PRV-FCM: an extension of fuzzy cognitive maps for prescriptive modeling

William Hoyos[a,b], Jose Aguilar[b,c,d,e], Mauricio Toro[b]

[a]*Grupo de Investigaciones Microbiológicas y Biomédicas de Córdoba, Universidad de Córdoba, Montería, Colombia*
[b]*Grupo de Investigación en I+D+i en TIC, Universidad EAFIT, Medellín, Colombia*
[c]*Centro de Estudios en Microelectrónica y Sistemas Distribuidos, Universidad de Los Andes, Merida, Venezuela*
[d]*IMDEA Networks Institute, Madrid, Spain*
[e]*Corresponding author*

Abstract

In this paper, we present a methodology based on fuzzy cognitive maps (FCMs) and metaheuristic algorithms to generate prescriptive models, called PRescriptiVe FCM (PRV-FCM). FCMs are a set of concepts interrelated that describe the behavior of a system. This kind of modeling has been extensively used to build descriptive and predictive models. We propose an extension of FCMs to develop prescriptive models and support decision-making in different domains. This adaptation characterizes FCMs, using system and prescriptive concepts. After that, it uses a metaheuristic algorithm (in this case, we use a genetic algorithm) to optimize prescriptive concepts based on system concepts and the stability of the FCM. Our proposed prescriptive approach was implemented and tested in four scenarios where it demonstrated its capability to find solutions that lead to desired values for the variables of interest. Specifically, no significant differences were found between the values of the prescriptive variables in the datasets and those generated by PRV-FCM.
Keywords: Fuzzy cognitive maps, Prescriptive models, Metaheuristics, Modeling, Genetic algorithm

1. Introduction

Prescriptive modeling is a domain of business analytics, which aims to recommend actions within a system to reach the desired objective (Poornima and Pushpalatha, 2020). It is one of the

Email addresses: whoyos@correo.unicordoba.edu.co (William Hoyos), aguilar@ula.ve (Jose Aguilar), mtorobe@eafit.edu.co (Mauricio Toro)

areas that has attracted the most interest in recent years because it provides valuable support for decision-makers (Lepenioti et al., 2020). In this paper, we propose a methodology to generate prescriptive models using fuzzy cognitive maps (FCMs) and metaheuristic algorithms. FCMs are a set of nodes, also called concepts, which represent variables within a system; and arrows directed between them that indicate the influence of one concept on another (Kosko, 1986). Metaheuristic algorithms are techniques used to search solutions in an n-dimensional space, imitating eventually the behavior of individuals in nature (Sharma and Tripathi, 2022). These types of algorithms are commonly used to solve optimization problems.

The use of FCMs is interesting because of their ease of construction, reasoning and interpretation (Pelaez, 2019; Aguilar, 2001). FCMs have been widely used in descriptive (Stylios and Groumpos, 2004; Sánchez et al., 2019), diagnostic (Hoyos et al., 2022) and predictive modeling (Puerto et al., 2019; Mago et al., 2012; Papageorgiou et al., 2009); however, it has not been sufficiently used for prescriptive modeling. Despite the increase in the development of frameworks to generate prescriptive models (see Section 2), a common finding in all of them is that the prescriptive model uses as input the output of a predictive model. That is, the framework contains a predictive model that generates an output, and based on this output, the prescriptive model generates recommendations.

In previous works, we have presented the development of prescriptive models for healthcare environments. For example, in Hoyos et al. (2022), we presented a prescriptive model for dengue treatment using a genetic algorithm (GA) and prior predictions with artificial neural networks (ANNs) and support vector machines (SVMs). However, it was only the application of an existing algorithm on a dataset. Additionally, the prescriptive part could not be validated due to the lack of datasets with prescriptive variables.

Unlike our previously published work, in this case, we present a new technique that combines FCMs with optimization algorithms for the generation of prescriptive models. Our methodology has the particularity of generating prescriptive models with excellent performance in a variety of domains such as business, health and education. Our methodology uses an initial desired instance of the system, the FCM inference process, and an optimization algorithm to find the optimal values of the action or decision options. Furthermore, because it uses the FCM inference process, it allows generating prescriptive models that can be explained using previously defined relationships between concepts. This methodology is validated in different datasets with system and prescriptive variables.

The main contribution of this work is a methodology to generate prescriptive models using FCMs and metaheuristic algorithms. Our methodology consists, first, in the characterization of the FCM where a division of the map into two layers is established: system concepts and prescriptive concepts. System concepts are the set of variables that describe/define the system to be modeled. Prescriptive concepts are action variables that the decision-maker executes to obtain a desired result in the system.

The discrimination of the concepts in two layers allows using a metaheuristic algorithm to optimize the prescriptive concepts, and thus obtain the desired result in the concepts related to the system. The metaheuristic algorithm optimizes the concepts of the prescriptive layer using as fitness function only the concepts that by the business logic can be modified. At the end, we obtain the values of the prescriptive variables that lead to desired results for the system concepts, depending on the proposed problem.

Our framework, which is called PRescriptiVe FCM (PRV-FCM), is validated in four case studies to demonstrate its ability to prescribe actions within a system. We use one synthetic and three real datasets in the experiments. The real datasets correspond to business, medical and education domains. Based on the results obtained, our methodology can be used in any application domain and has the potential to generate prescriptive models that support decision-making in organizations.

The remainder of this paper is organized as follows: Section 2 shows a literature review of the last trends in prescriptive modeling and optimization approaches. Section 3 describes an overview of FCMs (learning and inference process). Section 4 shows our proposed prescriptive approach with its stages. Section 5 shows the specification of the case studies. Section 6 shows the experiments and results. Section 7 discusses the results and shows a comparison with previous works. Finally, Section 8 concludes the paper.

2. State-of-the-art

In this section, we show a brief literature review on prescriptive modeling and the current status of optimization approaches and models in different fields of science.

2.1. *Prescriptive analytics*

Business analytics is a discipline that uses data to find patterns and extract knowledge (Lopes et al., 2020). In this discipline can be defined different models, for example: i) descriptive modeling, ii) predictive modeling and iii) prescriptive modeling, among others. In descriptive modeling, the objective is to investigate what has occurred using the data (Lopes et al., 2020). Predictive modeling is concerned with predicting what is going to happen, and prescriptive modeling is concerned with suggesting or prescribing the best decision options. In some cases, the latter type of modeling uses the output of predictive modeling and artificial intelligence (AI) techniques to optimize and provide automated decisions (Lepenioti et al., 2020). While descriptive and predictive modelings are the most studied domains in business analytics (Lepenioti et al., 2020; Hoyos et al., 2021); prescriptive modeling is a less studied area, and its research interest is increasing due to its importance for decision making.

Over the last years, the number of papers focused on the proposal of frameworks for the generation of prescriptive models in different application domains is increasing Lepenioti et al. (2020). Lepenioti et al. (2020) reviewed the main approaches to generate prescriptive models. These approaches depend on the category of methods used to build them, e.g., mathematical programming (Berk et al., 2019; Dey et al., 2019), logic-based rules (Ramannavar and Sidnal, 2018; Srinivas and Ravindran, 2018), simulation (Jank et al., 2019), and machine learning (ML) (Hoyos et al., 2022; Revathy and Mukesh, 2020). In what follows, we explain some recent works in the previous categories.

2.1.1. *Mathematical programming*

Berk et al. (2019) used a robust and adaptive optimization approach to improve human resource planning by modeling uncertainty in hiring requests in a corporation. The methodology proposed by Berk et al. allowed to prescribe hiring actions to maximize their benefits and reduce negative scenarios. Dey et al. (2019) proposed a hybrid approach implementing computational intelligence techniques such as, ANNs and GAs to optimize the combination of steel properties in the industry. The goal of the Dey et al's approach was to find the combination of composition and processing parameters for steel to meet desired conditions. The models developed by Dey et al. demonstrated their ability to recommend actions to improve the quality of the steel produced.

2.1.2. Rules-based on logic

Ramannavar and Sidnal (2018) proposed a context model for the analysis of resumes to recommend or prescribe the best job for a particular candidate. The goal was to map a job offer to a resume. To achieve the goal, they used logic-based models, discovering hierarchical correlations between concepts extracted from resumes. Srinivas and Ravindran (2018) developed a generic framework for optimizing an appointment system in hospital environments. The developed framework first predicts an outcome based on patient data, and then prescribes the best decision with logical rules. The proposed framework outperforms benchmark rules reported in the literature.

2.1.3. Simulation

Jank et al. (2019) used prescriptive modeling to improve product portfolio designs in the industry. The proposed model supported product managers in designing product portfolios to align them with company objectives. Jank et al. used ANNs to quantify the correlations between product portfolio metrics and the company's strategic objectives to maximize success.

2.1.4. Machine learning

Revathy and Mukesh (2020) used prescriptive models to assure the privacy of information in Hadoop (a distributed processing platform). The goal was to generate a prescriptive model to distribute the data on nodes to avoid data leaks. The developed model recommends strategies to protect data from misuse by classifying the nodes in the system based on the information sensitivity. This model was based on unsupervised learning and suggest the node where the information must be placed. Finally, Hoyos et al. (2022) developed an autonomous cycle of data analysis tasks to predict severity with ANNs and SVMs; and prescribe the best treatment options for dengue fever with a GA. The application developed by Hoyos et al. could predict severity with high accuracy (98%), and based on that result, it prescribed the best actions for every patient based on World Health Organization guidelines for diagnosis and treatment of dengue.

2.2. Optimization approaches

The development of optimization approaches and models has increased in recent years. Here, we present some interesting works that have generated important results in several areas of knowledge.

Singh and Shukla (2022) developed a hybrid precoding multiple optimization algorithm for both minimizing the bit error rate in Mm-wave massive MIMO system and maximizing the energy and spectral efficiency of millimeter-wave wireless communications. Simulations performed during the research showed improved efficiency and cost when compared to other conventional algorithms reported in the literature. Pozna et al. (2022) combined the particle filter algorithm and the particle swarm optimization algorithm to minimize the energy consumption of integral servo systems. The coupling of these two techniques allows particle generation and a broadening of the search field to avoid local minima. The proposed approach allowed significant energy reduction in the fuzzy control system used in the experiments. The comparative results with other metaheuristics reinforce the capabilities of the proposed hybrid approach for energy reduction in the studied systems. Zamfirache et al. (2022) proposed an optimization approach that integrates the gray wolf optimization algorithm, reinforcement learning and iteration policies. The objective of the proposed approach was to train ANNs to optimize servo motor tracking control. The proposed approach was compared with approaches based on reinforcement learning and iteration policies that implement PSO and down gradient for optimization. However, the use of the gray wolf optimizer generates better results for the defined problem.

3. Fuzzy cognitive maps (FCMs)

In this section, we present an overview of FCMs, and their learning and inference processes.

3.1. Mathematical notation

In this subsection, the mathematical notation used in this article is briefly described. Vectors will be represented in lowercase letters ($\mathbf{v}$) and matrices with capital letters ($\mathbf{W}$). Both with bold letters. Vectors are by default represented as columns. We used the notation $a \in \mathbb{R}$ to indicate that an element is a scalar, the notation $\mathbf{v} \in \mathbb{R}^n$ to indicate a vector of length n. To indicate that an element is a matrix, we use the convention $\mathbf{W} \in \mathbb{R}^{n \times n}$. Superscripts are used to indicate the type of1 variable, while subscripts indicate a specific element in a vector (the i-th element) or matrix (the i-th and j-th element)

3.2. Overview of FCMs

FCMs are directed graphs that were introduced by Kosko (1986), taking as an initial idea the development of cognitive maps developed by Axelrod (1976). The map consists of nodes representing concepts and arrows representing relationships or influences between them. Nodes are causally related variables within a system, where one variable can cause some kind of effect on another. This type of relationship is represented with arrows directed from a source node to a destination node. Fig. 1 shows an example of FCM with five nodes or concepts and five relationships. The subindex in the edge value indicates the direction of that relationship, i.e. W_{15} indicates that the relationship goes from concept 1 (C_1) to concept 5 (C_5).

An FCM is composed of five main elements, summarized by the following expression (Hoyos et al., 2022):

$$\Omega = \langle n, \mathbf{v}, \mathbf{W}, f(\cdots), \mathbf{r} \rangle \tag{1}$$

Where Ω represents the tuple that contains all the elements of an FCM; n is the number of nodes or variables, $\mathbf{v}$ is an activation or initial vector —$\mathbf{v} \in \mathbb{R}^n$— that stores the value of the concepts or nodes at time $t = 0$ (see Eq. 2); $\mathbf{W}$ is a matrix that stores the causal relationships between the concepts, —$\mathbf{W} \in \mathbb{R}^{n \times n}$— (an example of a weight matrix for the FCM of Fig. 1 is shown in Eq. 3); finally, $f(\cdots)$ is a nonlinear activation function that keeps the values within a given range determined by $\mathbf{r}$. This range of $\mathbf{r}$ depends on the activation function used.

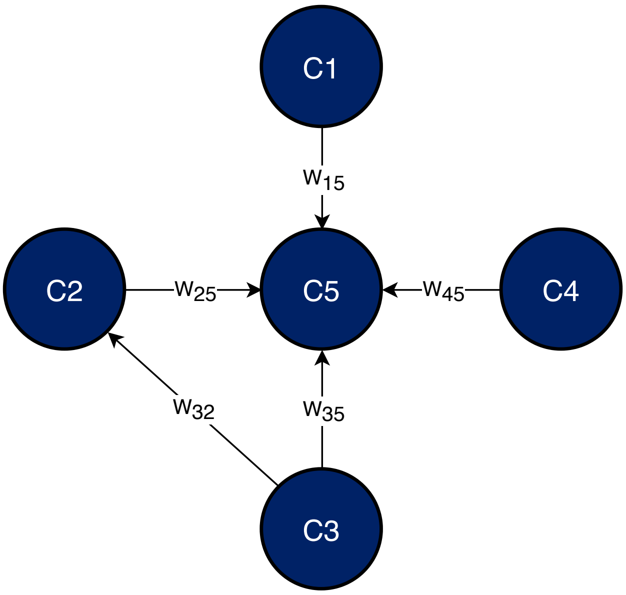

Fig. 1. Example of an FCM with five concepts and five relationships.

$$\mathbf{v}(0) = (\mathbf{v}_1(0), \mathbf{v}_2(0), \ldots, \mathbf{v}_n(0)) \tag{2}$$

$$\mathbf{W} = \begin{array}{c} \\ \boldsymbol{C_1} \\ \boldsymbol{C_2} \\ \boldsymbol{C_3} \\ \boldsymbol{C_4} \\ \boldsymbol{C_5} \end{array} \begin{array}{c} \begin{array}{ccccc} \boldsymbol{C_1} & \boldsymbol{C_2} & \boldsymbol{C_3} & \boldsymbol{C_4} & \boldsymbol{C_5} \end{array} \\ \begin{pmatrix} 0 & 0 & 0 & 0 & w_{15} \\ 0 & 0 & 0 & 0 & w_{25} \\ 0 & w_{32} & 0 & 0 & w_{35} \\ 0 & 0 & 0 & 0 & w_{45} \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix} \end{array} \tag{3}$$

Different activation functions can be used such as sigmoid and hyperbolic tangent. The use of each of them depends on the system to be simulated, i.e. the sigmoid function keeps the concept values between 0 and 1, while the hyperbolic tangent keeps them between -1 and 1. Table 1 shows the equations for some of the activation functions used in FCMs.

3.3. Learning of FCMs

The construction and optimization of FCMs can be carried out by two main approaches (Aguilar, 2013; Aguilar and Contreras, 2010): i) definition of concepts and assignment of relationships by

Table 1

Example of activation functions used in FCMs.

Activation function	Equation	Range
Bivalent	$f(x) = \begin{cases} 1 & x > 0 \\ 0 & x \leq 0 \end{cases}$	$f(x) \in \{0, 1\}$
Trivalent	$f(x) = \begin{cases} 1 & x > 0 \\ 0 & x = 0 \\ -1 & x < 0 \end{cases}$	$f(x) \in \{-1, 0, 1\}$
Sigmoid	$f(x) = \frac{1}{1+e^{-\lambda x x}}$	$f(x) \in [0, 1]$
Hyperbolic tangent	$f(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}}$	$f(x) \in [-1, 1]$

domain experts, and ii) optimization of the matrix that stores relationships between concepts. In the latter approach, ML algorithms are used to compute the matrix using historical data to fit specific patterns. It is worth mentioning that in the latter approach, human intervention is not necessary.

One of the most widely used algorithms for the construction of FCMs is Particle Swarm Optimization (PSO). PSO is a search algorithm described by Kennedy and Eberhart (1995), which is inspired by the behavior of insect swarms in nature. It can be used to train the weights matrix of an FCM where each particle i is an FCM and its position is a candidate weight matrix ($\mathbf{W}_i$). The process consists of two stages to move the particle to a new position: i) update the particle velocity, ii) update the particle position (Salmeron et al., 2017). Formally, the PSO algorithm can be described by two equations. First, the update of the velocities:

$$\mathbf{v}_i(t+1) = \mathbf{v}_i(t) + r_1 \cdot (\mathbf{W}_i^{best} - \mathbf{W}_i(t)) + r_2 \cdot (\mathbf{W}_i^{gbest} - \mathbf{W}_i(t)) \tag{4}$$

Where $\mathbf{v}_i(t)$ is the velocity of particle i at instant t, r_1 and r_2 are two random values generated during the search process; $\mathbf{W}_i^{best}$ is the best position the particle has passed through throughout the search process and $\mathbf{W}_i^{gbest}$ is the best global position of the whole swarm.

After the particle velocities are updated, the positions are updated using the following equation:

$$\mathbf{W}_i(t+1) = \mathbf{W}_i(t) + \mathbf{v}_i(t) \tag{5}$$

Finally, the algorithm generates the best weight matrix ($\mathbf{W}_i$) using previous updates. We used this algorithm to create and train FCMs (the predictive models) using synthetic and real datasets, due to the lack of domain experts and avoid bias introduced by humans.

3.4. Inference of FCMs

The reasoning or inference process of FCMs is carried out by successive multiplication of an activation vector $\mathbf{v}$ with a square matrix $\mathbf{W}$ corresponding to the influences between those concepts. One of the goals of this process is to predict an outcome. The inference procedure is iterative through time t and ends when the steady-state is reached. The equilibrium point is achieved when the difference between the value of the concept at time $t + 1$ and the value of the concept at time t is less than or equal to 0.0001. As an example, the following expression represents the calculation of the vector values in the first iteration of the inference process using the Kosko function (Kosko, 1986):

$$\mathbf{v}(1) = \begin{bmatrix} — & \mathbf{W}_1^T & — \\ — & \mathbf{W}_2^T & — \\ — & \vdots & — \\ — & \mathbf{W}_4^T & — \end{bmatrix} \begin{bmatrix} \mathbf{v}_1(0) \\ \mathbf{v}_2(0) \\ \vdots \\ \mathbf{v}_n(0) \end{bmatrix} \tag{6}$$

For simulation with FCMs, different inference functions have been developed such as the one developed by Kosko (Kosko, 1986), the modified Kosko (Stylios and Groumpos, 2004), and the rescaled one (Papageorgiou, 2011). Table 2 shows the equations of each of the inference functions and their main characteristics.

4. Our proposed prescriptive approach: PRV-FCM

In this work, we propose a methodology to generate prescriptive models with FCMs and meta-heuristic algorithms, called PRV-FCM. Prior to the generation of the prescriptive model, an FCM must be built for each specific problem. In this research, we used data-driven PSO-FCM for the construction of the predictive models with FCMs. In this case, FCMs were not built based on expert knowledge and experience but based on historical data, due to better model performance when this approach is used.

Table 2

Main properties of inference functions used for reasoning in FCMs.

Inference function	Equation	Main characteristics
Kosko (Kosko, 1986)	$\mathbf{v}_j(t+1) = f\left(\sum_{i=1,i\neg j}^{n} \mathbf{W}_{ij}\mathbf{v}_i(t)\right)$	It does not include the values of the concepts in the previous iteration. The FCM has no memory capacity and the change between iterations tends to be abrupt.
Modified Kosko (Stylios and Groumpos, 2004)	$\mathbf{v}_j(t+1) = f\left(\sum_{i=1,i\neg j}^{n} \mathbf{v}_j(t) + \mathbf{W}_{ij}\mathbf{v}_i(t)\right)$	It includes the value of the concept in the previous iteration; therefore, the FCM has memory capacity and the change after each iteration is done in a smoother way.
Rescaled (Papageorgiou, 2011)	$\mathbf{v}_j(t+1) = f\left(\sum_{i=1,i\neg j}^{n}(2 \times \mathbf{v}_j(t) - 1) + \mathbf{W}_{ij}(2 \times \mathbf{v}_i(t) - 1)\right)$	Solve the problem with initial concept values of 0, which when passed to the activation function take values of 0.5 in the second iteration.

The generation of a prescriptive model with PRV-FCM requires three steps: i) Characterization of the FCM, ii) Initial instantiation of the FCM, and, iii) Inference and optimization processes. Fig. 2 represents the methodological framework to generate prescriptive models using PRV-FCM. In the following, we describe briefly the stages of our approach.

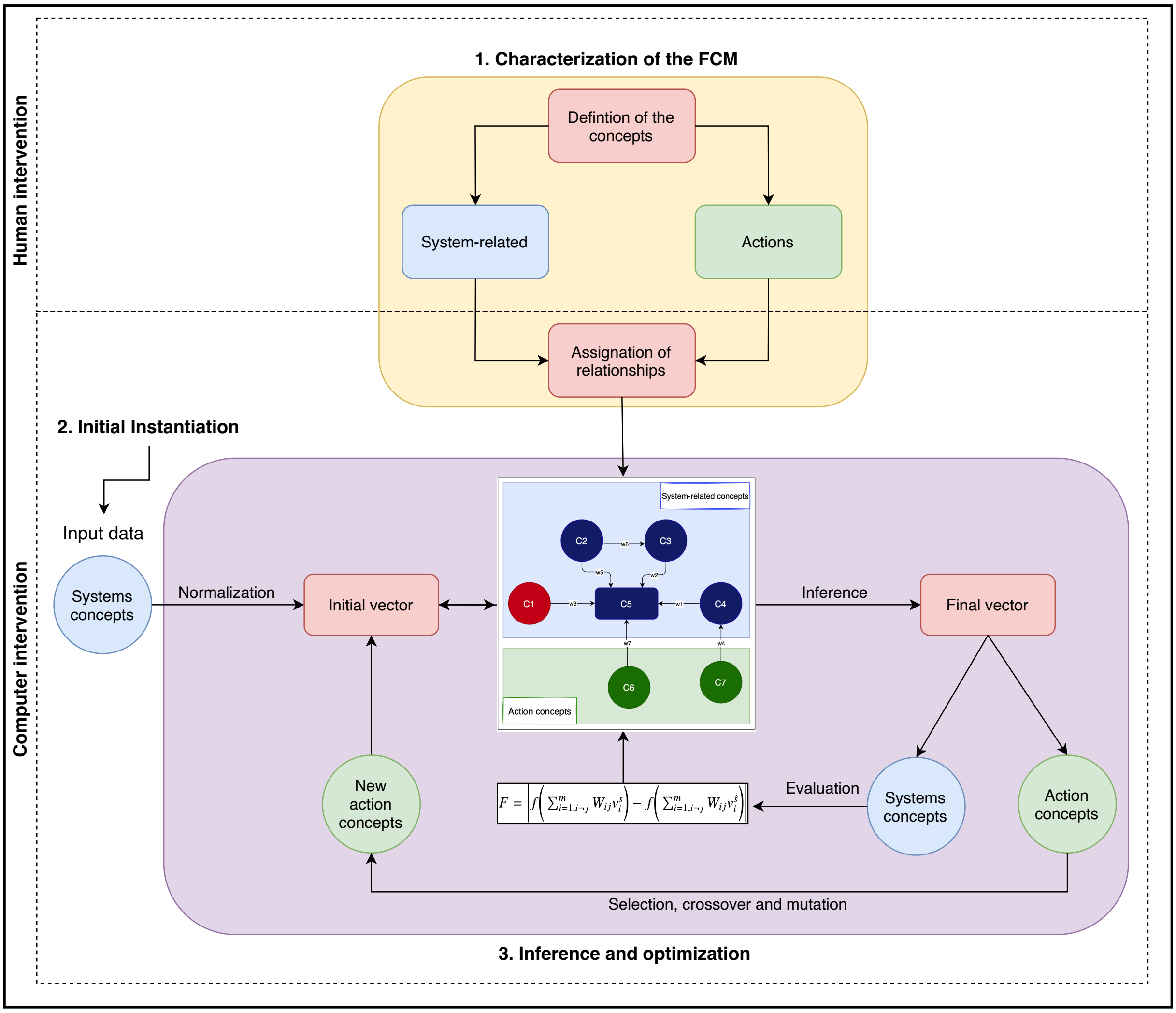

Fig. 2. Methodological framework to generate prescriptive models using PRV-FCM.

4.1. Characterization of the FCM

The first step consists of characterizing the FCM by classifying its concepts into two main layers: the system layer (blue area in Fig. 3) and the action layer (green area in Fig. 3). The first layer contains the concepts that are related to the system, which describe it. In the action layer, there are the prescriptive or action concepts that wish to be found for the system to reach the desired state. In the following, we define each of the concepts involved in the prescriptive model:

1. system-related concepts ($\mathbf{v}^s$): are those that belong to the system to be studied. System concepts are the variables that describe a system that relate to each other to achieve an objective. For example, a disease is a biological system where the system concepts correspond to signs, symptoms and alterations in the body of the sick person. This kind of concept is classified into changeable and non-changeable:
   - Non-changeable concepts ($\mathbf{v}^{nc}$): are those that cannot be modified in the logic of the system. For example: in a biological system, biological sex is a non-changeable concept in real-time.
   - Changeable concepts ($\mathbf{v}^c$): are those that can change during the simulation of the system. For example: in a biological system, changeable concepts could be those that can be minimized or maximized for a particular objective. Some examples of minimization could be symptoms of a patient with a disease. These concepts are the ones to be optimized following decision guidelines that are represented by the action concepts.
2. Action concepts ($\mathbf{v}^a$): are those that act on the changeable concepts of the system to optimize them to achieve a desired result in the system. These concepts have a causal effect on some changeable concepts related to the system. Moreover, these variables are the ones that make up the prescriptive model. An example of this type of concept could be a treatment (i.e. analgesic), which has an effect on reducing a patient's symptom (i.e. headache).

Fig. 3 shows a general example of an FCM to define a prescriptive model. In this figure, we can see the types of concepts for this problem: system-related and action concepts. Thus, to use an FCM as a prescriptive model, three types of concepts and causal relationships to the changeable concepts are defined.

4.2. Initial instantiation of the system-related concepts

The system's initial vector – $\mathbf{v}^s(0)$ – corresponds to the values of the system concepts desired by the decision maker. This vector serves as input to PRV-FCM to find the values of the action

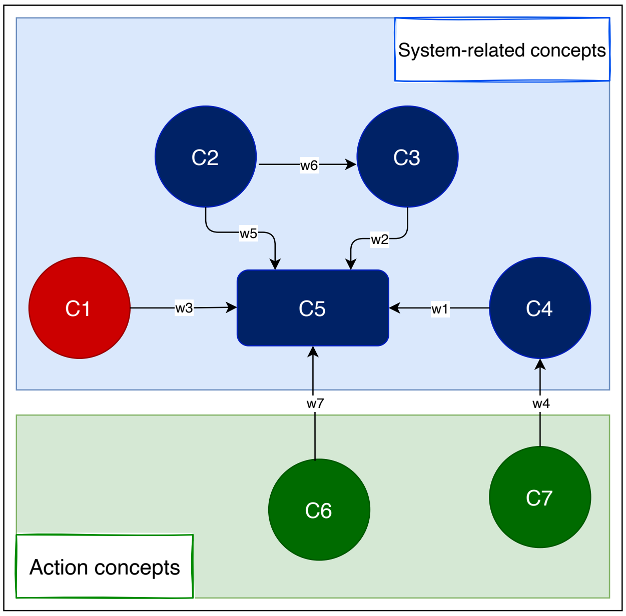

Fig. 3. Example of an FCM used as a prescriptive model. The non-changeable concepts are represented in red, while the changeable in dark blue.

concepts leading to that initially defined system state.

$$\mathbf{v}^s(0) = \begin{bmatrix} \mathbf{v}_i^s(0) \\ \vdots \\ \mathbf{v}_o^s(0) \end{bmatrix} = \begin{bmatrix} \mathbf{v}_i^{nc}(0) \\ \vdots \\ \mathbf{v}_p^{nc}(0) \end{bmatrix} \cup \begin{bmatrix} \mathbf{v}_i^c(0) \\ \vdots \\ \mathbf{v}_q^c(0) \end{bmatrix} \tag{7}$$

where, $\mathbf{v}^{nc}(0)$ is a vector with non-changeable concepts and p is the number of them, and the vector $\mathbf{v}^c(0)$ stores the changeable system-related concepts and q is the number of them.

4.3. Optimization process

In this section, we describe the process for finding the optimal action values that lead to the desired values of the system concepts. The optimization mechanism internally uses the PRV-FCM inference process to generate solutions and, once evaluated, they are discarded or selected. A more in-depth explanation of this process is described below.

Algorithm 1 describes the steps of the optimization process, which are carried out to obtain the optimal values of action concepts that generate the desired values of the system concepts. In our case, we used a GA for the optimization process, however, any metaheuristic can be used to obtain these values. GA is a probabilistic search technique used to find the optimal subset of features for a specific problem. This algorithm is an ideal choice for the optimization of prescriptive concepts with FCMs due to several reasons: 1) its ease of implementation and versatility; the general structure of a GA is the same regardless of the problem to be solved. 2) the ability to operate simultaneously with several solutions, instead of working sequentially like other techniques. 3) They use the information provided by the objective function and do not require other methods or auxiliary knowledge. These features make it more easily adaptable to different problems.

The input parameters of the algorithm are the initial vector of desired system concepts $\mathbf{v}^s(0)$. The algorithm must initially know the desired values of the system, a matrix of weights $\mathbf{W}$ that corresponds to the relationships that exist between all the concepts, a stop condition of the algorithm to avoid an infinite loop, the dimensions of the action vector, and the fitness function F to evaluate the individuals generated in the process.

To start the optimization process with the GA (Step 1), the generation counter is set to 0 (Step 2). Step 3 is the random generation of a population of individuals or vectors $\mathbf{v}^a(0)$ with dimension s corresponding to the action concepts. A solution (individual) is a set of possible values for each of the action concepts. The vector $\mathbf{v}^a(0)$ is defined as follows:

$$\mathbf{v}^a(0) = \begin{bmatrix} \mathbf{v}_i^a(0) \\ \vdots \\ \mathbf{v}_s^a(0) \end{bmatrix} \tag{8}$$

where $\mathbf{v}^a(0)$ stores the prescriptive concepts and s is the number of them.

For the inference process, an initial vector $\mathbf{v}(0)$ is required. This vector is constituted by an initial vector of the system $\mathbf{v}^s(0)$ corresponding to the desired state of the system that was defined in the previous stage. The other component of the initial vector is an initial action vector $\mathbf{v}^a(0)$, which was randomly generated at the beginning of the process and is the objective vector to optimize. These two vectors are combined to form the initial vector (Step 4). This step is important, so that the dimensions of the weight matrix should match the dimensions of the initial vector and thus the dot product is performed correctly (see Eq. 6). The initial vector is defined as:

Algorithm 1: Optimization algorithm to generate optimal values of action concepts

Input : $\mathbf{v}^s(0)$ = desired system vector, $\mathbf{W}$ = weigh matrix, F = fitness function, sc = stop condition

Output: Optimal $\mathbf{v}^a$

```
begin
    Set the generations counter g = 0
    Generate randomly one population P(0) of individuals v^a
    Generate initial vector v(g) with the combination of v^s(0) and v^a
    while sc is not met do
        for each initial vector v(g) do
            v_final = f( sum_{i=1}^{n} W v(g) )
            for each final vector v_final ∈ P(g) do
                Split final vector v_final in v^â and v^ŝ
                Evaluation of fitness:
                F = | f( sum_{i=1,i¬j}^{n} W_ij v_i^s ) − f( sum_{i=1,i¬j}^{n} W_ij v_i^ŝ ) |
            end
        end
        Selection of best individuals v^â by genetic operators
        Generation of a new population P(g + 1)
        g = g + 1
    end
    return optimal v^a
end
```

where the equations are

$$\mathbf{v}_{final} = f\left(\sum_{i=1}^{n} \mathbf{W}\mathbf{v}(g)\right)$$

$$F = \left| f\left(\sum_{i=1,i\neg j}^{n} \mathbf{W}_{ij}\mathbf{v}_i^{s}\right) - f\left(\sum_{i=1,i\neg j}^{n} \mathbf{W}_{ij}\mathbf{v}_i^{\hat{s}}\right)\right|$$

$$\mathbf{v}(0) = \mathbf{v}^s(0) \cup \mathbf{v}^a(0) = \begin{bmatrix} \mathbf{v}_i^s(0) \\ \vdots \\ \mathbf{v}_o^s(0) \end{bmatrix} \cup \begin{bmatrix} \mathbf{v}_i^a(0) \\ \vdots \\ \mathbf{v}_s^a(0) \end{bmatrix} \tag{9}$$

Subsequently, with the constructed initial vector $\mathbf{v}(0)$, the inference process is performed (Step 7). Unlike a classical FCM that consists of a five-element tuple (see Eq. 1), PRV-FCM is represented as a twelve-element tuple. Eq. 10 represents mathematically all the PRV-FCM elements and Table 3 shows a comparison between classical FCM and PRV-FCM elements.

$$\Psi = \langle n, \mathbf{W}, f(\cdots), \mathbf{r}, \mathbf{v}^s, o, \mathbf{v}^{nc}, p, \mathbf{v}^c, q, \mathbf{v}^a, s \rangle \tag{10}$$

Table 3

Comparison of elements used in classical FCMs and our proposed prescriptive approach (PRV-FCM).

Element	**Definition**	
	Classical FCM	**PRV-FCM**
n	Total number of variables	total number of variables
$\mathbf{W}$	Weight matrix for the FCM	Weight matrix for PRV-FCM
$f(\cdots)$	Threshold function	Threshold function
$\mathbf{r}$	Range of concept values	Range of concept values
$\mathbf{v}^s$	–	System concepts
o	–	Number of system concepts
$\mathbf{v}^{nc}$	–	Non-changeable concepts
p	–	Number of non-changeable concepts
$\mathbf{v}^c$	–	Changeable concepts
q	–	Number of changeable concepts
$\mathbf{v}^a$	–	Action concepts
s	–	Number of action concepts

The inference process consists of the iterative computation (t iterations) of initial vector $\mathbf{v}(0)$ with the weight matrix $\mathbf{W}$ to obtain a final stable vector. This process is defined by the equations described in Table 2. However, for the practical case, we will use a modification of the inference function proposed by Kosko (1986):

$$\mathbf{v}_{final} = f\left(\sum_{i=1}^{m} \mathbf{W}\mathbf{v}(t)\right) \tag{11}$$

The result of the inference process is a final vector that corresponds to the steady state of PRV-FCM (Step 8). This final vector is divided again into action vector $\mathbf{v}^{\hat{a}}$ and system vector $\mathbf{v}^{\hat{s}}$ (Step 9). The fitness of the action vector is evaluated using the $\mathbf{v}^{s}(0)$, $\mathbf{v}^{\hat{s}}$ and a fitness function (Steps 10 and 11). We proposed several fitness functions (see Table 4). For this explanation, we use the following function:

$$\min \left| f\left(\sum_{i=1,i\neg j}^{m} \mathbf{W}_{ij}\mathbf{v}_i^{s}\right) - f\left(\sum_{i=1,i\neg j}^{m} \mathbf{W}_{ij}\mathbf{v}_i^{\hat{s}}\right)\right| \tag{12}$$

$$\text{s.t.} \quad l_l < \mathbf{v}^{\hat{s}} < l_u \tag{13}$$

where $\mathbf{v}_i^{s}$ indicates the desired state of the system concepts, $\mathbf{v}_i^{\hat{s}}$ indicates the vector of systems concepts as a result of inference with the PRV-FCM; l_l and l_u are lower and upper limits ([0, 1] or [-1, 1], depending on the inference function used), respectively.

Table 4

Fitness functions used to generate prescriptive models with PRV-FCM.

Fitness function	**Equation**
Prescriptive Kosko	$F = \left\| f\left(\sum_{i=1,i\neg j}^{n} \mathbf{W}_{ij}\mathbf{v}_i^{s}\right) - f\left(\sum_{i=1,i\neg j}^{n} \mathbf{W}_{ij}\mathbf{v}_i^{\hat{s}}\right)\right\|$
Prescriptive Modified Kosko	$F = \left\| f\left(\sum_{i=1,i\neg j}^{n} \mathbf{v}_j^{s} + \mathbf{W}_{ij}\mathbf{v}_i^{s}\right) - f\left(\sum_{i=1,i\neg j}^{n} \mathbf{v}_j^{\hat{s}} + \mathbf{W}_{ij}\mathbf{v}_i^{\hat{s}}\right)\right\|$
Prescriptive rescaled	$F = \left\| f\left(\sum_{i=1,i\neg j}^{n}(2 \times \mathbf{v}_j^{s} - 1) + \mathbf{W}_{ij}\mathbf{v}_i^{\hat{s}}\right) - f\left(\sum_{i=1,i\neg j}^{n}(2 \times \mathbf{v}_j^{\hat{s}} - 1) + \mathbf{W}_{ij}v_i^{\hat{s}}\right)\right\|$

Subsequently, crossover and mutation genetic operators are applied to the action vector to select the best individuals (Step 14) and create a new population with the best individuals (Step 15). The main objective of this stage is to find the values of the action concepts that when used in the inference process minimize the difference between the initial values of system concepts and the values of system concepts generated during the inference process. Finally, when the stop condition is reached, the optimal values for the action concepts are obtained (Step 18), and thus, they constitute the optimal prescriptive variables that lead to the desired state of the system.

5. Specification of the case studies

In this section, we specify case studies to validate the proposed approach. We used several case studies from different domains. We tested our methodology in a synthetic dataset, and after in three real datasets. Although work on prescriptive modeling continues to increase, there are still challenges that need to be addressed. For example, there is currently a low availability of datasets with prescriptive variables included. Of the selected case studies, only one dataset had variables considered prescriptive. We reviewed hundreds of datasets that were hosted in different repositories and could be downloaded to determine the nature of the features present in each of them and used this criterion for the selection of case studies. Finally, we selected datasets where system-related variables could be assumed to be prescriptive variables (see Section 7 for more details).

5.1. Synthetic case study

To carry out this first case study, we generated a balanced synthetic dataset for classification with 1000 records, 10 features and a binary class.

5.2. Wine case study

For this case study, we use the *red wine quality dataset* available in UCI ML datasets (Cortez et al., 2009). This dataset comprises 1599 records, 11 physicochemical variables, and the class (wine quality). Table 6 shows the 11 variables used in this case.

The wine quality is defined by an expert team by assigning a score between 0 and 10 (Cortez et al., 2009). This score is considered the class within the red wine dataset. Although the score can be assigned from 0 to 10, in the dataset, there were no wines with class categories below 3 or above 8. In addition, some categories were unbalanced, and there were categories with few instances (see Plot A in Fig. 4). To overcome this problem and avoid using oversampling techniques for each

Table 5

Features included in the synthetic dataset.

Concept	Concept type
C1	System
C2	Prescriptive
C3	System
C4	System
C5	System
C6	System
C7	System
C8	System
C9	Prescriptive
C10	System

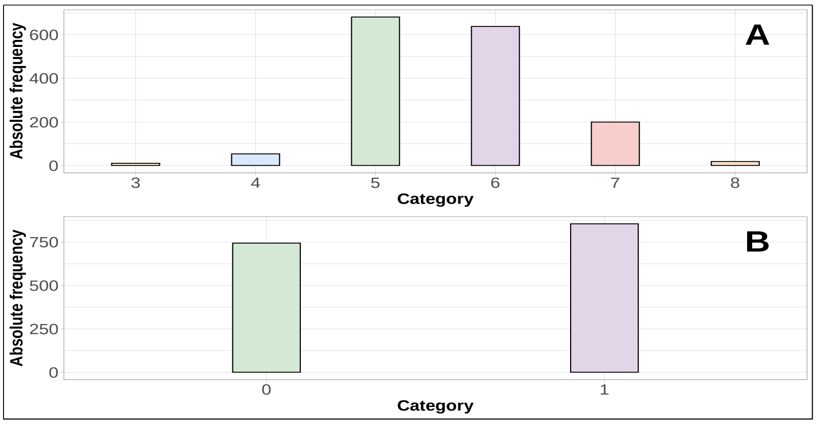

Fig. 4. Frequency distribution of the classes in the red wine dataset (Plot A corresponds to the frequency distribution of the classes in the original dataset. Plot B corresponds to the frequency distribution after reducing the classes to two categories).

minority class, we reduced the categories to two categories: *low quality*, when wine quality is lower and equal to 5 (class = 0); and *high quality*, when wine quality is greater than 5 (class = 1). At the end, we obtained a balanced dataset with 744 records for class 0 and 855 for class 1 (see Plot B in Fig. 4).

Table 6

Features included in the red wine dataset.

Feature	Concept	Concept type
Fixed acidity (g(tartaric acid)/dm^3)	C1	System
Volatile acidity (g(acetic acid)/dm^3)	C2	System
Citric acid (g/dm^3)	C3	System
Residual sugar (g/dm^3)	C4	System
Chlorides (g(sodium chloride)/dm^3)	C5	System
Free sulfur dioxide (mg/dm^3)	C6	Prescriptive
Total sulfur dioxide (mg/dm^3)	C7	System
Density (g/cm^3)	C8	Prescriptive
pH	C9	System
Sulphates (g(potassium sulphate)/dm^3)	C10	System
Alcohol (vol.%)	C11	System

Table 7

Features included in the diabetes dataset.

Feature	Concept	Concept type
Number of times pregnant	C1	System
Plasma glucose concentration a 2 hours in an oral glucose tolerance test	C2	Prescriptive
Diastolic blood pressure (mm Hg)	C3	Prescriptive
Triceps skin fold thickness (mm)	C4	System
2-Hour serum insulin (mu U/ml)	C5	System
Body mass index (weight in kg/(height in m)^2)	C6	System
Diabetes pedigree function (familiar and genetic antecedents)	C7	System
Age (years)	C8	System

5.3. Diabetes case study

Our third case study corresponds to diabetes, which is a disease characterized by high levels of glucose in the blood and complications related to this blood alteration (Pangaribuan and Suharjito, 2014). In this case, we used the *Pima Indians Diabetes Database* from the National Institute of Diabetes and Digestive and Kidney Diseases (Smith et al., 1988). The dataset is composed by 768 patients and 9 features, including the class (500 patients for class 0 = negative for diabetes, and

268 patients for 1 = positive for diabetes). Table 7 shows the variables included in this dataset. We used Synthetic Minority Oversampling Technique (SMOTE) (Chawla et al., 2002) to balance the classes, and at the end, we obtained 1000 records, 500 for each class.

Table 8

System-related categorical features included in the student academic performance dataset.

Feature	Feature type	Concept	Concept type	Categories
Gender	Demographic	C1	System	Male, female
Nationality	Demographic	C2	System	Kuwait, Lebanon, Egypt, Saudi Arabia, USA, Jordan, Venezuela, Iran, Tunis, Morocco, Syria, Palestine, Iraq, Lybia
Educational stages	Academic	C3	System	Lowerlevel, Middle School, High School
Grade levels	Academic	C4	System	01, 02, 03, 04, 05, 06, 07, 08, 09, 10, 11, 12
Classroom	Academic	C5	System	A, B, C
Topic	Academic	C6	System	English, Spanish, French, Arabic, IT, Math, Chemistry, Biology, Science, History, Quran, Geology
Semester	Academic	C7	System	First, Second
Responsible parent	Academic	C8	System	Mother, Father
Parent answering survey	Behavioral	C13	System	Yes, No
Parent school satisfaction	Behavioral	C14	System	Yes, No
Student absence days	Behavioral	C15	System	Above-7, Under-7

5.4. Student performance case study

Student academic performance is conceived as a construct that depends not only on student motivation, but also on other factors such as student-student relationships, demographic, socio-economic and psychological variables (Kumar and Pal, 2011). To test our approach, we used an educational dataset, called *Student Academic Performance*, with 480 students and 16 features (academic, demographics and behavioral) (Amrieh et al., 2016). This dataset has information collected from a Learning Management System called K360, which allows students access to online educational resources. Table 8 shows the categorical variables, and Table 9 shows the numeric features included in this dataset. These last variables will be the prescriptive concepts. For this experiment, we drop the variable *place of birth* because of its high correlation with the variable *nationality* (0.95). With respect to the class, we created two classes based on students' grade: class 0 when the grade is lower than 70, and class 1 when the grade is greater or equal to 70. We used SMOTE to balance the classes (353 records for each class).

Table 9

Action numerical features included in the student academic performance dataset.

Feature	Feature type	Concept	Concept type
Raised hand	Behavioral	C9	Prescriptive
Visited resources	Behavioral	C10	Prescriptive
Viewing announcements	Behavioral	C11	Prescriptive
Discussion groups	Behavioral	C12	Prescriptive

6. Experiments and results

In this section, we present the experiments set up to test the proposed approach. Additionally, we present the results of the generated models. First, we give an overview of the data preparation; then, we present the metrics to evaluate the performance of the models and finally, we present the results of the prescriptive models and their corresponding predictive model.

6.1. Data preparation

Normalization process was implemented to convert values in the range from 0 and 1 before feeding the FCMs (see Eq. 14). Specific processes of modification of variables in datasets are described in each case study's subsection. For all experiments, datasets were divided in training and testing in a proportion 70%/30%, respectively, and we use 10-fold cross-validation to find out the best configuration of hyperparameters (see Fig. 5).

$$x_{norm} = \frac{x_i - x_{min}}{x_{max} - x_{min}} \tag{14}$$

6.2. Evaluation metrics

We used accuracy as a metric for classification; mean absolute error (MAE), mean squared error (MSE), and root-mean-square error (RMSE) as error metrics to evaluate the prescriptive models; and prescriptive success rate (PSR) to determine the quality of the prescription. It's important to mention that we used accuracy as a metric because all datasets were balanced at the moment of developing the models, and error metrics because the prescriptive variables in datasets were numerical. In the following, we describe briefly each of these metrics.

- *Accuracy*: percentage of correctly classified examples among the total number of classified examples. Greater accuracy means a greater performance of the model.

$$Accuracy = \frac{TP + TN}{TP + FN + FP + TN} \tag{15}$$

  where TP are the true positives, TN are true negatives, FN are false negatives and TN are true negatives.

- *MAE*: calculated as an average of absolute differences between prescriptive concepts values and prescriptions.

$$MAE = \frac{1}{m}\sum_{i=1}^{m}\left|v_i^a - \hat{v}_i^a\right| \tag{16}$$

  where m is the number of records in the testing set, v_i^a is the actual prescriptive value and $\hat{v}_i^a$ is the prescribed value.

- *MSE*: measures the average square error of our prescriptions. For each point, it calculates the square difference between the prescriptions and the prescriptive concepts, and then averages those values.

$$MSE = \frac{1}{m}\sum_{i=1}^{m}(v_i^a - \hat{v}_i^a)^2 \tag{17}$$

- *RMSE*: is the squared root of the error described above.

$$RMSE = \sqrt{\frac{1}{m}\sum_{i=1}^{m}(v_i^a - \hat{v}_i^a)^2} = \sqrt{MSE} \tag{18}$$

- *PSR*: calculated as a ratio between prescribed values by PRV-FCM and actual values in the dataset. The numerator should be lower than the denominator to assure values between 0 and 1.

$$PSR = \frac{\sum_{i=1}^{s} \sum_{j=1}^{m} \hat{v}_i^a}{\sum_{i=1}^{s} \sum_{j=1}^{m} v_i^a} \tag{19}$$

6.3. *Training, validation and testing*

After data preparation, the model development process consisted of three stages: i) training, ii) validation and iii) testing. Seventy percent of the data from each case study was used for training and validation, while 30% was used for testing in cases not seen by the model. We used the 10-fold cross-validation technique to determine the best model and its hyperparameters. The 10-fold cross-validation process can be seen in Fig. 5. Specifically, this process divides the training dataset into ten subsets, taking nine for training and one for validation. Subsequently, it repeats the process by taking one subset different from the previous one for validation and the remaining nine for training. After ten training and validation processes, the best performing model and associated hyperparameters are selected. The best selected model is applied to the test data set to evaluate its performance on previously unseen models.

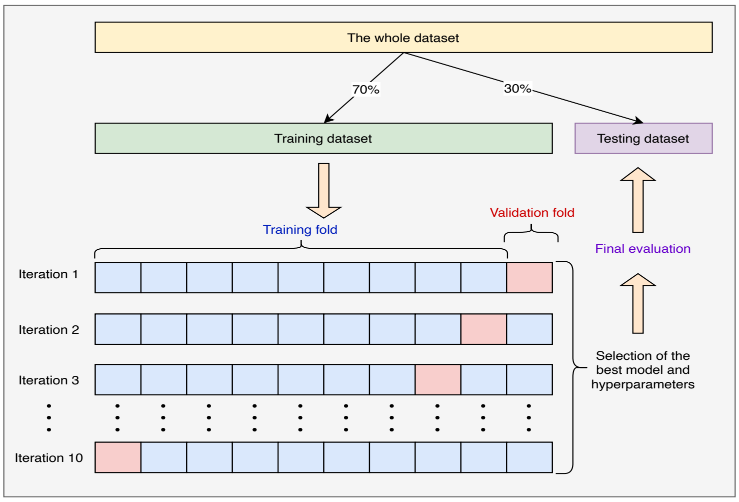

Fig. 5. Schematic representation of 10-fold cross-validation

For the case of predictive models, we use a grid of hyperparameters for tuning. We used the sigmoid and hyperbolic tangent activation functions with random values of slopes. We also use the

Kosko, modified Kosko and Rescaled inference functions for the reasoning process with the FCM. For the case of prescriptive models, we used a random grid of values for the hyperparameters of the GA such as initial population, number of generations, crossover probability and mutation probability.

6.4. Synthetic dataset

6.4.1. Predictive model

Before prescription with PRV-FCM, we generated an FCM using PSO. For that, using the training set from the previous dataset, we adjust the weights of FCM, and test using the testing set. The generated FCM was able to classify with 96% accuracy. Table 10 shows the results of the evaluation on the testing set, including the optimal hyperparameters. The results of this case study show an excellent performance of the model for prediction, and this result was expected because the dataset was built with well-differentiated clusters that allowed classifying the classes with high accuracy.

6.4.2. Prescriptive model

We selected two concepts from previously developed FCM as prescriptive variables: C_2 and C_9 (see Table 5). This decision was made because these two variables had no incoming influences on them, and in this way, we ensured that the value of them is not altered by other variables in the system. The idea is to find the optimal values of these prescriptive concepts that achieve a desired result in the system concepts. To find out these values, the PRV-FCM used a GA with different hyperparameters such as population, number of generations, crossover and mutation probabilities. The best configuration were: population = 50, number of generations = 20, crossover = 0.3 and mutation = 0.4. Table 11 shows the evaluation of PRV-FCM with respect to MAE, MSE and RMSE. The prescription results show an excellent performance of the model generated with PRV-FCM, such that it prescribes actions to achieve desired results of the system concepts in the dataset. The different levels of error measured between the values of the prescriptive variables in the dataset and the prescriptive values generated by PRV-FCM are very low (MAE < 0.014, MSE < 0.0004 and RMSE < 0.019) and demonstrate that our approach is useful for generating prescriptive models that optimize actions to achieve desired outcomes.

6.5. *Wine quality dataset*

6.5.1. *Predictive model*

To test the inference process, we generated an FCM using PSO. For that, using the training set generated from the previous dataset, we adjust the weights of generated FCM, and test in the testing set. The developed FCM was able to predict wine quality with 71% accuracy. Table 10 shows the results of the evaluation on the testing set, including the optimal hyperparameters.

Several predictive models have been developed with the previously described dataset in this case study. Our predictive model performed slightly better than that reported by Kumar et al. (2020), who developed a predictive model obtaining the best model with SVMs and an accuracy of 68%. On the other hand, our model had a similar performance to the one developed by Laughter and Omari (2020) where with Random Forest they obtained a performance of 72% accuracy. This comparison shows the competitive capacity of our model to predict wine quality using physicochemical characteristics as predictor variables. The performance of the predictive model was good, however, it could be improved by using other variables related to wine quality. For example, a high concentration of metals such as iron, aluminum and copper can alter the organoleptic properties of wine, and thus, its quality. In addition, these molecules have the ability to modify the turbidity and color of wine due to the formation of complexes with molecules present in wine such as tannins and anthocyanins (Frank and Kowalski, 1984).

6.5.2. *Prescriptive model*

In the wine dataset, there were no action concepts per se, but some concepts that can be modified by the decision-maker to achieve the desired result. We assume these variables as prescriptive concepts. For this case, we selected two of the FCM concepts as prescriptive variables: C_6 and C_8 (see Table 6). This decision was made because these two variables had no incoming influences on them, and in this way, we ensured that the value of them is not altered by other variables in the system. The idea is to find the optimal values of these prescriptive concepts that achieve a desired result in the system concepts. To find out these values, the PRV-FCM used a GA with different hyperparameters. The best configuration was population = 50, number of generations = 20, crossover = 0.5 and mutation = 0.5. Table 11 shows the evaluation of PRV-FCM with respect to

MAE, MSE and RMSE. As in the synthetic case study, the prescriptive results for this case study were excellent, finding MAE, MSE and RMSE values below 0.02, 0.0007 and 0.03, respectively. The prescriptive model generated with PRV-FCM has the ability to generate recommendations on prescriptive variables that lead to improved wine quality. Finally, because this dataset has not been used in any prescriptive model previously, unfortunately, we cannot compare it quantitatively with previous studies.

6.6. Diabetes dataset

6.6.1. Predictive model

The inference process was tested using an FCM built with PSO (with it was adjusted the FCM weights). Then, the FCM was tested in a testing set. The developed FCM model was able to predict diabetes with 70% accuracy. Table 10 shows the results of the evaluation in the testing set, including the optimal hyperparameters. Our model performed inferior to studies reported in the literature. For example, Olisah et al. (2022) and Hasan et al. (2020) developed predictive models for diabetes using feature selection techniques to improve model performance. The two studies presented performances above 90% using SVMs and correlation-based techniques, respectively. Despite the superiority of these models, our model has the advantage of being interpretable, where the inference process could be used to evaluate the behavior of variables over time.

The diagnosis or detection of diabetes is composed of the analysis of blood glucose levels, symptoms and risk factors present in the patient (Elliott and Pfotenhauer, 2022). The performance of predictive models for diabetes can be improved by using all the risk factors or symptoms used in the diagnosis of the disease. The acceptable performance of our predictive model could also be explained by the fact that many known symptoms of diabetes such as polydipsia, polyuria, polyphagia, abdominal girth, or risk factors such as physical exercise and obesity, were not available within the dataset. In addition, the small size of the dataset could influence the performance of the developed model. Several studies have shown that increasing the dataset size improves the performance of predictive models (Rácz et al., 2021; Barbedo, 2018; Quintero et al., 2021).

6.7. Prescriptive model

In the diabetes dataset there were no action concepts per se, but some concepts that can be modified by the decision-maker to achieve the desired result. We assume these variables as prescriptive concepts. For this case, we selected two of the FCM concepts as prescriptive variables: C_2 and C_3 (see Table 7). This decision was made because these two variables had no incoming influences on them, and in this way, we ensured that the value of them is not altered by other variables in the system. The algorithm finds the optimal values of these prescriptive concepts to achieve a desired result in the system concepts. To find out these values, the PRV-FCM used a GA with different hyperparameters, and the best configuration was population = 150, number of generations = 30, crossover = 0.5 and mutation = 0.3. Table 11 shows the evaluation of PRV-FCM with respect to MAE, MSE and RMSE. The very low values of MAE < 0.02, MSE < 0.0002 and RMSE < 0.015 of the prescriptive model generated with PRV-FCM demonstrate the ability of the model to generate recommendations that decrease the risk of diabetes. Like the wine quality dataset, the diabetes dataset has not been used in the literature to generate prescriptive models that generate recommendations or prescriptions to reduce the risk of diabetes.

6.8. Student performance dataset

6.8.1. Predictive model

To test the inference process, we generated an FCM using PSO and the training set from the previous dataset. Then, the FCM is tested in the testing set. Table 10 shows the results of the evaluation in the testing set, including the optimal hyperparameters. The FCM model can predict student academic performance with an accuracy of 85%. The performance of our predictive model was good and outperforms the results reported in the literature. Two studies reported by Amrieh et al. (2016, 2015) used this dataset to predict academic performance. The best performance of all experiments performed in the two studies was 80% accuracy using ANNs. Another work applied in the educational field was developed by Tan et al. (2014), who implemented a hybrid prediction approach composed of ANNs and structural equations to create a framework that identifies the factors that influence the adoption of mobile learning based on the technology acceptance model and psychological constructs. The results of applying the approach to academic datasets show that the technology acceptance model, social influence variables and academic qualifications significantly influence the intention to adopt mobile learning. According to these results, our model is superior because it better represents the functional dependencies between the predictor variables and student academic performance. Additionally, our model can be used to evaluate the behavior of student-performance-related variables in different scenarios.

6.8.2. *Prescriptive model*

In the student academic performance dataset, there were action concepts per se, such as C_9, C_{10}, C_{11} and C_{12} (see Table 9). These variables are behavioral, and the decision-maker decides if execute them or not. In other words, they are the actual actions that the student can take to achieve the desired result, in this case, to improve academic performance. The algorithm finds the optimal values of these prescriptive concepts that achieve a desired result in the system concepts. To find out these values, the PRV-FCM used a GA with the next configuration: population = 200, number of generations = 50, crossover = 0.2 and mutation = 0.3. Table 11 shows the evaluation of PRV-FCM with respect to MAE, MSE and RMSE. The results showed that the model generated with PRV-FCM generates prescriptions with very low error rates (MAE < 0.04, MSE < 0.0023 and RMSE < 0.048), which demonstrates that PRV-FCM is a useful methodology for the generation of prescriptions in the educational field.

Table 10

Performance and optimal hyperparameters of predictive models generated by classical FCMs for each dataset.

Case study	**Hyperparameters**			**Accuracy (%)**
	Activation function	**Slope**	**Inference function**	
Synthetic	Sigmoid	1	Modified Kosko	96.67
Wine	Sigmoid	10	Modified Kosko	70.62
Diabetes	Sigmoid	1	Modified Kosko	69.86
Student academic performance	Sigmoid	10	Modified Kosko	84.91

Table 11

Performance of prescriptive models developed with PRV-FCM. NA = not applicable.

Case study	FCM concept	Variable name	MAE	MSE	RMSE
Synthetic	C2	NA	0.01146	0.00024	0.01559
	C9	NA	0.01341	0.00035	0.01872
Wine	C6	Free sulfur dioxide	0.01761	0.00066	0.02583
	C8	Density	0.01991	0.00069	0.02644
Diabetes	C2	Plasma glucose concentration a 2 hours in an oral glucose tolerance test	0.01112	0.00020	0.01429
	C3	Diastolic blood pressure	0.00743	0.00010	0.01000
Student academic performance	C9	Raised hands	0.02674	0.00120	0.03466
	C10	Visited resources	0.03358	0.00188	0.04337
	C11	Viewing announcements	0.03560	0.00222	0.04717
	C12	Discussion groups	0.00999	0.00018	0.01349

6.9. Comparison of means

We performed a mean comparison test between the values of the variables in the dataset and the values of the variables prescribed by our approach. This is another way to test if our approach can generate prescriptive models with excellent performance because this comparison uses a hypothesis test to determine significant differences between two sets of data. Before comparing means, we tested the normality of the data using the Lilliefors test (Lilliefors, 1967). We used Student's t-test and Wilcoxon signed-rank test to compare the means between the two groups. The student's t-test was used when the two groups to be compared followed a normal distribution, while the Wilcoxon test was used when at least one comparison group did not follow a normal distribution. These tests use the following hypothesis to test:

- $H_0 : \bar{X}_{actual} = \bar{X}_{prescribed}$
- $H_1 : \bar{X}_{actual} \neq \bar{X}_{prescribed}$

Where H_0 is the null hypothesis stating that there are no significant differences between the prescriptive values of the dataset and the prescriptive values generated by our PRV-FCM approach. H_1 is the alternative hypothesis that states that there are significant differences between the prescriptive values of the dataset and the prescriptive values generated by our PRV-FCM approach. A significance value of 0.05 was established. If $p < 0.05$, the null hypothesis is rejected.

The results of this test are shown in Table 12. For all prescriptive variables in all case studies, $p > 0.05$ were found, indicating that there are no significant differences between the prescriptive values in the dataset with the prescriptive values generated with our approach. The p-value indicates the probability that the prescriptive values could have occurred under the previously defined null hypothesis (H_0). The higher the p-value, the closer the values prescribed by PRV-FCM are to the values included in the dataset. With respect to this comparison, the best performance was for the prescriptive models generated for diabetes and student academic performance case studies, which yielded p-values above 0.934 for all prescriptive variables. This result is possible because variables in these two datasets follow a normal distribution. For this reason, the parametric Student's t-test was used. It has been shown that this type of parametric test has greater statistical power than non-parametric tests such as the Wilcoxon test (Amandeep and Robin, 2015; Grech and Calleja, 2018). With respect to the variables C2 and C9 of the synthetic dataset and C8 for the wine dataset, they did not follow a normal distribution, a test with statistical power lower was used; therefore, the p-value is lower because a higher variability of the data influences the comparison test. Despite of these results, we can confirm the excellent performance of our approach to generate prescriptive models. To ensure full transparency of the results obtained with our approach, the synthetic data, those used for statistical comparison and the architecture of the FCM models are available in (Hoyos, 2023). Links to the datasets of the other case studies are cited in Section 5.

In summary, the prescriptive models generated with PRV-FCM in all case studies presented excellent performance with very low error rates and without significant differences between the actual and prescribed values. This demonstrates the general capacity of our approach for generating prescriptive models with excellent performance in any domain. A broader discussion of the results obtained with respect to prescriptive models is made in the next section.

7. Discussion

In this paper, we propose a methodology to generate prescriptive models. The main objective of this type of model is to find ideal actions that lead to the desired outcome. To test our approach,

Table 12

Mean comparisons between the values in the dataset and prescribed values using PRV-FCM. NA = not applicable († indicates the Wilcoxon test was performed because the corresponding variable did not follow a normal distribution, ⋆ indicates the Student's t-test was performed because the corresponding variable follows a normal distribution).

Case study	Prescriptive concept	Variable name	Mean±SD		p
			Actual	**Prescribed**	
Synthetic	C2	NA	0.448±0.305	0.432±0.315	0.844†
	C9	NA	0.523±0.313	0.500±0.317	0.776†
Wine	C6	Free sulfur dioxide	0.216±0.128	0.216±0.127	0.966⋆
	C8	Density	0.522±0.101	0.528±0.101	0.672†
Diabetes	C2	Plasma glucose concentration a 2 hours in an oral glucose tolerance test	0.547±0.130	0.547±0.129	0.973⋆
	C3	Diastolic blood pressure	0.557±0.172	0.558±0.169	0.970⋆
Student academic performance	C9	Raised hands	0.568±0.250	0.570±0.247	0.961⋆
	C10	Visited resources	0.697±0.263	0.694±0.257	0.934⋆
	C11	Viewing announcements	0.449±0.239	0.449±0.250	0.999⋆
	C12	Discussion groups	0.462 + 0.256	0.462±0.257	0.987⋆

we use four case studies in different domains. The results shown in the previous section demonstrate the capabilities of our approach to generate prescriptive models with excellent performance. In this section, we first analyze the results of the prescriptive models developed. Subsequently, we made a quantitative comparison with papers that used the same datasets that we used in the present study. Then, in the qualitative comparison, we focused on comparing computational intelligence models that had a similar architecture to the ones we used to develop our approach. In this case, we specifically refer to FCM and GAs.

7.1. *Analysis of results of the prescriptive models*

The generation of prescriptive models is increasing; however, the availability of data with prescriptive variables or actions is low. Therefore, in two case studies (wine and diabetes), we assumed some system variables to be "prescriptive". In those cases, the generated "prescriptive" model would behave as a recommender, let's see why. Fig. 6 represents the difference between actions variables and system variables defined as prescriptives:

- A variable is *prescriptive* when it is considered an action within the system (see side A in

Fig. 6). That is, this variable acts on the other concepts in the system, but there is no variable acting on it. For example, if we have a patient with a febrile illness, then the symptom *fever* is the system concept, while taking paracetamol is considered the prescriptive variable that will decrease the fever. Additionally, there is no variable within the logic that acts on the paracetamol.

- On the other hand, we have system variables that can be modified, but by actions that are not within the initial system or dataset. In these cases, the variable can behave as prescriptive, and a system could recommend modifications on this variable but not the action to modify it (see side B in Fig. 6). For example, *body temperature* is a system variable that can be used as a prescriptive variable. In this case, A prescriptive system might recommend raising or lowering the temperature, but does not specify the action to change the *body temperature* values.

It is crucial to keep this aspect in mind to explain the results of each case study.

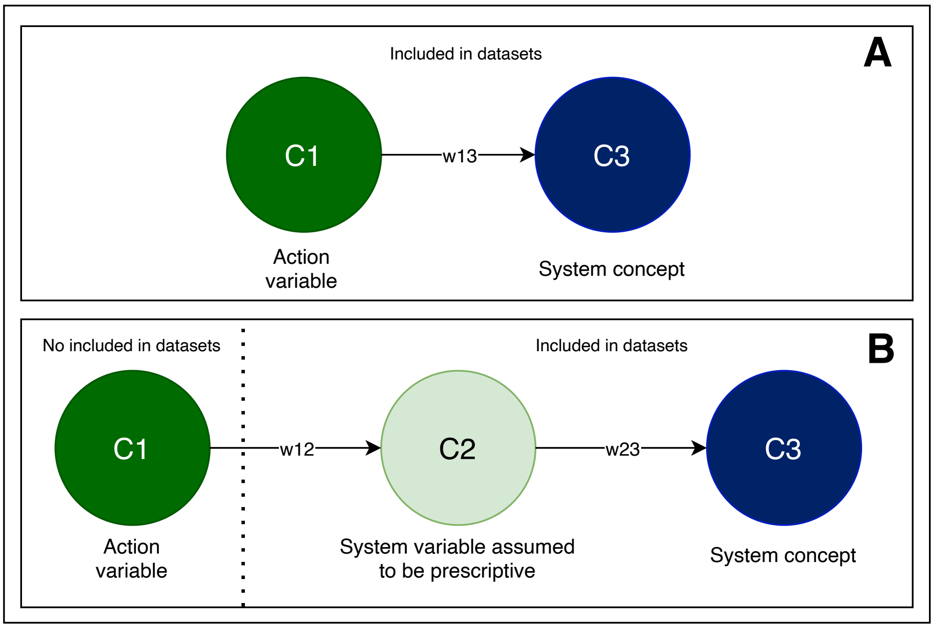

Fig. 6. Difference between using a system variable as “prescriptive” **(A)** and action or prescriptive variable **(B)**. In case A, a system concept is used as a prescriptive, but this concept is not an action. In case B, the system has action variables or prescriptive variables per se.

For the synthetic case study, prescriptive variables were assumed, due to the lack of context within the dataset. We assumed variables C_2 and C_9 because they were variables that did not have the influence of other variables in the FCM. These variables are considered actions or prescriptive because the decision-maker uses them to generate the desired outcome in the simulated system. According to the results of the prescriptive model, with our approach, it is possible to find the optimal values of decision-driven actions to obtain desired results with excellent performance (see Table 11).

Wine quality is a characteristic of wine obtained through the process where experts assign a quality score to different types of wine with different physicochemical characteristics (Cortez et al., 2009) (see Table 6). According to the prescriptive model, we identified prescriptive variables such as sulfur dioxide free and density. It is important to remember that these variables (sulfur dioxide and density) are not actions per se. They are system variables that we assign them the role of prescriptive to test our approach due to the lack of datasets with actions or prescriptions. For this particular case, what the prescriptive model does is to recommend what should be done with these two variables. For example, if the decision-maker wants to improve the quality of the wine, the model, in this case, will recommend that he/she adjusts the density values, but it will not specifically give him/her the action to change the density values to improve the quality of the wine. The decision-maker will be able to modify the wine density using different actions that were not initially included in the system, as they were not present in the dataset.

Diabetes is a disease that has high morbidity rates due to its associated complications. Our third case study consisted of generating both a predictive model to classify patients and a prescriptive model to recommend values for blood glucose concentrations and diastolic blood pressure. Similar to the previous case study, we assumed these variables as prescriptive, due to the lack of datasets with actions aimed at treating the disease. The performance of the model in prescribing is very outstanding, with very low errors when comparing what was prescribed with what was stored in the dataset (see Table 11). Our approach allows generating prescriptive models with high performance, even if the variables being prescribed are not prescriptive or actions themselves (in these last two cases, recommendations).

Student academic performance is a fundamental aspect in education and different strategies

Table 13

Behavior of prescriptive models generated with PRV-FCM according to types of prescriptive variables in the case studies' datasets.

Case study	**Prescriptive**	**Recommender**
Synthetic	✓	✓
Wine	✗	✓
Diabetes	✗	✓
Student academic performance	✓	✓

have been developed to maximize it in schools and universities. Unlike the case studies on wine and diabetes, in this dataset, there are prescriptive variables or actions (see Table 13), i.e., the decision-maker (in this case, the student) performs them directly to obtain the desired results. For this case study, it was not necessary to assume system variables as prescriptive because they are behavioral variables collected in the dataset. Clear examples are the variables *raised hands* and *Visited Resources* that correspond to the number of times the student raises his/her hand and the number of times he/she visits the academic resources available on the online education platform, respectively.

Thus, of the real datasets used in this work, only the student academic performance dataset has been used for prescriptive modeling. Harikumar et al. (2022) proposed an approach for prescriptive models. They used two classifiers (logistic regression and SVMs) and the student academic performance dataset, to demonstrate the applicability of their approach. The action variables selected by Harikumar et al. corresponded to the same variables that we selected to test our approach (see Table 9). For a quantitative comparison, we calculated the PSR defined in Eq. 19. Harikumar et al reported a PSR of 66% with logistic regression and 98% with SVMs. The PSR of our approach was 96%, a higher value than the logistic regression and a slightly lower than SVMs reported by Harikumar et al.

Although our approach was slightly inferior to the approach developed with SVMs by Harikumar et al., our approach has two advantages over this work: 1) usability: our approach proved to be excellent for prescribing in different fields or domains; Harikumar's approach is limited to preserving data privacy. 2) interpretability: our approach uses FCMs, which are interpretable and allow knowing the behavior of the variables over time (iterations); Harikumar et al's work used SVMs, known as a black box technique where it is difficult to know the behavior of the variables involved in the prescription.

7.2. *Qualitative comparison with previous works*

In this section, we compare our approach with previous work using qualitative criteria. Table 14 shows the criteria used and the evaluation.

He et al. He (2008) implemented a decision-oriented immune algorithm with FCMs. The results showed its capability for goal-oriented decision-making; however, there was no validation of the approach using synthetic or real data sets. In addition, the case study used does not allow for the real prescription because there were no prescriptive variables or actions per se. Thus, the model developed by He yue et al. is a recommender system that recommends what to do, but not how to do it.

Dey et al. (2019) implemented evolutionary and ML techniques such as GA and ANNs to recommend actions. The approach proposed by Dey et al. used a desirability function to improve the quality of steel in the industry, with the ability to recommend the properties that steel should have to be of the desired quality. Thus, the model recommends the properties but not the actions that lead to those properties. An advantage of this model is that it does not need a prior predictive model to recommend the desired characteristics.

Hoyos et al. (2022) implemented data analysis tasks to prescribe dengue treatment. The authors used a GA to find the optimal values of disease treatment options as reported by WHO. The prescriptive model developed had the ability to prescribe actions that reduce the severity of dengue. The only disadvantage of this model is the dependence on the output of a previous predictive model. The prescriptive model uses as input the outcome of the dengue severity prediction, and based on that outcome, it prescribes the best possible actions that minimize the severity of the disease.

Chalmers et al. (2015) proposed a prescriptive analysis approach to identify optimal orthotic corrections for adolescent idiopathic scoliosis. The authors implemented fuzzy logic to predict whether changes in bracing would improve or worsen the patient's deformity. This study was able to obtain good results to recommend actions to reduce the progression of the disease. The disadvantage of this study is the dependence on a previous predictive model. The prescriptive model developed needs to know the outcome of the prediction to generate an appropriate prescription. Another disadvantage of this model is its application. The model recommends adjusting variables to obtain the desired outcome but does not prescribe the action itself.

In summary, we propose a prescriptive approach using FCMs and metaheuristic algorithms, called PRV-FCM. The prescriptive and recommender capability of our approach has been validated on synthetic and real datasets. PRV-FCM has the ability to either recommend, prescribe, or perform both tasks with high performance. Regarding this last aspect, overfitting is one of the concerns that arise when models have excellent performance. Overfitting is a problem characterized by the inability of the model to generalize on unseen data. In our case, we use data partitioning into training and validation (70%) and testing (30%) to reduce overfitting. The random selection of subsets in the 10-run cross-validation process reduces the overfitting, considering that, if the solution performs consistently on several subsets of the population, its performance is likely to be consistent on unseen data. The results shown in this paper are the product of the application of the models on previously unseen data.

The convergence of the proposed algorithm is not affected by the metaheuristic because it uses the values of the final state of the FCM, which is a numerical vector and represents the stability of the system after successive multiplications with the weight matrix. Particularly, although our algorithm uses the combination of the metaheuristic algorithm and the FCM inference process, these processes are not combined within the learning process. The result of the inference process is a final numerical vector, which is used as the fitness function, while the metaheuristic searches the prescriptive values in the search space.

Finally, PRV-FCM only requires instantiating an FCM to obtain optimal prescriptions. Finally, our approach is intuitive because it is only necessary to define the variables involved in the system; it is extensible and easily adaptable to any domain in which it can be used.

Table 14

Results of qualitative comparison among our work and previous prescriptive approaches.

Qualitative criteria	Work				
	He (2008)	Dey et al. (2019)	Hoyos et al. (2022)	Chalmers et al. (2015)	Our work
Prescriptive capability	✗	✗	✓	✗	✓
Recommender capability	✓	✓	✓	✓	✓
Validated on datasets	✗	✓	✗	✓	✓
Intuitive, extensible and easily adaptable	✓	✓	✓	✓	✓

8. Conclusions

In this paper, we proposed a methodology to generate prescriptive models using FCMs and metaheuristic algorithms. First, we define a discriminated FCM with system concepts and action concepts. Subsequently, we implemented a GA to find the optimal values of action variables that lead to the desired outcome of the system variables using FCM inference. The results showed the ability of our approach to be used in different fields. We tested it on several datasets, one synthetic and others in the fields of business, health and education, with excellent performance.

The main goal of the proposed methodological framework was not to improve the performance of current FCM approaches but to introduce a useful methodology for the generation of prescriptive models. The particularity of our approach is the ability to recommend or prescribe actions, its good behavior with scarce datasets, and finally, its ease of use and adaptability to any area of knowledge.

This work has some limitations, such as: i) the use of experts at the beginning of the methodological framework to select the variables of interest, both system and action variables. Furthermore, in datasets where no action variables are stored, the human must select which system variables behave as prescriptive to generate the prescriptive models. ii) Other metaheuristic algorithms were not used to optimize the action concepts.

Future work should be aimed at using other metaheuristic techniques to improve the optimization process of our approach. In addition, automate the process so that the algorithm automatically selects prescriptive variables, for example, detecting those variables of interest that have no influence of other variables on them, differentiating them from variables that can be modified within the logic of the system. Also, testing other optimization algorithms or experts to build the FCMs could improve the performance of the developed models. Finally, a two-stage learning (first with the system concepts and then with the action concepts) could be useful to generate better-performing prescriptive models.

Acknowledgments

This study was partially funded by Colombian Administrative Department of Science, Technology and Innovation - COLCIENCIAS (grant number 111572553478) (M. Toro) and Colombian Ministry of Science and Technology *Bicentennial PhD Grant* (W. Hoyos). The authors are grateful to physicians for using their knowledge and experience to build the FCMs and interpreting the final results.

CRediT authorship contribution statement

William Hoyos: Conceptualization, Methodology, Software, Formal analysis, Investigation, Data curation, Validation, Visualization & Writing – original draft. **Jose Aguilar:** Conceptualization, Methodology, Formal analysis, Validation, Supervision, Writing – reviewing & editing. **Mauricio Toro:** Conceptualization, Resources, Supervision, Writing – reviewing & editing.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

S. Poornima, M. Pushpalatha, A survey on various applications of prescriptive analytics, International Journal of Intelligent Networks 1 (2020) 76–84. doi:10.1016/j.ijin.2020.07.001.

K. Lepenioti, A. Bousdekis, D. Apostolou, G. Mentzas, Prescriptive analytics: Literature review and research challenges, International Journal of Information Management 50 (2020) 57–70. doi:10.1016/J.IJINFOMGT.2019.04.003.

B. Kosko, Fuzzy cognitive maps, International Journal of Man-Machine Studies 24 (1986) 65–75. doi:10.1016/S0020-7373(86)80040-2.

V. Sharma, A. K. Tripathi, A systematic review of meta-heuristic algorithms in IoT based application, Array 14 (2022) 100164. doi:10.1016/j.array.2022.100164.

E. Pelaez, A fuzzy cognitive map (FCM) as a learning model for early prognosis of seasonal related virus diseases in tropical regions, 2019 6th International Conference on eDemocracy and eGovernment, ICEDEG 2019 (2019) 150–156. doi:10.1109/ICEDEG.2019.8734354.

J. Aguilar, A fuzzy cognitive map based on the random neural model, in: L. Monostori, J. Váncza, M. Ali (Eds.), Engineering of Intelligent Systems, Springer Berlin Heidelberg, Berlin, Heidelberg, 2001, pp. 333–338.

C. D. Stylios, P. P. Groumpos, Modeling Complex Systems Using Fuzzy Cognitive Maps, IEEE Transactions on Systems, Man, and Cybernetics Part A:Systems and Humans. 34 (2004) 155–162. doi:10.1109/TSMCA.2003.818878.

H. Sánchez, J. Aguilar, O. Terán, J. Gutiérrez de Mesa, Modeling the process of shaping the public opinion through Multilevel Fuzzy Cognitive Maps, Applied Soft Computing Journal 85 (2019) 105756. doi:10.1016/j.asoc.2019.105756.

W. Hoyos, J. Aguilar, M. Toro, A Clinical Decision-Support System for Dengue based on Fuzzy Cognitive Maps, Health Care Management Science (2022). doi:10.1007/s10729-022-09611-6.

E. Puerto, J. Aguilar, C. López, D. Chávez, Using Multilayer Fuzzy Cognitive Maps to diagnose Autism Spectrum Disorder, Applied Soft Computing Journal 75 (2019) 58–71. doi:10.1016/j.asoc.2018.10.034.

V. K. Mago, R. Mehta, R. Woolrych, E. I. Papageorgiou, Supporting meningitis diagnosis amongst infants and children through the use of fuzzy cognitive mapping, BMC Medical Informatics and Decision Making 12 (2012) 1–12. doi:10.1186/1472-6947-12-98.

E. I. Papageorgiou, N. Papandrianos, G. Karagianni, G. Kyriazopoulos, D. Sfyras, Fuzzy cognitive map based approach for assessing pulmonary infections, Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics) 5722 LNAI (2009) 109–118. doi:10.1007/978-3-642-04125-9_14.

W. Hoyos, J. Aguilar, M. Toro, An autonomous cycle of data analysis tasks for the clinical management of dengue, Heliyon 8 (2022) e10846. doi:10.1016/j.heliyon.2022.e10846.

J. Lopes, T. Guimarães, M. F. Santos, Predictive and Prescriptive Analytics in Healthcare: A Survey, Procedia Computer Science 170 (2020) 1029–1034. doi:10.1016/j.procs.2020.03.078.

W. Hoyos, J. Aguilar, M. Toro, Dengue Models Based on Machine-Learning Techniques: A Systematic Literature Review, Artificial Intelligence in Medicine 119 (2021) 102157. doi:10.1016/j.artmed.2021.102157.

L. Berk, D. Bertsimas, A. M. Weinstein, J. Yan, Prescriptive analytics for human resource planning in the professional services industry, European Journal of Operational Research 272 (2019) 636–641. doi:10.1016/j.ejor.2018.

06.035.

S. Dey, N. Gupta, S. Pathak, D. H. Kela, S. Datta, Data-Driven Design Optimization for Industrial Products, Springer, Cham, 2019, pp. 253–267. doi:10.1007/978-3-030-01641-8_9.

M. Ramannavar, N. S. Sidnal, A proposed contextual model for big data analysis using advanced analytics, in: Advances in Intelligent Systems and Computing, volume 654, Springer Verlag, 2018, pp. 329–339. doi:10.1007/978-981-10-6620-7_32.

S. Srinivas, A. R. Ravindran, Optimizing outpatient appointment system using machine learning algorithms and scheduling rules: A prescriptive analytics framework, Expert Systems with Applications 102 (2018) 245–261. doi:10.1016/j.eswa.2018.02.022.

M.-H. Jank, C. Dölle, G. Schuh, Product Portfolio Design Using Prescriptive Analytics, in: Advances in Production Research, Springer, Cham, 2019, pp. 584–593. doi:10.1007/978-3-030-03451-1_57.

P. Revathy, R. Mukesh, HadoopSec 2.0: Prescriptive analytics-based multi-model sensitivity-aware constraints centric block placement strategy for Hadoop, Journal of Intelligent and Fuzzy Systems 39 (2020) 8477–8486. doi:10.3233/JIFS-189165.

D. Singh, A. Shukla, Manifold Optimization with MMSE Hybrid Precoder for Mm-Wave Massive MIMO Communication, Romanian Journal of Information Science and Technology 25 (2022) 36–46.

C. Pozna, R. E. Precup, E. Horvath, E. M. Petriu, Hybrid Particle Filter-Particle Swarm Optimization Algorithm and Application to Fuzzy Controlled Servo Systems, IEEE Transactions on Fuzzy Systems 30 (2022) 4286–4297. doi:10.1109/TFUZZ.2022.3146986.

I. A. Zamfirache, R. E. Precup, R. C. Roman, E. M. Petriu, Policy Iteration Reinforcement Learning-based control using a Grey Wolf Optimizer algorithm, Information Sciences 585 (2022) 162–175. doi:10.1016/j.ins.2021.11.051.

R. Axelrod, Structure of decision: The cognitive maps of political elites, Princeton University Press, 1976.

J. Aguilar, Different dynamic causal relationship approaches for cognitive maps, Applied Soft Computing 13 (2013) 271–282. doi:https://doi.org/10.1016/j.asoc.2012.08.037.

J. Aguilar, J. Contreras, The FCM Designer Tool, Springer Berlin Heidelberg, Berlin, Heidelberg, 2010, pp. 71–87. doi:10.1007/978-3-642-03220-2_4.

J. Kennedy, R. Eberhart, Particle swarm optimization, in: Proceedings of ICNN'95 - International Conference on Neural Networks, volume 4, 1995, pp. 1942–1948 vol.4. doi:10.1109/ICNN.1995.488968.

J. L. Salmeron, S. A. Rahimi, A. M. Navali, A. Sadeghpour, Medical diagnosis of Rheumatoid Arthritis using data driven PSO–FCM with scarce datasets, Neurocomputing 232 (2017) 104–112. doi:10.1016/j.neucom.2016.09.113.

E. I. Papageorgiou, A new methodology for Decisions in Medical Informatics using fuzzy cognitive maps based on fuzzy rule-extraction techniques, Applied Soft Computing Journal 11 (2011) 500–513. doi:10.1016/j.asoc.

2009.12.010.

P. Cortez, A. Cerdeira, F. Almeida, T. Matos, J. Reis, Modeling wine preferences by data mining from physicochemical properties, Decision Support Systems 47 (2009) 547–553. doi:10.1016/j.dss.2009.05.016.

J. J. Pangaribuan, Suharjito, Diagnosis of diabetes mellitus using extreme learning machine, in: Information Technology Systems and Innovation (ICITSI), 2014 International Conference on, 2014, pp. 33–38. doi:10.1109/ICITSI.2014.7048234.

J. Smith, J. Everhart, W. Dickson, R. Johannes, Using the ADAP Learning Algorithm to Forecast the Onset of Diabetes Mellitus, in: Symposium on Computers Aplications in Medical Care, 1988, pp. 261–265.

N. V. Chawla, K. W. Bowyer, L. O. Hall, W. P. Kegelmeyer, SMOTE: Synthetic minority over-sampling technique, Journal of Artificial Intelligence Research 16 (2002) 321–357. doi:10.1613/jair.953. arXiv:1106.1813.

B. Kumar, S. Pal, Mining educational data to analyze students' performance, International Journal of Advanced Computer Science and Applications 2 (2011) 63–69.

E. A. Amrieh, T. Hamtini, I. Aljarah, Mining Educational Data to Predict Student's academic Performance using Ensemble Methods, International Journal of Database Theory and Application 9 (2016) 119–136. doi:10.14257/ijdta.2016.9.8.13.

S. Kumar, K. Agrawal, N. Mandan, Red wine quality prediction using machine learning techniques, in: 2020 International Conference on Computer Communication and Informatics, ICCCI 2020, Institute of Electrical and Electronics Engineers Inc., 2020. doi:10.1109/ICCCI48352.2020.9104095.

A. Laughter, S. Omari, A Study of Modeling Techniques for Prediction of Wine Quality, in: Advances in Intelligent Systems and Computing, volume 1228 AISC, Springer, 2020, pp. 373–399. doi:10.1007/978-3-030-52249-0_27.

I. Frank, B. R. Kowalski, prediction of wine quality and geographic origin from chemical measurements by parital least-squares regression modeling, Analytica Chimica Acta 162 (1984) 241–251. doi:10.1016/S0003-2670(00)84245-2.

C. C. Olisah, L. Smith, M. Smith, Diabetes mellitus prediction and diagnosis from a data preprocessing and machine learning perspective, Computer Methods and Programs in Biomedicine 220 (2022) 106773. doi:10.1016/j.cmpb.2022.106773.

M. K. Hasan, M. A. Alam, D. Das, E. Hossain, M. Hasan, Diabetes prediction using ensembling of different machine learning classifiers, IEEE Access 8 (2020) 76516–76531. doi:10.1109/ACCESS.2020.2989857.

T. L. Elliott, K. M. Pfotenhauer, Classification and Diagnosis of Diabetes, volume 49, Elsevier, 2022. doi:10.1016/j.pop.2021.11.011.

A. Rácz, D. Bajusz, K. Héberger, Effect of Dataset Size and Train/Test Split Ratios in QSAR/QSPR Multiclass Classification, Molecules 26 (2021). doi:10.3390/MOLECULES26041111.

J. G. A. Barbedo, Impact of dataset size and variety on the effectiveness of deep learning and transfer learning for plant disease classification, Computers and Electronics in Agriculture 153 (2018) 46–53. doi:10.1016/J.COMPAG.2018.08.013.

Y. Quintero, D. Ardila, E. Camargo, F. Rivas, J. Aguilar, Machine learning models for the prediction of the seird variables for the covid-19 pandemic based on a deep dependence analysis of variables, Computers in Biology and Medicine 134 (2021) 104500. doi:10.1016/j.compbiomed.2021.104500.

E. A. Amrieh, T. Hamtini, I. Aljarah, Preprocessing and analyzing educational data set using X-API for improving student's performance, in: 2015 IEEE Jordan Conference on Applied Electrical Engineering and Computing Technologies, AEECT 2015, Institute of Electrical and Electronics Engineers Inc., 2015. doi:10.1109/AEECT.2015.7360581.

G. W. H. Tan, K. B. Ooi, L. Y. Leong, B. Lin, Predicting the drivers of behavioral intention to use mobile learning: A hybrid SEM-Neural Networks approach, Computers in Human Behavior 36 (2014) 198–213. doi:10.1016/j.chb.2014.03.052.

H. W. Lilliefors, On the Kolmogorov-Smirnov Test for Normality with Mean and Variance Unknown, Journal of the American Statistical Association 62 (1967) 399–402. doi:10.1080/01621459.1967.10482916.

K. Amandeep, K. Robin, Comparative Analysis of Parametric and Non-Parametric Tests, Journal of Computer and Mathematical Sciences 6 (2015) 336–342. URL: http://www.compmath-journal.org/dnload/Amandeep-Kaur-and-Robin-Kumar/CMJV06I06P0336.pdf.

V. Grech, N. Calleja, WASP (Write a Scientific Paper): Parametric vs. non-parametric tests, Early Human Development 123 (2018) 48–49. doi:10.1016/j.earlhumdev.2018.04.014.

W. Hoyos, Supplementary data for the article entitled "PRV-FCM: an extension of fuzzy cognitive maps for prescriptive modeling", 2023. doi:10.17632/sz79bzd8mf.3.

H. Harikumar, S. Rana, S. Gupta, T. Nguyen, R. Kaimal, S. Venkatesh, Prescriptive analytics with differential privacy, International Journal of Data Science and Analytics 13 (2022) 123–138. doi:10.1007/s41060-021-00286-w.

Y. He, Application study in decision support with fuzzy cognitive map, in: Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), volume 5178 LNAI, Springer, Berlin, Heidelberg, 2008, pp. 324–331. doi:10.1007/978-3-540-85565-1_41.

E. Chalmers, D. Hill, V. Zhao, E. Lou, Prescriptive analytics applied to brace treatment for AIS: A pilot demonstration, Scoliosis 10 (2015) 2–5. doi:10.1186/1748-7161-10-S2-S13.