UNIVERSIDAD DE OVIEDO

PROGRAMA DE DOCTORADO EN INGENIERÍA INFORMÁTICA

DESIGN AND ANALYSIS OF FUZZY SYSTEMS SUPPORTED BY SOCIAL NETWORK ANALYSIS TECHNIQUES

David Pérez Pancho

Directores:
Jose María Alonso Moral
Luis Magdalena Layos

UNIVERSIDAD DE OVIEDO

PROGRAMA DE DOCTORADO EN INGENIERÍA INFORMÁTICA

DESIGN AND ANALYSIS OF FUZZY SYSTEMS SUPPORTED BY SOCIAL NETWORK ANALYSIS TECHNIQUES

David Pérez Pancho

Directores:

Jose María Alonso Moral

Luis Magdalena Layos

August 2015

RESUMEN DEL CONTENIDO DE TESIS DOCTORAL

1.- Título de la Tesis	
Español/Otro Idioma: Diseño y análisis de sistemas fuzzy apoyados en técnicas de análisis de redes sociales	Inglés: Design and analysis of fuzzy systems supported by social network analysis techniques

2.- Autor	
Nombre: David Pérez Pancho	
Programa de Doctorado: Ingeniería Informática	
Órgano responsable: Comisión Académica del Programa de Doctorado	

RESUMEN (en español)

Esta tesis doctoral propone la creación y desarrollo de una nueva metodología que permita comprender el funcionamiento de sistemas fuzzy basándose en Fuzzy Inference-grams, o Fingrams.

Los Fingrams son representaciones gráficas de sistemas de reglas fuzzy que muestran la interacción entre reglas a nivel de inferencia. Esta interacción se presenta a nivel de co-disparo entre reglas, esto es, reglas disparadas por una misma entrada. Los Fingrams muestran el mecanismo de inferencia de sistemas de reglas fuzzy tanto desde un punto de vista global, esto es, observando como todas las reglas cubren un conjunto de datos dado, como desde un punto de vista local, es decir, como una instancia es cubierta por el conjunto de reglas. Por tanto, podemos analizar un sistema en detalle, permitiendo su mejora con conocimiento experto, estudiando regla a regla e instancia a instancia como este se comporta.

Aún más, los Fingrams son una herramienta de análisis efectiva y eficiente en varias aplicaciones tanto para el diseño como para la mejora de sistemas fuzzy. La mejora de sistemas fuzzy puede ser realizada de forma sistemática tras analizar los gráficos manualmente o apoyado en técnicas de análisis de redes sociales (como la detección de comunidades) e índices de calidad (como centralidad o page rank). El análisis de Fingrams ofrece muchas posibilidades: medir la comprensibilidad de sistemas fuzzy, detectar redundancias y/o inconsistencias entre reglas, descubrir y analizar instancias no cubiertas por el sistema, identificar las reglas más significativas, etcétera.

La nueva metodología ha sido probada y validada sobre reglas de asociación, clasificadores y regresores de reglas fuzzy. La utilidad de los Fingrams sobre reglas de asociación fuzzy ha sido ilustrada en un problema real en el que se estima las valoraciones cualitativas de distintas muestras de diseño industrial. El algoritmo FURIA ha sido utilizado sobre un conjunto de datos real para mostrar las posibilidades de los Fingrams en un caso de clasificador basado en reglas fuzzy. Y hemos seleccionado un problema de distribución de la red eléctrica para presentar el potencial de los Fingrams en un contexto de sistemas de reglas de regresión.

Finalmente, cabe destacar que la metodología ha sido integrada en distintas herramientas software gracias a implementaciones específicas realizadas durante el período doctoral. La herramienta de modelado fuzzy GUAJE, y las suites software para minería de datos KEEL y KNIME han sido dotadas de módulos para la creación y análisis de Fingrams.

RESUMEN (en Inglés)

This doctoral dissertation proposes the creation and development of a new methodology for fuzzy system comprehensibility analysis based on fuzzy systems' inference maps, so-called fuzzy inference-grams, Fingrams in short.

Fingrams show graphically fuzzy rule-based systems, presenting the interaction between rules at the inference level in terms of co-fired rules, i.e., rules fired at the same time by a given input. Fingrams depict the inference mechanism of fuzzy rule-based systems from a global view point, i.e. observing how all the rules covered the complete given dataset, and from a local view point, i.e. illustrating a partial view of the system when focusing on those rules that participate in the inference process regarding a single instance, by the so-called instance-based Fingrams. In consequence, we can analyze the system in detail, and even improve it with expert knowledge, carefully checking rule by rule and instance by instance.

Even more, Fingrams are likely to act as an effective and efficient tool in several applications regarding both design and refinement of fuzzy systems. The human centric improvement of a fuzzy rule-based system could be done after analyzing the resulting graphs manually or assisted by well-known social network analysis techniques (such as community mining) and quality indexes (such as centrality, page rank and so on). The analysis of Fingrams offers many possibilities: measuring the comprehensibility of fuzzy systems, detecting redundancies and/or inconsistencies among fuzzy rules, finding out and analyzing instances not covered, identifying the most significant rules, and so forth.

The new methodology has been tested and validated for fuzzy association rules, fuzzy rule-based classifiers and regressors. The utility of Fingrams over fuzzy association rules was illustrated in a real-world problem dealing with qualitative assessment of industrial objects designed through cognitive engineering. FURIA algorithm was used over a real dataset to show the possibilities of Fingrams in fuzzy rule-based classifiers. And, we selected an electrical network distribution problem to present the potentials of Fingrams in the context of fuzzy rule-based regressors.

Finally, it is worthy to note that Fingrams are fully integrated in different software tools thanks to the specific software implemented during the thesis period. The fuzzy modeling toolbox GUAJE, and the software suites for data mining KEEL and KNIME have been enhanced allowing the creation and analysis of Fingrams.

SR. DIRECTOR DE DEPARTAMENTO DE INFORMÁTICA/ SR. PRESIDENTE DE LA COMISIÓN ACADÉMICA DEL PROGRAMA DE DOCTORADO EN INGENIERÍA INFORMÁTICA

Abstract

This doctoral dissertation proposes the creation and development of a new methodology for fuzzy system comprehensibility analysis based on fuzzy systems' inference maps, so-called fuzzy inference-grams, Fingrams in short.

Fingrams show graphically fuzzy rule-based systems, presenting the interaction between rules at the inference level in terms of co-fired rules, i.e., rules fired at the same time by a given input. Fingrams depict the inference mechanism of fuzzy rule-based systems from a global view point, i.e. observing how all the rules covered the complete given dataset, and from a local view point, i.e. illustrating a partial view of the system when focusing on those rules that participate in the inference process regarding a single instance, by the so-called *instance-based Fingrams*. In consequence, we can analyze the system in detail, and even improve it with expert knowledge, carefully checking rule by rule and instance by instance.

Even more, Fingrams are likely to act as an effective and efficient tool in several applications regarding both design and refinement of fuzzy systems. The human centric improvement of a fuzzy rule-based system could be done after analyzing the resulting graphs manually or assisted by well-known social network analysis techniques (such as community mining) and quality indexes (such as centrality, page rank and so on). The analysis of Fingrams offers many possibilities: measuring the comprehensibility of fuzzy systems, detecting redundancies and/or inconsistencies among fuzzy rules, finding out and analyzing instances not covered, identifying the most significant rules, and so forth.

The new methodology has been tested and validated for fuzzy association rules, fuzzy rule-based classifiers and regressors. The utility of Fingrams over fuzzy association rules was illustrated in a real-world problem dealing with qualitative assessment of industrial objects designed through cognitive engineering. FURIA algorithm was used over a real dataset to show the possibilities of Fingrams in fuzzy rule-based classifiers. And, we selected an electrical network distribution problem to present the potential of Fingrams in the context of fuzzy rule-based regressors.

Finally, it is worthy to note that Fingrams are fully integrated in different software tools thanks to the specific software implemented during the thesis period. The fuzzy modeling toolbox GUAJE, and the software suites for data mining KEEL and KNIME have been enhanced allowing the creation and analysis of Fingrams.

Resumen

Esta tesis doctoral propone la creación y desarrollo de una nueva metodología que permita comprender el funcionamiento de sistemas fuzzy basándose en Fuzzy Inference-grams, o *Fingrams*.

Los Fingrams son representaciones gráficas de sistemas de reglas fuzzy que muestran la interacción entre reglas a nivel de inferencia. Esta interacción se presenta a nivel de co-disparo entre reglas, esto es, reglas disparadas por una misma entrada. Los Fingrams muestran el mecanismo de inferencia de sistemas de reglas fuzzy tanto desde un punto de vista global, esto es, observando como todas las reglas cubren un conjunto de datos dado, como desde un punto de vista local, es decir, como una instancia es cubierta por el conjunto de reglas. Por tanto, podemos analizar un sistema en detalle, permitiendo su mejora con conocimiento experto, estudiando regla a regla e instancia a instancia como este se comporta.

Aún más, los Fingrams son una herramienta de análisis efectiva y eficiente en varias aplicaciones tanto para el diseño como para la mejora de sistemas fuzzy. La mejora de sistemas fuzzy puede ser realizada de forma sistemática tras analizar los gráficos manualmente o apoyado en técnicas de análisis de redes sociales (como la detección de comunidades) e índices de calidad (como centralidad o page rank). El análisis de Fingrams ofrece muchas posibilidades: medir la comprensibilidad de sistemas fuzzy, detectar redundancias y/o inconsistencias entre reglas, descubrir y analizar instancias no cubiertas por el sistema, identificar las reglas más significativas, etcétera.

La nueva metodología ha sido probada y validada sobre reglas de asociación, clasificadores y regresores de reglas fuzzy. La utilidad de los Fingrams sobre reglas de asociación fuzzy ha sido ilustrada en un problema real en el que se estima las valoraciones cualitativas de distintas muestras de diseño industrial. El algoritmo FURIA ha sido utilizado sobre un conjunto de datos real para mostrar las posibilidades de los Fingrams en un caso de clasificador basado en reglas fuzzy. Y hemos seleccionado un problema de distribución de la red eléctrica para presentar el potencial de los Fingrams en un contexto de sistemas de reglas de regresión.

Finalmente, cabe destacar que la metodología ha sido integrada en distintas herramientas software gracias a implementaciones específicas realizadas durante el período doctoral. La herramienta de modelado fuzzy GUAJE, y las suites software para minería de datos KEEL y KNIME han sido dotadas de módulos para la creación y análisis de Fingrams.

Contents

Abstract v

Resumen vii

1 Introduction 1
- 1.1 The History of Interpretable Fuzzy Systems . . . 2
- 1.2 State of the Art on Visual Description and Analysis of Fuzzy Systems . . . 5
- 1.3 A Brief Review on Social Network Analysis . . . 7

2 Objectives 13

3 Discussion of results 15
- 3.1 Methodology . . . 15
  - 3.1.1 Fingrams Generation . . . 16
  - 3.1.2 Fingrams Scaling . . . 18
  - 3.1.3 Fingrams Drawing . . . 19
  - 3.1.4 Types of Fingrams . . . 22
    - 3.1.4.1 Classification Fingrams . . . 22
    - 3.1.4.2 Regression Fingrams . . . 23
    - 3.1.4.3 Fuzzy Association Rule Fingrams . . . 24
    - 3.1.4.4 Instance-based Fingrams . . . 25
- 3.2 Cases of use . . . 27
  - 3.2.1 Case of use on fuzzy classifiers . . . 28
  - 3.2.2 Case of use on fuzzy regressors . . . 37
  - 3.2.3 Case of use on fuzzy association rules . . . 41
- 3.3 Software Implementations . . . 45
  - 3.3.1 Fingrams Generator . . . 45
  - 3.3.2 GUAJE . . . 50
  - 3.3.3 KEEL . . . 53
  - 3.3.4 KNIME . . . 54

4 Conclusions and future work 59

Conclusiones y trabajo futuro **61**

A Publications **63**

A.1 Compilation of publications . . . 63

A.1.1 First publication: Methodology . . . 64

A.1.2 JCR of IEEE Transactions on Fuzzy Systems . . . 82

A.1.3 Second publication: Software . . . 83

A.1.4 JCR of International Journal of Computational Intelligence Systems 100

A.1.5 Third publication: Case of use . . . 101

A.1.6 JCR of Fuzzy Sets and Systems . . . 144

A.2 Aditional publications . . . 145

A.2.1 Book chapters . . . 145

A.2.2 Conference proceedings . . . 145

Bibliography **147**

List of Figures

1.1 Publications per year related to interpretability issues. 2
1.2 Scientogram with the main thematic areas where works dealing with interpretability of fuzzy systems are published. 9
1.3 Scaled scientograms with edges among thematic areas thresholded. 10
1.4 Scaled scientogram by extracting a minimum spanning tree. 10
1.5 Scaled scientogram by Pathfinder algorithm. 11

3.1 Phases of building a Fingram. 16
3.2 Interpretation of Fingrams. 20
3.3 Node of uncovered instances 21
3.4 Example of node of uncovered instances. 23
3.5 Analysis of inference for an instance using instance-based Fingrams 26
3.6 Fingram of the set of classification rules induced for Ecoli dataset. 32
3.7 Fingram of the set of stretched rules for Ecoli dataset. 35
3.8 Analysis of inference for $I_{321} = \{0.68, 0.67, 0.48, 0.5, 0.49, 0.4, 0.34, pp\}$ in the Ecoli classification problem. 36
3.9 Analysis of inference for $I_{211} = \{0.69, 0.39, 0.48, 0.5, 0.57, 0.76, 0.79, im\}$ in the Ecoli classification problem. 36
3.10 Complete Fingram for the electrical distribution problem. 38
3.11 Fingram scaled with Pathfinder for the electrical distribution problem. . . 39
3.12 Fingram of the best simplified FRBS for the electrical distribution problem. 41
3.13 Examples of chairs used in the poll related to the quantification of intangible assets. 42
3.14 FAR-Fingram for the femininity of chairs assessment problem. 43
3.15 Filtered FAR-Fingrams for the femininity of chairs assessment problem. . . 44
3.16 Structure of *.fs* files handled by Fingrams Generator software. 45
3.17 Software scheme of Fingrams Generator software. 47
3.18 Screenshot of resultant window of Fingrams Generator software. 48
3.19 Generation of Fingrams in the software tool GUAJE. 50
3.20 GUAJE Fingrams window. 51
3.21 Fingram analysis window in the software tool GUAJE. 52
3.22 Experiments window of suite KEEL. 53

3.23 Fingrams workflow in KNIME software platform. 55
3.24 Illustrative example (IRIS) using KNIME software platform. 56
3.25 Comparison of layouts in KNIME. 57

List of Tables

1.1 Quadrant of interpretability indices. 4
1.2 Characteristics of visualization methods for multi-dimensional fuzzy rules. . 6
1.3 Most important areas where works related to fuzzy interpretability issues are published (ranked by number of publications) 8

3.1 Information of the set of classification rules induced for Ecoli dataset. . . . 31
3.2 Information of the set of stretched rules for Ecoli dataset. 34
3.3 Quality evaluation of the FRBSs proposed for the electrical distribution problem. 40

Chapter 1

Introduction

Fuzzy set theory –introduced by L. A. Zadeh in 1965 [83]– deals with the uncertainty and fuzziness arising from interrelated humanistic types of phenomena such as subjectivity, thinking, reasoning, cognition, and perception. Fuzzy systems can model complex phenomena yielding good results when conventional mathematical methods perform poorly. In general, fuzziness describes objects or processes that are not amenable to precise definition or measurement vaguely defined and have some uncertainty in their description. Fuzzy systems are widely used over different scenarios, e.g. control systems [78], classification [53] or regression tasks [22].

Since Zadeh's proposal [83] and Mamdani's seminal ideas [54], interpretability is acknowledged as one of the most appreciated and valuable characteristics of fuzzy systems. Interpretability of fuzzy systems represents their ability to formalize the behavior of a real system in a human understandable way [7, 19], becoming an essential requirement for those applications that involve extensive interaction with human beings. It takes advantage of the use of linguistic variables [85] and rules [54, 84] with high semantic expressivity close to natural language. According to some authors, interpretability is of subjective nature and depends on the expertise and background of the end-user [10].

Here we will focus on the so-called *humanistic systems*, defined by Zadeh [85] as those systems whose behavior is strongly influenced by human judgments, perceptions or emotions. For example, decision support systems in medicine [6, 60] must be easily understandable, for both physicians and patients, with the intention of being reliable, i.e., widely accepted and successfully applicable.

Unfortunately, fuzzy systems are not interpretable per-se. Although the use of linguistic variables and rules favors interpretability, this does not guarantee it. A careful design is demanded to simplify their understanding and ensure their interpretability [57, 80].

The rest of this chapter overviews the main fields of study of this thesis. It starts with an historical review of interpretable fuzzy systems; it continues reviewing the main works that graphically represent fuzzy systems; and it ends contextualizing social network analysis methods and applications.

1.1 The History of Interpretable Fuzzy Systems

The historic evolution of fuzzy modeling –system modeling of fuzzy systems– include three main periods regarding interpretable issues, as sketched in [7]: from 1965 to 1990 the focus was on expert knowledge, from 1990 to 2000 the focus shifted towards automatic knowledge extraction from data, and from 2000 until now the main challenge regards interpretability-accuracy trade-off. Figure 1.1 shows the distribution of publications per year along with the presented periods regarding interpretability issues[1].

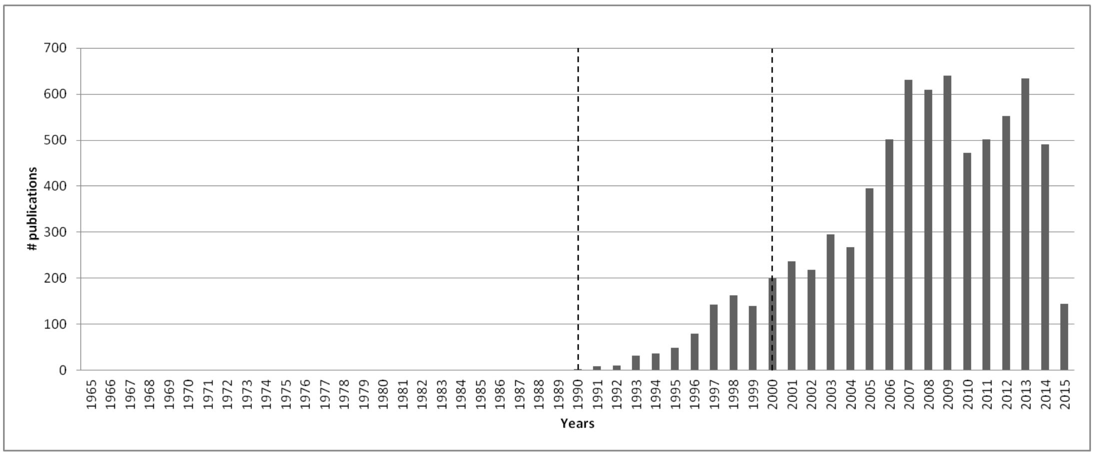


Figure 1.1: Publications per year related to interpretability issues.

In the early days of fuzzy systems, from 1965 to 1990, interpretability naturally emerged as the main advantage of fuzzy systems. Fuzzy models were mainly built up from expert knowledge, using simple linguistic variables and rules. As a result, those designed fuzzy models were characterized by their high interpretability. Moreover, interpretability was assumed as an intrinsic property of fuzzy systems. The first proposal of a fuzzy rule-based system (FRBS) was presented by Mamdani [54] who was able to augment Zadeh's initial formulation [83] allowing the application of fuzzy systems to a control problem. Mamdani-type FRBSs soon arised as the most popular tool to develop linguistic models. Other rule formats were developed in that period, but probably the most famous among them are those proposed by Takagi and Sugeno, the popular TSK fuzzy systems, where the conclusion is a function of the input values.

In the second period, from 1990 to 2000, researchers focused in improving the accuracy of fuzzy systems. They had realized that expert knowledge was not enough to deal with complex systems and they introduced the use of fuzzy machine learning techniques to automatically extract knowledge from data. Those designed fuzzy models presented advanced mechanisms in all the elements of the fuzzy system (antecedents, rules, inference mechanism, and so on). As a consequence, those systems were rarely as interpretable

[1]Data retrieved from the Thomson Reuters ISI Web of Science(*http://www.webofscience.com/*) on May the 28th 2015.

as desired. Along this period some researchers claimed that fuzzy models are not interpretable per se; they have to be designed carefully to fulfill that characteristic. Although the amount of publications related to interpretability issues was still small in this period, it began to grow exponentially at the end of this second phase.

In the last years, from 2000 until now, authors noticed that both previous approaches have their advantages and drawbacks, but they are complementary. As a result, the main challenge has been how to combine expert knowledge and knowledge extracted from data, with the aim of designing compact and robust systems with a good interpretability-accuracy trade-off.

In addition, fuzzy modeling has been carried out through two alternative approaches attending to the interpretability-accuracy trade-off: producing linguistic or precise fuzzy modeling. Linguistic fuzzy modeling (LFM) prioritizes interpretability, yielding fuzzy rules composed of linguistic variables taking terms with a real-world meaning. On the contrary, precise fuzzy modeling (PFM), which has accuracy as its main objective, constructs FRBSs that jeopardizes semantic expressivity. An effort has been done to obtain intermediate approaches that keep a good interpretability-accuracy trade-off. On the one hand, some works propose to improve accuracy of LFM [20]. On the other hand, others introduce techniques to enhance interpretability of PFM [19].

At the same time, during this last period, there has been a great effort for formalizing interpretability issues. According to some researches the assessment of interpretability has to face two main issues [10]: (1) readability (transparency) of the system description, related to the view of the model structure as a gray-box, and (2) comprehensibility of the system explanation, which is closer to cognitive aspects because it is always related to human beings. Of course, the analysis has to take into account all elements included in a fuzzy system, from the lowest (fuzzy partitions) to the highest (fuzzy rules) abstraction levels [88].

There are lots of interpretability indices focusing on specific characteristics of fuzzy systems. As exposed in [40], interpretability indices can be grouped according to two orthogonal criteria: the nature of the interpretability index (structure vs. semantics) and the elements of the fuzzy knowledge base that it considers (fuzzy partitions vs. rule base). Thus, Table 1.1 sketches the four derived groups: (Q1) structure at partition level, (Q2) structure at rule base level, (Q3) semantics at partition level, and (Q4) semantics at rule base level.

Most well-known existing interpretability indices correspond to groups Q1 and Q2, thus they focus on readability (in terms of complexity at structural level) of fuzzy systems. In consequence, they are objective indices since they basically count the number of elements (features/variables, membership functions, rules, premises, etc.) existing in the fuzzy system.

Indices included in group Q3 usually measure the degree of fulfillment of semantic constraints that should be overimposed during the design process. In [29] Oliveira pro-

Table 1.1: Quadrant of interpretability indices [40].

	Fuzzy Partition Level	Rule Base Level
Structural-based Interpretability	Q1 Number of membership functions Number of features/variables	Q2 Number of rules Number of conditions
Semantic-based Interpretability	Q3 Completeness or coverage Normalization Distinguishability Complementarity Relative measures	Q4 Consistency of rules Rules fired at the same time Transparencey of rule structure Cointension

posed some semantic constraints (coverage, normalization, distinguishability, etc.) required to have interpretable fuzzy partitions from the semantical point of view. The use of strong fuzzy partitions (SFP) [73] satisfies all these semantic constraints. Nonetheless, notice that, breaking the SFP property can yield more accurate systems. Therefore, there are proposals that ensure a good interpretability at this level without considering SFP [2, 36, 39].

The last group Q4 is the one that contains the lowest number of works in the literature. These indices advocate for extending the analysis of readability to evaluate the comprehensibility, i.e., the implicit and explicit semantics embedded in fuzzy systems [56]. There are also some papers dealing with the consistency of fuzzy rule bases and with the number of co-fired rules, i.e., rules simultaneously fired by a given input [9, 24, 55].

Finally, the comprehension of the fuzzy inference process is one of the key, and still open, issues regarding the interpretability of fuzzy systems [86]. Understanding such process becomes an arduous task even for fuzzy modeling experts. Fuzzy systems usually cover the input space densely, that is, several rules jointly cover same parts of the input space in common. When studying the inference of a single instance, some rules are simultaneously fired hindering its comprehension. Even more, fuzzy systems usually use weighted rules, advanced defuzzification strategies, and a high number of rules, variables or antecedents per rule, that occlude the system behavior at inference level [8, 19].

The use of different t-norms, t-conorms and fuzzy implication operators may also occlude the comprehension of the fuzzy inference process. Some works have put the effort in the semantics along with implicative and conjunctive fuzzy rules [50, 33]. Implicative fuzzy rules describe pieces of generic knowledge, while conjunctive fuzzy rules encode instances-originated information expressing either mere possibilities or how typical situations can be extrapolated. Depending on their interpretation, rules have to be represented and processed in a specific way at the inference level [32].

1.2 State of the Art on Visual Description and Analysis of Fuzzy Systems

Very few publications deal with visual analysis of fuzzy systems. Probably, this is due to the well-known linguistic expressivity of such systems what gives prominence to linguistic representations. However, when dealing with complex real-world problems, even when the design is made carefully to maximize interpretability, the number of elements (rules and/or antecedents) can become huge because of the curse of dimensionality characteristic of FRBSs. In those cases, looking for a plausible linguistic explanation of the inferred output, derived from the linguistic description of the fuzzy knowledge base, is not straightforward. When many rules are fired at the same time for a given input, explaining the inferred output as an aggregation of all the involved rules turns up very complicated, even when considering Mamdani fuzzy systems.

Phan and Brown provided in [70] a complete analysis of visualization requirements for fuzzy systems. That contribution overviews existing methodologies to yield two-dimensional (2D) and three-dimensional (3D) graphical representations of fuzzy data, fuzzy partitions, and fuzzy rules. Different alternatives are available depending on the requirements of the end-user (fuzzy designer, domain expert, etc). Moreover, requirements may change depending on the visualization tasks: interactive exploration; automatic computer-supported exploration; receiving feedback from users; and capturing users' profiles and adaptation. Notice that, the correspondence of generality and specificity in between the extracted knowledge and the available data instances is not always straightforward and may become a handicap. So for, a visual representation of the FRBS inference process allows us to find out how rules cover instances and how rules are related among them, because they interact to produce the overall behavior of the system.

Therefore, we will highlight here some of the most relevant works that propose graphical visualizations of fuzzy systems, from the most restrictive to the most general, emphasizing those that consider the inference mechanism of fuzzy systems.

First we spotlight the work [17] by Buck and Keller. Although it does not actually represent fuzzy systems, it reports valuable innovations and possibilities to be extended to deal with fuzzy systems. Authors proposed a method for visualizing vectors of fuzzy numbers developing extensions of the standard polar area diagrams that include uncertainty.

Some works [47, 48, 49] focused on providing visual representations able to explain the output of fuzzy rule-based classifiers to human users establishing a set of design constraints. They advocated for using sets of rules with only two antecedent conditions that can be represented in 2D. Nevertheless, considering only two antecedents per rule is a strong limitation that may penalize the accuracy of the system, especially when dealing with complex and high dimensional problems.

On a different basis, Casillas and Martínez-López [21] presented the so-called "tran-

Table 1.2: Characteristics of visualization methods for multi-dimensional fuzzy rules.

	[15, 43]	[21]	[34, 35]	[38]	[72]
Represent data samples		✓	✓	✓	✓
Represent fuzzy rules	✓	✓	✓	✓	✓
Represent overlapping among rules at descriptive level	✓	✓		✓	✓
Represent rule interaction at inferential level	✓				✓

sition chromatic maps" for fuzzy rules generated from uncertain data. These maps are generated by a visual modeling process that represents the extracted knowledge in a more understandable way. Thus, these maps help in the postprocessing, interpretation stage of knowledge discovery in databases. We can uncover relations among variables by observing the chromatic evolution of the surfaces at the map.

Gabriel et al. proposed in [38] a mapping from high dimensional feature spaces onto two-dimensional spaces which maintains the pairwise distances between rules. The established mapping also displays an approximation of each rule spread and overlapping. As a result, it is possible to visualize and explore multi-dimensional FRBSs in a 2D graphical representation. Authors claim such representation yields a user friendly and interpretable exploratory analysis. However, the complexity of the analysis grows exponentially with the number of variables and rules to be displayed. In consequence, in complex and high dimensional problems, the interpretation of the resulting graph is not straightforward.

Following a similar approach, Evsukoff et al. [34, 35] proposed the use of an interpretation framework that helps understanding multidimensional fuzzy rules. They assigned a symbol to each rule, which is represented by a Gaussian membership function. The model interpretation is based on the analysis of rule weights and on a 2D linear principal component analysis projection to visualize the model.

Rehm et al. introduced in [72] a 2D visualization for fuzzy classifiers where rules and data samples are represented. They used multidimensional scaling to estimate the position of rule centers. Rules are linked according to their neighborhood regarding their core region. Finally, data instances are placed in between the two rules that yield the highest firing degree.

Berthold and Hall represented rule interaction in terms of rule overlapping in [15]. Namely, they graphically represented fuzzy rules antecedents in parallel coordinates to visualize the overlapping among rules. Later, Henzgen et al. [43] enhanced the graphical representation of [15] for observing changes in evolving fuzzy rule-based systems.

Table 1.2 summarizes the main characteristics of the most relevant visualization methods for fuzzy systems previously introduced. All methods make a 2D representation of fuzzy rules. Some of them represent data and some others show the existing overlapping among rules at descriptive level, but only [15, 43, 72] partially represent rule interaction at inference level. This brief review shows that there is a lack of methods depicting the interaction among fuzzy rules that, however, could strongly help in the comprehension of the behavior of fuzzy systems.

1.3 A Brief Review on Social Network Analysis

A social network is a social structure made up of individuals called "nodes", which are connected or tied by "edges" (also called ties, links, or connections) corresponding to one or more specific types of interrelations, such as friendship, common interest, or knowledge. Social network analysis (SNA) [76, 82] views social relationships in terms of network theory regarding nodes and edges. Nodes represent individuals within networks, and ties are the relationships among individuals. Research in a number of academic fields has shown that social networks operate on many levels, from families up to the level of nations, and play a critical role in determining the way how problems are solved, organizations are run, and individuals succeed in achieving their goals.

The combination of SNA techniques has proved its capability to get high quality, schematic visualizations of the resulting networks in various fields: psychology (to represent the cognitive structure of a subject [30, 75]), software development (for debugging of multi-agent systems [77]), scientometrics (for the analysis of large scientific domains [59, 81]), etc.

The term *scientogram*, a particular case of social network, is coined in the specialized literature to make reference to visual science maps, i.e., visual representations of scientific domains. Moya-Anegón et al. [58, 59, 81] proposed a methodology to create scientograms with the aim of illustrating interactions among authors and papers through citations and co-citations. The basic idea turns up from the notion of manuscript co-citation which represents the frequency with which two documents are simultaneously cited by others. It is possible to group them by author, journal, or thematic category, for example. Of course, depending on the kind of grouping, the information that can be extracted from the generated maps is different.

The standardized co-citation measure was originally defined by Salton and Bergmark [74]:

$$MCN(ij) = \frac{Cc(ij)}{\sqrt{c(i) \cdot c(j)}} \tag{1.1}$$

where Cc means co-citation, c stands for citation, i and j represent two different entities (authors, documents, journals, categories, institutions, countries, etc).

As an illustrative example, we present the main areas of science where fuzzy interpretability issues have been studied. For that we use the Thomson Reuters ISI Web of Science[2], WoS now on, which is globally recognized as the premier research platform. This platform provides a unique search method that allows users to navigate through the literature to uncover all the information relevant to their research (filtering works by authors, publication types, time spans, etc.) and to access electronic full text journal articles. With the aim of exploring the visibility of interpretable fuzzy systems at the WoS, we query the following expression in "Advanced Search":

[2] http://www.webofscience.com/

TS=(((interpretab) OR (understandab*) OR (intelligib*) OR (complexity) OR (transpar*) OR (readab*))) AND (TS="fuzzy*" OR SO="fuzzy*" OR CF="fuzzy*")*

As result, we obtain 7455 records[3]. Table 1.3 summarizes the most representative fields of study and thematic areas where works related to fuzzy interpretability issues have been published. As seen, most publications belong to the fields of Computer Science, Engineering, and Mathematics. Artificial Intelligence and Electrical and Electronic are by far the two most important thematic areas.

Table 1.3: Most important areas where works related to fuzzy interpretability issues are published (ranked by number of publications)

Field of Study	Thematic Area	# publications
Computer Science	Artificial Intelligence	3684
Engineering	Electrical & Electronic	2116
Computer Science	Theory & Methods	1227
Engineering	Automation & Control Systems	878
Computer Science	Information Systems	832
Computer Science	Interdisciplinary Applications	770
Mathematics	Applied	469
Mathematics	Operations Research & Management Science	457
Computer Science	Cybernetics	375
Computer Science	Software Engineering	303
Engineering	Telecommunications	265
Mathematics	Statistics & Probability	258
Engineering	Multidisciplinary	204
Engineering	Industrial	202
Computer Science	Hardware & Arquitecture	202
Computer Science	Imaging Science & Photographic Technology	197
Engineering	Manufacturing	187
Engineering	Mechanical	174
	Environmental Sciences	173
Computer Science/Engineering	Robotics	172
Engineering	Biomedical	152
Engineering	Instruments & Instrumentation	147
	Management	139
Engineering	Civil	132
	Mathematical & Computational Biology	123
	Optics	103
Materials Science	Multidisciplinary	101

Then, we have depicted in Figure 1.2 the scientogram of thematic areas relating them according to Eq. 1.1. The picture shows nodes proportional to the volume of produced

[3]Data retrieved on May the 28th 2015.

documents, and edges among nodes represent the connections among the related areas. Notice that this graphical representation is usually very hard to understand because networks are usually very dense, i.e. most of the nodes are related to each other, thus hindering the important edges in the scientogram.

Given a network, scaling algorithms have the goal to take proximity information and to obtain structures revealing the underlying organization. They use similarities, correlations, or distances to prune a graph based on proximity among pairs of nodes. The three predominant ways of network scaling which are proposed in the literature are presented below [23].

The first option introduces an edge weight threshold and it only maintain those edges with weights above that threshold [89]. This approach is straightforward and easy to implement. However, it does not take the intrinsic structure of the underlying network into account, so the transformed network may not preserve the essence of the original one. Furthermore, the value of the threshold is usually hard to be adjusted by the user.

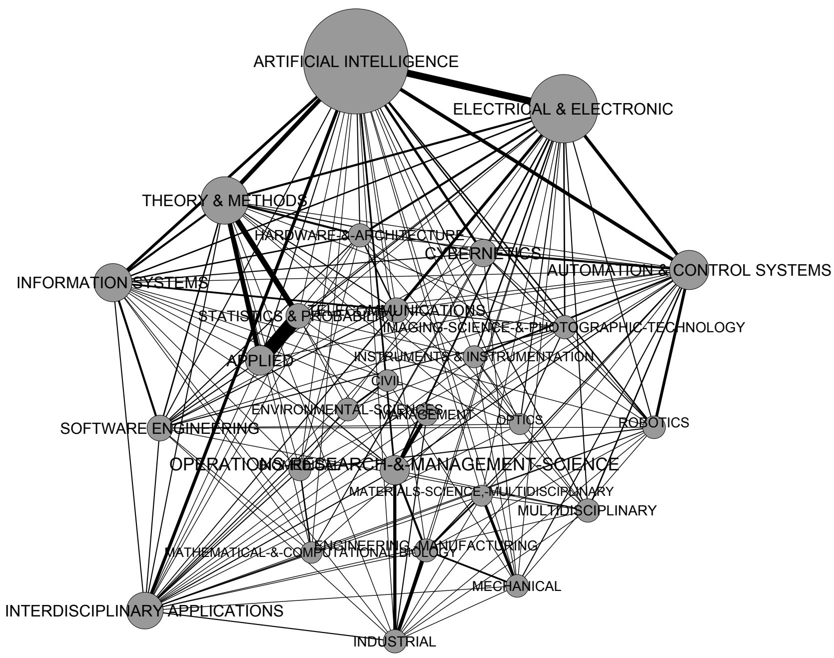
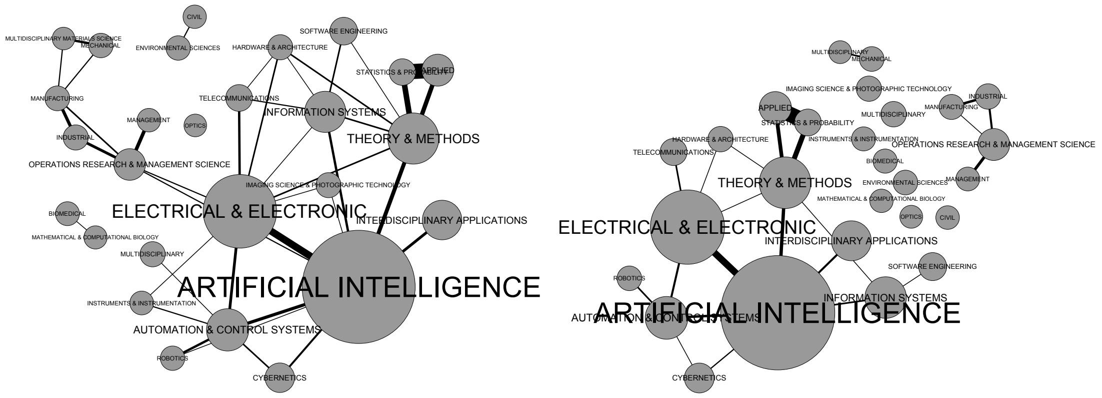


Figure 1.2: Scientogram with the main thematic areas where works dealing with interpretability of fuzzy systems are published.

Here, we scaled the scientogram in Figure 1.2 considering two different thresholds. Results are depicted in Figure 1.3. As seen, the use of a low value as threshold yields several connections that are not properly represented (Figure 1.3(a)). On the other hand, considering a high (and thus more restrictive) value as threshold produces several unconnected nodes that previously were connected (Figure 1.3(b)).

The second option extracts a minimum spanning tree from the network [61]. This approach formes trees by set of connected nodes, and guarantees the number of edges



(a) Low threshold. (b) High threshold.

Figure 1.3: Scaled scientograms with edges among thematic areas thresholded.

in this sets is always $N - 1$ (with N the number of connected nodes). Unfortunately, that does not always reflect all the subjacent relevant information. Figure 1.4 shows the scientogram in Figure 1.2 scaled extracting a minimum spanning tree.

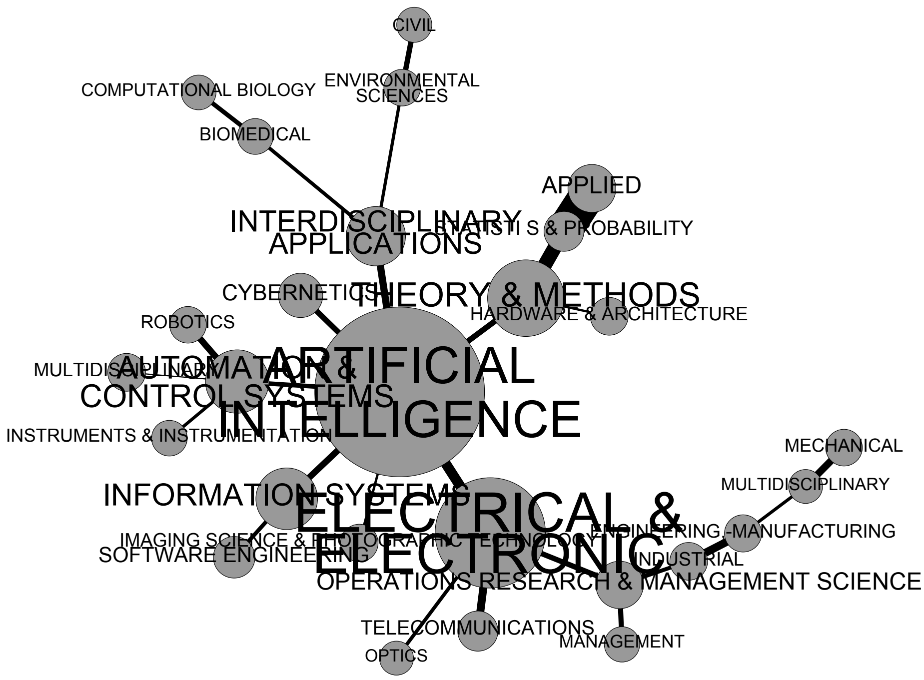


Figure 1.4: Scaled scientogram by extracting a minimum spanning tree.

The third option imposes several constraints on paths and then excludes those edges that do not satisfy such constraints. One of the most known methods, the Pathfinder algorithm [30, 75], is frequently used due to its mathematical properties related to the preservation of the triangular inequality. Those properties include the conservation of edges, the capability of modeling symmetrical but also asymmetrical relationships, and the representation of the most *salient* relationships present in the data.

Figure 1.5 depicts the scientogram in Figure 1.2 after network scaling by Pathfinder algorithm. This scientrogram nodes are colored depending on the fields of study to which they belong to. It is easy to appreciate how this scaling highlights the backbone in Figure 1.2. It is worthy to note that important thematic areas naturally take a central position in this representation. This solution do not produce unconnected nodes as occurs using thresholding (Figure 1.3). Even more, Pathfinder is deterministic keeping all the edges that produce ties, outperforming the use of minimum spanning trees (Figure 1.4). We can see that thematic areas in the field of Computer Science are in a central position and highly related. On the contrary, Engineering thematic areas are spread along the scientogram. Finally, thematic areas published in other fields are in the borders of the graphical representation.

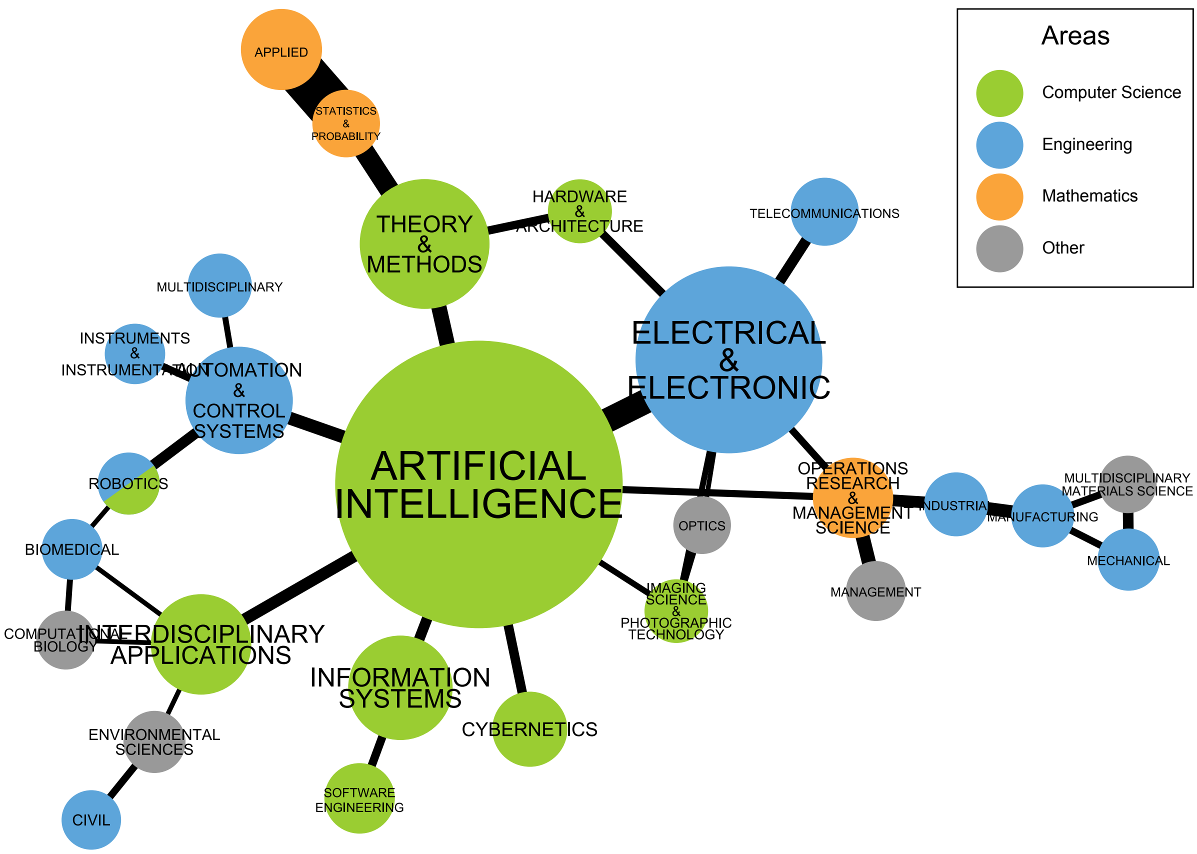


Figure 1.5: Scaled scientogram by Pathfinder algorithm.

There are many different methods for automatic visualization of social networks. Force-based or force-directed algorithms are the most widely used class of algorithms for drawing graphs in the area of information science [31, 52]. Their purpose is to locate the nodes of a graph in a two or three dimensional space so that all edges are approximately of equal length and there are as few crossing edges as possible, trying to obtain the most aesthetically pleasing view. All the visual representations of this section (from Figure 1.2 to Figure 1.5) are created using Force-based algorithms.

Chapter 2

Objectives

The main goal of this thesis is the study, representation and analysis of fuzzy systems emphasizing the search for a good interpretability-accuracy trade-off. We proposed taking advantage of social network analysis techniques for the understanding of the fuzzy systems' inference process.

Namely, the following enumeration summaries the objectives established at the beginning of the thesis:

1. To propose a new graphical representation based on techniques of social network analysis that shows the behavior of fuzzy systems at inference level.

2. To study the interaction among elements in fuzzy systems in order to include it in the new graphical representation.

3. To introduce a new interpretability index that reflects how complex the inference mechanism of the given fuzzy system is according to the complexity of the graphical representation.

4. To evaluate alternative metrics that relate rules of a fuzzy system.

5. To study different alternatives for the filtering, scaling, and drawing of the social networks that represent fuzzy systems.

6. To establish a methodology for designing and improving interpretable fuzzy systems guided by their graphical representation.

Note that these objectives have been successfully fulfilled along the thesis period, as we will thoroughly overview in the following chapter.

Chapter 3

Discussion of results

3.1 A new methodology for visual representation and exploratory analysis of the fuzzy inference process in fuzzy systems

This dissertation proposes a new methodology for visual representation and exploratory analysis of the fuzzy inference process in fuzzy systems. In such systems, an instance can simultaneously fire various rules. Moreover, the usual behavior of FRBSs is that, given a set of problem instances, several fuzzy rules are fired at the same time. In other words, the input space is usually covered by rules with dense overlapping among them.

We take advantage of this characteristic using a set of problem instances to identify co-fired rules. This co-firing information is used to create social networks representing fuzzy systems' inference maps, called fuzzy inference-grams or *Fingrams* in short. In these kinds of social networks each fuzzy rule is represented by a node, and the relations among rules are represented by weighted edges whose value is computed using a specific metric. Different metrics can be used to construct a social network (as it will be sketched in Section 3.1.1) given a dataset of intances representing the input-output relations existing in the problem tackled, a set of fuzzy rules, and a fuzzy inference mechanism. As a result, Fingrams graphically show the interaction among fuzzy rules at inference level in terms of co-fired rules.

Due to the high overlapping among rules, the complete Fingram is usually very dense difficulting its analysis even for medium-size FRBSs. Fortunately, network scaling methods can be used to simplify Fingrams while maintaining their most important relations, as Section 3.1.2 will overview.

Social networks can be represented by the use of drawing methods especially designed for that purpose. Here, a specific graph representation is developed to provide the relevant information of the FRBS under study. Colors and sizes highlight distinguishing characteristics of the system, allowing the end-user to do a thorough analysis. Section 3.1.3 will give a deeper explanation about drawing methods used.

From a formal viewpoint, we define a Fingram as follows:

Definition A *Fingram* is a tuple $(R, P, I, D, m, NSM, NDM)$ in which:

R is the set of fuzzy rules (nodes), denoted R_i, $1 \leq i \leq r$, with r being the number of rules.

P is the set of fuzzy partitions of input and output variables.

I is the fuzzy inference mechanism used.

D is the set of problem instances, denoted I_k, $1 \leq k \leq d$, with d being the number of instances.

m is the metric used to create M, the square weight matrix $(r \times r)$ that represents the firing interactions among fuzzy rules. The entries of that matrix are the weights associated with the edges; m_{ij} is the weight of the edge connecting R_i and R_j.

NSM is the considered network scaling method.

NDM is the considered network drawing method.□

The remaining of the section explains in detail the procedure followed to create Fingrams, sketched in Figure 3.1.

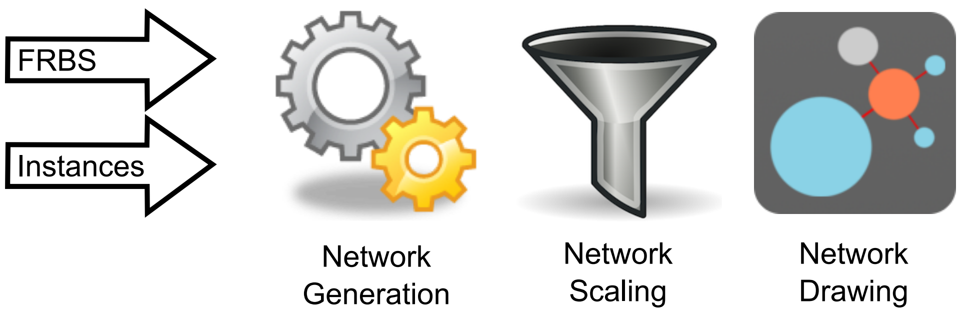


Figure 3.1: Phases of building a Fingram.

3.1.1 Fingrams Generation

Starting from a set of fuzzy rules R, a set of fuzzy partitions P, a fuzzy inference mechanism I, a set of problem instances D, and a metric m, a social network can be built, represented by a matrix M, that shows the relations among rules.

Each instance (I_k) is represented as an n-dimensional attribute vector $(\mathbf{x_k})$ plus its output (y_k):

$$I_k = \{(\mathbf{x_k}, y_k) \mid \mathbf{x_k} = \{x_k^1, ..., x_k^n\},\ x_k^h \in \mathbb{R},\ h \in [1, n]\} \quad (3.1)$$

In this context, we consider a set of r fuzzy rules of the form:

$$R_i\text{: IF } X_1 \text{ is } A_1^i \text{ \& ... \& } X_n \text{ is } A_n^i \text{ THEN } Y \text{ is } B_i \text{ with } w^i \tag{3.2}$$

where A_h^i is the rule antecedent for variable X_h ($h \in [1, n]$ with n the number of attributes), B_i denotes the consequent, and w^i is the weight associated to rule R_i.

We define the firing degree of a rule R_i for a single instance (I_k) as:

$$\mu_{R_i}(I_k) = \mu_{A_1^i}(x_k^1) \otimes ... \otimes \mu_{A_n^i}(x_k^n) \tag{3.3}$$

with $\otimes$ being a t-norm.

We distinguish between covered and uncovered instances. We define the set of covered instances (cv) as those firing at least one rule out of the set of rules, and the set of uncovered instances (ucv) as those ones that do not fire any of the rules.

$$cv = \{I_k \mid \sum_{i=1,...,r} (\mu_{R_i}(I_k)) > 0,\ k \in [1, d]\} \tag{3.4a}$$

$$ucv = \{I_k \mid \sum_{i=1,...,r} (\mu_{R_i}(I_k)) = 0,\ k \in [1, d]\} \tag{3.4b}$$

where d is the number of problem instances and $\mu_{R_i}(I_k)$ is the firing degree as presented in Eq. 3.3.

A square matrix M ($r \times r$) contains all interactions among rules:

$$M = \begin{pmatrix} 0 & m_{12} & \dots & m_{1r} \\ m_{21} & 0 & \dots & m_{2r} \\ \dots & \dots & \dots & \dots \\ m_{r1} & m_{r2} & \dots & 0 \end{pmatrix} \tag{3.5}$$

Different metrics have been proposed, named as m_{ij}^0, m_{ij}^1 and m_{ij}^2.

The simplest metric m_{ij}^0, inspired by the co-citation measure of scientograms (Eq. 1.1), relates two rules (R_i and R_j) according to the number of instances covered in common by them ($|D_i \cap D_j|$) with respect to the total number of instances they individually cover ($|D_i|$ and $|D_j|$):

$$m_{ij}^0 = \begin{cases} \frac{|D_i \cap D_j|}{\sqrt{|D_i||D_j|}} & \text{, if } i \neq j \\ 0 & \text{, if } i = j \end{cases} \tag{3.6}$$

with D_i representing the set of instances covered by rule R_i:

$$D_i = \{I_k \mid \mu_{R_i}(I_k) > 0, k \in [1, d]\} \tag{3.7}$$

and $\mu_{R_i}(I_k)$ being the firing degree up to which a single instance (I_k) fires the rule R_i (as presented in Eq. 3.3).

Notice that, m^0_{ij} is normalized and the matrix M is symmetrical. This metric obtains zero ($m^0_{ij} = m^0_{ji} = 0$) when two rules do not cover any instance in common ($D_i \cap D_j = \emptyset$), and one ($m^0_{ij} = m^0_{ji} = 1$) when both rules cover exactly the same data instances ($D_i = D_j$, $i \neq j$).

We have also defined a second more advanced metric m^1_{ij} which includes the firing degree up to which the data instances activate the rules as well as the rule weights:

$$m^1_{ij} = \begin{cases} \frac{\sum\limits_{I_k \in \{D_i \cap D_j\}} \left(\min(\mu_{R_i}(I_k) \cdot w^i, \mu_{R_j}(I_k) \cdot w^j) \right)}{\sqrt{(FD_{R_i} \cdot w^i) \cdot (FD_{R_j} \cdot w^j)}} & \text{, if } i \neq j \\ 0 & \text{, if } i = j \end{cases} \tag{3.8}$$

$\{D_i \cap D_j\}$ represents the set of instances firing both rules R_i and R_j at the same time. $\mu_{R_i}(I_k)$ is the firing degree of R_i given the data instance I_k (Eq. 3.3). And FD_{R_i} represents the accumulated firing degree for all the instances in D that fire rule R_i:

$$FD_{R_i} = \sum_{k=1,\ldots,d} (\mu_{R_i}(I_k)) \tag{3.9}$$

This metric produces symmetric relations and is normalized.

An asymmetric co-firing metric m^2_{ij} [63] is defined as:

$$m^2_{ij} = 1 - \frac{\sum_{I_k \in D_i} (|\mu_{R_i}(I_k) - \mu_{R_j}(I_k)|)}{\sum_{I_k \in D_i} \mu_{R_i}(I_k)} \tag{3.10}$$

with D_i being all the instances firing rule R_i (Eq. 3.7); $\mu_{R_i}(I_k)$ is the firing degree up to which a single instance I_k fires the rule R_i (Eq. 3.3).

This co-firing metric characterizes generalization/specialization relations between pairs of rules. Moreover, it yields a an asymmetrical matrix that produces a directed graph, i.e., each pair of nodes can be connected by two different edges. A rule R_i is highly related with another R_j, i.e. $m^2_{ij} \approx 1$, when R_j is fired at similar degrees by the same set of instances that fires R_i.

3.1.2 Fingrams Scaling

As usual in social network design, the initial Fingram is commonly quite dense and difficult to analyze even for medium-size FRBSs. So for, a network scaling method is required to simplify it, keeping all the nodes but only the most important relations.

Given a network, scaling algorithms have the goal to take proximity information and to obtain structures revealing the underlying organization (as shown in Section 1.3). They use similarities, correlations, or distances to prune a graph based on proximity among pairs of nodes. The three predominant ways proposed in the literature as presented in section 1.3 are listed below [23].

1. **Thresholding**: It only maintains edges with weights above a given threshold [89]. This approach is straightforward and easy to implement. However, it does not take the intrinsic structure of the underlying network into account, so the transformed network may not preserve the essence of the original one. Furthermore, the value of the threshold could be hard to adjust by the end-user.

2. **Minimum Spanning Tree**: It extracts a minimum spanning tree from a network of N nodes [61]. This approach guarantees the number of edges in the transformed network is always $N - 1$. However, this fact does not always reflect the subjacent relevant information.

3. **Imposing constraints**: This option imposes constraints on paths and excludes edges that do not satisfy the constraints. One of the most known methods, the Pathfinder algorithm [30, 75], is frequently used due to its mathematical properties that include the conservation of edges, the capability of modeling symmetrical but also asymmetrical relationships, and the representation of the most *salient* relationships present in social networks.

We performed an analysis of alternatives in [65] concluding that the use of Pathfinder algorithm fits our requirements. It yields a close-to-tree global structure which provides valuable information very easy to interpret. Namely, we use Fast Pathfinder [71], a variant of Pathfinder that reduces the computational complexity of the original algorithm.

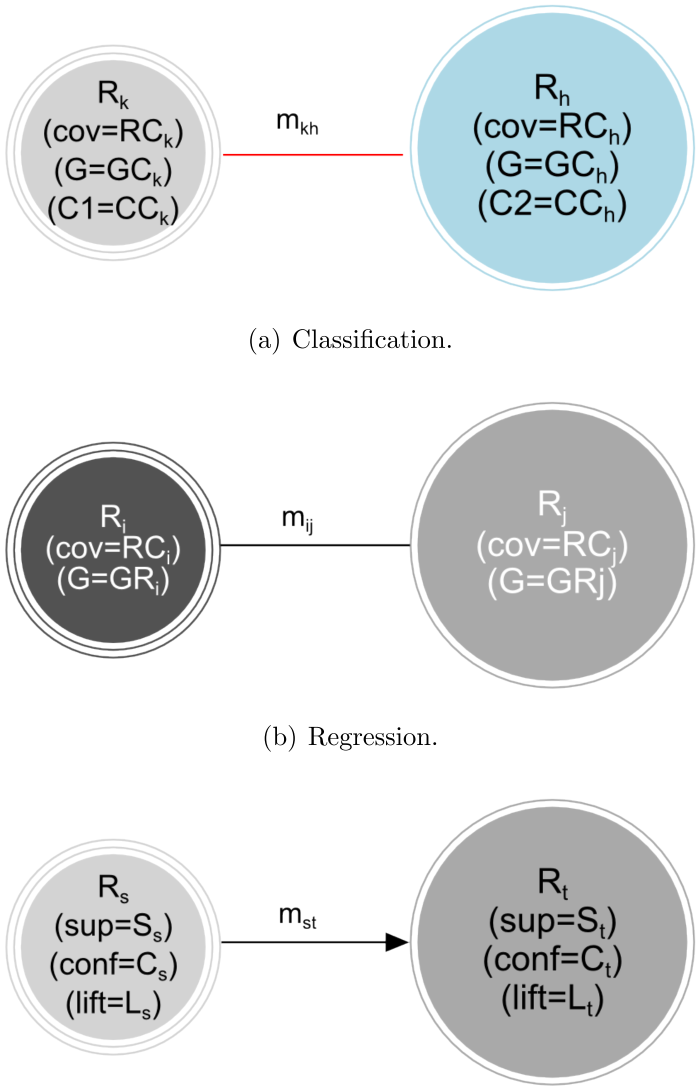
3.1.3 Fingrams Drawing

As previously outlined in Section 1.3, force-based algorithms are devoted to represent social networks in an aesthetically pleasing way. Those methods try to clarify as much as possible the resultant graph, allowing the user to easily identify and interpret the final distribution of elements. Thus, to satisfy this goal, two requirements have mainly to be achieved: reduce the crossover of edges and distribute the nodes and edges homogeneously through the canvas.

In order to visualize the scaled network in a 2D space, force-based algorithms assign coordinates to the nodes obtaining a graph with the most important elements placed toward the center of the image. Kamada-Kawai [51] and Fruchterman-Reingold [37] are the most representative force-based algorithms.

Kamada-Kawai [51] represents networks following aesthetic criteria trying to maximize the use of the available space, to minimize the number of crossed edges, to force the separation of nodes, etc. It assigns coordinates to the nodes trying to adjust as much as possible the distances existing among them with respect to actual network distances.

In the Fruchterman-Reingold Algorithm [37], the attraction or repulsion among nodes determines in which direction a node should move. Nodes move from an original layout



(a) Classification.

(b) Regression.

(c) Fuzzy Association Rules.

Figure 3.2: Interpretation of Fingrams.

step by step. The step width of node movements decreases at each iteration. Once nodes stop moving, the procedure ends.

Kamada-Kawai, through Graphviz implementation[1], is used in our approach because it has been proved very effective in combination with Pathfinder [81]. This solution is flexible to be adapted to the particularities of new scenarios.

The proposed representation includes graphical information of special interest for FRBS inference analysis:

- Rules are represented by nodes and labeled with useful textual information that depends on the type of problem we are dealing with (see Figure 3.2). Nodes are labeled in all of the cases with the rule identifier, and can include more information depending on the type of problem under consideration.

[1] *http://www.graphviz.org/* [41]

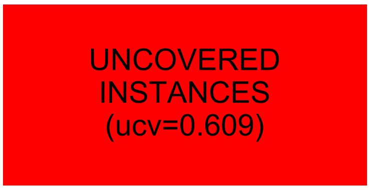


Figure 3.3: Node of uncovered instances

- Rules not covering any instance, $FD_{R_i} = 0$ (as presented in Eq. 3.9), are colored in red to facilitate their detection. Those rules are good candidates to be removed from the rule set.

- The node size is established according to the number of instances covered by the rule. The higher the amount of covered instances, the bigger the node size. For example, rule R_h covers more instances than R_k according to Figure 3.2(a).

- Node color is used to identify different particularities depending on the problem. For example, in classification problems node color indicates rule output, with same color for rules of the same output class. Rules R_k and R_h have different output class (as observed in Figure 3.2(a)).

- Node border indicates how complex the antecedents of the rules are. Single-line border indicates one premise, double-line border two premises, and so on. Thus, the rules R_i and R_j depicted in Figure 3.2(b) have three and two antecedents, respectively.

- Edges among nodes represent rule co-firing information. Each edge represents the relation between a pair of fuzzy rules. The higher the degree of overlapping existing over rules, the higher the edge weight (m_{ij}) and the thicker the edge width in the visual representation.

- In case of directed graphs created using metric m^2_{ij} (Eq. 3.10), arrows indicate the direction of the relation, as presented in Figure 3.2(c). On the contrary, Fingrams constructed with symmetric metrics (Eq. 3.6 and Eq. 3.8) use plain lines, as presented in Figure 3.2(a) and Figure 3.2(b).

- A visual artifact presents the set of instances not covered by the given set of rules (as presented in Eq. 3.4(b)). Instances not covered by any rule are represented by a rectangular node labeled "UNCOVERED INSTANCES". Its height is proportional to the number of instances not covered by any rule. Figure 3.3 presents an example where many instances are covered by none of the rules of the FRBS. In fact, more than half of the instances are uncovered ($ucv = 0.609$).

3.1.4 Types of Fingrams

Fingrams can already deal with fuzzy rule-based classifiers, fuzzy rule-based regressors [65], and fuzzy association rules [63]. The different adaptations show relevant information according to their characteristics.

3.1.4.1 Classification Fingrams

Fingrams that deal with fuzzy rule-based classifiers has colored nodes corresponding to the rule output classes. Rules yielding the same class are depicted by the same node color. Nodes include the following information line by line:

1. Rule identifier (R_i)

2. Coverage of the rule (cov_i) defined by:

$$cov_i = \frac{|D_i|}{|D|} \tag{3.11}$$

   with D_i the set of instances covered by rule R_i (Eq. 3.7) and D the total set of instances.

3. Goodness of the rule (G_i) i.e. how the rule behaves with respect to the problem instances available. This goodness measure reflects how well the problem instances covered by a rule are classified or modeled.

$$G_i = \frac{\sum_{I_k \in D_i^+} \mu_{R_i}(I_k) - \sum_{I_k \in D_i^-} \mu_{R_i}(I_k)}{FD_{R_i}} \tag{3.12}$$

   where D_i^+ and D_i^- denote respectively the sets of positive and negative instances for rule R_i. We call positive instances (D_i^+) to those covered by the rule in a consistent manner, i.e. the output class is the same for the rule and the data instance, being the remaining negative instances (D_i^-). FD_{R_i} is as defined in Eq. 3.9.

$$D_i^+ = \{I_k \mid \mu_{R_i}(I_k) > 0 \;\&\; y_k = B_i,\ k \in [1, d]\} \tag{3.13a}$$

$$D_i^- = D_i \cap D_i^+ = \{I_k \mid \mu_{R_i}(I_k) > 0 \;\&\; y_k \neq B_i,\ k \in [1, d]\} \tag{3.13b}$$

   In a similar vein, some authors propose the use of *purity* to measure the rules behavior [45]. We define the *purity* of rule R_i (pur_i) as follows:

$$pur_i = \frac{\sum_{I_k \in D_i^+} \mu_{R_i}(I_k)}{\sum_{I_k \in D_i} \mu_{R_i}(I_k)} \tag{3.14}$$

   where D_i^+ and D_i denote the set of positive and covered instances for rule R_i respectively (as presented in Eq. 3.13a and Eq. 3.7).

4. Class coverage of a rule (cc_i) as the proportion of covered instances consistent with the rule output class (D_i^+) with respect to the total instances of the dataset consistent with the rule output class (D^{c_l}).

$$cc_i = \frac{|D_i^+|}{|D^{c_l}|} \tag{3.15}$$

where D^{c_l} are the set of instances with output class c_l.

$$D^{c_l} = \{I_k \mid y_k = c_l, k \in [1, d]\} \tag{3.16}$$

The node of uncovered instances is filled using vertical colored strips that give the proportion of uncovered instances related to each class. Figure 3.4 presents an example where there are instances not covered corresponding to classes C0 and C1.

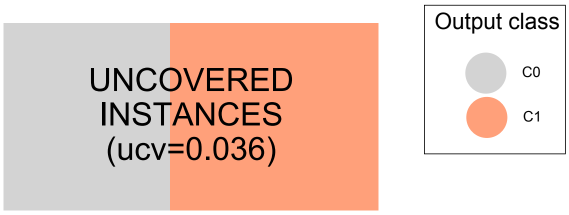


Figure 3.4: Example of node of uncovered instances.

The color of edges gives useful information as well. Edges between rules of the same output class c_l ($B_i = B_j$ for rules R_i and R_j) are colored in green while potential inconsistencies, i.e. edges between co-fired rules pointing out different classes ($B_i \neq B_j$) are remarked with red color (See Figure 3.2(a)).

3.1.4.2 Regression Fingrams

We are capable of constructing Fingrams to deal with multi-input-single-output (MISO) FRBSs for regression.

In this case, the output variable is ordered in its universe of discourse. This order is used to assign grey tones to nodes background, from black to white. So for, the typical behavior will relate nodes with similar greyness, and related nodes showing quite different tones should be studied in detail.

In this case there is no difference among edges, contrary to what happens in classification problems with redundancies and inconsistencies, and they just inform about their weight (See Figure 3.2(b)).

Nodes include the following information line by line:

1. Rule identifier (R_i)

2. Coverage of the rule (cov_i) as defined in Eq. 3.11

3. Goodness of the rule (G_i) similarly to the case of classification problems, but extended for regression problems.

In this case, the node of uncovered instances is colored in red in order to highlight the existence of uncovered instances (Figure 3.3).

3.1.4.3 Fuzzy Association Rule Fingrams

Fingrams for fuzzy association rules, in short FAR-Fingrams, are aimed at dealing with the particularities of those systems and facilitating their visual analysis and comprehension.

In this case, association rules uncover and represent dependencies among items in a dataset [87]. These are represented like $X \rightharpoonup Y$ where X and Y are itemsets and $X \cap Y = \oslash$ [1]. We should understand them as if X appears in a pattern is highly probable that Y appears there as well.

More precisely, we define fuzzy association rules as (substituting definition of rules gave in Eq. 3.2):

$$R_i : \{X_1 \text{ is } A_1^i \ \& \ ... \ \& \ X_n \text{ is } A_n^i\} \rightharpoonup \{Y_1 \text{ is } B_1^i \ \& \ ... \ \& \ Y_n \text{ is } B_n^i\} \tag{3.17}$$

Let us show an easy example of fuzzy association rule from a dataset with three attributes (Att_1, Att_2 and Att_3) and three linguistic terms each one (Low, Medium, and High). An illustrative fuzzy association rule could be:

$$R_i\text{: } \{Att_1 \textit{ is } \text{LOW} \textit{ and } Att_2 \textit{ is } \text{HIGH} \} \rightharpoonup \{Att_3 \textit{ is } \text{MEDIUM}\}.$$

In case of fuzzy association rules, and once they are considered as conjunctive rules in contraposition to implicative rules, the matching degree of the rule is calculated considering the rule antecedent and consequent of rule (substituting Eq. 3.3):

$$\mu'_{R_i}(I_k) = \mu_{A_1^i}(I_k) \otimes ... \otimes \mu_{A_n^i}(I_k) \otimes \mu_{B_1^i}(I_k) \otimes ... \otimes \mu_{B_n^i}(I_k) \tag{3.18}$$

with $\otimes$ being a t-norm.

Note that this substitution affects in the construction of Fingrams, and an instance is covered by a fuzzy association rule only when $\mu'_{R_i} > 0$. This influence metrics m_i^0 (Eq. 3.6) implicitly and m_i^1 and m_i^2 explicitly (Eqs. 3.8 and 3.10).

FAR-Fingram nodes are labeled with the following relevant textual information (as shown in Figure 3.2(c)):

1. Rule identifier (R_i)

2. *Support* of the rule (sup_i).

$$sup_i = \frac{\sum_{I_k \in D} \mu'_{R_i}(I_k)}{\mid D \mid} \tag{3.19}$$

being $\mu'_{R_i}(I_k)$ as defined in Eq. 3.18; and $\mid D \mid$ the cardinality of the dataset D.

3. *Confidence* of the rule ($conf_i$).

$$conf_i = \frac{\sum_{I_k \in D} \mu'_{R_i}(I_k)}{\sum_{I_k \in D} \mu_{X_i}(I_k)} \tag{3.20}$$

with $\mu'_{R_i}(I_k)$ the matching degree of the pattern I_k with the rule R_i; and $\mu_{X_i}(I_k)$ the matching degree of I_k with the rule antecedents of rule R_i:

$$\mu_{X_i}(I_k) = \mu_{A_1^i}(I_k) \otimes ... \otimes \mu_{A_n^i}(I_k) \tag{3.21}$$

with $\otimes$ being a t-norm.

4. *Lift* of the rule ($lift_i$).

$$lift_i = \frac{conf_i}{\sum_{I_k \in D} \mu_{Y_i}(I_k)/|D|} \tag{3.22}$$

being $\mu_{Y_i}(I_k)$ the matching degree of I_k with the rule consequent of R_i.

$$\mu_{Y_i}(I_k) = \mu_{B_1^i}(I_k) \otimes ... \otimes \mu_{B_n^i}(I_k) \tag{3.23}$$

with $\otimes$ being a t-norm.

In the classification and regression Fingrams, the size of nodes is proportional to the number of instances covered by the related rules. Dually, in FAR-Fingrams the node size is determined by the *Support* of the corresponding rule, because it plays a central role in the assessment of fuzzy association rules [79]. Even more, the number of node borders also shows the number of rule antecedents.

Regarding the color of nodes, FAR-Fingrams use a grey scale to indicate the *Lift* level of rules which yields an idea about the goodness of the rules. The higher the *Lift*, the darker the node background, being white the rule with the lowest *Lift*.

Thus, FAR-Fingrams show simultaneously information related to several quality measures. Obviously, this is much more effective than considering only one-ranking evaluation guided by a single measure as usual.

The node of uncovered instances is, as in the case of Regression Fingrams, colored in red.

3.1.4.4 Instance-based Fingrams

Previously presented Fingrams showed the inference mechanism of FRBSs from a global view point, i.e. observing how all the rules covered the complete given dataset. However, this fails in assisting the analysis of the inference mechanism at local level, i.e. for a single data instance.

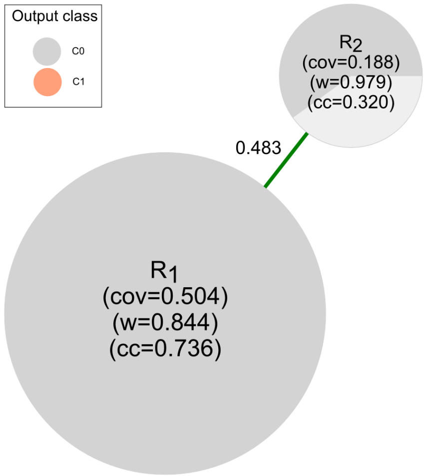


Figure 3.5: Analysis of inference for an instance using instance-based Fingrams

We proposed a new Fingram view aimed at illustrating a partial view of the system when focusing on those rules that participate in the inference process regarding a single instance. Thus, we focus on those rules that specifically cover the related part of the input space. This allows us to better understand the behavior of the system in a specific situation. This *instance-based Fingrams* filter the rules that are not fired by the selected instance. Notice that this task is not easy to carry out manually when dealing with complex sets of rules. Therefore, it provides a powerful filtering mechanism conducted by data.

This new representation is valid for any of the existing types of Fingrams and very valuable when studying instances that require a detailed analysis. The process to build instance-based Fingrams includes four steps:

1. Generation of the network using the desired rule co-firing metric m taking as inputs the whole set of fuzzy rules, the fuzzy inference mechanism, and the entire set of instances.

2. Filtering the network by only considering those rules that take part in the inference for the given instance I_k.

3. Scaling the network through the use of the network scaling method.

4. Graphical representation of the resulting scaled network according to the network drawing method.

The firing degree up to which the instance fires each rule (as seen in Eq. 3.3) is reflected by a radius dark zone in the corresponding node. The angle of dark color is proportional to the firing degree of the rule for the given instance, with a full dark colored node when the firing degree is 1.

We are overviewing this graphical representation with an illustrative example of classification instance-based Fingram. We take a two class dataset ($C0$ and $C1$) covered by four rules, two of each class ($B_1 = B_2 = C0$ and $B_3 = B_4 = C1$). We select an instance ($I_{80} = \{-0.61, 0.25, \mathrm{C0}\}$) to analyze the inference mechanism locally. We follow the building process previously presented, obtaining the Fingram of Figure 3.5.

We can see that the node related to $R1$ is fully dark colored ($\mu_{R_1}(I_{80}) = 1$). On the contrary, R_2 is partially fired ($\mu_{R_2}(I_{80}) = 0.6$), therefore the related node is colored with a radius dark area. Both rules have the same output class (class C0), what is reflected by the same node color. Class C0 is correctly inferred, and there is no rule with class C1 activated in this specific case.

3.2 Cases of use applying the proposed methodology

Fingrams provide an enormous potential for the representation and comprehension of the FRBS inference process. They relate rules jointly fired by a given input vector, allowing to understand how the rules of a FRBS actually cover the input space. Hence, Fingrams can be viewed as a powerful tool for dealing with FRBS comprehensibility analysis tasks related to the semantics at rule base level. It is worthy to note that as presented in Section 1.1, quadrant Q4 at Table 1.1) is the less studied category in the existing fuzzy system interpretability assessment literature.

We proposed a novel interpretability [12] index that enriches Q4 set of indexes. The proportion of co-fired rules can be considered to evaluate the global FRBS comprehensibility. The assumption is the following: the larger the number of simultaneously fired rules for a given input vector, the smaller the comprehensibility of the FRBS.

Thus, the Co-firing Based Comprehensibility Index ($COFCI$) [12] can be used to evaluate the complexity of understanding the inference process in terms of rule co-firing information. Eq. 3.24 presents this index:

$$COFCI = \begin{cases} 1 - \sqrt{\frac{CI}{MaxThr}}, & \text{if } CI \leq MaxThr \\ 0, & \text{otherwise} \end{cases} \tag{3.24}$$

$$CI = \sum_{i=1}^{r} \sum_{j=1}^{r} \left[(P_i + P_j) \cdot m_{ij}^a \right] \tag{3.25}$$

where r is the total number of rules in the fuzzy rule base, P_i and P_j count the number of premises (antecedent conditions) in rules R_i and R_j, while m_{ij}^a is the measure of co-firing (computed by any of the metrics of Eqs. 3.6, 3.8 or 3.10) for the rules R_i and R_j, and $MaxThr$ is a maximum value heuristically established to get a normalized measure in the interval [0,1].

The analysis of Fingrams offers many different possibilities thanks to the high amount of information this representation gives about a FRBS and its related fuzzy inference process. For example, one can directly analyze its global structure by the exploration of the number and location of the apparent groups of rules (nodes), analyze the respective location of the rules coding for different outputs, etc.

Firstly, it should be reminded that, because of the specific way network scaling and drawing are done, the most salient nodes and edges are likely to be placed towards the center of the graphical representation. Therefore, the most frequently fired rules (represented with bigger nodes) are usually placed in the center because they also tend to be co-fired with more rules. Those cases where nodes covering a large number of instances are placed in the periphery must be carefully analyzed. This can be due to a fuzzy rule which covers a large part of the input space in isolation.

Secondly, the interaction among fuzzy rules at inference level is very difficult to be appreciated by only reading the linguistic description of FRBSs. It should be remarked that this interaction depends on the rule description but also on the fuzzy rule semantics (fuzzy partitions included in the data base) and on the inference mechanism. Even when a rule base is apparently consistent at linguistic level, some possible inconsistencies may arise at inference level because of the FRBS semantics and fuzzy inference process. Such potential conflicts are difficult to detect mainly because they are partially hidden since they are typically produced by new unknown situations that were not taken into account during the learning stage (for example, data pairs not initially included when considering a data-driven FRBS derivation). Of course, such analysis is different depending on the kind of problem faced. For example, the meaning of overlapping rules is not the same when considering either classification, regression or association problems.

Thirdly, the information provided by the node of uncovered instances helps in the comprehension of the system and its behavior with respect to the given dataset. It should be noticed that uncovered instances penalize the precision of FRBSs and their early detection and correction is essential for the correct behavior of the system. We can compare node size with the size of the rest of nodes, giving us an idea of how many uncovered instances have.

The rest of the section presents three cases of use where the potentials of the Fingram-based methodology are deeply analyzed.

3.2.1 Case of use on fuzzy classifiers

Here we analyze, through the use of Fingrams, a complex fuzzy rule-based classifier over the real world dataset *ecoli* from UCI [14, 44]. Ecoli dataset includes 336 E.coli proteins of 8 unbalanced classes with 7 attributes calculated from the amino acid sequences.

We use the Fuzzy Unordered Rule Induction Algorithm [45], or FURIA in short, to induce rules and classify data instances. FURIA appears as one of the outstanding fuzzy classification algorithms when attending to accuracy [46], demonstrating to be a robust method and performing properly in a bunch of scenarios.

FURIA is a fuzzy rule-based classification method based on RIPPER algorithm [25]. It presents some modifications and extensions that outperforms the original [45, 46]. FURIA creates very compact FRBSs that achieve high performance thanks to a specific inference mechanism that operates as follows:

- If the instance I_k is covered by the set of induced rules, FURIA predicts the output class:

$$output(I_k) = y'_k \Rightarrow S_{y'_k}(I_k) = \max_{l=1...p} S_{c_l}(I_k) \tag{3.26}$$

  where l iterates over the p possible classes and S_{c_l} represents the activation level per class:

$$S_{c_l}(I_k) = \sum_{R_i \in R^{c_l}} \left(\mu_{R_i}(I_k) \cdot w^i\right) = \sum_{R_i \in R^{c_l}} \left(\mu_{R_i}(I_k) \cdot w^i\right) \tag{3.27}$$

  where $\mu_{R_i}(I_k)$ is the firing degree of rule R_i for instance I_k (as presented in Eq. 3.3) and w^i is the weight associated to rule R_i.

- If I_k is not covered, FURIA dynamically creates a new set of rules (now on SR_k) from the induced ones by the so-called rule stretching mechanism. It iteratively tours every induced rule removing antecedents in order one by one from the least to the most important, until the instance is covered. If all antecedents are removed from an individual rule, then this rule is discarded. When a stretched rule covers the instance I_k then the stretching mechanism stops for this rule, adds that stretched rule to SR_k and goes on with the next rule until all induced rules are checked for the given instance. Therefore, the new rule set SR_k includes at most the number of initially induced rules.

  – If SR_k is empty, i.e. all the rules are discarded: The majority class in the dataset D is taken as output for the instance I_k.

  – Otherwise: FURIA produces as output the winner class with the final set of rules (SR_k) for the instance I_k:

$$output'(I_k) = y'_k \Rightarrow S_{y'_k}(I_k) = \max_{l=1...p} S'_{c_l}(I_k) \tag{3.28}$$

$$S'_{c_l}(I_k) = \max_{R_{i.q} \in R^{c_l}} \left(\mu'_{R_{i.q}}(I_k) \cdot w^{i.q}\right) \tag{3.29}$$

  where

$$\mu'_{R_{i.q}}(I_k) = \begin{cases} \mu_{A_1^i}(x_k^1) \otimes ... \otimes \mu_{A_q^i}(x_k^q) & \text{, if } I_k \in ucv \ \& \ R_{i.q} \in SR_k \\ 0 & \text{, otherwise} \end{cases} \tag{3.30}$$

  and

$$w^{i.q} = w^i \cdot \frac{q+1}{P_i + 2} \tag{3.31}$$

being w^i the weight of the induced rule R_i, P_i the number of antecedents in the induced rule R_i, and q the number of antecedents in the stretched rule $R_{i.q}$.

In case of a tie, no matter if the instance is handled by the set of induced rules or by the stretching mechanism, a decision in favor of the majority class is made. The interested reader can find a deeper explanation of FURIA in [45, 46].

Although FURIA produces compact rule bases, its interpretability is arguable [18], being jeopardized by the absence of linguistic readability. Notice that FURIA usually generates low number of rules (and antecedents per rule), but they lack of linguistic readability because there is no global semantics. Rule antecedents are rule dependent and do not have linguistic terms associated. In addition, FURIA's inference mechanism occludes interpretability. It is based on a winner class method with weighted rules in combination with the rule stretching mechanism. In consequence, FURIA includes a close-to-black-box inference mechanism, very hard to predict and understand.

FURIA produces an accurate fuzzy system for ecoli dataset that outperforms several alternative methods as shown in [45]. It induces the following 20 fuzzy rules from the complete dataset. This is a compact but hard to interpret set of fuzzy rules.

R_1: IF alm1 in $[-\infty, -\infty, 0.38, 0.39]$ & gvh in $[-\infty, -\infty, 0.55, 0.57]$ THEN class is *cp* with w=0.973
R_2: IF mcg in $[-\infty, -\infty, 0.44, 0.52]$ & alm1 in $[-\infty, -\infty, 0.55, 0.58]$ THEN class is *cp* with w=0.951
R_3: IF alm1 in $[-\infty, -\infty, 0.47, 0.49]$ & mcg in $[-\infty, -\infty, 0.59, 0.63]$ & gvh in $[-\infty, -\infty, 0.57, 0.59]$ THEN class is *cp* with w=0.955
R_4: IF alm1 in $[0.75, 0.76, \infty, \infty]$ & mcg in $[-\infty, -\infty, 0.61, 0.62]$ THEN class is *im* with w=0.956
R_5: IF alm1 in $[0.55, 0.61, \infty, \infty]$ & mcg in $[-\infty, -\infty, 0.45, 0.47]$ THEN class is *im* with w=0.951
R_6: IF alm2 in $[0.59, 0.63, \infty, \infty]$ & mcg in $[-\infty, -\infty, 0.74, 0.79]$ & alm2 in $[-\infty, -\infty, 0.73, 0.74]$ & gvh in $[0.45, 0.46, \infty, \infty]$ THEN class is *im* with w=0.904
R_7: IF alm1 in $[0.82, 0.85, \infty, \infty]$ & mcg in $[-\infty, -\infty, 0.74, 0.86]$ & gvh in $[-\infty, -\infty, 0.52, 0.53]$ THEN class is *im* with w=0.902
R_8: IF alm1 in $[0.55, 0.62, \infty, \infty]$ & alm1 in $[-\infty, -\infty, 0.72, 0.74]$ & mcg in $[-\infty, -\infty, 0.61, 0.63]$ & gvh in $[-\infty, -\infty, 0.55, 0.6]$ THEN class is *im* with w=0.916
R_9: IF alm2 in $[0.35, 0.74, \infty, \infty]$ & alm1 in $[-\infty, -\infty, 0.72, 0.73]$ & mcg in $[0.81, 0.83, \infty, \infty]$ THEN class is *im* with w=0.692
R_{10}: IF alm2 in $[0.7, 0.74, \infty, \infty]$ & aac in $[0.7, 0.71, \infty, \infty]$ THEN class is *im* with w=0.615
R_{11}: IF gvh in $[0.58, 0.59, \infty, \infty]$ & aac in $[-\infty, -\infty, 0.47, 0.57]$ & alm1 in $[-\infty, -\infty, 0.65, 0.67]$ & alm1 in $[0.35, 0.36, \infty, \infty]$ THEN class is *pp* with w=0.954
R_{12}: IF gvh in $[0.53, 0.56, \infty, \infty]$ & mcg in $[0.61, 0.63, \infty, \infty]$ & aac in $[-\infty, -\infty, 0.63, 0.65]$ & alm1 in $[-\infty, -\infty, 0.52, 0.53]$ & aac in $[0.45, 0.46, \infty, \infty]$ THEN class is *pp* with w=0.911
R_{13}: IF mcg in $[0.67, 0.7, \infty, \infty]$ & aac in $[-\infty, -\infty, 0.5, 0.51]$ & mcg in $[-\infty, -\infty, 0.74, 0.75]$ THEN class is *pp* with w=0.899
R_{14}: IF alm2 in $[0.39, 0.62, \infty, \infty]$ & mcg in $[0.74, 0.75, \infty, \infty]$ THEN class is *imU* with w=0.800
R_{15}: IF alm2 in $[0.46, 0.66, \infty, \infty]$ & mcg in $[0.58, 0.62, \infty, \infty]$ & gvh in $[-\infty, -\infty, 0.45, 0.46]$ & mcg in $[-\infty, -\infty, 0.67, 0.69]$ THEN class is *imU* with w=0.713
R_{16}: IF alm2 in $[0.73, 0.74, \infty, \infty]$ & alm1 in $[-\infty, -\infty, 0.75, 0.76]$ & mcg in $[0.45, 0.47, \infty, \infty]$ THEN class is *imU* with w=0.581
R_{17}: IF aac in $[0.66, 0.68, \infty, \infty]$ & alm2 in $[-\infty, -\infty, 0.38, 0.66]$ & mcg in $[0.31, 0.52, \infty, \infty]$ THEN class is *om* with w=0.891
R_{18}: IF gvh in $[0.67, 0.68, \infty, \infty]$ & mcg in $[-\infty, -\infty, 0.61, 0.62]$ THEN class is *om* with w=0.687
R_{19}: IF lip in $[0.48, 1, \infty, \infty]$ & alm2 in $[-\infty, -\infty, 0.36, 0.52]$ & chg in $[-\infty, -\infty, 0.5, 1]$ THEN class is *omL* with w=0.719
R_{20}: IF lip in $[0.48, 1, \infty, \infty]$ & aac in $[-\infty, -\infty, 0.51, 0.52]$ & mcg in $[-\infty, -\infty, 0.75, 0.77]$ THEN class is *imL* with w=0.503

Rules involve the complete set of attributes and cover 7 out of the 8 classes in the dataset (there is no rule for the minority class *imS*, which cannot be predicted). Notice that rules R_6, R_8, R_{11}, R_{12}, R_{13}, and R_{15} repeat attributes in their antecedents (attributes *alm2*, *alm1*, *alm1*, *aac*, *mcg* and *mcg* respectively). This fact does not affect the inference performed by the set of induced rules but is transcendental in the stretching mechanism.

Table 3.1 summarizes the information of the set of rules induced by FURIA –rule identifiers R_i, rule output class B_i, coverage of rules cov_i (Eq. 3.11), purity of rules pur_i (Eq. 3.14), weights of rules w^i and coverage of rule output class cc_i (Eq. 3.15).– Last line in the table indicates the amount and output class of instances not covered by any of the induced rule, which take advantage of the stretching mechanism.

Table 3.1: Information of the set of classification rules induced for Ecoli dataset.

R_i	B_i	cov_i	pur_i	w^i	cc_i
R_1	cp	0.339	0.535	0.973	0.427
R_2	cp	0.393	0.485	0.951	0.469
R_3	cp	0.405	0.502	0.955	0.476
R_4	im	0.098	0.030	0.956	0.013
R_5	im	0.092	0.136	0.951	0.052
R_6	im	0.042	0.071	0.904	0.013
R_7	im	0.045	0.000	0.902	0.013
R_8	im	0.051	0.113	0.916	0.026
R_9	im	0.012	0.000	0.692	0.013
R_{10}	im	0.009	0.000	0.615	0.013
R_{11}	pp	0.113	0.029	0.954	0.019
R_{12}	pp	0.057	0.000	0.911	0.019
R_{13}	pp	0.057	0.000	0.899	0.019
R_{14}	imU	0.065	0.182	0.800	0.114
R_{15}	imU	0.015	0.941	0.713	0.114
R_{16}	imU	0.057	0.053	0.581	0.029
R_{17}	om	0.051	0.000	0.891	0.050
R_{18}	om	0.015	0.000	0.687	0.050
R_{19}	omL	0.018	0.000	0.719	0.200
R_{20}	imL	0.009	0.000	0.503	0.500
Instances Stretching	cp/im/pp/imU	0.039			

Figure 3.6 presents the Fingrams[2] depicting the induced rule set built with m^1_{ij} co-firing metric (Eq. 3.8). In the top left square we have included the same Fingram built with m^0_{ij} co-firing metric (as presented in Eq. 3.6) just for comparison purpose. In this case, both Fingrams present quite similar structure but weaker edges appear when using m^1_{ij} metric. Thus, we can analyze the system by studying any of them interchangeably. Therefore, we continue the analysis by regarding the Fingram built with m^1_{ij} metric.

The Fingram in Figure 3.6 includes one node per rule. These nodes take 7 different colors (out of the 8 possible, as presented at the bottom right legend) showing the 7 different classes the rules have as output. An additional multi-color node presents the instances not covered by any of the induced rules, i.e. the instances to be handled by the stretching mechanism.

Studying the structure of the Fingram in detail we can see that almost all the edges have low values meaning that rules cover few instances in common. Only rules of class *cp* (rules R_1, R_2 and R_3), colored in gray, cover several instances in common, showing high relations among them. This indicates that rules induced by FURIA cover instances

[2]Contrary to the previous Fingrams, those in Figs. 3.6 and 3.7 only include the rule identifiers in the nodes for the sake of clarity. Table 3.1 and Table 3.2 include additional information about them.

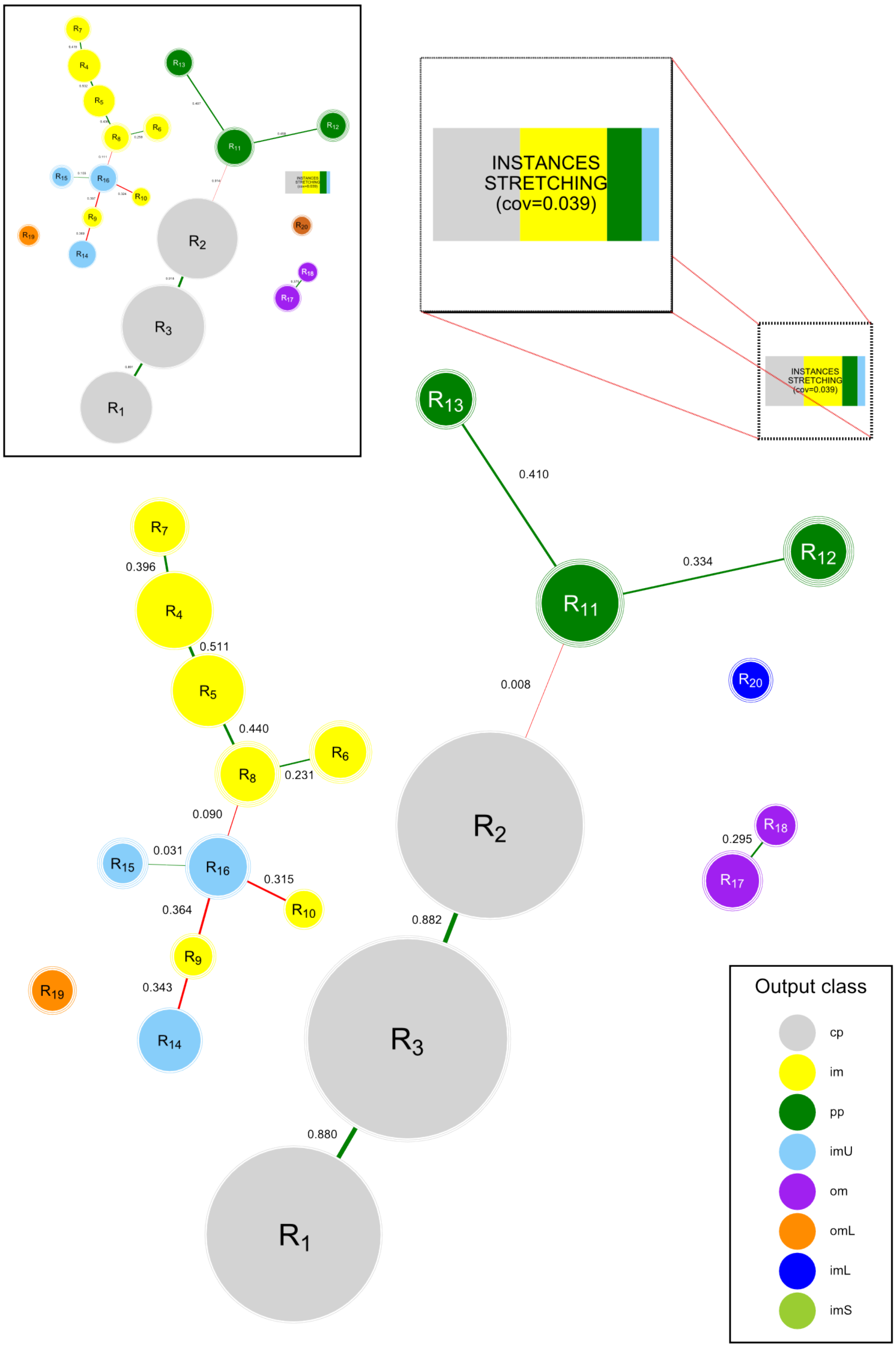


Figure 3.6: Fingram of the set of classification rules induced for Ecoli dataset.

scattered and as result most of the instances are just covered by a single rule. Moreover, it is easy to appreciate clusters of nodes of the same color showing that rules with different output class are rarely related. This is also reflected by the majority of green edges in the representation. This particularity indicates that rules of the same class jointly cover parts of the input space.

Focusing our attention in subsets of rules we observe that rules of class *cp* cover a large amount of instances (coverage larger than 0.3 as seen in Table 3.1) with a good ratio of correctly classified instances (purity larger than 0.48). Even more, they cover several instances in common, as previously mentioned. Notice that *cp* is the majority class in the dataset. On the contrary, rules R_{19} and R_{20} cover just a few instances, and not in accordance with their corresponding class, as we can find out observing that they have purity equal to zero. It is worthy to note that both rules are in charge of handling two of the minority classes (*omL* and *imL*). In an intermediate situation we find rules with output classes *im* and *imU* (yellow and blue nodes) that only cover a few instances in common and have a diverse range of purity values. This situation occurs when instances of two classes are spread along the same part of the input space and turn difficult the classification task.

As previously mentioned, a non negligible part of the instances are fired by none of the 20 induced rules. This information is given by the node labeled as "INSTANCES STRETCHING" (zoomed in the top right square to appreciate its details) which shows that 13 instances (3.9% of the total) were not covered, and their corresponding class distribution is depicted in color sector areas (5 of class *cp*, 5 of *im*, 2 of *pp*, and 1 of *imU*). These instances trigger the stretching mechanism.

18 rules were dynamically generated by the stretching mechanism to deal with the 13 uncovered instances. Table 3.2 shows the information of those stretched rules –rule identifiers $R_i.q$ where R_i is the original rule from the rule derives and q the number of antecedents kept, Antecedents shows the rule antecedents of each rule. In addition, the table includes rule output class B_i, coverage of rules $cov_{i.q}$ (Eq. 3.11), purity of rules $pur_{i.q}$ (Eq. 3.14), weight of rules $w^{i.q}$ (Eq. 3.31), and coverage of rule output class $cc_{i.q}$ (Eq. 3.15).– The last line in the table shows the amount and output class of instances even not covered by the stretching mechanism.

Figure 3.7 presents the Fingram representing the set of stretched rules. Metric m^1_{ij} yields the principal Fingram in the figure whereas Fingram in the bottom left square was constructed with metric m^0_{ij}. We observe that metric m^1_{ij} better avoid subsets of highly connected nodes and it allows an easier analysis.

The set of stretched rules covers 5 classes (classes *cp*, *im*, *pp*, *imU*, and *om*) out of the 8 classes in the dataset, even though the distribution of uncovered instances in Figure 3.6 was related to only four of such classes. Notice that class *om* was not among instances for stretching.

FURIA builds a high number of rules (18 rules) to deal with a few instances (13 instances). These rules are quite specific and they cover very few instances (rules $R_{1.1}$, $R_{3.1}$ or $R_{7.2}$ just one instance each) while each single instance is usually covered by several fuzzy rules. This particularity of the stretching mechanism occludes the interpretability of FURIA.

The structure of the Fingram of stretched rules is more complex than the previous. All the nodes are connected and the edges present higher values meaning that the rules

Table 3.2: Information of the set of stretched rules for Ecoli dataset.

$R_i.q$	Antecedents	B_i	$cov_{i.q}$	$pur_{i.q}$	$w^{i.q}$	$cc_{i.q}$
$R_{1.1}$	alm1 in $[-\infty, -\infty, 0.38, 0.39]$	cp	0.006	0.000	0.486	0.007
$R_{3.1}$	alm1 in $[-\infty, -\infty, 0.47, 0.49]$	cp	0.006	0.000	0.382	0.007
$R_{4.1}$	alm1 in $[0.75, 0.76, \infty, \infty]$	im	0.018	0.167	0.478	0.013
$R_{5.1}$	alm1 in $[0.55, 0.61, \infty, \infty]$	im	0.006	1.000	0.475	0.013
$R_{6.2}$	alm2 in $[0.59, 0.63, \infty, \infty]$ & mcg in $[-\infty, -\infty, 0.74, 0.79]$	im	0.018	0.167	0.452	0.013
$R_{7.2}$	alm1 in $[0.82, 0.85, \infty, \infty]$ & mcg in $[-\infty, -\infty, 0.74, 0.86]$	im	0.006	1.000	0.541	0.013
$R_{9.2}$	alm2 in $[0.35, 0.74, \infty, \infty]$ & alm1 in $[-\infty, -\infty, 0.72, 0.73]$	im	0.006	0.000	0.415	0.013
$R_{11.3}$	gvh in $[0.58, 0.59, \infty, \infty]$ & aac in $[-\infty, -\infty, 0.47, 0.57]$ & alm1 in $[-\infty, -\infty, 0.65, 0.67]$	pp	0.006	1.000	0.636	0.019
$R_{11.1}$	gvh in $[0.58, 0.59, \infty, \infty]$	pp	0.009	0.000	0.318	0.019
$R_{12.4}$	gvh in $[0.53, 0.56, \infty, \infty]$ & mcg in $[0.61, 0.63, \infty, \infty]$ & aac in $[-\infty, -\infty, 0.63, 0.65]$ & alm1 in $[-\infty, -\infty, 0.52, 0.53]$	pp	0.006	1.000	0.651	0.019
$R_{12.3}$	gvh in $[0.53, 0.56, \infty, \infty]$ & mcg in $[0.61, 0.63, \infty, \infty]$ & aac in $[-\infty, -\infty, 0.63, 0.65]$	pp	0.009	0.000	0.520	0.019
$R_{12.2}$	gvh in $[0.53, 0.56, \infty, \infty]$ & mcg in $[0.61, 0.63, \infty, \infty]$	pp	0.009	0.000	0.390	0.019
$R_{13.1}$	mcg in $[0.67, 0.7, \infty, \infty]$	pp	0.012	0.273	0.360	0.019
$R_{14.1}$	alm2 in $[0.39, 0.62, \infty, \infty]$	imU	0.021	0.000	0.400	0.029
$R_{15.3}$	alm2 in $[0.46, 0.66, \infty, \infty]$ & mcg in $[0.58, 0.62, \infty, \infty]$ & gvh in $[-\infty, -\infty, 0.45, 0.46]$	imU	0.009	0.000	0.475	0.029
$R_{15.2}$	alm2 in $[0.46, 0.66, \infty, \infty]$ & mcg in $[0.58, 0.62, \infty, \infty]$	imU	0.015	0.000	0.356	0.029
$R_{17.1}$	aac in $[0.66, 0.68, \infty, \infty]$	om	0.006	0.000	0.356	0.050
$R_{18.1}$	gvh in $[0.67, 0.68, \infty, \infty]$	om	0.012	0.000	0.343	0.050
Uncovered instances		im	0.006			

densely cover the input space. Most of those edges correspond to inconsistencies, i.e. they relate rules with different output class.

The special node that is labeled as "UNCOVERED INSTANCES" shows that there are some instances which remain uncovered even after running the rule stretching mechanism. In such case, the inference mechanism produces as output the majority class, *cp*.

Finally, we study in detail the inference mechanism for a couple of instances. This way we show the benefits of considering instance-based Fingrams to locally view the FRBS inference mechanism. We graphically observe the rules that participate in the inference process, understanding the behavior of the system in a specific situation.

Firstly, Figure 3.8(a) presents the instance-based Fingram for an instance covered by the set of induced rules (instance $I_{321} = \{0.68, 0.67, 0.48, 0.5, 0.49, 0.4, 0.34, pp\}$). This instance only fires rules R_{11}, R_{12} and R_{13} with level of firing 0.80, 1.00 and 0.33 respectively as shown by the colored sectors in the picture. The three rules have the

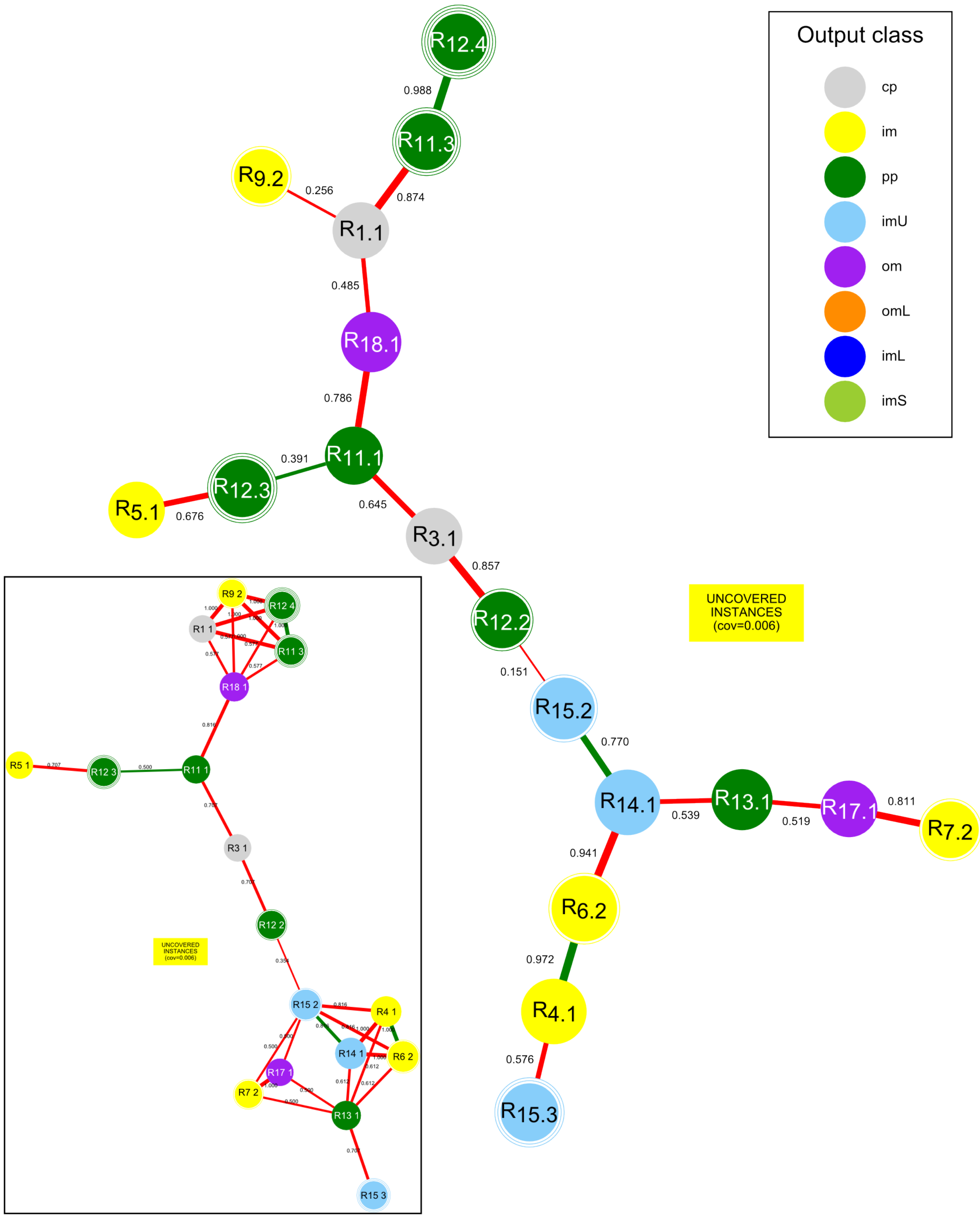
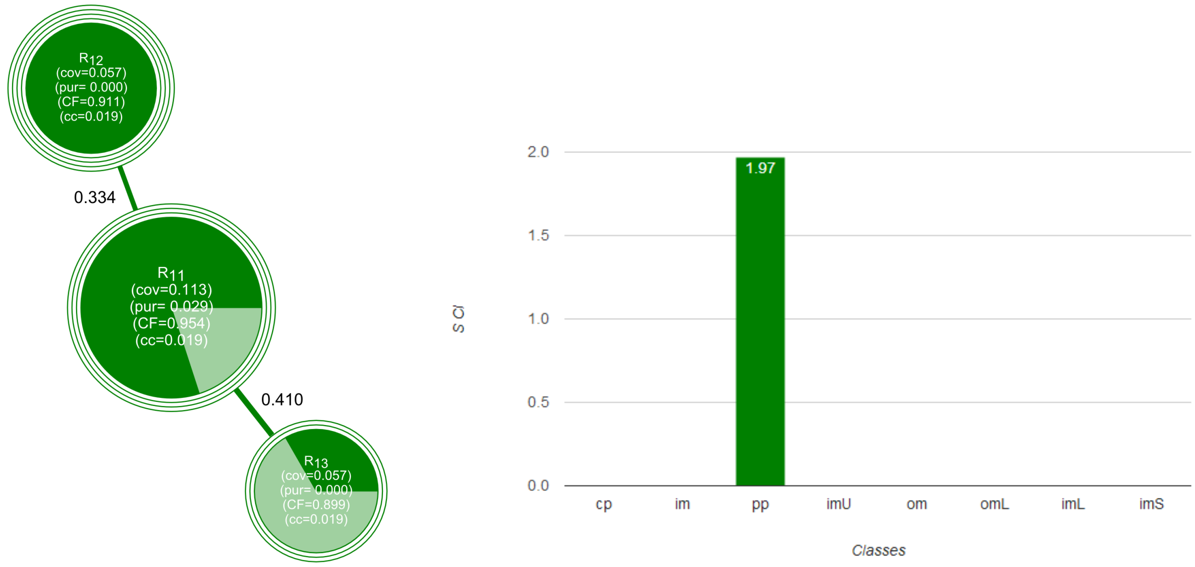


Figure 3.7: Fingram of the set of stretched rules for Ecoli dataset.

same output class *pp*, therefore the three related nodes have the same green color in the representation. The level of S_{c_l} (Eq. 3.27) for the given instance is presented in the bar chart of Figure 3.8(b). We clearly see that the system correctly infers class *pp*.

Secondly, we selected an instance that is not covered by the set of induced rules (instance $I_{211} = \{0.69, 0.39, 0.48, 0.5, 0.57, 0.76, 0.79, im\}$). In consequence, the stretching mechanism is run as part of the inference process. I_{211} is handled by the stretched rules $R_{4.1}$, $R_{6.2}$, $R_{13.1}$, $R_{14.1}$ and $R_{15.3}$, as seen in Figure 3.9(a). In this case, all but rule $R_{13.1}$ are

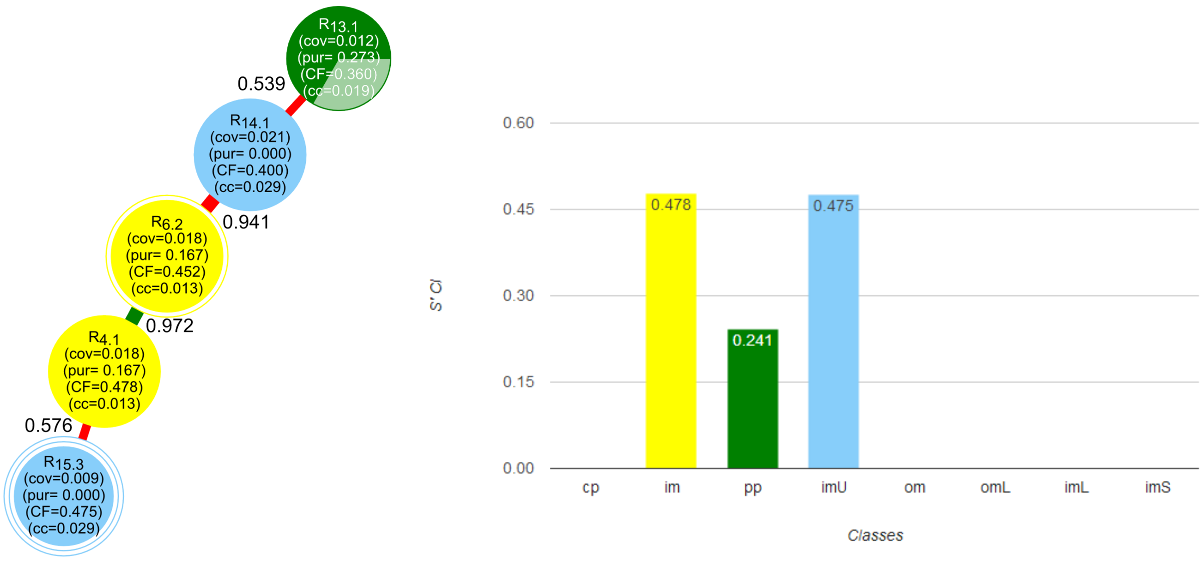


(a) Instance-based Fingram. (b) Levels of S_{C_l}.

Figure 3.8: Analysis of inference for $I_{321} = \{0.68, 0.67, 0.48, 0.5, 0.49, 0.4, 0.34, pp\}$ in the Ecoli classification problem.

fired to level 1 ($\mu_{R_{13.1}}(I_{211}) = 0.67$). The stretching mechanism correctly inferred class im because it is the winner class for S'_{c_l} (Eq. 3.29) as can be seen in Figure 3.9(b). Anyway, we observe that $S'_{im} \approx S'_{imU}$ what is not desirable because it may produce ambiguity since a small variation in the input may incorrectly infer imU as output class.

The interested reader can find additional examples on the of Fingrams on fuzzy classifiers at [65, 68].



(a) Instance-based Fingram. (b) Levels of S'_{C_l}.

Figure 3.9: Analysis of inference for $I_{211} = \{0.69, 0.39, 0.48, 0.5, 0.57, 0.76, 0.79, im\}$ in the Ecoli classification problem.

3.2.2 Case of use on fuzzy regressors

When dealing with regression problems, the well-known FRBS approximation capability is mainly based on the interpolative reasoning carried out among overlapping rules. Typically, two rules with similar premises may yield two different wrong outputs but their aggregation may result in the right inferred interpolated output. Unfortunately, this kind of situations are quite common but very difficult to identify. Of course, from the comprehensibility point of view it would be desirable to have only one rule that directly yields the right inferred output. However, this may produce a huge number of rules what is also undesirable. Fingrams permit the analysis of such behavior as we will overview in this section.

We selected an electrical network distribution problem [26, 27] to be analyzed. The system aims to estimate the length of the low voltage line installed in a certain village. The problem has two input variables (the *population of the village* and its *radius*) and one output variable (the *total length of the installed line*). Real data of 495 villages are available. The training set contains 396 elements and the test set includes 99 elements, randomly selected from the whole sample, taken from KEEL dataset repository[3]. Here we will use just the training set to create the Fingrams thus being able to compare the accuracy results with previous works.

First of all, the problem variables are partitioned using strong fuzzy partitions as presented in [65]. The partitions of the input variables (*Inhabitants* and *Distance*) are tuned to improve the performance, while the output variable is partitioned homogeneously covering the interest range, i.e., the range where problem instances are located. Using these fuzzy partitions along with the Fuzzy Prototype Algorithm FPA[42] the following set of rules is generated:

R_1: IF Distance is Very Low THEN Length is Very Low

R_2: IF Inhabitants is (Very Low OR Low OR Average) AND Distance is Low THEN Length is Low

R_3: IF Inhabitants is Very Low AND Distance is Average Low THEN Length is Low

R_4: IF Inhabitants is (Low OR Average) AND Distance is Average Low THEN Length is Average Low

R_5: IF Inhabitants is High AND Distance is Low THEN Length is Average Low

R_6: IF Inhabitants is (Very Low OR Low) AND Distance is Average High THEN Length is Average

R_7: IF Inhabitants is Very High AND Distance is Average Low THEN Length is Average

R_8: IF Inhabitants is Average AND Distance is (Average High OR High) THEN Length is Average High

R_9: IF Inhabitants is Very High AND Distance is Average High THEN Length is High

R_{10}: IF Inhabitants is Very High AND Distance is High THEN Length is Very High

[3] *http://sci2s.ugr.es/keel/datasets.php*

This FRBS exhibits a good accuracy [4] ($MSE = 130,045$), similar to the one obtained in [26] ($MSE = 133,763$). Anyway, we should again remind that we are not focused on finding the most accurate FRBS for the tackled problem. Our target is showing the utility of Fingrams in the context of a real-world regression problem.

As explained previously in Section 3.1.4.2, the output of each fuzzy rule will be reflected in the color of the nodes. From dark to light the node colors represent a range from low to high values, and red nodes indicate rules not covering any instance. So for, the output label *"Very Low"* will be represented by the darkest node while *"Very High"* corresponds to the lightest one close to white. Naturally, the system will have relations among close labels and close colors, and when nodes of quite different darkness are related the expert should focus her/his attention on them.

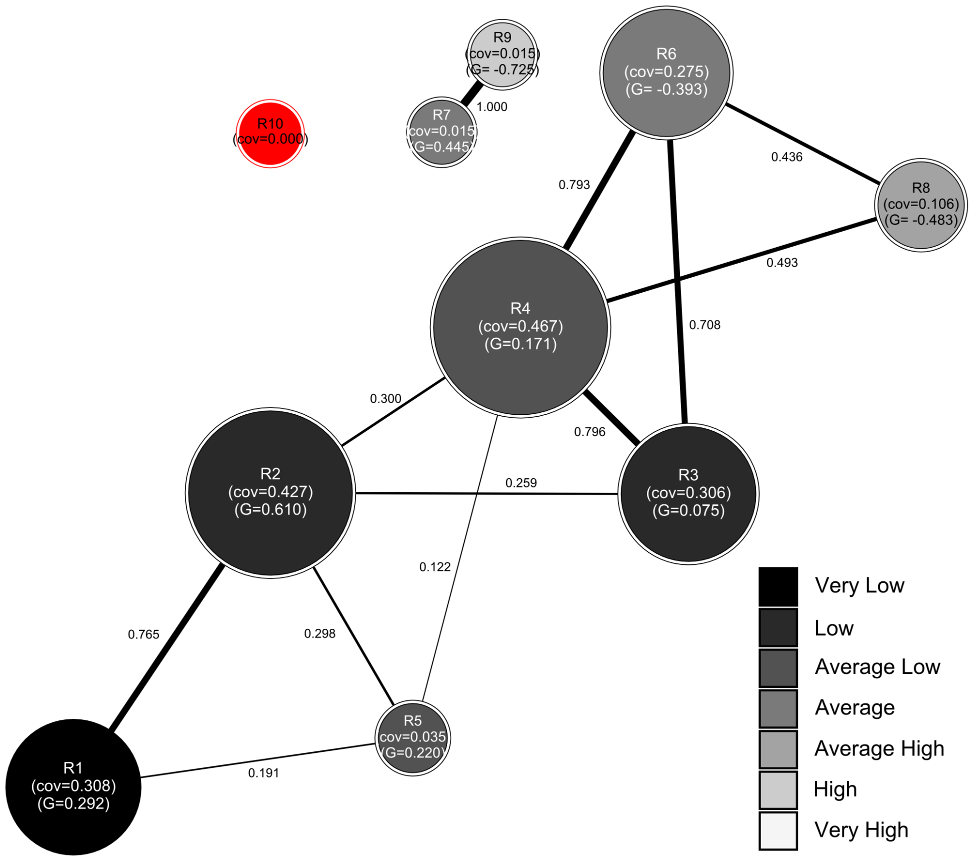


Figure 3.10: Complete Fingram for the electrical distribution problem.

Figure 3.10 shows the non-scaled Fingram constructed using metric m^0_{ij} for the FRBS previously presented. It can be seen that the two dimensions and the use of strong fuzzy

[4] Accuracy is computed as the mean square error (MSE).

$$MSE = \frac{1}{d}\sum_{k=1}^{d}(y_k - \hat{y}_k)^2 \tag{3.32}$$

where d means the number of problem instances, y_k is the real output value of instance I_k, and $\hat{y}_k$ is the inferred output by the FRBS.

partitions allow the Fingram to spread the nodes in a grid, relating close outputs, i.e the evolution of darkness of the nodes is mapped smoothly. Rules R_2 and R_4 are quite general, covering almost half of the problem instances ($cov_2 = 0.427$ and $cov_4 = 0.467$). Contrary, rules R_5, R_7, R_9 and R_{10} cover a small amount of problem instances, thus being very specific. Moreover, rule R_{10} does not cover any instance ($cov = 0$), easily appreciated by the red node, and thus it can be eliminated without any accuracy loss. In addition, all rules but R_1 have two antecedents, as it is appreciated in the single-line border of the nodes.

The Fingram analysis lets us discover a special relation between rules R_7 and R_9 that appear isolated in a group, composing a kind of "fuzzy rule cluster" in a specific problem domain region. They cover instances that no other rule covers. Moreover, they cover exactly the same instances ($m^0_{79} = 1.0$) but having different outputs. Even more, rule R_9 has a negative goodness, -0.725, so for it is a candidate to be removed, adjusting, if necessary, the output of R_7. A thorough analysis of these rules is required to avoid this kind of behavior. Notice that only looking R_7 and R_9 at linguistic level is not enough for detecting this kind of potential problems, but our Fingram-based analysis methodology allows us to quickly identify them.

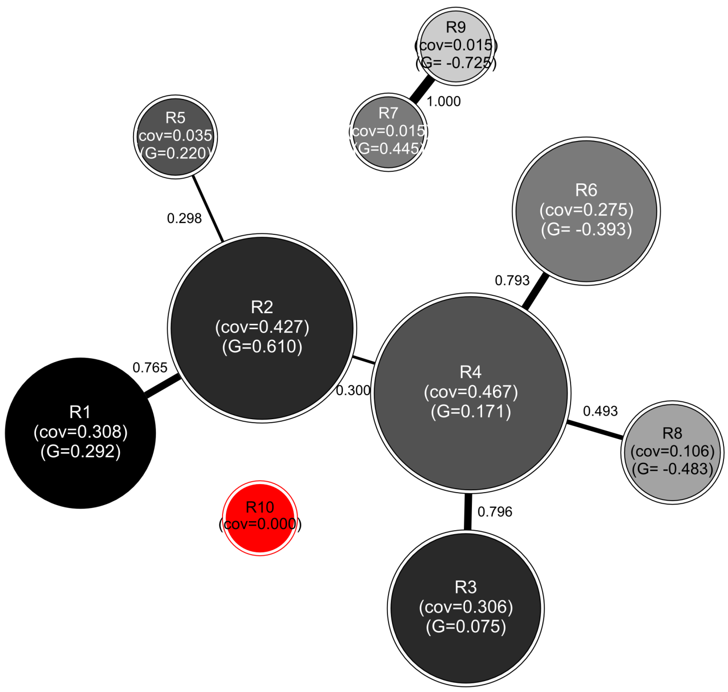


Figure 3.11: Fingram scaled with Pathfinder for the electrical distribution problem.

Figure 3.11 shows the network corresponding to the Fingram scaled with Pathfinder. It emphasizes a high relation among rules R_3, R_4, and R_6 showing that they cover many instances in common. This interrelation suggests to merge the three rules in a single one. To do so, a new rule, R_{346}, is constructed from R_3, R_4, and R_6 in an expert way. The antecedents of these rules are combined and the output is taken from the middle term.

This is done just as an illustrative example, and a more complex process, testing the alternatives, could be done.

R_3:	IF Inhabitants is Very Low	AND Distance is Average Low	THEN Length is Low
R_4:	IF Inhabitants is (Low OR Average)	AND Distance is Average Low	THEN Length is Average Low
R_6:	IF Inhabitants is (Very Low OR Low)	AND Distance is Average High	THEN Length is Average
R$_{346}$:	**IF Inhabitants is (Very Low OR Low OR Average)**	**AND Distance is (Average Low OR Average High)**	**THEN Length is Average Low**

We developed the suggested changes in a sequential fashion (i.e., first removing R_{10}, then removing R_9, and finally merging R_3, R_4, and R_6) and then we checked how they affect the resulting FRBS accuracy and interpretability (see Table 3.3). As was pointed out in Section 1.1, taking only one index is not enough to evaluate interpretability. Therefore, we have considered some of the interpretability indices that are commonly used in the literature. Probably, the most popular index is the number of rules (NR). As an alternative, the total rule length (TRL) represents the total number of linguistic propositions into the whole rule base. Another simple index is average rule length (ARL), computed as $ARL = TRL/NR$. We will also report the average number of fired rules with respect to problem instances (AFR).

Table 3.3: Quality evaluation of the FRBSs proposed for the electrical distribution problem.

Quality index	*Original FRBS*	*R_{10} removal*	*R_9 removal*	*R_3-R_4-R_6 fusion*
MSE	130,045.827	130,045.827	**125,510.863**	155,838.13
NR	10	9	**8**	6
TRL	19	17	**15**	11
ARL	1.9	1.889	**1.875**	1.83
AFR	2.463	2.463	**2.446**	1.695
$COFCI$	0.971	0.971	**0.974**	0.981

Analyzing the reported results we can conclude that the removal of R_{10} does not change the behavior of the system because, as mentioned, it does not cover any problem instance. Thus, MSE, AFR and $COFCI$ remain the same while the interpretability indices related to transparency (NR, TRL and ARL) are improved. However, deleting the rule R_9 simplifies the FRBS improving both accuracy (MSE decreases) and interpretability (all the considered interpretability indices get better values). The new Fingram resulting from these two eliminations can be observed in Figure 3.12.

Although the fusion of R_3, R_4, and R_6 reduces the accuracy of the FRBS, it could still be a good option to get a more compact and understandable FRBS (notice that, all the interpretability indices are clearly improved). Besides, a more elaborated rule fusion mechanism could be considered by the expert to minimize the accuracy loss.

The interested reader can find additional details about the use of Fingrams in regression problemas at [65].

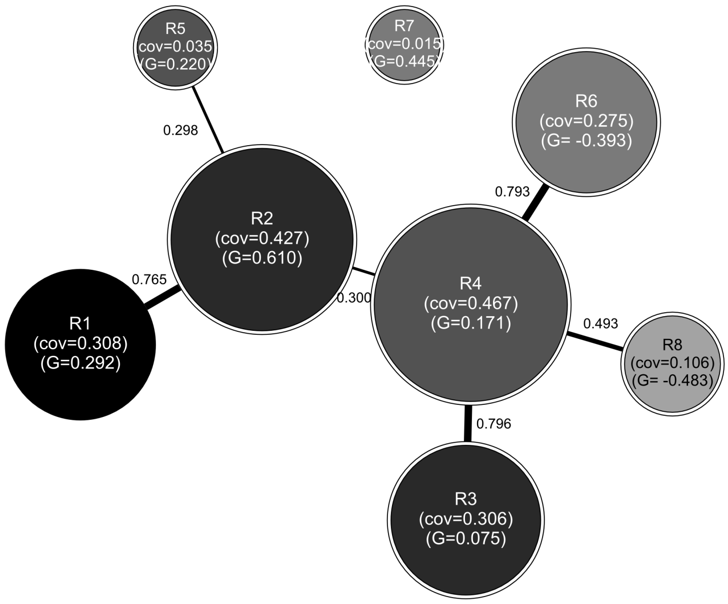


Figure 3.12: Fingram of the best simplified FRBS for the electrical distribution problem.

3.2.3 Case of use on fuzzy association rules

The analysis of Fingrams for fuzzy association rules (FAR-Fingrams) permits the detection of some common behaviors in groups of this kind of rules. The way how rules are connected to each other gives an idea of the complexity and interrelation among them. Sparse rules are usually easier to understand, whereas densely connected rules are more complex to comprehend. However, there may be cases that produced complete subgraphs with very tight relations what reflects a common behavior. On the one hand, highly related rules are candidates to be merged into a more general one because they normally share most of the antecedents and consequents. Those are easily recognized in Fingrams because they usually produce very dense structures. On the other hand, isolated sets of rules covering disjoint sets of instances appear like islands. Those connected rules inside the island do not share any instance with other external rules.

The utility of FAR-Fingrams is illustrated in a real-world problem dealing with qualitative assessment of industrial objects automatically designed through cognitive engineering, in the context of Quale® research line[5]. Namely, we focused on finding out the most interesting fuzzy association rules related to explain how different users evaluated the degree of femininity of a set of chairs. We considered data coming from a project[6] where people had to evaluate the femininity degree of 23 models of chairs. They were sequentially displayed, in a poll, allowing the users to introduce their appreciations in

[5]Quale® is a research line of the European Centre for Soft Computing that supports the human-centered design of customized products/services, considering the analysis and quantification of qualitative assessments. More info at: *http://www.softcomputing.es/quale/*.

[6]Project "Development of a Fuzzy Inferential System for the Quantification of Intangible Assets" at the European Centre for Soft Computing in collaboration with Vortica. More info at: *http://bit.ly/18ZsEt7*

turn (see two examples of chairs in Figure 3.13). The poll received 644 evaluations from 28 users (11 males and 17 females).

(a) Model with femininity "high".

(b) Model with femininity "low".

Figure 3.13: Examples of chairs used in the poll related to the quantification of intangible assets.

It is possible to relate the femininity degree associated to each chair with its physical properties. With that aim, once analyzed all collected data, we decided to induce fuzzy association rules from the aggregated answers provided by different groups of users. In this section, for the sake of clarity we will discuss only the analysis for the group of women who participated in the poll. Notice that, we use this real case study just as an illustrative example, giving an overview of the potentials of FAR-Fingrams. The main goal is to identify and analyze subsets of fuzzy association rules from an expert analysis point of view.

The learning algorithm used [3] extracted both membership functions (MFs) and fuzzy association rules for the given dataset. It tackles with quantitative values by means of a genetic learning of the MFs based on the 2-tuples linguistic representation model and the use of a basic method for mining the fuzzy association rules. The initial linguistic partitions are comprised by 5 linguistic terms with uniformly distributed triangular MFs. The parameters of the algorithm were selected according to the recommendations of the authors, which are the default parameter settings included in the KEEL software tool [4]. Notice that, in this case we have used 0.25 and 0.9 for the minimum support and minimum confidence, respectively.

As result, we generated 31 rules that relate the variables *Femininity*, *Distance between legs*, *Distance between armrests*, *Distance from the seat to the ground*, *Type of base*, and *Type of structure*. Then, we built FAR-Fingrams using metric m_{ij}^2 (as presented in Eq. 3.10) to represent and analyze the set of fuzzy association rules. We looked for the most interesting rules, thus illustrating the potentials of FAR-Fingrams.

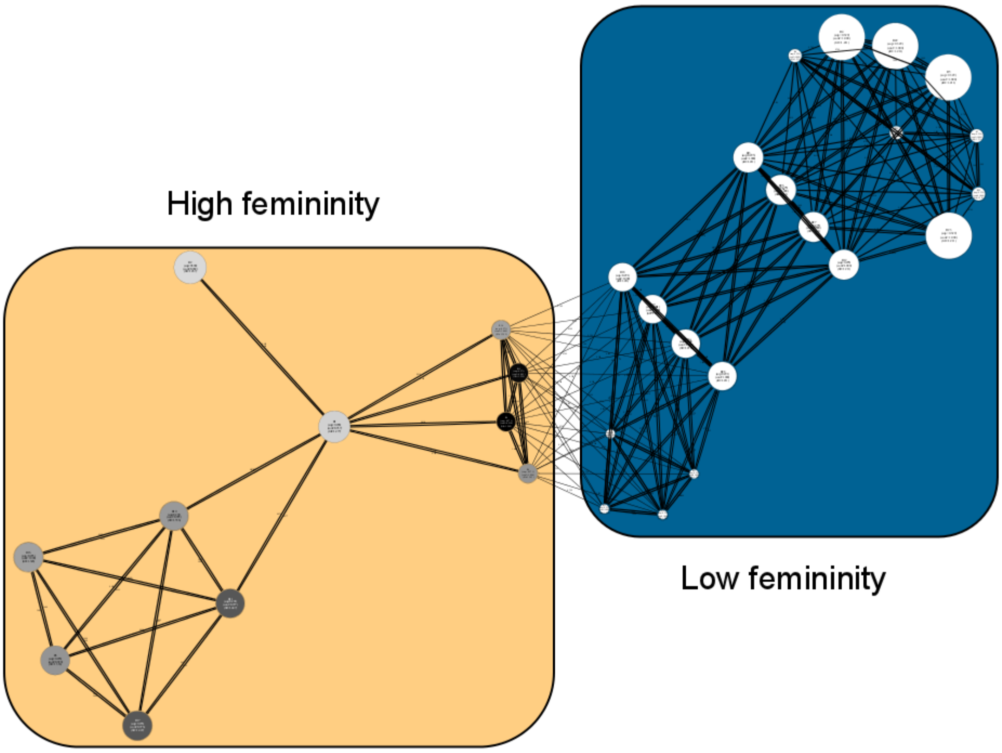
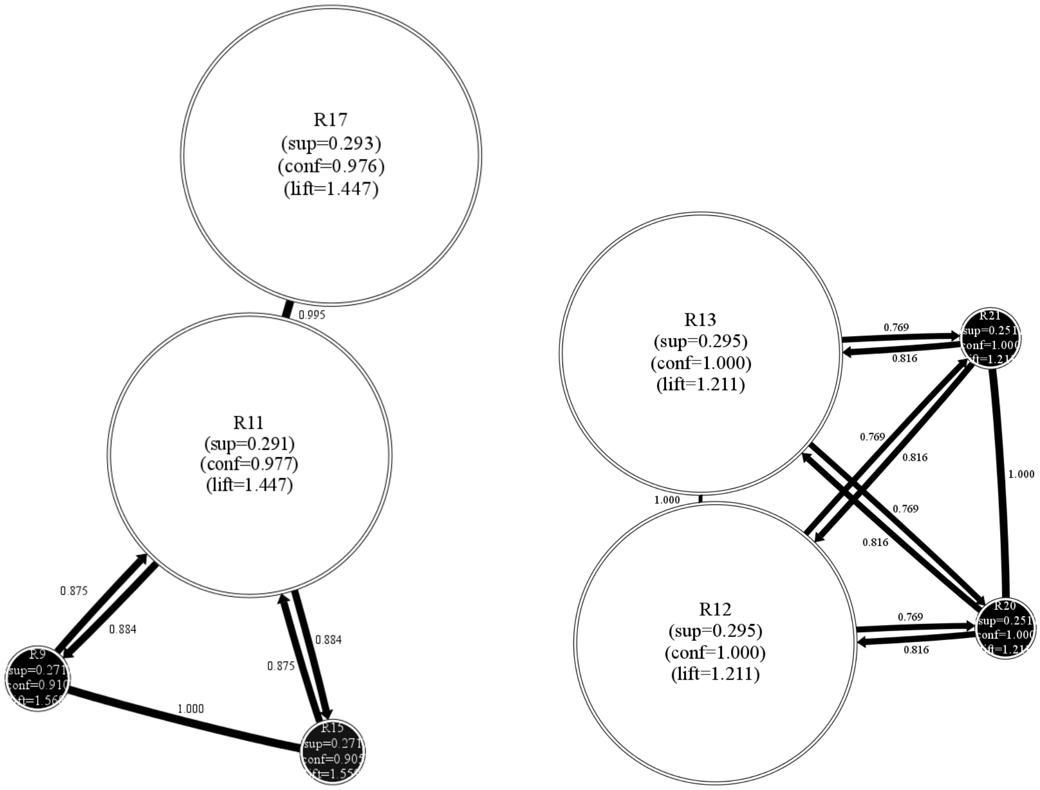


Figure 3.14: FAR-Fingram for the femininity of chairs assessment problem.

First of all, we constructed the complete FAR-Fingram regarding all the 31 rules (Figure 3.14). The structure of this Fingram reflects a clear separation among two groups of rules, those dealing with high femininity (left hand side of the figure) and those corresponding to low femininity (right hand side). Notice that weights of edges connecting rules inside each group are much greater than the weights of those edges connecting rules belonging to the two distinct groups. This is something really close to the so-called community mining in social network analysis. A group of nodes forms a community when inner connections among group members are stronger than outer connections with members of other groups. So, we can say that the two identified groups of rules form two well defined communities according to social network analysis. Moreover, rules referring to high femininity have higher *Lift* too. This fact is reflected with darker nodes.

Then, we conducted a detail analysis of each community in the quest for the most interesting rules regarding high or low femininity. In both cases, we discarded rules with more than 2 antecedents (giving priority to more general and shorter rules, from the interpretability viewpoint) and with lower *Lift*. We actually discarded those rules with *Lift* under the thresholds 1.44 and 1.21 in high and low femininity rules respectively. Figs. 3.15(c) and 3.15(d) show the resultant FAR-Fingrams. They include the rule descriptions at the bottom.

Figure 3.15(c) permits appreciating that rules R9 and R15 cover the same instances with the same levels of firing (the related edge weight equals 1.0). Looking carefully at the rule description, it is easy to deduce that variable *Distance between legs* is not changing



R9: {Distance between legs is Very Low AND Femininity is High} $\rightharpoonup$ {Distance between armrests is Very Low}
R11: {Distance between legs is Very Low AND Femininity is High} $\rightharpoonup$ {Distance between armrests is Low}
R15: {Distance between legs is Low AND Femininity is High} $\rightharpoonup$ {Distance between armrests is Very Low}
R17: {Distance between legs is Low AND Femininity is High} $\rightharpoonup$ {Distance between armrests is Low}

(c) High femininity.

R12: {Distance between legs is Very Low AND Femininity is Low} $\rightharpoonup${Base is Traditional}
R13: {Distance between legs is Very Low AND Femininity is Low} $\rightharpoonup$ {Structure is Geometric lines}
R20: {Distance between armrests is Low AND Femininity is Low} $\rightharpoonup$ {Base is Traditional}
R21: {Distance between armrests is Low AND Femininity is Low} $\rightharpoonup$ {Structure is Geometric lines}

(d) Low femininity.

Figure 3.15: Filtered FAR-Fingrams for the femininity of chairs assessment problem.

the firing degree of the handled instances. Moreover, R9 and R15 have lower *Support* (smaller nodes) and higher *Lift* (darker nodes) with respect to rules R11 and R17. In addition, we can see that all the rules are very similar and highly related. Therefore, we can look for the most interesting one considering measures of *Support*, *Confidence* and *Lift*, and highlight our selection. To do so, we have remarked, in bold, R17.

A dual analysis of Figure 3.15(d) leads us to highlight R12 and R13 as the most interesting rules among those ones related to low femininity. Moreover, pairs of rules R12–R13 and R20–R21 are covering exactly the same instances. Paying attention to the symmetrical structure in Figure 3.15(d) we see that R12 and R13 are equivalent. They emphasize a strong relation between low femininity and both traditional base and structure with geometric lines.

The interested reader can find additional information about FAR-Fingrams and the femininity of chairs assessment problem at [63, 64].

3.3 Software Implementations

This section presents the software designed and implemented during this dissertation to work with the Fingram-based methodology.

3.3.1 Fingrams Generator

Fingrams Generator software is a command-line tool that allows Fingram generation and analysis. It has been designed as a stand-alone tool which can be invoked from any fuzzy design software. With this solution, we encourage developers of fuzzy system software to enhance their own tools by adding modules able to take profit of the Fingrams produced by this tool. Fingrams Generator is freely available as open source software at *https://sourceforge.net/projects/fingrams/*.

This tool requires a *.fs* input file with all demanded information about the designed fuzzy system, such as the set of rules, the instances covered by each rule, the set of uncovered instances, and so on. More concisely, Figure 3.16 details the structure of such configuration file.

- The first line should contain the type of rules to construct Fingrams from (Classification, Regression or Association).

```
Classification/Regression/Association
C1(100),C2(50), .../Low(20), Medium(10), .../Vble_1(LABEL_0,LABEL_1), Vble_2(LABEL_0), ...

Source: GUAJE/FURIA/KEEL/...
Blank threshold: 0.1
Goodness Threshold (high): 0.5
Goodness Threshold (low): 0.1

Rules: 11
Instances: 200

Rule1: If Vble_1 is Low THEN Class is C1 (W=1)
Correct Instances => (22.93300) (0.63703) => 98(0.333), 99(0.264), 100(0.123), ...
Incorrect Instances => (4.89900) (0.22268) => 17(0.109), 21(0.235), 47(0.212), ...
Positive Instances  => (20.47700) (0.68257) => 98(0.333), 100(0.123), 102(0.421), ...
Negative Instances  => (4.89900) (0.22268) => 17(0.109), 21(0.235), 47(0.212), ...

Rule2: If Vble_2 is Medium THEN Vble_Output is Low (W=0.982)
Correct Instances  => (3.00300) (0.25025)  => 7 (1.0), 14(0.352), 17(0.829), ...
Incorrect Instances  => (0.00000) (0.00000)  => There are no items
Positive Instances => (0.00000) (0.00000)  => There are no items
Negative Instances => (0.00000) (0.00000)  => There are no items

Rule3: IF Vble_3 is LABEL_1 THEN Vble_4 is LABEL_0
Rule Support  => (737.32151) (0.67830) => 1(0.561), 2(0.407), 3(0.872), ...
Antecedent Support  => (747.00990) (0.68722) => 1(0.561), 2(0.407), 3(0.872), ...
Consequent Support  => (998.73089) (0.91879) => 1(0.561), 2(0.407), 3(0.872), ...

...

Rule11: UNCOVERED INSTANCES
80(C1), 90(C1), 93(C2), ... / 1, 2, 3, ... / 4, 5, 6, ...
```

Figure 3.16: Structure of *.fs* files handled by Fingrams Generator software.

- The second line changes depending on the selected type of rules. It includes the classes (with the number of instances per class in brackets) for classification problems; the output linguistic labels (with number of instances corresponding to each label in brackets) in case of regression problems[7]; and the possible output variables along with the attached linguistic terms for fuzzy association rules.

- The next four lines show information about the origin of the fuzzy system and the thresholds used. The first shows the software used to create the .fs file (GUAJE, FURIA and KEEL are shown as example). The next three show merely informative values of three thresholds. Blank threshold represents the minimum level an instance should fire a rule to be taken into account. Goodness thresholds (low and high) correspond to the minimum and maximum levels of rule firing that are considered when computing the *goodness measure* (as presented in Eq. 3.12) in classification and regression problems. These two final thresholds are not taken into account in case of fuzzy association rules.

- The next two lines yield the total number of rules (plus one in case of having uncovered instances) and the total number of data instances.

- Then, information related to each rule is detailed. For illustrative purposes, we have included three examples of rules (each one corresponding to one of the three types of rules that are considered) and a final line showing how to specify uncovered instances. Thus, Rule1, Rule2, and Rule3 exemplify rules of classification, regression and association respectively. In all cases the first line gives the rule identifier along with its linguistic description and their weight, if specified.

  - In case of classification rules, each rule is accompanied by four lines. The first two lines give the enumeration of correct and incorrect covered instances, i.e. data instances covered concordant or not with rule output class. The next line lists the positive instances, i.e. correct instances that fires the rule above the high goodness threshold. The last line enumerates the negative instances, i.e. incorrect instance plus the correct instances that are fired below the low goodness threshold. In all the four cases, the values in parenthesis at the the beginning of lines are the accumulated and average firing degrees of the enumerated instances, and the value after each instance identifier gives the level up to which it fires the corresponding rule.

  - For regression problems, each rule comes with almost the same information. Formally, the four lines are the same. The only difference turns out in the interpretation of the meaning of correct/incorrect and positive/negative instances. The interested reader can get additional information in [65].

[7]The software permits the use of multi-input-single-output (MISO) FRBSs for regression problems.

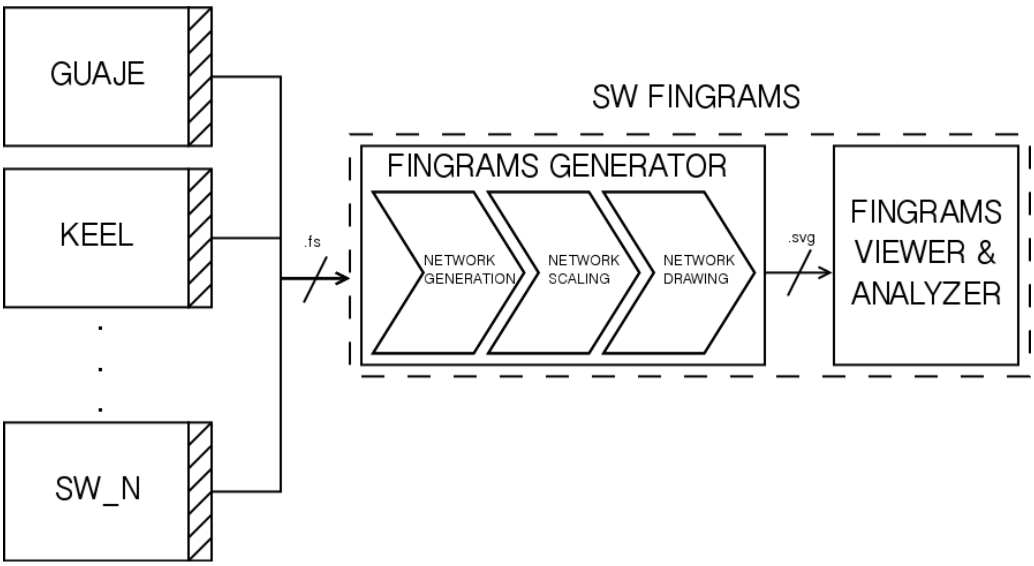


Figure 3.17: Software scheme of Fingrams Generator software.

– In case of fuzzy association rules, each rule description is accompanied by three lines. They are related to the support of the entire rule, of its antecedent and consequent parts alone. For each of them, we enumerate the list of supported instances along with the total accumulated support (first parenthesis) and the average support (second).

– Last part of Figure 3.16 shows how to specify uncovered instances (as presented in Eq. 3.4(b)) in this configuration file. First line shows the text *UNCOVERED INSTANCES*. Second line includes the identifier of instances not covered by the set of rules. Only in case of classification rules this identifier is accompanied by the output class of the instance in brackets.

We developed the software using Java. We chose this programming language because it is characterized by high penetration in academic and professional tools, modular design and maintenance, and interoperability through operating systems. The creation, visualization and analysis of Fingrams are done in four steps as sketched in Figure 3.17. The first three are as presented in Section 3 and the forth is in charge of showing the generated Fingrams to the user.

Some implementation details are given below:

1. **Network generation**: We have developed ad-hoc software for this task. It is in charge of creating the co-firing matrix that characterizes all the relations among rules.

2. **Network scaling**: We have included a Java implementation of Fast-Pathfinder [71], a variant of Pathfinder that reduces the complexity of the original Pathfinder algorithm.

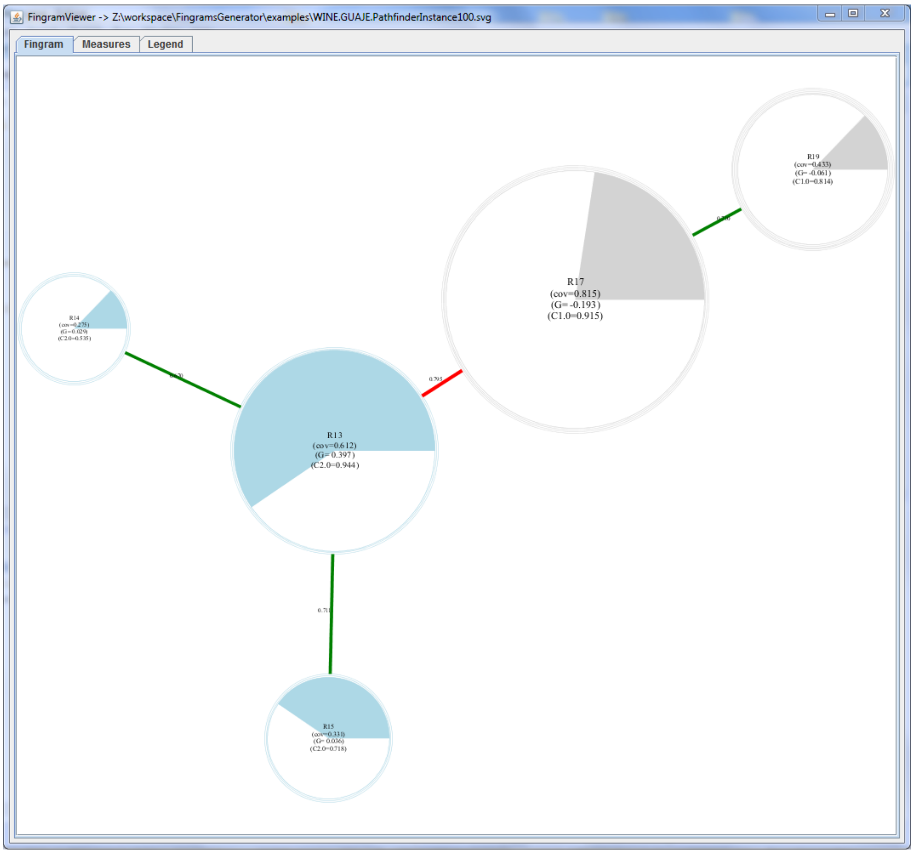


Figure 3.18: Screenshot of resultant window of Fingrams Generator software.

3. **Network drawing**: We generate the layout of the final Fingram using the open source visualization software Graphviz[8]. By default neato option is chosen to generate a layout based on the Kamada-Kawai [51] spring embedded algorithm. We can store the resultant image in different picture formats, but *.svg* is chosen by default.

4. **Network visualization**: We developed a SVG viewer (using Batik library for Java[9]) aimed at making user-friendly the visualization of the generated Fingrams. It includes three tabs: (1) *Fingram* tab; (2) *Measures* tab; and (3) *Legend* tab as presented in Figure 3.18.

   Thanks to the use of SVG format the user can interact with the graph in the first tab (Figure 3.18) through zooming, moving, and/or exploring in depth some zones of interest in the Fingram. In addition, when the user passes the mouse over a node or an edge, a text pops up with the additional information of the related element.

   The *Measures* tab gives several rule rankings based on some of the most popular measures in the context of social network analysis, such as *Page Rank* or *Centrality*.

   Finally, the *Legend* tab present a specific legend that helps in the comprehension of the Fingram.

[8]Graphviz is freely available at: *http://www.graphviz.org*.
[9]Batik is freely available at: *http://xmlgraphics.apache.org/batik/*.

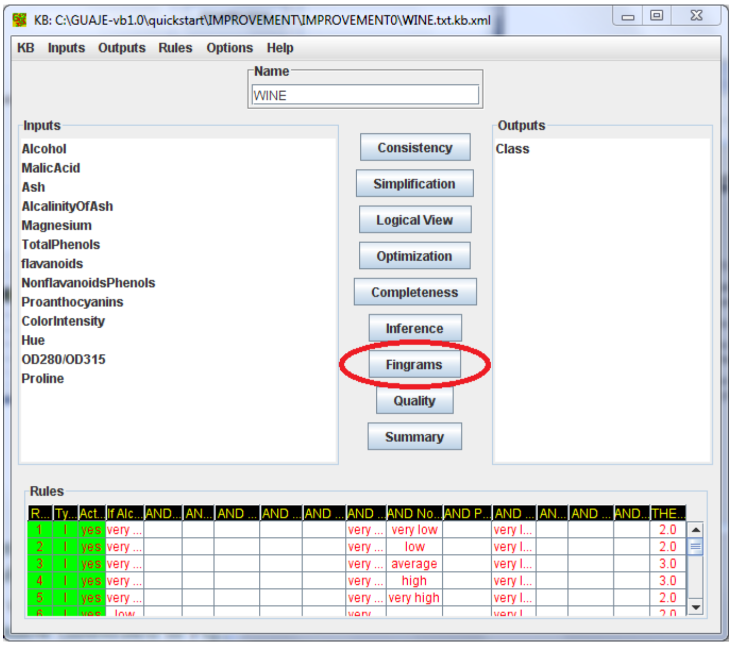
Fingrams Generator allows different parameters to construct Fingrams as presented below. A deeper explanation of those parameters is available at:
https://sourceforge.net/projects/fingrams/.

```
Usage: <main class> [options]
  Options:
    -d, --drawing
       Drawing algorithm
       Default: neato
    -e, --example
       Example to construct the instance-based Fingram for
       Default: -1
    -g, --guaje
       Execeute the code in GUAJE mode
       Default: -1
    -h, --help
       Shows the help of the Fingrams generator
       Default: false
  * -i, --input
       Input file
    -m, --metric
       Metric used to construct the social network
       Default: 0
    -n, --noWindow
       Do not show the Fingrams in a window
       Default: false
    -o, --output
       Format of the output file
       Default: SVG
    -t, --threshold
       Threshold to scale the network previous to apply Pathfinder
       Default: 0.0
    -v, --verbose
       Show the expended times per phase
       Default: false
    -q
       Parameter 'q' for pathfinder algorithm (N-value), default is
       1 (N-1) with N the number of nodes
       Default: 1
```

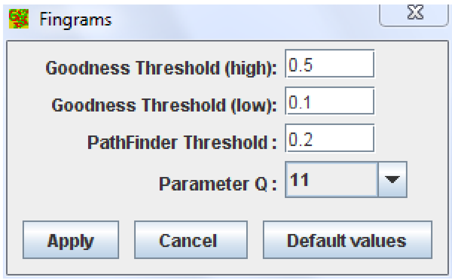
3.3.2 GUAJE

A software package for Fingrams generation and analysis is already implemented as part of the GUAJE tool[10]. GUAJE stands for Generating Understandable and Accurate fuzzy models in a Java Environment. It consists on a computational environment aimed at yielding a good interpretability-accuracy trade-off thanks to combining expert and induced knowledge in a common framework. GUAJE takes profit from the main advantages of several preexisting open source tools, as the Fingrams Generator software presented in the previous section.

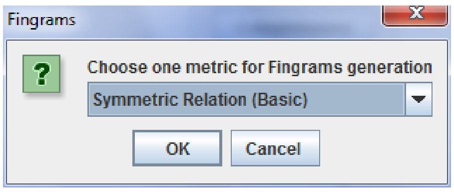
GUAJE has been enhanced with a software module for Fingram generation and analysis. It permits the creation of classification and regression Fingrams.



(a) Expert window.



(b) Selection of parameters.



(c) Selection of metric.

Figure 3.19: Generation of Fingrams in the software tool GUAJE.

Once a FRBS is generated, we can use Fingrams with the aim of understand the FRBS behavior at inference level. Several windows are presented, following a sequential procedure to construct Fingrams:

1. First of all, we have to launch the Fingrams module by clicking the corresponding button in the GUAJE expert window (highlighted in Figure 3.19(a)).

2. Then, we must set some parameters (see picture in Figure 3.19(b)):

   - **Goodness Threshold**. Upper and lower thresholds for estimating the goodness of coverage regarding each single rule. The goodness measure informs about how well each rule classifies the problem instances that it covers. A rule

[10] GUAJE is freely available as open source software at *http://sourceforge.net/projects/guajefuzzy/*.

covers one problem instance when the rule firing degree for that instance is greater than a predefined threshold.

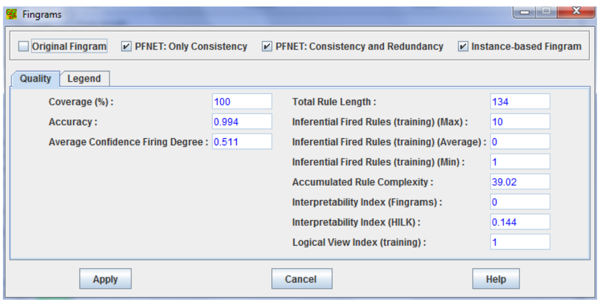
- **Pathfinder Threshold**. This parameter is used for pruning the initial graph (removing those edges with weights smaller or equal than the threshold), before running Pathfinder.
- **Q**. This is the specific parameter of Pathfinder which limits the number of edges in the paths respecting the triangle inequality. GUAJE suggests assigning to Q the maximum number of rules that can be simultaneously fired, which is estimated in an inferential way regarding the available dataset. In consequence, the network scaling will take shorter time. Anyway, by default $Q = N - 1$, with the aim of assuring that all paths are properly analyzed.

3. A popup panel messages allow us to select the metric used to construct Fingrams (Figure 3.19(c)). Three metrics (*Symmetric Relation (Basic)* corresponds to m_{ij}^0, *Symmetric Relation (Advanced)* to m_{ij}^1 and *Asymmetric Relation* to m_{ij}^2) are allowed in GUAJE.

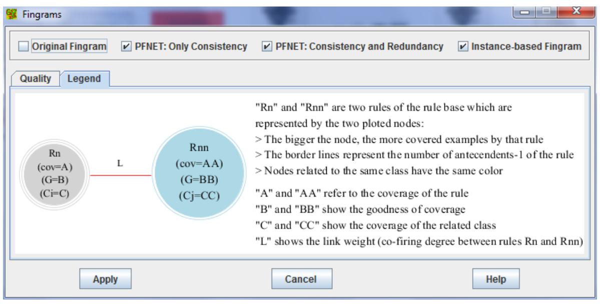
4. Afterwards, we have to choose a layout algorithm among those provided by Graphviz (*neato* corresponds to Kamada-Kawai [51], *fdp* to Fruchterman-Reingold [37], *circo* to a circular layout, and so on).

5. After that, we can also select an instance to create an instance-based Fingram from it.

6. Then, the GUAJE window of Figure 3.20(a) appears. The body of the window is structured in the form of a tabbed panel. The *Quality* tab gives an overview of the quality of the designed FRBS. It provides a list of quality indices[11] regarding both accuracy (on the left) and interpretability (on the right). Moreover, the user can interpret Fingrams according to the information presented in the *Legend* tab (Figure 3.20(b)).



(a) Quality view.



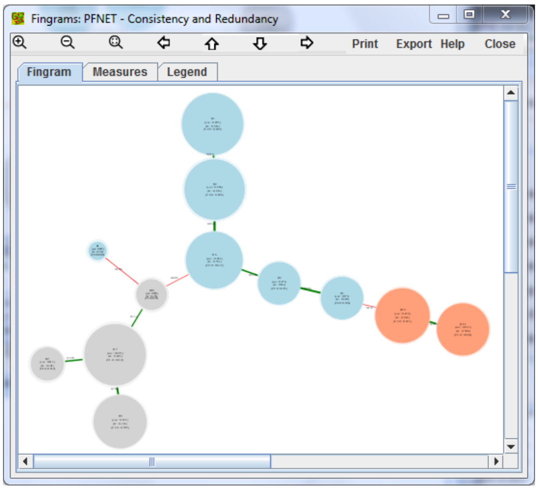
(b) Legend view.

Figure 3.20: GUAJE Fingrams window.

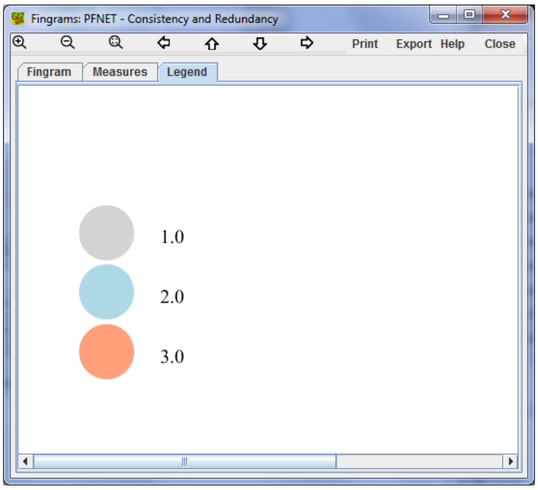
[11]Such indices are thoroughly explained in the GUAJE user manual.

7. Once selected the Fingrams to display, the GUAJE window for Fingram analysis turns out (Figure 3.21). This window is dual to the presented in section 3.3.1.

   Moreover, in this implementation the user can disable rules by clicking on its corresponding node, i.e., a rule is temporally deactivated in the rule base, and the Fingram is generated again without taking care of that rule. In consequence, fuzzy systems design in GUAJE becomes an interactive process which is effectively guided by decisions drawn from the expert analysis of Fingrams.



(a) *Fingram* tab.



(b) *Legend* tab.

(c) *Measures* tab.

Figure 3.21: Fingram analysis window in the software tool GUAJE.

Finally, it is worthy to remark that GUAJE comes with several intuitive and interactive tutorials. One of them shows the benefits and potentials of Fingrams for aiding the design of FRBSs [13]. It details, step by step, first how to build an interpretable fuzzy rule-based classifier and then how to simplify and optimize it, looking for the best balance between accuracy and interpretability supported by Fingrams.

The interested reader can find an illustrative case study in [67] where he/she can better understand the GUAJE module for creation of Fingrams.

3.3.3 KEEL

We have designed and developed a module that permits the creation of FAR-Fingrams (as introduced in Section 3.1.4.3) in the suite KEEL.

KEEL[12] is a software tool to assess computational intelligence algorithms for Data Mining problems including regression, classification, clustering, pattern mining and so on [5, 4].

Fingrams module takes as input the fuzzy association rules generated by one of the algorithms into KEEL and constructs FAR-Fingrams in vectorial SVG format. It can be used over fuzzy association rules created by the algorithms Alcalaetal-A, FuzzyApriori-A, GeneticFuzzyAprioriDC-A, and GeneticFuzzyApriori-A available in KEEL.

Figure 3.22 illustrates the use of this module. The blue nodes in background create fuzzy association rules, while the brown ones are in charge of constructing FAR-Fingrams. The dialog presented in the foreground of the figure shows the possibilities Fingram module provides. Fingrams module requires Graphviz 2.30 (or later) libraries installed in the computer. Thus, a pop-up message warns about that when first use of the module.

We will overview here the possibilities the KEEL Fingram module provides and how the different parameters should be selected (see Figure 3.22).

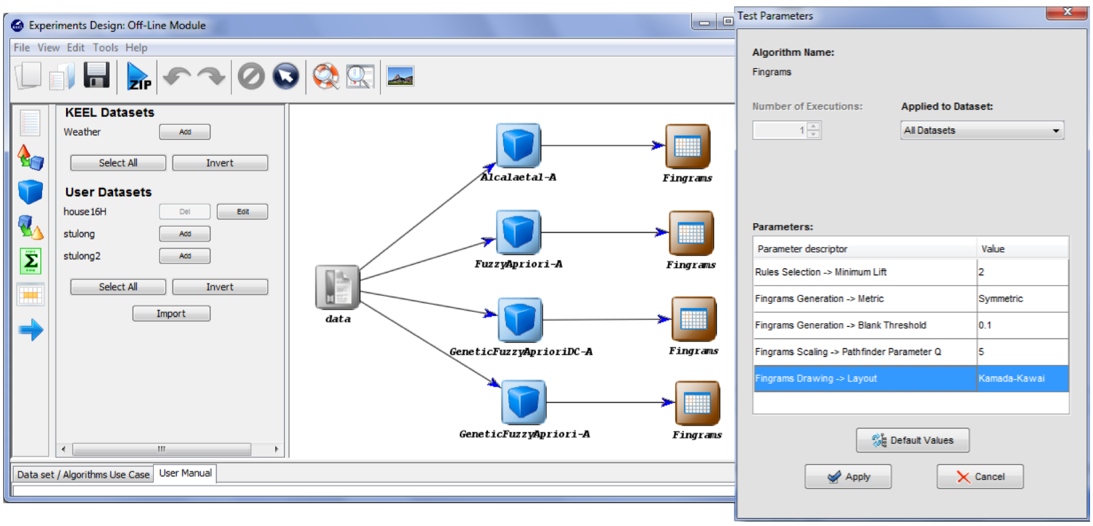


Figure 3.22: Experiments window of suite KEEL.

1. **Rule selection**: This option allows selecting those rules with a value for the Lift measure higher than a threshold. This reduces the number of rules to consider and allows the user to focus his/her attention in those more relevant.

2. **Fingram generation**: Two parameters can be selected in the dialog for this step. Parameter "Fingrams Generation → Metric" permits constructing FAR-Fingrams through m^0_{ij} and m^2_{ij} co-firing metrics (as presented in Section 3.1.1. And "Fingrams

[12]KEEL is freely available as open source software at: *http://www.keel.es/*.

Generation → Blank threshold" lets the user to discard instances that fire rules below a threshold.

3. **Fingram scaling**: This option allows us to create the complete and scaled FAR-Fingrams, allowing the user to study both in detail. Pathfinder algorithm requires a Q parameter that constrains the number of indirect proximities examined when generating the network. It must be an integer value between 1 and $N-1$, where N is the number of nodes to take into account. The configuration window of the module allows changing that value.

4. **Fingram drawing**: Fingrams can be displayed using Kamada-Kawai [51] and Fruchterman-Reingold [37], two of the most representative and used methods of force-directed algorithms. As result, we obtain SVG images enriched with additional information. The use of this vectorial format permits a comfortable analysis, zooming and moving around the interesting zones.

A deeper explanation of this software module and cases of use can be found in [63, 64].

3.3.4 KNIME

We designed and developed a set of software nodes that permit the creation and visualization of Fingrams in Konstanz Information Miner (KNIME) [16].

KNIME[13] is a modular, open platform for data integration, processing, analysis, and exploration. KNIME is increasingly used in industry and in academia in various areas of data mining and machine learning. Due to the modular nature of KNIME, it is straightforward to add other data types such as sequences, molecules, documents, or images.

Fingram nodes for KNIME take advantage of the KNIME plug-in for social networks as well as the KNIME fuzzy types and nodes. The modular environment of KNIME allows us to reuse the social networks plug-in nodes to filter, visualize and analyze Fingrams. Moreover, we developed a Pathfinder node to scale not only Fingrams but any social network in KNIME.

- **Fingram Generator:** This is in charge of creating Fingrams from a given fuzzy rule-based classifier, a set of data instances, and an empty social network. The output is a social network reflecting rule interaction at inference level.

- **Fingram Generator From File:** This node creates a Fingram from a .fs configuration file (as presented in Figure 3.16) that provides information related to fuzzy rules along with the lists of data instances firing them. As result, this node admits fuzzy association rules, fuzzy rule-based classifiers and regressors developed outside KNIME. It is important to remark that it requires an empty network as input, like

[13]KNIME is available as open source software at *http://www.knime.org*

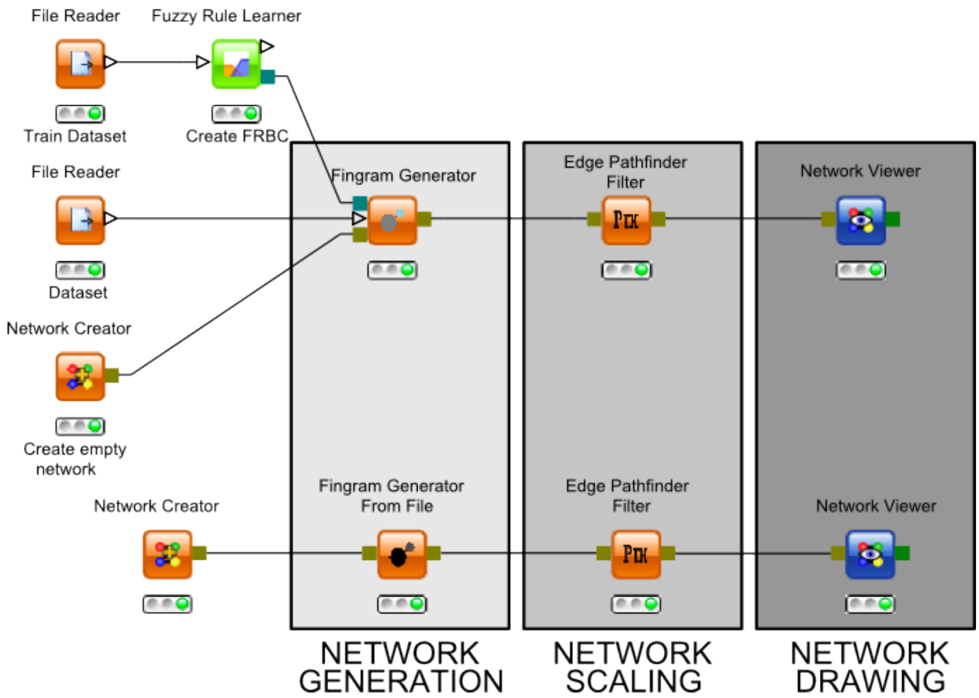


Figure 3.23: Fingrams workflow in KNIME software platform.

all social network nodes in KNIME. The output is a social network like the one given by the previous "Fingram Generator" node.

- **Edge Pathfinder Filter:** It permits the scaling of social networks using the Pathfinder Algorithm [75]. An edge is filtered if and only if it exists a path of length Q or smaller with higher associated value (minimum of the values along the path) than the direct edge. The parameter Q can be set in the configuration dialog and takes values between $[1, N-1]$ (with N the number of vertices).

Figure 3.23 illustrates the common use of the nodes introduced above. "Fingram Generator" and "Fingram Generator From File" construct Fingrams that are then scaled using "Edge Pathfinder Filter", and finally presented to the user with the KNIME "Network Viewer" node.

Figure 3.24 shows a Fingram constructed in KNIME. Nodes only include rule identifiers in this implementation More information of the elements of the Fingram is given in the right side table, such as the number of antecedents of the rules, weight of the edges or the text of the rule.

The implementation of these modules, along with exploration of other metrics and layouts, were done from May to July of 2013, during a stay at Department of Computer and Information Science, University of Konstanz, Germany. From this collaboration, we have published two congress papers [66, 69].

During this stay, we developed a new Fingram layout based on a multidimensional scaling approach. We take advantage of a mapping from high dimensional feature spaces onto

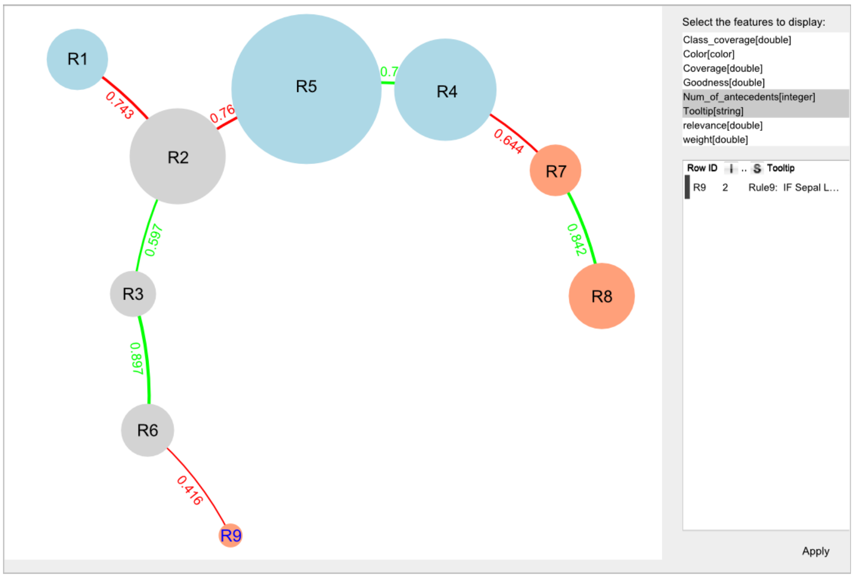
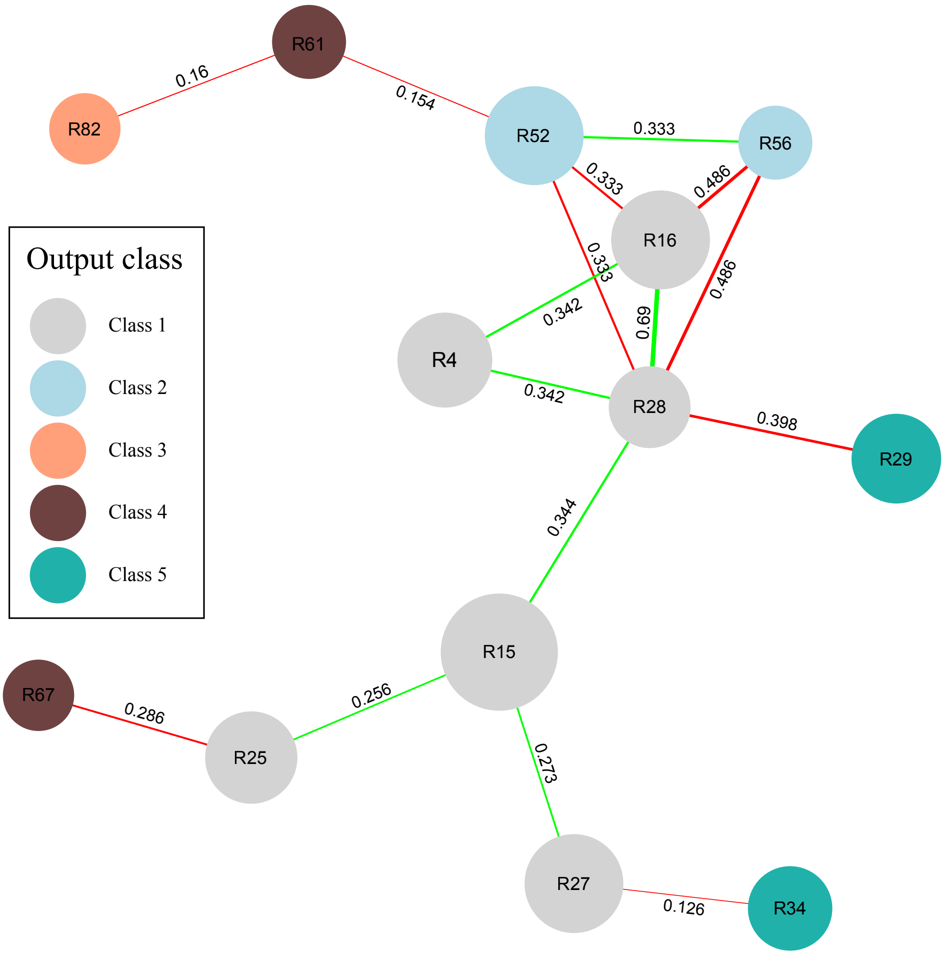


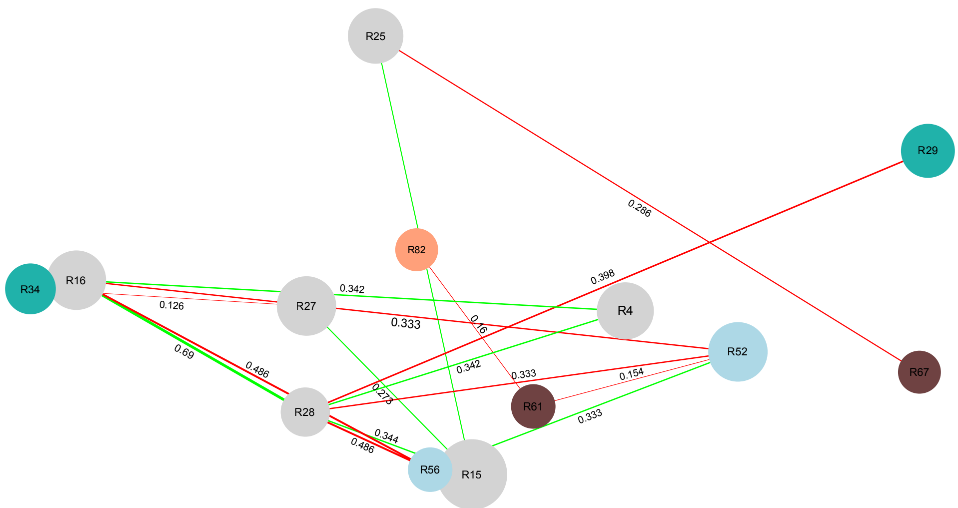
Figure 3.24: Illustrative example (IRIS) using KNIME software platform.

two-dimensional spaces which maintains the pairwise topological distances between fuzzy classification rules proposed by Gabriel et al. [38]. That method uses multi-dimensional scaling to place the rule centers and subsequently extends the rules regions to depict their overlap. This results not only in a visualization of the distribution of rules (coherent with the underlying data distribution) but also enables the relationship to their immediate neighbors to be judged. Notice that such relationship does not pay attention to rule co-firing. Therefore, we construct Fingrams as usual and the nodes are placed according to the multi-dimensional scaling.

As example, we present Figure 3.25 where both layout algorithms are presented. We can conclude that they provide complementary information. Kamada-Kawai (Figure 3.25(a)) offers a symbolic representation that allows to easily visualize all the elements of the graph. Complementary, the multi-dimensional scaling layout (Figure 3.25(b)) yields a topological representation, which helps in the comprehension of the system better showing the proximity of rules, but neglects aesthetical issues, difficulting the visualization of all the elements of the Fingram.



(a) Kamada-Kawai Layout.



(b) Multi-dimensional scaling layout.

Figure 3.25: Comparison of layouts in KNIME.

Chapter 4

Conclusions and future work

This Ph.D dissertation has introduced Fingrams as a new powerful methodology for exploratory analysis of fuzzy systems. An overview of the possibilities that Fingrams offer, for both design and analysis of fuzzy systems, has been illustrated through some cases of use. Moreover, we have provided free software implementations that permits the construction and analysis of Fingrams.

Fingrams show the inference mechanism of FRBSs from a global view point, i.e. observing how all the rules covered the complete given dataset, and from a local view point, i.e. illustrating a partial view of the system when focusing on those rules that participate in the inference process regarding a single instance, by the so-called *instance-based Fingrams*. In consequence, we can analyze the system in detail, and even improve it with expert knowledge, carefully checking rule by rule and instance by instance.

Additionally, Fingrams allow us to detect and analyze uncovered instances, a key behavior in fuzzy modeling because those instances directly penalize precision.

Fingrams are likely to be applied to several applications to design or improve fuzzy systems. The human centric simplification of an FRBS by means of the elimination or modification of rules could be done after analyzing the resulting graphs manually or assisted by well-known social network analysis techniques (such as community mining) and quality indexes (such as centrality, page rank and so on). For example, the detection of rules that do not cover any example is very easy using Fingrams; rules that have a low overlapping with others can be detected to proceed as desired; and so forth.

Three metrics have been proposed to construct Fingrams along the Ph.D. dissertation. The simplest symmetric metric (m^0_{ij}), inspired by the co-citation metric usually used in scientometrics, relates two rules (R_i and R_j) according to the number of instances covered in common by them [65]. A more advanced symmetric metric (m^1_{ij}) includes the firing degree up to which the data instances activate the rules as well as the rule weights [68], allowing to understand more subtly the behavior of fuzzy inference mechanisms. Finally, an asymmetric co-firing metric (m^2_{ij}) characterizes generalization/specialization relations between pairs of rules [64].

Fingrams can already deal with fuzzy association rules [63], fuzzy rule-based classifiers and regressors [65]. The different adaptations show relevant information according to their characteristics.

We validated our proposal building Fingrams over three cases of use. FURIA algorithm was used over a real dataset to show the possibilities of classification Fingrams. We selected an electrical network distribution problem to present the potentials of regression Fingrams. The utility of fuzzy association rules Fingrams was illustrated in a real-world problem dealing with qualitative assessment of industrial objects automatically designed through cognitive engineering, in the context of Quale® research line[1].

We have implemented Fingrams in different software tools as well as in a specific command-line software. Fingrams Generator [62] permits the creation of Fingrams no matter how the depicted FRBS was generated. The fuzzy modeling toolbox GUAJE [67], and the software suites for data mining KEEL [64] and KNIME [69] already allow the creation and analysis of Fingrams.

The future of this methodology is very promising. Notice that, there are still many research tasks to carry out.

The concept of Fingram can be extended to relate not only fuzzy rules, but also attributes/fuzzy terms appearing in the fuzzy rules. So for, new metrics will be proposed in the near future to produce complementary information about the system to the designer.

In addition, the metrics presented in this thesis are biased by the available training data. The availability, representability and quality of this data directly determine the validity of the obtained Fingrams. Other measures of overlapping between rules may be proposed for avoiding these limitations.

On the other hand, we will propose adaptations to other type of FRBSs, such as Takagi-Sugeno FRBSs. Even more, the methodology could be generalized to non-fuzzy rule based systems.

Moreover, we will propose a generalized local view of the system guided by a set of instances. This will extend instance-based Fingrams, allowing a detail study of a set of instances of special interest. For example, key instances that produce failure in known situations may be selected for observing their inference in detail.

Finally, the last but not the least important future challenge comes with the automatic creation of linguistic descriptions from Fingrams that explain the system behavior. These descriptions can textually summarize the behavior of a FRBS, highlighting its most relevant elements, such as, the number of uncovered instances, the most important rules, or rules that behave incorrectly.

[1] Quale® is a research line of the European Centre for Soft Computing that supports the human-centered design of customized products/services, considering the analysis and quantification of qualitative assessments.

Conclusiones y trabajo futuro

Esta tesis doctoral ha presentado una nueva metodología para el análisis de sistemas fuzzy mediante los denominados Fuzzy Inference-grams o *Fingrams*. Se han ilustrado las posibilidades que ofrece esta metodología tanto para diseño como para análisis de sistemas fuzzy, con algunos casos de uso. Durante la realización de la tesis se han implementado herramientas software que permiten la construcción y análisis de Fingrams, facilitando y extendiendo su uso.

Los Fingrams muestran el mecanismo de inferencia de sistemas de reglas fuzzy tanto desde un punto de vista global, esto es, observando como todas las reglas cubren el conjunto de datos completo, como desde un punto de vista local, es decir, como una instancia es cubierta por el conjunto de reglas, por los Fingrams basados en instancias. Por tanto, podemos analizar un sistema en detalle, permitiendo su mejora con conocimiento experto, estudiando regla a regla e instancia a instancia como éste se comporta.

Adicionalmente, la metodología facilita la detección y análisis de instancias no cubiertas, comportamiento muy importante en este tipo de sistemas ya que penaliza la precisión del mismo.

Los Fingrams pueden ser utilizados en múltiples aplicaciones para diseñar y/o mejorar sistemas fuzzy. Tras analizar un sistema fuzzy utilizando la representación gráfica propuesta, podemos simplificar el sistema de forma guiada. Por ejemplo, la detección de reglas que no cubren ninguna instancia es realizada de forma intuitiva con Fingrams; reglas que cubren pocas instancias en común con otras se identifican de forma sencilla pudiendo proceder como se desee; etcétera.

Hemos propuesto tres métricas para la construcción de Fingrams. La más simple (m_{ij}^0), inspirada por la métrica de co-citación habitualmente usada en cienciometría, relaciona dos reglas de acuerdo al número de instancias que cubren en común [65]. Una segunda métrica más avanzada (m_{ij}^1) incluye el nivel de disparo de las reglas para las instancias dadas así como el peso que pudieran tener las reglas [68]. Esto permite comprender más en detalle el comportamiento del mecanismo de inferencia. Finalmente, una métrica de co-disparo asimétrica (m_{ij}^2) caracteriza las relaciones de generalización/espeficidad entre pares de reglas [64].

La metodología puede ser utilizada en reglas de asociación fuzzy [63], clasificadores y regresores basados en reglas fuzzy [65]. Las distintas adaptaciones muestran información relevante en cada uno de los casos.

Hemos validado nuestra propuesta utilizando Fingrams en tres casos de uso. Mostramos las posibilidades de los Fingrams de clasificación para un ejemplo construido con el algoritmo de modelado preciso FURIA. Para el caso de Fingrams de regresión hemos seleccionado un problema de distribución de la red eléctrica. Por último, la utilidad de Fingrams en reglas de asociación ha sido ilustrado por un problema real que estima las valoraciones cualitativas de distintas muestras de diseño industrial.

Se han implementado módulos software para utilizar Fingrams dentro de distintas paquetes software, así como una herramienta de línea de comandos específica. Fingrams Generator [62] permite la creación de Fingrams independientemente de la herramienta de diseño utilizada para generar el sistema de reglas fuzzy. La herramienta de diseño GUAJE [67], y las suites software de minería de datos KEEL [64] y KNIME [69] cuentan con módulos que permiten la creación y análisis de Fingrams.

El futuro de esta metodología es muy prometedor, con mucho recorrido de investigación y aplicación de la misma.

El concepto de Fingram puede ser extendido no sólo relacionando reglas, sino también atributos o términos fuzzy presentes en el sistema fuzzy. Por tanto, se propondrán nuevas métricas que proporcionen información complementaria al diseñador sobre el sistema.

Las métricas propuestas a lo largo de la tesis están condicionadas por los datos disponibles. Su disponibilidad, representatividad y calidad directamente condicionan la validez de los Fingrams obtenidos. La propuesta de otras medidas de solapamiento entre reglas podrían proponerse para evitar estas limitaciones.

Por otro lado, se plantea la adaptación de Fingrams para representar otros tipos de reglas fuzzy, tales como sistemas fuzzy tipo Takagi-Sugeno. Aún más, la metodología puede ser generalizada a sistemas de reglas no fuzzy.

Se planea extender la vista local para que permita visualizar conjuntos de reglas que son activadas por un conjunto de instancias, en vez de instancias individuales. De esta forma se permite estudiar conjuntos de instancias de especial interés. Por ejemplo, instancias que produzcan fallo en circunstancias conocidas pueden ser seleccionadas para estudiar la inferencia del sistema en detalle.

Finalmente, un desafío al que enfrentarse será la creación automática de descripciones lingüísticas de Fingrams que expliquen el comportamiento del sistema. Estas descripciones podrán resumir textualmente el comportamiento de un sistema de reglas fuzzy, destacando aquellos elementos más importantes, tales como, el número de instancias no cubiertas, las reglas más importantes o las reglas que funcionan de forma incorrecta.

Appendix A

Publications

A.1 Compilation of publications

This section contains the three journal publications produced along the thesis. Each section contains the citation information of the corresponding publication, a briefly description of its content, the number of citations of each publication[1] and its final version. Additionally, and after each publication, we include the last published *Journal Citation Reports*, year 2013, of the journals where the previous publications appeared.

The *impact factor* of an academic journal is a measure reflecting the average number of citations to articles published in that journal. In any given year, the impact factor of a journal is the average number of citations received per paper published in that journal during the two preceding years.

This is frequently used as a proxy for the relative importance of a journal within its field. Journals with higher impact factor tend to be more important than those with lower. Impact factors are calculated yearly, since 1975, for those journals indexed in the *Journal Citation Reports*.

[1] We include information of Google Scholar and Web of Science retrieved on August the 13th 2015. We indicate with a − when we do not obtain any information from the mentioned web services.

A.1.1 First publication: Methodology

[65] Pancho, D. P., Alonso, J. M., Cordón, O., Quirin, A., and Magdalena, L. FINGRAMS: visual representations of fuzzy rule-based inference for expert analysis of comprehensibility. *IEEE Transactions on Fuzzy Systems 21*, 6 (2013), 1133–1149

This publication sets the methodology of Fingrams and presents the procedure to create and analyze them. A brief overview of the possibilities that Fingrams offer, for both design and analysis of fuzzy systems, has been illustrated through classification and regression fuzzy systems. The interpretability index is introduced.

Citations Google Scholar:	13
Citations Web of Science:	7

FINGRAMS: Visual Representations of Fuzzy Rule-Based Inference for Expert Analysis of Comprehensibility

David P. Pancho, Jose M. Alonso, *Member, IEEE*, Oscar Cordón, *Senior Member, IEEE*, Arnaud Quirin, and Luis Magdalena, *Senior Member, IEEE*

***Abstract*—Since Zadeh's proposal and Mamdani's seminal ideas, interpretability is acknowledged as one of the most appreciated and valuable characteristics of fuzzy system identification methodologies. It represents the ability of fuzzy systems to formalize the behavior of a real system in a human understandable way, by means of a set of linguistic variables and rules with a high semantic expressivity close to natural language. Interpretability analysis involves two main points of view: readability of the knowledge base description (regarding complexity of fuzzy partitions and rules) and comprehensibility of the fuzzy system (regarding implicit and explicit semantics embedded in fuzzy partitions and rules, as well as the fuzzy reasoning method). Readability has been thoroughly treated by many authors who have proposed several criteria and metrics. Unfortunately, comprehensibility has usually been neglected because it involves some cognitive aspects related to human reasoning, which are very hard to formalize and to deal with. This paper proposes the creation of a new paradigm for fuzzy system comprehensibility analysis based on fuzzy systems' inference maps, so-called fuzzy inference-grams (fingrams), by analogy with scientograms used for visualizing the structure of science. Fingrams show graphically the interaction between rules at the inference level in terms of co-fired rules, i.e., rules fired at the same time by a given input. The analysis of fingrams offers many possibilities: measuring the comprehensibility of fuzzy systems, detecting redundancies and/or inconsistencies among fuzzy rules, identifying the most significant rules, etc. Some of these capabilities are explored in this study for the case of fuzzy models and classifiers.**

***Index Terms*—Comprehensibility analysis, expert analysis, fuzzy modeling, information visualization, interpretability–accuracy tradeoff, social network analysis (SNA).**

Manuscript received June 5, 2012; revised October 22, 2012; accepted December 27, 2012. Date of publication February 5, 2013; date of current version November 25, 2013. This work was supported in part by the Spanish Government under program JCI-2011-09839 ("Juan de la Cierva"), the Spanish Ministry of Economy and Competitiveness under Projects TIN2011-29824-C02-01 and TIN2011-29824-C02-02, the Spanish Ministry of Education and Science under Project TIN2009-07727, the European Regional Development Fund, and the European Centre for Soft Computing, Mieres, Spain.

D. P. Pancho, J. M. Alonso, and L. Magdalena are with the European Centre for Soft Computing, Mieres 33600, Spain (e-mail: david.perez@softcomputing.es; jose.alonso@softcomputing.es; luis.magdalena@softcomputing.es).

O. Cordón is with the Department of Computer Science and Artificial Intelligence and the Research Center for Information and Communications Technologies, University of Granada, Granada 18071, Spain (e-mail: ocordon@decsai.ugr.es).

A. Quirin is with the Galician Research and Development Center in Advanced Telecommunications, University of Vigo, Vigo 36310, Spain (e-mail: aquirin@gradiant.org).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TFUZZ.2013.2245130

I. Introduction

INTERPRETABILITY of a fuzzy system involves the skill or talent of the specific end user, i.e., the person who interprets its linguistic description with the aim of inferring (conceiving) the significance of the system behavior. In consequence, characterizing and assessing interpretability is a very subjective task, which strongly depends on the background (experience, preferences, knowledge, etc.) of the person who makes the evaluation [1].

Interpretability is a distinguishing capability of fuzzy systems that is really appreciated in most applications. Even more, it becomes an essential requirement for those applications that involve extensive interaction with human beings. Thus, we will focus on the so-called *humanistic systems*, defined by Zadeh [2]–[4], as those systems whose behavior is strongly influenced by human judgment, perception, or emotions. For instance, decision support systems in medicine [5] must be easily understandable, for both physicians and patients, with the intention of being reliable, i.e., widely accepted, and successfully applicable.

Unfortunately, fuzzy systems are not interpretable *per se*; they have to be designed carefully to fulfill that characteristic. Of course, the use of linguistic variables [2]–[4] and rules [6], [7] favors interpretability due to their high semantic expressivity close to natural language. Nevertheless, there are many different issues that must be taken into account in order to design interpretable fuzzy systems. First, several interpretability constraints [8], [9] have to be imposed along the whole design process with the aim of producing fuzzy systems with the required interpretability level, i.e., systems capable of being understood, described, or accounted for by a human being. As a result of these constraints, interpretability is usually achieved at the cost of penalizing accuracy. For this reason, most fuzzy systems are built jeopardizing interpretability, only paying attention to accuracy. Even in those cases, authors usually claim that their fuzzy systems are much more interpretable than those systems based on black-box techniques, like neural networks, because they are based on fuzzy logic. Those claims are quite questionable and should be rejected because they are deceptive. Obtaining interpretable fuzzy systems is a matter of design, which must be carefully considered. Unless this is done neatly, produced fuzzy systems will be hardly interpretable, becoming black-boxes in that interpretability sense.

The assessment of interpretability has to face two main issues [1]: 1) readability (transparency) of the system description, which is related to the view of the model structure as a gray-box, and 2) comprehensibility of the system explanation, which is closer to cognitive aspects because it is always related to human beings. Of course, the analysis has to take into account all elements included in a fuzzy system, from the lowest (fuzzy partitions) to the highest (fuzzy rules) abstraction levels [10]. Namely, the analysis must range from the design of each individual linguistic term (and its related fuzzy set) to the analysis of the cooperation among several rules, what depends on the fuzzy inference mechanism.

Most previous works [11], [12] only analyze the readability of the designed fuzzy system. Moreover, the analysis of readability is usually reduced to a basic analysis of complexity, i.e., it consists of counting the number of elements included in the fuzzy knowledge base (number of rules, premises, linguistic terms, etc.). Other contributions also analyze structural properties of fuzzy partitions [8] such as distinguishability, coverage, and so on. Recently, a few authors have shown the importance of extending the analysis of readability to evaluate the implicit and explicit semantics embedded in a fuzzy knowledge base [13], [14]. Of course, keeping a small number of linguistic terms is appreciated due to the limits of human processing capabilities [15]. Nevertheless, not only the quantity but also the quality is very important. Thus, the selection of the right linguistic terms is essential to the yielding of interpretable systems. Notice that interpretable fuzzy partitions must represent prototypes that are meaningful for the interpreter.

Although there has been a huge effort for defining, characterizing, and assessing interpretability in the last decade, there is still a lot of work to be done. Namely, the comprehensibility analysis of the system explanation is almost negligible. Understanding the system behavior from its linguistic description becomes a very hard task that involves the inference level going beyond the simple assessment of the system structure readability.

This study presents a novel methodology, which was first sketched in [16], to analyze the fuzzy inference layer of a fuzzy rule-based system (FRBS) from the comprehensibility point of view. It is mainly based on the adaptation of recent analysis techniques from a completely different research field, that of Scientometrics [17]. We will consider the use and enrichment of existing techniques for visualizing scientific information based on social network analysis (SNA) [18], [19], called scientograms or visual science maps [20], to the visual analysis of the fuzzy systems' inference process. As a consequence, our new comprehensibility analysis tool will be called fuzzy inference-grams (*fingrams*) from now on.

FRBSs can be either designed from expert knowledge or automatically generated from experimental data with a specific learning technique. Anyway, the correspondence of generality and specificity in between the extracted knowledge and the available examples is not always straightforward. Moreover, this fact may become a handicap. Therefore, a visual representation of the FRBS inference process allows us to find out how rules cover examples and how rules are related among them, because they interact to produce the overall behavior of the system.

A first software package for generation and analysis of fingrams has been implemented. It is freely downloadable as open source software as part of the GUAJE tool.[1] All application examples presented in this paper are conducted using this software. Moreover, it includes an interactive guide tutorial that allows the user to become familiar with the tool. As a result, the interested reader can use GUAJE not only to reproduce the illustrative examples presented in this paper, but also to generate and analyze her/his own fingrams.

The rest of this paper is organized as follows. Section II presents some preliminaries including basic aspects related to interpretability assessment, a brief overview on existent methodologies for visual representation and analysis of fuzzy systems, and a short introduction to the most widely known techniques for SNA, extending the design and analysis of visual science maps. Section III introduces the fingram generation process, while Section IV presents the possibilities fingram analysis offers. Section V shows some illustrative application examples. Finally, some conclusions and future works are presented in Section VI.

II. Preliminaries

A. Assessing Interpretability of Fuzzy Rule-Based Systems

There are universal indices commonly accepted for accuracy assessment. For instance, the mean square error (MSE) and the number of misclassified patterns are widely used for regression and classification problems, respectively. However, this is not the case when dealing with interpretability evaluation, where the definition of such indices remains an open hot topic.

There are lots of interpretability indices that focus on specific characteristics of FRBSs. Nevertheless, finding a universal index for interpretability seems to be an impossible mission since the considered concept is strongly affected by subjectivity. In fact, there is a need to look for two kinds of complementary indices: objective and subjective ones. On the one hand, objective metrics are needed to make feasible fair comparisons among different fuzzy systems. On the other hand, subjective measures are demanded when looking for personalized fuzzy systems. Such systems require a flexible index to be easily adaptable to the context of each problem as well as to the end-user's preferences.

Interpretability indices can be grouped according to two different criteria [22]: the nature of the interpretability index (structure versus semantics), and the elements of the fuzzy knowledge base that it considers (fuzzy partitions versus rule base). The four derived groups are (Q1) structure at partition level, (Q2) structure at rule base level, (Q3) semantics at partition level, and (Q4) semantics at rule base level (see Fig. 1).

Most well-known existing interpretability indices correspond to groups Q1 and Q2; thus, they focus on readability (in terms of complexity at structural level) of fuzzy systems. In consequence, they are objective indices since they basically count the number of elements (features/variables, membership functions, rules, premises, etc.) existing in the FRBS.

[1] http://www.softcomputing.es/guaje [21]

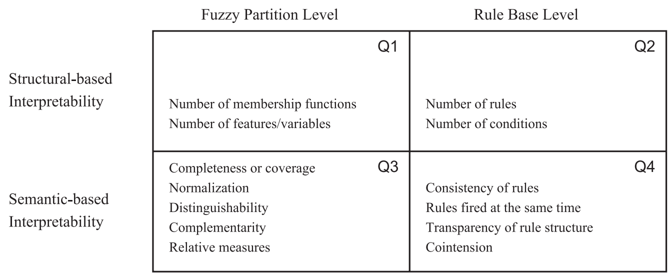


Fig. 1. Quadrant of interpretability indices [22].

Indices included in group Q3 usually measure the degree of fulfillment of semantic constraints that should be overimposed during the design process. In [8], de Oliveira proposed some semantic constraints (coverage, normalization, distinguishability, etc.) required to have interpretable fuzzy partitions from the semantical point of view. The use of strong fuzzy partitions (SFP) [23] satisfies all these semantic constraints. Nonetheless, notice that breaking the SFP property can yield more accurate systems. Therefore, there are proposals that ensure a good interpretability at this level, without considering SFP [13], [24], [25].

Finally, group Q4 is the one that contains the lowest number of works in the literature. These indices advocate extending the analysis of readability to evaluate the comprehensibility, i.e., the implicit and explicit semantics embedded in fuzzy systems [14]. There are also some papers dealing with the consistency of fuzzy rule bases and with the number of co-fired rules, i.e., rules simultaneously fired by a given input [26]–[28].

B. Visual Description and Analysis of Fuzzy Rule Bases

There are not many papers tackling visual analysis of the fuzzy system inference process. This is probably due to the well-known linguistic expressivity of fuzzy systems that gives prominence to linguistic representations. However, when dealing with complex real-world problems, even when the design is made carefully to maximize interpretability, the number of rules can become huge because of the curse of the dimensionality characteristic of FRBSs. In those cases, looking for a plausible linguistic explanation of the inferred output, which is derived from the linguistic description of the fuzzy knowledge base, is not straightforward. When many rules are fired at the same time for a given input, explaining the inferred output as an aggregation of all the involved rules can be very complicated.

Some authors [29] have searched for understandable ways of interpreting the system output by describing the inferred output possibility distribution with a set of previously defined linguistic terms and some linguistic modifiers and connectives. As an alternative, other authors have made a bet for searching visual explanations of the system output [30]–[32]. In these papers, Ishibuchi *et al.* established a set of design constraints with the aim of producing groups of rules with only two antecedent conditions that can be represented in a 2-D space. These works focus on providing a visual representation that is able to explain the output of fuzzy rule-based classifiers to human users. Nevertheless, considering only two antecedents per rule is a strong limitation that may penalize the accuracy of the system, especially when dealing with complex and high-dimensional problems.

A complete analysis of visualization requirements for fuzzy systems is provided in [33]. That contribution gives an overview on existing methodologies to yield 2-D and 3-D graphical representations of fuzzy systems. It comprises visualization of fuzzy data, fuzzy partitions, and fuzzy rules. Different alternatives are available, depending on the requirements of the end user (fuzzy designer, domain expert, etc.). Moreover, requirements may change according to the visualization tasks to perform: interactive exploration; automatic computer-supported exploration; receiving feedback from users; and capturing users' profiles and adaptation.

The most relevant works on the design of visual representations for multidimensional fuzzy rules are those developed by Berthold *et al.* [34], [35]. They make a mapping from high-dimensional feature spaces onto 2-D spaces, which maintains the pairwise distances between rules. The established mapping also displays an approximation of each rule spread and overlapping. As a result, it is possible to visualize and explore multidimensional FRBSs in a 2-D graphical representation. The authors claim such a representation yields a user-friendly and interpretable exploratory analysis. However, the complexity of the analysis grows exponentially with the number of variables and rules to be displayed. In consequence, in complex and high-dimensional problems, the interpretation of the resulting graph is not straightforward.

Evsukoff *et al.* [36], [37] propose the use of an interpretation framework that helps understanding multidimensional fuzzy rules. They assign a symbol to each rule, which is represented by a Gaussian membership function. The model interpretation is based on analysis of rule weights and on a 2-D linear principal component analysis projection to visualize the model.

On a different basis, Casillas and Martínez-López [38] present the so-called "transition chromatic maps" for fuzzy rules generated from uncertain data. These maps are generated as result of a visual modeling process that represents the extracted knowledge in a more understandable way, thus helping in the postprocessing, interpretation stage of knowledge discovery in databases. They allow us to see the relations among variables by observing the chromatic evolution of the surfaces on the graph.

Table I summarizes the main characteristics of the most relevant visualization methods for multidimensional fuzzy rules previously introduced. All methods make a 2-D representation of fuzzy rules. Some of them represent data, and some others show the existing overlapping among rules at descriptive level,

TABLE I
CHARACTERISTICS OF VISUALIZATION METHODS FOR MULTIDIMENSIONAL FUZZY RULES

	[34]	[35]	[36], [37]	[38]
Represent data samples		✓	✓	✓
Represent overlapping among rules at descriptive level	✓	✓		✓
Represent rule interaction at inferential level				

but none of them represents rule interaction at inference level. This brief review shows that there is a lack of methods depicting the interaction among rules that would strongly help in the comprehension of the rule base behavior.

C. Social Network Analysis

A social network is a social structure made up of individuals called "nodes," which are connected or tied by "edges" (also called ties, links, or connections) corresponding to one or more specific types of interrelations, such as friendship, common interest, or knowledge. SNA [18], [19] views social relationships in terms of network theory regarding nodes and edges. Nodes are the individual actors within the networks and ties are the relationships among the actors. Research in a number of academic fields has shown that social networks operate on many levels, from families up to the level of nations. They play a critical role in determining how problems are solved, organizations are run, and individuals succeed in achieving their goals.

Given a network, the goal of the scaling algorithms is to take the proximity information and obtain structures revealing the underlying organization. The scaling algorithms use similarities, correlations, or distances to prune a graph based on proximity among pairs of nodes. The three predominant ways proposed in the literature to perform this task are analyzed in the following [39].

The first option introduces a link weight threshold, and it only considers the links having weights above this threshold [40]. This approach is straightforward and easy to implement. However, it does not take the intrinsic structure of the underlying network into account; therefore, the transformed network may not preserve the essence of the original one. Furthermore, the value of the threshold could be hard to adjust for the user.

The second option extracts a minimum spanning tree (MST) from a network of N vertices [41]. This approach guarantees that the number of links in the transformed network is always $N - 1$. However, that does not always reflect the subjacent relevant information.

The third option imposes constraints on paths and excludes links that do not satisfy the constraints. One of the most known methods, the Pathfinder algorithm [42], [43], is frequently used due to its mathematical properties related to the preservation of the triangular inequality. Those properties include the conservation of links, the capability of modeling symmetrical but also asymmetrical relationships, and the representation of the most *salient* relationships present in the data. The result of applying Pathfinder to a network is a pruned network called PFNET.

Once PFNETs or any other kind of pruned networks are generated, there are many different methods for their automatic visualization. Force-based or force-directed algorithms are the most widely used class of algorithms for drawing graphs in the area of information science [44], [45]. Their purpose is to locate the nodes of a graph in a 2-D or 3-D space so that all the edges are approximately of equal length and there are as few crossing edges as possible, trying to obtain the most aesthetically pleasing view. The best representations of this family of methods are Kamada–Kawai [46] and Fruchterman–Reingold [47].

Kamada–Kawai [46] is one of the most extended methods for visualizing PFNETs. Starting from a circular position of the nodes, it generates networks with aesthetic criteria such as the maximum use of the available space, the minimum number of crossed links, the forced separation of nodes, the generation of balanced maps, etc. It assigns coordinates to the nodes trying to adjust as much as possible the distances existing among them with respect to actual network distances.

In the Fruchterman–Reingold algorithm [47], the attraction or repulsion among nodes determines in which direction a node should move. Nodes move from an original layout step by step. The step width of node movements decreases at each iteration. Once nodes stop moving, the procedure ends.

The combination of SNA through the use of network scaling algorithms and visualization methods has proved its capability for getting high-quality schematic visualizations of the resulting networks in various fields: psychology (to represent the cognitive structure of a subject [42], [43]), software development (for debugging of multiagent systems [48]), scientometrics (for the analysis of large scientific domains [20], [49]), etc.

D. Scientogram Design and Analysis

The term *scientogram*, which is a particular case of social network, refers to visual science maps, i.e., visual representations of scientific domains. Vargas-Quesada *et al.* [20], [49], [50] proposed a methodology for creating scientograms with the aim of illustrating interactions among authors and papers through citations and co-citations. The basic idea arises from the notion of manuscript co-citation that represents the frequency with which two documents are simultaneously cited by others. It is possible to group them by author, journal, or thematic category, for instance. Of course, depending on the kind of grouping, the information that can be extracted from the generated maps is different.

The standardized co-citation measure was originally defined by Salton and Bergmark [51]

$$\mathrm{MCN}(ij) = \frac{Cc(ij)}{\sqrt{c(i) \cdot c(j)}} \tag{1}$$

where Cc means co-citation, c stands for citation, and i and j represent two different entities (authors, documents, journals, categories, institutions, countries, etc.).

As an illustrative example, Fig. 2 represents the scientogram of the world production in 2002. It consists of 16 thematic areas where the volume of the nodes is shown proportional to the volume of produced documents. The links represent the main connections among these areas.

Notice that the combination of entities co-citation, PFNETs, and Kamada–Kawai considered building this scientogram makes the most important entities in the network (i.e., those sharing more sources with the rest) tend to be placed toward the center.

Finally, concerning the analysis of scientograms, according to [20] and [50], there are three main measures of centrality that yield useful information, with the aim of detecting and identifying the most significant nodes in a PFNET: *centrality*

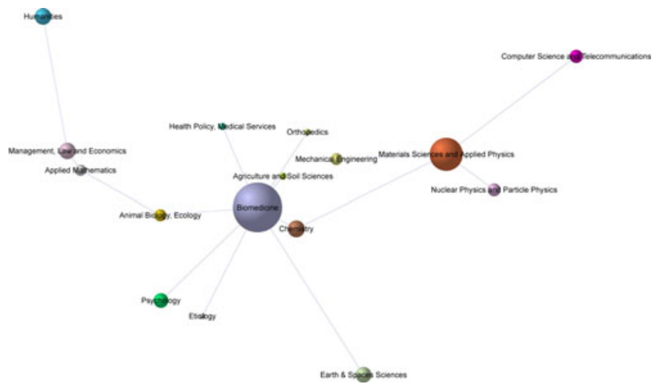


Fig. 2. Scientogram of the thematic areas of world science, 2002.

degree (regarding the number of direct links gathering in a node); *closeness centrality* (measuring the shortest paths among nodes, for which the inverse of the sum of the distance of a node to all other nodes would indicate its importance); and *intermediation centrality* or *betweenness* (looking at nodes that act as links between other nodes contained in the shortest path, for which the highest value would highlight the most central node).

III. Fuzzy Inference-Grams Design

This paper proposes a new methodology for visual representation and exploratory analysis of the fuzzy inference process in FRBSs. In such systems, various rules can be fired simultaneously by an input. Moreover, the usual behavior of FRBSs is that, given a set of problem inputs, several fuzzy rules are fired at the same time. In other words, the input space is usually covered by rules with dense overlapping among them.

In this proposal, we take advantage of this characteristic of FRBSs using a set of problem instances to uncover co-fired rules. This co-firing information is used to create social networks representing fuzzy systems' inference maps, the so-called *fingrams*. In these kinds of social networks, each fuzzy rule is represented by a node, and the relations among rules are represented by weighted edges whose value is computed using a specific metric. Different metrics can be used to construct a social network, given a dataset of cases representing the input–output relations existing in the problem tackled, a set of fuzzy rules, and a fuzzy reasoning mechanism. As a result, fingrams show graphically the interaction among fuzzy rules at the inference level in terms of co-fired rules.

Due to the high overlapping among rules, the complete fingram is usually quite dense and difficult to analyze even for medium-sized FRBSs. Fortunately, network scaling methods can be used to simplify fingrams, while maintaining their most important relations.

As seen in Section II-C, social networks can be represented by the use of drawing methods, especially designed for that purpose. Here, a specific graph representation is developed to provide the relevant information of the FRBS under study. Colors and sizes are also used to highlight distinguishing characteristics of the system, allowing the end user to do a systematic analysis.

From a formal viewpoint, the proposed fingram definition is as follows:

Definition A *fingram* is defined by a tuple $(R, P, I, E, m, NSM, NDM)$ in which:

R is the set of fuzzy rules (nodes), denoted R_i, $1 \leq i \leq r$, with r being the number of rules;

P is the set of fuzzy partitions of input and output variables;

I is the fuzzy inference mechanism used;

E is the set of problem instances, denoted E_k, $1 \leq k \leq d$, with d being the number of instances;

m is the metric used to create M, a square weight matrix $(r \times r)$ that represents the firing interactions among fuzzy rules. The entries of that matrix are the weights associated with the links; m_{ij} is the weight of the link connecting R_i and R_j;

NSM is the considered network scaling method;

NDM is the considered network drawing method.

The remainder of this section explains, in detail, the procedure followed to create fingrams and ends with an illustrative example.

A. Fingram Generation

The generation of a fingram from an FRBS, a fuzzy inference mechanism, and a set of problem instances is made by means of the following procedure.

Procedure FINGRAM(R,P,I,E,m,NSM,NDM)

```
begin
    /* Generation of the social network defined by M using the set of fuzzy
       rules R, the set of fuzzy partitions P, the fuzzy inference mechanism
       I, the set of instances E, and the metric m.                        */
    M ⟵ network generation (R, P, I, E, m)
    begin
        FR_i, FR_j ⟵ get number of fired rules (R, P, I, E);
        SFR_ij ⟵ get number of co-fired rules (R, P, I, E, m);
        M ⟵ compute M_ij (FR_i, FR_j, SFR_ij);
    /* Scaling of the social network defined by M through the use of the
       network scaling method NSM.                                         */
    MS ⟵ network scaling (M, NSM)
    begin
        EE ⟵ evaluate values of edges (M, NSM);
        MS ⟵ obtain the pruned network (M, EE);
    /* Graphical representation of the resulting pruned social network MS
       using the network drawing method NDM.                               */
    MD ⟵ network drawing (MS, NDM)
    begin
        NI ⟵ compute information related to nodes (MS);
        NP ⟵ compute the network layout (MS, NDM);
        MD ⟵ paint edges(MS, NDM, NI, NP);
```

Next, we will explain each of the steps of the procedure in detail.

1) Network Generation: Starting from a set of fuzzy rules R, a set of fuzzy partitions P, a fuzzy inference mechanism I, a set of problem instances E, and a metric m, a social network can be built, represented by a matrix M, which shows the relations among rules.

A square matrix M $(r \times r)$ that contains all interactions inside R is computed, regarding the proportion of problem instances co-firing the rules

$$M = \begin{pmatrix} 0 & m_{12} & \dots & m_{1r} \\ m_{21} & 0 & \dots & m_{2r} \\ \dots & \dots & \dots & \dots \\ m_{r1} & m_{r2} & \dots & 0 \end{pmatrix}. \quad (2)$$

We propose the following metric, which is inspired by the co-citation measure of scientograms (1):

$$m_{ij} = \begin{cases} \dfrac{\mathrm{SFR}_{ij}}{\sqrt{\mathrm{FR}_i \cdot \mathrm{FR}_j}}, & \text{if } i \neq j \\ 0, & \text{if } i = j. \end{cases} \quad (3)$$

SFR_{ij} corresponds to the number of instances for which rules R_i and R_j are fired simultaneously, while FR_i and FR_j account, respectively, for the total number of data pairs for which rules R_i or R_j are, respectively, fired, without taking care if they are fired together or not. Notice that m_{ij} is thus normalized, and the matrix M is symmetrical when using this metric.

2) Network Scaling: As usual, in a social network design, the initial fingram is commonly quite dense and difficult to analyze even for medium-sized FRBSs. Therefore, a network scaling method is required to simplify it while keeping the most important relations. Three options have been considered.

a) Prune the network to eliminate the least informative links, according to an expert. Contrary to what one may think by intuition when confronting the problem of pruning the graph, using a threshold to filter the graph is not worthy. There exist a large number of links with high weights that would imply the selection of a high threshold value, thereby producing a disconnected network. Of course, the latter does not help in the comprehension of the global system, which is our ultimate goal in this contribution.

b) Use a specific scaling algorithm that preserves the most important links without producing isolated nodes, such as Pathfinder,[2] previously introduced in Section II-C.

c) Use a combination of the previously mentioned alternatives. First, links are pruned, and then, Pathfinder scales the resulting graph. As we will show later, this hybrid option can be used to analyze classification problems. In such a case, potential inconsistencies among rules, i.e., relations among rules pointing out different classes, have to be treated carefully. Therefore, noninconsistent links can be pruned, keeping just inconsistent links. Finally, as the resulting graph is still likely to be quite complicated, Pathfinder is used to simplify it.

3) Network Drawing: As previously outlined in Section II-C, force-based algorithms are devoted to represent this kind of information in an aesthetically pleasing way. In order to visualize the pruned network in a 2-D space, they assign coordinates to the nodes obtaining a graph with the most important elements placed toward the center of the image. Kamada–Kawai through Graphviz[3] will be used in our approach because it has been proved very effective in combination with Pathfinder [20]. This solution is flexible enough to be adapted to the particularities of new scenarios with which we have to deal.

Nodes are represented by circles and labeled with useful textual information (see Fig. 3):

a) The first line shows the rule identifier R_k.

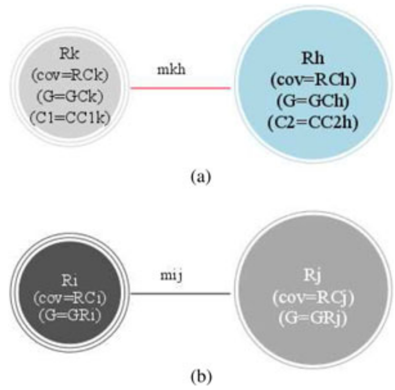


Fig. 3. Fingram's interpretation. (a) Classification. (b) Regression.

b) The second line provides the *relative coverage of that rule* (cov), i.e., the number of covered instances divided by the total number of instances. One problem instance is covered by rule R_k when the rule firing degree for that instance is greater than a predefined threshold (0.1 in this contribution)

$$\mathrm{cov}_{R_k} = \frac{\#\text{instances covered by } R_k}{\#\text{ instances}}.$$

c) The third line shows the goodness of the rule (G), i.e., how the rule behaves with respect to the problem instances available. This goodness measure reflects how well the problem instances covered by a rule are classified or modeled. It is computed as the ratio between the differences of cumulated firing degrees produced by positive instances (properly issued) and negative ones with respect to the total cumulated firing degrees regarding all covered instances. Hence, it can take values from -1 to 1, assigning -1 to rules with low number of problem instances correctly issued, and close to 1 when the rule correctly handles most problem instances

$$G_{R_k} = \frac{\sum \text{FDPI for } R_k - \sum \text{FDNI for } R_k}{\sum \text{FDCI for } R_k}$$

where FDPI stands for firing degree of positive instances; FDNI means firing degree of negative instances; and FDCI is the firing degree regarding all covered instances.

d) The fourth line of the nodes appears only in classification problems. It reflects the relative coverage of the rule output class, i.e., the number of problem instances covered by rule R_k that belong to class n divided by the total number of instances related to class n

$$C_{R_k} = \frac{\#\text{ instances of class } n \text{ covered by } R_k}{\#\text{ instances of class } n}.$$

B. Additional Fingram Visualization Capabilities

The proposed representation includes graphical information of special interest for FRBSs. Hence, once the fingram is pruned by Pathfinder and drawn by Kamada–Kawai, some additional visualization capabilities are incorporated that are specific to FRBS fuzzy inference analysis.

[2]MST-Pathfinder [52], a variant of Pathfinder that reduces the complexity of the original algorithm, is the method considered in this paper.

[3]*http://www.graphviz.org/* [53]

In this context, nodes represent the fuzzy rules of an FRBS, which are of the form:

$$R_x\text{: IF } \textit{Input}\,1 \textit{ is } LV_1 \text{ AND } \textit{Input}\,2 \textit{ is } LV_2 \text{ AND}\ldots$$

$$\ldots\text{AND } \textit{Input}\,n \textit{ is } LV_n \text{ THEN } \textit{Output is } CC$$

with (*Input i is* LV_i) being the antecedents of the fuzzy rule, and CC the output of the fuzzy rule.

The node size is established according to the number of examples covered by the rule. The higher the amount of covered examples, the bigger the node size. For instance, Fig. 3(a) shows an example of a network with two rules (R_k and R_h) where rule R_h covers more examples than rule R_k. In addition, the border of the nodes indicates how complex the antecedents of the rules are. Single-line border indicates two premises; double-line border means three premises; and so on. Thus, the rules R_k and R_h, which are depicted in Fig. 3(a), have three and two antecedents, respectively.

Furthermore, edges (links) among nodes represent rule co-firing information. Each link represents the relation between a pair of fuzzy rules. The higher the degree of overlapping existing over rules, the higher the edge weight and the thicker the link width in the visual representation to clearly represent this fact.

We deal with problems having either categorical or continuous outputs. Therefore, we distinguish between classification and regression problems, providing particularities in their representations.

1) *Classification:* Rules that yield the same class are depicted by the same color of nodes. The color of links gives useful information as well. Links between rules of the same class (output) are colored in green, while potential inconsistencies (links between co-fired rules pointing out different classes) are marked with red color [see Fig. 3(a)].
2) *Regression:* The output variable[4] is ordered in its universe of discourse. This order is used to assign gray tones to nodes, from black to white. Therefore, the typical behavior will relate nodes with similar grayness, and related nodes showing quite different tones should be studied in detail. In this case, there is no difference among links, contrary to what happens in classification problems with redundancies and inconsistencies, and they just inform about their weight [see Fig. 3(b)].

C. Illustrative Example

In this section, a fuzzy rule-based classification system (FRBCS) created for the popular WINE dataset [54] is considered. The dataset is made up of 178 examples and 13 attributes (alcohol, malic acid, ash, etc.) found in three types of wines. The FRBCS has 24 rules with three different output classes, corresponding to the three different kinds of wine.

Several fingrams are built with the aim of illustrating the effect of the different network scaling methods used. The fingram plotted in Fig. 4(a), which is obtained without applying any network scaling technique, clearly shows the previously mentioned

[4]We will only consider multi-input single-output (MISO) FRBSs.

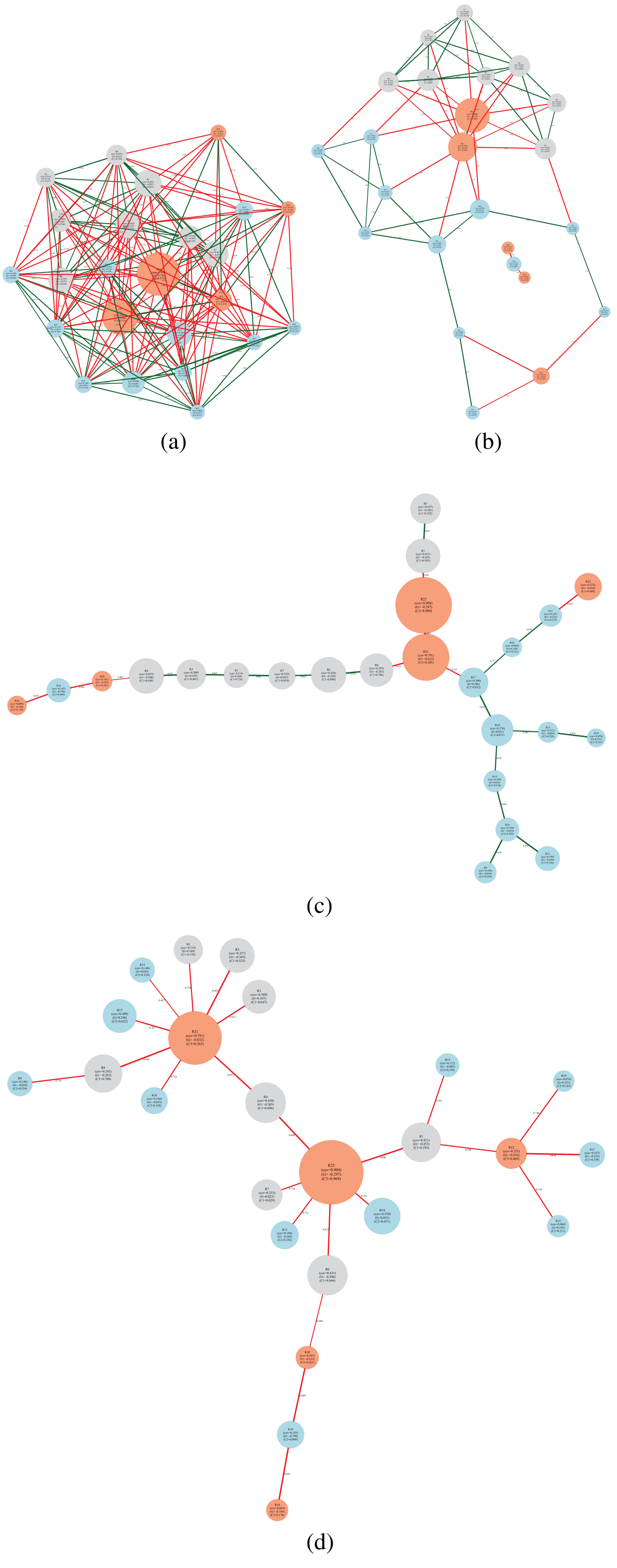


Fig. 4. Example of fingrams. (a) Complete fingram. (b) Fingram scaled using pruning with threshold ($\Delta = 0.6$). (c) Fingram scaled using Pathfinder. (d) Fingram scaled using a hybrid method: pruning + Pathfinder.

scaling motivations. A quite dense set of relationships among rules does not allow us to analyze easily the FRBCS behavior.

Then, the three scaling methods previously described are used to simplify the network. Fig. 4(b) shows the result of using a user-defined threshold ($\Delta = 0.6$) to prune edges. It can be seen how the network is still quite dense, some groups of rules are isolated, and the network is not visualized in an aesthetic way, thus hindering the comprehension of the whole set of rules. On the other hand, Fig. 4(c) shows the result of applying Pathfinder, whose global close-to-tree structure provides valuable information that is easy to interpret. As an illustration of the hybrid scaling method, Fig. 4(d) is created from the complete fingram of Fig. 4(a). There, nonpotential inconsistencies are pruned first (once we deal with a classification problem), while the resulting graph is simplified with Pathfinder. It can be seen how this graph only relates nodes of different colors (rules with a different output class).

It is remarkable that, thanks to the combination of rule co-firing, PFNETs, and Kamada–Kawai's algorithm, information related to the inference process of the FRBSs is displayed in pretty nice scalable fingrams, as seen in Fig. 4(c). As a side effect, the most relevant fuzzy rules, i.e., those more often fired, tend to be located toward the center of the scaled fingrams, while less salient ones (in this case, rules with the lowest co-firing degrees) go to the periphery. Hence, the shape of the fingram is quite informative.

Of course, fingrams must be carefully analyzed by an expert since rules that are apparently not very relevant (like those in the periphery) may be essential for properly handling important cases that only happen from time to time. For instance, uncommon cases dealing with failures in a system controlling a nuclear reactor could be extremely important.

Moreover, it is important to highlight that our proposal is not affected by the well-known curse of dimensionality that implies the number of fuzzy rules grows exponentially with the number of inputs. First, nodes directly represent fuzzy rules instead of premises. Second, PFNETs have been successfully applied to the analysis of large scientific domains with hundreds of co-cited entities (dual to our problem instances), allowing us to relate different thematic areas (dual to our fuzzy rules in the FRBS), with the chance of also considering hierarchical representations [20]. In consequence, fingrams are able to display the interactions among a few hundred rules in the form of highly interpretable trees. Even when the number of rules is huge, the scaled fingram can be still comfortably viewed by an expert.

For comparison purposes, Fig. 5 shows the same FRBS represented by the visualization method proposed by Berthold *et al.* [35]. As can be seen, this representation is mainly descriptive, placing rules in a 2-D space through a multidimensional scaling. Therefore, the distance among rules is relevant. However, it does not provide information for rule behavior at inference level. Moreover, the Delaunay triangulation indicates direct neighbors for each rule. Unfortunately, it relates rules far away in the 2-D space. Of course, that fact does not help in the comprehension of the system behavior. For example, rules R1 and R13, which do not co-fire for any problem instance [as can be seen in Fig. 4(a)], are strongly related in Fig. 5 because of their descriptive proximity.

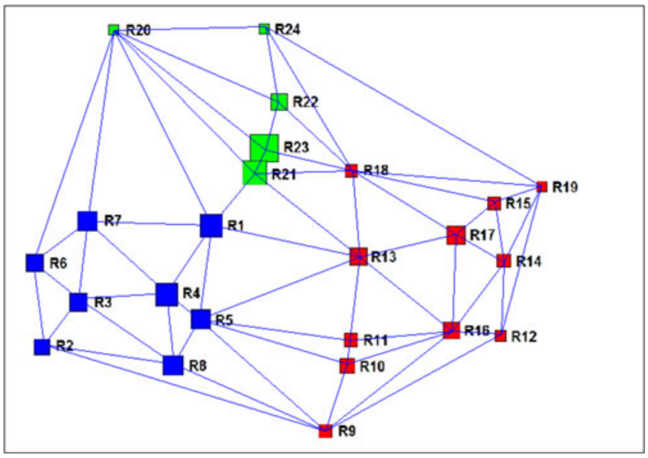


Fig. 5. Visualization of the fuzzy rule set constructed for the WINE problem using the method proposed by Berthold *et al.* [35]. It shows possible overlaps among rules, along with rule connections in terms of closeness by Delaunay triangulation.

IV. FINGRAMS EXPERT ANALYSIS

Fingrams provide enormous potential for the representation and comprehension of the FRBS inference process. They relate rules jointly fired by a given input vector, making it easy to uncover how the rules of an FRBS actually cover the input space. Hence, fingrams can be viewed as a powerful tool for dealing with FRBS comprehensibility analysis tasks that are related to quadrant Q4 (semantics at rule base level) in Fig. 1 (see Section II-A), i.e., the least studied category in the existing fuzzy system interpretability assessment literature.

The analysis of fingrams offers many different possibilities, thanks to the high amount of information this representation gives about an FRBS and its related fuzzy inference process. For instance, one can directly analyze its global structure by exploring the number and location of the apparent groups of rules (nodes), analyzing the respective location of the rules coding for different outputs, etc. As such, we would like to highlight two exploratory tasks that provide a good base for detecting and analyzing particularities or anomalies in an FRBS: 1) identifying the most significant rules in an FRBS from the inference viewpoint and 2) detecting potential inconsistencies among rules in the particular case of FRBCSs.

On the one hand, it should be reminded that, because of the specific way network scaling and drawing are done, the most salient links and nodes are likely to be placed toward the center of the graphical representation. Thus, those fuzzy rules that correspond to nodes located in the periphery of the fingram, especially those which are connected with a high weight (the value of the associated link is large) to the remaining graph nodes and show a low level of coverage (cov), are good candidates to be further studied. These rules usually cover the same space than others and do not change the final output of the system, thus not affecting the accuracy of the system. This could have an interesting collateral advantage in classification problems since removing such rules is likely to increase interpretability, while keeping almost the same accuracy. We will check that assumption in Section V.

Moreover, rules that are fired more frequently (represented with bigger nodes) are usually placed in the center because they also tend to be co-fired with more rules. Those cases where nodes covering a large number of examples are placed in the periphery must be carefully analyzed. This can be due to a fuzzy rule that covers a large part of the input space in isolation.

The usual centrality measures that are commonly considered in the analysis of scientograms [20], [50] (see Section II-D) can also be successfully applied to uncover the most significant rules within an FRBS. As a first approach, we advocate for the use of the so-called *degree of centrality*. This means that we will point out those fuzzy rules corresponding to the nodes that concentrate a larger number of links in a fingram as the most salient ones.

On the other hand, the interaction among fuzzy rules at inference level is very difficult to be appreciated by only reading the linguistic description of FRBSs. It should be remarked that this interaction depends not only on the rule description but also on the fuzzy rule semantics (fuzzy partitions included in the database) and on the inference mechanism. Even when a rule base is fully consistent at linguistic level, some possible inconsistencies may arise at inference level because of the FRBS semantics and fuzzy inference process. Such potential conflicts are difficult to detect mainly because they are partially hidden since they are typically produced by new unknown situations that were not taken into account during the learning stage (for example, data pairs not initially included when considering a data-driven FRBS derivation). Of course, such analysis is different depending on the kind of problem faced. For instance, the meaning of overlapping rules is not the same when considering either classification or regression problems.

In the former case, inconsistencies must be handled as conflicts to be solved. For instance, it may happen that several rules are jointly fired for a new given input vector, and, as a consequence, several outputs are activated with degrees higher than zero. When two different classes are activated with very similar degrees, the situation can be labeled as an ambiguous case. Such a situation is not desirable, no matter if the system is (or not) able to yield the right output class, because a slight modification in the input data may yield a wrong output. We can conclude that an FRBCS that produces many ambiguous cases is not reliable and should be corrected. Fortunately, looking at fingrams, we can easily uncover potential inconsistencies (when the co-fired rules yield different output classes). The larger the degree of inconsistency among fuzzy classification rules is, the higher the weight of the "inconsistent" links [co-firing degree computed in (3)] will become (red edges). See [55], where a detailed explanation of some possible inconsistency problems, along with a methodology to detect and correct such inconsistencies, is presented.

When dealing with regression problems, the well-known FRBS approximation capability is mainly based on the interpolative reasoning that is carried out among overlapping rules. Typically, two rules with similar premises may yield two different wrong outputs but their aggregation may result in the right inferred interpolated output. Unfortunately, these kinds of situations are quite common but very difficult to identify. Of course, from the comprehensibility point of view, it would be desirable to have only one rule that directly yields the right inferred output. However, this may produce a huge number of rules what is also undesirable. Fingrams allow the expert to study and improve the system systematically, as will be shown with an example in Section V-C.

V. Application Examples

Section V-A is devoted to the introduction of the quality indices to be considered. Then, two examples in Sections V-B and V-C, display the possibilities for considering fingrams in real-world problems. The first illustrative classification example discusses how to deal with the co-firing among rules, along with the inconsistencies and redundancies produced. The second example displays a small-sized but complex real-life regression application, where fingrams simplify the understanding of the rules constructed.

A. *Quality Indices*

We will now describe the accuracy and interpretability indices considered in this contribution.

Accuracy is computed as the percentage of misclassified instances (MC) in classification problems, and as the mean square error (MSE) in regression problems

$$\mathrm{MC} = \frac{1}{d}\sum_{i=1}^{d}\mathrm{err}_i;\quad \mathrm{err}_i = \begin{cases} 1, & \text{if } C_i \neq \hat{C}_i \\ 0, & \text{otherwise} \end{cases} \tag{4}$$

$$\mathrm{MSE} = \frac{1}{d}\sum_{i=1}^{d}(y_i - \hat{y}_i)^2 \tag{5}$$

where d means the number of problem instances, C_i is the class of instance i, and $\hat{C}_i$ is the class inferred by the FRBCS given the instance i in MC. For MSE, y_i is the real output value of instance i, and $\hat{y}_i$ is the inferred output by the FRBS.

Of course, as was pointed out in Section II-A, taking only one index is not enough to evaluate interpretability. Therefore, we have considered some of the interpretability indices that are commonly used in the literature. Probably, the most popular index is number of rules (NR). As an alternative, the total rule length (TRL) represents the total number of linguistic propositions into the whole rule base. Another simple index is average rule length (ARL), computed as TRL divided by NR. We will also report the average number of fired rules with respect to problem instances (AFR). Notice that a rule is counted as fired by a given data instance only in the case it is activated with a confidence firing degree greater or equal than a predefined threshold (0.1 in this contribution). In the case of classification problems, we will additionally compute the average confidence firing degree of winner rules (AFD). It is measured as the average of the firing degree of the winner rule for each data sample over the whole dataset.

Moreover, the proportion of co-fired rules can also be considered to evaluate the FRBS comprehensibility. The assumption is the following: the larger the number of simultaneously fired rules for a given input vector, the smaller the comprehensibility of the FRBS.

Thus, the co-firing-based comprehensibility index (COFCI) [56] can be used to evaluate the complexity of understanding the inference process in terms of rules co-firing information. Equation (6) presents this index

$$\mathrm{COFCI} = \begin{cases} 1 - \sqrt{\dfrac{\mathrm{CI}}{\mathrm{MaxThr}}}, & \text{if } \mathrm{CI} \le \mathrm{MaxThr} \\ 0, & \text{otherwise} \end{cases} \quad (6)$$

$$\mathrm{CI} = \sum_{i=1}^{r} \sum_{j=1}^{r} [(P_i + P_j) \cdot m_{ij}] \quad (7)$$

where r is the total number of rules in the fuzzy rule base, P_i and P_j count the number of premises (antecedent conditions) in rules R_i and R_j respectively, while m_{ij} is the measure of co-firing [computed by (3)] for the rules R_i and R_j, and MaxThr is a maximum value that is heuristically established to get a normalized measure in the interval [0,1].

B. Generation of Fingrams in a Simple Classification Problem: Analysis of Inconsistencies

As a first example, we will analyze a simple classification problem with two input variables, which can be represented in two dimensions, where the co-firing relations among rules can be easily understood. For that, the IRIS dataset from the University of California at Irvine [54] is considered.

IRIS is perhaps the best known database to be found in the pattern recognition literature. The dataset contains three classes of 50 instances each; therefore, it is perfectly balanced, where each class refers to a type of iris plant. Class 1 is linearly separable from the other two; the latter are not linearly separable from each other. Notice that only two of the four input variables of IRIS (SEPAL LENGTH and SEPAL WIDTH) have been used with the aim of allowing a 2-D representation that facilitates the understanding of fingram construction.

Fig. 6 shows graphically the distribution of examples, with the selected variables SEPAL LENGTH and SEPAL WIDTH, remarking the flower class ($C1 = \bigcirc$, $C2 = +$, and $C3 = \times$). Each input is characterized by a uniform SFP with three linguistic terms (LOW, AVERAGE, HIGH).

The rule base has been automatically extracted from the whole dataset, following the HILK fuzzy modeling methodology that is aimed to produce highly interpretable fuzzy systems [55], [57]. The rule base is generated by means of the fast prototyping algorithm (FPA)[5] [61]. It is made up of the following nine linguistic rules:

R_1: IF Sepal Length is Low AND Sepal Width is Low THEN Class is C2

R_2: IF Sepal Length is Low AND Sepal Width is Average THEN Class is C1

[5]We have used the implementation of FPA provided with the free software tool GUAJE [21]. Of course, other fuzzy modeling methods can be used (see, e.g., [58]–[60]).

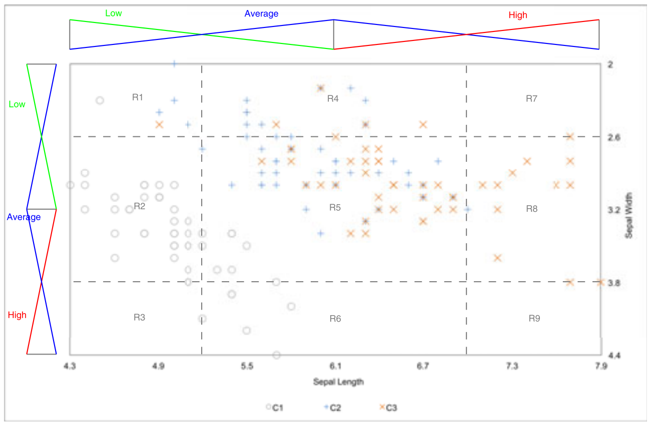


Fig. 6. Classification example. Problem instances, fuzzy partitions, and set of fuzzy rules used.

R_3: IF Sepal Length is Low AND Sepal Width is High THEN Class is C1

R_4: IF Sepal Length is Average AND Sepal Width is Low THEN Class is C2

R_5: IF Sepal Length is Average AND Sepal Width is Average THEN Class is C2

R_6: IF Sepal Length is Average AND Sepal Width is High THEN Class is C1

R_7: IF Sepal Length is High AND Sepal Width is Low THEN Class is C3

R_8: IF Sepal Length is High AND Sepal Width is Average THEN Class is C3

R_9: IF Sepal Length is High AND Sepal Width is High THEN Class is C3

It is possible to find more accurate FRBCSs for this problem in the fuzzy literature, but the objective of this example is to illustrate the creation and analysis of fingrams in classification problems.

We will detail, step by step, the different phases involved in the construction of fingrams, as they were described in Section III.

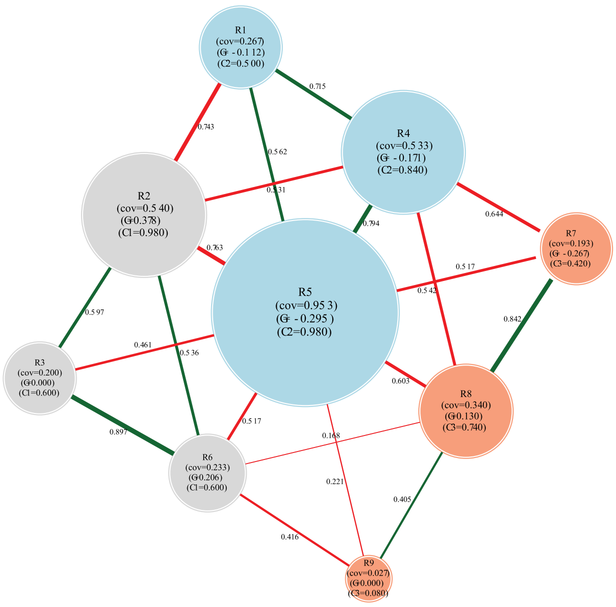
1) *Network generation:* With the problem instances, fuzzy partitions, and fuzzy rules previously presented (all of them illustrated in Fig. 6), we have generated a 9×9 matrix that represents the co-firing degrees. Fig. 7(a) shows that matrix with inconsistencies marked with an asterisk (*).
2) *Network scaling:* We have checked different scaling methods. First, Pathfinder is applied to the original network, obtaining a pruned matrix. Second, a hybrid scaling method is used to discover inconsistencies in the FRBCS. For that, noninconsistent links are first thresholded in the original network, and afterward, Pathfinder is enforced.
3) *Network drawing:* Kamada–Kawai's spring layout is selected for plotting the previously generated and scaled networks, considering the additional visualization capabilities in Section III-B.

The first graph, i.e., the complete nonscaled fingram [see Fig. 7(b)], shows the relations among rules displayed in a perfect grid, thanks to the dimensions and partitions considered.

A simple comparison between Figs. 6 and 7 makes it easy to appreciate the correspondence among the node sizes and how populated the input space regions are. For example, rule R_5 covers the central region with the largest number of instances, while rule R_9 covers the smallest amount of data samples.

	R1	R2	R3	R4	R5	R6	R7	R8	R9
R1	1.000	0.743(*)	0.000	0.715	0.562	0.000	0.000	0.000	0.000
R2	0.743(*)	1.000	0.597	0.531(*)	0.763(*)	0.536	0.000	0.000	0.000
R3	0.000	0.597	1.000	0.000	0.461(*)	0.897	0.000	0.000	0.000
R4	0.715	0.531(*)	0.000	1.000	0.794	0.000	0.644(*)	0.542(*)	0.000
R5	0.562	0.763(*)	0.461(*)	0.794	1.000	0.517(*)	0.517(*)	0.603(*)	0.221(*)
R6	0.000	0.536	0.897	0.000	0.517(*)	1.000	0.000	0.168(*)	0.416(*)
R7	0.000	0.000	0.000	0.644(*)	0.517(*)	0.000	1.000	0.842	0.000
R8	0.000	0.000	0.000	0.542(*)	0.603(*)	0.168(*)	0.842	1.000	0.405
R9	0.000	0.000	0.000	0.000	0.221(*)	0.416(*)	0.000	0.405	1.000

(a)



(b)

Fig. 7. Classification example. Original social network. (a) Co-firing matrix. (b) Complete fingram.

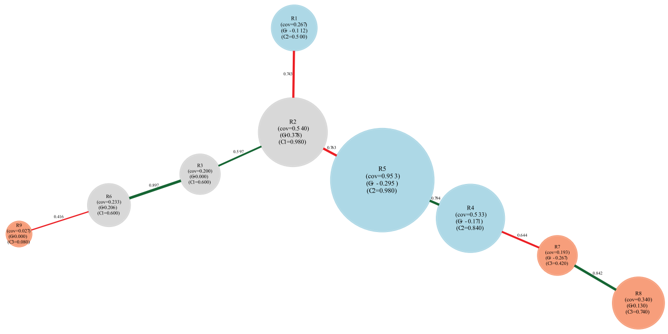


Fig. 8. Classification example. Fingram scaled with Pathfinder.

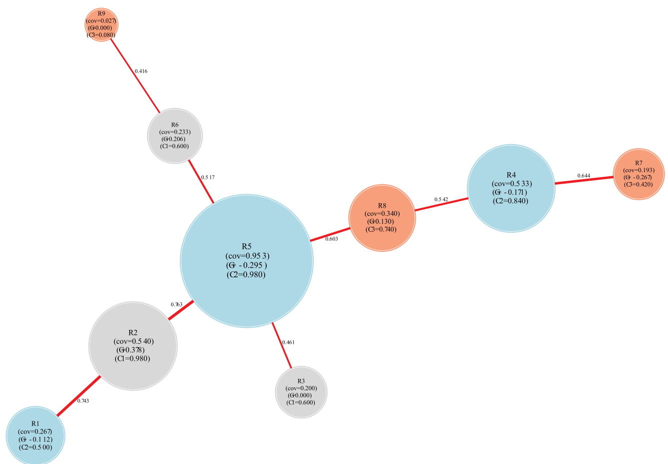


Fig. 9. Classification example. Fingram scaled with hybrid method (Threshold + Pathfinder).

In addition, the node layout perfectly reflects the relation among co-fired rules, with a central fuzzy rule (R_5) that highly overlaps with the rest, thus producing noninconsistencies (green links) or potential inconsistencies (red links).

By carefully analyzing the dataset, a high volume of instances can be appreciated in the regions of fuzzy rules R_4 and R_5 (see Fig. 6). This can also be observed in the fingram (see Fig. 7), which assigns a high value (0.794) to the connection between these two rules. In addition, the highest link weight (0.897) is related to rules R_3 and R_6, as most instances they cover are located close to the border between the input space regions they handle. Notice that a quick study of the input space can be done, even in multidimensional problems, following the same sketched procedure.

The use of Pathfinder algorithm yields a pruned fingram (see Fig. 8) that keeps the most salient links of the original network, what highlights those rules, which are fired simultaneously a larger number of times. This fingram shows that rule R_2 is quite important due to the high interrelations with others (producing inconsistencies with rules R_1 and R_5, and noninconsistencies with rule R_3).

The fingram in Fig. 9, which is scaled using the hybrid alternative with the aim of only keeping inconsistencies, emphasizes the main potential inconsistencies among rules, turning up those regions that do not belong clearly to a single class. Rule R_5 shows up as the main cause of conflicts. It is clear that this central rule covers most of the problem instances, and therefore, it overlaps with most rules. Notice that the input region covered by R_5 (as seen in Fig. 6) includes a large number of instances of different classes what produces these inconsistencies.

In addition, a linguistic simplification can be made from the previous FRBCS, which yields a new FRBCS with less rules but exactly the same accuracy:

R_1: IF Sepal Length is Low AND Sepal Width is Low THEN Class is C2

R_{23}: IF Sepal Length is Low AND Sepal Width is NOT(Low) THEN Class is C1

R_{45}: IF Sepal Length is Average AND Sepal Width is NOT(High) THEN Class is C2

R_6: IF Sepal Length is Average AND Sepal Width is High THEN Class is C1

R_{789}: IF Sepal Length is High THEN Class is C3

where R_{XY} represents the merge of original R_X and R_Y.

Fig. 10 shows the pruned fingram, which is created using Pathfinder, of the simplified FRBCS. As expected, it can be seen that the information associated with the new merged rules varies with respect to the original FRBCS (see Fig. 7) except for rules R_1 and R_6 which are unchanged. Nevertheless, it is

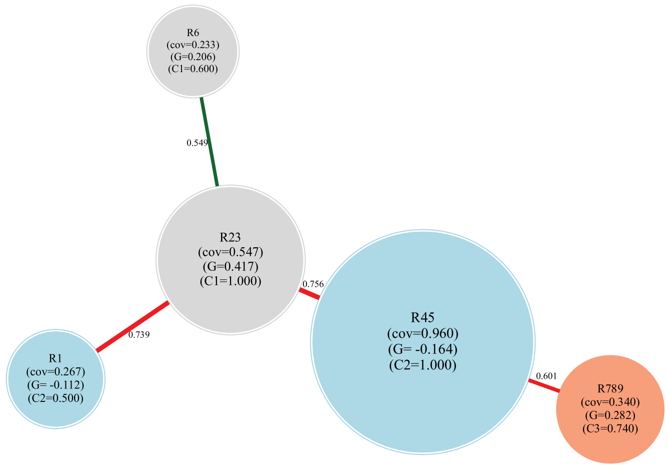


Fig. 10. Classification example. Fingram scaled with Pathfinder after linguistic simplification.

TABLE II
CLASSIFICATION EXAMPLE: QUALITY EVALUATION OF THE DIFFERENT FRBCSS GENERATED

Quality index	*Original FRBCS*	*Simplified FRBCS*	R_1 *removal*	R_6 *removal*	R_1 & R_6 *removal*
MC	0.32	0.32	**0.313**	0.34	0.333
NR	9	5	**4**	4	3
TRL	18	9	**7**	7	5
ARL	2	1.8	**1.75**	1.75	1.667
AFR	3.287	2.347	**2.08**	2.113	1.847
AFD	0.476	0.546	**0.54**	0.544	0.538
$COFCI$	0.643	0.782	**0.818**	0.815	0.86

remarkable how the new fingram in Fig. 10 keeps almost the same global shape of the original FRBCS (see Fig. 8). The new rule R_{23} gets the central position previously taken by rule R_2, distributing the remaining rules in three branches.

It can also be appreciated that rules R_{23} and R_{45} cover all the problem instances of their output classes ($C1 = 1.000$ in R_{23}, and $C2 = 1.000$ in R_{45}). Therefore, it is interesting to test the behavior of the system without the rest of the rules of output classes $C1$ and $C2$ (R_6 and R_1, respectively). With that aim, several FRBCSs are created and tested without those rules from the simplified FRBCS.

Table II summarizes the values for the quality indices in Section V-A before and after the linguistic simplification, but also after the elimination of R_1 and R_6. We should again remark that we are not focused on finding out the most accurate FRBCS for the tackled problem, but on exploring the opportunities fingrams offer.

As previously mentioned, the accuracy (see MC in Table II) is the same after applying the linguistic simplification, but the interpretability indices improve with the reduction of rules. The elimination of R_6 produces more classification errors, indicating that R_6 is the winning rule for some problem instances of class C2. Only the FRBCS produced from eliminating R_1, which is highlighted in boldface in the table, improves both the accuracy and the interpretability of the linguistically simplified FRBCS.

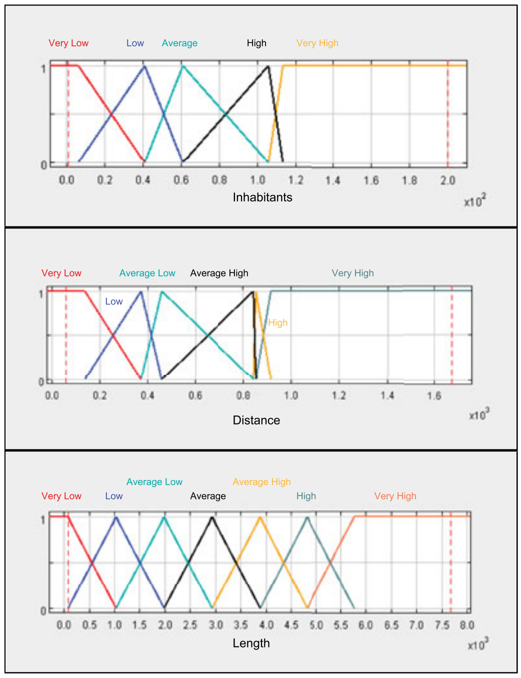


Fig. 11. Regression example. Fuzzy partitions for the electrical distribution problem.

C. Generation of Fingrams in a Small-Sized Regression Problem: Analysis of Specificity and Generality

This example illustrates the use of fingrams in regression problems. An electrical network distribution problem in northern Spain [62] is analyzed. The system aims to estimate the length of the low voltage line installed in a certain village. The problem has two input variables (the *population of the village* and its *radius*) and one output variable (the *total length of the installed line*). Real data of 495 villages are available. The training set contains 396 elements, and the test set includes 99 elements, randomly selected from the whole sample, taken from the KEEL dataset repository.[6] Here, we will use just the training set to create the fingrams, thus being able to compare the accuracy results with previous works.

First of all, the problem variables are partitioned, as shown in Fig. 11. The partitions of the input variables (INHABITANTS and DISTANCE) are tuned to improve the performance, while the output variable is partitioned homogeneously covering the interest range, i.e., the range where problem instances are located. Using these fuzzy partitions, along with FPA,[7] the following set of rules is generated:

R_1: IF Distance is Very Low THEN Length is Very Low

R_2: IF Inhabitants is (Very Low OR Low OR Average) AND Distance is Low
THEN Length is Low

R_3: IF Inhabitants is Very Low AND Distance is Average Low
THEN Length is Low

[6] *http://sci2s.ugr.es/keel/datasets.php*

[7] FPA can be used for classification and regression problems. Other fuzzy modeling methods can be used for regression problems (see [63]–[66]).

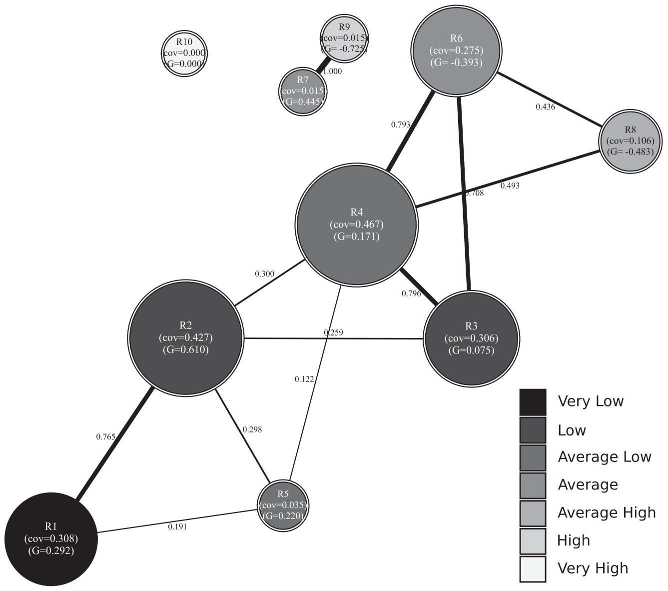


Fig. 12. Regression example. Complete fingram for the electrical distribution problem.

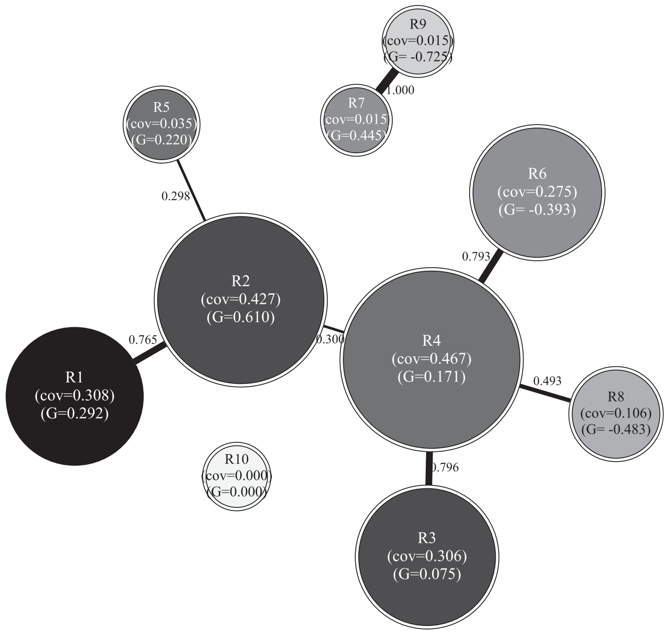


Fig. 13. Regression example. Fingram scaled with Pathfinder.

R_4: IF Inhabitants is (Low OR Average) AND Distance is Average Low THEN Length is Average Low

R_5: IF Inhabitants is High AND Distance is Low THEN Length is Average Low

R_6: IF Inhabitants is (Very Low OR Low) AND Distance is Average High THEN Length is Average

R_7: IF Inhabitants is Very High AND Distance is Average Low THEN Length is Average

R_8: IF Inhabitants is Average AND Distance is (Average High OR High) THEN Length is Average High

R_9: IF Inhabitants is Very High AND Distance is Average High THEN Length is High

R_{10}: IF Inhabitants is Very High AND Distance is High THEN Length is Very High

This FRBS exhibits a good accuracy (MSE $=$ 130 046), which is similar to the one obtained in [67] (i.e., MSE $=$ 133 763). We reiterate that we are not focused on finding the most accurate FRBS for the tackled problem. Our target is showing the utility of fingrams in the context of a real-world regression problem.

As explained in Section III-B, the output of each fuzzy rule will be reflected in the color of the nodes. From dark to light, the node colors represent a range from low to high values. Therefore, the output label "VERY LOW" will be represented by the darkest node, while "VERY HIGH" corresponds to the lightest one close to white. Naturally, the system will have relations among close labels and close colors, and when nodes of quite different darkness are related, the expert should focus her/his attention on them.

Fig. 12 shows the nonpruned fingram that is related to the inference process on the FRBS previously presented. It can be seen that the two dimensions allow the fingram to spread the nodes in a grid, relating close outputs, i.e., the evolution of darkness of the nodes is mapped smoothly. Rules R_2 and R_4 are quite general, covering almost half of the problem instances. On the contrary, rules R_5, R_7, R_9, and R_{10} cover a small amount of problem instances, thus being very specific. Moreover, it is easily appreciated that rule R_{10} does not cover any example[8] (cov $= 0$), and thus, it can be eliminated with no accuracy loss. In addition, all rules but R_1 have two antecedents, as appreciated in the single-line border of the nodes.

The fingram analysis lets us discover a special relation between rules R_7 and R_9 that appear isolated in a group, composing a kind of "fuzzy rule cluster" in a specific problem domain region. They cover some examples that no other rule covers. Moreover, they cover exactly the same examples (the related link takes value 1.0) but having different outputs. Even more, rule R_9 has a negative goodness, -0.725; therefore, it is a candidate to be removed, changing, if necessary, the output of R_7. An analysis of these rules must be achieved to avoid this kind of behavior. Notice that only looking R_7 and R_9 at the linguistic level is not enough to detect this kind of potential problem, but our fingram-based analysis methodology allows us to quickly identify them.

Fig. 13 shows the pruned network corresponding to the fingram scaled with Pathfinder. It emphasizes a high relation among rules R_3, R_4, and R_6. This interrelation suggests merging the three rules into a single one. To do so, a new rule R_{346} is constructed from R_3, R_4, and R_6 in an expert way. The antecedents of all these rules are combined, and the output is taken from the middle term. This is done just as an example,

[8]As explained in Section III-A, we consider an instance is covered by a rule when it fires the rule above a threshold (0.1 in this contribution).

TABLE III
REGRESSION EXAMPLE: QUALITY EVALUATION OF THE GENERATED FRBSS

Quality index	*Original FRBS*	R_{10} *removal*	R_9 *removal*	R_3-R_4-R_6 *fusion*
MSE	130,046	130,046	**125,511**	155,838
NR	10	9	**8**	6
TRL	19	17	**15**	11
ARL	1.9	1.889	**1.875**	1.83
AFR	2.463	2.463	**2.446**	1.695
$COFCI$	0.971	0.971	**0.974**	0.981

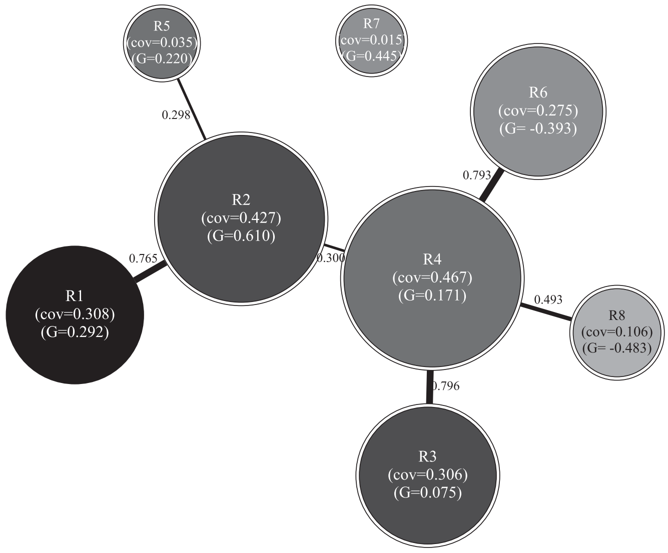


Fig. 14. Regression example. Fingram scaled of the best simplified FRBS.

and a more complex process, testing the alternatives, could be done.

R_3 : IF Inhabitants is Very Low AND Distance is Average Low THEN Length is Low

R_4 : IF Inhabitants is (Low OR Average) AND Distance is Average Low THEN Length is Average Low

R_6 : IF Inhabitants is (Very Low OR Low) AND Distance is Average High THEN Length is Average

R_{346} : **IF Inhabitants is (Very Low OR Low OR Average) AND Distance is (Average Low OR Average High) THEN Length is Average Low**

We will develop the proposed changes in a sequential fashion (i.e., first removing R_{10}, then removing R_9, and, finally, merging R_3, R_4, and R_6) and check how they affect the resulting FRBS accuracy and interpretability (as detailed in Table III).

Analyzing these results, we can conclude that the removal of R_{10} does not change the behavior of the system because, as mentioned, it does not cover any problem instance. Thus, MSE, AFR, and COFCI remain the same, while the interpretability indices related to transparency (NR, TRL, and ARL) are improved. However, deleting the rule R_9 simplifies the FRBS, improving both accuracy (MSE decreases) and interpretability (all the considered interpretability indices get better values). The new fingram that results from these two eliminations can be observed in Fig. 14. Finally, although the fusion of R_3, R_4, and R_6 reduces the accuracy of the FRBS, it could still be a good option for getting a more compact and understandable FRBS (notice that all the interpretability indices are clearly improved). Besides, a more elaborated rule fusion mechanism could be considered by the expert to reduce the accuracy loss.

VI. CONCLUSION AND FUTURE WORKS

This paper has introduced fingrams as a new powerful methodology for exploratory analysis of fuzzy rule bases. A brief overview of the possibilities that fingrams offer, for both design and analysis of fuzzy systems, has been illustrated through some examples. As it is a novel proposal, some of the potential uses are just outlined, opening the door to new alternatives and developments.

In the future, we will extensively validate and extend the methodology. For instance, we plan to look for asymmetrical co-firing metrics that are able to yield additional information about consistency, generality, and/or specificity of rules.

The future of this methodology is very promising, with several applications to design or improve fuzzy systems. The human-centric simplification of an FRBS by means of the elimination or modification of rules could be done after analyzing the resulting graphs. The detection of rules that do not cover any example is very easy by just looking fingrams at first sight. Rules that have a low overlapping with others can be detected to proceed as desired, building, maybe, more general rules.

A basic simplification procedure may consist of finding and removing those nonrelevant rules that are normally located at the periphery of the graph. Moreover, by carefully looking at fingrams, we can first set a ranking of rules, according to their relevance and, then, run a linguistic simplification procedure like the one proposed in [57].

A first software package for fingrams generation and analysis is already implemented [68] as part of the GUAJE tool, which is freely downloadable as open source software at http://www.softcomputing.es/guaje.

ACKNOWLEDGMENT

The authors would like to thank M. Berthold for providing them with the software to create the models of visualization proposed in [35], and also for his kind and quick response.

REFERENCES

[1] J. M. Alonso, L. Magdalena, and G. González-Rodríguez, “Looking for a good fuzzy system interpretability index: An experimental approach,” *Int. J. Approx. Reason.*, vol. 51, pp. 115–134, 2009.

[2] L. A. Zadeh, “The concept of a linguistic variable and its application to approximate reasoning,” *Parts I, Inf. Sci.*, vol. 8, pp. 199–249, 1975.

[3] L. A. Zadeh, “The concept of a linguistic variable and its application to approximate reasoning,” *Part II., Inf. Sci.*, vol. 8, 301–357, 1975.

[4] L. A. Zadeh, “The concept of a linguistic variable and its application to approximate reasoning,” *Part III, Inf. Sci.*, vol. 9, pp. 43–80, 1975.

[5] D. Nauck and R. Kruse, “Obtaining interpretable fuzzy classification rules from medical data,” *Artif. Intell. Med.*, vol. 16, pp. 149–169, 1999.

[6] L. A. Zadeh, “Outline of a new approach to the analysis of complex systems and decision processes,” *IEEE Trans. Syst., Man Cybern.*, vol. SMC-3, no. 1, pp. 28–44, Jan. 1973.

[7] E. H. Mamdani, “Application of fuzzy logic to approximate reasoning using linguistic synthesis,” *IEEE Trans. Comput.*, vol. C-26, no. 12, pp. 1182–1191, Dec. 1977.

[8] J. V. de Oliveira, “Semantic constraints for membership function optimization,” *IEEE Trans. Syst., Man, Cybern. A, Syst., Humans*, vol. 29, no. 1, pp. 128–138, Jan. 1999.

[9] C. Mencar and A. M. Fanelli, "Interpretability constraints for fuzzy information granulation," *Inf. Sci.*, vol. 178, no. 24, pp. 4585–4618, 2008.

[10] S. M. Zhou and J. Q. Gan, "Low-level interpretability and high-level interpretability: A unified view of data-driven interpretable fuzzy system modelling," *Fuzzy Sets Syst.*, vol. 159, no. 23, pp. 3091–3131, 2008.

[11] S. Guillaume, "Designing fuzzy inference systems from data: An interpretability-oriented review," *IEEE Trans. Fuzzy Syst.*, vol. 9, no. 3, pp. 426–443, Jun. 2001.

[12] J. Casillas, O. Cordón, F. Herrera, and L. Magdalena, Eds., Interpretability Issues in Fuzzy Modeling. (Studies in Fuzziness and Soft Computing Series), vol. 28. Berlin, Germany: Springer-Verlag, 2003.

[13] M. J. Gacto, R. Alcalá, and F. Herrera, "Integration of an index to preserve the semantic interpretability in the multiobjective evolutionary rule selection and tuning of linguistic fuzzy systems," *IEEE Trans. Fuzzy Syst.*, vol. 18, no. 3, pp. 515–531, Jun. 2010.

[14] C. Mencar, C. Castiello, R. Cannone, and A. M. Fanelli, "Interpretability assessment of fuzzy knowledge bases: A cointension based approach," *Int. J. Approx. Reason.*, vol. 52, no. 4, pp. 501–518, 2011.

[15] G. A. Miller, "The magical number seven, plus or minus two: Some limits on our capacity for processing information," *Psychol. Rev.*, vol. 63, no. 2, pp. 81–97, 1956.

[16] J. M. Alonso, O. Cordón, A. Quirin, and L. Magdalena, "Analyzing interpretability of fuzzy rule-based systems by means of fuzzy inference-grams," in *Proc. World Conf. Soft Comput.*, San Francisco, CA, USA, 2011, pp. 181–185.

[17] D. J. de Solla Price, *Little Science, Big Science*. New York, NY, USA: Columbia Univ. Press, 1986.

[18] J. Scott, *Social Network Analysis: A Handbook*, 2nd ed. New Delhi, India, Sage, 2000.

[19] S. Wasserman and K. Faust, *Social Network Analysis: Methods and Applications (Structural Analysis in the Social Sciences)*. Cambridge, U.K.: Cambridge Univ. Press, 1994.

[20] B. Vargas-Quesada and F. Moya-Anegón, *Visualizing the Structure of Science*. New York, NY, USA: Springer-Verlag, 2007.

[21] J. M. Alonso and L. Magdalena, "Generating understandable and accurate fuzzy rule-based systems in a java environment," in *Fuzzy Logic and Applications* (Lecture Notes in Computer Science Series), vol. 6857. Berlin, Germany: Springer-Verlag, 2011, pp. 212–219.

[22] M. J. Gacto, R. Alcalá, and F. Herrera, "Interpretability of linguistic fuzzy rule-based systems: An overview of interpretability measures," *Inf. Sci.*, vol. 181, no. 20, pp. 4340–4360, 2011.

[23] E. H. Ruspini, "A new approach to clustering," *Inf. Control*, vol. 15, no. 1, pp. 22–32, 1969.

[24] P. Fazendeiro, J. V. de Oliveira, and W. Pedrycz, "A multiobjective design of a patient and anaesthetist-friendly neuromuscular blockade controller," *IEEE Trans. Biomed. Eng.*, vol. 54, no. 9, pp. 1667–1678, Sep. 2007.

[25] R. Alcalá, J. Alcalá-Fdez, and F. Herrera, "A proposal for the genetic lateral tuning of linguistic fuzzy systems and its interaction with rule selection," *IEEE Trans. Fuzzy Syst.*, vol. 15, no. 4, pp. 616–635, Aug. 2007.

[26] J. M. Alonso, L. Magdalena, and O. Cordón, "Embedding HILK in a three-objective evolutionary algorithm with the aim of modeling highly interpretable fuzzy rule-based classifiers," in *Proc. 4th Int. Workshop Genet. Evol. Fuzzy Syst.*, Mar. 2010, pp. 15–20.

[27] F. Cheong and R. Lai, "Constraining the optimization of a fuzzy logic controller using an enhanced genetic algorithm," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 30, no. 1, pp. 31–46, Feb. 2000.

[28] A. Márquez, F. Márquez, and A. Peregrín, "A multi-objective evolutionary algorithm with an interpretability improvement mechanism for linguistic fuzzy systems with adaptive defuzzification," in *Proc. IEEE Int. Conf. Fuzzy Syst.*, Jul. 2010, pp. 277–283.

[29] C. Moraga, "An essay on the interpretability of Mamdani systems," in *Combining Experimentation and Theory* (Studies in Fuzziness and Soft Computing Series). vol. 271. Berlin, Germany: Springer, 2012, pp. 61–72.

[30] H. Ishibuchi, Y. Kaisho, and Y. Nojima, "Designing fuzzy rule-based classifiers that can visually explain their classification results to human users," in *Proc. 3rd Int. Workshop Genet. Evol. Fuzzy Syst.*, 2008, pp. 5–10.

[31] H. Ishibuchi, Y. Kaisho, and Y. Nojima, "A visual explanation system for explaining fuzzy reasoning results by fuzzy rule-based classifiers," in *Proc. Annu. Meet. North Amer. Fuzzy Inf. Process. Soc.*, 2008, pp. 1–6.

[32] H. Ishibuchi, Y. Kaisho, and Y. Nojima, "Complexity, interpretability and explanation capability of fuzzy rule-based classifiers," in *Proc. IEEE Int. Conf. Fuzzy Syst.*, Aug. 2009, pp. 1730–1735.

[33] B. Pham and R. Brown, "Analysis of visualisation requirements for fuzzy systems," in *Proc. 1st Int. Conf. Comput. Graph. Interact. Tech.*, 2003, pp. 181–187.

[34] M. R. Berthold and L. O. Hall, "Visualizing fuzzy points in parallel coordinates," *IEEE Trans. Fuzzy Syst.*, vol. 11, no. 3, pp. 369–374, Jun. 2003.

[35] T. R. Gabriel, K. Thiel, and M. R. Berthold, "Rule visualization based on multi-dimensional scaling," in *Proc. IEEE Int. Conf. Fuzzy Syst.*, 2006, pp. 66–71.

[36] A. G. Evsukoff, S. Galichet, B. S. L. P. de Lima, and N. F. F. Ebecken, "Design of interpretable fuzzy rule-based classifiers using spectral analysis with structure and parameters optimization," *Fuzzy Sets Syst.*, vol. 160, no. 7, pp. 857–881, 2009.

[37] A. G. Evsukoff, A. Branco, and S. Galichet, "Intelligent data analysis and model interpretation with spectral analysis fuzzy symbolic modeling," *Int. J. Approx. Reason.*, vol. 52, no. 6, pp. 728–750, 2011.

[38] J. Casillas and F. Martínez-López, "Mining uncertain data with multiobjective genetic fuzzy systems to be applied in consumer behaviour modelling," *Expert Syst. Appl.*, vol. 36, no. 2, pp. 1645–1659, 2009.

[39] C. Chen and S. Morris, "Visualizing evolving networks: Minimum spanning trees versus pathfinder networks," in *Proc. IEEE Symp. Inf. Vis.*, Oct. 2003, pp. 67–74.

[40] M. Zizi and M. Beaudouin-Lafon, "Accessing hyperdocuments through interactive dynamic maps," in *Proc. ACM Eur. Conf. Hypermedia Technol.*, 1994, pp. 126–135.

[41] S. Noel, C. H. Chu, and V. Raghavan, "Visualization of document co-citation counts," in *Proc. IEEE Symp. Inf. Vis.*, 2002, pp. 691–696.

[42] D. W. Dearholt and R. W. Schvaneveldt, "Properties of pathfinder networks," in *Pathfinder Associative Networks: Studies in Knowledge Organization*, R. Schvaneveldt, Ed. Norwood, NJ, USA: Ablex, 1990, pp. 1–30.

[43] R. W. Schvaneveldt, F. T. Durso, and D. W. Dearholt, "Network structures in proximity data," in *The Psychology of Learning and Motivation: Advances in Research and Theory*, vol. 24, New York, NY, USA, 1989, pp. 249–284.

[44] G. di Battista, P. Eades, R. Tamassia, and I. Tollis, *Graph Drawing: Algorithms for the Visualization of Graphs*. Upper Saddle River, NJ, USA: Prentice–Hall, 1998.

[45] S. G. Kobourov, "Force-directed drawing algorithms," in *Handbook of Graph Drawing and Visualization*, R. Tamassia, Ed. Boca Raton, FL, USA: CRC, 2012, ch. 12.

[46] T. Kamada and S. Kawai, "An algorithm for drawing general undirected graphs," *Inf. Process. Lett.*, vol. 31, no. 1, pp. 7–15, 1989.

[47] T. M. J. Fruchterman and E. M. Reingold, "Graph drawing by force-directed placement," *Softw. Pract. Experience*, vol. 21, no. 11, pp. 1129–1164, 1991.

[48] E. Serrano, A. Quirin, J. Botia, and O. Cordón, "Debugging complex software systems by means of pathfinder networks," *Inf. Sci.*, vol. 180, no. 5, pp. 561–583, 2010.

[49] F. Moya-Anegón, B. Vargas-Quesada, V. Herrero-Solana, Z. Chinchilla-Rodríguez, E. Corera-Álvarez, and F. J. Muñoz-Fernández, "A new technique for building maps of large scientific domains based on the cocitation of classes and categories," *Scientometr.*, vol. 61, no. 1, pp. 129–145, 2004.

[50] F. Moya-Anegón, B. Vargas-Quesada, Z. Chinchilla-Rodríguez, E. Corera-Álvarez, F. J. Muñoz-Fernández, and V. Herrero-Solana, "Visualizing the marrow of science," *J. Amer. Soc. Inf. Sci. Technol.*, vol. 58, no. 14, pp. 2167–2179, 2007.

[51] G. Salton and D. Bergmark, "A citation study of computer science literature," *IEEE Trans. Prof. Commun.*, vol. 22, no. 3, pp. 146–158, Sep. 1979.

[52] A. Quirin, O. Cordón, J. Santamaría, B. Vargas-Quesada, and F. Moya-Anegón, "A new variant of the pathfinder algorithm to generate large visual science maps in cubic time," *Inf. Process. Manag.*, vol. 44, no. 4, pp. 1611–1623, 2008.

[53] E. R. Gansner and S. C. North, "An open graph visualization system and its applications to software engineering," *Softw. Pract. Experience*, vol. 30, no. 11, pp. 1203–1233, 2000.

[54] A. Frank and A. Asuncion. (2010). UCI machine learning repository. [Online]. Available: http://archive.ics.uci.edu/ml

[55] J. M. Alonso, L. Magdalena, and S. Guillaume, "HILK: A new methodology for designing highly interpretable linguistic knowledge bases using the fuzzy logic formalism," *Int. J. Intell. Syst.*, vol. 23, no. 7, pp. 761–794, 2008.

[56] J. M. Alonso, D. P. Pancho, O. Cordón, A. Quirin, and L. Magdalena, "Social network analysis of co-fired fuzzy rules," in *Soft Computing: State of the Art Theory and Novel Applications*, R. R. Yager, A. M. Abbasov, M. Reformat, and S. N. Shahbazova, Eds. Berlin, Germany: Springer, 2013, pp. 113–128.

[57] J. M. Alonso and L. Magdalena, "HILK++: An interpretability-guided fuzzy modeling methodology for learning readable and comprehensible

fuzzy rule-based classifiers," *Soft Comput.*, vol. 15, no. 10, pp. 1959–1980, 2011.

[58] H. Bustince, M. Pagola, R. Mesiar, E. Hullermeier, and F. Herrera, "Grouping, overlap, and generalized bientropic functions for fuzzy modeling of pairwise comparisons," *IEEE Trans. Fuzzy Syst.*, vol. 20, no. 3, pp. 405–415, Jun. 2012.

[59] M. Kumar, S. Neubert, S. Behrendt, A. Rieger, M. Weippert, N. Stoll, K. Thurow, and R. Stoll, "Stress monitoring based on stochastic fuzzy analysis of heartbeat intervals," *IEEE Trans. Fuzzy Syst.*, vol. 20, no. 4, pp. 746–759, Aug. 2012.

[60] A. Lemos, W. Caminhas, and F. Gomide, "Multivariable Gaussian evolving fuzzy modeling system," *IEEE Trans. Fuzzy Syst.*, vol. 19, no. 1, pp. 91–104, Feb. 2011.

[61] P. Y. Glorennec, *Algorithmes D'apprentissage Pour Systèmes D'inférence Floue*. London, U.K.: Hermès, 1999.

[62] O. Cordón, F. Herrera, and L. Sánchez, "Solving electrical distribution problems using hybrid evolutionary data analysis techniques," *Appl. Intell.*, vol. 10, no. 1, pp. 5–24, 1999.

[63] Z. Deng, K.-S. Choi, F.-L. Chung, and S. Wang, "Scalable TSK fuzzy modeling for very large datasets using minimal-enclosing-ball approximation," *IEEE Trans. Fuzzy Syst.*, vol. 19, no. 2, pp. 210–226, Apr. 2011.

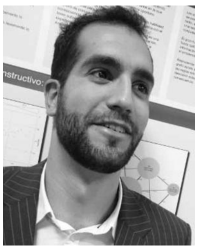
[64] M. H. Mashinchi, M. A. Orgun, M. Mashinchi, and W. Pedrycz, "A tabu-harmony search-based approach to fuzzy linear regression," *IEEE Trans. Fuzzy Syst.*, vol. 19, no. 3, pp. 432–448, Jun. 2011.

[65] C.-F. Juang and C.-D. Hsieh, "A fuzzy system constructed by rule generation and iterative linear SVR for antecedent and consequent parameter optimization," *IEEE Trans. Fuzzy Syst.*, vol. 20, no. 2, pp. 372–384, Apr. 2012.

[66] R. Alcalá, M. J. Gacto, and F. Herrera, "A fast and scalable multiobjective genetic fuzzy system for linguistic fuzzy modeling in high-dimensional regression problems," *IEEE Trans. Fuzzy Syst.*, vol. 19, no. 4, pp. 666–681, Aug. 2011.

[67] O. Cordón, F. Herrera, L. Magdalena, and P. Villar, "A genetic learning process for the scaling factors, granularity and contexts of the fuzzy rule-based system data base," *Inf. Sci.*, vol. 136, nos. 1–4, pp. 85–107, 2001.

[68] J. M. Alonso, D. P. Pancho, and L. Magdalena, "Enhancing the fuzzy modeling tool GUAJE with a new module for fingrams-based analysis of fuzzy rule bases," in *Proc. IEEE Int. Conf. Fuzzy Syst. World Congr. Comput. Intell.*, Jun. 2012, pp. 1082–1089.



David P. Pancho received the M.S. degree in computer engineering from the University of Oviedo, Oviedo, Spain, in 2010 and the M.S. degree in soft computing and intelligent data analysis, jointly from the European Centre for Soft Computing, Mieres, Spain, and the University of Oviedo, in 2011. Since 2011, he has been working toward the Ph.D. degree in computer science with the European Centre for Soft Computing, where the topic of his research is knowledge extraction and representation in fuzzy modeling, focusing on the visual representation of this kind of knowledge.

In 2008, he received a grant from the European Centre for Soft Computing to work with the "Fundamentals of Soft Computing" Unit. From 2008 to 2011, he was involved in research in the area of computing with words, actions, and perceptions.

Mr. Pancho received a diploma for the Best Transcript from the University of Oviedo in 2010 and the Best Paper Award from the XV Congreso Español de Tecnologías y Lógica Fuzzy, Huelva, Spain, in 2010.

Jose M. Alonso (M'10) received the M.S. and Ph.D. degrees in telecommunication engineering from the Technical University of Madrid, Madrid, Spain, in 2003 and 2007, respectively.

From November 2007 to January 2012, he was a Postdoctoral Researcher with the European Centre for Soft Computing, Mieres, Spain, where he is currently an Associate Researcher with the "Cognitive Computing: Computing with Perceptions" Unit. From February to October 2012, he was a Postdoctoral Researcher with the Robesafe Group, Department of Electronics, University of Alcalá, Madrid, Spain, under the JCI-2011-09839 project. His research interests include fuzzy modeling for control and classification problems, assessing interpretability of fuzzy systems, knowledge extraction and representation, integration of expert and induced knowledge, advance multisensor fusion, WiFi localization, autonomous robotic navigation in complex environments, and the development of free software tools.

Dr. Alonso has been the Vice Chair of the IEEE Computational Intelligence Society Task Force on Fuzzy Systems Software since 2011.

Oscar Cordón (SM'11) received the M.S. and Ph.D. degrees in computer science from the University of Granada, Granada, Spain, in 1994 and 1997, respectively.

He is currently a Full Professor with the University of Granada, Granada, Spain, where he was the Founder and Leader of the Virtual Learning Center between 2001 and 2005. From 2006 to 2011, he was one of the founding researchers of the European Centre for Soft Computing, Mieres, Spain, where he was a Principal Researcher and is still contracted as a Scientific Consultant. He has been, for more than 15 years, an internationally recognized contributor to R&D Programs in fuzzy systems, evolutionary algorithms, and ant colony optimization. He has published approximately 280 peer-reviewed scientific publications, including a research book on genetic fuzzy systems and 66 JCR-SCI-indexed journal papers. He has advised 13 Ph.D. dissertations, participated in 34 research projects and contracts (coordinating 20 of them), co-edited eight special issues in international journals and three research books, and given 12 plenary talks at national and international conferences. By October 2012, his publications had received 1956 citations, carrying an h-index of 23; he is, therefore, included in the 1% of most-cited researchers in the world (source: *Thomson Reuters's Web of Knowledge*). He also has an approved international patent on an intelligent system for forensic identification that is being commercialized in Mexico.

Prof. Cordón is currently an Associate Editor of 11 international journals and was recognized as the IEEE TRANSACTIONS ON FUZZY SYSTEMS Outstanding Associate Editor in 2008. He is a reviewer for more than 30 international journals. Since 2004, he has taken many different representative positions with Eusflat and the IEEE Computational Intelligence Society (CIS). From 2010 to 2012, he was a member of the IEEE CIS Administrative Committee. He received the Undergraduate Research Program Young Researcher Career Award in 2004. In July 2010, he received the IEEE CIS Outstanding Early Career Award in its 2011 edition, which was the first such award conferred. In 2011, his developments on soft computing methods in forensic identification were recognized with the International Fuzzy Systems Association Award for Outstanding Applications of Fuzzy Technology and with the granting of the European Society for Fuzzy Logic and Technology Best Ph.D. Thesis Award to Dr. Ibañez's Ph.D. dissertation, to whom he was a co-advisor.

Arnaud Quirin received the M.S. and Ph.D. degrees in computer science from the University Louis Pasteur of Strasbourg, Strasbourg, France, in 2002 and 2005, respectively.

He is currently a Data Mining Researcher with the Galician Research and Development Center in Advanced Telecommunications, Vigo, Spain. From 2006 to 2012, he was with the European Centre for Soft Computing, Mieres, Spain. He was a teaching and Research Assistant with the University Louis Pasteur of Strasbourg until September 2006. He has published more than 30 peer-reviewed scientific publications, including three book chapters and eight JCR-SCI-indexed journal papers. He was involved in several national projects, private contracts, and one European project related to the application of evolutionary algorithms to image classification. He currently participates in several R&D programs in the fields of evolutionary algorithms, multiobjective graph-based mining, fuzzy multiclassifier systems, and genetic fuzzy algorithms.

Dr. Quirin is a Reviewer for six international journals.

Luis Magdalena (SM'05) received the M.S. and Ph.D. degrees in telecommunication engineering from the Technical University of Madrid (UPM), Madrid, Spain, in 1988 and 1994, respectively.

Since 1995, he has been an Associate Professor in computer science with the Department of Applied Mathematics, UPM, where he was an Assistant Professor in 1990. In 1998, he co-founded the company Daedalus (a spin-off of UPM). From 2003 to 2005, he was the Deputy Director (and promoter) of the Centro de Domótica Integral (Smart House Research Center), UPM. In 2006, he joined the European Center for Soft Computing as a Director General. He has been actively involved in research projects funded by national and European programs, as well as by private companies, being project leader in more than 15 of those projects. He is the co-author of the book *Genetic Fuzzy Systems: Evolutionary Tuning and Learning of Fuzzy Knowledge Bases* (Singapore: World Scientific, 2001). He is the co-editor of the books *Technologies for Constructing Intelligent Systems* (two volumes, Heidelberg, Germany: Physica Verlag, 2002), *Interpretability Issues in Fuzzy Modeling* (Studies in Fuzziness and Soft Computing Series) (Berlin, Germany: Springer, 2003), *Accuracy Improvements in Linguistic Fuzzy Modeling* (Studies in Fuzziness and Soft Computing Series) (Berlin, Germany: Springer, 2003), and *Foundations of Reasoning under Uncertainty* (Studies in Fuzziness Series) (Berlin, Germany: Springer, 2010). He has also authored more than 150 papers in books, journals, and conferences and has edited special issues of different journals. His research interests include soft computing and its application, particularly in the field of robotics and automation.

Dr. Magdalena was an Area Editor for engineering applications of the *International Journal of Uncertainty, Fuzziness and Knowledge Based Systems* and a member of the editorial board or regular reviewer for more than ten international journals. From 2001 to 2005, he was President of the European Society for Fuzzy Logic and Technologies. From 2007 to 2011, he was Vice President of the International Fuzzy Systems Association. He is a member of the Administrative Committee from 2011 to 2013 and was the Chair of the University Curricula Sub-Committee of the IEEE Computational Intelligence Society from 2010 to 2011. He is also a member of the IEEE Systems, Man, and Cybernetics Society. Since 2009, he has been a member of the Scientific Advisory Committee of the Spanish Research Council. He has been a member of the Program Committee of more than 30 national and international conferences, serving as the General, Program, or Publicity Chair for some of them.

A.1.2 JCR of IEEE Transactions on Fuzzy Systems

Journal Citation Reports®

2013 JCR Science Edition

Journal: IEEE TRANSACTIONS ON FUZZY SYSTEMS

Journal Impact Factor

Cites in 2013 to items published in:		Number of items published in:	
	2012 = 513		2012 = 93
	2011 = 660		2011 = 93
	Sum: 1173		Sum: 186

Calculation: Cites to recent items / Number of recent items = 1173 / 186 = **6.306**

Journal Ranking

For **2013**, the journal **IEEE TRANSACTIONS ON FUZZY SYSTEMS** has an Impact Factor of **6.306**.

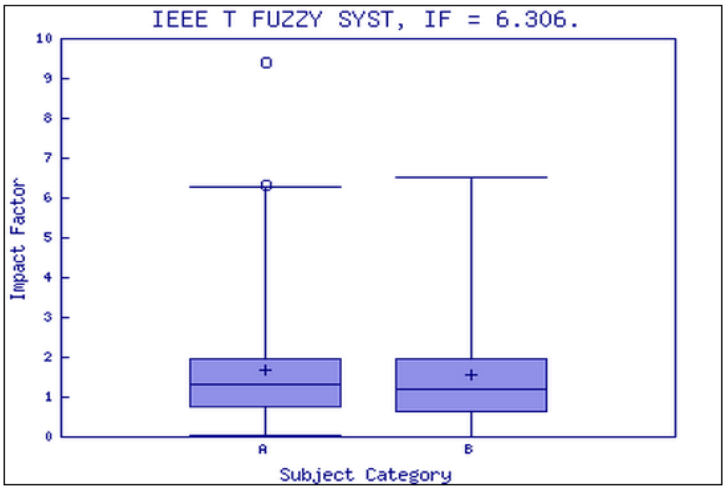
This table shows the ranking of this journal in its subject categories based on Impact Factor.

Category Name	Total Journals in Category	Journal Rank in Category	Quartile in Category
COMPUTER SCIENCE, ARTIFICIAL INTELLIGENCE	121	2	Q1
ENGINEERING, ELECTRICAL & ELECTRONIC	248	3	Q1

Category Box Plot

For **2013**, the journal **IEEE TRANSACTIONS ON FUZZY SYSTEMS** has an Impact Factor of **6.306**.

This is a box plot of the subject category or categories to which the journal has been assigned. It provides information about the distribution of journals based on Impact Factor values. It shows median, 25th and 75th percentiles, and the extreme values of the distribution.



Key

A - COMPUTER SCIENCE, ARTIFICIAL INTELLIGENCE

B - ENGINEERING, ELECTRICAL & ELECTRONIC

A.1.3 Second publication: Software

[67] Pancho, D. P., Alonso, J. M., and Magdalena, L. Quest for interpretability-accuracy trade-off supported by Fingrams into the fuzzy modeling tool GUAJE. *International Journal of Computational Intelligence Systems 6*, sup1 (2013), 46–60

This paper introduces a new module for Fingram generation and analysis included in the free software tool GUAJE. This tool aims to design, analyze and evaluate fuzzy systems with good interpretability-accuracy trade-off. In addition, GUAJE includes several intuitive and interactive tutorials to uncover the possibilities it offers. One of them generates and enhances a fuzzy system, analyzing each improvement through the use of Fingrams, and lets the user reproduce the illustrative case study described in this paper.

Citations Google Scholar:	6
Citations Web of Science:	2

International Journal of Computational Intelligence Systems, Vol. 6, Supplement 1 (2013), 46-60

Quest for Interpretability-Accuracy Trade-off Supported by Fingrams into the Fuzzy Modeling Tool GUAJE

David P. Pancho, Jose M. Alonso, L. Magdalena

European Centre for Soft Computing,
Edificio de Investigación, C/ Gonzalo Gutiérrez Quirós s/n,
33600 Mieres, Asturias, Spain
E-mail: {david.perez,jose.alonso,luis.magdalena}@softcomputing.es

Abstract

Understand the behavior of Fuzzy Rule-based Systems (FRBSs) at inference level is a complex task that allows the designer to produce simpler and powerful systems. The fuzzy inference-grams –known as fingrams– establish a novel and mighty tool for understanding the structure and behavior of fuzzy systems. Fingrams represent FRBSs as social networks made of nodes representing fuzzy rules and edges representing the degree of interaction between pairs of rules at inference level (no edge means no significant interaction). We can analyze fingrams obtaining helpful information such as detecting potential conflicts between rules, unused rules and redundant ones. This paper introduces a new module for fingram generation and analysis included in the free software tool GUAJE. This tool aims to design, analyze and evaluate fuzzy systems with good interpretability-accuracy trade-off. In addition, GUAJE includes several intuitive and interactive tutorials to uncover the possibilities it offers. One of them generates and enhances a fuzzy system, analyzing each improvement through the use of fingrams, and lets the user reproduce the illustrative case study described in this paper.

Keywords: Interpretability-accuracy trade-off, fuzzy modeling, fingrams, GUAJE.

1. Introduction

Fuzzy sets and systems have become a mature research field with many theoretical and applied works starting from Zadeh's seminal work [36]. Among the huge number of research lines developed by the fuzzy community, system modeling with fuzzy rule-based systems (FRBSs) –called fuzzy modeling [23]– has been a fruitful research line for years.

During a long period –from 1965 to 1990– fuzzy modeling was mainly supported by expert knowledge. Researchers concentrated on building fuzzy models made up of a few simple linguistic variables [38] and linguistic rules [37] usually referred as Mamdani rules [27]. Accordingly, those designed fuzzy models were easy to interpret, and interpretability emerged naturally as an important advantage. However, researchers realized that considering only expert knowledge was not enough when dealing with complex real-world problems. Fortunately, it is also possible to build fuzzy models automatically –following a machine learning approach– from experimental data [24].

Thus, in a second period –from 1990 to 2000– researchers focused on automatically creating accurate fuzzy systems from experimental data, although disregarding interpretability. Of course, fuzzy systems are not interpretable *per se*, and automatically generated rules are rarely as readable as desired.

Interpretability has recovered a main role inside the fuzzy community since 2000 [5]. Researchers realized that accuracy and interpretability should be cared together, although, both issues are somehow contradictory. High accuracy usually implies low interpretability and *vice versa*. In practice, fuzzy modeling involves careful design where both interpretability and accuracy must be taken into account along the design process. The quest for the right interpretability-accuracy trade-off has become a great challenge in the last decade [16]. Nowadays, many researchers are actively working on it but a lot still remains to be done.

The notion of fuzzy inference-gram (*fingram*), recently introduced [4], is a powerful tool supporting the quest for interpretability-accuracy trade-off. In short, the behavior of a fuzzy system is analyzed – at inference level– by looking at those pairs of co-fired (simultaneously fired) rules by a given input vector. Then, a social network represents the rule base interaction where each individual entity represents a rule, and edges connecting entities show the relations among rules.

This paper explains how the free open source software GUAJE has been recently enhanced with a new module in charge of fingram generation and analysis [10]. In addition, GUAJE offers several optimization and simplification tools –at both fuzzy partition and fuzzy rule level– devoted to improve the accuracy and interpretability of the entire fuzzy system. The new module for fingrams eases the quest for interpretability-accuracy trade-off along the entire modeling process. Fingrams let us visually analyze and uncover the behavior and consequences of the applied optimization and simplification techniques. As a result, the designer can dynamically change the related parameters and/or improvement strategies with the aim of achieving the best balance between interpretability and accuracy.

The rest of the paper is organized as follows. Section 2 gives a global overview on existing software for fuzzy systems. Then, Section 3 introduces fingrams and their uses in fuzzy modeling. Section 4 presents the open source tool GUAJE, devoted to design and analyze FRBSs, which includes a new module for handling fingrams. Afterwards, we use a simple but highly illustrative use case in Section 5 to show the potentials of fingrams. Finally, some conclusions and future works are sketched in Section 6.

2. Fuzzy systems software overview

Along this long trip (more than forty five years), the fuzzy community has produced many publications regarding both theoretical and practical issues, and as a side effect, a lot of software tools have been developed too.

There exist some powerful and widely known commercial tools like the Fuzzy Toolbox[a] and the Adaptive Neuro-Fuzzy Inference System (ANFIS[b]) both provided by Matlab; the software fuzzyTECH[c]; or the fuzzy package provided with Wolfram Mathematica[d].

Anyway, this contribution focuses on open source tools that have recently reached a high level of development. They offer the richness of quickly incorporation of new developments made by the active research community, playing an important role in academy and industry. Moreover, most of this software is freely downloadable for research and education purposes, and facilitates the design of advanced prototypes for many novel applications.

We would like to highlight the following open source tools for fuzzy modeling –our principal interest in this contribution– because of their good performance and human-friendly interfaces:

- FisPro[e] (Fuzzy Inference System Professional). An open source tool for creating fuzzy inference systems [21]. It includes many algorithms for generating fuzzy partitions and rules directly from experimental data. FisPro aims at simulating physi-

[a]http://www.mathworks.es/products/fuzzylogic/index.html
[b]http://www.mathworks.es/help/toolbox/fuzzy/anfisedit.html
[c]http://www.fuzzytech.com
[d]http://www.wolfram.com/products/applications/fuzzylogic
[e]http://www.inra.fr/internet/Departements/MIA/M/fispro/

cal or biological systems, making emphasis in reasoning purposes. It eases the integration of expert knowledge and knowledge extracted from data. FisPro has been successfully applied to agriculture and environmental modeling problems [22].

- Xfuzzy[f]. A development environment aimed at producing fuzzy inference-based systems [11,12]. It integrates a set of tools covering all design stages from description to synthesis. Xfuzzy has been recently enhanced with an XML-based language called XFSML [28] that makes easier the interoperability among complementary tools.
- GUAJE[g]. A free software tool for generating understandable and accurate FRBSs in a java environment [6]. It allows combining expert knowledge and knowledge automatically extracted from data. GUAJE integrates several algorithms provided by different open source software tools. Moreover, the user can export models generated by GUAJE to other program formats, like FisPro, Xfuzzy, or the Matlab Fuzzy Toolbox.

As previously hinted, establish a standard language is an increasing important requirement. Regarding fuzzy control, there is a standard language called Fuzzy Control Language (FCL) published by the International Electrotechnical Commission (IEC 61131-7). Notice that, FCL is implemented in the open source library named jFuzzyLogic [17]. A variant of FCL based on XML which is called Fuzzy Markup Language (FML [1]) has been recently proposed and it is under standardization process.

Other important and more ambitious open source tools are KNIME[h] (Konstanz Information Miner) [14], a modular environment which is especially endowed with data manipulation and visualization methods but also with fuzzy rule learning capabilities [13]; FRIDA[i] (Free Intelligent Data Analysis Toolbox) [15] that provides methods for statistical analysis but also with visualization capabilities; and KEEL[j] (Knowledge Extraction based on Evolutionary Learning) [2] that probably contains the most complete collection of algorithms for genetic fuzzy systems. In addition, KEEL offers a user-friendly GUI for designing experiments and an educational data mining tool.

3. Fingrams

The term *fingram* stands for fuzzy inference-gram. It was coined in [4] by inspiration on the term scientogram firstly introduced by Vargas-Quesada and Moya-Anegón [29] in the search for a new tool aimed at visualizing the structure of science [35].

We have recently proposed a methodology for visual representation and exploratory analysis of the fuzzy inference process in FRBSs [30]. With that aim, fingrams represent FRBSs as social networks, giving very useful information about the FRBS behavior. They are made of nodes representing fuzzy rules and weighted edges that show graphically the interaction between rules at inference level.

Different aspects of teamwork between rules can be considered, producing different fingrams. As a first approach, we use co-firing between rules, i.e., rules fired at the same time by a given input vector. Therefore, rules highly related are more frequently fired together. Given a fuzzy system containing N rules and an experimental dataset covering most possible situations, we automatically generate an $N \times N$ weight matrix M describing the interactions between the N rules in terms of frequency of co-firing.

$$M = \begin{pmatrix} 0 & m_{12} & \dots & m_{1N} \\ m_{21} & 0 & \dots & m_{2N} \\ \dots & \dots & \dots & \dots \\ m_{N1} & m_{N2} & \dots & 0 \end{pmatrix} \quad (1)$$

The co-firing measure (m_{ij}) is defined by the next equation:

$$m_{ij} = \begin{cases} \frac{SFR_{ij}}{\sqrt{FR_i \cdot FR_j}} & , \quad i \neq j \\ 0 & , \quad i = j \end{cases} \quad (2)$$

[f] https://forja.rediris.es/projects/xfuzzy/

[g] http://www.softcomputing.es/guaje

[h] http://www.knime.org

[i] http://www.borgelt.net/frida.html

[j] http://sci2s.ugr.es/keel/

SFR_{ij} counts the number of samples firing simultaneously rules R_i and R_j. FR_i and FR_j count respectively the total number of samples firing rules R_i and R_j, disregarding if they are fired together or not.

Once matrix M is obtained, then it becomes straightforward the generation of an initial network (undirected graph) made up of N nodes connected through edges whose weights are directly taken from M. However, since rules usually cover the input space with dense overlapping among them, the resultant network is usually quite dense and difficult to understand. Accordingly, we apply a scaling mechanism to simplify the representation what allows the users to focus their attention in the most transcendent relations. The Pathfinder algorithm [18] is chosen due to its mathematical properties including the conservation of links and the representation of the most salient relationships present in the data. Pathfinder considers two main parameters:

- $r \in [1,\infty)$. It defines the *Minkowski r-metric* considered to measure the distance between two nodes not directly connected:

$$D = \left\{ \sum_i D_i^r \right\}^{\frac{1}{r}} \tag{3}$$

  In case r takes value 1, then D results in the sum of the link weights; $r = 2$ yields the usual Euclidean metric; and when $r \to \infty$ the path weight is the same as the maximum weight associated with any link along the path.
- $Q \in [2, N-1]$. It limits the number of links in the paths for which the triangle inequality is ensured in the scaled network. Hence, Pathfinder removes every path that connect two nodes that violate the triangle inequality, having an associated distance greater than any other path between the same two nodes composed of up to Q links.

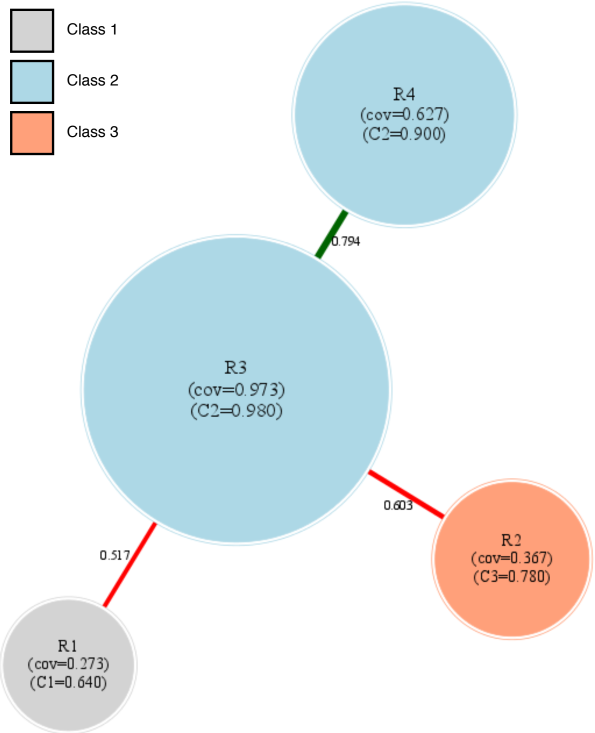
After scaling the network, the resultant network is visualized. Among the family of spring-embedder algorithms, we select the so-called *force-based algorithms* to automatically visualize the resultant networks. Kamada-Kawai [26], one of the most extended methods, assigns coordinates to the nodes trying to adjust, in an aesthetical pleasing way, the distances existing among them with respect to the actual interactions. The combination of rule co-firing, Pathfinder, and Kamada-Kawai places the most important nodes (i.e., those sharing more sources with the rest) toward the center. We call fingram to the final graphical representation of the network.

Fingrams have already been used in classification and regression problems [30]. They adopt different characteristics in each case, showing specific particularities of the problem represented. Fig. 3 shows an illustrative example related to the well-known IRIS classification problem[k]. The dataset contains 3 classes, each one referring to a type of iris plant. Note that rules pointing out the same output class are plotted with the same background color, and rules related to the same class are linked with green edges while red edges highlight potential inconsistency problems. In addition, nodes are labeled with informative values like coverage (*cov*) or goodness (*G*). In the figure, it is easy to appreciate how rule R3 covers most data samples and it goes in conflict against rules R1 and R2.

The analysis of a fingram can report helpful information about the analysis and verification of the related FRBS. We can systematically detect abnormal behaviors through carefully looking at the visual representation of a FRBS. Some simple but very useful examples of fingrams analysis are: identification of rules that cover a small amount of problem instances, perception of rules that exactly cover the same problem instances, detection of a rule that covers problem instances alone, assessment of FRBS comprehensibility, etc.

Regarding the comprehensibility analysis, we assume a large number of co-fired rules means a hardly comprehensible FRBS. Thus, the complexity of understanding the fuzzy inference process in terms of rules co-firing information can be evaluated by the Co-firing Based Comprehensibility Index (*COFCI*) [9]:

[k] IRIS dataset is freely available at the KEEL machine-learning repository [http://sci2s.ugr.es/keel/].



R1: IF Sepal Length is Average AND Sepal Width is High THEN Class is C1
R2: IF Sepal Length is High AND Sepal Width is Average THEN Class is C3
R3: IF Sepal Length is Average AND Sepal Width is Average THEN Class is C2
R4: IF Sepal Length is Average AND Sepal Width is THEN Class is C2

Figure 1: Intuitive example of fingram.

$$COFCI = \begin{cases} 1-\sqrt{\frac{CI}{MaxThr}}, & \text{if } CI \leqslant MaxThr \\ 0, & \text{otherwise} \end{cases} \tag{4}$$

$$CI = \sum_{i=1}^{N}\sum_{j=1}^{N}[(P_i + P_j)\cdot m_{ij}] \tag{5}$$

N is the number of rules, P_i and P_j count the number of antecedent conditions in rules R_i and R_j, respectively. m_{ij} is the measure of co-firing (Eq. 2) regarding rules R_i and R_j. $MaxThr$ is a normalization factor determined heuristically.

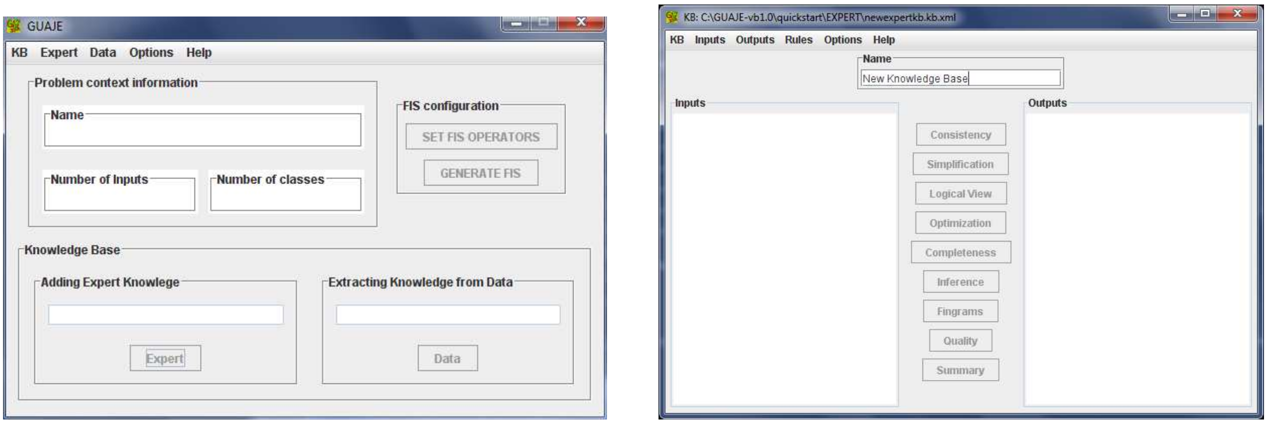
For getting a complete explanation about fingrams creation, interpretation and analysis, the interested reader is kindly referred to [30] where fingrams are deeply described.

4. GUAJE

GUAJE stands for Generating Understandable and Accurate fuzzy systems in a Java Environment [6]. Namely, GUAJE implements the fuzzy modeling methodology named as Highly Interpretable Linguistic Knowledge [7,8] that was conceived with the aim of yielding fuzzy systems endowed with good balance between interpretability and accuracy. GUAJE has been carefully developed in order to become user-friendly. In consequence, it makes the design of interpretable FRBSs easy and intuitive.

Fig. 2 shows the *Main* and *Expert Windows* of GUAJE. This free software actually consists of a modular architecture which is made up of several software modules in charge of the following tasks:

- **Data pre-processing**. It includes data visualization and analysis, re-sampling, etc.
- **Feature selection**. It focuses on identifying the



(a) Main window of GUAJE.

(b) Expert window of GUAJE.

Figure 2: Screenshots of GUAJE.

most significant input variables.

- **Partition design**. It deals with the characterization of each input variable as a linguistic variable with a justifiable number of meaningful linguistic terms. The attached membership functions can be defined by an expert and/or they can be automatically derived from data using machine learning techniques.
- **Rule base definition**. The system behavior can be described by a set of linguistic IF-THEN rules. An expert can define the rules and/or GUAJE can derive them from data.
- **Knowledge base verification**. This modeling stage verifies the consistency of the knowledge base previously defined. This analysis must be done at both linguistic and inference levels.
- **Knowledge base visualization**. The module for fingram generation and analysis shows graphically the system behavior in terms of rule co-firing at inference level.
- **Knowledge base improvement**. It tries on getting systems with better interpretability-accuracy trade-off. Two main options are available: linguistic simplification and partition tuning.
- **Knowledge base validation**. It checks if the designed fuzzy system matches with the expert requirements and expectations.
- **Knowledge base evaluation**. The quality assessment module reports tables including several indices for evaluating both interpretability and accuracy.

In the rest of this section we will provide a deeper analysis of those modules that are the most relevant for the remainder of this contribution.

4.1. Partition design

This module allows defining the relevant linguistic variables for the problem under consideration. GUAJE permits selecting the number of linguistic terms –from 2 to 9– for each input variable. They must be fully meaningful. Therefore, it is possible to choose among several pre-defined sets of vocabulary tuples (*low-high*, *small-large*, and so on) but also set new linguistic terms customized by the user.

The characterization of each linguistic term can be made by two different approaches. On the one hand, an expert can define them choosing prototype values by hand. On the other hand, GUAJE can derive partitions by using several induction methods. In both cases, GUAJE imposes the use of strong fuzzy partitions [33] with the objective of maximizing interpretability.

4.2. Rule base definition

This module provides mechanisms to define the rules which compound the FRBS. The user can create rules by two different, but not incompatible, approaches. Firstly, he/she can define the rules by expert knowledge. As an alternative, machine learning techniques can derive rules from data, either to constitute the rule base or to complement pre-existing expert rules. GUAJE permits the use of *Fuzzy Decision Trees*, *Wang and Mendel*, *Fast Prototyping*, and *Prototype Rule* algorithms [7].

In addition, the user can choose among typical methods for rule conjunction (*minimum*, *product*, or *Łukasiewicz*) and disjunction (*maximum* or *sum*).

4.3. Knowledge base visualization

This module shows graphically the interaction among rules at inference level [10].

GUAJE first generates a co-firing matrix regarding the pairs of rules simultaneously fired by each problem instance. Then, Pathfinder scales the graph related to such matrix. Afterwards, Kamada-Kawai algorithm determines the placement of nodes. Then, the graph is enhanced with information related to rules (coverage, goodness, etc.). Finally, the resultant fingram is displayed to the user who can analyze it and interact with it, making zoom in and zoom out, removing nodes, and so on.

4.4. Knowledge base improvement

This module aims to obtain a better balance between interpretability and accuracy.

GUAJE provides two ways to improve the interpretability-accuracy trade-off of a given FRBS. On the one hand, rule base simplification permits increasing readability of the FRBS by reducing its complexity (in terms of number of rules, premises, variables, linguistic terms, etc.) but without jeopardizing accuracy beyond a pre-defined threshold. On the other hand, fuzzy partitions can be tuned in order to increase accuracy while keeping comprehensibility (because of imposing several semantic constraints).

4.4.1. Rule Base Simplification

The goal is the generation of a more compact FRBS, regarding both fuzzy partitions and rules, thus improving interpretability while preserving accuracy. GUAJE offers three alternatives:

Genetic rule selection: The initial FRBS is used for building the first individual of the population. A binary-coded chromosome with size N (the number of initial rules) is generated. Depending on whether a rule is selected or not, values 1 or 0 are respectively assigned to the corresponding gene. At the beginning all rules are supposed to be selected. The rest of the population is randomly generated. In each generation, parents are selected by binary tournament. Then, uniform crossover and flip-flop random mutation are applied. The best individuals automatically pass to the next generation by elitism. The stopping criteria are the maximum number of generations, or fitness under the predefined threshold. Fitness function is defined as the weighted sum of the accuracy and interpretability indices selected by the user among all those ones provided by GUAJE.

Logical view reduction: First of all, the current rule base is transformed into several truth tables without any change of semantics. This is possible thanks to the propositional view of fuzzy rule-based systems handled by GUAJE. In a second step, the truth tables previously generated are, in turn, minimized by applying truth-preserving operators (but without taking care of accuracy which may go down dramatically depending on how the initial rules were defined). Then, the new set of truth tables is transformed into propositions in the third step, so that constituting a new rule base, different but derived from the original one.

Linguistic simplification: It includes rule-based reduction and partition simplification, at linguistic level. In short, it is an iterative process which first acts on the rules and then on the partitions at each iteration. This cycle is repeated until no more interpretability improvement is feasible without penalizing accuracy beyond a predefined threshold. Firstly, the procedure looks for redundant elements (linguistic terms, premises, rules, etc.) that can be removed without altering the system accuracy. Then, it tries to merge elements always used together. Finally, it removes elements apparently needed but not contributing too much to the final accuracy.

4.4.2. *Partition Optimization*

The goal is increasing accuracy without jeopardizing interpretability. The optimization task only affects the fuzzy partitions that define the system variables. The partition tuning is constrained to maintain strong fuzzy partitions. Two strategies are considered [3]:

Genetic tuning: An *all-in-one* optimization procedure based on a global search strategy. It is actually a genetic tuning process that looks for adjusting all system parameters at the same time. The procedure starts with a population of randomly generated solutions represented by real-coded chromosomes. Parents are selected by binary tournament at each generation. Then, $BLX-\alpha$ crossover and random mutation are applied. The best individuals are preserved by elitism. The stopping criteria are the maximum number of generations, or fitness under the predefined threshold. Fitness function is defined by the accuracy index selected by the user among all those ones provided by GUAJE.

Solis-Wets: An element by element optimization procedure based on the classical local search strategy proposed by Solis and Wets [34]: It is *a hill climbing method with memorization of the previous successes* [19]. The goal is not to find the global optimum, but to improve accuracy by performing a few iterations. Two alternatives are available: Variable by variable, and label by label.

4.5. *Knowledge base evaluation*

This module provides a complete overview about the quality (regarding both accuracy and interpretability) of the designed FRBSs.

For accuracy assessment there are universal indices commonly accepted, as the percentage of covered samples (*Coverage*) or the percentage of misclassified samples (*MC*), in classification problems:

$$MC = \frac{1}{d}\sum_{i=1}^{d} err_i;\ \ err_i = \begin{cases} 1, & \text{if } C_i \neq \widehat{C_i} \\ 0, & \text{otherwise} \end{cases} \tag{6}$$

d is the number of instances. C_i represents the class of instance i. $\widehat{C_i}$ is the inferred class.

On the contrary, there is not any well established and widely recognized interpretability index. Even more, there is a need to look for two kind of complementary indices, objective and subjective ones [5]. In this paper, two objective indices, the number of rules (*NR*) and the total rule length (*TRL*), and one subjective index, the *COFCI* index (Eq. 4), will be used to evaluate the interpretability of FRBSs. Notice that *TRL* counts the total number of linguistic propositions into the whole rule base.

5. Illustrative case study. Generation and analysis of fingrams with GUAJE.

GUAJE has been enhanced with a new software module for fingram generation and analysis. It is successfully integrated with the rest of modules of the software architecture as it was sketched in the previous section. This section details how to handle (generate, manipulate and analyze) fingrams in GUAJE through an example. For the sake of clarity the case study focuses only on a highly illustrative classification problem even though fingrams can also be applied to any classification or regression problem. We have selected a very well-known benchmark classification problem, WINE[1] just for illustrative purposes.

[1]WINE dataset is freely available at the KEEL machine-learning repository [http://sci2s.ugr.es/keel/].

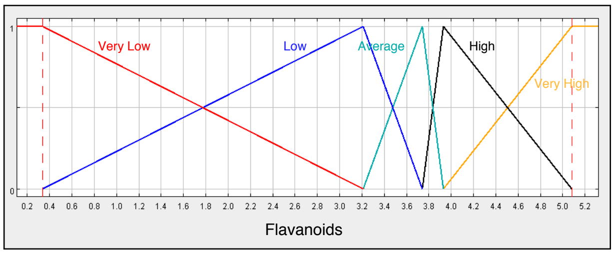


Figure 3: Linguistic terms and membership functions of variable *Flavanoids*.

WINE dataset contains 178 instances coming from the results of a chemical analysis of wines grown in the same region in Italy from three different cultivars. Thus, the output of the FRBS will be one categorical variable including the 3 classes of wines. In addition, the quantities of 13 constituents (*Alcohol*, *Malic acid*, *Ash*, *Alcalinity of ash*, *Magnesium*, *Total phenols*, *Flavanoids*, *Nonflavanoids phenols*, *Proanthocyanins*, *Color intensity*, *Hue*, *OD280/OD315 of diluted wines*, and *Proline*) are taken as inputs.

The dataset is split into two samples for training and test, trying to avoid overfitting. The training set comprises 75% (133 instances) of the available data picked at random. The test set is compound by the remaining 25% (45 samples).

Then, each input variable is characterized by a strong fuzzy partition which contains five membership functions (of triangular shape). They take values in the universe of discourse determined by the minimum and maximum available data. Fuzzy partitions are automatically derived from training data with the Hierarchical Fuzzy Partitioning (HFP) algorithm [20]. Fig. 3 shows, just for illustrative purpose, the membership functions attached to each linguistic term of variable *Flavanoids*. As it can be appreciated, partitions generated by HFP are always strong fuzzy partitions but not necessarily uniform. Notice that, interpretable fuzzy partitions must represent prototypes that are meaningful for the expert and context-dependant, but this fact does not imply they have to be uniform.

Moreover, we must set a meaningful linguistic term related to each membership function because we deal with linguistic variables. This way, we define the basic vocabulary to be used later in the definition of linguistic rules. We have considered the following linguistic terms: *Very Low* / *Low* / *Average* / *High* / *Very High*.

Before defining fuzzy rules, it is time to choose carefully the involved fuzzy operators, because they directly alter the inference mechanism. We have selected *minimum*, *maximum* and *max crisp* as t-norm, t-conorm and defuzzification method respectively, the usual inference scheme considered when dealing with classification problems.

Looking for a set of general rules that exhibit a good interpretability-accuracy trade-off, we have induced rules using Fuzzy Decision Trees (FDT) algorithm [25]. FDT can be seen as a fuzzy version of the popular decision tree induction algorithm defined by Quinlan [31]. The GUAJE implementation of FDT is actually based on the generation of a neuro-fuzzy decision tree which is easily translated into quite general incomplete rules where only a subset of input variables is considered. The result of running FDT (with maximum tree depth equal to 3; and using the fuzzy partitions previously generated) is a rule base made up of 32 rules. As expected, not all inputs are considered in all the rules. In fact, each rule uses only a subset of the input variables (3.4 inputs per rule in average). Picture on the left side in Fig. 4 shows the *Expert Window* of GUAJE after generating fuzzy partitions and rules. The top part of the picture shows the variables (13 inputs and 1 output) while a table represents the generated rule base at the bottom. Each row corresponds to one rule while each column represents one variable.

Table 1 summarizes the quality evaluation along the process. It reports the values computed for the quality indices previously introduced in Section 4.5. *NR* is the number of rules. *TRL* reports the to-

Table 1: FRBS quality along the design stages.

Quality Index	*Original FRBS*	*R5, R16, R31 R32 removal*	*FRBS after rule base simplification*	*FRBS after partition optimization*
NR	32	28	6	6
TRL	109	101	13	13
COFCI	0.065	0.075	0.787	0.787
MC Training	0.932	0.910	0.932	0.94
Coverage Training	100	98.496	100	100
MC Test	0.867	0.867	0.911	0.911
Coverage Test	97.778	97.778	100	100

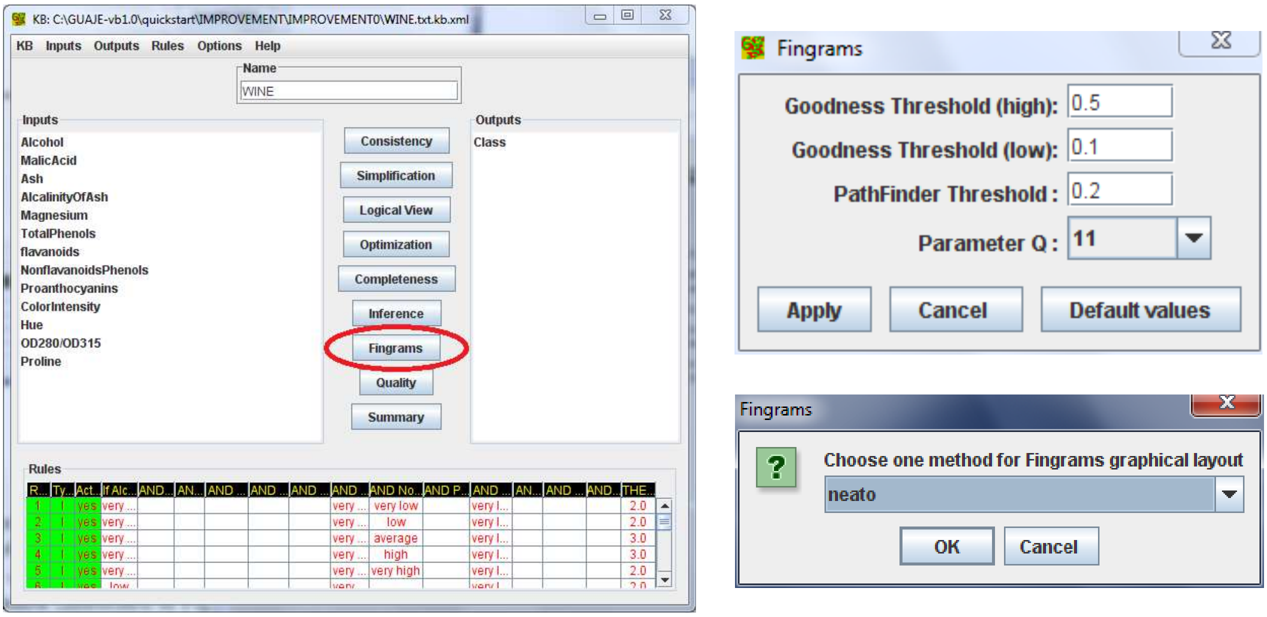
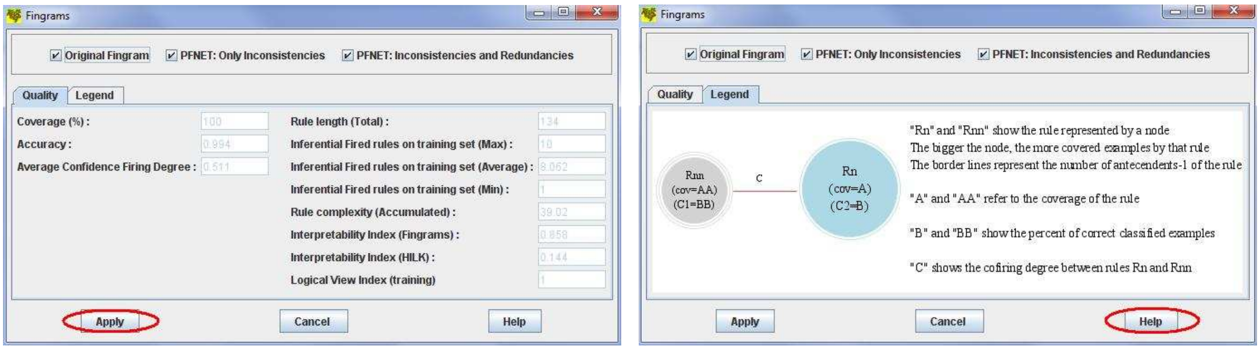


Figure 4: Generation of fingrams.

tal rule length. *COFCI* stands for co-firing based comprehensibility index. *MC* means misclassified cases. *MC* and *Coverage* are computed regarding both training and test datasets. The column entitled as *Original FRBS* corresponds to the current FRBS (the one displayed in Fig. 4). The rest of columns are related to the next design steps which will be discussed in the rest of this section.

Once the *Original FRBS* is generated, we can use fingrams with the aim of uncovering the FRBS behavior at inference level through a graphical analysis. First, we must set some parameters (see pictures on the right side in Fig. 4):

- **Goodness Threshold**. Upper and lower thresholds for estimating the goodness of coverage regarding each single rule. The goodness measure informs about how well each rule classifies the problem instances that it covers. A rule covers one problem instance when the rule firing degree for that instance is greater than a predefined threshold (0.1 in this work).
- **Pathfinder Threshold**. This parameter is used for pruning the initial graph (removing those edges with weights smaller or equal than the threshold),



(a) Quality view.

(b) Legend view.

Figure 5: GUAJE fingrams window.

before running Pathfinder.

- **Q**. This is the specific parameter of Pathfinder which limits the number of links in the paths respecting the triangle inequality. GUAJE suggests assigning to Q the maximum number of rules that can be simultaneously fired, which is estimated in an inferential way regarding the available dataset. In consequence, the network scaling will take shorter time. Anyway, by default $Q = N - 1$, with the aim of assuring that all paths are properly analyzed.

Second, we have to choose one layout algorithm (bottom right side in Fig. 4) among those provided by GUAJE (*neato*, *fdp*, *circo*, and so on). We have chosen *neato* which is an implementation of the Kamada-Kawai algorithm.

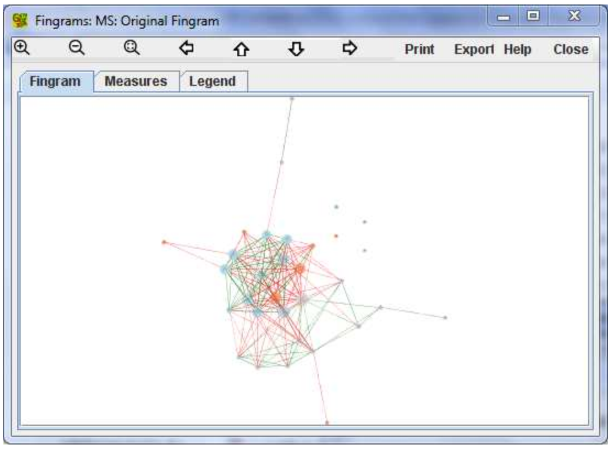
Then, the GUAJE window of Fig. 5(a) appears. The body of the window is structured in the form of a tabbed panel. The *Quality* tab gives an overview of the quality of the designed FRBS. It provides a list of quality indices[m] regarding both accuracy (on the left) and interpretability (on the right). Moreover, the user can interpret fingrams according to the information presented in the *Legend* tab (Fig. 5(b)).

Once selected the proper options, the pictures in Fig. 6 are displayed. Thanks to the use of SVG format the user can interact with the graph through zooming, moving, and/or exploring in depth some zones of interest in the entire network. In addition, when the user passes the mouse over a node or an edge, a text pops up with the linguistic description of the related rule or link. Moreover, the user can disable rules by clicking on its corresponding node, i.e., a rule is temporally deactivated in the rule base, and the fingram is generated again without taking care of that rule. In consequence, fuzzy systems design becomes an interactive process which is effectively guided by decisions drawn from the expert analysis of fingrams.

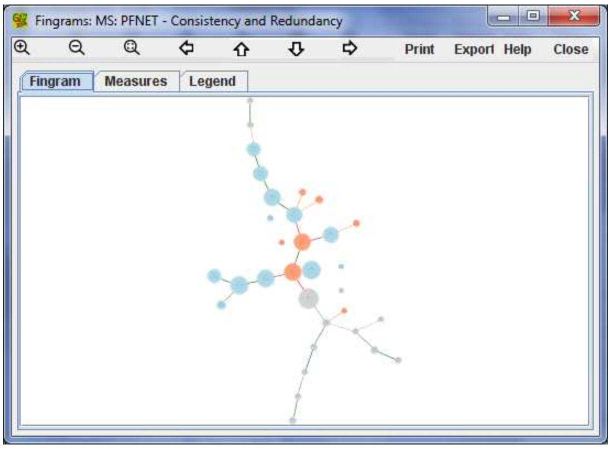
The GUAJE window for fingram analysis is illustrated in Fig. 6. In addition to the visualization panel, it contains other two tabs: the *Legend* tab with a specific legend of the fingram presented; and the *Measures* tab (Fig. 6(c)) which gives several rule rankings based on some of the most popular measures in the context of social network analysis, such as *Page Rank* or *Centrality*.

The complete fingram is usually quite dense and difficult to analyze as it can be appreciated in Fig. 6(a). Thus, the network scaling stage becomes essential to provide a good and efficient fingram analysis. Therefore, GUAJE uses Pathfinder[n] to keep only the most significant links, yielding as result the scaled fingram depicted in Fig. 6(b). Looking carefully at the scaled fingram, it is easy to detect rules that cover regions with a few examples and are

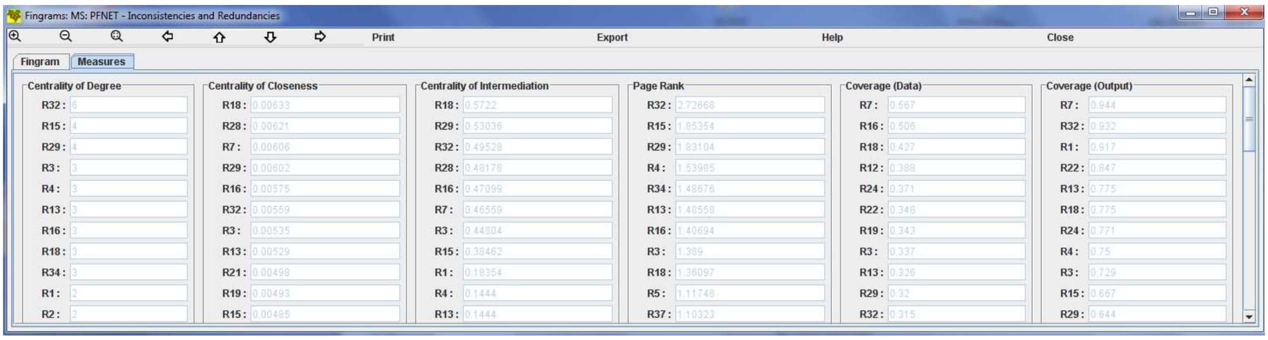
[m] Such indices are thoroughly explained in the GUAJE user manual.
[n] GUAJE actually makes use of MST-PathFinder [32]. It is a recently published variant of Pathfinder algorithm able to generate large science maps in cubic time.



(a) Complete fingram.



(b) Scaled fingram with Pathfinder.



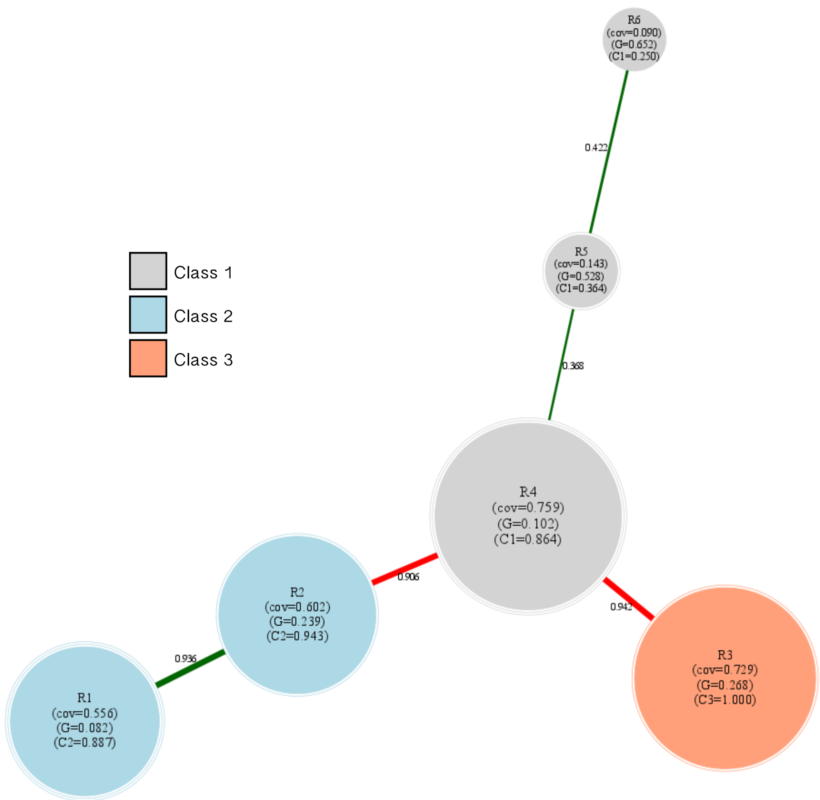
(c) Rule rankings.

Figure 6: Fingram analysis window.

therefore good candidates to be studied in detail. For instance, rules R5, R16, R31 and R32 appear isolated and they cover very few examples. Hence, they are good candidates to be removed. In fact, as shown in Table 1, after removing such rules accuracy only slightly decreases.

In the quest for even better interpretability-accuracy trade-off we have opted for applying first linguistic simplification (aimed at improving interpretability while preserving accuracy) and then partition tuning (with the goal of increasing accuracy without jeopardizing interpretability).

The linguistic simplification process yields a more compact FRBS. The number of variables is reduced from 13 to 4, the total number of membership functions goes down from 65 to 11, the number of rules drops from 28 to 6, and the total rule length pass from 101 to 13. This is because simplified rules are much more general than the original ones. The result of simplification can be seen in a new fingram which is depicted in Fig. 7. At first sight it is clear the high level of simplification obtained with a drastically reduction of the number of rules (as a side effect the *COFCI* index improves). Even more, the rules cover all the examples (regarding both training and test sets) because they are more general, and they also produce better accuracy (as it is detailed in Table 1). The rule R3 covers all examples related to class C3. In addition, it significantly overlaps with rule R4, thus yielding potential inconsistencies. Rule R4 turns up as the central rule in the fingram. It covers most data samples going in conflict against R3 but also against R2. Even though rules R2 and R4 cover many examples they are not



R1: IF Alcohol is Very Low AND Flavanoids is Very Low AND Color Intensity is Very Low THEN Class is C2

R2: IF Alcohol is Very Low AND Flavanoids is Low THEN Class is C2

R3: IF Flavanoids is Very Low AND Color Intensity is Low THEN Class is C3

R4: IF Alcohol is Low AND Flavanoids is Low AND Proline is (Low OR Average) THEN Class is C1

R5: IF Alcohol is (Average OR High) AND Flavanoids is Low THEN Class is C1

R6: IF Flavanoids is Average THEN Class is C1

Figure 7: Fingram of the simplified FRBS.

enough to handle properly their related classes, C2 and C1 respectively. Therefore, they have to be complemented with other rules, partly redundant with them, like R1, R5 and R6.

With respect to the optimization stage, we have selected the Solis-Wets algorithm able to tune the membership functions yielding marginal changes in the overall behavior of the system (look at quality indices reported in Table 1). For this particular case study, the tuning process only affects to the definition of the linguistic terms *Very Low* and *Low* of variable *Alcohol*. Therefore, only rules R1, R2 and R4 suffer slightly changes. In consequence, the resultant fingram remains almost the same than after simplification.

Finally, it is worthy to remark that GUAJE comes with several intuitive and interactive tutorials. One of them shows the benefits and potentials of fingrams for aiding the design of FRBSs. It details, step by step, first how to build an interpretable fuzzy rule-based classifier and then how to simplify and optimize it, looking for the best balance between accuracy and interpretability supported by fingrams. The illustrative case study presented above can be reproduced by the interested reader with the help of GUAJE and just following the related tutorial.

6. Conclusions and future works

This paper has explained how the software module for fingram generation and analysis is successfully integrated with the rest of modules provided by GUAJE. The new module is a powerful tool for understanding the system behavior at inference level. It becomes really useful in the design of fuzzy systems because the designer can check graphically, at any design stage (rule learning, simplification, optimization, etc.), the interaction between rules but also how each design decision affects to the interpretability-accuracy trade-off.

Please, notice that GUAJE is freely available (under GPL license) as open source software at:

http://www.softcomputing.es/guaje

The number of users of GUAJE is growing up all around the world. The users' feedback helps us to continue enhancing this free software tool. Thus, new releases will be available with improvements in the visualization of fingrams. In short-term we want to make even more dynamic and user-friendly the interaction with the user. Thus, he/she should be able to alter the graph layout through making drag and drop of some nodes or collapse/expand parts of interest in the entire graph. Regarding mid-term future we plan to develop a new software module that gives fully automatic support to the design of fuzzy systems guided by fingrams.

The co-firing measure presented in this contribution is biased by the training data. Such fact may be avoided by considering other measures of overlapping between rules, that will be part of our future work.

Acknowledgments

This work has been funded by the "Spanish Ministry of Economy and Competitiveness" under projects TIN2011-29824-C02-01 and TIN2011-29824-C02-02. It has also been partly supported by the European Centre for Soft Computing.

References

1. G. Acampora, V. Loia, and A. V. Vasilakos, "Autonomous composition of fuzzy granules in ambient intelligence scenarios," *Human-Centric Information Processing Through Granular Modelling*, Studies in Computational Intelligence, Springer, 265–287 (2009).
2. 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, J. C. Fernandez, and F. Herrera, "KEEL: a software tool to assess evolutionary algorithms for data mining problems," *Soft Computing*, **13(3)**, 307–318 (2009).
3. J. M. Alonso, O. Cordón, S. Guillaume, and L. Magdalena, "Highly interpretable linguistic knowledge bases optimization: Genetic Tuning versus Solis-Wetts. Looking for a good interpretability-accuracy trade-off," *Proc. IEEE International Conference on Fuzzy Systems*, 901–906 (2007).
4. J. M. Alonso, O. Cordón, A. Quirin, and L. Magdalena, "Analyzing interpretability of fuzzy rule-based systems by means of fuzzy inference-grams," *Proc. World Conference on Soft Computing*, San Francisco State University, 181.1–181.8 (2011).
5. J. M. Alonso and L. Magdalena, "Editorial: special issue on interpretable fuzzy systems," *Information Sciences*, **181(20)**, 4331–4339 (2011).
6. J. M. Alonso and L. Magdalena, "Generating understandable and accurate fuzzy rule-based systems in a Java environment," *Lecture Notes in Artificial Intelligence - Proc. 9th International Workshop on Fuzzy Logic and Applications*, **LNAI6857**, 212–219 (2011).
7. J. M. Alonso and L. Magdalena, "HILK++: an interpretability-guided fuzzy modeling methodology for learning readable and comprehensible fuzzy rule-based classifiers," *Soft Computing*, **15(10)**, 1959–1980 (2011).
8. J. M. Alonso, L. Magdalena, and S. Guillaume, "HILK: a new methodology for designing highly interpretable linguistic knowledge bases using the fuzzy logic formalism," *International Journal of Intelligent Systems*, **23(7)**, 761–794 (2008).
9. J. M. Alonso, D. P. Pancho, O. Cordón, A. Quirin, and L. Magdalena, "Social network analysis of co-fired fuzzy rules". In R. R. Yager, A. M. Abbasov, M. Z. Reformat, and S. N. Shahbazova, editors, *Soft Computing: State of the Art Theory and Novel Applications*, *Studies in Fuzziness and Soft Computing*, Springer Berlin Heidelberg, **291**, 113–128 (2013).
10. J. M. Alonso, D. P. Pancho, and L. Magdalena, "Enhancing the fuzzy modeling tool GUAJE with a new module for fingrams-based analysis of fuzzy rule bases," *Proc. IEEE International Conference on Fuzzy Systems*, 1082–1089 (2012).
11. I. Baturone, F. J. Moreno-Velo, S. Sánchez-Solano, A. Barriga, P. Brox, A. A. Gersnoviez, and M. Brox, "Using Xfuzzy environment for the whole design of fuzzy systems," *Proc. IEEE International Conference on Fuzzy Systems*, 517–522 (2007).
12. I. Baturone, F. J. Moreno-Velo, S. Sánchez-Solano, and A. Ollero, "Automatic design of fuzzy controllers for car-like autonomous robots," *IEEE Transactions on Fuzzy Systems*, **12(4)**, 447–465 (2004).
13. M. R. Berthold, "Mixed fuzzy rule formation," *International Journal of Approximate Reasoning*, **32**, 67–84 (2003).
14. M. R. Berthold, N. Cebron, F. Dill, T. R. Gabriel, T. Kötter, T. Meinl, P. Ohl, K. Thiel, and B. Wiswedel, "KNIME - the Konstanz information miner (version 2.0 and beyond)," *ACM SIGKDD Explorations*

Newsletter, **11**, 26–31 (2009).
15. C. Borgelt and G. González-Rodríguez, "FrIDA - a free intelligent data analysis toolbox," *Proc. IEEE International Conference on Fuzzy Systems*, 1892–1896 (2007).
16. J. Casillas, O. Cordón, F. Herrera, and L. Magdalena, "Interpretability improvements to find the balance interpretability-accuracy in fuzzy modeling: an overview," In J. Casillas, O. Cordón, F. Herrera, and L. Magdalena, editors, *Interpretability issues in fuzzy modeling*, Studies in Fuzziness and Soft Computing, Springer-Verlag, Heidelberg, **128**, 3–22 (2003).
17. P. Cingolani and J. Alcala-Fdez, "jFuzzyLogic: a robust and flexible Fuzzy-Logic inference system language implementation," *Proc. IEEE International Conference on Fuzzy Systems*, 1090–1097 (2012).
18. D. W. Dearholt and R. W. Schvaneveldt, "Properties of Pathfinder networks," In R. Schvaneveldt, editor, *Pathfinder associative networks: studies in knowledge organization*, Ablex Publishing Corporation, 1–30 (1990).
19. P. Y. Glorennec, "*Algorithmes d'apprentissage pour systemes d'inference floue (French Edition)*," Editions Hermès, Paris (1999).
20. S. Guillaume and B. Charnomordic, "Generating an interpretable family of fuzzy partitions," *IEEE Transactions on Fuzzy Systems*, **12(3)**, 324–335 (2004).
21. S. Guillaume and B. Charnomordic, "Learning interpretable fuzzy inference systems with FisPro," *Information Sciences*, **181(20)**, 4409–4427 (2011).
22. S. Guillaume and B. Charnomordic, "Fuzzy inference systems: an integrated modeling environment for collaboration between expert knowledge and data using FisPro," *Expert Systems with Applications*, **39(10)**, 8744–8755 (2012).
23. H. Hellendoorn and D. Driankov, "*Fuzzy model identification*," Springer-Verlag London, UK (1997).
24. E. Hüllermeier, "Fuzzy methods in machine learning and data mining: status and prospects," *Fuzzy Sets and Systems*, **156**, 387–406 (2005).
25. H. Ichihashi, T. Shirai, K. Nagasaka, and T. Miyoshi, "Neuro-fuzzy ID3: a method of inducing fuzzy decision trees with linear programming for maximizing entropy and an algebraic method for incremental learning," *Fuzzy Sets and Systems*, **81**, 157–167 (1996).
26. T. Kamada and S. Kawai, "An algorithm for drawing general undirected graphs," *Information Processing Letters*, **31**, 7–15 (1989).
27. E. H. Mamdani, "Application of Fuzzy Logic to approximate reasoning using linguistic systems," *IEEE Transactions on Computers*, **26(12)**, 1182–1191 (1977).
28. F. J. Moreno-Velo, A. Barriga, S. Sánchez-Solano, and I. Baturone, "XFSML: an XML-based modeling language for fuzzy systems," *Proc. IEEE International Conference on Fuzzy Systems*, 1146–1153 (2012).
29. F. Moya-Anegón, B. Vargas-Quesada, V. Herrero-Solana, Z. Chinchilla-Rodríguez, E. Corera-Álvarez, and F. Muñoz-Fernández, "A new technique for building maps of large scientific domains based on the cocitation of classes and categories," *Scientometrics*, **61**, 129–145 (2004).
30. D. P. Pancho, J. M. Alonso, O. Cordón, A. Quirin, and L. Magdalena, "Fingrams: Visual representations of fuzzy rule-based inference for expert analysis of comprehensibility," *IEEE Transactions on Fuzzy Systems*, **21(6)**, doi:10.1109/TFUZZ.2013.2245130 (2013).
31. J. R. Quinlan, "Induction of decision trees," *Machine Learning*, **1**, 81–106 (1986).
32. A. Quirin, O. Cordón, J. Santamaría, B. Vargas-Quesada, and F. Moya-Anegón, "A new variant of the Pathfinder algorithm to generate large visual science maps in cubic time," *Information Processing and Management*, **44**, 1611–1623 (2008).
33. E. H. Ruspini, "A new approach to clustering," *Information and Control*, **15(1)**, 22–32 (1969).
34. F. J. Solis and R. J. B. Wets, "Minimization by random search techniques," *Mathematics of Operations Research*, **6(1)**, 19–30 (1981).
35. B. Vargas-Quesada and F. Moya-Anegón, "*Visualizing the structure of science*," Springer-Verlag (2007).
36. L. A. Zadeh, "Fuzzy sets," *Information and Control*, **8**, 338–353 (1965).
37. L. A. Zadeh, "Outline of a new approach to the analysis of complex systems and decision processes," *IEEE Transactions on Systems, Man and Cybernetics*, **3**, 28–44 (1973).
38. L. A. Zadeh, "The concept of a linguistic variable and its application to approximate reasoning," *Parts I, II, and III. Information Sciences*, **8, 8, 9**, 199–249, 301–357, 43–80 (1975).

A.1.4 JCR of International Journal of Computational Intelligence Systems

Journal Citation Reports®

2013 JCR Science Edition

Journal: International Journal of Computational Intelligence Systems

Journal Impact Factor

Cites in 2013 to items published in:		Number of items published in:	
	2012 = 20		2012 = 89
	2011 = 81		2011 = 135
	Sum: 101		Sum: 224

Calculation: Cites to recent items / Number of recent items = 101 / 224 = **0.451**

Journal Ranking

For **2013**, the journal **International Journal of Computational Intelligenc...** has an Impact Factor of **0.451**.

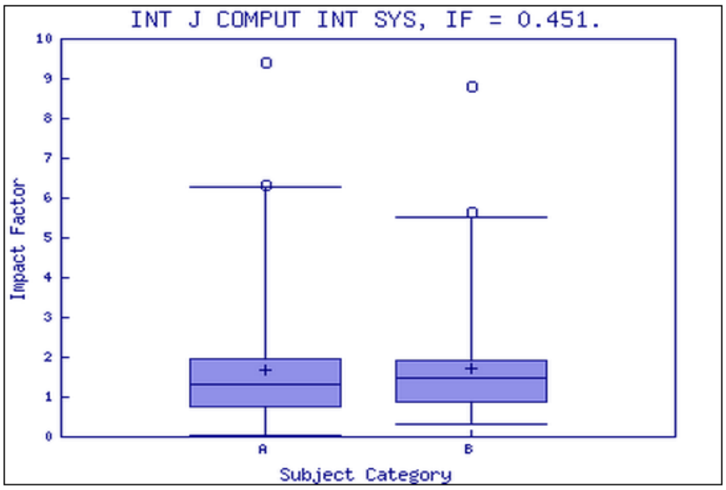
This table shows the ranking of this journal in its subject categories based on Impact Factor.

Category Name	Total Journals in Category	Journal Rank in Category	Quartile in Category
COMPUTER SCIENCE, ARTIFICIAL INTELLIGENCE	121	108	Q4
COMPUTER SCIENCE, INTERDISCIPLINARY APPLICATIONS	102	96	Q4

Category Box Plot

For **2013**, the journal **International Journal of Computational Intelligenc...** has an Impact Factor of **0.451**.

This is a box plot of the subject category or categories to which the journal has been assigned. It provides information about the distribution of journals based on Impact Factor values. It shows median, 25th and 75th percentiles, and the extreme values of the distribution.



Key

A - COMPUTER SCIENCE, ARTIFICIAL INTELLIGENCE

B - COMPUTER SCIENCE, INTERDISCIPLINARY APPLICATIONS

A.1.5 Third publication: Case of use

[68] Pancho, D. P., Alonso, J. M., and Magdalena, L. Enhancing Fingrams to deal with precise fuzzy systems. *Fuzzy Sets and Systems* (2015), In press, doi: 10.1016/j.fss.2015.05.019

This final publication extends Fingrams to better represent and analyze precise fuzzy systems. A specific metric and new representations handle the particularities of such systems. A new visual artifact allows to discover the set of data instances not covered by a given fuzzy system. A novel visual representation allows to study in detail the elements that are involved in the inference of a single data instance. The potentials of the enhanced methodology are sketched by taking the Fuzzy Unordered Rule Induction Algorithm (FURIA) as an illustrative example of precise fuzzy system.

Publication index:

Citations Google Scholar:	-
Citations Web of Science:	-

Enhancing Fingrams to Deal with Precise Fuzzy Systems

David P. Pancho, Jose M. Alonso, Luis Magdalena

European Centre for Soft Computing,
Edificio de Investigación, C/ Gonzalo Gutiérrez Quirós s/n,
33600 Mieres, Asturias, Spain,
e-mail: {david.perez,jose.alonso,luis.magdalena}@softcomputing.es
phone: +34 985 456 545

Abstract

Interpretability is a highly valued capability of fuzzy systems that turns essential when dealing with human interaction. Precise fuzzy modeling prioritizes performance at the cost of harming interpretability. Fuzzy Inference-grams (Fingrams) permit the graphical representation of fuzzy systems facilitating their comprehension, analysis and interpretation at inference level. We enhance Fingrams to better represent and analyze precise fuzzy systems. A specific metric and new representations handle the particularities of such systems. A new visual artifact allows to discover the set of data instances not covered by a given fuzzy system. A novel visual representation allows to study in detail the elements that are involved in the inference of a single data instance. The potentials of the enhanced methodology are sketched by taking the Fuzzy Unordered Rule Induction Algorithm (FURIA) as an illustrative example of precise fuzzy system. For instance, a highly valuable representation is obtained for the stretching mechanism of FURIA, thus facilitating its comprehensibility.

Keywords:
Fuzzy System Models, Fuzzy Inference Systems, Interpretability, Comprehensibility Analysis, Expert Analysis, Information Visualization, Social Network Analysis, Precise Fuzzy Modeling, FURIA.

FRBS stands for Fuzzy Rule-Based System, LFM for Linguistic Fuzzy Modeling, PFM for Precise Fuzzy Modeling, FURIA for Fuzzy Unordered Rule Induction Algorithm, and Fingrams for Fuzzy Inference-Grams.

1. Introduction

Interpretability is a highly appreciated capability of fuzzy systems that permits the correct understanding of systems behavior [1]. Interpretability of fuzzy systems represents their ability to formalize the behavior of a real system in a human understandable way [2, 3]. It takes advantage of the use of linguistic variables [4] and rules [5, 6] with high semantic expressivity close to natural language. According to some authors interpretability is of subjective nature and depends on the talent and background of the end-user [7].

Several factors influence in the assessment of interpretability [8] from low to high levels [9]. Mainly, they can be grouped in two main issues [7]: (1) readability (transparency) of the system description, and (2) comprehensibility of the system explanation. Notice that it only makes sense to take care about interpretable constraints [10] when there is a need of human interaction in any of the steps of the process (design, inference, improvement,...).

Fuzzy systems are not interpretable per-se. Although the use of linguistic variables and rules favors interpretability, this does not guarantee it. A careful design is demanded to simplify their understanding and ensure their interpretability [11, 12].

Fuzzy modeling [13] –system modeling with fuzzy rule-based systems (FRBSs)– can be done through two alternative approaches attending to the interpretability-accuracy trade-off: producing linguistic or precise fuzzy modeling [14]. Linguistic fuzzy modeling (LFM) prioritizes interpretability. It yields fuzzy rules composed of linguistic variables [4] taking terms with a real-world meaning [15, 16]. On the contrary, precise fuzzy modeling (PFM), which has accuracy as its main objective, constructs FRBSs that lack of semantic expressivity.

An effort has been done to obtain intermediate approaches that keep a good interpretability-accuracy trade-off. On the one hand, some works propose to improve accuracy of LFM [17]. On the other hand, others introduce techniques to enhance interpretability of PFM [18].

One of the key issues in the generation of a fuzzy system with good interpretability is the comprehension of the fuzzy inference process [19]. Understanding such process becomes an arduous task when dealing with PFM even for fuzzy modeling experts. We can highlight two of the most famous FRBSs for PFM: Mamdani FRBSs [6] and Takagi-Sugeno FRBSs [20]. The former, Mamdani FRBSs are typically multi-input-single-output systems with min-max inference mechanism, widely recognized as the easiest inference mechanism to understand. The latter, Takagi-Sugeno FRBSs produce as rule outputs non-linear combinations of the inputs involved in each rule, hindering its comprehension. Some authors have made an effort in simplifying and enhancing Takagi-Sugeno FRBSs [21, 22]. Anyway, in both cases the use of weighted rules, advanced defuzzification strategies, a high number of rules, variables or antecedents per rule, can make harder the understanding of the system behavior at inference level [1, 2, 23, 24, 25]. Moreover, when an instance fires several rules, as usual in PFM, the inferred output is hard to interpret.

There are not many publications tackling with visual analysis of the fuzzy system inference process. Probably, this is due to the well-known linguistic expressivity of LFM what gives prominence to linguistic representations. However, as mentioned before, not all fuzzy systems preserve such characteristic. Thus, visual tools would stand out in the case of PFM.

Pham et al. [26] overview the requirements to graphically represent fuzzy systems and critically review the existing methods for visualizing fuzzy data and relationships. Different alternatives support the visualization of fuzzy data, fuzzy partitions and fuzzy rules depending on the requirements the end-user may demand. A few authors [27, 28, 29, 30, 31, 32] present 2D graphical representations for FRBSs. Only [31, 32] represent rule interaction at inference level in terms of rule overlapping. Namely they use parallel coordinates to visualize high-dimensional fuzzy points and rules. This brief review shows that there is a lack of methods depicting the interaction among rules that, however, could strongly help in the comprehension of the rule base behavior at inference level.

Fuzzy Inference-grams [33], or Fingrams in short, have arisen as a powerful tool for visualizing and analyzing FRBSs. Fingrams give a global view of fuzzy systems, and allow us to understand its behavior at a high level of abstraction. They present fuzzy systems as social networks where rules are individuals that relate each other reflecting how they cover the input space. Different metrics and visual artifacts have been proposed to reflect the particularities of the different kinds of fuzzy systems [33, 34]. It is worthy to note the capability of Fingrams to graphically depict the inference mechanism of fuzzy systems. Fingrams let us visualize how instances are covered by rules, which are the rule outputs, and so on.

This paper proposes the use of Fingrams to understand the behavior of precise fuzzy systems –fuzzy systems constructed by PFM– and its particularities. To do so, we have extended Fingrams to effectively represent the characteristics of precise fuzzy systems.

Here we propose a visual representation that allows us to uncover how the inference mechanism operates for a single instance. It shows which rules cover the instance, giving us a complementary local view of the system.

Additionally, we propose a novel graphical representation for instances that are not covered by any rule in a given FRBS. This way we can quantify the uncovered instances and act consequently. It should be noticed that uncovered instances penalize the precision of FRBSs and their early detection and correction is essential for the correct behavior of the system.

In this paper, we use the Fuzzy Unordered Rule Induction Algorithm [35], FURIA in short, in order to illustrate the main potentials of applying Fingrams to understand the behavior of fuzzy systems generated by PFM.

FURIA is one of the most outstanding fuzzy rule-based classification methods attending to accuracy. However, although FURIA produces compact rule bases, its interpretability is arguable [36], being penalized by the absence of linguistic readability. Though FURIA usually generates low number of rules (and antecedents per rule), they lack of linguistic readability because there is no global semantics. Rule antecedents are rule dependent and do not have linguistic terms associated. In addition FURIA's inference mechanism occlude interpretability. It is based on a winner class mechanism with weighted rules in combination with the so-called rule stretching method which is in charge of handling uncovered instances. In consequence, FURIA includes a close-to-black-box inference mechanism, very hard to predict and understand.

The rest of the manuscript is organized as follows. Section 2 presents some preliminaries including a brief summary of PFM and Fingrams. Section 3 introduces the extension of Fingrams to assist the analysis of PFM. Section 4 focuses on the use of Fingrams for the visual analysis of precise fuzzy systems generated by FURIA. It contains a short overview of FURIA, the adaptations of Fingrams made to deal with FURIA, the analysis of some experimental results and a complete case of use where all the potentials of Fingrams are sketched. Finally, some conclusions and future work are pointed out in Section 5.

2. Preliminaries

2.1. Precise Fuzzy Modeling

PFM has as main objective to obtain precise fuzzy systems. There are several alternatives of PFM [37, 38, 35], but all of them induce rules from data instances and take advantage of sophisticated inference mechanisms to achieve high accuracy.

Namely, precise FRBSs can be constructed to deal with several different kinds of problems: classification, clustering, control, etc. In this paper, we deal only with multi-input-single-output Mamdani type classification FRBSs, establishing the basis to be generalized to other kind of fuzzy systems.

Given a classification problem with p classes:

$$C = \{c_l \mid l = 1, 2, ..., p\} \tag{1}$$

and a dataset (D) which contains (m) instances each with (n) attributes and an output class. Thus, an instance (I_k) is represented as a n-dimensional attribute vector ($\mathbf{x_k}$) plus its output class (y_k):

$$I_k = \{(\mathbf{x_k}, y_k) \mid \mathbf{x_k} = \{x_k^1, ..., x_k^n\},\ x_k^h \in \mathbb{R},\ h = 1, 2, ..., n\ y_k \in C\} \tag{2}$$

We denote D^{c_l} the set of instances with output class c_l.

$$D^{c_l} = \{I_k \mid y_k = c_l, k = 1, 2, ..., m\} \tag{3}$$

A precise FRBS is made up of a set of r rules trying to fit D. The usual rule format is as follows:

$$R_i\text{: IF } X_1 \text{ is } A_1^i \text{ \& ... \& } X_n \text{ is } A_n^i \text{ THEN } Y \text{ is } B^i \text{ with } w^i \tag{4}$$

where A_h^i is the rule antecedent for variable X_h ($h \in [1, n]$), B^i denotes the output class, and w^i is the weight associated to rule R_i.

We define R^{c_l} as the set of rules that have as output class c_l:

$$R^{c_l} = \{R_i \mid B^i = c_l,\ i = 1, 2, ..., r\} \tag{5}$$

and $\mu_{R_i}(I_k)$ as the firing degree up to a single instance (I_k) fires the rule R_i:

$$\mu_{R_i}(I_k) = \mu_{A_1^i}(x_k^1) \otimes ... \otimes \mu_{A_n^i}(x_k^n) \tag{6}$$

where $\otimes$ is a t-norm.

We distinguish between covered and uncovered instances. We define the set of covered instances (cv) as those firing at least one of the set of rules, and the set of uncovered instances (ucv) as those ones that do not fire any of the set of rules.

$$cv = \{I_k \mid \sum_{i=1,\dots,r} (\mu_{R_i}(I_k)) > 0,\ k = 1, 2, ..., m\} \tag{7a}$$

$$ucv = \{I_k \mid \sum_{i=1,\dots,r} (\mu_{R_i}(I_k)) = 0,\ k = 1, 2, ..., m\} \tag{7b}$$

Then, we define the firing degree of a rule (FD_{R_i}) as the accumulated firing degree for all the instances of D that fire that rule.

$$FD_{R_i} = \sum_{k=1,\dots,m} (\mu_{R_i}(I_k)) \tag{8}$$

We define D_i as the whole set of instances covered by rule R_i:

$$D_i = \{I_k \mid \mu_{R_i}(I_k) > 0, k = 1, 2, ..., m\} = D_i^+ \cup D_i^- \tag{9}$$

where D_i^+ and D_i^- denote respectively the set of positive and negative instances for rule R_i. We call positive instances to those covered by the rule in a consistent manner, i.e. the output class is the same for the rule and the data instance, being the remaining covered instances negative instances.

$$D_i^+ = \{I_k \mid \mu_{R_i}(I_k) > 0 \ \&\ y_k = B^i,\ k = 1, 2, ..., m\} \tag{10a}$$

$$D_i^- = \{I_k \mid \mu_{R_i}(I_k) > 0 \ \&\ y_k \neq B^i,\ k = 1, 2, ..., m\} \tag{10b}$$

The coverage (cov_i) of a rule R_i is defined as:

$$cov_i = \frac{|D_i|}{|D|} \tag{11}$$

with $|.|$ the cardinality of sets.

We define class coverage of a rule (cc_i) as the proportion of covered instances consistent with the rule output class with respect to the total instances of the dataset consistent with the rule output class.

$$cc_i = \frac{|D_i^+|}{|D^{B^i}|} \tag{12}$$

Finally, let's present the inference mechanism for this kind of systems. The level of class activation (S_{c_l}) is computed as:

$$S_{c_l}(I_k) = \max_{R_i \in R^{c_l}} \left(\mu_{R_i}(I_k) \cdot w^i \right) \tag{13}$$

and the predicted output class (y'_k) for the given instance as:

$$output(I_k) = y'_k \Rightarrow S_{y'_k}(I_k) = \max_{l=1\ldots p} S_{c_l}(I_k) \tag{14}$$

2.2. Fuzzy Inference-grams

Fuzzy Inference-grams, or Fingrams in short, have been recently introduced as a methodology for visual representation and exploratory analysis of FRBSs [33]. Fingrams are graphs representing fuzzy systems as social networks that overview at a glance the complete inference process. Rules are represented by nodes that are related if they cover instances in common. Node size is proportional to the number of instances covered by the selected rule. Relations reflect how pairs of rules cover a common input space; thus the larger number of instances commonly covered, the stronger relations.

To construct Fingrams from a FRBS we need a set of instances, the set of rules, and the inference mechanism used. The building process can be summarized by the following three phases:

1. **Fingram generation**: Construction of the complete graph according to a co-firing metric. The metric reflects how the rules of a FRBS cover the instances of the dataset. The simplest metric (as shown in Eq. 15) relates two rules (R_i and R_j) according to the number of instances covered in common by them ($|D_i \cap D_j|$) with respect to the total number of instances they individually cover ($|D_i|$ and $|D_j|$) where D_i and D_j are defined by Eq. 9.

$$m_{ij} = \begin{cases} \frac{|D_i \cap D_j|}{\sqrt{|D_i||D_j|}} & , \text{if } i \neq j \\ 0 & , \text{if } i = j \end{cases} \tag{15}$$

   with $m_{ij} \in [0, 1]$.
2. **Fingram scaling**: The graph generated in the previous phase usually appears highly dense and complex to analyze. Therefore, a suitable filtering of elements should produce a clearer graph where its backbone emerges. With that purpose, we take advantage of scaling algorithms that maintain all the nodes but only the most relevant links of

the previous graph. Scaling algorithms look at proximity information and yield new structures uncovering the underlying organization of the graph. These algorithms consider similarities, correlations or distances in order to prune a given graph according to the proximity between pairs of nodes.

We use Pathfinder algorithm [39] to scale Fingrams. It preserves the most important relations, producing no new unconnected nodes and showing up the underlying structure of the rule base.

3. **Fingram drawing**: A pleasant graphical representation is quite important to easily identify and understand the behavior of the FRBS under study. Force-based or force-directed algorithms are widely used for drawing graphs in the area of information science [40]. They locate the elements of a graph in a 2D or 3D space so that all the links are approximately of equal length and produce as few crosses as possible. Kamada-Kawai algorithm [41], one of the most relevant force-based algorithms, layouts Fingram elements in 2D following aesthetical criteria.

Fingrams can already deal with fuzzy association rules [34], fuzzy rule-based classifiers and regressors [33]. The different adaptations involve specific metrics and show relevant information according to their characteristics.

A specific software, Fingrams Generator [42], permits the creation of Fingrams no matter how the depicted FRBS was generated either by an expert or by another tool[1]. Also, a few software tools already allow the generation of FRBSs along with the creation and analysis of the related Fingrams, such as the data mining suites KEEL [43] and KNIME [44], or the fuzzy modeling tool GUAJE [45]. A complete survey on fuzzy systems software can be found at [46].

3. Extending Fuzzy Inference-grams to deal with Precise Fuzzy Systems

PFM creates accurate systems at the cost of being difficult to interpret. The data-driven process they follow and their inference mechanism occlude interpretability. This section presents the use of Fingrams for visual representation and exploratory analysis of the fuzzy inference mechanism in precise

[1]All the graphical representations shown in this paper are obtained by the use of Fingrams Generator available at: `http://www.sourceforge.net/projects/fingrams/`

fuzzy systems. To accomplish this challenging task, we enhance Fingrams with several adaptations and improvements. First, we propose a new co-firing metric that deals with rule firing degrees and weights of rules in accordance with the usual rule format (as shown in Eq. 4). Second, we introduce a graphical resource that displays the uncovered instances. And third, we define a new kind of Fingrams to visualize local inference for a given data instance.

Along the section we draw on an illustrative example to show the improvements introduced. We use the synthetic dataset prnn-synth [47] (used in [35] as benchmark) that has 250 instances (with 2 attributes) belonging to two classes. Here, we take advantage of its dimensionality to illustrate the generation of Fingrams. Fig. 1(a) shows a 2D plot with the instances in the dataset along with the input space covered by the FRBS created by FURIA. Fig. 1(b) provides the textual description of the FRBS. Notice that rules follow the format described by Eq. 4 and antecedents are defined by trapezoidal fuzzy sets.

3.1. PFM co-firing metric

A co-firing metric defines how nodes are related in accordance with how rules cover the input space. As seen in section 2.2, specific metrics cope with different types of fuzzy systems. The inference mechanism determines the used metric.

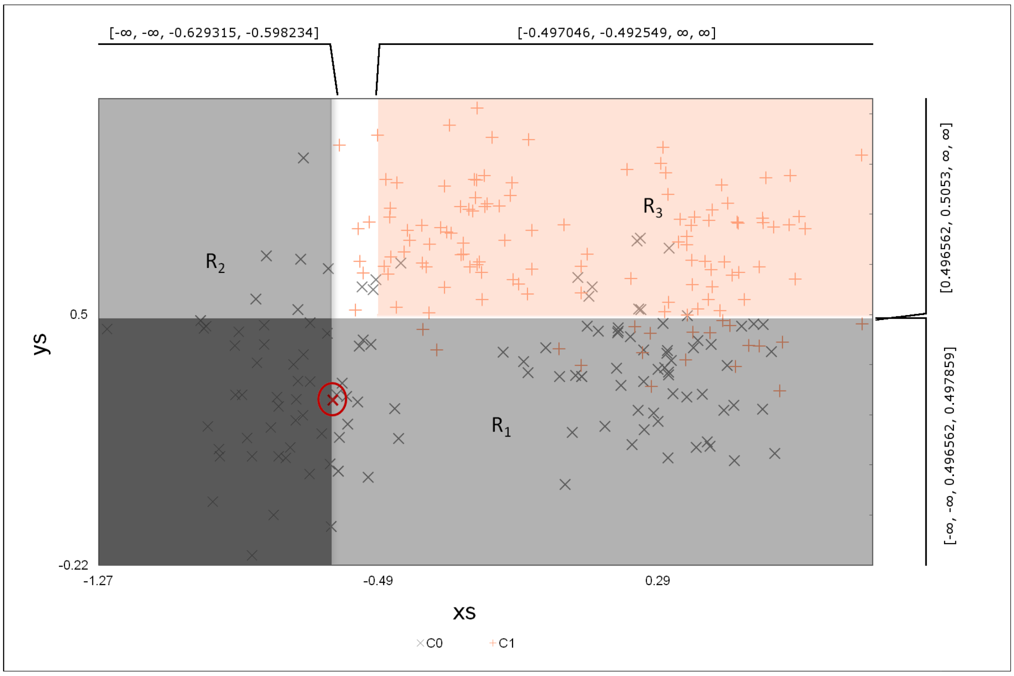
Here, we introduce a new metric that fits with the PFM inference mechanisms. This metric (Eq. 16) includes the firing degree up to which the data instances activate the rules as well as the rule weights:

$$m_{ij} = \begin{cases} \dfrac{\sum_{I_k \in \{D_i \cap D_j\}} \left(\min(\mu_{R_i}(I_k) \cdot w^i, \mu_{R_j}(I_k) \cdot w^j \right)}{\sqrt{(FD_{R_i} \cdot w^i) \cdot (FD_{R_j} \cdot w^j)}} & \text{, if } i \neq j \\ 0 & \text{, if } i = j \end{cases} \tag{16}$$

$\{D_i \cap D_j\}$ represents the set of instances firing both rules R_i and R_j at the same time. $\mu_{R_i}(I_k)$ is the firing degree of R_i given the data instance I_k (Eq. 6). FD_{R_i} reflects the firing degree of R_i (as presented in Eq. 8), and w^i states the weight assigned to R_i.

A square matrix M ($r \times r$ being r the number of rules in the FRBS) contains all interactions among rules. Each position in that matrix is computed using the co-firing metric.

Edges in the visual representation indicate interaction between rules. The thickness of the edge connecting rules R_i and R_j is proportional to m_{ij}. Its



(a) Input space.

R_1: IF ys in $[-\infty, -\infty$, 0.496562, 0.497859] THEN yc is C0 with w^1=0.844
R_2: IF xs in $[-\infty, -\infty$, -0.629315, -0.598234] THEN yc is C0 with w^2=0.979
R_3: IF ys in [0.496562, 0.5053, ∞, ∞] & xs in [-0.497046, -0.492549, ∞, ∞] THEN yc is C1 with w^3=0.909

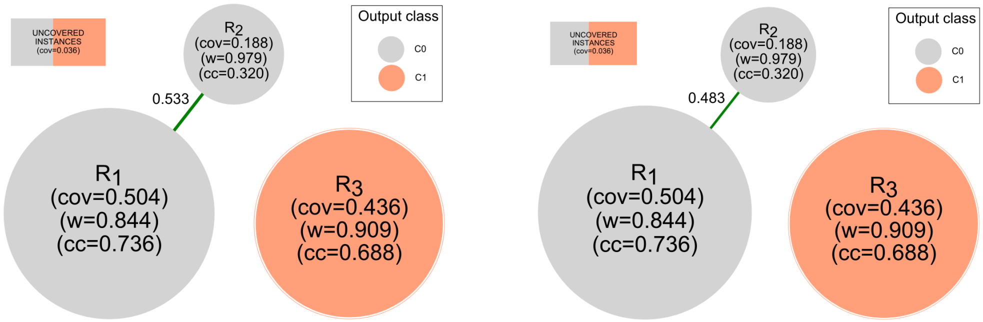
(b) Rules of the system.

Figure 1: FURIA system.

color depends on the class of the related rule. On the one hand, redundancies, i.e. links relating rules with the same output class, are represented in green. On the other hand, inconsistencies, i.e. relations between pairs of rules with different output class, are printed in red.

This new metric shows the behavior of the system at inference level more faithfully than the original one (presented in Eq. 15) but it is also more restrictive than the original. It is constrained by the firing degree of the data instances and the rule weights. As result, it yields weaker links among rules.

Nodes in Fingrams of PFM include the following information: rule identifier (R_i), coverage (cov), weight (w) and class coverage per rule (cc as presented in Eq. 12). Node color reflects the rule output class and the number of node borders indicates the number of antecedents in the rule depicted



(a) Fingram for the illustrative FRBS constructed with the original metric (Eq. 15).

(b) Fingram for the illustrative FRBS constructed with the new metric (Eq. 16).

Figure 2: FURIA-Fingrams of the illustrative example.

by the node. Each node size is proportional to the sum of firing degrees up to which the rule R_i covers all the instances (FD_{R_i})[2].

Fig. 2 presents the Fingrams built from the illustrative dataset and rules previously introduced (Fig. 1). Fig. 2(a) shows the Fingram built with the original metric (Eq. 15) whereas Fig. 2(b) presents the Fingram with the new metric (Eq. 16). Both pictures are in accordance with Fig. 1 which shows how the 3 rules cover the input space. Rules R_1 and R_2 cover some instances in common (lower left dark rectangle), so for, a link exists among the corresponding nodes in Fig. 2(a) and Fig. 2(b). On the contrary, rule R_3 does not cover instances in common with any of the remaining rules, thus it appears disconnected in the Fingram.

It is worthy to note that there are a few differences depending on the considered metric. Nodes of rule R_2 have slightly different sizes in both representations. Node size is a bit smaller when using the new metric due to the fact that a few instances are just partially covered by R_2. Furthermore, the thickness of the edge between rules R_1 and R_2 changes with each metric, $m_{1\,2} = 0.533$ with the original metric, while $m_{1\,2} = 0.483$ with the new metric.

[2]Note that node size is metric dependent. For the original metric nodes are proportional to the number of instances covered by the corresponding rule.

3.2. Representation of uncovered instances

The representation of Fingrams given in this proposal introduces a new visual artifact that shows the number of instances not covered by the set of rules (as presented in Eq. 7(b)). Instances not covered by any rule are represented by a rectangular node labeled as "UNCOVERED INSTANCES". Its height is proportional to the number of instances not covered by any rule. It is filled using vertical colored strips that give the proportion of uncovered instances related to each class.

The information provided by this special node helps in the comprehension of the system and its behavior with respect to the given dataset. We can compare its size with the size of the rest of nodes, giving us an idea of how important the set of uncovered instances is. Even more, we can identify visually whether instances of some classes are ignored by the system.

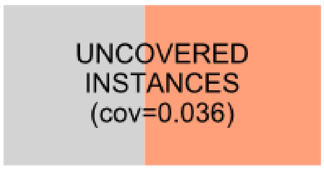


Figure 3: Node of uncovered instances (zoomed from Fig. 2).

We are analyzing the uncovered instances in the illustrative example. The white area in Fig. 1 shows that 9 out of 250 (3.6%) instances –4 of class C0 and 5 of class C1– are not covered. Fig. 3 (zoomed from Fig. 2) allows us to observe that situation. This node presents the output classes of the uncovered instances reflected in vertical colored strips.

3.3. Local view of the FRBSs inference mechanism

In previous proposals, Fingrams showed the inference mechanism of FRBSs from a global view point, i.e. observing how all the rules covered the complete given dataset. However, this fails in assisting the analysis of the inference mechanism at local level, i.e. for a single data instance.

Here, we propose a new Fingram view aimed at illustrating a partial view of the system when focusing on those rules that participate in the inference process regarding a single instance. Thus, we focus on those rules that specifically cover the related part of the input space. This allows us to better understand the behavior of the system in a specific situation. This *instance-based Fingram* filters the rules that are not fired by the selected instance. Notice that this task is not easy to carry out manually when dealing with complex sets of rules. Therefore, it provides a powerful filtering mechanism conducted by data.

This new representation is valid for any of the existing types of Fingrams and very valuable when studying instances that require a detailed analysis. The process to build these new instance-based Fingrams includes four steps. It is as follows:

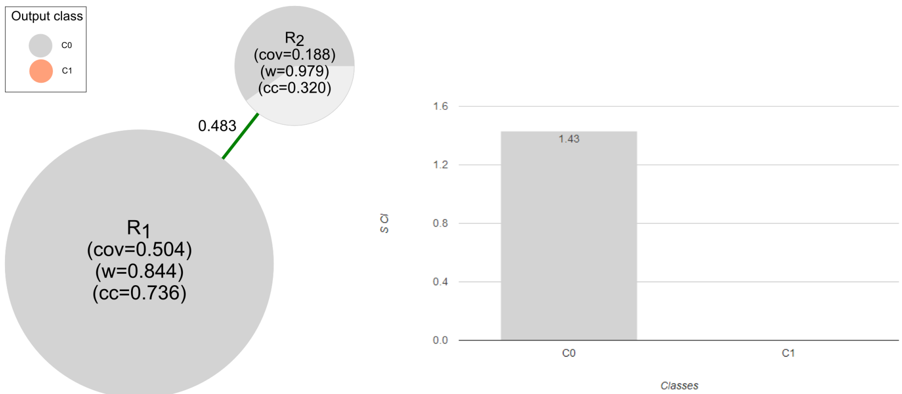
1. Generation of the network using the rule co-firing metric m taking as inputs the set of fuzzy rules, the fuzzy inference mechanism, and the set of instances.
2. Filtering of the network by only considering those rules that take part in the inference for the given instance I_k.
3. Scaling of the network through the use of the network scaling method.
4. Graphical representation of the resulting scaled network according to the network drawing method.

The firing degree up to the instance fires each rule (as seen in Eq. 6) is reflected by a radius dark zone in the corresponding node. The angle of dark color is proportional to the firing degree of the rule for the given instance, with a full dark colored node only in case that the firing degree is 1.

As complementary information to the Fingram, we propose to construct a histogram with the information of S_{c_l} for each class (as presented in Eq. 13). This way, we can compare the level of activation of each class for the given single instance.

We are overviewing this new graphical representation with the illustrative example. We select the instance highlighted in red in Fig. 1(a) (instance $I_{80} = \{-0.61, 0.25, \mathrm{C0}\}$) to analyze the inference mechanism locally. We follow the building process previously presented, obtaining the Fingram of Fig. 4(a). The histogram of class activation is shown in Fig. 4(b).

We can see that the node related to $R1$ is fully dark colored ($\mu_{R_1}(I_{80}) = 1$). On the contrary, R_2 is partially fired ($\mu_{R_2}(I_{80}) = 0.599$), therefore the related node is colored with a radius dark area. Both rules have the same output class (class C0), what is reflected by the same node color. In addition, Fig. 4(b) shows the levels of activation per class, where the winner class gives the



(a) Instance-based Fingram.

(b) Levels of firing per class.

Figure 4: Analysis of inference for instance $I_{80} = \{-0.61, 0.25, \text{C0}\}$.

inferred output[3]. Class C0 is correctly inferred, and there is no rule with class C1 activated in this specific case.

4. Analyzing FURIA Inference Mechanism by Fingrams

FURIA is a PFM algorithm that appears as one of the outstanding fuzzy classification algorithms when attending to accuracy [35]. It has demonstrated to be a robust method, performing properly in a bunch of scenarios, e.g., the prediction of missing attribute values on real cardiovascular data was improved by the use of FURIA in [48]; FURIA turned out as a competitive alternative for recommender systems in [49]; fuzzy rule-based ensembles were constructed including FURIA in order to derive good performance for high dimensional problems [50, 51]; etc.

FURIA creates very compact FRBSs that achieve high performance thanks to a specific inference mechanism. The comprehension of such inference mechanism is not straightforward [36]. However, it is a key issue to properly interpret the behavior of the FRBSs built from data.

[3]We should notice that the firing degree per class in FURIA is constructed by the addition of firing degrees of individual rules (as it will be seen in Eq. 20), instead of the usual maximum of firing (as shown in Eq. 13). That is the reason why the histograms may include values of S_{cl} greater than one.

This section presents the use of Fingrams for visual representation and exploratory analysis of the fuzzy inference mechanism in FURIA. To accomplish this challenging task, here we take advantage of the enhanced Fingrams presented in section 3. First, we present FURIA emphasizing in its induction of rules and inference mechanism. Second, we introduce some Fingram adaptations to deal with FURIA particularities. Then, we study the difficulties FURIA presents over benchmark datasets and how Fingrams can illuminate their comprehension. Finally, we analyze a complete case study of FURIA with Fingrams.

4.1. Fuzzy Unordered Rule Induction Algorithm

Fuzzy Unordered Rule Induction Algorithm, or FURIA in short, is a fuzzy rule-based classification method based on RIPPER algorithm [52]. It presents some modifications and extensions that outperforms the original [35, 53].

Now on, we are going to present the characteristics of FURIA emphasizing the procedure for rule induction as well as the inference mechanism.

4.1.1. Induction of rules

FURIA learning method follows RIPPER building strategy [52]. Nevertheless, differently from RIPPER, FURIA manages fuzzy rules. Thus, it does not construct a default rule, considers the order of rules irrelevant, and does not prune them.

FURIA fuzzifies the rule antecedents provided by RIPPER converting them to trapezoidal fuzzy sets with characteristic values $A_h^i = \{a_h^1, a_h^2, a_h^3, a_h^4\}$ (being the h antecedent of rule R_i). The original crisp intervals set the cores; and the supports are extended trying to maximize the *purity* of the rules, i.e. how well a rule covers the instances attending to the concordance of the outputs. We set p_i and n_i as the accumulative firing degrees of positive and negative covered instances for rule R_i respectively:

$$p_i = \sum_{I_k \in D_i^+} \mu_{R_i}(I_k) \tag{17a}$$

$$n_i = \sum_{I_k \in D_i^-} \mu_{R_i}(I_k) \tag{17b}$$

Thereby, we define the *purity* of rule R_i (pur_i) as follows:

$$pur_i = \frac{p_i}{p_i + n_i} \tag{18}$$

Rules in FURIA does not necessarily include an antecedent for all the input attributes and can also have more than one antecedent for the same attribute. We denote z_i the number of antecedents that are included in rule R_i. We can observe an example of rule induced by FURIA in Eq. 19.

$$R_i\text{: IF att2 in } [0.67, 0.7, \infty, \infty] \text{ \& att1 in } [-\infty, -\infty, 0.5, 0.51] \text{ \& att2 in } [-\infty, -\infty, 0.74, 0.75] \text{ THEN class is C0 with CF=0.899} \quad (19)$$

The order in which antecedents are learnt is important. It reflects the relevance of antecedents from the least to the most important one. Afterwards, the rule stretching mechanism exploits this ordering, as we will show bellow.

4.1.2. Inference mechanism

Given a set of rules, the inference mechanism operates as follows:

- If the instance I_k is covered by the set of induced rules, we define the level of activation for FURIA as:

$$S_{c_l}(I_k) = \sum_{R_i \in R^{c_l}} \left(\mu_{R_i}(I_k) \cdot w^i\right) = \sum_{R_i \in R^{c_l}} \left(\mu_{R_i}(I_k) \cdot CF_{R_i}\right) \quad (20)$$

  where the rule weight w^i is the *certainty factor* of rule R_i:

$$CF_{R_i} = \frac{2\frac{|D^{B^i}|}{|D|} + \sum_{I_k \in D^{B^i}} \left(\mu_{R_i}(I_k)\right)}{2 + \sum_{I_k \in D} \left(\mu_{R_i}(I_k)\right)} \quad (21)$$

  Then, we compute the output class as presented in Eq. 14, paying attention to the maximum of class activation levels.

  Note that S_{c_l} in FURIA (Eq. 20) is slightly different from the presented S_{c_l} in Eq. 13. FURIA considers the sum of levels of activation whereas, in general, PFM takes into account only the maximum level of activation.

- If I_k is not covered, FURIA dynamically creates a new set of rules (now on SR_k) from the initial ones by the so-called rule stretching mechanism. It iteratively tours every induced rule removing antecedents in order one by one (taking advantage of the ordering of antecedents set in the induction of rules) from the least to the most important, until the instance is covered. If all antecedents are removed from an individual rule, then this rule is discarded. When the stretched rule covers the instance I_k then the stretching mechanism stops for this rule, adds that stretched rule to SR_k and goes on with the next rule until all initial rules are checked for the given instance. Therefore, the new rule set SR_k includes at most the same number of initially induced rules (r).

- If SR_k is empty, i.e. all the rules are discarded: The class with the highest frequency in the dataset D is taken as output for instance I_k.
- Otherwise: FURIA produces as output the winner class with the final set of rules (SR_k) for the instance I_k:

$$output'(I_k) = y'_k \Rightarrow S_{y'_k}(I_k) = \max_{l=1...p} S'_{c_l}(I_k) \tag{22}$$

$$S'_{c_l}(I_k) = \max_{R_{i.q} \in R^{c_l}} \mu'_{R_{i.q}}(I_k) \cdot CF'_{R_{i.q}} \tag{23}$$

where

$$\mu'_{R_{i.q}}(I_k) = \begin{cases} \mu_{A_1^i}(x_k^1) \otimes ... \otimes \mu_{A_q^i}(x_k^q) & \text{, if } I_k \in ucv \ \& \ R_{i.q} \in SR_k \\ 0 & \text{, otherwise} \end{cases} \tag{24}$$

and

$$CF'_{R_{i.q}} = CF_{R_i} \cdot \frac{q+1}{z_i+2} \tag{25}$$

being CF_{R_i} the certain factor of the induced rule R_i, and q and z_i the number of antecedents in the stretched rule $R_{i.q}$ and in the induced rule R_i respectively.

It must be noticed that initial rules can produce as many stretching rules as the number of antecedents minus 1 ($q = 1, ..., z_i - 1$).

In case of a tie, no matter if the instance is handled by the set of induced rules or by the stretching mechanism, a decision in favor of the most frequent class is made.

The interested reader can find a deeper explanation of FURIA in [35, 53].

4.2. Fingram adaptations to deal with FURIA

4.2.1. Global view of the FURIA inference mechanism

As previously seen in section 4.1.2, the inference mechanism of FURIA has two modes of working depending on whether or not each data instance is covered by the set of induced rules. So for, we construct two different Fingrams, named as FURIA-Fingrams, with the aim of illustrating both situations:

Fingram from induced rules: We concentrate on the set of rules induced by FURIA for the given dataset and how such rules are co-fired according to the instances they cover. This Fingram is based on the co-firing metric presented in subsection 3.1. It may include a node labeled as "INSTANCES STRETCHING" that is built by following the procedure presented in subsection 3.2 to represent instances not covered by the induced rules.

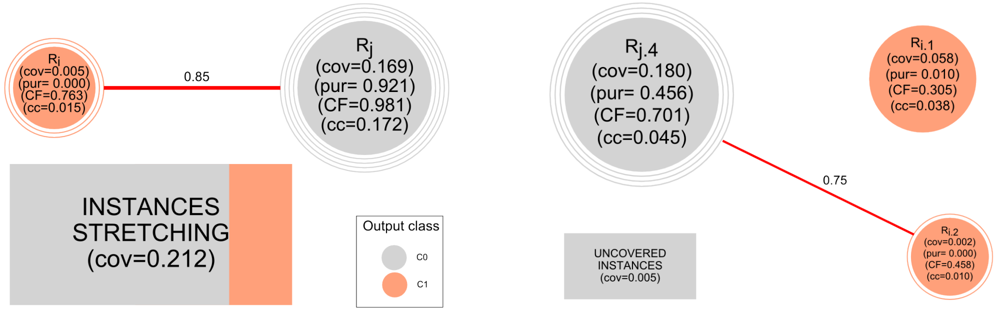
Fingram from stretched rules: We focus on the set of rules derived from the rule stretching mechanism in charge of handling those instances not covered by the induced rules, i.e. those instances that are included in the "INSTANCES STRETCHING" node in the previous Fingram. This Fingram uses the same metric presented in 3.1 but considering the new set of rules and only the instances not covered by the set of induced rules. The firing degrees of the stretched rules are computed as follows:

$$FD_{R_{i.q}} = \sum_{I_k \in ucv} \left(\mu'_{R_{i.q}}(I_k) \right) \tag{26}$$

where $\mu'_{R_{i.q}}(I_k)$ is computed by Eq. 24, and ucv is the set of uncovered instances as presented in Eq. 7(b).

The new rule identifiers preserve the names of the initial rules and add the number of antecedents kept. For example, if we have a rule R_5 with 3 antecedents, we may have $R_{5.2}$ and $R_{5.1}$ for rules with 2 and 1 antecedents respectively, derived by the rule stretching mechanism.

Notice that the stretching mechanism stops for a rule when each data instance is covered or there are no more antecedents to remove. So for, rules coming from the same initial rule cannot be related because they never handled any instance in common, e.g. $R_{5.2}$ and $R_{5.1}$ can not be



(a) Fingram from induced rules. (b) Fingram from stretched rules.

Figure 5: Interpretation of FURIA-Fingrams.

related because they do not jointly appear in any set of stretched rules (SR_k) for an instance I_k (as presented in subsection 4.1).

In the special case that all the initial rules yield to the empty rule for a given instance, then it remains uncovered after the rule stretching mechanism. Therefore, this Fingram would include a special node labeled as "UNCOVERED INSTANCES" which represents those instances that are not covered yet at the end of the stretching mechanism.

4.2.2. Local view of the FURIA inference mechanism

As previously seen in section 3.3, we have introduced a new local view of precise fuzzy systems. Here we slightly adapt the representation to the particularities of FURIA.

As shown in Eq. 20, the level of activation per class, when dealing with the set of induced rules, is the addition of activation levels of the individual rules for the given instance. This differs with respect to the usual behavior of PFM as presented in Eq. 13. Nevertheless, note that the activation per class in the stretching mechanism performs as usual (Eq. 23). Obviously, this kind of tricks make harder the comprehension of FURIA.

4.2.3. Visual adaptations to deal with FURIA-Fingrams

Fig. 5 shows an example that illustrates the interpretation of FURIA-Fingrams. Each node displays the information of the corresponding rule: coverage (cov) –as defined in Eq. 11–, purity (pur) –Eq. 18–, certain factor (CF) –Eq. 21– , and proportion of instances of the rule class that are covered by it (cc) –Eq. 12–. Node color indicates output class, and the number of borders indicates the number of rule antecedents. Edges represent the relation among rules as presented in section 3.1. An additional node represents the number of uncovered instances of the dataset.

Fig. 5(a) shows a Fingram constructed from the set of induced rules. R_i and R_j have different output class ($B^i \neq B^j$), and they are plotted in different colors. R_i has 3 antecedents and R_j has 5, as deduced from the number of node borders. Both rules cover some instances in common ($m_{ij} = 0.85$). Additionally, part of the instances of the dataset are not covered by the set of induced rules and they are included in the node "INSTANCES STRETCHING". Most of them are related to class C0 as deduced from its distribution of colors.

Fig. 5(b) shows a Fingram built from the rules created by the stretching mechanism. $R_{i.1}$ and $R_{i.2}$ are constructed from the initial rule R_i and $R_{j.4}$ from R_j. Note that these rules have the same output class (node color) as the ones from which they are derived. Node identifiers indicate the number of antecedents kept, piece of information that is also displayed in the number of borders the nodes have. We observe that even using the stretching mechanism, some instances (all with output class C0) are not covered.

4.3. Analysis of FURIA over benchmark datasets

This subsection has the aim to discuss the characteristics of FRBSs constructed with FURIA by means of Fingrams.

First of all, we repeated the experimentation made in [35] over 45 benchmark datasets. The focus is to identify how FURIA performs along with details about how the generated fuzzy rules cover the input space.

Table 1 summarizes the main characteristics –number of instances (*#INS*), attributes (*#ATT*), classes (*#CL*) and instances per class (*#INS/CL*)– of the datasets under study. They cover a wide range of scenarios; from small datasets with tens of instances, to medium-size and even large datasets including thousands of instances; with synthetic and real-world data sets; from low dimensional (from 2 attributes) to high dimensional data sets (up to 216 attributes); from 2-class to 16-class problems; and balanced but also highly unbalanced data sets.

Id	Dataset	#INS	#ATT	#CL	#INS/CL
1	acd-autorship	841	70	4	317/296/173/55
2	acd-bankruptcy	50	6	2	25/25
3	acd-cyyoung8092	97	10	2	73/24
4	acd-cyyoung9302	92	10	2	73/19
5	acd-esr	32	2	2	26/6
6	acd-halloffame	1340	18	3	1215/68/57
7	acd-lawsuit	264	4	2	245/19
8	acd-votesurvey	48	4	4	18/15/10/5
9	biomed	209	10	2	134/75
10	cars	406	9	3	254/79/73
11	collins	500	23	15	80/75/48/44/36/30/29/29/29/27/24/17/17/9/6
12	ecoli	336	7	8	143/77/52/35/20/5/2/2
13	eucalyptus	736	28	5	214/180/130/107/105
14	glass	214	9	7	79/70/29/17/13/9/0
15	haberman	306	3	2	225/81
16	heart-statlog	270	13	2	150/120
17	ionosphere	351	34	2	225/126
18	iris	150	4	3	50/50/50
19	liver-disorders	345	6	2	200/145
20	metStatCoordinates	4748	3	16	1043/504/475/387/384/355/353/299/276/219/188 /176/42/22/14/11
21	metStatRainfall	4748	12	16	1043/504/475/387/384/355/353/299/276/219/188 /176/42/22/14/11
22	metStatRST	336	3	12	69/64/43/29/28/25/18/16/12/11/13/8
23	metStatSunshine	422	12	14	85/78/52/34/34/32/24/17/16/15/13/12/5/5
24	metStatTemp	673	12	15	142/136/72/63/55/40/39/28/18/18/17/15/14/10 /6
25	mfeat-factors	2000	216	10	200/200/200/200/200/200/200/200/200/200
26	mfeat-fourier	2000	76	10	200/200/200/200/200/200/200/200/200/200
27	mfeat-karhunen	2000	64	10	200/200/200/200/200/200/200/200/200/200
28	mfeat-morphological	2000	6	10	200/200/200/200/200/200/200/200/200/200
29	mfeat-zernike	2000	47	10	200/200/200/200/200/200/200/200/200/200
30	optdigits	5620	64	10	572/571/568/566/562/558/558/557/554/554
31	page-blocks	5473	10	5	4913/329/115/88/28
32	pasture-production	36	22	3	12/12/12
33	pendigits	10992	16	10	1144/1144/1143/1143/1142/1056/1055/1055/1055 /1055
34	pima-diabetes	768	8	2	500/268
35	prnn-synth	250	2	2	125/125
36	schizo	340	25	2	177/163
37	segment	2310	19	7	330/330/330/330/330/330/330
38	sonar	208	60	2	111/97
39	squash-unstored	52	31	3	24/24/4
40	synthetic-control	600	61	6	100/100/100/100/100/100
41	vehicle	846	18	4	218/217/212/199
42	vowel	990	12	11	90/90/90/90/90/90/90/90/90/90/90
43	waveform	5000	40	3	1692/1655/1653
44	wine	178	13	3	71/59/48
45	wisconsin-breast-cancer	699	10	2	458/241

Table 1: Benchmark data sets under study. *#INS* is the number of instances, *#ATT* is the number of attributes, *#CL* is the number of classes and *#INS/CL* stands for the number of instances per class.

Table 2 shows the results of performing 3-fold cross validation 100 times by randomly splitting the datasets into 3 folds[4]. It presents the accuracy (*ACC*) and number of rules (*#R*) produced by FURIA for all the 45 datasets described in Table 1. Additionally, we reckon the percentage of instances not covered by the set of induced rules (*%INST2STR*), the number of rules created dynamically by the stretching mechanism (*#STR-R*), and the percentage of instances not covered even by the stretching mechanism (*%UNC-INS*). Finally, we include the percentage of classes with at least 1 rule in the set of induced rules (*%CL-1R*) –meaning that the system can infer that classes–, the number of induced rules per class (*#R/CL*), and the number of antecedents per induced rule (*#ANT/R*). For each dataset, we report the average (Avg.) and the standard deviation (Std.) of all indicators we have just introduced above.

We can learn valuable knowledge about fuzzy systems constructed by FURIA from Tables 1 and 2. FURIA produces accurate results for most of the benchmark datasets, overcoming other methods as presented in [35]. The most notable exceptions are datasets 8, 22, 23 and 24 where FURIA does not classified properly.

FURIA produces compact rule sets with low number of rules (25.31 in average), low number of rules per class (4.22 in average) and low number of antecedents per class (2.71 in average). However, this does not guarantee interpretability, and those systems are hard to interpret due to its inference mechanism.

Most of the systems FURIA create covered all the classes (95.69 of classes covered in average). For large balanced datasets FURIA induces rule sets that infer all the classes. However, this does not happen for unbalanced datasets, such as datasets 8, 22 and 23. We can observe results with a limited accuracy for unbalanced datasets.

We observe that in case of datasets with low number of instances triggering the stretching mechanism, FURIA creates low number of stretching rules (as the ones created for datasets 7, 11 or 31) and produces very good results. On the contrary, those systems that highly depend on the stretching mechanism perform badly. As example, datasets 8 and 22 fail to produce

[4]The authors in [35] split the data set into 2/3 for training and 1/3 for testing, repeating the procedure 100 times, that is slightly different to our experimentation but tend to stabilize and produce very similar results.

Id	ACC		#R		%INS2STR		#STR-R		%UNC-INS		%CL-1R		#R/CL		#ANT/R	
	Avg.	Std.	Avg.	Std.	Avg.	Std.	Avg.	Std.	Avg.	Std.	Avg.	Std.	Avg.	Std.	Avg.	Std.
1	95.51	0.66	16.19	1.24	4.56	1.50	13.61	3.26	0.21	0.33	100.00	0.00	4.05	0.31	2.68	0.14
2	82.54	3.45	3.30	0.70	2.24	4.74	0.23	0.50	0.76	3.36	100.00	0.00	1.65	0.35	1.63	0.31
3	79.88	2.91	4.55	1.43	5.14	6.69	1.03	1.34	0.92	2.97	99.83	2.88	2.28	0.71	1.57	0.38
4	82.87	2.78	4.31	1.31	4.62	5.69	0.64	0.99	1.87	3.70	100.00	0.00	2.16	0.65	1.53	0.35
5	81.06	3.12	**1.96**	0.56	2.78	8.02	0.03	0.17	2.28	6.96	91.17	19.07	1.08	0.24	**1.23**	0.31
6	92.92	0.38	15.10	3.08	3.50	1.38	17.60	5.03	0.00	0.03	99.00	5.69	5.1	1.09	2.94	0.28
7	**97.88**	0.61	3.40	0.73	**0.27**	0.73	**0.19**	0.48	0.00	0.07	100.00	0.00	1.7	0.36	1.46	0.19
8	35.73	3.99	2.35	1.38	**66.27**	23.42	1.15	1.22	**48.92**	32.36	**44.75**	20.40	1.25	0.59	1.62	0.63
9	87.88	1.91	8.39	1.41	5.53	3.30	3.83	2.06	0.14	0.65	100.00	0.00	4.19	0.71	2.12	0.26
10	79.28	1.44	14.04	2.61	11.06	4.18	13.30	3.30	0.00	0.00	100.00	0.00	4.68	0.87	2.61	0.23
11	96.60	0.79	16.03	0.72	0.78	1.22	0.65	1.46	0.41	0.77	100.00	0.00	1.07	0.05	1.98	0.06
12	83.04	1.34	12.80	2.17	6.09	3.47	7.74	3.75	0.45	0.79	79.08	6.04	2.03	0.32	2.61	0.20
13	60.85	1.36	14.86	2.76	29.36	6.82	21.25	7.44	0.17	1.73	98.47	5.57	3.03	0.56	2.89	0.34
14	68.43	2.83	11.76	2.29	16.10	6.06	10.59	3.23	0.81	1.51	92.11	8.32	2.13	0.39	2.66	0.24
15	72.97	1.16	3.55	1.13	11.55	9.53	0.73	0.98	7.84	9.57	88.50	21.04	2.07	0.63	1.38	0.29
16	79.86	1.86	7.88	2.39	10.34	5.91	7.49	3.93	0.02	0.23	100.00	0.00	3.94	1.2	2.62	0.44
17	89.42	1.14	9.19	1.84	4.83	2.85	5.93	3.08	0.04	0.31	100.00	0.00	4.59	0.92	2.30	0.34
18	94.11	1.26	4.32	0.74	1.04	1.97	0.54	0.97	0.13	0.87	100.00	0.00	1.44	0.25	1.80	0.28
19	66.92	2.06	8.65	3.32	18.82	9.05	9.59	5.61	0.63	2.21	100.00	0.00	4.33	1.66	2.50	0.43
20	93.00	0.27	71.16	4.42	1.70	0.48	63.16	11.83	0.00	0.01	99.96	0.51	4.45	0.28	3.71	0.08
21	64.53	0.51	**123.24**	7.09	23.14	1.74	**342.90**	22.05	0.00	0.00	96.77	3.70	7.97	0.54	4.61	0.10
22	**33.51**	2.38	10.16	2.23	50.68	10.49	13.77	3.72	6.15	8.55	65.39	11.25	1.3	0.21	2.75	0.24
23	48.74	1.59	24.74	3.87	28.03	5.39	32.54	6.18	0.65	1.37	75.79	8.01	2.34	0.33	2.82	0.18
24	50.59	1.59	31.75	4.67	33.22	5.75	45.29	8.47	0.38	1.30	86.22	7.39	2.46	0.34	2.94	0.16
25	92.28	0.48	45.25	2.50	6.21	1.01	56.21	5.57	0.02	0.05	100.00	0.00	4.52	0.25	2.98	0.09
26	76.76	0.76	53.41	4.02	13.44	1.73	115.58	10.13	0.00	0.00	100.00	0.00	5.34	0.4	3.83	0.14
27	86.50	0.57	59.27	3.54	9.02	1.27	100.07	7.53	0.00	0.01	100.00	0.00	5.93	0.35	3.27	0.10
28	71.83	0.81	25.22	2.83	11.57	4.89	28.50	6.11	0.09	0.33	98.67	3.59	2.56	0.28	2.77	0.16
29	72.10	10.33	44.19	4.83	17.22	3.42	97.10	13.81	0.00	0.01	97.27	11.85	4.46	0.62	3.74	0.45
30	94.83	0.28	97.39	3.82	4.11	0.56	248.15	14.18	0.00	0.00	100.00	0.00	9.74	0.38	5.00	0.09
31	97.03	0.17	23.65	2.98	0.91	0.43	27.59	7.85	0.00	0.01	100.00	0.00	4.73	0.6	3.42	0.24
32	74.11	4.95	3.18	0.63	21.72	12.98	0.49	0.62	16.00	13.39	95.89	10.96	**1.1**	0.17	1.35	0.26
33	97.73	0.13	112.90	3.99	1.25	0.24	212.54	20.04	0.00	0.00	100.00	0.00	11.29	0.4	4.87	0.08
34	74.94	1.09	7.30	2.77	10.21	5.85	7.18	5.50	0.19	0.91	100.00	0.00	3.65	1.38	2.28	0.53
35	83.75	1.56	4.47	1.64	3.73	4.97	0.99	1.26	0.66	2.62	100.00	0.00	2.23	0.82	1.44	0.34
36	80.42	3.03	15.22	2.78	14.85	7.04	11.83	4.01	0.35	1.58	100.00	0.00	7.61	1.39	2.34	0.18
37	96.58	0.35	27.10	2.25	2.00	0.64	29.28	5.56	0.00	0.00	100.00	0.00	3.87	0.32	3.41	0.16
38	76.55	2.78	8.44	1.14	10.90	4.24	6.55	2.01	0.20	0.62	100.00	0.00	4.22	0.57	2.41	0.22
39	77.00	4.28	4.13	0.78	6.49	9.23	0.45	0.65	2.35	6.38	97.00	9.54	1.43	0.28	1.46	0.23
40	89.53	1.40	16.75	1.36	7.66	2.48	15.84	3.25	0.17	0.33	100.00	0.00	2.79	0.23	2.77	0.17
41	70.20	1.22	20.71	2.98	20.57	5.10	32.82	6.28	0.04	0.22	100.00	0.00	5.18	0.75	3.28	0.20
42	75.83	1.69	53.12	3.38	12.23	2.27	86.89	8.53	0.00	0.00	100.00	0.00	4.83	0.31	3.63	0.13
43	82.13	0.40	76.07	4.81	10.77	1.17	265.52	13.99	0.00	0.00	100.00	0.00	**25.36**	1.6	**6.18**	0.14
44	93.58	1.56	5.84	0.88	4.52	3.17	1.38	1.22	2.12	2.58	100.00	0.00	1.95	0.29	1.92	0.17
45	95.50	0.55	11.77	2.01	1.98	1.19	7.71	3.51	0.00	0.02	100.00	0.00	5.88	1	2.86	0.27
Avg.	**79.50**	1.77	**25.31**	2.44	**11.84**	4.54	**43.70**	5.39	**2.11**	2.42	**95.69**	3.46	**4.22**	0.58	**2.71**	0.24
Std.	15.54	1.76	29.93	1.44	13.14	4.22	77.64	5.08	7.64	5.40	10.60	5.81	3.94	0.39	1.07	0.13

Table 2: Reported results repeating 100 times 3-fold cross validation for each dataset. *ACC* is the accuracy, *#R* is the number of rules, *%INST2STR* is the percentage of instances not covered by the set of induced rules, *#STR-R* is the number of rules created dynamically by the stretching mechanism, *%UNC-INS* is the percentage of instances not covered even by the stretching mechanism, *%CL-1R* is the percentage of classes with at least 1 rule in the set of induced rules, *#R/CL* is the number of induced rules per class, and *#ANT/R* is the number of antecedents per induced rule.

good results due to its highly dependency on the stretching mechanism. The stretching mechanism in dataset 8 fails to cover the instances and the majority class is inferred for 48.92% of the instances. In the other case, the system constructed for dataset 22 handles 44% of the instances in the stretching mechanism but with poor results.

Finally, we carried out an experiment that allowed us to understand the way how the rules built by FURIA cover the input space and how the stretching method works. To do so, we induced rules from the complete set of instances available for each dataset, and then we built and analyzed the associated FURIA-Fingrams.

Table 3 reports the number of rules induced by FURIA (*#R*), the percentage of instances triggering the stretching mechanism (*%INST2STR*), the number of rules created by the stretching mechanism (*#STR-R*), the percentage of uncovered instances (*%UNC-INS*), the density of the Fingram representing the induced rules (*DENS-IND-R*), the density of the Fingram depicting the stretched rules (*DENS-STR-R*), and the percentage of inconsistencies (*%INC*), i.e. relations between rules with different output class, for the two Fingrams, the one considering induced rules (*%INC-IND-R*) and the one regarding stretched rules (*%INC-STR-R*). We should point out that we compute the density of Fingrams as:

$$DENS = AE/PC * 100 \quad (27)$$

where AC is the number of actual edges (without scaling) in the graph and PC counts all the potential connections, i.e. $PC = \frac{r*(r-1)}{2}$ with r the total number of rules[5].

Comparing figures in tables 2 and 3 we observe that the number of induced rules grow according to the number of instances in the training dataset. Results in table 2 were computed considering 2/3 of all the instances for training whereas table 3 takes the whole dataset for training. The percentage of instances not covered by the set of induced rules decreases considerably when inducing rules with the complete dataset, as expected. Even more, the percentage of uncovered instances after stretching drops drastically for all but dataset 15. There, FURIA fails to induce a good set of rules and the low dimensions do not allow the stretching mechanism to perform properly.

[5]Nodes that graphically represent uncovered instances are not considered in this calculus.

Id	#R	%INS2STR	#STR-R	%UNC-INS	DENS-IND-R	DENS-STR-R	%INC-IND-R	%INC-STR-R
1	20	0.36	8	0.00	25.26	28.57	6.25	75.00
2	4	0.00	-	0.00	16.67	-	0.00	-
3	5	0.00	-	0.00	60.00	-	33.33	-
4	6	3.26	1	0.00	73.33	100.00	36.36	-
5	**2**	0.00	-	0.00	0.00	-	0.00	-
6	24	2.69	46	0.00	32.25	25.12	11.24	65.77
7	4	0.00	-	0.00	66.67	-	25.00	-
8	4	**60.42**	5	10.42	0.00	0.00	0.00	-
9	12	0.00	-	0.00	54.55	-	22.22	-
10	26	3.94	22	0.00	8.00	34.63	15.38	77.50
11	16	0.00	-	0.00	0.83	-	0.00	-
12	20	3.87	18	0.60	11.05	31.37	23.81	83.33
13	20	20.52	38	0.00	24.21	25.18	43.48	83.05
14	17	6.07	26	0.00	13.97	32.31	47.37	88.57
15	3	19.93	0	**19.93**	**100.00**	-	0.00	-
16	7	11.11	9	0.00	61.90	50.00	30.77	72.22
17	11	2.85	8	0.00	67.27	46.43	18.92	69.23
18	5	0.00	-	0.00	20.00	-	0.00	-
19	16	11.88	17	0.00	41.67	25.74	20.00	**54.29**
20	89	1.31	118	0.00	6.36	10.04	23.69	83.55
21	**157**	22.24	**532**	0.00	4.47	17.43	33.03	93.22
22	12	50.89	23	0.00	4.55	28.85	33.33	**97.26**
23	32	23.70	58	0.00	7.26	23.77	52.78	92.62
24	40	30.46	71	0.00	0.00	22.09	0.00	93.99
25	55	0.85	57	0.00	6.94	15.10	1.94	94.19
26	66	8.40	157	0.00	8.81	23.31	23.28	91.49
27	80	1.25	114	0.00	6.68	14.05	6.64	94.03
28	28	7.55	35	0.00	10.85	30.08	39.02	89.94
29	50	19.25	133	0.00	10.61	21.95	13.85	90.56
30	122	0.55	230	0.00	7.97	13.24	0.00	90.62
31	31	1.13	58	0.00	21.94	26.98	16.67	74.66
32	4	13.89	2	5.56	16.67	0.00	**100.00**	-
33	136	0.33	250	0.00	6.54	12.79	1.17	90.18
34	5	11.07	2	0.00	70.00	0.00	42.86	-
35	3	3.60	1	0.00	33.33	100.00	0.00	-
36	16	10.88	16	0.00	9.17	35.83	9.09	67.44
37	34	0.56	44	0.00	13.90	23.15	6.41	76.71
38	12	2.40	5	0.00	46.97	50.00	6.45	80.00
39	4	0.00	-	0.00	33.33	-	50.00	-
40	20	1.17	15	0.00	14.74	21.90	7.14	86.96
41	25	26.83	55	0.00	12.67	24.31	2.63	80.33
42	67	3.33	114	0.00	3.62	21.16	11.25	92.44
43	91	10.24	408	0.00	31.33	7.97	4.60	61.27
44	7	0.56	1	0.00	33.33	100.00	0.00	-
45	15	2.15	21	0.00	48.57	29.05	1.96	62.30
Avg.	**31.62**	**8.92**	**73.46**	**0.81**	**25.52**	**29.79**	**18.26**	**81.76**
Std.	37.28	13.16	114.96	3.40	24.48	24.58	20.33	11.65

Table 3: Results for all the datasets with all the available data. *#R* is the number of rules induced by FURIA, *%INST2STR* is the percentage of instances triggering the stretching mechanism, *#STR-R* is the number of rules created by the stretching mechanism, *%UNC-INS* is the percentage of uncovered instances, *DENS-IND-R* is the density of the Fingram representing the induced rules, *DENS-STR-R* is the density of the Fingram depicting the stretched rules, *%INC-IND-R* is the percentage of inconsistencies in the Fingram from induced rules, and *%INC-STR-R* is the percentage of inconsistencies in the Fingram from stretched rules.

The structure of Fingrams built from the set of induced rules gives us very useful information. They usually present a sparse structure that reflects the scarce interaction among rules (*DENS-IND-R* is about 25% in table 3). Notice that this is not the typical structure we observe with other kinds of FRBSs [33] where the input space tends to be densely covered, producing high interaction among rules. Fingrams representing the set of induced rules usually have a large number of redundancies, i.e. two rules with the same output class cover several instances in common, due to the building strategy followed by FURIA. Moreover, the few inconsistencies they present (18.26% in average) show that rules with different output class rarely cover instances in common.

On the contrary, Fingrams depicting stretched rules are slightly more dense (*DENS-STR-R* is close to 30% in table 3). The rules produced by the stretching mechanism generalize the induced rules by removing antecedents with the aim of covering more of the input space and, so for, they tend to cover many instances in common. Even more, they usually produce many inconsistencies (81.76% in average) due to the way the stretching of rules works.

4.4. Illustrative example of FURIA-Fingrams for a real world dataset

In this section we outline the potentials of Fingrams for the analysis of FURIA over a real world dataset. We have selected the dataset *ecoli* (dataset 12 in Table 1) from UCI [54, 55]. We select it because of its size (336 instances), number of classes (8 unbalanced classes) and the FRBS size and properties created by FURIA (as seen in Table 3), which are close to the average.

Ecoli dataset includes 336 E.coli proteins of 8 different classes with 7 attributes calculated from the amino acid sequences. The attributes are McGeoch's method for signal sequence recognition –*mcg*–, Von Heijne's method for signal sequence recognition –*gvh*–, Von Heijne's Signal Peptidase II consensus sequence score –*lip*–, presence of charge on N-terminus of predicted lipoproteins –*chg*–, score of discriminant analysis of the amino acid content of outer membrane and periplasmic proteins –*aac*–, score of the ALOM membrane spanning region prediction program –*alm1*–, and score of ALOM program after excluding putative cleavable signal regions from the sequence –*alm2*–. The distribution of instances per class is as follows: cytoplasm proteins –*cp*– (143 instances), inner membrane without signal sequence –*im*–

(77), perisplasmic proteins –*pp*– (52), inner membrane proteins with an uncleavable signal sequence –*imU*– (35), other outer membrane proteins –*om*– (20), outer membrane lipoproteins –*omL*– (5), inner membrane lipoprotein –*imL*– (2) and inner membrane proteins with cleavable signal sequence –*imS*– (2).

FURIA produces an accurate fuzzy system for ecoli dataset that outperforms several alternative methods as shown in [35]. It induces the following 20 fuzzy rules from the complete dataset. This is a compact rule set but hard to interpret.

R_1: IF alm1 in $[-\infty, -\infty, 0.38, 0.39]$ & gvh in $[-\infty, -\infty, 0.55, 0.57]$ THEN class is *cp* with CF=0.973
R_2: IF mcg in $[-\infty, -\infty, 0.44, 0.52]$ & alm1 in $[-\infty, -\infty, 0.55, 0.58]$ THEN class is *cp* with CF=0.951
R_3: IF alm1 in $[-\infty, -\infty, 0.47, 0.49]$ & mcg in $[-\infty, -\infty, 0.59, 0.63]$ & gvh in $[-\infty, -\infty, 0.57, 0.59]$ THEN class is *cp* with CF=0.955
R_4: IF alm1 in $[0.75, 0.76, \infty, \infty]$ & mcg in $[-\infty, -\infty, 0.61, 0.62]$ THEN class is *im* with CF=0.956
R_5: IF alm1 in $[0.55, 0.61, \infty, \infty]$ & mcg in $[-\infty, -\infty, 0.45, 0.47]$ THEN class is *im* with CF=0.951
R_6: IF alm2 in $[0.59, 0.63, \infty, \infty]$ & mcg in $[-\infty, -\infty, 0.74, 0.79]$ & alm2 in $[-\infty, -\infty, 0.73, 0.74]$ & gvh in $[0.45, 0.46, \infty, \infty]$ THEN class is *im* with CF=0.904
R_7: IF alm1 in $[0.82, 0.85, \infty, \infty]$ & mcg in $[-\infty, -\infty, 0.74, 0.86]$ & gvh in $[-\infty, -\infty, 0.52, 0.53]$ THEN class is *im* with CF=0.902
R_8: IF alm1 in $[0.55, 0.62, \infty, \infty]$ & alm1 in $[-\infty, -\infty, 0.72, 0.74]$ & mcg in $[-\infty, -\infty, 0.61, 0.63]$ & gvh in $[-\infty, -\infty, 0.55, 0.6]$ THEN class is *im* with CF=0.916
R_9: IF alm2 in $[0.35, 0.74, \infty, \infty]$ & alm1 in $[-\infty, -\infty, 0.72, 0.73]$ & mcg in $[0.81, 0.83, \infty, \infty]$ THEN class is *im* with CF=0.692
R_{10}: IF alm2 in $[0.7, 0.74, \infty, \infty]$ & aac in $[0.7, 0.71, \infty, \infty]$ THEN class is *im* with CF=0.615
R_{11}: IF gvh in $[0.58, 0.59, \infty, \infty]$ & aac in $[-\infty, -\infty, 0.47, 0.57]$ & alm1 in $[-\infty, -\infty, 0.65, 0.67]$ & alm1 in $[0.35, 0.36, \infty, \infty]$ THEN class is *pp* with CF=0.954
R_{12}: IF gvh in $[0.53, 0.56, \infty, \infty]$ & mcg in $[0.61, 0.63, \infty, \infty]$ & aac in $[-\infty, -\infty, 0.63, 0.65]$ & alm1 in $[-\infty, -\infty, 0.52, 0.53]$ & aac in $[0.45, 0.46, \infty, \infty]$ THEN class is *pp* with CF=0.911
R_{13}: IF mcg in $[0.67, 0.7, \infty, \infty]$ & aac in $[-\infty, -\infty, 0.5, 0.51]$ & mcg in $[-\infty, -\infty, 0.74, 0.75]$ THEN class is *pp* with CF=0.899
R_{14}: IF alm2 in $[0.39, 0.62, \infty, \infty]$ & mcg in $[0.74, 0.75, \infty, \infty]$ THEN class is *imU* with CF=0.800
R_{15}: IF alm2 in $[0.46, 0.66, \infty, \infty]$ & mcg in $[0.58, 0.62, \infty, \infty]$ & gvh in $[-\infty, -\infty, 0.45, 0.46]$ & mcg in $[-\infty, -\infty, 0.67, 0.69]$ THEN class is *imU* with CF=0.713
R_{16}: IF alm2 in $[0.73, 0.74, \infty, \infty]$ & alm1 in $[-\infty, -\infty, 0.75, 0.76]$ & mcg in $[0.45, 0.47, \infty, \infty]$ THEN class is *imU* with CF=0.581
R_{17}: IF aac in $[0.66, 0.68, \infty, \infty]$ & alm2 in $[-\infty, -\infty, 0.38, 0.66]$ & mcg in $[0.31, 0.52, \infty, \infty]$ THEN class is *om* with CF=0.891
R_{18}: IF gvh in $[0.67, 0.68, \infty, \infty]$ & mcg in $[-\infty, -\infty, 0.61, 0.62]$ THEN class is *om* with CF=0.687
R_{19}: IF lip in $[0.48, 1, \infty, \infty]$ & alm2 in $[-\infty, -\infty, 0.36, 0.52]$ & chg in $[-\infty, -\infty, 0.5, 1]$ THEN class is *omL* with CF=0.719
R_{20}: IF lip in $[0.48, 1, \infty, \infty]$ & aac in $[-\infty, -\infty, 0.51, 0.52]$ & mcg in $[-\infty, -\infty, 0.75, 0.77]$ THEN class is *imL* with CF=0.503

The rule set uses the complete set of attributes and covers 7 out of the 8 classes in the dataset (there is no rule for the minority class *imS*). Notice that rules R_6, R_8, R_{11}, R_{12}, R_{13}, and R_{15} repeat attributes in their antecedents (attributes *alm2*, *alm1*, *alm1*, *aac*, *mcg* and *mcg* respectively). This fact does not affect the inference by the set of induced rules but is transcendental in the stretching mechanism.

Fig. 6 presents the Fingrams[6] depicting the induced rule set built with the new PFM co-firing metric (Eq. 16) and the legend of class colors (additional details are reported in Table A.4 in Appendix A). In the top left

[6]Contrary to the previous Fingrams, those in Figs. 6 and 7 only include the rule identifiers in the nodes for the sake of clarity. Appendix A includes additional information about them.

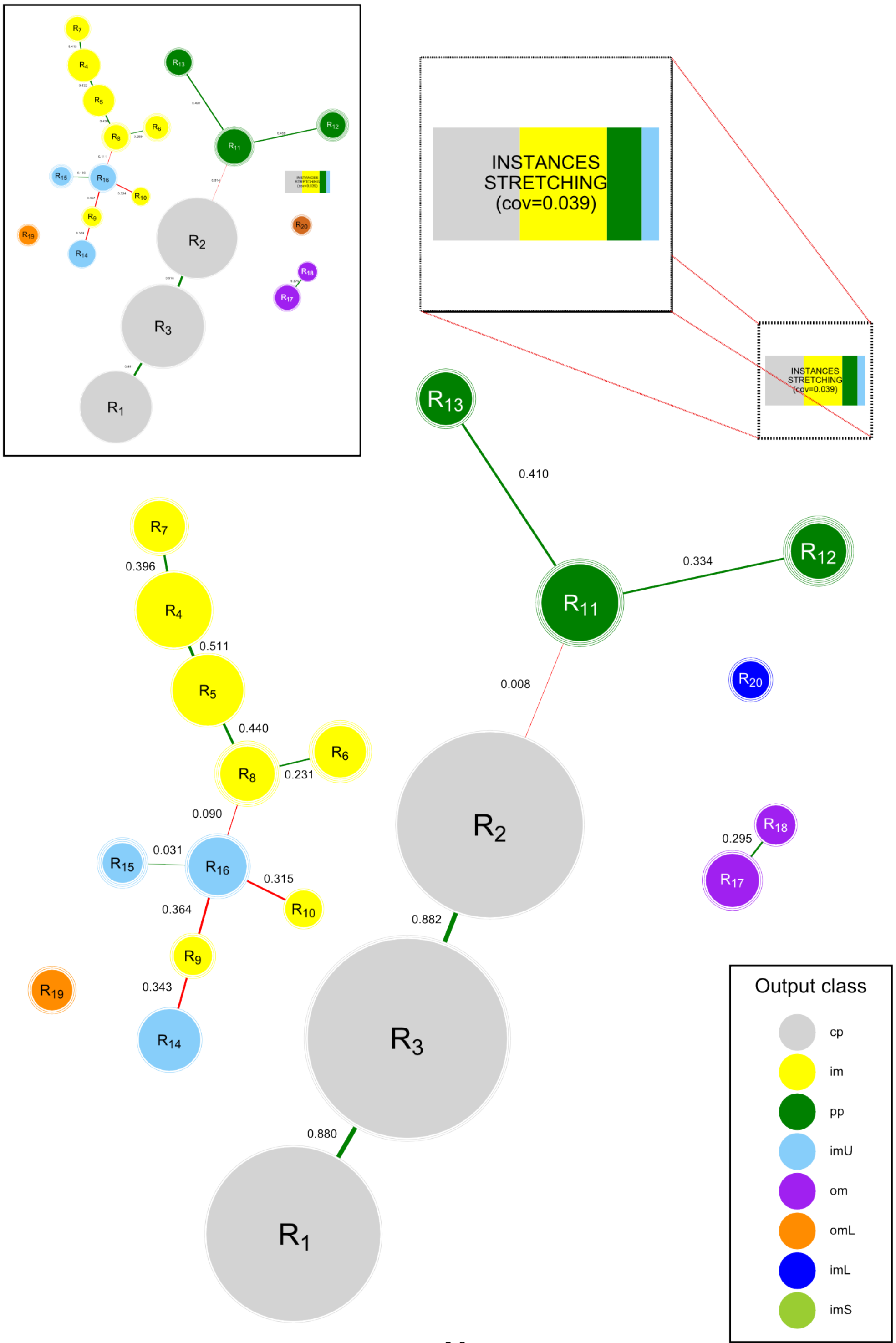


Figure 6: Fingram of the set of induced rules for Ecoli.

square we have included the same Fingram built with the original co-firing metric (as presented in Eq. 15) just for comparison purpose. In this case, both Fingrams present quite similar structure but weaker edges appear when using the PFM metric. Thus, we can analyze the system by studying any of them interchangeably. Therefore, we continue the analysis by regarding the Fingram built with the PFM metric.

The Fingram includes one node for each rule. These nodes take 7 different colors showing the 7 different classes the rules have as output. An additional multi-color node presents the instances not covered by any of the induced rules, i.e. the instances to be handled by the stretching mechanism.

Studying the structure of the Fingram in detail we can see that almost all the edges have low values meaning that rules cover few instances in common. Only rules of class *cp* cover several instances in common, showing high relations among them. This is because rules induced by FURIA typically cover instances scattered and as result most of the instances are just covered by a single rule. Moreover, it is easy to appreciate clusters of nodes of the same color because rules with different output classes are rarely related. This is also reflected by the majority of green links in the representation. This particularity indicates that rules of the same class jointly cover parts of the input space.

Focusing our attention in subsets of rules we observe that rules of class *cp* (rules R_1, R_2 and R_3), colored in gray, cover a large amount of instances (coverage larger than 0.3 as seen in Table A.4) with a good ratio of correctly classified instances (purity larger than 0.48). Even more, they cover several instances in common, as previously mentioned. Notice that *cp* is the majority class in the dataset. On the contrary, rules R_{19} and R_{20} cover just a few instances, and not in accordance with their corresponding class, as we can uncover by their purity equal to 0. It is worthy to note that both rules are in charge of handling two of the minority classes (*omL* and *imL*). In an intermediate situation we find rules with output classes *im* and *imU* (yellow and blue nodes) that only cover a few instances in common and have a diverse range of purity values. This situation occurs when instances of two classes are spread along the same part of the input space and turn difficult the classification task.

As previously mentioned, a non negligible part of the instances are fired by none of the 20 induced rules. This information is given by the node labeled as "INSTANCES STRETCHING" (zoomed in the top right square to appreciate its details) which shows that 13 instances (3.9% of the total) were not covered, and their corresponding class distribution is depicted in color sector areas (5 of class *cp*, 5 of *im*, 2 of *pp*, and 1 of *imU*). These instances trigger the stretching mechanism.

Fig. 7 presents Fingram representing the set of stretched rules. 18 rules were dynamically generated by the rule stretching mechanism (additional details are reported in Table A.5 in Appendix A). The new proposed PFM co-firing metric yields the principal Fingram in the figure whereas Fingram in the bottom left square was constructed with the original metric (Eq. 15). We observe that, in this case, the new metric is able to avoid subsets of highly connected nodes and it allows an easier analysis.

The set of stretched rules covers 5 classes (classes *cp*, *im*, *pp*, *imU*, and *om*) out of the 8 classes in the dataset. FURIA builds a high number of rules (18 rules) to deal with a few instances (13 instances). These rules are quite specific and they cover very few instances (rules $R_{1.1}$, $R_{3.1}$ or $R_{7.2}$ just one instance each) while each single instance is usually covered by several fuzzy rules. This particularity of the stretching mechanism occludes the interpretability of FURIA.

The structure of the Fingram of stretched rules is more complex than the previous. All the nodes are connected and the edges present higher values meaning that the rules densely cover the input space. Most of those edges correspond to inconsistencies, i.e. they relate rules with different output class.

The special node that is labeled as "UNCOVERED INSTANCES" shows that there are some instances which remain uncovered even after running the rule stretching mechanism. In such case, the inference mechanism produces as output the most frequent class, *cp*.

Finally, we study in detail the inference mechanism for a couple of instances. This way we show the benefits of considering instance-based Fingrams to locally view the FRBS inference mechanism. We graphically observe the rules that participate in the inference process, understanding the behavior of the system in a specific situation.

Fig. 8(a) presents the instance-based FURIA-Fingram for an instance covered by induced rules (instance $I_{321} = \{0.68, 0.67, 0.48, 0.5, 0.49, 0.4, 0.34, pp\}$). This instance only fires rules R_{11}, R_{12} and R_{13} with level of firing 0.80, 1.00 and 0.33 respectively as shown by the colored sectors in the picture. The three rules have the same output class *pp*, therefore the three related nodes have the same green color in the representation. The level of S_{c_l} (Eq. 20) for the given instance is presented in the bar chart of Fig. 8(b).

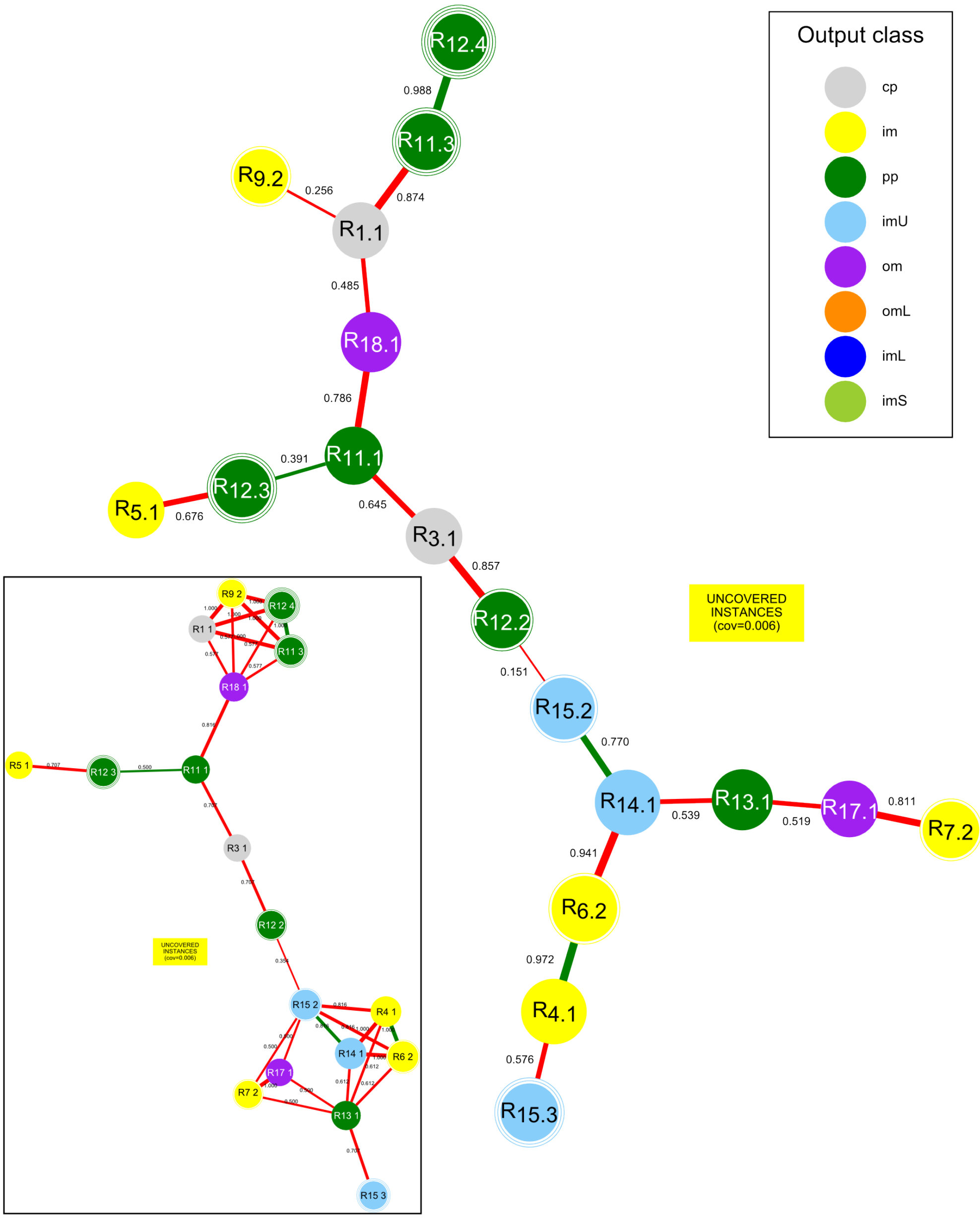
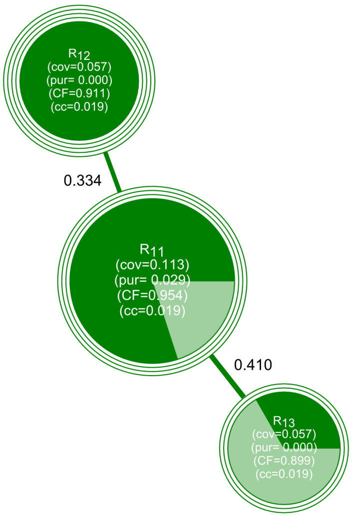
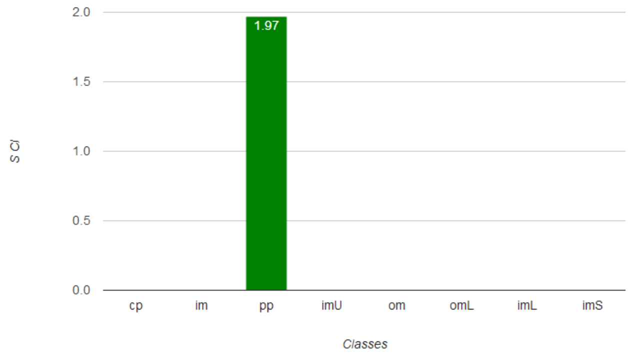


Figure 7: Fingram of the set of stretched rules.



(a) Instance-based FURIA-Fingram.



(b) Levels of S_{C_l}.

Figure 8: Analysis of inference for $I_{321} = \{0.68, 0.67, 0.48, 0.5, 0.49, 0.4, 0.34, pp\}$.

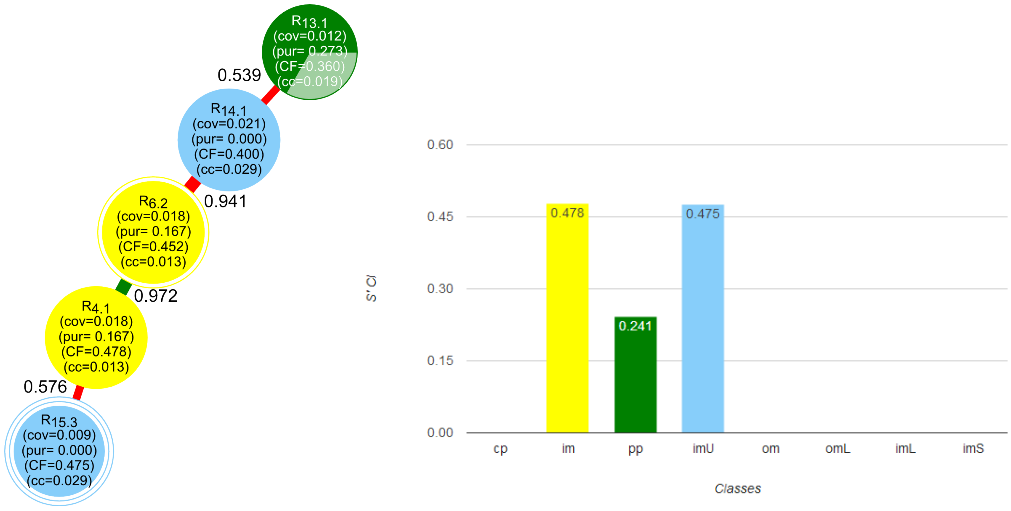
We clearly see that the system correctly infers class *pp*.

To conclude our analysis we will study a more complex case of inference. We selected an instance that is not covered by the set of induced rules (instance $I_{211} = \{0.69, 0.39, 0.48, 0.5, 0.57, 0.76, 0.79, im\}$). In consequence, the stretching mechanism is run as part of the inference process. I_{211} is handled by the stretched rules $R_{4.1}$, $R_{6.2}$, $R_{13.1}$, $R_{14.1}$ and $R_{15.3}$, as seen in Fig. 9(a). In this case, all but rules $R_{13.1}$ are fired to level 1 ($\mu_{R_{13.1}}(I_{211}) = 0.67$). The stretching mechanism correctly inferred class *im* because it is the winner class for S'_{c_l} (Eq. 23) as can be seen in Fig. 9(b). Anyway, we observe that $S'_{im} \approx S'_{imU}$ what is not desirable because it may produce ambiguity since a small variation in the input may incorrectly infer *imU* as output class.

5. Conclusions and Future Work

This paper has introduced the use of Fingrams for dealing with precise fuzzy systems. A complete study of PFM has been performed to adapt and produce the best visual representations, looking for enhancing the interpretability of such systems.

We adapt Fingrams to cope with precise fuzzy systems, obtaining a global



(a) Instance-based FURIA-Fingram.

(b) Levels of S'_{C_l}.

Figure 9: Analysis of inference for $I_{211} = \{0.69, 0.39, 0.48, 0.5, 0.57, 0.76, 0.79, im\}$.

view of the system. We take advantage of a previously published methodology and we develop new resources that fit PFM particularities:

- We have proposed a co-firing metric that reflects the particularities of PFM inference mechanisms.

- We have introduced a new visual artifact that represents instances not covered by a given FRBS. The detection and analysis of uncovered instances is key in fuzzy modeling because such instances directly penalize precision.

- We have developed a new visualization that gives us a local view of fuzzy systems, so called instance-based Fingrams. We focus our attention in the rules involved in the inference process of a single instance. This allows us to easily understand the behavior of fuzzy systems in specific situations. In consequence, we can analyze the system in detail, and even improve it with expert knowledge, carefully checking rule by rule and instance by instance.

We validated our proposal building Fingrams over FURIA, an outstanding PFM algorithm which is recognized because of its ability to construct accurate fuzzy classifiers but hardly to interpret. We studied over a bunch of benchmark datasets the difficulties FURIA inference mechanism presents and the opportunities Fingams offers to illuminate its inference mechanism.

To do so, we introduced adaptations to deal with FURIA particularities yielding the so-called FURIA-Fingrams. A twofold visualization allows us to comfortably visualize and analyze fuzzy systems learnt by FURIA.

Finally, we worked over a real world dataset where the potentials of the proposal are sketched and detailed. Fingrams have demonstrated very good behavior to show FURIA inference mechanism. Even more, their use in a high illustrative case of use makes possible an enriching discussion and analysis of FURIA advantages and drawbacks. The introduction of instance-based Fingrams allows the analysis of the FRBS behavior for key instances.

The future of Fingrams is very promising due to its benefits and possibilities. Its adaptation to other type of FRBSs, such as Takagi-Sugeno FRBSs, and its generalization to non-fuzzy rule based systems may spread its popularity. In addition, we will propose a generalized local view of the system, extending instance-based Fingrams, filtering the rules that are not fired by a set of instances, instead of a single instance.

Acknowledgment

We thank Eyke Hüllermeier for providing us with information and datasets used in [35] and his kind and quick response.

This work has been funded by the Spanish Ministry of Economy and Competitiveness under Grant TIN2014-56633-C3-3-R; and the European Centre for Soft Computing (ECSC) located at Mieres (Asturias, Spain).

[1] J. M. Alonso, L. Magdalena, Editorial: Special issue on interpretable fuzzy systems, Information Sciences 181 (2011) 4331–4339.

[2] J. Casillas, O. Cordón, F. Herrera, L. Magdalena (Eds.), Interpretability issues in fuzzy modeling, volume 128 of *Studies in Fuzziness and Soft Computing*, Springer-Verlag, Heidelberg, 2003.

[3] J. M. Alonso, C. Castiello, C. Mencar, Interpretability of fuzzy systems: Current research trends and prospects, in: J. Kacprzyk, W. Pedrycz (Eds.), Handbook of Computational Intelligence, Springer-Verlag, Heidelberg, 2014, pp. 181–199.

[4] L. A. Zadeh, The concept of a linguistic variable and its application to approximate reasoning, Parts I, II, and III. Information Sciences 8, 8, 9 (1975) 199–249, 301–357, 43–80.

[5] L. A. Zadeh, Outline of a new approach to the analysis of complex systems and decision processes, IEEE Transactions on Systems Man, and Cybernetics 3 (1973) 28–44.

[6] E. H. Mamdani, Application of fuzzy logic to approximate reasoning using linguistic synthesis, IEEE Transactions on Computers 26 (1977) 1182–1191.

[7] J. M. Alonso, L. Magdalena, G. González-Rodríguez, Looking for a good fuzzy system interpretability index: An experimental approach, International Journal of Approximate Reasoning 51 (2009) 115–134.

[8] M. J. Gacto, R. Alcalá, F. Herrera, Interpretability of linguistic fuzzy rule-based systems: An overview of interpretability measures, Information Sciences 181 (2011) 4340–4360.

[9] S.-M. Zhou, J. Q. Gan, Low-level interpretability and high-level interpretability: a unified view of data-driven interpretable fuzzy system modelling, Fuzzy Sets and Systems 159 (2008) 3091–3131.

[10] C. Mencar, A. M. Fanelli, Interpretability constraints for fuzzy information granulation, Information Sciences 178 (2008) 4585–4618.

[11] E. Trillas, S. Guadarrama, Fuzzy representations need a careful design, International Journal of General Systems 39 (2010) 329–346.

[12] R. Mikut, J. Jäkel, L. Gröll, Interpretability issues in data-based learning of fuzzy systems, Fuzzy Sets and Systems 150 (2005) 179–197.

[13] W. Pedrycz, Fuzzy modelling: fundamentals, construction and evaluation, Fuzzy Sets and Systems 41 (1991) 1–15.

[14] R. Alcalá, J. Alcalá-Fdez, J. Casillas, O. Cordón, F. Herrera, Hybrid learning models to get the interpretability–accuracy trade-off in fuzzy modeling, Soft Computing 10 (2006) 717–734.

[15] D. Dubois, H. Prade, The three semantics of fuzzy sets, Fuzzy sets and systems 90 (1997) 141–150.

[16] D. Dubois, H. Prade, Gradualness, uncertainty and bipolarity: Making sense of fuzzy sets, Fuzzy Sets and Systems 192 (2012) 3–24.

[17] J. Casillas, F. Herrera, O. Cordón, L. Magdalena, Accuracy improvements in linguistic fuzzy modeling, volume 129 of *Studies in Fuzziness and Soft Computing*, Springer, 2003.

[18] J. Casillas, F. Herrera, O. Cordón, L. Magdalena, Interpretability issues in fuzzy modeling, volume 128 of *Studies in Fuzziness and Soft Computing*, Springer, 2003.

[19] L. A. Zadeh, Toward a theory of fuzzy information granulation and its centrality in human reasoning and fuzzy logic, Fuzzy sets and systems 90 (1997) 111–127.

[20] T. Takagi, M. Sugeno, Fuzzy identification of systems and its applications to modeling and control, IEEE Transactions on Systems, Man and Cybernetics (1985) 116–132.

[21] S.-M. Zhou, J. Q. Gan, Extracting takagi-sugeno fuzzy rules with interpretable submodels via regularization of linguistic modifiers, IEEE Transactions on Knowledge and Data Engineering 21 (2009) 1191–1204.

[22] S.-M. Zhou, R. A. Lyons, S. Brophy, M. B. Gravenor, Constructing compact takagi-sugeno rule systems: identification of complex interactions in epidemiological data, PLoS ONE 7 (2012) e51468.

[23] D. Nauck, R. Kruse, How the learning of rule weights affects the interpretability of fuzzy systems, in: IEEE International Conference on Fuzzy Systems (FUZZ-IEEE1998), volume 2, pp. 1235–1240.

[24] H. Ishibuchi, T. Nakashima, Effect of rule weights in fuzzy rule-based classification systems, IEEE Transactions on Fuzzy Systems 9 (2001) 506–515.

[25] H. Ishibuchi, T. Yamamoto, Rule weight specification in fuzzy rule-based classification systems, IEEE Transactions on Fuzzy Systems 13 (2005) 428–435.

[26] B. Pham, R. Brown, Visualisation of fuzzy systems: requirements, techniques and framework, Future Generation Computer Systems 21 (2005) 1199–1212.

[27] T. R. Gabriel, K. Thiel, M. R. Berthold, Rule visualization based on multi-dimensional scaling, in: IEEE Interational Conference on Fuzzy Systems (FUZZ-IEEE2006), pp. 66–71.

[28] H. Ishibuchi, Y. Kaisho, Y. Nojima, Designing fuzzy rule-based classifiers that can visually explain their classification results to human users, in: International Workshop on Genetic and Evolving Fuzzy Systems (GEFS2008), pp. 5–10.

[29] A. G. Evsukoff, S. Galichet, B. S. L. P. de Lima, N. F. F. Ebecken, Design of interpretable fuzzy rule-based classifiers using spectral analysis with structure and parameters optimization, Fuzzy Sets and Systems 160 (2009) 857–881.

[30] A. R. Buck, J. M. Keller, Visualizing uncertainty with fuzzy rose diagrams, in: IEEE Symposium Series on Computational Intelligence (IEEE SSCI 2014).

[31] M. Berthold, L. Hall, Visualizing fuzzy points in parallel coordinates, IEEE Transactions on Fuzzy Systems 11 (2003) 369–374.

[32] S. Henzgen, M. Strickert, E. Hüllermeier, Visualization of evolving fuzzy rule-based systems, Evolving Systems 5 (2014) 175–191.

[33] D. P. Pancho, J. M. Alonso, O. Cordón, A. Quirin, L. Magdalena, FINGRAMS: visual representations of fuzzy rule-based inference for expert analysis of comprehensibility, IEEE Transactions on Fuzzy Systems 21 (2013) 1133–1149.

[34] D. P. Pancho, J. M. Alonso, J. Alcalá-Fdez, L. Magdalena, Interpretability analysis of fuzzy association rules supported by fingrams, in: Conference of the European Society for Fuzzy Logic and Technology (EUSFLAT2013), Atlantis Press, pp. 469–474.

[35] J. Hühn, E. Hüllermeier, FURIA: an algorithm for unordered fuzzy rule induction, Data Mining and Knowledge Discovery 19 (2009) 293–319.

[36] R. Cannone, J. M. Alonso, L. Magdalena, Multi-objective design of highly interpretable fuzzy rule-based classifiers with semantic cointension, in: IEEE International Workshop on Genetic and Evolutionary Fuzzy Systems (GEFS2011), pp. 1–8.

[37] J.-S. Jang, ANFIS: adaptive-network-based fuzzy inference system, IEEE Transactions on Systems, Man and Cybernetics 23 (1993) 665–685.

[38] A. González, R. Pérez, SLAVE: A genetic learning system based on an iterative approach, IEEE Transactions on Fuzzy Systems 7 (1999) 176–191.

[39] R. W. Schvaneveldt, F. T. Durso, D. W. Dearholt, Network structures in proximity data, The psychology of learning and motivation: Advances in research and theory 24 (1989) 249–284.

[40] S. G. Kobourov, Force-directed drawing algorithms, in: R. Tamassia (Ed.), Handbook of Graph Drawing and Visualization, CRC Press, 2012.

[41] T. Kamada, S. Kawai, An algorithm for drawing general undirected graphs, Information Processing Letters 31 (1989) 7–15.

[42] D. P. Pancho, J. M. Alonso, J. Alcalá-Fdez, A new fingram-based software tool for visual representation and analysis of fuzzy association rules, in: IEEE International Conference on Fuzzy Systems (FUZZ-IEEE2013), pp. 1–7.

[43] D. P. Pancho, J. M. Alonso, J. Alcalá-Fdez, L. Magdalena, Analyzing fuzzy association rules with fingrams in KEEL, in: IEEE International Conference on Fuzzy Systems (FUZZ-IEEE2014), pp. 1–8.

[44] D. P. Pancho, J. M. Alonso, L. Magdalena, T. Kötter, M. R. Berthold, Fingrams KNIME plugin for visual analysis of fuzzy rule-based classifiers, in: World Conference on Soft Computing (WConSC2014).

[45] D. P. Pancho, J. M. Alonso, L. Magdalena, Quest for interpretability-accuracy trade-off supported by fingrams into the fuzzy modeling tool GUAJE, International Journal of Computational Intelligence Systems 6 (2013) 46–60.

[46] J. Alcala-Fdez, J. M. Alonso, A survey of fuzzy systems software: Taxonomy, current research trends and prospects, IEEE Transactions on Fuzzy Systems (2015).

[47] W. N. Venables, B. D. Ripley, Modern applied statistics with S-PLUS, Springer, 1999.

[48] M. M. Rahman, D. N. Davis, Machine learning-based missing value imputation method for clinical datasets, in: IAENG Transactions on Engineering Technologies, volume 229 of *Lecture Notes in Electrical Engineering*, Springer, 2013, pp. 245–257.

[49] J. P. Lucas, A. Laurent, M. N. Moreno, M. Teisseire, A fuzzy associative classification approach for recommender systems, International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems 20 (2012) 579–617.

[50] O. Cordón, K. Trawiński, A novel framework to design fuzzy rule-based ensembles using diversity induction and evolutionary algorithms-based classifier selection and fusion, in: Advances in Computational Intelligence, Springer, 2013, pp. 36–58.

[51] K. Trawiński, O. Cordón, A. Quirin, On designing fuzzy rule-based multiclassification systems by combining FURIA with bagging and feature selection, International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems 19 (2011) 589–633.

[52] W. Cohen, Fast effective rule induction, in: International Conference on Machine Learning (ICML1995), pp. 115–123.

[53] J. C. Hühn, E. Hüllermeier, An analysis of the FURIA algorithm for fuzzy rule induction, in: Advances in Machine Learning I, Springer, 2010, pp. 321–344.

[54] P. Horton, K. Nakai, A probabilistic classification system for predicting the cellular localization sites of proteins., in: International Conference on Intelligent Systems for Molecular Biology (ISMB1996), pp. 109–115.

[55] K. Bache, M. Lichman, UCI machine learning repository, 2013.

Appendix A. Information of the set of rules in subsection 4.4

This appendix includes information about the sets of induced and stretched rules FURIA builds up for the dataset *ecoli* as presented in subsection 4.4 of this manuscript.

Table A.4 summarizes the information of the set of rules induced by FURIA –rule identifiers R_i, rule output class B^i (as presented in Eq. 4), coverage of rules cov_i (Eq. 11), purity of rules pur_i (Eq. 18), certain factor CF_{R_i} (Eq. 21), and coverage of rule output class cc_i (Eq. 12).–

R_i	B^i	cov_i	pur_i	CF_{R_i}	cc_i
R_1	cp	0.339	0.535	0.973	0.427
R_2	cp	0.393	0.485	0.951	0.469
R_3	cp	0.405	0.502	0.955	0.476
R_4	im	0.098	0.030	0.956	0.013
R_5	im	0.092	0.136	0.951	0.052
R_6	im	0.042	0.071	0.904	0.013
R_7	im	0.045	0.000	0.902	0.013
R_8	im	0.051	0.113	0.916	0.026
R_9	im	0.012	0.000	0.692	0.013
R_{10}	im	0.009	0.000	0.615	0.013
R_{11}	pp	0.113	0.029	0.954	0.019
R_{12}	pp	0.057	0.000	0.911	0.019
R_{13}	pp	0.057	0.000	0.899	0.019
R_{14}	imU	0.065	0.182	0.800	0.114
R_{15}	imU	0.015	0.941	0.713	0.114
R_{16}	imU	0.057	0.053	0.581	0.029
R_{17}	om	0.051	0.000	0.891	0.050
R_{18}	om	0.015	0.000	0.687	0.050
R_{19}	omL	0.018	0.000	0.719	0.200
R_{20}	imL	0.009	0.000	0.503	0.500
Instances Stretching	cp/im/pp/imU	0.039			

Table A.4: Information of induced rules of the case of use in subsection 4.4. R_i is the rule identifier, B^i is the rule output class, cov_i is the coverage, pur_i is the purity, CF_{R_i} is the certain factor, and cc_i is the coverage of the rule output class.

Table A.5 shows the information of stretched rules that produce the Fingram of Fig. 7 –rule identifiers $R_i.q$ where R_i is the original rule from the rule derives and q the number of antecedents kept, Antecedents shows the rule antecedents of each rule, rule output class B^i (as presented in Eq. 4), coverage of rules $cov_{i.q}$ (Eq. 11), purity of rules $pur_{i.q}$ (Eq. 18), certain factor $CF'_{R_{i.q}}$ (Eq. 25), and coverage of rule output class $cc_{i.q}$ (Eq. 12).–

$R_i.q$	Antecedents	B^i	$cov_{i.q}$	$pur_{i.q}$	$CF'_{R_{i.q}}$	$cc_{i.q}$
$R_{1.1}$	alm1 in $[-\infty, -\infty, 0.38, 0.39]$	cp	0.006	0.000	0.486	0.007
$R_{3.1}$	alm1 in $[-\infty, -\infty, 0.47, 0.49]$	cp	0.006	0.000	0.382	0.007
$R_{4.1}$	alm1 in $[0.75, 0.76, \infty, \infty]$	im	0.018	0.167	0.478	0.013
$R_{5.1}$	alm1 in $[0.55, 0.61, \infty, \infty]$	im	0.006	1.000	0.475	0.013
$R_{6.2}$	alm2 in $[0.59, 0.63, \infty, \infty]$ & mcg in $[-\infty, -\infty, 0.74, 0.79]$	im	0.018	0.167	0.452	0.013
$R_{7.2}$	alm1 in $[0.82, 0.85, \infty, \infty]$ & mcg in $[-\infty, -\infty, 0.74, 0.86]$	im	0.006	1.000	0.541	0.013
$R_{9.2}$	alm2 in $[0.35, 0.74, \infty, \infty]$ & alm1 in $[-\infty, -\infty, 0.72, 0.73]$	im	0.006	0.000	0.415	0.013
$R_{11.3}$	gvh in $[0.58, 0.59, \infty, \infty]$ & aac in $[-\infty, -\infty, 0.47, 0.57]$ & alm1 in $[-\infty, -\infty, 0.65, 0.67]$	pp	0.006	1.000	0.636	0.019
$R_{11.1}$	gvh in $[0.58, 0.59, \infty, \infty]$	pp	0.009	0.000	0.318	0.019
$R_{12.4}$	gvh in $[0.53, 0.56, \infty, \infty]$ & mcg in $[0.61, 0.63, \infty, \infty]$ & aac in $[-\infty, -\infty, 0.63, 0.65]$ & alm1 in $[-\infty, -\infty, 0.52, 0.53]$	pp	0.006	1.000	0.651	0.019
$R_{12.3}$	gvh in $[0.53, 0.56, \infty, \infty]$ & mcg in $[0.61, 0.63, \infty, \infty]$ & aac in $[-\infty, -\infty, 0.63, 0.65]$	pp	0.009	0.000	0.520	0.019
$R_{12.2}$	gvh in $[0.53, 0.56, \infty, \infty]$ & mcg in $[0.61, 0.63, \infty, \infty]$	pp	0.009	0.000	0.390	0.019
$R_{13.1}$	mcg in $[0.67, 0.7, \infty, \infty]$	pp	0.012	0.273	0.360	0.019
$R_{14.1}$	alm2 in $[0.39, 0.62, \infty, \infty]$	imU	0.021	0.000	0.400	0.029
$R_{15.3}$	alm2 in $[0.46, 0.66, \infty, \infty]$ & mcg in $[0.58, 0.62, \infty, \infty]$ & gvh in $[-\infty, -\infty, 0.45, 0.46]$	imU	0.009	0.000	0.475	0.029
$R_{15.2}$	alm2 in $[0.46, 0.66, \infty, \infty]$ & mcg in $[0.58, 0.62, \infty, \infty]$	imU	0.015	0.000	0.356	0.029
$R_{17.1}$	aac in $[0.66, 0.68, \infty, \infty]$	om	0.006	0.000	0.356	0.050
$R_{18.1}$	gvh in $[0.67, 0.68, \infty, \infty]$	om	0.012	0.000	0.343	0.050
Uncovered instances		im	0.006			

Table A.5: Information of stretched rules of the case of use in subsection 4.4. $R_{i.q}$ is the rule identifier, Antecedents shows rule antecedents, B^i is the rule output class, $cov_{i.q}$ is the coverage, $pur_{i.q}$ is the purity, $CF'_{R_{i.q}}$ is the certain factor, and $cc_{i.q}$ is the coverage of the rule output class.

A.1.6 JCR of Fuzzy Sets and Systems

Journal Citation Reports®

2013 JCR Science Edition

Journal: FUZZY SETS AND SYSTEMS

Journal Impact Factor

Cites in 2013 to items published in:	2012 = 262	Number of items published in:	2012 = 171
	2011 = 381		2011 = 171
	Sum: 643		Sum: 342

Calculation: Cites to recent items / Number of recent items = 643 / 342 = **1.880**

Journal Ranking

For **2013**, the journal **FUZZY SETS AND SYSTEMS** has an Impact Factor of **1.880**.

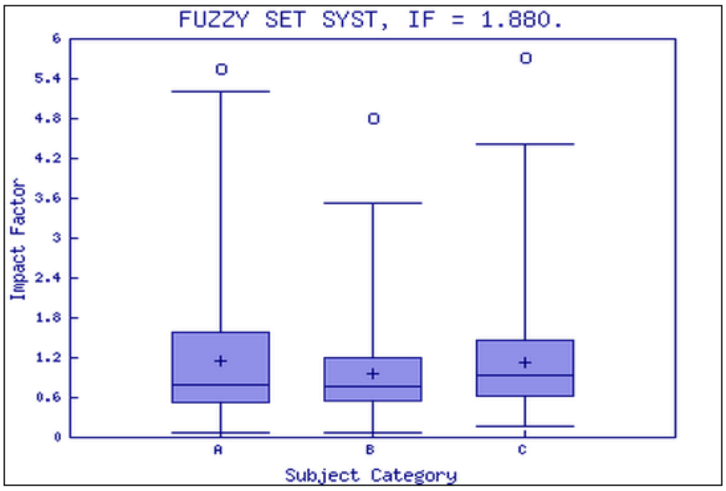
This table shows the ranking of this journal in its subject categories based on Impact Factor.

Category Name	Total Journals in Category	Journal Rank in Category	Quartile in Category
COMPUTER SCIENCE, THEORY & METHODS	102	16	Q1
MATHEMATICS, APPLIED	251	20	Q1
STATISTICS & PROBABILITY	119	14	Q1

Category Box Plot

For **2013**, the journal **FUZZY SETS AND SYSTEMS** has an Impact Factor of **1.880**.

This is a box plot of the subject category or categories to which the journal has been assigned. It provides information about the distribution of journals based on Impact Factor values. It shows median, 25th and 75th percentiles, and the extreme values of the distribution.



Key

A - COMPUTER SCIENCE, THEORY & METHODS

B - MATHEMATICS, APPLIED

C - STATISTICS & PROBABILITY

A.2 Aditional publications

Here we join up other publications published during the thesis period and related with the topic of the thesis. Those are included in books and conference proceedings and ordered chronologically. We present the citation information and the number of citations of each publication[2].

A.2.1 Book chapters

[12] Alonso, J. M., Pancho, D. P., Cordón, O., Quirin, A., and Magdalena, L. Social network analysis of co-fired fuzzy rules. In *Soft Computing: State of the Art Theory and Novel Applications*, R. R. Yager, A. M. Abbasov, M. Reformat, and S. N. Shahbazova, Eds. Springer, 2013, pp. 113–128

Citations Google Scholar:	6
Citations Web of Science:	-

A.2.2 Conference proceedings

[13] Alonso, J. M., Pancho, D. P., and Magdalena, L. Enhancing the fuzzy modeling tool GUAJE with a new module for Fingrams-based analysis of fuzzy rule bases. In *IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)* (2012), pp. 1082–1089

Citations Google Scholar:	4
Citations Web of Science:	0

[11] Alonso, J. M., Pancho, D. P., Cordón, O., Quirin, A., and Magdalena, L. Fingrams: Una nueva herramienta para análisis visual de sistemas fuzzy. In *XVI Congreso Español sobre Tecnologías y Lógica Fuzzy (ESTYLF)* (2012), pp. 585–590

Citations Google Scholar:	0
Citations Web of Science:	-

[62] Pancho, D. P., Alonso, J. M., and Alcalá-Fdez, J. A new Fingram-based software tool for visual representation and analysis of fuzzy association rules. In *IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)* (2013), pp. 1–7

Citations Google Scholar:	5
Citations Web of Science:	0

[2] We include information of Google Scholar and Web of Science retrieved on August the 13th 2015. We indicate with a – when we do not obtain any information from the mentioned web services.

[63] Pancho, D. P., Alonso, J. M., Alcalá-Fdez, J., and Magdalena, L. Interpretability analysis of fuzzy association rules supported by Fingrams. In *Conference of the European Society for Fuzzy Logic and Technology (EUSFLAT)* (2013), Atlantis Press, pp. 469–474

Citations Google Scholar:	8
Citations Web of Science:	1

[66] Pancho, D. P., Alonso, J. M., Kötter, T., Berthold, M. R., and Magdalena, L. Analyzing fuzzy rule-based systems with Fingrams in KNIME. In *XVII Congreso Español sobre Tecnologías y Lógica Fuzzy (ESTYLF)* (2014), pp. 597–602

Citations Google Scholar:	-
Citations Web of Science:	-

[64] Pancho, D. P., Alonso, J. M., Alcalá-Fdez, J., and Magdalena, L. Analyzing fuzzy association rules with Fingrams in KEEL. In *IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)* (2014), pp. 1–8

Citations Google Scholar:	3
Citations Web of Science:	0

[69] Pancho, D. P., Alonso, J. M., Magdalena, L., Kötter, T., and Berthold, M. R. Fingrams KNIME plugin for visual analysis of fuzzy rule-based classifiers. In *World Conference on Soft Computing (WConSC)* (2014)

Citations Google Scholar:	2
Citations Web of Science:	-

[28] D. P. Pancho, J. M. Alonso, L. M. Understanding the inference mechanism of FURIA by means of Fingrams. In *Conference of the European Society for Fuzzy Logic and Technology (EUSFLAT)* (2015), pp. 297–304

Citations Google Scholar:	0
Citations Web of Science:	-

Bibliography

[1] Agrawal, R., Imielinski, T., and Swami, A. Mining association rules between sets of items in large databases. In *ACM Conference of the Special Interest Group on Management of Data (SIGMOD)* (1993), pp. 207–216.

[2] Alcalá, R., Alcalá-Fdez, J., and Herrera, F. A proposal for the genetic lateral tuning of linguistic fuzzy systems and its interaction with rule selection. *IEEE Transactions on Fuzzy Systems 15*, 4 (2007), 616–635.

[3] Alcalá-Fdez, J., Alcalá, R., Gacto, M. J., and Herrera, F. Learning the membership function contexts for mining fuzzy association rules by using genetic algorithms. *Fuzzy Sets and Systems 160*, 7 (2009), 905–921.

[4] Alcalá-Fdez, J., Sánchez, L., Garcia, S., del Jesus, M., Ventura, S., Garrell, J., Otero, J., Romero, C., Bacardit, J., Rivas, V., et al. KEEL: A software tool to assess evolutionary algorithms for data mining problems. *Soft Computing 13*, 3 (2009), 307–318.

[5] Alcalá-Fdez et al., J. KEEL data-mining software tool: Data set repository, integration of algorithms and experimental analysis framework. *Journal of Mult.-Valued Logic & Soft Computing 17* (2011), 255–287.

[6] Alonso, J. M., Castiello, C., Lucarelli, M., and Mencar, C. Modelling interpretable fuzzy rule-based classifiers for medical decision support. In *Medical Applications of Intelligent Data Analysis: Research advancements*. IGI Global, 2012, pp. 255–272.

[7] Alonso, J. M., Castiello, C., and Mencar, C. Interpretability of fuzzy systems: Current research trends and prospects. In *Springer Handbook of Computational Intelligence*, J. Kacprzyk and W. Pedrycz, Eds. Springer Berlin Heidelberg, 2015, pp. 219–237.

[8] Alonso, J. M., and Magdalena, L. Editorial: Special issue on interpretable fuzzy systems. *Information Sciences 181*, 20 (2011), 4331–4339.

[9] Alonso, J. M., Magdalena, L., and Cordón, O. Embedding HILK in a three-objective evolutionary algorithm with the aim of modeling highly interpretable fuzzy

rule-based classifiers. In *International Workshop on Genetic and Evolutionary Fuzzy Systems (GEFS)* (2010), IEEE, pp. 15–20.

[10] Alonso, J. M., Magdalena, L., and González-Rodríguez, G. Looking for a good fuzzy system interpretability index: An experimental approach. *International Journal of Approximate Reasoning 51* (2009), 115–134.

[11] Alonso, J. M., Pancho, D. P., Cordón, O., Quirin, A., and Magdalena, L. Fingrams: Una nueva herramienta para análisis visual de sistemas fuzzy. In *XVI Congreso Español sobre Tecnologías y Lógica Fuzzy (ESTYLF)* (2012), pp. 585–590.

[12] Alonso, J. M., Pancho, D. P., Cordón, O., Quirin, A., and Magdalena, L. Social network analysis of co-fired fuzzy rules. In *Soft Computing: State of the Art Theory and Novel Applications*, R. R. Yager, A. M. Abbasov, M. Reformat, and S. N. Shahbazova, Eds. Springer, 2013, pp. 113–128.

[13] Alonso, J. M., Pancho, D. P., and Magdalena, L. Enhancing the fuzzy modeling tool GUAJE with a new module for Fingrams-based analysis of fuzzy rule bases. In *IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)* (2012), pp. 1082–1089.

[14] Bache, K., and Lichman, M. UCI machine learning repository, 2013.

[15] Berthold, M., and Hall, L. Visualizing fuzzy points in parallel coordinates. *IEEE Transactions on Fuzzy Systems 11*, 3 (2003), 369–374.

[16] Berthold, M. R., Cebron, N., Dill, F., Gabriel, T. R., Kötter, T., Meinl, T., Ohl, P., Sieb, C., Thiel, K., and Wiswedel, B. KNIME: The Konstanz Information Miner. In *Data Analysis, Machine Learning and Applications*, Studies in Classification, Data Analysis, and Knowledge Organization. Springer, 2008, pp. 319–326.

[17] Buck, A. R., and Keller, J. M. Visualizing uncertainty with fuzzy rose diagrams. In *IEEE Symposium Series on Computational Intelligence (IEEE SSCI)* (2014), pp. 30–36.

[18] Cannone, R., Alonso, J. M., and Magdalena, L. Multi-objective design of highly interpretable fuzzy rule-based classifiers with semantic cointension. In *IEEE International Workshop on Genetic and Evolutionary Fuzzy Systems (GEFS)* (2011), pp. 1–8.

[19] Casillas, J., Cordón, O., Herrera, F., and Magdalena, L., Eds. *Interpretability issues in fuzzy modeling*, vol. 128 of *Studies in Fuzziness and Soft Computing*. Springer, 2003.

[20] Casillas, J., Herrera, F., Cordón, O., and Magdalena, L. *Accuracy improvements in linguistic fuzzy modeling*, vol. 129 of *Studies in Fuzziness and Soft Computing*. Springer, 2003.

[21] Casillas, J., and Martínez-López, F. Mining uncertain data with multiobjective genetic fuzzy systems to be applied in consumer behaviour modelling. *Expert Systems with Applications 36*, 2 (2009), 1645–1659.

[22] Chang, Y.-H. O., and Ayyub, B. M. Fuzzy regression methods–a comparative assessment. *Fuzzy Sets and Systems 119*, 2 (2001), 187–203.

[23] Chen, C., and Morris, S. Visualizing evolving networks: Minimum spanning trees versus pathfinder networks. In *IEEE Symposium on Information Visualization* (2003), pp. 67–74.

[24] Cheong, F., and Lai, R. Constraining the optimization of a fuzzy logic controller using an enhanced genetic algorithm. *IEEE Transactions on Systems Man, and Cybernetics, Part B: Cybernetics 30*, 1 (2000), 31–46.

[25] Cohen, W. Fast effective rule induction. In *International Conference on Machine Learning (ICML)* (1995), pp. 115–123.

[26] Cordón, O., Herrera, F., Magdalena, L., and Villar, P. A genetic learning process for the scaling factors, granularity and contexts of the fuzzy rule-based system data base. *Information Sciences 136*, 1–4 (2001), 85–107.

[27] Cordón, O., Herrera, F., and Sánchez, L. Solving electrical distribution problems using hybrid evolutionary data analysis techniques. *Applied Intelligence 10*, 1 (1999), 5–24.

[28] D. P. Pancho, J. M. Alonso, L. M. Understanding the inference mechanism of FURIA by means of Fingrams. In *Conference of the European Society for Fuzzy Logic and Technology (EUSFLAT)* (2015), pp. 297–304.

[29] De Oliveira, J. V. Semantic constraints for membership function optimization. *IEEE Transactions on Systems, Man and Cybernetics, Part A: Systems and Humans 29*, 1 (1999), 128–138.

[30] Dearholt, D. W., and Schvaneveldt, R. W. Properties of pathfinder networks. In *Pathfinder associative networks: Studies in knowledge organization*, R. Schvaneveldt, Ed. Ablex Publishing Corporation, 1990, pp. 1–30.

[31] di Battista, G., Eades, P., Tamassia, R., and Tollis, I. *Graph drawing: algorithms for the visualization of graphs.* Prentice Hall, Upper Saddle River, N.J, 1998.

[32] Dubois, D., and Prade, H. What are fuzzy rules and how to use them. *Fuzzy Sets and Systems 84*, 2 (1996), 169–185.

[33] Dubois, D., Prade, H., and Ughetto, L. A new perspective on reasoning with fuzzy rules. *International Journal of Intelligent Systems 18*, 5 (2003), 541–567.

[34] Evsukoff, A. G., Branco, A., and Galichet, S. Intelligent data analysis and model interpretation with spectral analysis fuzzy symbolic modeling. *International Journal of Approximate Reasoning 52*, 6 (2011), 728–750.

[35] Evsukoff, A. G., Galichet, S., de Lima, B. S. L. P., and Ebecken, N. F. F. Design of interpretable fuzzy rule-based classifiers using spectral analysis with structure and parameters optimization. *Fuzzy Sets and Systems 160*, 7 (2009), 857–881.

[36] Fazendeiro, P., de Oliveira, J. V., and Pedrycz, W. A multiobjective design of a patient and anaesthetist-friendly neuromuscular blockade controller. *IEEE Transactions on Biomedical Engineering 54*, 9 (2007), 1667–1678.

[37] Fruchterman, T. M. J., and Reingold, E. M. Graph drawing by force-directed placement. *Software: Practice and experience 21*, 11 (1991), 1129–1164.

[38] Gabriel, T. R., Thiel, K., and Berthold, M. R. Rule visualization based on multi-dimensional scaling. In *IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)* (2006), pp. 66–71.

[39] Gacto, M. J., Alcalá, R., and Herrera, F. Integration of an index to preserve the semantic interpretability in the multiobjective evolutionary rule selection and tuning of linguistic fuzzy systems. *IEEE Transactions on Fuzzy Systems 18*, 3 (2010), 515–531.

[40] Gacto, M. J., Alcalá, R., and Herrera, F. Interpretability of linguistic fuzzy rule-based systems: An overview of interpretability measures. *Information Sciences 181*, 20 (2011), 4340–4360.

[41] Gansner, E. R., and North, S. C. An open graph visualization system and its applications to software engineering. *Software Practice and Experience 30*, 11 (2000), 1203–1233.

[42] Glorennec, P. Y. *Algorithmes d'apprentissage pour systèmes d'inférence floue.* Hermès, 1999.

[43] Henzgen, S., Strickert, M., and Hüllermeier, E. Visualization of evolving fuzzy rule-based systems. *Evolving Systems 5*, 3 (may 2014), 175–191.

[44] Horton, P., and Nakai, K. A probabilistic classification system for predicting the cellular localization sites of proteins. In *International Conference on Intelligent Systems for Molecular Biology (ISMB)* (1996), pp. 109–115.

[45] Hühn, J., and Hüllermeier, E. FURIA: an algorithm for unordered fuzzy rule induction. *Data Mining and Knowledge Discovery 19*, 3 (2009), 293–319.

[46] Hühn, J. C., and Hüllermeier, E. An analysis of the FURIA algorithm for fuzzy rule induction. In *Advances in Machine Learning I*. Springer, 2010, pp. 321–344.

[47] Ishibuchi, H., Kaisho, Y., and Nojima, Y. Designing fuzzy rule-based classifiers that can visually explain their classification results to human users. In *International Workshop on Genetic and Evolving Fuzzy Systems (GEFS)* (2008), pp. 5–10.

[48] Ishibuchi, H., Kaisho, Y., and Nojima, Y. A visual explanation system for explaining fuzzy reasoning results by fuzzy rule-based classifiers. In *NAFIPS* (2008), pp. 1–6.

[49] Ishibuchi, H., Kaisho, Y., and Nojima, Y. Complexity, interpretability and explanation capability of fuzzy rule-based classifiers. In *IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)* (2009), pp. 1730–1735.

[50] Jones, H., Charnomordic, B., Dubois, D., and Guillaume, S. Practical inference with systems of gradual implicative rules. *IEEE Transactions on Fuzzy Systems 17*, 1 (2009), 61–78.

[51] Kamada, T., and Kawai, S. An algorithm for drawing general undirected graphs. *Information Processing Letters 31*, 1 (1989), 7–15.

[52] Kobourov, S. G. Force-directed drawing algorithms. In *Handbook of Graph Drawing and Visualization*, R. Tamassia, Ed. CRC Press, 2012, ch. 12.

[53] Kuncheva, L. I. *Fuzzy classifier design*, vol. 49. Springer Science & Business Media, 2000.

[54] Mamdani, E. H. Application of fuzzy logic to approximate reasoning using linguistic synthesis. *IEEE Transactions on Computers 26*, 12 (1977), 1182–1191.

[55] Márquez, A., Márquez, F., and Peregrín, A. A multi-objective evolutionary algorithm with an interpretability improvement mechanism for linguistic fuzzy systems with adaptive defuzzification. In *IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)* (2010), pp. 277–283.

[56] Mencar, C., Castiello, C., Cannone, R., and Fanelli, A. M. Interpretability assessment of fuzzy knowledge bases: a cointension based approach. *International Journal of Approximate Reasoning 52*, 4 (2011), 501–518.

[57] Mikut, R., Jäkel, J., and Gröll, L. Interpretability issues in data-based learning of fuzzy systems. *Fuzzy Sets and Systems 150*, 2 (2005), 179–197.

[58] Moya-Anegón, F., Vargas-Quesada, B., Chinchilla-Rodríguez, Z., Corera-Álvarez, E., Muñoz-Fernández, F. J., and Herrero-Solana, V. Visualizing the marrow of science. *Journal of the American Society for Information Science and Technology 58*, 14 (2007), 2167–2179.

[59] Moya-Anegón, F., Vargas-Quesada, B., Herrero-Solana, V., Chinchilla-Rodríguez, Z., Corera-Álvarez, E., and Muñoz-Fernández, F. J. A new technique for building maps of large scientific domains based on the cocitation of classes and categories. *Scientometrics 61*, 1 (2004), 129–145.

[60] Nauck, D., and Kruse, R. Obtaining interpretable fuzzy classification rules from medical data. *Artificial Intelligence in Medicine 16* (1999), 149–169.

[61] Noel, S., Chu, C. H., and Raghavan, V. Visualization of document co-citation counts. In *IEEE Symposium on Information Visualisation* (2002), pp. 691–696.

[62] Pancho, D. P., Alonso, J. M., and Alcalá-Fdez, J. A new Fingram-based software tool for visual representation and analysis of fuzzy association rules. In *IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)* (2013), pp. 1–7.

[63] Pancho, D. P., Alonso, J. M., Alcalá-Fdez, J., and Magdalena, L. Interpretability analysis of fuzzy association rules supported by Fingrams. In *Conference of the European Society for Fuzzy Logic and Technology (EUSFLAT)* (2013), Atlantis Press, pp. 469–474.

[64] Pancho, D. P., Alonso, J. M., Alcalá-Fdez, J., and Magdalena, L. Analyzing fuzzy association rules with Fingrams in KEEL. In *IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)* (2014), pp. 1–8.

[65] Pancho, D. P., Alonso, J. M., Cordón, O., Quirin, A., and Magdalena, L. FINGRAMS: visual representations of fuzzy rule-based inference for expert analysis of comprehensibility. *IEEE Transactions on Fuzzy Systems 21*, 6 (2013), 1133–1149.

[66] Pancho, D. P., Alonso, J. M., Kötter, T., Berthold, M. R., and Magdalena, L. Analyzing fuzzy rule-based systems with Fingrams in KNIME. In *XVII Congreso Español sobre Tecnologías y Lógica Fuzzy (ESTYLF)* (2014), pp. 597–602.

[67] Pancho, D. P., Alonso, J. M., and Magdalena, L. Quest for interpretability-accuracy trade-off supported by Fingrams into the fuzzy modeling tool GUAJE. *International Journal of Computational Intelligence Systems 6*, sup1 (2013), 46–60.

[68] Pancho, D. P., Alonso, J. M., and Magdalena, L. Enhancing Fingrams to deal with precise fuzzy systems. *Fuzzy Sets and Systems* (2015), In press, doi: 10.1016/j.fss.2015.05.019.

[69] Pancho, D. P., Alonso, J. M., Magdalena, L., Kötter, T., and Berthold, M. R. Fingrams KNIME plugin for visual analysis of fuzzy rule-based classifiers. In *World Conference on Soft Computing (WConSC)* (2014).

[70] Pham, B., and Brown, R. Analysis of visualisation requirements for fuzzy systems. In *International conference on computer graphics and interactive techniques (GRAPHITE)* (2003), pp. 181–187.

[71] Quirin, A., Cordón, O., Santamaría, J., Vargas-Quesada, B., and Moya-Anegón, F. A new variant of the Pathfinder algorithm to generate large visual science maps in cubic time. *Information Processing and Management 44*, 4 (2008), 1611–1623.

[72] Rehm, F., Klawonn, F., and Kruse, R. Visualiztion of fuzzy classifiers. *International Journal of Uncertainty, Fuzziness and Knowledge Based Systems 15*, 05 (oct 2007), 615–624.

[73] Ruspini, E. H. A new approach to clustering. *Information and Control 15*, 1 (1969), 22–32.

[74] Salton, G., and Bergmark, D. A citation study of computer science literature. *IEEE Transactions on Professional Communication 22* (1979), 146–158.

[75] Schvaneveldt, R. W., Durso, F. T., and Dearholt, D. W. Network structures in proximity data. *The psychology of learning and motivation: Advances in research and theory 24* (1989), 249–284.

[76] Scott, J. *Social Network Analysis: A Handbook (2nd edition)*. Sage Publications, 2000.

[77] Serrano, E., Quirin, A., Botia, J., and Cordón, O. Debugging complex software systems by means of pathfinder networks. *Information Sciences 180*, 5 (2010), 561–583.

[78] Sugeno, M. An introductory survey of fuzzy control. *Information sciences 36*, 1 (1985), 59–83.

[79] Tan, P., Kumar, V., and Srivastava, J. Selecting the right interestingness measure for association patterns. In *ACM International Conference on Knowledge Discovery and Data Mining* (2002), pp. 32–41.

[80] Trillas, E., and Guadarrama, S. Fuzzy representations need a careful design. *International Journal of General Systems 39*, 3 (2010), 329–346.

[81] Vargas-Quesada, B., and Moya-Anegón, F. *Visualizing the structure of science*. Springer-Verlag, 2007.

[82] Wasserman, S., and Faust, K. *Social Network Analysis: Methods And Applications (Structural Analysis in the Social Sciences)*. Cambridge University Press, 1994.

[83] Zadeh, L. A. Fuzzy sets. *Information and Control 8* (1965), 338–353.

[84] Zadeh, L. A. Outline of a new approach to the analysis of complex systems and decision processes. *IEEE Transactions on Systems Man, and Cybernetics 3* (1973), 28–44.

[85] Zadeh, L. A. The concept of a linguistic variable and its application to approximate reasoning. *Parts I, II, and III. Information Sciences 8, 8, 9* (1975), 199–249, 301–357, 43–80.

[86] Zadeh, L. A. Toward a theory of fuzzy information granulation and its centrality in human reasoning and fuzzy logic. *Fuzzy Sets and Systems 90*, 2 (1997), 111–127.

[87] Zhang, C., and Zhang, S. *Association Rule Mining: Models and Algorithms*, vol. LNAI2307. Springer, 2002.

[88] Zhou, S. M., and Gan, J. Q. Low-level interpretability and high-level interpretability: a unified view of data-driven interpretable fuzzy system modelling. *Fuzzy Sets and Systems 159*, 23 (2008), 3091–3131.

[89] Zizi, M., and Beaudouin-Lafon, M. Accessing hyperdocuments through interactive dynamic maps. In *ACM European conference on Hypermedia technology* (1994), pp. 126–135.