A Model-Agnostic SAT-based Approach for Symbolic Explanation Enumeration

Ryma Boumazouza, Fahima Cheikh-Alili, Bertrand Mazure, Karim Tabia

A Model-Agnostic SAT-based approach for Symbolic Explanation Enumeration (Preprint version)

Ryma Boumazouza[0000−0002−3940−8578], Fahima Cheikh-Alili[0000−0002−4543−625X], Bertrand Mazure[0000−0002−3508−123X], and Karim Tabia[0000−0002−8632−3980]

CRIL, Univ. Artois and CNRS, F62307 Lens, France
{ryma.boumazouza,cheikh,bertrand.mazure,karim.tabia}@univ-artois.fr

Abstract. We propose a generic agnostic approach allowing to generate different and complementary types of symbolic explanations. More precisely, we generate explanations to locally explain a single prediction by analyzing the relationship between the features and the output. Our approach uses a propositional encoding of the predictive model and a SAT-based setting to generate two types of symbolic explanations which are *Sufficient Reasons* and *Counterfactuals*. The experimental results on imagee classification task show the feasibility of the proposed approach and its effectiveness in providing *Sufficient Reasons* and *Counterfactual* explanations.

Keywords: Explainable Artificial Intelligence (XAI) · Symbolic explanations · Model-Agnostic · Satisfiability testing.

1 Introduction

Modern machine learning (ML) and deep learning methods are nowadays widely used in many sensitive fields and industries. However, despite the good predictive performance of the ML models, there are critical applications that fundamentally require trust like applications applied to medicine, driverless cars and law enforcement. Therefore, it becomes increasingly important to explain the behavior of those models and their output to enhance trust in the model predictions and their adoption in real world applications. This leads to a rapid growth in attention to eXplainable AI (XAI). The XAI methods can be grouped into pre-model (ante-hoc), in-model, and post-model (post-hoc) methods. In the latter, we identify two types of explanations: (1) symbolic (knowledge-driven) methods that are based on logical representations used for explanation (e.g. [18],[10]), verification and diagnostic purposes (e.g. [15], [17], [10]), and (2) numerical feature-based methods that provide insights into how much each feature contributed to that outcome (e.g. SHAP[13], LIME[16]). The main limitation to the existing explainability methods based on symbolic representations is the fact that they are generally intended to specific models and cannot be applied to any model

(non agnostic). In the other hand, feature-based methods such as LIME[16] and SHAP[13] try to assess the amount of contribution of features into the predictions but fail at answering certain questions such as *What are the feature values which are sufficient in order to trigger the prediction whatever the values of the other variables ?* or *Which values are sufficient to change in the instance x to have a different prediction ?*
To address the problem of answering this type of fundamental questions, we propose in this paper an approach to provide "different" and "complementary" types of symbolic explanations: the ***Sufficient Reasons*** **(**SRx **for short)** and the ***Counterfactuals*** **(**CFx **for short)**. The main advantage of our approach is the fact of being model-agnostic, where we try to learn accurate yet simple models to emulate the given black-box. In the other hand, the approach is based on rigorous and well-known Boolean satisfiability concepts that allow us to exploit the availability of efficient SAT solvers. Accordingly, our approach is declarative and does not require the implementation of specific algorithms. We evaluate the feasibility and efficiency of our approach on image classification task.

2 Preliminaries and notations

We first introduce the necessary notations and recall some definitions used in the remainder of this paper. For the sake of simplicity, we will limit the presentation to binary classifiers with binary features. We also focus only on negative predictions where the outcome is 0. As for explaining positive predictions where the outcome is 1, the approach applies similarly as discussed in the Conclusion.

Definition 1. ***(Binary Classifier)*** *A Binary Classifier is defined by two sets of variables: A feature space $X = \{X_1,...,X_n\}$ where $|X| = n$, and a binary class variable denoted Y. Both the features and the class variable take values in $\{0,1\}$.*

A decision function describes the classifier's behavior independently from the way it is implemented. We define it as a function $f : X \rightarrow Y$ mapping each instantiation x of X to $y=f(x)$. A data instance x is the feature vector associated with an instance of interest whose prediction from the ML model is to be explained. We use interchangeably in this paper f to designate the classifier and its decision function. Let us now define the representation framework we use.

Definition 2. ***(SAT : The Boolean Satisfiability problem)*** *Usually called SAT, the Boolean satisfiability problem is the decision problem, which, given a propositional logic formula, determines whether there is an assignment of propositional variables that makes the formula true.*

The logic formulas are built from propositional variables and Boolean connectors "AND" ($\wedge$), "OR" ($\vee$), "NOT" ($\neg$). A formula is satisfiable if there is an assignment of all variables that makes it true. It is said inconsistent or unsatisfiable otherwise. A complete assignment of variables making a formula true is called a model while a complete assignment making it false is called a countermodel.

Definition 3. ***(CNF (Clausal Normal Form))*** *is a set of clauses seen as a conjunction. A clause is a formula composed of a disjunction of literals. A literal*

is either a Boolean variable p or its negation ¬p. A quantifier-free formula is built from atomic formulae using conjunction ∧, disjunction ∨, and negation ¬. An interpretation μ assigns values from {0, 1} to every Boolean variable. Let Σ be a CNF formula, μ satisfies Σ iff μ satisfies all clauses of Σ.

Thanks to the achievements that the SAT field had known in the recent years, the modern SAT solvers[1] have gained in performance and efficiency where they can handle now problems with several million clauses and variables. Recall that we use a SAT oracle to generate our formal symbolic explanations. We encode the explanation generation problem as two common problems related to SAT-solving which are enumerating *minimal reasons why a formula is inconsistent* and *minimal changes to a formula* in order to restore its consistency. Indeed, in the case of an unsatisfiable CNF, we can analyze the inconsistency by enumerating sets of clauses causing the inconsistency (called Minimal Unsatisfiable Subsets (MUS)), and other sets of clauses allowing to restore its consistency (called Minimal Correction Subsets (MCS)). The enumeration of MUS/MCS are well-known problems dealt with in many areas such as knowledge-base reparation. Several approaches and tools have been proposed in the SAT community for their generation (e.g [12], [6]).

3 Overview of the proposed approach

The objective of our approach is explaining the prediction made by a classifier for a given input data instance x. It associates a logical representation that is almost equivalent to the decision function of the model to explain. Figure 1 represents an overview of the proposed approach.

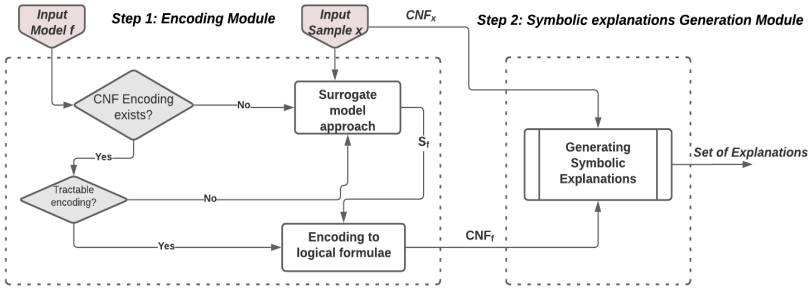

Fig. 1. A global overview of the proposed approach

Given a predictive model f, our approach proceeds as follows:

– **Step 1 (CNF encoding of the classifier)**: the goal is to encode f into an *equivalent* symbolic representation Σ_f. The generation of symbolic explanations in the next step is done using Σ_f. The encoding is done either by the means of model encoding algorithms if they are available and the encoding

[1] A SAT solver is a program for establishing the satisfiability of Boolean formulas encoded in conjunctive normal form.

remains tractable, or using a surrogate modeling approach as described in Section 4.

– **Step 2 (Explanation problem modeling)**: This step presented in Section 5 comes down to enumerating two types of symbolic explanations: SRx and CFx. This task is formulated as a partial maximum satisfiability problem Partial Max-SAT[2] using the CNF encoding Σ_f of f and Σ_x of the input instance x. The symbolic explanations respectively correspond to Minimal Unsatisfiable Subsets(MUS) and Minimal Correction Subsets(MCS) of $\Sigma_f \cup \Sigma_x$ in the SAT terminology.

The SAT solving is already used for providing some forms of symbolic explanations for some specific ML [3, 9, 11]. The novelty in our approach is the fact of being model-agnostic and succeeding in generating formal explanations based on rigorous concepts.

4 CNF encoding of classifiers

The starting point of our approach is the encoding of the input ML model f into a logical representation (CNF). This step is necessary in order to use a SAT oracle for the enumeration of the symbolic explanations. Mainly, two cases are considered: (1) an encoding of classifier f into an equivalent logical representation exists, in which case we can use it (e.g. Binarized Neural Networks (BNNs) [14], Naive and Latent-Tree Bayesian network [19]). (2) We consider the classifier f as a black-box and we use a surrogate model approach to approximate it in the vicinity of the instance to explain x (Agnostic option). The surrogate models are used to explain individual predictions of black-box ML models.
We set the focus on the agnostic option in this paper. This latter is applied when no direct CNF encoding exists for f or if the encoding is intractable.

Surrogate model encoding into CNF The approach proposed uses a surrogate model mainly characterized by its faithfulness to the initial model f (ensures same predictions) and its tractable logical representation (CNF). To ensure the local faithfulness to f, we use a surrogate model f_S trained on data instances in the vicinity of the data instance x whose prediction from the model needs to be explained. We construct the vicinity of x noted $V(x, r)$ by sampling new data instances within a radius r of x if the dataset is available[2].

A model that can guarantee a good trade-offs between faithfulness and giving a tractable CNF encoding is the one of random forests [7]. As shown in our experimental study, the random forest accuracy reflects a good level of faithfulness and its CNF encoding size remains tractable. The CNF encoding f_S of a classifier f should guarantee the equivalence of the two representations stated as follows :

Definition 4. ***(Equivalence of a classifier and its CNF encoding)*** *A binary classifier f (resp. f_S) can be equivalently encoded as a CNF Σ_f (resp. Σ_{f_S}) s.t. $f(x)$=1 (resp. $f_S(x)$=1) iff x is a model of Σ_f (resp. Σ_{f_S}).*

[2] Otherwise, we can draw new perturbed samples around x

Namely, data instances x predicted positively ($f(x)$=1) by the classifier are models of the CNF encoding the classifier. Similarly, data instances x predicted negatively ($f(x)$=0) are countermodels of the CNF encoding the classifier.

4.1 CNF encoding of random forests

In this work, we adopted the random forest[3] as the surrogate model f_S. Its associated CNF encoding resumes in i) encoding the decision trees individually and then ii) encoding the combination rule (which is a majority voting rule).
Encode in CNF every decision tree : Remember that all the features in our case are binary. Thus, each internal node of a decision tree DT_i represents a binary test on one of the features. The result of a test is either true or false. For the leaves of a decision tree, each one is annotated with the predicted class (namely, 0 or 1). The Boolean function encoded by a decision tree can be captured in CNF as the conjunction of the negation of paths leading to leaves labelled 0.
Encode in CNF the combination rule : Let y_i be a Boolean variable capturing the truth value of the CNF associated to a DT_i. Hence, the majority rule used in random forests to combine the predictions of m decision trees can be seen as a cardinality constraint[4] [20] that can be stated as follows :

$$y \Leftrightarrow \sum_{i=1..m} y_i \geq t, \tag{1}$$

where t is a threshold (usually t=$\frac{m}{2}$). To form the CNF corresponding to the entire random forest, it suffices to conjuct the m CNFs associated to the decisions trees, and, the CNF of the combination rule.

5 Generating symbolic explanations

This section will cover the presentation of both *Sufficient Reasons* and *Counterfactuals* explanations as well as the SAT-based setting we use to generate such explanations. This corresponds to **Step 2** within the Fig.1. This step takes as input the CNF encoding of a classifier Σ_f and a sample data instance Σ_x.

5.1 A SAT-based setting for the enumeration of explanations

We propose two complementary types of symbolic explanations: the *Sufficient Reasons* which are a minimal subset of the input data, that if fixed, lead to a given prediction and the *Counterfactuals* which are a minimal subset of the input data that we can act on to obtain a different outcome. The enumeration of those symbolic explanations in our approach is based on two very common concepts in SAT which are MUS and MCS (defined formally in the following). We use a variant of the SAT problem called Partial-Max SAT [2] in order to restrict the explanations only to clauses encoding the input data x and do not include clauses that encode the classifier f.

[3] Random Forests are used for XAI purposes in some works such was [1, 4, 11]
[4] In our case this constraint means that at least t decision trees predicted the label 1.

The Partial-Max SAT problem can be efficiently solved by the existing tools implementing the enumeration of MUSes and MCSes such as the tool in [5]. It is composed of two disjoint sets of clauses where Σ_H denotes the hard clauses (those that could not be relaxed) and Σ_S denotes the soft ones (those that could be relaxed). Concretely in our approach, the set of hard clauses corresponds to Σ_f and the soft clauses to Σ_x. The CNF Σ_x encoding the data instance x is formed by unit clauses where each clause $\alpha \in \Sigma_x$ is composed of exactly one literal ($\forall \alpha \in \Sigma_x, |\alpha| = 1$) and each literal representing a Boolean variable of Σ_x corresponds to a Boolean variable $\{X_i \in X\}$ where X is the feature space of f. Thanks to this Partial-MAX SAT setting, it is possible to both identify the subsets of Σ_x responsible for the unsatisfiability of a given CNF $\Sigma_f \cup \Sigma_x$ (corresponding to SRx of $f(x)$=0), and the subsets allowing to restore the consistency of $\Sigma_f \cup \Sigma_x$ (corresponding to CFx allowing to change the prediction to $f(x)$=1).

5.2 Sufficient Reason Explanations ($\mathbf{SR_x}$)

We are trying to find explanations that identify the relevant variables that could justify why the prediction is negative. This is carried out by identifying a subset of our input which causes the inconsistency of the CNF formula $\Sigma_f \cup \Sigma_x$ (recall that the prediction $f(x)$ is captured by the truth value of $\Sigma_f \cup \Sigma_x$). The identified subsets of the input x represent *Sufficient Reasons* for the prediction to be negative. We formally define the *Sufficient Reasons* explanations as follow:

Definition 5. ***(SR_x explanations)*** *Let x be a data instance and $f(x)$=0 its prediction by the classifier f. A sufficient reason explanation $\tilde{x}$ of x is such that:*

1. *$\tilde{x} \subseteq x$ ($\tilde{x}$ is a part of x)*
2. *$\forall \acute{x}$, $\tilde{x} \subset \acute{x} : f(\acute{x})=f(x)$ ($\tilde{x}$ suffices to trigger the prediction)*
3. *There is no partial instance $\hat{x} \subset \tilde{x}$ satisfying 1 and 2 (minimality)*

Intuitively, a *sufficient reason* $\tilde{x}$ is defined as the part of the data instance x such that $\tilde{x}$ is minimal and causes the prediction $f(x)$=0. Namely, to explain the classification it is "sufficient" to observe those features with disregard to the others. We define now the Minimal Unsatisfiable Subsets :

Definition 6. ***(MUS)*** *A Minimal Unsatisfiable Subset (MUS) is a minimal subset Γ of clauses of a CNF Σ such that $\forall\ \alpha \in \Gamma$, $\Gamma \backslash \{\alpha\}$ is satisfiable.*

A MUS for $\Sigma_f \cup \Sigma_x$ comes down to a subset of soft clauses, namely a part of x that is causing the inconsistency, hence the prediction $f(x)$=0.

Proposition 1. *Let f be a classifier, let Σ_f be its CNF representation. Let also x be a data instance predicted negatively ($f(x) = 0$) and $\Sigma_f \cup \Sigma_x$ the corresponding Partial Max-SAT encoding. Let $SR(x, f)$ be the set of Sufficient Reasons of x wrt. f. Let $MUS(\Sigma_{f,x})$ be the set of MUses of $\Sigma_f \cup \Sigma_x$. Then:*

$$\forall \tilde{x} \subseteq x, \tilde{x} \in SR(x, f) \iff \tilde{x} \in MUS(\Sigma_{f,x}) \tag{2}$$

Proposition 1 states that each MUS of the CNF $\Sigma_f \cup \Sigma_x$ is a *Sufficient Reason* for the prediction $f(x)$=0 and vice versa.

5.3 Counterfactual Explanations ($\mathbf{CF}_x$)

We are also interested in another type of explanation which would allow us to figure out what changes can be made to the input data in order to alter the initial outcome. Let us formally define the concept of counterfactual explanation.

Definition 7. *($\mathbf{CF}_x$ Explanations) Let x be a complete data instance and $f(x)$ its prediction by the decision function of f. A counterfactual explanation $\tilde{x}$ of x is such that:*

1. *$\tilde{x} \subseteq x$ ($\tilde{x}$ is a part of x)*
2. *$f(x[\vec{\tilde{x}}]) =$ 1-$f(x)$ (prediction inversion)*
3. *There is no $\hat{x} \subset \tilde{x}$ such that $f(x[\hat{x}])=f(x[\vec{\tilde{x}}])$ (minimality)*

In definition 7, the term $x[\vec{\tilde{x}}]$ denotes the data instance x where variables included in $\tilde{x}$ are inverted. In our approach, *Counterfactuals* are enumerated thanks to the Minimal Correction Subset enumeration [5].

Definition 8. ***(MCS)*** *A Minimal Correction Subset Ψ of a CNF Σ is a set of formulas $\Psi \subseteq \Sigma$ whose complement in Σ, i.e., $\Sigma \setminus \Psi$, is a maximal satisfiable subset of Σ.*

Following our modeling, an MCS for $\Sigma_f \cup \Sigma_x$ comes down to a subset of soft clauses denoted $\tilde{x}$, namely a part of x that is enough to remove (or reverse) in order to restore the consistency, hence to alter the prediction $f(x)$=0 to $f(x[\vec{\tilde{x}}])$=1.

Proposition 2. *Let f be the decision function of the classifier, let Σ_f be its CNF representation. Let also x be a data instance predicted negatively ($f(x) = 0$) and $\Sigma_f \cup \Sigma_x$ the corresponding Partial Max-SAT encoding. Let $CFx(x, f)$ be the set of counterfactuals of x wrt. f. Let $MCS(\Sigma_{f,x})$ the set of MCSs of $\Sigma_f \cup \Sigma_x$. Then:*

$$\forall \tilde{x} \subseteq x, \tilde{x} \in CFx(x, f) \iff \tilde{x} \in MCS(\Sigma_{f,x}) \tag{3}$$

Proposition 2 states that each MCS of the CNF $\Sigma_f \cup \Sigma_x$ represents a CF $\tilde{x} \subseteq x$ for the prediction $f(x)$=0 and vice versa.

6 Empirical evaluation

Experimentation set-up The black-box models considered are *"one-vs-all"* binary neural networks (BNNs)[5] trained on the widely used MNIST database [6]. MNIST is composed of 70,000 images of size 28×28 pixels. We use the pytorch implementation[7] of the Binary-Backpropagation algorithm "BinaryNets"[8] to train the BNN classifiers (one per digit from 0 to 9) on the binarized images (threshold $T = 127$). All experiments have been conducted on Intel Core i7-7700 (3.60GHz $\times$8) processors with 32Gb memory on Linux.

[5] defined as a neural networks with binary weights and activations at run-time
[6] MNIST: handwritten digit databse, available at http://yann.lecun.com/exdb/mnist/
[7] available at: https://github.com/itayhubara/BinaryNet.pytorch

Results The surrogate model considered is a random forest (RF) classifier trained on the vicinity of the input sample using the hyper-parameters $nb\_trees = 10$ and $max\_depth = 24$. We try out different values for the radius r but we only present the results for $r = 250$ with an average of 200 neighbors around x due to the limited number of pages. The black-box models (BNNs) trained to recognize the "0","2","5","6" and "8" digits are used as predictive models [8]. Around 1000 to 1500 images were picked randomly from the MNIST database for the experimental study conducted on each classifier.

Evaluating the CNF encoding in practice : We are interested in evaluating the size of the CNF encoding using the setting mentioned above. We use the Tseitin Transformation [21] to encode the propositional formulae into an equisatisfiable CNF formulae. The size of this latter is linear in the size of the original formulae. The results are presented in **Table 1**. The high accuracy of f_S shows that the generated RF classifier provide interesting results in term of fidelity. The number of variables/clauses of the CNF indicates that the logical representation remains tractable and makes the logical representation easily handled by the current SAT-solvers which confirms the feasibility of the approach.

	MNIST_0	MNIST_2	MNIST_5	MNIST_6	MNIST_8
avg acc of RF	98%	93%	99%	96%	95%
min size CNF	1744/4944	1941/5452	2196/6102	1978/5534	1837/5178
avg size CNF	1979/5540	2172/6050	2481/6856	2270/6293	2059/5727
max size CNF	2176/6066	2429/6760	2789/7694	2558/7028	2330/6408
min enc_runtime (s)	0.83	0.88	0.92	0.82	0.74
avg enc_runtime (s)	1.05	1.06	1.11	0.92	0.86
max enc_runtime (s)	1.51	1.92	1.56	1.31	1.32
min #CFs	10	13	10	15	6
avg #CFs	35790	63916	99174	79520	4846
max #CFs	285219	546005	633416	640868	65554
min enumtime (s)	0.005	0.11	0.006	0.11	0.008
avg enumtime (s)	21.49	42.11	77.72	50.86	2.35
max enumtime (s)	234.18	600	600	531.16	35.08

Table 1. Evaluating (1) the encoding into the logical representation and (2) the enumeration of explanations for different classifiers used to locally explain MNIST images.

Evaluating the feasibility of the enumeration of explanations : We want to assess the practical feasibility of the enumeration of *Sufficient Reasons* and *Counterfactual* explanations. To enumerate the CFx, we use the *EnumELSRMRCache tool*[1] with a timeout set to 600s. Thanks to the duality between MUSes and MCSes, the enumeration of SRx can be done by computing the minimal hitting set of CFx. However, the results in this paper only cover the enumeration of CFx due to the page limitation.

In **Table 1**, we report the average run-time (enumtime) needed to enumerate all the explanations within the timeout whereas in reality the solver manages to find explanations instantly in the majority of cases. Accordingly, the enumeration time remains reasonable and shows the practical feasibility of the enumeration of such explanations for medium size classifiers (like the BNNs used). Additionally,

[8] results for the other digits are similar but not be reported because of space limitation

[1] implementing the boosting algorithm for MCSes enumeration proposed in [5]

we notice that the number of CFx enumerated is significant and that it quickly becomes unmanageable for a user to process the result. This reinforces the need for quality metrics to filter the generated explanations.

$\mathbf{SR}_x$ and $\mathbf{CF}_x$ for MNIST : We use the "one-vs-all" BNN f_8 trained to recognize the eight "8" digit (positive prediction for an image representing "8", negative otherwise) that has achieved an accuracy of 97%. Fig.2 shows a few data samples negatively predicted by f_8. Heatmaps in the 3^{rd} column of Fig.2 show examples of CFx highlighting the necessary changes to be made on the input data sample in order to alter the outcome of f_8 from negative to positive. We can visually distinguish a sort of pattern of the digit "8" highlighting the pixels we need to act on. This actually matches the definition of CFx. Although the underlying mechanisms of our approach and SHAP differ which may lead to very different explanations for the same input, we can see that our explanations are visually simpler, clearer and easier to understand and use compared to SHAP explanations in the last column.

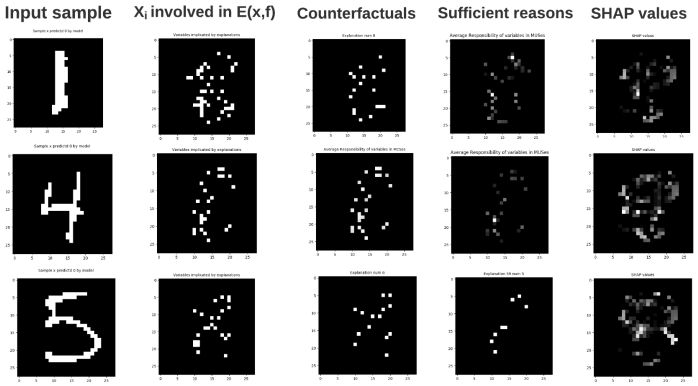

Fig. 2. Data samples from MNIST predicted negatively by f_8 in the 1^{st} column. The heatmap of the: 2^{nd} column represent the variables involved by the explanations, the 3^{th} and 4^{th} columns, a single counterfactual and sufficient reason explanation. The last column is the SHAP values of the variables contributing positively to the prediction.

7 Concluding remarks and Discussions

We try to explain individual outcomes of black-box models by the mean of a novel model agnostic generic approach presented within this paper in order to provide two complementary types of explanations: *Sufficient reasons* and *Counterfactuals*. The approach is based on the Boolean satisfiability concepts which allow us to take advantage of the strengths of already existing and proven solutions, and the powerful practical tools for the generation of MCS/MUS. We use the notion of surrogate model to overcome the complexity of encoding a ML classifier into an equivalent logical representation. It is a local encoding since we approximate the original model in the vicinity of the sample of interest. The same mechanism is used to explain positively predicted instances. It suffices to

work with the negation of the representation of f ($\neg f$) to enumerate the explanations in a similar way. We intend in future works to assess the relevance of explanations and features individually w.r.t a set of properties allowing to evaluate explanations in ways that are closer to how users consume them.

Acknowledgment : This work was supported by the Région Hauts-de-France.

References

1. Audemard, G., Koriche, F., Marquis, P.: On tractable xai queries based on compiled representations. In: Proceedings of the International Conference on Principles of Knowledge Representation and Reasoning. vol. 17, pp. 838–849 (2020)
2. Biere, A., Heule, M., van Maaren, H.: Handbook of satisfiability, vol. 185. IOS press (2009)
3. Boumazouza, R., Cheikh-Alili, F., Mazure, B., Tabia, K.: A symbolic approach for counterfactual explanations. In: International Conference on Scalable Uncertainty Management. pp. 270–277. Springer (2020)
4. Choi, A., Shih, A., Goyanka, A., Darwiche, A.: On symbolically encoding the behavior of random forests. arXiv preprint arXiv:2007.01493 (2020)
5. Grégoire, É., Izza, Y., Lagniez, J.M.: Boosting mcses enumeration. In: IJCAI. pp. 1309–1315 (2018)
6. Grégoire, E., Mazure, B., Piette, C.: Boosting a complete technique to find mss and mus thanks to a local search oracle. In: IJCAI-07. vol. 7, pp. 2300–2305 (2007)
7. Ho, T.K.: Random decision forests. In: Proceedings of 3rd international conference on document analysis and recognition. vol. 1, pp. 278–282. IEEE (1995)
8. Hubara, I., Courbariaux, M., Soudry, D., El-Yaniv, R., Bengio, Y.: Binarized neural networks. In: Advances in Neural Information Processing Systems. vol. 29 (2016)
9. Ignatiev, A., Marques-Silva, J.: Sat-based rigorous explanations for decision lists. arXiv preprint arXiv:2105.06782 (2021)
10. Ignatiev, A., Narodytska, N., Marques-Silva, J.: On relating explanations and adversarial examples. In: Advances in Neural Information Processing Systems. vol. 32 (2019)
11. Izza, Y., Marques-Silva, J.: On explaining random forests with sat. arXiv preprint arXiv:2105.10278 (2021)
12. Liffiton, M.H., Sakallah, K.A.: Algorithms for computing minimal unsatisfiable subsets of constraints. Journal of Automated Reasoning **40**(1), 1–33 (2008)
13. Lundberg, S.M., Lee, S.I.: A unified approach to interpreting model predictions. In: Guyon, I., Luxburg, U.V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., Garnett, R. (eds.) Advances in Neural Information Processing Systems. vol. 30. Curran Associates, Inc (2017), https://proceedings.neurips.cc/paper/2017/file/8a20a8621978632d76c43dfd28b67767-Paper.pdf
14. Narodytska, N., Kasiviswanathan, S., Ryzhyk, L., Sagiv, M., Walsh, T.: Verifying properties of binarized deep neural networks. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 32 (2018)
15. Reiter, R.: A theory of diagnosis from first principles. Artificial intelligence **32**(1), 57–95 (1987)
16. Ribeiro, M.T., Singh, S., Guestrin, C.: " 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. pp. 1135–1144 (2016)

17. Rymon, R.: An se-tree-based prime implicant generation algorithm. Annals of Mathematics and Artificial Intelligence **11**(1), 351–365 (1994)
18. Shih, A., Choi, A., Darwiche, A.: A symbolic approach to explaining bayesian network classifiers. In: IJCAI-18. pp. 5103–5111. International Joint Conferences on Artificial Intelligence Organization (7 2018). https://doi.org/10.24963/ijcai.2018/708, https://doi.org/10.24963/ijcai.2018/708
19. Shih, A., Choi, A., Darwiche, A.: Compiling bayesian network classifiers into decision graphs. In: Proceedings of the AAAI-19. vol. 33, pp. 7966–7974 (2019)
20. Sinz, C.: Towards an optimal cnf encoding of boolean cardinality constraints. In: International conference on principles and practice of constraint programming. pp. 827–831. Springer (2005)
21. Tseitin, G.S.: On the complexity of derivation in propositional calculus. In: Automation of reasoning, pp. 466–483. Springer (1983)