing health care services, through the distribution of health-related services and information via electronic information and telecommunication technologies. In particular, telemonitoring services using wearable sensors exploit the potential of the Internet of Things (IoT) to ensure an effective long-term medical assistance because they allow to extract physiological information of patients to assist the doctor in the decision-making process and to manage the disease remotely [2].

### 1.1. State of the art

In literature, different decision-making support tools can be found, and most of them were based on IT programmes and algorithms to analyse and elaborate biomedical data and signals [3]. The clinical parameters detected by the telemonitoring platforms must be processed to obtain meaningful and useful results for doctors and for the patients himself. The processing phase requires the introduction of algorithms that allow real-time processing of clinical parameters; in this way it is possible to detect pathologies and constantly monitor patients in chronic conditions, reducing the risk of serious damage deriving from non-timely healthcare, and to act promptly in case of critical issues [4]. CDSS, equipped with classification techniques or methodologies like Neural Network, Naïve Bayes, Support Vector Machine, have shown their ability in helping the diagnosis of pathologies. A problem that arises in the management of patients is extracting all the risk factors and building a treatment plan based on overall conditions of patients. To deal with uncertainty in health monitoring, it is necessary to refer to a methodology that allows automatically to take vital parameters and to calculate the risk value relating to the health condition to trigger if necessary, an alarm state [5]. A Fuzzy Inference System (FIS), obtained by combining the potential of Fuzzy Logic (FL) and Expert Systems, gives the possibility of patients' health conditions monitoring by modelling the uncertainty and ambiguity that characterize clinical data and by reproducing cognitive process of experts through inferential techniques [6,7]. In the state-of-art, many researchers apply classification algorithms and FIS to support care pathway planning starting from the evaluation of their physiological parameters, as shown in Table 1.

**Table 1.** Key features of the state-of-the-art studies.

Study	Research Goals	Classification Algorithms	FIS
[8]	Detection of Diabetes Mellitus to decrease the death rates	✓ SVM	–
[9]	Comparative study between algorithms to detect diabetes	✓ SVM ✓ Random Forest ✓ Decision Tree ✓ K-Nearest Neighbour	–
[10]	Early detection of chronic diseases related to kidney using clinical data	✓ SVM ✓ Decision Tree ✓ Logistic Regression	–
[11]	Detection of normal and abnormal regions in liver cancer	✓ Neural network classifier ✓ Multinomial multivariate Bayesian classifier	–
[12]	Clinical care of $\beta$ thalassemia patients of the Rare Red Blood Cell Disease Unit	-	2 inputs
[13]	Follow up of the transplanted patients due to renal pathology	-	2 inputs
[14]	Determination of coronary heart disease risk of patients	-	4 inputs
[15]	Classification of the conditions of ambulatory clinical parameters of patients and sending of alerts according to the pre-diagnosis	-	5 inputs
Our system	Classification of type-2 diabetic patients and monitoring of their physiological parameters to detect a critical condition	✓ Random Forest Classifier ✓ Multilayer Perceptron Classifier (Neural Network) ✓ Naïve Bayes ✓ SVM	5 inputs

CDSS, equipped with classification techniques or methodologies like Neural Network, Naïve Bayes, Support Vector Machine, have shown their ability in helping the diagnosis of diabetes and their type [16–19]. Among the different knowledge-based approaches, fuzzy-based CDSSs can be an effective diagnostic support for identifying these diseases and suggesting the actions to be followed depending on the disease severity [20,21]. However, commonly these systems require specific pathological tests for acquiring determined clinical parameters.

Differently from other studies, here we propose a complete CDSS, based on both FIS and classification algorithms, that recognizes subjects with type-2 diabetes between healthy and pathological ones with a high accuracy and, for the diabetic patients, can assess and monitor the health status by leveraging physiological parameters that can be acquired at home by commercial medical devices during daily care routines. The main advantage is that these parameters are easy to recover and allow a complete characterization of the clinical picture of the pathological subject before a clinical deterioration occurs. In so doing, the proposed approach allows the remote monitoring of patients' outcomes and clinical conditions and the system impacts on the containment of health expenditure thanks to the reduction of hospitalizations. Furthermore, note that the system knowledgebase is built

in agreement with information provided by medical researchers, thus obtaining efficient inference rules according to the medical standards.

### *1.2. Fuzzy Inference System*

A Fuzzy Inference System (FIS) is an intelligent system that allows to reproduce the ability of the human mind to approximate vague data, to extract from them useful information and to produce crisp output [22]. Its potential can be applied to numerous domains and particularly to the medical field to model the high complexity and uncertainty that characterizes medical processes [23,24]. Starting from the definition of the system knowledge-base, built through the help of doctors and clinical experts, a FIS gives the possibility to transfer human and expert knowledge into intelligent and automatic models using linguistic terms [25]. Fuzzy sets are used to treat uncertainty and to represent knowledge through rules since the Fuzzy Logic allows the interpretation of data with predefined linguistic variables according to appropriate IF-THEN rules written as:

IF situation THEN conclusion

where the situation represents the antecedent or the premise consisting of fuzzy terms connected by fuzzy operators, while the output is called consequent or conclusion [26]. Fuzzy logic defines the inferential mechanisms needed for reaching the output value related to the clinical status of type-2 diabetic patients starting from their physiological parameters and constitutes the inferential engine of the Fuzzy Inference System (FIS).

## **2. Methods**

This section provides the main concepts on which is based the processing algorithm created to identify the level of risk related to the health status of patients. To perform the data processing Colab platform, a Cloud computing environment that allows to use Python as a programming language, is used to create a script that automatically detects significant patterns in the patients' dataset and to make predictions on the future trend of clinical parameters. With respect to the diagnosis of type-2 diabetic patients, the capabilities of four different classification algorithms are exploited: a Random Forest Classifier, a Neural Network based on a Multilayer Perceptron Classifier, a Naïve Bayes Network and a Support Vector Machine (SVM). The choice of these algorithms was mainly driven by the intention of increasing the code quality and optimizing the performances of the learning operations on the dataset provided. Indeed, the classifiers used are all in the MLlib library, a machine-learning library created by Apache Software Foundation. The available data mining algorithms that allow realizing the multiclassification and a tuning operation in the parameters of the classifiers and that have vastly different designs are selected. The developed FIS system takes advantage from the Mamdani-type FIS, proposed by Professor Ebrahim Mamdani in 1975, which represents the method currently most used for the Fuzzy Systems design due to its simple structure [27]. As the implementation of the data mining architecture, Fuzzy Inference System takes advantage from the Colab platform and Python language. The Data Science libraries and tools Numpy, SciKit-Fuzzy and Matplotlib are imported and installed to perform the algorithm. The clinical variables exploited for the implementation of the FIS are the Systolic Blood Pressure (SBP), Heart Rate (HR), Temperature (T), Oxygen Saturation (SPO2) and Blood Sugar (BS), since, according to previous medical knowledge, they are the most significant for

determining the current health status of a generic diabetic patient [28]. Once implemented, the FIS must be validated on medical data related to real clinical condition of patients for testing the effectiveness of its real-life use. Indeed, the final use can also be tele-monitoring in home health care, where the FIS receives, as real-time inputs, the values of vital parameters coming from wearable sensors applied to the patient. Hence, to this final aim it is crucial that predictions and identification of the health risk level are always reliable and consistent with the actual user condition.

### 2.1. Type-2 diabetic patients' classifier

To obtain a classification, the information regarding some features of the objects in question are used, and compared, in a special multidimensional space, with those of a training set. A formalization of the classification problem can be the following: starting from a series of training data  $\{(x_1, y), (x_2, y), \dots, (x_n, y)\}$  we must produce a classifier  $h: X \rightarrow Y$  that maps new elements  $x \in X$  on the labels  $y \in Y$ . In this work four classifiers are used to identify subjects with diabetes: Random Forest, Multilayer Perceptron, Naïve Bayes and SVM. To test the goodness of the classification model and the precision value achieved, the dataset is randomly divided into two parts, respectively collecting 80% of the total data in trainingData, and in the testData the remaining 20%. The classifier training phase is performed on the trainingData set, while the test phase is on the testData set. Each classifier categorizes the incoming tuple according to the model learned during the training phase. The accuracy value achieved by each classifier could be influenced by the specific distribution of the dataset between trainingData and testData. The value recorded could be due to luck and therefore not representative of the model's level of goodness. To confirm that the accuracy value is not sporadic, but that it is the value of accuracy achieved by the classifier, a ten-fold cross-validation is performed.

#### 2.1.1. Dataset preparation

In this work, the statistical sample is selected from the data in the paper medical records stored at the Anti-Diabetes Centre (CAD) of the Local Health Authority ASL Naples 1, District 25 from the year 2004 to 2019. The collected data do not include sensitive, identification and personal data, to comply with the provisions of the Legislative Decree on Privacy 196/2003 [29].

The total number of records collected amounts to 1523, all belonging to type-2 diabetes patients (updated to 10/21/2019). The physiological parameters taken into account are as follows: temperature, blood sugar, systolic blood pressure, oxygen saturation and heart rate.

Below a table listing dataset characteristics is reported in Table 2.

Note that, to ensure that data were ready for their correct processing, sometimes it has been needed to cope with the absence of heart rate measurement within some of the diabetes medical records. However, during the data collection phase it was possible to attribute this lack mainly due to the alternative acquisition of electrocardiographic reports by the medical staff; for this reason, when the latter were regular and the clinicians confirmed a perfectly normal cardiovascular system for the subjects, a heart rate value belonging to the range of normal values, between 53 and 100 bpm, was randomly assigned considering the uniform distribution of our training data that were in a normal range. This dataset has been properly preprocessed and optimized, for performing the training phase of the three classification algorithms. Specifically, to allow the classifier to distinguish between diabetic and healthy patients, it was necessary to add artificially generated healthy patients' data to the dataset. Data

of healthy patients have been produced automatically by simply ensuring that the values generated are inside of intervals delineated by specific thresholds provided by domain experts.

**Table 2.** Dataset characteristics.

Disease	Type-2 Diabetes
#patients	1523
#male	744
#female	779
up to 30 y/o	4
between 30 and 40 y/o	28
between 40 and 50 y/o	86
between 50 and 60 y/o	283
between 60 and 70 y/o	637
between 70 and 80 y/o	382
between 80 and 90 y/o	103
#patients without heart rate value	188

Using this approach, with the help of Matlab scripts, new synthetic data representative of the new class attribute value ‘Healthy’ were generated. The new tuples are obtained by ensuring that the new randomly generated values are within a specific range and follow a well-defined distribution (the value of the standard deviation of each feature is a priori defined in the script). Since the new data are randomly generated, the classification models may be too biased towards the new “artificial” class. For this reason, the classifiers would have a very low accuracy. To avoid this kind of problem the number of new tuples of the “healthy” class is slightly less than the number of tuples of the other class. (1337 healthy versus 1523 diabetics).

The fuzzy sets of input/output variables have been defined with the help of the physicians for the correct labelling of the ranges relating to the parameters; the fuzzification of the inputs has been achieved by using triangular and trapezoidal membership functions defined in accordance with threshold values provided by medical researchers and clinicians.

### 2.1.2. MLlib model selection

MLlib’s tooling is used to optimize hyperparameters in algorithms and pipelines. MLlib supports model selection using data to find the best model or parameters for a given task. The tools require an Estimator, the algorithm or pipeline to tune, a set of ParamMaps, a “parameter grid” to search over, and an Evaluator, a metric to measure the performance of the model. At a high level, these tools split the input data into separate training and test dataset, they iterate through the set of ParamMaps for each pair, fit the Estimator using those parameters and get the fitted model. Finally, they evaluate the model’s performance using the Evaluator and select the model produced by the best-performing set of parameters. First, we create a single (training, test) dataset pair, split the dataset into two parts using a train ratio parameter and implement the classifier. According to the specific features of each of the classifiers, careful tuning of their parameters is carried out and the goodness of the achieved model is evaluated. Each of the algorithms will have specific parameters to set based on its characteristics. Then, the CrossValidator tool is exploited to split the dataset into a set of 10 folds, which are used as separate

training and test datasets (10 equal sized partitions of data at 10 instances of learning, using 9 of them for training and 1 for testing). To evaluate a particular ParamMap, CrossValidator computes the average evaluation metric for the models produced by fitting the Estimator on the 10 different (training, test) dataset pairs. To help construct the parameter grid, we use the ParamGridBuilder utility. After identifying the best ParamMap, CrossValidator finally re-fits the Estimator using the best ParamMap and the entire dataset. Figure 1 indicates the code section that implements the training and the accuracy test phases for the Random Forest classifier. The same operations are performed also for the other classification algorithms listed above.

```
# First algorithm ..... RandomForestClassifier
# Classifier declaration (RandomForestClassifier)
rfc = RandomForestClassifier()

# Linking of labelIndexer, vecAssembler and RFCmodel in a pipeline
pipelineRF=Pipeline(stages=[labelIndexer,vecAssembler,rfc])

# Execution of pipeline stages and model training
modelRF=pipelineRF.fit(trainingData)

#Testing of created model on test set
predictionsRF=modelRF.transform(testData)

#Evaluation of accuracy of RFCmodel with MulticlassClassificationEvaluator
evaluator=MulticlassClassificationEvaluator(labelCol="label"
                                           ,predictionCol="prediction",metricName="accuracy")
accuracyRF=evaluator.evaluate(predictionsRF)
print("Model Accuracy: ",accuracyRF)

#Creation of a ParamGrid and an Evaluator for CrossValidation
paramGridRF=ParamGridBuilder().addGrid(rfc.maxBins,[10,100,1000]).build()
cvEvaluatorRF=MulticlassClassificationEvaluator(metricName="accuracy")
#Execution of CrossValidation
cvRF=CrossValidator(estimator=pipelineRF,estimatorParamMaps=paramGridRF
                   ,evaluator=cvEvaluatorRF,numFolds=10)
cvModelRF=cvRF.fit(testData)
cvPredictionsRF=cvModelRF.transform(testData)
```

**Figure 1.** Random Forest Classifier training and test.

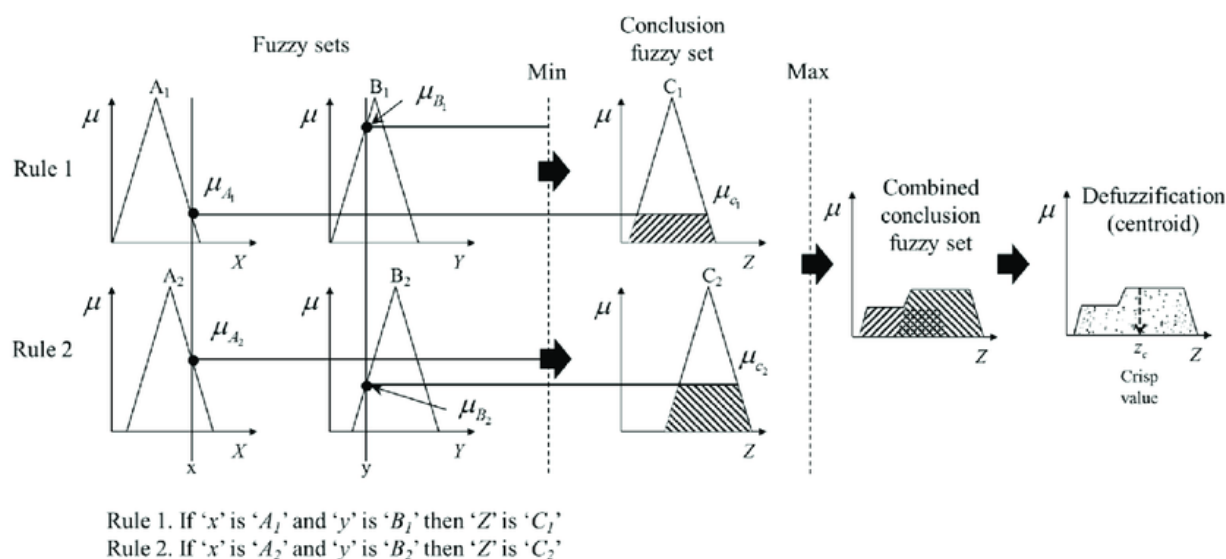
### 2.1.3. Voting technique

Each of the three classifiers can discriminate a specific pathology with a higher level of accuracy than the remaining two classes. So, once the three classifiers get the predictions in output, these are used by a voter to identify the majority class to be attributed to the tuple. The voter uses an ensemble technique based on majority policy to achieve better performances. This technique requires that, once the predictions from the four classifiers are obtained, they are used by a voter to identify the majority class to be attributed to the tuple. Indeed, voting makes a prediction corresponding to that which receives more than half of the votes, assigning to each tuple the value expected by at least three of the classifiers. If classifiers have different values, the voter chooses the prediction provided by the Random Forest Classifier because it is usually characterized by greater precision [30].

## 2.2. FIS implementation

The goal of the FIS is to predict the different grades of severity related to current clinical picture of the patients and, hence, synthesize them on a coloured graph for the ease of representation. The design of the FIS for the monitoring of the health status of diabetic patients can be summarized into 3 main stages: fuzzification, inference and defuzzification. All the designed clinical linguistic variables, membership functions and rules have been included into a Mamdani FIS, according to [31–34]. Mamdani linguistic model is built on Fuzzy IF-THEN rules where both the preceding and consequent sentences contain linguistic variables and, therefore, it is an intuitive model often used in CDSS thanks to its ability to implement human knowledge and human experience in the system, as done in this work where the knowledge-base is determined from the expertise of medical doctors and patients' information [35].

Following, the Mamdani framework and the basic knowledge implemented into the system are described with reference to a multi-inputs and single-output decision model, as shown in Figure 2.



**Figure 2.** Mamdani FIS process.

To perform inference, the first step is to “evaluate the antecedent”, which involves fuzzifying the inputs and applying any necessary fuzzy operators to each rule. Given the information input  $u = \{u_1, \dots, u_n\}$ , the strength level or firing level  $\alpha_i$  of the rule  $R_i$  is calculated in terms of the degrees of membership  $\mu_{A_{ij}}$ . If the antecedent clauses (the if part) are related by AND then:

$$[\alpha_i = \min(\mu_{A_{i,1}}(u_1), \dots, \mu_{A_{i,n}}(u_n))] \quad (1)$$

Else if the antecedent clauses are related by OR then:

$$[\alpha_i = \max(\mu_{A_{i,1}}(u_1), \dots, \mu_{A_{i,n}}(u_n))] \quad (2)$$

Indeed, the strength level is then used to shape the output fuzzy set that represents the consequent part of the rule [36]. The second step is the so-called “implication,” or, in other words, applying the



result of the antecedent to the consequent. The operator of implication for the rule  $R_i$  is defined as the shaping of the “consequent” (the output fuzzy set), based on the “antecedent”. The input of the implication process is a single number given by the “antecedent”, and the output is a fuzzy set:

$$[\mu_{B_i}(y) = \min(\alpha_i(u), \mu_{B_i}(y))] \quad (3)$$

where  $y$  is the variable that represents the support value of output the membership function  $\mu_{B_i}(\cdot)$ .

Now, to unify the outputs of all the rules, we need to aggregate the corresponding output fuzzy set into one single composite set. The inputs of the aggregation process are represented by the clipped fuzzy sets obtained by the implication process. The aggregation method exploited in our application is the  $\max(\cdot)$  one.

Finally, the defuzzification process has been performed starting from the output fuzzy set resulting from the aggregation process. The operations of defuzzification are computed as the centre of gravity (COG) of the strength levels:

$$[COG(y) = \frac{\sum_{i=1}^m y \mu_{B_i}(y)}{\sum_{i=1}^m \mu_{B_i}(y)}] \quad (4)$$

### 2.2.1. Identification of the fuzzy sets

Before their design, there is the need of a preliminary phase for the correct definition of the ranges into which the inputs variables values must be divided and the choice of the fuzzy sets to be used. In our work, this phase involved the intervention of the clinical experts of the Anti-Diabetes Centre (CAD) for the correct labelling of sets. It is necessary to specify that the therapeutic diagnostic path of a diabetic patient involves the participation of various professionals, from the general practitioner to the dietician; in our case, a diabetologist (8 years of experience), a general practitioner (3 years of experience) and a nurse (10 years of experience) of the Anti-Diabetes Centre help us in this preliminary phase. Each fuzzy set has been hence identified by a level and a score indicating the degree of physiological parameter impairment that the set describes. The ranges of each input and their corresponding fuzzy set are recorded in the following Table 3.

### 2.2.2. Fuzzification of inputs and output

Fuzzification follows the preliminary design phase and aims characterizing the inputs and determining the degree to which each of them belongs to a particular fuzzy set through membership functions definition. Input fuzzification has been here achieved by using trapezoidal membership functions in accordance with threshold values provided by clinician knowledge and medical standards (see the Membership Functions in Figure 3).

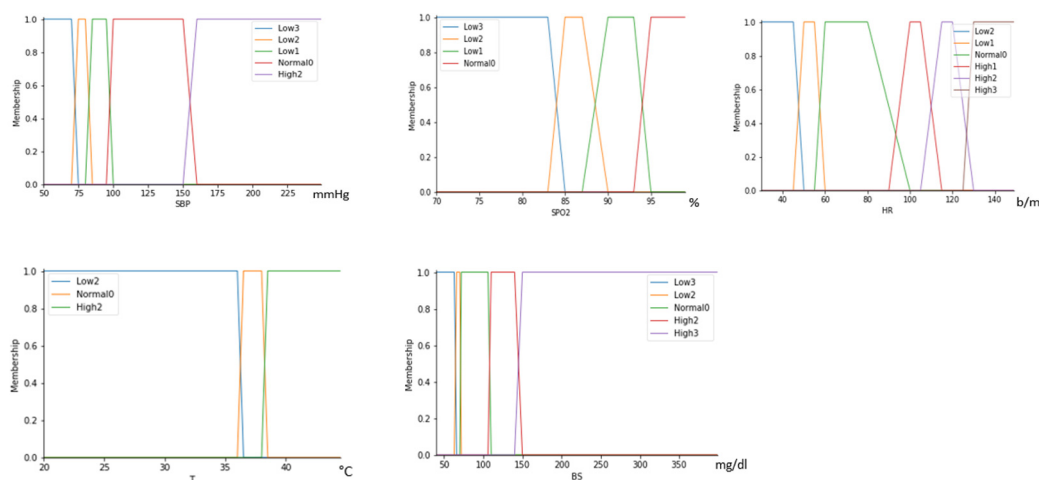
With respect to the output variable Risk Group (RG), which refers to the degree of illness of patients, 15 fuzzy sets have been defined whose membership functions are selected as triangular (see Figure 4). The built fuzzy sets are reported in Table 4.

Herein, the Normal fuzzy set (NRM) refers to a normal health status, the Low Risk Group fuzzy sets (LRG1-LRG4) indicate an intermediate alarm level and the High Risk Group fuzzy sets (HRG5-HRG14) are associated with a dangerous situation. The Risk Groups are inspired by a scoring scale validated and used in hospitals, the Early Warning Score (EWS) [37]. EWS scale requires that clinical

instability of the patient is calculated from the sum of the scores attributed to each parameter according to the value they assume. Thus, considering the scores defined by the input fuzzy sets, the output can take all values between 0 and 14.

**Table 3.** Inputs fuzzy sets.

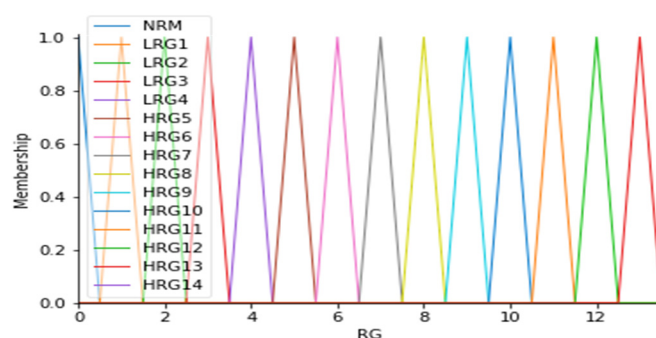
Input	Ranges	Fuzzy Sets
Systolic Blood Pressure (SBP)	< 75	Low3
	70–85	Low2
	80–100	Low1
	95–160	Normal0
	> 150	High2
Oxygen Saturation (SPO <sub>2</sub> )	< 85	Low3
	83–90	Low2
	87–95	Low1
	> 93	Normal0
Heart Rate (HR)	< 50	Low2
	45–60	Low1
	53–100	Normal0
	95–115	High1
	105–130	High2
Temperature (T)	< 36.5	Low2
	36–38.5	Normal0
	> 38	High2
Blood Sugar (BS)	< 66	Low3
	63–72	Low2
	60–110	Normal0
	106–150	High2
	> 140	High3



**Figure 3.** Inputs membership functions.

**Table 4.** Risk group fuzzy sets.

Output	Range	Fuzzy Sets
Risk Group	$0 < RG < 0.5$	NRM
	$0.5 < RG < 1.5$	LRG1
	$1.5 < RG < 2.5$	LRG2
	$2.5 < RG < 3.5$	LRG3
	$3.5 < RG < 4.5$	LRG4
	$4.5 < RG < 5.5$	HRG5
	$5.5 < RG < 6.5$	HRG6
	$6.5 < RG < 7.5$	HRG7
	$7.5 < RG < 8.5$	HRG8
	$8.5 < RG < 9.5$	HRG9
	$9.5 < RG < 10.5$	HRG10
	$10.5 < RG < 11.5$	HRG11
	$11.5 < RG < 12.5$	HRG12
	$12.5 < RG < 13.5$	HRG13
$13.5 < RG < 14.5$	HRG14	

**Figure 4.** Output membership function.

### 2.2.3. Design of Fuzzy rules and defuzzification

Starting from the membership functions, 1800 rules have been derived to cover all possible inputs combinations. Note that, the number of rules can be obtained from the following formula [27]:

$$[N = p_1 \times p_2 \times \dots \times p_n] \quad (5)$$

where N is the total number of possible rules, n is the number of linguistic variables and  $p_n$  is the number of linguistic terms for each linguistic variable.

A sample of the rules is shown in Figure 5.

Through the final defuzzification process, the combined fuzzy set from aggregation process will output a single scalar quantity (i.e., the diabetic patient's Risk Group). Depending on the numerical value assumed by the system output, an alert message displays the necessity of cares or physical examinations according to the clinical priority of the monitored diabetic patient depending from his/her Risk Group.

```

rule1 = ctrl.Rule(Input1['Low3'] & Input2['Low2'] & Input3['Low3'] & Input4['Low2'] & Input5['Low3'], Output1['HRG13'])
rule2 = ctrl.Rule(Input1['Low3'] & Input2['Low2'] & Input3['Low3'] & Input4['Low2'] & Input5['Low2'], Output1['HRG12'])
rule3 = ctrl.Rule(Input1['Low3'] & Input2['Low2'] & Input3['Low3'] & Input4['Low2'] & Input5['Normal0'], Output1['HRG10'])
rule4 = ctrl.Rule(Input1['Low3'] & Input2['Low2'] & Input3['Low3'] & Input4['Low2'] & Input5['High2'], Output1['HRG12'])
rule5 = ctrl.Rule(Input1['Low3'] & Input2['Low2'] & Input3['Low3'] & Input4['Low2'] & Input5['High3'], Output1['HRG13'])
rule6 = ctrl.Rule(Input1['Low3'] & Input2['Low2'] & Input3['Low3'] & Input4['Normal0'] & Input5['Low3'], Output1['HRG11'])
rule7 = ctrl.Rule(Input1['Low3'] & Input2['Low2'] & Input3['Low3'] & Input4['Normal0'] & Input5['Low2'], Output1['HRG10'])
rule8 = ctrl.Rule(Input1['Low3'] & Input2['Low2'] & Input3['Low3'] & Input4['Normal0'] & Input5['Normal0'], Output1['HRG8'])
rule9 = ctrl.Rule(Input1['Low3'] & Input2['Low2'] & Input3['Low3'] & Input4['Normal0'] & Input5['High2'], Output1['HRG10'])
rule10 = ctrl.Rule(Input1['Low3'] & Input2['Low2'] & Input3['Low3'] & Input4['Normal0'] & Input5['High3'], Output1['HRG11'])
rule11 = ctrl.Rule(Input1['Low3'] & Input2['Low2'] & Input3['Low3'] & Input4['High2'] & Input5['Low3'], Output1['HRG13'])
rule12 = ctrl.Rule(Input1['Low3'] & Input2['Low2'] & Input3['Low3'] & Input4['High2'] & Input5['Low2'], Output1['HRG12'])
rule13 = ctrl.Rule(Input1['Low3'] & Input2['Low2'] & Input3['Low3'] & Input4['High2'] & Input5['Normal0'], Output1['HRG10'])
rule14 = ctrl.Rule(Input1['Low3'] & Input2['Low2'] & Input3['Low3'] & Input4['High2'] & Input5['High2'], Output1['HRG12'])
rule15 = ctrl.Rule(Input1['Low3'] & Input2['Low2'] & Input3['Low3'] & Input4['High2'] & Input5['High3'], Output1['HRG13'])
rule16 = ctrl.Rule(Input1['Low3'] & Input2['Low2'] & Input3['Low2'] & Input4['Low2'] & Input5['Low3'], Output1['HRG12'])
rule17 = ctrl.Rule(Input1['Low3'] & Input2['Low2'] & Input3['Low2'] & Input4['Low2'] & Input5['Low2'], Output1['HRG11'])
rule18 = ctrl.Rule(Input1['Low3'] & Input2['Low2'] & Input3['Low2'] & Input4['Low2'] & Input5['Normal0'], Output1['HRG9'])
rule19 = ctrl.Rule(Input1['Low3'] & Input2['Low2'] & Input3['Low2'] & Input4['Low2'] & Input5['High2'], Output1['HRG11'])
rule20 = ctrl.Rule(Input1['Low3'] & Input2['Low2'] & Input3['Low2'] & Input4['Low2'] & Input5['High3'], Output1['HRG12'])
rule21 = ctrl.Rule(Input1['Low3'] & Input2['Low2'] & Input3['Low2'] & Input4['Normal0'] & Input5['Low3'], Output1['HRG10'])
rule22 = ctrl.Rule(Input1['Low3'] & Input2['Low2'] & Input3['Low2'] & Input4['Normal0'] & Input5['Low2'], Output1['HRG9'])
rule23 = ctrl.Rule(Input1['Low3'] & Input2['Low2'] & Input3['Low2'] & Input4['Normal0'] & Input5['Normal0'], Output1['HRG7'])

```

**Figure 5.** A sample of implemented rules.

### 3. Results

#### 3.1. Performance of the type-2 diabetic patients' classifier

Following the processing implemented through the classification algorithms, accuracy precision, recall and F1 values are obtained for each of the classifiers. These values are shown below:

**Table 5.** Evaluation metrics of the classifiers.

Algorithm	Accuracy	Precision	Recall	F1
Random Forest	0,91364	0,92648	0,91364	0,91375
Multilayer Perceptron	0,85492	0,90042	0,87219	0,85171
Naïve Bayes	0,81865	0,82758	0,81865	0,81900
SVM	0,82728	0,82961	0,82728	0,82765

In this work, more attention is given to the accuracy value to compare performances pre- and post-cross-validation.

#### 1) Random forest:

The model has an accuracy of 91.36%. To verify that the obtained accuracy value is not due to randomness and therefore not representative of the model's level of goodness, an approach with ten-fold cross-validation is considered. The new architecture reaches an accuracy of 92.57%.

#### 2) Multilayer perceptron:

For this algorithm it was necessary to set the maximum number of iterations, the number and size of the intermediate and output layers to stack the input data in matrices to speed up the calculation. This model reports an accuracy of 85.49%. After cross-validation, it shows an accuracy of 88.43%.

#### 3) Naïve Bayes:

The accuracy of the model reaches 81.86%. Following the 10-fold cross-validation, on the other hand, it is lower and equal to 81.00%.

#### 4) SVM:

For this algorithm, the accuracy is 82.73%. The cross-validation allows to reach an accuracy of 86.7%. Table 6 summarizes for each of the classifiers the accuracy values achieved with both the random case and the ten-fold cross-validation.

**Table 6.** Summary table of classifiers accuracy.

Algorithm	Accuracy	Accuracy post cross-validation
Random Forest	0,91364	0,92573
Multilayer Perceptron	0,85492	0,88428
Naïve Bayes	0,81865	0,81002
SVM	0,82728	0,86701

Specifically results summarized disclose that the classification algorithm that obtains the highest precision value is the Random Forest. Indeed, this algorithm achieves an accuracy of about 93%, unlike the other classification models that achieve lower performance. To obtain better performance from the data analysis, the ensemble learning technique was finally applied. An illustrative example of the results obtained applying the voting procedure are shown in Figure 6.

label	predictionRF	predictionMLP	predictionNB	predictionSVC	voter
1.0	1.0	1.0	1.0	1.0	1.0
1.0	1.0	1.0	1.0	1.0	1.0
1.0	1.0	1.0	1.0	1.0	1.0
1.0	1.0	1.0	1.0	1.0	1.0
1.0	1.0	1.0	1.0	1.0	1.0
1.0	1.0	1.0	1.0	1.0	1.0
0.0	0.0	0.0	0.0	0.0	0.0
1.0	1.0	1.0	1.0	0.0	1.0

**Figure 6.** Example of the voting procedure.

To evaluate the performances of the voter, the total value of accuracy reached is calculated and compared with the other methods. The voter system achieves an accuracy value of 93.501%. The overall architecture allows to obtain a better predictive model with higher accuracy levels than those obtained using single classification algorithms. The voting technique, therefore, allows to preserve the ability to correctly classify specific pathologies observed for each of the four classifiers, improving the total accuracy of the architecture.

### 3.2. FIS validation on four case studies

Here some illustrative results on the automatic evaluation of the Risk Group are presented for 4 given patients, among the 1523 accesses detected at the Anti-Diabetes Centre, in order to provide a usage example on the field. Hence, considering real physiological parameters values obtained from

database of data collected at the CAD as inputs of the system, it has been possible to compare FIS results with the real diabetic patients' health status; indeed, system output is a representative level of patients' clinical picture, whose actual condition are documented by the CAD medical records. The considered parameters characterize the 4 exemplar patients, or case study, that, at the time of the follow-up visit at the CAD, have presented the following different conditions:

- 1) Normal clinical picture;
- 2) Slightly compromised clinical picture;
- 3) Hyperglycaemic crisis clinical picture;
- 4) Compromised clinical picture.

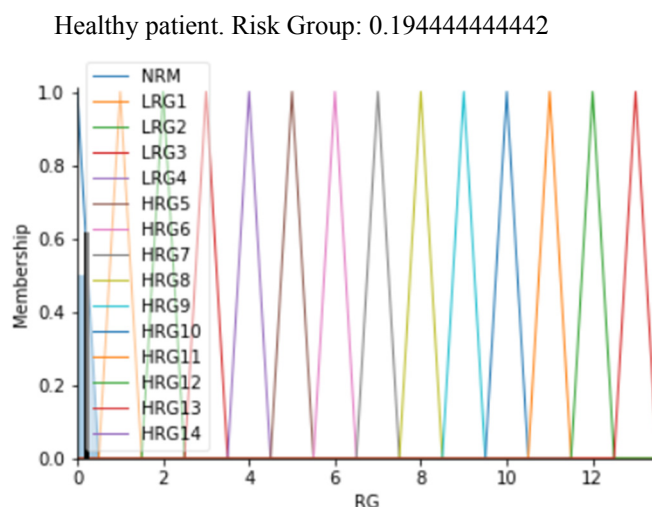
### 3.2.1. Case study 1

As the first case, it has been considered a diabetic patient who presents the following normal vital parameters at the follow-up visit:

- Systolic Blood Pressure (SBP) = 140 mmHg;
- Heart Rate (HR) = 90 b/m;
- Oxygen Saturation (SPO<sub>2</sub>) = 98%;
- Temperature (T) = 37°C;
- Blood sugar (BS) = 95 mg/dl.

```
RiskGroup_sim = control_sim1()
try:
    result = FuzzyInferenceSystem(RiskGroup_sim, 140, 90, 98, 37, 95)
    print(result)
    Output1.view(sim=RiskGroup_sim)
except:
    print("error")
```

**Figure 7.** Example of data entry via the Colab platform.



**Figure 8.** Results display for Case study 1.

The analysis ends with the display of a message that correctly predicts the severity of patient's health status and the risk level obtained from defuzzification. Indeed, results in Figure 8 disclose that there is no risk for the patient (in fact the message "Healthy patient" is displayed) and the identified Risk Group is 0.19. Note that the FIS allows to make the output of the system easier to be interpreted by both the patient and the clinician, since the output membership function graph is also displayed on the screen and the result of the defuzzification process is clearly highlighted on the display.

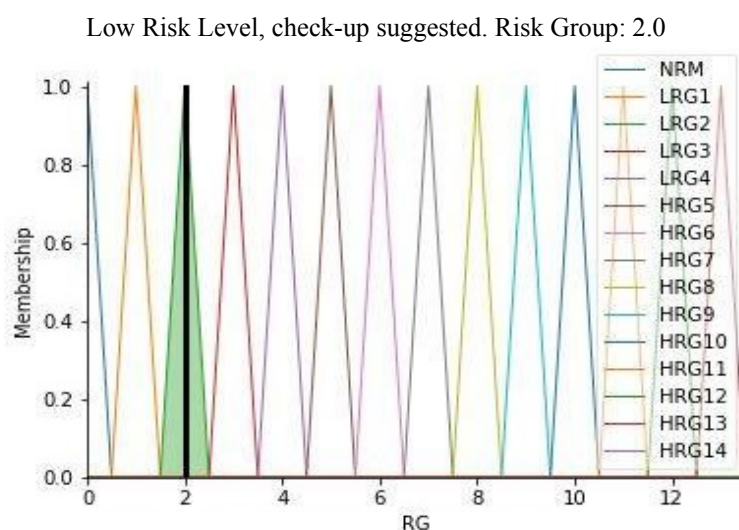
This result coincides with the diagnosis made by the clinician for the patient and documented in the diabetes medical record.

### 3.2.2. Case study 2

As a second case, a diabetic patient with normal blood sugar and the remaining parameters slightly outside the normal range is presented. Indeed, we have that the systolic blood pressure is slightly low and the heart rate is slightly above the norm:

- Systolic Blood Pressure (SBP) = 90 mmHg;
- Heart Rate (HR) = 100 b/m;
- Oxygen Saturation (SPO<sub>2</sub>) = 98%;
- Temperature (T) = 37 °C;
- Blood Sugar (BS) = 75 mg/dl.

Even then, results in Figure 9, correctly assess that the risk is very low (a simple check-up is suggested in the output FIS message) displaying a Risk Group 2, in accordance with the real condition of the patient.



**Figure 9.** Results display for Case study 2.

### 3.2.3. Case study 3

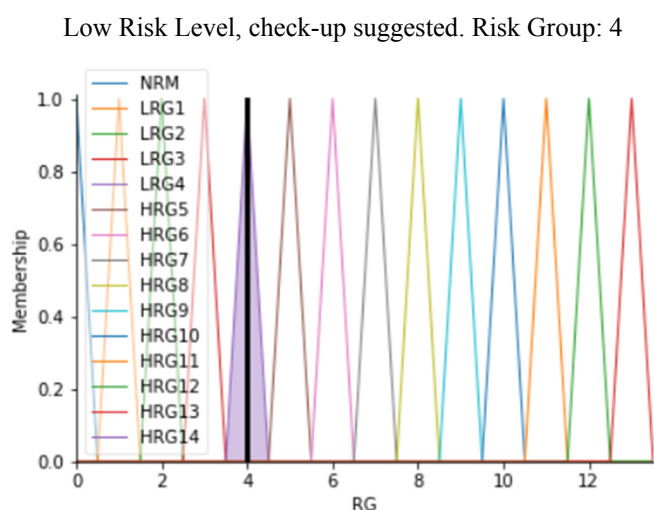
As a third exemplar case, we have a diabetic patient, who is at the check-up in a condition of hyperglycaemia. Chronic hyperglycaemia, which persists even under fasting conditions, is commonly caused by diabetes mellitus; in prediabetic states it can occur as intermittent hyperglycaemia. As



documented in the CAD medical record, the health status of the diabetic patient under examination is not worrying and the vital parameters are normal except for blood sugar, which instead reaches a peak of 300 mg/dl:

- Systolic Blood Pressure (SBP) = 130 mmHg;
- Heart Rate (HR) = 54 b/m;
- Oxygen Saturation (SPO<sub>2</sub>) = 98%;
- Temperature (T) = 37 °C;
- Blood Sugar (BS) = 300 mg/dl.

Being the analysis of a single episode, it is not possible to discriminate whether it is chronic or intermittent hyperglycaemia, but certainly, in the perspective of the periodic use of the FIS by a chronic patient, the identification of an adequate alarm state can be decisive for timely intervention. For this reason, it is interesting to understand the actual response of the system solicited by a clinical picture in which there is a single extremely high input. As shown in Figure 10, the level of risk identified is at the limit between low and high degree of alert; indeed, the identified Risk Group is 4.



**Figure 10.** Results display for Case study 3.

The answer of the FIS has been validated also for this case study, comparing it with the “true” answer provided by the clinicians. The discussion with the medical experts confirms that the output of our system is once again consistent with the patient's actual state of health, to which a check-up visit is in any case correctly suggested.

#### 3.2.4. Case study 4

As the last example, the case of a compromised clinical picture has been considered. Both the systolic blood pressure and blood sugar values are in fact altered:

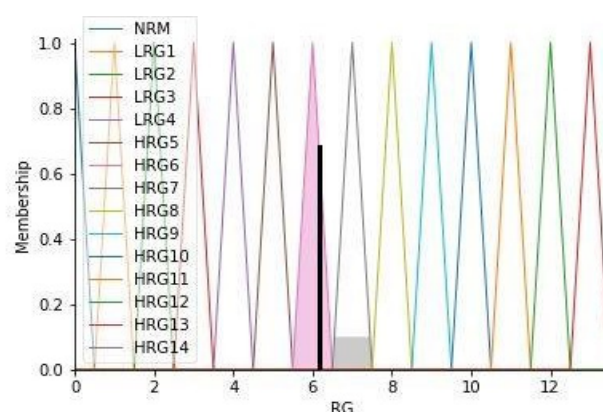
- Systolic Blood Pressure (SBP) = 200 mmHg;
- Heart Rate (HR) = 106 b/m;
- Oxygen Saturation (SPO<sub>2</sub>) = 98%;



- Temperature (T) = 37 °C;
- Blood Sugar (BS) = 181 mg/dl.

Therefore, according to the documented health status, the patient is exposed to a significant risk. As shown in Figure 11, the FIS returns a medium-high risk level; the identified Risk Group is 6.16 and a control visit is deemed necessary.

Medium-high, check-up required. Risk Group: 6.159663865546219



**Figure 11.** Results display for Case study 4.

Again, we have that the prediction made by the FIS is consistent with patient's condition, confirming the excellent functioning of the FIS. Indeed, the computed Risk Group is at an intermediate level of the risk level scale which includes 15 different groups and so the HRG6 consistently identifies an overall situation of intermediate severity.

#### 4. Discussion and conclusions

This paper proposes a CDSS for the identification of type-2 diabetic patients and for the determination of the risk level related to their health status, to early detect an alarm condition and prevent a critical situation. In particular, the use of classification techniques makes the system firstly able to recognize the type of user, distinguishing between pathological and non-pathological one. The choice of multiclass classifiers was dictated by the future possibility of expanding the sample of data with parameters from type 1 diabetes patients; this would allow us to instruct classifiers to distinguish between healthy, type 1 or type 2 diabetic users. After the recognition phase, the system will be able to determine the user's profile, in this paper characterized by the type-2 diabetes pathology, and to define his health state through a Fuzzy Inference System.

The mapping process of the inputs into the output is governed by appropriate inference rules: to include all the possibilities among the inputs, 1800 rules are developed, mathematically formulated to allow the conversion of the fuzzy system output into a single value, attributed to the health condition of the patients.

Four assessments shown in Results are generalizable and applicable to the entire CAD dataset; in this way medical and clinical conclusions can be easily presented for each patient. The FIS thus

obtained shows a marked ability to identify diabetic patients in critical conditions, consistently the information learned from CAD medical records; for these reasons it represents an efficient support system for clinical decisions, capable of strengthening staff skills in interpreting vital parameters of the patient. With the gradual development of health care systems exploiting Artificial Intelligence potential, the CDSSs should play a central role in reducing medical errors and in improving the quality of health care and the efficiency of the health care delivery system.

### Conflict of interest

All authors declare no conflicts of interest in this paper.

### Acknowledgment

This work has been carried out under TablHealth [CUP B69J17000660008] project, whose scientific director is Professor S. Santini.

### References

1. E. Bonora, S. Cataudella, G. Marchesini, R. Miccoli, O. Vaccaro, G. P. Fadini, Clinical burden of diabetes in Italy in 2018: a look at a systemic disease from the ARNO Diabetes Observatory, *BMJ Open Diabetes Res. Care*, **8** (2020), e001191.
2. T. Klingeberg, M. Schilling, Mobile wearable device for long term monitoring of vital signs, *Comput. Methods Prog. Biomed.*, **106** (2012), 89–96.
3. G. Improta, M. Triassi, G. Guizzi, L. C. Santillo, R. Revetria, A. Catania, et al., An innovative contribution to health technology assessment, in *Modern Advances in Intelligent Systems and Tools*, Springer, (2012), 127–131.
4. C. F. Lin, Mobile telemedicine: A survey study, *J. Med. Syst.*, **36** (2012), 511–520.
5. F. M. E. Uzoka, O. Obot, K. Barker, J. Osuji, An experimental comparison of fuzzy logic and analytic hierarchy process for medical decision support systems, *Comput. Methods Prog. Biomed.*, **103** (2011), 10–27.
6. I. Graham, P. L. Jones, *Expert Systems: Knowledge, Uncertainty, and Decision*, Chapman and Hall, 1998.
7. C. Habib, A. Makhoul, R. Darazi, R. Couturier, Multisensor data fusion and decision support in wireless body sensor networks., in *NOMS 2016-2016 IEEE/IFIP Network Operations and Management Symposium*, (2016), 708–712.
8. N. Barakat, A. P. Bradley, M. N. H. Barakat, Intelligible support vector machines for diagnosis of diabetes mellitus, *IEEE Trans. Inf. Technol. Biomed.*, **14** (2010), 1114–1120.
9. J. P. Kandhasamy, S. Balamurali, Performance analysis of classifier models to predict diabetes mellitus, *Procedia Comput. Sci.*, **47** (2015), 45–51.
10. A. Charleonnann, T. Fufaung, T. Niyomwong, W. Chokchueypattanakit, S. Suwannawach, N. Ninchawee, Predictive Analytics for Chronic Kidney Disease Using Machine Learning Techniques, in *2016 Management and Innovation Technology International Conference (MITicon)*, (2016).

11. S. Dwivedi, R. Borse, A. Yametkar, Lung cancer detection and classification by using machine learning & multinomial Bayesian, *IOSR J. Electron. Commun. Eng.*, **9** (2014), 69–75.
12. S. Santini; A. Pescapé; A. S. Valente; V. Abate; G. Improta, M. Triassi, et al., Using fuzzy logic for improving clinical daily-care of  $\beta$ -thalassemia patients, in *2017 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*, IEEE, (2017), 5090–6034.
13. G. Improta, V. Mazzella, D. Vecchione, S. Santini, M. Triassi, Fuzzy logic–based clinical decision support system for the evaluation of renal function in post-Transplant Patients, *J. Eval. Clin. Pract.*, **26** (2020), 1224–1234.
14. Novruz Allahverdi, S. Torun, I. Saritas, Design Of A Fuzzy Expert System For Determination Of Coronary Heart Disease Risk, in *International Conference on Computer Systems and Technologies*, (2007).
15. C. R. M. Leite, G. R. Sizilio, A. D. Neto, R. A. Valentim, A. M. Guerreiro, A fuzzy model for processing and monitoring vital signs in ICU patients, *Biomed. Eng. Online*, **10** (2011), 68.
16. G. Cappon, M. Vettoretti, F. Marturano, A. Facchinetti, G. Sparacino, A neural-network-based approach to personalize insulin bolus calculation using continuous glucose monitoring, *J. Diabetes Sci. Technol.*, **12** (2018), 265–272.
17. W. C. Hsu, K. H. K. Lau, R. Huang, S. Ghiloni, H. Le, S. Gilroy, et al., Utilization of a cloud-based diabetes management program for insulin initiation and titration enables collaborative decision making between healthcare providers and patients, *Diabetes Technol. Ther.*, **18** (2016), 59–67.
18. P. Chen, C. Pan, Diabetes classification model based on boosting algorithms, *BMC Bioinf.*, **19** (2018), 109.
19. B. J. Lee, J. Y. Kim, Identification of type 2 diabetes risk factors using phenotypes consisting of anthropometry and triglycerides based on machine learning, *IEEE J. Biomed. Health Inf.*, **20** (2016), 39–46.
20. P. Grant, A new approach to diabetic control: fuzzy logic and insulin pump technology, *Med. Eng. Phys.*, **29** (2007), 824–827.
21. B. Cosenza, Off-line control of the postprandial glycemia in type 1 diabetes patients by a fuzzy logic decision support, *Exp. Syst. Appl.*, **39** (2012), 10693–10699.
22. L. A. Zadeh, The role of fuzzy logic in modeling, identification and control, in *Fuzzy Sets, Fuzzy Logic, And Fuzzy Systems*, (1996), 783–795.
23. C. C. Lee, Fuzzy logic in control systems: fuzzy logic controller. I, *IEEE Trans. Syst. Man Cybern.*, **20** (1990), 404–418.
24. F. Baig, M. S. Khan, Y. Noor, M. Imran, Design model of fuzzy logic medical diagnosis control system, *Int. J. Comput. Sci. Eng.*, **3** (2011), 2093–2108.
25. D. Driankov, H. Hellendoorn, M. Reinfrank, *An Introduction to Fuzzy Control*, Springer Science & Business Media, 2013.
26. L. A. Zadeh, The concept of a linguistic variable and its application to approximate reasoning, *Inf. Sci.*, **8** (1975), 199–249.
27. E. H. Mamdani, Advances in the linguistic synthesis of fuzzy controllers, *Int. J. Man Mach. Stud.*, **8** (1976), 669–678.
28. M. K. Choudhury, N. Baruah, A fuzzy logic-based expert system for determination of health risk level of patient, *Int. J. Res. Eng. Technol.*, **4** (2015), 261–267.

29. M. A. Ameen, J. Liu, K. Kwak, Security and privacy issues in wireless sensor networks for healthcare applications, *J. Med. Syst.*, **36** (2012), 93–101.
30. L. I. Kuncheva, J. J. Rodríguez, A weighted voting framework for classifiers ensembles, *Knowl. Inf. Syst.*, **38** (2014), 259–275.
31. P. P. Wang, D. Ruan, E. E. Kerre, *Fuzzy logic: A spectrum of theoretical & practical issues*, Springer, 2007.
32. W. J. Wang, H. R. Lin, Fuzzy control design for the trajectory tracking on uncertain nonlinear systems, *IEEE Trans. Fuzzy Syst.*, **7** (1999), 53–62.
33. D. R. Keshwani, D. D. Jones, G. E. Meyer, R. M. Brand, Rule-based Mamdani-type fuzzy modeling of skin permeability, *Appl. Soft Comput.*, **8** (2008), 285–294.
34. S. Russel, P. Norvig, *Artificial Intelligence: A Modern Approach*, London: Pearson Education Limited, 2013.
35. D. H. Choi, B. S. Ahn, S. H. Kim, Multicriteria group decision making under incomplete preference judgments: using fuzzy logic with a linguistic quantifier, *Int. J. Intell. Syst.*, **22** (2007), 641–660.
36. E. H. Mamdani, S. Assilian, An experiment in linguistic synthesis with a fuzzy logic controller, *Int. J. Man Mach. Stud.*, **7** (1975), 1–13.
37. C. P. Subbe, M. Kruger, P. Rutheford, L. Gemmel, Validation of a modified Early Warning Score in medical admissions, *QJM*, **94** (2001), 521–526.

## Appendix

The essential definitions to describe the theoretical principles of a FIS are presented below. A linguistic variable is a variable whose values are words or sentences of a language, natural or artificial, exploited to ease a gradual transition between the two states of binary logic and to express in the most natural way the measurements' vagueness, which is not possible by using crisp variables. Hence, it holds:

**Definition 1.** *Linguistic variable.* A linguistic variable can be characterized by a quintuple (L, F(L), U, R, M) in which L is the name of the variable; F(L) is the term-set of L, that is the collection of its linguistic values; U is a universe of discourse; R is a syntactic rule that generates the terms in F(L); M is a semantic rule which associates to each linguistic value X its meaning, M(X), where M(X) denotes a fuzzy subset of U.

**Definition 2.** *Fuzzy variable.* A fuzzy variable is characterized by a triple (L, U, F(L; u)), in which L is the name of the variable; U is a universe of discourse (finite or infinite set); u is a generic name for the elements of U; and F(L; u) is a fuzzy subset of U which represents a fuzzy restriction on the values of u imposed by L. F(L; u) will be referred to as the restriction on u or the restriction imposed by L. The assignment equation for L has the form:

$$x = u : F(L)$$

and represents an assignment of a value u to x subject to the restriction F(L).

In the universe of discourse U, a fuzzy set F(L; u) is characterized by a *membership function* (MF)  $\mu_F$  that assigns a membership value to elements u, within a predefined range of U, as follows:  $F = \{(u, \mu_F) \mid u \in U \text{ and } \mu_F : U \rightarrow [0, 1]\}$ . Therefore, a membership function is a distribution that maps every single point of the input space (i.e. the universe of speech, which represents the set of the

linguistic variables) in a membership value between 0 and 1. The membership functions related to the various linguistic variables are composed in order to constitute a rule. Indeed, fuzzy sets are used to treat uncertainty and to represent knowledge through rules since the Fuzzy Logic allows the interpretation of data on the basis of predefined linguistic variables according to appropriate IF-THEN rules written as:

IF situation THEN conclusion

where the situation represents the antecedent or the premise consisting of fuzzy terms connected by fuzzy operators, while the output is called consequent or conclusion.



AIMS Press

©2021 the Author(s), licensee AIMS Press. This is an open access article distributed under the terms of the Creative Commons Attribution License (<http://creativecommons.org/licenses/by/4.0>)