Applied Soft Computing 13 (2013) 563–573

Contents lists available at SciVerse ScienceDirect

Applied Soft Computing

journal homepage: www.elsevier.com/locate/asoc

Neutrosophic classifier: An extension of fuzzy classifer

A.Q. Ansari [a], Ranjit Biswas [b], Swati Aggarwal [c,*]

[a] Department of Electrical Engineering, Jamia Millia Islamia, New Delhi 110025, India
[b] Department of Computer Science, Jamia Hamdard (Hamdard University), New Delhi 110062, India
[c] Department of Computer Science Engineering, ITM University, HUDA Sector 23-A, Gurgaon 122017, Haryana, India

ARTICLE INFO

Article history:
Received 23 March 2011
Received in revised form 17 May 2012
Accepted 4 August 2012
Available online 30 August 2012

Keywords:
Neutrosophic logic
Neutrosophic classifier

ABSTRACT

Fuzzy classification has become of great interest because of its ability to utilize simple linguistically interpretable rules and has overcome the limitations of symbolic or crisp rule based classifiers. This paper introduces an extension to fuzzy classifier: a neutrosophic classifier, which would utilize neutrosophic logic for its working. Neutrosophic logic is a generalized logic that is capable of effectively handling indeterminacy, stochasticity acquisition errors that fuzzy logic cannot handle. The proposed neutrosophic classifier employs neutrosophic logic for its working and is an extension of commonly used fuzzy classifier. It is compared with the commonly used fuzzy classifiers on the following parameters: nature of membership functions, number of rules and indeterminacy in the results generated. It is proved in the paper that extended fuzzy classifier: neutrosophic classifier; optimizes the said parameters in comparison to the fuzzy counterpart. Finally the paper is concluded with justifying that neutrosophic logic though in its nascent stage still holds the potential to be experimented for further exploration in different domains.

1. Introduction

Classification is the process of arranging data into homogeneous classes on the basis of the common features present in the data [1].

Various machine learning based techniques are used for input data classifications that provide a rational answer for all possible inputs [2]. Fuzzy matching of input and subsequent fuzzy processing is an active research area that has been successfully applied to varied domains from control theory to artificial intelligence [3,4].

This paper is written with the aim of focusing on the classification performed on the data which is uncertain, imprecise, incomplete and ambiguous. In this paper authors propose a new classification technique based on neutrosophic logic which is an extension of fuzzy logic.

2. Present work

Fuzzy logic was given by Prof. L.A. Zadeh in his seminal paper during second half of last century [5]. Though with weak acceptance initially, slowly it has emerged as one of the important soft computing techniques to model uncertainty [6]. Real world information is full of uncertainties, gaps and inconsistent information. This uncertainty can be encountered in varied forms like uncertainty in outcome of tossing a coin; whether it will be a head or tail is an example of classical bivalence where uncertainty disappears on the completion of event. Or else accurate description of the statement "Rose is red"; has fuzzy uncertainty associated with it as it is difficult to define red color due to various possible shades of the same color applicable [7].

The main work of this paper is dedicated in providing extension to commonly used fuzzy classifier in the form of neutrosophic classifier. Fuzzy classifier uses fuzzy logic. So this section gives brief details of a fuzzy logic and fuzzy classifier in its general form.

Prof. L. Zadeh had revolutionized the field of logics by proposing a novel fuzzy logic in 1965 where each element in fuzzy set has a degree of membership [5].

Definition 1. Fuzzy sets and membership functions

If X is a collection of objects denoted generically by x, then a fuzzy set A in X is defined as a set of ordered pairs:

$$A = \{(x, \mu_A(x)) | x \in X\} \quad (1)$$

$\mu_A(x)$ is called the membership function of x in A. The membership function maps each element of X to a continuous membership value between 0 and 1.

It also has the provision of allowing linguistic variables whose truth values may vary between 0 and 1; in contrast to two values of classical logic [8].

Ever since the beginning of fuzzy set theory [5], classification domain has been an important theoretical and practical fuzzy application area [9]. Crisp classes represent an unrealistic

* Corresponding author. Tel.: +91 9717995716.
E-mail address: swati1178@gmail.com (S. Aggarwal).

1568-4946/$ – see front matter © 2012 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.asoc.2012.08.002

oversimplification of reality, which fuzzy approach seems to handle easily. Fuzzy classification applications assume C, a set of classes. The problem is then to determine for every object x under consideration, $x \in X$, the degree $\mu_c(x)$ to which object x belongs to class $c \in C$.

So a membership function $\mu_c(x)$: $X \rightarrow [0,1]$ has been defined for each class $c \in C$ [10].

Fuzzy classifier uses informal knowledge about problem domain for classification. For example: "If it is sunny then it will not rain". Fuzzy classification is driven by creating fuzzy category membership functions that convert objectively measurable parameters to category memberships which are then used for fuzzy classification [11]. Membership functions refer to overlapping ranges of feature values.

Definition 2. Fuzzy classifier

Let x be a vector in an n-dimensional real space R^n (the feature space) and let $C = \{c_1, c_2, \ldots, c_c\}$, be a set of class labels. Bezdek et al. [12] has defined crisp and fuzzy classifier.

A crisp classifier is mapping of the type:

$$O_c : R^n \rightarrow C \tag{2}$$

A fuzzy classifier is any classifier which uses fuzzy sets either during training or during its operation. It uses fuzzy if-then inference system which yields a class label for x [10].

$$O_F : R^n \rightarrow [0, 1]^c \tag{3}$$

So, instead of assigning a class label from C, O_F assigns to $x \in R^n$ a soft class label with degrees of membership in each class.

$$R^n \rightarrow \mu_C(x) \quad \forall x \in R^n \quad \text{and} \quad \sum_{i=1}^{c} \mu_i(x) = 1 \tag{4}$$

The result of fuzzy classification is represented by $O_F = \{(x, \mu_C(x)) | x \in R^n\}$.

Next section is dedicated to the understanding of neutrosophic logic, which is essential in defining the underlying principle for the working of proposed neutrosophic classifier.

3. Neutrosophic logic

Quite recently, neutrosophic logic was proposed by Florentine Smarandache which is based on the non-standard analysis that was given by Abraham Robinson in 1960s [13]. Neutrosophic logic was developed to represent mathematical model of uncertainty, vagueness, ambiguity, imprecision, incompleteness, inconsistency, redundancy and contradiction [14]. Neutrosophic logic is a logic in which each proposition is estimated to have the percentage of truth in a subset T, the percentage of indeterminacy in a subset I, and the percentage of falsity in a subset F, where T, I, F are standard or non-standard real subsets of $]^-0,1^+[$ [15]:

$$\text{with } \sup T = t\_\sup, \inf T = t\_\inf$$

$$\sup I = i\_\sup, \inf I = i\_\inf$$

$$\sup F = f\_\sup, \inf F = f\_\inf$$

and

$$n\_\sup = t\_\sup + i\_\sup + f\_\sup$$

$$n\_\inf = t\_\inf + i\_\inf + f\_\inf.$$

The sets T, I and F are not necessarily intervals, but may be any real sub-unitary subsets: discrete or continuous; single-element, finite, or (countably or uncountably) infinite; union or intersection of various subsets; etc. They may also overlap [16]. Statically T, I and F are subsets. We use a subset of truth (or indeterminacy, or falsity), instead of a number only, because in many cases we are not able to exactly determine the percentages of truth and of falsity but approximate them: for example a proposition is between 30 and 40% true and between 60 and 70% false, even worst: between 30 and 40% or 45 and 50% true (according to various analyzers), and 60% or between 66 and 80% false. Neutrosophic logic suggests that neutrosophic probability (using subsets; not numbers as components) should be used for better representation as it is a more natural and justified estimation [15].

All the factors stated by neutrosophic logic are very integral to human thinking, as it is very rare that we tend to conclude/judge in definite environments, imprecision of human systems could be due to the imperfection of knowledge that human receives (observation) from the external world [17]. For example: for a given proposition "Movie *ABC* would be hit", human brain certainly in this situation cannot generate precise answers in terms of yes or no, as indeterminacy is the sector of unawareness of a proposition's value, between truth and falsehood; undoubtedly neutrosophic components best fits in the modeling of simulation of human brain reasoning.

Definition 3. *Neutrosophic set* [15]: Let X be a space of points (objects), with a generic element in X denoted by x.

A neutrosophic set A in X is characterized by a truth-membership function T_A, a indeterminacy-membership function I_A and a falsity-membership function F_A. $T_A(x)$, $I_A(x)$ and $F_A(x)$ are real standard or non-standard subsets of $]^-0,1^+[$. That is

$$\begin{aligned} T_A &: X \rightarrow]^-0, 1^+[\\ I_A &: X \rightarrow]^-0, 1^+[\\ F_A &: X \rightarrow]^-0, 1^+[\end{aligned} \tag{6}$$

There is no restriction on the sum of $T_A(x)$, $I_A(x)$ and $F_A(x)$, so

$$^-0 = \sup T_A(x) + \sup I_A(x) + \sup F_A(x) = 3^+ \tag{7}$$

Also as neutrosophy allows the provision of reflecting the dynamics of things and ideas [16]; the proposition "Movie *ABC* would be hit" does not mean fixed value components structure; the truth value of the proposition may change from place to place. For example: proposition "Movie *ABC* would be hit" may yield neutrosophic components 0% true, 0% indeterminate and 100% false in north sector and may yield (1,0,0) in south sector.

Neutrosophy also allows change in values with respect to the observer [16]. For example: proposition "Movie *ABC* would be hit" may yield neutrosophic components ($t = 0.60$, $i = 0.30$, $f = 0.20$) if observed by any film critic then results would differ; like ($t = 0.30$, $i = 0.15$, $f = 0.80$) if analyzed by other critic.

4. How neutrosophic logic is different from fuzzy logic

Neutrosophic logic proposes that between an idea ⟨A⟩ and its opposite ⟨Anti-A⟩, there exists a gamut of continuous power spectrum of neutralities which can be represented by ⟨Neut-A⟩ [14].

If ⟨α⟩ be an attribute, for a proposition ⟨P⟩ and a referential system $\{R\}$, applying Neutrosophic logic yields $(T, I, F)]^-0,1^+[$ [3]. Then:

- ⟨P⟩ is T% ⟨α⟩, I% indeterminate or ⟨Neut-α⟩, and F% ⟨Anti-α⟩.
- It can be shown that ⟨α⟩ is at some degree ⟨Anti-α⟩, while ⟨Anti-α⟩ is at some degree ⟨α⟩.

This important concept of range of neutralities is missing in fuzzy logic and other allied logics, as fuzzy logic is concerned about membership and non membership of a particular element to a particular class; and does not deals with indeterminate nature of data acquired that could happen due to various reasons like incomplete

knowledge (ignorance of the totality, limited view on a system because of its complexity), stochasticity (the case of intrinsic imperfection where a typical and single value does not exist), or the acquisition errors (intrinsically imperfect observations, the quantitative errors in measures) [17].

The concept of fuzzy logic is constrained with the fact that non-membership value = 1 − membership value. In contrast to this the advantage of utilizing neutrosophic logic is that the values of its components T, I and F are not confined to the range of [0,1], and it well distinguishes between absolute true/false values from relative true/false values [16]. As neutrosophic logic has the provision of assigning >1 as well as <1 values to its neutrosophic components, (t,i,f), so whenever for any tautology $t/i/f > 1$, it would imply absolute true/indeterminate/false similarly whenever $t/i/f < 1$, it would imply conditional (relative) truth/indeterminacy/falsity. This mechanism of assigning over boiling values (>1) or under dried values (<0) helps in justifying dissimilarity between unconditionally true ($t > 1$, and $f < 0$ or $i < 0$) and conditionally true propositions ($t \leq 1$, and $f \leq 1$ or $i \leq 1$) [16].

When the sets are reduced to an element only, then

$$t\_\sup = t\_\inf = t,\ i\_\sup = i\_\inf = i,\ f\_\sup = f\_\inf = f$$

$$\text{and} \quad n\_\sup = n\_\inf = n = t + i + f$$

Hence, the neutrosophic logic generalizes the fuzzy logic (for $n = 1$ and $i = 0$, and $0 \leq t, i, f \leq 1$) [17].

5. Neutrosophic classifier: an extension of fuzzy classifier

A classifier is an algorithm that predicts the class label on the basis of the object descriptor. Commonly used classifier in the soft computing domain is fuzzy classifier. Fuzzy classifier uses fuzzy sets or fuzzy logic in the course of its training or operation. This paper proposes extension of fuzzy classifier that is neutrosophic classifier that will use neutrosophic logic which is a superset of fuzzy logic.

Definition 4. Neutrosophic classifier: a classifier that would use neutrosophic logic principles and neutrosophic sets for the classification. Neutrosophic classifier incorporates a simple, neutrosophic rule based approach like: IF X and Y THEN Z, for solving problem rather than attempting to model a system mathematically similar to fuzzy classifier.

Let x be a vector in an n-dimensional real space R^n (the feature space) and let $C = \{c_1, c_2, \ldots, c_c\}$, be a set of class labels. A neutrosophic classifier is mapping of the type:

$$O_N : R^n \rightarrow \{T_C(x), I_C(x), F_C(x) | x \in R^n\} \tag{8}$$

If the result of neutrosophic classification is represented by O_N then

$$O_N = \{(x, [T_C(x), I_C(x), F_C(x)]) | x \in R^n\}$$

where

$$[T_C(x)] = \begin{bmatrix} t_{c_1}(x) \\ t_{c_2}(x) \\ . \\ . \\ t_{c_c}(x) \end{bmatrix}, \quad [I_C(x)] = \begin{bmatrix} i_{c_1}(x) \\ i_{c_2}(x) \\ . \\ . \\ i_{c_c}(x) \end{bmatrix}$$

$$\text{and} \quad [F_C(x)] = \begin{bmatrix} f_{c_1}(x) \\ f_{c_2}(x) \\ . \\ . \\ f_{c_c}(x) \end{bmatrix} \tag{9}$$

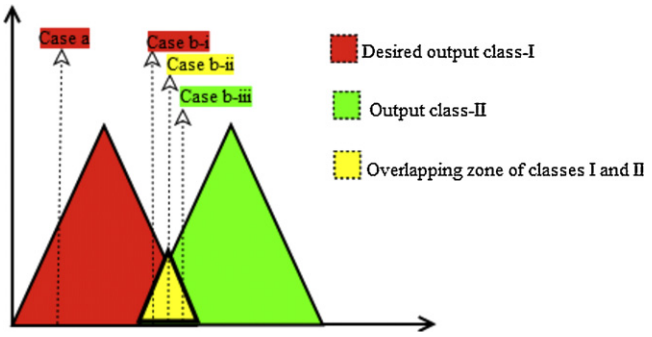

Fig. 1. Diagrammatic representation of different types of output generated by Mamdani type fuzzy classifier.

T, I and F component values are independent of each other and there is no restriction on the sum of $T_C(x)$, $I_C(x)$ and $F_C(x)$, so

$$^{-}0 \leq T_C(x) + I_C(x) + F_C(x) = 3^{+} \tag{10}$$

The non-standard unit interval $]^{-}0, 1^{+}[$ is merely used for philosophical applications, especially when distinction is required between absolute and relative truth/falsehood/indeterminacy. But for technical applications of neutrosophic logic and set, the domain of definition and range of the T, I and F can be restrained to the normal standard real unit interval [0,1], which is easier to use.

Sections 6 and 7 discuss implementation of fuzzy classifier and neutrosophic classifier, respectively. For simulations iris dataset (http://archive.ics.uci.edu/ml/datasets/Iris) is used. All experiments have been carried out on MATLAB 7.0 [18]. Iris dataset consists of 4 attributes; sepal length, sepal width, petal length and petal width and has 150 instances which are categorized into three classes: iris-setosa, iris-versicolor and iris-virginica. Thirty instances from each class have been used for training (for making rule set) and 20 from each class have been used as test case.

6. MATLAB implementation of fuzzy classifier

6.1. Fuzzy classifier—Matlab implementation of FIS-iris classification

Simple Mamdani type fuzzy classifier is designed using MATLAB for iris data set.

As overlapping is inherent of fuzzy logic so appropriate overlapping membership functions have been designed for all the Iris dataset attributes and output classes. Figs. 4a, 5a, 6a, 7a and 8a gives the membership function designed for Iris sepal length, sepal width, petal length, petal width and Iris output classes designed for Mamdani type fuzzy classifier.

It can be generalized that the outputs generated after defuzzification by FIS can be of two types:

Case a. When the output clearly lies in one of the output class.
Case b. When the defuzzified value belongs to the overlapping range, this indicates certain degree of indeterminacy associated for the values spanned by overlapping membership functions. In this case there are following three possibilities:

i. Higher membership value to correct class
ii. Equal membership value to two adjacent classes
iii. Higher membership value to wrong class

Cases a and b have been diagrammatically represented by Fig. 1.

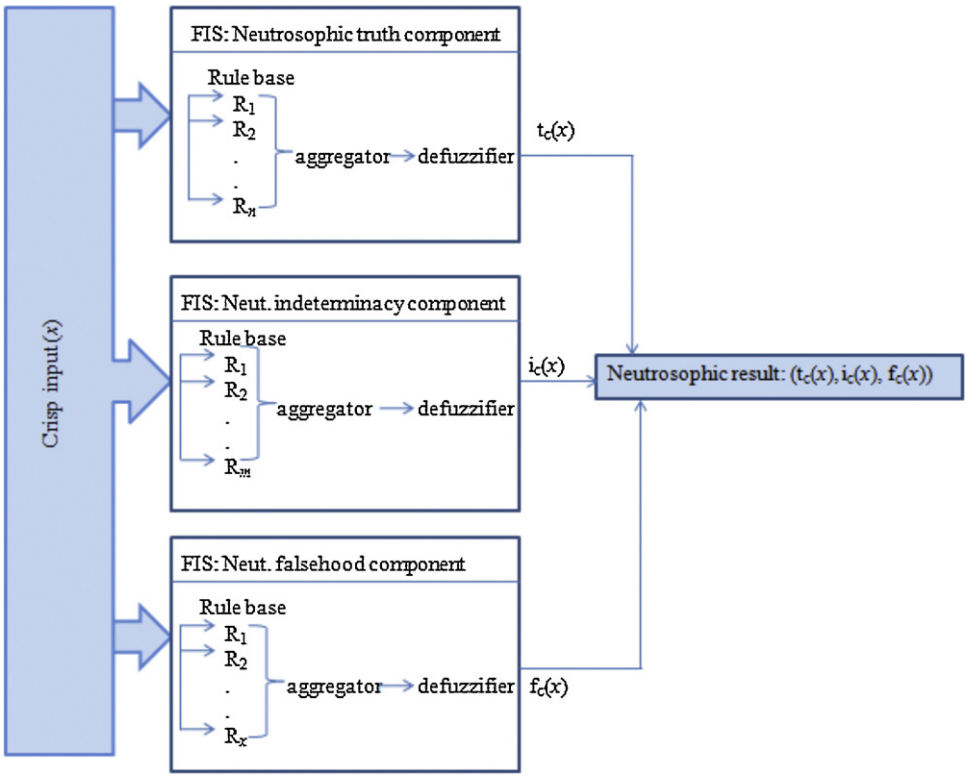

Fig. 2. Block diagram for a neutrosophic classification system using Fuzzy logic toolbox of Matlab.

When the output belongs to *case a*, then it is 100% sure that it belongs to a specific class, as for example 31st instance of iris dataset generates de-fuzzified value of 0.13, that indicates its 100% association with iris-setosa. But when output belongs to *case b*, then it lies in the indeterminacy range where the output membership value belongs to multiple classes with varying degree of membership.

7. Proposing neutrosophic classifier on the lines of fuzzy classifier

Neutrosophic systems similar to their fuzzy counterparts would be capable of utilizing knowledge obtained from human operators. In majority of the real world classifiers it is difficult to devise a precise mathematical model that would simulate system behavior; also it is unlikely that the data acquired by the system would be 100% complete and determinate [11]. Incompleteness and indeterminacy in the data can arise from inherent nonlinearity, time-varying nature of the process to be controlled, large unpredictable environmental disturbances, degrading sensors or other difficulties in obtaining precise and reliable measurements. Humans can take intelligent decisions in such situations. Though this knowledge is also difficult to express in precise terms, an imprecise linguistic description of the manner of control can usually be articulated by the operator with relative ease.

Neutrosophic classifier using neutrosophic logic is designed using MATLAB. It has been suggested on the lines of fuzzy logic but instead of giving one defuzzified value, output value in neutrosophic classifier takes the neutrosophic format of the type: output (truthness, indeterminacy, falsity) as represented by Eq. (9). Rest of the paper is organized in understanding of the concept that for applications where proportion of truthness, falsity and indeterminacy exists in the result generated, then it is essential to code using neutrosophic logic.

Designing of neutrosophic classification inference system using fuzzy methodology is based on the principles of Mamdani fuzzy inference method [19]. Currently there are no softwares available that supports neutrosophic logic, so the proposed work has been implemented on Fuzzy logic toolbox of Matlab 2007.

Fig. 2 gives the block diagram representation of a neutrosophic classification system using fuzzy logic toolbox of Matlab. As represented by Eq. (10), values of *T*, *I* and *F* neutrosophic components are independent of each other. So using fuzzy logic toolbox of Matlab, three FIS have been designed: one for neutrosophic truth component, second for neutrosophic indeterminacy component and third for neutrosophic falsity component. Though the working of these components are independent of each other but a correlation is drawn between membership functions of neutrosophic *T*, *I* and *F* components so as to capture the truthness, indeterminacy and falsity of the input as well as the output.

Pseudo code followed for implementation of neutrosophic classification inference system using fuzzy toolbox of Matlab is given below:

1. For the given input dataset, make the training and testing sets for each given class. Here first 30 instances from each of the Iris class are used as training sets and last 20 from each of the Iris class are used as testing cases.
2. Using FIS editor develop the following three inference systems which are independent of each other:
   a. Neutrosophic truth component
   b. Neutrosophic indeterminacy component
   c. Neutrosophic falsity component
3. Using the training set available, designing of inference system for truth component is done as follows:
   a. Membership functions for all the input and output variables are designed in such a way that there is no overlapping between any two membership functions using membership function editor.

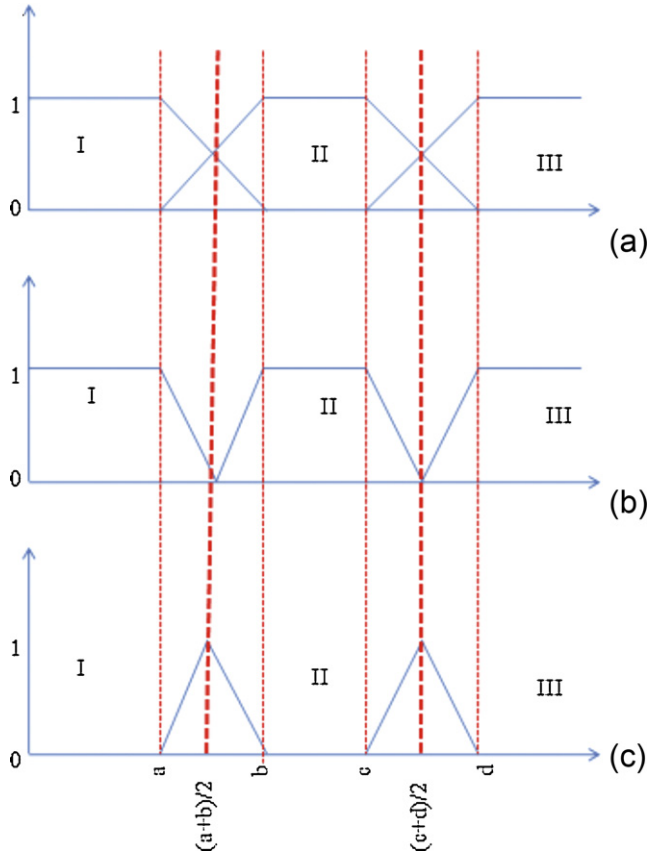

Fig. 3. Correlation and criteria for designing MFs for (a) fuzzy classifier, (b) neutrosophic truth component and (c) neutrosophic indeterminacy component.

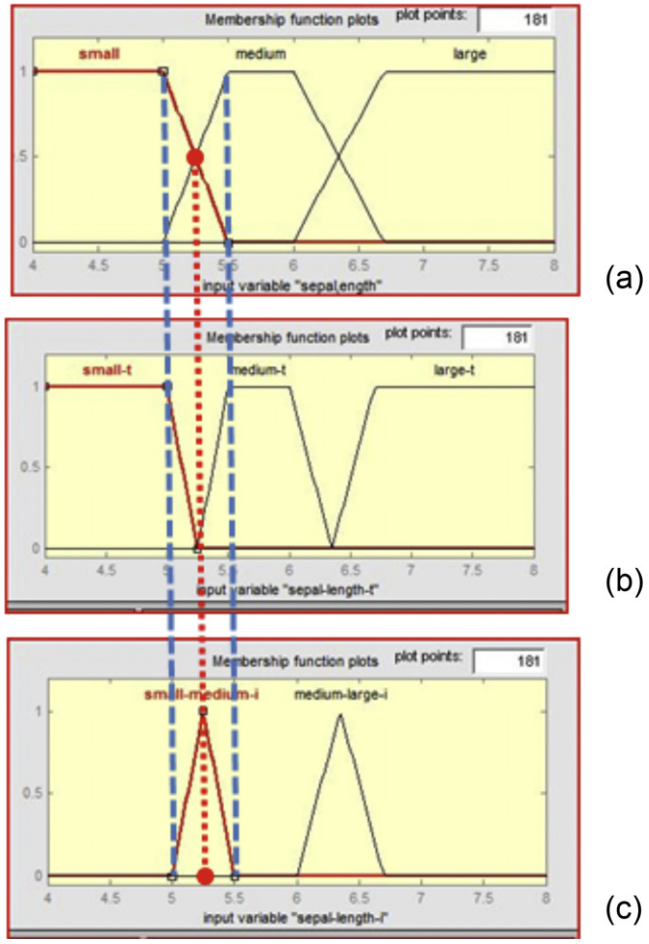

Fig. 4. Membership functions for attribute sepal length designed in (a) FIS, (b) neutrosophic truth component and (c) neutrosophic indeterminate component.

Fig. 3a and b gives the correlation and criteria for designing membership functions for FIS and neutrosophic truth component.

For range [0 – a]: fuzzy classifier shows 100% belongingness (Fig. 3a), same is retained in neutrosophic truth component for class I (Fig. 3b).

For range [a – (a + b)/2]: fuzzy classifier shows overlapping between class I, II and decrease in membership value. Till point (a + b)/2; class I has higher membership value as compared to class II (Fig. 3a), so decrease in truth MF for class I is shown for the range [a – (a + b)/2] (Fig. 3b).

For range [(a + b)/2 – b]: this is the overlapping zone represented by fuzzy classifier (Fig. 3a) in which MF value (class II) > MF value (class I). So increase in the neutrosophic truth MF for class II is shown for the range [(a + b)/2 – b] (Fig. 3b).

For ranges [b–c], [c – (c + d)/2] and [(c + d)/2 – d] same truth membership function designing criteria is followed as by [0 – a], [a – (a + b)/2] and [(a + b)/2 – b], respectively.

Figs. 4b, 5b, 6b, 7b and 8b shows truth membership functions for attribute sepal length, sepal width, petal length, petal width and 3 iris classes. Truth membership functions have been designed in such a way that there is zero overlapping, for the ranges where overlapping was designed using FIS. Overlapping regions that were recorded in the conventional FIS, have been captured by neutrosophic indeterminacy and falsity components. Neutrosophic truth component, here defined by Iris-*t* shows zero overlapping with truth value steadily decreasing for overlapping ranges contrary to what was designed for conventional FIS.

b. Appropriate rules are developed using rule editor.

Rule base for neutrosophic truth component for Iris dataset is shown in Fig. 9.

4. Using the training set available, designing of inference system for indeterminacy component is done as follows:

a. Membership functions for all the input and output variables are designed using membership function editor in such a way that there is no overlapping between any two membership functions and indeterminacy and falsity membership functions exist only for the ranges which would been spanned by two adjacent membership functions if were designed for fuzzy logic, as that common area has indeterminacy and falsity associated with it.

For range [0 – a]: fuzzy classifier shows 100% belongingness to class I (Fig. 3a) so indeterminacy for this range is 0 (Fig. 3c).

For range [a – (a + b)/2]: as truth value for class I is steadily decreasing here (Fig. 3a and b), so corresponding increase in indeterminacy is shown (Fig. 3c), with (a + b)/2 point representing highest indeterminacy value (because at point (a + b)/2 – both classes I and II give equal membership value in Mamdani fuzzy classifier (Fig. 3a)).

For range [(a + b)/2 – b]: as truth value for class II is steadily increasing (Fig. 3a and b) so corresponding decrease in indeterminacy value is shown (Fig. 3c).

As range [a – b], is the overlapping zone for class I and II in fuzzy classifier, so this range is represented as class I–class II–*i*; for indeterminacy component of neutrosophic logic.

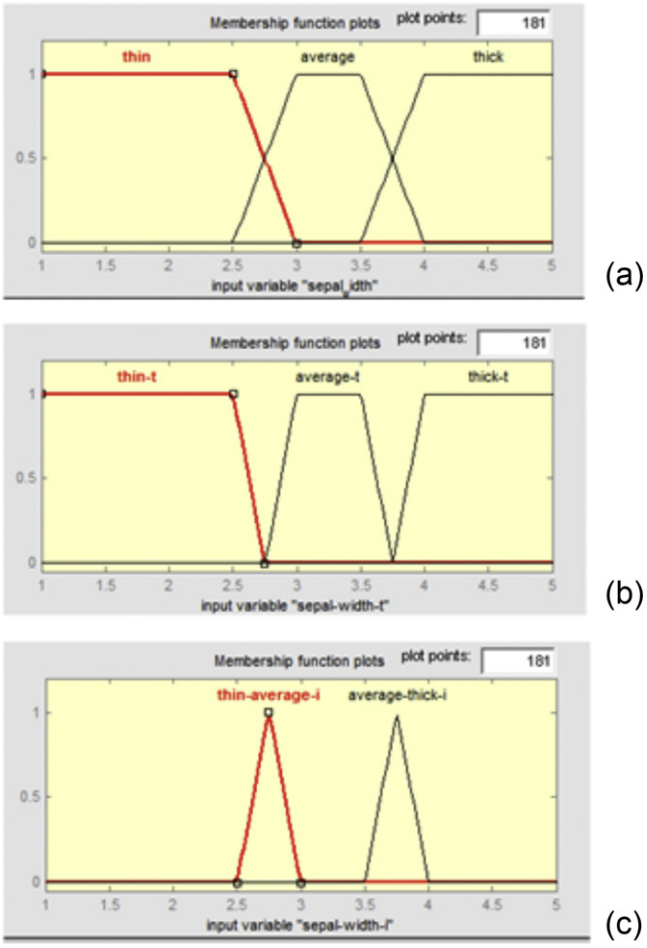

Fig. 5. Membership functions for attribute sepal width designed in (a) FIS, (b) neutrosophic truth component and (c) neutrosophic indeterminate component.

For ranges [b – c], [c – (c + d)/2] and [(c + d)/2 – d] same indeterminacy MF designing criteria is followed as above by [0 – a], [a – (a + b)/2] and [(a + b)/2 – b], respectively.

Indeterminacy and falsity neutrosophic components have been designed for Iris dataset for the ranges that were shown overlapping in FIS. Figs. 4c, 5c, 6c, 7c and 8c show membership functions of indeterminacy component Iris-*i* for attribute sepal length, sepal width, petal length, petal width and 3 iris classes.

b. Appropriate rules are developed using rule editor.

Rule base designed for neutrosophic indeterminacy component for Iris dataset is shown in Fig. 10.

5. Using the training set available, designing of inference system for neutrosophic falsity component is done in the same way as for indeterminacy component discussed in step 4; except that here for this classification example, height of all the membership functions is 0.5.
6. After training, the three components are tested independently using the testing data.
7. For each testing instance, final result is generated by consolidating results from truth, indeterminacy and falsity component in the triplet format of (*T*,*I*,*F*).
8. If a particular testing instance generates (*x*,*y*,*z*), it is interpreted as *x* grade of membership of instance to truth set, *y* and *z*; grade of indeterminacy and falsity membership to the respective sets.

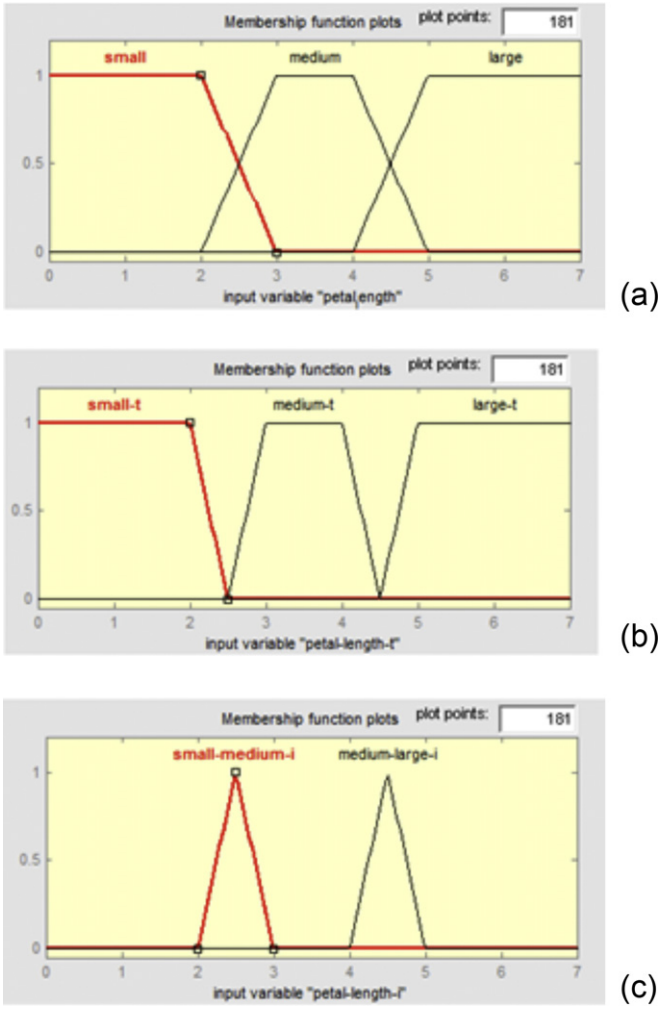

Fig. 6. Membership functions for attribute petal length designed in (a) FIS, (b) neutrosophic truth component and (c) neutrosophic indeterminate component.

Fig. 11 gives the number of training and testing data used for the implementation of fuzzy and neutrosophic classifier. As this work is dedicated to extend fuzzy classifier to neutrosophic classifier and discussing merits of neutrosophic classifier over conventional fuzzy classifier so same dataset and equal numbers of training and testing data cases are used for both.

8. Experimental results

Table 1 shows the details of training and testing sample using FIS. 30 instances from each class have been used for training (for making rule set) and 20 from each class have been used for testing.

Table 2 discusses the results of testing done using FIS. When FIS is used for classification, two overlapping zones are recorded for output classes (Fig. 8a).

Overlapping zone 1 Iris setosa and versicolor (no FIS result was recorded in this overlapping zone)
Overlapping zone 2 Iris versicolor and virginica

case i FIS output < 0.65, indicates higher membership with versicolor
case ii FIS output = 0.65, indicates equal membership with versicolor and virginica
case iii FIS output > 0.65, indicates higher membership with virginica

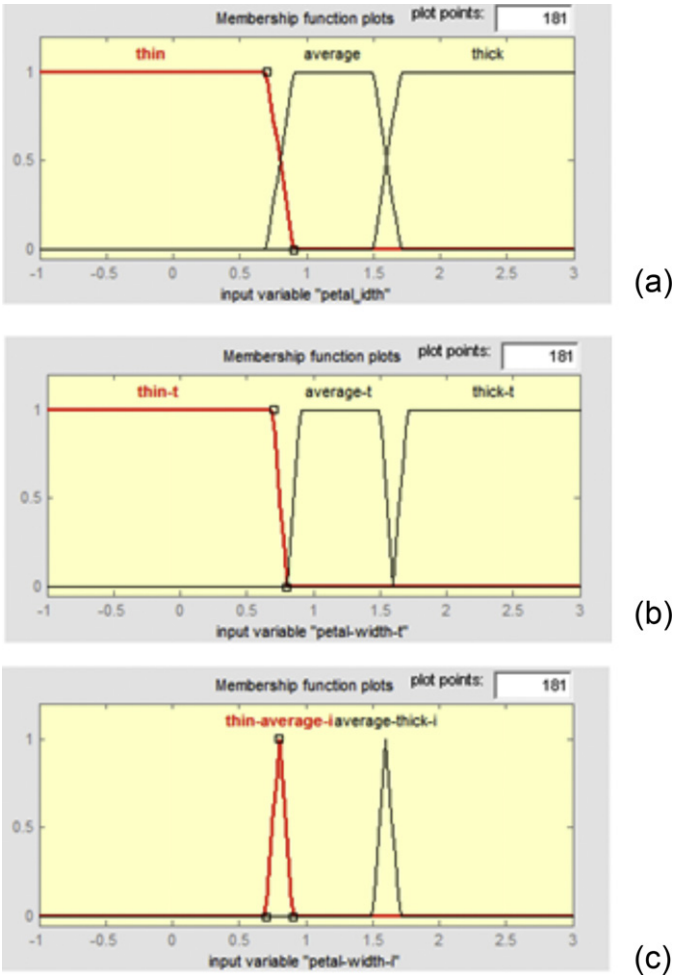

Fig. 7. Membership functions for attribute petal width designed in (a) FIS, (b) neutrosophic truth component and (c) neutrosophic indeterminate component.

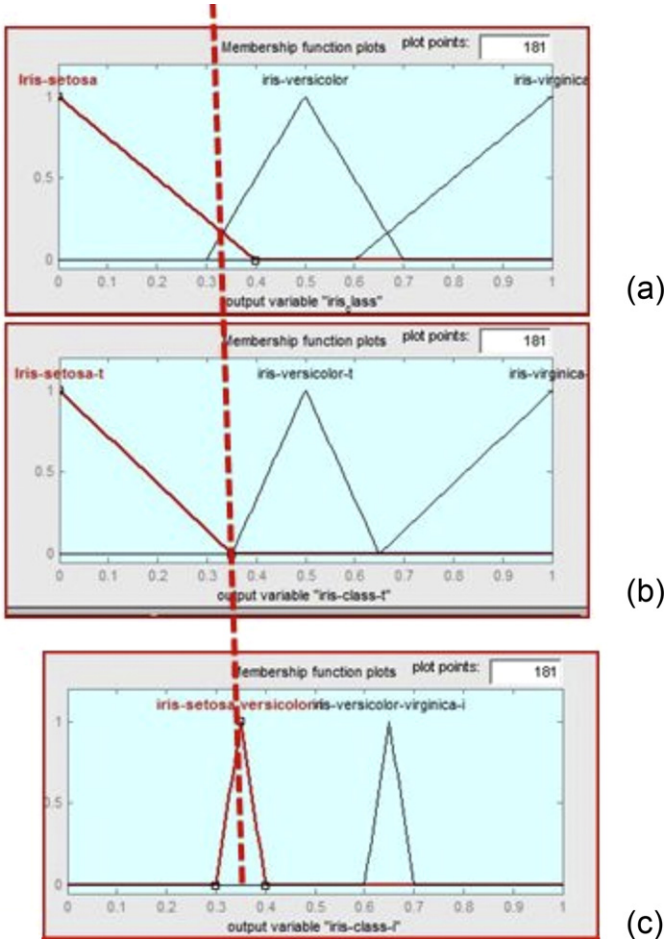

Fig. 8. Membership functions designed for output classes designed in (a) FIS, (b) neutrosophic truth component and (c) neutrosophic indeterminate component.

Here as the authors are concerned about dealing with the test cases whose result matched with the specifications of case b (Section 6), so for the 60 testing instances (20 from each of the three Iris classes); following Table 3 gives an overview of the results recorded in the overlapping zones for the three classes which account for the indeterminacy associated, when fuzzy classifier is employed.

Table 4 shows the details of training and testing sample using Neutrosophic truth component. Thirty instances from each class have been used for training (for making rule set) and 20 from each

Table 1
Details of training and testing samples using FIS.

Iris classes	Number of training samples used (serial number in the dataset)	Numbers of rules formed	Number of testing samples used (serial number in the dataset)
Iris-setosa	30 (1–30)		20 (31–50)
Iris-versicolor	30 (51–80)	28	20 (81–100)
Iris-virginica	30 (101–130)		20 (131–150)

Table 2
Details and interpretation of testing results lying in overlapping zones using FIS.

Iris classes	Analysis of the outputs (using test cases)	Details of outputs lying in overlapping zones
Iris-setosa	Results indicate clear belongingness to class setosa.	No result is recorded in the overlapping zone of setosa and versicolor
Iris-versicolor	16 cases indicate clear belongingness to class versicolor. 4 cases generated results lying in overlapping zone of versicolor and virginica.	*case (i)*: Output of instance 91 is 0.648 that is correct belongingness to the desired class *case (ii)*: Output of instance 88 is 0.65 that is ambiguous belongingness to two adjacent classes *case (iii)*: Output of instances 84 and 86 is 0.676 that is more belongingness to wrong class
Iris-virginica	2 cases indicate clear belongingness to class virginica 18 cases generated results lying in overlapping zone of versicolor and virginica.	*case (ii)*: Output of instance 134 is 0.65 that is ambiguous belongingness to two adjacent classes *case (iii)*: Output of 17 instances 131, 133, 135–146, 148–150 is >0.65 that is correct belongingness to the desired class

1. If (sepal-length-t is small-t) and (sepal-width-t is average-t) and (petal-length-t is small-t) and (petal-width-t is thin-t) then (iris-class-t is Iris-setosa-t) (1)
2. If (sepal-length-t is medium-t) and (sepal-width-t is thick-t) and (petal-length-t is small-t) and (petal-width-t is thin-t) then (iris-class-t is Iris-setosa-t) (1)
3. If (sepal-length-t is medium-t) and (sepal-width-t is average-t) and (petal-length-t is small-t) and (petal-width-t is thin-t) then (iris-class-t is Iris-setosa-t) (1)
4. If (sepal-length-t is small-t) and (sepal-width-t is thick-t) and (petal-length-t is small-t) and (petal-width-t is thin-t) then (iris-class-t is Iris-setosa-t) (1)
5. If (sepal-length-t is large-t) and (sepal-width-t is average-t) and (petal-length-t is large-t) and (petal-width-t is average-t) then (iris-class-t is iris-versicolor-t) (1)
6. If (sepal-length-t is medium-t) and (sepal-width-t is thin-t) and (petal-length-t is medium-t) and (petal-width-t is average-t) then (iris-class-t is iris-versicolor-t) (1)
7. If (sepal-length-t is medium-t) and (sepal-width-t is average-t) and (petal-length-t is large-t) and (petal-width-t is average-t) then (iris-class-t is iris-versicolor-t) (1)
8. If (sepal-length-t is medium-t) and (sepal-width-t is average-t) and (petal-length-t is large-t) and (petal-width-t is thick-t) then (iris-class-t is iris-versicolor-t) (1)
9. If (sepal-length-t is small-t) and (sepal-width-t is thin-t) and (petal-length-t is medium-t) and (petal-width-t is average-t) then (iris-class-t is iris-versicolor-t) (1)
10. If (sepal-length-t is medium-t) and (sepal-width-t is average-t) and (petal-length-t is medium-t) and (petal-width-t is average-t) then (iris-class-t is iris-versicolor-t) (1)
11. If (sepal-length-t is large-t) and (sepal-width-t is average-t) and (petal-length-t is medium-t) and (petal-width-t is average-t) then (iris-class-t is iris-versicolor-t) (1)
12. If (sepal-length-t is medium-t) and (sepal-width-t is thin-t) and (petal-length-t is large-t) and (petal-width-t is average-t) then (iris-class-t is iris-versicolor-t) (1)
13. If (sepal-length-t is large-t) and (sepal-width-t is average-t) and (petal-length-t is large-t) and (petal-width-t is thick-t) then (iris-class-t is iris-versicolor-t) (1)
14. If (sepal-length-t is medium-t) and (sepal-width-t is average-t) and (petal-length-t is large-t) and (petal-width-t is thick-t) then (iris-class-t is iris-virginica-t) (1)
15. If (sepal-length-t is medium-t) and (sepal-width-t is thin-t) and (petal-length-t is large-t) and (petal-width-t is thick-t) then (iris-class-t is iris-virginica-t) (1)
16. If (sepal-length-t is large-t) and (sepal-width-t is average-t) and (petal-length-t is large-t) and (petal-width-t is thick-t) then (iris-class-t is iris-virginica-t) (1)
17. If (sepal-length-t is small-t) and (sepal-width-t is thin-t) and (petal-length-t is large-t) and (petal-width-t is thick-t) then (iris-class-t is iris-virginica-t) (1)
18. If (sepal-length-t is large-t) and (sepal-width-t is thin-t) and (petal-length-t is large-t) and (petal-width-t is thick-t) then (iris-class-t is iris-virginica-t) (1)
19. If (sepal-length-t is large-t) and (sepal-width-t is thick-t) and (petal-length-t is large-t) and (petal-width-t is thick-t) then (iris-class-t is iris-virginica-t) (1)
20. If (sepal-length-t is medium-t) and (sepal-width-t is thin-t) and (petal-length-t is large-t) and (petal-width-t is average-t) then (iris-class-t is iris-virginica-t) (1)
21. If (sepal-length-t is small-t) and (sepal-width-t is thin-t) and (petal-length-t is small-t) and (petal-width-t is thin-t) then (iris-class-t is Iris-setosa-t) (1)

Fig. 9. Rule base for neutrosophic truth component.

1. If (sepal-length-i is small-medium-i) then (iris-class-i is iris-setosa-versicolor-i) (1)
2. If (sepal-width-i is average-thick-i) then (iris-class-i is iris-setosa-versicolor-i) (1)
3. If (sepal-length-i is small-medium-i) and (sepal-width-i is average-thick-i) then (iris-class-i is iris-setosa-versicolor-i) (1)
4. If (sepal-width-i is thin-average-i) then (iris-class-i is iris-setosa-versicolor-i) (1)
5. If (sepal-length-i is small-medium-i) and (sepal-width-i is thin-average-i) then (iris-class-i is iris-setosa-versicolor-i) (1)
6. If (sepal-length-i is medium-large-i) then (iris-class-i is iris-versicolor-virginica-i) (1)
7. If (sepal-width-i is thin-average-i) then (iris-class-i is iris-versicolor-virginica-i) (1)
8. If (sepal-length-i is medium-large-i) and (sepal-width-i is thin-average-i) then (iris-class-i is iris-versicolor-virginica-i) (1)
9. If (petal-length-i is medium-large-i) then (iris-class-i is iris-versicolor-virginica-i) (1)
10. If (sepal-width-i is average-thick-i) then (iris-class-i is iris-versicolor-virginica-i) (1)
11. If (sepal-width-i is thin-average-i) and (petal-length-i is medium-large-i) then (iris-class-i is iris-versicolor-virginica-i) (1)
12. If (sepal-length-i is medium-large-i) and (sepal-width-i is thin-average-i) and (petal-length-i is medium-large-i) then (iris-class-i is iris-versicolor-virginica-i) (1)
13. If (sepal-length-i is medium-large-i) and (petal-length-i is medium-large-i) then (iris-class-i is iris-versicolor-virginica-i) (1)
14. If (sepal-length-i is medium-large-i) and (petal-length-i is medium-large-i) and (petal-width-i is average-thick-i) then (iris-class-i is iris-versicolor-virginica-i) (1)
15. If (petal-length-i is medium-large-i) and (petal-width-i is average-thick-i) then (iris-class-i is iris-versicolor-virginica-i) (1)

Fig. 10. Rule base for neutrosophic indeterminacy component.

class have been used for testing. Also it lists the results of testing done using Neutrosophic truth component Iris-*t*. All testing samples generate results that indicate that samples clearly belong to desired class-*t*.

Table 5 shows the details of training and testing samples using Neutrosophic indeterminacy component and Neutrosophic falsity component. The results obtained are same for indeterminacy and falsity component, so same table (5) is used to gives details for both. 30 instances from each class have been used for training (for making rule set) and 20 from each class have been used for testing. Whenever result obtained is zero for falsity and indeterminacy components; this indicates that their generated truth component indicates correct belongingness to either of the classes: Iris-setosa/versicolor/virginica class.

Once the testing has been performed using all the three NIS-*t*, NIS-*i*, NIS-*f*, the results can be analyzed like if for example iris-setosa-$t = 0.127$, this quantifies the grade of membership of element 0.127 to neutrosophic set: iris-setosa-*t*; finally the results can be consolidated instance by instance in the triplet format of (*t*,*i*,*f*).

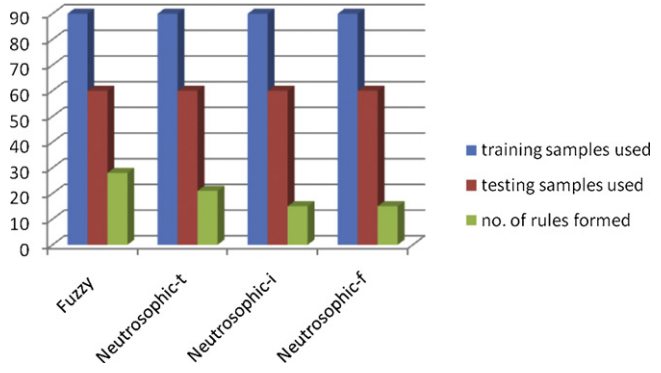

Fig. 11. Comparison of training and testing samples used for fuzzy, neutrosophic truth component, neutrosophic indeterminate component and neutrosophic falsity component, respectively.

Table 3
Summary of FIS results lying in overlapping range.

	Higher membership value to correct class	Equal membership value to two adjacent class	Higher membership value to wrong class
FIS-Iris-setosa	0	0	0
FIS-Iris-versicolor	1	1	2
FIS-Iris-virginica	17	1	0

Table 4
Details of training and testing samples using neutrosophic truth component Iris-*t*.

Iris classes	Training samples used (serial number in the dataset)	Numbers of rules formed	Testing samples used (serial number in the dataset)	Details and interpretation of incorrect results
Iris-setosa-*t*	30 (1–30)		20(31–50)	None
Iris-versicolor-*t*	30 (51–80)	21	20(81–100)	None
Iris-virginica-*t*	30 (101–130)		20(131–150)	None

Table 5
Details of training and testing samples using neutrosophic indeterminacy component Iris-*i*.

Iris classes Indeterminacy component/Falsity component	Training samples used (serial number in the dataset)	Numbers of rules formed	Testing samples used (serial number in the dataset)	Details of instances with non zero indeterminacy and falsity
Iris-setosa-versicolor-*i*/Iris-setosa-versicolor-*f* Iris-versicolor-virginica-*i*/Iris-versicolor-virginica-*f*	90 (1–30) (51–80) (101–130)	15	60 (31–50) (81–100) (131–150)	*Instances:* 32nd, 33rd and 40th 86th, 89th and 92nd, 96th, 99th 134th, 137–139th and 147–149th

Table 6
Analysis of NIS-*i*, NIS-*f* against values given by FIS.

Instance number	Values recorded using FIS	Correct/desired class	Values recorded using NIS (*t*,*i*,*f*)		
			Neutrosophic truth component (*t*)	Neutrosophic indeterminacy component (*i*)	Neutrosophic falsity component (*f*)
32	0.135	Setosa	Iris-setosa-*t* = 0.127	Iris-setosa-versicolor-*i* = 0.35	Iris-setosa-versicolor-*f* = 0.35
33	0.156		Iris-setosa-*t* = 0.156		
40	0.127		Iris-setosa-*t* = 0.127		
Interpretation: FIS results indicate correct membership to setosa, neutrosophic result is (*t*, *i*, *f*) = (>0.5, 1,0.5); indicating that truth values recorded are greater than 0.5 for all the three instances, with indeterminacy recorded is as high as 1 and falsity 0.5 (refer Fig. 8).					
86	0.676	Versicolor	Iris-versicolor-*t* = 0.5	Iris-versicolor-virginica-*i* = 0.65	Iris-versicolor-virginica-*f* = 0.65
87	0.5				
88	0.65				
89	0.5				
92	0.5				
96	0.5				
99	0.5		Iris-versicolor-*t* = 0.5	Iris-versicolor-virginica-*i* = 0.35	Iris-versicolor-virginica-*f* = 0.35
Interpretation: All FIS results indicate correct belongingness to versicolor, except 88th instance which reflects equal degree of membership to versicolor and virginica (refer Table 2); and 86th instance also which lies in the overlapping region of versicolor and virginica, shows higher membership value to virginica (refer Table 2). NIS result is (1,1,0.5); indicating that truth values recorded are 1 for versicolor, indeterminacy recorded is as high as 1 and falsity is 0.5 (refer Fig. 8).					
134	0.65	Virginica	Iris-versicolor-*t* = 0.5	0.5	0.5
Interpretation: Fuzzy results indicate ambiguity that is equal membership to versicolor and virginica, but neutrosophic results is (1,0,0); indicating clear membership to versicolor, also indeterminacy and falsity recorded for this instance is nil; which is not in accordance with the desired result (refer Fig. 8).					
137	0.678	Virginica	Iris-virginica-*t* = 0.683		
138	0.678		Iris-virginica-*t* = 0.684		
139	0.685		Iris-virginica-*t* = 0.703	Iris-versicolor-virginica-*i* = 0.65	Iris-versicolor-virginica-*f* = 0.65
147	0.842		Iris-virginica-*t* = 0.84		
148	0.683		Iris-virginica-*t* = 0.697		
149	0.683		Iris-virginica-*t* = 0.697		
Interpretation: All FIS results indicate correct belongingness to virginica. Neutrosophic result is (<0.5, 1,0.5); that reflects that the instances lie in the zone of high indeterminacy, result indicate that various truth values recorded are less than 0.5 for versicolor, with indeterminacy recorded is as high as 1 and falsity 0.5 (refer Fig. 8).					

Table 7
Analysis of wrong fuzzy results with neutrosophic results.

Instance number	Fuzzy results	Interpretation of fuzzy results	Neutrosophic truth component (*t*)	Neutrosophic indeterminacy component (*i*)	Neutrosophic falsity component (*f*)
88 (correct class: versicolor)	0.65	Instances have equal degree of membership in versicolor and virginica, which leads to ambiguity.	Iris-versicolor-*t* = 0.5	Iris-versicolor-virginica-*i* = 0.65	Iris-versicolor-virginica-*f* = 0.65
134 (correct class: virginica)			Iris-versicolor-*t* = 0.5	0.5	0.5
Interpretation 88: Fuzzy results indicate ambiguity, but neutrosophic results is (1,1,0.5); indicating clear membership to versicolor, also indeterminacy recorded for this instance is high as 1, and falsity is 0.5. 134: Fuzzy results indicate ambiguity, but neutrosophic results is (1,0,0); indicating clear membership to versicolor, also indeterminacy and falsity recorded for this instance is nil; which is not in accordance with the desired result. Neutrosophic component is generating versicolor because 134th instance specifications are covered by the rule designed for versicolor (for truth, indeterminacy and falsity component).					
84	0.676	Instances report higher membership values to virginica which is wrong class but versicolor is actual class	Iris-versicolor-*t* = 0.5	0.5	0.5
86			Iris-versicolor-*t* = 0.5	Iris-versicolor-virginica-*i* = 0.65	Iris-versicolor-virginica-*f* = 0.65
Interpretation 84: Fuzzy results indicate wrong higher membership to virginica, but neutrosophic result is (1,0,0); indicating clear membership to versicolor, also indeterminacy an falsity recorded for this instance is nil. 86: Fuzzy results indicate wrong higher membership to virginica, but neutrosophic result is (1,1,0.5); indicating clear membership to versicolor, also indeterminacy recorded for this instance is high as 1, and falsity is 0.5.					

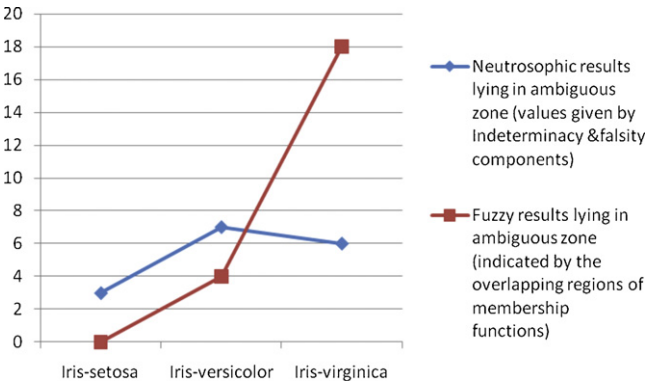

Fig. 12. Comparison of ambiguous results given by FIS and NIS.

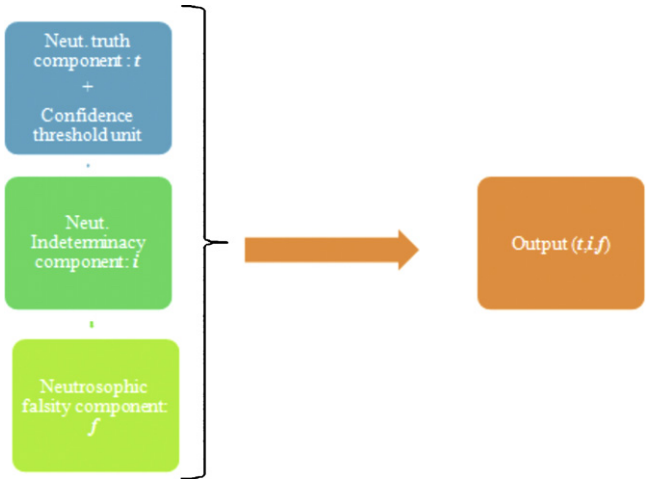

Fig. 13. Block diagram representing consolidation of independent results in neutrosophic format (t, i, f).

Table 6 gives the details of the instances in which indeterminacy and falsity values were recorded using neutrosophic indeterminacy and falsity component, respectively, against the values obtained from conventional FIS.

Table 7 does the comparison of wrong fuzzy results with neutrosophic results.

9. Evaluation of results

Fig. 12 gives the analysis of the results of testing done using neutrosophic classifier in comparison to the results generated by fuzzy classifier. It particularly gives the result analysis of the values lying in ambiguous zone (values given by indeterminacy and falsity in neutrosophic logic and results lying in overlapping regions for fuzzy logic).

Results generated by FIS and NIS are labeled as follows:

a. *Non-ambiguous*: FIS results that lie in single output membership function indicate clear belongingness to a particular class, hence non-ambiguous. NIS results which have zero indeterminacy and falsity component associated are clear non-ambiguous results.
b. *Ambiguous*: For the results generated by FIS, if they lie in the overlapping range of two adjacent membership functions, this indicates certain degree of ambiguity associated with it; hence are ambiguous. If the results generated by NIS have falsity and indeterminacy values, this indicates ambiguity in the results generated.

For the ambiguous results generated by NIS, a confidence value can be defined for the truth component. For example if for truth component the confidence value set $\geq 50\%$, then for the truth exceeding the confidence threshold, the associated indeterminacy and falsity values should be considered insignificant; else the result generated for the given instance has significant proportion of indeterminacy and/or falsity associated with it and would call for human expert intervention for final interpretation. Here in this paper confidence value $\geq 50\%$ is set for truth component, as shown in Fig. 13.

$$i/f = \begin{cases} \text{insignificant} & \text{if } t \geq 50\% \\ \text{significant} & \text{if } t < 50\% \end{cases} \quad (11)$$

Fig. 14 discusses the final non-ambiguous and ambiguous results once confidence value is taken into consideration. Conventional fuzzy classifier generated total 22 ambiguous results for 60 testing instances, which constitutes 36.6%. This is quite contrary

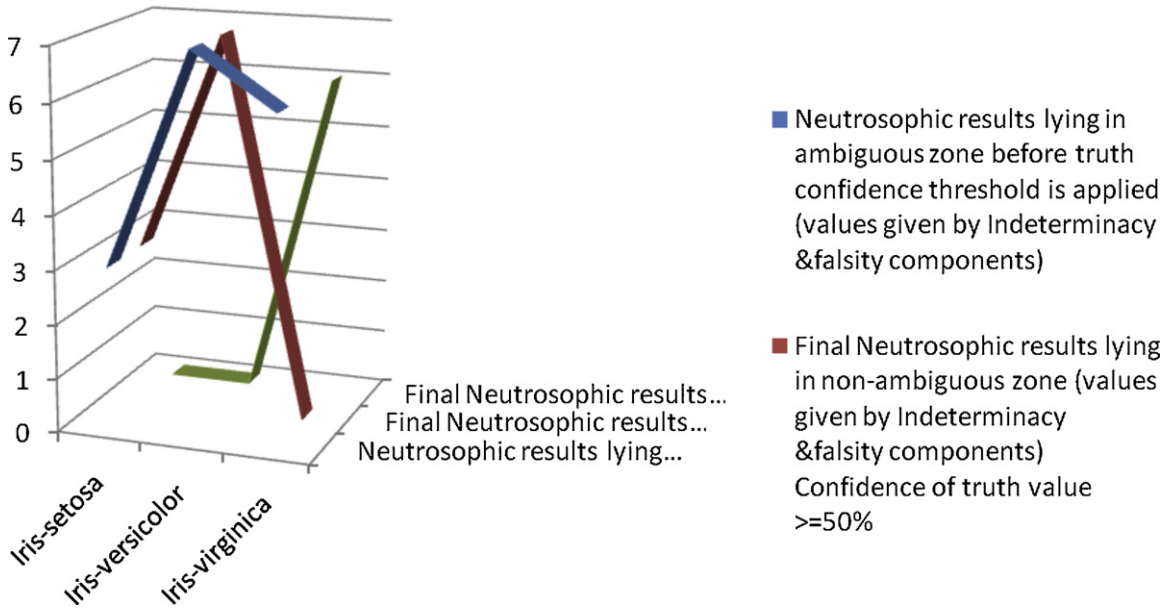

Fig. 14. Comparison of ambiguous and non-ambiguous results generated by NIS after being filtered from the truth confidence threshold limit.

to 6 ambiguous results out of 60 testing instance (which is 10%), that lie in the ambiguous zone for which human expert intervention is seeked for final interpretation of the result rest all results (except for 134th instance) generated by neutrosophic classifier are in accordance with the desired results.

10. Conclusions

As the proposed neutrosophic approach partitions the pattern space into non-overlapping decision regions for pattern classification so both the complexity and computational load of the classifier are reduced and thus the training time and classification time are extremely short. Although the decision regions are partitioned into non-overlapping subspaces, we can achieve good classification performance since the decision regions can be correctly determined via our proposed neutrosophic approach. Furthermore as the results generated by neutrosophic classifier has three components of truth, indeterminacy and falsity so the neutrosophic classifier would be a special system which would be more generalized and indeterminacy tolerant in its working as compared to the fuzzy counterparts; though neutrosophic systems classifiers as proposed would vary substantially according to the nature of the control problems that they are supposed to solve. Here we have confined ourselves to the explanation of relatively simple classifier problem.

11. Future directions

Results shown in the paper are encouraging so in future proposed extension of fuzzy classifier that is neutrosophic classifier can be extended by exploring more complicated domains in which indeterminacy and falsity is tightly integrated in the data captured. If after detailed investigation strong correlation is found between human reasoning and neutrosophic classifier results then definitely a real time application exploiting neutrosophic logic can be developed; possibly replacing existing conventional fuzzy classifier systems. Also with the optimistic results, possible integration of neutrosophic logic with other soft computing domains like neural network and genetic algorithm can also be tried.

References

[1] J.-M. Glubrecht, A. Oberschelp, G. Todt, Klassenlogik, Bibliographisches Institut, Mannheim/Wien/Zürich, 1983, ISBN:3-411-01634-5.
[2] C.A. Kulikowski, S.M. Weiss, Computer Systems That Learn: Classification and Prediction Methods from Statistics, Neural Nets, Machine Learning, and Expert Systems. Machine Learning, Morgan Kaufmann Publishers, San Francisco, 1991, ISBN:1-55860-065-5.
[3] G.J. Klir, U.H. St Clair, B. Yuan, Fuzzy Set Theory: Foundations and Applications, Prentice Hall, Englewood Cliffs, NJ, 1997, ISBN:0-13-341058-7.
[4] H. Zimmermann, Fuzzy Set Theory and its Applications, Kluwer Academic Publishers, Boston, 2001, ISBN:0-7923-7435-5.
[5] L.A. Zadeh, Fuzzy sets, Information and Control 8 (1965) 338–353.
[6] L.A. Zadeh, The role of fuzzy logic in the management of uncertainty in expert systems, Fuzzy Sets and Systems 11 (1983) 197–227.
[7] L.A. Zadeh, Outline of a new approach to the analysis of complex systems and decision processes, IEEE Transactions on Systems, Man and Cybernetics 3 (1973) 28–44.
[8] L.A. Zadeh, A theory of approximate reasoning Machine Intelligence, 9, John Wiley & Sons, New York, 1979, pp. 149–194.
[9] J.C. Bezdek, Fuzzy models for pattern recognition: background, significance and key points, in: J.C. Bezdek, S.K. Pal (Eds.), Fuzzy Models for Pattern Recognition, IEEE Press, New York, 1992, pp. 1–27.
[10] M. Roubens, Pattern classification problems and fuzzy sets, Fuzzy Sets and Systems 1 (1978) 239–253.
[11] N. Del Amo, A. Montero, J.V. Cutello, On the principles of fuzzy classification, in: Proc. 18th North American Fuzzy Information Processing Society Annual Conf., 1999, S. 675–679.
[12] J.C. Bezdek, J.M. Keller, R. Krishnapuram, N.R. Pal, Fuzzy Models and Algorithms for Pattern Recognition and Image Processing, Kluwer Academic Publishers, 1999.
[13] F. Smarandache, Linguistic Paradoxists and Tautologies, Libertas Mathematica, vol. XIX, University of Texas at Arlington, 1999, pp. 143–154.
[14] F. Smarandache, A unifying field in logics: neutrosophic logic, Multiple-Valued Logic/An International Journal 8 (3) (2002) 385–438.
[15] F. Smarandache (Ed.), Proceedings of the First International Conference on Neutrosophy, Neutrosophic Logic, Neutrosophic Set, Neutrosophic Probability and Statistics, University of New Mexico, Gallup Campus, Xiquan, Phoenix, 2002, 147 pp.
[16] F. Smarandache, Neutrosophy, a new branch of philosophy, Multiple-Valued Logic/An International Journal 8 (3) (2002) 297–384.
[17] F. Smarandache, Definition of neutrosophic logic—a generalization of the intuitionistic fuzzy logic, in: Proceedings of the Third Conference of the European Society for FuzzyLogic and Technology, EUSFLAT 2003, University of Applied Sciences at Zittau/Goerlitz, Zittau, Germany, September 10–12, 2003, 2003, pp. 141–146.
[18] Fuzzy Logic Toolbox User's Guide, The MathWorks Inc., 2009.
[19] E.H. Mamdami, S. Assilina, An experiment in linguistic synthesis with a fuzzy logic controller, International Journal of Man-Machine Studies 7 (1) (1975) 1–13.

A.Q. Ansari is working as a professor and HOD of the department of electrical engineering at Jamia Millia Islamia, New Delhi. His area of research and specialization are computer networking and data communication, image processing, networks on chip and soft computing.

Ranjit Biswas is working as a professor in Department of Computer Science at Jamia Hamdard University, New Delhi. His areas of research and specialization are rough set technology & applications, fuzzy logic & applications, soft sets & systems, AI and graph theory.

Swati Aggarwal is a pursuing PhD at Jamia Millia Islamia, New Delhi. She has done BTech (computer science) in 2001 and MTech (IT) in 2005. Currently she is working as assistant professor in computer science department at ITM University, Gurgaon. Her interests include neutrosophic logic, fuzzy sets, artificial intelligence, neural network and soft computing.