

Interval-Valued Fuzzy Decision Trees with Optimal Neighbourhood Perimeter

Youthachai Lertworapachaya^a, Yingjie Yang^{a,*}, Robert John^b

^a*School of Computer Science and Informatics, De Montfort University, The Gateway, Leicester, LE1 9BH, UK*

^b*Automated Scheduling, Optimisation and Planning Group (ASAP), School of Computer Science, University of Nottingham, Jubilee Campus, Nottingham, NG8 1BB, UK*

Abstract

This research proposes a new model for constructing decision trees using interval-valued fuzzy membership values. Most existing fuzzy decision trees do not consider the uncertainty associated with their membership values, however, precise values of fuzzy membership values are not always possible. In this paper, we represent fuzzy membership values as intervals to model uncertainty and employ the look-ahead based fuzzy decision tree induction method to construct decision trees. We also investigate the significance of different neighbourhood values and define a new parameter insensitive to specific data sets using fuzzy sets. Some examples are provided to demonstrate the effectiveness of the approach.

Keywords: Look-ahead based fuzzy decision tree induction, optimal perimeter, interval-valued fuzzy decision trees.

1. Introduction

Decision tree is a powerful induction method in data mining. However, real-world applications of decision trees exhibit uncertainty through imprecise data, vagueness, ambiguity etc [1–4, 25, 27]. To deal with these uncertainties, Fuzzy Decision Trees (FDT) employing type-1 fuzzy sets have been extensively investigated [11, 12, 22]. However type-1 fuzzy sets, by their very nature, require

*Corresponding author

Email addresses: youd@dmu.ac.uk (Youthachai Lertworapachaya), yyang@dmu.ac.uk (Yingjie Yang), Robert.John@nottingham.ac.uk (Robert John)

precise values in their application and determining the exact membership values in type-1 fuzzy sets is well known to be difficult [19]. In real world data, however, we may have information on the boundary of values but cannot determine where it is within the boundaries. In this case, interval values are usually adopted to represent the uncertainty. The present FDT cannot deal with these kind of interval values. There have been applications of type-2 fuzzy sets in decision tree construction [21]. However, these methods are designed to work with the same data as type-1 models. Therefore, there is a need to investigate the right way to treat with intervals in data mining problems. As an extension of the type-1 fuzzy sets, interval-valued fuzzy sets [5–7] are specifically designed to deal with fuzzy sets with interval representation of values. It is obviously a natural choice to deal with interval values in FDT problems. Here, we apply interval-valued fuzzy sets to construct an interval-valued fuzzy decision tree. In this way, the interval values in a data mining problem does not need to be converted into an average to employ FDT, instead, they can be directly mapped into an interval-valued fuzzy set and then an interval-valued fuzzy decision tree can be established without any aggregation operation. It is well known that aggregation can lose information, and an interval-valued fuzzy decision tree will certainly convey more information than a traditional FDT, hence the ability to reveal more useful knowledge from the data than a FDT. Obviously, such an interval-valued fuzzy decision tree is more powerful in dealing with data mining problems involving interval representation.

In addition to the novel interval-valued fuzzy decision trees, we present also a new way to determine the parameters of FDT in look-ahead based fuzzy decision trees (LAFDT). LAFDT was proposed by Dong and Kothari (2001) for data represented with type-1 fuzzy sets [11]. It can evaluate the classifiability of instances, that are split along branches of a given node based on evaluating the texture of the class label surface. This particular method works by finding the instances that are within a distance threshold from a given instance. This distance threshold has significant influence on the results of the classification and

depends on the data sets, however, there is no well developed methodology so far to determine its value[5]. In this paper, we investigate the significance of this threshold and define a new parameter which is not sensitive to a specific data set. In this way, the application of LAFDT is greatly simplified.

Combining interval-valued fuzzy sets and the LAFDT with the proposed new parameter, we propose a new model to construct a decision tree using interval-valued fuzzy membership values based on LAFDT induction and interval-valued fuzzy sets with a new optimal parameter [34]. We call this the Look-Ahead Based Interval-Valued Fuzzy Decision Tree with Optimal Perimeter of the Neighbourhood (LAIVFDT-OPN). This new model simplifies the application of LAFDT by employing a parameter insensitive to data set changes on one hand, and improves the classification quality on the other hand by means of the employment of the full information in interval representation through interval-valued fuzzy sets.

To demonstrate the feasibility of our proposed model, two well known data sets on weather condition for play [11] and car evaluation [22] are employed to verify the effectiveness of the proposed new threshold parameter. Although they have different data sizes and require different threshold values defined in LAFDT, but our result shows that they share the same threshold parameter defined in our model. To validate our interval-valued fuzzy decision tree model, the weather data set is converted into an interval data set by expanding each value into an interval. Then LAFDT is applied to the original weather data set, and LAIVFDT-OPN is applied to the interval data set of weather data. The comparison shows that their results are different, and LAIVFDT-OPN obtains a smaller tree in comparison with LAFDT. It demonstrates that the reservation of interval values provide more information than otherwise. In the end, we applied the proposed model to a real world data set from a factory in Thailand to demonstrate its usage.

Section 2 provides some necessary background material on LAFDT, and interval-valued fuzzy sets (IVFS). In Section 3 we propose an extension of LAFDT induction to LAIVFDT-OPN. Section 4 describes the effect of the neighbourhood parameter on decision tree construction and Section 5 demonstrates an example of the application of LAIVFDT to data with uncertain fuzzy membership values. In Section VI we present results of the experiment on a real world case study of factory data. Section 6 summaries the results of the study.

2. Preliminaries

2.1. The Look-Ahead Based Fuzzy Decision Tree

There are many different models of FDT [1, 13]. LAFDT induction is one of the more recent models employed to evaluate the classifiability of the instances along each branch of the split by linear discriminant (also called a single step) [1, 10, 11]. In a FDT, the key is to find the appropriate attribute to split samples into different branches along the tree [1, 15, 16]. The LAFDT has a particular method of evaluating the classifiability of attributes along the branches of a node to split and produce a smaller decision tree. A nonparametric method in LAFDT is to characterise the classifiability of attributes using an occurrence matrix.

The occurrence matrix is deployed to characterise the texture of the class label [11] (A data set consists of variables and one for the class label or $(n+1)$ dimension). The usual approach of LAFDT is, for any instance x , to measure a distance r between two instances that are within a circular neighbourhood of radius r based on the distance in Equation (1). The r distance assists in filtering the instances which exceed the radius r , and to reduce computing time. Considering the universe of objects described by n attributes, an attribute has values of fuzzy subsets $A_1^k, A_2^k, \dots, A_{m_k}^k$ (For the list of symbols adopted in this paper, please refer to appendix). The distance between two objects (or instance

x and y) can be measured using their fuzzy memberships.

Definition 1: [11] Let $\mu_i^{(k)}(x)$ ($1 \leq k \leq n, 1 \leq i \leq m_k, 1 \leq x \leq N$) denote the membership value of instance x for the i^{th} value of the k^{th} attribute. The distance between instance x and y is defined by

$$D_{xy} = \sum_{k=1}^n \sum_{i=1}^{m_k} |\mu_i^{(k)}(x) - \mu_i^{(k)}(y)|. \quad (1)$$

For any object x in the universe, we can restrict its circular neighbourhood to those objects within a radius r of x . Then local occurrence matrix P for object x is defined as follows.

Definition 2: [11, 12] Let $\mu_j(x)$, $1 \leq j \leq C$ denote the membership value of instance x for class j and let $\mu_j(x) = [\mu_1(x), \dots, \mu_C(x)]$. The local co-occurrence matrix of instance x is defined by

$$P(x) = \sum_{y, D_{xy} \leq r} \mu(x)^T \times \mu(y). \quad (2)$$

where, $\mu(x)^T$ is a transpose matrix and r is the neighbourhood radius of x .

With a local occurrence matrix, we can derive the co-occurrence matrix for each attribute.

Definition 3: [11, 12] The local co-occurrence matrix after attribute k is selected.

$$W^{(k)} = \sum_{i=1}^{m_k} \sum_x P(x). \quad (3)$$

Then, the classifiability of attribute k is

$$L^{(k)} = \sum_{i=1}^C W_{ii}^{(k)} - \sum_{i=1}^C \sum_{j=1, i \neq j}^C W_{ij}^{(k)}. \quad (4)$$

According to the values of $L^{(k)}$, we can identify the attribute with the highest classifiability in order to build a decision tree.

2.2. Interval-Valued Fuzzy Sets

In Definition 2, the membership values are required to be precise values. It is not always possible to have precise membership values, instead, we may restrict a membership value to an interval and replace a type-1 fuzzy set with an interval-valued fuzzy set.

Definition 4: [5, 7, 18] Let X denote a universe of discourse. An interval-valued fuzzy set is an expression A denoted by

$$A = \{(x_i, \mu_A(x_i)) | x_i \in X; i = 1, 2, \dots, n\}. \quad (5)$$

where $\mu_A(x_i) : X \rightarrow D([0, 1])$, and $x_i \rightarrow \mu_A(x_i) = [\underline{\mu}_A(x_i), \bar{\mu}_A(x_i)] \in D([0, 1])$.

If we represent the interval relationship with $\underline{\mu}_A(x_i)$ and $\nu_A(x_i) = 1 - \bar{\mu}_A(x_i)$ then we get intuitionistic fuzzy sets [5–7]. The interval of intuitionistic fuzzy sets is denoted by $[\mu_A(x_i), 1 - \nu_A(x_i)]$. In this paper, we transform the intuitionistic fuzzy sets into interval-valued fuzzy sets as follows.

Given $\mu_A(x_i) = [\underline{\mu}_A(x_i), \bar{\mu}_A(x_i)] = [\mu_A(x_i), 1 - \nu_A(x_i)]$. Then we can represent the distance between set A and B in the form of interval-valued fuzzy sets:

$$d = \frac{1}{2} \sum_{i=1}^n [|\underline{\mu}_A(x_i) - \underline{\mu}_B(x_i)| + |\bar{\mu}_A(x_i) - \bar{\mu}_B(x_i)|]. \quad (6)$$

3. Effect of the neighbourhood parameter on decision tree construction

As demonstrated in Equation (2), a predefined parameter neighbourhood radius r is necessary for a given data set in a look-ahead algorithm. Dong et al. (2001) comment that “ r should be large enough such that each instance has a few instances in its neighbourhood r also should be small enough to keep the calculation of co-occurrence matrix local” [12]. Obviously, it is necessary to

investigate the role of neighbourhood r values in constructing a decision tree [33].

To identify the role of neighbourhood r values for different data sets, two data sets are adopted in our experiment here: weather [11] and car [22]. The weather data set is obtained from [11, 33]. There are four attributes: “outlook”, “temperature”, “humidity”, “wind” and one classification attribute: “plan”. Their values are described as fuzzy subsets: the “outlook” can be sunny, cloudy or rain; “temperature” can be hot, mild or cool; “humidity” can be humid or dry; “wind” can be windy or calm and “plan” can be A,B or C. The car evaluation data set comes from [22]. It is comprised of six attributes: “buying”, “maint”, “doors”, “persons”, “lug_boot” and “safety” and one classification attribute: “car evaluation”. The concepts of car acceptability can be described by: “buying” for purchase price, “maint” for price of maintenance, “doors” for the number of doors, “Persons” for capacity in terms of persons to carry, “lug_boot” for the size of luggage and “safety” for estimated safety of the car. The attribute values of each attribute and classification attribute are described as fuzzy subsets: “buying” and “maint” can be very high, high, medium or low; “doors” can be 2, 3, 4 or 5-more; “persons” can be 2, 4 or more; “lug_boot” can be small, medium or big and “safety” can be low, medium and high and “car evaluation” can be unacceptable, acceptable, good or very good [22]. Both data sets are represented by their corresponding fuzzy membership values for the subsets of their attributes. There are 16 instances in the weather data set and 87 instances in the car evaluation data set. Based on these data sets, we can evaluate the effect of neighbourhood parameter on decision tree construction.

Based on the Equations (1-5), we got $L^{(k)}$ values for each attribute in each data set under different neighbourhood r values. The results are shown in Figure 1 and 2.

From Figure 1 and 2, it is obvious that the dominance of attributes can be classified into two different regions with respect to r values. In Figure 1, wind has the larger $L^{(k)}$ value when $r < 5$, but temperature gets a larger value when ≥ 5 . Similar situations happen in Figure 2. Obviously, the selection of r value

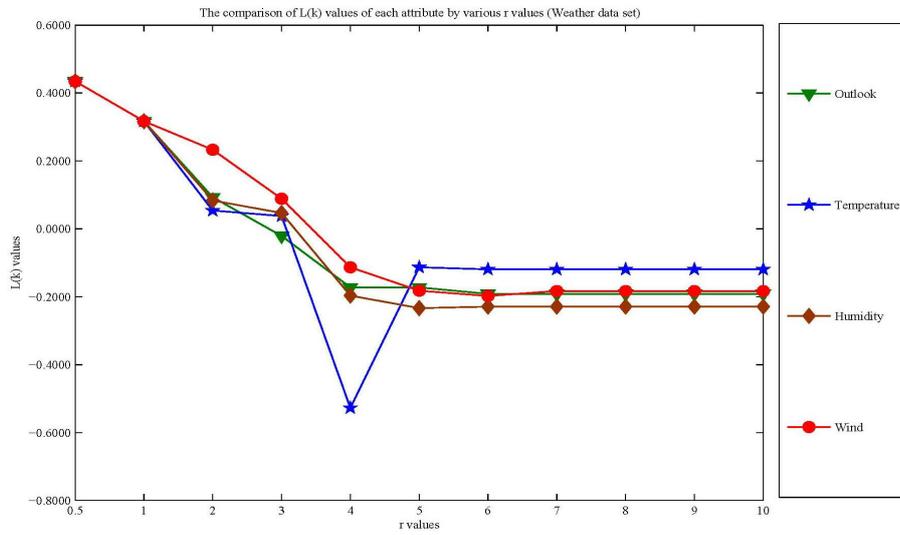


Figure 1: $L^{(k)}$ and r values classify by attributes (Weather data set)

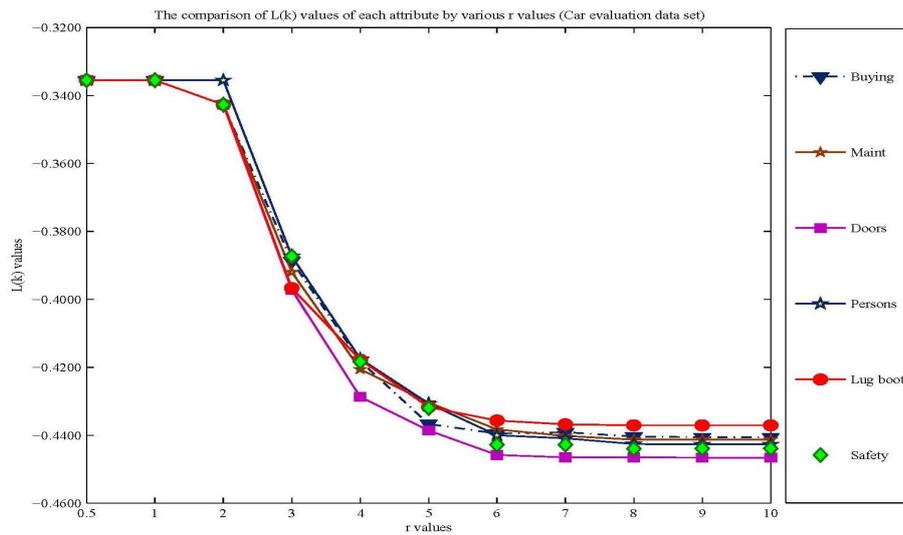


Figure 2: $L^{(k)}$ and r values classify by attributes (Car data set)

can change the preferred attribute for the same data set, and we have to know which r value is the ideal one.

For the weather data set, the ideal r value is 3 according to [11]. In this sense, the preferred attribute to split the tree in the root node is wind. It is

obvious that a less effective tree would be constructed if we selected a r value from the region where $r \geq 5$. This fact tells us that we have to know the right region before we determine the r value. In the weather data set, it is the region where $1 < r < 5$. In this region, the only r value satisfying “large enough” and “small enough” in the same time as required by Ming Dong [12] is 3. In weather data set, the smallest tree is established when $r = 3$.

Similar to the weather data set, there are two different regions of r values in the car evaluation data set as shown in Figure 8: one for $r \leq 5$ and the other for $r > 5$. Persons has the larger $L^{(k)}$ values when $r \leq 5$. However, Lug_boot would be selected when $r > 5$. We could not construct the tree by select a r value from the region where $0.5 < r < 1$ due to the equal $L^{(k)}$ values for all attributes. It is obvious that any r value between 1 and 5 will give the same dominate attribute. As an example, we tested $r = 2$ and $r = 8$, and the result tree has 87 nodes for $r = 2$, but 83 nodes for $r = 8$. Thus, Lug_boot appears as a better choice to split the root node compared with persons. Obviously, the region where $r > 5$ is preferred in the car data set. $r = 8$ looks fine with the “large enough” and “small enough” principle in the second region. In this case, we tested only some r values between 0.5 and 10. There is no reason to stop at 10. Then the question is how to know which region is ideal and the size of each region.

Obviously, the value of r has a significant impact on the structure of the tree. In fact, for each specific data set, there exists an ideal r value which helps to split the samples to construct a tree with a relatively small size. When the r value is too small, there are less samples in the neighbourhood belonging to the same class, so the tree would be very big; however, when r value is too big, most samples will go to one class, which cannot differentiate different classes. Therefore, it is essential here to find a right value for r , which cannot be too small or too big. The r value depends on the size of the data, which makes it more difficult to know the right value of r for each data set. So far, there is no clear way to identify the r value, and it has become a bottleneck for the

application of LAFDT. An alternative parameter insensitive to the change of data size would certainly be a significant help in this situation.

Considering the role of r value, it defines the neighbourhood of a sample x . If we consider those samples within the r distance as a neighbour set of x , we can actually define a fuzzy set for x 's neighbourhood, and an α cut of this fuzzy set could obviously determine the size of this neighbourhood.

Let S be the universe of samples (objects) $S=\{x_1, x_2, \dots, x_n\}$ and $x_c \in S$. The neighbourhoods of x_c is a subset $A^*(x_c) \subseteq S : A^*(x_c)=\{\text{neighbourhood of } x_c\}$. Obviously, $A^*(x_c)$ can be described by $A^*(x_c) = \{x_i \mid x_i \in S, 0 \leq \bar{\mu}_c(x_i) \leq 1\}$. Its fuzzy membership value could be derived according to the distance between x_i and $x_c : i = 1, 2, \dots, n$. For a given distance $D_{x_i x_c}$, we have $\bar{\mu}_c(x_i)$, which is a membership value for $x_i \in A^*(x_c)$.

$$\bar{\mu}_c(x_i) = \frac{D - D_{x_i x_c}}{D}. \quad (7)$$

Here D is the boundary value of $D_{x_i x_c}$. For example, we could use $D_{x_i x_{c \min(max)}}$, then $D = D_{x_i x_{c \min(max)}}$. $D_{x_i x_{c \min(max)}}$ is the minimum of the maximum values of $D_{x_i x_c}$ for all x_i in $A^*(x_c)$.

It is clear that when $\bar{\mu}_c(x_i)$ approaches to 1, x_i is near to x_c . In this sense, we can identify the neighbours of x_c using $\bar{\mu}_c(x_i)$. For a set $B^*(x_c) \subseteq A^*(x_c)$, if $x_i \in B^*(x_c)$, then x_i is near to x_c with a membership value higher than a given value α . Clearly, this is α -cut of $A^*(x_c)$. The set $B^*(x_c)$ can be represented as $B^*(x_c) = \{x_i \mid x_i \in S, \bar{\mu}_c(x_i) \geq \alpha\}; i = 1, \dots, n$. We call $B^*(x_c)$ as the **neighbourhood set** of x_c .

For a given optimal value of r , it is clear that we could find a corresponding value of α . We can get an α -cut equivalent to the neighbourhood set obtained according to neighbourhood radius r in LAFDT. The samples of set $A^*(x_c)$ are associated with a family of crisp subsets and the restriction of samples is that their membership values are greater than or equal to some chosen value

α in $[0,1]$ [25]. Consequently, we obtain a crisp subset $B^*(x_c)$. In this way, α can play the same role as r , and we can replace r with α in LAVIFDT-OPN. The r value depends on the data set. There is no bound to r value. However, α is bound within 0 and 1. Thus, we can replace the neighbourhood r with α in Equation (2). Through the known r values for some data sets, we can find the corresponding α values, and if its change is much less than r values, then it is obviously more convenient to use α value instead of r value. There is no systematic method for the selection of r value at the moment, but there are some well known data sets with r values which could be identified through comparative study in experiment. So, we can find α values which are equivalent to those r values, and if the ideal α values do not change significantly with a significant change of the ideal r values, then we can set that α value as the optimised one.

4. Look-Ahead Based Interval-Valued Fuzzy Decision Tree with Optimal Perimeter of the Neighbourhood

As discussed in section II, there are situations where a precise fuzzy membership value is not available but a restriction of the possible values into an interval is possible. Hence it is necessary to introduce interval-valued fuzzy sets into fuzzy decision trees. Such an extension will further complicate the identification of suitable neighbourhood parameter and a parameter insensitive to the change of data is required. Here, an interval-valued fuzzy decision tree with optimal parameter of neighbourhood (LAIVFDT-OPN) is developed to further enhance the existing LAFDT.

The key step in LAFDT is the calculation of the distance between two samples or instances. In LAIVFDT-OPN, the two samples or instances involved are combined using two interval-valued fuzzy sets with elements of attributes. In LAFDT, the distance between two instances is calculated as the distance between two fuzzy sets, as shown in Equation (1). Obviously, Equation (1) is

not applicable here, and the distance between two interval-valued fuzzy sets in Equation (6) should be applied.

Considering the same universe for attributes and instances in section II, we have the following definition for the distance between two instances.

Definition 5: Let $\mu_i^{(k)}(x) = [\underline{\mu}_i^{(k)}(x), \bar{\mu}_i^{(k)}(x)]$ and $\mu_i^{(k)}(y) = [\underline{\mu}_i^{(k)}(y), \bar{\mu}_i^{(k)}(y)]$; ($1 \leq k \leq n, 1 \leq i \leq m_k$) denote the interval-valued fuzzy membership value of instance x and y for the i^{th} value of k^{th} attribute. The distance between the two instances is

$$\hat{D}_{xy} = \frac{1}{2} \sum_{k=1}^n \sum_{i=1}^{m_k} [|\underline{\mu}_i^{(k)}(x) - \underline{\mu}_i^{(k)}(y)| + |\bar{\mu}_i^{(k)}(x) - \bar{\mu}_i^{(k)}(y)|]. \quad (8)$$

For any instance in the universe, we can restrict its circular neighbourhood to those objects within a radius r of x . Then a local co-occurrence matrix P for object x is defined.

Definition 6: Let $\hat{\mu}_j(x)$, $1 \leq j \leq C$ indicate the interval-valued fuzzy membership value of instance x for class j and let $\hat{\mu}_j(x) = [[\underline{\mu}_1(x), \bar{\mu}_1(x)], \dots, [\underline{\mu}_C(x), \bar{\mu}_C(x)]]$. The local co-occurrence matrix of instance x is

$$\hat{P}(x) = \sum_{y, \bar{\mu}_x(y) \geq \alpha} \hat{\mu}(x)^T \times \hat{\mu}(y). \quad (9)$$

where, $\bar{\mu}_x(y)$ represents the fuzzy membership for y belong to the neighbourhood set of x , $\hat{\mu}(x)^T$ is a transpose matrix and α is the α -cut value for the neighbourhood of x .

For look-ahead based interval-valued fuzzy decision tree (LAIVFDT), the local co-occurrence matrix of instance x is

$$\hat{P}(x) = \sum_{y, D_{xy} \leq r} \hat{\mu}(x)^T \times \hat{\mu}(y). \quad (10)$$

where, r is neighbourhood radius of x .

From Equation (4), let $L_{r=c}^{(k)}$ $\{k = 1, \dots, n; c = 1, \dots, m\}$ represent the corresponding values of L^k for different r values and $L_{\alpha=t}^{(k)}$ $\{k = 1, \dots, n; t = 0, \dots, 1\}$ for different α values. $TL_{(r,\alpha)}$ is an average difference between $L_{r=c}^k$ and $L_{\alpha=t}^k$

for all attributes. We have

$$TL_{(r,\alpha)}|_{r=c,\alpha=t} = \frac{1}{n} \sum_{k=1}^n \left| L_{r=c}^{(k)} - L_{\alpha=t}^{(k)} \right| \quad (11)$$

where c represents r value, $c \geq 0$, $0 \leq t \leq 1$ and $n \geq 1$.

If the minimum value of $TL_{(r,\alpha)}$ is reached at $t = T$, then we get α value: $\alpha = T$. Based on $TL_{(r,\alpha)}$ value we can identify the ideal α value.

According to the $\hat{P}(x)$ matrix, each matrix element is represented by an interval value. Schneider et al.(1996) described an interval X as a closed, bounded set of real numbers, in which $\{x | \underline{X} \leq x \leq \overline{X}, x \in X\}$ can be denoted as $X = [\underline{X}, \overline{X}]$ [9]. For all real number $\underline{X}, \overline{X}, \underline{Y}$ and \overline{Y} . Such that $0 \leq \underline{X} \leq \overline{X} \leq 1$ and $0 \leq \underline{Y} \leq \overline{Y} \leq 1$ [8–10]. The rules of interval arithmetic are as follows

- Addition: $[\underline{X}, \overline{X}] + [\underline{Y}, \overline{Y}] = [\underline{X} + \underline{Y}, \overline{X} + \overline{Y}]$.
- Subtraction: $[\underline{X}, \overline{X}] - [\underline{Y}, \overline{Y}] = [\underline{X} - \overline{Y}, \overline{X} - \underline{Y}]$.
- Multiplication: $[\underline{X}, \overline{X}] * [\underline{Y}, \overline{Y}] = [\underline{X} * \underline{Y}, \overline{X} * \overline{Y}]$.
- Division: $[\underline{X}, \overline{X}] / [\underline{Y}, \overline{Y}] = [\underline{X} / \overline{Y}, \overline{X} / \underline{Y}]$.
assuming $0 < \underline{Y}$
- Distribution law:
 $\min([\underline{X}, \overline{X}], [\underline{Y}, \overline{Y}]) = [\min(\underline{X}, \underline{Y}), \min(\overline{X}, \overline{Y})]$ and
 $\max([\underline{X}, \overline{X}], [\underline{Y}, \overline{Y}]) = [\max(\underline{X}, \underline{Y}), \max(\overline{X}, \overline{Y})]$.

Note that the operation $[\underline{X}, \overline{X}] / [\underline{Y}, \overline{Y}]$ is undefined, if $\underline{Y} = 0$, $\overline{Y} = 0$, or if both $\underline{Y} = 0$ and $\overline{Y} = 0$ [9].

Therefore, the rules of interval arithmetic above are employed for calculating the $\hat{P}(x)$ matrix in Equation (9), the $\hat{W}^{(k)}$ matrix in Equation (12) and the $\hat{L}^{(k)}$ matrix in Equation (13). With the local occurrence matrix, we can derive the co-occurrence matrix for each attribute.

Definition 7: The local co-occurrence matrix for attribute k is selected as follows:

$$\hat{W}^{(k)} = \sum_{i=1}^{m_k} \sum_x \hat{P}(x). \quad (12)$$

then, the classifiability of attribute k is

$$\hat{L}^{(k)} = \sum_{i=1}^C w'_{ii}^{(k)} - \sum_{i=1}^C \sum_{j=1, i \neq j} w'_{ij}^{(k)}. \quad (13)$$

where, $w'_{ii}^{(k)}$ and $w'_{ij}^{(k)}$ are elements of $\hat{W}(x)$. C is the number of fuzzy subsets of the classification attribute.

For classification of $L^{(k)}$ values, we can identify the attribute k with the highest classifiability to build an interval-valued fuzzy decision tree. For LAIVFDT-OPN, it is worked out comparing two or more $L^{(k)}$ values and the highest single value is chosen. When the membership values are represented by intervals, the $\hat{L}^{(k)}$ value is an interval. The comparison of $\hat{L}^{(k)}$ values is not as simple as other values. In this case, the values of $\hat{L}^{(k)}$ are compared through their probability [9]. This probability is used to consider the chance of the occurrence of a sample x and y in the intervals.

For example, let $X=[x, \bar{x}]$ and $Y=[y, \bar{y}]$ denote the interval of X and Y , respectively. Suppose that the sample x is in the interval X and y is in the interval Y . The relationship between x and y can be either $x < y$, $x = y$ or $x \geq y$. For a complete discussion of the relationship between x and y , Schneidel et al. (1996) evaluated $P(x \leq y)$, $P(x > y)$ and $P(x \geq y)$ as follows [9]. $P(x \leq y)$ is derived from $P(x < y)$ or the probability of $x = y$ within the interval of intersection $X \cap Y$. It is denoted by $P(x \leq y) = P(x < y) + P(x = y)$ or $P(x \leq y) = P(x < y)$. For the probability of $x > y$ and $x \geq y$, $P(x > y)$ is obtained by $P(x > y) = 1 - P(x \leq y)$ and $P(x \geq y)$ is derived from $P(x \geq y) = 1 - P(x < y)$. Only the probability of $x < y$ is needed in this context. The probability of $x < y$ is split into three types: $P(x < y)_I$, $P(x < y)_P$ and $P(x < y)_F$ [9].

1. $P(x < y)_I$: x and y are in $X_I = X \cap Y$ and $x < y$ is within the intersection.

2. $P(x < y)_P$: x precedes the intersecting intervals .
3. $P(x < y)_F$: x is in the intersecting interval and y follows.

According to [9], there are six possible cases of the probability $P(x < y)$. The six possible situations could be represented using probability $P(x < y)$ as follows [9].

1. If x precedes y entirely, there is no overlapping. $(X \cap Y) = \Phi$ and X precedes Y then $P(x < y) = 1$ (Figure 1).
2. If $(X \cap Y) \neq \Phi$ and $(x < y)$ then $P(x < y) = P(x < y)_I + P(x < y)_P + P(x < y)_F$ (Figure 2).
3. If $(x < y)$ and x precedes $(X \cap Y)$ then $P(x < y) = P(x < y)_I + P(x < y)_P$ (Figure 3).
4. If $(x < y)$ and x is in $(X \cap Y)$ and y follows then $P(x < y) = P(x < y)_I + P(x < y)_F$ (Figure 4).
5. If $(x < y)$ is in $(X \cap Y)$ then $P(x < y) = P(x < y)_I$ (Figure 5).
6. If x follows y entirely, there is no overlapping. $(X \cap Y) = \Phi$ and X follows Y . Then $P(x < y) = 0$ (Figure 6).

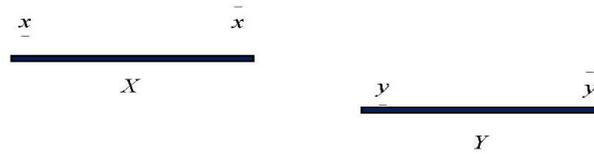


Figure 3: The probability of $P(x < y) = 1$

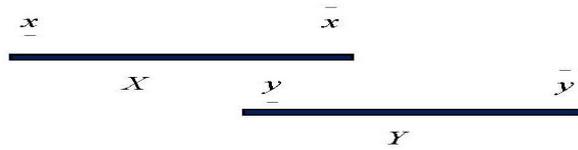


Figure 4: The probability of $P(x < y) = P(x < y)_I + P(x < y)_P + P(x < y)_F$

In general, the equation of $P(x < y)$ is $P(x < y) = P(x < y)_I + P(x < y)_P + P(x < y)_F$. Obviously, if $P(x < y) > 0.5$, it means that there is greater

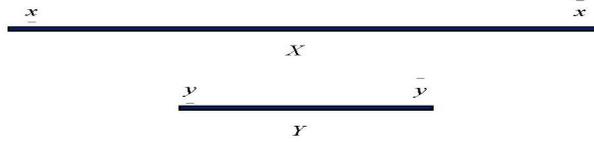


Figure 5: The probability of $P(x < y) = P(x < y)_I + P(x < y)_P$

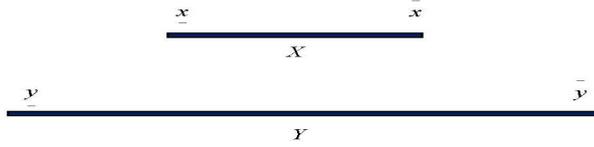


Figure 6: The probability of $P(x < y) = P(x < y)_I + P(x < y)_F$

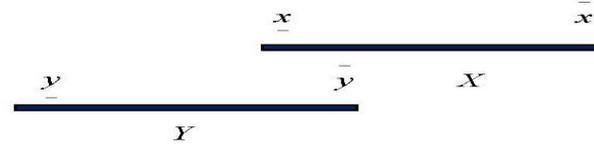


Figure 7: The probability of $P(x < y) = P(x < y)_I$

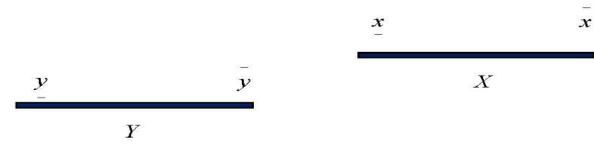


Figure 8: The probability of $P(x < y) = 0$

opportunity for $x < y$ then $x > y$. In this case, we can consider $x < y$ more possible than $x > y$. Then the attribute represented by y should be have priority over the attribute. The probability of $P(x < y)$ is employed to find the maximum $\hat{L}^{(k)}$ value.

Therefore, a fuzzy decision tree can be constructed by the following algorithm.

Step 1: Fuzzify the training data and testing data into interval-valued fuzzy sets.

Step 2: Compute the distance \hat{D}_{xy} between all instance x and y by Equation 8.

Step 3: Calculate the local co-occurrence matrix $\hat{P}(x)$ by comparison with the α value which is computed by Equation 9. Compute the local co-occurrence matrix $\hat{P}(x)$ by Equation 9. $\hat{P}(x)$ is subject to the restriction of α .

Step 4: Select an attribute and sum the local co-occurrence matrix ($\hat{W}^{(k)}$) along each branch using Equation 12.

step 5: Normalise the matrix and calculate $\hat{L}^{(k)}$ using equation 13.

Step 6: Repeating step 4 to step 5 for all attributes.

Step 7: The attribute with the maximum probability for having greater $\hat{L}^{(k)}$ than others is selected for the corresponding node to split the sample set into next layer branches.

7.1 For the root node, select the attribute with the highest possibility for having a greater value of the look-ahead term $\hat{L}^{(k)}$ than others.

7.2 For each child node, the attribute with the highest possibility to have a greater $\hat{L}^{(k)}$ value than that of the left attribute is selected to further split branches of the decision tree.

7.3 The node is a leaf node if enough of the instances corresponds to the same class of classification.

Note:

1. $L^{(k)}$ value and $TL_{(r,\alpha)}$ value are computed by Equation (4) and (11), respectively. The minimum value of $TL_{(r,\alpha)}$ at $\alpha = t$ is selected for the optimal α value.
2. The algorithm of LAIVFDT can be founded on [33].

The proposed algorithm has potentially significant computation complexity. In Step 2, if the number of attributes and their associated value sets are considered as equivalent which can increase by the same size m , the Equation 8 has complexity of $O(m^2)$. However, if we fix the number of attributes and their associated value sets, the complexity of Equation 8 is $O(1)$. In Step 3, if we still refer the size of samples as m , the Equation 9 introduces a computation complexity of $O(m)$. In Step 4, if we fix the sizes of the value subsets of attributes, the computational complexity of Equation 12 is $O(m^2)$. If we consider the size of the value subsets a variable in the same level as m , then we have the complexity as $O(m^3)$. Step 5 involves all the complexity in Step 4, and is compounded by the number of classes. If we consider the number of class as a variable size, then the complexity is even higher than Step 4. Due to this computational complexity, we restrict our applications here to some small size data sets where the computational complexity does not pose serious problems. However, the computational complexity is a potential problem of the proposed algorithm, and we will address it in our future work.

5. Applications

In this section, we verify the applicability of a single α parameter across two different data sets in Section 3 using the traditional LAFDT method firstly, and then demonstrate the feasibility of the proposed LAIVFDT together with the proposed α parameter using the weather data set. In the end, the proposed model is applied to a real world data set (a data set from real world factory). For the first two data sets, the primary aim is to compare their r values and identify their α value, so all samples in the data sets are applied to establish the trees to make comparison. For the real world application data set, we randomly divide the data set into two parts, 100 samples for training to establish the tree, and another 100 samples for testing.

5.1. Feasibility of the parameter α

Based on Equations (8-10) and our two data sets in section IV, some experiments are carried out in this section to demonstrate the feasibility of the proposed parameter α . The results of $TL(r, \alpha)$ is shown in Figure 9 and 10. In Figures 9 and 10, “alpha” refers to α , and each curve represents a different α value.

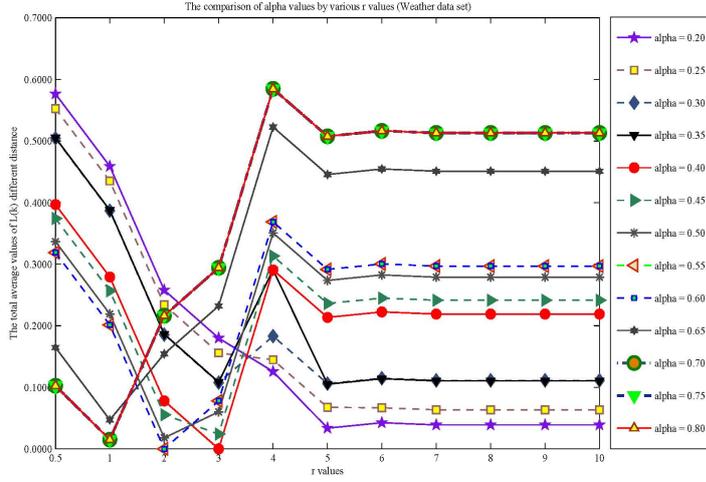


Figure 9: L , $TL^{(k)}$ and r values classify by α values (Weather data set)

Figure 9 gives the result of weather data set. It is clear that $TL(r, \alpha)$ reaches the minimum value when $\alpha = 0.4$ and $r = 3$ or $\alpha = 0.6$ and $r = 2$. It means that we can either select the corresponding α value as $\alpha = 0.4$ or $\alpha = 0.6$. For attribute selection in weather data set, there is no difference between $r = 2$ and $r = 3$, however, there is a different α value associated with each of them. To make a choice between these two values, we have to look at the decision trees constructed from each of them. Table II lists the corresponding experiment results for decision tree construction under different r and α values. Obviously, the decision tree for $\alpha = 0.4$ and $\alpha = 0.6$ are different: $\alpha = 0.4$ results in a smaller decision tree. Thus, the optimal α value is 0.40 for weather data set.

In Figure 10, both $\alpha = 0.4$ and $\alpha = 0.2$ reach the minimum when $r > 6$. α

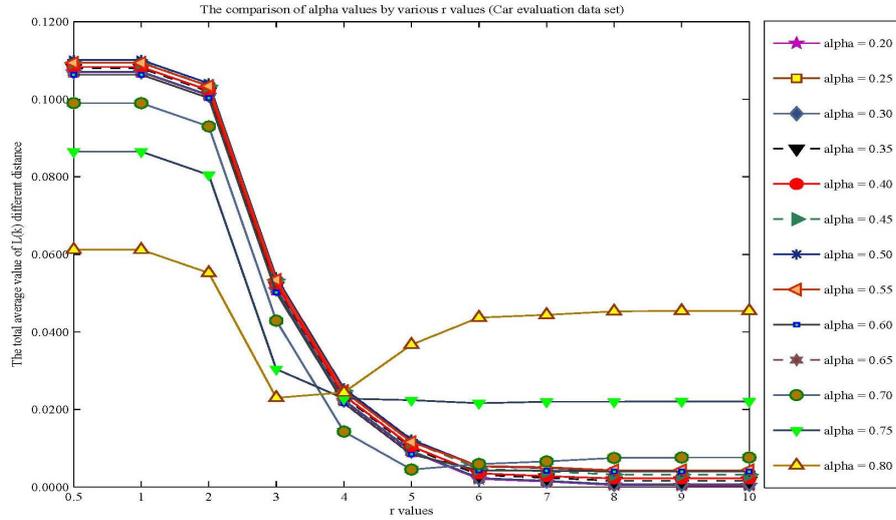


Figure 10: L , $TL^{(k)}$ and r values classify by α values (Car data set)

values could be 0.2 or 0.4 in this case. We constructed decision trees by using r values at $r = 6, 7$ and 8 , and the results in table 1 show that there is no big difference for the decision trees constructed with these values. The r value corresponding to $\alpha = 0.2$ is very large, and include nearly all samples. In that sense, it is not a “local occurrence” anymore. Therefore, $\alpha = 0.4$ should be selected to satisfy the “local” methodology.

Obviously, $\alpha = 0.4$ is valid in both data sets. Considering the fact that α is restricted in $[0,1]$, then its value is stable even if the r value changes significantly. With different data, r could change significantly, but α is relatively stable and it provides an alternative to r in LAFDT.

For LAFDT, the function of r value is to restrict a reasonable local area to construct a simple and effective tree. In this sense, the data size and data distribution should have a significant impact on its value. In comparison with r , α value is determined from a fuzzy set defined on the data set in relation to its size and data distribution. Therefore, for each ideal r value, there should be an ideal α corresponding to that value. Although the r value can change significantly from one data set to another, we expect a relative stable α value

due to its bounded domain $[0, 1]$. The two experiments here have confirmed this expectation as the first step, but we should certainly verify it with more experiments. This paper has just started this comparison, and we expect more test experiments in the future to further confirm it.

Table 1: Comparison of L , TL , r and α values

Data set	r value	L	TL	α value	Nodes	Attribute	Remark
Weather	2	0.2329	0.0000	0.60	19	Wind	* Decision tree could not construct due to the equal $L^{(k)}$ values for all attributes.
	3	0.0886	0.0000	0.40	18	Wind	
	6	-0.1192	0.0425	0.20	19	Temperature	
	7	-0.1192	0.0389	0.20	19	Temperature	
Car	1	-0.3355	0.0612	0.80	*	*	
	2	-0.3427	0.0552	0.80	87	Persons	
	6	-0.4357	0.0021	0.20	83	Lug_boot	
	7	-0.4368	0.0028	0.40	83	Lug_boot	
	8	-0.4371	0.0022	0.40	83	Lug_boot	

5.2. Feasibility of interval-valued fuzzy decision tree method

Table 2: Interval-valued weather data set

Outlook			Temperature			Humidity		Wind		Plan		
Sunny	Cloudy	Rain	Hot	Mild	Cool	Humid	Dry	Windy	Calm	Plan A	Plan B	Plan C
0.6-0.8	0.0-0.2	0.0-0.1	0.9-1.0	0.0-0.1	0.0-0.1	0.7-0.9	0.1-0.3	0.3-0.5	0.5-0.7	0.0-0.1	0.3-0.5	0.5-0.7
0.2-0.4	0.7-0.9	0.0-0.1	0.5-0.7	0.3-0.5	0.0-0.1	0.0-0.1	0.9-1.0	0.0-0.1	0.9-1.0	0.9-1.0	0.2-0.4	0.0-0.1
0.0-0.1	0.6-0.8	0.2-0.4	0.7-0.9	0.1-0.3	0.0-0.1	0.0-0.2	0.8-1.0	0.1-0.3	0.7-0.9	0.2-0.4	0.5-0.7	0.0-0.2
0.0-0.3	0.6-0.8	0.0-0.2	0.2-0.4	0.6-0.8	0.0-0.1	0.1-0.3	0.7-0.9	0.2-0.4	0.6-0.8	0.8-1.0	0.0-0.2	0.0-0.1
0.0-0.1	0.0-0.2	0.8-1.0	0.6-0.8	0.2-0.4	0.0-0.1	0.4-0.6	0.4-0.6	0.4-0.6	0.4-0.6	0.0-0.1	0.0-0.1	0.9-1.0
0.0-0.1	0.6-0.8	0.2-0.4	0.0-0.1	0.2-0.4	0.6-0.8	0.6-0.8	0.2-0.4	0.3-0.5	0.5-0.7	0.1-0.3	0.0-0.1	0.7-0.9
0.0-0.1	0.2-0.4	0.6-0.8	0.0-0.1	0.0-0.1	0.9-1.0	0.0-0.1	0.9-1.0	0.0-0.2	0.8-1.0	0.0-0.1	0.0-0.1	0.9-1.0
0.0-0.1	0.9-1.0	0.0-0.1	0.0-0.1	0.1-0.3	0.7-0.9	0.1-0.3	0.7-0.9	0.0-0.1	0.9-1.0	0.6-0.8	0.0-0.1	0.2-0.4
0.4-0.6	0.6-0.8	0.0-0.1	0.9-1.0	0.0-0.1	0.0-0.1	0.5-0.7	0.3-0.5	0.6-0.8	0.2-0.4	0.1-0.3	0.7-0.9	0.0-0.1
0.4-0.6	0.5-0.7	0.0-0.1	0.0-0.1	0.2-0.4	0.6-0.8	0.0-0.1	0.9-1.0	0.8-1.0	0.0-0.2	0.0-0.1	0.2-0.4	0.6-0.8
0.6-0.8	0.2-0.4	0.0-0.1	0.9-1.0	0.0-0.1	0.0-0.1	0.9-1.0	0.0-0.1	0.1-0.3	0.7-0.9	0.3-0.5	0.8-1.0	0.0-0.1
0.1-0.3	0.5-0.7	0.1-0.3	0.0-0.1	0.9-1.0	0.0-0.1	0.2-0.4	0.6-0.8	0.2-0.4	0.6-0.8	0.6-0.8	0.1-0.3	0.0-0.2
0.8-1.0	0.0-0.2	0.0-0.1	0.1-0.3	0.7-0.9	0.0-0.1	0.2-0.4	0.8-1.0	0.9-1.0	0.0-0.1	0.0-0.1	0.0-0.1	0.9-1.0
0.0-0.1	0.8-1.0	0.0-0.2	0.0-0.1	0.8-1.0	0.0-0.2	0.0-0.2	0.8-1.0	0.6-0.8	0.2-0.4	0.0-0.1	0.0-0.1	0.9-1.0
0.0-0.1	0.0-0.1	0.9-1.0	0.0-0.1	0.0-0.1	0.9-1.0	0.9-1.0	0.0-0.1	0.7-0.9	0.1-0.3	0.0-0.1	0.0-0.1	0.9-1.0
0.9-1.0	0.0-0.1	0.0-0.1	0.7-0.9	0.4-0.6	0.0-0.1	0.0-0.1	0.9-1.0	0.0-0.1	0.9-1.0	0.3-0.5	0.4-0.6	0.0-0.1

Mendel et al. (2006) suggested that a membership degree $\mu_A(x)$ can be provided by an expert with an appropriate degree $\tilde{\mu}_A(x)$ and a bound Δx describing his uncertainty [17]. For example, an interval of possible values of uncertainty can be expressed as $[\underline{\mu}_A(x), \bar{\mu}_A(x)] = [\tilde{\mu}_A(x) - \Delta x, \tilde{\mu}_A(x) + \Delta x]$. Therefore, an interval-valued membership value is assigned as $[\underline{\mu}_A(x), \bar{\mu}_A(x)]$, $0 \leq \underline{\mu}_A(x), \bar{\mu}_A(x) \leq 1$. In this way, the weather data set in Section 3 is converted into interval representation as shown in Table 2. The results are derived using the algorithm proposed in Section 4, as shown in Figure (11),(12) and (13).

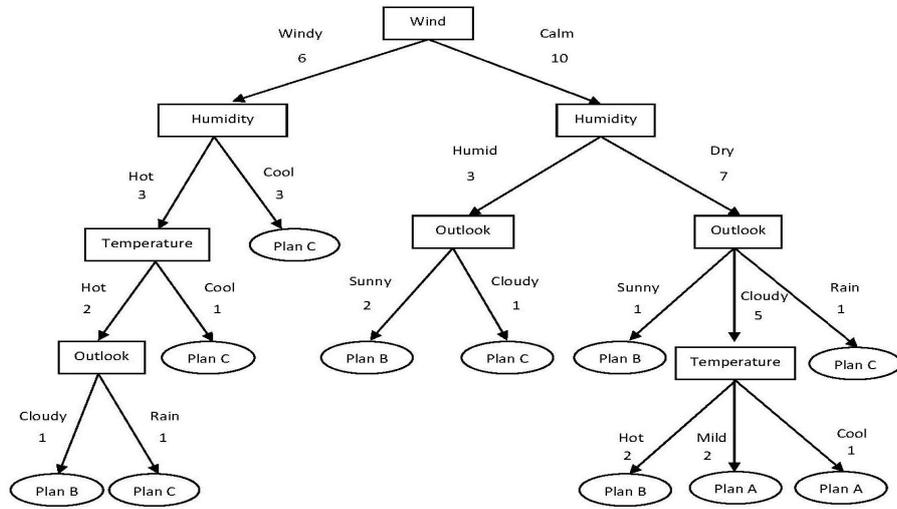


Figure 11: Interval-valued fuzzy decision tree of weather data set with $r = 2$

We used 16 instances of the interval valued data set and tested them with $r = 0.5, 1, 2, 3, 4, 5$ and 6 . Figure 11 to 13 illustrate the decision trees with different r values. The trees in Figures 11 and 13 have 19 nodes and Figure 12 has 21 nodes; the number of nodes in Figure 11 and 13 is less than in Figure 12. Thus, the decision tree in Figures 11 and 13 are better than the tree in Figure 12. The interval-valued fuzzy decision tree with $r = 2$ in Figure 11 and $r = 0.5$ in Figure 13 can be selected for a root node of the tree.

For the given data set in Table 2, we constructed decision trees with $r =$

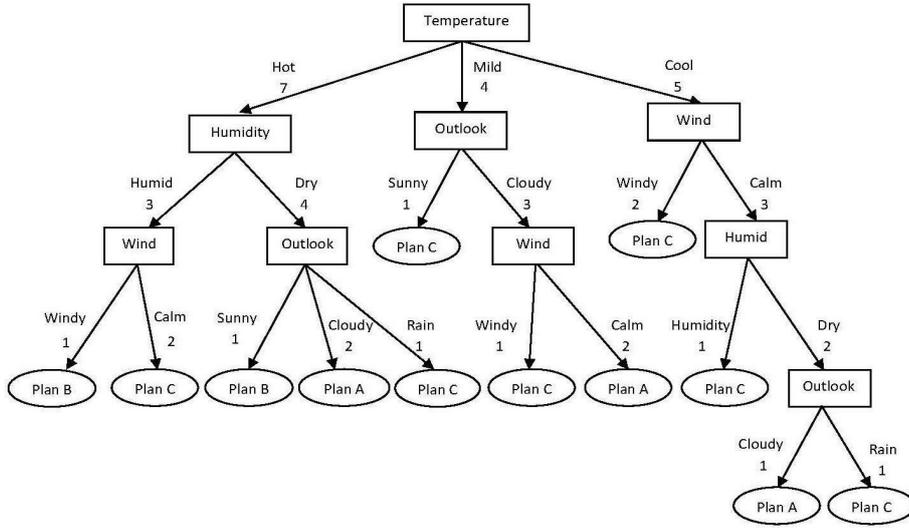


Figure 12: Interval-valued fuzzy decision tree of weather data set with $r = 1$

0.5, 1, 2, 3, 4, 5 and 6. When $r = 0.5$ and 2 we obtain the smallest tree with 19 nodes. when $r = 1$ the trees have 21 nodes. When $r = 3, 4, 5$ and 6 the trees could not be constructed, because there was not a dominant attribute for a root node. For example, if we select $r = 0.5$, we get the results of $\hat{L}^{(k)}$ for each attribute at root node as follows (see Table 3):

$$\hat{L}(\text{Outlook}) = [-0.89, 0.75]$$

$$\hat{L}(\text{Temperature}) = [-1.54, 1.40]$$

$$\hat{L}(\text{Humidity}) = [-0.97, 0.83]$$

$$\hat{L}(\text{Wind}) = [-1.22, 1.01]$$

Table 3: The results of $\hat{L}(x)$

r	Outlook	Temperature	Humidity	Wind
0.5	$[-0.89, 0.75]$	$[-1.54, 1.40]$	$[-0.97, 0.83]$	$[-1.22, 1.01]$
1	$[-0.94, 0.76]$	$[-1.48, 1.34]$	$[-1.04, 0.89]$	$[-1.33, 1.13]$
2	$[-1.15, 0.66]$	$[-1.66, 1.04]$	$[-1.27, 1.04]$	$[-1.62, 1.13]$
3	$[-1.51, 0.74]$	$[-1.96, 1.22]$	$[-1.34, 0.61]$	$[-1.57, 0.82]$
4	$[-1.61, 0.67]$	$[-2.04, 1.10]$	$[-1.39, 0.40]$	$[-1.75, 0.75]$
5	$[-1.61, 0.66]$	$[-2.08, 1.14]$	$[-1.35, 0.38]$	$[-1.71, 0.70]$
6	$[-1.56, 0.62]$	$[-2.06, 1.11]$	$[-1.36, 0.39]$	$[-1.70, 0.70]$

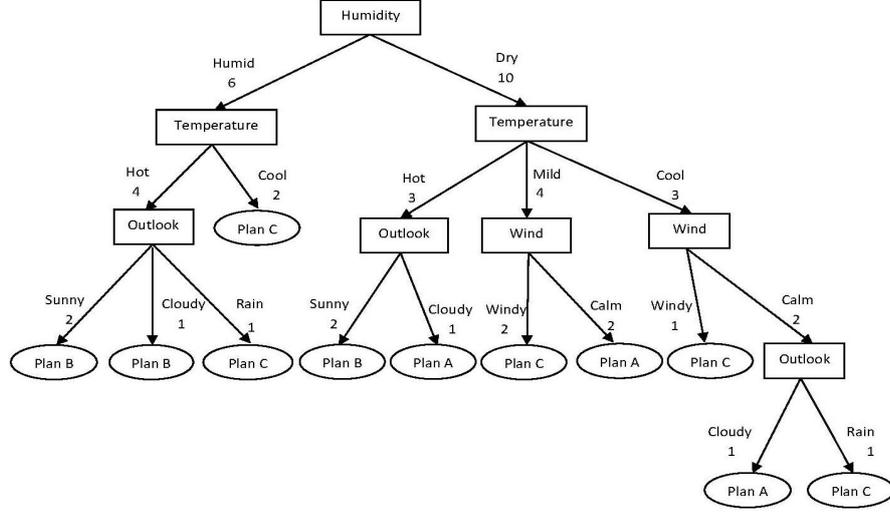


Figure 13: Interval-valued fuzzy decision tree of weather data set with $r = 0.5$

Table 4 illustrates the probability of $x < y$ for each pair of $\hat{L}^{(k)}$ with $r = 0.5, 1, 2, 3, 4, 5$ and 6 . Using the algorithm in section 4, each pair of $\hat{L}^{(k)}$ is compared, e.g. the probability for $\hat{L}(\text{Wind}) \leq \hat{L}(\text{Humidity})$ with $r = 0.5$ is 0.516 . As we can see, it has a confidence or about 51.6% . From Table 4, we can see all values in Humidity column with $r = 0.5$ are greater or equal to 0.5 . It indicates that the probability for any other attribute to have a lower $\hat{L}(x)$ value than Humidity is greater than 0.5 . Therefore, we can draw our conclusion that Humidity should be selected as the root attribute to split the tree into branches.

The probability for $\hat{L}(\text{Wind}) \leq \hat{L}(\text{Temperature})$ is 0.512 . As we can see, it has a confidence or about 51.1% . From Table 4, we can see all values in the Temperature column with $r = 0.5$ are greater or equal to 0.5 . It indicates that the probability for any other attribute to have a lower $\hat{L}(x)$ value than Temperature is greater than 0.5 . Therefore, we can draw our conclusion that Temperature should be selected as the root attribute to split the tree into branches.

However, the probability for Temperature to have a larger $L^{(k)}$ value than Wind is 0.512 , but the probability between Humidity and Wind is 0.516 and the probability of $L^{(k)}$ for Humidity is greater than Wind. Thus, we should select

Humidity.

Similar comparisons can be done for $r = 1$ and $r = 2$. With $r = 3, 4, 5$ and 6 , each column has both values greater than 0.5 and values less than 0.5 . It indicates that we cannot find any dominant attribute to start as a root node.

Table 4: The probability of $P(x < y)$ of $\hat{L}^{(k)}$

$x \setminus y$	r	Outlook	Temperature	Humidity	Wind
Outlook	0.5	–	0.500	0.500	0.484
Temperature		0.279	–	0.500	0.488
Humidity		0.500	0.500	–	0.484
Wind		0.516	0.512	0.516	–
Outlook	1	–	0.507	0.508	0.496
Temperature		0.301	–	0.498	0.489
Humidity		0.492	0.502	–	0.490
Wind		0.504	0.511	0.510	–
Outlook	2	–	0.476	0.485	0.500
Temperature		0.335	–	0.513	0.524
Humidity		0.515	0.487	–	0.511
Wind		0.500	0.476	0.489	–
Outlook	3	–	0.267	0.129	0.188
Temperature		0.116	–	0.502	0.498
Humidity		0.871	0.498	–	0.496
Wind		0.812	0.502	0.504	–
Outlook	4	–	0.244	0.067	0.166
Temperature		0.107	–	0.492	0.490
Humidity		0.933	0.508	–	0.498
Wind		0.834	0.510	0.502	–
Outlook	5	–	0.252	0.063	0.154
Temperature		0.102	–	0.495	0.489
Humidity		0.937	0.505	–	0.492
Wind		0.846	0.511	0.508	–
Outlook	6	–	0.252	0.070	0.162
Temperature		0.098	–	0.497	0.492
Humidity		0.930	0.503	–	0.494
Wind		0.838	0.508	0.506	–

Obviously, with the proposed look-ahead based interval-valued fuzzy decision tree (LAIVFDT), data with uncertain fuzzy membership values could be adopted to construct a fuzzy decision tree. Therefore, a precise fuzzy membership is not a precondition to construct a decision tree anymore. Such relaxation can significantly benefit data mining where precise fuzzy membership values are difficult to get. In this paper, we tested with different r such as $r = 0.5, 1, 2, 3, 4, 5$ and 6 , respectively. The difference of fuzzy decision trees using LAIVFDT and LAFDT are listed as follows:

1. A smaller decision tree is obtained when $r = 0.5$ or $r = 2$ in LAIVFDT and $r = 3$ in LAFDT, where the r value in LAIVFDT is less than the r value in LAFDT method.
2. If the distance r changes then the dominant attribute is changed. Thus, r is significant in constructing the tree.
3. LAIVFDT can construct a decision tree using intervalvalued fuzzy membership values.
4. The decision tree from uncertain membership values is different from that of precise membership values. We cannot take the average value of an interval-valued membership to construct the decision tree.

The data in Table 2 shows the data of interval-valued weather data set for construction of LAIVFDT. Figure 13 illustrates the LAIVFDT of weather data for $r = 0.5$. Figure 14 denotes the LAIVFDT-OPN of weather data using restriction $\alpha = 0.40$. There are 19 and 17 nodes in Figure 13 and 14, respectively. It demonstrates the superiority of the proposed LAIVFDT-OPN method over the traditional LAIVFDT method.

In comparison with LAIVFDT, The LAIVFDT-OPN uses a simple α value which significantly simplifies the task of r value determination, and provides a better value in most cases than a trial value of r .

5.3. Real World Case study

To demonstrate the applicability of the proposed LAIVFDT model, we apply it to a real world machine maintenance example.

5.3.1. The Experiment With DDK Factory Data

The data set is a historical record of daily preventive maintenance injection machine check sheet, and it contains 200 instances gathered by Department of Machine Maintenance, DDK (Thailand) Ltd. For the specific machines in this

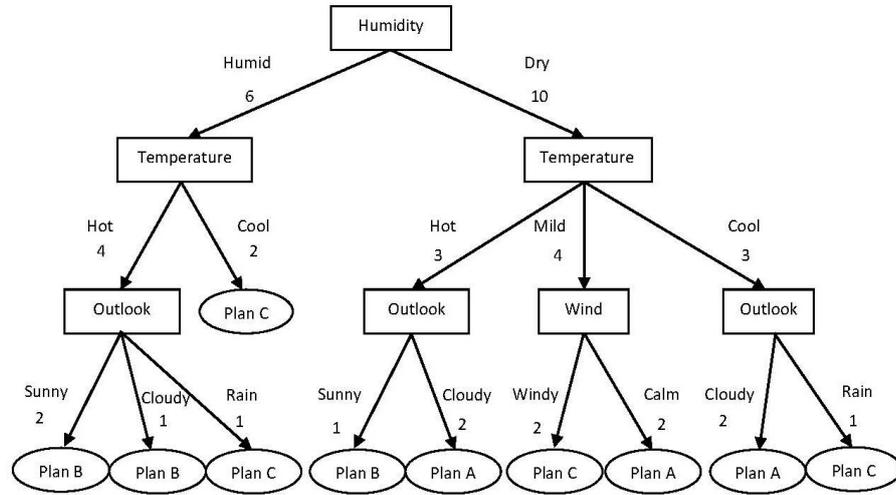


Figure 14: Interval-valued fuzzy decision tree of weather data set with $\alpha = 0.40$

research, eleven important positions must be checked every day. Their symbols and associated physical meaning are described as follows [35].

1. D1 is a position of safety door in non-operation side of an injection machine;
2. D2 is a position of emergency stop button operation (rear) on the molding machine;
3. D3 is a position of water jacket temperature setting;
4. D4 is a position of an electric line and sensor thermocouple line of heater barrel;
5. D5 is a position of purge cover interlock;
6. D6 is a position of melt leakage at a nozzle;
7. D7 is a position of emergency stop button operation (front) on the molding machine;
8. D8 is instead of a position of a safety door in operation side of injection molding machine;
9. D9 is a position of mold mounting bolts and bolts/nuts at fixed platen;

10. D10 is a position mold mounting bolts and bolts/nuts at moved platen;
11. D11 is mold die cleaning, e.g. Tie bar.

Table 5: Interval-valued factory data set

D1			D2			D3			D4			D5			D6		
NOP	PAF	COF															
0.7-0.9	0.4-0.6	0.0-0.1	0.8-1.0	0.5-0.7	0.0-0.2	0.6-0.8	0.4-0.6	0.0-0.1	0.7-0.9	0.4-0.6	0.0-0.2	0.7-0.9	0.4-0.6	0.0-0.1	0.8-1.0	0.5-0.7	0.0-0.2
0.6-0.8	0.4-0.6	0.0-0.1	0.8-1.0	0.5-0.7	0.0-0.1	0.7-0.9	0.3-0.5	0.1-0.3	0.6-0.8	0.3-0.5	0.0-0.1	0.7-0.9	0.4-0.6	0.0-0.2	0.8-1.0	0.5-0.7	0.1-0.3
0.6-0.8	0.3-0.5	0.0-0.2	0.6-0.8	0.4-0.6	0.0-0.1	0.3-0.5	0.6-0.8	0.2-0.4	0.7-0.9	0.4-0.6	0.0-0.1	0.8-1.0	0.5-0.7	0.1-0.3	0.6-0.8	0.4-0.6	0.0-0.1
0.9-1.0	0.5-0.7	0.0-0.2	0.7-0.9	0.3-0.5	0.0-0.1	0.7-0.9	0.4-0.6	0.0-0.2	0.7-0.9	0.4-0.6	0.0-0.1	0.8-1.0	0.4-0.6	0.0-0.2	0.1-0.3	0.7-0.9	0.3-0.5
0.7-0.9	0.5-0.7	0.0-0.1	0.0-0.2	0.8-1.0	0.5-0.7	0.6-0.8	0.3-0.5	0.0-0.1	0.6-0.8	0.3-0.5	0.0-0.1	0.0-0.2	0.8-1.0	0.5-0.7	0.7-0.9	0.4-0.6	0.0-0.1
0.8-1.0	0.5-0.7	0.0-0.1	0.7-0.9	0.4-0.6	0.1-0.3	0.7-0.9	0.4-0.6	0.0-0.1	0.7-0.9	0.4-0.6	0.1-0.3	0.6-0.8	0.4-0.6	0.0-0.1	0.8-1.0	0.5-0.7	0.0-0.1
0.6-0.8	0.4-0.6	0.0-0.1	0.7-0.9	0.5-0.7	0.0-0.1	0.7-0.9	0.4-0.6	0.0-0.1	0.6-0.8	0.4-0.6	0.0-0.1	0.6-0.8	0.3-0.5	0.0-0.1	0.6-0.8	0.3-0.5	0.0-0.2
0.3-0.5	0.8-1.0	0.2-0.4	0.6-0.8	0.3-0.5	0.0-0.1	0.8-1.0	0.5-0.7	0.0-0.2	0.0-0.2	0.7-0.9	0.3-0.5	0.7-0.9	0.4-0.6	0.0-0.1	0.9-1.0	0.5-0.7	0.0-0.2
0.6-0.8	0.3-0.5	0.0-0.1	0.8-1.0	0.5-0.7	0.1-0.3	0.8-1.0	0.5-0.7	0.1-0.3	0.7-0.9	0.4-0.6	0.0-0.1	0.7-0.9	0.4-0.6	0.0-0.2	0.8-1.0	0.5-0.7	0.0-0.1
0.8-1.0	0.4-0.6	0.1-0.3	0.7-0.9	0.4-0.6	0.0-0.1	0.7-0.9	0.4-0.6	0.0-0.1	0.8-1.0	0.5-0.7	0.0-0.1	0.8-1.0	0.5-0.7	0.0-0.1	0.8-1.0	0.4-0.6	0.0-0.2
0.7-0.9	0.4-0.6	0.0-0.2	0.8-1.0	0.4-0.6	0.0-0.1	0.6-0.8	0.3-0.5	0.0-0.1	0.7-0.9	0.4-0.6	0.0-0.1	0.8-1.0	0.4-0.6	0.0-0.1	0.7-0.9	0.4-0.6	0.0-0.1
0.7-0.9	0.3-0.5	0.0-0.1	0.8-1.0	0.5-0.7	0.0-0.1	0.6-0.8	0.4-0.6	0.0-0.1	0.6-0.8	0.4-0.6	0.0-0.1	0.7-0.9	0.4-0.6	0.0-0.1	0.0-0.2	0.8-1.0	0.3-0.5
0.6-0.8	0.4-0.6	0.0-0.1	0.6-0.8	0.3-0.5	0.0-0.1	0.9-1.0	0.5-0.7	0.0-0.1	0.6-0.8	0.3-0.5	0.0-0.1	0.7-0.9	0.4-0.6	0.0-0.1	0.6-0.8	0.4-0.6	0.0-0.1
0.8-1.0	0.5-0.7	0.1-0.3	0.9-1.0	0.5-0.7	0.0-0.2	0.8-1.0	0.5-0.7	0.0-0.2	0.9-1.0	0.5-0.7	0.1-0.3	0.6-0.8	0.3-0.5	0.0-0.1	0.6-0.8	0.3-0.5	0.0-0.1
...
0.7-0.9	0.3-0.5	0.1-0.3	0.7-0.9	0.4-0.6	0.0-0.1	0.6-0.8	0.4-0.6	0.0-0.1	0.6-0.8	0.4-0.6	0.0-0.1	0.6-0.8	0.3-0.5	0.0-0.1	0.8-1.0	0.5-0.7	0.0-0.1

Table 6: Interval-valued factory data set (Continue)

D7			D8			D9			D10			D11			Status	
NOP	PAF	COF	ACT	UAC												
0.7-0.9	0.4-0.6	0.0-0.1	0.7-0.9	0.5-0.7	0.0-0.2	0.6-0.8	0.3-0.5	0.0-0.1	0.8-1.0	0.4-0.6	0.0-0.2	0.7-0.9	0.3-0.5	0.0-0.2	0.7-0.9	0.1-0.3
0.8-1.0	0.5-0.7	0.0-0.1	0.8-1.0	0.5-0.7	0.0-0.1	0.7-0.9	0.4-0.6	0.0-0.1	0.8-1.0	0.5-0.7	0.1-0.3	0.7-0.9	0.4-0.6	0.0-0.1	0.8-1.0	0.0-0.2
0.6-0.8	0.4-0.6	0.0-0.1	0.3-0.5	0.8-1.0	0.0-0.2	0.1-0.3	0.7-0.9	0.3-0.5	0.7-0.9	0.3-0.5	0.0-0.1	0.6-0.8	0.3-0.5	0.0-0.1	0.2-0.4	0.7-0.9
0.9-1.0	0.5-0.7	0.0-0.1	0.6-0.8	0.4-0.6	0.0-0.1	0.6-0.8	0.3-0.5	0.0-0.1	0.7-0.9	0.3-0.5	0.0-0.1	0.6-0.8	0.4-0.6	0.0-0.1	0.7-0.9	0.2-0.4
0.8-1.0	0.5-0.7	0.0-0.2	0.7-0.9	0.4-0.6	0.0-0.1	0.8-1.0	0.5-0.7	0.0-0.2	0.6-0.8	0.4-0.6	0.0-0.1	0.8-1.0	0.4-0.6	0.0-0.2	0.6-0.8	0.1-0.3
0.8-1.0	0.4-0.6	0.0-0.1	0.7-0.9	0.3-0.5	0.0-0.1	0.7-0.9	0.4-0.6	0.0-0.1	0.8-1.0	0.4-0.6	0.0-0.1	0.7-0.9	0.4-0.6	0.0-0.2	0.7-0.9	0.3-0.5
0.7-0.9	0.4-0.6	0.0-0.1	0.7-0.9	0.4-0.6	0.0-0.1	0.8-1.0	0.4-0.6	0.0-0.1	0.8-1.0	0.4-0.6	0.0-0.2	0.6-0.8	0.3-0.5	0.0-0.1	0.6-0.8	0.2-0.4
0.7-0.9	0.3-0.5	0.0-0.1	0.8-1.0	0.4-0.6	0.0-0.2	0.8-1.0	0.5-0.7	0.0-0.2	0.7-0.9	0.3-0.5	0.0-0.1	0.8-1.0	0.5-0.7	0.0-0.1	0.0-0.2	0.8-1.0
0.8-1.0	0.5-0.7	0.0-0.2	0.8-1.0	0.5-0.7	0.1-0.3	0.7-0.9	0.4-0.6	0.0-0.1	0.7-0.9	0.4-0.6	0.0-0.1	0.7-0.9	0.4-0.6	0.0-0.1	0.8-1.0	0.1-0.3
0.8-1.0	0.4-0.6	0.0-0.1	0.7-0.9	0.3-0.5	0.0-0.1	0.7-0.9	0.3-0.5	0.0-0.1	0.6-0.8	0.3-0.5	0.0-0.1	0.6-0.8	0.3-0.5	0.0-0.1	0.7-0.9	0.1-0.3
0.2-0.4	0.6-0.8	0.4-0.6	0.7-0.9	0.4-0.6	0.0-0.1	0.1-0.3	0.4-0.6	0.7-0.9	0.6-0.8	0.3-0.5	0.0-0.2	0.1-0.3	0.7-0.9	0.4-0.6	0.1-0.3	0.7-0.9
0.7-0.9	0.4-0.6	0.0-0.1	0.6-0.8	0.3-0.5	0.0-0.1	0.8-1.0	0.4-0.6	0.0-0.1	0.8-1.0	0.5-0.7	0.0-0.1	0.8-1.0	0.4-0.6	0.0-0.1	0.8-1.0	0.0-0.2
0.8-1.0	0.5-0.7	0.0-0.1	0.6-0.8	0.4-0.6	0.0-0.1	0.8-1.0	0.5-0.7	0.0-0.1	0.8-1.0	0.4-0.6	0.0-0.1	0.7-0.9	0.4-0.6	0.0-0.2	0.6-0.8	0.3-0.5
0.7-0.9	0.4-0.6	0.0-0.2	0.8-1.0	0.5-0.7	0.0-0.1	0.6-0.8	0.3-0.5	0.1-0.3	0.6-0.8	0.3-0.5	0.0-0.1	0.6-0.8	0.3-0.5	0.0-0.1	0.7-0.9	0.2-0.4
...
0.8-1.0	0.5-0.7	0.0-0.1	0.7-0.9	0.3-0.5	0.1-0.3	0.8-1.0	0.5-0.7	0.0-0.1	0.8-1.0	0.4-0.6	0.0-0.2	0.8-1.0	0.5-0.7	0.0-0.1	0.6-0.8	0.1-0.3

All attributes are recorded as normal operation (NOP), partial failure (PAF) or completely failure (COF) and the classification attribute (status attribute) has its value as acceptable (ACT) or unacceptable (UAC). With the assistance of the technicians who recorded these data, we have converted these data into

normalised interval values as shown in Table 5 and 6.

5.3.2. Comparison of LAIVFDT-OPN, LAIVFDT and F-ID3

Using the proposed LAIVFDT-OPN method with $\alpha = 0.40$ and 100 instances randomly selected from the data set, a decision tree can be constructed from the DDK factory data set. According to the algorithm discussed in Section 4, the key is to choose the right attribute as the start node in root and each branch level. To identify the suitable attribute, the $L^{(k)}$ values have to be calculated firstly. Table 7 and 8 illustrate the $L^{(k)}$ values of each attribute under different levels of each branch. For example, the highest value of $L^{(k)}$ values for level 1 is $[-0.55, 1.02]$. Thus, the root node is “D9”. For level 2, the first branch is “D9=NOP” where the highest value $[-0.51, 1.04]$ is for “D10” so that “D10” is selected for the node of that branch. The next branch is “D9=PAF” and its highest value $[-1.61, 2.29]$ is for “D4” so “D4” is selected. The final branch of “D9” node is “D9=COF” where its $L^{(k)}$ values for all attributes are equal and there is only one instance in this branch. Thus, we come to a leaf node where status=UAC. In this way, a decision tree is established as shown in Figure 15. In this tree, the number of nodes is 40. Similarly, we can construct decision trees using LAFDT and F-ID3 as well. Together with our results from the weather data set in previous section, the results from LAIVFDT-OPN, LAIVFDT, LAFDT and F-ID3 is compared in Table 9. Obviously, for the weather data set, the LAIVFDT-OPN method is better than LAFDT method, LAIVFDT method and F-ID3 method due to the smallest amount of nodes and levels. There are 17 nodes in the resulted tree from LAIVFDT-OPN, but 18, 19 nodes and 20 nodes in the trees from LAFDT method, LAIVFDT method and F-ID3 respectively. For DDK factory data set, LAIVFDT-OPN method is also better than other methods due to the smallest amount of nodes and levels. There are 40 nodes for LAFDT and LAIVFDT methods and 52 nodes for F-ID3 method. There are 6 levels for the tree constructed using LAIVFDT-OPN method, 10 levels for the tree with LAFDT method and 11 levels for the tree from F-ID3

method.

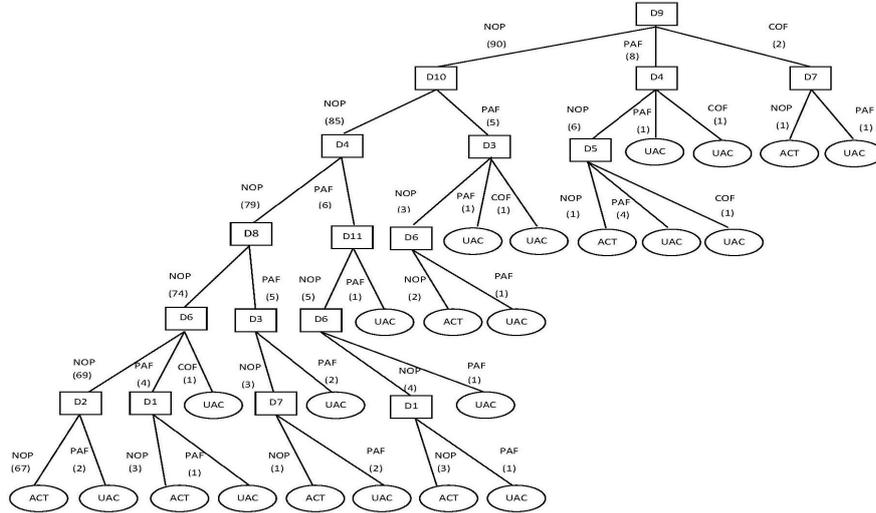


Figure 15: Interval-valued fuzzy decision tree of DDK factory data set with $\alpha = 0.40$

In Figure 15, the fuzzy decision tree looks unbalanced. The data set in Table 5 and 6 are not well balanced. The data of daily preventive maintenance injection machine check sheet contains more Normal Operation status, and only a small fraction of data for Partial Failure status and Completely Failure status. From Figure 15, the best node for a root node is “D9” node. The decision tree in 15 is obtained by choosing “D9” as root node. To check the tree shape of trees rooted from other attributes, a different attribute can be selected as the root node. For example, we take the second best node “D4” as the root node, there are 3 possible branches:

1. D4=NOP has 93 nodes.
2. D4=PAF has 7 nodes.

Table 7: $L^{(k)}$ values of each factor (with $\alpha = 0.40$)

Factor	$L^{(k)}$ values									
	Level 1	Level 2			Level 3			Level 4		
	-	D9=NOP	D9=PAF	D9=COF	D9=NOP D10=NOP	D9=NOP D10=PAF	D9=PAF D4=NOP	D9=NOP D10=NOP D4=NOP	D9=NOP D10=NOP D4=PAF	D9=NOP D10=PAF D3=NOP
D1	[-0.64, 0.98]	[-0.55, 1.01]	[-1.70, 2.33]	[-2.66, 2.66]*	[-0.51, 1.03]	[-0.62, 1.53]	[-1.73, 2.48]*	[-0.48, 1.01]	[-1.19, 1.25]	[-0.60, 1.96]
D2	[-0.63, 0.98]	[-0.54, 1.00]	[-1.70, 2.33]	[-2.66, 2.66]*	[-0.52, 1.03]	[-0.44, 1.30]	[-1.63, 1.35]	[-0.48, 1.02]	[-1.15, 1.23]	[-0.30, 1.30]
D3	[-0.62, 0.98]	[-0.55, 1.01]	[-1.63, 2.20]	[-2.66, 2.66]*	[-0.52, 1.03]	[-0.75, 2.00]*	[-1.53, 2.28]	[-0.46, 1.05]	[-1.15, 1.23]	-
D4	[-0.62, 0.99]	[-0.52, 1.01]	[-1.61, 2.29]*	[-2.66, 2.66]*	[-0.47, 1.04]*	[-0.62, 1.53]	-	-	-	[-0.60, 1.96]
D5	[-0.61, 0.99]	[-0.52, 1.02]	[-1.49, 2.12]	[-2.66, 2.66]*	[-0.52, 1.03]	[-0.80, 1.95]	[-1.73, 2.48]*	[-0.48, 1.02]	[-1.15, 1.23]	[-0.86, 2.46]*
D6	[-0.63, 0.99]	[-0.51, 0.99]	[-1.70, 2.33]	[-2.66, 2.66]*	[-1.67, 1.42]	[-0.77, 1.73]	[-1.73, 2.48]*	[-0.50, 1.05]	[-1.22, 1.33]	[-0.86, 2.46]*
D7	[-0.61, 1.01]	[-0.55, 1.02]	[-1.93, 2.54]	[-2.66, 2.66]*	[-0.52, 1.02]	[-0.87, 1.99]	[-1.77, 2.50]	[-0.62, 0.99]	[-1.15, 1.23]	[-0.86, 2.46]*
D8	[-0.60, 0.99]	[-0.51, 1.03]	[-1.60, 2.10]	[-2.66, 2.66]*	[-0.50, 1.04]	[-0.87, 1.99]	[-1.41, 1.94]	[-0.43, 1.04]*	[-1.15, 1.23]	[-0.86, 2.46]*
D9	[-0.55, 1.02]*	-	-	-	-	-	-	-	-	-
D10	[-0.59, 1.02]	[-0.51, 1.04]*	[-1.47, 2.05]	[-2.66, 2.66]*	-	-	[-1.53, 2.19]	-	-	-
D11	[-0.61, 1.03]	[-0.53, 1.03]	[-1.93, 2.54]	[-2.66, 2.66]*	[-0.51, 1.04]	[-0.87, 1.99]	[-1.77, 2.50]	[-0.49, 1.02]	[-0.77, 0.98]*	[-0.86, 2.46]*

Note: * The highest $L^{(k)}$ value is selected. If equal $L^{(k)}$ value, one $L^{(k)}$ value of them is selected.

3. D4=COF has 0 node.

Obviously, this is an unbalanced tree as well. In a similar way, we can show the trees rooted from other attributes are also unbalanced trees.

5.3.3. Accuracy of prediction of LAIVFDT-OPN algorithm

As aforementioned, 100 samples randomly selected from the data set are taken to establish the decision trees, and the 100 samples are employed to test the performance of the established trees. Three performance criteria are measured: accuracy, sensitivity and specificity [36, 37]. The performance evaluation are defined by

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}. \quad (14)$$

$$Sensitivity = \frac{TP}{TP + FN}. \quad (15)$$

$$Specificity = \frac{TN}{TN + FP}. \quad (16)$$

Table 8: $L^{(k)}$ values of each factor (with $\alpha = 0.40$) (Continue)

Factor	$L^{(k)}$ values							
	Level 5			Level 6				
	D9=NOP	D9=NOP	D9=NOP	D9=NOP	D9=NOP	D9=NOP	D9=NOP	D9=NOP
	D10=NOP	D10=NOP	D10=PAF	D10=NOP	D10=NOP	D10=NOP	D10=NOP	D10=NOP
D4=NOP	D4=PAF	D3=NOP	D4=NOP	D4=NOP	D4=NOP	D4=PAF	D4=PAF	
D8=NOP	D8=PAF	D11=NOP	D8=NOP	D8=NOP	D8=PAF	D11=NOP	D11=NOP	
	-	-	-	D6=NOP	D6=PAF	D3=NOP	D6=NOP	
D1	[-0.43, 1.02]	[-0.87, 0.99]	[-0.79, 0.97]	[-0.45, 1.07]*	[-0.48, 1.56]*	[-1.19, 1.28]*	[-0.67, 1.07]*	
D2	[-0.43, 1.02]	[-0.86, 0.98]	[-0.78, 0.98]	[-0.45, 1.07]*	[-0.87, 1.51]	[-0.86, 0.90]	[-0.82, 0.93]	
D3	[-0.44, 1.06]	[-0.98, 1.25]*	-	[-0.47, 1.07]	[-0.87, 1.51]	-	[-0.82, 0.93]	
D4	-	-	[-0.78, 0.98]	-	-	-	-	
D5	[-0.44, 1.03]	[-0.87, 0.99]	[-0.78, 0.98]	[-0.46, 1.08]	[-0.87, 1.51]	[-1.19, 1.28]*	[-0.82, 0.93]	
D6	[-0.45, 1.07]*	[-0.87, 0.99]	[-0.69, 1.06]*	-	-	[-1.19, 1.28]*	-	
D7	[-0.44, 1.02]	[-0.87, 0.99]	[-0.78, 0.98]	[-0.46, 1.07]	[-0.87, 1.51]	[-1.19, 1.28]*	[-0.82, 0.93]	
D8	-	-	[-0.78, 0.98]	-	-	-	[-0.82, 0.93]	
D9	-	-	-	-	-	-	-	
D10	-	-	-	-	-	-	-	
D11	[-0.44, 1.03]	[-0.87, 0.99]	-	[-0.46, 1.08]	[-0.87, 1.51]	[-1.19, 1.28]*	-	

Table 9: Comparison of F-ID3, LAFDT, LAIVFDT and LAIVFDT-OPN methods

Data set	Method	Data type	Instances	Restriction	Nodes	Levels	Root node
Weather	F-ID3	SV	16	-	20	3	Outlook
	LAFDT	SV	16	$r = 3.0$	18	4	Wind
	LAIVFDT	IV	16	$r = 6.0$	19	4	Humidity
	LAIVFDT-OPN	IV	16	$\alpha = 0.40$	17	3	Humidity
DDK factory	F-ID3	SV	100	-	52	11	D9
	LAFDT	SV	100	$r = 6.0$	40	10	D3
	LAIVFDT-OPN	IV	100	$\alpha = 0.40$	40	6	D9

Note: SV and IV stand for single value and interval value, respectively.

where TP and TN are the number of true positives and true negatives, respectively. FP and FN are the number of false positives and false negatives, respectively [36, 37].

In Table 11, the test results of LAIVFDT-OPN algorithm using Equation (14) - (16) are displayed to compare the correct and incorrect classification from each class. The number of acceptable status is higher than half of the testing samples at about 58 out of 100 instances. Table 11 illustrates the sensitivity and specificity by proportion of acceptable and unacceptable machine status.

The accuracy of prediction by LAIVFDT-OPN method is approximately 79.0%. Sensitivity and specificity of “ACT” is 0.879 and 0.618 respectively and “UAC” has inverse values.

Table 10: The outcome of LAIVFDT-OPN experiment

Model	Test outcome		Status of machine		Predictive value
LAIVFDT-OPN	Test outcome	Positive	58	21	79
		Negative	13	8	21
	Total		71	29	100

Table 11: The class-wise statistics for performance evaluation of LAIVFDT-OPN method

Classes	LAIVFDT-OPN model	
	Status	
	Acceptable (ACT)	Unacceptable (UAC)
Number of instances	76	24
True positives	58	21
True negatives	21	58
False positives	13	8
False negatives	8	13
Sensitivity	0.879	0.618
Specificity	0.618	0.879
Accuracy	79.0%	79.0%

Table 12: Execution Times (in Seconds) for LAIVFDT-OPN model

Data set	Execution time(s) of LAIVFDT-OPN model	
	Training data	Testing data
DDK factory (100 instances)	199.5	197.4

The execution time of LAIVFDT-OPN model associates with the size of the generated rule base. Table 12 compares the execution times of both the training data and the testing data by LAIVFDT-OPN model; the execution time in this table is not noticeable by the user on small data set.

Table 13 illustrates the outcome of F-ID3, LAFDT and LAIVFDT-OPN methods using the remaining 100 samples/instances for testing the models. The

Table 13: The outcome of F-ID3, LAFDT and LAIVFDT-OPN experiment

Model			Status of machine		Total of predictive value
			Acceptable(ACT)	Unacceptable(UAC)	
F-ID3	Test outcome	Positive	52	20	72
		Negative	16	12	28
	Total		68	32	100
LAFDT	Test outcome	Positive	56	20	76
		Negative	15	9	24
	Total		71	29	100
LAIVFDT-OPN	Test outcome	Positive	58	21	79
		Negative	13	8	21
	Total		71	29	100

total percentages of predictive value for the positive test outcome are at about 72%, 76% and 79% for F-ID3, LAFDT and LAIVFDT-OPN, respectively. And the negative test outcomes are at about 28%, 24% and 21% for F-ID3, LAFDT and LAIVFDT-OPN, respectively. 71% of the acceptable statuses are LAFDT and LAIVFDT-OPN methods and F-ID3 method has 68%. 29% of the unacceptable statuses are LAFDT and LAIVFDT-OPN methods and F-ID3 method has 32%.

Table 14: The class-wise statistics for performance evaluation of F-ID3, LAFDT and LAIVFDT-OPN methods

Classes	Model					
	F-ID3		LAFDT		LAIVFDT-OPN	
Number of leaves	27		21		23	
Status of machine	ACT	UAC	ACT	UAC	ACT	UAC
Number of instances	76	24	76	24	76	24
True positives	52	20	56	20	58	21
True negatives	20	52	20	56	21	58
False positives	16	12	15	9	13	8
False negatives	12	16	9	15	8	13
Sensitivity	0.813	0.556	0.862	0.571	0.879	0.618
Specificity	0.556	0.813	0.571	0.862	0.618	0.879
Accuracy	72.0%	72.0%	76.0%	76.0%	79.0%	79.0%

Table 14 illustrates the performance evaluation of F-ID3, LAFDT and LAIVFDT-OPN methods. In table 14, LAIVFDT-OPN method has the highest amount of instances of acceptable status in the true positives at 58 instances and the

F-ID3 has the lowest at 52 samples. F-ID3 method has the highest amount of instances of acceptable status in false positive at 16 instances and LAFDT and LAIVFDT-OPN have 15 and 13 instances, respectively. F-ID3 and LAFDT method have the lowest amount of instances of unacceptable status in true negative at 20 instances and LAIVFDT-OPN method has 21 instances. The lowest amount of instances of unacceptable status in false negative is LAIVFDT-OPN at 8 instances and the F-ID3 has the highest at 12 instances. LAIVFDT-OPN method has the highest percentage of accuracy at about 79.0%. The second highest percentage is LAFDT method at about 76.0% and the lowest percentage is F-ID3 method at about 72.0%. The most sensitivity to the least sensitivity are LAIVFDT-OPN, LAIVFDT and F-ID3, respectively.

Comparing the results in Table 14, it is clear that LAIVFDT-OPN outperforms both LAIVFDT and F-ID3, and LAIVFDT gives better results than F-ID3. As our analysis in previous sections, the full employment of interval information in LAIVFDT-OPN and LAIVFDT enables them to produce better trees in terms of size and suitability to wider uncertainties. The LAIVFDT-OPN directly applies a α value which has been known to be a near optimised value for α , but LAIVFDT has to make many trails to find a suitable r value. Therefore, LAIVFDT-OPN has good chance to outperform LAIVFDT in most cases.

6. Conclusions

In this paper, we proposed LAIVFDT-OPN method to apply interval-valued fuzzy sets to construct an interval-valued fuzzy decision tree. In the proposed model, Hamming distance between two interval-valued fuzzy sets is applied to measure the distance between the two instances. A probability model is employed to compare intervals to determine the classifiability of each attribute. A systematic algorithm was established to construct a decision tree from data with uncertain membership values. Our examples demonstrate that the proposed

method does construct an acceptable decision tree when interval-valued fuzzy membership values are involved in the data set. To determine the ideal distance restriction in LAIVFDT method, the significance of different distance restriction values was investigated and a new parameter α for restricting neighbourhood instances in LAIVFDT induction with optimal distance for fuzzy data was proposed and an optimal α value was identified for the experiment data sets. Our preliminary experiment results show that the ideal distance restriction changes with data set but α is much more stable. Our experiments are still limited in terms of the number of data sets and their scope, but it does demonstrate the potential of the proposed model.

Acknowledgment

The authors would like to thank Assistant Professor Dr. Panarat Lim (Dhurakij Pundit University), Associate Professor Suvimol Manjing (Rajmangala University of Technology Thanyaburi) and Assistant Professor Busabun Learungraung (Rajmangala University of Technology Srivichai). In particular, special thanks are also extended to Mr. Thammarat Watthanabutr (Human Resource Assistant Manager), Mr. Wallop Ruangngam (Plant Engineering Supervisor), Mr. Sanit Fuangnakorn (Molding Manager), Mr. Sophon Yosthasaen (Department of Molding Machine Maintenance), Mr. Prasong Soksom (Department of Molding Production) and Mr Surapong Sripaiboon (Department of Mold Die Maintenance) and other staff at DDK (Thailand) Ltd for their Help.

References

- [1] K.M. Lee, K.M. Lee, J.H. Lee and H.L. Kwang, "A fuzzy decision tree induction method for fuzzy data", *IEEE International Fuzzy Systems Conference Proceedings*, pp. 16-21, 1999.

- [2] X. Wang, L. Dong and J. Yan, “Maximum ambiguity based sample selection in fuzzy decision tree induction”, *IEEE Transactions on Knowledge and Data Engineering*, Vol. 24, pp.1491-1505, 2012.
- [3] X. Wang and C. Dong, “Improving generalization of fuzzy if-then rules by maximizing fuzzy entropy”, *IEEE Transactions on Fuzzy Systems*, Vol. 17, pp.556-567, 2009.
- [4] X. Wang, J. Zhai and S. Lun “Induction of multiple fuzzy decision trees based on rough set technique”, *Information Sciences*, Vol. 178, pp.3188-3202, 2008.
- [5] P. Grzeorzewski, “Distance between intuitionistic fuzzy sets and/or interval-valued fuzzy set based in the Hausdorff metric”, *Fuzzy Sets and Systems*, Vol. 148, pp.319-328, 2004.
- [6] Y. Yang and F. Chiclana, “Intuitionistic fuzzy sets: spherical representation and distance”, *International Journal of Intelligent Systems*, Vol. 24, pp.399-420, 2009.
- [7] H. Zhang, W. Zhang and C. Mei, “Entropy of interval-valued fuzzy sets based on distance and its relationship with similarity measure”, *Knowledge-Based Systems*, Vol. 22, pp.449-454, 2009.
- [8] S.Ferson, J. Hajagos, D. Berleant, J. Zhang, W. T. Tucker, L. Ginzburg and W. Oberkamf, “Dependence in Dempster-Shafer theory and probability bounds analysis”, *Applied Biomathematics*, U.S.A., pp.1-138, 2004.
- [9] M. Schneider, A. Kandel, G. Langholz and G. Chew, “*Fuzzy Expert System Tools*”, John Wiley & Sons Ltd., England, 1996.
- [10] Y. Yuan and M. J. Shaw, “Induction of fuzzy decision tree”, *Fuzzy Sets and Systems*, Vol. 69, pp.125-139, 1995.
- [11] M. Dong and R. Kothari, “Look-ahead based fuzzy decision tree induction”, *IEEE Transactions on Fuzzy Systems*, Vol 9, No 3, pp. 461-468, 2001.

- [12] M. Dong, R. Kothari, and M. Visscher and S. B. Hoath, “Evaluating skin using a new decision tree induction algorithm”, *Proceedings of the 2001 International Joint Conference on Neural Networks*, Vol. 4, pp. 2456-2460, 2001.
- [13] X. Wang, B. Chen, G. Qian and F. Ye, “On the optimization fuzzy decision tree”, *Fuzzy Sets and Systems*, Vol. 112, pp. 117-125, 2000.
- [14] K. T. Atanassov and G. Gargov, “Interval valued intuitionistic fuzzy sets”, *Fuzzy Sets and Systems*, Vol. 31, No. 3, pp.343-349, 1989.
- [15] C. Y. C. ChiangLin, “Comparison between Crisp and Fuzzy Stock Screening Models”, [Online]. Available: http://www.atlantispress.com/php/download_paper.php?id=33.
- [16] J. Zeidler and M. Schlosser, “Continuous-valued attributes in fuzzy decision trees”, *Proceeding of the 6th International Conference on Information Proceeding and Management of Uncertainty in Knowledge-Based Systems*, pp. 395-400, 1996.
- [17] J. M. Mendel, H. Wu, V. Kreinovich and G. Xiang, “Fast computation of centroids for constant-width interval-valued fuzzy sets”, *Fuzzy Information Processing Society, NAFIPS 2006, Annual Meeting of the North American*, pp. 621-626, 2006.
- [18] J. M. Mendel and R. I. John and F. Liu, “Interval type-2 fuzzy logic systems made simple”, *IEEE Transactions on Fuzzy Systems*, Vol.14,No. 6, pp. 808-821, 2006.
- [19] J.M. Mendel and R.I. John, “Type-2 fuzzy sets made simple”, *IEEE Transactions on Fuzzy Systems*, Vol. 10,No. 2, pp. 117-127, 2002.
- [20] D. Wu, “A vector similarity measure for type-1 fuzzy sets”, *Lecture Note In Artificial Intelligence, Proceeding of the 12th International Fuzzy Systems Association World Congress on Foundations of Fuzzy Logic and Soft Computing*, Vol. 4529, pp.575-583, 2007.

- [21] K. McKarty and M. Manic, "Contextual fuzzy type-2 hierarchical for decision trees (CoFuH-DT) - An accelerated data mining technique", *Conference on Human Systems Interactions*, pp. 699-704, 2008.
- [22] M. Bohanec and B. Zupan, "Car Evaluation Database: UCI Repository of Machine Learning Database, Department of Information and Computer Science. University of California" Irvine, Ca, [Online]. Available: http://www.ics.uci.edu/mlearn/MLR_Repository.html, 1997.
- [23] D. M. Huang, J. Yang, X. Z. Wang and M. H. Ha, "Induction of decision tree with fuzzy number-valued attribute", *Proceedings of the Second International Conference on Machine Learning and Cybernetics, Xi'an*, pp. 1446-1450, 2003.
- [24] I. Jenhani, N. B. Amor and Z. Elouedi, "Decision trees as possibilistic classifiers", *International Journal of Approximate Reasoning*, Vol. 48, pp. 784-807, 2008.
- [25] G. J. Klir, H. U. St. Clair and B. Yuan, "*Fuzzy Set Theory: Foundation and Application*", Prentice-Hall Inc., NJ, USA, 1997.
- [26] B. Kosko, "Fuzzy entropy and conditioning", *Information Sciences: an International Journal*, Vol. 40, pp. 165-174, 1986.
- [27] R. Kruse, J. Gebhardt and F. Klawonn, "*Foundation of Fuzzy Systems*", John Wiley & Sons Ltd., West Sussex, UK, 1994.
- [28] J. Mingers, "An empirical comparison of selection measures for decision-tree induction", *Machine Learning*, Vol. 3, pp. 319-342, 1989.
- [29] J. R. Quinlan, "Induction of Decision Trees", *Machine learning*, Vol. 1, pp. 81-106, 1986.
- [30] M. Umano, H. Okamoto, I. Hatono and H. Tamura, "Fuzzy decision trees by fuzzy ID3 algorithm and its application to diagnosis systems", *IEEE*

World Congress on Computational Intelligence, Proceedings of the Third IEEE Conference on Fuzzy Systems, Vol. 3, pp. 2213-2118, 1994.

- [31] J. Zeidler, M. Schlosser, A. Ittner and C. Posthoff, “Fuzzy decision tree and numerical attributes”, *In IEEE Fuzzy Sets Systems*, pp. 985-990, 1996.
- [32] M. H. Zhao, Y. Z. Chen, D. R. Liu and J. Li, “The application of optimized fuzzy decision trees in business intelligence”, *Proceeding of the Fifth International on Machine Learning and Cybernetics*, Dalian, pp. 2212-2217, 2006.
- [33] Y. Lertworaprachaya, Y. Yang and R. John, “Interval-valued fuzzy decision trees”, *WCCI 2010 IEEE World Congress on Computational Intelligence, CCIB*, Barcelona, Spain, pp. 2070-2076, 2010.
- [34] Y. Lertworaprachaya and Y. Yang, “Optimal perimeter of neighbourhood in fuzzy decision tree”, *IADIS European Conference Data Mining 2010*, Freiburg, Germany, pp. 117-121, 2010.
- [35] Sumitomo Heavy Industries, Ltd, “Sumitomo Injection Molding Machine SE-DU Series Instruction Manual”, *Sumitomo Heavy Industries Ltd*, Sumitomo Injection Molding Machine Manual, chapter 6, pp. 6-1 - 6-25, 2006.
- [36] U. K. Muhammad, P. C. Jong, S. Hyunjung and K. Minko, “Prediction Breast Cancer Survivability Using Fuzzy Decision Trees for Personalized Healthcare”, *Annual International IEEE EMBS Conference*, IEEE, pp. 5148-5151, 2008.
- [37] P. B. Andrew, “The Use of the Area Under the ROC Curve in the Evaluation of Machine Learning Algorithm”, *Pattern Recognition*, Elsevier Science, pp. 1145-1159, 1997.

Appendices

List of symbols

Notation	Description
r	A distance between the two instances.
$A_1^k, A_2^k, \dots, A_{m_k}^k$	An attribute has values of fuzzy sets $A_1^k, A_2^k, \dots, A_{m_k}^k$ $A_1^k, A_2^k, \dots, A_{m_k}^k$.
$\mu_i^k(x)$	The membership value of instance x for the i^{th} value of k^{th} attribute.
D_{xy}	The distance between instance x and y .
$\mu_j(x)$	The membership value of instance x for class j .
$P(x)$	The local co-occurrence matrix of instance x .
$W^{(k)}$	The local co-occurrence matrix after attribute k is selected.
$L^{(k)}$	The classifiability of attribute k .
$\mu_{A_k}(x)$	An interval-valued fuzzy set.
$[\mu_A(x_i), 1 - \nu_A(x_i)]$	The interval of intuitionistic fuzzy sets; where $\nu_A(x_i) = 1 - \bar{\mu}_A(x_i)$
$\mu_A(x_i)$	$\mu_A(x_i) = [\underline{\mu}_A(x_i), \bar{\mu}_A(x_i)] = [\underline{\mu}_A(x_i), 1 - \nu_A(x_i)]$.
d	The distance between A and B in the form of interval-valued fuzzy sets.
$\mu_i^{(k)}(x)$	The interval-valued fuzzy membership value of instance x for the i^{th} and k^{th} attribute, where $\mu_i^{(k)}(x) = [\underline{\mu}_i^{(k)}(x), \bar{\mu}_i^{(k)}(x)]$.
S	An universe of sample (objects) $S = S_1, S_2, \dots, S_n$ and $x_c \in S$.
$A^*(x_c)$	A set of the neighbourhoods of x_c that can be described by $A^*(x_c) = \{x_i \mid x_i \in S, 0 \leq \mu_c(x_i) \leq 1\}$.
$B^*(x_c)$	A set of the neighbourhoods of x_c described by $A^*(x_c)$ that it can be described by $B^*(x_c) = \{x_i \mid x_i \in S, \bar{\mu}_c(x_c) \geq \alpha\}$; $i = 1, 2, 3, \dots, n$.
$\hat{\mu}(x)_j$	The interval-valued fuzzy membership value of instance x for class j and let $[\underline{\mu}_1(x), \bar{\mu}_1(x)], \dots, [\underline{\mu}_c(x), \bar{\mu}_c(x)]$.
$\hat{\mu}(y)$	The fuzzy membership for y belong to the neighbourhood set of x .
$\hat{P}(x)$	The local co-occurrence matrix of instance x for the interval-valued fuzzy membership value.
α	The α -cut value for the neighbourhood of x ; where $0 \leq \alpha \leq 1$.
$A^{(k)}$ or $A^{(k)}(x)$	A fuzzy set.
$L_{r=c}^{(k)}$	The corresponding value of $L^{(k)}$ for the different r values.
$L_{\alpha=t}^{(k)}$	The corresponding value of $L^{(k)}$ for the different α values.
$TL_{(c,t) \mid r=c, \alpha=t}$	An average difference between an average difference between $L_{r=c}^{(k)}$ and $L_{\alpha=t}^{(k)}$ for all attributes.
$X = [\underline{X}, \bar{X}]$	An interval X as a closed, bounded set of real numbers, where $\underline{X} \leq x \leq \bar{X}$.
$\hat{W}^{(k)}$	The local co-occurrence matrix after attribute k is selected.
$\hat{L}^{(k)}$	The classifiability of attribute k for the interval-valued fuzzy membership value.
$P(x < y)_I$	Probability for x and y to belong to $X_i = X \cap Y$ and $x < y$ is within the intersection.
$P(x < y)_P$	Probability for x precedes the intersecting intervals.
$P(x < y)_F$	Probability for x to locate in the intersecting interval and y follows.
\emptyset	An empty set.