ECAI 2023
K. Gal et al. (Eds.)

doi:10.3233/FAIA230496

Symbolic Knowledge-Extraction Evaluation Metrics: The FiRe Score

Federico Sabbatini[a;*] **and Roberta Calegari**[b]

[a]University of Urbino
[b]University of Bologna
ORCiD ID: Federico Sabbatini https://orcid.org/0000-0002-0532-6777,
Roberta Calegari https://orcid.org/0000-0003-3794-2942

Abstract. Symbolic knowledge-extraction (SKE) techniques are becoming of key importance for AI applications since they enable the explanation of opaque black-box predictors, enhancing trust and transparency. Among all the available SKE techniques, the best option for the case at hand should be selected. However, an automatic comparison between different options can be performed only if an adequate metric – such as a scoring function resuming all the interesting features of the extractors – is provided. Regrettably, the literature currently lacks definitions of effective evaluation metrics for symbolic knowledge extractors. This paper proposes the novel FiRe score metric, which comprehensively assesses the quality of an SKE procedure by considering both its predictive performance and the readability of the extracted knowledge. FiRe is compared to another existing scoring metric and a rigorous mathematical formulation is provided along with several practical examples to highlight its effectiveness to the end of being exploited inside automatic hyper-parameter tuning procedures.

1 Introduction

One of the main strengths of machine learning (ML) models is their ability to deliver highly accurate predictions across a wide range of application scenarios [21]. Unfortunately, the most powerful ML predictors – such as deep neural networks, for instance – present a high price in terms of human interpretability of their outputs. Indeed, they acquire knowledge during a training phase and store it in a sub-symbolic way, in the form of internal parameters. This common *opaque* behaviour constitutes a real barrier to the exploitation of such models, named *black boxes* (BBs), in critical areas, that are all those real-world applications heavily impacting human lives, e.g., in terms of safety, health, and wealth.

Different solutions have been proposed by the explainable artificial intelligence community to combine human interpretability with the predictive performance of BB models [15]. Amongst the strategies available in the literature there is the choice of intrinsic explainable models [22], such as decision trees with a limited amount of internal nodes and leaves. When this option is not feasible or does not provide satisfying results, a different research branch suggests extracting the BB acquired knowledge by adhering to some *symbolic* representation, through a reverse-engineering of the BB behaviour [19]. This second strategy is the rationale behind SKE procedures. In the years, a plethora of SKE techniques have been proposed in the literature [1, 3, 4, 6, 8–10, 17, 20, 24, 29, 32–37, for instance]. Given the amount of available analogous algorithms applicable to the same tasks, it may be complex to find the most suitable. Furthermore, some procedures need the fine tuning of a set of hyper-parameters, usually requiring time and skills to be performed by users.

Comparisons between different instances of the same extractor, or different extractors, are usually carried out by observing *(i)* the predictive performance of the extractor, w.r.t. the underlying BB predictions and/or the actual data set output variables (i.e., the ground truth); *(ii)* the readability of the output human-intelligible knowledge; and *(iii)* the completeness of the knowledge in providing predictions [14, 25, 38]. The former can be easily assessed via the same metrics adopted to measure the predictive performance of the underlying BB (e.g., F_1 and accuracy scores for classification tasks and mean absolute/squared error and R^2 score for regression tasks). Readability may be measured through different indicators, however, to the best of our knowledge, a well-founded and sound definition has not yet been formulated. Finally, knowledge completeness may be calculated as the fraction of queries that the knowledge is able to predict or the percentage of covered input feature space.

Human comparisons and automated algorithmic comparisons can surely benefit from a unified scoring function encompassing the concepts of predictive performance, readability and completeness associated with the knowledge provided by SKE techniques. The definition of this kind of scoring function is the fundamental brick for moving towards automated ML (AutoML) [18]. Accordingly, in this paper we propose the FiRe score as a compact and expressive metric to evaluate and compare different knowledge extractors, also in association with automated parameter tuning procedures.

2 Motivations & State-of-the-Art

SKE techniques have been applied in a wide variety of areas [2, 5, 13, 16, 31, to cite some examples]. Comparisons between different instances of the same extractor, or different extractors, in order to select the "best extracted knowledge" are usually carried out by observing *i)* the extractor's predictive performance, its *ii)* completeness and *iii)* the corresponding readability in terms of number of rules, by considering high values of these indices as more desirable. Guidance on quantifying the three indices is available in Subsection 2.1.

Such a comparison in the literature is always done manually and

* Corresponding Author. Email: f.sabbatini1@campus.uniurb.it.

just by looking at the three indices separately. The comparison is therefore subjective: if knowledge K_a is compared with another knowledge K_b having the same readability and completeness but a smaller degree of predictive accuracy, then it is trivial to consider K_a as the best one. Conversely, if K_a has the same completeness and predictive accuracy as K_b, but a smaller readability extent, then the best knowledge is K_b. It is also trivial to select the best knowledge if K_a has at the same time all three indices with smaller values than knowledge K_b, which in this case would result as the best. The main challenge is to decide which is the best output amongst all those provided by SKE techniques when there is no knowledge maximising the three indices contemporaneously. Multiple optimum outputs can be highlighted, each corresponding to a specific evaluating criterion. However, when comparing slight improvements in predictive performance against minor losses in human readability, the process becomes subjective and dependent on human judgment. Consequently, the identification of a single, ultimate knowledge may be influenced by subjective biases in the selection process.

We believe that the formulation of a standardised, compact scoring function encompassing multiple indices may lead to a more impartial knowledge quality assessment than the human comparison performed separately for each available evaluating criterion. Furthermore, from an AutoML perspective, incorporating multiple indices into a unified metric enables the design and implementation of hyperparameter tuning algorithms for SKE techniques while considering all relevant evaluating criteria. For instance, such a knowledge quality evaluation metric would be invaluable in enhancing the robustness of the PEDRO procedure [24], an automated tool for identifying optimal parameter values in the GridEx and GridREx knowledge-extraction algorithms [24, 29].

To the best of our knowledge, the existing literature lacks both an automated evaluation approach and a formal definition of metrics that synergistically combine predictive performance and readability to provide a comprehensive and unified indicator for measuring the quality of extracted knowledge. A notable exception is presented in [25], where the authors propose a scoring function that incorporates predictive performance, readability, and completeness. In their approach, the authors introduce a multiplicative metric with three terms representing specific indices, each expressed as *loss* (i.e., predictive loss, readability loss, and coverage loss). For high-quality knowledge, these losses should be minimised. Consequently, the overall scoring function, obtained by multiplying the three losses, becomes smaller, signifying better knowledge quality. This scoring function enables automatic assignment of scores to extractors' outputs and facilitates comparisons within algorithmic parameter tuning routines. Besides this proposal, the literature lacks a more comprehensive automated evaluation approach that incorporates multiple criteria and enables the assignment of weights to each criterion.

FiRe differs from the existing metric in two significant ways. Firstly, FiRe does not incorporate the coverage index in its calculation of the overall score, relying solely on the concepts of predictive and readability losses. The motivation for this choice stems from the fact that, in certain applications, the primary focus is on selecting knowledge extracted by an extractor with superior predictive performance and readability. Completeness may be less important, especially for prediction and decision-making tasks where missing less relevant information has minimal impact. Additionally, prioritising accuracy and readability allows for the removal of less valuable knowledge and ensures retention of the most crucial and reliable information, especially in knowledge bases where noise or irrelevant data may be present. Secondly, FiRe introduces the option for users to input a parameter that determines the weight given to predictive loss compared to readability loss. This flexibility is crucial as it allows the scoring function to adapt to users' preferences. For instance, users may prioritise knowledge with minimal predictive error over highly readable knowledge that might have larger errors. By incorporating this parameter, FiRe accommodates a broader range of user needs and preferences, making it a versatile tool for automated knowledge evaluation.

2.1 Three Evaluation Indices: How to Quantify

In general, the predictive performance of the extracted knowledge may be assessed w.r.t. two different dimensions by using the same scoring function adopted for the underlying BB. These dimensions are: *(i)* the mimicking capabilities w.r.t. the underlying model predictions, usually called *fidelity*; and *(ii)* the *predictive performance* w.r.t. the data set output features.

Straightforward measurements of coverage and coverage loss are the rates of provided and missed predictions, respectively, w.r.t. the total amount of instances for which the knowledge has been queried. Alternatively, it is possible to calculate the percentage of covered and uncovered regions of the input feature space, respectively. Coverage is trivially equal to 1.0 for exhaustive knowledge-extraction algorithms, for instance, those based on decision trees.

The literature commonly assesses readability by comparing the number of extracted items, where algorithms with fewer extracted rules or shallower decision trees are considered more readable than those with larger numbers [11]. However, such an evaluation, solely based on this criterion, may be considered superficial as a comprehensive assessment demands a broader range of indicators [23]. To achieve a more thorough evaluation, a readability metric should encompass additional factors, including: *(i)* the shape of the extracted knowledge, e.g., list or trees of rules, decision tables, etc.; and *(ii)* the readability of individual atoms within the knowledge, such as how individual rules, tree nodes, and leaves are constructed. While these aspects are crucial for a comprehensive evaluation, their formalisation and numerical assessment require further investigation, which is beyond the scope of this current work. Akin to comparisons made by humans in the literature, FiRe considers knowledge size as the readability measure, similarly to the approach in [25].

3 The FiRe score

The FiRe (FIdelity vs. REadability) scoring metric formulated for measuring the quality of the knowledge extracted via SKE considers both the knowledge's predictive performance and its readability, intended as human interpretability w.r.t. the amount of extracted rules. The FiRe score is thus a multivariate function defined as follows:

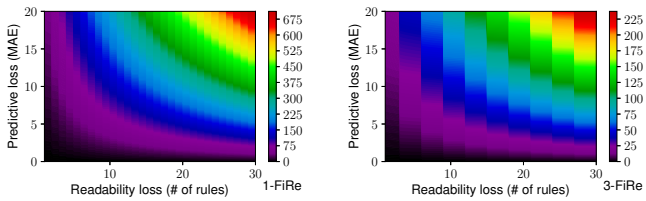
$$FiRe : (\mathbb{R}_{>0} \times \mathbb{R}_{\geq 0} \times \mathbb{R}_{\geq 1}) \mapsto \mathbb{R}_{\geq 0}, \tag{1}$$

$$FiRe(\psi, p, r) = p \left\lceil \frac{r}{\psi} \right\rceil r^{0.05}, \tag{2}$$

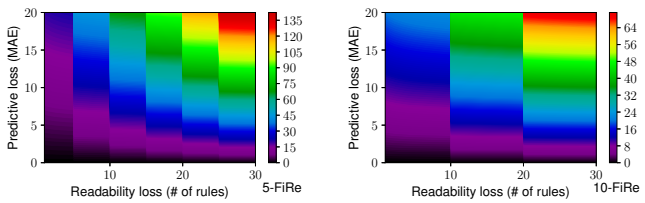
where $\lceil \cdot \rceil$ is the ceiling function, ψ is the fidelity/readability trade-off extent, p is a measure of the predictive loss of the extractor and r is a measure of its readability loss.

3.1 Variables and parameters

As for the predictive loss p, a good measure in regression tasks is the mean absolute error (MAE) of the extractor's predictions w.r.t.

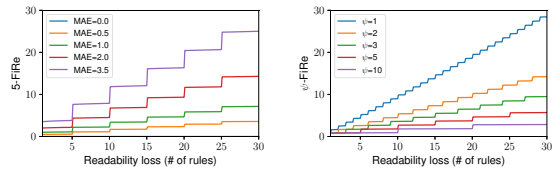


(a) 1-FiRe. (b) 3-FiRe.

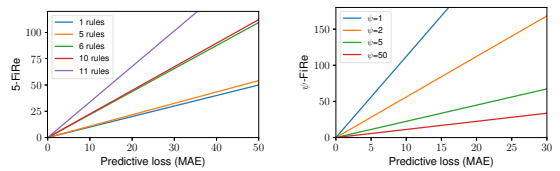


(c) 5-FiRe. (d) 10-FiRe.

Figure 1: Graphs of different ψ-FiRe scoring functions.



(a) 5-FiRe w.r.t. readability loss. (b) ψ-FiRe functions having equal predictive loss.



(c) 5-FiRe w.r.t. predictive loss. (d) ψ-FiRe functions having equal readability loss.

Figure 2: Projections of several ψ-FiRe functions.

the underlying BB predictions or the data set outputs, depending on the need. For classification tasks, it is possible to use metrics anti-correlated with the accuracy score, e.g., $1-accuracy$. The presented examples adopt the mean absolute error metric for regression, but it may be substituted with the mean squared error without substantial differences since both of them are generally correlated. Analogously, metrics inversely proportional to the R^2 index may be also exploited.

As for the readability loss r, the total amount of output rules is a suitable metric and thus it is the one adopted to calculate the FiRe score. More complex options will be evaluated in the future, e.g., by taking into account the complexity of individual rules.

Finally, the ψ parameter – the only user-defined parameter – describes how much the predictive loss can be penalised w.r.t. the readability loss. It is important because, depending on the task at hand, the two losses may have different weights. In particular, $\psi = 1$ assigns the same importance to both losses. Growing ψ values tend to neglect the readability loss impact. In other words, given the aforementioned FiRe score formulation and by assuming to have extracted m rules, if users set $\psi = n$ the FiRe score will consider only $\frac{m}{n}$ rules, rounded up to the nearest integer.

3.2 *Function domain*

The function domain is explained by the following observations.

- The ψ parameter is a positive real value by design. Limiting the admissible ψ values to $\mathbb{N}$ may be also reasonable, but an extension to $\mathbb{R}_{>0}$ makes the FiRe score more flexible.
- On the other hand, the p parameter is a measurement of a predictive error, so it may be equal to 0 in the best case, or arbitrarily larger otherwise since there is no upper bound to the predictive error of a model.
- Finally, r is an integer number greater or equal to 1 since it represents the number of extracted rules, that is a discrete quantity equal to 1 (in the best case) or larger (otherwise). However, the admissible values for this parameter have been extended to $\mathbb{R}_{\geq 1}$ for the sake of flexibility, analogously to the range for ψ. This choice enables, for instance, the FiRe score calculation for averaged sets of extractors trained with the same hyper-parameters, resulting in a more robust score.

As a result of the observations above, the FiRe score is defined as a continuous (yet non-differentiable) function in the aforementioned domain and it may assume any non-negative value. Therefore, the score is a function bounded from below by 0.

3.3 *Score meaning*

In the following, we use the notation ψ-FiRe(p, r) as alias of FiRe(ψ, p, r) and we consider w.l.o.g. the ψ-FiRe($\cdot$) function as a bivariate function, by assuming the ψ parameter fixed *a priori*.

The FiRe score has been formulated to assign low scores to desirable extractors. It assumes that a good extractor should exhibit a low predictive loss and a low readability loss. For this reason, it is a multiplicative score between the two parameters and, therefore, it is sufficient that only one amongst the predictive or the readability loss is high to bring the FiRe score towards high values corresponding to low-quality knowledge.

The ceiling function appearing as the second factor of the score has the purpose to give a step-function shape to the FiRe score. The exact shape of the steps is regulated through the ψ parameter, acting as a user-defined sensitivity threshold and thus appearing as the denominator in the second term of the score formulation. The core idea behind ψ is to minimise distinctions between extractors or knowledge with identical predictive loss and similar readability loss by flattening the FiRe isolines. This flattening is accomplished by adopting a step-like shape for the isolines, ensuring that all extractors sharing the same step also possess the same FiRe score. As ψ increases, the step width widens, leading to a higher tolerance for the presence of readability loss. By setting $\psi = n$, users impose these steps to have a length equal to n. Since a flat step would assign the same ψ-FiRe score to extractors having the same predictive loss and a different but similar readability loss (e.g., r = 1 and 2, respectively, and $\psi = 10$), a third factor is queued to the score definition to establish rankings within individual steps and thus discern amongst the extractors lying on the same step which one has to be considered the best. We found that an exponent equal to 0.05 allows the scoring function to preserve all its peculiarities, at the same time adding the desired increasing trend to the step function (monotonicity). In this way, the FiRe score keeps the step-function shape but becomes an increasing monotonic function (for any $p > 0$, since $\psi\text{-}FiRe(0, r) = 0, \forall r, \forall \psi$). We stress

here that $p = 0$ is a theoretically possible scenario, but it is practically impossible to obtain a model having no predictive error. Examples of ψ-FiRe graphs are reported in Figure 1, for different values of ψ, p and r.

3.4 Properties

The monotonicity of the ψ-FiRe score is ensured by:

monotonicity w.r.t. the projection of p (cf. Figure 2a)

$$r_1 < r_2 \iff \psi\text{-}FiRe(p, r_1) < \psi\text{-}FiRe(p, r_2), \quad \forall p \in \mathbb{R}_{>0}, \quad \forall r_1, r_2 \in \mathbb{R}_{\geq 1} \tag{3}$$

monotonicity w.r.t. the projection of r (cf. Figure 2c)

$$p_1 < p_2 \iff \psi\text{-}FiRe(p_1, r) < \psi\text{-}FiRe(p_2, r), \quad \forall r \in \mathbb{R}_{\geq 1}, \quad \forall p_1, p_2 \in \mathbb{R}_{>0} \tag{4}$$

Equations 3 and 4 can be substituted by the following condition:

monotonicity w.r.t. a partial order on the domain

$$\begin{aligned} &(p_1 < p_2) \wedge (r_1 < r_2) \iff \\ &\iff \psi\text{-}FiRe(p_1, r_1) < \psi\text{-}FiRe(p_2, r_2), \\ &\forall p_1, p_2 \in \mathbb{R}_{>0}, \quad \forall r_1, r_2 \in \mathbb{R}_{\geq 1} \end{aligned} \tag{5}$$

Equations 3, 4 and 5 also hold by substituting $<$ with $>$.

The increasing trend of the score may be observed in Figure 2 and it is demonstrated through its partial derivatives. Equations 3, 4 and 5 hold for any possible $\psi > 0$ that may be assigned to the ψ-FiRe score, as it is possible to notice from Figures 2b and 2d.

The partial derivative w.r.t. p is the following:

$$\frac{\partial \psi\text{-}FiRe}{\partial p} = \left\lceil \frac{r}{\psi} \right\rceil r^{0.05} \tag{6}$$

always positive and defined in the whole domain. The partial derivative w.r.t. r is the following:

$$\frac{\partial \psi\text{-}FiRe}{\partial r} = \frac{0.05p \left\lceil \frac{r}{\psi} \right\rceil}{r^{0.95}} \tag{7}$$

always positive for $p > 0$ and defined in the whole domain except for $\frac{r}{\psi} \in \mathbb{Z}$. The derivative is 0 for $p = 0$, indeed in this case the ψ-FiRe score is always 0 regardless of the values of ψ and r.

4 On the practical use of FiRe

This section discusses practical applications of the FiRe score. Subsection 4.1 provides examples demonstrating the FiRe score's application, along with relevant observations. Subsection 4.2 presents an analytical study of the balance between parameters representing predictive and readability losses. Lastly, Subsection 4.3 offers insights into tuning the ψ parameter from a practical perspective.

4.1 Comparing algorithms with FiRe

Given all the aforementioned properties, we exemplify here some applicative scenarios from a theoretical point of view. Let us assume to have an extraction procedure providing as output knowledge a single human-interpretable rule. The mean absolute error associated with this rule is equal to 40.0 and we chose to adopt $\psi = 1$. As a consequence, $1\text{-}FiRe(40.0, 1) = 40.0$.

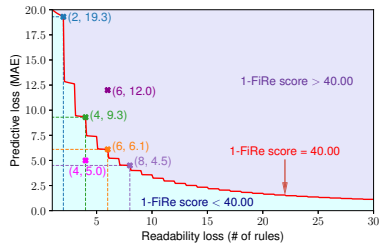


Figure 3: Graphical representation of the boundaries identified by the 1-FiRe score (isoline for $1\text{-}FiRe = 40.0$).

In Figure 3 the isoline corresponding to an 1-FiRe score equal to 40.0 is shown in red. Readability loss and predictive loss are reported on the x-axis and y-axis, respectively, as the number of extracted rules and mean absolute error. The described extractor, with 1 rule and a predictive error equal to 40.0, lies on the red isoline. The same condition holds for all extractors having the same 1-FiRe score value. This is the case, for example, of extractors providing 2, 4, 6 and 8 rules associated with MAE of 19.3, 9.3, 6.1 and 4.5, respectively. All these models are *equivalent* on the basis of the 1-FiRe score.

Conversely, a model able to extract 4 rules with a predictive error of 5.0 is considered *better*, since it has a smaller 1-FiRe score so it lies in the graph under the red isoline. More precisely, $1\text{-}FiRe(5.0, 4) = 21.4$. An extractor providing 6 rules with MAE = 12.0 is *worse* since its 1-FiRe score is greater than 40.0 and thus it graphically lies above the red isoline. Indeed, $1\text{-}FiRe(12.0, 6) = 78.7$. These two latter examples have the purpose of demonstrating that the FiRe score is still effective even if only one amongst the predictive or readability losses is subject to changes.

Figure 3 clearly highlights how the FiRe score identifies an exact boundary separating, w.r.t. a given extractor, the sets of equivalent, worse and better extractors. Furthermore, the isoline depicts the fidelity/readability trade-off correlated to $\psi = 1$. By observing the red isoline it is noticeable how a doubling of the readability loss (e.g., from 2 to 4) is accepted only if it is approximately balanced with a halving of the predictive loss. The curve can be also read in the opposite sense, e.g., a doubling of the predictive loss is accepted only when (approximately) compensated by a readability loss halving. With this reading key, it is possible to exploit the FiRe isolines to analytically study better and worse extraction procedures w.r.t. a fixed one, by taking into consideration both increases and decreases of the predictive loss and/or of the readability loss.

Finally, we exploit the same figure to stress the fact that the isoline presents an asymptotic trend when the p and r parameters tend to infinity. This behaviour reflects the actual quality of the knowledge provided by SKE techniques. Indeed, when the number of rules and/or the predictive error are very high, the evaluated knowledge has low quality and it is no more a relevant task to have a fine-grained measure of *how* a loss should be compensated by the other.

4.2 Predictive and readability loss equilibrium

Given the ability of the FiRe score in providing the notion of *knowledge equivalence* according to more than one index, SKE techniques' users have the possibility to perform analytical investigations on the knowledge quality to understand *(i)* the predictive loss decrease (resp. increase) exactly annihilating a readability loss increase (resp.

decrease); and *(ii)* the predictive loss decrease (resp. increase) having the same effect as a readability loss decrease (resp. increase). This enables not only knowledge evaluation and best knowledge selection but also a pairwise comparison based on predictive and readability loss alteration.

In the first case, the aim is to balance predictive loss and readability loss, despite their opposite variations. This allows users to spot a set of equivalent knowledge and to analytically observe the corresponding FiRe isolines, understanding how to set the ψ parameter.

In the second case, the focus is on studying the effects of individual decreases or increases in the losses to find equivalent pairs. Thanks to this, users highlight the effect of the fidelity/readability ψ trade-off and thus they may discover that it may be sufficient a small enhancement in one loss rather than big efforts on the other to achieve better knowledge quality.

In order to find the equilibrium between predictive and readability losses, the relationship to be satisfied can be expressed by the following equivalence, where α and β are two unknown constants:

$$\psi\text{-}FiRe(p, r) = \psi\text{-}FiRe(\alpha p, \beta r), \tag{8}$$

that is equivalent to:

$$p \left\lceil \frac{r}{\psi} \right\rceil r^{0.05} = \alpha p \left\lceil \frac{\beta r}{\psi} \right\rceil (\beta r)^{0.05}. \tag{9}$$

By resolving w.r.t. α we obtain:

$$\alpha = \frac{\left\lceil \frac{r}{\psi} \right\rceil}{\left\lceil \frac{\beta r}{\psi} \right\rceil \beta^{0.05}}. \tag{10}$$

These equations demonstrate that an extracted knowledge with a predictive loss p and readability loss r has the same quality as another knowledge having predictive loss αp and readability loss βr if and only if Equation 10 is satisfied.

Furthermore, Equation 10 highlights that the p parameter is not relevant in finding a pair of compensating α and β multiplicative constants for the losses involved in the FiRe score calculation, differently to r and ψ that play a role in the pair determination. This is not surprising, given that the FiRe score is directly proportional to the predictive loss and therefore the increasing score (denoting a quality worsening) due to a predictive loss increase is not dependent on the actual value of the loss before or after the increase. Conversely, how a readability loss increase affects the FiRe score is strictly dependent on the ψ parameter and the loss value itself.

When altering the predictive and readability losses in the FiRe score, given the monotonicity properties of the scoring function, intuitively we may have predictive loss increases compensated by readability loss decreases or, vice versa, readability loss increases balanced by predictive loss decreases. Equation 10 confirms this intuition. Indeed, readability loss increases are represented with $\beta > 1$ values. This implies that $\beta^{0.05} > 1$ and that $\left\lceil \frac{r}{\psi} \right\rceil \leq \left\lceil \frac{\beta r}{\psi} \right\rceil$. Thus, if $\beta > 1$ then $\alpha < 1$. Conversely, if $\beta < 1$ then $\beta^{0.05} < 1$ and $\left\lceil \frac{r}{\psi} \right\rceil \geq \left\lceil \frac{\beta r}{\psi} \right\rceil$. Therefore, if $\beta < 1$ then $\alpha > 1$. Obviously, if $\beta = 1$ then $\alpha = 1$.

All these relationships may be traced in Figure 4. In the figure, the relationship between the α and β values are shown for different values of the threshold parameter ψ and the readability loss r, given that α depends also on them other than β. It is possible to notice that with a fixed β and for growing values of ψ the effects of β are more and more neglected, whereas for growing values of r they are more pronounced. As a consequence, when $\beta > 1$ increasing ψ or decreasing r implies increasing α. Conversely, when $\beta < 1$ increasing ψ as well as decreasing r induces to decrease α values.

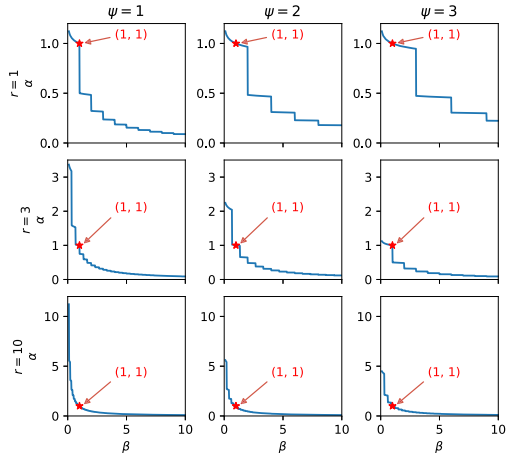


Figure 4: Equilibrium between α and β for different values of the readability loss r and the ψ parameter.

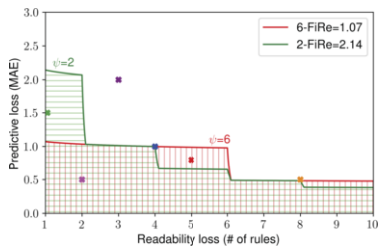


Figure 5: Boundaries associated to the 2-FiRe and 6-FiRe scores.

To conclude the analysis of the entanglement between the predictive and readability losses, we point out that in order to find equivalent pairs of increases/decreases in the predictive and readability losses it is sufficient to modify Equation 8 as follows:

$$\psi\text{-}FiRe(\alpha p, r) = \psi\text{-}FiRe(p, \beta r). \tag{11}$$

Developing the equation similarly to the equilibrium case we find:

$$\alpha = \frac{\left\lceil \frac{\beta r}{\psi} \right\rceil \beta^{0.05}}{\left\lceil \frac{r}{\psi} \right\rceil}. \tag{12}$$

By applying the same considerations as before, we find that in this case α and β are both $= 1$, < 1, or > 1.

4.3 *Tuning the ψ parameter*

It is of fundamental importance to carefully choose the ψ parameter of FiRe. Figure 2b already showed that larger values of ψ reduce the impact of the readability loss. However, it is important to know that different ψ values may lead to *opposite* results when applied to compare the same extractors. This peculiarity is depicted in Figure 5, representing the separating boundaries identified by the isolines obtained via the 2-FiRe and 6-FiRe scores w.r.t. a given extractor described in the following. The boundaries associated with the two

Algorithm	Accuracy	Coverage	Extracted rules (r)	Predictive loss (p)	Coverage loss (c)	Q_s	ψ-FiRe $\psi=1$	$\psi=2$	$\psi=3$
9-NN	0.97	-	-	-	-	-	-	-	-
CART	0.95	1.00	3	0.05	1.00	0.15	0.16	0.11	0.05
ITER	0.94	1.00	3	0.06	1.00	0.18	0.19	0.13	0.06
CREEPY (2 features)	0.97	1.00	3	0.03	1.00	**0.09**	**0.10**	**0.06**	**0.03**
CREEPY (1 feature)	0.93	1.00	3	0.07	1.00	0.21	0.22	0.15	0.07
GridEx (3 rules)	0.96	0.95	3	0.04	1.05	0.13	0.13	0.08	0.04
GridEx (4 rules)	0.87	1.00	4	0.13	1.00	0.52	0.56	0.28	0.28
GridEx (6 rules)	0.98	0.43	6	0.02	1.57	0.19	0.13	0.07	0.04

Table 1: Quality assessments for the knowledge extracted by different SKE algorithms from a 9-NN for the Iris data set.

scoring functions are represented as green and red isolines, respectively. The hatched area below each isoline highlights the parameter space region denoting "more desirable" extractors, providing knowledge with better quality w.r.t. extractors lying on the isoline.

Let us assume to have an extractor able to obtain 4 rules from a BB model with a mean absolute error equal to 1.0 (blue cross in Figure 5). The ψ-FiRe scores associated with this model are:

$$2\text{-}FiRe(1.0, 4) = 2.14, \qquad 6\text{-}FiRe(1.0, 4) = 1.07.$$

An SKE algorithm extracting 8 rules with MAE = 0.5 (orange cross) has the same FiRe scores for both values of ψ. The models are thus equivalent according to both of the considered scoring functions. Analogously, by assuming two extractors providing 2 and 3 output rules with predictive errors equal to 0.5 and 2.0, respectively (fuchsia and purple cross in the figure), both scores are unanimous in evaluating the former as a better extraction procedure and in considering worse the latter.

Different behaviours can be observed, for instance, by selecting an extracted knowledge composed of a single rule with MAE = 1.5 (green cross). In this case, the scores are evaluated as follows:

$$2\text{-}FiRe(1.5, 1) = 1.5 < 2.14 = 2\text{-}FiRe(1.0, 4),$$

$$6\text{-}FiRe(1.5, 1) = 1.5 > 1.07 = 6\text{-}FiRe(1.0, 4),$$

and their interpretation leads to opposite conclusions. In particular, the single-ruled knowledge is considered better than the others lying on the isoline if considering $\psi = 2$. It is worse if considering $\psi = 6$. The dual situation can be encountered with knowledge having 5 rules and MAE = 0.8 (red cross). In this case, the knowledge quality is considered better when evaluated through the 6-FiRe score and worse with the 2-FiRe score.

Considering these remarks, we recommend choosing the most suitable value for ψ by observing the corresponding isolines.

5 Experiments and discussion

The effectiveness of the FiRe score in evaluating and comparing the quality of SKE techniques' extracted knowledge has been assessed by running several experiments. The PSYKE framework[1] [7, 27, 28, 30] has been used to train a BB predictor and a set of extractors on the well-known Iris data set[2] [12]. Using a simple data set allows easy depiction of decision boundaries and facilitates visual comparisons of resulting knowledge. The adopted extractors are the following: CART [6], ITER [17], CREEPY [26] and GridEx [29]. All these techniques have been applied to a k-nearest neighbour (k-NN) classifier, having $k = 9$. Since all the extractors are model-agnostic algorithms, the provided output knowledge has been extracted only by observing the input/output response of the 9-NN.

The FiRe score relies on predictive and readability loss concepts, making it applicable to any model with expressible predictive and readability losses. It remains effective across various model and data set complexities without sacrificing generalisation.

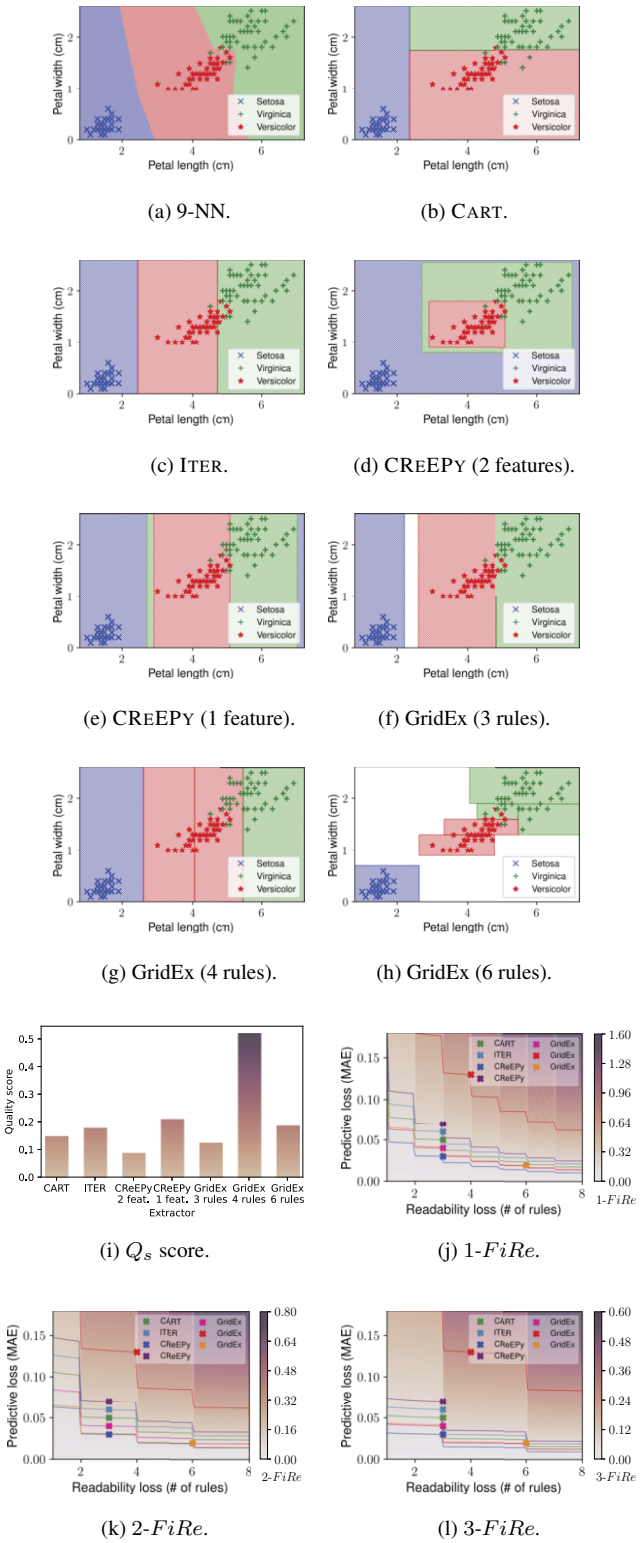
To better understand the example, we recall that CART induces a decision tree classifier on the 9-NN predictions and it has been executed with the default parameters of the PSYKE implementation (maximum 3 leaves). On the other hand, ITER, CREEPY and GridEx produce a hypercubic partitioning of the input feature space according to different strategies. ITER creates and expands cubes in a bottom-up iterative fashion. It relies on 4 hyper-parameters: *(i)* the number of starting cubes, set to 1; *(ii)* the minimum amount of instances to consider inside each cube, set to 75; *(iii)* the size of cube updates, set to 7% of each input feature range interval; *(iv)* the maximum number of iterations to be performed, set to 600.

GridEx partitions the input feature space in a top-down recursive and symmetric manner, starting from the whole space. It relies on 4 hyper-parameters: *(i)* the maximum depth of the recursive splitting; *(ii)* the minimum amount of instances to consider inside each cube, set to 1; *(iii)* the number of slices to perform at each iteration; *(iv)* the error threshold to decide if a hypercubic region should be further partitioned, set to 0.1. An error threshold equal to 0.1 means that all cubes having an accuracy smaller than 0.9 are further split. The number of slices to perform has been adaptively chosen. In particular, our experiments, resumed in Table 1, consider 3 GridEx instances. The first and the second perform 8 and 2 slices, respectively, only along the most relevant input feature. The third performs 4 slices on the 2 most relevant input dimensions. As for the maximum depth, the first instance has a value equal to 1, and the others equal to 2.

CREEPY adopts an underlying clustering technique to divide the input space into hypercubic hierarchical regions. We set equal to 2 the maximum depth parameter and equal to 0.1 the error threshold, which has the same semantics as that of GridEx. For our experiments, we trained 2 CREEPY instances, one considering the most relevant input feature and the other considering also the second one.

The classification accuracy of each extractor, as well as that of the 9-NN, has been reported in Table 1. The table also shows the number of extracted rules, representing the readability loss r of the extractors. Analogously, the predictive loss p is reported as $1 - accuracy$. The coverage achieved by the extraction techniques is reported as the portion of input space covered by the corresponding knowledge, in percentage. The coverage loss has been calculated as $2 - coverage$, according to the definition in [25]. Finally, the last four columns report the quality scores associated with each extractor. Best scores are highlighted in bold font. We compared the ψ-FiRe scores calculated for different values of ψ with the quality score (Q_s) described

[1] https://github.com/psykei/psyke-python
[2] https://archive.ics.uci.edu/ml/datasets/iris



(a) 9-NN. (b) CART.

(c) ITER. (d) CREEPY (2 features).

(e) CREEPY (1 feature). (f) GridEx (3 rules).

(g) GridEx (4 rules). (h) GridEx (6 rules).

(i) Q_s score. (j) 1-$FiRe$.

(k) 2-$FiRe$. (l) 3-$FiRe$.

Figure 6: Decision boundaries for the Iris data set obtained with different extractors applied to a 9-NN, corresponding quality score Q_s and 1-FiRe, 2-FiRe and 3-FiRe score isolines.

in [25][3]. Data has been averaged upon 5 executions for each extractor. The results' standard deviation has been omitted since each algorithm provided very similar outputs at every execution.

Figure 6 shows graphical representations of decision boundaries for the 9-NN and extractors. The bottom row displays the quality assessments. The Q_s score is reported in Figure 6i, whereas Figures 6j, 6k and 6l show the isolines for the ψ-FiRe scores adopted in the experiments. Notably, obtaining fewer than 3 rules is not possible since the Iris data set has 3 output classes, and the best-case knowledge includes exactly one rule per distinct class.

Table 1 and Figure 6 show that the CREEPY algorithm with 2 input features excels in terms of coverage, readability, and predictive performance, making it the top-performing algorithm. Indeed, it has the lowest Q_s and ψ-FiRe scores regardless of the adopted ψ. This result is true and acceptable since CREEPY is the algorithm providing the smallest amount of rules with the smallest predictive error.

Different conclusions may be drawn by comparing according to different quality scores CART, the GridEx instance providing knowledge with 3 rules and the one providing 6 rules. Indeed, according to the Q_s score, the best knowledge amongst these is the one having 3 rules and obtained via GridEx ($Q_s = 0.13$), whereas the GridEx instance providing 6 rules is the worst choice ($Q_s = 0.19$). CART's quality is assessed with $Q_s = 0.15$, representing a middle evaluation between the two GridEx instances. GridEx providing 6 rules is considered the worst algorithm despite its minimum predictive loss since it has very high readability and coverage losses w.r.t. CART and the other GridEx instance. CART is considered slightly worse than GridEx providing 3 rules despite having the same readability loss because GridEx has a higher coverage loss that is more than compensated by a better predictive loss.

The FiRe scores yield diverse conclusions when comparing the same three algorithms. The FiRe score is based on predictive and readability losses while neglecting coverage loss. As a result, Both GridEx instances outperform CART due to CART's higher predictive loss without adequate compensation from its lower readability loss, regardless of the chosen ψ value. The comparison between the two GridEx instances is influenced by the ψ parameter, which assigns the same quality score to both instances when $\psi = 1$ and $\psi = 3$. Only in these cases, the fluctuations in predictive and readability losses are balanced by the user-defined fidelity/readability trade-off value.

6 Conclusions

The paper presents FiRe, a scoring function to evaluate and compare SKE algorithms. It is a compact score encompassing both a readability assessment and a predictive performance evaluation and it may be exploited to help users choose the best extraction procedure w.r.t. a specific fidelity/readability trade-off, expressed as a parameter. The FiRe score may also be applied together with automatic parameter-tuning procedures. We show here the properties of the scoring function and a rigorous mathematical formulation is also provided.

Future works will focus on enhancing the FiRe score readability parameter, with a more expressive formulation than the mere amount of rules provided as output, e.g., by taking into account the complexity of rules' atoms in terms of number of constraints, variables, etc.

Acknowledgements

This work has been supported by the EU ICT-48 2020 project TAILOR (No. 952215).

[3] We recall that Q_s is calculated by multiplying the predictive loss, the coverage loss and the number of extracted rules.

References

[1] Robert Andrews and Shlomo Geva, 'RULEX & CEBP networks as the basis for a rule refinement system', in *Hybrid Problems, Hybrid Solutions*, ed., J. Hallam, pp. 1–12. IOS Press, (1995).

[2] Bart Baesens, Rudy Setiono, Christophe Mues, and Jan Vanthienen, 'Using neural network rule extraction and decision tables for credit-risk evaluation', *Management Science*, **49**(3), 312–329, (2003).

[3] Nahla Barakat and Joachim Diederich, 'Eclectic rule-extraction from support vector machines', *International Journal of Computer and Information Engineering*, **2**(5), 1672–1675, (2008).

[4] José Manuel Benítez, Juan Luis Castro, and Ignacio Requena, 'Are artificial neural networks black boxes?', *IEEE Trans. Neural Networks*, **8**(5), 1156–1164, (1997).

[5] Guido Bologna and Christian Pellegrini, 'Three medical examples in neural network rule extraction', *Physica Medica*, **13**, 183–187, (1997).

[6] Leo Breiman, Jerome Friedman, Charles J. Stone, and Richard A. Olshen, *Classification and Regression Trees*, CRC Press, 1984.

[7] Roberta Calegari and Federico Sabbatini, 'The PSyKE technology for trustworthy artificial intelligence', **13796**, 3–16, (March 2023). XXI International Conference of the Italian Association for Artificial Intelligence, AIxIA 2022, Udine, Italy, November 28 – December 2, 2022, Proceedings.

[8] Luis A. Castillo, Antonio González Muñoz, and Raúl Pérez, 'Including a simplicity criterion in the selection of the best rule in a genetic fuzzy learning algorithm', *Fuzzy Sets Syst.*, **120**(2), 309–321, (2001).

[9] Mark W. Craven and Jude W. Shavlik, 'Using sampling and queries to extract rules from trained neural networks', in *Machine Learning Proceedings 1994*, 37–45, Elsevier, (1994).

[10] Mark W. Craven and Jude W. Shavlik, 'Extracting tree-structured representations of trained networks', in *Advances in Neural Information Processing Systems 8. Proceedings of the 1995 Conference*, eds., David S. Touretzky, Michael C. Mozer, and Michael E. Hasselmo, 24–30, The MIT Press, (June 1996).

[11] Ireneusz Czarnowski, Alfonso Mateos Caballero, Robert J Howlett, and Lakhmi C Jain, *Intelligent Decision Technologies 2016: Proceedings of the 8th KES International Conference on Intelligent Decision Technologies (KES-IDT 2016)–Part I*, volume 56, Springer, 2016.

[12] R. A. Fisher, 'The use of multiple measurements in taxonomic problems', *Annals of Eugenics*, **7**(2), 179–188, (1936).

[13] Leonardo Franco, José Luis Subirats, Ignacio Molina, Emilio Alba, and José M. Jerez, 'Early breast cancer prognosis prediction and rule extraction using a new constructive neural network algorithm', in *Computational and Ambient Intelligence (IWANN 2007)*, volume 4507 of *LNCS*, pp. 1004–1011. Springer, (2007).

[14] Arthur S. d'Avila Garcez, Krysia Broda, and Dov M. Gabbay, 'Symbolic knowledge extraction from trained neural networks: A sound approach', *Artificial Intelligence*, **125**(1-2), 155–207, (2001).

[15] Riccardo Guidotti, Anna Monreale, Salvatore Ruggieri, Franco Turini, Fosca Giannotti, and Dino Pedreschi, 'A survey of methods for explaining black box models', *ACM Computing Surveys*, **51**(5), 1–42, (2018).

[16] Yoichi Hayashi, Rudy Setiono, and Katsumi Yoshida, 'A comparison between two neural network rule extraction techniques for the diagnosis of hepatobiliary disorders', *Artificial intelligence in Medicine*, **20**(3), 205–216, (2000).

[17] Johan Huysmans, Bart Baesens, and Jan Vanthienen, 'ITER: An algorithm for predictive regression rule extraction', in *Data Warehousing and Knowledge Discovery (DaWaK 2006)*, pp. 270–279. Springer, (2006).

[18] Shubhra Kanti Karmaker, Md Mahadi Hassan, Micah J Smith, Lei Xu, Chengxiang Zhai, and Kalyan Veeramachaneni, 'Automl to date and beyond: Challenges and opportunities', *ACM Computing Surveys (CSUR)*, **54**(8), 1–36, (2021).

[19] Eoin M. Kenny, Courtney Ford, Molly Quinn, and Mark T. Keane, 'Explaining black-box classifiers using post-hoc explanations-by-example: The effect of explanations and error-rates in XAI user studies', *Artificial Intelligence*, **294**, 103459, (2021).

[20] Haydemar Núñez, Cecilio Angulo, and Andreu Català, 'Rule extraction based on support and prototype vectors', in *Rule Extraction from Support Vector Machines*, ed., Joachim Diederich, volume 80 of *Studies in Computational Intelligence*, 109–134, Springer, (2008).

[21] Anderson Rocha, Joao Paulo Papa, and Luis A. A. Meira, 'How far do we get using machine learning black-boxes?', *International Journal of Pattern Recognition and Artificial Intelligence*, **26**(02), 1261001–(1–23), (2012).

[22] Cynthia Rudin, 'Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead', *Nature Machine Intelligence*, **1**(5), 206–215, (2019).

[23] Federico Sabbatini and Roberta Calegari, 'Evaluation metrics for symbolic knowledge extracted from machine learning black boxes: A discussion paper', *arXiv preprint arXiv:2211.00238*, (2022).

[24] Federico Sabbatini and Roberta Calegari, 'Symbolic knowledge extraction from opaque machine learning predictors: GridREx & PEDRO', in *Proceedings of the 19th International Conference on Principles of Knowledge Representation and Reasoning, KR 2022, Haifa, Israel. July 31 - August 5, 2022*, eds., Gabriele Kern-Isberner, Gerhard Lakemeyer, and Thomas Meyer, (2022).

[25] Federico Sabbatini and Roberta Calegari, 'On the evaluation of the symbolic knowledge extracted from black boxes', in *AAAI 2023 Spring Symposium Series (to appear)*, San Francisco, California, (March 2023).

[26] Federico Sabbatini and Roberta Calegari, 'Unveiling opaque predictors via explainable clustering: The CReEPy algorithm', in *Proceedings of the XXII International Conference of the Italian Association for Artificial Intelligence, AIxIA 2023, Rome, Italy, November 6–9, 2023, (submitted to)*, (2023).

[27] Federico Sabbatini, Giovanni Ciatto, Roberta Calegari, and Andrea Omicini, 'On the design of PSyKE: A platform for symbolic knowledge extraction', in *WOA 2021 – 22nd Workshop "From Objects to Agents"*, eds., Roberta Calegari, Giovanni Ciatto, Enrico Denti, Andrea Omicini, and Giovanni Sartor, volume 2963 of *CEUR Workshop Proceedings*, pp. 29–48. Sun SITE Central Europe, RWTH Aachen University, (October 2021). 22nd Workshop "From Objects to Agents" (WOA 2021), Bologna, Italy, 1–3 September 2021. Proceedings.

[28] Federico Sabbatini, Giovanni Ciatto, Roberta Calegari, and Andrea Omicini, 'Symbolic knowledge extraction from opaque ML predictors in PSyKE: Platform design & experiments', *Intelligenza Artificiale*, **16**(1), 27–48, (2022).

[29] Federico Sabbatini, Giovanni Ciatto, and Andrea Omicini, 'GridEx: An algorithm for knowledge extraction from black-box regressors', in *Explainable and Transparent AI and Multi-Agent Systems. Third International Workshop, EXTRAAMAS 2021, Virtual Event, May 3–7, 2021, Revised Selected Papers*, eds., Davide Calvaresi, Amro Najjar, Michael Winikoff, and Kary Främling, volume 12688 of *LNCS*, 18–38, Springer Nature, Basel, Switzerland, (2021).

[30] Federico Sabbatini, Giovanni Ciatto, and Andrea Omicini, 'Semantic Web-based interoperability for intelligent agents with PSyKE', in *Explainable and Transparent AI and Multi-Agent Systems*, eds., Davide Calvaresi, Amro Najjar, Michael Winikoff, and Kary Främling, volume 13283 of *Lecture Notes in Computer Science*, chapter 8, 124–142, Springer, (2022).

[31] Federico Sabbatini and Catia Grimani, 'Symbolic knowledge extraction from opaque predictors applied to cosmic-ray data gathered with LISA Pathfinder', *Aeronautics and Aerospace Open Access Journal*, **6**(3), 90–95, (2022).

[32] Kazumi Saito and Ryohei Nakano, 'Extracting regression rules from neural networks', *Neural Networks*, **15**(10), 1279–1288, (2002).

[33] Gregor P. J. Schmitz, Chris Aldrich, and François S. Gouws, 'ANN-DT: an algorithm for extraction of decision trees from artificial neural networks', *IEEE Transactions on Neural Networks*, **10**(6), 1392–1401, (1999).

[34] Rudy Setiono and Wee Kheng Leow, 'FERNN: An algorithm for fast extraction of rules from neural networks', *Appl. Intell.*, **12**(1-2), 15–25, (2000).

[35] Rudy Setiono, Wee Kheng Leow, and Jacek M. Zurada, 'Extraction of rules from artificial neural networks for nonlinear regression', *IEEE Transactions on Neural Networks*, **13**(3), 564–577, (2002).

[36] Rudy Setiono and Huan Liu, 'NeuroLinear: From neural networks to oblique decision rules', *Neurocomputing*, **17**(1), 1–24, (1997).

[37] Rudy Setiono and James Y. L. Thong, 'An approach to generate rules from neural networks for regression problems', *Eur. J. Oper. Res.*, **155**(1), 239–250, (2004).

[38] Son N. Tran and Artur S. d'Avila Garcez, 'Knowledge extraction from deep belief networks for images', in *IJCAI-2013 workshop on neural-symbolic learning and reasoning*, (2013).