# Do We Need Another Explainable AI Method? Toward Unifying Post-hoc XAI Evaluation Methods into an Interactive and Multi-dimensional Benchmark

**Mohamed Karim Belaid**
IDIADA Fahrzeugtechnik GmbH
Munich, Germany
Karim.Belaid@idiada.com

**Eyke Hüllermeier**
University of Munich (LMU)
Munich, Germany
eyke@if.lmu.de

**Maximilian Rabus**
Dr. Ing. h.c. F. Porsche AG
Stuttgart, Germany
maximilian.rabus2@porsche.de

**Ralf Krestel**
ZBW - Leibniz Centre for Economics
& Kiel University
Kiel, Germany
r.krestel@zbw.eu

## Abstract

In recent years, Explainable AI (xAI) attracted a lot of attention as various countries turned explanations into a legal right. xAI allows for improving models beyond the accuracy metric by, e.g., debugging the learned pattern and demystifying the AI's behavior. The widespread use of xAI brought new challenges. On the one hand, the number of published xAI algorithms underwent a boom, and it became difficult for practitioners to select the right tool. On the other hand, some experiments did highlight how easy data scientists could misuse xAI algorithms and misinterpret their results. To tackle the issue of comparing and correctly using feature importance xAI algorithms, we propose Compare-xAI, a benchmark that unifies all exclusive and unitary evaluation methods applied to xAI algorithms. We propose a selection protocol to shortlist non-redundant unit tests from the literature, i.e., each targeting a specific problem in explaining a model. The benchmark encapsulates the complexity of evaluating xAI methods into a hierarchical scoring of three levels, namely, targeting three end-user groups: researchers, practitioners, and laymen in xAI. The most detailed level provides one score per unit test. The second level regroups tests into five categories (fidelity, fragility, stability, simplicity, and stress tests). The last level is the aggregated comprehensibility score, which encapsulates the ease of correctly interpreting the algorithm's output in one easy to compare value. Compare-xAI's interactive user interface helps mitigate errors in interpreting xAI results by quickly listing the recommended xAI solutions for each ML task and their current limitations. The benchmark is made available at `https://karim-53.github.io/cxAI/`

## 1 Introduction

xAI algorithms are a set of approaches toward understanding black-box models. In recent years, xAI algorithms helped debug manifold issues in ML models, such as exposing underlying wrong patterns in classifying objects [1] or highlighting inequality and bias in decisions [2]. Moreover, given its essential impact on society, legislation in several countries now includes the "Right to explanation" [3] fulfilled by the various xAI tools available in the literature. It is indeed difficult to define the best xAI solution given the number of known evaluation metrics. Moreover, the long evolutionary history of

Preprint. Under review.

specific xAI methods makes it even more difficult to evaluate each version. The Shapley values are an excellent example of this challenge. Sundararajan et al. did state that "…the functional forms of the Shapley value…are sufficiently complex as to prevent direct understanding…" [4]. Indeed, going through the theoretical background of Shapley values [5], its multiple approximations [6, 7], generalizations [8, 4] and final implementations [9, 10] adapted to the AI field might mislead the end-user on the capability of the available tools.

Consequently, data scientists face considerable difficulties in accurately evaluating each xAI algorithm and remaining up-to-date on its evolution. This issue yields a clearly visible symptom known as the illusion of explanatory depth [11] in interpreting xAI results [12] and it has been confirmed that data scientists are prone to misuse interpretability tools [13]. Many researchers did address this question by stressing the importance of structuring and documenting xAI algorithms [14, 15], i.e., by highlighting the target end-users of the algorithm, its capability, limitations, and vulnerabilities. Finally, they recommend using quantitative metrics to make claims about explainability. Section 2 analyses further the related literature. Given the unsolved burden of evaluating and correctly interpreting xAI results, we propose Compare-xAI that mitigates these two issues (benchmark xAI results and the illusion of explanatory depth during the interpretation of results) by addressing three research questions:

1. How to select exclusive unit tests from those proposed in the literature?
2. How to score xAI algorithms in a simple way despite the multitude of evaluation dimensions?
3. How to reduce data scientists' potential misuse of the xAI algorithms?

The stated questions are resolved as follows: In Section 3, we propose a benchmark implementation easily scalable to new xAI algorithms and new unit tests. In Section 3.1, we propose a selection protocol for the quantitative evaluation of xAI algorithms. It is then applied to shortlist a selection of exclusive unit tests, each targeting a distinct problem. In Section 3.2, we explain the experiments' protocol to mimic the end-user's behavior. In Section 3.3, we propose an intuitive scoring method that scales in detail with the level of expertise of the data scientist: Layman data scientists are invited to manipulate one global score named comprehensibility. Practitioners are invited to compare xAI algorithms given five scores representing five subcategories of the comprehensibility metric. Finally, researchers are invited to study the detailed report (one score per unit test). In Section 4, we propose a user interface that encapsulates the benchmark's results. We seek to minimize the potential misuse of xAI algorithms by offering quick access to the limitation of each xAI algorithm. Finally, Section 5 is dedicated to the theoretical and practical limitations of the benchmark.

## 2 Related Work

This section is a survey for xAI evaluation methods. It contains examples contrasting the difference between unit tests and end-to-end tests (E2E). Following that, we examine some attempts to regroup them into surveys or benchmarks.

### 2.1 xAI Evaluation Methods: Unit Tests vs. E2E Tests

Research papers in the xAI field often propose a new method along with a set of tests that outline the contrast between former work and their contribution.

**Unit tests.** Clear unit tests usually exploit tabular synthetic data and few input features, e.g., considering the "cough and fever" unit test [9], The xAI algorithm is expected to detect symmetry between the two features. Simple examples showcase the undeniable limit of certain xAI methods. Nevertheless, specific unit tests could use real-world data. A good example is the MNIST dataset [16] used as a counterexample for the dummy axiom [17]: Since edge pixels are always black, a multi-layer perceptron will learn not to rely on these constant pixels. As a consequence, the xAI algorithm should confirm that the AI does not use these pixels. Papers proposing new xAI methods remain too short to list all known unit tests. Furthermore, some of the highlighted issues might be fixed without any publication.

**end-to-end tests (E2E).** E2E xAI tests evaluate real-word tasks and demonstrate the robustness of the xAI algorithm against multiple problems at once (noisy and correlated data, large input size, etc.).

In addition, they are used to claim the potential broad usage of one xAI method, e.g., showing the portability across recommendation tasks [18].

**Navigating the ocean of tests remains itself a huge challenge.** First, many examples in the literature are E2E tests which makes comparison between xAI algorithms complex. Second, tests could be redundant to emphasize the frequent occurrence of an issue, e.g., testing interaction detection with different transparent models [18]. Third, researchers could argue the correctness of specific unit tests' ground truth, e.g., causal explanation of the Shapley values [19] has been considered false in certain research [4].

**Unit test categories.** Unit tests do not only evaluate the fidelity of the xAI output to the AI model but can also evaluate the fragility of the xAI algorithm against adversarial attacks. Indeed, attackers can corrupt deployed models to influence their explanations without affecting individual predictions or overall accuracy [20, 21, 22, 23]. Please refer to Table 1 for the complete set of categories.

Given the tremendous amount of xAI algorithms and dedicated metrics, surveys [24, 25, 26, 27] have trouble providing an in-depth analysis of each algorithm and cannot cope with ongoing implementation updates. Nevertheless, Molnar's online book distinguishes itself with a continuously updated survey about xAI [28]. The initiative of a real-time survey faced great success and acceptance from the data science community.

### 2.2 Benchmark for xAI Algorithms

There are specialized benchmarks in the literature, like the SVEA benchmark [29]. The latter focuses on computer vision tasks and proposes faster evaluations based on the small mnist-1D dataset [30]. Another benchmark utilizes exclusively human evaluation to assess xAI algorithms on real-world tasks [31]. On the one hand, benchmarking using computer vision and NLP models permits to measure the real success of an xAI tool in helping end-users even though human evaluation could be considered subjective and more costly to obtain. On the other hand, evaluation using real-world tasks does not allow debugging the xAI algorithm, i.e., two algorithms might fail to explain one black-box model for two different reasons.

xAI-Bench [32] evaluates each xAI algorithm on five metrics. Faithfulness measures the Pearson correlation between the feature importance and the approximate marginal contribution of each feature. Of course, one could argue that the ground truth explanation of a model could be slightly different from the marginal contribution of each feature on the observed dataset. The same argument holds for the monotonicity, infidelity, and GT-Shapley metrics. They define a ground truth output, that is, a "better" xAI algorithm is the one outputting a result that is closer to the ground truth. In contrast, unit tests discussed in previous paragraphs evaluate the correctness of the output using a pattern (not an exact ground truth). This paper focuses on unit tests using patterns as evaluation methods. The fifth metric used in xAI-Bench is remove-and-retrain (ROAR). It involves a re-evaluation of the model, which could itself alter the evaluation. Another critical factor affecting the scores and the ranking of the algorithms is the data distribution. The authors did circumvent the issue by testing on different distributions (multivariate Gaussian, Gaussian mixture, and multinomial feature distributions). However, it remains difficult to decide if the algorithm is failing this specific test or if it is generally sensitive to the data distribution. xAI-Bench is an excellent initiative to benchmark the correctness of an xAI algorithm, except that it does not allow a clear debugging and does not propose any final ranking of the xAI algorithm to help practitioners and laymen quickly pick the right tool. Following the analysis of related work, Section 3 details how our proposed benchmark addresses the highlighted issues.

## 3 Compare-xAI

Compare-xAI is a quantitative benchmark based solely on unit tests. Compare-xAI is able to evaluate any new xAI algorithm or index any new unit test. Current proof-of-concept evaluates more than 16 post-hoc xAI algorithms on more than 22 unit tests. Compare-xAI outputs a series of scores that allows a multi-dimensional analysis of comparable xAI algorithms. Figure 1 illustrates Compare-xAI as a pipeline with three added values: First, we collect the known unit tests reported in related literature and filter them according to a clear protocol stated in Section 3.1. As a second step, each algorithm is tested automatically on each of the collected unit tests. Experiments follow a protocol detailed in Section 3.2. Each experiment results in one score ranging from 0 (failing) to 1 (succeeding). Compare-xAI has the exclusive advantage of reporting an intermediate score (between 0 and 1) if the algorithm is partially failing the test. Obtained raw results describe an xAI algorithm by 22 scores, and it is not easy, at this point, to compare two algorithms. Therefore, raw results are aggregated into a hierarchical scoring method, see Section 3.3.

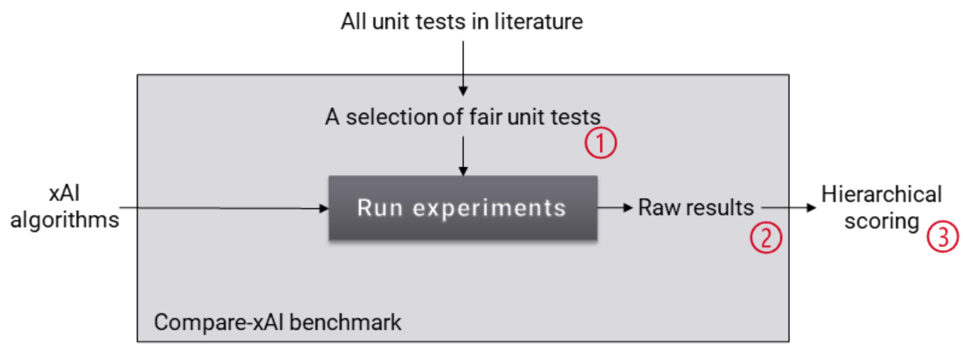


Figure 1: Compare-xAI's pipeline

### 3.1 Unit Tests Selection Protocol

A unit test consists of an environment (a synthetic or a real dataset plus a model to explain) and an assertion (e.g., an xAI algorithm is expected to detect symmetric input features, see Section 2.1 for more examples). Section 2.1 highlighted the diversity of tests used in the literature and the importance of filtering inadequate ones. We propose the following rules to ensure a multi-dimensional evaluation of all post-hoc xAI algorithms:

**Unit test, not E2E tests.** Selected unit tests should identify and debug a clear problem within an xAI algorithm. For example, one could say that "this xAI algorithm is failing this test because it is sensitive to correlated input features". This rule ensures that an algorithm will not fail multiple tests for the same reason, and it allows researchers quickly debug the xAI algorithm. Table 1 regroups a non-exhaustive set of unitary problems that could alter the algorithm's output and let it fail in explaining the model correctly.

**Undebatable unit test.** If two opposite explanations could be considered correct given different setups, then Compare-xAI would not include such a unit test. A popular example is the causal explanation of the SHAP values [33].

**No exact ground truth explanation.** A unit test contains a scoring function that compares the xAI output to the expected output. The expected output could be an exact set of values, e.g., GT-Shapley's expected output is the Shapley values [32]. The expected output could also be a pattern, e.g., given one specific model, feature A's importance should be the highest, or negative, or equal to feature B's. Compare-xAI's shortlisted unit tests rely only on patterns. Protecting this degree of freedom makes this benchmark compatible with different xAI approaches. For example, in the case of an adversarial attack unit test, it is expected that the ranking of the feature importance does not get affected by corrupted models, regardless of the exact ranking proposed. Another good example is the unit test that assesses the symmetry axiom. Its evaluation function verifies the equality between the feature importance values without having an exact value as a reference.

**Non-redundant unit tests.** There exists no unit test $i$ and $j$ such that for all algorithms (Algorithm $A$ succeeds in test $i$ iff algorithm $A$ succeeds in test $j$).

**Only unit tests proposed in the related work.** The first iteration of the Compare-xAI benchmark is limited to the unit tests reported in former research papers, as many have been presented, discussed, and heavily criticized in the literature. Compare-xAI takes advantage of this extensive research work to build a consistent benchmark. Identifying problems and proposing unit tests to cover them for the first time seems to be a rewarding field of future research.

Table 1: Samples from the shortlisted unit tests.

| Category | Grouped unit tests |
|---|---|
| Fidelity | **Does the algorithm's output reflect the underlying model (aka faithfulness [35])?**<br>• Counterexample for the symmetry axiom [9].<br>• Test whether features of different importance are represented correctly [9].<br>• Test the detection of feature interaction based on mathematical terms [18].<br>• Effect of feature product on local explanations [19].<br>• Test if one-hot encoded features are explained correctly [36].<br>• Test if main terms and interaction terms are evaluated correctly [37]. |
| Fragility | **Could the xAI algorithm be easily manipulated?**<br>• Adversarial attacks can exploit feature perturbation-based xAI algorithms as a vulnerability to lower the importance of specific features [22]. |
| Stability | **Is the algorithm's output too sensitive to slight changes in the data or model?**<br>• Effect of data distribution: Statistical dependence, non-uniform distribution [33]<br>• Effect of feature correlations [32, 13]<br>• Effect of noise in the dataset [4]<br>• Implementation invariance axiom |
| Simplicity | **Can users look at the explanation and easily reason about the model's behavior?**<br>• Counterexample of the dummy axiom [4]<br>• Counterexample of the linearity axiom [4] |
| Stress | **Can the algorithm explain models trained on big data?**<br>• Test if the xAI algorithm is sensitive to a high number of word tokens (NLP task) [7].<br>• Detect dummy pixels in the MNIST dataset [17].<br>• Effect of data sparsity [4]. |
| *Other* | **Remaining metrics are not integrated into the hierarchical scoring system albeit reported in the final dataset.**<br>• Portability [28] measures the diversity of models' implementation that an xAI algorithm can explain. Portability is tested implicitly by different unit tests as each one implements a different model/dataset.<br>• The relative execution time is an essential factor in choosing an algorithm, and it is mainly influenced by the stress tests. |

**Categorizing unit tests.** In order to cover a large variety of problems, we propose to categorize shortlisted unit tests into five common groups [34], see Table 1.

### 3.2 Experiments Protocol

An experiment takes one unit test and one xAI algorithm. First, the unit test environment is initialized by loading the data and training the model. Then the xAI algorithm is asked to explain the model (Globally or for specific data points). Finally, the explanation is compared to the correct answer, and one final score, a real number between 0 and 1, is returned.

It might seem to end-users that the stated patterns in Table 1 are self-evident and verified for every xAI algorithm. As a matter of fact, the public availability and wide usage of particular xAI tools "swayed several participants to trust the tools without fully understanding them" [13]. For this reason, we score each xAI algorithm by following the most common usage of the xAI algorithms. This policy induces the following rules:

**Experiments will only be run once.** Recent research revealed "a misalignment between data scientists' understanding of interpretability tools and these tools' intended use. Participants misused the tools (either over- or under-used them)." [13]. Compare-xAI is intended for this former group, that is, data scientists not running complementary experiments or repeating the same experiment to test the effect of the noise. Targeting the remaining data scientist, i.e., experts with advanced knowledge of the tools' limits, is left for future work.

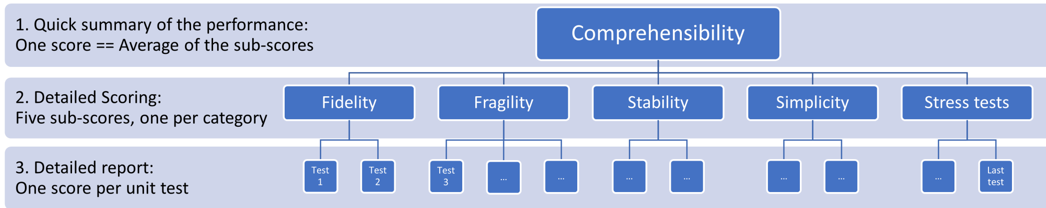


Figure 2: Hierarchical scoring

**No fixed seed for random number generators.** For the same reasons stated above, the seed is not fixed. Shortlisted unit tests' underlying noise does not hinder any xAI method from correctly explaining a model, i.e., initialized models always learn the tested patterns independently of the seed.

**No parameter tuning.** Compare-xAI evaluates algorithms using their default parameters for all tests. Nevertheless, certain xAI algorithm adapts their parameters internally given each task by relying on the model's structure, the dataset size, and the ML task. Around half of the indexed algorithms have at least one binary parameter, and the performance of some would vary with parameter tuning. As there are no precise methods or public tools to fine-tune parameters of xAI algorithms, we consider the usage of default parameters, by end-users, to be a common misuse of the xAI algorithm. It is important that Compare-xAI reproduces this behavior in order to calculate the comprehensibility score in practice, that is, the ease of correctly interpreting the algorithm's output, considering common practice.

### 3.3 Scoring Protocol

Compare-xAI's unit tests result in a set of scores per algorithm, which we call "raw" results. We define the dominance property in order to compare and sort algorithms by performance based on these raw results.

**Dominance property.** xAI algorithm A dominates B iff (1) A's score for each test is greater than or equal to B's score; (2) for each test, A's execution time is less than or equal to B's execution time; (3) B's supported models are a subset of A's (see the portability metric definition in Table 1); and (4) B's supported output explanations are a subset of A's.

As a consequence, sorting xAI algorithms by performance is a multi-metric ranking problem. Comparing even two algorithms using this property is impossible, especially with a considerable number of unit tests. This formulation of the problem outlines the difficulty faced by practitioners/laymen who are looking for a quick explanation of their models but cannot decide on which xAI algorithm to pick. This problem is addressed by proposing a relaxation of the dominance property.

First, let us suppose that each end-user is comparing a subset of xAI algorithms which are all able to explain the model of his/her interest, and they all offer the type of explanation required by the end-user (feature importance, attribution, interaction, etc.). On this account, requirements (3) and (4) might be skipped. Second, looking at the overall distribution of the scores, There is no state-of-the-art algorithm that succeeds in all tests. Thus, we opt to promote the "on-average" better xAI algorithm. Figure 2 explains the hierarchical scoring proposed to the end-user to simplify the comparison between xAI algorithms. Level three, the last level in the hierarchical scoring, represents the most detailed report (one score per unit test) used in the dominance ranking method. To simplify it, level two regroups tests per category to propose five scores described in Table 1. Finally, the first level aggregates the scores into one value, which we named the comprehensibility score. Comprehensibility is defined in the literature as "how much effort is needed for a human to interpret a model correctly?" [38]. We quantify this metric as the average over the five subcategories of the second level. Therefore, comparing xAI algorithms become easier using the following relaxed dominance property:

**Relaxed dominance property.** xAI algorithm A explains better than B on average iff (1) A's comprehensibility score is not lower than B's and (2) A's average execution time is not higher than B's.

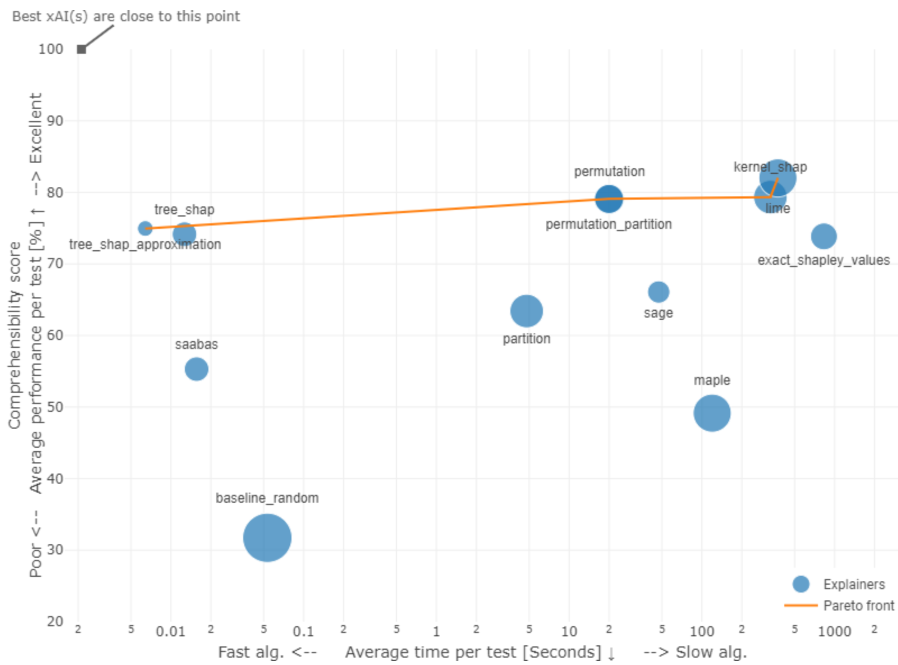


Figure 3: Global overview of the benchmark

Finally, the evaluation's result is reduced to the comprehensibility score and the average execution time. Remains the dilemma of choosing the fastest algorithm or the most "comprehensible" one from the Pareto front. Depending on his/her level of expertise, the end-user is invited to consult any of the three scoring methods made available via the web user interface.

## 4 Visualization of the Benchmark

For the proof of concept, we consider a small set of popular xAI methods, implement a user interface to easily explore the benchmark results, and make the results publicly available[1]. Figures 3 and 4 do not represent a final benchmark as the set of unit tests and xAI algorithms are constantly updated. In this section, the demonstration will focus on feature importance. Nevertheless, the benchmark remains adaptable to many forms of post-hoc xAI methods.

Figure 3 summarizes the performance of the considered set of xAI methods. The best algorithm has the lowest execution time, highest score, and highest portability (bigger dot size). The Pareto front regroups the closest algorithms to the perfect one. End-users could use the filters on the web interface to describe a specific use case: Figure 4 is restricted to the set of model-agnostic xAI algorithms that output (at least) global feature importance. Available filters help the end-user quickly and accurately navigate the massive amount of undocumented properties of available xAI tools.

At this stage, the end-user picks an xAI algorithm that matches his expected performance and execution time requirements. The detailed report is accessible by clicking on the blue dot representing the algorithm: the end-user gets access to information about the supported AI models, the output of the xAI algorithm, additional information required by the xAI algorithm to run correctly, and essentially the score obtained on each unit test. The detailed report helps the end-user quickly understand the limit of the xAI algorithm before using it. Finally, the end-user can compare multiple xAI algorithms given a set of unit tests that reflect his usage of AI. Please keep in mind that selected screenshots are for demonstration purposes only and we are not going to discuss individual scores, but only the benchmark itself.

[1] `https://karim-53.github.io/cxAI/`

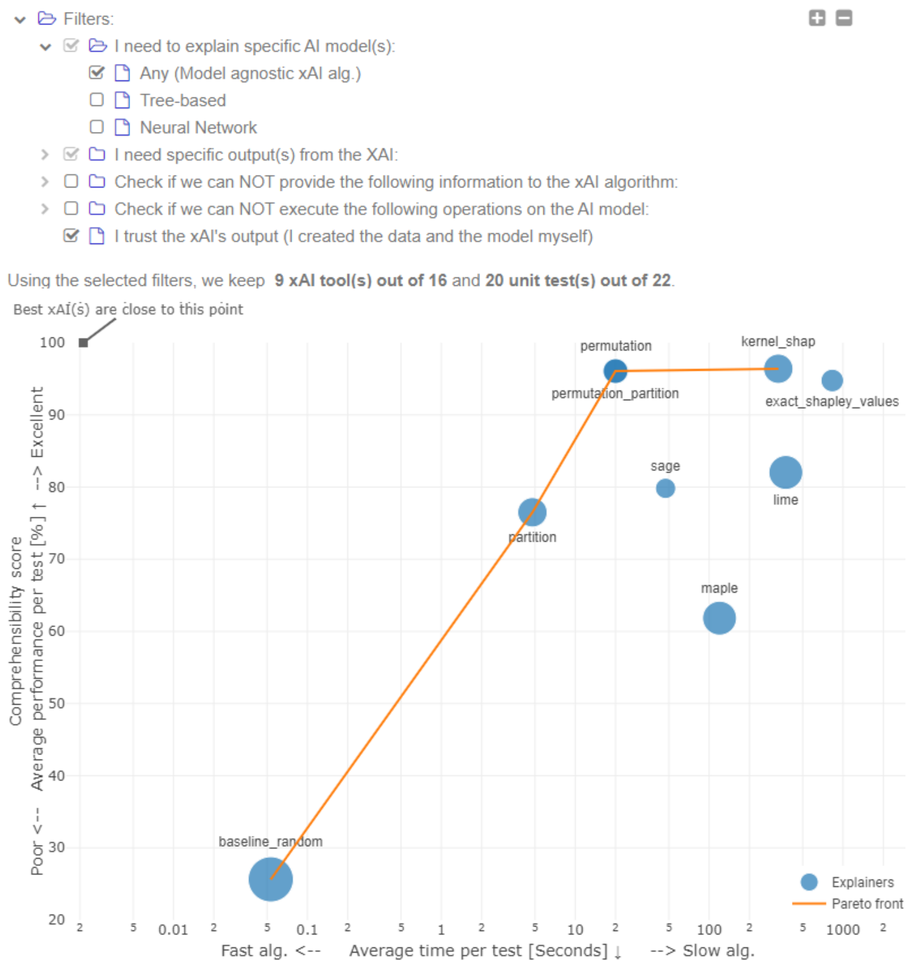


Figure 4: Benchmark of model-agnostic xAI algorithms outputting feature importance

## 5 Limitations and Future Work

Compare-xAI's weaknesses are classified into design-related and implementation-related limitations.

**Design-related limitations.** Compare-xAI is a benchmark made exclusively of quantitative metrics. It is objective as it does not include tests based on human evaluation. A common example from is the study of the human mental model like the investigation of users' preferred explanation style [39]). Another example is the study of the information overload, e.g., xAI's additional output information like the confidence interval [17]. Mainly, empirical studies are challenging to quantify [40] and integrate into the comprehensibility score. These and other non-quantifiable advantages/disadvantages will be included in the description of the xAI algorithm, in the future.

**implementation-related limitations.** The provided proof of concept includes, for now, 22 unit tests. Currently, none of them cover RL, GAN, or unsupervised learning tasks. The tested form of output is also limited to feature importance (global explanation). Testing feature interaction is still under development.

Despite the stated limitations, Compare-xAI should fulfill its primary objectives: first, helping laymen pick the right xAI method, and second, helping researchers, practitioners, and laymen avoid common mistakes in interpreting its output.

## 6 Conclusion

Explaining AI is a delicate task, and some end-users are prone to misuse dedicated tools [13]. We propose Compare-xAI a unified benchmark indexing +16 post-hoc xAI algorithms, +22 unit tests, and +40 research papers. Compare-xAI reproduces experiments following a selection protocol that highlights the contrast between the theoretical claims of the authors in a paper and the practical implementation offered to the end-user. Selected unit tests measure diverse properties. The authors did not create any xAI algorithm. Therefore, there is no conflict of interest. Compare-xAI proposes to deliver the results using an interactive interface as a solution to mitigate human errors in interpreting xAI outputs by making the limits of each method transparent. Compare-xAI proposes a simple and intuitive scoring method that efficiently absorbs the massive quantity of xAI-related papers. Finally, Compare-xAI proposes a partial sorting of the xAI methods, toward unifying post-hoc xAI evaluation methods into an interactive and multi-dimensional benchmark.

## Broader Impact

Compare-xAI is a benchmark with multiple use-cases. It can be seen as a debugging tool for individual xAI algorithms but simultaneously as a global benchmark. Even if Compare-xAI does not offer a total sorting per performance, still it separates comparable algorithms into the Pareto front and the rest. Compare-xAI allows practitioners to quickly and correctly filter xAI algorithms given their needs and to outline the limitations of the selected ones. The end-user, now aware of the limit of the xAI algorithm, would not over-trust the algorithm's output and would avoid common mistakes in explaining a model. On the other hand, Compare-xAI allows researchers to access a detailed scoring and to answer specific questions such as "In which case does this xAI algorithm fail?", "Is it the only one to solve this issue?", "What kind of cases are still not covered by any xAI algorithm?" etc. Compare-xAI continuously re-evaluates indexed xAI algorithms to keep an updated benchmark of the state-of-the-art. Finally, Compare-xAI is more than a benchmark: it is a comprehensive and standardized related work analysis, while it also works as an evaluation method for new research papers in xAI.

## Acknowledgments

We thank Dorra El Mekki, Jonas Steinhäuser, Tim Donkiewicz, Marius Daehling, Maximilian Muschalik, Patrick Kolpaczki, and Michael Rapp for their thoughtful feedback on earlier iterations of this work.

## References

[1] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. " why should i trust you?" explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, pages 1135–1144, 2016.

[2] Julia Dressel and Hany Farid. The accuracy, fairness, and limits of predicting recidivism. *Science advances*, 4(1):eaao5580, 2018.

[3] Bryce Goodman and Seth Flaxman. European union regulations on algorithmic decision-making and a "right to explanation". *AI magazine*, 38(3):50–57, 2017.

[4] Mukund Sundararajan and Amir Najmi. The many shapley values for model explanation. In *International conference on machine learning*, pages 9269–9278. PMLR, 2020.

[5] LS Shapley. Quota solutions op n-person games1. *Edited by Emil Artin and Marston Morse*, page 343, 1953.

[6] Erik Štrumbelj and Igor Kononenko. Explaining prediction models and individual predictions with feature contributions. *Knowledge and information systems*, 41(3):647–665, 2014.

[7] Scott M Lundberg and Su-In Lee. A unified approach to interpreting model predictions. *Advances in neural information processing systems*, 30, 2017.

[8] Mukund Sundararajan, Ankur Taly, and Qiqi Yan. Axiomatic attribution for deep networks. In *International conference on machine learning*, pages 3319–3328. PMLR, 2017.

[9] Scott M Lundberg, Gabriel G Erion, and Su-In Lee. Consistent individualized feature attribution for tree ensembles. *arXiv preprint arXiv:1802.03888*, 2018.

[10] Mateusz Staniak and Przemyslaw Biecek. Explanations of model predictions with live and breakdown packages. *arXiv preprint arXiv:1804.01955*, 2018.

[11] Leonid Rozenblit and Frank Keil. The misunderstood limits of folk science: An illusion of explanatory depth. *Cognitive science*, 26(5):521–562, 2002.

[12] Michael Chromik, Malin Eiband, Felicitas Buchner, Adrian Krüger, and Andreas Butz. I think i get your point, ai! the illusion of explanatory depth in explainable ai. In *26th International Conference on Intelligent User Interfaces*, pages 307–317, 2021.

[13] Harmanpreet Kaur, Harsha Nori, Samuel Jenkins, Rich Caruana, Hanna Wallach, and Jennifer Wortman Vaughan. Interpreting interpretability: understanding data scientists' use of interpretability tools for machine learning. In *Proceedings of the 2020 CHI conference on human factors in computing systems*, pages 1–14, 2020.

[14] Matthew L Leavitt and Ari Morcos. Towards falsifiable interpretability research. *arXiv preprint arXiv:2010.12016*, 2020.

[15] Michael Kearns and Aaron Roth. *The ethical algorithm: The science of socially aware algorithm design*. Oxford University Press, 2019.

[16] Yann LeCun, Corinna Cortes, and CJ Burges. Mnist handwritten digit database. *ATT Labs [Online]. Available: http://yann.lecun.com/exdb/mnist*, 2, 2010.

[17] Ian Covert, Scott M Lundberg, and Su-In Lee. Understanding global feature contributions with additive importance measures. *Advances in Neural Information Processing Systems*, 33:17212–17223, 2020.

[18] Michael Tsang, Sirisha Rambhatla, and Yan Liu. How does this interaction affect me? interpretable attribution for feature interactions. *Advances in neural information processing systems*, 33:6147–6159, 2020.

[19] I Elizabeth Kumar, Suresh Venkatasubramanian, Carlos Scheidegger, and Sorelle Friedler. Problems with shapley-value-based explanations as feature importance measures. In *International Conference on Machine Learning*, pages 5491–5500. PMLR, 2020.

[20] Amirata Ghorbani, Abubakar Abid, and James Zou. Interpretation of neural networks is fragile. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, pages 3681–3688, 2019.

[21] Juyeon Heo, Sunghwan Joo, and Taesup Moon. Fooling neural network interpretations via adversarial model manipulation. *Advances in Neural Information Processing Systems*, 32, 2019.

[22] Himabindu Lakkaraju and Osbert Bastani. "" how do i fool you?"" manipulating user trust via misleading black box explanations. In *Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society*, pages 79–85, 2020.

[23] Dylan Slack, Anna Hilgard, Himabindu Lakkaraju, and Sameer Singh. Counterfactual explanations can be manipulated. *Advances in Neural Information Processing Systems*, 34, 2021.

[24] Sina Mohseni, Niloofar Zarei, and Eric D Ragan. A multidisciplinary survey and framework for design and evaluation of explainable ai systems. *arXiv preprint arXiv:1811.11839*, 2018.

[25] Michael Tsang, James Enouen, and Yan Liu. Interpretable artificial intelligence through the lens of feature interaction. *arXiv preprint arXiv:2103.03103*, 2021.

[26] Plamen P Angelov, Eduardo A Soares, Richard Jiang, Nicholas I Arnold, and Peter M Atkinson. Explainable artificial intelligence: an analytical review. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 11(5):e1424, 2021.

[27] Benedek Rozemberczki, Lauren Watson, Péter Bayer, Hao-Tsung Yang, Olivér Kiss, Sebastian Nilsson, and Rik Sarkar. The shapley value in machine learning. *arXiv preprint arXiv:2202.05594*, 2022.

[28] Christoph Molnar. *Interpretable machine learning*. Lulu. com, 2020.

[29] Sam Sattarzadeh, Mahesh Sudhakar, and Konstantinos N Plataniotis. Svea: A small-scale benchmark for validating the usability of post-hoc explainable ai solutions in image and signal recognition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4158–4167, 2021.

[30] Sam Greydanus. Scaling down deep learning. *arXiv preprint arXiv:2011.14439*, 2020.

[31] Sina Mohseni, Jeremy E Block, and Eric D Ragan. Quantitative evaluation of machine learning explanations: A human-grounded benchmark. *arXiv preprint arXiv:1801.05075*, 2020.

[32] Yang Liu, Sujay Khandagale, Colin White, and Willie Neiswanger. Synthetic benchmarks for scientific research in explainable machine learning. *arXiv preprint arXiv:2106.12543*, 2021.

[33] Dominik Janzing, Lenon Minorics, and Patrick Bl¨obaum. Feature relevance quantification in explainable ai: A causal problem. In *International Conference on artificial intelligence and statistics*, pages 2907–2916. PMLR, 2020.

[34] Himabindu Lakkaraju, Ece Kamar, Rich Caruana, and Jure Leskovec. Faithful and customizable explanations of black box models. In *Proceedings of the 2019 AAAI/ACM Conference on AI, Ethics, and Society*, pages 131–138, 2019.

[35] David Alvarez Melis and Tommi Jaakkola. Towards robust interpretability with self-explaining neural networks. *Advances in neural information processing systems*, 31, 2018.

[36] Salim I Amoukou, Tangi Salaün, and Nicolas Brunel. Accurate shapley values for explaining tree-based models. In *International Conference on Artificial Intelligence and Statistics*, pages 2448–2465. PMLR, 2022.

[37] Andreas Tilevik. *Two-way anova - the basics*. payhip.com, 2021.

[38] Freddy Lecue, Krishna Gade, Sahin Geyik, Kenthapadi Krishnaram, Varun Mithal, Ankur Taly, Riccardo Guidotti, and Pasquale Minervini. Explainable ai: Foundations, industrial applications, practical challenges, and lessons learned. In *Proceedings of the AAAI Conference*, pages 131–138, 2020.

[39] Jeya Vikranth Jeyakumar, Joseph Noor, Yu-Hsi Cheng, Luis Garcia, and Mani Srivastava. How can i explain this to you? an empirical study of deep neural network explanation methods. *Advances in Neural Information Processing Systems*, 33:4211–4222, 2020.

[40] Tim Miller. Explanation in artificial intelligence: Insights from the social sciences. *Artificial intelligence*, 267:1–38, 2019.

## Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes] Compare-xAI is a benchmark for feature importance xAI algorithms. It includes a user interface that helps data scientists quickly run through the scores.
   (b) Did you describe the limitations of your work? [Yes] see 5
   (c) Did you discuss any potential negative societal impacts of your work? [No] We discuss the societal impacts in the "Broader impact" section, but we would not qualify it as negative.
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]

2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [N/A] There are no theoretical results.
   (b) Did you include complete proofs of all theoretical results? [N/A]
3. If you ran experiments (e.g., for benchmarks)...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [Yes] Our code is available at `https://github.com/Karim-53/Compare-xAI` and the main readme contains the instructions on how to re-run experiments.
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes] Each unit test has its own training details defined in its class.
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [No] The protocol of the benchmark reproduces a specific behavior of a layman: results are reported after only one run. We aim to target the second group of end-users (those who run experiments multiple times) in future work by elaborating more on what to report (min, max, or average) and how to keep an equitable comparison between stable and noisy algorithms. Nevertheless, the benchmark stays stable because it averages over multiple unit tests.
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [Yes] See `https://github.com/Karim-53/Compare-xAI/blob/main/README.md#23-computing-ressouces=`
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [Yes] See `https://github.com/Karim-53/Compare-xAI/blob/main/README.md#reference=`
   (b) Did you mention the license of the assets? [No]
   (c) Did you include any new assets either in the supplemental material or as a URL? [Yes] `https://github.com/Karim-53/Compare-xAI/blob/main/LICENSE`
   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [N/A] We do not survey people.
   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [N/A]
5. If you used crowdsourcing or conducted research with human subjects...
   (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A] No crowdsourcing.
   (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A] No crowdsourcing.
   (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A] No crowdsourcing.