Applied Soft Computing Journal

Z-Number-Valued Rule-Based Classification System

--Manuscript Draft--

Manuscript Number:	ASOC-D-22-04627R1
Article Type:	Full Length Article
Keywords:	Z-numbers; Fuzzy Modeling; Z-number-valued if-then rule; rule-based system; classification

Abstract:	The fuzzy rule-based classification system (FRBCS) is a popular tool for classification problems due to its interpretability and comprehensibility. As an extension of fuzzy numbers, the concept of Z-number is a more appropriate formal structure to describe uncertain and partially reliable information. A Z-number is an ordered pair of fuzzy numbers, where the second fuzzy number describes the reliability of the first one. As a result of its representation capability, it can receive better classification results. However, there is still a gap in the application of Z-numbers to classification problems due to their high computation complexity. To take advantage of the Z-number, we design a simple way to make Z-numbers apply to classification problems. Use the second fuzzy number to adjust the first fuzzy number to fit the training data. Then we create a kind of Z-number-valued if-then rule by extending the fuzzy if-then rule. In addition, a Z-number-valued rule-based classification system (ZRBCS) is developed, including two main processes: rule generation and new pattern classification. The developed system can cover more information than the classic fuzzy rule-based system, which can improve classification effects. The proposed ZRBCS is compared with classical FRBCS with/without certain degrees and three classical classification algorithms. According to statistical tests, ZRBCS is superior to FRBCS and two other algorithms.

RESPONSES TO THE COMMMENTS

Reply to Reviewer #1

Thank you for your valuable comments and suggestions to improve this paper. We do our best to revise the paper according to your comments and hope the revised version can be accepted. If you think there are some other issues to be addressed, we will revise again based on your suggestions and comments.

Comment 1: What about running time (execution time) of the methods?
Additional results (a new table or chart/graph) may also be given in terms of running times.
Response: Thank you for your valuable comments to improve our paper. We added two tables to compare the running times of different methods. The revisions are listed as follows.

Page 14, Table 6

For both FRBCS and ZRBCS, the algorithm finds a rule for each training pattern and iterates all patterns to compute the consequent class of this rule. Therefore, the time complexity of FRBCS without certain degrees is $O(n^2 * M * K)$. And ZRBCS needs to iterate all patterns to compute the second number, which means its time complexity is higher than FRBCS, $O(n^3 * M * K)$. Table 6 lists the average running rime (including training time and testing time) of FRBCS and ZRBCS without certain degrees over 20 datasets. It can be seen that the proposed ZRBCS is faster than FRBCS without certain degrees.

	K=2		K=3		K=4		K=5		K=6	
	FRBCS	ZRBCS	FRBCS	ZRBCS	FRBCS	ZRBCS	FRBCS	ZRBCS	FRBCS	ZRBCS
Banknote	52.4	64.1	75.7	63.6	85.1	80.3	96.9	85.3	110	95.4
Breast	36.3	44	46	41.5	53.4	60	64.9	65.2	79.9	61.5
Bupa	28.1	35.2	43.2	39.4	52.4	56.3	64.1	64.3	73.4	70.5
Diabetes	58.6	72.7	106.8	94.8	177.6	173.9	237.3	236.5	308.8	293.2
Ecoli	34.8	41.7	52.8	51.3	68.8	69.3	78.7	78.2	90.8	88
Glass	33.2	40.9	41.9	38.1	49.8	51.6	57.7	58.3	65.8	63.1
Haberman	30.7	37.9	39.2	33.7	42.9	43.3	48.1	47	50.4	45.2
HTRU-2	3520.4	3536	4455.6	3816.6	5676.9	4571.8	5467.7	4552.7	6677.2	5250.2
Ionosphere	306.3	316.3	304.6	295.1	408.6	429.8	389.7	362	414.5	409.3
Iris	34.1	36.2	37	30.4	42.8	45.5	47.3	47.3	50.9	42.2
Liver	33.2	36.8	46.2	40.3	55.7	58.6	64.6	66.1	73.5	68.3
Newthyroid	31.6	35.2	38.9	33.6	45.3	47.8	48.1	49.3	50.9	43.1
Occupancy	105.9	106.4	103.2	94.6	115.7	104.8	105.4	123.8	109.3	128.1
Pageblocks	542.6	552.6	619.8	500.5	756.8	548	813.3	563.1	881.5	559.1
Pendigits	7260	4078.9	16241.2	10169	24758.6	18622.8	32568	25414.4	36001.2	29833.2
Pima	103.8	119.5	126	116.8	198.7	207.5	264.6	252.4	314.2	298.7
Rice	335.9	319.3	472.5	392.9	562	396.4	637.5	442.5	708.2	496.5
Segment	343	295	695.8	557	1112.5	1004.3	1471.3	1303.6	1718.7	1568.4
Tranfusion	40.4	50.1	44.6	41	49.2	49.7	52.8	52	56.9	50.5
Vehicle	240.1	273.2	479.8	481.2	779.4	766.9	871	837	924.7	915.8
Average	658.57	504.6	1203.54	846.57	1754.61	1369.43	2172.45	1735.05	2438.04	2019.015

Table 6: Running time (in millisecond) of the FRBCS and ZRBCS without certain degrees.

Page 16, Table 10

For both FRBCS and ZRBCS, the algorithm finds a rule for each training pattern and iterates all patterns to compute the consequent class and certain degree of this rule. Therefore, the time complexity of FRBCS with certain degrees is $O(n^2 * M * K)$. And ZRBCS needs to iterate all patterns to compute the second number, thus its time complexity is higher than FRBCS, $O(n^3 * M * K)$. Table 10 lists the average running time (including training time and testing time) of FRBCS and ZRBCS with certain degrees over 19 datasets. It can be seen that FRBCS is faster than the proposed ZRBCS with certain degrees.

	K=2		K=3		K=4		K=5		K=6	
	FRBCS	ZRBCS	FRBCS	ZRBCS	FRBCS	ZRBCS	FRBCS	ZRBCS	FRBCS	ZRBCS
Banknote	55.6	81.3	74.3	101	87.5	116.5	94.6	136.3	109.7	139.5
Breast	40.1	56	47.7	67.4	59.1	66.3	64.5	79.9	73.1	81.2
Bupa	33.8	40.2	46.1	51.1	56.5	61.8	63	67.8	75.5	79.2
Diabetes	66.1	93.7	100	123.8	184	202.5	237.2	261.2	324.5	345.2
Ecoli	35.6	44.1	50.5	56.9	66	78	78.4	81.8	94	107.7
Glass	31.6	42.5	38.8	45.8	47.8	54.7	53.9	60.2	60.5	63.7
Haberman	36.7	42.4	39.6	47.9	46.4	48.1	48.3	52.3	54	54.4
HTRU-2	3564.7	7635.4	4444.3	8837.5	5626.8	11205.5	5428.1	11437.4	6685.8	13123.5
Ionosphere	306	329.1	316.7	332.8	441.2	437.7	379.4	401.6	437.3	436.8
Iris	33.7	36.3	38.2	41.1	42.9	43.6	46.4	52.9	51	52.7
Liver	33.5	40.9	43.9	50.6	56.7	62.2	66.6	71.4	75.7	79.1
Newthyroid	31.2	35.5	38.6	41.2	44.1	51.6	50.2	53	51.3	52.6
Occupancy	110.2	171.4	104.7	151.7	116.2	158.1	125.6	165.9	130.5	178.7
Pageblocks	543.5	1099.7	616.7	1239.2	750.5	1395.7	810.6	1410.1	881	1459.1
Pendigits	6895.8	10620.3	16092.3	20190.9	24885.2	30204	32085.8	39141.9	36378.8	44502.1
Pima	107.1	128.1	125.8	150	196.8	224.9	264.2	280.7	324.6	344.9
Rice	351.1	517	475.2	774.3	582.7	885.6	639.2	932.9	723.8	1041.5
Segment	304.1	474.2	628.9	802.4	1055.7	1179.1	1408.7	1579.7	1765	2039.2
Tranfusion	40.9	61.8	48.6	62.1	49	63.3	51.9	65	56	71.9
Vehicle	124.8	169.6	332.3	377.1	708.2	667.6	828.1	833.6	932.7	999.2
Average	637.31	1085.98	1185.16	1677.24	1755.17	2360.34	2141.24	2858.28	2464.24	3262.61

Table 10: Running time (in millisecond) of the FRBCS and ZRBCS with certain degrees.

Comment 2: The "Related Work" section is missing. A new section may be added between "Introduction" and "Section 2".

Some parts of the "Introduction" section can be moved to the new section.

Response: Thank you for your careful reading and valuable comments to improve our paper. We have added a new section "Related works" between "Introduction" and "Preliminaries" that reviews some representative FRBCSs and applications and mathematical frameworks of Z-numbers. The revisions are listed as follows.

Section 2	

Section 2

2. Related works

This section introduces some representative FRBCS and the mathematical framework and applications of Z-numbers.

2.1. *Review of fuzzy rule-based classification system*

It is believed that human recognition practice may adopt a set of linguistic rules. In learning, object recognition, understanding, and other activities, rule-based symbolic processing may be the most effective tool to explain human intelligence [37, 38]. The use of fuzzy rules in classification systems can help achieve a balance between interpretability and accuracy. Several researchers have proposed methods for generating and learning fuzzy if-then rules from numerical data. The simplest method is the simple fuzzy grid method [28]. Afterwards, Ishibuchi et al. extended this work later by using the idea of sequential partitioning of the feature space into fuzzy subspaces until a predetermined stopping criterion is satisfied [39]. After that, Ishibuchi et al. suggested a genetic-algorithm-based method to minimize the number of significant fuzzy rules [33]. Ref. [40] discussed that a neuro-fuzzy model can learn fuzzy rules and membership functions from training data. Ishibuchi and Nakaskima designed a fuzzy rule-based system with triangular fuzzy sets [41], in which a genetic algorithm is used to generate rules. They also proposed a heuristic algorithm to compute rule weights. Ref. [42] presented a neuro-fuzzy scheme for realizing a fuzzy rule-based classifier along with feature selection. The network is used to learn the important features and the classification rules. In Ref. [43], authors proposed a support vector learning approach to construct fuzzy classifiers. The fuzzy rule-based classification system is constructed directly from the given training samples using the support vector learning approach. Ishibuchi and Yamamoto proposed two heuristic methods for rule weight specification [44]. Eghbal et al. also peoposed mansoori rule weight for fuzzy rule-based systems to improve performance [45]. Ref. discussed a fuzzy association rule-based classification method for high-dimensional problems [46]. In this method, a search tree is employed to generate fuzzy association rules. Hu et al. designed granular fuzzy rule-based models [32]. In Ref. [47], authors extended FRBCS to be able to slove big data classification problems. Niu et al. designed a novel fuzzy rule-based classification model for fuzzy granular rule learning [48]. The methods of rule generation, reasoning and rule weights of some representative FRBCSs are concluded in Table 1.

Ref.	Rule generation	Fuzzy reasoning	Rule weights
Ishibuchi et al. [28]	fuzzy grid method	winner rule	certain grades
Ishibuchi et al. [39]	sequential subdivision, fuzzy grid method	winner rule	certain grades
Ishibuchi et al. [33]	multiple fuzzy rule tables, genetic algorithm	winner rule	certain degrees
Nauck and Kruse [40], Mitra and Hayashi[29]	neuro-fuzzy model	neuro-fuzzy model	no weights
Ishibuchi and Nakaskima [41]	genetic algorithm	winner rule	heuristic algorithm
Chen et al. [43]	support vector learning	input output mapping	no weights
Ishibuchi and Yamamoto [44]	fuzzy grid method	winner rule	penalized certainty factor
Eghbal et al. [45]	uzzy grid method	winner rule	Mansoori rule weight
Alcalá-Fdez et al.[46]	search tree	fuzzy associated rule	no weights
Hu et al. [32]	granular computing	input output mapping	no weights
Sanz et al. [49]	apriori algorithm	fuzzy associated rule	certain grades

Table 1: The methods of rule generation, reasoning and rule weights of some representative FRBCSs.

2.2. *Review of Z-numbers*

The Z-number has been proven to be significant by many researchers. For example, in decision-making problems, Z-numbers can be useful to identify the unreliable information and utilize the partially reliable information. A lot of proposed Z-number-based decision-making methods which are based on traditional fuzzy decision-making methods have explained the superiority of Z-numbers. Such as the Z-number-based Analytical Hierarchy Process (Z-AHP) method [2], Best Worst Method based on Z-number (Z-BWM) [3], Z-number based technique for order preference by similarity to ideal solution (Z-TOPSIS) [4], Z-number based vlsekriterijum-ska optimizacija I kompromisno resenje (Z-VIKOR) [5, 6], decision-making based on Z-relation equation [7]. Furthermore, due to its modeling features, the Z-number has native advantages for natural language processing. Banerjee and Pal [8] extended the basic Z-numbers into a tool for level-2 computing with words (CWW) and consequently subjective natural language understanding, then use this extension to precise knowledge-frames. Banerjee and Pal [9] proposed augmentation of the Z-numbers for machine-perception encapsulation. In addition, Z-numbers have been successful applied to solve real-world problems, including dynamic control systems [13], safety analytics [14], risk assessment [15], medical diagnosis [17], data envelopment analysis [18], social media evaluation [19], clustering [20], and other fields.

The development of the mathematical frame of Z-numbers can build a firm basis for applications of Z-numbers. General formulas for addition, standard subtraction, square, square-root, and ranking of discrete Z-numbers were established by Aliev et al. [21]. After that, Aliev et al. extended the arithmetic of discrete Z-numbers into continuous Z-numbers [22]. Based on the arithmetic of Z-numbers and differential evolution optimization, Aliev et al. developed the Z-number-based linear programming (Z-LP) model [23]. Aliev extended the concept of specificity to Z-numbers and suggested operations on Z-number by using the horizontal membership concept [24]. Furthermore, Aliev et al. defined the formulation of Z-valued If-Then rule-based interpolation for approximate reasoning [25]. Kang et al. designed the definition of the total utility of Z-number [26]. Mazandarani and Zhao [27] introduced the concepts of Z-differentiability, Z-integral, and Z-Laplace transform of a Z-number-valued function and constructed the framework of calculus for Z-number, Z-calculus.

Comment 3: Providing a table that summarizes the related work would increase the understandability of the difference from the previous studies in the "Related Works" section.

Response: Thank you for your valuable comments to improve our paper. In the "Related works" section, we reviewed some representative FRBCSs and listed the methods of rule generation, reasoning and rule weights of them to show their differences in Table 1. The revisions are listed as follows.

<table>
<tr><td>Section 2.1</td><td>

2.1. Review of fuzzy rule-based classification system

It is believed that human recognition practice may adopt a set of linguistic rules. In learning, object recognition, understanding, and other activities, rule-based symbolic processing may be the most effective tool to explain human intelligence [37, 38]. The use of fuzzy rules in classification systems can help achieve a balance between interpretability and accuracy. Several researchers have proposed methods for generating and learning fuzzy if-then rules from numerical data. The simplest method is the simple fuzzy grid method [28]. Afterwards, Ishibuchi et al. extended this work later by using the idea of sequential partitioning of the feature space into fuzzy subspaces until a predetermined stopping criterion is satisfied [39]. After that, Ishibuchi et al. suggested a genetic-algorithm-based method to minimize the number of significant fuzzy rules [33]. Ref. [40] discussed that a neuro-fuzzy model can learn fuzzy rules and membership functions from training data. Ishibuchi and Nakaskima designed a fuzzy rule-based system with triangular fuzzy sets [41], in which a genetic algorithm is used to generate rules. They also proposed a heuristic algorithm to compute rule weights. Ref. [42] presented a neuro-fuzzy scheme for realizing a fuzzy rule-based classifier along with feature selection. The network is used to learn the important features and the classification rules. In Ref. [43], authors proposed a support vector learning approach to construct fuzzy classifiers. The fuzzy rule-based classification system is constructed directly from the given training samples using the support vector learning approach. Ishibuchi and Yamamoto proposed two heuristic methods for rule weight specification [44]. Eghbal et al. also peoposed mansoori rule weight for fuzzy rule-based systems to improve performance [45]. Ref. discussed a fuzzy association rule-based classification method for high-dimensional problems [46]. In this method, a search tree is employed to generate fuzzy association rules. Hu et al. designed granular fuzzy rule-based models [32]. In

Ref. [47], authors extended FRBCS to be able to slove big data classification problems. Niu et al. designed a novel fuzzy rule-based classification model for fuzzy granular rule learning [48]. The methods of rule generation, reasoning and rule weights of some representative FRBCSs are concluded in Table 1.

Ref.	Rule generation	Fuzzy reasoning	Rule weights
Ishibuchi et al. [28]	fuzzy grid method	winner rule	certain grades
Ishibuchi et al. [39]	sequential subdivision, fuzzy grid method	winner rule	certain grades
Ishibuchi et al. [33]	multiple fuzzy rule tables, genetic algorithm	winner rule	certain degrees
Nauck and Kruse [40], Mitra and Hayashi[29]	neuro-fuzzy model	neuro-fuzzy model	no weights
Ishibuchi and Nakaskima [41]	genetic algorithm	winner rule	heuristic algorithm
Chen et al. [43]	support vector learning	input output mapping	no weights
Ishibuchi and Yamamoto [44]	fuzzy grid method	winner rule	penalized certainty factor
Eghbal et al. [45]	uzzy grid method	winner rule	Mansoori rule weight
Alcalá-Fdez et al.[46]	search tree	fuzzy associated rule	no weights
Hu et al. [32]	granular computing	input output mapping	no weights
Sanz et al. [49]	apriori algorithm	fuzzy associated rule	certain grades

Table 1: The methods of rule generation, reasoning and rule weights of some representative FRBCSs.

</td></tr>
</table>

Comment 4: The authors propose a method without giving its time complexity using Big-O notation. What is the time complexity of the proposed method?
O(n2), O(nlogn) etc.

Response: Thank you for your valuable comments to improve our paper. We analyzed the time complexity of the proposed method and compared it with FRBCS. The revisions are listed as follows.

Section 5.2	For both FRBCS and ZRBCS, the algorithm finds a rule for each training pattern and iterates all patterns to compute the consequent class of this rule. Therefore, the time complexity of FRBCS without certain degrees is $O(n^2 * M * K)$. And ZRBCS needs to iterate all patterns to compute the second number, which means its time complexity is higher than FRBCS, $O(n^3 * M * K)$. Table 6 lists the average running rime (including training time and testing time) of FRBCS and ZRBCS without certain degrees over 20 datasets. It can be seen that the proposed ZRBCS is faster than FRBCS without certain degrees.
Section 5.3	For both FRBCS and ZRBCS, the algorithm finds a rule for each training pattern and iterates all patterns to compute the consequent class and certain degree of this rule. Therefore, the time complexity of FRBCS with certain degrees is $O(n^2 * M * K)$. And ZRBCS needs to iterate all patterns to compute the second number, thus its time complexity is higher than FRBCS, $O(n^3 * M * K)$. Table 10 lists the average running time (including training time and testing time) of FRBCS and ZRBCS with certain degrees over 19 datasets. It can be seen that FRBCS is faster than the proposed ZRBCS with certain degrees.

Comment 5: Some abbreviations are used in the text without giving their expansion.
For example; Z-TOPSIS, Z-VIKOR, etc.
The authors should write that "these abbreviations stand for what".

Response: Thank you for your valuable comments to improve our paper. We give the meanings of all the abbreviations The revisions are listed as follows.

Section 2.2	*2.2. Review of Z-numbers* The Z-number has been proven to be significant by many researchers. For example, in decision-making problems, Z-numbers can be useful to identify the unreliable information and utilize the partially reliable information. A lot of proposed Z-number-based decision-making methods which are based on traditional fuzzy decision-making methods have explained the superiority of Z-numbers. Such as the Z-number-based Analytical Hierarchy Process (Z-AHP) method [2], Best Worst Method based on Z-number (Z-BWM) [3], Z-number based technique for order preference by similarity to ideal solution (Z-TOPSIS) [4], Z-number based vlsekriterijum-ska optimizacija I kompromisno resenje (Z-VIKOR) [5, 6], decision-making based on Z-relation equation [7]. Furthermore, due to its modeling features, the Z-number has native advantages for natural language processing. Banerjee and Pal [8] extended the basic Z-numbers into a tool for level-2 computing with words (CWW) and consequently subjective natural language understanding, then use this extension to precise knowledge-frames. Banerjee and Pal [9] proposed augmentation of the Z-numbers for machine-perception encapsulation. In addition, Z-numbers have been successful applied to solve real-world problems, including dynamic control systems [13], safety analytics [14], risk assessment [15], medical diagnosis [17], data envelopment analysis [18], social media evaluation [19], clustering [20], and other fields. The development of the mathematical frame of Z-numbers can build a firm basis for applications of Z-numbers. General formulas for addition, standard subtraction, square, square-root, and ranking of discrete Z-numbers were established by Aliev et al. [21]. After that, Aliev et al. extended the arithmetic of discrete Z-numbers into continuous Z-numbers [22]. Based on the arithmetic of Z-numbers and differential evolution optimization, Aliev et al. developed the Z-number-based linear programming (Z-LP) model [23]. Aliev extended the concept of specificity to Z-numbers and suggested operations on Z-number by using the horizontal membership concept [24]. Furthermore, Aliev et al. defined the formulation of Z-valued If-Then rule-based interpolation for approximate reasoning [25]. Kang et al. designed the definition of the total utility of Z-number [26]. Mazandarani and Zhao [27] introduced the concepts of Z-differentiability, Z-integral, and Z-Laplace transform of a Z-number-valued function and constructed the framework of calculus for Z-number, Z-calculus.

Reply to Reviewer #2

Thank you for your valuable comments and suggestions to improve this paper. We do our best to revise the paper according to your comments and hope the revised version can be accepted. If you think there are some other issues to be addressed, we will revise again based on your suggestions and comments.

Comment 1: Make better presentation of new aspects in your idea to show advances of proposed processing. How proposed model makes contribution to fuzzy systems also in application part?

Response: Thank you for your valuable comments to improve our paper. We added a lot of descriptions in Introduction to show the innovations of our approach. The revisions are listed as follows.

Introduction	The ~~concept of~~ antecedent fuzzy numbers in FRBCS are set at the beginning and will not be changed during the training process. These preset fuzzy numbers completely determine classification performance. The use of Z-numbers can change this situation and improve the classification performance of fuzzy system. As an extension of fuzzy numbers, Z-number is a more appropriate formal structure for representing real-world information than fuzzy numbers, which can describe both uncertainty and partial reliability of information. A Z-number consists of two fuzzy numbers, $Z = (A, B)$, where A is a fuzzy restriction on the values of the variable, and B is a fuzzy restriction on the reliability measure of A [1]. However, due to there being many optimization problems in computing the underlying probability density functions, Z-numbers have high computation complexity, making them inefficient to be applied to classification problems. And the ~~nderlying~~ underlying probability density functions are counterintuitive in classification problems. In this paper, we design a simple way to make Z-numbers apply to classification problems. ~~Using~~ Use the second fuzzy number to adjust the first fuzzy number to fit the ~~unevenly distributed~~ training data better. The larger the reliability, the larger the pattern space covered by the first fuzzy number. On the contrary, the smaller the reliability, the smaller the pattern space covered by the first fuzzy number. Based on this, we extend the FRBCS ~~into~~ to the Z-number-valued framework. There are two problems associated with it: generation of Z-number-valued if-then rules from training patterns and classification of new patterns. The aim of this paper is to construct a basic Z-number-valued rule-based classification system (ZRBCS). We ~~defined~~ define the basic form of the Z-number-valued if-then rule: The first fuzzy number of an antecedent Z-number is the same as the antecedent fuzzy number of the FRBCS. It is possible to adjust the first fuzzy number to fit training patterns better by using the second fuzzy number of an antecedent Z-number. ~~In this paper, the~~ The second fuzzy number describes information about negative patterns' positions. ~~It is possible to correctly classify the negative patterns by~~ By adjusting the first fuzzy number according to their specific positions, negative patterns can be correctly classified. More information is considered in the proposed method than in the classical FRBCS. The main contributions of this paper can be summarized as follows. (1) Designed a simple method to make Z-numbers applicable to classification problems, avoiding complex optimization problems with Z-numbers and fitting data better. (2) The structure of Z-number-valued if-then rule for classification problems is defined. The antecedent fuzzy numbers in the fuzzy rule are replaced with Z-numbers. (3) The Z-number-valued if-then rule generation method is proposed. We make full use of the position information of the negative patterns by the FRBCS to generate the second fuzzy number of the antecedent Z-number. (4) The classification process of the new patterns is presented. Using the second fuzzy number, the fuzzy partitions are adjusted to fit the training patterns better. (5) ~~The Z-number-valued rule-based classification system is designed. Two experiments on 20 public datasets show that it can improve the learning ability and classification performance.~~ In order to evaluate the effectiveness of ZRBCS, three experiments are conducted to compare ZRBCS to FRBCS with/without certain degrees and three classical classification algorithms, including support vector machine(SVM) [37], kernel classifier [38] and cam weighted distance nearest neighbor Classifier (CamNN)[39]. The results of the first and the second experiments show that ZRBCS is superior to FRBCS with/without certain degrees. The result of the third experiment demonstrates that ZRBCS is superior to SVM and CamNN.

Comment 2: Related applications: Driving support by type-2 fuzzy logic control model, 6G-enabled IoT Home Environment control using Fuzzy Rules.

Response: Thank you for your valuable comments to improve our paper. We added these related works in Section 2.1. The revisions are listed as follows.

Section 2.1	It is believed that human recognition practice may adopt a set of linguistic rules. In learning, object recognition, understanding, and other activities, rule-based symbolic processing may be the most effective tool to explain human intelligence [37, 38]. The use of fuzzy rules in classification systems can help achieve a balance between interpretability [37] M. Woźniak, A. Zielonka, A. Sikora, Driving support by type-2 fuzzy logic control model, Expert Systems with Applications 207 (2022) 117798. [38] M. Woźniak, A. Zielonka, A. Sikora, M. J. Piran, A. Alamri, 6g-enabled iot home environment control using fuzzy rules, IEEE Internet of Things Journal 8 (7) (2020) 5442–5452.

Comment 3: Define limitations for practical use of this model. How the model works on unbalanced data or incomplete inputs? What are limitations of your idea?

Response: Thank you for your valuable comments to improve our paper. We specify the limitations of this model in Conclusions. The revisions are listed as follows.

Conclusions	Due to the proposed system in this paper just being a basic Z-number-valued rule-based system, ~~we~~ it must be noted that the proposed ZRBCS, like FRBCS, cannot handle incomplete and unbalanced data. We can use some auxiliary means to further improve ~~the classification accuracyand interpretability~~ classification accuracy, interpretability and handle incomplete and unbalanced data, such as genetic algorithms and clustering algorithms, and to reduce the dimensionality based on this basic Z-number-valued rule-based system in further work.

Comment 4: What type of fuzzy measures are applicable to your model? Did you test other configirations?

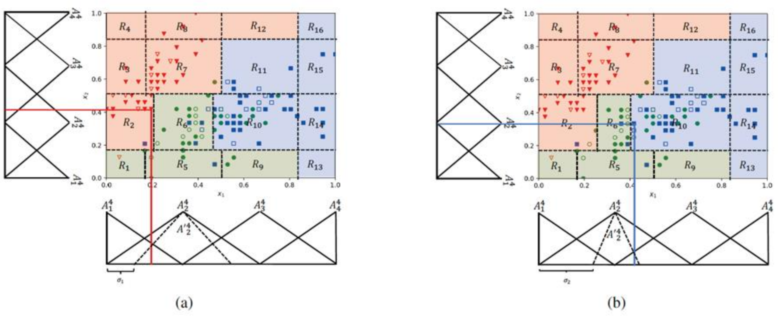
Response: Thank you for your valuable comments to improve our paper. The fuzzy rule-based classification system mentioned in this paper is based on the type-1 fuzzy measure. The proposed Z-number-valued rule-based classification system is based on the Z-number, which is composed of two type-1 fuzzy measures. The existing definition of Z-number is based on only type-1 fuzzy measures, so we did not consider other types in this paper. If more types of fuzzy measures are used in the definition of Z-numbers later, we will consider its classification model. We also mentioned it in the revised version. The revisions are listed as follows.

Section 3.1	*3.1. Z-numbers* **Definition 1.** *A Z-number, Z, has two components, (A, B). The first component A is a type-1 fuzzy number with membership function $\mu_A : U \rightarrow [0, 1]$, restricting on values of the variable, and the second component B is a type-1 fuzzy number with membership function $\mu_B : [0, 1] \rightarrow [0, 1]$ restricting on the reliability of A [1].* We only considers Z-numbers based on type-1 fuzzy numbers in this paper.

Reply to Reviewer #3

Thank you for your valuable comments and suggestions to improve this paper. We do our best to revise the paper according to your comments and hope the revised version can be accepted. If you think there are some other issues to be addressed, we will revise again based on your suggestions and comments.

Comment 1: Is there any theoretical, not just empirical, justification/proof for the correctness of the ZRBCS method in classification problems, as well as its superiority over FRBCS?
Response: Thank you for your valuable comments to improve our paper. We described in detail the theoretical process of the proposed method in adjusting the fuzzy partitions according to the negative patterns. The larger the reliability, the larger the pattern space covered by the first fuzzy number. On the contrary, the smaller the reliability, the smaller the pattern space covered by the first fuzzy number. Since in FRBCS, the fuzzy partitions are completely dependent on the initial K value, ZRBCS which can adjust the fuzzy partitions is definitely better than FRBCS which cannot adjust the fuzzy partitions. The revisions are listed as follows.

Introduction	The ~~concept of~~ antecedent fuzzy numbers in FRBCS are set at the beginning and will not be changed during the training process. These preset fuzzy numbers completely determine classification performance. The use of Z-numbers can change this situation and improve the classification performance of fuzzy system. As an extension of fuzzy numbers, Z-number is a more appropriate formal structure for representing real-world information than fuzzy numbers, which can describe both uncertainty and partial reliability of information. A Z-number consists of two fuzzy numbers, $Z = (A, B)$, where A is a fuzzy restriction on the values of the variable, and B is a fuzzy restriction on the reliability measure of A [1]. However, due to there being many optimization problems in computing the underlying probability density functions, Z-numbers have high computation complexity, making them inefficient to be applied to classification problems. And the ~~nderlying~~ underlying probability density functions are counterintuitive in classification problems. In this paper, we design a simple way to make Z-numbers apply to classification problems. ~~Using~~ Use the second fuzzy number to adjust the first fuzzy number to fit the ~~unevenly distributed~~ training data better. The larger the reliability, the larger the pattern space covered by the first fuzzy number. On the contrary, the smaller the reliability, the smaller the pattern space covered by the first fuzzy number. Based on this, we extend the FRBCS ~~into~~ to the Z-number-valued framework. There are two problems associated with it: generation of Z-number-valued if-then rules from training patterns and classification of new patterns.
Section 4.2	We can use the negative patterns to determine the second fuzzy number. If a negative pattern is just a bit over the threshold (the intersection of this fuzzy number with its adjacent fuzzy number), the fuzzy number just needs to adjust a bit so that it doesn't cover the pattern. For example ~~,~~ in Fig. 5a, the ~~fuzzy~~ 6th fuzzy partition covers some negative points, including some red triangles and some blue squares. If we adjust this fuzzy partition so that it does not cover these negative points. The fuzzy subset A_2^4 needs to be reduced by σ_1 to $A_2^{'4}$ to not cover the red point marked in the figure. Whenever a negative pattern is a lot over the threshold, the fuzzy number has to be adjusted by a lot to not cover it. In Fig. 5b, the fuzzy subset A_2^4 has to be reduced by σ_2 to $A_2^{'4}$ to not cover the blue point marked in the figure. It is obvious that σ_2 is greater than σ_1. The closer the membership degree is to the threshold, the closer the value $1.5 - \epsilon - \mu_{A_2^4}(x)$ is to $1 - \epsilon$, the smaller the adjustment needed for the fuzzy number. Where ϵ is the infinitesimals.  Figure 5: (a) A negative pattern that is a little over the threshold. (b) A negative pattern that is a lot over the threshold.

Comment 2: What is the difference in computational speed between the ZRBCS and FRBCS methods?
Response: Thank you for your valuable comments to improve our paper. We added two tables to compare the running times of different methods. The revisions are listed as follows.

Page 14, Table 6	For both FRBCS and ZRBCS, the algorithm finds a rule for each training pattern and iterates all patterns to compute the consequent class of this rule. Therefore, the time complexity of FRBCS without certain degrees is $O(n^2 * M * K)$. And ZRBCS needs to iterate all patterns to compute the second number, which means its time complexity is higher than FRBCS, $O(n^3 * M * K)$. Table 6 lists the average running rime (including training time and testing time) of FRBCS and ZRBCS without certain degrees over 20 datasets. It can be seen that the proposed ZRBCS is faster than FRBCS without certain degrees.

	K=2		K=3		K=4		K=5		K=6	
	FRBCS	ZRBCS	FRBCS	ZRBCS	FRBCS	ZRBCS	FRBCS	ZRBCS	FRBCS	ZRBCS
Banknote	52.4	64.1	75.7	63.6	85.1	80.3	96.9	85.3	110	95.4
Breast	36.3	44	46	41.5	53.4	60	64.9	65.2	79.9	61.5
Bupa	28.1	35.2	43.2	39.4	52.4	56.3	64.1	64.3	73.4	70.5
Diabetes	58.6	72.7	106.8	94.8	177.6	173.9	237.3	236.5	308.8	293.2
Ecoli	34.8	41.7	52.8	51.3	68.8	69.3	78.7	78.2	90.8	88
Glass	33.2	40.9	41.9	38.1	49.8	51.6	57.7	58.3	65.8	63.1
Haberman	30.7	37.9	39.2	33.7	42.9	43.3	48.1	47	50.4	45.2
HTRU-2	3520.4	3536	4455.6	3816.6	5676.9	4571.8	5467.7	4552.7	6677.2	5250.2
Ionosphere	306.3	316.3	304.6	295.1	408.6	429.8	389.7	362	414.5	409.3
Iris	34.1	36.2	37	30.4	42.8	45.5	47.3	47.3	50.9	42.2
Liver	33.2	36.8	46.2	40.3	55.7	58.6	64.6	66.1	73.5	68.3
Newthyroid	31.6	35.2	38.9	33.6	45.3	47.8	48.1	49.3	50.9	43.1
Occupancy	105.9	106.4	103.2	94.6	115.7	104.8	105.4	123.8	109.3	128.1
Pageblocks	542.6	552.6	619.8	500.5	756.8	548	813.3	563.1	881.5	559.1
Pendigits	7260	4078.9	16241.2	10169	24758.6	18622.8	32568	25414.4	36001.2	29833.2
Pima	103.8	119.5	126	116.8	198.7	207.5	264.6	252.4	314.2	298.7
Rice	335.9	319.3	472.5	392.9	562	396.4	637.5	442.5	708.2	496.5
Segment	343	295	695.8	557	1112.5	1004.3	1471.3	1303.6	1718.7	1568.4
Tranfusion	40.4	50.1	44.6	41	49.2	49.7	52.8	52	56.9	50.5
Vehicle	240.1	273.2	479.8	481.2	779.4	766.9	871	837	924.7	915.8
Average	658.57	504.6	1203.54	846.57	1754.61	1369.43	2172.45	1735.05	2438.04	2019.015

Table 6: Running time (in millisecond) of the FRBCS and ZRBCS without certain degrees.

Page 16, Table 10

For both FRBCS and ZRBCS, the algorithm finds a rule for each training pattern and iterates all patterns to compute the consequent class and certain degree of this rule. Therefore, the time complexity of FRBCS with certain degrees is $O(n^2 * M * K)$. And ZRBCS needs to iterate all patterns to compute the second number, thus its time complexity is higher than FRBCS, $O(n^3 * M * K)$. Table 10 lists the average running time (including training time and testing time) of FRBCS and ZRBCS with certain degrees over 19 datasets. It can be seen that FRBCS is faster than the proposed ZRBCS with certain degrees.

	K=2		K=3		K=4		K=5		K=6	
	FRBCS	ZRBCS	FRBCS	ZRBCS	FRBCS	ZRBCS	FRBCS	ZRBCS	FRBCS	ZRBCS
Banknote	55.6	81.3	74.3	101	87.5	116.5	94.6	136.3	109.7	139.5
Breast	40.1	56	47.7	67.4	59.1	66.3	64.5	79.9	73.1	81.2
Bupa	33.8	40.2	46.1	51.1	56.5	61.8	63	67.8	75.5	79.2
Diabetes	66.1	93.7	100	123.8	184	202.5	237.2	261.2	324.5	345.2
Ecoli	35.6	44.1	50.5	56.9	66	78	78.4	81.8	94	107.7
Glass	31.6	42.5	38.8	45.8	47.8	54.7	53.9	60.2	60.5	63.7
Haberman	36.7	42.4	39.6	47.9	46.4	48.1	48.3	52.3	54	54.4
HTRU-2	3564.7	7635.4	4444.3	8837.5	5626.8	11205.5	5428.1	11437.4	6685.8	13123.5
Ionosphere	306	329.1	316.7	332.8	441.2	437.7	379.4	401.6	437.3	436.8
Iris	33.7	36.3	38.2	41.1	42.9	43.6	46.4	52.9	51	52.7
Liver	33.5	40.9	43.9	50.6	56.7	62.2	66.6	71.4	75.7	79.1
Newthyroid	31.2	35.5	38.6	41.2	44.1	51.6	50.2	53	51.3	52.6
Occupancy	110.2	171.4	104.7	151.7	116.2	158.1	125.6	165.9	130.5	178.7
Pageblocks	543.5	1099.7	616.7	1239.2	750.5	1395.7	810.6	1410.1	881	1459.1
Pendigits	6895.8	10620.3	16092.3	20190.9	24885.2	30204	32085.8	39141.9	36378.8	44502.1
Pima	107.1	128.1	125.8	150	196.8	224.9	264.2	280.7	324.6	344.9
Rice	351.1	517	475.2	774.3	582.7	885.6	639.2	932.9	723.8	1041.5
Segment	304.1	474.2	628.9	802.4	1055.7	1179.1	1408.7	1579.7	1765	2039.2
Tranfusion	40.9	61.8	48.6	62.1	49	63.3	51.9	65	56	71.9
Vehicle	124.8	169.6	332.3	377.1	708.2	667.6	828.1	833.6	932.7	999.2
Average	637.31	1085.98	1185.16	1677.24	1755.17	2360.34	2141.24	2858.28	2464.24	3262.61

Table 10: Running time (in millisecond) of the FRBCS and ZRBCS with certain degrees.

Comment 3: Is there any superiority of methods based on fuzzy numbers compared to classical classification methods, such as SVM, XGBoost, etc.? The manuscript does not provide such comparisons.

Response: Thank you for your valuable comments to improve our paper. We added a new comparison experiment comparing the proposed ZRBCS with three classical classification algorithms, SVM, kernel and CamNN. statistic test results indicate that ZRBCS outperforms SVM and CamNN. The revisions are listed as follows.

Section 5.4

To further demonstrate the superiority of the proposed method, we further compare the ZRBCS with three classical classification algorithms, including support vector machine(SVM) [51], kernel classifier [52] and cam weighted distance nearest neighbor Classifier (CamNN)[53]. Since two datasets Ionosphere and Segment cannot be trained and tested by CamNN, there are 18 datasets to be compared. The classification accuracy rates over testing data are shown in Table 11. In which, the classification accuracy rates of FRBCS and ZRBCS are taken from the optimal values in Table 8. The values in parentheses mark the ranks of the classifiers: The better the performance, the higher the rank (in case of ties, average ranks are assigned). The last row lists the average rank of each classifier. It is shown that the proposed ZRBCS is superior to other methods. Nonparametric tests based on average ranks are used to compare classifiers statistically. First, the Friedman test is carried out to determine whether the results obtained by different methods present significant differences. Table 12 shows the result of Friedman test with a level of significance $\alpha = 0.05$. It can be seen that the statistic value is greater than the critical value, which means there are significant differences among the results. After that the Nemenyi test is used to compare the proposed ZRBCS with other ones, and the results are shown in Fig. 7. It is shown that the proposed ZRBCS is not significantly better than the methods kernel, but is significantly better than the methods SVM and CamNN. The superiority of the ZRBCS over FRBCS is discussed in the Section 5.2 and Section 5.3.

	SVM	kernel	CamNN	FRBCS	ZRBCS
Banknote	100.00% (1)	76.97% (5)	99.42% (2)	99.20% (4)	99.27% (3)
Breast	84.35% (5)	89.28% (4)	89.99% (3)	90.52% (2)	91.92% (1)
Bupa	63.16% (2)	68.92% (1)	54.43% (5)	61.34% (4)	62.46% (3)
Diabetes	64.84% (5)	72.39% (3)	68.36% (4)	72.92% (2)	75.14% (1)
Ecoli	61.96% (5)	81.58% (1)	80.69% (2)	79.20% (4)	79.47% (3)
Glass	70.01% (1)	35.57% (5)	46.42%)4	64.36% (2)	61.13% (3)
Haberman	70.25% (4)	72.23% (3)	66.37% (5)	73.53% (2)	74.17% (1)
HTRU-2	96.44% (5)	97.28% (4)	97.36% (3)	97.44% (2)	97.51% (1)
Iris	97.33% (1)	91.33% (5)	94.67% (4)	95.33% (2.5)	95.33% (2.5)
Liver	63.50% (2)	66.37% (1)	55.66% (5)	62.04% (4)	62.96% (3)
Newthyroid	93.96% (1)	88.40% (4)	87.45% (5)	91.26% (3)	93.10% (2)
Occupancy	97.90% (3)	98.31% (1.5)	98.31% (1.5)	97.82% (4.5)	97.82% (4.5)
Pageblocks	92.60% (4)	95.60% (1)	70.36% (5)	93.86% (3)	94.32% (2)
Pendigits	61.09% (5)	99.39% (1)	97.21% (4)	97.88% (3)	98.03% (2)
Pima	63.41% (5)	70.32% (3)	69.27% (4)	73.06% (2)	74.88% (1)
Rice	68.82% (5)	87.69% (4)	90.63% (3)	92.76% (1)	92.60% (2)
Tranfusion	71.25% (4)	77.13% (3)	67.38% (5)	77.14% (2)	78.61% (1)
Average rank	3.5	2.8611	3.8056	2.78	2.0556

Table 11: Classification accuracy rates (%) of SVM, kernel, CamNN, FRBCS and ZRBCS over testing data.

Statistic(χ_F)	Critical value	p-value	Hypothesis
3.8834	2.507	0.0067	Rejected

Table 12: The Firedman test result with α = 0.05.

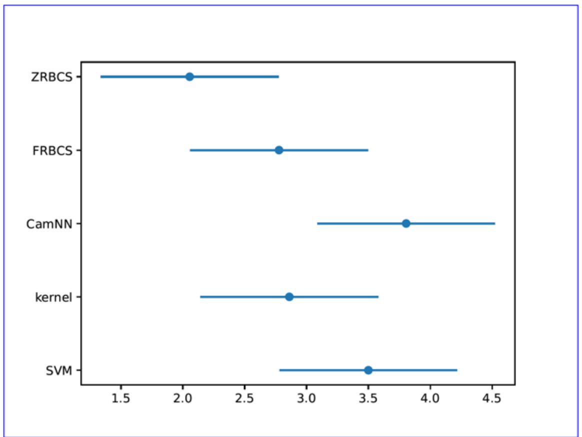


Figure 7: The Nemenyi test result with α = 0.05.

Comment 4: For testing, the Java-based software KEEL was used. Is it possible to implement these methods using programming languages like Python? How difficult is it? Could it speed up the calculations?

Response: Thank you for your valuable comments to improve our paper. Before using KEEL, we started out using Python to implement it. The implementation of algorithms by Python was easier than by Java. But for larger datasets, such as HTRU-2 and Pendigits, Python was too slow. Compared to Python, Java has higher speed and efficiency due to it being a compiled language. But as an interpreted language, Python has simpler, more concise syntax than Java. We also mentioned it in the paper. The revisions are listed as follows.

Section 3.1	~~on software KEEL~~in KEE (knowledge extraction based on evolutionary learning) software [53]. KEEL is an open source Java software tool that can be used for a large number of different knowledge data discovery tasks. Compared to Python, Java has higher speed and efficiency due to it being a compiled language. But as an interpreted language, Python has simpler, more concise syntax than Java. The details of ~~FRBCSs~~ other algorithms can be found on KEEL.

Highlights

- Designed a simple method to make Z-numbers apply to classification problems.
- The structure of Z-number-valued if-then rule for classification problems is defined.
- The Z-number-valued rule-based classification system is designed.

Z-Number-Valued Rule-Based Classification System

Yangxue Li[a], Enrique Herrera-Viedma[a,*], Ignacio Javier Pérez[a], Mónica Barragán-Guzmán[a], Juan Antonio Morente-Molinera[a,*]

[a]Andalusian Research Institute in Data Science and Computational Intelligence, Dept. of Computer Science and AI, University of Granada, 18071 Granada, Spain.

Abstract

The fuzzy rule-based classification system (FRBCS) is a popular tool for classification problems due to its interpretability and comprehensibility. As an extension of fuzzy numbers, the concept of Z-number is a more appropriate formal structure to describe uncertain and partially reliable information. A Z-number is an ordered pair of fuzzy numbers, where the second fuzzy number describes the reliability of the first one. As a result of its representation capability, it can receive better classification results. However, there is still a gap in the application of Z-numbers to classification problems due to their high computation complexity. To take advantage of the Z-number, we design a simple way to make Z-numbers apply to classification problems. Use the second fuzzy number to adjust the first fuzzy number to fit the training data. Then we create a kind of Z-number-valued if-then rule by extending the fuzzy if-then rule. In addition, a Z-number-valued rule-based classification system (ZRBCS) is developed, including two main processes: rule generation and new pattern classification. The developed system can cover more information than the classic fuzzy rule-based system, which can improve classification effects. The proposed ZRBCS is compared with classical FRBCS with/without certain degrees and three classical classification algorithms. According to statistical tests, ZRBCS is superior to FRBCS and two other algorithms.

Keywords: Z-numbers, fuzzy modeling, Z-number-valued if-then rule, rule-based system, classification

1. Introduction

The fuzzy rule-based classification system (FRBCS) is one of the most successful applications of fuzzy sets and fuzzy logic, because of its interpretability and comprehensibility. In classification problems, the main issue for constructing a fuzzy rule-based system is to generate fuzzy if-then rules from numerical data. A fuzzy if-then rule for a classification problem can be written as follows:

If input pattern is antecedent fuzzy numbers then consequent class.

In order to generate a fuzzy rule, we need to determine its antecedent fuzzy numbers, consequent fuzzy classes, and rule weight. The simplest method is the simple fuzzy grid method [1], which is based on local representations, and equally divides a dimension into K, $K = 2, 3, 4, \cdots$, triangular fuzzy numbers. These triangular fuzzy numbers divide the pattern space into M^K different fuzzy divisions, where M is the number of features. Its classification performance totally depends on fuzzy partitions. If the fuzzy partitions are too sparse, many patterns may be misclassified. But if the fuzzy partitions are too dense, many useless rules may be generated since the lack of training patterns. In Ref. [1], the certain grades of rules are used to improve the classification performance. This simple heuristic approach requires neither time-consuming iterative computation nor complicated learning procedures. Many different FRBCSs

*Corresponding authors.
Email addresses: viedma@decsai.ugr.es (Enrique Herrera-Viedma), ijperez@decsai.ugr.es (Ignacio Javier Pérez), monicabarragan@ugr.es (Mónica Barragán-Guzmán), jamoren@decsai.ugr.es (Juan Antonio Morente-Molinera)

have been proposed to further improve the performance and interpretability of the system, such as neuro-fuzzy rule-based classification system [2], rule-based granular classification [3, 4], genetic-fuzzy algorithms [5], fuzzy rule-based clustering system [6].

The antecedent fuzzy numbers in FRBCS are set at the beginning and will not be changed during the training process. These preset fuzzy numbers completely determine classification performance. The use of Z-numbers can change this situation and improve the classification performance of fuzzy system. As an extension of fuzzy numbers, Z-number is a more appropriate formal structure for representing real-world information than fuzzy numbers, which can describe both uncertainty and partial reliability of information. A Z-number consists of two fuzzy numbers, $Z = (A, B)$, where A is a fuzzy restriction on the values of the variable, and B is a fuzzy restriction on the reliability measure of A [7]. However, due to there being many optimization problems in computing the underlying probability density functions, Z-numbers have high computation complexity, making them inefficient to be applied to classification problems. And the underlying probability density functions are counterintuitive in classification problems. In this paper, we design a simple way to make Z-numbers apply to classification problems. Use the second fuzzy number to adjust the first fuzzy number to fit the training data better. The larger the reliability, the larger the pattern space covered by the first fuzzy number. On the contrary, the smaller the reliability, the smaller the pattern space covered by the first fuzzy number. Based on this, we extend the FRBCS to the Z-number-valued framework. There are two problems associated with it: generation of Z-number-valued if-then rules from training patterns and classification of new patterns.

The aim of this paper is to construct a basic Z-number-valued rule-based classification system (ZRBCS). We define the basic form of the Z-number-valued if-then rule:

If input pattern is antecedent Z-numbers then consequent class.

The first fuzzy number of an antecedent Z-number is the same as the antecedent fuzzy number of the FRBCS. It is possible to adjust the first fuzzy number to fit training patterns better by using the second fuzzy number of an antecedent Z-number. The second fuzzy number describes information about negative patterns' positions. By adjusting the first fuzzy number according to their specific positions, negative patterns can be correctly classified. More information is considered in the proposed method than in the classical FRBCS.

The main contributions of this paper can be summarized as follows.

(1) Designed a simple method to make Z-numbers applicable to classification problems, avoiding complex optimization problems with Z-numbers and fitting data better.
(2) The structure of Z-number-valued if-then rule for classification problems is defined. The antecedent fuzzy numbers in the fuzzy rule are replaced with Z-numbers.
(3) The Z-number-valued if-then rule generation method is proposed. We make full use of the position information of the negative patterns by the FRBCS to generate the second fuzzy number of the antecedent Z-number.
(4) The classification process of the new patterns is presented. Using the second fuzzy number, the fuzzy partitions are adjusted to fit the training patterns better.

In order to evaluate the effectiveness of ZRBCS, three experiments are conducted to compare ZRBCS to FRBCS with/without certain degrees and three classical classification algorithms, including support vector machine(SVM) [8], kernel classifier [9] and cam weighted distance nearest neighbor Classifier (CamNN)[10]. The results of the first and the second experiments show that ZRBCS is superior to FRBCS with/without certain degrees. The result of the third experiment demonstrates that ZRBCS is superior to SVM and CamNN.

This study is set out as follows: Section 2 reviews some representative FRBCSs, and the existing mathematical framework and applications of Z-numbers. Section 3 recalls some background knowledge of Z-numbers, the FRBCS without certain grades and with certain grades. Section 4 details the construction procedures of the Z-number-valued if-then rule-based classification system. In Section 5, two experiments on 20 public datasets are performed to analyze and compare the learning and predictive abilities of ZRBCSs and FRBCSs. Finally, we conclude this study and provide some suggestions for future works in Section 6.

2. Related works

This section introduces some representative FRBCS and the mathematical framework and applications of Z-numbers.

2.1. Review of fuzzy rule-based classification system

It is believed that human recognition practice may adopt a set of linguistic rules. In learning, object recognition, understanding, and other activities, rule-based symbolic processing may be the most effective tool to explain human intelligence [11, 12]. The use of fuzzy rules in classification systems can help achieve a balance between interpretability and accuracy. Several researchers have proposed methods for generating and learning fuzzy if-then rules from numerical data. The simplest method is the simple fuzzy grid method [1]. Afterwards, Ishibuchi et al. extended this work later by using the idea of sequential partitioning of the feature space into fuzzy subspaces until a predetermined stopping criterion is satisfied [13]. After that, Ishibuchi et al. suggested a genetic-algorithm-based method to minimize the number of significant fuzzy rules [14]. Ref. [15] discussed that a neuro-fuzzy model can learn fuzzy rules and membership functions from training data. Ishibuchi and Nakaskima designed a fuzzy rule-based system with triangular fuzzy sets [16], in which a genetic algorithm is used to generate rules. They also proposed a heuristic algorithm to compute rule weights. Ref. [17] presented a neuro-fuzzy scheme for realizing a fuzzy rule-based classifier along with feature selection. The network is used to learn the important features and the classification rules. In Ref. [18], authors proposed a support vector learning approach to construct fuzzy classifiers. The fuzzy rule-based classification system is constructed directly from the given training samples using the support vector learning approach. Ishibuchi and Yamamoto proposed two heuristic methods for rule weight specification [19]. Eghbal et al. also peoposed mansoori rule weight for fuzzy rule-based systems to improve performance [20]. Ref. discussed a fuzzy association rule-based classification method for high-dimensional problems [21]. In this method, a search tree is employed to generate fuzzy association rules. Hu et al. designed granular fuzzy rule-based models [4]. In Ref. [22], authors extended FRBCS to be able to slove big data classification problems. Niu et al. designed a novel fuzzy rule-based classification model for fuzzy granular rule learning [23]. The methods of rule generation, reasoning and rule weights of some representative FRBCSs are concluded in Table 1.

Ref.	Rule generation	Fuzzy reasoning	Rule weights
Ishibuchi et al. [1]	fuzzy grid method	winner rule	certain grades
Ishibuchi et al. [13]	sequential subdivision, fuzzy grid method	winner rule	certain grades
Ishibuchi et al. [14]	multiple fuzzy rule tables, genetic algorithm	winner rule	certain degrees
Nauck and Kruse [15], Mitra and Hayashi[24]	neuro-fuzzy model	neuro-fuzzy model	no weights
Ishibuchi and Nakaskima [16]	genetic algorithm	winner rule	heuristic algorithm
Chen et al. [18]	support vector learning	input output mapping	no weights
Ishibuchi and Yamamoto [19]	fuzzy grid method	winner rule	penalized certainty factor
Eghbal et al. [20]	uzzy grid method	winner rule	Mansoori rule weight
Alcalá-Fdez et al.[21]	search tree	fuzzy associated rule	no weights
Hu et al. [4]	granular computing	input output mapping	no weights
Sanz et al. [25]	apriori algorithm	fuzzy associated rule	certain grades

Table 1: The methods of rule generation, reasoning and rule weights of some representative FRBCSs.

2.2. Review of Z-numbers

The Z-number has been proven to be significant by many researchers. For example, in decision-making problems, Z-numbers can be useful to identify the unreliable information and utilize the partially reliable information. A lot of proposed Z-number-based decision-making methods which are based on traditional fuzzy decision-making methods have explained the superiority of Z-numbers. Such as the Z-number-based Analytical Hierarchy Process (Z-AHP) method [26], Best Worst Method based on Z-number (Z-BWM) [27], Z-number based technique for order preference by similarity to ideal solution (Z-TOPSIS) [28], Z-number based vlsekriterijum-ska optimizacija I kompromisno resenje (Z-VIKOR) [29, 30], decision-making based on Z-relation equation [31]. Furthermore, due to its modeling features, the Z-number has native advantages for natural language processing. Banerjee and Pal [32] extended the basic Z-numbers into a tool for level-2 computing with words (CWW) and consequently subjective natural language understanding, then use this extension to precise knowledge-frames. Banerjee and Pal [33] proposed augmentation of the Z-numbers for machine-perception encapsulation. In addition, Z-numbers have been successful applied to solve real-world problems, including dynamic control systems [34], safety analytics [35], risk assessment [36], medical diagnosis [37], data envelopment analysis [38], social media evaluation [39], clustering [40], and other fields.

The development of the mathematical frame of Z-numbers can build a firm basis for applications of Z-numbers. General formulas for addition, standard subtraction, square, square-root, and ranking of discrete Z-numbers were established by Aliev et al. [41]. After that, Aliev et al. extended the arithmetic of discrete Z-numbers into continuous Z-numbers [42]. Based on the arithmetic of Z-numbers and differential evolution optimization, Aliev et al. developed the Z-number-based linear programming (Z-LP) model [43]. Aliev extended the concept of specificity to Z-numbers and suggested operations on Z-number by using the horizontal membership concept [44]. Furthermore, Aliev et al. defined the formulation of Z-valued If-Then rule-based interpolation for approximate reasoning [45]. Kang et al. designed the definition of the total utility of Z-number [46]. Mazandarani and Zhao [47] introduced the concepts of Z-differentiability, Z-integral, and Z-Laplace transform of a Z-number-valued function and constructed the framework of calculus for Z-number, Z-calculus.

3. Preliminaries

This section recalls the definition of the Z-number, the structures of the FRBCS without certain grades and with certain grades proposed in [1].

3.1. Z-numbers

Definition 1. *A Z-number, Z, has two components, (A, B). The first component A is a type-1 fuzzy number with membership function $\mu_A : U \to [0,1]$, restricting on values of the variable, and the second component B is a type-1 fuzzy number with membership function $\mu_B : [0,1] \to [0,1]$ restricting on the reliability of A [7].*

We only considers Z-numbers based on type-1 fuzzy numbers in this paper.

Definition 2. *A ordered triple (Y, A, B) is viewed as a Z-restriction on random variable Y [7]:*

$$Prob(Y \text{ is } A) \text{ is } B, \tag{1}$$

where Y is A is a fuzzy restriction, $Poss(Y = u) = \mu_A(u)$, $Poss(\cdot)$ is a possibility measure, μ_A is the membership function of A, and $u \in \mathcal{R}$; $Prob(Y \text{ is } A)$ is the probability measure of A, given by:

$$Prob(Y \text{ is } A) = \int_{\mathcal{R}} \mu_A(u) p_Y(u) du, \tag{2}$$

where p_Y the underlying probability density of Y. If A and p_Y are consistent, the centroids of A is same with the expected value of p_Y:

$$\int_{\mathcal{R}} u p_Y(u) du = \frac{\int_{\mathcal{R}} u \mu_A(u) du}{\int_{\mathcal{R}} u du} \tag{3}$$

The underlying probability density functions are not considered in this paper. The reason for this is not only that computing them involves many optimization problems, increasing the difficulty of the algorithm, but also that they are counterintuitive in classification problems.

Example 1. *Let us consider two Z-numbers: $Z_1 = (A_1 = (2,5,7), B_1 = (0.2, 0.3, 0.4))$, $Z_2 = (A_2 = (4,7,9), B_2 = (0.7, 0.8, 0.9))$. Obviously, the second one is more reliable than the first one. We often assume the probability density functions (pdfs) underlying them are normal pdfs [42]. When the reliability value is 0.3, the underlying pdf of Z_1 is $p_{Y1} = \mathcal{N}(\frac{14}{3}, 6.6902)$. When the reliability value is 0.8, the underlying pdf of Z_2 is $p_{Y2} = \mathcal{N}(\frac{20}{3}, 0.5940)$. These normal pdfs are depicted in Fig. 1. Although the reliability of the second Z-number is more reliable, the pdf of the first one covers more space (for the same variable, its probability density value of the first pdf is greater than that of another), even covering $Y = 8$ to 10. This means that more patterns will be covered by a less reliable Z-number, which is counterintuitive.*

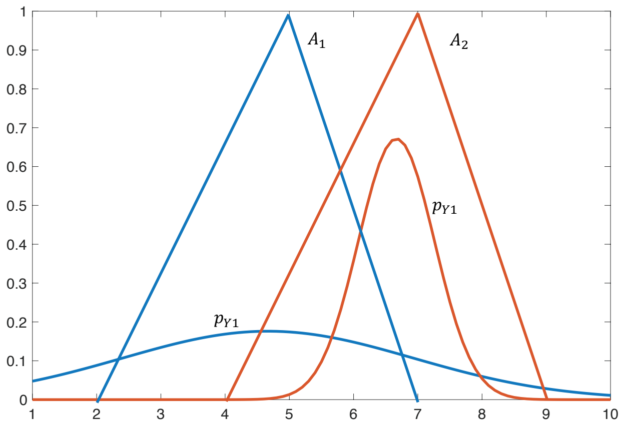


Figure 1: The underlying pdfs of two Z-numbers.

3.2. Fuzzy rule-based classification systems without certain grades

Definition 3. *For a W-class classification problem with M-dimensional inputs, the fuzzy if-then rule without certain grades is written as:*

$$\text{Rule } R_j\text{: If } x_{i1} \text{ is } A_{j1} \text{ and } x_{i2} \text{ is } A_{j2} \text{ and } \cdots \text{ and } x_{iM} \text{ is } A_{jM} \text{ then Class } C_j, \tag{4}$$

where $i = 1, 2, \cdots, n$, n is the number of patterns; $j = 1, 2, \cdots, N$, N is the number of rules; $x_i = \{x_{i1}, x_{i2}, \cdots, x_{iM}\}$ is the input vector; A_{jm} is the antecedent linguistic value, $m = 1, 2, \cdots, M$; C_j is the consequent class of j-th rule, $C_j \in \{c_1, c_2, \cdots, c_W\}$.

Ishibuchi et al. [1] proposed the simple fuzzy grid method to partition a dimension into some fuzzy numbers. The dimension is uniformly divided into K triangular fuzzy subsets: $A_1^K, A_2^K, \cdots, A_k^K, \cdots, A_K^K$, $k = 1, 2, \cdots, K$. The antecedent fuzzy subsets A_{jm} is one of $\{A_1^K, A_2^K, \cdots, A_k^K, \cdots, A_K^K\}$. Therefore, $N = K^M$ different fuzzy if-then rules partition the pattern space into N subspaces.

Definition 4. *The compatibility grade of rule R_j with the input pattern x_i is defined by product operator as*

$$\mu_j(x_i) = \mu_{A_{j1}}(x_{i1}) \times \mu_{A_{j2}}(x_{i2}) \times \cdots \times \mu_{A_{jM}}(x_{iM}), \tag{5}$$

where $\mu_{A_{jm}}$ is the membership function of fuzzy number A_{jm}, $j = 1, 2, \cdots, N$, $m = 1, 2, \cdots, M$.

Definition 5. *The compatibility grade of rule with the class c_ω is defined as:*

$$\psi_j(c_\omega) = \sum_{x_i \in c_\omega} \mu_j(x_i) \tag{6}$$

The consequent class of R_j, $C_j = c_{\omega^*}$, if

$$\psi_j(c_{\omega^*}) = \max(\psi_j(c_1), \psi_j(c_2), \cdots, \psi_j(c_W)) \tag{7}$$

If there are multiple maximum values in Eq. (7), the consequent class $C_j = \varnothing$.

For a new pattern x_{new}, it is classified by the single winner rule R_{j^*}, if

$$\mu_{j^*}(x_{new}) = \max(\mu_1(x_{new}), \mu_2(x_{new}), \cdots, \mu_N(x_{new})). \tag{8}$$

The class of this new pattern is the consequent class of R_{j^*}, C_{j^*}. When multiple rules take the maximum value in Eq. (8), the new pattern cannot be classified, considered as an unclassifiable pattern.

Example 2. *Let us consider a* 3*-class classification problem with two-dimensional input shown in Fig. 2. There are three classes noted by red inverted triangles, green points, and blue squares, in which the solid points are training patterns and the hollow points are testing patterns. When* $K = 2, 3, 4, 5$*, the pattern space is divided into* 4, 9, 16, 25 *fuzzy partitions (subspaces) shown as Figs. 3a, 3b, 3c and 3d, respectively. The consequent classes of subspaces are marked by the corresponding colors.*

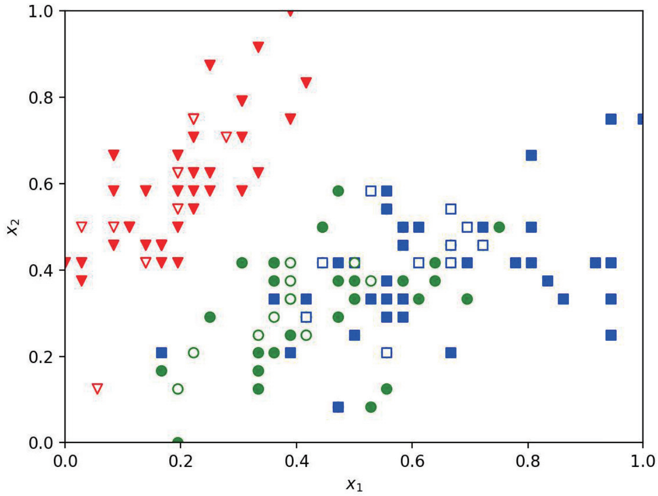


Figure 2: An example of 3-class classification problem with two-dimensional input.

3.3. Fuzzy rule-based classification systems with certain grades

Definition 6. *For a classification problem with a* M*-dimensional input vector* $x_i = \{x_{i1}, x_{i2}, \cdots, x_{iM}\}$*, the fuzzy if-then rule without certain grades is written as:*

$$\text{Rule } R_j: \text{ If } x_{i1} \text{ is } A_{j1} \text{ and } x_{i2} \text{ is } A_{j2} \text{ and } \cdots \text{ and } x_{iM} \text{ is } A_{jM} \text{ then Class } C_j \text{ with } CF_j, \tag{9}$$

where $i = 1, 2, \cdots, n$*;* $j = 1, 2, \cdots, N$*,* N *is the number of rules;* A_{jm} *is the antecedent linguistic value,* $m = 1, 2, \cdots, M$*;* C_j *is the consequent class of* j*-th rule,* $C_j \in \{c_1, c_2, \cdots, c_W\}$*;* CF_j *is the certain grade of rule* R_j *and* $CF_j \in [0, 1]$*.*

The certain grade is determined as:

$$CF_j = \frac{\psi_j(C_j) - \overline{\psi}}{\sum_{\omega=1}^{W} \psi_j(c_\omega)}, \tag{10}$$

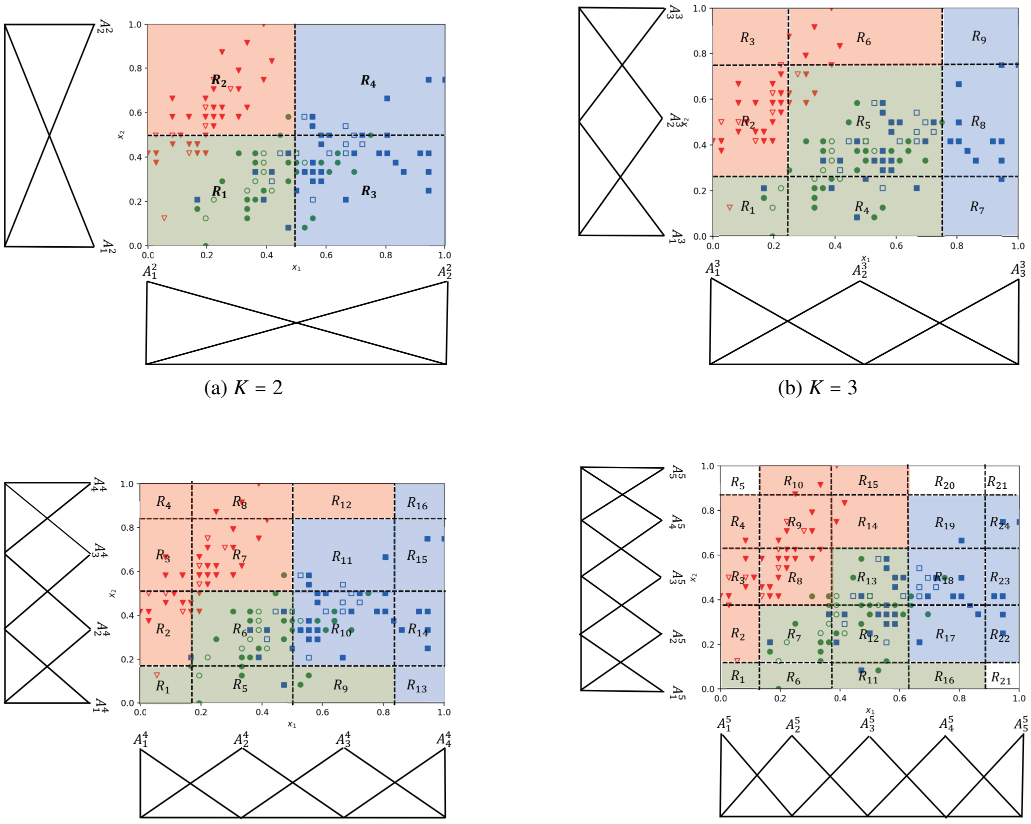
where

$$\overline{\psi} = \frac{\sum_{c_\omega \neq C_j} \psi_j(c_\omega)}{W - 1}. \tag{11}$$

For a new pattern x_{new}, it is classified as:

$$\mu_{j^*}(x_{new}) \times CF_{j^*} = \max(\mu_1(x_{new}) \times CF_1, \mu_2(x_{new}) \times CF_2, \cdots, \mu_N(x_{new}) \times CF_N). \tag{12}$$

The class of this new pattern is the consequent class of R_{j^*}, C_{j^*}. If multiple rules take the maximum value in Eq. (12), the new pattern cannot be classified, considered as an unclassifiable pattern.



(a) $K = 2$ (b) $K = 3$

(c) $K = 4$ (d) $K = 5$

Figure 3: Fuzzy partitions of each fuzzy rules when $K = 2$ (a), 3 (b), 4 (c), 5 (d).

4. Z-number-valued rule-based classification system

Based on the Z-number's powerful ability to represent and reason with uncertain and partially reliable information, we develop a Z-number-valued if-then rule-based classification system (ZRBCS). As shown in Fig. 4, the ZRBCS involves two main procedures: generating Z-number-valued if-then rules and classifying new patterns. The procedure of generating Z-number-valued if-then rule includes the first fuzzy numbers generation, the second fuzzy number generation and certain degrees determination.

4.1. Z-number-valued if-then rule structure

When we replace the antecedent fuzzy numbers in Eq. (4) with Z-numbers, the structure of the Z-number-valued if-then rule can be defined as follow.

Definition 7. *For a W-class classification problem with a M-dimensional input vector $x_i = \{x_{i1}, x_{i2}, \cdots, x_{iM}\}$, the Z-number-valued if-then rule is written as:*

$$\text{Rule } R_j^Z\text{: If } x_{i1} \text{ is } Z_{j1} \text{ and } x_{i2} \text{ is } Z_{j2} \text{ and } \cdots \text{ and } x_{iM} \text{ is } Z_{jM} \text{ then Class } C_j, \tag{13}$$

where $i = 1, 2, \cdots, n$, n is the number of patterns; $j = 1, 2, \cdots, N$, N is the number of rules; $Z_{jm} = (A_{jm}, B_{jm})$ is the antecedent Z-number, $m = 1, 2, \cdots, M$; C_j is the consequent class of j-th rule, $C_j \in \{c_1, c_2, \cdots, c_W\}$.

Similarly, when we replace the antecedent fuzzy subsets in Eq. (9), the structure of the Z-number-valued if-then rule with certain degree can be written as follows.

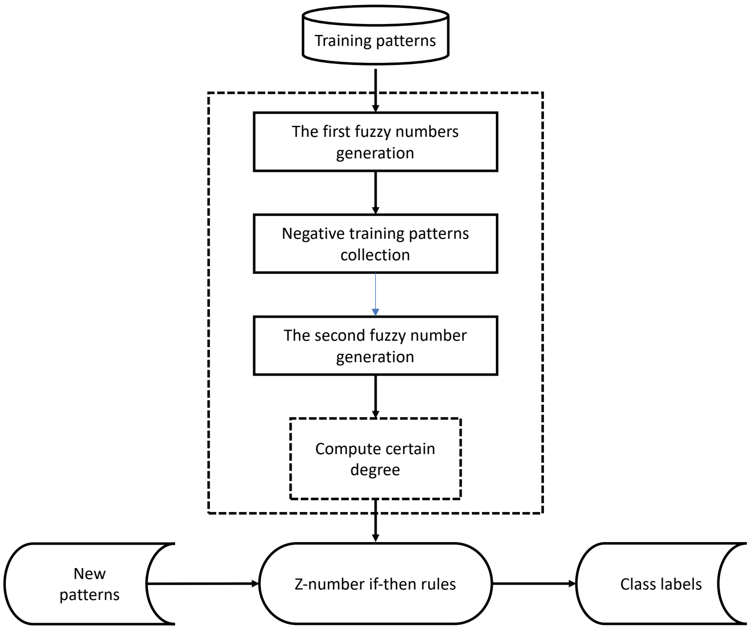


Figure 4: Z-number-valued rule-based classification system.

Definition 8. *For a W-class classification problem with a M-dimensional input vector* $x_i = \{x_{i1}, x_{i2}, \cdots, x_{iM}\}$*, the Z-number-valued if-then rule is written as:*

$$\text{Rule } R_j^Z\text{: If } x_{i1} \text{ is } Z_{j1} \text{ and } x_{i2} \text{ is } Z_{j2} \text{ and } \cdots \text{ and } x_{iM} \text{ is } Z_{jM} \text{ then Class } C_j \text{ with } CF_j, \tag{14}$$

where $i = 1, 2, \cdots, n$*, n is the number of patterns;* $j = 1, 2, \cdots, N$*, N is the number of rules;* $Z_{jm} = (A_{jm}, B_{jm})$ *is the antecedent Z-number,* $m = 1, 2, \cdots, M$*;* C_j *is the consequent class of j-th rule,* $C_j \in \{c_1, c_2, \cdots, c_W\}$*;* CF_j *is the certain degree of rule* R_j *and* $CF_j \in [0, 1]$.

4.2. Z-number-valued if-then rules generation

To generate Z-number-valued if-then rules from training patterns, three issues need to be addressed: the generation of the first antecedent fuzzy number, the generation of the second antecedent fuzzy number, and the determination of the consequent class. As with the FRBCS, the first antecedent fuzzy number and consequent class are generated using the simple fuzzy grid method and maximum compatibility grade method, respectively. The first step of generating Z-number-valued if-then rules is creating fuzzy if-then rules.

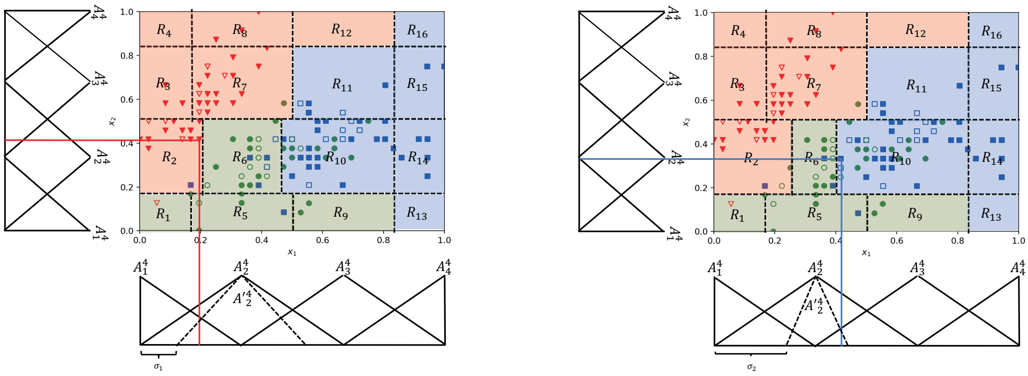
Let us consider the two-dimensional classification problem in Example 2. Let $K = 4$ and each dimension is divided into $K = 4$ triangular fuzzy numbers: A_1^4, A_2^4, A_3^4 and A_3^4. The pattern space is partitioned into 16 subspaces as shown in Fig. 3c. The fuzzy if-then rule represented by the sixth fuzzy partition is:

$$\text{Rule } R_6^Z\text{: If } x_{i1} \text{ is } A_2^4 \text{ and } x_{i2} \text{ is } A_2^4 \text{ then Class } c_2,$$

The consequent class of the sixth rectangle is c_2, but there are some red inverted triangles and blue squares are misclassified as c_2. Our aim is correctly to classify these patterns. In the FRBCS without certain degrees, the pattern space is uniformly divided into some fuzzy partitions. However, the patterns are not uniformly distributed. The certain degree is used to adjust these partitions vertically. The Z-numbers can adjust them horizontally in the ZRBCS.

Definition 9. *For a Z-number* $Z = (A, B)$*, where A and B are triangular fuzzy numbers* $A = (a_1, a_2, a_3)$ *and* $B = (b_1, b_2, b_3)$*,* $a_1, a_2, a_3, b_1, b_2, b_3 \in \mathcal{R}$*, we can use the reliability fuzzy number B to adjust A to obtain a new fuzzy number* $A' = (a'_1, a'_2, a'_3)$*, defined as:*

$$\begin{aligned} a'_1 &= a_2 - (a_2 - a_1) \times Rel(B), \\ a'_2 &= a_2, \\ a'_3 &= a_2 + (a_3 - a_2) \times Rel(B). \end{aligned} \tag{15}$$



(a) (b)

Figure 5: (a) A negative pattern that is a little over the threshold. (b) A negative pattern that is a lot over the threshold.

Where $Rel(B) = (b_1 + b_2 + b_3)/3$ is the reliability value.

We can use the negative patterns to determine the second fuzzy number. If a negative pattern is just a bit over the threshold (the intersection of this fuzzy number with its adjacent fuzzy number), the fuzzy number just needs to adjust a bit so that it doesn't cover the pattern. For example in Fig. 5a, the 6th fuzzy partition covers some negative points, including some red triangles and some blue squares. If we adjust this fuzzy partition so that it does not cover these negative points. The fuzzy subset A_2^4 needs to be reduced by σ_1 to $A_2^{'4}$ to not cover the red point marked in the figure. Whenever a negative pattern is a lot over the threshold, the fuzzy number has to be adjusted by a lot to not cover it. In Fig. 5b, the fuzzy subset A_2^4 has to be reduced by σ_2 to $A_2^{'4}$ to not cover the blue point marked in the figure. It is obvious that σ_2 is greater than σ_1. The closer the membership degree is to the threshold, the closer the value $1.5 - \epsilon - \mu_{A_2^4}(x)$ is to $1 - \epsilon$, the smaller the adjustment needed for the fuzzy number. Where ϵ is the infinitesimals.

The second step of generating Z-number-valued if-then rules is to collect the negative training patterns for each fuzzy if-then rule.

Definition 10. *For the rule R_j, its set of negative training patterns is defined as:*

$$E_j = \{x_e | \forall m = 1, 2, \cdots, M, \mu_{A_{jm}}(x_{em}) \geq 0.5, x_e \in C_j\}, \tag{16}$$

where 0.5 is the membership degree of a pattern that is on the intersection of two adjacent fuzzy numbers.

Based on the above analysis, we can define the second antecedent fuzzy number as:

Definition 11. *For the rule R_j, the second fuzzy number of Z_{jm}, B_{jm} is a triangular fuzzy number:*

$$B_{jm} = \begin{cases} (\min\limits_{x_e \in E_j}(1.5 - \epsilon - \mu_{A_{jm}}(x_{em})), \\ \dfrac{\sum_{x_{em} \in E_j}(1.5 - \epsilon - \mu_{A_{jm}}(x_{em}))}{|E_j|}, \\ \max\limits_{x_e \in E_j}(1.5 - \epsilon - \mu_{A_{jm}}(x_{em}))), & |E_j| \neq 0, \\ (1, 1, 1), & |E_j| = 0, \end{cases} \tag{17}$$

where $m = 1, 2, \cdots, M$, $|E_j|$ is the cardinality of E_j.

The algorithm of Z-number-valued if-then rules generation can be concluded as Algorithm 1.

Algorithm 1 Z-number-valued if-then rules generation algorithm.

Input: The training dataset $T = \{(x_i, c^i)|x_i \in \Re^M, c^i \in \{c_1, c_2, \cdots, c_M\}\}$, the number of fuzzy partitions K.
Output: A set of Z-number-valued if-then rules R^Z and the corresponding consequent classes set C

$N \leftarrow K^M$
$R^F, C = Fuzziness(T, K)$ // The simple fuzzy grid if-then rules generation, $R^F_j = \{A_{j1}, A_{j2}, \cdots, A_{jM}\}$, $C = \{C_1, C_2, \cdots, C_N\}$, $j = 1, 2, \cdots, N$
$E \leftarrow N * \varnothing$
for x_i in T **do**
    for R^F_j in R^F **do**
        if $\forall m = 1, \cdots, M, \mu_{A_{jm}}(x_{im}) \geq 0.5$ **then**
            x_i add in E_j
        end if
    end for
end for
for R^F_j in R^F **do**
    for $m = 1$ to M **do**
        B_{jm} calculated by Eq. (17)
    end for
    $R^Z_j \leftarrow \{(A_{j1}, B_{j1}), (A_{j2}, B_{j2}), \cdots, (A_{jM}, B_{jM})\}$
end for

return R^Z, C

Example 3. *Considering the previous two-dimensional classification problem with $K = 4$ shown in Fig. 3c in Example 2, the first rule is:*

$$\text{Rule } R^Z_1\text{: If } x_{i1} \text{ is } (A^4_1, B_{11}) \text{ and } x_{i2} \text{ is } (A^4_1, B_{12}) \text{ then Class } c_2.$$

Its set of misclassified training patterns is: $E_1 = \varnothing$. Thus the second fuzzy numbers B_{11} and B_{12} are $(1, 1, 1)$. The complete first Z-number-valued if-then rule is:

$$\text{Rule } R^Z_1: \text{ If } x_{i1} \text{ is } ((0, 0, 0.3333), (1, 1, 1)) \text{ and } x_{i2} \text{ is } ((0, 0, 0.3333), (1, 1, 1)) \text{ then Class } c_2.$$

The 7-th rule is:

$$\text{Rule } R_7\text{: If } x_{i1} \text{ is } (A^4_2, B_{71}) \text{ and } x_{i2} \text{ is } (A^4_3, B_{72}) \text{ then Class } c_1.$$

Its set of misclassified training datapoints is: $E_7 = \{\{0.4444, 0.5\}, \{0.4722, 0.5833)\}\}$. According to the Eq. (17), The second fuzzy number of Z_{71} is:

$$\begin{aligned}
B_{71} =& \Big(\min(1.5 - \epsilon - \mu_{A^4_2}(0.4444), 1.5 - \epsilon - \mu_{A^4_2}(0.4722)), \frac{(1.5 - \epsilon - \mu_{A^4_2}(0.4444)) + (1.5 - \epsilon - \mu_{A^4_2}(0.4722))}{2}, \\
& \max(1, 5 - \epsilon - \mu_{A^4_2}(0.4444), 1.5 - \epsilon - \mu_{A^4_2}(0.4722))\Big) \\
& = (\min(0.8333 - \epsilon, 0.9166 - \epsilon), \frac{0.8333 - \epsilon + 0.9166 - \epsilon}{2}, \max(0.8333 - \epsilon, 0.9166 - \epsilon)) \\
& = (0.8333 - \epsilon, 0.8750 - \epsilon, 0.9166 - \epsilon).
\end{aligned}$$

The second fuzzy number of Z_{72} is:

$$\begin{aligned}
B_{72} =& \Big(\min(1.5 - \epsilon - \mu_{A^4_3}(0.5), 1.5 - \epsilon - \mu_{A^4_3}(0.5833)), \frac{(1.5 - \epsilon - \mu_{A^4_3}(0.5)) + (1.5 - \epsilon - \mu_{A^4_3}(0.5833))}{2}, \\
& \max(1, 5 - \epsilon - \mu_{A^4_3}(0.5), 1.5 - \epsilon - \mu_{A^4_2}(0.5833))\Big) \\
& = (\min(0.9999 - \epsilon, 0.75 - \epsilon), \frac{0.9999 - \epsilon + 0.75 - \epsilon}{2}, \max(0.9999 - \epsilon, 0.75 - \epsilon)) \\
& = (0.75 - \epsilon, 0.8750 - \epsilon, 0.9999 - \epsilon).
\end{aligned}$$

The complete 7*-th Z-number-valued if-then rule is:*

$$\text{Rule } R_7^Z: \text{If } x_{i1} \text{ is } ((0, 0.3333, 0.6667), (0.8333-\epsilon, 0.8750-\epsilon, 0.9166-\epsilon)) \text{ and } x_{i2} \text{ is } ((0.3333, 0.6667, 1.0), (0.75-\epsilon, 0.8750-\epsilon, 0.9999-\epsilon)) \text{ then Class } c_1.$$

The other 14 *Z-number-valued if-then rules are generated by the same way.*

4.3. ZRBCS with certain grades

In the FRBCS, the certain grades are calculated by Eq. (10). The certain grades in the ZRBCS are calculated in the same way. Only the fuzzy numbers adjusted by the second fuzzy number are used. For a Z-number-valued rule R_j,

$$\text{Rule } R_j^Z\text{: If } x_{i1} \text{ is } Z_{j1} \text{ and } x_{i2} \text{ is } Z_{j2} \text{ and } \cdots \text{ and } x_{iM} \text{ is } Z_{jM} \text{ then Class } C_j \text{ with } CF_j. \tag{18}$$

It is first necessary to introduce a few relevant definitions before discussing the calculation method of certain grades.

Definition 12. *The Z-compatibility grade of the Z-number-valued if-then rule R_j with the input pattern x_i is defined as*

$$\mu_j^Z(x_i) = \mu_{A'_{j1}}(x_{i1}) \times \cdots \times \mu_{A'_{jM}}(x_{iM}), \tag{19}$$

where A'_{jm} is the adjusted fuzzy number of the Z-number $Z_{jm} = (A_{jm}, B_{jm})$.

Definition 13. *The Z-compatibility grade of Z-number-valued rule R_j with class c_ω is defined as:*

$$\psi_j^Z(c_\omega) = \sum_{x_i \in C_j} \mu_j^Z(x_i). \tag{20}$$

CF_j can be calculated by:

$$CF_j = \frac{\psi_j^Z(C_j) - \overline{\psi}^Z}{\sum_{\omega=1}^{W} \psi_j^Z(c_\omega)}, \tag{21}$$

where

$$\overline{\psi}^Z = \frac{\sum_{c_\omega \neq C_j} \psi_j^Z(c_\omega)}{W-1}. \tag{22}$$

4.4. New patterns classification procedure

For a new pattern x_{new}, it is classified by rule R_{j^*}, if

$$\mu_{j^*}^Z(x_{new}) = \max(\mu_1^Z(x_{new}), \mu_2^Z(x_{new}), \cdots, \mu_M^Z(x_{new})). \tag{23}$$

Algorithm 2 New pattern classification algorithm.

Input: A new pattern $x_{new} = \{x_{new,1}, x_{new,2}, \cdots, x_{new,M}\}$, a set of Z-number-valued if-then rules R^Z and its corresponding consequent classes set C and certain degrees CF.
Output: The class of the new pattern, c_{new}.

for R_j^Z in R^Z **do**
  $\mu_j^Z \leftarrow 1$
  for $m = 1$ to M **do**
    $\mu_j^Z \leftarrow \mu_j^Z \times \mu_{A'_{jm}}(x_{new,m})$
  end for
end for
$j^* = argmax_{j=1,2,\cdots,N}(\mu_j^Z)$
if $max_{j=1,2,\cdots,N, j\neq j^*}(\mu_j^Z) = max_{j=1,2,\cdots,N}(\mu_j^Z)$ **then**
  $c_{new} \leftarrow misclassified$
else
  $c_{new} \leftarrow C_{j^*}$
end if
return c_{new}

If there are multiple rules that have the maximum Z-compatibility values, the new pattern cannot be classified, considering as an unclassifiable pattern. Else the class of x_{new} is the consequent class of R^Z_{j*}, $x_{new} \in C_{j*}$. The algorithm of the new pattern classification can be concluded as Algorithm 2.

Example 4. *A new pattern $x_{new} = \{0.1944, 0.6249\}$ input the ZRBCS generated in Example 3. The Z-compatibility grade of rules are:*

$$\mu^Z_2(x_{new}) = 0.0521;$$
$$\mu^Z_3(x_{new}) = 0.3646;$$
$$\mu^Z_7(x_{new}) = 0.4490.$$

The other Z-compatibility grades are zero. The 7-th rule R^Z_7 takes the maximum Z-compatibility grade, thus this new pattern is c_1, a red inverted triangle.

The certain degrees of 2-th, 3-th and 7-th rules are $CF_2 = 0.8412$, $CF_3 = 0.9963$ and $CF_7 = 0.7994$. The Z-compatibility grade with CF of rules are:

$$\mu^Z_2(x_{new}) \times CF_2 = 0.0438;$$
$$\mu^Z_3(x_{new}) \times CF_3 = 0.3633;$$
$$\mu^Z_7(x_{new}) \times CF_7 = 0.3589.$$

The 3-th rule R^Z_3 takes the maximum value. The consequent class of R^Z_3 also is c_1, a red inverted triangle.

5. Performance evaluation

This section conducts three experiments comparing ZRBCS and FRBCS with and without certain grades, and some classical classification algorithms.

Dataset	#Patterns	#Dimensions	#Classes
Banknote	1372	4	2
Breast	569	5	2
Bupa	345	6	2
Diabetes	768	8	2
Ecoli	336	7	8
Glass	214	9	6
Haberman	306	3	2
HTRU-2	17898	8	2
Ionosphere	351	34	2
Iris	150	4	3
Liver	345	6	2
Newthyroid	215	5	3
Occupancy	2665	5	2
Pageblocks	5473	10	5
Pendigits	10992	16	10
Pima	768	8	2
Rice	3810	7	2
Segment	2310	19	7
Tranfusion	748	4	2
Vehicle	846	18	4

Table 2: The description of used datasets.

5.1. Experimental set-up

The performance of the proposed ZRBCS is evaluated by three experiments over 20 datasets. These datasets vary in the number of classes, dimensions, and patterns. The description of these datasets is shown in Table 2, where '#Patterns' is the number of patterns, '#Dimensions' is the number of dimensions, and '#classes' is the number of classes.

For evaluating generalization ability, 10-fold cross-validation (10CV) is used. In it, the patterns are divided into 10 folds, each of them contains 10% patterns. Nine folds are used to train the model, and the other is used to test it. The average accuracy of 10 folds can show the performance of the model. All algorithms are implemented in KEE (knowledge extraction based on evolutionary learning) software [48]. KEEL is an open source Java software tool that can be used for a large number of different knowledge data discovery tasks. Compared to Python, Java has higher speed and efficiency due to it being a compiled language. But as an interpreted language, Python has simpler, more concise syntax than Java. The details of other algorithms can be found on KEEL. Five different fuzzy simple grid partitions ($K = 2, 3, 4, 5, 6$) are used to make the comparison, where each dimension is homogeneously divided into K triangular fuzzy numbers in the same manner.

5.2. Comparison without certain degrees

TTable 3 and Table 4 show the classification accuracy rates of the ZRBCS in comparison with FRBCS without certain degrees over the training data and the testing data, respectively. Bold-faced results indicate that the current method is superior. As listed in Table 3, the ZRBCS shows better learning ability over most datasets, meaning Z-numbers can indeed improve the learning ability of the FRBCS. When K = 2, 3, 4, 5 and 6, it performs better on predictive ability than FRBCS in 16, 17, 17, 18 and 17 out of 20 datasets. It also does not perform poorly in 17, 18, 18, 19 and 18 out of 20 datasets, respectively. We can conclude that ZRBCS outperforms FRBCS on most datasets. In order to find the superior approach statistically, we use nonparametric tests for multiple comparisons, taking into account the ranks over the test data. We use the Wilcoxon signed-rank test to compare the ZRBCS with FRBCS for each number of partitions, and $\alpha = 0.05$ with two-side. The statistic values and probability (p) values for each number of partitions are shown in Table 5. The critical value can be found in the table of critical values for the Wilcoxon signed-rank test. Clearly, the Wilcoxon signed-rank test rejects all hypotheses that there are statistically significant differences between the two methods. As a result, we can conclude from the statistical study in Table 5 that the ZRBCS is superior to FRBCS without certain degrees.

	K=2		K=3		K=4		K=5		K=6	
	FRBCS	ZRBCS	FRBCS	ZRBCS	FRBCS	ZRBCS	FRBCS	ZRBCS	FRBCS	ZRBCS
Banknote	64.07%	**80.29%**	69.55%	**76.08%**	72.64%	**81.37%**	79.69%	**89.39%**	94.84%	**97.50%**
Breast	37.26%	37.26%	62.35%	**85.59%**	79.77%	**85.26%**	84.57%	**87.97%**	80.98%	**87.82%**
Bupa	57.13%	**59.26%**	58.65%	**59.58%**	62.09%	**64.28%**	**69.27%**	68.63%	**78.32%**	76.46%
Diabetes	53.76%	**62.28%**	**71.12%**	70.10%	78.95%	**81.65%**	85.88%	**88.99%**	**94.85%**	93.88%
Ecoli	**61.28%**	57.31%	76.49%	**80.06%**	82.90%	**83.90%**	**90.91%**	90.67%	94.41%	**94.68%**
Glass	57.38%	**58.83%**	**57.32%**	50.96%	67.08%	**69.47%**	71.59%	**78.09%**	**85.93%**	83.55%
Haberman	73.60%	**73.64%**	**71.57%**	71.20%	**71.72%**	70.88%	**74.51%**	73.57%	**75.74%**	74.11%
HTRU-2	92.22%	**94.01%**	92.91%	**96.23%**	94.64%	**96.73%**	95.56%	**97.28%**	95.80%	**96.92%**
Ionosphere	97.59%	**98.35%**	**98.45%**	98.42%	100.0%	100.0%	100.0%	100.0%	100.0%	100.0%
Iris	60.52%	**65.33%**	92.00%	**94.59%**	85.41%	**87.63%**	84.96%	**85.26%**	96.96%	**97.56%**
Liver	57.13%	**59.13%**	58.78%	**59.45%**	62.45%	**63.96%**	**69.34%**	68.34%	**78.74%**	77.17%
Newthyroid	**83.72%**	80.36%	87.96%	**91.11%**	94.16%	**94.88%**	95.35%	**97.26%**	94.01%	**94.06%**
Occupancy	86.33%	**86.43%**	83.56%	**86.37%**	**97.90%**	97.86%	96.93%	**97.58%**	**96.48%**	96.37%
Pageblocks	89.66%	**90.31%**	91.16%	**91.48%**	92.67%	**92.88%**	91.06%	**92.83%**	92.48%	**92.98%**
Pendigits	85.34%	**89.74%**	96.67%	**98.44%**	99.47%	**99.78%**	99.887%	**99.892%**	99.97%	**99.98%**
Pima	58.28%	**64.42%**	41.70%	**54.76%**	76.30%	**80.64%**	87.44%	**87.51%**	**94.27%**	92.51%
Rice	45.39%	**87.53%**	57.02%	**78.54%**	48.49%	**64.11%**	69.46%	**83.79%**	68.07%	**81.01%**
Segment	**76.44%**	66.96%	82.36%	**85.65%**	**95.94%**	94.30%	**94.66%**	94.33%	96.41%	**96.91%**
Tranfusion	40.24%	**40.78%**	**51.77%**	49.82%	69.31%	**70.23%**	52.42%	**57.32%**	72.67%	**73.19%**
Vehicle	56.50%	**64.33%**	**80.98%**	77.96%	95.17%	**95.42%**	98.50%	**98.57%**	99.68%	**99.79%**

Table 3: Classification accuracy rate (%) of the FRBCS and ZRBCS without certain degrees over training data.

	K=2		K=3		K=4		K=5		K=6	
	FRBCS	ZRBCS	FRBCS	ZRBCS	FRBCS	ZRBCS	FRBCS	ZRBCS	FRBCS	ZRBCS
Banknote	64.07%	**80.25%**	69.39%	**76.02%**	72.52%	**81.27%**	79.52%	**89.28%**	94.39%	**97.30%**
Breast	37.26%	37.26%	61.33%	**84.72%**	78.20%	**83.99%**	82.25%	**86.47%**	78.03%	**86.48%**
Bupa	55.87%	**57.85%**	53.80%	**56.12%**	51.81%	**52.66%**	49.58%	**51.02%**	53.93%	**57.04%**
Diabetes	51.96%	**60.03%**	62.78%	**63.57%**	62.50%	**67.84%**	61.33%	**67.19%**	64.07%	**64.20%**
Ecoli	**58.02%**	55.32%	69.06%	**73.54%**	70.57%	**73.27%**	**77.40%**	77.39%	74.12%	**74.39%**
Glass	51.44%	**55.28%**	**47.76%**	41.23%	52.49%	**55.53%**	53.09%	**57.70%**	**63.32%**	56.76%
Haberman	72.87%	**73.19%**	69.92%	**70.26%**	66.04%	**66.06%**	64.71%	**65.71%**	60.48%	**61.47%**
HTRU-2	92.21%	**93.95%**	92.81%	**96.14%**	94.50%	**96.71%**	95.37%	**97.13%**	95.44%	**96.56%**
Ionosphere	64.96%	**66.10%**	63.79%	63.79%	56.98%	56.98%	54.40%	54.40%	46.43%	46.43%
Iris	60.00%	**64.67%**	91.33%	**94.67%**	84.00%	**86.67%**	82.00%	**84.67%**	90.00%	**91.33%**
Liver	55.92%	**59.09%**	52.18%	**53.94%**	48.99%	**51.88%**	50.20%	**54.55%**	51.93%	**54.80%**
Newthyroid	**83.72%**	80.48%	86.10%	**88.90%**	90.78%	**91.26%**	92.64%	**94.46%**	89.33%	**90.28%**
Occupancy	86.15%	**86.23%**	83.56%	**86.30%**	**97.90%**	97.86%	96.85%	**97.64%**	**96.44%**	96.29%
Pageblocks	89.60%	**90.26%**	90.72%	**91.19%**	92.05%	**92.36%**	90.19%	**92.03%**	91.21%	**91.92%**
Pendigits	84.08%	**88.63%**	94.88%	**97.03%**	97.64%	**98.02%**	95.62%	**95.65%**	90.40%	**90.44%**
Pima	56.10%	**62.13%**	33.84%	**49.22%**	59.88%	**66.01%**	63.03%	**66.03%**	64.17%	**65.60%**
Rice	45.54%	**87.51%**	56.90%	**78.48%**	48.27%	**64.20%**	68.87%	**83.46%**	67.30%	**80.42%**
Segment	**74.72%**	66.54%	79.48%	**84.03%**	**91.60%**	90.09%	89.65%	**89.74%**	90.35%	**90.95%**
Tranfusion	40.23%	**40.90%**	**52.01%**	51.07%	69.50%	**70.57%**	50.38%	**56.00%**	71.00%	**71.67%**
Vehicle	46.00%	**58.51%**	58.50%	**60.27%**	61.49%	**63.14%**	59.91%	**60.27%**	62.88%	**63.11%**

Table 4: Classification accuracy rate (%) of the FRBCS and ZRBCS without certain degrees over testing data.

K	Statistic value	p-value	Critical value	Hypothesis
$K = 2$	30	0.00285	52	Rejected
$K = 3$	20	0.00058	52	Rejected
$K = 4$	20	0.00058	52	Rejected
$K = 5$	10	9.61658e-05	52	Rejected
$K = 4$	20	0.00058	52	Rejected

Table 5: Wilcoxon signed-rank test of the accuracy rate for comparing ZRBCS to FRBCS without certain degrees.

	K=2		K=3		K=4		K=5		K=6	
	FRBCS	ZRBCS	FRBCS	ZRBCS	FRBCS	ZRBCS	FRBCS	ZRBCS	FRBCS	ZRBCS
Banknote	52.4	64.1	75.7	63.6	85.1	80.3	96.9	85.3	110	95.4
Breast	36.3	44	46	41.5	53.4	60	64.9	65.2	79.9	61.5
Bupa	28.1	35.2	43.2	39.4	52.4	56.3	64.1	64.3	73.4	70.5
Diabetes	58.6	72.7	106.8	94.8	177.6	173.9	237.3	236.5	308.8	293.2
Ecoli	34.8	41.7	52.8	51.3	68.8	69.3	78.7	78.2	90.8	88
Glass	33.2	40.9	41.9	38.1	49.8	51.6	57.7	58.3	65.8	63.1
Haberman	30.7	37.9	39.2	33.7	42.9	43.3	48.1	47	50.4	45.2
HTRU-2	3520.4	3536	4455.6	3816.6	5676.9	4571.8	5467.7	4552.7	6677.2	5250.2
Ionosphere	306.3	316.3	304.6	295.1	408.6	429.8	389.7	362	414.5	409.3
Iris	34.1	36.2	37	30.4	42.8	45.5	47.3	47.3	50.9	42.2
Liver	33.2	36.8	46.2	40.3	55.7	58.6	64.6	66.1	73.5	68.3
Newthyroid	31.6	35.2	38.9	33.6	45.3	47.8	48.1	49.3	50.9	43.1
Occupancy	105.9	106.4	103.2	94.6	115.7	104.8	105.4	123.8	109.3	128.1
Pageblocks	542.6	552.6	619.8	500.5	756.8	548	813.3	563.1	881.5	559.1
Pendigits	7260	4078.9	16241.2	10169	24758.6	18622.8	32568	25414.4	36001.2	29833.2
Pima	103.8	119.5	126	116.8	198.7	207.5	264.6	252.4	314.2	298.7
Rice	335.9	319.3	472.5	392.9	562	396.4	637.5	442.5	708.2	496.5
Segment	343	295	695.8	557	1112.5	1004.3	1471.3	1303.6	1718.7	1568.4
Tranfusion	40.4	50.1	44.6	41	49.2	49.7	52.8	52	56.9	50.5
Vehicle	240.1	273.2	479.8	481.2	779.4	766.9	871	837	924.7	915.8
Average	658.57	504.6	1203.54	846.57	1754.61	1369.43	2172.45	1735.05	2438.04	2019.015

Table 6: Running time (in millisecond) of the FRBCS and ZRBCS without certain degrees.

For both FRBCS and ZRBCS, the algorithm finds a rule for each training pattern and iterates all patterns to compute the consequent class of this rule. Therefore, the time complexity of FRBCS without certain degrees is $O(n^2 * M * K)$. And ZRBCS needs to iterate all patterns to compute the second number, which means its time complexity is higher than FRBCS, $O(n^3 * M * K)$. Table 6 lists the average running rime (including training time and testing time) of FRBCS and ZRBCS without certain degrees over 20 datasets. It can be seen that the proposed ZRBCS is faster than FRBCS without certain degrees.

5.3. Comparison with certain degrees

Certain degrees adjust the fuzzy partitions vertically, and Z-numbers adjust them horizontally. On the basis of ZRBCS, certain degrees further improve the system's learning and prediction capabilities. Table 7 and Table 8 show the classification accuracy of the FRBCS and ZRBCS with certain degrees over the training data and testing data, respectively. Bold-faced results indicate that the current method is superior to the other one. As listed in Table 7, the ZRBCS with certain degrees shows better learning ability than FRBCS over all datasets. This model performs better on predictive ability in 18, 18, 18, 12 and 11 out of 20 datasets. It is not worse in 19, 19, 19, 15 and 15 datasets, respectively, when $K = 2$, 3, 4, 5 and 6. The ZRBCS outperforms FRBCS for most datasets. To compare the results statistically, the Wilcoxon signed-rank test with $\alpha = 0.05$ with two-side is used to compare these two methods on predictive ability. The statistic values and p-values are shown in Table 9. The test rejects all hypotheses that there are statistically significant differences between the two methods. Therefore, we can conclude that ZRBCS is superior to FRBCS with certain degrees.

For both FRBCS and ZRBCS, the algorithm finds a rule for each training pattern and iterates all patterns to compute the consequent class and certain degree of this rule. Therefore, the time complexity of FRBCS with certain degrees is $O(n^2 * M * K)$. And ZRBCS needs to iterate all patterns to compute the second number, thus its time complexity is higher than FRBCS, $O(n^3 * M * K)$. Table 10 lists the average running time (including training time and testing time) of FRBCS and ZRBCS with certain degrees over 19 datasets. It can be seen that FRBCS is faster than the proposed ZRBCS with certain degrees.

	K=2		K=3		K=4		K=5		K=6	
	FRBCS	ZRBCS	FRBCS	ZRBCS	FRBCS	ZRBCS	FRBCS	ZRBCS	FRBCS	ZRBCS
Banknote	76.68%	**84.95%**	95.12%	**96.05%**	94.91%	**96.31%**	96.39%	**97.16%**	99.34%	**99.42%**
Breast	75.81%	**81.57%**	90.18%	**90.49%**	88.87%	**91.62%**	91.68%	**92.42%**	91.97%	**93.40%**
Bupa	58.00%	**58.16%**	59.87%	**62.68%**	67.79%	**71.43%**	73.53%	**74.04%**	75.91%	**77.49%**
Diabetes	65.76%	**67.29%**	75.49%	**78.82%**	80.87%	**82.48%**	85.79%	**86.46%**	90.19%	**90.31%**
Ecoli	72.06%	**74.21%**	77.74%	**83.53%**	86.04%	**88.82%**	92.92%	**93.59%**	92.89%	**93.52%**
Glass	53.43%	**59.40%**	**71.55%**	62.93%	65.73%	**71.50%**	**79.86%**	79.34%	78.61%	**81.31%**
Haberman	**73.75%**	73.71%	74.26%	**74.84%**	75.42%	**77.41%**	**79.08%**	79.05%	79.56%	**79.99%**
HTRU-2	93.36%	**95.33%**	95.60%	**96.78%**	96.36%	**96.98%**	97.11%	**97.52%**	97.57%	**97.66%**
Ionosphere	91.23%	**93.67%**	97.66%	**97.97%**	99.49%	99.49%	99.43%	99.43%	100.0%	100.0%
Iris	67.48%	**78.15%**	93.78%	**94.89%**	89.26%	**93.11%**	**96.15%**	96.07%	**98.15%**	97.56%
Liver	58.00%	**58.16%**	59.68%	**61.80%**	68.02%	**71.47%**	73.43%	**74.17%**	75.84%	**77.62%**
Newthyroid	79.54%	**82.43%**	86.15%	**90.13%**	91.63%	**94.42%**	93.44%	**94.94%**	93.54%	**94.94%**
Occupancy	85.44%	**85.80%**	93.03%	**96.11%**	97.05%	**97.77%**	**97.89%**	97.86%	97.79%	97.79%
Pageblocks	89.99%	**90.83%**	91.93%	**92.71%**	92.75%	**93.29%**	93.72%	**94.25%**	94.62%	**95.04%**
Pendigits	85.13%	**91.71%**	98.18%	**98.53%**	99.35%	**99.50%**	99.79%	**99.82%**	99.96%	**99.97%**
Pima	65.29%	**66.19%**	75.22%	**77.37%**	80.02%	**80.66%**	84.62%	**85.07%**	88.79%	**89.00%**
Rice	91.12%	**91.82%**	88.26%	**91.59%**	91.99%	**92.39%**	91.55%	**91.99%**	92.83%	**92.87%**
Segment	64.85%	**69.10%**	85.66%	**88.67%**	91.25%	**91.63%**	**94.11%**	93.87%	96.43%	**96.52%**
Tranfusion	76.46%	**76.75%**	76.59%	**77.12%**	77.06%	**78.02%**	77.45%	**77.87%**	78.06%	**79.65%**
Vehicle	41.84%	**47.60%**	69.88%	**70.19%**	84.84%	**86.01%**	92.84%	**93.42%**	**98.24%**	98.23%

Table 7: Classification accuracy rate (%) of the FRBCS and ZRBCS with certain degrees over training data.

	K=2		K=3		K=4		K=5		K=6	
	FRBCS	ZRBCS	FRBCS	ZRBCS	FRBCS	ZRBCS	FRBCS	ZRBCS	FRBCS	ZRBCS
Banknote	76.60%	**85.13%**	94.97%	**95.92%**	94.68%	**96.21%**	96.36%	**97.08%**	99.20%	**99.27%**
Breast	75.57%	**81.55%**	89.63%	**90.34%**	88.58%	**91.03%**	90.52%	**91.05%**	90.34%	**91.92%**
Bupa	57.89%	57.89%	57.87%	**58.74%**	61.34%	**62.46%**	59.42%	**60.56%**	57.03%	**58.12%**
Diabetes	65.63%	**66.54%**	72.79%	**75.14%**	**72.92%**	72.66%	71.48%	**71.62%**	**69.02%**	68.37%
Ecoli	69.96%	**71.47%**	73.22%	**79.19%**	73.86%	**76.81%**	79.20%	**79.47%**	**77.69%**	77.09%
Glass	50.50%	**56.98%**	**64.36%**	55.35%	52.66%	**57.93%**	**61.36%**	61.13%	56.97%	**60.94%**
Haberman	**73.53%**	72.54%	73.19%	**74.17%**	71.23%	**73.82%**	**72.54%**	70.59%	70.25%	70.25%
HTRU-2	93.40%	**95.31%**	95.59%	**96.78%**	96.36%	**96.95%**	97.05%	**97.46%**	97.44%	**97.51%**
Ionosphere	65.52%	**68.67%**	64.65%	**64.94%**	56.70%	56.70%	54.40%	54.40%	45.86%	45.86%
Iris	67.33%	**78.00%**	92.67%	**94.67%**	87.33%	**89.33%**	95.33%	95.33%	**93.33%**	92.00%
Liver	57.98%	**58.27%**	58.54%	**60.01%**	61.77%	**62.96%**	**62.04%**	60.87%	56.83%	**57.97%**
Newthyroid	79.57%	**81.88%**	84.70%	**88.42%**	89.85%	**93.10%**	91.26%	**92.64%**	90.26%	**92.12%**
Occupancy	85.48%	**85.63%**	93.09%	**96.17%**	97.04%	**97.71%**	97.82%	97.82%	97.71%	97.71%
Pageblocks	89.95%	**90.81%**	91.76%	**92.47%**	92.38%	**93.00%**	93.13%	**94.25%**	93.86%	**94.32%**
Pendigits	84.87%	**91.35%**	97.52%	**97.85%**	97.88%	**98.03%**	95.88%	**95.90%**	90.44%	**90.45%**
Pima	65.11%	**65.89%**	73.06%	**74.88%**	72.53%	**73.18%**	71.76%	**72.14%**	69.40%	**69.53%**
Rice	91.13%	**91.92%**	88.19%	**91.52%**	92.02%	**92.36%**	91.31%	**91.76%**	**92.76%**	92.60%
Segment	64.72%	**68.61%**	84.03%	**87.01%**	88.74%	**89.05%**	**90.30%**	90.00%	**91.60%**	91.47%
Tranfusion	76.07%	**76.74%**	76.61%	76.61%	76.87%	**77.67%**	**76.47%**	76.34%	77.14%	**78.61%**
Vehicle	39.61%	**45.51%**	63.01%	**63.37%**	65.97%	**66.92%**	63.10%	**63.11%**	62.40%	62.40%

Table 8: Classification accuracy rate (%) of the FRBCS and ZRBCS with certain degrees over testing data.

K	Statistic value	p-value	Critical value	Hypothesis
$K = 2$	10	9.61658e-05	52	Rejected
$K = 3$	10	9.61658e-05	52	Rejected
$K = 4$	10	9.61658e-05	52	Rejected
$K = 5$	45	0.08956	52	Rejected
$K = 4$	42.5	0.13361	52	Rejected

Table 9: Wilcoxon signed-rank test of the accuracy rate for comparing ZRBCS to FRBCS with certain degrees.

	K=2		K=3		K=4		K=5		K=6	
	FRBCS	ZRBCS	FRBCS	ZRBCS	FRBCS	ZRBCS	FRBCS	ZRBCS	FRBCS	ZRBCS
Banknote	55.6	81.3	74.3	101	87.5	116.5	94.6	136.3	109.7	139.5
Breast	40.1	56	47.7	67.4	59.1	66.3	64.5	79.9	73.1	81.2
Bupa	33.8	40.2	46.1	51.1	56.5	61.8	63	67.8	75.5	79.2
Diabetes	66.1	93.7	100	123.8	184	202.5	237.2	261.2	324.5	345.2
Ecoli	35.6	44.1	50.5	56.9	66	78	78.4	81.8	94	107.7
Glass	31.6	42.5	38.8	45.8	47.8	54.7	53.9	60.2	60.5	63.7
Haberman	36.7	42.4	39.6	47.9	46.4	48.1	48.3	52.3	54	54.4
HTRU-2	3564.7	7635.4	4444.3	8837.5	5626.8	11205.5	5428.1	11437.4	6685.8	13123.5
Ionosphere	306	329.1	316.7	332.8	441.2	437.7	379.4	401.6	437.3	436.8
Iris	33.7	36.3	38.2	41.1	42.9	43.6	46.4	52.9	51	52.7
Liver	33.5	40.9	43.9	50.6	56.7	62.2	66.6	71.4	75.7	79.1
Newthyroid	31.2	35.5	38.6	41.2	44.1	51.6	50.2	53	51.3	52.6
Occupancy	110.2	171.4	104.7	151.7	116.2	158.1	125.6	165.9	130.5	178.7
Pageblocks	543.5	1099.7	616.7	1239.2	750.5	1395.7	810.6	1410.1	881	1459.1
Pendigits	6895.8	10620.3	16092.3	20190.9	24885.2	30204	32085.8	39141.9	36378.8	44502.1
Pima	107.1	128.1	125.8	150	196.8	224.9	264.2	280.7	324.6	344.9
Rice	351.1	517	475.2	774.3	582.7	885.6	639.2	932.9	723.8	1041.5
Segment	304.1	474.2	628.9	802.4	1055.7	1179.1	1408.7	1579.7	1765	2039.2
Tranfusion	40.9	61.8	48.6	62.1	49	63.3	51.9	65	56	71.9
Vehicle	124.8	169.6	332.3	377.1	708.2	667.6	828.1	833.6	932.7	999.2
Average	637.31	1085.98	1185.16	1677.24	1755.17	2360.34	2141.24	2858.28	2464.24	3262.61

Table 10: Running time (in millisecond) of the FRBCS and ZRBCS with certain degrees.

	SVM	kernel	CamNN	FRBCS	ZRBCS
Banknote	100.00% (1)	76.97% (5)	99.42% (2)	99.20% (4)	99.27% (3)
Breast	84.35% (5)	89.28% (4)	89.99% (3)	90.52% (2)	91.92% (1)
Bupa	63.16% (2)	68.92% (1)	54.43% (5)	61.34% (4)	62.46% (3)
Diabetes	64.84% (5)	72.39% (3)	68.36% (4)	72.92% (2)	75.14% (1)
Ecoli	61.96% (5)	81.58% (1)	80.69% (2)	79.20% (4)	79.47% (3)
Glass	70.01% (1)	35.57% (5)	46.42% ()4	64.36% (2)	61.13% (3)
Haberman	70.25% (4)	72.23% (3)	66.37% (5)	73.53% (2)	74.17% (1)
HTRU-2	96.44% (5)	97.28% (4)	97.36% (3)	97.44% (2)	97.51% (1)
Iris	97.33% (1)	91.33% (5)	94.67% (4)	95.33% (2.5)	95.33% (2.5)
Liver	63.50% (2)	66.37% (1)	55.66% (5)	62.04% (4)	62.96% (3)
Newthyroid	93.96% (1)	88.40% (4)	87.45% (5)	91.26% (3)	93.10% (2)
Occupancy	97.90% (3)	98.31% (1.5)	98.31% (1.5)	97.82% (4.5)	97.82% (4.5)
Pageblocks	92.60% (4)	95.60% (1)	70.36% (5)	93.86% (3)	94.32% (2)
Pendigits	61.09% (5)	99.39% (1)	97.21% (4)	97.88% (3)	98.03% (2)
Pima	63.41% (5)	70.32% (3)	69.27% (4)	73.06% (2)	74.88% (1)
Rice	68.82% (5)	87.69% (4)	90.63% (3)	92.76% (1)	92.60% (2)
Tranfusion	71.25% (4)	77.13% (3)	67.38% (5)	77.14% (2)	78.61% (1)
Average rank	3.5	2.8611	3.8056	2.78	2.0556

Table 11: Classification accuracy rates (%) of SVM, kernel, CamNN, FRBCS and ZRBCS over testing data.

Statistic(χ_F)	Critical value	p-value	Hypothesis
3.8834	2.507	0.0067	Rejected

Table 12: The Firedman test result with $\alpha = 0.05$.

5.4. Synthetic comparison

To demonstrate the advantages of the system in a more comprehensive and visual way, Fig. 6 shows the classification accuracy rate of each dataset over testing data. The test accuracy for ZRBCS is always higher than that for FRBCS for most datasets. The ZRBCS with certain degrees obtains the highest classification accuracy rate in 16 out of 20 datasets. And we can also see that the learning ability of the model increases as the value of K increases. The predictive ability of the model does not always increase, especially for data sets with more features. This is caused by the limited number of training patterns. For smaller data sets ($M < 10$), we suggest using a partition number ($K = 5$); in other cases, a partition number ($K = 3$) is recommended to achieve a better balance between accuracy and complexity.

To further demonstrate the superiority of the proposed method, we further compare the ZRBCS with three classical classification algorithms, including support vector machine(SVM) [8], kernel classifier [9] and cam weighted distance nearest neighbor Classifier (CamNN)[10]. Since two datasets Ionosphere and Segment cannot be trained and tested by CamNN, there are 18 datasets to be compared. The classification accuracy rates over testing data are shown in Table 11. In which, the classification accuracy rates of FRBCS and ZRBCS are taken from the optimal values in Table 8. The values in parentheses mark the ranks of the classifiers: The better the performance, the higher the rank (in case of ties, average ranks are assigned). The last row lists the average rank of each classifier. It is shown that the proposed ZRBCS is superior to other methods. Nonparametric tests based on average ranks are used to compare classifiers statistically. First, the Friedman test is carried out to determine whether the results obtained by different methods present significant differences. Table 12 shows the result of Friedman test with a level of significance $\alpha = 0.05$. It can be seen that the statistic value is greater than the critical value, which means there are significant differences among the results. After that the Nemenyi test is used to compare the proposed ZRBCS with other ones, and the results are shown in Fig. 7. It is shown that the proposed ZRBCS is not significantly better than the methods kernel, but is significantly better than the methods SVM and CamNN. The superiority of the ZRBCS over FRBCS is discussed in the Section 5.2 and Section 5.3.

(a) Banknote (b) Breast (c) Bupa (d) Diabetes

(e) Ecoli (f) Glass (g) Haberman (h) HTRU-2

(i) Ionosphere (j) Iris (k) Liver (l) Newthyroid

(m) Occupancy (n) Pageblocks (o) Pendigits (p) Pima

(q) Rice (r) Segment (s) Tranfusion (t) Vehicle

Figure 6: Classification accuracy rate (%) of FRBCS and ZRBCS over testing data. The symbol '$*$' denotes the FRBCS without certain degrees, '•' denotes the ZRBCS without certain degrees, '⧫' denotes the FRBCS with certain degrees, and '■' denotes the ZRBCS with certain degrees.

6. Conclusions

The Z-number is more suitable for describing uncertain and partially reliable information due to its model feature. In this paper, we extend the fuzzy if-then rules in the Z-number-valued framework to propose Z-number-valued if-then rules. Then based on the suggested form of the Z-number-valued if-then rule, a basic Z-number-valued rule-based classification system is established. The proposed system is based on the negative patterns of the fuzzy rule-based

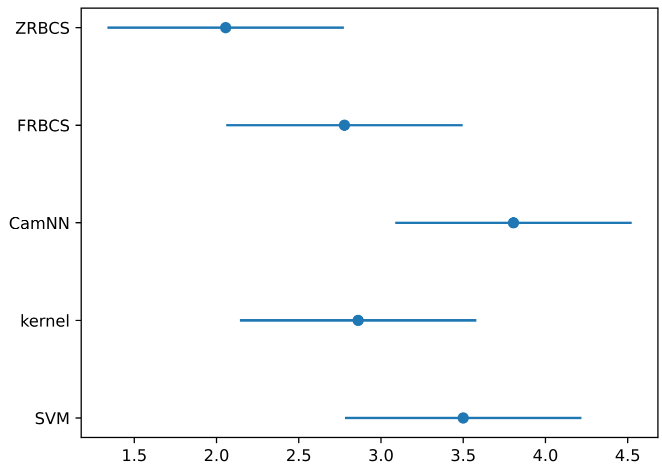


Figure 7: The Nemenyi test result with $\alpha = 0.05$.

system. The first fuzzy numbers of the antecedent Z-numbers are generated by the classical fuzzy rule extraction method, the fuzzy simple grid method. In the second fuzzy numbers of antecedent Z-numbers, information about negative patterns is stored, such as maximum, average, and minimum values. When classifying, the second fuzzy number can adjust the first fuzzy number based on negative patterns. Three experiments of 20 datasets have been conducted to demonstrate the learning and predictive abilities and running time of our classification system, and the results show that our method achieves better performance than the basic fuzzy rule-based system and two classical classification algorithms, SVM and CamNN.

Due to the proposed system in this paper just being a basic Z-number-valued rule-based system, it must be noted that the proposed ZRBCS, like FRBCS, cannot handle incomplete and unbalanced data. We can use some auxiliary means to further improve classification accuracy, interpretability and handle incomplete and unbalanced data, such as genetic algorithms and clustering algorithms, and to reduce the dimensionality based on this basic Z-number-valued rule-based system in further work.

Acknowledgements

This work was supported by the project B-TIC-590-UGR20 co-funded by the Programa Operativo FEDER 2014-2020 and the Regional Ministry of Economy, Knowledge, Enterprise and Universities (CECEU) of Andalusia, by the China Scholarship Council (CSC), by the project PID2019-103880RB-I00 funded by MCIN / AEI / 10.13039 /501100011033 and by the Andalusian goverment through project P20_00673.

References

[1] H. Ishibuchi, K. Nozaki, H. Tanaka, Distributed representation of fuzzy rules and its application to pattern classification, Fuzzy sets and systems 52 (1) (1992) 21–32.
[2] K. Siminski, An outlier-robust neuro-fuzzy system for classification and regression, International Journal of Applied Mathematics and Computer Science 31 (2).
[3] J. Niu, D. Chen, J. Li, H. Wang, A dynamic rule-based classification model via granular computing, Information Sciences 584 (2022) 325–341.
[4] C. Fu, W. Lu, W. Pedrycz, J. Yang, Rule-based granular classification: A hypersphere information granule-based method, Knowledge-Based Systems 194 (2020) 105500.
[5] Y. Farjami, K. Abdi, A genetic-fuzzy algorithm for spatio-temporal crime prediction, Journal of Ambient Intelligence and Humanized Computing (2021) 1–13.
[6] E. G. Mansoori, Frbc: A fuzzy rule-based clustering algorithm, IEEE transactions on fuzzy systems 19 (5) (2011) 960–971.

[7] L. A. Zadeh, A note on z-numbers, Information Sciences 181 (14) (2011) 2923–2932.
[8] C. Cortes, V. Vapnik, Support-vector networks, Machine learning 20 (3) (1995) 273–297.
[9] G. J. McLachlan, Discriminant analysis and statistical pattern recognition, John Wiley & Sons, 2005.
[10] C. Y. Zhou, Y. Q. Chen, Improving nearest neighbor classification with cam weighted distance, Pattern recognition 39 (4) (2006) 635–645.
[11] M. Woźniak, A. Zielonka, A. Sikora, Driving support by type-2 fuzzy logic control model, Expert Systems with Applications 207 (2022) 117798.
[12] M. Woźniak, A. Zielonka, A. Sikora, M. J. Piran, A. Alamri, 6g-enabled iot home environment control using fuzzy rules, IEEE Internet of Things Journal 8 (7) (2020) 5442–5452.
[13] H. Ishibuchi, K. Nozaki, H. Tanaka, Efficient fuzzy partition of pattern space for classification problems, Fuzzy Sets and Systems 59 (3) (1993) 295–304.
[14] H. Ishibuchi, K. Nozaki, N. Yamamoto, H. Tanaka, Selecting fuzzy if-then rules for classification problems using genetic algorithms, IEEE Transactions on fuzzy systems 3 (3) (1995) 260–270.
[15] D. Nauck, R. Kruse, A neuro-fuzzy method to learn fuzzy classification rules from data, Fuzzy sets and Systems 89 (3) (1997) 277–288.
[16] H. Ishibuchi, T. Nakaskima, Improving the performance of fuzzy classifier systems for pattern classification problems with continuous attributes, IEEE Transactions on industrial electronics 46 (6) (1999) 1057–1068.
[17] D. Chakraborty, N. R. Pal, A neuro-fuzzy scheme for simultaneous feature selection and fuzzy rule-based classification, IEEE Transactions on Neural Networks 15 (1) (2004) 110–123.
[18] Y. Chen, J. Z. Wang, Support vector learning for fuzzy rule-based classification systems, IEEE Transactions on fuzzy systems 11 (6) (2003) 716–728.
[19] H. Ishibuchi, T. Yamamoto, Rule weight specification in fuzzy rule-based classification systems, IEEE transactions on fuzzy systems 13 (4) (2005) 428–435.
[20] E. G. Mansoori, M. J. Zolghadri, S. D. Katebi, A weighting function for improving fuzzy classification systems performance, Fuzzy sets and systems 158 (5) (2007) 583–591.
[21] J. Alcalá-Fdez, R. Alcala, F. Herrera, A fuzzy association rule-based classification model for high-dimensional problems with genetic rule selection and lateral tuning, IEEE Transactions on Fuzzy systems 19 (5) (2011) 857–872.
[22] M. Elkano, M. Galar, J. Sanz, H. Bustince, Chi-bd: A fuzzy rule-based classification system for big data classification problems, Fuzzy Sets and Systems 348 (2018) 75–101.
[23] J. Niu, D. Chen, J. Li, H. Wang, Fuzzy rule-based classification method for incremental rule learning, IEEE Transactions on Fuzzy Systems 30 (9) (2021) 3748–3761.
[24] S. Mitra, Y. Hayashi, Neuro-fuzzy rule generation: survey in soft computing framework, IEEE transactions on neural networks 11 (3) (2000) 748–768.
[25] J. Sanz, M. Sesma-Sara, H. Bustince, A fuzzy association rule-based classifier for imbalanced classification problems, Information Sciences 577 (2021) 265–279.
[26] A. Azadeh, M. Saberi, N. Z. Atashbar, E. Chang, P. Pazhoheshfar, Z-ahp: A z-number extension of fuzzy analytical hierarchy process, in: 2013 7th IEEE International Conference on Digital Ecosystems and Technologies (DEST), IEEE, 2013, pp. 141–147.
[27] H. Aboutorab, M. Saberi, M. R. Asadabadi, O. Hussain, E. Chang, Zbwm: The z-number extension of best worst method and its application for supplier development, Expert Systems with Applications 107 (2018) 115–125.
[28] R. A. Krohling, A. G. Pacheco, G. A. dos Santos, Todim and topsis with z-numbers, Frontiers of Information Technology & Electronic Engineering 20 (2) (2019) 283–291.
[29] K.-w. Shen, J.-q. Wang, Z-vikor method based on a new comprehensive weighted distance measure of z-number and its application, IEEE Transactions on Fuzzy Systems 26 (6) (2018) 3232–3245.
[30] A. Garg, S. Das, J. Maiti, S. K. Pal, Granulized z-vikor model for failure mode and effect analysis, IEEE Transactions on Fuzzy Systems (2020) 1–1doi:10.1109/TFUZZ.2020.3037933.
[31] R. A. Aliev, B. G. Guirimov, O. H. Huseynov, R. R. Aliyev, Z-relation equation-based decision making, Expert Systems with Applications 184 (2021) 115387.
[32] R. Banerjee, S. K. Pal, On z-numbers and the machine-mind for natural language comprehension, in: Fifty Years of Fuzzy Logic and its Applications, Springer, 2015, pp. 415–457.
[33] R. Banerjee, S. K. Pal, Z*-numbers: Augmented z-numbers for machine-subjectivity representation, Information Sciences 323 (2015) 143–178.
[34] R. H. Abiyev, Dynamic system control using z-number based fuzzy neural network, in: International Conference on Intelligent and Fuzzy Systems, Springer, 2020, pp. 964–972.
[35] I. M. Jiskani, F. Yasli, S. Hosseini, A. U. Rehman, S. Uddin, Improved z-number based fuzzy fault tree approach to analyze health and safety risks in surface mines, Resources Policy 76 (2022) 102591.
[36] M. Abbaspour Onari, S. Yousefi, M. Jahangoshai Rezaee, Risk assessment in discrete production processes considering uncertainty and reliability: Z-number multi-stage fuzzy cognitive map with fuzzy learning algorithm, Artificial Intelligence Review 54 (2) (2021) 1349–1383.
[37] D. Wu, X. Liu, F. Xue, H. Zheng, Y. Shou, W. Jiang, A new medical diagnosis method based on z-numbers, Applied Intelligence 48 (4) (2018) 854–867.
[38] R. Yazdanparast, R. Tavakkoli-Moghaddam, R. Heidari, L. Aliabadi, A hybrid z-number data envelopment analysis and neural network for assessment of supply chain resilience: a case study, Central European Journal of Operations Research 29 (2) (2021) 611–631.
[39] Z. Bobar, D. Božanić, K. Djurić, D. Pamučar, Ranking and assessment of the efficiency of social media using the fuzzy ahp-z number model-fuzzy mabac, Acta Polytechnica Hungarica 17 (3) (2020) 43–70.
[40] R. A. Aliev, W. Pedrycz, B. Guirimov, O. H. Huseynov, Clustering method for production of z-number based if-then rules, Information Sciences 520 (2020) 155–176.
[41] R. A. Aliev, A. V. Alizadeh, O. H. Huseynov, The arithmetic of discrete z-numbers, Information Sciences 290 (2015) 134–155.

[42] R. A. Aliev, O. H. Huseynov, L. M. Zeinalova, The arithmetic of continuous z-numbers, Information Sciences 373 (2016) 441–460.
[43] R. A. Aliev, A. V. Alizadeh, O. H. Huseynov, K. Jabbarova, Z-number-based linear programming, International Journal of Intelligent Systems 30 (5) (2015) 563–589.
[44] R. Aliev, Operations on z-numbers with acceptable degree of specificity, Procedia computer science 120 (2017) 9–15.
[45] R. A. Aliev, W. Pedrycz, O. H. Huseynov, S. Z. Eyupoglu, Approximate reasoning on a basis ofz-number-valued if–then rules, IEEE Transactions on Fuzzy Systems 25 (6) (2017) 1589–1600.
[46] B. Kang, Y. Deng, R. Sadiq, Total utility of z-number, Applied Intelligence 48 (3) (2018) 703–729.
[47] M. Mazandarani, Y. Zhao, Z-differential equations, IEEE Transactions on Fuzzy Systems 28 (3) (2019) 462–473.
[48] J. Alcalá-Fdez, L. Sanchez, S. Garcia, M. J. del Jesus, S. Ventura, J. M. Garrell, J. Otero, C. Romero, J. Bacardit, V. M. Rivas, et al., Keel: a software tool to assess evolutionary algorithms for data mining problems, Soft Computing 13 (3) (2009) 307–318.