Explaining Computer Predictions with Augmented Appraisal Degrees

Marcelo Loor[a,b] **and Guy De Tré**[a]
[a]Dept. of Telecommunications and Information Processing, Ghent University,
Sint-Pietersnieuwstraat 41, B-9000, Ghent, Belgium, {Marcelo.Loor, Guy.DeTre}@UGent.be
[b]Dept. of Electrical and Computer Engineering, ESPOL Polytechnic University,
Campus Gustavo Galindo V., Km. 30.5 Via Perimetral, Guayaquil, Ecuador

Abstract

An *augmented appraisal degree* (AAD) has been conceived as a mathematical representation of the connotative meaning in an *experience-based evaluation*, which depends on a particular experience or knowledge. Aiming to improve the interpretability of computer predictions, we explore the use of AADs to represent evaluations that are performed by a machine to predict the class of a particular object. Hence, we propose a novel method whereby predictions made using a *support vector machine* classification process are augmented through AADs. An illustrative example, in which the classes of handwritten digits are predicted, shows how the augmentation of such predictions can favor their interpretability.

Keywords: Explainable artificial intelligence, Augmented appraisal degrees, Augmented fuzzy sets, Support vector machines

1 Introduction

As the use of *artificial intelligence* (AI) for business or user needs increases, the demand for transparency and interpretability on its predictions is also growing [8]. Such demand is mainly created because AI involves complex techniques and algorithms that, in most of the cases, do not offer explanations for their outcomes and, thus, the reasons behind computer predictions might remain unknown [15, 17].

As an example in which the demand for transparency and interpretability could be strong, consider a system that uses *support vector machines* (SVMs) [18, 19] to classify animal diseases according to the common or distinctive characteristics that those diseases may have. Although the system uses the characteristics of a disease to predict the class it belongs to, the system only offers the predicted class as output. In this case, a decision maker, say a veterinarian, might be reluctant to make a key decision, say prescription of antibiotics, based on a computer prediction without knowing what has been relevant to support that prediction. A challenge in this regard is, *how can the reason of such a prediction be explained?*

To address that challenge, in this paper we explore the use of *augmented appraisal degrees* [11] (AADs) for the characterization of evaluations that are performed by a computer to predict the class of an object. By means of an AAD a computer can record not only the level to which an object belongs (or not) to a particular class, but also several hints that justify that level. Hence, we propose a novel method by which predictions offered by a computer during a SVM classification process are augmented with AADs to make such predictions (better) interpretable.

To illustrate how the proposed method works, we describe an example in which the class of a handwritten number is predicted by a classification process that uses SVMs. The example's idea is depicted in Figure 1: while Figure 1(a) shows a handwritten number, which is used as input, Figure 1(b) shows a representation of why the proposition "the handwritten number is a '6' " is true up to a specific level. Along with the visual representation, the proposed method can return the following output: "The green part suggests that the drawing is a '6' with a computed grade of 0.23; yet, the red part, which a '6' should have, and the gray part, which a '6' should not have, indicate that it is not a '6' with a computed grade of 0.37."

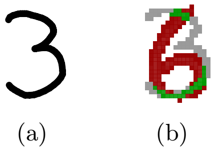


Figure 1: Predicting handwritten numbers.

As could be noticed in the above illustration, the output not only indicates why a proposition (or prediction) is true, but also why it is not. This important aspect illustrates why AADs have been integrated into the *intuitionistic fuzzy set* (IFS) [3, 4] concept.

In the next section, we present the AAD concept and briefly explain how it can be integrated into the IFS concept. Then, we explain our novel method to augment computer predictions with AADs in Section 3 and illustrate how to use it in Section 4. After that, other existing techniques for explaining individual predictions are reviewed in Section 5. Finally, we conclude the paper in Section 6.

2 Preliminaries

A classification is commonly understood as a process in which several objects are arranged in (usually well-known) classes (or categories) according to the common or distinctive characteristics that those objects may have. Hence, to predict the class of an object, a classification algorithm can first look into the features of the object. Based on this, it can evaluate the level to which the object is part of each of the well-known classes. Finally it can provide the best evaluated class(es) as an answer.

In the framework of *fuzzy set theory* [20], an evaluation of the level to which an object, say x, belongs to a class, say A, can be denoted by a *membership grade*, which is a number $\mu_A(x)$ in the unit interval $[0, 1]$. For instance, if x represents the handwritten number shown in Figure 1(a) and A denotes (what has been learned about) the class of handwritten 6's, then $\mu_A(x) = 0.23$ indicates the level to which x is member of A. It is worth mentioning that, after representing two evaluations by membership grades, a numeric comparison can be used to compare them. Thus, e.g., $\mu_A(x) < \mu_B(x)$ means that the level to which x belongs to A is less than the level to which x belongs to B – in this case, B would be the best evaluated class of x if the collection of well-known classes is only constituted by A and B.

In some situations, an object can also have features suggesting it does not belong to a given class – see, e.g., Figure 1(b) in which the gray and the red parts suggest the handwritten number is not a '6'. In such situations, the evaluation of an object x can be better described in the *intuitionistic fuzzy set* (IFS) framework [3, 4]. In this framework, an evaluation is denoted by two numbers in the unit interval $[0, 1]$: a *membership grade*, say $\mu_A(x)$, and a *nonmembership grade*, say $\nu_A(x)$. These numbers, which must satisfy the consistency condition $0 \leq \mu_A(x) + \nu_A(x) \leq 1$, together constitute an IFS element $\langle x, \mu_A(x), \nu_A(x)\rangle$. For example, the evaluation of the proposition "the handwritten number depicted in Figure 1(a) is a '6' " can be denoted by $\langle x, 0.23, 0.37\rangle$. Regarding the comparison of two evaluations characterized by two IFS elements, say $\langle x, \mu_A(x), \nu_A(x)\rangle$ and $\langle x, \mu_B(x), \nu_B(x)\rangle$, one can first compute the *buoyancy* of each IFS element, i.e., $\rho_A(x) = \mu_A(x) - \nu_A(x)$ and $\rho_B(x) = \mu_B(x) - \nu_B(x)$ respectively [13] and then compare the resulting values. Thus, e.g., $\rho_A(x) > \rho_B(x)$ suggests that x belongs more to A than to B.

While a membership grade and an IFS element make it possible to record the level(s) to which an object belongs (or not) to a given class, none of these representations enables the recording of the object's characteristics that lead to and hence explain this (these) level(s). To do so, the idea of *augmented appraisal degrees* (AADs) has been introduced in [11]. An AAD of an object x, say $\hat{\mu}_{A@K}(x)$, can be seen as a pair $\langle \mu_{A@K}(x), F_{\mu_{A@K}}(x)\rangle$ that denotes the level $\mu_{A@K}(x)$ to which x belongs to the class A, as well as the particular collection of x's features $F_{\mu_{A@K}}(x)$ considered to evaluate x according to the knowledge K. For instance, the evaluation depicted in Figure 1(b) can be denoted by $\langle 0.23, F_{\mu_{A@K}}(x)\rangle$, where: (i) x and A represent, in that order, the handwritten number in Figure 1(a) and a class of handwritten 6's; (ii) K symbolizes the knowledge about handwritten 6's used to evaluate x; and (iii) $F_{\mu_{A@K}}(x)$ represents a collection consisting of the green pixels that indicate why x should be a '6' according to K[1].

As previously stated, there are situations where an object can have features suggesting it does not belong to a given class. To handle this kind of situations, the augmentation of IFS elements with AADs has been proposed in [11]. An augmented IFS element, say $\langle x, \hat{\mu}_{A@K}(x), \hat{\nu}_{A@K}(x)\rangle$, consists of a membership AAD, $\hat{\mu}_{A@K}(x)$, and a nonmembership AAD, $\hat{\nu}_{A@K}(x)$: while $\hat{\mu}_{A@K}(x)$ indicates the level to which x belongs to A and the features of x considered for quantifying this *membership* level, $\hat{\nu}_{A@K}(x)$ indicates the level to which x does not belong to A and the features of x considered for quantifying this *nonmembership* level. For instance, keeping x, A, K and $F_{\mu_{A@K}}(x)$ as given in the previous example, one can represent the evaluation depicted in Figure 1(b) by $\langle\langle 0.23, F_{\mu_{A@K}}(x)\rangle, \langle 0.37, F_{\nu_{A@K}}(x)\rangle\rangle$, where $F_{\nu_{A@K}}(x)$ represents a collection consisting of the red and the gray pixels that indicate why x should not be a '6' according to K. In the next section, we explain how to use these concepts to make predictions in artificial intelligence better interpretable.

[1] In this example, one can also say that $A@K$ represents what has been learned about a class of handwritten 6's after following a learning process that yields K as a result.

3 Augmenting Predictions with AADs

As was mentioned earlier, our aim is the augmentation of predictions in artificial intelligence to make them better interpretable. For that purpose, in this section we describe a novel method by which predictions made by a classification process that uses SVMs are augmented by AADs.

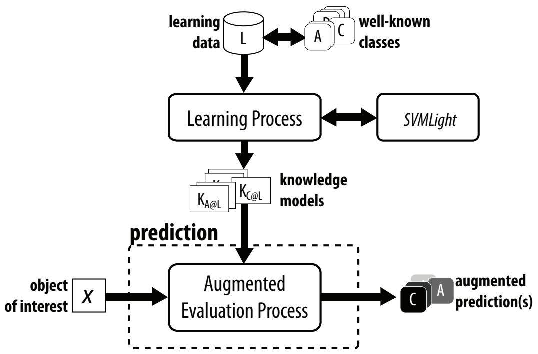


Figure 2: A contextual view of the proposed method.

A contextual view of the proposed prediction method is shown in Figure 2. As can be noticed, one of the inputs of the method is a collection of knowledge models that result after following a learning process, which uses SVMs to learn about a set of well-known classes. In this section we first explain how such knowledge models can be obtained through the learning process proposed in [12]. After that, we explain how the prediction of the class(es) of an object can be made better interpretable by augmenting the evaluations that are performed to determine if this object belongs or not to each of the well-known classes.

3.1 Obtaining knowledge models

The learning process proposed in [12] mimics a learning behavior where a person learns about a class by studying the features of the objects in a training collection. The process is based on a *feature-influence* representational model, which is summarized below.

Consider a m-dimensional feature space $\mathcal{M}$ in which each dimension corresponds to a feature f_j in a collection $\mathcal{F} = \{f_1, \cdots, f_m\}$. Also consider an object x with a collection of features $\mathcal{F}_x \subseteq \mathcal{F}$. Let p_A be a proposition having the canonical form 'x IS A' meaning x is member of the class A. Under these considerations, the influence of the features of x on the appraisal of p_A is modeled as follows:

- The *overall influence* $\mathbf{x}$ of the features of x on the classification is given by the vector

$$\mathbf{x} = \sum_{j=1}^{m} \beta_j \hat{\mathbf{f}}_j, \qquad (1)$$

  where β_j denotes the *overall importance* (or weight) on the classification of f_j among the features in $\mathcal{F}$, and $\hat{\mathbf{f}}_j$ is the unit vector representing the dimension related to f_j in $\mathcal{M}$.

- A particular knowledge about A, say K_A, is represented by a line in $\mathcal{M}$ and it is described by a pair $\langle \hat{\mathbf{u}}_A, t_A \rangle$. In this pair, while $\hat{\mathbf{u}}_A$ represents a unit vector that points to a place in $\mathcal{M}$ where the fulfillment of p_A is favored, t_A is a point on the line (defined by $\hat{\mathbf{u}}_A$) that identifies a location in $\mathcal{M}$ where the fulfillment of p_A is neither favored nor disfavored.

- The *specific influence* of the features of x on the appraisal of p_A is represented by the vector

$$\mathbf{x}_A = (\mathbf{x} \cdot \hat{\mathbf{u}}_A)\hat{\mathbf{u}}_A = \sum_{j=1}^{m} \beta_{j_A} \hat{\mathbf{u}}_A, \qquad (2)$$

  where $\beta_{j_A} \hat{\mathbf{u}}_A$ denotes the *specific influence* of f_j on the appraisal of p_A, and '$\cdot$' denotes a dot product. As noticed, this vector corresponds to the *vector projection* of the overall influence vector, i.e., $\mathbf{x}$, on $\hat{\mathbf{u}}_A$, i.e., the line that represents K_A.

- The *level to which* x satisfies (or dissatisfies) p_A is determined by the magnitude of the vector

$$\mathbf{l}_A = \mathbf{x}_A - t_A \hat{\mathbf{u}}_A, \qquad (3)$$

  i.e, it is determined by

$$||\mathbf{l}_A|| = \sqrt{\mathbf{l}_A \cdot \mathbf{l}_A}. \qquad (4)$$

  If the directions of $\mathbf{l}_A$ and $\hat{\mathbf{u}}_A$ are the same, x satisfies p_A to the extent $||\mathbf{l}_A||$. By the contrary, if the direction of $\mathbf{l}_A$ is opposite to the direction of $\hat{\mathbf{u}}_{,A}$, x dissatisfies p_A to the extent $||\mathbf{l}_A||$.

To extract a model of the knowledge about a class A, say[2] $K_A = \langle \hat{\mathbf{u}}_A, t_A \rangle$, from a training collection, say $X_0 = \{x_1, \cdots, x_n\}$, a computer can follow the steps of the aforementioned learning process and study the features of each object $x_i \in X_0$. The main steps are given below – the interested reader is referred to [12] for a detailed description of this learning process:

[2] To be consistent with the notation introduced in Figure 2 where the *"source"* of the knowledge about A is explicitly denoted, we should say $K_{A@X_0} = \langle \hat{\mathbf{u}}_{A@X_0}, t_{A@X_0} \rangle$. For the sake of readability we use hereafter this simplified form of the notation.

1. For each $x_i \in X_0$, identify its features and put them into $\mathcal{F}$.

2. Assign an overall importance $\beta_{i,j}$ to each feature $f_j \in \mathcal{F}$ based on its overall influence on the appraisal of p_A for each $x_i \in X_0$.

3. Compute $\langle \hat{\mathbf{u}}_A, t_A \rangle$ in such a way that (i) the correspondence between each $x_i \in X_0$ satisfying or dissatisfying p_A and the resulting specific influence of its features is preserved, and (ii) both the aggregate of the specific influences that favor the fulfillment of p_A and the aggregate of the specific influences that disfavor such fulfillment are maximized.

Due to conditions (i) and (ii) in third step of the learning procedure, the computation of $\langle \hat{\mathbf{u}}_A, t_A \rangle$ can be done based on the *separable case* of a linear SVM [18, 19]. Indeed, considering the equations

$$\hat{\mathbf{u}}_A = \frac{\mathbf{w}}{||\mathbf{w}||} \tag{5}$$

and

$$t_A = -\frac{b}{||\mathbf{w}||}, \tag{6}$$

a hyperplane $\mathbf{w} + b$ has to be determined in such a way that the gap between the vectors corresponding to *positive examples* and the vectors corresponding to *negative examples* is maximized – herein, a positive example is an object $x_i \in X_0$ that satisfies p_A, whereas a negative example is an object that dissatisfies p_A.

The values of $\mathbf{w}$ and b can be computed by the Lagrangian formulation of the separable case [7], in which the value of Λ, given by equation

$$\Lambda = \frac{1}{2}\|\mathbf{w}\|^2 - \sum_{i=1}^{n} \lambda_i \left(y_i(\mathbf{w} \cdot \mathbf{x}_i + b) - 1\right), \tag{7}$$

should be minimized subject to $y_i(\mathbf{w} \cdot \mathbf{x_i} + b) - 1 \geq 0$ and $(\forall \lambda_i \in \{\lambda_1, \cdots, \lambda_n\})(\lambda_i > 0)$. In (7), while $\mathbf{x_i}$ denotes a vector related to an object x_i in X_0, $y_i \in \{-1, 1\}$ is a label that indicates whether x_i is a positive example ($y_i = 1$) or a negative example ($y_i = -1$). To find the values of $\mathbf{w}$, b and all $\lambda_i \in \{\lambda_1, \cdots, \lambda_n\}$, the software package *SVMLight* [10] can be used.

3.2 Augmenting evaluations

After obtaining a model of the knowledge about a particular class, a classification algorithm can use that model to evaluate the level to which an object is member of that class. For instance, a classification algorithm can use $K_A = \langle \hat{\mathbf{u}}_A, t_A \rangle$, which represents a model of the knowledge about (class) A, to evaluate, by means of (3) and (4), the level to which an object x is member of A. Likewise, the algorithm can use $K_B = \langle \hat{\mathbf{u}}_B, t_B \rangle$, which represents a model of the knowledge about B, to evaluate the level to which x is a member of B. Next, the resulting levels can be used to make a prediction about the class of x: if the level to which x is member of A, i.e., $||\mathbf{l}_A||$, is greater than the level to which x is member of B, i.e., $||\mathbf{l}_B||$, A can be returned as the predicted class of x.

If a user likes to know in the previous example why the predicted class is A, the conventional classification algorithm is limited to offer an answer like "x is more A than B because $||\mathbf{l}_A|| > ||\mathbf{l}_B||$. Notice in this answer that nothing is mentioned about the *relevant features* of x that support that prediction.

To make such a prediction available for interpretation, the result of an evaluation can be augmented as explained below.

Consider a class A, an object x and a proposition p_A having the canonical form 'x IS A', which means "x is a member of A." Assume that $K_A = \langle \hat{\mathbf{u}}_A, t_A \rangle$ is a representation of a particular knowledge about A as stated by the feature-influence representation model presented in Section 3.1. Assume also that $\hat{\mathbf{u}}_A = \omega_1 \hat{\mathbf{f}}_1 + \cdots + \omega_m \hat{\mathbf{f}}_m$ and $\mathbf{x} = \beta_1 \hat{\mathbf{f}}_1 + \cdots + \beta_m \hat{\mathbf{f}}_m$ respectively represent the directional vector and the overall influence vector. Under these considerations, a procedure for performing an augmented evaluation of p_A that yields an augmented IFS element $\langle \hat{\mu}_A(x), \hat{\nu}_A(x) \rangle = \langle \langle \mu_A(x), F_{\mu_A}(x) \rangle, \langle \nu_A(x), F_{\nu_A}(x) \rangle \rangle$ (see Section 2) as a result, consists of the following steps:

1. For each $f_j \in \mathcal{F}_x$, i.e., for each of the features of x, compute its *specific influence* on the appraisal of p_A, i.e., compute $\mathbf{f}_{j_A} = \beta_{j\,A} \hat{\mathbf{u}}_A = \beta_j \omega_j \hat{\mathbf{u}}_A$. Include f_j in $F_{\mu_A}(x)$ if $\beta_{j\,A} > 0$; otherwise include f_j into $F_{\nu_A}(x)$ if $\beta_{j\,A} < 0$.

2. Compute $\mu_A(x_i)$ and $\nu_A(x_i)$ by means of the equations

$$\mu_A(x) = \check{\mu}_A(x)/\eta \tag{8}$$

and

$$\nu_A(x) = \check{\nu}_A(x)/\eta \tag{9}$$

respectively, where

$$\check{\mu}_A(x) = \begin{cases} \frac{|t_A| + \sum_{j=1}^{m} \beta_{j\,A}}{\|\mathbf{x}\|} & : \left(\forall \beta_{j\,A} > 0\right) \wedge (t_A < 0)\,; \\ \frac{\sum_{j=1}^{m} \beta_{j\,A}}{\|\mathbf{x}\|} & : \left(\forall \beta_{j\,A} > 0\right) \wedge (t_A \geq 0)\,; \\ 0 & : \text{otherwise;} \end{cases} \tag{10}$$

$$\check{\nu}_A(x) = \begin{cases} \frac{t_A + \sum_{j=1}^{m} |\beta_{j\,A}|}{\|\mathbf{x}\|} & : \left(\forall \beta_{j\,A} < 0\right) \wedge (t_A > 0) \\ \frac{\sum_{j=1}^{m} |\beta_{j\,A}|}{\|\mathbf{x}\|} & : \left(\forall \beta_{j\,A} < 0\right) \wedge (t_A \leq 0)\,; \\ 0 & : \text{otherwise;} \end{cases} \tag{11}$$

and

$$\eta = \max\left(1, \check{\mu}_A(x) + \check{\nu}_A(x)\right). \quad (12)$$

An interpretable classification algorithm can use the above procedure to perform an augmented evaluation of the membership (or nonmembership) of an object in a given class. For instance, to evaluate the membership of an object, say x, in a class, say A, the algorithm can use the procedure with a model of the knowledge about A, say $K_A = \langle \hat{\mathbf{u}}_A, t_A \rangle$, to obtain $\langle \hat{\mu}_A(x), \hat{\nu}_A(x) \rangle$ as a result. In a similar way, the algorithm can use the procedure with the knowledge model about another class, say $K_B = \langle \hat{\mathbf{u}}_B, t_B \rangle$, to evaluate the membership of x in (class) B and, so, obtain $\langle \hat{\mu}_B(x), \hat{\nu}_B(x) \rangle$.

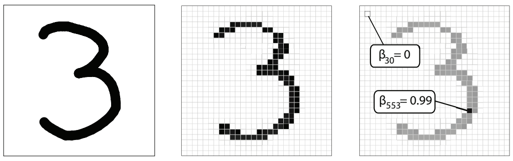
After that, the algorithm can use those evaluations to predict whether the class of x is A or B: if the buoyancy of $\langle \hat{\mu}_A(x), \hat{\nu}_A(x) \rangle$, i.e., $\rho_A(x) = \mu_A(x) - \nu_A(x)$, (see Section 2) is greater than the buoyancy of $\langle \hat{\mu}_B(x), \hat{\nu}_B(x) \rangle$, i.e., $\rho_B(x) = \mu_B(x) - \nu_B(x)$, the predicted class will be A. In this case, if a user asks why the predicted class of x is A, the algorithm might offer an answer like this: "the features in $F_{\mu_A}(x)$ suggest that x is A with a grade of $\mu_A(x)$; however, the features in $F_{\nu_A}(x)$ indicate that x is not A with a grade of $\nu_A(x)$."

Notice in the previous answer that, by means of AADs, a prediction can be augmented with contextual information that makes the prediction interpretable. Thus, the user who asked why the predicted class of x is A can make a more informed decision with that prediction. In addition, this user can have an idea about the quality of both the prediction and the model behind it.

In the next section, we describe an example where predictions about the classes of handwritten numbers are augmented to make those predictions better interpretable.

4 Illustrative Example

Aiming to show how the characterization of an evaluation by means of an AAD can favor the interpretability of computer predictions, in this section we present an example where the classes of handwritten digits are predicted.



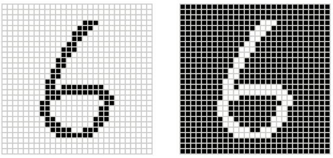
(a) (b) (c)

Figure 3: Characterization of a handwritten '3'.

As was mentioned in the previous section, a prediction of the class of an object can be made after evaluating the membership (and nonmembership) of this object in each of the well-known classes. Thus, to predict the class of a handwritten digit, first an algorithm needs to learn about handwritten numbers such as handwritten 1's or handwritten 2's. To do so, the algorithm can use a training collection consisting of digitized handwritten numbers like the one depicted in Figure 3(b), which corresponds to the handwritten number represented in Figure 3(a).

In this example, a digitized handwritten number consists of 784 pixels, each of them representing a feature of the handwritten number. Each of those 784 pixels has associated a value between 0 and 1, where 0 and 1 denote, in that order, no strength and the maximum strength of a pen while handwriting on that pixel.

Under that setting and according to the feature-influence representational model (see Section 3.1), the influence of the pixels of a digitized handwritten number, say x, is represented in a 784-dimensional feature space $\mathcal{M}$ by a vector $\mathbf{x} = \beta_1 \hat{\mathbf{f}}_1 + \cdots + \beta_{784} \hat{\mathbf{f}}_{784}$, such that β_j corresponds to the strength of the pen in pixel f_j. For instance, while in Figure 3(c) the value of β_{30} is 0 since no strength has been put on pixel f_{30}, the value of β_{553} is 0.99 since the strength of the pen in this pixel is almost the maximum.



(a) (b)

Figure 4: Complement of a handwritten '6'.

A training collection consisting of 50 digitized (and feature-influence vectorized) handwritten numbers (5 per each decimal digit) accompanied by their corresponding complements was built with the purpose of obtaining a knowledge model for each handwritten decimal digit. For instance, the handwritten '6' depicted in Figure 4(a) and its complement, which is depicted in Figure 4(b), are included in the training collection. In this case, while the group of handwritten 6's constitute the positive examples, the complement of this group along with the other handwritten numbers constitute the negative examples.

During the learning process, the training collection was used as input for the process described in Section 3.1 to obtain the knowledge models. Then, during the evaluation process, the resulting models were used as input for the process described in Section 3.2 to evaluate both the membership and nonmembership of the above-mentioned handwritten ‘3’ (see Figure 3(a)) in each of the classes of handwritten decimal digits.

To predict the class of the handwritten ‘3’, a feature-influence vector of its complement was used for the evaluation of the influence of the features that a handwritten number should have, but the handwritten ‘3’ does not have. More specifically, if $\bar{x}$ denotes the complement of x, in this example the evaluation of the membership and nonmembership of x in a particular class, say A, is given by and augmented IFS element $\langle \hat{\mu}_A(x), \hat{\nu}_A(x)\rangle$ such that

$$\hat{\mu}_A(x) = \langle \mu_A(x), F_{\mu_A}(x)\rangle \tag{13}$$

and

$$\hat{\nu}_A(x) = \langle \frac{\nu_A(x) + \mu_A(\bar{x})}{\max(1, \nu_A(x) + \mu_A(\bar{x}))}, F_{\nu_A}(x) \cup F_{\mu_A}(\bar{x})\rangle. \tag{14}$$

As a consequence, the buoyancy of $\langle \hat{\mu}_A(x), \hat{\nu}_A(x)\rangle$ (see Section 2) is given by

$$\rho_A(x) = \mu_A(x) - \frac{\nu_A(x) + \mu_A(\bar{x})}{\max(1, \nu_A(x) + \mu_A(\bar{x}))}. \tag{15}$$

The results of our experimental evaluations are shown in Table 1 and Figure 5.

A	$\mu_A(x)$	$\frac{\nu_A(x)+\mu_A(\bar{x})}{\max(1,\nu_A(x)+\mu_A(\bar{x}))}$	$\rho_A(x)$
‘0’	0.40	0.32	0.08
‘1’	0.24	0.32	-0.08
‘2’	0.39	0.28	0.11
‘3’	**0.59**	**0.14**	**0.45**
‘4’	0.18	0.39	-0.21
‘5’	0.43	0.27	0.16
‘6’	0.23	0.37	-0.14
‘7’	0.26	0.32	-0.06
‘8’	0.42	0.30	0.12
‘9’	0.25	0.35	-0.10

Table 1: Results of the evaluations of the membership and nonmembership of a handwritten ‘3’ in each of the classes of handwritten decimal digits.

An interpretable classification algorithm can use those results to offer an explanation like the following: “*The green part* (which is obtained from $F_{\mu_A}(x)$) *suggests that your drawing is a ‘3’* (which is obtained from A) *with a grade of* 0.59 (which is obtained from $\mu_A(x)$); *however, the red part* (which is obtained from $F_{\nu_A}(x)$), *which a ‘3’ should have, and the gray part* (which is obtained from $F_{\mu_A}(\bar{x})$), *which a ‘3’ should not have, indicate that it is not a ‘3’ with a grade of* 0.14 (which is obtained from $\frac{\nu_A(x)+\mu_A(\bar{x})}{\max(1,\nu_A(x)+\mu_A(\bar{x}))}$).

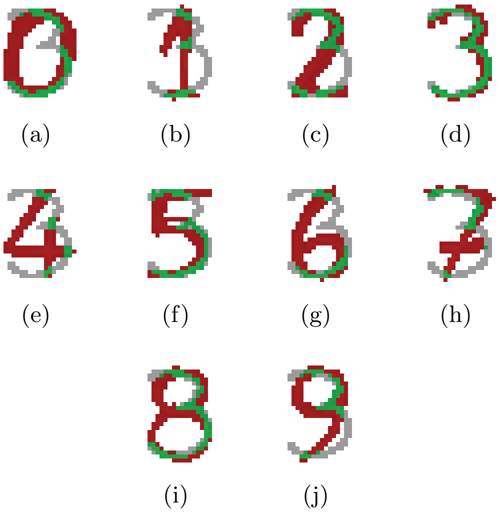


Figure 5: Visual results of the evaluations of the membership and nonmembership of a handwritten ‘3’ in each of the classes of handwritten decimal digits.

Notice that not only the predicted class but also the reasons behind that prediction are given – such an explanation can be used, e.g., by someone who is trying to improve the predictions of models that are trained with limited handwritten numbers. Because of this, it is argued in this paper that the augmentation of such predictions can favor the interpretability of them. Even so, qualitative attributes like coherence, clearness, credibility, relevance or naturalness that might be perceived on such augmented predictions are still subject to validation. In this regard, studies aiming to perform such validations are considered (and suggested) as future work.

5 Related Work

In the literature, one can find methods that, like ours, are oriented to explain individual predictions by decomposing the classification decision in terms of the features of the object that are relevant to that decision. For instance, in [5] a method for decomposing a nonlinear image classification decision has been proposed. Through that method, a computer can produce a heat map that highlights the relevant pixels, i.e., the pixels that have a significant influence on the classification decision. Another example related to computer vision is the visualization method proposed in [21], which offers an interpretation of the influence of the features (pixels) and the behavior of the model. A peculiarity about those methods is that the influence of the features of an object is determined after the prediction of its class. In contrast, our method computes the influence of the features before the prediction. This aspect constitutes an advantage as the influence of the features can be taken into account to guide the classification decision.

Other methods are oriented to explain individual predictions by building interpretable local models that mimic the behavior of unknown classifiers. In one of such methods a prediction is explained after extracting an interpretable local model from the prediction [16]. To build such a model, the method evaluates samples that are closer to or far away of the object whose class is being predicted. A similar strategy is applied by the method proposed in [6], but in that case, the method tries to extract a local model consisting of "explanation vectors," which contain features that are relevant to a prediction. As noticed, in both methods not only the object whose class is being predicted but other objects are needed to explain a prediction. An advantage of our method in this regard is that only an augmented evaluation of the object of interest is needed.

An explanation model that combines classification and sentence generation has been proposed in [9]. Images with annotated features are used as input to train such a model, which is then used to make predictions accompanied with sentences (explanations) in natural language. Although discriminative features that justify why an object belongs to the predicted class are include in such a sentence, features justifying why the object does not belong the class, as our method does, are omitted.

A study analyzing the contributions made by the fuzzy logic community to the development of the explainable AI research field has been presented in [2]. The results of that study suggest that, although those efforts seem to be distant with the efforts made by the non-fuzzy community, both efforts can be linked to address the challenges arising in that field. In this regard, while potential options in the fuzzy logic community can be found in [14], non-fuzzy options oriented to explain individual predictions can be found in [1].

6 Conclusions

In this paper, we considered the use of *augmented appraisal degrees* (AADs) to improve the interpretability of predictions in artificial intelligence methods and proposed a novel method whereby predictions made by a conventional *support vector machine* (SVM) classification process are augmented with AADs.

In the proposed method, an evaluation of the membership (and nonmembership) of an object in a particular class is augmented in such a way that the object's features that support the evaluation are recorded. Such an augmented evaluation is then used to augment and explain the reasons behind a prediction.

By means of an example where the class of a handwritten number is predicted, we have shown how the characterization of evaluations through AADs can favor the interpretability of computer predictions made during a SVM classification process. Nevertheless, qualitative attributes like coherence or relevance that might be perceived by a person on explanations based on AADs are still subject to validation. In this regard, as future work we plan (and suggest) to perform such validations.

Other planned (and suggested) studies concern (i) the applicability of AADs to augment the predictions made by other classifiers like the ones based on convolutional neural networks or Bayesian networks, and (ii) the use of AADs to improve the reliability of predictions that result from models that are trained with limited data.

References

[1] A. Adadi, M. Berrada, Peeking Inside the Black-Box: A Survey on Explainable Artificial Intelligence (XAI), IEEE Access 6 (2018) 52138–52160.

[2] J. M. Alonso, C. Castiello, C. Mencar, A bibliometric analysis of the explainable artificial intelligence research field, in: J. Medina, M. Ojeda-Aciego, J. L. Verdegay, D. A. Pelta, I. P. Cabrera, B. Bouchon-Meunier, R. R. Yager (Eds.), Information Processing and Management of Uncertainty in Knowledge-Based Systems. Theory and Foundations, Springer International Publishing, Cham, 2018, pp. 3–15.

[3] K. T. Atanassov, Intuitionistic fuzzy sets, Fuzzy sets and Systems 20 (1) (1986) 87–96.

[4] K. T. Atanassov, On Intuitionistic Fuzzy Sets Theory, Vol. 283 of Studies in Fuzziness and Soft Computing, Springer Berlin Heidelberg, Berlin, Heidelberg, 2012.

[5] S. Bach, A. Binder, G. Montavon, F. Klauschen, K.-R. Müller, W. Samek, On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation, PLOS ONE 10 (7) (2015) 1–46.

[6] D. Baehrens, T. Schroeter, S. Harmeling, M. Kawanabe, K. Hansen, K.-R. MÃžller, How to explain individual classification decisions, Journal of Machine Learning Research 11 (Jun) (2010) 1803–1831.

[7] C. J. Burges, A tutorial on support vector machines for pattern recognition, Data mining and knowledge discovery 2 (2) (1998) 121–167.

[8] H. Hagras, Toward human-understandable, explainable AI, Computer 51 (9) (2018) 28–36.

[9] L. A. Hendricks, Z. Akata, M. Rohrbach, J. Donahue, B. Schiele, T. Darrell, Generating visual explanations, in: B. Leibe, J. Matas, N. Sebe, M. Welling (Eds.), Computer Vision – ECCV 2016, Springer International Publishing, Cham, 2016, pp. 3–19.

[10] T. Joachims, Making large-scale SVM learning practical, in: B. Schölkopf, C. Burges, A. Smola (Eds.), Advances in Kernel Methods - Support Vector Learning, MIT Press, Cambridge, MA, 1999, Ch. 11, pp. 169–184.

[11] M. Loor, G. De Tré, On the need for augmented appraisal degrees to handle experience-based evaluations, Applied Soft Computing 54 (2017) 284–295.

[12] M. Loor, G. De Tré, Identifying and properly handling context in crowdsourcing, Applied Soft Computing 73 (2018) 203–214.

[13] M. Loor, A. Tapia-Rosero, G. De Tré, Usability of concordance indices in FAST-GDM problems, in: Proceedings of the 10th International Joint Conference on Computational Intelligence (IJCCI 2018), 2018, pp. 67–78.

[14] C. Mencar, J. M. Alonso, Paving the way to explainable artificial intelligence with fuzzy modeling, in: R. Fullér, S. Giove, F. Masulli (Eds.), Fuzzy Logic and Applications, Springer International Publishing, Cham, 2019, pp. 215–227.

[15] A. Preece, Asking 'Why' in AI: Explainability of intelligent systems–perspectives and challenges, Intelligent Systems in Accounting, Finance and Management 25 (2) (2018) 63–72.

[16] M. T. Ribeiro, S. Singh, C. Guestrin, "Why Should I Trust You?": Explaining the Predictions of Any Classifier, in: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '16, ACM, New York, NY, USA, 2016, pp. 1135–1144.

[17] W. Samek, T. Wiegand, K.-R. Müller, Explainable artificial intelligence: Understanding, visualizing and interpreting deep learning models, ITU Journal: ICT Discoveries - Special Issue 1 - The Impact of Artificial Intelligence (AI) on Communication Networks and Services 1 (2017) 39–48.

[18] V. N. Vapnik, The nature of statistical learning theory, Springer-Verlag New York, Inc., 1995.

[19] V. N. Vapnik, V. Vapnik, Statistical learning theory, Vol. 1, Wiley New York, 1998.

[20] L. Zadeh, Fuzzy sets, Information and control 8 (3) (1965) 338–353.

[21] M. D. Zeiler, R. Fergus, Visualizing and understanding convolutional networks, in: European conference on computer vision, Springer, 2014, pp. 818–833.