HAL
open science

Towards a framework for challenging ML-based decisions

Clément Henin, Daniel Le Métayer

To cite this version:

Clément Henin, Daniel Le Métayer. Towards a framework for challenging ML-based decisions. DeceptECAI 2020 - 1st International Workshop on Deceptive AI @ECAI2020, Aug 2020, Santiago de Chili, Chile. pp.1-13. hal-02932467

HAL Id: hal-02932467

https://inria.hal.science/hal-02932467v1

Submitted on 7 Sep 2020

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Towards a framework for challenging ML-based decisions

Clément Henin[1,2] ⋆ and Daniel Le Métayer[1]

[1] Univ Lyon, Inria, INSA Lyon, CITI, Villeurbanne, France {clement.henin, daniel.le-metayer}@inria.fr
[2] École des Ponts ParisTech, Champs-sur-Marne, France

Abstract. The goal of the work presented in this paper is to provide techniques to challenge the results of an algorithmic decision system relying on machine learning. We highlight the differences between explanations and justifications and outline a framework to generate evidence to support or to dismiss challenges. We also present the results of a preliminary study to assess users' perception of the different types of challenges proposed here and their benefits to detect incorrect results.

Keywords: challenge · justification · machine learning · training · dataset · evidence.

1 Introduction

When Machine Learning (ML) is used in a decision process, which is increasingly common, a key issue is the trust that can be placed in the system. Trusting an ML system in this context cannot be taken for granted, especially if the decision can have a significant impact on the affected people. Indeed, it is well-known that these systems can go wrong for many reasons, intentional or not. The goal of the work presented in this paper is to provide techniques to challenge the results of an Algorithmic Decision System (hereinafter "ADS") relying on ML and to reply to these challenges. We assume that the code of the ADS is not available or accessible; therefore, we have to follow a "black box" analysis approach. A first idea to achieve this goal would be to resort to post-hoc explanations. Explainable AI has become a very active research area in recent years and many explanation methods have been published in the literature. However, explanations fall short of our objective for different reasons. The first and most fundamental reason is that their goal is to ensure that the results of the system can be understood by humans, not that they are necessarily correct, or "good". Understanding is obviously a precondition for trust but it is not sufficient, as discussed below. In order to address this issue, we focus on *justifications* rather than explanations here. The word "justification" has been used in the XAI litterature, but very often without any precise characterization and sometimes as a synonym of "explanation". In this paper, we propose a precise definition of justifications and

⋆ Corresponding author (clement.henin@inria.fr)

highlight the differences between explanations and justifications in Section 2. Then, we outline our interactive framework based on challenges and justifications in Section 3. The results of a preliminary study to assess users' perception of the different types of justifications proposed here and their benefits to detect incorrect results are sketched in Section 4. We discuss related work in Section 5 and conclude with avenues for further research in Section 6.

2 Challenges and justifications

As suggested in the introduction, we believe that a clear distinction should be made between explanations and justifications. Many definitions have been proposed for these terms which are also used interchangeably by some authors. In this paper, we propose the following characterizations, which are consistent with [15]:

- The goal of an explanation is to make it possible for a human being (designer, user, affected person, etc.) to understand (a result or the whole system). In contrast, the goal of a justification is to make it possible for a human being to challenge (a result or the whole system) or to enhance his trust in the system (or a particular result). Even if they often support each other, the two goals are different: a user can understand the logic leading to the production of a particular result without agreeing on the fact that this result is correct or good; vice versa, he may want to challenge a result (being convinced that it is incorrect or bad) without understanding the logic behind the algorithm.
- Explanations are descriptive and intrinsic in the sense that they only depend on the system itself[3] . In contrast, justifications are normative and extrinsinc in the sense that they depend on a reference (or a norm) according to which the correctness or quality of the results can be assessed. Indeed, in order to claim that a result is correct or good, it is necessary to refer to an independent definition of what a correct or good result is.

Considering that our goal in this paper is to help human beings to challenge the results of an ADS or to enhance their trust in these results, we focus on justifications here. Technically speaking, to design a system allowing users to challenge a decision based on an ADS, we first need to define precisely the types of challenges that the user can express and the ways for the system to address these challenges, that is to say to produce justifications. Different forms of challenges and justifications are conceivable. For example, justifications can refer to explicit norms such as "the gender attribute must not have any impact on the results" or to implicit norms expressed through datasets of previous decisions. In this paper, we focus on ADS relying on machine learning and justifications

[3] This is also the case for "causal explanations" : even though the notion of cause is very complex and it is used with a variety of different meanings in the literature, causal explanations are generally based on relations between ADS inputs and outputs, without reference to any external norm [2].

expressed with respect to the learning dataset. This dataset is therefore considered as the reference for correct or "good" decisions, which implies that much attention must be paid to reviewing it and ensuring that it is indeed representative of the requirements for the ADS. If it is not the case, for instance if the dataset is biased against a minority group, then justifications may be useful to highlight the bias. For example, if a justification relies on a racial bias and it turns out to be technically valid (i.e. supported by the learning dataset), then the affected person can challenge the system itself and bring the case to court. In this regard, we should stress the fact that such a justification system should be seen as a support tool for individuals and human decision makers rather than an automatic tool to establish the legitimacy of a challenge or a justification. In practice, the dataset is used by the system to generate (1) evidence to support the challenge or (2) evidence to the contrary (to support a justification for the decision) or (3) both types of evidence (when the dataset is not conclusive).

The precise definitions of challenges and justifications and the generation of evidence to support them are presented in the next section.

3 Outline of the framework

In this paper, we focus on dynamic or interactive justifications, that is to say justifications that are produced as a reply to a given challenge, leading to an interactive challenge and justification process. The intuition is that interactive justifications are more likely to address the concerns of the challenger. This is also in line with the current trend towards interactive explanations. In the case of an ADS based on machine learning, justifying a decision amounts to convince the user that the decision is consistent with the training samples.

We present the two types of challenges considered here in Section 3.1 and justifications in Section 3.2. Then we sketch the algorithm used to produce evidence to support a challenge or a justification in Section 3.3. The outcomes of the process are described in Section 3.4. For the sake of simplicity, we assume that the decision, which is taken by a machine learning algorithm D, is binary[4] with output 1 associated with a positive decision (e.g. credit request accepted) and output 0 associated with a negative decision (e.g. credit request rejected). A challenge is used by the "plaintiff" to reverse a decision (i.e. to obtain a positive decision) while a justification is produced by the "defendant" to support a decision.

3.1 Challenges

In order to illustrate the different types of challenges considered here, let us take as an example of plaintiff a student whose application to a university has been rejected. A possible way to challenge the decision would be for the student to argue that his or her application should have been accepted because he or she

[4] The framework can be easily extended to non-binary decisions.

has an average grade of 16 in mathematics and 15 in geography. The implicit rule used by the student, which we call an *absolute claim*, is: "any application with an average grade equal to (or greater than) 16 in mathematics and 15 in geography should be accepted". It is worth noting that the claim used by the student does not need to involve all grades. For example, if the student has an average grade of 7 in English, it is not in his or her interest to put forward this grade in a claim.

The general definition of this type of claim is provided by Equation 1:

$$\forall x \in \Delta,\ C(x) \implies D(x) = 1 \tag{1}$$

where Δ is the set of input data (e.g. average grade records for the university application example), C is the condition and D the decision. For absolute claims, we assume that condition C takes the form of a conjunction of properties of the attributes of its argument. In the above example, we have $C(x) = x.maths \geq 16 \wedge x.geo \geq 15$.

Another way to challenge the decision would be for the student to compare his or her application with the application of another student who has been accepted. For example, the student having better average grades than this other student in mathematics and in geography could argue that he or she should therefore be accepted also. In this case, the implicit rule used by the student, which we call a *relative claim*, is: "if student S_1 has better average grades than student S_2 in mathematics and in geography and S_2's application is accepted, then S_1's application should also be accepted".

The general definition of this type of claim is provided by Equation 2.

$$\forall x \in \Delta, \exists y \in \Delta,\ D(y) = 1 \wedge C(x, y) \implies D(x) = 1 \tag{2}$$

In the above example, we have $C(x, y) = x.maths \geq y.maths \wedge x.geo \geq y.geo$.

3.2 Justifications

The notation introduced in the previous section makes it possible to express challenges based on claims but we have not discussed the validity of these claims so far. For example, it is possible to define nonsensical claims such as: "any application with an average grade less than 10 in mathematics should be accepted". The next step is therefore to provide ways to process a claim and to build a reply. To this respect, it is worth pointing out that the reference for the assessment of a claim is the training dataset of the ADS. In other words, we consider a statistical setting rather than a logical framework here. Let us assume, for example, that in the training dataset of the ADS, 70 % of the 600 applicants with grades above 16 in mathematics and above 15 in geography were accepted. At first sight, the challenge of the student would seem legitimate. However, a more precise analysis may show that none of the 125 applications having grades above 16 in mathematics, above 15 in geography, but less than 8 in English were accepted. This justification may reflect the fact that a minimum

grade in English is necessary to enter this university. Generally speaking, a justification is a refinement of the claim of the challenge providing evidence that the decision for the case under consideration is justified by the training dataset. By refinement, we mean the conjunction of the claim with further conditions on the attributes of the case under consideration ($x.english \leq 8$ in the above example). In this example, the refinement of the claim is provided by Equation 3 with $C(x) = x.maths \geq 16 \wedge x.geo \geq 15 \wedge x.english \leq 8$.

$$\forall x \in \Delta,\ C(x) \implies D(x) = 0 \tag{3}$$

As discussed above, both challenges and justifications rely on claims that should be supported by evidence extracted from the training data set. In the next section, we show how this evidence can be generated and how the resulting claims can be assessed.

3.3 Generation of evidence

In the example discussed in the previous section, the evidence for the justification seems convincing because its coverage (the size of the group) is not too low (125) and the ratio (0 %) leaves little room for doubt. The two main outstanding issues at this stage are the following : first, how to generate such evidence and then, how to assess and compare them ? The goal of a justification system is to exploit the training dataset to generate the strongest evidence to support or to dismiss a claim. Since bigger groups tend to have a ratio closer to the population average, there is usually a tension between high ratio and high coverage objectives. Finding an acceptable compromise between these two objectives is the biggest challenge to generate strong evidence. Before getting into more technical details about the generation algorithm, it is important to note that the system can be used to generate both evidence to support a challenge (hence in favour of the plaintiff) and evidence to support a justification (hence in favour of the defendant). This is all more important given that the plaintiff may not have access to the training dataset or have the required expertise to produce on his first try the best challenge.

Technically speaking, a justification is a refinement of the initial challenge, that is to say the conjunction of the property defining the challenge and additional conditions on attributes. Therefore, the generation of evidence amounts to a rule mining problem: the goal is to find the set of rules achieving the best compromise between the ratio and coverage objectives. To avoid exponential complexity in the number of rules, we use a greedy algorithm adding at each step the best rule according to a heuristic selection process. More precisely, the algorithm enumerates all possible rules and selects the candidate leading to the best ratio/coverage compromise. To this aim, we use the p-value of a T-test to select the rule that defines the subset of the training data which is the most significantly different from the subset of the preceding iteration. Only evidence with a p-value below a certain threshold is considered as valid. If no rule meeting this requirement can be found, then the algorithm returns en empty result. More details and the pseudo-code can be found in B.

3.4 Possible outcomes

When the evidence supporting the challenge is weak and strong evidence can be provided to support a justification, the outcome of the process described in the previous section should be that the challenge is not valid. Vice versa, when evidence can be generated to support a challenge but the claim cannot be refined to generate a justification, the outcome should be that the challenge is valid. However, there are situations in which reasonable evidence can be produced to support both a challenge and a justification of the decision. The next issue to address is therefore the assessment of the strength of evidence data and the comparison between different types of evidence. The first rule is that if Evidence E_1 has a higher coverage and higher ratio[5] than Evidence E_2, then E_1 is stronger than E_2. However, the outcome is less obvious when E_1 has higher coverage than E_2 but E_2 has higher ratio. Different heuristics can be used to deal with this kind of cases. For example, a threshold can be defined to filter out evidence with a too small coverage, which should be considered as non-significant.

In some situations, the outcome of the process can be non-conclusive, meaning that reasonable evidence is available to support both the challenge and the justification. This type of conclusion should not be seen as a failure of the system as such decisions are likely to be close to a decision boundary of the ADS and may thus be open to discussion. In any case, it should be emphasized that the outcomes of a justification system should always be seen as suggestions to a human agent in charge of taking the final decision (i.e. validating the challenge or the justification).

4 User study

In order to evaluate the efficiency of the techniques proposed here to detect unjustified decisions and the intuitive nature of our framework, we conducted a user study using an online platform. Considering that the design of the framework is still in progress, this study should be seen as a preparatory step for a more ambitious user validation of the final version. The experimental protocol was the following: decisions were presented to the users, some of these decisions were produced by the ADS, while others had purposely been modified in such a way that they were not supported by the training data. Users were asked to find which decisions were unjustified in different contexts corresponding to the possibility to issue a given type of challenge and to receive a justification in return.

The analysis of the empirical data consisted of a comparison of the proportions of successfully identified unjustified decisions for users benefitting from one of our approaches (absolute claims and relative claims) or not (considering three baseline scenarios). We also collected the levels of confidence declared by the users on a 5-Likert scale. Although the relative claims and justifications defined

[5] Ratio of decisions equal to 1 for challenge claims and ratio of decisions equal to 0 for justification claims.

by Equation 2 performed better than all baseline scenarios, the differences are not significant (see the results in Appendix A). Thus the evaluation does not provide strong evidence of the benefit of this type of help to detect unjustified decisions. Several reasons may explain these non-conclusive results. First, the sample size is rather small (16 people). Second, the interface should be more self-explanatory, both about the approach and about the use case. Indeed, most users did not have any expertise in the justice sector (to which the decision system was related) and some of them explicitly complained about the fact that they did not understand very well the impact of the different features on the decision. Last but not least, more effort should be made to improve the case study to put users in a situation that would reflect as closely as possible a real decision challenging situation. Therefore, although the experiment is not conclusive, it provides good insights on the way to improve it to assess the final version of the system. Further details on the study can be found in Appendix A.

5 Related works

In the field of explainable AI, the distinction between explanations and justifications is sometimes blurred. However, a series of works [5],[6],[7], [15], [18], [19] refer to justifications as ways of ensuring that a decision is good (in contrast to understanding a decision), which is in line with the approach followed in this paper. However, the need to refer to an extrinsic norm is usually not mentioned explicitly. In addition, previous work does not involve the notion of challenge and the generation of interactive justifications. In other papers [13], justifications are seen as ways to make understandable the inner operations of a complex system (in a white-box setting). The normative nature of justifications was mentioned in the field of intelligent systems [11]: "an intelligent system exhibits justified agency if it follows society's norms and explains its activities in those terms". However, these norms are not characterized precisely, in particular the role of the training data is not mentioned. On this matter, [12] qualifies explanations as "unjustified" when there are not supported by training data. Therefore, justifiability applies to explanations in this context rather than to the decisions themselves. From a different perspective, [8], introduces several justifications in machine learning that aim at justifying an ADS as a whole rather than individual decisions. To the best of our knowledge, none of these contributions introduce a precise definition of justification of a decision based on a machine learning algorithm, or define a practical framework to challenge and to justify decisions.

The interest for more interactions with humans when conceiving and using ML takes several forms [1]. The need to conceive explanations as an interactive process has been argued by several authors [16], [17]. The "human-in-the-loop" approach leverages on human feedback during the training process to obtain more accurate classifiers [10]. A lot of work has also been done on argumentation and dialog games [3, 4, 14] but the focus in these areas is generally the logical structure of the framework to express and to relate arguments or the protocol to exchange arguments. In contrast, we take an empirical approach to

assess challenges and justifications here and we consider a very basic protocol (challenge-justification sequences). Closer to our work, [9] relies on "debates" between two competing algorithms exchanging arguments and counterarguments to convince a human user that their classification is correct. However, the goal of this work is to "align an agent's actions with the values and preferences of humans" which is seen as a "training-time problem". Our objective in this paper was different but an interesting avenue for further research could be the application of our approach to design or to improve an ADS.

6 Conclusion

The need to provide ways to challenge the results of an ADS is often highlighted but, to our best knowledge, no dedicated framework has been proposed so far. In this paper we have presented a work in progress whose ultimate goal is precisely to address this need. As mentioned earlier, challenges and justifications can take many different forms. In this paper, we have focused on ADS relying on machine learning and justifications expressed with respect to the learning dataset. In practice, justifications can also refer to explicit norms (e.g. legal or ethical norms) expressed by logical rules. Furthermore, norms may be mandatory or relative. An example of relative norm could be "minimize payment defaults" in a bank credit ADS. Similarly, different types of justifications can be generated depending on the available data (learning data set, ground truth data, historical ADS data, etc.). However, beyond these differences, a common "challenge and justification" framework can be defined to accomodate different types of situations and provide more control to human beings on ADS. The work outlined in this paper is a first step towards this framework, which is currently under construction.

References

1. Abdul, A., Vermeulen, J., Wang, D., Lim, B.Y., Kankanhalli, M.: Trends and trajectories for explainable, accountable and intelligible systems: An HCI research agenda. In: Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems - CHI '18. p. 1–18. ACM Press (2018). https://doi.org/10.1145/3173574.3174156, http://dl.acm.org/citation.cfm?doid=3173574.3174156
2. Alvarez-Melis, D., Jaakkola, T.S.: A causal framework for explaining the predictions of black-box sequence-to-sequence models (2017)
3. Atkinson, K., Baroni, P., Giacomin, M., Hunter, A., Prakken, H., Reed, C., Simari, G., Thimm, M., Villata, S.: Towards artificial argumentation. AI Magazine **38**(3), 25–36 (Oct 2017). https://doi.org/10.1609/aimag.v38i3.2704, https://www.aaai.org/ojs/index.php/aimagazine/article/view/2704
4. Bex, F., Walton, D.: Combining explanation and argumentation in dialogue. Argument and Computation **7**(1), 55–68 (2011)
5. Biran, O., Cotton, C.: Explanation and justification in machine learning: A survey. In: IJCAI-17 Workshop on Explainable AI (XAI). p. 8 (2017)
6. Biran, O., McKeown, K.R.: Justification narratives for individual classifications. In: ICML (2014)
7. Biran, O., McKeown, K.R.: Human-centric justification of machine learning predictions. In: IJCAI. p. 1461–1467 (2017)
8. Corfield, D.: Varieties of justification in machine learning. Minds and Machines **20**(2), 291–301 (Jul 2010). https://doi.org/10.1007/s11023-010-9191-1
9. Irving, G., Christiano, P., Amodei, D.: AI safety via debate. arXiv:1805.00899 [cs, stat] (Oct 2018), http://arxiv.org/abs/1805.00899, arXiv: 1805.00899
10. Kim, B.: Interactive and interpretable machine learning models for human machine collaboration p. 143
11. Langley, P.: Explainable, normative, and justified agency. Proceedings of the AAAI Conference on Artificial Intelligence **33**, 9775–9779 (Jul 2019). https://doi.org/10.1609/aaai.v33i01.33019775
12. Laugel, T., Lesot, M.J., Marsala, C., Renard, X., Detyniecki, M.: The dangers of post-hoc interpretability: Unjustified counterfactual explanations. arXiv:1907.09294 [cs, stat] (Jul 2019), http://arxiv.org/abs/1907.09294, arXiv: 1907.09294
13. Lei, T., Barzilay, R., Jaakkola, T.: Rationalizing neural predictions. In: Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing. p. 107–117. Association for Computational Linguistics (2016). https://doi.org/10.18653/v1/D16-1011, http://aclweb.org/anthology/D16-1011
14. Madumal, P., Miller, T., Sonenberg, L., Vetere, F.: A grounded interaction protocol for explainable artificial intelligence. In: Proceedings of the 18th International Conference on Autonomous Agents and MultiAgent Systems. p. 1033–1041. AAMAS '19, International Foundation for Autonomous Agents and Multiagent Systems, Richland, SC (2019)
15. Miller, T.: Explanation in artificial intelligence: Insights from the social sciences. Artificial Intelligence **267** (2017). https://doi.org/10.1016/j.artint.2018.07.007
16. Miller, T., Howe, P., Sonenberg, L.: Explainable AI: Beware of inmates running the asylum. In: IJCAI-17 Workshop on Explainable AI (XAI). vol. 36 (2017)
17. Mittelstadt, B., Russell, C., Wachter, S.: Explaining explanations in AI. arXiv:1811.01439 [cs] (Nov 2018). https://doi.org/10.1145/3287560.3287574, http://arxiv.org/abs/1811.01439, arXiv: 1811.01439

18. Mueller, S.T., Hoffman, R.R., Clancey, W., Emrey, A.: Explanation in human-AI systems: A literature meta-review synopsis of key ideas and publications and bibliography for explainable AI p. 204 (2019)
19. Swartout, W.R., Swartout, W.R.: Producing Explanations and Justifications of Expert Consulting Programs (1981)

A User Study: methods and results

The user study was conducted using an online platform[6]. We trained a K-nearest neighbors classifier on the Propublica COMPAS[7] database to predict the likelihood of recidivism. To keep the task simple, 6 features only were used. Decisions are sampled from the training data. We modified the model classification of half of them to create unjustified cases. Five types of assistance were randomly assigned: *absolute* claim, *relative* claim, counterfactual, k-neighbours and no assistance. The first two are derived directly from the framework, while the last three are part of the control group. Each user performed 10 tasks, 2 tasks per assistance type. In addition, users were asked to give their levels of confidence on a 5-Likert scale and an optional comment. We collected 160 completed tasks from 16 people (32 per assistance type) over 10 days of experiment. Users were mostly researchers and PhD students in the field of applied mathematics and computer science. No significant differences (p-value $> .05$) was found in the average number of correct answers between the different assistance types neither in the declarative confidence levels, although relative claims had best average values (see all result in table A).

Assistance type	N tasks	Mean	95 % conf. interval	Likert mean
Absolute claim	32	53 %	(35 %, 71 %)	3.5
Relative claim	32	63 %	(45 %, 80 %)	3.3
Counterfactual	32	56 %	(38 %, 74 %)	3.4
K-neighbours	32	56 %	(38 %, 74 %)	3.3
$\emptyset$	32	56 %	(38 %, 74 %)	3.3

Table 1. Results of the user study. Relative claims performed better than all other assistance although difference is not significant. There is no significant difference in Likert declarative confidences.

B Rule search algorithm

The objective of the rule search algorithm is to find the set of rules that defines the subset of the training data for which the average decision is the most significantly distinct (lowest p-value) from the average decision of the training subset defined by the claim. The set of rules is found with a greedy algorithm. At each step, the rule performing best according to an heuristic is selected until one of the two stopping criteria is met. At each step, the algorithm works as follow (see Algorithm 1):

1. select all training data satisfying the rules of the claim and the rule set.

[6] https://ml-advocate.inrialpes.fr/ (in French)
[7] https://github.com/propublica/compas-analysis

2. Enumeration of all possible rules. For all values appearing in the selected data, and for all operators ($\geq$, $\leq$, $=$ for numerical value or $=$ for categorical values) create a rule candidate.
3. select only candidate rules that the file of interest, that is the subject of the challenge, satisfies,
4. select Pareto optimal rules with respect to the coverage and ratio. High coverage and low ratio are used to search for evidence to support justifications and high coverage and high ratio are to search for evidence to support challenges (see Algorithm 2),
5. then select the rule with the lowest p-value.

The search continues until one of the two stopping criteria is met:

- ratio of the resulting rule is sufficiently low (to support justifications) or sufficiently high (to support challenges). See line 5-6 of Algorithm 1. In this case, the algorithm returns the current rule set.
- the difference of means after addition of the best rule candidate is not significant (p-value $> .2$ in the current implementation). See line 9-16 of Algorithm 1. In this case, the algorithm returns nothing.

Input: FileOfInterest, claim, trainingData, pvalThreshold
Result: Best significant set of rules, if any

1 initialization
2 $ruleSet \leftarrow \emptyset$
3 $d \leftarrow \{x \in trainingData, claim(x)\}$
4 $ratio \leftarrow size(\{x \in d, D(x) = 1\})/size(d)$
5 $threshold \leftarrow ratio/2$
`/*` $threshold \leftarrow (1 + ratio)/2$ `for supporting justifications */`
6 **while** $ratio > threshold$ **do**
  `/* Or` $ratio < threshold$ `if is used to find an evidence to support the challenge */`
7 $paretoFront \leftarrow$`ParetoOptimalRules(`d, $FileOfInterest$`)`
8 Select RuleBest with lowest p-value from paretoFront
9 $d_{RuleBest} \leftarrow \{x \in d, RuleBest(x)\}$
10 $pValBest \leftarrow T\text{-}test(\{D(x), x \in d_{RuleBest}\}, \{D(x), x \in d \setminus d_{RuleBest}\})$
11 **if** $pvalue < pvalThreshold$ **then**
12 Append $rule$ to $ruleSet$
13 $d \leftarrow \{x \in d, RuleBest(x)\}$
14 $ratio \leftarrow size(\{x \in d, D(x) = 1\})/size(d)$
15 **else**
16 Exit While and return nothing
17 **end**
18 **end**
19 **return** $RuleSet$

Algorithm 1: Rule Search Algorithm

```
Function ParetoOptimalRules(TrainingSubset, FileOfInterest):
    pareto ← ∅
    for candidate in all possible rules that FileOfInterest satisfies do
        d_candidate ← {x ∈ TrainingSubset, candidate(x)}
        cover ← size(d_candidate)
        ratio ← size({x ∈ d_candidate, D(x) = 1})/cover
        If no rule in pareto has bigger cover and smaller ratio than candidate
          then append candidate to pareto
        If any rule in pareto has smaller cover and bigger ratio than candidate
          then remove it from pareto
    end
    /* If used to find an evidence to support the challenge, ratio
       should be bigger as a condition to append and smaller as a
       condition to remove.                                          */
    return pareto
```

Algorithm 2: Function definition: ParetoOptimalRules