Criteria for Fuzzy Rule-based Systems and its Applicability on Examples

Erika Mináriková[1]

[1] University of Economics in Bratislava, Faculty of Economic Informatics, Dolnozemskácesta 1, Bratislava, 85235 (Slovak Republic)

erika.minarikova@euba.sk

https://doi.org/10.53465/EDAMBA.2021.9788022549301.327-337

Abstract. Classification allows us to handle the large amount of data that is available nowadays. In our work, we use the classification features to divide employees into the several classes and examine the differences between the classical and flexible classification. We also emphasize the advantages of classical classification as well as the disadvantages, and how we can solve them by fuzzy logic. Fuzzy rule-based systems are explainable and therefore interpretable because the rules are defined by linguistic variables. Design of a more complex system is a tedious task. To resolve this, we examine interpretability criteria for fuzzy rule-based systems. We examine this topic on the examples with two classification attributes because it is easily illustrated graphically. To use more attributes is mathematically possible, but it is harder to visualize for users in a three and more dimensional spaces. In our work, we propose how to create an explainable design for classification and propose possibilities how to expand it.

Keywords: Fuzzy logic, Rule-based systems, Classification, Explainability.

JEL classification: *C 4, D 8, C 9*

1 Introduction

Classification splits large amounts of data into several predefined classes. It is used in many industries such as biology, medicine, geography, as well as in business, where it has found a great advantage in categorizing individual data, such as grouping products, customers, and employees [4]. Today, we recognize many classification methods, e.g., rule-based systems, fuzzy classification systems, Naive Bayes, and machine learning methods, which include neural networks and logistic regression.

Companies and institutions currently dispose with large amount of data and information. In order to use this data effectively, we need to handle them correctly.

Rule-based systems classify data according to the defined rules provided by domain expert. Thanks to the users input, classification models are explainable and therefore easily interpretable, but with a more complex model, it is difficult to define consistent rules and input parameters. Methods based on learning procedures from the data have proven their efficiency, but for modeling the correct design, it is necessary to have a sufficient amount of data for learning and validation [13] as well as criteria to evaluate interpretability [10].

Explainability is the crucial factor in many systems, especially in the medical sector, but also in the economy, or in everyday life. Different classification systems help us to make decisions. They are an increasing part of our lives and therefore, it is very important to trust their outputs.

In our work, we examine important rules, summarized in [10], which should be considered during creating explainable and reliable rule-based systems, and propose how to deal with such issues.

The remainder of the paper is organized as follows. Section 2 briefly describes classification methods regarding explainability. Section 3 shows explainability and interpretability issues which should be considered. Section 4 is devoted to experiments. Section 5 discusses obtain results and the implications for the future research. Finally, Section 6 concludes the paper.

2 Classification Methods

Today, many algorithms focused on resolving classification problems exist. But the question stays, how we design a trustworthy system. In this section, we will shortly discuss features of rule-based systems and fuzzy rule-based systems.

Rule-based systems as well as fuzzy rule-based systems use IF-THEN rules to define a classification model [7]. The main difference is that crisp classification consists of precise values and sharp rules, whereas fuzzy classification uses fuzzy sets and fuzzy logic.

Crisp classification requires rules such:

- IF $x \leq Q$ and $y \geq P$ THAN good performance (G)
- IF $x \geq Q$ and $y \geq P$ THAN medium performance (M)
- IF $x \leq Q$ and $y \leq P$ THAN medium performance (M)
- IF $x \geq Q$ and $y \leq P$ THAN bad performance (B)

where *P* and *Q* are values of variables *x* and *y*, respectively. We can see this model on the Fig. 1.

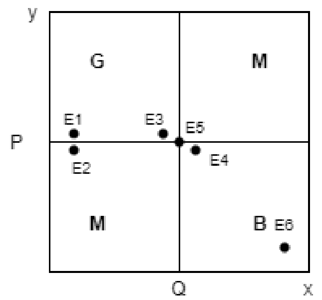

Fig. 1. Classification into three classes by four crisps rules.
Source: Own processing.

The problem of crisps rules can be seen on entities near the class's borders [8]. The entities *E1* and *E2* have similar entry values, but they are treated differently. The same applies for the entities *E3* and *E4*. On the other hand, the entity *E4* has significantly better performance than the entity E6 and yet they belong to the same class.

As we explained, the classical classification might not guarantee us a fair classification of entities. One of the possible solutions is to divide the classification space into several additional classes, which would increase the complexity and reduce the transparency of the classification [9].

Fuzzy rule-based systems allow entity to belong to more than one fuzzy class. If corresponding values are similar for two entities, their membership functions are similar too. The fuzzy rules for classification into three classes with two attributes are as follows (see Fig. 2):

- IF x is high and y is low THAN bad (B),
- IF x is high and y is low THAN medium (M),
- IF x is low and y is high THAN medium (M),
- IF x is low and y is high THAN good (G).

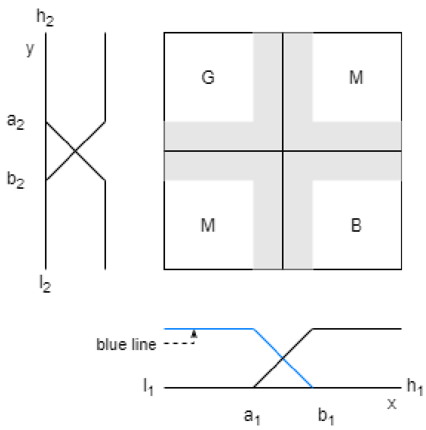

Fig. 2. Classification into three classes by four fuzzy rules.
Source: Adapted from [9].

Fuzzy rule-based systems better represent experts' requirements and the classification is fairer with the same number of rules. The possibility of using fewer rules is because the classes overlap, elements do not only belong to the set, but belong to the set with a certain membership degree [8]. There are no strict boundaries between classes, i.e., fuzzy logic ensures a smooth transition.

For the simplicity reason (which do not affect generality), the examples consider the afore-mentioned classification into three classes defined by two attributes.

Systems using rule-based classification achieve explainable and therefore easy interpretable results [7] [16]. Rule-based systems are interpretable, but there is a problem with constructing consistent rule-based systems and when required application in different areas for the same task. For instance, the afore mentioned rule base is self -explanatory, but the meaning of attributes differs among departments for the same product. For instance, selling air-condition equipment in Rome and Reykjavik.

In the next section, we describe possible problems which can occur and disturb the explainability. We show it on the examples of evaluating performance of employees.

3 Explainability in Fuzzy Classification Systems

This section demonstrates several key problems, which might appear during the construction of the classification space. To tackle this problem, Alonso et al. [10] have summarized the interpretability criteria.

Criteria for fuzzy sets include normality, continuity, and convexity. On the level of linguistic variables and fuzzy partitions, constrains are justifiable number of elements, distinguishability, and relation preservation among others. On the fuzzy rules level, criteria are description length and granular outputs. Finally, on the fuzzy rule bases level, criteria are consistency, average firing rules, completeness.

When a rule-based system is growing, these criteria become more relevant. In the next section, we look closer at the constraints and criteria.

3.1 Definition of Classification Space

The design of classification space should ensure normality requirements [10]. It means that at least one element should have full membership to the fuzzy set. When we define class, we should also identify element which represent that class as a prototype. Then, we compare real values with the prototype and classify accordingly.

When we design classification space, we also have to have in mind that each element should be represented at least by one fuzzy set. This is crucial, how we define the space and the boundaries. This is also connected with definition of leftmost and rightmost fuzzy sets, which represent the limit values of a classification space.

In Fig. 2 the leftmost fuzzy set for the attribute *x* is marked with blue. This fuzzy set represent the low values with degree equal to 1, which we should define as a prototype. Otherwise, it is not interpretable. The same approach applies for all defined fuzzy sets.

When we define rule-based systems, expert give us the rules at the beginning. For example: If the performance is good (higher then 600) than seller get the highest reward

(100). We see in the Fig. 1, that *E5* is partially in the class good performance (0.5) so seller should get reward 50. As we can see, expert besides of the rules should also define the parameters. With a higher number of classes and classification attributes, it can be a tedious task. With high complexity of systems, it also becomes harder to maintain. In our classification space, the parameters which should be defined are marked as a_1, b_1, a_2, b_2 for input attributes, three parameters for output classes as well as the lowest values (l_1, l_2) and highest values (h_1, h_2) in the Fig. 2.

3.2 Classification Classes

Membership functions and classification algorithm should have continuous effect [10]. For example, if the employee is better, we evaluate him higher, or if the customer is more loyal then receives a higher discount.

The classification classes should be convex, i.e., we can identify how far is the item from the ideal value. It is difficult to label the non-convex set with linguistic term, which is key to explainability. In our example, we define several linguistic terms within one attribute: low - high, short - long, good - bad. In the Fig. 2, we can see two fuzzy sets which define attribute *x* and two fuzzy sets which define attribute *y*, and the resultis union of these fuzzy sets.

The classes should be explicitly defined and ensure that the understanding between users is the same:

- All users agree on the implicit comparison of terms (bad < medium < good).
- Same width to all fuzzy sets – Differences might lead to improper classification (see Fig. 3).
- The sets should be well distinguishable.
- To ensure complementarity - sum of membership degree should be equal 1 [12]. The classified item can be part of more than one fuzzy set. In Fig. 1, we can see that *E5* partially belongs to all classes. In this case, membership degree has to be 0.25 foreach class. Otherwise, it would violate the continuity and the classification would be less fair.

The proof of this criteria can be seen on the next example:

In Fig. 3 are defined three fuzzy sets: low (L) < medium (M) < high (H). The reward in class L is 0 in class M is 50 and in class H is 100. We compare the results of values a_1, a_2, a_3 for the considered entities. The expected result is $a_1 < a_2 < a_3$.

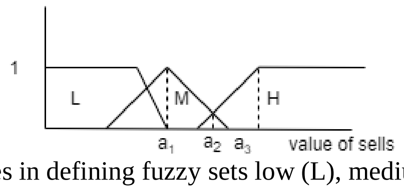

Fig. 3. Inconsistencies in defining fuzzy sets low (L), medium (M), high (H).
Source: Own processing.

- a_1 belong to classes (L, M, H) with membership degrees (0; 1; 0)

- a_2 belong to classes (L, M, H) with membership degrees (0; 0,25; 0,25)
- a_3 belong to classes (L, M, H) with membership degrees (0; 0; 1)

Rewards:
- $a_1 = 0*0 + 1*50 + 0*100 = 50$
- $a_2 = 0*0 + 0,25*50 + 0,25*100 = 37,5$
- $a_3 = 0*0 + 0*50 + 1*100 = 100$

The sum of membership degree is $0,5 \neq 1$ which cause the problem of unfair classification. As we can see, that the expected result and the actual result does not match: $a_1 < a_2 < a_3 \neq a_1 > a_2 < a_3.$

3.3 A Note to Using Software for Creating Classification Model

Software helps us to create classification models. For example, MatLab is a programming platform providing possibility to analyze and design systems and products. MatLab is the computing environment for engineers and scientists, but it is not often used in companies.

Commonly used software is MS Excel. Microsoft Excel is a part of the Microsoft Office tolls. It is a spreadsheet program. Mostly used to create tabular forms, create specifying calculations, or for further graphical processing. It is a very useful and frequently used tool in many areas.

Another possibility of creating classification model is by using the Python programing language. In contrast to MatLab, Python is a general-purpose programming language. It is universal, suitable for creating applications for data analysis. Among other things, Python provides us with the advantages of fast processing as well as large volumes of data and simple programming syntax [11].

Using software provide us possibility to check syntax side of the classification problem but not the semantic. What we code to the program, it will compile that way. It is why we should be very careful with creating a reliable design.

4 Examples on Data

In this section, we introduce two examples, where the criteria from Section 3 are considered.

4.1 Evaluation of Employees – Applicability of Different Parameters

A hypothetical organization has departments in different part of the word where managers evaluate workers considering local specifies of regions where sellers operate by a universally accepted model. Managers wish to provide bonuses for sellers by two attributes: turnover and persuasion time.

The rule base is as follows:
- IF turnover is low AND persuasion time is high THEN reward is low.
- IF turnover is low AND persuasion time is low THEN reward is medium.

- IF turnover is high AND persuasion time is high THEN reward is medium.
- IF turnover is high AND persuasion time is low THEN reward is high.

The number of rules and linguistic variables is low, creating the base for an interpretable rule-based system and transferable to other departments. Generally, Takagi-Sugeno-Kang (TSK) rule-bases systems are less interpretable than Mamdani rule-based systems, due to linguistic interpretation of the output attribute. But, when we create a zero-ordered TSK, it become interpretable for domain experts.
We transform rules to the structure:

- IF turnover is low AND persuasion time is high THEN reward is 0.
- IF turnover is low AND persuasion time is low THEN reward is p.
- IF turnover is high AND persuasion time is high THEN reward is p.
- IF turnover is high AND persuasion time is low THEN reward is m.

In this structure 0, *p* and *m* (*m* > *p*) are, e.g., money units. For instance, when *m* = 50 and *p* = 100, the fourth rule is activated with 0.6 and third rule with 0.4, the reward is between 50 and 100. It is a flexible solution and easily applicable in any data processing tool like broadly used MS Excel in offices.

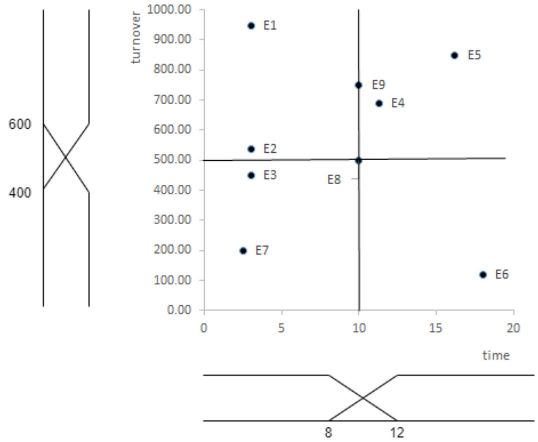

Fig. 4. Classification space for evaluating the employees.
Source: Own processing.

When the requirements of normality, convexity, relation preservation, justifiable number of linguistic terms, distinguishability, etc. is met, the rule-base is as depicted in Fig. 4.

Parameters of fuzzy sets in Fig. 4 can be assigned by managers in each department or mined from the data. In the latter, we can e.g., adopt uniformly divided domains and calculate parameters by the method proposed in [15].

The motivation is a key aspect in improving performances [6]. In our case, motivation should be based on sellers' results and the environment in which seller operates.

Table 1. Results for 1. department.

ID	TIME	TURNOVER	μG	μM1	μM2	μB	REWARD
E1	3	947.00	1	0	0	0	100
E2	3	537.00	0.685	0	0.315	0	84.25
E3	3	450.00	0.25	0	0.75	0	62.5
E4	11.3	689.00	0.175	0.825	0	0	58.75
E5	16.2	850.00	0	1	0	0	50
E6	18	120.00	0	0	0	1	0
E7	2.5	200.00	0	0	1	0	50
E8	10	500.00	0.25	0.25	0.25	0.25	50
E9	10	750.00	0.5	0.5	0	0	75

Source: Own processing.

For one department parameters for turnover might be 0, 400, 600, 1000 (see Fig. 4), whereas for another department are 0, 700, 800, 1200, i.e., selling air-condition is less demanding in, for instance, Rome than in Reykjavik and moreover, the population in Reykjavik is significantly lower. In business applications, mixture of constructing rule - based system from experts and adjusting to data in diverse regions is the option which should be considered. Anyway, the subset of the interpretability criteria (relevant for this task) should be met.

In the Table 1, we can see those values of both attributes of *E8* are in the middle of boundaries (400, 600) and the reward is the medium (50). *E9* gains higher reward because the value of *y* attribute is also higher. In the Table, the parameters for attribute turnover are different (i.e., (700, 800)) and therefore *E9* having the values 750 is in the middle of the defined boundaries. It causes that the reward is now medium (50), whereas *E8* gains lower reward. The persuasion time stays the same for both departments also the boundaries, so it does not affect the results.

Table 2. Results for 2. department.

ID	TIME	TURNOVER	μG	μM1	μM2	μB	REWARD
E1	3	947.00	1	0	0	0	100
E2	3	537.00	0	0	1	0	50
E3	3	450.00	0	0	1	0	50
E4	2.5	200.00	0	0	0.175	0.825	8.75
E5	16.2	850.00	0	1	0	0	50
E6	9.2	865.00	0	0	0	1	0
E7	11.3	689.00	0	0	1	0	50
E8	10	500.00	0	0	0.5	0.5	25
E9	10	750.00	0.25	0.25	0.25	0.25	50

Source: Own processing.

4.2 Evaluation of Employees – Easily Interpretable Model

In the example of evaluation of employees, our main purpose is to create easily interpretable model for employer/manager as well as for employee. The model can be extended for more attributes, but we could not represent results graphically.Two-dimensional space is easily readable and give us immediate insight into the results.For example, in the Fig. 4, we see that employee *E2* should increase the value of turnover. Contrary, *E9* has to decrease the persuasion time, whereas *E6*, and *E8* should improve in both classification attributes.

When using more attributes not only that we cannot easily visualize the results but also, we should consider possible correlations and so-called coalitions among atomic conditions. For example, the simultaneous occurrence of attributes A, B, C is less significant than the occurrence of A, G. Such a situation is captured by the Choquet integral [1], which also becomes an object of interest for an explainable classification.

5 Discussion

In this work, we examined interpretability criteria for rule-based systems and illustrated on an example related to selling the same item in different regions.

To compute the interpretability of a model is a hard task, as the definition of interpretability cannot be formulated in strict mathematical sense. It also involves the human factor, which is hard to formalize [10]. For instance, in our example the criterion of unimodality is not relevant, or in some other applications it might be less important than the other. In this work, we evaluated a subset of explainability criteria relevant for our examples. The other criteria are examined in [10].

An interpretable rule-based model, like the model for reward explained in this work, can be used among departments. The only adoption is in adjusting parameters to the environments in which sellers are operating. The same holds for using the rule-based model in different time frame, where economic growth and crisis (like the current pandemic situation) appears.

An oversimplified option is to have a list of the interpretability criteria and mark the filled ones. A more reliable option is aggregating atomic criteria. But, it is not an easy task. In a conjunctive aggregation if a single criterion is not met, the interpretability degree is equal to 0. Contrary, in a disjunctive aggregation a single met criterion ensures the full interpretability. Hence, an option is quantified aggregation of atomic attributes[14], where interpretability increases as the number of met criteria increases. On the other hand, in various tasks several criteria might be mandatory, whereas other optional, which leads us to the asymmetric conjunction proposed in [2] and fully axiomatized in[5]. The future tasks should evolve around aggregating elementary interpretability criteria summarized in [10] and for additional ones like dominance of rules.

6 Conclusion

We examined explainability in fuzzy rule-based systems. These systems are generally explainable, but with an increased complexity, inconsistencies might occur.

In example we discussed criteria which we should have in mind during creating a classification space in order to develop a reliable design. Fuzzy rule-based systems bring us a fairer classification, but it should be consistent.

We can use our results wherever we want to evaluate similar entities similarly. This means, that the boundaries of sets are not expressed by an exact number, like" about, around, much, little, etc." Therefore, we replace sharp set with a fuzzy set, which describes the statement more realistically, leading us to the fuzzy classification. Management applications and databases from individual departments and divisions within the company provide us with quantitative as well as qualitative performance indicators, from which we can obtain the data and subsequently create an application of fuzzy classification. Such application provides various analyzes for the company [8]. By using conceptually described models, we can easily solve practical examples. The model can be reused by adaptation to changing data and parameters over time and in different areas.

The explainability and interpretability are important features for the systems, whenever we want to trust the output. In our examples, we used two classification attributes to make the output easy to read and also easy to visualize graphically. In practice, a large number of attributes can be used, but to meet all criteria is a more demanding task.

Acknowledgement

This work is supported by a project KEGA No. 025EU-4/2021 entitled "Knowledge discovery from data for business practice" by the Ministry of Education, Science, Research and Sport of the Slovak Republic.

References

1. Beliakov, G., Calvo, T.: Construction of aggregation operators with noble reinforcement. In. *IEEE transactions on fuzzy systems*, 15(6), 1209-1218 (2007).
2. Bosc P., Pivert O.: On four noncommutative fuzzy connectives and their axiomati- zation, Fuzzy Sets and Systems, 202, 42–60 (2012).
3. Dubois, D., Prode, H.: On the use of aggregation operations in information fusion processes. In: Fuzzy sets and systems 142(1), 143-161 (2004).
4. Dujmovic, J.: Soft Computing Evaluation Logic: The LSP Decision Method and Its Applications. John Wiley & Sons, (2018).
5. Hudec M., Mesiar R.: The axiomatization of asymmetric disjunction and conjunction. Information Fusion, 2020, vol. 53, 165-173 (2020).
6. Hudec M., Torres Van Grinsven V.: Business' participants motivation in official surveys by fuzzy logic. In 1st Eurasian Multidisciplinary Forum, (EMF 2013). Tbilisi, 2013, 24 – 26 October, pp. 42-52 (2013).
7. Hudec, M.: Fuzziness in information systems. Springer International Publishing, (2016).
8. Meier, H. Werro, N.: A fuzzy classification model for online customers. In: Informatica

31(2), (2007).
9. Mináriková, E.: Data Classification in MS Excel [bachelor thesis]. Bratislava: Ekonomická univerzita v Bratislave [s. n.], 44s. (2019).
10. Moral, J. M. A., Castiello, C., Magdalena, L., & Mencar, C.: Explainable Fuzzy Systems: Paving the way from Interpretable Fuzzy Systems to Explainable AI Systems. Springer. Swizerland (2021).
11. Phillips, D.: Python 3 object-oriented programming: build robust and maintainable software with object-oriented design patterns in Python 3.8. Packt Publishing Ltd, (2018).
12. Ruspini, E. H.: A new approach to clustering. Information and control, 15(1), 22-32 (1969).
13. Singh, S., Ribeiro, M., Guestrin, C.: Programs as black–box explanations, in: NIPS 2016 Workshop on Interpretable Machine Learning in Complex Systems, 1–5. Barcelona (2016).
14. Sojka P., Hudec M., Švaňa M.: Linguistic summaries in evaluating elementary conditions, summarizing data and managing nested queries. Informatica, 31(4), pp. 841-856 (2020).
15. Tudorie, C.: Qualifying objects in classical relational database querying. In: Galindo, J. (Ed.), Handbook of Research on Fuzzy Information Processing in Databases. Information Science Reference, Hershey, pp. 218–245 (2008).
16. Vučetić, M., Hudec. M., Božilović, B.: Fuzzy functional dependencies and linguistic interpretations employed in knowledge discovery tasks from relational databases. In: Engineering Applications of Artificial Intelligence, (2020), 88: 103395.